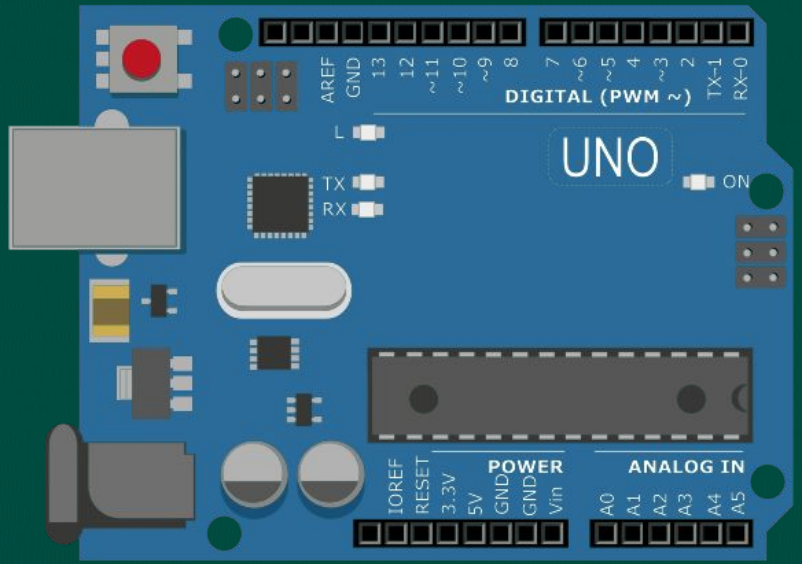
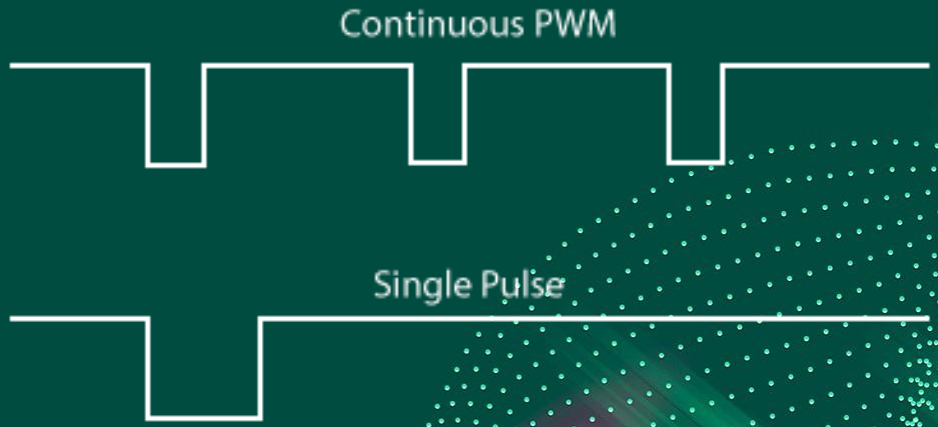
Lesson 2

Arduino Basics: LEDs



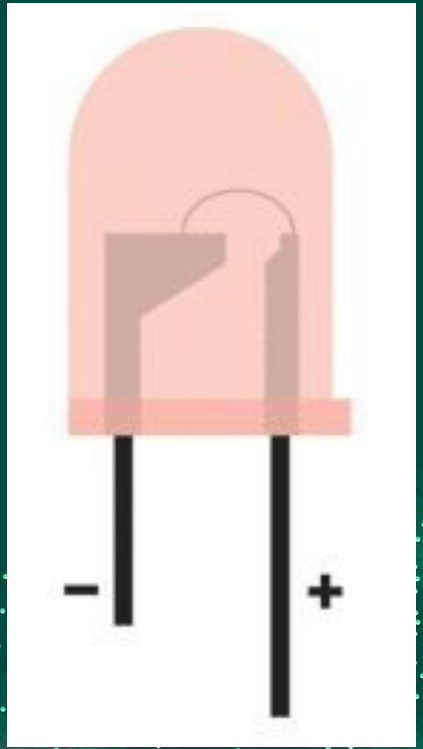
Today's Goal:

- Today we will use what we learned last time, and add an LED to the mix.
- Using the LED we will signal the same Morse Code message with our Arduino ("Hello World!").
- We will then experiment with Pulse Width Modulation, also known as PWM, and see what we can do with the LEDs.



