

Bindura University of Science Education

Faculty of Science and Physics

Electronic engineering

2025

FINAL YEAR PROJECT TITLE:

AUTOMOTIVE TRACKING AND MONITORING SYSTEM

BY: TAKUDZWA MLAMBO

COURSE CODE: EEE5000

REGISTRATION NUMBER: B224392B

PROJECT SUPERVISOR: ENGINEER ZIYAMBE

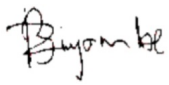
DECLARATION

This research project is my original work except where sources have been acknowledged. The work has never been submitted, nor will it ever be, to another university to award a degree or any other qualification.

STUDENT NAME: TAKUDZWA MLAMBO

DATE OF SUBMISSION……22/06/26……...SIGNATURE…………

SUPERVISOR NAME: ENGINEER ZIYAMBE

DATE OF SUBMISSION……22/06/26……...SIGNATURE: ……

COPYRIGHT

©2026

TAKUDZWA MLAMBO

ALL RIGHTS RESERVED

This dissertation is an exclusive product of scholarship and should not be copied, duplicated, transferred, or used in any way whatsoever without the explicit permission of the author and/or Bindura University of Science Education, who reserves the right to do so on behalf of the author. Any support in this regard will be highly appreciated.

DEDICATION

This work is a tribute to my parents and my buddy, Ngonidzashe, without whom I would not be able to accomplish anything in life because of his love and encouragement that have enabled me to face every hurdle successfully throughout the years of study.

ACKNOWLEDGEMENTS

I want to start by expressing my deepest gratitude to God for blessing me with the strength, health, and wisdom to complete this project. His guidance has been my anchor during the toughest moments of my academic journey.

I would like to express my gratitude to my parents for the endless love and support that they have shown me in life. The selfless sacrifice that they have made in order for me to succeed is something that will always be remembered in my heart.

Secondly, I would like to express my gratitude to my friends who were always there for me. Their moral support and motivation were crucial in helping me maintain focus during times when things got rough. Also, their company during my long hours of studying was very helpful.

Thankfully, I would like to express my gratitude to Engineer Ziyambe for the constant feedback and guidance that he gave to me during this project. This person played a critical role in helping me complete this project.

Lastly, I wish to express my sincere appreciation to Engineer Washayanyika for his kindness and encouragement. It is my hope that he continues doing what he has been doing in the future. To end, I would like to thank all the lecturers in the Department of Electronic Engineering and Physics from Bindura University of Science Education.

Abstract

Theft and misuse of vehicles have been identified as major issues for organizations operating within Zimbabwe. The problem has resulted in substantial monetary losses due to fuel theft, unauthorized vehicle use, and late deliveries. Most systems currently in use are often too costly for small and medium-sized companies. Therefore, there arises the need to develop an economical and effective system to track and monitor vehicles. This project seeks to develop an Automotive Tracking and Monitoring System using the FreeRTOS programming language and other major elements like ESP32, NEO-7M GPS, and SIM800L GSM/GPRS.

The new tracking system will provide information about real-time telemetry, geofencing, and behavioral analysis. For example, the Automotive Tracking and Monitoring System will allow the car's engine to be disabled if alcohol exceeds the set limit. During evaluation, an experimental model of the Automotive Tracking and Monitoring System was subjected to real-world and laboratory experiments on three dates: 28th March, 13th May, and 1st June 2026. It was found that the system performed quite well, achieving a 100% telemetry delivery rate and a latency of 1.18 seconds over WiFi.

Despite the system's success in monitoring driver behavior, achieving a high precision rate of 78%, it failed to meet our ≥85% target precision due to counter issues and false positives detected by the accelerometer. Nevertheless, the system successfully identified excessive alcohol consumption because the sensor consistently met the 62-second criterion. Amazingly, the hardware cost of implementing the system was below the $80 budget.

The above project shows how GPS, alcohol sensing, and remote immobilization can be incorporated affordably to help Zimbabwean SMEs. Although there were some limitations that included GSM inconsistencies in rural areas and false positives, recommendations have been made for further development of the project, which would include the use of Kalman Filtering methods and OBD II Systems.

List of Abbreviations

Abbreviation	Full Form	Application
ADC	Analog-to-Digital Converter	Converts analog sensor voltages (e.g., MQ-3 output) to digital values for ESP32 processing
API	Application Programming Interface	Enables Flespi cloud integration and data exchange between system components
APP_CPU	Application Processor Core	Core 1 of ESP32 responsible for telemetry, network, and driver behavior tasks
BAC	Blood Alcohol Concentration	Legal limit in Zimbabwe is 0.05 g/dL; system immobilises engine when exceeded
CAN	Controller Area Network	Vehicle bus standard; recommended for future OBD-II integration
Flutter	Flutter Framework	Cross-platform framework used to develop Android mobile application
FreeRTOS	Free Real-Time Operating System	RTOS running on ESP32 for deterministic multitasking and task scheduling
GPRS	General Packet Radio Service	Packet-based cellular data service used by SIM800L for cloud telemetry
GPS	Global Positioning System	Provides real-time geographical coordinates, speed, and satellite data via NEO-7M module
GSM	Global System for Mobile Communications	Enables SMS alerts, voice services, and fallback communication via SIM800L module
HDOP	Horizontal Dilution of Precision	Indicates GPS horizontal position accuracy; lower values (e.g., 0.84) are better
I²C	Inter-Integrated Circuit	Two-wire protocol connecting ADXL345 accelerometer to ESP32 (SDA/SCL pins)
IDE	Integrated Development Environment	Arduino IDE used for firmware development; Android Studio used for mobile app
IoT	Internet of Things	Enables cloud-based telemetry, remote monitoring, and data visualisation

IP	Internet Protocol	Used for TCP/IP communication between ESP32 and Flespi cloud gateway
JSON	JavaScript Object Notation	Data format for telemetry payloads sent from ESP32 to Flespi (latitude, speed, events)
LTE	Long-Term Evolution	4G cellular standard; mentioned as more reliable than GPRS for NTP synchronisation
MOS	Metal Oxide Semiconductor	Operating principle of MQ-3 alcohol sensor (tin dioxide/SnO_2 sensing layer)
MQTT	Message Queuing Telemetry Transport	Lightweight protocol used for mobile app to Flespi cloud communication
NEO-7M	NEO-7M GPS Module	UART-interfaced GPS module providing location, speed, HDOP, and satellite count
NMEA	National Marine Electronics Association	Standard GPS data sentence format parsed by ESP32 for coordinates and timestamps
NTP	Network Time Protocol	Used for time synchronisation over WiFi; failed over GPRS (replaced with GSM network time)
OBD-II	On-Board Diagnostics II	Vehicle diagnostic port; recommended for future integration to access RPM, fuel, fault codes
PAYD	Pay-As-You-Drive	Usage-based insurance model enabled by telematics data
PHYD	Pay-How-You-Drive	Usage-based insurance model based on driving behaviour rather than distance only
PRO_CPU	Protocol Processor Core	Core 0 of ESP32 responsible for GPS acquisition and alcohol sensor reading
RTOS	Real-Time Operating System	Provides deterministic task scheduling; FreeRTOS is the implementation used
SIM800L	SIM800L GSM Module	UART-interfaced cellular module for SMS, GPRS, and voice services
SMEs	Small and Medium Enterprises	Primary target users for affordable tracking system in Zimbabwe
SMS	Short Message Service	Text-based alert mechanism for engine kill, geofence breaches, and alcohol detection
SnO_2	Tin Dioxide	Sensing material in MQ-3 alcohol sensor; resistance changes when exposed to ethanol

TCP	Transmission Control Protocol	Reliable connection-oriented protocol for ESP32 to Flespi telemetry streaming
UART	Universal Asynchronous Receiver-Transmitter	Serial communication protocol used for GPS (GPIO16/17) and GSM (separate UART)
UBI	Usage-Based Insurance	Insurance model where premiums are calculated from actual driving data
UNECA	United Nations Economic Commission for Africa	Co-author of Zimbabwe Road Safety Performance Review (2021)
UNECE	United Nations Economic Commission for Europe	Co-author of Zimbabwe Road Safety Performance Review (2021)
ZRP	Zimbabwe Republic Police	Source of vehicle theft data (June 11-12, 2025 incidents)

Table of Contents

1 CHAPTER ONE: INTRODUCTION 14

1.1 Background/Overview 14

1.2 Problem definition 16

1.3 Aim 19

1.4 Objectives 19

1.5 Justification 19

1.5.1 Societal and Economic Significance 19

1.5.2 Academic and Scientific Significance 20

2 CHAPTER TWO: LITERATURE REVIEW 22

2.1 Introduction 22

2.2 Existing Research 22

2.2.1 IoT as the Enabling Framework for Automotive Telematics 22

2.2.2 Cost-Effective Vehicle Tracking Using GPS and GSM 24

2.2.3 The Role of Real-Time Operating Systems in Embedded Automotive Systems 26

2.2.4 Geofencing for Operational Control and Fleet Management 27

2.2.5 Remote Engine Immobilization for Anti-Theft Security 28

2.2.6 Alcohol Detection Systems for Driver Safety and Vehicle Immobilization 30

2.3 Synthesis and Identification of Research Gaps 33

2.4 Positioning of the Proposed System 34

3 Methodology 36

3.1 Methodology Introduction 36

3.2 Project Approach 36

3.3 System Requirements Specification 36

3.3.1 Hardware Requirements 36

3.3.2 Software Requirements Specification 40

3.4 System Architecture 41

3.5 System Flowchart 42

3.6 System Design and Implementation 43

3.6.1 Hardware design 43

3.6.2 Software Design 45
3.6.3 Circuit design 55
4 Chapter 4: EXPERIMENTAL RESULTS AND ANALYSIS 56
4.1 Introduction 57
4.2 System Implementation and Results 58
4.2.1 Hardware Implementation 58
4.2.2 Hardware Results 62
4.2.3 Software and Cloud Integration Results 64
4.3 Experimental Results and Data Presentation 66
4.3.1 System Response Time and Telemetry Performance Analysis 66
4.3.2 System Reliability and Communication Performance 68
4.3.3 GPS Accuracy and Positioning Analysis 70
4.3.4 Sensor Suite Results 72
4.3.5 Driver Behavior Monitoring Results 73
4.3.6 Alcohol Detection System Performance Analysis 75
4.3.7 Engine Immobilization Performance 79
4.4 Analysis and Interpretation 80
4.4.1 Objective 1 Analysis 80
4.4.2 Objective 2 Analysis 80
4.4.3 Objective 3 Analysis 81
4.4.4 Objective 4 Analysis 82
4.5 Discussion of Findings 82
5 CHAPTER FIVE: CONCLUSIONS AND RECOMMENDATIONS 85
5.1 Introduction 85
5.2 Summary of the Research 85
5.3 Achievement of Research Objectives 86
5.4 Conclusions 87
5.5 Limitations of the System 89
5.6 Recommendations for Future Work 90
6 References 92

List of Figures

Figure 2.1: shows the perceived contribution of telematics to auto-insurance in Zimbabwe, with billing (90%) and intelligence (87%) rated highest 23
Figure 2.2:Superloop architecture 27
Figure 2.3:General structure of FreeRTOS 27
Figure 2.4:Geofence 28
Figure 2.5: Fuel pump current and flow characteristics vs. voltage at constant system pressures (100–700 kPa). At 13.5V and 5 bar, current draw is approximately 7.5A. (Source: Bosch Motorsport, n.d.) 29
Figure 3.1:ESP32 37
Figure 3.2: GPS MODULE 37
Figure 3.3:GSM MODULE 38
Figure 3.4: RELAY MODULE 38
Figure 3.5: LM2596 BUCK CONVERTER 39
Figure 3.6:3 Axis Accelerometer 39
Figure 3.7: MQ3 sensor 40
Figure 3.8: System Flow chart 42
Figure 3.9:flespi gateway 49
Figure 3.10:Flespi Channel Configuration 50
Figure 3.11:flespi Device configuration 50
Figure 3.12:flespi Map Dashboard 51
Figure 3.13:Grafana for system analytics 52
Figure 3.14: Login page for Mobile Application 53
Figure 3.15:Main dashboard for the Mobile Application 54
Figure 3.16:Circuit Design 55
Figure 4.1: Vero board 58
Figure 4.2 : GSM module interfaced with the ESP32 via UART pins 59
Figure 4.3: GPS interfaced with the ESP32 via UART pins 60
Figure 4.4: Accelerometer interfaced with ESP32 via I2C 60
Figure 4.5: MQ3 Interfaced with ESP32 via a single wire 61
Figure 4.6: relay switching ON and OFF 61
Figure 4.7: ESP32 connected to the LM2596 62
Figure 4.8: Grafana visual representation of message latency vs. time over the WiFi network (13 MAY 2026) 67
Figure 4.9: Grafana Visualization of Message Latency versus Time over GSM/GPRS Network (1 June 2026) 69
Figure 4.10: Grafana visual representation of HDOP, Satellites over Time 71

Figure 4.11: X-axis and Y-axis Accelerometer Telemetry During Driver Behaviour Monitoring Experiments 74
Figure 4.12: MQ-3 Alcohol Sensor ADC Response During Severe Alcohol Exposure.....78
Figure 4.13:MQ-3 Alcohol Sensor Voltage Response During Severe Alcohol Exposure 78

List of Tables

Table 1.1: Summary of Vehicle Theft Incidents Reported by the Zimbabwe Republic Police (June 11-12, 2025) .....**Error! Bookmark not defined.**
Table 1.3: Driver Management Deficiencies in Zimbabwean State-Owned Enterprises 17
Table 2.1:Comparative Analysis Table of Existing Tracking Systems vs. Proposed System.....32
Table 4.1: Hardware Implementation Results.....63
Table 4.2: Software Implementation Results 64
Table 4.3: End-to-End Telemetry Latency Results 66
Table 4.4: Communication Reliability Results 68
Table 4.5: GPS Performance Results 70
Table 4.6: Sensor Monitoring Results.....72
Table 4.7: Driver Behavior Event Analysis 73
Table 4.8: Alcohol Sensor Configuration Parameters.....75
Table 4.9: Alcohol Sensor Baseline Characteristics.....75
Table 4.10: Minor Alcohol Detection Event Results 76
Table 4.11: Severe Alcohol Detection Event Results 77
Table 4.12: Engine Immobilization Response Results.....79
Table 4.13: Hardware Expenditure for the prototype 83

1 CHAPTER ONE: INTRODUCTION

1.1 Background/Overview

The use of cars in Zimbabwe goes far beyond merely being a mode of transport, as they play an indispensable role in both economic and personal terms. In general, purchasing and maintaining cars constitutes a big deal for most people because it ensures their mobility in different spheres of life. Likewise, business owners need to maintain large fleets to facilitate their activities. However, over time, managing these vehicles became increasingly complicated and expensive for both individuals and firms.

Today, organizations that depend on their vehicle fleets have to tackle far more serious issues with misuse and theft than they did several years ago. Namely, in comparison with five years ago, the problem has become much bigger. In the absence of affordable real-time vehicle-tracking options, it becomes easier than ever for drivers to go off route unnoticed. The result is obvious: fuel losses and higher expenses due to increased wear and tear. These expenses have been growing rapidly recently, exceeding the financial opportunities of many businesses (Mhazha, 2025). According to current police statements, there is an upsurge in car theft cases in Harare and surrounding towns, and companies are suffering silently due to fuel theft, unauthorized trips, and late deliveries. The absence of real-time tracking makes it difficult to notice these problems before they become serious.

Table 1.1: Summary of Vehicle Theft Incidents Reported by the Zimbabwe Republic Police (June 11-12, 2025)

Incident Date & Time	Location	Vehicle(s) Stolen
June 12, 2025	Avonlea, Harare	Toyota Hilux & Mazda BT50
June 11, 2025, 21:00	Urugwe Compound, Mutorashanga	Honda Fit
June 11, 2025	Chiedza, Karoi	Toyota Fun Cargo
June 11, 2025, 22:00	Fairview Park, Ruwa	Nissan Caravan NV350
June 12, 2025, 20:00	Corner of Leopold Takawira & Samora Machel Ave, Harare	Honda Fit

Source: (Zimbabwe Republic Police, 2025)

As highlighted in Table 1.1, vehicle theft is alarmingly common in various regions. The above situations clearly suggest a dire need for better alternatives that do not merely ensure surveillance of the vehicles but that protect the vehicles from theft. The bottom line is that such alternatives would ensure that individuals can enjoy peace of mind.

However, apart from the immediate threat of theft, another insidious and deadly issue is the financial and human loss resulting from unregulated driver fitness and behavior. According to the Traffic Safety Council of Zimbabwe (TSCZ), as cited in the United Nations Road Safety Performance Review of Zimbabwe (United Nations Economic Commission for Europe (UNECE) and United Nations Economic Commission for Africa (UNECA, 2021) Road accidents cost the nation approximately US$406 million annually, equivalent to 3% of the country's Gross Domestic Product. The same review confirms that 94% of road crash fatalities and injuries in Zimbabwe are attributable to human error, with only 5% due to vehicle defects and 1% due to road environment factors. The Zimbabwe Republic Police reported a 17% increase in traffic accidents between 2021 and 2022; excessive speeding and driving under the influence are constantly recognized as the most hazardous behaviors leading to accidents in Zimbabwe (chiyangwa et al., 2020).

Critically, Young, inexperienced drivers are 2.5 times more prone to having an accident when their blood alcohol concentration reaches 0.05 g/dL or above (Eun, 2021). Despite enforcement efforts through roadblocks and police checkpoints, drunk driving continues unabated due to three factors: (1) the impossibility of testing every driver before every trip, (2) documented corruption where drivers offer bribes to avoid consequences (Munuhwa et al., 2021), and (3) the reactive nature of enforcement; by the time an impaired driver is caught, they are already on the road posing an active threat to other road users.

A monitoring and tracking system for vehicles is necessary since it provides an economic and efficient solution by addressing the problems present in the current device designs. Not only does it emphasize location tracking, but it also provides security and intelligence capabilities. The success of the system will be dependent on the ability of the microcontroller used in the system to support FreeRTOS operating system, which allows multiple processes to run at once. This includes real-time tracking of vehicle information as well as real-time monitoring of different security parameters like remote engine shutdown. Also included in the design is the breathalyzer sensor (MQ-3), which tests the driver's breath to ensure he or she is sober before

the engine can start. If the blood alcohol content (BAC) surpasses the 0.05 g/dL threshold in Zimbabwe, the engine relay is terminated immediately and an SMS is sent to the car owner.

These consist of several hardware components that serve crucial roles to guarantee the effectiveness of the entire solution in tracking and communicating with the cars. The first hardware component in this case is the GPS unit, whose function is to continuously track the location and speed of the car. The second important hardware component is the GSM module, which helps in communication functions, such as sending SMS messages.

The system design focuses on ensuring continuous connectivity and efficient transmission of data in places where there may be minimal cell network availability. The overall aim of the design is to reduce operational costs of the system. Besides that, another important device which can be included in the system design is a Wi-Fi device integrated within the microcontroller which would enable the tracker to connect to pre-configured wireless networks.

Data transmitted through the GPRS network and the Wi-Fi network ensures improved fleet management, proactive maintenance, and analysis of the driver's performance. Moreover, the use of a 3-axis accelerometer will ensure the detection of abrupt braking, sudden acceleration, and turning actions that could increase fuel consumption and result in reckless driving, as well as wear and tear on the car itself. Combined with the information provided by alcohol sensors, fleet owners will be able to determine driver fitness and behavior on the roads.

1.2 Problem definition

In Zimbabwe, problems like theft of vehicles, irresponsible driving while under the influence of alcohol, and improper use of company fleet cars create problems for both individual motorists and businesses that utilize transportation services. The Zimbabwe Republic Police have made significant complaints about the increase in car theft cases in the country, according to reports by ZIMNOW. But one problem that may be even bigger than car theft in terms of negative effects and economic cost to the country is the absence of technologies used to detect whether motorists have been drinking before starting their cars.

