Week 5

Expressions --- [Analog output]

Analog Output

Analog output is really a form of digital output being cycled at a very fast, controlled rate.

This is also known as Pulse Width Modulation.

But first...

A Few Things About Analog Output

3 (peculiar) things about analog output:

- Range is 0~255, not 0~1023.
- no pinMode() necessary.
- Only digital pins prefixed with a ~ are usable.

Pulse Width Modulation (PWM)

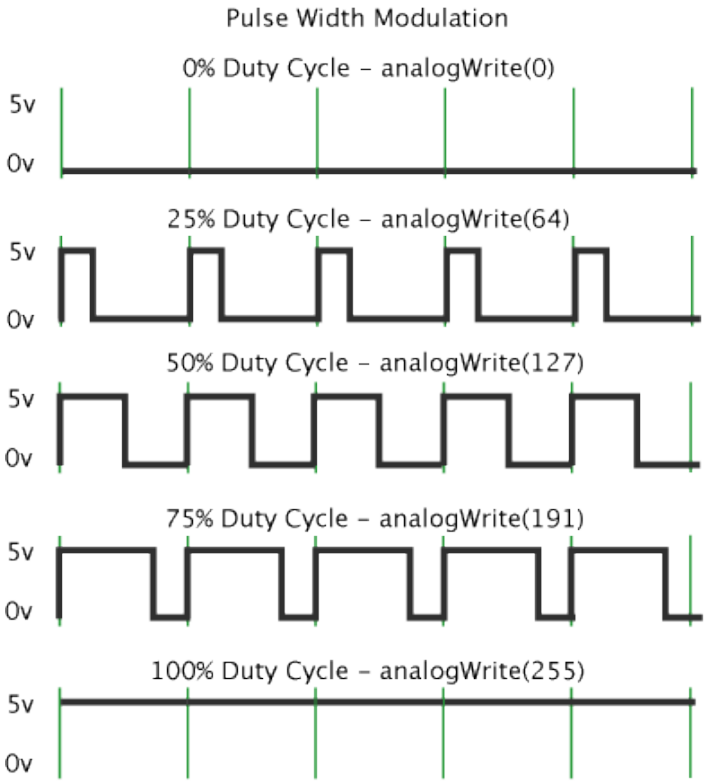
PWM is a way to get an analog signal using digital means.

The HIGH LOW pattern of a digital signal can simulate the full range of voltages between 0-5V by changing the relationship of when the signal is HIGH to when it is LOW.

The duration of time when it is HIGH is called the pulse width.

Duty Cycle

A complete segment of the HIGH LOW duration is referred to as a duty cycle.



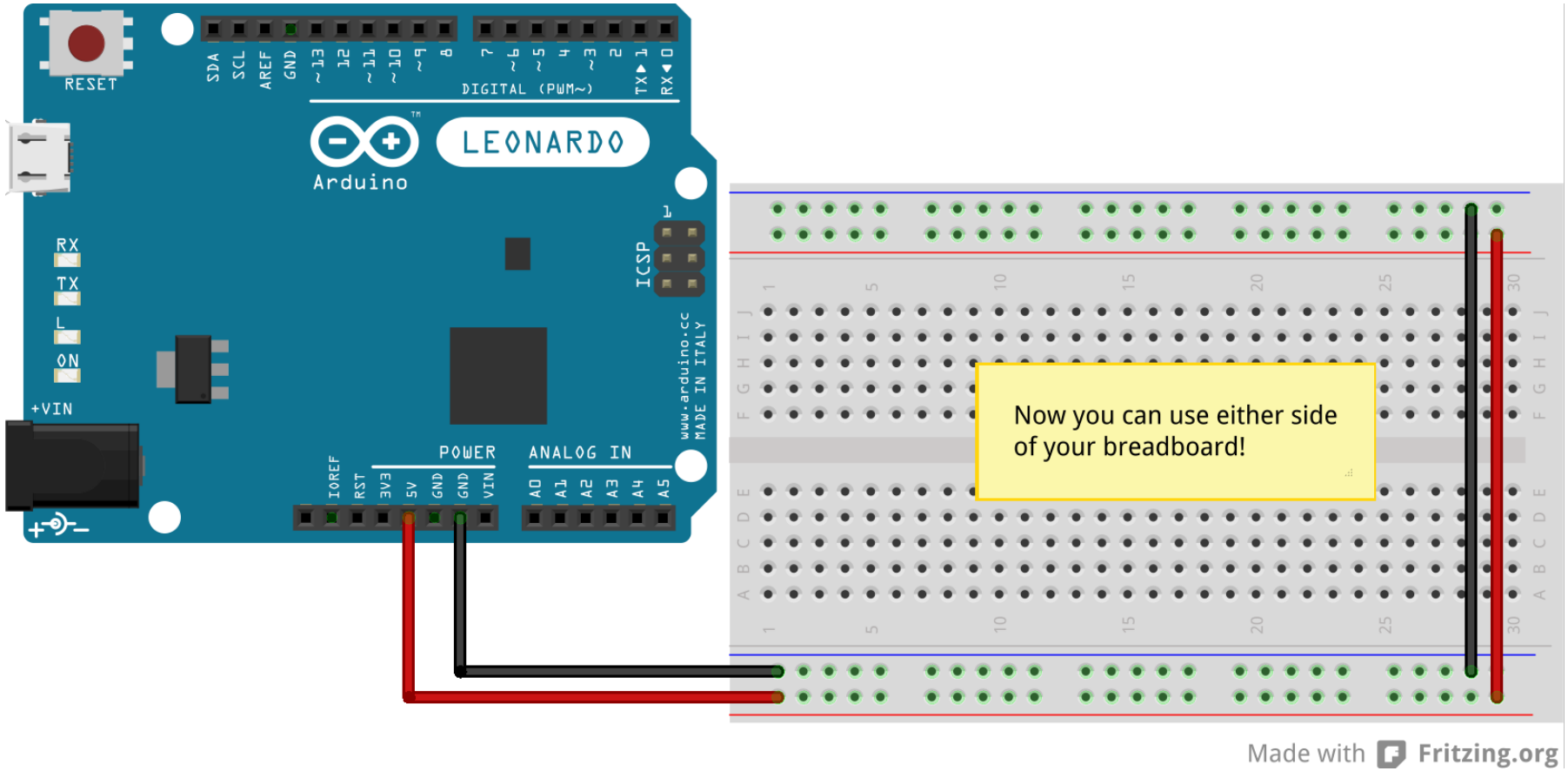
If you imagine the high points in the square waves to the left as being the amount of time when an LED is on, and the low points being when it is off, you can get a good sense of how a duty cycle works.

