IO,Modbus&Remote FunctionUser Manuel

<<<<

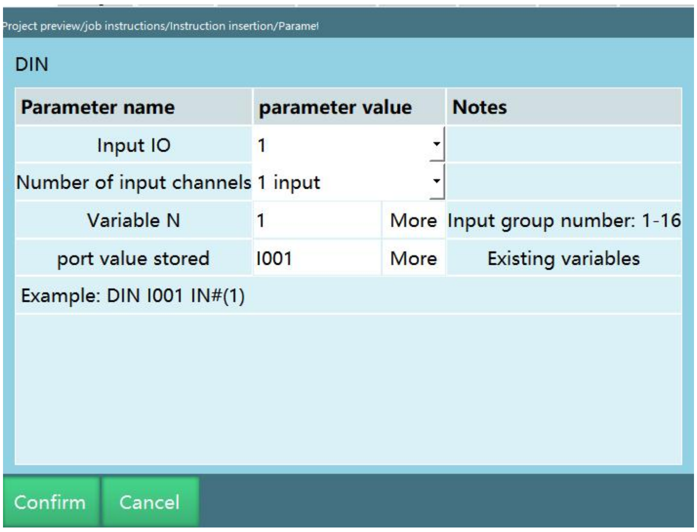
Catalogue

IO, Modbus and Remote Program

> IO

> Input/Output instructions

DIN-IO input



Project preview/job instructions/Instruction insertion/Paramet

DIN

Parameter name	parameter value		Notes
Input IO	1		
Number of input channels	1 input		
Variable N	1	More	Input group number: 1-16
port value stored	I001	More	Existing variables

Example: DIN I001 IN#(1)

This instruction is used to read the digital input status into a variable that can be local/global integer (INT, GINT) variable or local/global Boolean (B, GB) variable.

Input IO board: You can select from the IO boards 1-4.

Number of input channels: IN#-1 channel input, at this time, 1 channel is 1 group, and groups 1-16 correspond to ports 1-16 respectively;

IGH#-4 channel input, at this time, every 4 channels is a group, that is, 1-4 channel ports, 5-8 channel ports, 9-12 channel ports, and 13-16 channel ports are 1-4 groups respectively. The group number can be filled with 1-4. If you want to read the input status of 5-8 channel ports at the same time, you can fill in the group number 2.

IG#-8 channel input, at this time, every 8 channels is a group, that is, 1-8 is the 1st group, and 9-16 is the 2nd group. If you want to read the input status of ports 9-16 at the same time, fill in the group number 2.

If multiple channel ports are read at the same time, the port status will be converted to decimal and saved into the variables. And the group number being read can be obtained from the corresponding variable.

For example, read 5-8 channel ports while there are 4 channels, their status are as follows, and will be stored in I001

1	2	3	4
0	1	1	0

Then the binary value is 0110, which is converted to 6 in decimal.

Then it is saved in the system as DIN I001 IGH#(1) 6

For example, read 9-16 channel ports while there are 8 channels, their status are as follows, and will be stored in GI001

16	15	14	13	12	11	10	9
0	1	1	0	1	0	0	1

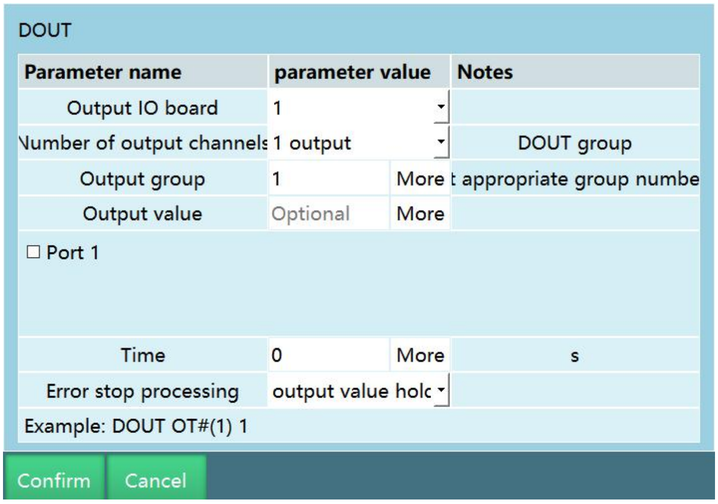
Then the binary value is 01101001, which is converted to 105 in decimal.

Then it is saved in the system as DIN GI001 IG#(2) 105

Input group number: It can be set to read the 1/4/8 channel input status at the same time, or set through the variable values of the binding variable.

Port value storage: Store the value read from the IO input into the selected variable.

DOUT-IO output



DOUT

Parameter name	parameter value		Notes
Output IO board	1		
Number of output channels	1 output		DOUT group
Output group	1	More	t appropriate group numbe
Output value	Optional	More	

□ Port 1

Time	0	More	s
Error stop processing	output value holc		

Example: DOUT OT#(1) 1

This instruction is used to output digital signals through the digital IO board.

Output IO board: Select the IO board you need to output, you can choose from 1-4.

Number of output channels: OT#-1 channel output, at this time, 1 channel is a group, and groups 1-16 correspond to ports 1-16 respectively.

OGH#-4 channel output, at this time, every 4 channels is a group, that is, 1-4 channel ports, 5-8 channel ports, 9-12 channel ports, and 13-16 channel ports are 1-4 groups respectively. The group number can be filled with 1-4. If you want to output the 5-8 channel ports at the same time, you can fill in the group number 2.

OG#-8 channel output, at this time, every 8 channels is a group, that is, 1-8 is the 1st group, and 9-16 is the 2nd group. If you want to output the ports 9-16 at the same time, fill in the group number 2.

Output group number: It can be set to output 1/4/8 channel IO at the same time or set through the variable values of the binding variable.

Output value: You can select "Optional" or choose to output through variables, or set through the variable values of the binding variable.

If "Optional" is selected, then check the status of each port in each group of IOs, and the output is 1 if checked and 0 if unchecked.

If you choose to output through variable, the variable value will be converted from decimal to binary at the time of output, as shown in DIN.

Time: Wait for the specified time after the instruction is executed, and then invert the output.

Error stop processing: In the process of IO signal output, an error alarm is generated. The IO signal will make different processing methods.

If you select "Output value hold", then while the program is running, the port output will remain as is and the timing time will be suspended when an alarm is triggered or in case of other unexpected situations. When the alarm error is cleared and the program begins to run normally, the IO output timing continues for the remaining time before the pause, and the port will be inverted when the timing is over.

If you select "Time-out stop", no matter what the situation is, as long as the port timing is over, the port value will be inverted, and it will not be affected by the pause or error.

AIN-Analog input

This instruction is used to read a single port input value from an analog IO board into a variable.

Analog input port: Select the input port to be read.

Variable name: Please select the variable name of the variable you need to read into, such as GD001.

AOUT-Analog output

This instruction is used to set the output value of a single port of the analog IO board. The output value can be a floating point number.

Analog output port: Select the output port whose value needs to be set.

Variable value source: Please select global floating point GDOUBLE or local floating point DOUBLE variable or hand-filled value

PULSEOUT-Pulse output

This instruction is used to control the pulse output of the IO board that supports PWM.

Number: Total number of pulses output.

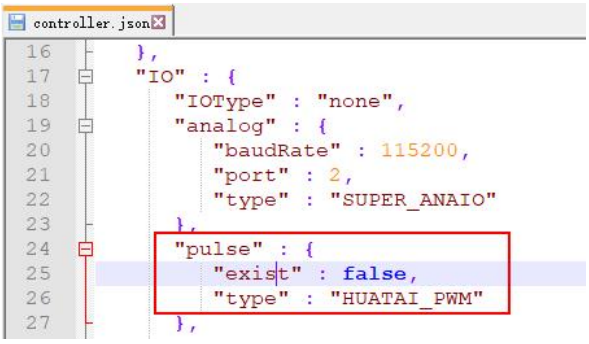
Frequency: Pulse output frequency; for example, the default value is 100, then 100 pulses are output in 1s

The IO boards that support this function are as follows:

HUATAI IOPWM

INEXBOT R1PWM

Usage:



```
    },
    "IO" : {
        "IOType" : "none",
        "analog" : {
            "baudRate" : 115200,
            "port" : 2,
            "type" : "SUPER_ANAIO"
        },
        "pulse" : {
            "exist" : false,
            "type" : "HUATAI_PWM"
        },
```

Modify profile controller.json;

Find the "exist" parameter in "IO" - "pulse" and change it to "turn";

turn: function on;

false: function off;

Find the "type" parameter in "IO" - "pulse" and change it to the corresponding IO board;

HUATAI_PWM: HUATAI IO

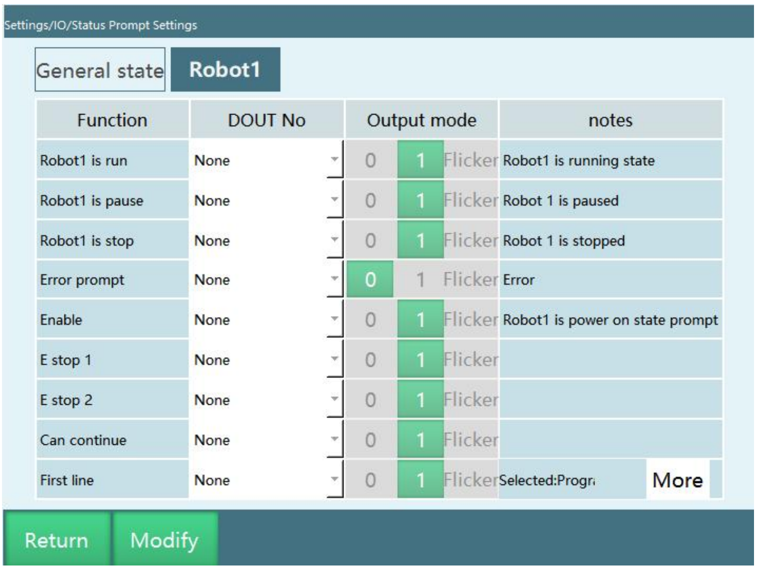
INEXBOT_PWM: INEXBOT R1

READ_DOUT-Read output

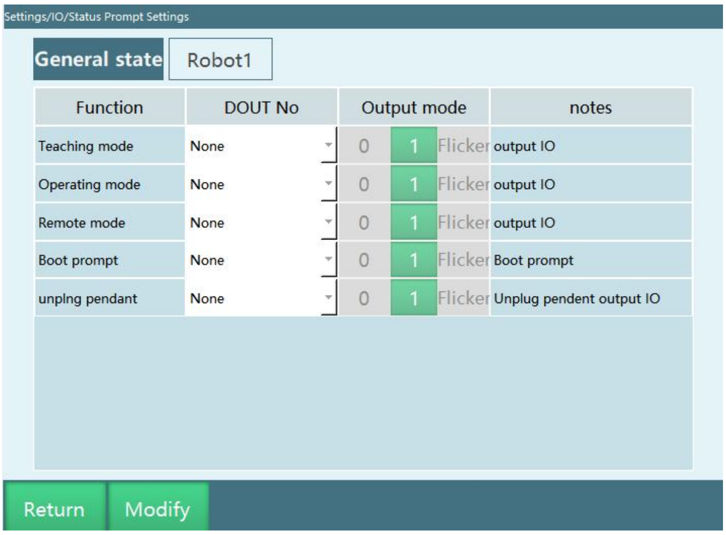
This instruction is used to read the output status of the current digital IO board into a variable. It is used in the same way as DIN, except that the reading is the status of the output.

> IO status prompt settings

In the "Status prompt settings" interface, you can set the I/O port corresponding to the "Boot prompt", robot running status, "Error prompt", "Enable", mode status, emergency stop, and other functions and the level corresponding to that port.