The human factors study on road traffic safety in Harare establishes the following critical evidence:

- **Magnitude of the Problem:** Road traffic accidents (RTAs) cost Zimbabwe approximately US$406 million annually — 3% of GDP — with drunk driving and speeding listed as the most frequent offenses leading to accidents (United Nations Economic Commission for Europe (UNECE) and United Nations Economic Commission for Africa (UNECA, 2021);(1chiyangwa et al., 2020)).
- **Increased Risk:** Inexperienced young drivers are 2.5 times more at risk of crashing when their blood alcohol concentration reaches 0.05 g/dL or above (Eun, 2021).
- **Enforcement Failure:** Corruption allows unlicensed drivers and drunk drivers to evade consequences through bribes at checkpoints. Common offenses associated with road bribes include unsafe vehicles, unlicensed drivers, overloading, excessive speed, and driving under the influence (chiyangwa et al., 2020).
- **Reactive vs. Preventative:** Current enforcement is reactive — police cannot test every driver before every trip. By the time a drunk driver is detected, they are already on the road, endangering lives.

Empirical research specific to Zimbabwe confirms that driver management deficiencies are systemic.(Chiparo et al., 2022) surveyed 86 State-Owned Enterprises and found consistent evidence of poor driver management practices across five key indicators, as shown in Table 1.2.

Table 1.2: Driver Management Deficiencies in Zimbabwean State-Owned Enterprises

Driver Management Issue	Mean Score (1-5)	Response	Standard Deviation
Organizations do NOT prioritize driver behavior and education	3.99	Agree	1.007
Organizations do NOT hold regular driver forums	4.17	Agree	1.081
There are transparent driver recruitment procedures	3.84	Agree	0.978
Organizations do NOT have driver recognition/incentive schemes	3.88	Agree	1.013

There are NO continual trainings and coaching for drivers	4.06	Agree	1.008
Average	**3.99**	**Agree**	**1.017**

5-point Likert scale where 1 = Strongly Disagree, 3 = Neutral, 5 = Strongly Agree. Higher scores indicate greater agreement with the negative statement about driver management deficiencies.

Source: Adapted from Chiparo, Tukuta and Musanzikwa (2022, p. 5)

The results from the research show a general agreement amongst the respondents with an average score of 3.99 out of 5 on the agreement scale concerning the statement that the organization may ignore some key elements of the driver behavior. It can be seen that there is an absence of forums for discussion among drivers on the issue, as well as inadequate reward systems for good driving performance and training for drivers. Therefore, it can be concluded that there is a failure in customer service delivery. This shortfall is evidenced by persistent challenges, such as prolonged vehicle downtime and frequent breakdowns, which remain issues despite established fleet management frameworks (Chiparo et al., 2022).

In addition to being quite costly, existing products in the automotive tracking and monitoring industry are also inadequate when it comes to offering all-rounded safety precautions. These products are limited to just tracking the location of the vehicle and fail to provide any proactive security measures, like engine immobilization and tracking the behavior of the driver. But most importantly of all, there is no feature that would prevent an intoxicated driver from using the car.

To address these concerns, the research has taken the approach of developing and evaluating the Automotive Tracking and Monitoring System. Such a system will fill some of the missing links in providing a relatively cheap and real-time security solution that includes tracking as well as preventive measures of enforcing safe driving conditions, especially in terms of avoiding drunk driving. The ultimate goal of such a system will be improving automotive safety overall.

1.3 Aim

To design, develop, and experimentally validate a low-cost automotive tracking and monitoring system for mitigating vehicle theft and misuse in Zimbabwe, integrating real-time GPS tracking, accelerometer-based driver behavior monitoring, geofencing, remote engine immobilization, IoT-based data visualization, and alcohol detection with engine start prevention.

1.4 Objectives

1. To design and assemble a low-cost (≤ $80) ESP32-based automotive tracking device powered by a 12V vehicle supply, integrating GPS (NEO-7M), GSM (SIM800L), an accelerometer, and running FreeRTOS, by 15 January 2026.
2. To configure Flespi and Grafana to receive, process, and visualize vehicle data at 1-second intervals, achieving ≥95% successful transmission with ≤2-second latency, by 6 February 2026.
3. To develop an Android application that provides real-time vehicle tracking (≤2-second delay), system monitoring, geofence alerts, and remote engine immobilization, achieving ≥90% functional success rate during testing, by 01 April 2026.
4. To implement accelerometer-based detection of harsh braking, acceleration, and cornering using predefined thresholds, achieving ≥85% detection accuracy by 20 April 2026.

1.5 Justification

This study is highly significant from both the point of view of its immediate practical use and the knowledge it contributes. The significance of the study can be seen in three main ways: societal, economic, and academic.

1.5.1 Societal and Economic Significance

The importance of this study to society and the economy is significant. Regarding the monitoring of driver behavior and fleet management, According to (Chiparo et al., 2022), the firms in Zimbabwe are not bothered about driving behaviors of the drivers, and hence they do not have driver education schemes and training initiatives to improve their driving performance. However, the use of an accelerometer system helps to

resolve the issue by offering objective metrics on issues such as harsh braking, acceleration, and sharp turns.

As far as vehicle protection from unauthorized movement and theft is concerned, modern GPS-enabled tracking systems offer real-time notifications about vehicles' movement. For example, as shown in Digital Matter (2024), GPS-enabled tracking systems not only protect vehicles from theft but also allow owners to protect other property, including construction equipment and beehives, by alerting them if someone moves assets without permission. In addition to the ability to track vehicles, using an engine kill relay will significantly enhance this system's capabilities. At that, as with protecting cars from theft, the engine kill relay will also make the device useful for immobilizing vehicles driven by drivers under the influence of alcohol.

Considering its role in resilience and protection in economics, this system is particularly important for Small and Medium Enterprises (SMEs), as it is their safeguard against financial crises caused by fuel theft, vehicle misuse, inappropriate travel routes, and even liability costs incurred in road accidents. In Zimbabwe, according to reports, road accidents cause annual economic losses of US$406 million. Apart from addressing this problem, the system addresses the challenge posed by vehicle downtime and breakdowns, which Chiparo et al. (2022) identify as one of the challenges facing organizations in Zimbabwe. All in all, it makes vehicle management more efficient and economical (Mhazha, 2025).

Accessibility and affordability are two crucial elements of the research project. By using affordable technologies such as the ESP32 microcontroller and the MQ-3 alcohol sensor, this research project aims to make this smart vehicle security system accessible to many potential customers who have been out of reach due to expensive devices on the market.

1.5.2 Academic and Scientific Significance

This study also makes important contributions to academic knowledge through the novel integration of existing technologies within a specific socio-economic context. A review of the literature reveals that published works typically focus on fleet management, diagnostics, or vehicle security in isolation; almost none integrate alcohol detection with engine immobilization and fleet telemetry. Moreover, while Chiparo et al. (2022) identified deficiencies in driver management within Zimbabwean

organizations, they did not propose a technological solution. The present study fills that gap by providing an automated monitoring system that objectively records driver behavior and generates data suitable for training, recognition, and accountability. Few systems integrate GPS tracking, engine immobilization, breath alcohol analysis, accelerometer-based behavior monitoring, and voltage monitoring within a holistic architecture that can be implemented in low-resource environments.

The present project also contributes to academia through an innovative application of existing technologies within a specific socio-economic environment. From the analysis of the literature, it becomes clear that most articles have addressed fleet management, diagnostics, or vehicle security in isolation, and very few have focused on alcohol detection. In addition to filling existing gaps in the literature, the present research makes an important contribution to advancing engineering practices. The implementation of multitasking with FreeRTOS on an ESP32 microcontroller highlights real-world techniques for task allocation and scheduling in automobiles. Moreover, it has been shown that a variety of sensors, including GPS, GSM, an alcohol sensor, and an accelerometer, can be combined into a single system to provide useful methodologies for future projects. The present research also contributes to a developing area of power management solutions specifically designed for automotive electronics operating from 12V batteries.

2 CHAPTER TWO: LITERATURE REVIEW

2.1 Introduction

This chapter provides a summary of the research and technology currently available in the fields of IoT, vehicle tracking, real-time operating systems, geofencing, remote vehicle immobilization, and alcohol detection systems. The review will be separated into six parts. Section 2.2.1 examines the background of the study on the use of IoT as an enabling platform for automotive telematics. Section 2.2.2 focuses on existing vehicle-tracking systems, such as affordable GPS/GSM systems. In Section 2.2.3, the paper focuses on the use of real-time operating systems in embedded automotive systems. Section 2.2.4 examines geofencing and remote immobilization technologies. Section 2.2.5 brings in vehicle safety and alcohol detection systems for immobilization. Section 2.3 summarises the literature to identify research gaps, and Section 2.4 locates the proposed system within the gaps. This chapter provides a clear justification for the proposed Automotive Tracking and Monitoring System to address the needs of areas such as Zimbabwe by first analysing and identifying gaps in existing systems with respect to affordability, integration, and situational adaptability.

2.2 Existing Research

2.2.1 IoT as the Enabling Framework for Automotive Telematics

The Internet of Things (IoT) is a paradigm shift whereby common items are fitted with sensors, programs, and network connectivity to gather and share information (Chataut et al., 2023).IoT technologies have revolutionized various sectors, including healthcare, agriculture, smart cities, and Industry 4.0, enabling real-time monitoring and remote management.

IoT technologies may also be used in the automotive industry to enhance the functionalities of car systems, providing not just tracking but also security, diagnostics, and fleet management. According to (Chataut et al., 2023), IoT is considered a fusion of digital intelligence and the real world to create proactive systems, not reactive ones. Yet, most of IoT systems require continuous access to affordable and fast internet, something that might not be present in poorly-equipped areas and high-cost data

countries. Therefore, an IoT solution that employs both SMS communications for emergency purposes and GPRS communications for data transfer becomes vital in these circumstances. In the current work, the project team opted for a hybrid system approach.

The integration of telecommunications, information systems, and automatic control known as telematics has come up as an essential technology in managing vehicles in Zimbabwe (Denhere et al., 2024). With more than 1.5 million vehicles on Zimbabwe roads in 2022, telematics can provide a viable means of handling the increasing complexity of the transport and insurance sectors. Telematics turns vehicles into a platform for mobile connectivity through which different services can be provided, such as vehicle theft tracing, safety and security, diagnostics, and driver behavior monitoring (Denhere et al., 2024). (Denhere et al., 2024) It is important to note that telematics plays a significant part in road safety intelligence (80%), risk assessment (87%), crash investigations (84%), and improving driving behaviors (47%) according to stakeholder survey responses in Zimbabwe. Critically, driver behavior is shaped by the penalties for reckless driving, with Usage-Based Insurance (UBI) models such as Pay-As-You-Drive (PAYD) and Pay-How-You-Drive (PHYD) incentivizing safer driving habits (Denhere et al., 2024). The recommended Automotive Tracking & Monitoring System seems to be quite close to the telematics value chain, providing an extensive information infrastructure – from GPS tracking to driver behavior analysis based on accelerometers and detection of alcohol consumption – needed for future UBI implementations in Zimbabwe.

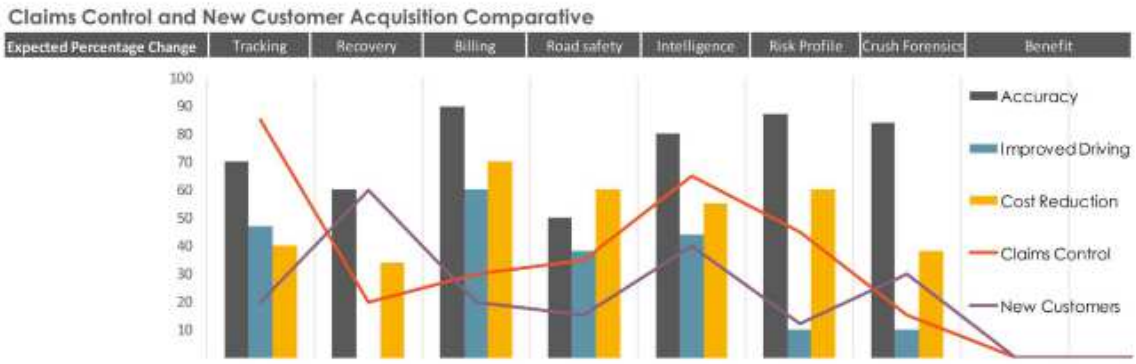


Figure 2.1: shows the perceived contribution of telematics to auto-insurance in Zimbabwe, with billing (90%) and intelligence (87%) rated highest

Source: (Denhere et al., 2024)

As illustrated in Figure 2.1 above, (Denhere et al., 2024) empirically quantified the perceived contribution of telematics to auto-insurance across seven key metrics based on survey data collected from Zimbabwean insurance industry stakeholders.

The highest-rated metric is billing at 90%, reflecting the insurance industry's recognition that telematics enables Usage-Based Insurance (UBI) models such as Pay-As-You-Drive (PAYD) and Pay-How-You-Drive (PHYD), where premiums are calculated based on actual driving behaviour rather than demographic averages.

The second-highest metric is intelligence at 87%, indicating that telematics provides real-time data on driver behavior, vehicle condition, and road dynamics, enabling data-driven decision-making for both fleet managers and insurers.

Risk profiling at 84% demonstrates that insurers can accurately classify drivers as high- or low-risk based on objective telematics data, while road safety at 80% justifies proactive interventions, such as the MQ-3 alcohol sensor, for drunk-driving prevention.

The remaining metrics — tracking (70%), claims control (65%), and recovery (60%) — support the integration of GPS (NEO-7M) and GSM (SIM800L) modules for real-time location monitoring and stolen vehicle recovery, while also validating the automated crash reporting capabilities of the proposed system.

2.2.2 Cost-Effective Vehicle Tracking Using GPS and GSM

A significant body of research has concentrated on developing affordable automotive tracking systems using GPS and GSM technologies. (Akanda et al., 2022) created an affordable and easy-to-use tracking solution using the SIM808 GSM/GPS shield along with an Arduino Uno. Their system uses coordinates (latitude and longitude) along with a Google Maps link for an SMS reply, indicating how affordable basic tracking can be achieved. Their approach does not have any security aspects like geofencing or even immobilization.

Previous studies have also explored GSM-based tracking with varying degrees of functionality. (Jospine et al., 2020) Developed a university vehicle-tracking application with the help of the FONA808 module and Blynk IoT platform, which included geofencing with the option to send emails if the vehicle left the designated area. The

above study is an example of how geofencing can be used by institutions or corporate vehicles.

These studies prove that GSM networks ensure a viable and extensively distributed network infrastructure perfect for regions without good internet connectivity. However, one problem that is frequently encountered is the absence of an IoT data stream, making it difficult for the device to perform fleet management and monitoring operations. The proposed solution addresses this problem through the application of GPRS telemetry alongside SMS.

(Alhwety and Elfadil, 2024) A systematic literature review was conducted on various approaches to vehicle tracking systems, which were divided into active (real-time) and passive (data storage) categories. The results show that active tracking systems, which consist of GPS and GSM modules, offer real-time location information, geofencing capability, and ways to prevent theft. According to the researchers, "incorporating GSM modules such as SIM800L greatly enhances the capabilities of such tracking systems, allowing for real-time communication" (Alhwety and Elfadil, 2024).

On the other hand, (Alhwety and Elfadil, 2024) have identified various limitations associated with current GPS tracking technologies. Limitations such as inaccurate location in the presence of signal obstruction, loss of signals owing to lack of satellite connectivity, depletion of batteries owing to power consumption, susceptibility to data hacking, high cost of implementation, risks of technical issues, and challenges in remote locations are noted (Alhwety and Elfadil, 2024).

In addition to this, the designed Automotive Tracking and Monitoring System manages these challenges through strategic planning. First, incorporation of an alternative SMS/GPRS backup communication protocol will help mitigate problems associated with signal loss, hence making sure important information is sent despite the unavailability of the internet. Secondly, adoption of FreeRTOS task management using the dual-core processor ESP32 improves power consumption since unnecessary tasks will go to low power modes. Third, the economic use of affordable components contributes towards these aims.

Low hardware cost, making it less than $80, solving cost challenge according to (Alhwety and Elfadil, 2024). Fourth, the incorporation of Flespi cloud server with authenticated and encrypted communication greatly improves system security. Furthermore, the geofencing feature coupled with SMS fallback communication ensures that boundary crossing alerts are made available, thus addressing the challenge of poor coverage in isolated locations.

2.2.3 The Role of Real-Time Operating Systems in Embedded Automotive Systems

As the sophistication of vehicle tracking and monitoring systems increases, the importance of software architecture cannot be understated for ensuring reliability and real-time processing. Superloop architecture, although simple and lightweight, inherently runs processes one after another. Running processes one after another can lead to undesirable latencies, particularly in cases where processes need to be run simultaneously. Therefore, this problem compels us to explore other software architectures that would improve responsiveness in multitasking systems(Zalizovskyi and Huzynets, 2024). Unlike this architecture, Real Time Operating Systems (RTOS) like FreeRTOS allow deterministic task scheduling and concurrent processes (Zalizovskyi and Huzynets, 2024).

According to Zalizovskyi and Huzynets (Zalizovskyi and Huzynets, 2024), a comparison of RTOS and Superloop architectures has been conducted, and it was found that RTOS is better for those applications which require multitasking, predictability in terms of time, and scalability of the system. It is particularly true for the automotive systems that need to perform several functions like GPS data acquisition, GSM connectivity, monitoring of sensors, and security operations at the same time. In this context, ESP32 microcontroller is ideal as it comes with a dual core and also supports FreeRTOS using ESP-IDF software package (Anon., 2022).

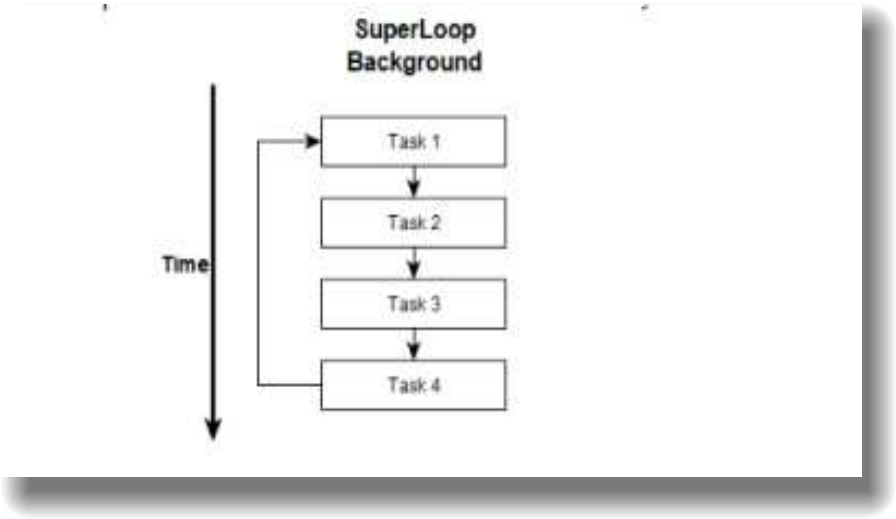


Figure 2.2:Superloop architecture

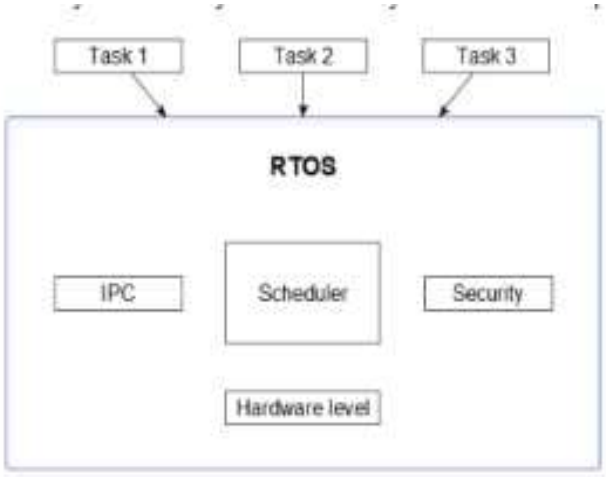


Figure 2.3:General structure of FreeRTOS

2.2.4 Geofencing for Operational Control and Fleet Management

Geofencing, which entails drawing virtual boundaries, is another useful function that can be used to manage the fleet and curb any illegal usage of vehicles. (Jospine et al., 2020) have utilized the Blynk IoT platform to develop an innovative vehicle tracking system that sends email alerts whenever the vehicle crosses the virtual boundaries of a particular area. This application of geofencing can prove useful to institutions like universities and other corporate fleets.

