



# runlinc Online Project A3: Traffic Lights (runlinc Online Version)

---

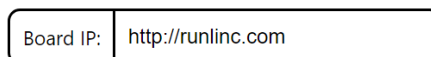
## Contents

Introduction .....	1
Program the project.....	2
Challenge .....	5
Summary .....	6

This project uses runlinc online which can be found here:

<http://www.runlinc.com/online/control.html>

Make sure to right-click the link and click 'open in new tab'.



Check that your board IP says <http://runlinc.com> as shown to the right. If it does not try clearing your site data.

This project does not require a runlinc Wi-Fi Inventors' Kit.

## Introduction

### Problem

The correct order and timing of instructions are critical. To investigate this, we want to program a set of traffic lights by using runlinc online.

### Background

By learning runlinc, you can learn how to program microchips and tell them what to do. Microchips can monitor devices and warn people like when your laptop battery is low, it will tell you that you need to plug the cable in. However, they can also control people's behaviour more directly, such as electronic road signs and traffic lights. But the microchips must send the right message, otherwise, there can be problems, like turning on the green lights of both directions at the same time.

It is not only important to consider the correct sequence, but also it is important to think about how long each colour should be on for. For example, if the green light was only on for one second, the cars would not have enough time to get across the intersection. Microchips should be programmed so that the function of the traffic lights are correct to avoid accidents.

### Ideas

What kind of things will we need for our traffic light? What if the pedestrians want to cross? How many traffic lights are there when two roads intersect? What is the correct

sequence of the lights? What if the green light was only on for 1 second, would that be long enough for the cars to cross? How long should each light be on for?

## Plan

As we all know, all traffic lights have a red light on the top, yellow light on the middle and green light on the bottom. Why are the lights safer if the red light is at the top? The correct sequence of lights is green -> yellow -> red, then back to green again. For timing in Australia, the law states that the yellow light should be on for at least 3 seconds. If we turn on the green light for 7 seconds to let the cars go across the intersection, how long should the red light be on for? Well, while the traffic light facing one way is red, the traffic light facing the other way will be green and then yellow. Therefore, our traffic lights will need to be red for  $7+3 = 10$  seconds.

## runlinc Background

runlinc online is a stand-alone version of runlinc, which is a web page inside a Wi-Fi chip. The programming is done in the browser and sent to the chip over Wi-Fi. 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.

## Program the project

### HTML

All of our code for this project will be written in the HTML box. The HTML we will write is as follows.

```
<h1>Traffic Lights</h1>

<script>
</script>
```

This starts with a heading for our page, put whatever you would like between the `<h1>` tags. Then it will insert a picture of the traffic lights. Three pictures are stored on runlinc.com which are accessed by putting their link (URL) in the `src` attribute.

Finally, we need a pair of `<script>` tags. This is another way you can link JavaScript to your HTML. We will write our JavaScript between these tags.

This is the JavaScript we will be using. (Remember it needs to go between the script tags).

```
function greenON(){
  document.getElementById("img").src="http://runlinc.com/online/picGreenLED.jpg";
}

function yellowON(){
  document.getElementById("img").src="http://runlinc.com/online/picYellowLED.jpg";
}

function redON(){
  document.getElementById("img").src="http://runlinc.com/online/picRedLED.jpg";
}

function animate(){
  greenON();
  setTimeout(function(){
    yellowON();
    setTimeout(function(){
      redON();
    }, 3000);
  }, 7000);
}

animate();

setInterval(function(){
  animate();
}, 20000);
```

The first three functions will replace the image with a different image depending on which colour light is on. These functions use the 'img' ID as a reference to the HTML element.

The animate function will animate our traffic lights. It will show the image with the green light on, wait 7 seconds, show the image with the yellow light on, wait 3 seconds, and then show the image with the red light on. This function uses setTimeout which waits an amount of time before running the code inside it.

e.g. This will wait for 3000 milliseconds (3 seconds) before running the code.

```
setTimeout(function(){
  //code to run
}, 3000);
```

The last part of our code will call the 'animate' function and set an interval to call it every 20000 milliseconds (20 seconds). Green is on for 7 seconds, yellow for 3, and red for 10 so 7+3+10 = 20 seconds. setInterval works in the same way as setTimeout except it keeps looping the code with the delay in between.

