FABRICADEMY 2022-23

TUTORIALS

ARDUINO

FABRICADEMY
textile and technology academy

ARDUINO and SENSORS TUTORIAL

EMMA PARESCHI

Fabricademy | textile & technology academy 2022

Arduino

``***Physical Computing*** is about prototyping with electronics, turning sensors, actuators and microcontrollers into materials for designers and artists."

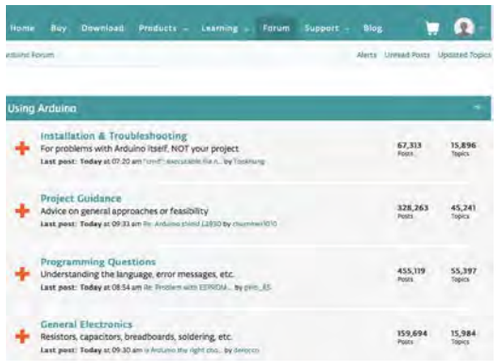
``It involves the design of interactive objects that can communicate with humans using sensors and actuators controlled by a behaviour implemented as software running inside a microcontroller."

Massimo Banzi, Tinker.it & Arduino Co-Founder

Hardware

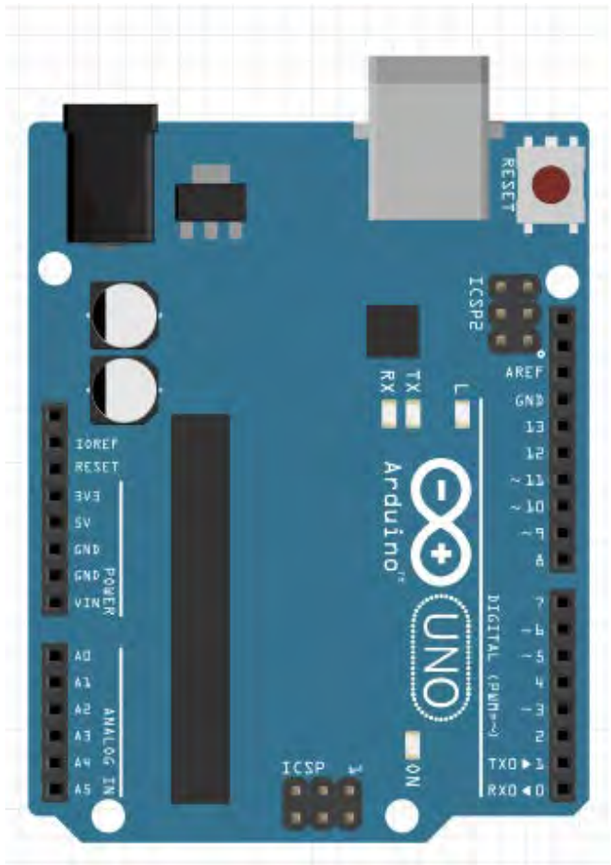
Software

Community



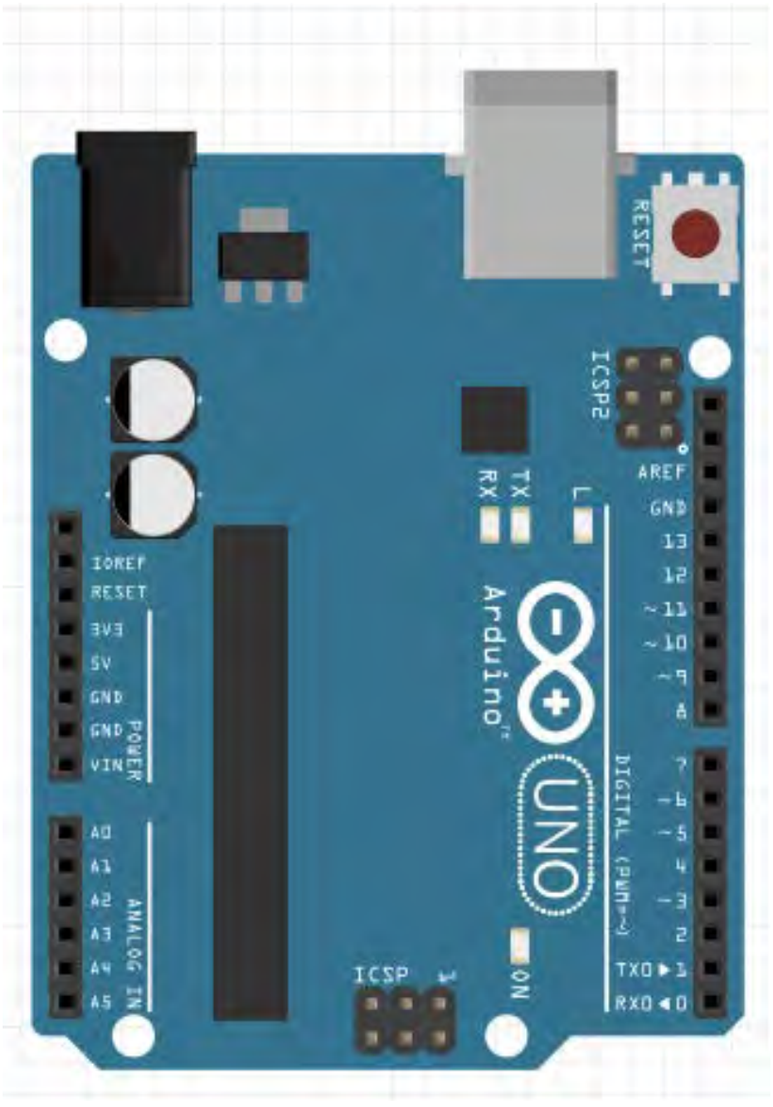
Arduino

Sensor
(input device)



Led
(output device)

Arduino today



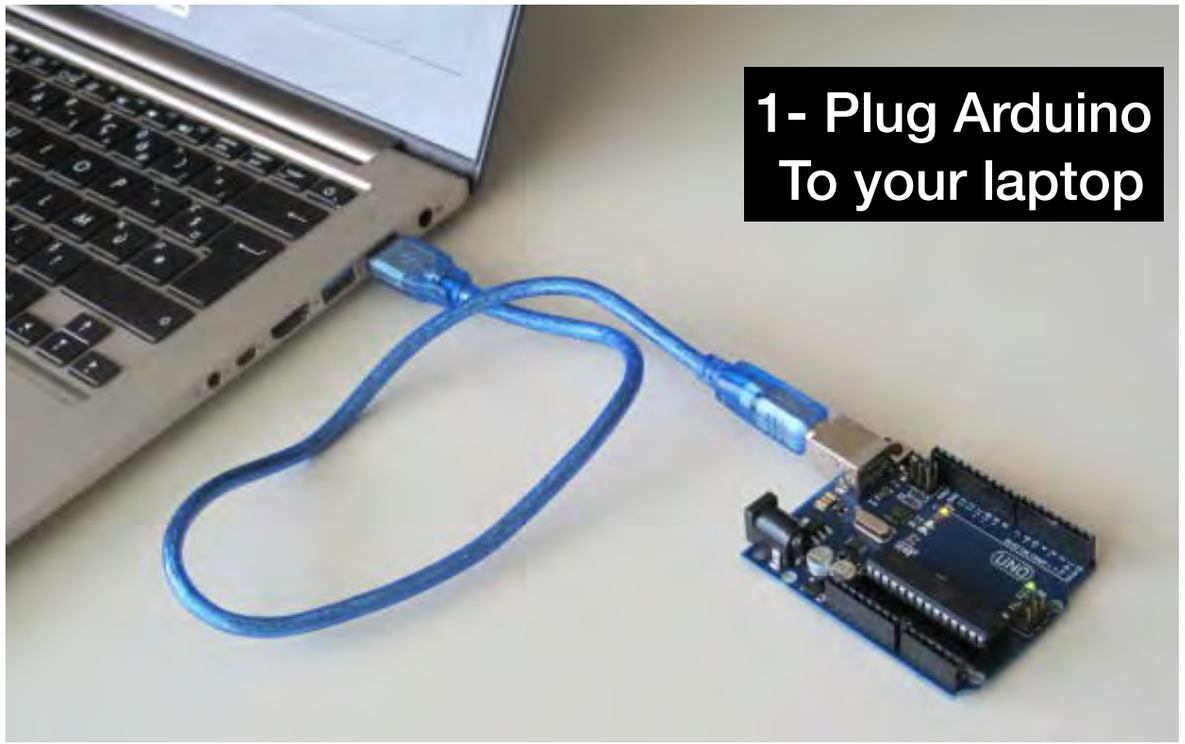
How we will use Arduino

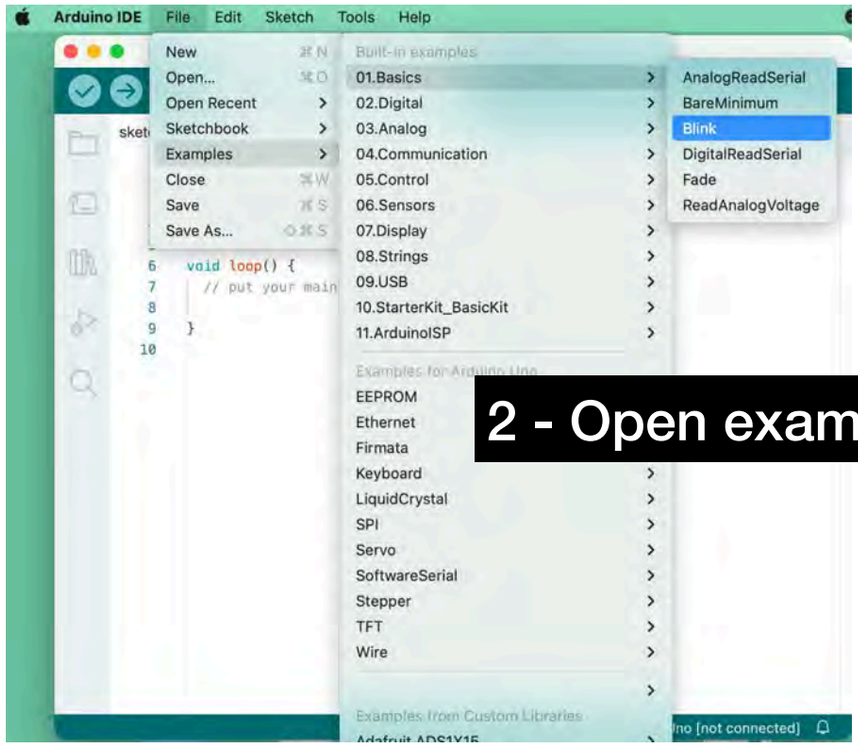
- **control a Led**
- **read a sensor**
- **to control a Led based on the interaction with the sensor**

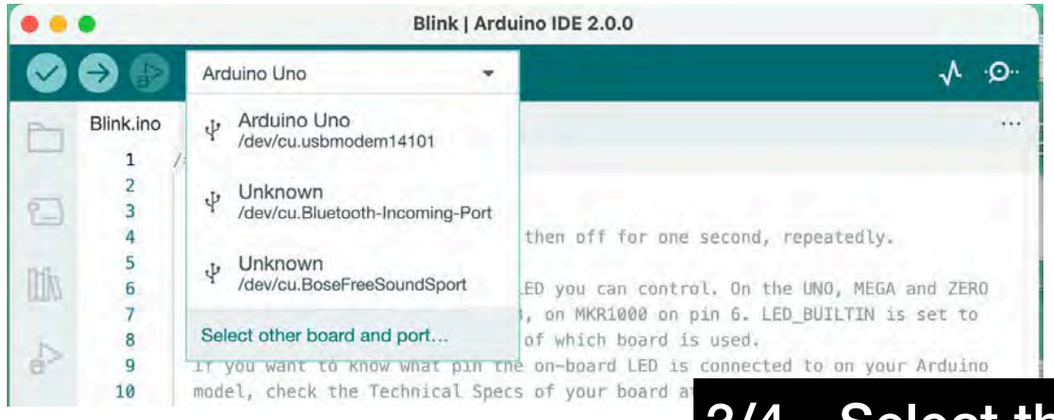
What do we need:

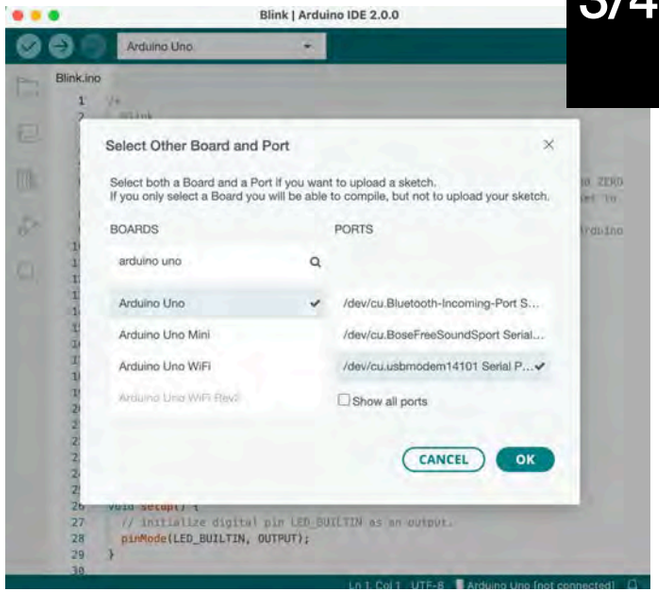
- **make a circuit (with Arduino)**
- **‘write’ (use the right) the code for the circuit**

...check if the environment works

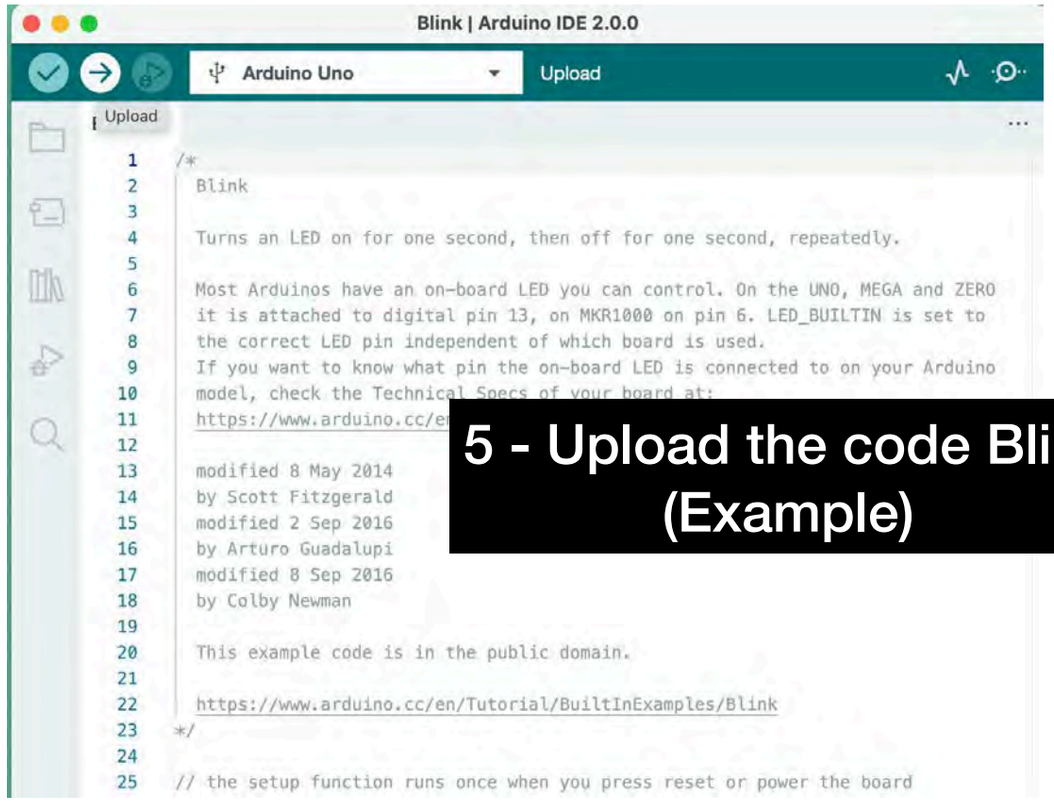








3/4 - Select the port and the Board

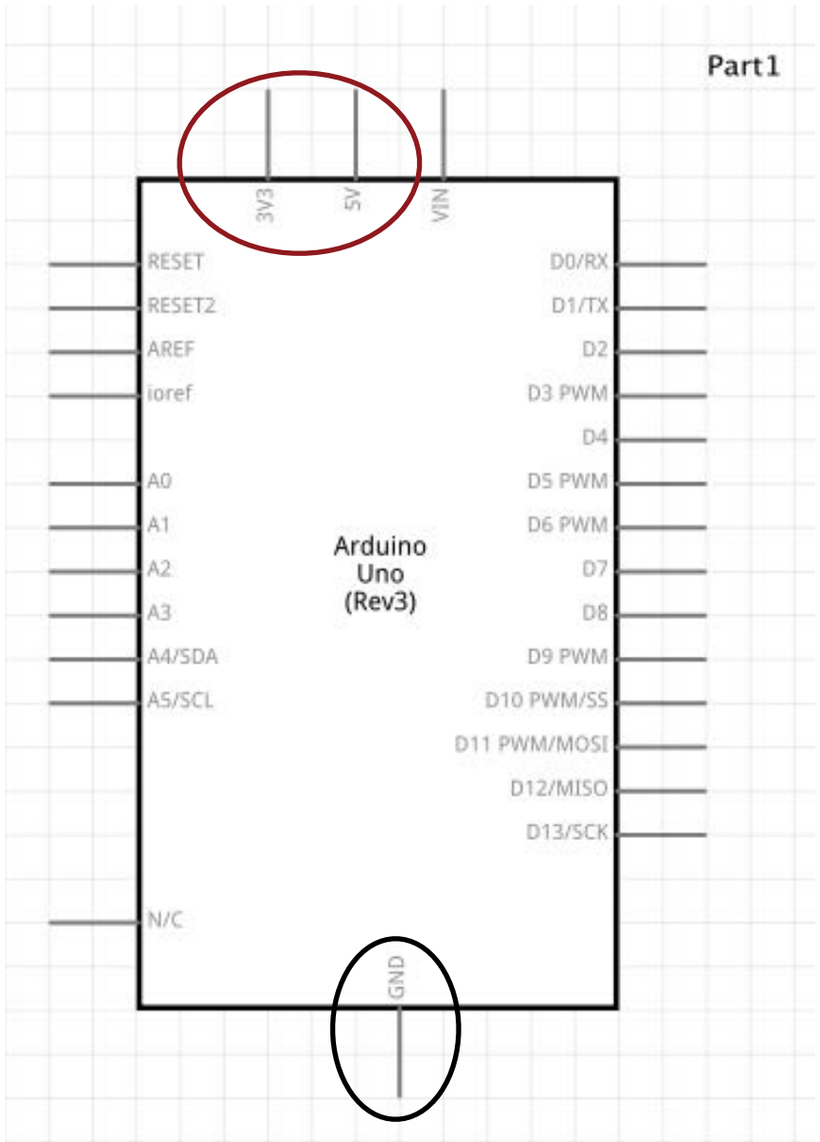


5 - Upload the code Blink (Example)

