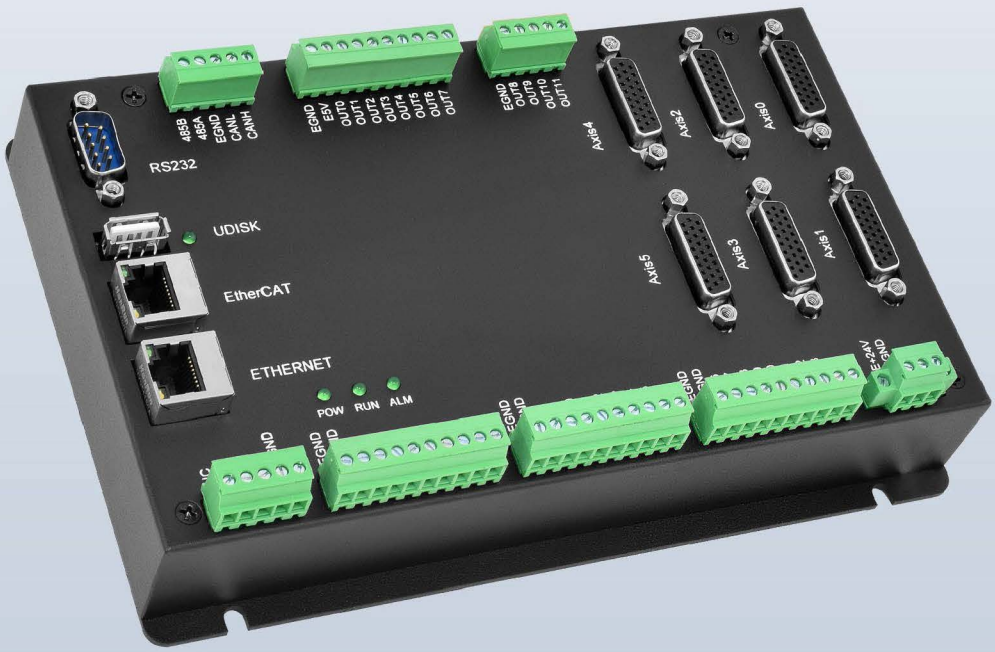
Zmotion®
正运动技术

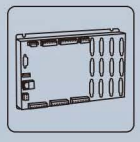
做更好用的运动控制，智造美好生活
Better Motion Control, Smarter Life

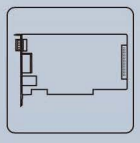
Pulse + EtherCAT Motion Controller

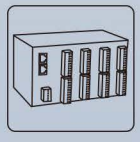
ZMC432CL-V2

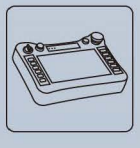


Vision Motion Controller


Motion Controller


Motion Control Card


IO Expansion Module


HMI

Foreword

The motion controller provides rich interface, and it has excellent motion control performance, which can meet the expansion requirements of various projects.

This manual is copyrighted by Shenzhen Technology Co., Ltd., without the written permission of the Zmotion Technology, no person shall reproduce, translate and copy any content in this manual. The above-mentioned actions will constitute an infringement of the copyright of the company's manual, and Zmotion will investigate legal responsibility according to law.

For details about the ZMC controller software and the introduction and routine of each command, please refer to the ZBASIC software manual.

Information contained in this manual is only for reference. Due to improvements in design and functions and other aspects, Zmotion Technology reserves the final interpretation! Subject to change without notice!

Pay attention to safety when debugging the machine!

Please be sure to design an effective safety protection device in the machine, and add an error handling program in the software, otherwise Zmotion has no obligation or responsibility for the loss caused.

In order to ensure the safe, normal and effective use of the product, please be sure to read this product manual carefully before installing and using the product.

Safety Statement

- This chapter describes the safety precautions required for the correct use of this product. Before using this product, please read the instructions for use and correctly understand the relevant information on safety precautions.
- This product should be used in an environment that meets the design specifications, otherwise it may cause equipment damage or personal injury, and malfunctions or component damage caused by failure to comply with relevant regulations are not within the scope of product quality assurance.
- Zmotion will not take any legal responsibility for personal safety accidents and property losses caused by failure to comply with the contents of this manual or illegal operation of products.

Safety Level Definition

According to the level, it can be divided into " Danger " and " Caution ". Failure to operate as required may result in moderate injury, minor injury or equipment damage.

Please keep this guide in a safe place for reading when needed, and be sure to hand this manual to the end user.

Install	
Danger	◆ When the controller is disassembled, all external power supplies used by the system should be disconnected before operation, otherwise it may cause misoperation or damage to the equipment. ◆ It is forbidden to use in the following places: places with dust, oil fume, conductive dust, corrosive gas and flammable gas; places exposed to high temperature, condensation, wind and rain; places with vibration and shock. Electric shock, fire and misuse can cause product damage and deterioration.
Notice	◆ Avoid metal shavings and wire ends falling into the hardware circuit board during installation. ◆ After installation, ensure that there are no foreign objects on the hardware circuit board. ◆ When installing, make it tightly and firmly with the mounting frame.

	◆ Improper installation of the controller may result in misoperation, failure and fire.
Wiring	
 Danger	◆ The specifications and installation methods of the external wiring of the equipment shall comply with the requirements of local power distribution regulations. ◆ When wiring, all external power supplies used by the system should be disconnected before operation. ◆ When powering on and running after the wiring work is completed, the terminals attached to the product must be installed. ◆ Cable terminals should be well insulated to ensure that the insulation distance between cables will not be reduced after the cables are installed on the terminal block.
 Notice	◆ Avoid metal shavings and wire ends falling into the hardware circuit board during installation. ◆ The cable connection should be carried out correctly on the basis of confirming the type of the connected interface. ◆ It should be confirmed that the cables pressed into the terminals are in good contact. ◆ Do not bundle the control wires and communication cables with the main circuit or power supply wires, etc., and the distance between the wires should be more than 100 mm, otherwise noise may cause malfunction. ◆ If the controller is not installed properly, it may cause electric shock or equipment failure or malfunction.

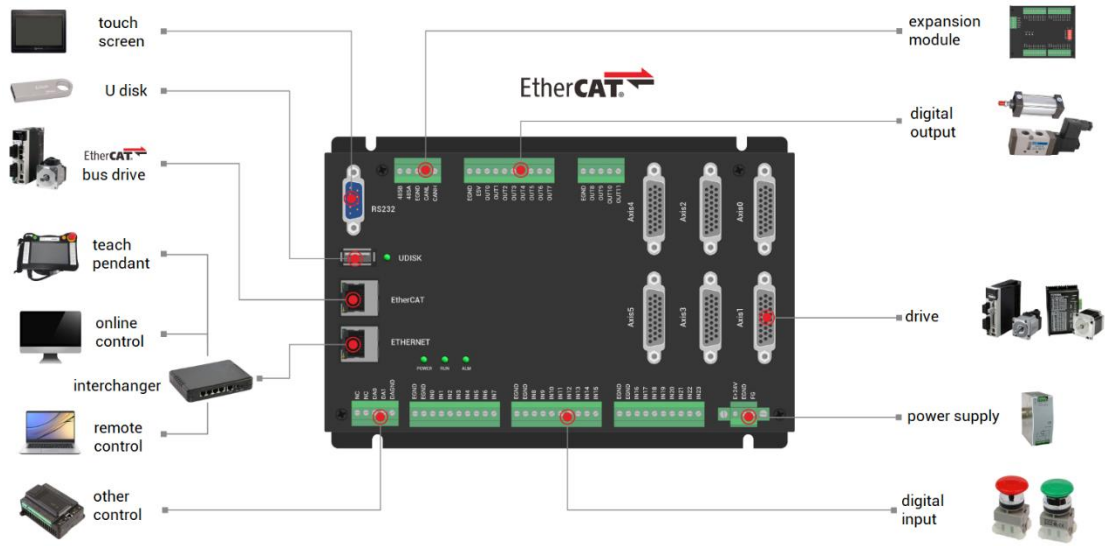
Content

Chapter I Production Information....2
1.1. Product Information....2
1.2. Interface Introduction....3
1.3. Specification Model....4
1.4. Environment Requirements....5
1.5. Hardware Installment....7
Chapter II Hardware Interface....8
2.1. Power Input....8
2.2. IN: General Digital Inputs....8
2.3. OUT: General Digital Output....11
2.4. DA Analog Output....15
2.5. RS485, CAN Communication Interfaces....17
2.6. RS232 Serial Port....20
2.7. U Disk....22
2.8. ETHERNET....23
2.9. EtherCAT Bus Interface....24
2.10. AXIS Differential Pulse Axis Interface....26
2.10.1. AXIS Interface Signal Specification & Wiring....28
2.11. Pulse Full Closed-Loop Usage Methods....32
Chapter III Expansion Module....37
3.1. CAN Bus Expansion....37
3.2. EtherCAT Bus Expansion....41
Chapter IV Program & Applications....45
4.1. ZDevelop Software Usage....45
4.2. PC Upper-Computer Program Application....50
Chapter V Run and Maintain....53
5.1. Regular Inspection and Maintenance....53
5.2. Common Problems....54

Chapter I Production Information

1.1. Product Information

ZMC432CL-V2 **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 32 axes at most to achieve complex continuous trajectory control requirements.

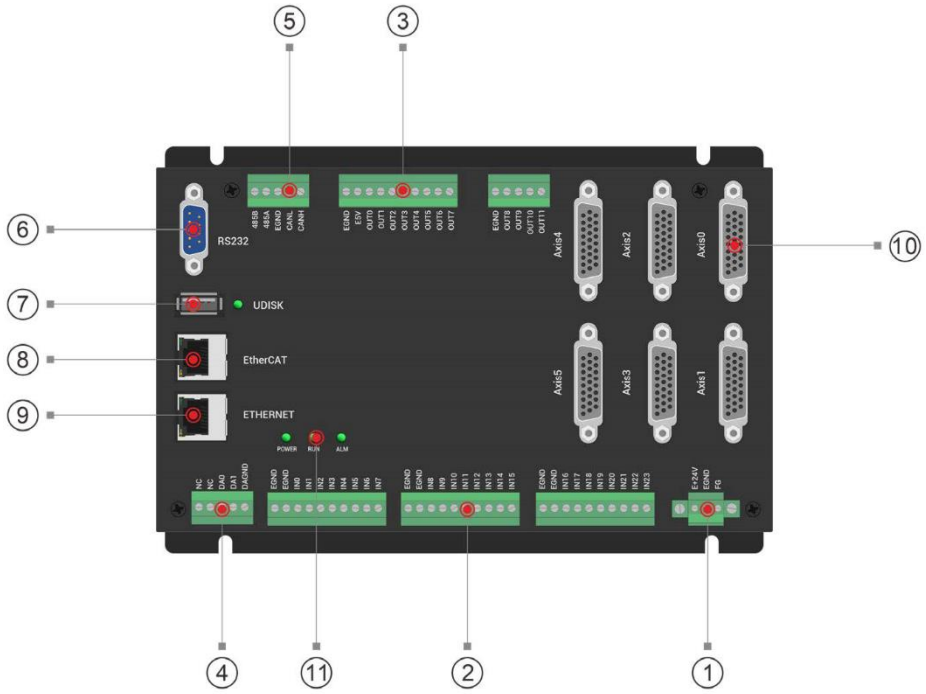


- It has EtherCAT bus axis, single-ended pulse axis, differential pulse axis and differential encoder interface.
- **Basic Motion Control Function:** point motion, electronic cam, linear interpolation, circular interpolation, continuous interpolation, SCARA, etc.
- Special Motion Control Function: HW hardware comparison output, high-speed latch, PWM, pulse full closed-loop, helical compensation, etc.

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. Interface Introduction



ZMC432CL-V2

No.	Interface	Description
①	Power	Connect to 24V DC power supply
②	General Digital IN	NPN type, IN0-23, it supports single-ended encoder, latch functions
③	General Digital OUT	NPN / leakage type, OUT0-11, it supports single-ended pulse, HW, PWM functions
④	Analog DA	Single-ended type, 12bit, 0-10V
⑤	CAN	Connect to CAN expansion modules to expand more resources.
	RS485	Connect to host computer, use MODBUS_RTU protocol
⑥	RS232	Connect to host computer, use MODBUS_RTU protocol
⑦	UDISK	Connect to UDISK to update the program,

		import and export data.
⑧	EtherCAT	Connect to EtherCAT bus drive and EtherCAT bus expansion module.
⑨	EtherNET	Connect to host computer, use MODBUS_TCP protocol, it can be expanded through interchanger.
⑩	AXIS Differential Pulse Axes	Connect to servo drives, including direction pulse, encoder, specialized digital IO.
⑪	Controller Status Led	POW led: it is ON when power is connected.
		RUN led: it is ON when it runs normally.
		ALM led: it is ON when it runs abnormally.

1.3. Specification Model

Model	ZMC432CL-V2
Basic Axes	32
All Axes	32 (basic axis + virtual axis)
EtherCAT	√
AXIS (differential pulse axis)	6 (direction pulse + encoder + digital IO)
OUT (single-ended pulse axis)	2 (depend on firmware)
IN (single-ended encoder axis)	1
Digital IN	24 (general) + 6 (specialized for AXIS)
Digital OP	24 (general) + 6 (specialized for AXIS)
Max Digital IN to be Expanded	≤4096
Max Digital OP to be Expanded	≤4096
DA Output	2 (0-10V, 12bit)
Max AD to be Expanded	≤512

Max DA to be Expanded	≤512
EtherNET	1
EtherCAT	1
CAN	1
RS232	1
RS485	1
UDISK	1
High-Speed Latch	2
Hardware Comparison Output HW	4
General PWM	4
Point Motion	√
Electronic Cam	
Linear Interpolation	
Circular Interpolation	
Continuous Interpolation	
SCARA Robotic Arm	
Program Space	32M Byte
Power Down Storage	√
Dimensions	205*135*32

1.4. Environment Requirements

Item	Parameters
Work Temperature	-10℃-55℃
Work relative Humidity	10%-95% non-condensing

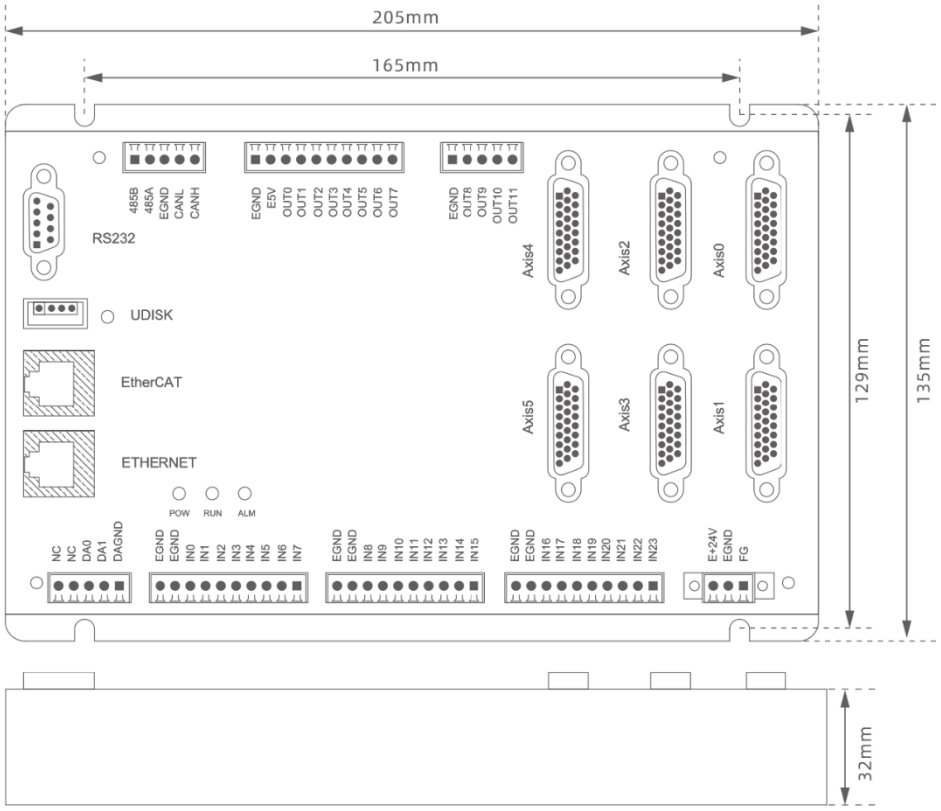
Storage Temperature		-40℃ ~ 80℃ (not frozen)
Storage Humidity		Below 90%RH (no frost)
vibration	Frequency	5-150Hz
	Displacement	3.5mm(directly install)(<9Hz)
	Acceleration	1g(directly install)(>9Hz)
	Direction	3 axial direction
Shock (collide)		15g, 11ms, half sinusoid, 3 axial direction
Degree of Protection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

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. ● 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 e) there is direct sunlight

1.5. Hardware Installment

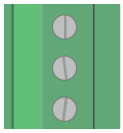
The ZMC432CL-V2 motion controller is installed horizontally with screws, and each controller should be fastened with 4 screws.

→ Unit: mm → Mounting Hole Diameter 4.5mm



Chapter II Hardware Interface

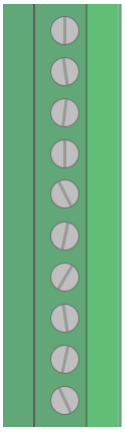
2.1. Power Input

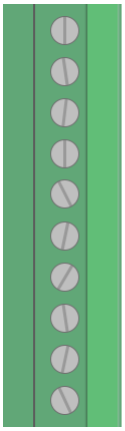
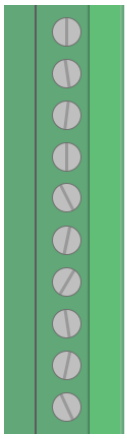
Terminal	Name	Function
	E+24V	Positive (+) terminal of DC power input (connect positive of power to positive of controller)
	EGND	Negative (-) terminal of power input
	FG	Protection

→ Specification

Item	Description
Voltage	DC24V(-5%~5%)
Max Power	10W
Anti-reverse connection	YES
Overcurrent Protection	YES
Cable Type	Recommend "1.0 mm² copper conductor cable"

2.2. IN: General Digital Inputs

Terminal	Name	Type	Function 1	Function 2	Function 3
	EGND	/	IO Public End	/	/
	EGND	/		/	/
	IN0	NPN type, high-speed input	Input 0	High Speed Latch	EA (AXIS 6)
	IN1		Input 1		EB (AXIS 6)
	IN2	NPN type, low-speed input	Input 2	/	
	IN3		Input 3	/	
	IN4		Input 4	/	
	IN5		Input 5	/	
	IN6		Input 6	/	
	IN7		Input 7	/	

	EGND	/	IO Public End	/	
	EGND	/		/	
	IN8	NPN type, low-speed input	Input 8	/	
	IN9		Input 9	/	
	IN10		Input 10	/	
	IN11		Input 11	/	
	IN12		Input 12	/	
	IN13		Input 13	/	
	IN14		Input 14	/	
	IN15		Input 15	/	
	EGND	/	IO Public End	/	
	EGND	/		/	
	IN16	NPN type, low-speed input	Input 16	/	
	IN17		Input 17	/	
	IN18		Input 18	/	
	IN19		Input 19	/	
	IN20		Input 20	/	
	IN21		Input 21	/	
	IN22		Input 22	/	
	IN23		Input 23	/	

Note:

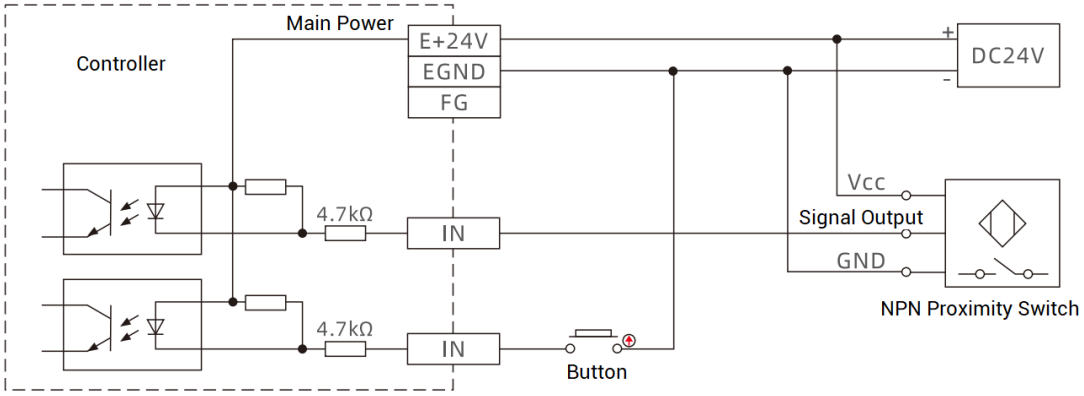
When the controller's firmware version is above 20170515, IN0-1 support single-ended encoder function, and it must use AXIS_ADDRESS command to map, then configure ATYPE, while mapping, the physical axis No. axis 6 should be used. If AXIS_ADDRESS command is not used for mapping, it only sets ATYPE, then it can't identify physical axis 6. At this time, if operates axis 6, it may recognize axis 0, then operate axis 0, the error will be caused. When ATYPE = 0, IN 0 -1 are ordinary inputs.

→ Specification

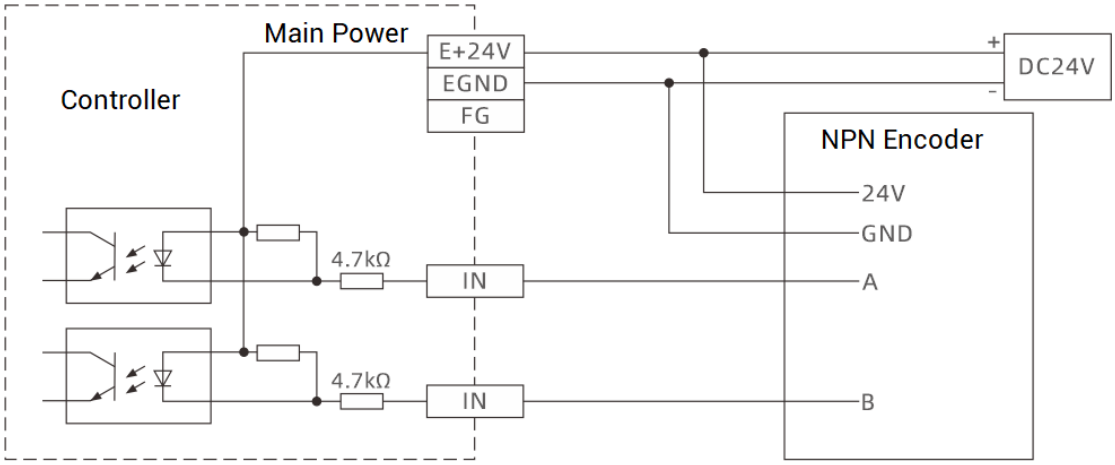
Item	High-Speed Input (IN0-1)	Low-Speed Input (IN2-23)
Input mode	NPN type, the input is triggered by low-electric level	
Frequency	< 400kHz	< 5kHz
Impedance	4.7KΩ	4.7KΩ
Voltage	≤24V	≤24V
Communication Isolation	√	√

→ Wiring Reference

General Wiring:



Single-Ended Encoder Wiring:



● Notes:

- The wiring for the digital input is as shown in the diagram above. The external load can be a button switch, sensor, etc., but it is important to ensure that the signal specifications match.
- It is recommended to use the same power supply for both the load and the controller, otherwise, the negative terminals of two power supplies should be connected.
- During on-site wiring, maintain a distance from high-voltage power lines, preferably 30 cm or more.

- Ensure that the controller is properly grounded, and the chassis ground should be connected to a standard factory ground rod.

→ Basic Usage Method

(1) Please follow the above wiring instructions to wiring correctly, then connect the controller to RTSys through ethernet / serial port.

(2) 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 “ZBasic" for details.

(3) Latch function can be set and triggered through **“REGIST”, “REV_IN”, “DATUM_IN”** commands.

(4) Axis position limit signals / origin signals can be set by **“FWD_IN”, “REV_IN”, “DATUM_IN”** commands.

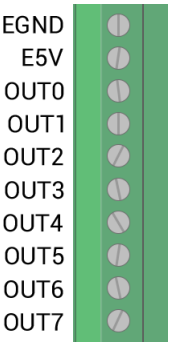
(5) Above commands and other commands, please refer to “BASIC Programming Manual".

(6) How to map axis while using IN0-1 single-ended encoder function

AXIS_ADDRESS (axis No. to be mapped) = (-1<<16) + the local pulse axis No. to be modified, now physical axis 6 is used.
BASE (axis No. to be mapped)
ATYPE = 0
BASE (the local pulse axis No. to be modified, now physical axis 6 is used)
ATYPE = 0
BASE (axis No. to be mapped)
ATYPE = the axis is with encoder

2.3. OUT: General Digital Output

Terminal	Name	Type	Function 1	2	Function 3	4
	EGND	/	IO public end / 5V power	/	/	

			ground			
	E5V	/	5V power output, max is 300mA	/	/	
	OUT0	NPN Leakage type, high-speed out	Output 0	PWM 0	Hardware Comparison Output	PUL6 (AXIS 6)
	OUT1		Output 1	PWM 1		DIR6 (AXIS 6)
	OUT2	NPN Leakage type, low-speed out	Output 2	PWM 2		PUL7 (AXIS 7)
	OUT3		Output 3	PWM 3		DIR7 (AXIS 7)
	OUT4		Output 4	/	/	
	OUT5		Output 5	/	/	
	OUT6		Output 6	/	/	
	OUT7		Output 7	/	/	
	EGND	/	IO public end	/	/	
	OUT8	NPN Leakage type, low-speed out	Output 8	/	/	
	OUT9		Output 9	/	/	
	OUT10		Output 10	/	/	
	OUT11		Output 11	/	/	

Notes:

- ✧ 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.
- ✧ OUT0-3 have the functions of PWM and hardware comparison output, among which OUT2 and 3 are low-speed outputs.
- ✧ OUT0-3 support single-ended pulse-axis (axis 6 & axis 7), and it must use AXIS_ADDRESS command to map, then configure ATYPE. When ATYPE = 0, they are general outputs. Please attention the highest frequency of axis 6 of OUT0-1 is 400kHz, for axis 7 of OUT2-3 is 8kHz.

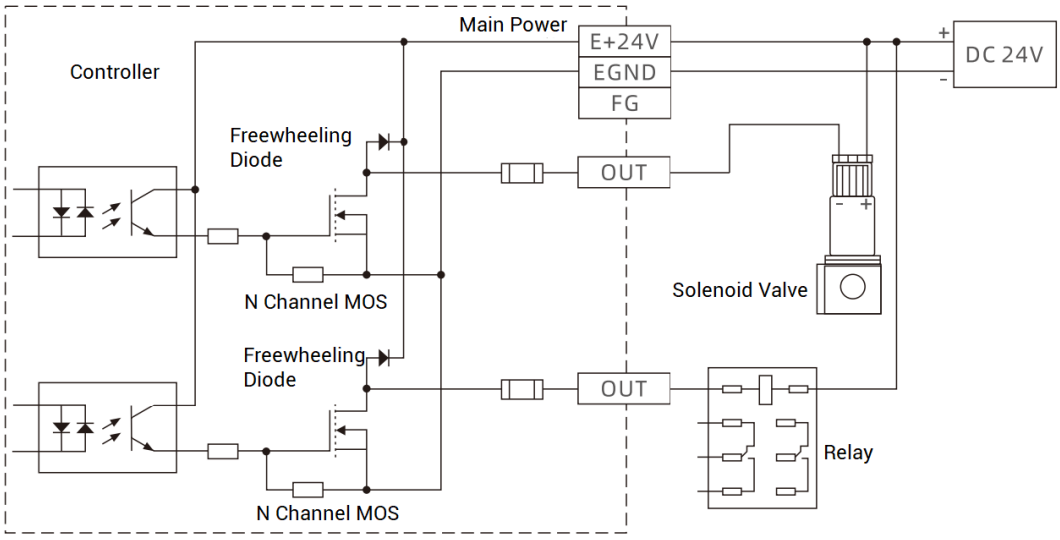
→ Specification

Item	High Speed Output (OUT0-1)	Low Speed Output (OUT2-11)
Output mode	NPN / leakage type	
Frequency	< 400kHz	< 8kHz

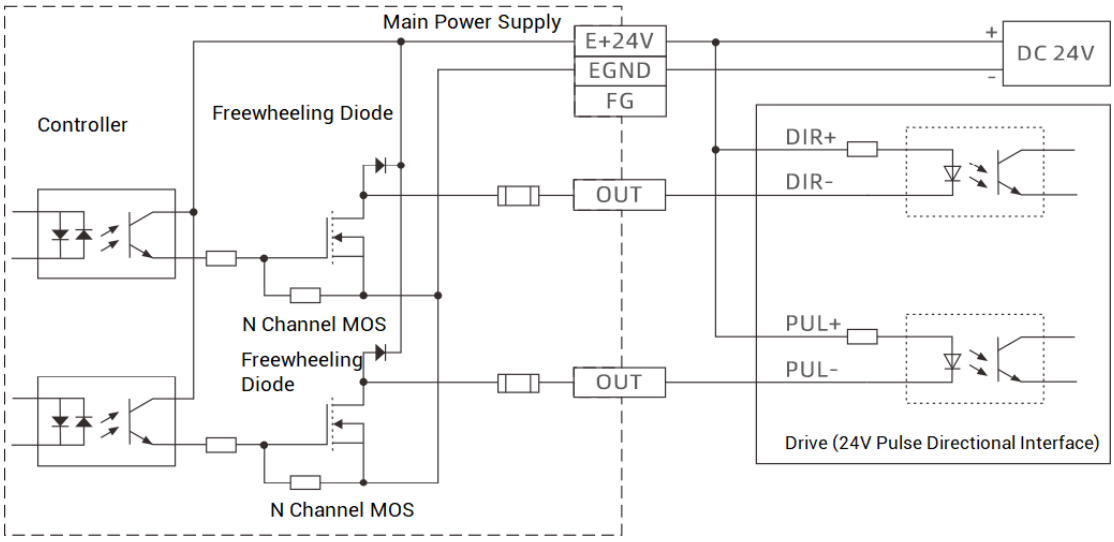
Voltage	≤24V	≤24V
Output Current	≤300mA	≤300mA
Overcurrent protection	√	√
Communication Isolation	√	√

→ Wiring Reference

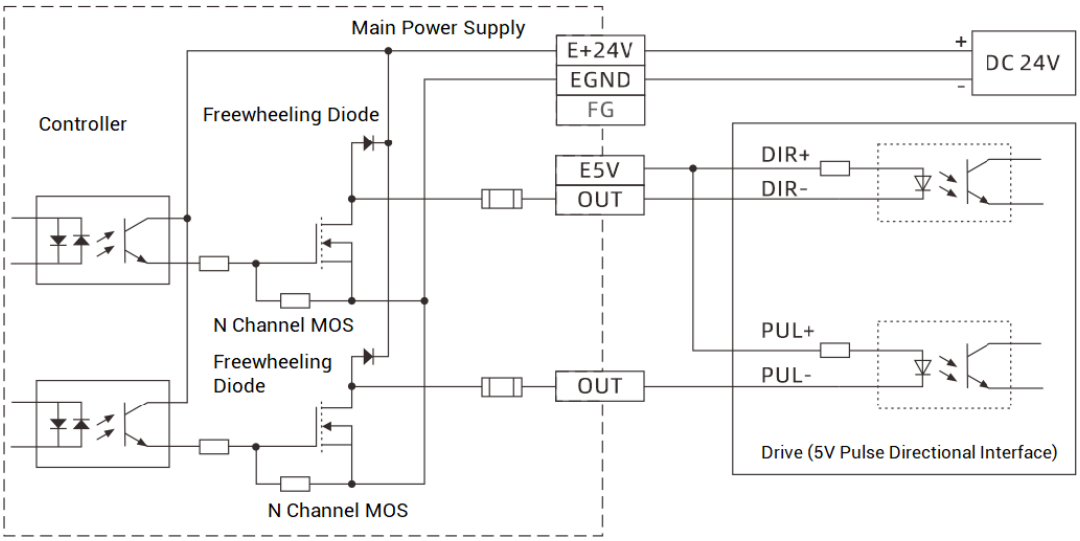
General Wiring:



Single-Ended Pulse Wiring (24V):



Single-Ended Pulse Wiring (5V):



Notes

- The wiring for the digital output is as shown in the diagram above. The external load can be a relay, valve solenoid etc., but it is important to ensure that the signal specifications match.
- For 5V pulse direction interface, please connect PUL+ and DIR+ to E5V interface.
- It is recommended to use the same power supply for both the load and the controller, otherwise, the negative terminals of two power supplies should be connected.
- During on-site wiring, maintain a distance from high-voltage power lines, preferably 30 cm or more.
- Ensure that the controller is properly grounded, and the chassis ground should be connected to a standard factory ground rod.

→ Basic Usage Method

(1) Please follow the above wiring instructions to wiring correctly, then connect the controller to RTSys through ethernet / serial port.

(2) State values of relative output ports can be operated directly through “OP” command,

also, it can be read through "RTSys/Tool/OP". Please refer to "ZBasic" for details.

(3) PWM frequency can be set by **"PWM_FREQ"** command, PWM duty cycle can be set by **"PWM_DUTY"** command.

(4) Hardware comparison output can be set and used by **"HW_PWSIWTCH2"**.

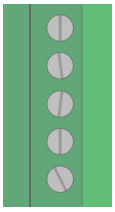
(5) When it is used as pulse axis, the usage method is same as AXIS. For details, please refer to "2.10 AXIS Interface" – "Usage Method".

(6) Above commands and other commands, please refer to "BASIC Programming Manual".

(7) How to map axis while using IN0-1 single-ended encoder function

AXIS_ADDRESS (axis No. to be mapped) = (-1<<16) + the local pulse axis No. to be modified, now physical axis 6-7 is used.
BASE (axis No. to be mapped)
ATYPE = 0
BASE (the local pulse axis No. to be modified, now physical axis 6-7 is used)
ATYPE = 0
BASE (axis No. to be mapped)
ATYPE = the axis is with pulse output

2.4. DA Analog Output

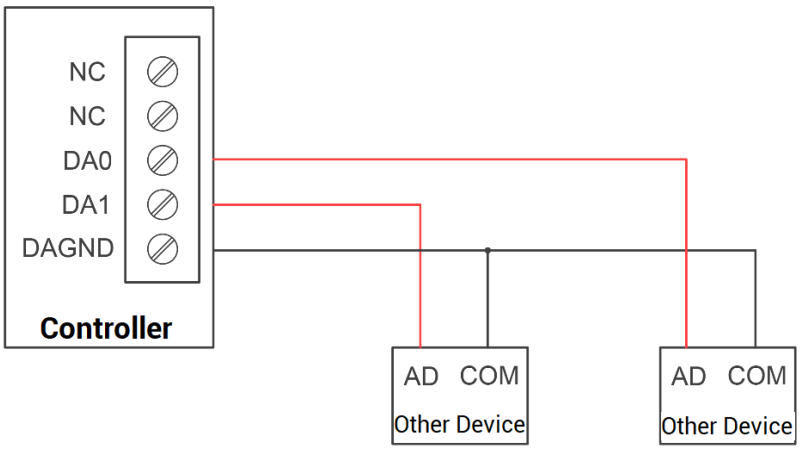
Terminal	Name	Function
	NC	Reserved
	NC	Reserved
	DA0	Analog output terminal: AOUT(0)
	DA1	Analog output terminal: AOUT(1)
	DAGND	Analog public end

→ Specification

Item	DA（0-1）
Resolution	12-bit
Data range	0-4095
Signal range	0-10V output

Data refresh ratio	1KHz (refresh ratio)
Input impedance / output load	>10KΩ (load requirement)
Signal Type	Single-ended
Cable Length	Recommend <5m

→ Wiring Reference



● Notes:

- The wiring for the analog is as shown in the diagram above. The load signal should be matched with impedance, otherwise, it will affect precision.
- Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.
- During on-site wiring, maintain a distance from high-voltage power lines, preferably 30 cm or more.
- Ensure that the controller is properly grounded, and the chassis ground should be connected to a standard factory ground rod.

→ Basic Usage Method

(1) Please follow the above wiring instructions to wiring correctly, then connect the

controller to RTSys through ethernet / serial port.

(2) Analog input can be read by AIN, analog output can be operated by AOUT. Also, "RTSys>Tool>AD/DA" can directly check and operate each channel data.

(3) Above commands and other commands, please refer to "BASIC Programming Manual".

2.5. RS485, CAN Communication Interfaces

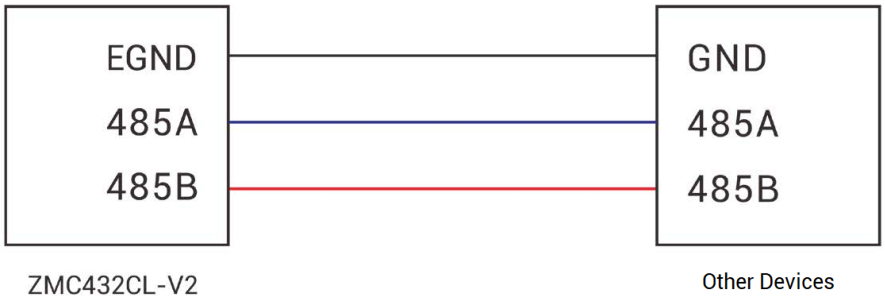
Terminal	Name	Function
	485B	RS485 Signal B/ -
	485A	RS485 Signal A/ +
	EGND	Communication Public End
	CANL	CAN Side L
	CANH	CAN Side H

→ Specification

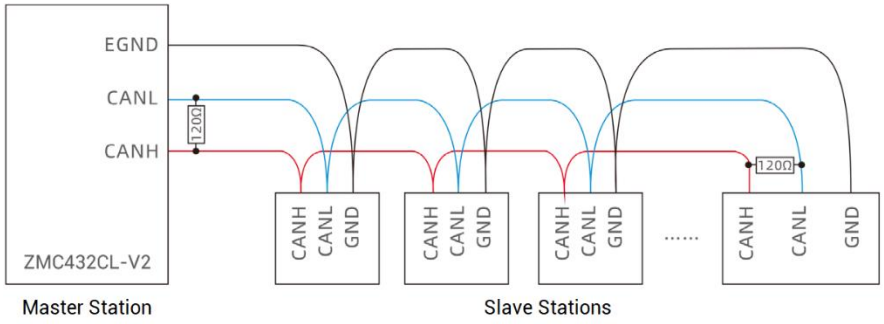
Item	RS485 (port1)	CAN
Communication Rate	≤115200bps	≤1Mbps
Terminal Resistor	120Ω	120Ω
Topological Structure	Daisy Chain Topology	
The number of nodes can be extended	Up to 127	≤16
Wiring Length	Recommend <30m	Recommend <30m (500kbps)
Communication Isolation	√	

→ Wiring Reference

- RS485 Wiring:



- CAN Wiring:



Notes:

- As above, the daisy chain topology is used for wiring (the star topology structure cannot be used). The distance between nodes, the better.
- 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 / RS485 chip from burning out.
- Please use STP (Shielded Twisted Pair), especially in bad environments, and make sure the shielding layer is fully grounded.
- During on-site wiring, maintain a distance from high-voltage power lines, preferably 30 cm or more.

- Ensure that the controller is properly grounded, and the chassis ground should be connected to a standard factory ground rod.

→ Usage Methods:

(1) Please follow the above wiring instructions to wiring correctly, then connect the controller to RTSys through ethernet / serial port.

(2) How to use CAN:

1) Configure controller CAN master station:

   a) Use **"CANIO_ADDRESS"** command to set master station "address" and "velocity".

   b) Use **"CANIO_ENABLE"** command to enable / disable CAN master station function.

   c) In "RTSys>Controller>Controller Status>Communication Config" interface, you can check communication parameters.

   d) In "RTSys>Controller>Controller Status>ZCanNodes" interface, you can check bus nodes parameters.

2) Correctly set the "address" and " velocity" of the slave station CAN expansion module for completing resources mapping, you can refer to "3.1 CAN Expansion".

3) After configured, repower on all slave stations to build normal communication, if the slave module "ALM" led is ON, which means the communication fails.

4) Note that the "velocity" settings of each node on the CAN bus must be consistent, and the "address" settings cannot cause conflicts, otherwise the communication establishment will fail or the communication will be disordered.

(3) How to use RS485:

1) Please use "ADDRESS" and "SETCOM" commands to check protocol station No. and set parameters, also, you can check and set in "RTSys>Controller>Controller

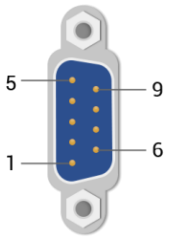
Status>Communication Config" interface.

2) According to description, set the third-party device parameters correctly to match each node.

3) After all configured, it is time to communicate.

(4) For above commands and others, please check "BASIC Programming Manual".

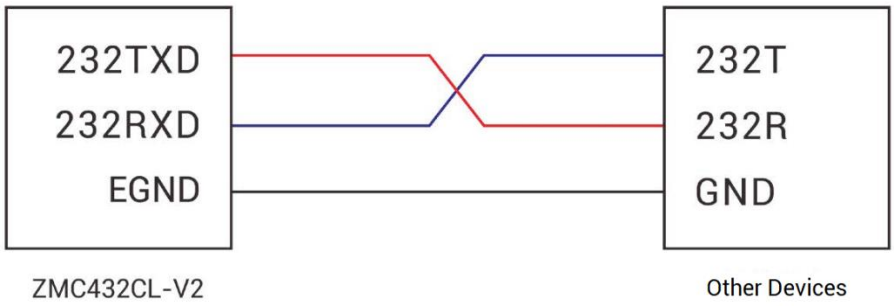
2.6. RS232 Serial Port

Terminal	PIN	Name	Function
	1, 4, 6, 7, 8	NC	Reserved
	2	RXD	RS232 signal, receive data
	3	TXD	RS232 signal, send data
	5	EGND	Negative pole output of 5V power, and output for the public end
	9	E5V	Positive pole output of 5V power, maximum is 300mA

→ Specification:

Item	RS232 (port 0)
Maximum Communication Rate	≤115200bps
Terminal Resistor	No
Topology Structure	Connect correspondingly (1 to 1)
The number of nodes can be extended	1
Wiring Distance	Recommend <5m
Communication Isolation	√

→ Wiring Reference:



→ Wiring Notes:

- The wiring of RS232 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.
- Please be sure to connect the public ends of each communication node to prevent RS232 chip from burning out.
- Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.
- During on-site wiring, maintain a distance from high-voltage power lines, preferably 30 cm or more.
- Ensure that the controller is properly grounded, and the chassis ground should be connected to a standard factory ground rod.

● Notes

(1) Please follow the above wiring instructions to wiring correctly, then connect the controller to RTSys through ethernet / serial port.

(2) Please use "ADDRESS" and "SETCOM" commands to check protocol station No. and set parameters, also, you can check and set in "RTSys>Controller>Controller Status>Communication Config" interface.

(3) According to description, set the third-party device parameters correctly to match

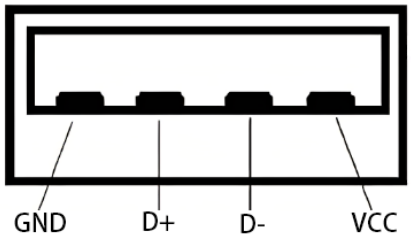
each node.

(4) After all configured, it is time to communicate.

(5) For above commands and others, please check “BASIC Programming Manual”.

2.7. U Disk

The ZMC432CL-V2 motion controller provides a USB communication interface, which can insert the U disk device. It is used for ZAR program upgrading, controller data importing and exporting, file 3 executing, etc. Its schematic diagram is shown in the figure below:



→ Specification

Item	Description
Communication Protocol	USB2.0
Communication Velocity	≤12Mbps
Whether Isolates	No

→ Usage Methods

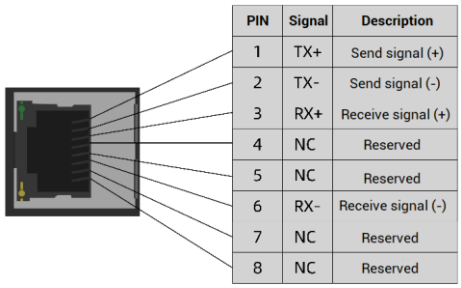
(1) Please follow the above wiring instructions to wiring correctly, then connect the controller to RTSys through ethernet / serial port.

(2) Insert UDISK into controller UDISK terminal, when it is connected successfully, the UDISK led will be ON, then you can use U_STATE command to check UDISK state. When you make sure the communication is OK, corresponding operations (firmware upgrade, data copy, etc.) can be done (by commands of FILE series).

(3) For above commands and others, please check “BASIC Programming Manual”.

2.8. ETHERNET

→ Specification

PIN Definition	Item	Description
	Communication Protocol	MODBUS_TCP
	Communication Velocity	100Mbps
	Default IP	192.168.0.11
	Communication Cable	Shielded Cat 5e Cable
	Cable Length	Recommend <50m

→ Wiring

(1) Controller Ethernet can be connected to PC / HMI by one shielded cat 5e cable.

(2) Controller also can be connected to the interchanger for expanding ethernet channels, then connect to other devices.

(3) Ethernet LED state:

Led	Common-ON	Shrink
Green	Build 100M communication	Now it is sending & receiving data
Yellow	Build 10M communication	Now it is sending & receiving data

- Notes

- Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.
- During on-site wiring, maintain a distance from high-voltage power lines, preferably 30 cm or more.
- Ensure that the controller is properly grounded, and the chassis ground should be

connected to a standard factory ground rod.

→ Usage Methods

(1) Please follow the above wiring instructions to wiring correctly, then connect the controller to RTSys through ethernet / serial port.

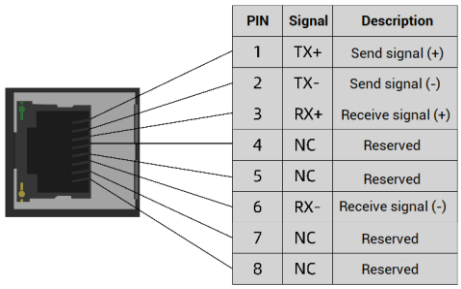
(2) "IP_ADDRESS" command can modify controller IP, please note controller IP and PC IP should be in same net segment.

(3) It supports custom ethernet communication, "OPEN#" can open custom ethernet communication, "CLOSE#" can close it, "GET#" can read data from the channel and save data into it.

(4) For above commands and others, please check "BASIC Programming Manual".

2.9. EtherCAT Bus Interface

→ Specification

PIN Definition	Item	Description
	Communication Protocol	MODBUS_TCP
	Communication Velocity	100Mbps
	Refresh Ratio	Max: 500us
	Communication Cable	Shielded Cat 5e Cable
	Cable Length	Recommend <50m

→ Wiring

(1) Controller Ethernet can be connected to PC / HMI by one shielded cat 5e cable.

(2) Controller also can be connected to the interchanger for expanding ethernet channels,

then connect to other devices.

(3) Ethernet LED state:

Led	Common-ON	Shrink
Green	Build 100M communication	Now it is sending & receiving data
Yellow	Build 10M communication	Now it is sending & receiving data

- Notes

- Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.
- During on-site wiring, maintain a distance from high-voltage power lines, preferably 30 cm or more.
- Ensure that the controller is properly grounded, and the chassis ground should be connected to a standard factory ground rod.

→ Usage Methods

(1) Please follow the above wiring instructions to wiring correctly, then connect the controller to RTSys through ethernet / serial port.

(2) How to connect to drive devices through EtherCAT:

1) Use SLOT_SCAN command to scan the slot No. on the bus.

2) Use AXIS_ADDRESS command to map axis No., you can refer to "3.2 EtherCAT Expansion" – "Resource Mapping".

3) Use SLOT_START to open bus, SLOT_STOP to stop.

4) After completed, configure and operate local pulse axis, for details, please refer to "2.10 AXIS Interface" – "Usage Methods".

(3) How to connect to expansion module by EtherCAT:

1) Use SLOT_SCAN command to scan the slot No. on the bus.

2) Use AXIS_ADDRESS command to map axis No., use NODE_IO / NODE_AIO command to map IO No., you can refer to "3.2 EtherCAT Expansion" – "Resource Mapping".

3) Use SLOT_START to open bus, SLOT_STOP to stop.

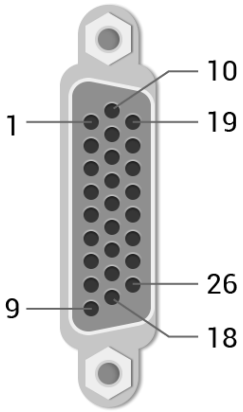
4) After completed, configure and operate local IO, same as axis, for details, please refer to "2.2, 2.3, 2.10" – "Usage Methods".

(4) For slot No. and device node information, you can check in "RTSys>controller>controller status>slot0node".

(5) For above commands and others, please check "BASIC Programming Manual".

2.10. AXIS Differential Pulse Axis Interface

This product provides 6 local differential pulse axis interfaces to control and communicate with servo drive.

Interface	Pin	Signal	Description
	1	EGND	Negative pole of IO 24V power
	2	IN24-29/ALM	General input (recommended as driver alarm)
	3	OUT12-17/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- / SSI_DAT- / BISS_DAT -
	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 - / SSI_CLK- / BISS_CLK -
	12	Reserved	Reserved
	13	GND	Negative pole of 5V power of pulse/encoder signal
	14	OVCC	Positive pole of IO 24V power
	15	Reserved	Reserved
	16	Reserved	Reserved
	17	EA+	Encoder differential input signal A+
	18	EB+	Encoder differential input signal B+
	19	EZ+	Encoder differential input signal Z+ / SSI_DAT+ / BISS_DAT +
	20	GND	Negative pole of 5V power of pulse/encoder signal
	21	GND	
	22	DIR-	Servo or step direction output differential signal -
	23	PUL+	Servo or step pulse output differential signal + / SSI_CLK+ / BISS_CLK +
	24	GND	Negative pole of 5V power of pulse/encoder signal
	25	Reserved	Reserved
	26	Reserved	Reserved

Note:

- ALM and ENABLE 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.
- SSI / BISS absolute values use PUL signal to do as CLK output, and use EZ signal to do as DAT input, therefore, they can't use as pulse-axis at the same time. But when EZ is not needed for AB encoder, synchronous use can be valid.
- If the controller's firmware is with Ssi, which means it supports SSI/BISS absolute value encoder.

- ✧ Then, check the controller state, it can be seen ZMC432CL-V2 Axis 4 and Axis 5 support SSI/BISS absolute value encoder.
- ✧ When using absolute encoders, please attention resolution and clock frequency of controller and encoder, if their clock frequencies are different, ENCODER_BITS can help modify, after setting, repower on it.

Pulse-axis PIN and IO corresponding relation:

Pulse-Axis No.	Corresponding IN (PIN2)	Corresponding OUT (PIN3)
AXIS 0	24	12
AXIS 1	25	13
AXIS 2	26	14
AXIS 3	27	15
AXIS 4	28	16
AXIS 5	29	17

2.10.1. AXIS Interface Signal Specification & Wiring

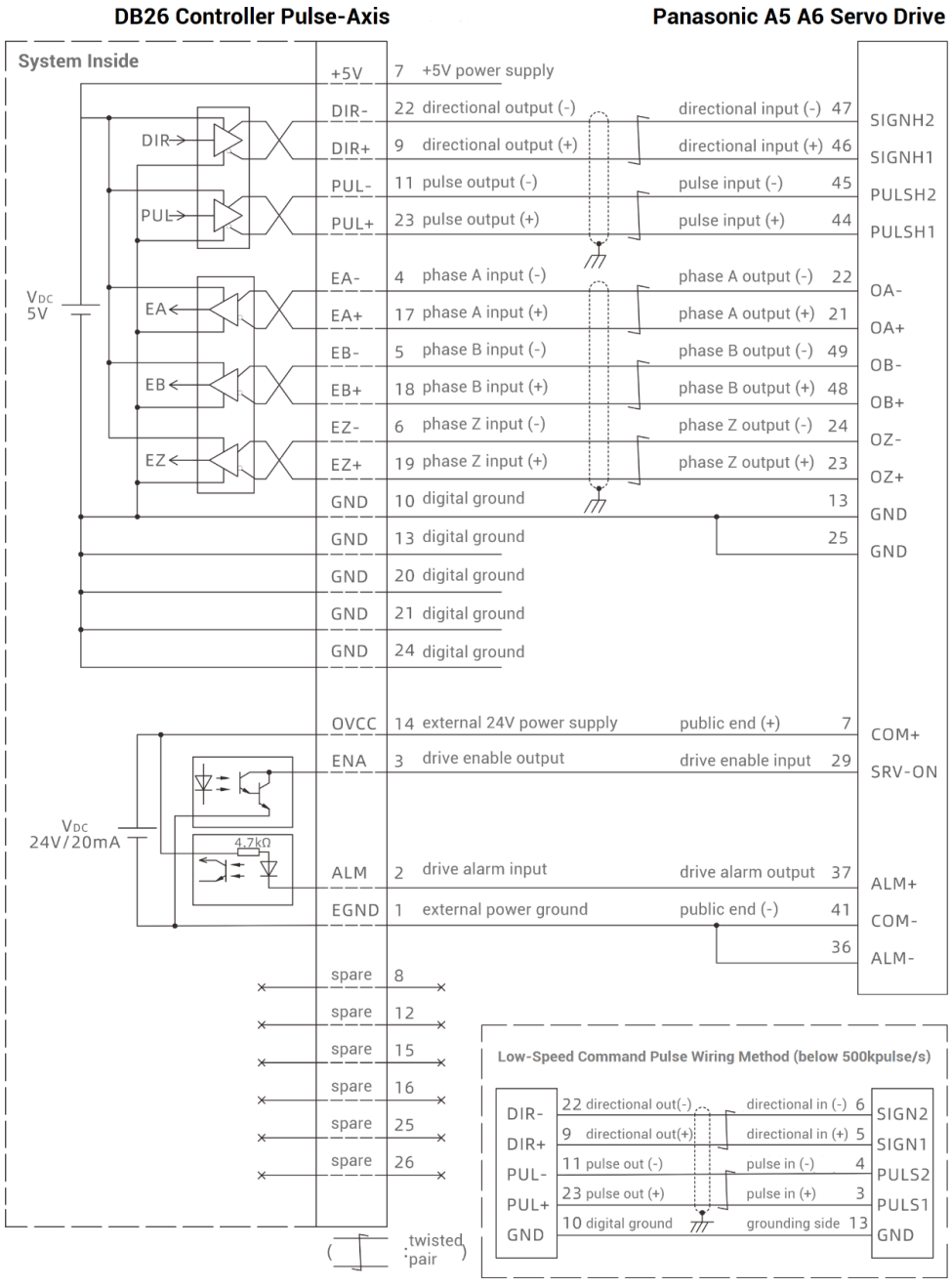
→ Specification:

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
IN24-29	Input method	NPN type, it is triggered by low electric level
	Frequency	< 5kHz
	Impedance	6.8KΩ
	Voltage level	≤24V
	Communication Isolation	YES
OUT12-17	Output method	NPN / leakage type
	Frequency	< 8kHz

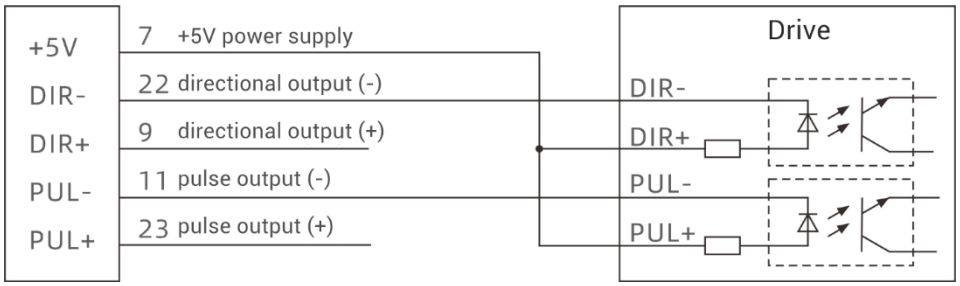
	Voltage level	≤24V
	Output Current	≤50mA
	Overcurrent protection	No
	Communication Isolation	YES
+5V	Output Current	≤50mA
OVCC	Output Current	≤50mA

→ Wiring Reference:

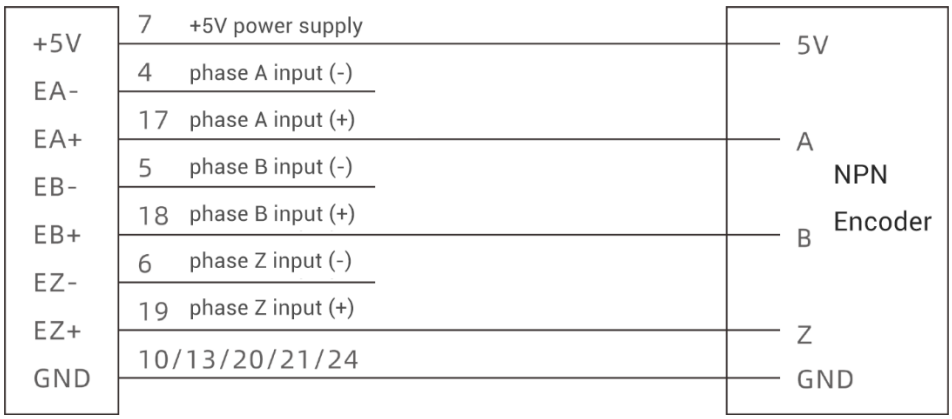
Reference example of wiring with Panasonic A5/A6 servo driver.



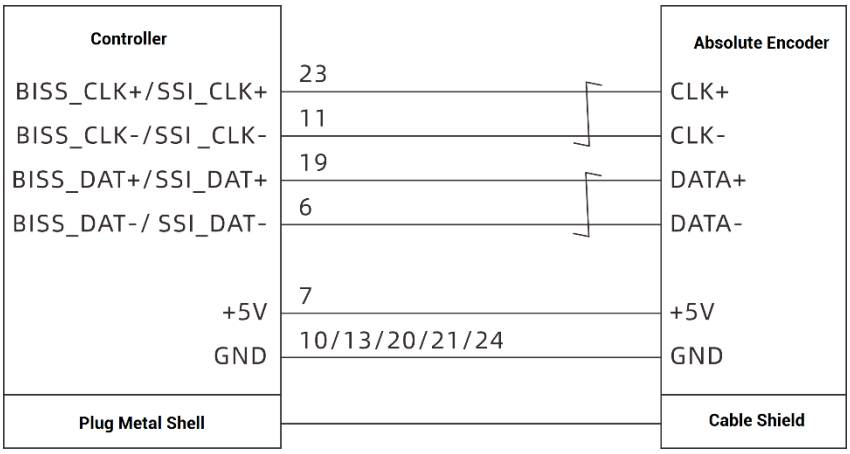
Wiring Reference of PUL/DIR common anode:



Wiring Reference of EA/EB/EZ common anode:



Wiring of absolute encoder:



Notes:

- The wiring principle for the AXIS interface is as shown in the diagram above. Please

note that wiring methods may vary depending on the drive model, so connect with caution.

- When using differential signals, ensure that the common ground terminals on both sides are connected to guarantee communication stability and equipment safety.
- Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.
- During on-site wiring, maintain a distance from high-voltage power lines, preferably 30 cm or more.
- Ensure that the controller is properly grounded, and the chassis ground should be connected to a standard factory ground rod.

→ Usage Methods

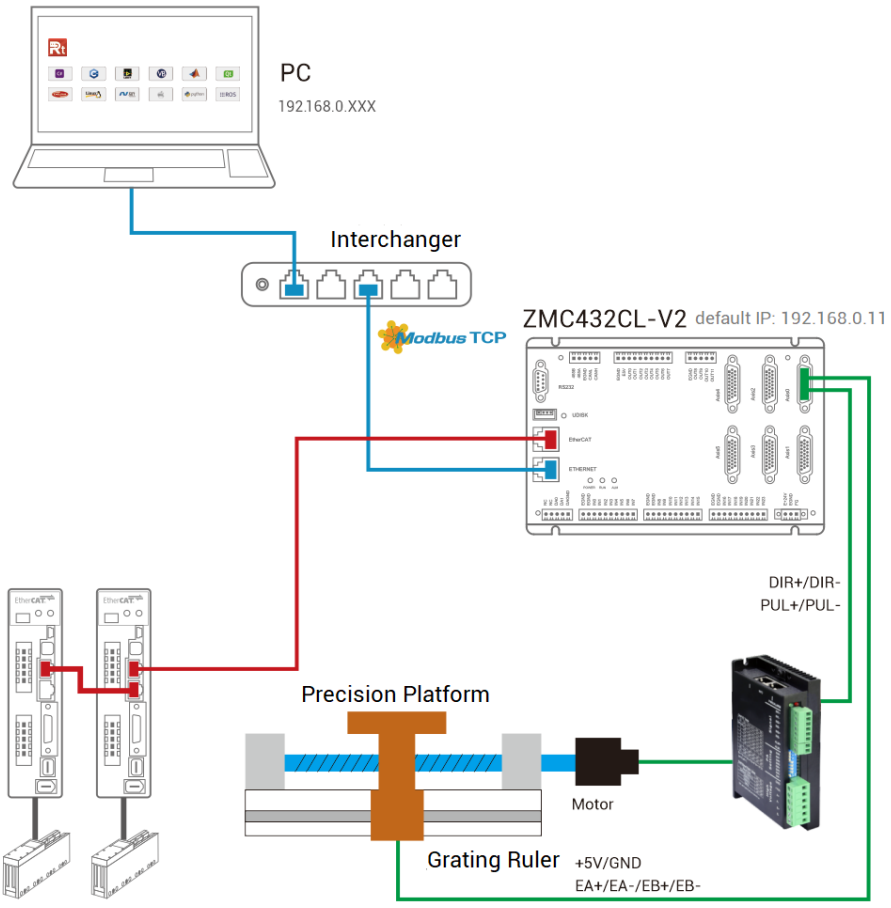
(1) Please follow the above wiring instructions to wiring correctly, then connect the controller to RTSys through ethernet / serial port.

