#include <SPI.h>

#include <MFRC522.h>

#include <SoftwareSerial.h>

// RFID module pins

#define SS\_PIN 10 // Slave Select (SS) pin for RFID module

#define RST\_PIN 9 // Reset pin for RFID module

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

// LED pins

#define RED\_LED\_NORTH 2

#define YELLOW\_LED\_NORTH 3

#define GREEN\_LED\_NORTH 4

#define RED\_LED\_SOUTH 5

#define YELLOW\_LED\_SOUTH A0

#define GREEN\_LED\_SOUTH A1

#define RED\_LED\_EAST 8

#define YELLOW\_LED\_EAST 11

#define GREEN\_LED\_EAST 12

#define RED\_LED\_WEST A4

#define YELLOW\_LED\_WEST A2

#define GREEN\_LED\_WEST A3

// GSM module pins

#define GSM\_TX\_PIN 6 // Define the GSM module TX pin

#define GSM\_RX\_PIN 7 // Define the GSM module RX pin

SoftwareSerial gsmSerial(GSM\_TX\_PIN, GSM\_RX\_PIN); // Create a SoftwareSerial object for GSM communication

void setup() {

Serial.begin(9600);

gsmSerial.begin(9600); // Start GSM module communication

SPI.begin(); // Initiate SPI bus

mfrc522.PCD\_Init(); // Initiate MFRC522

// Set LED pins as OUTPUT

pinMode(RED\_LED\_NORTH, OUTPUT);

pinMode(YELLOW\_LED\_NORTH, OUTPUT);

pinMode(GREEN\_LED\_NORTH, OUTPUT);

pinMode(RED\_LED\_SOUTH, OUTPUT);

pinMode(YELLOW\_LED\_SOUTH, OUTPUT);

pinMode(GREEN\_LED\_SOUTH, OUTPUT);

pinMode(RED\_LED\_EAST, OUTPUT);

pinMode(YELLOW\_LED\_EAST, OUTPUT);

pinMode(GREEN\_LED\_EAST, OUTPUT);

pinMode(RED\_LED\_WEST, OUTPUT);

pinMode(YELLOW\_LED\_WEST, OUTPUT);

pinMode(GREEN\_LED\_WEST, OUTPUT);

// Initialize traffic signals to red for all directions

setTrafficLightsRed();

// Test communication with the GSM module

gsmSerial.println("AT");

delay(1000);

printResponse();

// Set SMS mode to text

gsmSerial.println("AT+CMGF=1");

delay(1000);

printResponse();

Serial.println("System is ready.");

}

void loop() {

// Reset traffic signals to red by default

setTrafficLightsRed();

// Look for new RFID cards

if (mfrc522.PICC\_IsNewCardPresent() && mfrc522.PICC\_ReadCardSerial()) {

// A card is present; you can add more logic here for card-specific actions

// For now, change the traffic signal to green for the relevant direction

setTrafficLightsGreen();

// Send SMS indicating an ambulance is coming

sendSMS("Ambulance Alert: An ambulance is approaching the intersection.");

// Print the card's UID

Serial.print("Card UID: ");

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

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

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

}

Serial.println();

delay(2000); // Keep the green signal for 2 seconds

}

// Halt the program for a moment to avoid excessive readings

delay(1000);

}

void setTrafficLightsRed() {

digitalWrite(RED\_LED\_NORTH, HIGH);

digitalWrite(YELLOW\_LED\_NORTH, LOW);

digitalWrite(GREEN\_LED\_NORTH, LOW);

digitalWrite(RED\_LED\_SOUTH, HIGH);

digitalWrite(YELLOW\_LED\_SOUTH, LOW);

digitalWrite(GREEN\_LED\_SOUTH, LOW);

digitalWrite(RED\_LED\_EAST, HIGH);

digitalWrite(YELLOW\_LED\_EAST, LOW);

digitalWrite(GREEN\_LED\_EAST, LOW);

digitalWrite(RED\_LED\_WEST, HIGH);

digitalWrite(YELLOW\_LED\_WEST, LOW);

digitalWrite(GREEN\_LED\_WEST, LOW);

}

void setTrafficLightsGreen() {

// Determine the direction based on the card UID

// For example, you can use specific bits of the UID to determine the direction

// Then, set the corresponding traffic lights to green

// For now, let's assume the direction is North

digitalWrite(RED\_LED\_NORTH, LOW);

digitalWrite(YELLOW\_LED\_NORTH, LOW);

digitalWrite(GREEN\_LED\_NORTH, HIGH);

}

void sendSMS(String message) {

String phoneNumber = "+9347945022"; // Replace with your phone number

gsmSerial.print("AT+CMGS=\"");

gsmSerial.print(phoneNumber);

gsmSerial.println("\"");

delay(1000);

printResponse();

gsmSerial.println(message);

delay(1000);

gsmSerial.write(26); // Send Ctrl+Z (ASCII 26) to send the SMS

delay(1000);

printResponse();

Serial.println("Message sent to " + phoneNumber);

}

void printResponse() {

while (gsmSerial.available()) {

Serial.write(gsmSerial.read());

}

}