introduction to

Digital Electronics

LED Arrays & Strips

MIT ILLUMINATIONS SEMINAR

Hello!

Warm Up!

Come up with as many uses as possible for **10,000 red light bulbs**

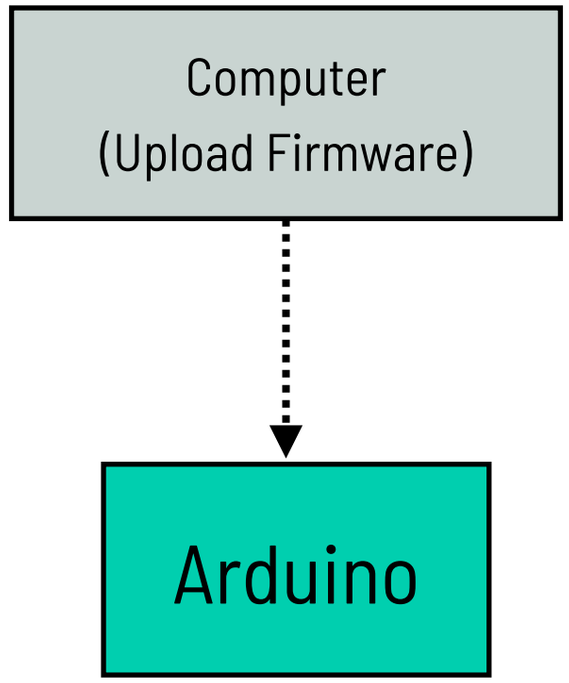
Today

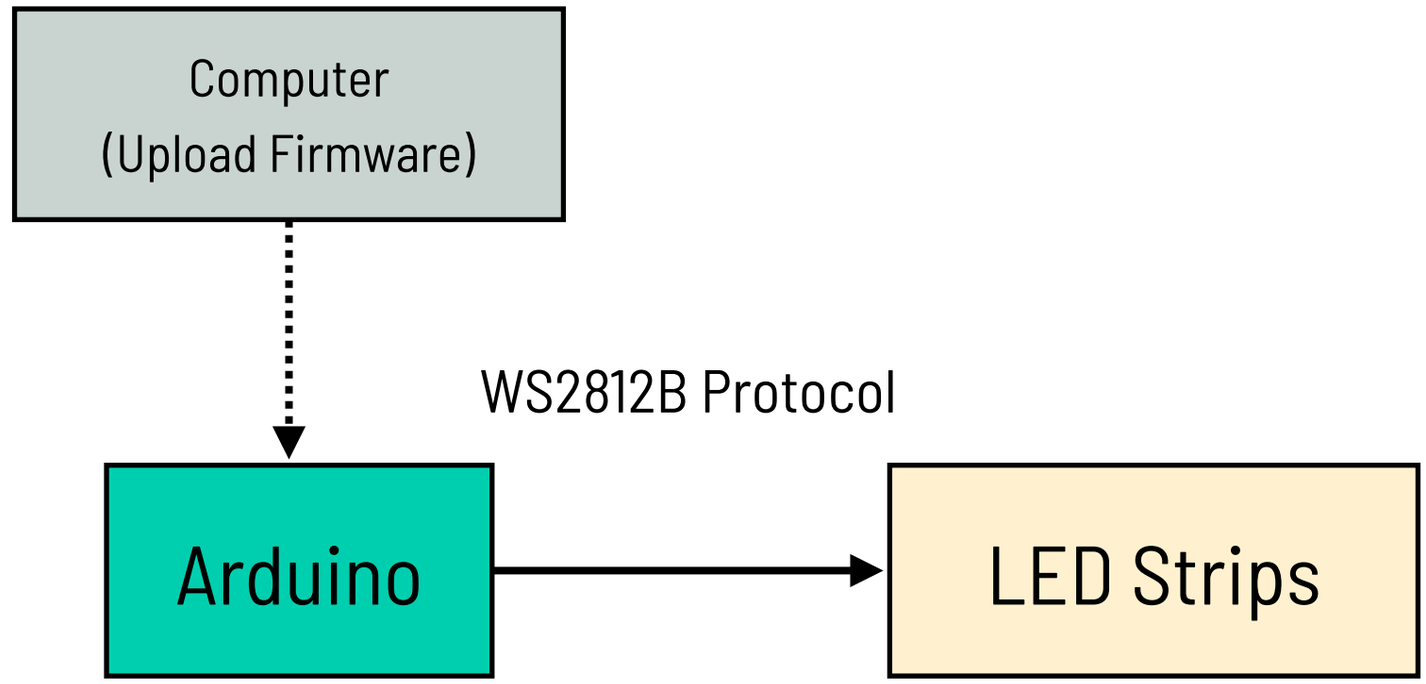
Deeper dive into RGB

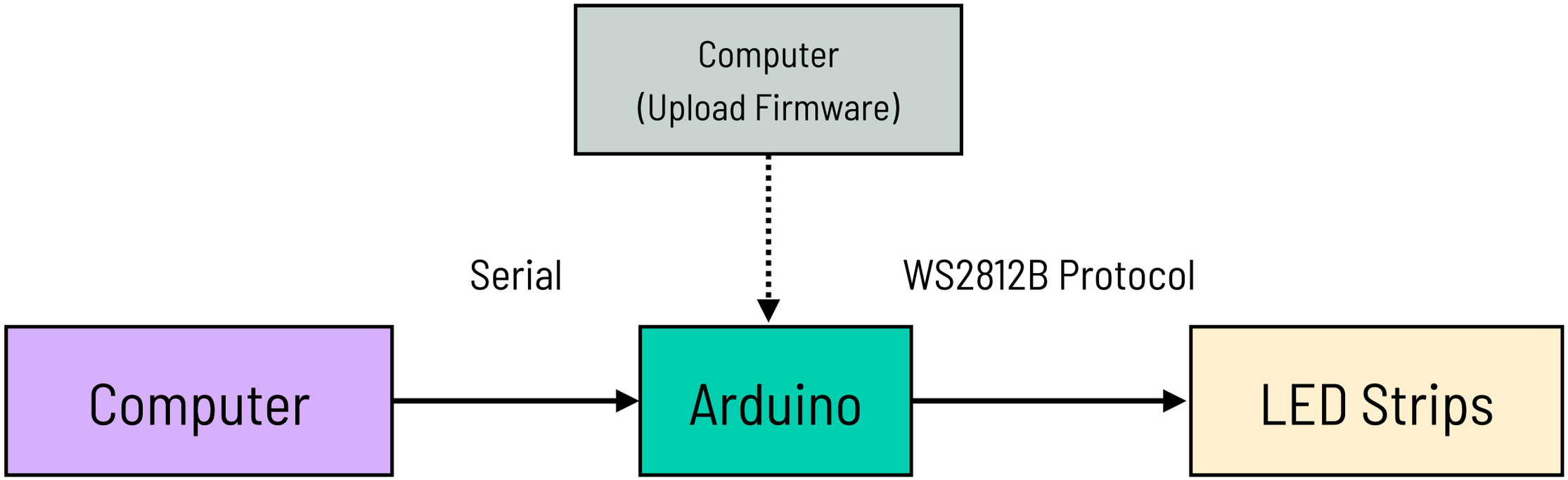
Pulse Width Modulation

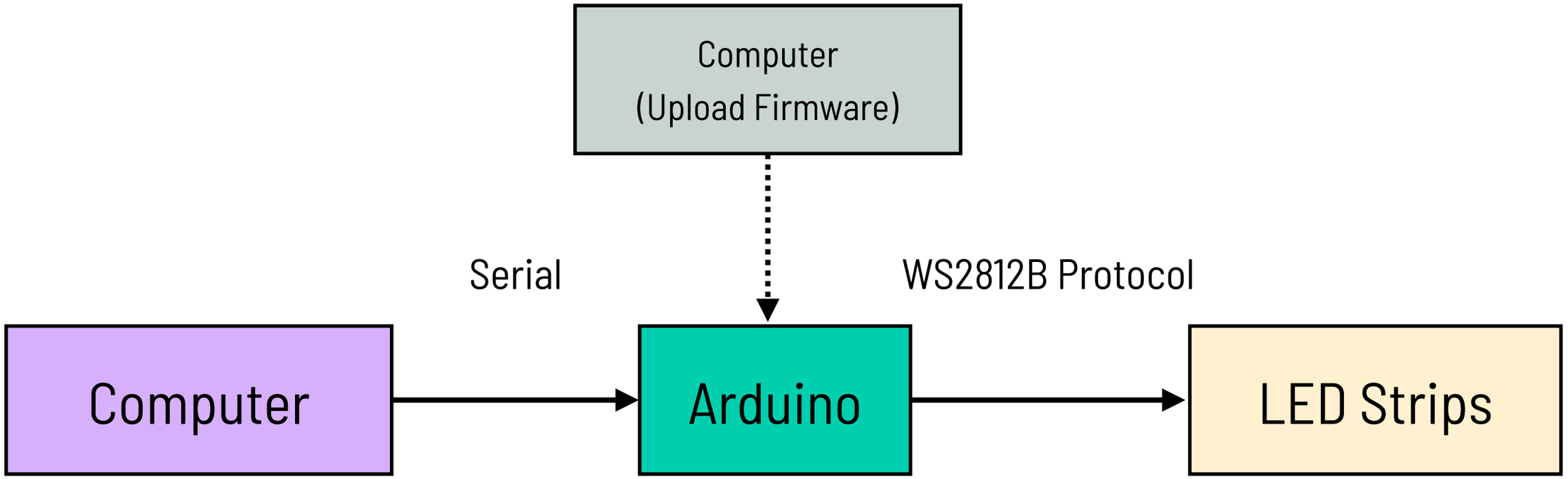
Light arrays & controlling many lights at once

