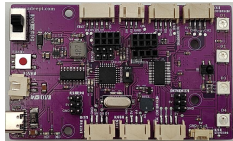




Lesson 6 How to Control the RGB LED

In this lesson, we will learn how to control the RGB LED.

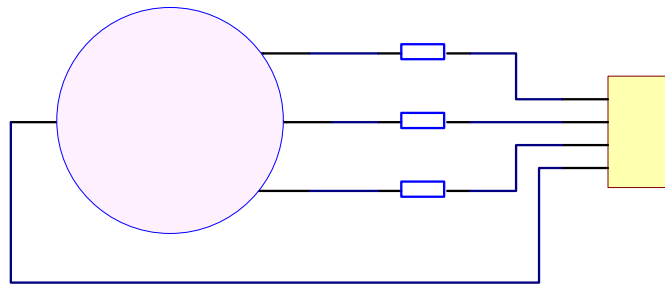
6.1 Components used in this course

Components	Quantity	Picture
Adeept Robot Control Board	1	
RGB LED	1	
Type-C USB Cable	1	

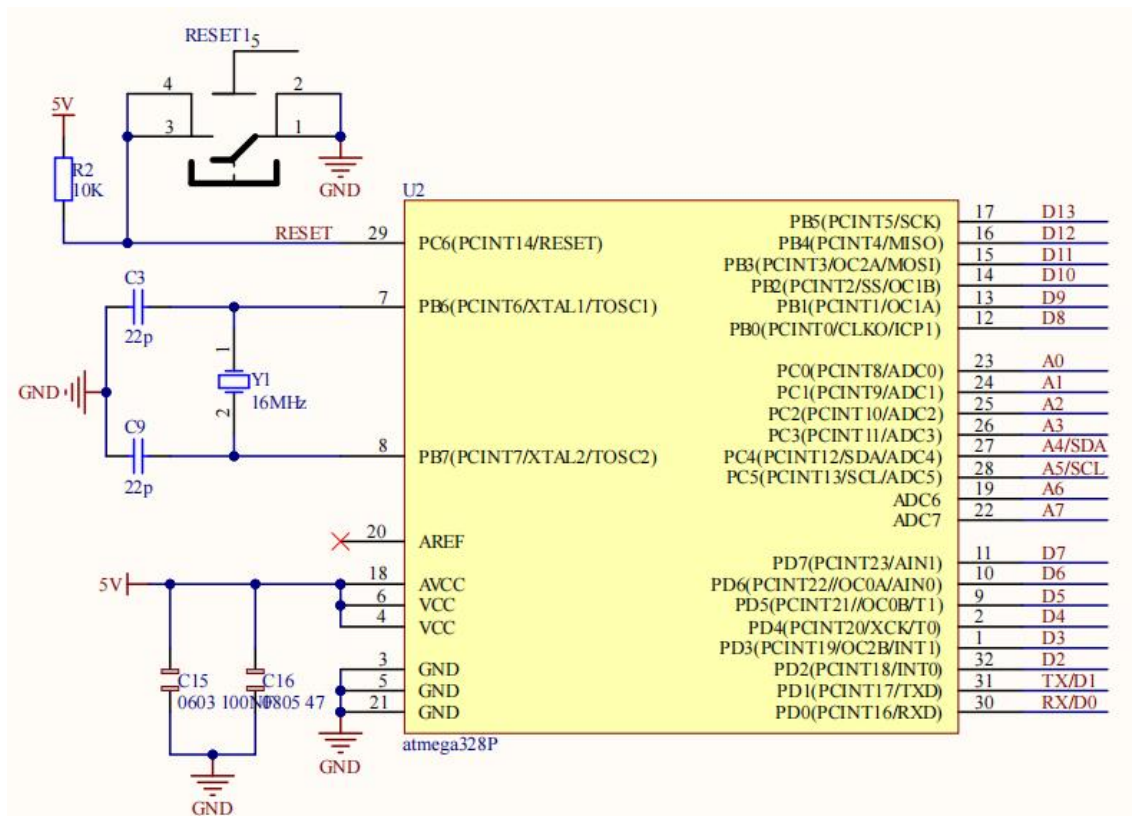
6.2 The introduction of the RGB LED

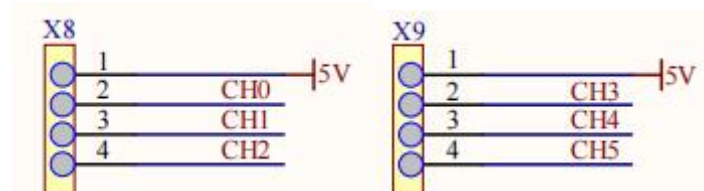
