Lesson 7 Control Bus Servo Speed

1. Working Principle

According to the communication protocol, servo is controlled to rotate by sending the commands including servo ID, rotation angle and time.

The path to the source code of the program is 5.Hardware Basic Learning/Python Development/Program Files/Control Bus Servo Speed/ main.py

```python
1   import time
2   from BusServo import BusServo
3
4   # Control the bus servo speed
5
6   bus_servo = BusServo()
7
8   if __name__ == '__main__':
9       bus_servo.run(1, 500, 1500) # The ID1 bus servo is set as 500 width pulse and the running
        time is set as 1500ms.
10      time.sleep_ms(1500)        # The delay of 1500ms
11
12      for t in (500, 2000):      # Run one round at different times, the longer the time, the slower
        the speed
13          bus_servo.run(1, 600, t) # : The ID1 servo is set to run to 700 pulse width.
14          time.sleep_ms(t)
15
16          bus_servo.run(1, 500, t) # : The ID1 servo is set to run to 500 pulse width.
17          time.sleep_ms(t)
18
19          bus_servo.run(1, 400, t) # : The ID1 servo is set to run to 300 pulse width.
20          time.sleep_ms(t)
21
22          bus_servo.run(1, 500, t) # : The ID1 servo is set to run to 500 pulse width.
23          time.sleep_ms(t)
```

Control bus servo by calling run() function in BusServo library. Take the code "bus_servo.run(1, 500, 1000)" as example.

The first parameter "1" is the servo ID. Here is ID1 servo.

The second parameter "500" represents the rotation position. The parameter is the data converted by angle.

The third parameter "1000" represents the rotation time (unit is ms). Here the

time is 1000ms.

The rotation range of bus servo is between 0 and 1000 pulse width which corresponds to 0°-240°, i.e, 1° is roughly equal to 4.2 pulse width. The conversion formula for angle and pulse width is: pulse width=4.2×angle (just for reference).

2. Preparation

Please make sure the bus servo is individually connected to bus servo port on MaxArm controller. The wiring method is as follow:
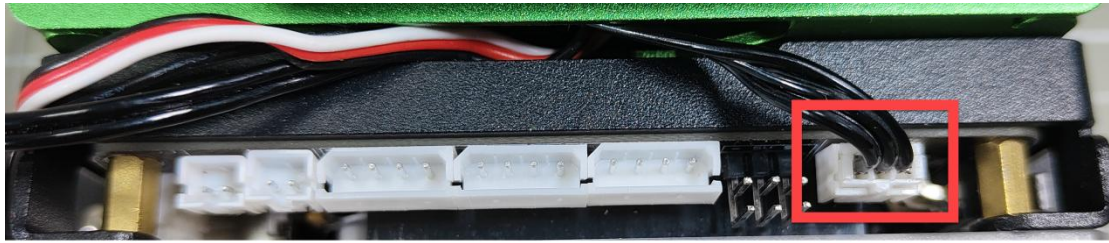
Note: The servo cable uses anti-reverse plug. Please do not insert it violently.

