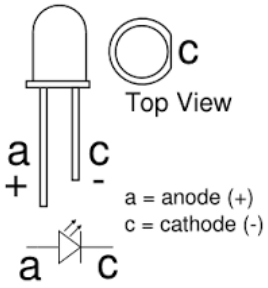
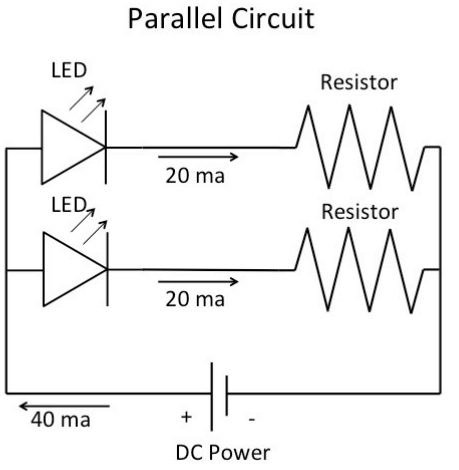
Model Railway Animation: Part 1, LEDs - Expanded
By David King

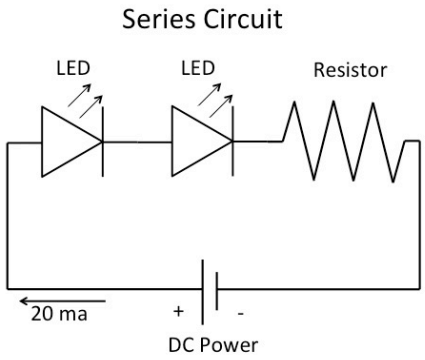
By now you are most likely ready to proceed past the simple Blink sketch so that is what we will do now. A couple of simple sketches we can create will be to simulate a simple grade crossing flasher and a traffic light.

A quick note about the LED itself is that you may have noticed that it may not have worked for you at first and there is a possible reason why this might of happened. The LEDs in kits have 2 leads with one being longer and the other shorter. The Long lead is the anode and it needs to be connected closest to the +5 volt power and the shorted lead is the cathode and should be located closest to the ground connection. We also use a resistor connected from the anode to +5 volts or from the cathode to ground. It doesn't matter which location the resistor is placed in the circuit just as long as it is in the circuit. If at first the LED doesn't light check to see if you have the LED wired up backwards. If it is backwards remove the LED and turn it around to swap those leads.



When using a LED with the output of Arduino Uno it is okay to 1 or 2 LED's to a single pin. The output from each pin has a maximum rating of about 40 milliamps and each LED connected can draw up to 20 milliamps. As you can see if you connect 2 LEDs to a single output you could be at the limit for that pin. With all of that said there is a way to connect 2 LEDs to a single pin that only uses the 20 milliamps and not the 40 milliamps. This can be accomplished by the way we connect the LEDs in the circuit by connecting the LEDs in parallel or series. This image should help make this a little clearer.





