DESIGN, DEVELOPMENT AND EVALUATION OF A SMART DETECTION SYSTEM FOR UNATTENDED CHILDREN IN VEHICLES

MADIHAH BINTI MOHD FIRDAUS

BACHELOR OF MECHATRONICS ENGINEERING WITH HONOURS
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

DESIGN, DEVELOPMENT AND EVALUATION OF A SMART DETECTION SYSTEM FOR UNATTENDED CHILDREN IN VEHICLES

MADIHAH BINTI MOHD FIRDAUS

A report submitted
in partial fulfilment of the requirements for the degree of
Bachelor of Mechatronics Engineering with Honours

Faculty of Electrical Technology and Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

DECLARATION

I declare that this thesis entitled "DESIGN, DEVELOPMENT AND EVALUATION OF A SMART DETECTION SYSTEM FOR UNATTENDED CHILDREN IN VEHICLES" is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in the candidature of any other degree.

Signature	:	
Name	:	MADIHAH BINTI MOHD FIRDAUS
Date	:	20 JANUARY 2024

APPROVAL

I hereby declare that I have checked this report entitled "DESIGN, DEVELOPMENT AND EVALUATION OF A SMART DETECTION SYSTEM FOR UNATTENDED CHILDREN IN VEHICLES", and in my opinion, this thesis fulfils the partial requirement to be awarded the degree of Bachelor of Mechatronics Engineering with Honours

Signature :

Supervisor Name : FADILAH BINTI ABDUL AZIS
PENSYARAH KANAN
Fakulti Teknologi dan Kejuruteraan Elektrik
Universiti Teknikal Malaysia Melaka

Date : 24 JUNE 2024

DEDICATIONS

To my beloved mother and father

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to everyone who has contributed to the successful completion of this final year project. First and foremost, I extend my heartfelt thanks to my supervisor, Puan Fadilah binti Abdul Azis for her invaluable guidance, unwavering support, and constructive feedback throughout the entire research process. Her expertise and encouragement have been instrumental in shaping the direction of this project.

I am also grateful to the faculty members of the electrical department for providing an enriching academic environment and for fostering a culture of intellectual curiosity. The resources and facilities offered by the department have significantly contributed to the quality of this research.

I extend my appreciation to my classmates and friends who have been a source of motivation and inspiration. Their companionship and shared experiences have added a meaningful dimension to my academic journey all throughout the years.

Last but not least, I want to express my deepest gratitude to my family for their unwavering support, understanding, and encouragement. Their love and belief in my abilities have been the driving force behind my academic pursuits.

In conclusion, I acknowledge the collective efforts and support of all those who have played a role, big or small, in the completion of this project. Your contributions are truly appreciated.

ABSTRACT

Vehicles have become a vital part of modern life, functioning as reliable and efficient means of transportation. Cars, being one of the most widely used types of transportation, serves an important part in facilitating people's daily routines, whether it is commuting to work, conducting errands, or taking leisurely trips. However, tragic incidents occur every year involving vehicles where children are inadvertently left unattended in cars, leading to severe health complications or even fatalities due to temperature extremities. This research aims to develop an alert mechanism that informs guardians or emergency responders promptly and validate the system's accuracy and reliability in various conditions. This project involves designing and validating a detection system that's reliable and efficient, assessing the effectiveness and reliability of current child-in-car detection methods, and developing a mobile alert system to detect children that are left unattended in parked vehicles. This project will design a system that combines multiple sensors and data sources for reliable detection, and by using Internet of Things, the system is able to detect when a child moves in the vehicle and alerts the parents by SMS and phone call. A Smart Child-in-Car Detection System that consists of Arduino microcontroller, GSM module, and sensors can serve as an essential tool to mitigate such incidents by alerting guardians, authorities, or people nearby promptly. This project is expected to reduce the number of incidents where children are left unattended in vehicles by approximately 70-80%, significantly contributing to a reduction in heat-related injuries and fatalities.

ABSTRAK

Kenderaan telah menjadi sebahagian penting kehidupan moden, dimana ia berfungsi sebagai alat pengangkutan yang boleh diharap dan cekap. Kereta merupakan salah satu jenis pengangkutan yang paling banyak digunakan dan berfungsi sebagai bahagian penting dalam memudahkan rutin harian seseorang, sama ada untuk pergi bekerja, menjalankan tugas, atau untuk tujuan berekreasi. Walau bagaimanapun, insiden tragis berlaku setiap tahun yang melibatkan kenderaan di mana kanak-kanak secara tidak sengaja ditinggalkan tanpa pengawasan di dalam kereta, yang membawa kepada komplikasi kesihatan yang teruk atau juga kematian akibat suhu panas. Penyelidikan ini bertujuan untuk mencipta sistem amaran yang memaklumkan penjaga atau pihak kecemasan dengan segera dan juga untuk mengesahkan ketepatan dan kebolehpercayaan sistem ini dalam pelbagai keadaan. Projek ini melibatkan reka bentuk dan pengesahan sistem pengesanan yang boleh dipercayai dan berkesan, menilai keberkesanan dan kebolehpercayaan kaedah pengesan kanak-kanak semasa di dalam kereta, dan mencipta sistem amaran mudah alih untuk mengesan anak-anak yang ditinggalkan tanpa pengawasan dalam kenderaan yang diparkir. Projek ini akan merancang sistem yang menggabungkan pelbagai sensor dan sumber data untuk pengesanan yang boleh digunapakai, dan juga menggunakan 'Internet of Things', sistem ini boleh mengesan apabila seorang kanak-kanak bergerak di dalam kenderaan dan memberi amaran kepada ibu bapa melalui SMS dan panggilan telefon. Sistem Pengesanan Kanak-kanak dalam Kereta yang terdiri daripada mikrokontroler Arduino, modul GSM, dan pelbagai sensor boleh berfungsi sebagai alat penting untuk mengurangkan insiden tragis dengan segera memberi amaran kepada penjaga, pihak berkuasa, atau orang-orang yang berhampiran. Projek ini dijangka dapat mengurangkan bilangan insiden di mana kanak-kanak ditinggalkan tanpa pengawasan dalam kenderaan lebih kurang 70-80%, yang secara signifikan menyumbang kepada pengurangan kecederaan dan kematian akibat haba.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

