

Expand course ---4.Ultrasonic RGB change color

1. Learning goal

In this lesson, we will learn how to change the color of Ultrasonic RGB light.

2. Preparation

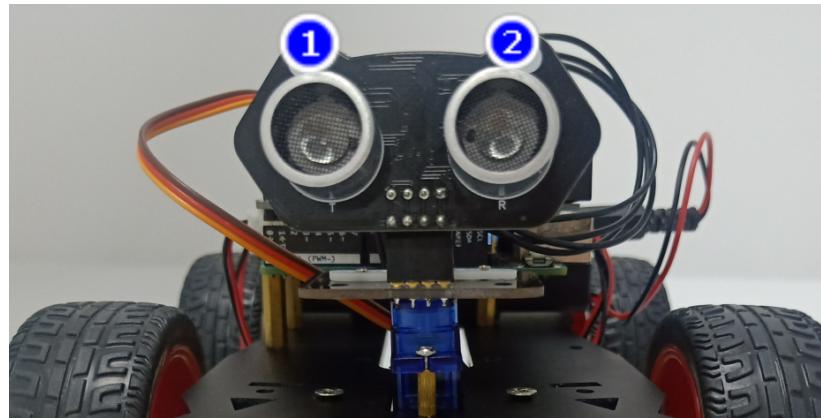
- 2.1 The position of the RGB light of the ultrasonic module.
- 2.2 The pin of UNO board is connected the RGB light of ultrasonic module.

3. Principle of experimental

RGB light (red, green, blue) are packaged in the LED module. We can mix different colors(256*256*256) by controlling the brightness of the three LEDs.

From the hardware interface manual, we can know that Ultrasonic RGB light is driven by Pin 11 of UNO board.

Classification	Function	The number of Drive chip PCA9685	Drive Method	Connection with CPU	Uno board			
Left Motor	Left front motor forward	LINB(13)	PCA9685	I2C_SDA/I2C_SCL	A4/A5			
	Left front motor reverse	LINA(12)						
	Left rear motor forward	RINB(15)						
	Left rear motor reverse	RINA(14)						
Right Motor	Right front motor forward	LED10						
	Right front motor reverse	LED11						
	Right rear motor forward	LED8						
	Right rear motor reverse	LED9						
Servo	Control S1	LEDO			A0			
	Control S2	LED1						
	Control S3	LED2						
	Control S4	S1(3)						
LOGO light	Control bluelight	LED7			A1			
Tarcking sensor	Left tracking sensor							
	Middle tracking sensor							
	Right tracking sensor							
Ultrasonic sensor	Ultrasonic Echo		Uno board drive directly		12			
	Ultrasonic RGB light							
Key	K1				11			
IR	IR control							
Bluetooth interface	RX							
	TX							
On board RGB Light	RGB Light on expansion board				6			
Buzzer	Control buzzer							



4. About code

For the code of this course, please refer to:

[**Ultrasonic_RGB_change_color.ino**](#) in the **Ultrasonic_RGB_change_color** folder.

```
#include <Wire.h>
#include <Adafruit_PWMServoDriver.h>
Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver(0x40);

#include "RGBLed.h"
#define RGB_GREEN    0xFF0000    //Define different color
#define RGB_RED     0x00FF00
#define RGB_BLUE    0x0000FF
#define RGB_YELLOW   0xFFFF00
#define RGB_PURPLE  0x00FFFF
#define RGB_WHITE   0xFFFFFFFF
#define RGB_OFF     0x00000000

const int RgbPin = 12;      //Define pin of Ultrasonic RGB light
RGBLed mRgb(RgbPin,2);

void setup()
{
    pinMode(RgbPin,OUTPUT);
    mRgb.setColor(1,RGB_OFF);
    mRgb.setColor(2,RGB_OFF);
    mRgb.show();
    pwm.begin();
    pwm.setPWMFreq(60); // Analog servos run at ~60 Hz updates
    LOGO_breathing_light(255, 40, 5); //Gradually light the blue light of the
Yhaboom_LOGO
}

void LOGO_breathing_light(int brightness, int time, int increment)
```

```

{
    if (brightness < 0)
    {
        brightness = 0;
    }
    if (brightness > 255)
    {
        brightness = 255;
    }
    for (int b = 0; b < brightness; b += increament)
    {
        int newb = map(b, 0, 255, 0, 4095);
        pwm.setPWM(7, 0, newb);
        delay(time);
    }
}

void loop()
{
    mRgb.setColor(1,RGB_RED); //There are two RGB light on the Ultrasonic
module No.1 and No.2
    mRgb.setColor(2,RGB_GREEN);
    mRgb.show();
    delay(500);
    mRgb.setColor(1,RGB_BLUE);
    mRgb.setColor(2,RGB_YELLOW);
    mRgb.show();
    delay(500);
    mRgb.setColor(1,RGB_PURPLE);
    mRgb.setColor(2,RGB_WHITE);
    mRgb.show();
    delay(500);
    mRgb.setColor(1,RGB_OFF); //Turn off two RGB light on the Ultrasonic
module
    mRgb.setColor(2,RGB_OFF);
    mRgb.show();
    delay(500);

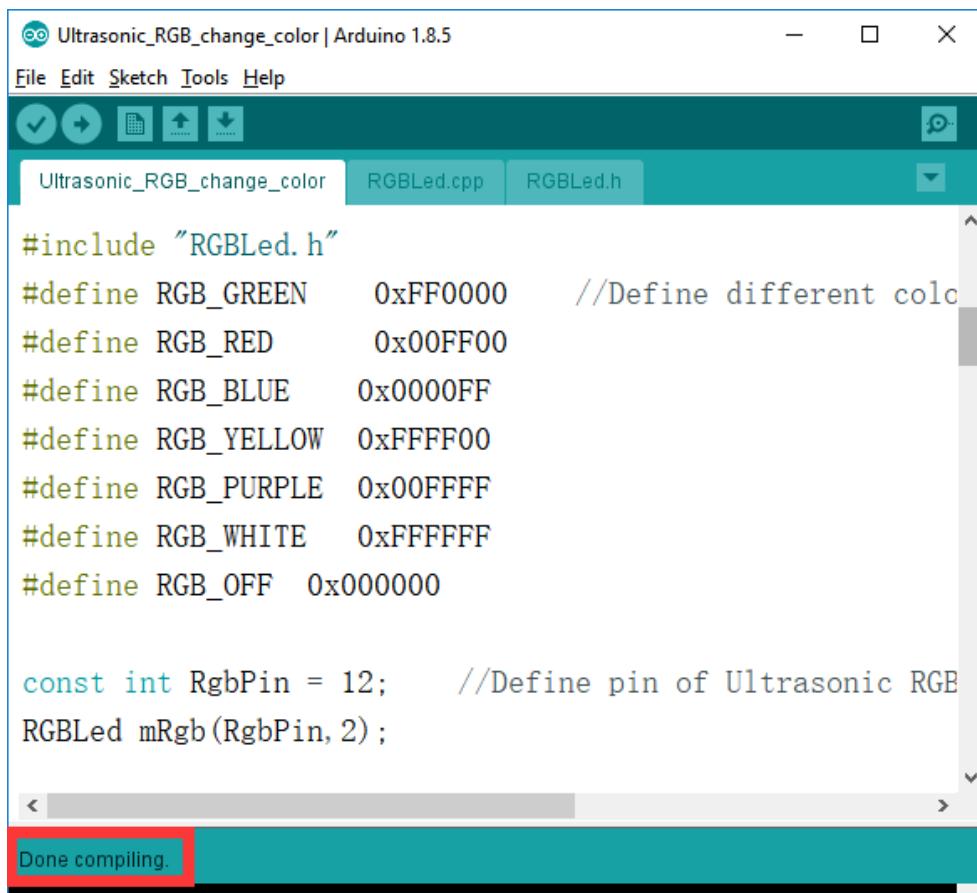
}

```

5. Compiling and downloading code

5.1 We need to open the **Ultrasonic_RGB_change_color.ino** file by Arduino IDE software. Then click “√” under the menu bar to compile the code, and wait for the word “**Done compiling** ” in the lower left corner, as shown in the figure

below.



The screenshot shows the Arduino IDE interface with the title bar "Ultrasonic_RGB_change_color | Arduino 1.8.5". The menu bar includes File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with icons for save, upload, and other functions. The main workspace displays the following code:

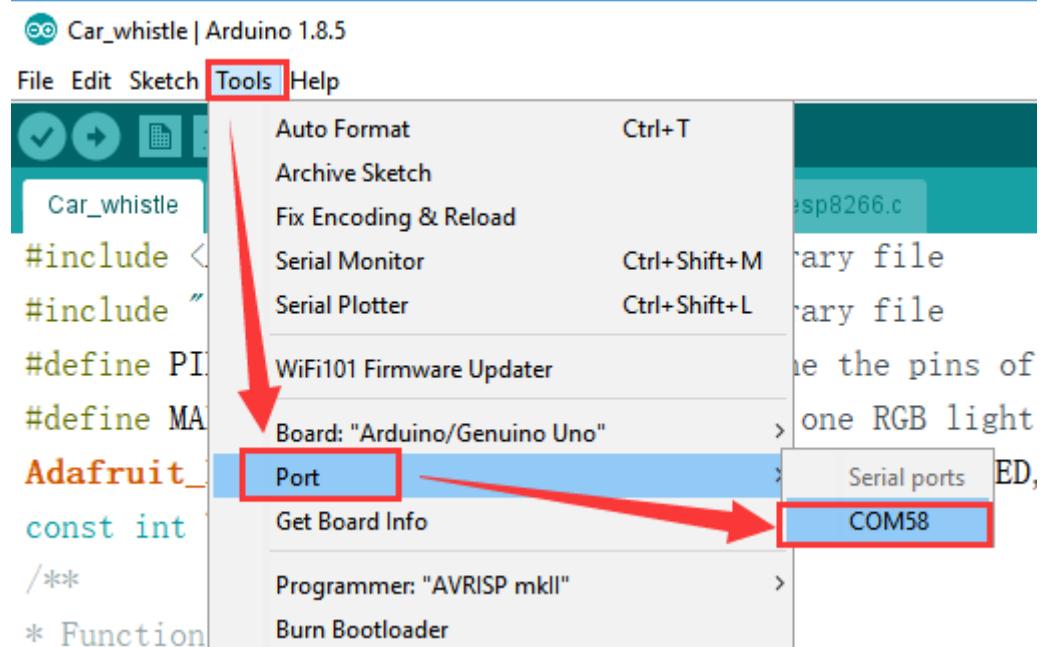
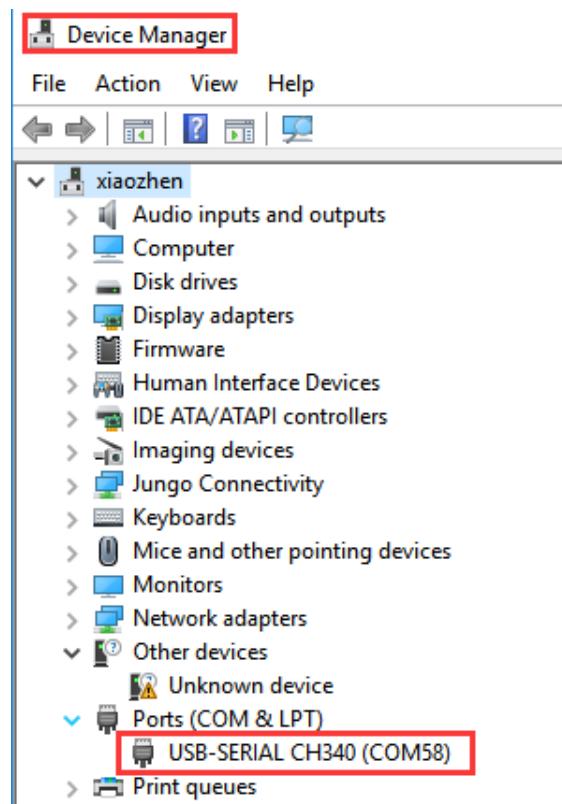
```
#include "RGBLed.h"

#define RGB_GREEN    0xFF0000      //Define different colors
#define RGB_RED     0x00FF00
#define RGB_BLUE    0x0000FF
#define RGB_YELLOW  0xFFFF00
#define RGB_PURPLE  0x00FFFF
#define RGB_WHITE   0xFFFFFFFF
#define RGB_OFF     0x00000000

const int RgbPin = 12;      //Define pin of Ultrasonic RGB
RGBLed mRgb(RgbPin, 2);
```

In the status bar at the bottom, the message "Done compiling." is displayed, which is highlighted with a red rectangular box.

5.2 In the menu bar of Arduino IDE, we need to select 【Tools】---【Port】--- selecting the port that the serial number displayed by the device manager just now, as shown in the figure below.



5.3 After the selection is completed, you need to click “→”under the menu bar to upload the code to the UNO board. When the word “**Done uploading**” appears in the lower left corner, the code has been successfully uploaded to the UNO board, as shown in the figure below.

The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** Ultrasonic_RGB_change_color | Arduino 1.8.5
- Menu Bar:** File Edit Sketch Tools Help
- Toolbars:** Standard toolbar with icons for file operations.
- Sketch Tabs:** Ultrasonic_RGB_change_color (selected), RGBLed.cpp, RGBLed.h
- Code Area:** C++ code for the sketch, including header includes and color definitions.

```
#include <Adafruit_PWMServoDriver.h>
Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver(0x40);

#include "RGBLed.h"

#define RGB_GREEN    0xFF0000      //Define different colors
#define RGB_RED      0x00FF00
#define RGB_BLUE     0x0000FF
#define RGB_YELLOW   0xFFFF00
#define RGB_PURPLE   0x00FFFF
#define RGB_WHITE    0xFFFFFFFF
#define RGB_OFF      0x000000
```
- Status Bar:** Done uploading.

6. Experimental phenomena

After the program is downloaded, we can see that RGB light of the ultrasonic module will change color red and green --> blue and yellow --> purple and white --> off.