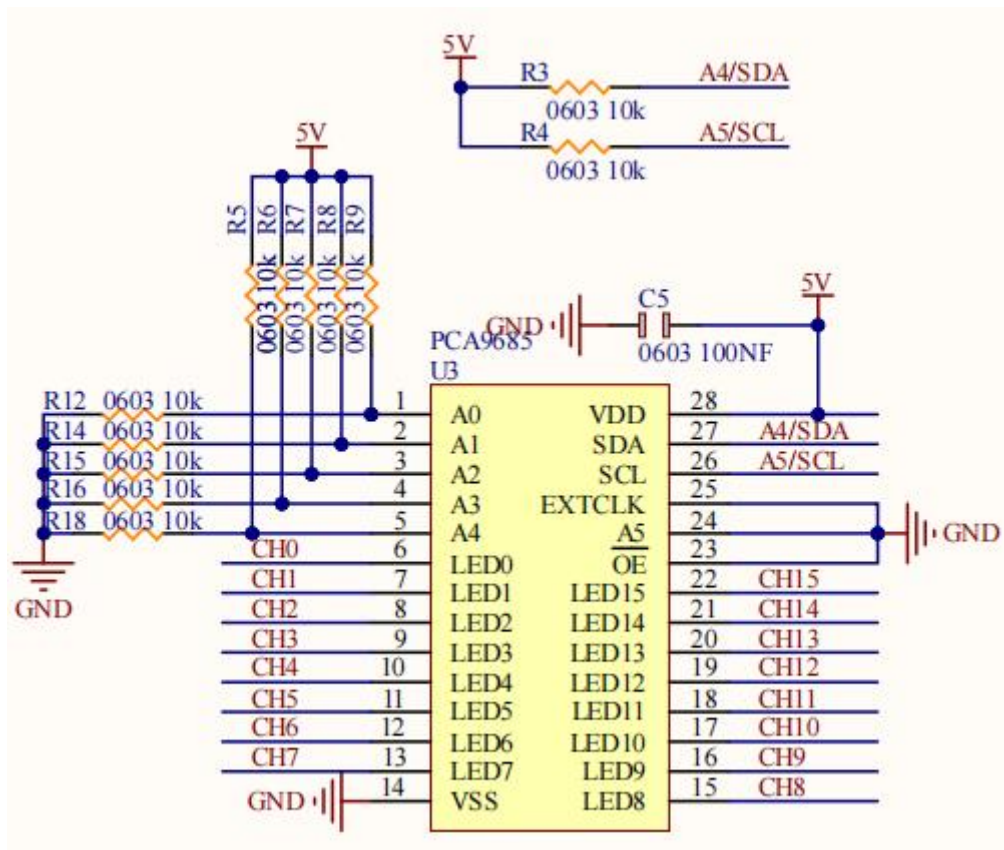
RGB LEDs consist of three LEDs. Each LED actually has one red, one green and one blue light. These three colored LEDs are capable of producing any color. Tri-color LEDs with red, green, and blue emitters, in general using a four-wire connection with one common lead (anode or cathode). These LEDs can have either common anode or common cathode leads.