GSM	-	Global System for Mobile Communications
PIR	-	Passive Infrared
SMS	-	Short Message Service
A.I.	-	Artificial Intelligence
FSR	-	Force Sensitive Resistor
IoT	-	Internet of Things
IDE	-	Integrated Development Environment
Wi-Fi	-	Wireless Fidelity
CPU	-	Central Processing Unit
GPS	-	Global Positioning System
CPD	-	Child Presence Detection
GHz	-	Gigahertz
mAh	-	Miliampere-hour
V	-	Volts
NodeMCU	-	Node MicroController Unit
MIMO	-	Multiple Input Multiple Output
PIC	-	Peripheral Interface Controller
FYP	-	Final Year Project
LED	-	Light Emitting Diode
RX	-	Receive
TX	-	Transmit
VCC	-	Voltage Common Collector
RST	-	Reset
GND	-	Ground
ADC	-	Analog to Digital Converter
PWM	-	Pulse Width Modulation

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

Vehicles have become a vital part of modern life, functioning as reliable and efficient means of transportation. Cars, being one of the most widely used types of transportation, serves an important part in facilitating people's daily routines, whether it is commuting to work, conducting errands, or taking leisurely trips [1]. Moreover, over the last decade, vehicles have gotten much faster and smarter, and with these advances has come the development of innovative cars safety measures.

Cars in particular have quite a few systems involved, one of it is the anti-locking braking system (ABS) which functions to detect the rotational speed of the vehicle's individual wheels and release hydraulic fluid when the wheel rotates too slowly. Other than that, cars have a lighting system where they must have three types of lights which are brake lights, turn indicators, and headlights. Moreover, many modern vehicles are now equipped with blind spot warning systems where it can detect vehicles or objects that enter the driver's blind spots and assist the driver in avoiding collisions [2].

Nowadays, cars also have a smart detection system which involves a smart sensor. It is a device that collects input from the physical world and utilises built-in computational resources to perform predetermined operations when certain input is detected, and then processes data before carrying it on [3]. In addition, a smart detection system can also be referred to the system's ability to use advanced technology, such as sensors, cameras, and artificial intelligence, to intelligently detect and respond to situations that it is programmed to. It also often implies the integration of various sensors and algorithms that enable the system to make informed decisions or take specific actions based on the data it collects [4].

1.2 Motivation

The serious dangers related with leaving a child unattended in a car warrant the use of a child detection system in unattended cars. Unfortunately, occurrences of children being mistakenly left in vehicles can have serious implications, and these incidents have been occurring quite often for the past few years.

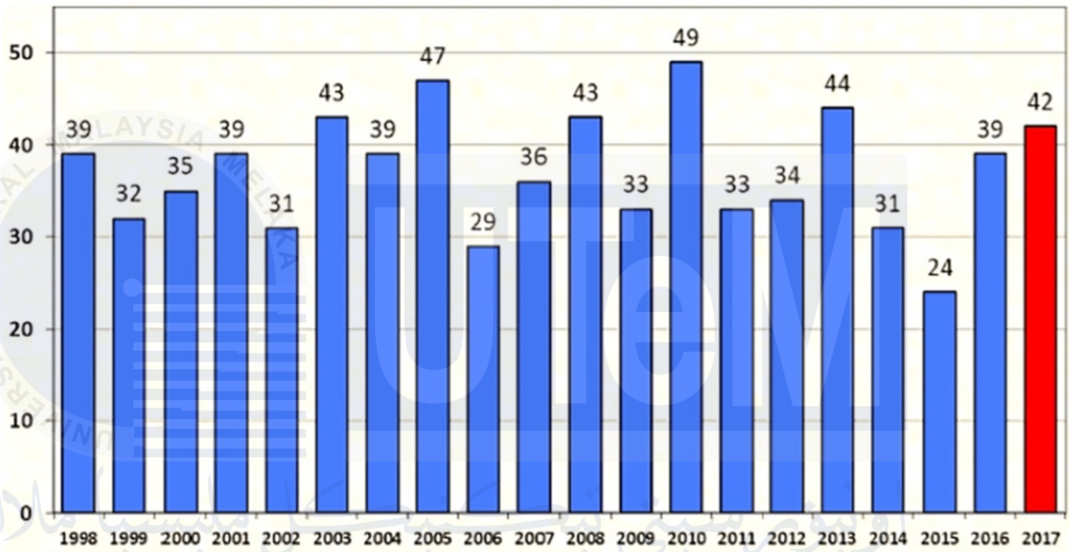

Figure 1.1: Child deaths in USA for heat stroke in cars from 1998 to 2017 [5]

As seen in Figure 1.1, an increasing number of incidents involving children being left unattended in cars by their parents have occurred in the United States in recent years. Unfortunately, the same findings can be found for countries with high stress levels. During these circumstances, the vehicle's internal temperature can typically reach abnormally high or low levels, resulting in hypothermia and other hazardous consequences that might result in a child's death. In the United States, around 37 children die each year from heat stroke or hyperthermia in cars, where most of them are at the ages of one and three [5]. Hence, these incidents have increased the demand for innovative technologies capable of detecting the presence of a child in an unattended car using suitable sensors and alerting the driver or other possible contacts with various kinds of alarm mechanisms.

