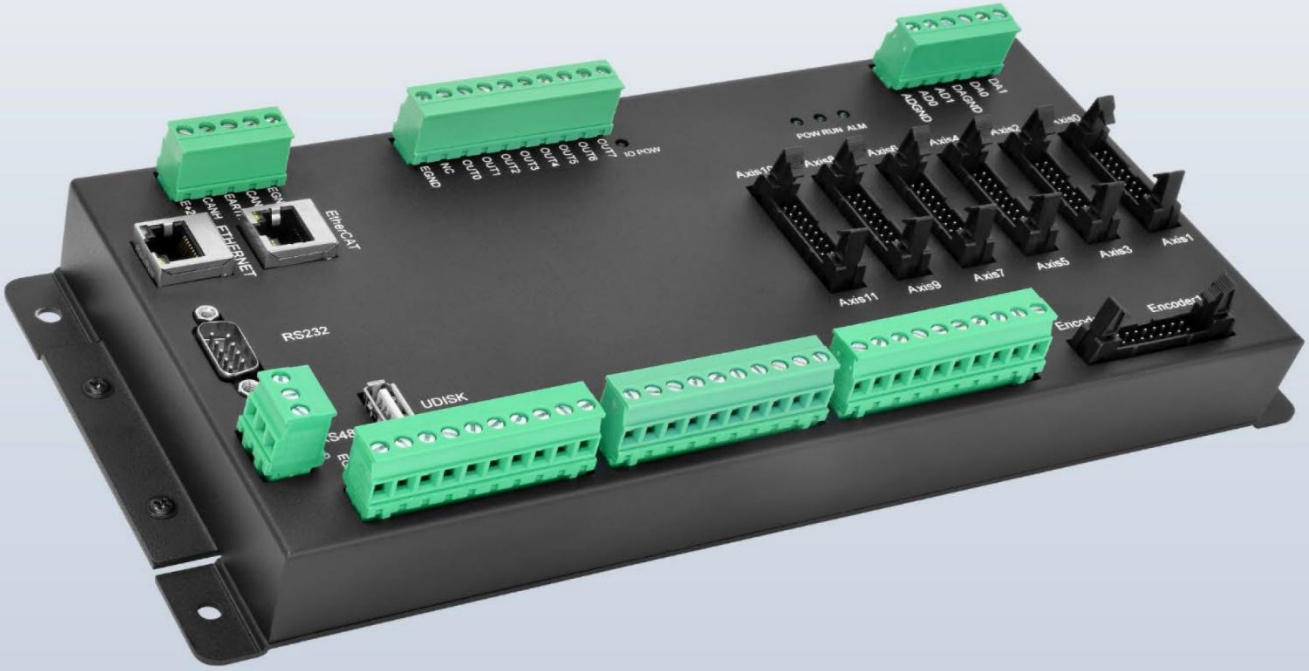
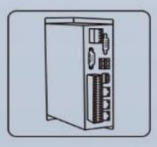
Zmotion®
正运动技术

做更好用的运动控制，智造美好生活
Better Motion Control, Smarter Life

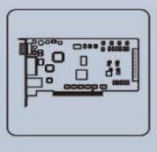
EtherCAT & Pulse Motion Controller

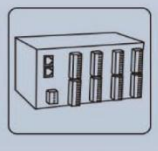
ZMC212BE

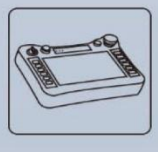



Vision Motion Controller

Motion Controller


Motion Control Card


IO Expansion Module


HMI

Statement

Thank you for choosing our Zmotion products. Please be sure to read this manual carefully before use so that you can use this product correctly and safely. Zmotion is not responsible for any direct or indirect losses caused by the use of this product.

The copyright of this manual belongs to Shenzhen Zmotion Technology Co., Ltd. And reproduction, translation, and plagiarism of any content in this manual in any form is strictly prohibited without the written permission of Zmotion.

The information in this manual is for reference only. Due to design improvements and other reasons, Zmotion reserves the right of final interpretation of this information! Contents are subject to change without prior notice!

➢ Notes

In order to prevent possible harm and damage caused by incorrect use of this product, the following instructions are given on matters that must be observed.

■ Danger

Do not use it in places with water, corrosive or flammable gases, or near flammable substances.	May cause electric shock, fire, damage, etc.
When installing or disassembling, make sure the product is powered off.	
Cables should be connected securely, and exposed parts that are energized must be insulated by insulators.	
Wiring work must be performed by professionals.	

■ Notes

It should be installed within the specified environmental range.	May cause damage, mis-operation, etc.
Make sure there are no foreign objects on the product hardware circuit board.	
After installation, the product and the mounting bracket should be tight and firm.	
After installation, at least 2-3cm should be left between the product and surrounding components for ventilation and replacement.	
Never disassemble, modify, or repair it by yourself.	

Content

Chapter I Production Information.....5
1.1. Product Information.....5
1.2. Function Features.....5
1.3. System Frame.....6
1.4. Hardware Installment.....7
Chapter II Product Specification.....9
2.1. Basic Specification.....9
2.2. Interface Definition.....10
2.3. Work Environment.....11
Chapter III Wiring, Communication Configuration.....12
3.1. Power Input.....12
3.1.1. Power Specification.....12
3.1.2. CAN Communication Specification & Wiring.....12
3.1.3. Basic Usage Method.....14
3.2. RS232 Serial Port.....15
3.2.1. RS232 Communication Specification & Wiring.....16
3.2.2. Basic Usage Method.....17
3.3. RS485 Serial Port.....18
3.3.1. RS485 Communication Specification & Wiring.....18
3.3.2. Basic Usage Method.....20
3.4. IN Digital Input & High-Speed Latch Port.....21
3.4.1. Digital Input Specification & Wiring.....22
3.4.2. Basic Usage Method.....23
3.5. OUT Digital Output.....24
3.5.1. Digital Output Specification & Wiring.....24
3.5.2. Basic Usage Method.....25
3.6. AD/DA Analog Input & Output.....26
3.6.1. Analog Input / Output Specification & Wiring.....26
3.6.2. Basic Usage Method.....27
3.7. U Disk.....28
3.8. ETHERNET.....29

3.9. EtherCAT Bus Interface 30
3.10. ENCODER Interface.....32
3.10.1. ENCODER Interface Specification & Wiring 33
3.11. AXIS Differential Pulse Axis Interface 35
3.11.1. AXIS Interface Signal Specification & Wiring.....36
3.11.2. Basic Usage 37
Chapter IV Expansion Module 39
4.1. CAN Bus Expansion.....39
4.1.1. CAN Bus Expansion Wiring 39
4.1.2. CAN Bus Expansion Resource Mapping.....41
4.2. EtherCAT Bus Expansion 45
4.2.1. EtherCAT Bus Expansion Wiring 45
4.2.2. EtherCAT Bus Expansion Resource Mapping 47
Chapter V Programming.....49
5.1. Program in RTSys Software 49
5.2. Upgrade Controller Firmware 54
5.3. Program in Host-Computer by PC Languages 55
Chapter VI Operation and Maintain.....58
6.1. Regular Inspection and Maintenance 58
6.2. Common Problems & Solutions 59

Chapter I Production Information

1.1. Product Information

ZMC212BE economical multi-axis motion controller is a stand-alone motion controller that is compatible with bus and pulse type. The controller itself supports 12 axes at most, but it can be extended to 16 axes for complex continuous trajectory control requirements.

ZMC212BE economical multi-axis motion controller can be applied in electronic semiconductor equipment (testing equipment, assembly equipment, locking equipment, soldering machine), dispensing equipment, non-standard equipment, printing and packaging equipment, textile and garment equipment, stage entertainment equipment, medical equipment, assembly line, etc.

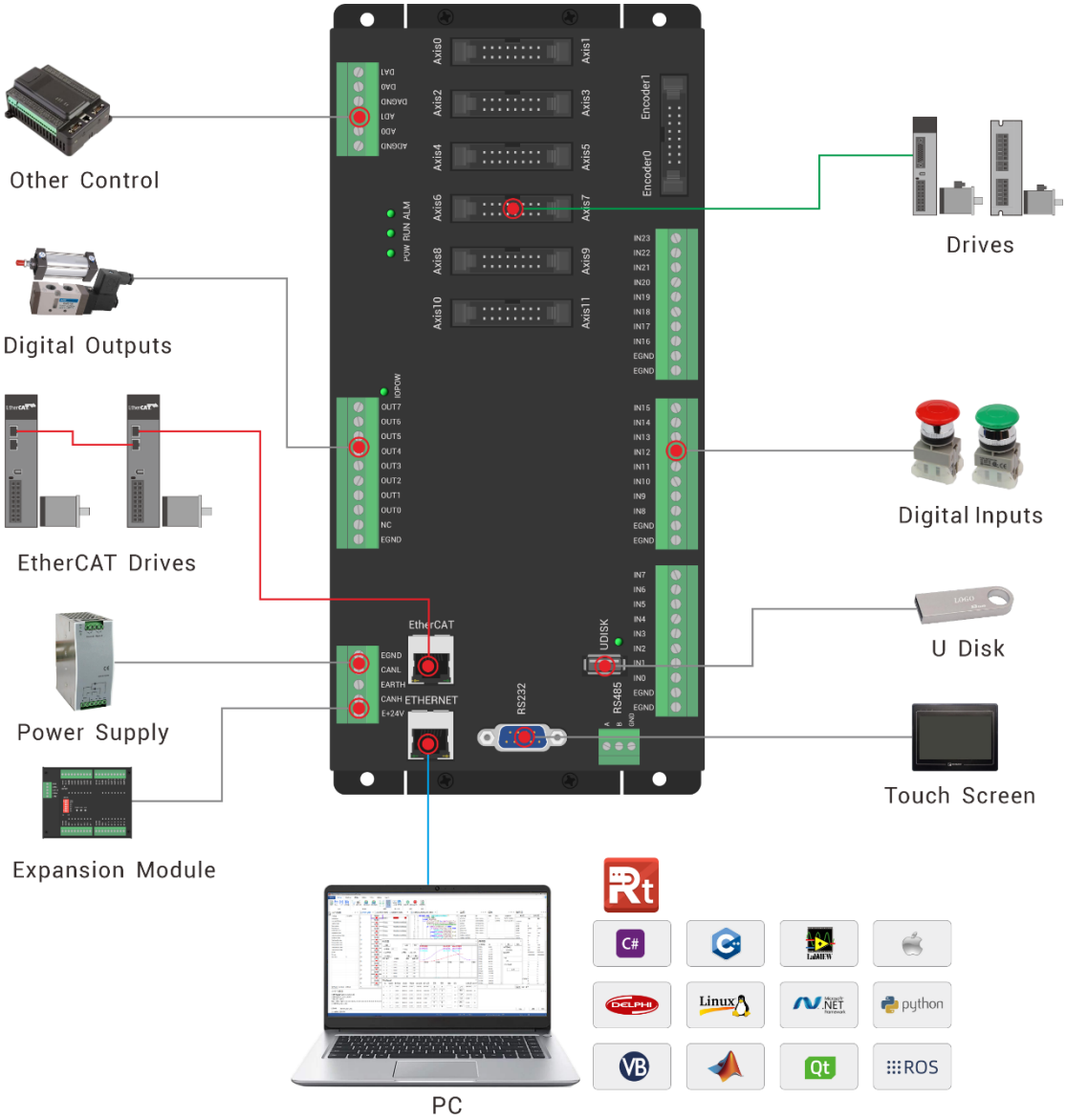
1.2. Function Features

- Motion control of up to 16 axes.
- Pulse output mode: pulse / direction or dual pulses.
- Support encoder position measurement, which can be configured as handwheel input mode.
- Maximum pulse frequency output of each axis: 10MHZ.
- 4096 isolated inputs and 4096 isolated outputs can be extended at most through CAN bus and EtherCAT bus.
- Axis position limit signal / origin signal port can be configured as any input at will.
- The maximum output current of general digital outputs can reach 300mA, which can directly drive some kinds of solenoid valves.
- Interfaces: EtherCAT, RS232, RS485, U Disk, Ethernet.
- Support linear interpolation, any circular interpolation, helical interpolation, and

spline interpolation of 16 axes at most.

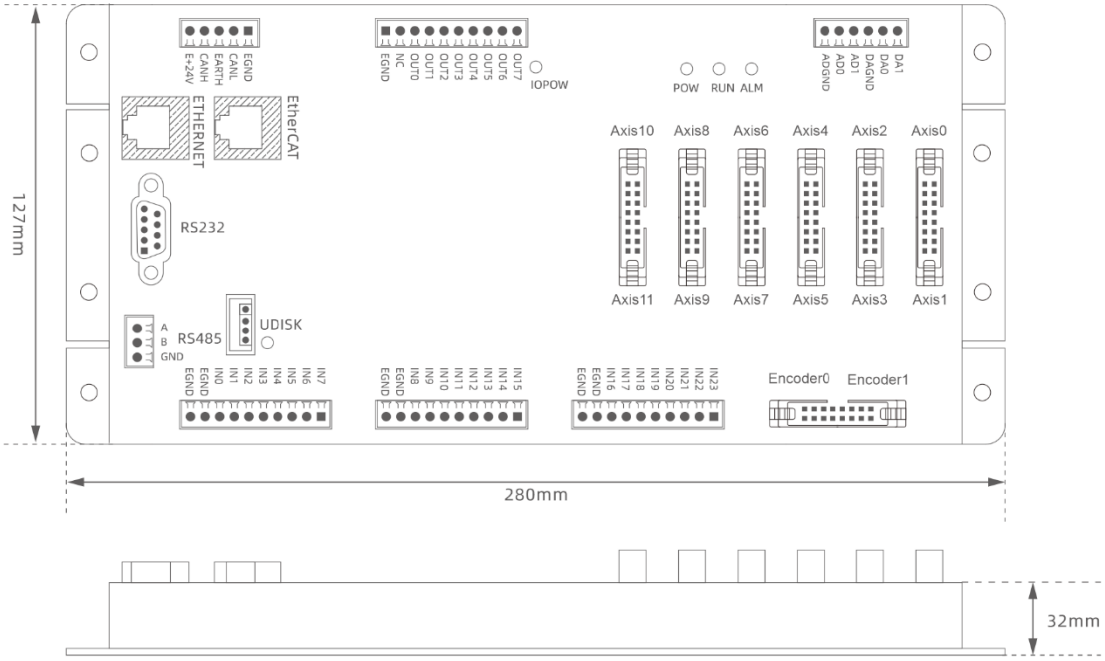
- Support electronic cam, electronic gear, position latch, synchronous follow, virtual axis, etc.
- Support pulse closed loop, pitch compensation and other functions.
- Support multi-file and multi-task programming in ZBasic.
- A variety of program encryption methods to protect your intellectual property rights.
- Power failure detection and power failure storage.

1.3. System Frame



1.4. Hardware Installment

The ZMC212BE motion controller adopts the horizontal installation method of screw fixing, and each controller should be installed with 4 screws for fastening.



→ Unit: mm → Installment Hole Diameter: 4.5mm

Installation attention	● Non-professionals are strictly prohibited to operate. Specifically, professionals who had been trained related electrical equipment, or who master electrical knowledge. ● Please be sure to read the product instruction manual and safety precautions carefully before installation. ● Before installation, please ensure that the product is powered off. ● Do not disassemble the module, otherwise the machine may be damaged. ● Avoid direct sunlight installation. ● In order to facilitate ventilation and controller replacement, 2-3cm should be left between the upper and lower parts of the controller and the installation environment and surrounding components. ● Considering the convenient operation and maintenance of the controller, please **do not** install the controller in the following places:

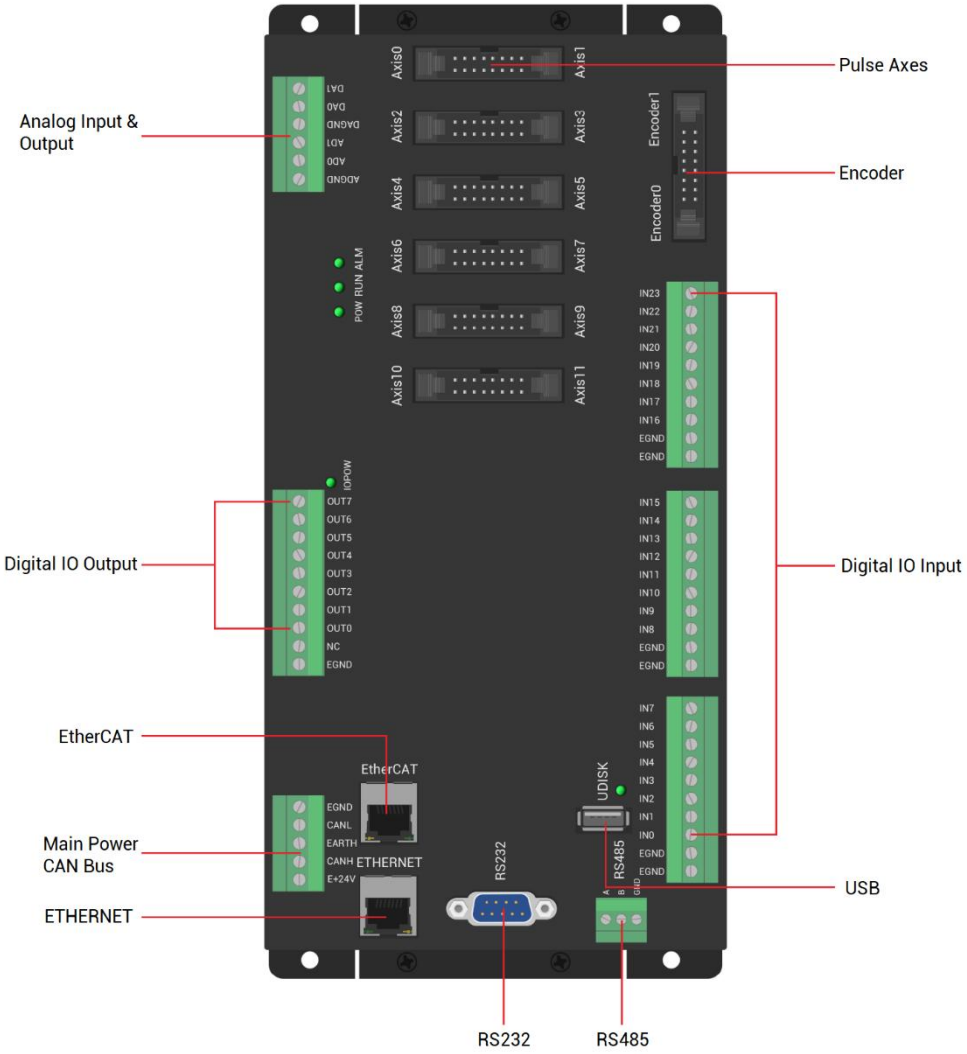
	a) places where the surrounding ambient temperature exceeds the range of -10°C-55°C b) places where the ambient humidity exceeds the range of 10%-95% (non-condensing) c) places with corrosive gases and flammable gases d) places with many conductive powders such as dust and iron powder, oil mist, salt, and organic solvents

Chapter II Product Specification

2.1. Basic Specification

Item	Description
Model	ZMC212BE
Basic Axes	12
Max Extended Axes	16
Basic Axes Type	Local pulse axis / encoder axis / EtherCAT bus axis
Digital IO	24 inputs, 8 outputs
IO in Axis Interface	0 input, 0 output
Max Extended IO	4096 inputs, 4096 outputs
AD/DA	2 general ADs, 0-10V. 2 general DAs, 0-10V.
Max Extended AD/DA	512ADs, 512 DAs
Pulse Bit	32
Encoder Bit	32
Speed Acceleration Bit	32
Pulse Max Frequency	10MHz
Motion Axis Buffer	512
Array Space	40000
Program Space	460KByte
Flash Space	128MByte
Power Supply Input	24V DC input
Communication Interfaces	RS232, RS485, Ethernet, U disk, CAN, EtherCAT
Dimensions	280mm*127mm*32mm

2.2. Interface Definition



→ Interface Description

Mark	Interface	Number	Description
IO POW	Status Indication Led	1	IO power indicator: it lights when IO power is conducted.
POW		1	Power indicator: it lights when power is conducted.
RUN		1	Run indicator: it lights when runs normally
ALM		1	Error indicator: it lights when runs abnormally
RS232	RS232 serial port (port0)	1	Use MODBUS_RTU protocol

RS485	RS485 serial port (port1)	1	Use MODBUS_RTU protocol
EtherCAT	EtherCAT bus interface	1	Connect to EtherCAT bus drive and EtherCAT bus expansion modules.
ETHERNET	Ethernet	1	Use MODBUS_TCP protocol, expand Ethernet through interchanger, the number of net port channels can be checked through "?*port", default IP address id 192.168.0.11
UDISK	U disk	1	Insert U disk equipment
E+24V	Main power	1	24V DC power supplies for controller
CAN	CAN bus interface	1	Connect to CAN expansion module or controller.
IN	Digital IO input	24	NPN type, internal 24V supply power, IN0-1 have the latch function.
OUT	Digital IO output	8	NPN leak type, internal 24V supply power.
AD	Analog input	2	Resolution: 12 bits, 0-10V
DA	Analog output	2	Resolution: 12 bits, 0-10V
ENCODER	Encoder	2	Include differential encoder input
AXIS	Pulse axis	12	Include differential pulse output

2.3.Work Environment

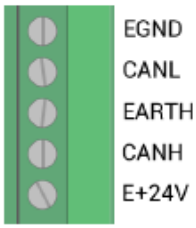
Item		Parameters
Work Temperature		-10℃-55℃
Work relative Humidity		10%-95% non-condensing
Storage Temperature		-40℃ ~ 80℃ (not frozen)
Storage Humidity		Below 90%RH (no frost)
vibration	Frequency	5-150Hz
	Displacement	3.5mm(directly install)(<9Hz)
	Acceleration	1g(directly install)(>9Hz)
	Direction	3 axial direction
Shock (collide)		15g, 11ms, half sinusoid, 3 axial direction
Degree of Protection		IP20

Chapter III Wiring, Communication Configuration

3.1. Power Input

The power input adopts a screw-type pluggable terminal with a 5Pin pitch of 5.08mm, which is shared by controller main power supply and CAN communication.

→ Terminal Definition:

Terminal	Name	Type	Function
	EGND	Input	Main power / CAN public end
	CANL	Input / output	CAN differential data -
	EARTH	Connect to ground	Case Protection Ground
	CANH	Input / output	CAN differential data +
	E+24V	Input	Main power 24V input

3.1.1. Power Specification

→ Specification

Item	Description
Input Voltage	DC24V(-5%~5%)
Current to open	≤0.5A
Current to work	≤0.4A
Anti-reverse connection	YES
Overcurrent Protection	YES

3.1.2. CAN Communication Specification & Wiring

CAN interface of the controller uses standard CAN communication protocol, there are

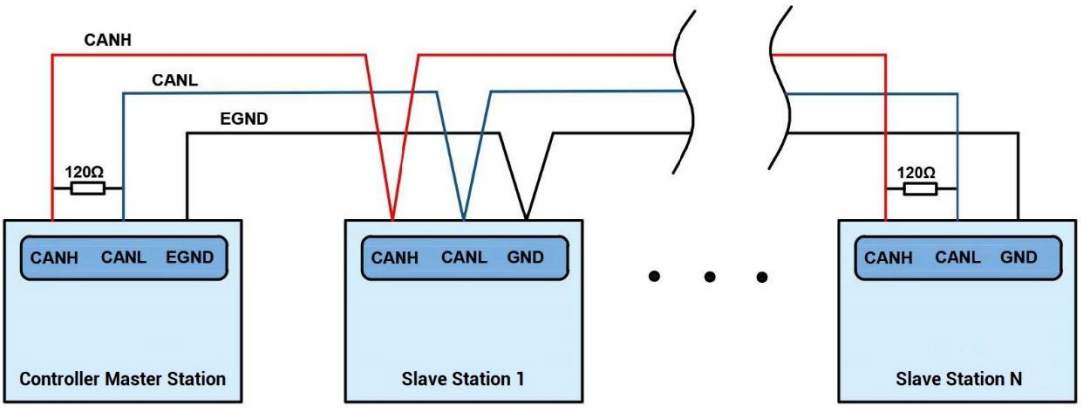
mainly 3 terminals, CANL, CANH and public end. And it supports connecting ZIO/ZMIO expansion module and other standard CAN devices.

→ Specification

Item	CAN
Max Communication Rate	1M(bps)
Terminal Resistor	120Ω
Topology	Daisy chain connection structure
Nodes can be extended	Up to 16
Communication Distance	Longer communication distance, lower communication rate, max 30m is recommended.

→ Wiring Reference

Connect the CANL and CANH of the standard CAN module to the CANL and CANH of the other side correspondingly. And public ends of the CAN bus communication both parties are connected together. In CAN bus left and right sides, connect a 120Ω resistor respectively (please see below graphic).



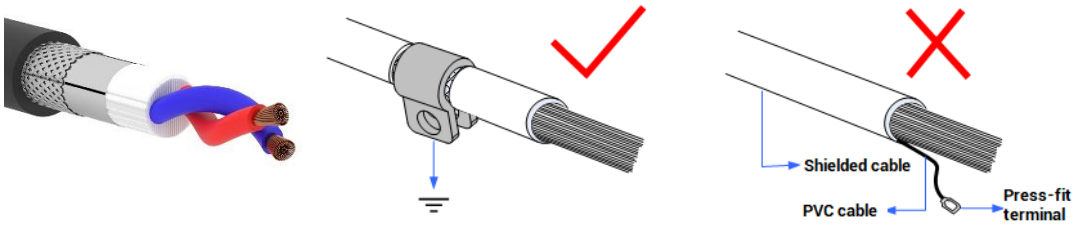
→ Wiring Notes:

- As above, the daisy chain topology is used for wiring, so the star topology cannot be used. When the use environment is ideal and there are few nodes, the branch structure can also be considered.

- Please connect a 120Ω terminal resistor (when there is one to one connection or the number of slave stations is few and the wiring is relatively short, the terminal resistor can not be added) in parallel at the two ends of the CAN bus to match the circuit impedance and ensure communication stability.
- Please be sure to connect the common terminal of each node on the CAN bus to prevent the CAN chip from burning out.
- Please use twisted-pair shielded wires, especially in harsh environments, make sure the shielding layer is fully grounded.
- On-site wiring should also pay attention to the distance between strong current and weak current wiring, it is recommended to be more than 20cm.
- It should be noted that the equipment grounding (chassis) on the entire line must be good, and the grounding of the chassis should be connected to the standard factory ground pile.

→ Cable Requirements:

Twisted pair shielded wire, and shielded cable grounded.



3.1.3. Basic Usage Method

(1) Please follow the above wiring instructions for correct wiring.

(2) After power on, please use ETHERNET, RS232 or RS485 to connect to RTSys.

(3) Please use the "CANIO_ADDRESS" command to set the master's "address" and

"speed" according to the needs, and use the "CANIO_ENABLE" command to enable or disable the internal CAN master function, or through "RTSys/Controller/State the Controller/Communication Info" to view the CAN status intuitively, and refer to the "RTBasic Programming Manual" for details.

```
CAN communication settings:
CANIO_ADDRESS = 32, CANIO_ENABLE = 1
ZCAN Master
CAN baud: 500KBPS
CAN enable: ON

Serial port configuration:
Port0:(RS232) is ModbusSlave Mode.
Address:1, variable:2
Baud:38400
DataBits:8
StopBits:1
Parity:0
Port1:(RS485) is ModbusSlave Mode.
Address:1, variable:2
Baud:38400
DataBits:8
StopBits:1
Parity:0
```

(4) Correctly set the "address" and "speed" of the slave expansion module according to the manual of the slave station.

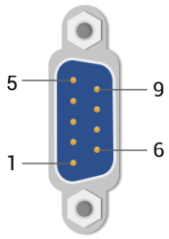
(5) After all the settings are completed, restart the power supply of all stations to establish communication.

(6) Note that the "speed" settings of each node on the CAN bus must be consistent, and the "address" settings cannot cause conflicts, otherwise the "ALM" alarm light will be on, and the communication establishment will fail or the communication will be disordered.

3.2. RS232 Serial Port

RS232 is in a standard DB9 male socket and supports MODBUS_RTU protocol and custom communication.

→ Interface Definition:

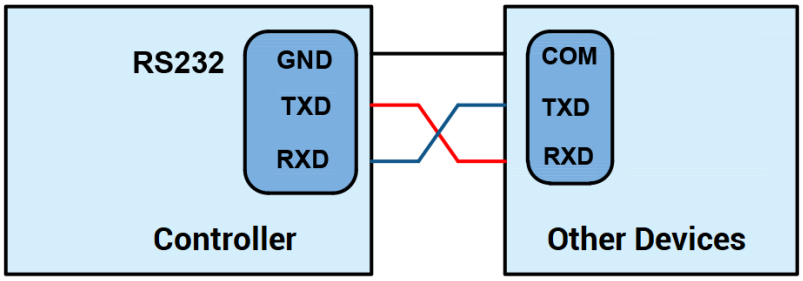
Terminal	PIN	Name	Type	Function
	1, 4, 6, 7, 8	NC	Spare	Reserved
	2	RXD	Input	RS232, receive signal
	3	TXD	Output	RS232, send signal
	5	GND	Output	+5V power supply output (-) and communication public end
	9	+5V	Output	+5V power supply output (+), max is 300mA

3.2.1. RS232 Communication Specification & Wiring

→ Specification

Item	RS232 (port 0)
Max Communication Rate	115200 (bps)
Terminal Resistor	/
Topology	Point to point (one to one)
Nodes can be extended	1
Communication Distance	Longer communication distance, lower communication rate, max 5m is recommended.

→ Wiring Reference



→ Wiring Notes:

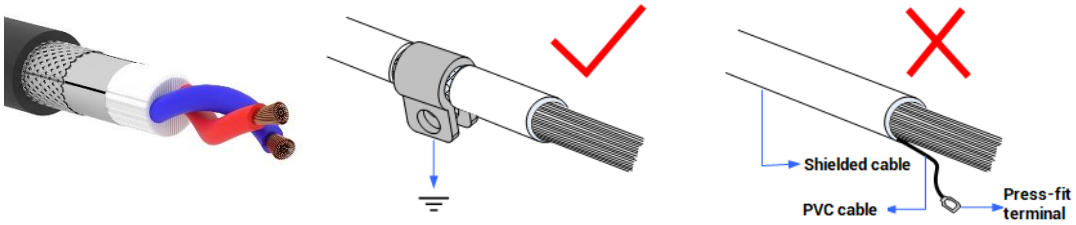
- The wiring of RS232 is as above, the sending and receiving signals need to be cross-connected, and it is recommended to use a double-female cross line when

connecting to a computer.

- Please be sure to connect the public end of each communication node to prevent the communication chip from burning out.
- Please use twisted-pair shielded cables, especially in harsh environments, and make sure the shielding layer is fully grounded.

→ Cable Requirements:

Twisted pair shielded wire, and shielded cable grounded.



3.2.2. Basic Usage Method

(1) Please follow the above wiring instructions for correct wiring.

(2) After power on, please use ETHERNET, RS232 (default parameter, it can be connected directly) and RS485 (default parameter can be directly connected, one adapter is needed for hardware) to connect to RTSys.

(3) Please use the "ADDRESS" and "SETCOM" commands to set and view the protocol station No. and configuration parameters, see "RTBasic Programming Manual" for details.

(4) Correctly set the relevant parameters of the third-party equipment according to their respective instructions to match the parameters of each node.

(5) When all are configured, it's time to do communicating.

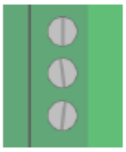
(6) Communication data of RS232 can be directly viewed through "RTSys / Controller / State the Controller / CommunicationInfo".

```
CAN communication settings:
CANIO_ADDRESS = 32, CANIO_ENABLE = 1
ZCAN Master
CAN baud: 500KBPS
CAN enable: ON

Serial port configuration:
Port0:(RS232) is ModbusSlave Mode.
Address:1, variable:2
Baud:38400
DataBits:8
StopBits:1
Parity:0
Port1:(RS485) is ModbusSlave Mode.
Address:1, variable:2
Baud:38400
DataBits:8
StopBits:1
Parity:0
```

3.3. RS485 Serial Port

The communication interface adopts a screw-type pluggable terminal with a 3Pin pitch of 5.08mm. RS485 communication can be connected through the corresponding interface of this terminal.

