

**Manual for Simple PLC All-In-One Programmable Time Relay  
(V3.6)  
(The manual is applicable to all the models of flagship products  
in 2021. )**

**Contents**

Chapter 1	Summary-----	2
Chapter 2	Hard ware Parameters -----	4
Chapter 3	Human-Computer Interface and Its Direction for Use-----	6
Chapter 4	WIFI Connection-----	9
Chapter 5	Instructions-----	16
Chapter 6	RS485 Communication and Its Touching Screen Links-----	29
	1: The Pricinple of Touching Screen Linking-----	29
	2: Superior PC Communication and Extended Communication Examples-----	30
	3: Operation Manual for Computer Software-----	35
Chapter 7	Programming Exemplification-----	39
	1: Cases of Common Programming-----	39
	2: Cases of Accessing to External Registers Through RS485-----	45
	3: Cases of High Speed Pulse Output-----	46
Chapter 8	Wiring Diagram-----	48

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

## Chapter 1 Summary

Welcome!

The simple PLC all-in-one programmable time relays produced by us cover the following models.

8 in 8 out model(relay output and transistor output optional)

12 in 12 out model(relay output and transistor output optional)

16 in 16 out model(relay output and transistor output optional)

32 in 32 out model(relay output and transistor output optional)

The products are characterized by simple programming and easy operation. A common engineer can understand the program and make program on site, which reduces the cost to a great degree. At the same time, the products can be used as time relays of multiply paths. Each path of the module can work either independently or dependently, which makes it possible to realize more complex functions than the traditional products.

The products boast of the following characters:

**Compatible with two pulse outputs, and adjustable with the frequency scope of 0Hz~150KHz**

**Capable of free switching between Chinese and English interface.**

**Support logical AND, OR, Addition, and Subtraction operations.**

**Capable of working with 20 programs simultaneously.**

**Afford 25 timers, 50 counters and 50 time meters**

**Capable of working with 20 external registers, which makes the access to external devices, data exchange and processing possible.**

**Time delay precision: 0.01 second.**

**Compatible with NPN and PNP digital signal.**

**Compatible with 2 high speed pulse inputs**

**Optional matching with 2 12-bit analog signal inputs(able to detect the signals of 0~20mA, 4~20mA, 0~5V, 0~10V)**

**Optional matching with 2 high precision analog outputs( able to realize the outputs 0~20mA, 4~20mA, 0~5V, 0~10V)**

**Optional matching with 2/4 high speed pulse outputs with the frequency of 0Hz~150KHz**

**Optional matching with RS485-MODBUS-RTU communication.**

**Compatible with 2 independent 485 ports (32-in-32-Out, PLC exclusive)**

**The main 485 port: used to download programs, connect computer and touching screen, read**

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

**external devices, control external RTU equipments and DTU communication.**

**The subordinate 485 port: used to connect touching screen, 2G/4G DTU and 485 RTU equipments.**

**USB Port (32-in-32-out, PLC exclusive), standard USB2.0 high-speed Port**

**Connected to the computer, the port is used to realize the functions such as download, upload, simulation, manual control and etc. it replaces the tradition 485 port and boasts of high speed.**

**Compatible with flickering signal trigger**

**Compatible with all the expansion output ports of 485 communication module.**

**Compatible with touching screens.**

**Compatible with arbitrary combination of perpetual calendar control**

**Without trapezoid programming**

**Replaceable with common PLC**

**Boasting of user-defined key imitation pause /scram functions, several digital inputs accessible with different sensors, logic operation of PLC, user-defined programming of output, and independent or combined timing functions in different time quantum.**

**Able to replace several time relays.**

**More powerful programming combination and function than time relays with 0.01 second precision. Arbitrary combination of year, month, date, day and hour makes a serial of specific operation possible.**

**Able to replace current modules and voltage modules of PLC analog. Internal analog operation makes programming visible.**

**Program encrypt: encryption of programming can avoid alteration by others.**

**Offline management: direct editing on the control panel can be done without connecting to the computer. A HD color liquid display is used with options of simplified Chinese, traditional Chinese and English in the interface. It is very user friendly with corresponding displays of menu management, orders and input, which makes operation very easy. Common users can operate it very freely without the trouble of learning trapezoid diagram.**

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

## Chapter 2 Hardware Parameters

### 一、 Specification

#### 1: Current supply of Controller

An Internal DC-DC voltage reduction device is used to guarantee working under the constant voltage

Reversal connection proof

Current supply with recommended switch power supply

Voltage : 12~24VDC with transistor output model

24VDC, 12VDC with relay output model (customized)

Attention: the current supply of terminal(24V GND) in the simple PLC all-in-one relay can only supply the control itself(see the third table for parameters).

#### 2: Power supply of the load

Another power supply is used for the output terminal. If the output voltage is same as that of the simple PLC relay, they can share a power supply. Otherwise, different power suppliers are used.

### 二、 Input and output Specification

#### Input and output port:

The input port, with a built-in 5V up and over current and over voltage protection device, can connect with keys, proximity switches, air cylinder magnetism switches, touching points of relay, photoelectric sensors, Hoare sensors and so on.

The output port can work with all kinds of switch loads, contactors, electromagnetic valves, relays, LEDs and alarming devices. With the devices of analog output, the speed can be adjusted by controlling a 0-10V frequency transformer.

#### Pulse output port:

Used to control stepper motors/sever motors

Realize high precision location of high speed response.

Several axles working simultaneously.

Supporting the modification of motor speed and distance through touching screen.

A built-in auto-superposed trapezoid diagram makes the motor slow start and stop possible and ensures the maximum torque.

Three units adjustable: millimeter, round and pulse quantity

Zero setting function makes the device look for mechanical zero automatically every time when the

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

equipment is started.

### 三、 Performance Parameters

Item	Parameters	
Product name	Simple PLC All-In-One Programmable Time Relay	
Output type	Relay type	Transistor type
Power supply	10W (<300mA/ DC24V) <b>Reversal connection proof</b> <b>Input voltage is not greater than 28V</b>	6W (<200mA/ DC12-24V) <b>Reversal connection proof</b> <b>Input voltage is not greater than 33V</b>
Output format	8/12/16 relays(constant open)	8/12/16 transistors(Open-drain output)
Output load	5A/250VAC、 5A/30VDC	2A/12-24VDC
Contacto r lifetime	The life span of electric parts and mechanical parts is 100,000 times and 3,000,000 times	Over one hundred million times
Input points	8 / 12 / 16 points	
Input signal	>2mA /DC12-24V compatible with PNP and NPN signals	
Analog input	<b>0~20mA ,4~20mA, 0~5V, 0~10V</b> <b>Maximum voltage &lt;33V Maximum current &lt;30mA(continuous input)</b>	
Analog output	<b>0~20mA (24 V output)</b> <b>4~20mA (24V output)</b> <b>0~5V, 0~10V</b>	
Dimensions	8-in 8-out、 12-in 12-out: 145mmX90mmX40mm, 16-in 16-out: 160mmX95mmX56mm , 32-in32-out: 300mmX110mmX60mm	

**Attention: the module of transistor output is suitable for high frequency and the module of relay output for low frequency.**

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

## Chapter 3 Human-Computer Interface and Its Direction for Use

### 1、Description of functions of keys

**Setting: 1, press the setting key for 3 seconds and get access to menu under the working condition.**

**2, Press the setting key and enter into the programming state.**

**“←”“→”**: the user can move the cursor right or left with these two keys under the setting or programming state. Under the working condition, the user can switch the following interfaces of time piece display, timing device monitor, counter monitor, timer monitor and analog input monitor and pulse date monitor, temperature collector monitor and pulse collector monitor)

**“↑”“↓”**: with these two keys, the user can move the cursor up/down under the condition of programming or setting.

Under the working condition, the user can check more information with these two keys.

**“+”“-”**: the user can add or minus the data of the selected items. Under working condition, the user can browse the states of 20 programs with these two keys.

**“Delete”**: the user can delete the contents specified by the cursor under the programming state. (successive deletion with pressing the key)

**“Insert”**: under the condition of programming, the user just needs to press the Delete key to insert a new order before the place of cursor.

**Remarks: input is saved automatically with “Confirmation” or “Saving .**

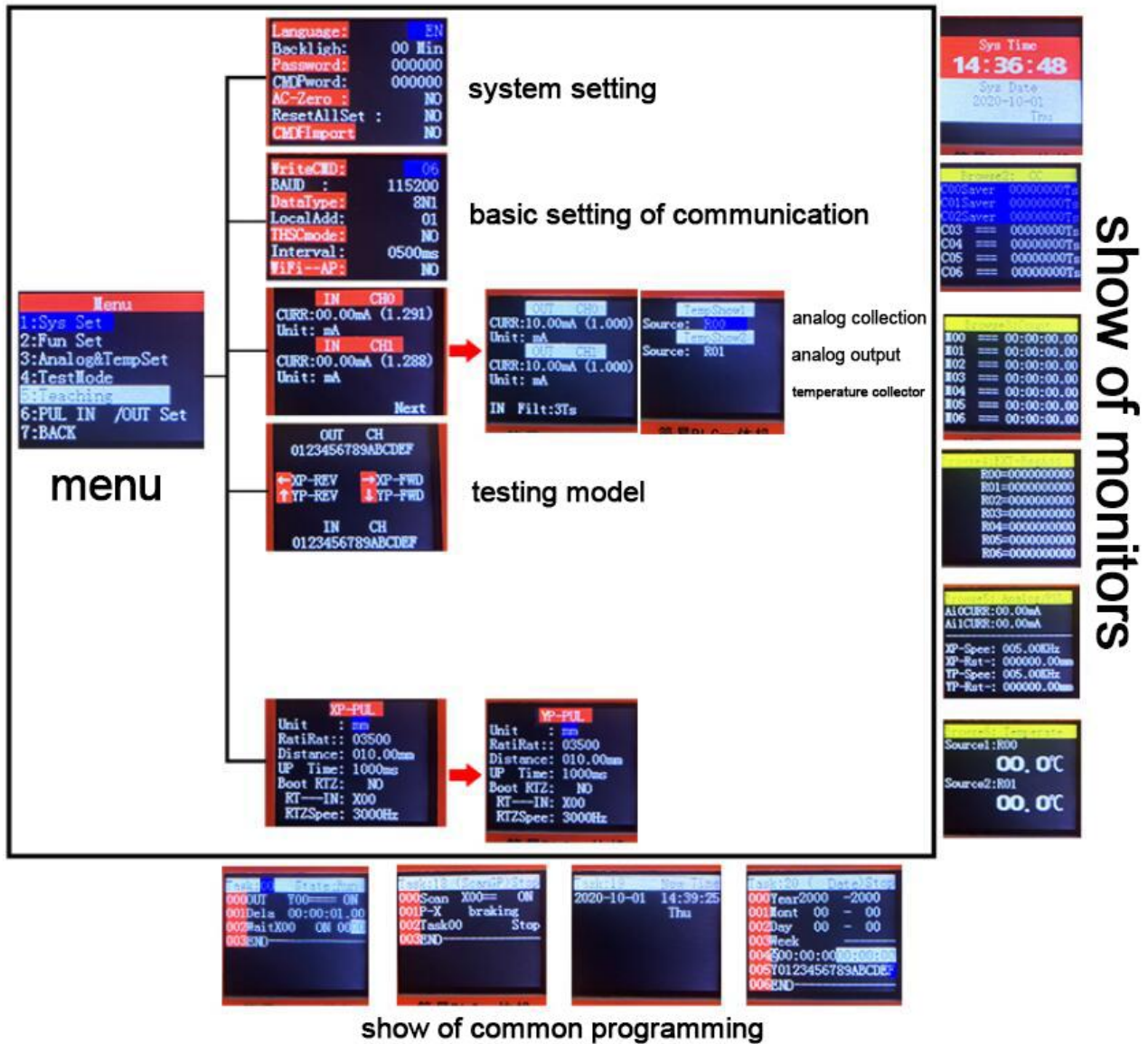
### 2、Menu Explanation

- 1, the user gets access to the menu with pressing the setting key for 3 seconds;
- 2, use the keys of “←” “→” to choose desired menu.
- 3, press the setting key to enter into sub-menu to modify the corresponding parameters
- 4, press the key setting after the modification to save the modification and then return to the menu at the next higher level.
- 5, in the interface of the menu, choose Return key and press Set to exit the menu and return to the working condition of the program.

Please see the diagram below for more details of the menu and sub-menus.

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

# Menu Functions

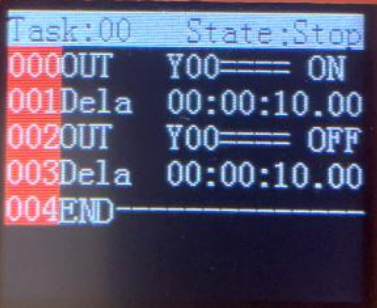
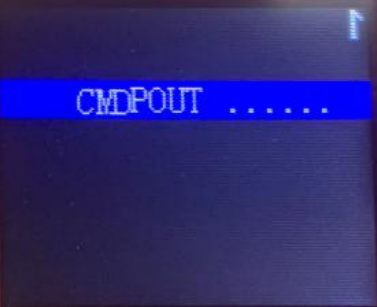
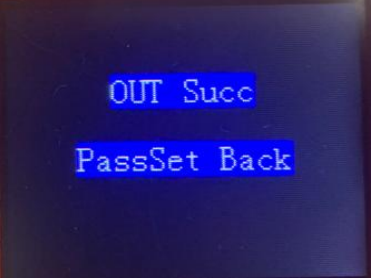
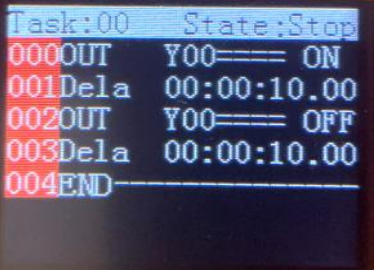
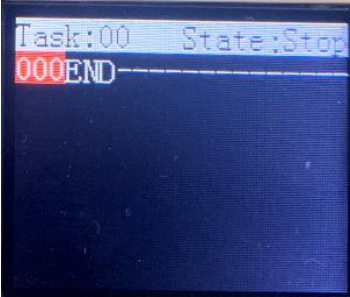
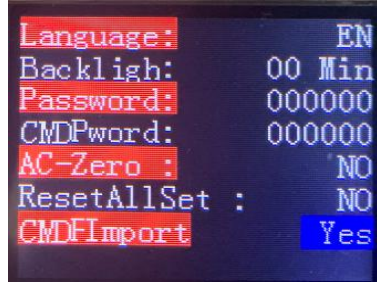
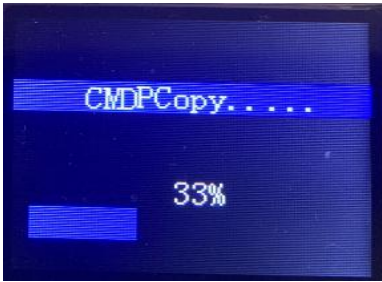
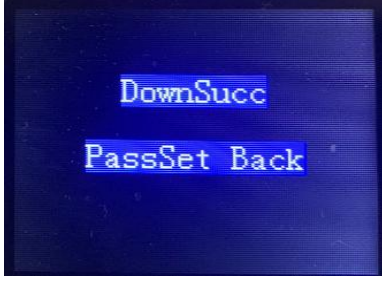
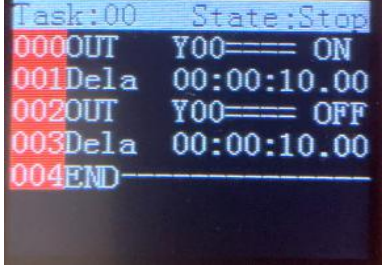


### 3、Description of program copy functions

First, attention should be paid to the version of PLC hardware, which is displayed when the device is started. Program copy function and programming function are available only with the devices of version 3.0 or higher level. Copy can be made when the two functions' requirements are met. Once the version is confirmed, connect the 485 ports of two devices with cable.

After the programming of the main station is finished and set the passwords of program and management to be 000000. When the main station is working, long press the key Setting to enter into system setting interface. Click Program Import and press the setting key to copy the codes automatically from the main station.

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

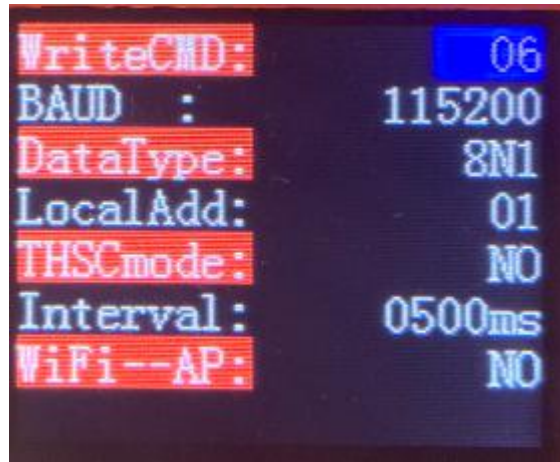
Main PLC	Sub PLC	Wiring and Explanation
   	    	<p>Operation methods: Long press the setting key of sub PLC to enter into the menu. In the system setting interface, move the cursor to the icon of Program Import, press + and change the state to Open and then click the Set key to copy the program from the main station PLC to the substation PLC. When the download is over, click the Setting key to exit.</p> <p><b>Attentions:</b></p> <ol style="list-style-type: none"> <li><b><u>1, Both the main station and the sub-station are compatible with RS485 ports.</u></b></li> <li><b><u>2, Copy operation can only be made when the main station is under working condition.</u></b></li> <li><b><u>3, The codes for the administrator in the System setting and the program are 000000.</u></b></li> <li><b><u>4, Ensure the 485 connector is fixed firmly during the copy. Otherwise, the copy will be cancelled automatically and Failure will be displayed on the substation screen. In that case, the program in the sub station is incomplete and has to be copied again.</u></b></li> </ol>

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*



## Chapter 4 WIFI Connection

**Open WIFI hotspot ( the software version for PLC is required to be V3.6 or a higher version)**

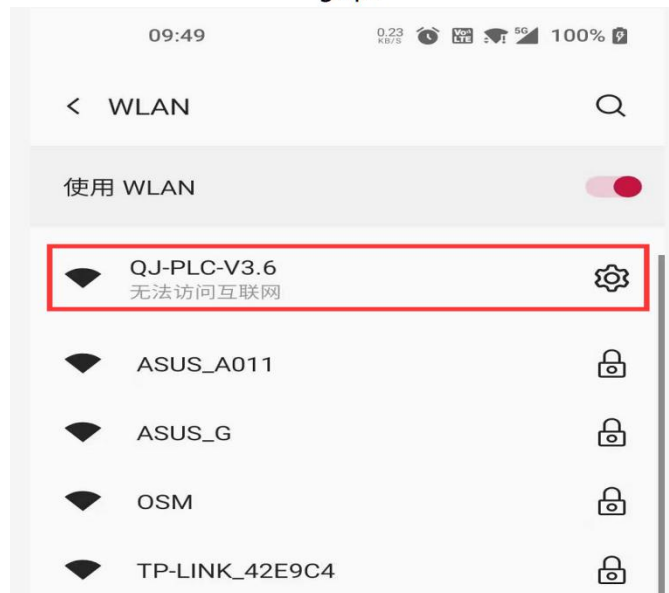


Long press Setting key to enter the main menu-->485 setting-->WIFI hotspot (as the photo above)

Install “android-build-debug.apk” PLC programming software (only Android version is available at present and the software for Apple IOS is under development)



android-build-debug.apk

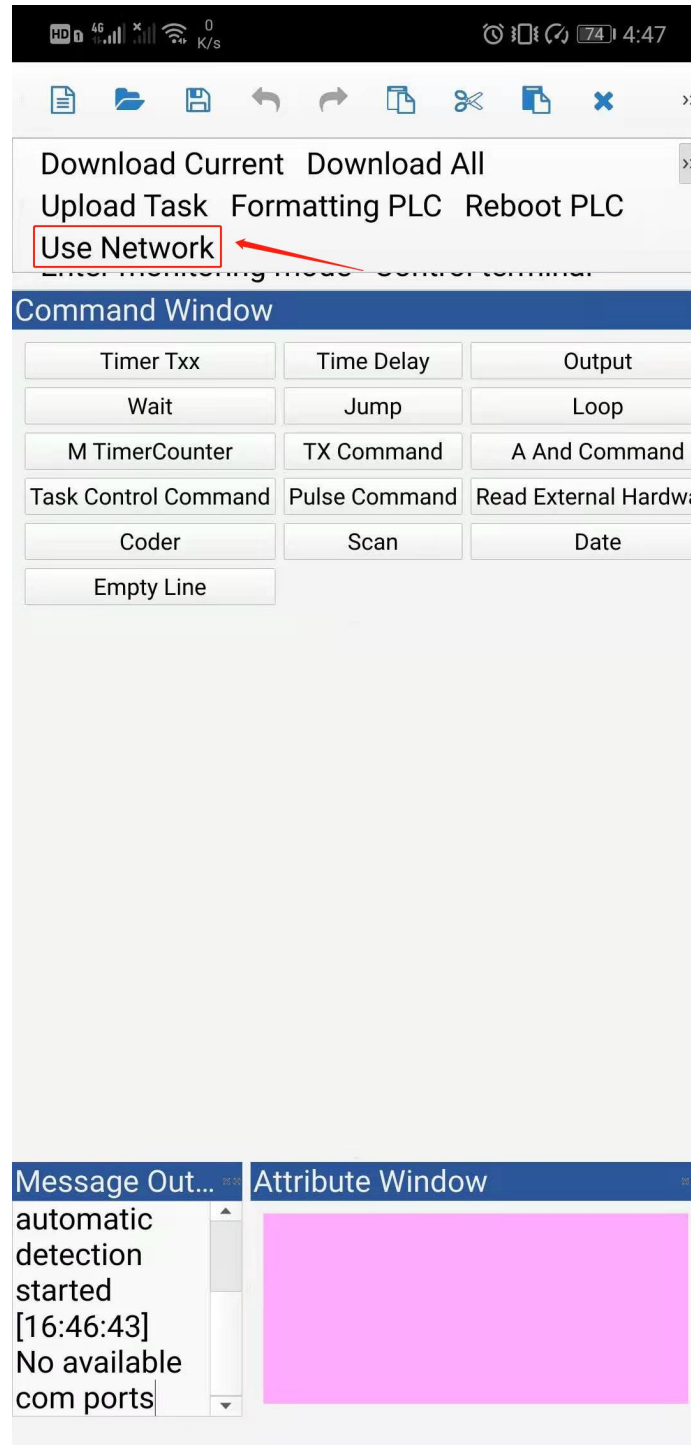


Password: 88888888

When WIFI is connected, the screen will display” connected, no website found”

Open the“QingJunPLC”APP and click the icon to use the website.

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*



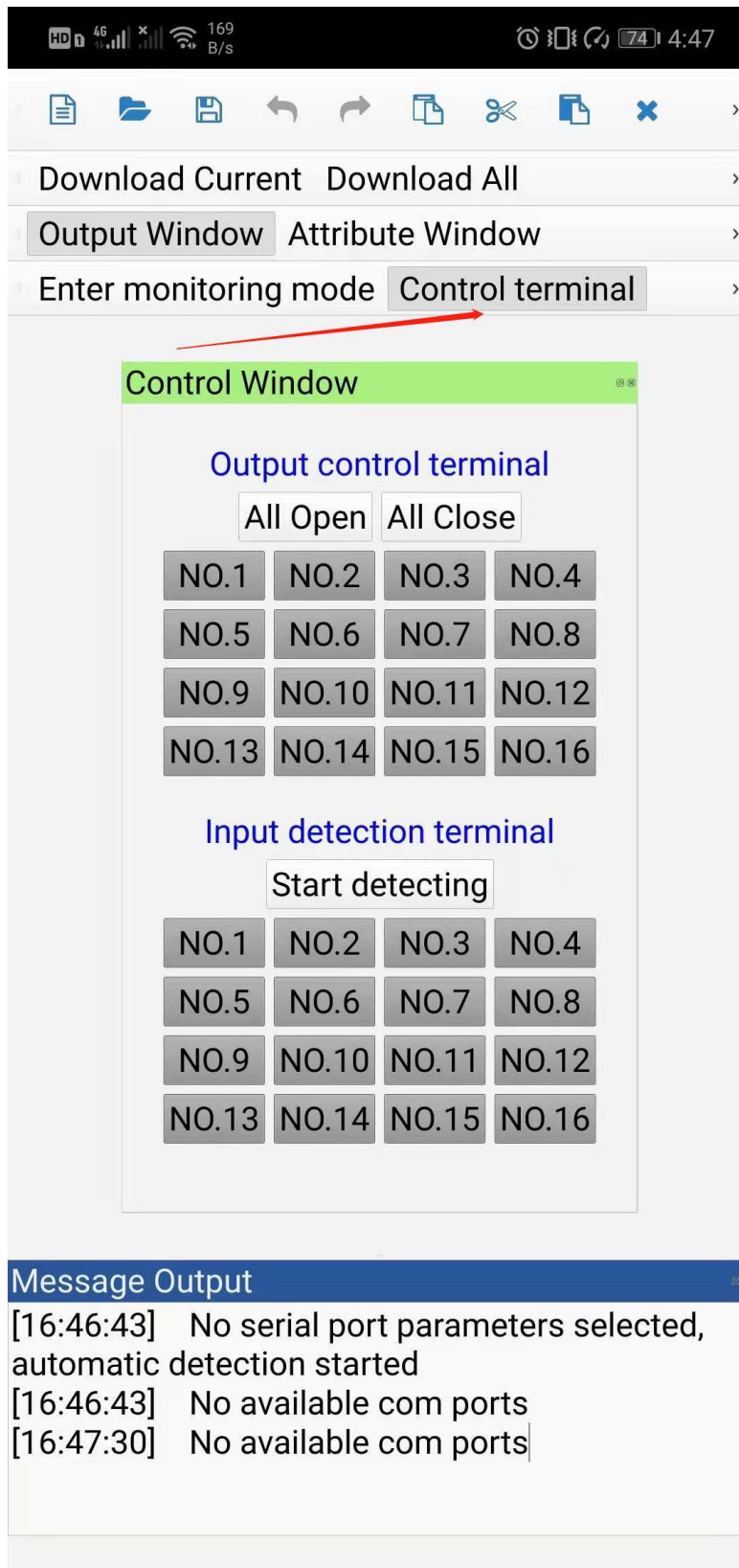
When “Connected to the sever” is shown on the Attribute window, it means that the target PLC is connected.

(attention: one mobile phone can only work with one PLC)

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

Click the control terminal above and control manually the output of PLC.

Click Control Checking to update the state of PLC input port automatically



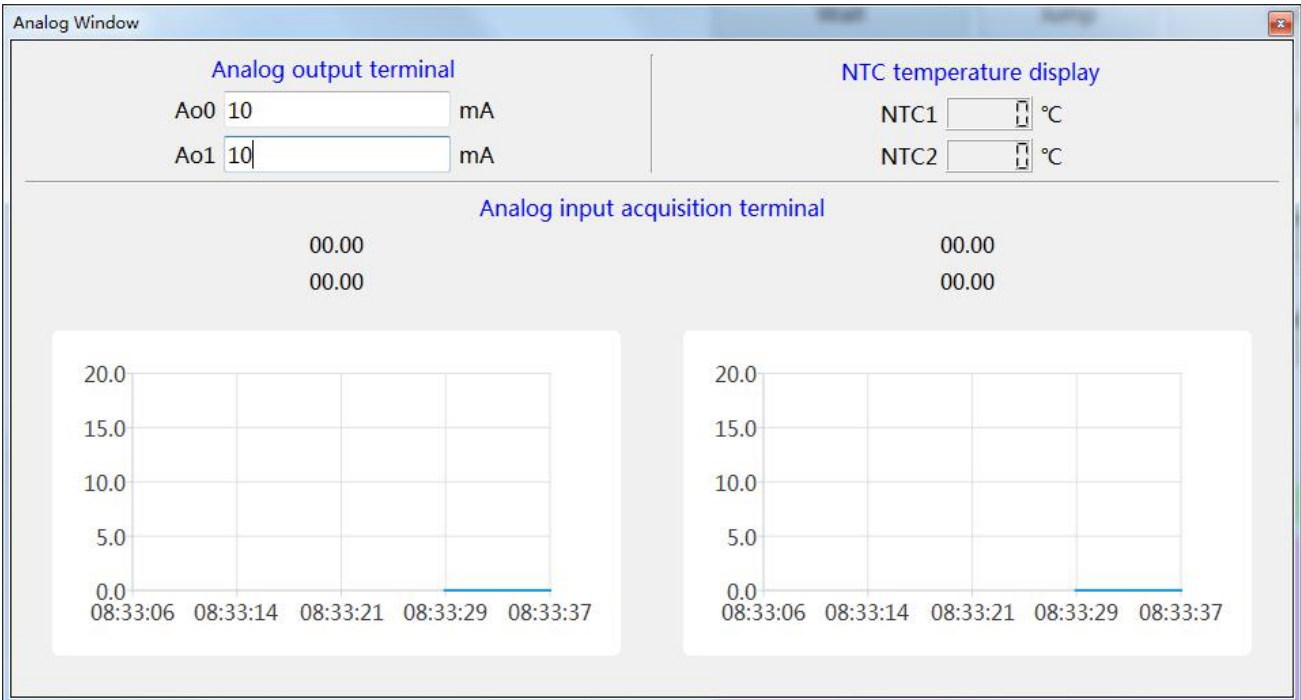
Click "Download" to download the current internal program to mobile phone. User can also click the

icon  to save the program to the mobile phone for back up. (for the sake of program safety,

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

PLCs are required to be deciphered before they are downloaded to the mobile phones.

Record display of analog output



The superior computer supports 2 analog outputs online collection record and real time curve depiction, which makes the user know the current and voltage curve easily.

//=====Programming methods(it is similar with a PC and a mobile phone)

There are 3 methods to make program as below:

- 1, Input the order through the 9 keys on the PLC panel,
- 2, Software in the PC
- 3, APP in the Android mobile.

The programming on the computer mainly consists of the following procedures:

- 1, Connect the terminal A and B of PLC with the cable of USB adapter for 485 and its USB port to the computer.



QingJunPLC.exe

- 2, Open PLC programming software

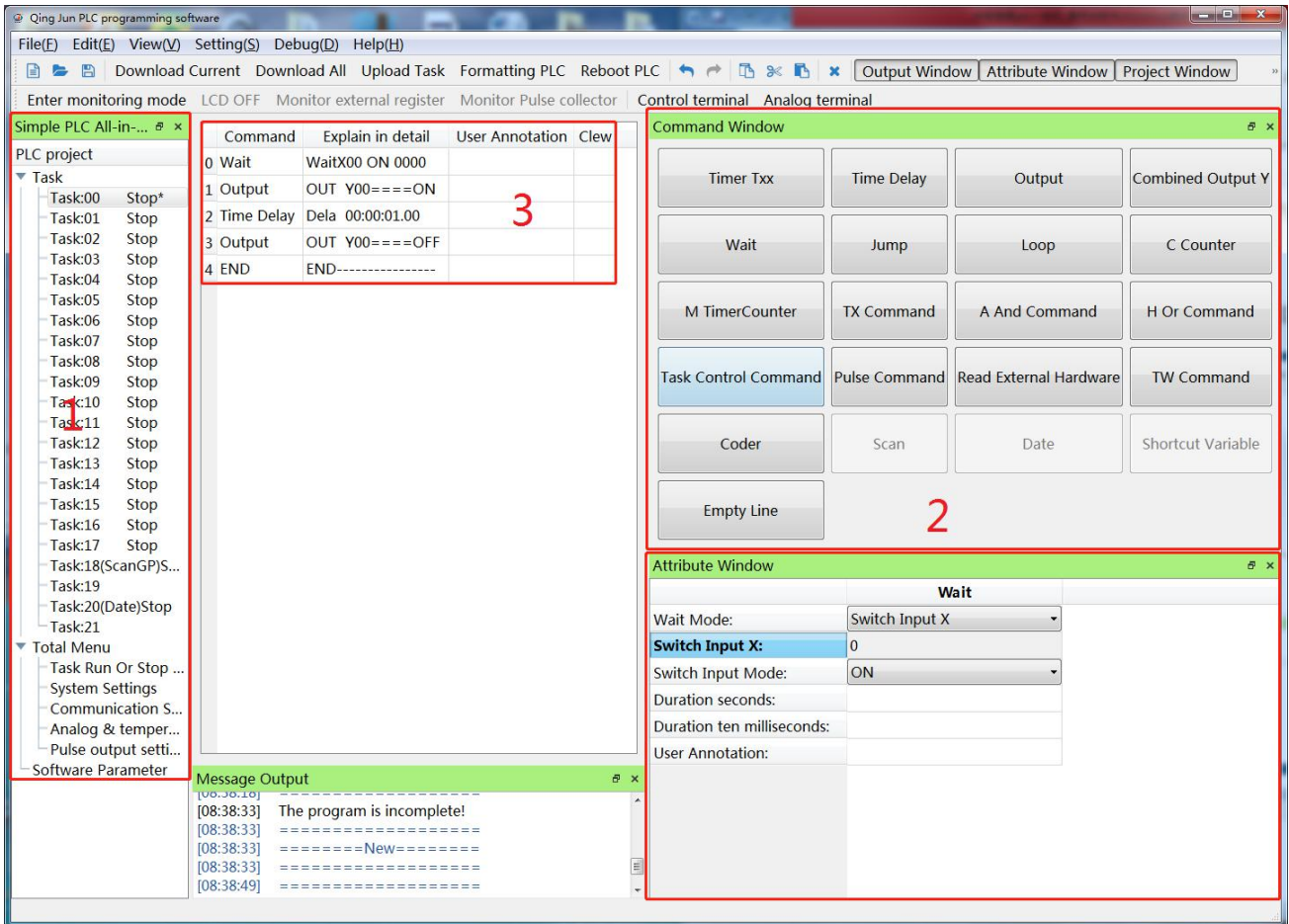
- 3, the software search automatically PLC and the current program for the user to edit.

The user can edit the program with the 5 steps shown in the picture below,

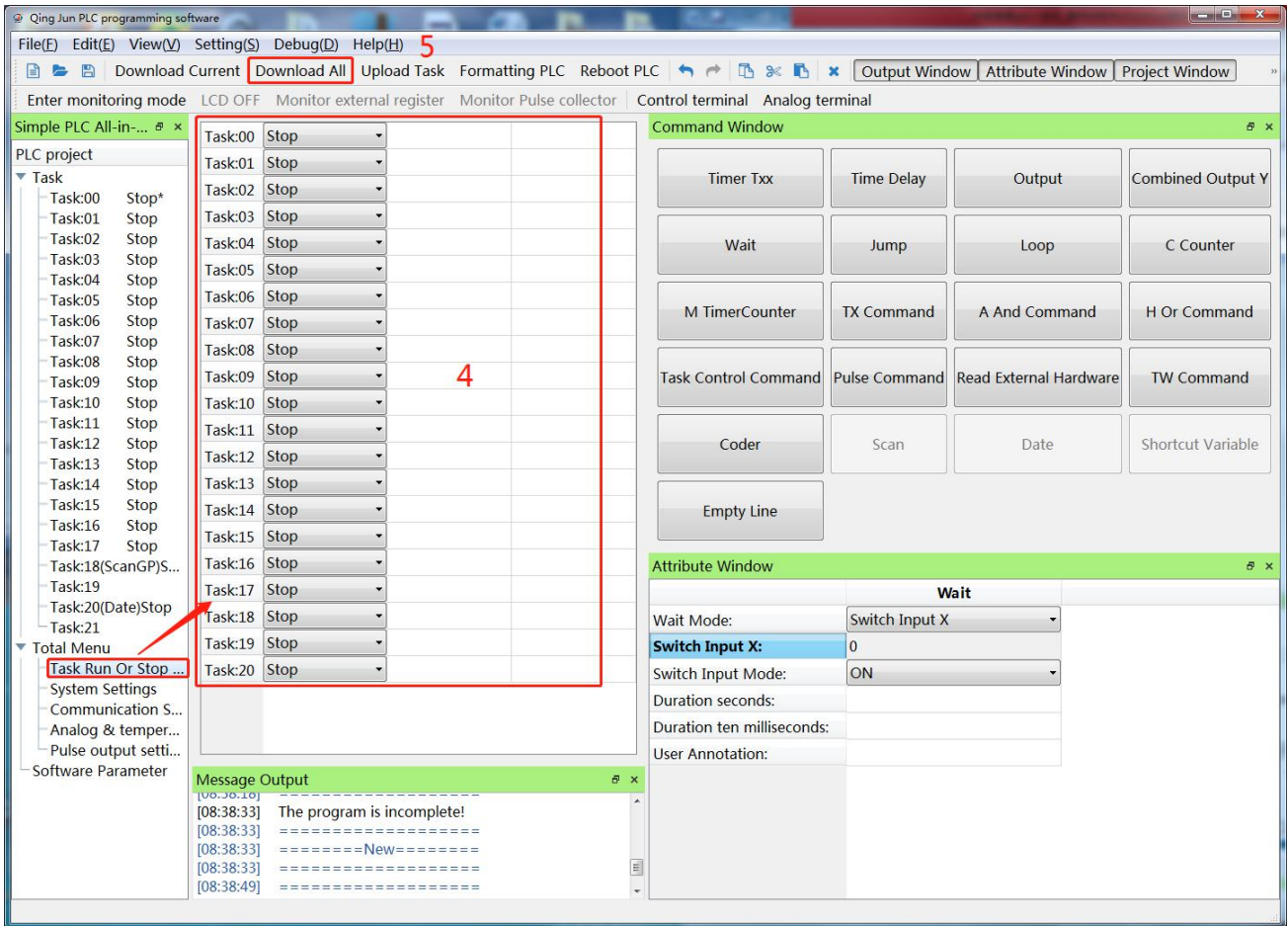
- 1) Select the serial No. of the program
- 2) Select the order to be added

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

- 3) Edit the attribute of the order according to the user's requirement(the contents in the 3<sup>rd</sup> window are to be adjusted in the attribute column)
- 4) Set the initial state when the program is powered on, Start/Close
- 5) Click “Copy All” to download the programs to the PLC
- 6) PLC will stay in the interface of Setting menu after the programs are saved. The programs will run automatically after manual exit from the interface.



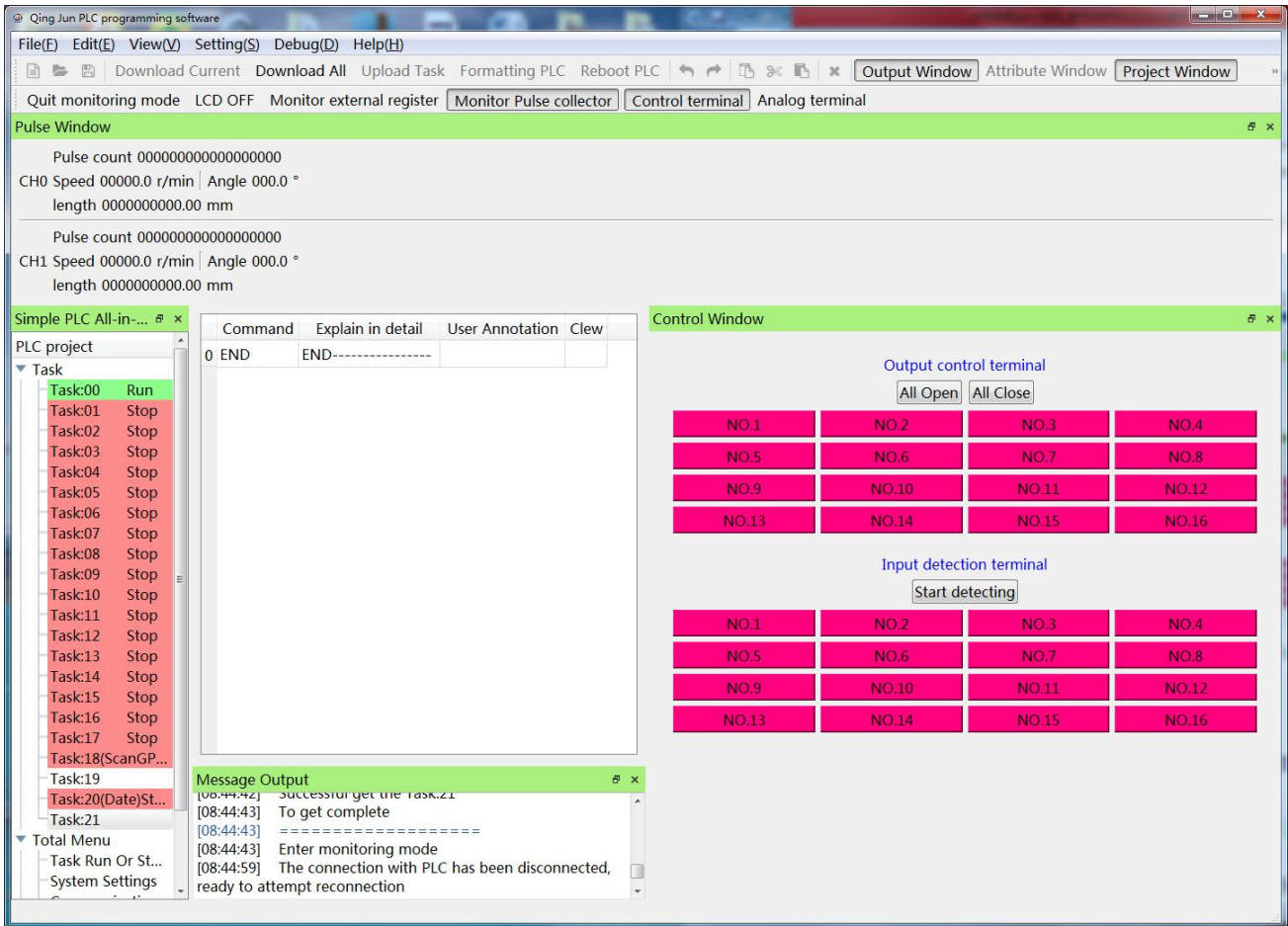
*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*



**Real time imitation of software==//**

- 1 The program enters into the state of Real time imitation with a click of “Monitoring”
- 2 The software will give a hint to the user to save the edited program and the user can save it at the selected route.
- 3 The user can check the collected data and diagrams with Pulse collector and Analog ports in the monitoring mode(see the picture below)
- 4 Click the Control Unit to control manually the output port of PLC

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

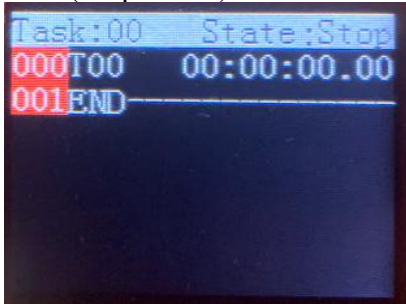
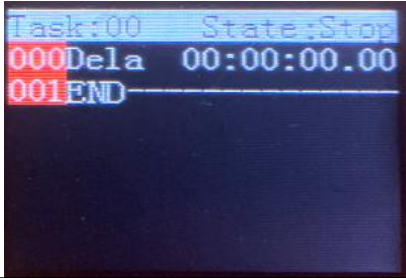
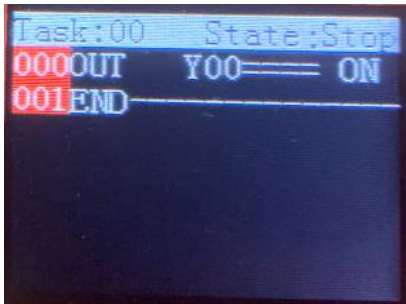
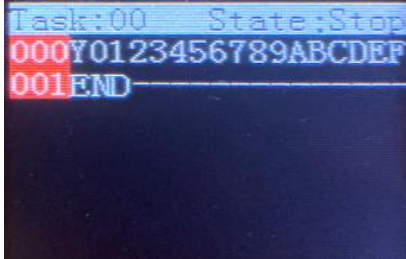


- Under monitoring state, the program in the 4<sup>th</sup> interface will update timely according to the actual state, which is convenient for the user to know the working process.
  - Supporting 2 high speed input encoders working simultaneously
  - Compatible with incremental encoders and absolute value encoders
  - Applicable to single line input and double line input
  - Compatible with NPN and PNP input
  - Applicable to FW direction and REV direction
  - Capable of checking angle, speed and distance

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

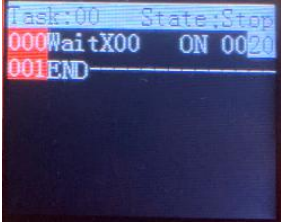
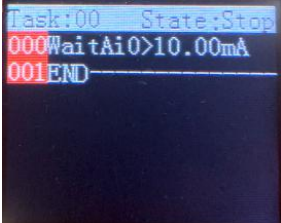
## Chapter 5 Instructions

(Attentions: the system is designed according to the 16 in and 16 out hardware. Therefore it is compatible with 12 in and 12 out and 8 in and 8 out hardware. If the products you bought belong to 8 in and 8 out, you can operate with the frontal 8 input, i.e. X00-X07 and output Y00-Y07. It does not work with the output Y13.)

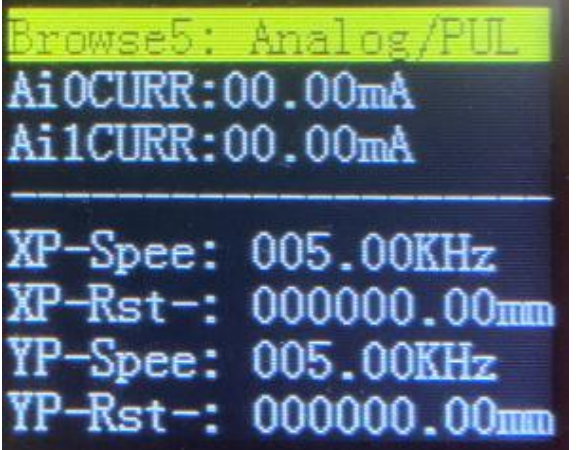
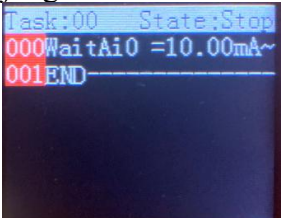
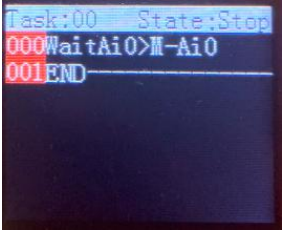
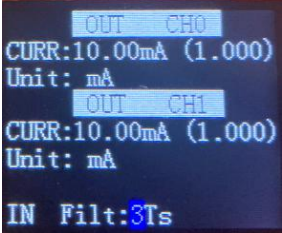
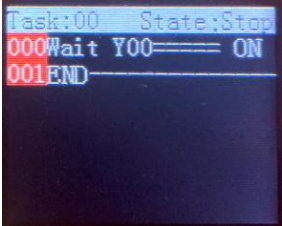
Item	Instructions	Function Description
Instructions related to program 00-17	1: Timer TXX (scope 0-49) 	Timer can be employed quickly to take place of common delay. At the same time, it is more convenient to modify the delay in a batch. Capable of working independently
	2: Delay 	Common time delay(hour: minute: second: 10 millisecond) Special function: the dynamical value of the timer is activated by the hidden symbol behind delay to realize the special effect.
	3: Output Y00~Y15 Ao0~Ao1 	1: Close, Open and and Turn over  1 Voltage(0-10V), Current(0~20mA),supporting the variable shortcut of Aval00~49 2 Output following input
	4: Combined output Y 	<b>All output</b> Y0123456789ABCDEF All Open Yxxxxxxxxxxxxxxxx Y08~Y15 output Y----- 89ABCDEF Y00~Y07 turn over Y↑↑↑↑↑↑↑↑ - - - - -

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

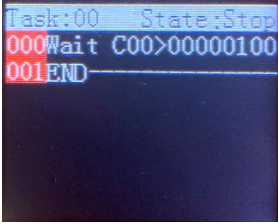
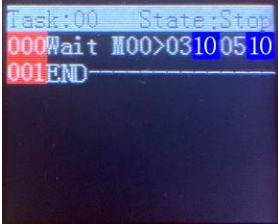
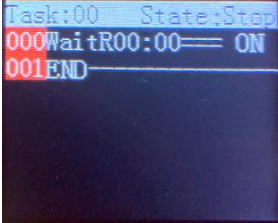


<p>5: Wait until all the conditions meet the requirement to execute the next order</p>	<p>X00~X15 output trigger</p> 	<p>When X00 keeps close, it takes 20ms to execute the next operation (the frontal oo in 0000 means second and the rear 00 10 millisecond)</p> <p>Example 1 Close 0020 means that the next order is executed after the device is closed for 200mS.</p> <p>Example 2 Close 0030 means that the next order is executed after the device is closed for 300mS.</p> <p>Example: Rising edge (execution is done from up to down after triggering)</p> <p>Example: Falling edge(execution is done from down to up after triggering)</p> <p>For example, if a skipping order is added after a waiting order, these 3 orders become an instant judgment. That is say, if the conditions are met, it will skip to the target line to execute the order. If the conditions are not met, the program will execute the order from the line after the Skipping order.</p>
	<p>Ai0~Ai1 voltage current input</p> 	<p>Execution is done when the condition is &gt; (10.00mA or Aval )</p> <p>Execution is done when the condition is &lt; (10.00mA or Aval )</p> <p>Execution is done when the condition is = (10.00mA or Aval )</p> <p>Execution is done when the condition is ≥ (10.00mA or Aval )</p> <p>Execution is done when the condition is ≤ (10.00mA or Aval )</p> <p>Attention: The checked analog data can be viewed in the screen of monitor 5.</p>

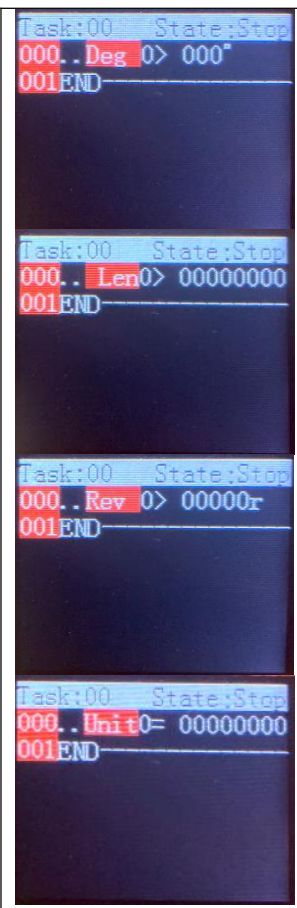
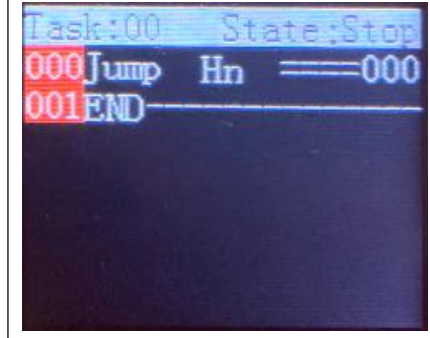

We just provide consult and suggestions to users about the programming and we do not provide free programming service.

	
<p>Ai0~Ai1 current voltage internal judgment</p> 	<p>Execution is done when the condition is 10.00mA ~ 13.00mA          Execution is done when the condition is 5.00 V ~ 8.00 V.          Supporting the variable shortcut Aval00-49 (with reference to the description in program 21)</p>
<p>M-Ai0~M-Ai1</p> 	<p>The data are used to remember the collected analog data for the comparison later.</p>
	<p>The filtration of the channel of analog collection(default value is 3)          This function is applicable to different sample collection speed or filtration smoothness(0-5) with the actual corresponding updating time of 10ms~200ms.</p>
<p>Y00~Y15 output state judgment</p> 	<p>The next order is executed when it is closed.          The next order is executed when it is open.</p>

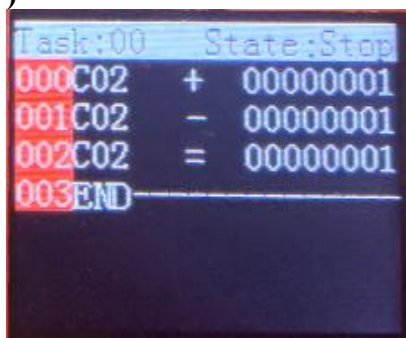
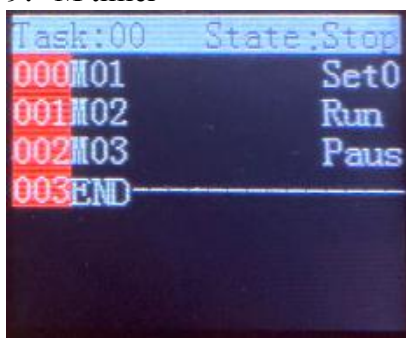
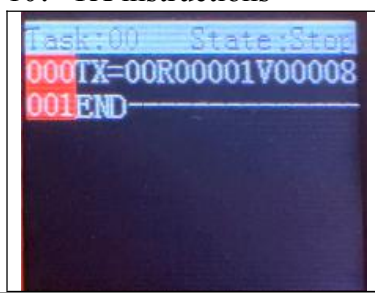
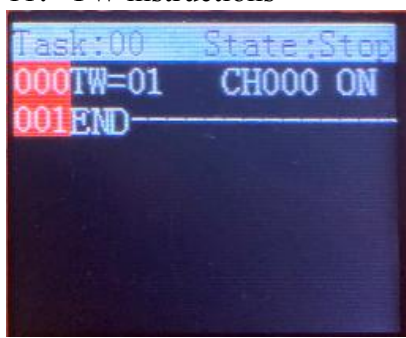
We just provide consult and suggestions to users about the programming and we do not provide free programming service.

	<p>C judgment of counter value</p> 	<p>Example 1 C00 &gt; 00000100 When C00 of the counter is greater than 100, the next order is executed.</p> <p>Example 2 C00 &lt; Cval03 When Coo is less than the variable shortcut, the next order is executed. (with reference to the description in program 21)</p>
	<p>M judgment of timer value</p> 	<p>Wait for the value of counter M00 When the result is greater than 03:10: 05, the next order is executed.</p>
		<p>Wait R00:00=close It is used to judge the product state of input port of RS485 R00 refers to the mapping result from reading the external hardware and 00 refers to the data at the input port of the extended panel.</p>

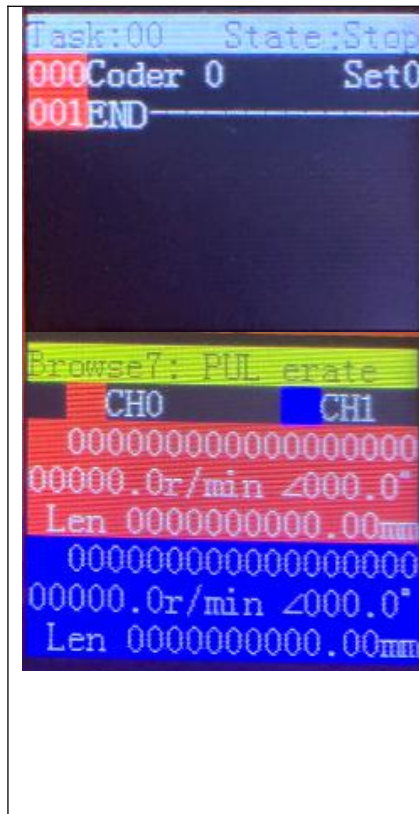
*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

	<p>equal angle 0&gt;000°          equal length 0&gt;00000000          equal distance 0=00000000</p> <p>These concepts are used to judge the input of high speed pulse.</p> <p>Wiring: X04 is connected to the phase A of encoder 0 and X05 is connected to phase B of encoder 0 in order to differentiate the clockwise turning from anticlockwise turning. If it is not necessary to make this differentiation, it goes without connecting phase B.</p> <p>X06 is connected to the phase A of encoder 1 and X07 is connected to phase B of encoder 1 in order to differentiate the clockwise turning from anticlockwise turning. If it is not necessary to make this differentiation, it goes without connecting phase B.</p>
<p>6: Skipping</p> 	<p>Skipping to line 099 to execute the order          If the line No. is out the range of the specification, it will skip to the first line to execute the order.</p> <p>If a skipping order is added after an order such as “wait”, “A” or “H”, these 3 orders become an instant judgment. That is say, if the conditions are met, it will skip to the target line to execute the order. If the conditions are not met, the program will execute the order from the line after the Skipping order.</p>
<p>7: Circling</p> 	<p>Circle 000 line 0000 times          Circle the program between 000 line and this order          0000 times are the circling times.</p>

We just provide consult and suggestions to users about the programming and we do not provide free programming service.

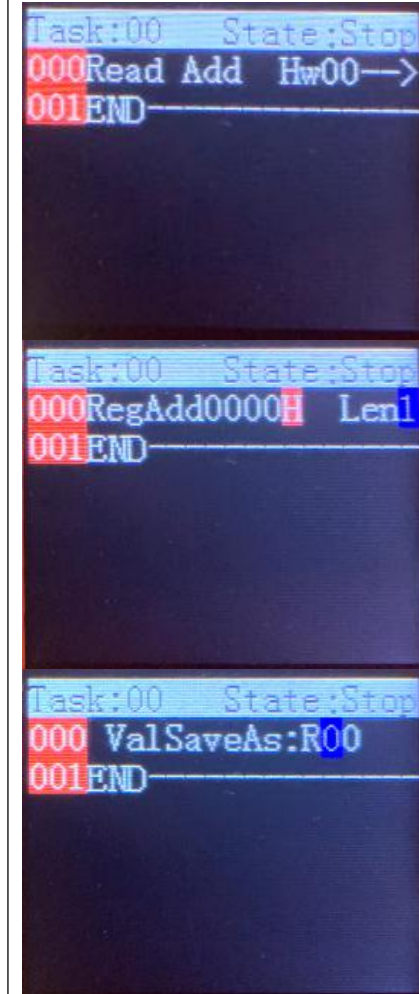
<p>8: Counter <b>(Only C00, C01 and Co2 have the function of blackout holding.)</b></p> 	<p>C02 + 00000001 the counter 2 plus 1 C02 - 00000001 the counter 2 minus 1 C02 = 00000001 the value of counter 2 is 1, Supporting the variable shortcut(with reference to the description in program 21)</p>
<p>9: M timer</p> 	<p>M01 clearing counter 01 returning to 0 M01 start counter 01 starts timing M01 pause counter 01 stops timing</p>
<p>10: TX instructions</p> 	<p>TX=00R00001V00008 An instruction of 8 is sent from the register of 00001 to the equipment of 00 through 485. The maximum address of equipment, register and register value is 99, 65535 and 65535 respectively.</p>
<p>11: TW instructions</p> 	<p>TW=01 Channel 000 Close A signal from equipment 01 is sent to the first path of an external register through 485. Applicable to the bit operation of an external relay module(code 05)</p>
<p>12: Encoders</p>	<p>Encoder 0 clear Applicable to the data clearance of high speed pulse input X04 is connected to the phase A of encoder 0 and X05 is connected to phase B of encoder 0 in order to differentiate the clockwise turning from anticlockwise</p>

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*



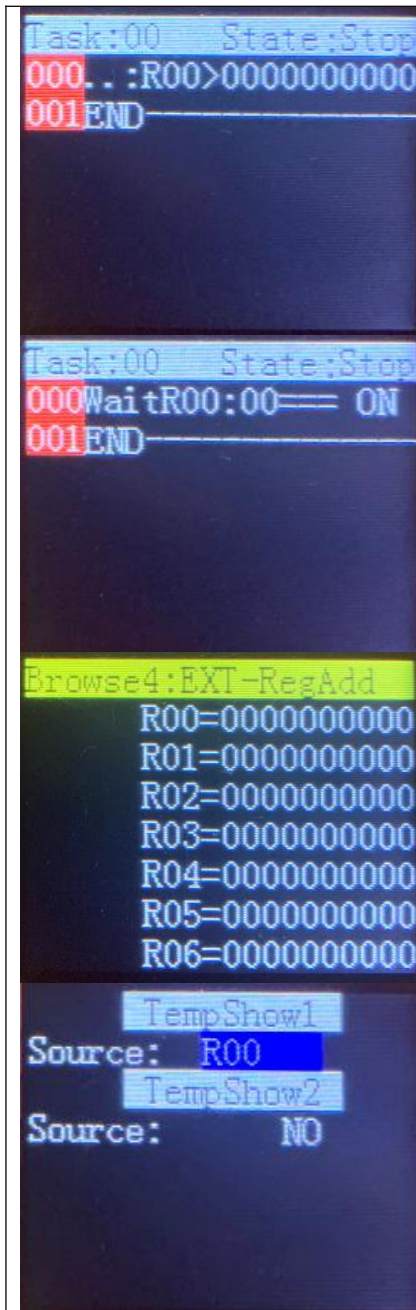
turning. If it is not necessary to make this differentiation, it goes without connecting phase B.  
 X06 is connected to the phase A of encoder 1 and X07 is connected to phase B of encoder 1 in order to differentiate the clockwise turning from anticlockwise turning. If it is not necessary to make this differentiation, it goes without connecting phase B.  
 It can realize the following functions of two encoders,  
 Measuring angle(compatible with APP monitor display)  
 Measuring revolution(compatible with APP monitor display)  
 Measuring distance (compatible with APP monitor display)  
 The obtained data are displayed in the small screen of PLC  
 Also applicable to length counter, revolution monitor, and angle gauge.

13: Reading external hardware



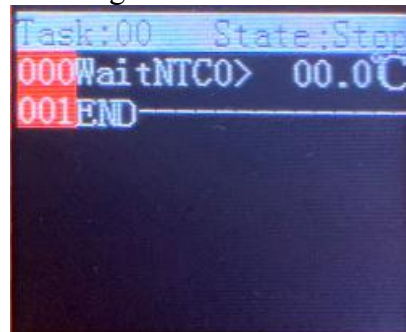
When it is ordered to read an external hardware, No. R must be specified because the display allocation and handling are made according to No. R.  
 PLC can be used as a temporary buffering saver when it is used as main station to inquire the external data. User can execute or trigger a instruction or program based on the result.  
 Generally, PLC can read the external temperature, moisture, revolution and the value of voltage and current if the subordinate equipment is compatible with MODBUS-RTU protocol.  
 The data regarding R can be found in monitor 4. If it is about temperature, the user just needs to change the temperature into the corresponding No.R in setting of analog and temperature to get the data conveniently.  
 Used to read the data of temperature and moisture sensor.  
 Work with all the external registers that are compatible with MODBUS Code 03. It can work with 18 external registers at most. PLC will read automatically without interference the instructions, which must be added to the program. The user just needs

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*



to judge the value of Rxx.

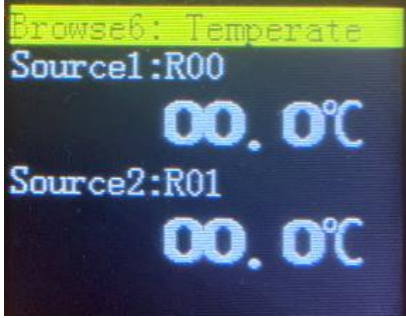
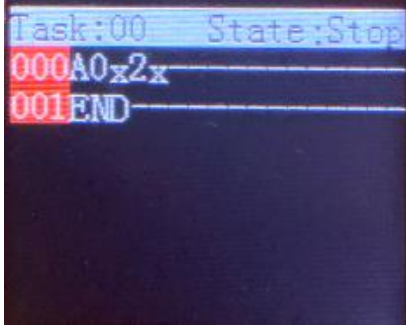
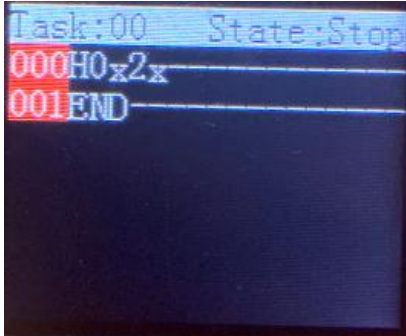
14: NTC temperature checking



Supporting NTC temperature checking of 2 channels.

Checking scope: 0~100 °C (resolution rate of 0.1 °C )

Temperature data can be used directly to program. There is an interface for temperature monitoring, which makes condition judgment more conveniently.

	
<p>15: A AND instruction</p> 	<p>Example: A 0 x 2 x -----</p> <p>The next order is executed when all the following conditions are met, i.e. X0 close, X1 open, X2 close and X3 open.</p> <p>If a skipping order is added after an order such as “A” or “H”, these 3 order s become an instant judgment. That is say, if the conditions are met, it will skip to the target line to execute the order. If the conditions are not met, the program will execute the order from the line after the Skipping order.</p>
<p>16: H OR instruction</p> 	<p>Example: H 0 x 2 x -----</p> <p>The next order is executed when one of the following conditions is met, such as, X0 close, X1 open, X2 close or X3 open.</p>

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*



17: Program control instructions

```
Task:00 State:Stop
000Task00 Paus
001Task00 Stop
002Task00 Run
003Task00 Hold
004Task00 Rst
005END
```

```
Task:18 (ScanGP)Stop
000Task00 Run /Paus
001Task00 Run /Stop
002Task00 Run /Hold
003Task00 Wait
004END
```

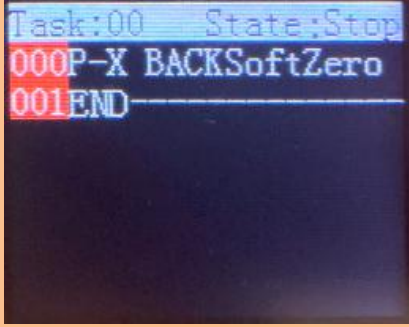
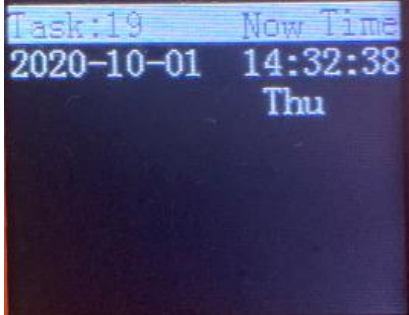
Program 00-- --Pause(when program 00 pauses temporarily, all the related outputs automatically open and all the related timer and time delay pause)  
 Program 00----Stop(when program 00 stops and the related outputs open automatically, the related timers and time delay return to their original states)  
 Program 00----Start( starting executing program 00)  
 Program 00---- Holding( program 00 stop running promptly but all the related outputs are not effected)  
 Program 00----Restart(when program 00 is restarted immediately, all the related output is open and program 00 starts to run from the beginning)  
 Program 00----Start/pause(switching between pause and start)  
 Program 00----Start/stop(switching between start and stop)  
 Program 00---Start/holding(switching between start and holding)  
 Program 00—Wait to stop(waiting until the target program to finish its operation and then stop)

18: Pulse instructions  
 Axle X, Axle Y or Axle Z

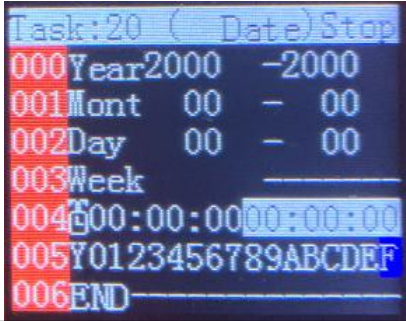
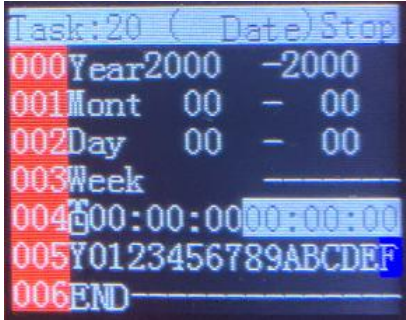
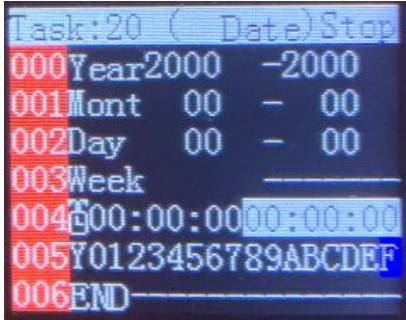
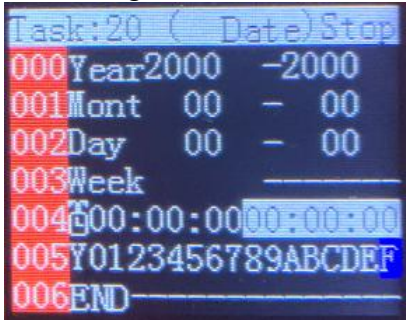
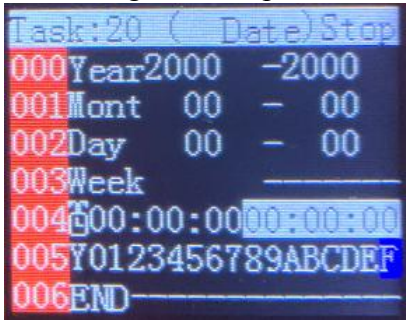
```
Task:00 State:Stop
000P-X Spee020.00KHz
001P-X FWD 0003.50mm
002P-X REV 0005.00mm
003P-X braking
004P-X Waitting
005P-X SetSoftZero
006END
```

