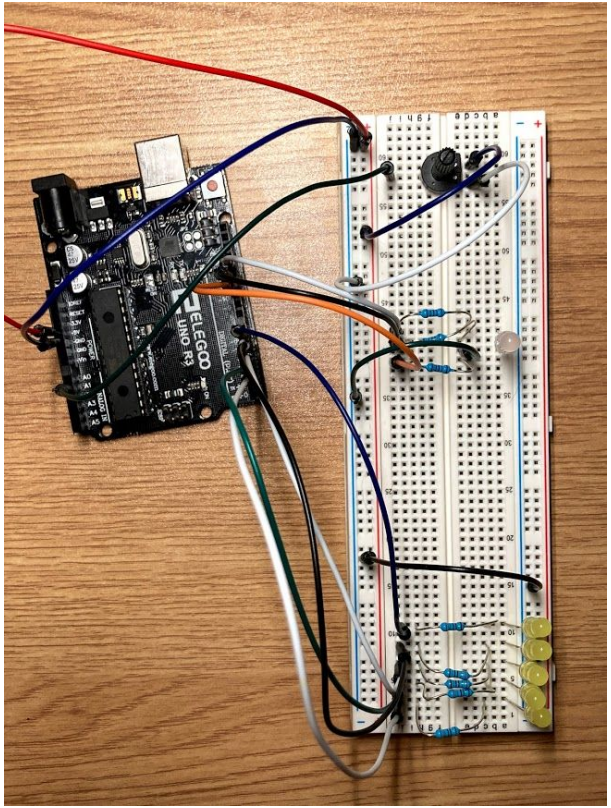
Embedded System Workshops

07. Analog Inputs & Outputs
CCA Girls Who Code

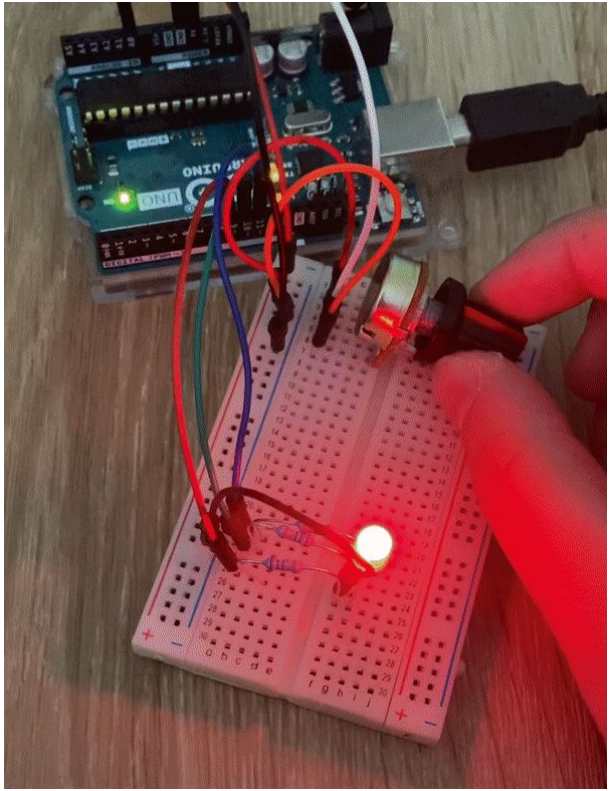
Project Overview

- Purpose
  - Introduce analog inputs and outputs and what distinguishes them from digital ones
- Project
  - 3 exercises in Analog Inputs/Outputs
- Grab your kit, and let’s get started!



What are we making?