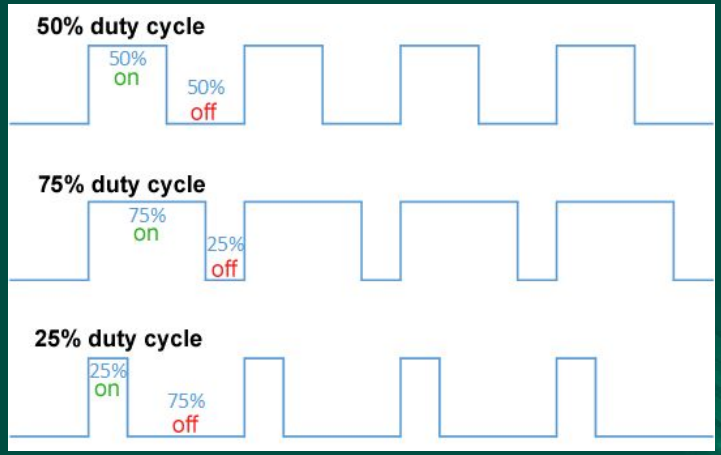
How do LED's Work?

- The name "LED" actually means Light Emitting Diode.
- **Light emitting** just means that they are able to light up.
- **Diode** means that they only conduct electricity one way. This is why you must be sure to always plug an LED in the right way.
- LEDs have a "+" leg, always the longer one, and a "-" leg.
- LEDs are actually made out of the same thing as computer chips, semiconductors! Semiconductors themselves are made out of a refined form of sand. Thus, computer chips, as well as the lights we use (as they are often LED), are partially made out of sand!



What is a PWM Signal?

- A PWM signal is also known as a Pulse Width Modulation signal.
- **PWM signals are digital**, meaning that the voltage is either a “1” or a “0.”
- “**Pulse Width Modulation**” basically means that information is carried in how long the “1” portion of the signal is compared to the “0” portion.
- The Arduino has special pins specifically designed to both input and output PWM signals.
- PWM signals are used everywhere in the modern world, including motors, audio, and telecommunications (like in a phone)!



How Can We Use PWM signals, LEDs, and Arduinos all together?

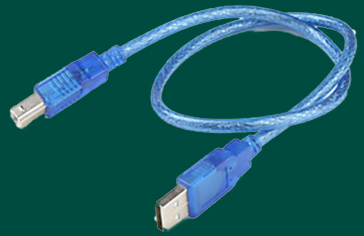
LEDs Can Be Turned on and Off By an Arduino

- LEDs can be connected to the output pins of an Arduino and a voltage can be used to turn them on.
- Additionally, just like the frequency of the buzzer could be controlled by the frequency of the outputted signal, the **brightness** of the LED can be controlled by they **duty cycle** of the PWM signal.
- **Duty cycle** is essentially just what percent of a cycle is “1”.
- Say a complete period of the signal is 10ms, then with a duty cycle of 25%, the voltage would be at “1” for 2.5ms and at “0” for 7.5ms.
- The greater the duty cycle, the brighter the LED.
- One might think that we would see the LED flickering as the voltage goes to “1” and “0”, but if the frequency is greater than 60Hz (meaning there is a complete cycle at least 60 times a second), the blinking is too quick to see with the human eye!

What We Need

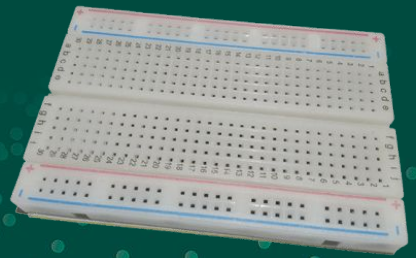
1 x Arduino Uno
(Elegoo brand)