Final Code:

**runlinc V1.1** Copyright and International Patent Pending. All rights reserved.

**File**

Load File  Save

**Board**

Send  Get

Run Code  Stop Code

Board IP:

⊕STEMSEL

PORT	CONFIGURATION	NAME	STATUS
A3	DISABLED	<input type="text"/>	
B4	DISABLED	<input type="text"/>	
B6	DISABLED	<input type="text"/>	
C0	DISABLED	<input type="text"/>	
C1	DISABLED	<input type="text"/>	
C2	DISABLED	<input type="text"/>	
C3	DISABLED	<input type="text"/>	
C4	DISABLED	<input type="text"/>	
C5	DISABLED	<input type="text"/>	
C6	DISABLED	<input type="text"/>	
C7	DISABLED	<input type="text"/>	
GPIO0	DISABLED	<input type="text"/>	

Network Status: Active

**CSS**

---

**HTML**

```
<h1>Traffic Lights</h1>


<script>
function greenON(){
document.getElementById("img").src="http://runlinc.com/online/picGreenLED.jpg";
}

function yellowON(){
document.getElementById("img").src="http://runlinc.com/online/picYellowLED.jpg";
}

function redON(){
document.getElementById("img").src="http://runlinc.com/online/picRedLED.jpg";
}

function animate() {
greenON();
setTimeout(function () {
yellowON();
setTimeout(function () {
redON();
}, 3000);
}, 7000);
}

animate();
setInterval(function(){
animate();
}, 20000);
</script>
```

**JavaScript**

---

**JavaScript Loop**

---

Figure 4: HTML on runlinc IDE

## Traffic Lights

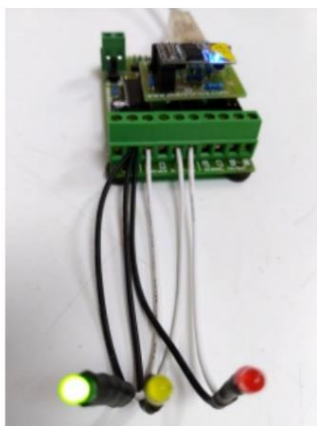


Figure 5: Final web page

## Challenge

Let's change our traffic lights so they use an HTML graphic instead of the images.

Replace the image element with the following code:

```
<svg height="1000" width="100">
  <rect height="300" width="100" rx="20" ry="20"/>
  <circle id="R" cx="50" cy="50" r="40" stroke="black" stroke-width="3" fill="firebrick"/>
</svg>
```

This code will create an SVG with a height of 1000 pixels and a width of 100 pixels. Within this SVG we have created a rectangle with a height of 300 pixels, a width of 100 pixels, and rounded corners (rx, ry) with a radius of 20 pixels. After the rectangle, we have a circle with ID 'R', x position of 50, y position of 50, a radius of 40, a black border (stroke) with width 3 and a fill of 'firebrick' (a dark red so the light looks turned off). This circle will be our red light.

Inside of this SVG, we also need two circles for the yellow and green lights. You will need to give each of them unique IDs, make sure that they are placed correctly, and give them appropriate dark fills so the lights look turned off. (Hint: try changing the cy value for positioning).

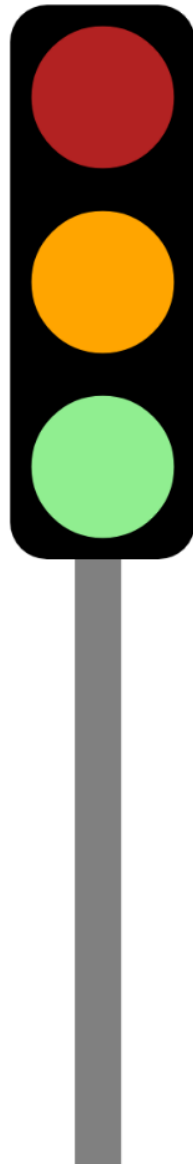
We also need a pole for the traffic light to stand on, so add another rectangle with height 500, width 25, grey fill, y position 300, and x position 35. (The x and y position attributes are just 'x', and 'y' for the rectangle).

Now we need to change our greenON, yellowON and redON functions so that they change the fill of the circles to a lighter colour, which will make the light appear on. We don't need the lines that changed the image anymore, so you can remove them from the functions. This line will 'turn on' the green light.

```
document.getElementById("G").style.fill = "lightgreen";
```

Also inside the greenON function, we need to turn off the light that was last on (red in this case). This is done the same way, just changing the fill back to the darker colour. (Hint: the "G" is the ID of the green circle, make sure what you put here in your code matches the ID of the circle you want to effect). Similarly, make changes to your yellowON and redON functions.

## Traffic Lights



**Figure 5:** Challenge web page

### Summary

runlinc can be used as a stand-alone programming environment, however, with the STEMSEL board, you would be able to add LEDs to this project. You would be able to program them to light up as the HTML traffic lights do.