The more the LED is on, the brighter it will appear.

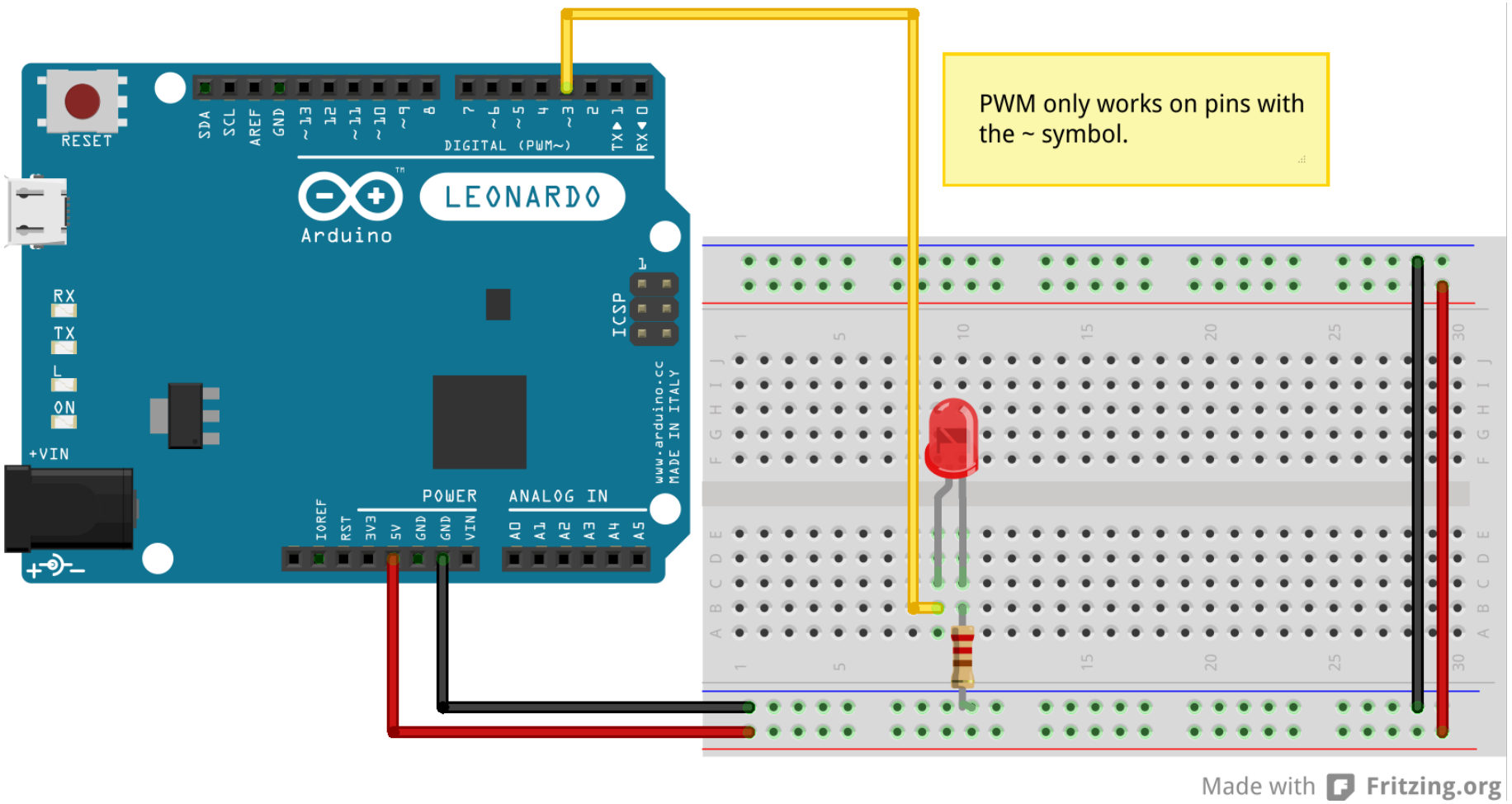
AnalogWrite LED exercise

1) Fading LED (fading.ino)
2) Fading back and forth (fadeBounce.ino)
3) Random numbers (AnalogWrite_random.ino)
4) AnalogIn → AnalogOut (AnalogIn_AnalogOut.ino)
5) FadeBounce with Analog Switch (fadeBounce_analogSwitch.ino)
6) RGB LEDs (fadeBounceRGB.ino)