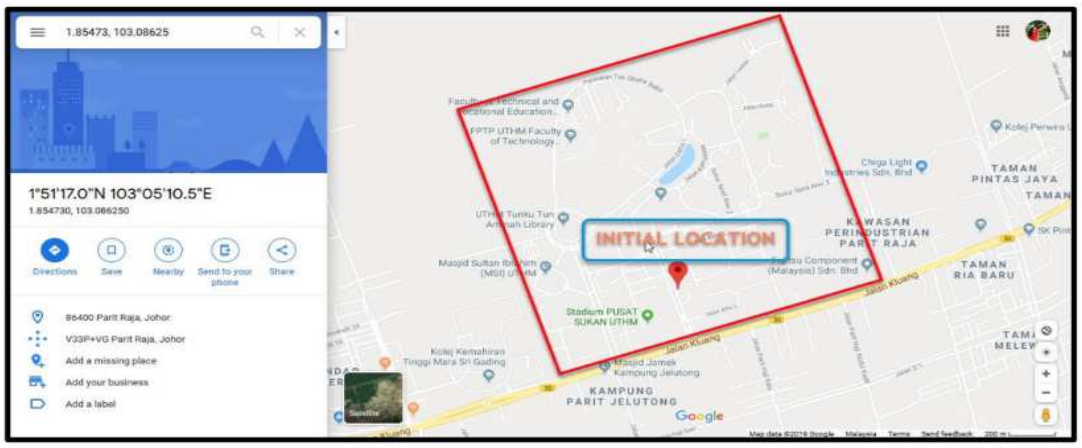


Figure 2.4:Geofence

However, existing geofencing solutions often depend solely on GPS data and internet-dependent alerts, which may fail in places with poor connectivity. To address this, the proposed system integrates SMS-based geofence alerts as a fallback mechanism, ensuring notifications are delivered even when internet access is unavailable. This dual alert strategy enhances reliability and guarantees that fleet managers are promptly informed of boundary violations, thereby mitigating the risk of vehicle abuse.

2.2.5 Remote Engine Immobilization for Anti-Theft Security

The engine immobilizer is an essential component of the automobile security system which is used to safeguard the automobile from being stolen or hi-jacked. According to (Jimada-Ojuolape et al., 2023), the demobilizer was created using the existing mobile phone and the NE555 timer. The activation process takes place once the demobilizer gets the phone calls. This proves that the cheap components like the GSM module and other simple components could be employed to construct the security system. In similar manner, the demobilizers would be triggered through the SMS commands sent via mobile application or SMS and through geofencing.

Although the two mechanisms described above are quite effective, they have a flaw because the two systems do not have any integration in their monitoring and tracking processes. In the process, a combination of these two mechanisms can offer an integrated solution to this problem. In this project, the immobilizer can be activated through commands sent from either an SMS or an application. Additionally, the system

will be triggered through security alerts sent when there are irregularities detected from both geofencing and usage pattern monitoring.

The immobilizer works on the principle of a 5V relay module of 30VDC / 10A (Handson Technology, n.d., p. 2). For verification purposes, the fuel pump current requirement was obtained from a Bosch Motorsport graph shown in Figure 2.5 below. From this figure, we can deduce that the fuel pump requires about 7.5A from 13.5V at 5bar fuel pressure. Since the relay is rated at 10A, it has 25% safety margins. Additionally, the relay's optocoupler provides electrical isolation between ESP32 (3.3V) and the 12V car system (Handson Technology, n.d., p. 1).

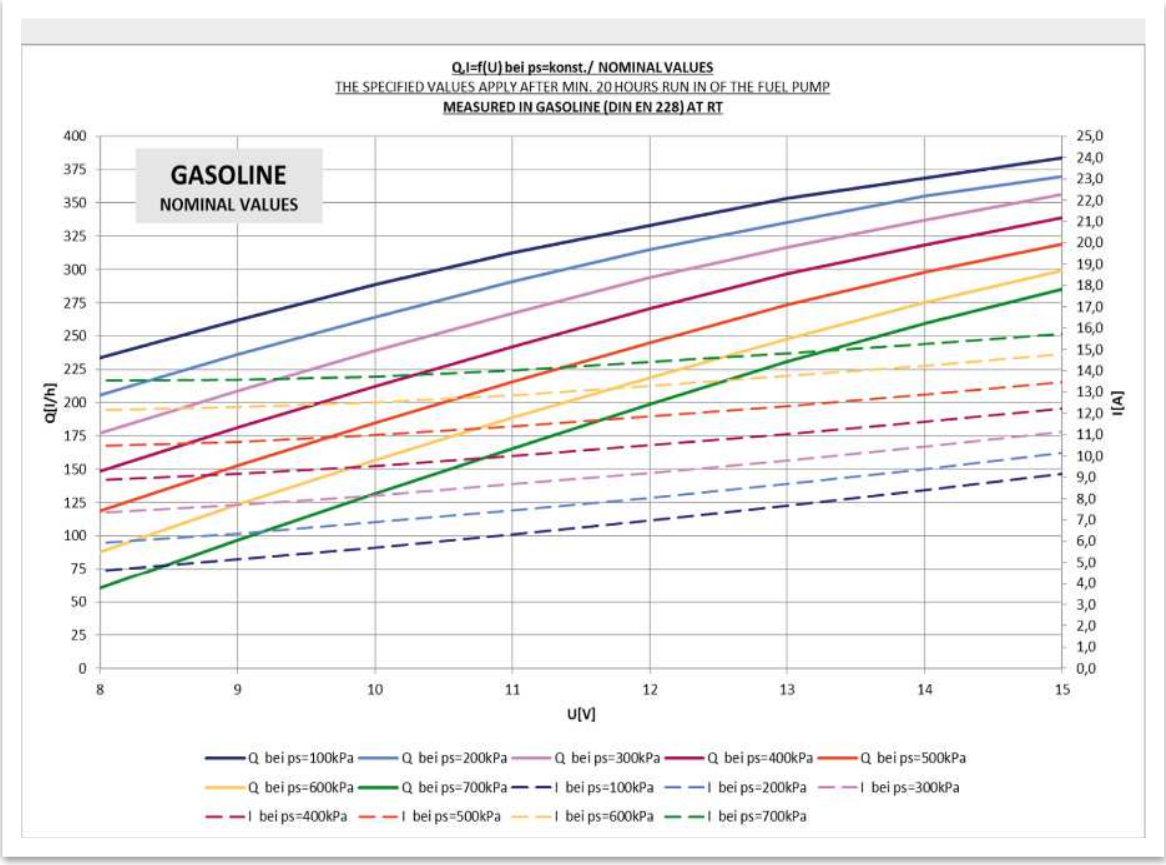


Figure 2.5: Fuel pump current and flow characteristics vs. voltage at constant system pressures (100–700 kPa). At 13.5V and 5 bar, current draw is approximately 7.5A. (Source: Bosch Motorsport, n.d.)

2.2.6 Alcohol Detection Systems for Driver Safety and Vehicle Immobilization

The lack of measures to stop drunk drivers constitutes one of the main disadvantages of existing vehicle tracking systems. As (Alabi et al., 2024) state, impaired drivers who drink before or while driving represent one of the most prevalent causes of accidents. The researchers suggest that drinking influences decision-making and response time as well as makes a driver less able to see potential dangers on the road and react appropriately to such. That is why much attention has been paid to developing technologies aimed at detecting drunk drivers.

(Alabi et al., 2024) created an alcohol detector driver system utilizing MQ-3 alcohol sensor, NodeMCU/Arduino microcontroller, GSM module, GPS tracker, and relay control mechanism for shutting down the fuel pump. Alcohol in drivers' breath triggers the system that creates an audio signal, sends alcohol content information to cloud services and turns on the relay for stopping fuel pumps when alcohol is found to exceed the predetermined amount. In addition, vibrations indicate accident and send signals and location data to hospitals close to where the accident happens.

Practical issues regarding the implementation of such systems based on the use of alcohol are additionally discussed in the paper. A MQ-3 sensor was placed close to the steering wheel where it could measure the driver’s breath. The other element – the relay 12V car was linked to the circuit of the vehicle’s fuel pump in order to avoid an abrupt shutdown of the engine while operating.

(Dakhare et al., 2022) Investigated the sensing characteristics of the MQ-3 alcohol sensor for ethanol detection at room temperature using an Arduino Uno platform. Their experimental results demonstrated that the MQ-3 sensor can successfully distinguish between various alcohol samples, including ethanol, methanol, isopropyl alcohol, wine, and beer. The study confirmed that the sensor exhibits high selectivity towards ethanol

and provides rapid response characteristics suitable for real-time alcohol monitoring applications.

As described by Dakhare et al. (2022), the MQ-3 sensor functions as a Metal Oxide Semiconductor (MOS) sensor, with a sensing layer of tin dioxide (SnO_2). The MQ-3 sensor operates on 5V power and measures alcohol levels in the range of about 25 ppm to 500 ppm. The presence of alcohol vapor in the atmosphere around the sensing layer changes the sensor's resistance, generating an output voltage proportional to the alcohol concentration. Both analog and digital sensors allow the use of the MQ-3 sensor in embedded systems.

However, despite the aforementioned strengths, both studies point out various limitations of inexpensive alcohol-measuring techniques. As pointed out by Alabi et al. (2024), the maximum measurement error of their MQ-3 device was about 35% based on measurements performed with a professional breathalyzer, while still maintaining a similar level of performance to that of a low-end commercial breathalyzer. Moreover, the accuracy of their method could be improved by choosing a lower blood alcohol concentration threshold.

Similarly, (Dakhare et al., 2022) observed that although the MQ-3 sensor could differentiate between alcohol samples at identical concentrations, output values overlapped when different alcohol types were tested at varying concentrations. This indicates that calibration and threshold optimisation are essential for improving reliability in real-world environments. Furthermore, the sensor's performance may be affected by environmental conditions such as temperature variations, humidity, and sensor aging.

(Bobade, 2023) Developed an immobilizing device that requires authorized fingerprints to start the car, with the inclusion of GPS technology that gives instant alerts in case any unauthorized entry is detected. While this innovation highlights the importance of biometric recognition in securing the car, it still does not solve the problem of alcohol intoxication.

Although these studies demonstrate the feasibility of low-cost alcohol detection and vehicle immobilisation systems, they focus primarily on alcohol sensing and engine locking. They lack any complete solution for integration with geofencing, driver behavior analysis, cloud-based fleet analysis, and FreeRTOS based real-time task management. In view of the above shortcomings, this study proposes an Automotive Tracking and Monitoring System that incorporates MQ-3 alcohol detector with a GPS tracker, GSM communication, geofencing, driver behavior analysis using an accelerometer sensor, cloud telemetry using Flespi and Grafana, and task management using FreeRTOS. Furthermore, this proposed system locks the engine if the estimated BAC exceeds 0.05g/dl, the permissible alcohol level in Zimbabwe, at the same time sending SMS alerts.

Table 2.1:Comparative Analysis Table of Existing Tracking Systems vs. Proposed System

Criteria	**(Akanda et al., 2022)**	**(Jospine et al., 2020)**	**(Alabi et al., 2024)**	**Proposed System**
Microcontroller	Arduino UNO R3	Arduino Mega 2560	NodeMCU/Arduino	ESP32
Software Architecture	Superloop	Superloop	Arduino-based	FreeRTOS
Geofencing	Not present	Yes	No	Yes
Engine Immobilization	No	No	Yes (alcohol-triggered)	Yes (SMS, app, or alcohol)
Alcohol Detection	No	No	Yes (MQ-3)	Yes (MQ-3)
Accident Detection	No	No	Yes (vibration sensor)	Planned
Cloud/IoT Integration	No	Blynk	ThingSpeak	Flespi + Grafana
Driver Behavior Monitoring	No	No	No	Yes (accelerometer)
Proactive Prevention	No	No	Yes (pre-start)	Yes (pre-start)
Communication Method(s)	SMS only	Internet (WiFi/GPRS)	Internet	Hybrid (SMS + GPRS/WiFi)

Sources: (Akanda et al., 2022; Jospine et al., 2020; Jimada-Ojuolape, Spencer and Balogun, 2023; Alabi et al., 2024)

2.3 Synthesis and Identification of Research Gaps

The analysed literature shows several consistent research gaps:

1. Lack of combining Solutions: Most studies target independent functionalities — tracking, geofencing, immobilization, or alcohol detection, without integrating them into a single system ((Akanda et al., 2022); (Jimada-Ojuolape et al., 2023); (Jospine et al., 2020); (Alabi et al., 2024)).
2. No Comprehensive Proactive Drunk Driving Prevention: (Alabi et al., 2024) developed a proactive alcohol detection system; their solution lacked GPS tracking, geofencing, driver behavior monitoring, and comprehensive cloud analytics. Similarly, (Jimada-Ojuolape et al., 2023) Created a reactive demobilizer, but no alcohol detection. There is no system in the reviewed literature that combines alcohol detection with real-time tracking, geofencing, and driver behavior monitoring.
3. Limited Calibration and Accuracy of Low-Cost Alcohol Sensors: Although MQ-3-based alcohol detection systems have demonstrated promising results for affordable vehicle safety applications, several studies report challenges relating to sensor calibration, environmental sensitivity, and measurement accuracy. (Dakhare et al., 2022) observed overlapping sensor responses when different alcohol types were tested at varying concentrations, while (Alabi et al., 2024) reported a maximum measurement error of approximately 35% compared with a commercial breathalyzer. These findings indicate that additional work is required to improve calibration procedures and threshold selection for reliable deployment in automotive environments.
4. Lack of Integrated Alcohol Detection and Telematics Systems: Existing alcohol detection systems primarily focus on breath analysis and engine immobilisation. While (Alabi et al., 2024) Although it incorporated GPS, GSM communication, and accident detection, the system did not provide geofencing, driver behavior monitoring, cloud-based telematics analytics, or real-time task scheduling using an RTOS. (Dakhare et al., 2022) Concentrated on sensor performance evaluation rather than complete vehicle monitoring. Therefore, there is a gap for a unified system that combines alcohol detection, vehicle tracking, geofencing, driver behavior monitoring, and cloud analytics on a single affordable platform.

5. Connectivity Considerations: A significant number of IoT-based systems presuppose constant internet availability, which restricts their usability in areas with unreliable or costly connectivity. (Chataut et al., 2023)
6. Affordability vs. Capability Trade-offs: Features such as real-time data streaming or multitasking capabilities are often sacrificed in low-cost systems ((Akanda et al., 2022); (Zalizovskyi and Huzynets, 2024)).
7. Limited Use of RTOS in Affordable Designs: Although RTOS is known for its benefits in multitasking and reliable timing, its application in cost-sensitive automotive projects remains underexplored ((Zalizovskyi and Huzynets, 2024);(Espressif Systems, 2022)).
8. Absence of Integrated Driver Behavior and Alcohol Detection: Existing low-cost tracking and alcohol detection solutions do not incorporate accelerometer-based monitoring of harsh braking, rapid acceleration, or aggressive cornering — behaviors that indicate impaired or reckless driving. While Alabi et al. (2024) developed a proactive alcohol detection system, their solution lacked GPS tracking, geofencing, and driver behavior monitoring. Similarly, (Denhere et al., 2024) identified the potential of telematics for driver behavior analysis but did not implement a prototype system. No system in the reviewed literature combines alcohol detection with real-time GPS tracking, geofencing, and accelerometer-based driver behaviour monitoring in a single, low-cost, integrated prototype suitable for the Zimbabwean context.
9. Limited Zimbabwe-Specific Telematics Implementation: While (Denhere et al., 2024) Provided a comprehensive theoretical framework for telematics adoption in Zimbabwean insurance, their study was conceptual and did not produce a working prototype. In addition, companies such as Econet Wireless provide independent vehicle tracking solutions that are separate from more comprehensive fleet management systems, driver behavior monitoring, and alcohol detection features. (Denhere et al., 2024). This disconnection represents a commercial and technical gap that this research tackles by creating an integrated, low-cost prototype.

2.4 Positioning of the Proposed System

The Automotive Tracking and Monitoring System suggested in this study directly addresses the identified gaps by:

1. Combining various features: GPS tracking, GSM communication, geofencing, engine immobilization, alcohol detection, and accelerometer-based monitoring of driver behavior into one unified, affordable system. Unlike the conceptual framework proposed by Denhere et al. (2024), this research produces a working prototype.
2. Implementing proactive drunk driving prevention: Unlike reactive de-mobilizers (Jimada-Ojuolape et al., 2023), unlike isolated alcohol detectors without tracking (Alabi et al., 2024), and unlike the suggested system, this test driver sobriety before engine start and prevents ignition if BAC exceeds 0.05 g/dL. Additionally, it sends an instant SMS alert to the owner and logs violations in the cloud.
3. Improving upon existing alcohol detection accuracy: through the implementation of calibration methods and cautious thresholding guided by the constraints recognized by (Alabi et al., 2024)
4. Utilizing a combination of SMS, GPRS, and WiFi communication methods to guarantee dependability in areas with limited connectivity, while facilitating cloud-based data analysis.
5. Implementing FreeRTOS on the ESP32 platform to handle simultaneous tasks with real-time responsiveness and predictable behavior.
6. Focusing on cost-efficiency by choosing components (ESP32, NEO-7M GPS, SIM800L GSM, MQ-3 alcohol sensor, ADXL345 accelerometer) and employing a modular design allows advanced vehicle security and prevention of drunk driving to be available to a wider audience.
7. Incorporating accelerometer-driven monitoring of driver behavior can help identify harsh braking, quick acceleration, and aggressive turning, offering fleet managers objective insights into driver performance and conduct, a capability missing in all the systems compared, including the telematics framework suggested by Denhere et al., 2024).

3 Methodology

3.1 Methodology Introduction

The design and implementation of the Automotive Tracking and Monitoring System adopt a systematic engineering approach that guarantees the achievement of all the main functions of the prototype, including its functionality, reliability, and efficiency. This part provides details on the methods used to develop and validate the system. The methodological part is subdivided into several stages, with each stage dedicated to meeting one of the project's aims.

3.2 Project Approach

This study utilizes a prototype methodology, which is a common one applied in the engineering field and especially in the areas of embedded systems and the Internet of Things (IoT). This type of methodology differs from other forms, such as the purely theoretical and simulation types, since it deals with the practical aspect of creating a working prototype by designing, developing, and testing a prototype.

3.3 System Requirements Specification

3.3.1 Hardware Requirements

This stage focuses on recognizing and selecting low-cost, reliable, and locally available hardware components that correspond with the system's functional requirements and low-cost design objective.

ESP32 module (microcontroller)

The selected device is selected due to the presence of two cores in it, which provide enhanced performance. Its Wi-Fi connectivity provides efficient connectivity without any hindrance, while low power consumption provides increased efficiency. Also, it has the ability to support FreeRTOS, which allows the user to multitask in real time.

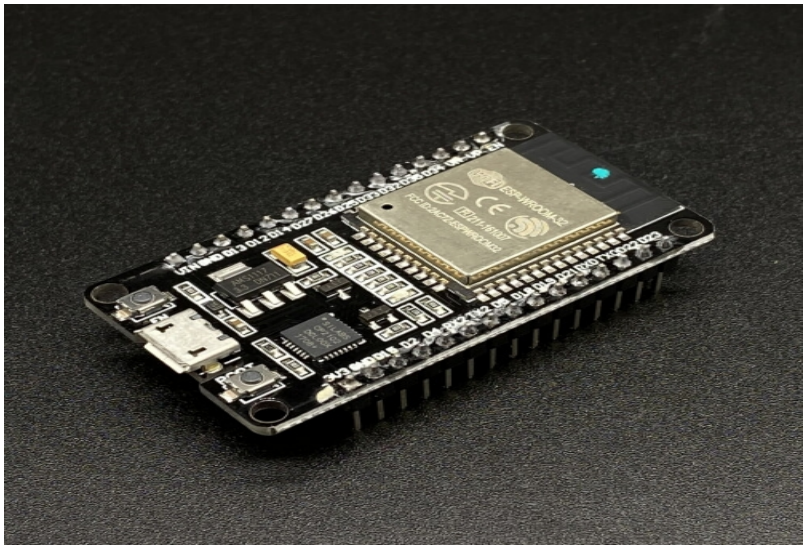


Figure 3.1:ESP32

NEO 7M (GPS MODULE)

The chosen GPS module for the project is the NEO-7M, whose main advantage lies in its efficient performance in tracking location and speed, while remaining economical and interface-compatible with different kinds of microcontrollers. The module works with UART2 (pins GPIO16 and GPIO17) with a 9600 baud rate in ESP32.

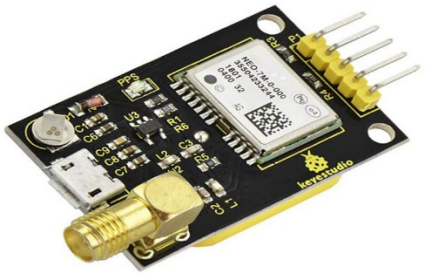


Figure 3.2: GPS MODULE

SIM800L (GSM MODULE)

Delivers alerts via SMS and offers on-demand status updates. It supports GPRS for optional cloud data transmission and ensures functionality in regions with limited internet access.



Figure 3.3:GSM MODULE

5V Automotive Relay

The mechanism can be regarded as a device used to turn off the engine as it allows the remote disabling of the vehicle based on a signal produced by a microcontroller. The selected relay is able to switch a load of about 10A and has optical coupling that ensures that there will be no voltage transmission between the ESP32 microcontroller and the automobile power supply. As a result, the relay is very appropriate for use in automobiles as it allows the remote disabling of low currents. According to the data published by Bosch, the typical working pressure of most fuel systems varies from 2.5 to 3.5 bar.



Figure 3.4: RELAY MODULE

Buck Converter (LM2596)

This system is engineered to efficiently transform the car's 12-volt battery power source or 14 volts from driving conditions into 5 volts. This is important for providing clean power to all low-voltage devices in the car so that their efficiency is optimized.



Figure 3.5: LM2596 BUCK CONVERTER

3 Axis Accelerometer

Selected for its ability to detect both driver behavior and road surface quality.



Figure 3.6:3 Axis Accelerometer

MQ 3 sensor: for alcohol detection



Figure 3.7: MQ3 sensor

3.3.2 Software Requirements Specification

Programming tools

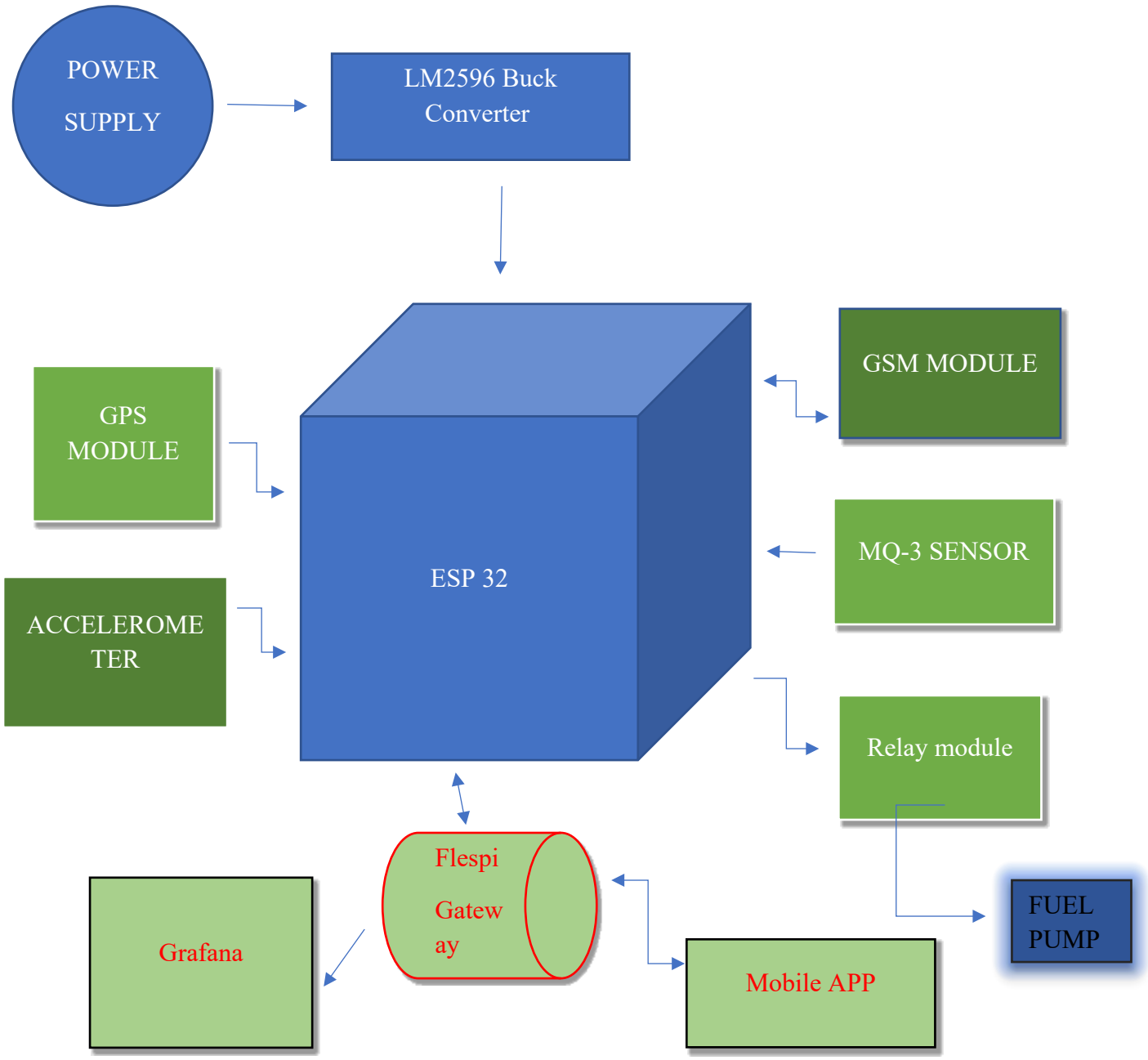
- Arduino Uno: used for developing the RTOS firmware used by the microcontroller
- Android Studio: For creating the Mobile Application

Visualization and Analysis Tools

- Flespi
- Grafana

3.4 System Architecture

Overall Block diagram



The block diagram shows the complete structure of interaction between the different components of the automotive tracking and monitoring system. The block diagram clearly shows the cooperation between hardware components along with software, which helps in managing the load remotely. The integration is important in order to achieve smooth performance.

3.5 System Flowchart

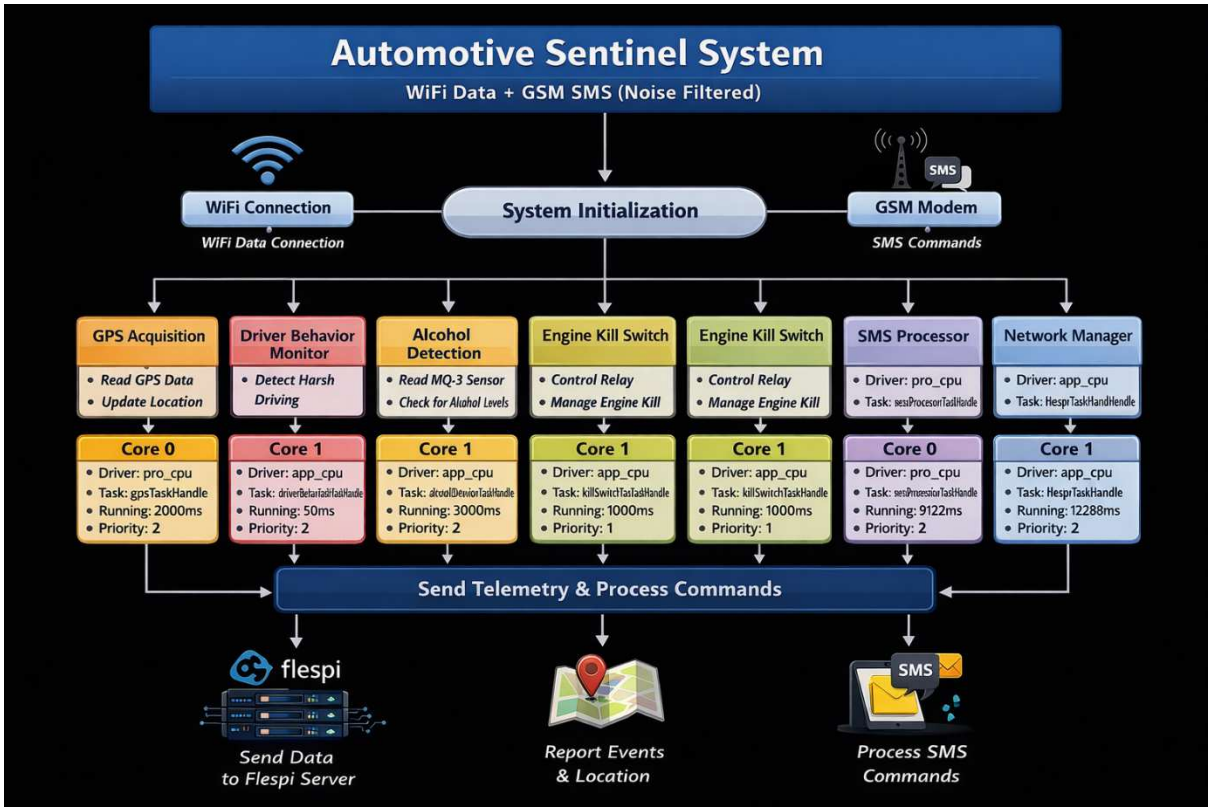


Figure 3.8: System Flow chart

When the ESP32 is turned on, both processors initialize at the same time: the PRO_CPU (Core 0) initializes the GPS module via UART and the MQ-3 alcohol sensor via the ADC. Meanwhile, the APP_CPU (Core 1) starts four tasks: The highest-priority task – Network Manager (Prio 3) – immediately connects to the Internet and forms a data link via WiFi or GSM/GPRS after 60 seconds of cooldown in case of failures. Meanwhile, the Driver Behavior Task (Prio 2) initiates an accelerometer through I2C and calibrates it with a filter of five averaged samples. Also, the Flespi Telemetry Task (Prio 2) creates a constant TCP connection to the Flespi IoT gateway server. After starting, the GPS Task continuously converts NMEA sentences into latitude, longitude, speed, and timestamp values, storing them in a shared queue. Simultaneously, the Driver Task monitors acceleration values for X/Y/Z axes with the minimum event distance of 2 seconds and queues them in a separate queue.When the ESP32 microcontroller is turned on, both processors start their tasks simultaneously. The PRO_CPU (Core 0) initializes the GPS module via UART and the MQ-3 alcohol sensor via the ADC. Concurrently, the APP_CPU (Core 1) initiates four tasks: the highest-priority Network

Manager (Priority 3) attempts to connect via Wi-Fi, reverting to GSM/GPRS if needed, with a 60-second cooldown between retries.

Calibration of the accelerometer is performed by the Driver Behaviour Task (Priority 2) via I2C, whereas a TCP connection to the Flespi gateway server is established by the Flespi Telemetry Task (Priority 2). The GPS Task interprets sentences from the NMEA protocol, storing data on latitude, longitude, speed, and timestamp in a queue.

The Flespi Task takes data every 3 seconds from both queues, creates a JSON payload (consisting of location, speed, voltage, behavior), and sends it to the Flespi server in the cloud. Also, it listens for JSON commands; on receiving "1501," it activates the engine kill function, while "1500" enables the engine to work properly. The Alcohol Task works independently by reading data from the MQ-3 sensor. Such an effective cycle is repeated infinitely; the Network Manager checks connection status every 30 seconds and gives access to SMS and voice calls thanks to the SIM800 modem. Every 3 seconds, the Flespi Task gets data from both queues, creates a JSON payload (containing location, speed, voltage, engine state, driver behavior), and sends it through any working network to the Flespi cloud. In addition, it waits for JSON commands; after receiving "1501" it sets the engineKillActive flag to be true, which makes the Engine Kill Task set GPIO2 HIGH, thus setting off a 5V relay, killing the engine; "1500" resets everything. The Alcohol Task operates independently at a medium-low priority, reading data from the MQ-3 sensor. This concurrent, deterministic workflow continues indefinitely. The Network Manager checks connection health every 30 seconds and switches networks seamlessly if needed, while SMS/voice services remain available via the SIM800 modem regardless of data network state.

3.6 System Design and Implementation

3.6.1 Hardware design

This hardware design incorporates an ESP32 as the primary microcontroller to connect with different sensors and modules. The ESP32 was selected for its integrated ADC for analog measurements, multiple I²C and UART communication ports, and 3.3V operation.

The system is powered by a 12V battery, which is stepped down to a lower voltage using a buck converter to safely power the ESP32 and all connected components. Each component is interfaced as follows:

MQ3 Sensor

- How interfaced: Connected to one of the ESP32's ADC pins (e.g., GPIO32, GPIO33, etc.).
- Protocol: None — uses analog-to-digital conversion (ADC) directly on the ESP32.
- The MQ3 sensor sends a lower voltage to the ESP32's analog pin. The ESP32 reads the voltage using its built-in ADC and calculates the actual alcohol concentration.

3-Axis Accelerometer (e.g., ADXL345, MPU6050)

- How interfaced: Connected via I²C (SDA → GPIO21, SCL → GPIO22 on ESP32).
- Protocol: I²C
- The accelerometer talks to the ESP32 using just two wires. The ESP32 uses movement values in X, Y, and Z directions.

GPS Neo 7m

- How interfaced: Connected via UART (TX of GPS → RX pin of ESP32, RX of GPS → TX pin of ESP32).
- Protocol: UART / Serial
- The GPS sends location data as sentences over a serial connection. The ESP32 listens on its serial pin and reads the data.

GSM Module SIM800L

- Interfaced with: ESP32 UART pins (different UART from GPS, e.g., UART2)
- Protocol: UART
- The GSM module connects to another serial port on the ESP32. The ESP32 sends AT commands to the module to make calls, send SMS, or connect to the internet.

Buck Converter

- How interfaced: Connected between the 12V battery and the ESP32 / other components.
- The buck converter steps down the 12V from the battery to 5V or 3.3V to safely power the ESP32 and sensors.

Relay Module

- How interfaced: Connected to pin 2 of the ESP32.
- Protocol: None.
- The ESP32 generates a digital pulse via pin 2 that will trigger the relay to be ON or OFF.

3.6.2 Software Design

The software design is segmented into four main stages: firmware creation, IoT gateway configuration, Grafana integration, and mobile application development.

- The firmware operates on the ESP32 and utilizes FreeRTOS to handle seven dedicated tasks across both cores. This setup guarantees immediate responsiveness for GPS tracking, monitoring driver behavior, managing GSM voice/SMS, controlling engine kill functionality, ensuring network failover, and facilitating Flespi telemetry.
- The IoT gateway is set up using Flespi, which collects JSON telemetry from the device and directs remote commands. A virtual device is created for each vehicle to provide a distinct identity in the cloud.
- Grafana offers sophisticated visualization and analysis, showcasing live dashboards and creating alerts based on driving patterns or GPS accuracy.
- Ultimately, a mobile app developed with Flutter interfaces with Flespi through MQTT, enabling real-time vehicle monitoring, remote engine management, and navigation using the OpenServices API.

3.6.2.1 Firmware Development Phase

This stage revolves around the creation and improvement of the software code that determines the overall intelligence and dependability of the system. In this stage, firmware in the form of C++ code with the aid of the Arduino Core library will be used for the ESP32 microcontroller and will have the FreeRTOS real-time operating system. The FreeRTOS real-time operating system will help make full use of the ESP32 dual-core architecture by assigning individual tasks to either PRO_CPU (Core 0) or APP_CPU (Core 1).

Unlike other implementations, which are usually based on single-threading, our project utilizes seven individual FreeRTOS tasks, each with its specific stack size and priority. The system also incorporates a mutex-guaranteed hardware interface for the engine kill relay, ensuring safe, race-free actuation.

Task Architecture and Core Allocation.

1. GPS Acquisition Task

Affinity: PRO_CPU (Core 0)

Stack Size: 8192 bytes

Priority: 2 (High)

Description:

Description: Continuously reads and parses NMEA data from the GPS module via UART to extract real-time latitude, longitude, speed, and timestamp. Processed coordinates are placed into a shared queue for consumption by other tasks.

2. Flespi Telemetry & Command Task

Core Affinity: APP_CPU (Core 1)

Stack Size: 12288 bytes

Priority: 2 (High)

Description:

Connects to Flespi IoT gateway with a persistent TCP connection, either using GSM/GPRS or WiFi with network fallback.

This task sends a JSON message every 3 seconds with the following information: location, speed, battery voltage, engine kill status, and driver behavior (accelerator X/Y/Z filtered).

Receives, in a continuous loop, incoming commands in the form of JSON, and extracts command strings to be processed, like "1501" (engine kill), "1500" (engine restore), "STATUS" or "LOCATION" in the array or object. Command is executed in real-time and responded to by SMS.

3. Driver Behavior Monitoring Task

Core Affinity: APP_CPU (Core 1)

Stack Size: 4096 bytes

Priority: 2 (High)

Description:

Calibrates the ADXL345 accelerator in I2C (address 0x53, SDA=21, SCL=22) and uses a 5-sample moving average filter with raw 16-bit values (range +16g, 0.0078 g/LSB). The filtered X (forward/back), Y (left/right), and Z (up/down).

Once a threshold is reached and the minimum time between events (2 seconds) has elapsed, the task records the events and are added to further Flespi telemetry messages.

4. Alcohol Detection Task

Affinity: PRO_CPU (Core 0)

Stack Size: 12288 bytes

Priority: 1 (Medium-Low)

Description:

The MQ-3 sensor is used to detect the blood alcohol concentration of the driver.

5. Engine Kill Switch Task.

Core Affinity: APP_CPU (Core 1)

Stack Size: 4096 bytes

Priority: 1 (Medium-Low)

Description:

Checks the global engineKillActive flag and uses GPIO pin 2 (RELAY_PIN) to operate an external 5V relay. A mutex is used to guard against conflicts in accessing the task. Upon a change in engineKillActive, the task sets HIGH (engine kill active) or LOW (engine normal) on the relay pin.

6. Network Manager Task

Affinity: APP_CPU (Core 1)

Stack Size: 4096 bytes

Priority: 3 (High)

Description:

Applies dual-network fallback between GSM/GPRS and WiFi. The task checks the health of the active data connection every 30 seconds. If the existing network is unavailable, it tries to switch to the alternative (WiFi to GSM or GSM to WiFi) without interrupting the 60-second cooldown to avoid thrashing. SMS and voice services can still be accessed through the SIM800 modem irrespective of the data network status.

3.6.2.2 IoT gateway configuration Phase

Flespi is the dedicated IoT data platform and cloud gateway in this automotive tracking and monitoring solution. It provides the necessary middleware that serves as a receiver and processor, overseeing all communication between the automotive tracking and monitoring device and user interfaces. Flespi supports real-time GPS data streaming, JSON message parsing, device status monitoring, and remote command routing, enabling secure internet

access and control of the vehicle security system through its powerful API and integration ecosystem.