1 x USB Type B to
USB Type A Cable



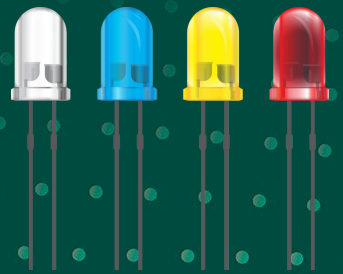
1 x Push Button
w/ Cap

1 x Breadboard



6 x Male-Male
Wires

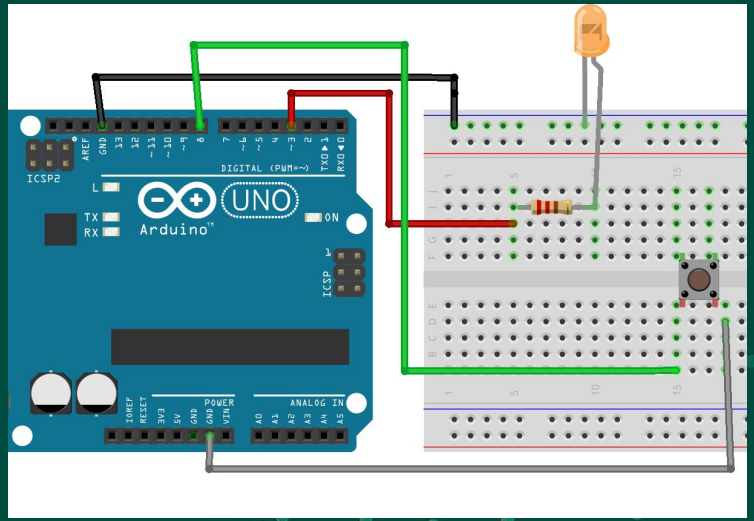
3 x LED (any color)



What We're Building:

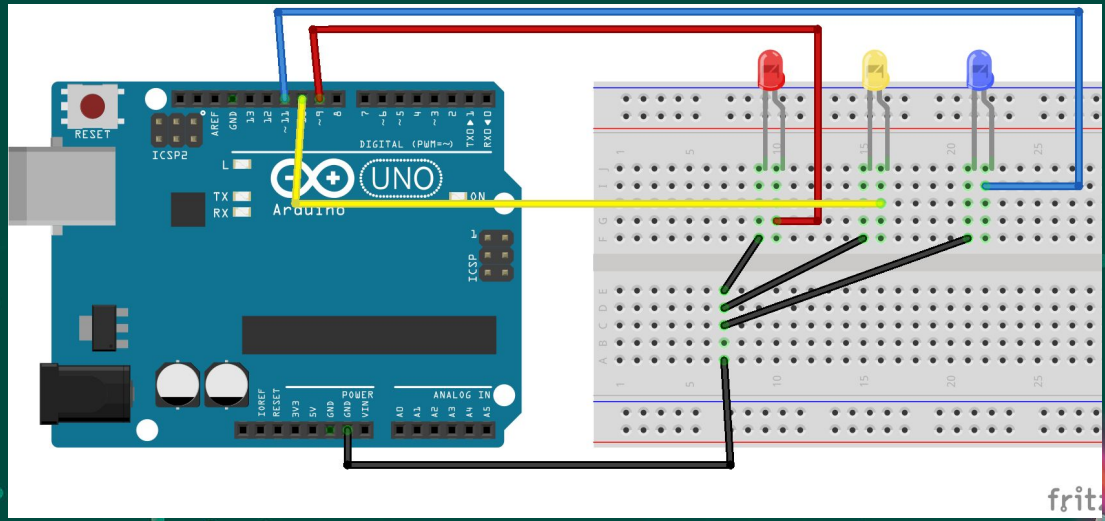
Push Button LED

LED connected to pin 3 through a 220Ω resistor, and a button connected to pin 8.