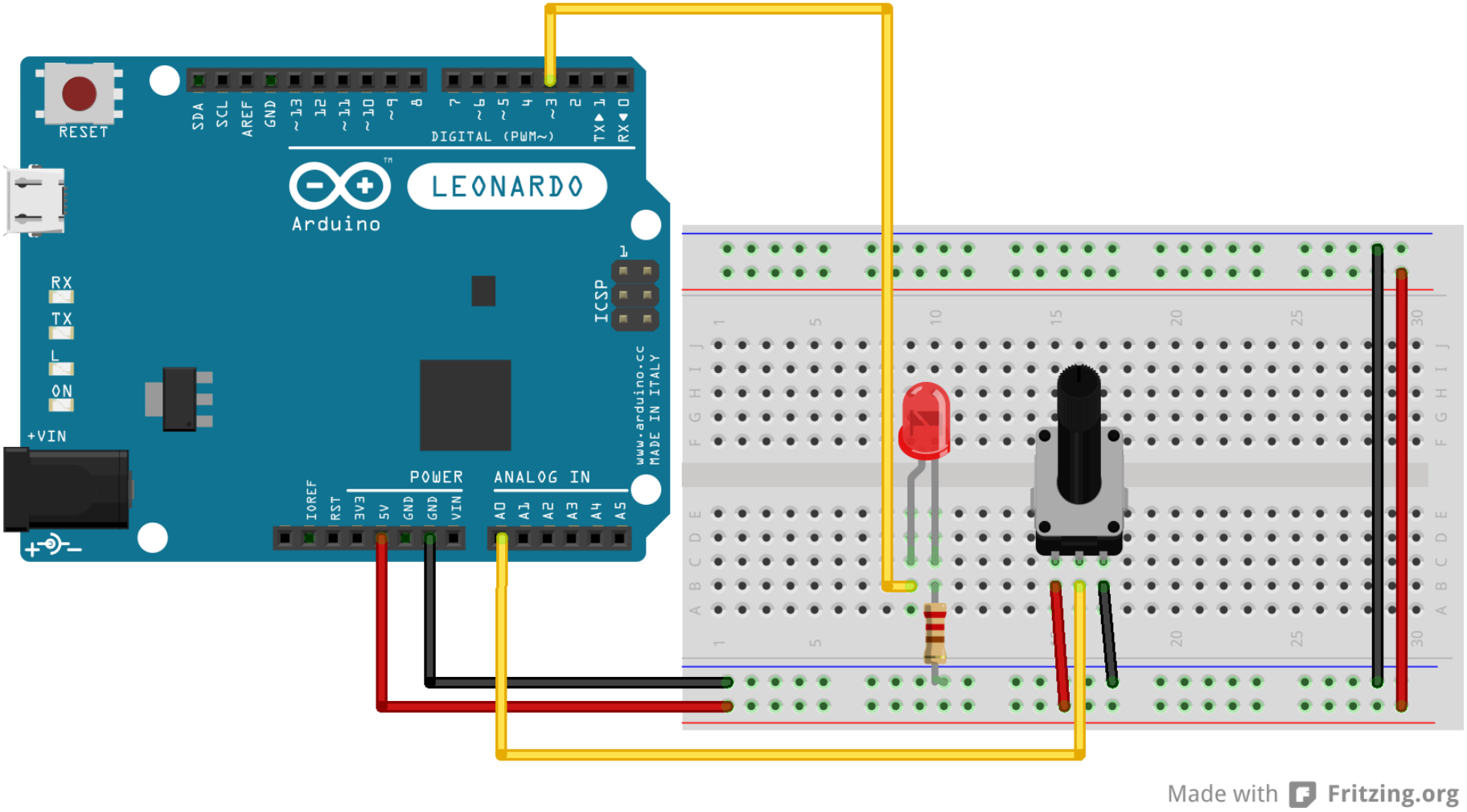
Extending power



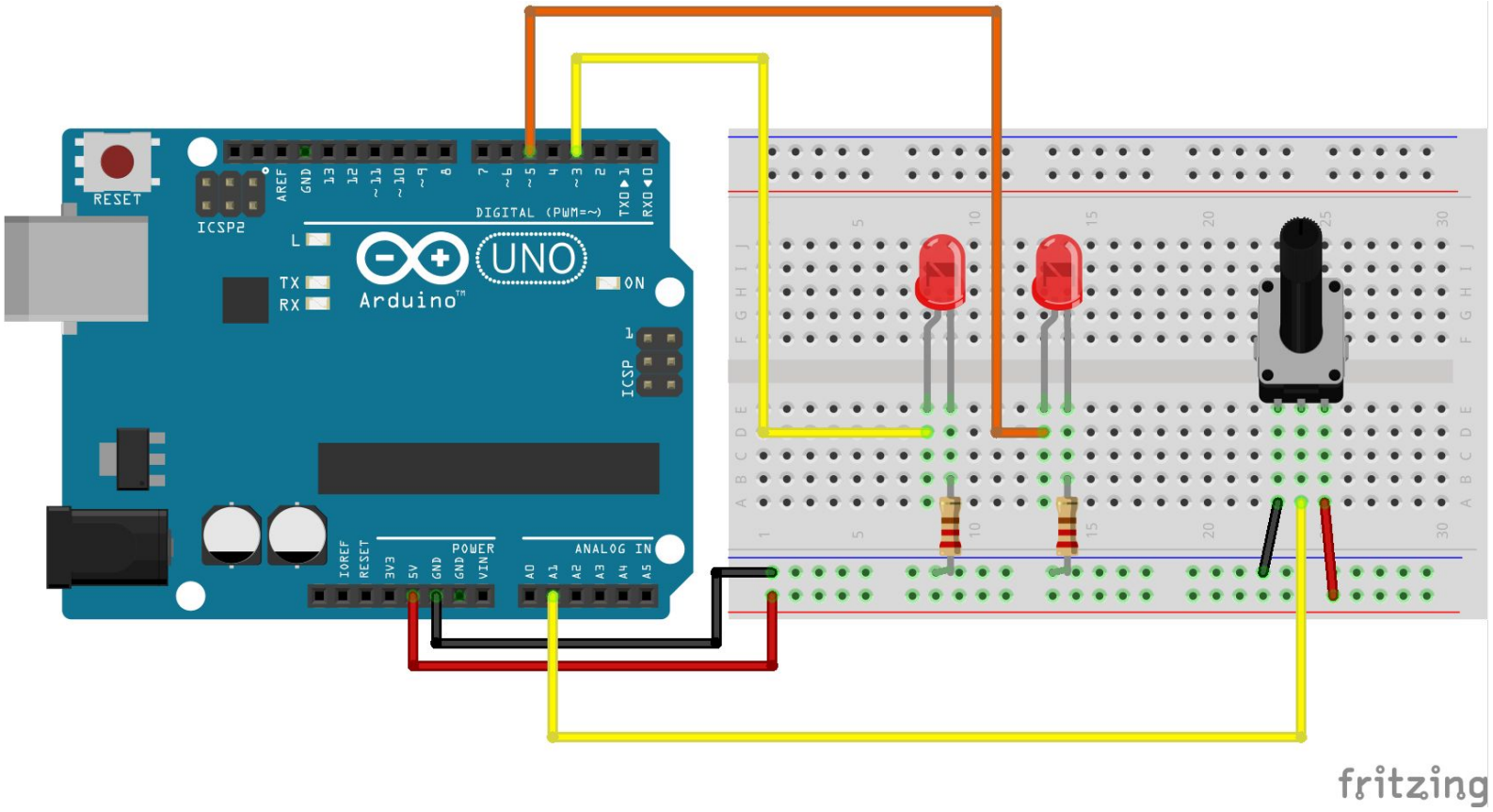
Connecting an LED for PWM



Analog Input + Analog Output



5) FadeBounce with Analog Switch



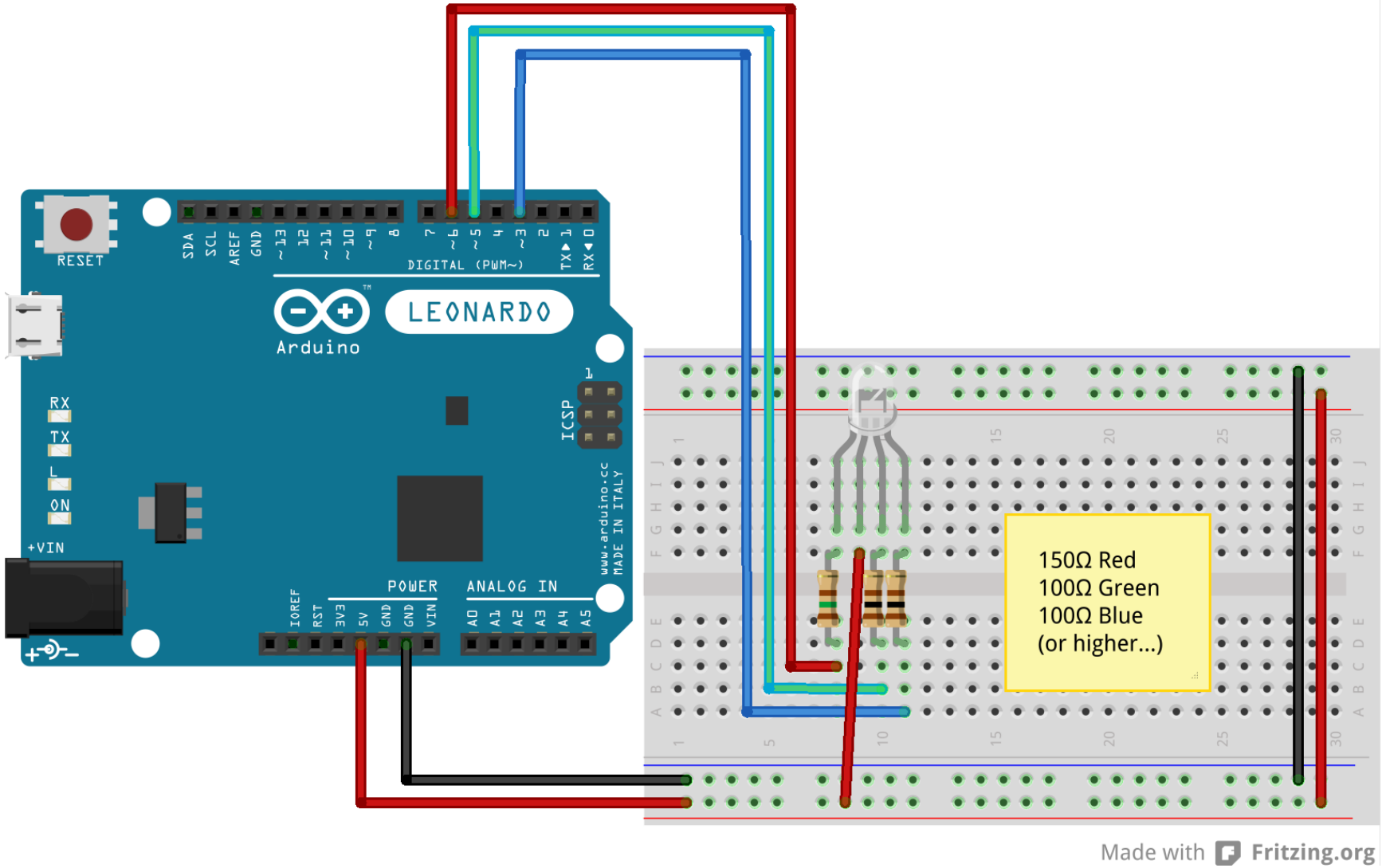
RGB LED

RGB LEDs contain a die for all three colors which can be controlled individually like normal LEDs.

The longest leg on our RGB LED is a “shared anode” which should be connected directly to 5V.

The other legs control the individual colors and must be connected to the your output using appropriate resistors.

Connecting an RGB LED



AnalogWrite exercises 2

1) Solder leads to Piezo element
2) Using Piezo as a speaker (sounds_basic.ino)