Speed 000.00KHz~150.00KHz  
 Set the speed before employing clockwise revolution and anti-clockwise revolution)  
 Clockwise revolution 0000.00 round(mm or piece) (the unit can be selected in the pulse output setting menu)  
 Anti-Clockwise revolution 0000.00 round(mm or piece) (the unit can be selected in the pulse output setting menu)  
 Prompt stop  
 The pulse output can be stopped promptly anytime and surplus pulse is not remembered)  
 Wait to stop  
 The next order is not executed until the present motor is stopped.  
 Setting as the zero point of program  
 Mark the present position as the zero point of the program

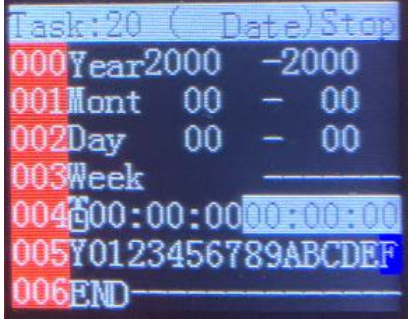
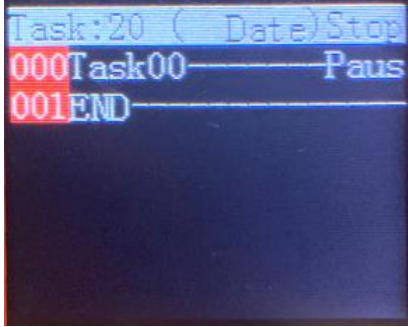
*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

		<p>Return to the zero point Order the motor to return to the zero point</p>
<p>Program18</p>	<p>1: Scan <b>(Simultaneous checking without order)</b></p> <p>2: Output</p> <p>3: Combined output Y</p> <p>4: C counter</p> <p>5: M timer</p> <p>6: TX Instructions</p> <p>7: A And Instructions</p> <p>8: H OR instructions</p> <p>9: Program control instructions</p> <p>10: Speed of Axle X 000.00KHz</p> <p>11: TW=00 channel 000 close</p> <p>12: Encoder 0 clear</p>	<p>Programming same as that of program 00-17 Program state judgment is added. For example, scan program 00 stop It is judged based on the state of program 00(start/ stop)</p> <p>Programming same as that of program 00-17</p>
<p>Program 19</p>	<p>System time verification</p> 	<p>Used to set system time It necessitates to correct the system time in advance if it is used as a timer.</p>

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

<p>Program 20(setting the calendar)</p>	<p>1: Setting the year</p> 	<p>The restrictive conditions of adding year Year 2019-2019 means the year 2019 is specified Example 2: Year 2019-2023 means the period from 2019-2023 is specified.</p>
	<p>2: Setting the month</p> 	<p>The restrictive conditions of adding month Example 1: Month 03-03 means March is specified Example 2 Month 07-08 means the period from July to August is specified.</p>
	<p>3: Setting the date</p> 	<p>The restrictive conditions of adding a day Example 1 Day 03-03 means the 3<sup>rd</sup> day of a month is specified. Example 2 Day 03-08 means the period from the 3<sup>rd</sup> day to the 8<sup>th</sup> day is specified.</p>
	<p>4: Setting the week</p> 	<p>The restrictive conditions of adding a week, Example, - - 2 3 4 5 – means Tuesday, Wednesday, Thursday , Friday...</p>
	<p>5: Setting the time period</p> 	<p>Restrictive conditions of adding time Example 08:30:00 12:00:00 means the period from 8:30 to 12 o'clock proper is specified</p>

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

	<p>6: Combined output Y</p>  <pre> Task:20 ( Date) Stop 000 Year 2000 -2000 001 Mont 00 - 00 002 Day 00 - 00 003 Week 004 00:00:00 00:00:00 005 Y0 123456789ABCDEF 006 END                     </pre>	<p>Refer to program 00--17</p>
	<p>7: Program control instructions</p>  <pre> Task:20 ( Date) Stop 000 Task00 ----- Paus 001 END -----                     </pre>	<p>Refer to program 00--17</p>
<p>Program 21 (quick interface)</p>	<p>1: Txx timer Scope 0-24</p>	<p>T08: 02:02:00:00 Set the value of timer 2 to be 2 hours and 2 minutes</p>
	<p>2: Aval variable shortcut special for analog (scope 0-49)</p>	<p>Aval01 10.00 V/mA Set the value of variable shortcut Aval 01 to be 10.00</p>
	<p>3: Cval variable shortcut special for counter (scope 0-49)</p>	<p>20000000 Set the variable shortcut Cval01 to be 20000000</p>
	<p>4: Sval variable shortcut special for pulse output (scope 0~24)</p>	<p>Sval01 010.00KHz Set the variable shortcut Sval to be 10Khz</p>
	<p>5: Lval variable shortcut special for pulse output (scope 0-24)</p>	<p>Lval01 0010.00mm/round Set the variable shortcut Sval to be 10mm/round</p>
	<p>6: System clock</p>	<p>Display the present system time</p>

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

## Chapter 6 RS485 Communication and Its Touching Screen Links

### 1: The working principle of touching screen links

On the condition that the screens are compatible with 485 port and Modbus RTU protocol, the links are applicable to the screens of any brand.

Principle: touching screen visits the internal registers in the all- in –one relay in order to exchange the data.

The compatible registers are listed as below:

Register Address (decimal)	Description	Read/Write	Compatible with successive reading or editing or not	Compatible function codes
00000	Address of the internal register	R/W	No	03 and 10
00001	Address of input	R	No	03
00002	Ai0 collected current	R	No	03
00003	Ai1 collected current	R	No	03
00004	Ao0 output	R/W	No	03 and 10
00005	Ao1 output	R/W	No	03 and 10
00006	Ai0 collected voltage	R	No	03
00007	Ai1 collected voltage	R	No	03
00008	NTC0 collected temperature	R	No	03
00009	NTC1 collected temperature	R	No	03
00100~00120	Program 00-19 start/stop	R/W	No	03 and 10
00200~00299	Variable shortcut T00~T24 (Inserted in front of program 21 before using)	R/W	Yes	03 and 10
00300~00349	Variable shortcut Aval 00~49 (Inserted in front of program 21 before using)	R/W	Yes	03 and 10
00350~00399	Variable shortcut Cval 00~49 (Inserted in front of program 21 before using)	R/W	Yes	03 and 10
00400~00424	Variable shortcut Sval 00~24 (Inserted in front of program 21 before using)	R/W	Yes	03 and 10

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

00425~00474	Variable shortcut Lval 00~24 (Inserted in front of program 21 before using)	R/W	Yes	03 and 10
10200	Angle 0	R	No	03
10201~10202	Speed 0	R	No	03
10203~10206	Length 0	R	No	03
10207~10210	Pulse quantity 0	R	No	03
10300	Angle 1	R	No	03
10301~10302	Speed 1	R	No	03
10303~10306	Length 1	R	No	03
10307~10310	Pulse quantity 1	R	No	03

**Attention: start or stop the register through program: 00 stop , 01 start , 02 pause , 03 re-start . Lval data are of 32 bytes and occupy two registers of 16 bytes, i.e. 425 and 426. Attention should be paid to the internal update of waiting instructions and scanning instructions, adding distance instructions within 00-18 program and the data update of setting of pulse input/output in the general menu.**

## 2. Superior PC Communication and Extended Communication Examples

Communication explanation of RS485 Modbus-RTU

What follows are the common addresses of internal registers:

Read or edit with the codes 03, 06 and 10 in the Modbus-RTU

Baud rate and verification can be modified in the communication setting in the general menu of simple PLC. Open 485 communication before using 485 communication. The address of the PLC refers to the physical address of the MODBUS( the address is exclusive and can not be repeated by other equipment.

---

### Communication adjustment of serial ports of superior PC

Example 1

Read the coil state of address 00000 through code 03 and send the message **00 03 00 00 00 01 85 DB**

Analysis:

00=PLC hardware address

03= operation code

00 00= initial address of register

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

00 01 register quantity that can be read successively

85 DB=CRC Verification

Returning code: 00 03 02 00 01 44 44

Analysis:

00=PLC hardware address

03=operation code

02= valid byte quantity

00 01=16 inputs state(Y00 close, y01-15 open)

44 44=CRC verification

---

Example 2:

Read the coil state of address 00001 through code 03 and send the message 00 03 00 01 00 01 D4 1B

Analysis:

00=PLC hardware address

03= Operation code

00 01= Initial address

00 01 register quantity that can be read successively

D4 1B=CRC verification

Returning code: 00 03 02 00 02 04 45

Analysis:

00=PLC hardware address

03= operation code

02= valid bytes

00 02=16 inputs state(X01 triggered and others untriggered)

04 45=CRC verification

---

Example 3

Operate output coil by editing the address 00000 through the code 06 and send the message 00 06  
00 00 00 03 C8 1A

Analysis:

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

00=PLC hardware address

06=operation code

00 00= initial address of registers

00 03=edit the value of register

C8 1A=CRC verification

Returning code: 00 06 00 00 00 03 C8 1A

Analysis:

00=PLC hardware address

06=operation code

00 00=initial address of register

00 03=current value of register

C8 1A=CRC verification

Final state: Y00 Close, Y01 Close Y02-Y15 Open

#### Example 4

Start simultaneously the program 00 and 01 by editing the addresses 00100 and 00101 by operation code 01 and send 00 10 00 64 00 02 04 00 01 00 01 60 88

Analysis:

00=PLC hardware address

10=operation code

00 64=initial address of register

00 02=register quantity that can be operated successively

04= quantity of the related bytes(quantity of register \*2)

00 01= the value of the first register( representing the starting program 00)

00 01=the value of the 2<sup>nd</sup> register(representing the starting program 01)

60 88=CRC verification

Returning code: 00 10 00 64 00 02 01 C6

Analysis:

00=PLC hardware address

10=operation code

00 64=initial address of register

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*



00 02=quantity of registers

01 C6=CRC verification

Final state: start program 00 and 01 at the same time.

Example 5:

Read the counter C00 through code 03 and send the message **00 03 27 10 00 02 CE AB**

Hint: two bytes are to be read successively since each counter occupies the saving space of 2 bytes.

Analysis:

00=PLC hardware address

03=operation code

27 10=initial address(decimal 10000)

00 02 the quantity of registers that are read successively

CE AB=CRC verification

Returning codes: **00 03 04 00 00 00 C5 2A A0**

Analysis:

00=PLC hardware address

03=operation code

04=valid bytes

00 00=high 16 bits data

00 C5=low 16 bits data

2A A0=CRC verification

Final state: =197 (decimal)

Example 6: assign the counter C00 through code 10(used to clear or change arbitrary value) and send the code **00 10 27 10 00 02 04 00 01 86 A0 B1 F0** to change the value of counter C00 into 100000. **100000(decimal)= 00 01 86 A0(hexadecimal)**

Hint: two bytes are to be read successively since each counter occupies the saving space of 2 bytes.

Analysis:

00=PLC hardware address

10=operation code

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

27 10=the initial address of register

00 02=the quantity of registers that needs to be operated successively

04= the related byte quantity(quantity of register\*2)

00 01=the value of the 1<sup>st</sup> register (pay attention to the high-low byte format and sequential position)

86 A0=the value of the 2<sup>nd</sup> register(pay attention to the high-low byte format and sequential position)

B1 F0=CRC verification

**Returning code: 00 10 27 10 00 02 4B 68**

Analysis:

00=PLC hardware address

10=operation code

27 10=the initial address of register

00 02=the quantity of register

4B 68=CRC verification

Final result: change the value of counter C00 into 100000

Example 7:

Change the value of timer T00 through code 10 in the aim of modifying the value of time delay online.

Hint: each timer consists of four types of data, i.e. hour, minute, second and millisecond and occupies four registers accordingly. For example, the addresses in the case of T00 are 200 201 202 203 respectively.

Send the message 00 10 00 CA 00 01 02 00 0A 3A 6D to change the internal value of second of the timer T00 into 10 seconds.

Analysis:

00=PLC hardware address

10=operation code

00 CA=the initial address of register

00 01=the quantity of registers that needs to be operated successively

02= the related byte quantity (the quantity of register \*2)

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

00 0A=target value

3A 6D=CRC verification

Returning code: 00 10 00 CA 00 01 20 26

Analysis:

00=PLC hardware address

10=operation code

00 CA=the initial address of register

00 01=the quantity of registers

20 26=CRC verification

### 3: Operation manual for computer software

#### Software environment

The software can only work with Windows 7 system or other superior windows systems at present

#### Installation of software

The software is free of installation. The software can run with double click “QingJunPLC.exe” after it is decompressed.

#### Software functions

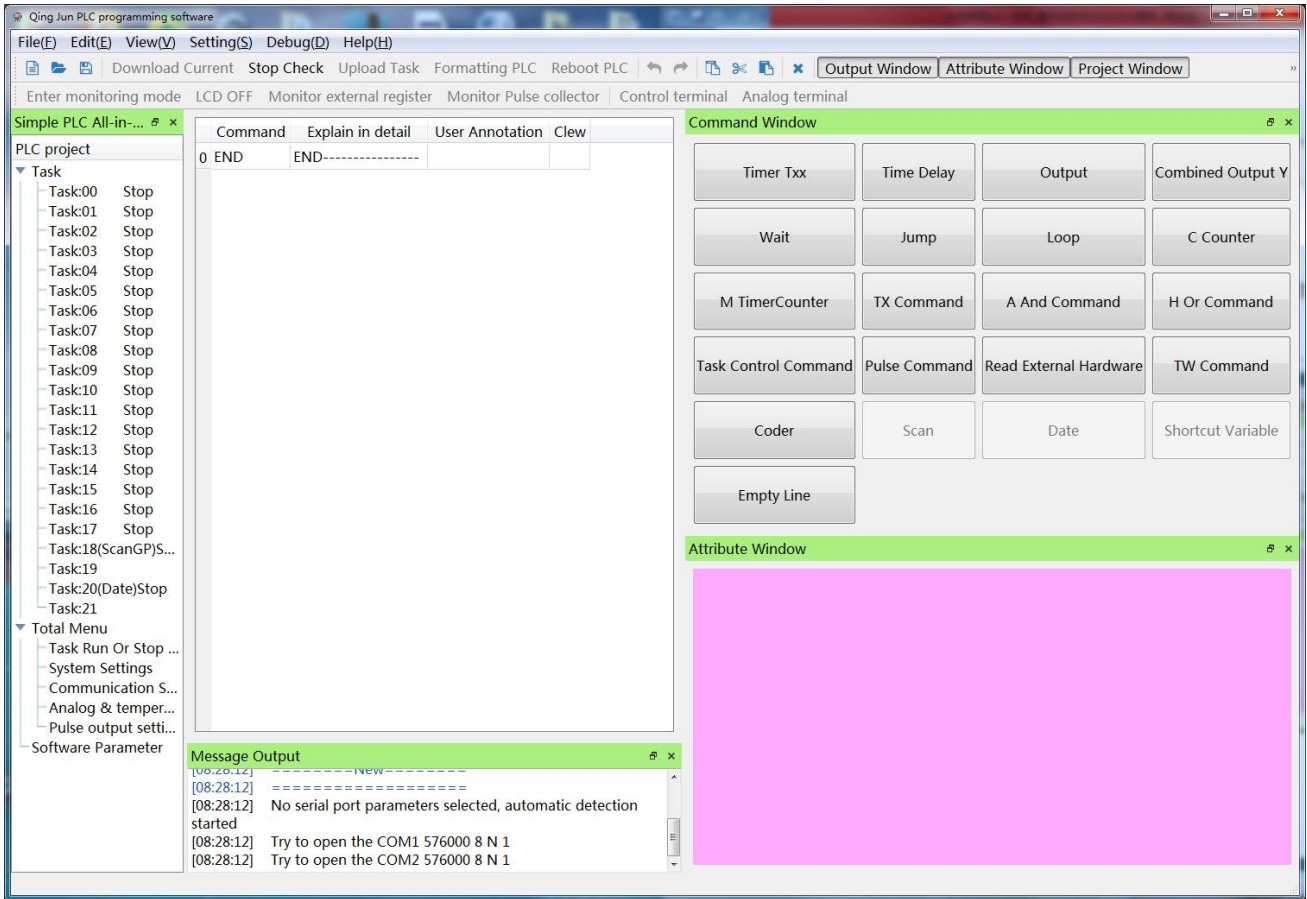
Support the upload and download between the software and PLC

Support the real time monitor of PLC working state

Support the opening and saving of software

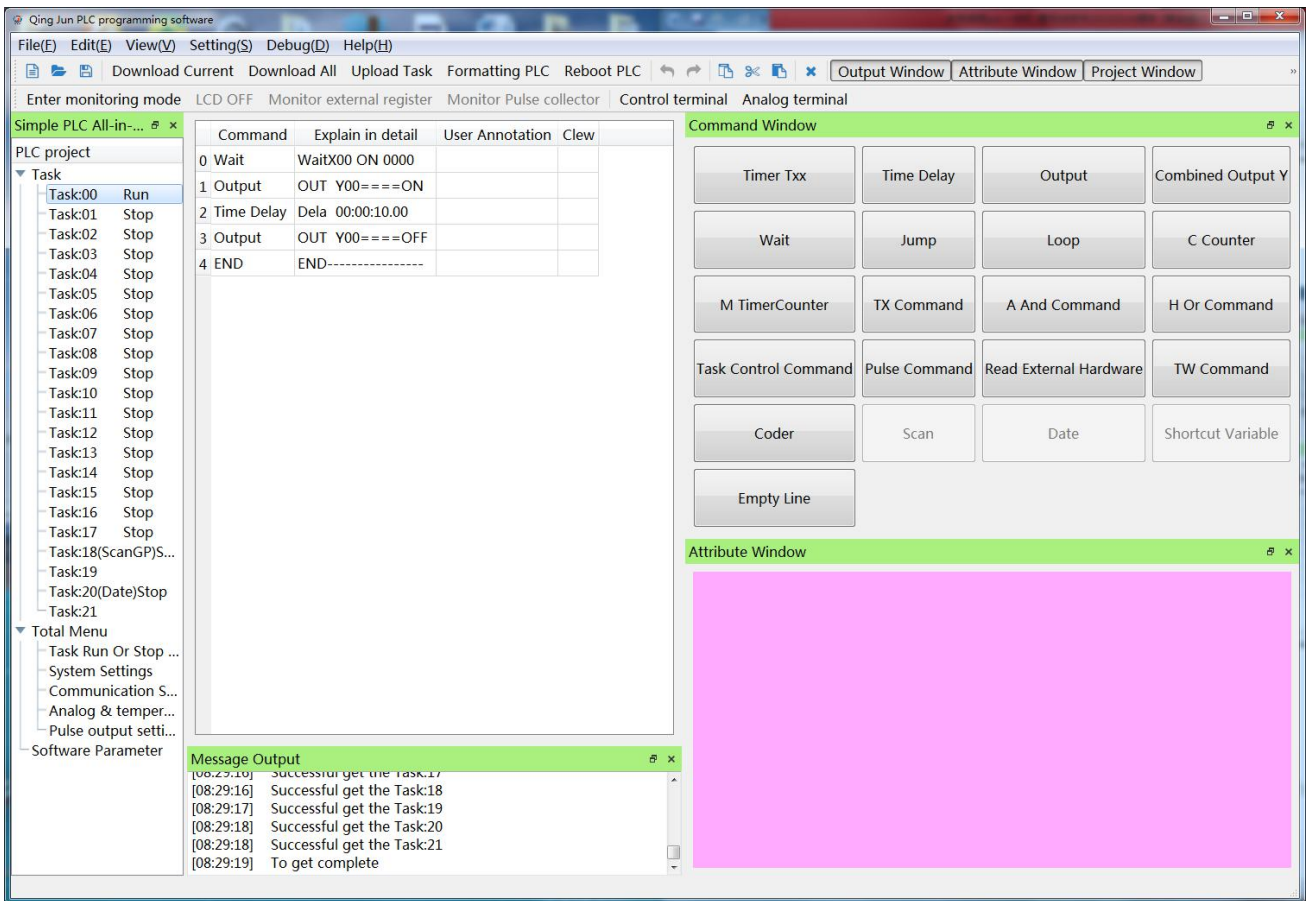
*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

**Software application**



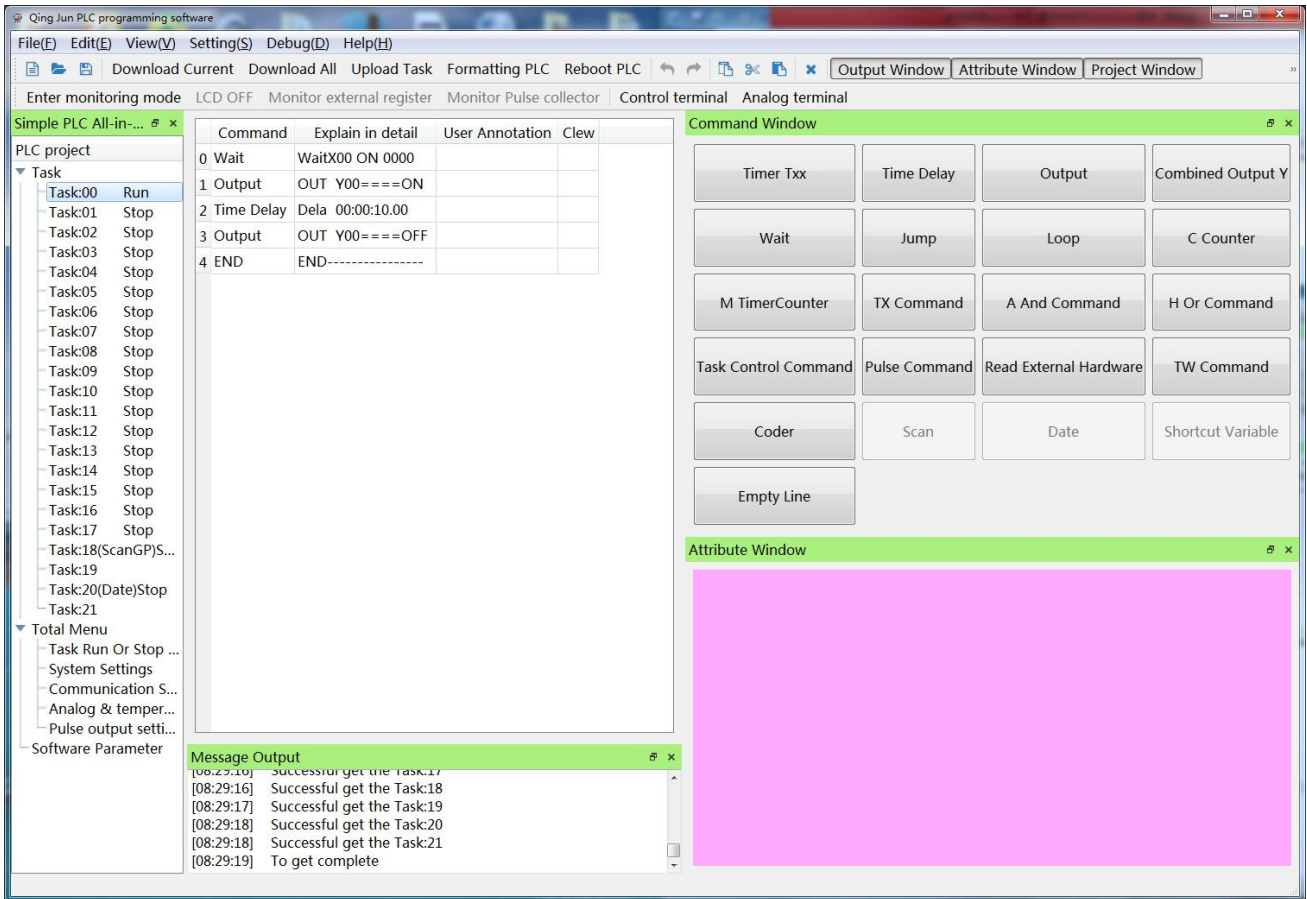
**Click the software to obtain the PLC program automatically**

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*



**Software obtaining over**

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*



**Program copy**

**(Click copy all and successful verification time will be displayed)**

**Programming method: what the user needs to do is just key in the order to be executed in the instruction window and modify the parameters in the attribute window.**

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

## Chapter 7: Programming Exemplification

### 1. Cases of common programming, examples are given to explain how to use the following instructions

(delay, output, circling, waiting, scanning, counting, keying, analog input and output, calendar)

#### 1) 、 Cases of delay instruction control

Program: 00	Running	Explanation:
000 output Y00== close	004 output Y00== open	the registers Y00-Y01 500mS close one by one and then open one by one <b>Default: infinite circling</b> <b>Program 00 just works once if line 012 is inserted</b>
001 delay 00:00:00.50	005 delay 00:00:00.50	
002 output Y01== close	006 output Y01== open	
003 delay 00:00:00.50	007 delay 00:00:00.50	
	<b>008 program 00-----stop</b>	

#### 2) 、 Cases of timer T in lieu of time delay control

Program: 00	Running	Explanation
000 output Y00== close	004 output Y00==open	the registers Y00-Y01 500mS close one by one and then open one by one the value of T00 can be modified in the shortcut interface
001 T00 00:00:00.50	005 T00 00:00:00.50	
002 output Y01==close	006 output Y01==open	
003 T00 00:00:00.50	007 T00 00:00:00.50	

#### 3) 、 Cases of judgment based on Counter C

Program: 00	Running	Program: 18(with Scanner) running
000 C00=00000000	006 delay 00:00:00.50	000 scan C00> <b>00000030</b> 001 program00 ----- stop
001 output Y00==close	007 output Y01== close	
002 time -delay 00:00:00.50	008 delay 00:00:00.50	
003 output Y01==close	009 C00 + 00000001	
004time--delay 00:00:00.50	010 circling 001 line 0000 times	
005 output Y00==open		
<p><b>Explanation: the counter re-sets automatically when it is powered on. Y00 and Y01 close one after another and then open one by one. When the circle is over, counter C00 plus 1 automatically.</b></p> <p><b>Program 18 keeps checking the value of Counter C00 and the program 00 stops when the value is greater than 30.</b></p>		

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

**4) 、 Cases of Timer M**

Program: 00 stop	Program : 18( with scanner) running	Explanation: X00 starting switch X01, X02 limit switch Y00 ,Y01 positive and negative control ports of motor
000 output Y00===close 001 outputY01===open 002 M00 clear 003 M00 start 004wait X01close 0.00 005 M00 pause 006 M00 clear 007 output Y00=== open 008 outputY01=== open 009 delay 00:00:01:00 010 output Y00=== open 011 output Y01=== close 012 M01 clear 013 M01 start 014 wait X02 close 0.00 015 M01 pause 016M01 clear 017 output Y00=== open 018output Y01=== open 019program 00 -----stop	000 scan X00===close 001 program 00-----start 002 output Y03===== open  003 scan M00>00:00:01:30 004 M00 clear 005 program 00 -----stop 006 output Y03===== close  007 scan M01>00:00:01:00 008 M01 clear 009 program 00 ----- stop 010 outputY03===== close	

**5) 、 Program 00 cases of key testing(sequential testing of input ports)**

Program: 00 running	Explanation: The next order is executed only when X00 is Close.
000 wait X00==close 001 output Y00== close 002 output Y01== close	

**6) 、 Program 18 cases with scanner(parallel testing of input ports)**

Program: 18( with scanner) running	Explanation: Test the states of input ports X00, X01 and X02 simultaneously If X00 closes, Y00 and Y01 Close If X01 and X02 close, Y02 close(realization and function) If X00, X01 and X02 close simultaneously, Y00, Y01 and Y02 close at the same time.
000 scan X00== close 001 output Y00== close 002 output Y01== close  003 scan X01== close 004 scan X02== close 005 output Y02== close	

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*



**7) 、 Scan 3 keys to control 3 corresponding programs in lieu of 3 time relays**

Program : 00 stop 000 output Y00==close 001 delay 00:00:01.00 002 output Y00==open 003delay 00:00:01.00 004 program 00----stop	Program : 01 stop 000output Y01== close 001 delay 00:00:01.00 002 output Y01== open 003 delay 00:00:01.00 004 program 01---- stop	Program : 02 stop 000 output Y02== close 001 delay 00:00:01.00 002 output Y02== open 003 delay 00:00:01.00 004 program 02---- stop	Program:18(with scanner) run 000 scan X00== close 001 program 00---- run 002 scan X01== close 003 program 01----- run 004 scan X02== close 005program 02----- run
---	--	---	--

Explanation: scan three keys X00 X01 X02 simultaneously,  
 program 00 works if X00 closes  
 Program 01 works if X01 closes  
 Program 02 works if X02 closes

**8) 、 Cases of sensors with cylinder and limits of “start, pause ans stop”**

Program: 00 stop 000 output Y00== close 001 wait X03 close <b>0.02</b> 002 output Y01== close 003 wait X04 close <b>0.02</b> 004 output Y02== close 005 wait X05 close <b>0.02</b>	006 output Y00== open 007 output Y01== open 008 output Y02== open 009 C00 +00000001 010 program 00--- stop	Program : 18 (with scanner) start 000 scan X00==close 001 program 00---- run 002 scan X01== close 003 program 00----- pause 004 scan X02== close 005 program 00-----stop
--	--	---

Explanation :  
 Press X00 to start program 00. When cylinder Y00 moves to the limit X3, cylinder Y01 starts to work. When cylinder moves to the limit X4, cylinder Y02 starts to work. When cylinder Y03 moves to the limit X5, all the three cylinders close at the same time and the value of counter 00 added 1 automatically.  
 The program suspends if press the key pause midway. It will continue to work if press the key again.  
 The program stops if press the key stop midway. If it is restarted, program 00 will reset and Y00 Y01 Y02 open automatically and start to work until the program is restarted next time.  
 0.02 refers to the time to remove vibration, i.e. 20mS

**9) 、 Cooperation of several paths**

Program 00 Run	
000 output Y00==close 001 delay 00:00:05.00 002 output Y02==close 003 delay 00:00:03.00 004 output Y02==open	005 output Y04==close 006 delay 00:00:02.00 007 output Y00==open 008 output Y04==open 009 delay 00:00:01.00

Explanation: several paths work together within a circle.  
 Y00 opens after it closes for 10 seconds. Y02 opens after it closes for 3 seconds and Y04 closes when Y00 closes for 5 seconds. 2 seconds later, Y00 and Y04 open at the same time and repeat the above movements infinitely .

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

**10) 、 Cases of inset circles.**

Program: 00 run		
000 output Y00== close 001 delay 00:00:01.00 002 output Y00==open 003 delay 00:00:01.00 004 circle 000 line 0005 times	005 output Y01== close 006 delay 00:00:01.00 007 output Y01==open 008 delay 00:00:01.00 009 circle 000 line 0005 times 010 program 00----- stop	Explanations: Y01 closes and opens once when Y00 closes and opens 5 times. It stops when the above circle repeats 5 times.

**11) 、 Cases of imitating key clicking( realize the function of opening and closing with clicking the key once)**

Program: 00 stop	Program : 01 start	Explanations: Press X00 and wait for 20mS to remove the vibration. Program 00 starts to work if vibration removal is successful. Otherwise a testing must be done again. After the key is released, the program continues to check the pressing of X00 next time, which triggers program00 stop. The circle repeats this way.
000 output Y00== close 001 delay 00:00:01:00 002 output Y00== close 003 delay 00:00:01:00	000 wait X00 close 0.02 001 program 00--- start 002 wait X00 open 0.02 003 wait X00 close 0.02 004 program 00---stop 005 wait X00 open 0.02	

**12) 、 Cases of Input signal triggering of edge**

The motor reverses when it turns to the preceding limit and it stops when it reaches the back limit.

Starting key X00 frontal limit X01 back limit X02

Program: 00 stop	Program : 18 (with scanner) start
000 output Y00==close 001 output Y01== open 002 wait X01 rising edge 0.00 003 output Y00== open 004 output Y01== open 005 delay 00:00:01:00	006 output Y00==open 007 output Y01== close 008 wait X02 rising edge 0.00 009 output Y00== open 010 output Y01==open 011 program 00----- stop
<p>Explanations: Press X00, and the program starts to work. The motor keeps turning until it reaches the limit X01 and it stops. The motor reverses 1 second later and it keeps turning until it reaches the limit X02 and then it stops. Program 00 stops at that time and wait the triggering start next time.</p>	

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

**13) 、 It starts to work when there is an input signal and it stops when there is no input signal.**

Program 00: stop	Program18 : (with scanner) start	Explanations: Program 00 starts when X00 closes Program 00 stops when X00 opens. Output is related to the program start and open states intelligently.
000 output Y00== close 001 delay 00:00:01.0 002 output Y00==open 003 output Y01== close 004 delay 00:00:01.0 005 output Y01== open 006 program 00----- stop	000 scan X00=== close 001 program 00----- start 002 scan X00=== open 003 program 00-----stop	

**14) 、 Testing of analog input common testing**

**The units of the input and output must be set in the general menu in advance**

Program : 00 stop	Program 18:(with scanner) start	Explanations: Continuous testing If the input current value is greater than 10.00mA, program 00 starts. Otherwise program 00 stops.
000 output Y00== close 001 delay 00:00:01.0 002 output Y00== open 003 delay 00:00:01.0	000 scan Ai0 > 10.00mA 001program 00----- start 002 scan Ai0 < 10.00mA 003program 00----- stop	

**15) 、 Testing of analog input judgment of scope**

**The units of the input and output must be set in the general menu in advance**

Program: 00 stop	Program 18:( with scanner) start	Explanations: Continuously scan the current value of input channel Ai0 Program 00 starts if the conditions conforms to 5.00mA~10.00mA
000 output Y00== close 001 delay 00:00:01.0 002 output Y00== open 003 delay 00:00:01.0	000 scan Ai0 = 05.00mA ~ 10.00mA 001 program 00===== start 002 scan Ai0 > 10.00mA 003 program 00=====stop 004 scan Ai0 < 05.00mA 005 program 00=====stop	

**16) 、 Cases of analog output**

**The units of the input and output must be set in the general menu in advance**

Program18: (with scanner) start	Explanations: Press X00, the Ao0 output current is 4.00mA Press X01, the Ao0 output current is 8.50mA Press X02, the Ao1 voltage is 5V Press X03, Ao0 and Ao1 are close
000 scan X00===== close 001output Ao0= 04.00mA  002 scan X01===== close 003 output Ao0= 08.50mA  004 scan X02===== close 005 output Ao1= =05.00V  006 scan X03===== close 007 output Ao0= 00.00mA 008 output Ao1= =00.00V	

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

**17) 、 Cases of external relay modules controlled by 485**

**485 parameters are to be set in the general menu in advance**

Program 18 : ( with scanner) start	Explanations: Press X00 and send an order in the format of MODBUS RTU code 06 to operate the external register 00000 and change it to 00001(coil 1 close)
000 scan X00=====rising edge 001 TX=08 R 00000 V 00001	In the same way, press X01 to change the external register 00000 to 00002(coil 2 close)
002 scan X01=====rising edge 003 TX=08 R 00000 V 00002	
004 scan X02=====rising edge 005 TX=08 R 00000 V 00004	Press X02 to change the external register 00000 to 00004(coil 3 close)

**18) 、 Setting the calendar**

(hint: just add what you want)

**Case 1**

Program 20: ( setting calendar) start	Explanation:
000 setting the week: 0 - - - - - 6 001 Y0 - - - - - - - - - -	In the cases of Sunday or Saturday, Y00 is close.

