



# PROYECTO N° 3

## Controlar el zumbador

## PROYECTO 3: Controlar el zumbador

Aprende a programar cómo controlar el sonido del zumbador a través de estos tres ejercicios.

### ÍNDICE DE EJERCICIOS:

- Ejercicio 3.1: Encender y apagar el zumbador con el pulsador
- Ejercicio 3.2: Encender el zumbador y dos LEDs con el potenciómetro
- Ejercicio 3.3: Controlar todo el sistema con el potenciómetro

**Ejercicio 3.1:** Encender y apagar el zumbador con el pulsador

Programa para que, al presionar el pulsador, el zumbador emita un sonido, y al soltar el pulsador el zumbador no suene.

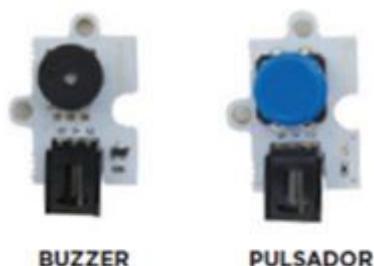
**NIVEL DE DIFICULTAD:** Principiante.

**DURACIÓN DEL EJERCICIO:** 20 min.

**MATERIALES:**

- 1 Zumbador
- 1 Pulsador
- 1 Cable USB - Micro USB
- Ordenador

El Mini Lab deberá estar montado de acuerdo a las instrucciones del manual.



**¿Que és un zumbador?**

Un zumbador es un transductor electroacústico que produce un sonido continuo o intermitente de un mismo tono, generalmente agudo. Sirve como mecanismo de señalización o aviso y se utiliza en múltiples sistemas, como automóviles y electrodomésticos, incluidos los despertadores.

**CONEXIONES:**

1. Conecta el pulsador al puerto digital 4 de la placa controladora Build&Code 4in1.
2. Conecta el zumbador al puerto digital 5 de la placa controladora Build&Code 4in1.

**CÓDIGO DE PROGRAMACIÓN:**

Puedes realizar esta actividad utilizando los *software* Arduino y Bitbloq, además de otros software de programación por bloques compatibles. A continuación encontrarás el código de programación necesario.

**Código con Arduino**

1. [Descarga el software Arduino](#) y realiza el proceso de instalación.
2. Abre el programa Arduino y, una vez en él, copia el siguiente código:

```
int valuepuls = 0, pinpuls = 4; // VARIABLE Y PIN DEL PULSADOR
int pinBuzzer = 5; // PIN DEL BUZZER

void setup() {
  // put your setup code here, to run once:
  //CONFIGURACIÓN DEL PULSADOR
  pinMode (pinpuls, INPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  // SI PULSAMOS EL PULSADOR EL BUZZER EMITIRA UNA MELODIA.
  valuepuls = digitalRead (pinpuls);
  if (valuepuls == HIGH) // PULSADOR NO PULSADO
  {
    noTone(pinBuzzer);
  }
  else
  {
    tone(pinBuzzer, 600); // tone (PIN DEL ZUMBADOR, FREQUÉNCIA);
    delay (100); // TIEMPO DE MANTENER EL TONO.
    tone(pinBuzzer, 294);
    delay (100);
    tone(pinBuzzer, 494);
    delay (100);
  }
}
```

3. Configura y carga el código, siguiendo las instrucciones indicadas en la [guía de Primeros Pasos del Mini Lab](#).
4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 esté en posición USB, para una correcta carga del código.

### **Código para software de programación por bloques compatible**

1. [Descarga el software](#) y realiza en proceso de instalación.
2. Abre el programa y, una vez en él, copia el siguiente código:

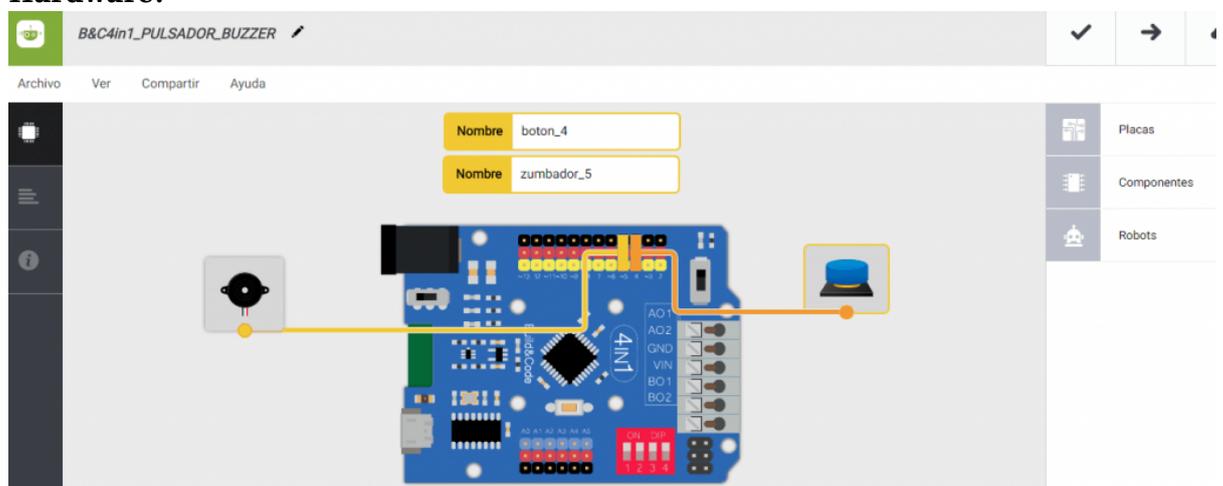


3. Configura y carga el código, siguiendo las instrucciones indicadas en la [guía de Primeros Pasos del Mini Lab](#).
4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 está en posición USB, para una correcta carga del código.

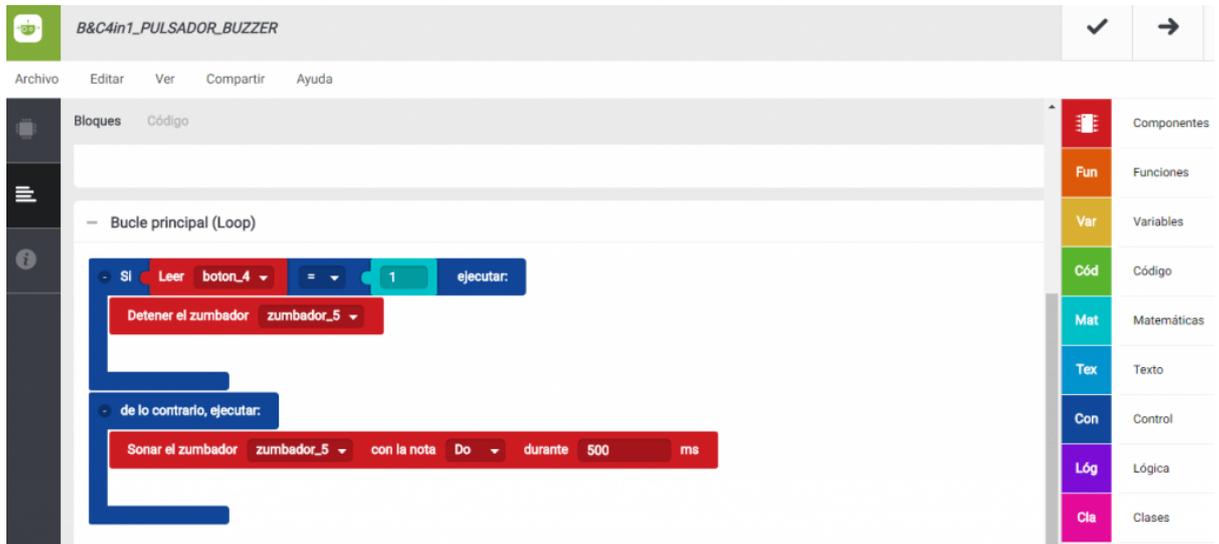
### Código Bitbloq

1. Accede [al software Bitbloq](#) y realiza el proceso de instalación de la aplicación Web2board.
2. Abre el programa Bitbloq y, una vez en él, copia el siguiente código:

- o **Hardware:**



- o **Software:**



3. Configura y carga el código, siguiendo las instrucciones indicadas en la [guía de Primeros Pasos del Mini Lab](#).
4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 está en posición USB, para una correcta carga del código.

### **Resultado del ejercicio:**

Al pulsar el botón, el zumbador hará un sonido.

### **Ejercicio 3.2:** Controla el zumbador y los LEDs con el potenciómetro

Crea un programa que permita utilizar el potenciómetro para variar el tiempo de apagado y encendido de ambos LEDs y el tiempo de sonido del zumbador.

**NIVEL DE DIFICULTAD:** Principiante.

**DURACIÓN DEL EJERCICIO:** 30 min.

### **MATERIALES:**

- 1 LED rojo
- 1 LED verde
- 1 Buzzer
- 1 Potenciómetro
- 1 Cable USB - Micro USB
- Ordenador

El Mini Lab deberá estar montado de acuerdo a las instrucciones del manual.



## CONEXIONES:

1. Conecta el buzzer al puerto digital 5 de la placa controladora Build&Code 4in1.
2. Conecta el potenciómetro al puerto analógico A1 de la placa controladora Build&Code 4in1.
3. Conecta los LEDs en los puertos digitales 10 y 3 de la placa controladora Build&Code 4in1.

## CÓDIGO DE PROGRAMACIÓN:

Puedes realizar esta actividad utilizando los *software* Arduino y Bitbloq, además de otros software de programación por bloques compatibles. A continuación encontrarás el código de programación necesario.

### Código con Arduino

1. [Descarga el software Arduino](#) y realiza en proceso de instalación.
2. Abre el programa Arduino y, una vez en él, copia el siguiente código:

```
int led10 = 10, led3 = 3, brightness; // PIN Y VARIABLE DE BRILLO
DEL LED 10
int pot = 0, pinpot = A1; //VARIABLES PINS DEL POTENCIÓMETRO
int pinBuzzer = 5; // PIN DEL BUZZER

void setup() {
  // put your setup code here, to run once:
  // CONFIGURACIÓN DE LOS LEDs 10 Y 3
  pinMode ( led10, OUTPUT);
  pinMode ( led3, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  pot = analogRead (pinpot); // LECTURA DEL VALOR DEL POTENCIÓMETRO
  digitalWrite (led10, HIGH); // LED 10 = ON
  digitalWrite (led3, LOW); // LED 3 = OFF
```

```

tone (pinBuzzer, 600); // ZUMBADOR NOTA 1
delay ((pot)); // TIEMPO DE ESPERA SEGÚN LA LECTURA DEL
POTENCIÓMETRO
digitalWrite (led10, LOW); // LED 10 = OFF
digitalWrite (led3, HIGH); // LED 3 = ON
tone (pinBuzzer, 294); // ZUMBADOR NOTA 2
delay ((pot)); // TIEMPO DE ESPERA SEGÚN LA LECTURA DEL
POTENCIÓMETRO
}

```

3. Configura y carga el código, siguiendo las instrucciones indicadas en la [guía de Primeros Pasos del Mini Lab](#).
4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 está en posición USB, para una correcta carga del código.

### Código para software de programación por bloques compatible

1. [Descarga el software](#) y realiza en proceso de instalación.
2. Abre el programa y, una vez en él, copia el siguiente código:



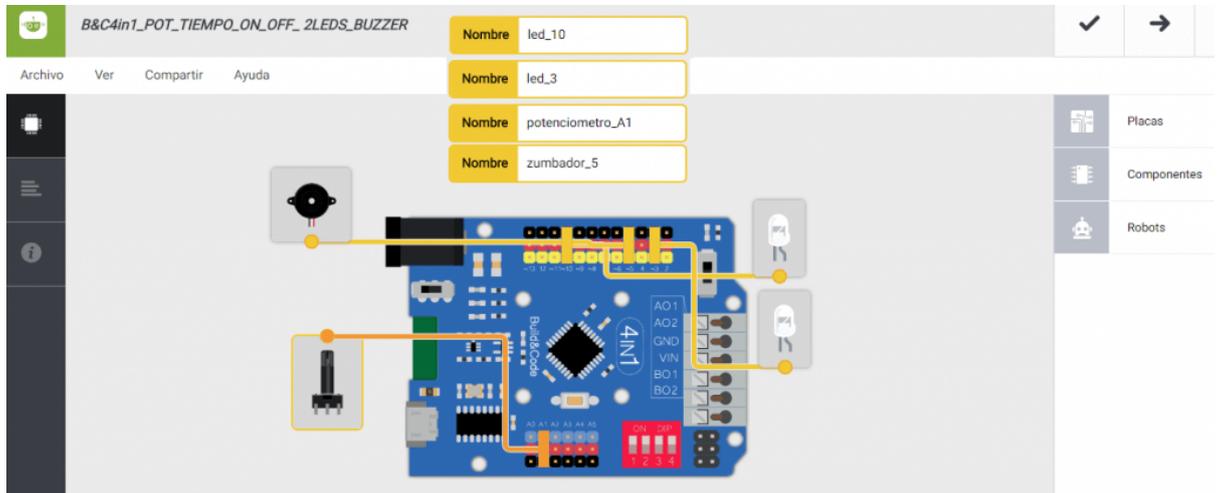
3. Configura y carga el código, siguiendo las instrucciones indicadas en la guía de Primeros Pasos del Mini Lab.
4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 está en posición USB, para una correcta carga del código.

### Código Bitbloq

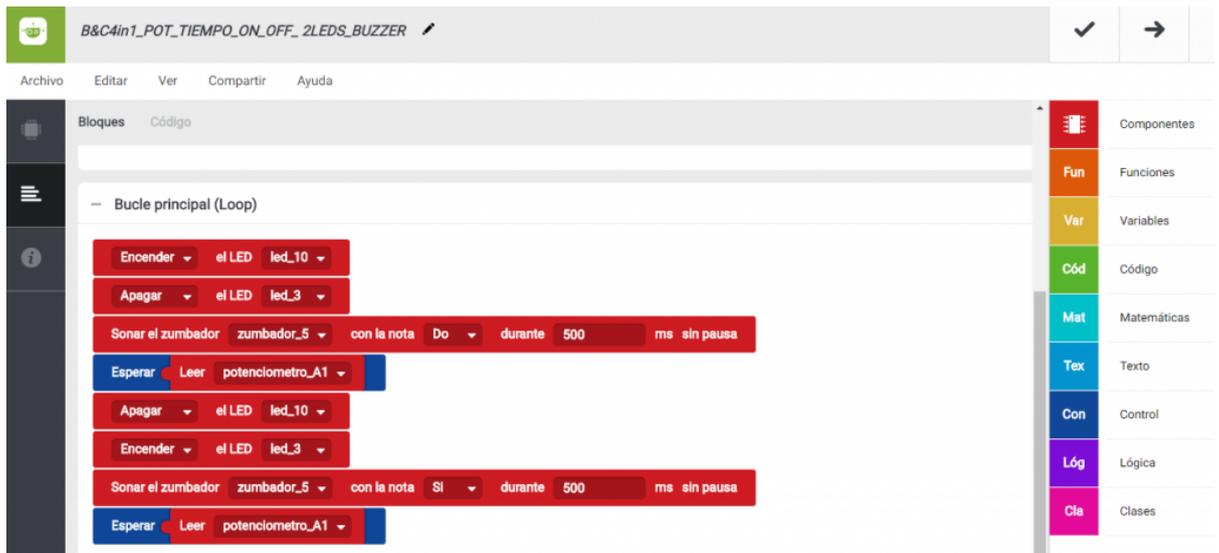
1. Accede [al software Bitbloq](#) y realiza el proceso de instalación de la aplicación Web2board.

2. Abre el programa Bitbloq y, una vez en él, copia el siguiente código:

o **Hardware**



o **Software**



3. Configura y carga el código, siguiendo las instrucciones indicadas en la [guía de Primeros Pasos del Mini Lab](#).

4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 está en posición USB, para una correcta carga del código.

**Resultado del ejercicio:**

Al girar el potenciómetro, los LEDs y el zumbador se encenderán. El zumbador emitirá un sonido diferente para cada LED, con un intervalo de tiempo marcado por el giro del potenciómetro.

**Ejercicio 3.3:** Controlar todo el sistema con el potenciómetro

Este ejercicio es similar al 3.2, con la diferencia de que ahora, al presionar el pulsador, se activará todo el sistema. El zumbador emitirá un sonido cuando se encienda el LED verde y otro sonido cuando se encienda el LED rojo, con un intervalo de tiempo marcado por el potenciómetro. Al soltar el pulsador, los LEDs y el zumbador se apagarán.

**NIVEL DE DIFICULTAD:** Intermedio.

**DURACIÓN DEL EJERCICIO:** 30 min.

### **MATERIALES:**

- 1 LED rojo
- 1 LED verde
- 1 Buzzer
- 1 Potenciómetro
- 1 Pulsador
- 1 Cable USB - Micro USB
- Ordenador

El Mini Lab deberá estar montado de acuerdo a las instrucciones del manual.



### **CONEXIONES:**

1. Conecta el pulsador al puerto digital 4 de la placa controladora Build&Code 4in1.
2. Conecta el zumbador al puerto digital 5 de la placa controladora Build&Code 4in1.
3. Conecta el potenciómetro al puerto analógico A1 de la placa controladora Build&Code 4in1.
4. Conecta los LEDs a los puertos digitales 10 y 3 de la placa controladora Build&Code 4in1.

### **CÓDIGO DE PROGRAMACIÓN:**

Puedes realizar esta actividad utilizando los *software* Arduino y Bitbloq, además de otros software de programación por bloques compatibles. A continuación encontrarás el código de programación necesario.

#### **Código Arduino**

1. [Descarga el software Arduino](#) y realiza el proceso de instalación.
2. Abre el programa Arduino y, una vez en él, copia el siguiente código:

```
int led10 = 10, led3 = 3, brightness; // PINS Y VARIABLE DE BRILLO
DEL LED 10 y 3
int pot = 0, pinpot = A1; //VARIABLES PINS DEL POTENCIÓMETRO
int pinBuzzer = 5; // PIN DEL BUZZER
int valuepuls = 0, pinpuls = 4; // VARIABLE Y PIN DEL PULSADOR

void setup() {
  // put your setup code here, to run once:

  // CONFIGURACIÓN LEDs 10 Y 3
  pinMode ( led10, OUTPUT);
  pinMode ( led3, OUTPUT);
  // CONFIGURACIÓN PULSADOR
  pinMode ( pinpuls, INPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  pot = analogRead (pinpot); // LECTURA DEL VALOR DEL POTENCIOMETRO
  // SI PULSAMOS EL PULSADOR LOS LEDS PARPADEARAN, SONARÁ EL BUZZER
  Y CON EL POTENCIOMETRO REGULAS LA VELOCIDAD DE PARPADEO
  valuepuls = digitalRead (pinpuls);
  if ( valuepuls == HIGH) // PULSADOR NO PULSADO
  {
    noTone(pinBuzzer); // NO SUENA EL ZUMBADOR
    digitalWrite (led3, LOW); // LED 3 -> OFF
    digitalWrite (led10, LOW); // LED 10 -> OFF
  }
  else
  {
    digitalWrite (led10, HIGH); // LED 10 = ON
    digitalWrite (led3, LOW); // LED 3 = OFF
    tone (pinBuzzer, 600); // ZUMBADOR = NOTA 1
    delay ((pot)); // TIEMPO DE ESPERA SEGUN LA LECTURA DEL
POTENCIOMETRO
    digitalWrite (led10, LOW); // LED 10 = OFF
    digitalWrite (led3, HIGH); // LED 3 = ON
    tone (pinBuzzer, 294); // ZUMBADOR = NOTA 2
    delay ((pot)); // TIEMPO DE ESPERA SEGUN LA LECTURA DEL
POTENCIOMETRO
  }
}
```

}

3. Configura y carga el código, siguiendo las instrucciones indicadas en la [guía de Primeros Pasos del Mini Lab](#).
4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 está en posición USB, para una correcta carga del código.

