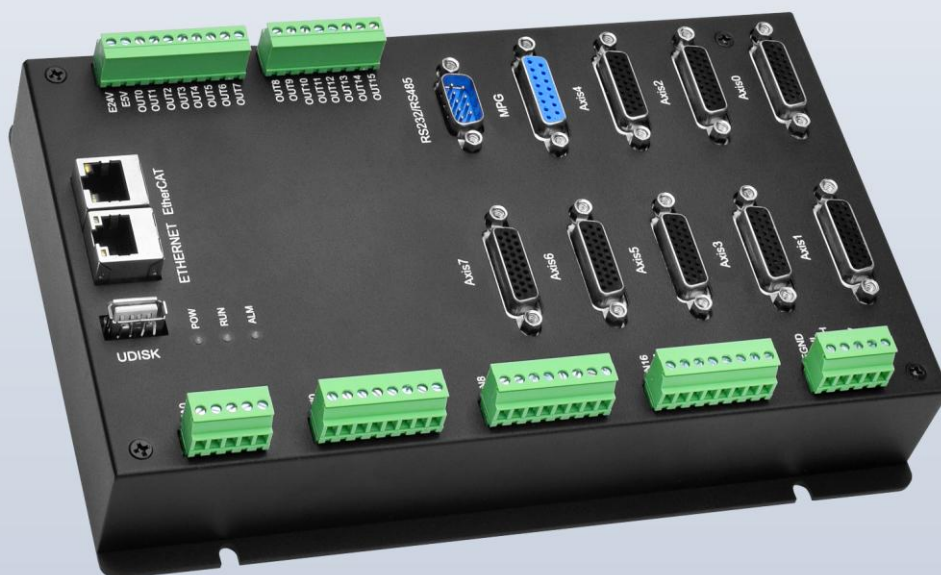
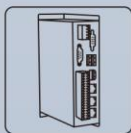


Pulse & EtherCAT Controller

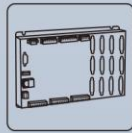
ZMC308H



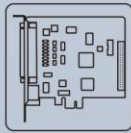
This manual is mainly for ZMC308H, ZMC308H-HW, ZMC308HL5, ZMC308HL5-HW
ZMC308HL24, ZMC308HL24-HW.



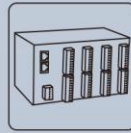
Vision Motion
Controller



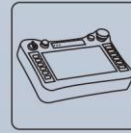
Motion
Controller



PC-based Motion
Control Card



IO Expansion
Module



Teach
Pendant

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.	

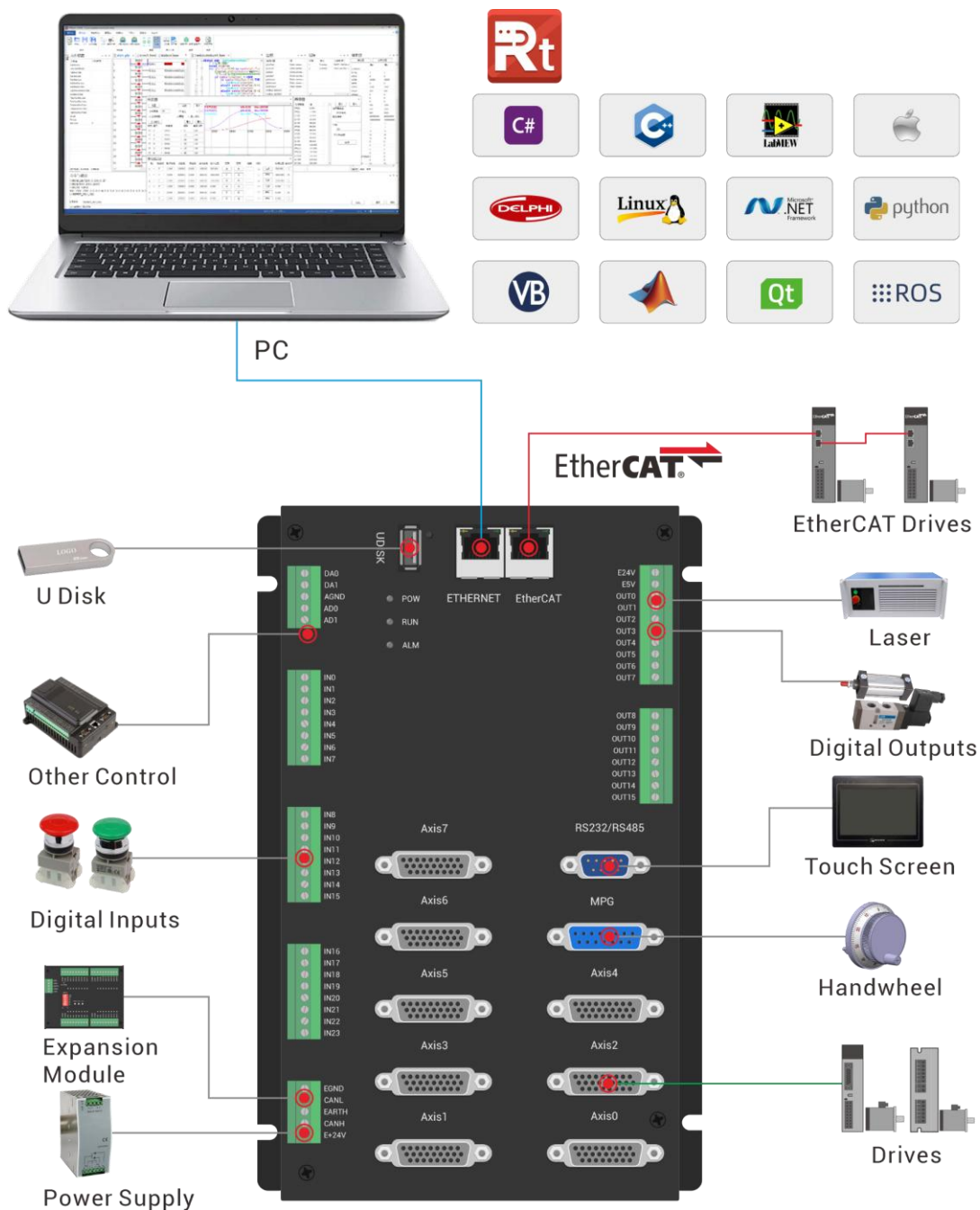
Content

Chapter I Production Information.....	4
1.1. Product Information	4
1.2. Interface Introduction.....	6
1.3. Specification Model.....	7
1.4. Nameplate & Model	8
1.5. Environment Requirements	9
1.6. Hardware Installment	10
Chapter II Hardware Interface	11
2.1. Power Input, CAN Communication Interface	11
2.2. IN: General Input.....	13
2.3. OUT: General Output	16
2.4. AD / DA Analog Interface.....	20
2.5. RS232 / RS485 Serial Port	22
2.6. U Disk.....	24
2.7. ETHERNET	25
2.8. EtherCAT	26
2.9. AXIS (Differential Pulse Axes)	28
2.10. MPG Handwheel Interface.....	33
Chapter III Expansion Module	36
3.1. CAN Bus Expansion	36
3.2. EtherCAT Bus Expansion	40
4.1. Program in RTSys Software	44
4.2. Upgrade Controller Frimware	49
4.3. Program in Host-Computer by PC Languages	50
Chapter V Operation and Maintain	53
5.1. Regular Inspection and Maintenance	53
5.2. Common Problems & Solutions	54

Chapter I Production Information

1.1. Product Information

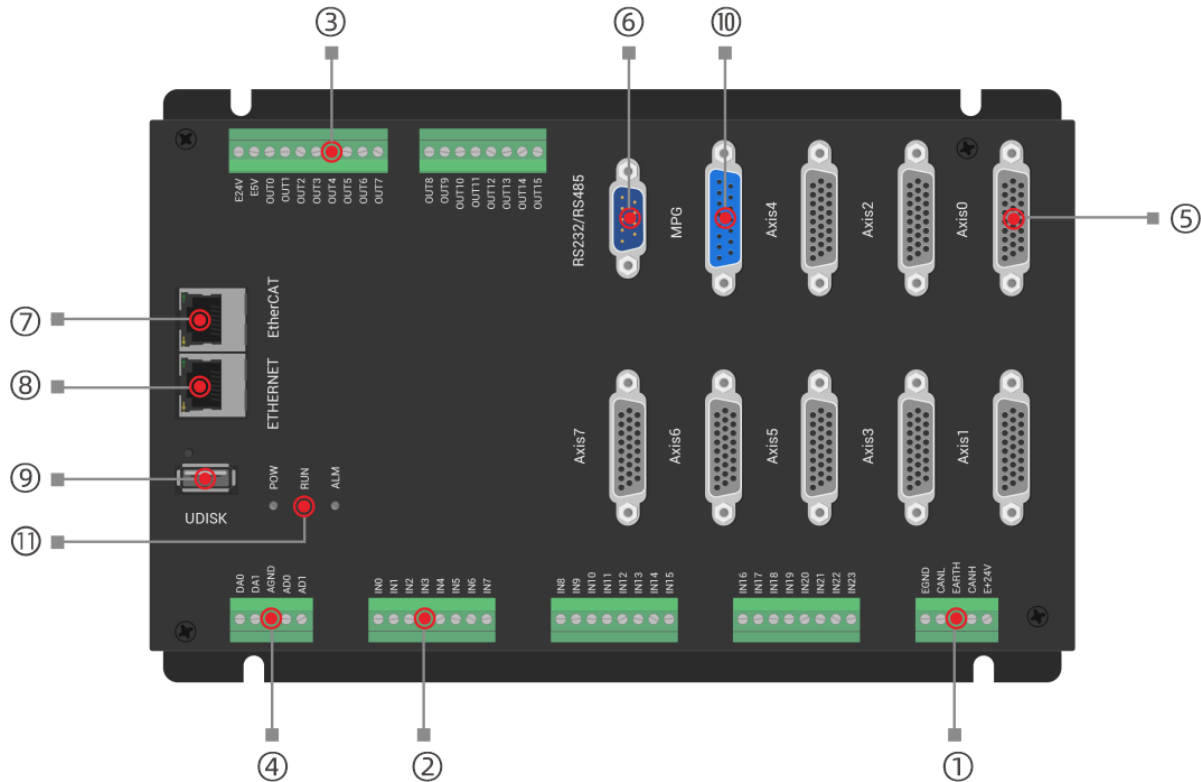
ZMC308H **high-performance multi-axis** motion controller is a stand-alone motion controller that is compatible EtherCAT with pulse type. The controller itself supports 8 axes at most to achieve complex continuous trajectory control requirements.



- ✚ There are EtherCAT axis, single-ended pulse axes, differential pulse axes and differential encoder interfaces.
- ✚ It supports MPG handwheel encoder specialized interface.
- ✚ **Basic Motion Control Functions:** point motion, electronic cam, linear interpolation, circular interpolation, continuous interpolation, Robot instruction, etc.
- ✚ **Special Motion Control Functions:** HW hardware comparison output, high-speed latch, PWM, etc., and for special models, it supports 5V/ 24V laser PWM output (ZMC308HL series).
- ✚ Built-in Linux system.

