

Práctica 1 LED - Proyecto 1.1 Blink con ESP32-53 WROOM

Descripción del proyecto:

El proyecto consiste en hacer parpadear un led conectado al pin GPIO 2 del microcontrolador.

Componentes utilizados.

ESP 32-S3 WROOM

Cable Tipo C

LED (integrado en la placa, PIN 2)

Esquema del circuito:

El pin GPIO2 del ESP 32-S3 se conecta al LED. Cuando este está a nivel alto, el LED se enciende, cuando está a nivel bajo se apaga. El comportamiento se controla mediante el código cargado en el microcontrolador.

Código implementado:

```
#define LED_BUILTIN 2

void setup() {
    // Configura el pin GPIO2 como salida
    pinMode(LED_BUILTIN, OUTPUT);
}

void loop() {
    digitalWrite(LED_BUILTIN, HIGH); // Enciende el LED
    delay(1000);                 // Espera 1 segundo
    digitalWrite(LED_BUILTIN, LOW); // Apaga el LED
    delay(1000);                 // Espera 1 segundo
}
```

Proceso de carga del código:

Hay dos métodos para cargar el código en el ESP32-S3 WROOM:

1. Opción 1: UART0 / Hardware CDC

- Conectar el ESP32-S3 WROOM al computador mediante un cable Tipo C.
- Seleccionar la placa "ESP32S3 Dev Module" en Arduino IDE.
- Seleccionar el puerto serial correspondiente.
- Cargar el código y verificar que el LED comience a parpadear.

2. Opción 2: USB CDC On Boot

- Habilitar la opción "USB CDC On Boot" en Arduino IDE.
- Seleccionar el puerto serial correspondiente.
- Cargar el código y observar el parpadeo del LED.

Resultados obtenidos:

Tras cargar el código, se observó que el LED conectado al pin GPIO2 del ESP32-S3 WROOM parpadeaba de manera cíclica, encendiéndose y apagándose cada segundo. Esto confirma que el microcontrolador está funcionando correctamente y que el código se ejecuta como se esperaba.

Aprendizajes:

- Control del GPIO: Aprendizaje para conectar un pin como salida y controlar su estado para controlar elementos como un led.
- Uso funciones arduino: pinMode(), digitalWrite(), delay().
- Proceso de carga del código: Experimento con las dos opciones de carga del código en el aduino.

Evidencias de la práctica:

