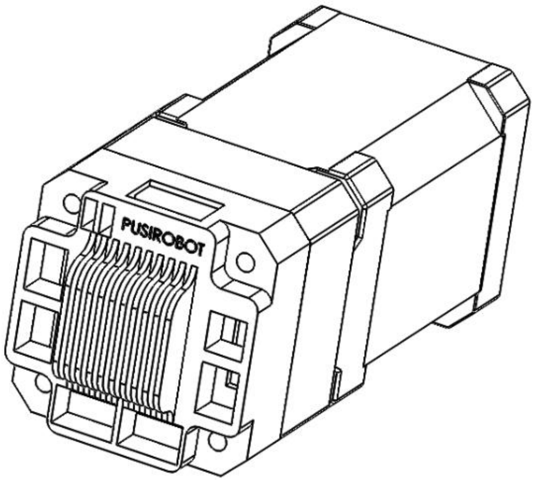
PUSIROBOT

CQPUSI ROBOT CONTROL SYSTEM

User Manual

PMC006x3s Series

Miniature Integrated Stepper Motor Controller



1. Version Control

1）Update Records

Date	Author	Version	Remark
2023-09-19	Tony	V0.1.0	Initial

Catalog

1 Introduction

1.1 Statement of intellectual property right

PMC006x3s series controller has been applied for the following national patent:

• Controller scheme and method have been applied for the protection of the invention patent.

• Controller circuit has been applied for the protection of utility model patent.

• Controller appearance has been applied for the protection of appearance patent protection.

Since PMC006x3s series controller has embedded firmware code, it would be considered as a violation of intellectual property protection act and regulations that any behavior of trying to destroy the function of firmware code protection. If this behavior acquires the software or other achievements of intellectual property protection without authorization of CQPUSI, CQPUSI has the right to stop such behavior by filing a lawsuit according to the act.

1.2 Disclaimer

The using method of the device and other content in the description of this manual is only used to provide convenience for you. To ensure the application conforms to the technical specifications is the responsibility of your own. CQPUSI does not make any form of statement or guarantee to the information, which include but not limited to usage, quality, performance, merchantability or applicability of specific purpose. CQPUSI is not responsible for these information and the consequences result caused by such information. If the CQPUSI device is used for life support and/or life safety applications, all risks are borne by the buyer. The buyer agrees to protect the CQPUSI from legal liability and compensation for any injury, claim, lawsuit or loss caused by the application.

2 Overview

2.1 General Description

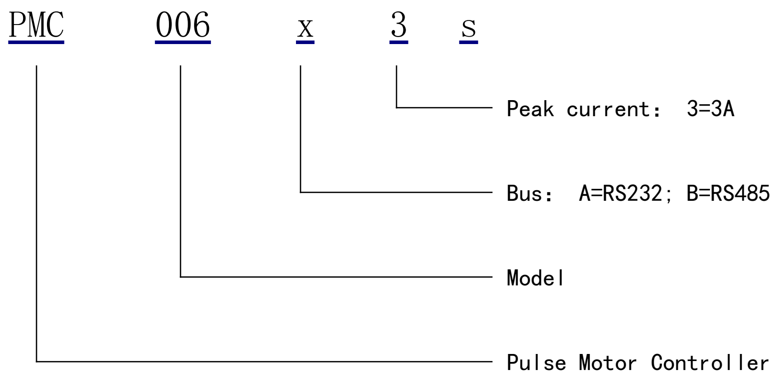
PMC006x3s is a miniature integrated stepper motor controller, which can be directly installed on the back cover of 28/42/57 series stepper motors. PMC006x3s provides a simple and rich set of control commands, which not only greatly simplifies the complexity of the upper control system, but also retains the flexibility of control to the greatest extent, and is suitable for a variety of high-precision and wide-range industrial applications.

2.2 Features

- ✓ Wide range of 9-36V single voltage supply
- ✓ Output current 0.2A ~ 3.0A, adjustable phase current by commands
- ✓ Automatic control of S curve acceleration and deceleration
- ✓ Start speed and stop speed can be configured by command
- ✓ external switches input ports
- ✓ Support 0/2/4/8/16/32/64/128 microstepping resolution
- ✓ Support 4/6/8 lines of 2 phase stepper motor
- ✓ Solenoid brake control function
- ✓ User-defined programs are written and executed offline
- ✓ Hot-plug protection
- ✓ Miniature size 42mm*42mm*26mm
- ✓ Precision aluminum shell, conducive to the protection and heat dissipation
- ✓ Reverse power supply protection
- ✓ One-key zero function
- ✓ Can be used as an analog input read (special version)

2.3 Production & Ordering Information

In order to serve you quicker and better, please provide the PMC006x3s product model number in following format when ordering.



Remark:

E：Closed-loop type.

Please be sure to contact the sales staff to confirm whether the required model is in a normal state of supply before placing an order.