ZMC3 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
	CAN	Connect to CAN expansion modules to expand more resources.
②	General Digital IN	NPN type, IN0-23, it supports latch.
③	General Digital OUT	NPN type, OUT0-15, it supports single-ended pulse, HW, PWM functions
④	Analog DA	Single-ended type, 12bit, 0-10V
⑤	RS232/RS485	Connect to host computer, use MODBUS_RTU protocol.
⑥	UDISK	Connect to UDISK to update the program, import and export data.
⑦	EtherNET	Connect to host computer, use MODBUS_RTU protocol, which can be expanded by interchanger.
⑧	EtherCAT	Connect to EtherCAT drive and EtherCAT expansion module.

⑨	AXIS Differential Pulse Axis Interface	Connects to a servo drive, including direction pulses, an encoder, and dedicated digital I/O
⑩	MPG Handwheel Interface	Connects to 5–24 V handwheel
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.

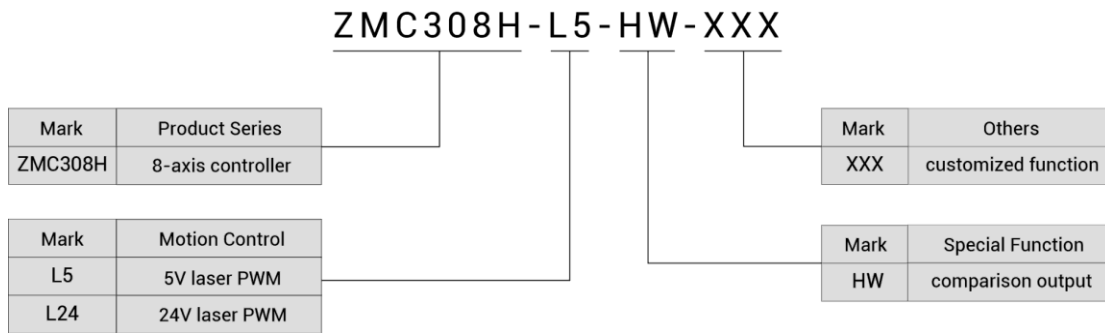
1.3. Specification Model

Model	ZMC308H	ZMC308H-HW	ZMC308HLx-HW
Basic Axes	8		
All Axes	32 (basic axis + virtual axis)		
EtherCAT	√		
AXIS (differential pulse axes)	8 (directional pulse + encoder + digital I/O)		
OUT (single-ended pulse axes)	4		
MPG (handwheel encoder)	1		
Digital IN	24 (General) + 16 (AXIS) + 12(MPG)		
Digital OP	24 (General) + 16 (AXIS)		
Max Extended Digital IN	≤4096		
Max Extended Digital OP	≤4096		
AD Input	2 (0-10V, 12bit)		
DA Output	2 (0-10V, 12bit)		
Max Extended AD	≤512		
Max Extended DA	≤512		

EtherNET	1		
EtherCAT	1		
CAN	1		
RS232	1		
RS485	1		
UDISK	1		
High-Speed Latch	8		
Hardware Comparison Output HW	0	4	4
General PWM	0	8	8
Laser PWM	0	0	6
Point Motion	√		
Electronic Cam			
Linear Interpolation			
Circular Interpolation			
Continuous Interpolation			
SCARA Robotic Arm			
Program Space	15MByte		
Power Down Storage	√		
Dimensions	221*114*37		

1.4. Nameplate & Model

■ Nameplate Information



■ Product Models

No.	Models	Description
1	ZMC308H	Not support Hardware comparison output, not support laser PWM output.
2	ZMC308H-HW	support Hardware comparison output, not support laser PWM output.
3	ZMC308HL5	Not support Hardware comparison output, support 5V laser PWM output.
4	ZMC308HL5-HW	support Hardware comparison output, support 5V laser PWM output.
5	ZMC308HL24	Not support Hardware comparison output, support 24V laser PWM output.
6	ZMC308HL24-HW	support Hardware comparison output, support 24V laser PWM output.

Note: ZMC308H doesn't support laser PWM output, if you need this function, please select corresponding models.

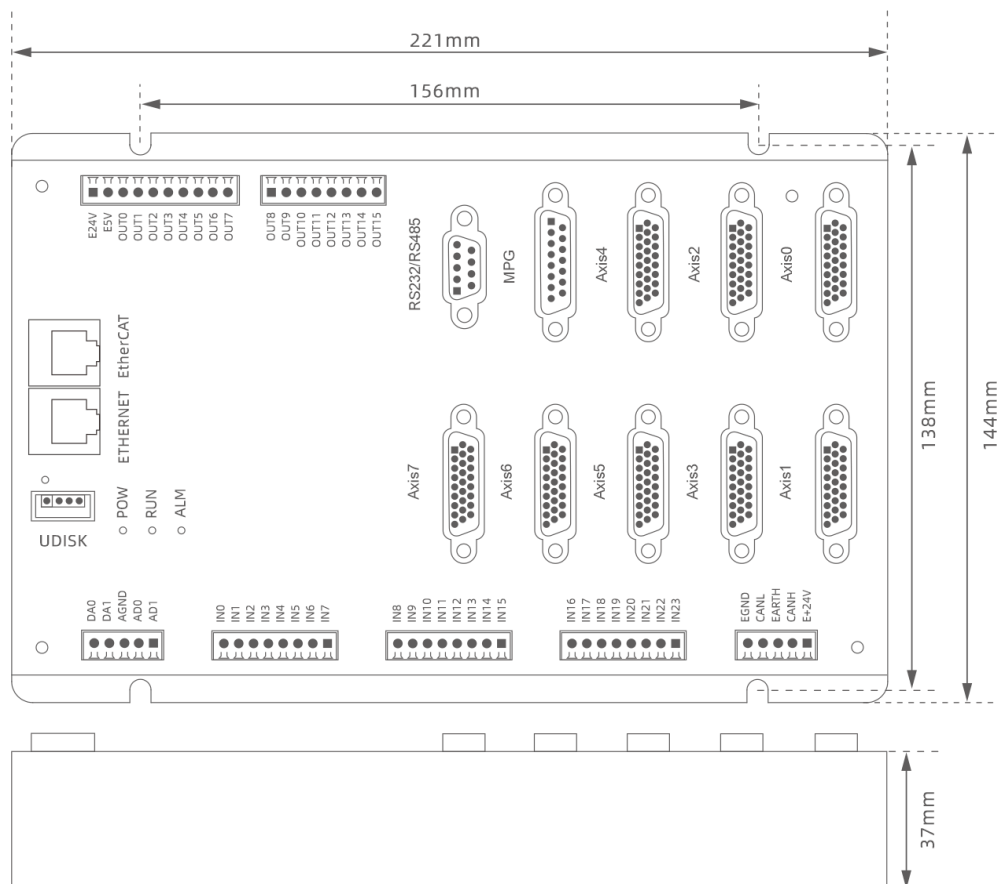
1.5. Environment Requirements

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

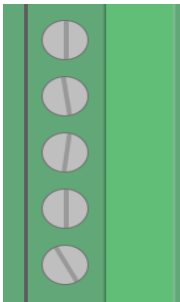
1.6. Hardware Installment

The ZMC308H 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, CAN Communication Interface

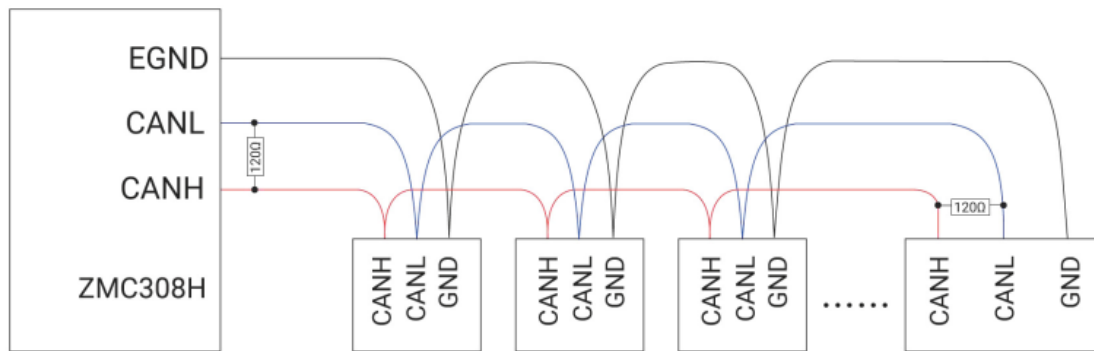
Terminal	Name	Function
	EGND	Power IN - / CAN Communication Public End
	CANL	CAN Communication L Side
	EARTH	Protect, shield
	CANH	CAN Communication H Side
	E+24V	Power IN +

→ Specification

Power	Description
Voltage	DC24V $\pm$ 5%
Max Power	10W
Anti-reverse connection	YES
Overcurrent Protection	YES
Isolation Power	YES
Cable Type	Recommend "1.0 mm ² copper conductor cable"

CAN	Description
Communication Rate	$\leq$ 1Mbps
Terminal Resistor	120 Ω
Topological Structure	Daisy Chain Topology
The number of nodes can be extended	$\leq$ 16
Wiring Length	Recommend <30m (500kbps)
Communication Isolation	YES

→ Wiring Reference



● Notes:

- As above, the daisy chain topology is used for wiring, the star topology structure cannot be used. The shorter 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 chip from burning out.
- Please use STP (Shielded Twisted Pair), especially in bad environments, and make sure the shielding layer is fully grounded (chassis).
- When on-site wiring, pay attention to make the distance between strong current and weak current, it is recommended for the distance to be more than 30cm.
- It should be noted that the equipment grounding on the entire line must be good, and the grounding of the chassis should be connected to the standard factory ground pile.

→ Usage Methods:

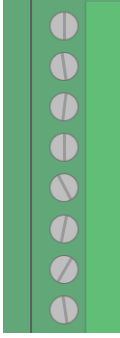
1. Please follow the above wiring instructions to wiring correctly, then connect the controller to [RTSys](#) through ethernet / serial port.
2. Configure controller CAN master station:
 - a. Use "CANIO_ADDRESS" command to set master station "address" and "velocity".
 - b. Use "CANIO_ENABLE" command 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.
3. 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”](#).
 4. After configured, repower on all slave stations to build normal communication, if the slave module “ALM” led is ON, which means the communication fails.
 5. 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.
 6. For above commands and others, please check [“RTBASIC Programming Manual”](#).