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".
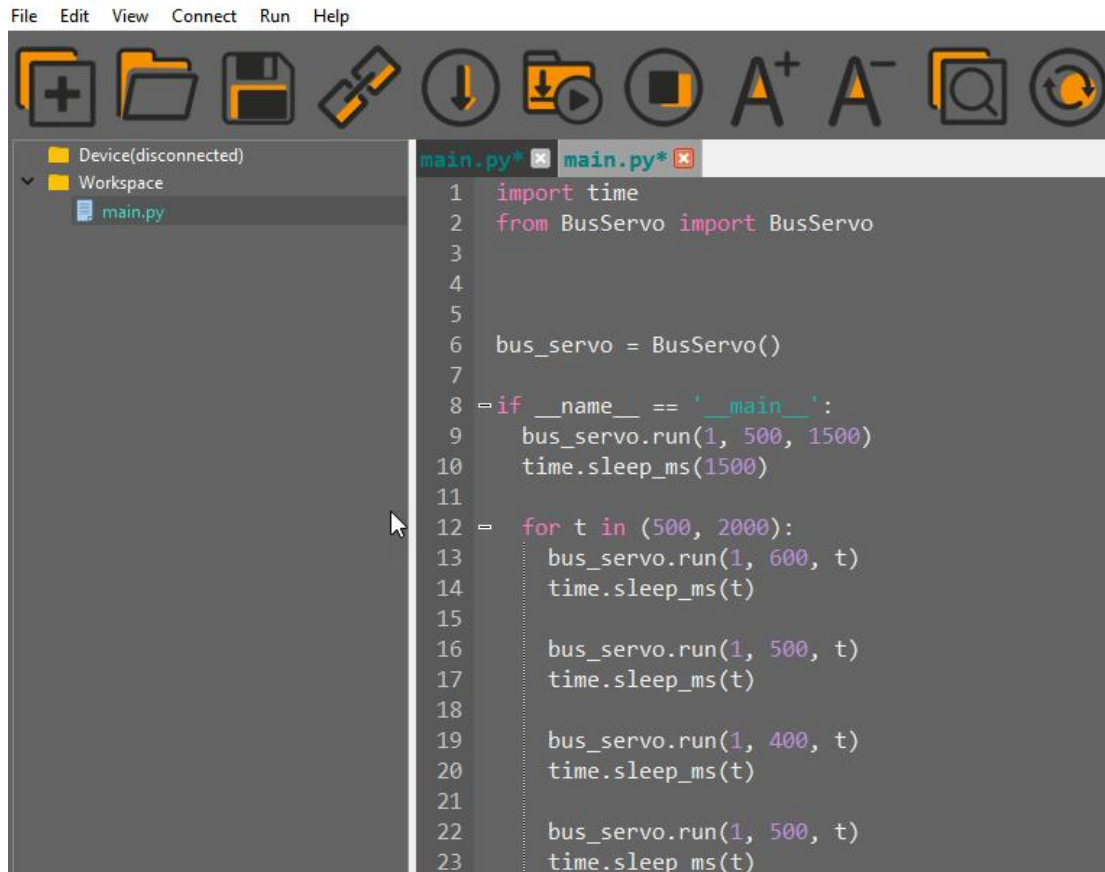

