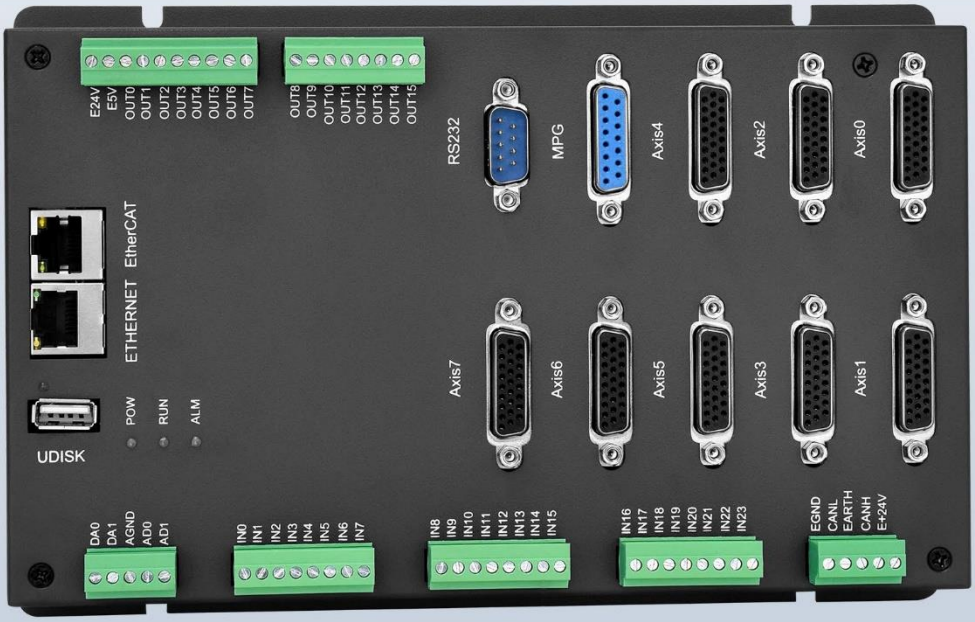
Zmotion®
正运动技术

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

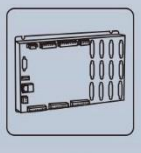
Pulse & EtherCAT Motion Controller

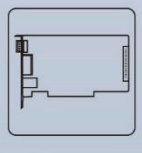
ZMC408CE

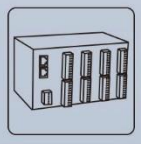


This manual is mainly for ZMC408CE, ZMC408CE-16.

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.....7
1.4. Hardware Installment.....7
Chapter II Product Specification.....9
2.1. Basic Specification.....9
2.2. Nameplates & Models.....10
2.3. Interface Definition.....11
2.4. Work Environment.....12
Chapter III Wiring & Communication.....14
3.1. Power Input, CAN Communication Interface.....14
3.1.1. Power Specification.....14
3.1.2. CAN Communication Specification & Wiring.....14
3.1.3. Basic Usage Method.....16
3.2. IN: Digital Input & High-Speed Latch Port.....17
3.2.1. Digital Input Specification & Wiring.....18
3.2.2. Basic Usage Method.....19
3.3. OUT: Digital Output & PWM & Hardware Comparison Output.....20
3.3.1. Digital Output Specification & Wiring.....21
3.3.2. Basic Usage Method.....23
3.4. AD/DA Analog Input/Output.....23
3.4.1. AD/DA Analog Input/Output Specification & Wiring.....24
3.4.2. Basic Usage Method.....25
3.5. RS232/RS485 Serial Port.....25
3.5.1. RS232/RS485 Specification & Wiring.....26
3.5.2. Basic Usage Method.....27
3.6. U Disk.....28
3.7. ETHERNET.....29
3.8. EtherCAT Bus Interface.....30
3.9. AXIS Differential Pulse Axis Interface.....32

3.9.1. AXIS Interface Signal Specification & Wiring....34
3.9.2. Basic Usage Method....37
3.10. MPG Handwheel Interface....39
3.10.1. MPG Specification & Wiring....39
3.10.2. Basic Usage Method....41
Chapter IV Expansion Module....43
4.1. CAN Bus Expansion....43
4.1.1. CAN Bus Expansion Wiring....43
4.1.2. CAN Bus Expansion Resource Mapping....45
4.2. EtherCAT Bus Expansion....49
4.2.1. EtherCAT Bus Expansion Wiring....49
4.2.2. EtherCAT Bus Expansion Resource Mapping....51
5.1. Program in RTSys Software....53
5.2. Upgrade Controller Firmware....58
5.3. Program in Host-Computer by PC Languages....59
Chapter VI Operation and Maintain....62
6.1. Regular Inspection and Maintenance....62
6.2. Common Problems & Solutions....63

Chapter I Production Information

1.1. Product Information

ZMC is the abbreviation of the motion controller model launched by Zmotion Technology.

ZMC408CE high-performance multi-axis motion controller is a stand-alone motion controller that is compatible with EtherCAT bus and pulse type. The controller itself supports 8 axes, but 32 axes can be expanded at most to achieve complex continuous trajectory control requirements.

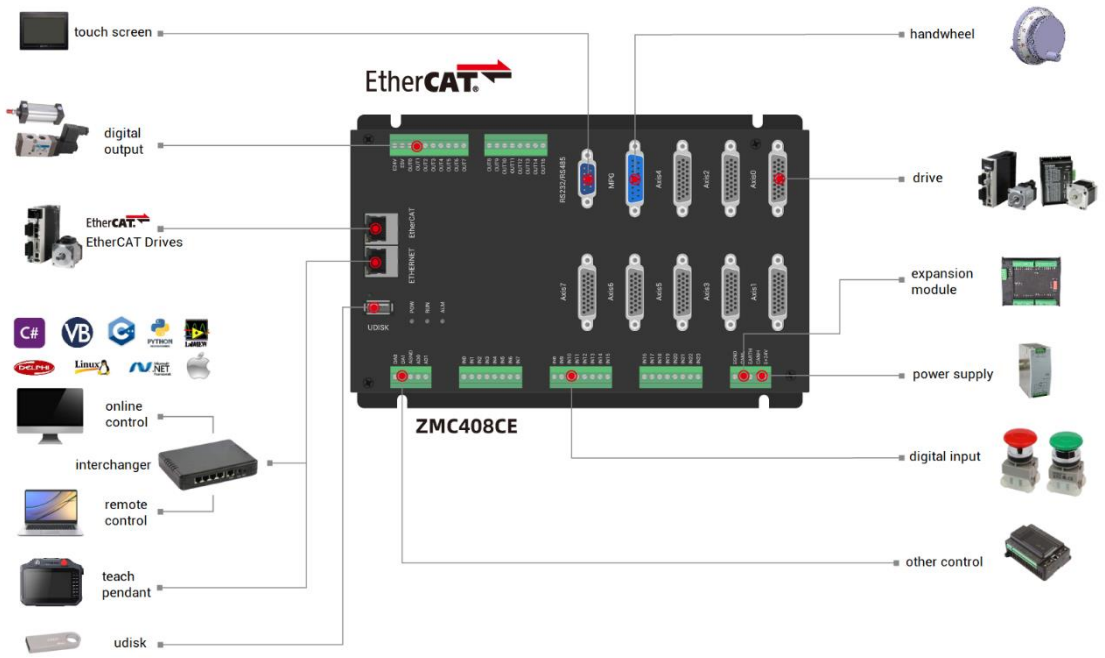
ZMC4 series high-performance multi-axis motion controllers can be applied in robots (SCARA, Delta, 6 joints), electronic semiconductor equipment (testing equipment, assembly equipment, locking equipment, soldering machine), dispensing equipment, non-standard equipment, printing and packaging equipment, textile and garment equipment, stage entertainment equipment, medical equipment, assembly line, etc.

1.2. Function Features

- Support 8 axes motion control.
- Pulse output mode: pulse / direction or dual pulses or quadrature pulse.
- AXIS interface supports encoder position measurement, which can be configured as handwheel input mode.
- Specialized handwheel input interface.
- Maximum pulse output frequency of each axis is 10MHZ.
- IO can be expanded through CAN and EtherCAT bus, 4096 isolated inputs and 4096 isolated outputs can be extended at most.

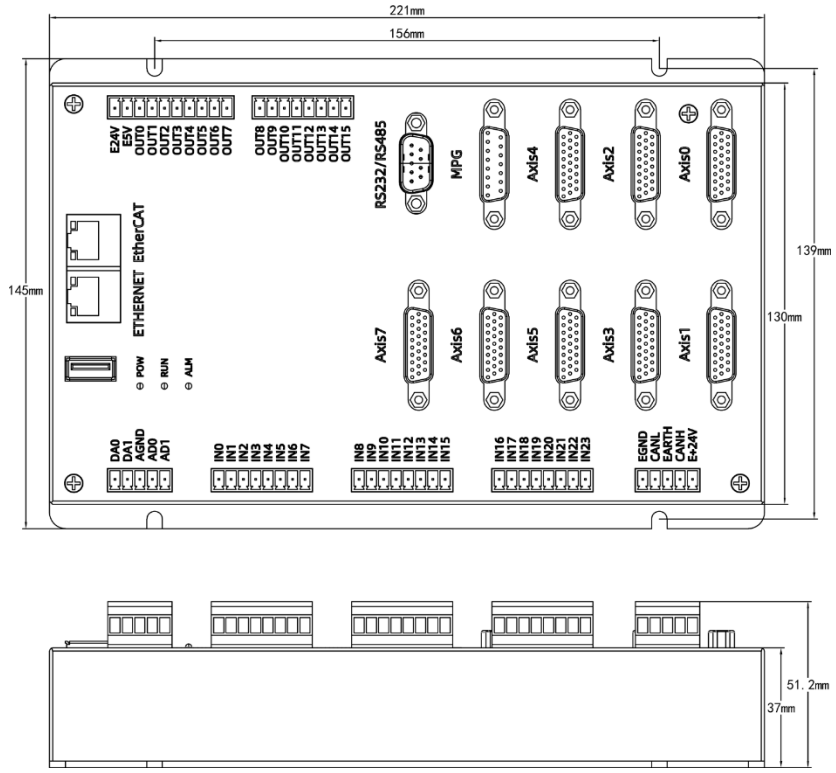
- Axis position limit signal / origin signal port can be configured as any input at will.
- The maximum output current of general digital outputs can reach 300mA, which can directly drive some kinds of solenoid valves.
- Interfaces: RS232, EtherNET, EtherCAT, CAN.
- Support up to 32 axes linear interpolation, arbitrary circular interpolation, helical interpolation.
- Support electronic cam, electronic gear, position latch, synchronous follow, virtual axis and other functions.
- Support 8-channel PWM output.
- Support hardware comparison output, hardware timer, precision output when in motion.
- Support encoder position comparison, HW_PSWITCH2 supports 8 HWOP outputs.
- Support high-speed hardware latch, encoder axis supports 4-channel, handwheel axis only supports 2-channel, other axes support 4-channel.
- A variety of program encryption methods to protect the intellectual property rights of customers.

1.3. System Frame



1.4. Hardware Installment

The ZMC408CE motion controller is installed horizontally with screws, and each controller should be fastened with 4 screws.



