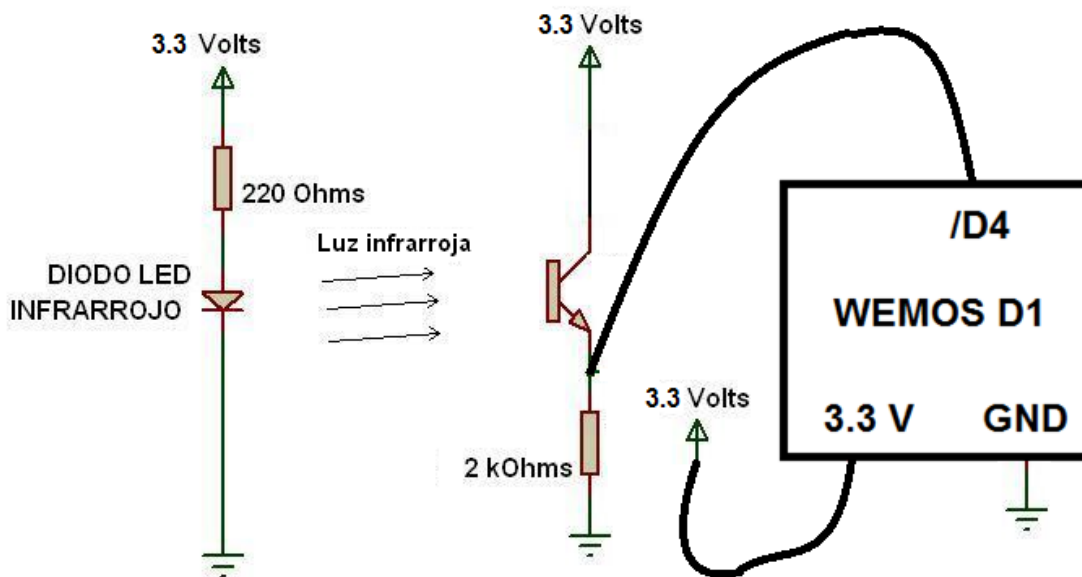


DETECCIÓN DE CUERPOS OPACOS ENVIANDO MENSAJE EN TELEGRAM



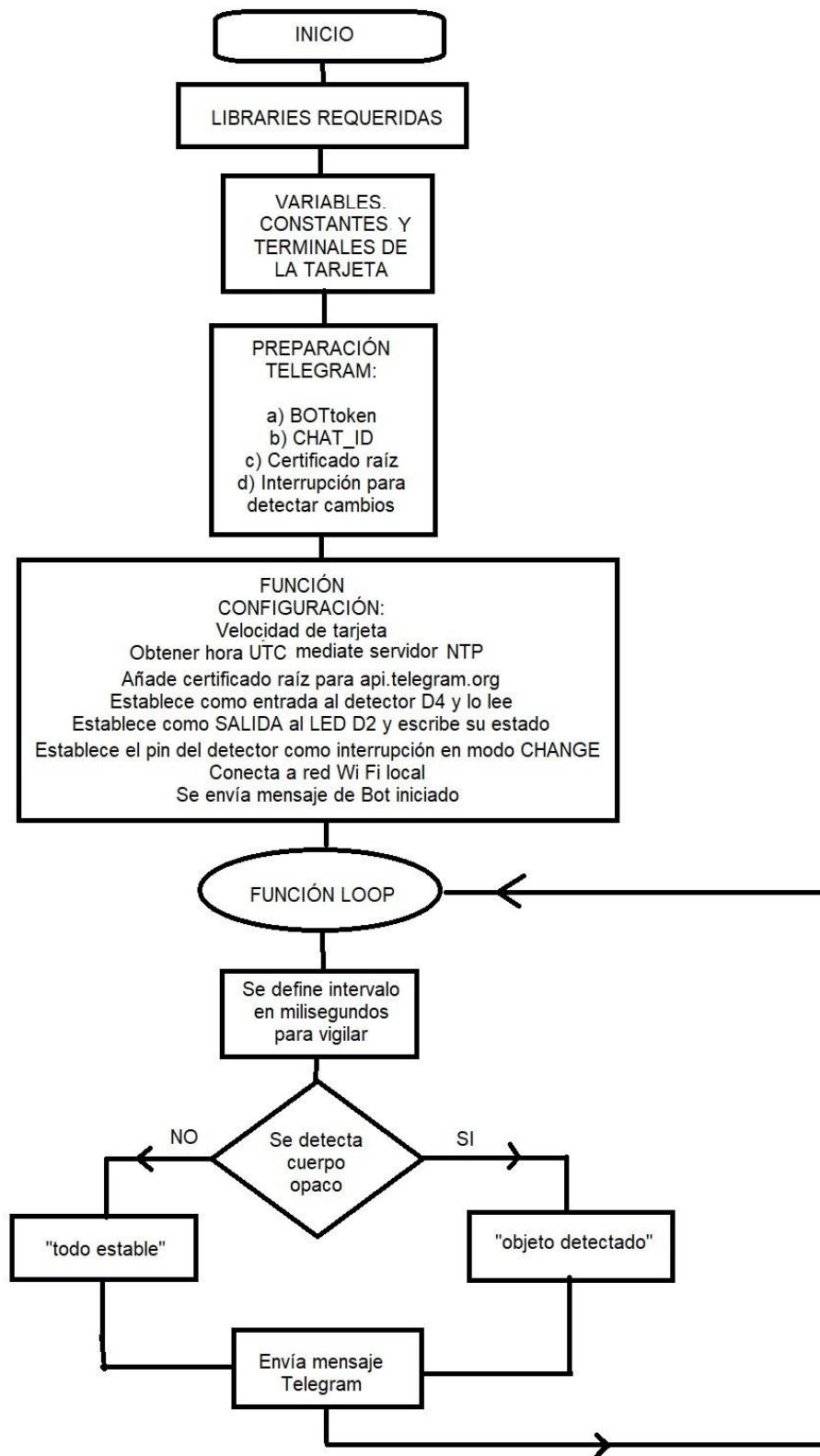
LISTA DE MATERIALES:

- 1 Wemos D1 R1.
- 1 Teléfono celular con Telegram.
- 1 Fotodiodo infrarrojo.
- 1 Fotoreceptor infrarrojo.
- 1 Resistor de 2.2 KOhms (RO-RO-RO)
- 1 Resistor de 220 Ohms (RO-RO-CA).
- Alambres dupont o alambres para protoboard.
- 1 Protoboard



NOTA IMPORTANTE: La tierra del circuito y la tierra (GND) de Wemos D1 deben estar unidas.

Analiza el diagrama de flujo que muestre el proceso detectar un cambio y enviar un mensaje en Telegram.



ARCHIVO main.cpp

```
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>
#include <ArduinoJson.h>

// establece los GPIOs para el LED y para el detector
const int detector = 4;
const int led = 2; //opcional
// para detectar cuando hay un cambio:
bool cambiaEstado = false;

// maneja el estado del detector (1=todo normal, 0=cuerpo detectado)
bool estado;
String estadodelLugar;

// variables auxiliares (detectarán cambios solo después que transcurran 1500 milisegundos )
unsigned long previoMillis = 0; // variable para almacenar tiempo
const long intervalo = 1500; // 1500 ms

const char* ssid = "MEGACABLE-TR23A";
const char* password = "w275RsJMd";

// Inicializar Telegram BOT
#define BOTtoken "2128271332:AAHm2CaCrdPrI21gg5nj5G7wTiaambPN5_Y" // tu Bot Token
// (se obtiene de Botfather)

// Usa @myidbot para encontrar el ID de chat para un contacto o grupo
// También necesitas pulsar en "inicio" en un bot para que pueda enviarte el mensaje
#define CHAT_ID "1467234516"

X509List cert(TELEGRAM_CERTIFICATE_ROOT); // Certificado raíz de Telgram
WiFiClientSecure client; // se crea el objeto client
UniversalTelegramBot bot(BOTtoken, client);

// Se ejecuta siempre que el detector cambia de estado
ICACHE_RAM_ATTR void cambiaEstadodelLugar() {
    Serial.println("Estado cambiado");
    cambiaEstado = true;
}

void setup() {
    // Velocidad del puerto serie de Wemos D1 R1:
    Serial.begin(115200);
    configTime(0, 0, "pool.ntp.org"); // obtiene la hora UTC mediante servidor NTP
    client.setTrustAnchors(&cert); // añade un certificado raíz para api.telegram.org
```

```

// Lee el estado actual del detector
pinMode(detector, INPUT_PULLUP);
estado = digitalRead(detector);

// Establece el estado del LED de acuerdo al detector de estado
pinMode(led, OUTPUT);
digitalWrite(led, estado);

// Establece el pin del detector as interrupción, asigna la función de interrupción y establece
// el modo CHANGE
attachInterrupt(digitalPinToInterrupt(detector), cambiaEstadodeLugar, CHANGE);

// Conecta a Wi-Fi
WiFi.mode(WIFI_STA);
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
Serial.println("");
Serial.println("WiFi conectado ");

bot.sendMessage(CHAT_ID, "Bot iniciado", ""); // método del bot para enviar mensajes
}

void loop() {
    if (cambiaEstado){
        unsigned long actualMillis = millis();
        if(actualMillis - previoMillis >= intervalo) {
            previoMillis = actualMillis;
            // si se ha interpuesto un cuerpo opaco, definir el estado del lugar
            estado = !estado;
            if(estado) {
                estadodelLugar = "todo estable";
            }
            else{
                estadodelLugar = "objeto detectado";
            }
            digitalWrite(led, estado);
            cambiaEstado = false;
            Serial.println(estado);
            Serial.println(estadodelLugar);

            //envía notificación a telegram:
            bot.sendMessage(CHAT_ID, "La situación es: " + estadodelLugar, "");
        }
    }
}

