

CONTROL DE ENCENDIDO MEDIANTE GRUPO EN TELEGRAM

CONTESTA LOS SIGUIENTES PUNTOS SEGÚN TUS CONOCIMIENTOS PREVIOS.

1.- ¿Qué es un grupo en Telegram?

2.- ¿Qué funciones tiene el administrador de un grupo en Telegram?

3.- ¿Qué son los comandos?

4.- ¿Qué es la API de Telegram?

5.- ¿Qué es el BotToken de Telegram?

6.- ¿Cómo debe ser el Chat_ID para que todos los integrantes del equipo puedan comunicarse con la tarjeta Wemos D1 R1?

ELABORACIÓN DEL CÓDIGO PARA CONTROLAR EL ENCENDIDO DEL LED 2 DE WEMOSD1 R1

1.- Explica la lógica como está implementado el LED construido en la tarjeta Wemos D1 R1



2.- Crea un grupo de Telegram e invita al maestro y a algunos compañeros.

3.- Una vez creado el grupo, pulsa clic en el nombre del grupo para agregar el bot, por ejemplo el mío se llama Oswagar_bot.

4.- Para interactuar con el grupo de Telegram, la Wemos D1 necesita conocer el ID del grupo de telegram. En tu cuenta de Telegram, abre tu grupo. El ID de grupo debe estar en la URL como se muestra a continuación.



5.- Guarda el ID del grupo, lo necesitaremos en el código.

ARCHIVO main.cpp

```
#include <Arduino.h>
#ifdef ESP32
    #include <WiFi.h>
#else
    #include <ESP8266WiFi.h>
#endif
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>
// Te recomiendo consultar la library: Universal Telegram Bot Library escrita por Brian Lough:
// https://github.com/witnessmenow/Universal-Arduino-Telegram-Bot
#include <ArduinoJson.h>

const char* ssid = "MEGACABLE-F732A";
const char* password = "w354KrMd";

// Inicializa Telegram BOT
// con el mismo BOTtoken de ejemplo anterior
#define BOTtoken "2134156332:AAHm2CaCrdDtD21gg6gj5G7wTYIOmbPN5_Y" // tu Bot
// Token (se obtiene de Botfather)
// En el grupo de telegram incluir como miembro al bot que creaste en el ejemplo anterior,
// por ejemplo el mío es: Oswagar_bot

// Usa @myidbot para encontrar el ID de chat para forma individual o de grupo
// También observa que debes dar click a "inicio" en el bot antes de poder
// enviarte mensajes
// El CHAT_ID debe iniciar con el caracter -
#define CHAT_ID "-632893036"

#ifdef ESP8266
    X509List cert(TELEGRAM_CERTIFICATE_ROOT); // Certificado raíz de Telegram
#endif
// Crea un nuevo cliente WiFi con WiFiClientSecure:
WiFiClientSecure client;
// Crea un bot con el token y el cliente definidos anteriormente.
UniversalTelegramBot bot(BOTtoken, client); // API de Telegram conectada con mi BOTtoken

// Verificar nuevos mensajes cada segundo.
int RetardoSolicitudBot = 1000;
/* RetardoSolicitudBot y UltimoArranque se utilizan para comprobar si hay nuevos mensajes
de Telegram cada x número de segundos. En este caso, el código buscará nuevos mensajes
cada segundo (1000 milisegundos). Puede cambiar ese tiempo de retraso en la variable
RetardoSolicitudBot. */
unsigned long UltimoArranque;

const int ledPin = 2; // Lógica invertida
bool EstadoDeLed = HIGH; //Apagado LED GPIO2 en Wemos D1 R1
```

```

// Manejar lo que sucede cuando tu recibes nuevos mensajes
void manejarNuevosMensajes(int numNuevosMensajes) {
    Serial.println("Se maneja un nuevo mensaje");
    Serial.println(String(numNuevosMensajes));
    // verifica los mensajes disponibles:
    for (int i=0; i<numNuevosMensajes; i++) {
        /* ID de Chat del solicitante:
        Obtener el ID de chat para ese mensaje en particular y se
        guarda en la variable chat_id. El ID de chat nos permite identificar
        quién envió el mensaje */
        String chat_id = String(bot.messages[i].chat_id);
        /*Si el chat_id es diferente de su ID de grupo de chat (CHAT_ID), significa
        que alguien (que no está en el grupo) ha enviado un mensaje a su bot.
        Si ese es el caso, ignore el mensaje y espere el siguiente. */
        if (chat_id != CHAT_ID){
            bot.sendMessage(chat_id, "Usuario no autorizado", "");
            continue;
        }

        /* Presenta el mensaje recibido:
        De lo contrario, significa que el mensaje fue enviado por alguien de tu grupo,
        por lo que lo guardaremos en la variable de texto y verificaremos su contenido. */
        String text = bot.messages[i].text;
        Serial.println(text);

        // La variable remitente guarda el nombre del remitente.
        String remitente = bot.messages[i].from_name;
        Serial.println(remitente);
        /* Si se recibe el mensaje /led_encendido, enciende el LED y envía un mensaje confirmando
        que se ha recibido el mensaje. Además, actualiza la variable EstadoDeLed con el nuevo
        estado. */
        if (text == "/led_encendido") { // texto comando
            bot.sendMessage(chat_id, "El estado del LED es ENCENDIDO", "");
            EstadoDeLed = LOW; //lógica invertida
            digitalWrite(ledPin, EstadoDeLed);
        }
        /* Enviar un mensaje al bot es muy sencillo. Solo necesitas usar el método sendMessage ()
        en el objeto bot y pasar como argumentos el ID de chat del destinatario, el mensaje y
        el modo de análisis. */

        // Realiza algo similar para el mensaje "/led_apagado"
        if (text == "/led_apagado") { // texto comando
            bot.sendMessage(chat_id, "El estado del LED es APAGADO", "");
            EstadoDeLed = HIGH; //lógica invertida
            digitalWrite(ledPin, EstadoDeLed);
        }
    }
}
/*
Finalmente, si el mensaje recibido es /estado, verifica el estado actual de GPIO

```

```

y envía un mensaje en consecuencia. */
if (text == "/estado") { // texto comando
  if (digitalRead(ledPin)){
    bot.sendMessage(chat_id, "LED está APAGADO", "");

  }
  else{
    bot.sendMessage(chat_id, "LED está ENCENDIDO", "");
  }
}
}
}
}

void setup() {
  // inicializa la velocidad de comunicación serial
  Serial.begin(115200);
  // si estás usando ESP8266:
  #ifdef ESP8266
    configTime(0, 0, "pool.ntp.org"); // obtiene la hora UTC via servidor NTP
    /*Network Time Protocol (NTP) es un protocolo de Internet para sincronizar los relojes
    de los sistemas informáticos a través del enrutamiento de paquetes en redes con latencia
    variable. */
    client.setTrustAnchors(&cert); // Agrega el certificado raíz para api.telegram.org
  #endif

  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, EstadoDeLed);

  // Conectar a el entorno Wi-Fi de Wemos d1 r1
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  #ifdef ESP32

    client.setCACert(TELEGRAM_CERTIFICATE_ROOT); // Agrega el certificado raíz para
    //api.telegram.org
  #endif
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);

    Serial.println("Conectando al entorno WiFi..");
  }
  // Presenta la dirección IP local de la placa electrónica en uso
  Serial.println(WiFi.localIP());

  bot.sendMessage(CHAT_ID, "Bot Iniciado", "");
}

void loop() { //

```

```

if (millis() > UltimoArranque + RetardoSolicitudBot) {
  int numNuevosMensajes = bot.getUpdates(bot.last_message_received + 1);

  while(numNuevosMensajes) {
    Serial.println("se obtuvo respuesta");
    manejarNuevosMensajes(numNuevosMensajes);
    numNuevosMensajes = bot.getUpdates(bot.last_message_received + 1);
  }
  UltimoArranque = millis(); /* la función millis() permite obtener la cantidad de milisegundos
que han pasado desde que comenzó la ejecución del código. */
}
}

```

ACHIVO platformio.ini

```

[env:d1]
platform = espressif8266
board = d1
framework = arduino
monitor_speed = 115200
lib_deps =
    bblanchon/ArduinoJson@^6.18.5
    witnessmenow/UniversalTelegramBot@^1.3.0

```

CONEXIONES Y PRUEBAS:

Compila el código estudiado en el desarrollo, grábalo, implementa y comprueba.

Cuando todo esté preparado, en Telegram, mediante tu bot personal (por ejemplo Oswi_bot) aparecerá en tu teléfono celular el mensaje 'Bot iniciado' y a continuación la tarjeta Wi Fi Wemos D1 R1 a continuación puedes enviarle los siguientes comandos:

```

/led_encendido
/led_apagado

```

Verifica que cada miembro del grupo puede acceder de manera remota a tu tarjeta Wemos D1 R1 y controlar el encendido del LED construido en la tarjeta Wemos D1 R1.

Si dos placas Wemos usan el mismo token e ID_Chat, ambas responderán de manera similar aún estando ubicadas de manera remota una de otra.

Cada mensaje se enviará a cada miembro del grupo aunque no se tenga abierto Telegram.