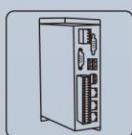
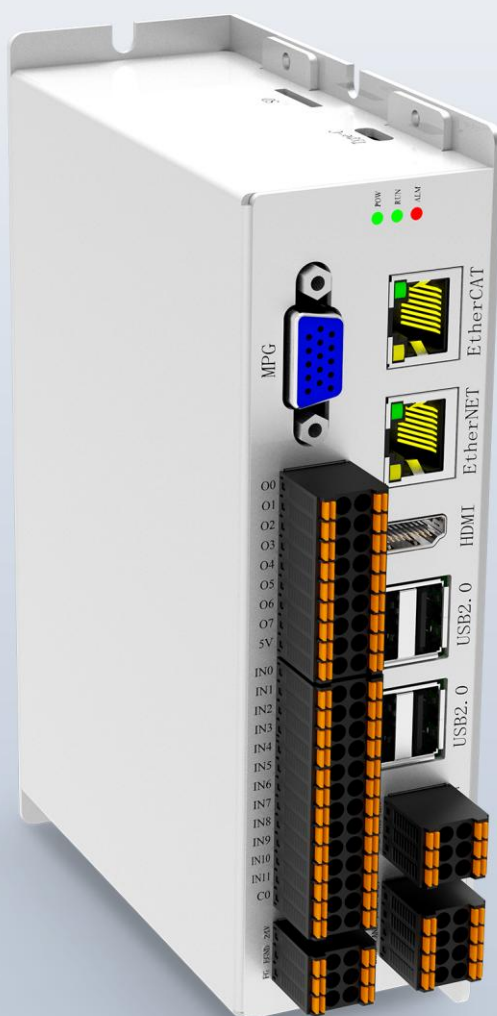
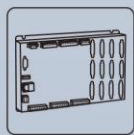


EtherCAT Motion Controller

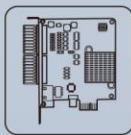
ZMC600M Series



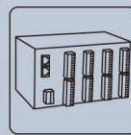
Vision Motion
Controller



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

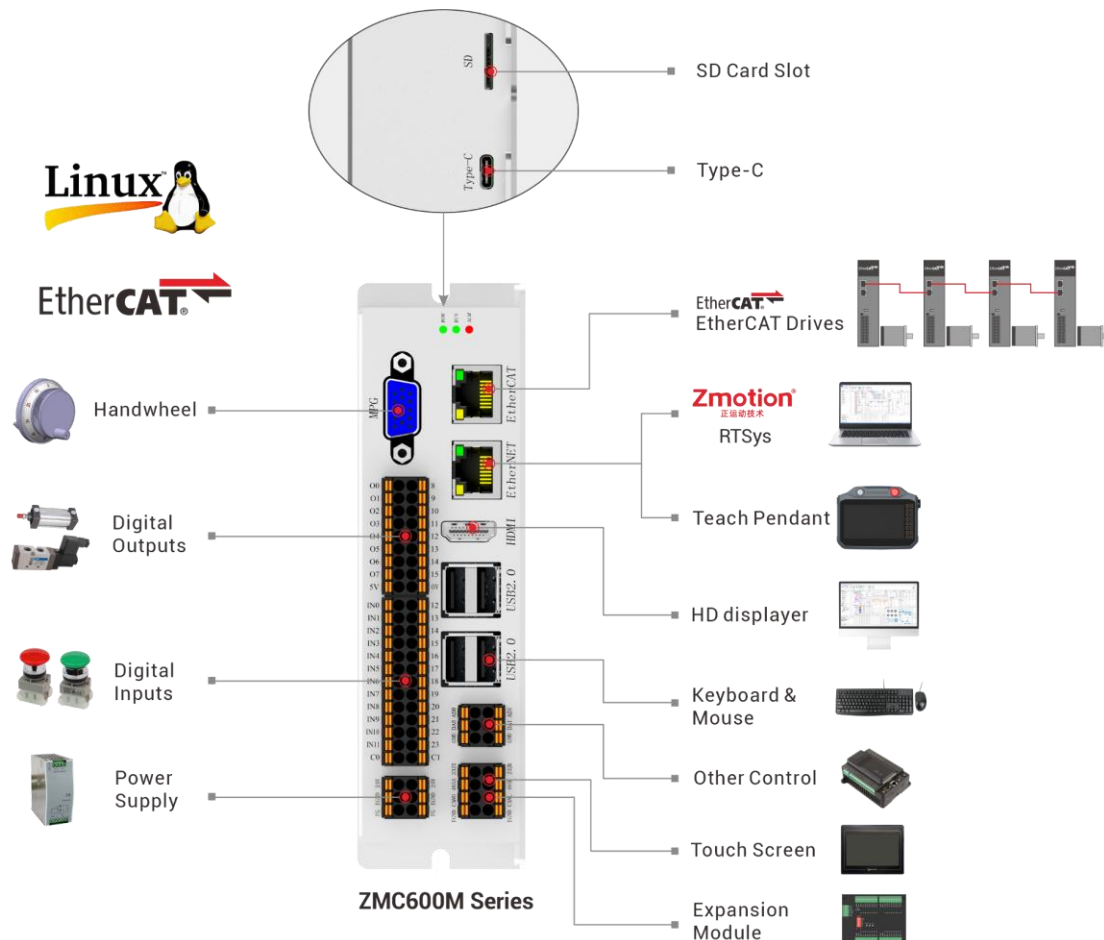
Chapter I Production Information.....	5
1.1. Product Information	5
1.2. Interface Introduction.....	6
1.3. Product Specifications	8
1.4. Nameplate & Model	9
1.5. Parameters Configurations	10
1.6. Environment Requirements	12
1.7. Hardware Installment	12
Chapter II Hardware Interface	13
2.1. Power Input	13
Specification	13
2.2. DI: Digital Inputs	13
Specification	14
Wiring	15
Usage Method	17
2.3. DO: Digital Outputs	18
Specification	19
Wiring	19
Usage Method	21
2.4. AD / DA Analog Interface.....	22
Specification	22
Wiring	22
Usage Method	23
2.5. Communication Interfaces: CAN / RS485 / RS232.....	23
Specification	24
Wiring	24
Usage Method	25
2.6. U Disk.....	26
Specification	27
Usage Method	27
2.7. HDMI Interface	27
2.8. ETHERNET	28
Specification	28
Wiring	28
Usage Method	29
2.9. EtherCAT Bus Interface	30
Specification	30
Wiring	30
Usage Method	31
2.10. MPG Handwheel Interface.....	32

Specification	32
Wiring	33
Usage Method	34
Chapter III Resources Expansion	35
3.1. CAN Expansion	35
Wiring	35
Resources Mapping	36
3.2. EtherCAT Bus Expansion	39
Wiring	39
Resources Mapping	40
Chapter IV Programming	42
4.1. Program in RTSys Software	42
4.2. Upgrade Controller Firmware	47
4.3. Program in Host-Computer by PC Languages	47
Chapter V Operation and Maintain	51
5.1. Regular Inspection and Maintenance	51
5.2. Common Problems & Solutions	52

Chapter I Production Information

1.1. Product Information

ZMC600M high-performance multi-axis motion controller is a EtherCAT stand-alone motion controller, which supports max 32 axes for complex continuous trajectory control.



✚ **Linux system is built in.**

✚ There are interfaces of EtherCAT, single-ended pulse axes, single-ended encoders.

✚ **Basic Motion Control Functions:** point motion, electronic cam, linear interpolation, circular interpolation, continuous interpolation, robot algorithms, etc.

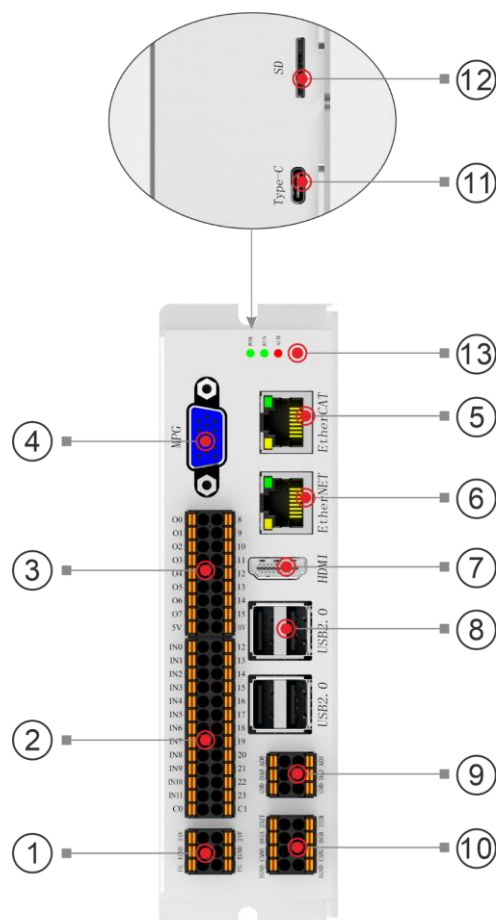
✚ **Special Motion Control Functions:** HW hardware comparison output, high-speed latch, PWM, etc., and **for special models (ZMC600ML series), it supports 5V/ 24V**

(can't be switched) laser specialized PWM push-pull output.

- 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.

ZMC6XX 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.	Interfaces	Descriptions
①	Power	connect to 24V DC power supply
②	DI	- NPN / PNP (depend on public end type); - IN0-23; - some inputs can be set as single-ended encoder, or be 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	MPG	connect to 5-24V handwheel.
⑤	EtherCAT	- connect to EtherCAT drive; - or connect to EtherCAT expansion modules.
⑥	EtherNET	- connect to upper device (PC, etc.); - MODBUS_TCP protocol; - it can be extended by one industrial switch.
⑦	HDMI	- connect to displayer; - HD display (resolution of 1080p).
⑧	USB2.0	connect to USB external devices (mouse, keyboard, U disk, etc.).
⑨	AD / DA	- single-ended type; 12-bit, 0-10V.
⑩	CAN	connect to CAN expansion modules to expand more resources;
	RS485	connect to upper computer;
	RS232	MODBUS_RTU
⑪	Type-C	USB2.0 type-c OTG interface
⑫	SD Card-Slot	identify SD card.
⑬	Controller Status Led	PWR led: it is green when power is connected.
		RUN led: it is green when it runs normally.
		ERR led: it is read when it runs abnormally.

