Kingdom of Saudi Arabia

Ministry of Education

Prince Sattam bin Abdulaziz University

College of Computer Eng .& Sciences

Computer Engineering Department

المملكة العربية السعودية

وزارة التعليم

جامعة الأمير سطام بن عبد العزيز

كلية هندسة وعلوم الحاسب

قسم هندسة الحاسب

Vehicle Black Box System

Sultan Mohammed Alanazi – 441050120

Saleh Abdulrahman Alobaidi - 441051182

Ibrahim Suliman Alluhimi - 441050034

Supervised by

Dr. Ahmed Elmogy

Department of Computer Engineering
College of Computer Engineering and Sciences

May. 2024

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations which have been duly acknowledged. We also declare that it has not been previously or concurrently submitted for any other degree at PSAU or any other institution.

ID Name Signature Date:

1…441051182……………Saleh Alobaidi…………………………………………………………...

2…441050034……………Ibrahim Alluhimi…………………….…………………………………….

3…441050120……………Sultan Alanazi…..………………...………………...…………..…………

APPROVAL FOR SUBMISSION

I certify that this project report entitled "**Vehicle Black Box System**" Was prepared by **Sultan Alanazi - Ibrahim Alluhimi – Saleh Alobaidi** has met the required standard for submission in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Engineering at PSAU.

Approved by

Signature: ……...……..…………………………..................

Supervisor: Dr Ahemd Elmogy..……..…………………

Date: ……………..…………………………………………

ACKNOWLEDGEMENTS

We would like to express our sincere gratitude to all those who have contributed to the successful completion of this final project. Special thanks to our supervisor, Dr. Ahmed Elmogy, whose guidance and support were insightful throughout the entire process

ABSTRACT

The "Vehicle Black Box System" is a project aimed at creating an affordable and versatile vehicle data recording and analysis system using Arduino-based components. This system will focus on capturing essential vehicle data and providing a cost-effective solution for various applications, including accident data recording, driver behavior monitoring, and self-reporting accidents.

Utilizing a combination of GPS, accelerometer, and gyroscope sensors, the system accurately tracks the vehicle's location, speed, acceleration, and deceleration. This data is continuously recorded and stored securely in the onboard memory. In the event of an accident, the system triggers an automatic event recording, preserving critical information leading up to and during the incident. The Vehicle Black Box System is equipped with a high-resolution Dashcam that captures images and videos of the road ahead. This visual data, synchronized with the sensor readings, provides a comprehensive understanding of the circumstances surrounding any noteworthy events. The captured media is stored locally and can be accessed for post-incident analysis.

The system interfaces with the vehicle's OBD-II port to gather information about engine performance, fuel consumption, and diagnostic trouble codes. This additional data contributes to an overview of the vehicle's condition and helps identify potential issues before they escalate.
A key feature of the system is its connectivity options, allowing users to retrieve data remotely. Through a secure online site or a mobile application, users can access real-time information, and

incident recordings. This remote accessibility aids insurance claims processing, and overall vehicle monitoring.

In conclusion, the Vehicle Black Box System presented in this project serves as a powerful tool for improving road safety, analyzing driving behavior, and providing valuable insights for both individual drivers and Insurance Services Companies. Its robust design, integration of various sensors, and remote accessibility make it a versatile solution for enhancing the overall efficiency and safety of vehicular operations.

عنوان المشروع

المستخلص

نظام "صندوق السيارة الأسود" هو مشروع يهدف إلى إنشاء نظام تسجيل وتحليل بيانات المركبات بتكلفة معقولة وقابلة للتعديل وباستخدام مكونات مبنية على تقنية أردوينو. سيتم تركيز هذا النظام على التقاط البيانات الأساسية للمركبة وتوفير حلاً فعالاً من حيث التكلفة لتطبيقات متنوعة، بما في ذلك تسجيل بيانات الحوادث، ومراقبة سلوك السائق، والإبلاغ الذاتي عن الحوادث

من خلال استخدام مجموعة من أجهزة الاستشعار مثل نظام تحديد المواقع(GPS)، وجهاز الاستشعار للتسارع، وجهاز الاستشعار للزاوية (جيروسكوب)، يتتبع النظام بدقة موقع المركبة وسرعتها وتسارعها وتباطؤها. يتم تسجيل هذه البيانات بشكل مستمر وتخزينها بشكل آمن في الذاكرة الداخلية للمركبة. في حالة وقوع حادث، يقوم النظام بتشغيل تسجيل حدث تلقائي، محافظًا على المعلومات الحيوية التي تسبق الحادث وأثناءه. يأتي نظام صندوق السيارة الأسود مزودًا بكاميرا عالية الدقة تلتقط صورًا ومقاطع فيديو للطريق المتجه إلى الأمام. توفر هذه البيانات البصرية، المتزامنة مع قراءات الاستشعار، فهمًا شاملاً للظروف المحيطة بأي حدث قابل للإشارة. تخزين الوسائط الملتقطة محليًا ويمكن الوصول إليها لتحليل ما بعد الحادث.

يتفاعل النظام مع منفذ OBD-II في المركبة لجمع معلومات حول أداء المحرك واستهلاك الوقود ورموز الأخطاء التشخيصية. تساهم هذه البيانات الإضافية في فهم حالة المركبة وتساعد في تحديد المشكلات المحتملة قبل أن تتفاقم. ميزة رئيسية في النظام هي خيارات الاتصال، مما يتيح للمستخدمين استرجاع البيانات عن بُعد. من خلال موقع آمن على الإنترنت أو تطبيق الجوال، يمكن للمستخدمين الوصول إلى معلومات فورية وتسجيل الحوادث. يسهم هذا التواصل عن بُعد في معالجة مطالبات التأمين ومراقبة المركبة بشكل عام.

في الختام، يُعَدُّ نظام صندوق السيارة الأسود الذي قُدِّمَ في هذا المشروع أداةً قويةً لتحسين سلامة الطرق، وتحليل سلوك القيادة، وتقديم رؤى قيمة لكل من السائقين الفرديين وشركات خدمات التأمين. يتميز بتصميمه القوي وتكامله مع مجموعة متنوعة من الأجهزة الاستشعارية وإمكانية الوصول عن بُعد، مما يجعله حلاً متعدد الاستخدامات لتعزيز الكفاءة والسلامة العامة لعمليات النقل.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS/ABBREVIATIONS

IoT   Internet of Things

IDE Integrated Development Environment

PCBs printed circuit boards

OBD On-Board Diagnostic

PIDs Parameter IDs

GUI Graphical User Interface

CHAPTER 1

INTRODUCTION

1.1 Problem Statement

The primary mode of transportation in Saudi Arabia is cars. According to a statistic made by the Ministry of Health, road traffic accidents caused 105,209 deaths or injuries between 2018 and 2020 [1]. This shows the urgent need for effective accident investigation techniques and improved data collection methods. Implementing advanced technologies and conducting thorough analysis can help identify the root causes of accidents and develop preventive measures to reduce their occurrence. Additionally, Najm (accident reporting company) employees are experiencing difficulties due to getting the accident report late or having trouble getting to the accident scene.

1.2 Project Objectives

- In order to lower the death rate, our project's goal is to install a black box system inside cars that will help with vehicle safety, improve crash victim care, assist insurance companies with their auto crash investigations, improve road conditions and provide real-time tracking of the vehicle through a phone app.
- The black box system will collect and record data of various parameters such as speed, acceleration, braking, and GPS location. This data will be crucial for accident investigators to accurately reconstruct the events leading up to a crash. Additionally, the black box system will provide real-time alerts to emergency services in the event of a crash, allowing for quicker response times and improved care for crash victims. This will also support smart cities development through the use of IoT (Internet of Things).

1.3 Project Motivation

Most of our experiences regarding accedints have been difficult in terms of contacting Najm and giving our exact location. The Najm employee is approximately taking 30-45 minutes till arrival. The accident investigation also takes another 30 minutes. Dealing with the other parties may sometimes be difficult as they may lie or report a differint story to the Najm emloyee. Also, car manufacturing companies sometimes needs accident information on their cars to study and develop new and improved safety options or features.

With the Black Box system, we aim to reduce the time taken for the Najm employee to come and report filling by providing the accident report immediately after it occurs and sending it to Najm company thus saving time and resources and getting a clear report of the accident in case the other party disagrees with the employee. Car companies can implement the black box system on their cars and get information easily instead of doing on-field information gathering on past accidents.

1.4 Potential impact of project on society (global, national, local)

1. Improved safety: Vehicle black box can help investigate accidents and provide data that can be used to improve road safety. This data can lead to the development of better safety features and more effective road policies.
2. Accident reconstruction: black box data can provide crucial information for accident reconstruction, helping law enforcments (Najm), insurance companies, and the legal system to determine fault and liability more accurately.

3. Insurance benefits: insurers can use the data to assess risk more accurately, leading to easier insurance claims and encouraging better driving habits.
4. Smart city: the black box system can help in developing a smart city through the use of IoT.

This doesn't mean that the proposed black box system is perfect. Some challenges need to be addressed in the future to strengthen the proposed system.

1.5 Approach

We first identify the specific data elements that the black box should record, such as speed, brake status, GPS location, and car body status. Then choose the appropriate sensors (e.g, acceleometers, GPS) and familiarize ourselves with them and their connection to the Arduino board, and determine the optimal location of these sensors within the vehicle to collect the required data accurately. We store the data in a micro-SD chip, considering factors like data protection, implementing security measures to protect the data from unauthorized access, and hacking.
The black box system will transmit data in real-time, so we will add a mobile app for the vehicle owner to display the car status and real-time tracking using the GPS module. The application interface should be user friendly and it will be used as data display for the black box system so that the insurance companies and Najm employess can read the data easily.

1.6 Project Organization

Chapter 2 covers the components used and the specifications of the sensors and how they work.
Chapter 3 covers the connections to the Arduino board and building the software.

CHAPTER 2

BACKGROUND

Introduction:

In this chapter, we will specify the components used in the project while giving a description on each one to compare and find the best possible optin to integrate into our project.

2.1 Sensors

OBD-II UART:

On-Board Diagnostics II Universal Asynchronous Receiver/Transmitter, or OBD-II UART, is the name of a communication standard used in cars for reporting and diagnostics of vehicle-related data [2]. A standardized protocol called OBD-II enables external devices to interface with a car's onboard computer systems.