- Analog Inputs/Outputs Projects
  - Randomized RGB LED: Changes to a random color every second
  - Potentiometer to light up a row of LEDs
  - Mood lighting using a potentiometer



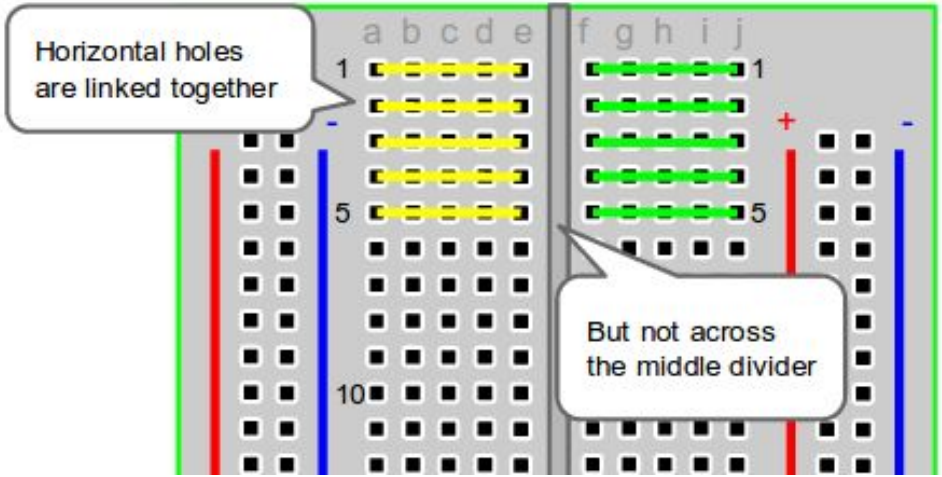
Parts List

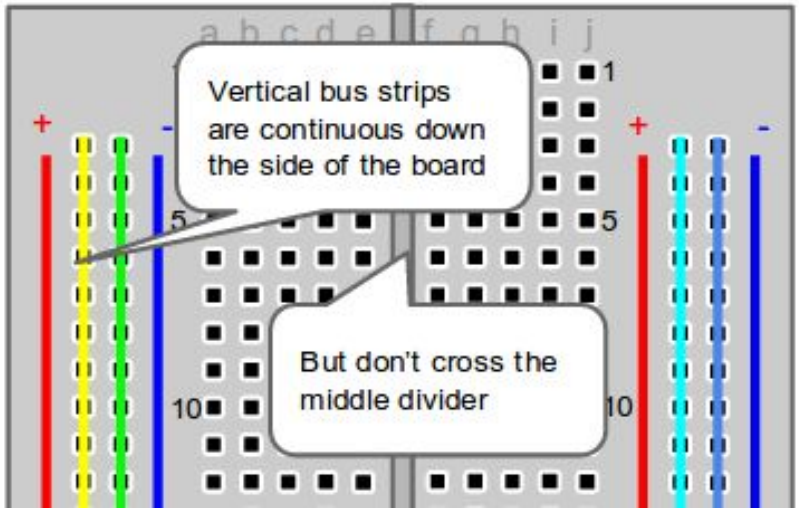
Analog Inputs/Outputs Project

- Arduino UNO R3 Controller Board
- USB Cable
- Breadboard
- 10KΩ Potentiometer
- RGB LED
- 5x LEDs of your choice of color
- 5x 330Ω Resistors
- 3x 220Ω Resistors
- Male-to-male jumper wires

