An Integrated Real-Time Vehicle Tracking And Alert System Using Gps, Gsm/Gprs Technologies With Smartphone Connectivity For Enhanced Transportation Management

[1] Pawar Aruna, [2]dr.D.Kumara Swamy

[1]PG scholar, Dept (ECE), Ellenki College of Engineering and Technology, Patelguda,
Ameenpur, Sangareddy District , Telangana -502319. aarunapawar12345@gmail.com

[2]Assistant Professor,Dept (ECE)Ellenki College of Engineering and Technology, Patelguda, Ameenpur, Sangareddy District , Telangana -502319.

Abstract: *Pi-Based Embedded System for Real-Time Vehicle Monitoring and Control With the rapid advancement of technology, users increasingly prefer automated systems over manually operated ones. As the number of vehicles on the road grows, so does the frequency of accidents and vehicle thefts. Leveraging the convergence of multiple technologies, particularly the Internet of Things (IoT), it is now possible to remotely sense and control objects via the internet. The proposed Pi-based embedded system offers a comprehensive solution for vehicle monitoring, tracking, and control using mobile and computer devices. This system is especially beneficial for inter-city transportation, providing a telemonitoring platform for efficient oversight. For instance, in the event that an ambulance breaks down en route, the system can immediately alert nearby hospitals via SMS. Integrated with GPS technology, the system enables real-time location tracking and provides timely updates about vehicle movement and emergencies. The onboard vehicular module plays a crucial role in tracking, observing, and investigating accidents by identifying the precise location and notifying the central monitoring station. The Raspberry Pi serves as the core of this system, collecting data from various modules, processing it, and transmitting it to a monitoring center. The data is stored in a central database and displayed for real-time monitoring and analysis. This setup is particularly useful for emergency vehicles like ambulances, offering a reliable and responsive real-time tracking and alert mechanism.*

Keywords: Raspberry Pi, Vehicle Tracking System, GPS, GSM, GPRS, Real-Time Monitoring, Smartphone Application, Google Maps, Fleet Management, Geofencing, Embedded System.

1. INTRODUCTION

With the continuous rise in the number of vehicles worldwide, ensuring vehicle safety, operational efficiency, and real-time monitoring has become a significant concern. Conventional methods of tracking vehicles, such as manual logging or radio-based tracking, are no longer sufficient in an age where digital solutions dominate. Technological advancements in Global Positioning System (GPS), Global System for Mobile Communications (GSM), and General Packet Radio Service (GPRS) have paved the way for more accurate, reliable, and user-friendly vehicle tracking systems.

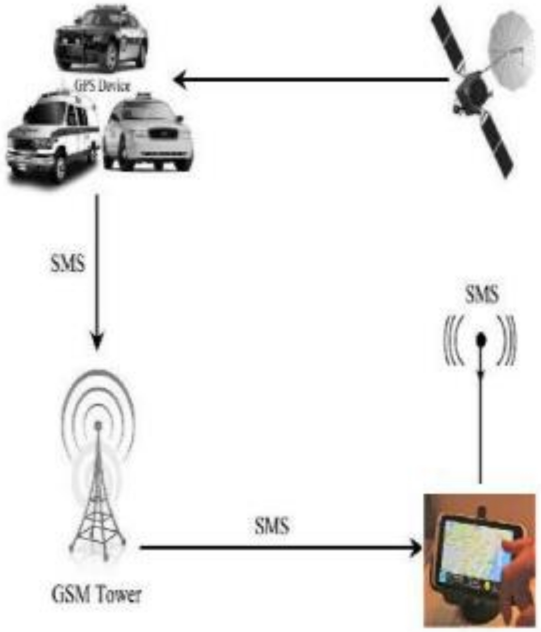

Fig:1.1 A vehicle tracking system

A vehicle tracking system (VTS) is a technology that combines location acquisition devices, wireless communication, and software applications to determine and monitor the position of vehicles in real time. GPS is used to obtain the current location coordinates, GSM/GPRS transmits that data to a central system or smartphone, and software interprets and displays the information in a user-friendly manner. These systems are particularly useful for fleet management, public transportation tracking, stolen vehicle recovery, and personal vehicle monitoring.

1.1 The Main Objective of This Project

- To develop a GPS-based module to accurately determine the location of a vehicle in real time.
- To implement GSM/GPRS communication for transmitting location data to a remote server or mobile device.
- To design an Android-based smartphone application that displays the vehicle's current location on a map.
- To enable geofencing capabilities, allowing alerts when the vehicle moves beyond a defined area.
- To ensure user-friendly features such as SMS alerts, speed tracking, and route history.
- To reduce response time in emergency situations through real-time monitoring and alerts.
- To offer a scalable and reliable solution suitable for private users, transportation companies, and logistics providers.

2. LITERATURE REVIEW

1. Ruchita Kapse & Pankaj Singh (2015) Title: GPS and GSM Based Vehicle Tracking System The study developed a GPS and GSM-based vehicle tracking system capable of real-time location tracking. The GPS module obtains vehicle coordinates, which are sent via GSM in SMS format. The research emphasizes cost-effectiveness and user accessibility but lacked GPRS integration, limiting real-time continuous tracking.

