



Control programado con Arduino

Práctica 2: El parpadeo alterno (Variables y Control Secuencial)

En la práctica anterior aprendimos a mantener un único LED encendido de forma continua. En esta sesión daremos un paso más introduciendo un segundo LED para crear un efecto de parpadeo alternativo (como las luces de un cruce ferroviario o un vehículo de emergencia). Para lograr un código ordenado y escalable, aprenderás a declarar e inicializar múltiples variables independientes.

Objetivos de Aprendizaje

Al finalizar esta sesión, serás capaz de:

- Declarar y utilizar múltiples variables de tipo entero (`int`) en tu código.
- Diseñar secuencias temporales lógicas y ordenadas utilizando la instrucción `delay` .
- Comprender el comportamiento físico de salidas independientes conectadas en paralelo.
- Optimizar el mantenimiento de tu programa mediante el uso de nombres de variables descriptivos.

Componentes Necesarios

Busca y coloca los siguientes componentes en tu mesa de trabajo de TinkerCad:

- 1 Placa Arduino Uno.
- 1 Placa de pruebas pequeña (Protoboard).
- 2 Diodos LED (te recomendamos usar dos colores diferentes, por ejemplo: Verde y Rojo).
- 2 Resistencias de $220\ \Omega$ (una para proteger cada LED).
- Cables de conexión virtuales.

Código Base: Declaración de Variables y Alternancia

Abre el editor de código en TinkerCad en modo texto. Borra el programa previo y escribe el siguiente código secuencial. Analiza cómo el estado de cada salida digital se coordina de manera síncrona en el tiempo.

```
1 // 1. DECLARACIÓN DE VARIABLES
2 // Reservamos espacios de memoria para guardar los pines de los LEDs
3 int ledRojo = 13;
4 int ledVerde = 12;
5
6 void setup() {
7   // Configuramos ambos pines como salidas de corriente
8   pinMode(ledRojo, OUTPUT);
9   pinMode(ledVerde, OUTPUT);
10 }
11
12 void loop() {
13   // ESTADO 1: Rojo encendido y Verde apagado
14   digitalWrite(ledRojo, HIGH);
15   digitalWrite(ledVerde, LOW);
16   delay(1000); // Pausa de 1 segundo en este estado
17
18   // ESTADO 2: Rojo apagado y Verde encendido
19   digitalWrite(ledRojo, LOW);
20   digitalWrite(ledVerde, HIGH);
21   delay(1000); // Pausa de 1 segundo en este estado
22 }
```

Listing 1: Código para controlar el parpadeo alternativo de dos LEDs.

¿Cómo funciona el código?

En este programa, el Arduino ejecuta las instrucciones de forma estrictamente lineal dentro de la función `loop` :

1. **Variables globales:** Declaramos `int` `ledRojo` e `int` `ledVerde` al principio. Esto nos permite usar palabras comprensibles en lugar de escribir números fijos (*hardcoded*) a lo largo del código.
2. **Alternancia síncrona:** Para lograr que un LED se apague mientras el otro se enciende, enviamos señales opuestas de forma simultánea. En el Estado 1 aplicamos `HIGH` al rojo y `LOW` al verde. Tras un segundo de pausa, invertimos los valores en el Estado 2.
3. **Bucle continuo:** Al terminar el Estado 2, el flujo vuelve de inmediato al principio de `loop` , encendiendo el rojo y apagando el verde, creando así el bucle infinito de alternancia.

Importancia del orden lógico

Si no apagásemos explícitamente el LED contrario al cambiar de estado (es decir, omitiendo las instrucciones `LOW`), el programa simplemente encendería ambos LEDs secuencialmente y los dejaría prendidos de forma indefinida, perdiendo el efecto de parpadeo alternativo.

El Reto de la Práctica 2

Comprueba que comprendes el flujo temporal y el uso de variables resolviendo estas tres propuestas en tu simulador:

1. **Efecto Policial (Parpadeo Estroboscópico):** Modifica los tiempos de los retardos (`delay`) para lograr una alternancia muy rápida que emule las luces de emergencia de una patrulla (por ejemplo, 150 ms de pausa).
2. **El poder de las variables:** Cambia el cable del LED Verde al **pin digital 8** en TinkerCad. Realiza un único cambio en tu código (aprovechando las variables) para que tu programa siga funcionando perfectamente sin alterar el `setup` ni el `loop` .
3. **Fase de Apagado Completo:** Modifica el programa para que, antes de cada cambio de luz, exista una breve fase intermedia donde los dos LEDs permanezcan apagados al mismo tiempo durante medio segundo (500 ms).