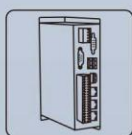


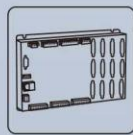
# Vertical Motion Controller

## ZMC432M

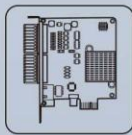
This manual is for ZMC432M, ZMC432ML5, ZMC432ML24.



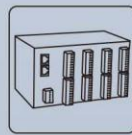
Vision Motion  
Controller



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.	

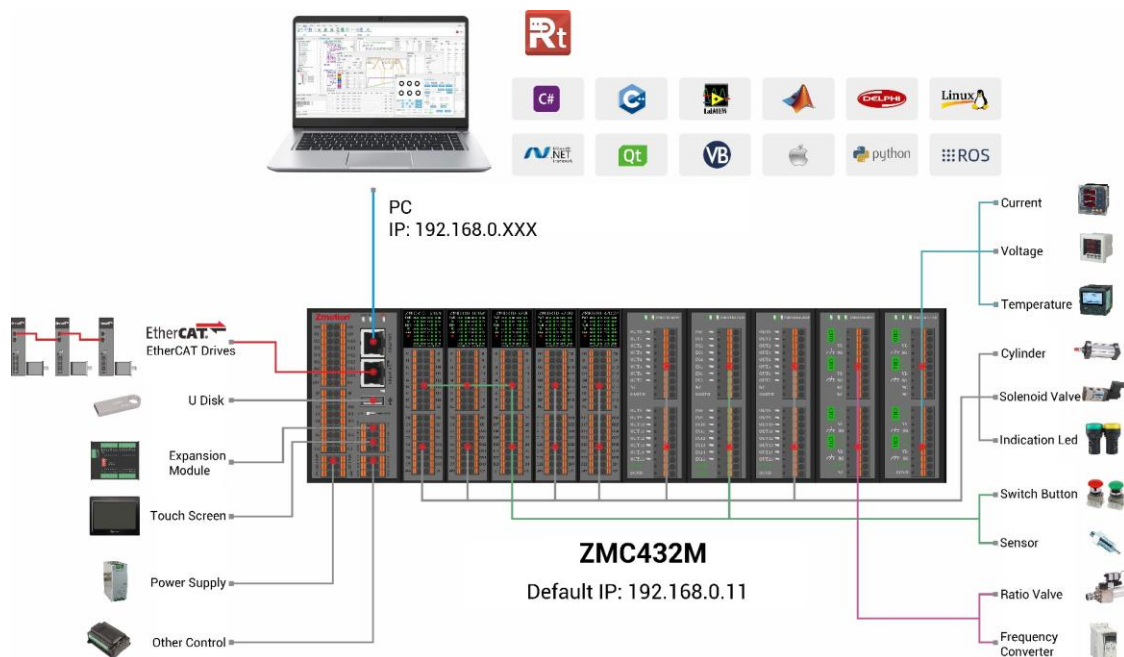
# Content

Chapter I Production Information.....	4
1.1. Product Information .....	4
1.2. Interface Introduction.....	5
1.3. Specification Model.....	6
1.4. Nameplate & Model .....	8
1.5. Environment Requirements .....	9
1.6. Hardware Installment .....	10
Chapter II Hardware Interface .....	11
2.1. Power Input .....	11
2.2. IN: General Digital Inputs .....	11
2.3. OUT: General Digital Output.....	15
2.4. AD / DA Analog Interface.....	19
2.5. Communication Interfaces: CAN / RS485 / RS232.....	21
2.6. U Disk.....	24
2.7. RUN / STOP Dial-Code Switch .....	25
2.8. ETHERNET .....	26
2.9. EtherCAT Bus Interface .....	27
Chapter III Local Expansion.....	30
3.1. Local Configuration .....	30
3.2. Configuration Function.....	31
Chapter IV Expansion Module .....	36
4.1. CAN Bus Expansion.....	36
4.2. EtherCAT Bus Expansion .....	40
Chapter V Programming.....	44
5.1. Program in RTSys Software .....	44
5.2. Upgrade Controller Frimware .....	49
5.3. Program in Host-Computer by PC Languages .....	50
Chapter VI Operation and Maintain.....	53
6.1. Regular Inspection and Maintenance .....	53
6.2. Common Problems & Solutions .....	54

# Chapter I Production Information

## 1.1. Product Information

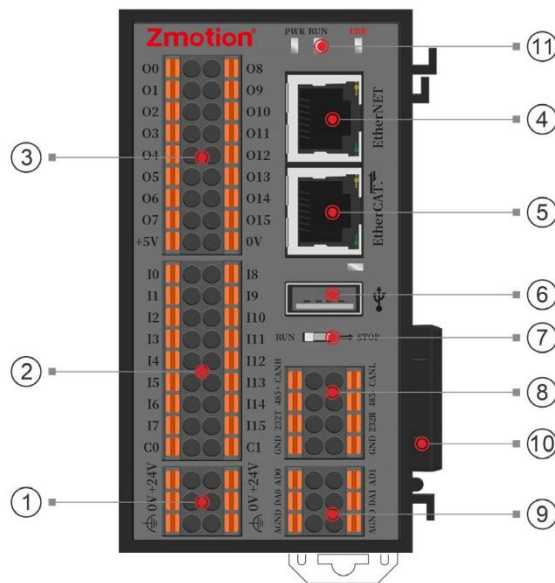
ZMC432M **high-performance multi-axis** motion controller is a stand-alone motion controller that is compatible EtherCAT with pulse type. The controller itself supports 32 axes at most to achieve complex continuous trajectory control requirements.



- ✚ It has EtherCAT bus axis, single-ended pulse axis, and differential encoder interface.
- ✚ **Basic Motion Control Functions:** point motion, electronic cam, linear interpolation, circular interpolation, continuous interpolation, SCARA, etc.
- ✚ **Special Motion Control Function:** HW hardware comparison output, high-speed latch, PWM, etc., and for special models, it supports 5V/ 24V laser specialized PWM output (ZMC432ML series).
- ✚ UDISK can be used to write and read files, and update the program, which is convenient to remote support and maintain.
- ✚ It can be programmed in multi-file and multi-task by BASIC, also, PC program and controller inner controller can work at the same time.

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



No.	Interface	Description
①	Power	Connect to 24V DC power supply
②	General Digital IN	NPN / PNP type, depend on public end type, IN0-15, it supports single-ended encoder, latch.
③	General Digital OUT	NPN type, OUT0-15, it supports single-ended pulse, HW, PWM functions
④	EtherNET	Connect to host computer, use MODBUS_RTU protocol, which can be expanded by interchanger.
⑤	EtherCAT	Connect to EtherCAT drive and EtherCAT expansion module.
⑥	UDISK	Connect to UDISK to update the program, import and export data.
⑦	RUN / STOP	Switch run / stop modes.

⑧	CAN	Connect to CAN expansion modules to expand more resources.
	RS485	Connect to host computer, use MODBUS_RTU protocol.
	RS232	
⑨	Analog DA	Single-ended type, 12bit, 0-10V
⑩	Local Slave Expansion Interface	Connect to expansion submodules, please note it doesn't support "hot-swap".
⑪	Controller Status Led	PWR led: it is ON when power is connected.
		RUN led: it is ON when it runs normally.
		ERR led: it is ON when it runs abnormally.

**--Notes--**

- If no need to connect ZMIO310 submodules, please set ZMIO\_OFFSET command as negative value / make it exceed IO starting No. range (must be a multiple of 8), for example, ZMIO\_OFFSET = -8 (when the ZMC432M doesn't bring with IO expansion module), ZMIO\_OFFSET = 5000 (the valid range is 0-4095 when it connects to IO expansion module), otherwise, it will report the error of code "201".
- If not to connect with expansion sub-module, the system appears 201 error code, you could add "ZMIO310-OFFSET=-8" at the first line of the program, which means without extended IO.

**1.3. Specification Model**

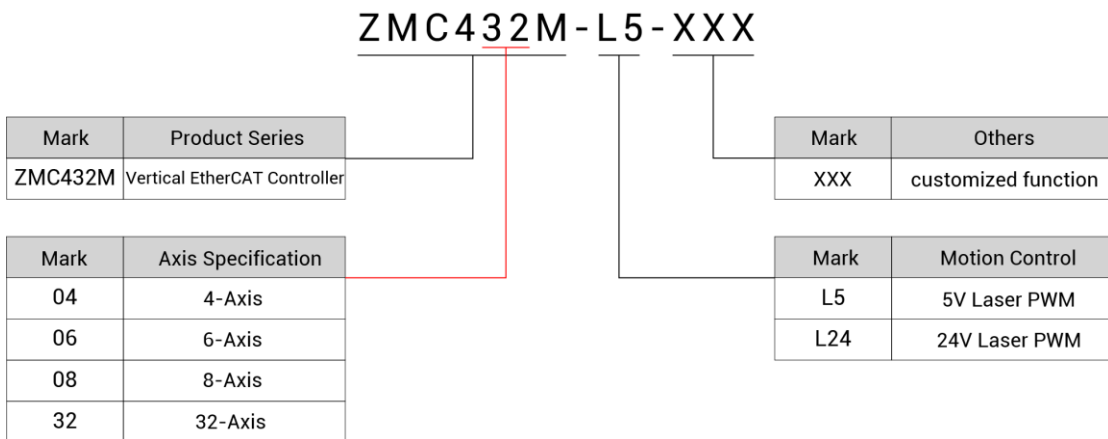
Model	ZMC432M	ZMC432ML5	ZMC432ML24
Basic Axes	32		
All Axes	32 (basic axis + virtual axis)		
EtherCAT	√		
OUT (single-ended pulse axis)	8		
IN (single-ended encoder axis)	4		

Digital IN	16		
Digital OP	16		
Max Digital IN to be Expanded	≤4096		
Max Digital OP to be Expanded	≤4096		
AD Input	2 (0-10V, 12bit)		
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	4		
Hardware Comparison Output HW	4		
General PWM	4 (OUT0-3)	2 (OUT2-3)	2 (OUT2-3)
Laser PWM	0	2 (5V, OUT0-1)	2 (24V, OUT0-1)
Point Motion	√		
Electronic Cam			
Linear Interpolation			
Circular Interpolation			
Continuous Interpolation			
SCARA Robotic Arm			

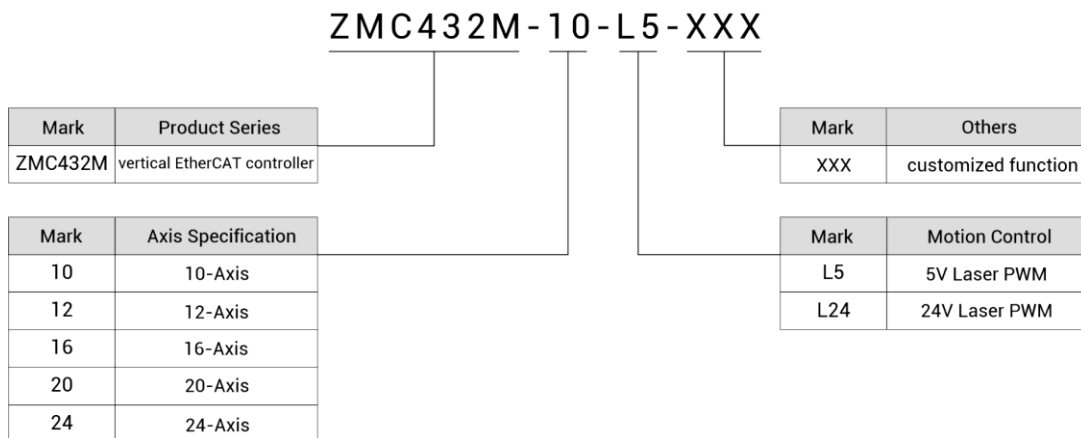
Program Space	128MByte
Power Down Storage	√
Dimensions	95*93*46

## 1.4. Nameplate & Model

### ■ Nameplate Information (32-axis & <10-axis specifications)



### ■ Nameplate Information (10-24 axes specifications)



### ■ Product Models

No.	Models	Description
1	ZMC432M	Hardware comparison output, <b>not support</b> laser PWM output.
2	ZMC432ML5	Hardware comparison output, <b>support 5V laser PWM output.</b>

3	ZMC432ML24	Hardware comparison output, support 24V laser PWM output.
---	------------	---

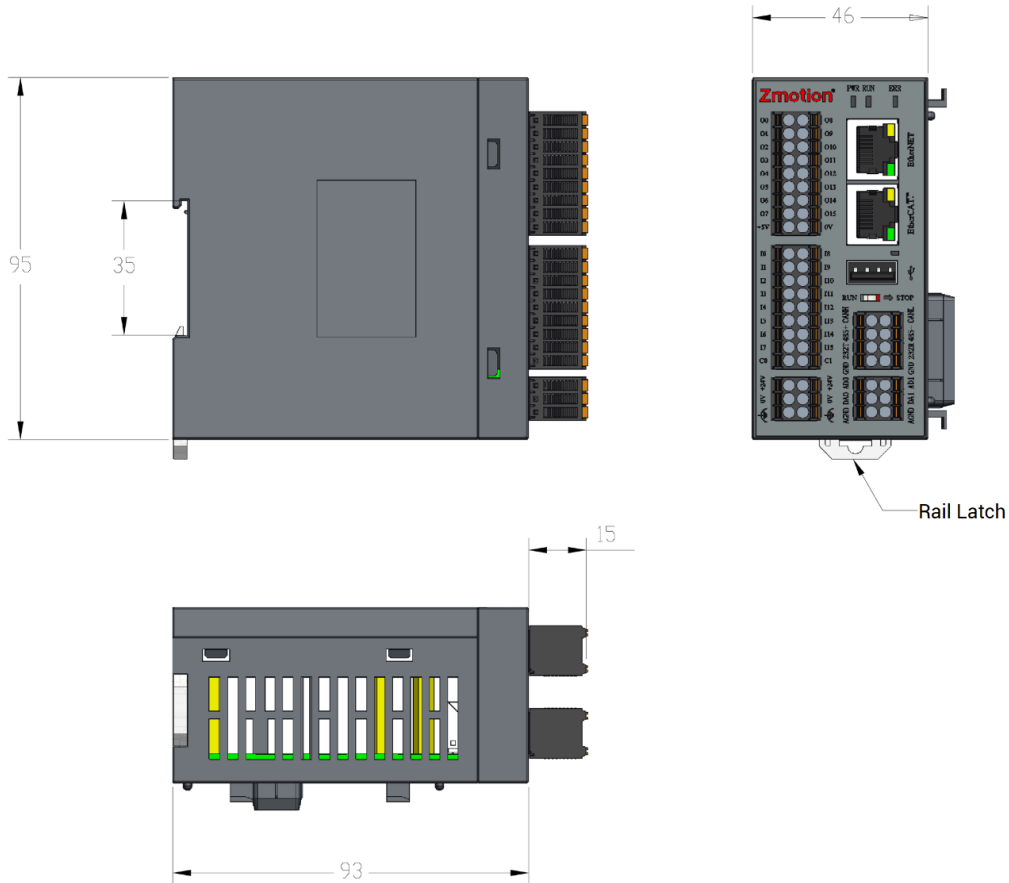
Note: ZMC432M doesn't support laser PWM output, if you need this function, please select corresponding models.

## 1.5. Environment Requirements

Item		Parameters
Work Temperature		-10°C-55°C
Work relative Humidity		10%-95% non-condensing
Storage Temperature		-40°C ~ 80°C (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

## 1.6. Hardware Installment

ZMC432M motion controller installment size (unit: mm):



### How to Install:

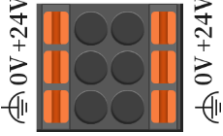

Step 1: please use standard 35m DIN rail.

Step 2: open controller rail latch, then embed the controller into DIN rail.

Step 3: press controller rail latch, and fix controller on the DIN rail.

## Chapter II Hardware Interface

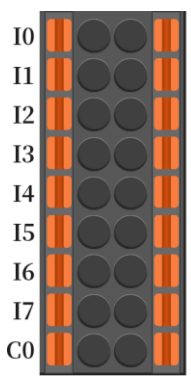
### 2.1. Power Input

Terminal	Name	Function	Description
	+24V	24V main power +	Isolation Power
	0V	24V main power -	
		Protection	

#### → Specification

Item	Description
Voltage	DC24V±5%
Max Power	10W
Anti-reverse connection	YES
Overcurrent Protection	YES
Isolation Power	YES
Cable Type	Recommend "1.0 mm <sup>2</sup> copper conductor cable"

### 2.2. IN: General Digital Inputs

Terminal	Name	Type	Function 1	Function 2	Function 3	
	I0	high-speed input NPN / PNP	Input 0	High Speed Latch	/	
	I1		Input 1		/	
	I2		Input 2		/	
	I3		Input 3		/	
	I4		Input 4	/	EA0	
	I5		Input 5	/	EB0	
	I6		Input 6	/	EZ0	
	I7		Input 7	/	EA1	
	C0		C0	COM0: IN (0-7) public end to determine the input type		
	I8		I8	high-	Input 8	/

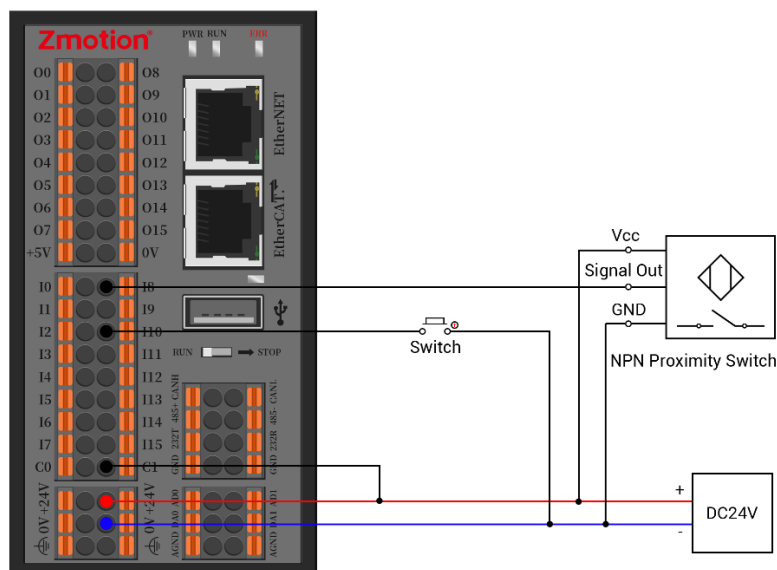
	I9	speed input NPN / PNP	Input 9	/	EZ1
	I10		Input 10	/	EA2
	I11		Input 11	/	EB2
	I12		Input 12	/	EZ1
	I13		Input 13	/	EA3
	I14		Input 14	/	EB3
	I15		Input 15	/	EZ3
C1	COM0: IN (8-15) public end to determine input type				

→ Specification

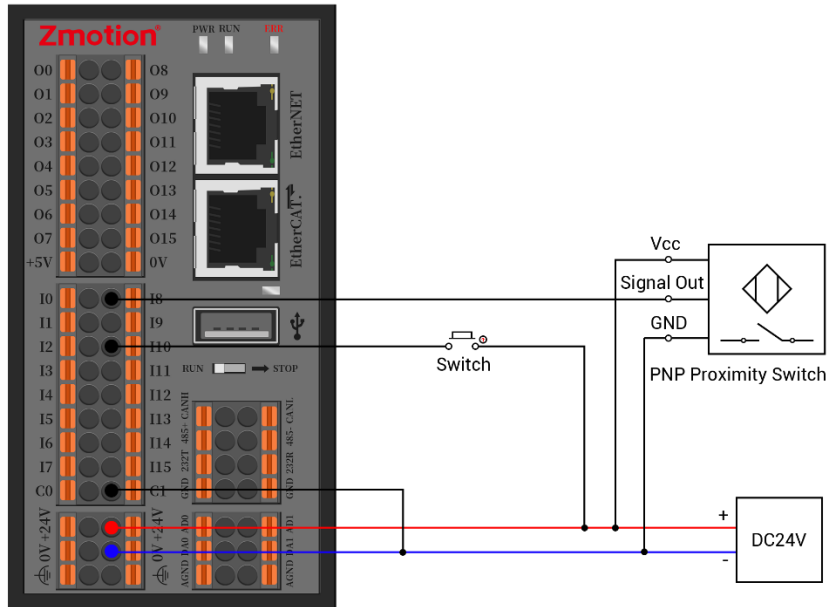
Item	High-Speed Input (IN0-15)	
Input mode	NPN, the input is triggered by <b>low-electric level</b>	PNP, the input is triggered by <b>high-electric level</b>
Frequency	≤400kHz	≤400kHz
Impedance	4.7KΩ	4.7KΩ
Voltage	0V/24V	0V/24V
Communication Isolation	√	√