**Case 2**

Program 20: ( setting calendar ) start	Explanations:
000 setting the week: 0 - - - - - 6 001 setting the month : <b>February -- March</b> 002 Y0 - - - - - - - - - -	Y00 is close in the cases of Sundays or Saturdays in February or March.

**Case 3**

Program 20 : ( setting the calendar ) start	Explanations:
000 setting the week: 0 - - - - - 6 001 setting the month : <b>February -- March</b> 002 setting the year : <b>Year 2019- year 2020</b> 003 Y0 - - - - - - - - - -	Y00 is close on the Sundays or Saturdays in February or March from 2019 to 2020.

**Case 4**

Program 20: (setting the calendar) start	Explanations:
000 setting the week: 0 - - - - - 6 001 setting the month : <b>February -- March</b> 002 setting the year : <b>Year 2019-year 2020</b> 003 setting the hour: 00:23:00 00:06:30 004 Y0 - - - - - - - - - - 005 program -----start	Y00 is close and program 00 works when it is from 23:30 to 6:30 am on Sundays or Saturdays in February or March of year 2019 and 2020.

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

**19) 、 Setting the calendar in order to realize different actions under different conditions.**

Program 20: (setting the calendar) start	Explanations: Y00 closes and program 00 starts when it comes to the period from 23:00 to 6:30 am on the day from 2 <sup>nd</sup> to 6 <sup>th</sup> every month.
000 setting the date: 2 <sup>nd</sup> to 6th	
001 setting the hour: 00:23:00 00:06:30	
002 Y0 -----	
003 program 00 ----- start	
004 setting the date: 2 <sup>nd</sup> to 6th	
005 setting the hour: 00:08:00 00:09:30	
006 Y-- 2 -----	
007 program 01 ----- start	

**2: Cases of accessing to external registers through RS485**

**20) 、 The case is based on the realization of 485 Modbus-RTU protocol**

Attentions: simple PLC all-in-one with 485 modules are the prerequisite condition.

The following parameters are to be set in advance in Setting of 485 communication in the general menu if 485 port is used for the first time.

485 communication (open)

Baud rate( select the proper baud rate according to the external equipment)

Data format(select the proper format of data according to the external equipment)

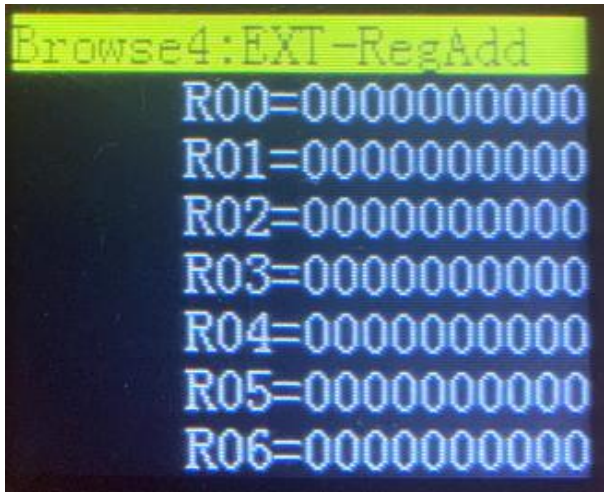
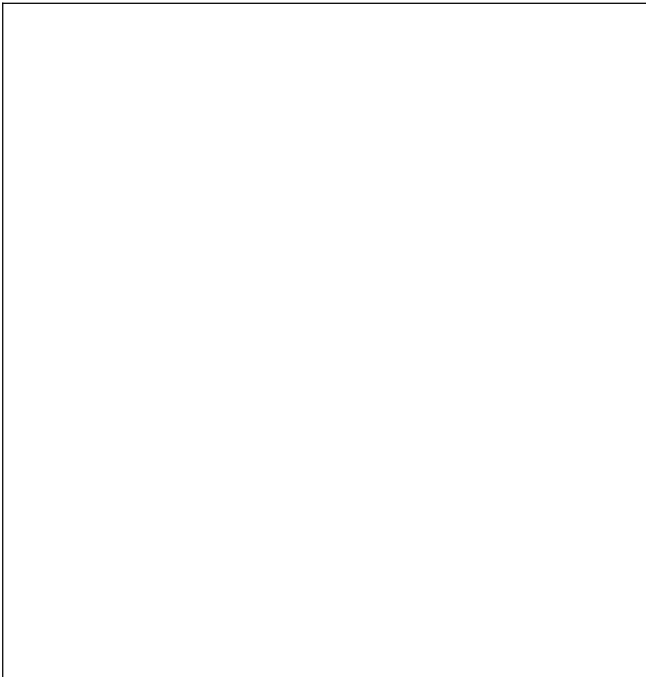
Address (the modbus address of the module can not be the same as others)

Touching screen: if you want to connect the device to the touching screen or you want to enhance the speed of communication, set the mode to the open state.

Driving interval: the time interval of reading the external communication and it is commonly set to be 500mS

Program 00: start	Analysis
000 read external hardware 3H--->register 2710H---> length of 2 bytes is mapped: R00	Powered on, PLC starts to run program 00 automatically. It executes the order of line 000 and checks the equipment of external address 03H and reads the register at the address of 2710H and read 2 successive bytes and feedback the result to R00 through 485 port. If R00 is greater than 1, output 002 is executed and Y0 is close.
001 wait: R00>000000000001	
002 output Y00 ----- close	

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*



Hints: the collected data from an external register can be monitored in the monitor 4. The result is displayed on the register 00 if program 00 reads the external register. It is true of other cases.

**3: Cases of pulse output of high speed**

**21) 、 Inching control of the positive and negative rotation of stepper motors**

Program 00: start	Program 01: start
000 wait X02 close 0.02	000 wait X03 close 0.02
001 axle X speed 010.00KHz	001 axle X speed 010.00KHz
002 axle X rotation infinite round	002 axle X reversal infinite round
003 wait X02 open 0.02	003 wait X03 open 0.02
004 axle X stop promptly	004 X axle stop promptly
005 end-----	005 end-----

Explanations:  
 Program 00 takes responsibility to make axle X turn clockwise and program 01 its reversal.  
 Just take program 00 for example to explain. When X02 closes for more than 20mS, set the speed of X axle motor to be 10KHz( the higher the frequency, the faster the speed)  
 The motor starts to turn clockwise (turn length is infinite until X02 is released. The motor stops immediately X02 is released.  
 It is true of the case when X axle reverses.

**22) 、 Control the guide screw of X motor to complete the following tasks.**

- 1, Press Start key to make the axle X to rotate 100mm at the speed of 10KHz and reverses 50mm with an interval of 0.5 second delay respectively. It returns to zero point automatically with 3 circles.
- 2, Press the emergency key to stop the motor and returns to the zero point at the speed of 5KHz and wait for restart next time.

Attentions: since mm is used as the unit of length, before the program is made, it would be better to

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

set the following parameters once for all in setting of pulse output of the general menu, such as common unit, subdivision coefficient of driver, round distance, and etc.

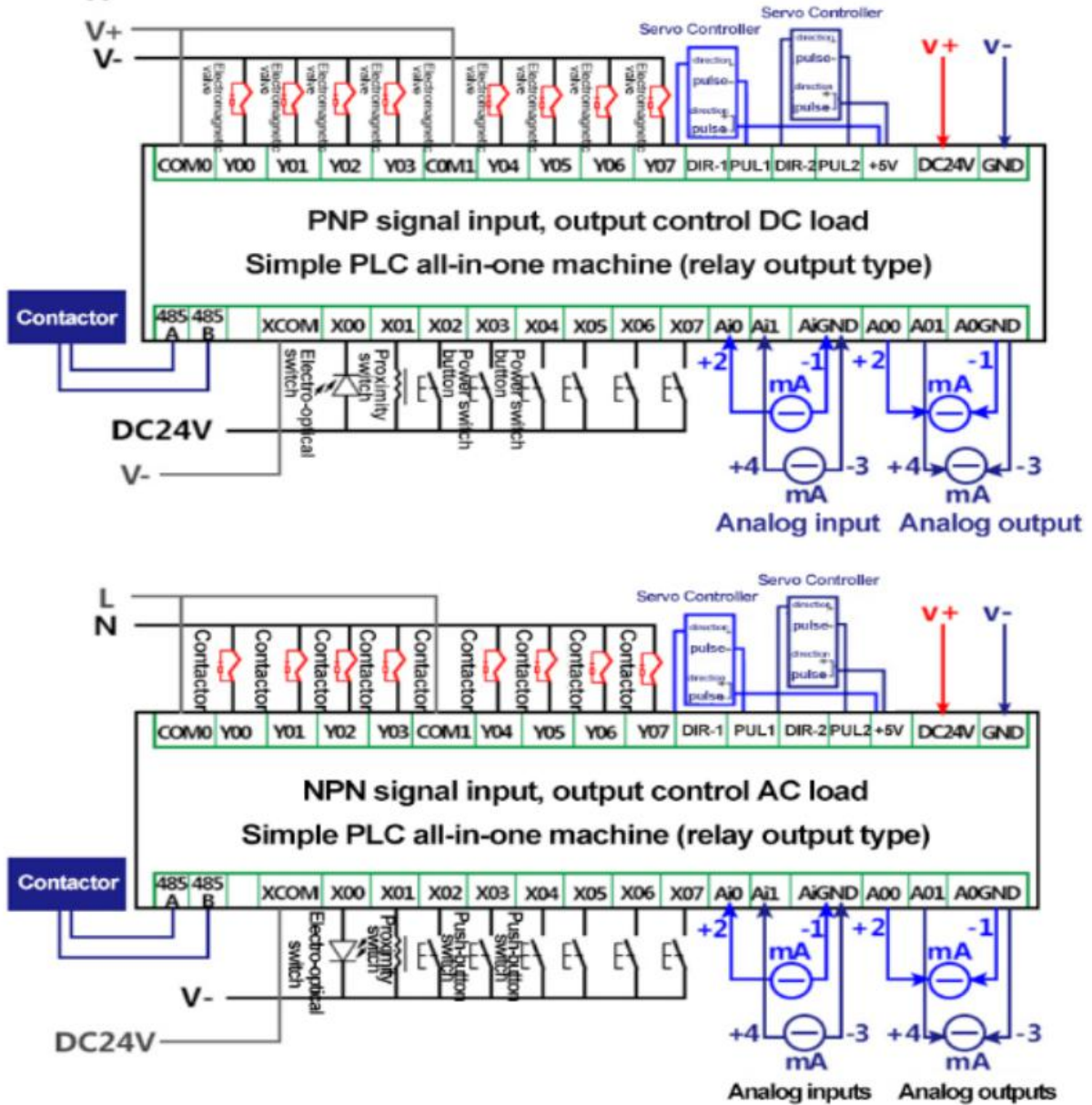
Program 00: start	Program 01: start
000 wait X01 close 0.00	000 wait X02 close 0.00
001 axle X speed 010.00KHz	001 program 00----- stop
002 axle X rotate 0100.00mm	002 axle X prompt stop
003 axle X waiting for stop	003 axle X speed 005.00KHz
004 delay 00:00:00.50	004 axle X reverse infinite mm
005 axle X reverse 0050.00mm	005 wait X00 close 0.00
006 axle X waiting for stop	006 X axle prompt stop
007 delay 00:00:00.50	007 program 00----- start
008 circle 002 line 0003 times	008 end-----
009 axle X reverse infinite mm	
010 wait X00 close 0.00	
011 axle X prompt stop	
012 end -----	
<p>Explanations:</p> <p>Mechanical zero point sensor connected to X00</p> <p>Start key X01</p> <p>Emergent stop key X02</p> <p>Program 00 takes responsibilities of normal processing.</p> <p>Program 01 takes responsibilities of emergent cases.</p> <p>Analysis of program 00</p> <p>Start program 00 and keep it run until it detects X01 is close and assign the motor speed to be 10KHz. Order the motor to rotate 100 mm until it stops. Order it to reverse 50mm after 0.5 second delay. Skip to 002 line with 0.5 second delay to recircle the above process 3 times. The motor finally stops when the motor reverse to the limit switch.</p> <p>Analysis of program 01</p> <p>Start the machine to run program 01, looking for X02. Stop program 00 if X02 is close. Stop the motor and set its speed to be 5 KHz and order the motor to reverse for infinite length. The motor stops when limit switch X00 closes. Restart program 00 for next task.</p>	

**Please get access to website for more video explanations or contact the custom service for video links.**

*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

## Chapter 8 Wiring Diagram

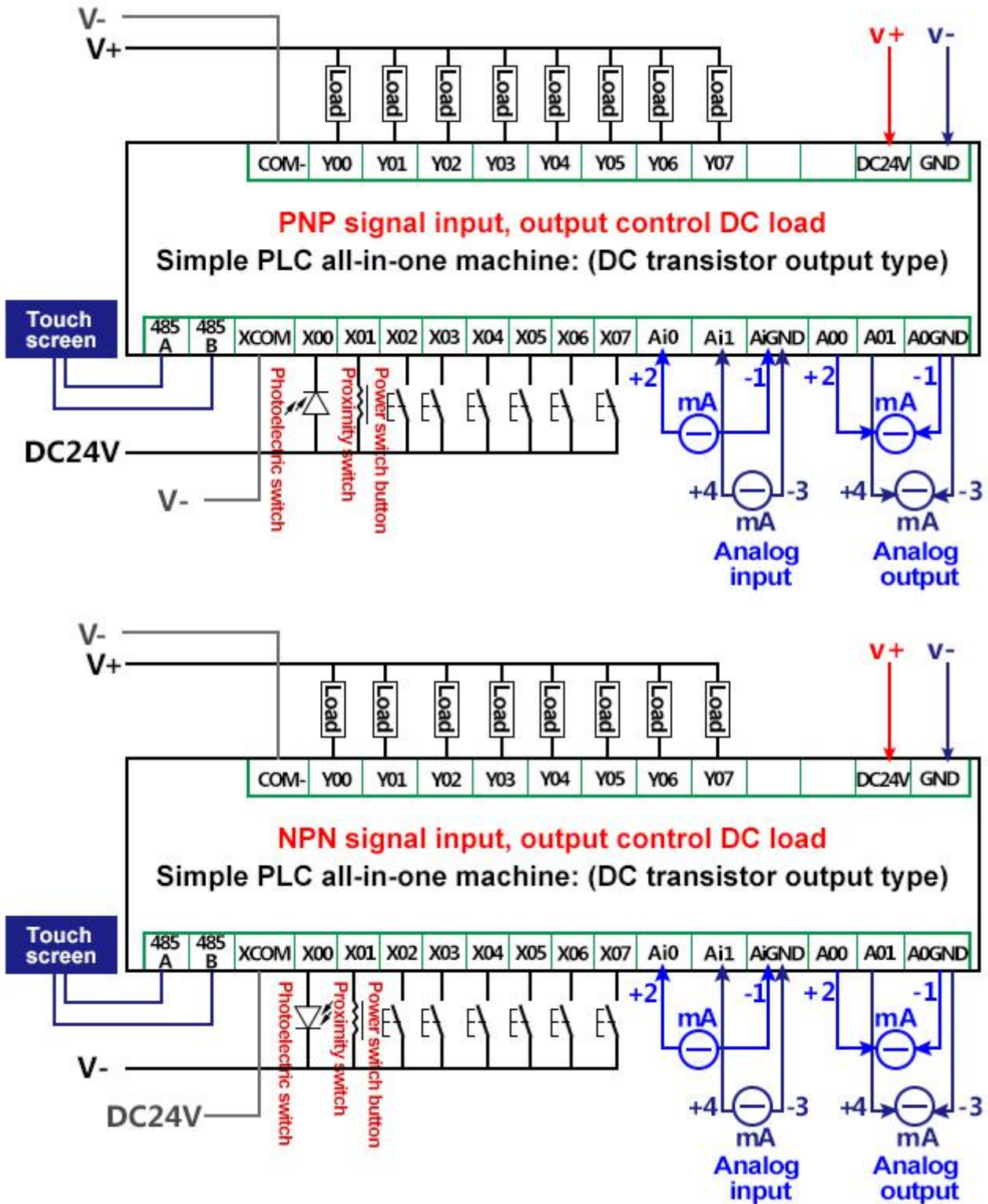
Wiring diagram of simple PLC all-in-one modules ( relay output) 8-in and 8-out.



We just provide consult and suggestions to users about the programming and we do not provide free programming service.