(2) Set basic axis parameters, such as, BASE, ATYPE, UNITS, SPEED, ACCEL, FWD_IN, REV_IN, etc., and do axis mapping through AXIS_ADDRESS command. Then, enable by AXIS_ENABLE, do linear motion by MOVE.

(3) Above parameters can be checked and configured directly in "RTSys>View>Axis Parameter", for some position parameters, they also can be operated in "RTSys>Tool>Manual".

(4) There are rich pulse axis commands, for more and details, please check in "BASIC Programming Manual".

2.11. Pulse Full Closed-Loop Usage Methods



Pulse axis Axis0~Axis5 all support sending DPOS (demand position), and getting encoder feedback MPOS, also, you can set parameters to achieve real-time closed-loop of pulse sending and encoder feedback.

--Test Program—

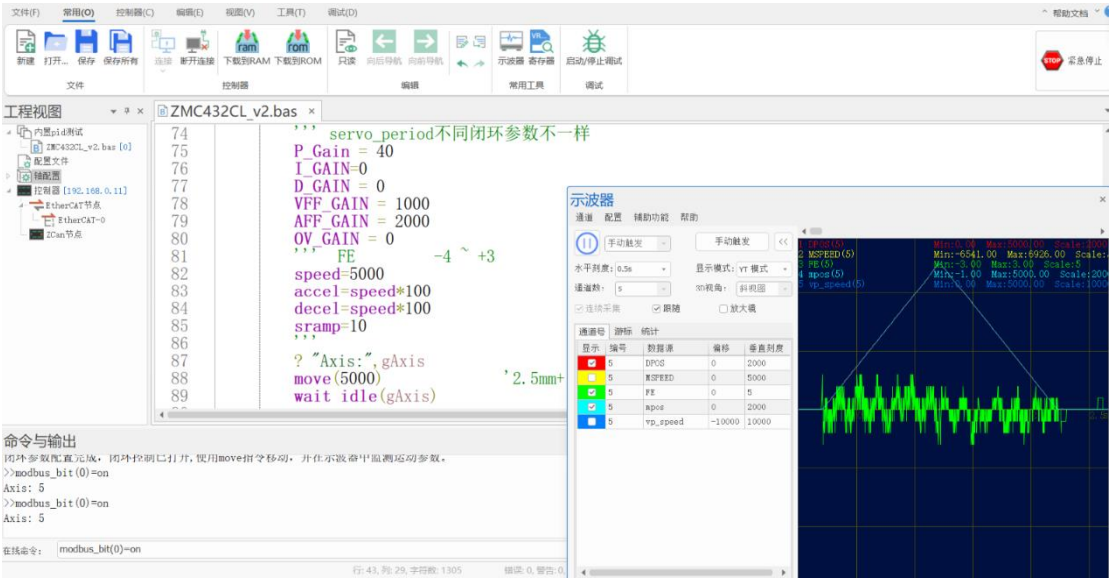
```
''' Axis0~Axis5 test, stepper pulse / closed-loop are OK…
'''pulse: stepper drive, suppose the screw pitch is 1mm, one circle means 2000 pulses, 1mm = 2000 pulses
'''feedback ruler: grating ruler, suppose 2000 count/mm
ERRSWITCH = 3    'in print window, “TRACE WARN ERROR" info all output
RAPIDSTOP(2)
DATUM(0)
WAIT IDLE(0)
```

```
global gAxis
gAxis = 5

base(gAxis)          """for convenience, select axis 5
ENCODER_SERVO = OFF   '0-OFF (default). 1-open closed-loop function, using pulse
                      speed to do PID output
SERVO = OFF           'servo, configure closed-loop switch, close
AXIS_ENABLE = 0    '''close inner enable

datum(0)

'''configure corresponding axis' motion parameters
ATYPE = 4
UNITS = 1
SPEED = 10000       '10000units/s = 5r/s = 5mm/s, max 10mm/s
ACCEL = 1000000    '500mm/s*s acceleration, 100 times of acceleration & deceleration
DECEL = 1000000
SRAMP = 10          'S acceleration & deceleration
MERGE = ON
BACKLASH(1,0)       'cancel backlash
INVERT_STEP = 256*100+0 'set the feedback pulse direction of grating ruler

'''for ratio of pulse sending and encoder, depend on real situation
STEP_RATIO(1,1)
ENCODER_RATIO(1,1)

FE_LIMIT= 2000       'set max following error, now set as 2000
FE_RANGE = 1000    'following error when alarm, now set as 1000
mpos=0
dpos=0

DIM switch          'closed-loop switch, 1-ON, 0-OFF
switch = 1
IF switch = 1 THEN
    ?"closed-loop parameters in configuration... "
```

```
    '1ms, 500us, 250us, different periods, different data
    P_Gain = 60
    I_GAIN = 0
    D_GAIN = 50
    VFF_GAIN = 2000
    AFF_GAIN = 0
    OV_GAIN = 0

    axis_enable = 1
    ENCODER_SERVO = ON    '0-OFF (default). 1-open closed-loop function
    servo = ON
    ?"closed-loop parameters are configured, now the closed-loop control is opened,
then use MOVE to move, and watch motion parameters in SCOPE.
ELSE
    ENCODER_SERVO = OFF
    SERVO = OFF
    axis_enable = 0
    ?"not open closed-loop control (switch), use open-loop to move"
ENDIF

global itick
'test control program (user assigns the movement)
while (1)
    itick = itick +1
    delay 100

    if modbus_bit(0)=on THEN
        modbus_bit(0)=0

        trigger
        delay(200)

        '''different closed-loop parameters, different servo_period
        P_Gain = 40
        I_GAIN=0
```

```
        D_GAIN = 0
        VFF_GAIN = 1000
        AFF_GAIN = 2000
        OV_GAIN = 0
        '''FE                -4 ~ +3

        speed=5000
        accel=speed*100
        decel=speed*100
        sramp=10
        '''
        ? "Axis:",gAxis
        move(5000)          '2.5mm+
        wait idle(gAxis)

        move(-5000)         '2.5mm-
        wait idle(gAxis)
        delay(3000)
    endif
wend
```

--Test Result--



--in program, enable related parameters--

```
base(gAxis) """for convenience, select axis gAxis…..
axis_enable = 1
ENCODER_SERVO = ON       '0-OFF (default). 1-open closed-loop function
servo = ON
```

--in program, disable related parameters--

```
base(gAxis) """for convenience, select axis gAxis…..
ENCODER_SERVO = OFF      ' 0-OFF (default). 1-
SERVO = OFF              'servo, closed-loop switch setting, close it
AXIS_ENABLE = 0          '"closed inner enable...
```

--in program, control effect related parameters--

```
base(gAxis) """for convenience, select axis gAxis…..
P_Gain = 40
I_GAIN=0
D_GAIN = 0
VFF_GAIN = 1000
AFF_GAIN = 2000
OV_GAIN = 0
```

- When “servo_period” is different, the closed-loop parameter is different, attention when debugging!

Chapter III Expansion Module

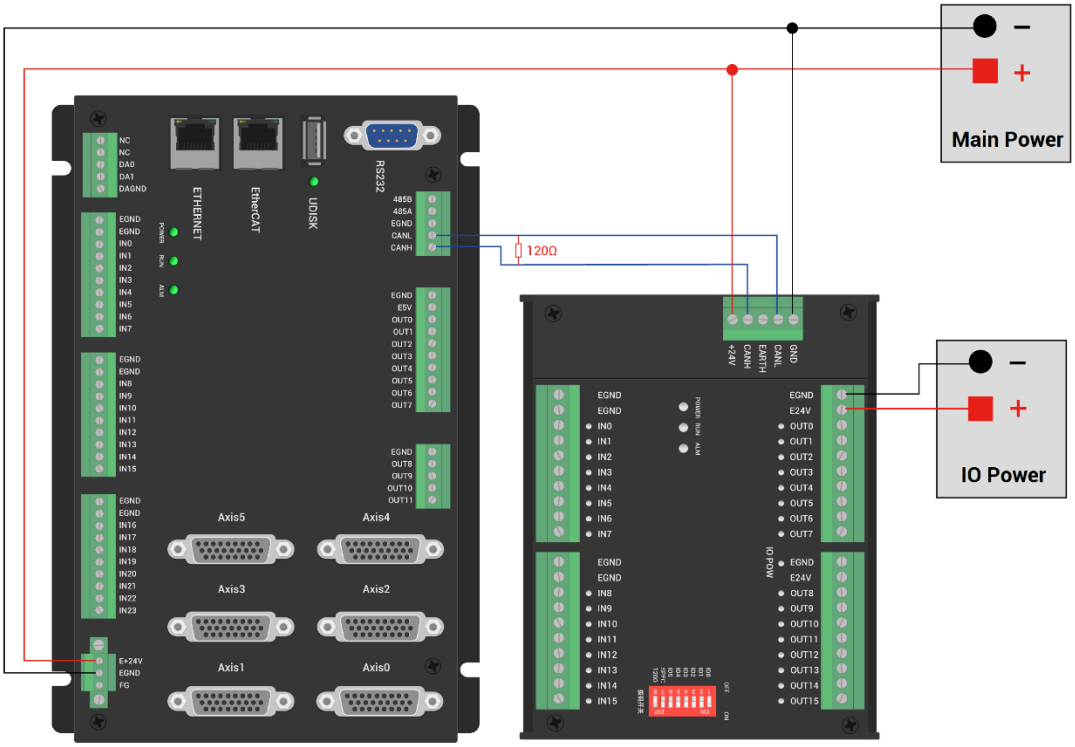
When you need more resources, expansion modules can be used by CAN / EtherCAT.

3.1. CAN Bus Expansion

For Zmotion, there are 3 types CAN bus expansion modules: ZIO, ZAIO, ZMIO-CAN, through them, more digital IO, analog IO, axes (up to 2) can be expanded.

Please select the expansion module according to the requirements, and select IO mapping or axis mapping according to the resources of the expansion module. Attention the No. must be different while mapping.

→ Wiring Reference:



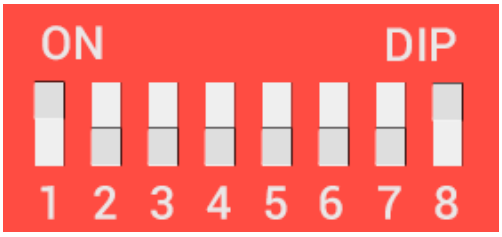
● Notes:

✧ Controller and expansion module should share one same main power supply, for ZIO and ZMIO310-CAN, their IO power supplies need to be powered separately to achieve

isolation.

✧ When connecting multiple 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).

→ Resource Mapping



DIP Switch

Generally, the ZCAN expansion module has an 8-code DIP switch to do communication configuration and resource mapping, 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.

Dial 1-4 to select the CAN address, then the controller automatically maps expansion module IO range according to this address ID, but for axis No., please map manually.

Dial code 5-6 to select CAN bus communication speed, the corresponding speeds are as follows:

DIP 5-6 combination value	CAN communication speed

0	500KBPS (default value)
1	250KBPS
2	125KBPS
3	1MBPS

● Notes:

✧ "How to configure controller as master station", please check "2.5 CAN Communication Interface" – "Usage Methods".

✧ Communication velocities of each node on the CAN bus must be consistent, please note mapped IO No. and axis No. can't conflict.