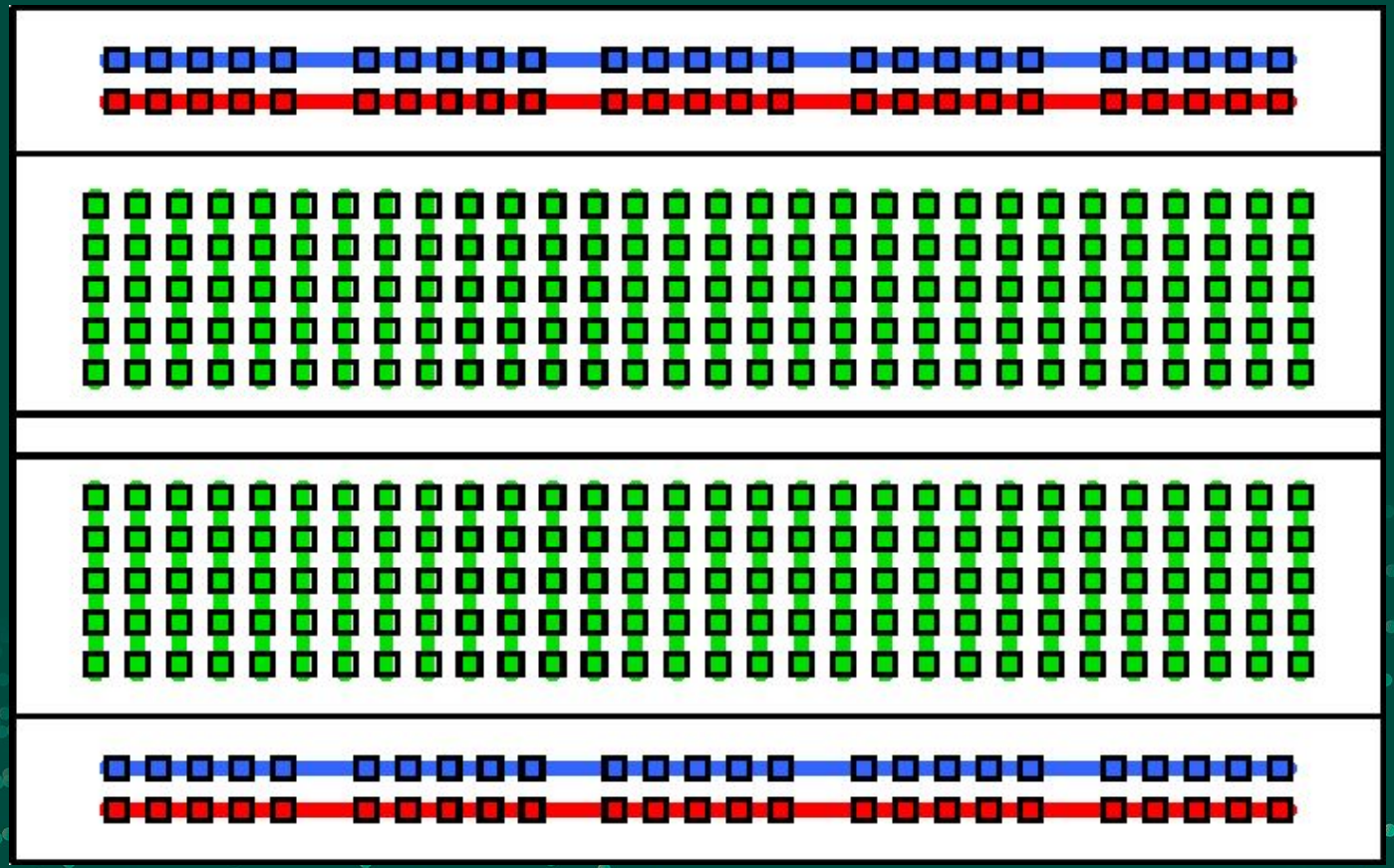


Multi-LED System

3 different colors of LEDs connected to one GND pin, and one each connected to pins 9, 10, and 11.