Review

Review: Breadboards Explained

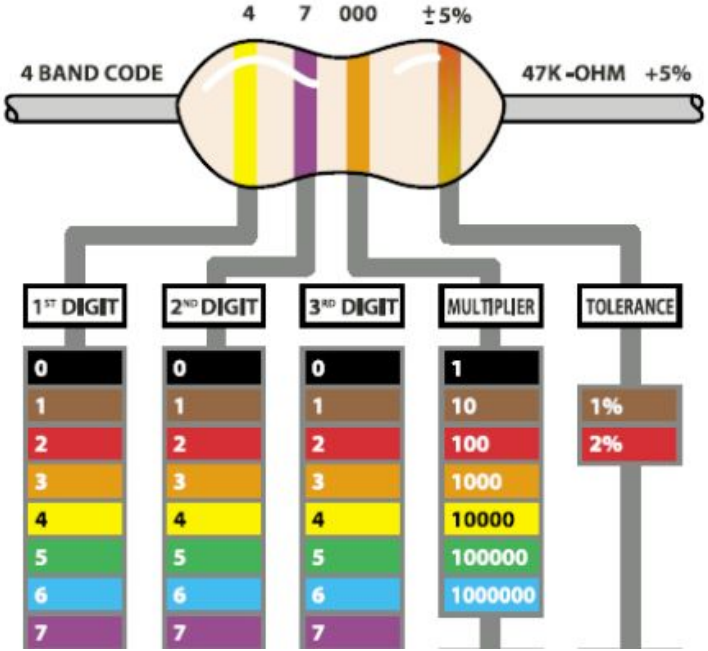




Tip: It is good practice to have your power input connected to the red/positive rail and your ground pin connected to the blue/negative rail.

Review: Resistors

- Resistors slow the electric current, and control where and how fast the current flows
- Resistance value is measured in ohms Ω, which is represented by colored stripes on the body of the resistor
- Each stripe has a different value depending on the color and location as shown in the reference chart

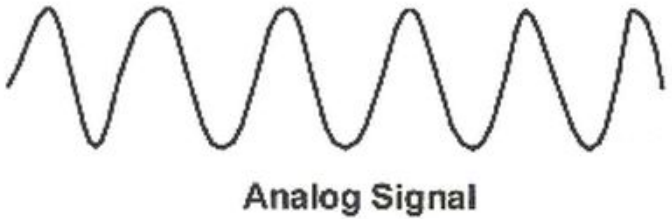


Project

Introduction to Analog Signals

An **analog signal** is one that can be adjusted between **a range of values**. This is in contrast to a digital signal, which is either on or off.

A helpful analogy is to think of a digital signal as a switch or a button, whereas an analog signal is a dial or knob.



Analog Signal

Digital Signal

Examples of Digital vs. Analog

Analog

- Potentiometer
  - Vary the brightness of a lightbulb
- Analog Pressure Sensor
  - Record a varying weight
- Joystick
  - Sends two analog signals for X and Y axis

Digital

- Switch
  - Turn a lightbulb on or off
- Digital Pressure Sensor
  - Sense if weight is above a certain value
- Button
  - Sends a digital signal, whether the button is pressed or not

Examples of Digital vs. Analog (continued)

Analog

- Potentiometer
- Stepper, Servo, and DC Motors
- Ultrasonic Sensor (distance sensor)
- Temperature and Humidity Module
- Joystick Module
- Photoresistors (light sensors)
- Thermistor (Temperature)

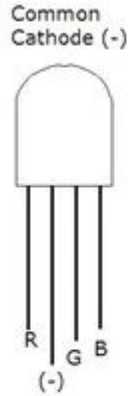
Digital

- Buttons
- LEDs/Lightbulbs
- Proximity Switch (detects if an object comes within a range)
- Level Switch (detects specific level of liquids)
- Circuit Breakers

RGB LED: 16 million different colors?!

Two of our three exercises involve using an RGB LED, which is essentially three different LEDs (red, green, and blue) in one housing. Varying the colors of the three LEDS allows us to produce an insane amount of different color combinations (over 16 million, to be exact).

The longest of the four pins goes to ground, and the other three pins each corresponds to a certain color LED.

