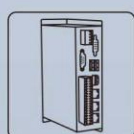


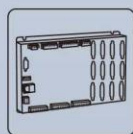
Vertical Motion Controller

ZMC432M

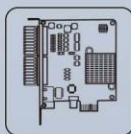
This manual is for ZMC432M, ZMC432ML5, ZMC432ML24.



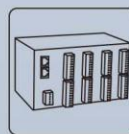
Vision Motion
Controller



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

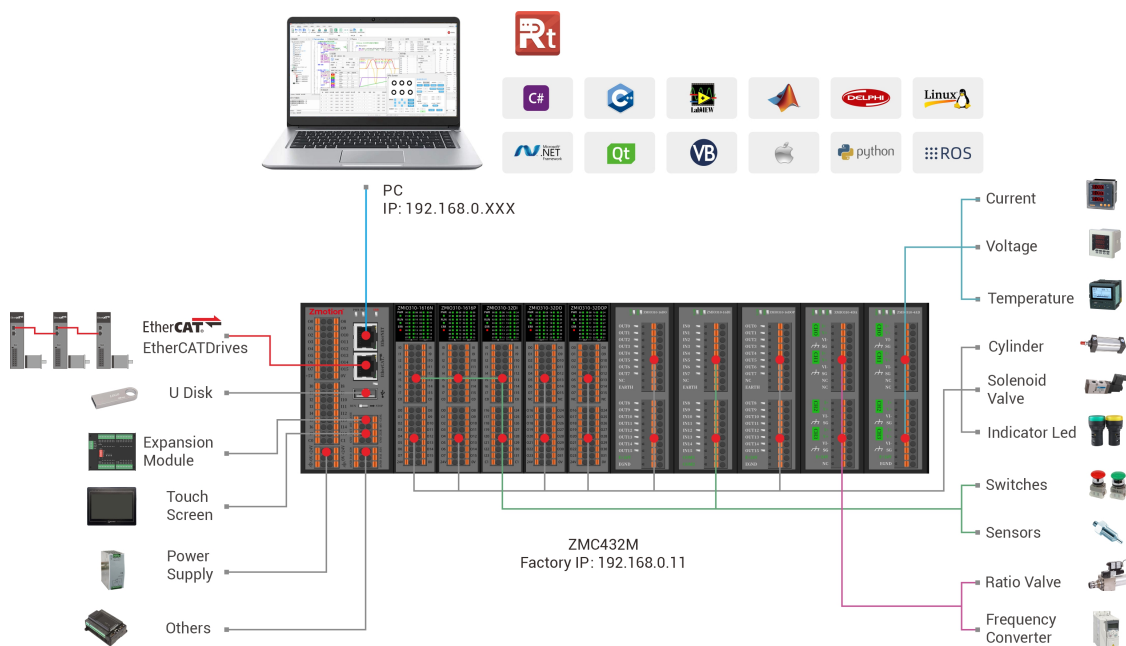
Chapter I Production Information.....	5
1.1. Product Information	5
1.2. Interface Introduction.....	6
1.3. Specification Model.....	8
1.4. Nameplate & Model	9
1.5. Hardware Installment	10
1.6. Environment Requirements	11
Chapter II Hardware Interface	12
2.1. Power IN	12
2.2. DI: General Digital Inputs.....	12
Specification	13
Wiring	14
Usage Method	16
2.3. DO: Digital Outputs	17
Specification (ZMC432M)	17
Specification (ZMC432ML Series)	18
Wiring	18
Usage Method	20
2.4. AD / DA Analog Input & Output	21
Specification	21
Wiring	21
Usage Method	22
2.5. Communication Interfaces: CAN / RS485 / RS232.....	22
Specification	23
Wiring	23
Usage Method	24
2.6. U Disk.....	25
Specification	26
Usage Method	26
2.7. RUN / STOP DIP Switch.....	26
2.8. ETHERNET	27
Specification	27
Wiring	28
Usage Method	28
2.9. EtherCAT Bus Interface	29
Specification	29
Wiring	29
Usage Method	30
Chapter III Local Expansion.....	31
3.1. Local Configuration	31

3.2.	Configuration Functions.....	32
	ZMIO310 I/O Offset Configuration.....	32
	ZMIO310 AIO Offset Configuration.....	34
	Analog Range & Channel States Set / Get.....	35
Chapter IV	Expansion Module	37
4.1.	CAN Expansion	37
	Wiring	37
	Resources Mapping	38
4.2.	EtherCAT Expansion	41
	Wiring	41
	Resources Mapping	43
Chapter V	Programming.....	44
5.1.	Program in RTSys Software	44
5.2.	Upgrade Controller Firmware	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

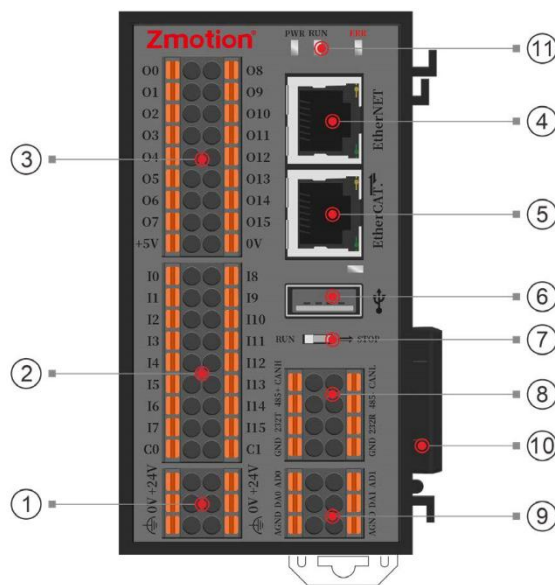
ZMC432M high-performance multi-axis motion controller is a stand-alone motion controller that is compatible EtherCAT with pulse type, which supports max 32 axes for complex continuous trajectory control.



- ✚ 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 Functions:** HW hardware comparison output, high-speed latch, PWM, etc., and for special models, it supports 5V/ 24V laser specialized PWM output (ZMC432ML series).
- ✚ One UDISK interface is equipped, for writing and reading files, updating the program, easy to do remote support and maintain.
- ✚ It can be programmed in multi-file and multi-task by BASIC, also, PC program and controller inner RTBasic program can work at the same time.

ZMC4 series high-performance multi-axis motion controllers can be applied in robots (SCARA, DELTA, 6 DOF), 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.	Interfaces	Descriptions
①	Power	connect to 24V DC power supply
②	DI	- NPN / PNP; - IN0-15; - some inputs can be set as single-ended encoder, or used to do position latch.
③	DO	- NPN type; - OUT0-15; - some outputs can be used as single-ended pulse axes, or can be used to do PWM, HW.
④	EtherNET	- connect to upper device (PC, etc.); - MODBUS_TCP protocol; - it can be extended by one industrial switch.

⑤	EtherCAT	- connect to EtherCAT drive; - or connect to EtherCAT expansion modules.
⑥	UDISK	connect to USB external devices (mouse, keyboard, U disk, etc.) for program upgrading, data importing & exporting.
⑦	RUN/ STOP	- Switch run / stop modes; - Restore IP to factory settings in STOP mode.
⑧	CAN	connect to CAN modules to expand more resources;
	RS485	connect to upper computer;
	RS232	MODBUS_RTU.
⑨	AD / DA	- single-ended type; 12-bit, 0-10V.
⑩	Local Slave Expansion	connect to expansion submodules, please note it doesn't support "hot-swap".
⑪	Controller Status Led	PWR Led: ON when power is connected.
		RUN Led: ON when it runs normally.
		ERR Led: 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.

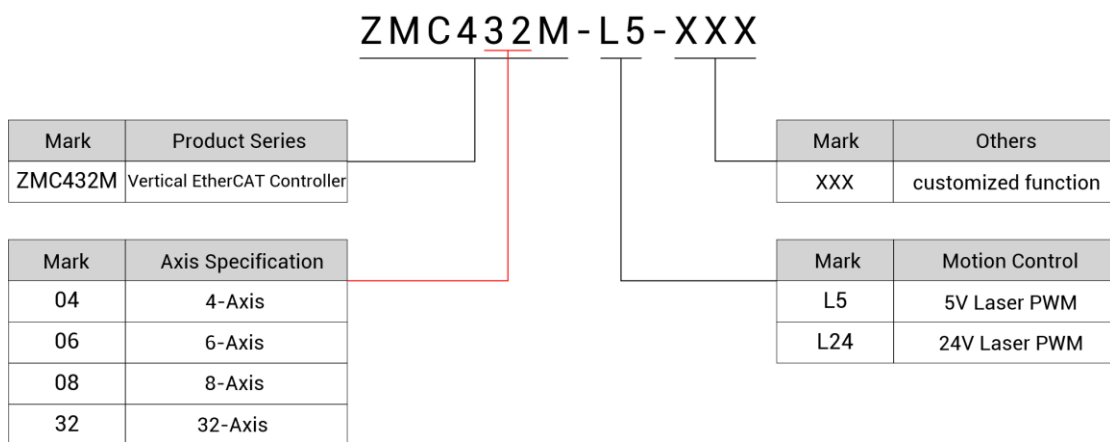
1.3. Specification Model

ZMC432M / ZMC432ML5 / ZMC432ML24	
Base Descriptions	
Motor Axes	32
All Axes	32 (motor axis + virtual axis)
Program Space	128MByte
Dimensions (mm)	95*108*46
Axis Type	
EtherCAT Axis	√; 1 EtherCAT bus axis
Single-Ended Encoder in DI	4
Single-Ended Pulse in DO	8
DI & DO	
DI	16
DO	16
Max Extended DI	≤4096
Max Extended DO	≤4096
AD & DA	
AD	2 (0-10V, 12-bit)
DA	2 (0-10V, 12-bit)
Max Extended AD	≤512
Max Extended DA	≤512
Communication Interfaces	
EtherNET	1
EtherCAT	1
CAN	1
RS232	1
RS485	1
UDISK	1
Basic Motion Control Functions	
Point Motion	√
Electronic Cam	
Linear Interpolation	
Circular Interpolation	