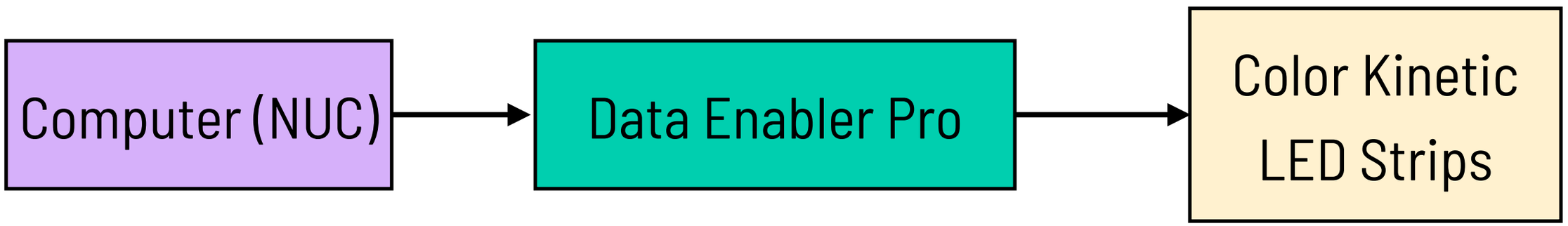
WS2812B Protocol & LEDs

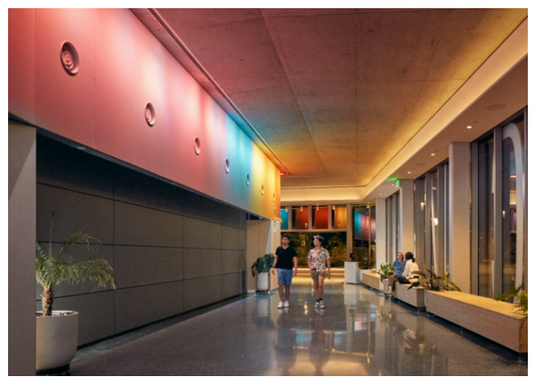

Computer
(Upload Firmware)
Arduino

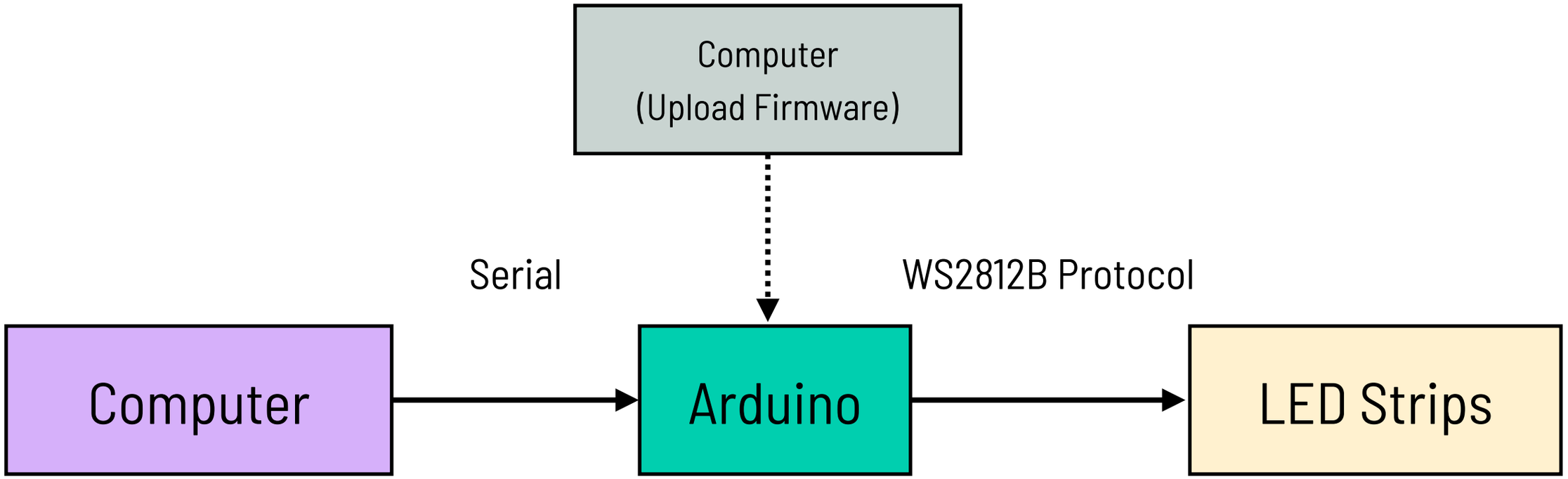

Computer
(Upload Firmware)
WS2812B Protocol
Arduino
LED Strips

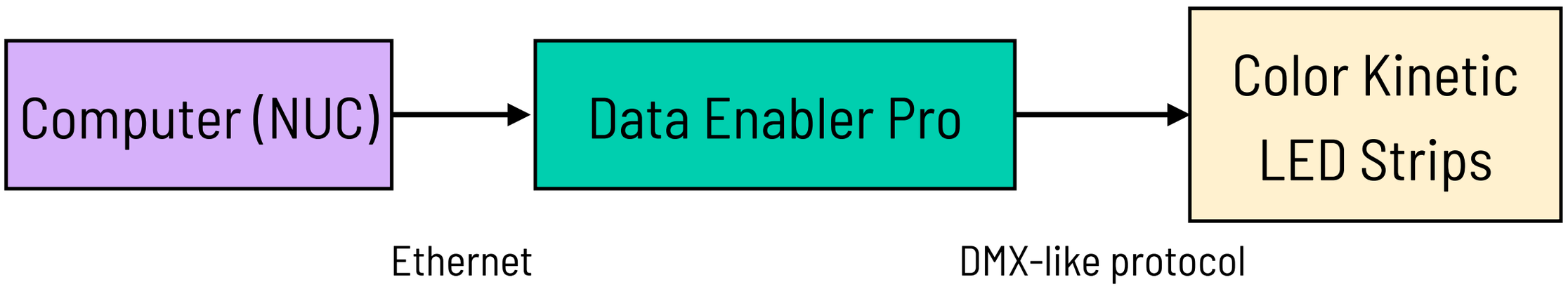

Computer
(Upload Firmware)
Serial
WS2812B Protocol
Computer
Arduino
LED Strips

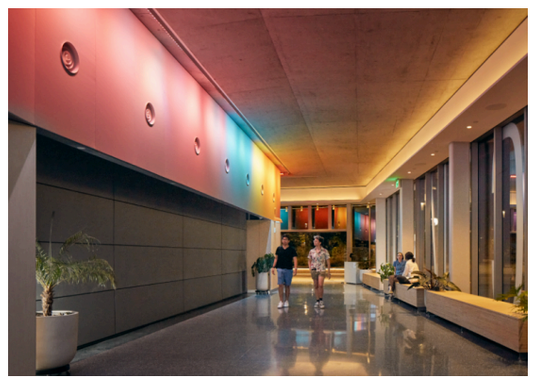

Computer
(Upload Firmware)
Serial
WS2812B Protocol
Computer
Arduino
LED Strips

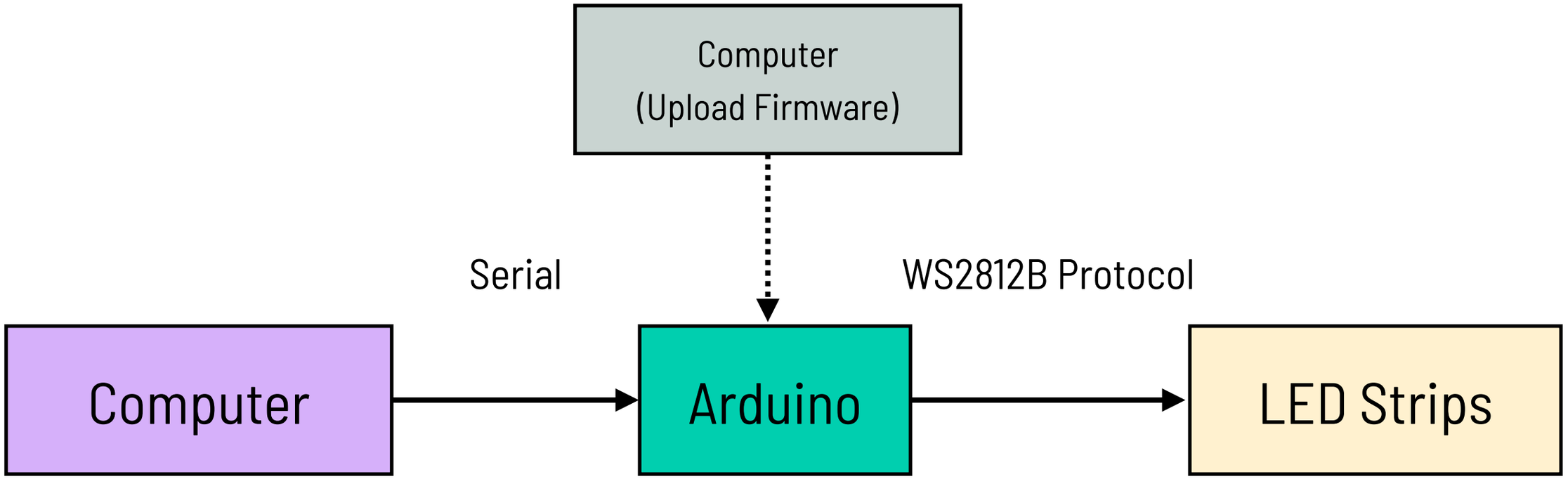

Computer (NUC)
Data Enabler Pro
Color Kinetic
LED Strips

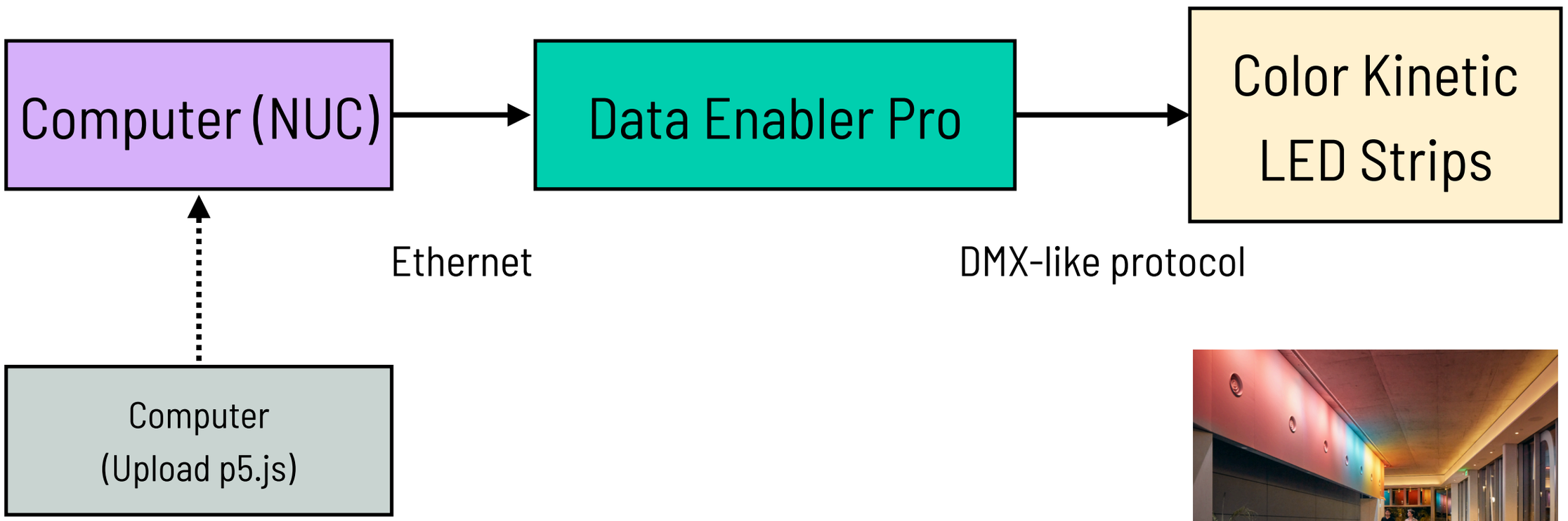


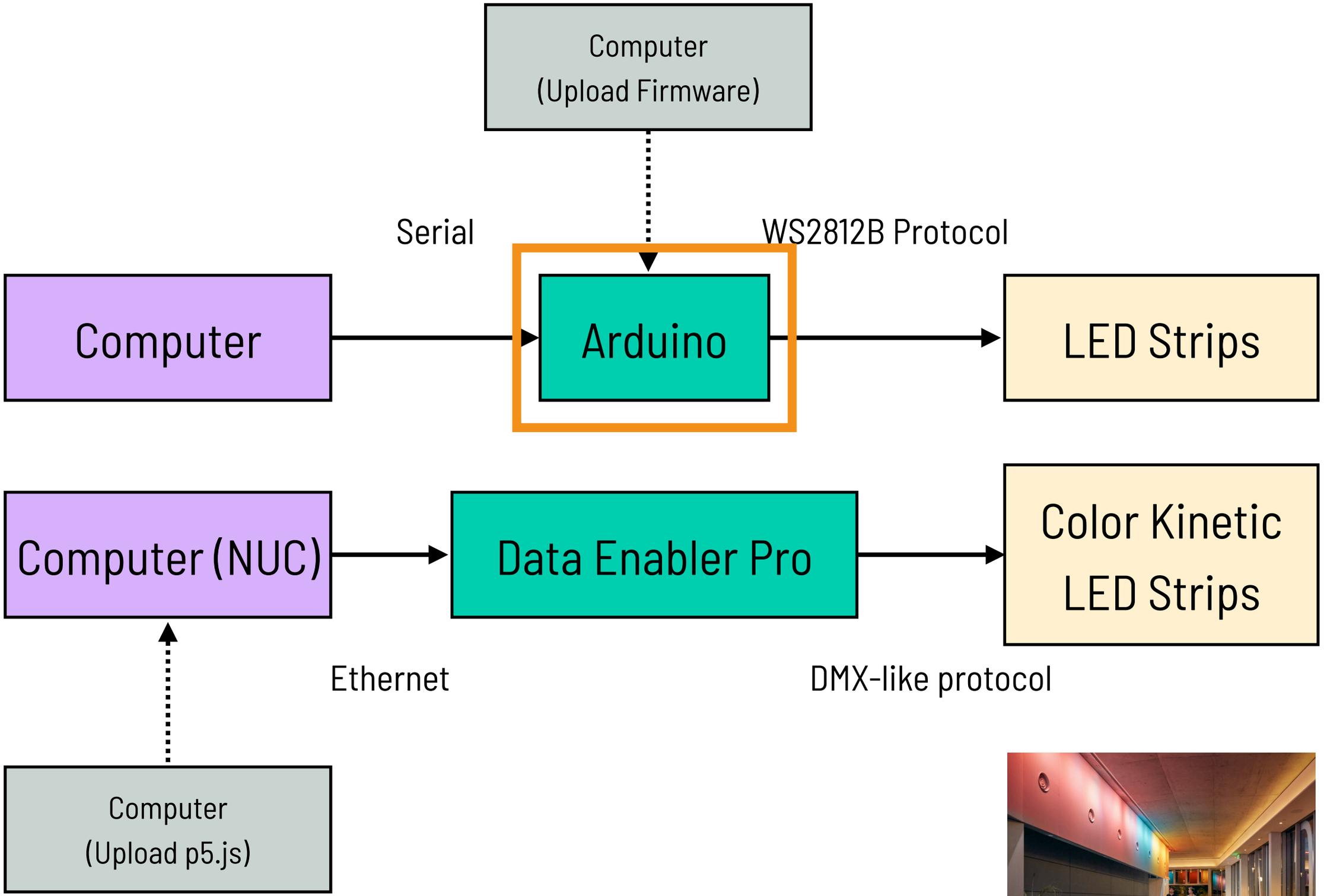

Computer
(Upload Firmware)
Serial
WS2812B Protocol
Computer
Arduino
LED Strips


Computer (NUC)
Data Enabler Pro
Color Kinetic
LED Strips
Ethernet
DMX-like protocol




Computer
(Upload Firmware)
Serial
WS2812B Protocol
Computer
Arduino
LED Strips


Computer (NUC)
Data Enabler Pro
Color Kinetic
LED Strips
Ethernet
DMX-like protocol
Computer
(Upload p5.js)


Computer
(Upload Firmware)
Serial
WS2812B Protocol
Computer
Arduino
LED Strips
Computer (NUC)
Data Enabler Pro
Color Kinetic
LED Strips
Ethernet
DMX-like protocol
Computer
(Upload p5.js)

Computer
(Upload Firmware)

Functional

Timing-Based Effects

Serial

WS2812B Protocol

Computer

Arduino

LED Strips

Computer (NUC)

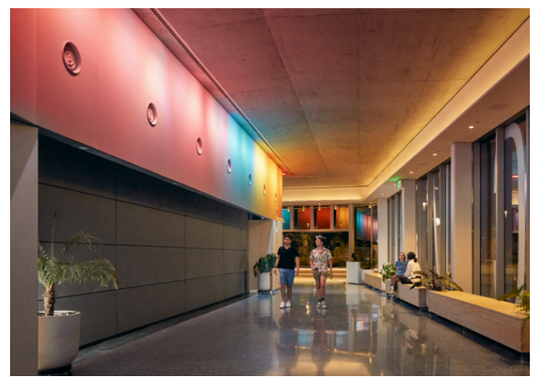
Data Enabler Pro

Color Kinetic
LED Strips

Ethernet

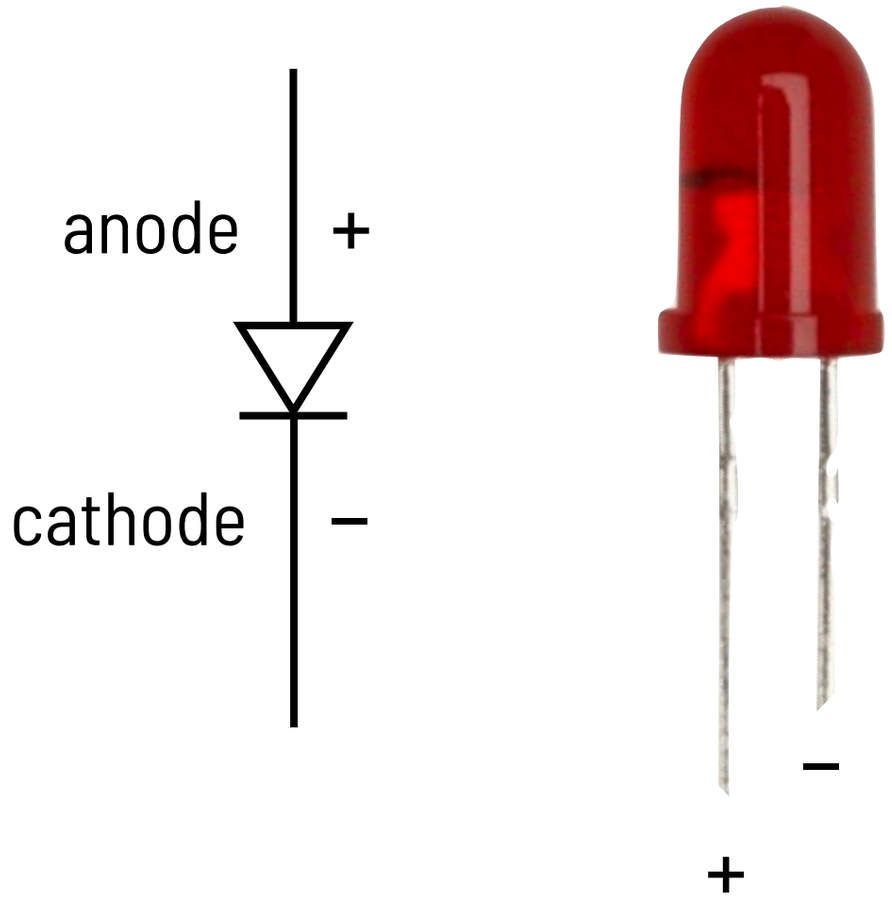
DMX-like protocol

Computer
(Upload p5.js)