Function	DOUT No	Output mode	notes
Robot1 is run	None	0 [✓ 1] Flicker	Robot1 is running state
Robot1 is pause	None	0 [✓ 1] Flicker	Robot 1 is paused
Robot1 is stop	None	0 [✓ 1] Flicker	Robot 1 is stopped
Error prompt	None	[✓ 0] 1 Flicker	Error
Enable	None	0 [✓ 1] Flicker	Robot1 is power on state prompt
E stop 1	None	0 [✓ 1] Flicker	
E stop 2	None	0 [✓ 1] Flicker	
Can continue	None	0 [✓ 1] Flicker	
First line	None	0 [✓ 1] Flicker	Selected:Progr



Function	DOUT No	Output mode			notes
Teaching mode	None	0	1	Flicker	output IO
Operating mode	None	0	1	Flicker	output IO
Remote mode	None	0	1	Flicker	output IO
Boot prompt	None	0	1	Flicker	Boot prompt
unplng pendant	None	0	1	Flicker	Unplug pendent output IO

Robot1 run: The corresponding DOUT port outputs a high level when Robot1 is running

Robot1 pause: The corresponding DOUT port outputs a high level when Robot1 is paused

Robot1 stop: The corresponding DOUT port outputs a high level when Robot1 stops

Error prompt: When the robot servo reports an error, the corresponding DOUT port outputs the corresponding signal which can be set to "Bright" or "Flashing"

Enable: Output high level when the robot is powered on

E-stop 1: Output high or low level after "E-stop" signal is triggered, this can be set by yourself

E-stop 2: Output high or low level after "E-stop" signal is triggered, this can be set by yourself

Main program first line: Output a signal with a high level parameter of 1 and the program cursor jumps to the first line of the main program

Continuable: Output a signal with a high level parameter of 1, you can run a paused program

Boot prompt: Controller power-on output status, output high level when power on

Teach mode: Output high level when in teach mode

Run mode: Output high level when in run mode

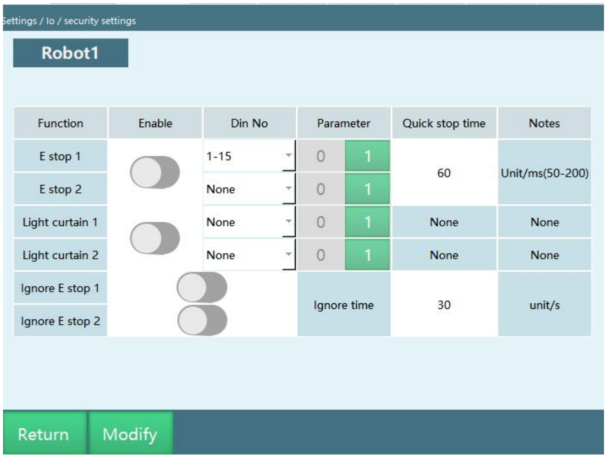
Remote mode: Output high level when in remote mode

Unplug teach pendant: Output high or low level after unplugging the teach pendant, this can be set by yourself

> IO safety settings

In the "Security settings" interface, you can set the I/O port corresponding to the emergency stop, safety light curtain and other functions and the level corresponding to that port.

After the IO E-stop has been lifted, you need to click the "Clear error" button to clear the error before you can perform other operations.



Settings / Io / security settings

Robot1

Function	Enable	Din No	Parameter	Quick stop time	Notes
E stop 1	○	1-15	0 1	60	Unit/ms(50-200)
E stop 2		None	0 1		
Light curtain 1	○	None	0 1	None	None
Light curtain 2		None	0 1	None	None
Ignore E stop 1	○		Ignore time	30	unit/s
Ignore E stop 2	○				

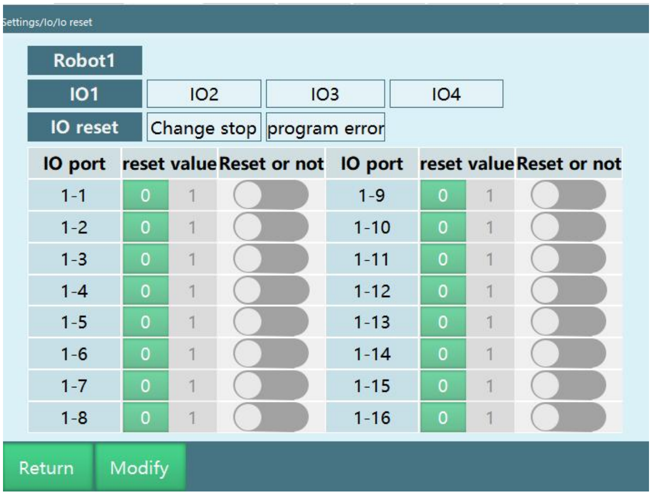
E-stop: The robot is powered off and switched to servo stop status after the emergency stop signal is triggered

Safety light curtain: The robot pauses after the safety light curtain is triggered, you can press the start button again to resume operation

Block E-stop: After turning on, the emergency stop signal is blocked during the blocking time

> IO reset

IO reset function can restore the output port of IO to the initial status when the program is stopped or error is reported. IO reset is divided into three types: remote IO reset, switch mode stop, and program error stop.



Settings/Io/Io reset

Robot1

IO1 | IO2 | IO3 | IO4

IO reset | Change stop | program error

IO port	reset value	Reset or not	IO port	reset value	Reset or not
1-1	0 1	○	1-9	0 1	○
1-2	0 1	○	1-10	0 1	○
1-3	0 1	○	1-11	0 1	○
1-4	0 1	○	1-12	0 1	○
1-5	0 1	○	1-13	0 1	○
1-6	0 1	○	1-14	0 1	○
1-7	0 1	○	1-15	0 1	○
1-8	0 1	○	1-16	0 1	○

Remote IO reset: In remote mode, when a reset signal is given, the robot will execute the reset program to return to the reset point, which will reset the IO port set in this interface to the reset value. If the reset program is stopped in the middle, the IO reset operation will not be performed.

Switch mode stop: When running a program, switching the mode to teach or remote mode will cause the program to stop, which will reset the IO port set in this interface to the reset value.

Program error stop: Program error causes the program to stop, which will reset the IO port set in this interface to the reset value. Specific types of errors: servo error, error of IO setting, error in system operation

Usage steps:

Enter the "IO reset" interface;

Select the robot;

Click to enter the reset scenario (IO reset, switch mode stop, program error stop);

Select the IO board;

Turn on the "Reset" switch corresponding to the IO port to be reset;

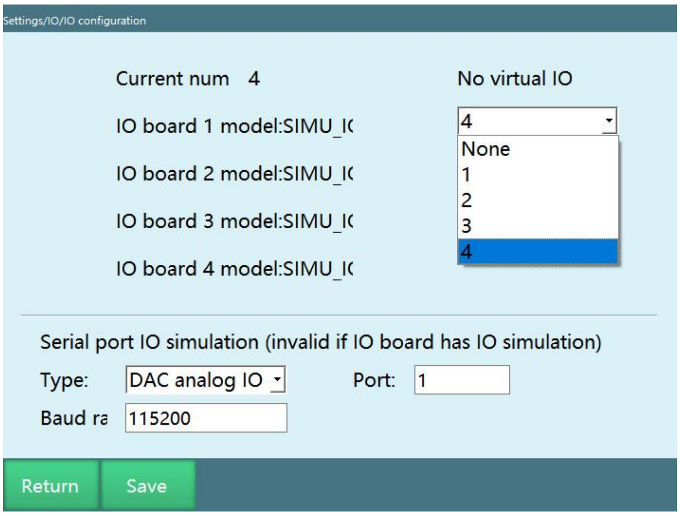
Select the reset value (0/1), 0 is low level and 1 is high level.

> IO configuration

The system will automatically identify the IO model according to the hardware connection order, no need to set; it can be used to view the number and model of IO boards.

Enter the [Settings] - [IO] - [IO configuration].

The input box is grayed out and no value can be entered.


Settings/IO/IO configuration

Current num 4

No virtual IO: 4

IO board 1 model:SIMU_I(

IO board 2 model:SIMU_I(

IO board 3 model:SIMU_I(

IO board 4 model:SIMU_I(

Serial port IO simulation (invalid if IO board has IO simulation)

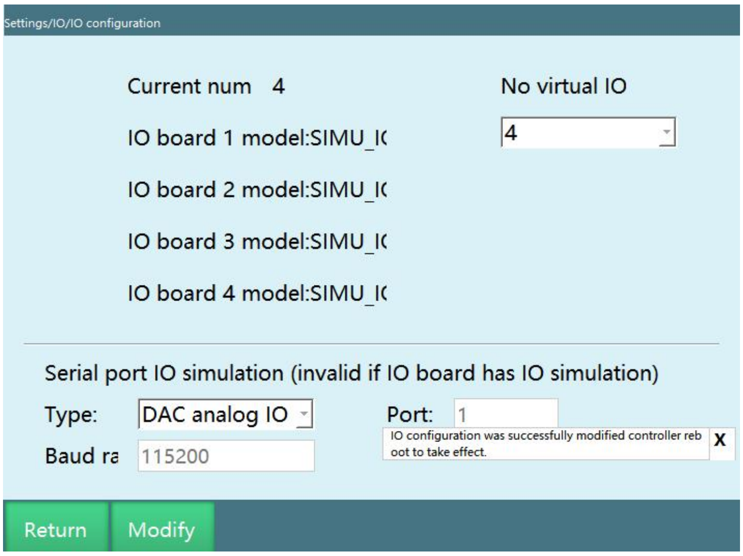
Type: DAC analog IO

Port: 1

Baud ra 115200

After clicking "Modify", the "Modify" button becomes "Save", then select the desired virtual IO from the "Number of virtual IO" drop-down box

Note: Virtual IO is only for program debugging and program demonstration, and does not have any IO signal access


Settings/IO/IO configuration

Current num 4

No virtual IO: 4

IO board 1 model:SIMU_I(

IO board 2 model:SIMU_I(

IO board 3 model:SIMU_I(

IO board 4 model:SIMU_I(

Serial port IO simulation (invalid if IO board has IO simulation)

Type: DAC analog IO

Port: 1

Baud ra 115200

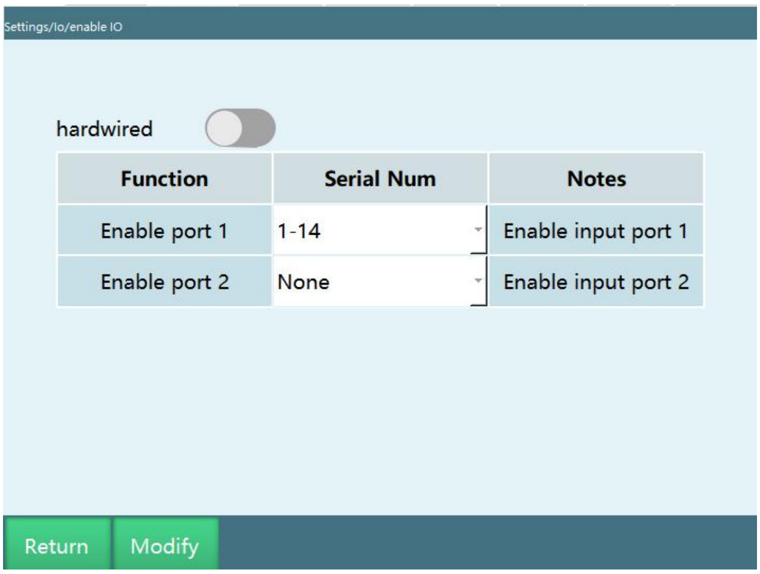
IO configuration was successfully modified controller reb oot to take effect.

Click "Save", restart to take effect, and the modification is successful.

> Enable IO

If you use the enable hardwired teach pendant, you need to select the corresponding DIN port and turn on the enable switch in this interface after connecting the cable. The power-on enable function is controlled by the IO board input signal; for the non-enable hardwired teach pendant, no need to set.

When this function is turned on, the teach pendant enable button is disabled and is not available for use.



Settings/Io/enable IO

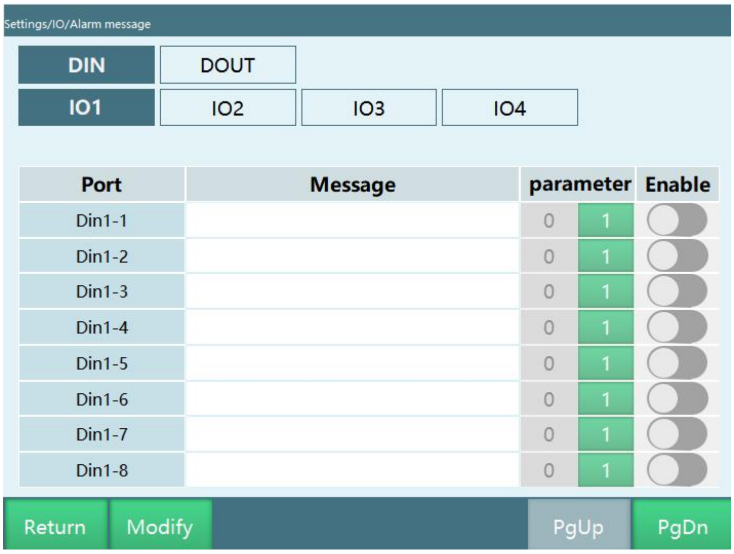
hardwired

Function	Serial Num	Notes
Enable port 1	1-14	Enable input port 1
Enable port 2	None	Enable input port 2

Enable port 1 is enabled for power-on, and enable port 2 is enabled for power-off. To power on, you only need to turn on the enable port 1. In any case, as long as the enable port 2 is turned on, it will be powered off.