→ Wiring Reference

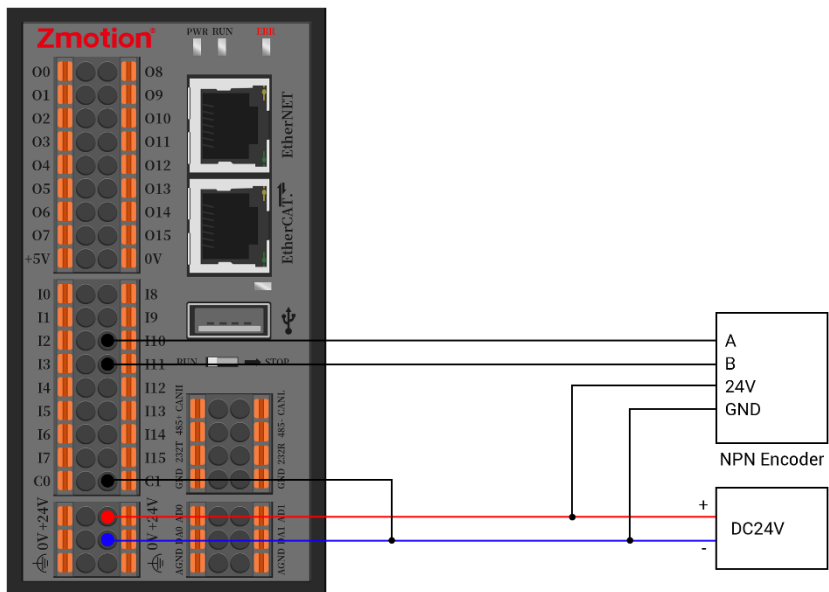
NPN Wiring:

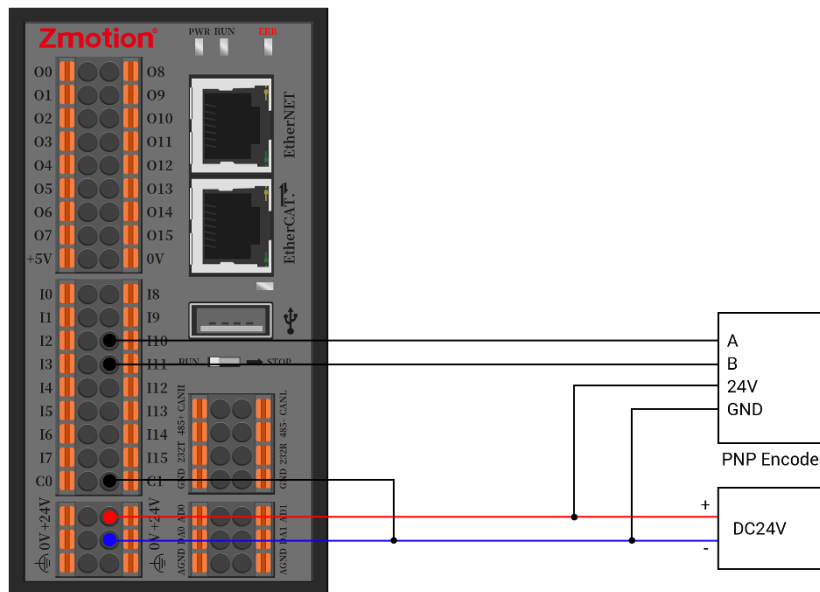


**PNP Wiring:**



**NPN Single-Ended Encoder Wiring:**



**PNP 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 30cm 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".

## 2.3. OUT: General Digital Output

Terminal	Name	Type	Functions			
			1	2	3	4
<p>Terminal block diagram showing 16 digital output terminals (00-15) and power terminals (+5V, 0V). The terminals are arranged in a 4x4 grid. The left column contains terminals 00, 01, 02, 03, 04, 05, 06, 07, +5V. The right column contains terminals 08, 09, 010, 011, 012, 013, 014, 015, 0V.</p>	00	NPN, high- speed outputs, max is 500mA.	output 0	PWM0	HW_0	PUL0
	01		output 1	PWM1	HW_1	DIR0
	02		output 2	PWM2	HW_2	PUL1
	03		output 3	PWM3	HW_3	DIR1
	04		output 4	/	/	PUL2
	05		output 5	/	/	DIR2
	06		output 6	/	/	PUL3
	07		output 7	/	/	DIR3
	08		output 8	/	/	PUL4
	09		output 9	/	/	DIR4
	010		output 10	/	/	PUL5
	011		output 11	/	/	DIR5
	012		output 12	/	/	PUL6
	013		output 13	/	/	DIR6
	014		output 14	/	/	PUL7
	015		output 15	/	/	DIR7
+5V	5V power output +, max is 300mA					
0V	5V power output -, digital output public end					
<b>Notes:</b>						

- ◇ For PWM0 and PWM1, there are 5V / 24V "push-pull type" PWM function specified for laser. if you need this, please attention the model.
- ◇ **OUT0-15** support axis 0-7 function, when ATYPE = 1, it is single-ended pulse axis function, when ATYPE = 0, it is normal output.

### → Specification (ZMC432M)

Item	High Speed Output (OUT0-15)
Output mode	NPN Type
Frequency	≤400kHz
Voltage	≤24V
Output Current	≤500mA
Overcurrent protection	√
Communication Isolation	√

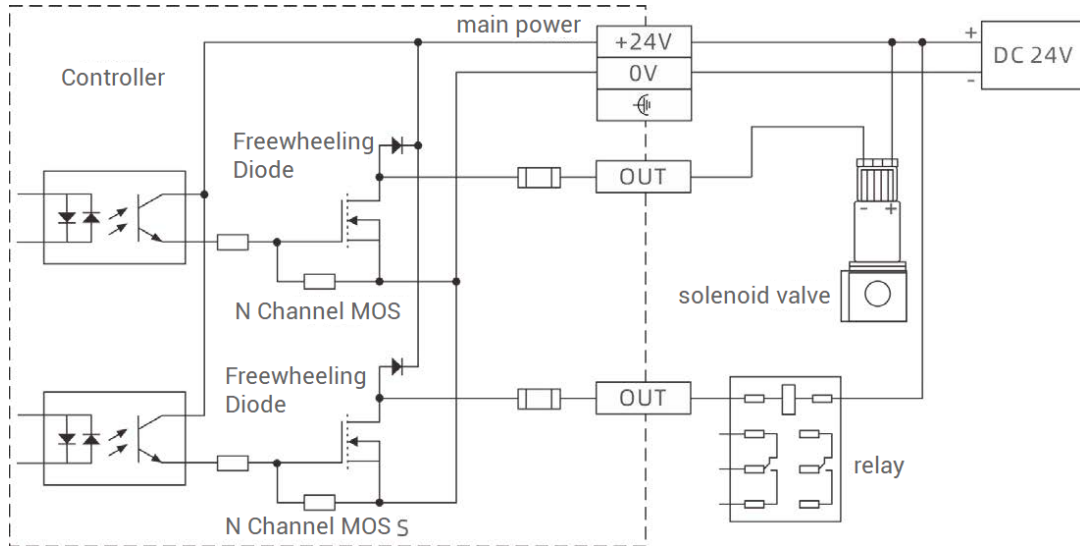
### → Specification (ZMC432ML)

Item	Laser PWM Output (OUT0-1)	High Speed Output (OUT2-15)
Output mode	Push-pull type	NPN type
Frequency	≤1MHz	≤400kHz
Voltage	≤5V / ≤24V (optional)	≤24V
Output Current	≤500mA	≤500mA
Overcurrent protection	√	√
Communication Isolation	√	√

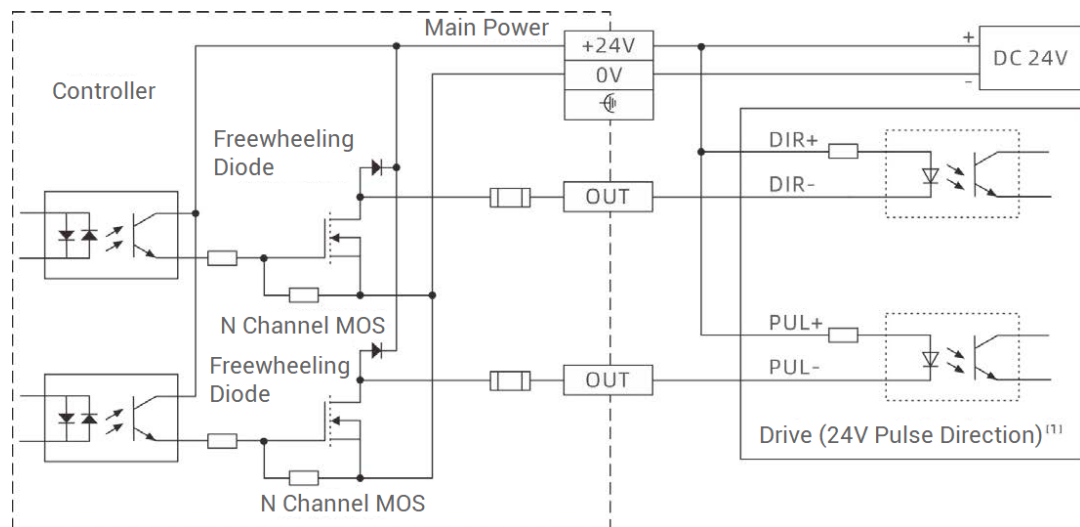
Note: L5/L24 models (laser PWM & push-pull output) are dedicated for laser and PWM signals, the power is small, no freewheeling protection, so it doesn't support inductive load – don't directly drive inductive loads, like, relay, solenoid valve, motor, etc.

