FABRICADEMY
textile and technology academy

#FABRICADEMY2021 TUTORIALS

SENSING & OUTPUTS
ATTINY TUTORIAL - PART 2

17-11-2021 | EMMA PARESCHI

Fabricademy | textile & technology academy 2021

Program

Attiny and Serial Monitor:

- How to use the Serial Monitor with the Attiny
  Circuits:
    - Attiny reads a digital sensor
      or
    - Attiny reads an analog sensor

Arduino Uno:

- Capacitive Sensor
  Circuits:
    - Arduino Uno and Capactive Sensor
    - Arduino Uno and 2 Capactive Sensors
    - Arduino Uno, 2 capacitive sensors and 2 Leds

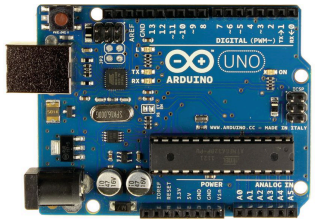
- PiezoResistive Matrix (Touchpad)
  Circuits:
    - Arduino Uno and Matrix
    - Arduino Uno, Matrix and Neopixel

Attiny and Serial Monitor

How to send Data from the micro controller to PC

Input Device

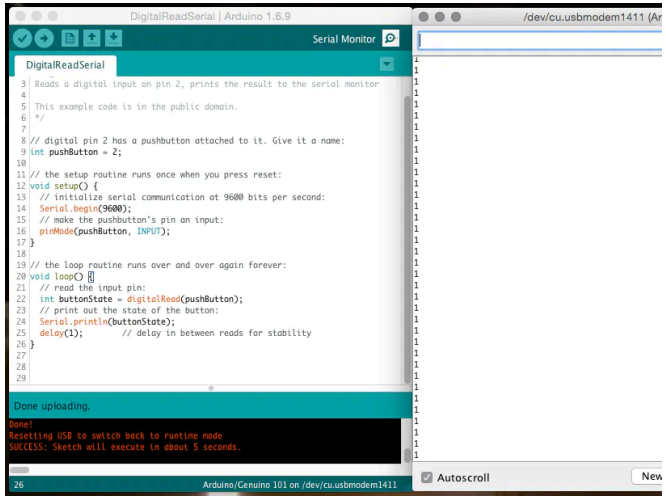
Data flow



Arduino Uno

USB cable

Laptop



Arduino IDE and Serial Monitor

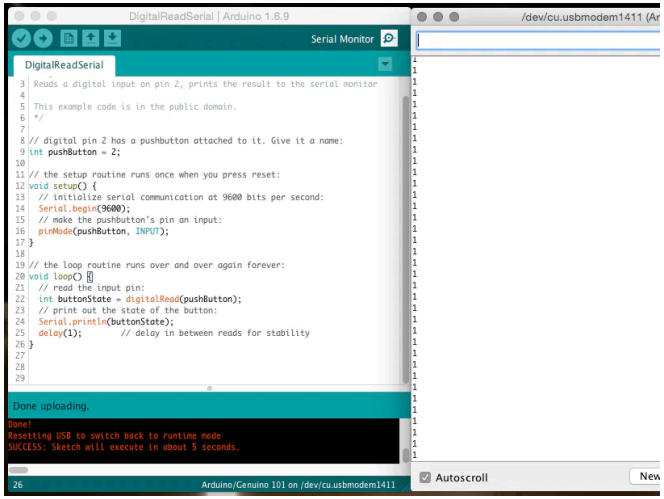
Input Device

Attiny45
Attiny85

Data flow

?

Laptop



Arduino IDE and Serial Monitor

How to send Data from the micro controller to PC

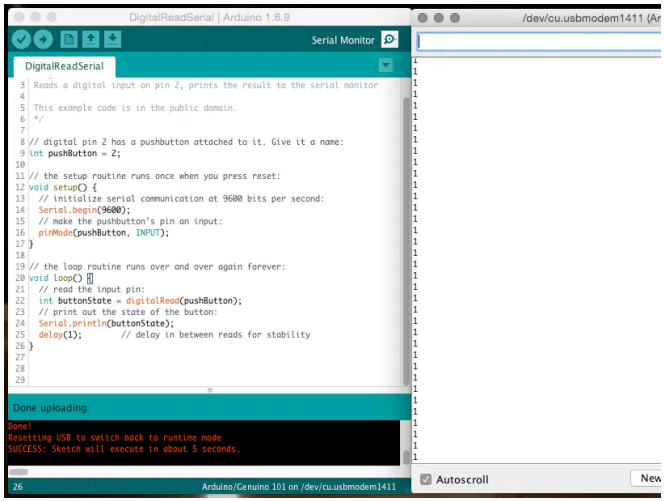
Input Device

Attiny45
Attiny85

Data flow

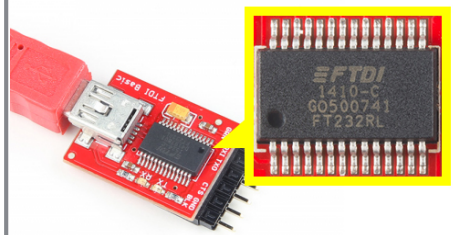
?

Laptop



Arduino IDE and Serial Monitor

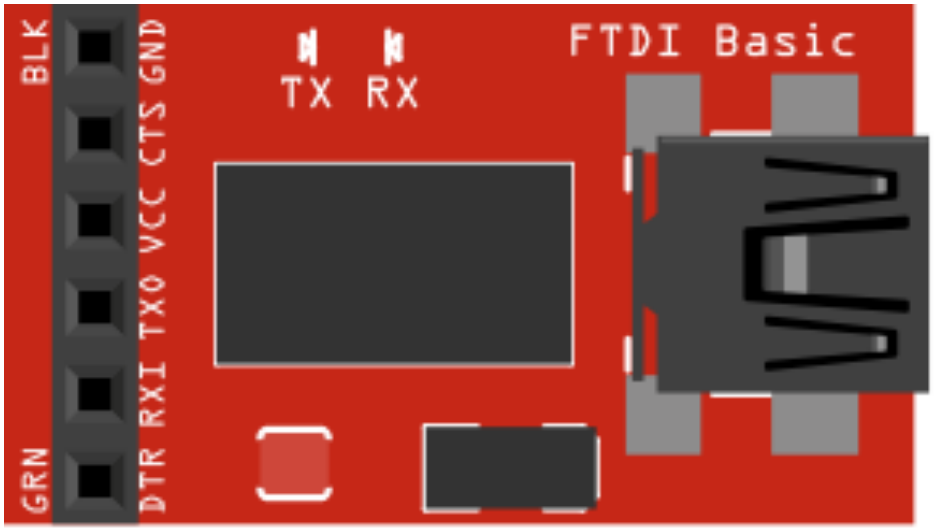
FTDI chip:



FTDI Chip and Cable

To be used they need drivers:
https://learn.sparkfun.com/tutorials/how-to-install-ftdi-drivers/introduction

FTDI board or cable



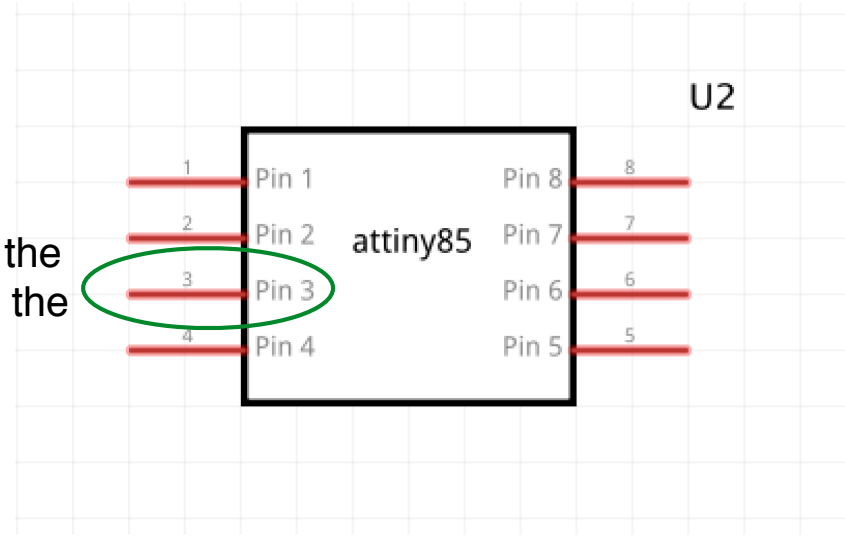
Pins that we will use:

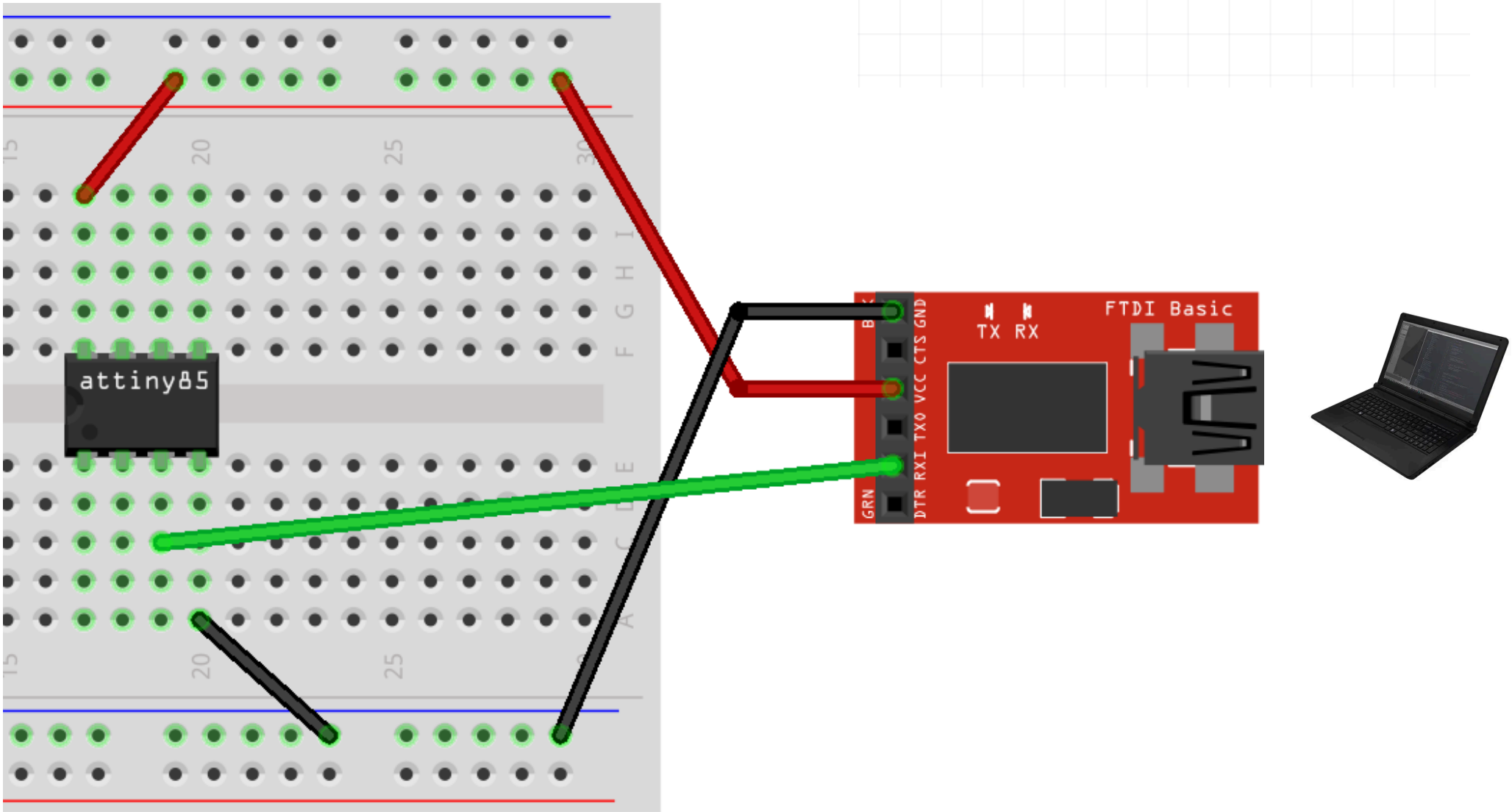
- VCC
- GND
- RX

Check the reference of you hardware to know the orientation of the pins or use the multimeter to find GND and VCC

FTDI board or cable

We use pin 3 to send the data from the Attiny to the PC





Steps

- 1. Decide what you want to measure

- 2. Connect the Attiny to your Programmer:
  - AVR programmer
  - Arduino UNO (make sure that the Arduino is uploaded with the ARduinoISP sketch).

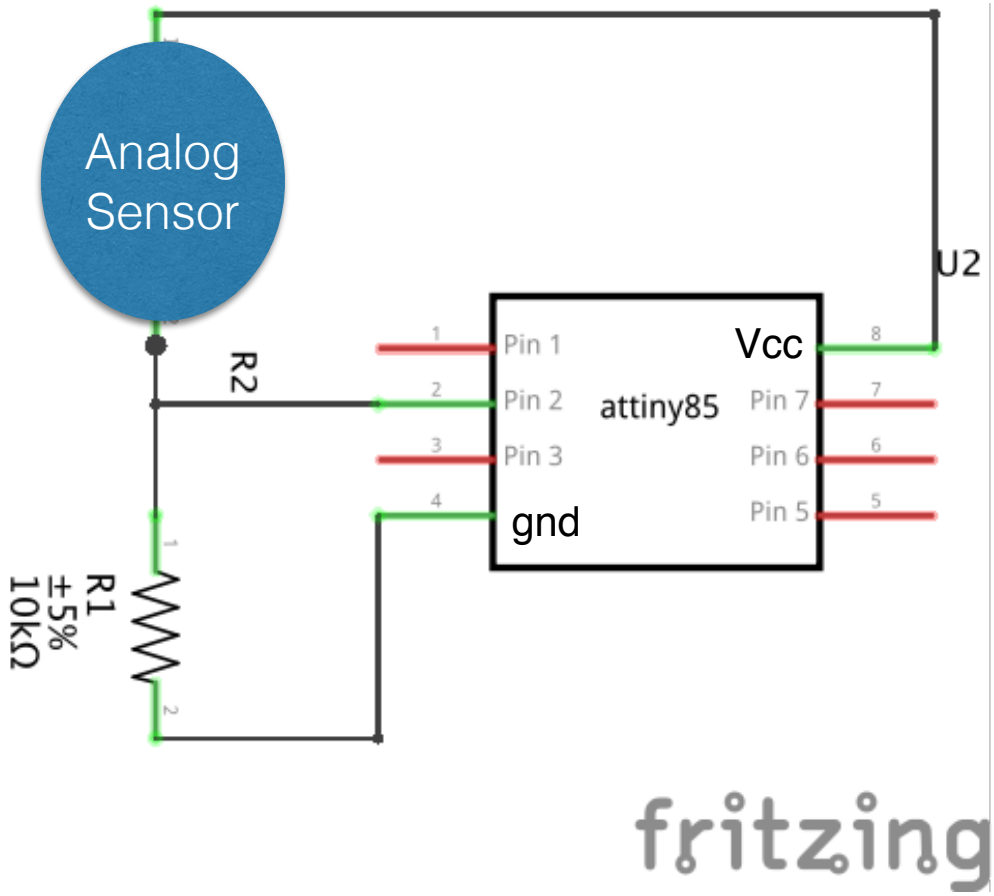
- 3. Program the Attiny to read the sensor and send the data trough Serial Communication.

- 4. Remove the programmer. Connect your Input device to the Attiny.
  Connect the FTDI Board

- 5. Plug the FTDI board (or cable) and Open the Serial Monitor or Serial Plotter

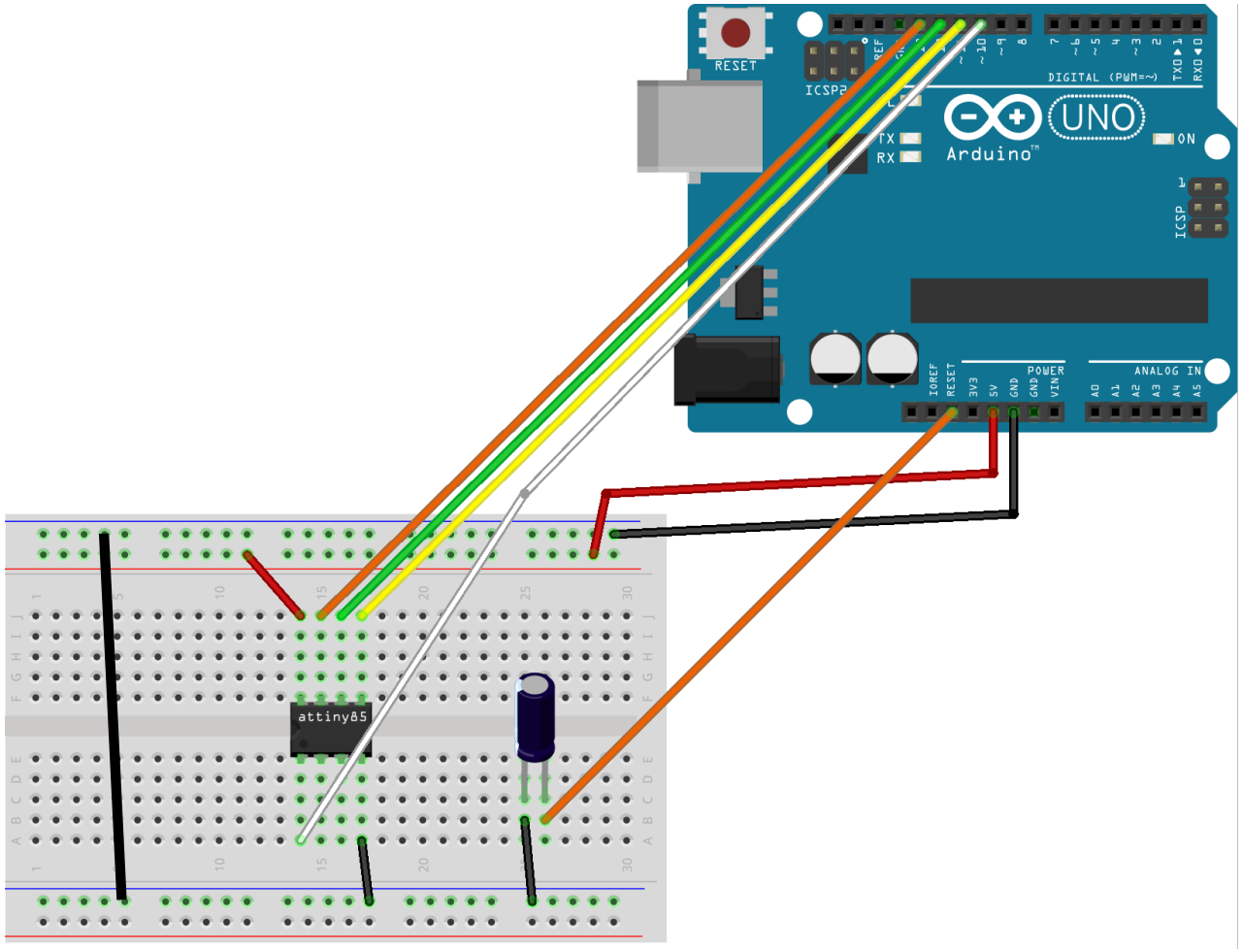
Step 1

- We want to read the data of an analog sensor



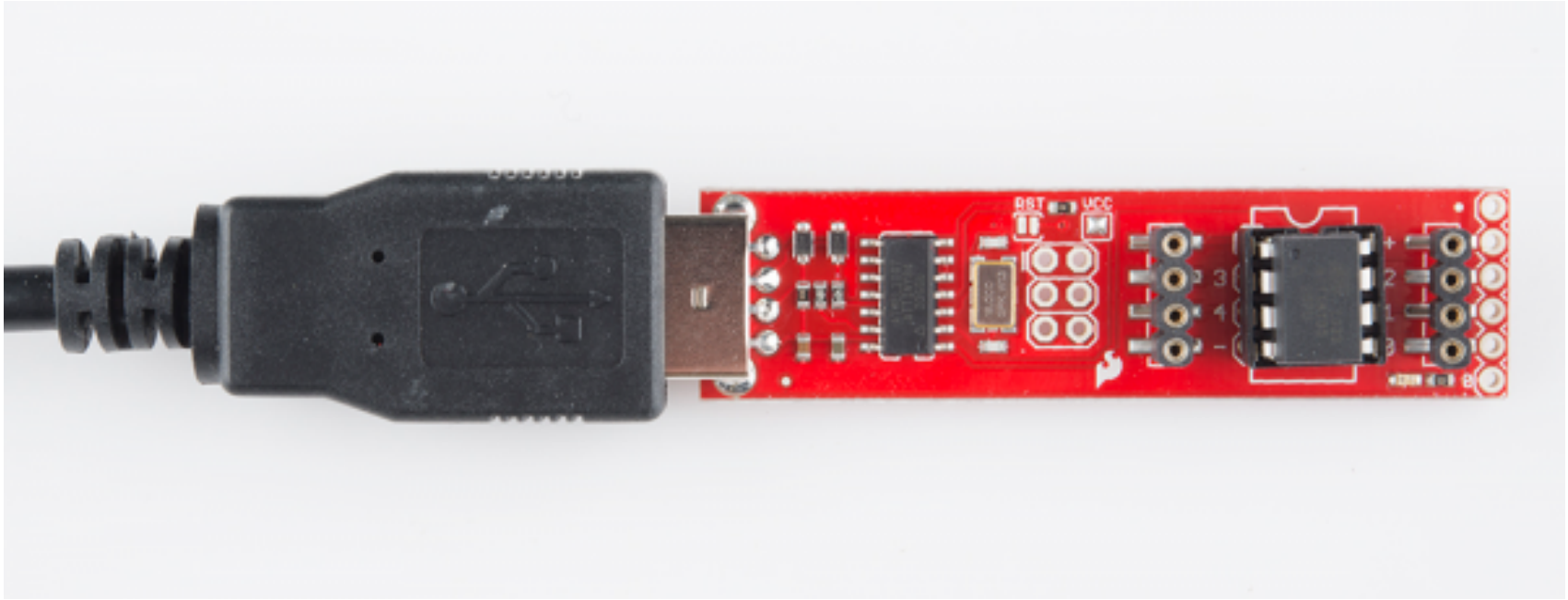
Step 2

- Connect the Attiny to your Programmer:
  - AVR programmer
  - Arduino UNO (make sure that the Arduino is uploaded with the ArduinoISP sketch)



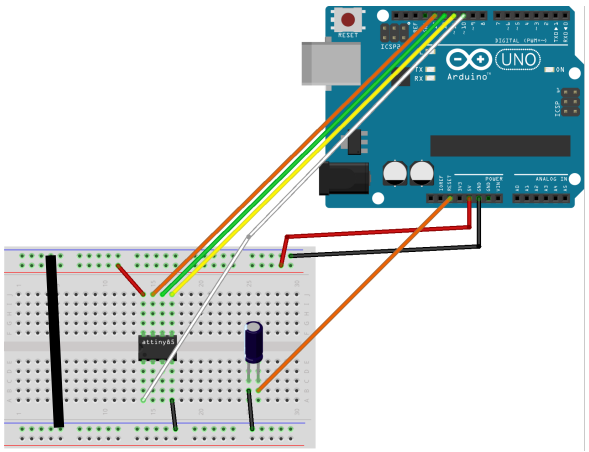
Step 2

- Connect the Attiny to your Programmer:
  - AVR programmer
  - Arduino UNO (make sure that the Arduino is uploaded with the ArduinoISP sketch)



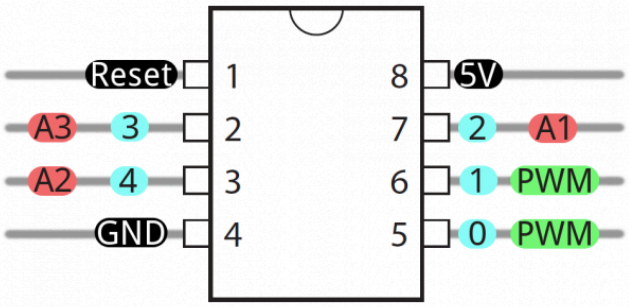
Step 3

- Program the Attiny to read the sensor and send the data trough Serial Communication.