1.3. Product Specifications

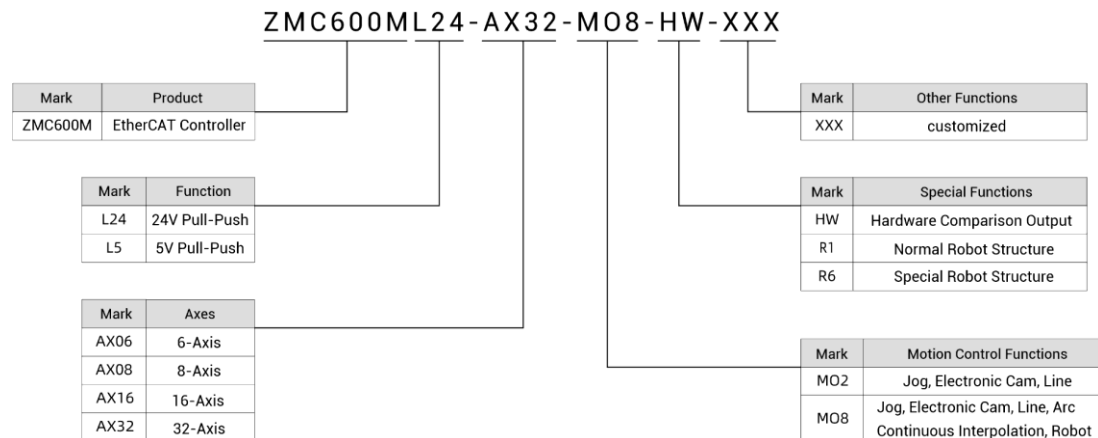
Model	ZMC600ML24-AX32-M08-HW (example)
Basic Motor Axes	32
Total Axes	32 (basic axis + virtual axis)
EtherCAT	√
OUT (single-ended pulse axis)	4
IN (single-ended encoder axis)	3
Digital IN	24
Digital OP	16
Max Extended DI	≤4096
Max Extended DO	≤4096
AD Input	2 (0-10V, 12bit)
DA Output	2 (0-10V, 12bit)
Max Extended AD	≤512
Max Extended DA	≤512
EtherNET	1
EtherCAT	1
CAN	1
RS232	1
RS485	1
USB	4
Type-C	1
High-Speed Latch	2
Hardware Comparison Output	4
General PWM	4
Laser PWM (pull-push)	2 (5V/24V – only valid in L5/L24 laser version)
Point Motion	√

Electronic Cam	
Linear Interpolation	
Circular Interpolation	
Continuous Interpolation	
Robotic Arm	
Program Space	4G/8G (storage space for NC user processing file)
Power Down Storage	√
RTC Clock	It can run one-year continuously without power supply.
Dimensions (mm)	47*162*102

Please check 1.4 information also.

1.4. Nameplate & Model

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



■ Product Models

No.	Models	Description
1	ZMC600M	optional axes numbers; MO2 / MO8 motion control functions.
2	ZMC600ML5	optional axes numbers; MO2 / MO8 motion control functions; 5V laser PWM, pull-push output.

3	ZMC600ML24	optional axes numbers; MO2 / MO8 motion control functions; 24V laser PWM, pull-push output.
---	------------	---

Note: ZMC600M doesn't support laser PWM output, if you need this function, please select corresponding models.

1.5. Parameters Configurations

Do configurations in PC/home/zmotion/zmcapp – ZmotionRtConfig.ini.

```
[config]
memory = 268435456
cpuusage = 1000
hminum = 2
hmisize = 5242880
zvlathum = 4
zvlathsize = 5242880
zarsize = 33554432
tasknum = 28
arrayspace = 2560000
zvobjmax = 2048
submax = 4096
comportmax = 2
ethportmax = 4
ethcustommax = 0
solid3dmax = 128
operpassword = 0
comport0 = /dev/ttyS0
comport1 = /dev/ttyS2
comif485model = 1
ethset0=lan1
autorunapp = 1
autorunrt = 1
rtlogsave = 0
showhide = 0
zfmname = rtkernelarm.zfm
pcinum = 1
pciid0 = 7146
pciboma0 = 0
vcardnum = 0
```

Item	Specification	Details
memory	≤512MB	Total allocated memory is the upper limit, and the sum of all other memory items must not exceed 50% of the total allocated memory.
cpuusage	0-100%	CPU usage limit.

hminum	1-4	HMI numbers (max 4)
hmisize	128k-100m	Hmi occupied bytes
zvlathum	1-8	Vision latch channels
zvlathsize	128k-100m	zv latch size, bytes
zarsize	2-30	ZAR file max bytes
tasknum	2-30	BASIC tasks
arrayspace	128k-100m	array space
zvobjnum	2048-16000	ZVOBJ numbers
submax	256-10000	SUB numbers
comportmax	0-8	Serial ports
comport0	/	1 st serial port configuration (specified serial port character string name in Linux)
comport1	/	N serial port configuration
ethportmax	0-12	The number of PORT net-port.
ethset0	/	1 st ethernet configuration (ethernet card name in Linux)
ethset1	/	N ethernet configuration
ethcustommax	0-12	Self-defined ethernet PORT numbers
solid3dmax	16-512	3D numbers
autorunapp	/	Whether to auto-run the program (reserved for Linux)
autorunrT	/	Whether to auto-start RT software (reserved for Linux)
rtlogsave	/	Whether to save RTLOG files
cpunum	/	CPU No. used for Linux Kernel
pcinum	/	PCIE numbers
pciid	/	PCIE configurations, card type ID + card No.
pciboma	/	PCIE configurations, card DIP + card No.
zfmname	/	Used zfm file name
[reserved] vardnum	/	Virtual net-card numbers
[reserved] vcardid	/	Virtual net-card configurations, type ID
[reserved] vcardset	/	Virtual net-card configurations, settings
[reserved] vcardname	/	Virtual net-card configurations, name

1.6. Environment Requirements

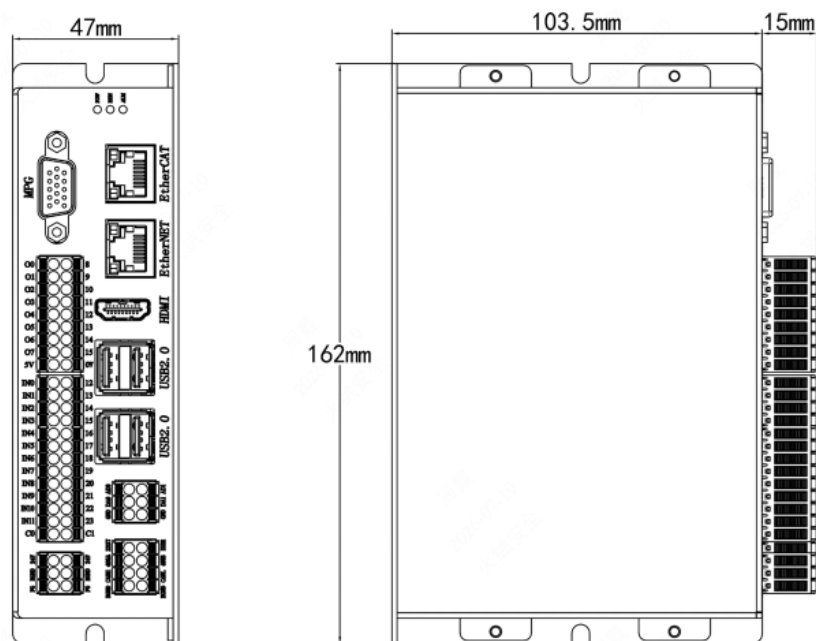
Item		Parameters
Work Temperature [1]		-10°C-55°C
Work relative Humidity		10%-95% non-condensing
Storage Temperature		-40°C ~ 80°C (not frozen)
Storage Humidity		Below 90%RH (no frost)
vibration	Frequency	5-150Hz
	Displacement	3.5mm(directly install)(<9Hz)
	Acceleration	1g(directly install)(>9Hz)
	Direction	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.