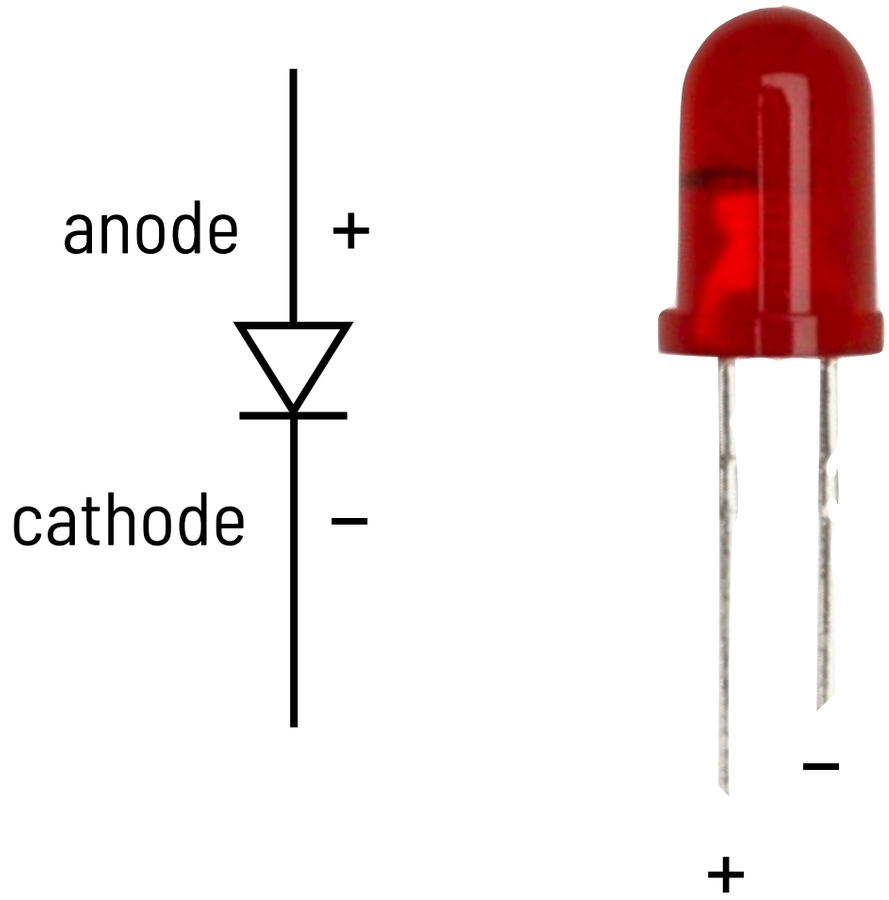
Recap

The Humble Diode



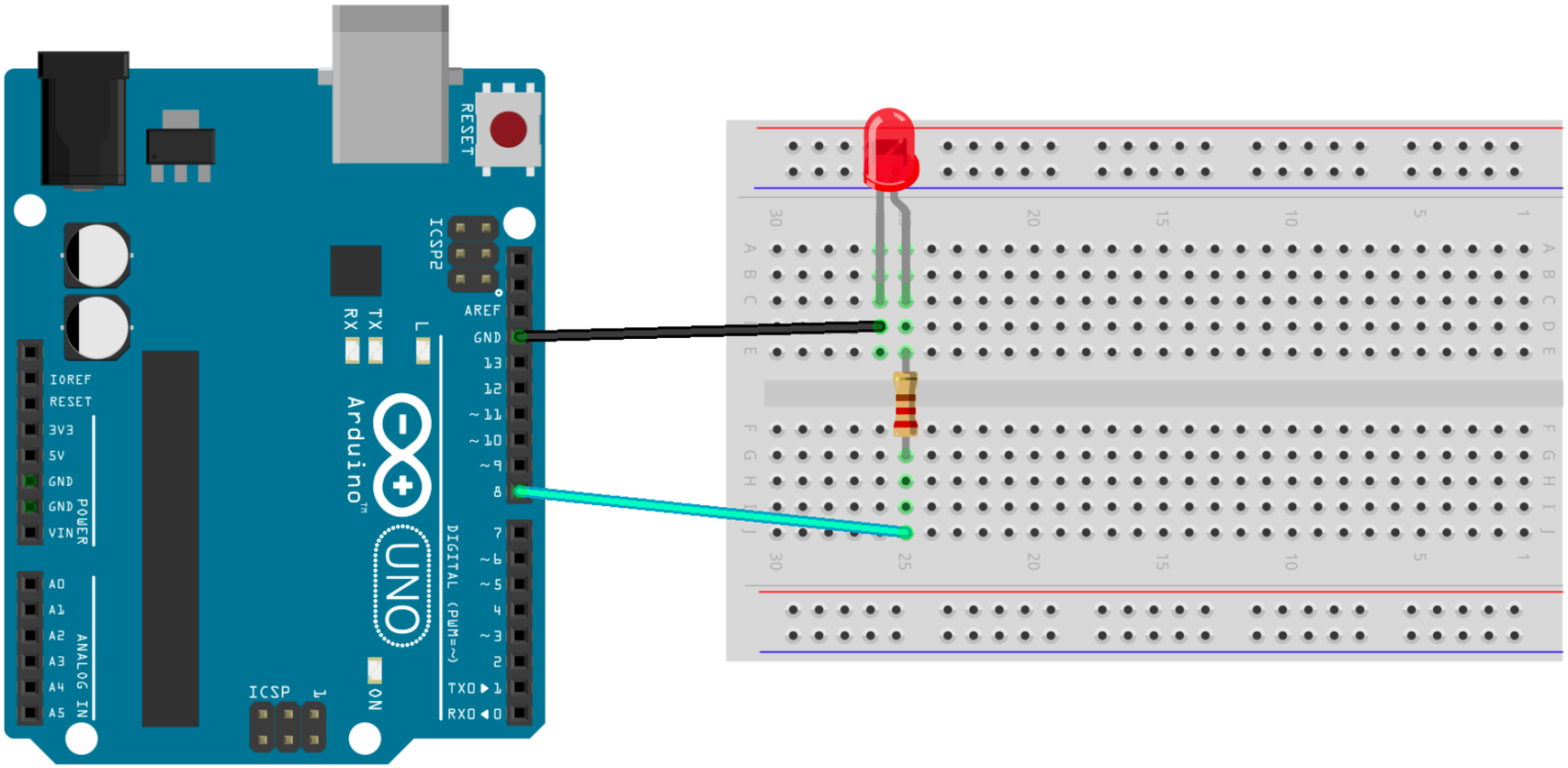
Recap

The Humble Light Emitting Diode



Recap

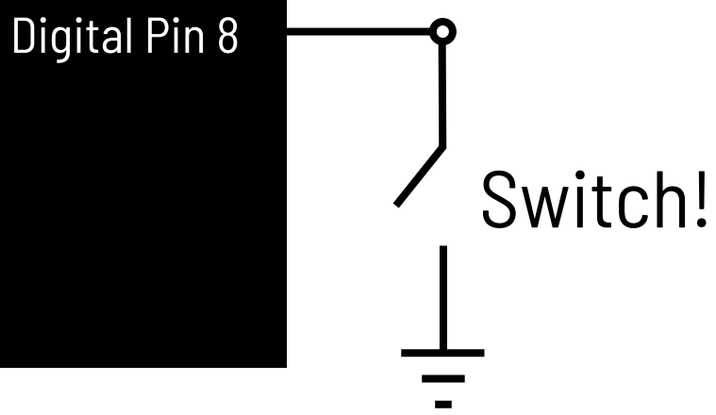
Output



Recap

Input

```
int inPin = 8;      // pushbutton connected to digital pin 8
int val = 0;        // variable to store the read value

void setup()
{
  pinMode(inPin, INPUT);    // sets the digital pin 8 as input
}

void loop()
{
  val = digitalRead(inPin); // read the input pin
}
```



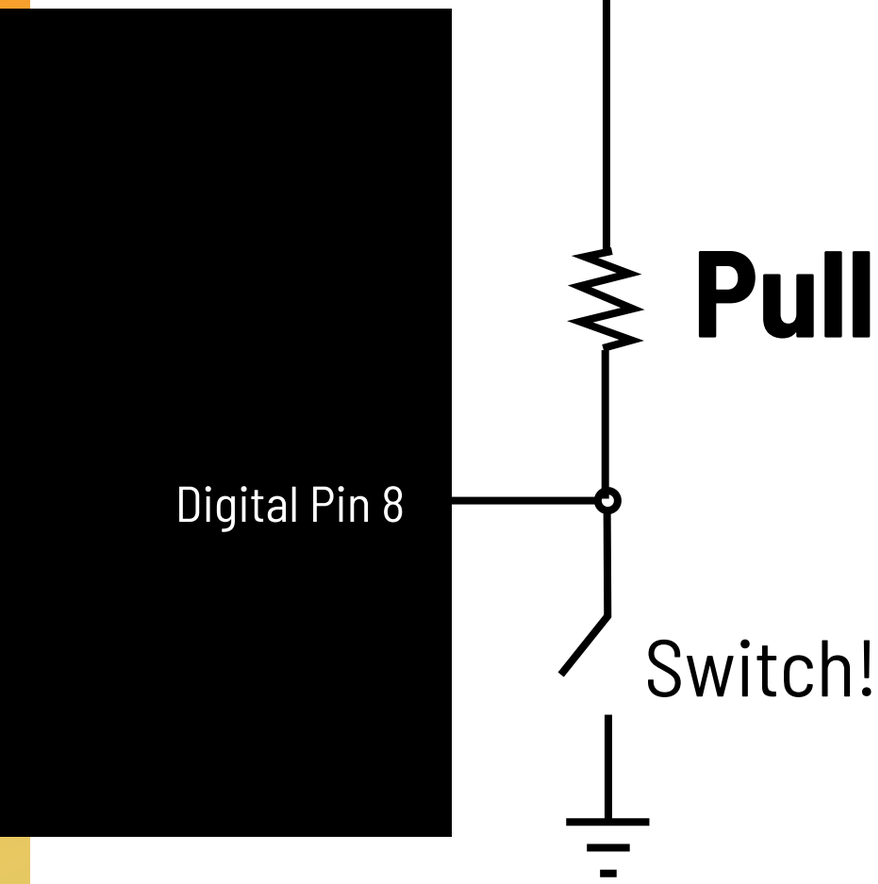
MINIQUIZ!

1. What is the value of **val** when the button is pressed
2. What about when the button is not pressed?
3. What are 2 ways of fixing this?

Recap

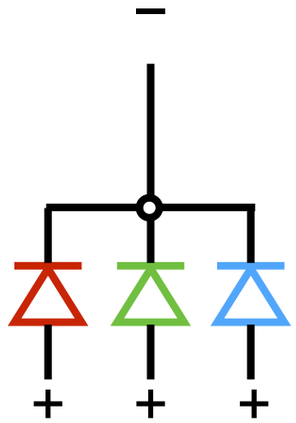
Input

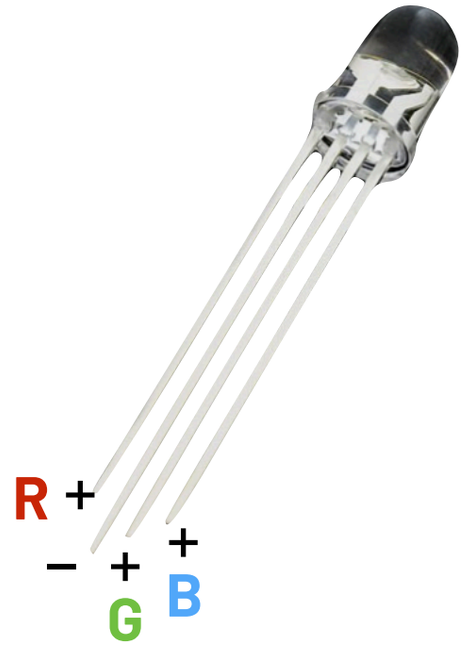
```
int inPin = 8;      // pushbutton connected to digital pin 8
int val = 0;        // variable to store the read value

void setup()
{
  pinMode(inPin, INPUT_PULLUP); // sets the digital pin 8 as input
}

void loop()
{
  val = digitalRead(inPin); // read the input pin
}
```