The schematic diagram:



Adeept Robot Control Board uses PCA9685 extended pins CH0-CH5 as the interface of two RGB LEDs. The schematic diagram of the circuit is as follows:

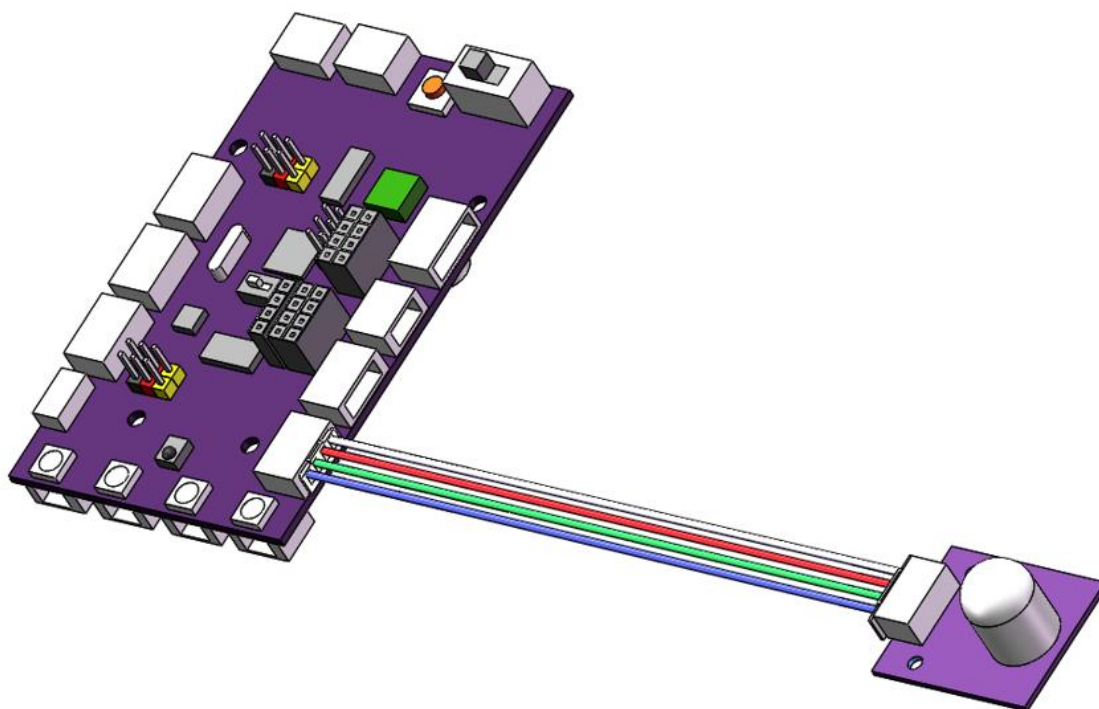
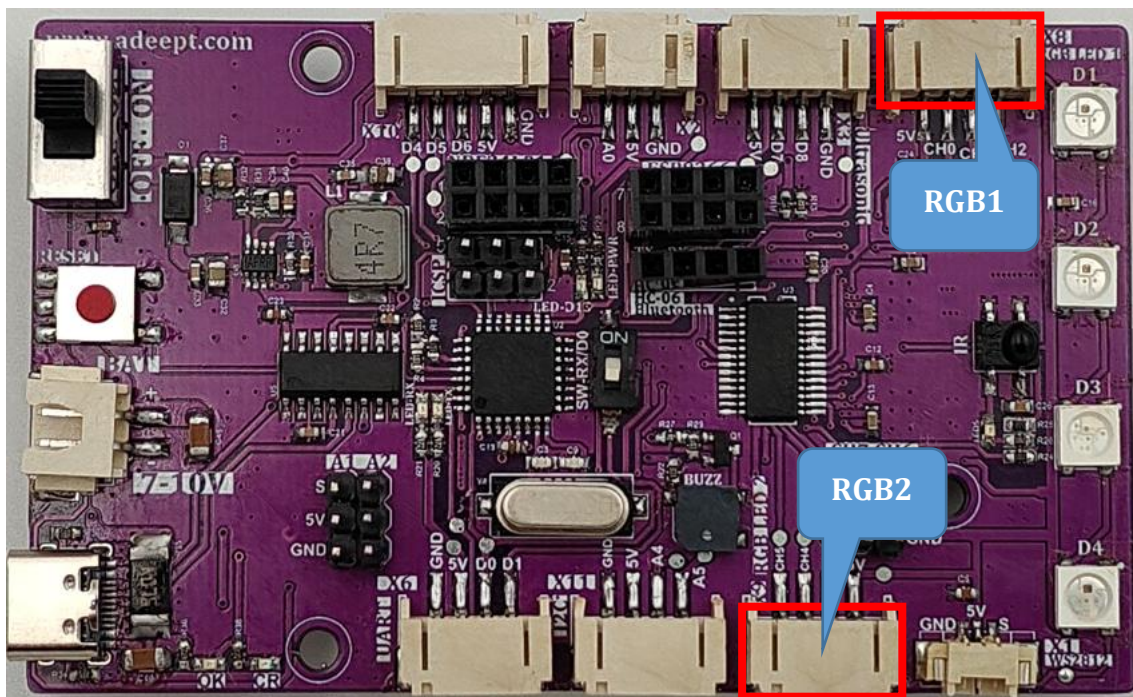