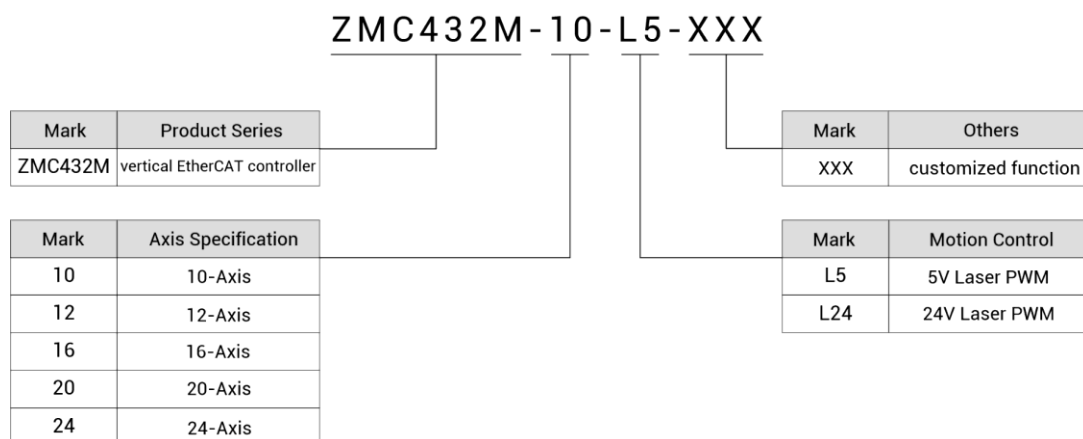
Continuous Interpolation	
SCARA Robotic Arm	
Advanced & Special Motion Control Functions	
High-Speed Latch	4
Hardware Comparison Output HW	4
General PWM	ZMC432M Series: 4 (OUT0-3) ZMC432ML5/L24 Series: 2 (OUT2-3)
Laser PWM	ZMC432M Series: 0 ZMC432ML5 Series: 2 (5V, OUT0-1) ZMC432ML24 Series: 2 (24V, OUT0-1)

1.4. Nameplate & Model

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



■ Nameplate Information (10-24 axes specifications)



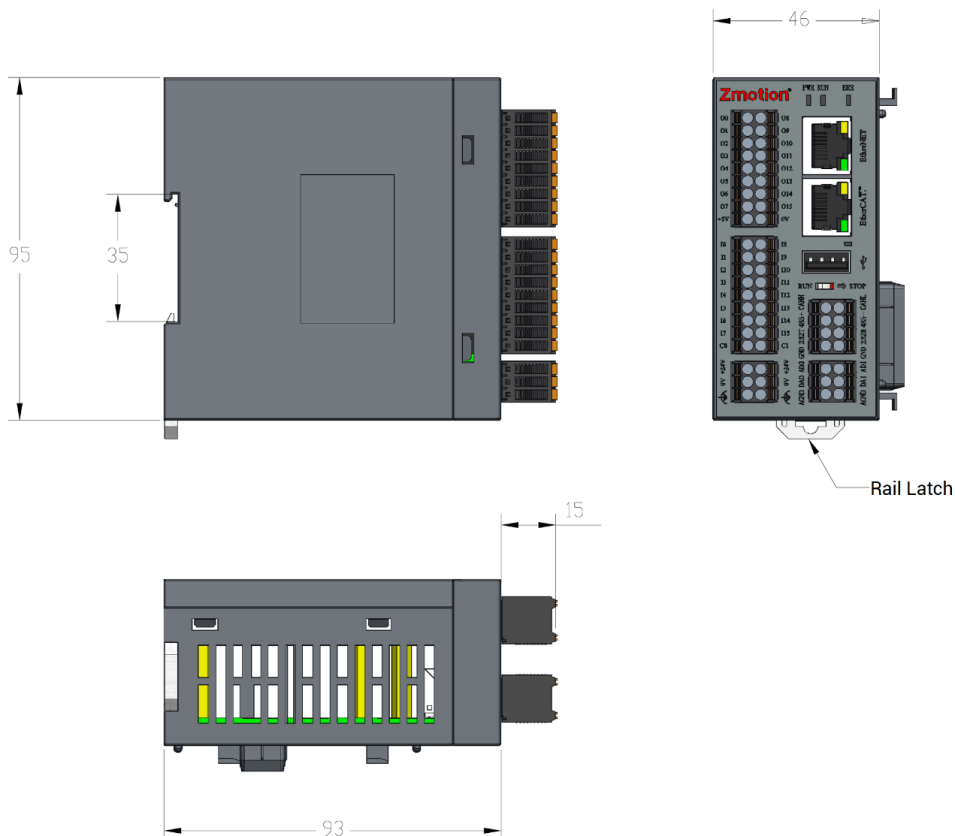
■ Product Models

No.	Models	Description
1	ZMC406M	6-axis EtherCAT control, support HW, not support Laser PWM.
2	ZMC408M	8-axis EtherCAT control, support HW, not support Laser PWM.
2	ZMC432M-16	16-axis EtherCAT control, support HW, not support Laser PWM.
4	ZMC432M	32-axis EtherCAT control, support HW, not support Laser PWM.
5	ZMC432ML5	32-axis EtherCAT control, support HW, support 5V Laser PWM.
6	ZMC432ML24	32-axis EtherCAT control, support HW, support 24V Laser PWM.

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

1.5. 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.

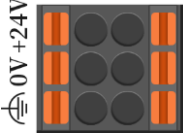
1.6. Environment Requirements

Item		Parameters
Work Temperature ^[1]		-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

Note [1]: if the environment temperature is too high or too low, please operate fan cooling (fan) or heat Preservation, to ensure controller can work normally and stably.

Chapter II Hardware Interface

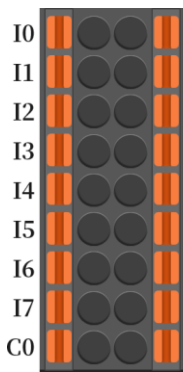
2.1. Power IN

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	√
Overcurrent Protection	√
Isolation Power	√
Cable Type	Recommend "1.0 mm ² copper conductor cable"

2.2. DI: 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

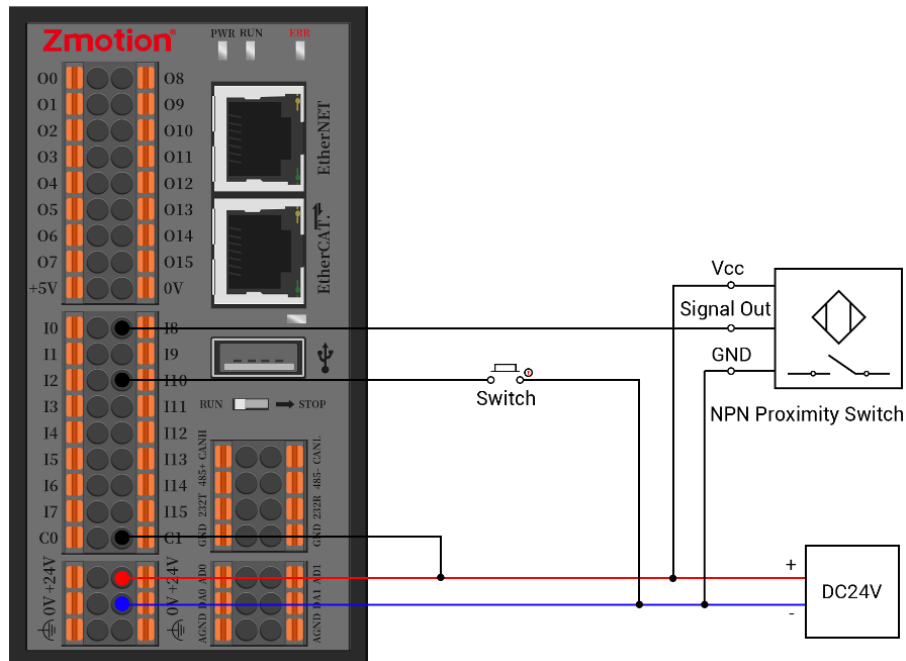
	I7		Input 7	/	EA1
	C0	COM0: IN (0-7) public end to determine the input type			
	I8	high-speed input	Input 8	/	EB1
	I9		Input 9	/	EZ1
	I10		Input 10	/	EA2
	I11		Input 11	/	EB2
	I12		Input 12	/	EZ1
	I13		Input 13	/	EA3
	I14	NPN / PNP	Input 14	/	EB3
	I15		Input 15	/	EZ3
	C1	COM0: IN (8-15) public end to determine input type			

