INSTITUTO POLITÉCTICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

ESCOM

Carrera:

Ingeniería en sistemas computacionales

Unidad de Aprendizaje:

Sistemas embebidos | IoT

Práctica

ESP8266 Led

Integrantes:

Cazares Cruz Jeremy Sajid

Acosta Cortes Gerardo Michell

Bazo Machuca Ilse Paola

Guzman Ballesteros Armando

Fecha de entrega

19 de diciembre de 2023

ÍNDICE

INTRODUCCIÓN

En esta práctica, nos enfocaremos en explorar y comprender en profundidad el funcionamiento del módulo ESP8266, una poderosa herramienta en el campo de la Internet de las Cosas (IoT). Nuestro objetivo principal es aprender a controlar un diodo LED, no solo el integrado en la placa ESP8266, sino también uno externo, utilizando un pin de propósito general del módulo.

Para lograr este control, emplearemos una interfaz de usuario basada en HTML. Esta interfaz estará alojada directamente en el ESP8266, aprovechando la capacidad del sistema de archivos SPIFFS (Sistema de Archivos SPI Flash) del módulo. Esta característica nos permite cargar páginas web y otros recursos directamente en la memoria de la placa, lo que facilita una interacción más dinámica y flexible con el hardware, sin necesidad de incluir el código HTML directamente en el código fuente de Arduino.

El uso de SPIFFS es un punto clave en nuestra práctica, ya que nos permite separar la lógica de programación del diseño y la presentación de la interfaz de usuario, lo que resulta en un proyecto más modular y fácil de mantener. Además, nos brinda la oportunidad de explorar cómo las aplicaciones web pueden interactuar con el hardware en tiempo real, un concepto crucial en el desarrollo de soluciones IoT.

A lo largo de esta práctica, cubriremos aspectos fundamentales como la configuración inicial del ESP8266, la programación del pin GPIO para controlar el LED y el desarrollo de una interfaz web sencilla pero funcional. También discutiremos cómo cargar archivos en la memoria SPIFFS y cómo estos archivos se utilizan para interactuar con el hardware del ESP8266.

MARCO TEÓRICO

ESP8266: Microcontrolador con Wi-Fi

El ESP8266 es un microcontrolador con capacidades de Wi-Fi, desarrollado por Espressif Systems. Se destaca por su bajo costo y facilidad de uso, lo que lo hace popular en aplicaciones de Internet de las Cosas (IoT). Este chip permite a los dispositivos electrónicos conectarse y comunicarse a través de redes Wi-Fi. Es comúnmente usado en proyectos de automatización del hogar y dispositivos de sensores inalámbricos. Además, es compatible con la plataforma Arduino, ofreciendo un entorno de programación accesible y una amplia gama de bibliotecas para su uso.

SPIFFS: Almacenamiento de Datos en el ESP8266

SPIFFS, o Sistema de Archivos SPI Flash, es una funcionalidad del ESP8266 que proporciona un método para almacenar archivos en la memoria flash del dispositivo. Esta característica es útil para alojar páginas web, configuraciones y otros datos necesarios para las aplicaciones IoT. SPIFFS facilita la gestión de archivos de manera eficiente y segura, permitiendo un mayor control y flexibilidad en el diseño y funcionamiento de proyectos basados en el ESP8266.

Pines del ESP8266: Interfaz de Hardware

Los pines del ESP8266 son los puntos de conexión física que permiten al microcontrolador interactuar con otros dispositivos y sensores. Estos pines incluyen entradas y salidas digitales, pines analógicos, así como conexiones para comunicación serial. Su configuración y programación son fundamentales para controlar y recibir datos de sensores, actuadores y otros componentes electrónicos. La correcta utilización de estos pines es esencial para maximizar las capacidades del ESP8266 en proyectos de IoT.

DESARROLLO

Como se mencionó anteriormente en el desarrollo de esta práctica usaremos el ESP8266 para encender un led externo para esto tenemos el siguiente código para el ESP, el cuál iremos explicando paso a paso

```
#include <ESP8266WiFi.h>

#include <ESP8266WebServer.h>

#include <FS.h>
```

Tenemos primeramente la inclusión de las bibliotecas donde cada una sirve de lo siguiente:

- ESP8266WiFi.h: Biblioteca para manejar la conectividad WiFi del ESP8266.
- ESP8266WebServer.h: Biblioteca que permite crear un servidor web en el ESP8266.
- FS.h: Biblioteca para el manejo del sistema de archivos SPIFFS, usado para almacenar y acceder a archivos como páginas web.