Few consideration that are needed to write the code:

- the sensor is connected to PIN 2 (Arduino pin3)
- the FTDI board is connected to PIN 3 (Arduino pin 4)



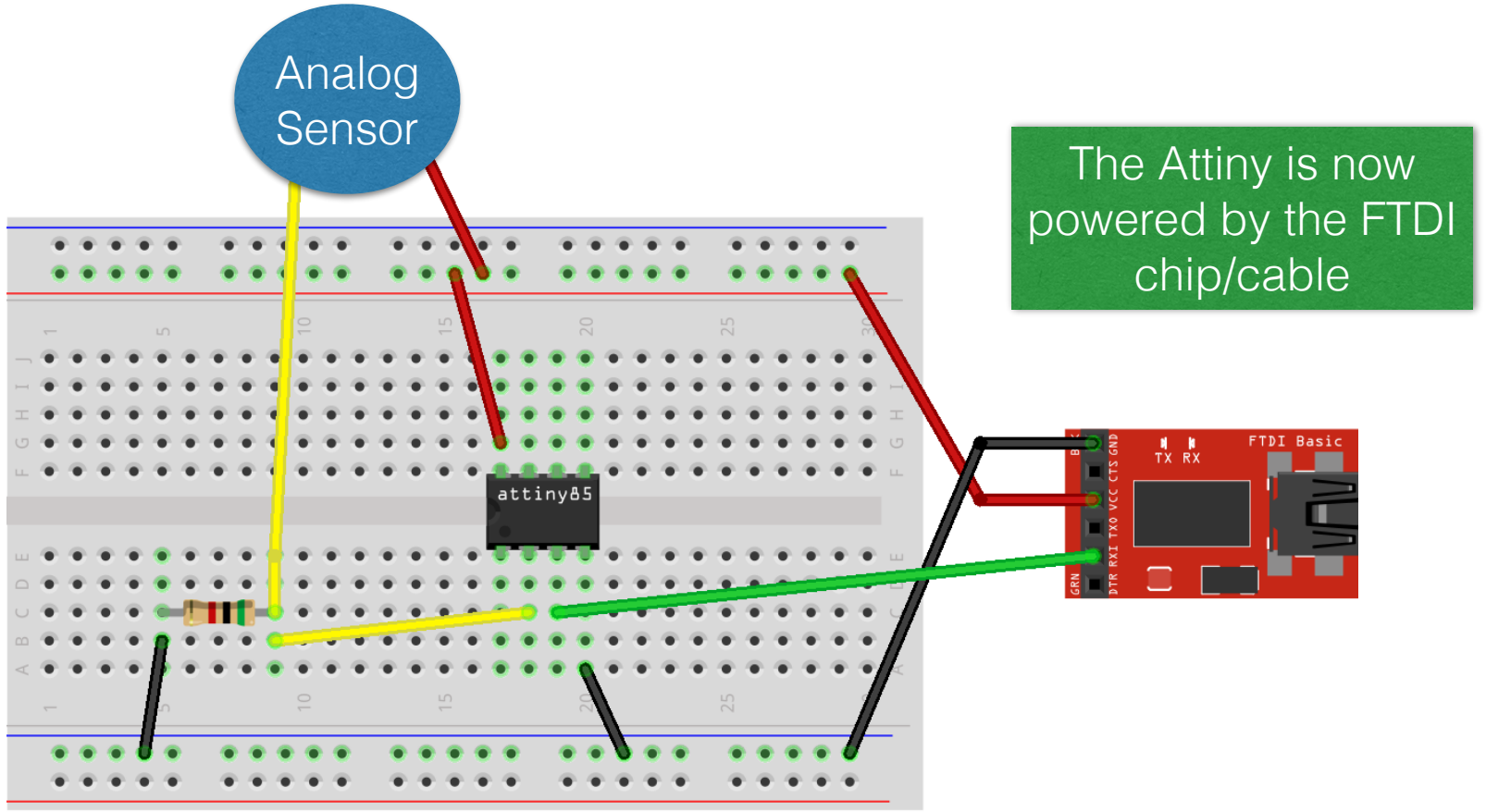
code: attiny_analog_sensor

attiny_analog_sensor

```
#include <SoftwareSerial.h>

#define rxPin 11  //not used pin
#define txPin 4   //the pin we will use to transmit the data (it i

SoftwareSerial serial(rxPin, txPin);     //create the communicatic

int analog_sensor_pin = 3;   //change the pin, where the sensor is
int analog_sensor_value = 0;

void setup() {

pinMode(rxPin, INPUT);
pinMode(txPin, OUTPUT);
pinMode(analog_sensor_pin, INPUT); //define the pin as INPUT
serial.begin(9600);   //open communication

}

void loop(){

 analog_sensor_value = analogRead(analog_sensor_pin); // read the
 serial.println(analog_sensor_value); //print on Serial Monitor

delay(100);
}
```

Step 4

- Remove the programmer.
- Connect your Input device to the Attiny.
  Connect the FTDI Board

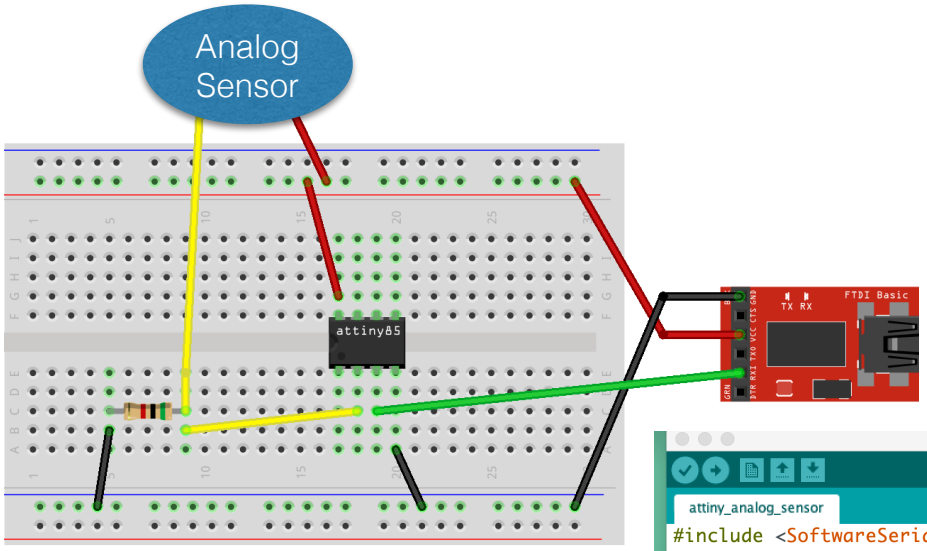


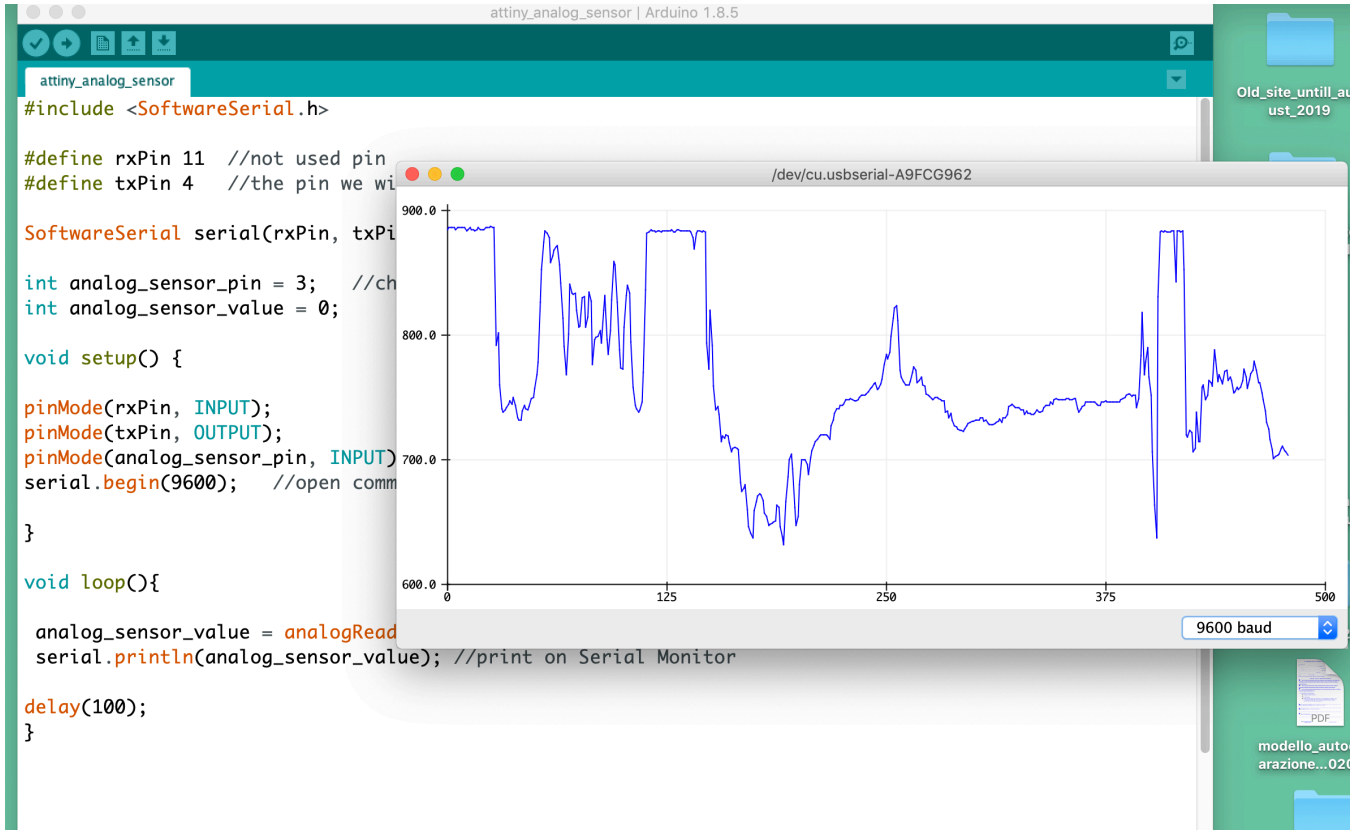
Few consideration that are needed to write the code:

- the sensor is connected to PIN 2
- the FTDI board is connected to PIN 3

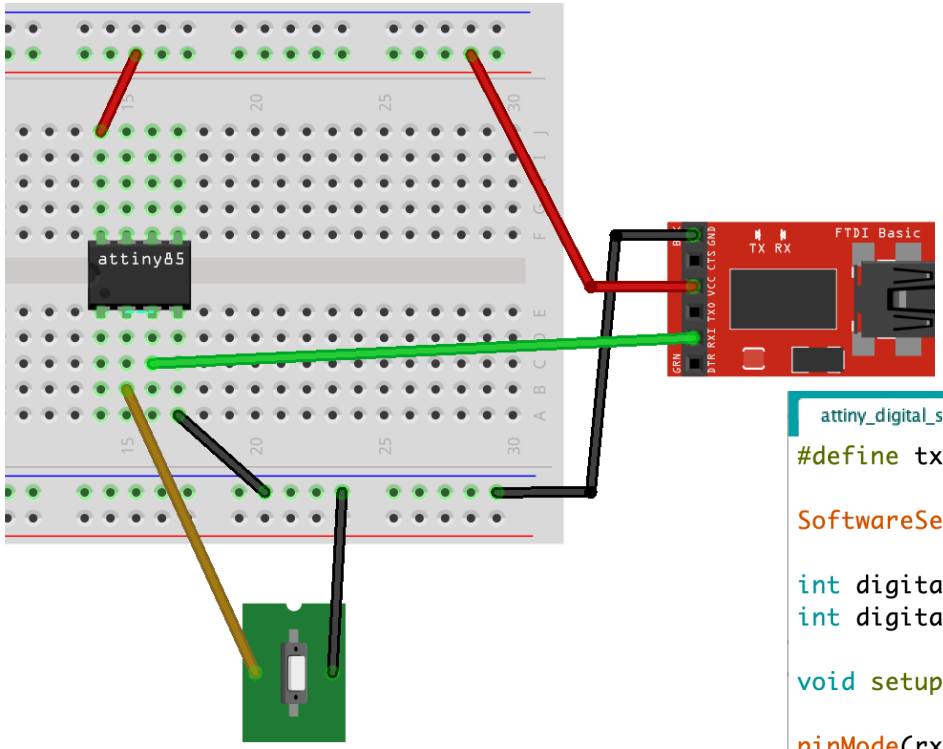
Step 5

- Plug the FTDI board (or cable), select the USB port and Open the Serial Monitor or Serial Plotter





Attiny and Digital Sensor



code: attiny_digital_sensor

attiny_digital_sensor

```
#define txPin 4

SoftwareSerial serial(rxPin, txPin);

int digital_sensor_pin = 3;     //change the pin, where the sensor is connecte
int digital_sensor_value = 0;

void setup() {

pinMode(rxPin, INPUT);
pinMode(txPin, OUTPUT);
pinMode(digital_sensor_pin, INPUT_PULLUP); //define the pin as INPUT
serial.begin(9600);

}

void loop(){

 digital_sensor_value = digitalRead(digital_sensor_pin); // read the sensor
 serial.println(digital_sensor_value);

delay(1000);
}
```

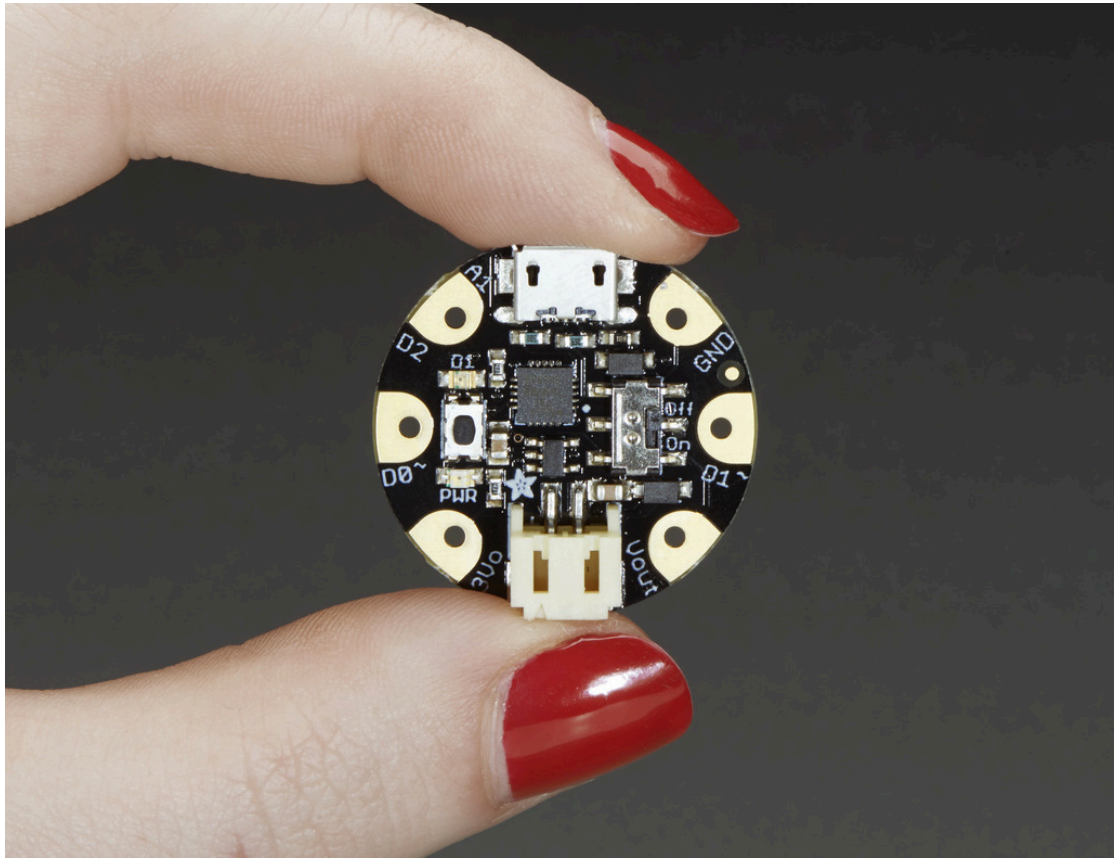
Done Saving.