1.7. Hardware Installment

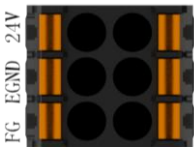
ZMC600M motion controller installment size (unit: mm);

installment hole diameter: 4.5mm



Chapter II Hardware Interface

2.1. Power Input

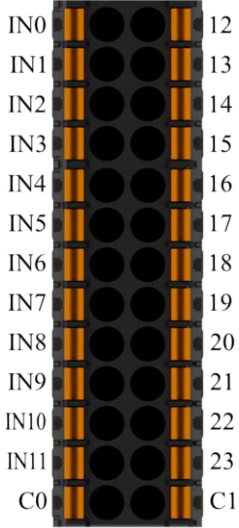
Terminal	Name	Function	Description
	24V	24V main power +	Isolation Power
	EGND	24V main power -	
	FG	Protection	

Specification

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

2.2. DI: Digital Inputs

Terminal	Name	Type	Function 1	Function 2
	IN0	high-speed input NPN / PNP	Digital Input 0	High Speed
	IN1		Digital Input 1	Latch
	IN2		Digital Input 2	/
	IN3		Digital Input 3	/
	IN4		Digital Input 4	EA1
	IN5		Digital Input 5	EB1
	IN6		Digital Input 6	EZ1
	IN7		Digital Input 7	/
	IN8		Digital Input 8	EA2
	IN9		Digital Input 9	EB2

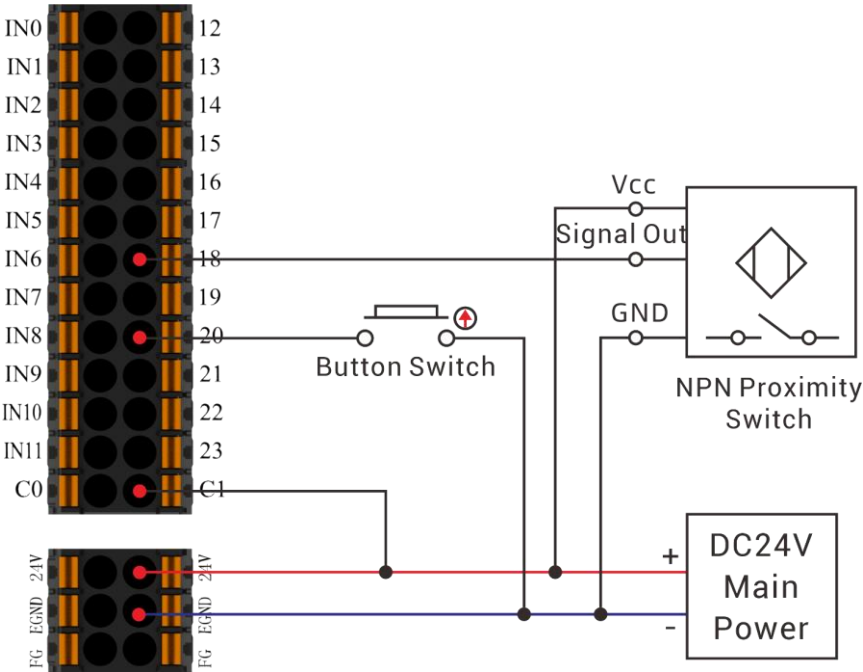
	IN10		Digital Input 10	EZ2
	IN11		Digital Input 11	/
	C0	COM0: IN (0-11) public end -- NPN / PNP		
	12	high-speed input NPN / PNP	Digital Input 12	EA3
	13		Digital Input 13	EB3
	14		Digital Input 14	EZ3
	15		Digital Input 15	/
	16		Digital Input 16	/
	17		Digital Input 17	/
	18		Digital Input 18	/
	19		Digital Input 19	/
	20		Digital Input 20	/
	21		Digital Input 21	/
	22		Digital Input 22	/
	23		Digital Input 23	/
	C1	COM1: IN (12-23) public end -- NPN / PNP		

Specification

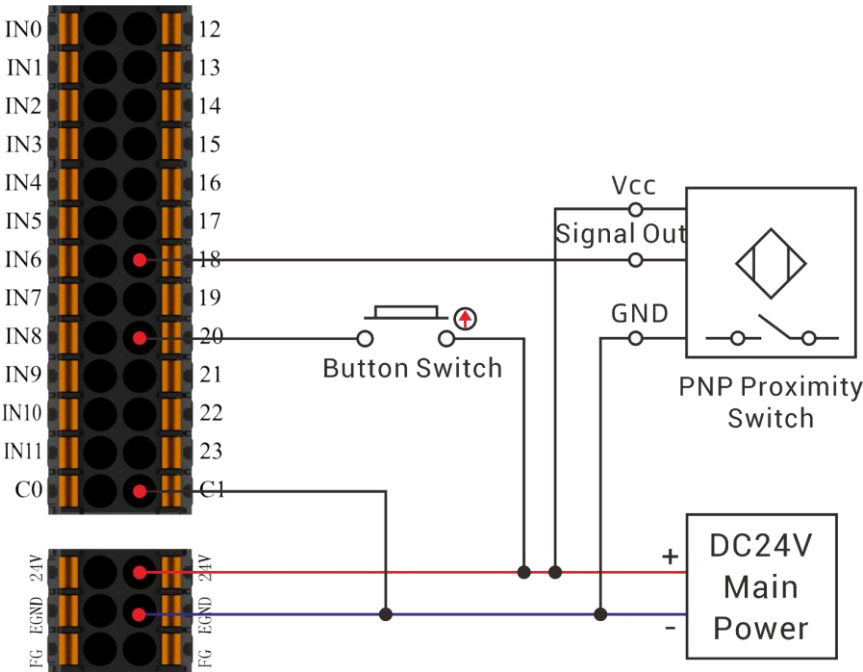
Item	High-Speed Input (IN0-23)	
Input mode	NPN, the input is triggered by low-electric level	PNP, the input is triggered by high-electric level
Frequency	≤400kHz	≤400kHz
Impedance	4.7KΩ	4.7KΩ
Voltage	DC24V	DC24V
Voltage when IN ON	<15V	>14.5V
Voltage when IN OFF	>15.1V	<14.7V
Min IN Current	-2.3mA	2.3mA
Max IN Current	-7.5mA	7.5mA
Isolation	Optical Isolation	
Note: above parameters are obtained when the controller power voltage is 24V.		

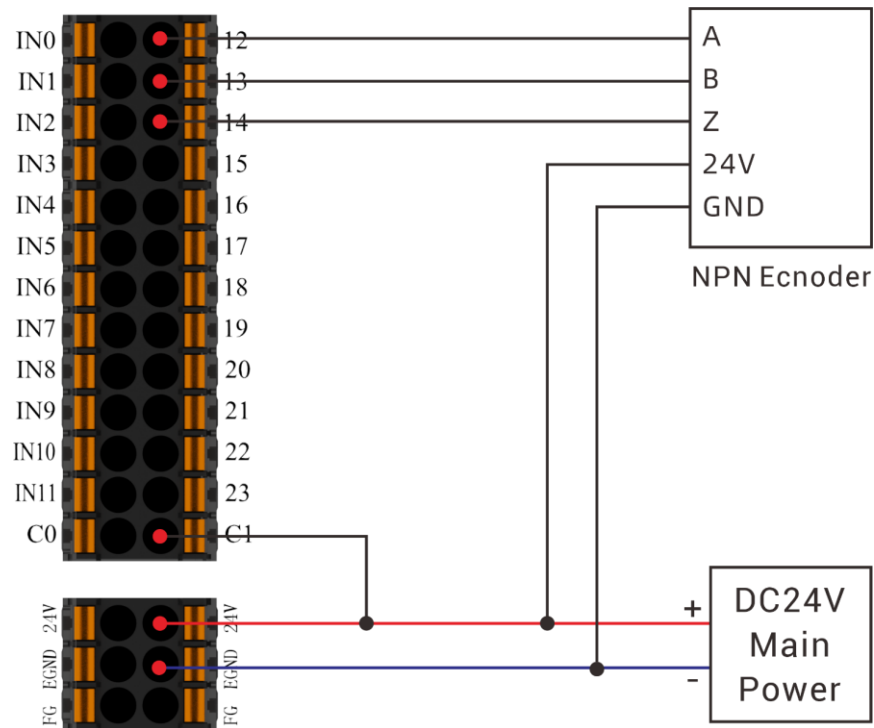
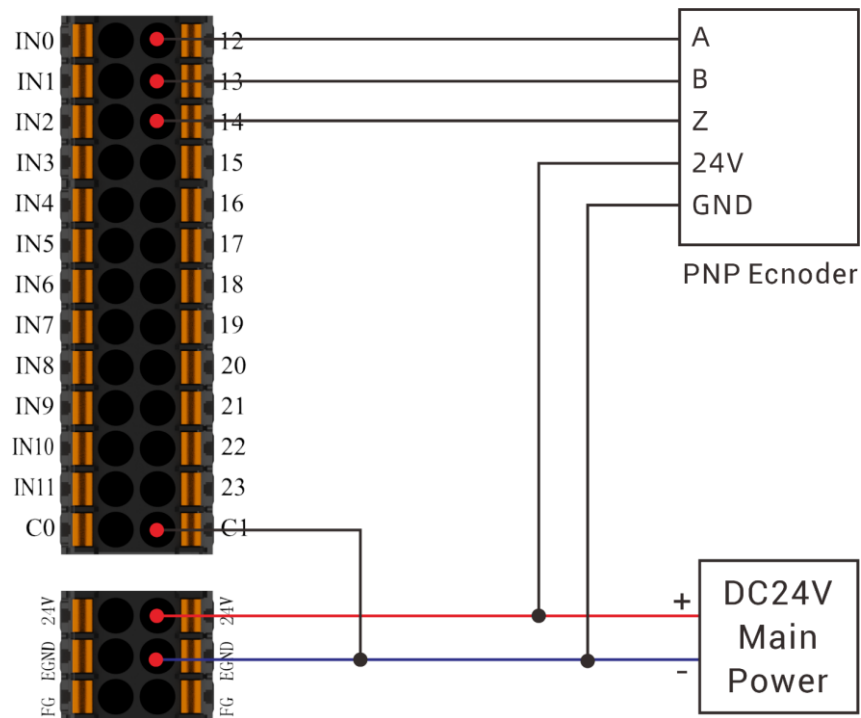
Wiring

NPN Wiring:



PNP Wiring:



NPN Single-Ended Encoder Wiring (example: EA3, EB3, EZ3):**PNP Single-Ended Encoder Wiring (example: EA3, EB3, EZ3):**

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) Set ATYPE of IN4-6, IN8-10, IN12-14 as 3 or 6 to achieve encoder axis function. If 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":

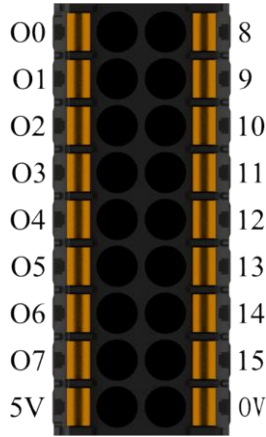
'encoder axis mapping (use EA1, EB1, EZ1)

BASE (10) 'the new (remapping) axis No., don't be same as other axis No.

