

Periféricos y dispositivos de interfaz humana

Práctica 3



**UNIVERSIDAD
DE GRANADA**

Ejercicio 1 - Parpadeo de leds consecutivos

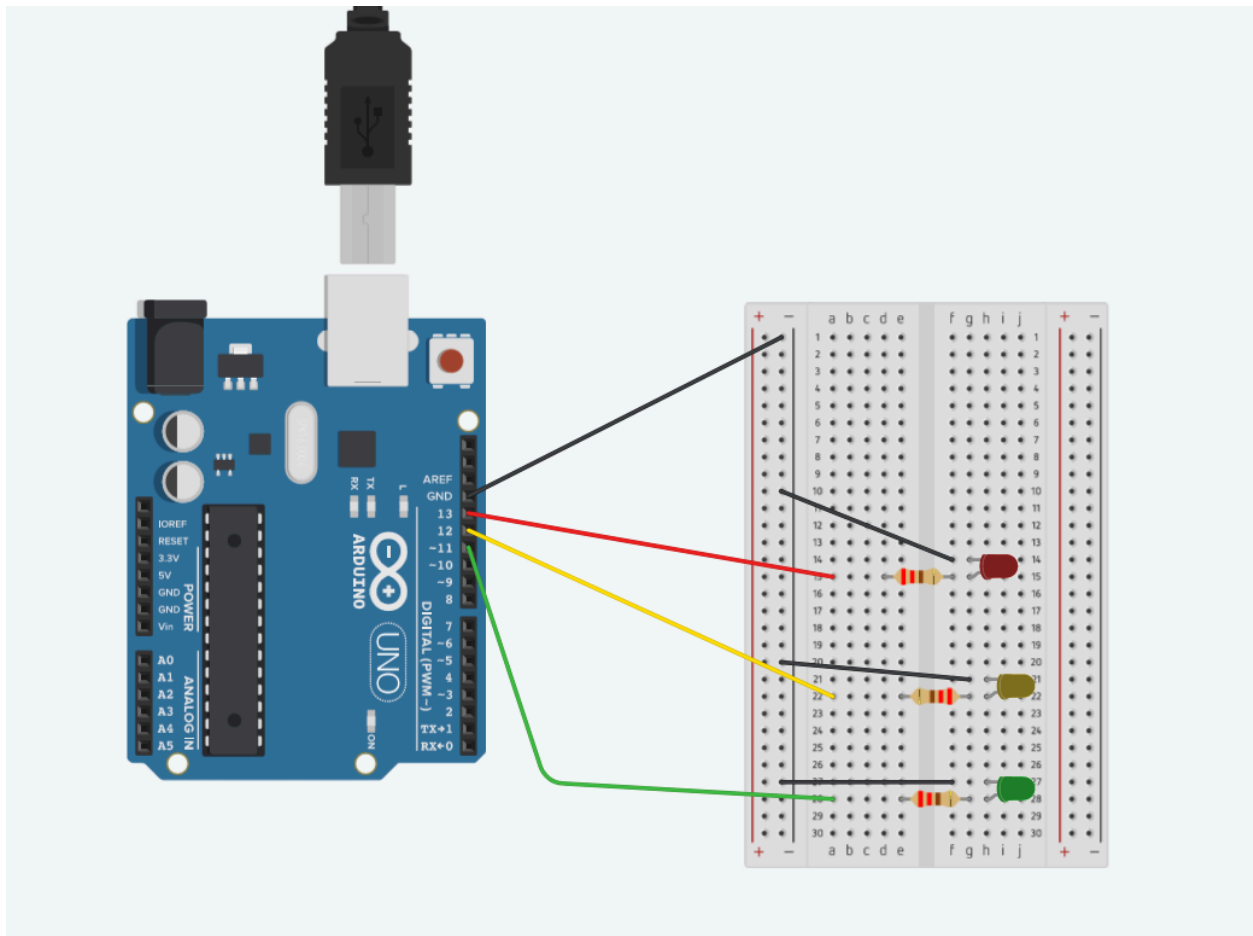
Configuración

Para este ejercicio hemos conectado 3 leds a los pines 13, 12 y 11 respectivamente acompañado de sus resistencias conectadas al ánodo de las mismas y hemos creado un código que enciende los leds de manera cíclica y consecutiva.

Materiales

- 3 Resistencias 220 Ω
- 1 led rojo
- 1 led amarillo
- 1 led verde

Diseño

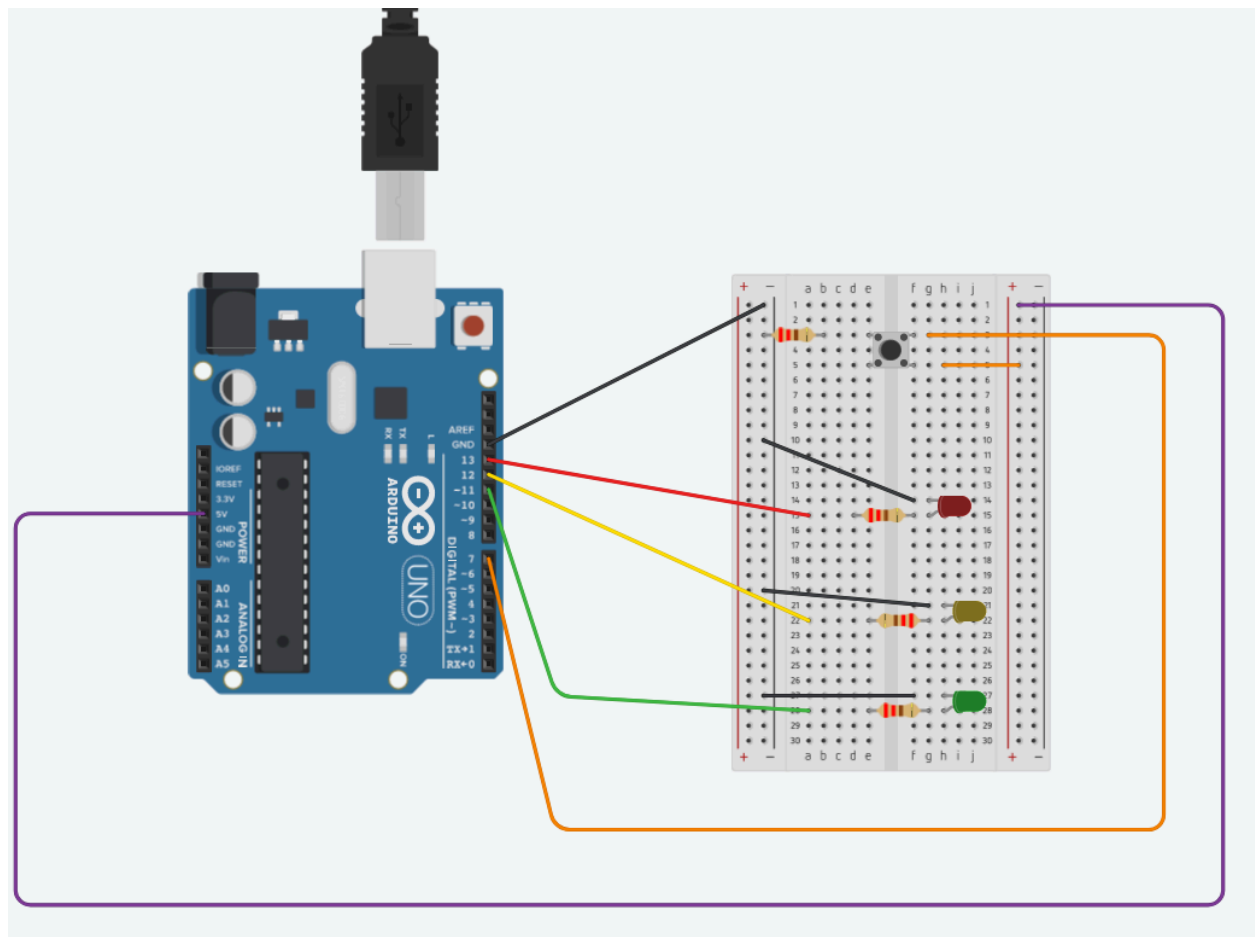


Código

```
C/C++  
void setup()  
{  
  pinMode(13, OUTPUT); //Led rojo  
  pinMode(12, OUTPUT); //Led amarillo  
  pinMode(11, OUTPUT); //Led verde  
  
}  
  
void loop()  
{  
  digitalWrite(13, HIGH);  
  digitalWrite(12, LOW);  
  digitalWrite(11, LOW);  
  
  delay(500); // Wait for 500 millisecond(s)  
  
  digitalWrite(13, LOW);  
  digitalWrite(12, HIGH);  
  digitalWrite(11, LOW);  
  
  delay(500); // Wait for 500 millisecond(s)  
  
  digitalWrite(13, LOW);  
  digitalWrite(12, LOW);  
  digitalWrite(11, HIGH);  
  
  delay(500); // Wait for 500 millisecond(s)  
}
```

Vídeo

El vídeo de la ejecución del programa se encuentra en  E1.mp4 .



Código

```
C/C++
int pulsador =0;

void setup()
{

    pinMode(13, OUTPUT); // led rojo
    pinMode(12, OUTPUT); // led amarillo
    pinMode(11, OUTPUT); // led verde

    pinMode(7, INPUT); //pulsador
}

void loop()
{
    pulsador = digitalRead(7);
    if( pulsador == HIGH) { // Si el pulsador está pulsado
        digitalWrite(13, HIGH);
        digitalWrite(12, LOW);
        digitalWrite(11, LOW);
    } else {
        digitalWrite(13, LOW);
        digitalWrite(12, HIGH);
        digitalWrite(11, LOW);

        delay(500); // Wait for 500 millisecond(s)

        digitalWrite(12, LOW);
        digitalWrite(11, HIGH);

        delay(500); // Wait for 500 millisecond(s)
    }
}
```

Vídeo

El vídeo de la ejecución del programa se encuentra en  E2.mp4 .

Ejercicio 3 - Parpadeo coche fantástico

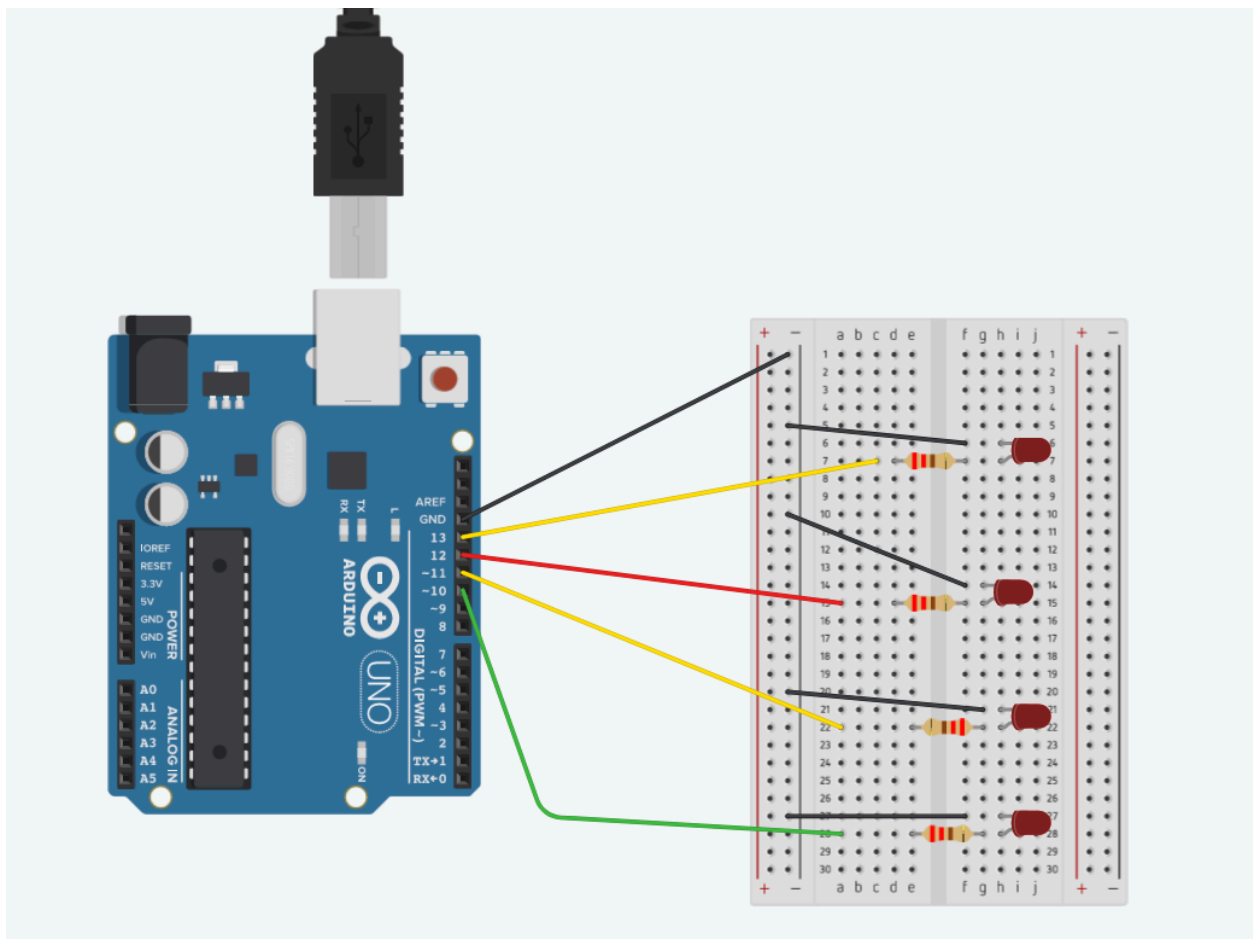
Configuración

En este ejercicio hemos conectado 3 leds a los pines 13, 12, 11 y 10 respectivamente acompañado de sus resistencias conectadas al ánodo de las mismas, para el código lo que hemos hecho ha sido encender los leds de manera consecutiva en los 2 sentidos, dando la sensación de que es una estela que viene y va como se puede ver en Kit del coche fantástico.

Materiales

- 4 Resistencias 220 Ω
- 4 leds rojos

Diseño



Código

C/C++

```
void setup()
{
  pinMode(13, OUTPUT); //led
  pinMode(12, OUTPUT); //led
  pinMode(11, OUTPUT); //led
  pinMode(10, OUTPUT); //led
}

void loop()
{
  digitalWrite(13, HIGH);
  digitalWrite(12, LOW);
  digitalWrite(11, LOW);
  digitalWrite(10, LOW);

  delay(100); // Wait for 500 millisecond(s)

  digitalWrite(13, LOW);
  digitalWrite(12, HIGH);
  digitalWrite(11, LOW);
  digitalWrite(10, LOW);
  delay(100); // Wait for 500 millisecond(s)

  digitalWrite(13, LOW);
  digitalWrite(12, LOW);
  digitalWrite(11, HIGH);
  digitalWrite(10, LOW);

  delay(100); // Wait for 500 millisecond(s)

  digitalWrite(13, LOW);
  digitalWrite(12, LOW);
  digitalWrite(11, LOW);
  digitalWrite(10, HIGH);

  delay(100);

  digitalWrite(13, LOW);
  digitalWrite(12, LOW);
  digitalWrite(11, HIGH);
  digitalWrite(10, LOW);
```

```
delay(100); // Wait for 500 millisecond(s)

    digitalWrite(13, LOW);
    digitalWrite(12, HIGH);
    digitalWrite(11, LOW);
    digitalWrite(10, LOW);
    delay(100); // Wait for 500 millisecond(s)

    digitalWrite(13, HIGH);
    digitalWrite(12, LOW);
    digitalWrite(11, LOW);
    digitalWrite(10, LOW);
}
```