→ Specification

Terminal	Name	Function
GND B A 	GND	485 communication public end
	B	485-
	A	485+

3.3.1. RS485 Communication Specification & Wiring

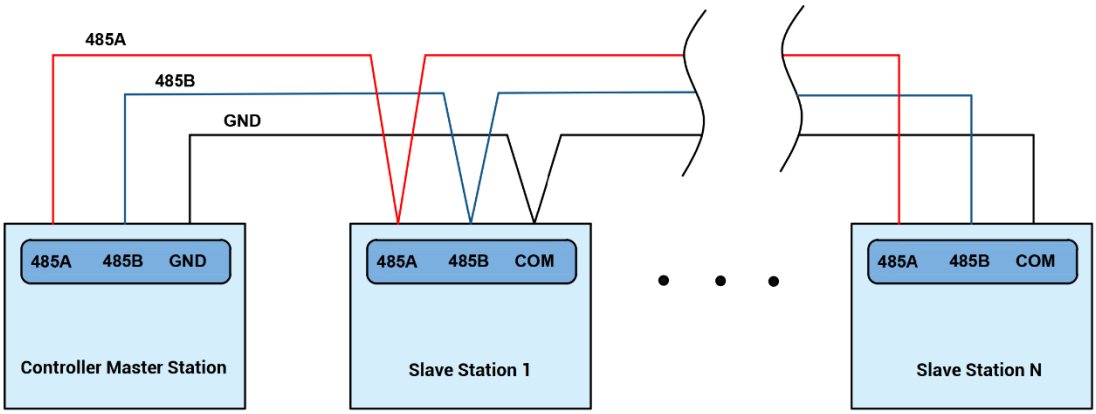
RS485 supports MODBUS_RTU protocol and custom communication, which mainly includes 485A, 485B and public end.

→ Specification

Item	RS485 (port 1)
Max Communication Rate	115200 (bps)
Terminal Resistor	/
Topology	Daisy chain connection structure
Nodes can be extended	Up to 127
Communication Distance	Longer communication distance, lower communication rate, max 30m is recommended.

→ Wiring Reference

Connect 485A and 485B of RS485 to 485A and 485B of the controller correspondingly, and connect the public end of both parties together.



→ Wiring Notes:

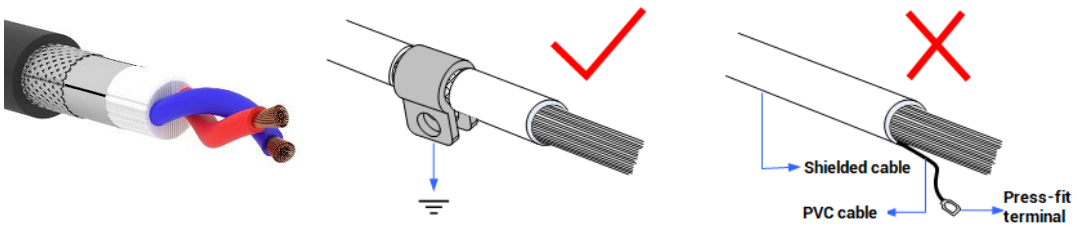
- As above, the daisy chain topology is used for wiring, so the star topology cannot be used. When the use environment is ideal and there are few nodes, the branch structure can also be considered.
- Please connect a 120Ω terminal resistor in parallel at the two ends of the CAN bus to match the circuit impedance and ensure communication stability.
- Please be sure to connect the common terminal of each node on the CAN bus to prevent the CAN chip from burning out.
- Please use twisted-pair shielded wires, especially in harsh environments, make sure

the shielding layer is fully grounded.

- On-site wiring should also pay attention to the distance between strong current and weak current wiring, it is recommended to be more than 20cm.
- It should be noted that the equipment grounding (chassis) on the entire line must be good, and the grounding of the chassis should be connected to the standard factory ground pile.

→ Cable Requirements:

Twisted pair shielded wire, and shielded cable grounded.



3.3.2. Basic Usage Method

(1) Please follow the above wiring instructions for correct wiring.

(2) After power on, please use ETHERNET, RS232 or RS485 to connect to RTSys.

(3) Please use the "ADDRESS" and "SETCOM" commands to set and view the protocol station number and configuration parameters, see "RTBasic Programming Manual" for details.

(4) Correctly set the relevant parameters of the third-party equipment according to their respective instructions to match the parameters of each node.

(5) After all the settings are completed, restart the power supply of all stations to establish communication.

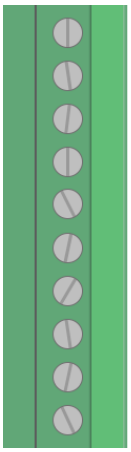
(6) RS485 communication data can be checked directly through “RTSys/ Controller / Controller State / CommunicationInfo".

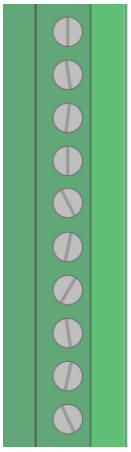
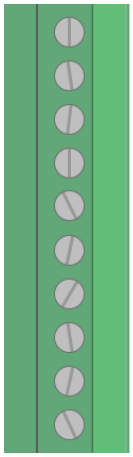
```
CAN communication settings:
CANIO_ADDRESS = 32, CANIO_ENABLE = 1
ZCAN Master
CAN baud: 500KBPS
CAN enable: ON

Serial port configuration:
Port0:(RS232) is ModbusSlave Mode.
Address:1, variable:2 delay:400ms
Baud:38400
DataBits:8
StopBits:1
Parity:0
Port1:(RS485) is ModbusSlave Mode.
Address:1, variable:2 delay:400ms
Baud:38400
DataBits:8
StopBits:1
Parity:0
```

3.4. IN Digital Input & High-Speed Latch Port

The digital input adopts 3 groups of 10Pin screw-type pluggable terminals with a pitch of 5.08mm, and the latch function is integrated in digital signal inputs.

→ Wiring Definition

Terminal	Name	Type	Function 1	Function 2
 EGND	EGND	/	IO public end	/
EGND	EGND	/		/
IN0	IN0	NPN type, general inputs.	Input 0	Latch A
IN1	IN1		Input 1	Latch B
IN2	IN2		Input 2	/
IN3	IN3		Input 3	/
IN4	IN4		Input 4	/
IN5	IN5		Input 5	/
IN6	IN6		Input 6	/
IN7	IN7		Input 7	/
	EGND	/	IO public end	/
	EGND	/		/
	IN8	NPN type, general inputs.	Input 8	/
	IN9		Input 9	/
	IN10		Input 10	/
	IN11		Input 11	/

 EGND EGND IN8 IN9 IN10 IN11 IN12 IN13 IN14 IN15	IN12		Input 12	/
	IN13		Input 13	/
	IN14		Input 14	/
	IN15		Input 15	/
 EGND EGND IN16 IN17 IN18 IN19 IN20 IN21 IN22 IN23	EGND	/	IO public end	/
	EGND	/		/
	IN16	NPN type, general inputs.	Input 16	/
	IN17		Input 17	/
	IN18		Input 18	/
	IN19		Input 19	/
	IN20		Input 20	/
	IN21		Input 21	/
	IN22		Input 22	/
	IN23		Input 23	/
Note: ✧ IN 0 and IN 1 both support latch function. IN 0 supports latching A, and IN 1 supports latching B.				

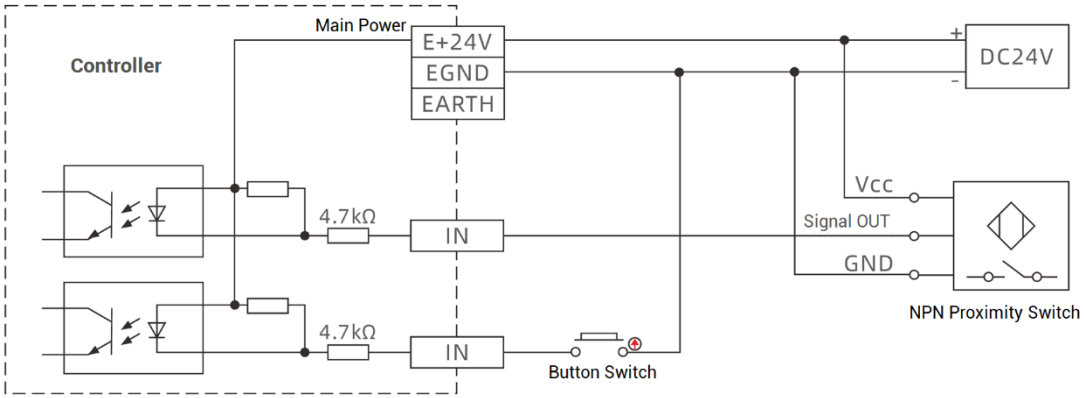
3.4.1.Digital Input Specification & Wiring

→ Specification

Item	General input (IN0-23)
Input mode	NPN type, triggered by low level
Input frequency	< 5kHz
Input impedance	4.7KΩ
Input voltage level	DC24V
Voltage to ON	<14.5V
Voltage to OFF	>14.7V
Min input current	-1.8mA
Max input current	-6mA
Isolation mode	Optical isolation

Note: the above parameters are standard values when the controller power supply voltage (E+24V port) is 24V.

→ Wiring Reference



→ Wiring Note:

- The wiring principle of general input (IN0-23) is shown in the figure above. The external signal source can be an optocoupler or a key switch or sensor, etc., all can be connected as long as the output level meets the requirements.
- For the public end, please select the "EGND" port on the IO power supply terminal to connect to the "COM" terminal of the external input device. If the power supply of the signal area of the external device and the power supply of the controller are in the same power supply system, this connection can also be omitted.

3.4.2. Basic Usage Method

(1) Please follow the above wiring instructions for correct wiring.

(2) After power on, please use any of the three interfaces ETHERNET, RS232 and RS485 to connect to RTSys.

(3) State values of relative inputs can be read directly through “IN” command, also, it can be read through “RTSys/Tool/In”. Please refer to “RTBasic” for details.

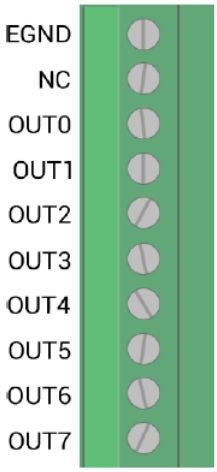
(4) Latch function can be set and opened through “REGIST” instruction, in software, use

REG_INPUTS to configure. Please refer to "RTBasic" for details.

3.5. OUT Digital Output

The digital output adopts a set of screw-type pluggable terminals with a spacing of 5.08mm.

→ Wiring Definition

Terminal	Name	Type	Function 1
	EGND	/	IO public end
	NC	/	Reserved
	OUT0	NPN leakage, general outputs	Output 0
	OUT1		Output 1
	OUT2		Output 2
	OUT3		Output 3
	OUT4		Output 4
	OUT5		Output 5
	OUT6		Output 6
	OUT7		Output 7

3.5.1.Digital Output Specification & Wiring

→ Specification

Item	General Output (OUT0-7)
Output mode	NPN, it is 0V when outputs
Output frequency	< 8kHz
Output voltage level	DC24V
Max output current	+300mA
Max leakage current when OFF	25μA
Respond time to ON	12μs
Respond time to OFF	80μs
Overcurrent protection	Support

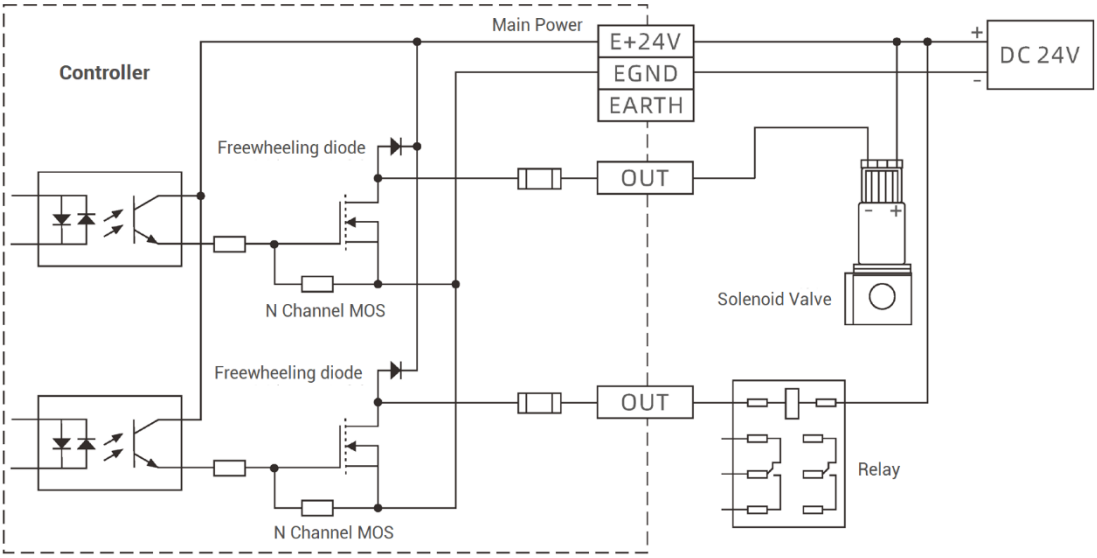
Isolation method	Photoelectronic isolation

Note:

✧ The times in the table are typical values based on the resistive load, and may change when the load circuit changes.

✧ Due to the leak-type output, the shutdown of the output will be more obviously affected by the external load circuit, and the output frequency should not be set too high in the application.

→ Wiring Reference



→ Wiring Note:

- The wiring principle of general output OUT0-7 is shown in the figure above. The external signal receiving end can be an optocoupler or a relay or solenoid valve, all can be connected as long as the input current does not exceed 300mA.
- For the connection of the public end, please select the "EGND" port on the IO power terminal to connect to the negative pole of the DC power supply of the external input device. If the DC power supply of the external device and the controller power supply are in the same power supply system, this connection can also be omitted.

3.5.2. Basic Usage Method

(1) Please follow the above wiring instructions for correct wiring.

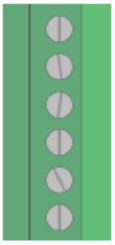
(2) After power on, please use any of the three interfaces ETHERNET, RS232 and RS485 to connect to RTSys.

(3) Terminal can be operated to ON or OFF directly through "OP" command, also, it can be opened or closed through "RTSys/Tool/Op". Please refer to "RTBasic" for details.

3.6. AD/DA Analog Input & Output

The analog port adopts a set of 6Pin screw-type pluggable terminals with a pitch of 5.08mm.

→ Wiring Definition

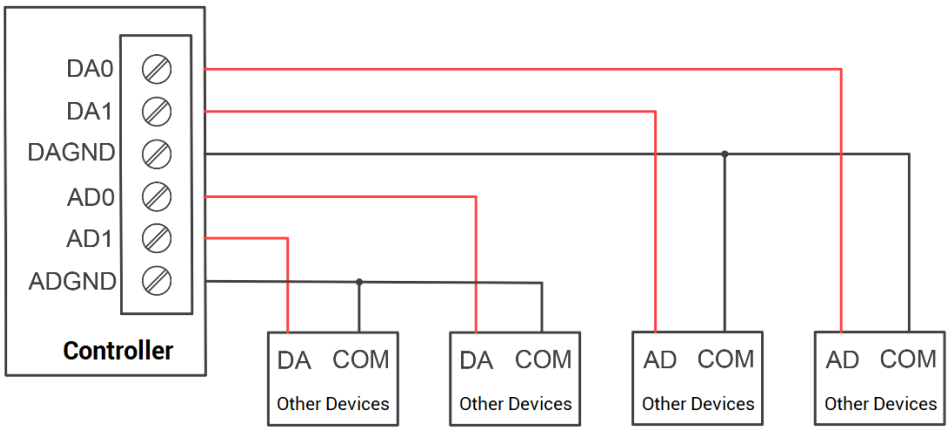
Terminal	Name	Type	Function
	DA1	Output	Analog output terminal AOUT(1)
	DA0	Output	Analog output terminal AOUT(0)
	DAGND	Public end	Analog public end
	AD1	Input	Analog input terminal AIN(1)
	AD0	Input	Analog input terminal AIN(0)
	ADGND	Public end	Analog public end

3.6.1. Analog Input / Output Specification & Wiring

→ Specification

Item	DA (0-1)	AD (0-1)
Resolution	12-bit	12-bit
Data range	0-4095	0-4095
Signal range	0-10V output	0-10V input
Data refresh ratio	1KHz	1KHz
Voltage input impedance / output load	>33KΩ (voltage output load)	300KΩ (voltage input impedance)

→ Wiring Reference

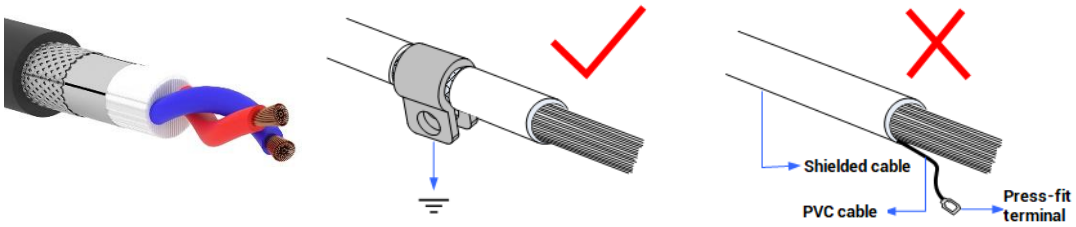


→ Wiring Note:

- The analog input/output wiring method is as shown in the figure above, and the external load signal range must match it.
- Please use twisted-pair shielded cables, especially in harsh environments, and make sure the shielding layer is fully grounded.

→ Cable Requirements:

Twisted pair shielded wire, and shielded cable grounded.



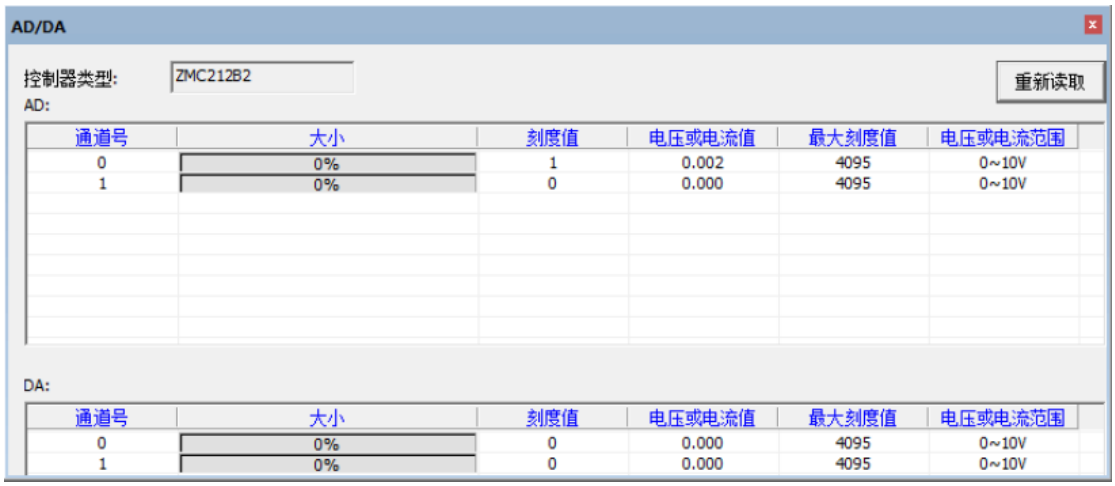
3.6.2. Basic Usage Method

(1) Please follow the above wiring instructions for correct wiring.

(2) After power on, please use any of the three interfaces ETHERNET, RS232 and RS485

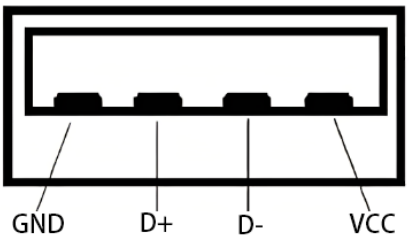
to connect to RTSys.

(3) Analog input voltage can be read through “AIN” command and corresponding analog voltage can be output through “AOUT” command, also, data of each channel can be checked through “RTSys/Tool/AD/DA”. Please refer to “RTBasic” for details.



3.7. U Disk

The ZMC212BE motion controller provides a USB communication interface to insert a U disk device, which is used for ZAR program upgrade, controller data import and export, and 3 file executions. Its schematic diagram is shown in the figure below:

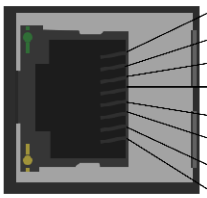


→ Specification

Item	USB2.0
Max Communication Ratio	12Mbps
Max Output Current of 5V	500mA
Whether Isolates	No

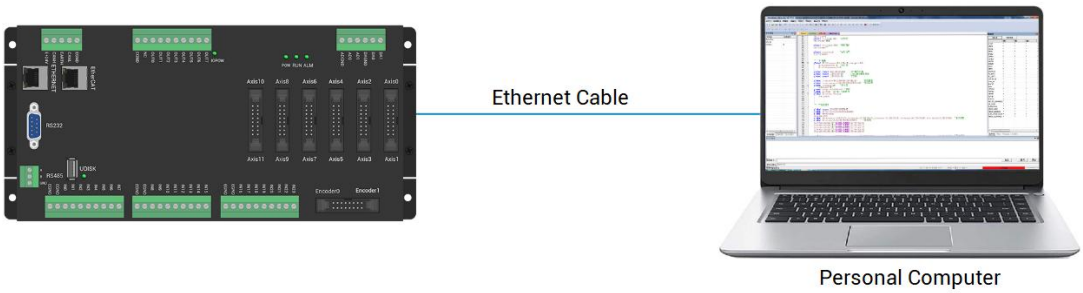
3.8. ETHERNET

ZMC212BE motion controller has a 100M Ethernet port, supports MODBUS_TCP protocol and custom communication, and the default IP address is 192.168.0.11. The pin definition diagram is as follows:

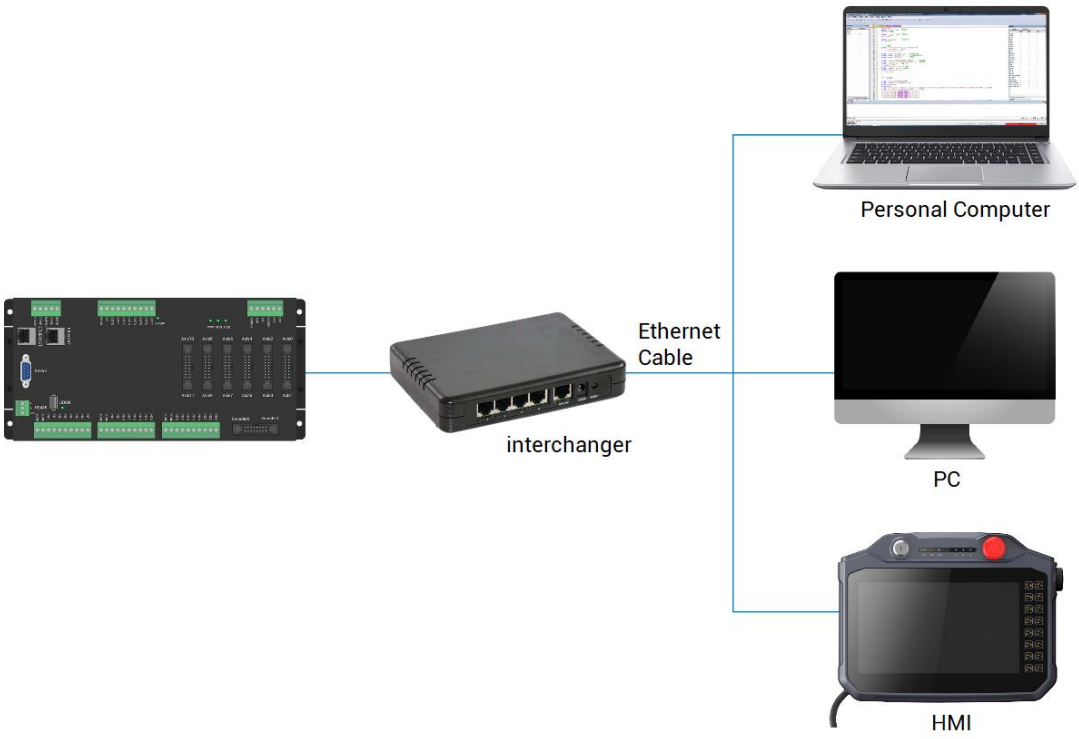


PIN	Signal	Description
1	TX+	Send signal (+)
2	TX-	Send signal (-)
3	RX+	Receive signal (+)
4	NC	Reserved
5	NC	Reserved
6	RX-	Receive signal (-)
7	NC	Reserved
8	NC	Reserved

The Ethernet port of the controller can be connected point-to-point with a computer, HMI, etc. through an Ethernet cable. The schematic diagram is as follows:

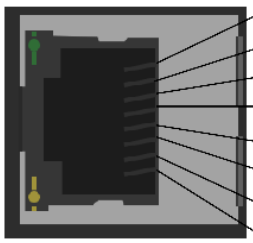


The controller can also be connected to the interchanger through an Ethernet cable, and connected to other devices through the interchanger to realize multi-point connection. The schematic diagram is as follows:



3.9. EtherCAT Bus Interface

ZMC212BE motion controller has a 100M EtherCAT communication interface, supports EtherCAT protocol, and connects to EtherCAT driver or EtherCAT expansion module. The pin definition diagram is as follows:



PIN	Signal	Description
1	TX+	Send signal (+)
2	TX-	Send signal (-)
3	RX+	Receive signal (+)
4	NC	Reserved
5	NC	Reserved
6	RX-	Receive signal (-)
7	NC	Reserved
8	NC	Reserved

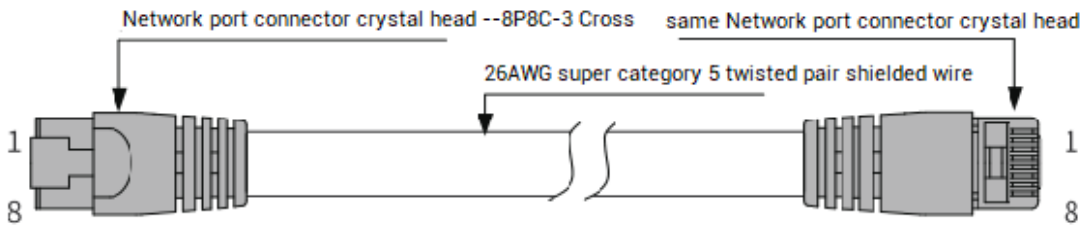
→ Specification

Item	Specification
Communication protocol	EtherCAT protocol
Valid service	CoE（PDO、SDO）、FoE
Synchronization method	IO adopts input and output synchronization / DC-distributed clock
Physical level	100BASE-TX
Duplex mode	full duplex
Topology	linear topology
Transfer media	Cable
Transfer distance	It is less than 100M between 2 nodes
Process data	Max 1486 bytes of single frame
Synchronization shaking of two slave stations	<1us
Refresh	1000 digital inputs and outputs are about 30us

→ Communication Cable Requirements

Both ETHERNET communication interface and EtherCAT communication interface adopt standard Ethernet RJ45 interface.

The network cable adopts Category 5e shielded twisted pair, and the crystal head has a metal shell to reduce interference and prevent information from being eavesdropped. As shown below:



Item	Specification
Cable type	Flexible crossover cable, Cat 5e
Cable type	twisted pair
Cable pair	4
Isolation	cross skeleton

Connector	Crystal head with iron shell
Cable material	PVC
Cable length	Less than 100m

Use RJ45 network cable connection method:

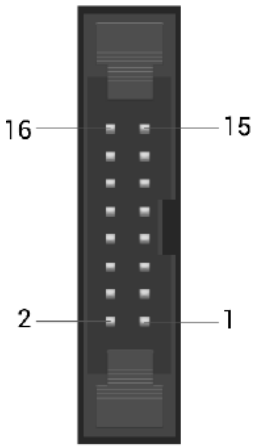
- When installing, hold the crystal head with the cable and insert it into the RJ45 interface until it makes a "click" sound (kada).
- In order to ensure the stability of communication, please fix the cables with cable ties.
- When disassembling, press the tail mechanism of the crystal head and pull out the connector and the module in a horizontal direction.

Please use tube-type pre-insulated terminals and cables with appropriate wire diameters to connect the user terminals.

3.10. ENCODER Interface

ZMC212BE provides 2 encoder interfaces, and they are integrated in one 16pin horn connector male seat. It provides 0V and +5V output, which can provide 5V power for encoders.

→ Interface Definition

Interface	Pin	Signal	Description
	1	EA1+	Encoder differential input signal A1+
	2	EA1-	Encoder differential input signal A1-
	3	EB1+	Encoder differential input signal B1+
	4	EB1-	Encoder differential input signal B1-
	5	GND	Encoder signal 5V power negative pole
	6	EZ1+	Encoder differential input signal Z1+
	7	EZ1-	Encoder differential input signal Z1-
	8	+5V	Encoder signal 5V power positive pole
	9	EA0+	Encoder differential input signal A0+
	10	EA0-	Encoder differential input signal A0-
	11	EB0+	Encoder differential input signal B0+

	12	EB0-	Encoder differential input signal B0-
	13	GND	Encoder signal 5V power negative pole
	14	EZ0+	Encoder differential input signal Z0+
	15	EZ0-	Encoder differential input signal Z0-
	16	+5V	Encoder signal 5V power positive pole

Note:

✧ +5V is only used for the encoder, please don't use for others.

✧ Pin1-8 is used for encoder 1, Pin9-16 is used for encoder 0.

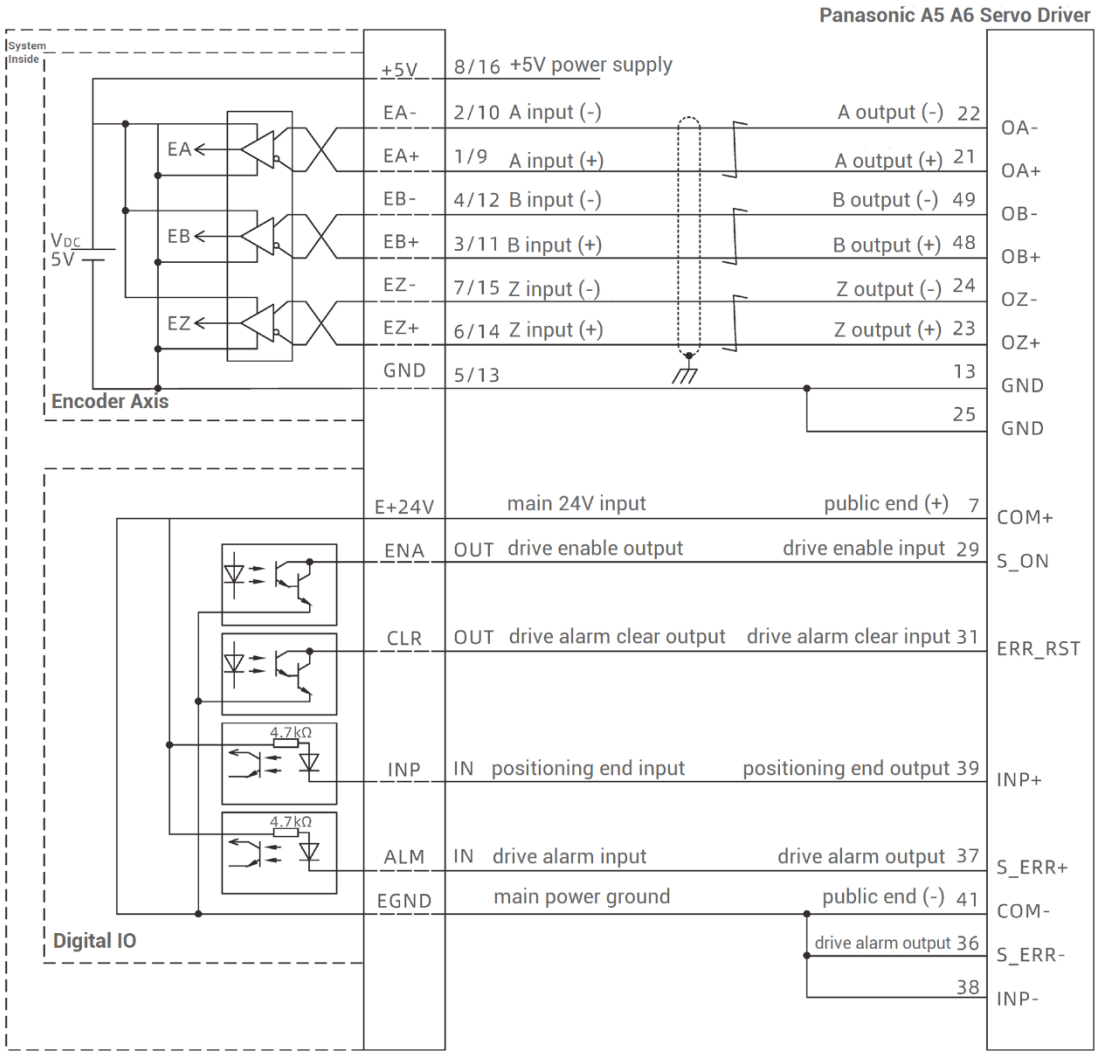
3.10.1. ENCODER Interface Specification & Wiring

→ Interface Definition

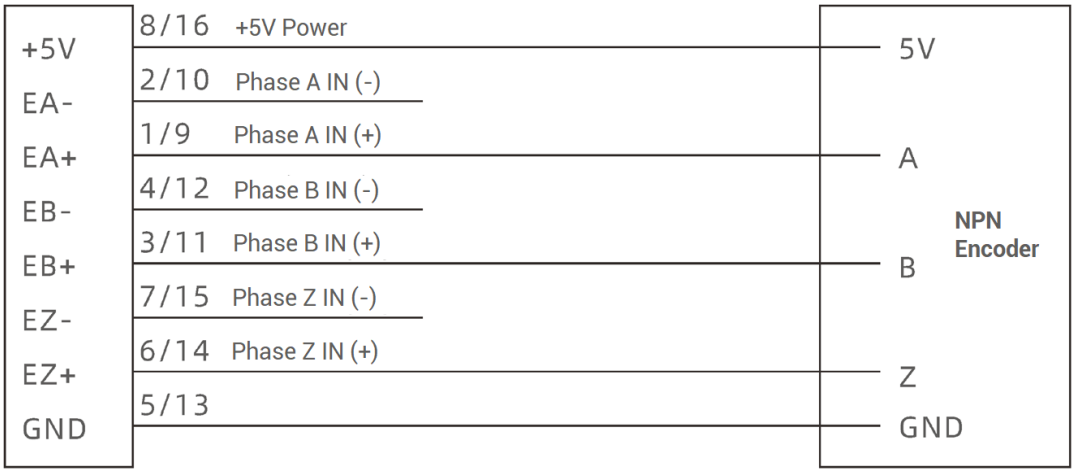
Interface	Item	Description
EA/EB/EZ	Signal type	Differential input signal
	Signal voltage range	0-5V
	Signal max frequency	5MHz
+5V, GND	5V power max output current	50mA

→ Wiring Reference:

Reference example of wiring with Panasonic A5/A6 servo driver:



"Single-Ended Encoder Wiring"



Notes:

- The wiring principle of the encoder interface is as shown in the figure above. There are differences in the wiring methods of different models of drivers, so please

connect with caution.

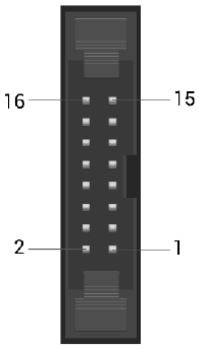
- Please use twisted-pair shielded wires, especially in harsh environments. Make sure the shielding layer is fully grounded.

3.11. AXIS Differential Pulse Axis Interface

This product provides 12 local differential pulse axis interfaces, and they integrate in six 16pin horn connector male seats.

Before using, please configure axis type through ATYPE parameter.

→ Interface Definition

Interface	Pin	Signal	Description
	1	PUL1+	Servo/step pulse output differential signal +
	2	PUL1-	Servo/step pulse output differential signal -
	3	DIR1+	Servo/step directional output differential signal +
	4	DIR1-	Servo/step directional output differential signal -
	5	GND	Pulse signal 5V power negative pole
	6	/	/
	7	/	/
	8	+5V	Pulse signal 5V power positive pole
	9	PUL0+	Servo/step pulse output differential signal +
	10	PUL0-	Servo/step pulse output differential signal -
	11	DIR0+	Servo/step directional output differential signal +
	12	DIR0-	Servo/step directional output differential signal -
	13	GND	Pulse signal 5V power negative pole
	14	/	/
	15	/	/
	16	+5V	Pulse signal 5V power positive pole

Note:

- ✧ +5V is only used for communication between controller and servo driver, please don't use for others.
- ✧ Pin1-8 is used for axis 1, Pin9-16 is used for axis 0.
- ✧ Take axis 0 and axis 1 as the example, remaining 10 axes are the same.

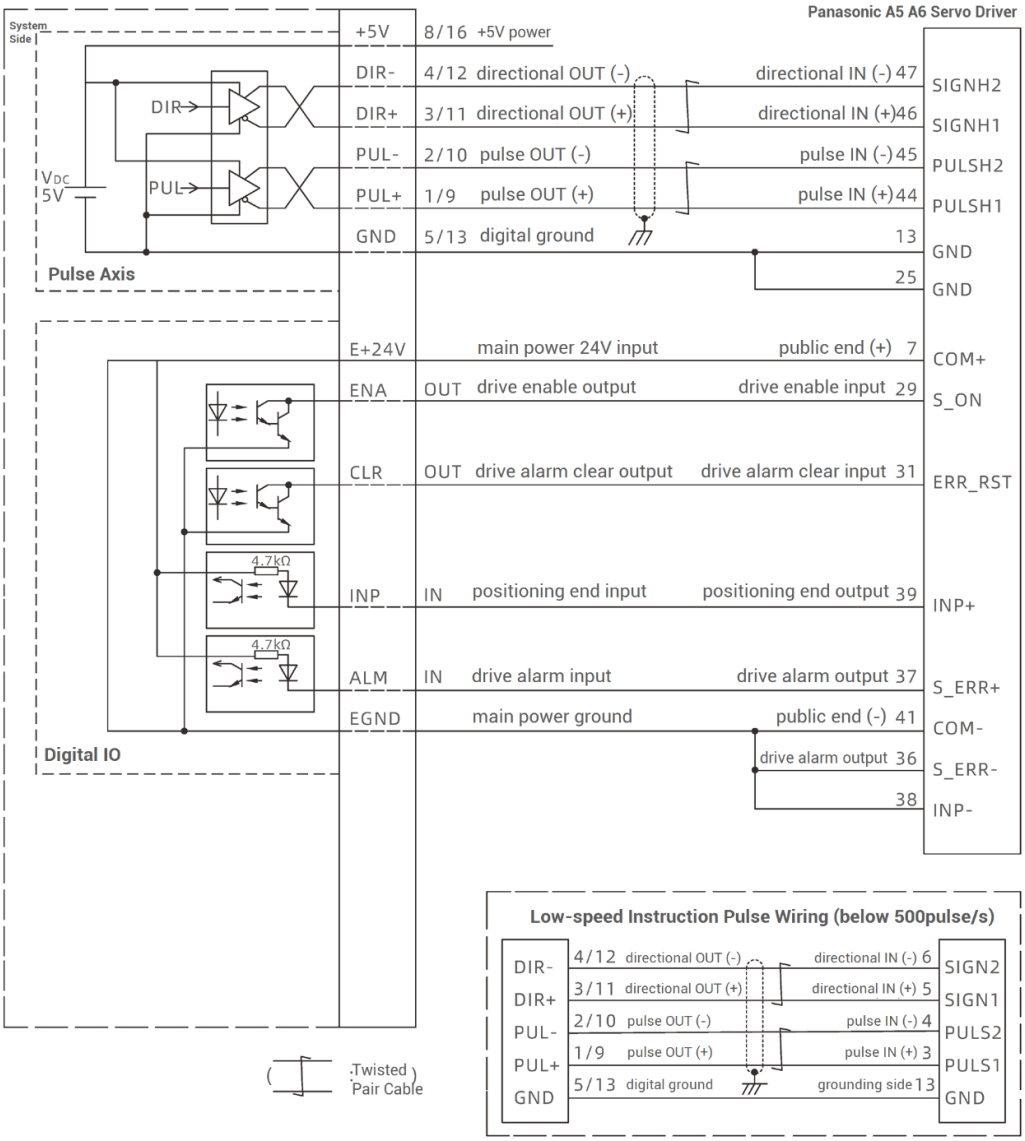
3.11.1. AXIS Interface Signal Specification & Wiring

→ Specification:

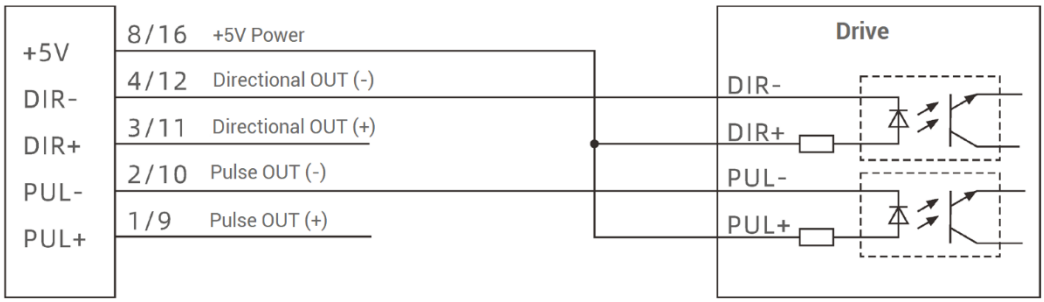
Interface	Item	Description
PUL/DIR	Signal type	Differential input signal
	Signal voltage range	0-5V
	Signal max frequency	10MHz
+5V, GND	5V power max output current	50mA

→ Wiring Reference:

Reference example of wiring with Panasonic A5/A6 servo driver:



"Single-Ended Pulse Axis Wiring"



→ Wiring Note:

✧ The wiring principle of the differential pulse axis interface is shown in the figure above, and the wiring methods of different types of drivers are different, please connect carefully.

✧ Please use twisted-pair shielded cables, especially in harsh environments, and make sure the shielding layer is fully grounded.

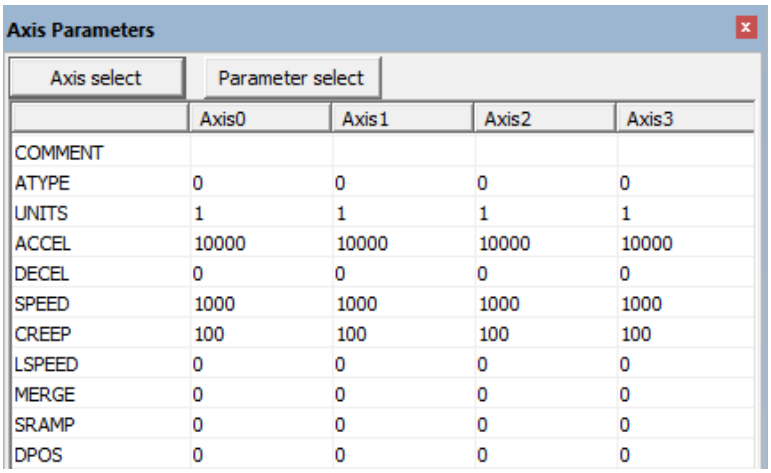
3.11.2. Basic Usage

(1) Please follow the above wiring instructions for correct wiring.

(2) After power on, please use any of the three interfaces ETHERNET, RS232 (default parameter, it can be connected directly) and RS485 (default parameter can be connected directly, one adapter is needed for hardware) to connect to RTSys.

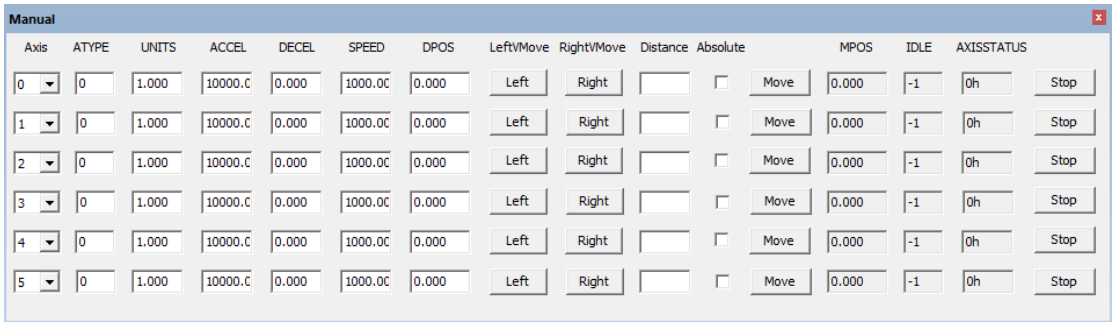
(3) Set axis parameters, such as, ATYPE, UNITS, SPEED, ACCEL, FWD_IN, REV_IN, etc.

(4) There are many parameters related to pulse axis, they can be set and checked through relative instructions, please see "axis parameter and axis status" of "RTBasic", or see "RTSys/Tool/Axis parameter".



	Axis0	Axis1	Axis2	Axis3
COMMENT				
ATYPE	0	0	0	0
UNITS	1	1	1	1
ACCEL	10000	10000	10000	10000
DECEL	0	0	0	0
SPEED	1000	1000	1000	1000
CREEP	100	100	100	100
LSPEED	0	0	0	0
MERGE	0	0	0	0
SRAMP	0	0	0	0
DPOS	0	0	0	0

(5) Control corresponding motion through “Tool – Manual”.



Refer to BASIC Routine:

BASE(0,1)	'select axis 0 and axis 1
ATYPE = 1,1	'set axis 0 and axis as pulse axes
UNITS = 1000,1000	'set pulse amount as 1000 pulses
SPEED = 10,10	'set axis speed as 10*1000 pulse/s
ACCEL = 1000,1000	‘set axis acceleration as 1000*1000 pulse/s/s
FWD_IN = -1,-1	'prohibit using axis positive hardware position limit
REV_IN = -1,-1	'prohibit using axis negative hardware position limit
MOVE(10) AXIS(0)	'axis 0 moves distance of 10*1000 pulses in positive
MOVE(-20) AXIS(0)	'axis 0 moves distance of 20*1000 pulses in negative

Chapter IV Expansion Module

The controller can expand digital IO, analog IO, pulse axis and other resources through CAN bus (ZIO series expansion modules), EIO series EtherCAT bus expansion modules or ZMIO310 vertical bus expansion modules. For details, please refer to each manual.

4.1. CAN Bus Expansion

ZIO series expansion modules or ZMIO310-CAN coupler with sub modules can be used.

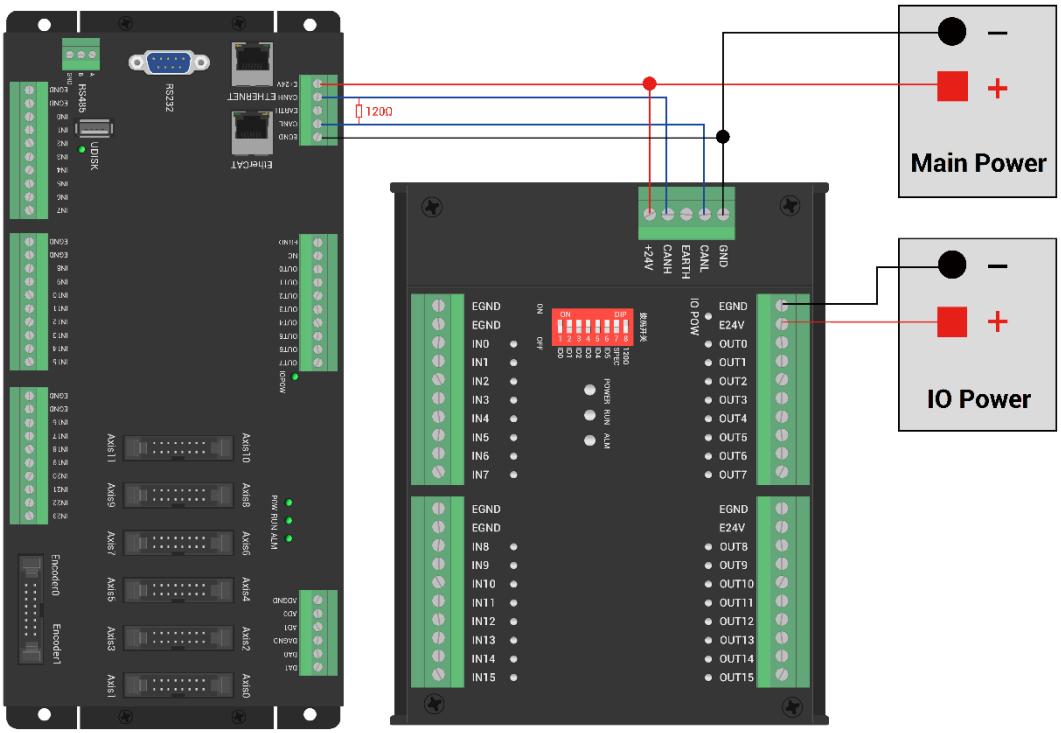
4.1.1. CAN Bus Expansion Wiring

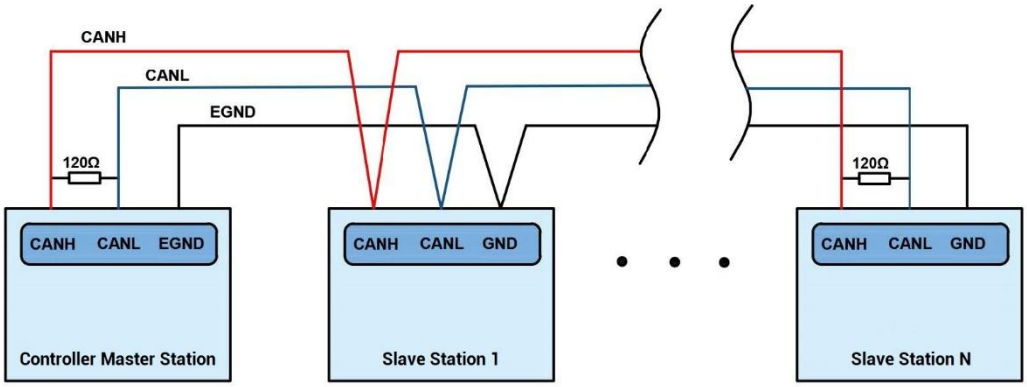
The ZIO expansion module is powered by the dual power supply. Except the main power supply, an additional IO power supply is required to supply independent power for IO. Both the main power supply and the IO power supply use 24V DC power supply. For ZAIO, it only needs to connect to the main power supply.

To prevent interference, separate the IO power supply from the main power supply.

Please select the expansion module according to the requirements, and select IO mapping or axis mapping according to the resources of the expansion module.

Wiring reference of connection between ZIO expansion module and control card and standard wiring of CAN bus are shown as below:





→ Wiring Note:

✧ ZMC212BE controller uses the single power, and CAN expansion module uses dual-power. When using, main power supply of expansion module and main power supply of controller can share one power. When they use different power supplies, controller power EGND needs to connect to expansion module power GND, otherwise CAN may be burnt out.

✧ When connecting multiple CAN expansion modules on the CAN bus, a 120-ohm resistor needs to be connected in parallel between the CANL and CANH terminals, for the expansion module that is with 8-digit dialing codes, the terminal resistor can be realized by dialing the code (DIP).

4.1.2. CAN Bus Expansion Resource Mapping

The ZCAN expansion module generally has an 8-bit DIP switch, dial ON to take effect, and the meaning of the DIP is as follows:

1-4: they are used for ZCAN expansion module IO address mapping, the corresponding value is 0-15.

5-6: CAN communication speed, corresponding value is 0-3, four different speeds are optional.

7: reserved.

8: 120-ohm resistor, dial ON means a 120-ohm resistor is connected between CANL and CANH.

The IO numbers of the entire control system cannot be repeated, and existed numbers must be avoided when mapping resources. And the DIP switch must be dialed before power-on, if re-dial after power-on, it is invalid. It needs to be powered on again to take effect.

Dial 1-4 to select the CAN address, and the controller sets the IO number range of the corresponding expansion module according to the CAN DIP address. When each is dialed as OFF, the corresponding value is 0, when it is ON , it corresponds to a value of 1, and the address combination value = dial 4 × 8 + dial code 3 × 4 + dial code 2 × 2+ dial code 1.

Dial code 5-6 to select CAN bus communication speed, speed combination value=dial code 6 × 2 + dial code 5×1, the combined value range is 0-3.

The corresponding speeds are as follows:

DIP 5-6 combination value	CANIO_ADDRESS high 8-bit value	CAN communication speed
0	0 (corresponds to decimal 128)	500KBPS (default value)
1	1 (corresponds to decimal 256)	250KBPS

2	2 (corresponding to decimal 512)	125KBPS
3	3 (corresponding to decimal 768)	1MBPS

The controller side sets the CAN communication speed through the CANIO_ADDRESS command. There are also four speed parameters that can be selected. The communication speed must be consistent with the communication speed of the expansion module that corresponds to the combination value, then they can communicate with each other.

The factory default communication speed is 500 KBPS on both sides, there is no need to set this, unless you need to change the speed.

The CANIO_ADDRESS command is a system parameter, and it can set the master-slave end of CAN communication. The default value of the controller is 32, that is, CANIO_ADDRESS=32 is the master end, and the slave end is set between 0-31.

The CAN communication configuration can be viewed in the "State the Controller" window.

→ IO Mapping:

the CAN expansion module uses bit1-4 of the DIP switch. According to the number of currently included IO points(the largest number in IN and OP must include IO point in the axis interface), use the bit 1-4 to set the ID, so as to determine the number range of IO to be expanded.

If the controller itself contains 28 INs and 16 OPs, then the starting address set by the first extended board should exceed the maximum value of 28. According to below rule, the dial code should be set to the combination value 1 (binary combination value 0001, from right to left, dial code 1-4, at this time dial 1 is set to ON, and the others are set to OFF), the IO number on the expansion board = the expansion board number value + the initial IO number value, among them, the IOs that are vacant from 29-31 Numbers are not used. Subsequent extended boards continue to confirm the dial settings according to the IO points in turn.

The initial digital IO mapping number starts from 16 and increases in multiples of 16. The distribution of digital IO numbers corresponding to different dial IDs is as follows:

DIP 1-4 combination value	Starting IO number	Ending IO number
0	16	31
1	32	47
2	48	63
3	64	79
4	80	95
5	96	111
6	112	127
7	128	143
8	144	159
9	160	175
10	176	191
11	192	207
12	208	223
13	224	239
14	240	255
15	256	271

The initial IO mapping number of the analog AD starts from 8 and increases in multiples of 8. The initial IO mapping number of the analog DA starts from 4 and increases in multiples of 4. The allocation of digital IO numbers corresponding to different dial code IDs is as follows:

DIP 1-4 combination value	Starting AD number	End AD number	Starting DA number	End DA number
0	8	15	4	7
1	16	23	8	11
2	24	31	12	15
3	32	39	16	19
4	40	47	20	23
5	48	55	24	27
6	56	63	28	31
7	64	71	32	35
8	72	79	36	39
9	80	87	40	43
10	88	95	44	47

11	96	103	48	51
12	104	111	52	55
13	112	119	56	59
14	120	127	60	63
15	128	135	64	67

→ Axis Mapping:

When the CAN bus expansion mode is used to expand the pulse axis, ZIO16082M can be selected to expand two pulse axes. These two pulse axes need to be mapped and bound with the axis No., then access.

Extended axes need to perform axis mapping operations, using the AXIS_ADDRESS command to map, and the mapping rules are as follows:

AXIS_ADDRESS(axis No.)=(32*0)+ID

'the local axis interface of the expansion module AXIS 0

AXIS_ADDRESS(axis No.)=(32*1)+ID

'the local axis interface of the expansion module AXIS 1

The ID is the combined value of the DIP bit1-4 of the expansion module. After the mapping is completed and the axis parameters such as ATYPE are set, the expansion axis can be used.

Example:

```
ATYPE(6)=0                    'set as virtual axis
AXIS_ADDRESS(6)=1+(32*0)
        'ZCAN expansion module ID 1 axis 0 is mapped to axis 6
ATYPE(6)=8      'ZCAN extended axis type, pulse direction stepping or servo
UNITS(6)=100 0 'pulse equivalent 1000
SPEED(6)=100    'speed 100uits/s
ACCEL(6)=1000   'acceleration 1000units/s^2
MOVE(100) AXIS(6) 'extended axis movement 100units
```

Extended resource viewing:

According to the CAN connection, after the power is turned on, and the wiring

resistance dial code is set correctly, the power indication led (POWER) and the running indication led (RUN), the IO power indication led (IO POWER) are on, and the alarm indication led (ALM) is off. At the same time, the "Controller" - "State the controller" - "ZCanNodes" in the RTSys software displays the expansion module information and the extended IO number range.

The dial ID and the corresponding resource number when connecting multiple expansion modules are as follows:

Local	432-0(ZMC432)	32	30(0-29)	18(0-17)	0	2(0-1)
1	48(ZIO1632)	0	16(32-47)	32(32-63)	0	0
3	26(ZIO16082)	2	16(64-79)	8(64-71)	0	0
4	10(ZAIO0802)	0	0	0	8(40-47)	2(20-21)

ALM indicator light is on, please check whether the wiring, resistor and dial setting are correct, and whether the CANIO_ADDRESS command of the controller is set as the master end (32), and whether the CAN communication speed is consistent.

4.2. EtherCAT Bus Expansion

The EIO expansion modules and ZMIO310-ECAT are expansion modules used by the EtherCAT bus controller. For example, EIO series can expand the resources of digital IO and pulse axis. When the resources of the controller are insufficient, the EtherCAT bus controller can be connected to multiple EIO expansion modules for expansion, you can view the maximum number of IO expansion points and the maximum number of expansion axes of the controller, and in this way, it supports IO remote expansion.

4.2.1. EtherCAT Bus Expansion Wiring

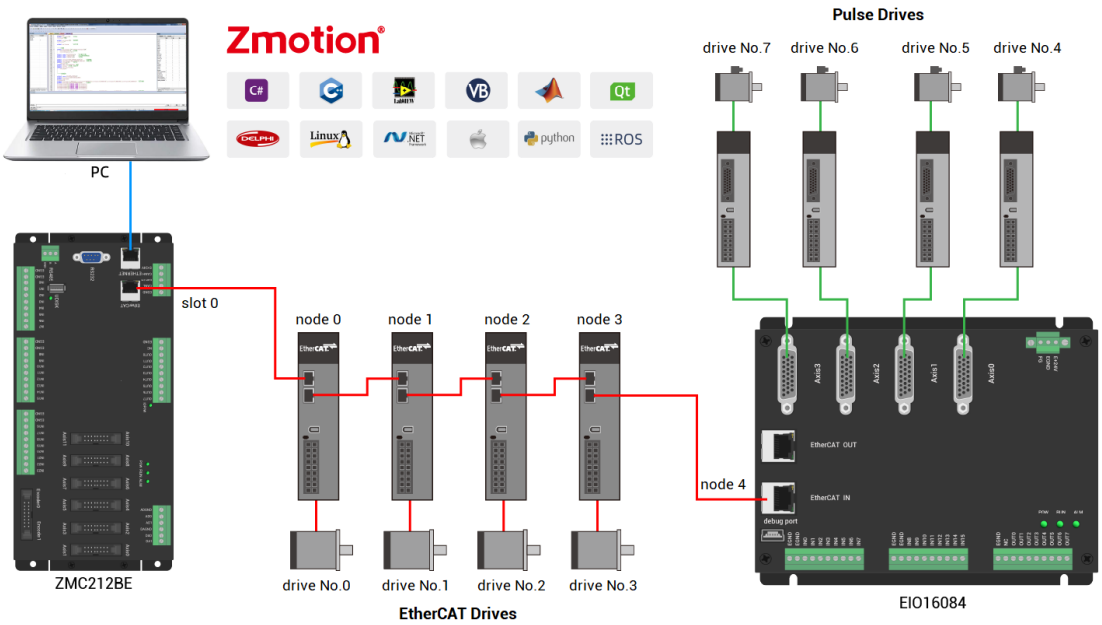
After the expansion wiring is completed, each EIO expansion module does not need to develop again. It only needs to manually configure the unique IO address and axis address in the EtherCAT master controller, and it can be accessed after the configuration is completed.

The IO address number is set through the bus command NODE_IO, and the program on the controller can access the resources on the expansion module only through the IO number. The configuration of the axis address uses the AXIS_ADDRESS command to map

axis number, and when the binding is completed, specify the axis number through the BASE or AXIS command.

When wiring, pay attention that EtherCAT IN is connected to the upper-level module, and EtherCAT OUT is connected to the lower-level module. The IN and OUT ports cannot be mixed.

EIO expansion module wiring reference example:



Involved number concepts in above figure are as follows: the bus-related command parameters will use the following numbers:

Slot No. (slot):

The slot number refers to the number of the bus interface on the controller, and the slot number of the EtherCAT bus is 0.

Device No. (node):

The device number refers to the number of all devices connected to a slot. It starts from 0 and is automatically numbered according to the connection sequence of the devices on the bus. You can view the total number of devices connected to the bus through the NODE_COUNT(slot) command.

Drive No:

The controller will automatically identify the drive on the slot, and the number starts

from 0, and the number is automatically numbered according to the connection sequence of the drive on the bus.

The drive number is different from the device number. Only the drive device number on the slot is assigned, and other devices are ignored. The drive number will be used when mapping the axis number.

4.2.2. EtherCAT Bus Expansion Resource Mapping

→ IO Mapping:

The program on the controller can access the resources on the expansion module only through the IO number. The IO number of the EtherCAT bus expansion module is set through the bus command NODE_IO, and the input and output are configured at the same time.

When IO mapping, first check the maximum IO number of the controller itself (including the external IO interface and the interface in the pulse axis), and then use the command to set.

If the extended IO coincides with the IO number of the controller itself, the two will work at the same time, so the mapped number of the IO mapping must not be repeated in the entire control system.

IO mapping syntax:

NODE_ IO(slot, node) = iobase

slot: slot No., 0-default

node: device No, starting from 0

iobase : mapping the IO start No, the setting result will only be a multiple of 8

Example:

NODE_IO(0,0)=32 'set the IO starting No. of slot 0 interface device 0 to 32

If device 0 is EIO16084, after configuration according to the above syntax, the IO numbers corresponding to input IN0-15 are 32-47 in turn, the general input port numbers in the axis interface are 48-55, and the drive alarm inputs of axes AXIS 0-3 are 48-51

respectively. The IO numbers corresponding to the output OUT0-7 are 32-39 in sequence, the general output port numbers in the axis interface are 40-47, and the drive enable outputs of the axes AXIS 0-3 are 40-43 respectively.

0	41bh	1918h	0	4	24(32-55)	16(32-47)	0

→ AXIS Mapping:

Before using the axis of the expansion module, you need to use the AXIS_ADDRESS command to map the axis number, and the axis mapping also needs to pay attention to the axis number of the entire system cannot be repeated. The mapping syntax of the EIO series extended axis is the same as that of the bus driver.

Axis mapping syntax:

AXIS_ADDRESS(axis No.)=(slot No.<<16)+driver No.+1

Example:

AXIS_ADDRESS(0)=(0<<16)+0+1

'the first drive on the EtherCAT bus, drive No. 0 is bound as axis 0

AXIS_ADDRESS(1)=(0<<16)+1+1

'the second drive on the EtherCAT bus, drive No. 1 is bound as axis 1

If the first node is EIO16084, and EIO16084 is connected to drive, then driver 0 here is the first pulse driver connected to EIO16084, otherwise it is the EtherCAT driver.

Chapter V Programming

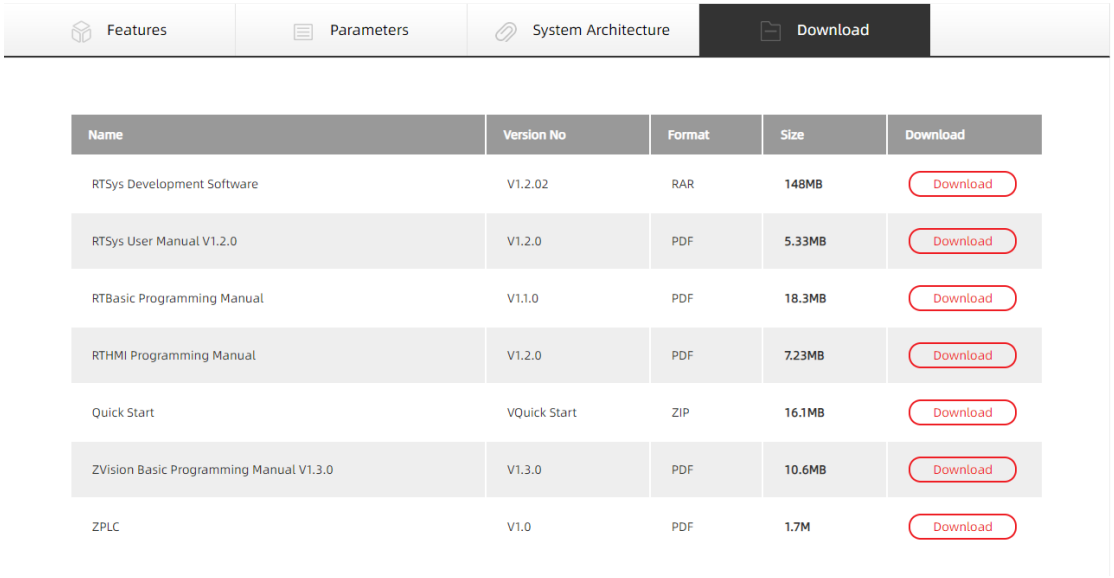
5.1. Program in RTSys Software

RTSys is a PC-side program development, debugging and diagnostic software for the Zmotion motion controllers. Through it, users can easily edit and configure the controller program, quickly develop applications, diagnose system operating parameters in real time, and debug the running program in real time. What's more, it supports Chinese and English bilingual environments.

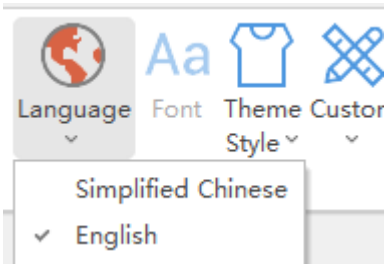
In RTSys, there are 4 programming languages for motion control development, Basic, PLC, HMI and C language, they can run multi-tasks among them, especially for Basic, multi-task running can be achieved separately, hybrid programming is also OK with PLC, HMI and C language.

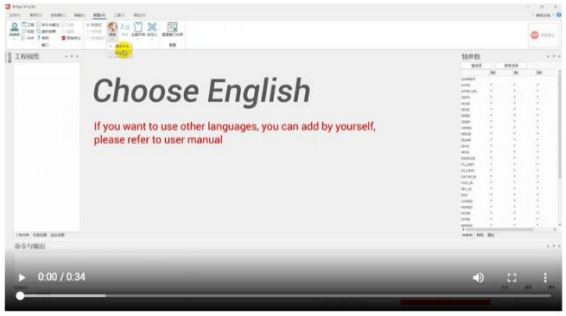
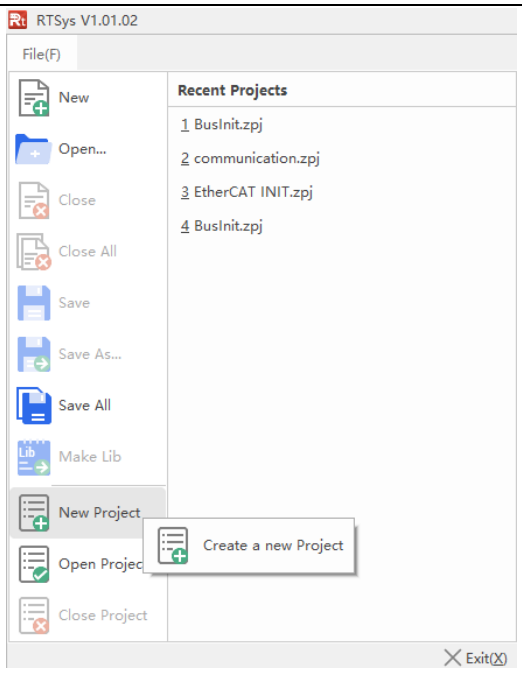
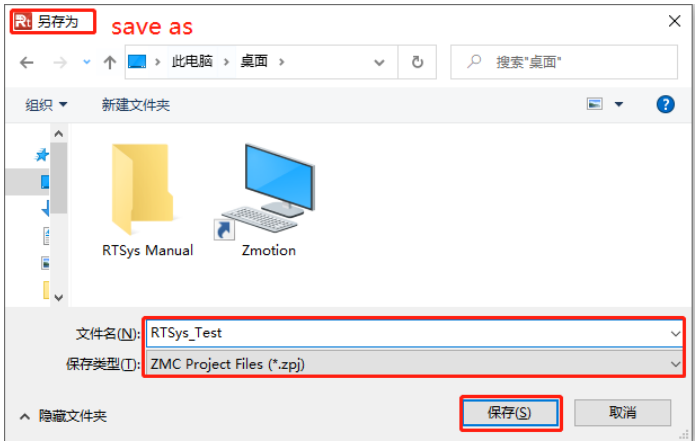
RTSys Downloading Address: https://www.zmotionglobal.com/pro_info_282.html

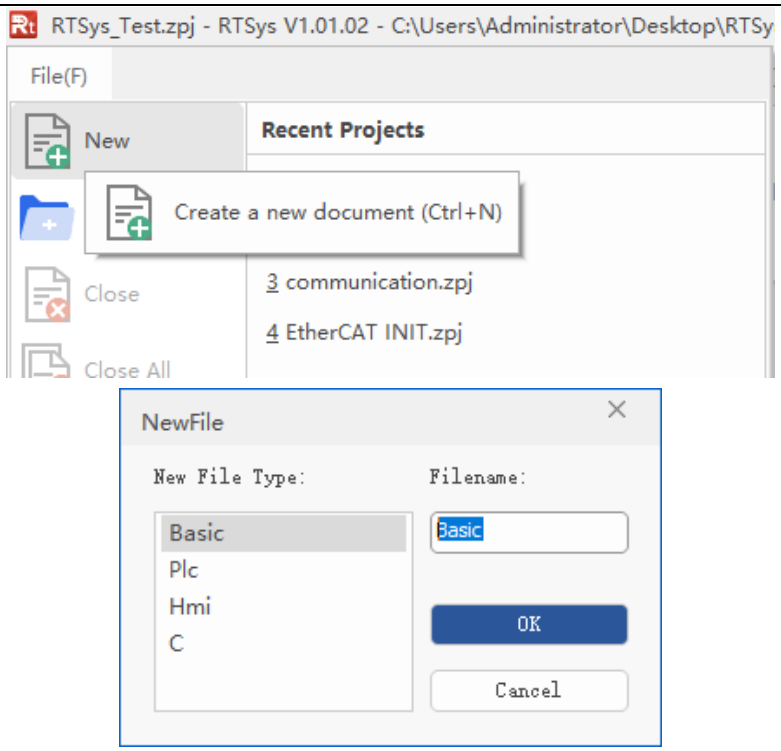
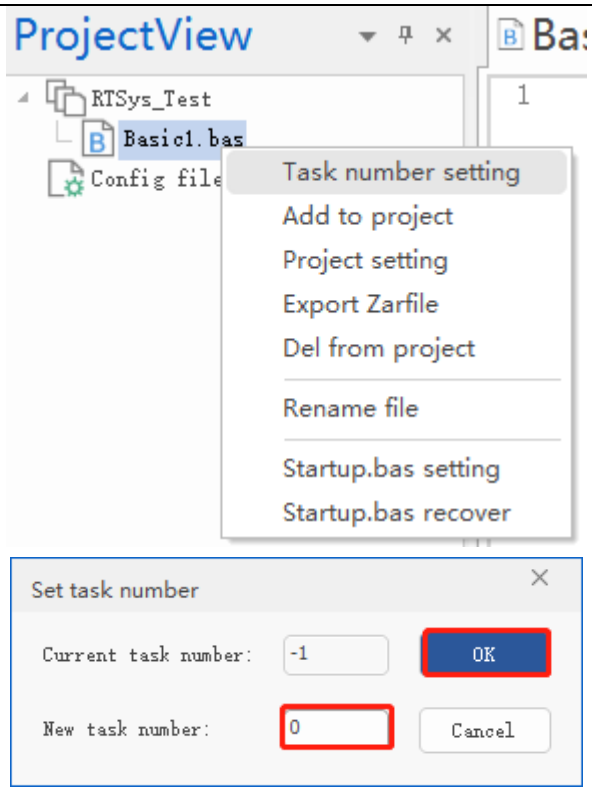
And related manuals can be found in "Download":

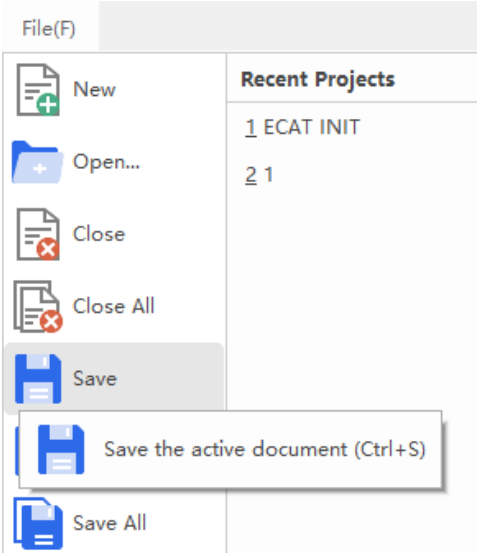
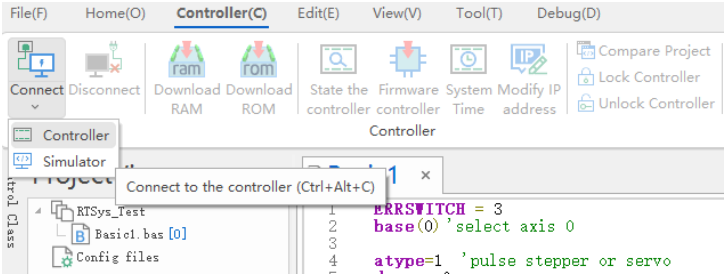
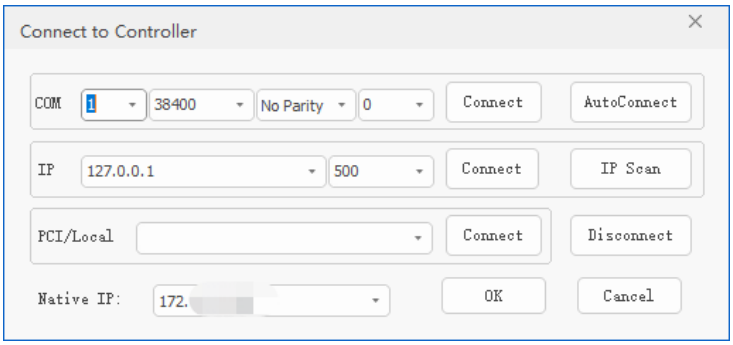


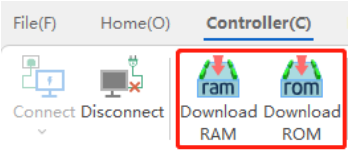
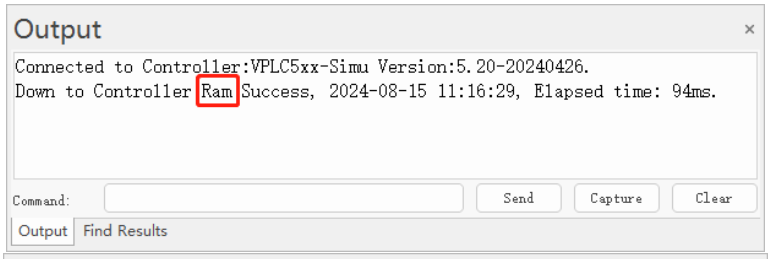
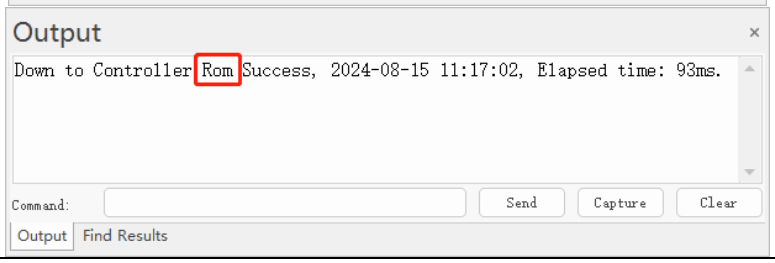
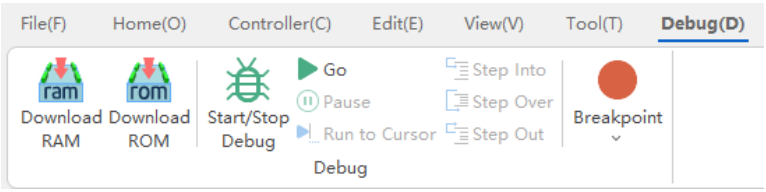
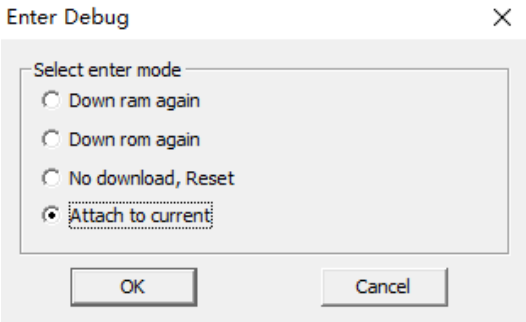
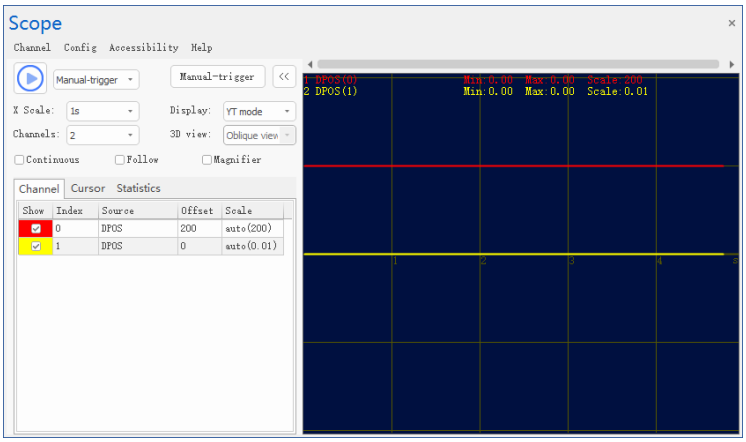
Name	Version No	Format	Size	Download
RTSys Development Software	V1.2.02	RAR	148MB	Download
RTSys User Manual V1.2.0	V1.2.0	PDF	5.33MB	Download
RTBasic Programming Manual	V1.1.0	PDF	18.3MB	Download
RTHMI Programming Manual	V1.2.0	PDF	7.23MB	Download
Quick Start	VQuick Start	ZIP	16.1MB	Download
ZVision Basic Programming Manual V1.3.0	V1.3.0	PDF	10.6MB	Download
ZPLC	V1.0	PDF	1.7M	Download

Step	Operations	Display Interface
1	Switch the Language: "Language" – "English", then there will pop	

	up one window, click OK, and restart it.	Language Switch Video Showing: E. How to Switch the Language Find "视图" (the fourth one in the above menu), then find the "语言", choose English, restart RTSys. Enligsh RTSys will take effect when opened again. 
2	New Project: "File" – "New Project", Save as window will pop up, then enter file name, save the project file with suffix "zpj.".	 

3	New File: “File” – “New File”, select file type to build, here select Basic, click “OK”.	
4	Set Auto Run No.: right click the file, open task number setting window, enter task No., which can be any + value, no priority, but not the same.	