The Universal Asynchronous Receiver/Transmitter (UART) communication protocol is the one that is utilized in OBD-II. A typical serial communication protocol called UART allows data to be sent between devices bit by bit in a serial way.

The UART communication protocol is used in OBD-II to provide diagnostic data to external devices from the vehicle's onboard computer (also known as the Engine Control Unit, or ECU). This data may include performance, pollution, and other sensor readings for the car.

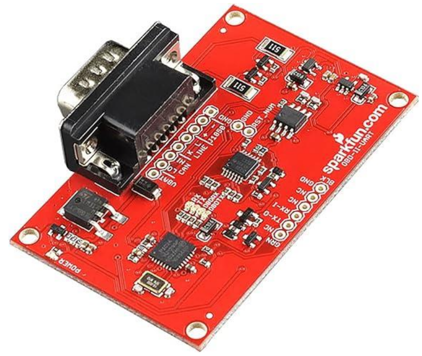

Figure 2.1: OBD-II UART

Accelerometer and Gyroscope (MPU-6050):

A sensor module called the MPU-6050 combines an accelerometer and gyroscope. It combines a 3-axis gyroscope, 3-axis accelerometer, and a digital motion processor into a compact 6-axis motion tracking device. Additionally, it incorporates an on-chip temperature sensor as an extra function. It is frequently utilized in applications where tracking movement and location is essential, such as orientation detection and motion tracking.

Figure 2.2: MPU-6050 Module

As shown in figure 1, the module has 8 pins with 4 pins being part of the I2c bus. The SCL (Serial Clock Live) and SDL (Serial Data Live), wich only require two wires for communicating, when the MPU-6050 acts as an I2c slave, while the microcontroller acts as an I2c master [3]. And the XCL (Auxiliary Clock) and XDL (Auxiliary Data) are a master I2c clock and data pins, The ADO is an I2C Slave Address LSB pin. This is 0th bit in 7-bit slave address of device. If the pin is low, the address of the I2c is 0x68. If the pin is high, the address of the I2c is 0x69. Three-axis gyroscope: The MPU6050 utilizes Micro Electromechanical (MEMS) technology in its three-axis gyroscope. It measures rotational velocity along the X, Y, and Z axes.

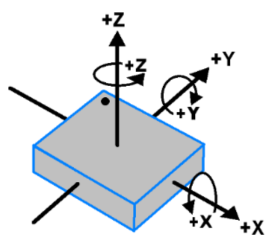

Figure2.3: MPU-6050 Orientation and Polarity of Rotation

A MEM inside the MPU6050 detects the vibrations that the Coriolis Effect produces when the gyros are rotated around any of the axes shown in figure 2. The resulting signal is amplified, demodulated, and filtered to produce a voltage that is proportional to the angular rate. This voltage is digitized using 16-bit ADC to sample each axis, the full-scale range of the output are ±250, ±500, ±1000, and ±2000 °/sec degree per second or (dps) [4].

Three-axis Acceleometer: The MPU-6050 uses MEMS to detect the tilting angle and incline along the X, Y, Z axes. The displacement of a moving plate unbalances the differential capacitor, causing sensor output. Acceleration is directly proportional to output amplitude, and the device's full-scale range of acceleration is ±2g, ±4g, ±8g, and ±16g (gravity force).

The device measures 0g on the X and Y axis and +1g on the Z axis when placed on a flat surface.

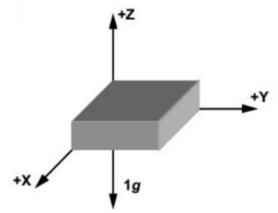

Figure 2.4: Acceleometor axis on flat surface

Table 2.1: Specification of MPU6050 Sensor

Voltage Supply	**3.3-5V**
Logic level	**3.3V**
Acceleometer Measurement Range	±2g, ±4g, ±8g, ±16g
Acceleometer Sensivity	16384, 8192, 4096, 2048 LSB per g (LSB/g)
Output Data Rate (ODR)	8kHz – 1.25kHz
Gyroscope Measurement Range	±250, ±500, ±1000, and ±2000°/sec
Gyroscope Sensivity	131 LSB/dps ±250 dps, 65.5 LSB/dps ±500 dps, 32.8 LSB/dps ±1000dps, 16.4 LSB/dps ±2000dps
Output Data Rate (ODR)	8kHz – 1.25kHz

Digital Motion Processor (DMP): The DMP can be used to process complex algorithms directly on the board, it processes algorithms to turn the raw value from the sensors into stable precision data.

Micro SD storage:

This kind of flash memory storage is frequently found in portable electronics like tablets, digital cameras, and smartphones. It's a small, detachable storage medium that gives these gadgets more storage space. MicroSD cards can be used to expand storage or transfer data. They are available in several capacities, such as 16GB, 32GB, 64GB, 128GB, and 256GB. Micro SD (Secure Digital) storage works by using flash memory technology to store and retrieve digital data.

Figure2.5: Micro SD Card

The Micro-SD Arduino module is a compact add-on component that enables Arduino microcontrollers to interface with Micro-SD cards for data storage. It provides a convenient way to read and write data to these small, high-capacity storage cards, expanding the data storage capabilities of Arduino projects. The module typically includes a slot for the Micro-SD card and simplifies the communication between the Arduino board and the external storage device, making it easy to incorporate data logging, file storage, or other applications requiring extended memory in Arduino-based projects. Since we only need up to 2 minutes of recording, we will be using 16GB Micro-SD.

GSM SIM900A:

In order to allow the Arduino board to interface with the GSM (Global System for Mobile Communications) cellular network, a hardware component known as a GSM shield can be added. Usually, it allows you to use a SIM card to access mobile data services, send and receive SMS, and make and receive calls. Applications needing remote communication or cellular network control frequently use these shields.

Figure 2.6: GSM SIM900A

We attach the GSM shield to the Arduino, typically using pins or a dedicated slot.

insert a SIM (Subscriber Identity Module) card into the GSM shield. The SIM card contains important network credentials and identifies your device on the mobile network. The GSM shield requires power, which can be supplied by the microcontroller or an external source. The microcontroller is used to transmit AT (Attention) commands to the GSM module. These text-based commands tell the GSM module how to do different things, such send SMS messages, make calls, and browse the internet. To link the GSM shield to the network, it connects with neighboring cell towers. It can send and receive data, including voice conversations, SMS messaging, and internet connectivity once it is connected. To operate the GSM module, we program the microcontroller with code. enabling mobile network connectivity-dependent features including security systems, remote monitoring, and Internet of Things applications. Essentially, a GSM shield acts as a bridge between our microcontroller-based project and the GSM network, enabling us to send data via SIM cards and AT commands via cellular networks. This is very helpful for a variety of applications that require remote control and communication.

GPS module NEO-7M:

The NEO-7M is a GPS (Global Positioning System) module commonly used in electronic projects. Compact and versatile, it employs the u-blox NEO-7M chipset to provide accurate positioning and timing information. This module communicates with microcontrollers like Arduino through serial communication, offering latitude, longitude, altitude, and time data. Its high sensitivity and quick time-to-first-fix make it suitable for our project, where precise location information is essential

Figure 2.7: GPS module NEO-7M

Being equipped with a GPS receiver enables the NEO-7M to pick up signals from several GPS satellites. The module can pinpoint its exact geographic location in terms of latitude, longitude, and altitude by gathering data from various satellites. It is also capable of giving precise timing data. The NEO-7M is compatible with various global navigation satellite systems, such as GLONASS (Russian), QZSS (Japanese), BeiDou (Chinese), and Galileo (European), even though its primary purpose is GPS. The precision and dependability of positioning are enhanced by this multi-GNSS compatibility, particularly in situations where GPS signals are blocked.

Because of its high-sensitivity GPS technology, the NEO-7M can detect and track satellite signals even in difficult-to-reach places like densely forested areas or urban canyons. This qualifies it for uses where a dependable and strong GPS signal is necessary.

The module communicates with a microcontroller or computer using UART (Universal Asynchronous Receiver/Transmitter) serial communication. This allows you to send commands to the module, configure settings, and receive location data and other information.

NMEA (National Marine Electronics Association) sentences are standard messages used in GPS communication, and the NEO-7M can deliver a variety of data types. Latitude, longitude, altitude, speed, course, and the number of satellites visible are among the usual data that the module outputs. We can analyze and process this data for our project. An external antenna connector is typically included with the NEO-7M, allowing user to improve signal reception by attaching an active or passive GPS antenna. In situations where the module's integrated antenna might not be able to receive signals well, this can be helpful. The module typically operates on a voltage supply in the range of 2.7V to 3.6V, making it compatible with a wide range of power sources, including batteries. The NEO-7M is highly configurable through commands and software. We can adjust parameters like update rate, data format, and power modes to suit our specific requirements

IR Flame sensor:

An IR (Infrared) Flame Sensor is a device that is designed to detect the presence of a flame or fire by sensing the infrared radiation emitted by flames. IR Flame Sensors work on the principle that flames emit infrared radiation in the form of heat. This radiation is not visible to the human eye but can be detected by specialized sensors. When a flame is present, it emits infrared light with a specific wavelength range, and the sensor detects this radiation.

Figure 2.8: IR Flame Sensor

The typical function of infrared flame sensors is to identify flames within a certain range, which varies depending on the sensor. The size and intensity of the flame, as well as the sensitivity of the sensor, all affect how far away a flame may be detected. Applications where fire detection is essential, like industrial safety, fire alarm systems, and monitoring devices, frequently use infrared flame sensors. Applications in the gas and oil industries use them to find flames in furnaces and burners.

These sensors can provide both analog and digital outputs. An analog output provides a varying voltage or current signal based on the intensity of the detected infrared radiation, allowing for finetuning the sensitivity. Digital outputs typically provide a binary signal (flame detected or not detected) and are commonly used in simple fire detection systems. Although IR flame sensors work well to detect flames, they can also pick up false positives from other heat or infrared radiation sources, such sunshine or heated objects. The positioning and design of sensors are essential for lowering false alerts. Since IR flame sensors have a fast response time, they are a good fit for applications where a fire must be detected quickly.

Digital camera:

It is an electronic device that captures and stores photographs or videos in a digital format. It replaces traditional film-based cameras and allows for easy viewing, sharing, and editing of images using digital technology.

Figure 2.9: Digital Camera

The camera uses an image sensor, typically a CCD (Charge-Coupled Device) or CMOS (Complementary Metal-Oxide-Semiconductor) sensor, to convert the incoming light into an electrical signal. Each pixel on the sensor represents a tiny light-sensitive element. The electrical signals from the image sensor are converted from analog to digital using an Analog-to-Digital Converter (ADC). This results in a digital representation of the image. The digital image goes through in-camera processing. This includes adjustments for color balance, exposure, sharpness, and other settings depending on the camera's configuration. The processed digital image is saved to a memory card or internal storage in formats like JPEG or RAW.

ISD1820 :

The ISD1820 Voice Recorder Module is built around the ISD1820 IC, a single chip voice recorder IC designed for recording and replaying a single message.

Figure 2.10: ISD1820

It is possible to set the ISD1820 Voice Recorder Module to store messages for a duration of 8 to 20 seconds in its non-volatile memory. A number of control pins are available on the IC, such as FWD (Forward), PLAYE (Playback Enable), PLAYL (Playback Load), and REC (Record) [3]. This module's integrated audio amplifier, which can directly drive an 8-inch, 0.5-watt speaker without the need for an additional amplifier circuit. Still, you can use an external amplifier integrated circuit (IC) to connect this module's output to stronger speakers.

Attach a tiny 8Ω speaker across the SP+ and SP-pins at the module's output. The module begins recording as soon as the record button (REC) is pressed. Press the button again and again until the entire message is recorded, which should take about 10 seconds. You can press PLAYE or PLAYL to start playing back. The message is played back in its entirety when you press the PLAYE button once. The message begins to play when you push and hold the PLAYL button. You can release the button at any moment to halt the playback. When the PE Jumper turns on, playback is in continuous loop mode.

Water Level sensor:

It is a device that detects the presence or absence of water and is commonly used to alert individuals or systems to potential water leaks or flooding.

Figure 2.11: Water Level Sensor

Water sensors work by detecting the presence or absence of water through various methods. One common type of water sensor uses conductivity. Since water is a good electrical conductor, these sensors usually feature two electrodes. Water creates an electrical circuit when it interacts with the electrodes, which allows current to flow through the sensor. This change in conductivity is then detected. Another method involves measuring the resistance in the presence of water. When water contacts the sensor, it decreases the electrical resistance, which is then measured.

Some sensors use changes in capacitance to detect water. When water is present, it changes the capacitance of the sensor, and this change is used for detection. With this sensor, we can accurately determine if the vehicle has encountered slippery road conditions and if it was the cause for the accident.

2.2 Microcontrollers

Microcontrollers are compact, integrated circuits that serve as the brains of embedded systems. These tiny computing systems are made up of a single chip that houses the input/output peripherals, memory, and processing core. Designed to function in low-power, real-time situations, microcontrollers find widespread use in many applications, ranging from household appliances and automotive systems to industrial machinery and consumer electronics.

They are essential in the field of automation and smart devices because of their adaptability, allowing developers to customise them for specific applications. Microcontrollers are programmed to execute predefined functions, responding to inputs from sensors and interacting with other components to perform complex operations with efficiency and precision. We chose a microcontroller for its convenience, and it meets all of our needs perfectly.

2.2.1 Arduino

Arduino is an open-source platform designed for creating interactive projects, prototypes, and various electronic applications. Arduino borads are equipped with microcontrollers that can be programmed to sense and control physical devices. The platform is popular among hobbyists, students, and professionals due to its ease of use and accessibility [5].

Arduino programming is typically done in the Arduino IDE using a simplified version of C/C++. It's well-suited for embedded programming and hardware control.

2.2.2 Types of Arduino

1. Arduino Uno
2. Arduino Mega
3. Arduino Due
4. Arduino ADK

Arduino Uno:

Figure 2.12: Arduino uno

- Microcontroller: ATmega328P
- Digital I/O Pins: 14

- Analog Input Pins: 6
- Clock Speed: 16MHz
- Flash Memory: 32KB
- SRAM: 2KB
- EEPROM: 1KB

Arduino Mega:

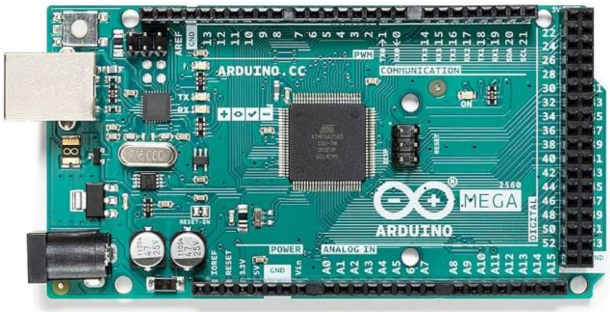

Figure 2.13: Arduino Mega

- Microcontroller: ATmega2560
- Digital I/O pins: 54
- Analog Input Pins: 16
- Clock Speed: 16MHz
- Flash Memory: 256KB
- SRAM: 8KB
- EEPROM: 4KB

Arduino Leonardo:

Figure 2.14: Arduino Leonardo

- Microcontroller: ATmega32U4
- Digital I/O Pinss: 20
- Analog Input pins: 12
- Clock Speed: 16MHz
- Flash Memory: 32KB
- SRAM: 2.5KB
- EEPROM: 1KB

Arduino Due:

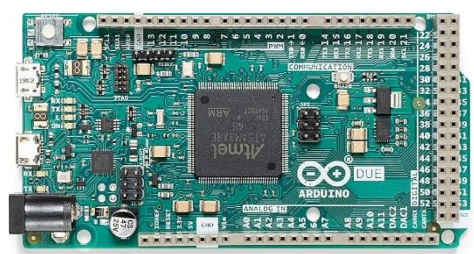

Figure 2.15: Arduino Due

- Microcontroller: AT91SAM3X8E
- Digital I/O Pins: 54
- Analog Input Pins: 20

- Clock Speed: 84MHz
- Flash Memory: 512KB
- SRAM: 96KB
- EEPROM: None

Table 2.2: Comparison Between Arduino types

Type	Microconroller	Digital Pins	Analog Pins	Clock Speed	Flash Memory	SRAM	EEPROM
UNO	ATmega328P	14	6	16MHz	32KB	2KB	1KB
Mega	ATmega2560	54	16	16MHz	256KB	8KB	4KB
Leonardo	ATmega32U4	20	12	16MHz	32KB	2.5KB	1KB
Due	AT91SAM3X8E	54	20	84MHz	512KB	96KB	None

2.2.3 Raspberry Pi

The Raspberry Pi is a series of single-noard computers developed by the Raspberry Pi Foundation. These small and affordable computers are designed to promote the teaching of basic computer science in schools and developing countries. They have gained popularity among hobbyists and DIY enthusiasts for their versatility and ease of use. With its low power consumption and compact size, the Raspberry Pi has also been used in various applications such as media centers, home automation systems, and even as the brain of robots. Raspberry has four generations, with every generation having a model A and B [6].

Raspberry Pi supports a wide range of programming languages, including Python, C/C++, Java, and more. It is more versatile for software development

2.2.4 Types of Raspberry Pi

- Raspberry Pi Model A+
- Raspberry Pi 2 Model B
- Raspberry Pi 3 Model B
- Raspberry Pi 4 Model B

Raspberry Pi Model A+:

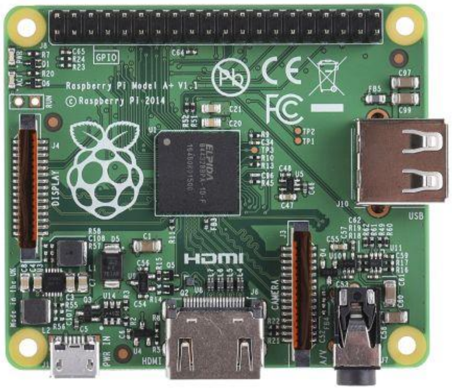

Figure 2.16: Raspberry Pi Model A+

- Processor: BCM2835
- RAM: 512MB
- Connectivity: None
- GPIO: 40 Pins
- USB: 1x USB 2.0
- Clock Speed: 700MHz

Raspberry Pi 2 model B:

Figure 2.17: Raspberry Pi 2 Model B

- Processor: BCM2836 / 7
- RAM: 1GB
- Connectivity: Ethernet
- GPIO: 40 Pins
- USB: 4x USB 2.0
- Clock Speed: 900MHz

Raspberry Pi 3 model B:

Figure 2.18: Raspberry Pi 3 Model B

- Processor: BCM2837
- RAM: 1GB
- Connectivity: Ethernet, wireless
- GPIO: 40 Pins

- USB: 4x USB 2.0
- Clock Speed: 1.2GHz

Raspberry Pi 4 Model B:

Figure 2.19: Raspberry Pi 4 Model B

- Processor: BCM2711
- RAM: 2GB, 4GB, 8GB
- Connectivity: Gigabit ethernet, dual-band wireless, Bluetooth
- GPIO: 40 Pins
- USB: 2x USB 3.0 and USB 2.0
- Clock Speed: 1.5GHz

Table 2.3: Comparison Between Raspberry Pi Types

Type	Processor	RAM	Connectivity	GPIO	USB	Clock Speed
Raspberry Pi Model A+	BCM2835	512MB	None	40 Pin	1x USB 2.0	700MHz
Raspberry Pi 2 Model B	BCM2836 / 7	1GB	Ethernet	40 Pin	4x USB 2.0	900MHz
Raspberry Pi 3 Model B	BCM2837	1GB	Ethernet, wirelwess	40 Pin	4x USB 2.0	1.2GHz
Raspberry Pi 4 Model B	BCM2711	2GB, 4GB, 8GB	Gigabit ethernet, dual-band wireless, Bluetooth	40 Pin	2x USB 3.0 and USB 2.0	1.5GHz

Table 2.4: Comparison Between Arduino and Raspberry Pi

Type	Arduino Uno	Raspberry Pi 3
Price	$29	$37
CPU Type	8-bit Microcontroller	64-bit Microcontroller
Clock Speed	16MHz	1.2GHz
Networking	None	Wi-Fi, Bluetooth, Ethernet
USB Ports	1x USB	4x USB
Power consumption	0.3 W	1.9 -2.3 W

The Arduino Mega 2560 board is reasonably priced, simple to use, program, and integrate. Offering a wide range of analog and digital I/O pins, which makes it suitable for applications that require a large number of inputs and outputs. And that is what we are looking for.

2.3 Related Works

1- In [7] the author's primary goal is to evaluate and apply the Black-Box system prototype in all types of vehicles. All of the fundamental characteristics needed to analyze system and vehicle environment actions are present in this prototype. This system assists in determining the precise cause of an accident, enhances vehicle safety, and supports the car's owner as well as other parties in the event of an insurance claim.

The system contains two sub-systems. A vehicle monitoring system, and a driver monitoring system. The first contains vehicle recording ensors sush as, temperature and humidity sensor and speed sensor. Similar to our project it saves the recorded data on a micro-SD chip. The Second sub-system monitors the driver by a cctv night-vision camera and a microphone to record audio inside the vehicle, and uses a digital video recorder to store the video and audio on a hard disk drive.

2-This study [8] introduces a black box automated vehicle accident detection and reporting system. The proposed system uses an ARM controller, with LCD display and a GPS and GSM module. The black box is placed inside a moving vehicle to detect and report accident in case of emergency, when the accident occours the microcontroller communicates with the GPS module in predefined functions and sends the longitude and latitude of the vehicle to the first responder using GSM.

The system gathers position information for real-time tracking using Google Earth GPRS. MEMS sensor is used to detect the place of the accident, sensor recognizes the vehicle's vibration and sends electric signal to controller, the controller utilizes GPS to locate specific area where accident happens, along with recording audio for 2 minutes when the accident occurs

3- The author’s main idea in [9] is that Black Box systems automatically record and report traffic rule breaches to the traffic authority system, or they monitor fluctuations in cars to reduce and analyze accidents. In many countries, traffic violation recording is implemented manually, resulting in false traffic fine recordings even when the vehicle is not at the same location at traffic fine recording, so it is necessary to develop such systems that confirm the availability of the vehicle in that location during violation occurrence.

This paper presents the development of a Vehicle Black Box system based on 433MHz long-range wireless modulation technology for two purposes: first, to address safety concerns while driving by utilizing gas and flame sensors, and second, to address issues with false traffic violation recordings by using a GPS module to verify the vehicle's presence at the traffic violation location.

The vehicle-to-Infrastructure communication method is formed by the employment of one base station and two nodes, which represent cars, in the system-based Internet of Vehicles model to send the collected data. The system is tested in real time, and the sensed data is sent and saved in a database for analysis in the future. The GPS data is utilized to verify the precise position of the vehicle with time and date stamps. A database in the base station had the data from two vehicles, which showed their precise location along with time and date stamps. For analytical reasons, the measured values of the Signal to Noise ratio and Received Signal Strength indicator are saved. The readings address how well the system performs in an Internet of Vehicles context.

4- The paper [10] describes the design and implementation of a GPS-GSM-based system for tracking cars across large geographic regions in an efficient manner. A plan is put up to lower the number and cost of SMS texts sent. An analysis of the system's performance under both artificial and real-world test scenarios has shown its ability to significantly reduce costs. Additionally, a trade-off between cost savings and tracking precision is offered by the proposed system.

5- This article’s [11] main goal is to build a few sensor-based black box systems that will help us prevent traffic accidents by continuously providing the driver with precise instructions. It will upload the evidence to its server for additional processing at the same time. First off, this black box system has several different sensors, such as RFID, alcohol sensors, LIDAR, and cameras. Additionally, this technology provides a way to determine how sleepy the driver is. A monitor mounted directly in front of the driver's seat will display all of the information. Finally, GPS and GSM will be used to send information about the vehicle's position and condition to the appropriate authority.

6- The main objective of this article [12] is to create a Vehicle Black Box System prototype that can be put in any type of vehicle worldwide. It is possible to design this prototype using the fewest possible circuits. The Black box has the potential to lower the fatality rate by helping to build safer cars, better care for collision victims, assistance for insurance companies during vehicle incident investigations, and improved road conditions.

7- In this article [13] an intended tracking device known as a "vehicle black box" is mounted on the dashboard and monitors driving behaviour and vehicle performance to guarantee both the driver's and the vehicle's safety and security. This project's primary goal is to develop an Internet of Things (IoT) version of the Vehicle Black Box System that can be installed in any kind of vehicle, anywhere on the globe. The automobile will have sensors and a camera installed to record activities inside. Data and images will be delivered to the mail and short message service, and a real-time web page tracking system will be in place. For medical aid, the accident and its approximate location are provided. The primary goals of this article are to assist in identifying fraud with ease and to enhance the care provided to crash victims.

8- The work described in article [14] attempts to provide an affordable solution for the design and development of an event data recorder, which has mostly been adopted from the aviation industry due to the associated benefits and need. The paper presents an integrated design of the black box with the fundamental data recorder features, which could be very helpful for domestic vehicles. The black box has several other features that could help reduce the number of accidents or, at the very least, act as an analysis tool to stop accidents by looking at past accidents.

In addition, the black box has an automatic accident notification system that helps notify the closest hospital and the traffic authority by giving not only the accident's coordinates but also the precise physical address where emergency medical attention can be received right away, potentially saving many lives every day. Other features of the black box include sophisticated web tracking that can be accessed from anywhere at any time, CAN compatibility, and an aesthetically pleasing user panel. Thus, by combining so many features, the overall cost is greatly optimized.

CHAPTER 3

SYSTEM DESIGN

3.1 System Software

3.1.1 Arduino IDE

The Arduino Integrated Development Environment (IDE) is a user-friendly software platform designed to facilitate the programming and development of Arduino microcontroller boards. It provides a streamlined and accessible interface for writing, compiling, and uploading code to Arduino boards. The IDE supports the Arduino programming language, which is a simplified version of C and C++. It offers a range of built-in functions and libraries that simplify common tasks, allowing us to focus on our project rather than intricate programming details. The IDE includes a text editor with syntax highlighting, a message console for debugging, and a simple one-click upload process to transfer code to the Arduino board [15].

Figure 3.1: Arduino IDE

3.1.2 Fritzing

Fritzing is a popular open-source software tool designed to facilitate the design, documentation, and sharing of electronic circuits and prototypes. Fritzing provides a user-friendly platform for creating visual representations of circuits through a drag-and-drop interface. It allows users to design schematics, design custom printed circuit boards (PCBs), and create interactive and visually appealing layouts of their electronic projects. Fritzing also supports the generation of breadboard views, enabling users to virtually prototype their circuits before moving on to the physical construction.

3.1.3 OBD Fusion

OBD Fusion is a powerful automotive diagnostic app designed for use with OBD-II (On-Board Diagnostics)-compliant vehicles. Available for both iOS and Android platforms, OBD Fusion transforms a smartphone or tablet into a sophisticated car diagnostic tool. The app connects to a vehicle's OBD-II port using a compatible OBD-II Bluetooth or Wi-Fi adapter, allowing users to access real-time data from the vehicle's onboard computer. OBD Fusion provides a comprehensive set of features, including the ability to read and clear diagnostic trouble codes (DTCs), monitor live sensor data, log performance metrics, and even create custom dashboards for personalized data visualization. It supports a wide range of OBD-II protocols, making it compatible with various vehicle makes and models.

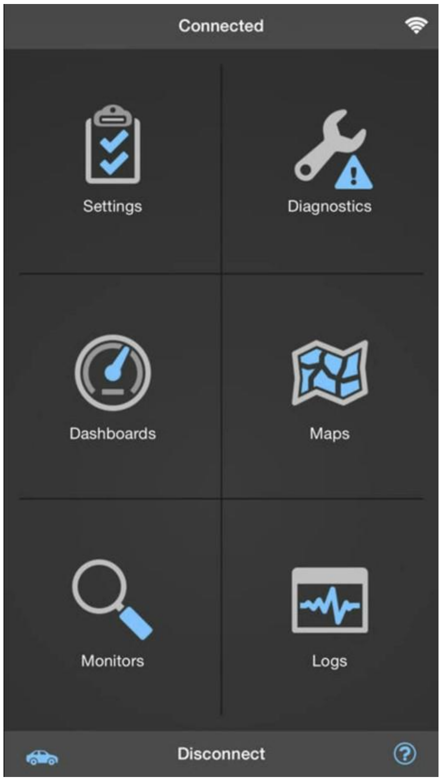

Figure 3.2: OBD Fusion Dashboard

OBD PIDs:

As a diagnostic tool, OBD-II PIDs (On-board Diagnostics Parameter IDs) are codes that are used to request data from a vehicle. Several OBD-

II PIDs are defined under SAE standard J1979. Manufacturers also define additional PIDs specific to their vehicles.

The standard OBD-II PIDs as specified by SAE J1979 are displayed in the table below. For every PID, the expected response is given along with information on how to convert the response into practical data. Once again, not every car will accept every PID, and some manufacturers may specify unique PIDs that are not covered by the OBD-II standard [16].

Table 3.1: OBD-II PIDs

PIDs (hex)	PIDs (decimal)	Description	Min value	Max value	Unit
03	3	Fuel system status	-	-	-
05	5	Engine coolant temperature	-40	215	°C
0C	12	Engine speed	0	16,383,75	Rpm
0D	13	Vehicle speed	0	255	Km/h
1F	31	Runtime since engine start	0	65,535	S
2E	46	Fuel tank level input	0	100	%
5C	92	Engine oil temperature	-40	210	°C

3.2 System Architecture

The moment the Arduino board is working, the sensors recording begins and saved on the micro-SD, with the information from the OBD-II. When an accident is detected by the infrared sensor`s threshold crossing, all the recorded data by the sensors in the last two minuets is uploaded to the GSM and sends a notification to the smartphone. The Arduino displays the information taken by the sensors and the OBD-II port in report file indicating the vehicle part that involved in the accident as it was the part that was crossed by the threshold, the report also includes the speed, location, date and time with the vehicle`s owner information (name, ID)

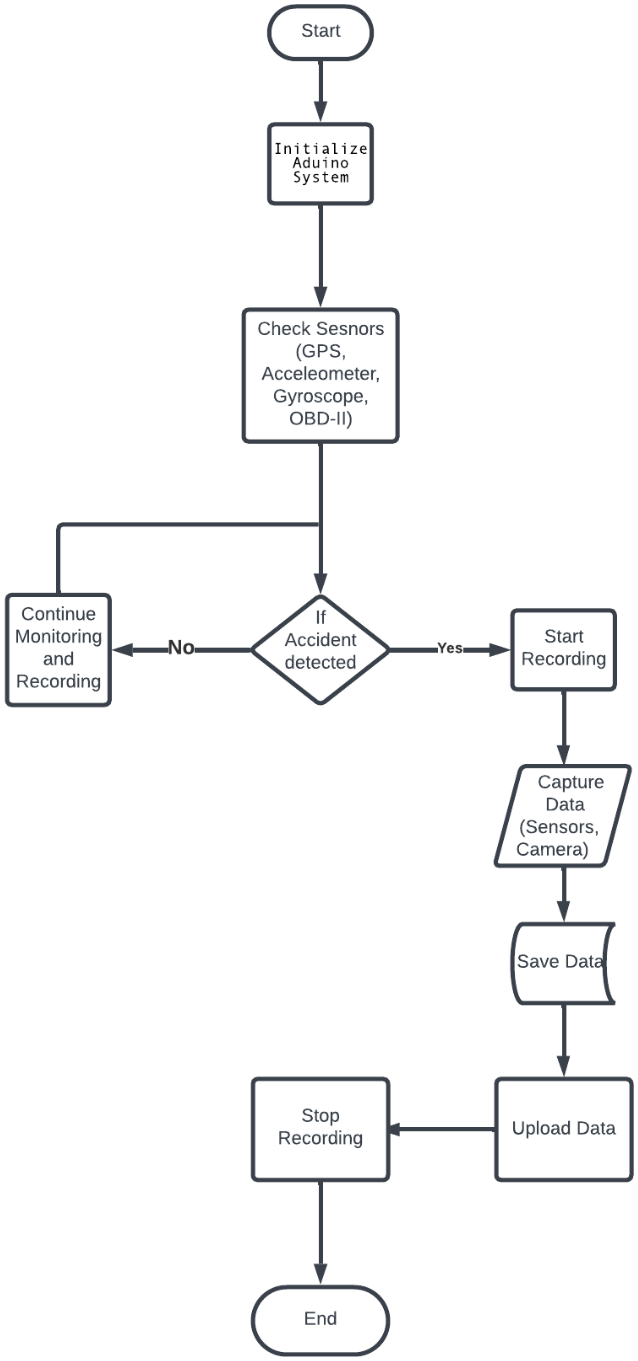

Figure 3.3: System Flowchart

3.3 Project Design Constraints

Economic Constraints:

- **Cost Efficiency:** The implementation of a black box system should be economically viable for both manufacturers and consumers to encourage widespread adoption.
- **Production Costs:** The cost of integrating and manufacturing the black box technology should not significantly increase the overall cost of the vehicle.
- **Overall Cost:** implementing and maintaining a black box system in vehicles can increase their cost, potentially leading to higher purchase prices for cosumres, especially in low-income communities

Environmental Constraints:

- **Emission Reduction:** The black box system should not contribute to increased environmental pollution, and efforts should be made to reduce the environmental impact of its production and disposal.
- **Energy Efficiency:** The system's energy consumption during operation should be optimized to minimize its carbon footprint.

Ethical Constraints:

- **Privacy:** The black box system should be designed and implemented with a strong emphasis on user privacy, ensuring that personal data is handled responsibly and transparently.
- **Privacy concerns**: collecting data about an individual's driving habits raises significant privacy concerns. There is the potential for abuse or unauthorized access to this information, leading to concerns about surveillance.
- **Data security:** ensuring the security of the datacollected by these devices is crucial. If the data falls into the wrong hands, it could be misused for malicious purposes.
- **Inaccurate data interpretation:** misinterpretation of black box data or relaying solely on this data for accidents investigation can lead to incorrect conclusions and potentially unfair legal consequences.

Health and Safety Constraints:

- **System Reliability:** The black box should be designed to operate safely and reliably to avoid malfunctions that could lead to accidents or harm.

Manufacturability Constraints:

- **Integration Complexity:** The black box system should be designed for easy integration into existing vehicle architectures without requiring extensive modifications.

- **Supply Chain Considerations:** The availability and sustainability of the materials required for manufacturing the black box should be taken into account.

Political Constraints:

- **Regulatory Compliance:** The system must adhere to national and international regulations and standards to ensure legal compliance and public safety.
- **Government Policies:** Consideration should be given to policies and incentives that promote the development and adoption of black box technology in vehicles.
- **Data ownership:** there may be dispute over who owns the data generated by the system. The vehicle owner, the manufacturer, or the insurance company. This could lead to legal and ethical challenges.

Social Constraints:

- **Acceptance:** Public perception and acceptance of the black box technology should be considered, and efforts may be needed to educate and address any concerns among the general population.
- **Inclusivity:** The technology should be designed to accommodate users of diverse backgrounds and abilities.

Sustainability Constraints:

- **Resource Conservation:** The production and operation of the black box system should not deplete finite resources and should aim for sustainable practices.
- **Long-term Viability:** The technology should be designed with a focus on longevity and upgradability to minimize electronic waste.

3.3 Engineering Standards

SAE J1979 Diagnostics Test Mode:

Several OBD-II PIDs are defined under SAE standard J1979. A subset of these rules must be supported by all on-road cars and trucks sold in North America, mostly for state-mandated emissions inspections. Additional PIDs unique to each cars are also defined by the manufacturers. Many bikes also accept OBD-II PIDs, however they are not required.

IEEE 802.11 (Wi-Fi) Standards:

The system's remote connectivity options, such as Wi-Fi for data retrieval, adhere to IEEE 802.11 standards. This ensures compatibility with common networking protocols and enhances interoperability

ISO 26262 Functional Safety for Road Vehicles:

In the context of automotive functional safety, ISO 26262 is followed to systematically identify and mitigate risks associated with potential hazards. This standard guides the development of safety features within the system, especially during event recording and incident handling.

IEEE 1471 Recommended Practice for Architectural Description:

IEEE 1471 is followed in documenting the system architecture. This practice assists in creating clear and comprehensive architectural descriptions, aiding in effective communication and understanding of the system's design.

3.4 Methodology

1. Project Definition and Scope:

Clearly define the objectives and scope of the Vehicle Black Box System. Identify the specific features and functionalities we aim to implement, such as data recording, event detection, and connectivity options.

2. Literature Review:

Conduct a literature review to understand existing technologies, and practices related to vehicle black box systems. Explore relevant research papers, articles, and documentation as mentioned in the related works to gather insights into sensor integration, data processing, and communication protocols.

3. Sensor Selection and Integration:

Choose appropriate sensors for the black box system, considering factors such as GPS accuracy, accelerometer sensitivity, gyroscope precision, and compatibility with Arduino. Integrate these sensors into the Arduino platform to capture essential vehicle data.

4. Arduino System Setup:

Set up the Arduino environment, selecting the appropriate Arduino board and configuring the necessary libraries for sensor communication. Develop the initial code structure to read data from sensors and ensure proper communication between Arduino and the sensors.

5. Event Detection Mechanism:

Design and implement an event detection mechanism that triggers recording in response to specific events, such as sudden collisions. Utilize sensor data to define thresholds for triggering events.

6. Camera Integration and Computer Vision:

Select a high-resolution camera compatible with Arduino and integrate it into the system. Ensure synchronization between sensor readings and visual data.

7. OBD-II Integration:

Interface with the vehicle's OBD-II port to gather information about engine performance, fuel consumption, and diagnostic trouble codes. Develop code to communicate with the OBD-II system and retrieve relevant data.

8. Connectivity Options:

Implement connectivity options for remote data retrieval. Choose communication protocol such as GSM for transmitting data to a remote server. Develop the necessary code to establish and manage remote connections.

9. User Interface Design: - Create a user-friendly interface for configuring the system and accessing recorded data. Develop a graphical user interface (GUI) if applicable, ensuring ease of use for both technical and non-technical users.

CHAPTER 4

SIMULATION

Introduction

In order to prepare for final manufacturing, we will produce a prototype to conduct practical experiments and connect the software and hardware together by writing code for an Arduino programme.

4.1 Implementation

This project has been developed with Arduino microcontroller that collects the data from the connected sensors and saves the data on the Micro-SD to analyze. This chapter focuses on the project's implementation and testing phase, which is what needs to be done in order to meet the project's goals and design limitations. The implementation has been done by dividing the whole project into small parts:

1. Simulation using Arduino IDE software.
2. Hardware integration and testing.
3. Prototyping an application.

4.1.1 Connection between GSM/GPRS and Arduino

Instead of connecting to the 5V pin on the Arduino, the GSM/GPRS has a power port that it can use to connect to an external battery. The RX/TX pins connect to the TX/RX pins on the Arduino. Finally, the GND pin on the GSM/GPRS to the Arduino GND.

Table 4.1: Connection between GSM/GPRS and Arduino Mega

GSM/GPRS	Arduino Mega
RX	TX (D18)
TX	RX (D19)
GND	GND

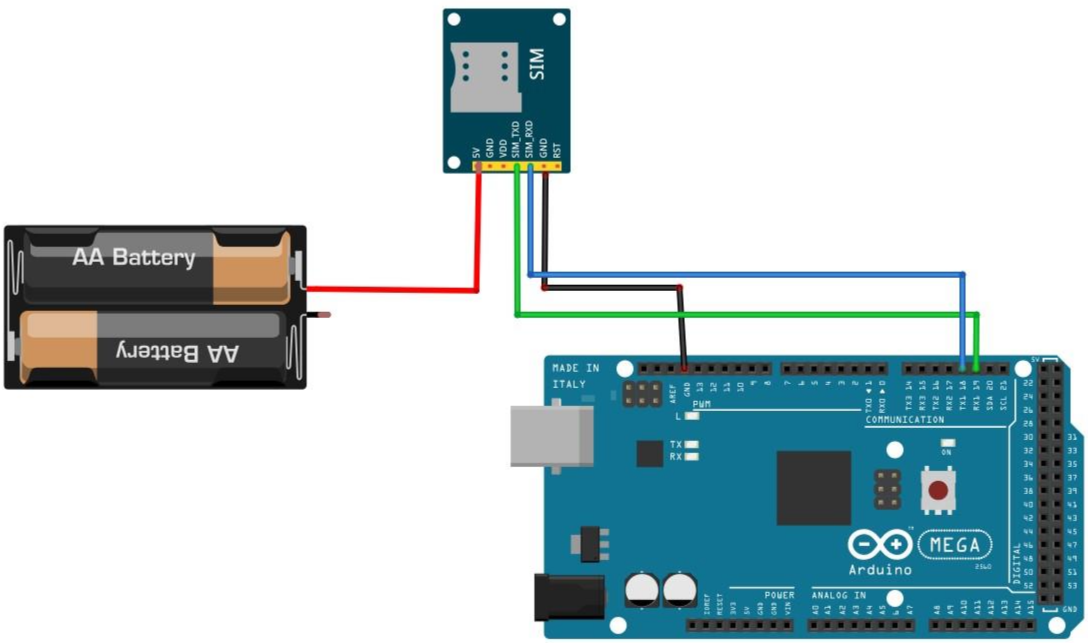

Figure 4.1: GSM/GPRS connection to Arduino Mega by fritzing

4.1.2 Connection between MPU-6050 and Arduino

The MPU-6050 module vcc pin is connected directly to the Arduino Mega's 5V pin. SDA pin and SCA pin is connected to D20 and D21 pin.

Table 4.2: Connection between MPU-6050 and Arduino Mega

MPU-6050	**Arduino Mega**
VCC	5V
GND	GND
SDA	SDA (D20)
SCL	SCL (D21)

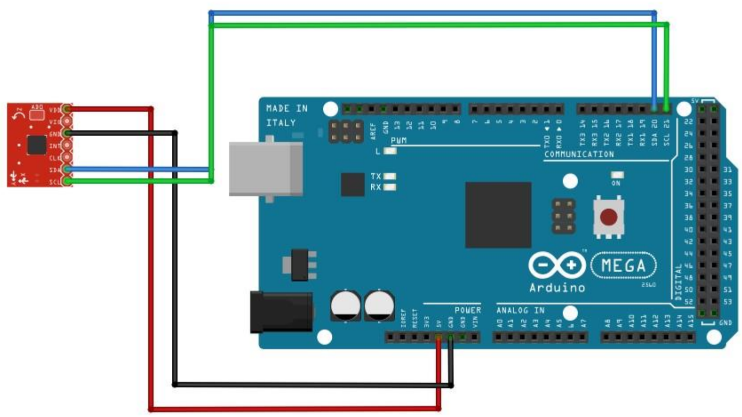

Figure 4.2: MPU-6050 connection to Arduino Mega by fritzing

4.1.3 Connection between Micro-SD and Arduino

Two rows of identical male connectors on the micro-SD card reader allow multiple SPI peripherals to be connected to an Arduino at once, either row can be used interchangeably. In addition, the micro-SD card operates at a voltage varying between 4.5V and 5.5V, for this, we connect the module pin with the 5V pin of the Arduino board.

Table 4.3: Connections between the SD module and the Arduino Mega

Micro-SD	**Arduino Mega**
VCC	5V
GND	GND
CS	Pin 10
SCK	Pin 11
MOSI	Pin 12
MISO	Pin 13

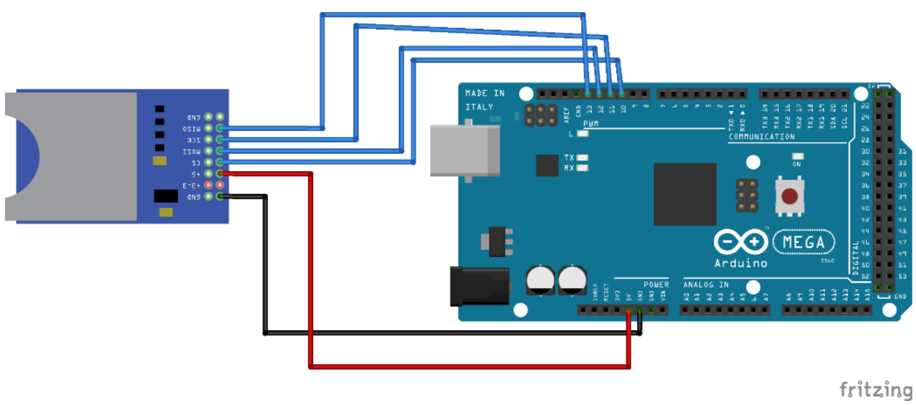

Figure 4.3: Micro-SD connection to Arduino Mega by fritzing

4.1.4 Final Assembly

We will show you the finished assembly, which has the extra OBD-II connection in addition to all the system's previously mentioned parts. The finished system assembly under Fritzing is seen in Figure 4.4.

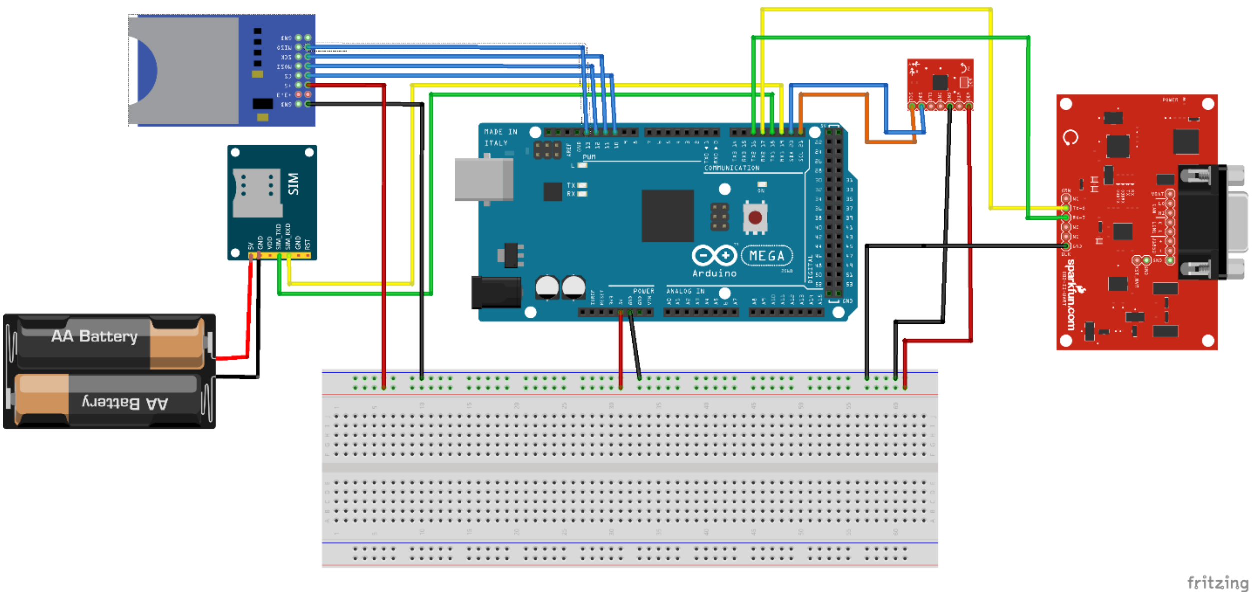

Figure 4.4: Final assembly by fritzing

4.2 Results and Testing

The Arduino IDE programming software will be used to programme the microcontroller so that it can control all of the attached modules. The Arduino IDE platform utilizes the C programming language, which follows specific code guidelines. It provides a software library that contains numerous standard input and output operations, taken from the Wiring project. In order to control and operate the GSMGPRS module and OBD-II module, we installed the SIM808 and ELMDuino library.

4.2.1 Implemetation and results of the GSM/GPRS module

We started with the GSM/GPRS module to test the gps accuracy and availability in our area. The GPS can work umder the roof of buildings and needs a warm startup of 32s to connect to the network, same as the GSM with the 2G network. While testing the module we got the output of the latitude and longtitude with time and date, and even the speed of the vehicle, which could help us replace some of the OBD-II code the includes the speed.Figure 4.5 shows the GPS output.

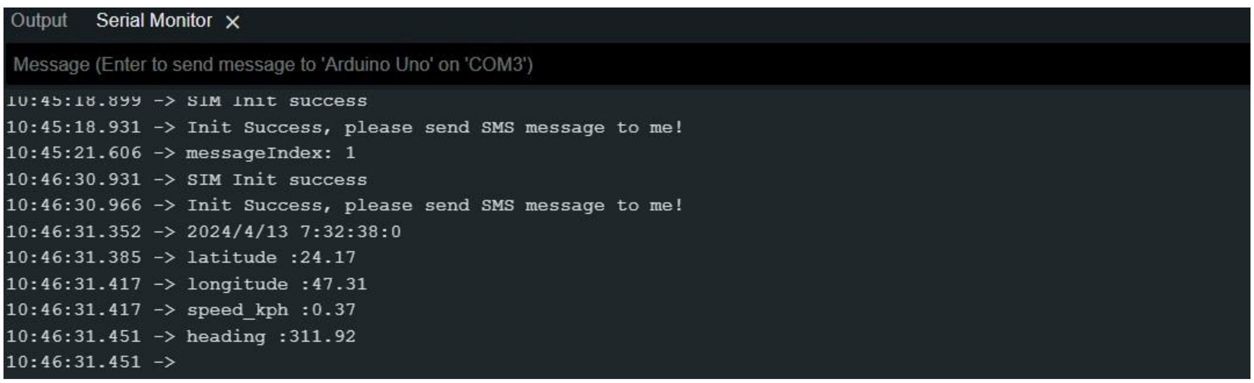

Figure 4.5: Results of the GPS module

We operated the GSM to callup a phone number as soon as a trigger event occurs, which is the motion detection on the Gyroscope. Originally, we planned to have the GSM send the GPS information to the phone number via SMS, but we ran into coding obstacles, in which the function that sends the SMS only takes the data type Char. The GPS function give the longitude and latitude in float wich we couldn't parse as Char to the SMS function. Figure 4.6 shows the wiring and results of the GSM/GPRS module.,