6.3 Wiring diagram

RGB1	Arduino(X8)	RGB2	Arduino(X9)
+	+5V	+	+5V
R	CH0	R	CH3
G	CH1	G	CH4
B	CH2	B	CH5

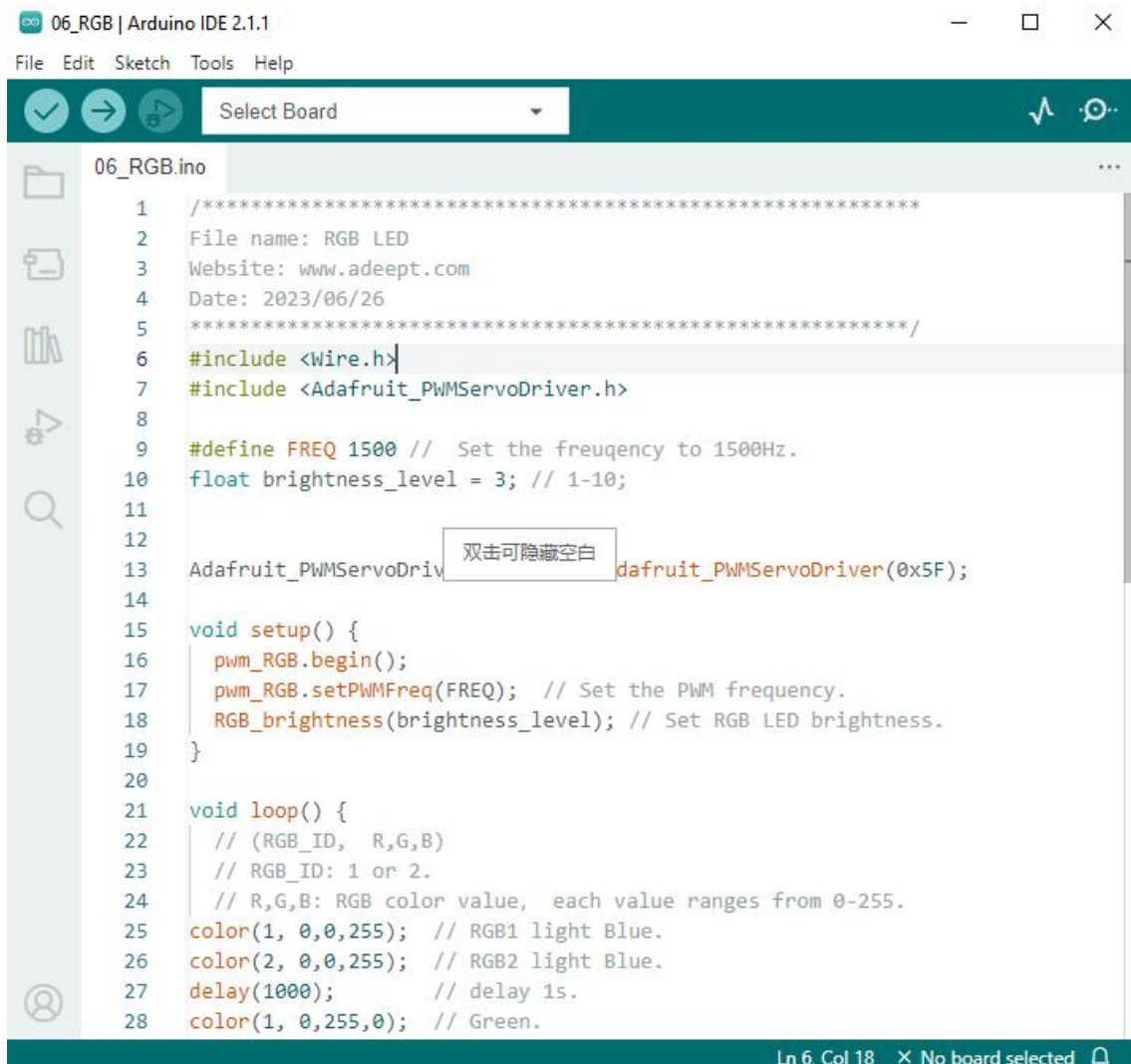
Figure as below:



The RGB module uses 4Pin cables, the colors are as shown in the picture, and the length is 13CM.

6.4 How to control the RGB LED

1. Connect your computer and Adeept Robot Control Board(Arduino Board) with a USB cable.
2. Open "06_RGB" folder in "[Adeept_UnoCar-B/Code](#)", double-click "05_RGB.ino".



```
06_RGB | Arduino IDE 2.1.1
File Edit Sketch Tools Help

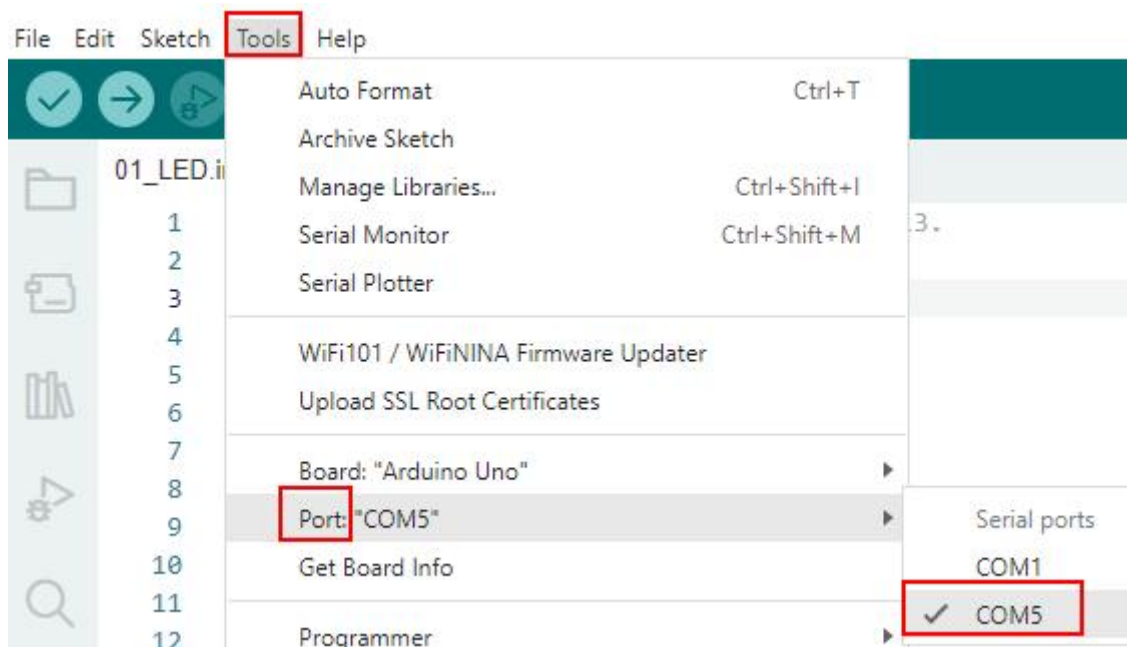
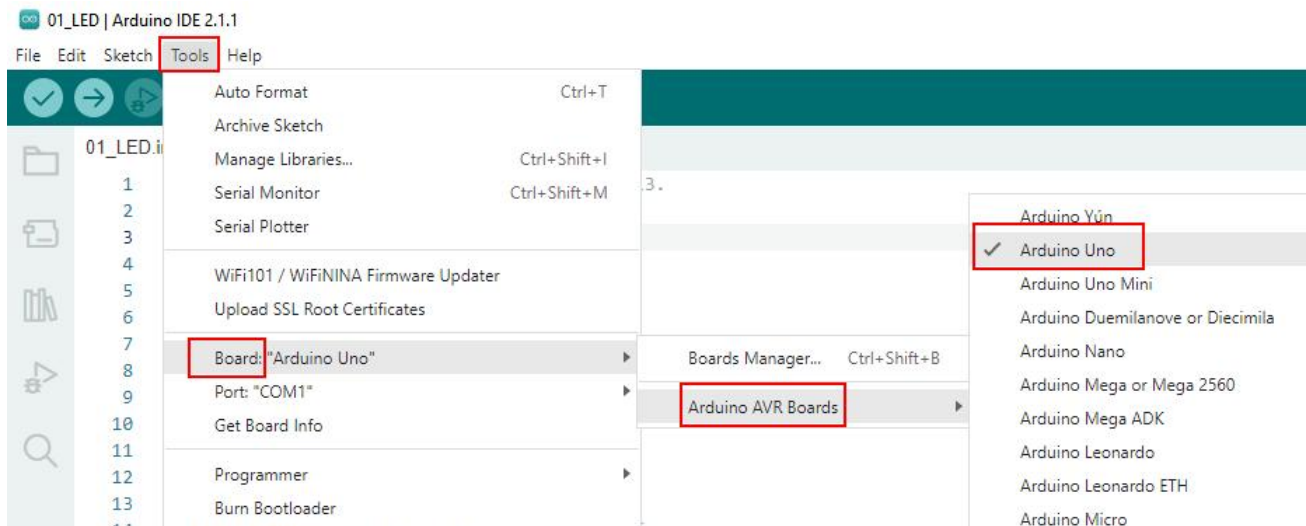
06_RGB.ino
1  /*****
2  File name: RGB LED
3  Website: www.adeept.com
4  Date: 2023/06/26
5  *****/
6  #include <Wire.h>
7  #include <Adafruit_PWMServoDriver.h>
8
9  #define FREQ 1500 // Set the frequency to 1500Hz.
10 float brightness_level = 3; // 1-10;
11
12 Adafruit_PWMServoDriver pwm_RGB = Adafruit_PWMServoDriver(0x5F);
13
14 void setup() {
15   pwm_RGB.begin();
16   pwm_RGB.setPWMFreq(FREQ); // Set the PWM frequency.
17   RGB_brightness(brightness_level); // Set RGB LED brightness.
18 }
19
20 void loop() {
21   // (RGB_ID, R,G,B)
22   // RGB_ID: 1 or 2.
23   // R,G,B: RGB color value, each value ranges from 0-255.
24   color(1, 0,0,255); // RGB1 light Blue.
25   color(2, 0,0,255); // RGB2 light Blue.
26   delay(1000); // delay 1s.
27   color(1, 0,255,0); // Green.
28 }
```

