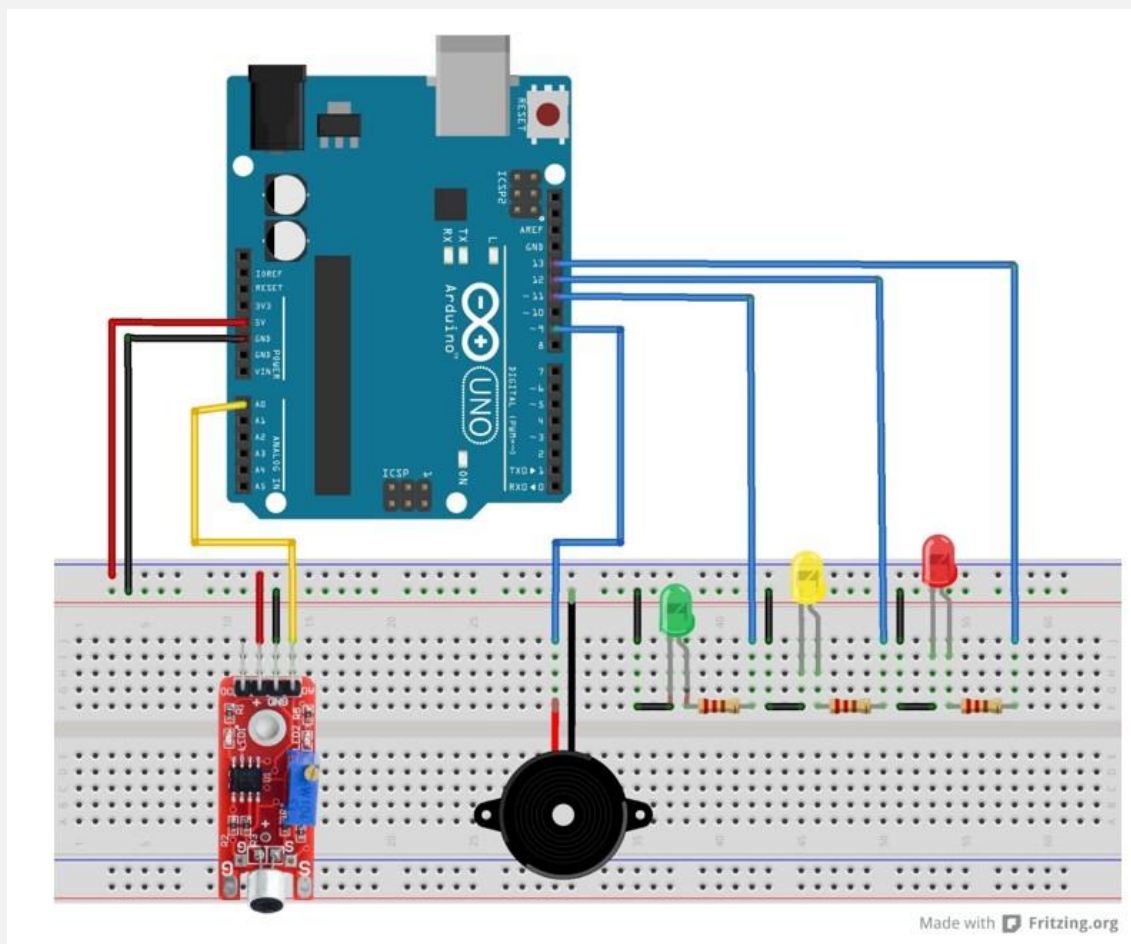


GUÍA PARA CONSTRUIR PASO A PASO UN SEMÁFORO DEL RUIDO EN ARDUINO

Materiales

- Sensor de ruido KY038
- Led rojo
- Led verde
- Led amarillo
- 3 resistencias de 220 Ohm
- Zumbador “Active Buzzer”
- Arduino
- Protoboard
- 13 cables macho-macho

Esquema



Código programación

Práctica de Semáforo Digital: Cuando se supera un primer umbral de ruido, se enciende el led amarillo. Si se supera el segundo umbral, se enciende el led rojo y pita