In Kuala Lumpur (2023), a heartbreaking incident occurred recently in which an eight-month-old newborn girl tragically passed. She was left in a car in a hospital parking lot for nearly 10 hours. The mother, a doctor at the hospital, had placed her kid in the back seat of the car with the intention of driving her to a daycare. The mother, on the other hand, drove to work, presuming that the child had already been dropped off at the centre. The realisation came later that evening, when her husband informed her that the child had not been taken to daycare [6]. In Sungai Petani (2021), A three-year-and-11-month-old boy passed away after being left inside a car for more than four hours in front of a kindergarten in Taman Ria, Sungai Petani. The principal had picked up the young child, who lived near the kindergarten, earlier in the morning to take him to kindergarten. He was unfortunately left in the back seat of the car. When the principal discovered what had happened, he promptly called an ambulance and attempted cardiopulmonary resuscitation (CPR). Despite their attempts, the child remained unresponsive. The child was rushed to the hospital, where he was tragically pronounced dead [7]. In Kuantan (2020), Annur Jannatul Syuhada Mohd Khalil, a nine-month-old daughter, passed away of heatstroke after being left unattended in the back seat of a multipurpose vehicle for roughly four hours. Around 1pm, tragedy struck at the Tun Abdul Razak compound in Indera Mahkota. The baby was discovered unresponsive in her child seat, prompting immediate rescue operations. She was quickly evacuated to the Indera Mahkota health centre, but despite the best efforts of the medical staff, she was declared dead upon arrival [8].

Hence, these tragic cases have motivated the development of this project because a smart detection system can significantly decrease the number of accidents caused by children left unattended in vehicles. The system provides real-time alerts to parents, caregivers, or authorities, allowing for prompt intervention and averting potentially dangerous situations.

1.3 Problem Statement

Every year, tragic incidents occur when children are inadvertently left unattended in vehicles, leading to severe health complications or even fatalities due to temperature extremities. A Smart Child-in-Car Detection System can serve as an essential tool to mitigate tragic incidents by alerting guardians or authorities promptly. This research aims to design and validate such a system that's reliable and efficient. However, despite the many potential dangers regarding unattended children in vehicles, many vehicles lack an efficient and integrated detection system that alerts occupants in real time when the child is left unattended accidentally. While some newer, more advanced, and more expensive cars include such features, the majority of conventional cars do not, highlighting a significant gap in the market. This stresses the urgent need for a cost-effective, universally compatible solution to prevent such avoidable tragedies. Without widespread implementation, the risk to children remains unacceptably high. Therefore, developing an affordable and easily integrated system is crucial for enhancing vehicle safety standards.

1.4 Objectives

1. To design a system that combines multiple sensors and data sources for reliable detection.
2. To develop an alert mechanism that informs guardians responders promptly.
3. To validate the system's accuracy and reliability in various conditions.

1.5 Scope

1. Child Detection Capability: Develop a robust system capable of smart detection for unattended children in vehicles. The system will utilize sensor fusion to accurately identify the presence of a child.
2. Alarm and Notification System: Include an alarm system to alert nearby people and a SMS notification system to notify guardians of unattended children left in vehicles, ensuring rapid safety response.
3. Presence Monitoring: Implement a pressure sensor and motion sensor to detect when children are in a vehicle and when the driver is not present.
4. Testing and Evaluation: Conduct thorough testing in controlled environments to validate the accuracy and reliability of the detection system. Evaluate the system's performance under various environmental conditions (temperature, humidity, lighting, vibration, and electromagnetic interference) to ensure consistent operation and effectiveness.
5. Technical Specifications: Ensure the system operates with a low power consumption design using SIM800L GSM module, Arduino Uno microcontroller, FSR sensor, PIR sensor, and batteries. It also has a wide

detection range, and integrates seamlessly with existing vehicle electronics for easy installation and maintenance.

1.6 Limitations

1. Environmental Factors: System performance is affected by environmental factors such as extreme temperatures, humidity and other atmospheric conditions, which can impact the accuracy of children and driver detection. This experiment was only tested during the time when parents or guardians leave for work, which is around 7:00 am to 10:00 am and when they return home, which is around 6:00 pm to 9:00 pm.
2. Detection Accuracy: System may have limitations in achieving 100% accuracy due to factors such as visual obstructions or sensor limitations.
3. Area of the environment: Areas where the environment is tested by this system is in a conventional Axia car with measurement of 3.64m x 1.63m x 1.53m (Length x Width x Height). Sensor accuracy and system readings may not be accurate when outside of this system specifications as the system is only able to be tested in the type of car that is usually driven by parents with children.
4. Motion Detection Limitation: The motion sensor can only detect the movement of the baby. Therefore, the device might not be able to detect the presence of a baby if the baby is not moving or is sleeping, which could limit the effectiveness of the system.

1.7 K Chart

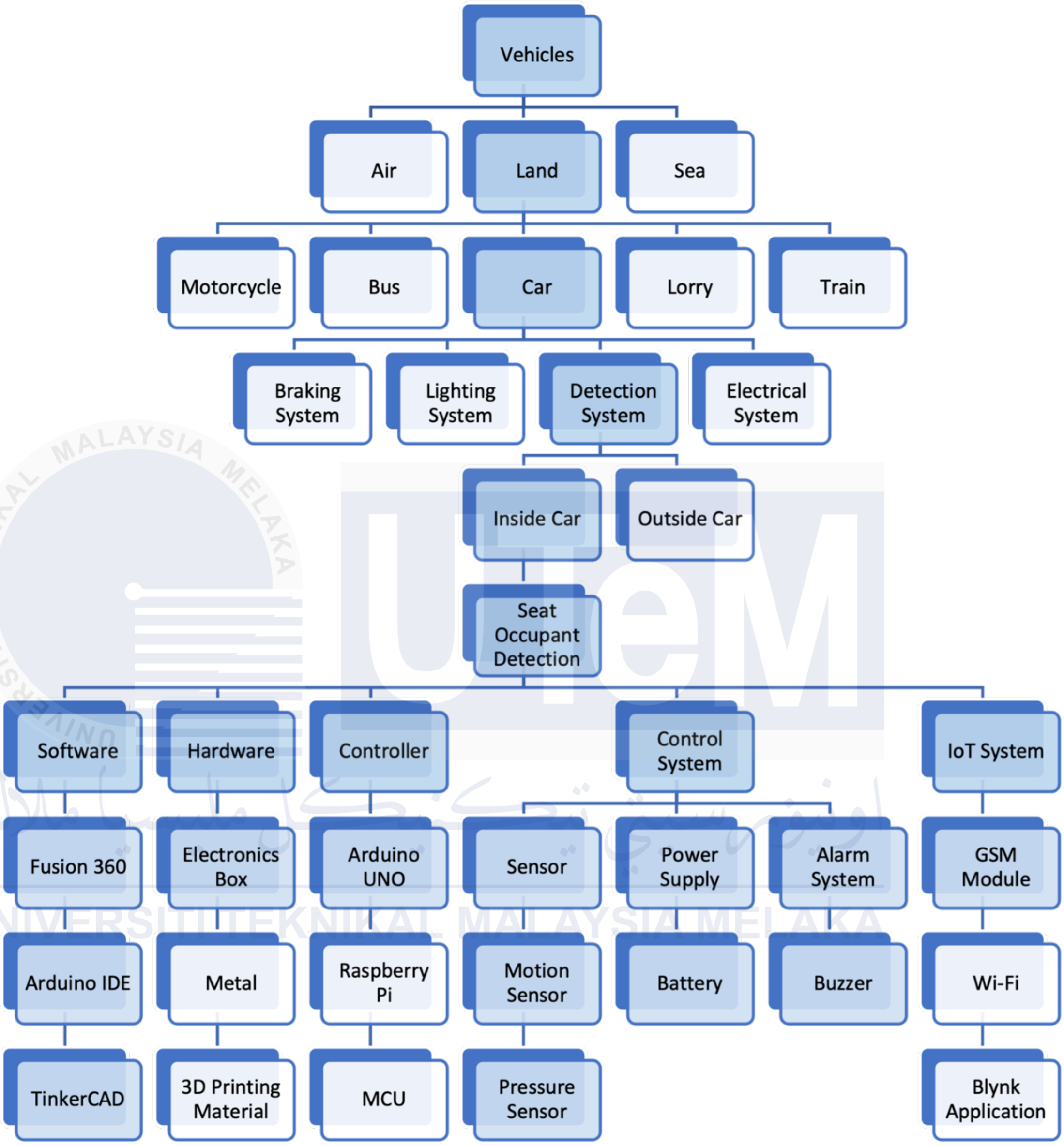

Figure 1.2: K-Chart

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter summarizes and discusses a variety of research or journal publications relevant to the project, including software, mechanical systems, and control systems, based on the K-chart in Chapter 1.

2.2 Overview of Smart Detection System for Unattended Children in Vehicles

A smart detection system for unattended children in vehicles were developed due to incidences involving the deaths of children who have been left unattended in vehicles that continue to occur. The system is meant to prevent these tragic incidents from occurring which involves children's deaths from heat stroke or hypothermia. The major goal of the system is to construct a full system that can connect with humans via Internet of Things where the system can be used to engage and communicate with the module that can be used to transmit and receive notifications depending on which the user performs appropriate actions. As for the microcontroller, it serves as the core of the entire controlling system. It will connect with each component such as the IoT module, battery, resistors, and sensors to detect the presence of children and absence of driver. The device needs to be able to detect movements made by a person as well as the weight of the driver and passenger within the vehicle. Then, the device would have to have a notification and alarm system to alert the parents and people nearby. Moreover, a system such as this one that has been established and equipped with advanced technology is predicted to continue to increase with simultaneous change in time. It is anticipated that the system would be able to solve the problem of accidents involving children who are frequently left unattended in cars [9].

2.3 Overview of Microcontroller-based Seat Occupancy Detection System

A microcontroller is a small integrated circuit that controls a specific operation in an embedded system. On a single chip, a typical microcontroller has a CPU, memory, and input/output (I/O) peripherals. Microcontrollers, also known as embedded controllers or microcontroller units (MCU), are found in a variety of equipment such as cars, robotics, office machinery, medical devices, mobile wireless transceivers, vending machines, and household appliances. They are basically modest little personal computers (PCs) meant to control minor aspects of a bigger component without the need for a complex front-end operating system (OS) [10].