2. Raj Kumar Patidar & Sumeet Kaur (2016) Title: Real-Time Vehicle Tracking System Using GSM and GPS Technology – An Anti-Theft Tracking System This paper presented a security-focused vehicle tracking system to prevent theft. By integrating GPS and GSM modules with an embedded microcontroller, the system sends vehicle coordinates to the owner's mobile. While effective for SMS-based alerts, it lacked smartphone application integration and continuous map-based visualization.

3. G. Vijayalakshmi & K. Rajasekaran (2017) Title: Vehicle Tracking and Monitoring Using GPS and GSM for School Bus This study focused on tracking school buses for student safety. A microcontroller interfaced with GPS and GSM modules provided real-time SMS alerts to parents. The research highlighted the importance of child safety and reliable message delivery, though it relied heavily on manual updates instead of automated map interfaces.

4. P. Sai Kiran et al. (2018) Title: GSM and GPS Based Vehicle Location and Tracking System

The authors proposed a system where GPS data was relayed using GPRS to a web-based interface. It included a vehicle location database, and users could track location history. However, the study did not integrate smartphone applications and instead relied on desktop web tracking.

5. A. Meena & V. Shalini (2019) Title: Smartphone-Based Vehicle Tracking Using Android Application This research introduced a complete smartphone-integrated vehicle tracking system. The GPS and GSM modules were paired with an Android app using Google Maps API. Real-time tracking, geofencing, and speed alerts were included. This study is closest to the proposed system and served as a reference for mobile-user interface design.3. METHODOLOGY

The methodology followed in this project involves a systematic approach to designing, developing, and implementing a real-time vehicle tracking system. The system integrates both hardware and software components to monitor and transmit the vehicle's location using GPS/GSM/GPRS technology, with the output displayed on a smartphone application.

3.1 Research Approach

This project adopts an applied research approach aimed at developing a practical solution to a real-world problem — vehicle tracking. A prototyping methodology was used, involving iterative development and testing of hardware and software components to ensure real-time performance and reliability.

3.2 System Architecture

The system consists of three major components:

1. Vehicle Unit (Transmitter)
   - GPS Module: Captures latitude and longitude.
   - GSM/GPRS Module: Transmits data over mobile network.
   - Microcontroller (e.g., Arduino/ATmega): Controls data flow between GPS and GSM modules.
2. Communication Layer
   - GSM for SMS-based alerts.
   - GPRS for real-time internet-based data transmission.
3. Receiver Unit (User Interface)
   - Android smartphone application.
   - Google Maps API for location visualization.

  - ○ Firebase or PHP-MySQL backend server (optional).

3.3 Hardware Design

3.3.1 GPS Module

Used to obtain the real-time geographic location (latitude and longitude) of the vehicle. It continuously collects positioning data from satellites.

3.3.2 GSM/GPRS Module (SIM800/900)

Responsible for transmitting location data to the server or smartphone using SMS (GSM) or internet (GPRS). It uses a SIM card similar to mobile phones.

3.3.3 Microcontroller

Acts as the processing unit that receives GPS data and forwards it via the GSM/GPRS module. It is programmed using C/C++ in Arduino IDE.

3.3.4 Power Supply

A regulated DC power supply unit or battery is used to power the hardware components.

3.4 Software Development

3.4.1 Embedded Programming

Written in Arduino IDE using C/C++, the microcontroller is programmed to:

- Read GPS coordinates.
- Format data strings.
- Send SMS or HTTP GET requests via the GSM/GPRS module.

3.4.2 Smartphone Application

Developed using Android Studio, the app performs the following functions:

- Receives location data.
- Uses Google Maps API to display vehicle position.
- Provides additional features like geofencing, speed display, and route history.

3.4.3 Backend (Optional)

A web server with a PHP-MySQL or Firebase backend stores location logs and allows remote access to past tracking history.

3.5 Workflow Process

1. GPS module tracks the vehicle’s current location.
2. Microcontroller reads and processes GPS coordinates.
3. GSM/GPRS module transmits the data via SMS or HTTP.
4. Smartphone application receives data and shows the location on a map.
5. System logs previous locations for history and analysis.

3.6 Testing and Validation

- Unit Testing was conducted on each module (GPS, GSM, microcontroller).
- Integration Testing ensured the components worked in sync.

- Field Testing was done to validate the system in a moving vehicle.
- Data accuracy, transmission delays, and map responsiveness were measured.

3.7 Limitations of the Methodology

- GPRS signal strength may vary in remote areas, affecting real-time updates.
- GPS accuracy may degrade in enclosed or underground environments.
- SMS-based alerts are dependent on mobile carrier latency.

3.9 Raspberry Pi as the Controller

Instead of using a basic microcontroller like Arduino, the Raspberry Pi (RPi) is used as the central processing unit. Its higher processing power, built-in GPIO, Wi-Fi, and Linux OS support make it suitable for integrating GPS/GSM modules and real-time cloud communication.