> Alarm message

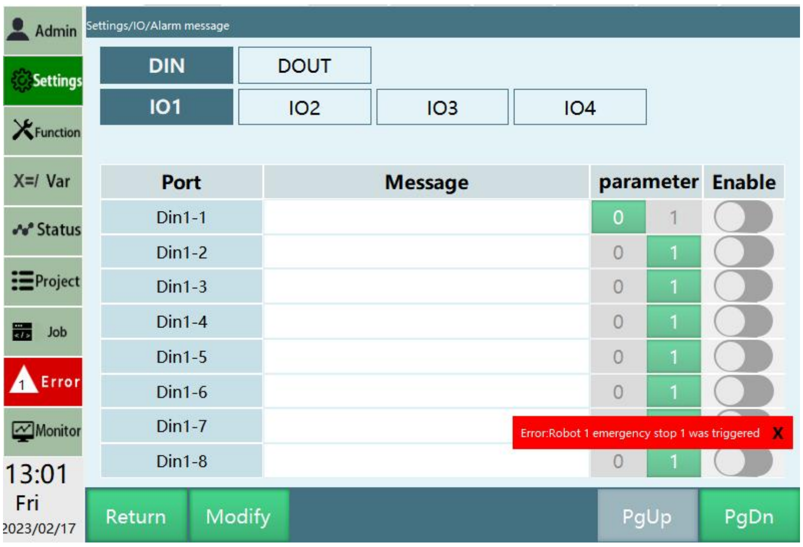
This function allows you to customize the alarm content of IO input and output ports, and the priority of alarm message is higher than that of other types of IO alarm messages.



Settings/IO/Alarm message

Port	Message	parameter		Enable
Din1-1		0	1	○
Din1-2		0	1	○
Din1-3		0	1	○
Din1-4		0	1	○
Din1-5		0	1	○
Din1-6		0	1	○
Din1-7		0	1	○
Din1-8		0	1	○

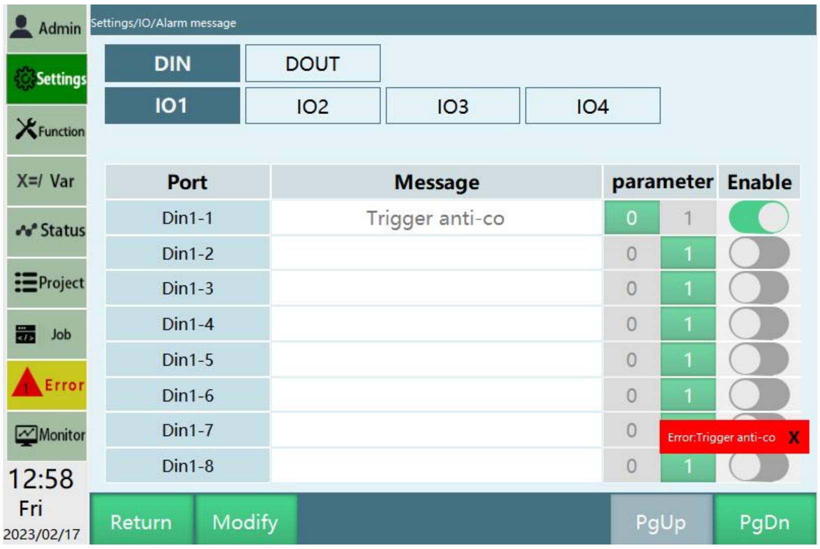
For example: set the IO emergency stop signal port to 15 to connect to the anti-collision IO, 1 to trigger and 0 to release; if DIN1 is triggered, it will report "Robot 1 IO emergency stop is triggered"; at this time, find DIN1 on the alarm message interface, and enter " Trigger anti-collision" in the message field, then triggering DIN15 again will report an error "Trigger anti-collision" instead of "Robot 1 IO emergency stop is triggered".



Settings/IO/Alarm message

Port	Message	parameter		Enable
Din1-1		0	1	○
Din1-2		0	1	○
Din1-3		0	1	○
Din1-4		0	1	○
Din1-5		0	1	○
Din1-6		0	1	○
Din1-7				
Din1-8		0	1	○

Error:Robot 1 emergency stop 1 was triggered X



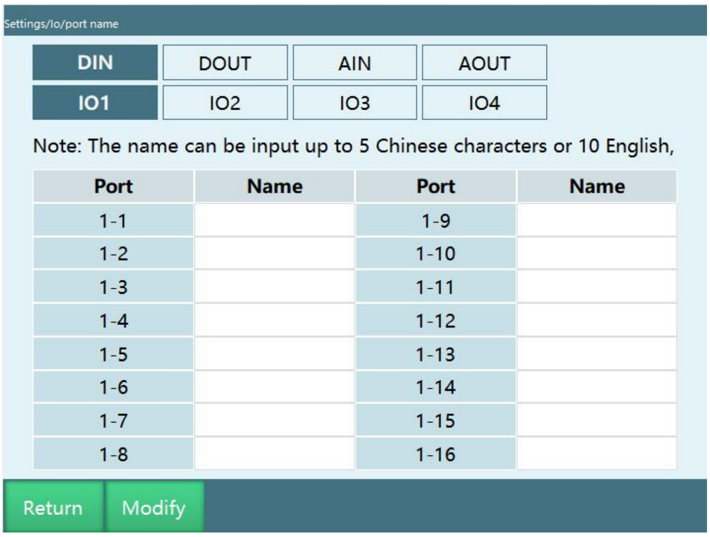
Settings/IO/Alarm message

Port	Message	parameter	Enable
Din1-1	Trigger anti-co	0 1	●
Din1-2		0 1	○
Din1-3		0 1	○
Din1-4		0 1	○
Din1-5		0 1	○
Din1-6		0 1	○
Din1-7		0	
Din1-8		0 1	○

Error:Trigger anti-co X

> Port name

The port name supports a maximum of 5 Chinese characters or 10 English characters. After the setting is successful, the name will be automatically displayed when using the IO port related drop-down box option.

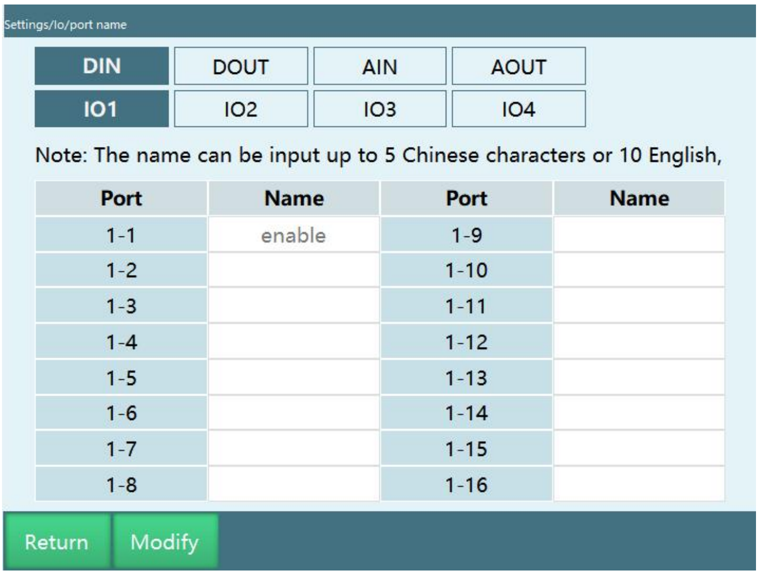


Settings/Io/port name

Note: The name can be input up to 5 Chinese characters or 10 English,

Port	Name	Port	Name
1-1		1-9	
1-2		1-10	
1-3		1-11	
1-4		1-12	
1-5		1-13	
1-6		1-14	
1-7		1-15	
1-8		1-16	

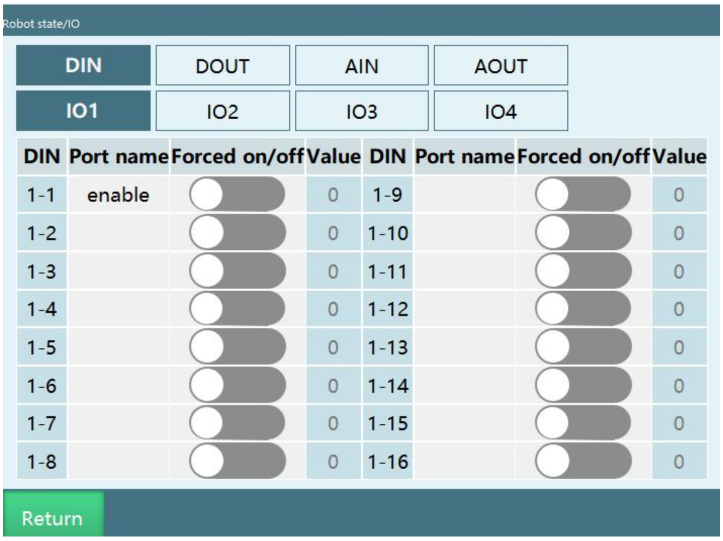
If the name of DIN1-1 is set as "enable ", the DIN1 name "enable " will be displayed in the [IO] of [Status]



Settings/Io/port name

Note: The name can be input up to 5 Chinese characters or 10 English,

Port	Name	Port	Name
1-1	enable	1-9	
1-2		1-10	
1-3		1-11	
1-4		1-12	
1-5		1-13	
1-6		1-14	
1-7		1-15	
1-8		1-16	



Robot state/IO

DIN	Port name	Forced on/off	Value	DIN	Port name	Forced on/off	Value
1-1	enable	○	0	1-9		○	0
1-2		○	0	1-10		○	0
1-3		○	0	1-11		○	0
1-4		○	0	1-12		○	0
1-5		○	0	1-13		○	0
1-6		○	0	1-14		○	0
1-7		○	0	1-15		○	0
1-8		○	0	1-16		○	0

Brief description of remote mode IO reservation

Signal description

	Function	Support mode	Trigger/output method	Description
Digital IO input	Start	remote	rising edge	When the parameter is 1, the signal is valid when it changes from 0 to 1
	Stop	remote	continuously valid	When the parameter is 1,

				the signal is continuously valid
	Pause	remote	continuously valid	When the parameter is 1, the signal is continuously valid
	Clear error	remote	rising edge	When the parameter is 1, the signal is valid when it changes from 0 to 1
	Start as reservation	remote	no	When it is turned on, the robot will be powered on when the reservation is successful
	I/O program 1-10	remote	pulse (period 0.6s)	When the parameter is 1, the signal is valid at 0-1-0, and the program needs to be triggered for at least 0.6 seconds for successful reservation.
	E-stop 1	teach, run, remote	high level	Scan once in 1 ms, trigger when scanned
	E-stop 2	teach, run, remote	high level	
	Safety light	run (running), remote	high level	

	curtain 1	(running)		
	Safety light curtain 2	run (running), remote (running)	high level	
	Block E-stop 1	used in conjunction with E-stop	The emergency stop function is blocked when the button is turned on, and the emergency stop signal is re-detected after the set time is up	
	Block E-stop 2	used in conjunction with E-stop		
Digital IO output	Boot prompt	no mode limit	bright, flashing, output only at power on	Output high level
	Robot1 run	teach, run, remote	bright, flashing	Output high level when the program is running
	Robot1 pause	teach, run, remote	bright, flashing	Output high level when the program is paused
	Robot1 stop	teach, run, remote	bright, flashing	Output high level when the program is stopped
	Error prompt	no mode limit	bright, flashing	If bright, then output high level; if flashing, then output pulse (period

				1s, 0.5s on, 0.5s off)
	Enable	no mode limit	bright, flashing	Output high level
	IO program 1-10 reservation output	remote	bright, flashing	Not bright when already reserved/not reserved; Flashing during reservation, period 1.2s, 0.6s on, 0.6s off; Bright when in operation, output high level
	E-stop 1	when signal is triggered	high level, low level, flashing	When the parameter is 1, output high level
	E-stop 2	when signal is triggered		
	Unplug teach pendant	no mode limit	high level, low level, flashing	Click "Unplug teach pendant", output 1 or 0
	Continuable	when signal is triggered	high level, low level, flashing	Output a signal with a high level parameter of 1, you can run the paused program
	Main program first line	teach, run, remote	high level, low level, flashing	Output a signal with a high level parameter of 1 and the

				program cursor jumps to the first line of the main program

Note: In this description, output 1 means output high level

Description of remote mode status

Not reserved: After entering remote mode, if no reservation has been made for the program, or the reservation was made and then cancelled, then "Not reserved" will be displayed.

Reserving: If the reservation is successful, "Reserving" will be displayed.

