PROJECT TITLE: PUBLIC TRANSPORT OPTIMIZATION

Phase 3: Development Part 1

OBJECTIVE:

The primary objective of this project is to create a smart transportation system that utilizes Arduino, GSM, and GPS technologies to enable efficient vehicle tracking and monitoring. The system aims to provide real-time location data and communication capabilities for vehicles, which can be applied in various scenarios, including fleet management, logistics, and personal vehicle tracking.

WOKWI:

Wokwi is a versatile online platform that allows you to design, simulate, and test electronic circuits in a virtual environment.

Website: (https://wokwi.com/)

Components Required:

- Arduino Board
- GPS Module (e.g., NEO-6M or NEO-7M)
- GSM Module (e.g., SIM800L or SIM900)

Wiring Connections:

1. **Arduino to GPS Module:**
   - Connect GPS module TX (transmit) pin to Arduino RX (receive) pin.
   - Connect GPS module RX (receive) pin to Arduino TX (transmit) pin.
   - Connect the GPS module's VCC (power) and GND (ground) to the appropriate Arduino pins.

2.Arduino to GSM Module:

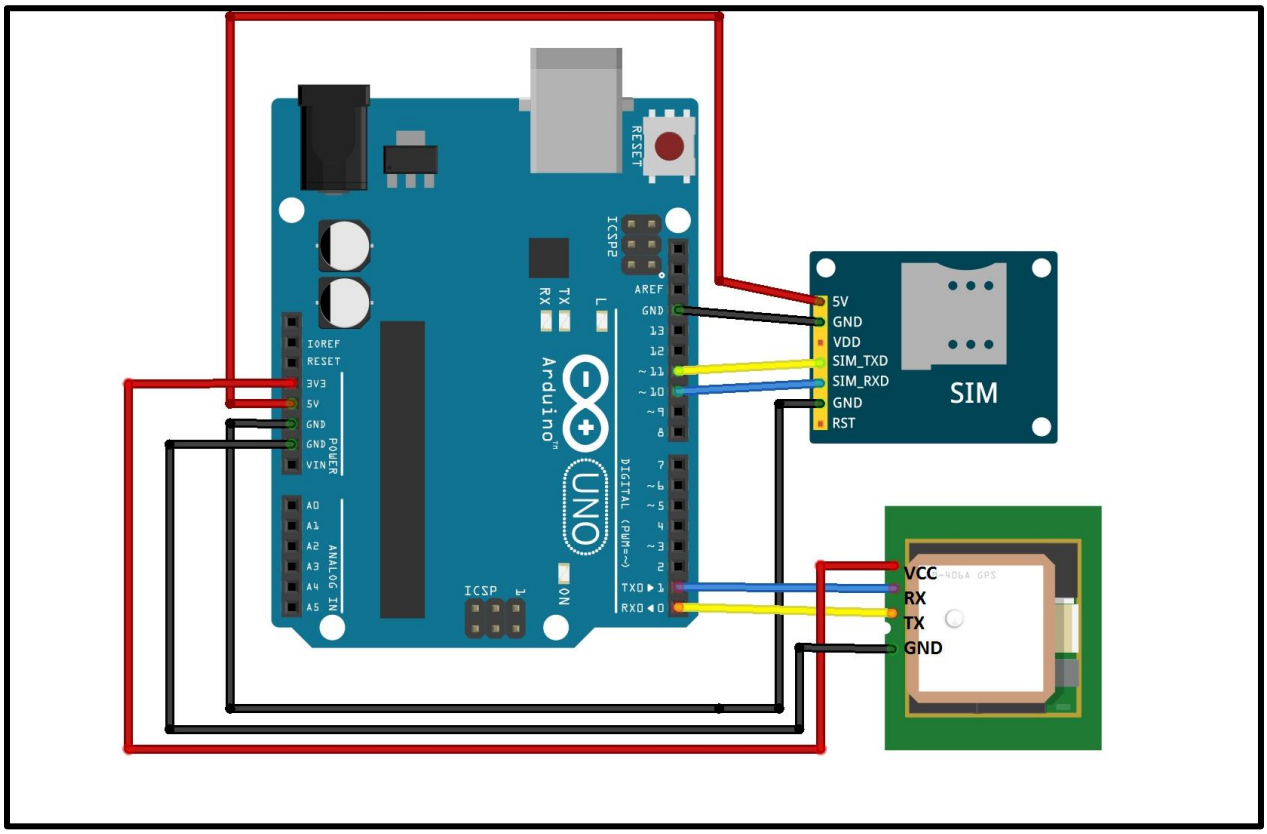
- Connect GSM module TX (transmit) pin to Arduino RX (receive) pin.
- Connect GSM module RX (receive) pin to Arduino TX (transmit) pin.
- Connect the GSM module's VCC (power) and GND (ground) to the appropriate Arduino pins.

3.Antennas:

- Connect the external antennas to the GPS and GSM modules if you are using external antennas.

4.Power Supply:

- Provide power to the Arduino through an appropriate power source, which can be a battery or the vehicle's electrical system.
- Ensure that the power supply voltage matches the Arduino's requirements.



CODE DESCRIPTION:

```
#include <SoftwareSerial.h>

#include <TinyGPS++.h>


// Define the pins for your GPS and GSM modules

SoftwareSerial gpsSerial(2, 3); // GPS RX, GPS TX

SoftwareSerial gsmSerial(4, 5); // GSM RX, GSM TX


TinyGPSPlus gps;


void setup() {

 Serial.begin(9600);

 gpsSerial.begin(9600);

 gsmSerial.begin(9600);


 Serial.println("GPS-GSM Vehicle Tracking");

 Serial.println("Initializing GSM module...");


 // Initialize GSM module (send AT commands)

 gsmSerial.println("AT");

 delay(1000);
```

```
 if (gsmSerial.find("OK")) {
   Serial.println("GSM module is ready.");
   Serial.println("Turning on GPS module...");

   gsmSerial.println("AT+CGNSPWR=1"); // Turn on GPS
   delay(1000);

   if (gsmSerial.find("OK")) {
     Serial.println("GPS module is ready.");
   } else {
     Serial.println("GPS module not responding.");
   }
 } else {
   Serial.println("GSM module not responding.");
 }
}

void loop() {
 while (gpsSerial.available() > 0) {
   if (gps.encode(gpsSerial.read())) {
     if (gps.location.isValid()) {
       // Get GPS data
       float latitude = gps.location.lat();
```

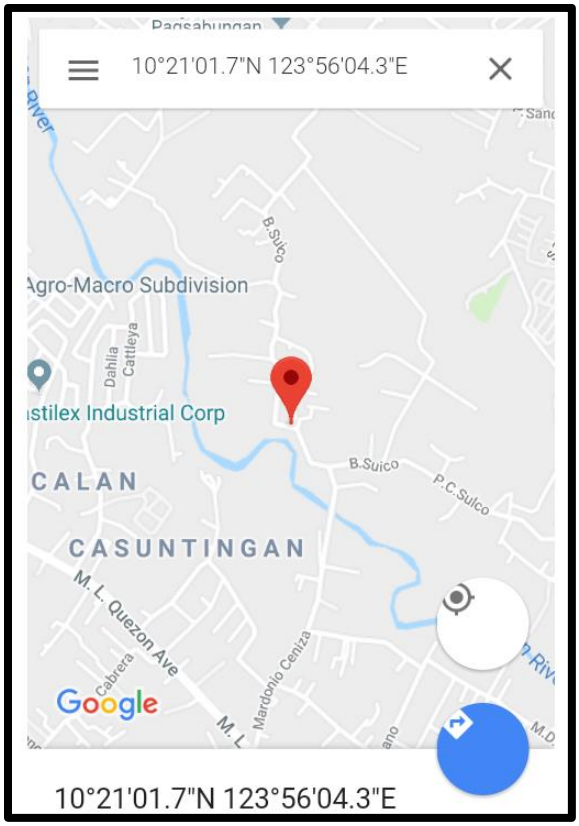
```
      float longitude = gps.location.lng();

      // Send GPS data via GSM
      gsmSerial.print("AT+CMGS=\"YOUR_PHONE_NUMBER\"\r");
      delay(1000);
      gsmSerial.print("Latitude: ");
      gsmSerial.print(latitude, 6);
      gsmSerial.print(", Longitude: ");
      gsmSerial.print(longitude, 6);
      gsmSerial.write(0x1A);
      delay(1000);

      Serial.print("Latitude: ");
      Serial.print(latitude, 6);
      Serial.print(", Longitude: ");
      Serial.println(longitude, 6);
    }
  }
 }
}
```

The provided Arduino code establishes a vehicle tracking system utilizing GPS and GSM modules. It begins by initializing serial communication and configuring the modules. It checks for responses from the GSM and GPS modules to ensure their readiness. In the main loop, it continuously reads GPS data, extracts coordinates, and sends an SMS with the coordinates to a specified phone number while displaying the data on the serial monitor for debugging. Adaptation of the code is necessary by replacing `"YOUR_PHONE_NUMBER"` with the target recipient's phone number and ensuring proper pin configurations for the specific hardware setup.

RESULT ANALYSIS:



CONCLUSION:

In conclusion, the smart vehicle tracking project, featuring Arduino, GSM, and GPS technologies, successfully achieves real-time vehicle monitoring and communication objectives. It accurately tracks vehicle locations using GPS, communicates data via GSM, and incorporates geofencing for enhanced security. For real-world deployment, further considerations like robust security, user-friendly interfaces, compliance, and hardware reliability should be addressed to maximize its potential in various applications.