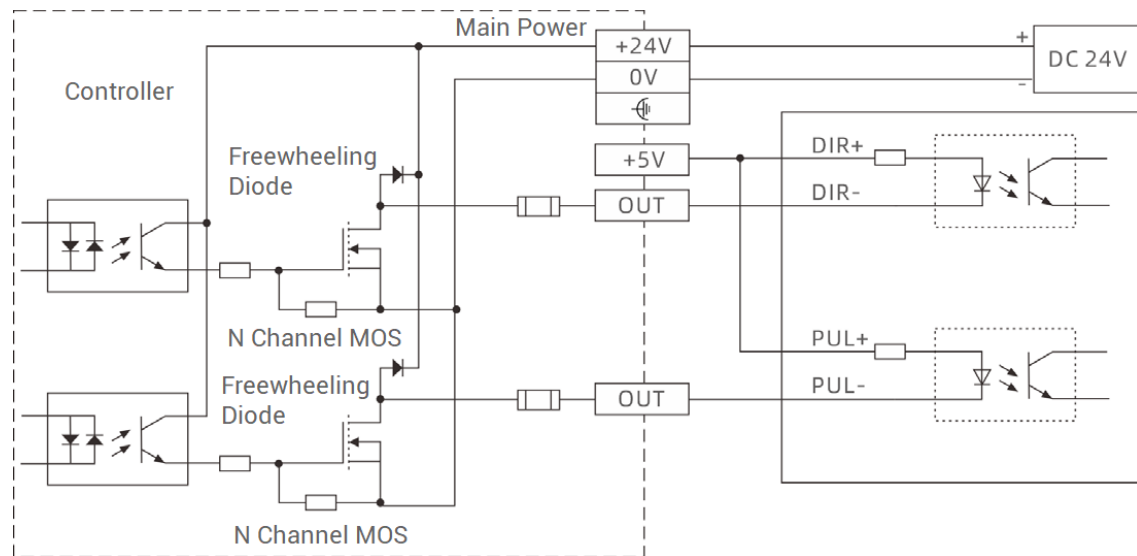
→ **Wiring Reference**

**General Output Wiring:**



**Pulse Axis Wiring (24V):**



**Pulse Axis Wiring (5V):**

- **Notes**

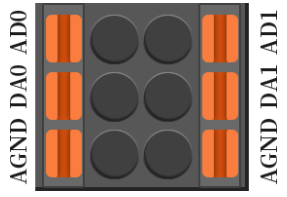
- The wiring for the digital output is as shown in the diagram above. The external load can be a relay, valve solenoid etc., but it is important to ensure that the signal specifications match.
- For 5V pulse direction interface, please connect PUL+ and DIR+ to E5V interface.
- It is recommended to use 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 30cm 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\_PWSIWTC2**".
- (5) When it is used as pulse axis, the usage method is same as AXIS. please refer to "BASIC Programming Manual".
- (6) Above commands and other commands, please refer to "BASIC Programming Manual".

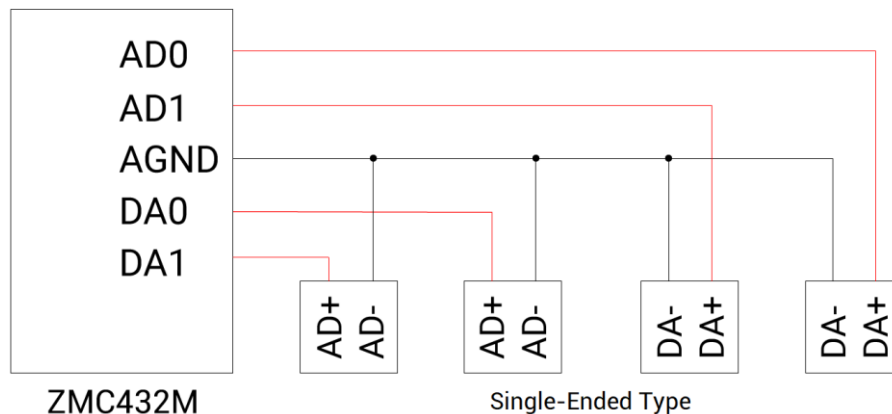
## 2.4. AD / DA Analog Interface

Terminal	Name	Function
	AD0	Analog input terminal: AIN(0)
	AD1	Analog input terminal: AIN (1)
	DA0	Analog output terminal: AOUT(0)
	DA1	Analog output terminal: AOUT(1)
	AGND	Analog public end

### → Specification

Item	AD (0-1)	DA (0-1)
Resolution	12-bit	12-bit
Data range	0-4095	0-4095
Signal range	0-10V output	0-10V output
Data refresh ratio	1KHz (refresh ratio)	1KHz (refresh ratio)
Input impedance / output load	>40KΩ (input impedance)	>10KΩ (load requirement)
Signal Type	Single-ended	Single-ended
Cable Length	Recommend <5m	Recommend <5m

## → Wiring Reference



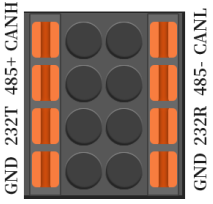
### ● Notes:

- 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. Communication Interfaces: CAN / RS485 / RS232

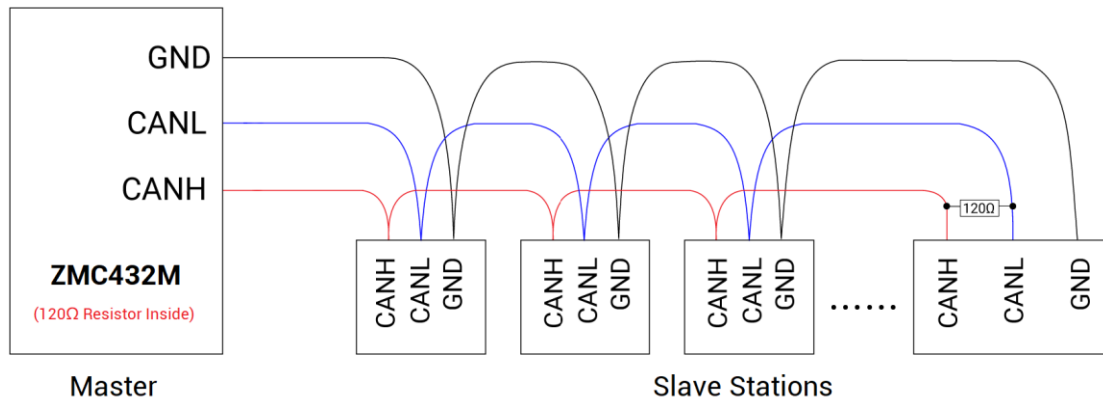
Terminal	Name	Function
 <p style="color: red; text-align: center;">Isolated Communication</p>	CANH	CAN communication signal cable side H
	CANL	CAN communication signal cable side L
	485+ / A	RS485 communication signal cable + / side A
	485- / B	RS485 communication signal cable - / side B
	232T	RS232 communication signal side TX
	232R	RS232 communication signal side RX
	GND	Communication public end

### → Specification

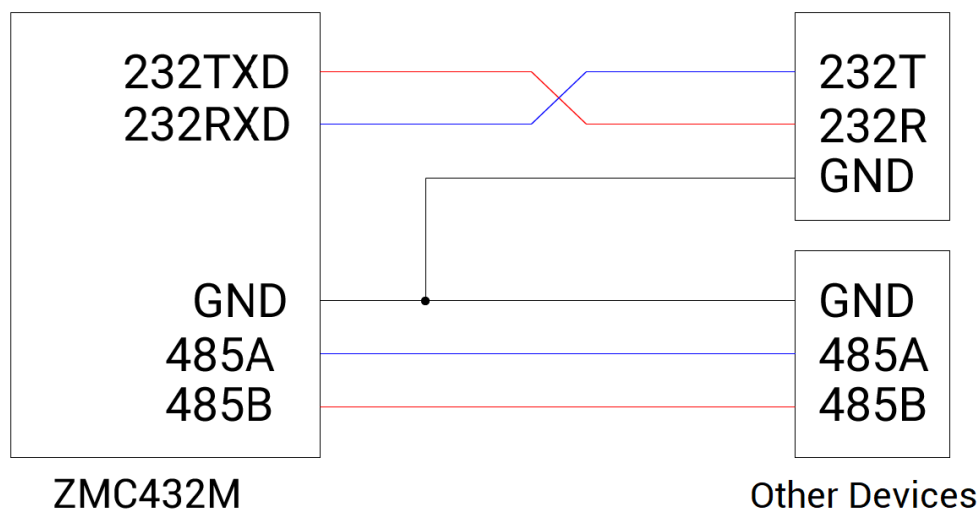
Item	CAN	RS485 (port1)	RS232 (port0)
Communication Rate	≤1Mbps	≤115200bps	≤115200bps
Terminal Resistor	120Ω (inside in controller)	/	/
Topological Structure	Daisy Chain Topology		One to One
The number of nodes can be extended	≤16	≤127	1
Wiring Length	Recommend <30m (500kbps)	Recommend <30m	Recommend <5m
Communication Isolation	√		

## → Wiring Reference

- CAN Wiring:



- RS485 / RS232 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.
- The wiring of RS232 is also shown above, while receiving and sending the signals, please cross-wire them. And when connecting to PC, please use dual-female-head cross cable.

- Please connect a 120Ω terminal resistor in parallel to each end of the CAN bus for matching the circuit impedance and ensuring communication stability.
- For RS485 (point to point), no need the 120Ω terminal resistor, but when there are many nodes on the bus, connect resistors on each side in parallel to promote communication stability.
- Please be sure to connect the public ends of each communication node to prevent CAN / RS485 / RS232 chips 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:

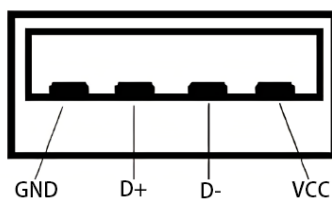
1) Configure controller CAN master station:

- 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 ["4.1 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/R232:](#)
- 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. U Disk

The ZMC43M 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".

## 2.7. RUN / STOP Dial-Code Switch

Terminal	Name	Description
	RUN	Running mode, controller is powered on and runs normally.
	STOP	Stop mode, controller stops running, and its IP resume factor address.

**Note:** while switching these two modes, please do it before controller powered on.

- RUN mode

"RUN" mode is factory default mode, the controller can be set and used normally.

- STOP mode

When in "STOP" mode, ETHERNET IP will resume factory default value temporarily,

the application program in ROM will not run. At this time, you can **modify ethernet IP, update program, and take effect immediately.**

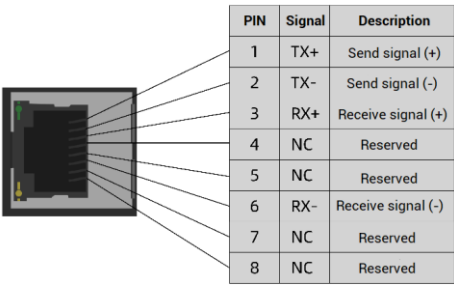
**✚ Modes Switch Description:**

**\*keep the configuration:** after switched to "RUN", the controller will automatically resume as "user define state" (IP & application program are not changed) before "STOP" mode.