Vídeo

El vídeo de la ejecución del programa se encuentra en  E3.mp4 .

Ejercicio 4 - Parpadeo de leds y pulsador

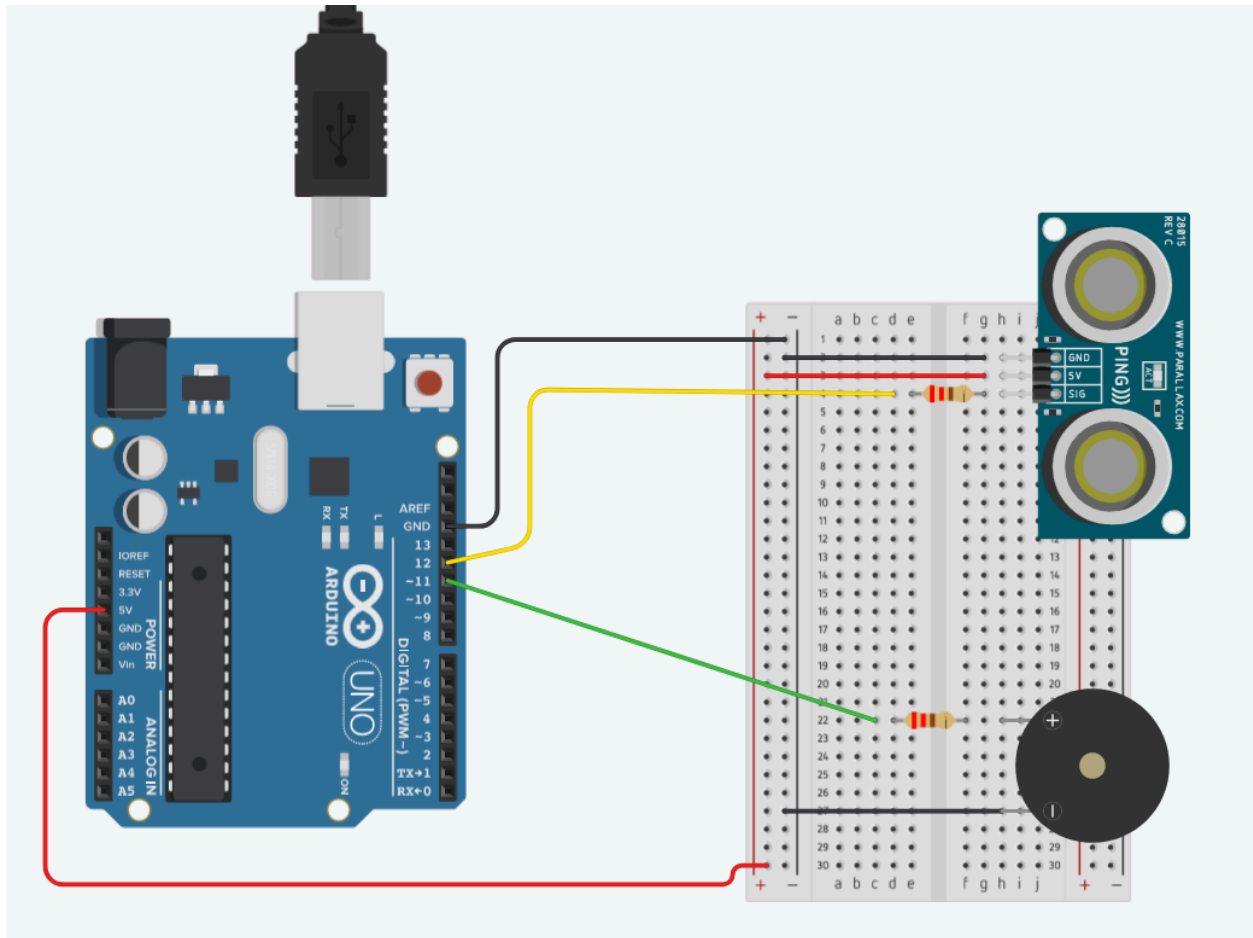
Configuración

En este ejercicio hemos conectado un sensor de distancia a sus pines necesarios (tierra, 5V y pin 12 con una resistencia) y un buzzer conectado al pin 11 con su resistencia. Lo que hace este diseño es que cuando el objeto está a menos de 200 cm del sensor de distancia el buzzer empieza a sonar hasta que éste se aleja.

Materiales

- 2 Resistencias 220 Ω
- 1 sensor de distancia
- 1 buzzer

Diseño



Código

C/C++

```
// Definimos el pin para el sensor ultrasónico (equivalente a SIG)
#define SIG_PIN 12
#define BUZ_PIN 11
int pulso;
float distancia;

void setup()
{
  pinMode(LED_BUILTIN, OUTPUT);
}
```

```

void loop()
{
  pinMode(SIG_PIN, OUTPUT);
  digitalWrite(SIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(SIG_PIN, LOW);
  pinMode(SIG_PIN, INPUT);
  pulso = pulseIn(SIG_PIN, HIGH); //medimos la duración del pulso ultrasónico
  distancia = (pulso * 0.0343) / 2; //asumimos que el sonido viaja a una
  velocidad de 343 metros por segundo en el aire

  if(distancia < 200) {
    digitalWrite(LED_BUILTIN, HIGH);
    tone(BUZ_PIN, 1000); //se activa el buzzer
  } else { //se apaga el buzzer
    digitalWrite(LED_BUILTIN, LOW);
    noTone(BUZ_PIN);
  }

  delay(10);
}

```

Vídeo

El vídeo de la ejecución del programa se encuentra en  E4.mp4 .

