



REPORTE DE PRÁCTICA

Practica 1

Participantes:

Lorena Villalobos Carrillo

Ximena Lucía Rodríguez Oliva

Brayam De Jesús De La Cerda Valdivia

Juves 23 de enero de 2025

Seminario de problemas de programación de embebidos

(DO1 - I9893)

Este documento fue elaborado en \LaTeX .

Índice general

0.1. Resumen	2
0.2. Objetivos	2
0.3. Introducción	2
0.4. Marco teórico	3
0.5. Metodología	4
0.6. Resultados	5
0.7. Conclusión	6

0.1. Resumen

El objetivo de esta práctica fue comprender el manejo de salidas digitales mediante la programación en Arduino IDE, específicamente controlando la frecuencia de encendido y apagado de un LED usando el ESP32. Para ello, se diseñó un código en C++, utilizando instrucciones como `digitalWrite()` para modificar el estado del LED y `delay()` para establecer los tiempos de parpadeo. Se implementó una secuencia en la que el LED alterna entre encendido y apagado con tiempos crecientes y decrecientes en un rango de 100 ms a 1000 ms.

Los resultados obtenidos permitieron comprobar el correcto funcionamiento del ESP32 como controlador de dispositivos digitales y su capacidad para ejecutar instrucciones en tiempo real. Esta práctica sienta las bases para futuros proyectos en los que se requiera la interacción con otros periféricos y el desarrollo de sistemas más complejos.

0.2. Objetivos

Los estudiantes deberán comprender los conceptos básicos de microprocesadores, microcontroladores y sistemas embebidos, incluyendo sus arquitecturas, funciones y capacidades. Posteriormente, analizarán el funcionamiento del microcontrolador ESP32 para lograr encender un LED y controlar su frecuencia de parpadeo utilizando Arduino IDE (Entorno de Desarrollo Integrado).

0.3. Introducción

En esta práctica, se explorará el uso del microcontrolador ESP32 (un dispositivo de bajo costo y alto rendimiento, ampliamente utilizado en proyectos de sistemas embebidos debido a su capacidad de procesamiento y conectividad) para controlar la frecuencia de encendido y apagado de un LED. Este ejercicio no solo tiene un valor práctico, sino que también nos introduce al manejo de entradas y salidas digitales en el contexto de sistemas embebidos.

A través de esta práctica, aprenderemos cómo el parpadeo de un LED con intervalos cambiantes puede utilizarse para mostrar diferentes estados de un sistema (como encendido, apagado o modo de espera). Además, el control de la frecuencia de parpadeo podría formar parte de un proyecto de efectos de iluminación, simulando efectos visuales como la respiración o la atenuación de luces.

La relevancia de esta práctica radica en que nos proporciona una comprensión fundamental de los conceptos del control de dispositivos electrónicos mediante microcontroladores. Este conocimiento es esencial en la programación de sistemas embebidos, un área clave en la ingeniería Robótica.

0.4. Marco teórico

En esta sección, se presentarán los conceptos fundamentales relacionados con el microcontrolador ESP32. Para comprender su funcionamiento, es necesario primero definir algunos términos clave como microprocesador, microcontrolador y su funcionamiento básico en proyectos de tecnología.

- **Microprocesador:** Es un chip constituido de una base principalmente de silicio en la que se encuentran millones de dispositivos electrónicos interconectados, tales como diodos, transistores y capacitores. Es el componente encargado de ejecutar instrucciones en un sistema digital. Estas instrucciones se procesan en lenguaje de bajo nivel, comprendiendo solo los valores binarios 0 y 1, lo que se denomina lenguaje máquina. (P., 2010)
En términos de arquitectura, los microprocesadores modernos manejan palabras de 32 o 64 bits, lo que determina la cantidad de datos que pueden procesar en una sola operación. La placa base o placa madre sirve de soporte físico tanto para el microprocesador como para otros chips, además de facilitar la comunicación entre ellos.
- **Microcontrolador:** Es un circuito integrado que en su interior contiene un gran número de componentes conectados entre sí de forma muy específica, entre los cuales debe incluir estos 3 elementos básicos: Microprocesador, diferentes tipos de memorias y diferentes plantillas de entrada y salida. Un microcontrolador es un microprocesador completo rodeado de ciertos bloques periféricos básicos, de modos que en un chip se puedan realizar muchas de las aplicaciones que normalmente requerirían de un microprocesador y por lo menos de dos o tres chips adicionales. (L., 2010)
- **Sistema embebido:** Hace referencia a todo circuito electrónico digital capaz de realizar operaciones de computación, generalmente en tiempo real, que sirven para cumplir una tarea específica en un producto.
El procesamiento de programación de un sistema embebido requiere inicialmente de un computador o Host, en el cual se ejecute una herramienta de software denominada IDE (Integrated Development Environment). El programa que ejecuta un sistema embebido es por lo general elaborado en el lenguaje como el ensamblador, ANSI, C, C++, o Basic (S., 2015)
- **LED:** Es un diodo emisor de luz, un dispositivo semiconductor de la familia de los diodos de unión p-n que emite luz cuando se polariza apropiadamente. La recombinación de huecos y electrones en la unión p-n libera energía, que en ciertos materiales semiconductores se emite en forma de fotones en lugar de calor. Dependiendo de la composición del material, un LED puede emitir LUZ en el espectro visible de distintos tonos o infrarrojo. (R., 2009)
- **El ESP32** Es un microcontrolador desarrollado por Espressif Systems que cuenta con conectividad Wi-Fi y Bluetooth de bajo costo y alto rendimiento. Puede encontrarse en dos presentaciones principales: como chip o como módulo, cada una con diferentes tamaños y números de pines. La elección de la forma del ESP32 depende del diseño y propósito del proyecto en el que se utilizará. Este microcontrolador es ampliamente utilizado en sistemas embebidos debido a su capacidad de procesamiento y versatilidad.