Para la configuración del Wi-Fi tenemos lo siguiente:

```
const char* ssid = "kamehouse";

const char* password = "Oliver-jeremy-051177";
```

Establece el nombre (ssid) y la contraseña (password) de la red Wi-Fi a la cual el ESP8266 se conectará.

Para inicializar el servidor web de la ESP8266 tenemos lo siguiente:

```
ESP8266WebServer server(80);

const int ledExternoPin = 2; // Pin para el LED externo
```

Crea una instancia de un servidor web en el puerto 80.

Define el pin para el LED externo.

Función setup()

Se ejecuta una vez al inicio del programa:

Inicialización de la Comunicación Serial: Serial.begin(115200);.

Conexión a Wi-Fi: Se conecta a la red Wi-Fi especificada y espera hasta que la conexión sea exitosa. Inicialización de SPIFFS: Se inicia el sistema de archivos SPIFFS. Configuración de Pines: Se configura el LED interno y el LED externo como salidas. Configuración del Servidor Web: Se establecen las rutas y las funciones que manejarán las peticiones a esas rutas.

Funciones handleRoot() y Otros handle...()

handleRoot(): Maneja las peticiones a la ruta raíz ("/"). Abre y envía el archivo index.html al cliente.

handleEncenderInterno() y handleApagarInterno(): Encienden y apagan el LED interno, respectivamente.

handleEncenderExterno() y handleApagarExterno(): Encienden y apagan el LED externo, respectivamente.

Función loop()

Se ejecuta continuamente después de setup(): server.handleClient(): Maneja las peticiones entrantes y ejecuta las funciones correspondientes definidas en setup().

Funcionamiento General

El ESP8266 se conecta a una red Wi-Fi y configura un servidor web. A través de una interfaz web (usualmente index.html), el usuario puede enviar comandos para controlar los LEDs. Las peticiones son recibidas por el servidor y procesadas, resultando en la activación o desactivación de los LEDs.

El HTML para realizar las acciones de encender el led interno de la esp es el siguiente:

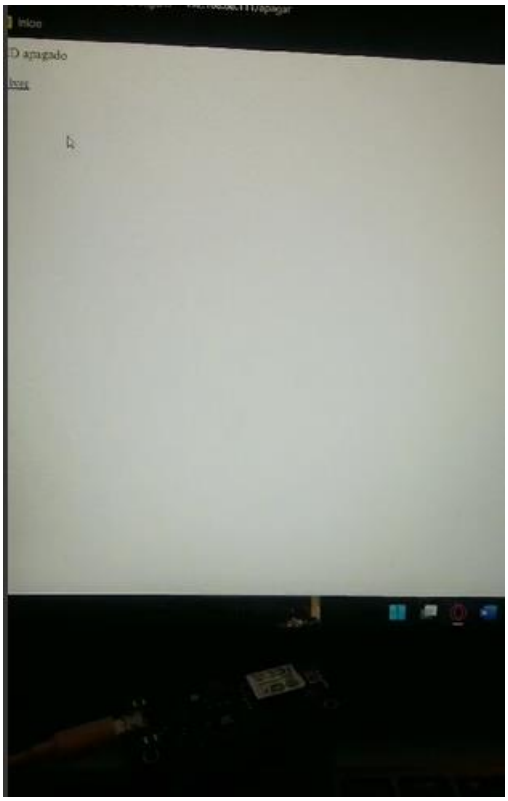
<!DOCTYPE html>

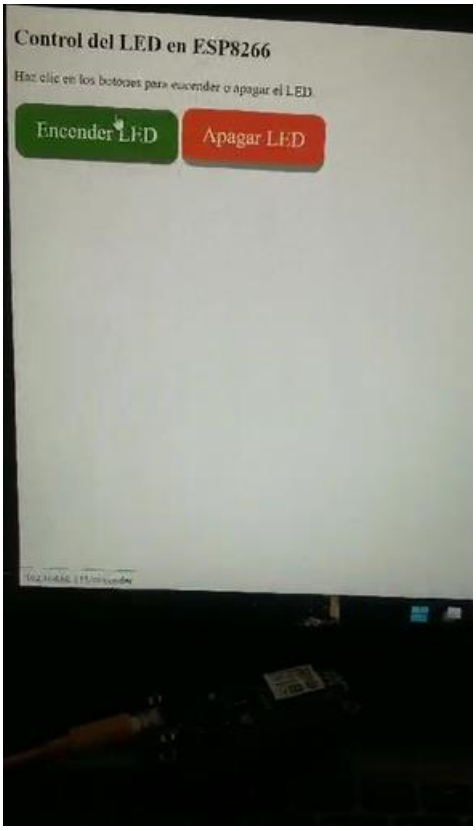
```
<html>
<head>
  <title>Control LED ESP8266</title>
  <style>
    .button {
      display: inline-block;
      padding: 15px 25px;
      font-size: 24px;
      cursor: pointer;
      text-align: center;
      text-decoration: none;
      outline: none;
      color: #fff;
      background-color: #4CAF50;
      border: none;
      border-radius: 15px;
      box-shadow: 0 9px #999;
    }

    .button:hover {background-color: #3e8e41}
```

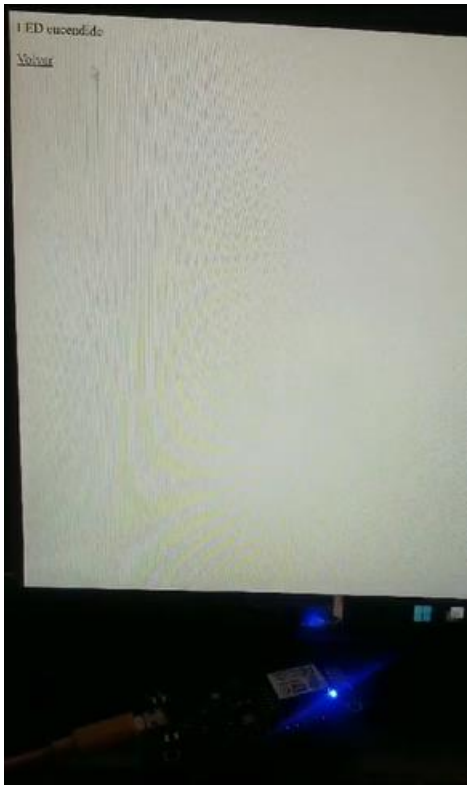
```
        .button:active {
            background-color: #3e8e41;
            box-shadow: 0 5px #666;
            transform: translateY(4px);
        }

        .button-red {
            background-color: #f44336;
        }

        .button-red:hover {background-color: #d73833}
    </style>
</head>
<body>
    <h2>Control del LED en ESP8266</h2>
    <p>Haz clic en los botones para encender o apagar el LED.</p>
    <a href="/encender" class="button">Encender LED</a>
    <a href="/apagar" class="button button-red">Apagar LED</a>
</body>
```

</html>

Finalmente, las pruebas de esto es lo siguiente:




Control del LED en ESP8266
Haz clic en los botones para encender o apagar el LED
Encender LED
Apagar LED


LED encendido
Volver

ESP8266 con Led externo

Inclusión de Bibliotecas

#include <ESP8266WiFi.h>

#include <ESP8266WebServer.h>

#include <FS.h>

Estas líneas incluyen las bibliotecas necesarias para manejar la conectividad WiFi (ESP8266WiFi.h), crear un servidor web (ESP8266WebServer.h) y trabajar con el sistema de archivos SPIFFS (FS.h).

Configuración de WiFi

const char* ssid = "kamehouse";

const char* password = "Oliver-jeremy-051177";

Aquí se define el SSID y la contraseña de la red WiFi a la que se conectará el ESP8266.

Creación del Servidor Web y Configuración del Pin

ESP8266WebServer server(80);

const int ledExternoPin = 2;

Se crea una instancia del servidor web en el puerto 80 y se define el pin del LED externo.

Función setup()

Esta función se ejecuta una vez al iniciar el programa:

Inicializa la comunicación serial.

Conecta el ESP8266 a la red WiFi especificada.

Inicia el sistema de archivos SPIFFS.

Configura los pines del LED interno y externo como salidas.

Define las rutas del servidor web y las funciones que responderán a las solicitudes a esas rutas.

Inicia el servidor web.

