



Ingeniería en Robótica y Telecomunicaciones
Departamento de computación, electrónica y
mecatrónica.

Course Name:

TEMAS SELECTOS 2-023-LRT4032-1

Members & ID:

André Federico López Hernández - 167564

Jonathan Eliasib Rosas Tlaczani-168399

Proyecto WebServer (Segundo parcial)

1.- Video:

Primer intento:

https://youtu.be/qSFwhCWE4U?si=kz_RPbguRy-90Krv

Video Final de sistema:

<https://youtu.be/6THzpcdC8VI?si=KBaLHRoEKDgE1N7p>

2. Reporte

Planteamiento del problema.

Diseñar un sistema IoT basado en ESP32 y HTML que cumpla con lo siguiente:

- Encendido de un foco de 120Vac asistido por un relevador.
 - Encendido de un foco de 120Vac asistido por un optotriac.
 - Encendido de un foco de 120Vac asistido por un relevador. > El control de encendido se debe realizar al aplaudir.
- El control de encendido se realiza mediante botones en un servidor web.

Objetivo

El objetivo de este proyecto es diseñar un sistema IoT (Internet de las Cosas) que permita el control de encendido de focos de 120Vac utilizando diferentes métodos: a través de un relevador, un optotriac y por medio de la detección de aplausos. El control se realizará mediante una interfaz web accesible desde cualquier dispositivo con un navegador.

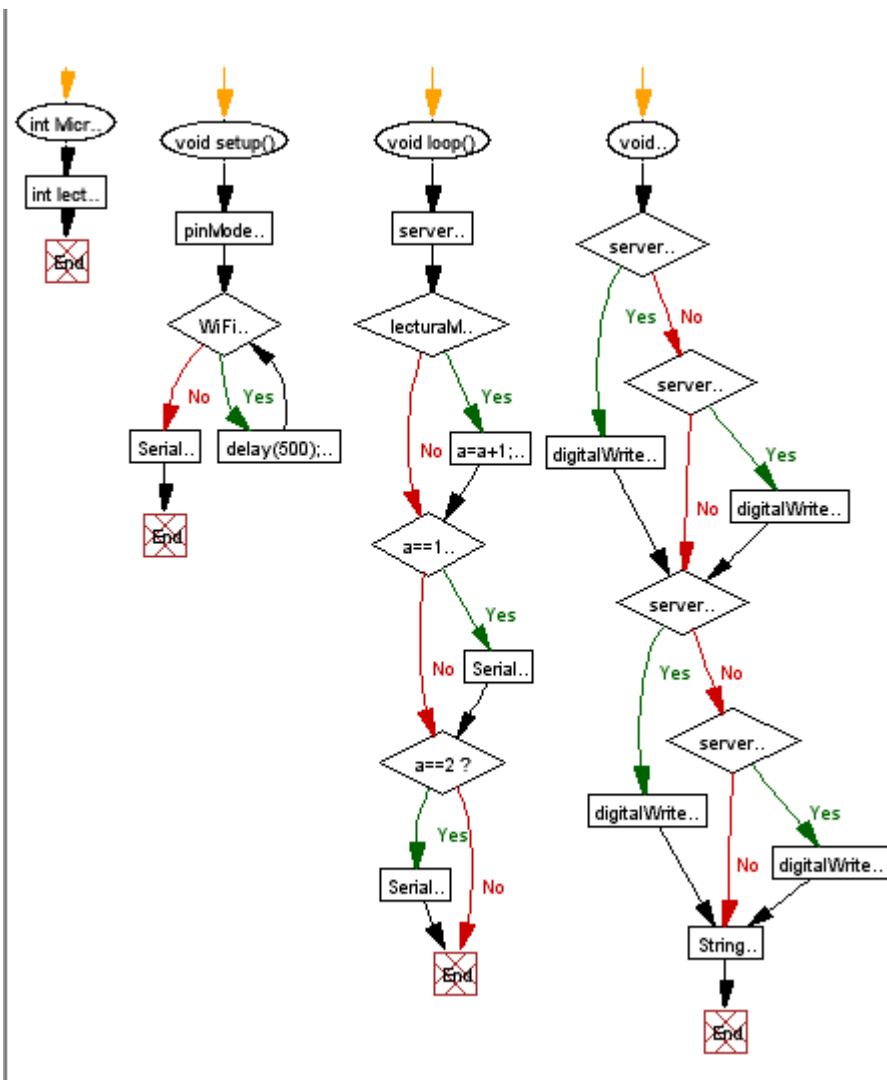
Descripción del Problema

En la actualidad, la automatización del hogar y la gestión eficiente de la energía son temas de gran relevancia. El control de dispositivos eléctricos a distancia es una necesidad común en muchas aplicaciones. En este proyecto, nos enfrentamos al desafío de diseñar un sistema que permita encender focos de 120Vac de tres maneras diferentes:

1. Relevador: Controlar el encendido del foco mediante un relevador, que se activa y desactiva a través de una interfaz web.
2. Optotriac: Utilizar un optotriac para controlar el encendido del foco desde la interfaz web.
3. Aplauso: Activar el encendido del foco al detectar un aplauso.

Diagrama de flujo del programa de control.





Circuitos esquemáticos

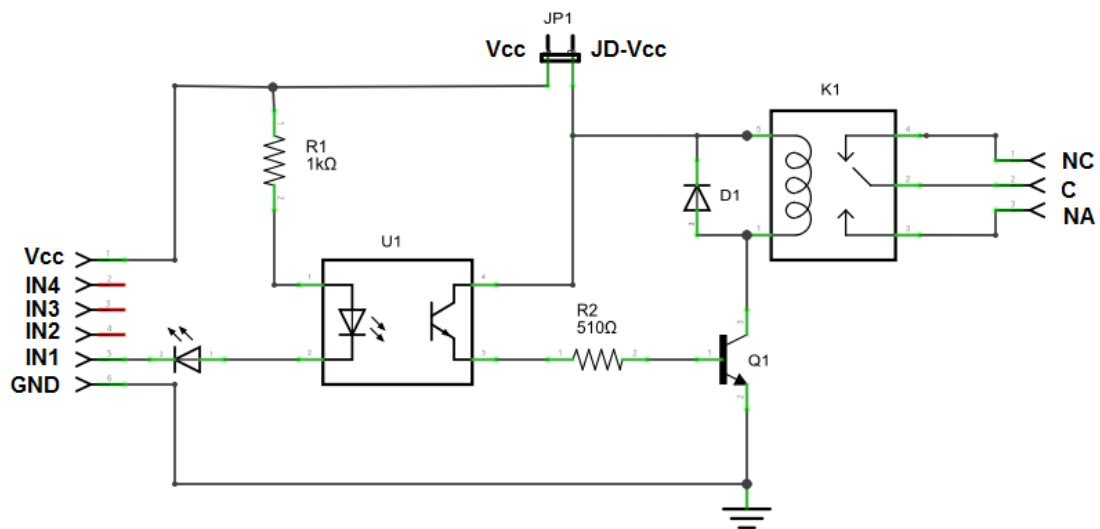


Fig. 1 circuito esquemático de un canal del modulo de relevadore, el resto de los canales repite la misma configuración.

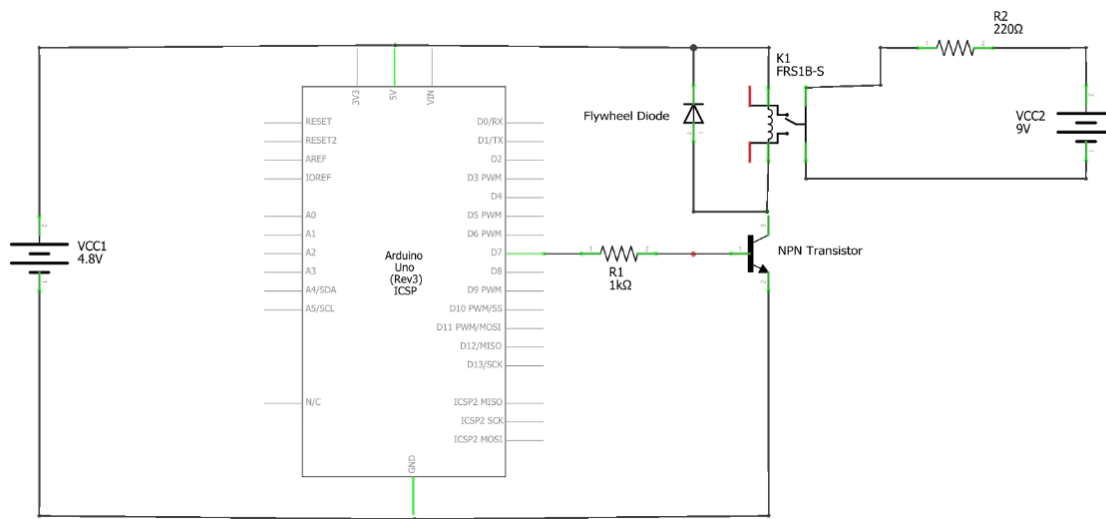


Fig. 2 circuito que muestra cómo controlar un relé con su microcontrolador Arduino o ESP32

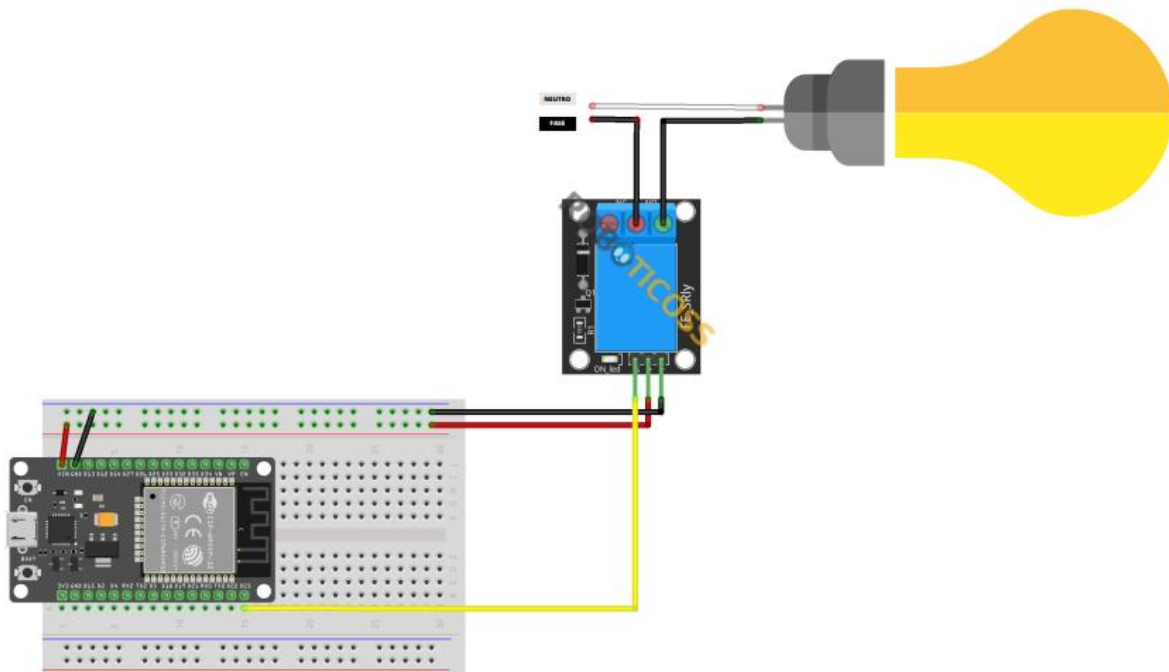


Fig. 3 Conexiones físicas para circuito de encendido de foco con rele

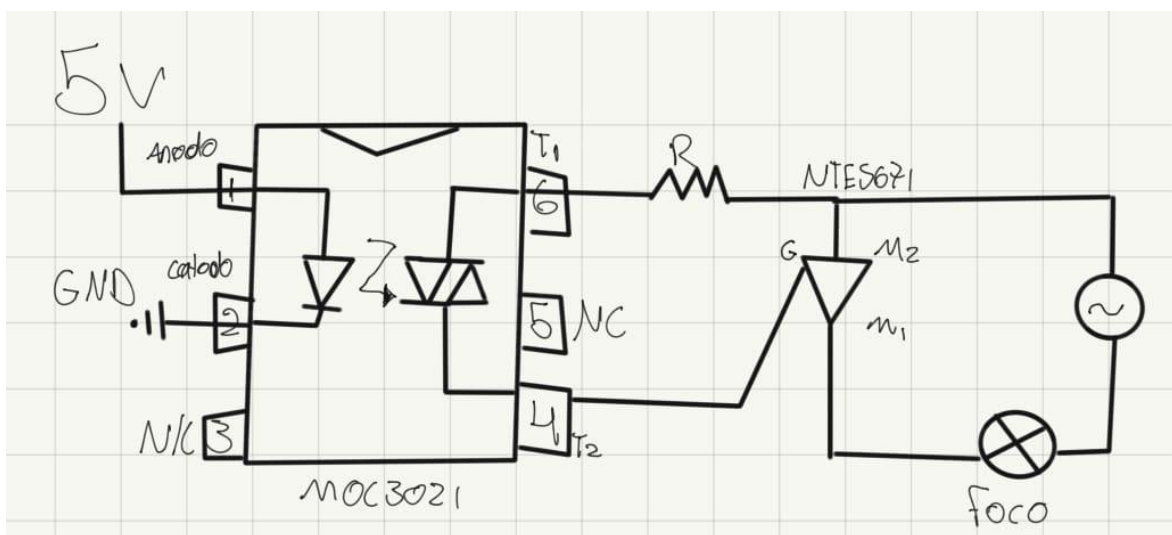


Fig. 4 circuito de encendido de foco asistido por optotriac

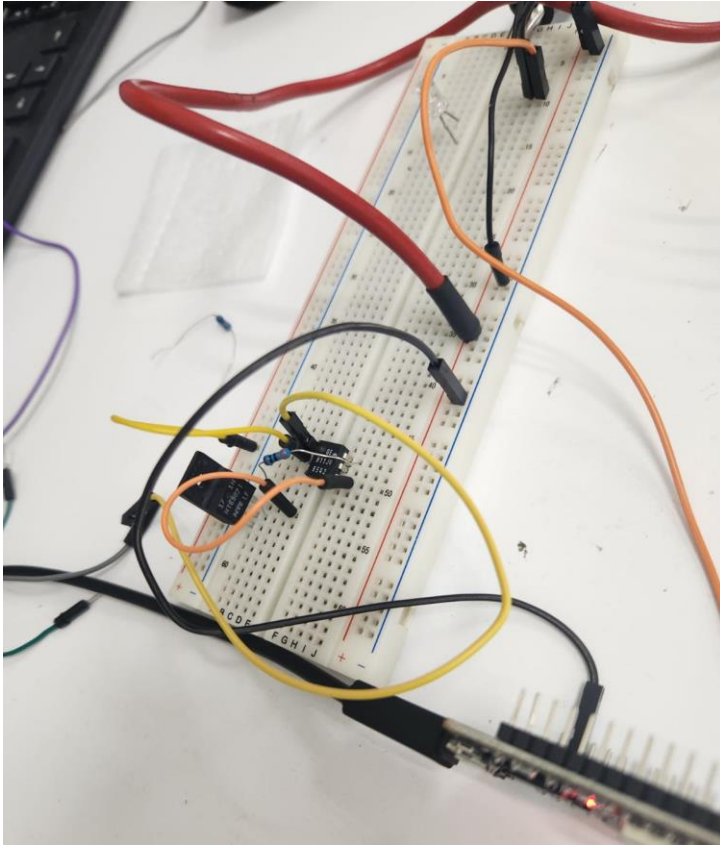


Fig. 5 circuito físico del optoacoplador

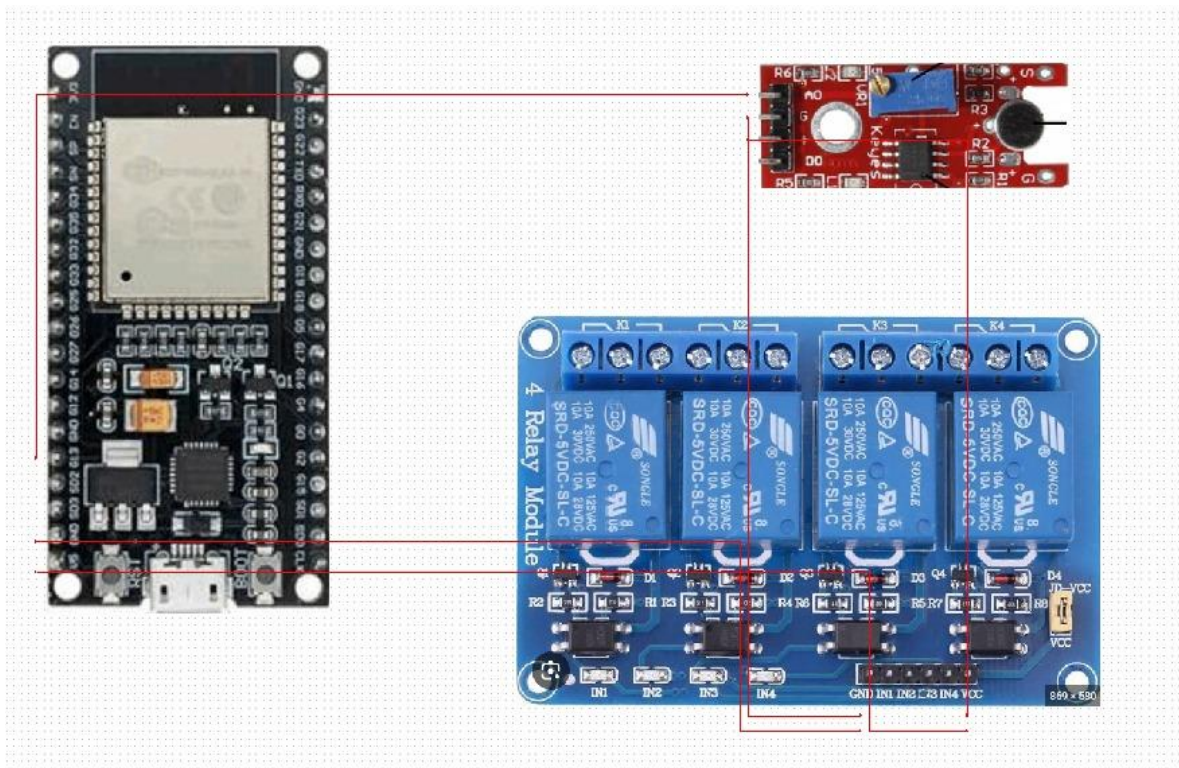


Fig. 6 circuito de encendido de foco con rele y sensor de sonido

Listas de materiales.

- Placa Arduino ESP32
- Protoboard
- 3 sockets
- 3 bombillas o focos
- Optotriac MOC3020
- Módulo de 4 relevadores de 5V
- 2 resistencias de 330ohms
- Triac NTE5671
- Sensor de sonido KY-037

Experiencia adquirida, problemas resueltos.

Durante la ejecución de este proyecto, hemos adquirido una valiosa experiencia en diversas áreas, incluyendo:

Desarrollo de Hardware

- Diseño y montaje de circuitos electrónicos para controlar el relevador y el optotriac.

- Selección y configuración de componentes como sensores de sonido para detectar aplausos.

Programación de Microcontroladores

- Programación del ESP32 para interactuar con los dispositivos de hardware.
- Desarrollo de firmware para gestionar la comunicación con una red Wi-Fi.
- Creación de aplicaciones web embebidas en el ESP32 para el control de dispositivos.

Diseño de Interfaz Web

- Diseño y desarrollo de una interfaz web amigable para el usuario que permita controlar los dispositivos y visualizar el estado.
- Implementación de funciones de control remoto de los focos a través de la interfaz web.

Solución de Problemas

Durante el proceso de desarrollo, nos enfrentamos a varios desafíos, como:

- Resolución de problemas de conectividad y configuración de la red Wi-Fi en el ESP32.
- Ajuste de sensibilidad de los sensores de sonido para detectar aplausos de manera confiable.
- Manejo de problemas de seguridad en la interfaz web para garantizar el acceso seguro y autorizado.

Trabajo a futuro e innovación.

Mejoras y Trabajo a Futuro

Para mejorar y expandir este proyecto, se pueden considerar las siguientes áreas:

- Integración de más dispositivos: Agregar la capacidad de controlar y monitorear más dispositivos y electrodomésticos en el hogar.
- Optimización de la eficiencia energética: Implementar algoritmos de gestión de energía para optimizar el consumo eléctrico de los focos.
- Seguridad Avanzada: Incorporar medidas de seguridad avanzadas, como autenticación de usuarios y cifrado de datos, para proteger la interfaz web.
- Integración de Asistentes Virtuales: Habilitar la integración con asistentes virtuales como Amazon Alexa o Google Assistant para comandos de voz.

Innovación

La innovación en este proyecto podría llevarse a cabo mediante:

- Machine Learning y Reconocimiento de Voz: Implementar un sistema de reconocimiento de voz que permita controlar los dispositivos mediante comandos de voz.
- Integración de Sensores Ambientales: Utilizar sensores ambientales para monitorear la calidad del aire, la temperatura y la humedad en el entorno.
- Integración con Plataformas IoT en la Nube: Conectar el sistema a una plataforma de IoT en la nube para un monitoreo y control más avanzado y accesible desde cualquier lugar.

