runlinc Project 105 I12:

RGB Color Control (E32W Version)

Contents

Introduction

Problem

We want to be able to change the color of the RGB lights by controlling the ratio of red, blue and green.

Background

PWM stands for Pulse Width Modulation. It is a technique for modulating analog signals and is commonly used to control voltage, current or power in electronic devices.

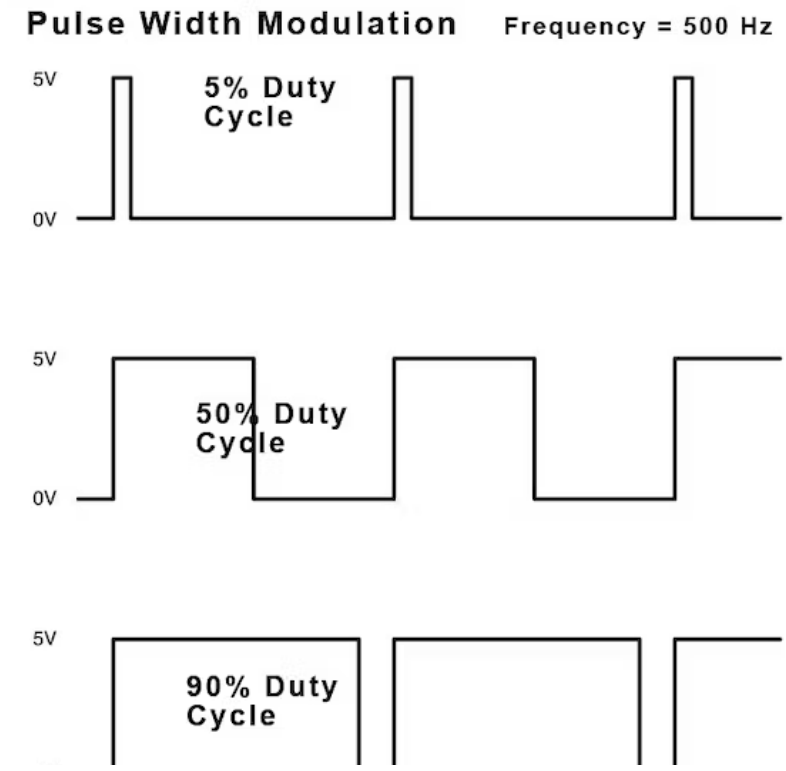


Figure 1PWM Frequency

PWM output is specified for the runlinc E32W using the setPin(PortName,DutyCycle); instruction. The duty cycle is specified using an 8 bit value between 0 to 255, where 0 is 0% duty cycle and 255 is 100% duty cycle.

PWM is used in a wide variety of applications such as electronic devices, motor control, LED dimming and more. In electronic devices, PWM can be used to regulate the power of a circuit so that the device operates at different voltage or current levels.

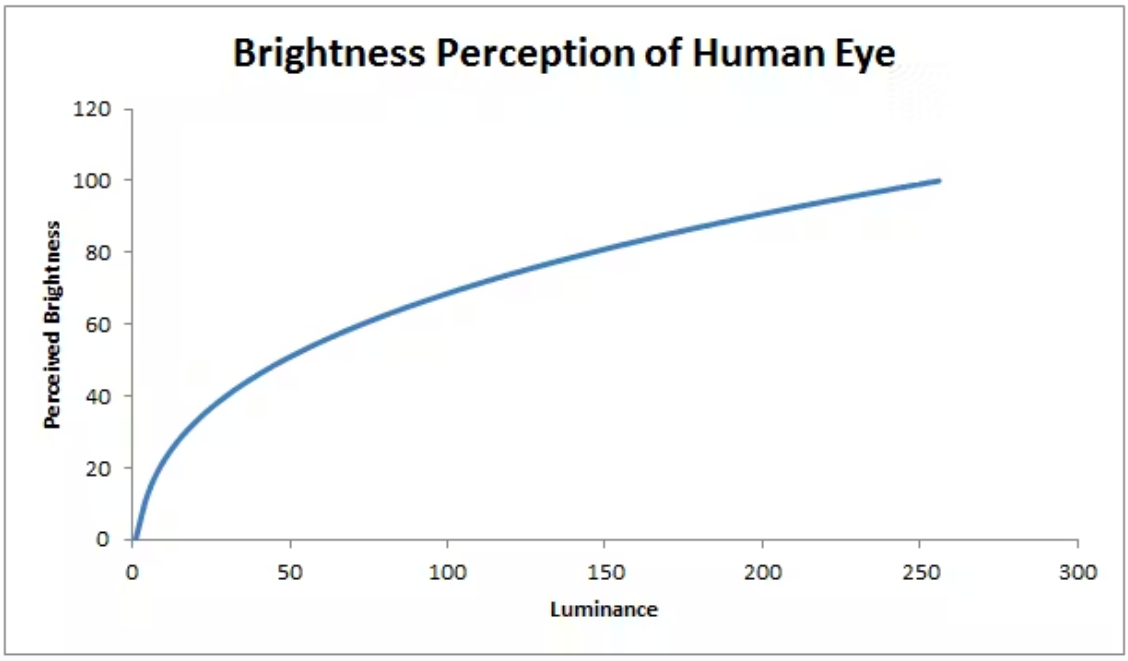


Figure 2 The energy output of the LED is proportional to the PWM duty cycle

Ideas

How do we change the RGB color scale? How to display the percentage of each color? How to change the color of RGB light automatically?

