



Control programado con Arduino

Práctica 1: Mi Primer LED (Encendido Continuo)

La primera parada de nuestro viaje consiste en entender cómo fluye la energía por nuestra placa Arduino Uno y aprender a dar instrucciones muy básicas. Hoy lograrás encender un componente emisor de luz (LED) programando tus primeras líneas de código en C++.

Objetivos de Aprendizaje

Al finalizar esta sesión, serás capaz de:

- Identificar la estructura fundamental de un programa en Arduino (`setup` y `loop`).
- Configurar un pin digital como canal de salida empleando `pinMode` .
- Controlar el paso de corriente eléctrica mediante la orden `digitalWrite` .
- Conectar correctamente un LED y una resistencia protectora en una placa de pruebas (Protoboard).

Componentes Necesarios (TinkerCad)

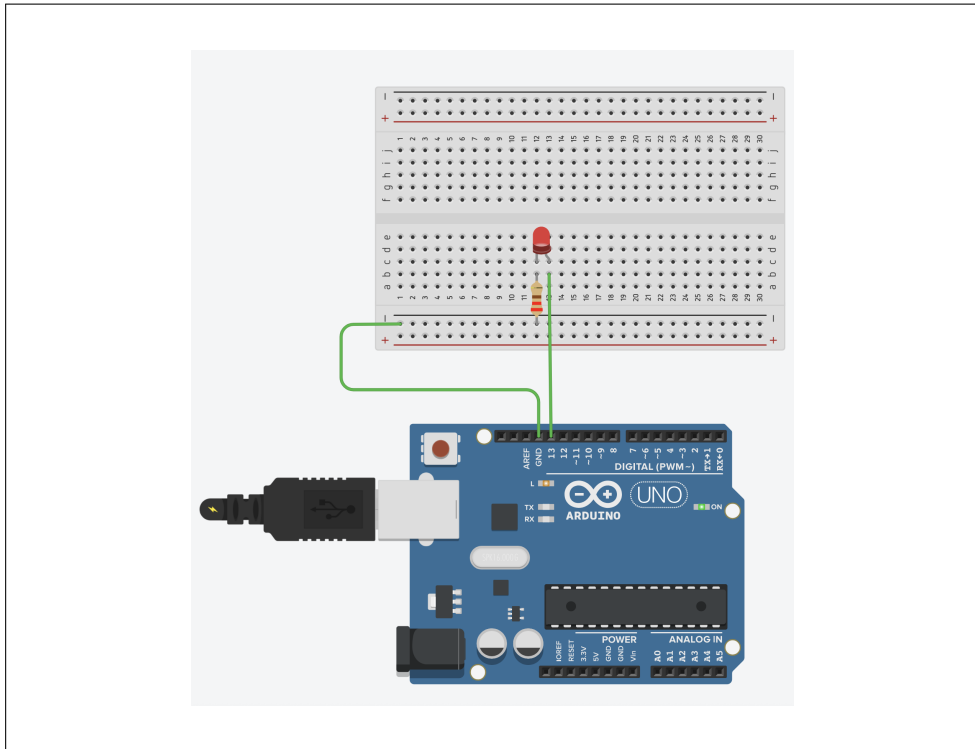
Para esta práctica solo necesitarás arrastrar los siguientes componentes en tu pantalla:

- 1 Placa Arduino Uno.
- 1 Placa de pruebas pequeña (Protoboard).
- 1 Diodo LED (puedes elegir el color que más te guste).
- 1 Resistencia de $220\ \Omega$ (*¡Importante! Si no colocas la resistencia, el LED podría quemarse debido al exceso de corriente*).
- Cables de conexión virtuales.

□ Esquema de Montaje en la Protoboard

Para conectar los componentes en TinkerCad de forma segura, sigue atentamente estas pautas:

Configuración Inicial del objeto Terminal



□ Código Base: El Andamiaje

Abre el editor de código en TinkerCad, selecciona el modo **Texto** (borra el código que venga por defecto si es necesario) y escribe el siguiente programa. Lee detenidamente los comentarios en color verde para comprender qué hace cada línea.

```
1 // Practica 1: Encendido continuo de un LED
2 // Esta funcion se ejecuta una sola vez cuando el Arduino se enciende o se
  reinicia
3 void setup() {
4   // Configura el Pin Digital 13 como una SALIDA (OUTPUT) de corriente
5   pinMode(13, OUTPUT);
6 }
7
8 // Esta funcion se ejecuta constantemente de forma infinita (en bucle)
9 void loop() {
10  // Envia un estado ALTO (HIGH / 5 Voltios) al Pin 13 para encender el LED
11  digitalWrite(13, HIGH);
12 }
```

Listing 1: Código para encender un LED continuamente.

□ ¿Cómo funciona el código?

- `setup` : Es el bloque de preparación. Aquí le decimos a Arduino qué pins usaremos y cómo se van a comportar (como entradas o salidas de información).
- `loop` : Es el motor principal del programa. Cuando termina de ejecutar la última instrucción de este bloque, vuelve inmediatamente a empezar desde la primera en un ciclo sin fin.
- `pinMode` : Define si el pin seleccionado funciona enviando energía (`OUTPUT`) o recibéndola ().
- `digitalWrite` : Permite aplicar tensión eléctrica al pin especificado. Si le enviamos `HIGH`, el pin proporcionará 5V. Si le enviamos `LOW`, se quedará en 0V (sin corriente).

□ El Reto de la Práctica 1

Ahora que has comprobado que el circuito funciona en el simulador, demuestra lo aprendido resolviendo este reto:

1. Modifica las conexiones en TinkerCad para que el LED no use el pin 13, sino el **pin digital 8**.
2. Realiza los cambios necesarios en el código para que el programa siga encendiendo el LED de manera continua.
3. *Pregunta para reflexionar:* Si quisieras apagar por completo el LED utilizando el programa, ¿qué instrucción tendrías que modificar en la función `loop` y con qué parámetro?

¡Prueba a cambiar el límite del bucle a 100 y verás qué rápido cuenta tu ordenador!

Este documento se publica bajo licencia
© Creative Commons Atribución 4.0 Internacional (CC BY 4.0)