```

ACHIVO platformio.ini

```
[env:d1]
platform = espressif8266
board = d1
framework = arduino
monitor_speed = 115200
lib_deps =
    bblanchon/ArduinoJson@^6.19.3
    me-no-dev/ESPAsyncTCP@^1.2.2
    witnessmenow/UniversalTelegramBot@^1.3.0
```

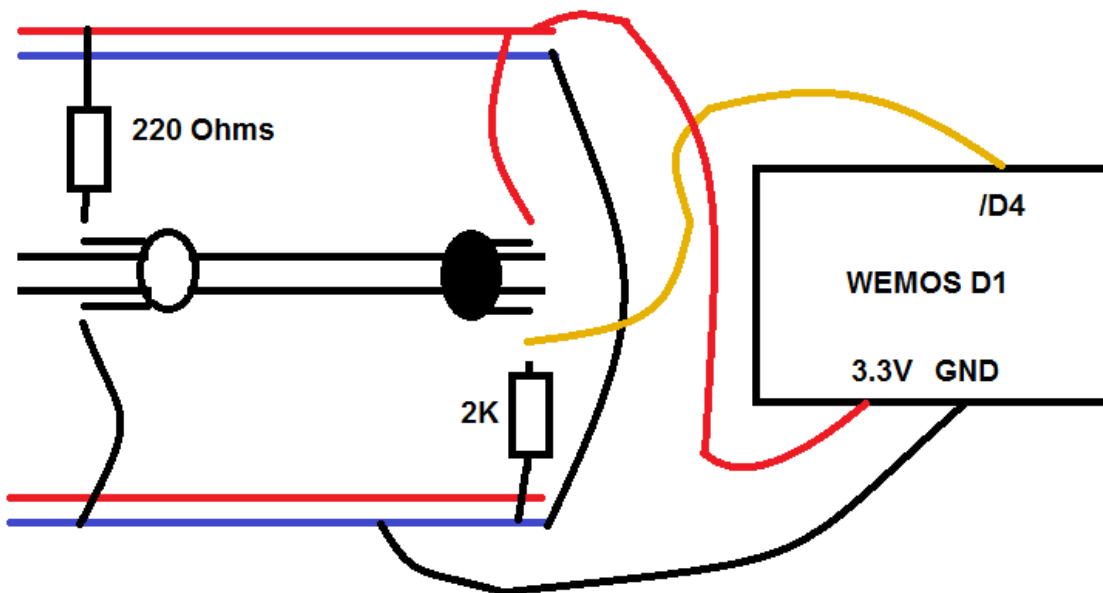
Compila el código estudiado en el desarrollo, grábalo, implementa y comprueba.

CONEXIONES Y PRUEBAS: Conecta la salida del detector de cuerpos opacos a la entrada GPIO4 de Wemos D1 R1 y presenta obstáculos de cuerpos opacos para verificar los mensajes recibidos en Telegram.

El fotoemisor y el fotoreceptor deben acomodarse con la inclinación para poder emitir y recibir la señal de luz infrarroja, tal como se muestra en la siguiente figura:



NOTA IMPORTANTE: En el fotoemisor, la marca plana en el redondel del mismo indica la terminal NEGATIVA, mientras que en el fotoreceptor, la marca plana en el redondel indica la terminal POSITIVA. Al armar el circuito en el protoboard queda de la siguiente manera:



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 enviará uno de los siguientes mensajes "La situación es: todo estable" (cuando en D4 se reciba un UNO), o bien "La situación es: objeto detectado" (cuando en D4 se reciba un CERO). Dependiendo de los valores que la tarjeta reciba, provenientes de la salida del fotoreceptor. De esta manera la tarjeta Wi Fi te comunicará si ocurre un cambio mediante mensaje de Telegram.

El mensaje se enviará aunque no se tenga abierto Telegram.