Running: If the program is running, "Running" will be displayed.

Reserved: If the program finishes running or is triggered to stop, "Reserved" will be displayed.

The speed cannot be modified in the remote mode, and the speed needs to be modified in advance in [Settings-Remote program setting]

Program reservation

Trigger the IO port corresponding to the program to successfully reserve the program. To cancel the reservation, you need to trigger the IO port corresponding to the program again.

Start: Directly trigger the IO port corresponding to the trigger

Start as reservation: signal 0-1 (press the button), 0.6 seconds or more time later, signal 1-0 (release the button), the program runs directly; when "Start as reservation" is selected, the start signal may not be set.

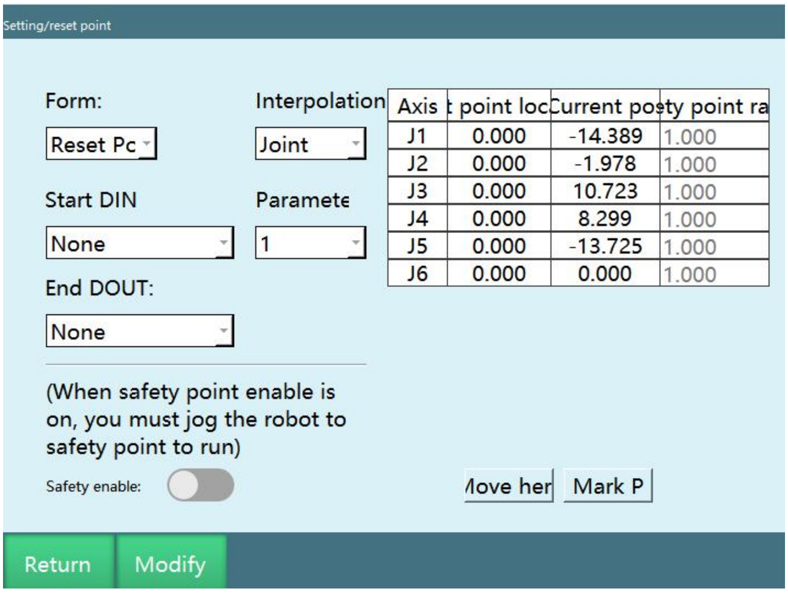
You can reserve again after the reserved program runs

Troubleshooting

After the IO function is successfully set, please go to "Status"-"IO function status" to check whether the setting is successful or whether there are conflicting functions.

Reset point setting

The reset point function supports movement to a safe point by means of joint and linear interpolation, and you can also use the reset program instruction to customize the reset trajectory and position.



Setting/reset point

Form: Reset Pc

Interpolation: Joint

Start DIN: None

Paramete: 1

End DOUT: None

(When safety point enable is on, you must jog the robot to safety point to run)

Safety enable: ○

Axis	t point loc	Current po	ty point ra
J1	0.000	-14.389	1.000
J2	0.000	-1.978	1.000
J3	0.000	10.723	1.000
J4	0.000	8.299	1.000
J5	0.000	-13.725	1.000
J6	0.000	0.000	1.000

Form: reset point, reset program;

Interpolation method: joint, linear; the movement speed is 10% of global speed when joint interpolation is selected, and 100mm/s when linear interpolation is selected; the running speed is equal to instruction speed x speed in status bar when reset program is selected.

Safety enable: When turned on, program will run to determine whether the robot is in the reset point (safety point) position, and it must be in the reset point position to continue running the program.

Start DIN: Reset point trigger signal;

Parameter: Reset point trigger signal 0 valid or 1 valid.

End DOUT: Status signal output after returning to the reset point;

Safety point range: The safe range error of each axis, if within the range, the robot will be judged to be at the reset point (safety point);

Mark this point: Set the current robot coordinate as the reset point, and click "OK" to set successfully;

Move here: Move to the reset point by joint interpolation;

Description of remote mode control right

When there are teach pendant, touch screen and I/O control device in the control system, the priority of the control right is teach pendant>touch screen>I/O control device.

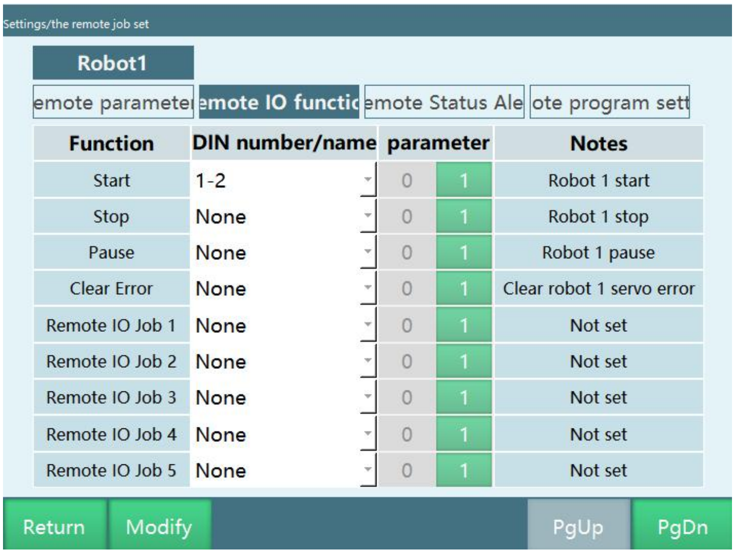
After switching to the remote mode, the control right is switched to the touch screen. If there is no touch screen, switch to I/O control device. At this time, the interface of the teach pendant only displays the connection status of the Modbus module and the I/O module and the I/O program.

When there are touch screen and I/O module at the same time, set the I/O module enable in the touch screen.

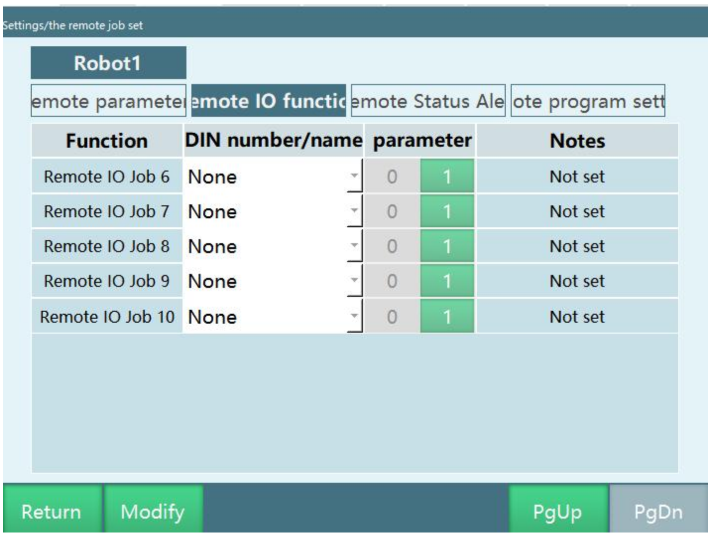
Remote IO control

Remote I/O function selection setting

In "Remote program setting-Remote IO function", you can set the I/O port corresponding to the remote IO control (start, stop, pause, emergency stop, clear alarm, etc.) and the level corresponding to that port, and you can also set the program run by the I/O module remote control.



Function	DIN number/name	parameter		Notes
Start	1-2	0	1	Robot 1 start
Stop	None	0	1	Robot 1 stop
Pause	None	0	1	Robot 1 pause
Clear Error	None	0	1	Clear robot 1 servo error
Remote IO Job 1	None	0	1	Not set
Remote IO Job 2	None	0	1	Not set
Remote IO Job 3	None	0	1	Not set
Remote IO Job 4	None	0	1	Not set
Remote IO Job 5	None	0	1	Not set



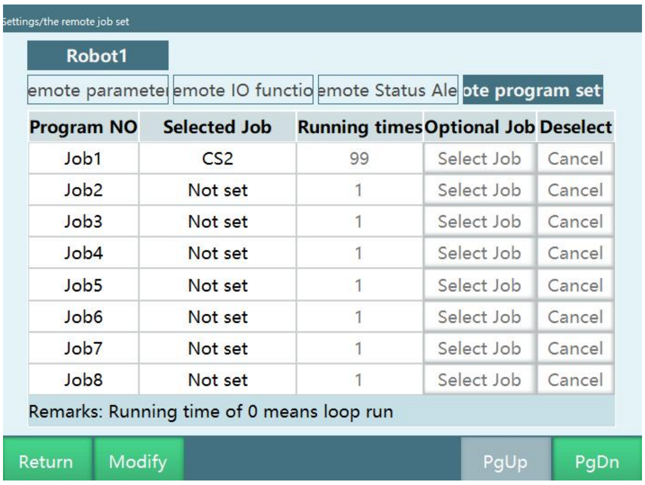
Function	DIN number/name	parameter		Notes
Remote IO Job 6	None	0	1	Not set
Remote IO Job 7	None	0	1	Not set
Remote IO Job 8	None	0	1	Not set
Remote IO Job 9	None	0	1	Not set
Remote IO Job 10	None	0	1	Not set

The program of the set I/O module can only be selected from the program set in the "Remote program setting" interface.

There can be up to 10 remote reservation programs

Start as reservation: After it is turned on, the robot will be powered on and run the first reserved program immediately after the reservation is successful, and other programs can be reserved at this time.

Remote program setting



Program NO	Selected Job	Running times	Optional Job	Deselect
Job1	CS2	99	Select Job	Cancel
Job2	Not set	1	Select Job	Cancel
Job3	Not set	1	Select Job	Cancel
Job4	Not set	1	Select Job	Cancel
Job5	Not set	1	Select Job	Cancel
Job6	Not set	1	Select Job	Cancel
Job7	Not set	1	Select Job	Cancel
Job8	Not set	1	Select Job	Cancel

Remarks: Running time of 0 means loop run

The programs used by the touch screen and the I/O control module can be set in the "Remote program setting" interface.

If there are multiple robots, you can select the robot to be set at the robot section, and set each program of the robot.

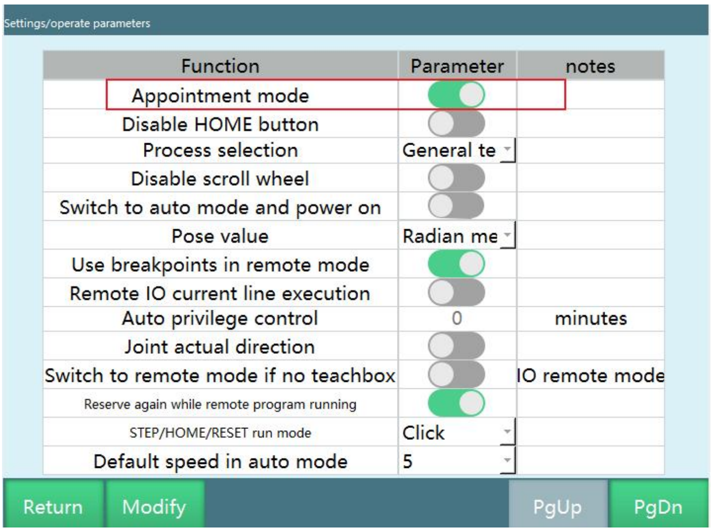
The program used by the I/O control module needs to be set in the I/O function interface.

The selected program on the remote program interface can be canceled by clicking the "Cancel" button.

Just fill in the corresponding number for the running times, 0 means cycle running.

Reservation mode

In "Settings/Operation parameters":



Function	Parameter	notes
Appointment mode	☑	
Disable HOME button	☐	
Process selection	General te	
Disable scroll wheel	☐	
Switch to auto mode and power on	☐	
Pose value	Radian me	
Use breakpoints in remote mode	☑	
Remote IO current line execution	☐	
Auto privilege control	0	minutes
Joint actual direction	☐	
Switch to remote mode if no teachbox	☐	IO remote mode
Reserve again while remote program running	☑	
STEP/HOME/RESET run mode	Click	
Default speed in auto mode	5	

After the "Reservation mode" is enabled, if the remote IO program signal is triggered, the program reservation is successful, and if the start signal is triggered, the robot runs;

After the "Reservation mode" is disabled, if the remote IO program signal is triggered, the robot runs directly. At this time, triggering other remote IO program signals is invalid. After the robot runs, the remote IO program signal can be triggered again. There is no need to set a start signal.

Use of remote function (IO)

Remote function overview

Set 10 remote programs and the running times of each program, queue the 10 programs before running, run them according to the order in the queue and the running times, and stop and wait to queue again after the queue is finished.

Steps to use remote function

Write program——Set remote program——Set IO——Switch to remote mode——Reserve and sort——Run

1.Write program

Create a new program and insert the instruction, please make sure the program can run properly.