3 Connector Description

3.1 Terminal port location

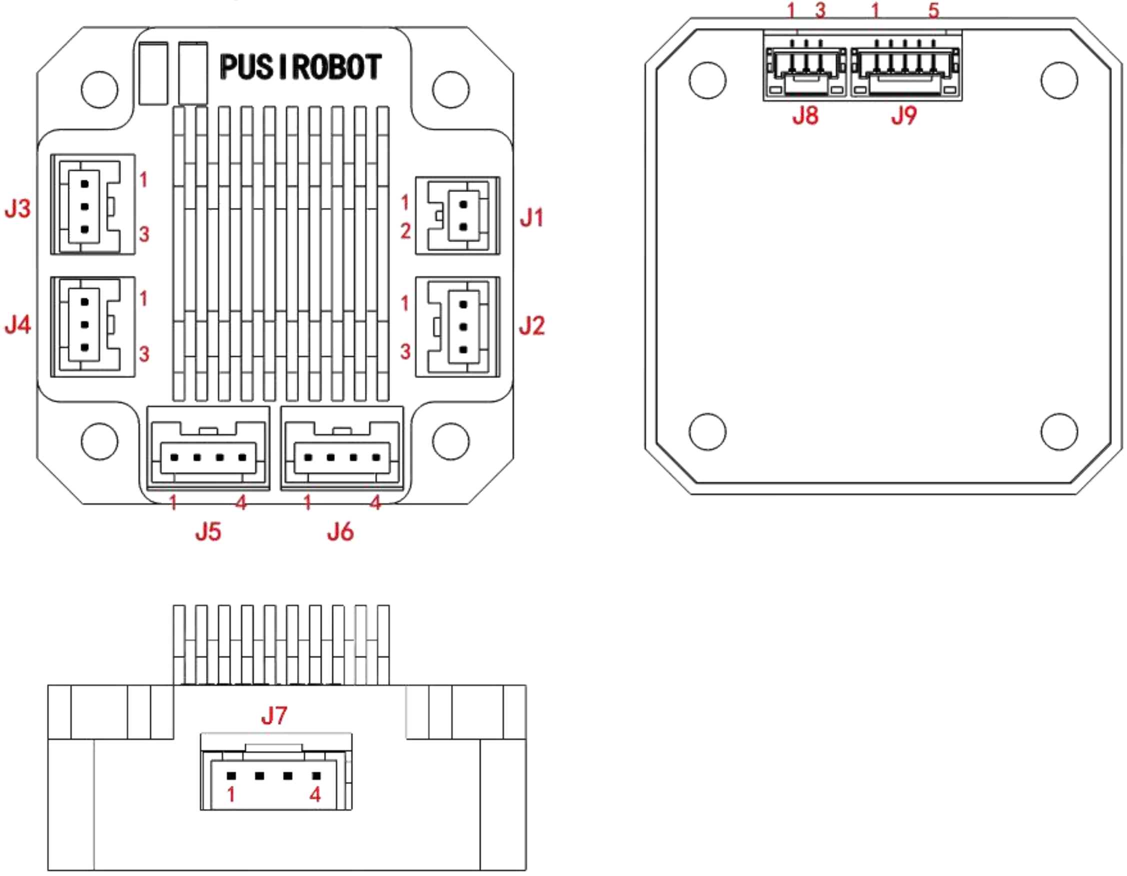


Figure 3-1-1

3.2 Motor connection J7

Pin no:	1	2	3	4
Designator:	M10	M11	M20	M21

Description:

M10: Connect to the stepper motor phase A+

M11: Connect to the stepper motor phase A-

M20: Connect to the stepper motor phase B+

M21: Connect to the stepper motor phase B-

WARNING: Incorrect connection of power or phase will permanently damage the controller!(In the case of closed loop, the red, blue, black, and green lines correspond to each other)

3.3 Power connection J1

Pin no:	1	2
Designator:	GND	VCC

Description:

VCC: Supply voltage, 9~36V

GND: Supply voltage ground

3.4 Solenoid valve connection J2

Pin no:	1	2	3
Designator:	Coil-	Coil+	NC

Description:

Coil+: Solenoid valve/brake positive control terminal, whose voltage is equal to the voltage of power supply VCC.

Coil-: Solenoid valve/brake negative control terminal.

NC: Reserved port, no function.

WARNING:The Coil+ pin is the same as the power supply voltage. Please be careful not to connect with DVDD,GND or other signal lines.

3.5 J3-Signal connection 1

Pin	1	2	3
Designator	GND	TXD	RXD

Description:

TXD: RS232 sender;

RXD: RS232 receiver;

GND: Digital ground

3.6 J4-Signal connection 2

Pin	1	2	3
Designator	GND	TXD	RXD

Description:

TXD: RS232 sender;

RXD: RS232 receiver;

GND: Digital ground

3.7 J5-Limit connection 1

Pin	1	2	3	4
Designator	EXT1-	EXT1+	GND	DVDD

Description:

DVDD: Controller +5V output, maximum 100mA;

GND: Controller digital ground;

EXT1+: External limit switch signal 1 input positive, 3~5.5V;

EXT1-: External limit switch signal 1 input negative

3.8 J6-Limit connection 2

Pin	1	2	3	4
Designator	EXT2-	EXT2+/AN	GND	OUT-VDD

Description:

OUT-VDD: GPIO8 controlled 5V output, 50mA max;

GND: Controller digital ground;

EXT2+: External limit switch signal 2 input positive, 3~5.5V;

AN: It can be used as an analog input interface in the special version, and the input voltage range is 0-3.3V, and the input can be 0-24V (or 10mA) in the special version. (Note: The analog input and the second limit port can only be realized by selecting one of the two functions.)

EXT2-: External limit switch signal 2 input negative.

3.9 J9-Encoder connection

Pin	1	2	3	4	5
Designator	DVDD	GND	A	B	Z

Description:

DVDD: Controller +5V output, 100mA max;

GND: Controller digital ground;

A: Encoder phase A signal;

B: Encoder phase B signal;

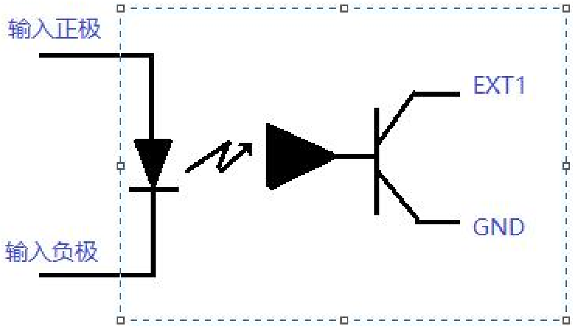
Z: Encoder phase Z signal.

3.10 RS232 Connection

PMC006x3s the connection mode of the controller under the stand-alone operation, the device address of each PMC006x3s controller is 0xff when it leaves the factory, and the stand-alone connection mode must be used when the device address is burned to the controller for the first time. When wiring, the GND cable of the 232 converter needs to be connected to the power ground or the digital ground on the controller.

3.11 Limit switch connection

PMC006x3s controller has two dedicated pins EXT1/EXT2 for connecting external limit switches, the trigger mode of each pin can be selected in real time through the instruction, the factory default value is effective for falling edge triggering, at this time the corresponding EXT1/EXT2 is from high level to low level jump, there is optodeaf isolation inside the controller, when the optocoupler input positive and input negative correctly form a loop, and are in the normal conduction state, then EXT1/EXT2 will change from high level to low level, Figure 3-11-1 below.



.

Figure 3-11-1

There are two types of common external limit switches, one is a four-wire through-beam optocoupler without an internal circuit, and the other is a three-wire or five-wire optocoupler with an internal circuit, which is divided into PNP type and NPN type. And there is a difference between light conduction and shading conduction. For different types of optocouplers, you can refer to the following wiring methods:

a) Four-wire optocoupler:

When entering the light, the light is turned on, the occlusion is triggered, and the rising edge trigger mode is configured, as shown in Figure 3-11-2. Note that in this case, EXT1/EXT2 both need the OUT-VDD pin of J6 as the power supply. If the DVDD pin of J5 is used, an external current-limiting resistor is required to avoid burning the light-emitting diode of the optocoupler.