### Código para software de programación por bloques compatible

1. [Descarga el software](#) y realiza en proceso de instalación.
2. Abre el programa y, una vez en él, copia el siguiente código:

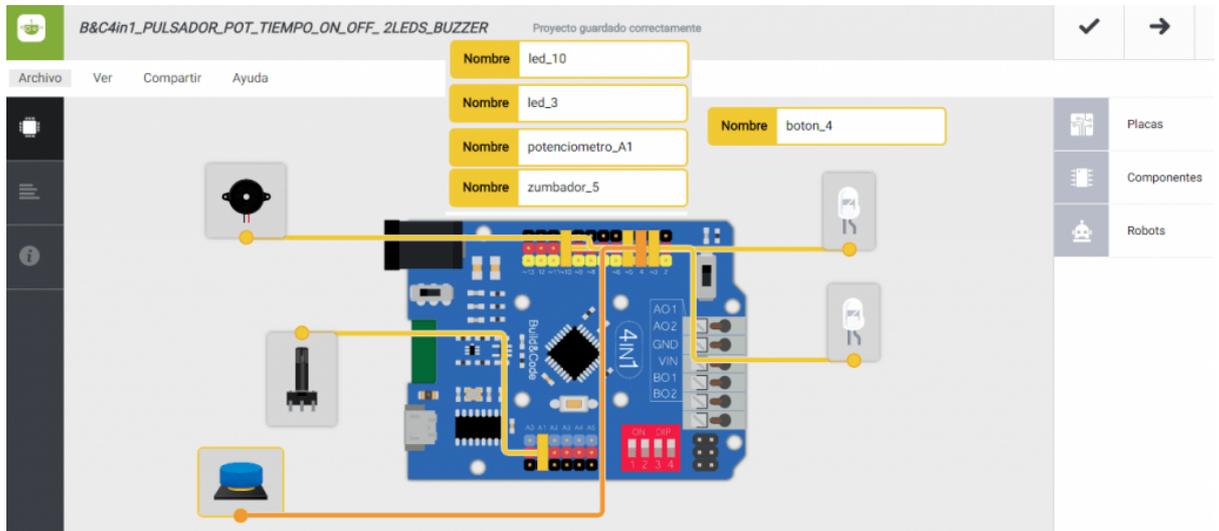
```

Programa de Arduino
por siempre
  fijar TIEMPO a leer pin analógico (A) 1 / 150  ► TIEMPO -> GUARDAR VALOR POTENCIOMETRO
  si leer pin digital 4 = 0 entonces ► BOTON -> SÍ
    fijar salida pin digital 10 a ALTO
    fijar salida pin digital 3 a BAJO
    reproducir tono 5 en nota C2 beat Cuarto
    esperar redondear TIEMPO segundos ► TIEMPO ENTRE ACCIONES
    fijar salida pin digital 10 a BAJO
    fijar salida pin digital 3 a ALTO
    reproducir tono 5 en nota B2 beat Cuarto
    esperar redondear TIEMPO segundos
  si no
    fijar salida pin digital 10 a BAJO ► BOTÓN -> NO / LEDs -> OFF / ZUMBADOR -> OFF
    fijar salida pin digital 3 a BAJO
  