2.2. IN: General Input

Terminal	Name	Type	Function 1	Function 2
	IN0	High-Speed Input	Input 0	High-Speed Latch
	IN1		Input 1	
	IN2		Input 2	
	IN3		Input 3	
	IN4		Input 4	
	IN5		Input 5	
	IN6		Input 6	
	IN7		Input 7	
	IN8	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	/

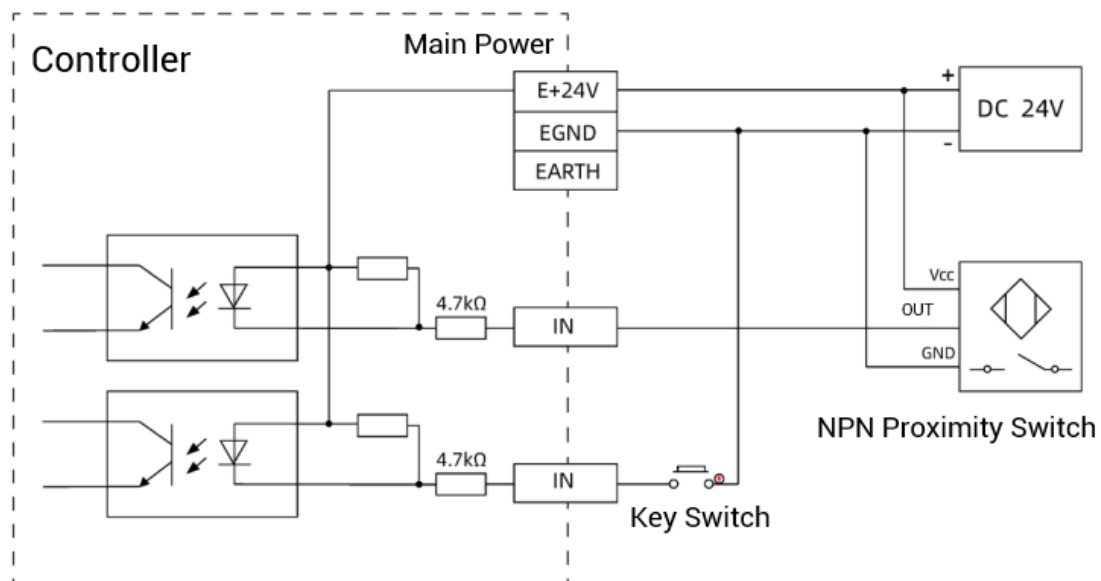
	IN16	I16	Low-Speed Input	Input 16	/
	IN17	I17		Input 17	/
	IN18	I18		Input 18	/
	IN19	I19		Input 19	/
	IN20	I20		Input 20	/
	IN21	I21		Input 21	/
	IN22	I22		Input 22	/
	IN23	I23		Input 23	/

Note: there are restrictions on the use of the IN0-7 latch channels; only any 4 of the 8 channels (IN0-7) can be selected for latching.

→ Specification

Item	High-Speed Input (IN0-7)	Low-Speed Input (IN8-23)
Input mode	NPN type, the input is triggered when by low-electric level	
Frequency	<400kHz	<5kHz
Impedance	4.7KΩ	4.7KΩ
Voltage	≤24V	≤24V
Communication Isolation	√	√

→ Wiring Reference



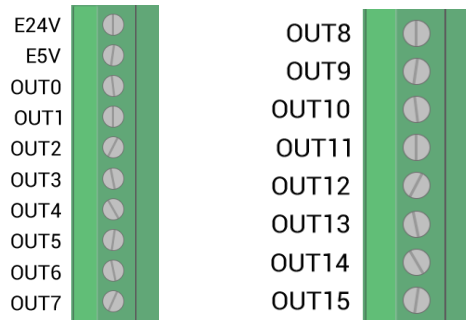
● Notes:

- The digital input wiring is shown as above, and the external loads can be key switches or sensors, but please note signals should be matched.
- It is better to use the same one power supply for the load and the controller, or you could connect both power negative poles together.
- When on-site wiring, pay attention to make the distance between strong current and weak current, it is recommended for the distance to be more than 30cm.
- It should be noted that the equipment grounding on the entire line must be good, and the grounding of the chassis should be connected to the standard factory ground pile

→ 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 ["RTBASIC Programming Manual"](#).

2.3. OUT: General Output



Name	Type	Functions			
		1	2	3	4
E24V	Freewheeling Enable	freewheeling clamp OUT	/	/	/
E5V	/	5V power output, max is 300mA	/	/	/
OUT0	High-Speed Outputs	output 0	PWM0	HW_0	Laser PWM0
OUT1		output 1	PWM1	HW_1	Laser PWM1
OUT2		output 2	PWM2	HW_2	Laser PWM2
OUT3		output 3	PWM3	HW_3	Laser PWM3
OUT4		output 4	PWM4	/	Laser PWM4
OUT5		output 5	PWM5	/	Laser PWM5
OUT6		output 6	PWM6	/	/
OUT7		output 7	PWM7	/	/
OUT8		output 8	DIR (AXIS 13)	/	/
OUT9		output 9	PUL (AXIS 13)	/	/
OUT10		output 10	DIR (AXIS 12)	/	/
OUT11		output 11	PUL (AXIS 12)	/	/
OUT12		output 12	DIR (AXIS 11)	/	/
OUT13		output 13	PUL (AXIS 11)	/	/
OUT14		output 14	DIR (AXIS 10)	/	/
OUT15		output 15	PUL (AXIS 10)	/	/
Notes: ✧ The E24V pin on the OUT is a freewheeling clamp port, please externally connect					

E24V to power E+24V, which can enable OUT0-15 freewheeling diode function.

- ✧ The E5V pin on the OUT is a voltage-converted output from the E+24V power supply port, providing a 5V power supply output with a maximum current of 300mA. It is commonly used to supply the 5V single-ended common-anode configuration for stepper driver control circuits.
- ✧ General PWM is an NPN/open-drain output; laser PWM is a push-pull output.
- ✧ OUT8-15 have the functions of axes 10/11/12/13. And it must do **remapping (please check AXIS_ADDRESS command)** at first, that is, map the target axis No. to its axis No. or other axes No.; when ATYPE = 1, it uses single-ended pulse axis function, when ATYPE = 0, it uses normal digital output function.
- ✧ The ZMC308H-HW also supports "Function 3 (HW0-HW3)"; the ZMC308HLx supports "Function 4 (Laser PWM0-Laser PWM5)"; the ZMC308HLx-HW supports both Function 3 (HW0-HW3) and Function 4 (Laser PWM0-Laser PWM5).

→ Specification (ZMC308H-HW)

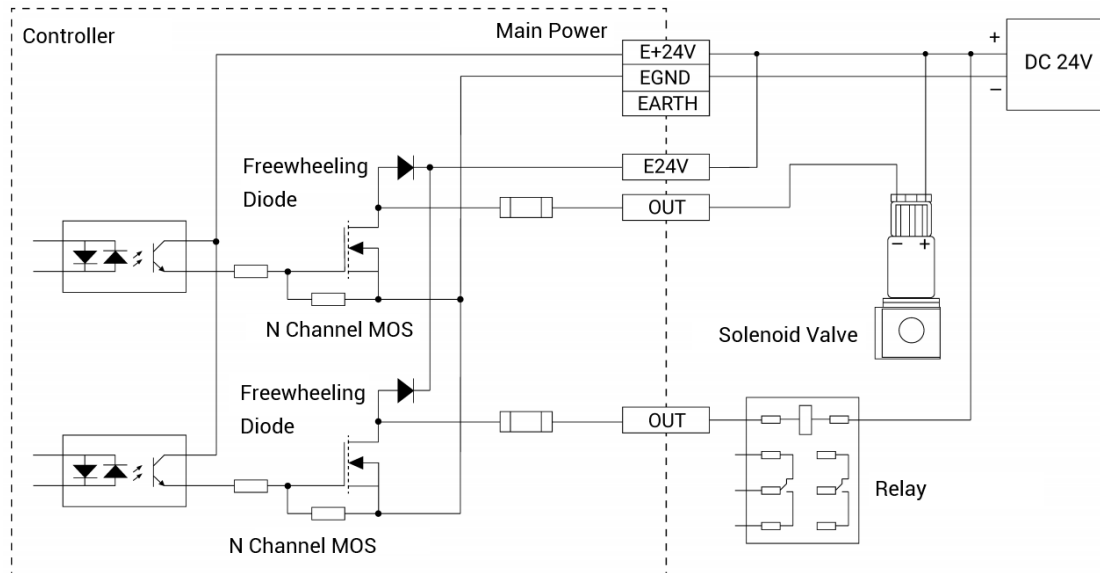
Item	High Speed Output (OUT0-15)
Output mode	NPN Type
Frequency	<400kHz
Voltage	≤24V
Output Current	≤500mA
Overcurrent protection	√
Communication Isolation	√

→ Specification (ZMC308HL-HW)

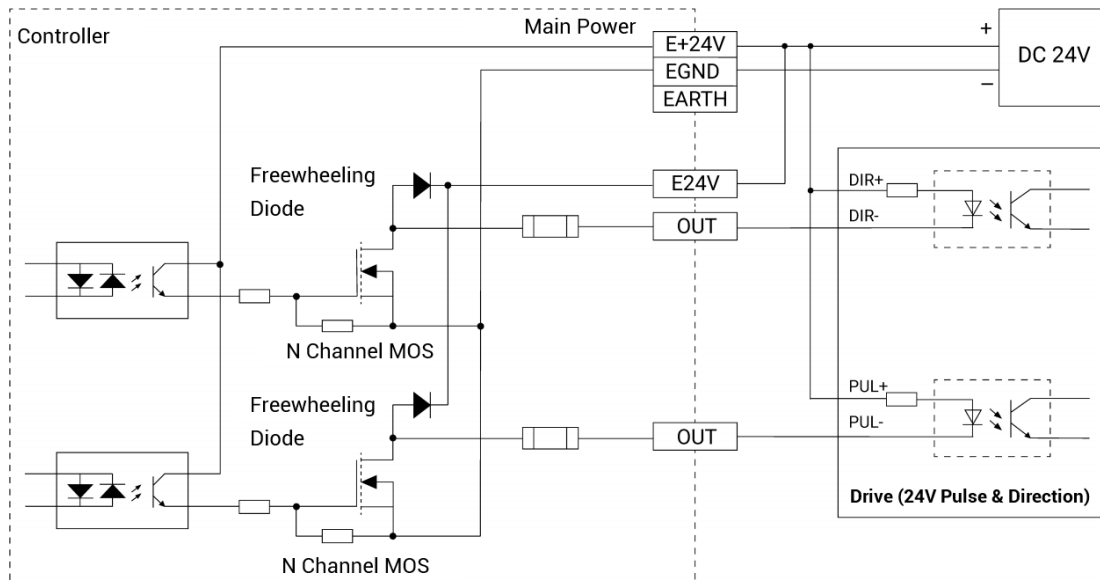
Item	Laser PWM Output (OUT0-5)	High Speed Output (OUT6-15)
Output mode	Push-pull type	NPN type
Frequency	<1MHz	<400kHz
Voltage	≤5V / ≤24V (optional)	≤24V
Output Current	≤50mA	≤500mA
Overcurrent protection	√	√
Communication Isolation	√	√

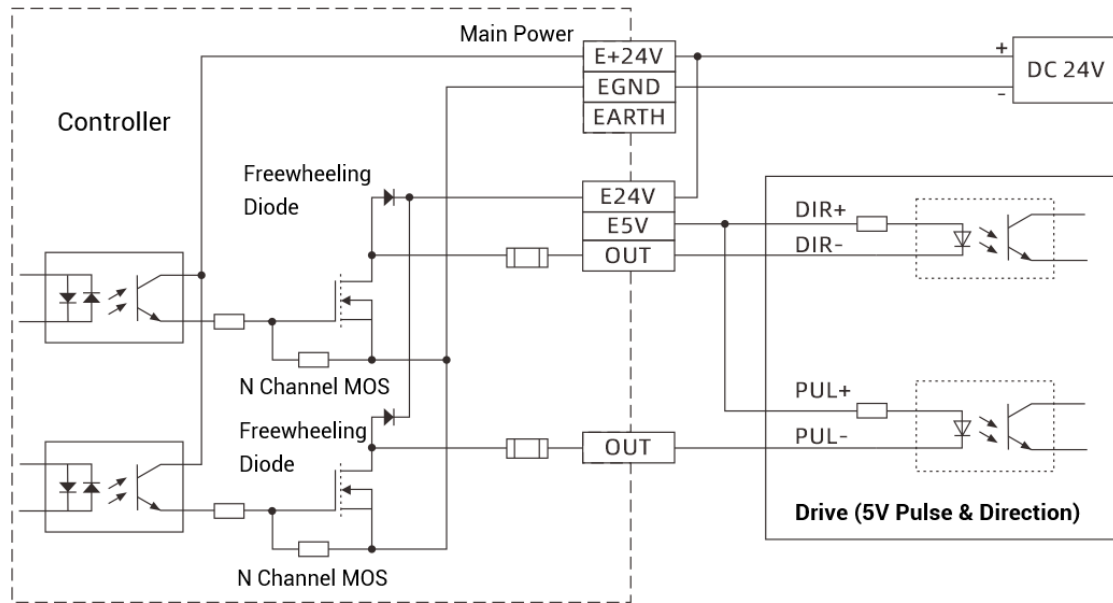
→ Wiring Reference

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 "RTBasic" 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**": **please note it is only to do position comparison output for MPOS encoder feedback position while using HW_PSWITCH command; if you want to compare DPOS or MPOS, HW_PSWITCH2 should be used.**
- (5) When it is used as pulse axis, the usage method is same as [2.9 AXIS](#). please refer to "BASIC Programming Manual – AXIS_ADDRESS".

axis_local = 10

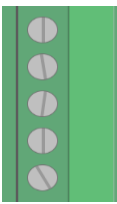
AXIS_ADDRESS (axis_local) = $(-1 \ll 16) + 10$ 'map target axis No. to other axes

BASE (axis_local)

ATYPE (axis_local) = 1

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

2.4. AD / DA Analog Interface

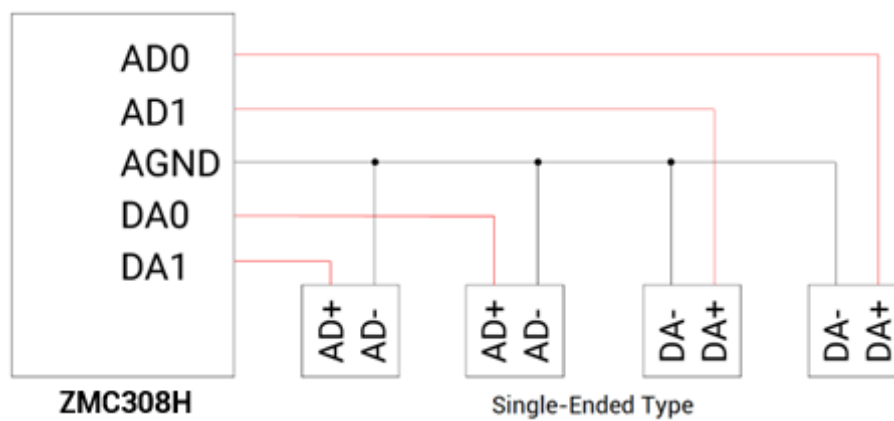
Terminal	Name	Function
 DA0 DA1 AGND AD0 AD1	DA0	Analog output terminal: AOUT(0)
	DA1	Analog output terminal: AOUT(1)
	AGND	Analog public end
	AD0	Analog input terminal: AIN(0)
	AD1	Analog input terminal: AIN (1)

→ 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	>40K Ω (input impedance)	>10K Ω (load requirement)
Signal Type	Single-ended	Single-ended
Cable Length	Recommend <5m	Recommend <5m

→ Wiring Reference



● Notes:

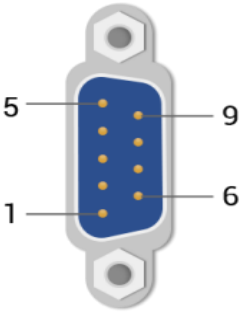
- The wiring for the analog is as shown in the diagram above. The load signal should be matched with impedance, otherwise, it will affect precision.
- Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.
- During on-site wiring, maintain a distance from 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 "RTBASIC Manual".

2.5. RS232 / RS485 Serial Port

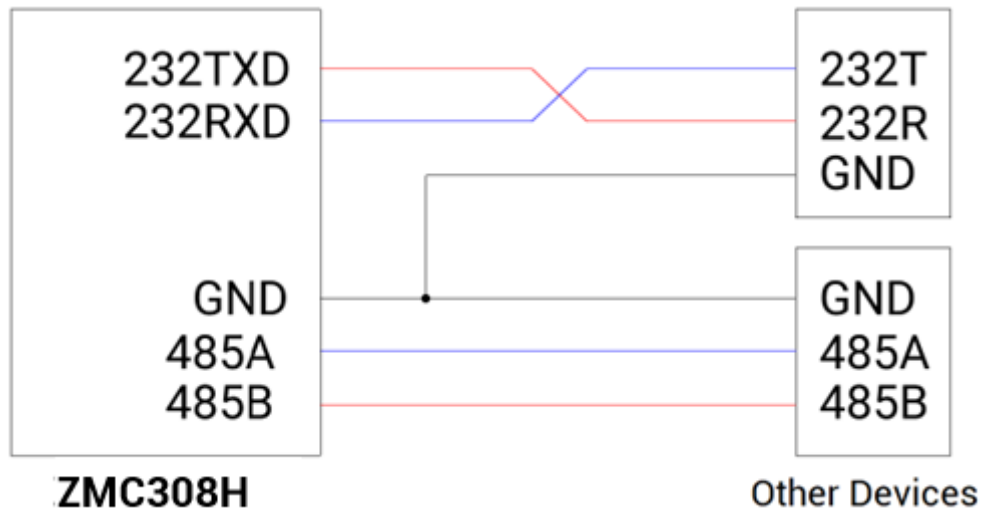
Terminal	PIN	Name	Function
	1, 6, 8	NC	Reserved
	2	232RXD	RS232, receive data
	3	232TXD	RS232, send data
	4	485A	RS485 signal A
	5	EGND	5V power -, and output for the public end
	7	485B	RS485 signal B
	9	E+5V	5V power +, maximum is 300mA

→ Specification

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

→ Wiring Reference

- RS485 / RS232 Wiring:



● Notes:

- 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.
- As above, the daisy chain topology is used for wiring (the star topology structure cannot be used). The distance between nodes, the better.
- Please be sure to connect the public ends of each communication node to prevent 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.

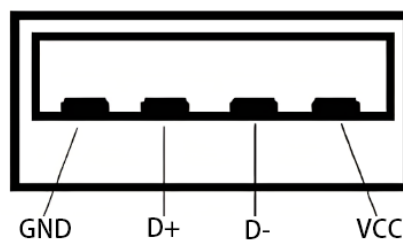
(2) How to use RS485:

- 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.

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

2.6. U Disk

The ZMC308H 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

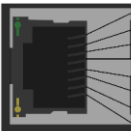
Item	Description
Communication Protocol	USB2.0
Communication Velocity	≤12Mbps
Whether Isolates	No

→ 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 "RTBASIC Programming Manual".

2.7. ETHERNET

→ Specification

PIN Definition				Item	Description
	PIN	Signal	Description	Communication Protocol	MODBUS_TCP
	1	TX+	Send signal (+)	Communication Velocity	100Mbps
	2	TX-	Send signal (-)		
	3	RX+	Receive signal (+)	Default IP	192.168.0.11
	4	NC	Reserved		
	5	NC	Reserved	Communication Cable	Shielded Cat 5e Cable
	6	RX-	Receive signal (-)		
	7	NC	Reserved		
8	NC	Reserved	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.

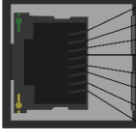
→ Usage Methods

- (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 "RTBASIC Programming Manual".

2.8. EtherCAT

→ Specification

PIN Definition	Item	Description
	Communication	EtherCAT

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

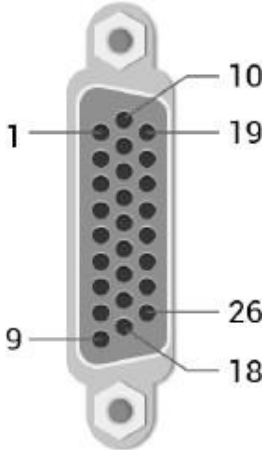
(3) How to connect to expansion module by EtherCAT:

- 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 "RTBASIC Programming Manual".

2.9. AXIS (Differential Pulse Axes)

This product provides 8 AXIS differential pulse axes interfaces for communication with servo drives.

Interface	Pin	Signal	Description
	1	EGND	IO public end / Common Zero
	2	IN (24-31)/ALM	General IN (better to do driver alarm)
	3	OUT (16.18.20.22. 24.26.28.30) / ENA	General OUT (better to do driver enable)
	4	EA-	Encoder IN A-

	5	EB-	Encoder IN B-
	6	EZ-	Encoder IN Z-
	7	+5V	Differential 5V common anode
	8	Reserved	Reserved
	9	DIR+	Directional output +
	10	GND	Differential signal public end / common zero
	11	PUL-	Pulse output -
	12	Reserved	Reserved
	13	GND	Differential signal public end / common zero
	14	OVCC	IO 24V common anode
	15	OUT (17.19.21.23.25 27.29.31) / CLR	OUT, better to do clear signal
	16	IN (32-39) / INP	OUT, better to do on-position signal
	17	EA+	Encoder IN A+
	18	EB+	Encoder IN B+
	19	EZ+	Encoder IN Z+
	20	GND	Differential signal public end / common zero
	21		
	22	DIR-	Directional output-
	23	PUL+	Pulse output +
	24	GND	Differential signal public end / common zero
	25	Reserved	Reserved
	26	Reserved	Reserved
Note: ✧ ALM, ENABLE, CLR and INP are recommended to be used as axis IO, because the drive capacity is small. ✧ OVCC, +5V are only used for communication between the controller and the servo driver, please do not use it as power supply for other places.			