→ IO Mapping:

IO mapping of CAN expansion module is determined by dial code 1-4, below shows digital IO mapping and analog IO mapping No. allocation:

--Digital IO Mapping No.--

Code 4	Code 3	Code 2	Code 1	Address ID	Start IO No.	End IO No.
0	0	0	0	0	16	31
0	0	0	1	1	32	47
0	0	1	0	0	0	63
0	0	1	1	3	64	79
0	1	0	0	4	80	95
0	1	0	1	5	96	111
0	1	1	0	6	112	127
0	1	1	1	7	128	143
1	0	0	0	8	144	159
1	0	0	1	9	160	175
1	0	1	0	10	176	191
1	0	1	1	11	192	207
1	1	0	0	12	208	223

1	1	0	1	13	224	239
1	1	1	0	14	240	255
1	1	1	1	15	256	271

--Analog IO Mapping, for bit 1-4 code state and corresponding ID, please refer to above form--

Address ID	Start AD No.	End AD No.	Start DA No.	End AD No.
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:

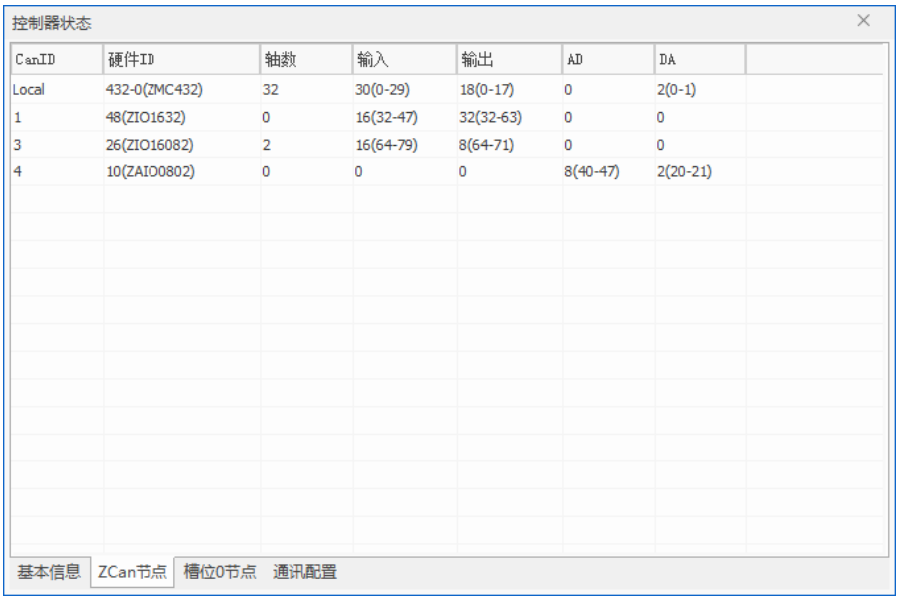
When expanding pulse axis through CAN, max 2 pulse axes can be expanded, and these two should be bound with axis No. by **AXIS_ADDRESS** at first.

"AXIS_ADDRESS(axis No.)=(32*expanded axis)+ID "

```
AXIS_ADDRESS(6)=(32*0)+2    'map axis 0 expanded by CAN module (ID is 2) as axis 6
AXIS_ADDRESS(7)=(32*1)+2    'map axis 1 expanded by CAN module (ID is 2) as axis 6
```

→ Check Expanded Resources:

Connect the controller to RTSys at first, then in RTSys, enter “controller > controller status > ZCanNode" interface, you can know expansion module ID and corresponding mapping No. (below take ZMC432 controller as the example).



3.2. EtherCAT Bus Expansion

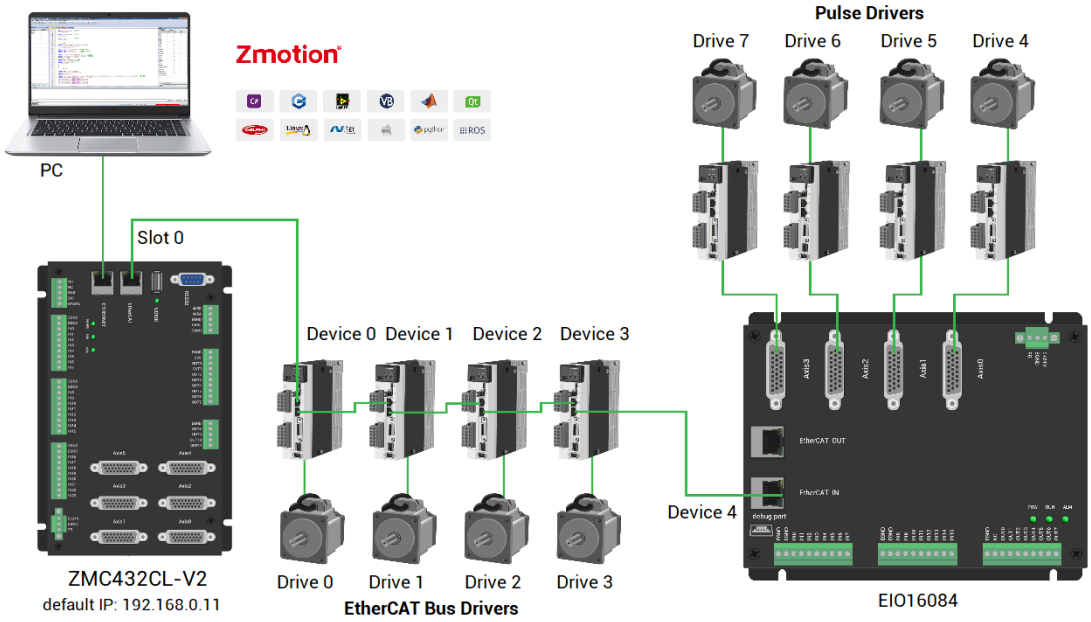
The EIO expansion modules and ZMIO310-ECAT are expansion modules used to expand digital IO / analog IO / pulse axis by EtherCAT.

→ Wiring Reference:

During wiring, use a “shielded category 5e (Cat 5e) twisted pair cable" to connect the controller's EtherCAT port to the EtherCAT IN port of the slave expansion device. Additionally, more slave devices can be connected, also, connect the EtherCAT OUT port of this expansion = to the EtherCAT IN port of the slave device to achieve multi-level expansion.

After completing the wiring for each EIO expansion module, no secondary development is required. Simply map the IO numbers and axis numbers of the expansion modules in the EtherCAT master controller to access them.

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 No. is different from the device No. Only the drive device number on the slot is assigned, and other devices are ignored. While mapping axis No., drive No. will be used.

→ 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.

→ 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

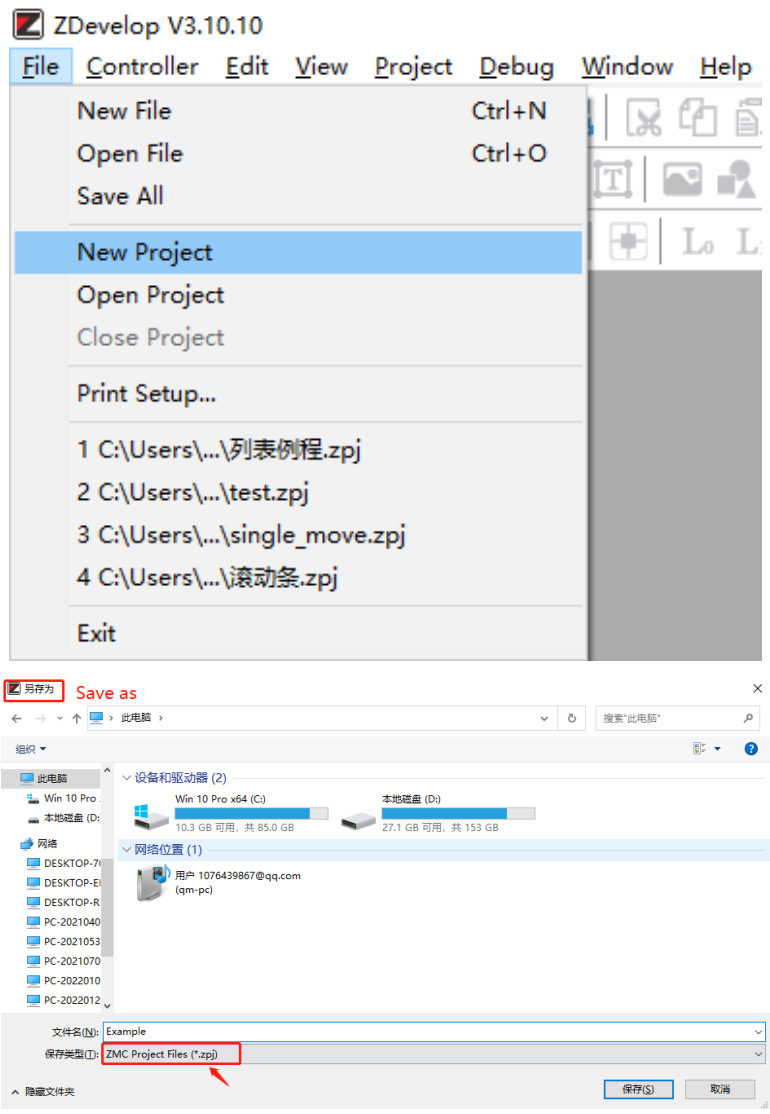
If the first node is EIO16084, and EIO16084 is connected to drive, then driver 0 here is the first pulse driver connected to EIO16084, otherwise it is the EtherCAT driver.

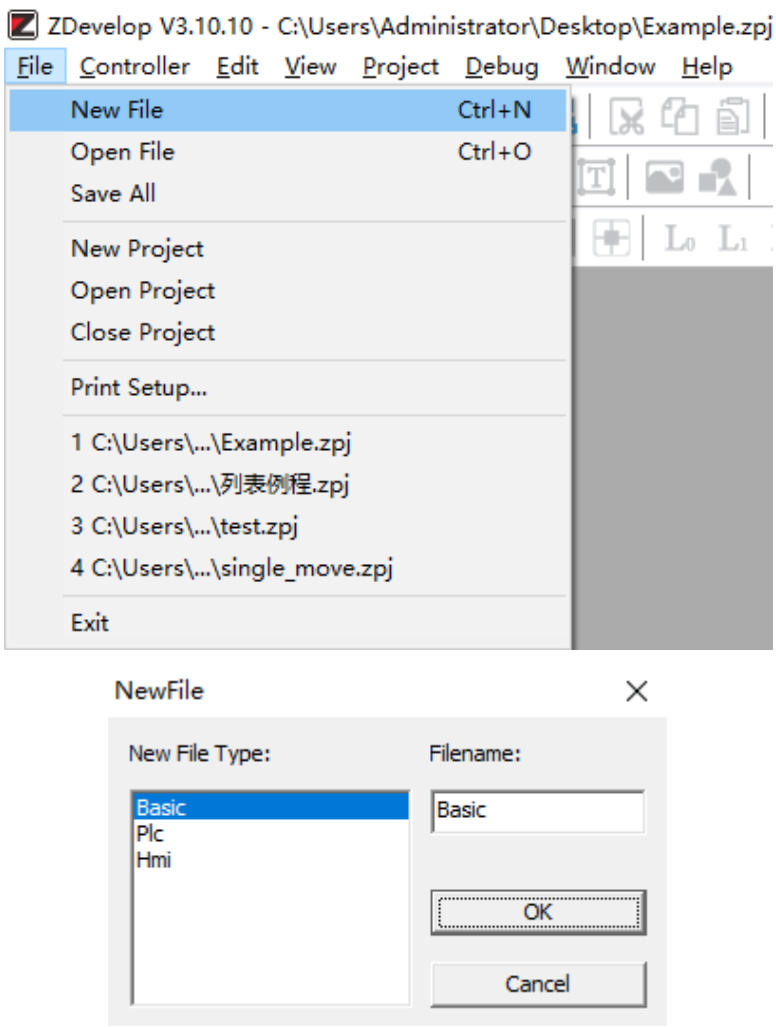
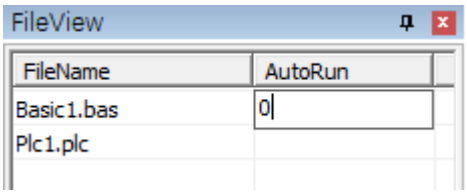
Chapter IV Program & Applications