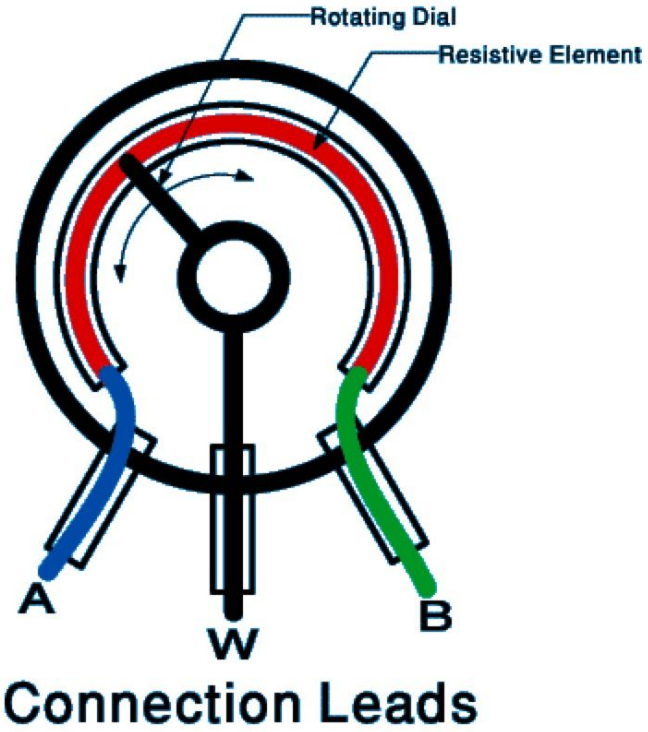


The Truth About Potentiometers

(yes, I lied to you)

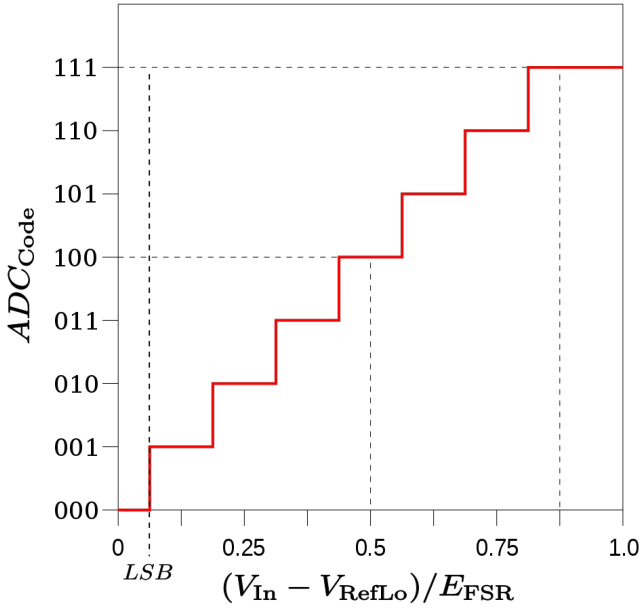
A potentiometer is not actually a variable resistor, *per se.* Rather, it is essentially two variable resistors placed in series with a wire between the two connected to an input.

Turning the dial essentially varies the ratio between the two resistors, giving us different voltages at point W on this diagram. The voltage at point W can then be used as an analog signal.



Analog-to-Digital Conversion

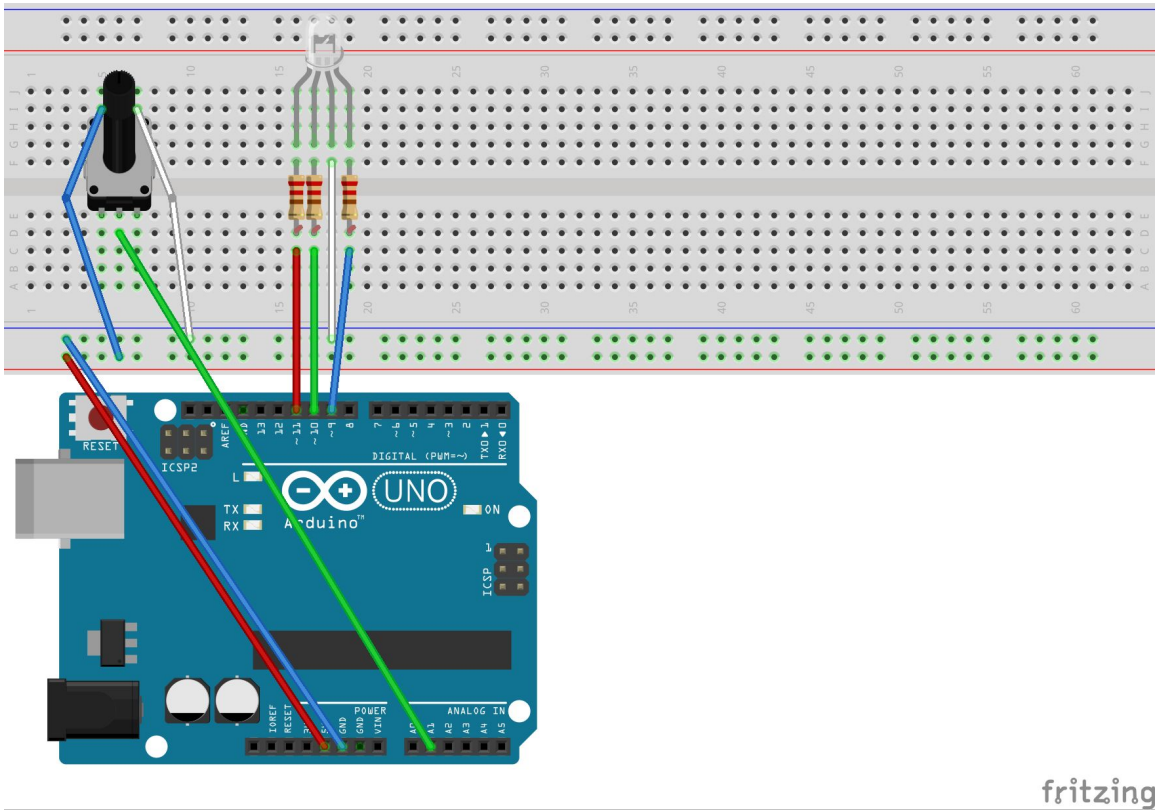
Essentially, an analog to digital converter converts an analog signal into a digital signal, usually through various boolean-style functions.



