# Dimming LED

Previously, you've programmed the LED fades into your code. For this challenge, you're going to be using a *potentiometer* to control the fades.

This is much like the project you did for the *potentiometer* guide, however this time there are added challenges!

# Base Activity

These challenges build on the basic activity that you did in the *potentiometer* guide where you faded an LED by turning the potentiometer. Each challenge is designed to expand a skill that you covered in that project!

The main concept is that you're mapping the input sent by the potentiometer to an output of the LED.

This basic concept can be extended and used with *other components* as well!

## Supplies

The supplies are the same as for the basic potentiometer activity. Here's a reminder:

- Uno and breadboard
- Jumper wires
- Potentiometer
- 10k Ohm resistor for the potentiometer
- LED circuit supplies:

- LED and 220 (or 560) Ohm resistor

Remember, since these challenges use more than 1 LED, you'll need a resistor for *each* LED!

# Steps

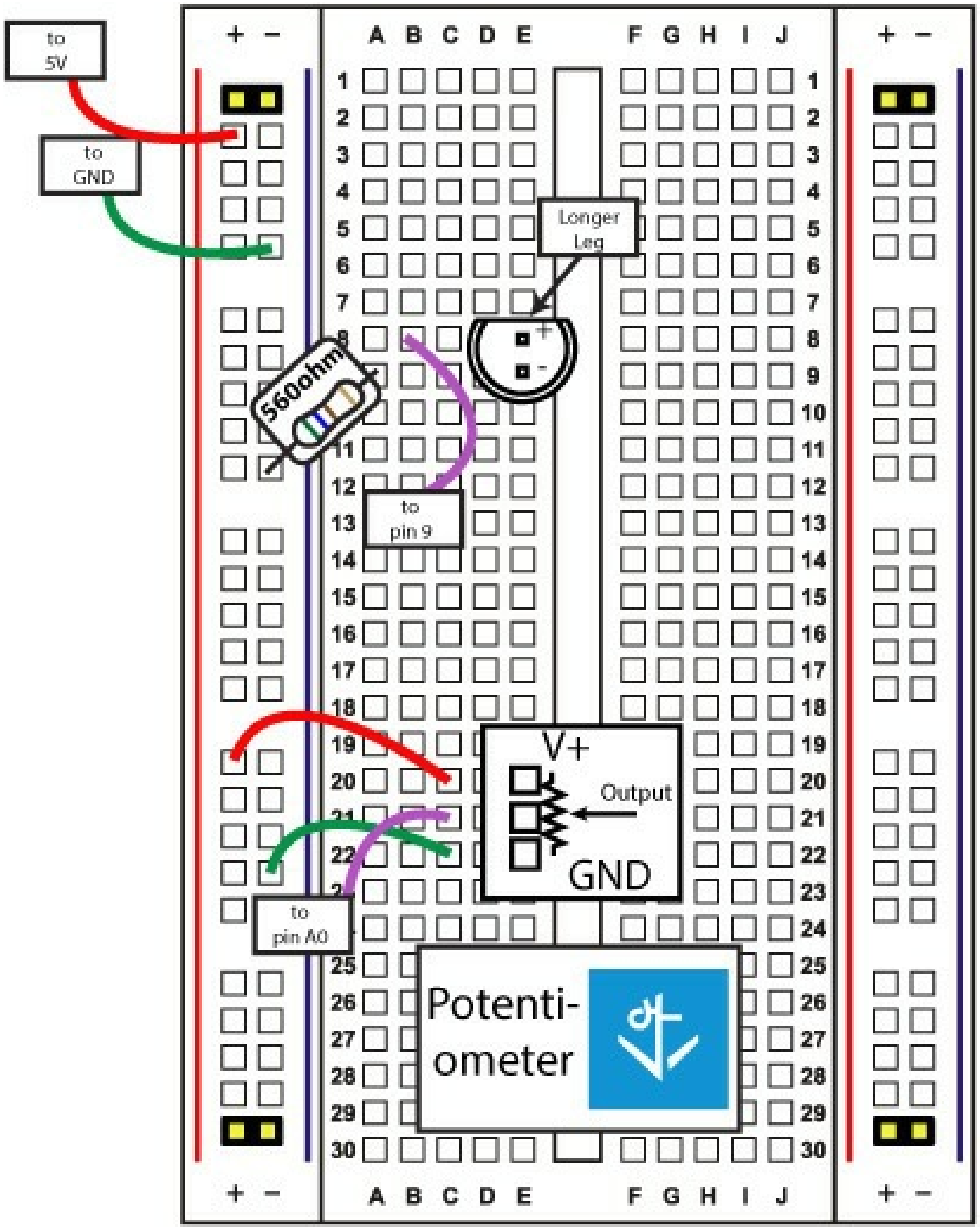
This challenge builds on the example project that you did in the *potentiometer guide*. If you need a refresher on any concepts in this experiment, head back to that guide!