Cuando el ruido es inferior a ambos umbrales, el led verde permanece encendido

```

const int LEDV = 11; // pin LED Verde
const int LEDA = 12; // pin LED Amarillo
const int LEDR = 13; // pin LED Rojo
const int Piezo = 9; // pin Altavoz, se enciende con PWM, tren de pulsos

const int Microfono = 0; // pin entrada analógica del Micrófono

int val = 0; // variable usada para guardar el valor devuelto por la
salida analógica del micrófono (entre 0 y 1023)

void setup() {
  pinMode(LEDV, OUTPUT); // LEDV es OUTPUT
  pinMode(LEDA, OUTPUT); // LEDA es OUTPUT
  pinMode(LEDR, OUTPUT); // LEDR es OUTPUT
  pinMode(Piezo, OUTPUT); // Piezo es OUTPUT
  pinMode(Microfono, INPUT); // Micrófono es INPUT
}

void loop() {

  val = analogRead(0); // Lee el valor del Micrófono

  if (val<=850) {
    digitalWrite(LEDV, HIGH); // Enciende el LED Verde
    digitalWrite(LEDA, LOW); // Apaga LED Amarillo
    digitalWrite(LEDR, LOW); // Apaga LED Rojo
    analogWrite(Piezo, LOW); // Apaga Piezo. Se puede escribir LOW ó 0
  }
  else if ((val>850) and (val<860)){
    digitalWrite(LEDA, HIGH); // Enciende el LED Amarillo
    digitalWrite(LEDV, LOW); // Apaga LED Verde
    digitalWrite(LEDR, LOW); // Apaga LED Rojo
    analogWrite(Piezo, LOW); // Apaga Piezo. Se puede escribir LOW ó 0
    delay (100); // Retardo de 1 segundo
  }
}

```

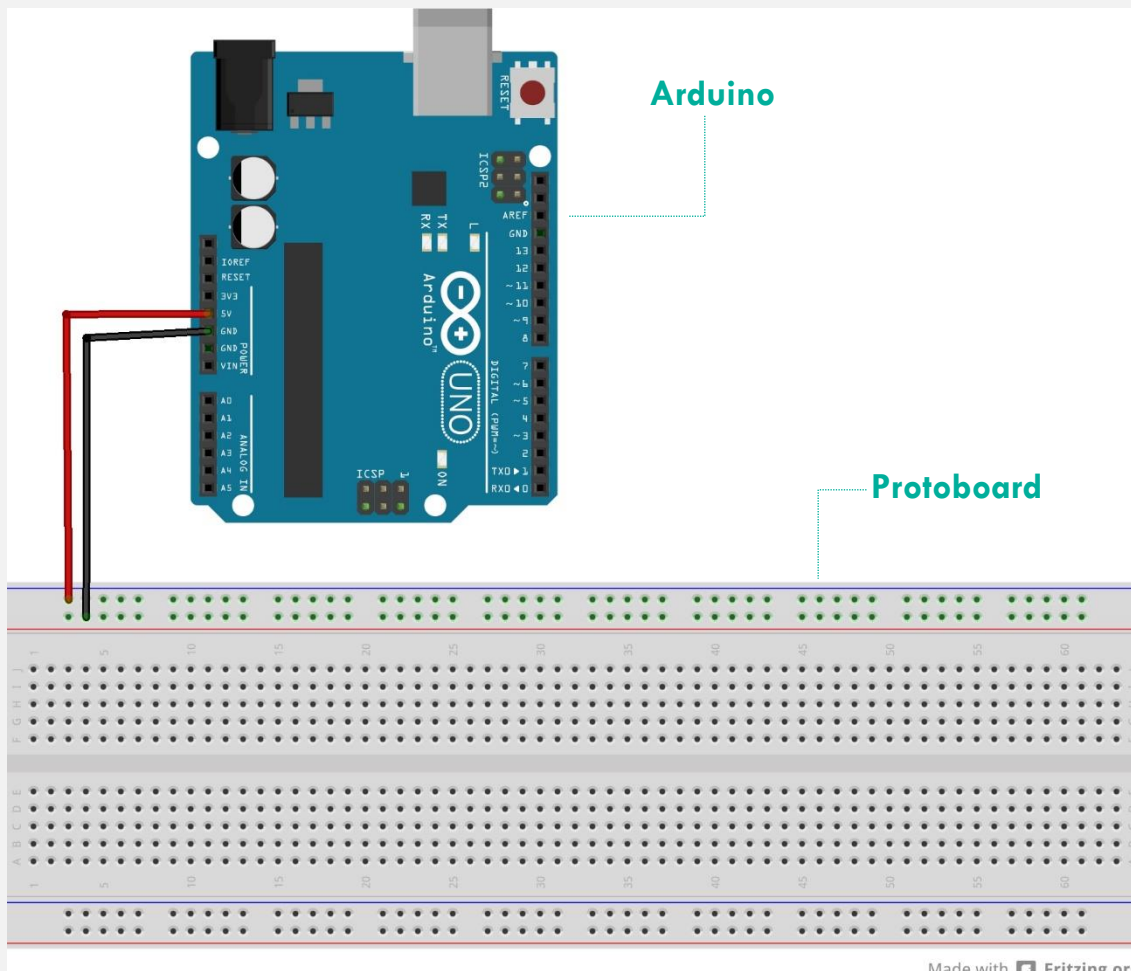
```
else{
  digitalWrite(LED_R, HIGH); // Enciende el LED Rojo
  analogWrite(Piezo, HIGH); // Hace pitar el Altavoz. Se puede escribir
HIGH ó 1023
  digitalWrite(LED_V, LOW); // Apaga LED Verde
  digitalWrite(LED_A, LOW); // Apaga LED Amarillo

  delay (1000); // Retardo de 1 segundo para mantener encendido el led
rojo y pitando el piezo
}
}
```