If it works, the led embedded on the board blinks

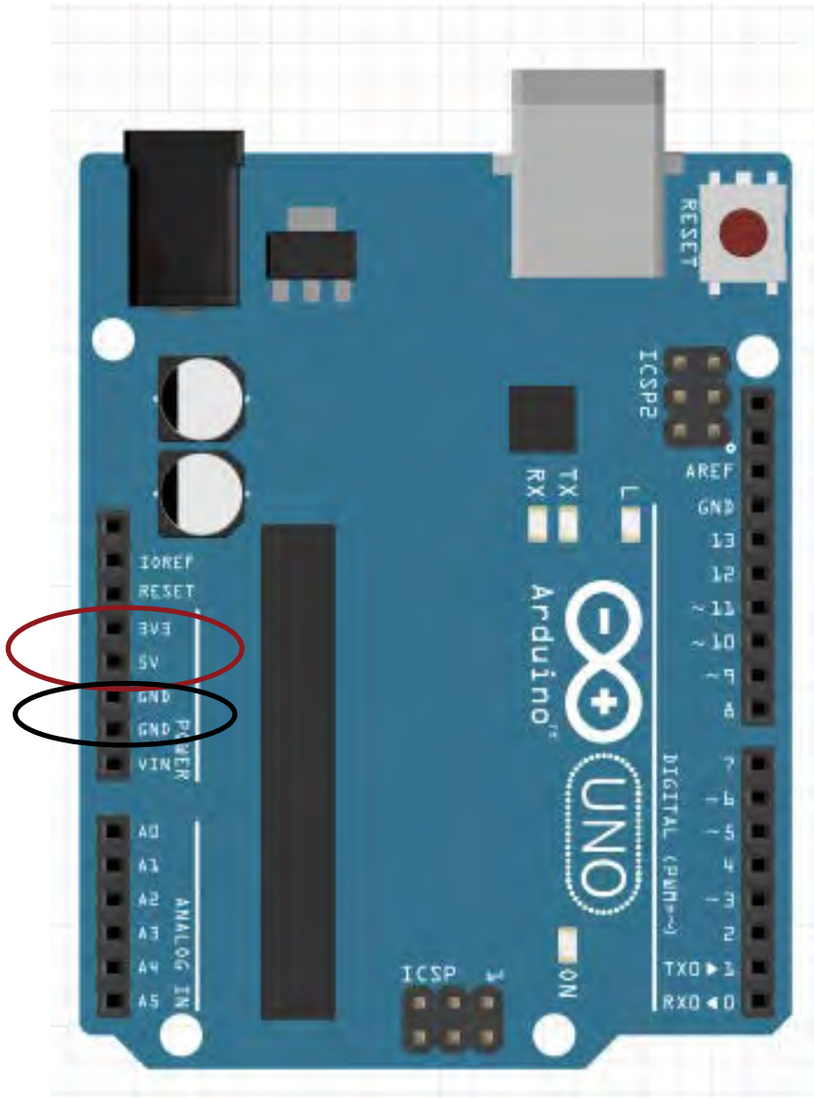
Arduino - Power Pins

Power Pins

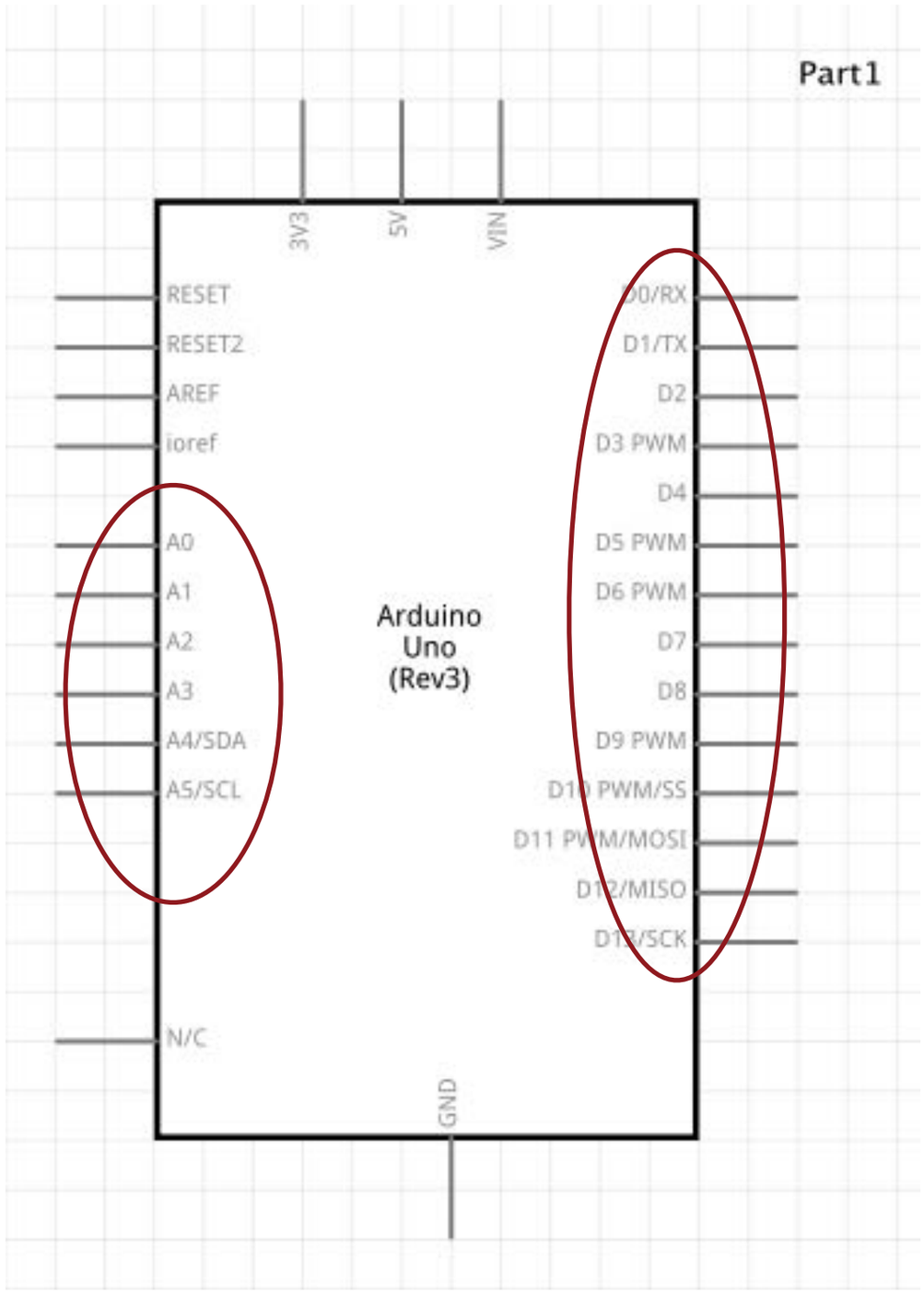


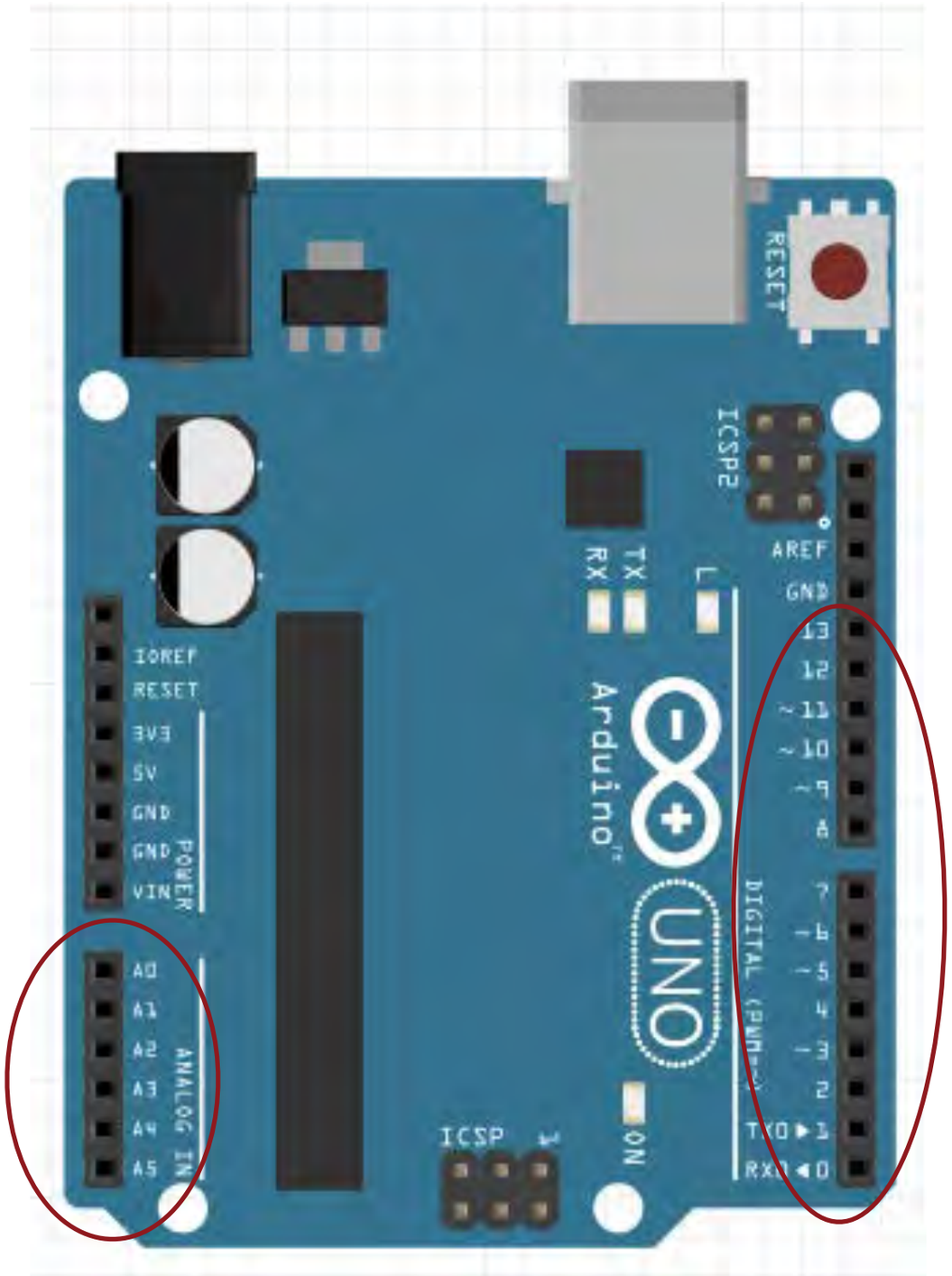
5V / 3.3V:
Power

GND:
Ground Pin

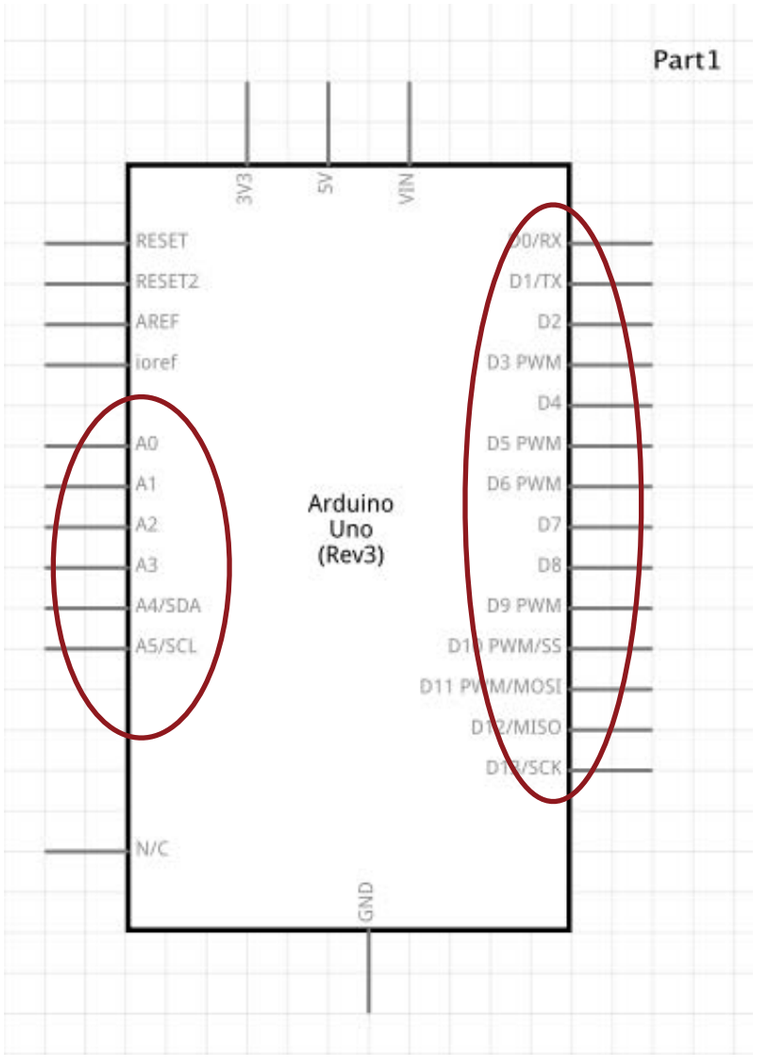


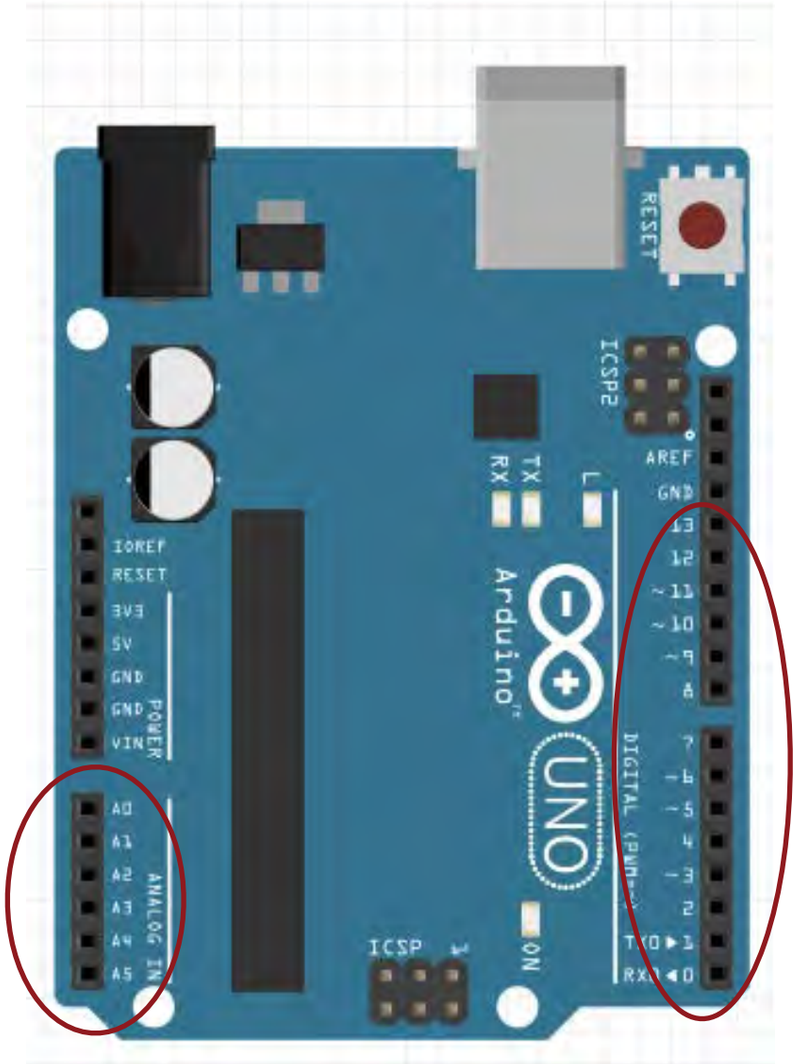
Arduino - Input/Output pins





Arduino - Input/Output pins



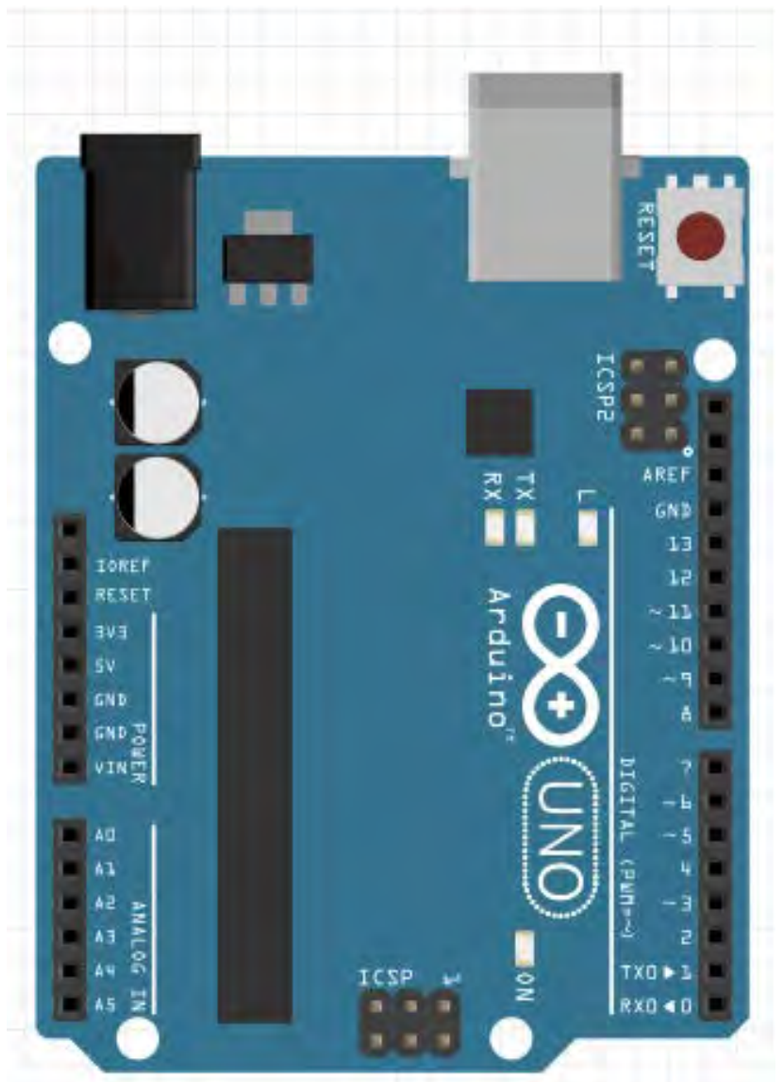


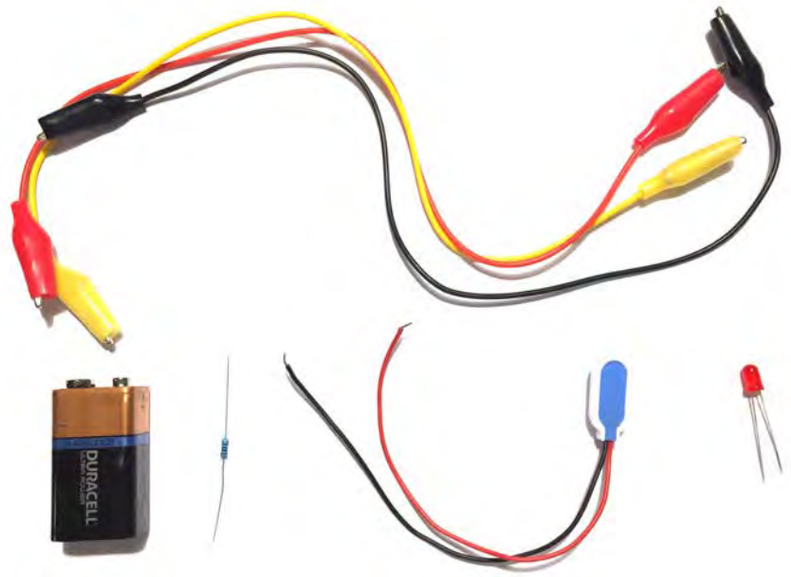
With the INPUT - OUTPUT pins you can:

- **Read Voltage**
- **Control Voltage**

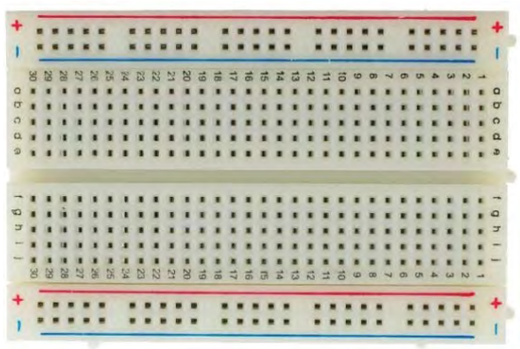
Like we did yesterday with the MULTIMETER!!!!

To make the circuit: tools





?



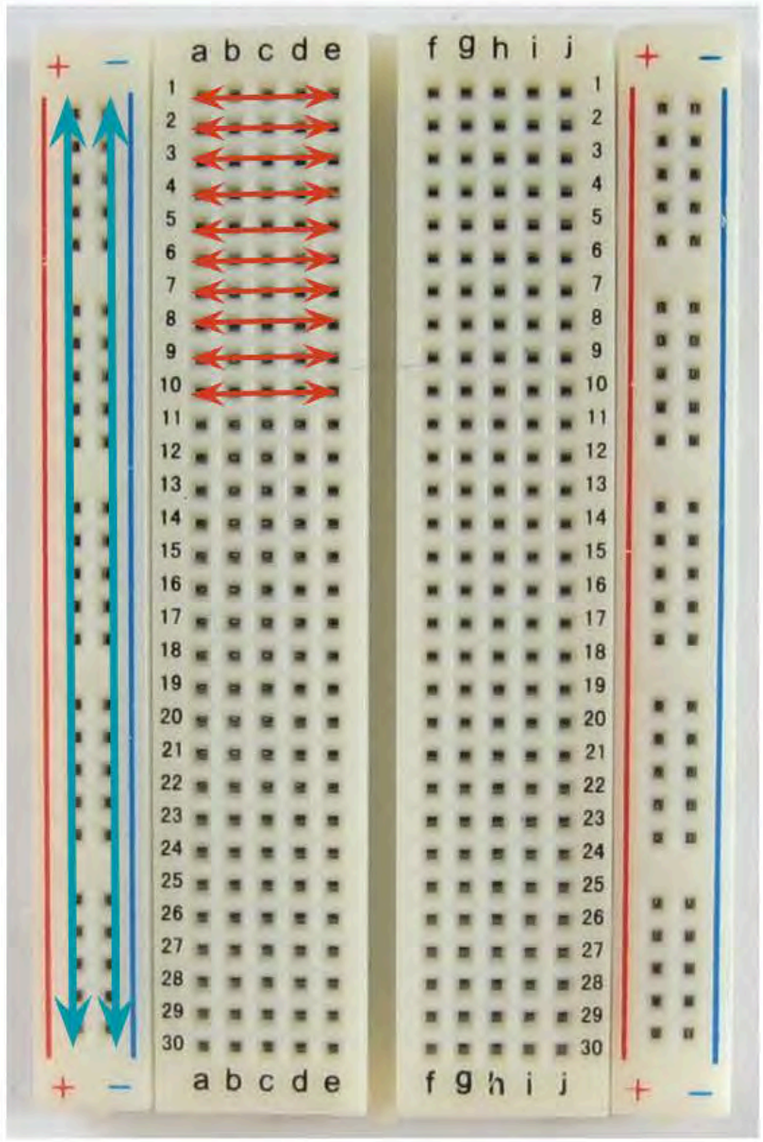
Use to prototype your circuit first.

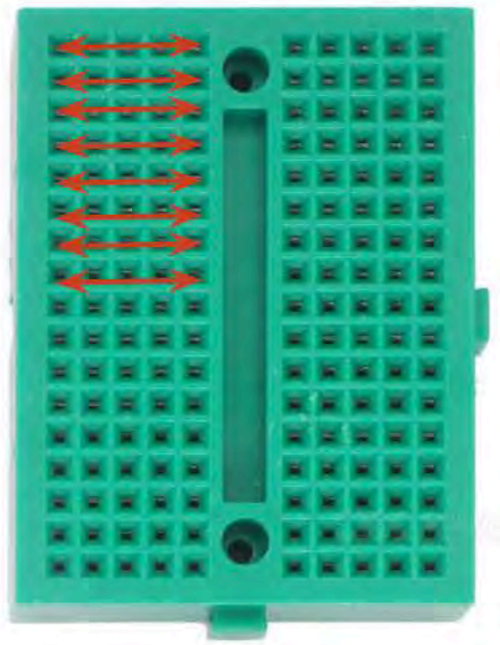
Vertical lines are connected

Horizontal lines are connected

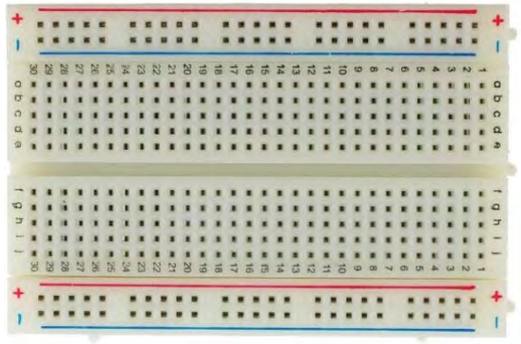
BREADBOARDS

COMPONENTS+TOOLS

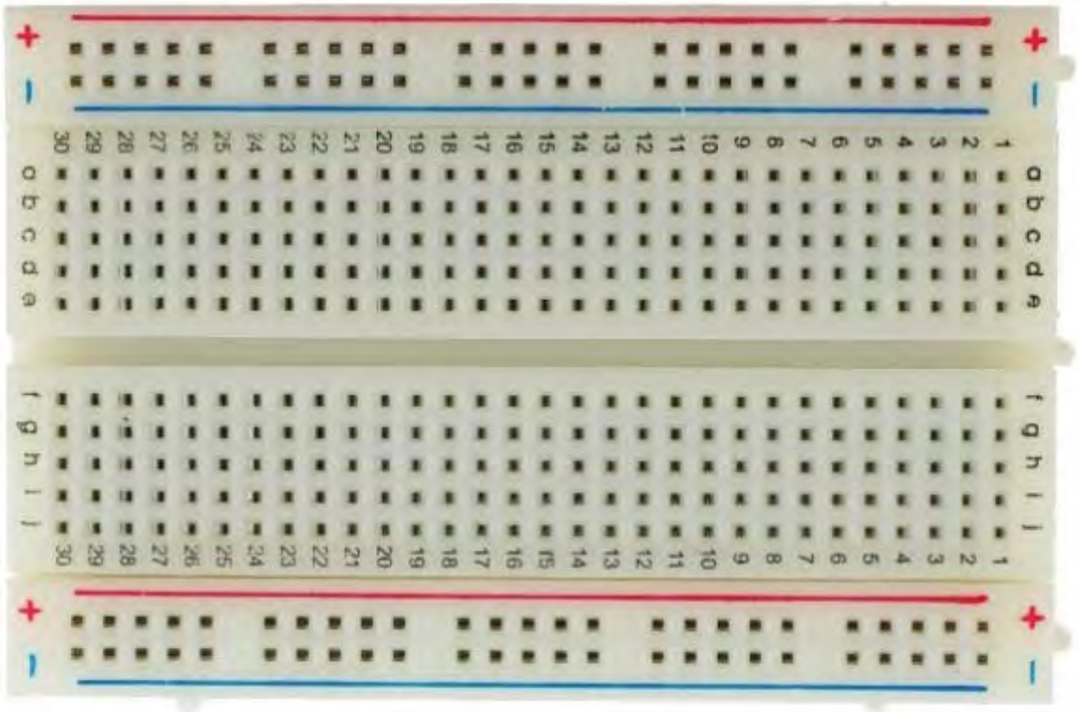


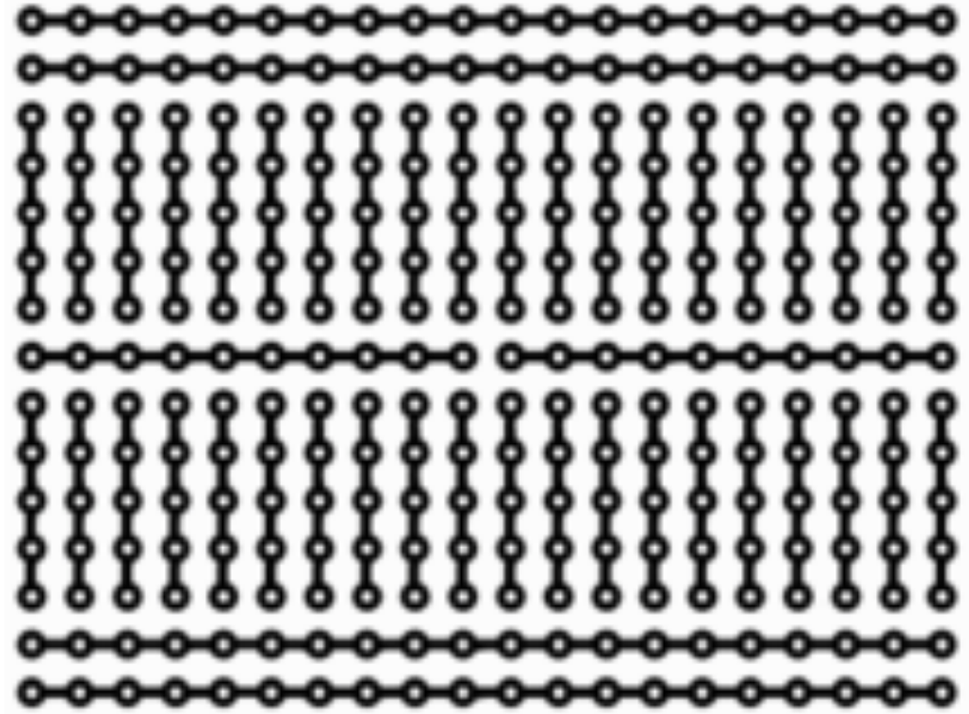


BreadBoard



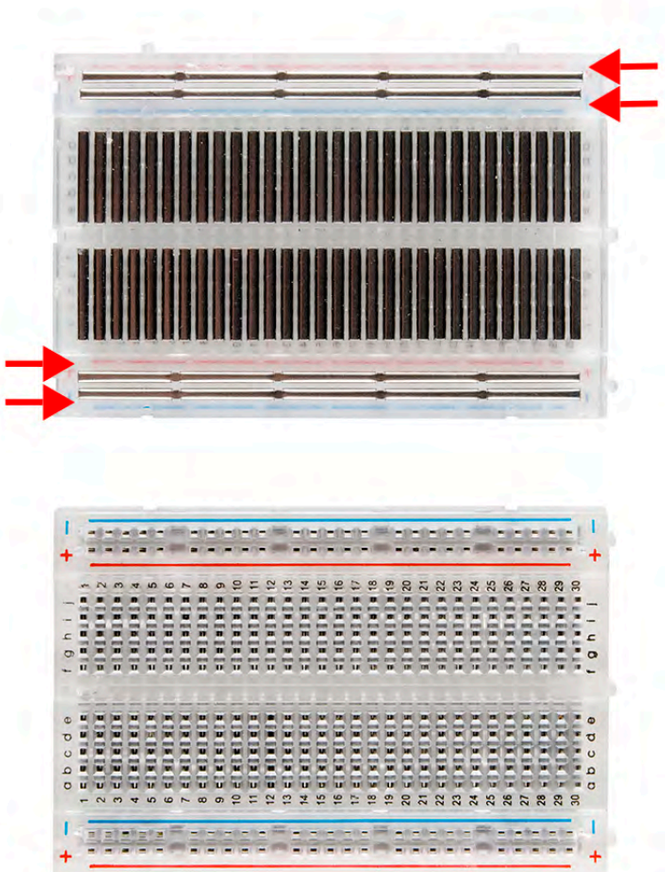
It is a physical support for making temporary circuits and prototyping, and they require absolutely no soldering.



