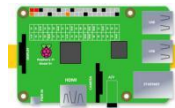
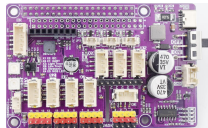



Lesson 9 How to Control RGB LED

In this lesson, we will learn how to control warm color LED with Raspberry Pi.

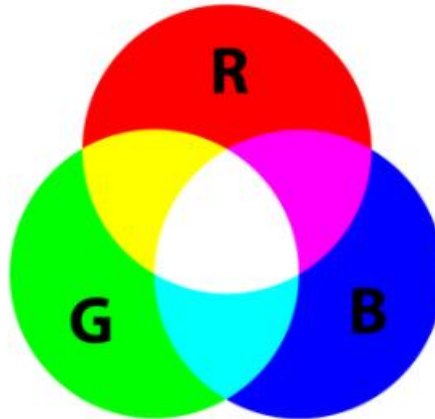
9.1 Components used in this Course

Components	Quantity	Picture
Raspberry Pi	1	
Aadept Robot HAT V3.1	1	
RGB LED	1	

9.2 Introduction of RGB LED

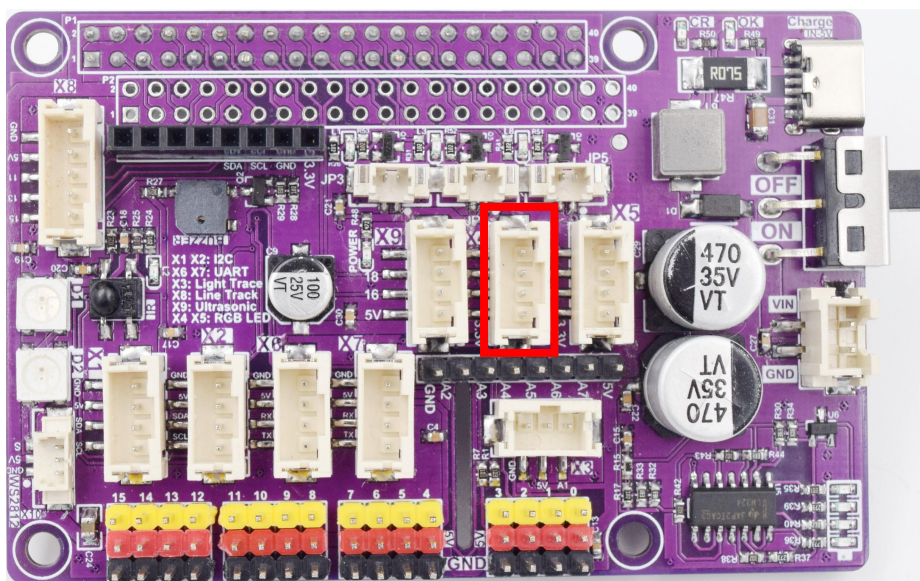
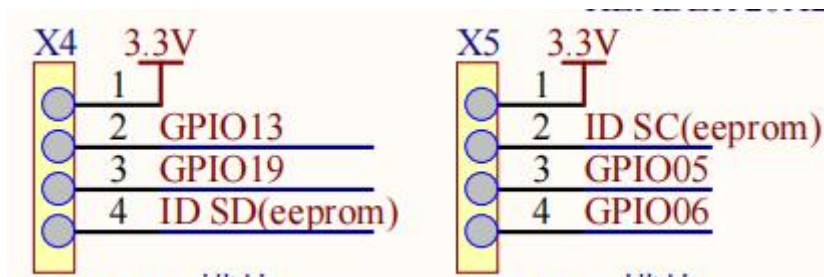
Using RGB LEDs, you can produce red, green and blue light, and by configuring the intensity of each LED, you can also produce other colors. For example, to produce pure blue light, you can set the red LED to the highest intensity, and the green and blue LEDs to the lowest intensity. For white light, replace with all three LEDs set to the highest intensity. To produce other colors, you can combine the three colors with different intensities. To adjust the intensity of each LED, you can use a PWM signal. Because the LEDs are very close to each other, what our eyes see is the result of the color combination, not the divided three colors.

The following chart. This is the simplest color mixing diagram.



9.3 Wiring Diagram

Use a 4Pin cable to connect the RGB LED to X4 of Adept Robot HAT V3.1.