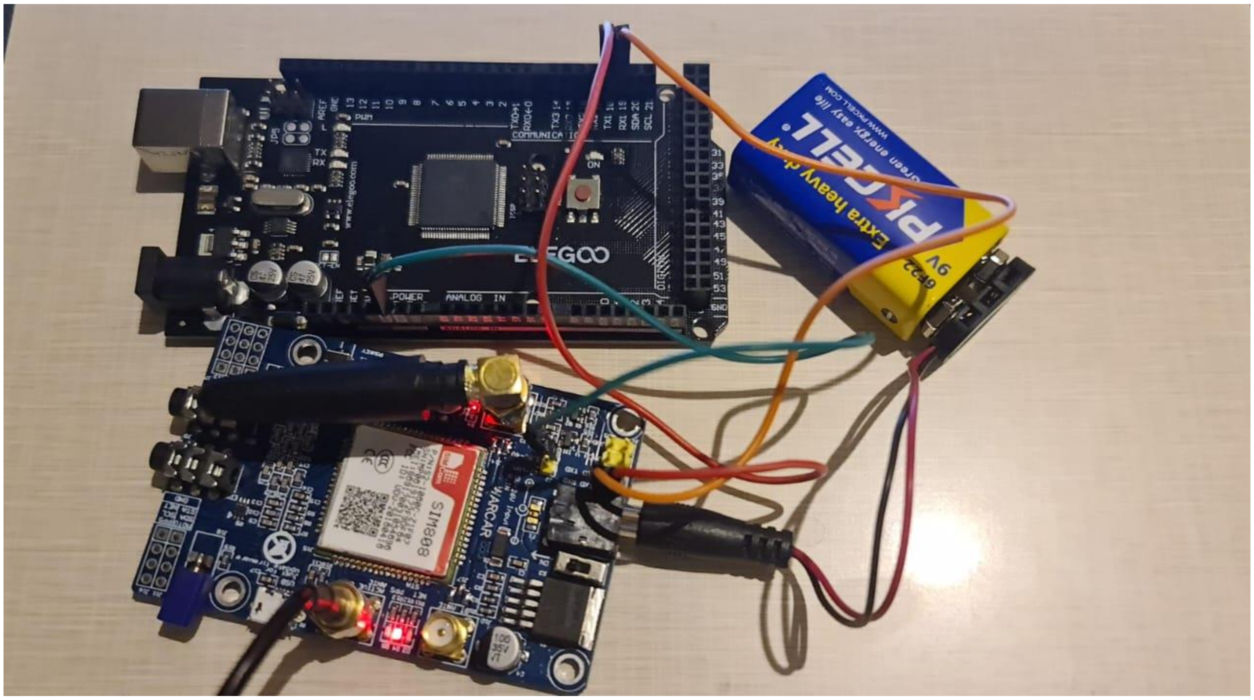

Figure 4.6: Connections between the GSM/GPRS and the Arduino board

4.2.2 Implementation and results of the MPU-6050

The MPU-6050 will serve as our trigger event with motion detection. Motion detection can have a certain threshold to trigger the rest of the sensors, and in our case for the prototype the threshold was set at 20. In real life circumstances 20 is too sensitive for a car since anything can trigger the threshold, like speed bumbs or tight turns. We can also get the angle of the vehicle at the time of the accident, we took the output of the angle before and after the accident to try and calculate the change of the vehicle, and get an estimation on the part of the vehicle's body that got hit. Since the change of the angle given by the MPU-6050 has to be manually calculated, we ended up with empirical data that is not always accurate. Figure 4.7 and 4.8 show the output and connections of the MPU-6050.

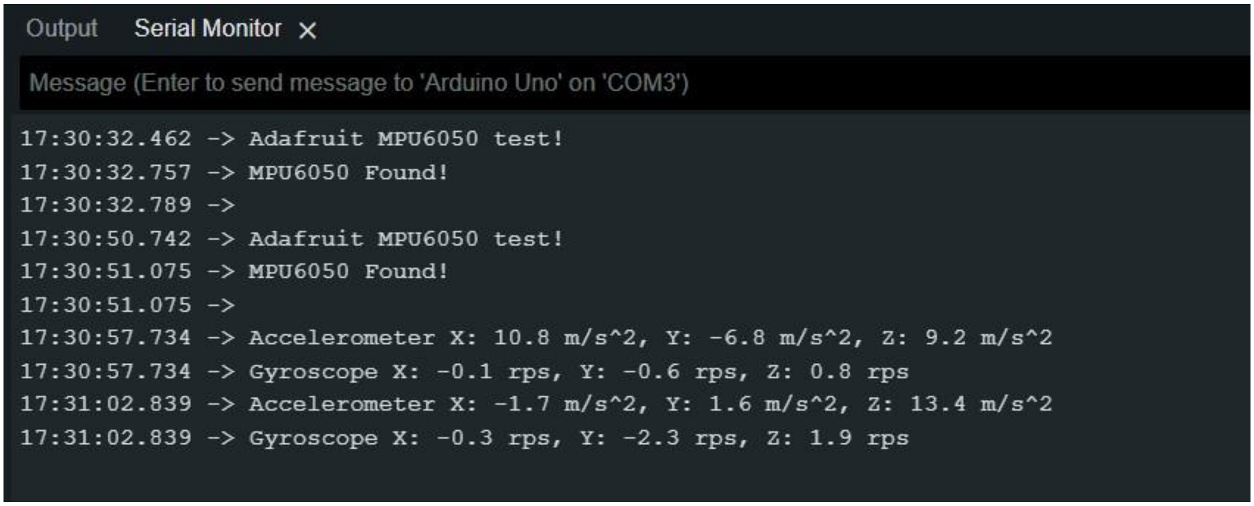

Figure 4.7: Results of the MPU-6050 module

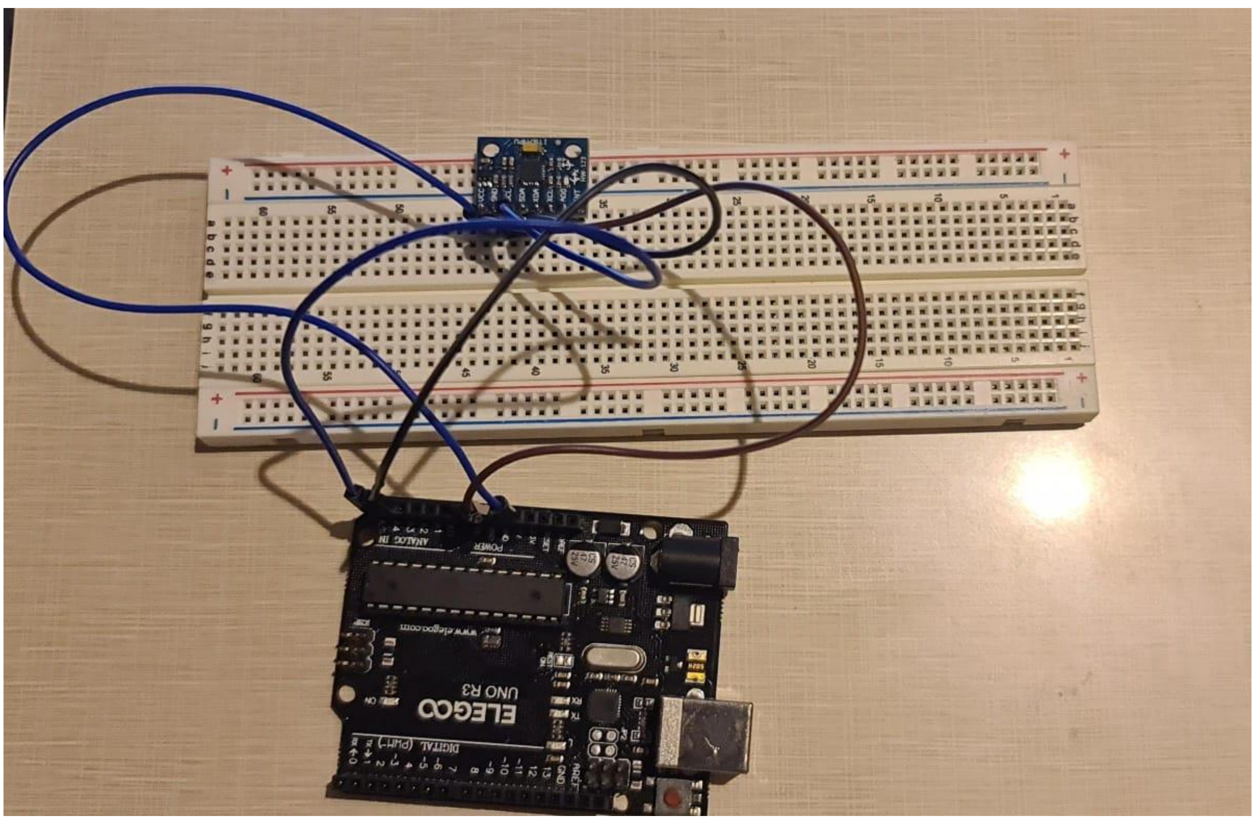

Figure 4.8: Conncetions between the MPU-6050 and the Arduino board

4.2.3 Final Assembly

We assembled the system in a small wooden box to hold all the wiring of the sensors. Figure 4.9 and 4.10 show the outside of the box with the Dash-cam and the wiring inside.

Figure 4.9: Outside view of the box

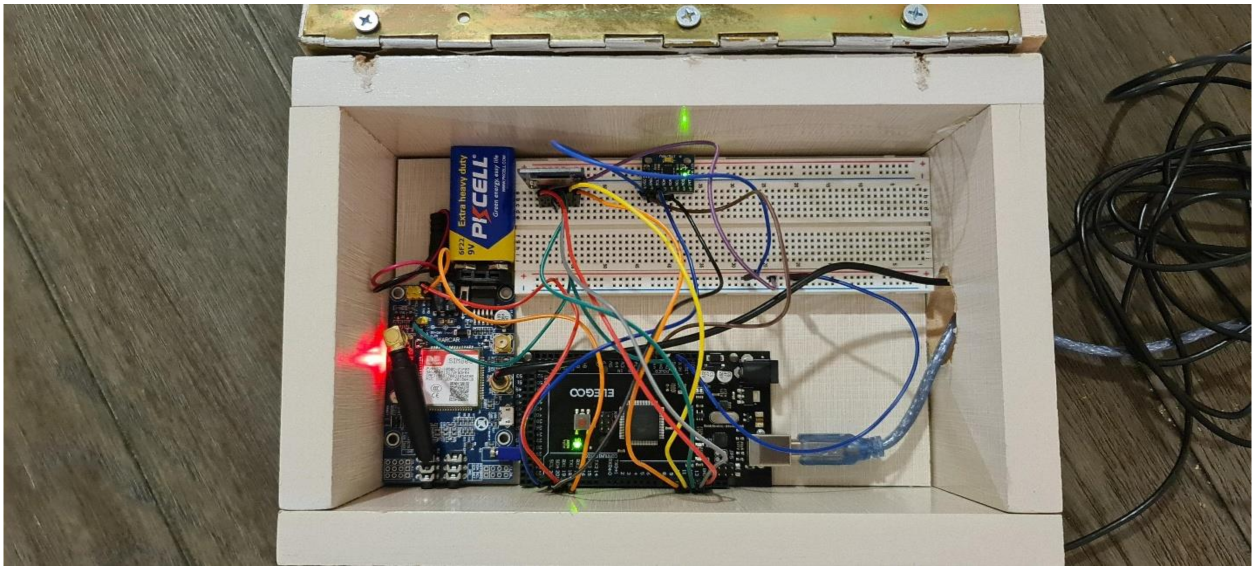

Figure 4.10: Inside view of the box

4.3 OBD-II Application

We designed a prototype application to simulate the data collecting from the OBD-ii port. The app allows the user to connect to the port via Bluetooth and WI-FI from the ELM327 module. The user can register more than one vehicle and track it's location at any time and check the vehicle sensors and performance and clear DTCs, with the user's imformation logged so that he can be contacted once an accident occurs.

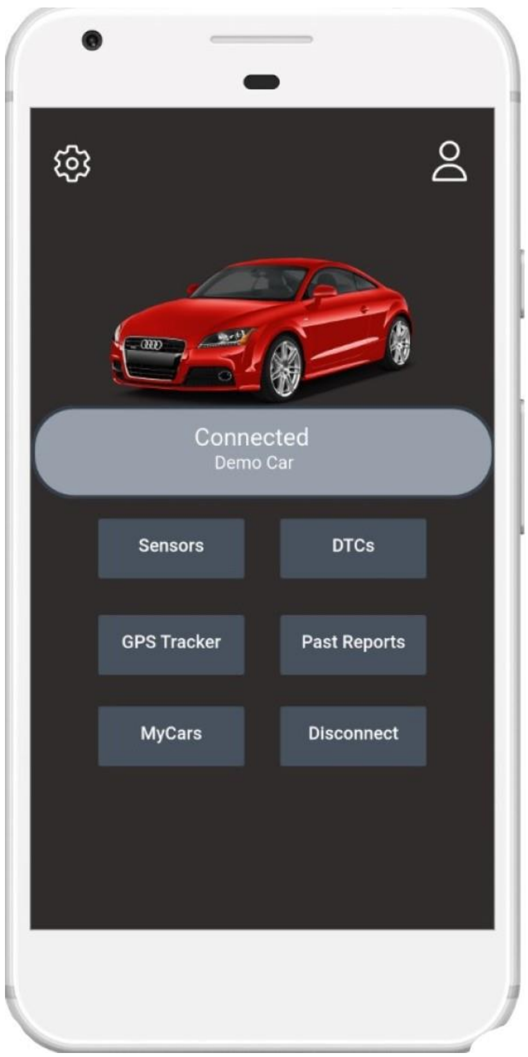

Figure 4.11: Application overview

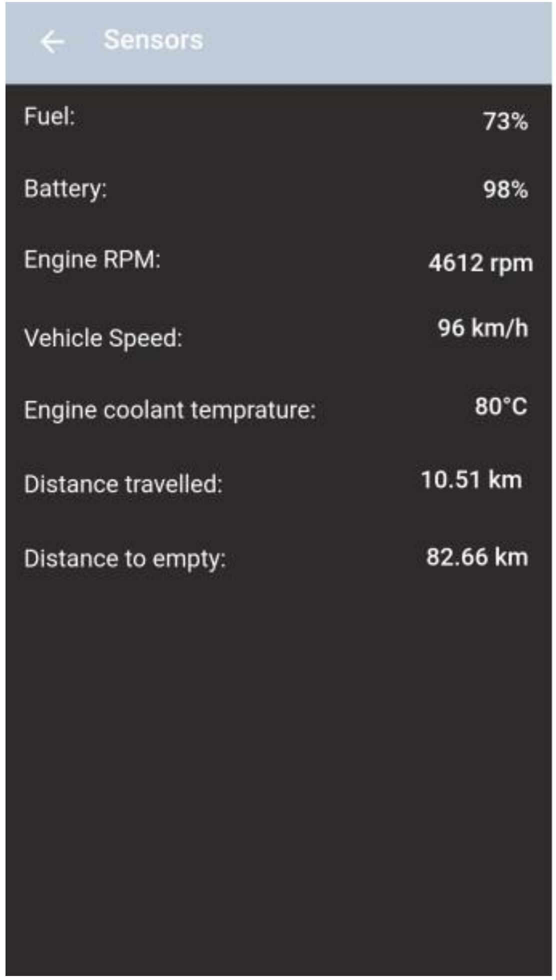

Figure 4.12: Application sensors page

This app is also available for employees of insurance companies as a desktop version. This version of the app provides a secure and efficient way for insurance company employees to respond to accidents promptly and accurately. By limiting access to only necessary information, privacy and security concerns are effectively addressed. Only the user's phone number and their family's phone number, along with the location, are available.

CHAPTER 5

CONCLUSIONS

5.1 Summary

This project aimed to conceive and implement a Vehicle Black Box System, with Arduino technology to create a cost-effective and versatile solution for vehicle data recording and analysis. The system's primary focus was on capturing essential vehicle data, offering a practical and affordable solution applicable to various scenarios such as accident data recording, driver behavior monitoring, and self-reporting accidents. The project initiated with the careful selection and integration of a suite of sensors, including GPS, accelerometer, gyroscope, and an OBD-II interface. Arduino technology played a pivotal role in seamlessly integrating these sensors, facilitating accurate real-time data capture.

The project successfully interfaced with the vehicle's OBD-II port, extracting crucial engine performance, fuel consumption, and diagnostic trouble codes. This additional data provided a comprehensive overview of the vehicle's condition, aiding in understanding the data needed in our project. One of the project's findings, is accurate and comprehensive data capture. The multisensor integration, delivered accurate and comprehensive data capture of the vehicle's performance, location, and condition.

In conclusion, the Vehicle Black Box System developed in this project gives a comprehensive yet accessible solution for vehicular data recording and analysis. Its effective design, integration of various sensors, and remote accessibility make it a versatile and powerful tool for improving road safety, analyzing driving behavior, and providing valuable insights for individual drivers and insurance services companies. The system has the potential to help transform vehicular data analysis and raise the bar for efficiency and safety in transportation systems, as demonstrated by the project's findings.

5.2 Contributions -or- Potential Impact -or- Significance

The successful implementation of the Vehicle Black Box System demonstrates the benefits of integrating advanced technologies such as Arduino, AI, and sensor systems in the automotive industry. It could inspire further innovations in vehicle monitoring and safety. The project has the potential to contribute to a shift in driving culture by promoting awareness of driving habits and safety. The availability of detailed data can encourage responsible driving practices among individuals.

Accurate and real-time data transmitted by the system can enhance emergency response efforts. First responders can receive critical information immediately, optimizing their actions and potentially saving lives in the event of an accident.

The system's comprehensive data, including visual and sensor recordings, aids insurance claim processing. Insurers can access accurate and timely information, leading to quicker and fairer resolution of claims, reducing fraud and operational costs.

The captured data allows for in-depth analysis of driving behavior. This information can be used for driver training programs, and measures to improve overall driving habits and safety.

5.3 Future Work

Security Measures:

Strengthen security measures to protect the integrity of the data, considering encryption, secure authentication, and secure communication protocols.

Real-Time Data Processing Optimization:

Explore ways to optimize real-time data processing on the Arduino platform, aiming for minimal latency and efficient utilization of computing resources.

Power Consumption Management:

Implement power-saving strategies to manage the energy consumption of the system, ensuring efficient use of the vehicle's power source and minimizing impact on battery life.

Durability in Challenging Conditions:

Improve the system's durabilityin challenging conditions, such as strong weather or low visibility, ensuring reliable performance regardless of external factors.

Integration with In-Vehicle Systems:

Explore deeper integration with a vehicle's in-built systems, aiming for smoother communication with onboard computers and improving the overall integration with the vehicle's system.

REFERENCES

[1] www.moh.gov.sa/en/Ministry/Statistics/Pages/Traffic-accidents.aspx.

[2] https://www.sparkfun.com/products/9555

[3]https://www.electronicwings.com/sensors-modules/mpu6050-gyroscope-accelerometer-temperature-sensor-module.

[4]https://howtomechatronics.com/tutorials/arduino/arduino-and-mpu6050-accelerometer-and-gyroscope-tutorial/

[5] https://docs.arduino.cc/learn/starting-guide/whats-arduino

[6] https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/

[7] Rushikesh Milind Puranik "An Analysis and Implementation of Vehicle Black-Box System" January 2019.

[8] Mazher Iqbal "Automatic Vehicle Accident Detection and Reporting with Black Box" January 2016.

[9] Gregor A. Aramice, Abbas H. Miry, Tariq M. Salman "VEHICLE BLACK BOX IMPLEMENTATION FOR INTERNET OF VEHICLES BASED LONG RANGE TECHNOLOGY" March 2023.

[10] Ziad Saeed "Cost Minimization of GPS-GSM Based Vehicle Tracking System" April 2020.

[11] Mohammad Minhazur Rahman "Smart Vehicle Management System for Accident Reduction by Using Sensors and An IoT Based Black Box" Ocotber 2021.

[12] Abdallah Kassem "Vehicle Black Box System" April 2010.

[13] Sethuraman and Santhanalakshmi "Implementing Vehicle Black Box System by IoT based approach" July 2020.

[14] Chetan Patil "Low Cost Black Box for Cars" March 2014.

[15] https://www.arduino.cc/en/software

[16] https://en.wikipedia.org/wiki/OBD-II_PIDs