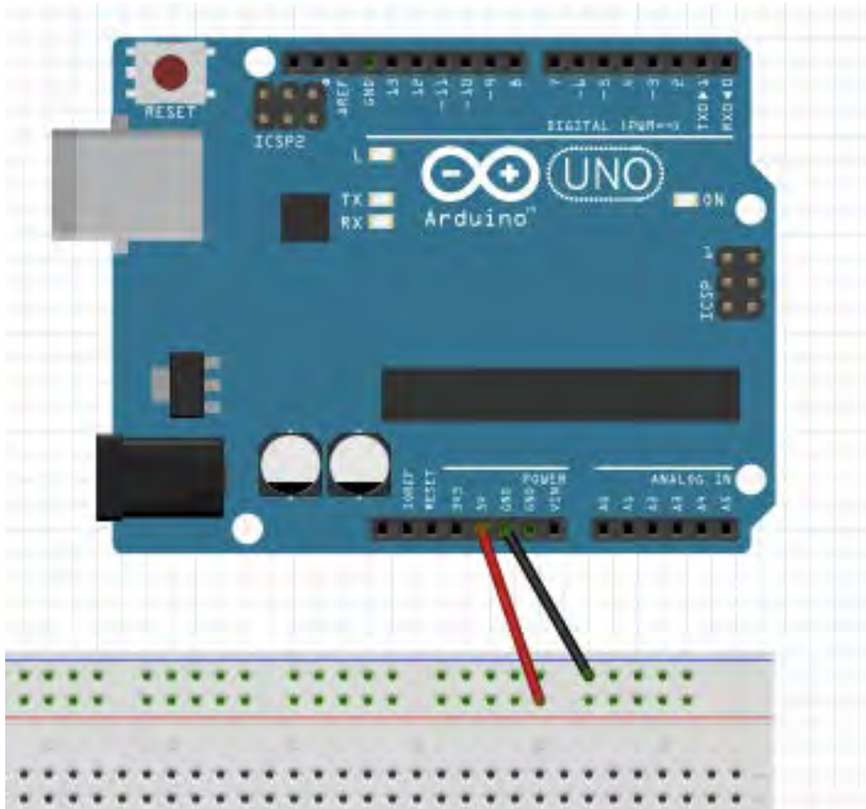


https://learn.sparkfun.com/tutorials/how-to-use-a-breadboard/all

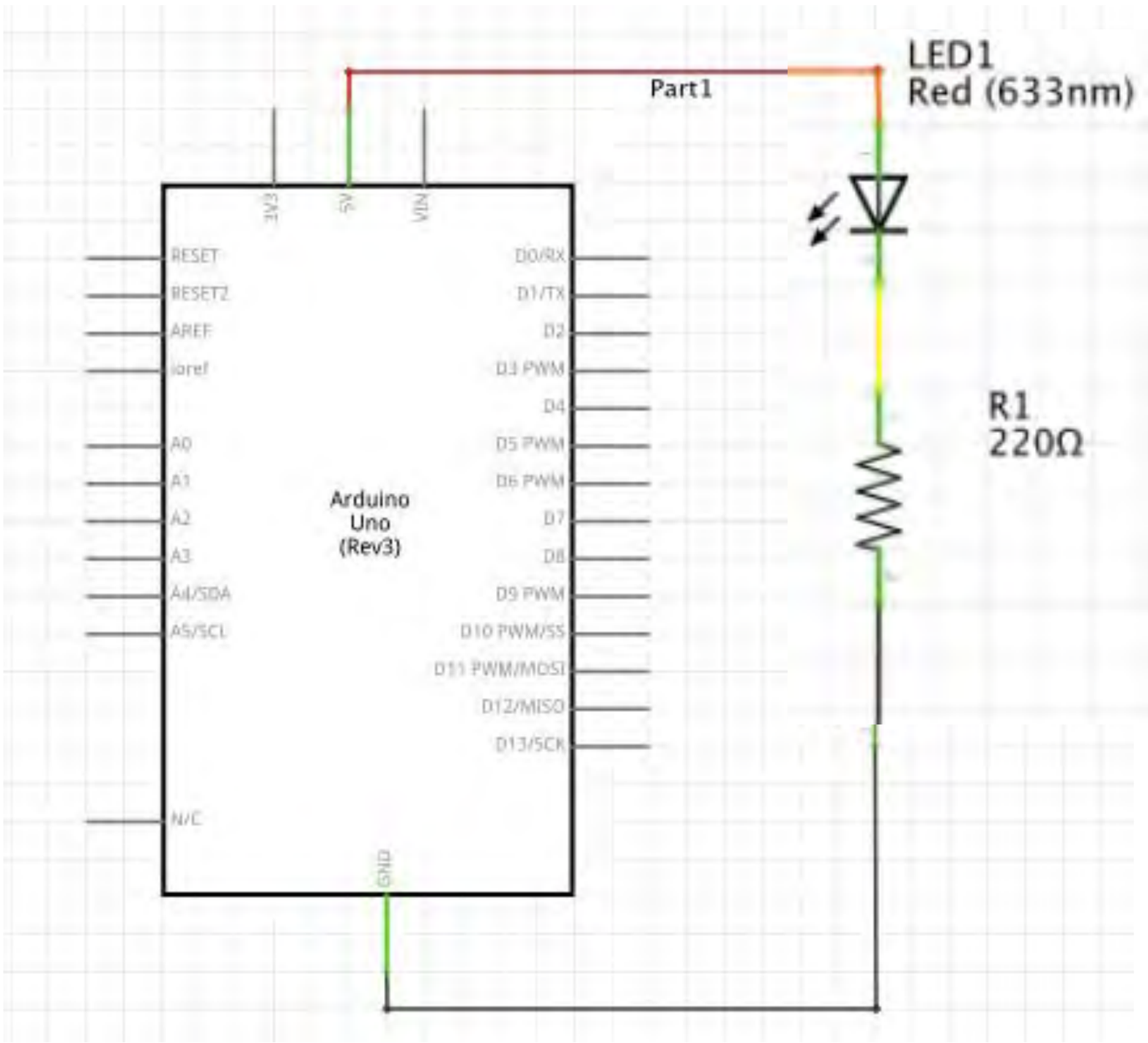
BreadBoard - best practice



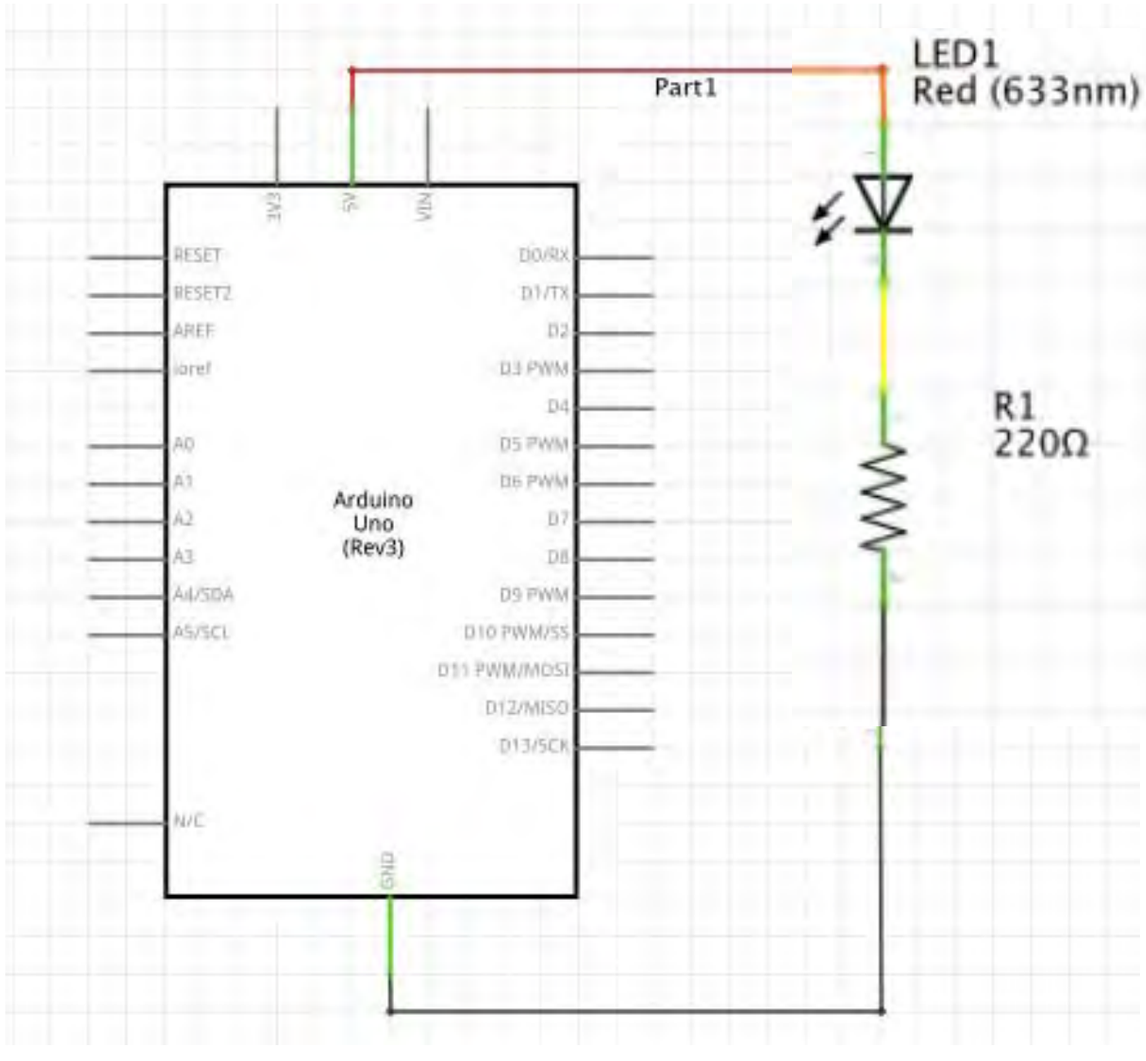
Power Rails

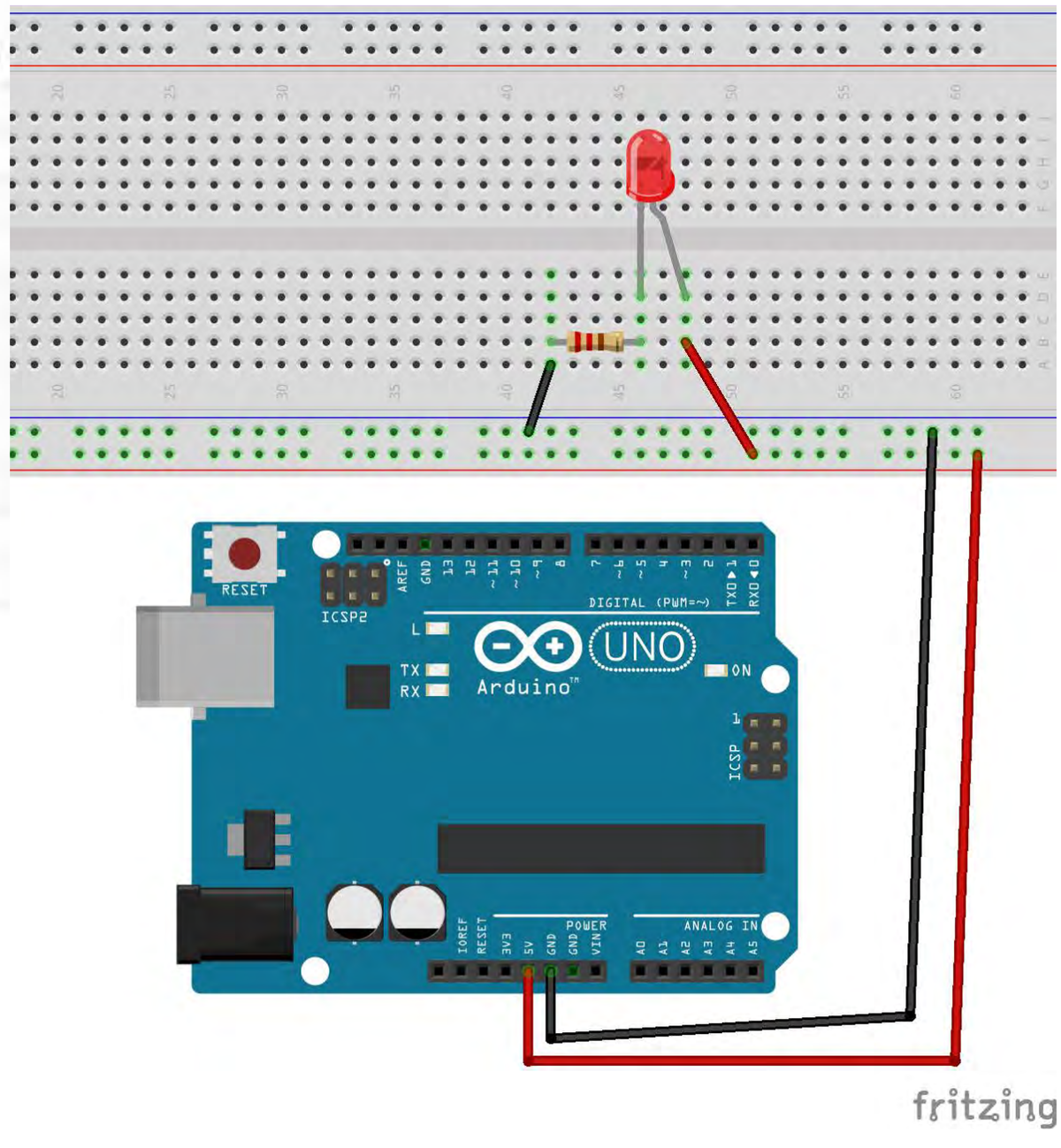


Exercise: Led ON



Exercise: Led ON

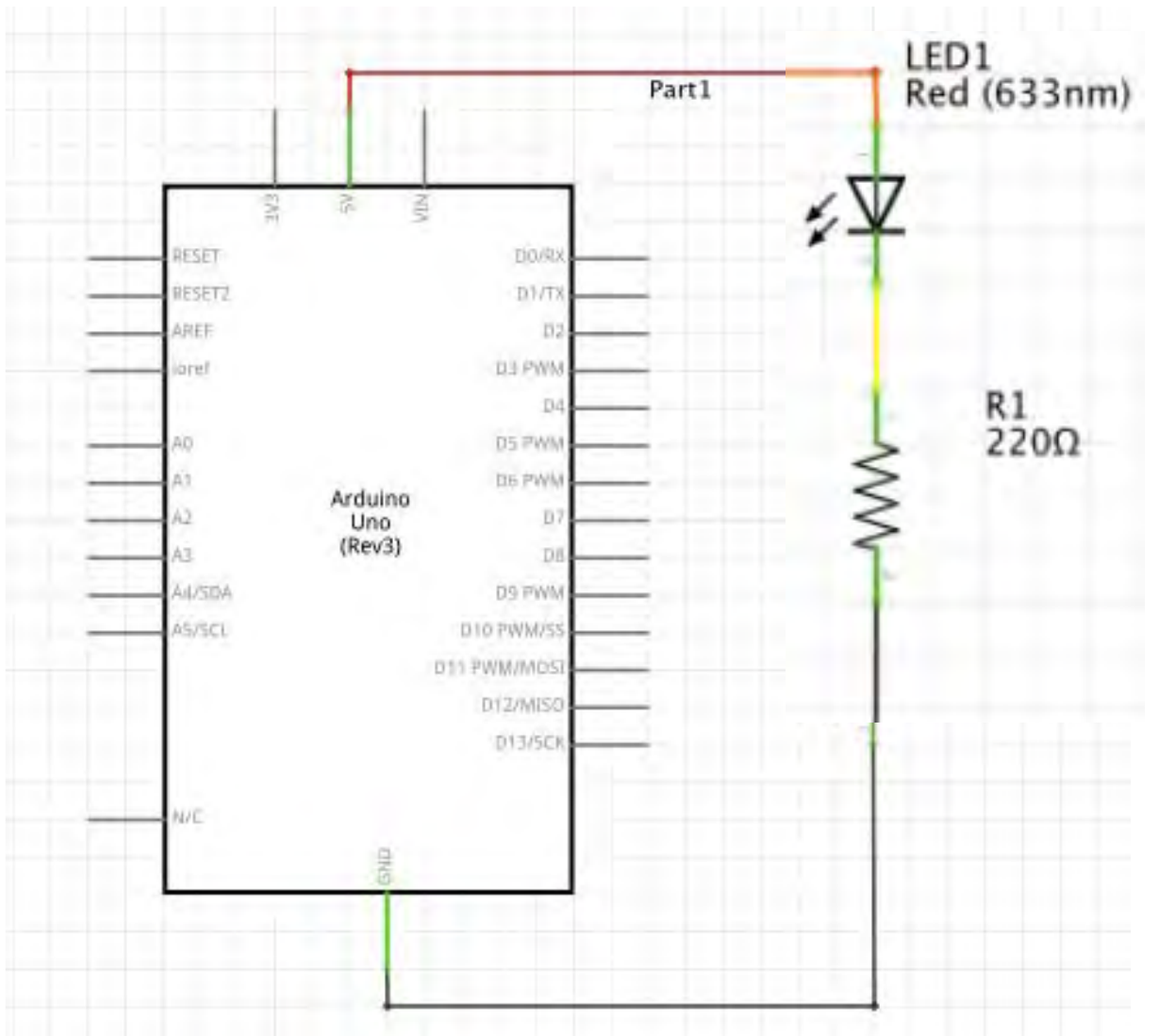




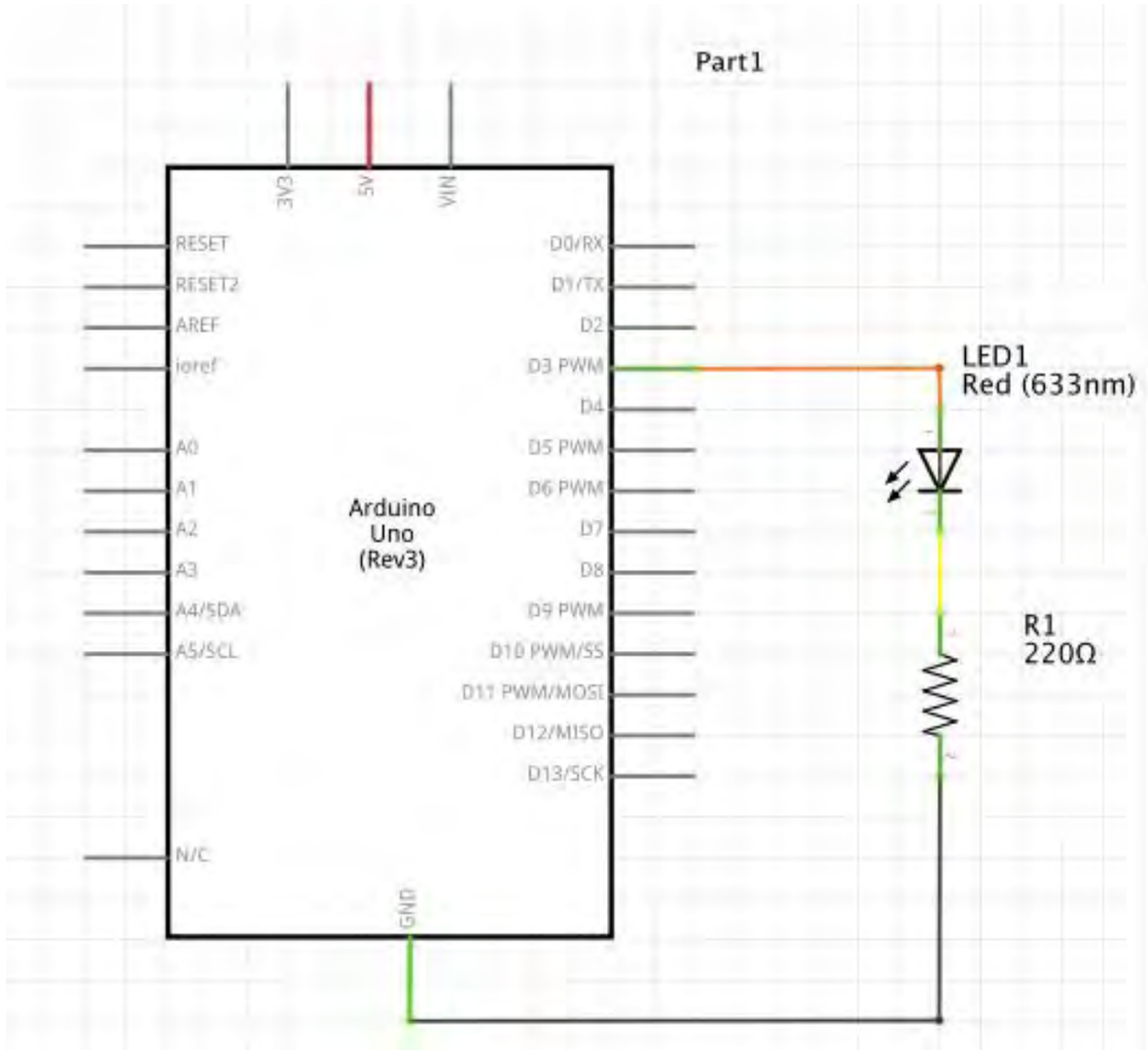
Arduino: Led

Make a Led blinks

Arduino and a Led

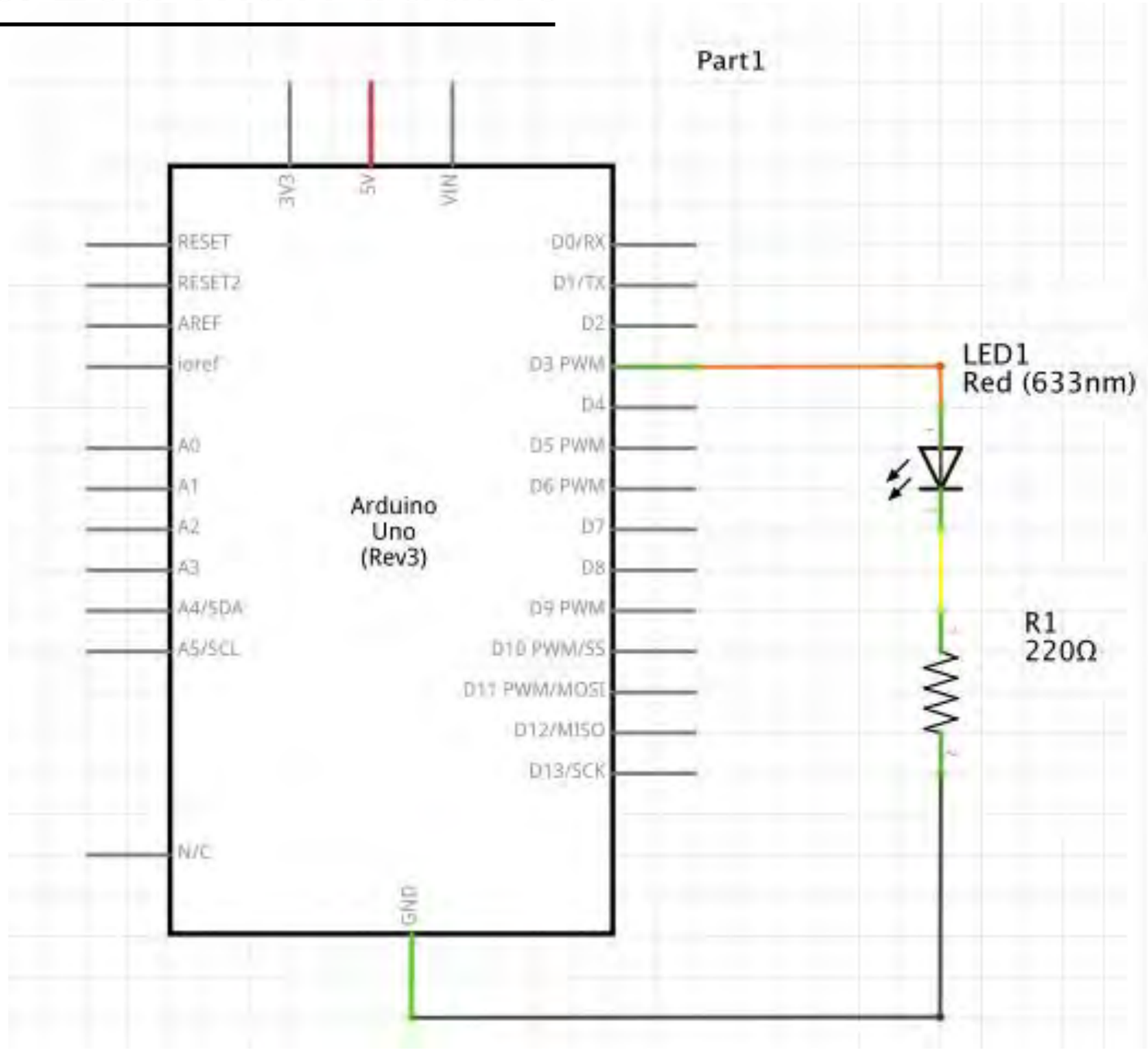


We can’t control the power pins.



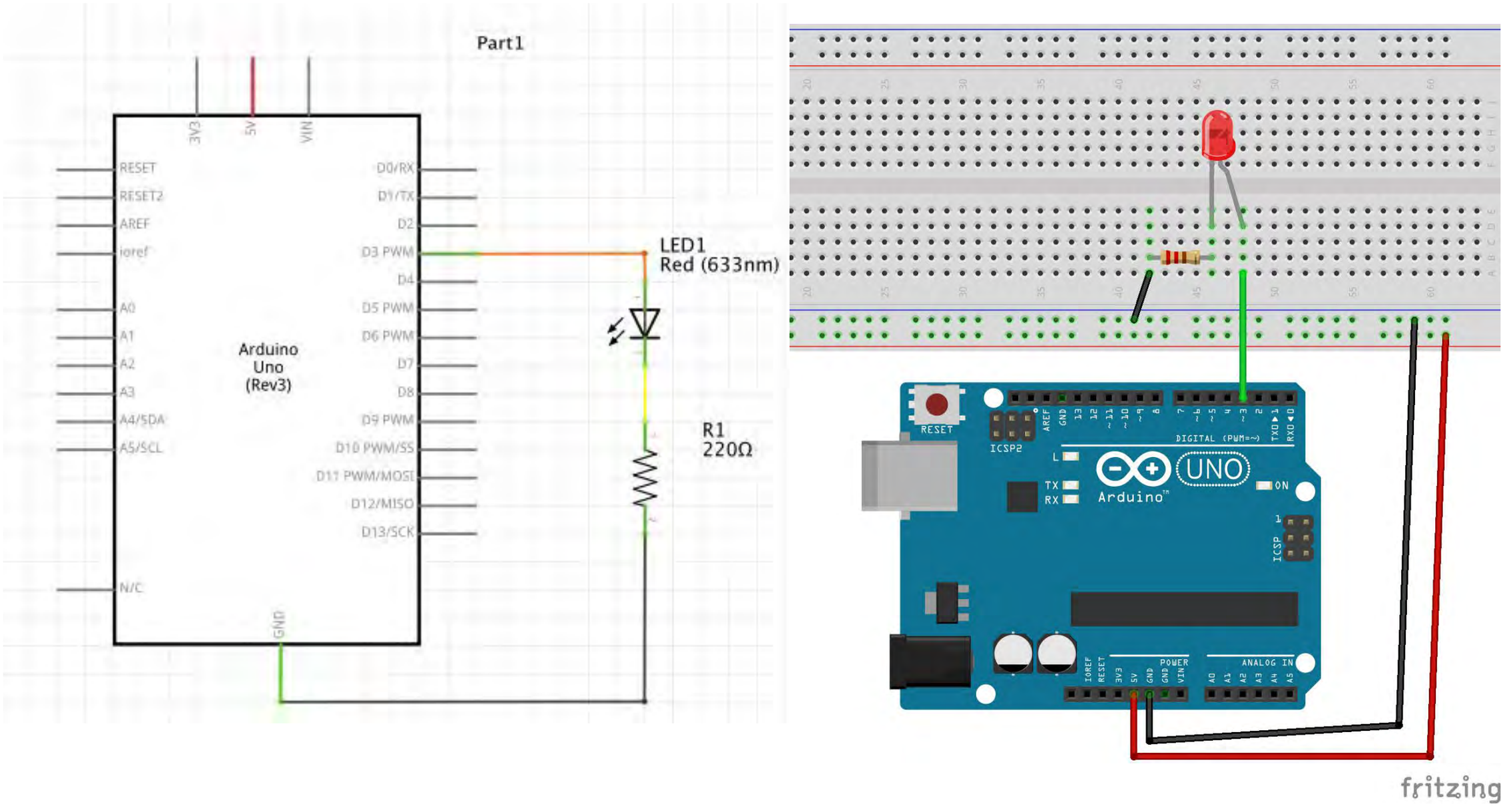
Connect the LED to pin D3

Control a Led with Arduino

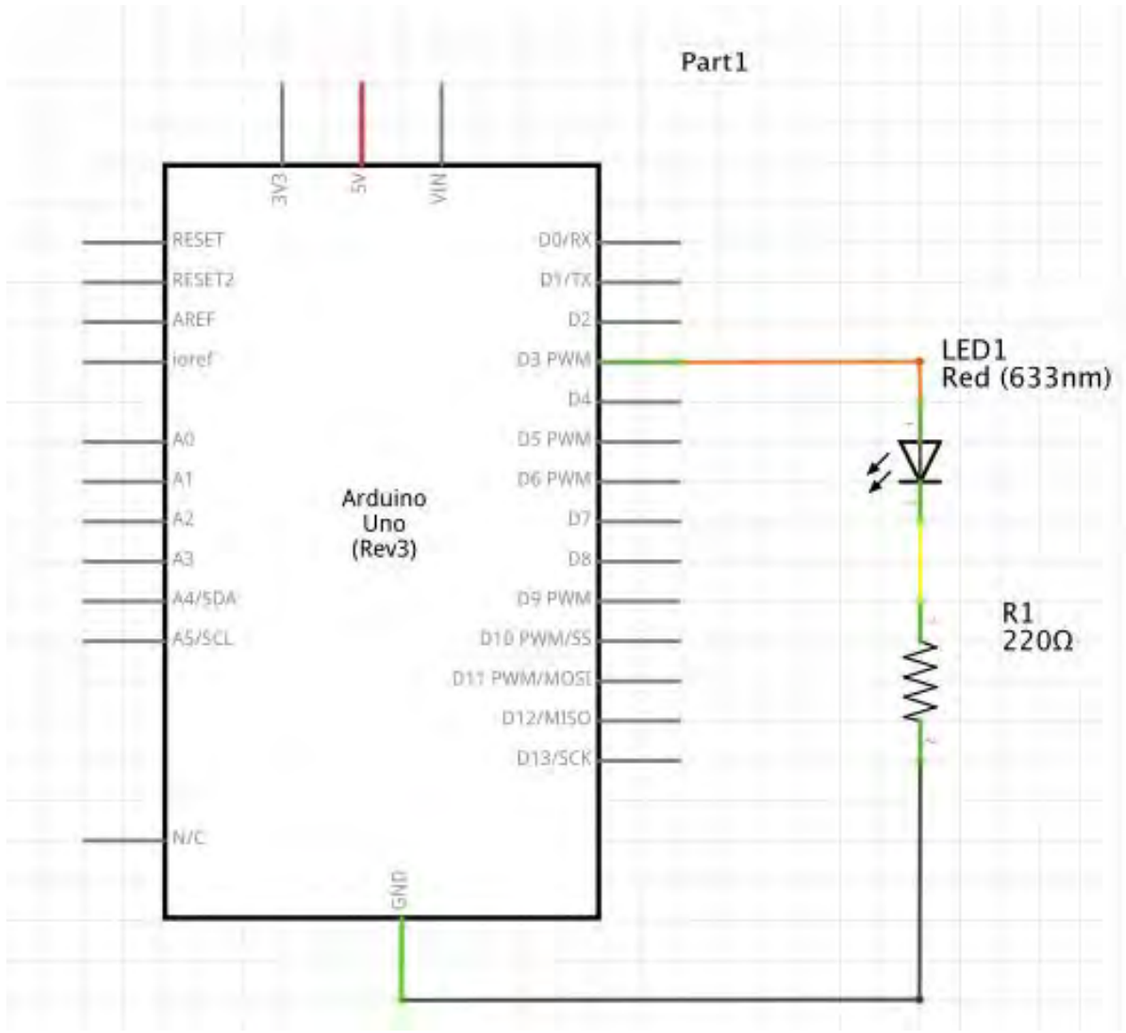


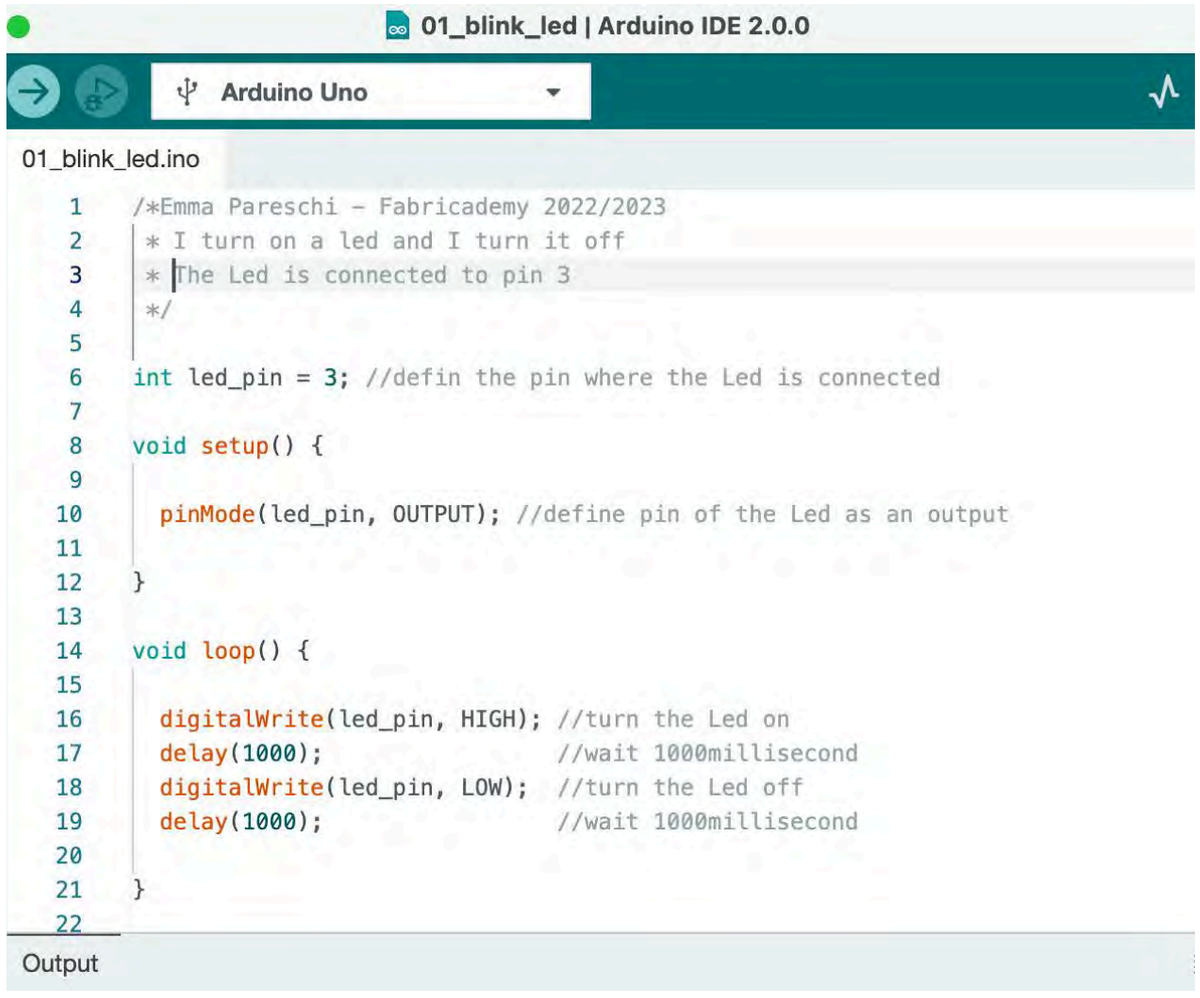
Connect the LED to pin D3

Control a Led with Arduino



Control a Led with Arduino



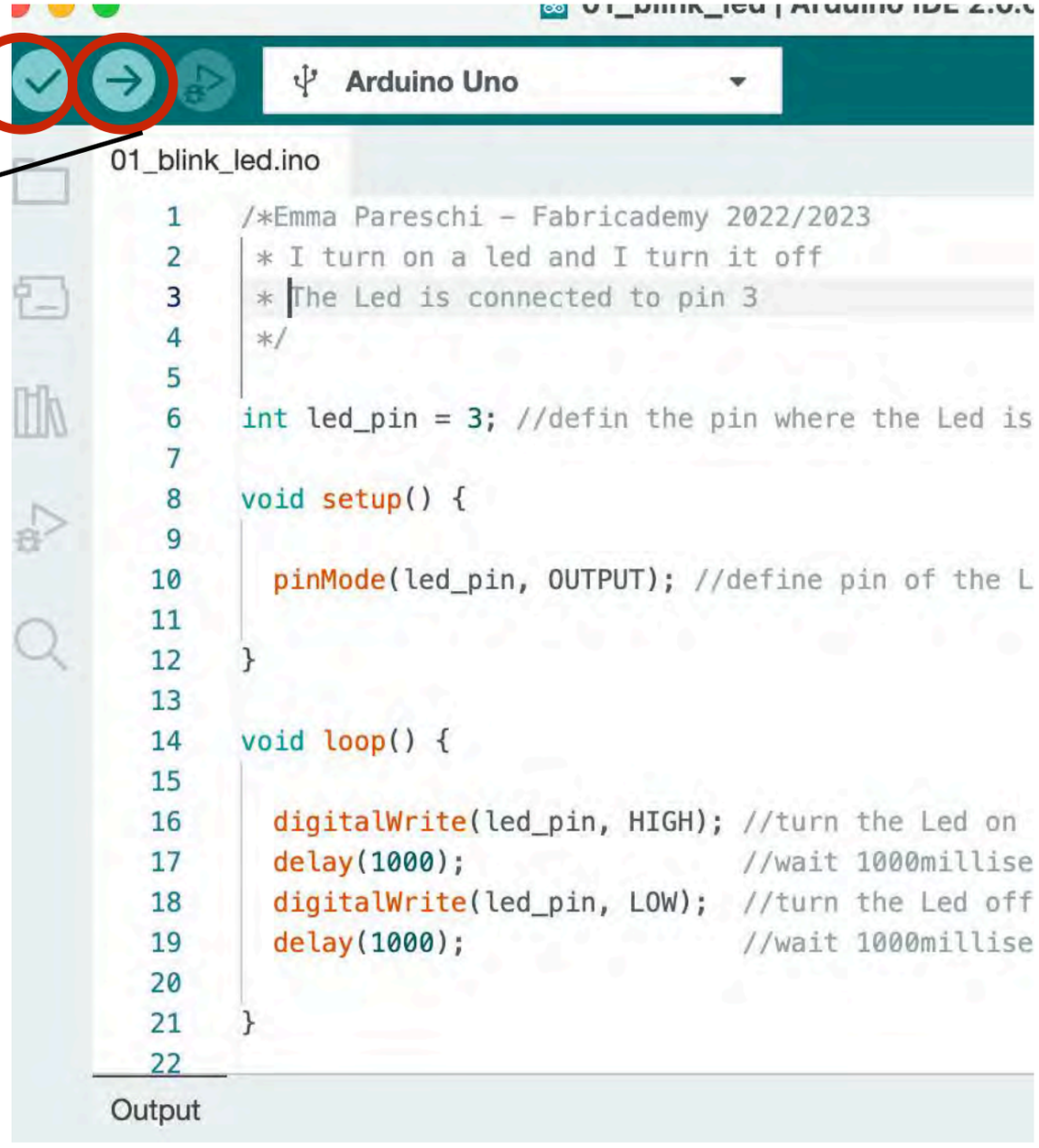


01_blink_led

Connect, Compile and Upload

1. Compile

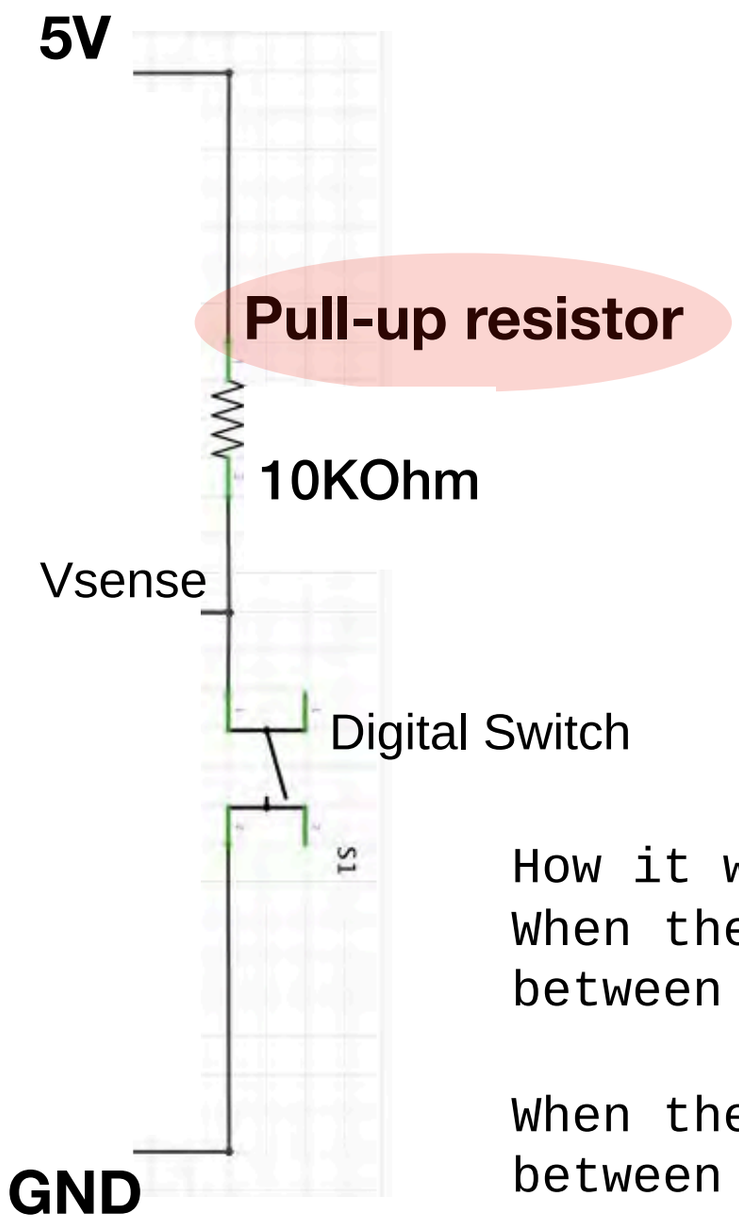
2. Upload



Arduino: Digital Sensors

Read a Digital Sensor and print the value on the Serial Monitor

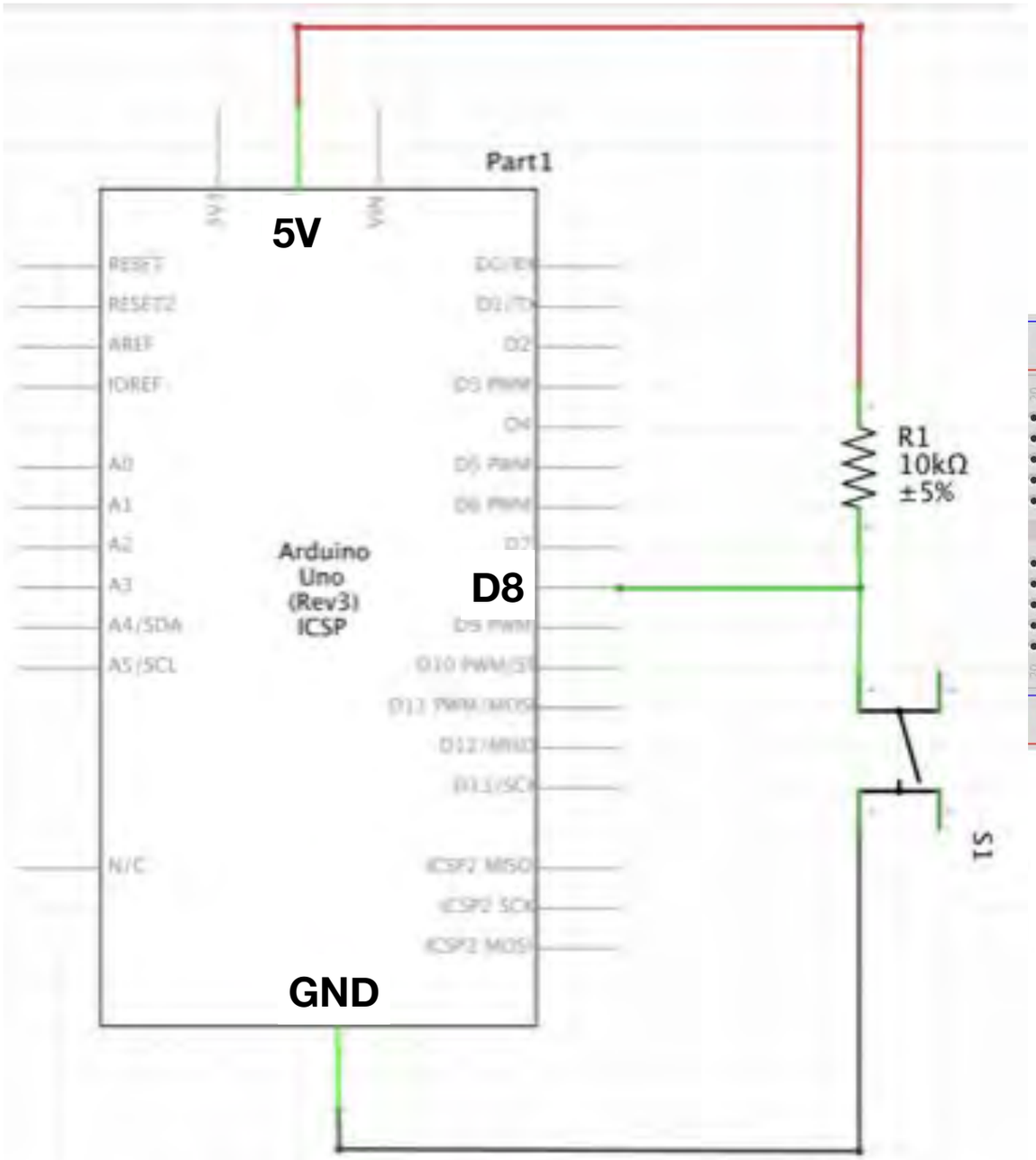
How the digital sensor changes the Voltage

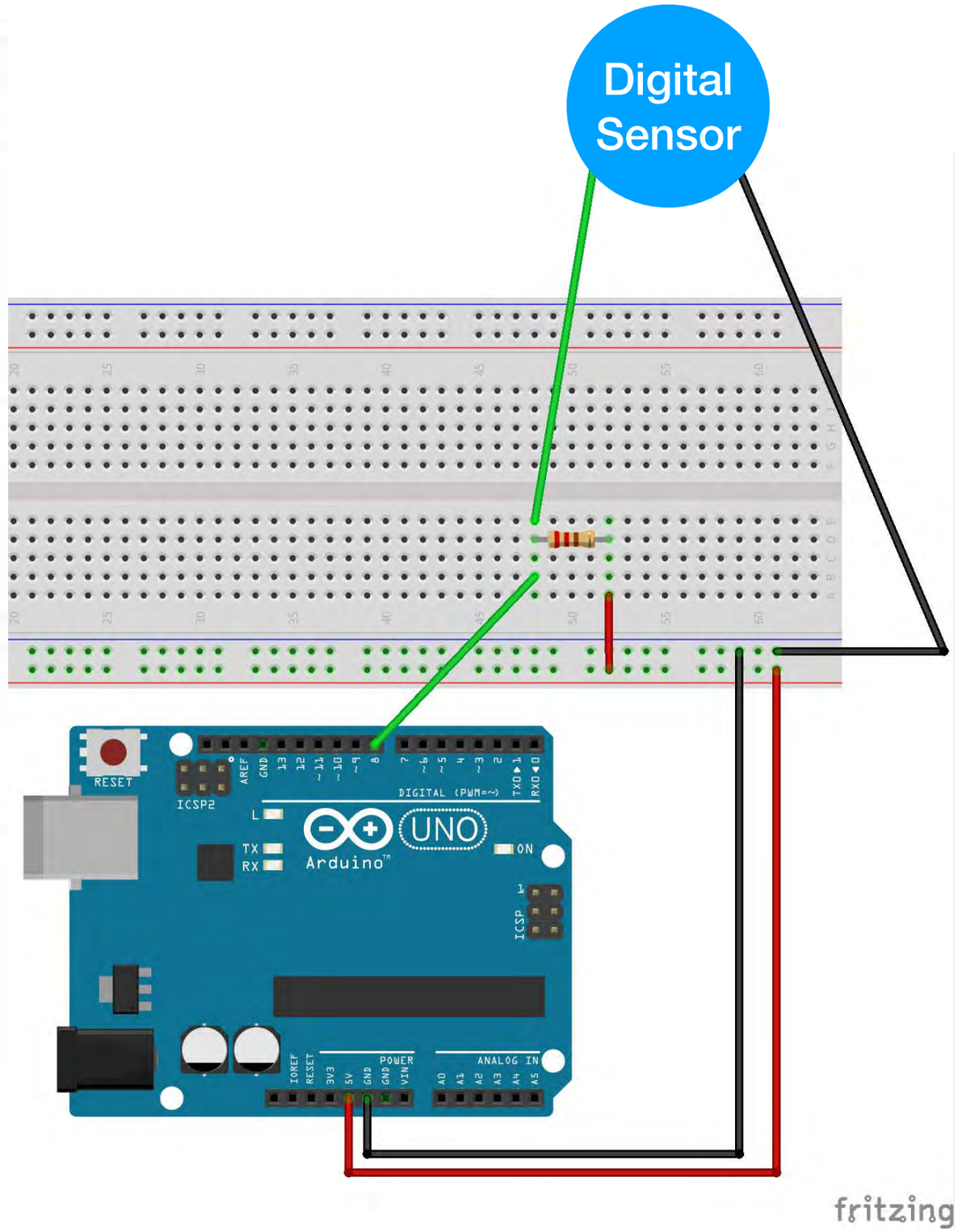


How it works:
When the switch is NOT pushed the voltage between Vsense and GND is 5 Volt

When the switch is NOT pushed the voltage between Vsense and GND is 0

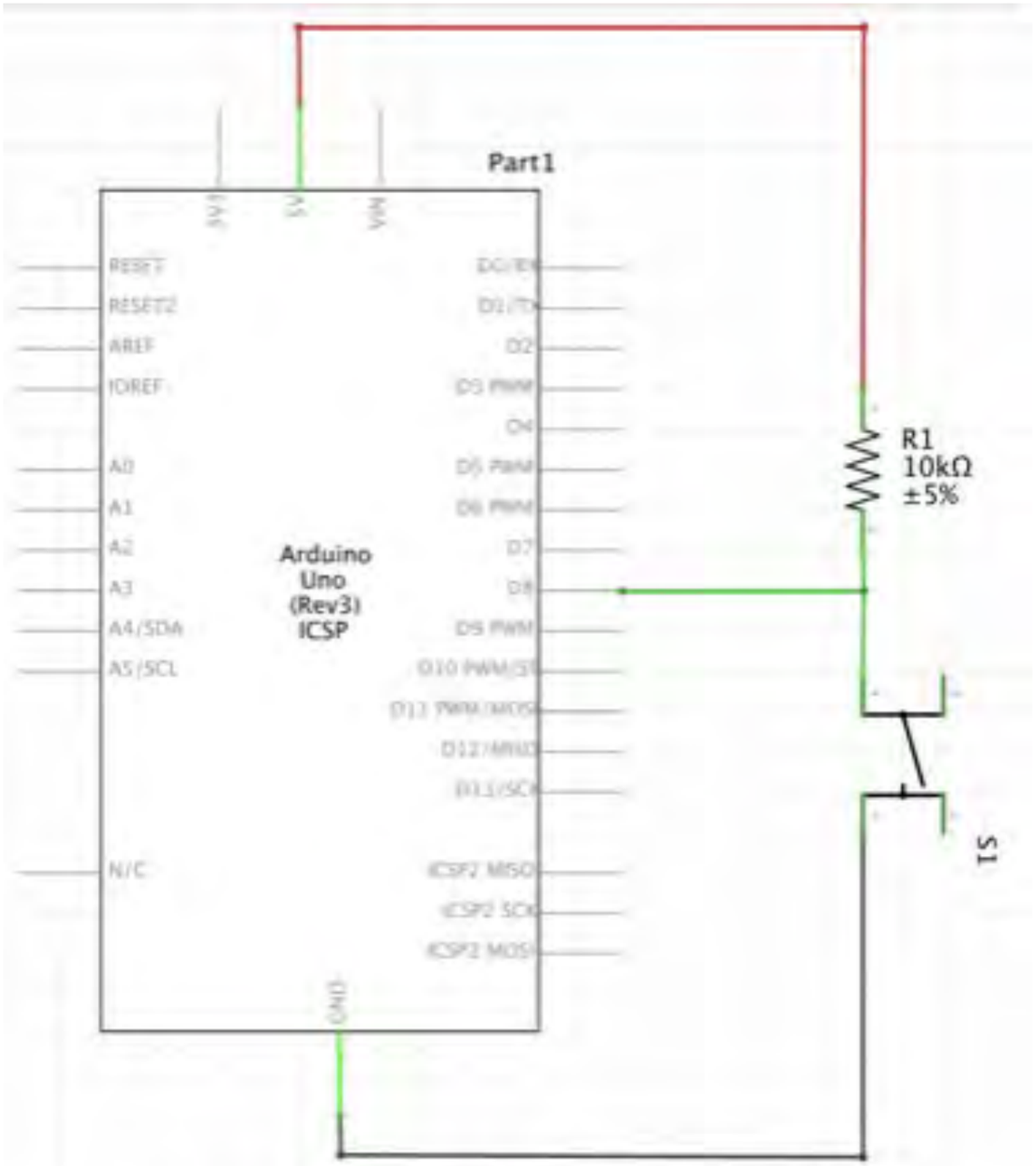
Digital switch and Arduino






```
pinMode(8, INPUT);
```

Digital switch and Arduino



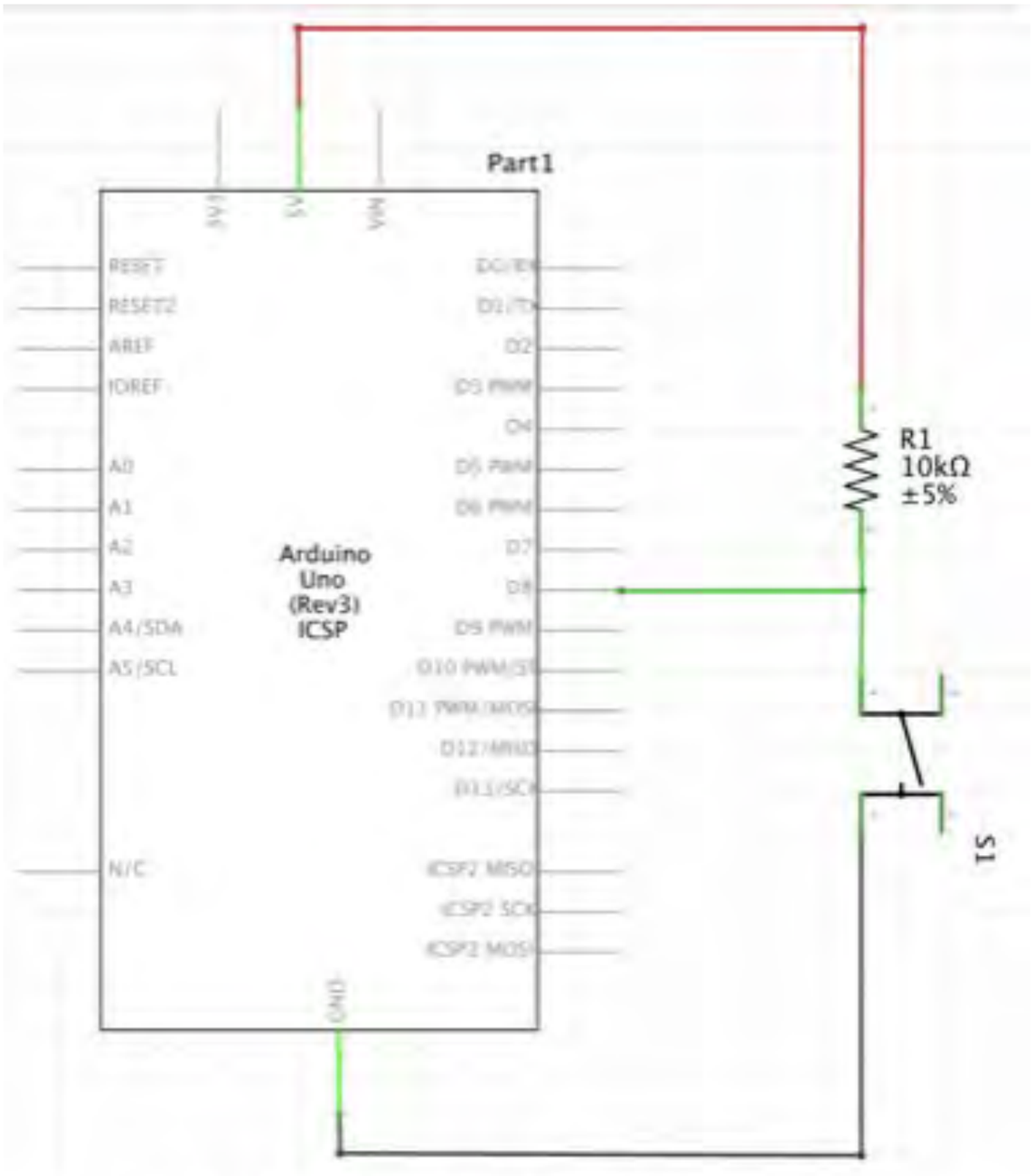
pinMode(8, INPUT);

03_read_DigitalSensor

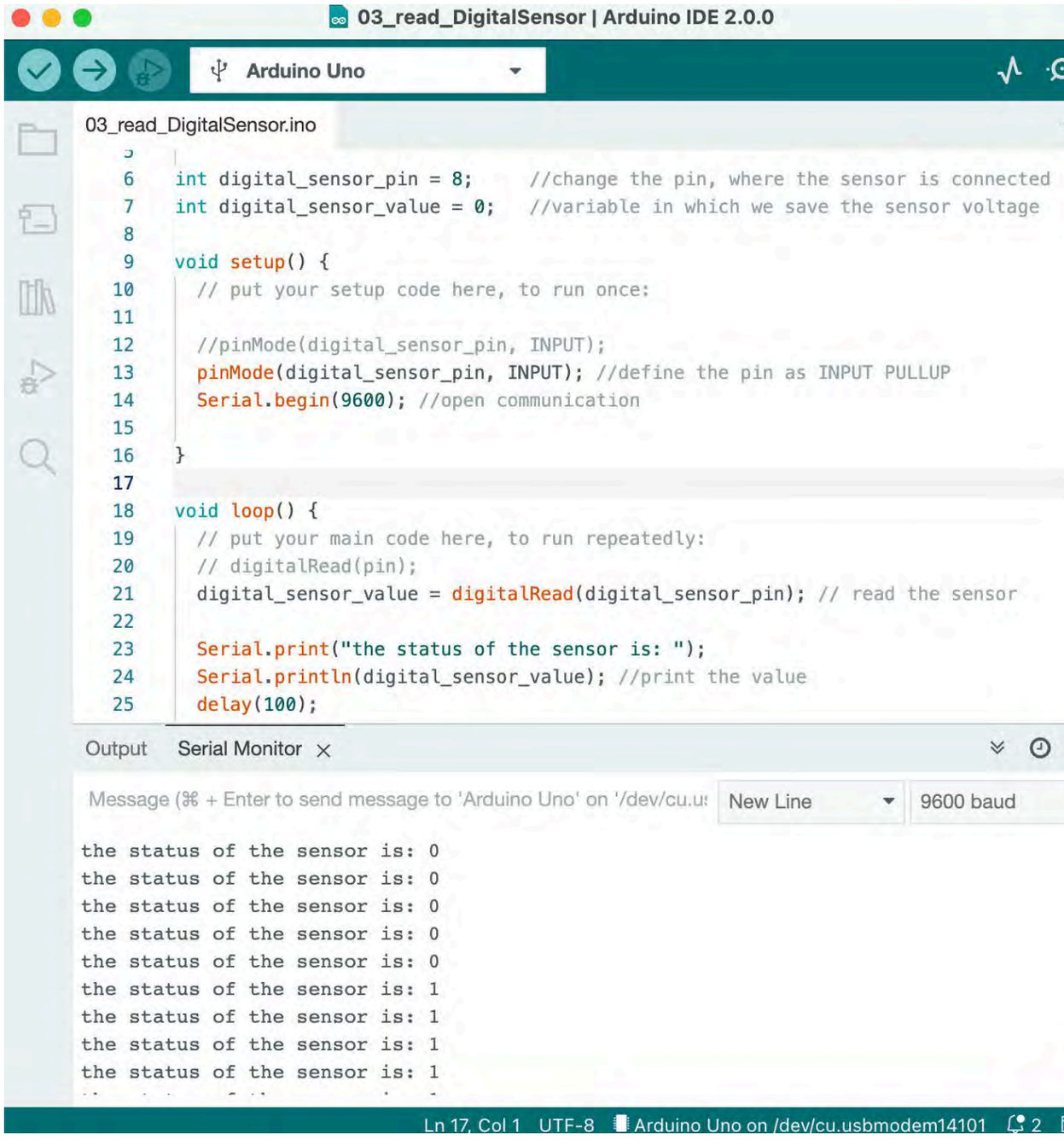
```
/*Emma Pareschi
 * we read the value of a digital sensor connected to pin digital_sensor_pin and
 * we print it on the Serial Monitor
 */

int digital_sensor_pin = 8;     //change the pin, where the sensor is connected?
int digital_sensor_value = 0;

void setup() {
  // put your setup code here, to run once:
  pinMode(digital_sensor_pin, INPUT); //define the pin as INPUT
  Serial.begin(9600);

}

void loop() {
  // put your main code here, to run repeatedly:
  digital_sensor_value = digitalRead(digital_sensor_pin); // read the sensor

  Serial.println(digital_sensor_value); //print the value
  delay(100);

}
```

03_read_DigitalSensor

Digital switch and Arduino - Serial Monitor

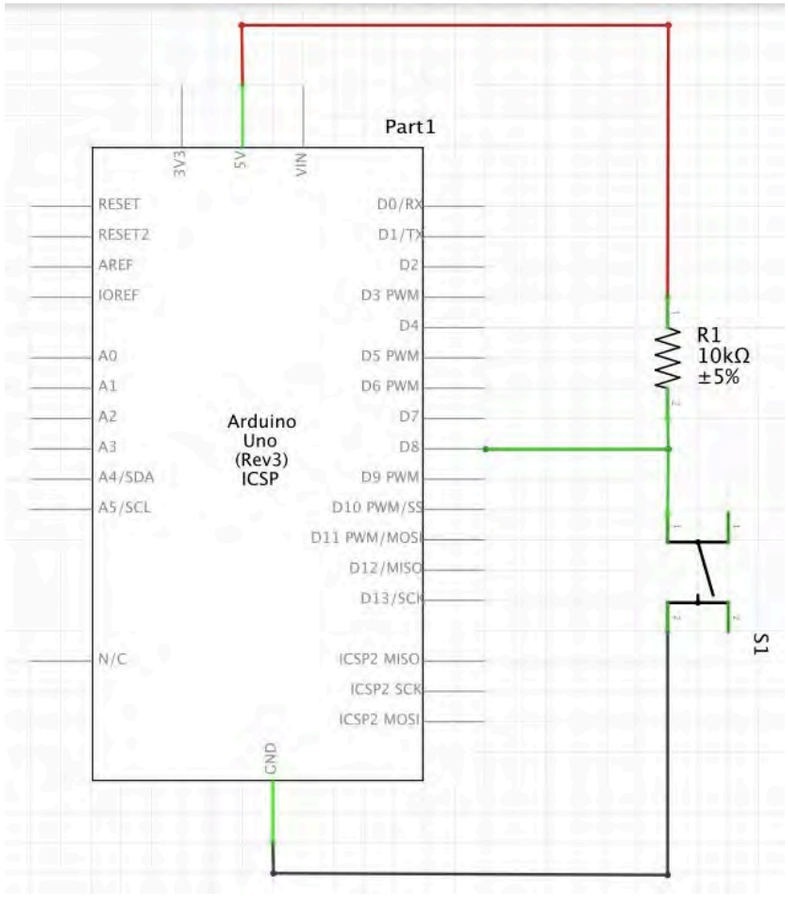


pinMode(8, INPUT);

