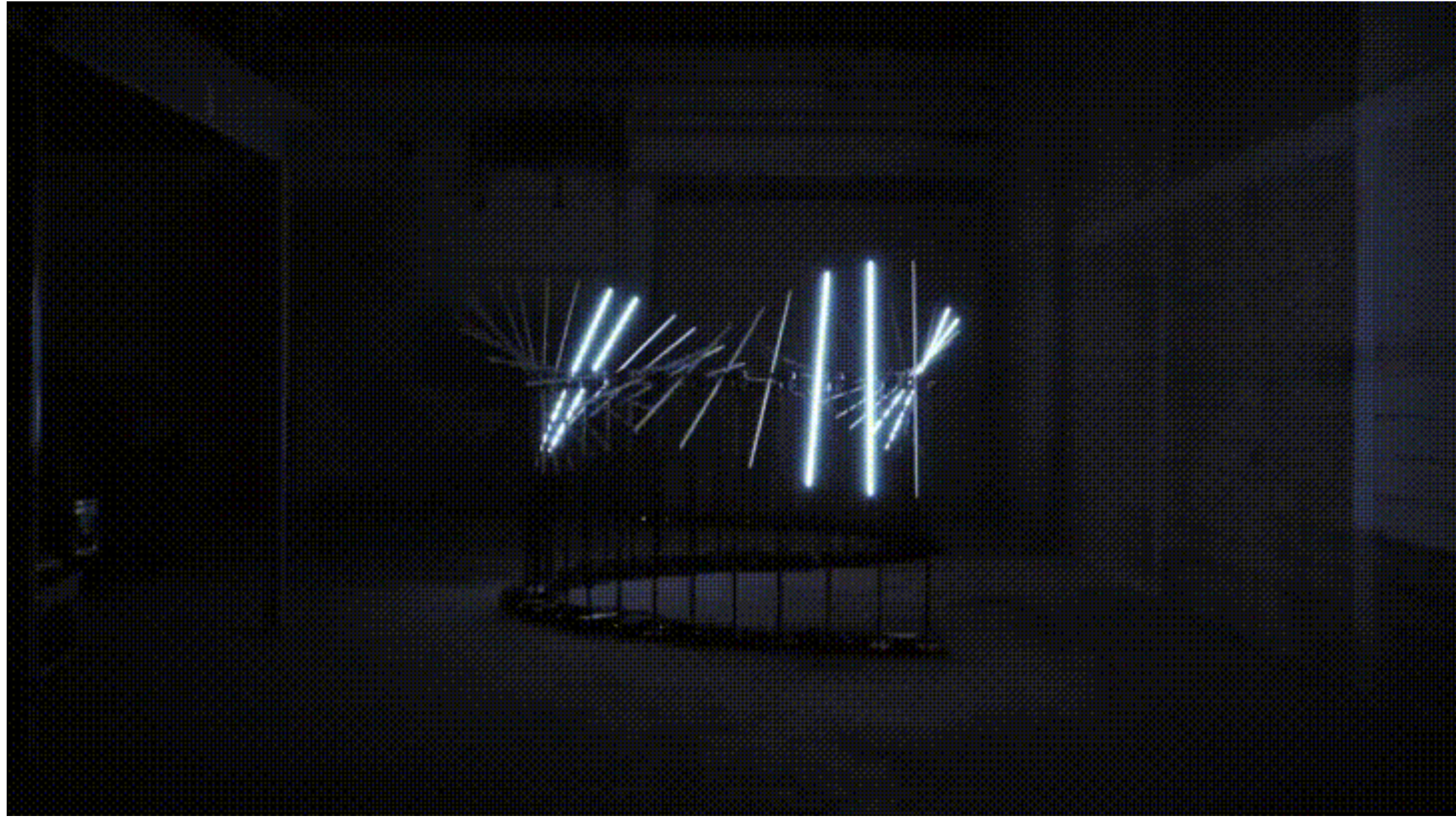


INTERACTION DESIGN

# NEOPIXELS

Physical Computing HS21



Collectif Scale "Flux"



Workshop Madmapper Geneva 2014



Tales of the Jedi Comics Lightsaber

# LED

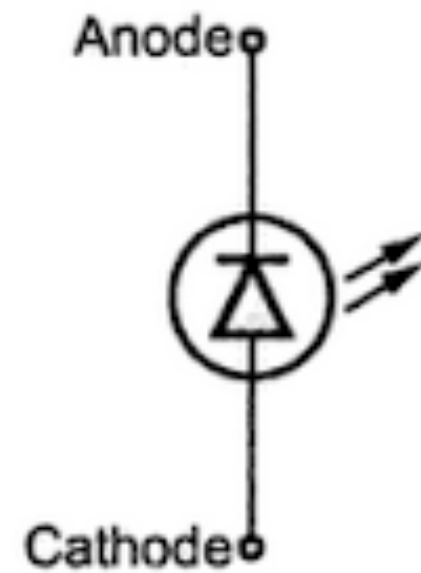


Fig. 1 Symbol of LED

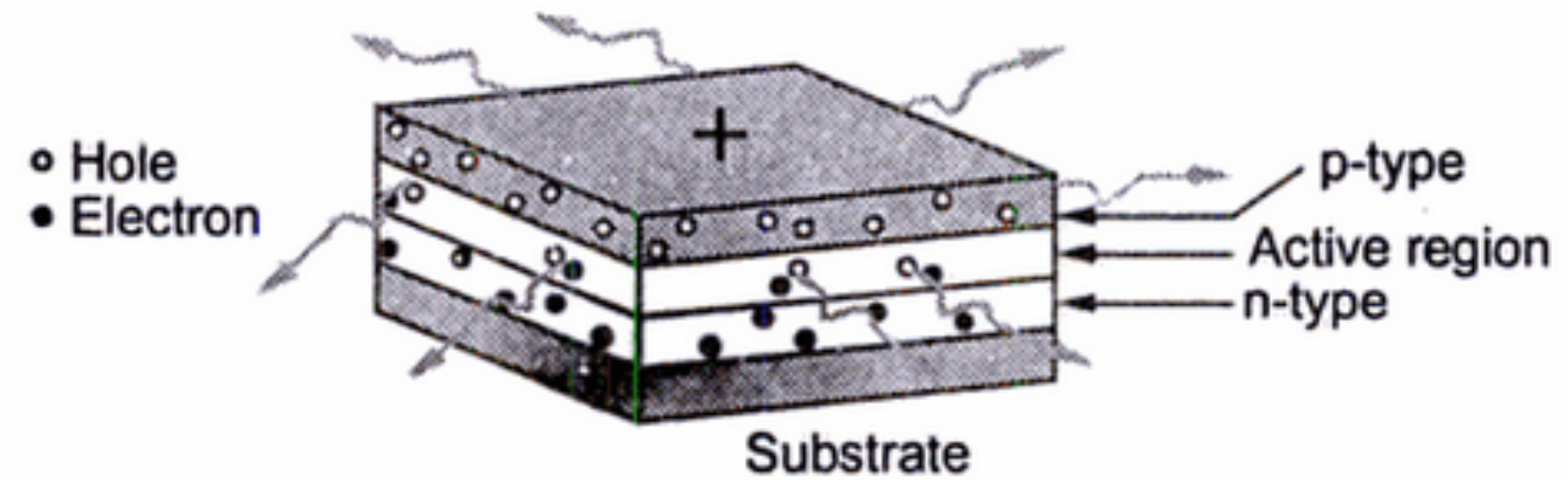


Fig. 3 LED construction

**LED (Light Emitting Diode)** emits light as a result of electric luminescence. The light is not produced by heating a metal filament, instead the diode, emits light when flowing through two specially coated silicon semiconductors.

