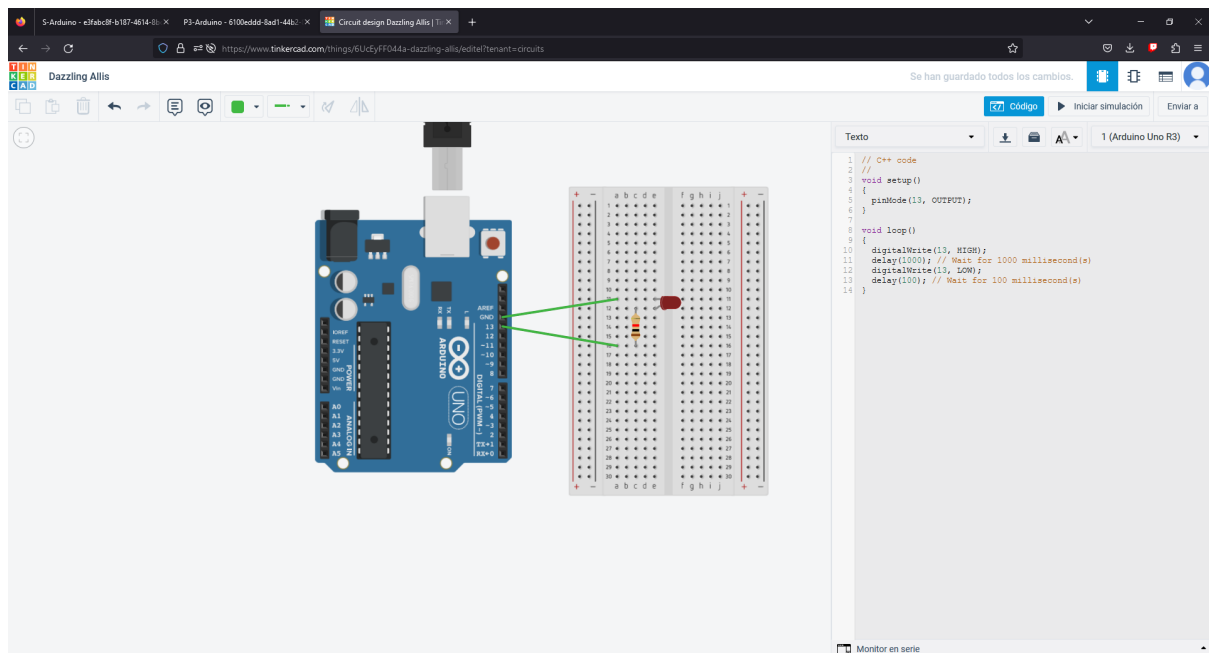


# Práctica 3. Experimentación con Arduino

El objetivo de esta práctica consiste en realizar una serie de programas simples en Arduino. Además de las capturas de pantalla aquí presentes y el código creado se presentarán enlaces a dichos circuitos.

En primer lugar se crea el proyecto básico creado en la práctica en el que un led se apaga y enciende. Quedaría algo así:



## Led Simple

A continuación se pasa a crear el programa que enciende 3 leds alternativamente cada 1,5 segundos. Se utiliza el siguiente código y tiene la siguiente apariencia:

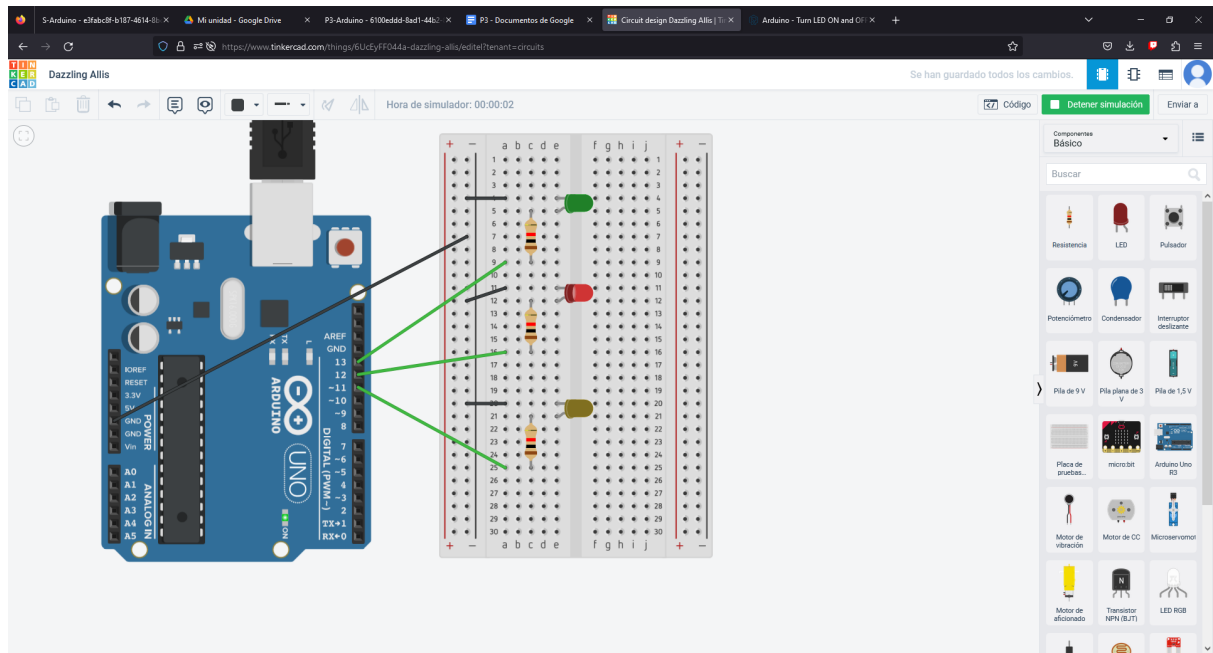
```
void setup()
{
  pinMode(13, OUTPUT);
  pinMode(12, OUTPUT);
  pinMode(11, OUTPUT);
}
```

```
void loop()
{
  digitalWrite(13, HIGH);
  delay(1500);
  digitalWrite(13, LOW);
  digitalWrite(12, HIGH);
  delay(1500);
```

```

digitalWrite(12, LOW);
digitalWrite(11, HIGH);
delay(1500);
digitalWrite(11, LOW);
}

```



### 3 Leds

El siguiente paso es añadir un botón a la ecuación. La luz roja se encenderá cuando el botón sea pulsado; si no es pulsado, se encenderán los otros 2 leds. El código será:

```

void setup() {
  pinMode(11, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(7, OUTPUT);
  pinMode(6, INPUT);
}

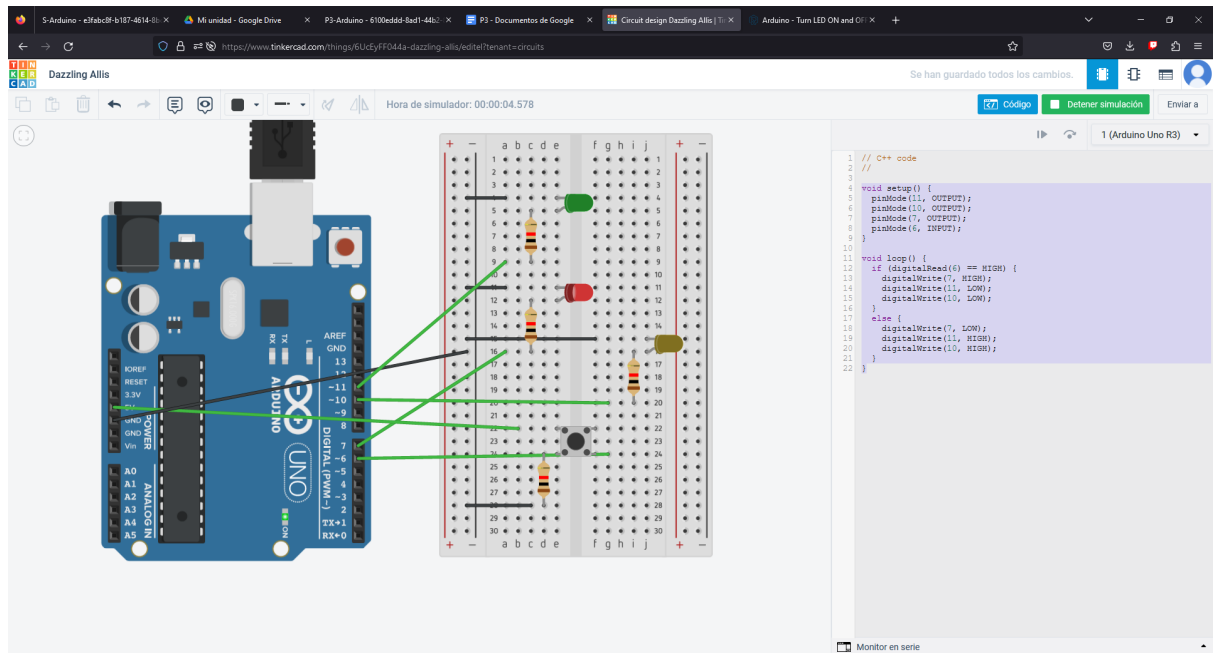
void loop() {
  if (digitalRead(6) == HIGH) {
    digitalWrite(7, HIGH);
    digitalWrite(11, LOW);
    digitalWrite(10, LOW);
  }
  else {
    digitalWrite(7, LOW);
    digitalWrite(11, HIGH);
    digitalWrite(10, HIGH);
  }
}