ATYPE = 0

BASE (1)	'the original axis No.
ATYPE = 0	'set axis type as 0
AXIS_ADDRESS (10) = (0<<16) + 1 + 1	
'the sub-card No. is 0, the physical axis 1 on sub-card is bound with axis 10	
BASE (10)	
ATYPE = 3	'set axis type as encoder axis for axis 10

2.3. DO: Digital Outputs

Terminal	Name	Type	Functions			
			1	2	3	4
	O0	NPN, high-speed outputs, max is 500mA.	output 0	PWM0	Pull-Push	HW
	O1		output 1	PWM1	Pull-Push	HW
	O2		output 2	PWM2	/	HW
	O3		output 3	PWM3	/	HW
	O4		output 4	/	/	/
	O5		output 5	/	/	/
	O6		output 6	/	/	/
	O7		output 7	/	/	/
	8		output 8	DIR3	/	/
	9		output 9	PUL3	/	/
	10		output 10	DIR2	/	/
	11		output 11	PUL2	/	/
	12		output 12	DIR1	/	/
	13		output 13	PUL1	/	/
	14		output 14	DIR0	/	/
	15		output 15	PUL0	/	/
	5V	5V power output +, max is 300mA				
	0V	5V power output -, digital output public end				

Descriptions

- ✧ 5V power terminal is mainly used for PWM wiring or single-ended pulse axis. Not recommend to do other usages due to its small power.
- ✧ OUT8-15 support axis 0-3 function, when ATYPE = 1, they are used as single-ended pulse axes; when ATYPE = 0, they are general digital outputs.
- ✧ OUT0-1 supports laser push-pull output, only valid in laser version ZMC600ML5

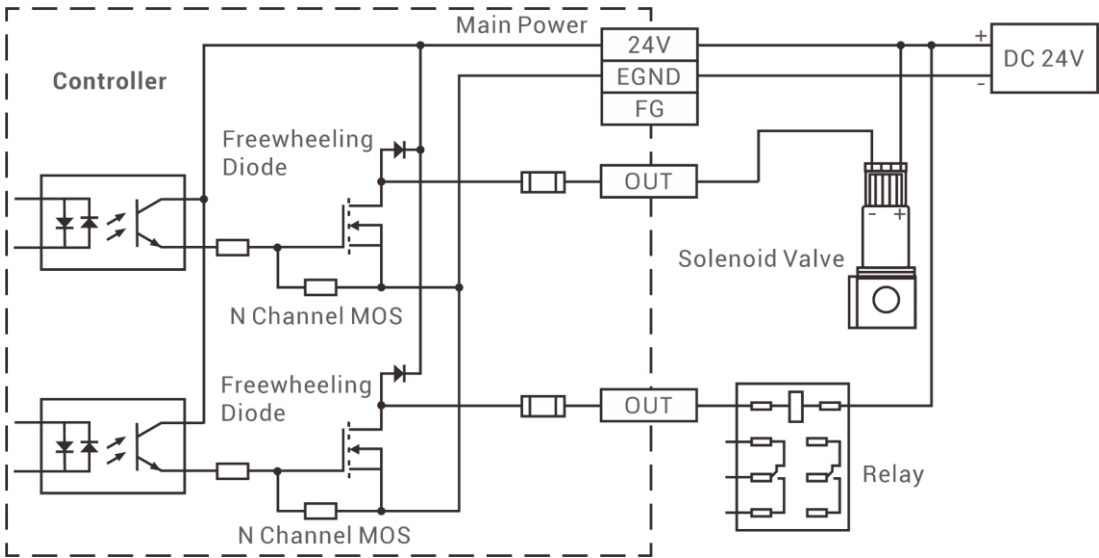
/ ZMC600ML24. Attention the power of laser PWM (OUT0 & OUT1) is not big, without overcurrent protection, and they don't support inductive loads, which means don't directly drive the relay, motor, solenoid valve, etc.

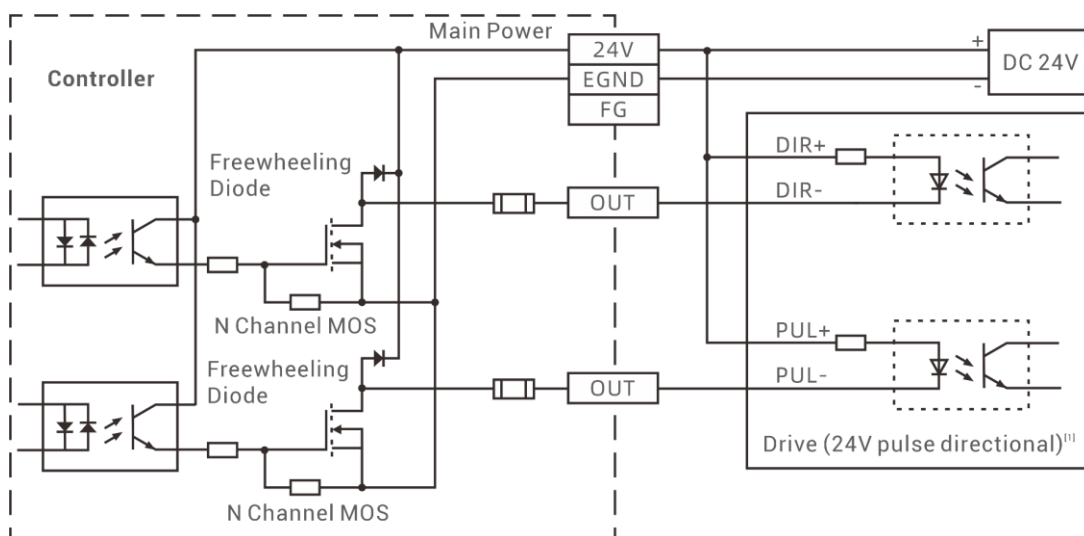
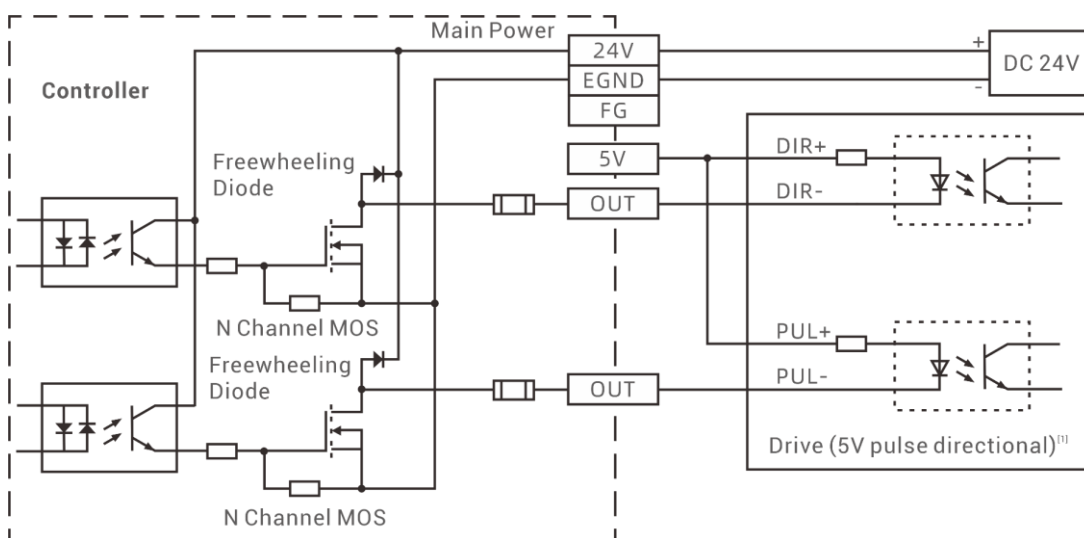
Specification

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

Wiring

General Output Wiring:



Pulse Axis Wiring (24V, PUL0 & DIR0):**Pulse Axis Wiring (5V, PUL0 & DIR0):****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.

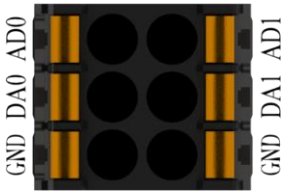
- During on-site wiring, maintain a distance from high-voltage power lines, preferably 30cm or more.
- Ensure that the controller is properly grounded, and the chassis ground should be connected to a standard factory ground rod.

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.
- (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 PUL0, DIR0)	
BASE (16)	'the new (remapping) axis No., don't be same as other axis No.
ATYPE = 0	
BASE (0)	'the original axis No.
ATYPE = 0 'set axis type as 0	
AXIS_ADDRESS (10) = (0<<16) + 0 + 1	
'the sub-card No. is 0, the physical axis 0 on sub-card is bound with axis 16	
BASE (16)	
ATYPE = 1	'set axis type as pulse axis for axis 16

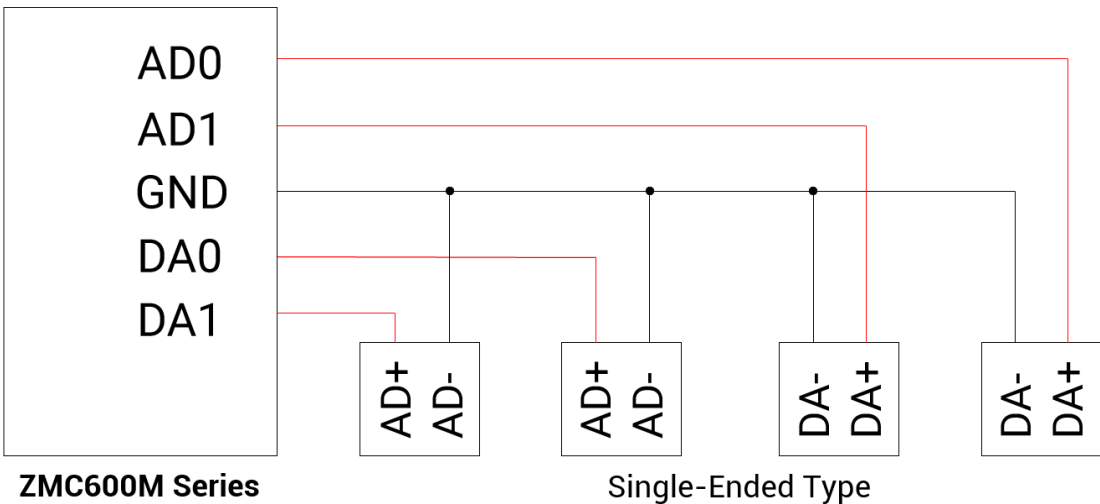
2.4. AD / DA Analog Interface

Terminal	Name	Function
	AD0	analog input 0
	AD1	analog input 1
	DA0	analog output 0
	DA1	analog output 0
	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	0-10V
Data refresh ratio	1KHz (sampling ratio)	1KHz (refresh ratio)
Input impedance / output load	>40KΩ (input impedance)	>10KΩ (load requirement)
Signal Type	Single-ended	Single-ended
Cable Length	Recommend <5m	Recommend <5m

Wiring



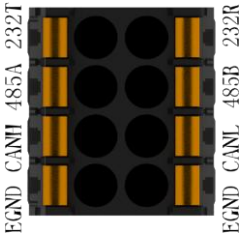
Notes:

- 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

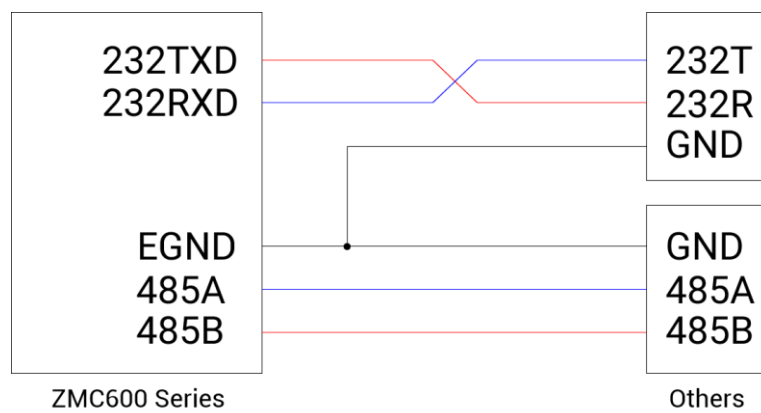
Terminal	Name	Function
 Isolated Communication	232T	RS232 communication signal side TX
	232R	RS232 communication signal side RX
	485A/+	RS485 communication signal cable + / side A
	485B/-	RS485 communication signal cable - / side B
	CANH	CAN communication signal cable side H
	CANL	CAN communication signal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

Specification

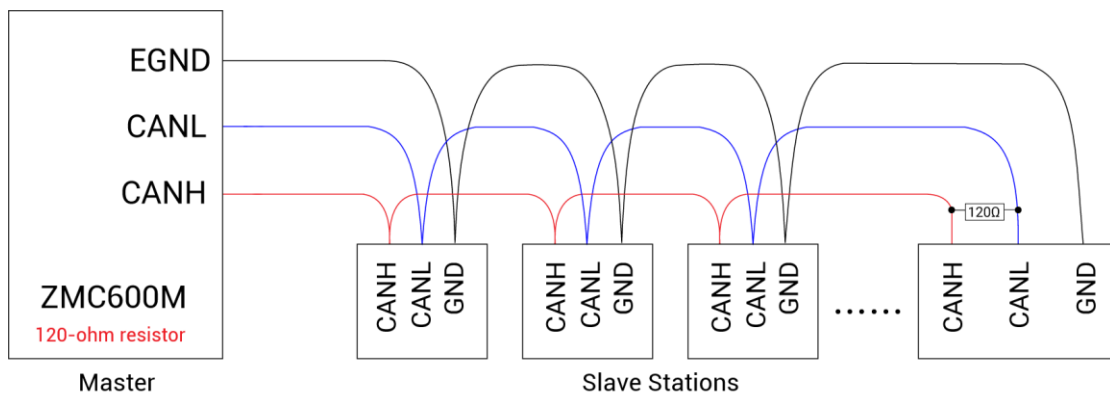
Item	CAN	RS485 (port1)	RS232 (port0)
Communication Rate	≤1Mbps	≤115200bps	≤115200bps
Terminal Resistor	120Ω (inside in controller)	/	/
Topological Structure	Daisy Chain Topology		One to One
Max Extended Nodes	≤16	≤127	1
Wiring Length	<30m (500kbps)	<30m	<5m
Communication Isolation	√		

Wiring

RS485 / RS232 Wiring:



CAN Wiring:



Notes:

- As above, the daisy chain topology is used for CAN / RS485 wiring (the star topology structure cannot be used). The shorter 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 ["3.1 CAN Expansion"](#).
 - 3) After configured, repower on all slave stations to build normal communication, if the slave module "ALM" led is ON, which means the communication fails.
 - 4) Note that the "velocity" settings of each node on the CAN bus must be consistent, and the "address" settings cannot cause conflicts, otherwise the communication establishment will fail or the communication will be disordered.

(3) How to use RS485/R232:

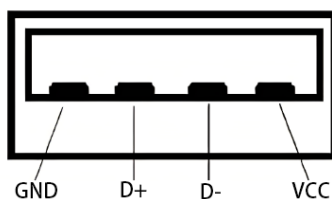
- 1) Please use "ADDRESS" and "SETCOM" commands to check protocol station No. and set parameters, also, you can check and set in **"RTSys>Controller>Controller Status>Communication Config"** interface.
 - 2) According to description, set the third-party device parameters correctly to match each node.
 - 3) After all configured, it is time to communicate.
- (4) Please refer to ["RTBASIC Programming Manual"](#), or check in **RTSys / Help / RTBasic**.

2.6. U Disk

The ZMC600M motion controller provides 4 USB communication interfaces for inserting the U disk device, and they are with indicator lights.

Main Functions: ZAR program upgrading, controller data importing and exporting, file 3

executing. USB external devices wiring (keyboard & mouse, U disk), etc.



Specification

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

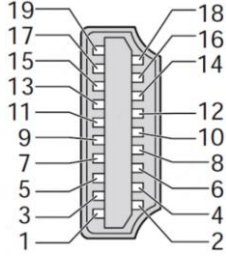
Usage Method

- (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. HDMI Interface

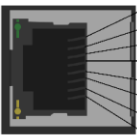
There is one HDMI interface, used to connect to displayer (1080p-only).

Terminal	PIN No.	Signals	PIN No.	Signals
	1	TMDS DATA 2+	11	TMDS CLOCK SHIELD
	2	TMDS DATA 2 SHIELD	12	TMDS CLOCLK -

	3	TMDS DATA 2-	13	CEC
	4	TMDS DATA 1+	14	N.C.
	5	TMDS DATA 1 SHIELD	15	DDC CLOCK
	6	TMDS DATA 1-	16	DDC DATA
	7	TMDS DATA 0+	17	GND
	8	TMDS DATA 0 SHIELD	18	+5V PWR
	9	TMDS DATA 0-	19	HOT PLUG DETECT
	10	TMDS CLOCLK +		

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	Normal-ON	Shrink
Green	Build 100M communication	Now it is sending & receiving data
Yellow	Build 1000M 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.

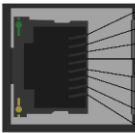
Usage Method

- (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*.
- (5) How to modify gateway when ZMC600M is connect to internet (here uses the controller default IP):

step 1: set controller IP address as 192.168.0.11 (default);
step 2: modify controller file contents under path "/etc/resolv.config" as:
 nameserver 180.76.76.76
 nameserver 114.114.114.114
step 3: execute "route add default gw 192.168.0.1" (needed IP gateway).

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 (+)	Refresh Ratio	Max: 500us
	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) How to connect to drive devices through EtherCAT:

- 1) Use SLOT_SCAN command to scan the slot No. on the bus.
- 2) Use AXIS_ADDRESS command to map axis No., you can refer to "[3.2 EtherCAT Expansion](#)" – "Resource Mapping".
- 3) Use SLOT_START to open bus, SLOT_STOP to stop.
- 4) After completed, configure and operate local pulse axis.

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

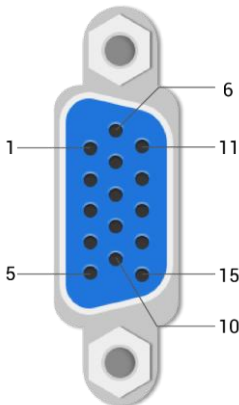
- 1) Use SLOT_SCAN command to scan the slot No. on the bus.
- 2) Use AXIS_ADDRESS to map axis No., use NODE_IO / NODE_AIO command to map IO No., you can refer to "[3.2 EtherCAT Expansion](#)" – "Resource Mapping".
- 3) Use SLOT_START to open bus, SLOT_STOP to stop.
- 4) After completed, configure and operate local IO, same as axis, for details, please refer to "2.2, 2.3, 2.10" – "Usage Methods".