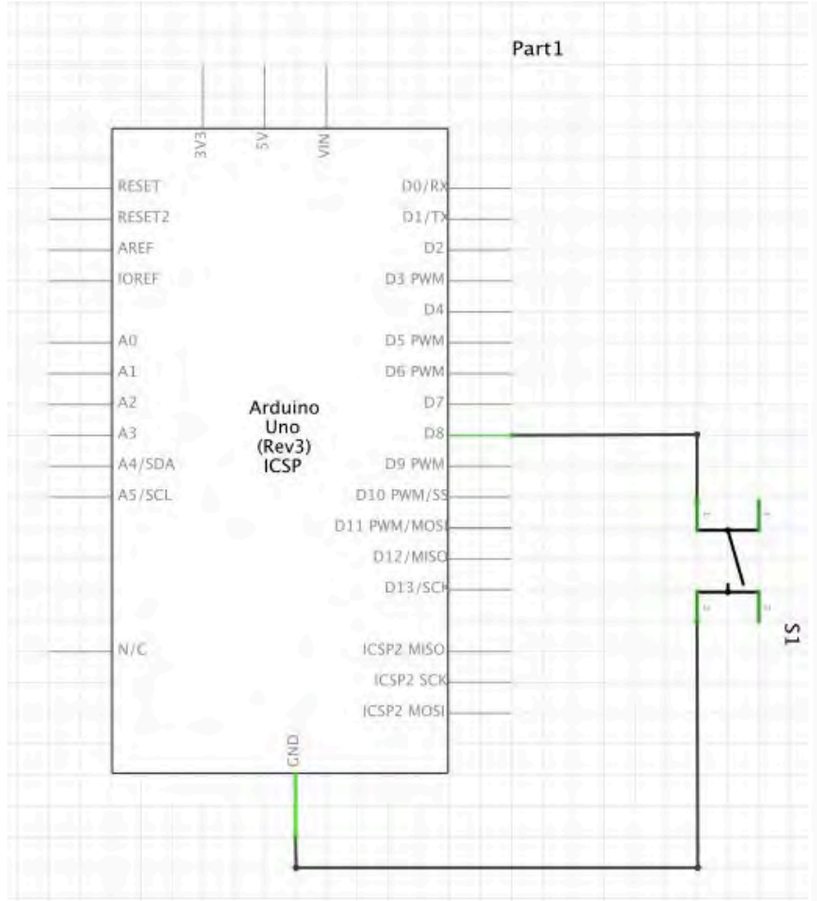


03_read_DigitalSensor

Digital switch and Arduino



pinMode(8, INPUT);

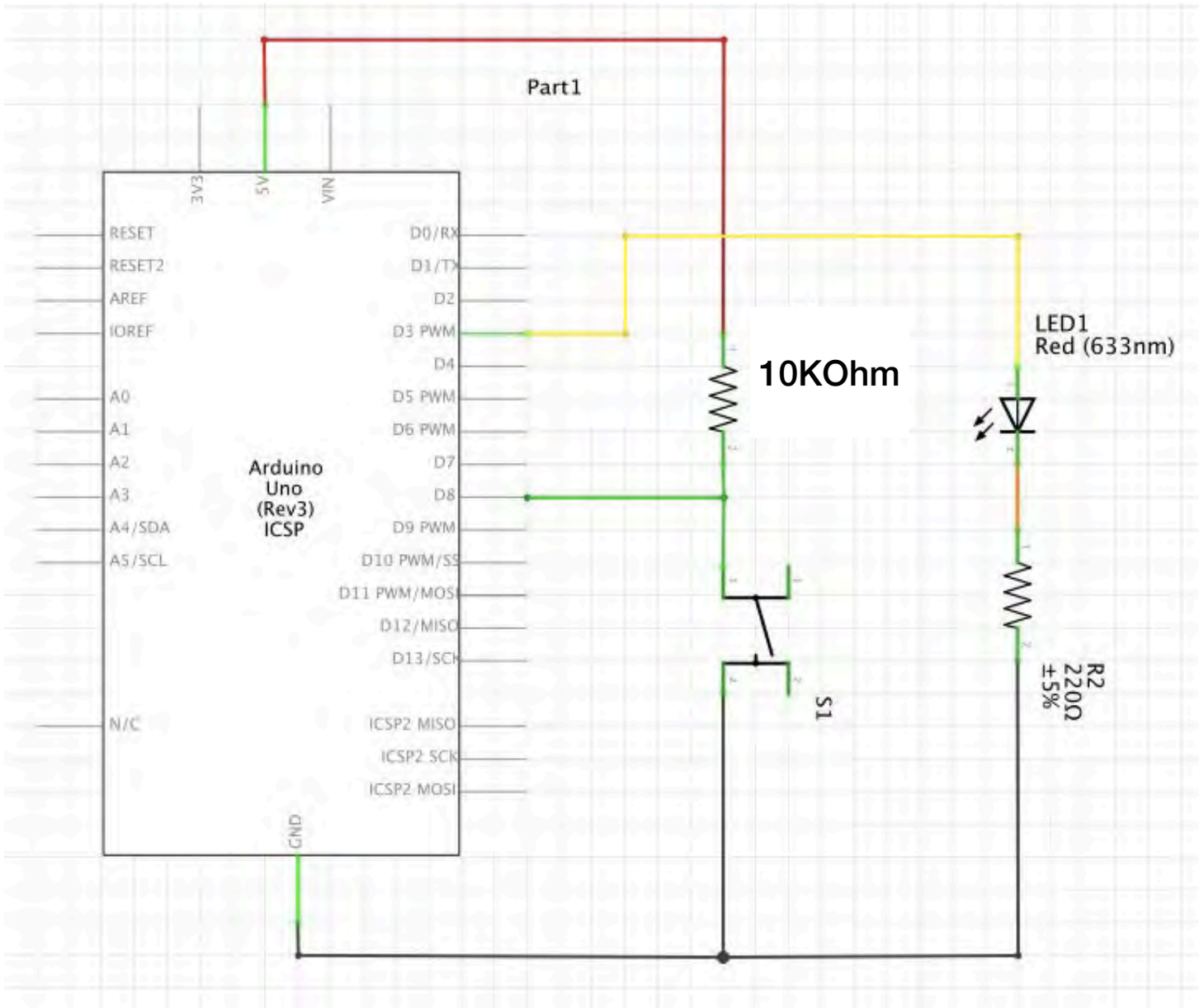



```
pinMode(8, INPUT_PULLUP);
```

Arduino: Digital Sensors - Led

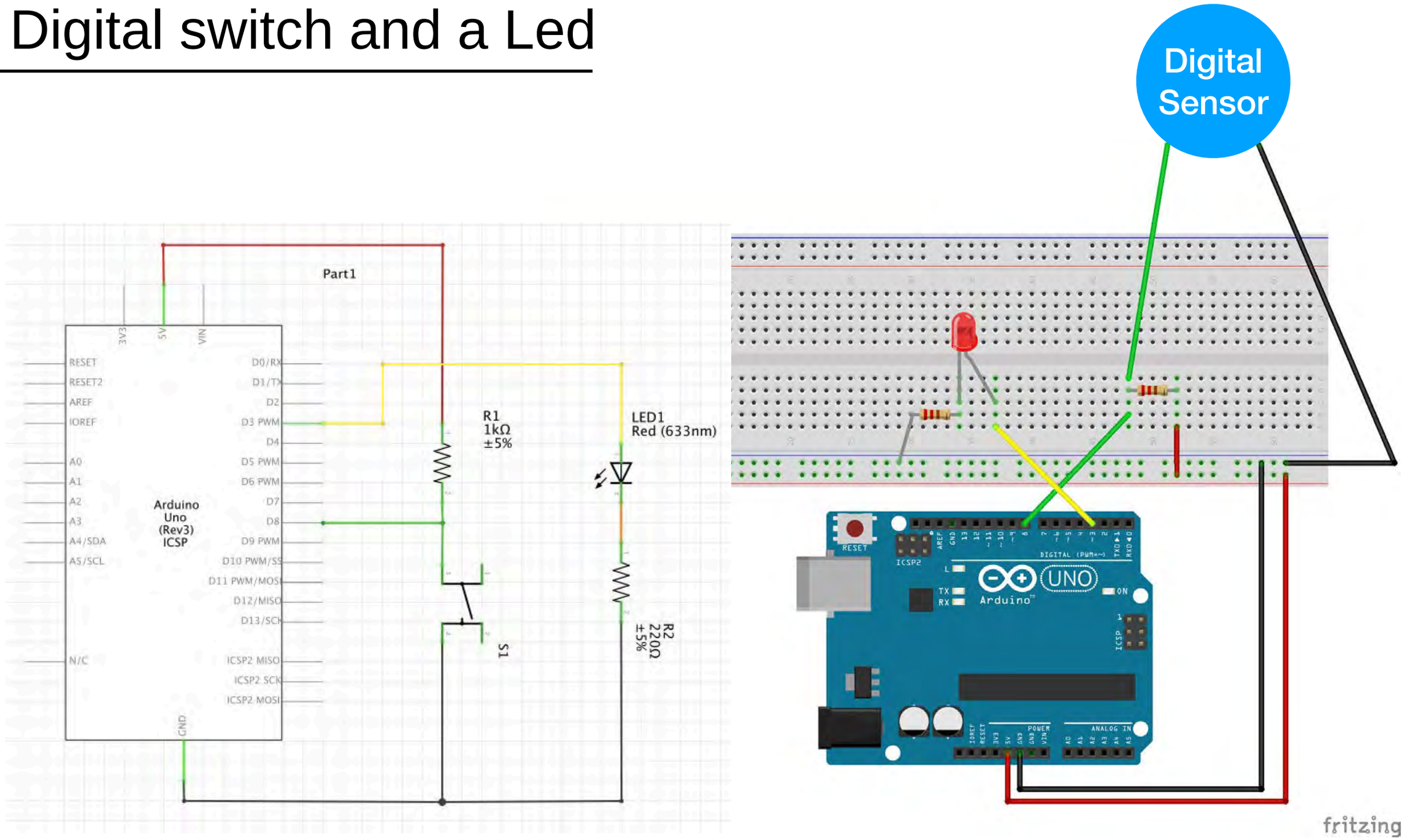
Control a Led with a Digital Sensor

Digital switch and a Led



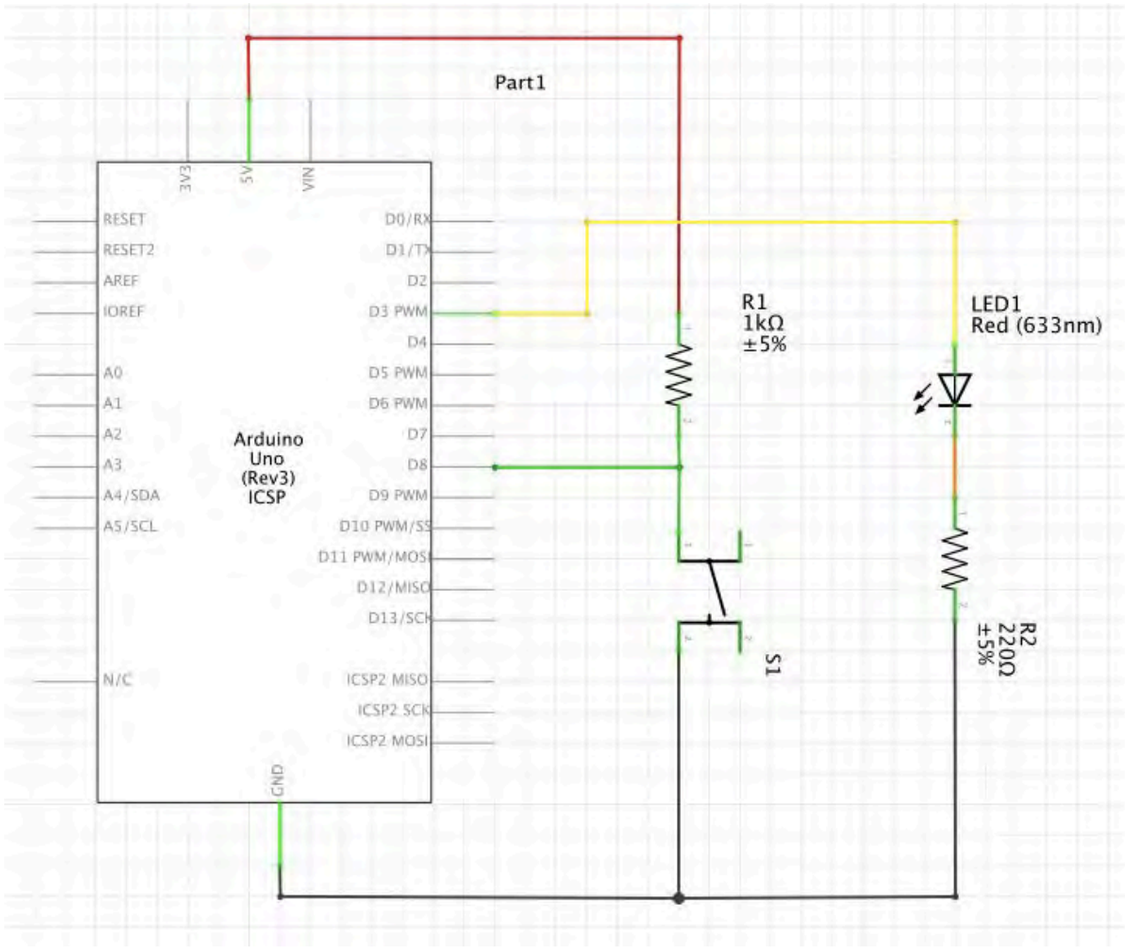
You already have the circuit of the sensor, you have to add the Led!

Digital switch and a Led



You already have the circuit of the sensor, you have to add the Led!

Digital switch and a Led



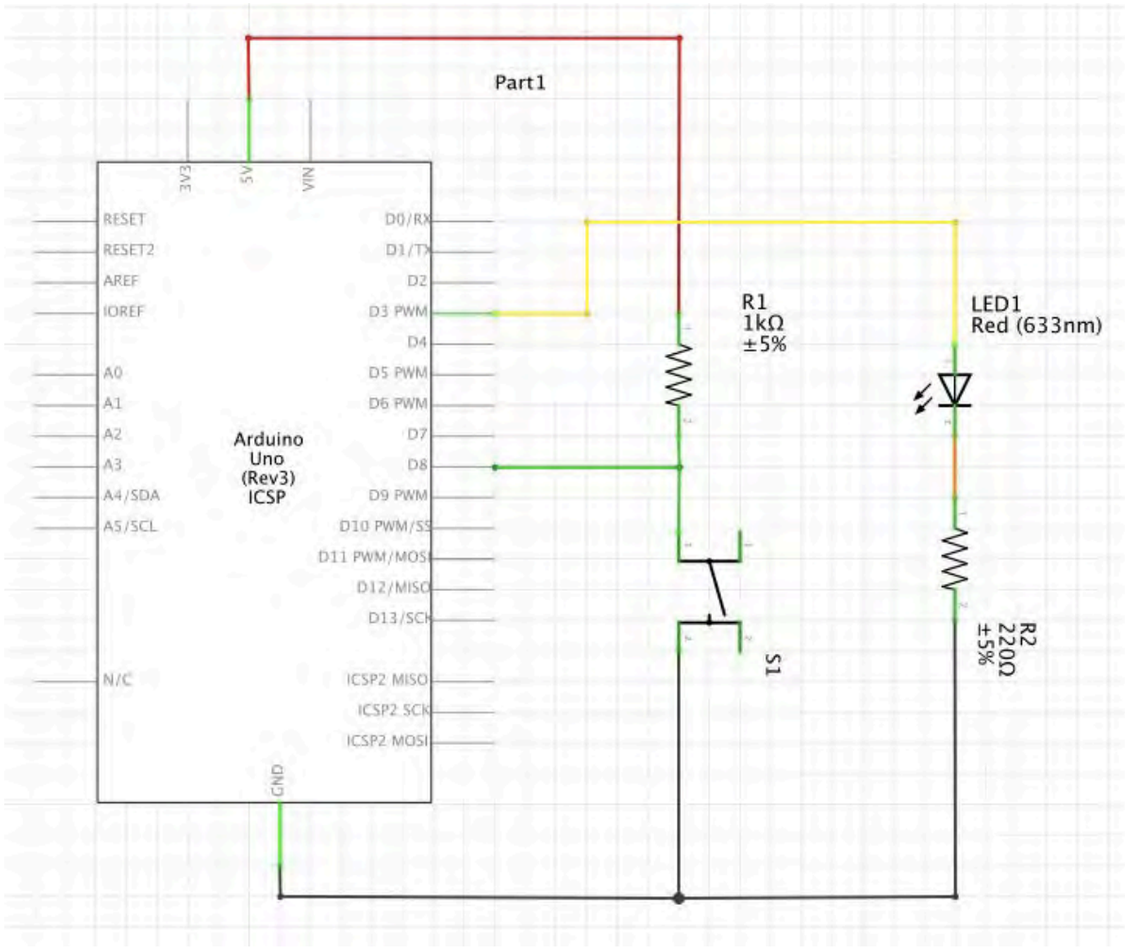
03_button_led

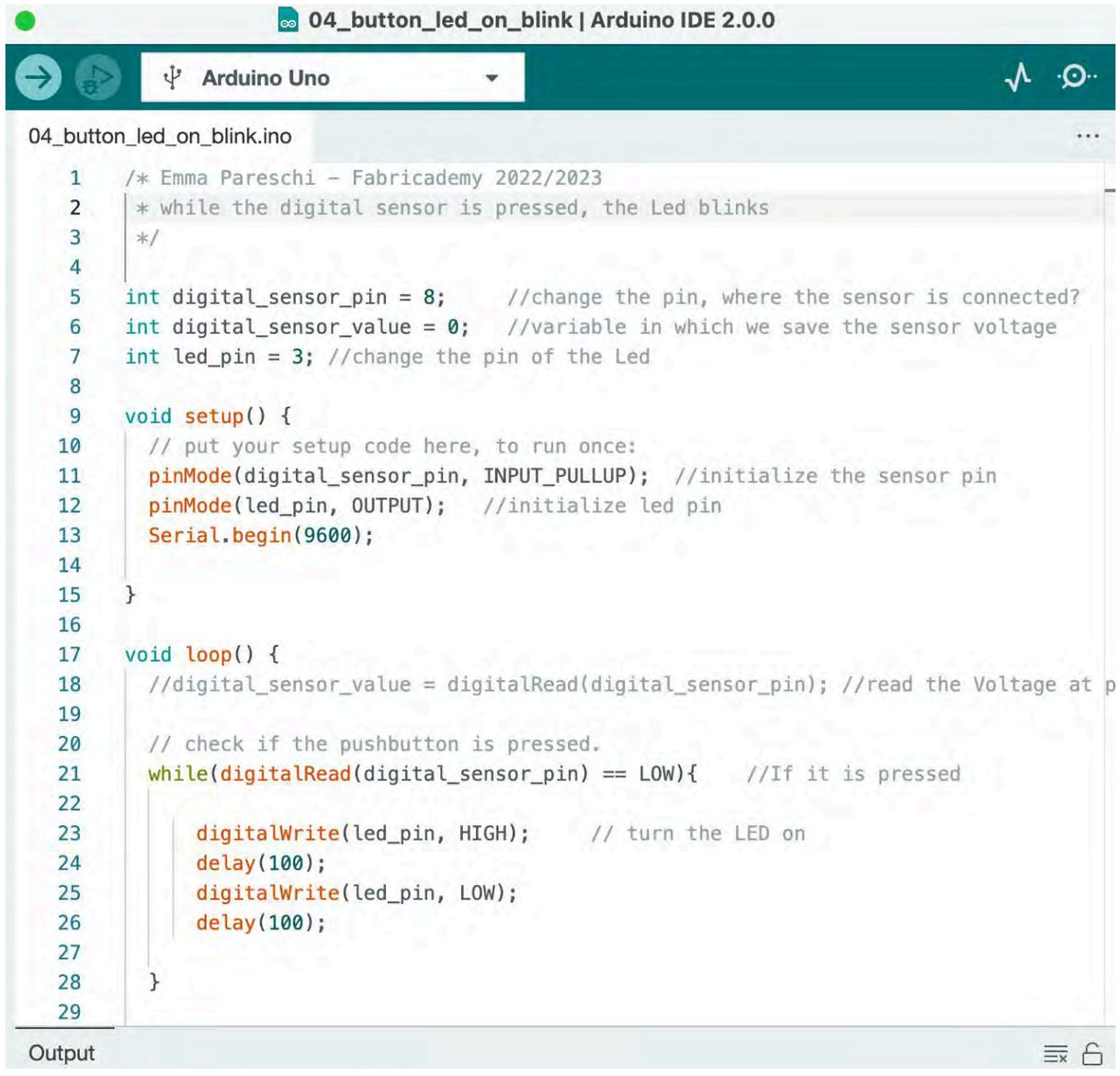
```
/*Emma Pareschi,
 * this skecth is a modification of the example button!!
 */

int digital_sensor_pin = 7;     //change the pin, where the sensor is connected?
int digital_sensor_value = 0;
int led_pin = 3; //change the pin of the Led

void setup() {
  // put your setup code here, to run once:
  pinMode(digital_sensor_pin, INPUT);
  Serial.begin(9600);
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(led_pin, OUTPUT);

}

void loop() {
  // put your main code here, to run repeatedly:
  digital_sensor_value = digitalRead(digital_sensor_pin);

  // check if the pushbutton is pressed. If it is, the buttonState is HIGH:
  if(digital_sensor_value == HIGH){
        // turn LED on:
      digitalWrite(led_pin, HIGH);
  } else {
        // turn LED off:
      digitalWrite(led_pin, LOW);    // turn the LED off by making the voltage LOW

  }

}
```

04_button_led_on_off

From Example -> Digital -> Button

Digital switch and a Led



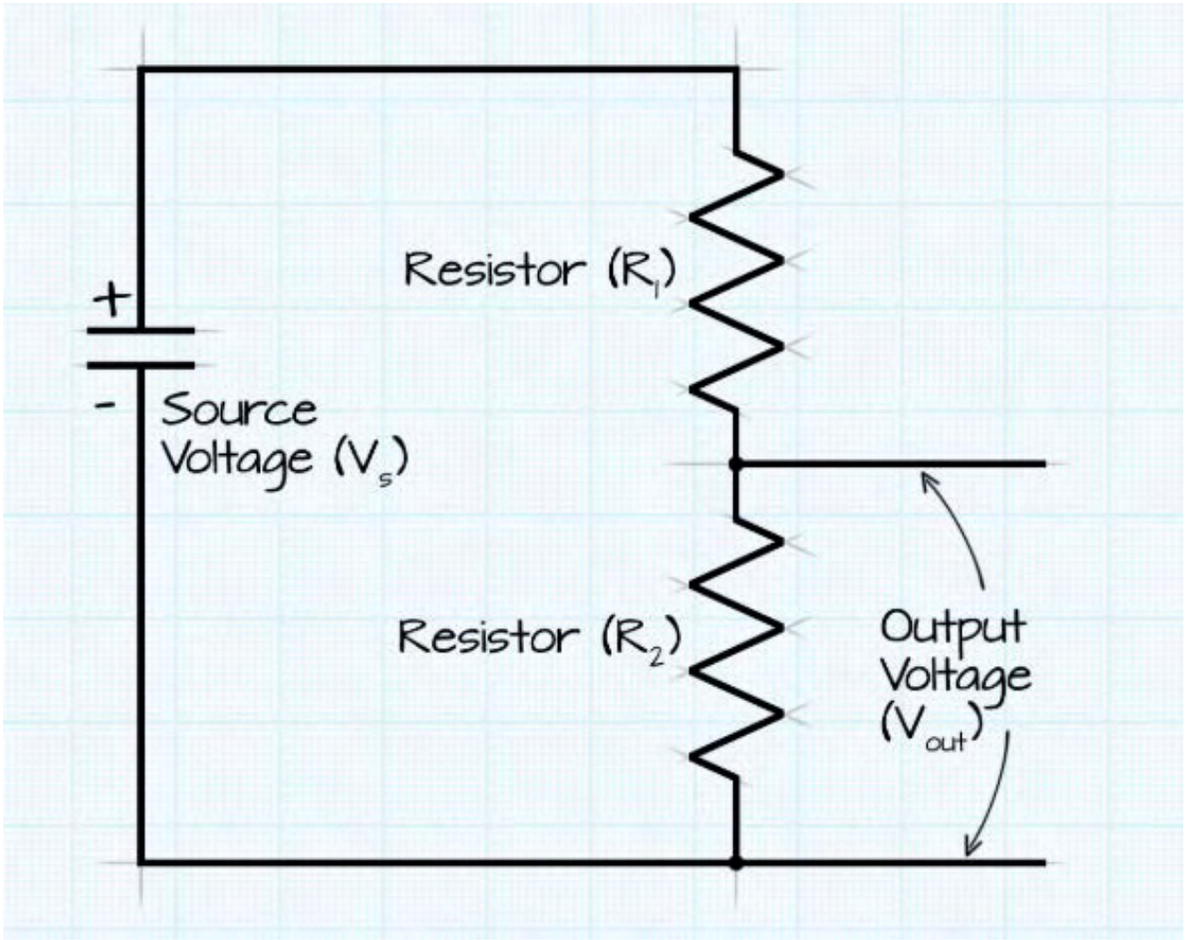


04_button_led_blink

Arduino: Analog Sensor

Read and print the value of an Analog Sensor connected to Arduino

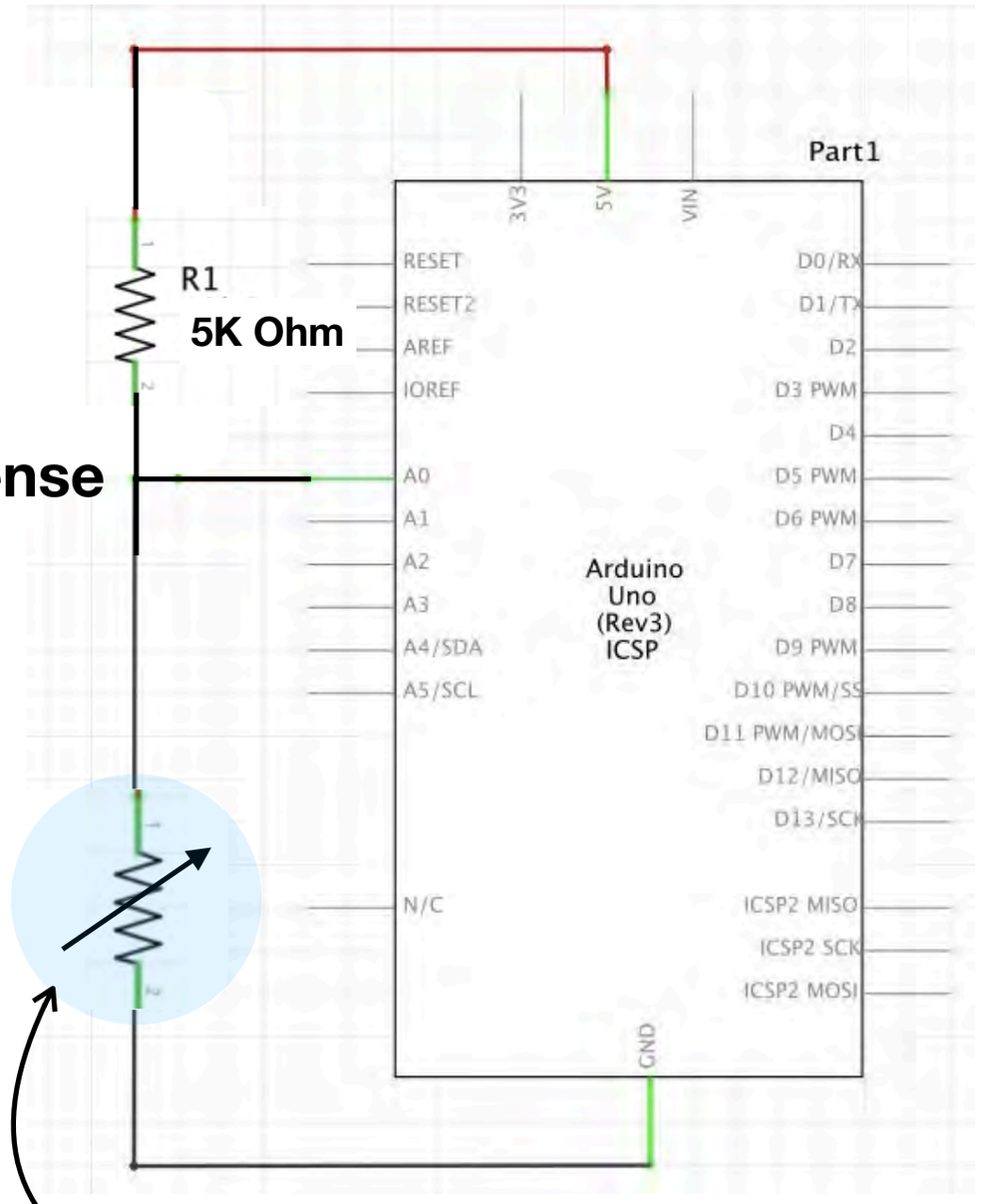
Voltage divider (analog sensor)