```

```

}
}

```



### 3 leds y botón

Por último, como ejercicio opcional para subir nota, se hará la tercera propuesta: un detector de la cantidad de luz que haya en el momento, utilizando una luz led como indicador.

```

const int LIGHT_SENSOR_PIN = A0;
const int LED_PIN           = 3;

```

```

int analogValue;

```

```

void setup() {
  pinMode(LED_PIN, OUTPUT); // set arduino pin to output mode
  pinMode(LIGHT_SENSOR_PIN, INPUT);
}

```

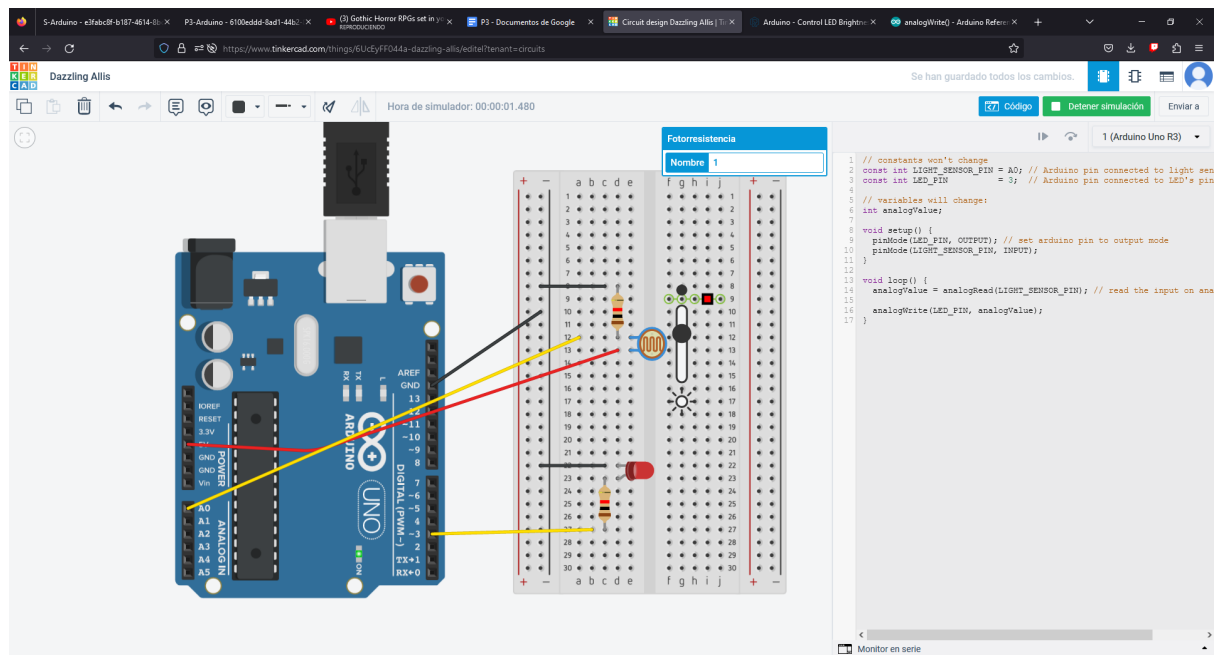
```

void loop() {
  analogValue = analogRead(LIGHT_SENSOR_PIN);

  analogWrite(LED_PIN, analogValue);
}

```

En este caso usamos *analogRead* y *analogWrite* ya que se usan los valores del sensor directamente en el código.



## Sensor Luz