Exercise 1: Randomized RGB LED

Our first exercise involves making an RGB LED change to a random color every second.

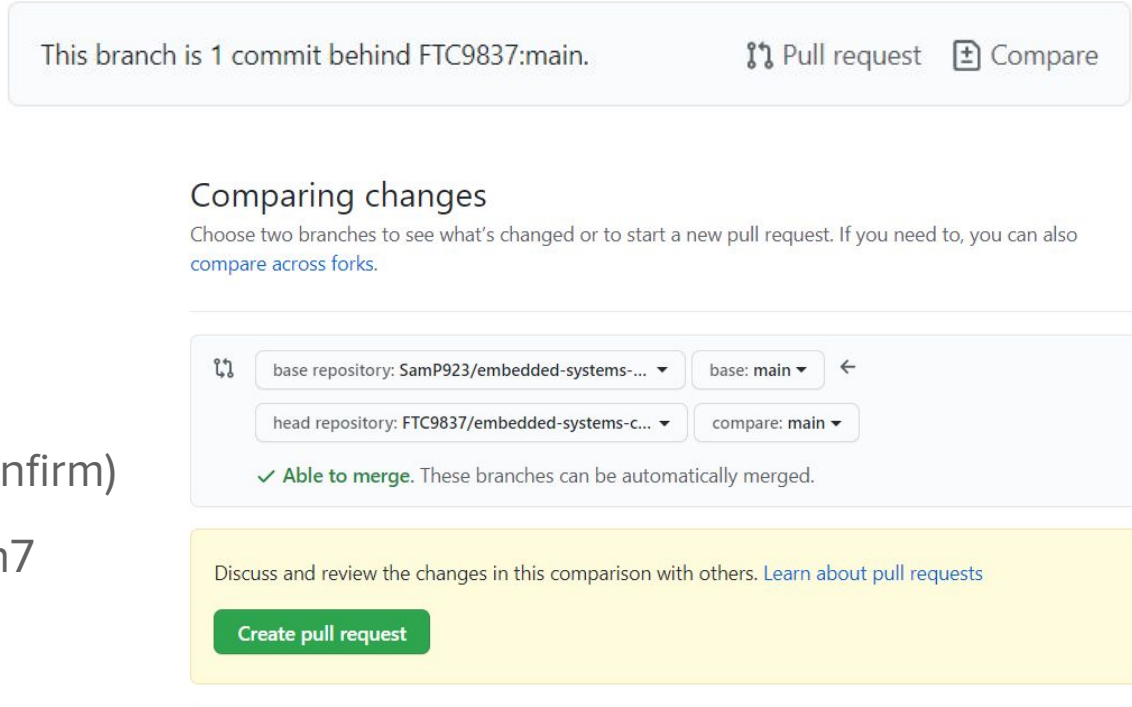
NOTE: while there is a potentiometer in this schematic, it will not be used just yet.