Plan

We will use the control bar on the web page to manipulate the RGB colors blue, red and green. We need to drag the control bar to change the percentage of different colors in order to change the RGB colors. The percentage of each color will be displayed next to

the control bar.

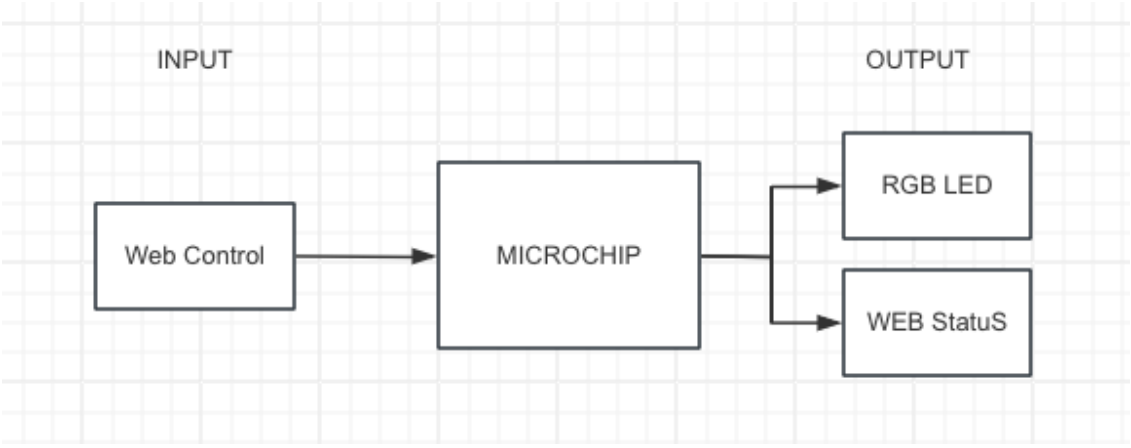


Figure 3 I/O Plan for this project.

runlinc Background

runlinc is a web page inside a Wi-Fi chip. The programming is done inside the browsers compare to programming inside a chip. The runlinc web page inside the Wi-Fi chip will command the microchips to do sensing, control, data logging Internet of Things (IoT). It can predict and command.

Part A1: Design the Circuit on runlinc

Note: Refer to runlinc Wi-Fi Setup Guide document to connect to runlinc

Use the left side of the runlinc web page to construct an input/output (I/O).

For port D05 name it Negative and set it as DIGITAL_OUT.

For port D18 name it RedLED and set it as PWM.

For port D19 name it GreenLED and set it as PWM.

For port D21 name it BlueLED and set it as PWM.