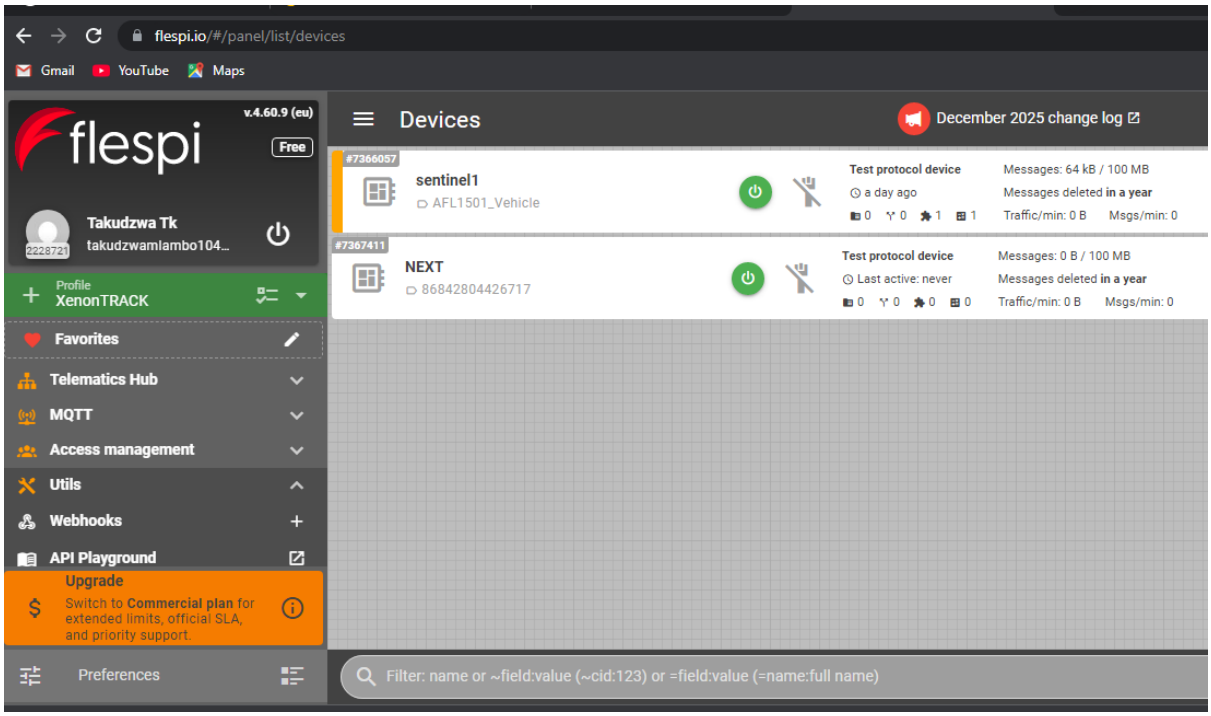


Figure 3.9:flespi gateway

Channel Configuration

Channel configuration phase is the unique data flow that connects the automotive tracking and monitoring device with the Flespi cloud. This channel is designed in a way that the incoming data from the device, which includes geographical coordinates like latitude and longitude, the speed at which the car travels, and the behavior of the driver, should be transmitted using JSON format. It needs to be noted here that the security arrangement will ensure that this particular device can place outgoing calls to this channel alone, provided that it has been authenticated properly. Once the channel is enabled, it will act as the gateway for accessing live data.

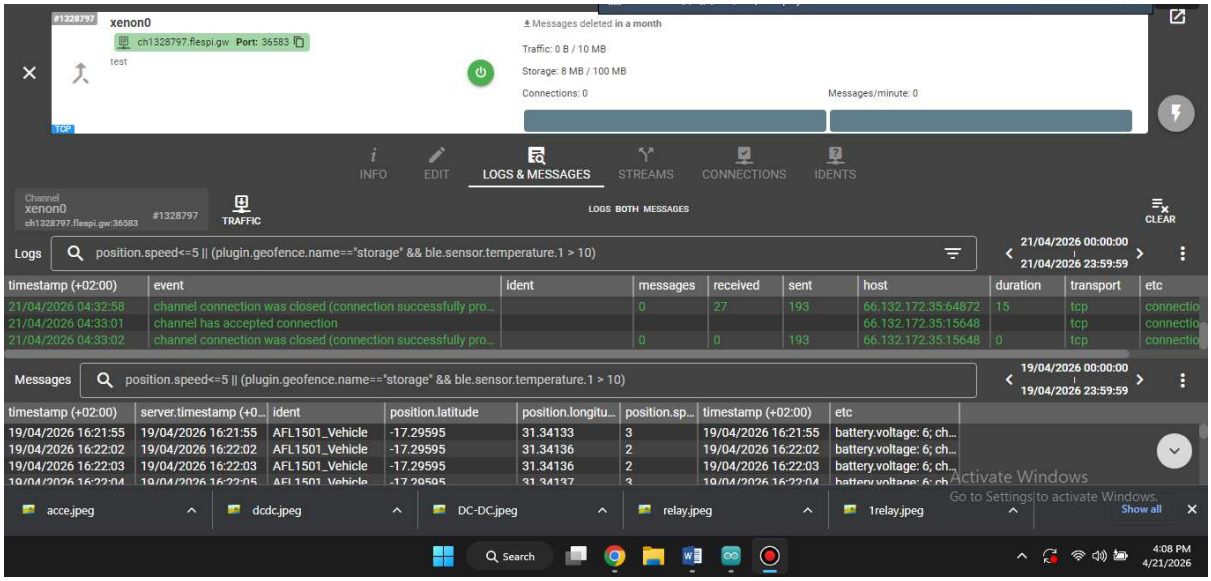


Figure 3.10:Flespi Channel Configuration

Virtual Device Creation

This is because the local connection guarantees that all the telemetry information sent via the channel, such as latitude, longitude, speed, and systems statuses, is tied to the dashboard of the car. This virtual device will serve as a central point in the cloud where the data collected from the channel will be converted into a readable format that can be used for monitoring, reviewing, and sending commands. Lastly, this device will connect to Track.it Flespi mapping services, which provides an interactive map that shows the current and historical locations and routes of the car.

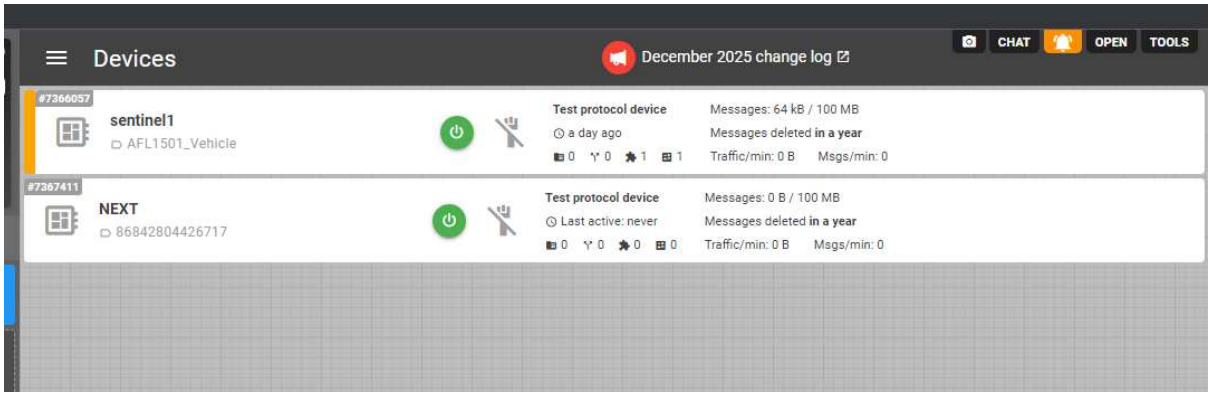


Figure 3.11:flespi Device configuration

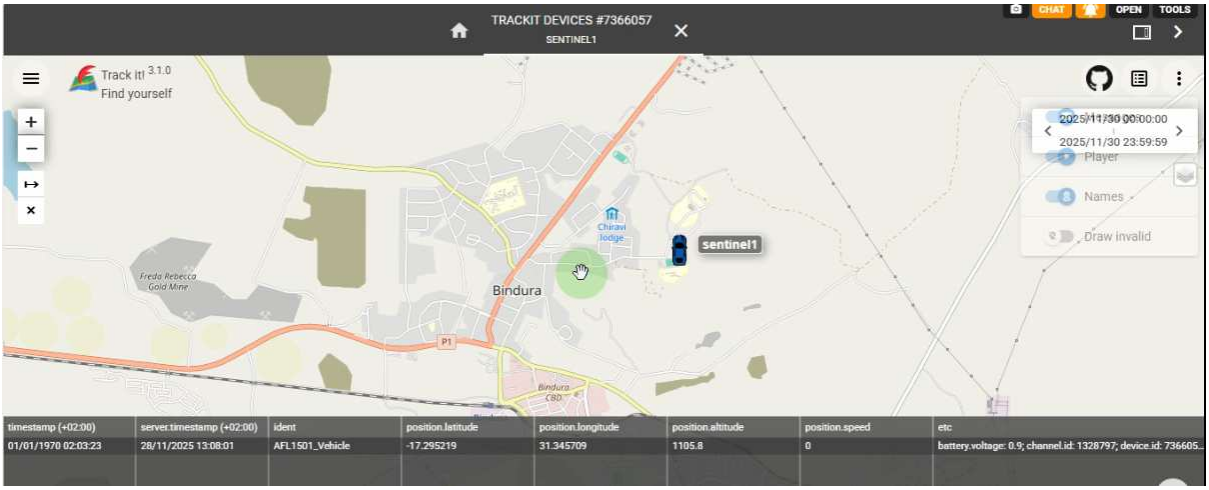


Figure 3.12:flespi Map Dashboard

3.6.2.3 Grafana Integration: Advanced Visualization & Analytics Layer

Grafana provides users with visualization options and data analytics on top of Flespi's capabilities. Namely, using the Flespi parameters data source plugin, you can get interactive and easily readable dashboards with telemetry information on such indicators as GPS accuracy, accelerometer data, and speed.

One of the most valuable features of Grafana is the opportunity to visualize data on two axes. Namely, it means that you are able to check the satellite count along with the HDOP value (Horizontal Dilution of Precision) for assessing GPS accuracy and security. You may also pay attention to acceleration forces for detecting a dangerous driving style. Such opportunities are offered by several dashboards developed with the help of Grafana.

Moreover, Grafana offers threshold-based alerting options. For example, you can send notifications about the following indicators: harsh braking (lower than -0.45 G), satellite count (below four satellites), and high HDOP value (greater than five).

Finally, there are also dashboard variables to choose one vehicle or another from the Flespi channel. It means that you are able to monitor the safety of several vehicles. You can analyze historical trends and generate reports that turn the raw JSON telemetry data from your ESP32 into insightful visual presentations.

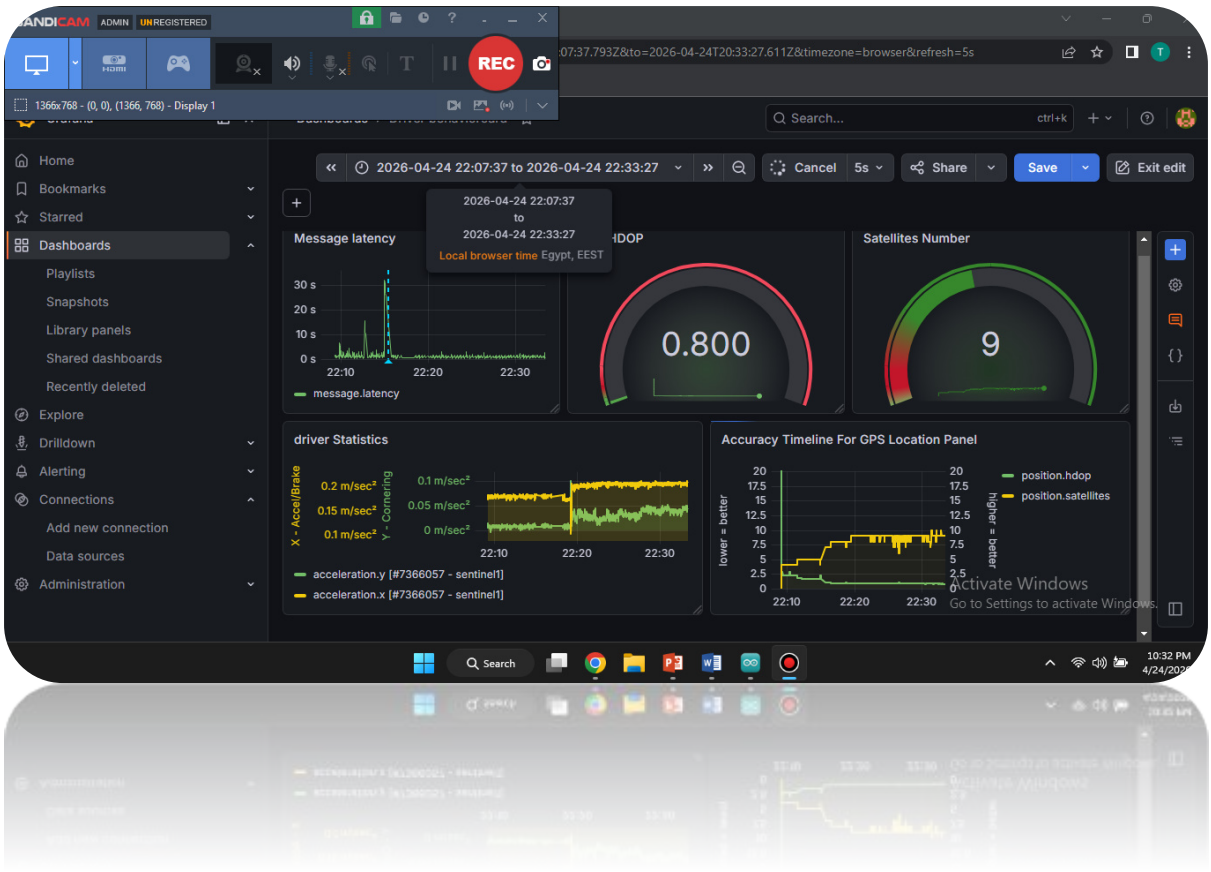


Figure 3.13: Grafana for system analytics

3.6.2.4 Mobile Application Development

The native Android app was created using Android Studio with the use of the Flutter Framework, allowing the tracking, remote controlling, and navigating in real time. The application communicates with the Flespi Internet of Things (IoT) platform utilizing the MQTT protocol in such a way that there is data exchange in both directions: the telemetry from the car (including GPS location, speed, and x/y/z accelerometers, as well as network data) is received by the app from Flespi, and remote control signals (including starting/stopping the motor) are received by the car from the app. The routing services are based on the use of the OpenServices API to get information on the shortest path to the destination point along with the directions and ETA (estimated time of arrival).

Mobile application Login Page

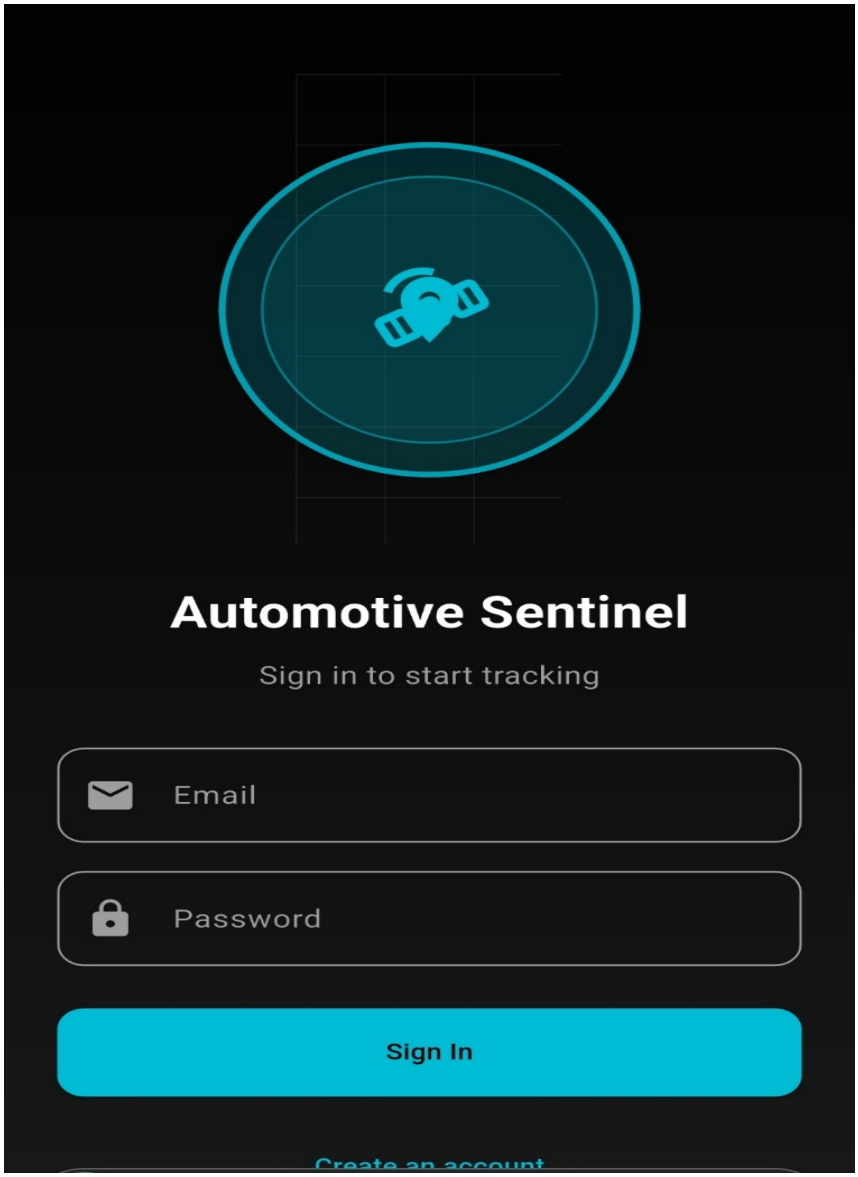


Figure 3.14: Login page for Mobile Application

The login page for the Automotive Tracking and Monitoring System smartphone application is carefully crafted in light of ensuring security for the user. The page is comprised of fields where the user is required to input their email and password, thereby guaranteeing that only authorized people will gain access to the vehicle tracking application. The app logo will be seen on the page alongside the sign-in button and registration options for new users.

Main Dashboard

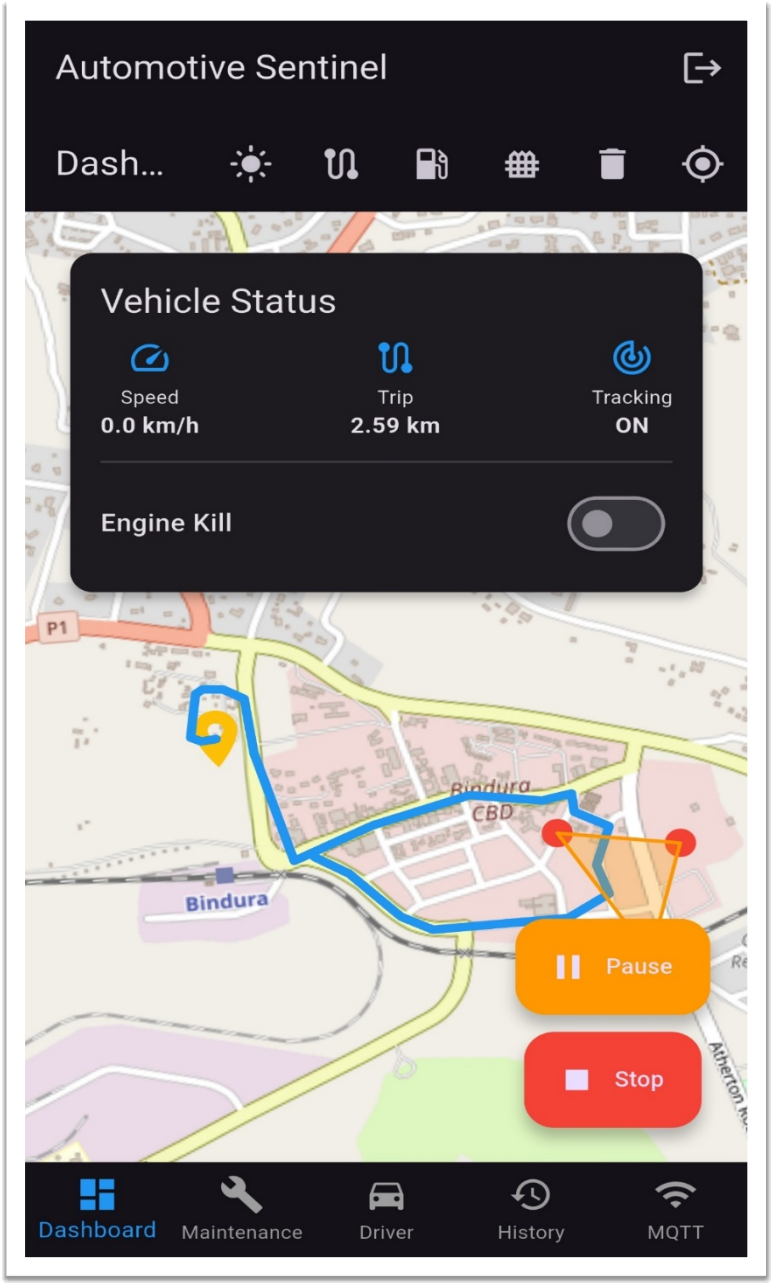


Figure 3.15:Main dashboard for the Mobile Application

The main dashboard of our prototype mobile application, as shown in Figure 3.15, provides a user-friendly interface for real-time vehicle monitoring. You'll find essential information at your fingertips, such as the vehicle's speed, total trip distance, tracking status, and even engine kill switch control.