Specification

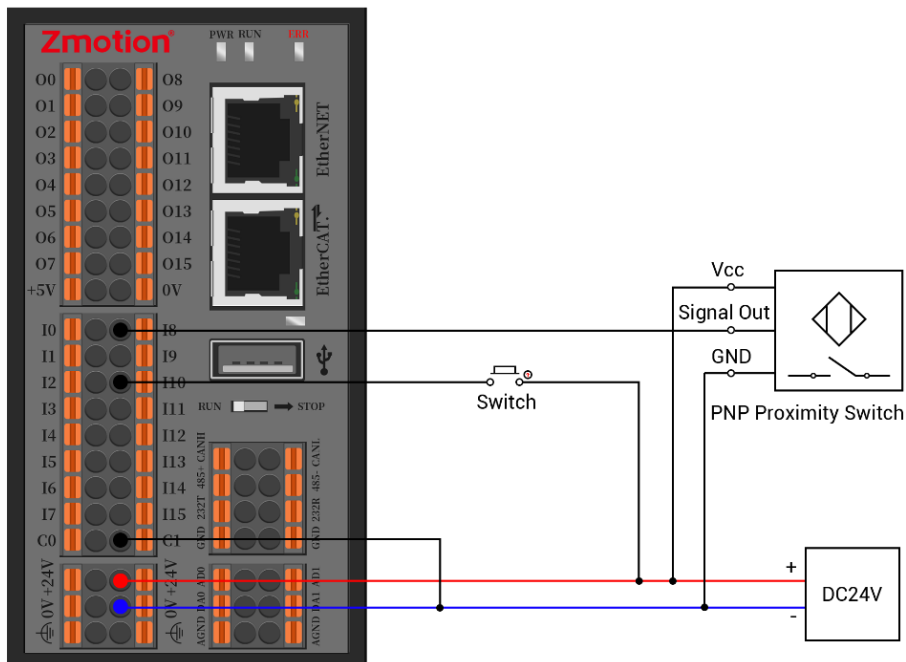
Item	High-Speed Input (IN0-15)	
Input mode	NPN, IN is triggered when there is low level	PNP, IN is triggered when there is low level
Frequency	≤400kHz	≤400kHz
Impedance	4.7KΩ	4.7KΩ
Voltage level	0V/24V	0V/24V
The voltage to open	<15V	<15V
The voltage to close	>15.1V	>15.1V
Minimal current	-2.3mA (negative)	-2.3mA (negative)
Max current	-7.5mA (negative)	-7.5mA (negative)
Communication Isolation mode	√	√

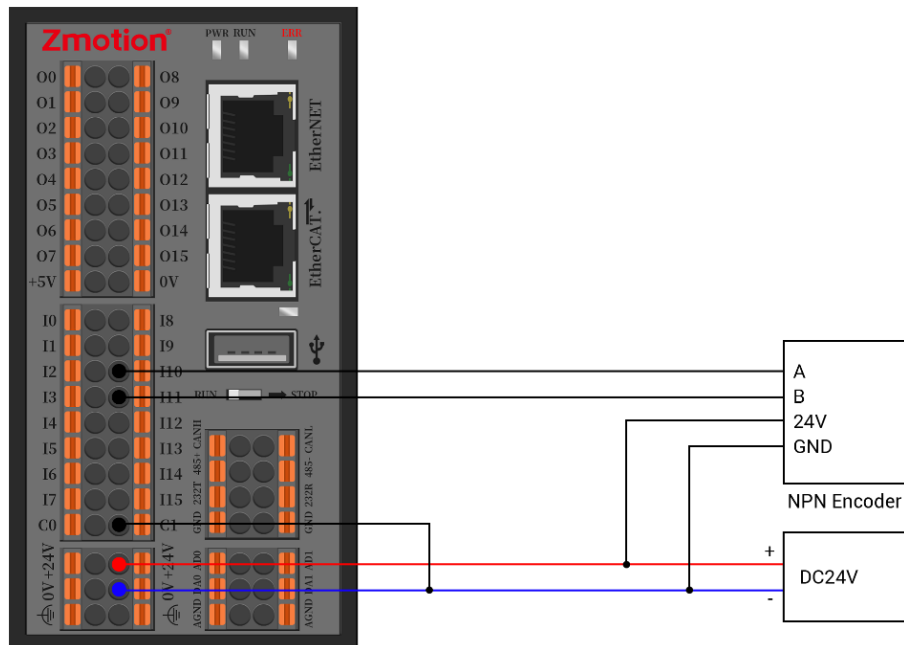
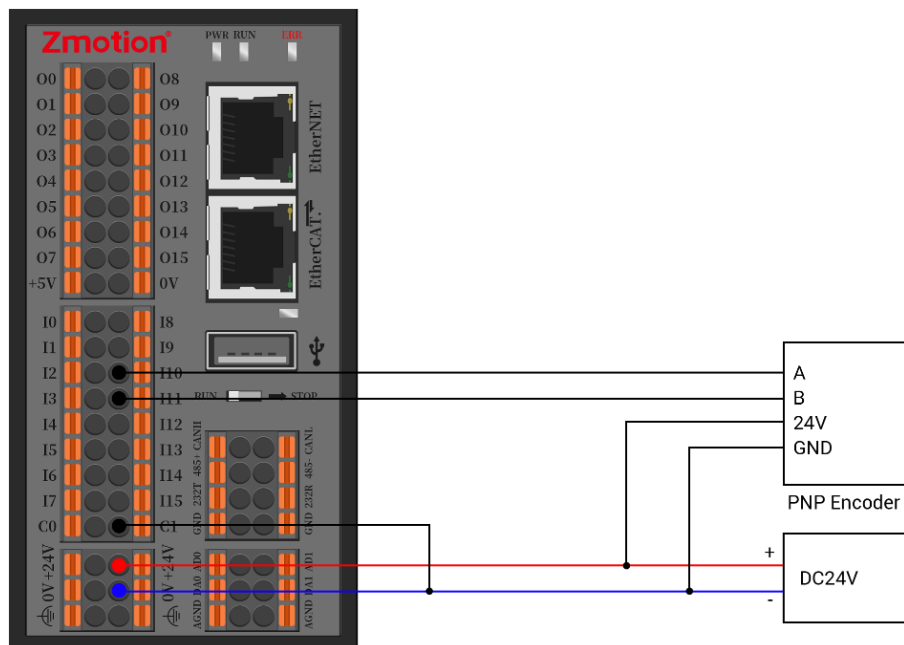
Wiring

NPN Wiring:



PNP Wiring:



NPN Single-Ended Encoder Wiring (example: EA0, EB0, EZ0):**PNP Single-Ended Encoder Wiring (example: EA0, EB0, EZ0):****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.

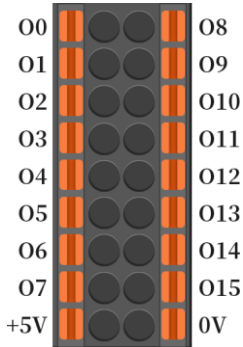
- It is recommended to use the same power supply for both the load and the controller, otherwise, the negative terminals of two power supplies should be connected.
- During on-site wiring, maintain a distance from high-voltage power lines, preferably 30cm or more.
- Ensure that the controller is properly grounded, and the chassis ground should be connected to a standard factory ground rod.

Usage Method

- (1) Please follow the above wiring instructions to wiring correctly, then connect the controller to [RTSys](#) through ethernet / serial port.
- (2) IN states can be read directly through "IN" command, also, it can be read through *"RTSys / Tool / IN"*.
- (3) Latch function can be set and triggered by **"REGIST"**, **"REV_IN"**, **"DATUM_IN"**.
- (4) Axis position limit signals / origin signals can be set by **"FWD_IN"**, **"REV_IN"**, **"DATUM_IN"** commands.
- (5) IN14-15 can be set as encoder axis by setting ATYPE as 3 or 6. When ATYPE = 0, they are still used as general digital inputs.
- (6) For above and other commands, please refer to ["RTBASIC Programming Manual"](#), or check in *RTSys / Help / RTBasic*.
- (7) Codes of "encoder-axis mapping":

'encoder axis mapping (use EA0, EB0)	
BASE (10)	'the new (remapping) axis No., don't be same as other axis No.
ATYPE = 0	
BASE (6)	'the original axis No.
ATYPE = 0	'set axis type as 0
AXIS_ADDRESS (10) = (-1<<16) + 6	'bind local axis 6 with axis 10
BASE (10)	
ATYPE = 3	'set axis type as encoder axis for axis 10

2.3. D0: Digital Outputs

Terminal	Name	Type	Function 1	2	3	4
	O0	NPN, high-speed outputs, max is 500mA.	output 0	PWM0	HW_0	PUL0
	O1		output 1	PWM1	HW_1	DIR0
	O2		output 2	PWM2	HW_2	PUL1
	O3		output 3	PWM3	HW_3	DIR1
	O4		output 4	/	/	PUL2
	O5		output 5	/	/	DIR2
	O6		output 6	/	/	PUL3
	O7		output 7	/	/	DIR3
	O8		output 8	/	/	PUL4
	O9		output 9	/	/	DIR4
	O10		output 10	/	/	PUL5
	O11		output 11	/	/	DIR5
	O12		output 12	/	/	PUL6
	O13		output 13	/	/	DIR6
	O14		output 14	/	/	PUL7
	O15	output 15	/	/	DIR7	
	+5V	5V power output +, max is 300mA				
	0V	5V power output -, digital output public end				

Notes:

- ✧ 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.
- ✧ Functions can't be reused (please choose one function).

Specification (ZMC432M)

Item	High Speed Output (OUT0-15)
Output mode	NPN Type
Frequency	≤400kHz
Voltage	≤24V

Output Current	≤500mA
Overcurrent protection	✓
Communication Isolation	✓

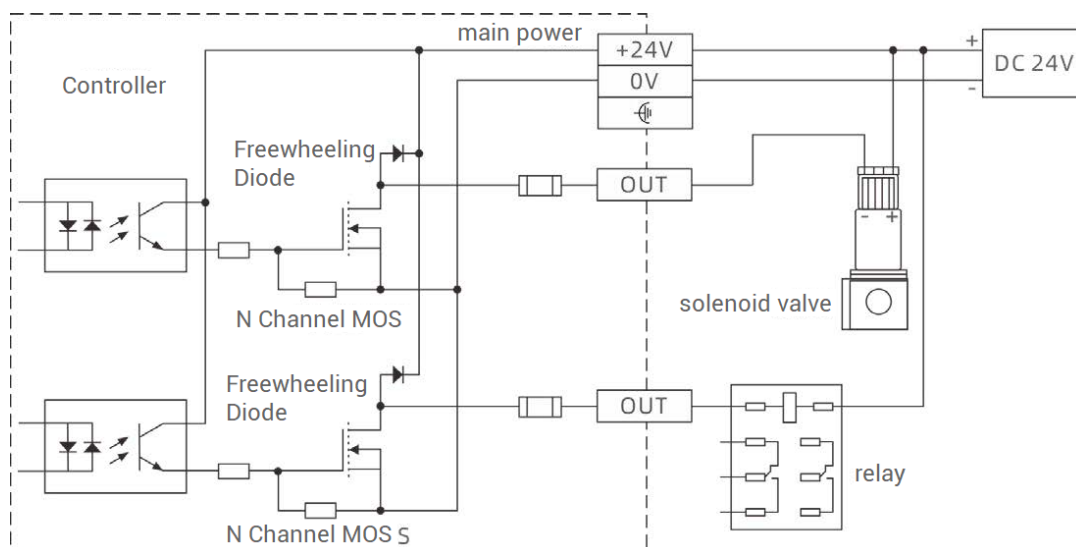
Specification (ZMC432ML Series)

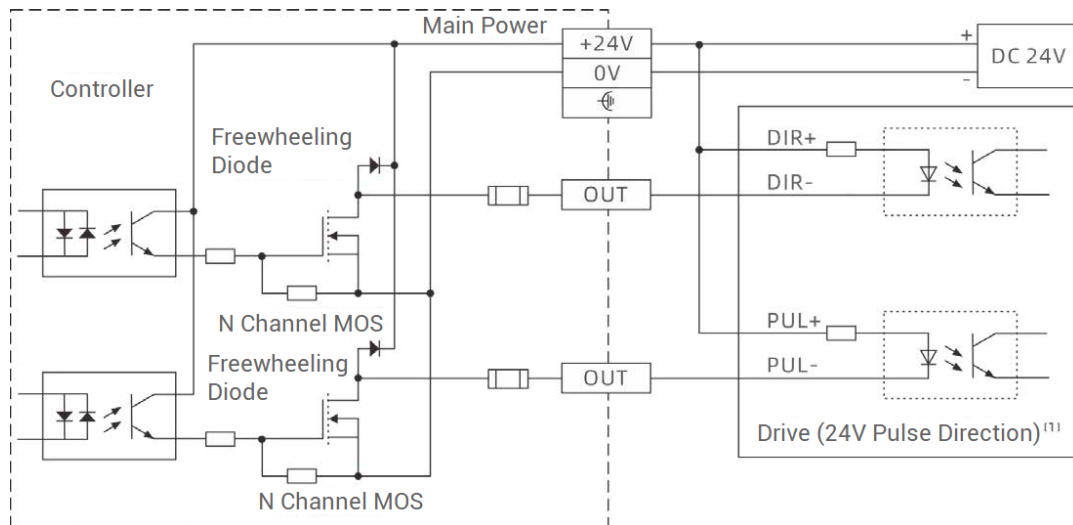
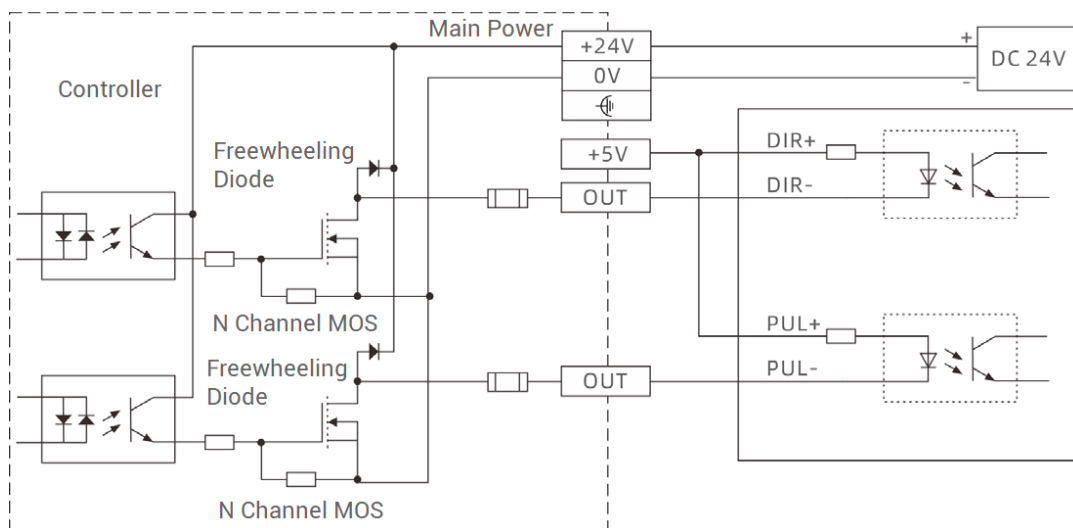
Item	Laser PWM Output (OUT0-1)	High Speed Output (OUT2-15)
Output mode	Push-pull type	NPN type
Frequency	≤1MHz	≤400kHz
Voltage	≤5V / ≤24V (optional)	≤24V
Output Current	≤50mA	≤500mA
Overcurrent protection	✓	✓
Communication Isolation	✓	✓

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

General Output Wiring:



Pulse Axis Wiring (24V) (example: PUL2, DIR2):**Pulse Axis Wiring (5V) (example: PUL2, DIR2):****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.

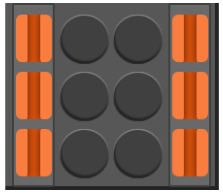
- Ensure that the controller is properly grounded, and the chassis ground should be connected to a standard factory ground rod.

Usage Method

- (1) After correct wiring & powered-on, please connect controller to [RTSys](#) by EtherNET or COM ports.
- (2) OUT can be set directly through "OP" command, also, it can be operated by *"RTSys / Tool / OP"*.
- (3) PWM function can be set by **"PWM_FREQ" (frequency)**, **"PWM_DUTY"(duty cycle)**.
- (4) Hardware comparison output can be set by **"HW_PSWITCH2"** command.
- (5) Set ATYPE of IN8-15 as 1 to achieve pulse axis function. If ATYPE = 0, they are still used as general digital inputs.
 - 1) please follow the above wiring instructions to wiring correctly, then connect the controller to RTSys through ethernet / serial port.
 - 2) set axis basic parameters via commands of BASE, ATYPE, UNITS, SPEED, ACCEL, DECEL, etc.; and do axis remapping through AXIS_ADDRESS; enable through AXIS_ENABLE; do linear motion through MOVE.
 - 3) some parameters can be set and checked in "RTSys / View / Axis Parameter" window; at the same time, motions also can be operated in "RTSys / Tool / Manual" window
- (6) For above and other commands, please refer to ["RTBASIC Programming Manual"](#), or check in *RTSys / Help / RTBasic*.
- (7) Codes of "pulse-axis mapping":

'pulse axis mapping (use PUL2, DIR2)	
BASE (16)	'the new (remapping) axis No., don't be same as other axis No.
ATYPE = 0	
BASE (2)	'the original axis No.
ATYPE = 0 'set axis type as 0	
AXIS_ADDRESS (16) = (-1<<16) + 2 'remapping the physical axis 2 as axis 16	
BASE (16)	
ATYPE = 1 'set axis type as pulse axis for axis 16	

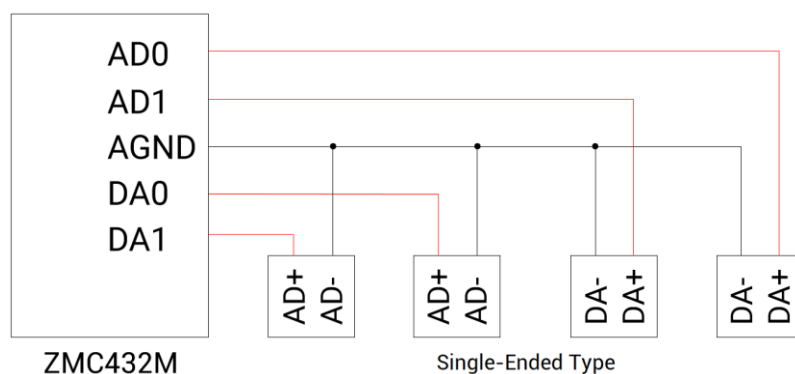
2.4. AD / DA Analog Input & Output

Terminal	Name	Function
	AD0	Analog Input 0
	AD1	Analog Input 1
	DA0	Analog Output 0
	DA1	Analog Output 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	>40K Ω (input impedance)	>10K Ω (load requirement)
Signal Type	Single-ended	Single-ended
Cable Length	Recommend <5m	Recommend <5m

Wiring



Notes:

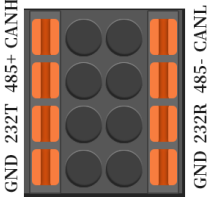
- The wiring for the analog is as shown in the diagram above. The load signal should be matched with impedance, otherwise, it will affect precision.

- 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.

Usage Method

- (1) After correct wiring & powered-on, please connect controller to [RTSys](#) by EtherNET or COM ports.
- (2) AD states can be read directly through "AIN" command, and DA can be set directly through "AOUT", also, they can be checked and operated in *"RTSys / Tool / AD/DA"*.
- (3) Please refer to ["RTBASIC Programming Manual"](#), or check in *RTSys / Help / RTBasic*.

2.5. Communication Interfaces: CAN / RS485 / RS232

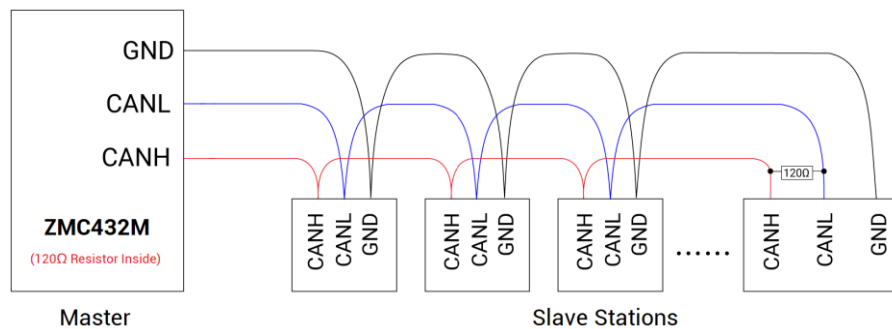
Terminal	Name	Function
 Isolated Communication	CANH	CAN communication signal cable side H
	CANL	CAN communication signal cable side L
	485+ / A	RS485 communication signal cable + / side A
	485- / B	RS485 communication signal cable - / side B
	232T	RS232 communication signal side TX
	232R	RS232 communication signal side RX
	GND	Communication public end

Specification

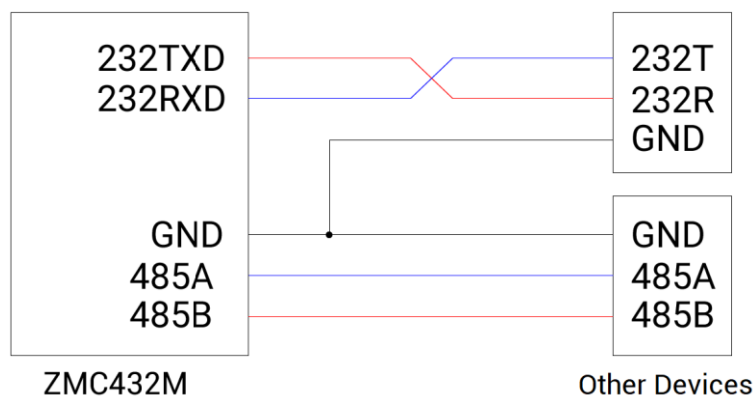
Item	CAN	RS485 (port1)	RS232 (port0)
Communication Rate	≤1Mbps	≤115200bps	≤115200bps
Terminal Resistor	120Ω (inside in controller)	/	/
Topological Structure	Daisy Chain Topology		One to One
The number of nodes can be extended	≤16	≤127	1
Wiring Length	<30m (500kbps)	<30m	<5m
Communication Isolation	√		

Wiring

- 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 Method

- (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) Set controller as CAN master station:

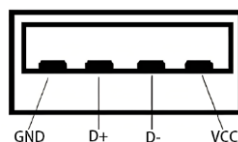
- a) Use "**CANIO_ADDRESS**" to set master station "address" and "velocity".
 - b) Use "**CANIO_ENABLE**" to enable / disable CAN master station function.
 - c) In "*RTSys>Controller>Controller Status>Communication Config*" interface, you can check communication parameters.
 - d) In "*RTSys>Controller>Controller 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:

- 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) Please refer to ["RTBASIC Programming Manual"](#), or check in *RTSys / Help / RTBasic*.

2.6. U Disk

The ZMC432M 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, Z3P file executing, etc.



Specification

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

Usage Method

- (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 (using FILE series instructions).
- (3) Please refer to "[RTBASIC Programming Manual](#)", or check in *RTSys / Help / RTBasic*.

2.7. RUN / STOP DIP Switch

Terminal	Name	Description
RUN  STOP	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.**

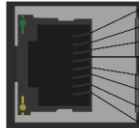
- ✚ Modes Switch Description:

***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
	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 (-)	Cable Length	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:

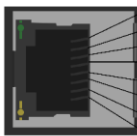
- Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.
- During on-site wiring, maintain a distance from high-voltage power lines, preferably 30 cm or more.
- Ensure that the controller is properly grounded, and the chassis ground should be connected to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

Usage Method

- (1) Please follow the above wiring instructions to wiring correctly, then connect the controller to [RTSys](#) through ethernet / serial port.
- (2) "IP_ADDRESS" can be used to 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) Please refer to "[RTBASIC Programming Manual](#)", or check in *RTSys / Help / RTBasic*.

2.9. EtherCAT Bus Interface

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 (+)		
	4	NC	Reserved	Refresh Ratio	Max: 500us
	5	NC	Reserved		
	6	RX-	Receive signal (-)	Communication Cable	Shielded Cat 5e Cable
	7	NC	Reserved	Cable Length	Recommend <50m
8	NC	Reserved			

Wiring

(1) Controller EtherCAT can be connected to EtherCAT drives / other EtherCAT devices via one shielded cat 5e cable: connect to controller EtherCAT to slave EtherCAT IN, then connect slave EtherCAT OUT to next one EtherCAT IN.

(2) 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 Method

(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 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"*.

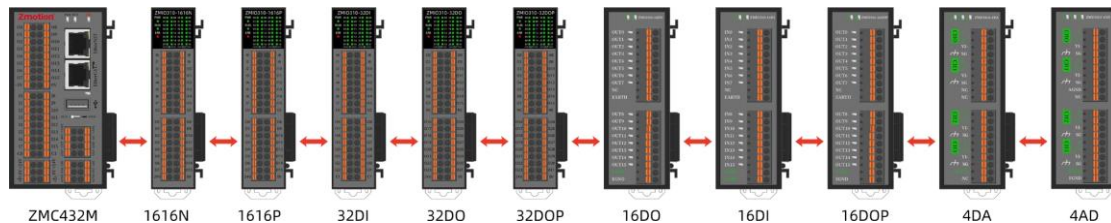
(5) Please refer to ["RTBASIC Programming Manual"](#), or check in *RTSys / Help / RTBasic*.