2.Set remote program

Enter the "Settings-Remote program setting" interface, set the program name and running times for program 1 to program 10, if you want the single program to run in an infinite loop, then set the running times for the program to 0. The program name here refers to the program in the "Project" interface, and the remote program will be modified automatically after the instruction in the program is modified, so there is no need to reset the remote program.

If the program name of the program is modified, please reset the program in the "Remote program setting" interface.

3.Set IO

In the "IO-IO function" interface, set the corresponding IO port and effective value of each function. When the effective value is 1, the high level is valid, and when the effective value is 0, the low level is valid.

The function of the IO port corresponding to program 1-program 10 is not to select the program to run, but to queue the program in the remote mode.

4.Switch to remote mode

Turn the mode selection key to the remote mode position or click the mode status in the program to select the remote mode.

When the teach pendant is not connected to the controller, the controller will automatically enter the remote mode when it is started.

When the controller is connected to IO, Modbus device, and teach pendant at the same time, the priority of the three devices is teach pendant>Modbus device>IO device. After switching to the remote mode, the Modbus device is valid and the IO device is invalid. At this time, if you turn off the enable button in the Modbus device, the IO device will be valid.

5.Reserve and sort

For example: The IO function in the IO function is set to

run port 1 valid value 1

stop port 2 valid value 1

pause port 3 valid value 1

clear alarm port 4 valid value 1

program 1	port 5	valid value 1
program 2	port 6	valid value 1
program 3	port 7	valid value 1
program 4	port 8	valid value 1
program 5	port 9	valid value 1
program 6	port 10	valid value 1
program 7	port 11	valid value 1
program 8	port 12	valid value 1
program 9	port 13	valid value 1
program 10	port 14	valid value 1

Then the sorting is such that if you give port 6 a high level for 1 second and then release it, program 2 will be the first in line, if you give port 8 a high level for 1 second and then release it, program 4 will be the second in line, and so on. If you want to dequeue a program in the queue, give the corresponding IO port a high level for 1 second, the program will be dequeued in the queue.

There can only be 10 programs in the queue, and the same program cannot be queued repeatedly.

When a program is running, it can be re-added to the end of the queue.

6.Run

Give a high level to the port with running function, and the robot will start to run according to the order in the queue and running times. After the operation is completed, the servo will not be powered off. At this time, add the program to the queue, and the robot will run the program immediately.

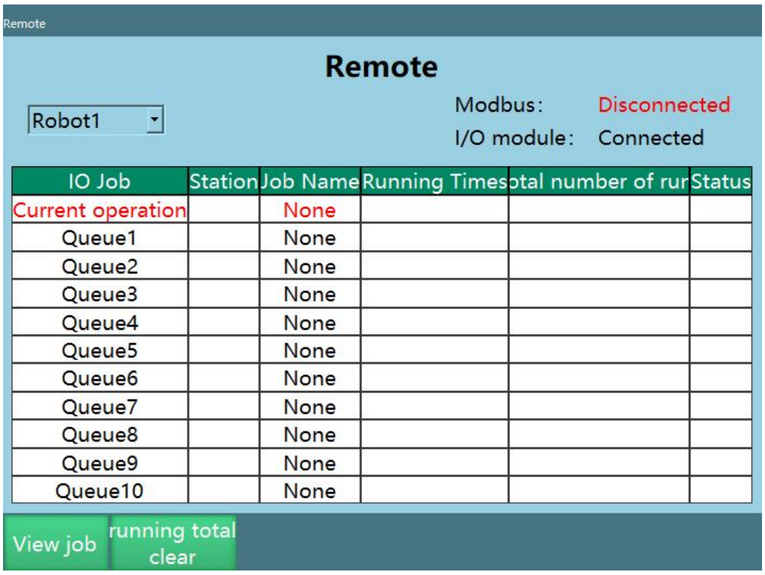
When there is no program in the queue, if you let the robot run, it will power on but not operate. At this time, if you place the program in the queue, the robot will execute the program immediately.

View operation

To view the details of program operation through remote IO control, click the "View program" button in the remote mode interface, modbus can also be viewed through this function.

Total running clear

Clear the total running times of the currently running program, only the total running times can be cleared, but the running times cannot be cleared.



Remote

Remote

Robot1

Modbus: Disconnected
I/O module: Connected

IO Job	Station	Job Name	Running Times	otal number of rur	Status
Current operation		None			
Queue1		None			
Queue2		None			
Queue3		None			
Queue4		None			
Queue5		None			
Queue6		None			
Queue7		None			
Queue8		None			
Queue9		None			
Queue10		None			

MODBUS

> Modbus instructions

Open MODBUS connection

This instruction is used to open the modbus communication connection in run mode, the process number bound is the modbus master process number

Disconnect modbus connection

This instruction is used to disconnect the modbus communication connection in run mode, the process number bound is the modbus master process number

Get modbus connection status

This instruction stores the connection status of modbus in the bool variable, and determines the connection status of modbus by getting the value of the variable. This instruction gets status every time it is run, and the status is often placed under "Open modbus connection".

Modbus read

This instruction is used to read the address code of the corresponding location in modbus, and the address types that can be set are 3x, 4x-bit, 3x-bit, 0x.

Slave register first address: the first address to read

Number of reads: the total number of addresses to be read

First variable type: get the variable where the data is stored

First variable name: get the first variable name where the data is stored

Modbus write

This instruction is used to write variables into the address code of the corresponding location in the slave register via modbus. The address types that can be set are 4x, 4x-bit, and 0x, and 3x and 3x-bit are missing compared to Modbus read.

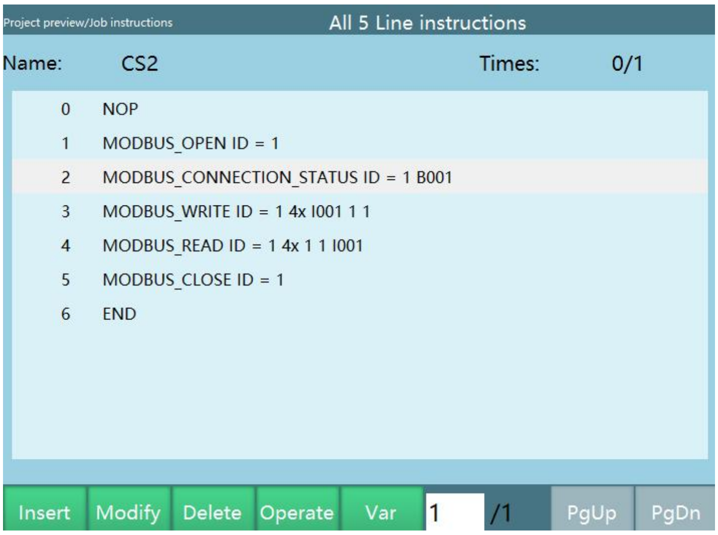
First variable type: the type of the first variable written

First variable name: the first written variable name

Slave register first address: the first address to write

Number of writes: the total number of addresses to be written

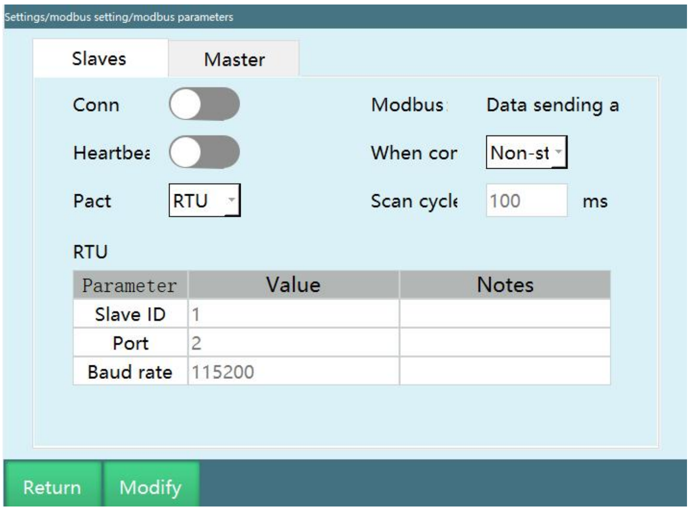
The specific usage is as follows:



Name: CS2 Times: 0/1

```
0  NOP
1  MODBUS_OPEN ID = 1
2  MODBUS_CONNECTION_STATUS ID = 1 B001
3  MODBUS_WRITE ID = 1 4x I001 1 1
4  MODBUS_READ ID = 1 4x 1 1 I001
5  MODBUS_CLOSE ID = 1
6  END
```

> Modbus slave



Conn

Heartbea

Pact: RTU

Modbus: Data sending a

When cor: Non-st

Scan cycle: 100 ms

RTU

Parameter	Value	Notes
Slave ID	1	
Port	2	
Baud rate	115200	

When acting as slave, you can set heartbeat detection to confirm the communication status with the master station, and whether to stop when the communication is disconnected. You can choose RTU or TCP protocol, and its corresponding port settings.

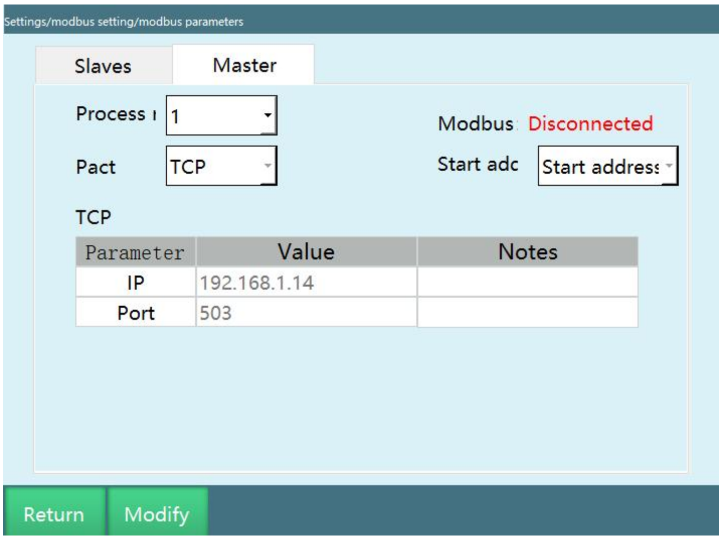
Heartbeat detection: When turned on, the system will determine the status of communication with the modbus, and when turned off, it will not monitor the communication connection with the modbus, which is generally used for RTU protocol.

When communication is disconnected: If "Stop" is selected, the robot is powered off when modbus communication is disconnected. Conversely, if "Non-stop" is selected, the robot does not power off when the modbus communication is disconnected.

Protocol: RTU or TCP.

Scan period: Refers to how often the system scans for data in range in the modbus

> Modbus master



Settings/modbus setting/modbus parameters

Process ı 1

Pact TCP

Modbus Disconnected

Start adc Start address

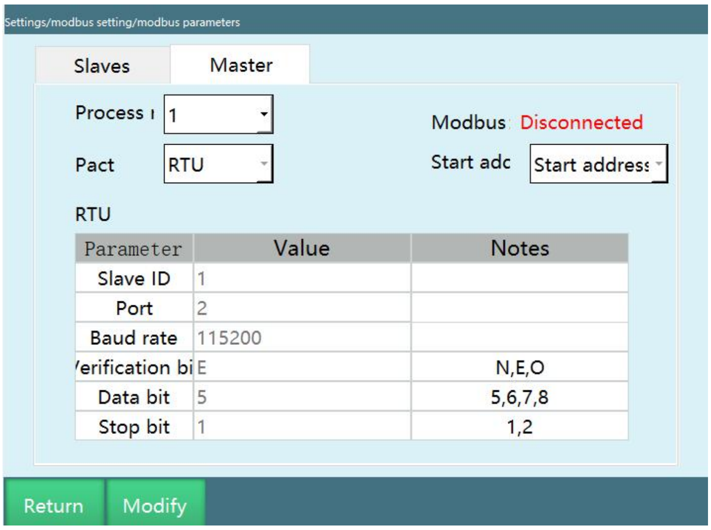
TCP

Parameter	Value	Notes
IP	192.168.1.14	
Port	503	

When acting as master, only the communication method and its corresponding port settings can be selected.

"Start address" can be set to "Start address is 0" or "Start address is 1"

When the master protocol is set to RTU, the "Check bit", "Data bit", and "Stop bit" need to be set.