Manejadores de Rutas del Servidor Web

void handleRoot() {...}

void handleEncenderInterno() {...}

void handleApagarInterno() {...}

void handleEncenderExterno() {...}

void handleApagarExterno() {...}

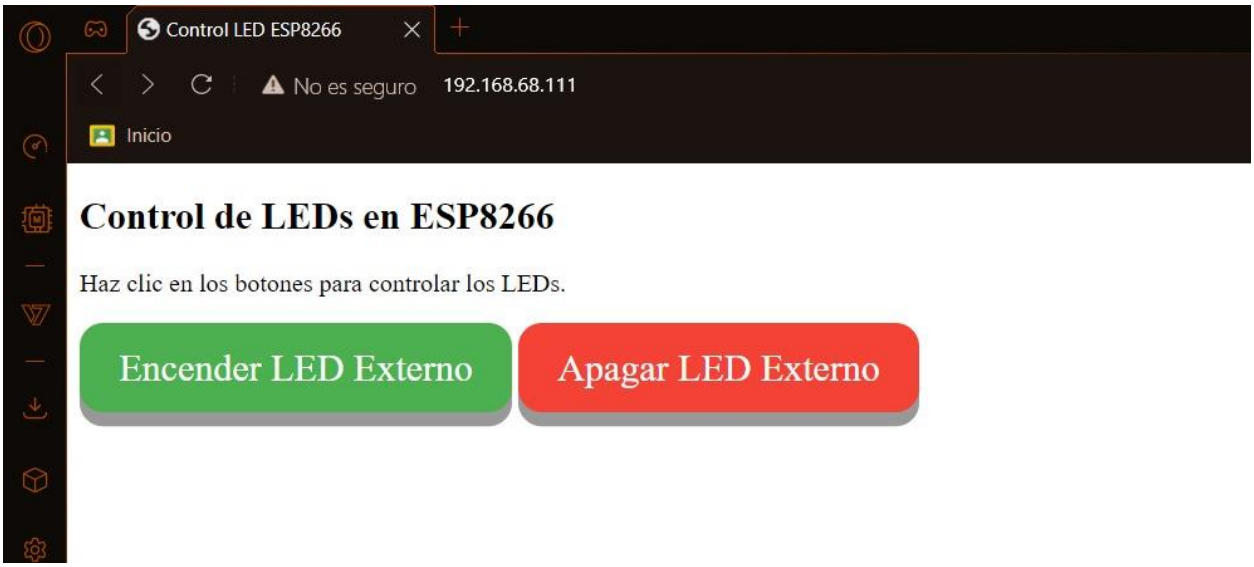
Cada una de estas funciones maneja una ruta específica del servidor web. Por ejemplo, handleRoot() maneja las solicitudes a la raíz del servidor ("/") y envía el archivo index.html. Las demás funciones controlan el encendido y apagado de los LEDs interno y externo.

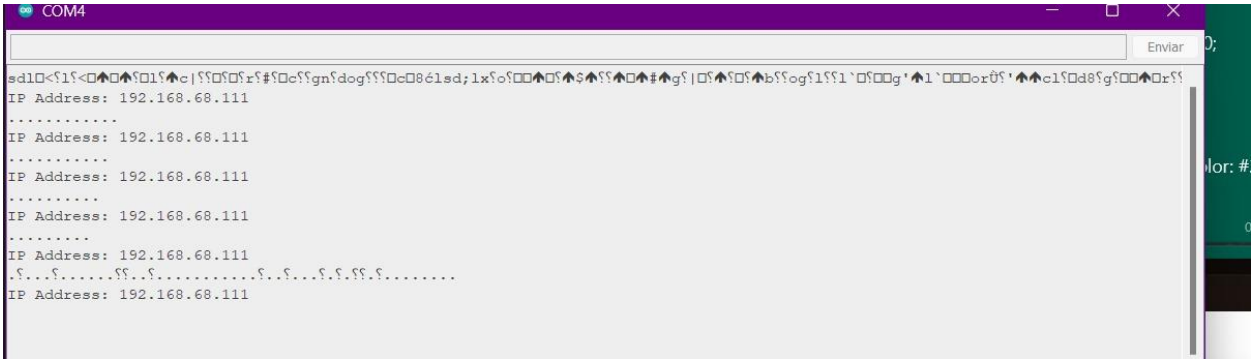
Función loop()