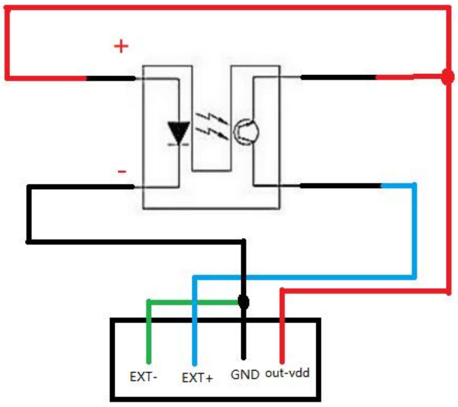


Figure 3-11-2

b) NPN type three-wire optocoupler:

When the light is in, it is turned on and the occlusion is triggered, and the rising edge trigger mode is configured, as shown in Figure 3-11-3

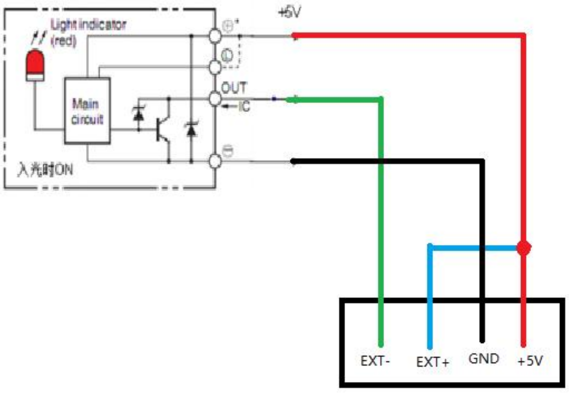


Figure 3-11-3

c) PNP type three-wire optocoupler:

When the light is turned on, the occlusion trigger is configured, and the rising edge trigger mode is configured, as shown in Figure 3-11-4.

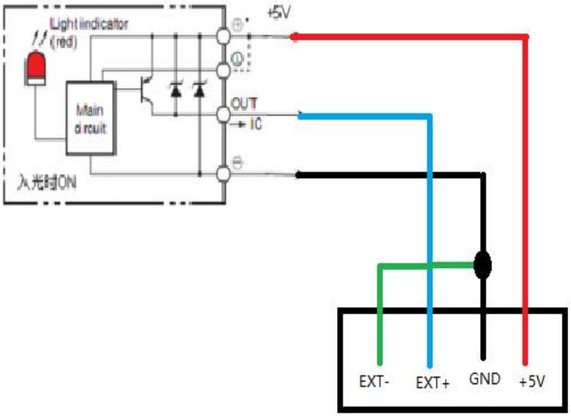


Figure 3-11-4

3.12 Mechanical Switch Connection

When using mechanical button switch or relay contact as a limit, it should be directly connected, EXT+ access DVDD, EXT- access GND, usually not conduction, press on, configure the falling edge trigger mode. Figure 3-12 below shows the following figure,

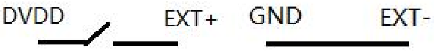


Figure 3-12

3.13 Analog speed regulation

In the special version, the PMC006x3s controller can use the analog speed control function in offline mode. In this case, the AN pin is used as an analog input port, so please contact sales in advance if you need to purchase it.

In the user's offline program, the program can be edited, so that the controller can enter the analog debugging function when it is powered on or when receiving the input signal, from the wiring, the AN pin needs to be used as the analog input port, the input voltage range is 0~3.3V, and the negative pole is connected to the digital signal GND. The maximum speed corresponding to 3.3V and the minimum trigger threshold voltage at a speed of 0 can be set in the offline program.

4 Instruction Set

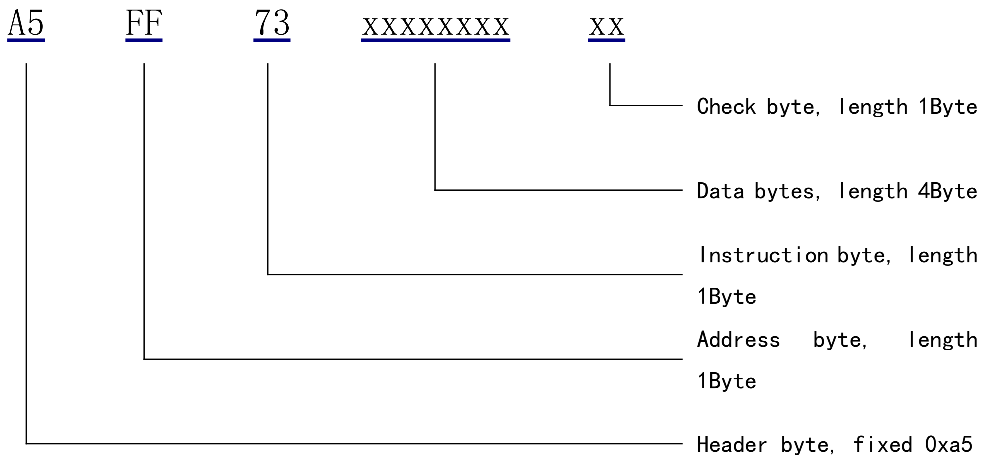
PMC006x3s controller uses a set of reduced instruction sets to standardize and simplify the operation of the host computer on the controller, the controller receives the operation instructions sent by the host computer, and returns the ACK to the host computer after execution, and returns the data or status required by the user.

Half-duplex communication is adopted between the host computer and the PMC006x3s controller, and the information at the same time can only be transmitted from the host computer to the controller or the controller to the host computer in one direction, and the PMC006x3s controllers cannot communicate with each other.

4.1 Instruction Structure

The communication between the PMC006x3s and the host computer adopts a transparent transmission format based on RS232 strings, and the frame content is transmitted sequentially in bytes, and each frame is fixed at 8 bytes in length.

Instruction Structure::

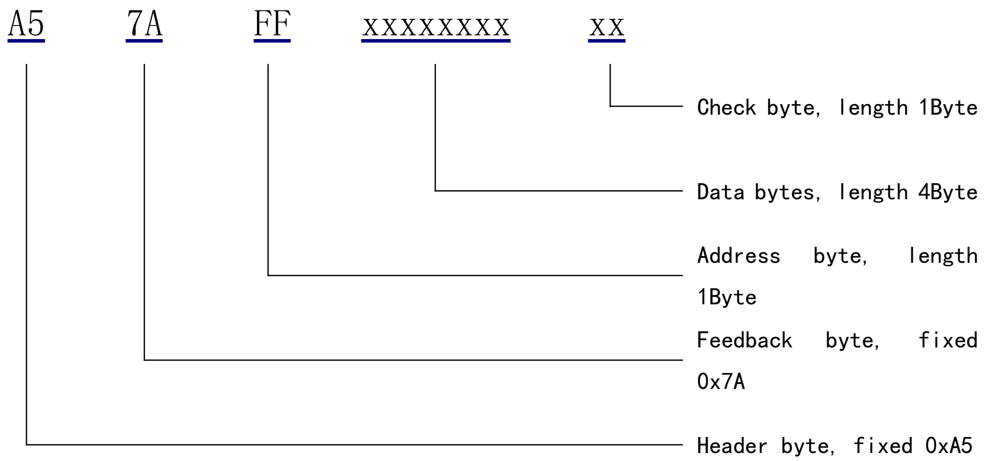


The length of command frame which is transmit to controller by host controller and the acknowledgement frame which is responded to host computer by controller is fixed 8 bytes (without any spaces or separators). If the controller sends out an instruction frame which has valid address and is verified correctly, the PMC006x3s controller returns an answer frame (even if the instruction is invalid).