(6) How to configure EtherCAT axis in RTSys: [RTSys – EtherCAT Quick Configurations](#).

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.

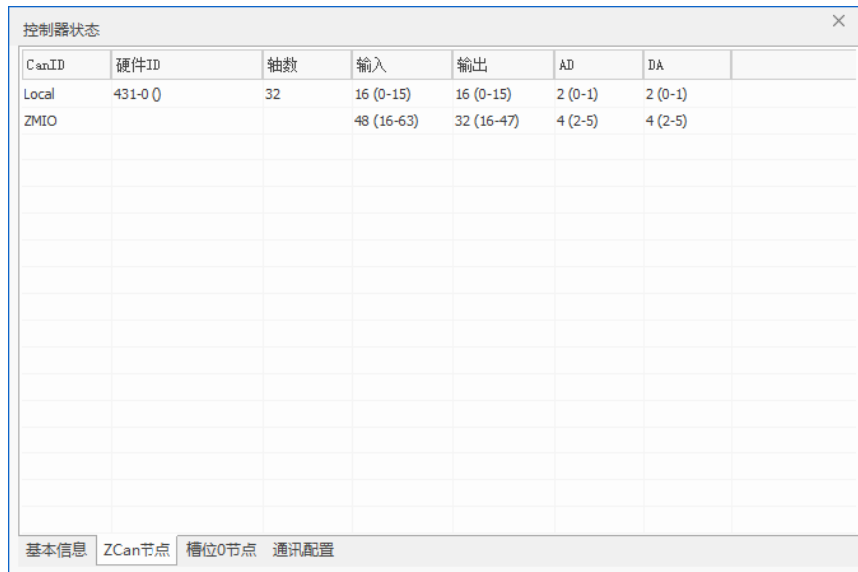
Step 2: align and insert the local expansion front-stage interface of the expansion sub-module into the local expansion slave interface of the ZMC432M controller (or another expansion sub-module).

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:



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 Functions

ZMIO310 extended configurations: I/O configuration & AIO configuration.

ZMIO310 I/O Offset Configuration

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

- 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".

[illegible][illegible]

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

[illegible][illegible]

--How to check itself ZMIO expansion situation--

Function Description	Used to check ZMC432M itself ZMIO expansion situation.				
Usage Syntax	Syntax 1: var = ZMIO_INFO (sel) Syntax 2: var = ZMIO_INFO (17, node)				
Parameter List	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		
Example	?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".

Analog Range & Channel States Set / Get

Function Description	Used to read / configure expansion sub-module's AD / DA channel switch state and range type.
Usage Syntax	Syntax 1: be read: var = ZMIO_CONFIG (sel, moduleid) Syntax 2: be written: ZMIO_CONFIG (sel, moduleid, value) ● sel: function No. ● moduleid: expansion submodule address ● value: the channel value / range type expansion submodule to be configured. For details, please refer to RTBASIC Programming Manual .

Example	<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 Expansion

For Zmotion, there are 3 types [CAN bus expansion modules](#): ZIO, ZAIIO, 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

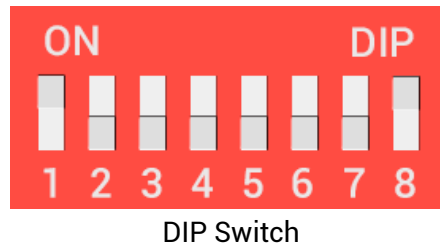


Notes:

- ✧ The controller and expansion modules need to share a main power supply, while the I/O power supplies of ZIO and ZMIO310-CAN need to be independently powered to achieve isolation.
- ✧ When connecting multiple expansion modules on CAN, a 120-ohm resistor is needed to be connected in parallel between the CANL & CANH, for the ZIO expansion module that is with 8-digit dialing codes, the terminal resistor can be realized by dialing the code (DIP).

Resources Mapping

A. DIP Switch



Generally, the ZCAN expansion module has an 8-code DIP switch to do communication configuration and resource mapping, dial ON to take effect, and the meaning of the DIP is as follows:

- ✚ 1-4: they are used for ZCAN expansion module IO address mapping, the corresponding value is 0-15.
- ✚ 5-6: CAN communication speed, corresponding value is 0-3, four different speeds are optional.
- ✚ 7: reserved.
- ✚ 8: 120-ohm resistor, dial ON means a 120-ohm resistor is connected between CANL and CANH.

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

Dial code 5-6 to select CAN communication speed, 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 ["2.4 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.

B. I/O 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.--

(0: OFF, 1: ON)

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

--AIO 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

C. AIO 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

Please refer to "[RTBASIC Programming Manual](#)", or check in *RTSys / Help / RTBasic*.

D. Extended Resources Checking

Connect the controller to RTSys at first, then in RTSys, enter "*controller > controller status > ZCanNode*" interface, you can know expansion module ID and corresponding mapping No. (below take ZMC432-V2 controller as the example).

[illegible]

4.2. EtherCAT Expansion

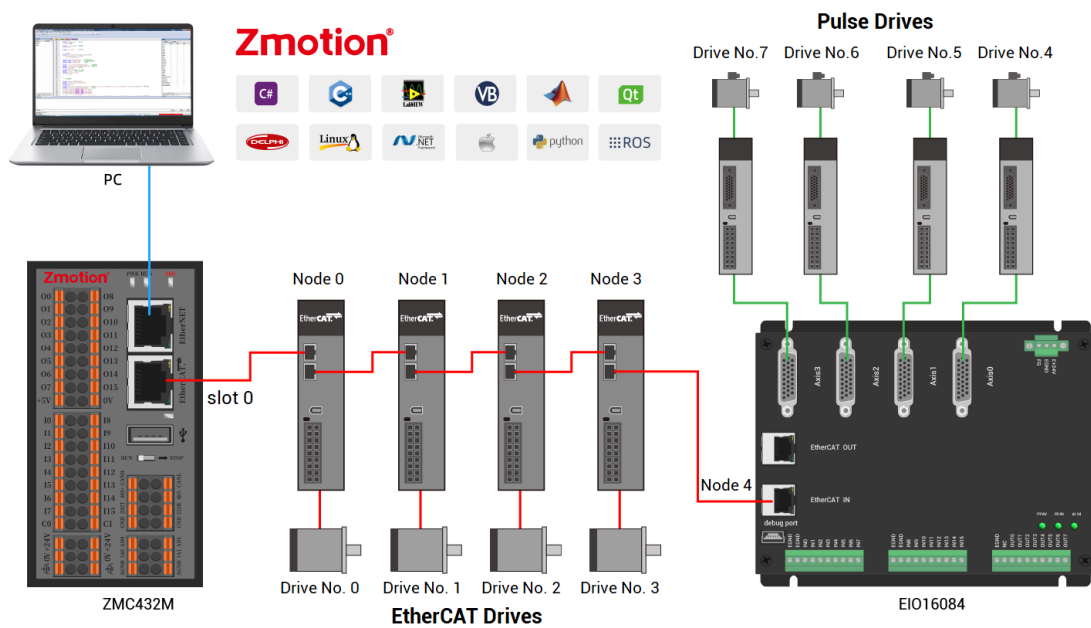
The [EI0 expansion modules](#) and [ZMIO310-ECAT](#) are expansion modules used to expand digital IO / analog IO / pulse axis by EtherCAT.

Wiring

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.

Resources Mapping

A. IO Mapping

For EtherCAT expansion module IO No., they are set by **NODE_IO** and **NODE_AIO**.

When IO mapping, first check the maximum IO No. 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 No. of the controller itself, both two will work at the same time, so the IO mapped No. must not be repeated in the entire control system.

Examples (IO & AIO):

`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

B. Axis Mapping

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

Axis mapping syntax:

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

Example:

`AXIS_ADDRESS(6)=(0<<16)+0+1` '1st drive on EtherCAT, drive No. 0, bound as axis 6

`AXIS_ADDRESS(7)=(0<<16)+1+1` '2nd drive on EtherCAT, drive No. 1, bound as axis 7

C. Extended Resources Checking

Connect the controller to RTSys at first, then in RTSys, enter **"controller > controller status > Slot"** interface, you can know expansion module ID and corresponding mapping No.