Pulse Axis PIN No. & IO:

Pulse Axis No.	IN (PIN 2)	OUT (PIN 3)	OUT (PIN15)	IN (PIN16)
AXIS0	IN24	OUT16	OUT17	IN32
AXIS1	IN25	OUT18	OUT19	IN33
AXIS2	IN26	OUT20	OUT21	IN34

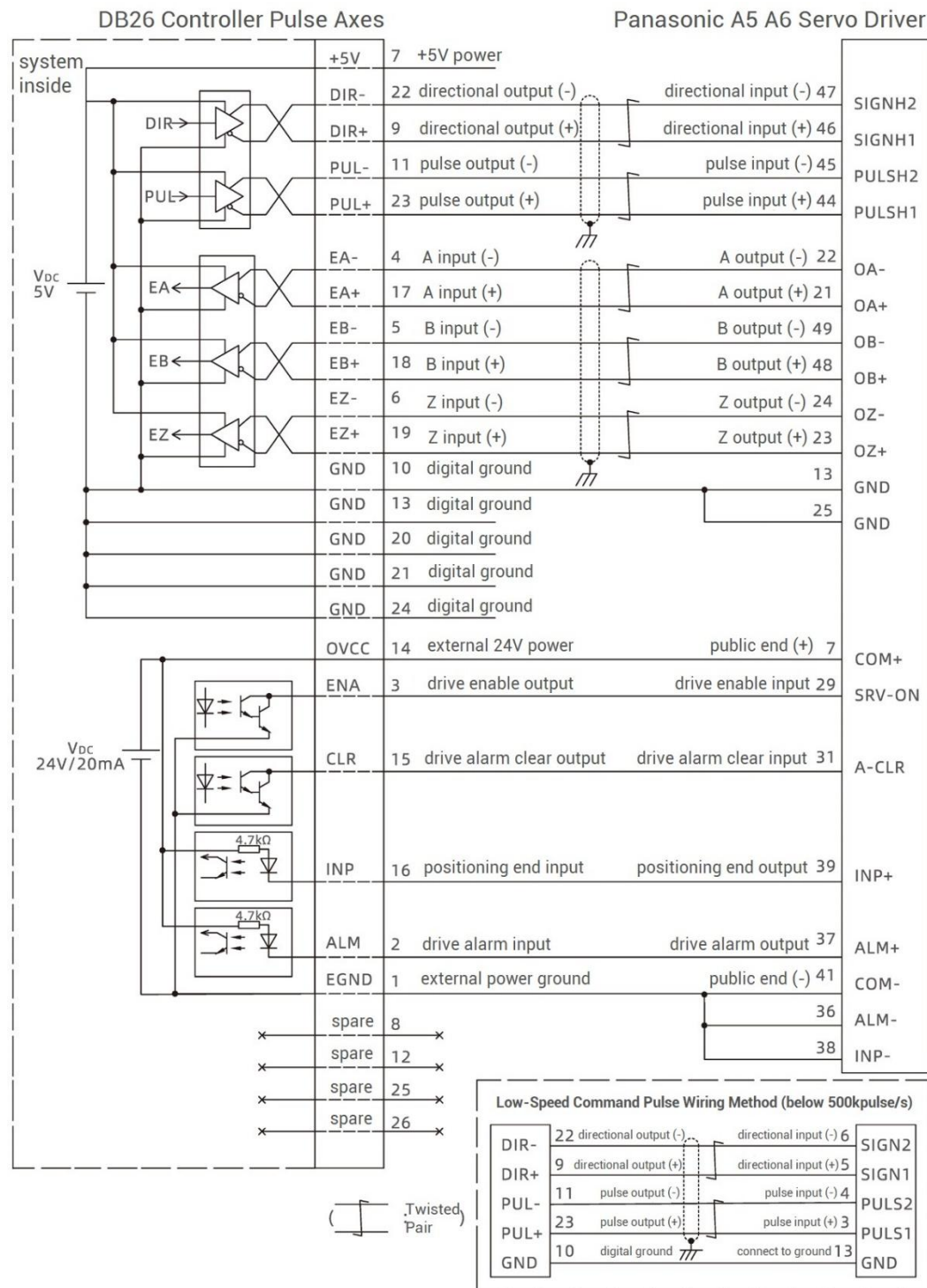
AXIS3	IN27	OUT22	OUT23	IN35
AXIS4	IN28	OUT24	OUT25	IN36
AXIS5	IN329	OUT26	OUT27	IN37
AXIS6	IN30	OUT28	OUT29	IN38
AXIS7	IN31	OUT30	OUT31	IN39

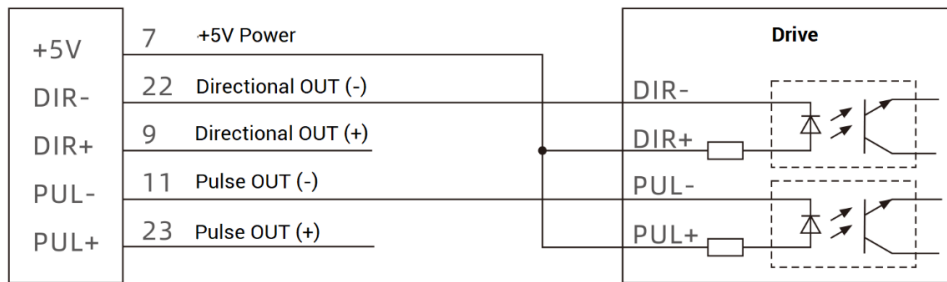
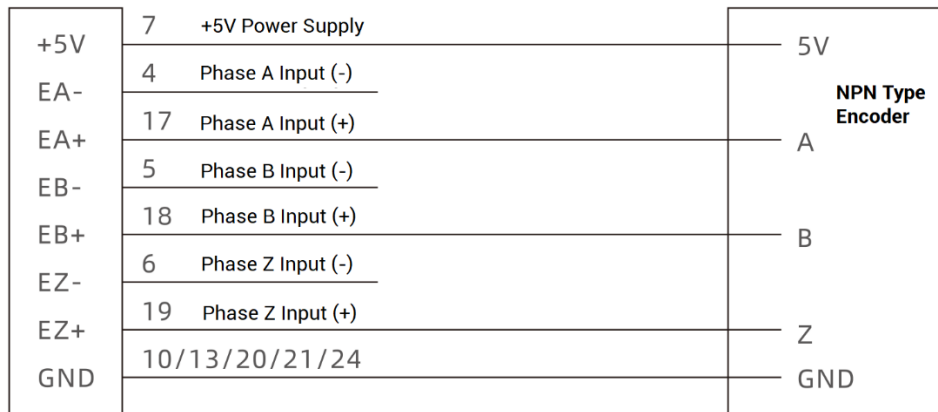
→ **Specification:**

Signal	Item	Description
PUL/DIR	Signal type	Differential output signal
	Voltage range	0-5V
	Maximum frequency	5MHz
EA/EB/EZ	Signal type	Differential input signal
	Voltage range	0-5V
	Maximum frequency	5MHz
IN24-39	Input method	NPN type, it is triggered when low electric level is input.
	Frequency	<5kHz
	Impedance	6.8KΩ
	Voltage level	≤24V
	Communication Isolation	√
OUT16-31	Output method	NPN type
	Frequency	<8kHz
	Voltage level	≤24V
	Maximum current	≤50mA
	Overcurrent protection	No
	Communication Isolation	√
+5V	Output Current	≤50mA
OVCC	Output Current	≤50mA

→ Wiring Reference:

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



➤ **Single-Ended Pulse Axis Output (PUL / DIR) Wiring:**➤ **Single-Ended Encoder Input (EA / EB / EZ Common Anode) Wiring:**→ **Wiring Note:**

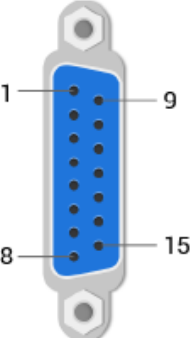
- ✧ The wiring principle of the differential pulse axis interface is shown in the figure above, and the wiring methods of different types of drivers are different, please connect carefully.
- ✧ When speed is met, use low-speed differential pulse preferentially. When high-speed differential pulse interface is used, please make sure controller internal digital ground is connected to driver high-speed pulse reference ground.
- ✧ Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.
- ✧ Some servo drives are not isolated by optocoupler. At this time, the GND must be connected to the GND of the driver. The encoders of most drives are not isolated by optocoupler. When connecting the encoder, GND must be connected.

→ Basic Usage Method:

1. Please follow the above wiring instructions to wiring correctly.
2. After powered on, please use ETHERNET or RS232 (default parameter, it can be
3. connected directly) to connect to RTSys.
4. Set axis parameters, such as, ATYPE, UNITS, SPEED, ACCEL, FWD_IN, REV_IN, etc.
5. There are many parameters related to pulse axis, they can be set and checked through relative instructions, please see "axis parameter and axis status" of "Basic", or see "RTSys/View/Axis parameter".

2.10. MPG Handwheel Interface

→ Interface Definition

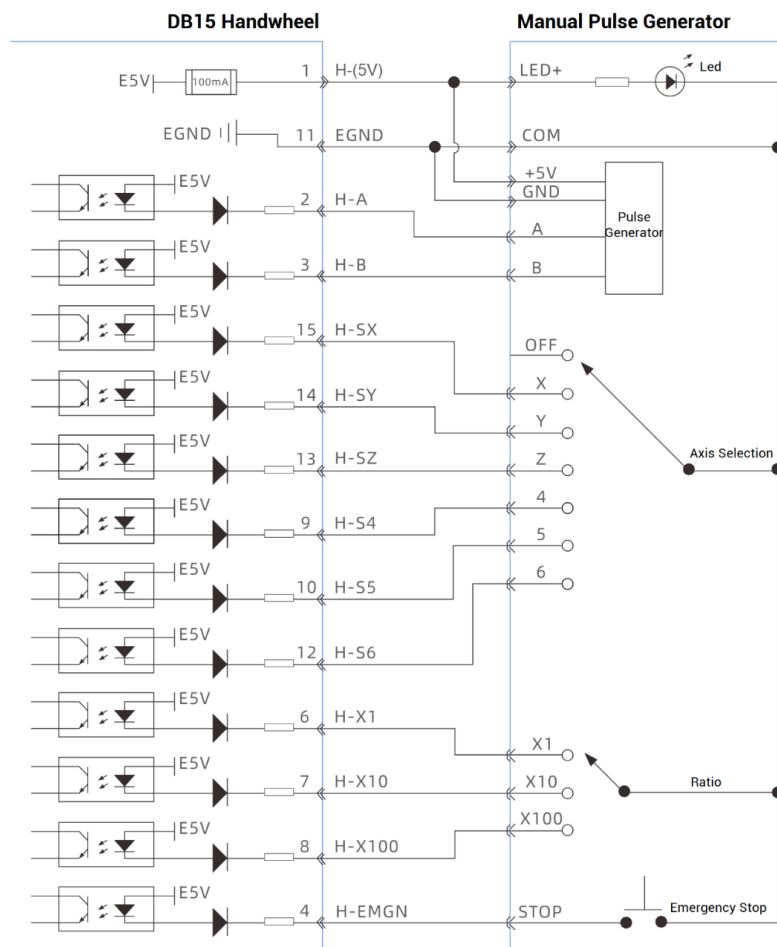
Interface	Pin	Signal	Description
	1	H-5V	The positive pole of 5V power supply, which supplies power only for handwheel
	2	H-A	Encoder signal A (IN40)
	3	H-B	Encoder signal B (IN41)
	4	H-EMGN	Emergency stop signal (IN51)
	5	NC	Spare (reserved)
	6	H-X1	Select the ration as X1 (IN42)
	7	H-X10	Select the ratio as X10 (IN43)
	8	H-X100	Select the ratio as X100 (IN44)
	9	H-S4	Select axis 3 (IN48)
	10	H-S5	Select axis 4 (IN49)
	11	EGND	5V power output (-), signal public end
	12	H-S6	Select axis 5 (IN50)
	13	H-SZ	Select axis 2 (IN47)
	14	H-SY	Select axis 1 (IN46)
	15	H-SZ	Select axis 0 (IN45)
Notes:			

- H-5V power output is only for handwheel power supply, don't use for other devices;
- All signals in this interface are digital input signals, No. are IN (40-51);
- H-A / H-B corresponding encoder IN is AXIS8 (default). While reading handwheel data, ATYPE (8) = 3 / 6, and MPOS (8) read data is the handwheel data; if you want to map it to other axes, please do remapping in software at first.