In previous case studies, there are several types of microcontrollers that were used to develop the system involving a few numbers of sensors. Firstly, one of the previous case studies was a development of a project titled 'Low-cost Bus Seating Information Technology System' which uses Raspberry Pi microcontroller to operate the seat occupancy detection system. The Raspberry Pi is a small, single-board computer that was launched in 2012. Since then, its tremendous processing capabilities and small size, combined with its affordable cost, expansion opportunities, and low power consumption, have made it extremely popular among developers. The system which uses the microcontroller is able to determine the vehicle occupancy when the vehicle is in service where it detects events in the vehicle using a variety of sensors like infrared sensors which detects when passengers enter or exit the bus, force-sensitive resistors that detects seat occupancy, a Global Positioning System shield paired with the Raspberry Pi microcomputer which allows for real-time bus tracking, and a USB camera connected to the same Raspberry Pi for cross-checking and validating the data obtained. The data is then uploaded to an online IoT platform called Thinger.io, by using 3G or 4G if available, and the data collected can be viewed using both an Android app and a desktop computer user interface. The block diagram of the system using Raspberry Pi microcontroller is as shown in Figure 2.1 [11].

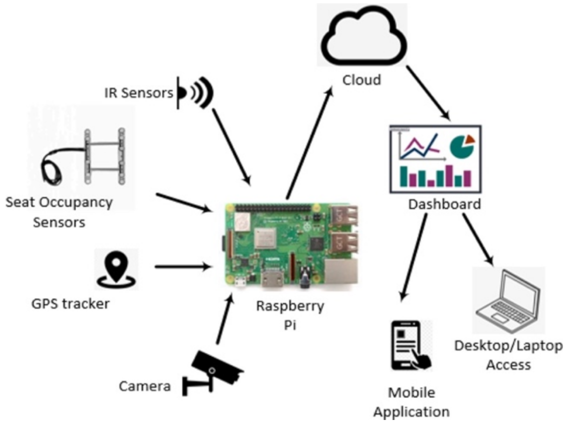

Figure 2.1: Block Diagram of the Seat Occupancy Detection System using Raspberry Pi Microcontroller [11]

Secondly, another case study developed a project titled 'Sensors-Based Automatic Human Body Detection and Prevention System to Avoid Entrapment Casualties inside a Vehicle', using Arduino Mega 2560 microcontroller. The case study develops a system that involves human body detection and prevention system for cars which is enabled by the Internet of Things. It is used to prevent injuries when a person is inside a vehicle that is completely closed. The Arduino Mega 2560 board is utilised in this setup. This system is built around two sensors: a PIR sensor and an ultrasonic sensor. An Arduino Mega 2560 is connected to the vehicle's power source and a buzzer and a 5V relay are also attached to the Arduino Mega 2560, which is programmed to run the entire system. When the car receives an alarm message, all of its indicators begin to blink. A GSM module is linked to the Arduino Mega 2560, which sends the message and makes the call to the designated number [12]. The Arduino Mega 2560 microcontroller diagram is as shown below in Figure 2.2 [13].

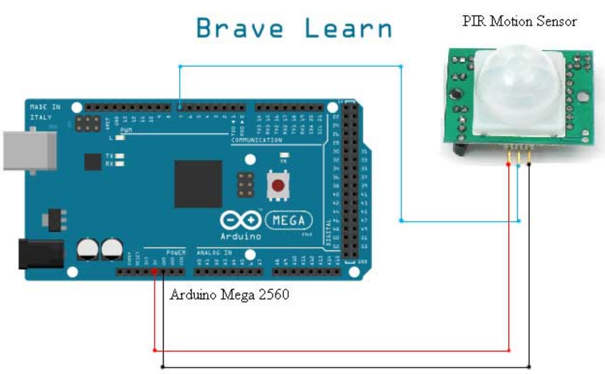

Figure 2.2: Diagram of the Arduino Mega 2560 Microcontroller [13]

Lastly, another case study developed a project titled 'The Development of Detection System for Bus Passenger Monitoring' where it uses Arduino Uno R3 as its microcontroller. The project involves a device that can provide real-time data on bus passengers and overhead storage compartment contents. It improves bus operations and consumer convenience by providing real-time data on capacity and vacant seats. The project prioritizes user-friendliness, ease of installation, excellent quality, understanding, and cost. The Bus Passenger Detector project is a useful tool for monitoring passengers and items on a bus. The Arduino Uno R3 microcontroller processes sensor signals and controls output devices including LCD, LED, and buzzer. The system can connect with a central server to provide real-time information on seat availability, passenger items of value, and other important data. The block diagram of the system using Arduino Uno microcontroller is as shown in Figure 2.3 below [14].