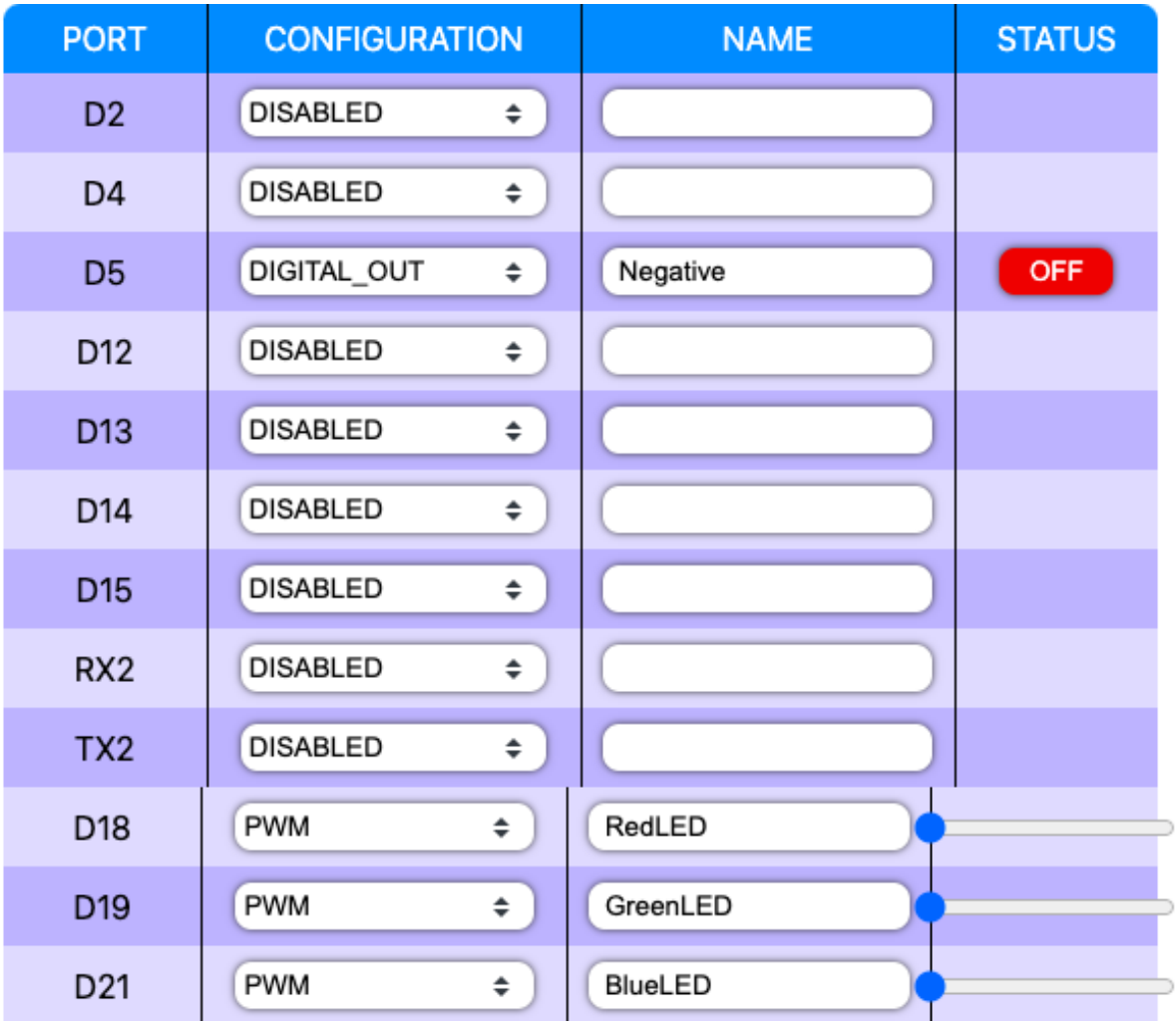


Figure 4: I/O configuration connections.

Part B1: Build the Circuit

Use the E32 board to connect the hardware. For this project we are using both the left and right I/O ports, with **negative port (-ve)** on the outer side, **positive port (+ve)** on the middle and **signal port (s)** on the inner side (as shown below).

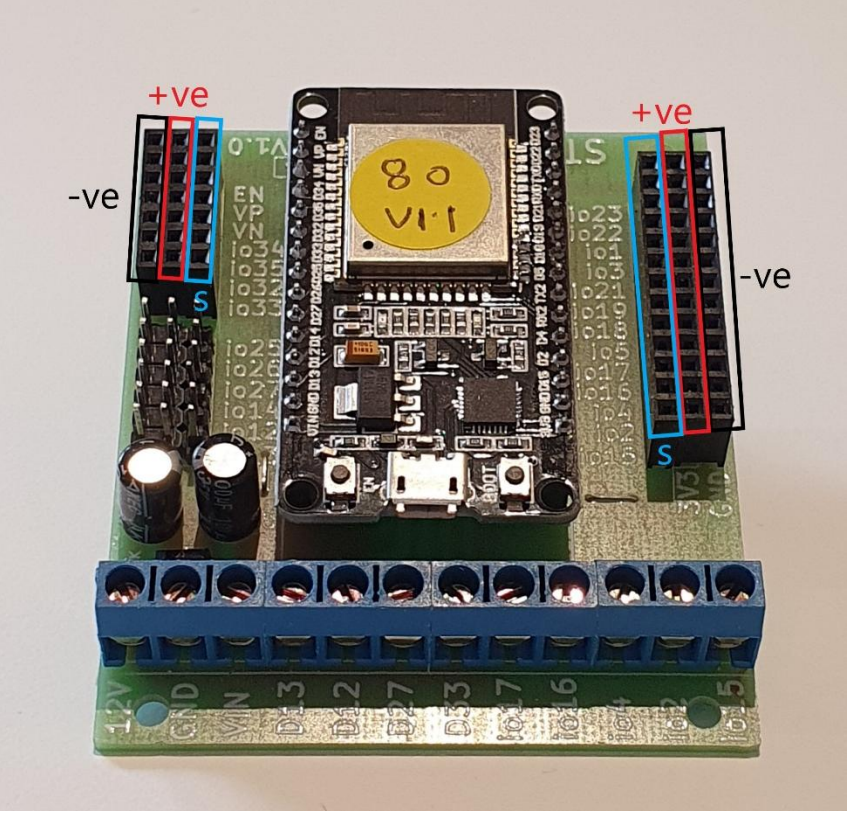


Figure 5: Negative, Positive and Signal port on the E32 board.

There are two I/O part we are using for this project, a DC motor with fan blade, and a 4-pin RGB LED module (KY-016), their respective pins are shown in the figure below.

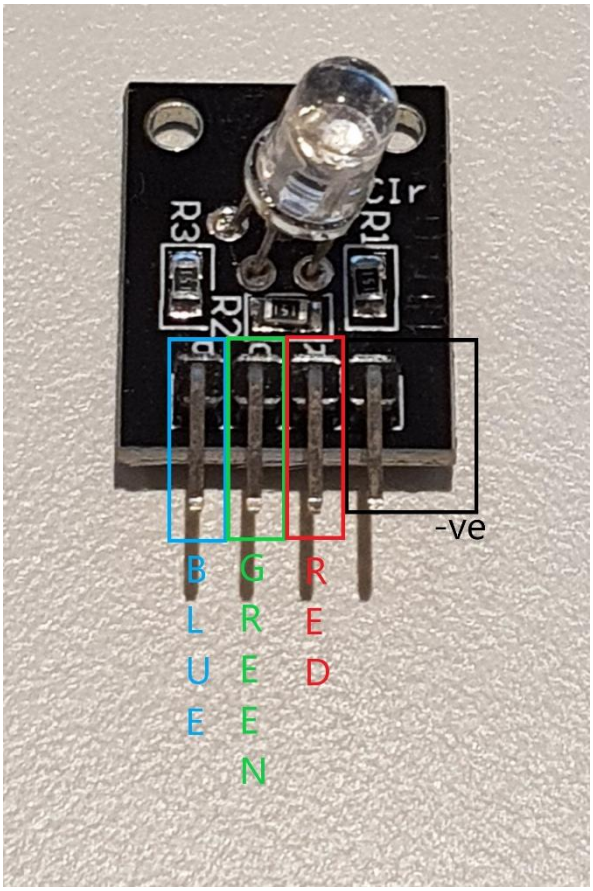


Figure 6: 4-Pin RGB LED with respective pins indicated.