3) Using Piezo with a Potentiometer (Pseudo_theremin_map.ino)
4) Using Piezo with a Photoresistor (Pseudo_theremin_map.ino)
5) Using random() function
6) Optional: controlling motor speed with analogWrite
7) Optional: using Piezo as a vibration sensor

Soldering leads onto Piezo element

Soldering Piezo element Steps

1. Scratch the surface of a positive side of Piezo to expose a shiny part that you can solder onto
   ***Be careful not to scratch off the crystal itself!**

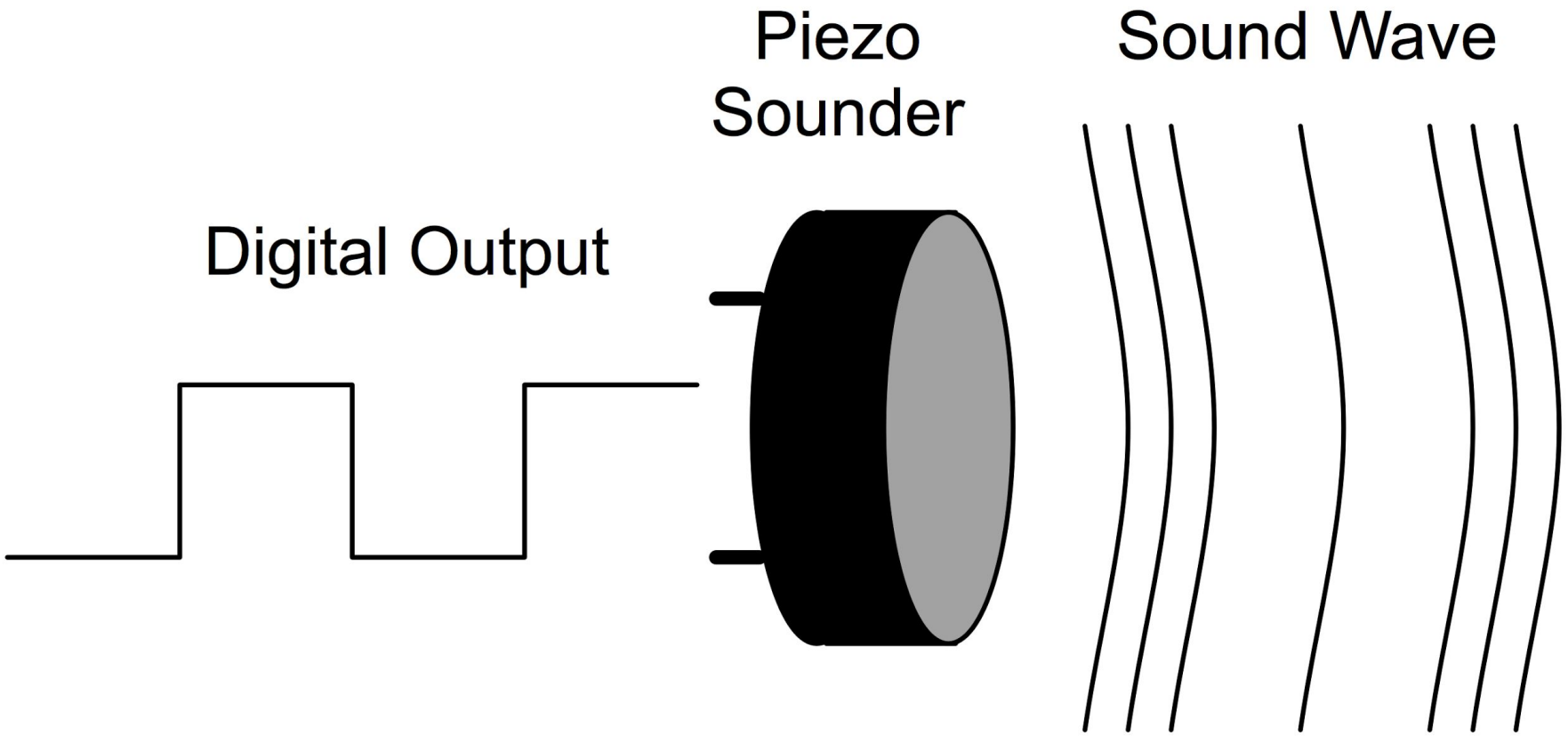
2. Solder two leads onto the disk. One on the positive and one on the negative
   ***Do not stay on the crystal too long when you are soldering**

Making noise

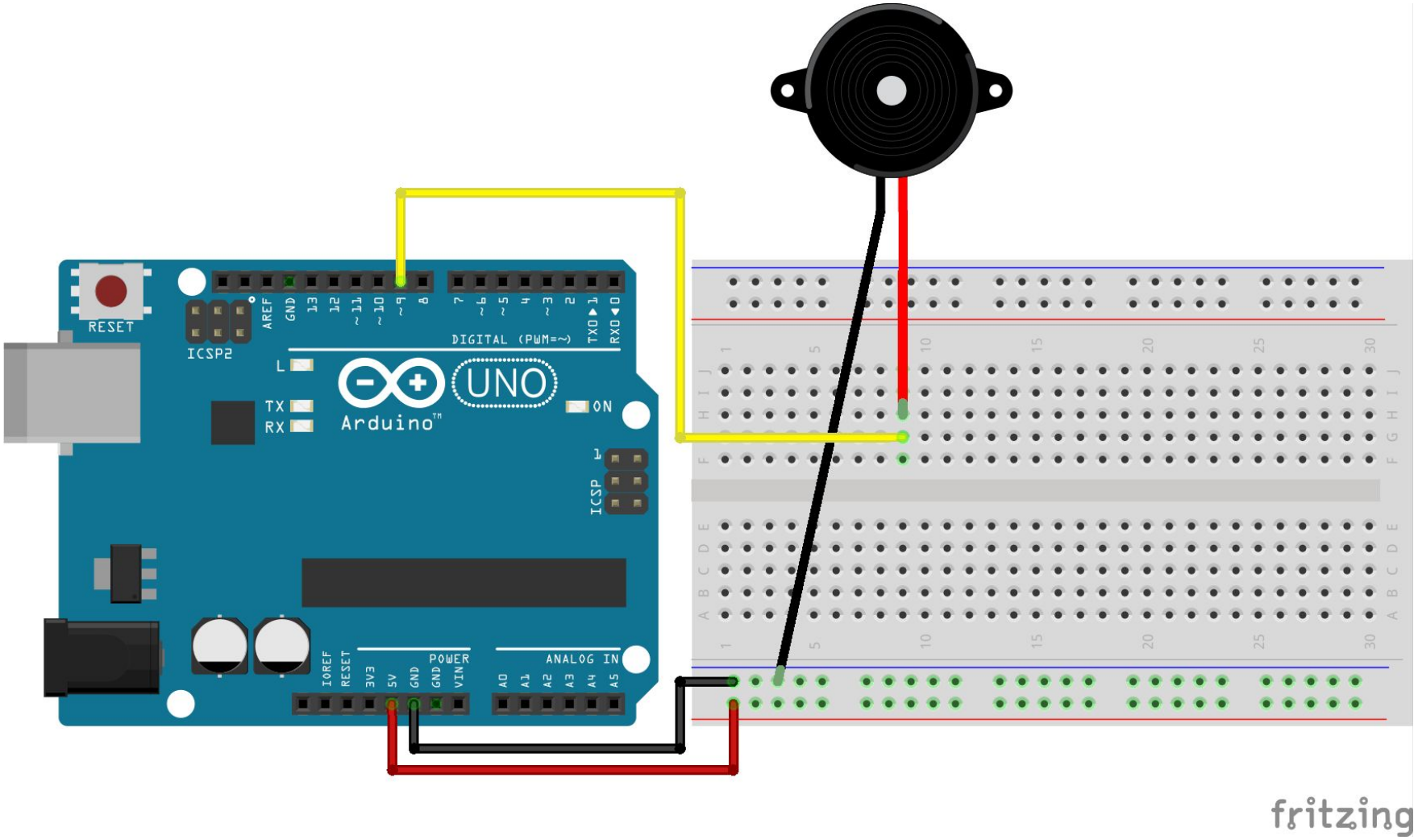
Though PWM can be used to create some “interesting” sounds with a piezo buzzer, better results can be had using a function called tone().

Note: tone() can produce some sweet, sweet music, it will also wreak havoc on PWM pins 3 and 11, so try to avoid using it at the same time as other analog output on those pins.

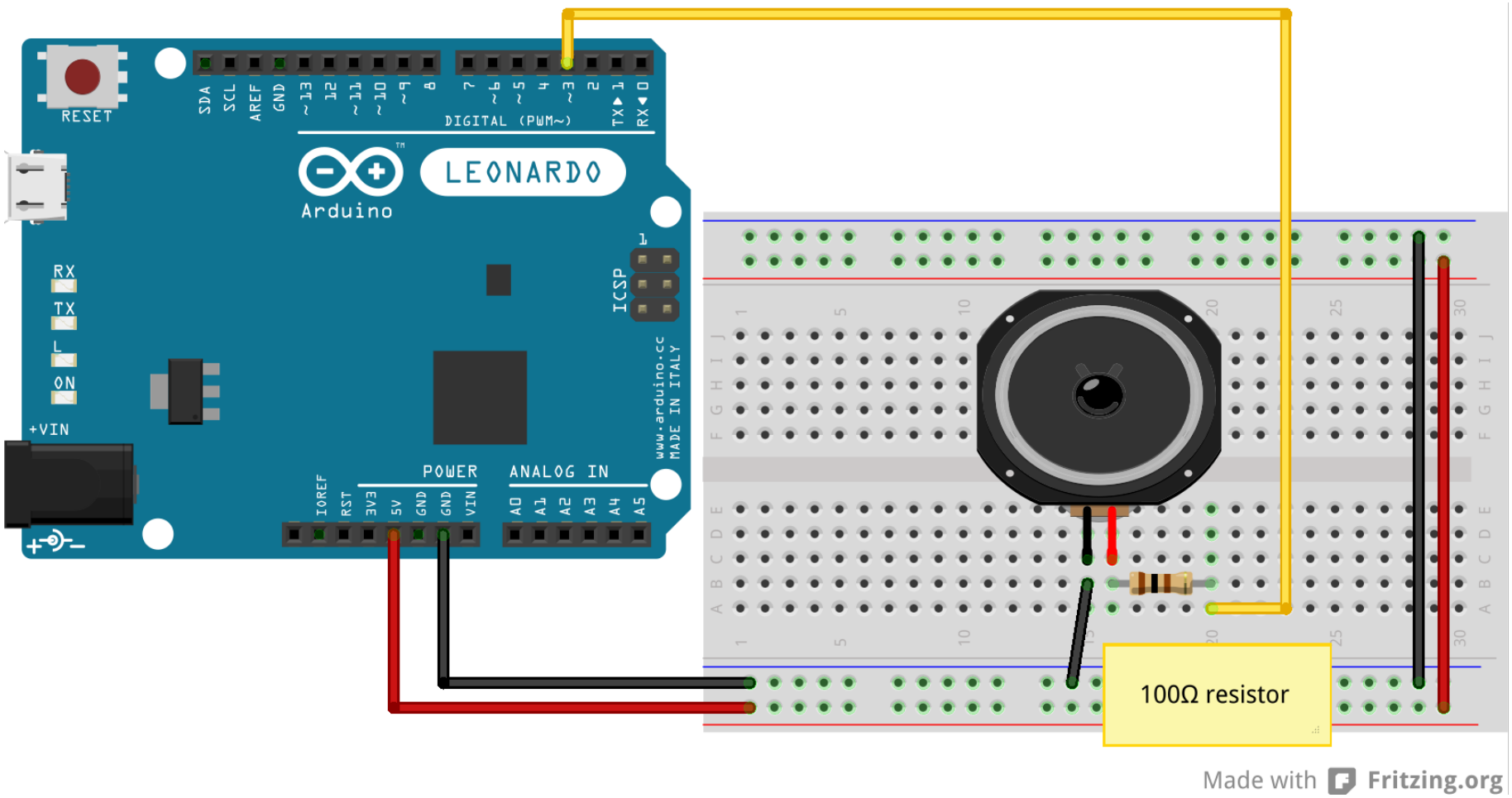
Piezo making a sound



Using piezo as a speaker



Connecting an 8Ω Speaker



tone()

tone() is used like this:

tone(pin, frequency in Hz);
delay(in milliseconds);

* Frequency to note chart here.

Too ~~lazy~~ busy to mess with Hz?

Me too!

Download the pitches.h file linked to from the class site. Add it to your Arduino sketch folder.

Then use it like this...

Using pitches.h

```
#include "pitches.h" // Add the pitches.h file
int speakerPin = 3;

void setup() {
  // Nothing needed!
}

void loop() {
  tone(speakerPin, NOTE_C4); // Refer to the variable from pitches.h
  delay(1000);
  tone(speakerPin, NOTE_FS4);
  delay(1000);
}
```

Map function

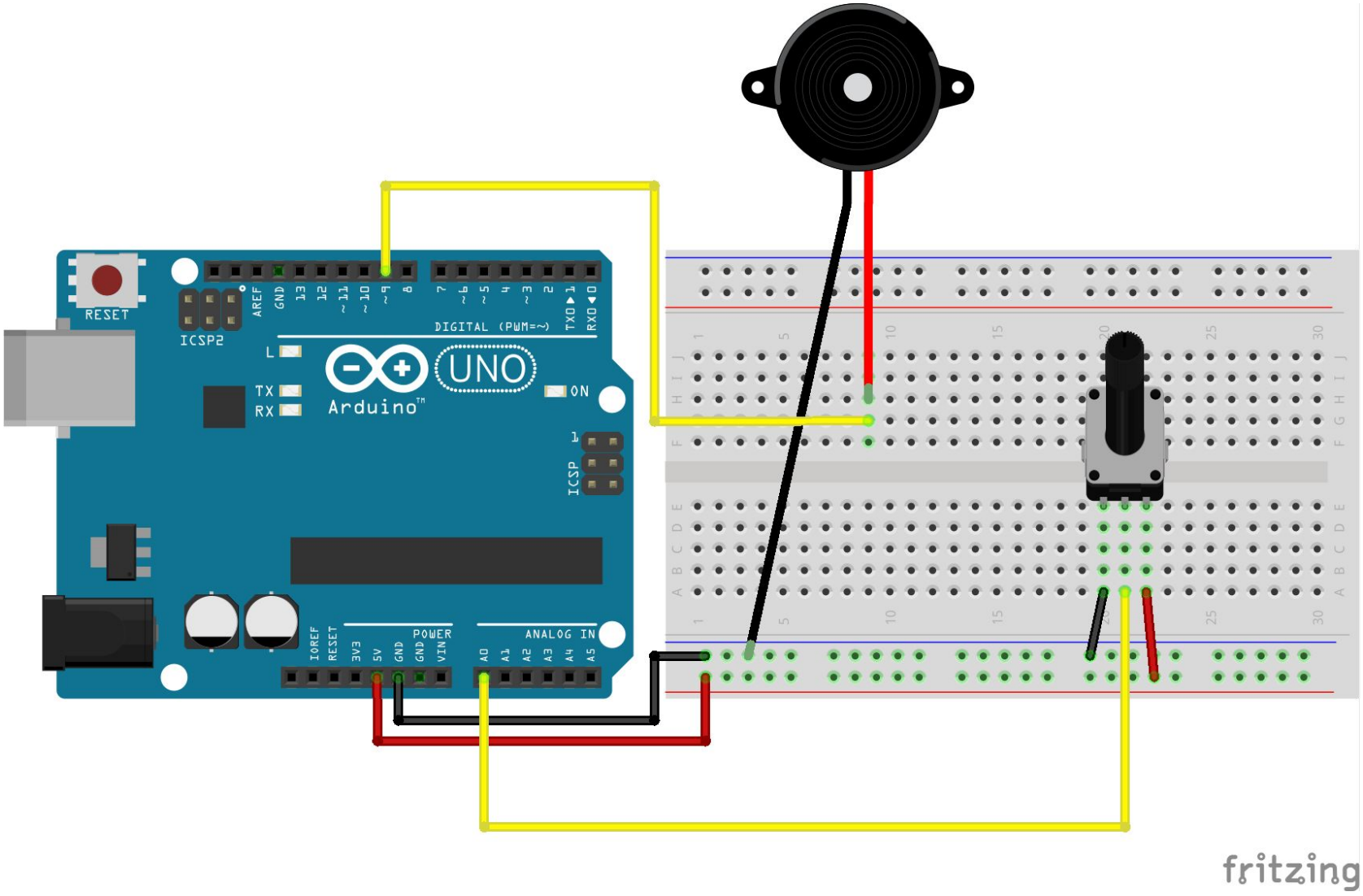
map(value, fromLow, fromHigh, toLow, toHigh)

Re-maps a number from one range to another. That is, a value of fromLow would get mapped to toLow, a value of fromHigh to toHigh, values in-between to values in-between, etc.

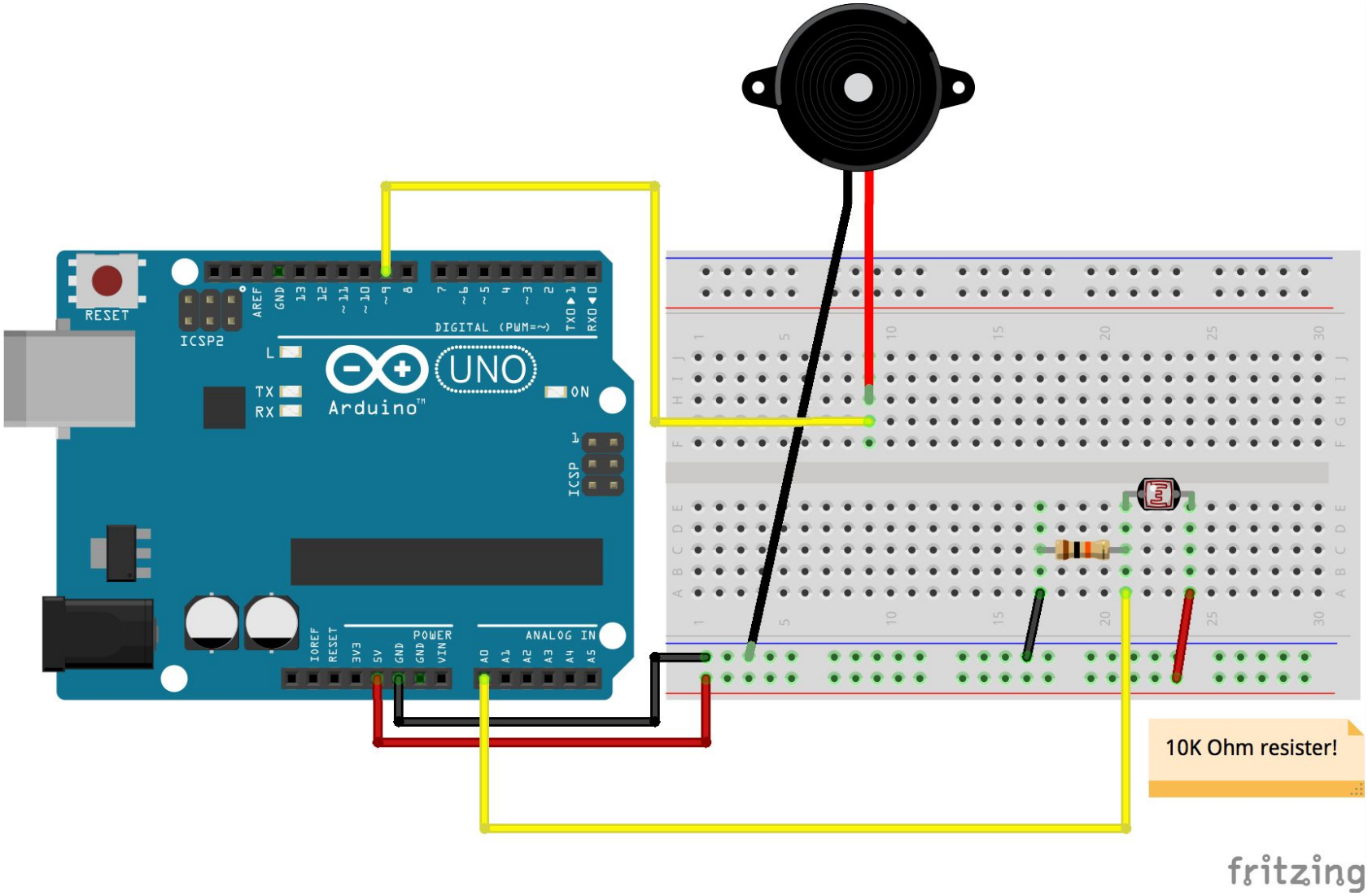
```
int reading = analogRead(photo1);
int pitch;
 pitch = map(reading, 0,1023,0,3000);
 tone(speakerPin, pitch);
 delay(1);
```

Map function on Arduino reference page

Piezo with Analog Input



Piezo with Analog Input2



Random Function

The random function generates **pseudo**-random numbers. (meaning, they are generated by mathematical formula that restarts at the same place everytime arduino turns on, and it cycle back the same sequence at one point)

Syntax

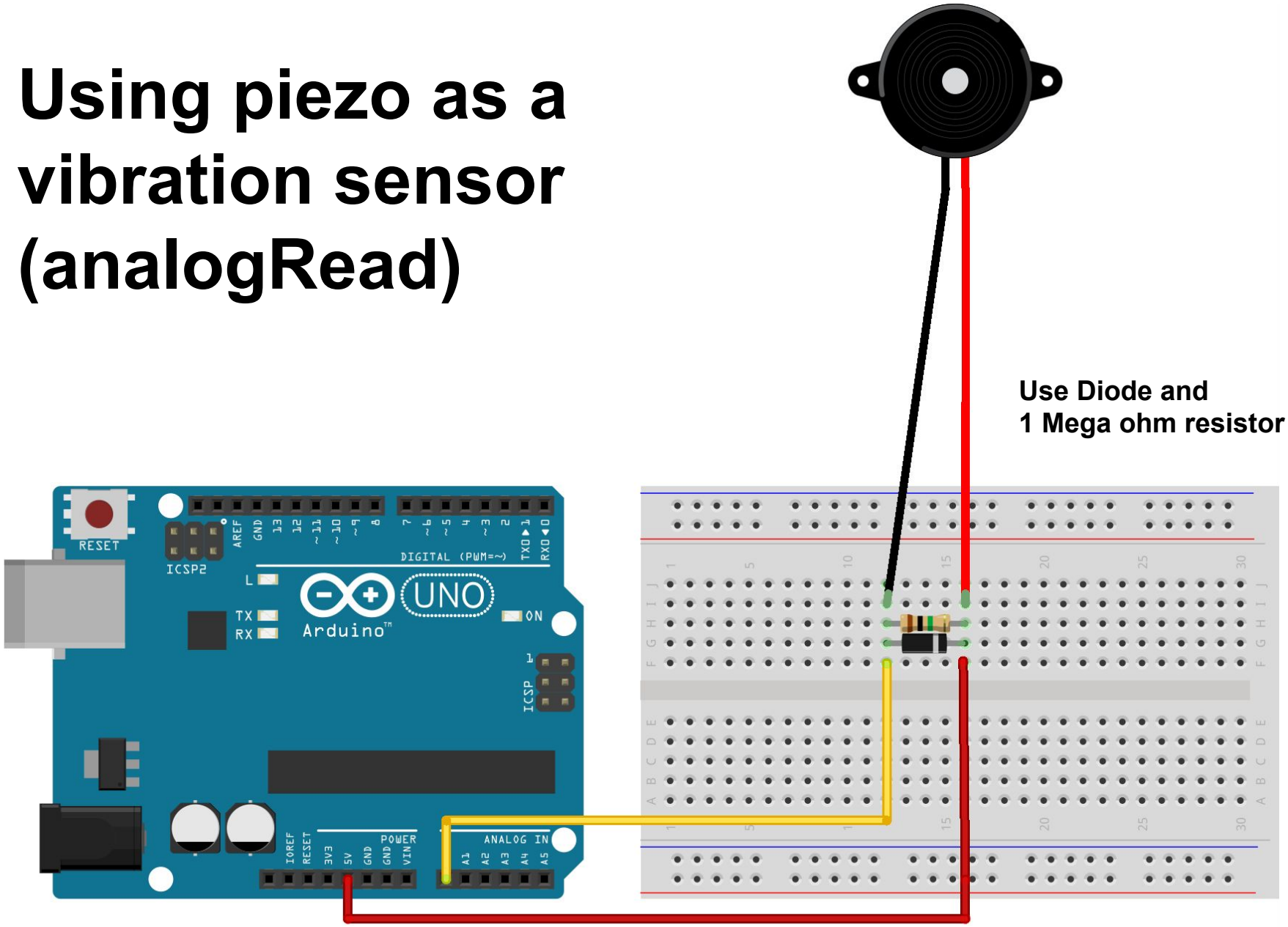
random(max)

random(min, max)

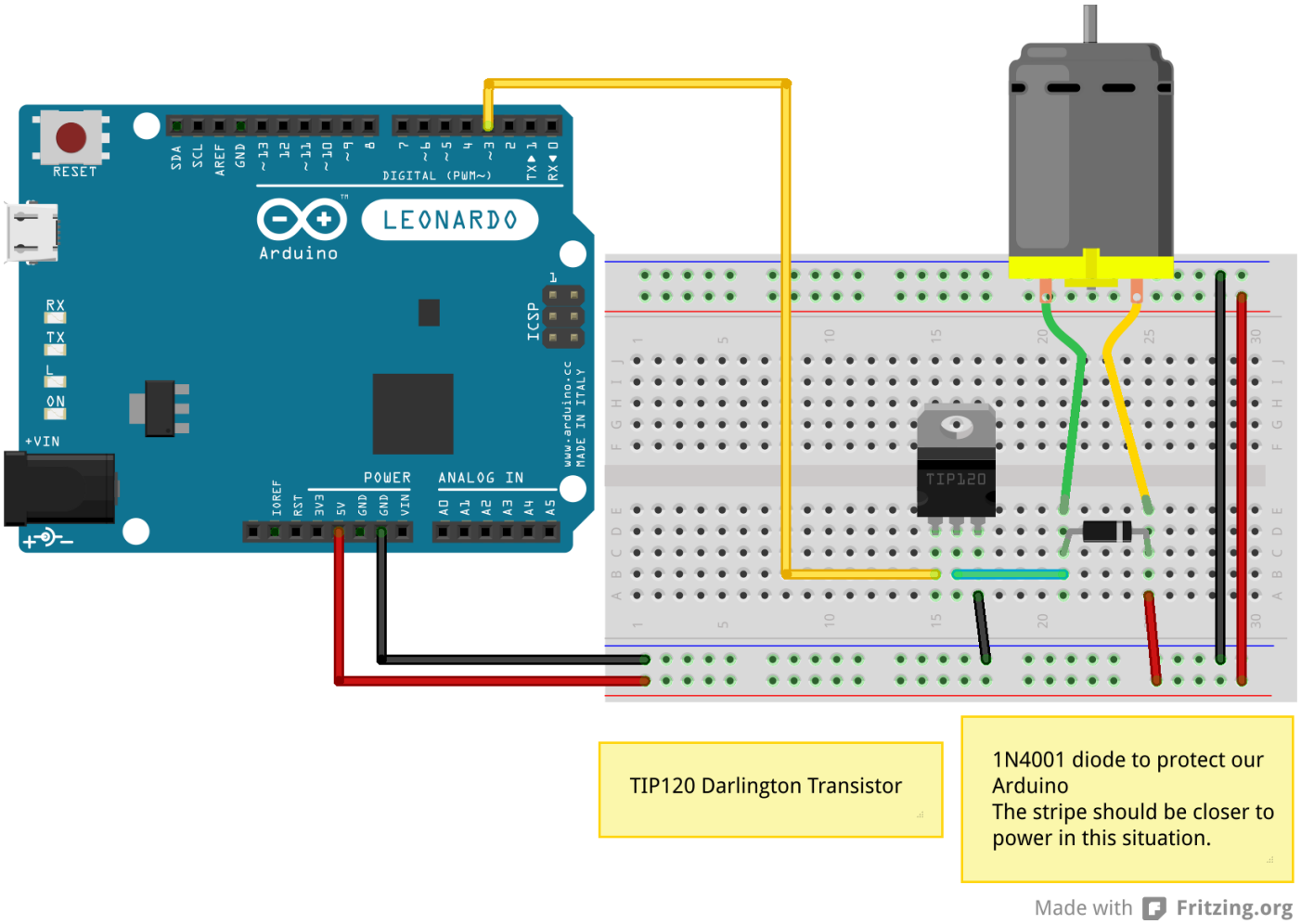
More about Random function at Random

If you want to get closer to true random function, refer to RandomSeed

Using piezo as a vibration sensor (analogRead)



Connecting a small DC motor*



*Do not attempt this with a larger DC motor!