→ Specification

Item	Descriptions
Input Mode	NPN type, input will be triggered by low electric level
Frequency	<5kHz
Impedance	510Ω
Voltage	≤24V
Communication Isolation	√
H-5V Output Current	≤100mA

→ Wiring Reference



→ **Wiring Notes**

- The wiring principle of handwheel encoder axis interface is shown above, the design of handwheel is very rich, please pay attention to do connection.
- 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. Refer to above handwheel wiring graphic, correctly connect the handwheel and controller.
2. After powered on, please select ETHERNET or RS232 to connect to RTSys.
3. Configure axis No., for ZMC308H, the default handwheel axis No. is 8, and it must use AXIS_ADDRESS command to do remapping.
4. Configure IO: assign axis selection (SX, SY, SZ, S4, S5, S6) and ratio (X1, X10, X100) and emergency stop (EMGN) functions as required. These signals are essentially digital input signals with fixed No. but no fixed functions. It needs RTSys development (the axis selection is the connected axis of "connect" synchronization motion, and the ratio is the "connect" ratio).
5. After completed above steps, it still needs to set some basic axes parameters related to handwheel axis and the connected axis, like, "ATYPE, UNITS, etc."; for corresponding configuration information, it can be checked in ["2.9 AXIS -- Usage"](#).

Chapter III Expansion Module

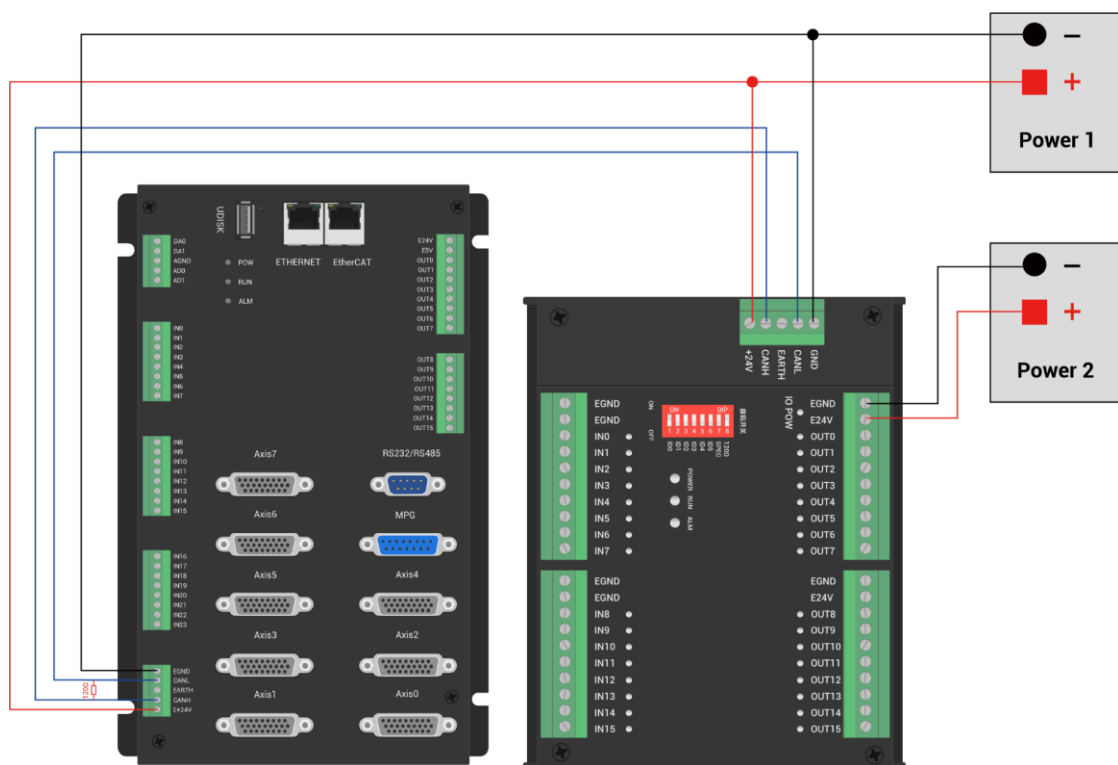
When you need more resources, expansion modules can be used by CAN / EtherCAT.

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, pulse 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:

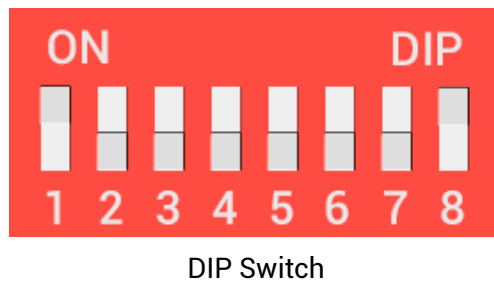


● Notes:

- ✧ 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: CAN address ID, the combination value is 0-15 (convert from 4-bit binary to decimal system);
- ✧ 5-6: CAN communication speed, corresponding value is 0-3 (convert from 2-bit binary to decimal system), 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 (*for ZMC308H, its local max IO No. is IN51 and OUT31, then the expansion board IO starting No. is 64 at least, and DIP ID address starts from 3*):

--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: (for ZMC308H, its local max AIO No. is AIN1 and AOUT1, then the expansion board AIO starting No. is 8 and 4 at least, the DIP ID address can start from 0 directly):

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.

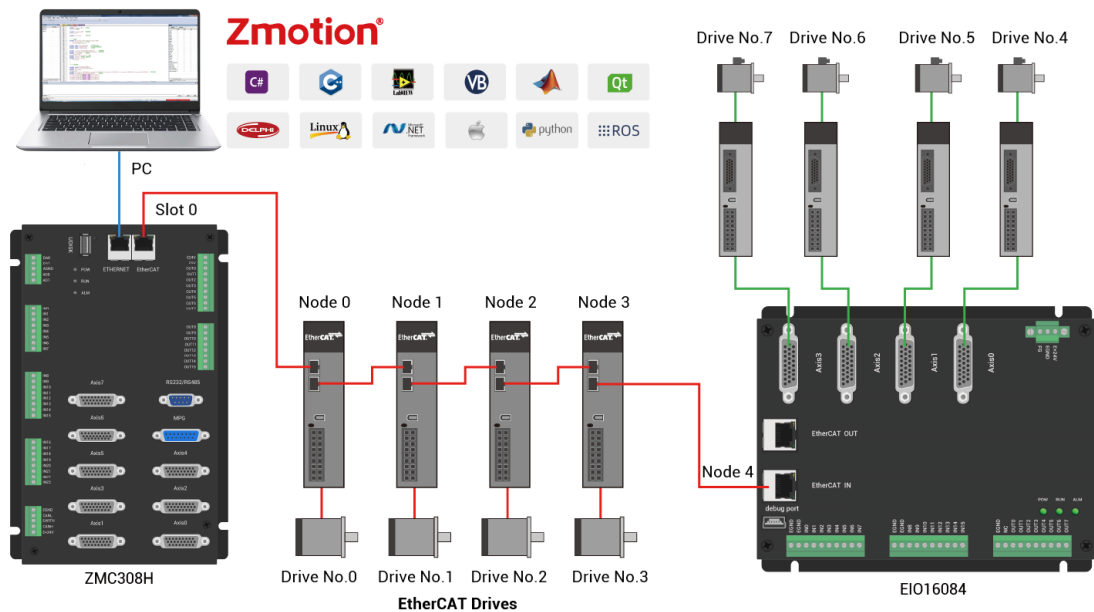
"AXIS_ADDRESS(axis No.)=(32*expanded axis)+ID "

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

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:

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:

$\text{AXIS_ADDRESS}(\text{axis number}) = (\text{slot number} \ll 16) + \text{driver number} + 1$

Example:

$\text{AXIS_ADDRESS}(6) = (0 \ll 16) + 0 + 1$

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

$\text{AXIS_ADDRESS}(7) = (0 \ll 16) + 1 + 1$

'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 IV Programming

4.1. Program in RTSys Software

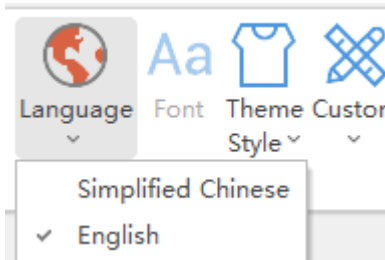
RTSys is a PC-side program development, debugging and diagnostic software for the Zmotion motion controllers. Through it, users can easily edit and configure the controller program, quickly develop applications, diagnose system operating parameters in real time, and debug the running program in real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