El entendimiento de estos conceptos es esencial para comprender cómo el ESP32 puede ser utilizado en proyectos de control y automatización.

En esta práctica, se emplearán estos conocimientos para controlar un LED, lo que constituye una aplicación básica pero fundamental para el manejo de entradas y salidas digitales en sistemas embebidos.

0.5. Metodología

MATERIALES Y HERRAMIENTAS

- Microcontrolador ESP32.
- Computadora con Arduino IDE instalado.
- Cable USB para cargar el código en el ESP32.

PROCEDIMIENTO

■ Configuración del entorno de desarrollo

- Se instaló Arduino IDE en la computadora.
- Se agregó la placa ESP32 en el Gestor de Tarjetas dentro de Arduino IDE.
- Se seleccionó el puerto y el modelo adecuado del ESP32 en las opciones de Arduino IDE.

■ Desarrollo del código en Arduino IDE

- Se definió el pin del LED como salida utilizando la función `pinMode(LED_PIN, OUTPUT)`; en `setup()`.
- Se escribió un bucle `for` para hacer parpadear el LED con intervalos crecientes y decrecientes.
- Se utilizó `digitalWrite(LED_PIN, HIGH)`; y `digitalWrite(LED_PIN, LOW)`; para encender y apagar el LED.
- Se implementó la función `delay(i)`; para variar el tiempo de encendido y apagado en un rango de 100 ms a 1000 ms.

■ Código diseñado para la ejecución de la práctica

```
#define LED_PIN 2

void setup() {
    pinMode(LED_PIN, OUTPUT);
}

void loop() {
    for(int i=100;i<1000;i+=100){
        digitalWrite(LED_PIN, HIGH);
        delay(i);
        digitalWrite(LED_PIN, LOW);
        delay(i);
    }
    for(int i=1000;i>0;i-=100){
        digitalWrite(LED_PIN, HIGH);
        delay(i);
        digitalWrite(LED_PIN, LOW);
        delay(i);
    }
}
```

■ Carga del código en el ESP32

- Se verificó y compiló el código en Arduino IDE.
- Se conectó el ESP32 a la computadora mediante USB y se subió el programa a la placa.
- Se presionó el botón EN (Reset) en la placa ESP32 si fue necesario para iniciar la ejecución del código.

■ Pruebas y observaciones

- Se observará el comportamiento del LED para confirmar que parpadeaba con tiempos crecientes y decrecientes.
- Se verificó que el código cumpliera con los objetivos esperados y se realizaron ajustes si fuese necesario.

0.6. Resultados

Tras la ejecución del código en el ESP32, se obtuvieron los siguientes resultados:

■ Funcionamiento del código:

- Se logró cargar y ejecutar correctamente el código en la placa ESP32 mediante Arduino IDE.
- El LED se encendió y apagó de manera controlada, respetando los tiempos de espera definidos en el código.
- Se observará una variación en la frecuencia de parpadeo, aumentando y luego disminuyendo progresivamente.

■ Comportamiento del sistema:

- Cuando `delay(i)`; aumentó desde 100 ms hasta 1000 ms, el LED parpadeó más lentamente.
- Cuando `delay(i)`; disminuyó desde 1000 ms hasta 100 ms, el LED parpadeó más rápidamente.
- El código demostró ser eficiente y estable, sin errores de ejecución en el ESP32.

■ Dificultades encontradas y ajustes realizados:

- En algunos casos, fue necesario presionar el botón en (Reset) del ESP32 para cargar correctamente el programa.
- Se verificó la correcta configuración del puerto COM en Arduino IDE para evitar errores de conexión.
- Se realizaron pequeñas modificaciones en los valores de `delay()` para mejorar la visualización del parpadeo del LED.

0.7. Conclusión

La práctica nos permitió familiarizarnos con el uso del ESP32 y su integración con Arduino IDE, sentando las bases para el desarrollo de proyectos de sistemas embebidos. Se comprendió el manejo de salidas digitales, utilizando `digitalWrite()` para controlar un LED de forma programada, lo que nos permitió observar cómo se pueden generar diferentes patrones de parpadeo variando los tiempos de encendido y apagado.

Además, a lo largo de la práctica, se evidencia la importancia de una configuración adecuada del entorno de desarrollo, ya que la correcta instalación de las herramientas y la configuración del puerto COM fueron factores clave para la correcta ejecución del código. La necesidad de presionar el botón de reinicio en algunos casos también destacó la importancia de comprender el hardware del ESP32 y sus particularidades en la carga de programas.

Finalmente, la experiencia adquirida en esta práctica será de gran utilidad en proyectos futuros que requieran la implementación de sistemas embebidos en aplicaciones reales. Se ha demostrado que el ESP32 es una potente y flexible para el desarrollo de soluciones tecnológicas, lo que refuerza su importancia en el aprendizaje y aplicación de sistemas electrónicos y de control.

Bibliografía

L., P. (2010). *Microprocesadores*. RED TERCER MILENIO S.C., 1ra edition.

P., Q. (2010). *Aquitectura de computadoras*. Alpha Editorial, 1ra edition.

R., B. (2009). *Electrónica: Teoría de Circuitos y Dispositivos Eléctricos*. PEARSON EDUCACIÓN, 10ma edition.

S., S. (2015). *Todo sobre sistemas embebidos*. UPC, 1ra edition.