Port:COM18 Board:esp32,connect successfully!
>>>

2) After connecting, change the path of Workspace to "5.Hardware Basic Learning/Python Development" and select "Program Files".
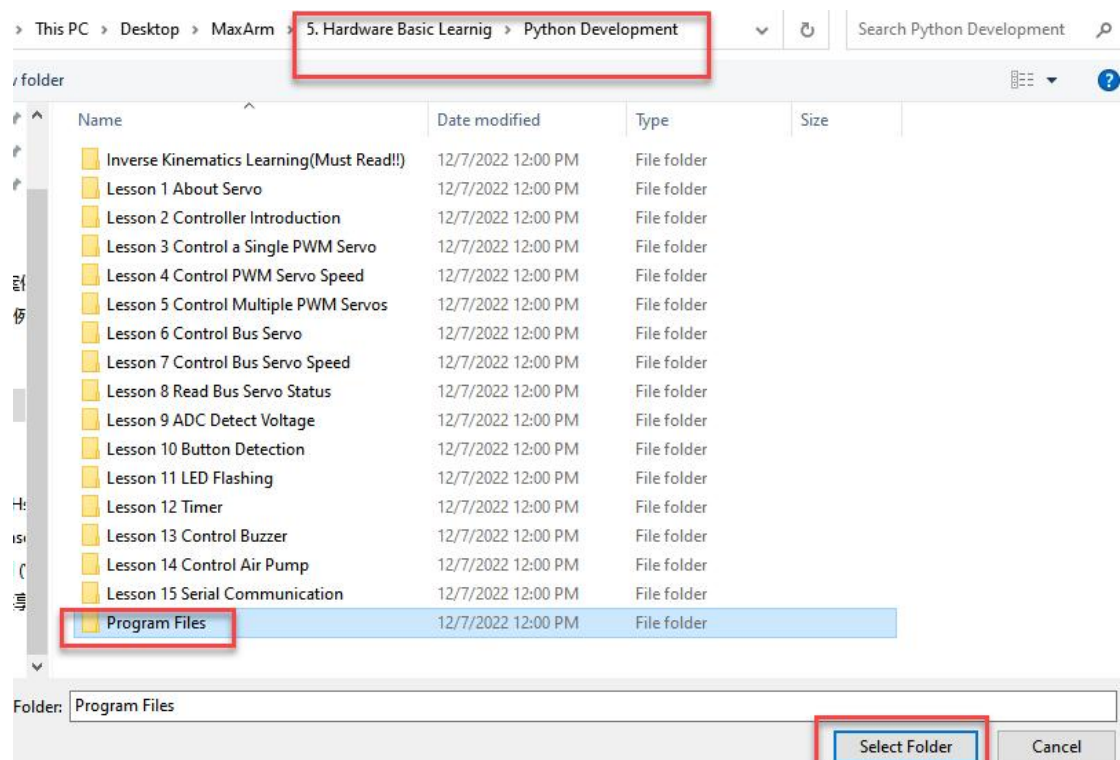
3) Double click folder "Control Bus Servo Speed", and then double click "main.py" to open program.

```
File  Edit  View  Connect  Run  Help

Device(disconnected)
Workspace
    main.py

main.py*  main.py*

 1    import time
 2    from BusServo import BusServo
 3
 4
 5
 6    bus_servo = BusServo()
 7
 8    if __name__ == '__main__':
 9        bus_servo.run(1, 500, 1500)
10        time.sleep_ms(1500)
11
12        for t in (500, 2000):
13            bus_servo.run(1, 600, t)
14            time.sleep_ms(t)
15
16            bus_servo.run(1, 500, t)
17            time.sleep_ms(t)
18
19            bus_servo.run(1, 400, t)
20            time.sleep_ms(t)
21
22            bus_servo.run(1, 500, t)
23            time.sleep_ms(t)
```

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.


```
>>>
Downloading....
main.py Download ok!
>>>
```

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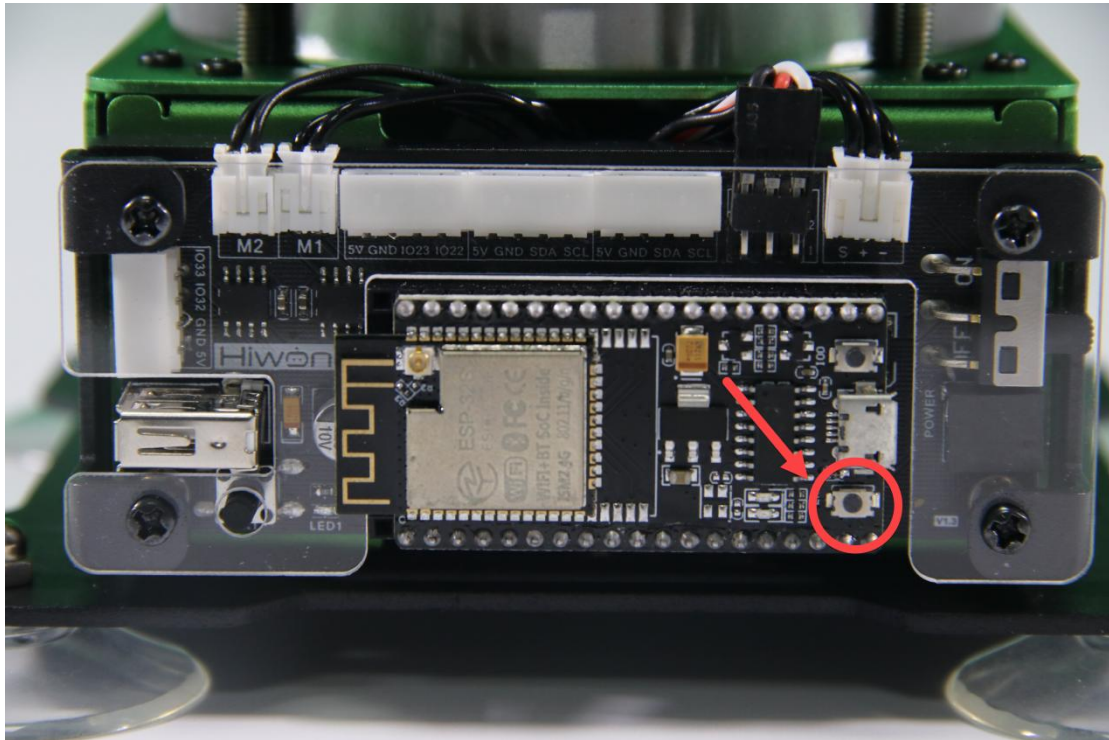
4. Project Outcome