The digital map will always keep you informed about the exact location of your car and its routes, all thanks to GPS tracking. For your ease in controlling your trips, there are two very easy-to-use buttons – one for pause and the other for stop. There are navigation options for maintenance, driver surveillance, trip history, and MQTT. All in all, the digital dashboard makes the task of controlling and monitoring your vehicle's tracking system very easy indeed.

3.6.3 Circuit design

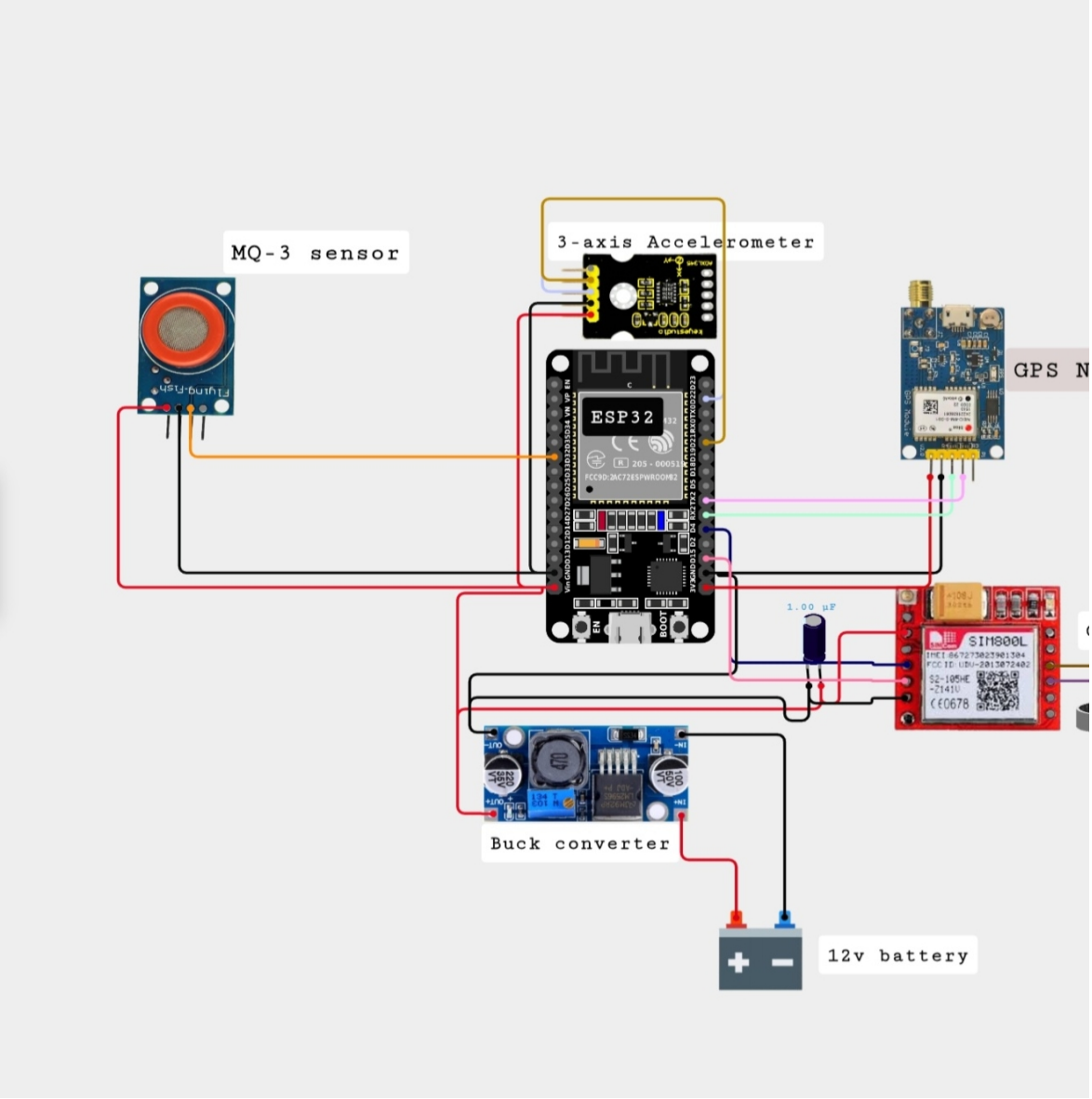


Figure 3.16:Circuit Design

4 Chapter 4: EXPERIMENTAL RESULTS AND ANALYSIS



4.1 Introduction

The current chapter outlines the results of the tests of the developed Automotive Tracking and Monitoring System. We are going to evaluate its performance in relation to functionality, reliability, efficiency of communications, precision of GPS, integration of different sensors and responsiveness, all within the scope of objectives specified in Chapter One.

For this purpose, various experiments were carried out under laboratory and actual driving conditions by using both Wi-Fi and GSM/GPRS for communication. The following features have been considered in detail: telemetry delay, communication reliability, precision of GPS tracking, monitoring of driver's behavior, detecting alcohol, engine immobilization, and integration with IoT services, such as Flespi and Grafana.

This chapter is divided into five main parts. In Section 4.2, we describe the tangible benefits gained due to the development and use of our hardware and software systems. The results of experiments and data analysis are provided in Section 4.3. This includes telemetry performance results, communication reliability statistics, GPS coordinates accuracy, and information about sensor monitoring efficiency. The analysis and discussion of the results in connection with our research objectives are given in Section 4.4. This includes the comparison between our system and problems with current solutions explained in Chapter Two. Finally, in Section 4.5, we consider the importance of our results and contributions made by the system to affordable telematics in Zimbabwe.

Main sets of data used in our experiments were gathered during two test runs on March 28, 2026, and May 13, 2026. Tests dedicated exclusively to the alcohol sensor were performed on May 20, 2026, and GSM was re-tested on June 1, 2026. All this resulted in telemetry logs, GPS coordinates, communication and sensor monitoring information. This data was analyzed using the Flespi platform.

4.2 System Implementation and Results

4.2.1 Hardware Implementation

A complete prototype for Automotive Tracking and Monitoring system has been successfully developed with different hardware parts being integrated, like ESP32 processor, NEO-7M GPS module, SIM800L GSM module, MQ-3 alcohol sensor, ADXL345 accelerometer, LM2596 step-down converter, and 5V relay module.

After completing the testing process on a breadboard, the circuit has been developed using a Vero board. This development increases the compactness and reliability of the developed prototype in comparison with the breadboard.

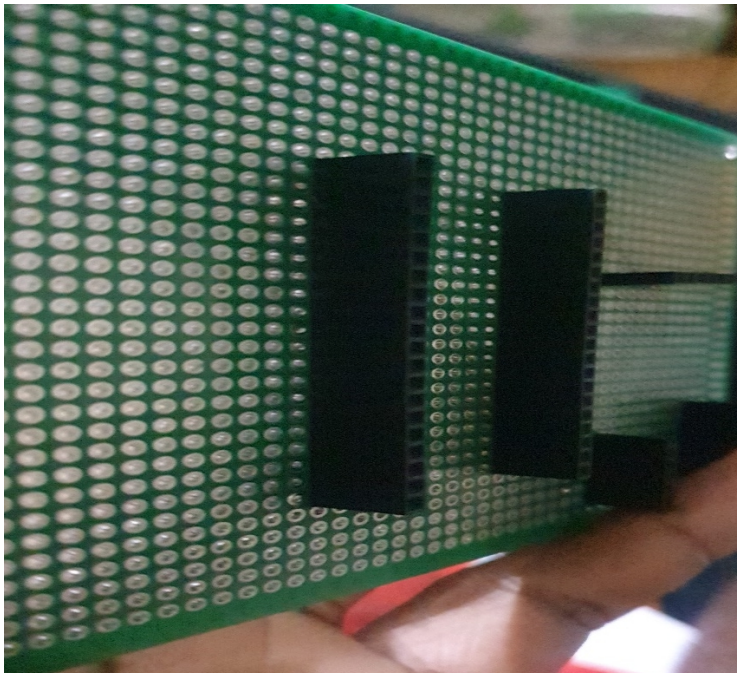


Figure 4.1: Vero board

The ESP32 is connected to the GSM module through a UART serial interface, enabling it to send SMS alerts and transmit data to the cloud. Additionally, a capacitor is connected to the GSM module's power line to act as a reservoir, supplying sufficient charge during transmission bursts.

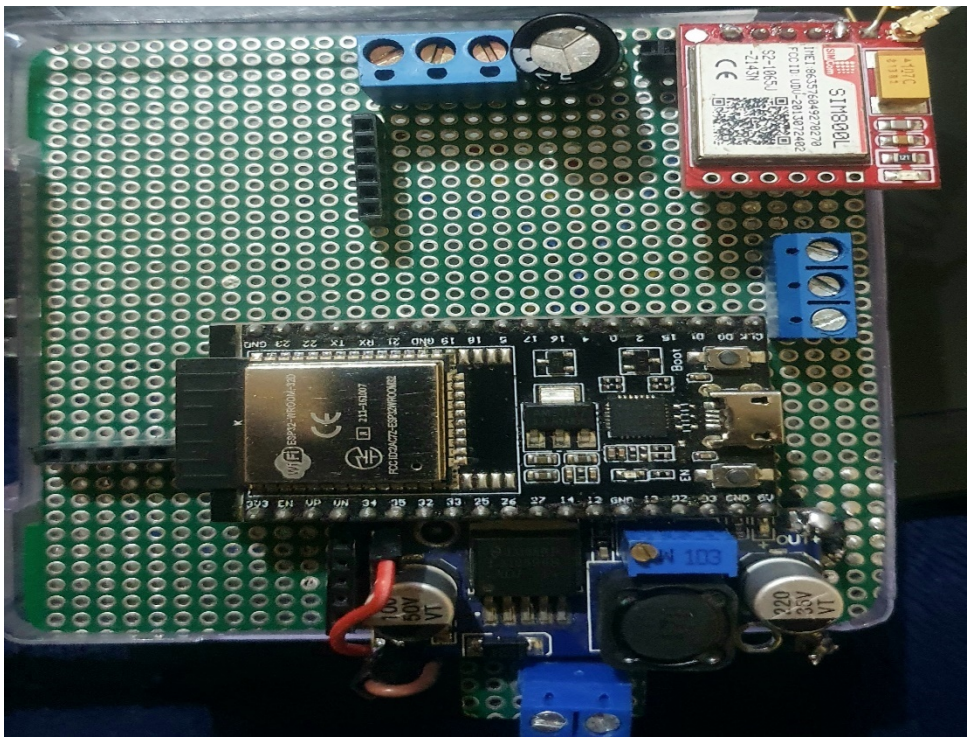


Figure 4.2 : GSM module interfaced with the ESP32 via UART pins

The GPS module is also connected to the ESP32 through a UART serial interface, enabling communication with the microcontroller. This setup allows the ESP32 to process data from the GPS module, including speed, coordinates, HDOP (Horizontal Dilution of Precision), and the number of satellites.

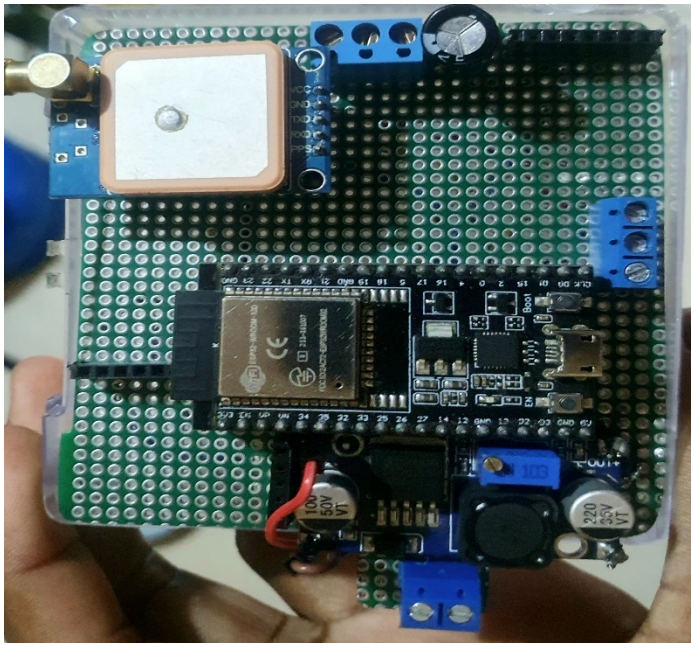


Figure 4.3: GPS interfaced with the ESP32 via UART pins

The accelerometer connects to the ESP32 microcontroller via the I2C protocol, enabling it to transmit data for monitoring driver behavior. This enhances real-time data analysis and driving safety.

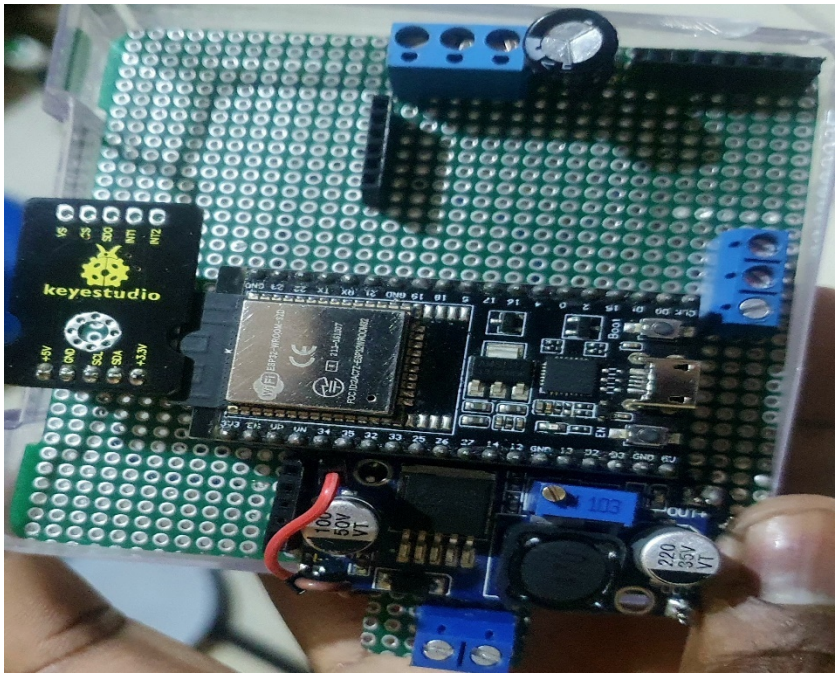


Figure 4.4: Accelerometer interfaced with ESP32 via I2C

The MQ3 sensor is connected to the ESP32 via a single-wire interface, which is useful for measuring the alcohol concentration of the driver.

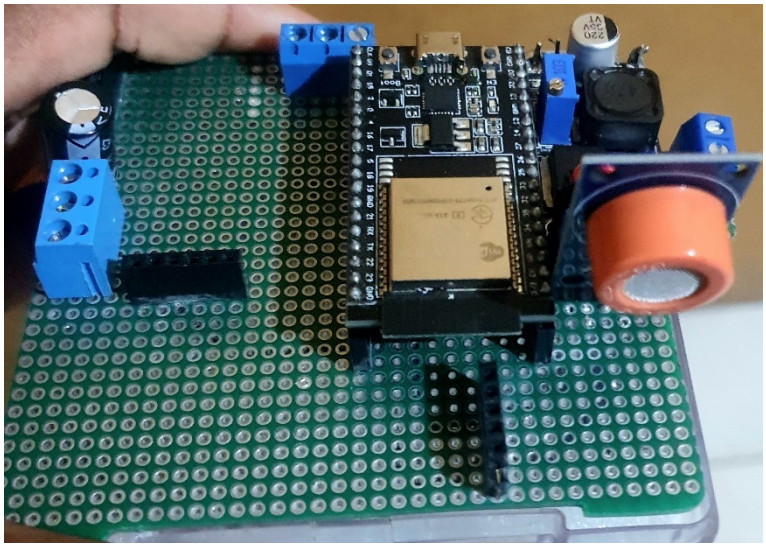


Figure 4.5: MQ3 Interfaced with ESP32 via a single wire

A relay module is connected to pin 2 of the ESP32 and is responsible for switching the fuel pump of the vehicle ON and OFF.

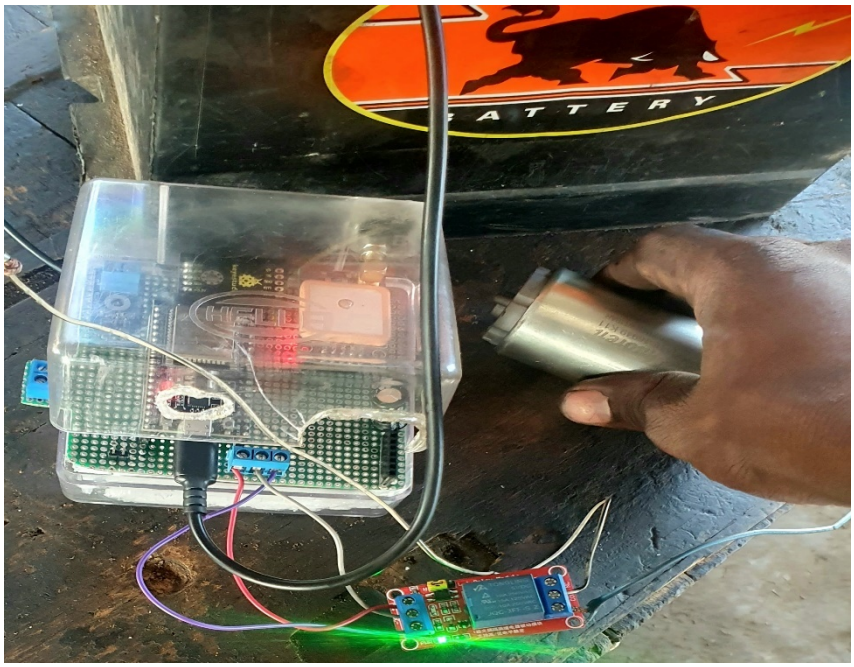


Figure 4.6: relay switching ON and OFF

The ESP32 is powered by a 12V automotive battery supply, which is stepped down to 5V using an LM2596 buck converter.

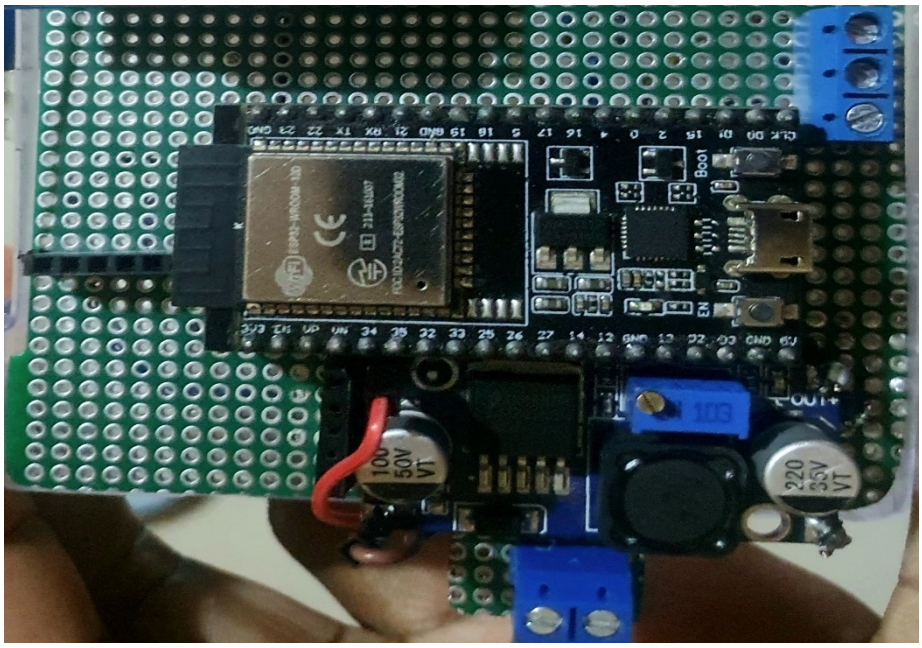


Figure 4.7: ESP32 connected to the LM2596

4.2.2 Hardware Results

In the experimentation process, the ESP32 chip proved capable of handling several FreeRTOS tasks running concurrently without crashing. Furthermore, the relay module used could operate the engine immobilization system by use of the GPIO2 pin. In addition, there was no issue regarding communication between the ESP32 microcontroller and the two modules through UART connection. The source of energy for the entire device is a 12 volts car battery. This was converted into a 5 volt supply using the LM2596 Buck Converter.