$$V_{out} = \frac{V_s \times R_2}{(R_1 + R_2)}$$

http://www.ohmslawcalculator.com/voltage-divider-calculator

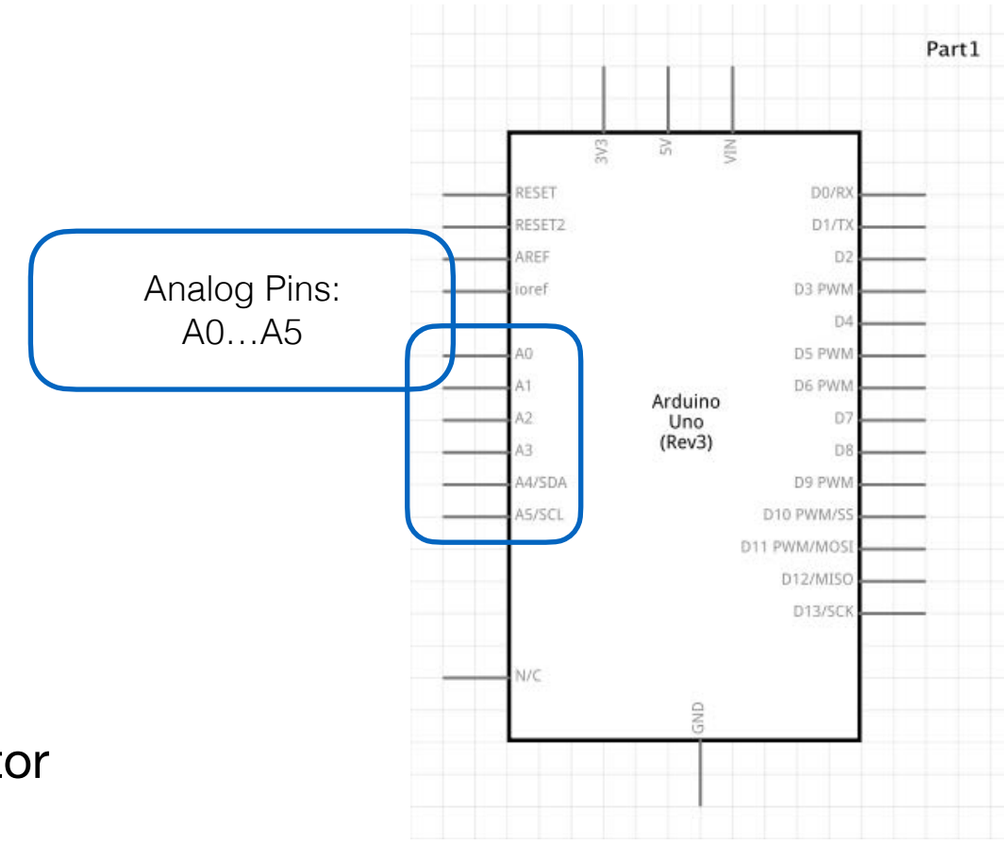
Analog sensor - schematic



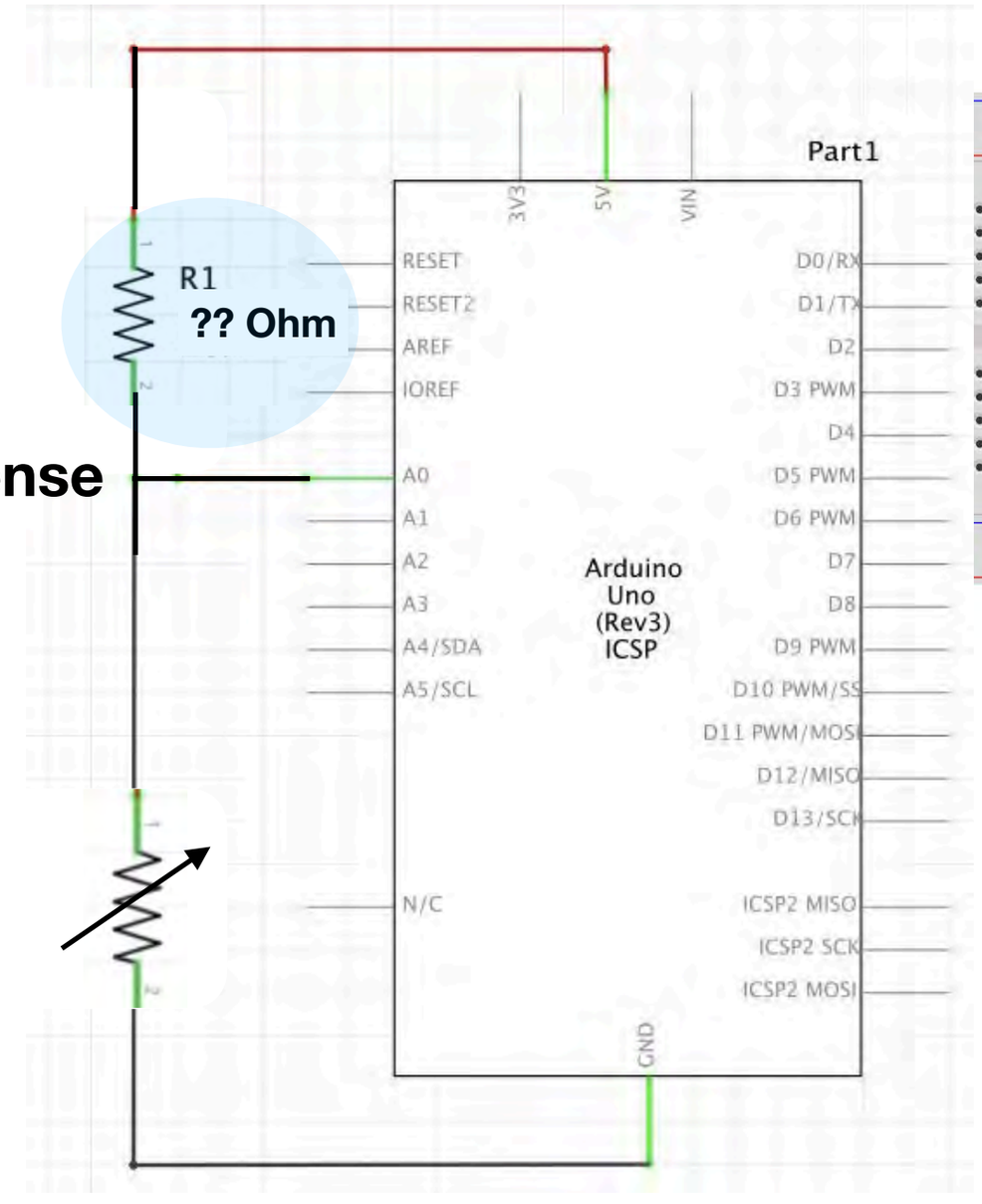
This is the symbol of a Variable Resistor

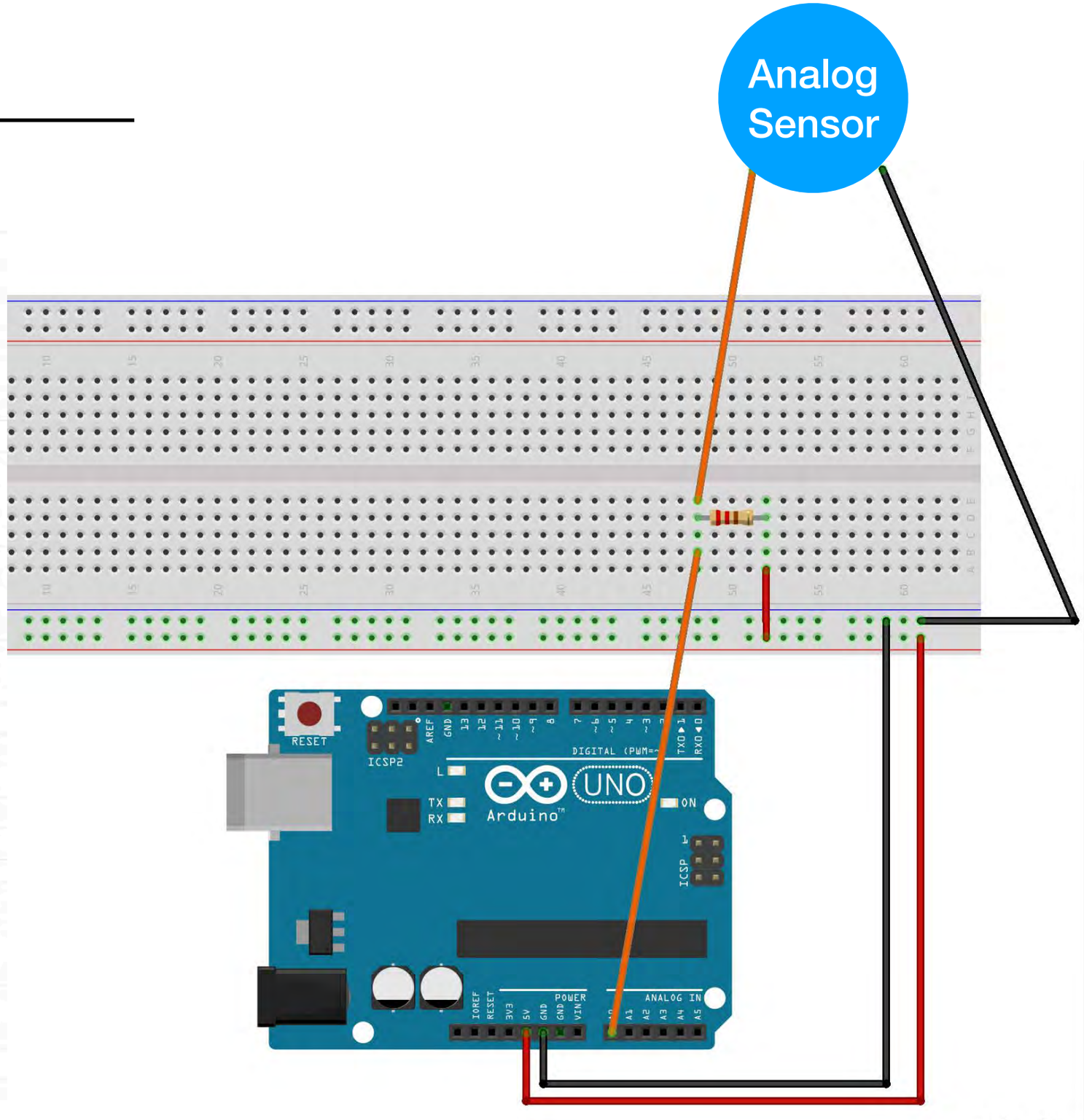
An Analog sensor is a variable resistor!! Change the resistance and you change the Voltage Vsense….the Arduino will read this Voltage.

To read an Analog sensor that generates an analog voltage => you need to use the ANALOG PINS.



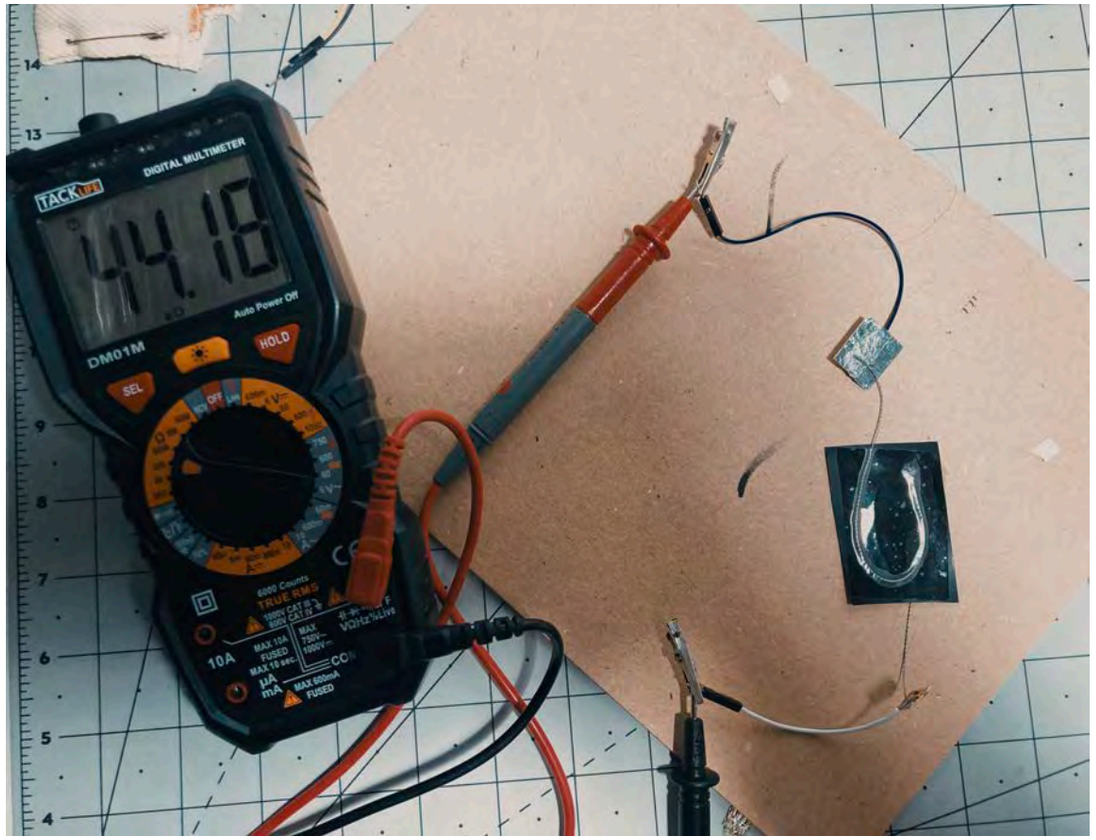
Analog sensor

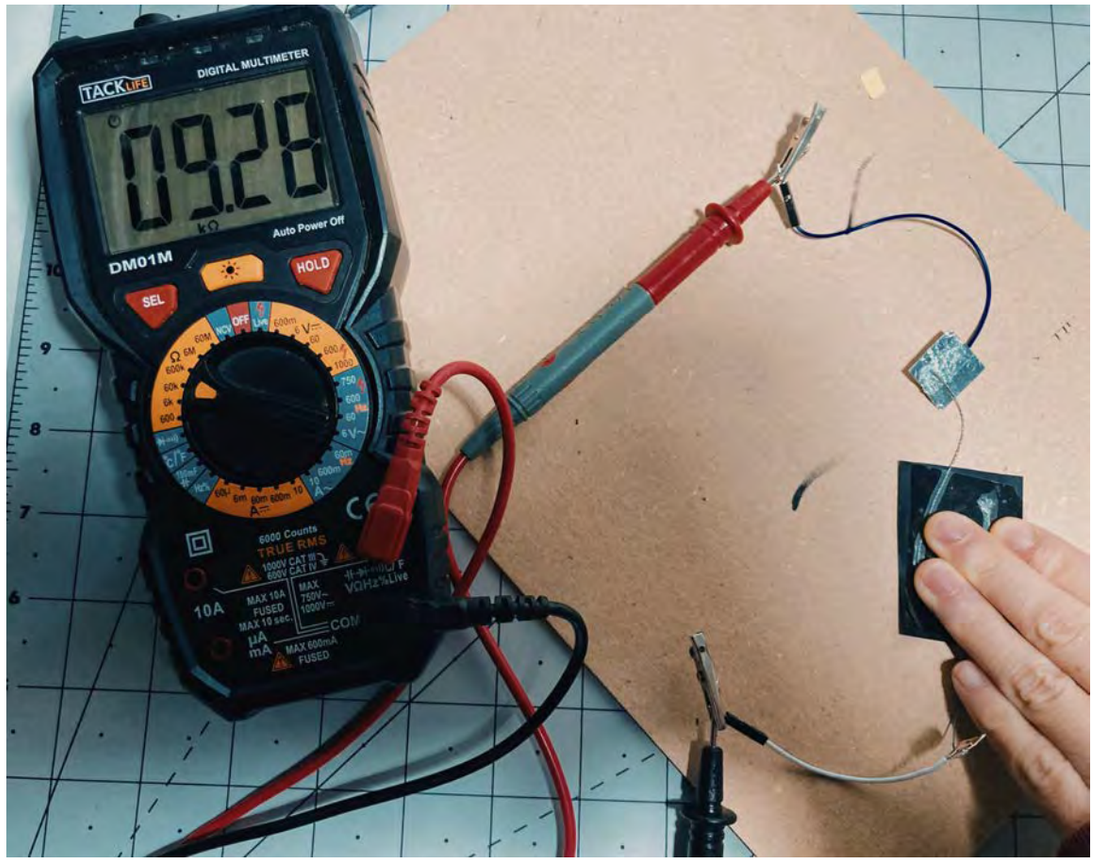




How to select the right resistor?

Analog sensor - voltage divider



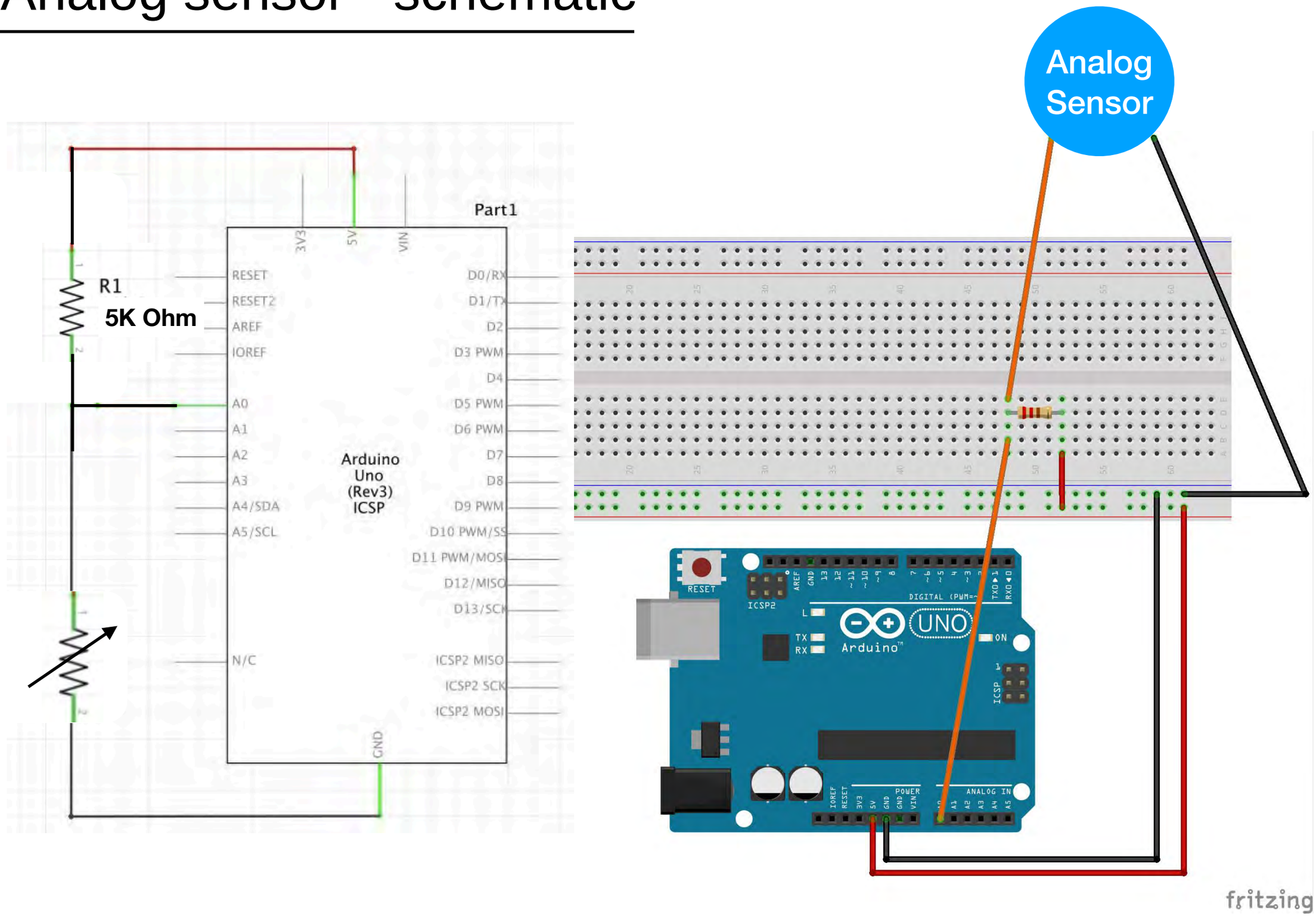


- With the multimeter measure the resistance on the sensor while you interact with it.
- Take note of the min and max resistance values
- Calculate the average of the min and max.
- The average is the optimal value of the extra resistor to use with the analog sensor.

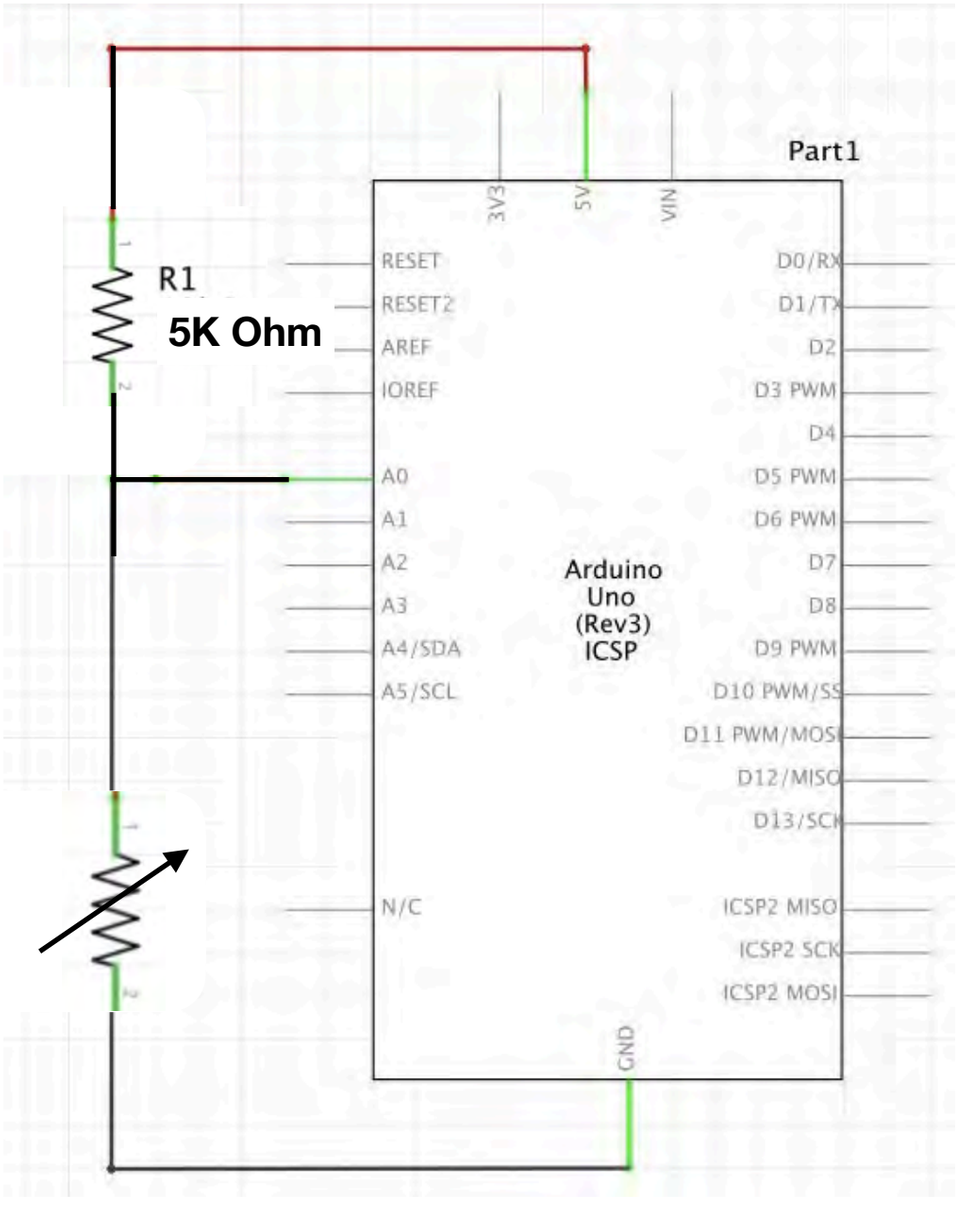
In this case, the resistor should be around 27 kOhm.

You can also just try with a 5k or 10k Ohm.

Analog sensor - schematic



Read an Analog sensor

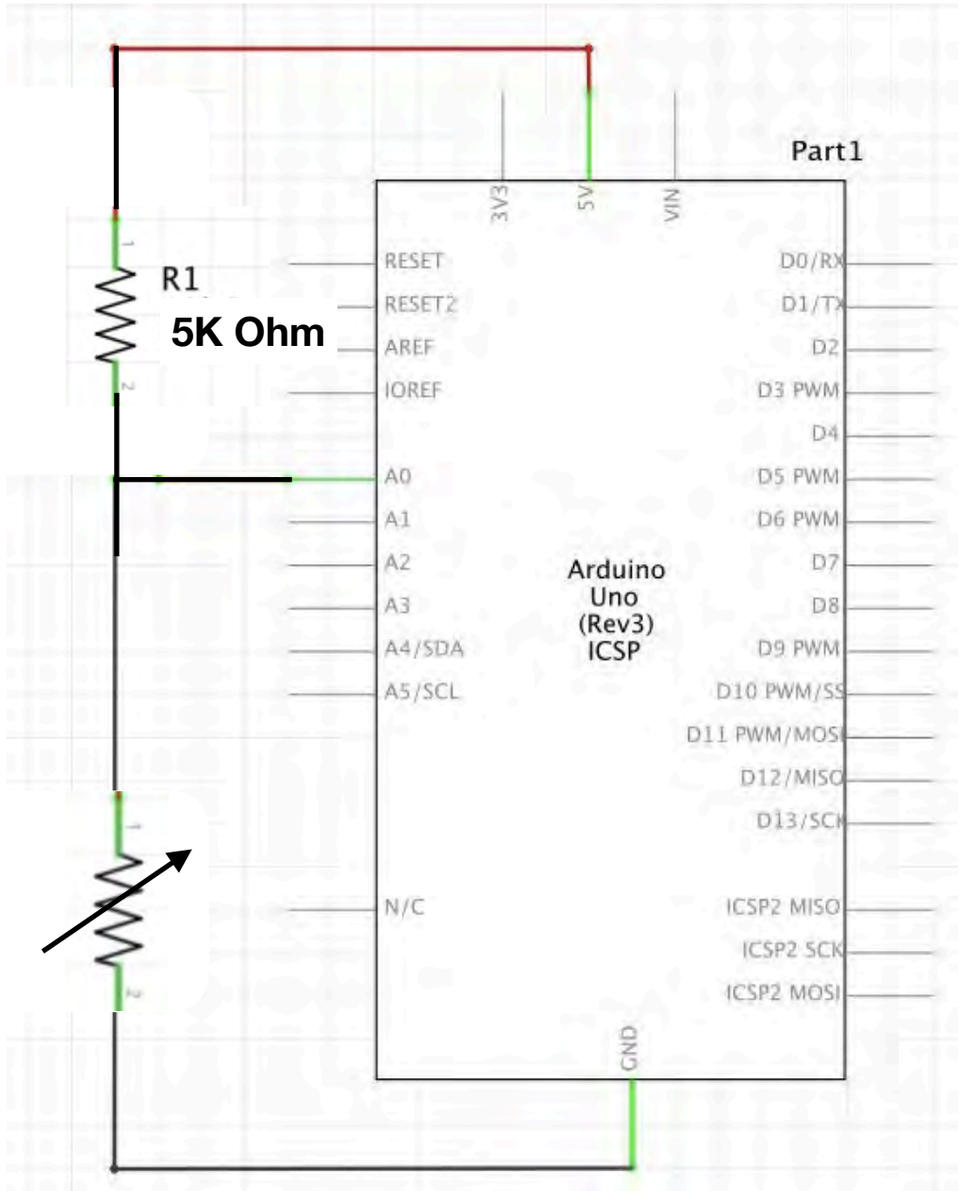


05_read_AnalogSensor.ino

```ino
/* Emma Pareschi - Fabricademy 2022/2023
 * with this sketch we read the analog sensor connected to pin analog
 */

int analog_sensor_pin = A0;   //change the pin, where the sensor is c
int analog_sensor_value = 0;

void setup() {
  // put your setup code here, to run once:
  pinMode(analog_sensor_pin, INPUT);
  Serial.begin(9600);

}

void loop() {
  // put your main code here, to run repeatedly:
  analog_sensor_value = analogRead(analog_sensor_pin); //read the Vol

  Serial.println(analog_sensor_value); // print the value on the Seri
  delay(100);

}

```

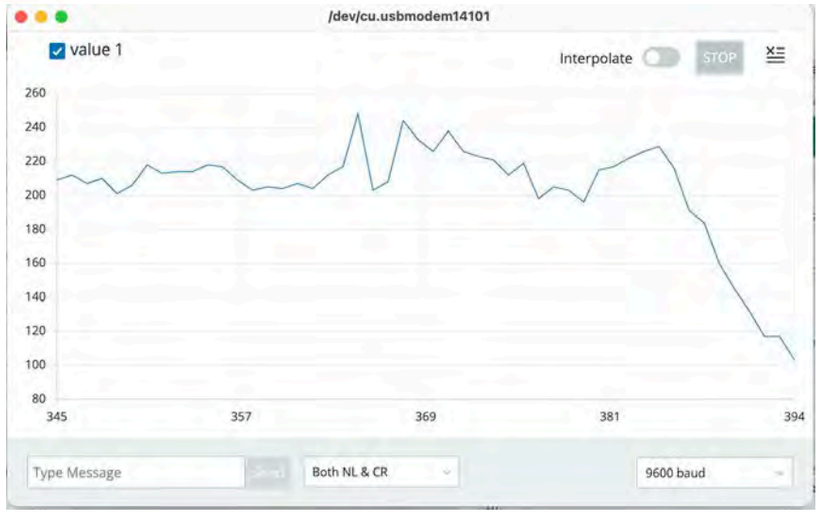
05_read_AnalogSensor

Read an Analog sensor



05_read_AnalogSensor.ino

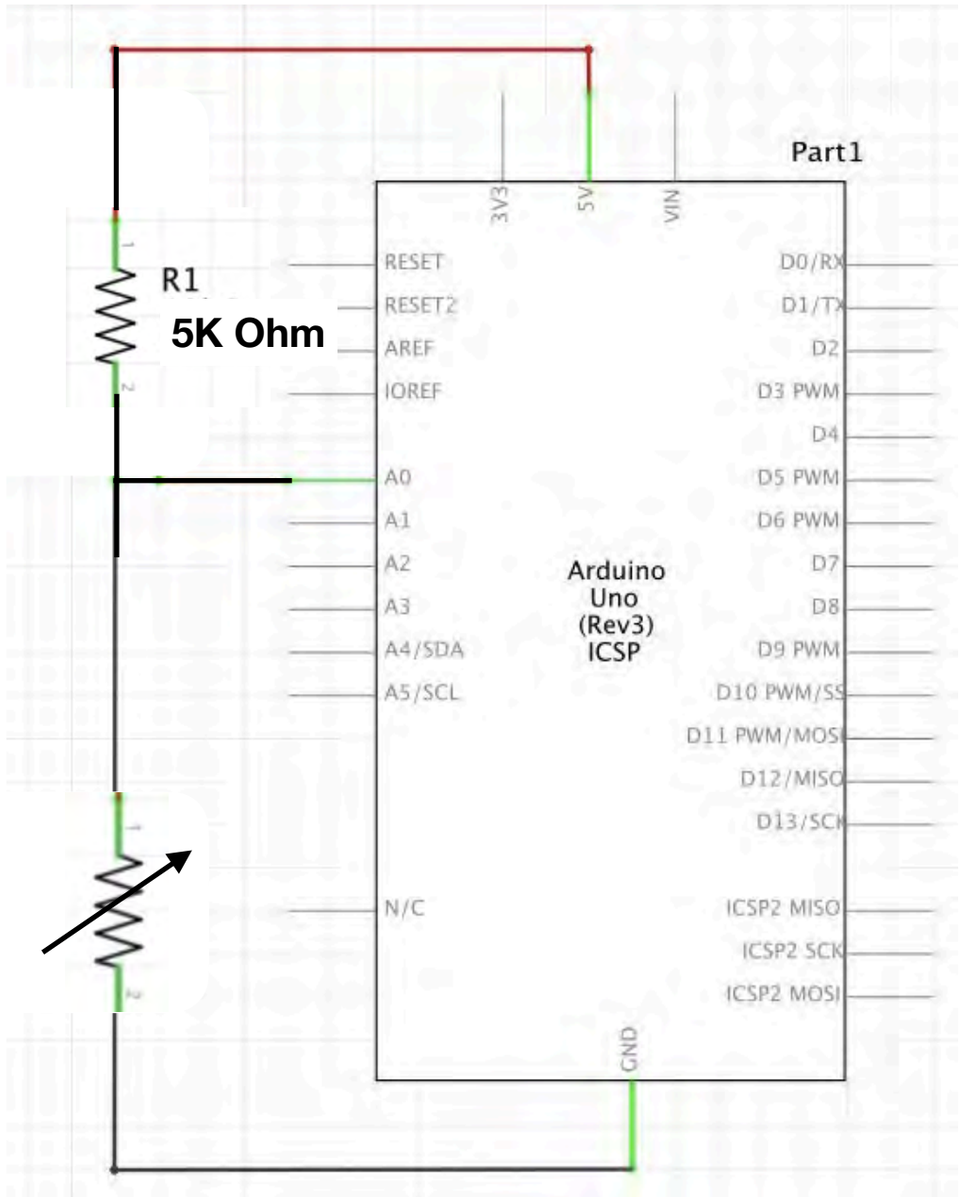
```ino
/* Emma Pareschi - Fabricademy 2022/2023
 * with this sketch we read the analog sensor connected to pin analog_sensor_pin
 */

int analog_sensor_pin = A0;   //change the pin, where the sensor is connected?
int analog_sensor_value = 0;

void setup() {
  // put your setup code here, to run once:
  pinMode(analog_sensor_pin, INPUT);
  Serial.begin(9600);

}

void loop() {
  // put your main code here, to run repeatedly:
  analog_sensor_value = analogRead(analog_sensor_pin); //read the Voltage of the pin sensor

  Serial.println(analog_sensor_value); // print the value on the Serial monitor
  delay(100);

}
```



05_read_AnalogSensor

Write down the minimum and the maximum.
We will use them in the next demo.

Read and map an Analog sensor



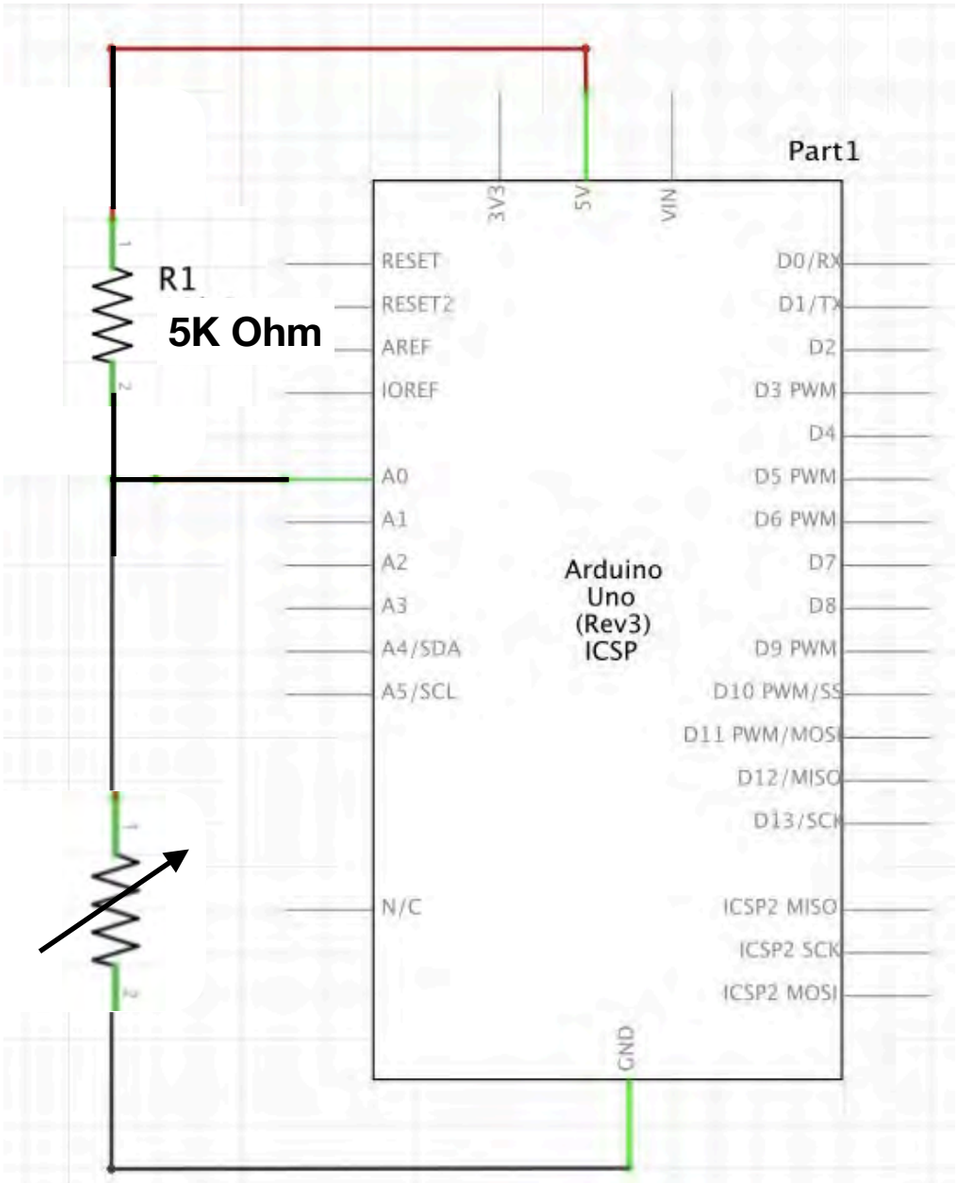

```
Minimum value: 50
Maximum value: 250
```

06_read_AnalogSensor_map.ino

```ino
/*Emma Pareschi,
 * with this sketch we read the analog sensor connected to pin analog_sensor_pin
 */

int analog_sensor_pin = A0;   //change the pin, where the sensor is connected?
int analog_sensor_value = 0;

void setup() {
  // put your setup code here, to run once:
  pinMode(analog_sensor_pin, INPUT);
  Serial.begin(9600);

}

void loop() {
  // put your main code here, to run repeatedly:
  analog_sensor_value = analogRead(analog_sensor_pin); //read the Voltage of the pin sensor

  analog_sensor_value = map(analog_sensor_value, 50, 250, 0, 255); //we change the range
  analog_sensor_value = constrain(analog_sensor_value, 0, 255); //we apply the limits

  Serial.println(analog_sensor_value); // print the value on the Serial monitor
  delay(100);

}
```

06_read_AnalogSensor_map

Smooth the value of an Analog sensor



Smoothing

```
// Define the number of samples to keep track of. The higher the number, the
// more the readings will be smoothed, but the slower the output will respond to
// the input. Using a constant rather than a normal variable lets us use this
// value to determine the size of the readings array.
const int numReadings = 10;

int readings[numReadings];      // the readings from the analog input
int readIndex = 0;              // the index of the current reading
int total = 0;                  // the running total
int average = 0;                // the average

int inputPin = A0;

void setup() {
  // initialize serial communication with computer:
  Serial.begin(9600);
  // initialize all the readings to 0:
  for (int thisReading = 0; thisReading < numReadings; thisReading++) {
    readings[thisReading] = 0;
  }
}

void loop() {
  // subtract the last reading:
  total = total - readings[readIndex];
  // read from the sensor:
  readings[readIndex] = analogRead(inputPin);
  // add the reading to the total:
  total = total + readings[readIndex];
  // advance to the next position in the array:
  readIndex = readIndex + 1;

  // if we're at the end of the array...
  if (readIndex >= numReadings) {
    // ...wrap around to the beginning:
    readIndex = 0;
```

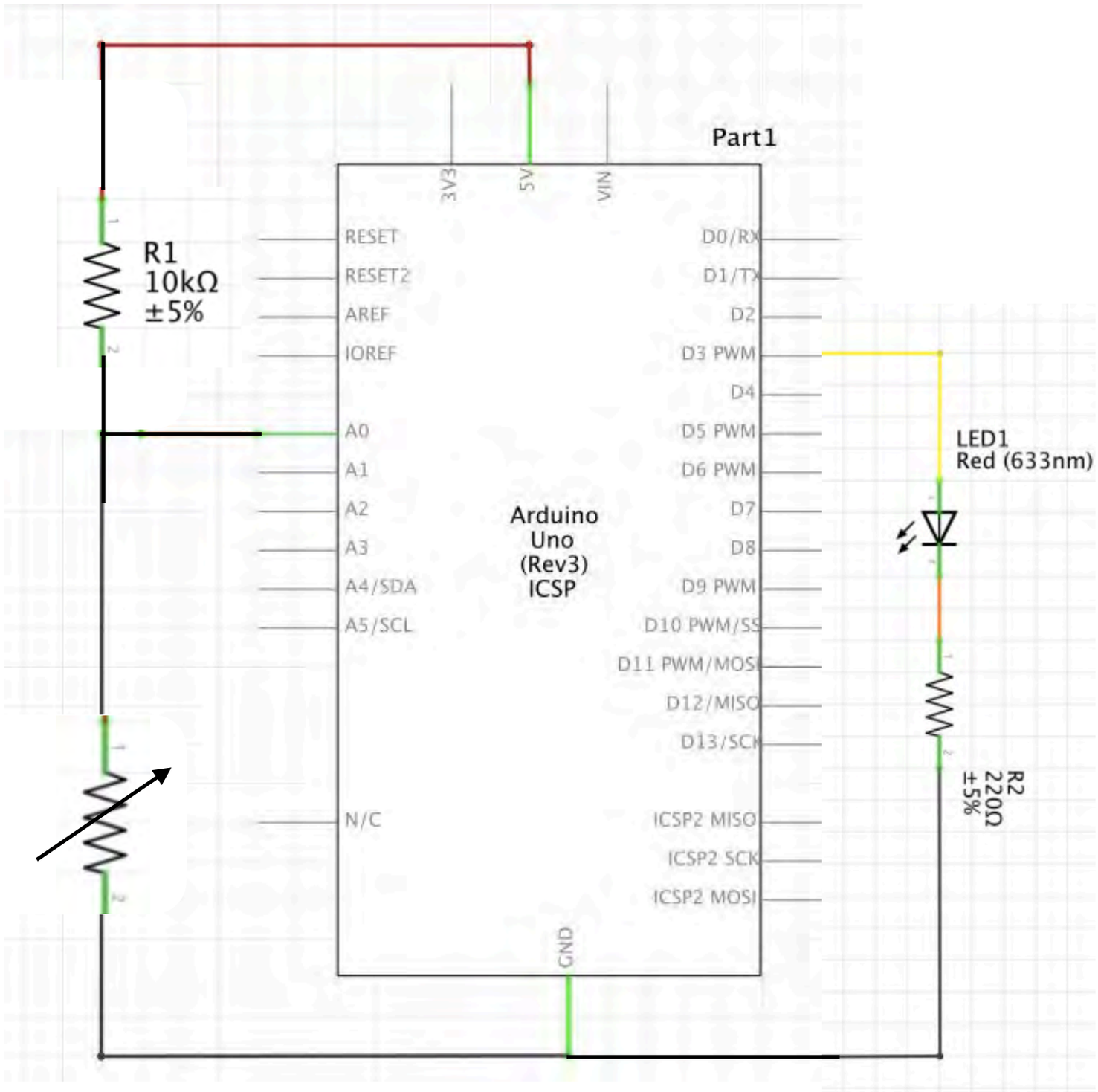
Example -> Analog -> Smoothing

Open Tools -> Serial Plotter and you will see how much smooth is the signal of the sensor… nice :)

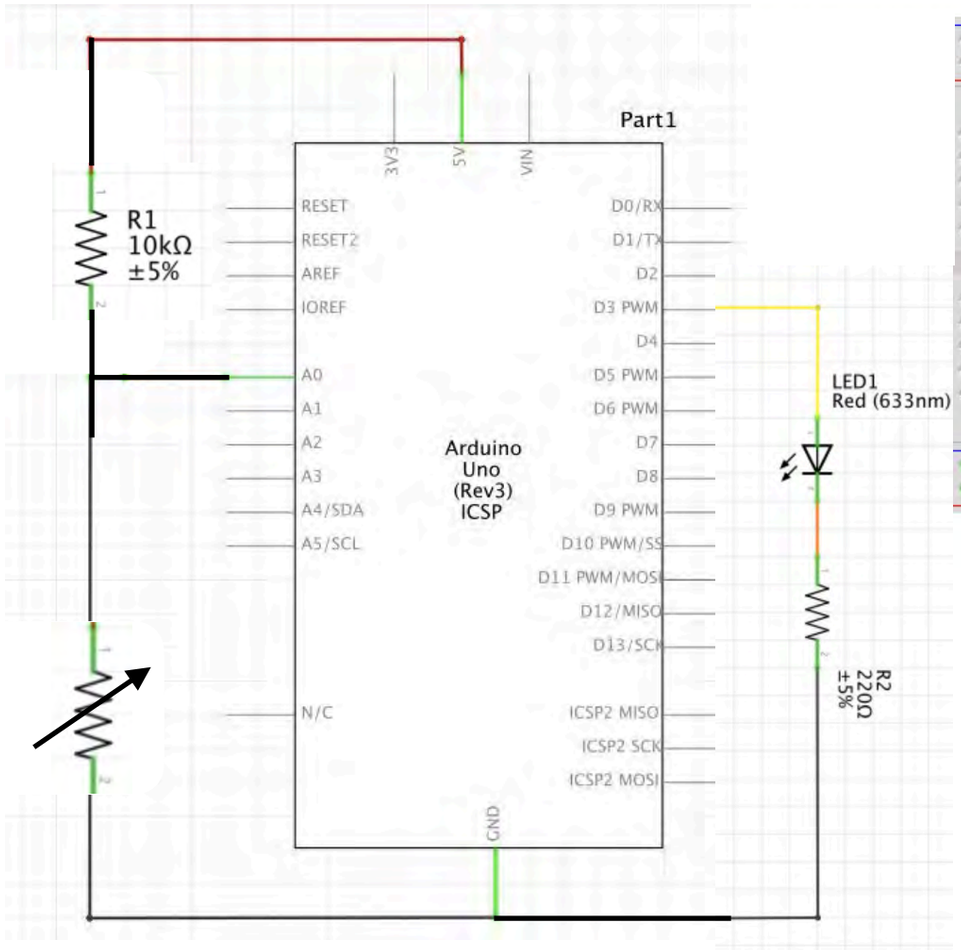
Arduino: Analog Sensor and Led

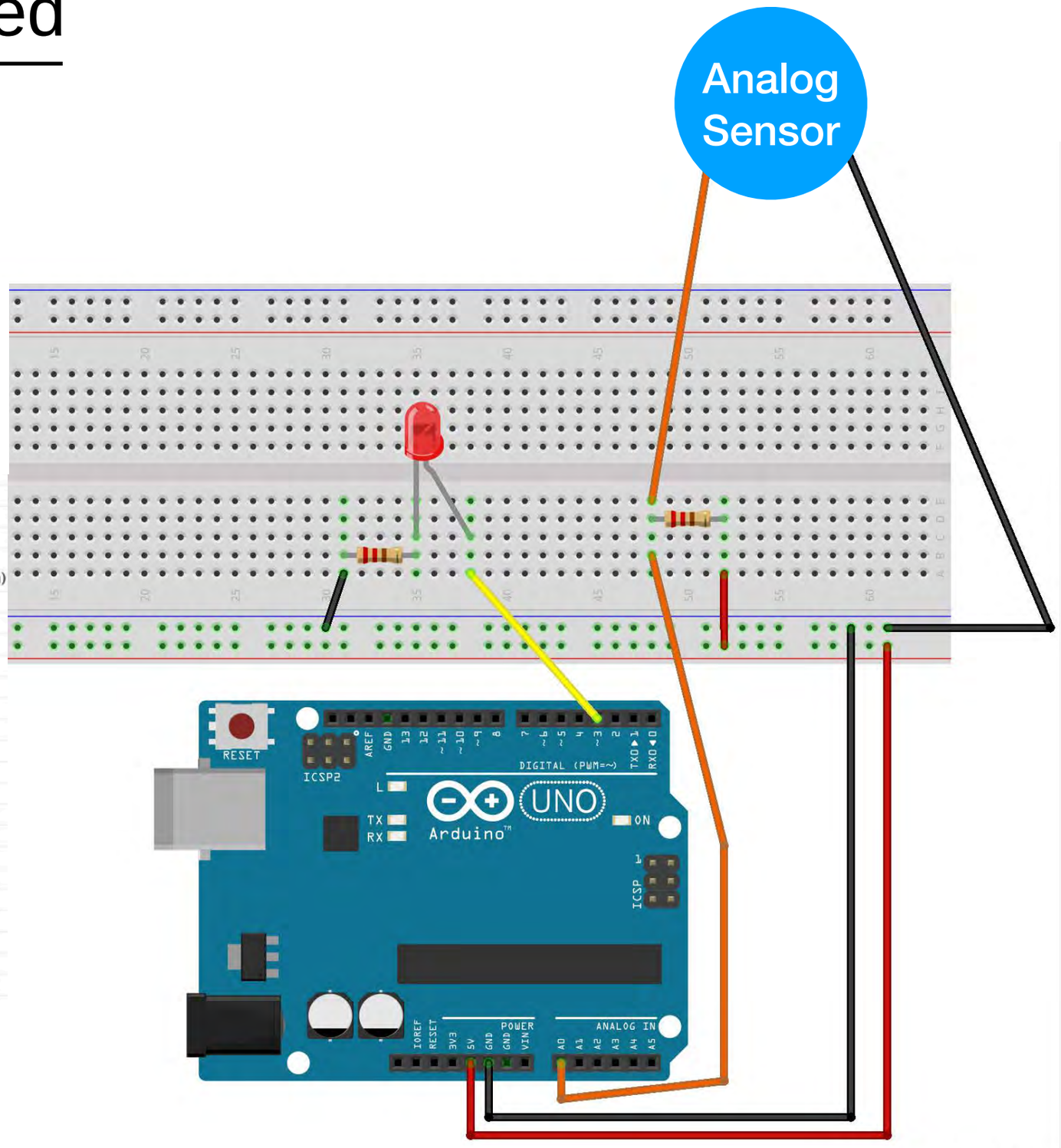
Use the Analog sensor to control a Led

Analog sensor and a Led

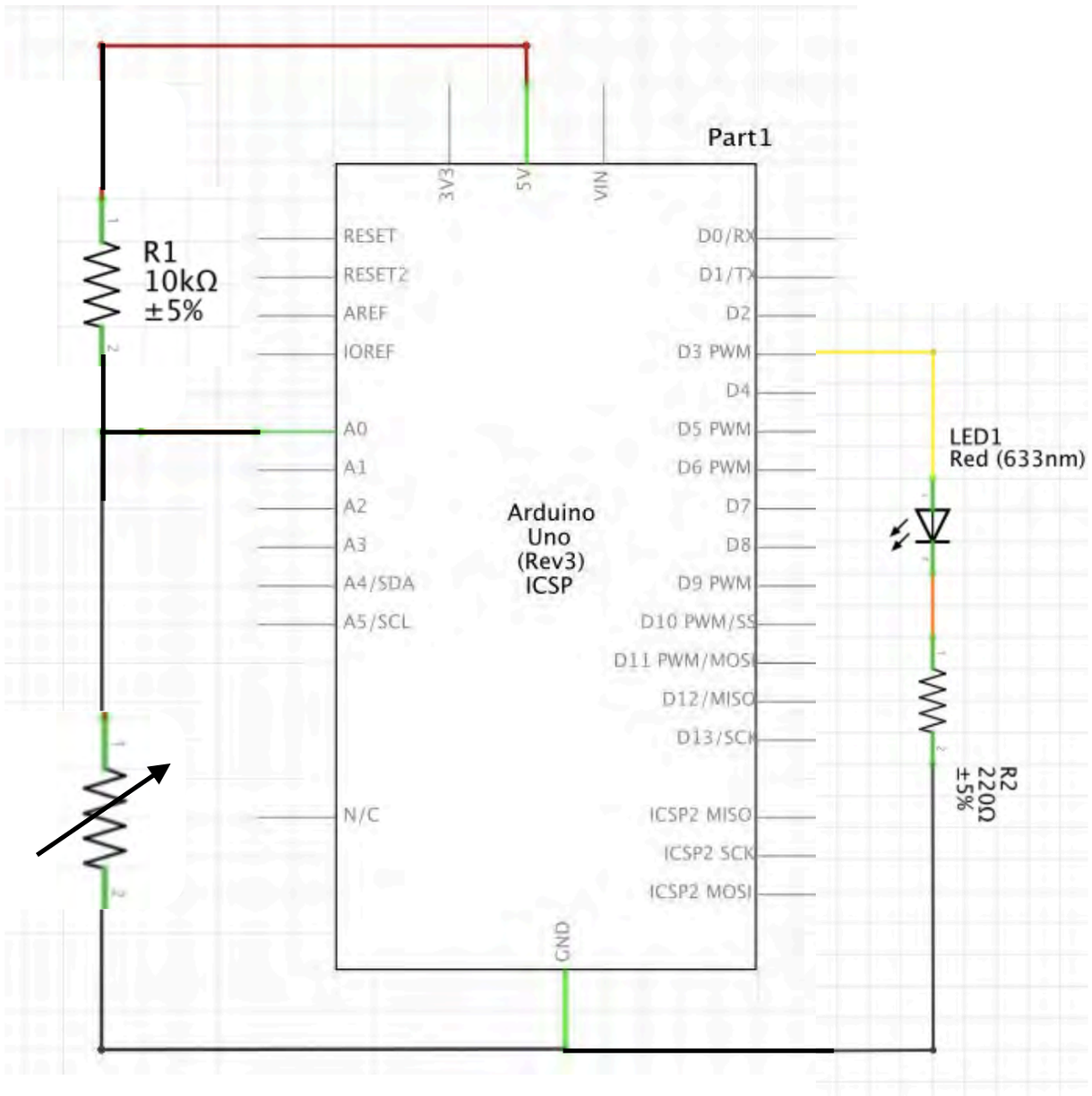


Analog sensor and a Led





Analog sensor and a Led

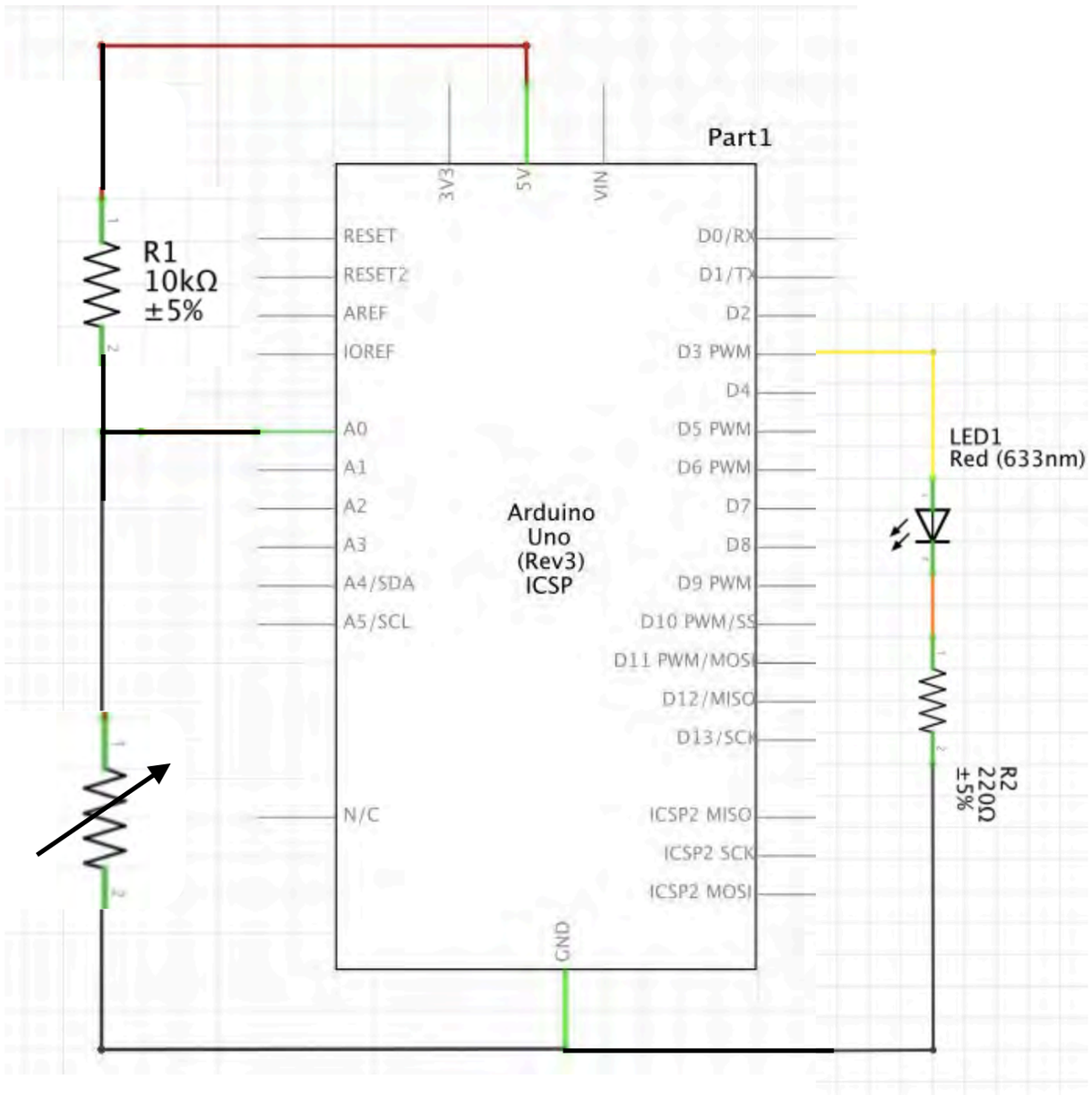


07_AnalogSensor_led_fade.ino

```
37    Serial.begin(9600);
38    // initialize all the readings to 0:
39    for (int thisReading = 0; thisReading < numReadings; thisReading++) {
40      readings[thisReading] = 0;
41    }
42
43      pinMode(led_pin, OUTPUT);   //initialize led pin
44  }
45
46  void loop() {
47    // subtract the last reading:
48    total = total - readings[readIndex];
49    // read from the sensor:
50    readings[readIndex] = analogRead(inputPin);
51    // add the reading to the total:
52    total = total + readings[readIndex];
53    // advance to the next position in the array:
54    readIndex = readIndex + 1;
55
56    // if we're at the end of the array...
57    if (readIndex >= numReadings) {
58      // ...wrap around to the beginning:
59      readIndex = 0;
60    }
61
62    // calculate the average:
63    average = total / numReadings;
64
65
66    // send it to the computer as ASCII digits
67    Serial.println(average);
68    average = map(average, 60, 230, 0, 255); //we change the range
69    average = constrain(average, 0, 255); //we apply the limits
70
71    delay(50);        // delay in between reads for stability
72    analogWrite(led_pin, average);
73  }
74
```

07_AnalogSensor_led_fade

Analog sensor and a Led



07_AnalogSensor_led_fade.ino

```
  Serial.begin(9600);
  // initialize all the readings to 0:
  for (int thisReading = 0; thisReading < numReadings; thisReading++) {
    readings[thisReading] = 0;
  }

  pinMode(led_pin, OUTPUT);   //initialize led pin
}

void loop() {
  // subtract the last reading:
  total = total - readings[readIndex];
  // read from the sensor:
  readings[readIndex] = analogRead(inputPin);
  // add the reading to the total:
  total = total + readings[readIndex];
  // advance to the next position in the array:
  readIndex = readIndex + 1;

  // if we're at the end of the array...
  if (readIndex >= numReadings) {
    // ...wrap around to the beginning:
    readIndex = 0;
  }

  // calculate the average:
  average = total / numReadings;


  // send it to the computer as ASCII digits
  Serial.println(average);
  average = map(average, 60, 230, 0, 255); //we change the range
  average = constrain(average, 0, 255); //we apply the limits

  delay(50);        // delay in between reads for stability
  analogWrite(led_pin, average);
}

```

08_AnalogSensor_led_blink

Sketches

This is the list of sketches we used in class, some are provided with this presentation, some come from the Example of Arduino IDE.

Control a Led

- 01_blink_Led
- Fade (examples)

Read a Digital Sensor and print the value on the Serial Monitor

- 03_read_DigitalSensor

Control a Led with a Digital Sensor

- 04_button_led_blink
- 04_button_led_on_off

Read an Analog Sensor and print the value on the Serial Monitor

- 05_read_AnalogSensor
- 06_read_AnalogSensor_map
- Smoothing (from Example)

Control a Led with an Analog Sensor

- 07_AnalogSensor_led_fade
- 08_AnalogSensor_led_blink

Workflow

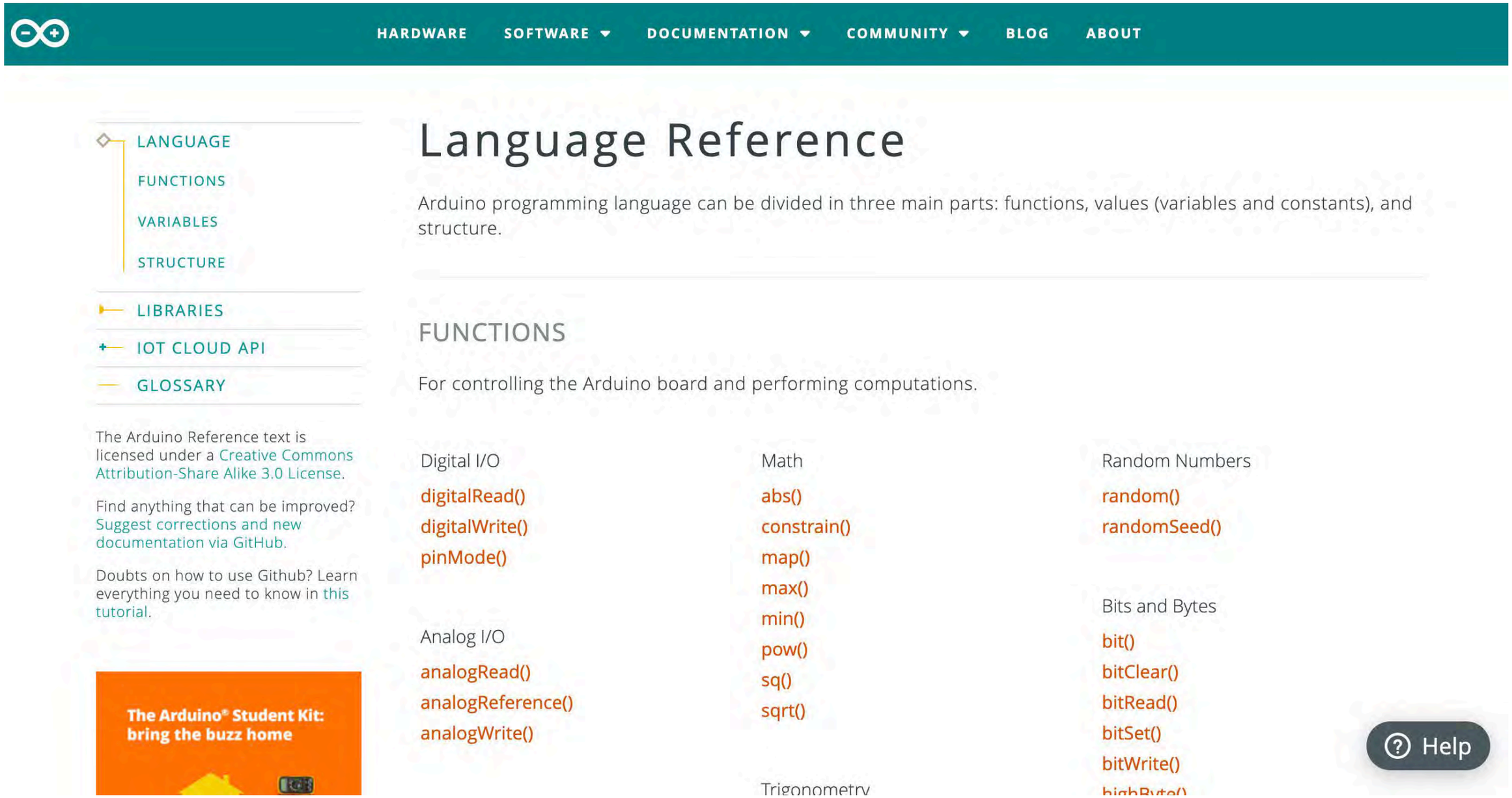
1- decide which sensor to make, if it is your first sensor, follow a tutorial. If you already have experience, have fun and try something new or change a sensor already documented.

2- Once you have the sensor, test it with the multimeter.

3- Once the sensor looks ok (with the multimeter), connect it to the Arduino and read the value on the Serial Monitor or Serial Plotter.

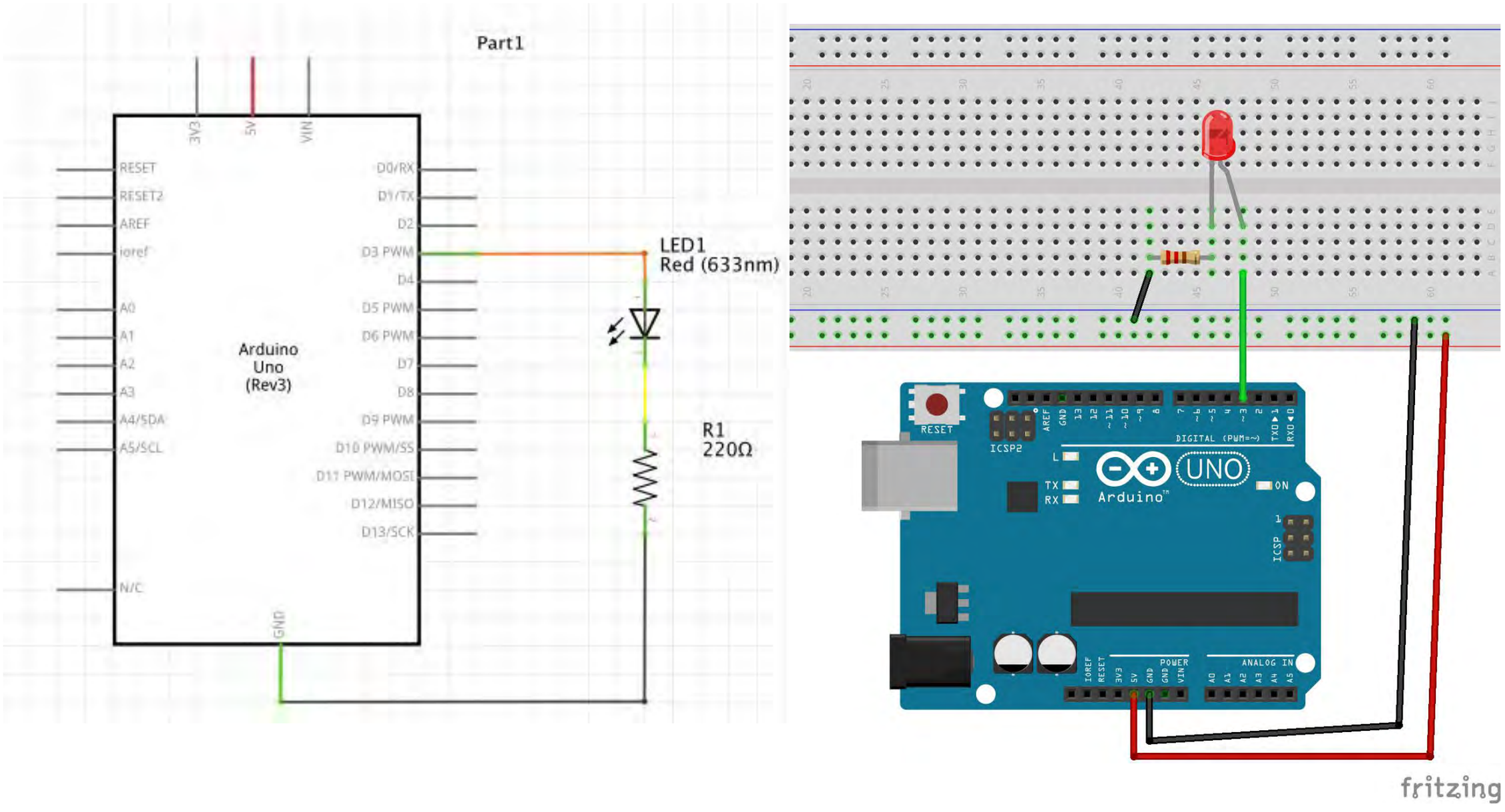
4- Once you can read the values of the sensor with the Arduino, connect the Led and try to control the Led based on the interaction with the sensor.

Arduino Language Reference

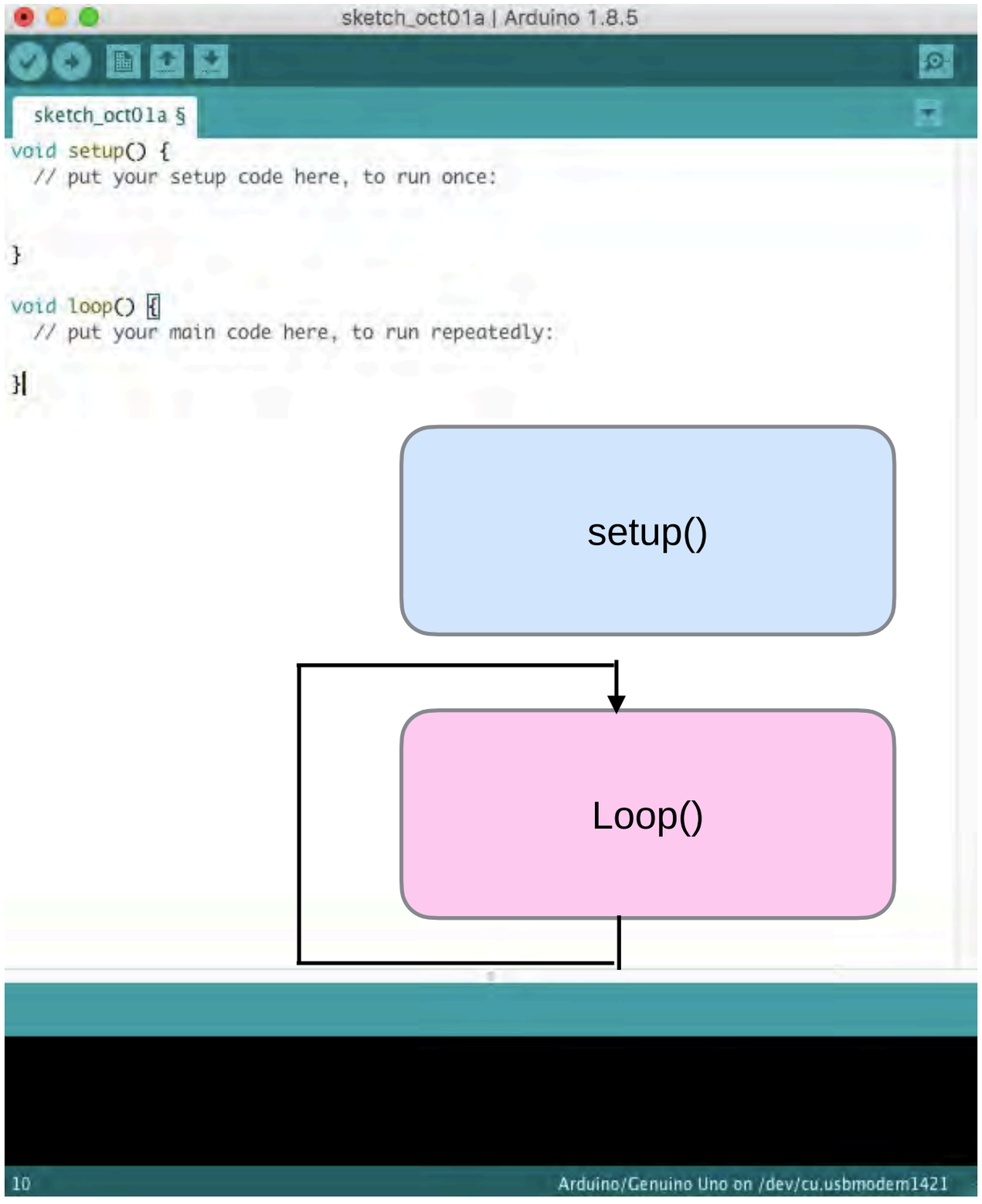


https://www.arduino.cc/reference/en/

Control a Led with Arduino



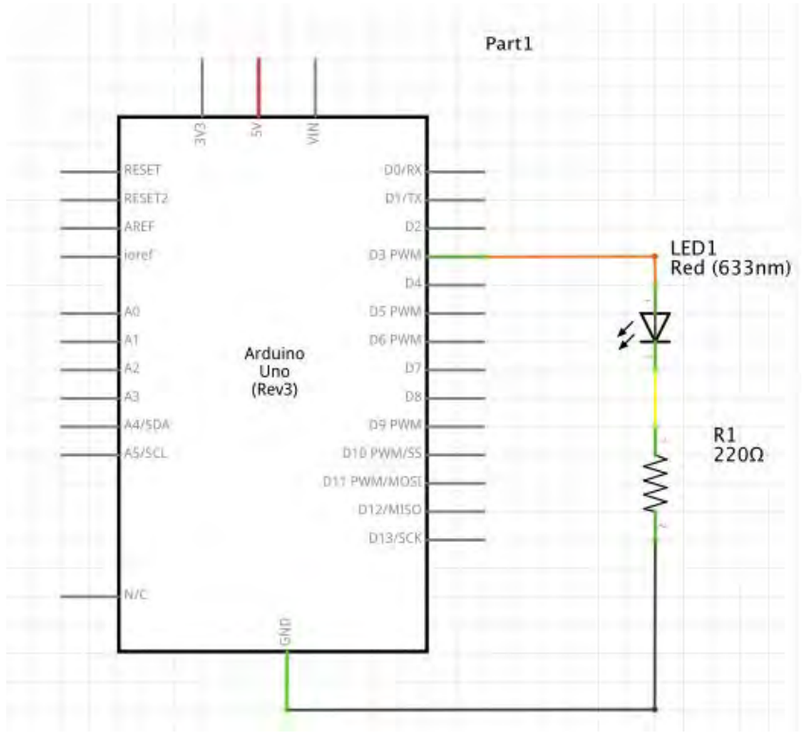
Time to Program

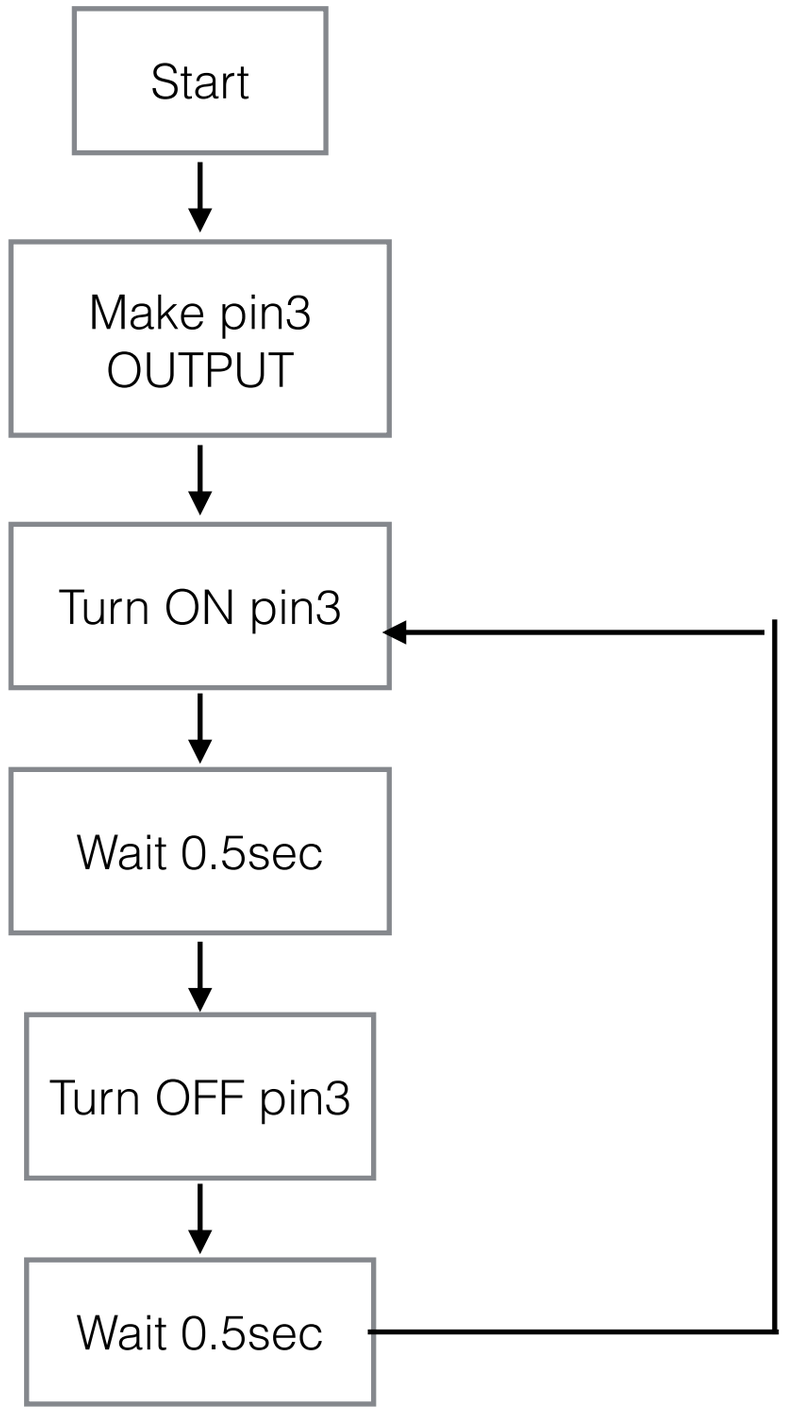


Initialise
Runs once, at the beginning when the Arduino is powered on.
Define the pins.

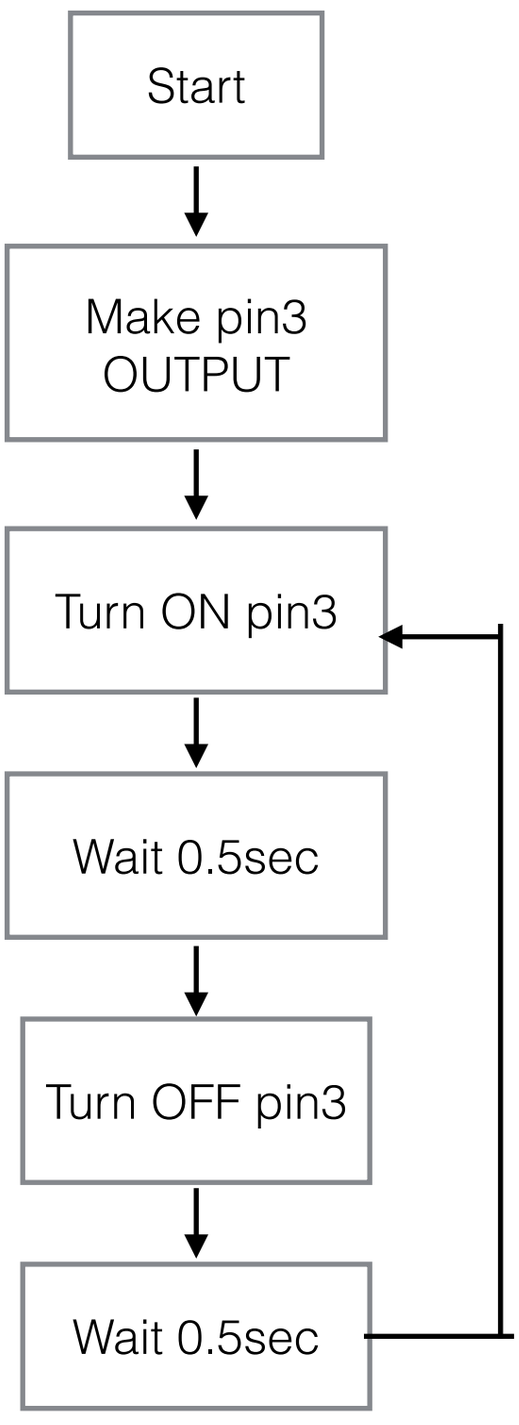
Running
Run repeatedly, after setup.

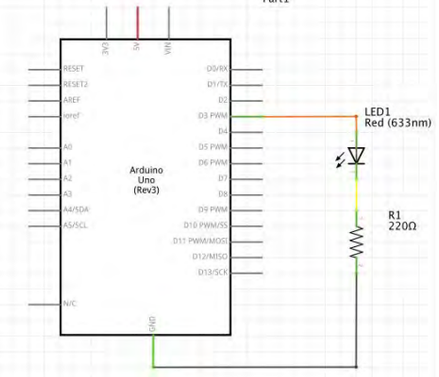
Control Flow





Control Flow





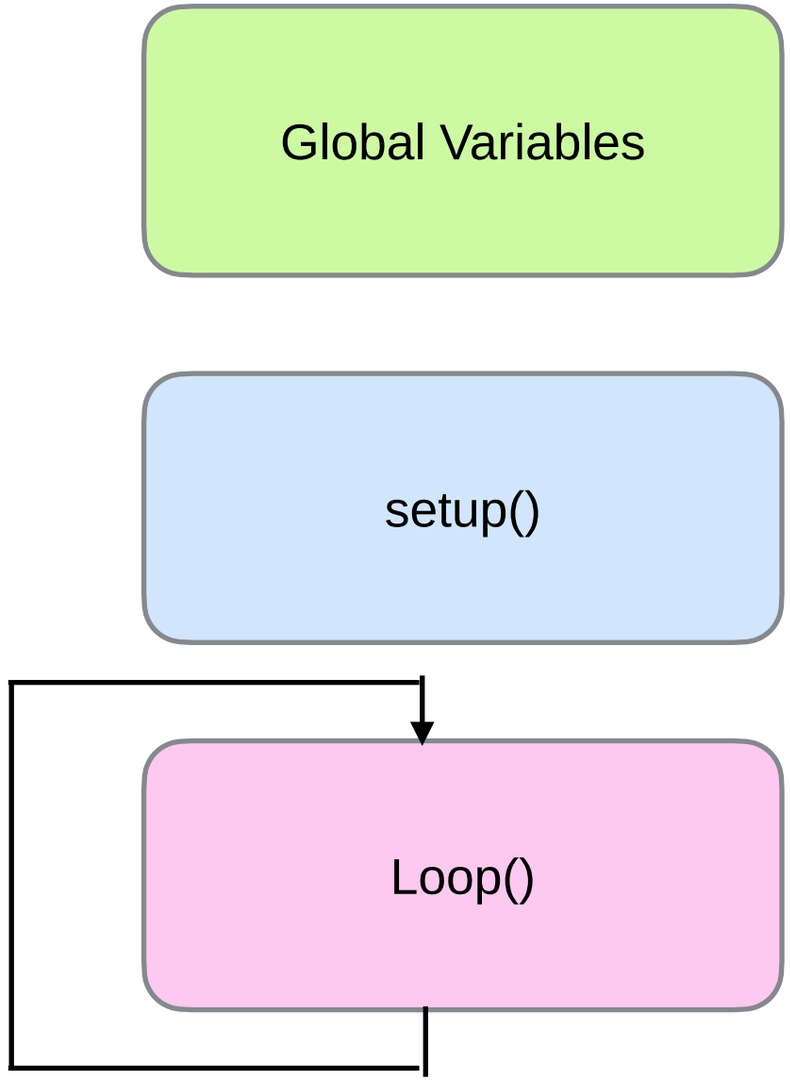
In human language

```
void setup() {
  make the pin 3 an OUTPUT
}

void loop() {
 turn Led ON
 wait 0.5 sec
 turn Led OFF
 wiat 0.5 sec
}
```

In Arduino language

```
void setup() {
  // put your setup code here, to run once:
  pinMode(3, OUTPUT);

}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(3, HIGH);
  delay(500);
  digitalWrite(3, LOW);
  delay(500);
}
```

SKETCH



Declare the variables

Initialise
Runs once, at the beginning
Define the pins

Running
Run repeatedly, after setup

01_blink_led Sketch

Global variables

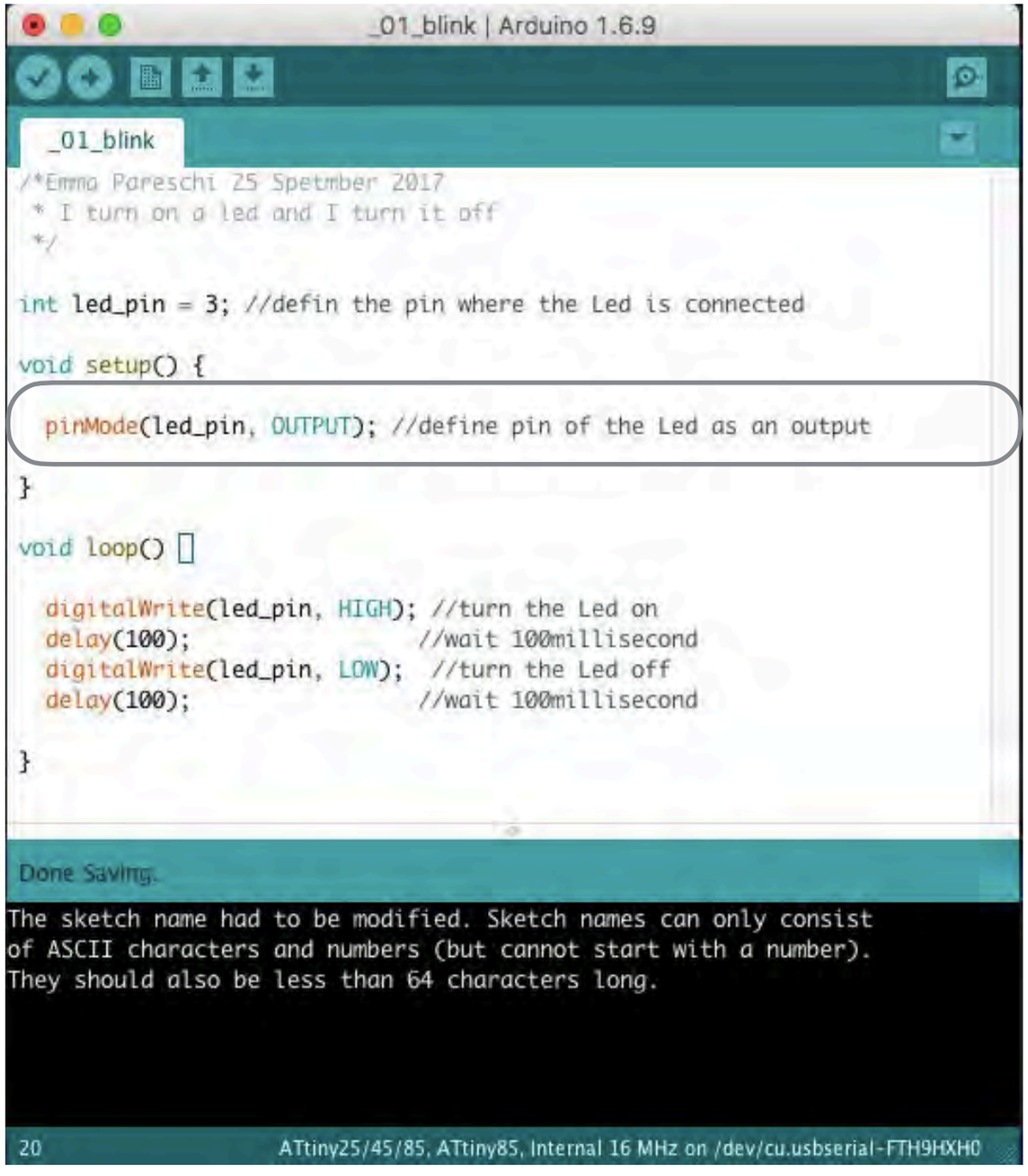
setup()

loop()

Blink Sketch

setup()
Define the function of the pin
pinMode(pin, function);

The pin is ‘led_pin
The functionality is Output



Blink Sketch

Loop()
The list of commands that arduino runs repeatedly.

DigitalWrite(pin, level);
we control the voltage on the pin
HIGH: high voltage (5V)
LOW: low voltage (ground)

Delay(time);
We add a delay in millisecond