The Grade Crossing Flasher

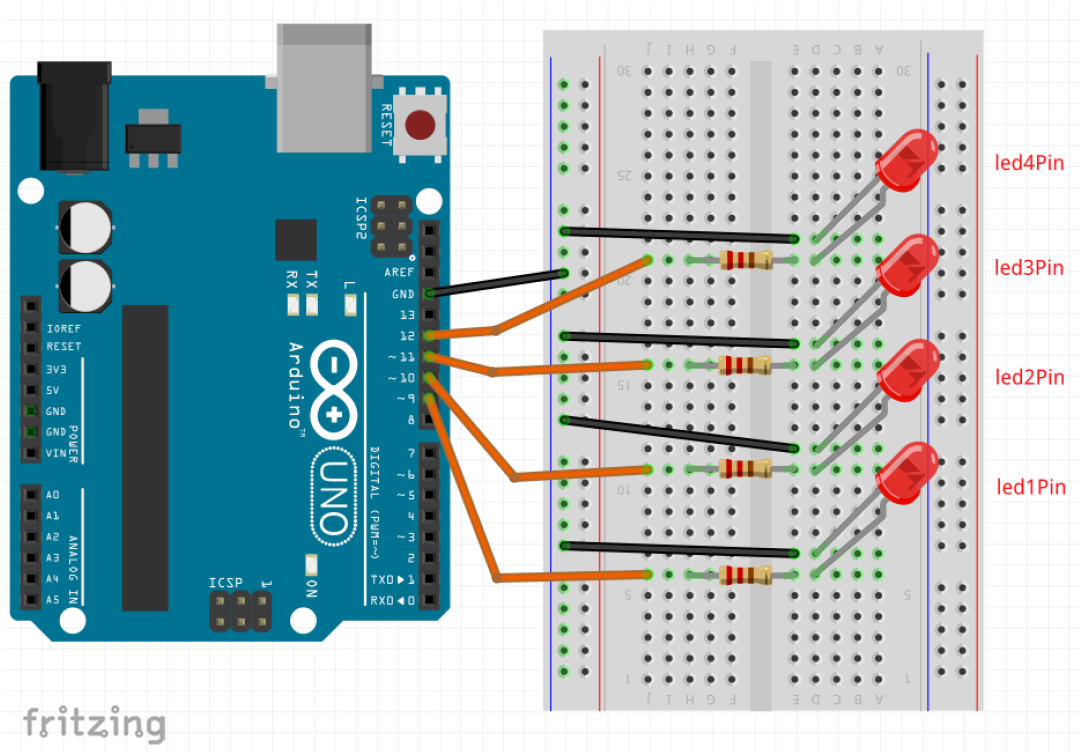
The grade crossing flasher is just a pair of LEDs or sets of pairs of LEDs that need to flash at a set rate of speed and each LED pair must have one LED on while the other is off. This is the alternating pattern of LEDs that we want. In the real world this alternating flashing happens at about a rate of 1/3rd a second for each of the LEDs. This is the time that we will use to keep it easy. Your completed sketch would be similar to my sketch shown below.

The_Grade_Flasher

```
// The_Grade_Flasher by David King
// This is a simple grade crossing flasher that controls the output of 2 pairs
// of LEDs, one pair for each side of the railway tracks.

// include any associated files in this location

// declare any variables needed in you file here
int led1Pin = 9;      // This is where I declare the pin assignments for my LEDs
int led2Pin = 10;
int led3Pin = 11;
int led4Pin = 12;

void setup() {
  // put your setup code here, to run once:
  pinMode(led1Pin, OUTPUT);     // This is where I set the pins as output devices
  pinMode(led2Pin, OUTPUT);
  pinMode(led3Pin, OUTPUT);
  pinMode(led4Pin, OUTPUT);

}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(led1Pin, HIGH);
  digitalWrite(led2Pin, LOW);
  digitalWrite(led3Pin, HIGH);
  digitalWrite(led4Pin, LOW);
  delay(333);

  digitalWrite(led1Pin, LOW);
  digitalWrite(led2Pin, HIGH);
  digitalWrite(led3Pin, LOW);
  digitalWrite(led4Pin, HIGH);
  delay(333);

}
```

In this sketch it is just a matter of adjusting the times from the original Blink sketch and adding the addition outputs for both pinMode and digitalWrite.

As seen in this sketch I have connected 4 LEDs to the Arduino Uno just to keep the wiring very simple. There are other methods of wiring the LEDs but this is simple to understand and easy to connect. I have created a Fitzing diagram to show you how to connect the wires to the LEDs along with using some resistors. Remember that the resistor values that were supplied with your kit my vary slightly from the one that I'm using in the diagram but that is okay as long as there are in a range of 220 to 560 ohms.