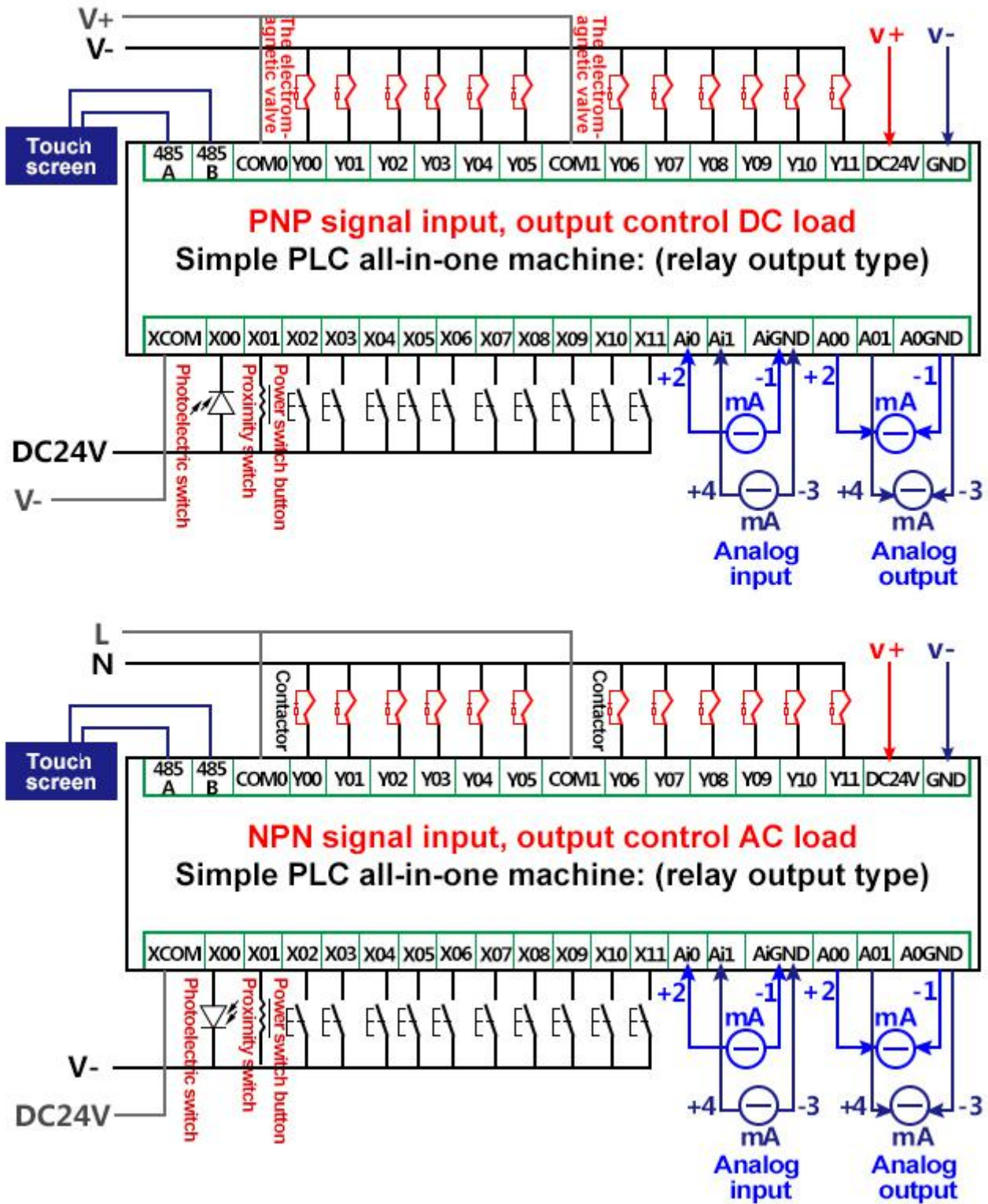


**Wiring diagram of simple PLC all-in-one modules(DC transistor output)  
8-in and 8-out**



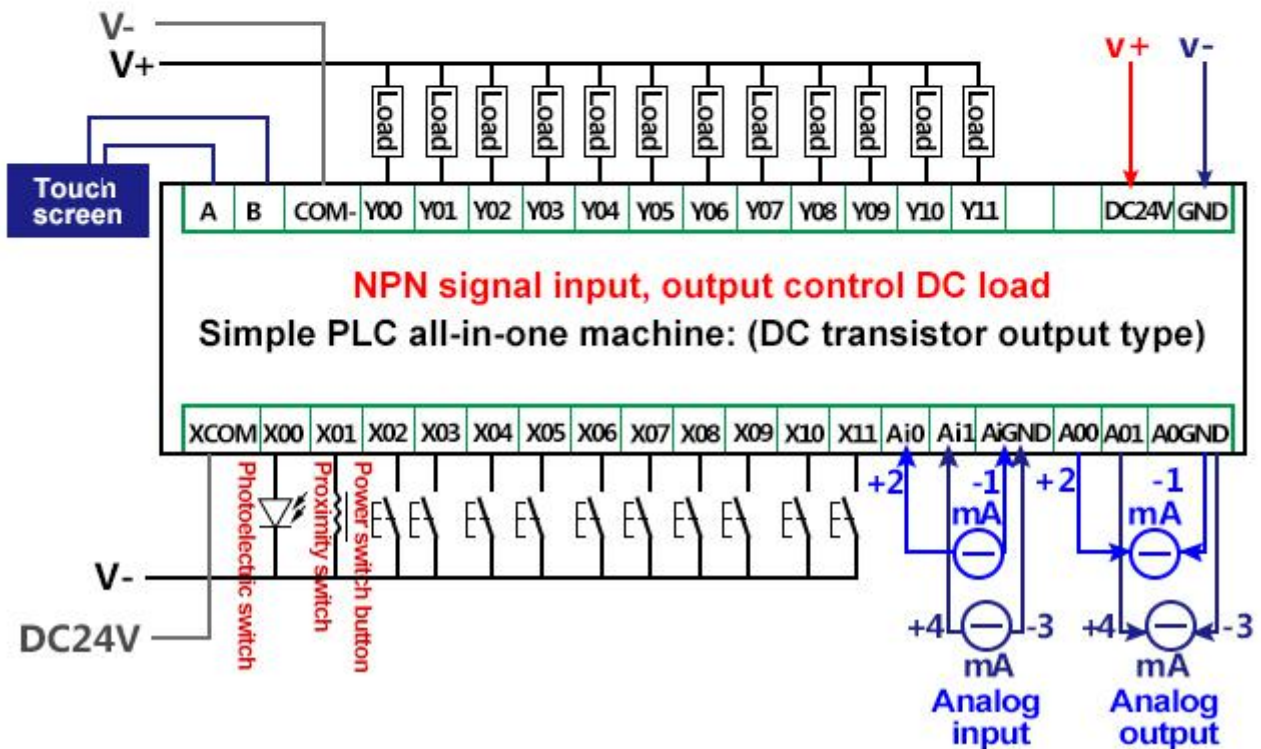
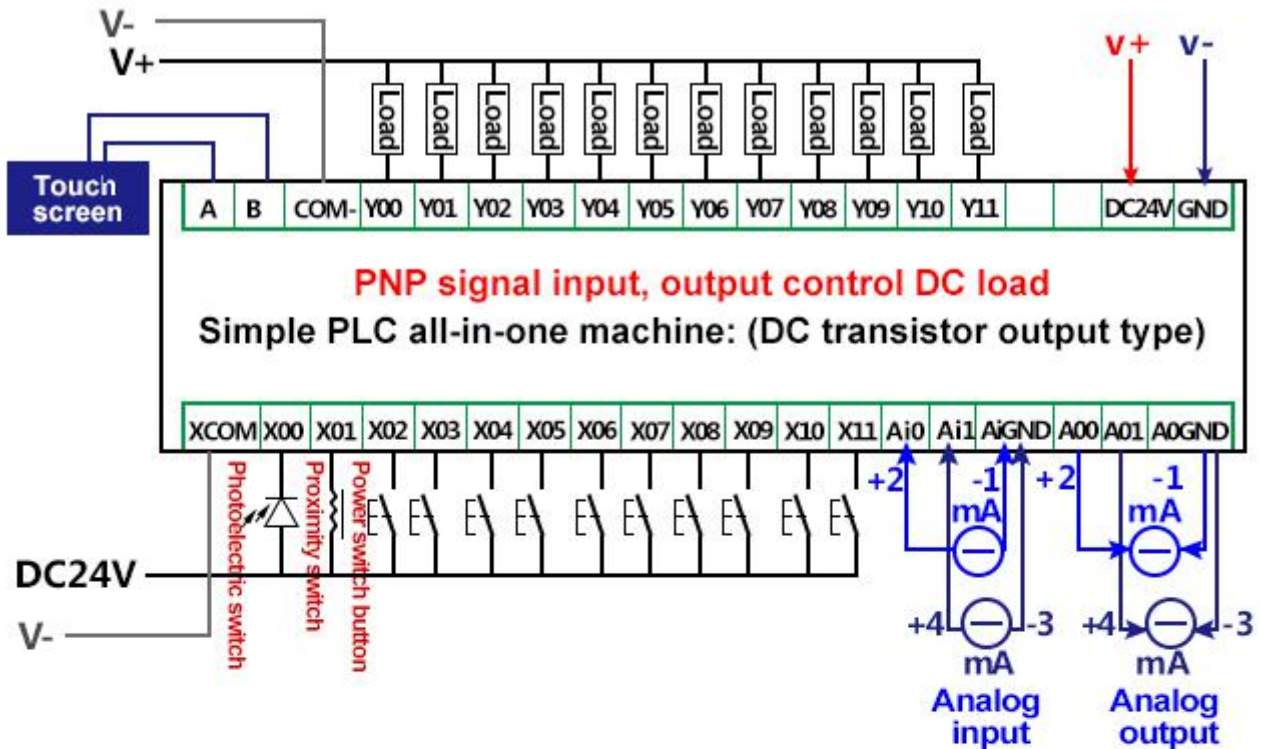
*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*

### Wiring diagram of simple PLC all- in-one modules ( relay output) 12-in and 12-out



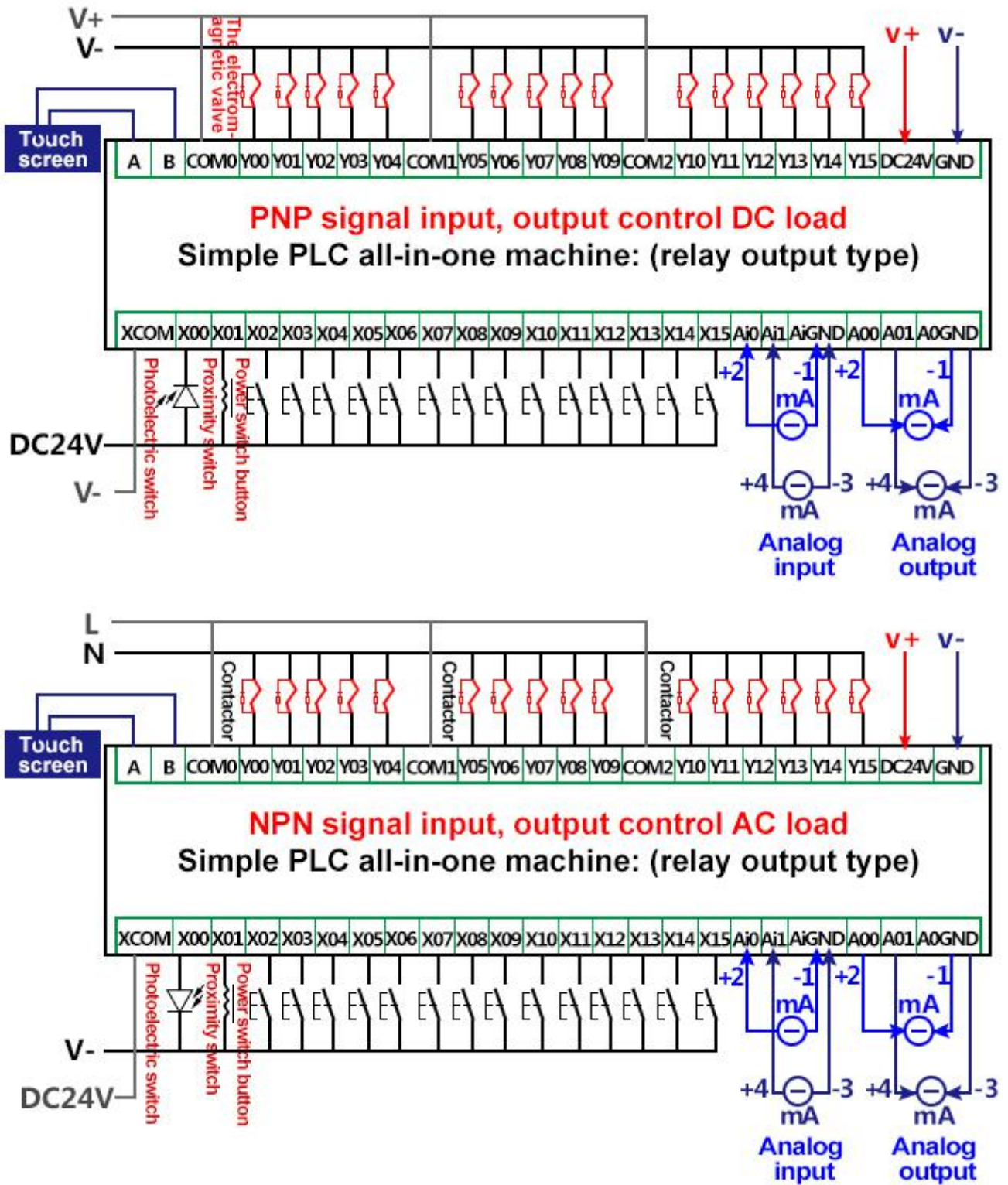
We just provide consult and suggestions to users about the programming and we do not provide free programming service.

### Wiring diagram of simple PLC all-in-one modules(DC transistor output) 12-in and 12-out



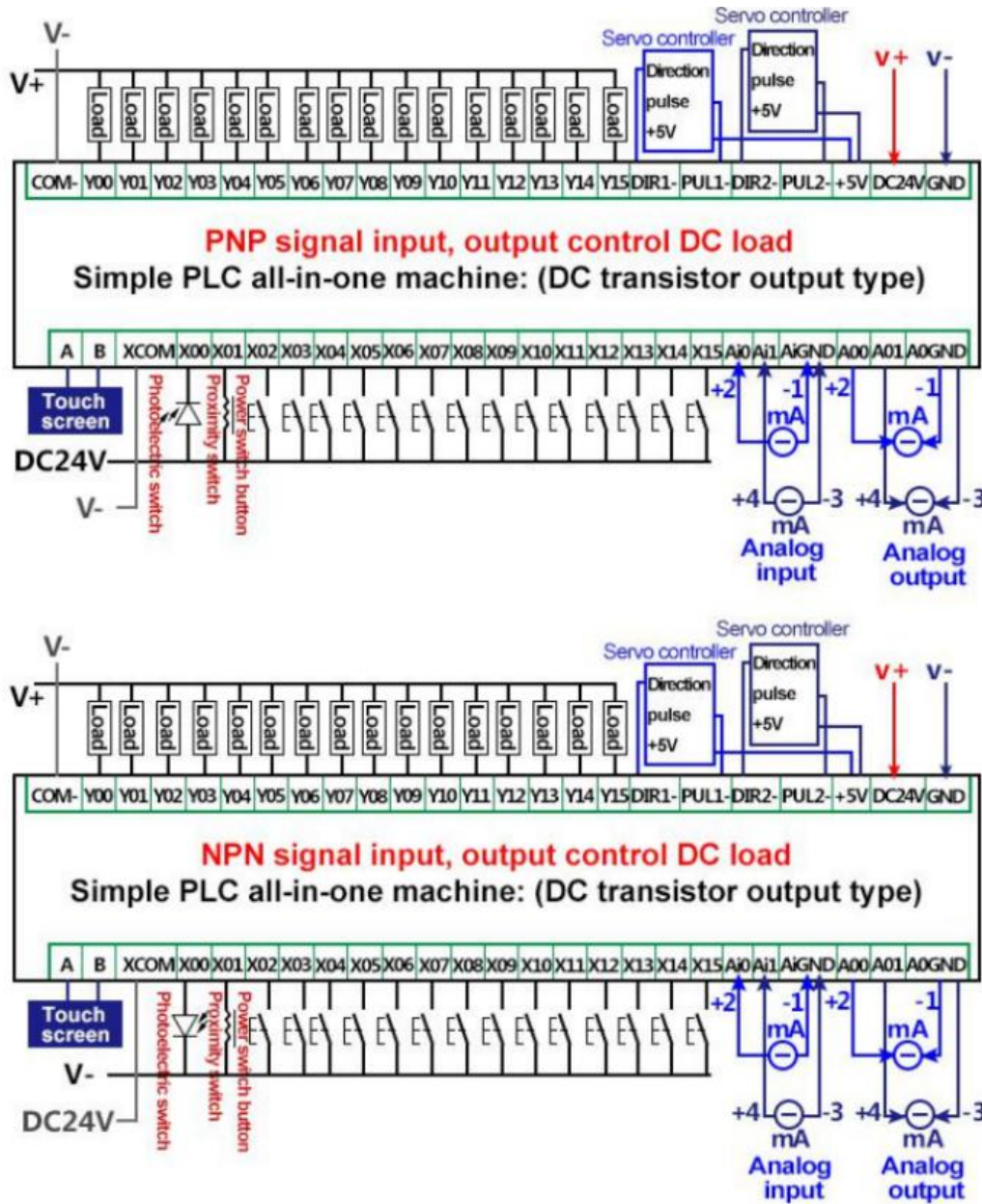
We just provide consult and suggestions to users about the programming and we do not provide free programming service.

### Wiring diagram of simple PLC all- in-one modules ( relay output) 16-in and 16-out



We just provide consult and suggestions to users about the programming and we do not provide free programming service.

**Wiring diagram of simple PLC all-in-one modules(DC transistor output)**  
**16-in and 16-out**



*We just provide consult and suggestions to users about the programming and we do not provide free programming service.*