Settings/modbus setting/modbus parameters

Process ı 1

Pact RTU

Modbus Disconnected

Start adc Start address

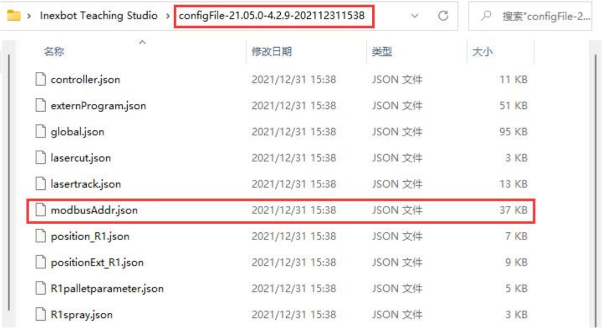
RTU

Parameter	Value	Notes
Slave ID	1	
Port	2	
Baud rate	115200	
'erification bi	E	N,E,O
Data bit	5	5,6,7,8
Stop bit	1	1,2

> Modbus address code modification

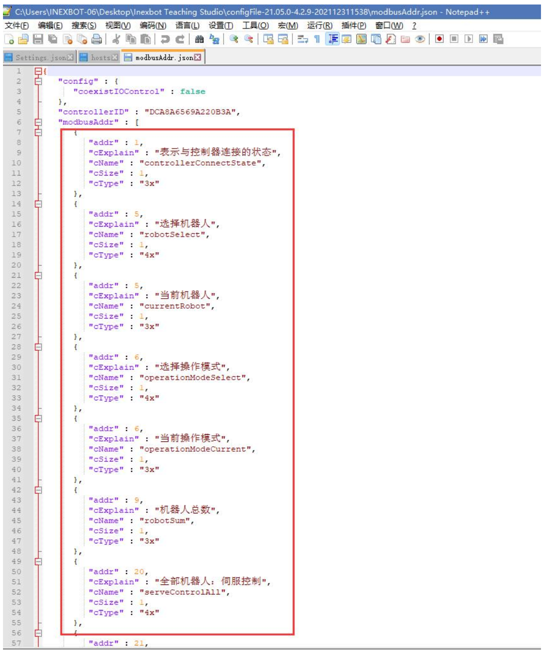
1.Insert the U disk, export the controller configuration

2.Find the configuration file modbusAddr.json in the configFile+date folder


Inexbot Teaching Studio › configFile-21.05.0-4.2.9-202112311538
搜索"configFile-2...

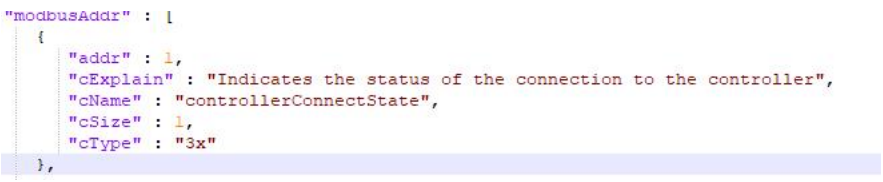
名称	修改日期	类型	大小
controller.json	2021/12/31 15:38	JSON 文件	11 KB
externProgram.json	2021/12/31 15:38	JSON 文件	51 KB
global.json	2021/12/31 15:38	JSON 文件	95 KB
lasercut.json	2021/12/31 15:38	JSON 文件	3 KB
lasertrack.json	2021/12/31 15:38	JSON 文件	13 KB
modbusAddr.json	2021/12/31 15:38	JSON 文件	37 KB
position_R1.json	2021/12/31 15:38	JSON 文件	7 KB
positionExt_R1.json	2021/12/31 15:38	JSON 文件	9 KB
R1palletparameter.json	2021/12/31 15:38	JSON 文件	5 KB
R1spray.json	2021/12/31 15:38	JSON 文件	3 KB

3.Open it with a text editor such as Notepad++


C:\Users\INEXBOT-06\Desktop\Inexbot Teaching Studio\configFile-21.05.0-4.2.9-202112311538\modbusAddr.json - Notepad++
文件(F) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言(L) 设置(T) 工具(O) 宏(M) 运行(R) 插件(P) 窗口(W) ?
Settings.json
hosts
modbusAddr.json

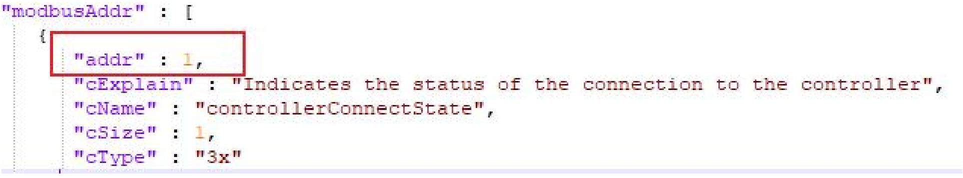
```
{
    "config" : {
        "coexistIOControl" : false
    },
    "controllerID" : "DCA8A6569A220B3A",
    "modbusAddr" : [
        {
            "addr" : 1,
            "cExplain" : "表示与控制器连接的状态",
            "cName" : "controllerConnectState",
            "cSize" : 1,
            "cType" : "3x"
        },
        {
            "addr" : 5,
            "cExplain" : "选择机器人",
            "cName" : "robotSelect",
            "cSize" : 1,
            "cType" : "4x"
        },
        {
            "addr" : 5,
            "cExplain" : "当前机器人",
            "cName" : "currentRobot",
            "cSize" : 1,
            "cType" : "3x"
        },
        {
            "addr" : 6,
            "cExplain" : "选择操作模式",
            "cName" : "operationModeSelect",
            "cSize" : 1,
            "cType" : "4x"
        },
        {
            "addr" : 6,
            "cExplain" : "当前操作模式",
            "cName" : "operationModeCurrent",
            "cSize" : 1,
            "cType" : "3x"
        },
        {
            "addr" : 9,
            "cExplain" : "机器人总数",
            "cName" : "robotSum",
            "cSize" : 1,
            "cType" : "3x"
        },
        {
            "addr" : 20,
            "cExplain" : "全部机器人：伺服控制",
            "cName" : "serveControlAll",
            "cSize" : 1,
            "cType" : "4x"
        },
        {
            "addr" : 21,
```

4.When opened, you can see a {...} containing a set of address code parameters (the system will automatically generate a copy of the original address code)



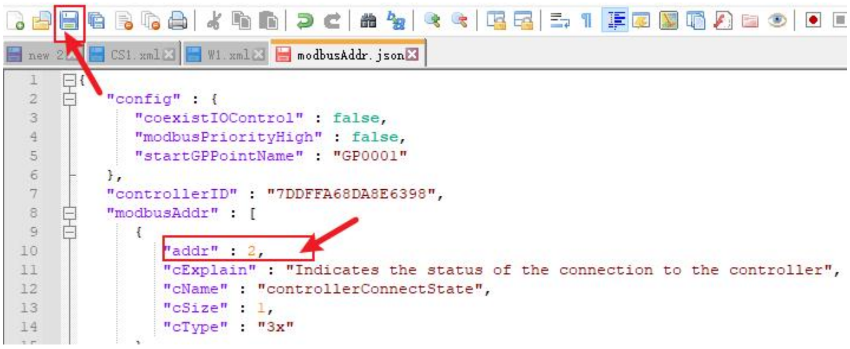
```
"modbusAddr" : [
    {
        "addr" : 1,
        "cExplain" : "Indicates the status of the connection to the controller",
        "cName" : "controllerConnectState",
        "cSize" : 1,
        "cType" : "3x"
    },
```

5.To modify the address, you only need to directly change the number after addr. When the number is 0, the function of the address code is invalid



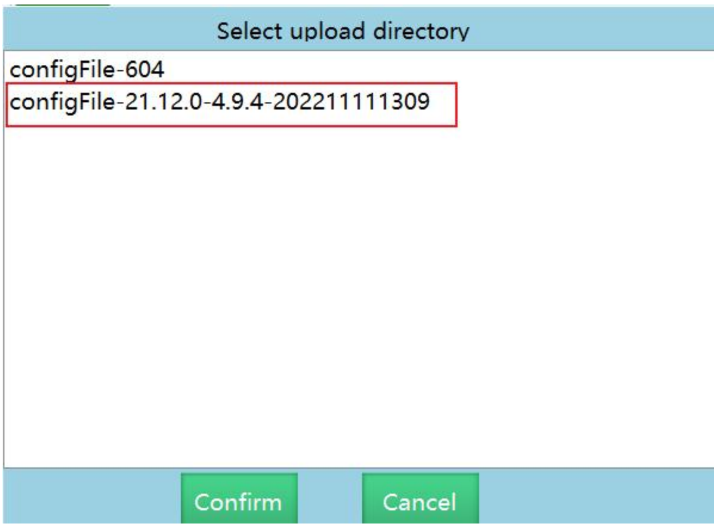
```
"modbusAddr" : [
    {
        "addr" : 1,
        "cExplain" : "Indicates the status of the connection to the controller",
        "cName" : "controllerConnectState",
        "cSize" : 1,
        "cType" : "3x"
```

6.Click "Save" after modification



```
{
    "config" : {
        "coexistIOControl" : false,
        "modbusPriorityHigh" : false,
        "startGPPointName" : "GP0001"
    },
    "controllerID" : "7DDFFA68DA8E6398",
    "modbusAddr" : [
        {
            "addr" : 2,
            "cExplain" : "Indicates the status of the connection to the controller",
            "cName" : "controllerConnectState",
            "cSize" : 1,
            "cType" : "3x"
```

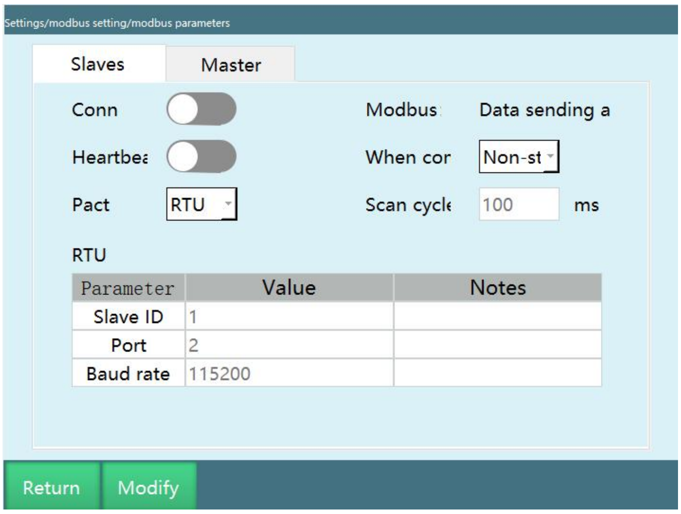
7.Then re-import the parameter into the controller, restart to take effect



Select upload directory

configFile-604
configFile-21.12.0-4.9.4-202211111309

8.After modifying the parameters, restart or reopen the connection to take effect (The system will automatically restart if the configuration file is imported)



Settings/modbus setting/modbus parameters

Conn
Heartbea
Pact RTU
Modbus Data sending a
When cor Non-st
Scan cycle 100 ms

RTU

Parameter	Value	Notes
Slave ID	1	
Port	2	
Baud rate	115200	

> Use of Modbus

Function overview

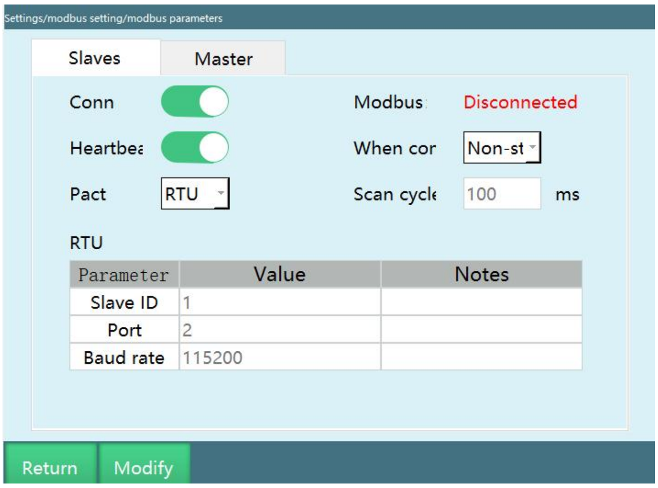
The Modbus function can replace some functions of the teach pendant, remotely control robot operation, teach, view status, etc.

Modbus supports modbusTCP and modbusRTU protocols.

Modbus has two modes: teach and run. For address codes, see "MODBUS Address Code List.xls" for details.

Modbus poll connection method

1.Enter the "Settings/modbus settings/modbus parameters" interface and turn on the connection switch



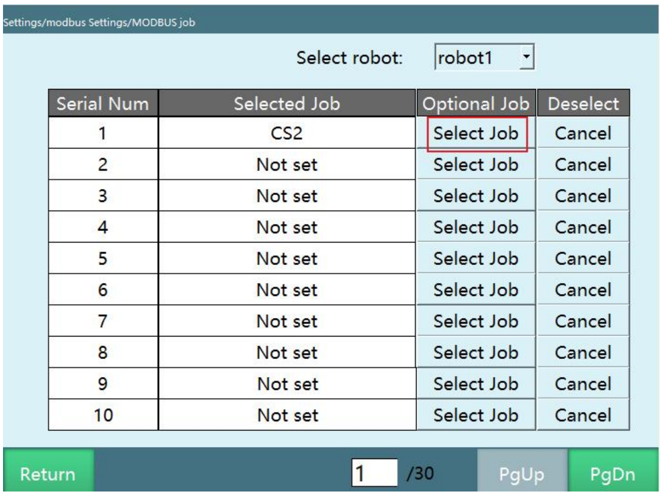
Settings/modbus setting/modbus parameters

Conn
Heartbea
Pact RTU
Modbus Disconnected
When cor Non-st
Scan cycle 100 ms

RTU

Parameter	Value	Notes
Slave ID	1	
Port	2	
Baud rate	115200	

2.In the "Settings/modbus settings/modbus program" interface, select the program



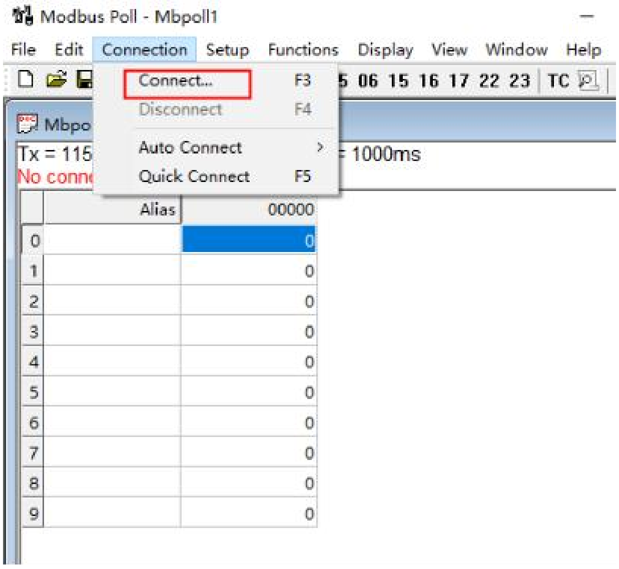
Settings/modbus Settings/MODBUS job

Select robot: robot1

Serial Num	Selected Job	Optional Job	Deselect
1	CS2	Select Job	Cancel
2	Not set	Select Job	Cancel
3	Not set	Select Job	Cancel
4	Not set	Select Job	Cancel
5	Not set	Select Job	Cancel
6	Not set	Select Job	Cancel
7	Not set	Select Job	Cancel
8	Not set	Select Job	Cancel
9	Not set	Select Job	Cancel
10	Not set	Select Job	Cancel

3.Open the ModbusPoll software

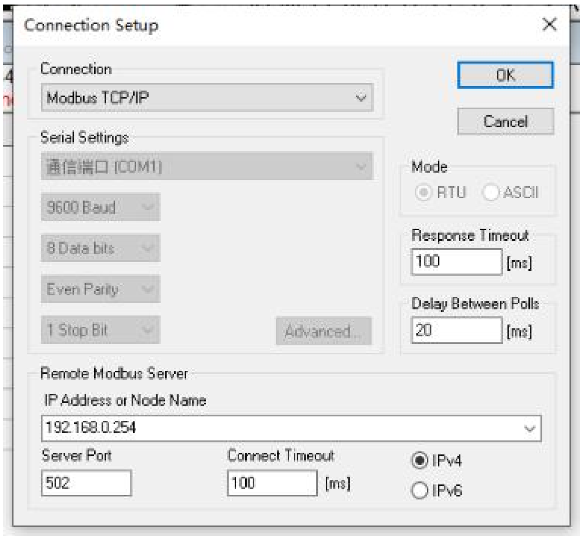
4.After opening the software, we need to connect and set (Connection——Concection Setup), set the required parameters (the parameters in the picture are just examples), click the "OK" button, and the teach pendant page will display connected, if connected and unconnected blinking screen is displayed, you need to change the Scan Rate parameter from 1000ms to 100ms in Setup——Read/Write Definition



Modbus Poll - Mbpoll1

Tx = 115
No conn
= 1000ms

	Alias	00000
0		0
1		0
2		0
3		0
4		0
5		0
6		0
7		0
8		0
9		0



Connection: Modbus TCP/IP

Serial Settings: 通信端口 (COM1); 9600 Baud; 8 Data bits; Even Parity; 1 Stop Bit

Mode: ● RTU ○ ASCII

Response Timeout: 100 [ms]

Delay Between Polls: 20 [ms]

Remote Modbus Server

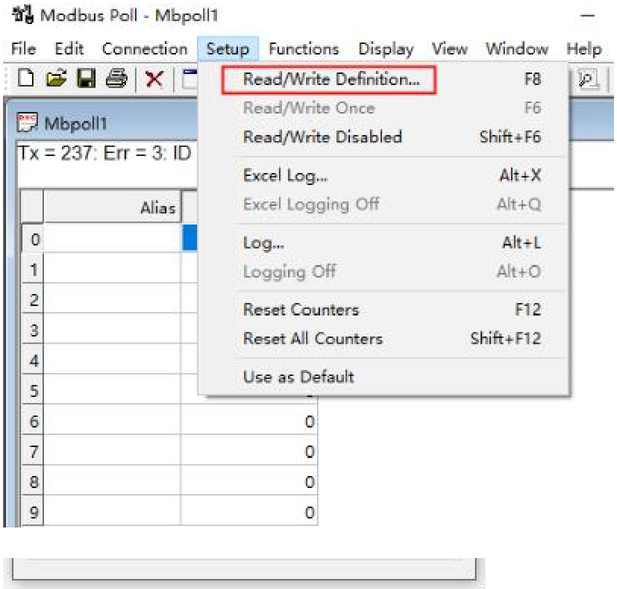
IP Address or Node Name: 192.168.0.254

Server Port: 502

Connect Timeout: 100 [ms]

● IPv4 ○ IPv6

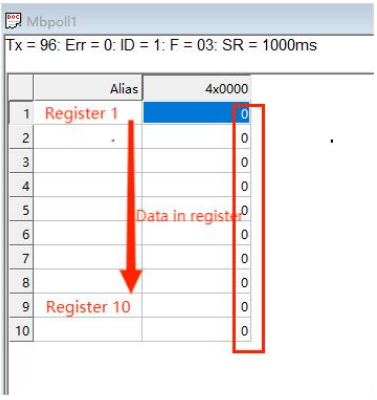
5.Set the parameters under Setup-----Read/Write Definition (the parameters in the picture are only examples). If the address code does not take effect, you can troubleshoot: change the start address to 1, check the "PLC Addresses" option, and click "OK"



Tx = 237: Err = 3: ID

	Alias	
0		
1		
2		
3		
4		
5		
6		0
7		0
8		0
9		0

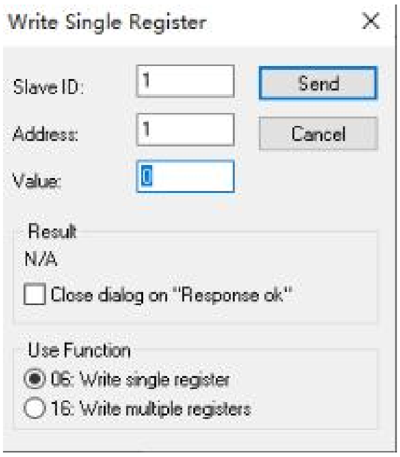
6.Double-click the register data and fill in the relevant address code



Mbpoll1

Tx = 96: Err = 0: ID = 1: F = 03: SR = 1000ms

	Alias	4x0000
1	Register 1	0
2		0
3		0
4		0
5		0
6		0
7		0
8		0
9	Register 10	0
10		0



Write Single Register

Slave ID: 1

Address: 1

Value: 0

Result

N/A

☐ Close dialog on "Response ok"

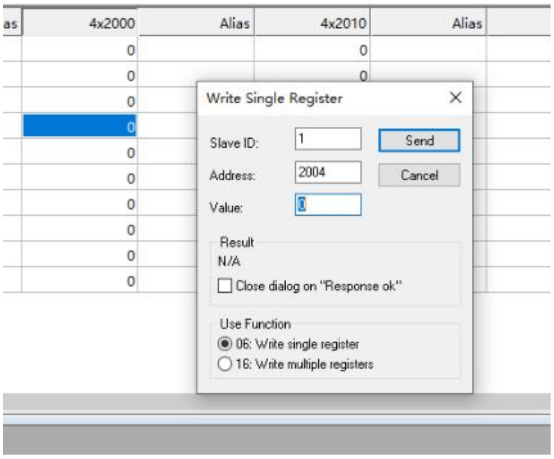
Use Function

● 06: Write single register

○ 16: Write multiple registers

Modbus readable global positions in any mode

1.Select the global type to be read according to Modbus address 2004: 0 means GP point, 1 means GE point.



as	4x2000	Alias	4x2010	Alias
	0		0	
	0		0	
	0			
	0			
	0			
	0			
	0			
	0			
	0			
	0			

Write Single Register

Slave ID: 1

Address: 2004

Value: 0

Result

N/A

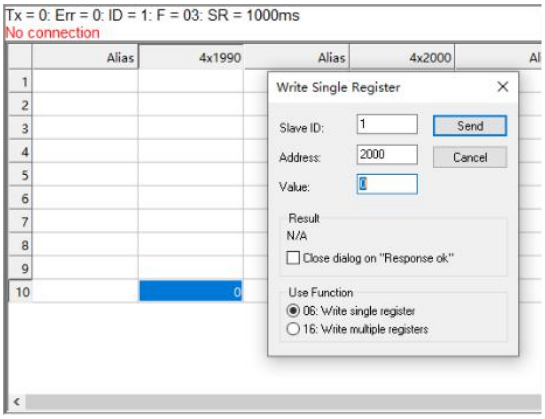
☐ Close dialog on "Response ok"

Use Function

● 06: Write single register

○ 16: Write multiple registers

2.Select the number of the global point to be read according to Modbus address 2000, numbered from 1 to 999.



Tx = 0: Err = 0: ID = 1: F = 03: SR = 1000ms
No connection

	Alias	4x1990	Alias	4x2000	Al
1					
2					
3					
4					
5					
6					
7					
8					
9					
10		0			

Write Single Register

Slave ID: 1

Address: 2000

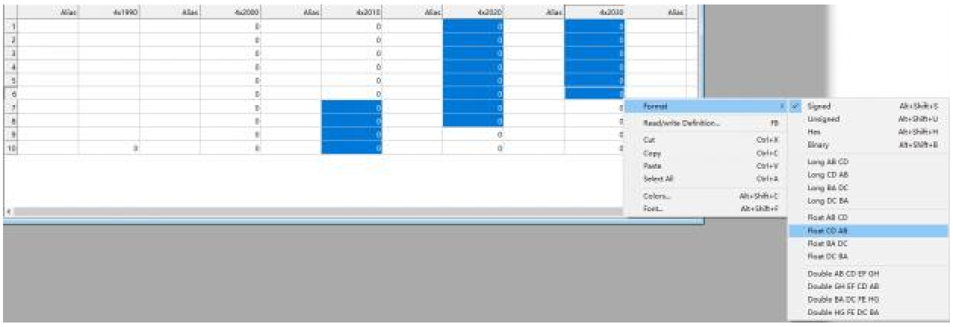
Value: 0

Result
N/A

□ Close dialog on "Response ok"

Use Function
● 06: Write single register
○ 16: Write multiple registers

3.Change the Format of 2017-2028 and 2031-2036 to FloatCD AB



	Alias	4x1990	Alias	4x2000	Alias	4x2010	Alias	4x2020	Alias	4x2030	Alias
1				0		0		0		0	
2				0		0		0		0	
3				0		0		0		0	
4				0		0		0		0	
5				0		0		0		0	
6				0		0		0		0	
7				0		0		0		0	
8				0		0		0		0	
9				0		0		0		0	
10		0		0		0		0		0	

4.The Modbus address code starts from 2017 and includes 2017, and every 2 address codes represent the value of one axis of the robot, that is, 2017 and 2018 represent axis 1.

