#define BLYNK\_TEMPLATE\_ID "TMPL3rLFxSrDX"

#define BLYNK\_TEMPLATE\_NAME "Street Light Fault Detection"

#define BLYNK\_AUTH\_TOKEN "GIFwLJBa4ed14uffkMHGE066JcuiwsRX"

#define BLYNK\_PRINT Serial

#include <WiFi.h>

#include <BlynkSimpleEsp32.h>

#include <TinyGPSPlus.h>

#include <SPI.h>

#include <MFRC522.h>

#include <HTTPClient.h> // Include the HTTPClient library to send SMS via a web service

char auth[] = BLYNK\_AUTH\_TOKEN;

char ssid[] = "realme"; // Your WiFi SSID

char pass[] = "12345678"; // Your WiFi password

#define ANALOG\_IN\_PIN 32

#define LDR\_Sensor 34

#define IR\_Sensor 14

#define RELAY\_PIN 5

#define SS\_PIN 21 // Slave Select Pin

#define RST\_PIN 22 // Reset Pin for RC522

int a;

MFRC522 mfrc522(SS\_PIN, RST\_PIN); // Create MFRC522 instance

float adc\_voltage = 0.0;

float in\_voltage = 0.0;

float R1 = 250.0;

float R2 = 250.0;

float ref\_voltage = 2.0;

int adc\_value = 0;

TinyGPSPlus gps;

// Twilio API credentials

const String TWILIO\_ACCOUNT\_SID = "ACeea9b193a1253eb91962897233fabdc7";

const String TWILIO\_AUTH\_TOKEN = " 899da176cf9333c42d1a5b88f4da2f17"; // Replace with your actual Twilio Auth Token

const String TWILIO\_FROM\_PHONE = "+12513020745"; // Your Twilio phone number

const String TO\_PHONE = "+919834932740"; // The phone number to send SMS to

BLYNK\_WRITE(V4) {

// Control relay from Blynk app

digitalWrite(RELAY\_PIN, param.asInt());

}

void setup() {

Serial.begin(9600);

Serial2.begin(9600);

delay(3000);

Blynk.begin(auth, ssid, pass);

SPI.begin(); // Initialize SPI communication

mfrc522.PCD\_Init(); // Initialize RFID sensor

pinMode(RELAY\_PIN, OUTPUT);

digitalWrite(RELAY\_PIN, HIGH);

pinMode(LDR\_Sensor, INPUT);

pinMode(IR\_Sensor, INPUT);

a = 1;

Blynk.virtualWrite(V3, ".....");

}

void loop() {

Blynk.run();

adc\_value = analogRead(ANALOG\_IN\_PIN);

adc\_voltage = (adc\_value \* ref\_voltage) / 1024.0;

in\_voltage = adc\_voltage / (R2 / (R1 + R2));

Serial.print("Input Voltage = ");

Serial.println(in\_voltage, 2);

int value1 = analogRead(LDR\_Sensor);

Serial.println(value1);

int value2 = digitalRead(IR\_Sensor);

Serial.println(value2);

while (Serial2.available() > 0) {

if (gps.encode(Serial2.read())) {

displayInfo();

}

}

if (millis() > 5000 && gps.charsProcessed() < 10) {

Serial.println(F("No GPS detected: check wiring."));

while (true);

} else {

// Other processing

}

if (value2 == 0 && value1 > 3500) {

digitalWrite(RELAY\_PIN, LOW);

delay(5000);

digitalWrite(RELAY\_PIN, HIGH);

}

if (in\_voltage < 5 && value1 > 3500) {

Serial.print("Lat: ");

Serial.print(gps.location.lat(), 6);

Serial.print(" Lng: ");

Serial.print(gps.location.lng(), 6);

Serial.println();

Blynk.virtualWrite(V0, "Wired Fault Detected");

displayInfo();

Blynk.virtualWrite(V1, gps.location.lat());

Blynk.virtualWrite(V2, gps.location.lng());

if (in\_voltage == 0) {

sendSms(); // Send SMS when the voltage is zero

}

} else {

Blynk.virtualWrite(V0, "NO Fault Detected");

}

// Look for the RFID Card

if (!mfrc522.PICC\_IsNewCardPresent()) {

return;

}

// Select Card

if (!mfrc522.PICC\_ReadCardSerial()) {

return;

}

// Show UID for Card/Tag on serial monitor

Serial.print("UID tag :");

String content = "";

byte letter;

for (byte i = 0; i < mfrc522.uid.size; i++) {

Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");

Serial.print(mfrc522.uid.uidByte[i], HEX);

content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));

content.concat(String(mfrc522.uid.uidByte[i], HEX));

}

Serial.println();

Serial.print("Message : ");

content.toUpperCase();

if (content.substring(1) == "04 A6 1A 9A") { // UID for the Card/Tag to give access

Serial.println("Authorized access"); // Print message if UID matches the database

Serial.println();

delay(500);

Blynk.virtualWrite(V3, "Wireman Visited");

a = 0;

content = "";

} else {

Blynk.virtualWrite(V3, "-");

}

}

void displayInfo() {

char locationStr[20];

if (gps.location.isValid()) {

sprintf(locationStr, "%.6f,%.6f", gps.location.lat(), gps.location.lng());

} else {

sprintf(locationStr, "NO\_GPS");

}

}

void sendSms() {

if ((WiFi.status() == WL\_CONNECTED)) { // Check the current connection status

HTTPClient http;

String messageBody = "Hello! Wire cut fault has been occurred on location: https://maps.app.goo.gl/c49BLxUnyHFVQBUZ7 please visit";

String url = "https://api.twilio.com/2010-04-01/Accounts/" + TWILIO\_ACCOUNT\_SID + "/Messages.json";

http.begin(url); // Specify the URL

http.setAuthorization(TWILIO\_ACCOUNT\_SID.c\_str(), TWILIO\_AUTH\_TOKEN.c\_str());

http.addHeader("Content-Type", "application/x-www-form-urlencoded");

String requestBody = "From=" + TWILIO\_FROM\_PHONE + "&To=" + TO\_PHONE + "&Body=" + messageBody;

int httpResponseCode = http.POST(requestBody);

if (httpResponseCode > 0) {

String response = http.getString(); // Get the response to the request

Serial.println(httpResponseCode); // Print return code

Serial.println(response); // Print request answer

} else {

Serial.print("Error on sending POST: ");

Serial.println(httpResponseCode);

}

http.end(); // Free resources

} else {

Serial.println("Error in WiFi connection");

}

}