Wiring Instructions

Since we only have 3-pins port slots, we need to plug in all 4 pins on the RGB LED into the signal ports.

a.) Plug in the negative pin on the 4-pins RGB LED into port io5, the rest should be in horizontal order and plug into ports io18, io19, io21. Please refer to **Figure 7** for more detail.

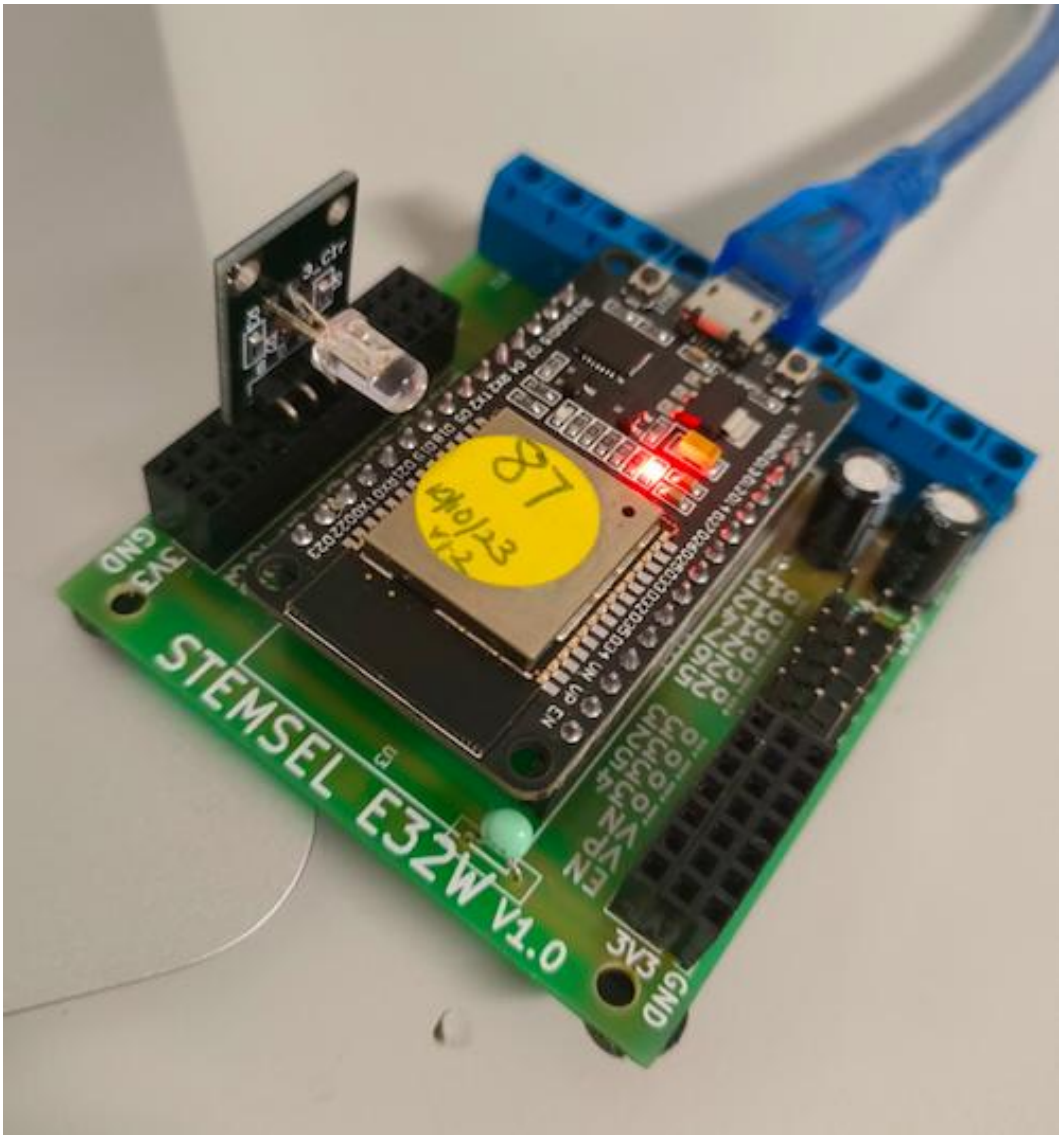


Figure 7: E32W board connection with RGB module (side view).

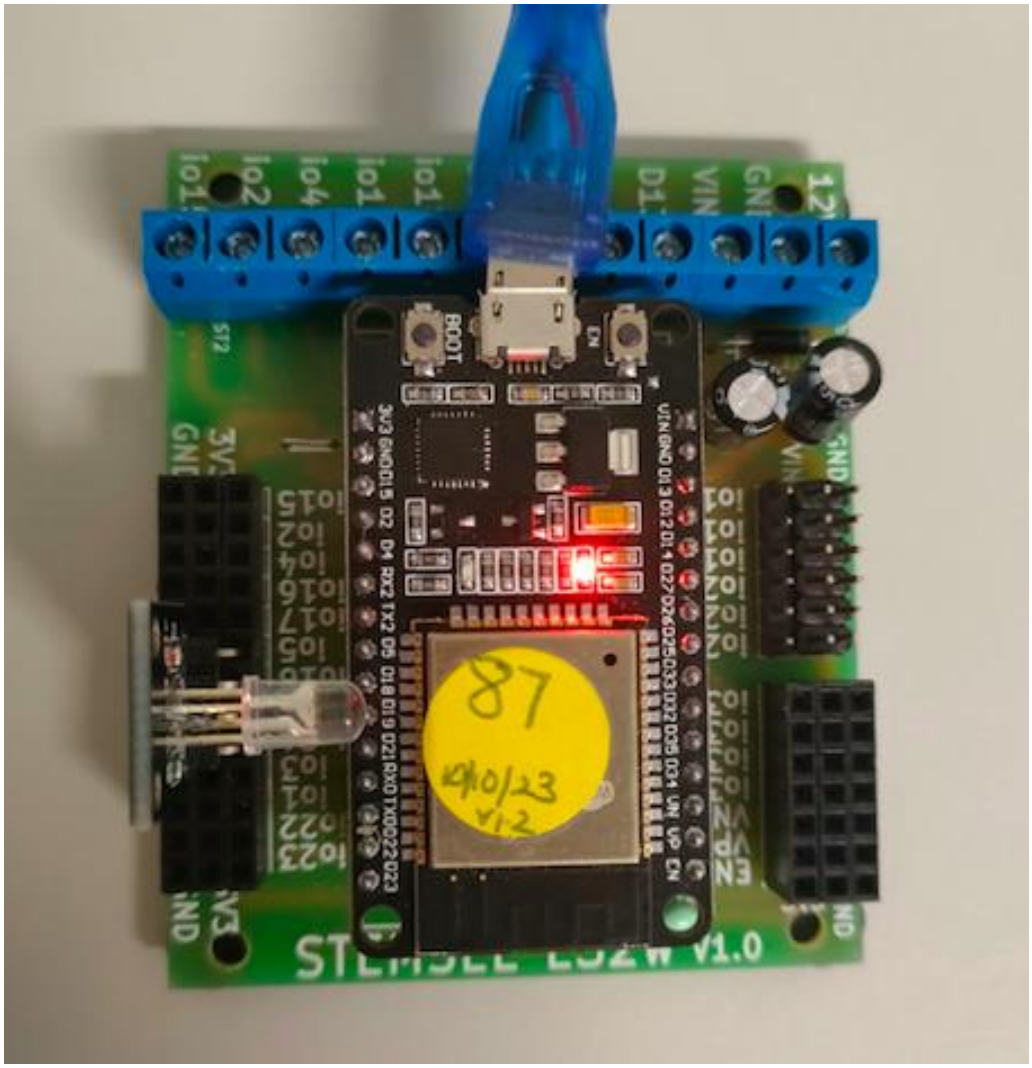


Figure 8: E32W board connection with both I/O parts connected (top view).

Part C1: Program the Circuit

HTML:

Before we program the logic into the E32W board, we will first set up an HTML page.

1. Set up the text of the page to be centred:

```
<div style="text-align:center">
<h3>RGB color Control</h3>
</div>
```

2. Within the tags we will have a control bar for each colour and also display the percentage of each color next to the control bar.:

```
Red LED
<br>
<input type="range" id="RedLed" min="0" max="100" step="1" value="0">
<font id="RedPer">Loading...</font>%
<br>
<br>
Green LED
<br>
<input type="range" id="GreenLed" min="0" max="100" step="1"value="0">
<font id="GreenPer">Loading...</font> %
<br>
<br>
Blue LED
<br>
<input type="range" id="BlueLed" min="0" max="100" step="1" value="0">
<font id="BluePer">Loading...</font> %
</div>
```

JavaScript Loop:

Then we will program the code into the E32W board using the JavaScript loop.

If the code doesn't work, check spelling as well as case. It is also not recommended to copy the code directly, as this may cause the code to break into new lines and not work.

1. First we need to enter the values from the control bar into the PWM.

```
Red = document.getElementById("RedLed").value;
setPin(RedLED,Red);
```

2. The resulting data is then displayed in a web page.

```
Red = document.getElementById("RedLed").value;
document.getElementById("RedPer").innerHTML = Red;
setPin(RedLED,Red);
```

3. Then we need to do the same for the remaining two colors.

```
Green = document.getElementById("GreenLed").value;
document.getElementById("GreenPer").innerHTML = Green;
setPin(GreenLED,Green);
Blue = document.getElementById("BlueLed").value;
document.getElementById("BluePer").innerHTML = Blue;
setPin(BlueLED,Blue);
```

Final Code:

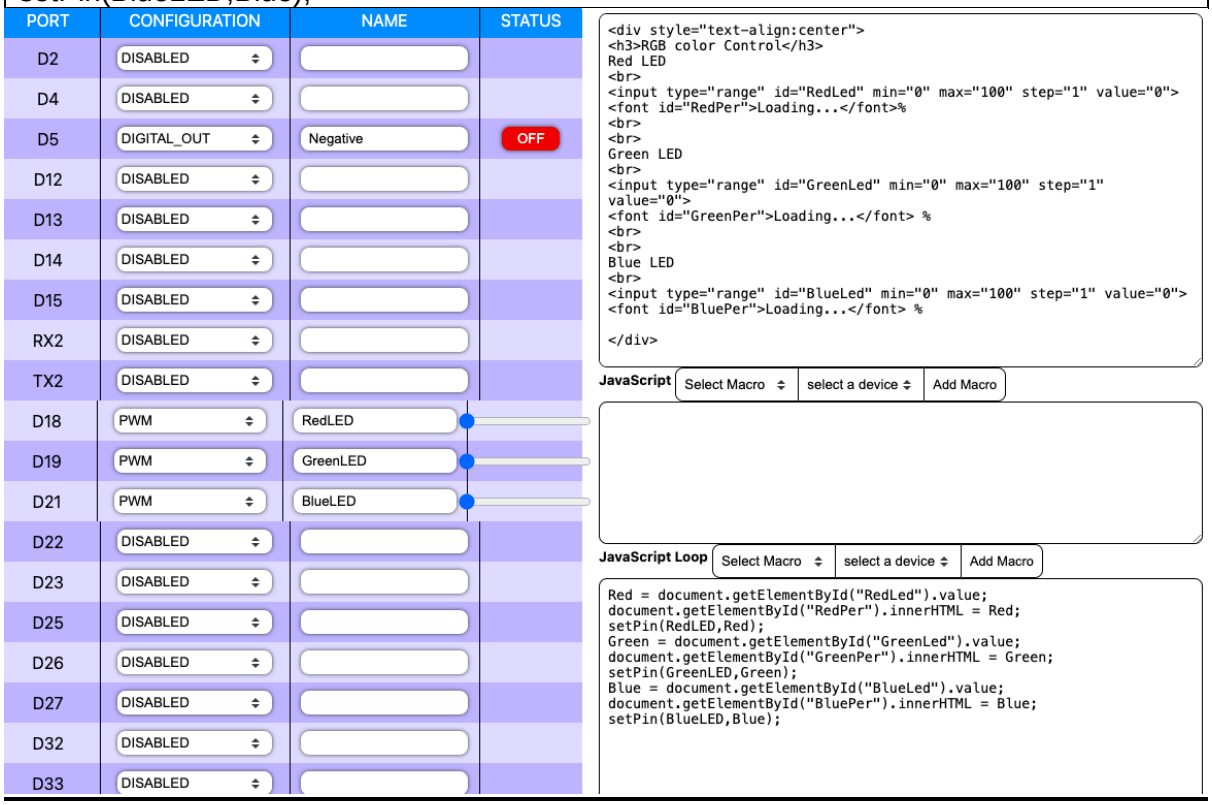
The final code for **HTML** block:

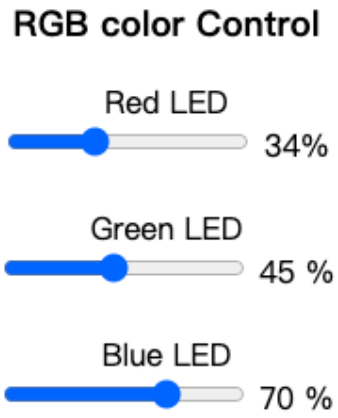
```
<div style="text-align:center">
<h3>RGB color Control</h3>
Red LED
<br>
<input type="range" id="RedLed" min="0" max="100" step="1" value="0">
<font id="RedPer">Loading...</font>%
<br>
<br>
Green LED
<br>
<input type="range" id="GreenLed" min="0" max="100" step="1" value="0">
<font id="GreenPer">Loading...</font> %
<br>
<br>
Blue LED
<br>
<input type="range" id="BlueLed" min="0" max="100" step="1" value="0">
```

```
<font id="BluePer">Loading...</font> %

</div>
```

The final code for **JavaScript** block:

```
Red = document.getElementById("RedLed").value;
document.getElementById("RedPer").innerHTML = Red;
setPin(RedLED,Red);
Green = document.getElementById("GreenLed").value;
document.getElementById("GreenPer").innerHTML = Green;
setPin(GreenLED,Green);
Blue = document.getElementById("BlueLed").value;
document.getElementById("BluePer").innerHTML = Blue;
setPin(BlueLED,Blue);
```





Challenge

Plan

We're going to have the lights change color automatically. Let the red light start coming on first, and when it reaches 50%, add the green light. When the green light reaches 50%, add the blue light. When the red light reaches 100%, stop increasing the percentage of red light. The same goes for the green light, if the percentage of all three colors reaches 100%, clear all three colors to zero.

Part A2: Program the Circuit (Challenge)

We don't have to change the HTML code. There is a need to make changes to the javascript code.

JavaScript Loop:

First we need to take the value of the control bar and name it with a relative color.

```
Red = parseInt(document.getElementById ("RedLed").value);

Green = parseInt(document.getElementById ("GreenLed").value) ;

Blue = parseInt(document.getElementById ("BlueLed").value) ;
```

At the same time, we have several situations, the red light is on but did not reach 50%; the red light reaches 50% of the green light starts to light; the red light reaches 100%, the green light reaches 50%, the blue light starts to light; the green light red light reaches 100%, the blue light reaches 50%; and finally, all the lights reach 100%.

First of all, let's consider the first situation, the red light is on but did not reach 50%, at this time only the color of the red light is increasing.

```
while(Red < 50&&Green<100&&Blue<100){
Red = Red + 10;
setPin(RedLED, Red) ;
document.getElementById("RedLed").value = Red;
document.getElementById("RedPer").innerHTML = Red;
await mSec(100);
}
```

In the second case, the red light reaches 50% and the green light starts to light up;

```
while(Red>=50&&Green<50&&Blue<100){
Red = Red + 10;
setPin(RedLED, Red) ;
document.getElementById("RedLed").value = Red;
document.getElementById("RedPer").innerHTML = Red;
Green =Green + 10;
setPin(GreenLED, Green) ;
document.getElementById("GreenLed").value = Green;
document.getElementById("GreenPer").innerHTML = Green;
await mSec(100);
}
```

Then there is the third scenario, where the red light reaches 100%, the green light reaches 50%, and the blue light starts to come on;

```
while(Red==100&&Green>=50&&Green<100&&Blue<100){
Green =Green + 10;
setPin(GreenLED, Green) ;
document.getElementById("GreenLed").value = Green;
document.getElementById("GreenPer").innerHTML = Green;
Blue = Blue +10;
setPin(BlueLED, Blue) ;
document.getElementById ("BlueLed").value = Blue;
document.getElementById ("BluePer").innerHTML = Blue;
await mSec(100);
}
```

The fourth scenario is that the green and red lights reach 100% and the blue light reaches 50%; only the percentage of the blue light is increasing.

```
while(Red==100 && Green >= 100 && Blue<100){
Blue = Blue +10;
setPin(BlueLED, Blue) ;
document.getElementById ("BlueLed").value = Blue;
```

```
document.getElementById ("BluePer").innerHTML = Blue;
await mSec(100);
}
```

Finally, all the lights accounted for 100%, at this time all the colors accounted for zero, in other words, the RGB lights off.

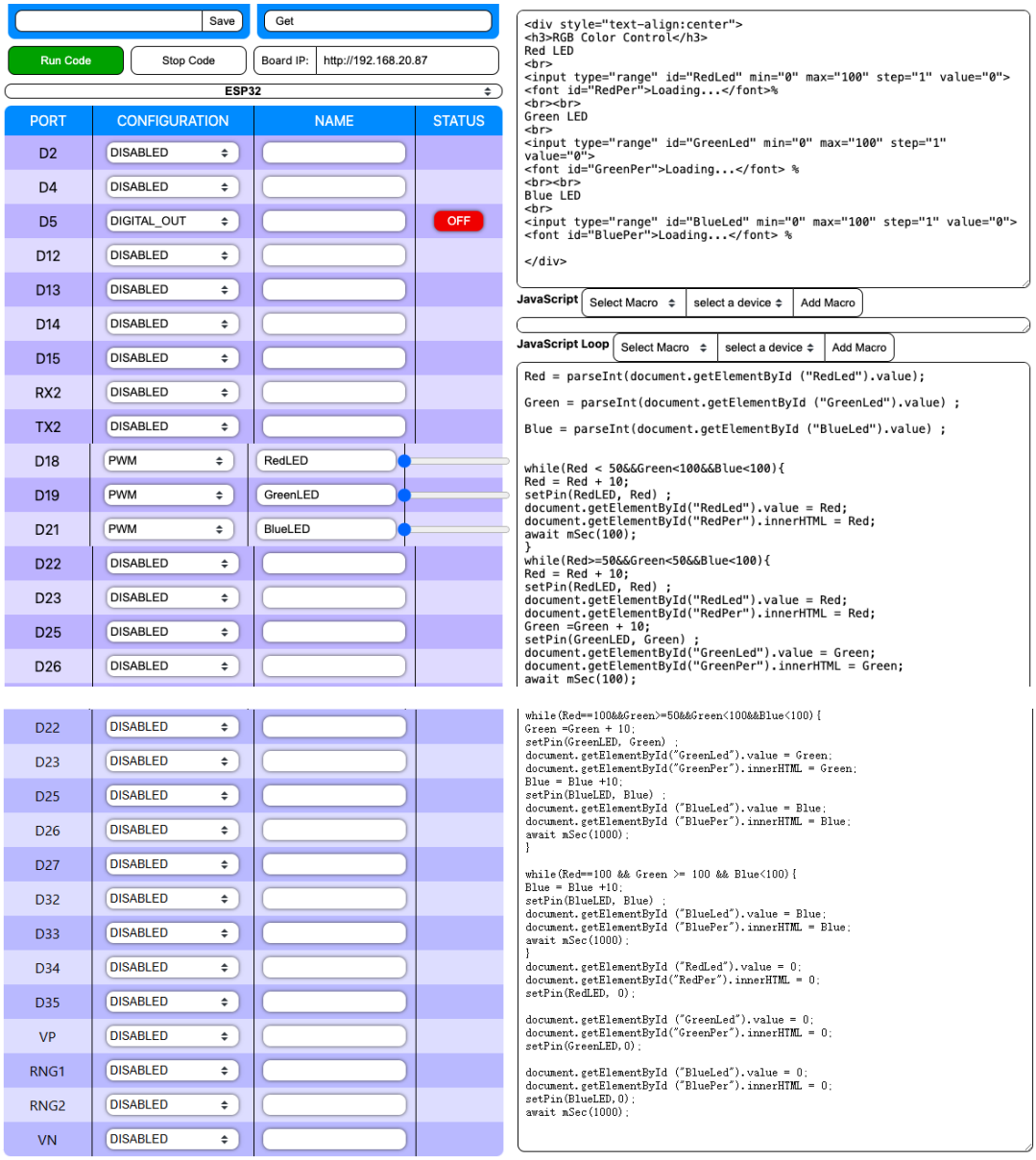
```
document.getElementById ("RedLed").value = 0;
document.getElementById("RedPer").innerHTML = 0;
setPin(RedLED, 0);

document.getElementById ("GreenLed").value = 0;
document.getElementById("GreenPer").innerHTML = 0;
setPin(GreenLED,0);

document.getElementById ("BlueLed").value = 0;
document.getElementById ("BluePer").innerHTML = 0;
setPin(BlueLED,0);
await mSec(1000);
```

