

PDIH - PRÁCTICA 3

Jorge Sánchez - Carlos Guasp



UNIVERSIDAD DE GRANADA

ÍNDICE

Ejercicio 1: Implementar el programa de parpadeo de LED, ampliándolo para que encienda y apague alternativamente tres LEDs (uno rojo, otro amarillo y otro verde), conectados a las salidas digitales 11, 12 y 13 del Arduino, a un intervalo de 1.5 segundos. Simular primero el prototipo en Tinkercad y sacar captura de pantalla del prototipo (esquema) para incluirla en el documento de la práctica. **2**

Ejercicio 2: Partir del programa de parpadeo de LEDs anterior y ampliarlo con las modificaciones necesarias para que se encienda el LED rojo solo cuando se pulse un interruptor conectado a la entrada digital 7, y en ese momento se apaguen los LEDs amarillo y verde. Simular primero el prototipo en Tinkercad y sacar captura de pantalla del prototipo (esquema) para incluirla en el documento de la práctica. A continuación, cargar el programa en el Arduino físico para comprobar que funciona correctamente (incluir foto en el documento de la práctica) **4**

Ejercicio 3: Secuencia de LEDs, encendiendo y apagando 4 LEDs secuencialmente, de forma similar a las lucecitas de "El coche fantástico": <https://bit.ly/3Lu7YNz> **7**

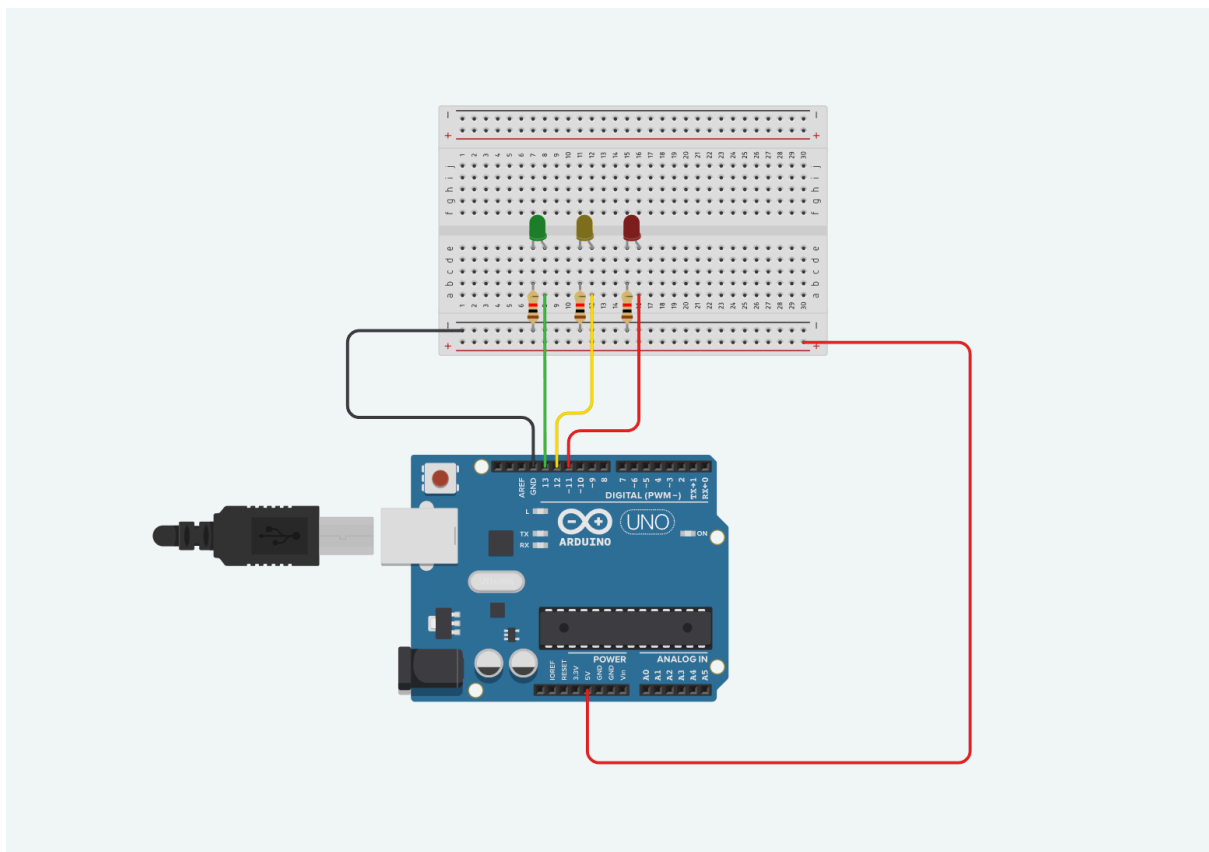
Ejercicio 4: Detector de la distancia a un objeto (usar el buzzer para hacer sonar un pitido en función de la distancia detectada por el sensor de ultrasonidos). **9**