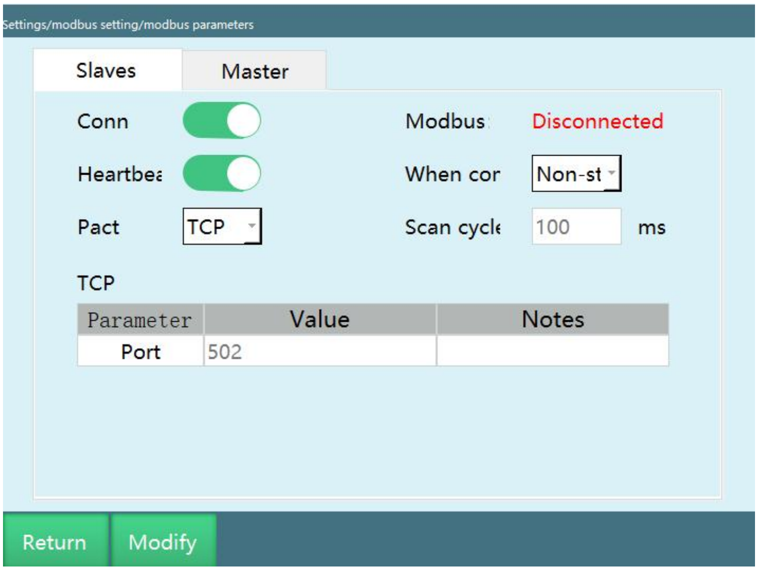
5.Address codes 2031-2036 represent external axes, and only support 3 external axes.

6.After the Modbus is successfully connected, select the global point number through the address code 2000, and select the global point type through the address code 2004, then you can view the global point in address code 2017.

Modbus multi-master connection

1.Connect the computer and one or more touchpads to the controller through the switch.

2.The controller acts as a slave station, and the modbus poll and touchpad act as the master station. You can open multiple modbus polls to act as multiple master stations. Currently, the controller supports up to 9 master stations to be connected at the same time.



Conn: on

Heartbea: on

Pact: TCP

Modbus: Disconnected

When cor: Non-st

Scan cycle: 100 ms

TCP

Parameter	Value	Notes
Port	502	

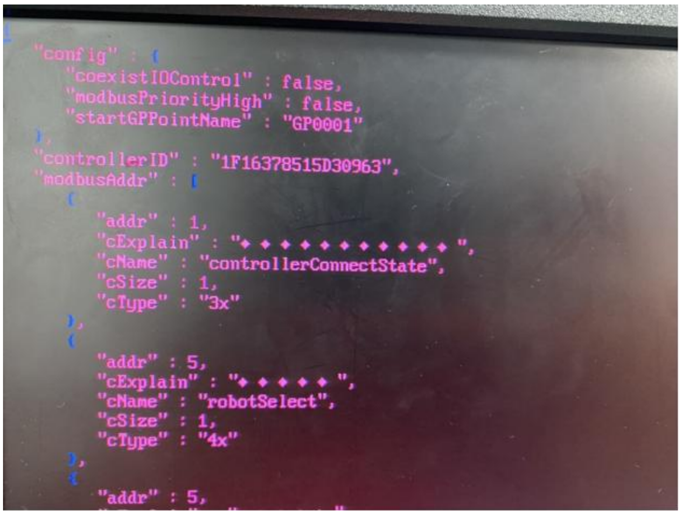
3.In the modbus poll, click "Connection" and select "Connect", select TCP as the connection type, keep the same IP address and port number as the teach pendant, and keep the same scan cycle as the teach pendant.

4.Modbus Poll and touchpad can control the robot at the same time.

Modbus and IO priority

1.Enter the teach pendant through the display, and find modbusAddr.json in the robot/config/ directory.

2.Open modbusAddr.json with vi editor.



```
{
  "config" : {
     "coexistIOControl" : false,
     "modbusPriorityHigh" : false,
     "startGPPointName" : "GP0001"
  },
  "controllerID" : "1F16378515D30963",
  "modbusAddr" : [
     {
        "addr" : 1,
        "cExplain" : "◆◆◆◆◆◆◆◆◆◆◆◆ ",
        "cName" : "controllerConnectState",
        "cSize" : 1,
        "cType" : "3x"
     },
     {
        "addr" : 5,
        "cExplain" : "◆◆◆◆◆ ",
        "cName" : "robotSelect",
        "cSize" : 1,
        "cType" : "4x"
     },
     {
        "addr" : 5,
```

3.coexistIOControl: false means modbus and IO are not used at the same time, i.e. IO cannot control the robot when modbus is connected; true means modbus and IO are used at the same time, i.e. modbus and IO can control the robot at the same time.

4.When coexistIOControl is false, whether modbusPriorityHigh is false or true, modbus always has a higher priority than IO by default, and IO cannot control the robot when modbus is connected.

5.When coexistIOControl is true, if modbusPriorityHigh is false, it means that modbus and IO are used at the same time and IO has a higher priority than modbus, i.e. modbus and IO can control the robot at the same time and modbus runs according to the IO settings (settings on the teach pendant), e.g. breakpoint and current line run.

6.When coexistIOControl is true, if modbusPriorityHigh is true, it means that modbus and IO are used at the same time and modbus has a higher priority than IO, that is, modbus and IO can control the robot at the same time, but modbus and IO each run according to their own settings, for example, if you close the breakpoint execution in the teach pendant, then inputting 0 (stop) and then 3 (breakpoint execution) in modbus address code 19 will start breakpoint execution, but IO control cannot.

> Modbus touch screen usage process

This section uses Weiluntong touch screen and modbusTCP protocol as examples; the touch screen model is MT6071iP.

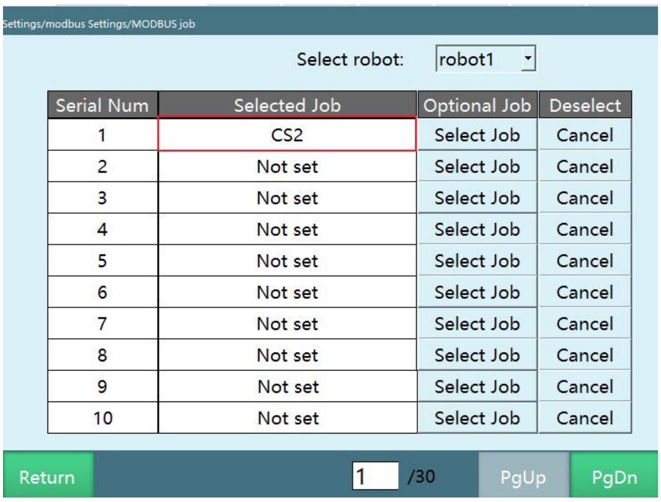
Write program——Set Modbus program——Set Modbus parameters——Switch to remote mode——Touch screen preparation——Select program——Run

(1)Write program

Write the program with the teach pendant, and make sure it can run normally.

(2)Set Modbus program

Set the program in "Settings-Modbus setting-Modbus program", if the setting is successful, the selected program list will display the program name.



Settings/modbus Settings/MODBUS job

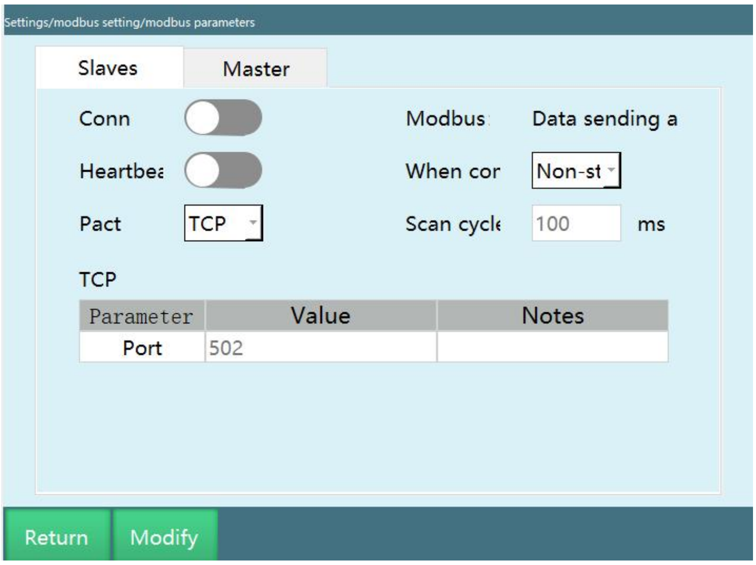
Select robot: robot1

Serial Num	Selected Job	Optional Job	Deselect
1	CS2	Select Job	Cancel
2	Not set	Select Job	Cancel
3	Not set	Select Job	Cancel
4	Not set	Select Job	Cancel
5	Not set	Select Job	Cancel
6	Not set	Select Job	Cancel
7	Not set	Select Job	Cancel
8	Not set	Select Job	Cancel
9	Not set	Select Job	Cancel
10	Not set	Select Job	Cancel

You can set up to 1000 programs in total

(3)Set Modbus parameters

Set the protocol to TCP in "Settings-Modbus settings-Modbus parameters", set the controller as slave, leave the IP unchanged, set the port to 502, and enable the connection; it will take effect after restarting the controller.



Settings/modbus setting/modbus parameters

Conn ○

Heartbea ○

Pact TCP

Modbus Data sending a

When cor Non-st

Scan cycle 100 ms

TCP

Parameter	Value	Notes
Port	502	

Modbus parameter description

Connect: You need to turn on the connection button after Modbus setup, and you can check the connection status on the right side.

Heartbeat detection: turn on to detect the frequency of sending and receiving between Modbus and the controller, and heartbeat detection shows that data sending and receiving is off after the Modbus connection is disconnected

Protocol: TCP, RTU.

Master/Slave: master station, slave station.

TCP parameters

IP: Modbus device IP address, only valid when it is set as the master station.

Port: Modbus device port

RTU parameters

Slave ID: the default is 1

Port: serial port number of the controller

Baud rate: fill in the baud rate corresponding to the touch screen

1.Switch to remote mode

Turn the mode selection key to the remote mode position or click the mode status in the program and select the remote mode.

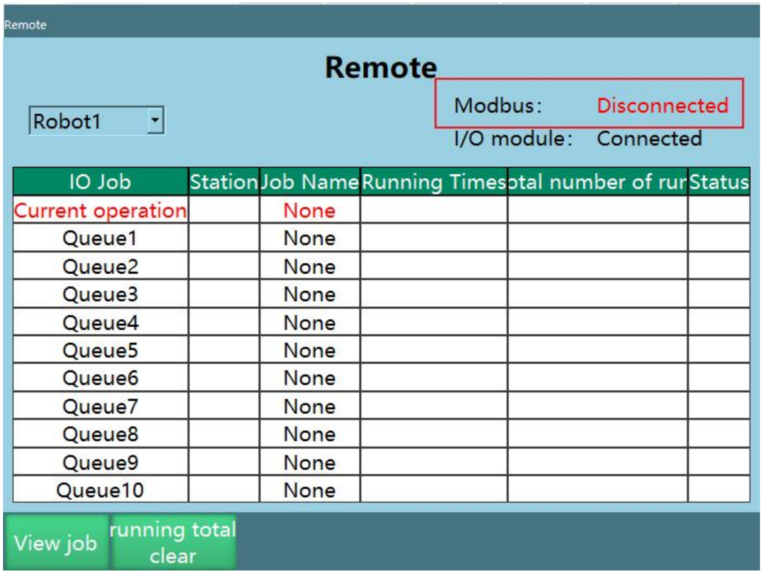
Note: When the controller is connected to IO, Modbus device and teach pendant at the same time, the priority of the three devices is teach pendant>Modbus device>IO device. When switching to remote mode, the Modbus device is valid and the IO device is invalid. If the enable button in the Modbus device is turned off at this time, the IO is valid.

2.Touch screen preparation

Connect the RJ45 network port of the touch screen, the network port of the teach pendant, and the network port of the teach pendant on the controller to the same switch.

Connect the touch screen to the controller: IP: 192.168.1.13; port: 502.

After the touch screen program is edited and run, the modbus will change from unconnected to connected status on the remote interface of the teach pendant.



Remote

Robot1

Modbus: Disconnected

I/O module: Connected

IO Job	Station	Job Name	Running Times	otal number of rur	Status
Current operation		None			
Queue1		None			
Queue2		None			
Queue3		None			
Queue4		None			
Queue5		None			
Queue6		None			
Queue7		None			
Queue8		None			
Queue9		None			
Queue10		None			

3.Select program

Use the touch screen to write 1 to address code 45 of type 4x, select demo program 1 for robot 1.

Use the touch screen to write 5 to address code 61 of type 4x, set running times 5 for robot 1 (not valid); use the touch screen to write 1 to address code 71 of type 4x, confirm the modification of running times (running times 5 takes effect).

Run

Use the touch screen to write 1 to address code 29 of type 4x, switch to servo-ready status.

Use the touch screen to write 1 to address code 19 of type 4x, run the job file.

LIGENT TECH CO., LTD

Website

YouTube

Official Website: http://ligentrobot.com
E-Mail: info@sotrobot.com
whatsapp:+86-13551010933

The introduction of the products is only for reference, the products and services delivered shall be subject to the specific contract.