# LED

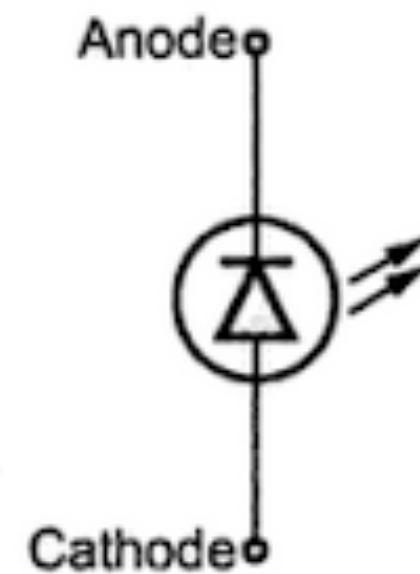


Fig. 1 Symbol of LED

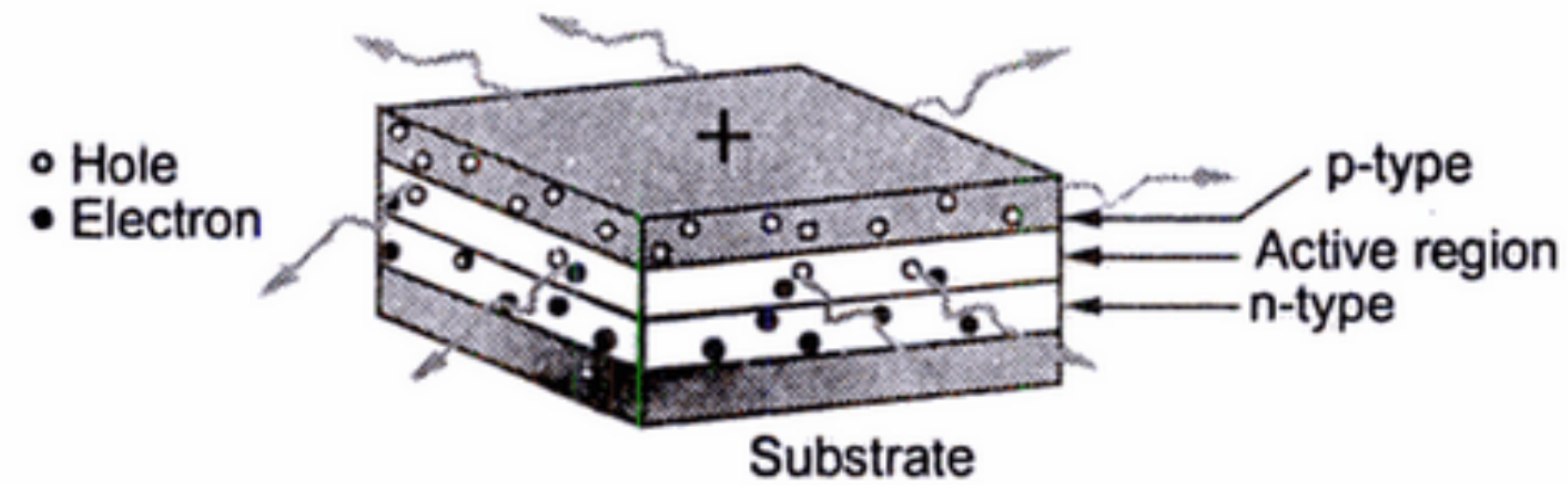
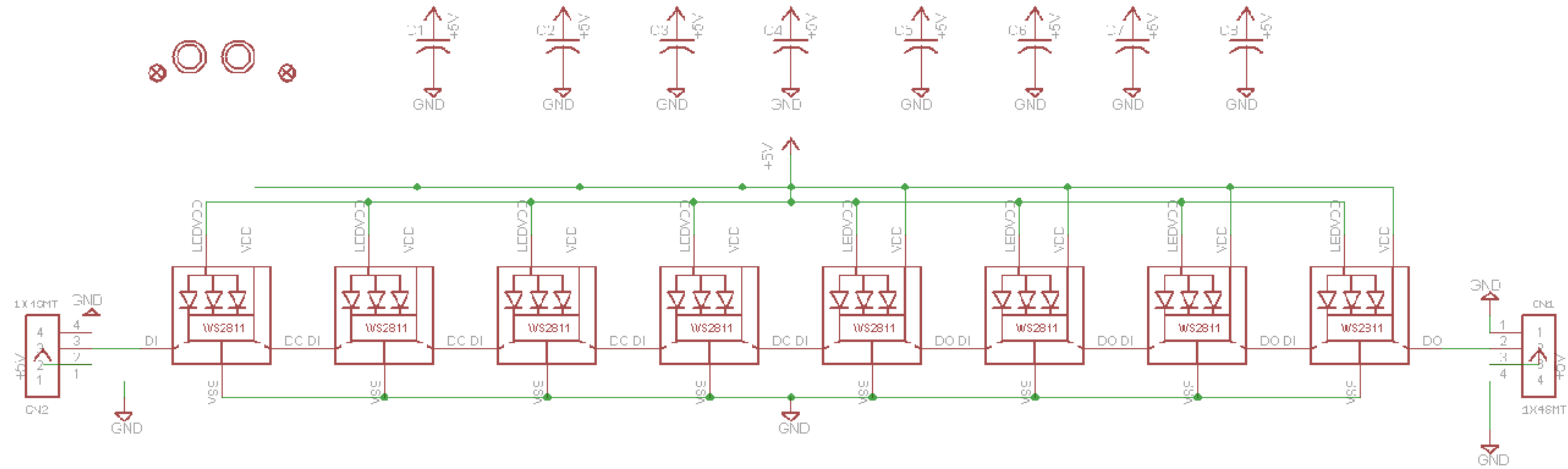