9.4 How to Control the RGB LED

Run the code

1. Log in to your Raspberry Pi via SSH (the method to log in to the Raspberry Pi has been introduced in Lesson 3):

```
Linux raspberrypi 4.19.118-v7l+ #1311 SMP Mon Apr 27 14:26:42 BST 2020 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Aug 29 08:17:49 2020 from 192.168.3.208

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.

pi@raspberrypi:~ $
```

2. Run the command to enter the folder. This folder stores the sample code program for controlling the robot. Enter the following command and press Enter:

```
cd adeept_picar-b2/examples/
```

```
pi@raspberrypi:~ $
pi@raspberrypi:~ $ cd adeept_picar-b2/examples/
pi@raspberrypi:~/adeept_picar-b2/examples $
```

3. Enter the command to view the contents of the current directory:

```
ls
```

```
pi@raspberrypi:~/adeept_picar-b2/examples $ ls
01_LED.py      03_servo.py   05_RGB.py      07_ultra.py      09_lightTracking.py
02_buzzer.py  04_motor.py   06_ws2812.py   08_lineTracking.py
pi@raspberrypi:~/adeept_picar-b2/examples $
```

4. Enter the command and press Enter to run the program:

```
sudo python3 05_RGB.py
```

```
pi@raspberrypi:~/adeept_picar-b2/examples $  
pi@raspberrypi:~/adeept_picar-b2/examples $ sudo python3 05_RGB.py
```

5. After successfully running the program, you can observe that the RGB led will light up in different colors.
6. When you want to terminate the running program, you can press the shortcut key "Ctrl + C" on the keyboard.

9.5 Main code

The complete code reference file [05_RGB.py](#).

```
01  #import RPi.GPIO as GPIO  
02  from gpiozero import PWMOutputDevice as PWM  
03  import time  
04  
05  Left_R = 19  
06  Left_G = 13  
07  Left_B = 0  
08  
09  Right_R = 1  
10  Right_G = 5  
11  Right_B = 6  
12  
13  colors = [0xFF0000, 0x00FF00, 0x0000FF, 0xFFFF00, 0xFF00FF, 0x00FFFF, 0X6F00D2, 0xFF5809]  
14  # colors = [0xFF0000,0x00FF00,0x00FF00]  
15  
16  def setup():  
17      global L_R, L_G, L_B, R_R, R_G, R_B  
18  
19      L_R = PWM(pin=Left_R, initial_value=1.0, frequency=2000)  
20      L_G = PWM(pin=Left_G, initial_value=1.0, frequency=2000)  
21      L_B = PWM(pin=Left_B, initial_value=1.0, frequency=2000)  
22  
23      R_R = PWM(pin=Right_R, initial_value=1.0, frequency=2000)  
24      R_G = PWM(pin=Right_G, initial_value=1.0, frequency=2000)  
25      R_B = PWM(pin=Right_B, initial_value=1.0, frequency=2000)  
26  
27
```

```
28
29 def map(x, in_min, in_max, out_min, out_max):
30     return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min
31
32
33 def setAllColor(col):    # For example : col = 0x112233
34     R_val = (col & 0xff0000) >> 16
35     G_val = (col & 0x00ff00) >> 8
36     B_val = (col & 0x0000ff) >> 0
37
38     R_val = map(R_val, 0, 255, 0, 1.00)
39     G_val = map(G_val, 0, 255, 0, 1.00)
40     B_val = map(B_val, 0, 255, 0, 1.00)
41
42     L_R.value = 1.0-R_val
43     L_G.value = 1.0-G_val
44     L_B.value = 1.0-B_val
45
46     R_R.value = 1.0-R_val
47     R_G.value = 1.0-G_val
48     R_B.value = 1.0-B_val
49
50 def setAllRGBColor(R,G,B):    # For example : col = 0x112233
51
52     R_val = map(R, 0, 255, 0, 1.00)
53     G_val = map(G, 0, 255, 0, 1.00)
54     B_val = map(B, 0, 255, 0, 1.00)
55
56     L_R.value = 1.0-R_val
57     L_G.value = 1.0-G_val
58     L_B.value = 1.0-B_val
59
60     R_R.value = 1.0-R_val
61     R_G.value = 1.0-G_val
62     R_B.value = 1.0-B_val
63
64 def loop():
65     while True:
66         for col in colors:
67             setAllColor(col)
68             time.sleep(0.5)
69
```

```
70 def destroy():
71     L_R.stop()
72     L_G.stop()
73     L_B.stop()
74     R_R.stop()
75     R_G.stop()
76     R_B.stop()
77
78 if __name__ == "__main__":
79
80     setup()
81     try:
82         loop()
83     except KeyboardInterrupt:
84         destroy()
```