When running program, ID1 servo will rotate 25° to the right, then 45° to the left, finally 25° to the right to return to the initial position. This process will repeat twice and the first rotation will be faster than the first rotation. After the servo stops rotating, exit the program automatically.
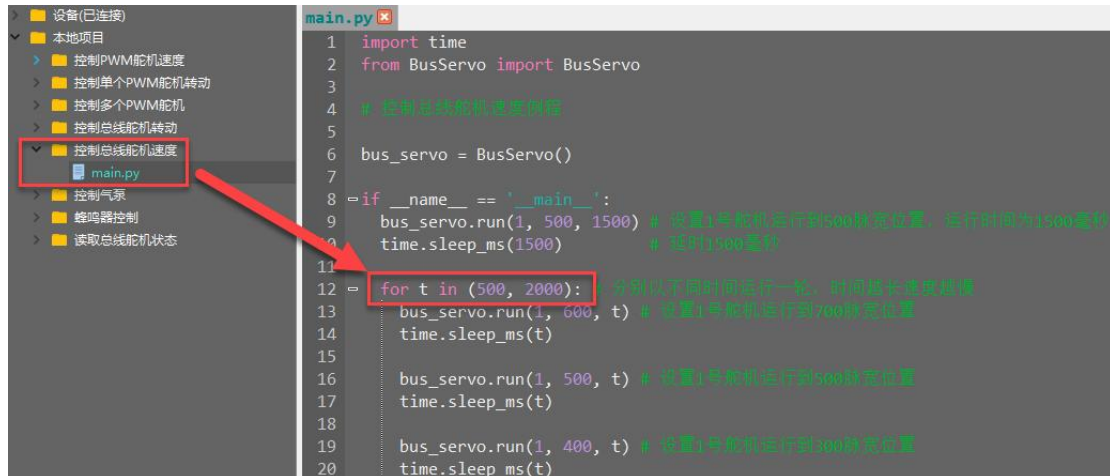
The rotation range of bus servo is between 0 and 1000 pulse width which corresponds to 0°-240°, i.e, 1° is roughly equal to 4.2 pulse width. The conversion formula of angle and pulse width is pulse with=4.2×angle (just for reference).

5. Function Extension

The first rotation speed set in program is faster than the second one. If want to change its rotation speed, you can modify the corresponding code to

implement. Here the "t" parameter value is changed from (500,2000) to (2000,500). The specific operation steps are as follow:

Find the following program:

1) Change the first parameter of "t" to 2000 and the second parameter to 500, as shown in the image below:

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

3) Click on icon.

4) Refer to "Operation Steps 4-6" to download and run the program.