En resumen, este proyecto ha proporcionado una experiencia valiosa en el desarrollo de sistemas IoT, y presenta oportunidades emocionantes para la mejora y la innovación en el futuro. La automatización del hogar y la gestión eficiente de la energía son áreas en constante evolución, y este proyecto puede contribuir al avance en estas áreas.

3.- Códigos de programación.

```
#include <WiFi.h>
#include <WebServer.h>

int Microfono=14; //MICROFONO EN PIN 14
int lecturaMicrofono=0; //GUARDAMOS LA LECTURA DEL MICROFONO PIN 27
int Rele=27; //RELE EN pin 23
int a=0; //BANDERA QUE CONTARA EL NUMERO DE APLAUSOS
int b=0; //BANDERA PARA SALIR DE BUCLE

const char* ssid = "jonyrosni";
const char* password = "123456789";

WebServer server(80);

//const int ledPin = 2;

void setup() {
  pinMode(22, OUTPUT);
  pinMode(23, OUTPUT);
  pinMode(Microfono, INPUT); // MICROFONO COMO ENTRADA DE SEÑAL
  pinMode (Rele,OUTPUT); //RELE COMO SALIDA DE SEÑAL
```

```

Serial.begin(115200);
Serial.println("Espere, conectando");
Serial.println(ssid);

WiFi.mode(WIFI_STA);
WiFi.begin(ssid,password);
Serial.println("");

while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}

Serial.println("");
Serial.print("WiFi conectado a: ");
Serial.println(ssid);
Serial.print("Direccion IP asignada por el router: ");
Serial.println(WiFi.localIP());

server.on("/",inicio);

server.begin();
Serial.println("Servidor HTTP iniciado");
}

void loop() {
    server.handleClient();
    lecturaMicrofono=digitalRead (Microfono); //LEEMOS EL ESTADO QUE TIENE EL
MICROFONO

    if (lecturaMicrofono==HIGH) //SI TENEMOS UN APLAUSO?
    {
        a=a+1; // A=0+1=1 TENEMOS UN APLAUSO Y SEGUIRA AUMENTANDO CON CADA APLUSO
        b=0; //B=0 BANDERA PARA ENTRAR A BUCLES
        delay(200);
    }
    if(a==1 && b==0) // SI TENEMOS 1 APLAUSO Y B=0 ENTONCES ENCENDEMOS EL FOCO
    {
        Serial.println("ENCENDIDO");
        digitalWrite(Rele,HIGH); //ENCENDEMOS FOCO
        b=1; //B=1 PARA SALIR DE BUCLE Y PODER CAMBIAR
    }

    if (a==2) //SI TENEMOS 2 APALUSOS?
    {

```

```

        Serial.println ("APAGADO"); //APAGAMOS
        digitalWrite(Rele,LOW);
        a=0; //REGRESAMOS EL CONTADOR A 0
    }
}

void inicio () {
    if (server.hasArg("foco1") && server.arg("foco1") == "on"){
        digitalWrite(22,HIGH);
    }
    else if (server.hasArg("foco1") && server.arg("foco1") == "off"){
        digitalWrite(22,LOW);
    }
    if (server.hasArg("foco2") && server.arg("foco2") == "on"){
        digitalWrite(23,HIGH);
    }
    else if (server.hasArg("foco2") && server.arg("foco2") == "off"){
        digitalWrite(23,LOW);
    }
}

// Pagina HTML
String html = "<html><body>";
html += "<head><meta name=\"viewport\" content=\"width=device-width,
initial-scale=1.0, user-scalable=no\">\n";
html += "<title>Control LED HTML con webserver y ESP32 </title>\n";
html += "<h1> UDLAP </h1>";
html += "<h1> Foco1 control </h1>";
html += "<form method='get'>";
html += "<button type='submit' name='foco1' value='on'>ON</button>";
html += "<button type='submit' name='foco1' value='off'>OFF</button>";
html += "<h1> Foco2 control </h1>";
html += "<button type='submit' name='foco2' value='on'>ON</button>";
html += "<button type='submit' name='foco2' value='off'>OFF</button>";
html += " </form>";
html += "</body></html><html>";
server.send(200, "text/html", html);
}

```