```

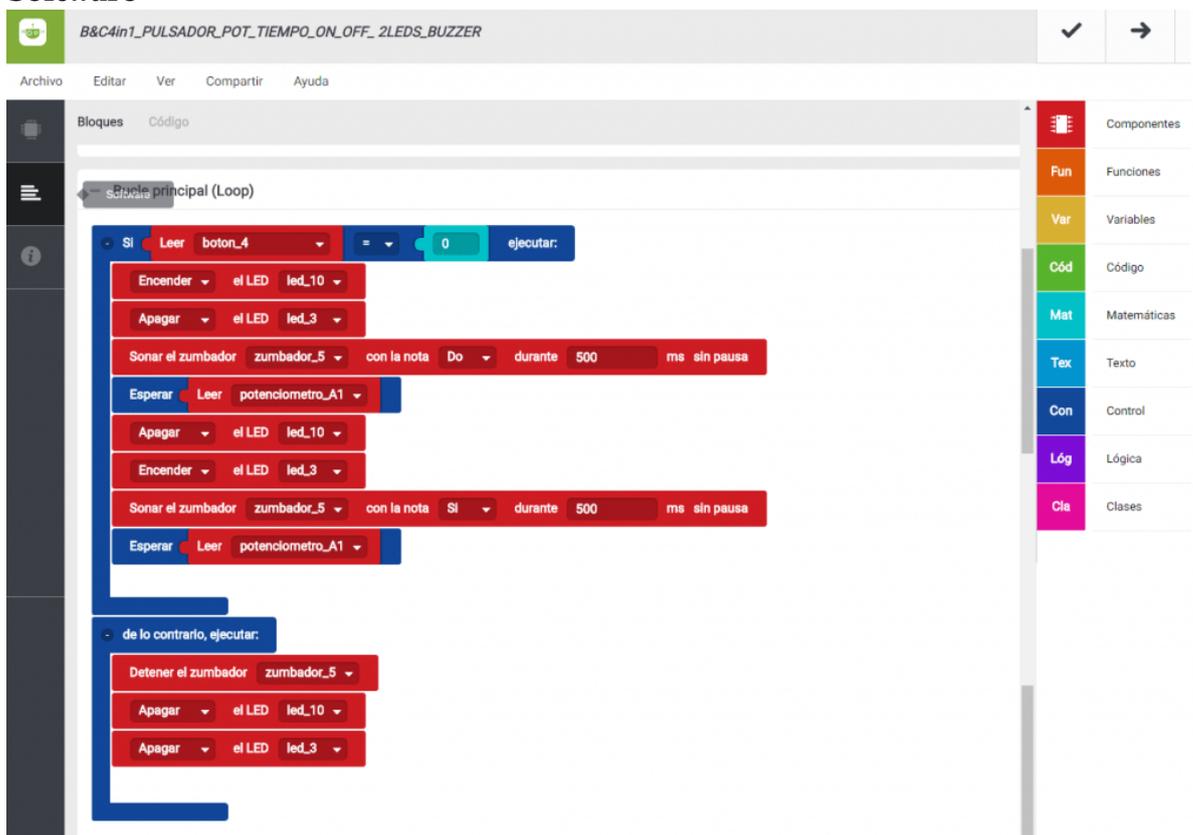
3. Configura y carga el código, siguiendo las instrucciones indicadas en la [guía de Primeros Pasos del Mini Lab](#).
4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 está en posición USB, para una correcta carga del código.

### Código Bitbloq

1. Accede [al software Bitbloq](#) y realiza el proceso de instalación de la aplicación Web2board.
2. Abre el programa Bitbloq y, una vez en él, copia el siguiente código:
  - **Hardware**



### ○ Software



3. Configura y carga el código, siguiendo las instrucciones indicadas en la [guía de Primeros Pasos del Mini Lab](#).
4. Revisa que el interruptor BLT/USB de la placa controladora Build&Code 4in1 está en posición USB, para una correcta carga del código.

### RESULTADO DEL EJERCICIO:

Al pulsar el botón, los LEDs y el zumbador se encenderán. El zumbador emitirá un sonido diferente para cada LED y éstos se encenderán de forma intermitente. Todos los componentes estarán encendidos con un intervalo de tiempo marcado por el giro del potenciómetro.