The Traffic Light

The basic traffic light that we see out in the real world is a very simple device that has 3 lights, red, amber and green, with one set facing the northbound traffic, one set facing the eastbound traffic, one set facing the southbound traffic and a set facing the westbound traffic. In most cases the eastbound and westbound traffic light sets working in the same pattern. As well the southbound and northbound traffic light sets working in the same pattern. We can create a simple little chart to display this pattern and make it easier to understand. In the chart I used NS for North/South and EW for East/West. Also I added an extra row at the bottom of the chart so that I could use this for the pin assignments on my Arduino Uno.

Step	NS Green	NS Amber	NS Red	EW Green	EW Amber	EW Red	Time
1	On	Off	Off	Off	Off	On	25 sec.
2	Off	On	Off	Off	Off	On	7 sec.
3	Off	Off	On	Off	Off	On	1 sec.
4	Off	Off	On	On	Off	Off	22 sec.
5	Off	Off	On	Off	On	Off	6 sec.
6	Off	Off	On	Off	Off	On	1 sec.
Uno Pin	7	8	9	10	11	12	

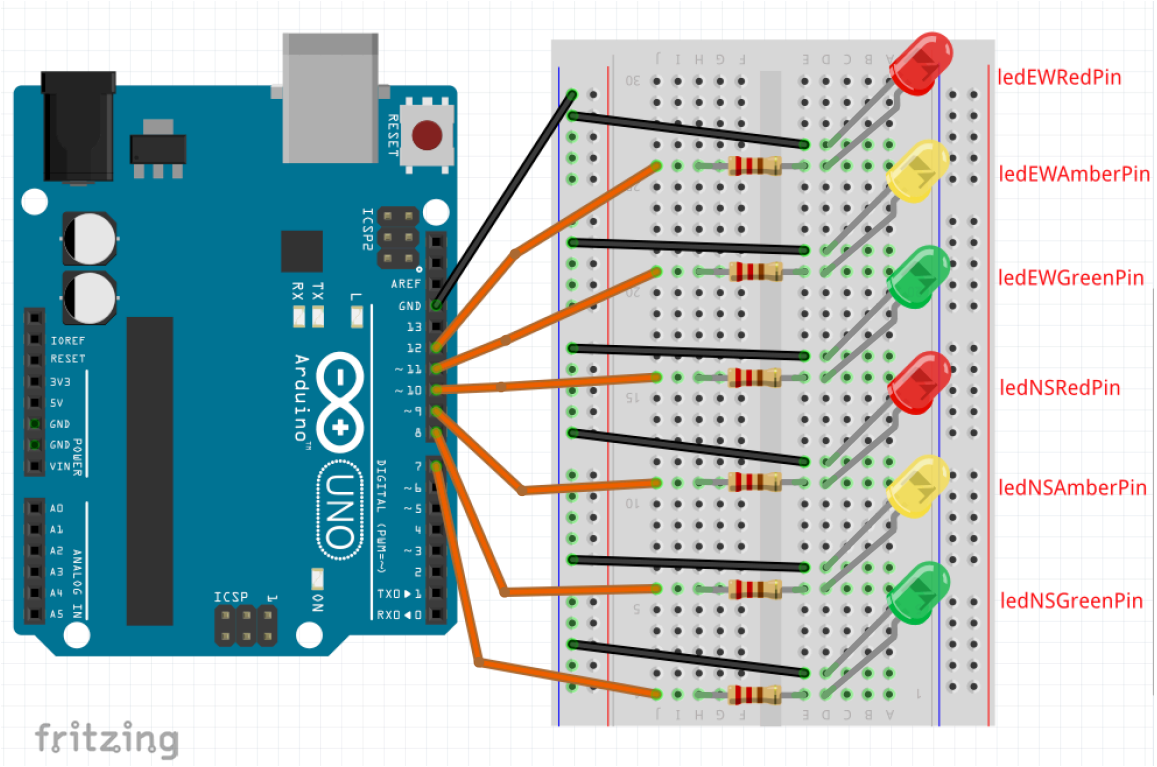
As far as the sketch goes this is just and extended version of the sketch we used for the grade crossing flasher and uses 6 outputs to run all of the LEDs. Be sure to assign pinMode and digitalWrite for each LED in your sketch. Once complete you should have a sketch that looks similar to mine shown below.