Chapter V Programming

5.1. Program in RTSys Software

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

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

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

And related manuals can be found in "Download":

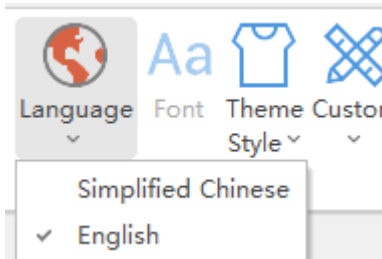
Features

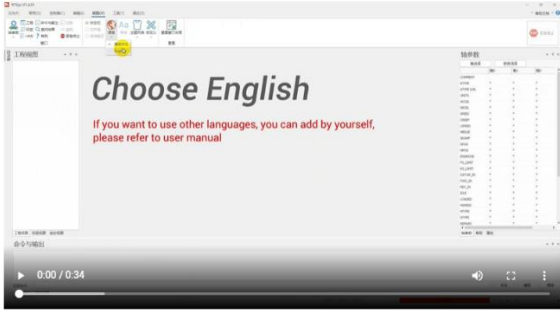
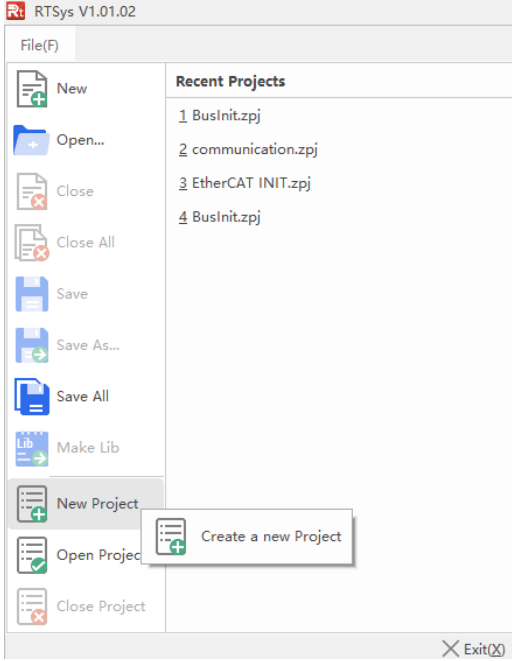
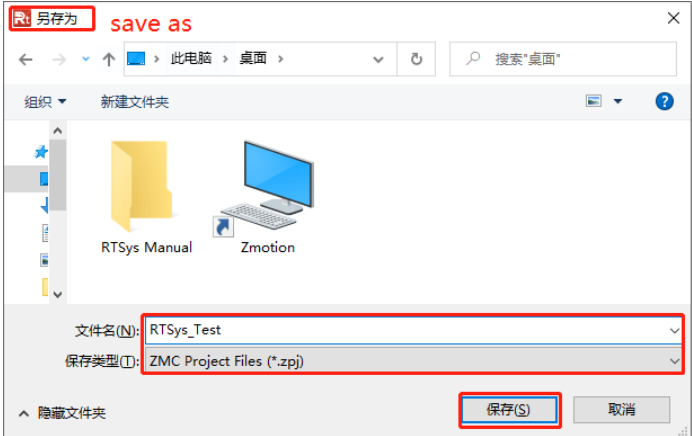
Parameters

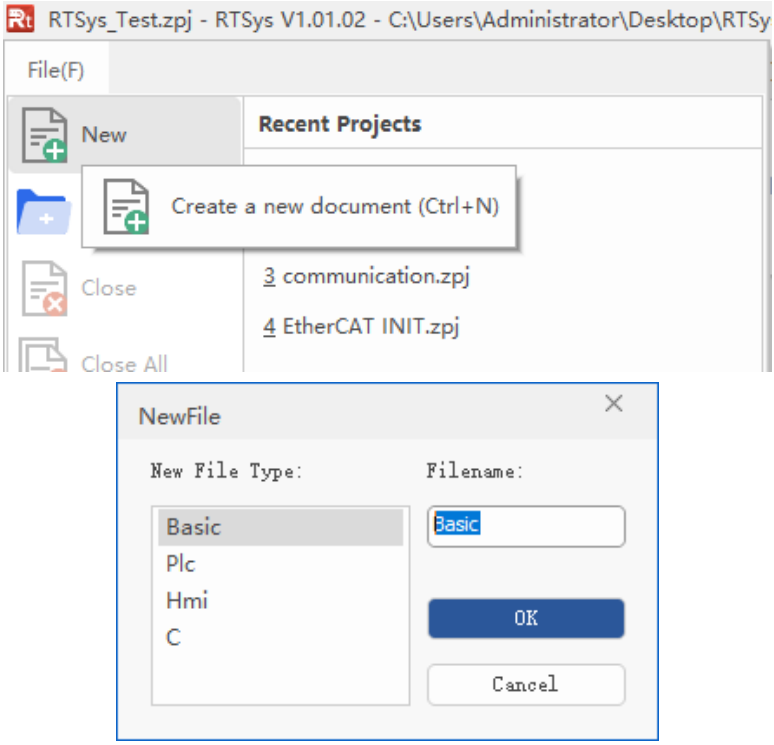
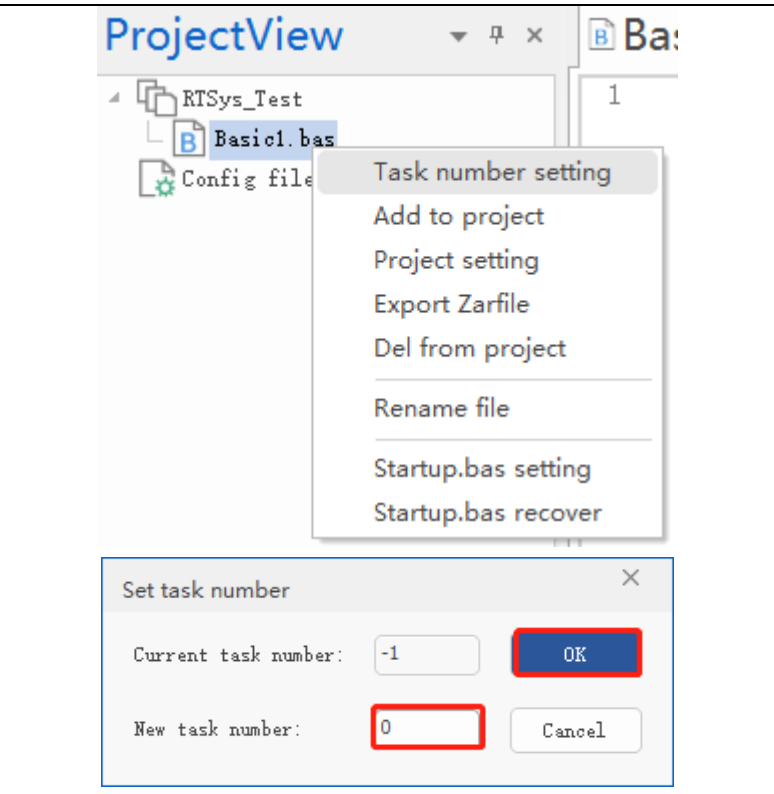
System Architecture

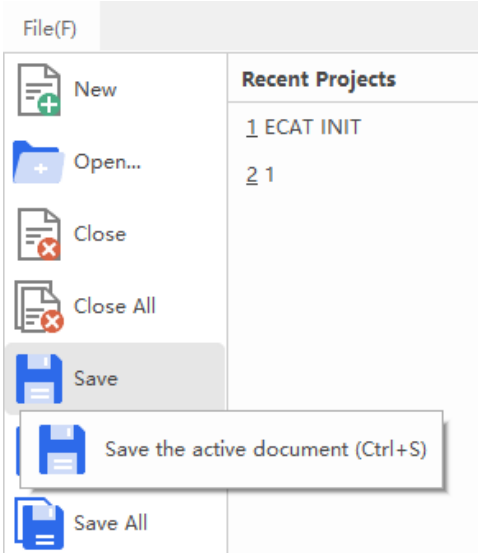
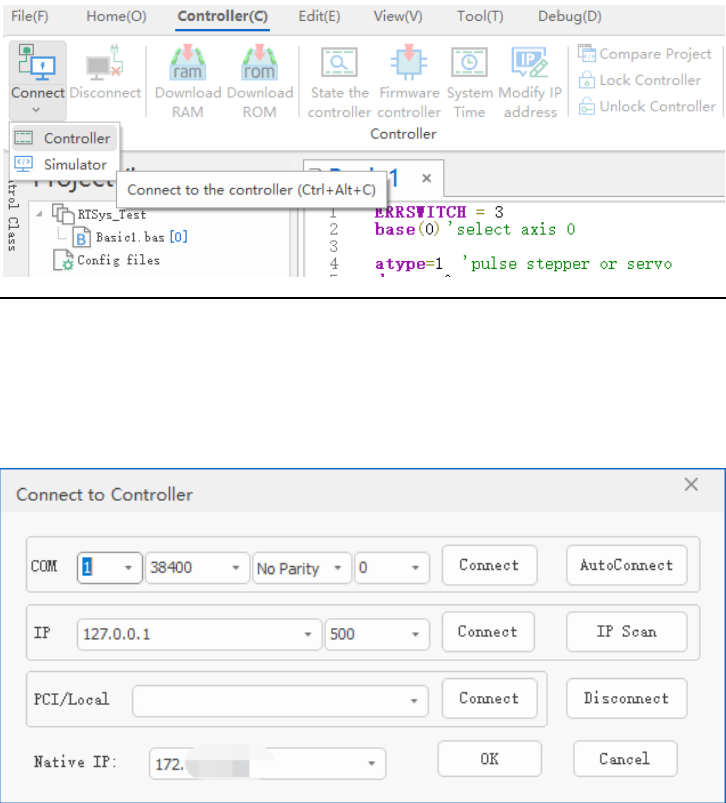
Download

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.	Language Switch Video Showing: E. How to Switch the Language Find "视图" (the fourth one in the above menu), then find the "语言", choose English, restart RTSys. 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

Notes:

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

5.2. Upgrade Controller Firmware

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

How to update:

- a. Open [ZDevelop](#) / [RTSys](#) software, then click "controller – connect", find PCI/LOCAL method, click "connect". If connected, there will be "Connected to Controller: PCIE464 Version: 4.93 – 20231220." In "output" window.
- b. Click "controller – state the controller", find basic info, then current software version can be checked.
- c. Click "controller – update firmware", current controller model and software version can be viewed.
- d. Click "browse", and select saved firmware file, click "update", then one window will pop up, please click "ok".
- e. After that, "connect to controller" window appears again, and please select "PCI/Local" again, and click "connect".
- f. When connection is successful, "firmware update" interface is shown. Now

system enters ZBIOS state, please click "update" again.

