Traffic Light

Now that you've practiced setting up (and controlling) multiple LEDs on a breadboard, it's time to create a variation of a traffic light!

There are two versions of this project. Make sure you try both!

The template sketch that you'll be using for this project contains some logic errors. The code *works* but you'll need to follow the comments and figure out how to improve it!

Base Activity

The base activity for this project is to create an LED and Arduino powered traffic light! You'll be using *at least 3* unique LEDs that you'll need to program to change based on what you know about traffic lights.

This experiment is designed to use the skills you've been practicing while also test your critical thinking and problem solving abilities!

Supplies

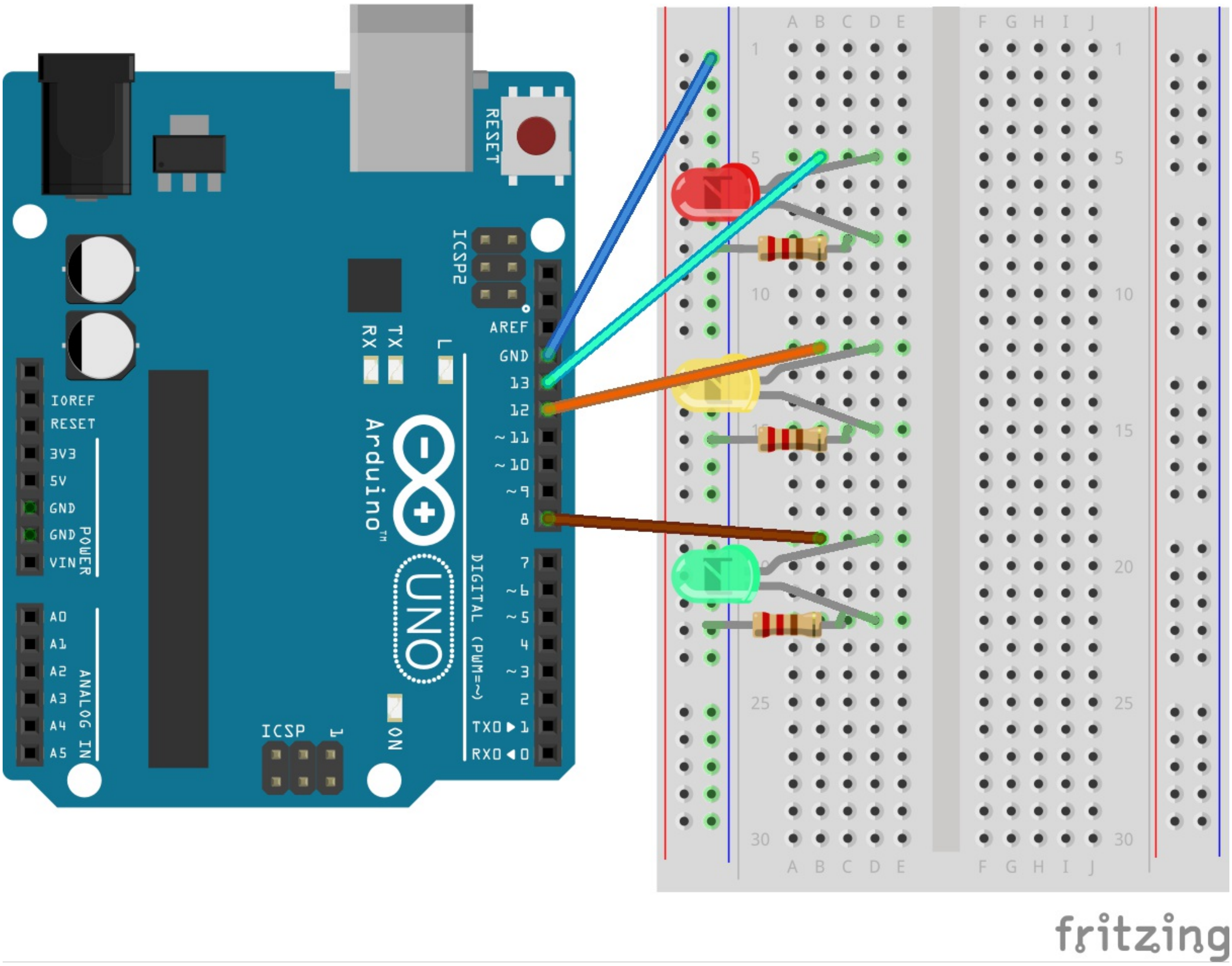
- Uno and breadboard
- Jumper wires
- *at least 3* uniquely colored LEDs
  - Recommended: Red, Yellow, Green
- 220 or 560 Ohm resistor for *each* LED

Steps

The steps for this challenge are broken into two parts: the circuit setup and the sketch. Begin with the circuit setup.

Circuit:

Refer to this *Fritzing* diagram throughout the steps:



1. Connect the GND of the Arduino to the - rail of the breadboard.
2. Connect *each* LED to the breadboard.
   - Recommended: Align the LEDs so that the positive leg (long leg) is at the top.
3. Connect the positive leg (long leg) of *each* LED to a digital pin on the Arduino.
   - For example, you'll want to connect the positive leg of each LED to a unique pin, such as pins 13, 12, and 8 in the diagram.

    - Since you're using 3 LEDs, you'll need to use 3 pins.
4. Connect a 220 (or 560) Ohm resistor between the - rail of the breadboard and the negative (short leg) of the LED.

Sketch:

Here is the basic sketch for this project. Remember that there will be some problems that you'll need to fix!

Gist Link : If you prefer to get the code from a Gist, go to this link. You can click the *Raw* button to get a format that's easier to copy.

Suggested: Instead of copying the code, read the comments and type it yourself! This is the best way to develop an understanding of each piece.

```
/* Arduino Experimenter Challenge: Broken Traffic Light
   * This traffic light is -sooo close- to being correct, bu
   * off with the logic...
   *
   * Can you fix it?
   *
   * Hint: Think about how a functioning traffic light works
   *  what steps should the lights go in?
   *  should two colors ever be on at the same time?
   *
   * Going Further:
   *  after you get the light working correctly, invent your
   *
   *
   */

  // global variables go here. set and change the led pin va
  int red = 13;
  int yellow = 12;
  int green = 8;
```

```
void setup() {
  // make sure to set the pinMode to output for EACH led var
  pinMode(red, OUTPUT);
  pinMode(yellow, OUTPUT);
  pinMode(green, OUTPUT);

}

void loop() {
  changeLights();
  delay(10000); //process starts over every 10 seconds
}

void changeLights() {
  //first, green is on and the other two are off:
  digitalWrite(red,HIGH);
  digitalWrite(yellow,LOW);
  digitalWrite(green,LOW);
  delay(5000); //red light for 5 seconds!

  //now, yellow light:
  digitalWrite(yellow,HIGH);
  digitalWrite(green,LOW);
  digitalWrite(red,LOW);
  delay(2000); //yellow light for 2 seconds!

  //now, green light:
  digitalWrite(green,HIGH);
  digitalWrite(red,LOW);
  digitalWrite(yellow,LOW);
  //green light!

}
```

Traffic Light: Experimenter Challenges

Once you've figured out how to fix the example, go ahead and make some changes and program your own light!

Experiment with different *delay()* values as well as different choices. Perhaps invent your own system of traffic control where you need *two lights* to be on in a quick pattern to signal traffic to move!