Lesson 5 Control Multiple PWM Servos

1. Working Principle

PWM servo can be controlled by sending pulse signal so that you can change servo port, rotation angle and rotation time in program to control servo.

The path to the source code of the program is 5.Hardware Basic Learning/ Python Development/Program Files/Control Multiple PWM Servos /main.py

```python
1   import time
2   from PWMServo import PWMServo
3
4   # Control multiple PWM servos
5
6   pwm = PWMServo()
7   pwm.work_with_time()
8
9   if __name__ == '__main__':
10      pwm.run(1, 500, 1000) #ID1 PWM servo is set as 500 pulse width. The running time is 1000ms (PWM
        servo cannot read the current position, so the runing time can not be controlled at the firs time)
11      pwm.run(2, 500, 1000) # The ID2 PWM servo is set as 500 width pulse and the running time is set as
        1000ms.
12      time.sleep_ms(2000)    # The delay of 2000ms
13
14      pwm.run(1, 2500, 2000) # The ID1 PWM servo is set as 2500 width pulse and the running time is set
        as 2000ms.
15      pwm.run(2, 2500, 2000) # The ID2 PWM servo is set as 500 width pulse and the running time is set as
        2000ms.
16      time.sleep_ms(2000)    # The delay of 2000ms
17
18      pwm.run(1, 500, 2000) # The ID1 PWM servo is set as 500 width pulse and the running time is set as
        2000ms.
19      pwm.run(2, 500, 2000) # The ID2 PWM servo is set as 500 width pulse and the running time is set as
        2000ms.
20      time.sleep_ms(2000)    # The delay of 2000ms
```

PWM servo mainly calls run() function in PWMServo library. Take the code "pwm.run(1, 500, 1000)" as example.

The first parameter "1" is the port number of PWM servo. Here is No.1 port.

The second parameter "500" is the rotation position which is converted by pulse width data (pulse width=11.1×angle+500, the formula just for your information). Therefore, the parameter 500 corresponds to 0° rotation angle.

The third parameter "1000" is the rotation time (unit: ms). The parameter here is 1000, i.e, 1000ms.

2. Preparation

Connect two PWM servos to PWM servo ports on MaxArm controller. Take connecting two LFD-01 servos (5V servo) to No.1 and No.2 ports as example. The wiring method is as follow:
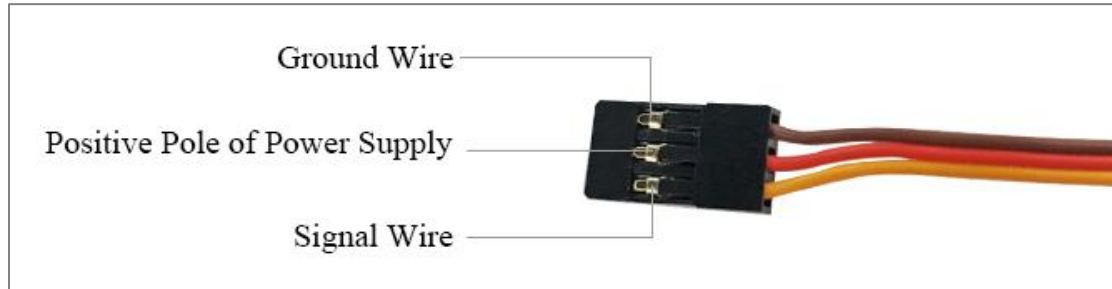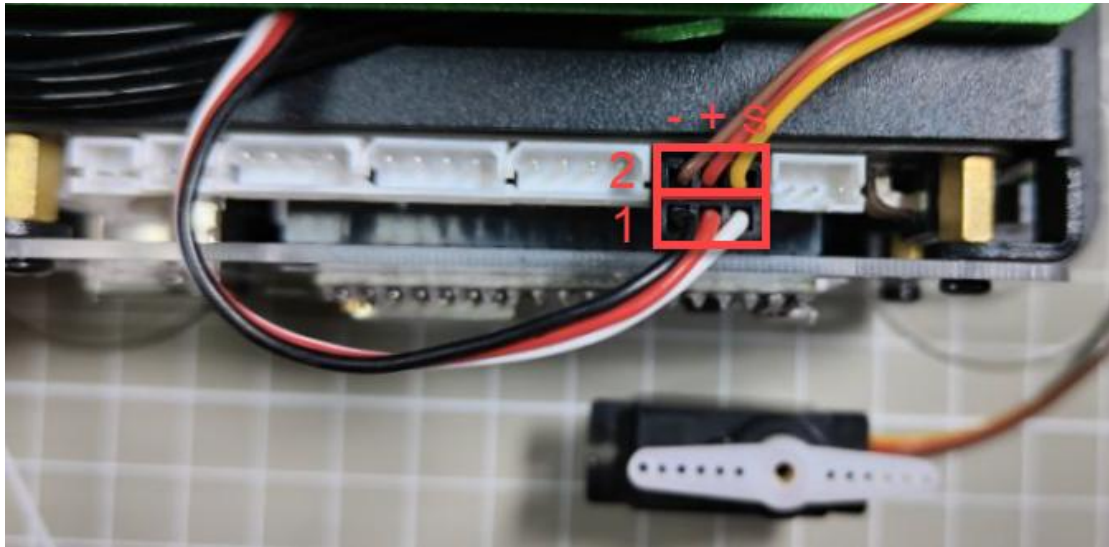
Note: Please note the direction of servo cable, otherwise servo may burn out (S pin is signal terminal).

3. Operation Steps

1) Please connect MaxArm to Python editor according to the tutorial in folder "4. Underlying Program Learning/Python Development/Lesson 1 Set Development Environment".

Port:COM18 Board:esp32,connect successfully!
>>>
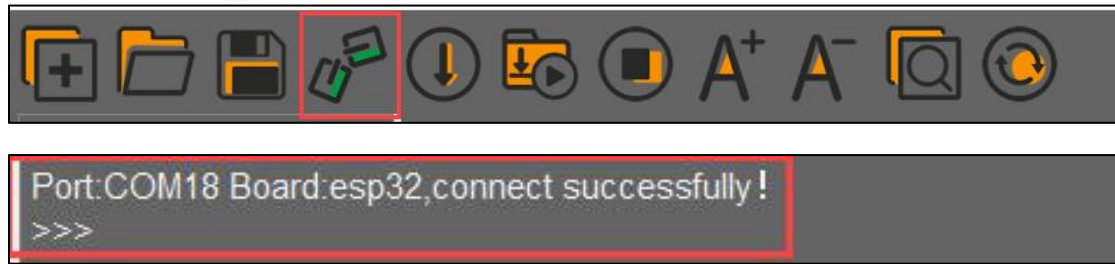
2)　After connecting, change the path of Workspace to "5.Hardware Basic Learning/Python Development" and select "Program Files".
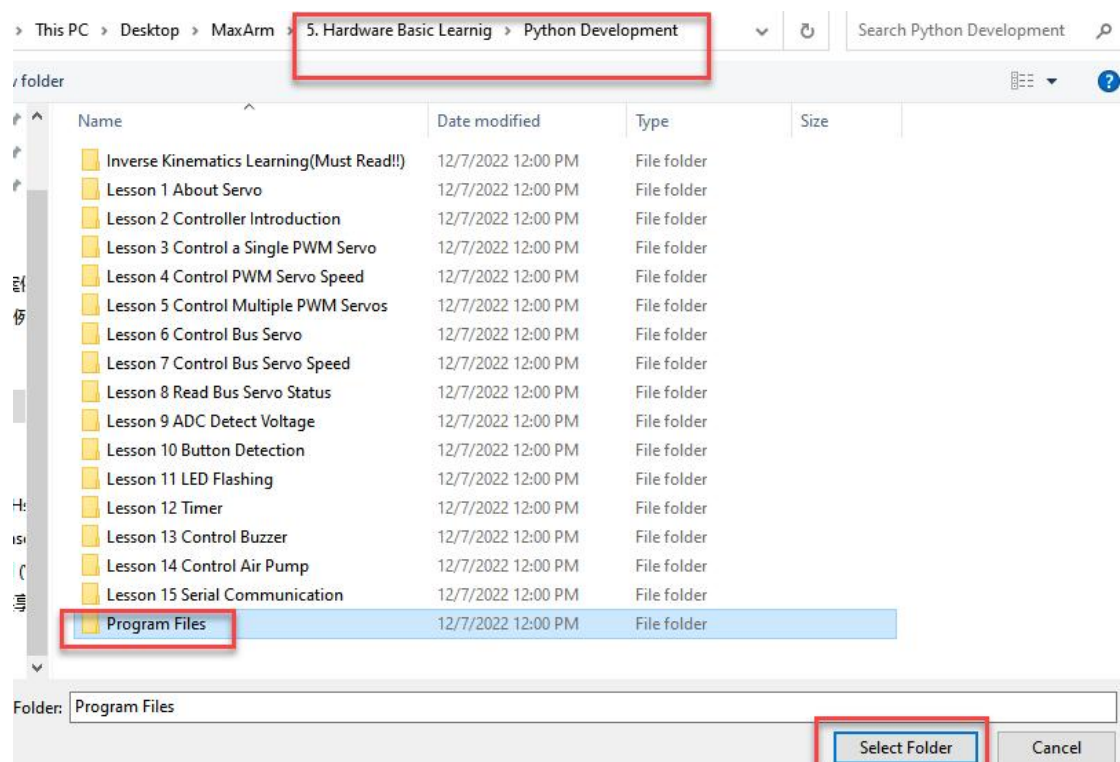
Device(connected)
Workspace
Change workspace
Show in explorer