Both 8 bytes of the command frame and the response frame must be transmitted before the bus can be released, otherwise the host computer or the slave machine will wait for a timeout.

When the address and the check byte in the instruction frame that the host computer sends out is not correct, it will not receive any response frame. At this time the host computer will continue to send an effective instruction frame after its own timeout mechanism is effect, which will not cause the bus conflict or hang up.

4.1.1 Return Value



The returned data varies depending on the instruction sent and the response.

1. When the instruction is a read instruction, the data byte is displayed as the specific instruction data;

2. When the command is a set instruction, the response value is the last read value. Some setting instructions can write 0 to read the function;

3. Error code: When sending a write command, the returned data contains a specific error code. The definition is as follows:

06: The operation status is not satisfied; or write in an allowable state: 0x 00000006

07: Unknown command or Still trying to go up while on optocoupler position: 0x 00000007

(Note: When the response value is an error code, this instruction is considered to have not been written normally.))

4.1.2 Address Byte

When the host computer sends out the instruction, the byte value range is 0x1~0x32(The default address is 0xff when leaving the factory). Because 232 communication does not have the function of networking, so each COM port has only one site, if the set site number is beyond the value range, the subsequent communication will not be able to be carried out, and can only be restored by reset or re-erasing; when the PMC006x3s controller feedback instruction, the byte is fixed at 0x7a.

4.1.3 Instruction Byte

When the host computer sends a command, the byte content is the corresponding instruction. See details the 4.2 instruction summary table.

When the PMC006x3s controller feeds back instruction, the byte is the controller’s address.

4.1.4 Data Byte

When the host computer sends an instruction, the content of the 4 bytes is the data required by the corresponding instruction, and for the instruction that does not require data, the 4 bytes can be any value.

When PMC006x3s controller feedback commands, the 4 bytes are the data or status returned, and the 4 bytes should be ignored for feedback that does not require data.

Whether it is a host computer or a controller, the 4-byte data type is defined as a signed integer (ANSC C: signed long), and the low byte is always sent or received first when transmitted, that is, big-end mode, such as data 0x12345678, the transmission order is 0x78 -> 0x56 ->0x34 ->0x12, and the input instruction is 78563412.

4.1.5 CRC Byte

The check byte is defined as the sum of the previous 7 bytes (discarding carry); in the instruction frame, the host computer calculates and transmits. In the acknowledgement frame, the PMC006x3s controller calculates and transmits. The frame whose check byte is not correct will be discarded without processing.

4.2 Instruction Set Summary

Instruction	Description	Data range	Note
4(0x34)	Solenoid valve/brake control	0~100	
7(0x37)	Write user data	0~0xffffffff	Stored in EEPROM
8(0x38)	Back to zero	1~0x7fffffff	The parameter is the maximum stroke of the structure
9(0x39)	Set the speed back to zero	1~256000 PPS (step/s)	
F(0x46)	Set offline execution	0/1	Can be saved after power-down
I(0x49)	Abort the current step command	0	Stop immediately, no deceleration
J(0x4a)	Set ext_stop trigger mode	0~3	Can be saved after power-down
L(0x4c)	Set startup speed	65-30000	Can be saved after power-down
M(0x4d)	Set encoder CPR	200-1600	Can be saved after power-down
N(0x4e)	Set speed mode enable	0/1	
Q(0x51)	Automatic Current Attenuation Factor	0-3	Can be saved after power-down
S(0x53)	Set stop speed	65-30000	Can be saved after

			power-down
z(0x5a)	Test run the current user program	----	
a(0x61)	Set the automatic current decay enable	0/1	Can be saved after power-down
b(0x62)	Clean flag of ext_stop1	----	
d(0x64)	Set direction of rotation	0：backward，1：forward	
e(0x65)	Set peak phase current	400~3000（Unit：mA）	Can be saved after power-down
f(0x66)	Set the external emergency stop	0/1/2/3	
g(0x67)	Set offline enable	0/1	
i(0x69)	Set current position	0-0x7fffffff	
l(0x6c)	Clean flag of ext_stop2	----	
m(0x6d)	Set microstepping	0/2/4/8/16/32/64/128/256	Can be saved after power-down
o(0x6f)	Set deceleration coefficient	0-8	Can be saved after power-down
s(0x73)	Run setting steps	1~0x7fffffff	Sent successfully start running
t(0x74)	Save all parameters after power-down	----	Save other parameters，can not write them in process
u(0x75)	Set acceleration coefficient	0-5	Can be saved after power-down
v(0x76)	Set maximum speed	1~256000 PPS (step/second)	
w(0x77)	Set device address	1~120	Can be saved after power-down
y(0x79)	Write IO port	0~0xff	
:(0x3A)	Set silence coefficient	0~7	Default 0，minimum 7
6(0x36)	Read user data	----	
K(0x4b)	Read ext_stop trigger mode	----	
O(0x4f)	Read status 2 of controller	----	
P(0x50)	Read device ID	0~0xffffffff	
R(0x52)	Read firmware version	0	

	information		
c(0x63)	Read current position	----	
h(0x68)	Read deceleration coefficient	----	
j(0x6a)	Read status 1 of controller	The definition of Bit [3:0] is explained in detail in the instruction	
k(0x6b)	Read microstepping setting	----	
;(0x3B)	Read silence coefficient	----	
x(0x78)	Read level value	0~0xff	

4.3 Instruction Details

In the tool package provided by CQPUSI, the VC++-based communication interface DLL and related examples are included, and the user program can directly call the interface function of the DLL to operate the PMC006x3s controller without knowing the specific sending method and feedback information of each instruction. The purpose of some of the directives is explained in detail below.

4.3.1 Step Instruction

The step instruction 0x73 can control the stepper motor to rotate specified number of steps according to the setting direction, speed, acceleration. The number of steps is calculated according to the current microstepping settings.

If the controller is busy, stepping commands will be ignored. So, you should first check the controller's status bits through the 0x6A command. For a detailed explanation, please refer to the description of controller status.