Grab the Starter Code from GitHub

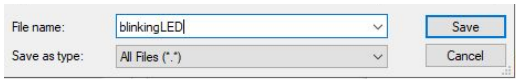
If you haven't made the repository yet, check these slides

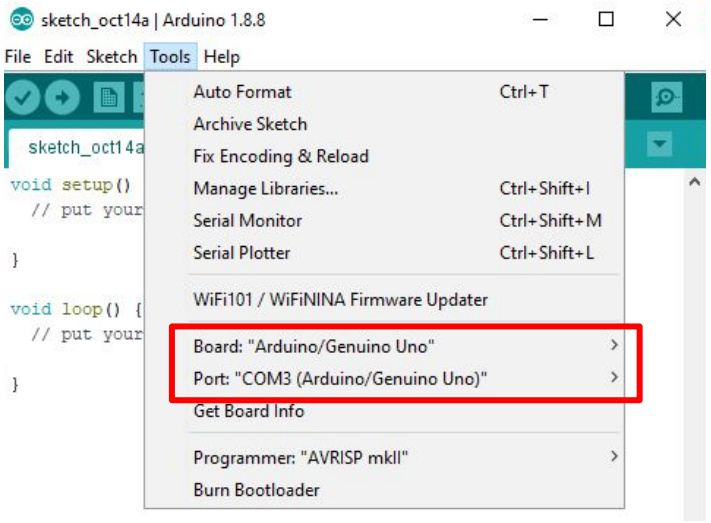
- Go to your repository (username/embedded-systems-course)
- Click "Compare"
- Switch the repos so yours is the base repository and FTC9837 is the head
- Create pull request (x2)
- Merge pull request (and confirm)
- You should have the lesson7 folder in your repository



Review: Setting up Arduino

- Find and open Arduino on your desktop
- Click “File” in the top left corner and click save
- Save this tab as “randomRGB”
- Connect the USB cord in your kit to the Arduino and the computer (USB port is on the left side of the monitor)
- Open the “Tools” Window and make sure the board has been recognized and the port is “COM#(Arduino/Genuino Uno)”





Code

Verify

RandomRGB

```
int rgb[3] = {9,10,11};
int randint = 0;

void setup() {
  // put your setup code here, to run once:
  for(int i = 0; i < 3; i++){
     pinMode(rgb[i], OUTPUT);
  }
}

void loop() {
  // put your main code here, to run repeatedly:
  for(int j = 0; j < 3; j++){
     randint = random(255);
     analogWrite(rgb[j], randint);
  }
  delay(1000);
}
```

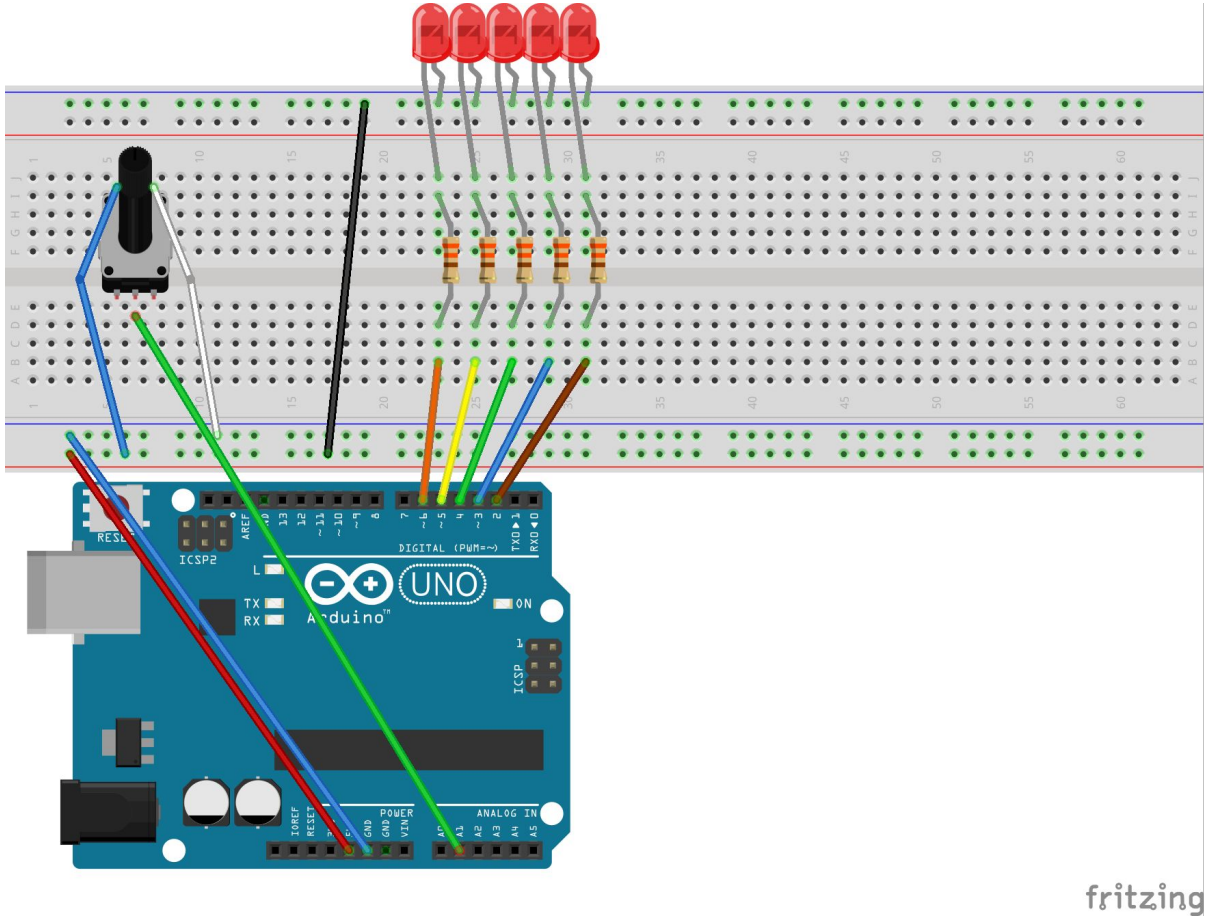
Initializes the pins for the RGB LED

Sets the RGB pins to outputs

Sets each color of the RGB LED to a random value

To test your code, click the checkmark then the arrow!

Exercise 2: Lighting a Row of LEDs



LED Row Code

Initializes the pins for the five LEDs

These **pinMode** functions set the potentiometer pin to input and LED pins to outputs

This if-statement checks the **value of the potentiometer pin** and turns on the corresponding number of LEDs

RowOfLEDs

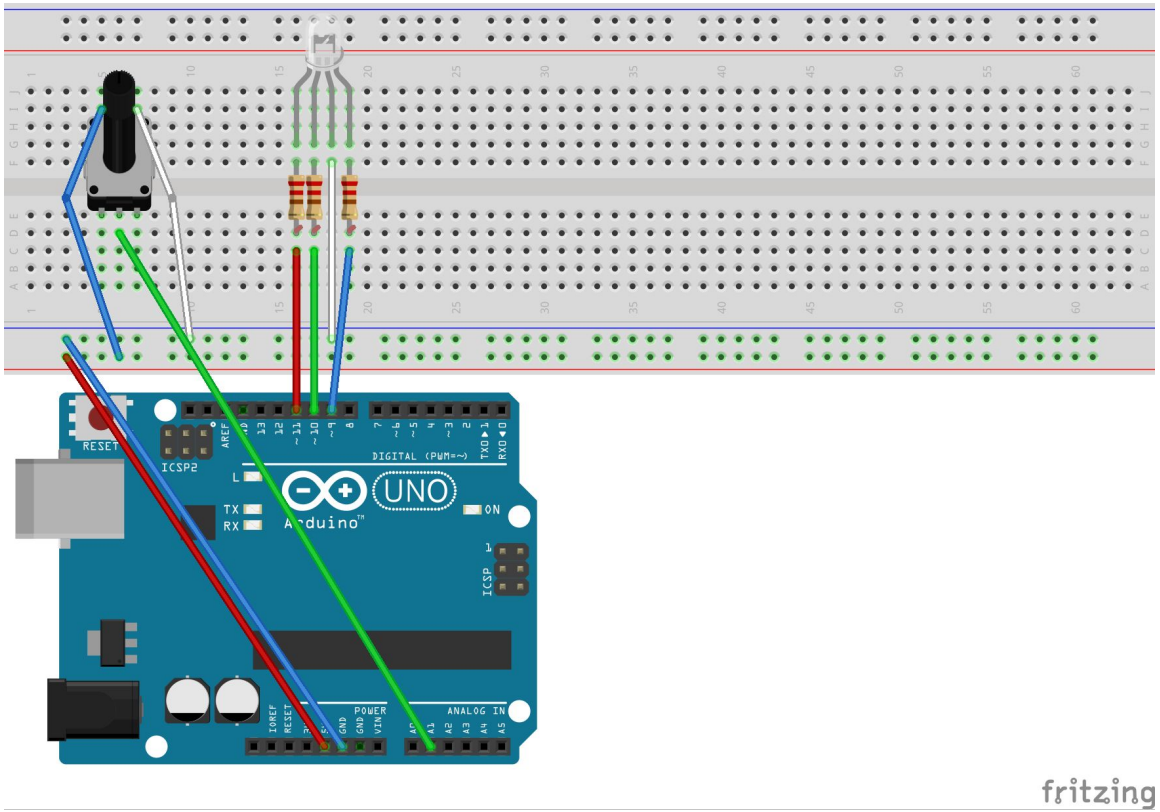
```
int ledpins[5] = {2,3,4,5,6};

void setup() {
  // put your setup code here, to run once:
pinMode(A1, INPUT);
for(int i = 0; i < 5; i++){
  pinMode(ledpins[i], OUTPUT);
}
}

void loop() {
  // put your main code here, to run repeatedly:
  for(int j = 0; j < 5; j++){
    if(analogRead(A1) >= (190 * (j+1))){
      digitalWrite(ledpins[j], HIGH);
    }
    else{
      digitalWrite(ledpins[j], LOW);
    }
}
}
```

Exercise 3: Mood Lighting

Same schematic as Exercise 1.



Mood Lighting Code

Initializes the variables and pins for the RGB LED

Sets the values of the three colors based on the analog input

MoodLighting

```
int volts = 0;
int red= 0;
int green = 0;
int blue = 0;
int rgb[3] = {9,10,11};

void setup() {
  // put your setup code here, to run once:
  pinMode(A1, INPUT);
  for(int i = 0; i < 3; i++){
     pinMode(rgb[i], OUTPUT);
  }
  Serial.begin(9600);
}
```

```
void loop() {
  // put your main code here, to run repeatedly:
  volts = (analogRead(A1) * 0.5);
  if(volts < 255){
    red = 255;
    green = 255;
    blue = volts;
  }
  if(volts >= 255){
    red = 255 - (volts - 255);
    green = 255 - (volts - 255);
    blue = 255;
  }

  analogWrite(rgb[0], red);
  analogWrite(rgb[1], green);
  analogWrite(rgb[2], blue);
  Serial.println(volts);

}
```

Sets potentiometer pin to input, RGB pins to output

Writes the values to the three light colors

Thank you!

CCA Girls Who Code
ccagirlswhocode@gmail.com
www.ccagirlswhocode.weebly.com

Production Team

Curriculum Director: Stefan Prestrelski

Teaching Assistants: Sarah Luo, Samantha Prestrelski