This PC > Desktop > MaxArm > 5. Hardware Basic Learnig > Python Development

Name	Date modified	Type	Size
Inverse Kinematics Learning(Must Read!!)	12/7/2022 12:00 PM	File folder	
Lesson 1 About Servo	12/7/2022 12:00 PM	File folder	
Lesson 2 Controller Introduction	12/7/2022 12:00 PM	File folder	
Lesson 3 Control a Single PWM Servo	12/7/2022 12:00 PM	File folder	
Lesson 4 Control PWM Servo Speed	12/7/2022 12:00 PM	File folder	
Lesson 5 Control Multiple PWM Servos	12/7/2022 12:00 PM	File folder	
Lesson 6 Control Bus Servo	12/7/2022 12:00 PM	File folder	
Lesson 7 Control Bus Servo Speed	12/7/2022 12:00 PM	File folder	
Lesson 8 Read Bus Servo Status	12/7/2022 12:00 PM	File folder	
Lesson 9 ADC Detect Voltage	12/7/2022 12:00 PM	File folder	
Lesson 10 Button Detection	12/7/2022 12:00 PM	File folder	
Lesson 11 LED Flashing	12/7/2022 12:00 PM	File folder	
Lesson 12 Timer	12/7/2022 12:00 PM	File folder	
Lesson 13 Control Buzzer	12/7/2022 12:00 PM	File folder	
Lesson 14 Control Air Pump	12/7/2022 12:00 PM	File folder	
Lesson 15 Serial Communication	12/7/2022 12:00 PM	File folder	
Program Files	12/7/2022 12:00 PM	File folder	

Folder: Program Files

Select Folder　Cancel

3)　Double click folder "Control Multiple PWM Servos", and then double click "main.py" to open program.
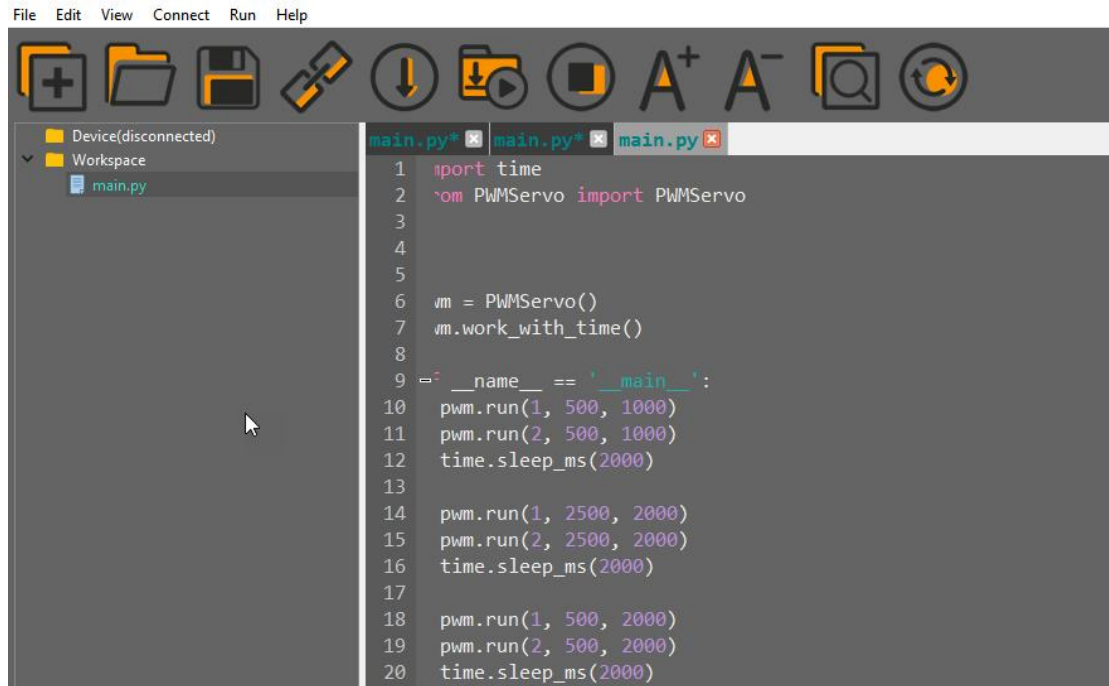
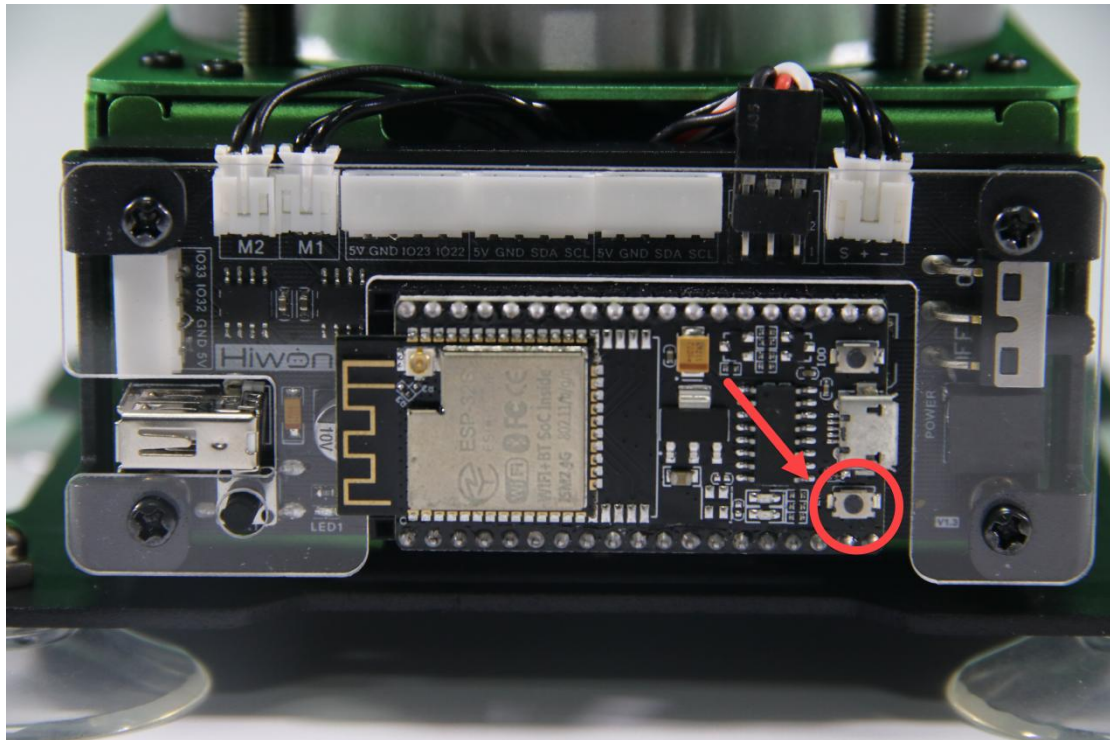4) Click on the download icon to download program to ESP32 controller.

5) When the terminal prints the prompt, as shown in the image below, it means download completed.

6) After downloading, click on the reset icon or press the reset button on ESP32 controller to run program.
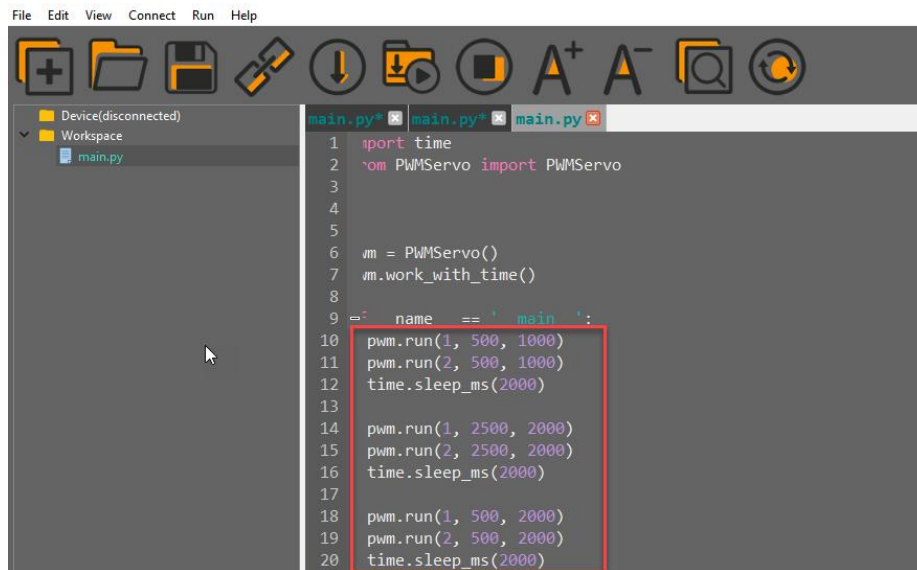
4. Project Outcome