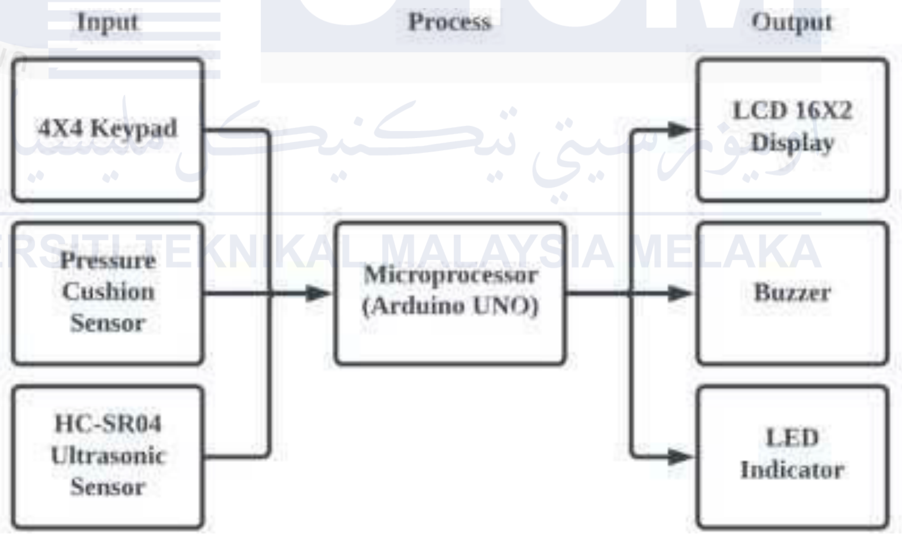

Figure 2.3: Schematic Diagram of the Child Detection System using PIC Microcontroller [14]

2.3.1 Summary of Microcontroller-based Child Detection System

Table 2.1: Summary of Microcontroller-based Seat Occupancy Detection System

Journal Author	**Murdan et al. [11]**	**Arya and Sharma [12]**	**Amin et al. [14]**
Microcontroller used for seat occupancy detection	Raspberry Pi	Arduino Mega 2560	Arduino Uno R3
Advantages	• Powerful processing capability. • Small size. • Affordable. • Has Wi-Fi and Bluetooth support. • Open-source platform.	• Powerful microcontroller board. • Large number of input/output pins. • Higher memory capacity than other Arduino boards. • Cost-effective. • Open-source platform.	• Compatible with wide range of sensors. • Open-source platform. • Cost-effective. • Easy to use. • Convenient. • Real-time data processing. • Decent size for simple projects.
Disadvantages	• Limited in its input/output pins. • Higher power consumption compared to traditional microcontrollers. • Complexity of programming and setup.	• Limited RAM and Flash memory. • Not optimized for low power consumption. • Relatively large microcontroller board.	• Limited processing power and memory capacity. • Less input/output pins compared to Mega. • Limited flash memory.

Based on Table 2.1 where the previous case studies have been reviewed and summarized, there are a few criteria and methods that are suitable to be implemented in this project in order to achieve the desired objectives. The comparisons that need to be discussed include the controller that needs to be used for child detection, the control system, the sensors to detect the seat occupancy, IoT module for the controller, and method of analysis. In terms of the controller used for child detection system, journal [11] used Raspberry Pi microcontroller, journal [12] used Arduino Mega 2560 and journal [14] used Arduino Uno. All three journals have their pros and cons in operating as the microcontroller for each project. When comparing all three microcontrollers, it can be concluded that the Arduino Uno microcontroller is the most appropriate one to be used which journal [14] utilizes in their project. Although it is similar to journal [12] due to both of them using Arduino boards, however the Arduino Mega board is bigger in size as compared to the Arduino nano due to its many inputs and outputs pins. For this project, Arduino Uno is already sufficient to be used for this child detection system project as it is smaller in size, convenient, low cost and easy to use.

2.4 Seat Occupant Detection in Vehicles using Motion Sensor

A motion sensor, often known as a motion detector, is a type of electrical device that detects moving people or objects. Motion sensors are an integral part of any security system [15]. There are two types of motion sensors which are active and passive. Active sensors have both a transmitter and a receiver. This sort of sensor detects motion by detecting variations in the quantity of sound or radiation that reflects back to the receiver. When an item disrupts or affects the sensor's field, an electric pulse is transmitted to the embedded computer, which then interacts with the mechanical component. The most popular form of active motion detector employs ultrasonic sensor technology, which produces sound waves to detect the presence of objects. Microwave sensors emit microwave radiation, while tomographic sensors transmit and receive radio waves [16].

As for the previous case studies, there are several types of sensors that were used to develop the system. Firstly, one of the previous case studies was a development of a project titled ‘IoT based Approach to Estimate the Vacant Seats Available’ where they used PIR sensor and ultrasonic sensor. This study aims to find out if a seat is occupied or not by combining sensors such as a Passive Infrared Sensor (PIR) and an Ultrasonic Sensor paired to a Node MCU microcontroller. The sensor output is analyzed, and the vacancy/occupancy of each seat is estimated. PIR sensors are utilized because they can detect changes in heat emission from the body and thus recognize only living objects which are people. Thus, seats with luggage can be considered empty. The ultrasonic sensor generates sound at an ultrasonic frequency and detects the object after receiving reflection from the object in front of it. The time interval between emission and detection is utilized to calculate the distance between the sensor and the object in front. Thus, the combination of two sensors gives precise data. This data is transferred to Google Firebase's server and stored in Real Time Database format, allowing the user to retrieve it via web or mobile app. The Node MCU microcontroller has an ESP8266 Wi-Fi chip that allows you to send and receive data from the internet. This project's mobile app is built with Google's "Flutter" Software Development Kit (SDK). The block diagram of the system using PIR sensor and ultrasonic sensor is as shown in Figure 2.4 below [17].