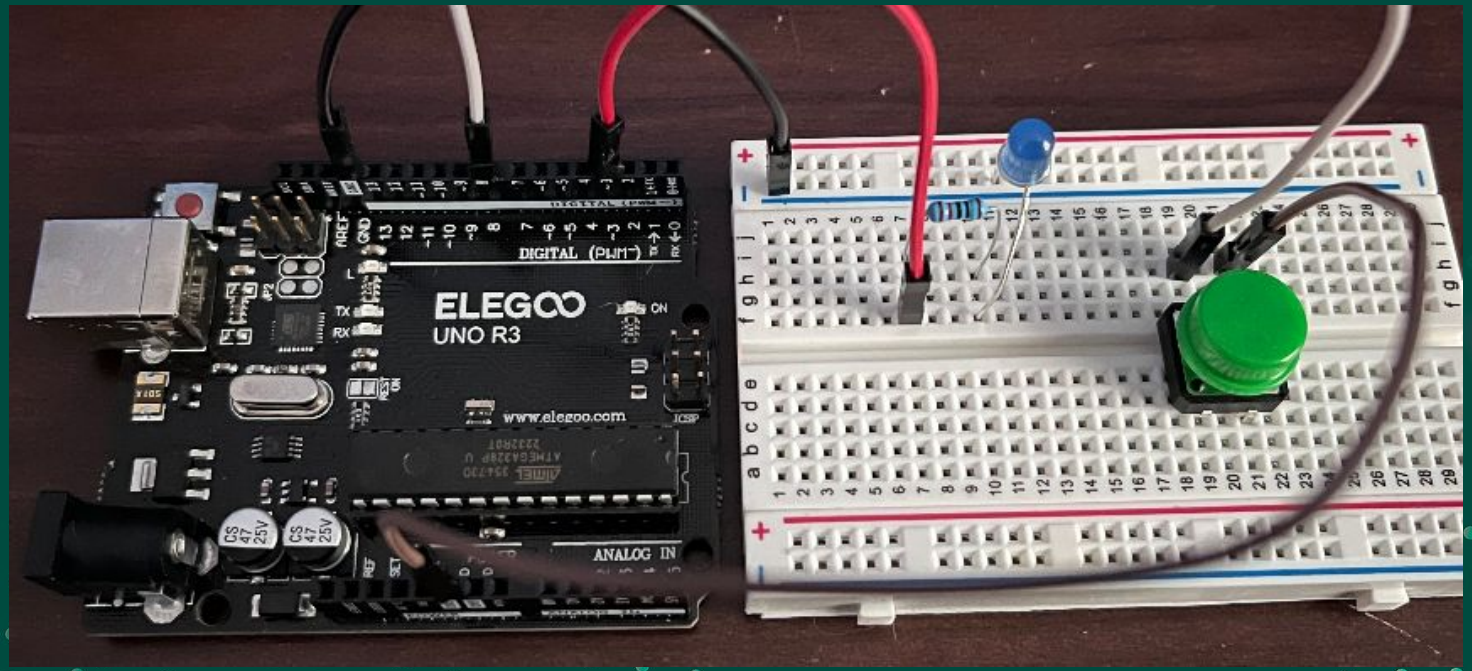
Breadboard Wiring - Reminder



Everything Wired Up - Simple Push Button LED

Notes:

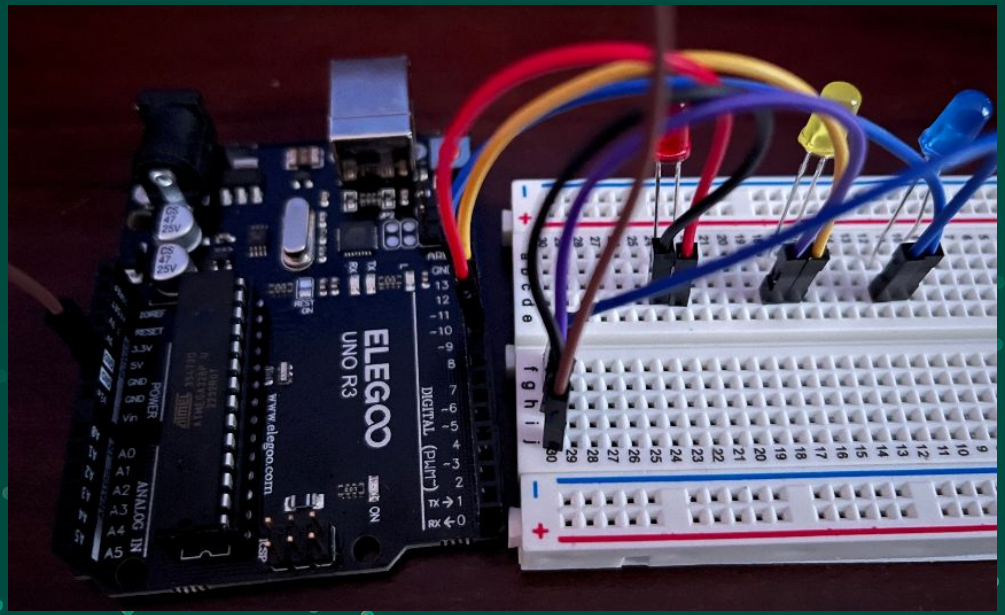
- Make sure to push the button in fully.
- Make sure to orient the “+” and “-” legs of the LED correctly.



Everything Wired Up - Triple LED PWM Test

Notes:

- Make sure to connect each of the “+” and “-” legs of the LEDs correctly.
- Try not to confuse the wires, so use colors you can remember.
- Remember how breadboards are wired.



Testing the Circuits

To test the single LED circuit, follow these steps:

1. Connect your Arduino to your Chromebook
2. Upload the code (found on the GitHub in the Lesson 2 folder, called *LED_Push*)
3. Try pressing the button and see if you can blink the Morse Code message.

Morse Code:

.... . .-.. .-.. --- / .-- --- .-. .-.. -.. -.-.--

To test the 3 LED circuit, follow these steps:

1. Connect your Arduino to your Chromebook
2. Upload the code (found on the GitHub in the Lesson 2 folder, called *3LED_PWM*)
3. Try to modify the code and see what patterns you can make

Thank you for coming to this lesson, and I hope you learned something!

Come back next week for more, including some displays!