INSTRUCCIONES PASO A PASO

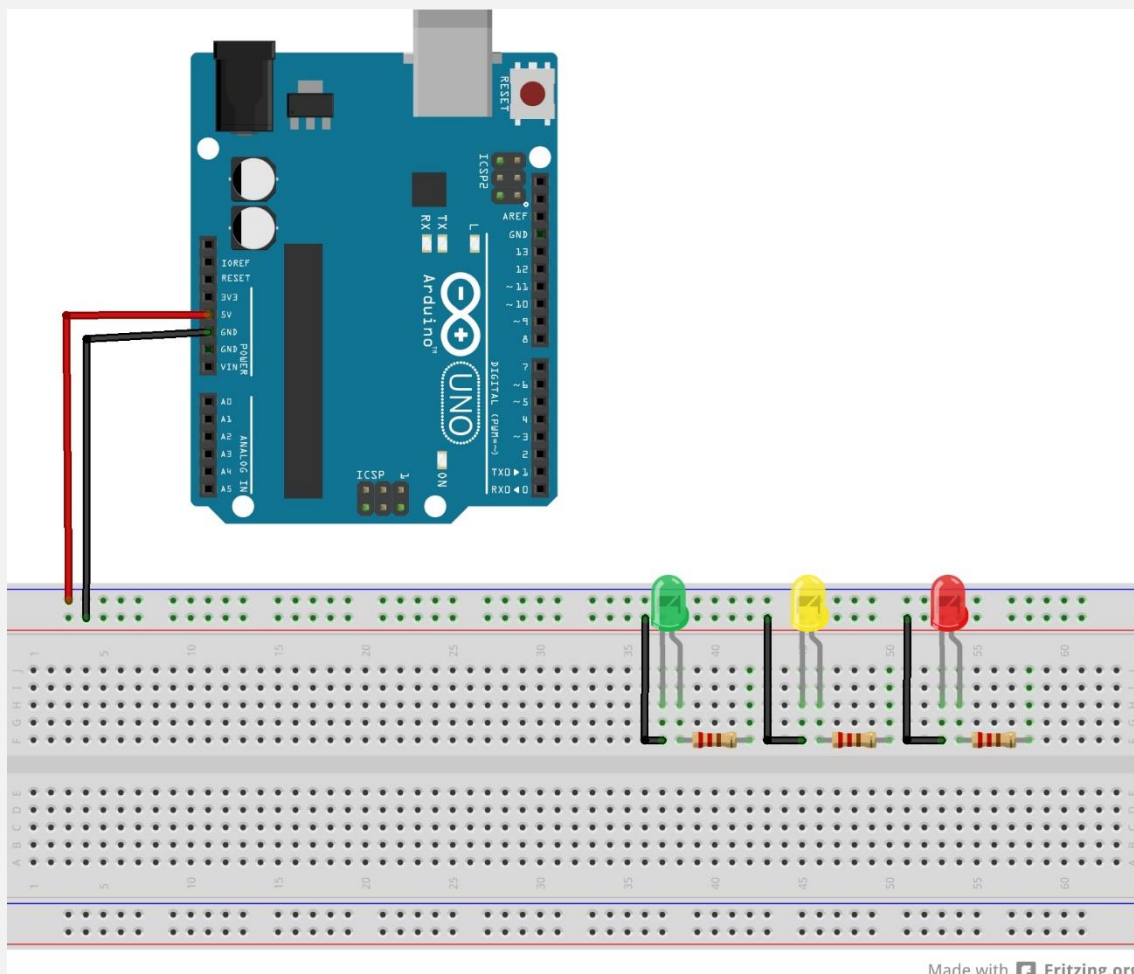
PASO 1: Conexión Arduino a Protoboard

- 1 Unimos con un cable macho-macho los 5V de la placa (pin indicado en esta) con la fila de los 5V.
- 2 Unimos con un cable macho-macho masa (GND) de la placa (pin indicado en esta) con la fila de GND.

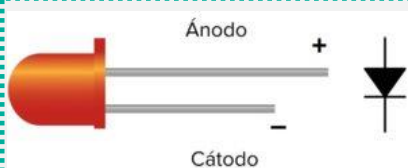


PASO 2: LAS LUCES DEL SEMÁFORO

- 1 Conectamos en la protoboard los LEDs siguiendo el orden de un semáforo y dejando un espacio de unos cuatro agujeros entre ellos. La patilla larga del led es ánodo y la patilla corta el cátodo*.
- 2 Una vez colocados los LEDs. En las columnas de los ánodos (patilla larga) colocamos una resistencia por cada led
- 3 Unimos con tres cable macho-macho la fila de GND con cada una de las columnas donde hayamos colocado los 3 cátodos de los LEDs.