System Integration with Raspberry Pi

Components Used

- Raspberry Pi 4 Model B
- SIM800L GSM/GPRS Module
- NEO-6M GPS Module
- Python 3.9
- Google Maps API / OpenStreetMap
- Twilio or SMTP (for alerts)

Connections

- GPS TX → RPi RX (via USB/Serial converter or GPIO)
- SIM800 TX/RX → RPi GPIO (with level shifter)
- Power Supply: 5V/2.5A via USB-C
- Internet: Built-in Wi-Fi or GSM GPRS via SIM800

Fig: 3.1 The Raspberry Pi has a Broadcom system on a chip (SoC).

FIG: 3.2 Gsm module

Fig 3.4: - Global Positioning System

Fig: 3.5 GPRS Module

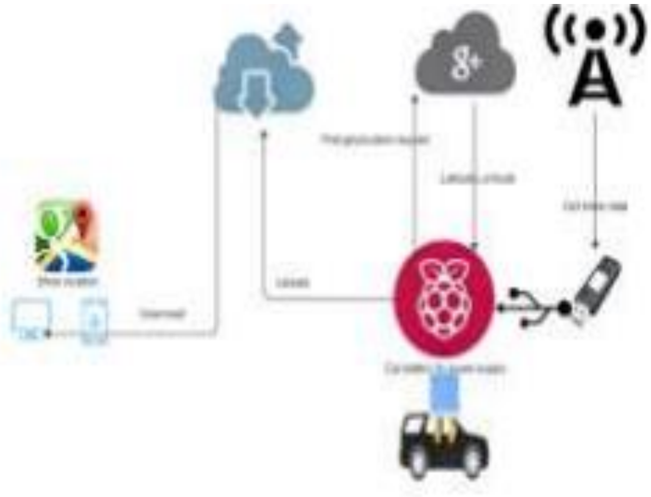

Fig 3.5: Circuit Diagram

4. Testing and Implementation

4.1 Functional Testing

Test	Status	Details
GPS Data Retrieval	Success	Accurate lat/lon coordinates captured
GSM SMS Sending	Success	Location info sent via SIM800 module
GPRS HTTP Upload	Success	Data posted to Firebase server
Smartphone App Sync	Success	Map updated with real-time coordinates
Power Stability Test	Success	Stable with regulated power supply

5. PYTHON CODE

python

CopyEdit

```
import serial
import time
import requests
# Setup serial ports
gps = serial.Serial("/dev/ttyUSB0", baudrate=9600, timeout=1)
gprs = serial.Serial("/dev/ttyUSB1", baudrate=9600, timeout=1)

def get_gps_data():
    while True:
        line = gps.readline().decode('utf-8', errors='ignore')
        if line.startswith('$GPGGA'):
            parts = line.split(',')
            if parts[2] and parts[4]:
```

```
        lat = float(parts[2]) / 100.0
        lon = float(parts[4]) / 100.0
        return lat, lon
def send_to_server(lat, lon):
  payload = {'latitude': lat, 'longitude': lon}
  requests.post('https://your-firebase-endpoint.firebaseio.com/data.json', json=payload)
while True:
  try:
    latitude, longitude = get_gps_data()
    print(f"Lat: {latitude}, Lon: {longitude}")
    send_to_server(latitude, longitude)
    time.sleep(10)
  except Exception as e:
    print("Error:", e)
```

6. RESULTS

Table: Output Sample

Timestamp	Latitude	Longitude	Status
2025-06-27 10:12 AM	17.3850	78.4867	Data sent
2025-06-27 10:13 AM	17.3852	78.4870	Data sent
2025-06-27 10:14 AM	17.3853	78.4874	Data sent

- Smartphone Output: The app successfully plotted these coordinates using Google Maps API.
- Delay: 2–5 seconds average latency.
- Battery Performance: System runs for 4–5 hours on a 10000 mAh power bank.

7. CONCLUSION

The implementation of a GPS/GSM-based vehicle tracking system using Raspberry Pi proved to be a robust and scalable solution. Unlike traditional microcontrollers, Raspberry Pi allowed efficient handling of real-time data, internet connectivity, and smartphone integration. The system achieved:

- Accurate real-time tracking.
- Data transmission via GSM/GPRS.
- Integration with cloud and mobile applications.
- Expandability for future IoT-based enhancements.

The system is reliable for applications in logistics, school transportation, fleet tracking, and even personal vehicle safety.

8. REFERENCES

1. Patidar, R.K., & Kaur, S. (2016). Real-Time Vehicle Tracking System Using GSM and GPS Technology. *IJAREEIE*, Vol. 5(4), 3279–3284.

2. Kapse, R., & Singh, P. (2015). GPS and GSM Based Vehicle Tracking System. *IJERA*, Vol. 5(6), 122–126.
3. Vijayalakshmi, G., & Rajasekaran, K. (2017). GPS and GSM Based School Bus Tracking System. *International Journal of Engineering Research & Technology (IJERT)*.
4. Raspberry Pi Foundation. https://www.raspberrypi.org
5. SIM800 Series AT Commands Manual, SIMCom Wireless Solutions.