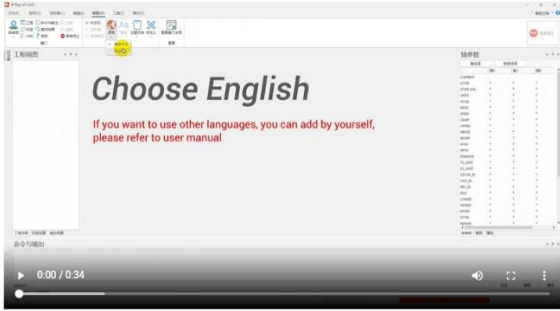
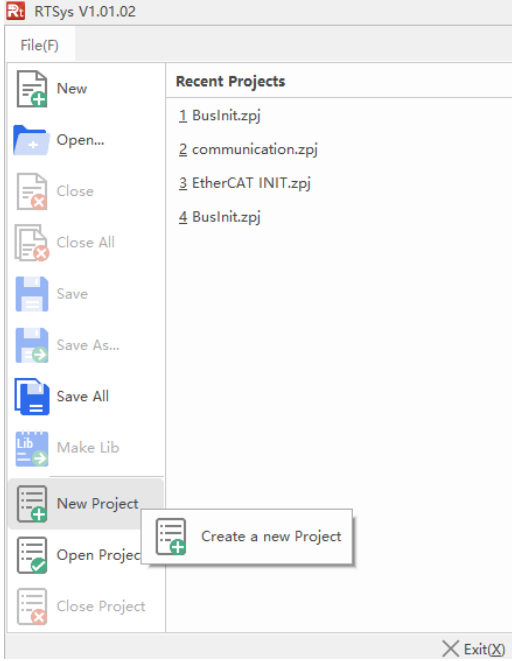
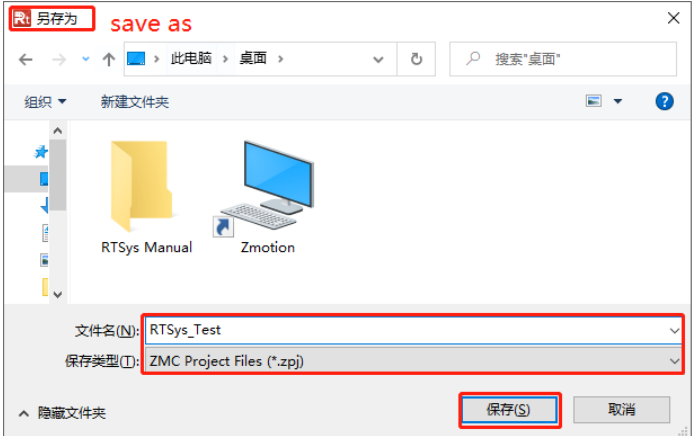
In RTSys, there are 4 programming languages for motion control development, Basic, PLC, HMI and C language, they can run multi-tasks among them, especially for Basic, multi-task running can be achieved separately, hybrid programming is also OK with PLC, HMI and C language.

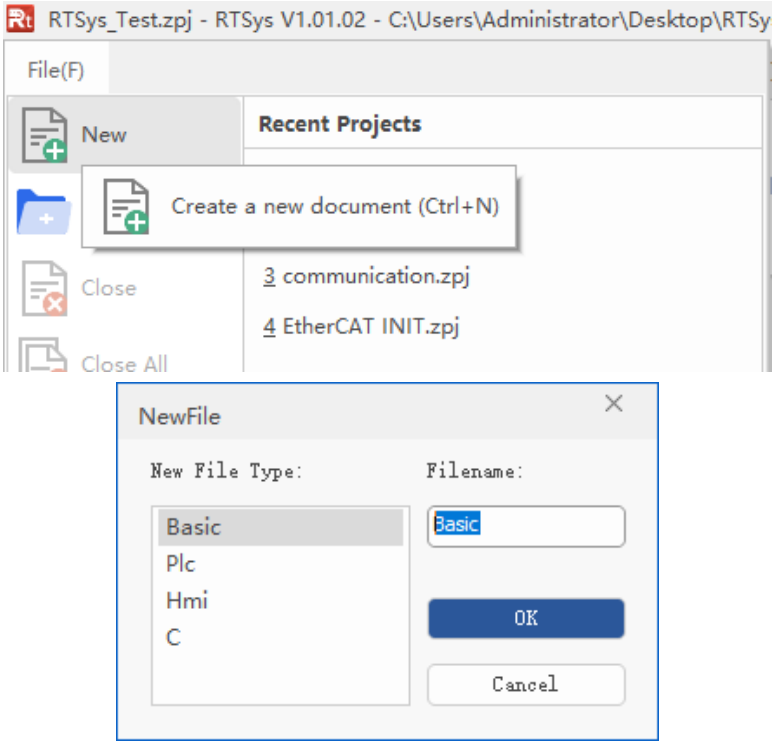
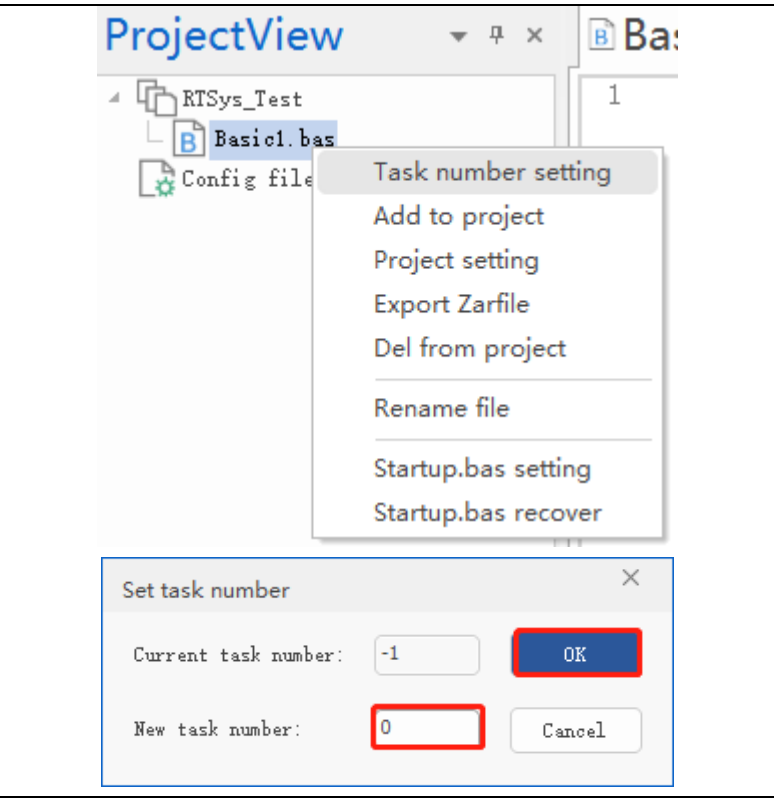
RTSys Downloading Address: https://www.zmotionglobal.com/pro_info_282.html

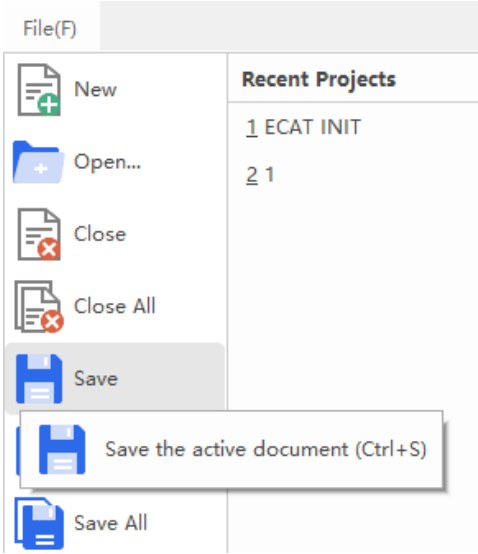
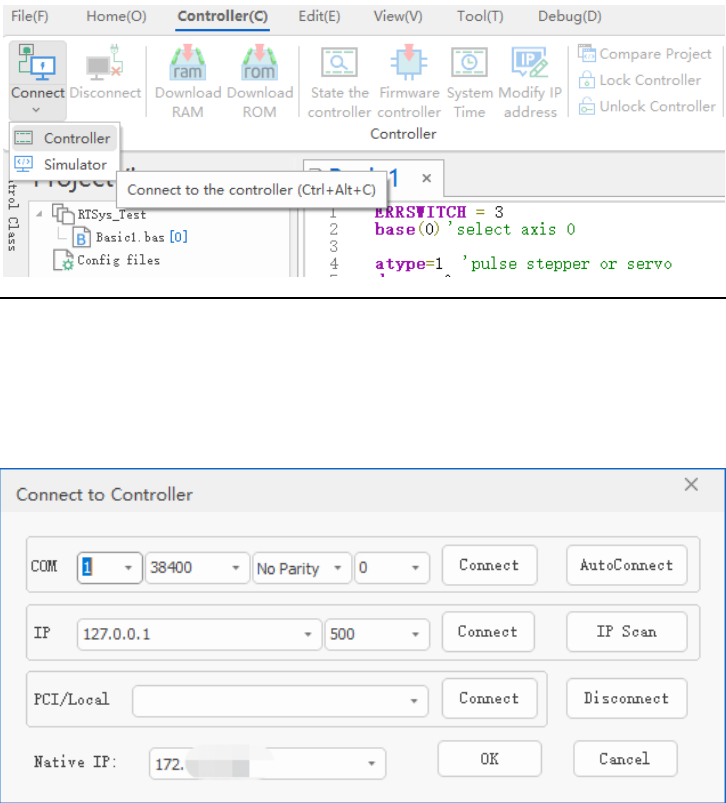
And related manuals can be found in “Download”:

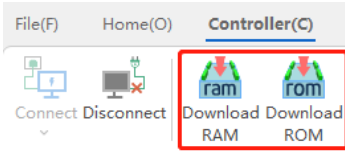
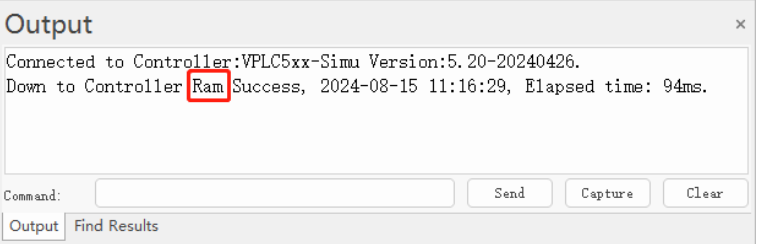
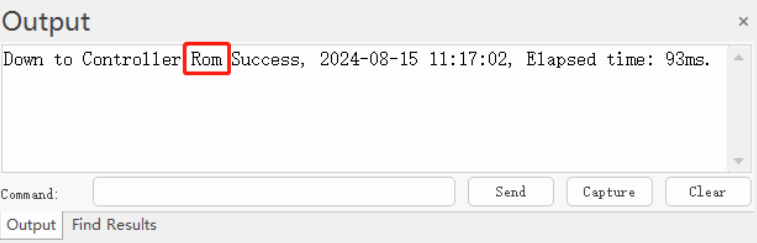
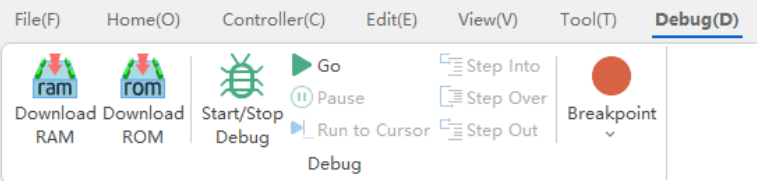
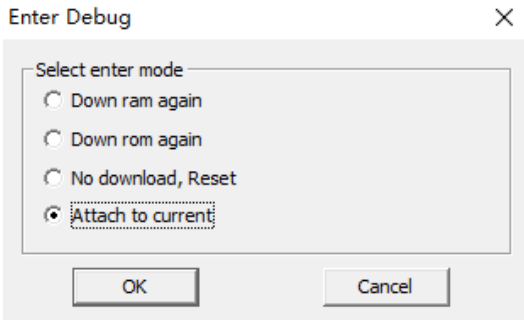
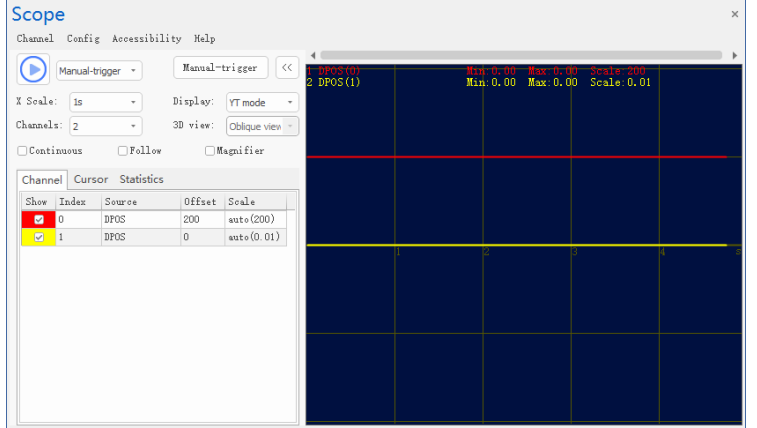
Features	Parameters	System Architecture	Download	
Name	Version No	Format	Size	Download
RTSys Development Software	V1.2.02	RAR	148MB	Download
RTSys User Manual V1.2.0	V1.2.0	PDF	5.33MB	Download
RTBasic Programming Manual	V1.1.0	PDF	18.3MB	Download
RTHMI Programming Manual	V1.2.0	PDF	7.23MB	Download
Quick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

Step	Operations	Display Interface
1	Switch the Language: “Language” – “English”, then there will pop up one window,	

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

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

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

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

Notes:

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

4.2. Upgrade Controller Frimware

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.

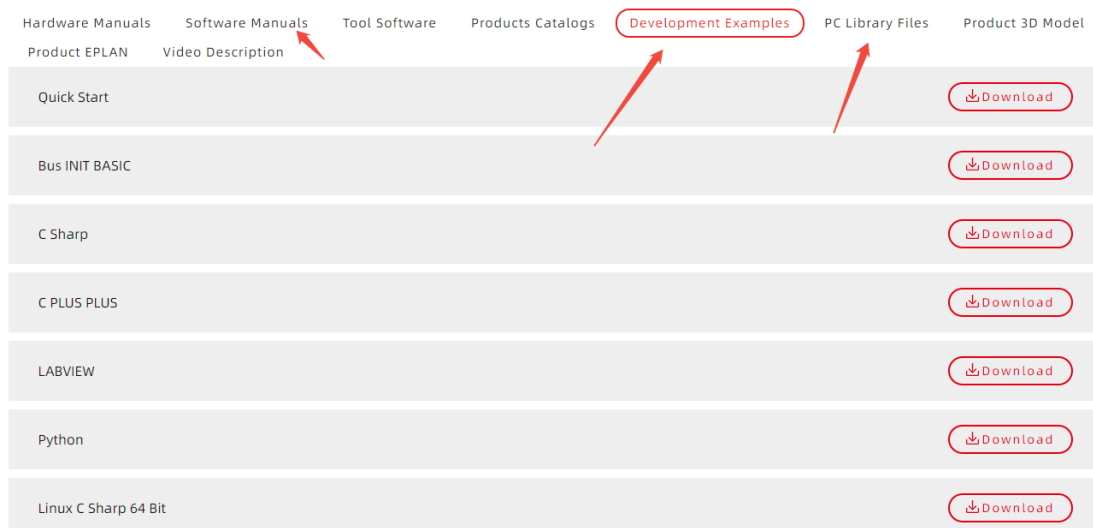
4.3. Program in Host-Computer by PC Languages

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

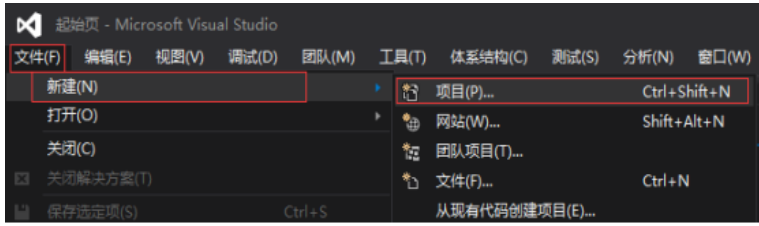
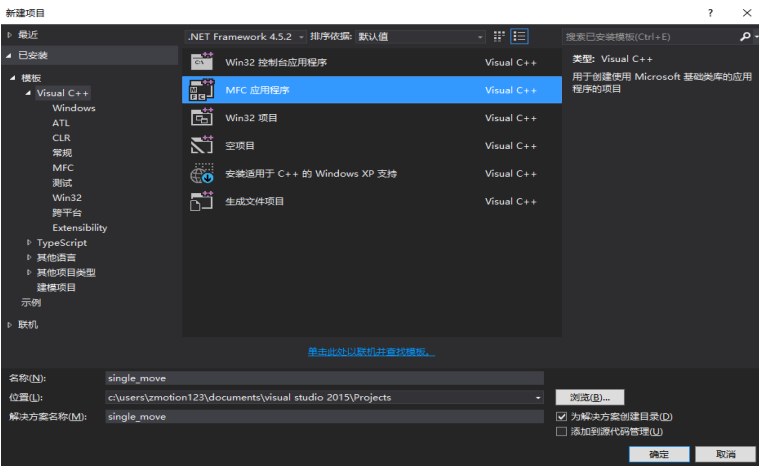
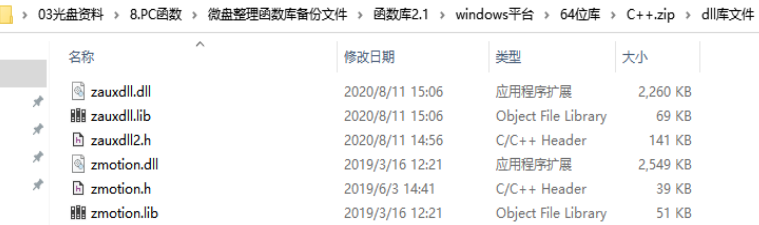


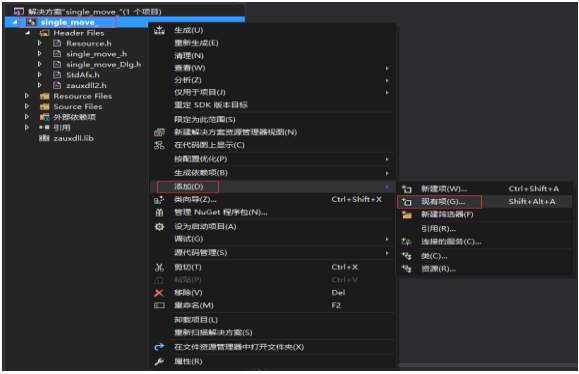
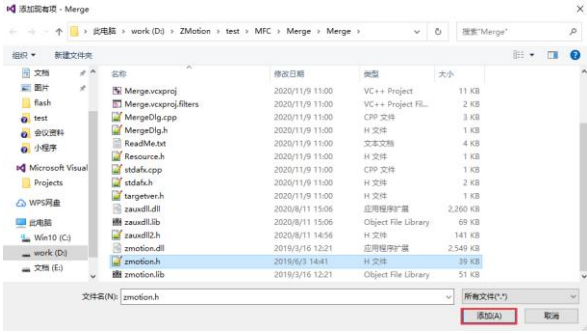
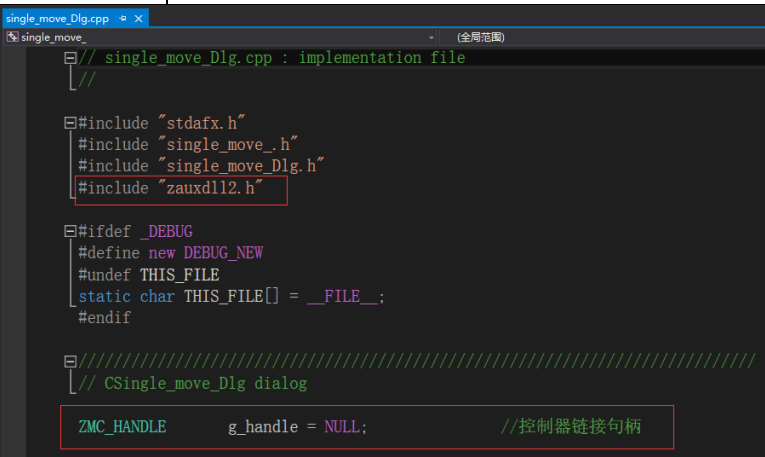
The program developed using the PC software cannot be downloaded to the controller, and it is connected to the controller through the dll dynamic library. The dll library needs to be added to the header file and declared during development.

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



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

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

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

Chapter V 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.

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	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

5.2. Common Problems & Solutions

Problems	Suggestions
Motor does not rotate.	6. Check whether the ATYPE of the controller is correct. 7. Check whether hardware position limit, software position limit, alarm signal work, and whether axis states are normal. 8. Check whether motor is enabled successfully. 9. Confirm whether pulse amount UNITS and speed values are suitable. If there is the encoder feedback, check whether MPOS changes. 10. Check whether pulse mode and pulse mode of drive are matched. 11. Check whether alarm is produced on motion controller station or drive station. 12. Check whether the wiring is correct. 13. Confirm whether controller sends pulses normally.

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

	ends. 2. Check the master-slave configuration, communication speed configuration, etc. 3. Check the DIP switch to see if there are multiple expansion modules with the same ID. 4. Use twisted-pair cables, ground the shielding layer, and use dual power supplies for severe interference (the main power supply of the expansion module and the IO power supply are separately powered)
Fail to connect controller to PC through net port.	1. Check IP address of PC, it needs to be at the same segment with controller IP address. 2. Check controller IP address, it can be checked and captured after connection through serial port. 3. When net port led is off, please check wiring. 4. Check whether controller power led POWER and running indicator led RUN are ON normally. 5. Check whether the cable is good quality, change one better cable to try again. 6. Check whether controller IP conflicts with other devices. 7. Check whether controller net port channel ETH are all occupied by other devices, disconnect to other devices, then try again. 8. When there are multiple net cards, don't use other net cards, or change one computer to connect again. 9. Check PC firewall setting. 10. Use "Packet Internet Groper" tool (Ping), check whether controller can be Ping, if it can't, please check physical interface or net cable. 11. Check IP address and MAC address through arp-a.
RTSys reports 201 error code	1. 201 means the sub-module doesn't exist, no ZMIO310 submodule is connected to ZMC432 slave expansion interface. 2. Set ZMIO_OFFSET as negative value / the value that is more than IN & OUT starting No. (must be the

	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.