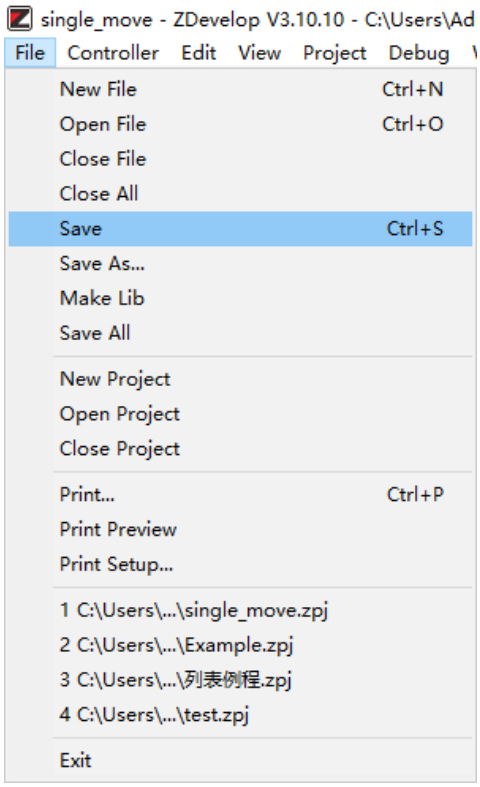
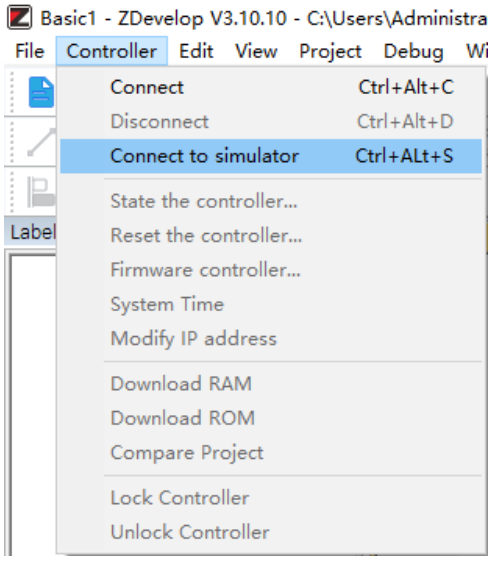
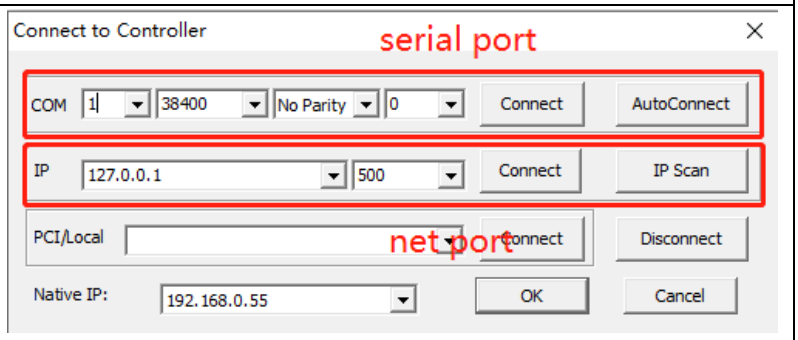
4.1. ZDevelop Software Usage

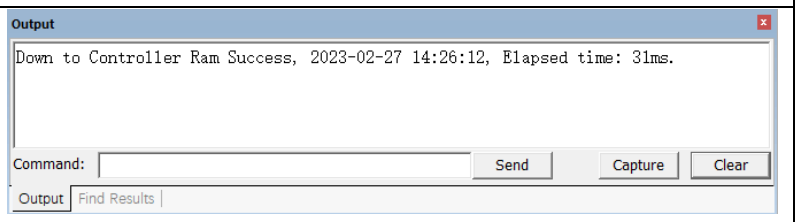
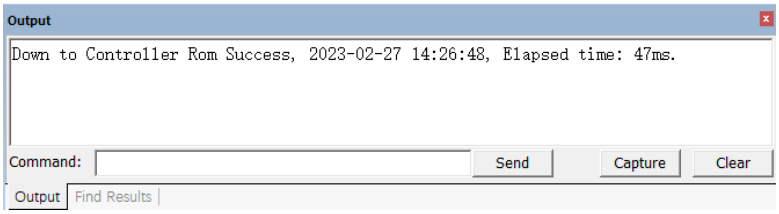
ZDevelop is a PC-side program development, debugging and diagnostic software for the ZMoiton series motion controllers of Zmotion Technology. Through it, users can easily edit and configure the controller program, quickly develop applications, diagnose system operating parameters in real time, and watch the motion controller. The running program is debugged in real time and supports Chinese and English bilingual environments.

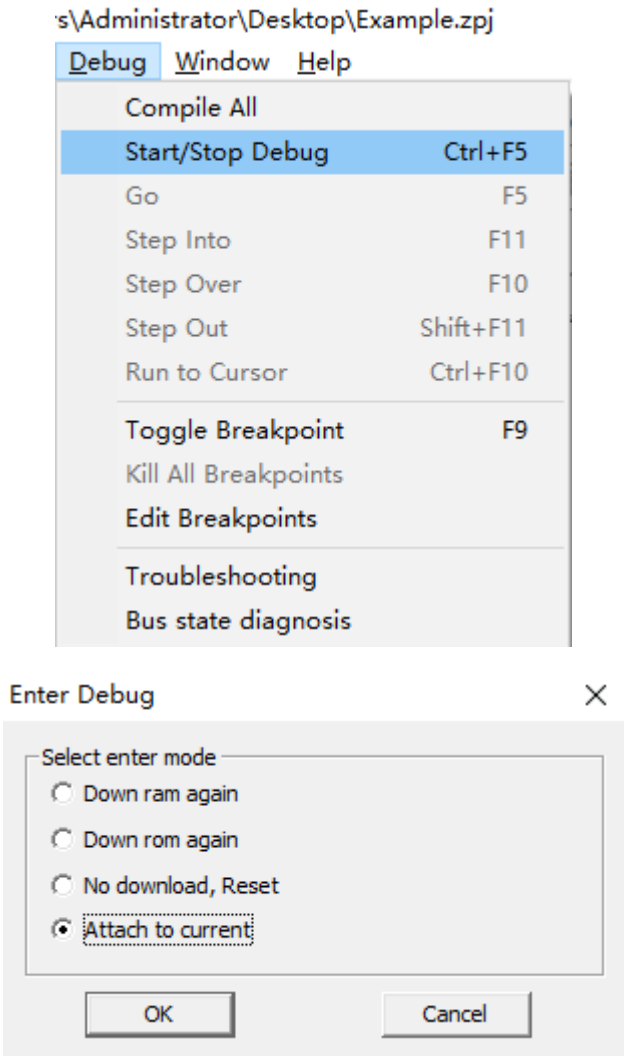
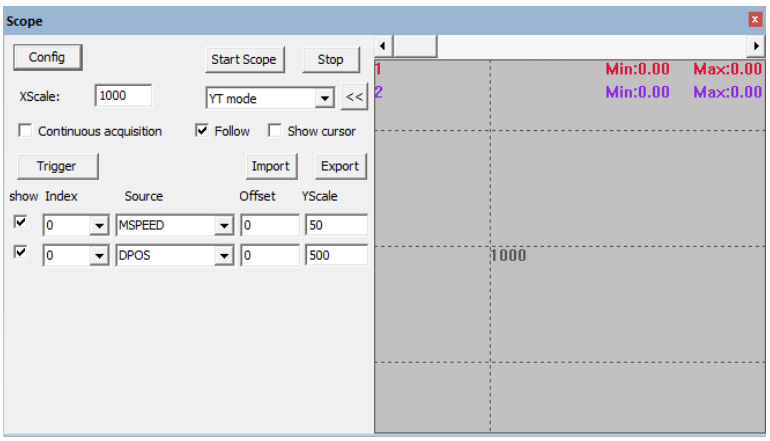
ZBasic, ZPLC and ZHMI can run multi-tasks, and ZBasic can run multi-tasks, and can be mixed with ZPLC and ZHMI.

Step	Operations	Display Interface
1	Open ZDevelop, click "File" – "New Project", Save as window will pop up, then enter file name, save the project file with suffix "zpj.".	

2	Click “File” – “New File”, select file type to build, here select Basic, click “OK”.	
3	Double click “AutoRun”, enter task number 0.	

4	Edit the program in program editing window, click "save", new built basic file will be saved under "zpj." project automatically. "Save all" means all files under this project will be saved.	
5	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”.	
6	Click “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. RAM: it will not save when power off. ROM: it will save data when power off, and when the program is connected to controller again, running according to task number.	 

7	Click “Debug” – “Start/Stop Debug” to call “Task” and “Watch” window, because it was downloaded before, here select “Attach the current".	
8	Click “View” – “Scope” to open oscilloscope.	

Note:

- 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.

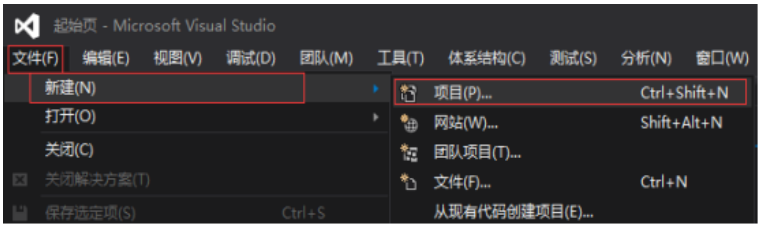
- If no task number is set for the files in the entire project, when downloading to the controller, the system prompts the following message WARN: no program set autorun

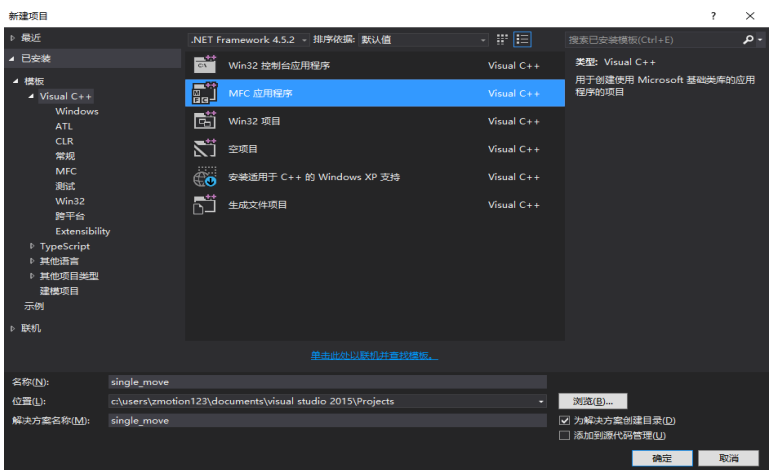
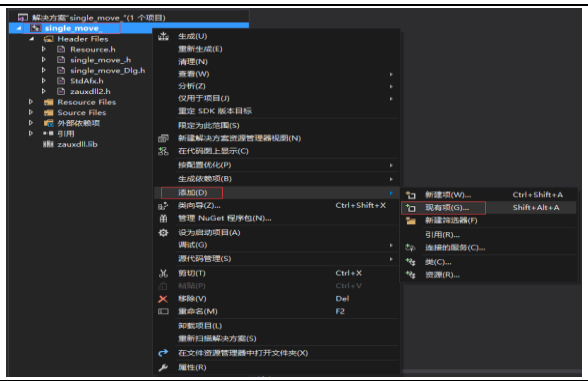
4.2. PC Upper-Computer Program Application

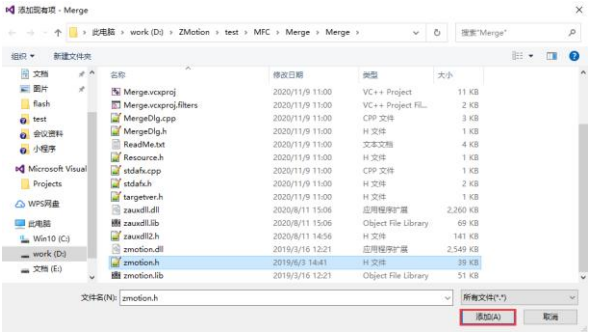
The controller supports development under various operating systems such as windows, linux, Mac, Android, and wince, and provides dll libraries in various environments such as vc, c#, vb.net, and labview, as shown in the figure below. PC software programming refers to "ZMotion PC Function Library Programming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

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,	1) Right-click the header file first, and then select: "Add" → "Existing	

<table>
<tr><td rowspan="2"></td><td rowspan="2">zmotion.lib
Related header files:
zauxdll2.h,
zmotion.h</td><td>Item".</td><td></td></tr>
<tr><td>2) Add static libraries and related header files in sequence in the pop-up window.</td><td>  </td></tr>
<tr><td>7</td><td>Declare the relevant header files and define the controller connection handle, so far the project is newly created.</td><td colspan="2">

```
single_move_Dlg.cpp
single_move_                                   (全局范围)
// 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
```

</td></tr>
</table>

Chapter V Run 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.

5.1. 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 explosive gases or articles	No

	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

5.2. Common Problems

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	1. Check whether the limit sensor is working normally,

invalid.	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.