3. Select development board and serial port.


Board: Tools--->Board--->Arduino AVR Boards--->Arduino Uno

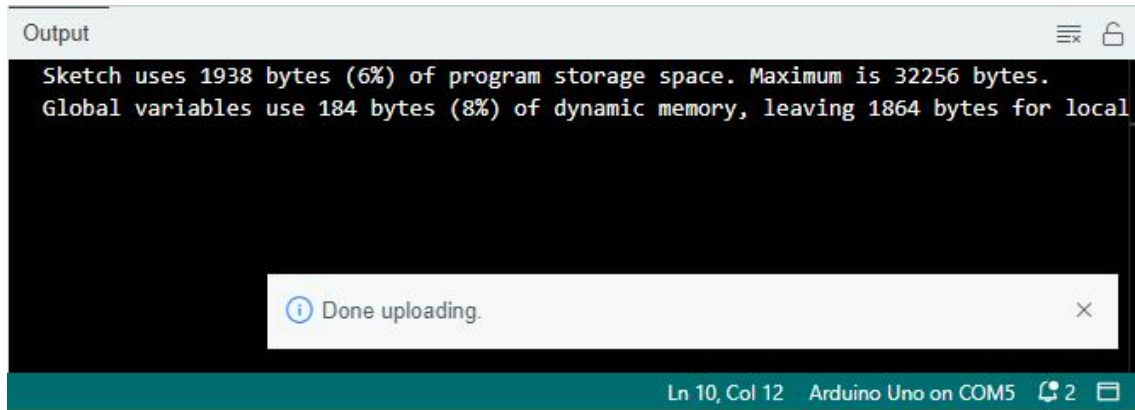
Port: Tools --->Port--->COMx

Note: The port number will be different in different computers.





4. After opening, click  to upload the code program to the Arduino. If there is no error warning in the console below, it means that the Upload is successful.



5. After successfully running the program, You will see the RGB lights cycle through the blue, green and red colors.

6.5 Code

```
1. #include <Wire.h>
2. #include <Adafruit_PWMServoDriver.h>
3.
4. #define FREQ 1500 // Set the frequency to 1500Hz.
5. float brightness_level = 3; // 1-10;
6.
7. Adafruit_PWMServoDriver pwm_RGB = Adafruit_PWMServoDriver(0x5F);
8.
9. void setup() {
10.   pwm_RGB.begin();
11.   pwm_RGB.setPWMFreq(FREQ); // Set the PWM frequency.
12.   RGB_brightness(brightness_level); // Set RGB LED brightness.
13. }
14.
15. void loop() {
16.   // (RGB_ID, R,G,B)
17.   // RGB_ID: 1 or 2.
```

```
18. // R,G,B: RGB color value, each value ranges from 0-255.
19. color(1, 0,0,255); // RGB1 light Blue.
20. color(2, 0,0,255); // RGB2 light Blue.
21. delay(1000); // delay 1s.
22. color(1, 0,255,0); // Green.
23. color(2, 0,255,0);
24. delay(1000);
25. color(1, 255,0,0); // Red.
26. color(2, 255,0,0);
27. delay(1000);
28. color(1, 0,0,0); // Light off.
29. color(2, 0,0,0);
30. delay(1000);
31. }
32.
33. // Set RGB LED brightness value.
34. void RGB_brightness(int level){
35.     brightness_level = level/10.0;
36. }
37.
38. // Set RGB LED color value.
39. void color(int index,float R,float G,float B){
40.     R = int(map(R, 0,255, 0, 4095*brightness_level));
41.     G = int(map(G, 0,255, 0, 4095*brightness_level));
42.     B = int(map(B, 0,255, 0, 4095*brightness_level));
43.
44.     if (index == 1){
45.         pwm_RGB.setPWM(0, 0, 4095-R); // (CH0, 0, PWM value)
46.         pwm_RGB.setPWM(1, 0, 4095-G); // (CH1, 0, PWM value)
47.         pwm_RGB.setPWM(2, 0, 4095-B); // (CH2, 0, PWM value)
48.     }
49.     else if (index == 2){
50.         pwm_RGB.setPWM(3, 0, 4095-R);
51.         pwm_RGB.setPWM(4, 0, 4095-G);
52.         pwm_RGB.setPWM(5, 0, 4095-B);
53.     }
54. }
```