**\*new configuration:** the new parameter / new program will take effect under RUN mode, please note new IP can't be conflict with controller default IO, otherwise, it can't be saved after switching RUN mode.

## 2.8. ETHERNET

### → Specification

PIN Definition	Item	Description																											
 <table border="1" data-bbox="488 1249 715 1532"> <thead> <tr> <th>PIN</th> <th>Signal</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>TX+</td> <td>Send signal (+)</td> </tr> <tr> <td>2</td> <td>TX-</td> <td>Send signal (-)</td> </tr> <tr> <td>3</td> <td>RX+</td> <td>Receive signal (+)</td> </tr> <tr> <td>4</td> <td>NC</td> <td>Reserved</td> </tr> <tr> <td>5</td> <td>NC</td> <td>Reserved</td> </tr> <tr> <td>6</td> <td>RX-</td> <td>Receive signal (-)</td> </tr> <tr> <td>7</td> <td>NC</td> <td>Reserved</td> </tr> <tr> <td>8</td> <td>NC</td> <td>Reserved</td> </tr> </tbody> </table>	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	Communication Protocol	MODBUS_TCP
	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																											
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 10M communication	Now it is sending & receiving data
Yellow	Build 100M 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.

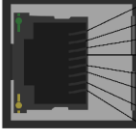
→ **Usage Methods**

- (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	MODBUS_TCP

	<b>PIN</b>	<b>Signal</b>	<b>Description</b>	<b>Protocol</b>	
	1	TX+	Send signal (+)	<b>Communication Velocity</b>	100Mbps
	2	TX-	Send signal (-)		
	3	RX+	Receive signal (+)	<b>Refresh Ratio</b>	Max: 500us
	4	NC	Reserved	<b>Communication Cable</b>	Shielded Cat 5e Cable
	5	NC	Reserved		
	6	RX-	Receive signal (-)	<b>Cable Length</b>	Recommend <50m
	7	NC	Reserved		
8	NC	Reserved			

## → 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 10M communication	Now it is sending & receiving data
Yellow	Build 100M 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 ["4.2 EtherCAT Expansion" – "Resource Mapping"](#).
- 3) Use SLOT\_START to open bus, SLOT\_STOP to stop.
- 4) After completed, configure and operate local pulse axis.

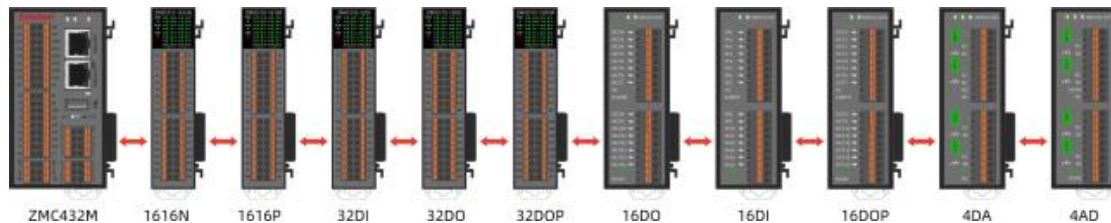
**(3) How to connect to expansion module by EtherCAT:**

- 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 "4.2 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".

## Chapter III Local Expansion

### 3.1. Local Configuration

Through ZMC432M local slave expansion interface, it can connect to ZMIO310 series submodules directly.



#### ■ How to Install:

Step 1: release all module clearance latches.

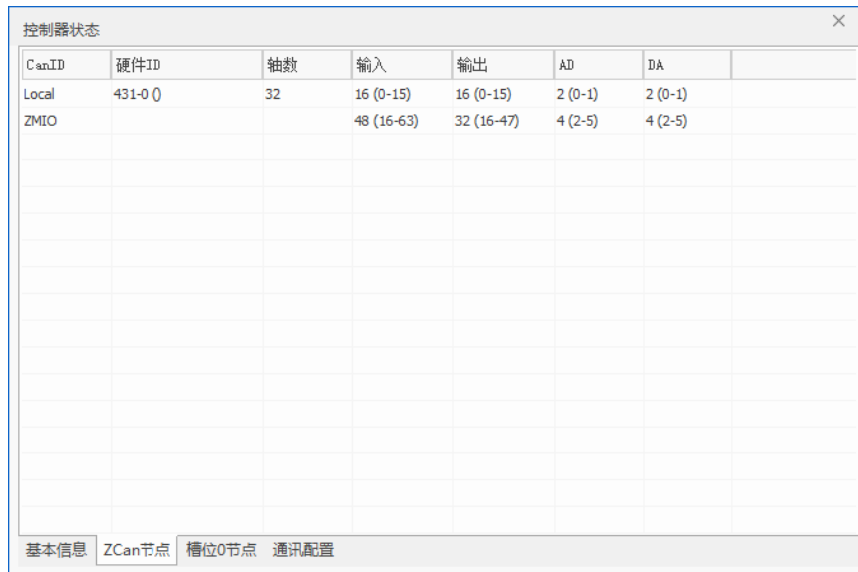
Step 2: align and insert the local expansion front-stage interface of the expansion sub-module into the local expansion slave interface of the ZMC432M controller (or another expansion sub-module).

Step 3: engage all module clearance latches.

**Note:** for ZMIO310 series expansion submodules, please refer to "ZMIO310 Expansion Module User Manual".

#### ■ Expansion Example:

For example: the ZMC432M expands 3 input modules (ZMIO310-16DI), 2 output modules (ZMIO310-16DO or ZMIO310-16DOP), 1 AD module (ZMIO310-4AD) and 1 DA module (ZMIO310-4DA), no need to do program initialization, install according to above steps, then power on them. The controller state will be:



CanID	硬件ID	轴数	输入	输出	AD	DA
Local	431-0 0	32	16 (0-15)	16 (0-15)	2 (0-1)	2 (0-1)
ZMIO			48 (16-63)	32 (16-47)	4 (2-5)	4 (2-5)

The address of the ZMIO310 expansion IO starts from 16 by default, for AD and DA, starting from 2. The same type modules addresses will be numbered automatically. If you need to offset the starting address, please refer to below:

Type	Command	View	Example
Input	IN	IN View	IN (16) – IN (63)
Output	OP	OP View	OP (16) – OP (47)
AD	AIN	AD/DA View	AIN (2) – AIN (5)
DA	AOUT	AD/DA View	AOUT (2) – AOUT (5)

**Note:** ZMIO expansion AD and DA use standard range 0-10V by default, if you need to change it, please contact with us.

## 3.2. Configuration Function

You can configure ZMIO expansion, specifically, configure IO and analog.

### → Configure ZMIO IO Offset

<b>Function Description</b>	Used to offset the IO address of ZMC432M itself ZMIO310 expansion		
<b>Usage Syntax</b>	ZMIO_OFFSET=value		
<b>Parameter List</b>	value	IO starting address	Default is 32 (the multiple of 8)
<b>Example</b>	ZMIO_OFFSET=48 'offset it to 48		

**Notes:**

- The value only can be the multiple of 8.
- If no need to connect ZMIO310 submodules, please set ZMIO\_OFFSET command as negative value / make it exceed IO starting No. range (must be a multiple of 8), for example, ZMIO\_OFFSET = -8 (when the ZMC432M doesn't bring with IO expansion module), ZMIO\_OFFSET = 5000 (the valid range is 0-4095 when it connects to IO expansion module), otherwise, it will report the error of code "201".

--controller state before configuration--

CanID	硬件ID	轴数	输入	输出	AD	DA
Local	431-0 0	32	16 (0-15)	16 (0-15)	2 (0-1)	2 (0-1)
ZMIO			48 (16-63)	32 (16-47)	4 (2-5)	4 (2-5)

--controller state after configuration--

CanID	硬件ID	轴数	输入	输出	AD	DA
Local	431-0 0	32	16 (0-15)	16 (0-15)	2 (0-1)	2 (0-1)
ZMIO			48 (48-95)	32 (48-79)	4 (2-5)	4 (2-5)

→ **Configure ZMIO AIO Offset**

<b>Function Description</b>	Used to offset the AIO address of ZMC432M itself ZMIO310 expansion		
<b>Usage Syntax</b>	ZMAIO_OFFSET=value		
<b>Parameter List</b>	value	AIO stating address	Default is 32
<b>Example</b>	ZMAIO_OFFSET=33 'offset it to 48		

--controller state before configuration--

CanID	硬件ID	轴数	输入	输出	AD	DA
Local	431-0 0	32	16 (0-15)	16 (0-15)	2 (0-1)	2 (0-1)
ZMIO			48 (16-63)	32 (16-47)	4 (2-5)	4 (2-5)

--controller state after configuration--

CanID	硬件ID	轴数	输入	输出	AD	DA
Local	431-0 0	32	16 (0-15)	16 (0-15)	2 (0-1)	2 (0-1)
ZMIO			48 (16-63)	32 (16-47)	4 (33-36)	4 (33-36)

--How to check itself ZMIO expansion situation--

<b>Function Description</b>	Used to check ZMC432M controller itself ZMIO expansion situation.			
<b>Usage Syntax</b>	Syntax 1: var = ZMIO_INFO (sel) Syntax 2: var = ZMIO_INFO (17, node)			
<b>Parameter List</b>	sel	Select functions	Function No.	Information
			10	Max inputs
11			Max outputs	
12			Max AIN	
13			Max AOUT	
16			Modules	
	node	Module No.	Start from 0, one module connected, No. + 1	
<b>Example</b>	?ZMIO_INFO (10) 'check how many inputs at most of ZMIO ?ZMIO_INFO (11) 'check how many outputs at most of ZMIO ?ZMIO_INFO (12) 'check how many AIN at most of ZMIO ?ZMIO_INFO (13) 'check how many AOUT at most of ZMIO ?ZMIO_INFO (16) 'check how many modules of ZMIO ?ZMIO_INFO (17, 0) 'check the type No. of the first module			

Note: for ZMIO310 series expansion submodules, please refer to "ZMIO310 Expansion Module User Manual".