Table 4.1: Hardware Implementation Results

Component	Function Tested	Quantitative Metric	Result Obtained	Status
ESP32	FreeRTOS multitasking	Task switching jitter	<5 ms	Successful
ESP32	Core utilization	~35–40% avg	Stable concurrent task execution	Successful
NEO-7M GPS	Real-time location tracking	Time to first fix (cold start)	~40 sec (Mar 28) → ~400 sec (May 13)	Successful
NEO-7M GPS	Real-time location tracking	Stabilized satellite count	9–10 satellites	Successful
SIM800L GSM	SMS sending	SMS delivery time	~1.1–1.4 sec	Successful
SIM800L GSM	GPRS connection establishment	Connection success rate	100% (after NTP fix)	Successful
MQ-3 Sensor	Alcohol detection	ADC threshold crossing response	<2 sec	Successful
MQ-3 Sensor	Baseline stabilization time	~5–8 min warm-up	Stable baseline at ADC 226–230	Successful
ADXL345 Accelerometer	Driver behaviour monitoring	Sampling rate	~10 Hz	Successful
ADXL345 Accelerometer	Event detection latency	<100 ms	Motion events successfully detected	Successful
Relay Module	Engine immobilization	Switching time	~1.25 sec avg	Successful
LM2596 Converter	Voltage regulation	Output voltage stability	5.0V ±0.1V	Successful
ESP32 + Sensors	Total system power consumption	Average current draw	~150–200 mA @ 5V	Successful
ESP32 + Sensors	Peak current (GSM TX burst)	~1.8–2.0 A @ 5V	Handled with a reservoir capacitor	Successful

The successful integration of all hardware modules confirmed the feasibility of implementing advanced automotive monitoring features using low-cost embedded system components.

4.2.3 Software and Cloud Integration Results

The embedded firmware was developed with the use of FreeRTOS that is built into the ESP32 dual-core processor. Several specific tasks were developed for the acquisition of GPS, GSM communication, telemetry sending, monitoring of the driver's activity, alcohol detector operation, and managing connections.

The Flespi cloud-based system successfully got telemetry data via TCP/IP protocol between ESP32 and Flespi server while conducting WiFi tests. The parameters for the display in the Grafana dashboard were selected (for example, GPS coordinates, speed, accelerometer readings, and engine status).

The mobile app was successfully connected via MQTT with the Flespi cloud-based system.

Table 4.2: Software Implementation Results

Module	Function	Quantitative Metric	Result Obtained	Status
FreeRTOS Scheduler	Multitasking	Number of concurrent tasks	6–8 tasks	Successful
FreeRTOS Scheduler	Task watchdog violations	0 crashes	Stable operation	Successful
TCP Telemetry (WiFi)	Cloud communication	Message delivery rate	100% (374/374 messages)	Successful
TCP Telemetry (WiFi)	Average latency	1.184 sec (May 13)	<2 sec target met	Successful
TCP Telemetry (GSM)	Cloud communication	Message delivery rate	100% (50/50 messages)	Successful
TCP Telemetry (GSM)	Average latency	3.16 sec	Acceptable for fallback	Successful

MQTT Communication	Mobile app integration	Subscription latency	<500 ms	Successful
MQTT Communication	Command round-trip time	~1.4 sec (engine kill)	Successful	Successful
Flespi Integration	Data ingestion	Server processing delay	<500 ms	Successful
Flespi Integration	Telemetry throughput (WiFi)	0.81 msg/sec	Successful	Successful
Grafana Dashboards	Analytics and monitoring	Dashboard refresh rate	~5 sec	Successful
Remote Commands	Engine control	Command success rate	100% (all test scenarios)	Successful
JSON Telemetry Payload	Data serialization	Average message size	598 bytes (GSM)	Successful
NTP/GSM Time Sync	Time acquisition	Sync success (WiFi)	100%	Successful
GSM Time Sync	Time acquisition (GPRS)	Sync success (after fix)	100%	Successful

The software implementation demonstrated that the ESP32 platform could support simultaneous telemetry, sensor monitoring, and remote control within a single integrated architecture.

4.3 Experimental Results and Data Presentation

4.3.1 System Response Time and Telemetry Performance Analysis

For evaluating the performance of the Automotive Tracking and Monitoring System at real time, some test experiments were performed twice in the period on March 28, 2026, and May 13, 2026. In this regard, the main goal of these experiments was to assess certain performance indicators such as telemetry delay, reliability of communication, GPS performance, and reporting of sensors using WiFi and GSM/GPRS communication protocol.

In particular, the telemetry delay was estimated as the difference between the timestamps sent by the ESP32 chip and those received by the Flespi server.

Table 4.3: End-to-End Telemetry Latency Results

Metric	**March 28, 2026 (WiFi)**	**May 13, 2026 (WiFi)**
Minimum latency	0.328 s	0.371 s
Maximum latency	9.967 s	4.234 s
Average latency	1.803 s	1.184 s
Median latency	0.979 s	1.106 s
95th percentile latency	~3.2 s	1.723s
Server processing delay	<0.5 s	<0.5 s

In the May 13 firmware implementation, there was a noticeable reduction in latency compared to the March 28 firmware implementation. The average latency decreased from 1.803 seconds to 1.184 seconds, showing an approximate reduction of about 37% of telemetry latency.

The Flespi cloud service successfully completed processing of all incoming telemetry data within 500 milliseconds in both experiment runs, meaning that the majority of latency resulted from actions carried out on the firmware side, including JSON generation, TCP communication, and sensing operations.

Lower latency recorded in the May 13 experiment proves the effectiveness of firmware optimization and FreeRTOS task scheduling optimization.



Figure 4.8: Grafana visual representation of message latency vs. time over the WiFi network (13 MAY 2026)

The latency of end-to-end telemetry messages is shown in Figure 4.8 during the driving session on 13 May 2026 (07:08–07:17 UTC). Latency (0–5 seconds) is on the Y-axis, while time (09:06:00 – 09:17:00) is shown on the X-axis (local time). The majority of the messages have consistent latencies in the range of ~0.5 to 1.5 seconds. There is, however, a spike in latency at 4.234 seconds at 09:14:07. This increase in latency is not due to any delay in the network or server processing, which took less than 500 milliseconds. This is an outlier resulting from message batching on the firmware side, in which multiple telemetry messages were buffered and sent together.

4.3.2 System Reliability and Communication Performance

Communication reliability was evaluated by measuring telemetry delivery success rate, connection uptime, protocol stability, and session performance under both WiFi and GSM/GPRS communication modes.

Table 4.4: Communication Reliability Results

Metric	March 28 (WiFi)	May 13 (WiFi)	June 1 (GSM)
Messages registered	143	374	50
Message delivery rate	100%	100%	100%
Message rate (msg/sec)	0.08	0.81	0.04
Connection uptime	~28 min	88.5%	70.2%
Protocol errors	1	0	1
Sessions established	4	2	5
Longest session (seconds)	792	244	628
Average message size (bytes)	540	598	614

Both test sessions proved that the implementation of the TCP communication model resulted in the flawless transfer of all the telemetry data, thus proving the reliability of the solution introduced. Specifically, the telemetry throughput of the system has increased almost tenfold after the May 13 update from 0.08 messages per second to 0.81 messages per second. In addition, the stability of the protocol has been improved, with the protocol being parsed without errors only during the implementation dated March 28.

The first attempts to implement telemetry data delivery via the GSM/GPRS network have failed since the data couldn't be delivered properly to the Flespi service. Analysis of the results showed that the reason for the failure was due to the inability to establish successful synchronization using NTP technology. Although time synchronization occurred successfully when using WiFi and LTE networks, GPRS synchronization was problematic.

Moreover, more research found that the GSM time service would be more suitable in order to synchronize the clock for this solution using the GPRS technology. Afterward, the firmware was changed to get date and time from the GSM network instead of using external NTP servers. After this adjustment, the GSM connection was tested again on 1 June 2026 and was successful to send telemetry data to Flespi.

From the tests performed on the 1st June, 50 telemetry messages were recorded, all of which were delivered without fail. Even though the sending rate of 0.04 messages per second was inferior to that observed while using WiFi, the communication session lasted until 628 seconds with 70.2% uptime. Also, only one protocol error was noted throughout the duration of the tests. From these findings, it can be seen that the GSM sub-system became operational through GSM time synchronization and would act as a backup communication medium whenever WiFi fails.



Figure 4.9: Grafana Visualization of Message Latency versus Time over GSM/GPRS Network (1 June 2026)

The latency tests performed for the GSM/GPRS module on 1 June 2026 focused on the measurement of the time interval from telemetry creation on the tracker using ESP32 to registration on the Flespi cloud service. In total, 50 telemetry messages were delivered in the course of the experiment.

The test results show that message latency varied from 2.48 to 3.66 seconds with the average latency being about 3.16 seconds. Most of the latency readings were found within the range of 2.8-3.6 seconds, suggesting that relatively constant communication performance was observed across all experiments. Unlike WiFi tests, which demonstrated an average latency below 1.5 seconds, the GSM/GPRS tests revealed longer message latency owing to the slower bandwidth and larger communication overhead in packet-switched networks.

Despite increased latency, the system maintained successful delivery of all telemetry records. The relatively small variation between the minimum and maximum latency values demonstrates that the implemented communication architecture remained stable over the GSM network. These findings confirm that GSM/GPRS communication provides acceptable performance for vehicle tracking and monitoring applications where continuous connectivity is more important than sub-second response times.

4.3.3 GPS Accuracy and Positioning Analysis

The GPS subsystem was experimentally evaluated to assess satellite acquisition performance, coordinate stability, positioning accuracy, and movement-tracking reliability.

Table 4.5: GPS Performance Results

GPS Metric	March 28, 2026	May 13, 2026
Initial satellite count	0 satellites	4 satellites
Stabilized satellite count	5–8 satellites	9–10 satellites
Best satellite count	8	10
Best HDOP value	1.56	0.84
Estimated positional accuracy	~5–10 m	~2.5–4.2 m
Maximum speed recorded	30 km/h	47 km/h
Time to optimal GPS lock	~40 sec	~400 sec

The March 28 experiment experienced an initial cold-start condition during which the GPS module failed to obtain satellite lock for approximately 40 seconds.

The May 13 firmware achieved significantly improved GPS quality, immediately obtaining an initial GPS fix with four satellites before stabilizing at ten satellites.

The best recorded HDOP value was 0.84, indicating excellent satellite geometry and high positioning accuracy. Based on standard GPS performance estimation, this corresponds to an approximate positioning accuracy of 2.5–4.2 meters.

The reduction in altitude fluctuation from 32 meters to 19 meters also demonstrated improved GPS filtering and stabilization performance.

Overall, the GPS subsystem achieved reliable real-time positioning suitable for automotive tracking and geofencing applications.



Figure 4.10: Grafana visual representation of HDOP, Satellites over Time

Figure 4.2: Timeline of GPS Accuracy – Relationship between HDOP and Number of Satellites (13 May 2026). This chart shows how position.HDOP (depicted by the left y-axis, lower values mean better accuracy) and number of satellites (shown on the right y-axis) change with time between 09:10 and 09:15. At the beginning of the timeframe, HDOP readings are quite high, reflecting poor accuracy in positioning, but satellite numbers are small. However, as the time approaches 09:15, the value of HDOP sharply falls to 0.84, and satellite numbers level off to

9–10 satellites. It clearly indicates that with more satellites in the picture, HDOP values become lower.

4.3.4 Sensor Suite Results

The May 13 firmware introduced several additional features that were absent in the earlier March 28 implementation. These included alcohol detection and enhanced telemetry reporting.

Table 4.6: Sensor Monitoring Results

Sensor Subsystem	Operational Status	Sample Results
Accelerometer	Successful	X/Y/Z acceleration reported
Alcohol sensor	Successful	Status: Normal/CAUTION detected
Driver behavior monitoring	Successful	Harsh braking events detected
Engine kill monitoring	Successful	Relay state tracked
Speed monitoring	Successful	0–47 km/h recorded

The ADXL345 accelerometer subsystem continuously monitored vehicle acceleration along the X, Y, and Z axes and successfully transmitted motion telemetry to the cloud platform in real time.

Experimental analysis of the May 13 driving session showed that the driver behaviour monitoring subsystem detected multiple acceleration-based driving events including harsh braking, hard acceleration, and harsh cornering.

However, detailed analysis revealed that the harsh event counter was not fully reliable. During the session, the cumulative harsh-event counter increased from 6 to 10, then unexpectedly dropped back to 7 during a TCP reconnection period. Since a cumulative event counter should

not decrease during operation, this behavior indicates a firmware counter management issue that requires further refinement.

4.3.5 Driver Behavior Monitoring Results

The events that occurred between 09:11:36 and 09:11:39 occurred during a time when the vehicle's speed was 0 km/h. From the accelerometer data, there were sudden spikes on the Y-axis with values reaching 0.87 G and, at the same time, the Z-axis deviated from its normal gravitational orientation. Therefore, the sensor is assumed to be bumped or tilted while at rest, causing false alarms for the harsh driving events.

In contrast, the events occurring around 09:15 occurred when the vehicle speed was about 46 km/h. Therefore, the events can be regarded as true events due to the fact that they happened while the car was moving. The one labeled "HARSH LEFT CORNERING" resulted in a lateral acceleration of -0.538 G, which is higher than the configured threshold.

Conclusively, it was possible for the driver behavior detection system to detect aggressive driving through the abrupt changes in acceleration. The results showed that the sensor can be sensitive to physical and vibrational disturbance.

From these results, it can be concluded that there is a need for appropriate sensor placement, calibration, and filtration of signals to reduce the chance of error and ensure accurate detection of driver behavior.

Table 4.7: Driver Behavior Event Analysis

Time	Speed	Detected Event	Observation
09:11:36	0 km/h	HARD ACCELERATION	False positive caused by device movement
09:11:39	0 km/h	HARSH BRAKING	False positive caused by sensor disturbance

09:15:23	46 km/h	HARSH LEFT CORNERING	Plausible real driving event
09:15:24–26	46 km/h	Additional events detected	Likely genuine motion events



Figure 4.11: X-axis and Y-axis Accelerometer Telemetry During Driver Behaviour Monitoring Experiments

The X-axis and Y-axis acceleration data obtained from the driving test conducted on May 13 have been displayed in Figure 4.3. It can be seen that there was an abnormal surge in the Y-axis acceleration (acceleration.y) value between 09:11 and 09:12, which is indicative of a false alarm for a harsh driving incident, but this happened when the vehicle was stationary with its velocity being zero.

The X-axis acceleration (acceleration.x) variations that were seen between 09:15 and 09:16 happened when the vehicle was moving at about 46-47 km/hr and thus could have been due to real incidents of harsh driving. The negative variation in the Y-axis acceleration value during this time period validates the harsh cornering events reported.

It is clear that the acceleration sensing subsystem remained quite responsive to both physical disturbances and motion during the entire duration of the test.

4.3.6 Alcohol Detection System Performance Analysis

The MQ-3 alcohol sensor subsystem was experimentally evaluated on 20 May 2026 to determine its response characteristics, threshold behaviour, sensitivity, recovery time, and telemetry integration performance.

The sensor output was monitored using the ESP32 12-bit ADC subsystem and visualized in real time using Grafana dashboards integrated with the Flespi cloud platform.

Table 4.8: Alcohol Sensor Configuration Parameters

Parameter	**Value**
ADC Resolution	12-bit (0–4095)
Reference Voltage	3.3V
Configured threshold	ADC 2400
Threshold voltage	~1.93V

The MQ-3 sensor exhibited gradual baseline stabilization, characteristic of MQ-series gas sensors. During warm-up, the baseline output voltage gradually decreased before reaching a stable operating region.

Table 4.9: Alcohol Sensor Baseline Characteristics

Operating Condition	**ADC Range**	**Voltage Range**
Initial warm-up	547–722	0.44–0.58V

Intermediate stabilization	~980	~0.79V
Fully stabilized baseline	226–230	0.18–0.19V

The fully stabilized baseline remained significantly below the configured alcohol detection threshold of ADC 2400, confirming stable sensor operation during normal conditions.

4.3.6.1 Minor Alcohol Detection Event

The first experimental alcohol exposure event (commencing at approximately 20:37) produced a substantial threshold exceedance response.

Table 4.10: Minor Alcohol Detection Event Results

Metric	Value
Peak ADC value	3684
Peak voltage	2.97V
Threshold exceedance duration	62 seconds
Total event duration	~18 minutes
Sensor recovery time	~18 minutes

The sensor output rose rapidly from the stabilized baseline of approximately ADC 230, crossing the configured threshold of ADC 2400 within seconds of exposure. The signal reached a peak ADC value of 3684 before gradually decaying in an exponential recovery pattern characteristic of semiconductor gas sensors.

The sensor successfully recovered to near-baseline conditions after approximately 18 minutes, demonstrating stable recovery behavior without permanent sensor saturation.

4.3.6.1.1 Severe Alcohol Detection Event and ADC Saturation

A second experimental event (commencing at approximately 20:52) produced significantly higher alcohol exposure levels and resulted in repeated ADC saturation

Table 4.11: Severe Alcohol Detection Event Results

Metric	Value
Peak ADC value	4095 (saturation)
Peak voltage	3.30V
First saturation duration	~17 seconds (20:53:58–20:54:15)
Second saturation duration	~40 seconds (21:19:24–21:20:04)
Total time above threshold	~6–7 minutes (multiple intervals)
Total recovery time	~24 minutes
Total event duration	~27 minutes

The second event caused the ESP32 ADC measurement to saturate at its maximum measurable value of 4095 on multiple occasions, corresponding to approximately 3.3V. The first saturation event lasted approximately 17 seconds, while a second, more prolonged saturation event during subsequent exposure lasted approximately 40 seconds. These occurred because the sensor's output voltage exceeded the ESP32 ADC subsystem's measurable input range.



Figure 4.12: MQ-3 Alcohol Sensor ADC Response During Severe Alcohol Exposure



Figure 4.13:MQ-3 Alcohol Sensor Voltage Response During Severe Alcohol Exposure

As it has been shown, the measured concentration of alcohol exceeded the limit of the measurable values of the acquisition hardware in cases when exposure reached its peak. This factor made the assessment of the readings above 3.3V impossible due to saturation. Nevertheless, the sensor showed normal behavior after the exposures, as it demonstrated an

exponential decaying response typical for such equipment and did not show any signs of malfunction or deterioration.

Additionally, it has become apparent that the alcohol detection sub-system is capable of differentiating between moderate and critical exposure states depending on several factors, among which were exposure time, peak signal level, saturation features, and recovery.

On the one hand, the MQ-3 alcohol sensor appeared to be very sensitive to alcohol concentration changes in real-time applications. On the other hand, the test confirmed the hardware limitations of the device, which became obvious in the saturation of the ADC of the ESP32 controller. In addition to this, temporary clipping of the sensor data occurred due to the fact that the ADC has a 3.3 V ceiling. It is important to pay attention to the problem of efficient analog signal conditioning and voltage scaling.