Ejercicio 6: Implementar un proyecto en el que se active un motor (DC Motor o Servo motor) cuando se pulse un pulsador. **11**

Ejercicio 1: Implementar el programa de parpadeo de LED, ampliándolo para que encienda y apague alternativamente tres LEDs (uno rojo, otro amarillo y otro verde), conectados a las salidas digitales 11, 12 y 13 del Arduino, a un intervalo de 1.5 segundos. Simular primero el prototipo en Tinkercad y sacar captura de pantalla del prototipo (esquema) para incluirla en el documento de la práctica.

A continuación, cargar el programa en el Arduino físico para comprobar que funciona correctamente (incluir foto en el documento de la práctica).

Para el desarrollo de este ejercicio montamos en TinkerCad el siguiente prototipo:



Además realizamos el siguiente código:

```
const int ledRojo=11;
const int ledAmarillo=12;
const int ledVerde=13;

void setup() {

//Configurar los pines como salidas
pinMode(ledRojo, OUTPUT);
pinMode(ledAmarillo, OUTPUT);
pinMode(ledVerde,OUTPUT);
```

```

}

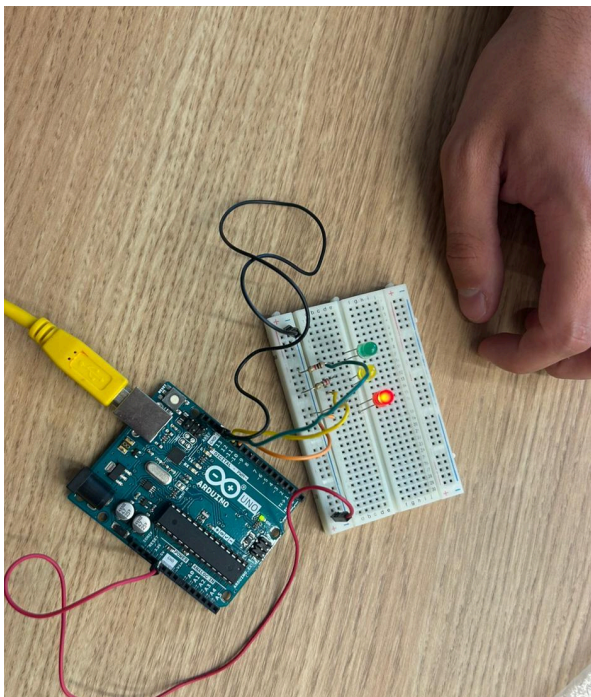
void loop() {
//Encender el LED rojo, apagar los otros
digitalWrite(ledRojo, HIGH);
digitalWrite(ledAmarillo, LOW);
digitalWrite(ledVerde, LOW);
delay(1500);

//Encender el LED amarillo, apagar los otros
digitalWrite(ledRojo, LOW);
digitalWrite(ledAmarillo, HIGH);
digitalWrite(ledVerde, LOW);
delay(1500);

//Encender el LED verde, apagar los otros
digitalWrite(ledRojo, LOW);
digitalWrite(ledAmarillo, LOW);
digitalWrite(ledVerde, HIGH);
delay(1500);
}