The_Traffic_Light

```
// The_Traffic_Light by David King
// This is a simple traffic light sketch that controls the output of sets
// of LEDs, one set for the North/South traffic and a second set for the
// East/West traffic.

// include any associated files in this location

// declare any variables needed in you file here
int ledNSGreenPin = 7;      // This is where I declare the pin assignments for my LEDs
int ledNSAmberPin = 8;
int ledNSRedPin = 9;
int ledEWGreenPin = 10;
int ledEWAmberPin = 11;
int ledEWRedPin = 12;

void setup() {
  // put your setup code here, to run once:
  pinMode(ledNSGreenPin, OUTPUT);     // This is where I set the pins as output devices
  pinMode(ledNSAmberPin, OUTPUT);
  pinMode(ledNSRedPin, OUTPUT);
  pinMode(ledEWGreenPin, OUTPUT);
  pinMode(ledEWAmberPin, OUTPUT);
  pinMode(ledEWRedPin, OUTPUT);

}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(ledNSGreenPin, HIGH);      // Step 1
  digitalWrite(ledNSAmberPin, LOW);
  digitalWrite(ledNSRedPin, LOW);
  digitalWrite(ledEWGreenPin, LOW);
  digitalWrite(ledEWAmberPin, LOW);
  digitalWrite(ledEWRedPin, HIGH);
  delay(25000);

  digitalWrite(ledNSGreenPin, LOW);      // Step 2
  digitalWrite(ledNSAmberPin, HIGH);
  digitalWrite(ledNSRedPin, LOW);
  digitalWrite(ledEWGreenPin, LOW);
  digitalWrite(ledEWAmberPin, LOW);
  digitalWrite(ledEWRedPin, HIGH);
  delay(7000);

  digitalWrite(ledNSGreenPin, LOW);      // Step 3
  digitalWrite(ledNSAmberPin, LOW);
  digitalWrite(ledNSRedPin, HIGH);
  digitalWrite(ledEWGreenPin, LOW);
  digitalWrite(ledEWAmberPin, LOW);
  digitalWrite(ledEWRedPin, HIGH);
  delay(1000);

```

```
  digitalWrite(ledNSGreenPin, LOW);      // Step 4
  digitalWrite(ledNSAmberPin, LOW);
  digitalWrite(ledNSRedPin, HIGH);
  digitalWrite(ledEWGreenPin, HIGH);
  digitalWrite(ledEWAmberPin, LOW);
  digitalWrite(ledEWRedPin, LOW);
  delay(22000);

  digitalWrite(ledNSGreenPin, LOW);      // Step 5
  digitalWrite(ledNSAmberPin, LOW);
  digitalWrite(ledNSRedPin, HIGH);
  digitalWrite(ledEWGreenPin, LOW);
  digitalWrite(ledEWAmberPin, HIGH);
  digitalWrite(ledEWRedPin, LOW);
  delay(6000);

  digitalWrite(ledNSGreenPin, LOW);      // Step 6
  digitalWrite(ledNSAmberPin, LOW);
  digitalWrite(ledNSRedPin, HIGH);
  digitalWrite(ledEWGreenPin, LOW);
  digitalWrite(ledEWAmberPin, LOW);
  digitalWrite(ledEWRedPin, HIGH);
  delay(1000);
}
```

The Fritzing or wiring diagram should look similar to the image below.



Conclusion

This should help you create a few more sketches from items that you see out on every day life. As an example if you model railroad layout has a wharf maybe a couple of navigation lights at or near the dock should be added. You could research the flashing pattern and colour required for each of these lights. Also there are other warning lights that have patterns that are not simple on/off cycle such as the red/blue lights of a police car.

Have fun and check out the video and we will continue this adventure in the next issue of ***The Canadian***!