When running program, two LFD-01M servos will rotate from 0° to 180°, and then to 0°. After the servos stop rotating, exit program automatically.
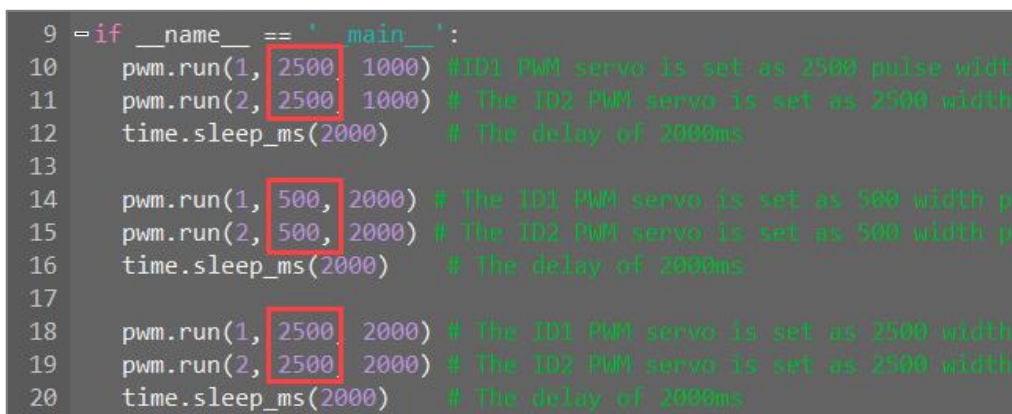
5. Function Extension

The rotation position set in program is from 0° to 180°, and then to 0°. You can modify the rotation position by modifying the corresponding code. Here the second parameter of run() function of No.1 servo is changed from 500 to 2500, and the second parameter of the run() function of No.2 servo is changed from 2500 to 500. The specific operation steps are as follow:

Find the following program code:

```
File  Edit  View  Connect  Run  Help

Device(disconnected)
Workspace
   main.py

main.py*   main.py*   main.py
 1   import time
 2   from PWMServo import PWMServo
 3
 4
 5
 6   pwm = PWMServo()
 7   pwm.work_with_time()
 8
 9   if __name__ == '__main__':
10       pwm.run(1, 500, 1000)
11       pwm.run(2, 500, 1000)
12       time.sleep_ms(2000)
13
14       pwm.run(1, 2500, 2000)
15       pwm.run(2, 2500, 2000)
16       time.sleep_ms(2000)
17
18       pwm.run(1, 500, 2000)
19       pwm.run(2, 500, 2000)
20       time.sleep_ms(2000)
```

1) Change the second parameter of run() function of No.1 servo from 500 to 2500, and the second parameter of the run() function of No.2 servo from 2500 to 500, as shown in the image below:

```
 9   if __name__ == '__main__':
10       pwm.run(1, 2500, 1000) #ID1 PWM servo is set as 2500 pulse width
11       pwm.run(2, 2500, 1000) # The ID2 PWM servo is set as 2500 width
12       time.sleep_ms(2000)    # The delay of 2000ms
13
14       pwm.run(1, 500, 2000) # The ID1 PWM servo is set as 500 width p
15       pwm.run(2, 500, 2000) # The ID2 PWM servo is set as 500 width p
16       time.sleep_ms(2000)    # The delay of 2000ms
17
18       pwm.run(1, 2500, 2000) # The ID1 PWM servo is set as 2500 width
19       pwm.run(2, 2500, 2000) # The ID2 PWM servo is set as 2500 width
20       time.sleep_ms(2000)    # The delay of 2000ms
```

2) After modifying, click on icon to check grammar. In the mean time, the terminal will show the following prompt.

```
>>>
Syntax check completed,no errors
```

3) Click on icon and refer to "Operation Steps 4-6" to download and run the program.