Host computer send: 0xa5 0x01 0x73 0x1f 0x01 0x00 0x00 0x39

Controller ACK: 0xa5 0x7a 0x01 0x03 0x00 0x00 0x00 0x23

Description: Send a command to the controller whose address is 0x01 to rotate 287.

4.3.2 Set micro-stepping resolution

PMC006x3s controller supports maximum 128 microstepping resolution。It would not cause loss of accuracy when the controller is converted from a low microstepping to a high microstepping. However, it would cause loss of accuracy when the controller is converted from a high microstepping to a low microstepping, because the motor rotor is in a middle position.

When the acceleration value is set high, it may cause the motor out of step in a low microstepping. In this condition, it's better to increase the microstepping.

The setting value of microstepping will be stored after the controller is powered off. And also the final configuration is enabled automatically when the controller is powered on next time

Host computer send: 0xa5 0x02 0x6d 0x04 0x00 0x00 0x00 0x18

Controller ACK: 0xa5 0x7a 0x02 0x04 0x00 0x00 0x00 0x25

Description: Set the number of subdivisions of the controller with the address of 0x02 to 4.

4.3.3 Set acceleration and deceleration

PMC006x3s controller supports the S-curve automatic acceleration and deceleration function, without the need for additional module support. When the acceleration/deceleration/start-up speed/stop speed is set by the command, the controller calculates the acceleration curve in real time until the rotational speed reaches the set maximum speed, and then starts to calculate the deceleration curve and controls the motor to decelerate, as shown in Figure 4-1 below.

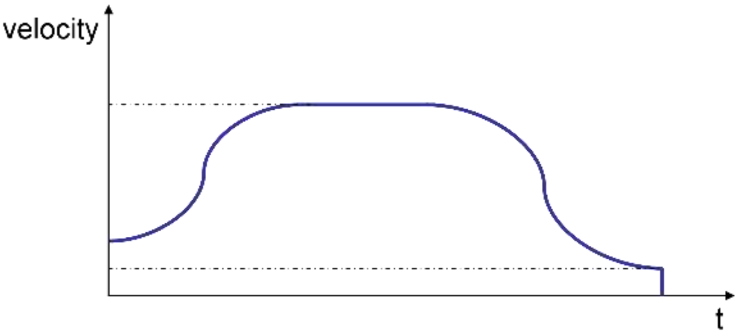


Figure 4-1

The acceleration coefficient and deceleration need to be configured separately, but they have the same dimension, and the correspondence between the coefficient value and the acceleration value is as follows. Acceleration and deceleration support 1~8 a total of 8 gears, and the corresponding acceleration values of each gear are as follows.

Coefficient	Acceleration PPS^2
0	Acceleration and deceleration not be enable
1	77440
2	48410
3	27170
4	21510
5	14080
6	10460
7	6915
8	5210

When the start command is started, the controller controls acceleration and deceleration adaptively, in order to make the motor run the specified number of steps at the expected speed. Therefore, when the step numbers are few and the setting velocity is high, the actual acceleration and deceleration coefficient may increase automatically.

When the step numbers are less than 256, the controller would not use the method of S curve automatic acceleration and deceleration to control, but directly control the motor decelerate to the stop speed.

When the speed change process of the motor runs, it jumps from 0 to the starting speed, then accelerates to the maximum speed, runs to the stop stage, decelerates from the maximum speed to the stop speed, and then jumps from the stop speed to 0. The start-stop speed is fine-tuned.

4.3.4 Read motor position

Whenever a rotation command is issued, according to the number of steps of the command the controller automatically records the current position which is represented by a signed integer. A positive value indicates clockwise position, while a negative value indicates the anti clockwise position.

Because the value of current position is calculated by the number of steps, when a user needs to change the microstepping resolution, you should first read the current location information, and then reset the position by the 0x69 instruction to avoid error of conversion location.

When the controller is powered off, the position information is automatically cleared.

4.3.5 Set peak phase current

PMC006x3s controller has a built-in current closed-loop control circuit, the maximum range of current setting is 400~3000mA, and the recommended setting current should not exceed +-20% of the rated current of the motor. Generally, according to the rated current setting, in the application of high and low speed, the appropriate increase or decrease, low speed adjustment, high speed adjustment.

Note: When the motor works at a very low speed, it is required to pay attention to the heating of the motor and the controller. Usually current only needs to set a smaller value in this condition.

4.3.6 Set the external emergency stop

PMC006x3s controller provides a limit switch input port EXT1 that can be used as an emergency stop function in open-loop or closed-loop mode. You can set whether to enable the limit function through the 0x66 command, the data value range in this command is 0~3, 0 means that neither is enabled, 1 means EXT1 is enabled, 2 means EXT2 is enabled, and 3 means EXT1 and EXT2 are enabled.

When the emergency stop function is enabled, if the corresponding input pin detects a valid low level, the controller will lock the motor and stop responding to the stepper command, and the user can query the 1st and 2nd bits of the status bit through the 0x6a command to see which one triggered the emergency stop. The controller continues to respond to new step commands only after the user uses the 0x62 0x6c command to clear the corresponding status bit. Otherwise, the motor cannot be operated.

4.3.7 Set automatic current decay

Use this command to set whether to automatically attenuate the phase current when the controller does not receive a step command. The default attenuation amplitude is 50%, and the user can set the attenuation coefficient by 0x4d command, the higher the value, the greater the attenuation amplitude. When the phase current is more than 2.5A, it is recommended to set it to 2 or more.

0: 100% 1: 50%; 2: 25%; 3: 12.5%;

The setting can not be set to 0 all the time, otherwise the motor is also running at full power when idle, and there is no work consumption, all will be converted into heat, and the heating is serious, which will affect the service life of the motor itself.

4.3.8 Set speed mode

Use this command to set the controller to enter the speed mode in which the motor will continue to run at the setting speed. In speed mode, can use this command with 0 parameters or use the 0x49 command to exit the speed mode. Notice the difference between these two instructions: Set the speed mode to 0 will make the motor gradually stopped, and the use of the 0x49 command will immediately stop the motor.

Speed mode operation does not change the operating speed in the state of motion.

4.3.9 Set start speed and stop speed

Use 0x4c and 0x53 commands to set the motor start speed and stop speed, the default value is 600pps, and the minimum value is 65pps.

Note: When any one of these two values is higher than the setting value of maximum speed, the step command will not work.

4.3.10 Set encoder CPR

During the closed loop, the controller recognizes and counts the encoder according to the parameters set by the 0x4d, and the values are 200/400/500/600/800/1000/1000/1024/1200/1600.

