

Proyecto Nº 3: Lámpara con intensidad de luz variable

Utiliza los componentes del Maker Control Kit para aprender a construir y programar una lámpara que, al pulsar el botón, aumente la intensidad de su luz hasta el máximo y luego la disminuya gradualmente hasta apagarse.

Para realizar este proyecto utilizarás el botón y el LED blanco. Al pulsar el botón, el LED aumentará la intensidad de luz gradualmente hasta el máximo y luego la disminuirá hasta apagarse.

NIVEL DE DIFICULTAD: Principiante.

DURACIÓN DEL EJERCICIO: 30 min.

MATERIALES

- 1 Botón
- 1 LED Blanco
- 1 Vaso de plástico transparente o blanco
- 1 Placa controladora Build&Code 4in1
- 1 Cable USB - Micro USB
- Ordenador
- Material para hacer la estructura de la lámpara
- Adhesivo

¿Qué es un LED?

Un LED es un componente eléctrico semiconductor (diodo). Al ser atravesado por una corriente pequeña, el diodo emite luz.

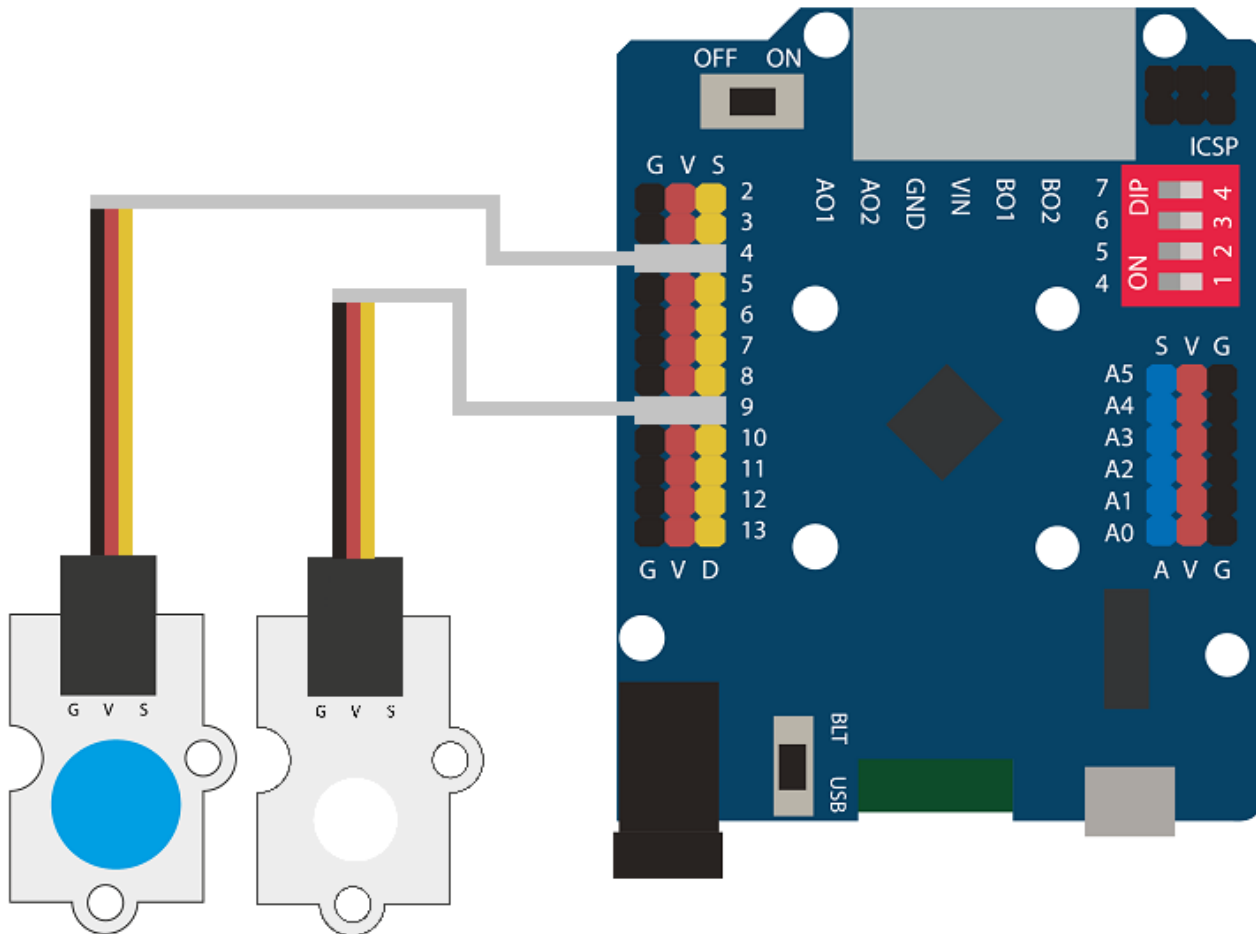
¿Qué es un botón?