## Circuit

This is the diagram and steps for the basic activity. For the Experimenter Challenges, you'll need to build on this basic circuit to solve the problem!

Use this breadboard diagram as your guide:

# Analog Input



1. Connect the 5V (Power) on the Arduino to the + rail of the breadboard.
2. Connect the GND on the Arduino to the - rail of the breadboard.
3. Wire up the LED circuit that you've previously done. If you need help, look back to one of the previous projects.
4. Attach the potentiometer to the breadboard. Place it so that the pins are closest (on the left side) to the Arduino. The edge of the potentiometer can hang over into the middle of the breadboard where the rails split.
5. Use 1 jumper wire to connect the power rail of the breadboard to the **V+ (top pin)** on the potentiometer.

6. Use 1 jumper wire to connect the output (middle pin) of the potentiometer to the **A0 analog input** on the Arduino board. Make sure to use the **A0** input and not a digital input.
7. Use 1 jumper wire to connect the **GND (bottom pin)** of the potentiometer.

## Sketch

The sketch for the Experimenter Challenge will build on the basic potentiometer sketch that you've previously used.

Here is a refresher:

```
int ledPin=9; //Variable to store pin of LED
int potentPin=A0; //Variable to store pin of potentiometer
int potentValue=0; //Variable to store last known value of p
int brightnessValue=0; //Variable to store LED brightness

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT); //Setup LED pin for output
}

void loop() {
  // put your main code here, to run repeatedly:
  potentValue=analogRead(potentPin); //Read the value of the
  brightnessValue=map(potentValue,0,1023,0,255); //Map the p
  analogWrite(ledPin,brightnessValue); //Set the brightness
}
```

Remember, any changes that you make to the board (such as adding more LEDs) will need to be included in the sketch as well.

# Dimming LED: Experimenter Challenges

Now for the challenges!

In the base activity, your potentiometer controlled the brightness of a single LED. This challenge is going to be fairly open ended: add more LEDs!

Here's some suggestions for what this could look like:

1. *Multi Fades* : Leave the mapping of the first LED similar to how you already have it, and then have another LED with a *completely different* mapping!

2. *LED-o-meter* : Use a row of LEDs that turn on as you dial the potentiometer. Think of the row of LEDs like a meter that lights up as you turn the knob. You could even build out a simple interface for this!

3. *LED Dial* : Using at least 3 unique LEDs, map each color to a range of *potentiometer* values. For example, if the value is low, the green LED can be on.

These are just some suggestions to get you started! You're able to solve each of these challenges with the skills that you currently know- you'll just need to combine several concepts!

# Overview

The *potentiometer* is just one type of *analog input* that can be used with an Arduino. There are lots of things that can be done by combining the potentiometer with other components.

Start experimenting and see what you come up with!

If you're running into a lot of trouble, you can take a glimpse at the solutions. However, a lot of the fun of these projects is working through the logic and problem solving required to solve the challenge!