→ **Set / Get AIO Range & Channel States**

<b>Function Description</b>	Used to read / configure expansion sub-module's AD / DA channel switch state and range type.
<b>Usage Syntax</b>	Syntax 1: be read: var = ZMIO_CONFIG (sel, moduleid) Syntax 2: be written: ZMIO_CONFIG (sel, moduleid, value) <ul style="list-style-type: none"> <li>● sel: function No.</li> <li>● moduleid: expansion submodule address</li> <li>● value: the channel value / range type expansion submodule to be configured.</li> </ul> For details, please refer to BASIC Programming Manual.

<b>Example</b>	<pre>ZMAIO_CONFIG(1, 0, 10) 'configure the submodule DA range type of address 0 as 0-10V ZMAIO_CONFIG(2, 0, 15) 'open all AD channels of address 0 ZMAIO_CONFIG(1, 0) 'get submodule's AD / DA range types of address 0 ZMAIO_CONFIG(2, 0) 'get submodule's AD channel switch states of address 0</pre>
----------------	---

# Chapter IV Expansion Module

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

## 4.1. 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.

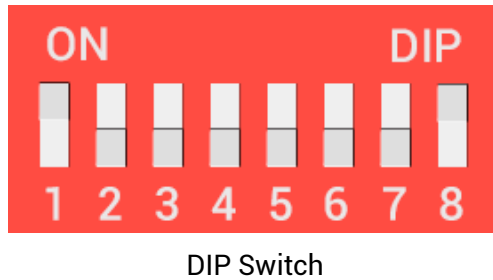
→ **Wiring Reference:**



### ● 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



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"

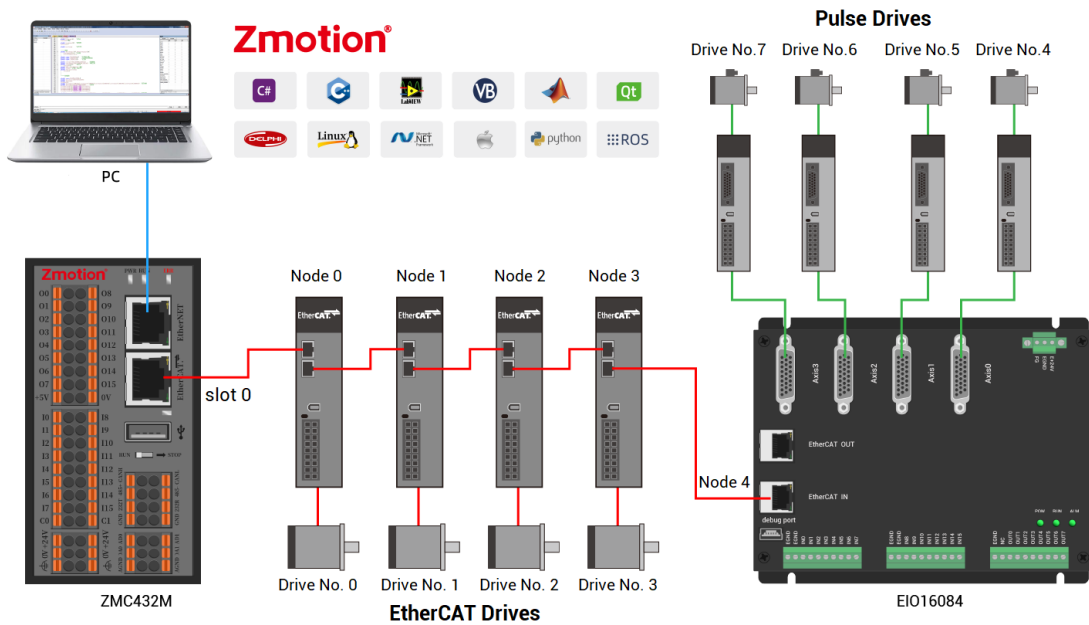
AXIS\_ADDRESS(6)=(32\*0)+2 'map axis 0 expanded by CAN module (ID is 2) as axis 6

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.





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:

For EtherCAT expansion module IO No., they are set by **NODE\_IO** and **NODE\_AIO** commands.

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 No. of slot 0 interface device 0 to 32

**NODE\_AIO(0,0,3)=3**    'set the AIO start No. of slot 0 interface device 0 to 3

## → 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:**

$AXIS\_ADDRESS(6)=(0<<16)+0+1$

'the first drive on the EtherCAT bus, drive number 0, bound as axis 6

$AXIS\_ADDRESS(7)=(0<<16)+1+1$

'the second drive on the EtherCAT bus, drive number 1, bound as axis 7

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

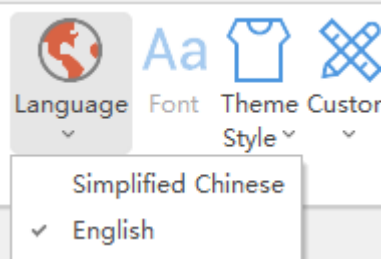
RTSys is a PC-side program development, debugging and diagnostic software for the Zmotion motion controllers. Through it, users can easily edit and configure the controller program, quickly develop applications, diagnose system operating parameters in real time, and debug the running program in real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

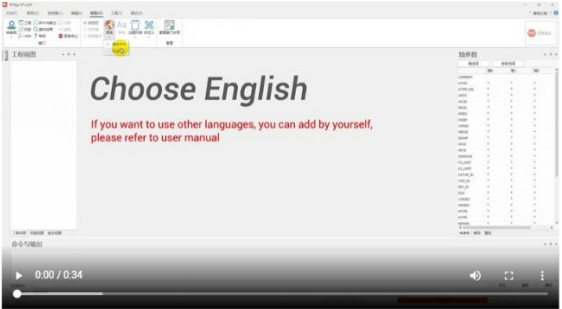
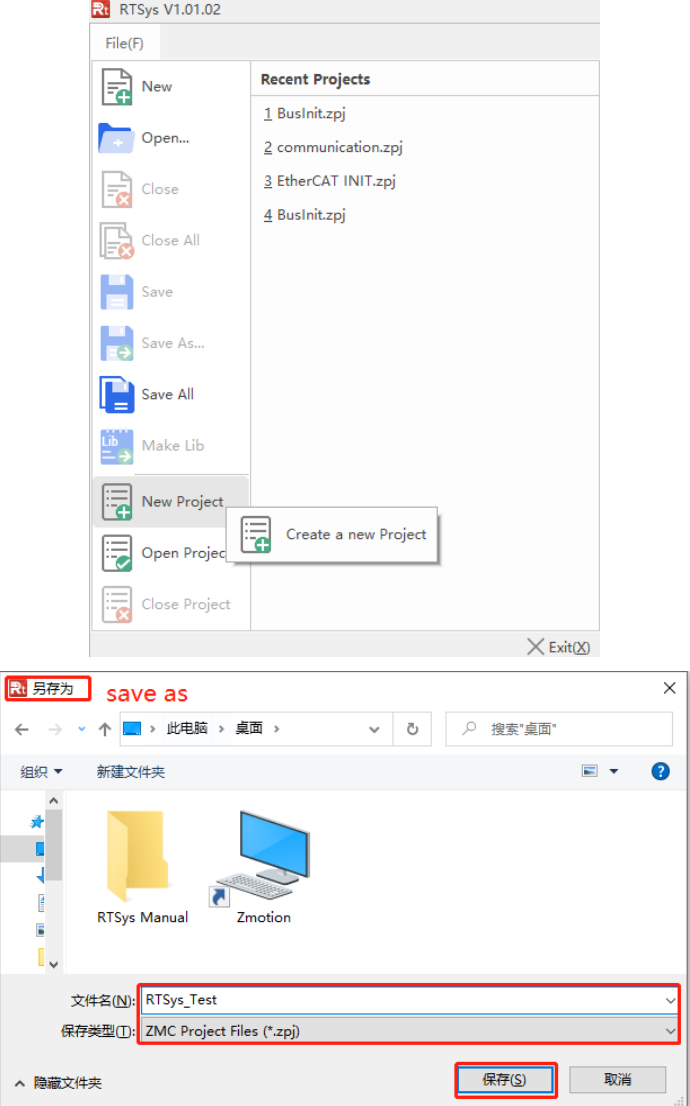
In RTSys, there are 4 programming languages for motion control development, Basic, PLC, HMI and C language, they can run multi-tasks among them, especially for Basic, multi-task running can be achieved separately, hybrid programming is also OK with PLC, HMI and C language.

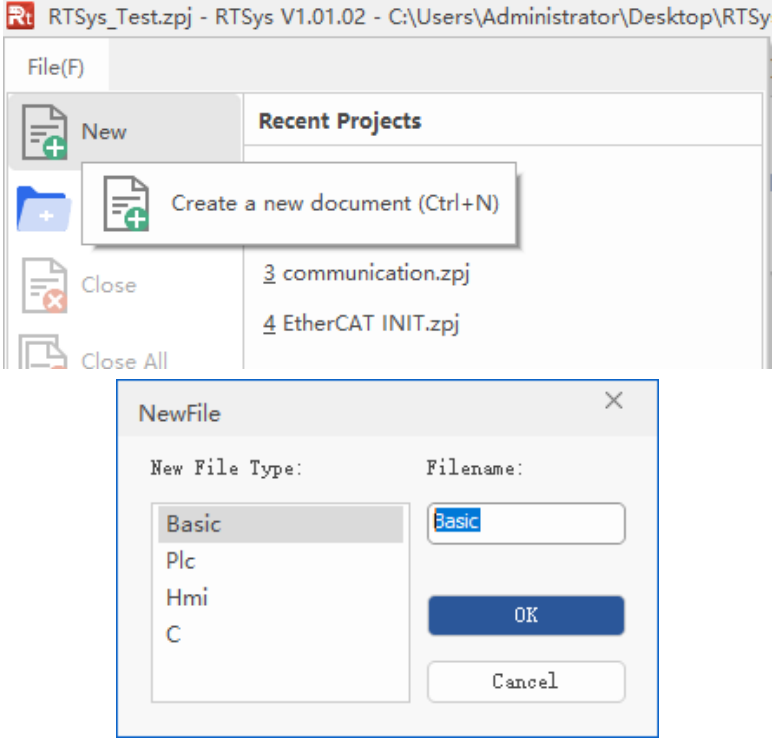
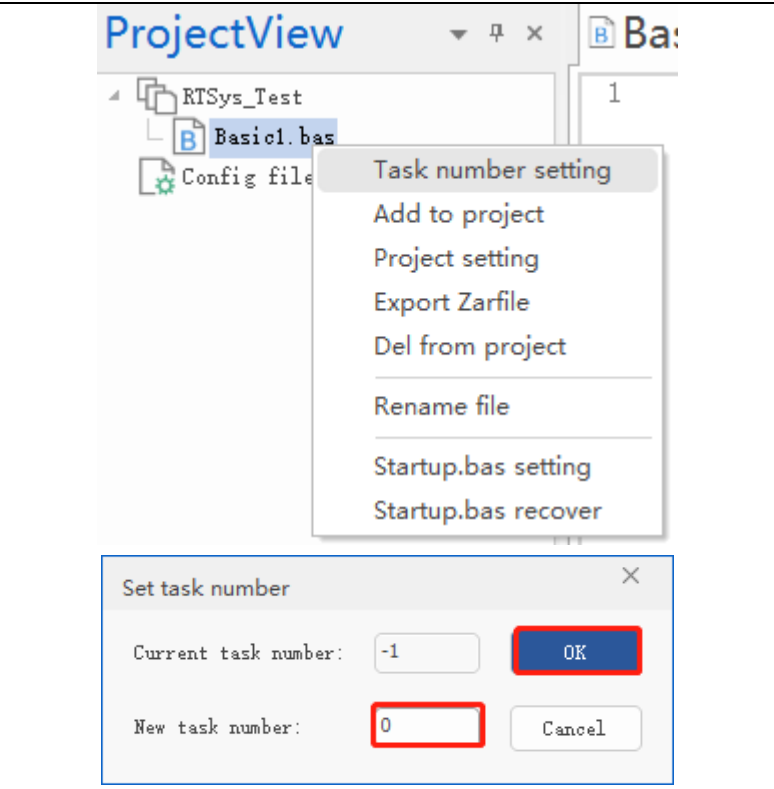
RTSys Downloading Address: [https://www.zmotionglobal.com/pro\\_info\\_282.html](https://www.zmotionglobal.com/pro_info_282.html)