```

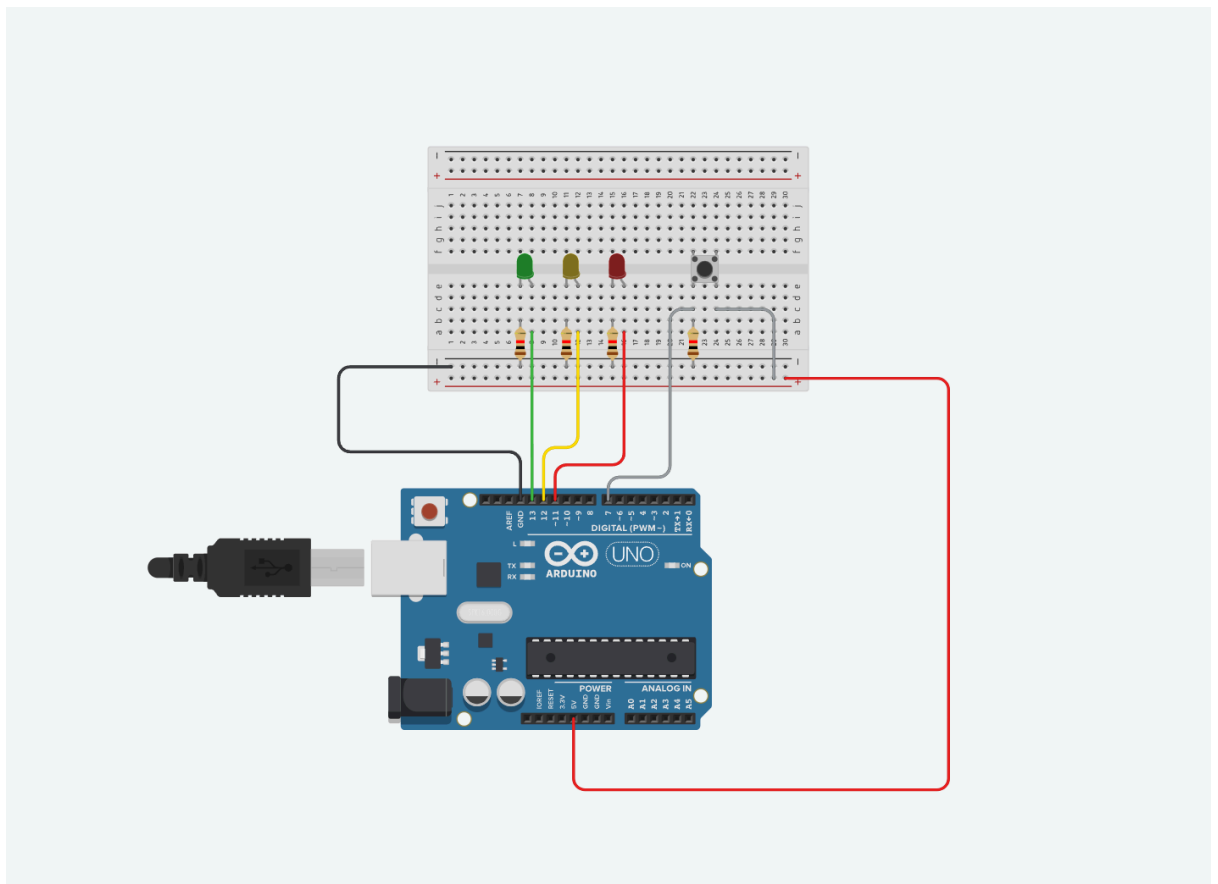
Tras eso lo montamos en físico y obtuvimos este montaje:



Además hay un vídeo del funcionamiento llamado Ej1.mp4

Ejercicio 2: Partir del programa de parpadeo de LEDs anterior y ampliarlo con las modificaciones necesarias para que se encienda el LED rojo solo cuando se pulse un interruptor conectado a la entrada digital 7, y en ese momento se apaguen los LEDs amarillo y verde. Simular primero el prototipo en Tinkercad y sacar captura de pantalla del prototipo (esquema) para incluirla en el documento de la práctica. A continuación, cargar el programa en el Arduino físico para comprobar que funciona correctamente (incluir foto en el documento de la práctica)

Para el segundo ejercicio también repetimos el mismo proceso, de diseñar el prototipo y el código antes de montarlo en el Arduino físico.



El código correspondiente al ejercicio es:

```
int val = 0;
const int ledRojo=11;
const int ledAmarillo=12;
const int ledVerde=13;
const int pulsador=7;
```

```

void setup()
{
    pinMode(ledVerde, OUTPUT);
    pinMode(ledAmarillo, OUTPUT);
    pinMode(ledRojo, OUTPUT);
    pinMode(pulsador, OUTPUT);
}

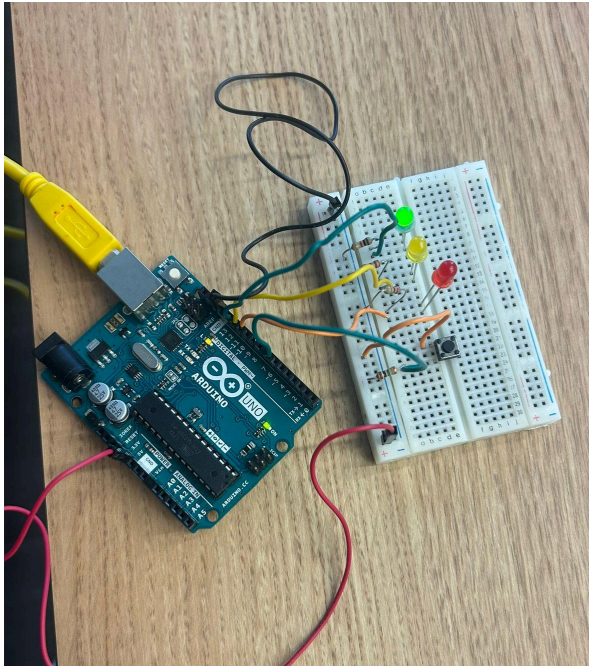
void loop()
{
    // Leer el valor del input
    val = digitalRead(7);

    // Si se suelta el pulsador empieza el ciclo en el verde
    if (val == LOW) {
        digitalWrite(ledRojo, LOW);
        digitalWrite(ledVerde, HIGH);
        digitalWrite(ledAmarillo, LOW);

        delay(1500);
        digitalWrite(ledAmarillo, HIGH);
        digitalWrite(ledVerde, LOW);
        delay(1500);
    }
    // Si el pulsador esta presionado se enciende el rojo
    else {
        digitalWrite(ledVerde, LOW);
        digitalWrite(ledAmarillo, LOW);
        digitalWrite(ledRojo, HIGH);
    }
}