Recap

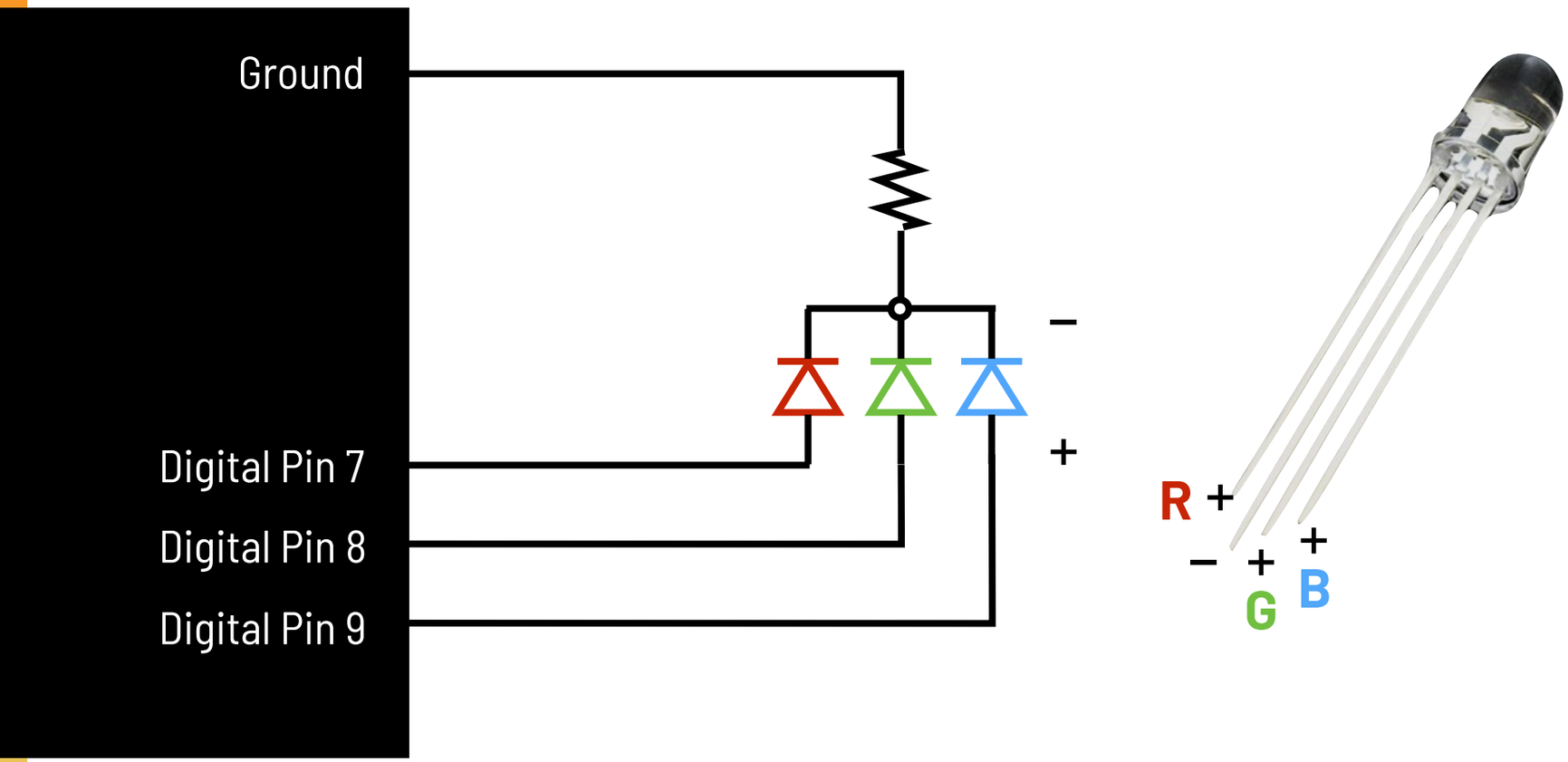
Single RGB LEDs





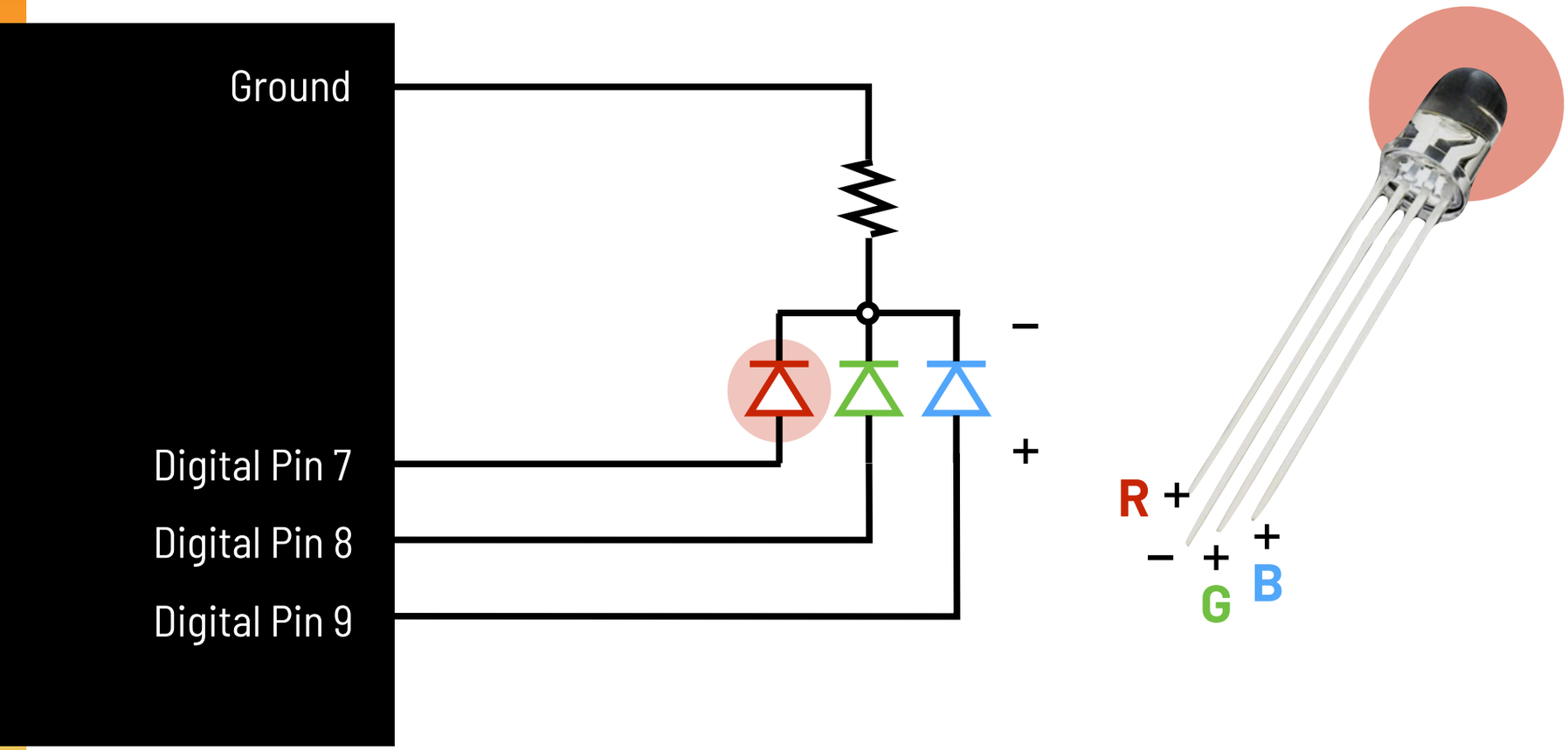
Recap

Single RGB LEDs



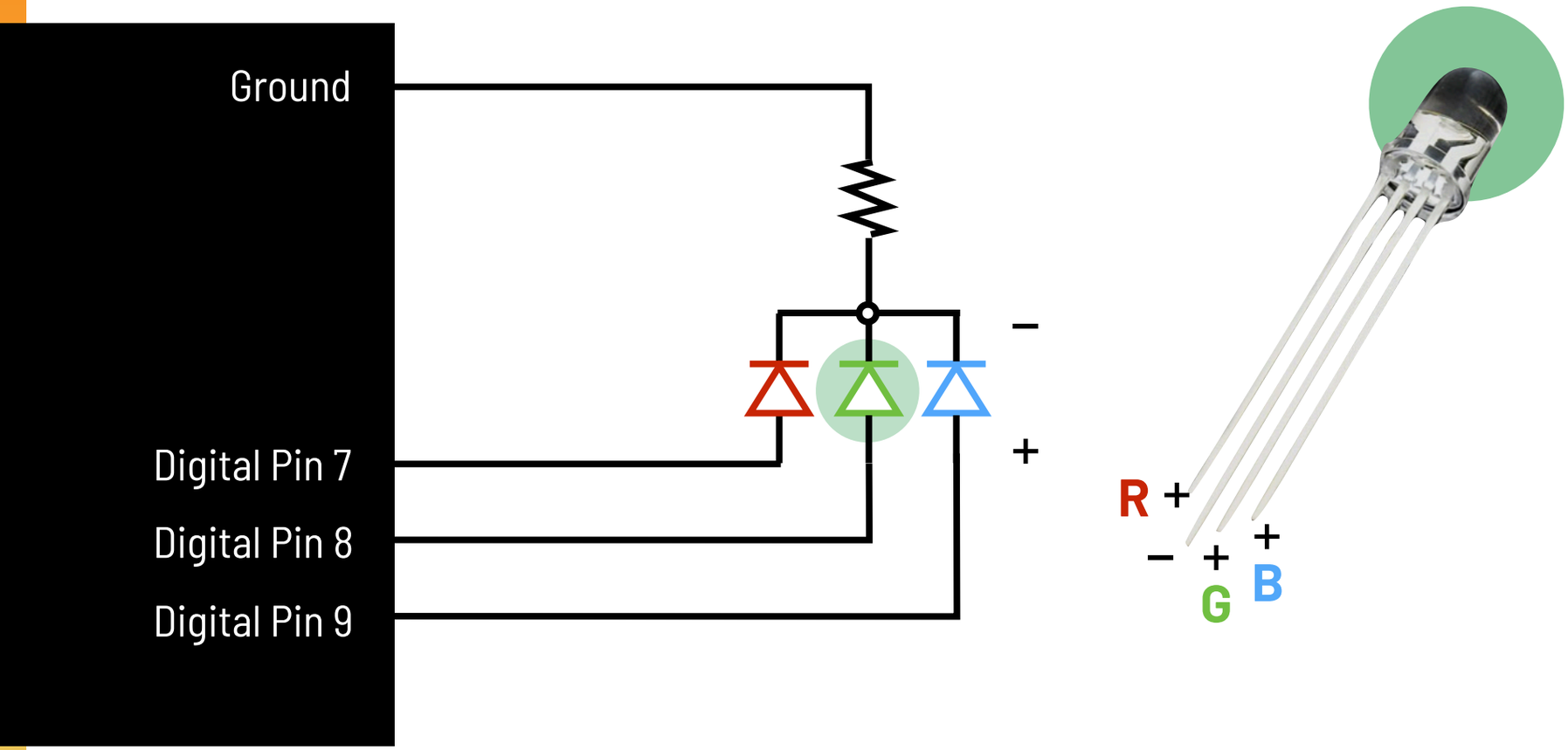
Recap

Single RGB LEDs



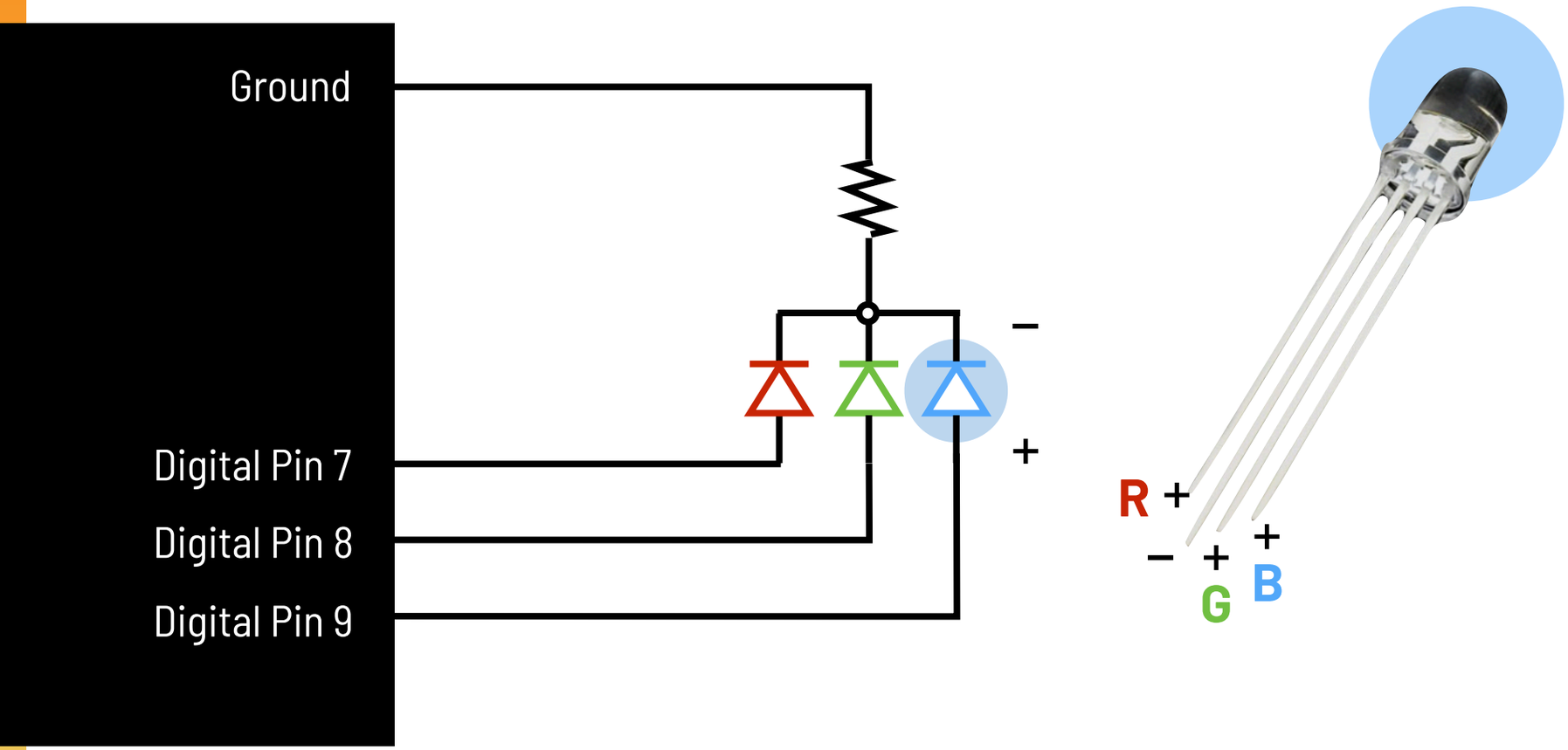
Recap

Single RGB LEDs



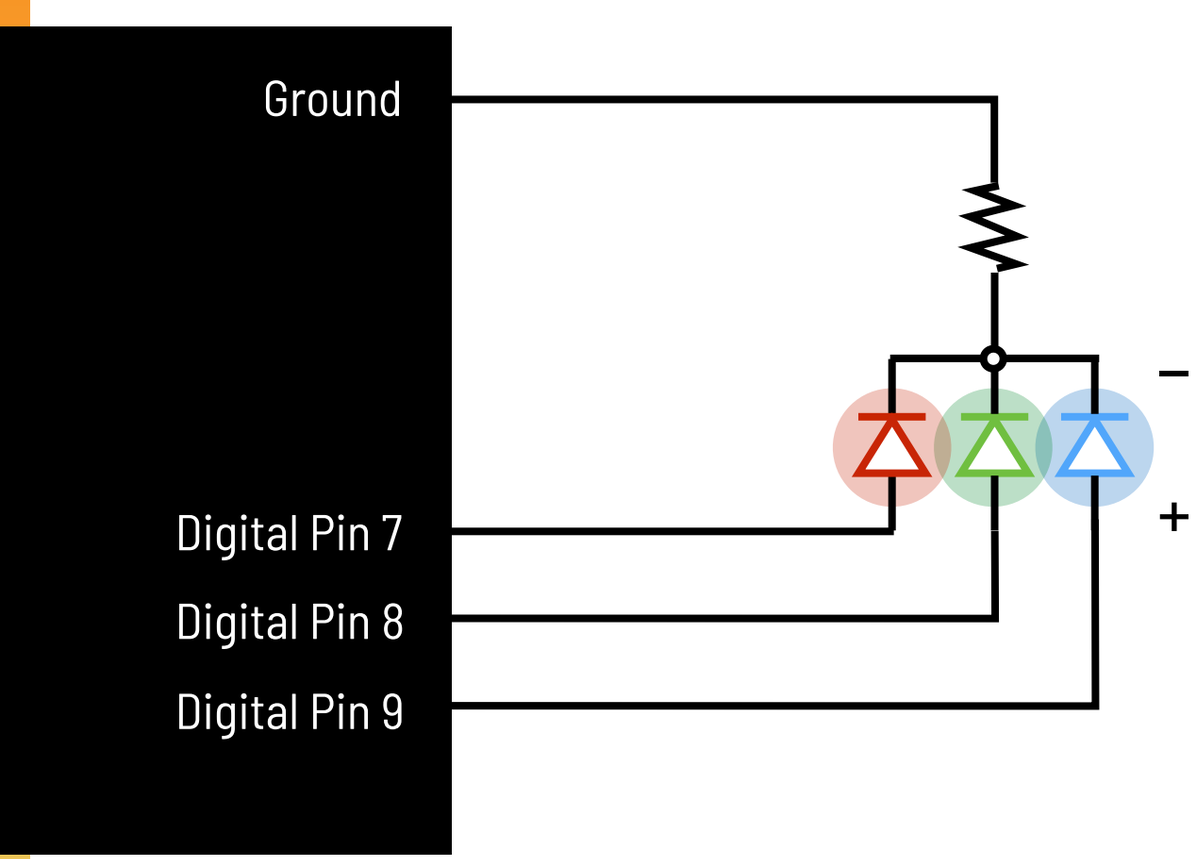
Recap

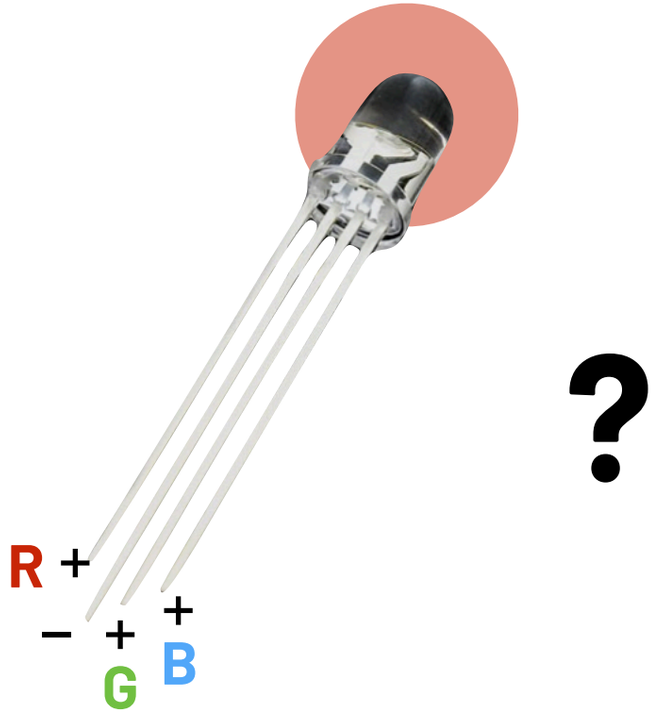
Single RGB LEDs



Recap

Single RGB LEDs





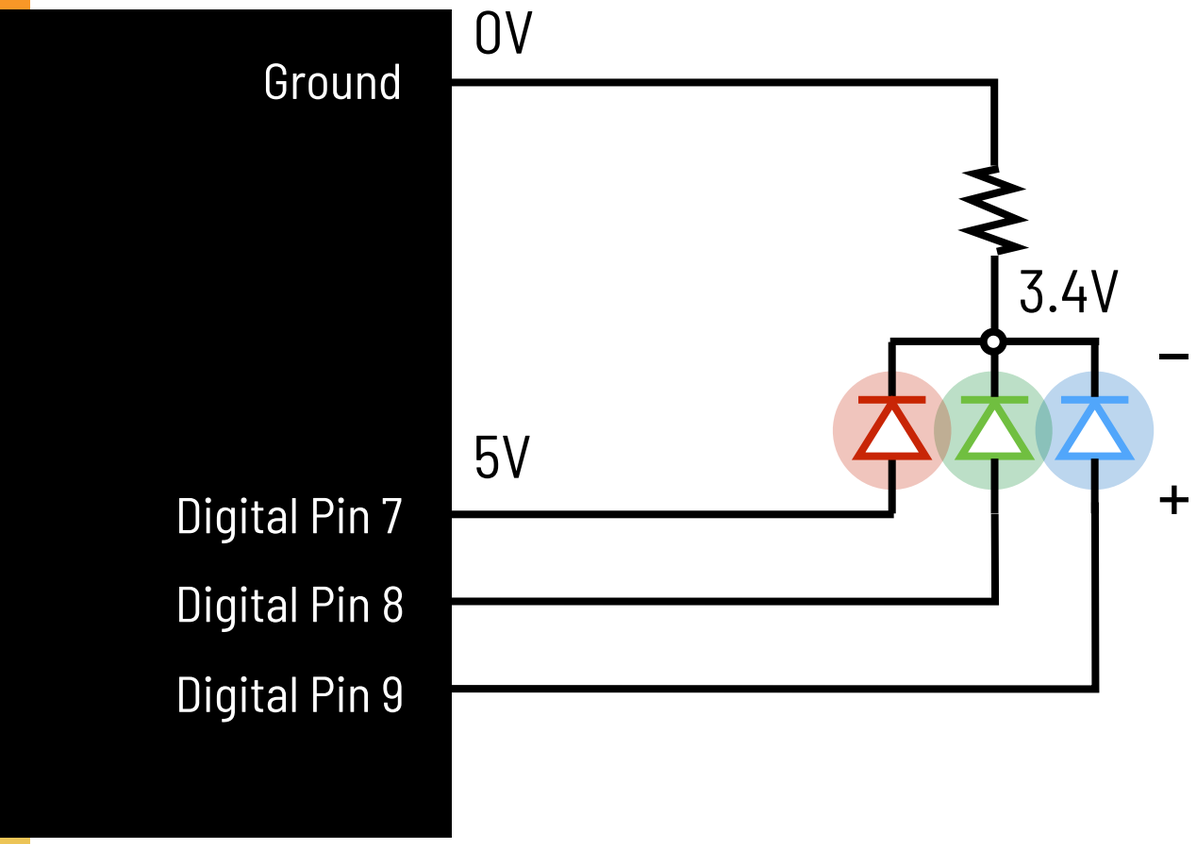
?

LED COLORS AND MATERIALS

Color	Wavelength Range (nm)	Forward Voltage (V)	Material
Ultraviolet	< 400	3.1 - 4.4	Aluminium nitride (AlN) Aluminium gallium nitride (AlGaN) Aluminium gallium indium nitride (AlGaInN)
Violet	400 - 450	2.8 - 4.0	Indium gallium nitride (InGaN)
Blue	450 - 500	2.5 - 3.7	Indium gallium nitride (InGaN) Silicon carbide (SiC)
Green	500 - 570	1.9 - 4.0	Gallium phosphide (GaP) Aluminium gallium indium phosphide (AlGaInP) Aluminium gallium phosphide (AlGaP)
Yellow	570 - 590	2.1 - 2.2	Gallium arsenide phosphide (GaAsP) Aluminium gallium indium phosphide (AlGaInP) Gallium phosphide (GaP)
Orange / Amber	590 - 610	2.0 - 2.1	Gallium arsenide phosphide (GaAsP) Aluminium gallium indium phosphide (AlGaUInP) Gallium phosphide (GaP)
Red	610 - 760	1.6 - 2.0	Aluminium gallium arsenide (AlGaAs) Gallium arsenide phosphide (GaAsP) Aluminium gallium indium phosphide (AlGaInP) Gallium phosphide (GaP)
Infrared	> 760	> 1.9	Gallium arsenide (GaAs) Aluminium gallium arsenide (AlGaAs)

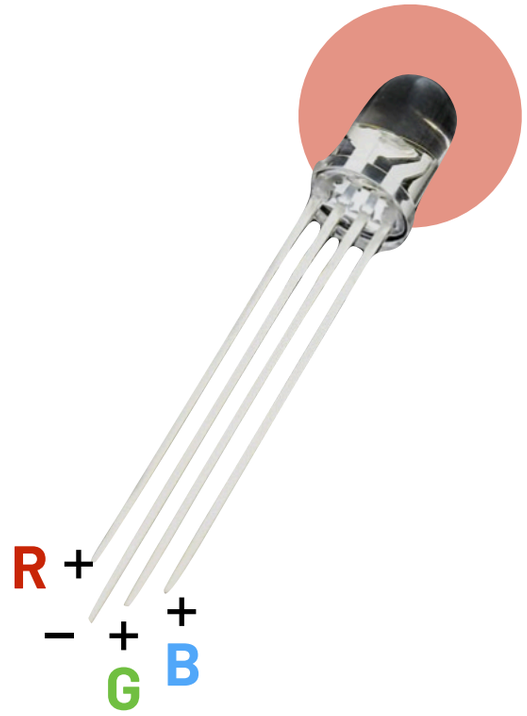
Recap

Single RGB LEDs



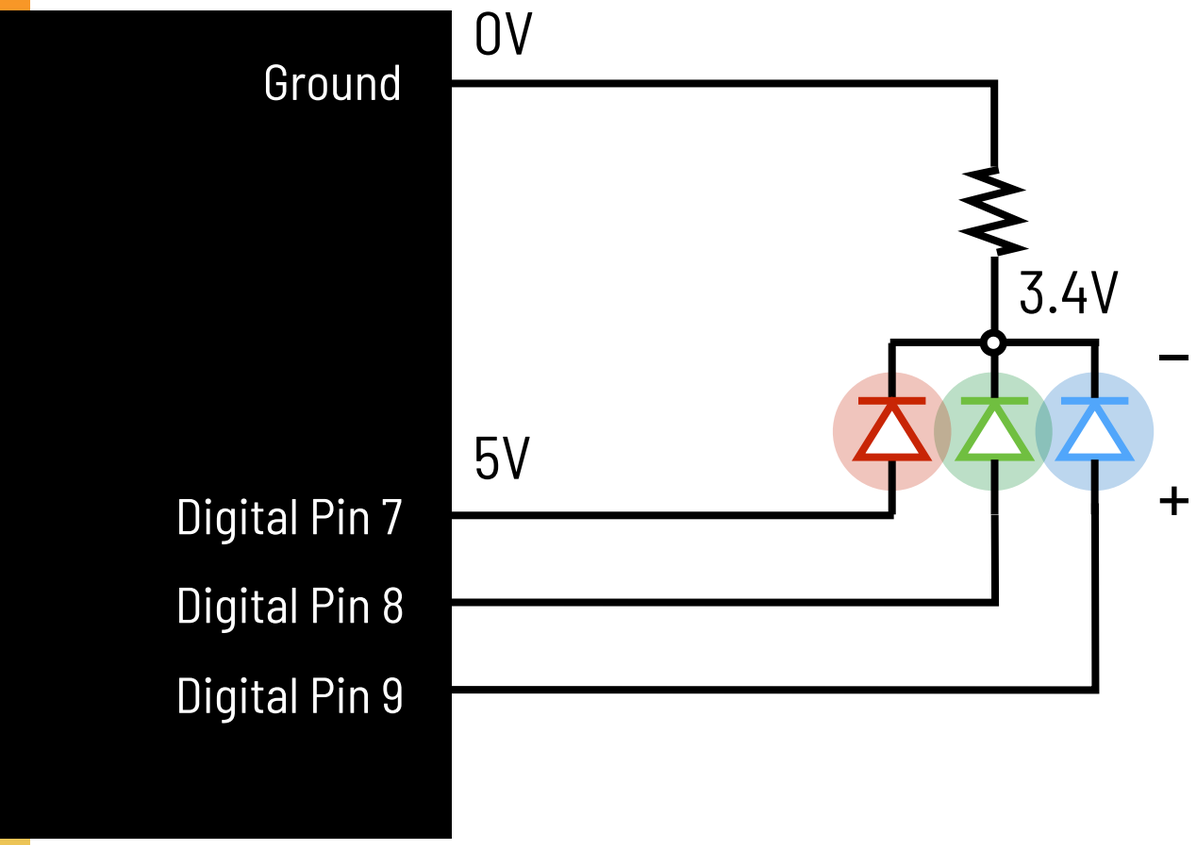
LED Forward Voltage Drop

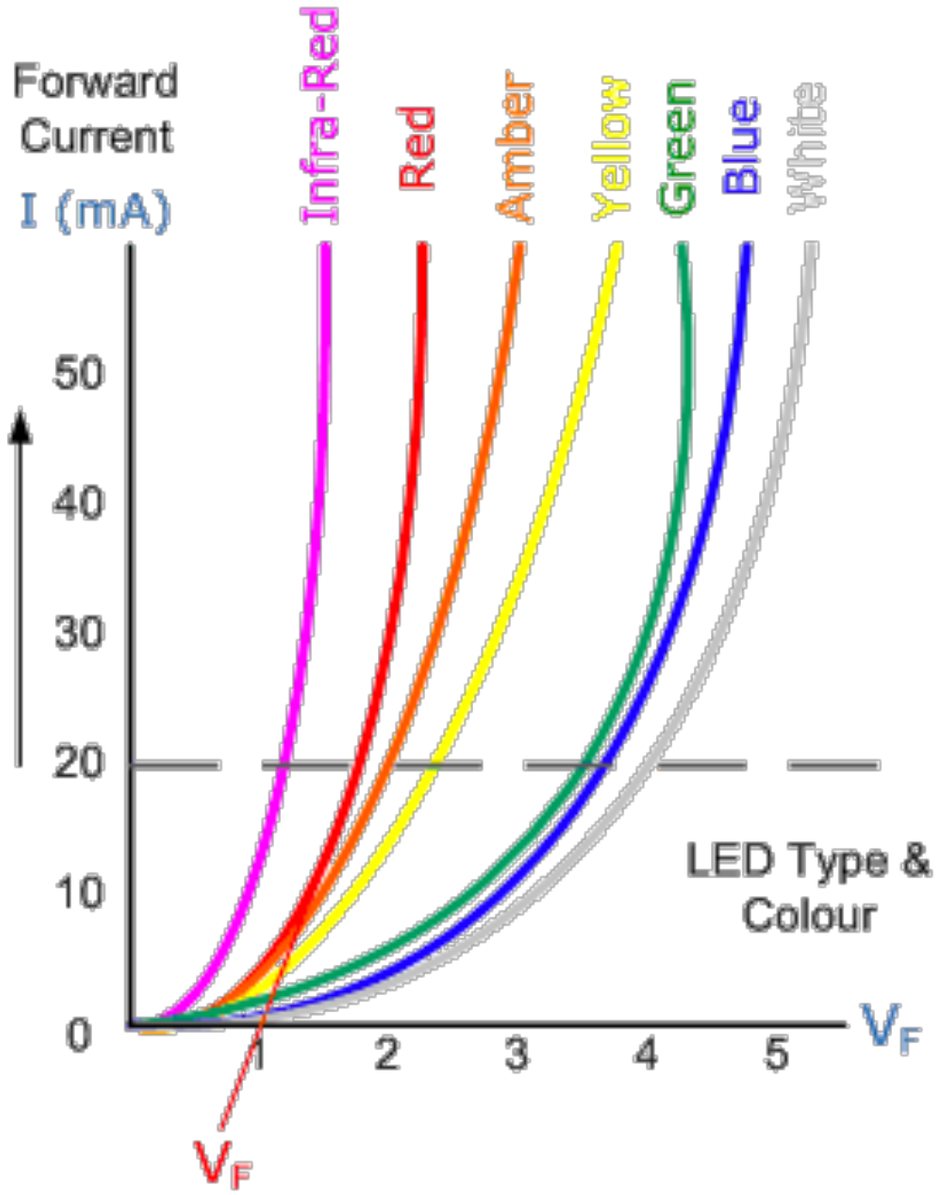
Red ~ 1.6V
Green ~ 1.9V
Blue ~ 2.5V



Recap

Single RGB LEDs



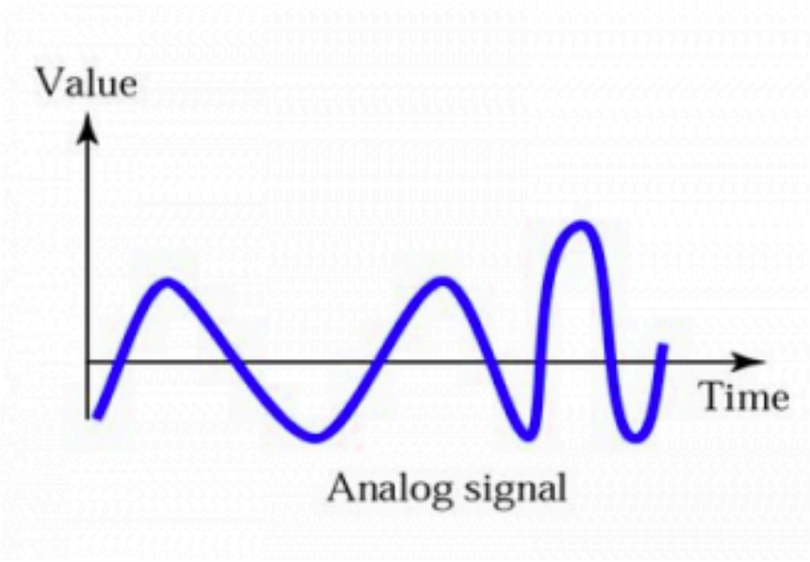


Digital vs Analog Circuits

Analog Circuits

Range of voltages

Usually requires math!

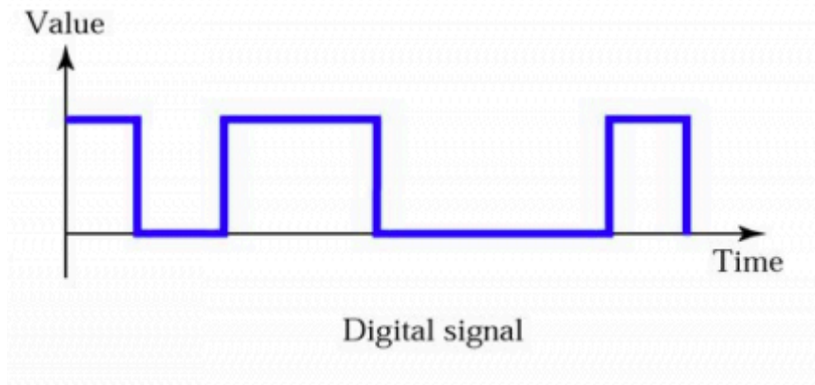


Digital Circuits

Usually 2 distinct voltages (high & low)

5v and 0v (roughly)

0110010101101101011000010110100101101100
0111010101110011011000100110100101110100
0111001100100001

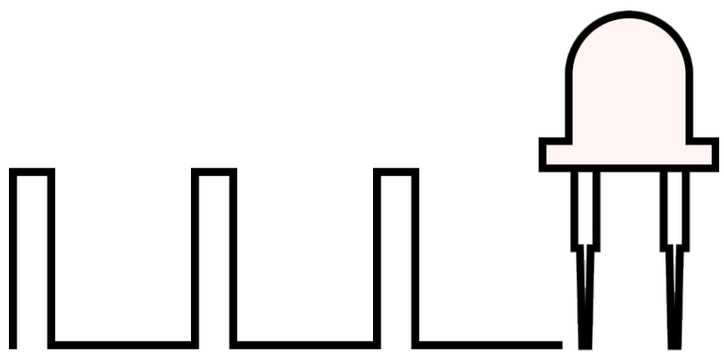


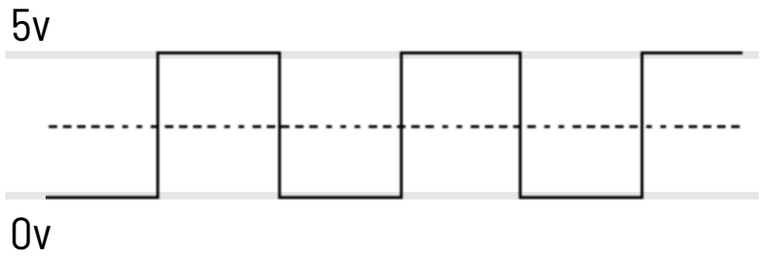
Pulse Width Modulation

PWM!

Digital is only 0v or 5v, so how do we get values in between?

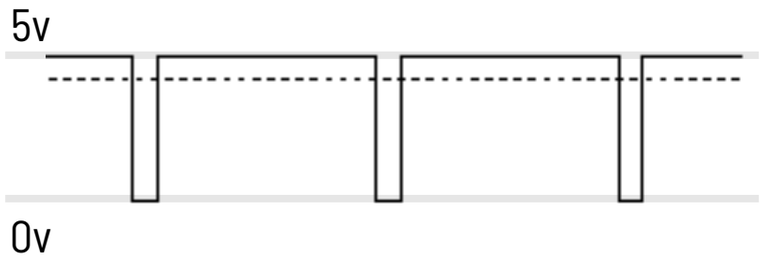
PWM to make the LED a different brightness





50% Duty Cycle

50% of the time on,
50% of the time off



90% Duty Cycle

90% of the time on,
10% of the time off

PWM

The Fading example demonstrates the use of analog output (PWM) to fade an LED. It is available in the File->Sketchbook->Examples->Analog menu of the Arduino software.

Pulse Width Modulation, or PWM, is a technique for getting analog results with digital means. Digital control is used to create a square wave, a signal switched between on and off. This on-off pattern can simulate voltages in between full on (5 Volts) and off (0 Volts) by changing the portion of the time the signal spends on versus the time that the signal spends off. The duration of "on time" is called the pulse width. To get varying analog values, you change, or modulate, that pulse width. If you repeat this on-off pattern fast enough with an LED for example, the result is as if the signal is a steady voltage between 0 and 5v controlling the brightness of the LED.

In the graphic below, the green lines represent a regular time period. This duration or period is the inverse of the PWM frequency. In other words, with Arduino's PWM frequency at about 500Hz, the green lines would measure 2 milliseconds each. A call to analogWrite() is on a scale of 0 - 255, such that analogWrite(255) requests a 100% duty cycle (always on), and analogWrite(127) is a 50% duty cycle (on half the time) for example.

PWM

```
void loop() {
  digitalWrite(6, HIGH);
  delay(1000);
  analogWrite(6, 100);
  delay(1000);
}
```

LED Arrays

More Pins?

Arduinos only have a limited number of output.

There are different methods we can “get more outputs”

Method 1: **Multiplexing**

Method 2: Specific **Communication** Protocols

Now you’re the Array!

Scan the code or go to

bit.ly/ledarray

Enter your kerberos and fill the background color of one of the cells with a color

More Pins?

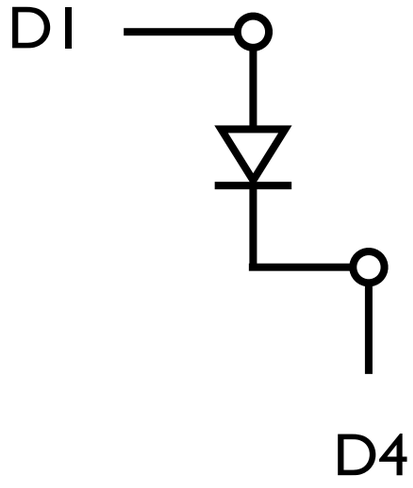
Arduinos only have a limited number of output.

There are different methods we can “get more outputs”

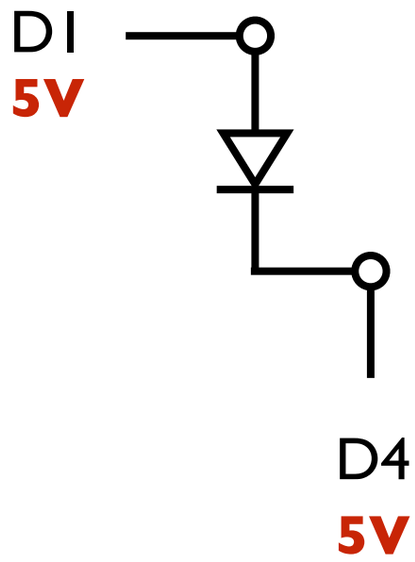
Method 1: **Multiplexing**

Method 2: Specific **Communication** Protocols

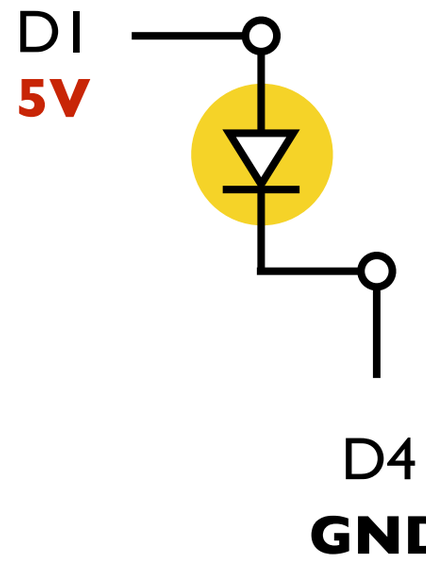
Multiplexing



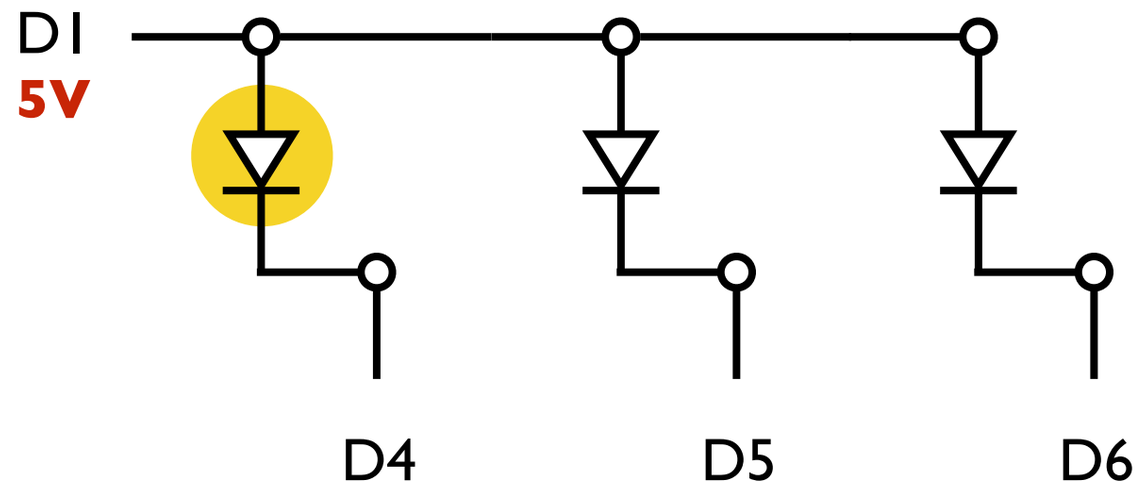
Multiplexing



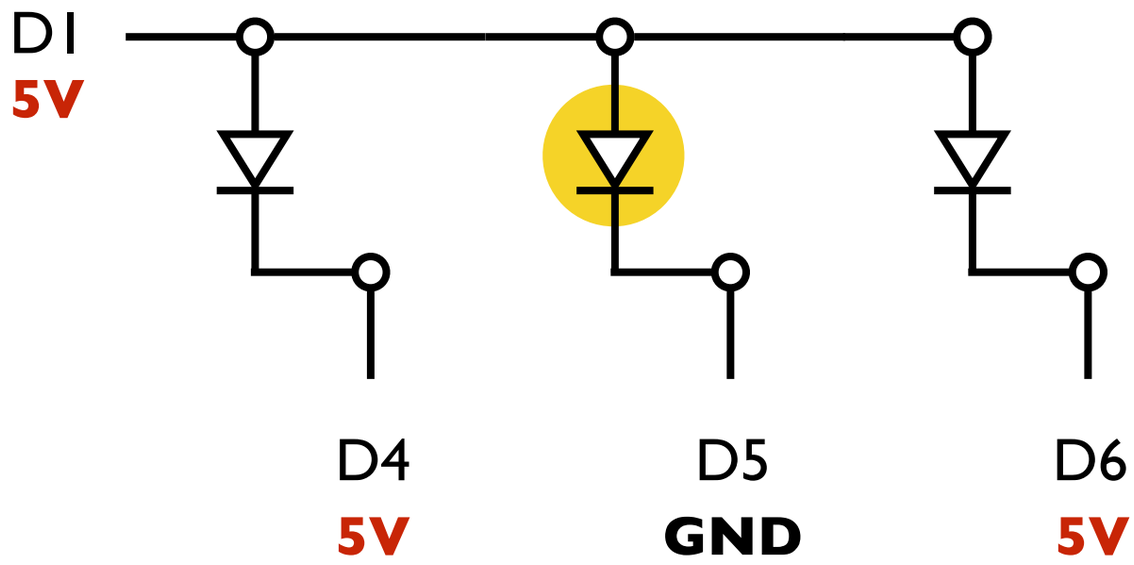
Multiplexing



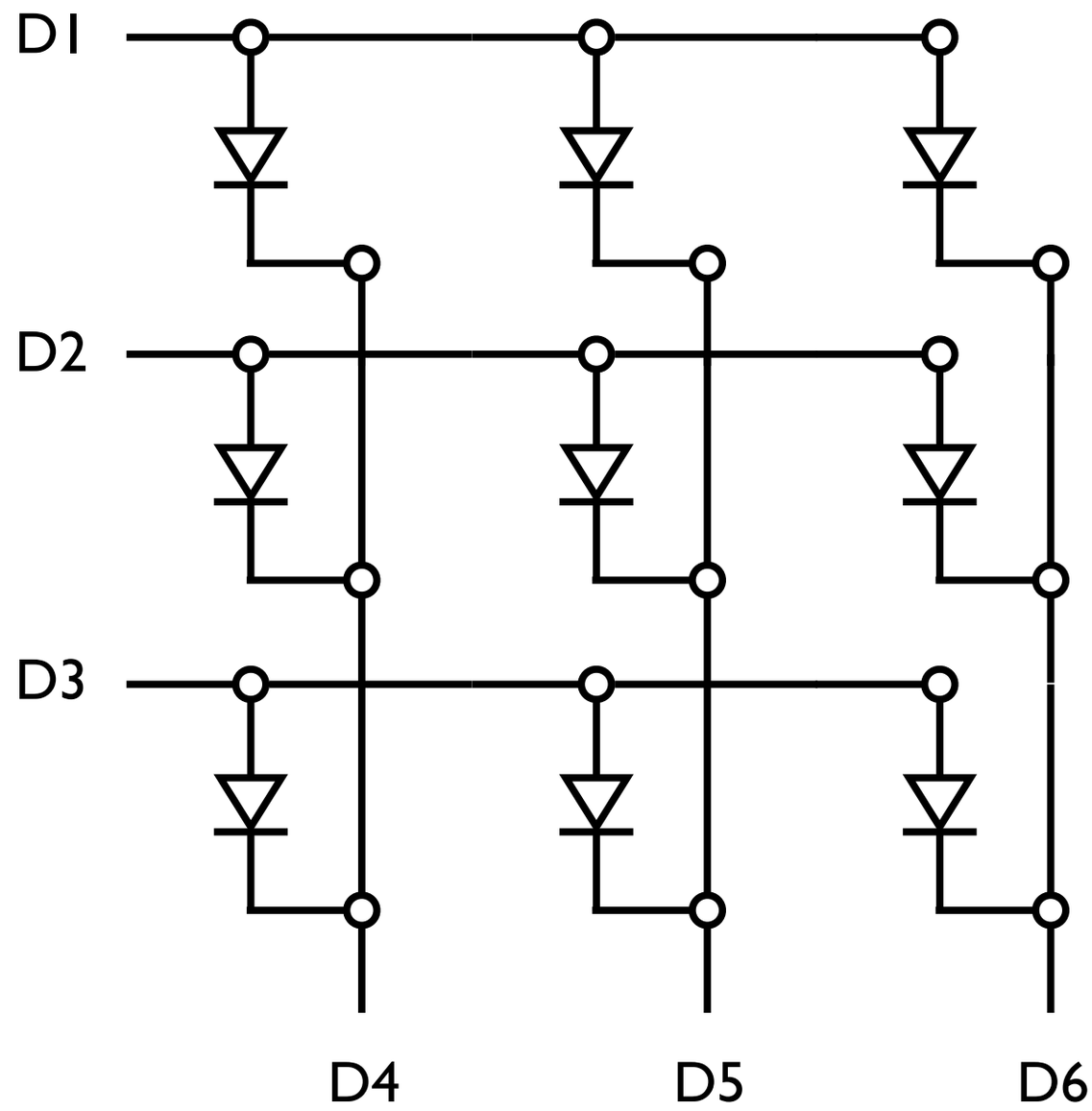
Multiplexing



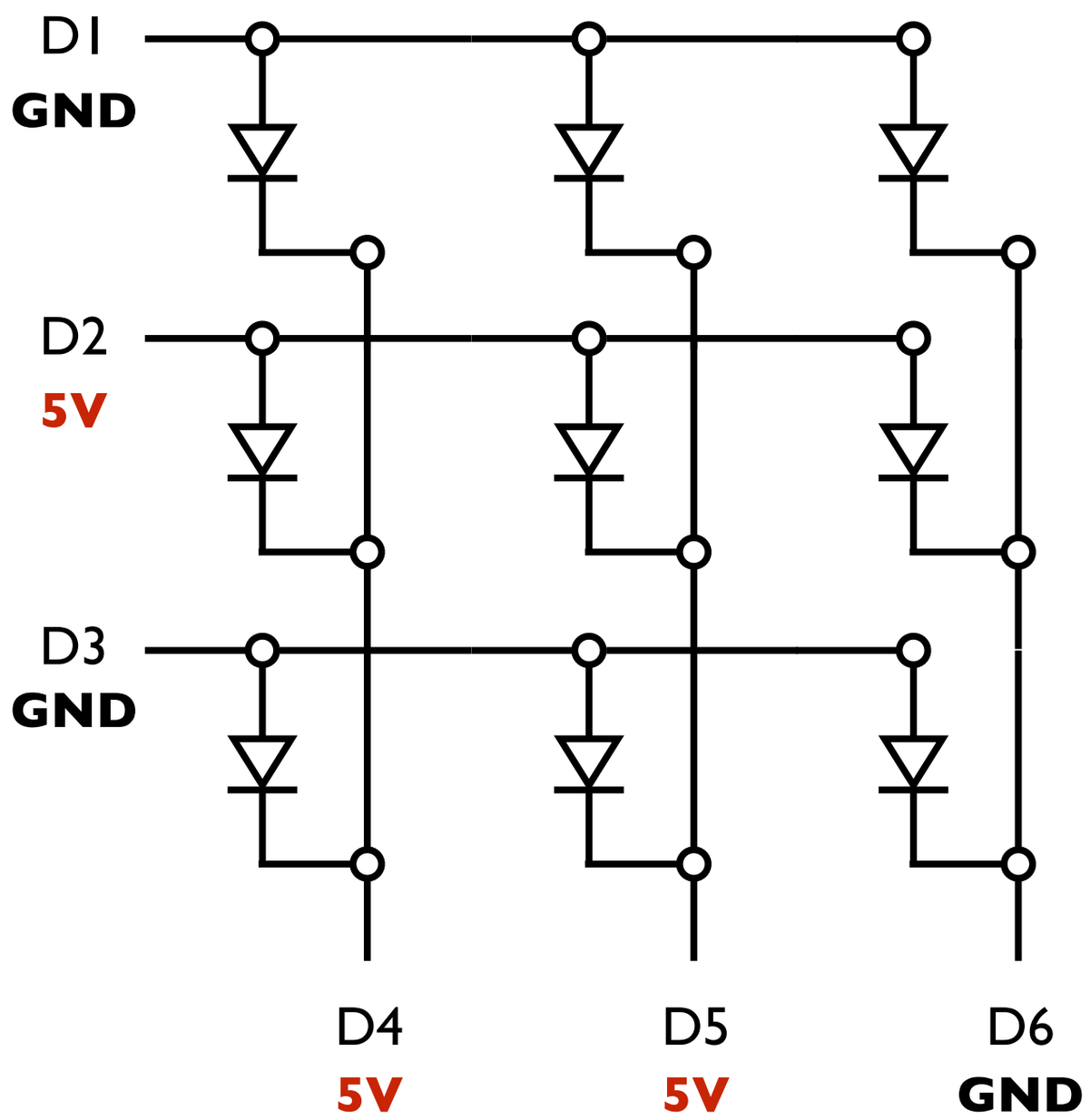
Multiplexing



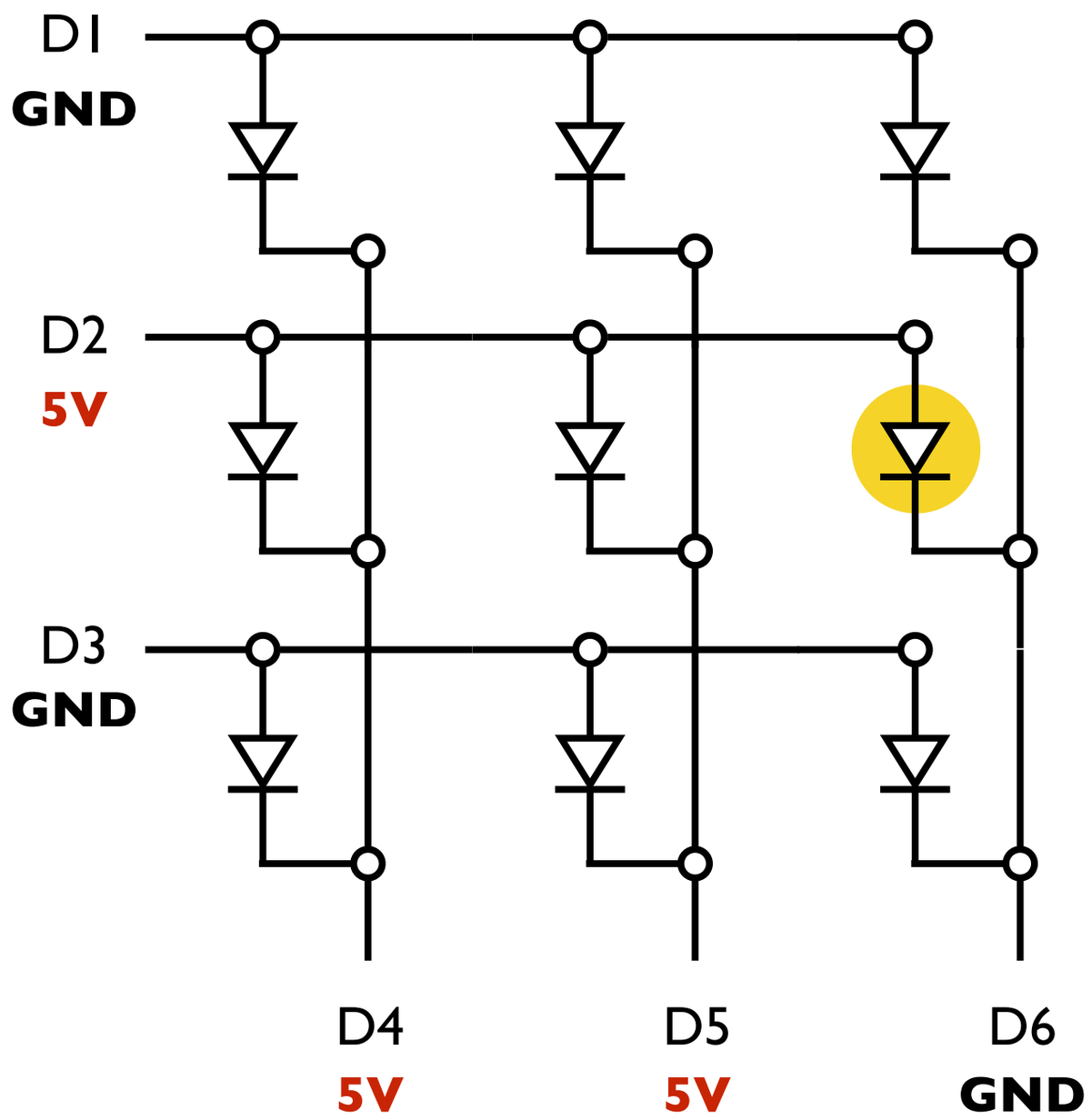
Multiplexing



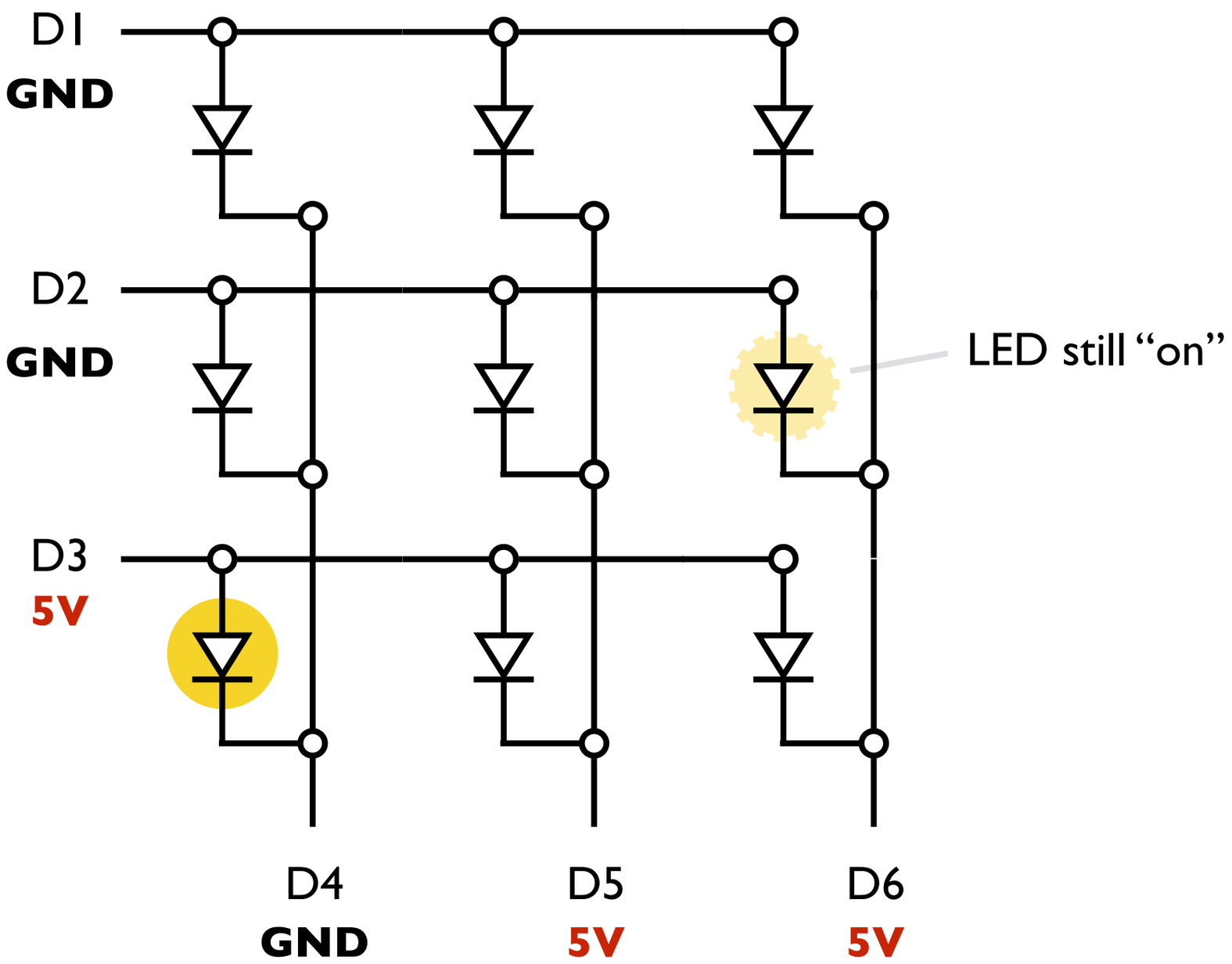
Multiplexing



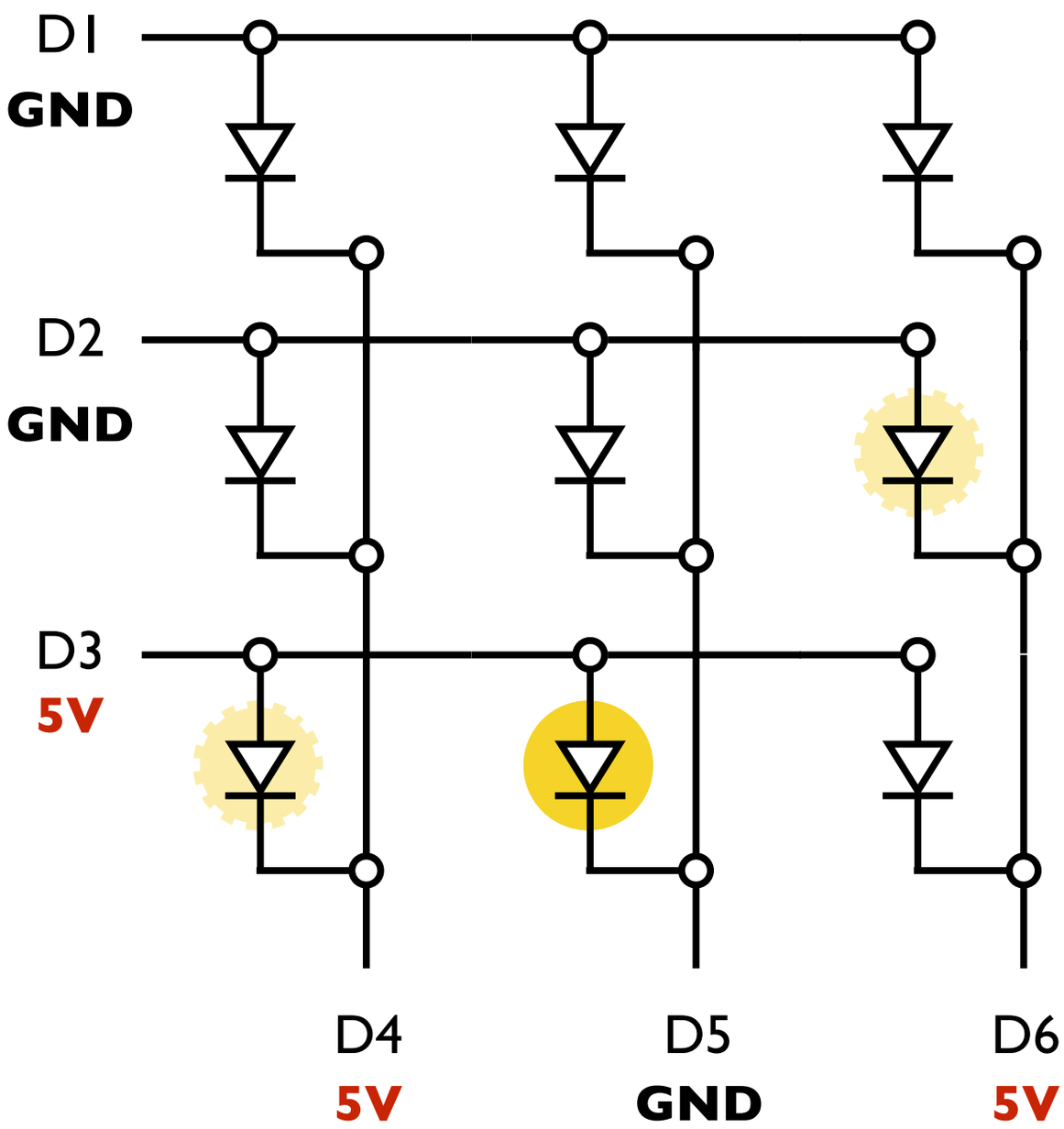
Multiplexing



Multiplexing



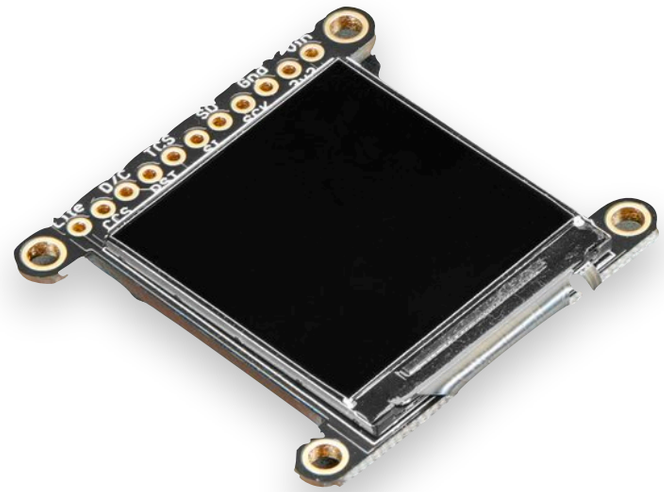
Multiplexing

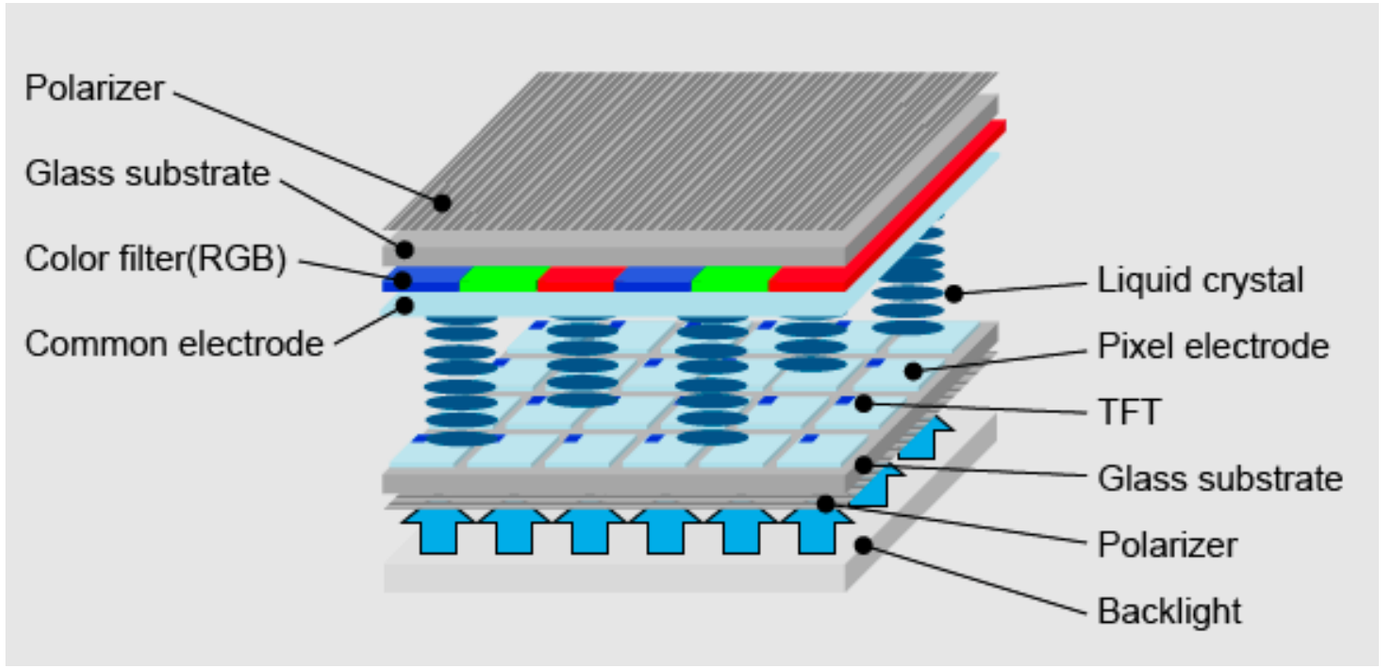


TFT Display

Thin **F**ilm **T**ransistor Displays

Breakout board/ Arduino library handles a lot of the logic!



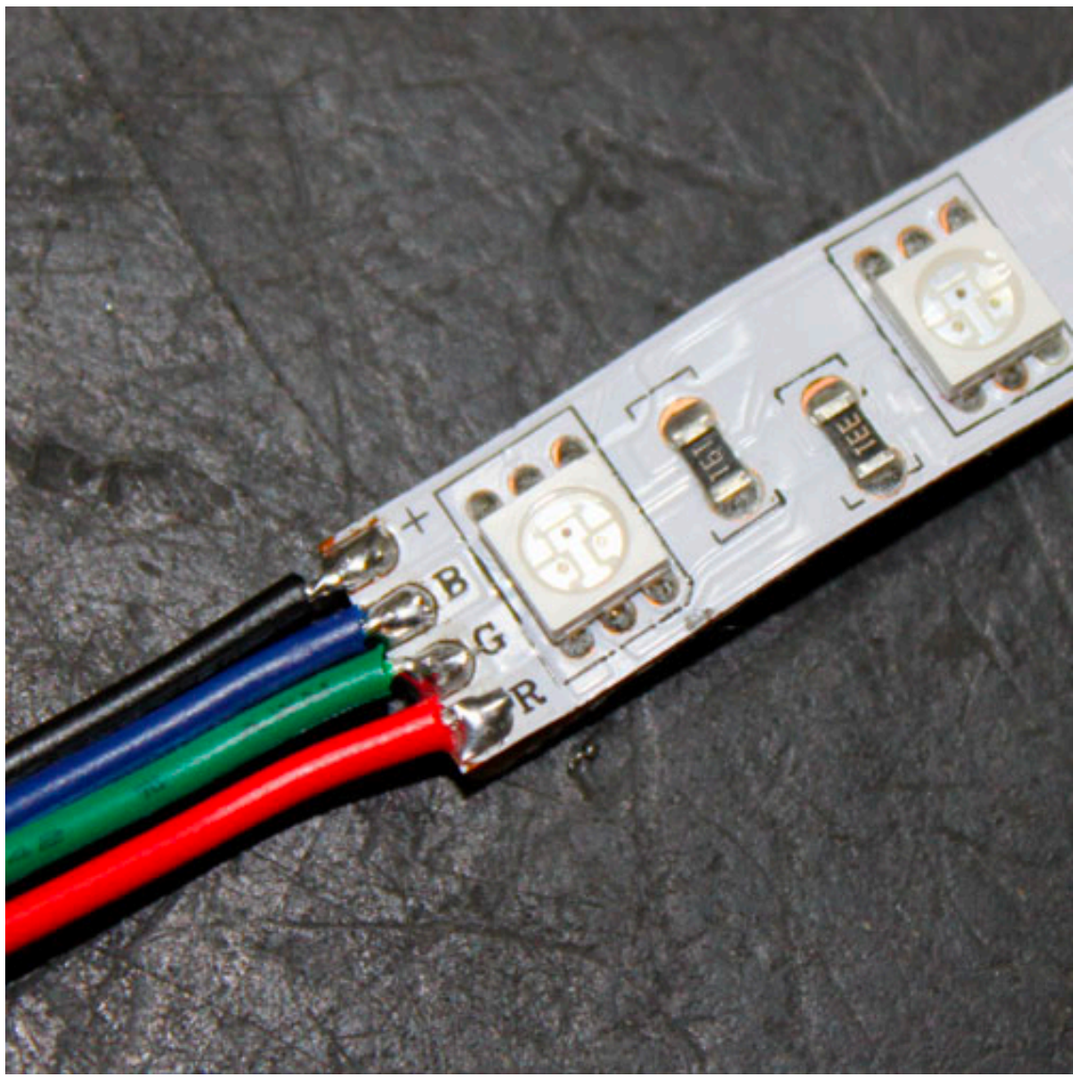


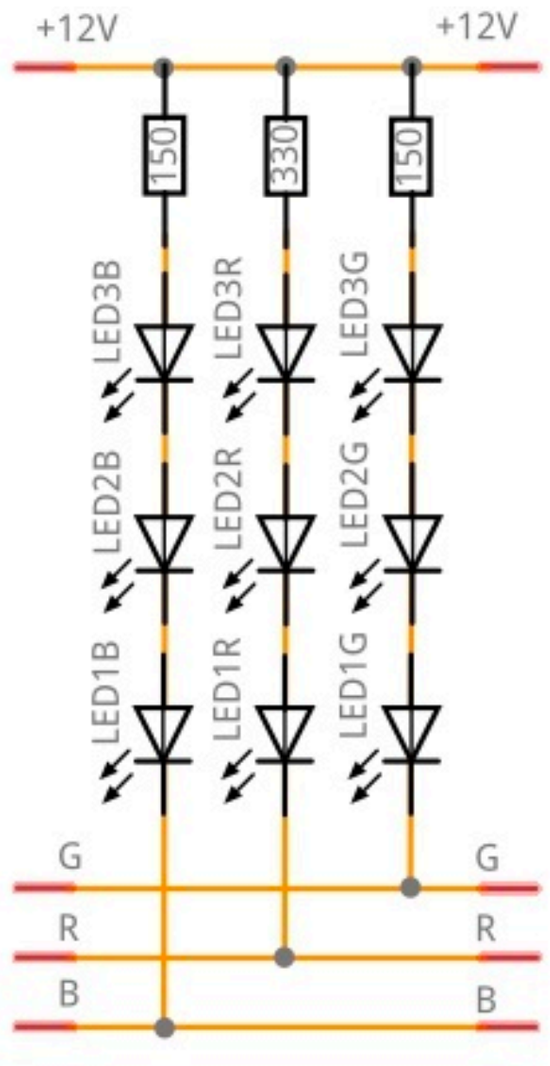
Sensitive Electronics!

Static Shielding bags are bags that prevent a build up of static electricity inside or outside the bag

LED strips

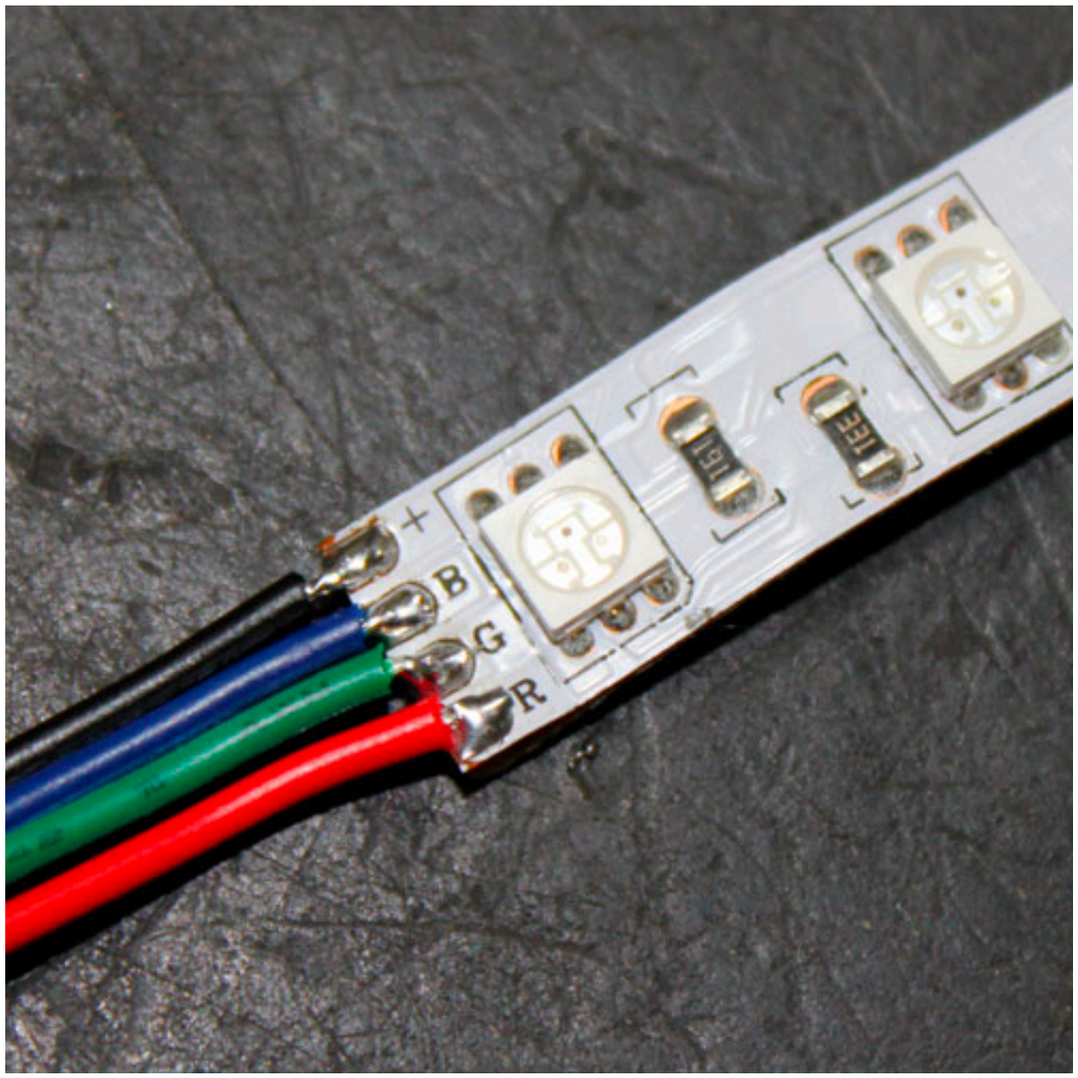
A strip of LEDs

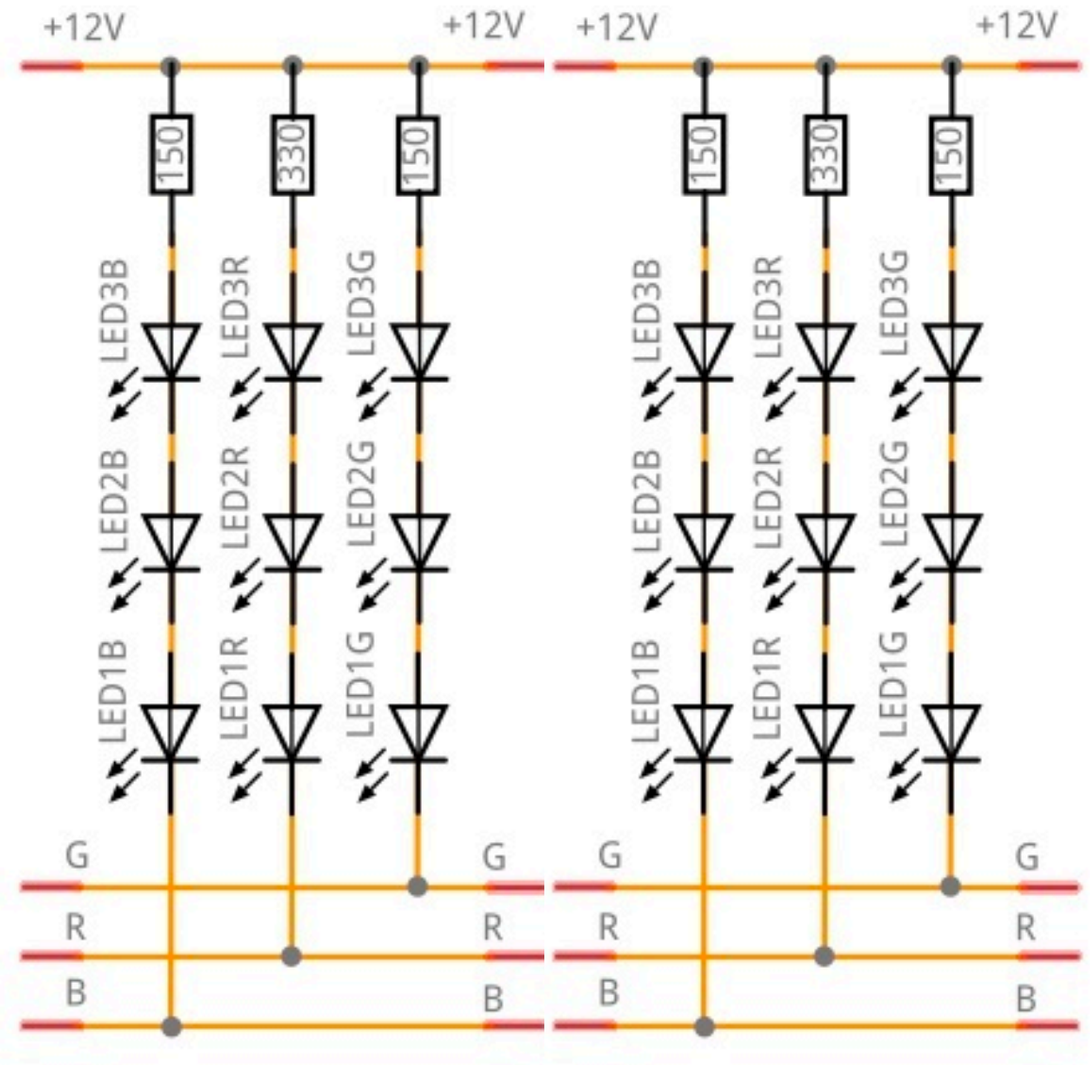




LED strips

A strip of LEDs





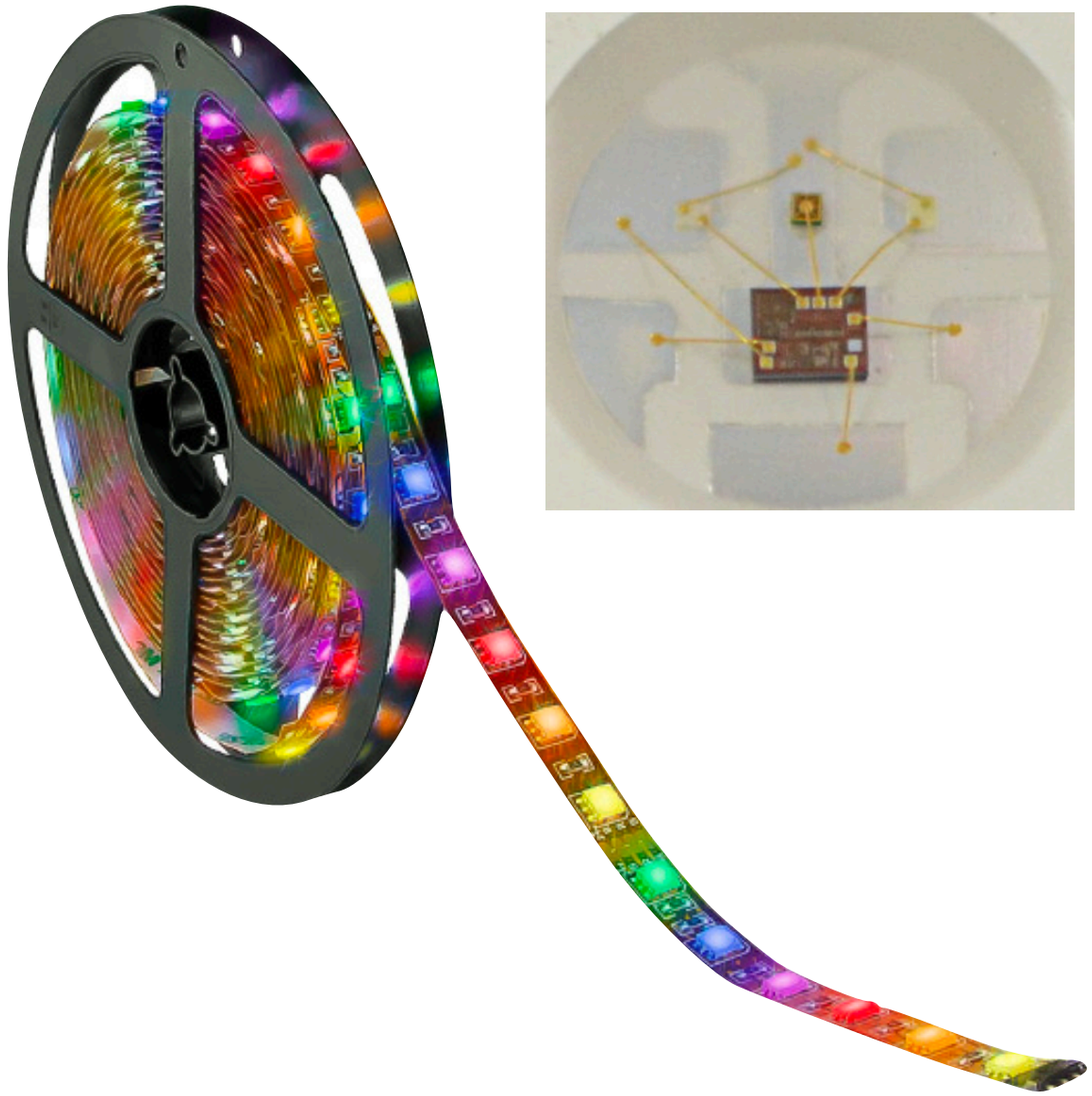
Individually Addressable LED strips!

Objectively pretty neat!

Integrated Circuit

ws2812b RGB LED

How does this work?



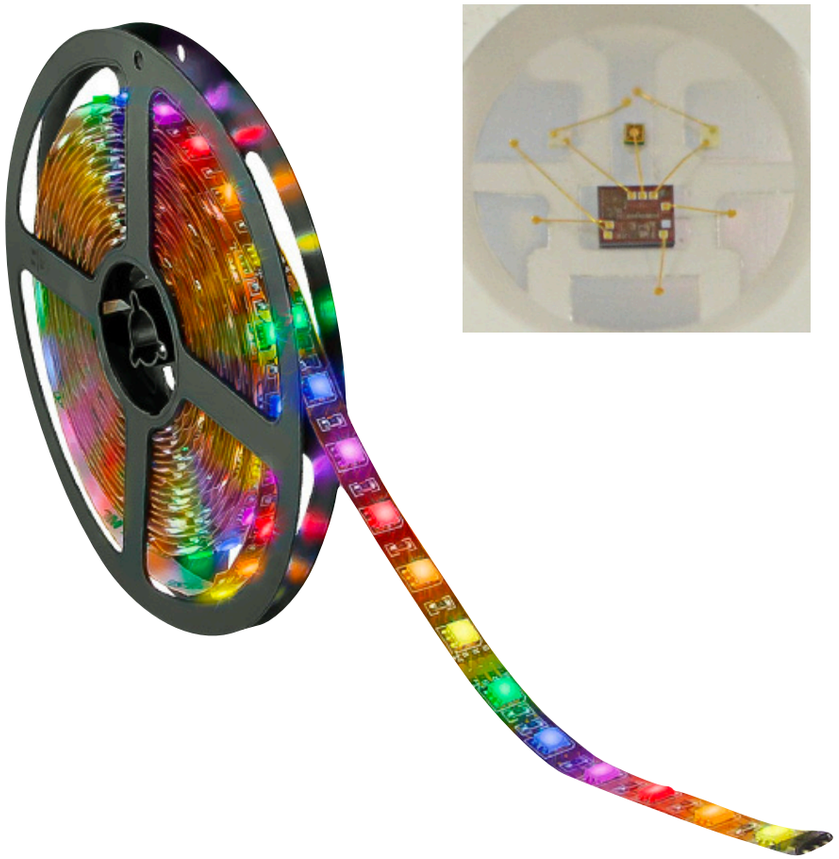
Individually Addressable LED strips!

Objectively pretty neat!

Integrated Circuit

ws2812b RGB LED

How does this work?

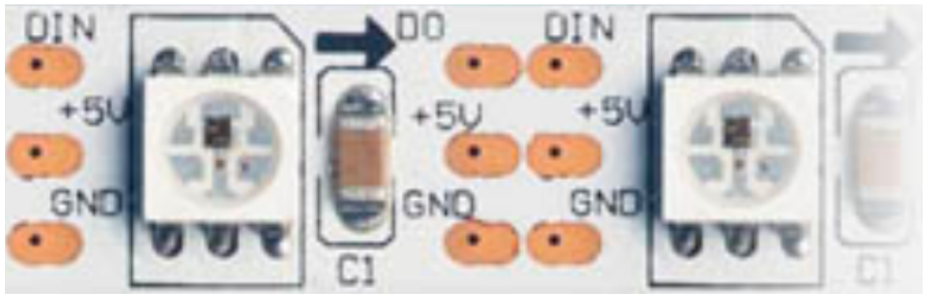


Data Out

Data In

5V

Gnd



Directional!

Individually Addressable LED strips!

Objectively pretty neat!

Integrated Circuit

ws2812b RGB LED

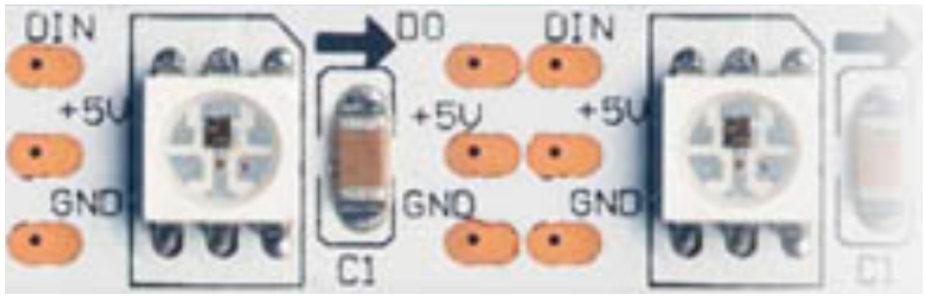
How does this work?

Data Out

Data In

5V

Gnd



Directional!

PROTOCOL

LED1:

Red at 100%

Green at 0%

Blue at 50%

LED 2:

Red at 0%

Green at 100%

Blue at 0%

Individually Addressable LED strips!

Objectively pretty neat!

Integrated Circuit

ws2812b RGB LED

How does this work?

Data Out

Data In

5V

Gnd

Directional!

Individually Addressable LED strips!

Run on 5v and can be controlled with an Arduino!

Watch out for current!
Each color ~ 20mA, total 60mA on ‘white’.

Arduino max current
~200mA - 1000mA

https://cdn-shop.adafruit.com/datasheets/WS2812B.pdf

```
#include <FastLED.h>

// How many leds in your strip?
#define NUM_LEDS 10
#define DATA_PIN 2

// Define the array of leds
CRGB leds[NUM_LEDS];

void setup() {
  FastLED.addLeds<WS2812B,DATA_PIN,RGB>(leds,NUM_LEDS);
  FastLED.setBrightness(60);
}
```

```
void loop() {
  for(int i = 0; i < NUM_LEDS; i++) {
    leds[i] = CRGB(255, 255, 255);
    // Show the leds
    FastLED.show();
    delay(100);
  }

  delay(2000);

  FastLED.clear();
  FastLED.show();
}
```

See you next week!

Try to complete the non-bonus parts of the handout

Office Hours! On learn.illuminations.mit.edu. Come ask questions, hang out, etc.

We'll be on Slack if you have any questions or need any help!