Fig. 3 LED construction

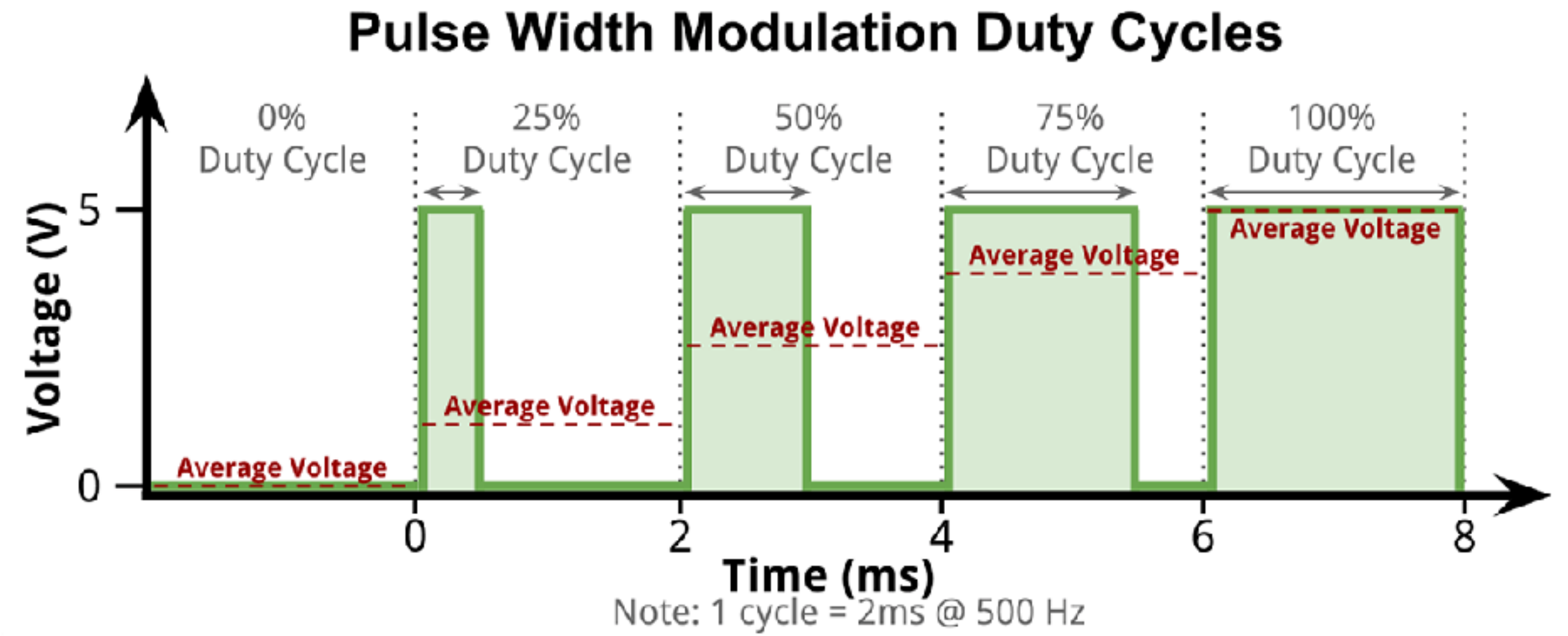
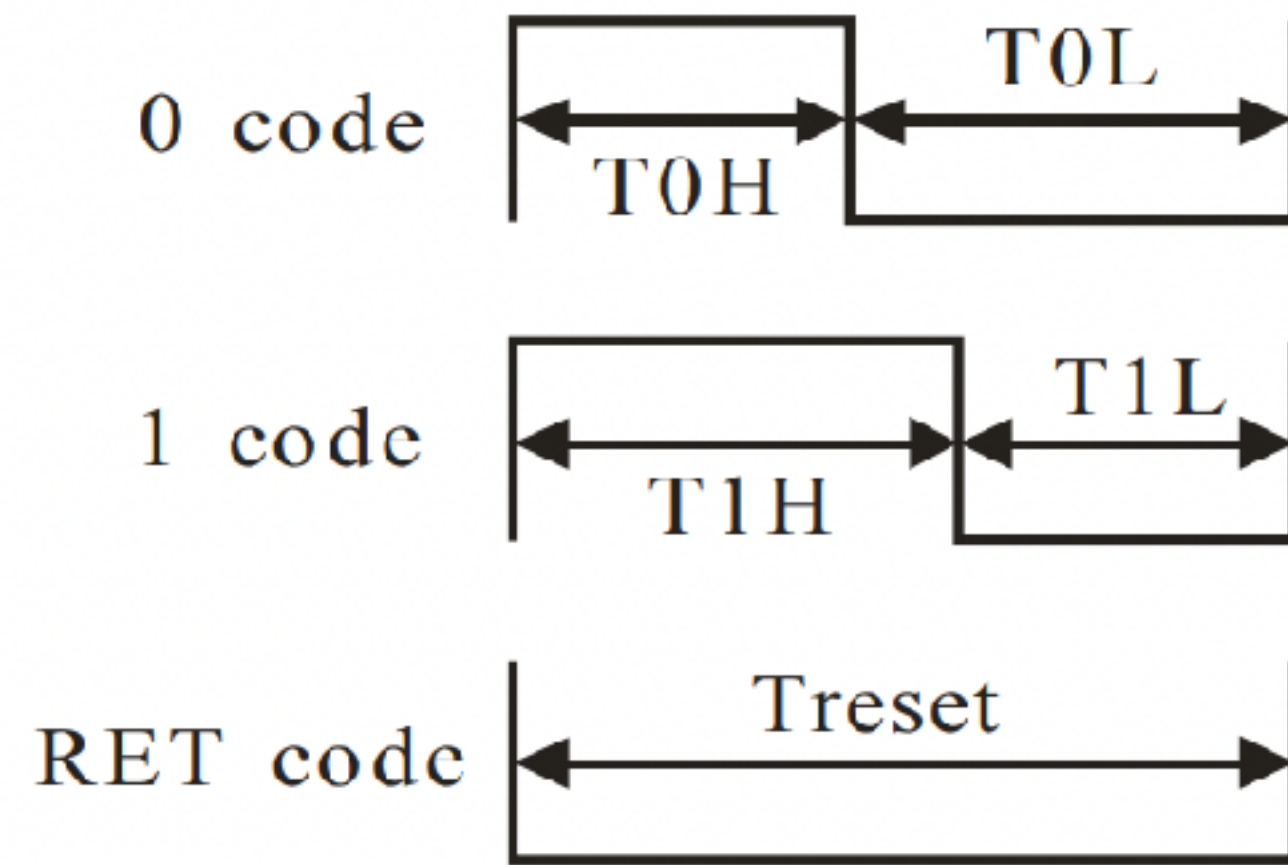
**LED (Light Emitting Diode)** emits light as a result of electric luminescence. The light is not produced by heating a metal filament, instead the diode, emits light when flowing through two specially coated silicon semiconductors. The first infrared LED was developed in 1961 by Robert Biard and Gary Pittman, but it wasn't until 1970's that the advances in LED technology led to the development of green, orange, and yellow LEDs.

# NEOPIXELS



- Individually addressable RGB color pixels and strips based on the **WS2811**, **WS2812**, **WS2815**, **SK9822** and **SK6812** integrated light source packages
- The term has become the standard name for addressable single wire controlled LEDs regardless of manufacturer
- NeoPixels cannot be lit with a simple circuit, they require a microcontroller to send the control signals

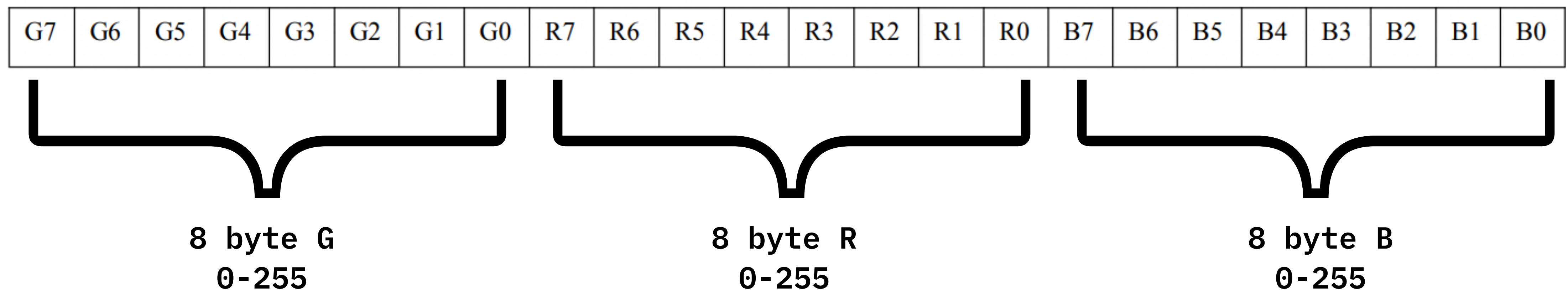
# WS2812



WS2812 light source uses a single wire control protocol(SWP), not separate data and clock lines. "1" and "0" bits are indicated by varying the duty cycle of a fixed-frequency square wave.

# COLOR DATA

When Neopixels receive data from microcontroller, each pixel collects 24bit data that is composed of **3 bytes** of color, with the most-significant bit first.



$$256^3 = 16777216 \text{ colors per pixel}$$

# MEMORY IN AVR-BASE ARDUINO

```
Flash 32k bytes (of which .5k is used for the bootloader)
SRAM 2k bytes
EEPROM 1k byte
```

Flash memory (program space), is where the Arduino sketch is stored.

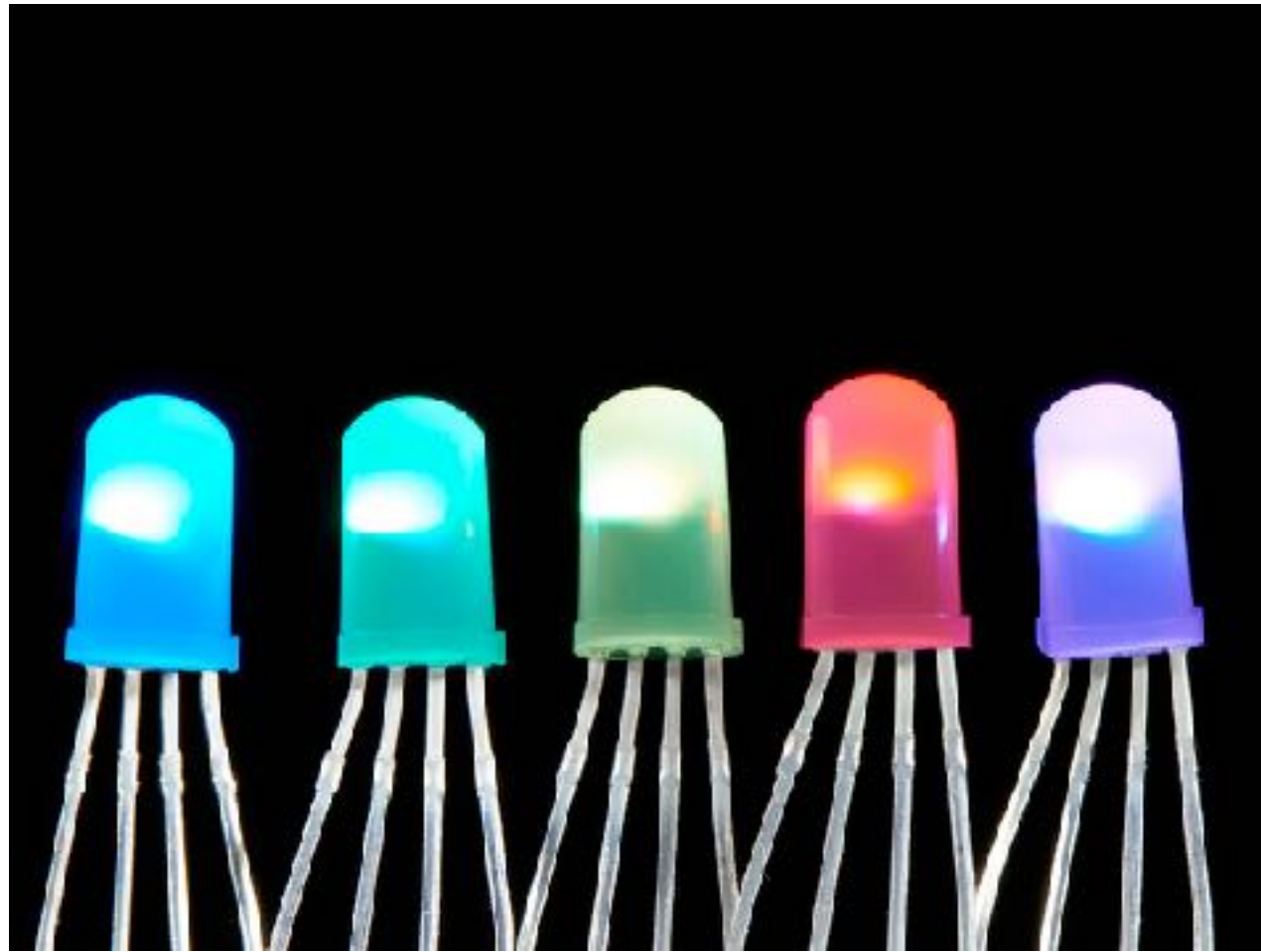
SRAM (static random access memory) is where the sketch creates and manipulates variables when it runs.

EEPROM is memory space that programmers can use to store long-term information.

**If each NeoPixel requires about 3 bytes of RAM you will need to add another Arduino after  $\approx 600$  LEDs ( $\approx 10m$ )**



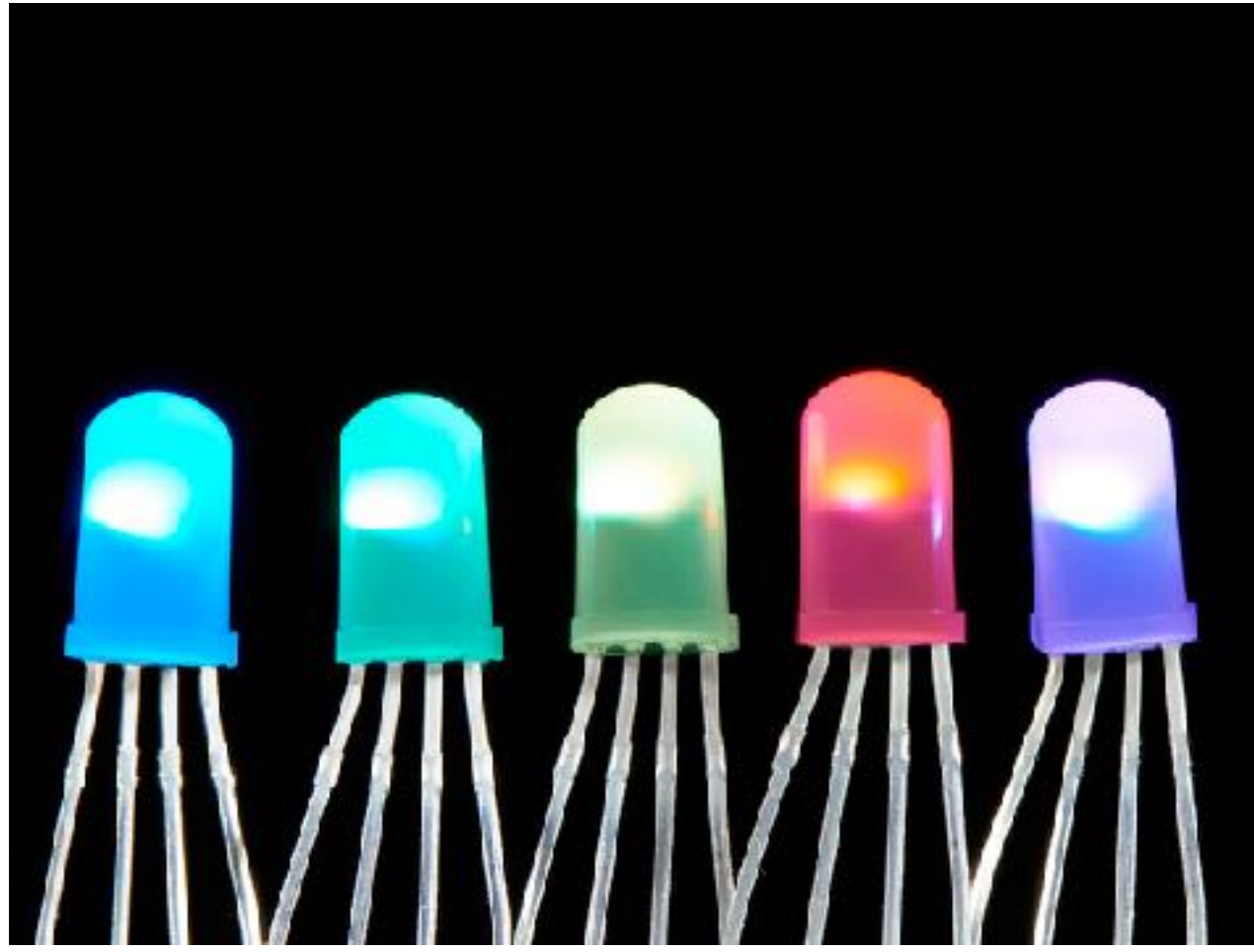
# TYPES



Through-Hole

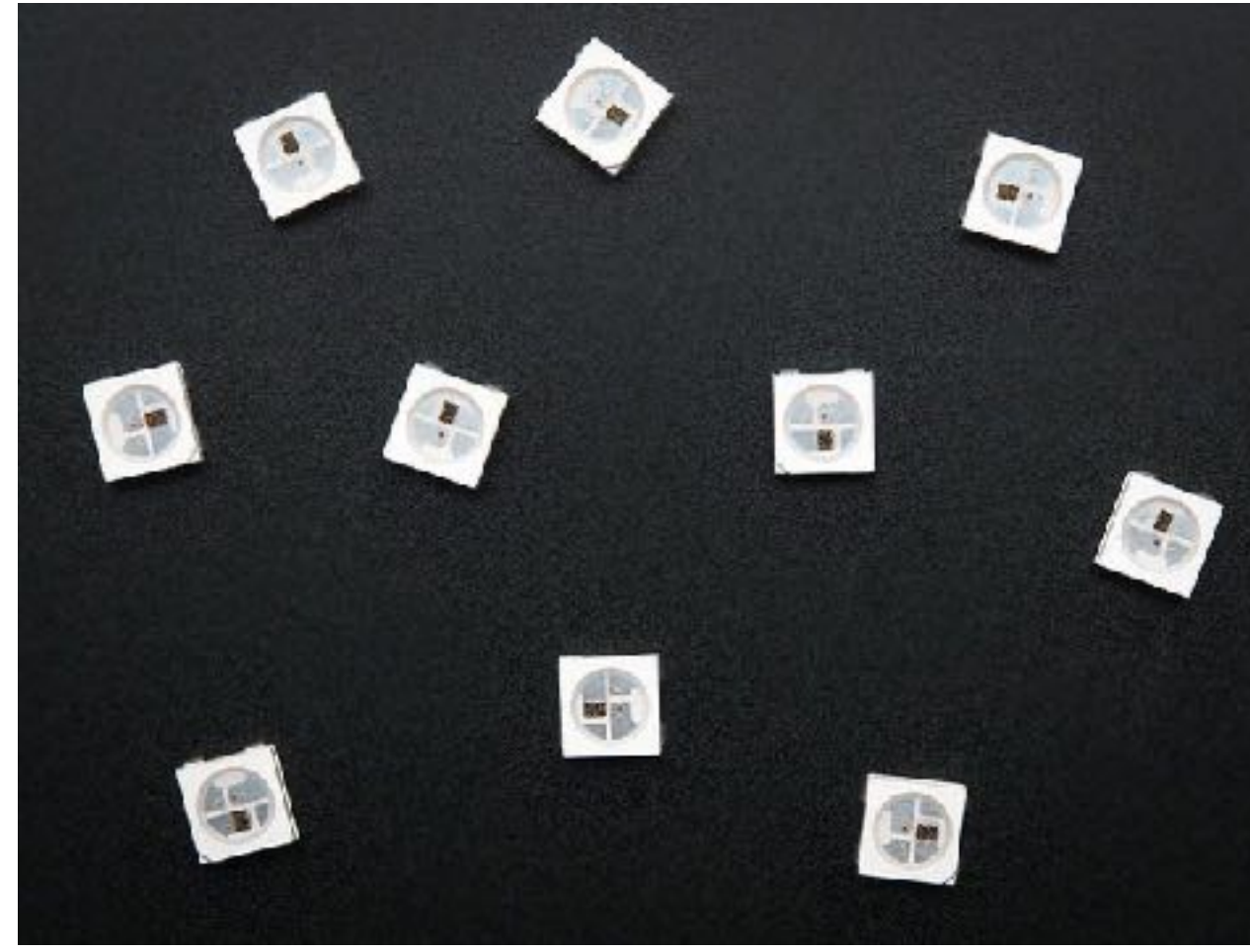
NEO\_RGB

# TYPES



Through-Hole

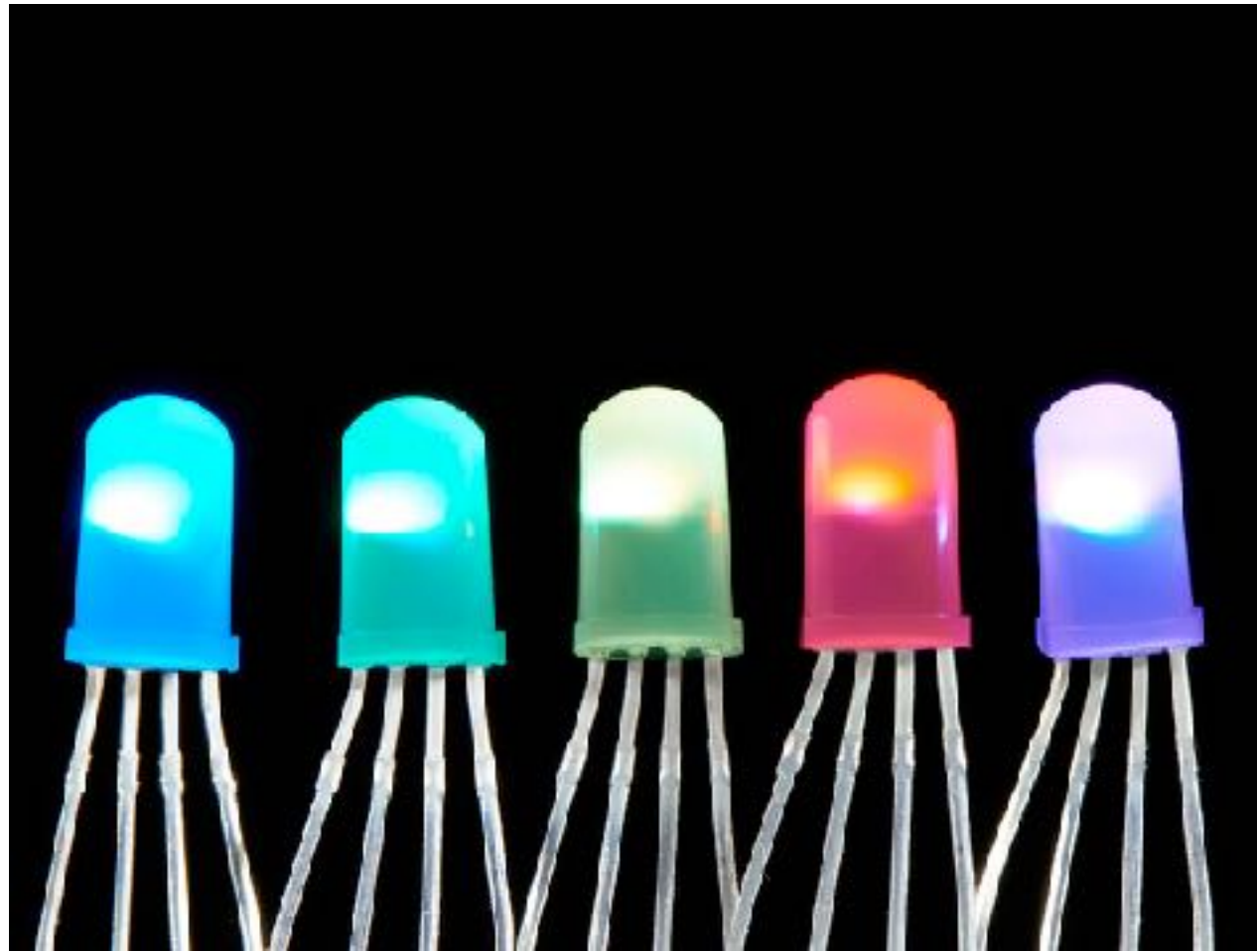
NEO\_RGB



Surface Mount

NEO\_GRB

# TYPES



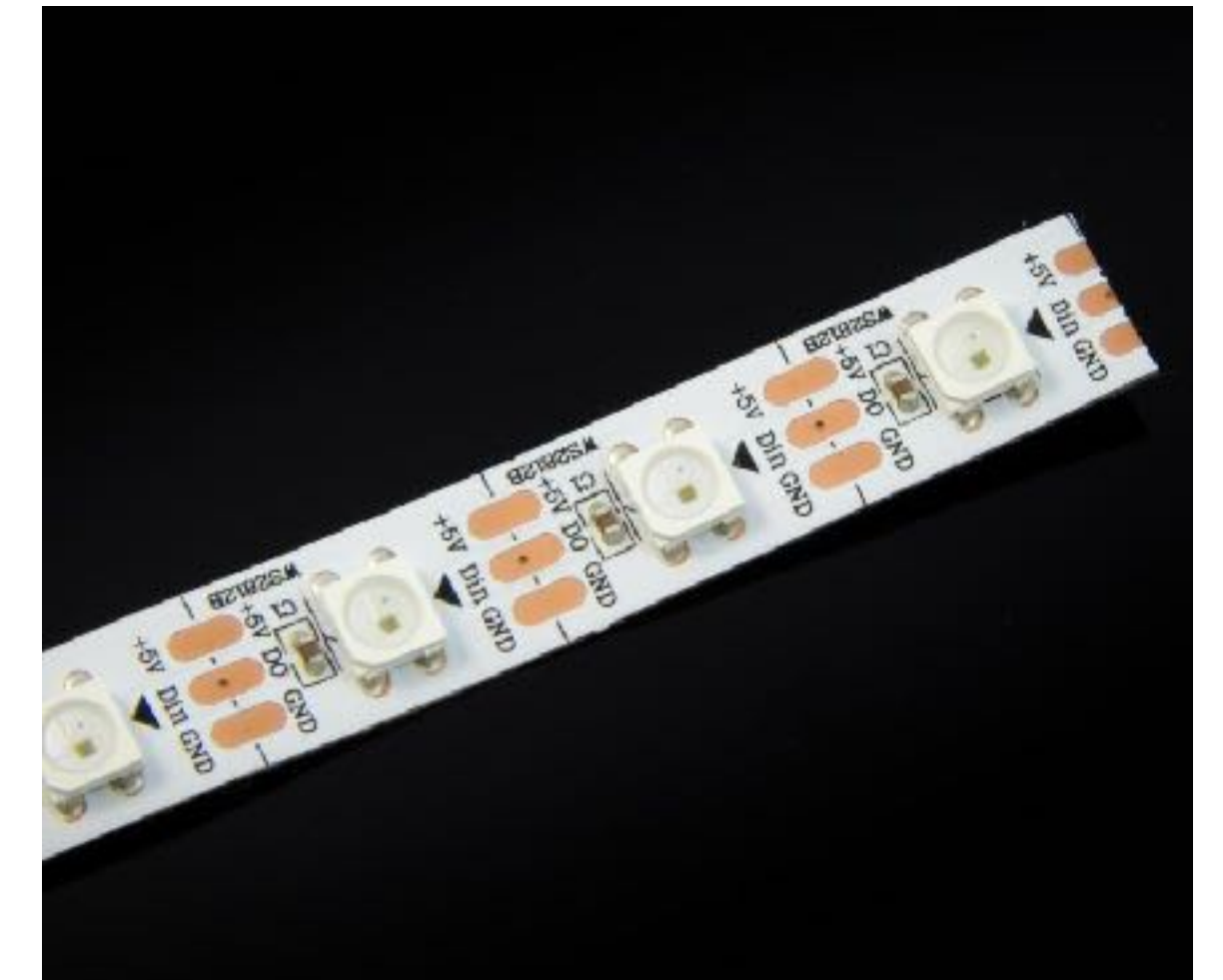
Through-Hole

NEO\_RGB



Surface Mount

NEO\_GRB



Strip

NEO\_GRB

# TYPES

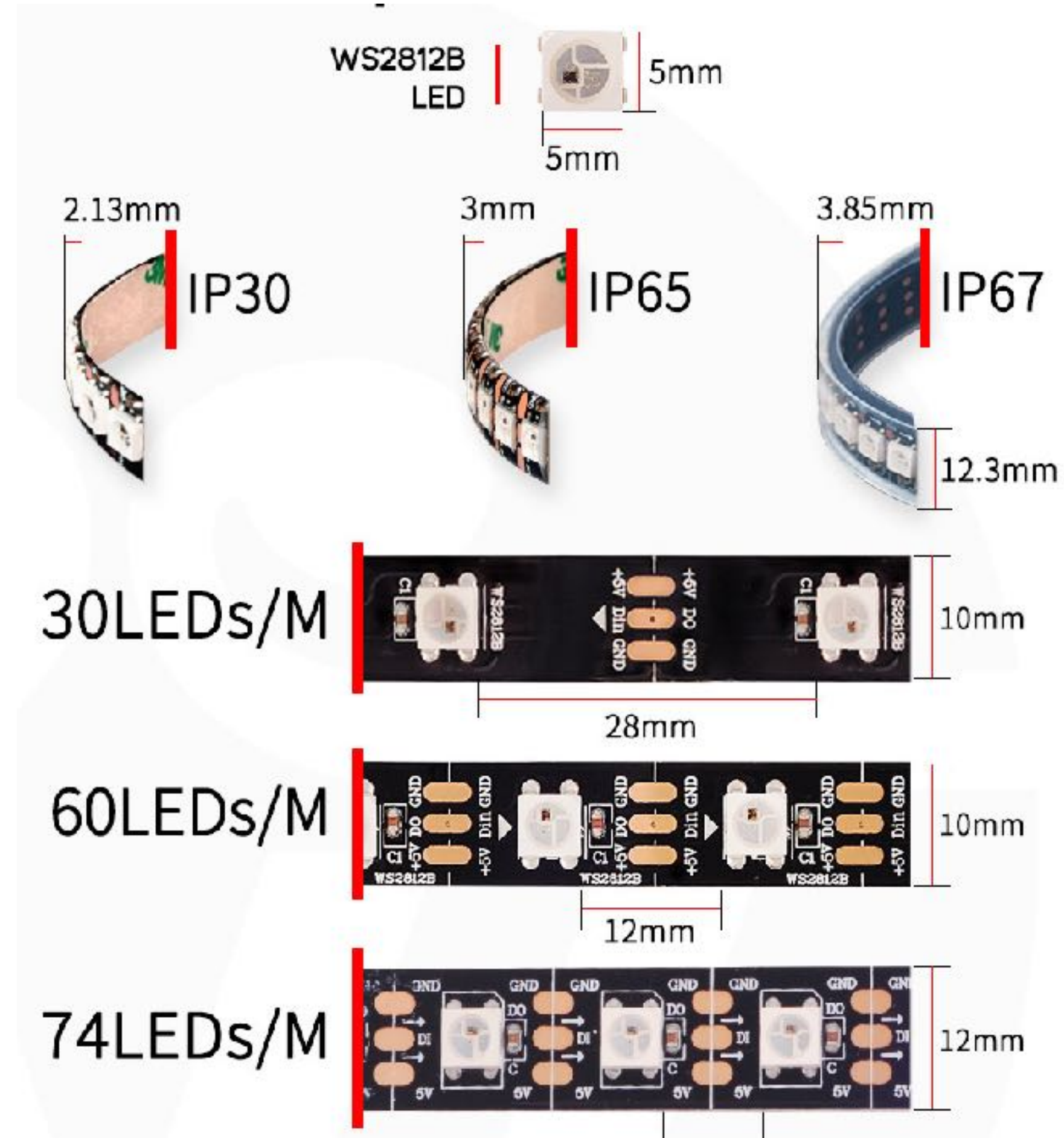
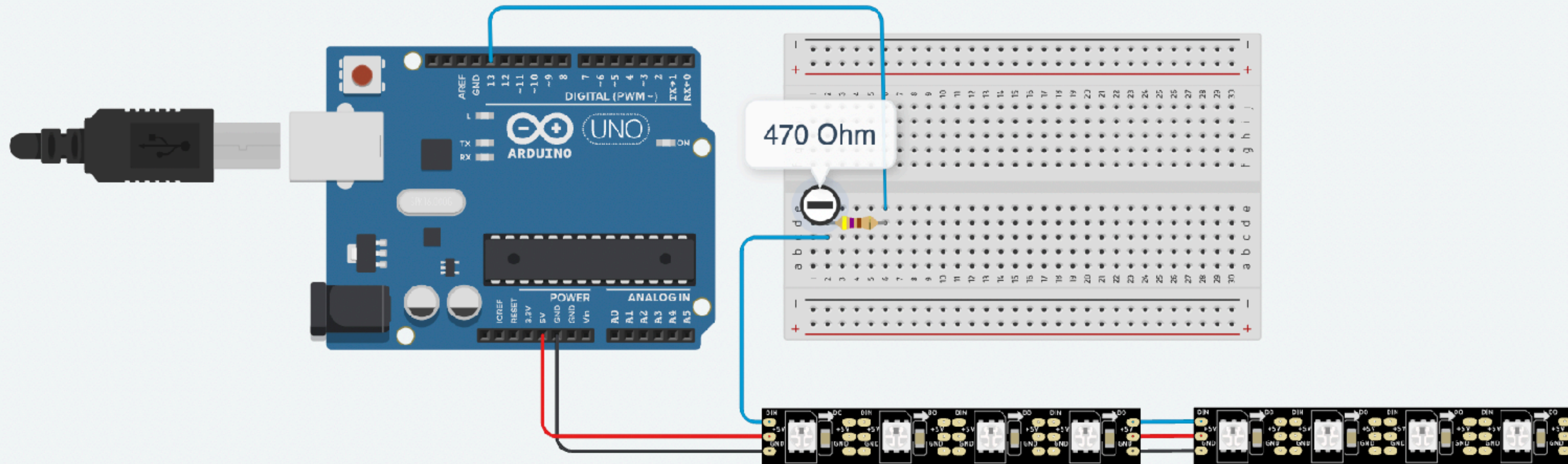
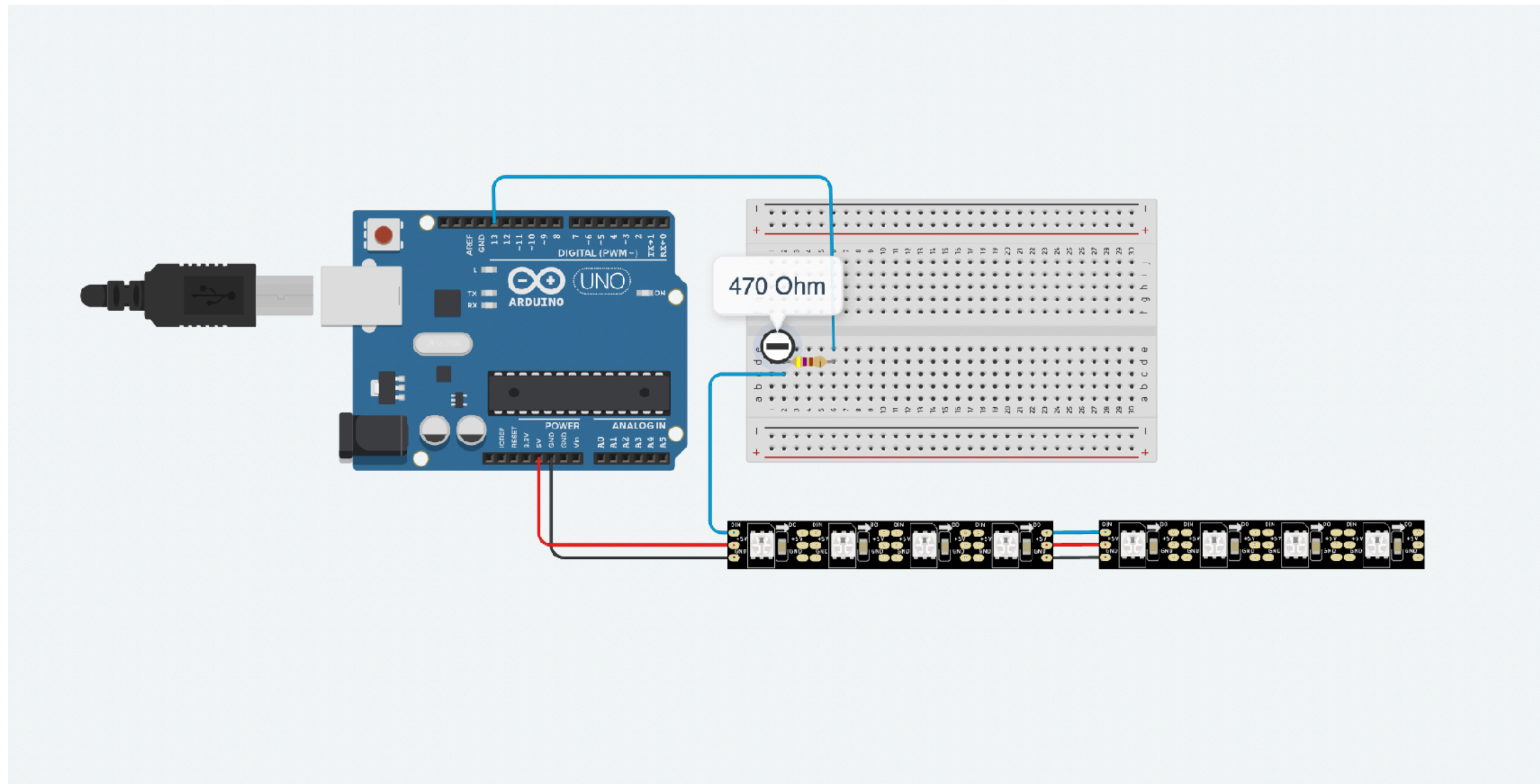


Image source

# WIRING (15 Min)



# WIRING



- Do not connect to a live circuit
- Use a resistor between 300 and 500 $\Omega$ (at 5V)
- Add a capacitor before connecting to stronger power source
- Minimize the distance between the first pixel and Arduino for a better signal
- When using external power source apply power to the pixels before applying power to the microcontroller.
- There's principally no inherent limit in the maximum length of a NeoPixel chain.

# POWER REQUIREMENTS

Generally, we can use a simple formula to calculate power consumption and current, which is also good to check, when you are choosing a power supply.  
Each pixels

**I [A] = 0.036 [A] (current of one GRB LED) \* number of LEDs \* meters**

**I [A] = 0.048 [A] (current of one RGB LED) \* number of LEDs \* meters**

LED consumption can be easily calculated with this formula:

**P [W] = U [V] (5 in case we have 5V DC LED strips) \* I [A]**

## EXERCISE 1 (10min)

Calculate the power consumption and current that will be needed to power up a 10 m WS2812 strip with 30 LEDs per 1 m.



# POWER INJECTION

For longer strip, if pixels are not bright enough, use **power injection** method each 4A to output 100% brightness at every pixel.

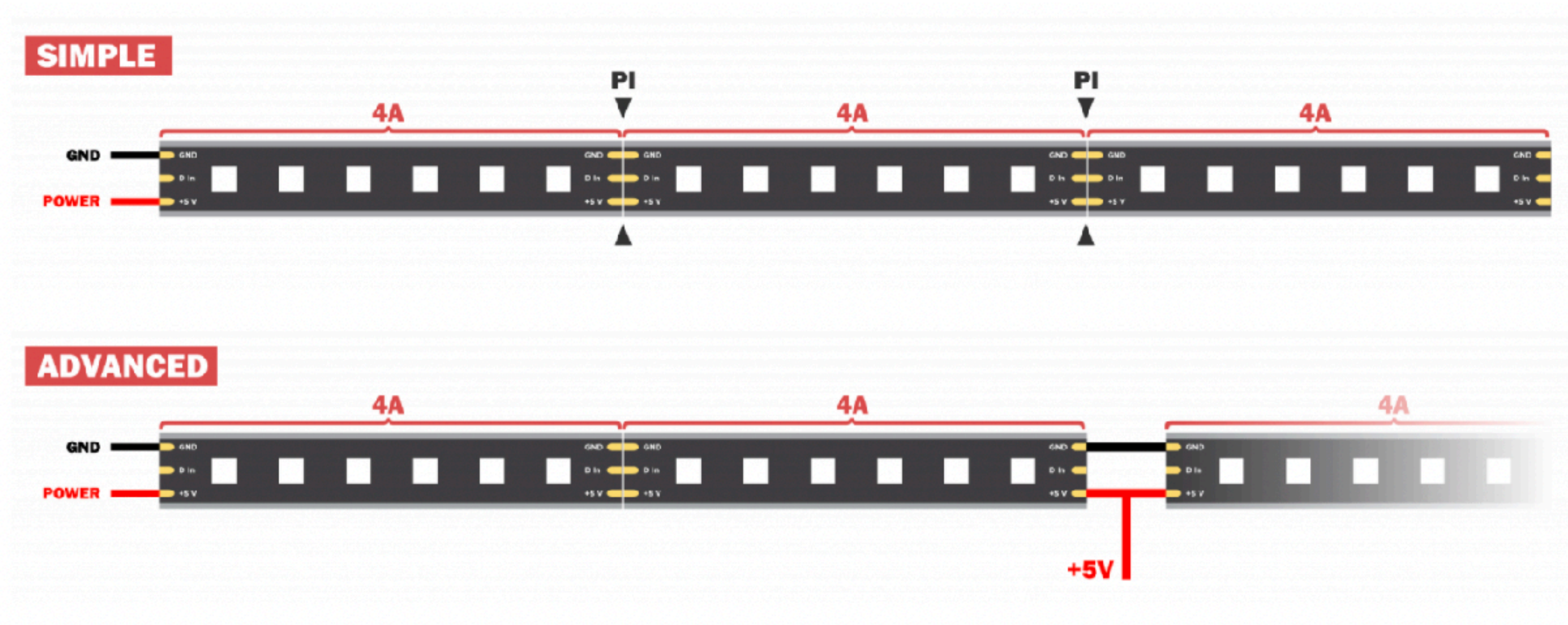


Image source

# ARDUINO LIBRARY

- Install **Adafruit\_NeoPixel** library via library manager and include it in your sketch using **#include<Adafruit\_NeoPixel.h>**

- Define an object to reference the NeoPixels:

```
Adafruit_NeoPixel strip = Adafruit_NeoPixel(8, 13, NEO_GRB + NEO_KHZ800);
```

- Prime your NeoPixels :

**LED AMOUNT**

**PIN**

**COLOR FLAG**  
**NEO\_GRB OR NEO\_RGB(W)**

**BITSTREAM FLAG**  
**NEO\_KHZ800 OR NEO\_KHZ400**

```
void setup() {  
  strip.begin();  
  strip.show();  
}
```

- Assign colors:

```
void loop() {  
  strip.setPixelColor(2, 255, 0, 0);  
  strip.show();  
  delay(10);  
}
```

**LED NUMBER**

**RED**

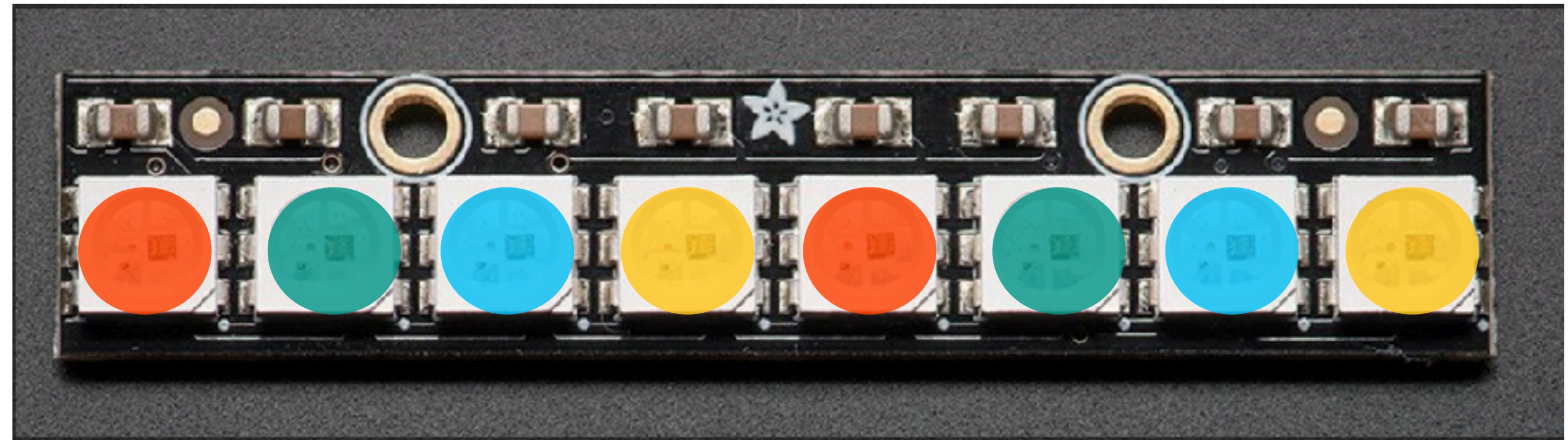
**GREEN**

**BLUE**

## EXERCISE 2 (30min)

Write a code that will make every fourth LED blink in the same color every 200ms, e.x.:

0, 4 RED  
1, 5 GREEN  
2, 6 BLUE  
3, 7 YELLOW



**TIPP: Use modulo function**