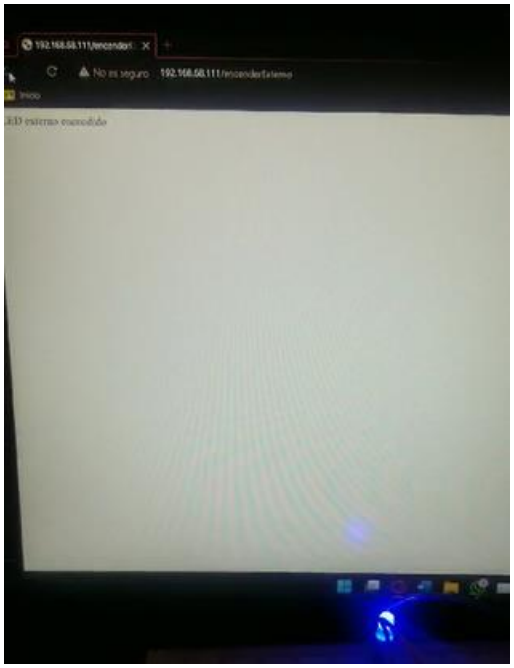
Esta función se ejecuta continuamente y maneja las solicitudes entrantes al servidor web.

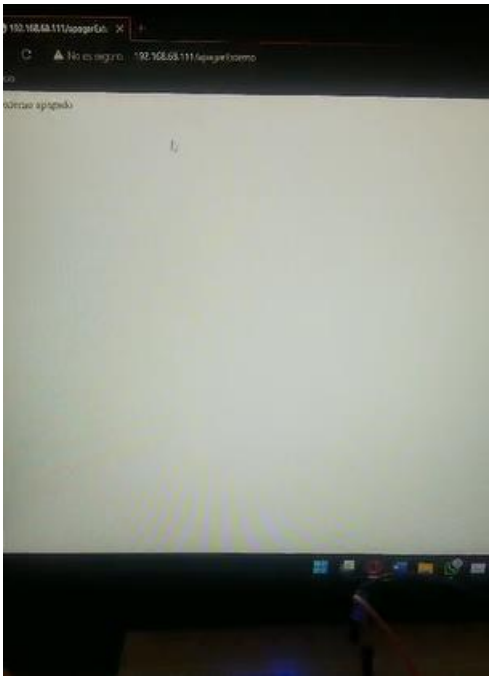
Funcionamiento General

El programa convierte el ESP8266 en un servidor web que permite controlar un LED interno y un LED externo. Cuando un usuario accede a una ruta específica (por ejemplo, "/encenderInterno"), el servidor ejecuta la función correspondiente para encender o apagar el LED indicado. El estado del LED se controla mediante las funciones digitalWrite(), que activan o desactivan los pines correspondientes al LED interno y externo.


Control LED ESP8266
No es seguro
192.168.68.111
Inicio
Control de LEDs en ESP8266
Haz clic en los botones para controlar los LEDs.
Encender LED Externo
Apagar LED Externo


COM4
Enviar
IP Address: 192.168.68.111
IP Address: 192.168.68.111
IP Address: 192.168.68.111
IP Address: 192.168.68.111
IP Address: 192.168.68.111
IP Address: 192.168.68.111





CONCLUSIONES

En conclusión, el código proporcionado demuestra eficazmente cómo el ESP8266, un microcontrolador versátil con capacidades Wi-Fi, puede ser utilizado para crear aplicaciones interactivas en el ámbito del Internet de las Cosas (IoT). Mediante la inclusión de bibliotecas específicas, este programa establece una red Wi-Fi, configura un servidor web y controla componentes electrónicos, en este caso, un LED interno y un LED externo.

La habilidad de integrar funciones de servidor web en un dispositivo como el ESP8266 abre puertas a una multitud de aplicaciones IoT, permitiendo no solo la automatización local sino también el control remoto y la monitorización a través de la red. El uso del sistema de archivos SPIFFS para alojar la interfaz de usuario (como un archivo HTML) simplifica la interacción con el usuario final y hace que el sistema sea más robusto y fácil de manejar.

Este ejemplo específico ilustra la capacidad del ESP8266 para manejar solicitudes de red y responder mediante la activación física de componentes electrónicos, ofreciendo una introducción práctica y efectiva al desarrollo de soluciones IoT. Además, el código es un excelente punto de partida para aquellos interesados en explorar más profundamente la automatización del hogar, la robótica y otros proyectos donde el control inalámbrico es esencial.