Ejercicio 5 - Luz led en función de fotorresistencia

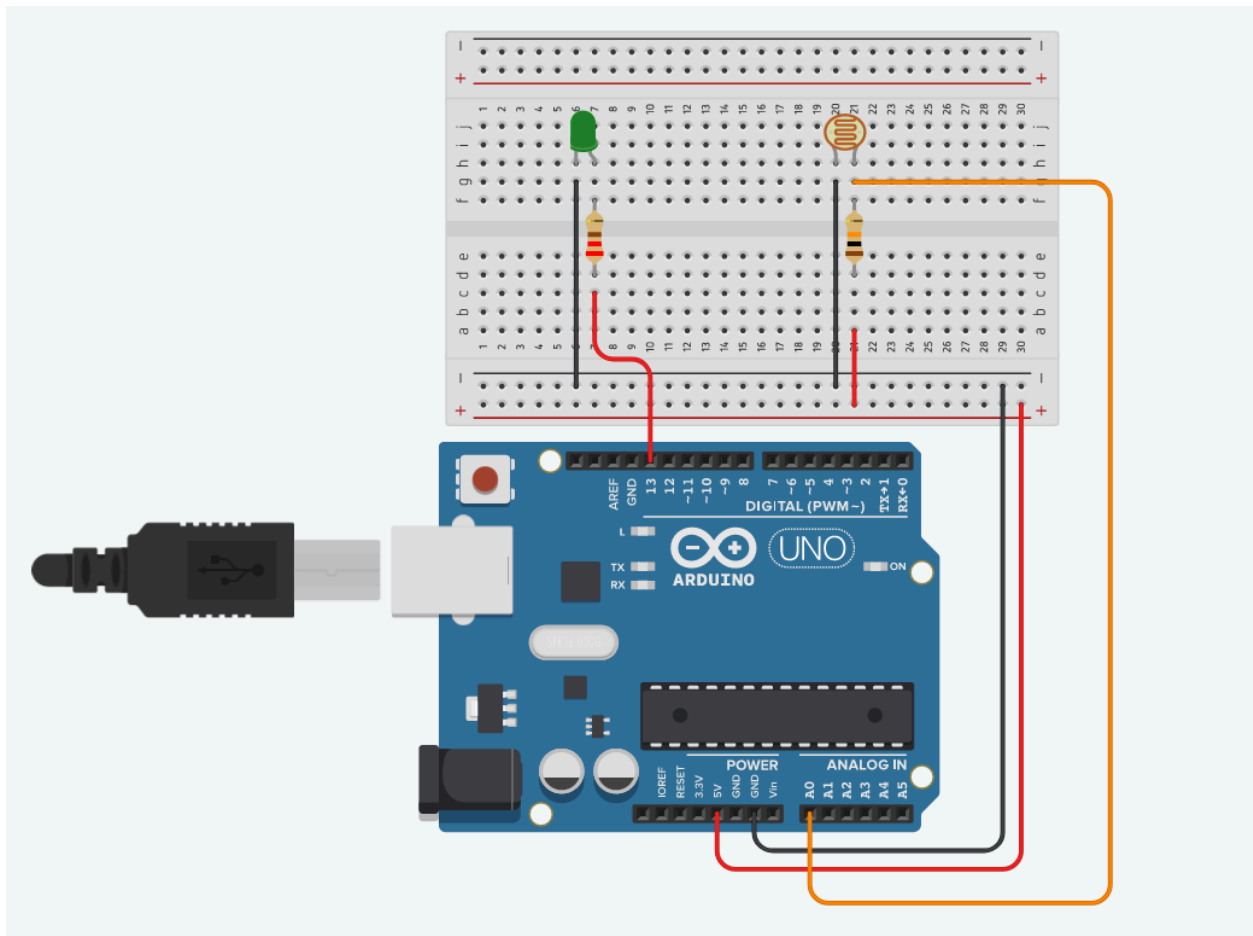
Configuración

En este ejercicio hemos conectado led al pin 13 y una fotorresistencia al pin de entrada analógica A0 con esto y el código correspondiente conseguimos que en función de la luz que le llegue a la fotorresistencia se encienda más o menos el led verde.

Materiales

- 1 resistencia 220 Ω
- 1 resistencia de 10 k Ω
- 1 led verde
- 1 fotorresistencia

Diseño



Código

```
C/C++
#define LED_PIN 13
int sensor = 0;

void setup()
{
    pinMode(A0, INPUT); //configuramos el pin A0 como uno de entrada
    Serial.begin(9600); //Se inicializa la comunicación serial a una velocidad de
    9600
    pinMode(LED_PIN, OUTPUT); //se configura el pin 13 como una salida
}

void loop()
{
    sensor = analogRead(A0); //Leemos el valor analógico del pin A0 utilizando
    Serial.println(sensor);
    /*
    El valor del sensor se mapea desde el rango de 0 a 1023 al 0 255 que
    es el que puede controlar el pin analógico
    */
    analogWrite(LED_PIN, map(sensor, 0, 1023, 0, 255));
    delay(10);
}
```

Vídeo

El vídeo de la ejecución del programa se encuentra en  E5.mp4 .

Nota: los dos últimos ejercicios no tiene vídeo del montaje porque no tenía los materiales necesarios para su realización, de ahí que el vídeo sea de simulación.