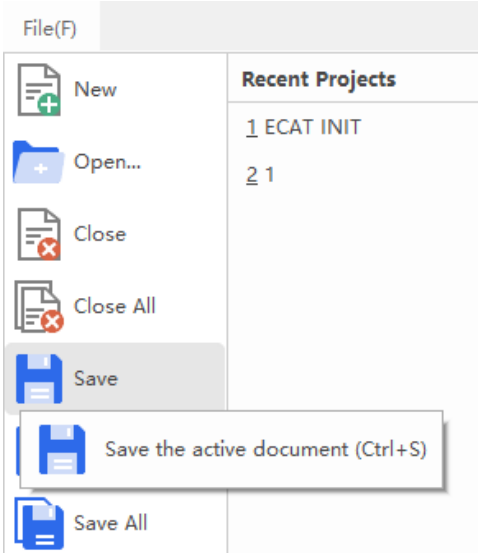
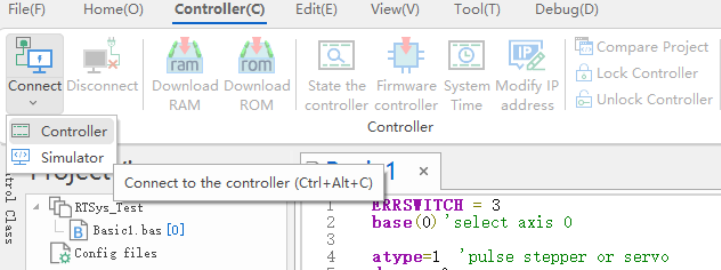
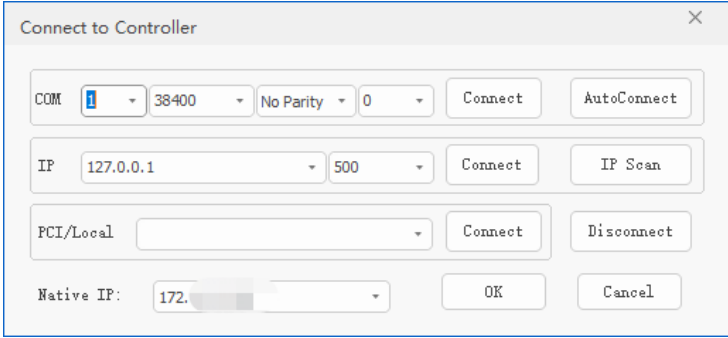
And related manuals can be found in "Download":

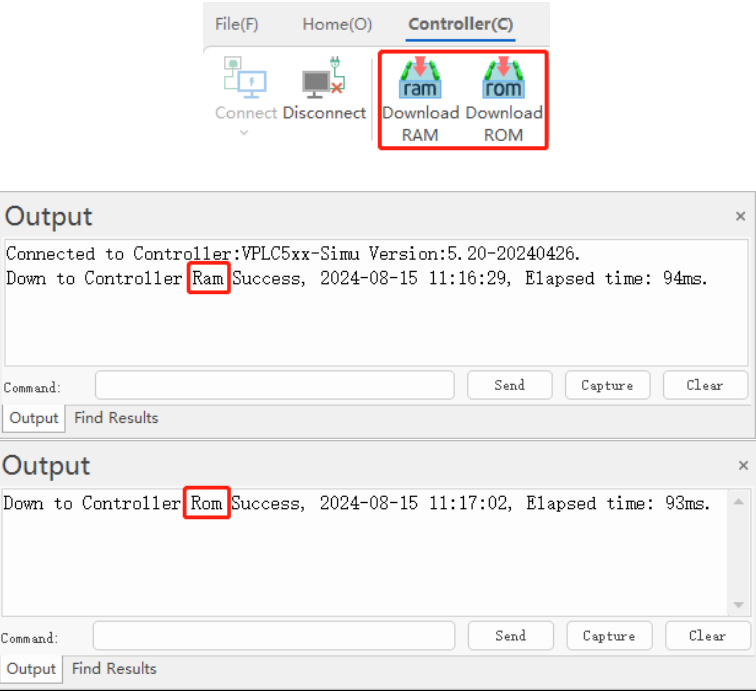
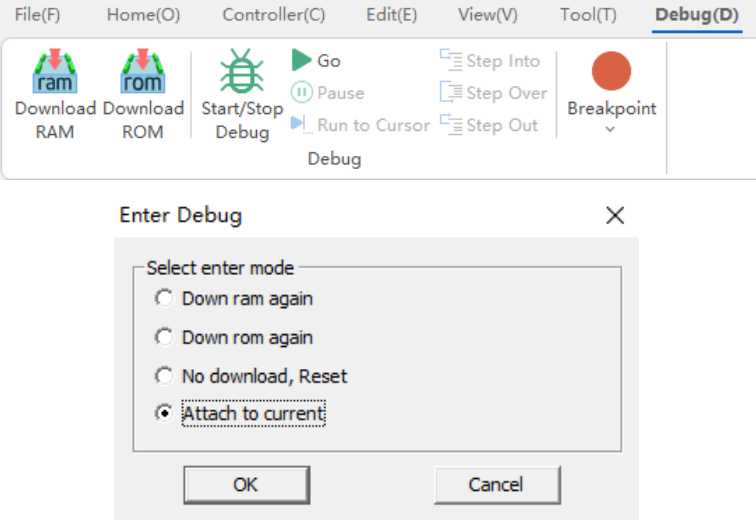
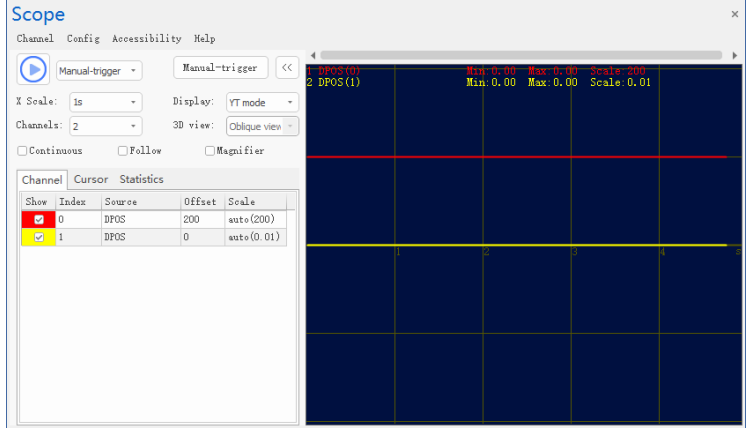
Name	Version No	Format	Size	Download
RTSys Development Software	V1.2.02	RAR	148MB	<a href="#">Download</a>
RTSys User Manual V1.2.0	V1.2.0	PDF	5.33MB	<a href="#">Download</a>
RTBasic Programming Manual	V1.1.0	PDF	18.3MB	<a href="#">Download</a>
RTHMI Programming Manual	V1.2.0	PDF	7.23MB	<a href="#">Download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

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

	<p>up one window, click OK, and restart it.</p>	<p style="text-align: center;"><u><a href="#">Language Switch Video Showing:</a></u></p> <p><b>E. How to Switch the Language</b></p> <p>Find “视图” (the fourth one in the above menu), then find the “语言”, choose English, restart RTSys. English RTSys will take effect when opened again.</p> 
<p>2</p>	<p><b>New Project:</b> “File” – “New Project”, Save as window will pop up, then enter file name, save the project file with suffix “zpj.”.</p>	

<p>3</p>	<p><b>New File:</b> "File"          – "New File",          select file type          to build, here          select Basic,          click "OK".</p>	 <p>The screenshot shows the 'File(F)' menu with 'New' and 'Recent Projects' options. A 'NewFile' dialog box is open, showing 'Basic' selected as the 'New File Type' and 'Basic' as the 'Filename'. The 'OK' button is highlighted.</p>
<p>4</p>	<p><b>Set Auto Run No.:</b> right click          the file, open          task number          setting window,          enter task No.,          which can be          any + value, no          priority, but not          the same.</p>	 <p>The screenshot shows the 'ProjectView' window with a context menu for 'Basic1.bas'. The 'Task number setting' option is selected. A 'Set task number' dialog box is open, showing 'Current task number' as -1 and 'New task number' as 0. The 'OK' button is highlighted.</p>

<p>5</p>	<p><b>Save File:</b> 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.</p>	
<p>6</p>	<p><b>Connection:</b> 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".</p>	 
<p>7</p>	<p><b>Download Program into</b></p>	<ul style="list-style-type: none"> <li>● <b>RAM:</b> it will not save when power off.</li> <li>● <b>ROM:</b> it will save data when power off, and when the program</li> </ul>

	<p><b>Controller:</b></p> <p>“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.</p>	<p>is connected to controller again, running according to task No.</p> 																		
<p>8</p>	<p><b>Debug:</b> “Debug” – “Start/Stop Debug” to call “Task” and “Watch” window, because it was downloaded before, here select “Attach the current”.</p>																			
<p>9</p>	<p><b>Scope function:</b></p> <p>Click “View” – “Scope” to open oscilloscope. It can capture needed data, for debugging.</p>	 <table border="1" data-bbox="587 1765 874 1883"> <thead> <tr> <th>Channel</th> <th>Cursor</th> <th>Statistics</th> </tr> <tr> <th>Show</th> <th>Index</th> <th>Source</th> <th>Offset</th> <th>Scale</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td> <td>0</td> <td>DPOS</td> <td>200</td> <td>auto (200)</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>1</td> <td>DPOS</td> <td>0</td> <td>auto (0.01)</td> </tr> </tbody> </table>	Channel	Cursor	Statistics	Show	Index	Source	Offset	Scale	<input checked="" type="checkbox"/>	0	DPOS	200	auto (200)	<input checked="" type="checkbox"/>	1	DPOS	0	auto (0.01)
Channel	Cursor	Statistics																		
Show	Index	Source	Offset	Scale																
<input checked="" type="checkbox"/>	0	DPOS	200	auto (200)																
<input checked="" type="checkbox"/>	1	DPOS	0	auto (0.01)																

**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.

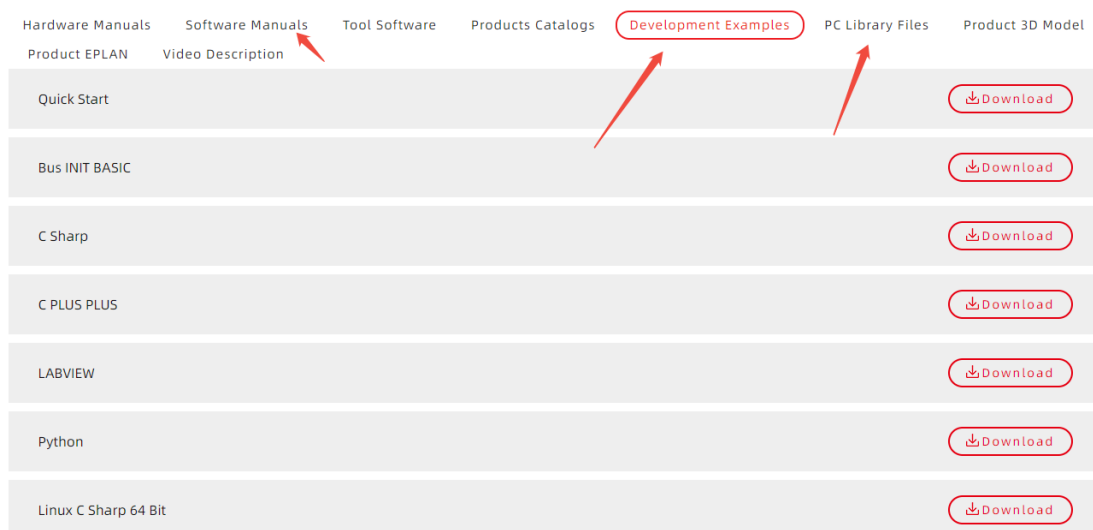
### 5.3. Program in Host-Computer by PC Languages

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](#)".



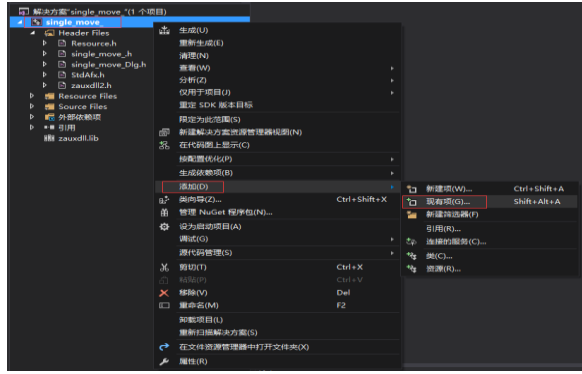
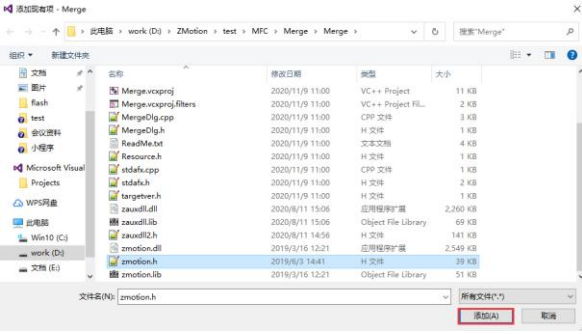
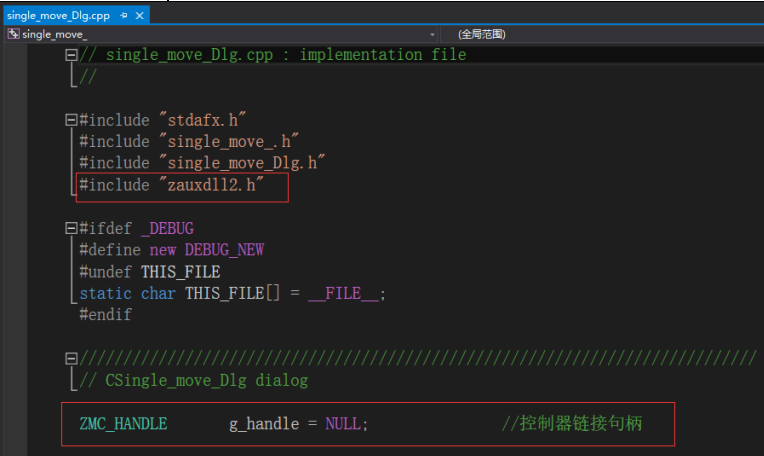
The program developed using the PC software cannot be downloaded to the controller, and it is connected to the controller through the dll dynamic library. The dll library needs to be added to the header file and declared during development.

- Get PC library file, example: [https://www.zmotionglobal.com/download\\_list\\_17.html](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.	

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

## 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.	<ol style="list-style-type: none"> <li>1. Check whether the ATYPE of the controller is correct.</li> <li>2. Check whether hardware position limit, software position limit, alarm signal work, and whether axis states are normal.</li> <li>3. Check whether motor is enabled successfully.</li> <li>4. Confirm whether pulse amount UNITS and speed values are suitable. If there is the encoder feedback, check whether MPOS changes.</li> <li>5. Check whether pulse mode and pulse mode of drive are matched.</li> <li>6. Check whether alarm is produced on motion controller station or drive station.</li> <li>7. Check whether the wiring is correct.</li> <li>8. Confirm whether controller sends pulses normally.</li> </ol>

The position limit signal is invalid.	<ol style="list-style-type: none"> <li>1. Check whether the limit sensor is working normally, and whether the "input" view can watch the signal change of the limit sensor.</li> <li>2. Check whether the mapping of the limit switch is correct.</li> <li>3. Check whether the limit sensor is connected to the common terminal of the controller.</li> </ol>
No signal comes to the input.	<ol style="list-style-type: none"> <li>1. Check whether the limit sensor is working normally, and whether the "input" view can watch the signal change of the limit sensor.</li> <li>2. Check whether the mapping of the limit switch is correct.</li> <li>3. Check whether the limit sensor is connected to the common terminal of the controller.</li> </ol>
The output does not work.	<ol style="list-style-type: none"> <li>1. Check whether IO power is needed.</li> <li>2. Check whether the output number matches the ID of the IO board.</li> </ol>
POWER led is ON, RUN led is OFF.	<ol style="list-style-type: none"> <li>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.</li> <li>2. Check whether the ALM light flickers regularly (hardware problem).</li> </ol>
RUN led is ON, ALM led is ON.	<ol style="list-style-type: none"> <li>1. Program running error, please check ZDevelop error code, and check application program.</li> </ol>
Fail to connect controller to PC through serial port.	<ol style="list-style-type: none"> <li>1. Check whether the serial port parameters are modified by the running program, you can check all the current serial port configurations through ?*SETCOM.</li> <li>2. Check whether the serial port parameters of the PC match the controller.</li> <li>3. Open the device manager and check whether the serial driver of the PC is normal.</li> </ol>
CAN expansion module cannot be connected.	<ol style="list-style-type: none"> <li>1. Check the CAN wiring and power supply circuit, whether the 120 ohm resistor is installed at both</li> </ol>

	<p>ends.</p> <ol style="list-style-type: none"> <li>2. Check the master-slave configuration, communication speed configuration, etc.</li> <li>3. Check the DIP switch to see if there are multiple expansion modules with the same ID.</li> <li>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)</li> </ol>
Fail to connect controller to PC through net port.	<ol style="list-style-type: none"> <li>1. Check IP address of PC, it needs to be at the same segment with controller IP address.</li> <li>2. Check controller IP address, it can be checked and captured after connection through serial port.</li> <li>3. When net port led is off, please check wiring.</li> <li>4. Check whether controller power led POWER and running indicator led RUN are ON normally.</li> <li>5. Check whether the cable is good quality, change one better cable to try again.</li> <li>6. Check whether controller IP conflicts with other devices.</li> <li>7. Check whether controller net port channel ETH are all occupied by other devices, disconnect to other devices, then try again.</li> <li>8. When there are multiple net cards, don't use other net cards, or change one computer to connect again.</li> <li>9. Check PC firewall setting.</li> <li>10. Use "Packet Internet Groper" tool (Ping), check whether controller can be Ping, if it can't, please check physical interface or net cable.</li> <li>11. Check IP address and MAC address through arp-a.</li> </ol>
RTSys reports 201 error code	<ol style="list-style-type: none"> <li>1. 201 means the sub-module doesn't exist, no ZMIO310 submodule is connected to ZMC432 slave expansion interface.</li> <li>2. Set ZMIO_OFFSET as negative value / the value that is more than IN &amp; OUT starting No. (must be the</li> </ol>

---

	multiples of 8), if not, it will report 201 error code, for example, ZMIO310_OFFSET = -8, ZMIO310_OFFSET = 24.
If the controller IP is modified, and power it off, but it becomes default IP 192.168.0.11	1. Please check DIP state, make sure the code is under RUN mode.