Un botón o pulsador es un operador eléctrico que, cuando se oprime, permite el paso de la corriente eléctrica y cuando se deja de oprimir, lo interrumpe.

CONEXIONES:

1. Conecta el **LED blanco** al **puerto digital 9** de la placa controladora Build&Code 4in1.
2. Conecta el **botón** al **puerto digital 4** de la placa controladora Build&Code 4in1.

Para guiarte, mira los colores de los cables y los colores de los terminales de la placa controladora Build&Code 4in1. Cada cable debe ir conectado a su color:



CONSTRUCCIÓN DE LA ESTRUCTURA:

Para construir la estructura de la lámpara usarás palos de madera de 150 mm x 18 mm, un vaso de plástico, blanco o transparente y silicona caliente para hacer todas las uniones de la estructura.

[Descarga la guía rápida de montaje](#) y sigue los pasos indicados.

CÓDIGO DE PROGRAMACIÓN:

El programa consiste en controlar el comportamiento del LED con el botón. Al pulsar el botón una vez, el LED se encenderá y aumentará su intensidad de forma gradual hasta llegar al máximo. Luego, el LED disminuirá la intensidad de luz, también de forma gradual, hasta apagarse.

Para controlar el brillo del LED deberás conectarlo a uno de los puertos digitales con salida PWM, marcados con este símbolo: ~

Entonces, el puerto digital PWM estará trabajando como un puerto de salida, pero en vez de dar un valor alto o bajo de manera constante, podrá hacer cambios muy rápidos para jugar con valores medios. Esto permitirá al LED emitir distintos niveles de brillo.

El PWM tiene una resolución de 256 estados que van desde 0 hasta 255. Si escribes el valor 15, el LED apenas se encenderá, ya que estará más tiempo apagado que encendido. Si escribes el valor 127, verás el LED brillar con el 50% de su potencia máxima, ya que estará el 50% del tiempo encendido y el otro 50% apagado. Si escribes el valor 250, lo verás como si brillara al 100%.

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 para cada *software*.

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 programa:

```
int PLED = 9; // Puerto digital 9 donde va conectado el LED
int PButton = 4, ValueButton; // Puerto digital 4 donde va conectado el
botón // Variable para saber el estado del botón

void setup() {
  // put your setup code here, to run once:
  pinMode (PLED, OUTPUT); // Configurar la señal del LED como salida
  pinMode (PButton, INPUT); // Configurar la señal del Botón como entrada
}

void loop() {
  // put your main code here, to run repeatedly:
  ValueButton = digitalRead(PButton); // Leer el valor del botón y
guardarlo en la variable
  if(ValueButton == HIGH) // Si el botón es pulsado
  {
    for(int x=0; x<256; x=x+10) // Incrementar el valor de brillo del LED {
analogWrite (PLED, x); // Iluminar el LED en función del brillo delay(50);
// Espera de 50 ms } for(int x=255; x>0; x=x-10) // Disminuir el valor de
brillo del LED
    {
      analogWrite (PLED, x); // Iluminar el LED en función del brillo
delay(50); // Espera de 50 ms
    }
  }
}
```

```
else // Si el botón no está pulsado
{
  digitalWrite(PLED,LOW); // LED = OFF apagado
}
}
```

3. Configura y carga el código, siguiendo las instrucciones indicadas en el [documento de Primeros Pasos de la placa Build&Code 4in1](#).