```

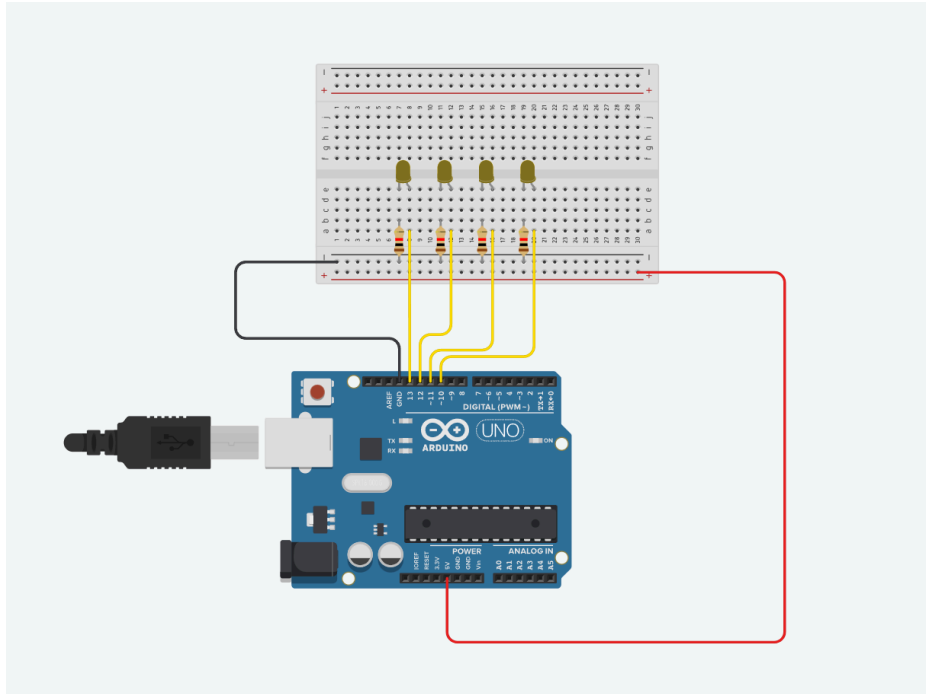
El montaje final fue:



Además hay un vídeo del funcionamiento llamado Ej2.mp4

Ejercicio 3: Secuencia de LEDs, encendiendo y apagando 4 LEDs secuencialmente, de forma similar a las lucecitas de "El coche fantástico": <https://bit.ly/3Lu7YNz>

El prototipo correspondiente a este ejercicio es el siguiente:



El código del ejercicio es:

```
const int led1 = 13;
const int led2 = 12;
const int led3 = 11;
const int led4 = 10;

void setup(){
  // Configurar los pines como salidas
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);
}

void loop(){

  // De derecha a izquierda
  digitalWrite(led1, HIGH);
  delay (300);
```



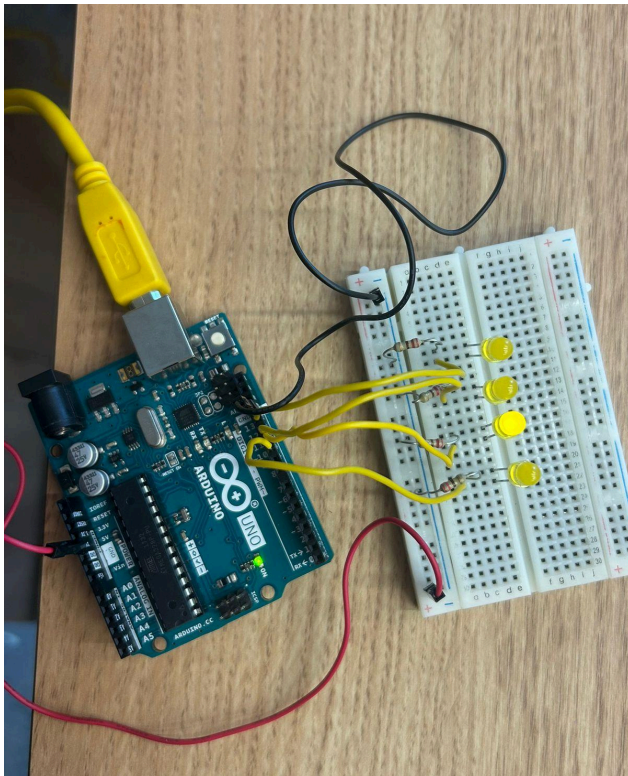
```

digitalWrite(led1, LOW);
digitalWrite(led2, HIGH);
delay(3000);
digitalWrite(led2, LOW);
digitalWrite(led3, HIGH);
delay (300);
digitalWrite(led3, LOW);
digitalWrite(led4, HIGH);
delay(300);
digitalWrite(led4, LOW);

// De izquierda a derecha
digitalWrite(led3, HIGH);
delay (300);
digitalWrite(led3, LOW);
digitalWrite(led2, HIGH);
delay (300);
digitalWrite(led2, LOW);
}

```

El montaje es el siguiente:

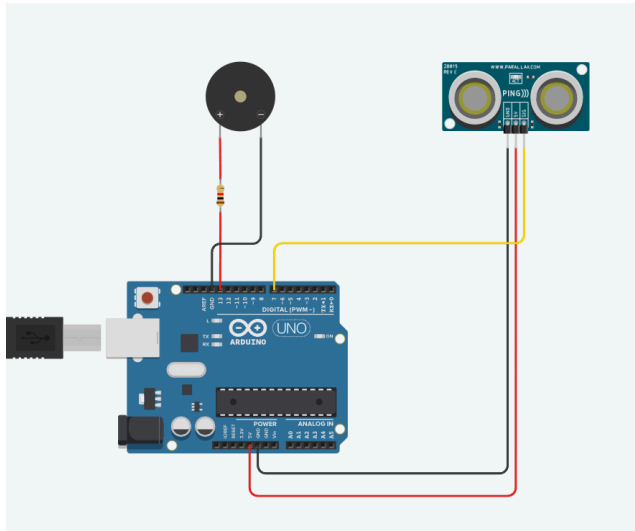


Además hay un vídeo del funcionamiento llamado Ej3.mp4

Ejercicio 4: Detector de la distancia a un objeto (usar el buzzer para hacer sonar un pitido en función de la distancia detectada por el sensor de ultrasonidos).

Este ejercicio tratamos de realizarlo, pero nos encontramos con que el Kit de arduino proporcionado por la universidad le faltaba la pieza del sensor de distancia de ultrasonidos, por lo que solamente realizamos el prototipo en Tinkercad y programamos el código, pero no pudimos montarlo.

Prototipo:



Código:

```
const int ultra= 9;
const int echo = 10;
const int buzzer = 8;

void setup() {
  pinMode(ultra, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(buzzer, OUTPUT);
}

void loop() {
  long duration;
  int distance;

  digitalWrite(ultra, LOW);
  delayMicroseconds(2);
  digitalWrite(ultra, HIGH);
  delayMicroseconds(10);
  digitalWrite(ultra, LOW);
```

```
duration = pulseIn(echo, HIGH);

// Calcular distancia en cm
distance = duration * 0.034 / 2;

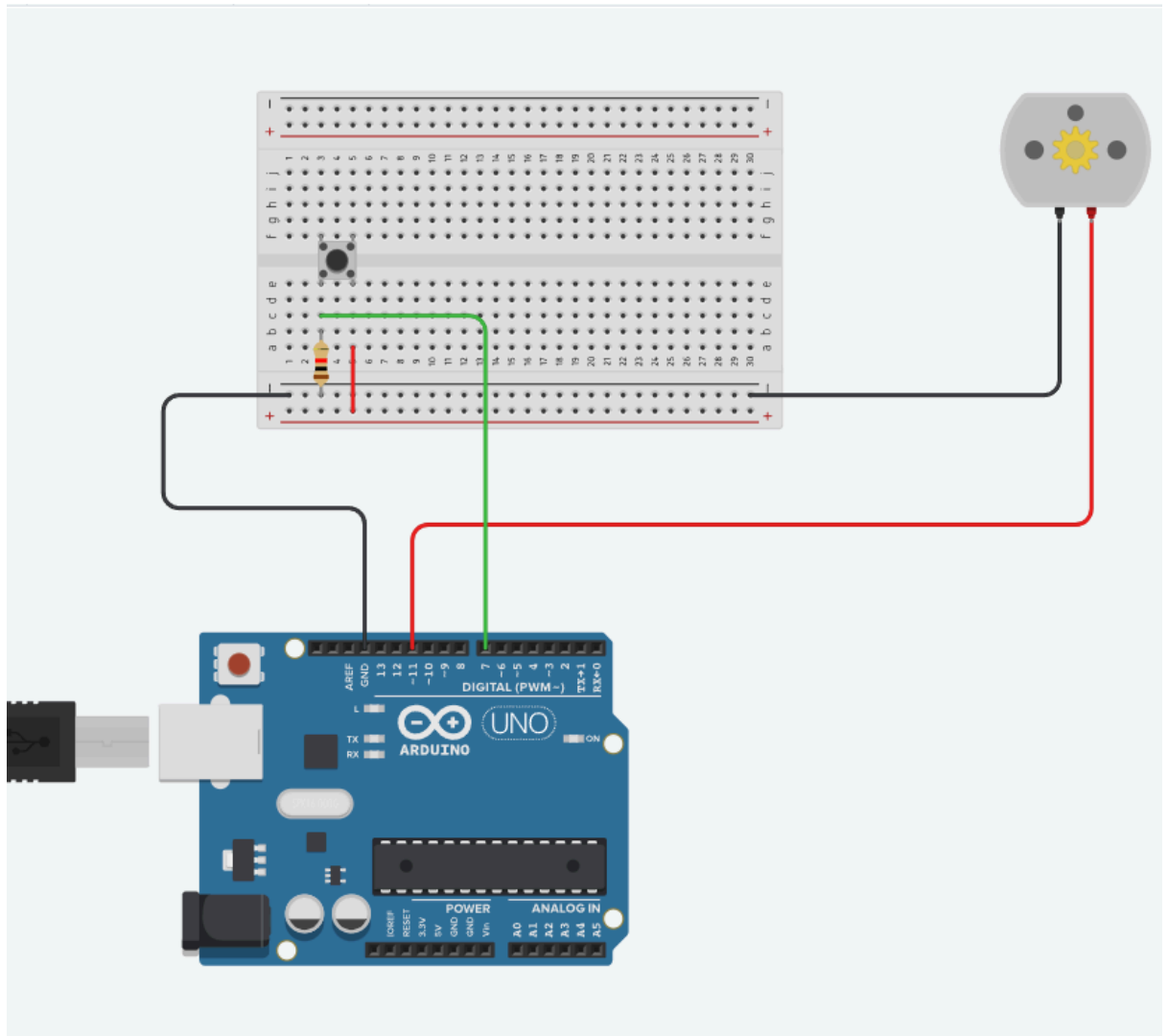
// Si está a menos de 5 cm suena
if (distance < 5 && distance > 0) {
    tone(buzzer, 1000);
} else {
    noTone(buzzer);
}

// Se ejecuta una vez cada segundo
delay(1000);
}
```

Ejercicio 6: Implementar un proyecto en el que se active un motor (DC Motor o Servo motor) cuando se pulse un pulsador.

Este ejercicio desarrollamos el código y lo montamos, pero el motor correspondiente a nuestro motor el cable correspondiente al positivo estaba roto y no conseguimos que funcionara, pensamos que el hecho de que no funcionara era debido a que el motor no lo hacía, ya que probamos a hacerlo con un len en lugar del motor y si funcionaba.

Prototipo:



Código:

```
int pulsador = 7;
int motor = 11;
int val= 0;

void setup() {
    pinMode(motor, OUTPUT);
    pinMode(pulsador, INPUT);
    digitalWrite(motor, LOW);
}

void loop() {
    val = digitalRead(pulsador);

    // Si esta pulsado el pulsador el motor funciona
    if(val == HIGH) {
        digitalWrite(motor, HIGH);
    } else {
        digitalWrite(motor, LOW);
    }
}
```

Montaje:

