

Lesson 12 Control PWM Servo Speed

1. Working Principle

Control PWM servo by sending pulse signal. You can modify the running time in program to modify the rotation speed.

The source code of program is located in:

`/home/ubuntu/armpi_pro/src/armpi_pro_demo/expansion_board_demo/PWMServo_`

`Speed.py`

```

1  #!/usr/bin/python3
2  # coding=utf8
3  import sys
4  import time
5  import threading
6  import Board
7
8  print("
9  *****Function: Hiwonder Raspberry Pi expansion board. PWM Servo Variable Routine*****
10 *****
11 *****
12
13 Official website:https://www.hiwonder.com
14 Online mall:https://hiwonder.tmall.com
15
16 Tips:
17 * Press "Ctrl+C" to close the running program. If fail to close, please try several times!
18
19 ")
20
21 if sys.version_info.major == 2:
22     print('Please run this program with python3!')
23     sys.exit(0)
24
25
26 if __name__ == '__main__':
27     for i in range(5): # loop 5 times
28
29         Board.setPWMServoPulse(1, 500, 1000) # PWM servo connected to No.1 port rotated to 500 in 1000ms.
30         time.sleep(1)
31         Board.setPWMServoPulse(1, 1500, 300) # PWM servo connected to No.1 port rotated to 1500 in 1000ms.
32         time.sleep(1)
33         Board.setPWMServoPulse(1, 500, 1000) # PWM servo connected to No.1 port rotated to 500 in 1000ms.

```

Control PWM servo by calling `setPWMServoPulse()` function under Board library. Take the code “`Board.setPWMServoPulse(1, 500, 1000)`” as example:

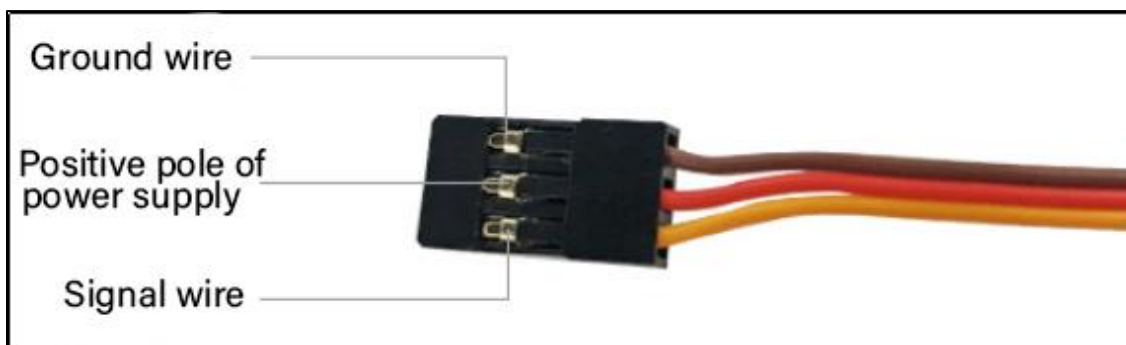
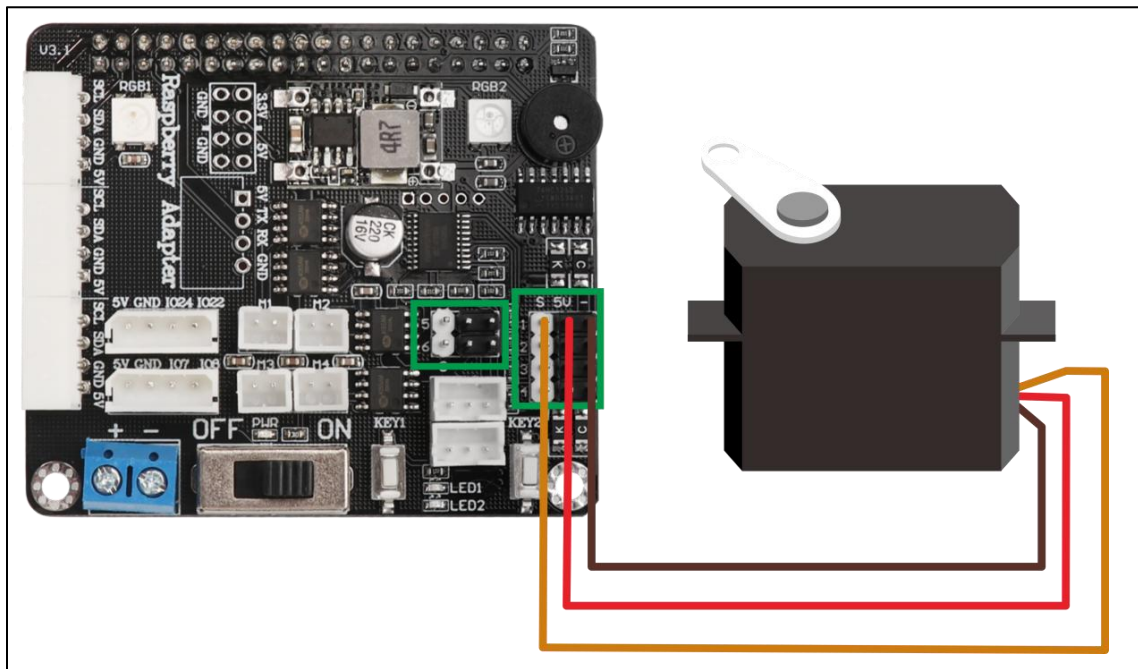
The first parameter “1” is the port number of the connected PWM servo. The port number here is 1.

The second parameter “500” is the rotation position, which is converted by angle conversion formula.

The third parameter “1000” is the rotation time (the unit is ms). Here is 1000ms.

2. Preparation

Connect single PWM servo to PWM servo ports on Raspberry Pi expansion board. Take LFD-01 servo (5V) as example, The wiring method is as the figure shown below:



Note:

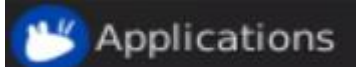

1. No.1-No.6 are PWM servo ports. The working voltage of No.1-No.4 servo

ports are 5V. The working voltage of No.5-No.6 servo ports are the same as that of power supply ports.

2. Please take notice of the wiring direction when connecting servo, otherwise it is easy to burn out servo (the white interfaces on the expansion board are the signal terminal).

3. Operation Steps

1) Please refer to the tutorial in “6.Raspberry Pi and Expansion Board Lessons/2.Raspberry Pi Expansion Board/Lesson 4 Set Environment Development” to remotely connect through NoMachine.

2) Open the terminal. Click  in the lower left corner and select  to enter the terminal.

3) In the opened interface, enter the command “`cd armpi_pro/src/armpi_pro_demo/expansion_board_demo/`” and press “Enter” to access to game programmings directory .

```
ubuntu@ubuntu:~$ cd armpi_pro/src/armpi_pro_demo/expansion_board_demo/
```

4) Then enter command “`sudo python3 PWMServo_Single.py`” and press “Enter” to start the game.

```
ubuntu@ubuntu:~/armpi_pro/src/armpi_pro_demo/expansion_board_demo$ sudo python3 PWMServo_Speed.py
```

5) If want to exit the program, you can press “Ctrl+C”. If fail to exit, please try it multiple times.

4. Project Outcome

When the program is running, LFD-01 servo turns from 0° to 90° and then back to 0°. After 5 cycles, it will automatically exit the program. You can find that the rotating from 0° to 90° is faster the rotation from 90° to 0° .

5. Function Extension

5.1 Adjust Speed

The program defaults that the servo takes 1000ms to rotate to 500. We can modify rotation speed by modifying running time. Take speeding up the servo rotation as example.

1) Enter command “`cd armpi_pro/src/arpmp_demo/expansion_board_demo/`” and press “Enter” to access to the game programmings directory.

```
ubuntu@ubuntu:~$ cd armpi_pro/src/arpmp_demo/expansion_board_demo/
```

2) Enter command “`sudo vim PWMServo_Speed.py`” and press “Enter” to open the program file.

```
ubuntu@ubuntu:~/arpmp_demo/expansion_board_demo$ sudo vim PWMServo_Speed.py
```

3) Find the code shown in the figure below.

```
28 Board.setPWMServoPulse(1, 500, 1000)
29 time.sleep(1)
30 Board.setPWMServoPulse(1, 1500, 300)
31 time.sleep(1)
32 Board.setPWMServoPulse(1, 500, 1000)
```

4) Press “i” key. When “INSERT” word appears, which means it has been switched to the editing mode.

```
32      Board.setPWMServoPulse(1, 500, 1000)
-- INSERT --
```

5) We can change the connected servo port by modifying servo parameters (Board.setPWMServoPulse (1, 500, 1000)). The first parameter “1” is the connected servo port; the second parameter “500” is the pulse width; the third parameter “1000” is the running time, as the figure shown below:

```
29      Board.setPWMServoPulse(1, 500, 500)
30      time.sleep(1)
31      Board.setPWMServoPulse(1, 1500, 300)
32      time.sleep(1)
33      Board.setPWMServoPulse(1, 500, 500)
```

6) After modifying, press “Esc”. Then enter “:wq” and press “Enter” to save and exit.

```
33      Board.setPWMServoPulse(1, 500, 500)
:wq
```

7) After saving program, please refer to “3.Operation Steps” to check the outcome.

5.2 Change Servo Connection Port

- 1) The default servo connection port is No.1. Take connecting No.5 servo port as example.
- 2) You can refer to the operation steps in “5.1 Adjust speed (1-4) to switch to the editing mode.
- 3) Next, start to change No.1 port to No.5 port

```
29 Board.setPWMServoPulse(5, 500, 500)
30 time.sleep(1)
31 Board.setPWMServoPulse(5, 1500, 300)
32 time.sleep(1)
33 Board.setPWMServoPulse(5, 500, 500)
```

- 4) After modifying, press “Esc” and input “:wq”. Then press “Enter” to save and exit.

```
33 Board.setPWMServoPulse(5, 500, 500)
:wq
```

- 5) After saving program, please refer to “3.Operation Steps” to check the outcome.