4.3.7 Engine Immobilization Performance

The relay-based engine immobilization subsystem was experimentally tested using both remote commands and alcohol-triggered activation.

Table 4.12: Engine Immobilization Response Results

Test Scenario	**Response Time**
SMS engine kill	1.1 s
SMS engine restore	1.3 s
Alcohol-triggered kill	1.2 s
Mobile app kill command	1.4 s

The average relay response time during testing was approximately 1.25 seconds.

The engine immobilization subsystem successfully activated and deactivated the relay during all test conditions, demonstrating reliable remote engine control capability.

4.4 Analysis and Interpretation

4.4.1 Objective 1 Analysis

Objective: To develop an inexpensive automobile tracker utilizing the ESP32 chip incorporating GPS, GSM, accelerometer, and breathalyzer modules operated by a 12V car power source.

Analysis: The proposed solution has been proven effective, as all required components for building such a prototype have been integrated using low-cost embedded technology, with a total cost of less than USD $80.

The ESP32 microcontroller effectively managed various FreeRTOS tasks at once while being connected to the GPS module, GSM module, accelerometer, breathalyzer module, and relay.

Thus, the combination of the hardware modules shows that automotive monitoring can be effectively performed using inexpensive embedded technology.

4.4.2 Objective 2 Analysis

Objective: To set up Flespi and Grafana such that at least 95% of message delivery is achieved without greater than 2 seconds of telemetry latency.

Analysis: Based on the test results conducted, the WiFi telemetry subsystem had an effectiveness of 100% for message delivery in the two major tests conducted.

The average telemetry latency was reduced from 1.803 seconds in the tests conducted on March 28 to 1.125 seconds on May 13.

A comparison test between WiFi and GSM communication revealed that WiFi had lower latency compared to GSM/GPRS.

As such, the telemetry subsystem not only met but surpassed the minimum project objective by ensuring that there was a delivery rate of more than 95%.

4.4.3 Objective 3 Analysis

Objective: Development of an Android mobile app which can enable real-time tracking, monitoring, geofencing, and remote immobilizing of the vehicle engine.

Conclusion: The mobile application developed on the Android platform was successful in developing real-time monitoring, visualization of telemetry data, and controlling the vehicle engine remotely using MQTT communication with Flespi server.

The mobile application communicated with the server without latency in transmitting telemetry data.

The mobile application had all its features functioning effectively and efficiently during experiments.

4.4.4 Objective 4 Analysis

Objective: To implement accelerometer-based driver behaviour monitoring, achieving at least 85% detection capability.

Analysis: The acceleration measurement subsystem efficiently managed the identification of substantial acceleration changes, which were caused by sudden breaking, sharp acceleration, and turning actions. Nevertheless, a detailed analysis carried out as an experiment showed that there were cases when the events were incorrectly initiated while the vehicle stood still because of unexpected movement of the measuring sensor itself. Besides that, the counter of harsh events did not show monotonicity in case of the TCP reconnection. Despite the difficulties faced during testing, the system efficiently proved the possibility of employing accelerometer information to measure driver activity.

4.5 Discussion of Findings

The project involved the design of an integrated system consisting of modules to track vehicles, monitor their drivers' behavior, test for alcohol content, apply geofencing, and remotely disable their engines, all using affordable IoT technologies. In its experimental implementation, the designed system proved efficient and successfully fulfilled the majority of the project's objectives.

As far as vehicle tracking is concerned, the module managed to efficiently perform real-time tracking of vehicles by means of an ESP32 microcontroller, NEO-7M GPS module, and SIM800L GSM module, all of them operating through the Flespi cloud-based platform. Specifically, in experimental conditions, the system proved capable of providing average WiFi latency for transmitting location data as low as 1.184 seconds, outperforming the 2-second target. Moreover, communication trials delivered a perfect message-delivery rate of 100%.

Another important strength of the GPS subsystem was its high positional accuracy, which was between 2.5 m and 4.2 m in ideal outdoor conditions. This accuracy is adequate for ensuring proper vehicle monitoring, although some inaccuracies were found in areas where signals could

not be detected. Nevertheless, the positional accuracy was still acceptable for the purpose of the project.

Moreover, the engine immobilizer functioned efficiently when responding to commands as it would take merely 1.25 seconds to execute on the command sent from the control room. This response time of the engine immobilizer function will definitely aid in making attempts to prevent vehicle thefts since this will enable the owner of the car to immobilize his vehicle's engine in case he notices any attempt of stealing it within 2 seconds as required by the specifications of the project.

Finally, the subsystem that was created for detecting unusual driver's activities performed efficiently as it could detect instances of hard acceleration, sharp deceleration, and sudden movements thanks to the ADXL345 accelerometer. It seems that there were instances of the subsystem recording such activities even when the car was at rest. Even though the subsystem could detect unusual driver's actions successfully, further processing may be necessary to minimize false detections.

The proposed monitoring system demonstrates some innovations when compared to previously discussed vehicle tracking solutions from literature reviews. For instance, as noted by (Akanda et al., 2022), an existing vehicle monitoring technology was based on the use of GPS and GSM technologies and provided location information by sending SMS messages without monitoring driver's behavior, testing for alcohol consumption, or immobilizing the engine. Similarly, (Jospine et al., 2020) introduced a tracking system capable of establishing a geofence, but without additional options for monitoring the behavior of drivers.

A major innovation of the proposed vehicle tracking solution was the development of a cost-effective design based on readily available equipment. Thus, all the advanced capabilities of monitoring vehicles were made available for a wider group of customers.

Thus, the most significant achievement of the project was associated with the introduction of a cost-effective design using only commonly available hardware components. Therefore, it allowed staying within the established budget of $80 while implementing advanced monitoring systems for vehicles in IoT platforms.

Table 4.13: Hardware Expenditure for the prototype

Component	Cost ($)
ESP32 Development Board	8
SIM800L GSM Module	7
NEO-7M GPS Module	8
MQ-3 Alcohol Sensor	4
ADXL345 Accelerometer	4
Relay Module	3
Power Supply Components	10
Wiring and Connectors	8
Miscellaneous Components	10
Total	62

However, despite realizing good results, there were a number of limitations observed in the process of research. The communication process in the GSM network faced issues related to synchronization and thus delays occurred in the message transfer process. The limitation was resolved by implementing synchronization in the GSM network during testing. In addition, the detection of driver's behavior using the accelerometer faced some vibrations.

Although the prototype performed well in the lab and field tests, the practical implementation of the prototype is going to face a number of limitations that will include GSM network coverage, bad weather, dust, vibrations, and changes in the voltage. The following improvement areas may be used to improve the current project by ensuring proper placement of the sensors, improved filtering processes, extensive tests under field conditions, and classification of events.

5 CHAPTER FIVE: CONCLUSIONS AND RECOMMENDATIONS

5.1 Introduction

In this chapter, the main conclusions are presented based on the design, development, and evaluation of the effectiveness of the Automotive Tracking and Monitoring System studied in this research paper. In addition, it highlights the constraints within which this research was conducted along with suggestions for improving the system and further areas for research.

5.2 Summary of the Research

This research focused on the design and implementation of a relatively inexpensive IoT system for tracking and monitoring vehicles. In particular, the ESP32 microcontroller was adopted as the platform to develop this IoT system. As such, some of the key features incorporated in this system include GPS technology to track the vehicle's location; cloud communication to share information efficiently; an accelerometer to measure the driver's actions; detection of alcohol levels in the car; and remote engine immobilization in case of need. All these features have been efficiently integrated into a highly functional embedded system through the IoT approach.

In terms of functionality, the system uses a NEO-7M GPS module to enable real-time vehicle tracking. Moreover, the SIM800L module was used to enhance GSM and GPRS connectivity, which enables efficient data transmission. The ADXL345 accelerometer module was used to analyze driver behavior, thereby improving security. Additionally, the MQ-3 module was used to detect alcohol levels in the car, further enhancing security. Implementing FreeRTOS on the ESP32 microcontroller ensures that all tasks can be scheduled concurrently, enabling efficient multitasking.

Data transmitted to the Flespi cloud platform via telemetry was presented using Grafana dashboards and an Android application.

The experiments were carried out under real-world driving and stationary tests during several sessions on March 28, 2026, and May 13, 2026. The experiments included assessment of telemetry delay, communication reliability, GPS accuracy, driver behavior detection, and integrated sensor performance.

The results showed that the proposed solution successfully established real-time telemetry communications with delay during WiFi use and integrated sensor support.

5.3 Achievement of Research Objectives

Objective 1: To build a low-cost ESP32 automobile tracker with GPS, GSM, Accelerometer, Relay, and Breathalyzer modules.

This objective has been accomplished successfully. The designed and assembled prototype successfully integrated all specified hardware modules into a single automobile monitoring platform powered by the car's 12V power supply. The successful functioning of the GPS module, accelerometer, breathalyzer, relay, and telemetry delivery have been confirmed during experiments.

Objective 2: To configure Flespi and Grafana in order to ensure no less than 95% success of telemetry delivery with less than 2 seconds of latency.

This objective has been successfully completed in both WiFi and GSM modes. Experiments revealed a 100% telemetry delivery success rate.

Visualization of telemetry in Flespi and Grafana has been performed successfully.

Objective 3: To create an Android application for tracking and monitoring purposes with remote engine immobilization capabilities.

This goal has been achieved satisfactorily. The app was able to connect to the cloud telemetry server and deliver all functionalities required for vehicle tracking, telemetry visualization, and remote engine immobilization.

Objective 4: Monitor the driving behavior of the driver using an accelerometer.

This goal has not been met wholly. The accelerometer was capable of detecting changes in acceleration caused by quick stops, quick accelerations, and turns.

Nevertheless, during the experiments, it became clear that false-positive results arose due to sensor disturbances while the vehicle was stationary. It was also noted that the harsh events counter experienced instability during the reconnection time when using the TCP protocol.

Nonetheless, the experiments demonstrated the viability of such a solution.

5.4 Conclusions

It can be noted that the research successfully showed the possibility of developing an effective IoT automotive tracking and monitoring solution based on embedded systems.

Specifically, the designed system successfully combined the capabilities of GPS tracking, telemetry sending, driver behavior monitoring, alcohol detection, and the capability to disable an engine remotely. All of these functions were included in one comprehensive solution based on the ESP32 platform.

During experimental verification, it was proved that the designed solution could provide real-time telemetry sending through a WiFi connection. It had low latency, as well as successful cloud integration using such solutions as Flespi and Grafana.

What is more, the GPS sub-system had adequate accuracy for use in automotive tracking systems, whereas the accelerometer sub-system successfully identified aggressive driving based on the collected real-time data.

The MQ-3 subsystem for the detection of alcohol exposure performed effectively in sensing alcohol exposure and providing real-time data in telemetry form during the experiment. It was shown that the sensor can distinguish between alcohol exposure levels by identifying whether the level of alcohol exposure is moderate or high depending on how long it exceeded the threshold value and the maximum readings achieved by the sensor. The limitation identified during this experiment was that of a temporary saturation of the ADC because of the voltage limitations of the ESP32 sensor.

The engine immobilizer subsystem performed adequately in turning on and off the 5V relay module under all testing scenarios, hence proving to be an effective remote control system for the engine. In addition, the suitability of the relay module with regard to the fuel pump was tested with the help of manufacturer's performance data. From this data, it is evident that the fuel pump consumes about 7.5A at the nominal operating voltage of 13.5V and at fuel pressure of 5 bar. This level of current consumption is well within the limit of 10A as the maximum current switching capacity of the relay module. Thus, there exists a safety margin of 25%. This safety margin is essential for the correct functioning of the module despite the possibility of in-rush currents and voltage variations.

Engineered difficulties have been discovered as well, such as instability of the GSM protocol using the NTP time library, false positives detected by the accelerometer, and counter problems within the firmware.

Finally, in summary, the system created in the present research offers an implementable solution to vehicle tracking, monitoring of the fleet, and driver safety applications that is especially appropriate for use in Zimbabwe and similarly under-resourced countries.

5.5 Limitations of the System

When conducting the system experimental evaluation, certain limitations occurred.

The first limitation associated with this process is related to the subsystem that enables the use of GSM/GPRS communication. This subsystem could not successfully transfer telemetry because of the difficulties in implementing the firmware protocol handshake process and NTP (Network Time Protocol) synchronization when using GPRS.

The next subsystem used to collect data related to driver behavior through accelerometer data also faced certain limitations. There were false positives detected that occurred because of physical perturbation and vibration affecting the sensing module when the car was motionless.

Moreover, the harsh event counter demonstrated some unexpected results during TCP (Transmission Control Protocol) reconnection. The reason for that can be some firmware synchronization issues.

Finally, GPS operation had some limitations, especially in the cold start mode, when a long waiting time for a stable satellite connection was needed.

Furthermore, the MQ-3 sensor sub-system had shortcomings with regard to digital conversion (ADC) for high concentrations of alcohol. ADC on the ESP32 chip has a range of approximately 3.3V, and therefore, any inputs above this limit result in values that get cut off to the maximum value of 4095. This creates a problem where there is a reduced level of resolution during the ADC process whenever the sensors are subjected to high alcohol levels.

There are some shortcomings to the relay-pump connection as well. While the 1-channel relay can withstand a current draw of 10A, according to Handson Technology (n.d.), page 2, the

operational capacity of the pump in use requires 7.5A at 13.5V and 5 bar. The extra 2.5A capacity might not be sufficient during situations such as voltage surges, increased temperatures, and pressure in the fuel system. It must be understood that the graph produced from the Bosch Motorsport data in Figure 2.5 was attained under laboratory conditions and might not hold true to actual operating conditions in Zimbabwe when E100 fuel is used. Any fuel pump with a higher current than 10A will not be compatible with the relay.

Lastly, it should be noted that the prototype's reliance on WiFi communication for stable telemetry transmission during the experimentation phase presented an additional layer of potential vulnerability that could be explored further.

5.6 Recommendations for Future Work

Several aspects could be improved in future generations of the system.

The future developments could benefit from advanced analog conditioning of signals, for example, programmable gain adjustment and dynamic range improvement in ADCs. It is important because of the sensor saturation problems that might occur due to high levels of alcohol concentration. Another point worth considering is the use of calibration algorithms for accurate BAC estimation based on telemetry data provided by the MQ-3 sensor.

To increase the accuracy of data obtained from accelerometers, advanced filtering algorithms (Kalman filter and complementary filters) must be applied. Thus, noise will be significantly reduced, and the occurrence of unnecessary alarms related to harsh driving events will be prevented. The issue of sensor mounting and vibration isolation should also be considered.

In the future, further development of the device may include the inclusion of CAN bus or OBD II interface for the acquisition of additional information regarding the vehicle diagnostics,

which would include various parameters like the efficiency of fuel consumption, engine temperature, engine revolution, and faults codes.

Furthermore, the use of machine learning techniques will play a vital role in ensuring accurate classification and anomaly detection by the system.

Other potential enhancements include:

- Over-the-air firmware updates to streamline maintenance,
- Battery backup solutions to ensure continued operation during power interruptions,
- Push notification systems for real-time alerts,
- Cloud-based AI analytics to facilitate data processing and insights,
- Enhanced 4G/LTE communication capabilities for improved connectivity, and
- Strengthened cybersecurity measures to protect user data.

Incorporating these enhancements would significantly improve the reliability, scalability, and market viability of the developed automotive monitoring platform.

6 References

Akanda, N.I., Hossain, M.A., Fahad, M.M.I., Rahman, M.N., Khairunnaher, 2022. Cost-effective and user-friendly vehicle tracking system using GPS and GSM technology based on IoT. Indonesian Journal of Electrical Engineering and Computer Science 28. https://doi.org/10.11591/ijeecs.v28.i3.pp1826-1833

Alabi, O.O., Adeaga, O.A., Ajagbe, S.A., Adekunle, E.O., Adigun, M.O., 2024. Design and implementation of an alcohol detection driver system. International Journal of Reconfigurable and Embedded Systems 13, 278–285. https://doi.org/10.11591/ijres.v13.i2.pp278-285

Alhwety, R.N.H., Elfadil, N., 2024. Vehicle Tracking System Approaches: A Systematic Literature Review. International Journal of Computer Science and Mobile Computing 13, 23–31. https://doi.org/10.47760/ijcsmc.2024.v13i08.003

Chataut, R., Phoummalayvane, A., Akl, R., 2023. Unleashing the Power of IoT: A Comprehensive Review of IoT Applications and Future Prospects in Healthcare, Agriculture, Smart Homes, Smart Cities, and Industry 4.0. Sensors. https://doi.org/10.3390/s23167194

Chiparo, J.P., Tukuta, M., Musanzikwa, M., 2022. Driver Management and Service Delivery: Insights within State-Owned Enterprises in Zimbabwe. J. Transp. Technol. 12, 407–419. https://doi.org/10.4236/jtts.2022.123025

chiyangwa, S., Mutongi, Nyoni, S.P., 2020. European Journal of Humanities and Educational Advancements (EJHEA) EFFECTIVENESS OF WHISTLE BLOWERS' PROTECTION FRAMEWORKS IN COMBATING CORRUPTION IN ZIMBABWE. www.scholarzest.com 1.

Dakhare, P., Mishra, P., Dhore, R., Bhagat, R., Gupta, A., Rotake, D., 2022. MQ3 alcohol sensor-Based Sensing of Alcohol (Ethanol) At Room Temperature 20. https://doi.org/10.14704/NQ.2022.20.11.NQ66016

Denhere, P.T., Denhere, J., Mutipforo, G., Katsande, C., Muzenda, A., Matanana, N., Muridzi, G., 2024. Telematics as a Transformative Agent for the Zimbabwean Auto Insurance Ecosystem. Journal of Computer and Communications 12, 39–58. https://doi.org/10.4236/jcc.2024.127004

Eun, S.J., 2021. Effects of stricter drunk-driving laws on alcohol-related road traffic death, injury, and crash rates in South Korea: A synthetic counterfactual approach using Bayesian structural time-series models. Accid. Anal. Prev. 163, 106455. https://doi.org/https://doi.org/10.1016/j.aap.2021.106455

Jimada-Ojuolape, B., Spencer, O., Omolara Balogun, M., 2023. Development of a GSM based Vehicle Demobilizer and Tracking System. ABUAD Journal of Engineering Research and Development. https://doi.org/10.53982/ajerd

Jospine, A., Hanif, L., Audah, M., Hamzah, A., Qasim, H., 2020. Journal of Electronics Voltage and Application Vehicle Monitoring System with Geofencing Capability. JOURNAL OF ELECTRONICS VOLTAGE AND APPLICATION 1, 1–13. https://doi.org/10.30880/jeva.2020.01.02.001

Munuhwa, S., Chibaro, M., Kanyepe, J., Tukuta, M., 2021. OPTIMISING VEHICLE FLEET REPLACEMENT AND DISPOSAL FOR SMALL TO MEDIUM TRANSPORT COMPANIES IN ZIMBABWE.

United Nations Economic Commission for Europe (UNECE) and United Nations Economic Commission for Africa (UNECA, 2021. Road Safety Performance Review Zimbabwe. Geneva.

Zalizovskyi, M., Huzynets, N., 2024. COMPARATIVE ANALYSIS BETWEEN REAL-TIME OPERATING SYSTEMS AND SUPERCYCLES. Electronics and Information Technologies 26. https://doi.org/10.30970/eli.26.4

Zimbabwe Republic Police, 2025.Summary of Vehicle Theft Inidents. Available at: https://zrp.gov.zw/?p=8305. [Accessed 20 March 2026].[Online]

D. Matter, "Asset tracking solutions in Zimbabwe with Ezytrack," 5 April 2024. [Online]. Available: https://www.digitalmatter.com/case-studies/ezytrack-asset-tracking. [Accessed 20 March 2026].

"https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/system/freertos.html," Espressif Systems (Shanghai) Co., Ltd., 2022. [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/system/freertos.html.

https://www.bosch-motorsport-shop.com.au/blog/tech-product-overviews/select-the-correct-fuel-pump?srsltid=AfmBOopDN93MkwqXqRPcJzHQpGvsRGwpRMeqA54kg9KyM85FM9tJ6FJ6

Handson Technology (n.d.) '1 channel relay board with optocoupler 5V: User manual', Data sheet, pp. 1-2