→ Unit: mm → Mounting Hole Diameter 4.5mm

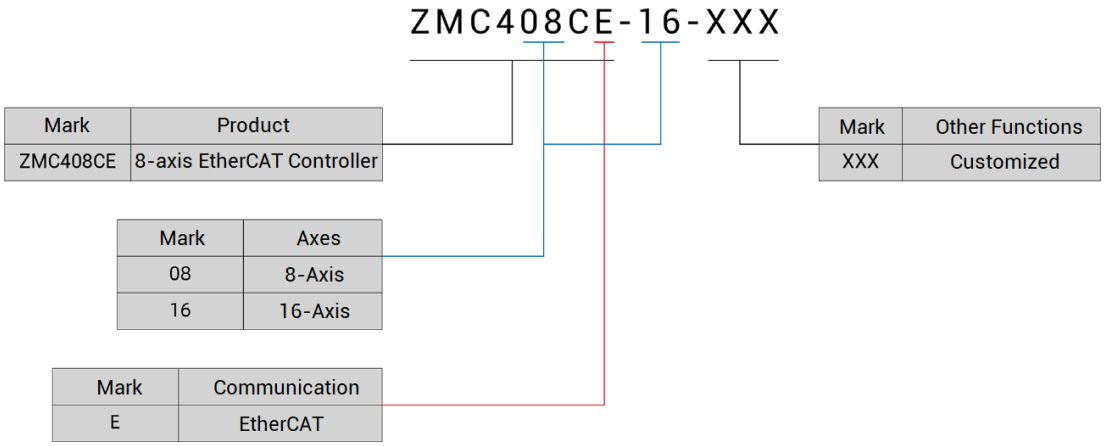
Installation attention	● Non-professionals are strictly prohibited to operate. Specifically, professionals who had been trained related electrical equipment, or who master electrical knowledge. ● Please be sure to read the product instruction manual and safety precautions carefully before installation. ● 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

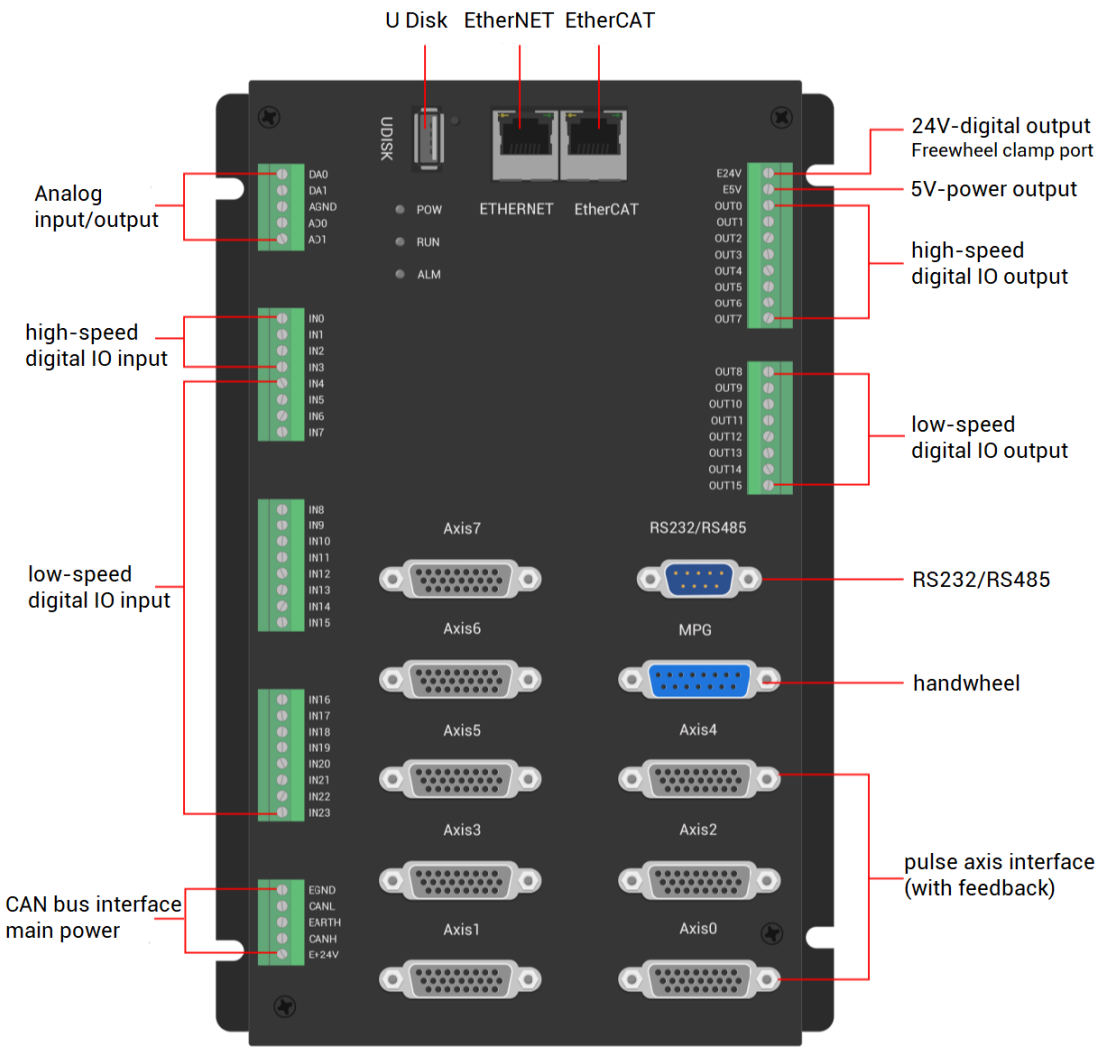
Item	Description
Model	ZMC408CE
Basic Axes	8
Max Extended Axes	32
Type of Basic Axes	Pulse axis / encoder axis / ECAT axis
Digital IO	24 inputs and 16 outputs, and each axis has 2 additional inputs and 2 additional outputs.
Max Extended IO	4096
AD/DA	2 AD inputs, 0-10V. 2 DA outputs, 0-10V.
Max Extended AD/DA	512 ADs and 512 DAs.
Pulse Bit	64
Encoder Bit	64
Speed and Acceleration Bit	64
Highest Pulse Frequency	10MHz
Motion Buffer of Each Axis	4096
Array Space	320000
Program Space	32MByte
Flash Space	256MBye
Power Supply Input	24V DC input
Communication Interfaces	RS232, RS485, ETHERNET, CAN, EtherCAT
Dimension	221*144*37mm

2.2. Nameplates & Models



Model	Description
ZMC408CE	8 axes, point to point, linear, circular, electronic cam, continuous trajectory motion, robot structure.
ZMC408CE-16	16 axes, point to point, linear, circular, electronic cam, continuous trajectory motion, robot structure.

2.3. Interface Definition



→ Interface Description

Mark	Interface	Number	Description
POW	The led that indicates the current state.	1	Power state: it lights when power is conducted.
RUN		1	Run state: it lights when runs normally
ALM		1	Error state: it lights when runs incorrectly
RS232	RS232 serial port	1	Use MODBUS_RTU protocol
RS485	RS485 serial port	1	Use MODBUS_RTU protocol
EtherCAT	EtherCAT bus interface	1	EtherCAT bus interface, connect to EtherCAT bus drive and EtherCAT bus expansion

			module
ETHERNET	Network port	1	Use MODBUS_TCP protocol, expand the number of network ports through the interchanger, and the number of net port channels can be checked through "?*port" command, default IP address is 192.168.0.11
UDISK	U disk interface	1	Insert U disk equipment
E+24V	Main power supply	1	24V DC power, it supplies the power for controller.
CAN	CAN bus interface	1	Connect CAN expansion modules and CAN equipment of other standards.
IN	Digital IO input port	24	NPN type, there are 4 high-speed inputs, and IN0-3 have the latch function.
OUT	Digital IO output port	16	NPN type, there are 4 high-speed outputs, OUT0-3 have PWM, precision output and PSO functions.
AD	Analog input port	2	12-bit resolution, 0-10V.
DA	Analog output port	2	12-bit resolution, 0-10V.
Axis	Pulse axis interface	8	Each interface includes differential pulse output and differential encoder input.
MPG	Handwheel	1	5-24V handwheel signa input

2.4. 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)

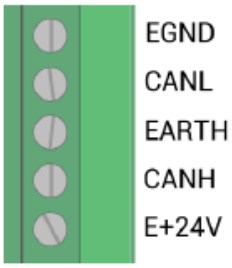
	Direction	3 axial direction
Shock (collide)		15g, 11ms, half sinusoid, 3 axial direction
Degree of Protection		IP20

Chapter III Wiring & Communication

3.1. Power Input, CAN Communication Interface

The power supply input adopts a 5Pin screw-type pluggable wiring terminal, and the interval (means the gap distance between two ports) should be 3.81mm. This 5Pin terminal is the power supply of the controller.

→ Terminal Definition:

Terminal	Name	Type	Function
	EGND	Input	Negative (-) terminal of power input
	CANL	Input/output	CAN communication L
	EARTH	Earthing	Protect
	CANH	Input/output	CAN communication H
	E+24V	Input	Positive (+) terminal of power input

3.1.1. Power Specification

→ Specification

Item	Description
Voltage	DC24V(-10%~10%)
The current to open	≤0.5A
The current to work	≤0.4A
Anti-reverse connection	YES
Overcurrent Protection	YES

3.1.2. CAN Communication Specification & Wiring

The CAN interface of the controller adopts the standard CAN communication

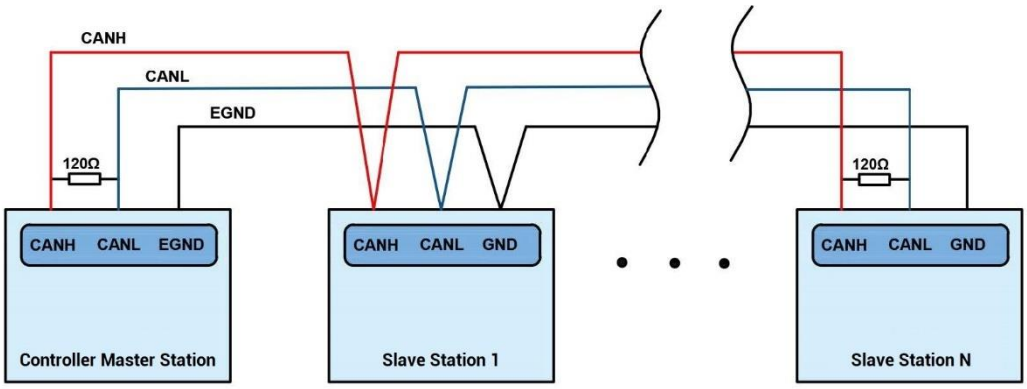
protocol, which mainly includes three ports, CANL, CANH and the public end. And it can connect CAN expansion modules and other standard CAN devices.

→ Specification

Item	CAN
Maximum Communication Rate (bps)	1M
Terminal Resistor	120Ω
Topological Structure	Daisy Chain Topology
The number of nodes can be extended	Up to 16
Communication Distance	The longer communication distance is, the lower communication rate is, and maximum of 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:

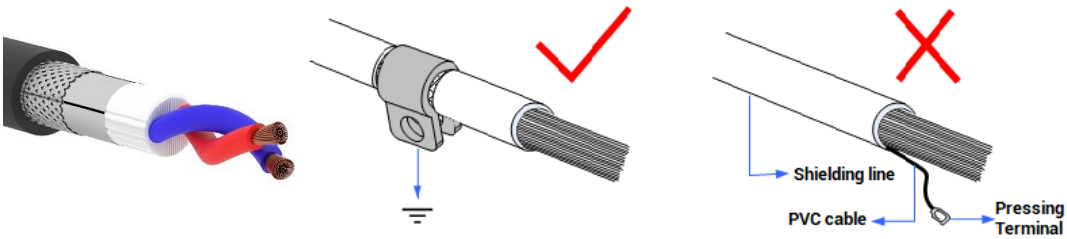
- As above, the daisy chain topology is used for wiring (the star topology structure cannot be used). When the use environment is ideal and there are no many nodes,

the branch structure also can be used.

- Please connect a 120Ω terminal resistor in parallel to each end of the CAN bus for matching the circuit impedance and ensuring communication stability.
- Please be sure to connect the public ends of each node on the CAN bus to prevent the CAN chip from burning out.
- Please use STP (Shielded Twisted Pair), especially in bad environments, and make sure the shielding layer is fully grounded.
- When on-site wiring, pay attention to make the distance between strong current and weak current, it is recommended for the distance 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:

Shielded Twisted Pair, and the shielded cable is grounded.



3.1.3. Basic Usage Method

(1) Please follow the above wiring instructions to wiring correctly.

(2) After powered on, please use EtherNET or RS232 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 "Basic 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 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
```

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

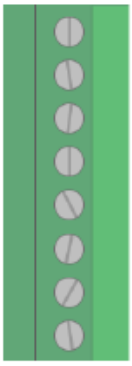
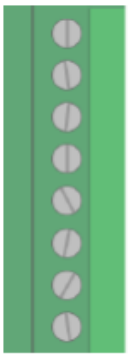
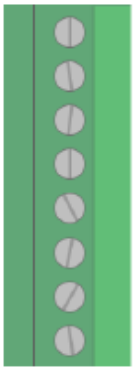
(5) After all the settings are completed, restart the power supply of all stations to establish communication.

(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. IN: Digital Input & High-Speed Latch Port

The digital input adopts 3 groups of 10Pin (there are 3 groups of 10 terminals) screw-type pluggable terminals, and the gap distance between terminals should be 3.81mm. In addition, the high-speed latch function is integrated in digital input signal.

→ Terminal Definition

Terminal	Name	Type	Function 1	Function 2
 IN0	IN0	NPN type, high-speed input	Input 0	High Speed Latch
IN1	IN1		Input 1	
IN2	IN2		Input 2	
IN3	IN3		Input 3	
IN4	IN4	NPN type, low-speed input	Input 4	/
IN5	IN5		Input 5	/
IN6	IN6		Input 6	/
IN7	IN7		Input 7	/
 IN8	IN8		Input 8	/
IN9	IN9		Input 9	/
IN10	IN10		Input 10	/
IN11	IN11		Input 11	/
IN12	IN12		Input 12	/
IN13	IN13		Input 13	/
IN14	IN14		Input 14	/
IN15	IN15		Input 15	/
 IN16	IN16		Input 16	/
IN17	IN17		Input 17	/
IN18	IN18		Input 18	/
IN19	IN19		Input 19	/
IN20	IN20		Input 20	/
IN21	IN21		Input 21	/
IN22	IN22		Input 22	/
IN23	IN23		Input 23	/

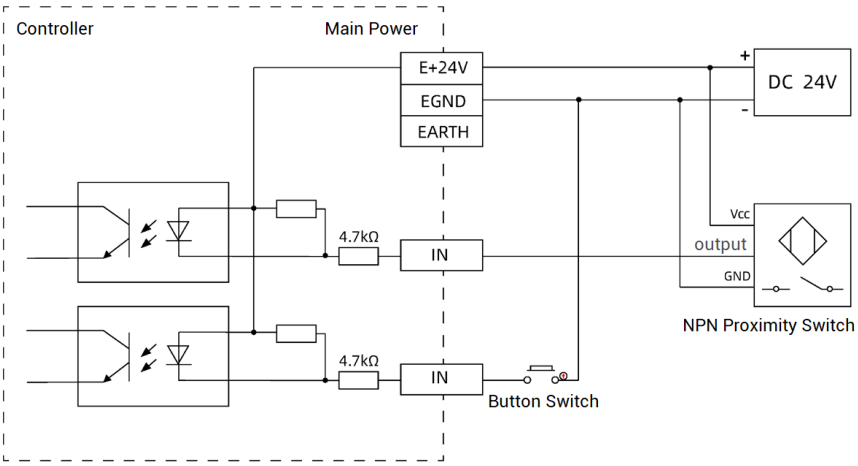
3.2.1. Digital Input Specification & Wiring

→ Specification

Item	High-Speed Input (IN0-3)	Low-Speed Input (IN4-23)
Input mode	NPN type, the input is triggered when by low-electric level	
Frequency	< 100kHz	< 5kHz
Impedance	3.3KΩ	4.7KΩ
Voltage level	DC24V	DC24V
The voltage to open	<15V	<14.5V

The voltage to close	>15.1V	>14.7V
Minimal current	-2.3mA (negative)	-1.8mA (negative)
Max current	-7.5mA (negative)	-6mA (negative)
Isolation mode	optoelectronic isolation	
Note: the above parameters are standard values when the voltage of controller power supply (E+24V port) is 24V.		

→ Wiring Reference



→ Wiring Note:

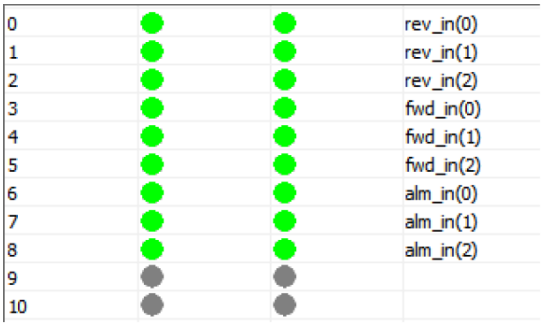
- The wiring principle of high-speed digital input IN (0-3) and low-speed digital input IN (4-23) is shown in the figure above. The external signal source can be an optocoupler, a key switch or a sensor, etc., all can be connected as long as the requirements on output of electric level can be achieved.
- For the public end, please connect the "EGND" port on the power supply to the "COM" terminal of the external input device. If the signal area power supply of the external device and the power supply of the controller are in the same power supply system, this connection also can be omitted.

3.2.2. Basic Usage Method

(1) Please follow the above wiring instructions to wiring correctly.

(2) After powered on, please select ETHERNET or RS232 to connect to RTSys.

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

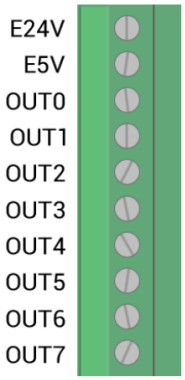


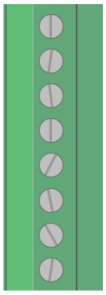
(4) Latch function can be set and triggered through “REGIST” instruction, in software, use REG_INPUTS to configure. Please refer to “Basic” for details.

3.3. OUT: Digital Output & PWM & Hardware Comparison Output

The digital output adopts 2 sets of screw-type pluggable terminals with a spacing of 3.81mm, and the PWM and hardware comparison output functions are integrated in digital output signals.

→ Terminal Definition

Terminal	Name	Type	Function 1	Function 2	Function 3
	E+24V	/	Output freewheeling cramp port	/	/
	E5V	/	5V power output, max is 300mA	/	/
	OUT0	NPN type, high-	Output 0	PWM 0	Hardware Comparison Out 0
	OUT1		Output 1	PWM 1	Hardware Comparison Out 1
	OUT2		Output 2	PWM 2	Hardware Comparison Out 2

	OUT3	speed output	Output 3	PWM 3	Hardware Comparison Out 3
	OUT4	NPN type, low-speed output	Output 4	PWM 4	Hardware Comparison Out 4
	OUT5		Output 5	PWM 5	Hardware Comparison Out 5
	OUT6		Output 6	PWM 6	Hardware Comparison Out 6
	OUT7		Output 7	PWM 7	Hardware Comparison Out 7
OUT8 OUT9 OUT10 OUT11 OUT12 OUT13 OUT14 OUT15 	OUT8	NPN type, low-speed output	Output 8	/	/
	OUT9		Output 9	/	/
	OUT10		Output 10	/	/
	OUT11		Output 11	/	/
	OUT12		Output 12		
	OUT13		Output 13		
	OUT14		Output 14		
	OUT15		Output 15		

Note:

- ✧ The E5V power output port is used for PWM or common anode wiring of single-ended axis. It is not recommended for other purposes due to lower power.
- ✧ E24V is freewheeling cramp port, when there is the inductive load on output port, connect E24V to load power supply for protecting output circuit.
- ✧ OUT0-7 support PWM and hardware comparison output functions. Please note they don't support HW_PSWITCH command, recommend use HW_PSWITCH2 command.

3.3.1. Digital Output Specification & Wiring

→ Specification

Item	High Speed Output (OUT0-3)	Low Speed Output (OUT4-15)
Output mode	Transistor NPN type, OD output	
Frequency	< 400kHz	< 8kHz
Voltage level	DC24V	DC24V
Max output current	+300mA/point	+300mA/point
Max leakage current when off	25μA	25μA
Respond time to	1μs (resistive load typical	12μs

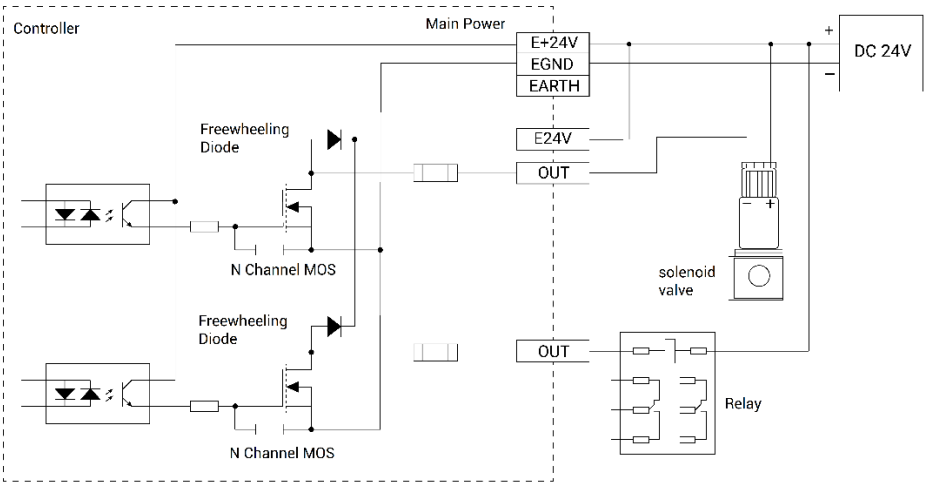
conduct	value)	
Respond time to close	3μs	80μs
Overcurrent protection	Support	Support
Isolation method	optoelectronic isolation	

Note:

✧ The times in the form are typical 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 obviously affected by the external load circuit, and the output frequency should not be set too high in the application. For high-speed output, it is recommended to set below 400KHz, for low-speed output, it is recommended to set below 8KHz. For more higher speed, please contact us to adjust parameters or custom hardware.

→ Wiring Reference



→ Wiring Note:

- The wiring principle of high-speed digital output OUT (0-3) and low-speed digital output OUT (4-15) 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 connect the "EGND" port on the power supply 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.

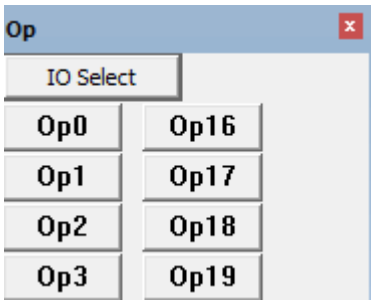
- The E5V port is a 5V power output port, which can be used when some loads need to provide an external 5V power input, the maximum current is 300mA.

3.3.2. Basic Usage Method

(1) Please follow the above wiring instructions to wiring correctly.

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

(3) Open or close output port directly through "OP" command, also, it can be opened or closed through "RTSys/Tool/Op". Please refer to "Basic" for details.



(4) The PWM function, set the frequency and duty cycle through "PWM_FREQ" and "PWM_DUTY". Please refer to ZBasic for details.

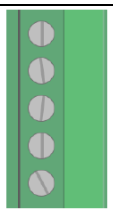
(5) Hardware comparison output can be set and opened through "HW_PSWITCH2". Please refer to ZBasic for details.

3.4. AD/DA Analog Input/Output

The analog port adopts a set of 5Pin screw-type pluggable terminals with a spacing of 3.81mm.

→ Terminal Definition

Terminal	Name	Type	Function
	DA0	Output	Analog output terminal: AOUT(0)
	DA1		Analog output terminal: AOUT(1)
	AGND	Public End	Unique public end of this analog

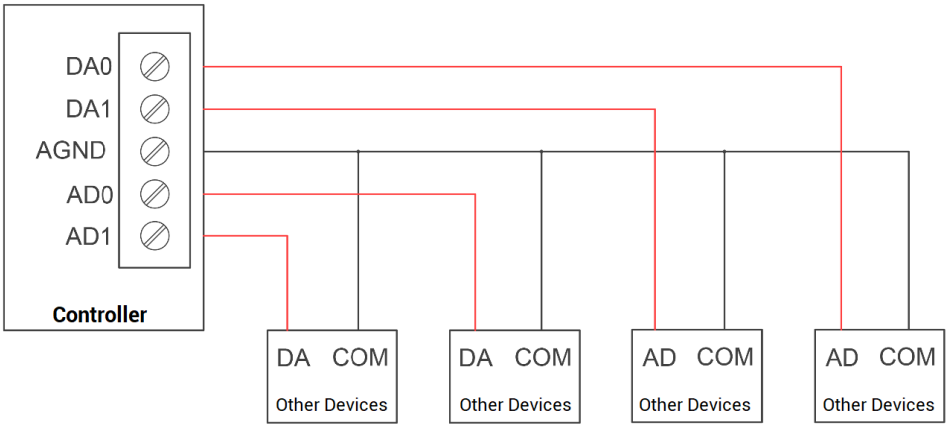
	AD0	Input	Analog input terminal: AIN(0)
	AD1		Analog input terminal: AIN(1)

3.4.1. AD/DA Analog Input/Output Specification & Wiring

→ Specification

Item	AD（0-1）	DA（0-1）
Resolution	12-bit	12-bit
Data range	0-4095	0-4095
Signal range	0-10V input	0-10V output
Data refresh ratio	1KHz	1KHz
Voltage input impedance / output load	>40KΩ (voltage input impedance)	>1KΩ (voltage output load)

→ Wiring Reference



→ Wiring Note:

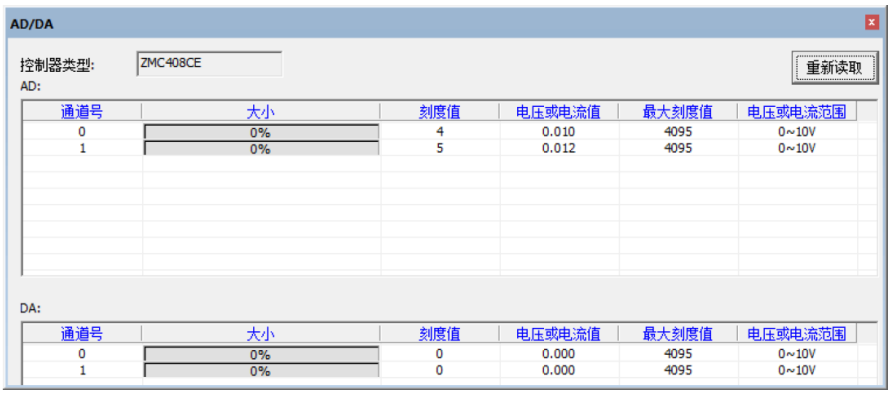
- The analog input/output wiring method is as shown in the figure above, and the external load signal range must match with this signal range.
- Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.

3.4.2. Basic Usage Method

(1) Please follow the above wiring instructions to wiring correctly.

(2) After powered on, please use EtherNET or RS232 o 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 “Basic” for details.



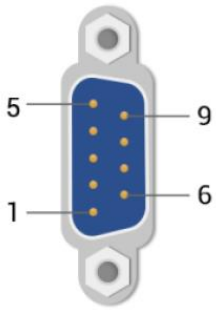
3.5. RS232/RS485 Serial Port

RS232 and RS485 are integrated in one standard DB9 male socket and support MODBUS_RTU protocol and custom communication.

→ Interface Definition:

Terminal	PIN	Name	Type	Function
	1, 6, 8	NC	Spare	Reserved
	2	232RXD	Input	RS232 (port 0) signal, receive data
	3	232TXD	Output	RS232 (port 0) signal, send

				data
	4	485A/+	Input/Output	RS485 (port1) signal A/+
	5	EGND	Output	Negative pole output of 5V power, and output for the public end
	7	485B/-	Input/Output	RS485 (port1) signal B/-
	9	E5V	Output	Positive pole output of 5V power, maximum is 300mA

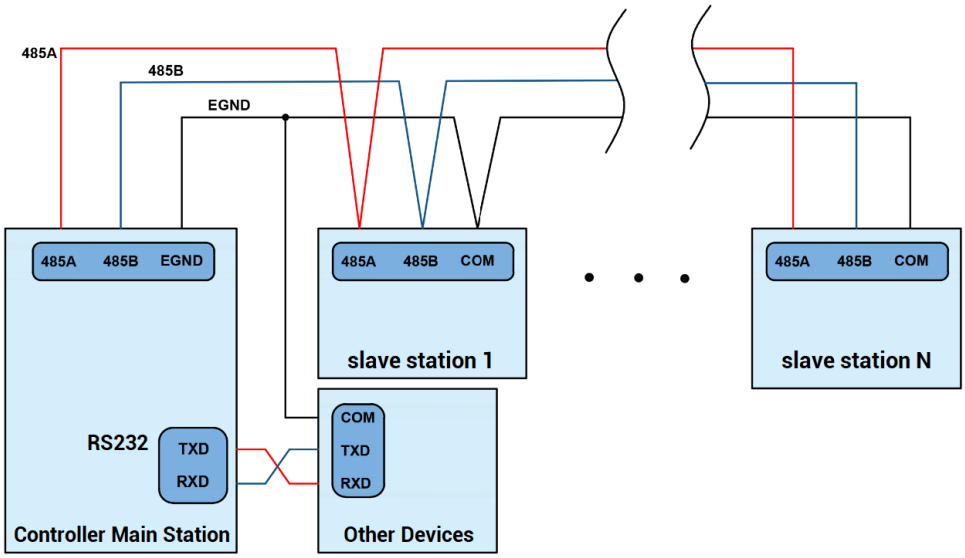


3.5.1. RS232/RS485 Specification & Wiring

→ Specification:

Item	RS232 (port0)	RS485 (port1)
Maximum Communication Rate	115200bps	115200bps
Terminal Resistor	No	No
Topology Structure	Connect correspondingly (1 to 1)	Daisy chain structure
The number of nodes can be extended	1	127
Communication Distance	The Longer communication distance is, the lower communication rate is, maximum 5m is recommended.	The Longer communication distance is, the lower communication rate is, maximum 30m is recommended.

→ Wiring Reference:



→ Wiring Notes:

- The wiring of RS232 (port0) is as above, it needs to cross-wiring for sending and receiving signals, and it is recommended to use a double-female head cross line when connecting to a computer.
- The wiring of RS485 (port1) is above, it is the daisy chain topology is used for wiring (the star topology structure cannot be used). When the use environment is ideal and there are no many nodes, the branch structure also can be used.
- Please be sure to connect the public ends of each communication node to prevent the communication chip from burning out.
- Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.

3.5.2. Basic Usage Method

(1) Please follow the above wiring instructions to wiring correctly.

(2) After powered on, please use any one interface among the three interfaces ETHERNET, RS232 (there is default parameter, which can be connected directly) and RS485 (there

is default parameter, which can be connected directly, but for hardware, adapter head is needed) to connect to RTSys.

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

(4) According to their respectively instructions, correctly set the relevant parameters of the third-party equipment to match the parameters of each node.

(5) When all is configured, it can start to do communicating.

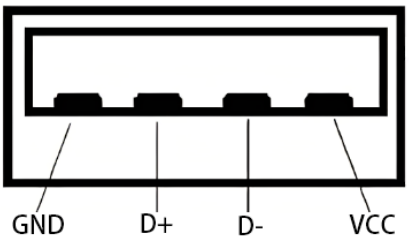
(6) Communication data of RS232 / RS485 can be directly viewed through “RTSys / Controller / State the Controller / CommunicationInfo".

```
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 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
```

3.6. U Disk

The ZMC408CE 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:

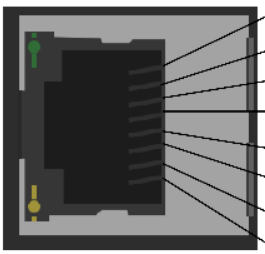


→ Specification

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

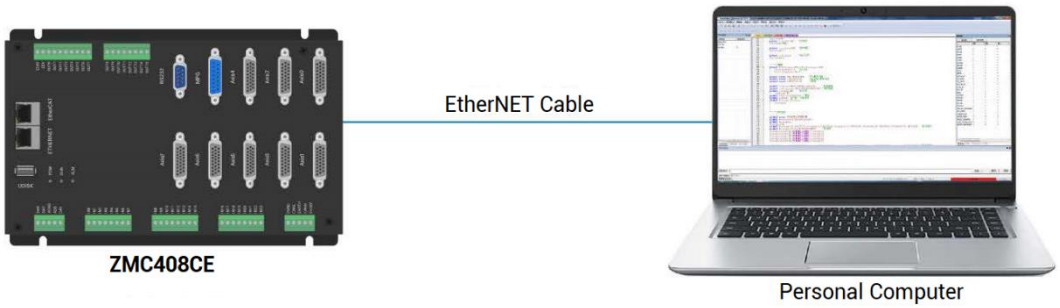
3.7. ETHERNET

ZMC408CE motion controller has one ethernet port, and it supports MODBUS_TCP protocol and custom communication, the default IP address is 192.168.0.11. The pin definition is as follows:

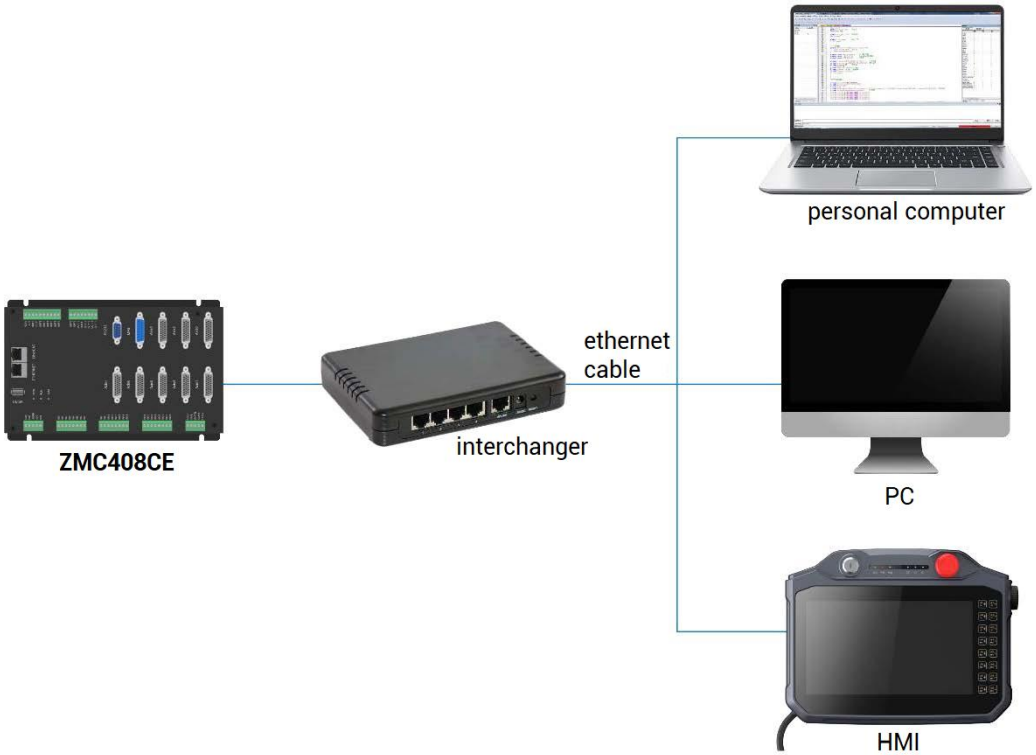


PIN	Signal	Description
1	TX+	Send signal (+)
2	TX-	Send signal (-)
3	RX+	Receive signal (+)
4	NC	Reserved
5	NC	Reserved
6	RX-	Receive signal (-)
7	NC	Reserved
8	NC	Reserved

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

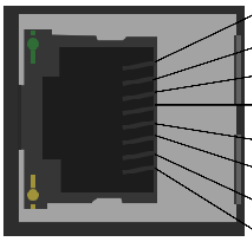


The controller can also be connected to the interchanger through an Ethernet cable, and then use interchanger to connect to other devices, then multi-point connection can be achieved. The schematic diagram is as follows:



3.8. EtherCAT Bus Interface

ZMC408CE motion controller has a 100M EtherCAT communication interface, and it supports EtherCAT protocol. In addition, EtherCAT driver or EtherCAT expansion module can be connected. The pin definition 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

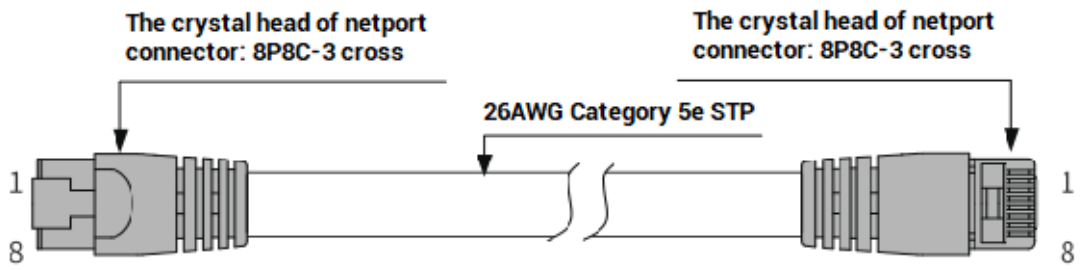
Item	Specification
Communication protocol	EtherCAT protocol
Valid service	CoE(PDO, SDO), FoE
Synchronization method	IO adopts input and output synchronization / DC-distributed clock
Physical level	100BASE-TX
Duplex mode	Full duplex
Topology	linear topology
Transfer media	Cable
Transfer distance	It is less than 100M between 2 nodes
Process data	Maximum 1486 bytes of one single frame
Synchronization shaking of two slave stations	<1us
Refresh	1000 digital input and output about is 30us, 16 servo axes are about 100us

→ Communication Cable Requirements

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

The network cable adopts Category 5e STP, and the crystal head has a metal shell to

reduce interference and to prevent information from being eavesdropped. As shown below:



Item	Specification
Cable type	Flexible crossover cable, Category 5e
traverse	twisted pair
Line pairs	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 that is 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.

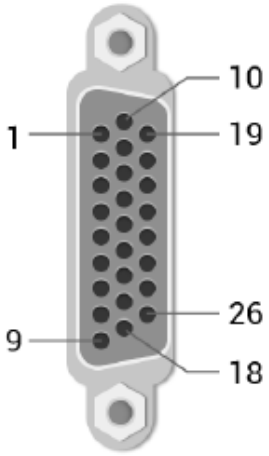
Please use tube-type pre-insulated terminals and cables with appropriate wire diameters to connect the user terminals.

3.9. AXIS Differential Pulse Axis Interface

This product provides 8 local differential pulse axis interfaces, each interface is a standard DB26 female socket. Each terminal provides 0V and +5V output, which can provide 5V power for the encoder.

Before the axis is used, use ATYPE instruction to configure the axis type.

→ Interface Definition

Interface	Pin	Signal	Description
	1	EGND	Negative pole of IO 24V power
	2	IN24-31/ALM	General input (recommended as driver alarm)
	3	OUT16.18.20.22. 24.26.28.30/ENABLE	General output (recommended as driver enable)
	4	EA-	Encoder differential input signal A-
	5	EB-	Encoder differential input signal B-
	6	EZ-	Encoder differential input signal Z-
	7	+5V	Positive pole of 5V power of pulse/encoder signal
	8	Reserved	Reserved
	9	DIR+	Servo or step directional output (differential signal) +
	10	GND	Negative pole of 5V power of pulse/encoder signal
	11	PUL-	Servo or step pulse output (differential signal) -
	12	Reserved	Reserved
	13	GND	Negative pole of 5V power of pulse/encoder signal
	14	OVCC	Positive pole of IO 24V power
	15	OUT17.19.21.23.25 27.29.31/CLR	Digital output, recommended to do alarm clear
	16	IN32-39/INP	Digital input, recommended to do position arrival signal
	17	EA+	Encoder differential input signal A+
	18	EB+	Encoder differential input signal B+
	19	EZ+	Encoder differential input signal Z+
	20	GND	Negative pole of 5V power of pulse/encoder signal
	21	GND	
	22	DIR-	Servo or step directional output (differential signal) -

	23	PUL+	Servo or step pulse output (differential signal) +
	24	GND	Negative pole of 5V power of pulse/encoder signal
	25	Reserved	Reserved
	26	Reserved	Reserved

Note:

- ALM, ENABLE, CLR and INP are recommended to be used as axis IO, because the drive capacity is small.
- OVCC, +5V are only used for communication between the controller and the servo driver, please do not use it as power supply for other places.

Pulse Axis PIN No. & IO:

Pulse Axis No.	IN (PIN 2)	OUT (PIN 3)	OUT (PIN15)	IN (PIN16)
AXIS0	IN24	OUT16	OUT17	IN32
AXIS1	IN25	OUT18	OUT19	IN33
AXIS2	IN26	OUT20	OUT21	IN34
AXIS3	IN27	OUT22	OUT23	IN35
AXIS4	IN28	OUT24	OUT25	IN36
AXIS5	IN29	OUT26	OUT27	IN37
AXIS6	IN30	OUT28	OUT29	IN38
AXIS7	IN31	OUT30	OUT31	IN39

3.9.1. AXIS Interface Signal Specification & Wiring

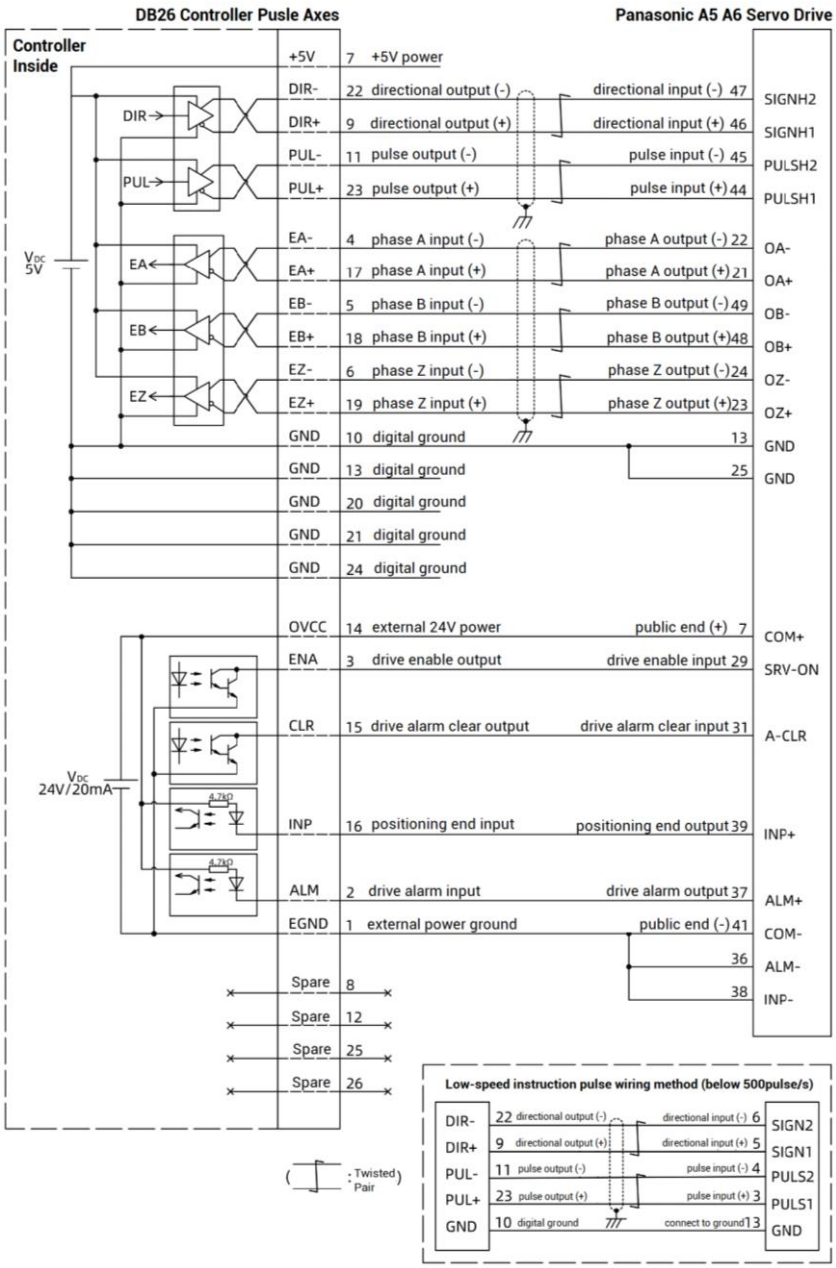
→ Specification:

Signal	Item	Description
PUL/DIR	Signal type	Differential output signal
	Voltage range	0-5V
	Maximum frequency	10MHz
EA/EB/EZ	Signal type	Differential input signal
	Voltage range	0-5V
	Maximum frequency	5MHz

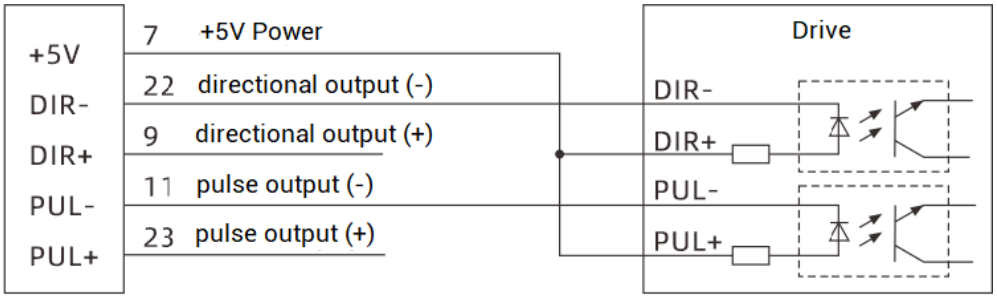
IN24-39	Input method	NPN type, it is triggered when low electric level is input.
	Frequency	< 5kHz
	Impedance	6.8KΩ
	Voltage level	DC24V
	The voltage to open	<10.5V
	The voltage to close	>10.7V
	Minimal current	-1.8mA (negative)
	Maximum current	-4mA (negative)
	Isolation	optoelectronic isolation
OUT16-31	Output method	NPN type, it is 0V when outputs
	Frequency	< 8kHz
	Voltage level	DC24V
	Maximum current	+50mA
	Overcurrent protection	No
	Isolation	optoelectronic isolation
+5V, GND	Maximum output current for 5V	50mA
OVCC, EGND	Maximum output current for 24V	50mA

→ Wiring Reference:

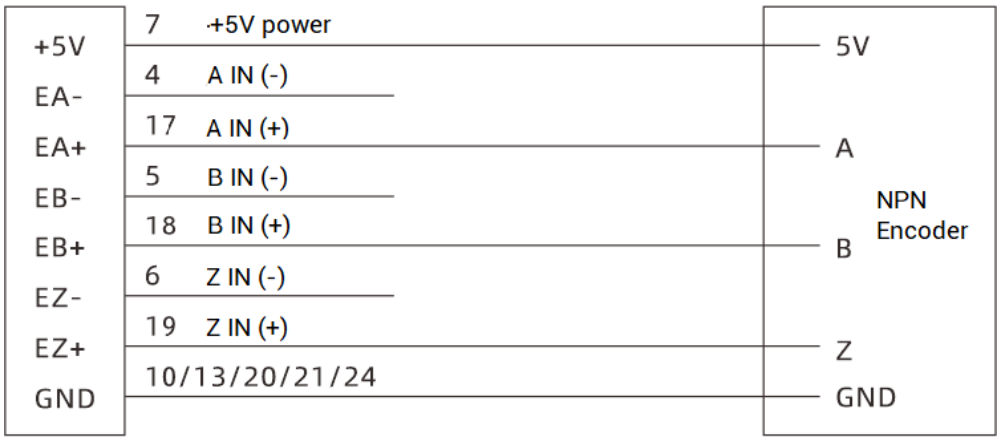
Reference example of wiring with Panasonic A5/A6 servo driver:



Single-Ended Pulse Axis Wiring:



Single-Ended Encoder 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.
- ✧ When speed is met, use low-speed differential pulse preferentially. When high-speed differential pulse interface is used, please make sure controller internal digital ground is connected to driver high-speed pulse reference ground.
- ✧ Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.
- ✧ Some servo drives are not isolated by optocoupler. At this time, the GND must be connected to the GND of the driver. The encoders of most drives are not isolated by optocoupler. When connecting the encoder, GND must be connected.

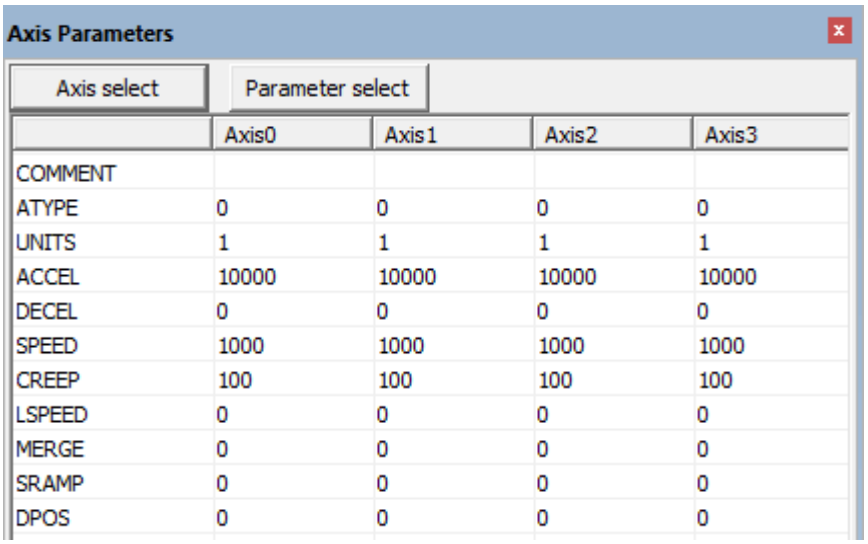
3.9.2.Basic Usage Method

(1) Please follow the above wiring instructions to wiring correctly.

(2) After powered on, please use ETHERNET or RS232 (default parameter, it can be connected directly) to connect to RTSys..

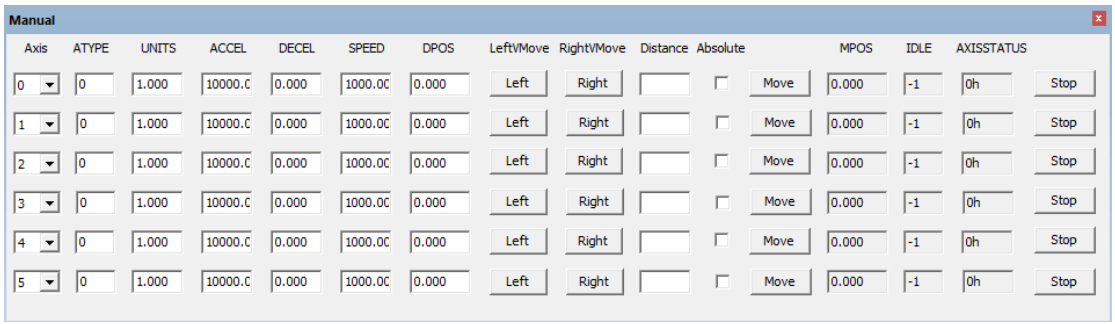
(3) Set axis parameters, such as, ATYPE, UNITS, SPEED, ACCEL, FWD_IN, REV_IN, etc.

(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 “Basic”, or see “RTSys/View/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 “View – Manual”.



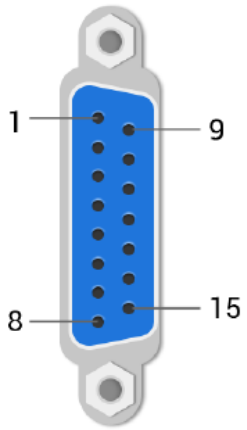
Refer to BASIC Routine:

```
BASE(0,1)              'select axis 0 and axis 1
ATYPE = 1,1            'set axis 0 and axis 1 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
```

3.10. MPG Handwheel Interface

This product provides a special interface for the local handwheel encoder axis, which is a double-row standard DB15 female seat.

→ Interface Definition

Interface	PIN	Signal	Description
	1	H-5V	Positive pole of 5V power supply for output, which supplied power for handwheel
	2	HA-	Encoder phase A signal (IN40)
	3	HB-	Encoder phase B signal (IN41)
	4	HEMGN	Emergency stop signal (IN51)
	5	NC	Reserved
	6	HX1	Select X1 ratio (IN42)
	7	HX10	Select X10 ratio (IN43)
	8	HX100	Select X100 ratio (IN44)
	9	HSU	Select axis 3 (IN48)
	10	HSV	Select axis 4 (IN49)
	11	EGND	Negative pole of 5V power supply for output, signal public end
	12	HSW	Select axis 5 (IN50)
	13	HSZ	Select axis 2 (IN47)
	14	HSY	Select axis 1 (IN46)
	15	HSX	Select axis 0 (IN45)

Note:

- 5V power supply only supplies for handwheel, don't supply power for others.
- All signals of this interface are digital input signals, number is IN (40-51).

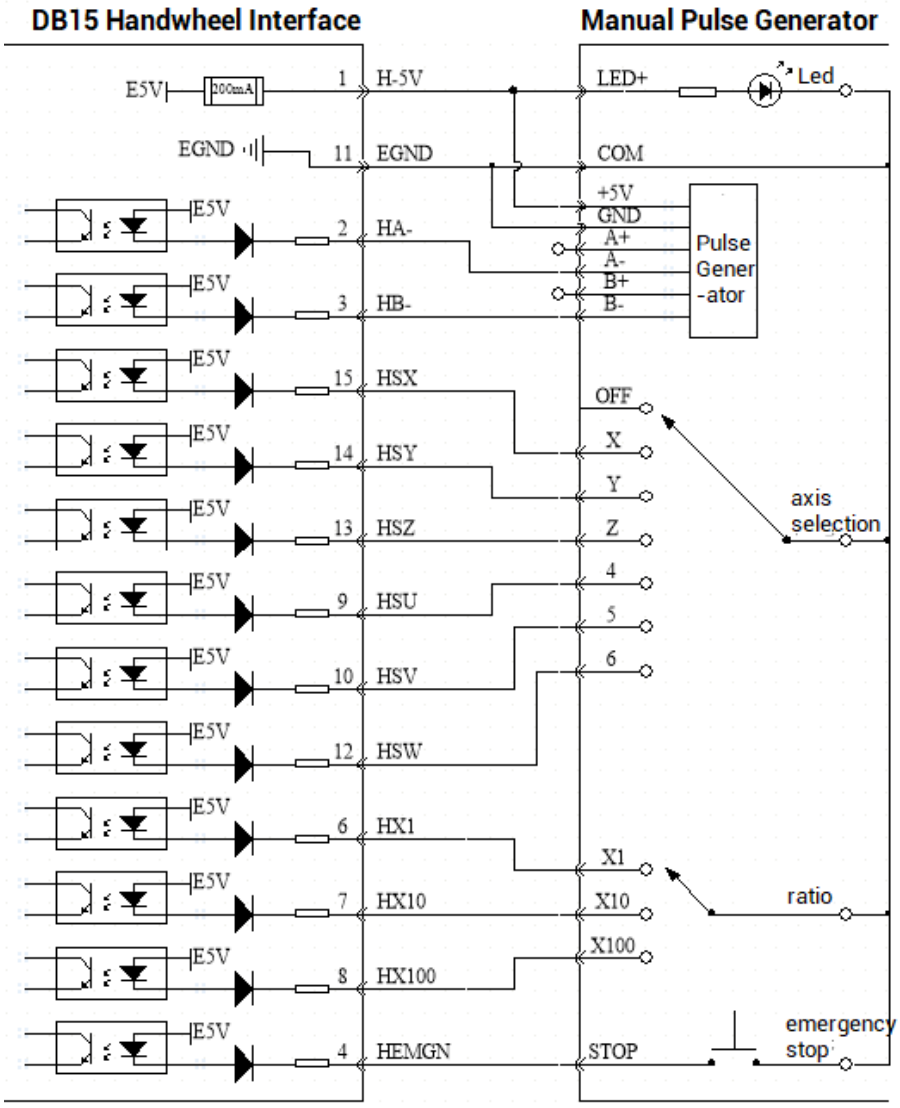
3.10.1. MPG Specification & Wiring

→ Specification

Item	IN (40-51)

Mode (input)	NPN, it is triggered when low electric inputs.
Frequency (input)	<5kHz is recommended
Impedance (input)	510Ω
Max voltage (input)	24V
The current to open (input)	<2.8
The current to close (input)	>2.9A
Min current (input)	-1.8mA
Max current (input)	-5.5mA
Isolation	optoelectronic isolation
5V power supply (H-5V, EGND) max output current	100mA

→ Wiring Reference



→ Wiring Note

- The wiring principle of handwheel encoder axis interface is shown above, please connect carefully due to diversified handwheel designs.
- Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.

3.10.2. Basic Usage Method

1. Refer to above handwheel wiring graphic, correctly connect the handwheel and controller.

2. After powered on, please select ETHERNET or RS232 or RS485 to connect to ZDevelop.

3. Configure axis No., if there is no default AXIS axis No. (axis 10, 11, 12, 13 are recommended) of controller handwheel interface, remapped must be done. Followings are processes.

```
BASE(target axis number)          'the axis No. to be remapped
ATYPE(target axis number) = 0     'set axis type as 0
BASE(8)                           'handwheel interface initial axis No. is 8 (invalid)
ATYPE(8) = 0                      'set initial type of handwheel interface as 0
AXIS_ADDRESS(target axis number)=(-1<<16) + 8
                                  'bind initial axis 8 to target axis No.
ATYPE(target axis number) = 6
                                  'set this new axis interface as required axis type, such as 3 or 6
```

4. Configure IO: assign axis selection (HSX, HSY, HSZ, HSU) and ratio (HX1, HX10, HX100) and emergency stop (HEMGN) functions as required. These signals are essentially digital input signals with fixed numbers but no fixed functions. It needs ZDevelop development (the axis selection is the connected axis of "connect" synchronization motion, and the ratio is the "connect" ratio.

5. When completed above steps, it can start to use handwheel.

BASIC Routine Reference:

```
ATYPE(4) = 0                      'restore axis type of axis 10
ATYPE(8) = 0                      'restore default handwheel axis type
AXIS_ADDRESS(4) = (-1<<16)+ 8     'map the address of MPG manual pulse axis to axis 4
ATYPE(4) = 3                      'set manual pulse axis as quadrature encoder type
UNITS(4) = 1                      'set the unit as pulse for pulse amount of manual pulse axis
CONNECT(100,4) AXIS(0)
'axis 0 connects to manual pulse axis at the synchronous ration of 100
```

Chapter IV Expansion Module

The controller can expand digital IO, analog IO, pulse axis and other resources through CAN bus (ZIO series expansion modules). For details, please refer to "ZIO Expansion Card Hardware Manual". Also, through EtherCAT bus (EIO series expansion cards) expansion of these resources also can be achieved, please refer to each EIO hardware manual for details.

4.1. CAN Bus Expansion

ZIO series expansion module or ZMIO310-CAN coupler expansion module can be used.

Controller + ZIO expansion module, when the eighth bit of the DIP switch of the expansion module is turned to ON, which indicates that a 120 ohm resistor has been connected, but needs to connect one 120 ohm resistor externally. When connecting multiple CAN expansion modules, you only need to dial ON for the eighth digit of the last expansion module, which means please do not dial bit-8 of other modules.

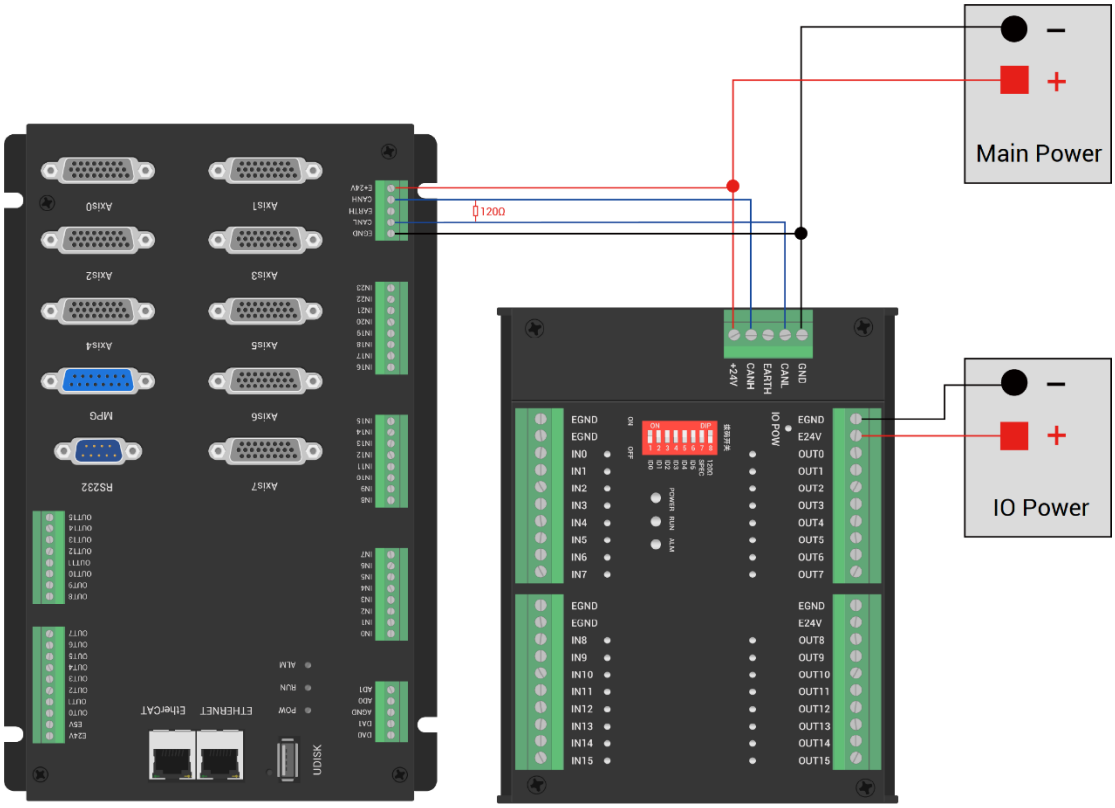
4.1.1. CAN Bus Expansion Wiring

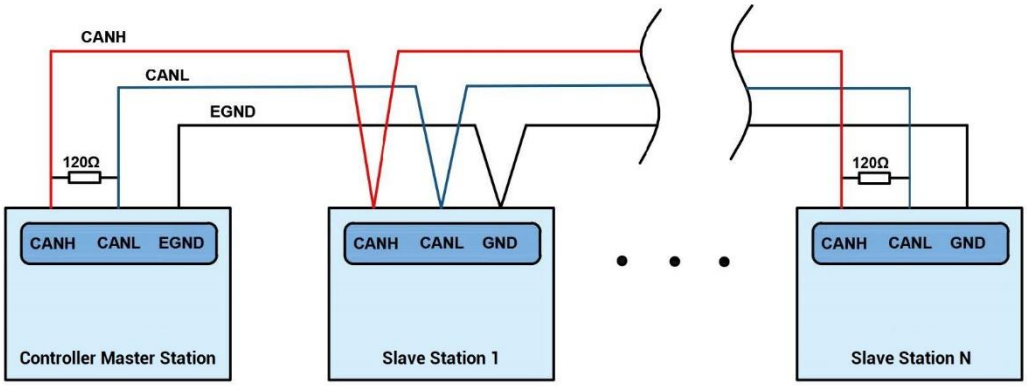
The ZIO expansion module is powered by the dual power supply. Except the main power supply, an additional IO power supply is required to supply independent power for IO. Both the main power supply and the IO power supply use 24V DC power supply. For ZAIO, it only needs to connect to the main power supply.

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:

- ZMC408CE controller uses the single power, and ZIO 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 ZIO expansion modules on the CAN bus, a 120-ohm resistor needs to be connected in parallel between the CANL and CANH terminals, for the ZIO 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 (ZMC408CE motion controller expansion board's DIP ID starts from 3 at least, so starting IO No. starts from 64 at least):

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 ZDevelop 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)

ALMRM 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.

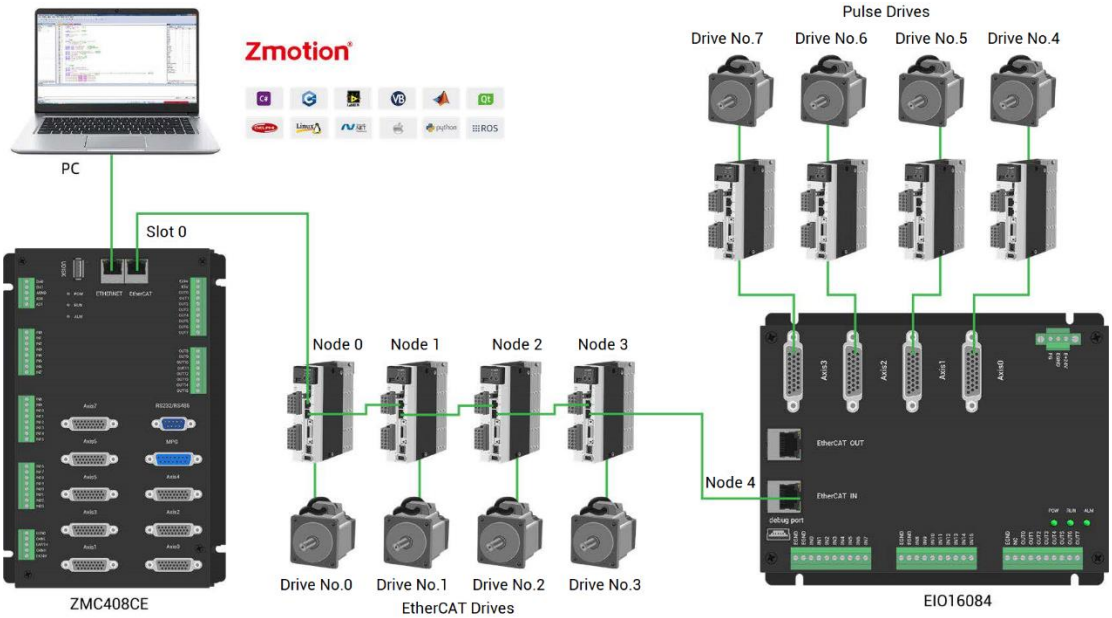
4.2.1. EtherCAT Bus Expansion Wiring

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 number (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 number (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 number:

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 number, 0-default

node: device number, starting from 0

iobase : mapping the IO start number, the setting result will only be a multiple of 8

Example:

NODE_IO(0,0)=32 'set the IO start number 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 number)=(slot number<<16)+driver number+1

Example:

```
AXIS_ADDRESS(0)=(0<<16)+0+1
                'the first drive on the EtherCAT bus, drive number 0, bound as axis 0
AXIS_ADDRESS(1)=(0<<16)+1+1
                'the second drive on the EtherCAT bus, drive number 1, 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

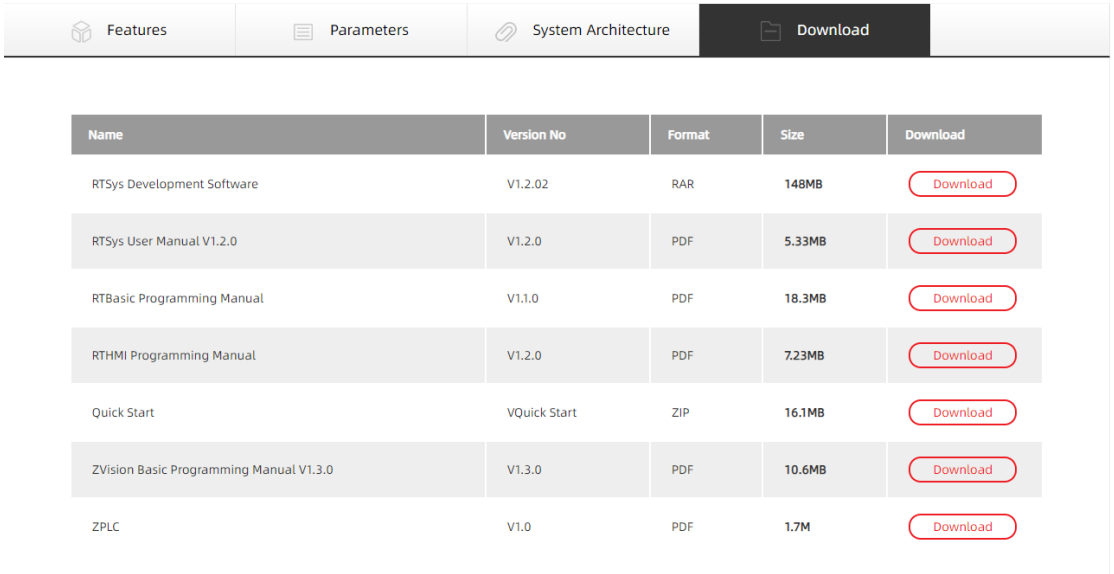
5.1. Program in RTSys Software

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.

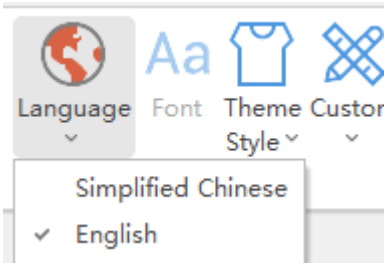
In RTSys, there are 4 programming languages for motion control development, Basic, PLC, HMI and C language, they can run multi-tasks among them, especially for Basic, multi-task running can be achieved separately, hybrid programming is also OK with PLC, HMI and C language.

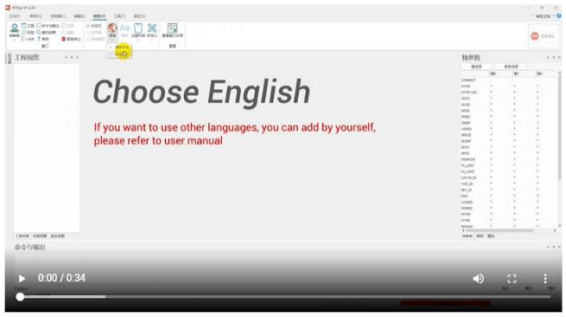
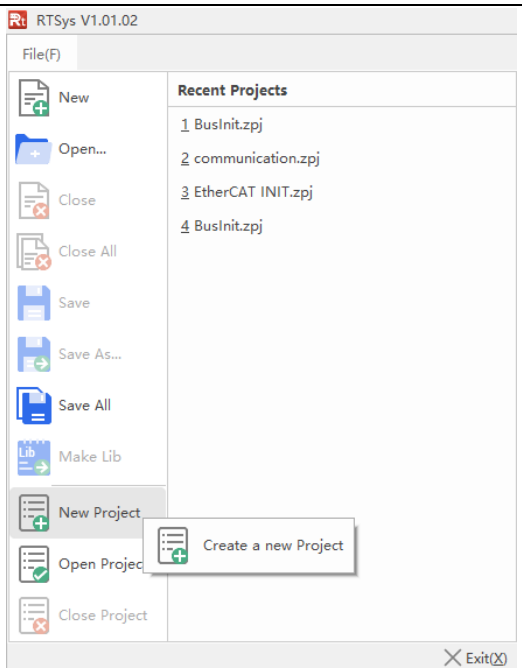
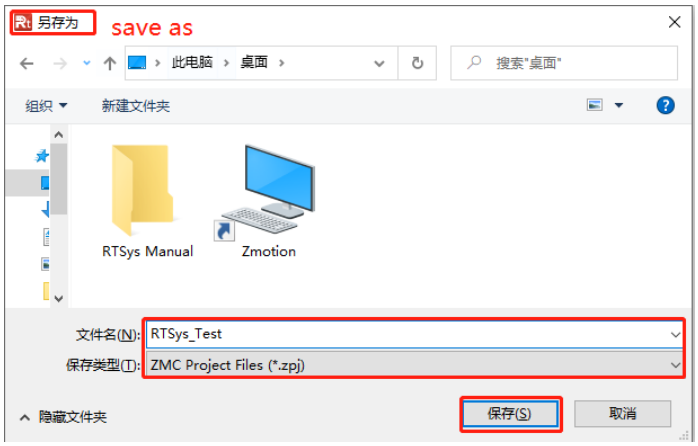
RTSys Downloading Address: https://www.zmotionglobal.com/pro_info_282.html

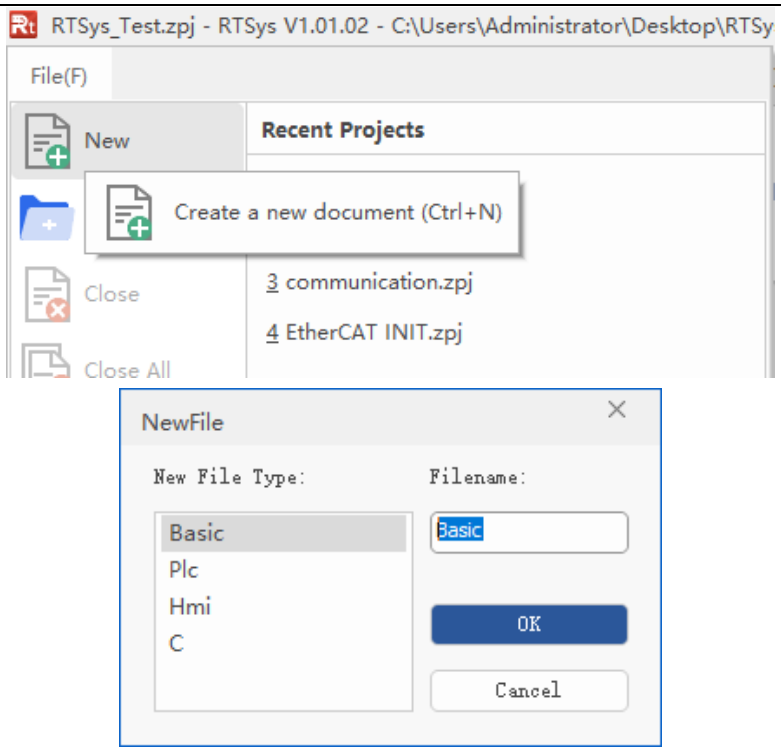
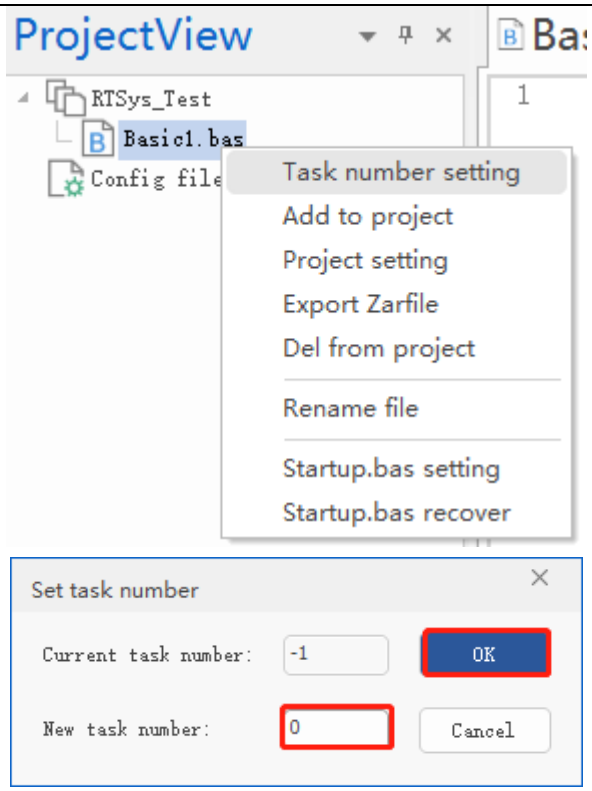
And related manuals can be found in "Download":

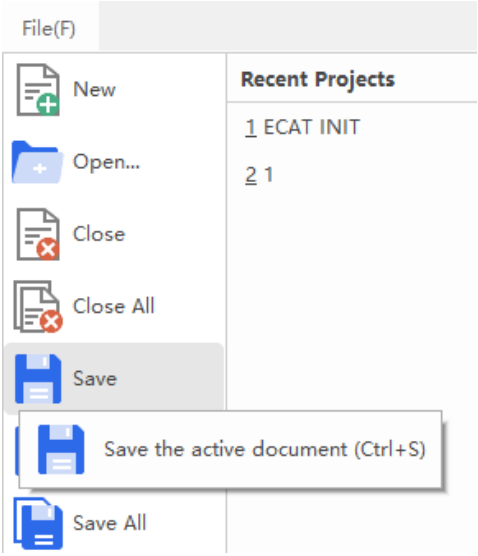
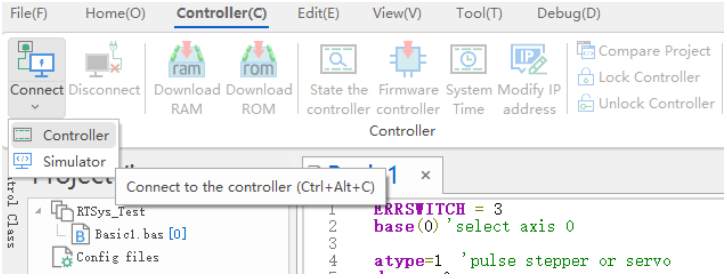
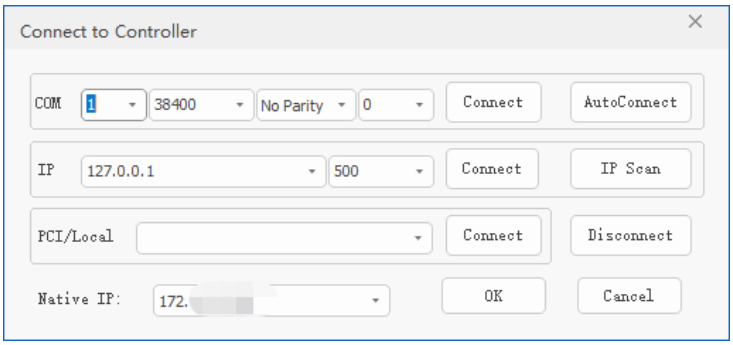