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

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

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

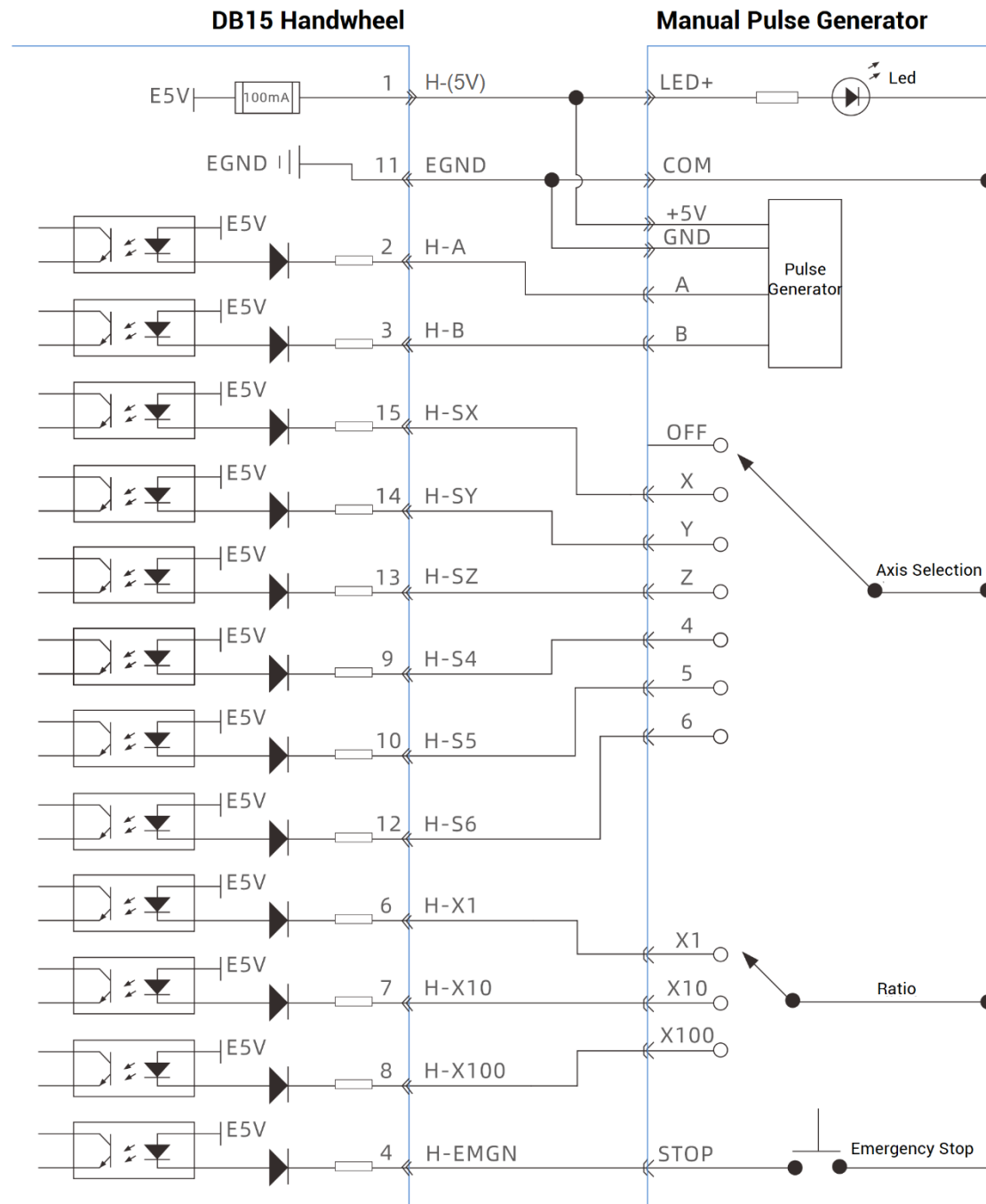
2.10. MPG Handwheel Interface

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 phase A (IN24)
	3	H-B	Encoder signal phase B (IN25)
	4	H-EMGN	Emergency stop signal (IN35)
	5	NC	Spare (reserved)
	6	H-X1	Select the ration as X1 (IN26)
	7	H-X10	Select the ration as X10 (IN27)
	8	H-X100	Select the ration as X100 (IN28)
	9	H-S4	Select axis 3 (IN32)
	10	H-S5	Select axis 4 (IN33)
	11	EGND	external power ground
	12	H-S6	Select axis 5 (IN34)
	13	H-SZ	Select axis 2 (IN31)
	14	H-SY	Select axis 1 (IN30)
	15	H-SX	Select axis 0 (IN29)

Specification

Item	High-Speed IN (24-35)
Input Mode	NPN type, ON when low-level
Frequency	<5kHz (recommended)
Impedance	510Ω
Voltage (max)	24V
Voltage to open voltage	<2.8
Voltage to close voltage	>2.9V
Current (min)	-1.8mA (negative)
Current (max)	-5.5mA (negative)
Isolation	optoelectronic isolation
5V power max OUT current (H-(5V), EGND)	100mA

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.
- The controller supports 5V/24V handwheels; for 24V type, it also can connect one

24V power supply for the handwheel.

- Please use STP, especially in bad environments, and make sure the shielding layer is fully grounded.

Usage Method

- (1) Refer to above handwheel wiring, correctly connect the handwheel to controller.
- (2) After powered on, please select ETHERNET or RS232 to connect to [RTSys](#).
- (3) Configure axis No., **the default handwheel axis No. is AXIS0, no need to do remapping:**

BASE (0) 'select axis 0
ATYPE (0) = 6 'set needed axis type, 3/6
- (4) Configure IO: assign axis selection (H-SX, H-SY, H-SZ, H-S4, H-S5, H-S6) and ratio (H-X1, H-X10, H-X100) and emergency stop (H-EMGN) functions as required. These signals are essentially digital input signals with fixed No. but no fixed functions. It needs developing by RTSys (the axis selection is the connected axis of "connect" synchronization motion, and the ratio is the "connect" ratio).
- (5) When completed above steps, it can start to use handwheel.

Chapter III Resources Expansion

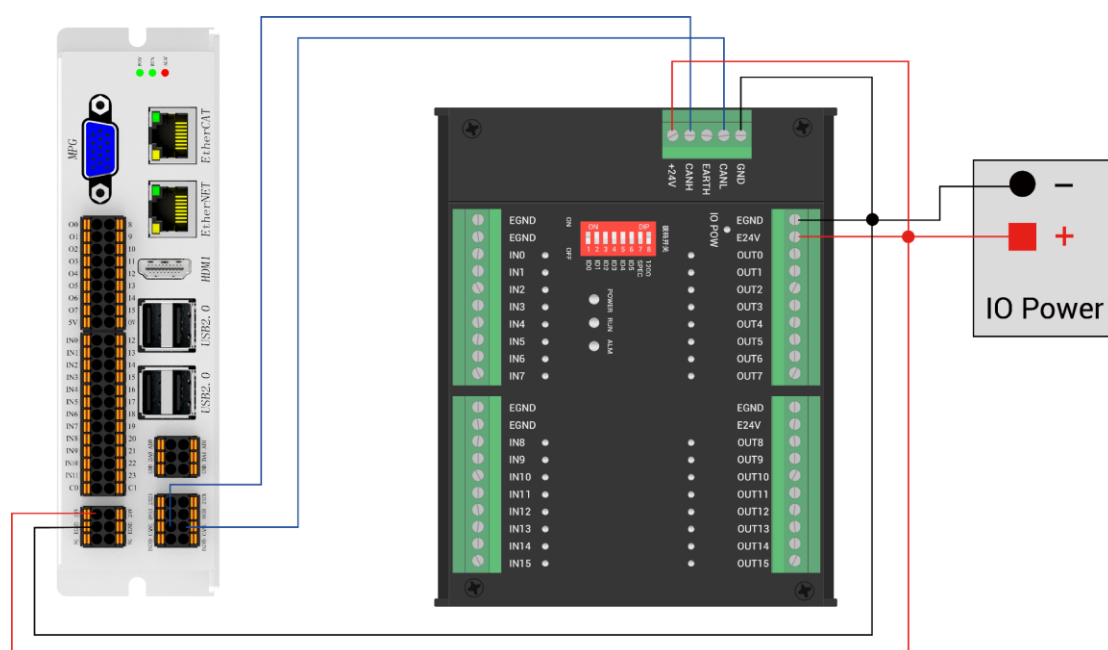
When you need more resources, expansion modules can be used by CAN / EtherCAT.

3.1. CAN Expansion

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

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

Wiring

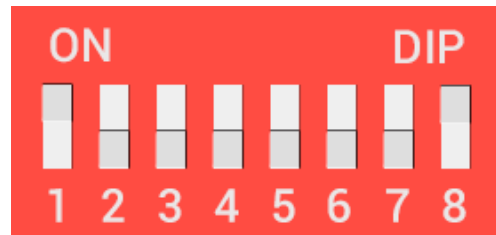


Notes:

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



DIP Switch

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

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

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

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

DIP 5-6 combination value	CAN communication speed
0	500KBPS (default value)
1	250KBPS
2	125KBPS
3	1MBPS

Notes:

- ✧ "How to configure controller as master station", please check ["2.5 CAN Communication Interface" – "Usage Methods"](#).

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

B. IO Mapping

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

--Digital IO Mapping No.--

Code 4	Code 3	Code 2	Code 1	Address ID	Start IO No.	End IO No.
0	0	0	0	0	16	31
0	0	0	1	1	32	47
0	0	1	0	0	0	63
0	0	1	1	3	64	79
0	1	0	0	4	80	95
0	1	0	1	5	96	111
0	1	1	0	6	112	127
0	1	1	1	7	128	143
1	0	0	0	8	144	159
1	0	0	1	9	160	175
1	0	1	0	10	176	191
1	0	1	1	11	192	207
1	1	0	0	12	208	223
1	1	0	1	13	224	239
1	1	1	0	14	240	255
1	1	1	1	15	256	271

--Analog IO Mapping, for bit 1-4 code state & 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



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):

<code>NODE_IO(0,0)=32</code>	'set the IO start No. of slot 0 interface device 0 to 32
<code>NODE_AIO(0,0,3)=3</code>	'set the AIO start No. of slot 0 interface device 0 to 3

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

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

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

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 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.

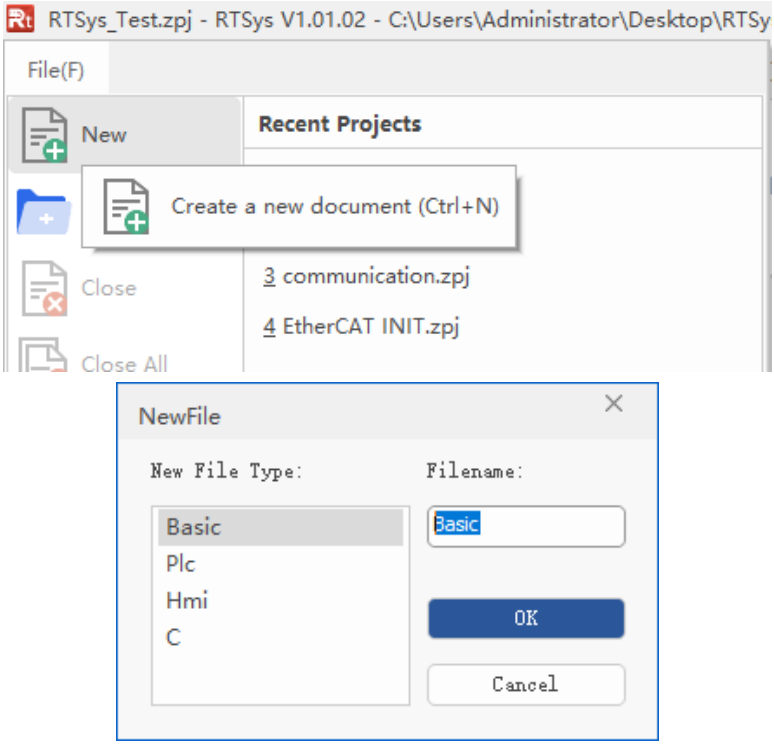
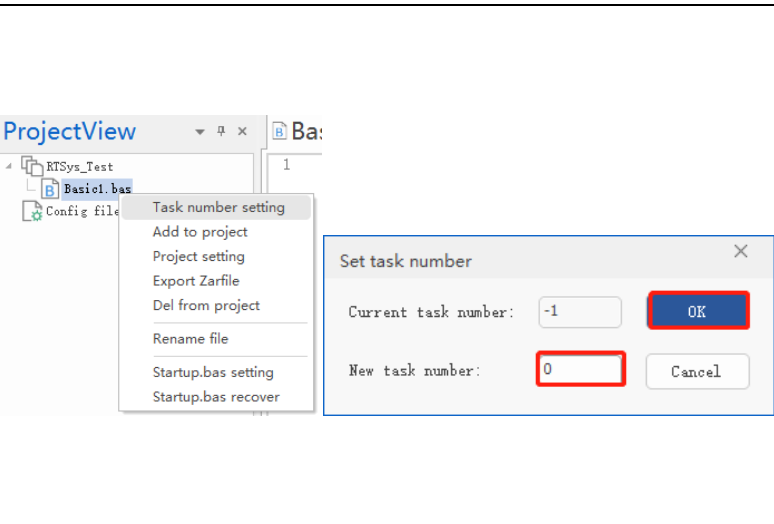
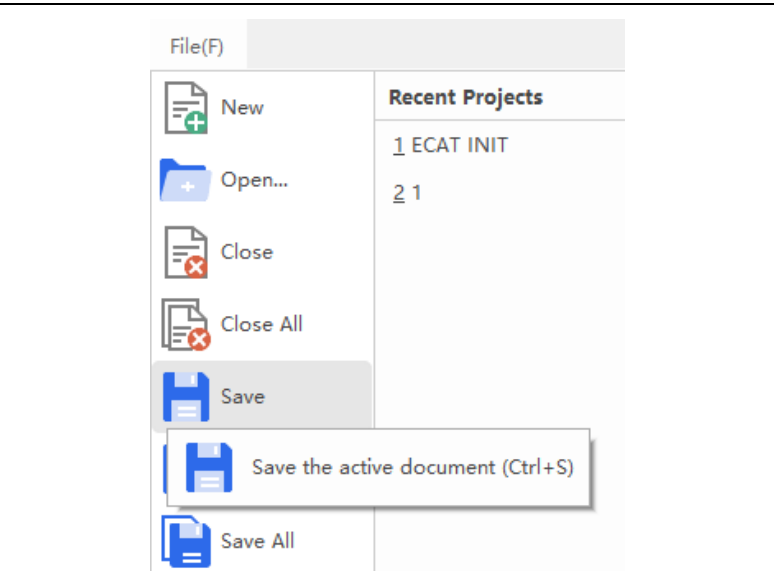
RTSys Downloading Address: https://www.zmotionglobal.com/pro_info_282.html

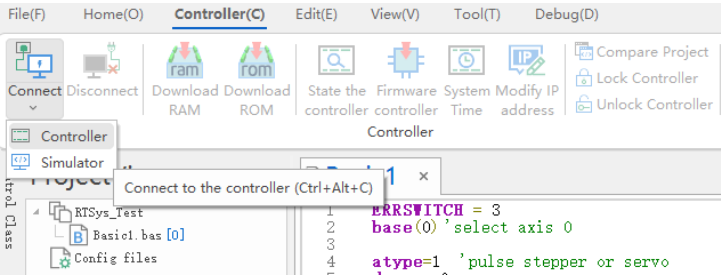
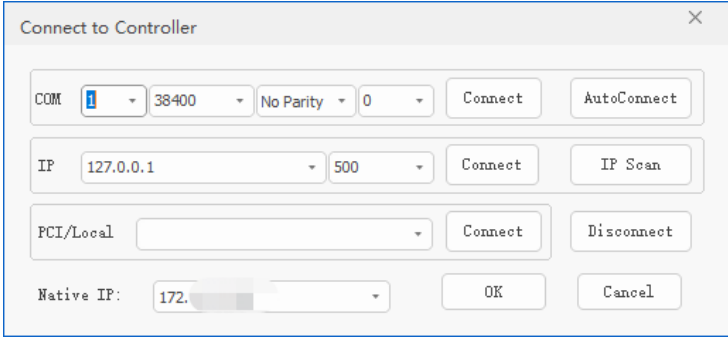
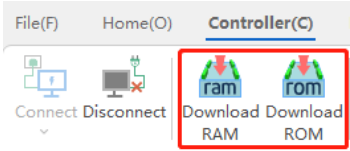
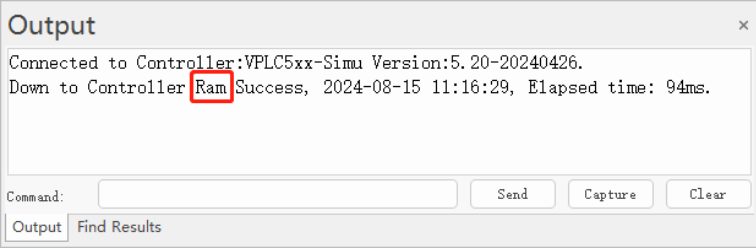
And related manuals can be found in "Download":

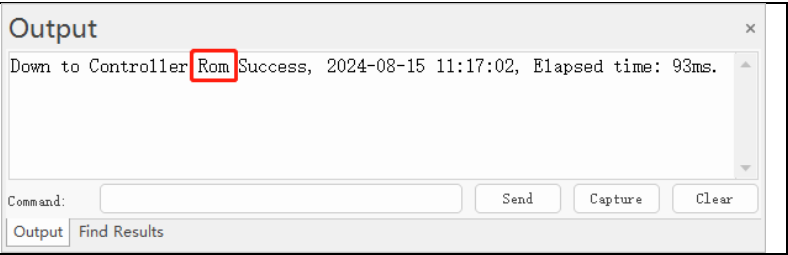
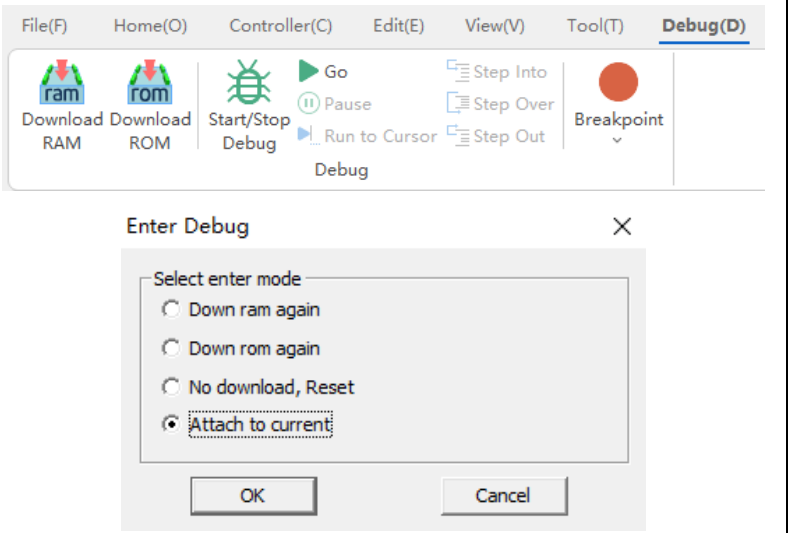
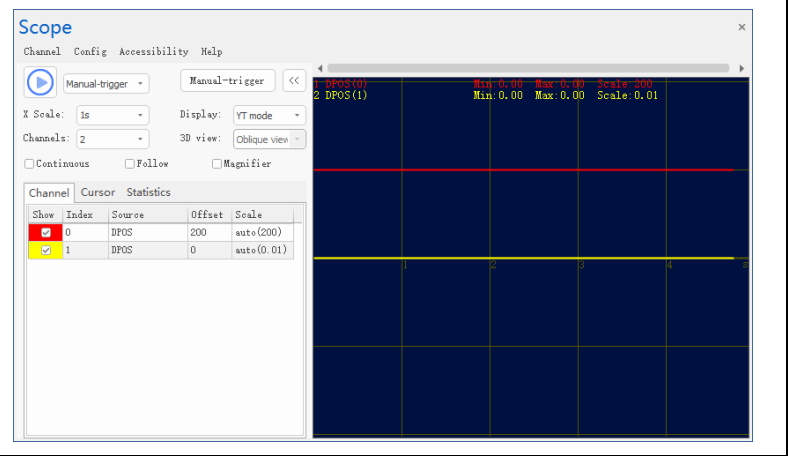
Features Parameters System Architecture Download				
Name	Version No	Format	Size	Download
RTSys Development Software	V1.2.02	RAR	148MB	Download
RTSys User Manual V1.2.0	V1.2.0	PDF	5.33MB	Download
RTBasic Programming Manual	V1.1.0	PDF	18.3MB	Download
RTHMI Programming Manual	V1.2.0	PDF	7.23MB	Download
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.	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".	 RTSys_Test.zpj - RTSys V1.01.02 - C:\Users\Administrator\Desktop\RTSy File(F) New Recent Projects Create a new document (Ctrl+N) Close Close All 3 communication.zpj 4 EtherCAT INIT.zpj NewFile New File Type: Basic Plc Hmi C Filename: Basic OK Cancel
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.	 ProjectView RTSys_Test Bas1.bas Config file Task number setting Add to project Project setting Export Zarfile Del from project Rename file Startup.bas setting Startup.bas recover Set task number Current task number: -1 New task number: 0 OK Cancel
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	 File(F) New Open... Close Close All Save Save the active document (Ctrl+S) Save All Recent Projects 1 ECAT INIT 2 1

	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 Controller: “Ram/Rom” – “download RAM / ROM”, if it is successful, there is print indication, at the same time, program is	● RAM: it will not save when power off. ● ROM: it will save data when power off, and when the program is connected to controller again, running according to task No.  

	downloaded into controller and runs automatically.	
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 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.

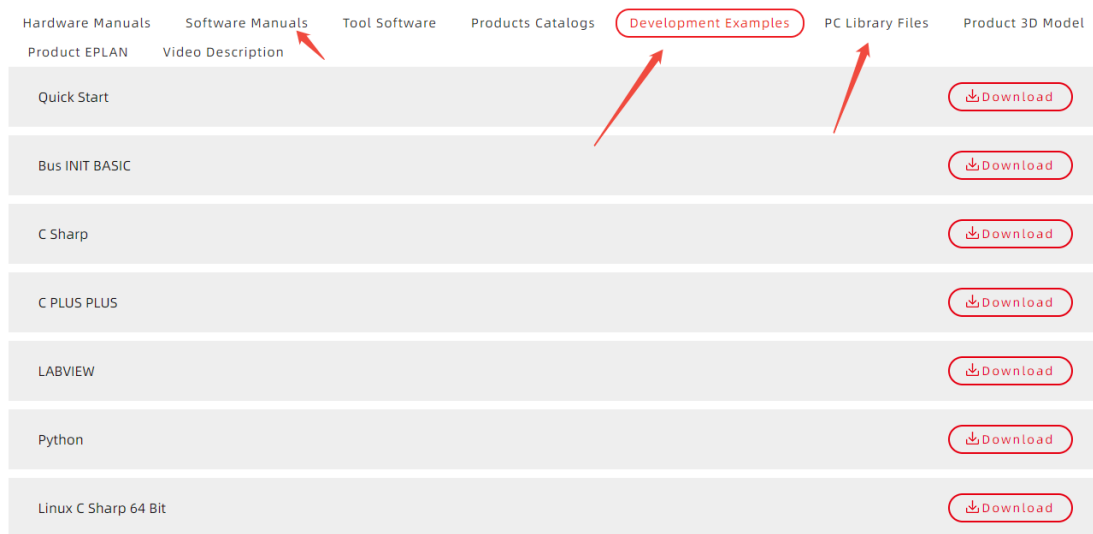
4.3. Program in Host-Computer by PC Languages

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.

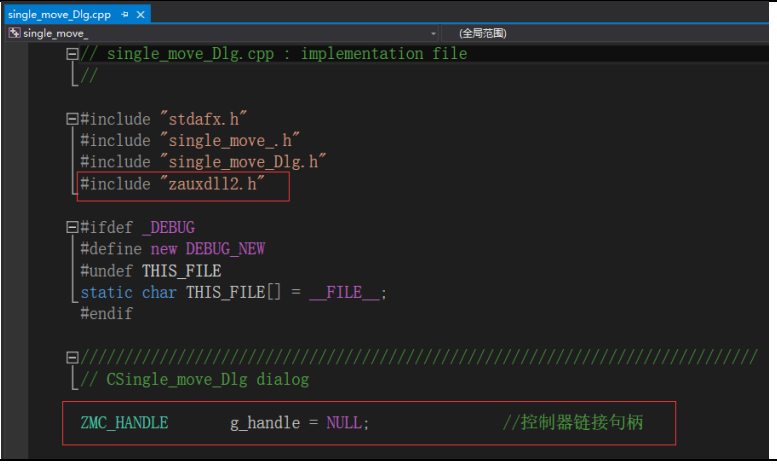
- Get PC library file, example: https://www.zmotionglobal.com/download_list_17.html



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)	<table border="1"> <thead> <tr> <th>名称</th><th>修改日期</th><th>类型</th><th>大小</th></tr> </thead> <tbody> <tr> <td>zauxdll.dll</td><td>2020/8/11 15:06</td><td>应用程序扩展</td><td>2,260 KB</td></tr> <tr> <td>zauxdll.lib</td><td>2020/8/11 15:06</td><td>Object File Library</td><td>69 KB</td></tr> <tr> <td>zauxdll2.h</td><td>2020/8/11 14:56</td><td>C/C++ Header</td><td>141 KB</td></tr> <tr> <td>zmotion.dll</td><td>2019/3/16 12:21</td><td>应用程序扩展</td><td>2,549 KB</td></tr> <tr> <td>zmotion.h</td><td>2019/6/3 14:41</td><td>C/C++ Header</td><td>39 KB</td></tr> <tr> <td>zmotion.lib</td><td>2019/3/16 12:21</td><td>Object File Library</td><td>51 KB</td></tr> </tbody> </table>		名称	修改日期	类型	大小	zauxdll.dll	2020/8/11 15:06	应用程序扩展	2,260 KB	zauxdll.lib	2020/8/11 15:06	Object File Library	69 KB	zauxdll2.h	2020/8/11 14:56	C/C++ Header	141 KB	zmotion.dll	2019/3/16 12:21	应用程序扩展	2,549 KB	zmotion.h	2019/6/3 14:41	C/C++ Header	39 KB	zmotion.lib	2019/3/16 12:21	Object File Library	51 KB
名称	修改日期	类型	大小																												
zauxdll.dll	2020/8/11 15:06	应用程序扩展	2,260 KB																												
zauxdll.lib	2020/8/11 15:06	Object File Library	69 KB																												
zauxdll2.h	2020/8/11 14:56	C/C++ Header	141 KB																												
zmotion.dll	2019/3/16 12:21	应用程序扩展	2,549 KB																												
zmotion.h	2019/6/3 14:41	C/C++ Header	39 KB																												
zmotion.lib	2019/3/16 12:21	Object File Library	51 KB																												
5	Copy all DLL related library files under the above path to the newly created project.																														
6	Add a static library and related header files to the project. Static library: zauxdll.lib, zmotion.lib. Related header files: zauxdll2.h, zmotion.h	1) Right-click the header file first, and then select: "Add" → "Existing Item". 2) Add static libraries and related header files in sequence in the pop-up window.																													

7	Declare the relevant header files and define the controller connection handle, so far the project is newly created.	 <pre>single_move_Dlg.cpp : implementation file // #include "stdafx.h" #include "single_move.h" #include "single_move_Dlg.h" #include "zauxdll2.h" #ifdef _DEBUG #define new DEBUG_NEW #undef THIS_FILE static char THIS_FILE[] = __FILE__; #endif // CSingle_move_Dlg dialog ZMC_HANDLE g_handle = NULL; //控制器链接句柄</pre>
---	---	--

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.	1. Check whether the ATYPE of the controller is correct. 2. Check whether hardware position limit, software position limit, alarm signal work, and whether axis states are normal. 3. Check whether motor is enabled successfully. 4. Confirm whether pulse amount UNITS and speed values are suitable. If there is the encoder feedback, check whether MPOS changes. 5. Check whether pulse mode and pulse mode of drive are matched. 6. Check whether alarm is produced on motion controller station or drive station. 7. Check whether the wiring is correct. 8. Confirm whether controller sends pulses normally.

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

	ends. 2. Check the master-slave configuration, communication speed configuration, etc. 3. Check the DIP switch to see if there are multiple expansion modules with the same ID. 4. Use twisted-pair cables, ground the shielding layer, and use dual power supplies for severe interference (the main power supply of the expansion module and the IO power supply are separately powered)
Fail to connect controller to PC through net port.	1. Check IP address of PC, it needs to be at the same segment with controller IP address. 2. Check controller IP address, it can be checked and captured after connection through serial port. 3. When net port led is off, please check wiring. 4. Check whether controller power led POWER and running indicator led RUN are ON normally. 5. Check whether the cable is good quality, change one better cable to try again. 6. Check whether controller IP conflicts with other devices. 7. Check whether controller net port channel ETH are all occupied by other devices, disconnect to other devices, then try again. 8. When there are multiple net cards, don't use other net cards, or change one computer to connect again. 9. Check PC firewall setting. 10. Use "Packet Internet Groper" tool (Ping), check whether controller can be Ping, if it can't, please check physical interface or net cable. 11. Check IP address and MAC address through arp-a.