

Reporte de práctica

Encender un LED con el ESP32



Presenta:

ZAHARA NATHALIA IBARRA CASILLAS
KEVIN EDUARDO LÓPEZ MARTÍNEZ
BRIAN MICHEL RUBIO MARTÍNEZ

19398 - D01

Seminario de Solución de Problemas de Programación de Sistemas Embebidos

Hecho en \LaTeX

Índice

1. Introducción	2
1.1. Objetivos	2
2. Desarrollo	2
2.1. Marco teórico	2
2.2. Metodología/Procedimiento	2
3. Resultados	3
4. Conclusión	4

1. Introducción

Esta actividad se realizó dentro del horario de clase, por lo cual, sólo se dará énfasis en el contenido visto en clase.

1.1. Objetivos

- (i) Crear un código usando la IDE de Arduino.
- (ii) Conocer y usar las funciones principales (básicas) del programa.
- (iii) Observar el algoritmo (una vez cargado el programa) en el LED del microcontrolador ESP32.

2. Desarrollo

Primero, veremos unos conceptos fundamentales para comprender el por qué usar el microcontrolador ESP32 y el desarrollo de nuestro código.

2.1. Marco teórico

- (i) ¿Qué es un microcontrolador?
Una unidad de microcontrolador (MCU) es esencialmente una computadora pequeña en un solo chip. Está diseñado para administrar tareas específicas dentro de un sistema integrado sin requerir un sistema operativo complejo.
- (ii) ¿Qué son las funciones setup() y loop() en Arduino?
El caso de la función setup, es un caso especial, ya que será, una función que va a ser llamada, sí o sí, la primer vez que se ejecute cualquier sketch de Arduino. La segunda característica, es que sólo se ejecutará una sola vez, a lo largo de la ejecución del sketch, por lo que debemos aprovechar dicha función, para inicializar variables, componentes, pines de Arduino, etcétera.
La palabra loop, significa ciclo, y eso es precisamente, lo que hace esta función. Su principal característica, es que se ejecutará infinitamente, por lo que, aquí tendremos que escribir toda la lógica de programación, que forme parte de nuestro programa.
- (iii) ¿Cómo funciona una estructura de control if?
La estructura condicional se utiliza para tomar decisiones, si la «expresión booleana» regresa un resultado verdadero (true) se ejecuta el bloque de código, si no, se omite y el flujo termina en la siguiente instrucción fuera de la estructura if.

2.2. Metodología/Procedimiento

Para comenzar, definimos una constante y le dimos el valor de 23, a la cual llamamos "LEDBUILTIN". Y también, agregamos un contador y lo inicializamos en 0, al cual llamamos count que se usará para controlar el tiempo de encendido y apagado del LED. Después, dentro de la función setup() definimos que LEDBUILTIN tendrá salida (en este caso,

el LED del microcontrolador).

Y, dentro de la función loop() dimos la instrucción de encender el LED, esperar un tiempo determinado, apagar el LED (también, esperar con un tiempo determinado) y después de esto, volver a repetir el ciclo.

Luego, definimos que la variable coun incremeta en 10 unidades después de cada ciclo.

Y por último, agregamos una estructura de control if para que coun se reinicie cada vez que coun alcance 200.

A continuación, adjuntamos el código:

```
1 // Pr ctica 1 encender un led
  // Team Septiembre(9) + compa ero Dario
  // Prop sito: Encender un led con el ESP32 y que varie su encendido
    conforme avanza el tiempo
  // 23/01/2025

6 #define LED_BUILTIN 23
  int coun=0;

  void setup() {
    // put your setup code here, to run once:
11   pinMode(LED_BUILTIN, OUTPUT);

    }

  void loop() {
16   // put your main code here, to run repeatedly:
    digitalWrite(LED_BUILTIN, HIGH);
    delay(coun);
    digitalWrite(LED_BUILTIN, LOW);
    delay(coun);
21   coun+=10;

    if (coun>=200){
      coun=0;
    }
26  }

}
```

3. Resultados

A continuación, añadimos algunas fotos del microcontrolador con el código ya cargado.

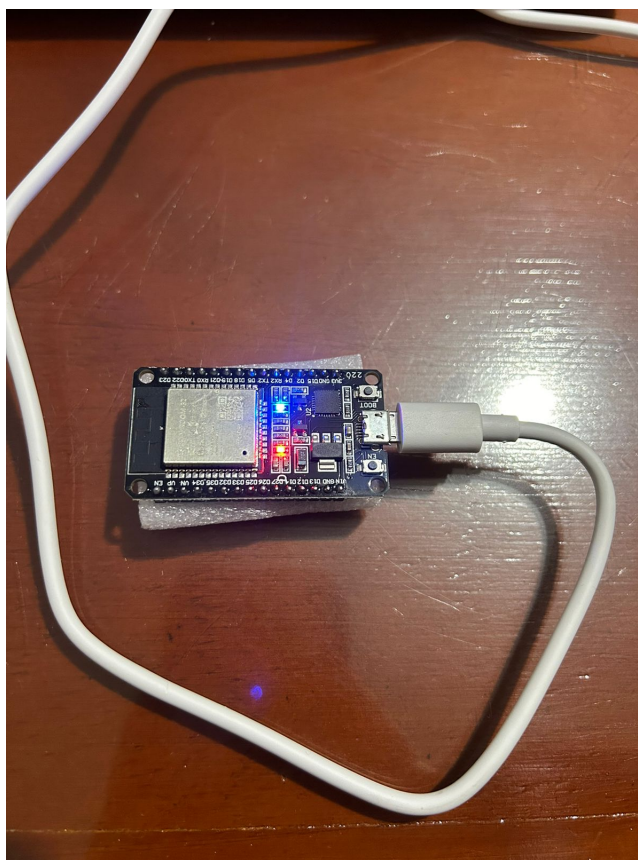


Figura 1: Foto tomada con el código ya cargado en el microcontrolador.

4. Conclusión

Esta actividad fue sencilla de realizar en el aspecto de que sólo se tuvo que implementar una estructura de control simple (en este caso `if()`) para poder percibir en el LED del microcontrolador ESP32 un pulso de encendido y apagado acelerado, y que después de este periodo, que el pulso decaída, siguiendo este ciclo continuamente. Lo que nos pudo haber dado conflictos fue que era nuestra primera vez compilando un código que no fuese para usar en un Arduino y, probablemente el que no tuviésemos frescos nuestros conocimientos en programación en el lenguaje C. Pero, con el contenido previamente visto se pudo lograr entregar esta actividad. Así que, se pudo lograr el objetivo usando el microcontrolador ESP32.

Referencias

- [1] J. Schneider, I. Smalley, (2024, Junio 04). *¿Qué es un microcontrolador?*[En línea]. Disponible en: <https://www.ibm.com/mx-es/think/topics/microcontroller>.
- [2] Proyectos Interesantes, *¿Qué son las funciones void setup y void loop?*[En línea]. Disponible en: <https://proyectosinteresantes.com/que-son-las-funciones-void-setup-y-void-loop/>.
- [3] Ávila, J. y Baltazar, J. M. (2023). *Estructura de control condicional simple (if)*. En Estructuras condicionales. Portal Académico del CCH, UNAM. <https://portalacademico.cch.unam.mx/cibernetica2/estructuras-condicionales/condicional-simple>