Few consideration that are needed to write the code:

- the sensor is connected to PIN 2 (Arduino pin3)
- the FTDI board is connected to PIN 3 (Arduino pin 4)

Gemma

Gemma



Adafruit's mini wearable microcontroller

It is an Attiny85 breakout board.

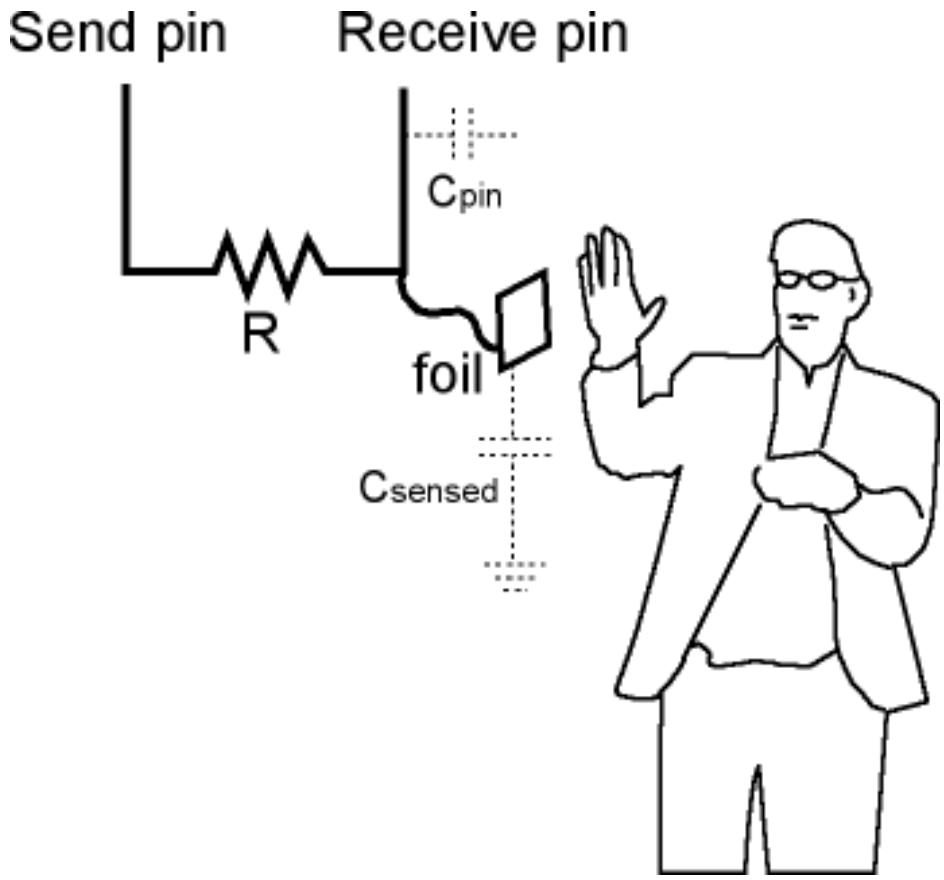
You can program it with just an USB cablee.
No Need of Programmer!!!

It doesn’t have a Serial port, you can’t use the Serial Monitor.

https://learn.adafruit.com/introducing-gemma/introduction

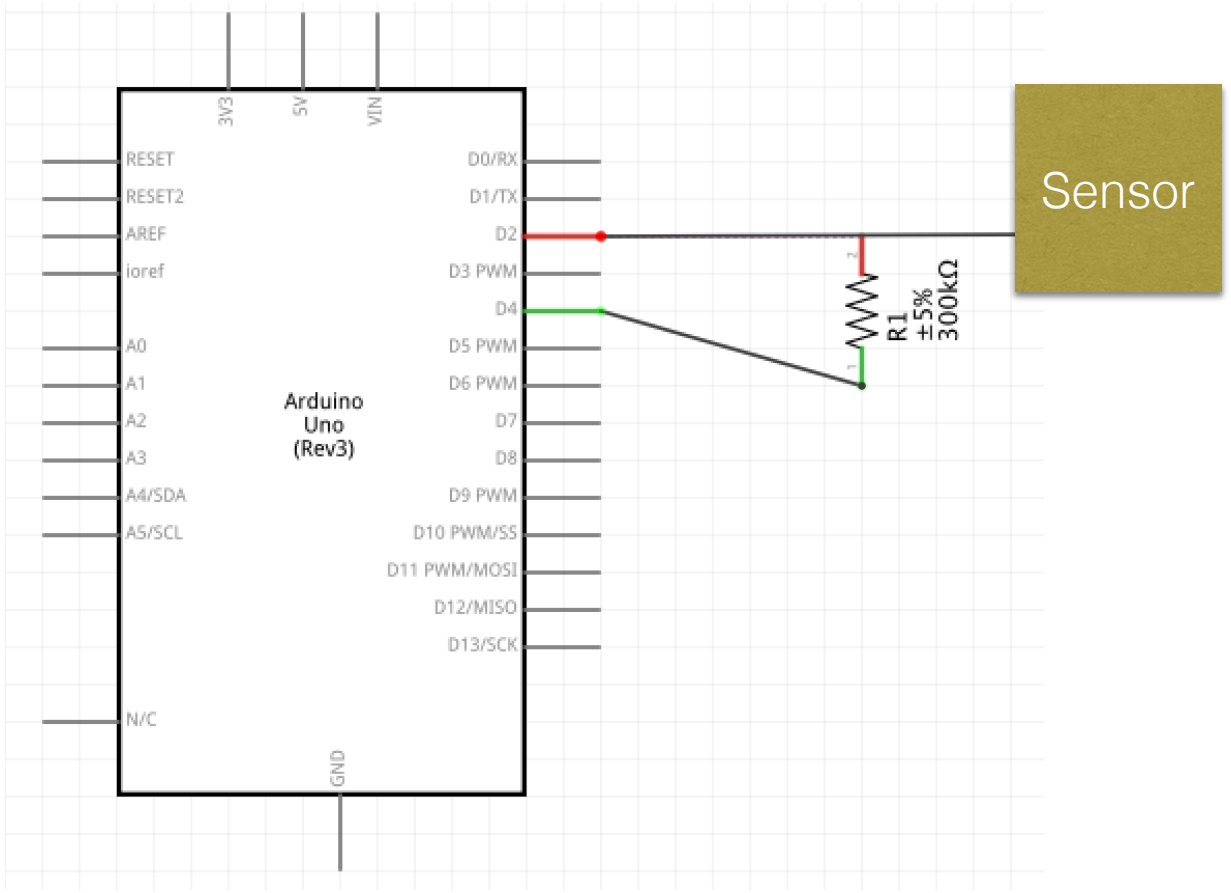
Capacitive Sensor

Capacitive Sensor

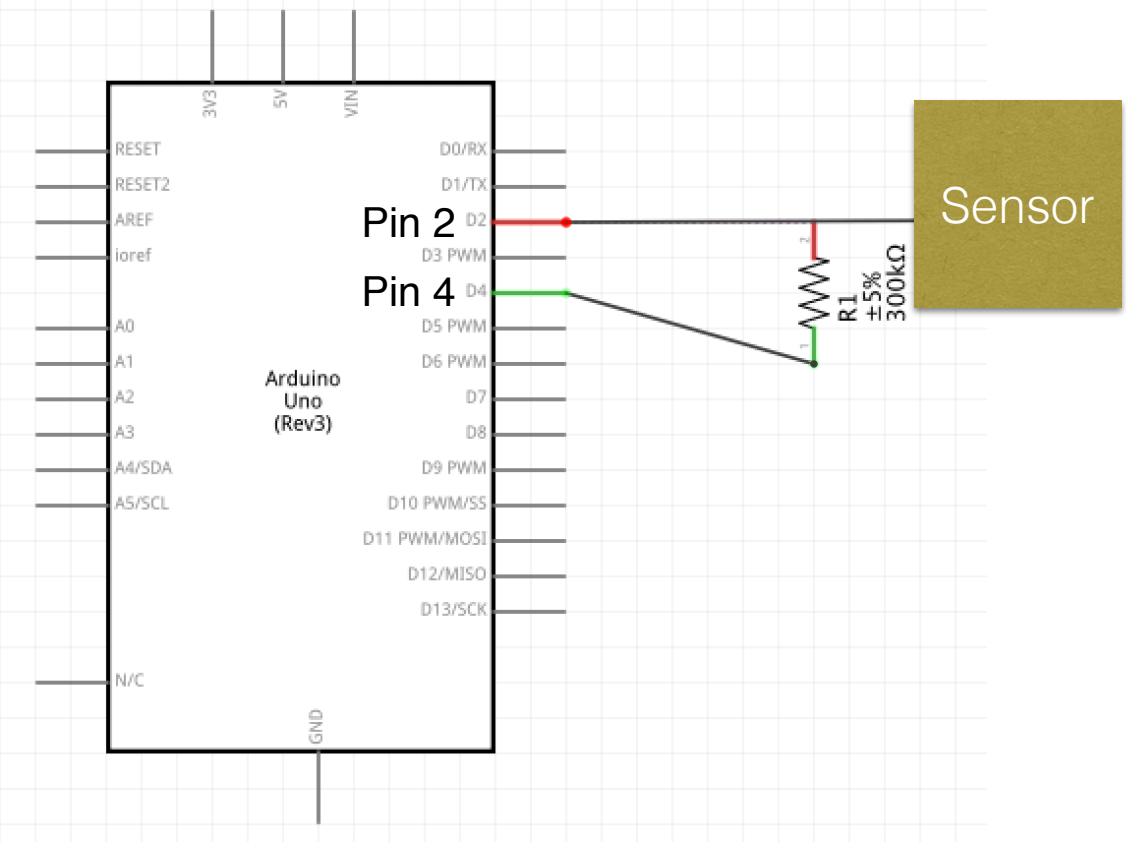


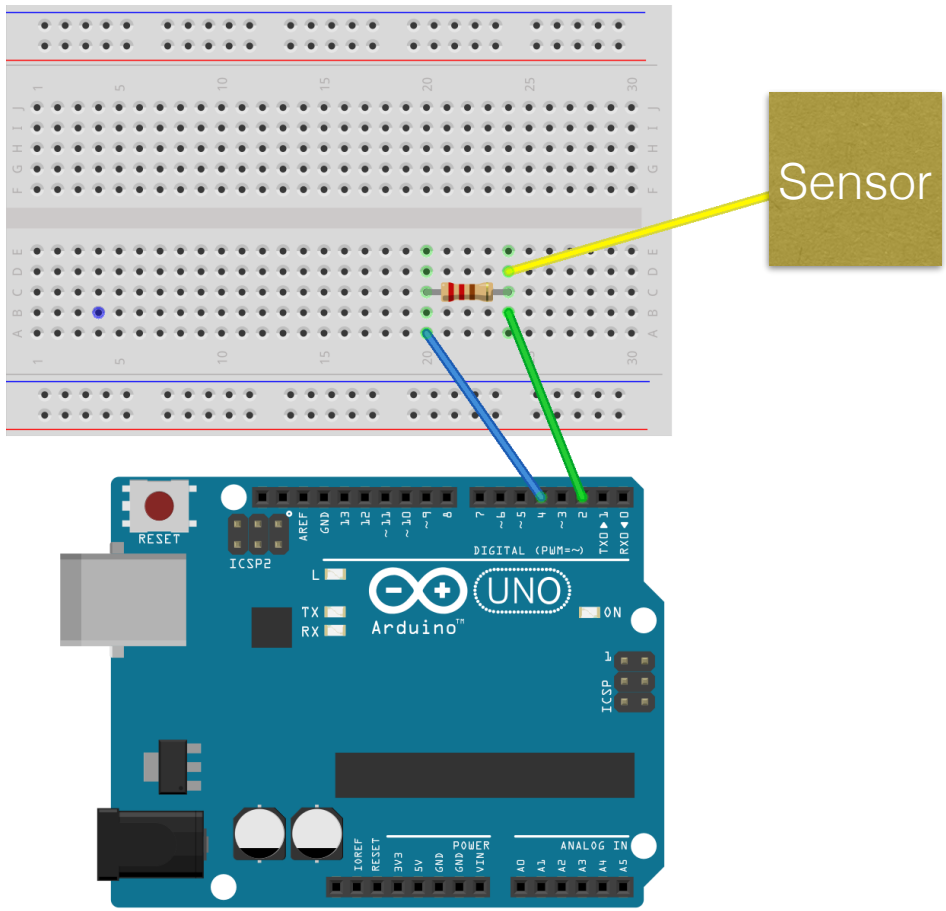
Reference page: https://playground.arduino.cc/Main/CapacitiveSensor/

Capacitive Sensor - Circuit

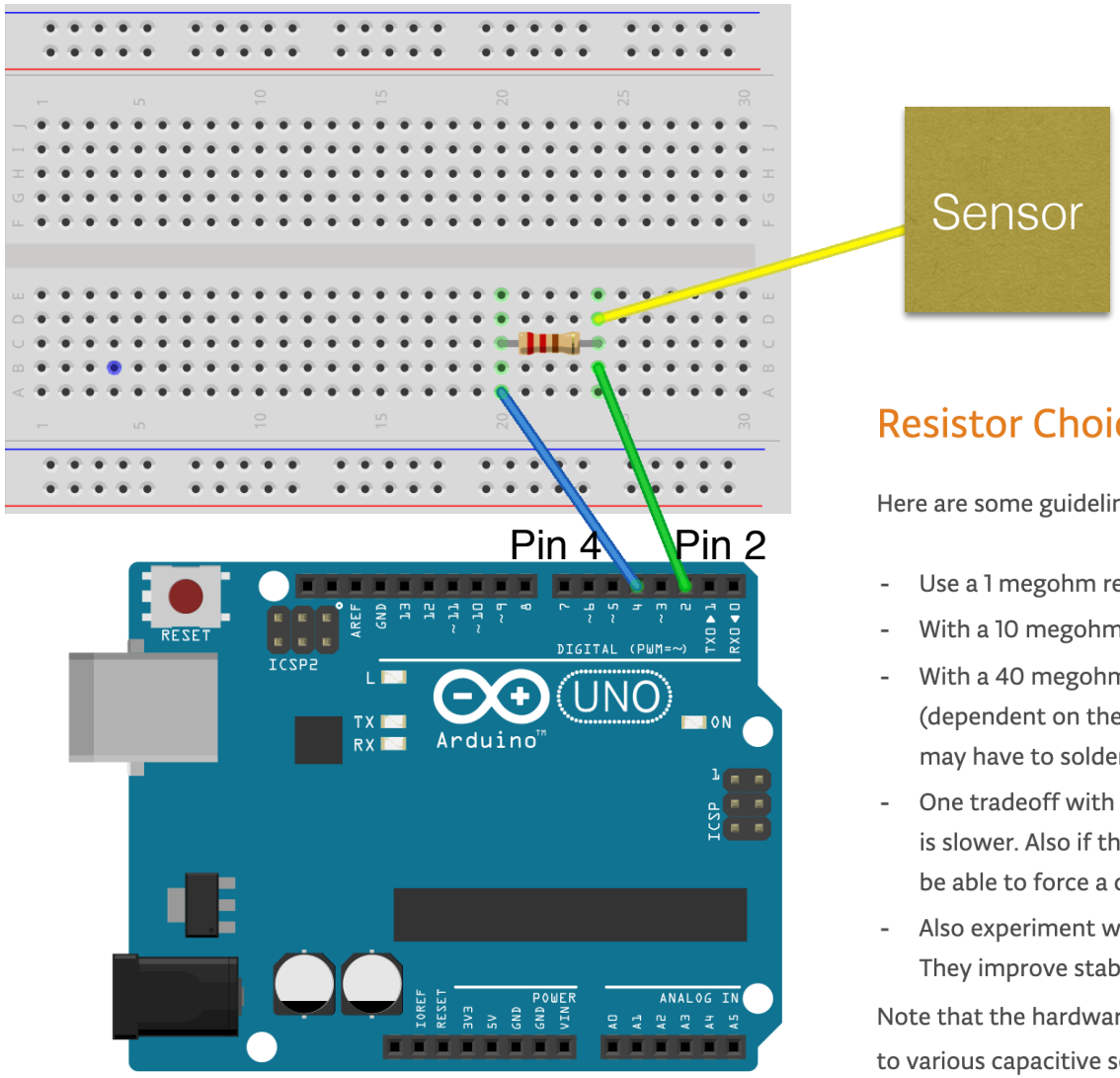


Capacitive Sensor - Circuit





Capacitive Sensor - Circuit



I tested different values:
- 50KOhm
- 300KOhm

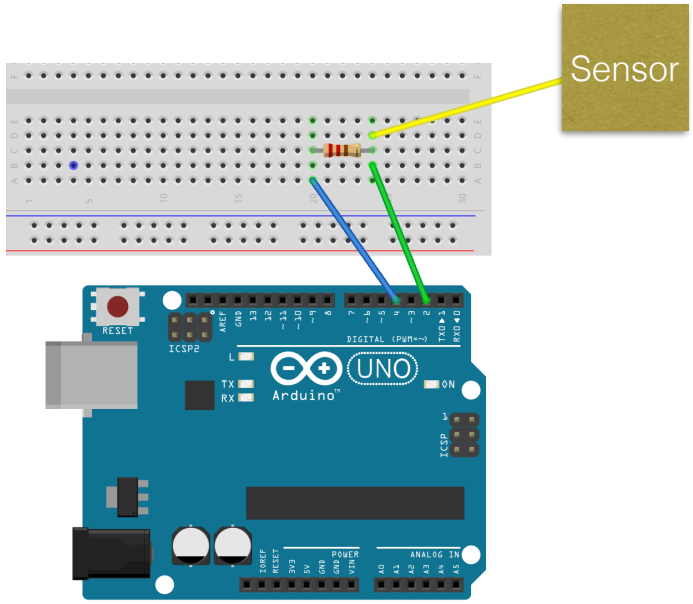
Resistor Choice

Here are some guidelines for resistors but be sure to experiment for a desired response.

- Use a 1 megohm resistor (or less maybe) for absolute touch to activate.
- With a 10 megohm resistor the sensor will start to respond 4-6 inches away.
- With a 40 megohm resistor the sensor will start to respond 12-24 inches away (dependent on the foil size). Common resistor sizes usually end at 10 megohm so you may have to solder four 10 megohm resistors end to end.
- One tradeoff with larger resistors is that the sensor's increased sensitivity means that it is slower. Also if the sensor is exposed metal, it is possible that the send pin will never be able to force a change in the receive (sensor) pin, and the sensor will timeout.
- Also experiment with small capacitors (100 pF - .01 uF) to ground, on the sense pin. They improve stability of the sensor.

Note that the hardware can be set up with one sPin and several resistors and rPin's for calls to various capacitive sensors. See the example sketch.

Capacitive Sensor - Code



You need to install the library:
CapacitiveSensor

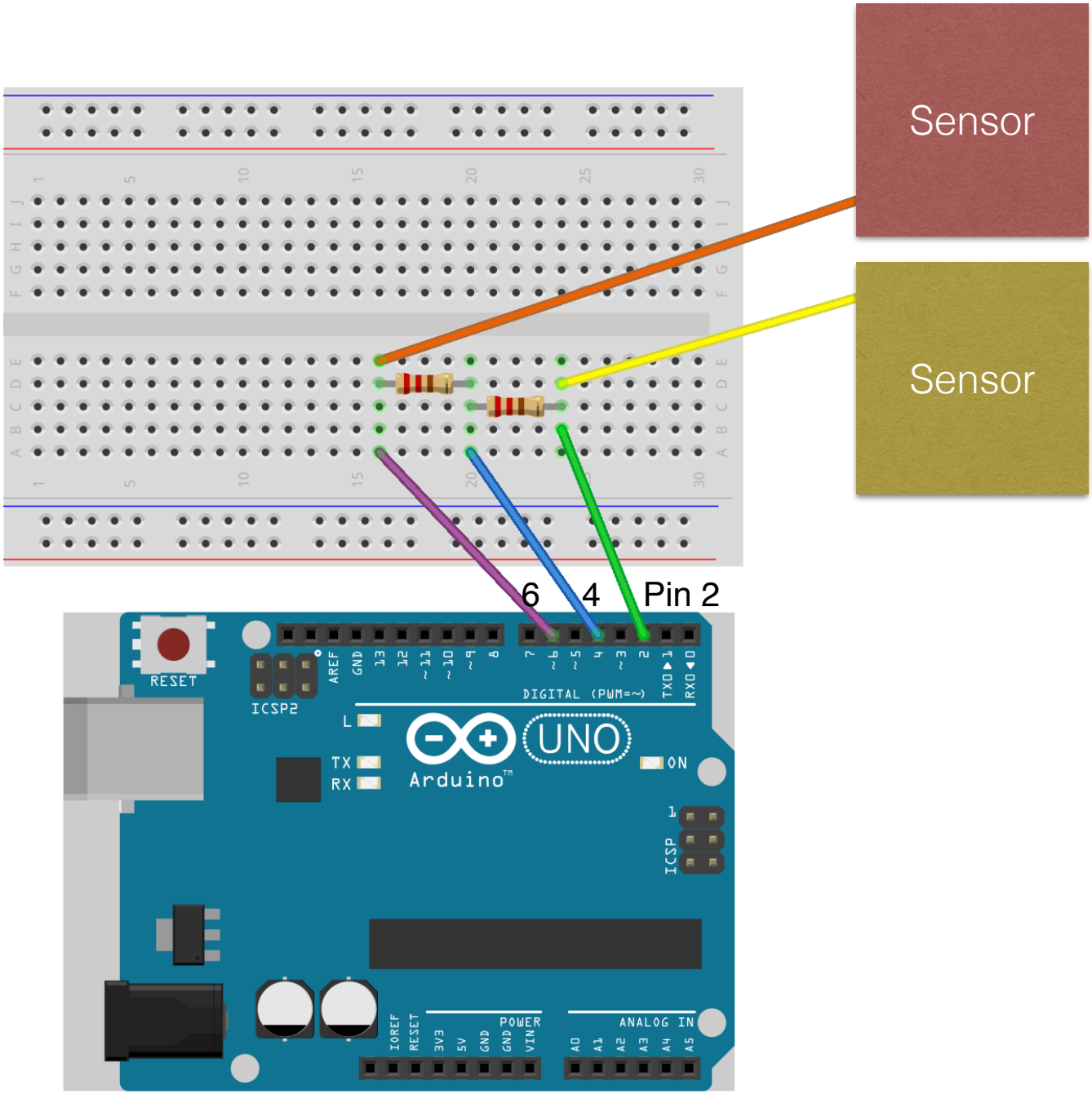
CapacitiveSensor §

```
#include <CapacitiveSensor.h>

/*
 * Modified example from CapitiveSense Library Demo Sketc
 * Paul Badger 2008
 * Uses a high value resistor e.g. 10M between send pin c
 * Resistor effects sensitivity, experiment with values,
 * Receive pin is the sensor pin - try different amounts
 */

CapacitiveSensor   cs_4_2 = CapacitiveSensor(4,2);    // 1

void setup()
{
   cs_4_2.set_CS_AutocaL_Millis(0xFFFFFFFF);  // turn off
   Serial.begin(9600);
}

void loop()
{
    long total1 =  cs_4_2.capacitiveSensor(30);
    Serial.println(total1);                    // print ser
    delay(10);
}
```

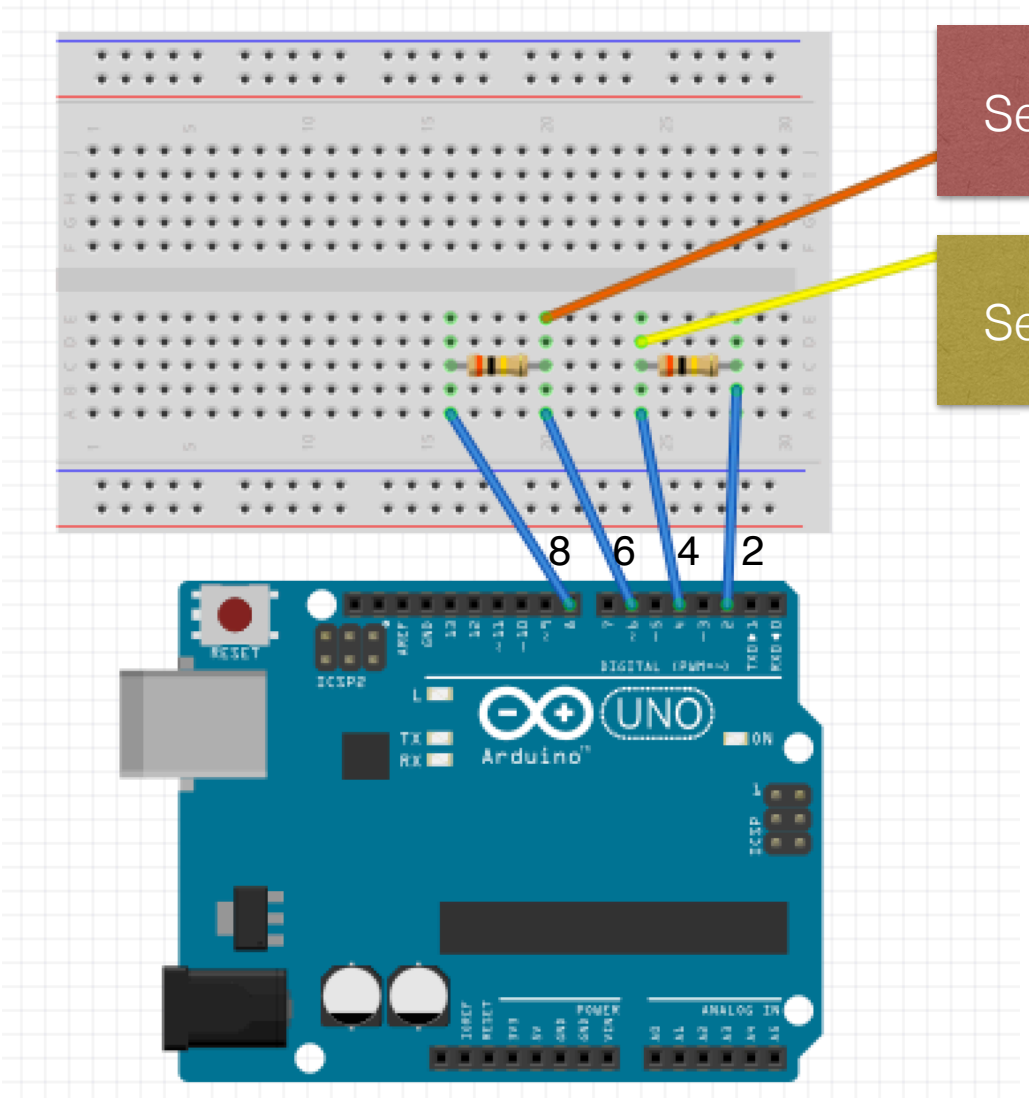
Done Saving.

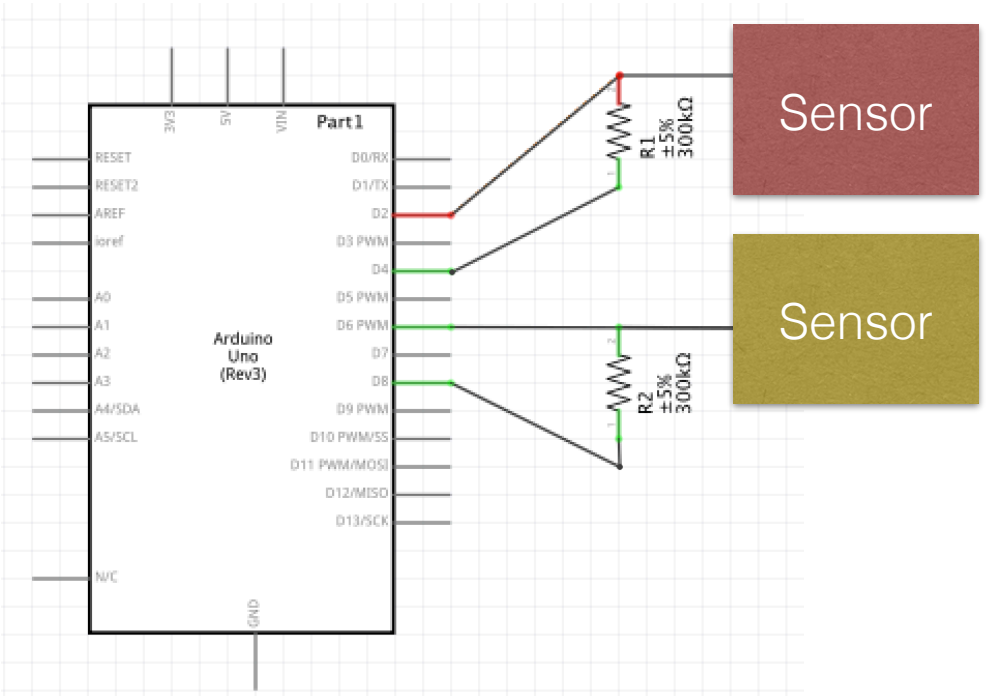
code: CapacitiveSensor

Capacitive Sensors - Circuit

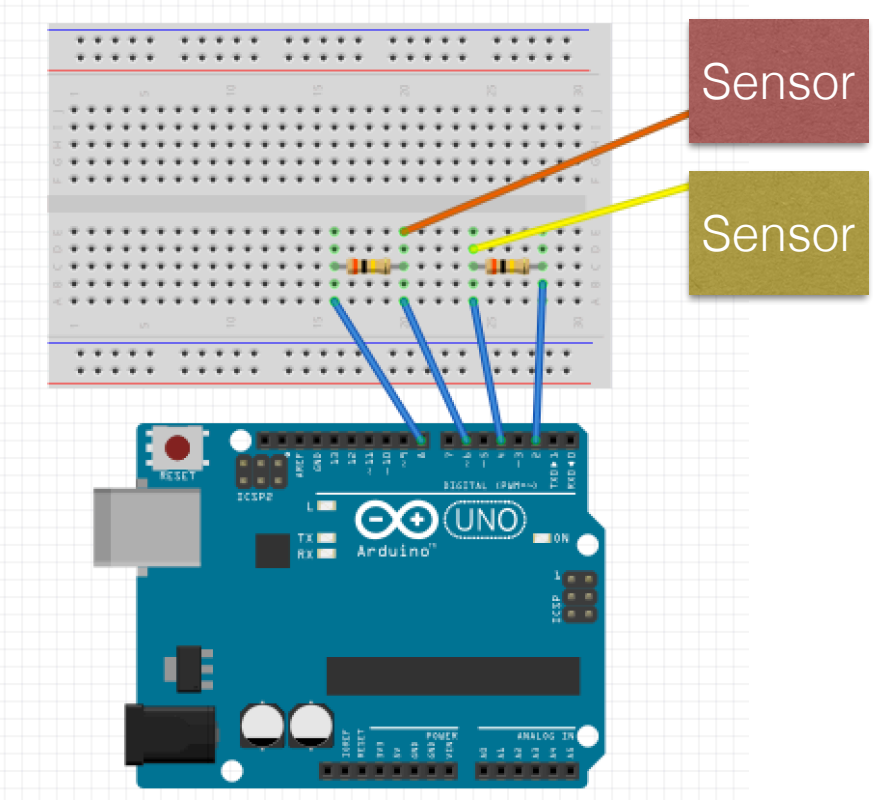


Capacitive Sensors - Circuit





Capacitive Sensor - Circuit



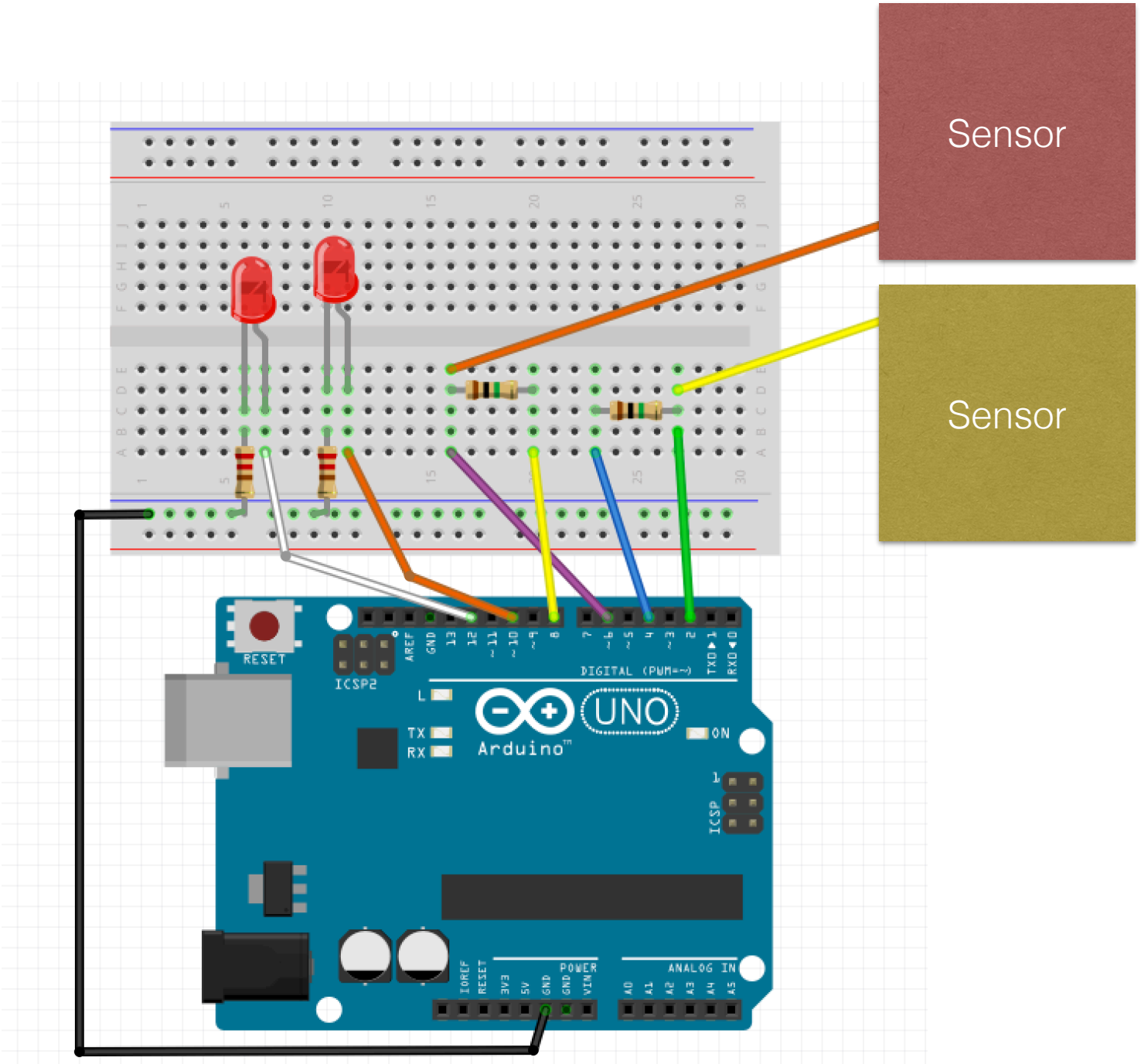
CapacitiveSensor_2

```
#include <CapacitiveSensor.h>

CapacitiveSensor   cs_4_2 = CapacitiveSensor(4,2);
CapacitiveSensor   cs_8_6 = CapacitiveSensor(8,6);

void setup()
{
   Serial.begin(9600);
}

void loop()
{
    long total1 =  cs_4_2.capacitiveSensor(30);
    long total2 =  cs_8_6.capacitiveSensor(30);

    Serial.print(total1);                    // print se
    Serial.print(" ");
    Serial.println(total2);                    // print
    delay(10);                               // arbitrar

}
```

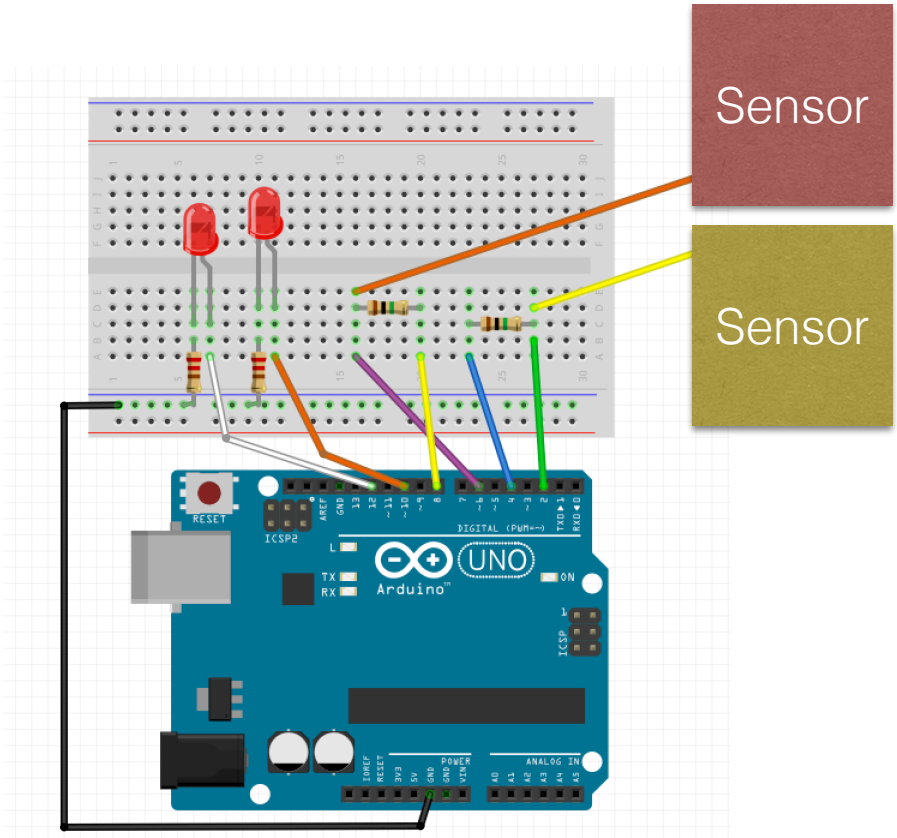
code: CapacitiveSensor_2

Capacitive Sensors and Leds - Circuit

The LEDs are connected to pin10 and pin 12



Capacitive Sensors and Leds - Code

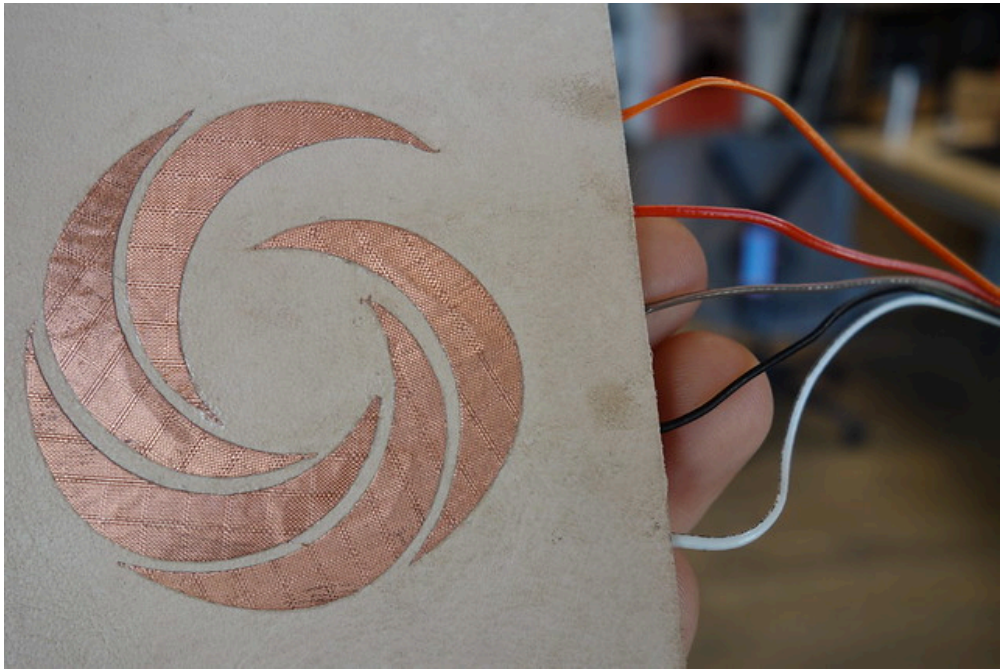


code: CapacitiveSensor_Leds

CapacitiveSensor_Leds

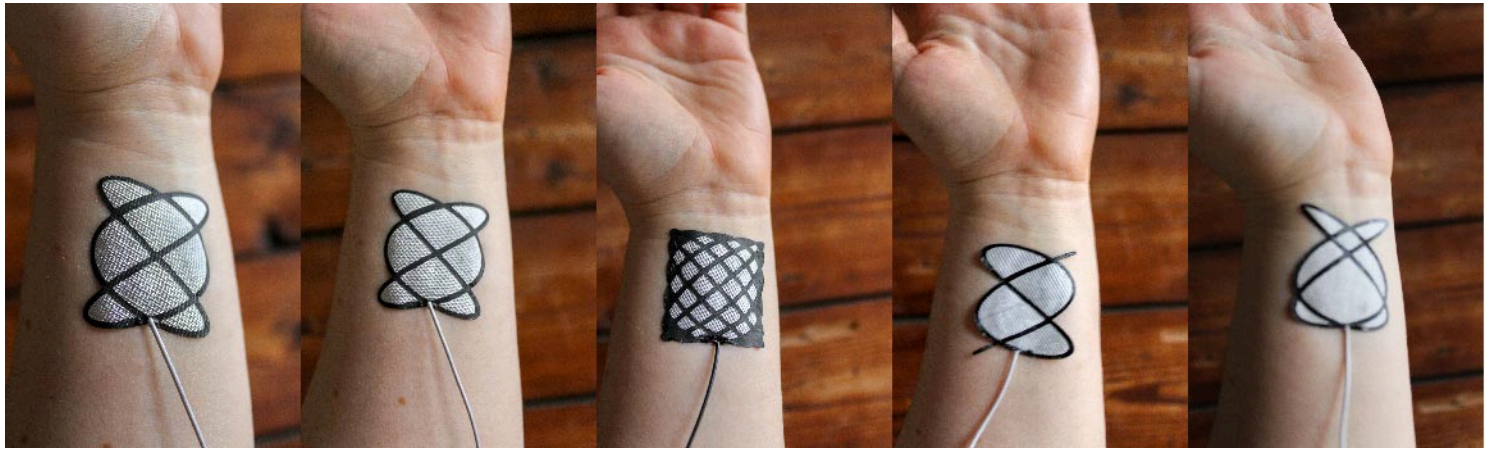
```
void setup()
{
   Serial.begin(9600);
    pinMode(led1_pin, OUTPUT);
    pinMode(led2_pin, OUTPUT);
}

void loop()
{
    long total1 =  cs_4_2.capacitiveSensor(30);
    long total2 =  cs_8_6.capacitiveSensor(30);

    Serial.print(total1);                    // print senso
    Serial.print(" ");
    Serial.println(total2);                   // print sen
    delay(10);                               // arbitrary d

  if(total1 > thr & total2 < thr){

    digitalWrite(led1_pin, HIGH);
    digitalWrite(led2_pin, LOW);

  } else if ( total1 < thr & total2 > thr){

    digitalWrite(led1_pin, LOW);
    digitalWrite(led2_pin, HIGH);

  } else if( total1 > thr & total2 > thr){

    digitalWrite(led1_pin, LOW);
    digitalWrite(led2_pin, LOW);

  } else {

    digitalWrite(led1_pin, HIGH);
    digitalWrite(led2_pin, HIGH);
    }
}
```

Capacitive Sensors



https://www.kobakant.at/DIY/?p=6607

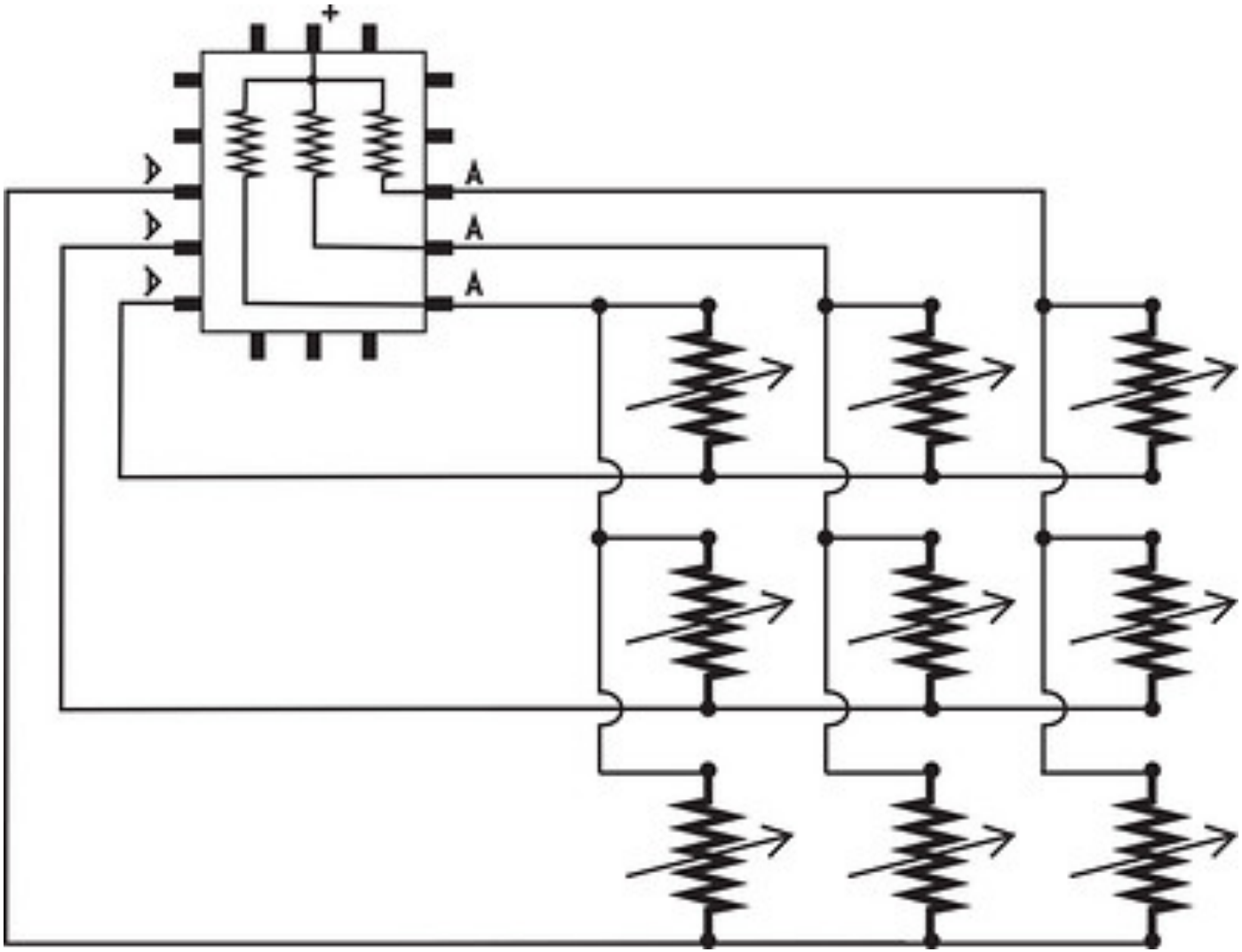
https://class.textile-academy.org/2019/jessica.stanley/assignments/week12/



Conductive Paint.

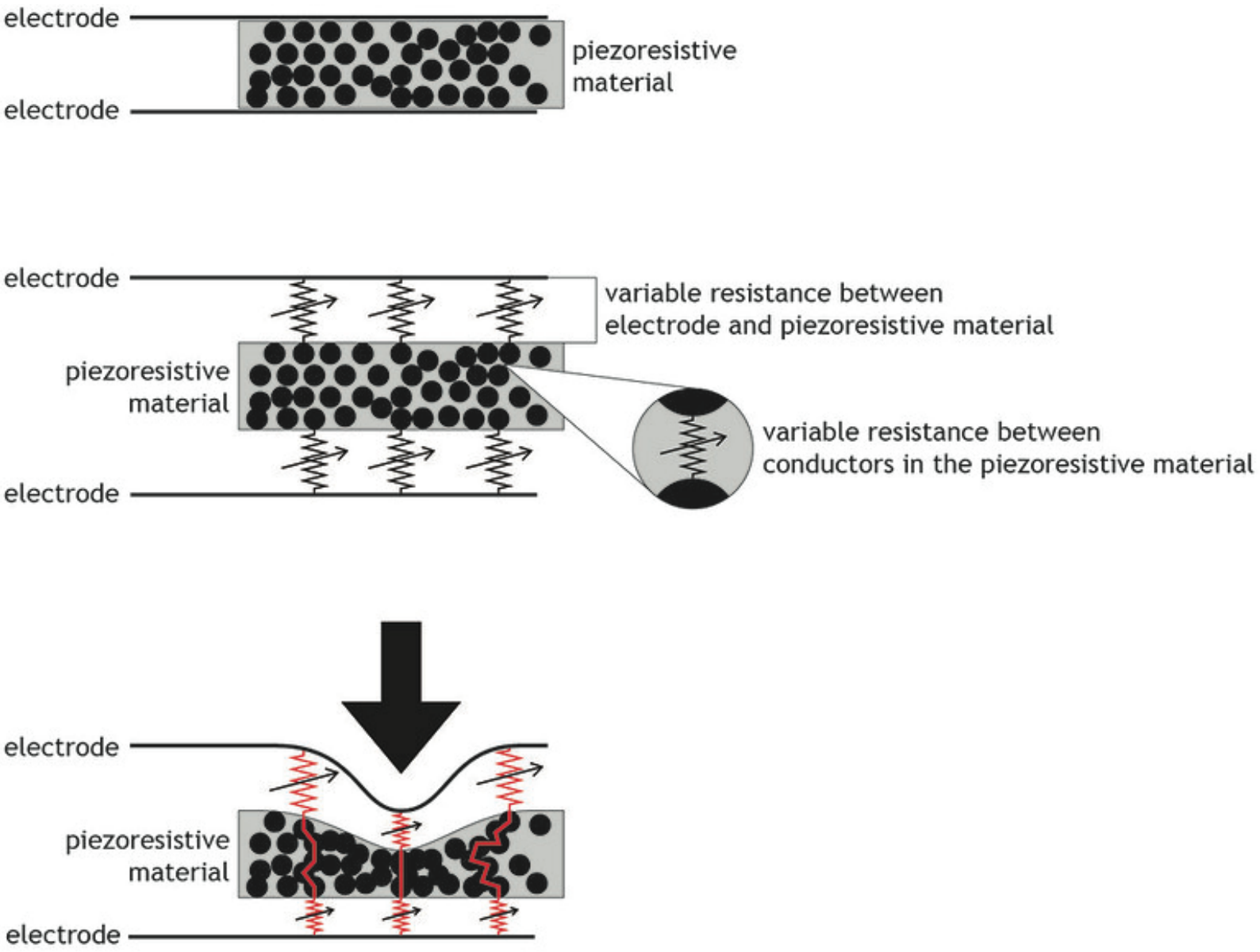
Piezoresistive Matrix

Piezoresistive Matrix



Reference page: https://www.kobakant.at/DIY/?p=7943

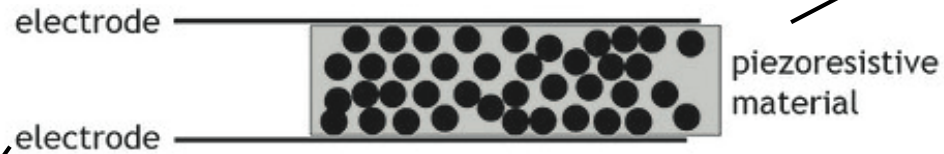
Piezoresistive Matrix



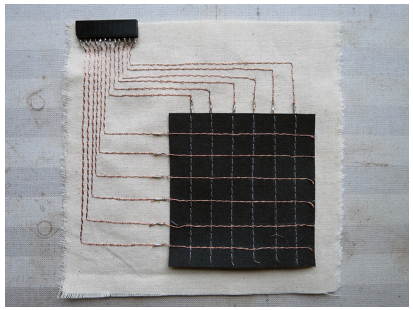
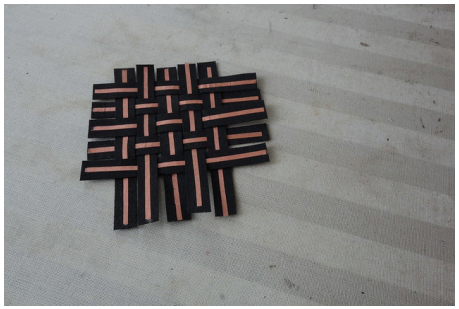
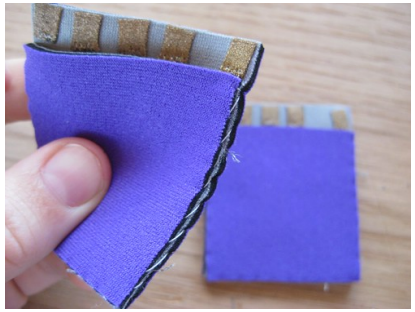
Reference page: https://www.kobakant.at/DIY/?p=7943

Piezoresistive sensors

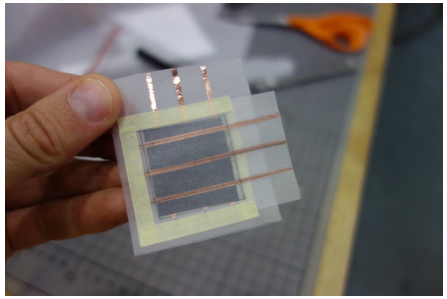
Eeonyx, Velostat, Antistatic foam



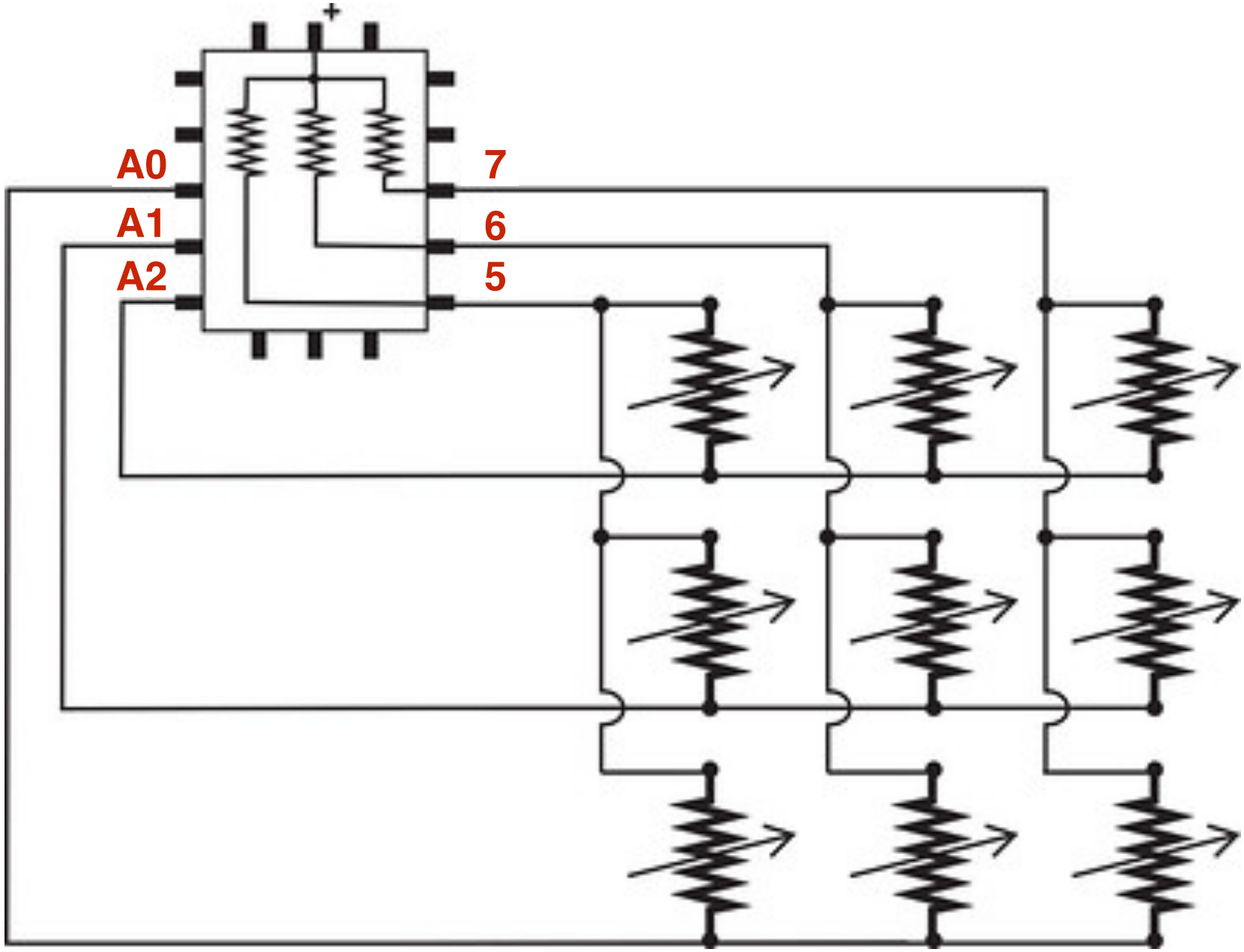
Conductive fabric, Conductive thread,
Copper tale, Copper wires

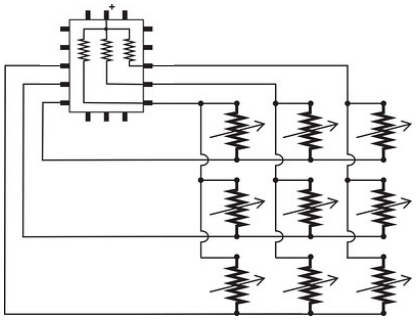




Arduino and Piezoresistive Matrix



Arduino and Piezoresistive Matrix - Code

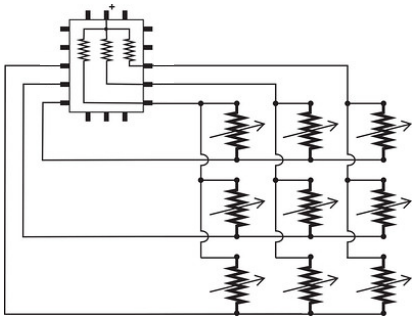


code: Matrix

Matrix

```
//Emma Pareschi- Dec 2020
//Modify example from Pressure Sensor Matrix Code
//parsing through a pressure sensor matrix grid by switching individual
//rows/columns to be HIGH, LOW or INPUT (high impedance) to detect
//location and pressure.
//>> https://www.kobakant.at/DIY/?p=7443

#define numRows 3
#define numCols 3
#define sensorPoints numRows*numCols

int rows[] = {A0, A1, A2};
int cols[] = {5,6,7};
int incomingValues[sensorPoints] = {};

void setup() {
// set all rows and columns to INPUT (high impedance):
for (int i = 0; i < numRows; i++) {
pinMode(rows[i], INPUT_PULLUP);
}
for (int i = 0; i < numCols; i++) {
pinMode(cols[i], INPUT);
}
Serial.begin(9600);
}
```

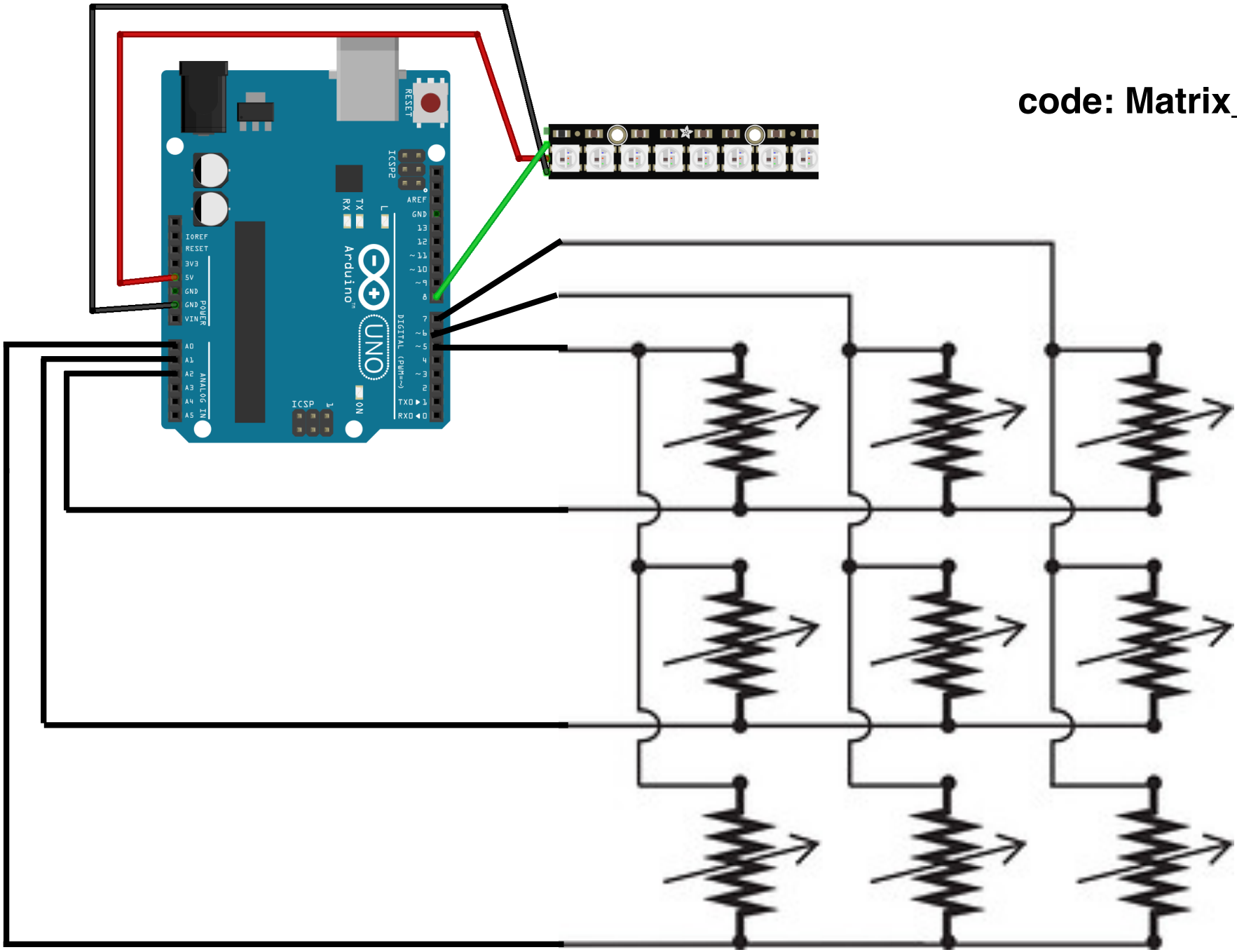
Arduino and Piezoresistive Matrix - Code



code: Matrix

```
void loop() {
for (int colCount = 0; colCount < numCols; colCount++) {
angouts ode(cols[colCount], OUTPUT); // set as OUTPUT
digitalWrite(cols[colCount], LOW); // set LOW

for (int rowCount = 0; rowCount < numRows; rowCount++) {
incomingValues[colCount * numRows + rowCount] = analogRead(rows[rowCount]); /
}// end rowCount

pinMode(cols[colCount], INPUT); // set back to INPUT!

}// end colCount

// Print the incoming values of the grid:
for (int i = 0; i < sensorPoints; i++) {

  Serial.print(incomingValues[i]);
  if (i < (sensorPoints-1)) {
    Serial.print("\t");}
//    Serial.println("");}
  }


Serial.println();
delay(10);
```

Arduino and Piezoresistive Matrix



code: Matrix_neopixel