*Recuerda

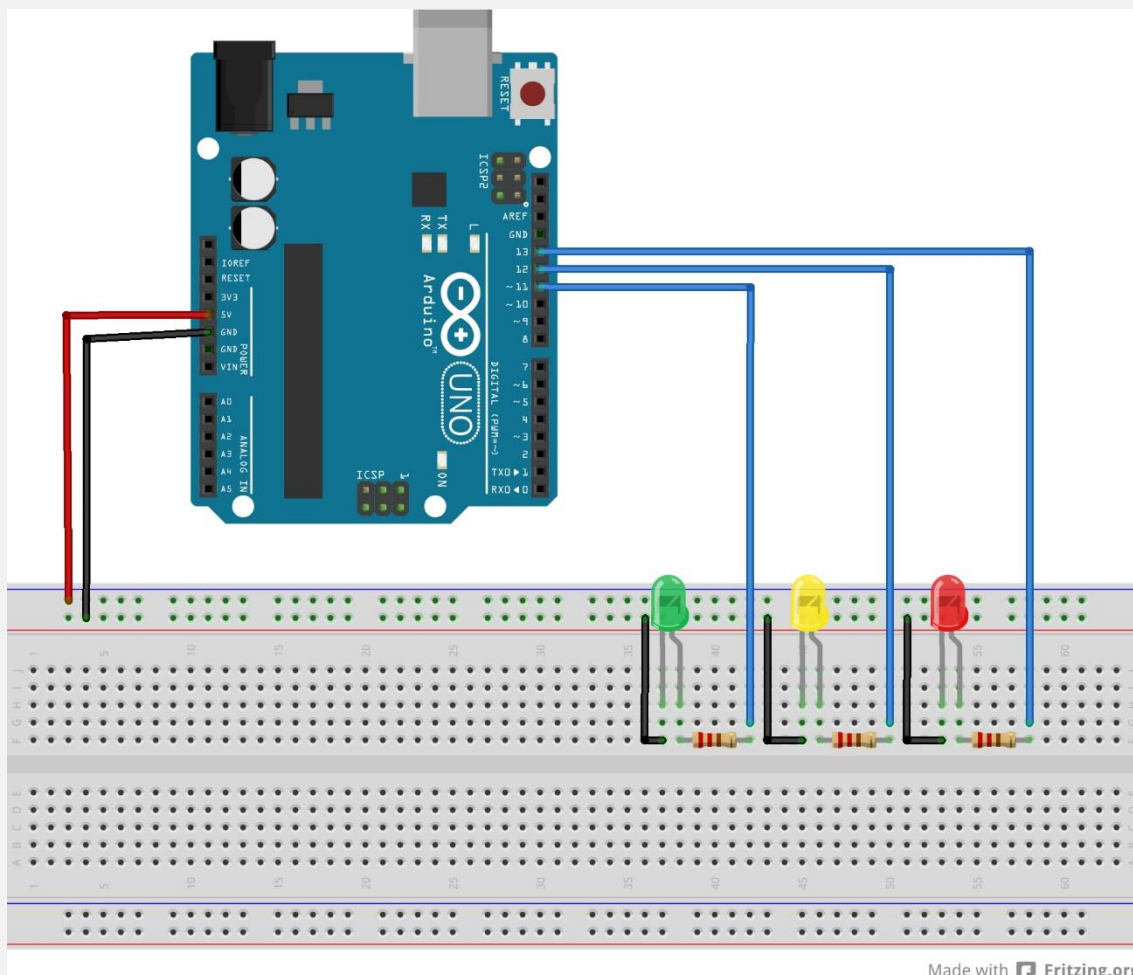


Ánodo, es donde se produce la oxidación (pierde electrones)

Cátodo es donde se produce la reducción (gana electrones)

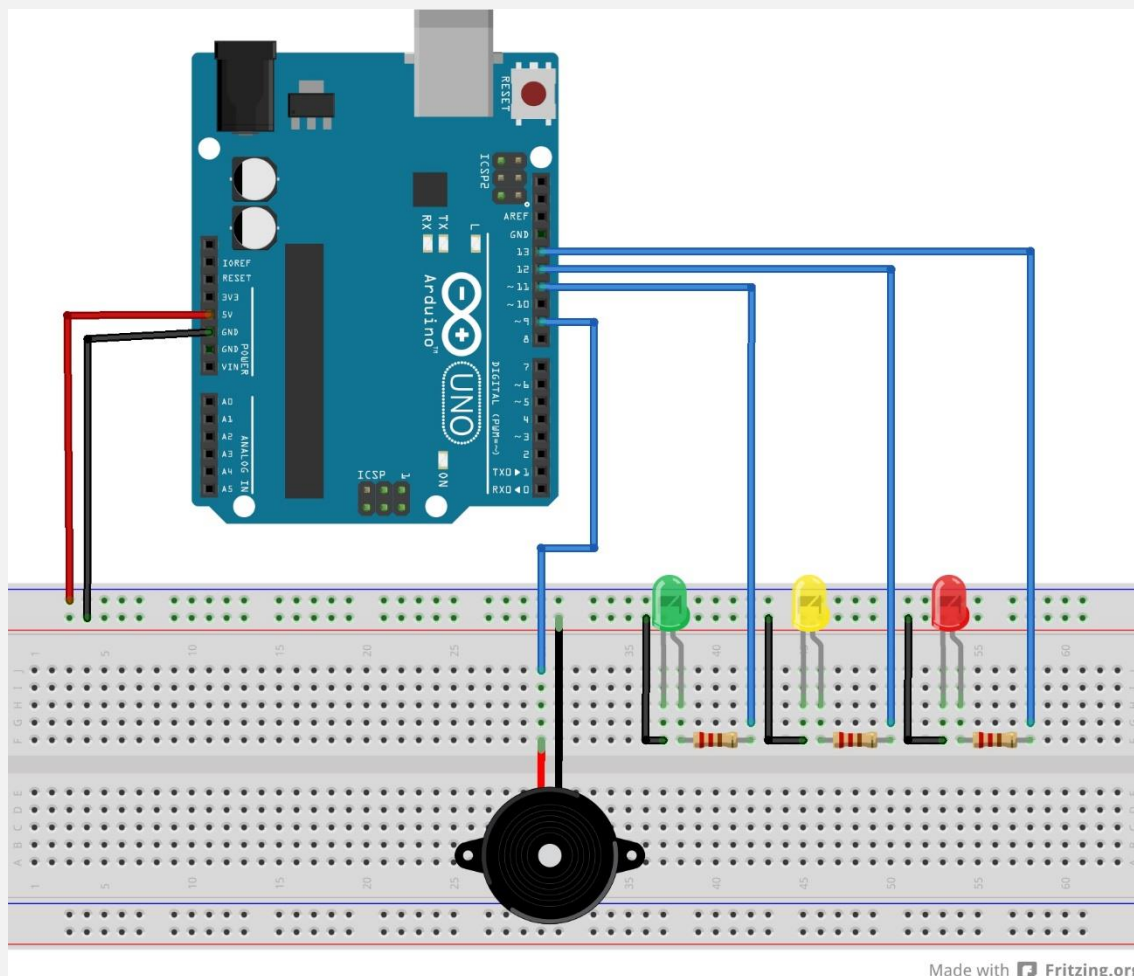
PASO 4

- 1 Unimos con 3 cables macho-macho las columnas donde hayamos puesto el otro terminal de la resistencia con los pines 11, 12 y 13 de la placa Arduino.



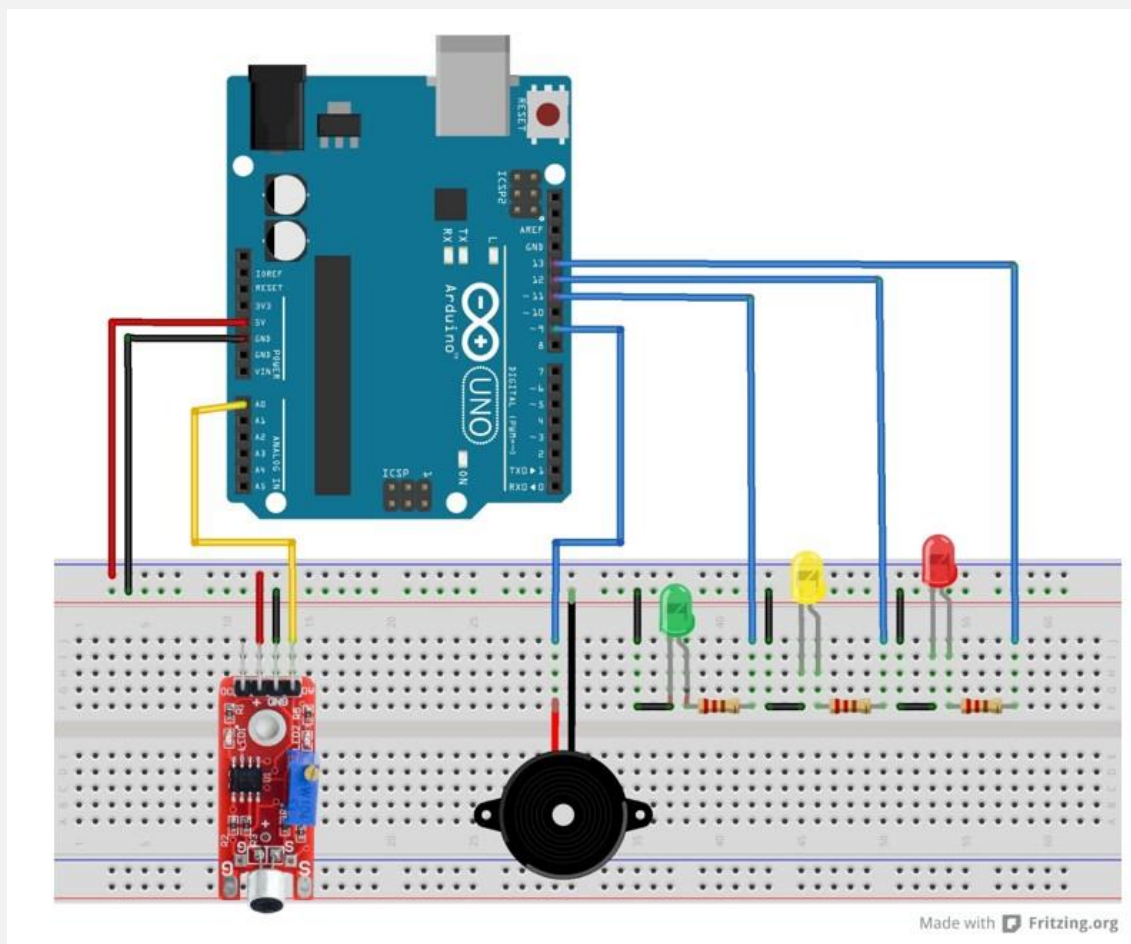
PASO 5

- 1 Colocamos el altavoz piezoeléctrico teniendo en cuenta cual es la patilla con el símbolo + que indica la pegatina que tiene
- 2 Conectamos con un cable macho-macho la columna donde hayamos colocado la patilla positiva del altavoz (la de la pegatina con el +) con el pin 9 de la placa
- 3 Conectamos con un cable macho-macho la columna donde hayamos colocado la patilla negativa del altavoz

Made with  Fritzing.org

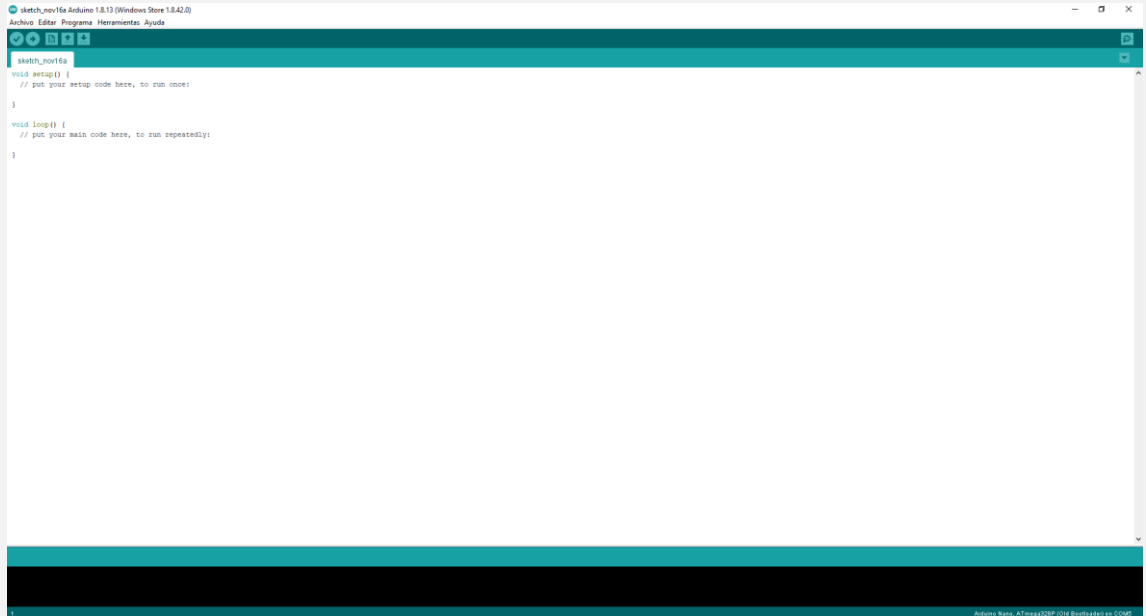
PASO 6

- 1 Conectamos el micrófono con la protoboard.
- 2 Conectamos con un cable macho-macho la columna del pin del micrófono marcado como AO con el pin A0 de la placa Arduino.
- 3 Conectamos con un cable macho-macho la fila de GND con el pin marcado como GND del micrófono.
- 4 Conectamos con un cable macho-macho la fila de 5V con el pin marcado como + del micrófono.



PASO 7

- 1 Conectamos el Arduino por USB al ordenador
- 2 Iniciamos Arduino



- 3 Copiamos el siguiente código:

```
const int LEDV = 11; // pin LED Verde
const int LEDA = 12; // pin LED Amarillo
const int LEDR = 13; // pin LED Rojo
const int Piezo = 9; // pin Altavoz, se enciende con PWM, tren de pulsos

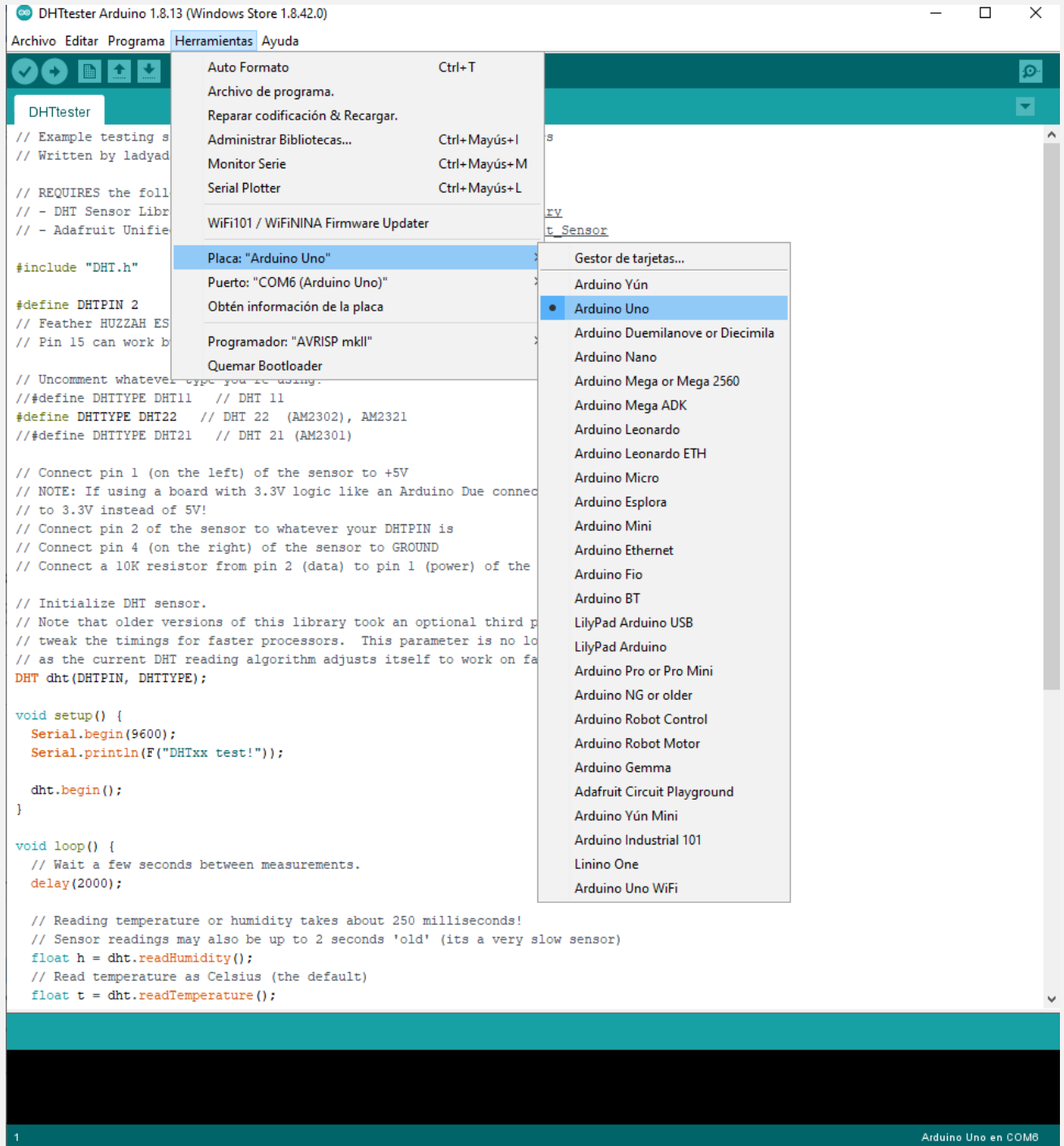
const int Microfono = 0; // pin entrada analógica del Micrófono

int val = 0; // variable usada para guardar el valor devuelto por
la salida analógica del micrófono (entre 0 y 1023)

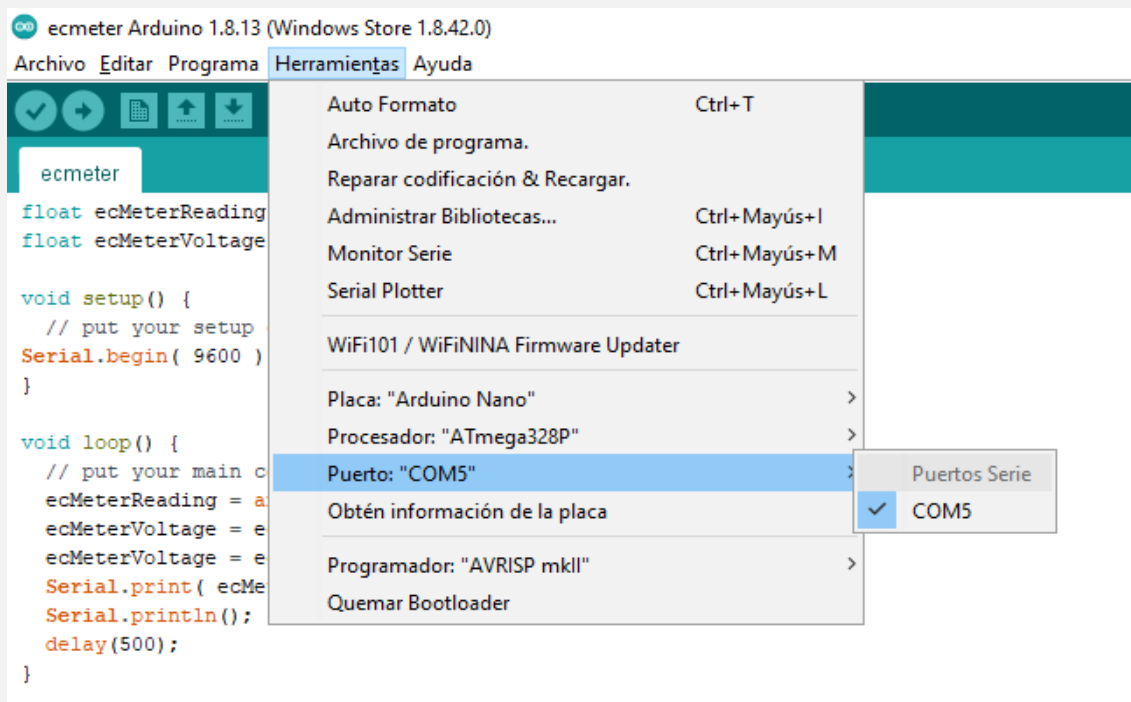
void setup() {
  pinMode(LEDV, OUTPUT); // LEDV es OUTPUT
  pinMode(LEDA, OUTPUT); // LEDA es OUTPUT
  pinMode(LEDR, OUTPUT); // LEDR es OUTPUT
  pinMode(Piezo, OUTPUT); // Piezo es OUTPUT
  pinMode(Microfono, INPUT); // Micrófono es INPUT
}

void loop() {

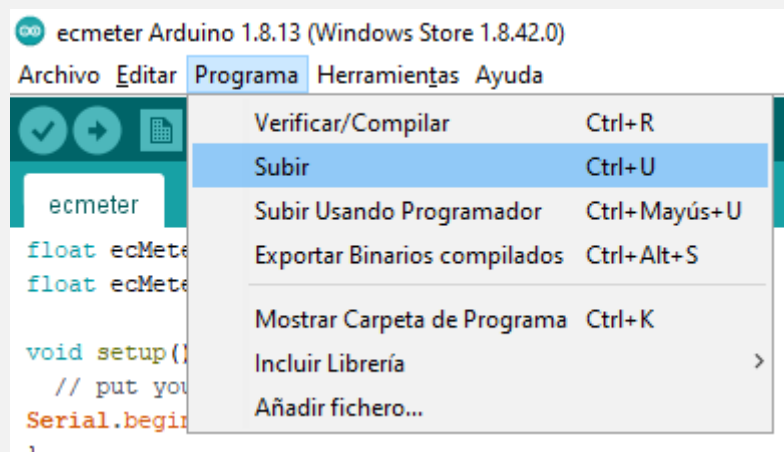
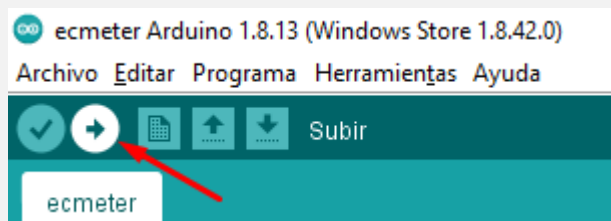
  val = analogRead(0); // Lee el valor del Micrófono
```

- 5 Elegimos el puerto de comunicación designado por Windows a nuestro Arduino (Herramientas>Puerto):



- 6 Compilamos y cargamos el programa en nuestro Arduino (Pulsamos la flecha que apunta hacia la derecha o Programa>Subir)



PASO 8 Ya podremos comprobar si funciona