- g. When it is loaded, "firmware update" window disappears, now in output window, it shows "Update firmware to Controller Success".
- h. Do step a and step b again, check whether the firmware is updated or not.

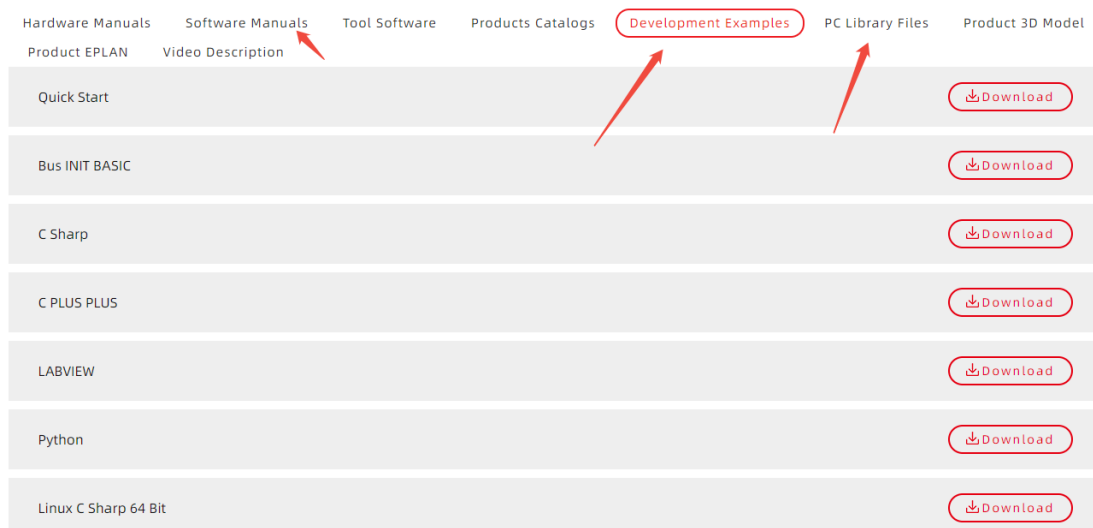
5.3. Program in Host-Computer by PC Languages

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

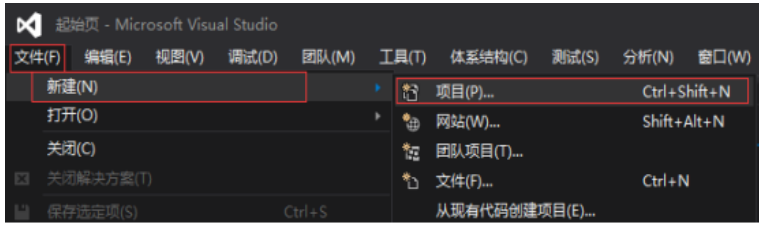
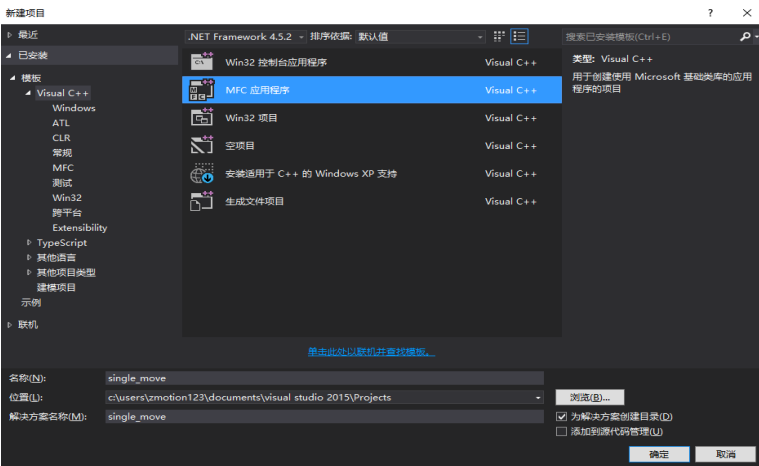
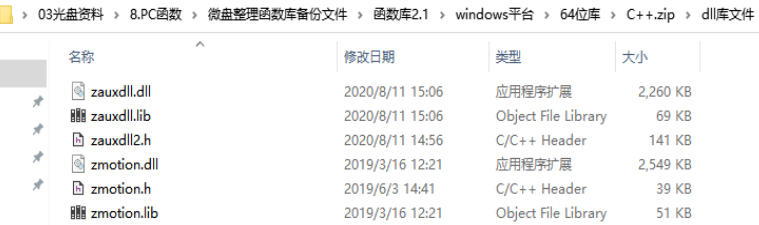


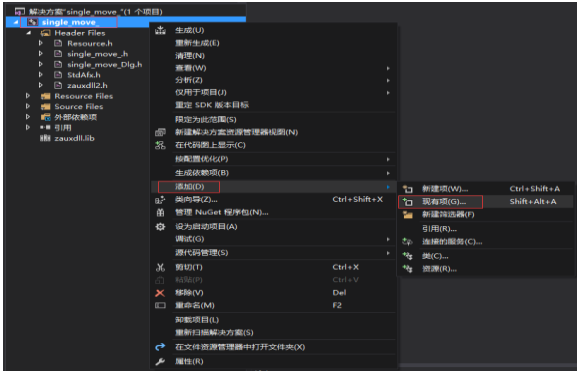
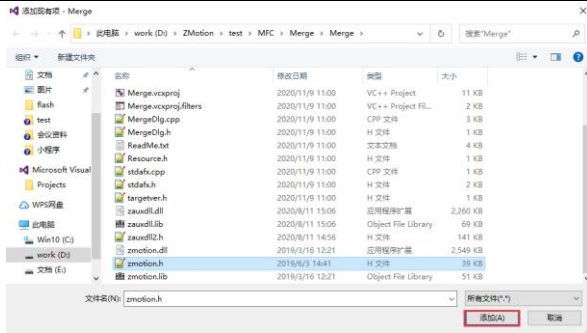
The program developed using the PC software cannot be downloaded to the controller, and it is connected to the controller through the dll dynamic library. The dll library needs to be added to the header file and declared during development.

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



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

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

6	Add a static library and related header files to the project. Static library: <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.	<pre>single_moveDlg.cpp : implementation file // #include "stdafx.h" #include "single_move.h" #include "single_moveDlg.h" #include "zauxdll2.h" #ifdef _DEBUG #define new DEBUG_NEW #undef THIS_FILE static char THIS_FILE[] = __FILE__; #endif // CSingle_moveDlg dialog ZMC_HANDLE g_handle = NULL; //控制器链接句柄</pre>	

Chapter VI Operation and Maintain

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

6.1. Regular Inspection and Maintenance

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

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

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

6.2. Common Problems & Solutions

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

The position limit signal is invalid.	1. Check whether the limit sensor is working normally, and whether the "input" view can watch the signal change of the limit sensor. 2. Check whether the mapping of the limit switch is correct. 3. Check whether the limit sensor is connected to the common terminal of the controller.
No signal comes to the input.	1. Check whether the limit sensor is working normally, and whether the "input" view can watch the signal change of the limit sensor. 2. Check whether the mapping of the limit switch is correct. 3. Check whether the limit sensor is connected to the common terminal of the controller.
The output does not work.	1. Check whether IO power is needed. 2. Check whether the outputs are matched.
POWER led is ON, RUN led is OFF.	1. Check whether the power of the power supply is sufficient. At this time, it is best to supply power to the controller alone, and restart the controller after adjustment. 2. Check whether the ALM light flickers regularly (hardware problem).
RUN led is ON, ALM led is ON.	1. Program running error, please check RTSys error code, and check application program.
Fail to connect controller to PC through serial port.	1. Check whether the serial port parameters are modified by the running program, you can check all the current serial port configurations via ?*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 & power supply circuit, whether the 120 ohm resistor is installed at both ends. 2. Check 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 if 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 / value that is > 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.
"IP Modify" Failed	1. Please check DIP, make sure the code is under RUN.