5	Save File: edit the program in program editing window, click "save", new built file will be saved under "zpj." project automatically. "Save all" means all files under this project will be saved.	
6	Connection: Click "controller – connect", if no controller, select connect to simulator.	
	Then, "connect to controller" window will pop up, you can select serial port or net port to connect, select matched serial port parameters or net port IP address, then click "connect".	
7	Download Program into	● **RAM:** it will not save when power off. ● **ROM:** it will save data when power off, and when the program

	Controller: "Ram/Rom" – "download RAM / download ROM", if it is successful, there is print indication, at the same time, program is downloaded into controller and runs automatically.	is connected to controller again, running according to task No.   
8	Debug: "Debug" – "Start/Stop Debug" to call "Task" and "Watch" window, because it was downloaded before, here select "Attach the current".	 
9	Scope function: Click "View" – "Scope" to open oscilloscope. It can capture needed data, for debugging.	

Notes:

- When opening an project, choose to open the zpj file of the project. If only the Bas file is opened, the program cannot be downloaded to the controller.
- When the project is not created, only the Bas file cannot be downloaded to the controller.
- The number 0 in automatic operation represents the task number, and the program runs with task 0, and the task number has no priority.
- If no task number is set for the files in the entire project, when downloading to the controller, the system prompts the following message WARN: no program set autorun

5.2. Upgrade Controller Firmware

Firmware upgrade can be achieved by downloading zfm firmware package in RTSys. zfm file is the firmware upgrade package of controller, please select corresponding firmware because different models are with different packages, please contact manufacturer).

How to update:

a. Open ZDevelop / RTSys software, then click "controller – connect", find PCI/LOCAL method, click "connect". If connected, there will be "Connected to Controller: PCIE464 Version: 4.93 – 20231220." In "output" window.

b. Click "controller – state the controller", find basic info, then current software version can be checked.

c. Click "controller – update firmware", current controller model and software version can be viewed.

d. Click "browse", and select saved firmware file, click "update", then one window will pop up, please click "ok".

e. After that, "connect to controller" window appears again, and please select "PCI/Local" again, and click "connect".

f. When connection is successful, "firmware update" interface is shown. Now

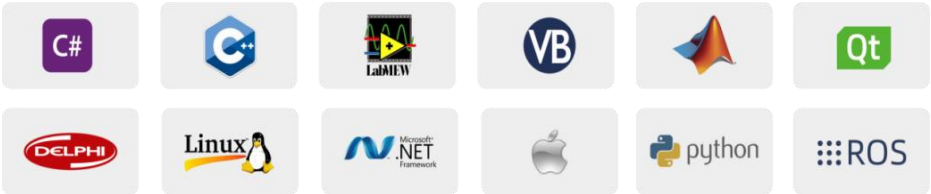
system enters ZBIOS state, please click “update” again.

g. When it is loaded, “firmware update” window disappears, now in output window, it shows “Update firmware to Controller Success”.

h. Do step a and step b again, check whether the firmware is updated or not.

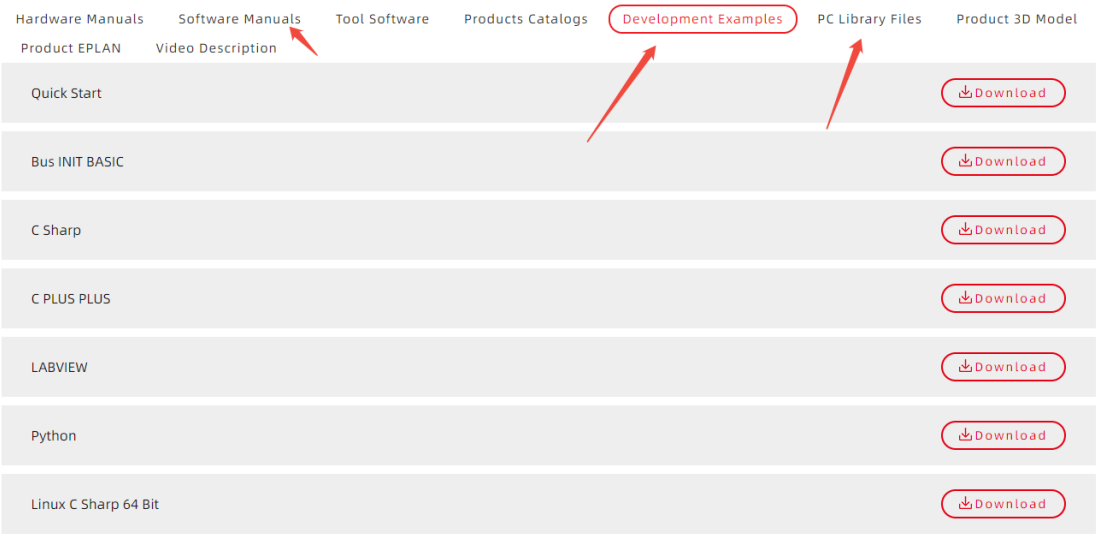
5.3. Program in Host-Computer by PC Languages

The controller supports development under various operating systems such as windows, linux, Mac, Android, and wince, and provides dll libraries in various environments such as vc, c#, vb.net, and labview, as shown in the figure below. PC software programming refers to "Zmotion PC Function Library Programming Manual".

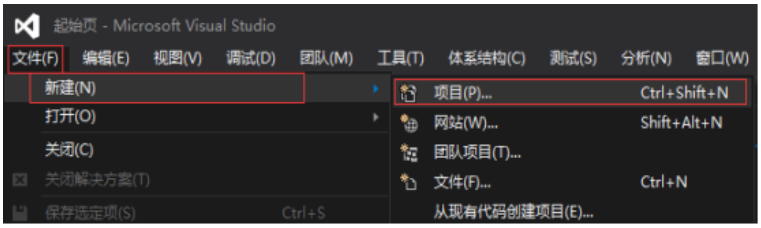
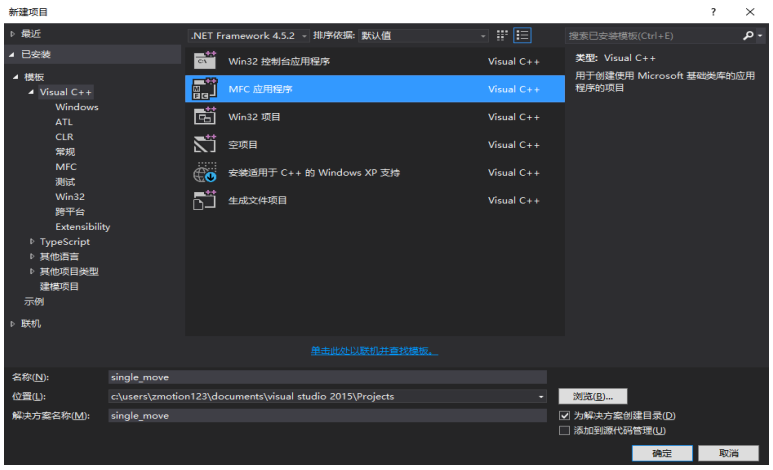
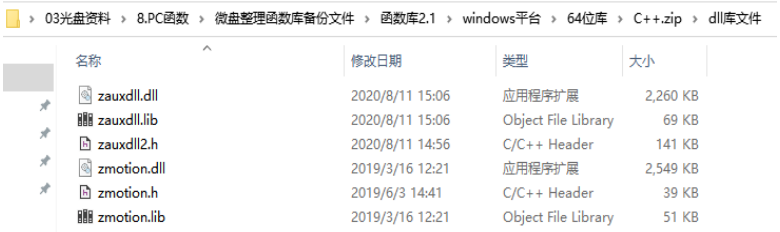


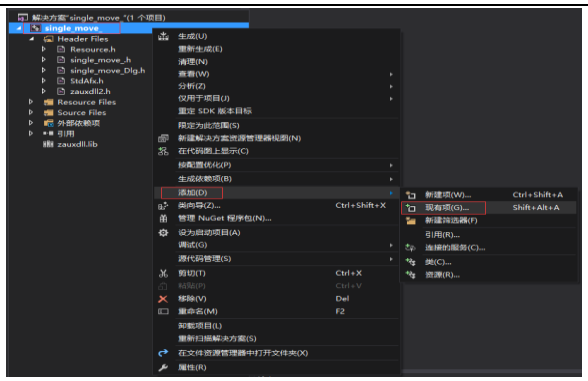
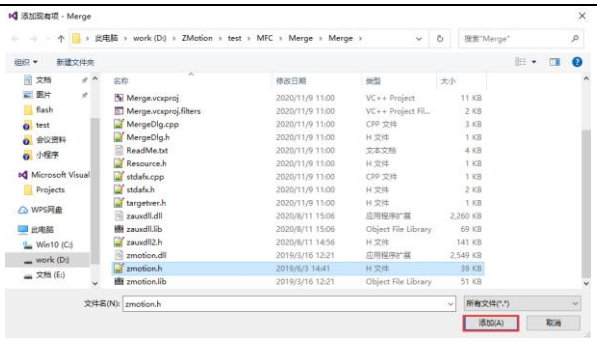
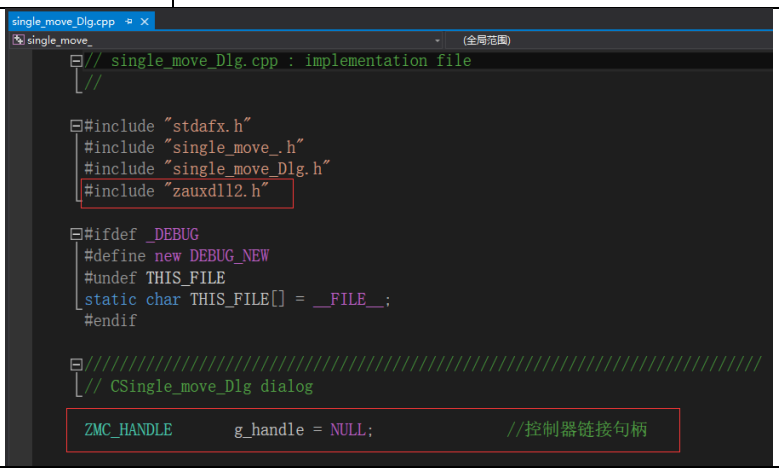
The program developed using the PC software cannot be downloaded to the controller, and it is connected to the controller through the dll dynamic library. The dll library needs to be added to the header file and declared during development.

- Get PC library file, example: https://www.zmotionglobal.com/download_list_17.html



The c++ project development process in VS is as follows:

Step	Operations	Display Interface
1	Open VS, click “File” – “New” – “Project”.	
2	Select development language as “Visual C++” and the select program type as “MFC application type”.	
3	Select “Based on basic box”, click “next” or “finish”	
4	Find C++ function library provided by manufacturer. Routine is below (64-bit library)	
5	Copy all DLL related library files under the above path to the newly created project.	

6	Add a static library and related header files to the project. Static library: zauxdll.lib, zmotion.lib Related header files: zauxdll2.h, zmotion.h	1) Right-click the header file first, and then select: "Add" → "Existing Item".	
		2) Add static libraries and related header files in sequence in the pop-up window.	
7	Declare the relevant header files and define the controller connection handle, so far the project is newly created.		

```
// single_move_Dlg.cpp : implementation file
//

#include "stdafx.h"
#include "single_move_.h"
#include "single_move_Dlg.h"
#include "zauxdll2.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

/////////////////////////////////////////////////////////////////////////////
// CSingle_move_Dlg dialog

ZMC_HANDLE        g_handle = NULL;              //控制器链接句柄
```

Chapter VI Operation and Maintain

The correct operation and maintenance of the device can not only guarantee and extend the life cycle of the equipment itself, but also take technical management measures according to the pre-specified plan or the corresponding technical conditions to prevent equipment performance degradation or reduce the probability of equipment failure.

6.1. Regular Inspection and Maintenance

The working environment has an impact on the device. Therefore, it is usually inspected regularly based on the inspection cycle of 6 months to 1 year. The inspection cycle of the device can be appropriately adjusted according to the surrounding environment to make it work within the specified standard environment.

Check item	Check content	Inspection standards
power supply	Check whether the voltage is rated	DC 24V (-5%~5%)
surroundings	Whether the ambient temperature is within the specified range (when installed in the cabinet, the temperature inside the cabinet is the ambient temperature)	-10°C - 55°C
	Whether the ambient humidity is within the specified range (when installed in the cabinet, the humidity in the cabinet is the ambient humidity)	10%-95% non-condensing
	Is there direct sunlight	No
	With or without droplets of water, oil, chemicals, etc.	No
	Whether there is dust, salt, iron filings, dirt	No
	Whether there is corrosive gas	No
	Whether there are flammable and	No

	explosive gases or articles	
	Whether the device is subjected to vibration or shock	Should be within the range of vibration resistance and impact resistance
	Is the heat dissipation good	Keep good ventilation and heat dissipation
Installation and Wiring Status	Whether the basic unit and the expansion unit are installed firmly	The mounting screws should be tightened without loosening
	Whether the connecting cables of the basic unit and the expansion unit are fully inserted	The connection cable cannot be loosened
	Are the screws of the external wiring loose	Screws should be tightened without loosening
	Whether the cable is damaged, aged, cracked	The cable must not have any abnormal appearance

6.2. Common Problems & Solutions

Problems	Suggestions
Motor does not rotate.	1. Check whether the ATYPE of the controller is correct. 2. Check whether hardware position limit, software position limit, alarm signal work, and whether axis states are normal. 3. Check whether motor is enabled successfully. 4. Confirm whether pulse amount UNITS and speed values are suitable. If there is the encoder feedback, check whether MPOS changes. 5. Check whether pulse mode and pulse mode of drive are matched. 6. Check whether alarm is produced on motion controller station or drive station. 7. Check whether the wiring is correct. 8. Confirm whether controller sends pulses normally.

The position limit signal is invalid.	1. Check whether the limit sensor is working normally, and whether the "input" view can watch the signal change of the limit sensor. 2. Check whether the mapping of the limit switch is correct. 3. Check whether the limit sensor is connected to the common terminal of the controller.
No signal comes to the input.	1. Check whether the limit sensor is working normally, and whether the "input" view can watch the signal change of the limit sensor. 2. Check whether the mapping of the limit switch is correct. 3. Check whether the limit sensor is connected to the common terminal of the controller.
The output does not work.	1. Check whether IO power is needed. 2. Check whether the output number matches the ID of the IO board.
POWER led is ON, RUN led is OFF.	1. Check whether the power of the power supply is sufficient. At this time, it is best to supply power to the controller alone, and restart the controller after adjustment. 2. Check whether the ALM light flickers regularly (hardware problem).
RUN led is ON, ALM led is ON.	1. Program running error, please check RTSys error code, and check application program.
Fail to connect controller to PC through serial port.	1. Check whether the serial port parameters are modified by the running program, you can check all the current serial port configurations through ?*SETCOM. 2. Check whether the serial port parameters of the PC match the controller. 3. Open the device manager and check whether the serial driver of the PC is normal.
CAN expansion module cannot be connected.	1. Check the CAN wiring and power supply circuit, whether the 120 ohm resistor is installed at both

	ends. 2. Check the master-slave configuration, communication speed configuration, etc. 3. Check the DIP switch to see if there are multiple expansion modules with the same ID. 4. Use twisted-pair cables, ground the shielding layer, and use dual power supplies for severe interference (the main power supply of the expansion module and the IO power supply are separately powered)
Fail to connect controller to PC through net port.	1. Check IP address of PC, it needs to be at the same segment with controller IP address. 2. Check controller IP address, it can be checked and captured after connection through serial port. 3. When net port led is off, please check wiring. 4. Check whether controller power led POWER and running indicator led RUN are ON normally. 5. Check whether the cable is good quality, change one better cable to try again. 6. Check whether controller IP conflicts with other devices. 7. Check whether controller net port channel ETH are all occupied by other devices, disconnect to other devices, then try again. 8. When there are multiple net cards, don't use other net cards, or change one computer to connect again. 9. Check PC firewall setting. 10. Use “Packet Internet Groper” tool (Ping), check whether controller can be Ping, if it can't, please check physical interface or net cable. 11. Check IP address and MAC address through arp-a.