Final Code:

```
Red = parseInt(document.getElementById ("RedLed").value);

Green = parseInt(document.getElementById ("GreenLed").value) ;

Blue = parseInt(document.getElementById ("BlueLed").value) ;


while(Red < 50&&Green<100&&Blue<100){
Red = Red + 10;
setPin(RedLED, Red) ;
document.getElementById("RedLed").value = Red;
document.getElementById("RedPer").innerHTML = Red;
await mSec(100);
}
while(Red>=50&&Green<50&&Blue<100){
Red = Red + 10;
setPin(RedLED, Red) ;
document.getElementById("RedLed").value = Red;
document.getElementById("RedPer").innerHTML = Red;
Green =Green + 10;
setPin(GreenLED, Green) ;
document.getElementById("GreenLed").value = Green;
document.getElementById("GreenPer").innerHTML = Green;
await mSec(100);
}

while(Red==100&&Green>=50&&Green<100&&Blue<100){
```

```
Green =Green + 10;
setPin(GreenLED, Green) ;
document.getElementById("GreenLed").value = Green;
document.getElementById("GreenPer").innerHTML = Green;
Blue = Blue +10;
setPin(BlueLED, Blue) ;
document.getElementById ("BlueLed").value = Blue;
document.getElementById ("BluePer").innerHTML = Blue;
await mSec(100);
}

while(Red==100 && Green >= 100 && Blue<100){
Blue = Blue +10;
setPin(BlueLED, Blue) ;
document.getElementById ("BlueLed").value = Blue;
document.getElementById ("BluePer").innerHTML = Blue;
await mSec(100);
}


document.getElementById ("RedLed").value = 0;
document.getElementById("RedPer").innerHTML = 0;
setPin(RedLED, 0);

document.getElementById ("GreenLed").value = 0;
document.getElementById("GreenPer").innerHTML = 0;
setPin(GreenLED,0);

document.getElementById ("BlueLed").value = 0;
document.getElementById ("BluePer").innerHTML = 0;
setPin(BlueLED,0);
await mSec(1000);
```

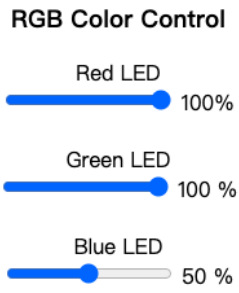
Expected Result (Challenge part):




```
<div style="text-align:center">
<h3>RGB Color Control</h3>
Red LED
<br>
<input type="range" id="RedLed" min="0" max="100" step="1" value="0">
<font id="RedPer">Loading...</font>%
<br><br>
Green LED
<br>
<input type="range" id="GreenLed" min="0" max="100" step="1"
value="0">
<font id="GreenPer">Loading...</font> %
<br><br>
Blue LED
<br>
<input type="range" id="BlueLed" min="0" max="100" step="1" value="0">
<font id="BluePer">Loading...</font> %

</div>
```

```
Red = parseInt(document.getElementById ("RedLed").value);

Green = parseInt(document.getElementById ("GreenLed").value) ;

Blue = parseInt(document.getElementById ("BlueLed").value) ;


while(Red < 50&&Green<100&&Blue<100){
Red = Red + 10;
setPin(RedLED, Red) ;
document.getElementById("RedLed").value = Red;
document.getElementById("RedPer").innerHTML = Red;
await mSec(100);
}
while(Red>=50&&Green<50&&Blue<100){
Red = Red + 10;
setPin(RedLED, Red) ;
document.getElementById("RedLed").value = Red;
document.getElementById("RedPer").innerHTML = Red;
Green =Green + 10;
setPin(GreenLED, Green) ;
document.getElementById("GreenLed").value = Green;
document.getElementById("GreenPer").innerHTML = Green;
await mSec(100);
```

```
while(Red==100&&Green>=50&&Green<100&&Blue<100){
Green =Green + 10;
setPin(GreenLED, Green) ;
document.getElementById("GreenLed").value = Green;
document.getElementById("GreenPer").innerHTML = Green;
Blue = Blue +10;
setPin(BlueLED, Blue) ;
document.getElementById ("BlueLed").value = Blue;
document.getElementById ("BluePer").innerHTML = Blue;
await mSec(1000);
}

while(Red==100 && Green >= 100 && Blue<100){
Blue = Blue +10;
setPin(BlueLED, Blue) ;
document.getElementById ("BlueLed").value = Blue;
document.getElementById ("BluePer").innerHTML = Blue;
await mSec(1000);
}
document.getElementById ("RedLed").value = 0;
document.getElementById("RedPer").innerHTML = 0;
setPin(RedLED, 0);

document.getElementById ("GreenLed").value = 0;
document.getElementById("GreenPer").innerHTML = 0;
setPin(GreenLED,0);

document.getElementById ("BlueLed").value = 0;
document.getElementById ("BluePer").innerHTML = 0;
setPin(BlueLED,0);
await mSec(1000);
```



Summary

The color space seen can usually be expressed by the three basic colors, which are called the "three primary colors", that is, red, green and blue. rgb lights can reflect this, by adjusting the intensity of the three colors, you can achieve rgb light color change.