4.3.11 Solenoid control

This command switches the solenoid valve/brake and energy-saving control. The value range is 0~100. When the value is 0, disconnects the solenoid valve for power supply. When the value is 100, the solenoid valve is supplied with 100%power supply voltage. The intermediate value will perform energy-saving control for the solenoid valve according to the proportion. The solenoid can be opened and closed by 0x34 command.

4.3.12 Set offline enable

Using the command can set the offline enabled or disabled. When the function is enabled, the controller releases the control of motor immediately, the current step instruction is terminated, and the phase current is reduced to 0. All subsequent step instructions send by host computer will not be processed, until the user use 0x67 instruction to set offline disabled.

The difference between 0x67 and 0x49 is: The 0x49 only can stop the current step instruction, and the phase current is not 0. The 0x49 will not set status flag, dose not to prevent the execution of the next step instruction.

4.3.13 Zero-back instruction

The command data is similar to the step number command, and the default reverse is to run towards the photoswitch position. After sending the command, the controller internally determines the port level status of EXT1. When the EXT1 port is at a low level, that is, it does not run to the photoelectric switch position, the motor directly runs in reverse until it blocks the optocoupler, triggering the limit stop function. When the EXT1 port is at a high level, that is, the photoelectric switch has been blocked, at this time, the motor will first run in the forward direction, away from the optocoupler position, and then run in the opposite direction, triggering the optocoupler again.

Generally, the data of the zero return instruction is greater than the whole process, so as to ensure that any point in the stroke can return to the photoelectric switch position. It is also necessary to ensure that the position of the automatic distance away from the optocoupler is determined to be out of the occlusion of the photoelectric switch. The default number of steps away from is 1/16 of the trip back to zero.

4.3.14 GPIO port operation

The AN/GPIO8 of the PMC006x3s controller can be reused as a general-purpose input and output port. This can be done by 0x79 commands, both of which operate on two bytes, and the position of the 12 bits in the bytes from high to low is:

{Z(bit11)、B(bit10)、A(bit9)、GPIO9(bit8)、GPO8_DIR(bit7)、AN_DIR(bit6)、GPIO8_PUD(bit5)、AN_PUD(bit4)、AN_EN(bit3)、EXT1(bit2)、GPIO8(bit1)、AN(bit0)},

Z, B, A, GPIO9, EXT1, AN (EXT2) are input ports, which cannot be configured as outputs, nor can the initial level be set, and the default is high. GPO8 is the output port, the default 5V output, can be controlled by command. AN_EN is an analog enable control port. GPO8_DIR, AN_DIR, the GPO8_PUD configuration does not take effect.

When using the port output command 0x79, only the input form of AN and the level of output GPIO8 can be selected.

The upper computer sends out: 0xa5 0x02 0x79 0x82 0x01 0x00 0x00 0xA3

Controller Response: 0xa5 0x7a 0x02 0x00 0x00 0x00 0x00 0x21

Analysis: Output GPIO8 high.

4.3.15 Level status read operation

Level status reading is performed by 0x78 commands. The valid bit bit is bit0-bit3, and the rest of the bits are invalid, where:

{bit3: FSET pin level; bit2: EXT1 pin level; bit1: GPIO pin level; bit0: EXT2(AN) pin level}

When the bit value is 1, the level state is high, and when the bit value is 0, the level state is low.

Example:

The host computer sends out: 0xa5 0xff 0x78 0x00 0x00 0x00 0x00 0x1c

Controller Response: 0xa5 0x7a 0xff 0x03 0x00 0x00 0x00 0x21

Analysis: The controller of site 255 reads the level status, the reading value is 0x03, and the bit1 and bit0 are 1, and bit2 and bit3 are 1, that is, the GPIO pin level, the EXT2(AN) pin level is high, the EXT1 pin level, and the FSET pin level is low.

4.3.16 Query controller status

The controller's status has two registers, which can be read by the command 0x6a and 0x4f. Respectively, 0x6a command is used to read the state 1 of the controller. There are a total of 8 bits in the status, the definition is as follows:

Bit	Designator	Description
0	STATUS	0: IDLE; 1: Busy
1	EXT1	0: No emergency stop; 1: emergency stop
2	EXT2	0: No emergency stop; 1: emergency stop
3	AUTO_DEC	0: auto decay disable; 1: auto decay
4	STALL	0: no stall; 1: stalled
5	Reserved	
6	HOME	0: NOT Homing; 1: Homing
7	UVL	Supply voltage under dedicated value

Note: When the zeroing operation is executed, the BUSY flag does not indicate whether the zeroing is over, and should be judged by the HOME flag.

8 bits of the status register 2 are defined as follows:

Bit	Designator	Description
0	FREE_RUN	0: Offline disable; 1: Offline enable
1	EXT1_EN	0: Ext_stop disable; 1: Ext_stop enable
2	Coil	0:solenoid close;1: solenoid open
3	SPEED_MODE	0: displacement mode; 1: speed mode
4	DIR	0: backward; 1: forward
5	OFFLINE_AUTO	0: Not automatically run in offline mode; 1: automatically run in offline mode
6	EXT2_EN	0: Ext_stop disable; 1: Ext_stop enable
7	OTS	0: Not overheated; 1: Overheated

4.3.17 Parameters save

All parameters that can be saved when power down are indicated in the summary table in the 4.2 section. These parameters can be saved in EEPROM by sending 0x74 commands to controller. Because all the non-lost memory has a limited erase life, it is recommended that the user use the command only in the case of the real need.

It will take effect immediately after the address of controller is modified. So, when modify the address of controller by using the debug tool, user should click “scan” button to retrieve the site number after write the parameters of controller. Then, click “save all parameter” button to save address to EEPROM.

5 User customization program

PMC006x3s can be configured into offline mode. In this mode, controller automatically execute custom user code after powered on, the code is compiled and in advance burned to the EEPROM through CQPUSI tool software. Please refer to the "controller offline programming guide" for details about operation method and paradigm.

When the PMC006x3s controller works in offline mode, RS232 communication interface is still responsive to the user's online instruction.

The maximum number of instructions that PMC006x3s controller supports for user is 100.

5.1.1 User Instruction Set Summary

PMC006x3s controller supports the following user defined instructions, these commands are provided by the CQPUSI tool software to interact with the controller automatically, and users do not need to write their own programs, only need to operate the command in the "custom programming interface".