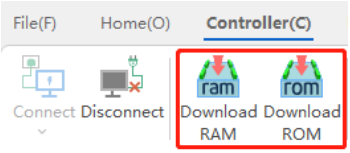
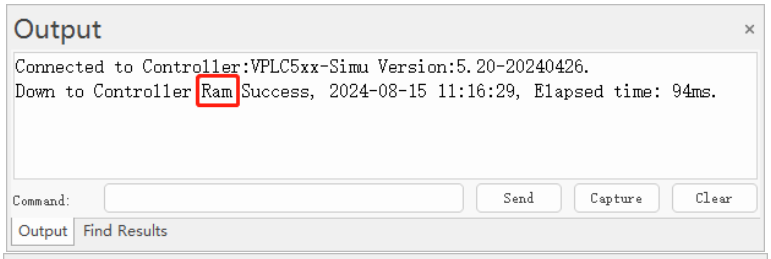
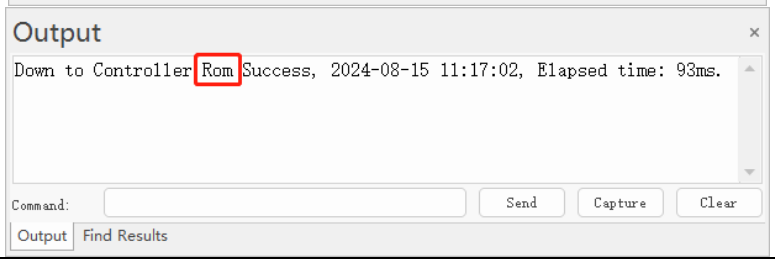
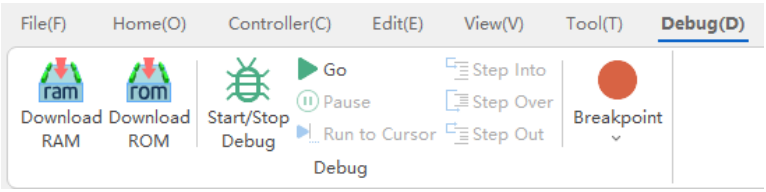
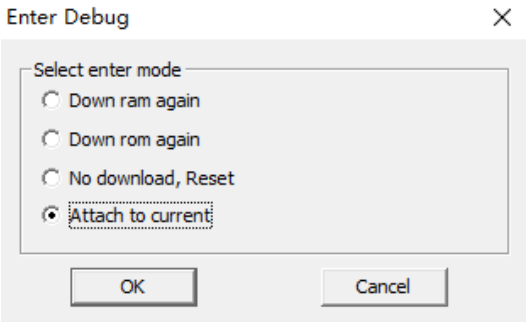
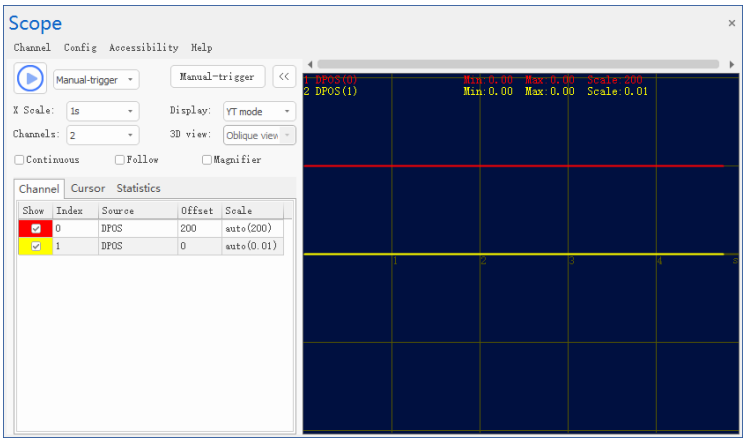
Name	Version No	Format	Size	Download
RTSys Development Software	V1.2.02	RAR	148MB	Download
RTSys User Manual V1.2.0	V1.2.0	PDF	5.33MB	Download
RTBasic Programming Manual	V1.1.0	PDF	18.3MB	Download
RTHMI Programming Manual	V1.2.0	PDF	7.23MB	Download
Quick Start	VQuick Start	ZIP	16.1MB	Download
ZVision 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,	 Language Switch Video Showing:

	click OK, and restart it.	**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

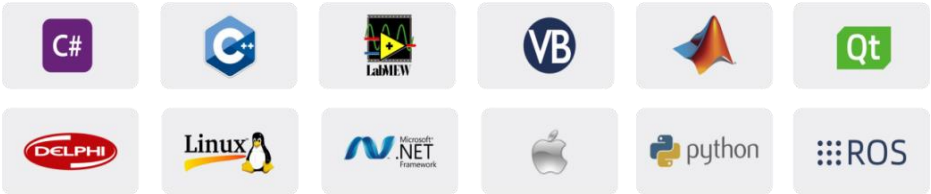
system enters ZBIOS state, please click “update” again.

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.

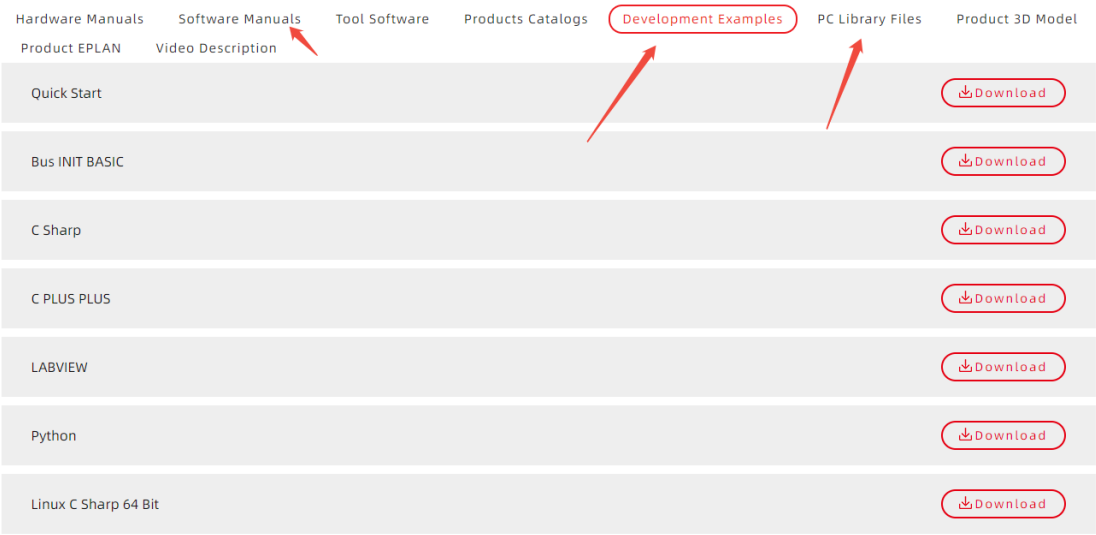
5.3. Program in Host-Computer by PC Languages

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".

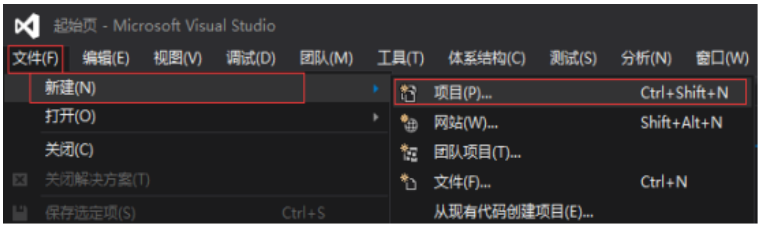
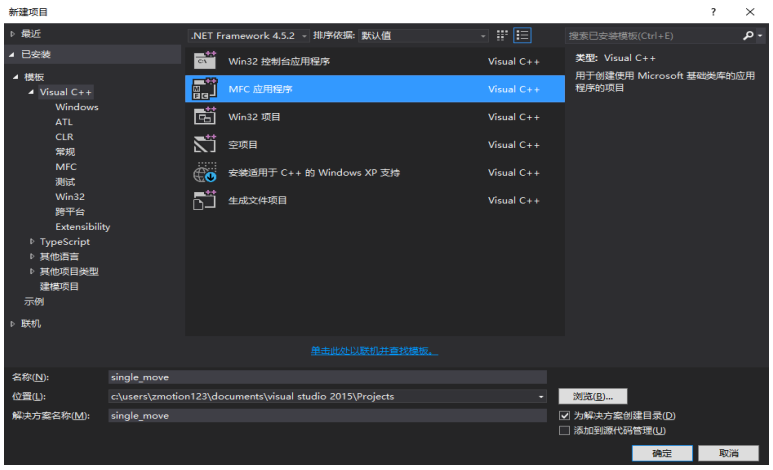
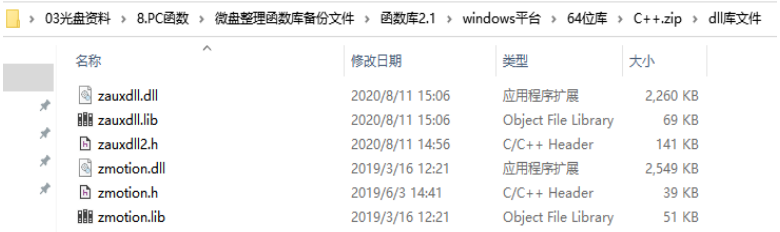


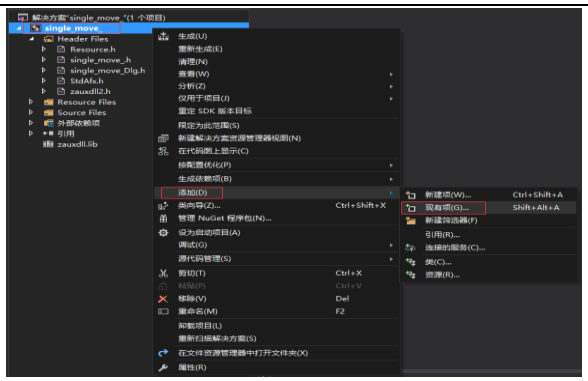
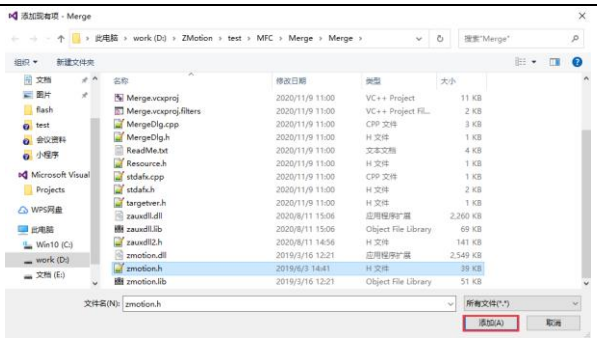
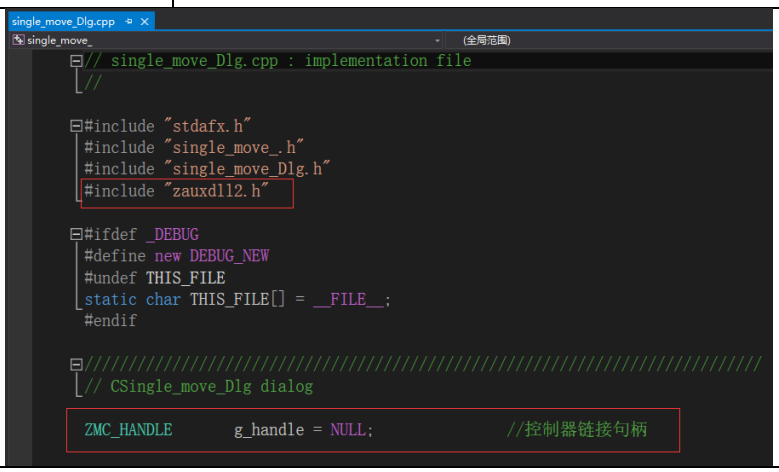
The program developed using the PC software cannot be downloaded to the controller, and it is connected to the controller through the dll dynamic library. The dll library needs to be added to the header file and declared during development.

- 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 ZDevelop 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.