Código para software de programación por bloques compatibles

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

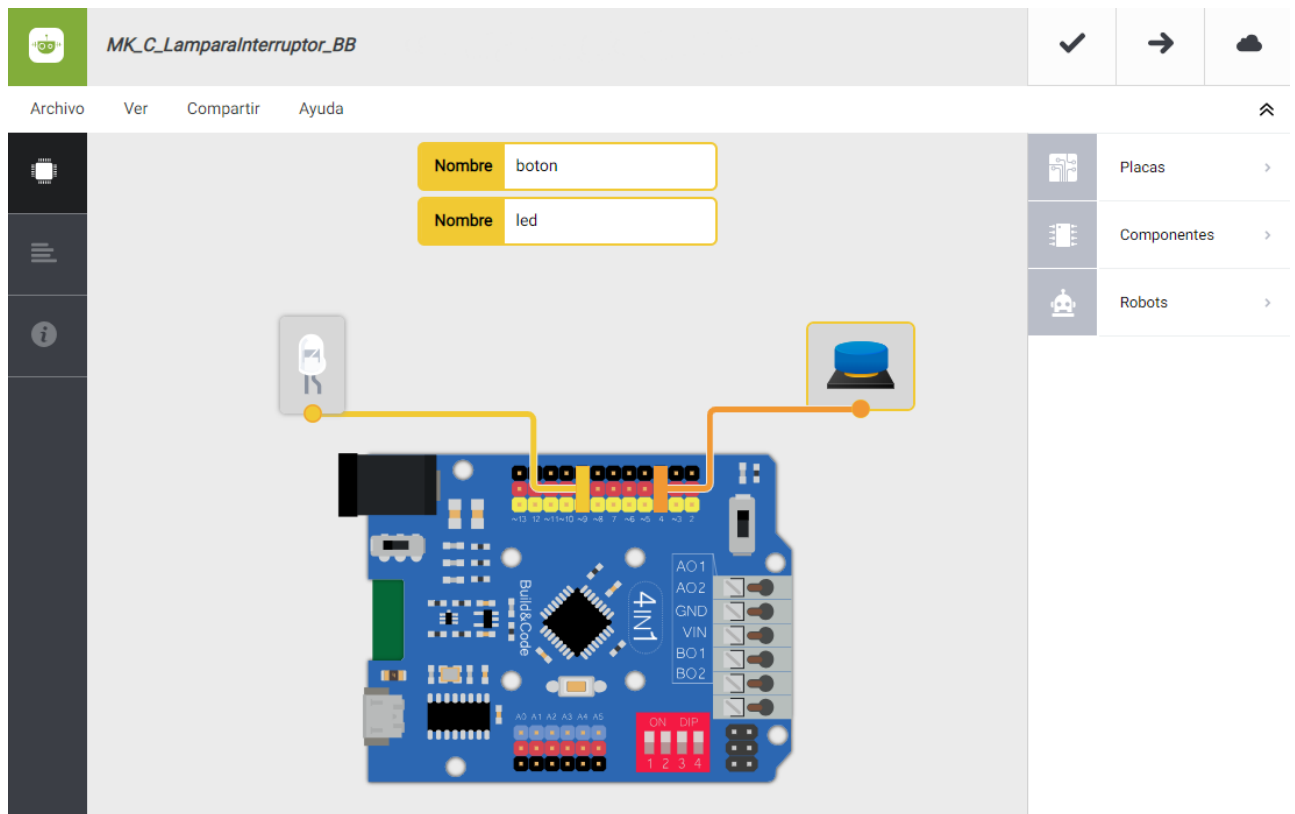


3. Configura y carga el código, siguiendo las instrucciones indicadas en el [documento de Primeros Pasos de la placa Build&Code 4in1](#).

Código BitBloq

1. Accede al software [Bitbloq](#).
2. Abre el programa Bitbloq y, una vez en él, copia el siguiente código:

- **Hardware**



- **Software**

The screenshot shows the ebotics IDE interface. At the top, there's a header with the project name 'MK_C_Onda_BB' and navigation icons. Below that is a menu bar with 'Archivo', 'Editar', 'Ver', 'Compartir', and 'Ayuda'. The main workspace is divided into sections: 'Variables globales y funciones' (containing a 'Declarar variable X = 0' block), 'Instrucciones iniciales (Setup)' (with a placeholder 'Arrastra un bloque aquí para empezar tu programa'), and 'Bucle principal (Loop)'. The loop contains a 'Si' block with 'Leer boton' set to 'Verdadero'. Inside the loop, there are two 'Contar con' blocks: one counting from 0 to 255 (sumando) and another counting from 255 to 0 (restando). Each counting block is followed by 'Escribir en el pin digital' (set to 'led') and 'Esperar 50 ms'. A 'de lo contrario, ejecutar:' block contains 'Apagar el LED'.

3. Configura y carga el código, siguiendo las instrucciones indicadas en el [documento de Primeros Pasos de la placa Build&Code 4in1](#).

RESULTADO DEL EJERCICIO:

Has construido una lámpara que al pulsar el botón el LED se enciende y se apaga, variando su brillo forma gradual.