Command	Function	Options	Data range
ROT	Rotate a given number of steps	0	1~0x7fffffff
MIC	Set microstepping	0	0/2/4/8/16/32/64/128
DIR	Set direction of rotation	0	0: reverse, 1: Forward
EXTEN	Set ext_stop enable	0	0/1
FREE	Set freerun enable	0	0/1
CLR2	clean ext_stop2 flag	0	----
CLR1	clean ext_stop1 flag	0	----
VSET	Set speed(pps)	0	0-6000PPS
ACC	Set Acceleration	0	3-6
TRIG	Set external trigger mode	0	0-3
CNTI	Internal counter plus1	0	----
CNTC	Internal counter reset	0	----
JMP	Unconditional jump	0	0-50

JNE	Unequal jump	0	0-50
JEQ	Equal jump	0	0-50
WAIT	Waiting condition	1-7	0/1
OUT	Port output	1-5	0/1
CMP	Comparison	1-6	0~65536
RESET_EN	Set GPI1 reset enable	0-1	0-50
PAUSE_EN	Set GPI2 pause enable	0-1	0

5.1.2 User command description

The detail information for some of command is described as following.

5.1.2.1 CNTI，CNTC command

These two instructions are used to add and reset the internal counter, and the internal counter can be used as a function of the cycle count in the user's custom program. The value of the counter can be used as a comparison condition in the CMP command.

5.1.2.2 JMP command

Unconditional jump instruction, the program jumps to the specified location.

5.1.2.3 JNE，JEQ command

Conditional jump instruction. Based on the flag which is generated by CMP instruction, jump to the specified position. If the flag bit is 1, the program will jump to the specified position by the JEQ instruction; if the flag is 0, the program will jump to the specified position by the JEQ instruction.

5.1.2.4 WAIT command

Pause the program execution. Execute the next instruction until the condition of the option is satisfied. A total of 9 options can be selected, please refer to the CQPUSI tool software for "custom programming" interface settings for details.

Note: when using the ROT command to launch the motor rotation, the next instruction is executed immediately and don’t wait until the rotation command is completed. So generally, there should be a WAIT instruction following the rotation command.

5.1.2.5 OUT command

Output value to GPO1~5. The instruction can only output a port value at a time.

5.1.2.6 CMP command

Compare the value of option with the setting value. Option can be the value of the internal counter, or any one input port or an external stop status or all input ports as a bus data comparison, a total of 9 options can be selected. After comparing the internal flag will be set, if the result of comparison is equal, the flag is set to 1, otherwise set to 0.

5.1.2.7 RESET_EN and PAUSE_EN command

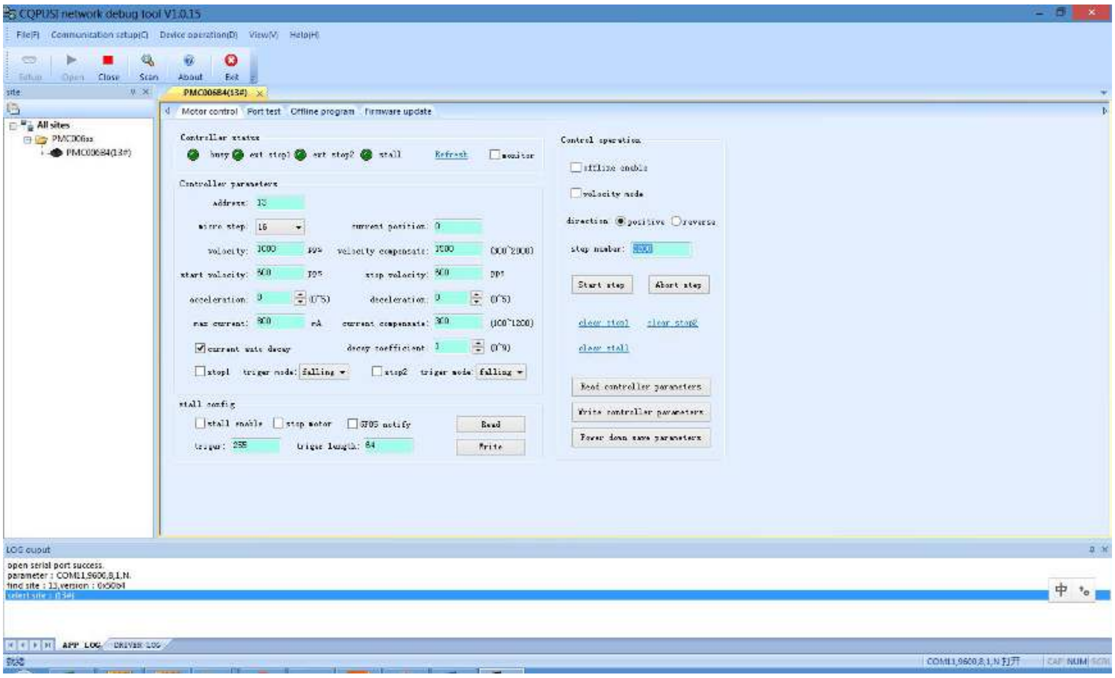
The controller can choose to bind GPI1 as the external reset stop key input, GPI2 as external pause/start key input. These functions can only be enabled by the offline program and only need to be done once in the program to take effect globally, so the user should try to put these two statements in the beginning of the offline program. When the external suspension / launch function is enabled, the low-level pulse on the GPI2 will alternately start or pause the execution of the offline program, but the rotation command which has been issued will not be stopped. When the external reset stop function is enabled, the low-level pulse on the GPI1 will immediately stop all operation instructions, including the rotation command being executed, and put the program pointer to the set position.

6 Introduction to Debug Tool

PMC006x3s can use the CQPUSI tool software Tool Debug to set control parameters of motor, port detection, custom programming.

6.1.1 Main GUI

Firstly, according to System Settings, select the serial port number. And the port is configured to 9600, n, 8, 1 format, open the serial port. Then you can click the “Scan” button to scan the site. All sites connected to the bus will be display to the left of the window. User can double-click to select any of the sites to operate.



Controller status

● busy ● ext stop1 ● ext stop2 ● stall □ monitor

Controller parameters

address: 13

micro step: 16

current position: 0

velocity: 1000 pps

velocity compensate: 1500 (800~2000)

start velocity: 500 pps

stop velocity: 500 pps

acceleration: 0 (0~5)

deceleration: 0 (0~5)

max current: 800 mA

current compensate: 300 (100~1200)

☑ current auto decay

decay coefficient: 1 (0~9)

□ stop1 triger mode: falling

□ stop2 triger mode: falling

stall config

□ stall enable □ stop motor □ SFOS notify

triger: 255

triger length: 64

Control operation

□ offline enable

□ velocity mode

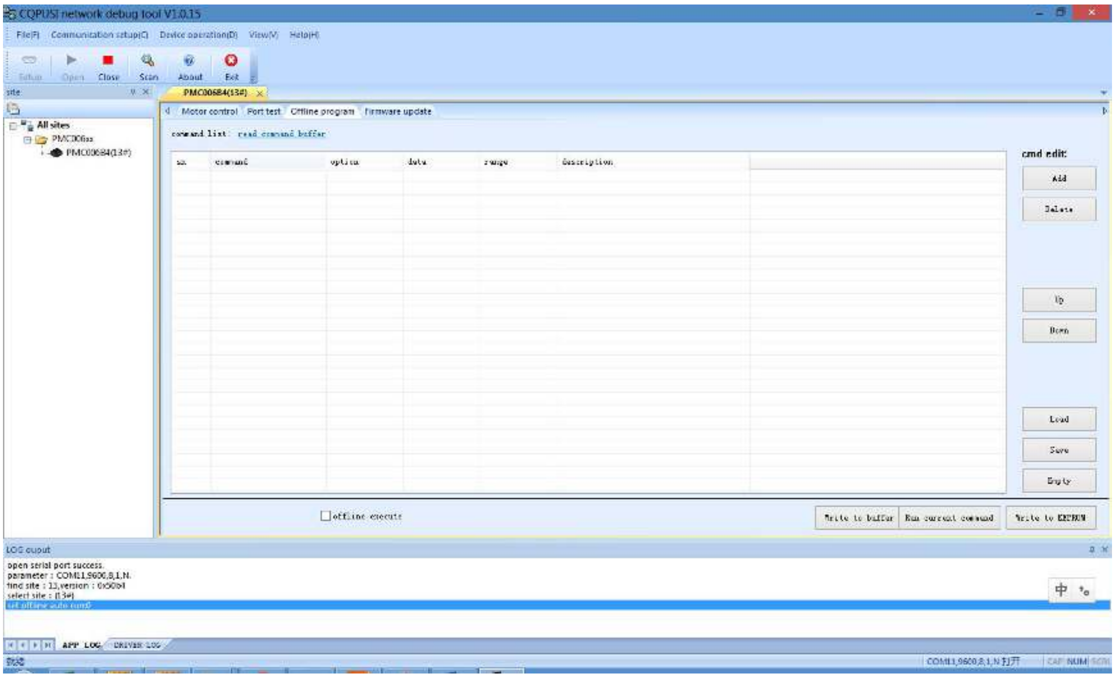
direction ● positive ○ reverse

step number: 3200

```
open serial port success.
parameter : COM11,9600,8,1,N.
find site : 13,version : 0x50b4
select site : (13#)
```

COM11,9600,8,1,N 打开

6.1.2 Custom Programming Interface



command list:

sn.	command	option	data	range	description

□ offline execute

```
open serial port success.
parameter : COM11,9600,8,1,N.
find site : 13,version : 0x50b4
select site : (13#)
set offline auto run0
```

COM11,9600,8,1,N 打开

Click the "offline programming" button in the main interface to enter this interface. If there is already a user instruction in the PMC006x3s controller, click the "read command buffer", all instructions can be read and displayed automatically. Users can operate the instruction through the "insert", "delete", "up", "down" and other buttons.

Once editing is completed when users need to do online debugging, first press the "write cmd to the buffer" button to download the program to on-chip memory of

PMC006xx Controller, and then press the "try run current command" button, and then debug instruction pointed by the cursor. After confirmation, press the "Write to EEPROM" button to burn all program to non-volatile memory. If you select the "offline exec", PMC006xx controller will automatically run the program which has been already burned next time power is on.

After successful commissioning instruction, you can press the "Save" button to save the instruction to your local disk, so that when the batch burn is needed, users can only need to click the "Open" button to read the program stored in disk, then press "write instruction to the buffer" and "Write to EEPROM" button to download the program to the controllers.

Note: Once the user enters the offline programming interface, it will be prohibited that user defined program is automatically executed even if there is no operation in the interface. Therefore, when user leaves this interface, need to click "off line execute" button to make it open. Otherwise, the user defined program will not be automatically run next time controller powers on.

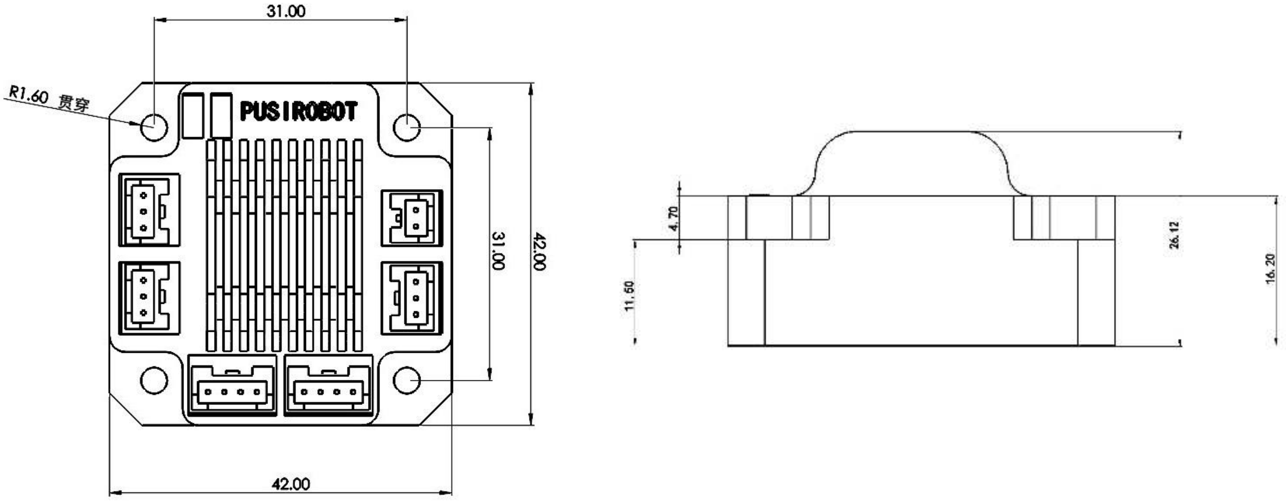
6.1.3 Routine program

CQPUSI tool software Debug Tool is mainly used to test motor control parameters through a visual interface, which is convenient for direct writing or editing in the program in the future. There is a complete python routine for running tests on the upper computer, and you can contact the sales staff if necessary.

7 Electrical Characteristics

Parameter	Condition	Min	Typical	Max	Unit
Supply Power Voltage	Normal 25℃	9	24	36	V
Operation Temperature	12V DC	-20		85	℃
IO maximum current	source/sink current	0		20	mA
Each phase output current	Normal 25℃	0	2.5	4	A

8 Dimensions（Unit：mm）


31.00
R1.60 贯穿
PUSIROBOT
31.00
42.00
42.00
4.70
11.50
26.12
16.20