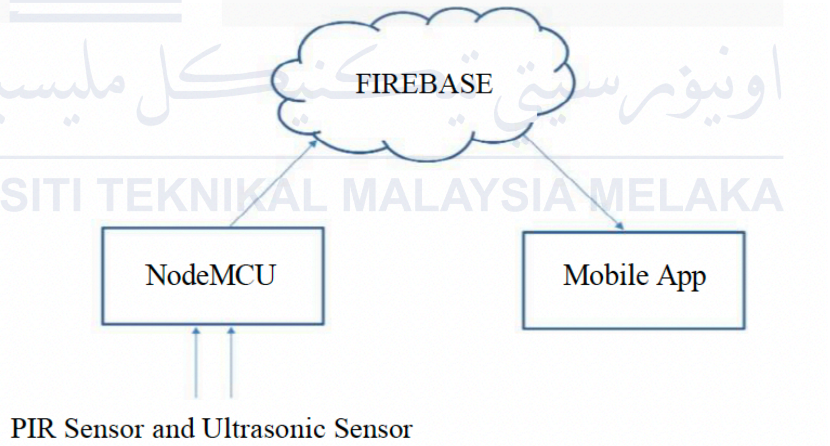

Figure 2.4: Block Diagram of Child Detection System using PIR sensor and FSR Sensor [17]

Secondly, in another case study, there was a development of a project titled 'Car Occupancy Detection using Ultra-Wideband Radar' involving an ultra-wideband (UWB) as its motion sensor. The case study develops a system that can detects the breathing movements of individuals. The pinhole channel model is used to represent the target signal and multipath propagation within the vehicle. The Kronecker product of channel and breathing motion vectors accurately describes the received signal. The covariance may be calculated using the delay profile of the channel and the power spectrum of chest motion. An estimator-correlator is used for detection. Measurements confirm structural assumptions for the received signal, while Monte-Carlo simulations evaluate detection performance [18]. The UWB radar device diagram used as the motion sensor as shown in Figure 2.5 [19].

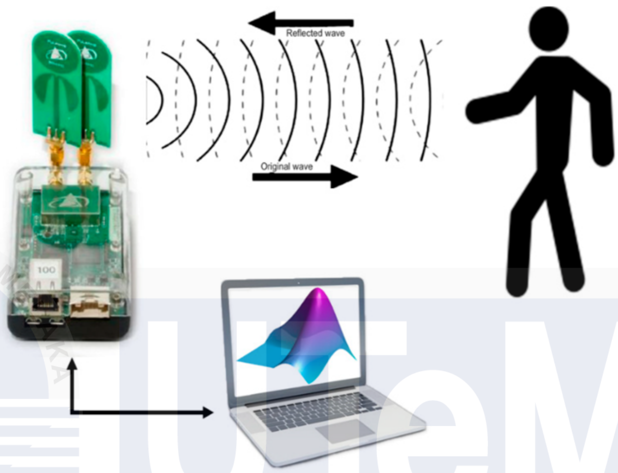

Figure 2.5: UWB Radar Device Diagram used as the Motion Sensor [19]

Lastly, another case study developed a project titled ‘In-Vehicle Occupant Detection based on Zone Source Feature using MM-Wave MIMO Radar’ where the system used a Multiple-Input Multiple-Output (MIMO) millimeter wave radar-based sensors as its motion sensors. They proposed this approach which can detect in-vehicle occupants based on zone source features. The zone source feature is retrieved from a super-resolution range-azimuth (RA) map created using the multiple signal classification (MUSIC) technique. Experiment data shows that the zone source feature follows a normal distribution with minimal overlap, regardless of seat state (absence or occupancy). Setting an adequate threshold can result in a low false alarm rate (1%) while maintaining a high detection rate (99.09%). The suggested technique is built on an integrated digital signal processor (DSP) and tested in real-time stationary and driving scenarios. The substantial accuracies demonstrate its durability and efficiency. The block diagram of the system using MIMO millimeter wave radar-based sensor as the motion sensor is as shown in Figure 2.6 [20].

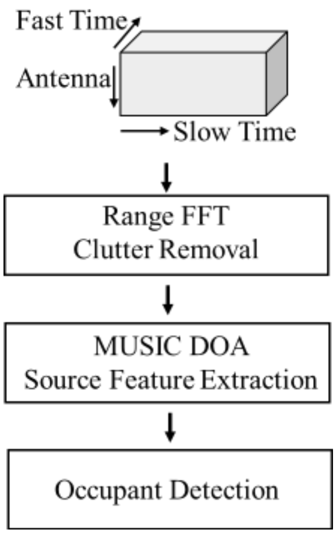

Figure 2.6: Block Diagram of Occupancy Detection System using MIMO Millimetre Wave Radar-based Sensor [20]

2.4.1 Summary of Seat Occupant Detection in Vehicles using Motion Sensor

Table 2.2: Summary of Seat Occupant Detection in Vehicles using Motion Sensor

Journal Author	**U. S et al. [17]**	**Moderl et al. [18]**	**Tan and Yang [20]**
Motion sensors used for seat occupant detection	PIR sensor and Ultrasonic sensor	Ultra-Wideband Radar sensor	MIMO Millimeter Wave Radar Sensor
Advantages	• Detect changes in thermal radiation. • Detects humans while ignoring other objects. • Detects reflection from objects and able to determine the distance. • Affordable.	• Not impacted if there is obstacle covering the occupant. • Improved detection rate compared to others. • Adaptable to different car models. • Versatile and effective.	• Environmental robustness. • Precise detection. • Low-cost. • Wide range of applications.

Disadvantages	• Limited range and detection angle. • Susceptible to false triggers. • Environmental factors can affect the accuracy.	• Complex signal processing. • More expensive due to its hardware components. • Consume more power.	• Limited penetration. • Limited detection range. • Complex signal processing.

Based on Table 2.2 where the previous case studies have been reviewed and summarized, there are a few criteria involving the sensors used for occupant detection system that are suitable to be used for this project for child detection system. From the research that has been done, journal [17] used PIR and ultrasonic sensor, journal [18] used ultra-wideband radar sensor and journal [20] used MIMO Millimeter wave radar sensor. All three journals are able to detect motion successfully although each and every one of them have their own pros and cons. When comparing all three journals' sensors, it can be concluded that PIR and ultrasonic sensor is the most appropriate one to be used as although it has limited range of detection angle, however the size of a conventional car that is commonly used by parents with children is within the range, and the sensors are affordable to be used and journal [17] utilizes both sensors in their project. For this child detection system project, only the PIR sensor is chosen to be used as motion sensor as it is already sufficient enough without the need of the ultrasonic sensor.

2.5 Seat Occupant Detection in Vehicles using Pressure Sensor

Pressure sensors are devices that detect and measure pressure. In this situation, pressure is defined as the amount of force applied to an area. Pressure sensors enable more specialized maintenance approaches, such as predictive maintenance. These devices capture immediate data about the state of equipment. Sensors may predict and prepare for failure patterns based on the information available to them. Pressure sensors operate by monitoring the physical change that occurs in response to pressure changes. After measuring the physical changes, the data is converted into electric signals. These signals can then be shown as useful information for the team to interpret.

Pressure sensors operate by monitoring the physical change that occurs in response to pressure changes. After measuring the physical changes, the data is converted into electric signals. These signals can then be shown as useful information to be collected and analyzed [21].

As for the previous case studies, there are several types of sensors that were used to develop the system. Firstly, one of the previous case studies was a development of a project titled 'Smart Occupancy Monitoring System using IoT Technologies for South African Minibus Taxis' using Force Sensitive Resistor sensor as its pressure sensor. This study proposes an IoT-based smart occupancy monitoring system (IoT-SOMS) prototype that can detect seat occupancy, calculate distance travelled, and charge fees in real time. The developed prototype incorporates a Force Sensitive Resistor (FSR) to identify occupancy, a proximity sensor to detect the presence of passengers and avoid overloading, an Ultrasonic sensor to compute distance, GPS to track the location of the minibuses, and Arduino Uno as an IoT gateway. The block diagram of the system using FSR as the pressure sensor is as shown in Figure 2.7 below [22].

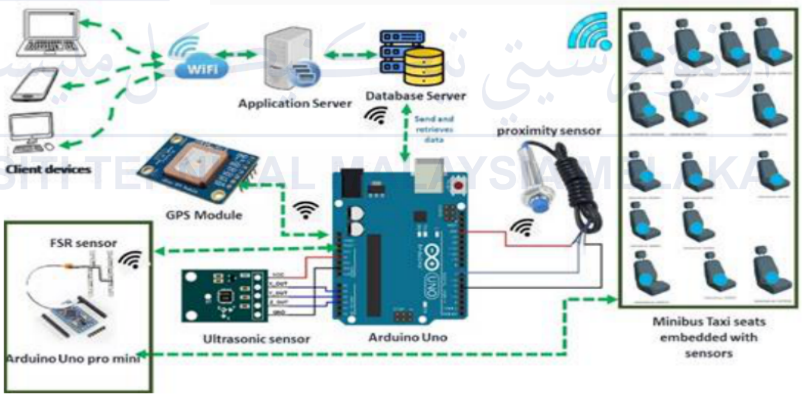

Figure 2.7: Block Diagram of Occupancy Detection System using FSR Sensor [22]

Secondly, in another case study, there was a development of a project titled 'Smart Bus: Seat Occupancy Detection System' which used pressure pad sensors. Designed and implemented a system that combines hardware and a web application for passengers to verify bus seat occupancy and status. This can be done via linked gadgets or display systems at the bus entry point. The prototype solution effectively reduces seat hogging and time spent searching for vacant seats on buses. This project aims to create a prototype system utilizing a pressure pad sensor and Arduino uno module for bus seats. The system will report real-time seat usage data to a web application via a localhost interface using NodeRed software. The pressure pad detects pressure on the seat when it is occupied. When a seat is unoccupied, no pressure is perceived. The pressure pad is created by pressing a sheet of aluminium foil between two pieces of cardboard. The pressure pad contains two wires as output, which are connected to the Arduino Uno. The Arduino Sketch includes code for sensing pressure from a pressure pad and the pressure sensed from the pressure pad is simply a measurement of resistance. The block diagram of the system using pressure pads as the pressure sensor is as shown in Figure 2.8 below [23].

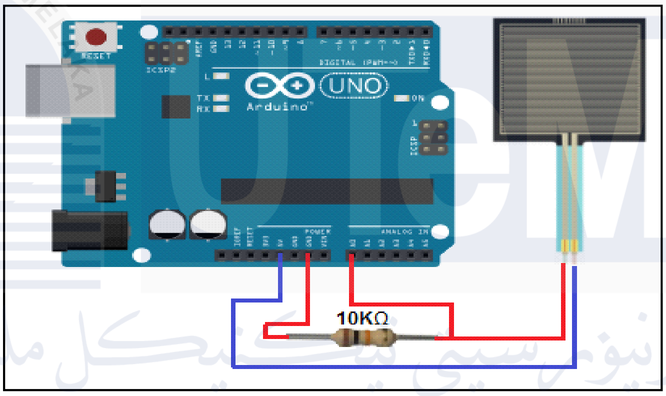

Figure 2.8: Block Diagram of Occupancy Detection System using Pressure pads Sensor [23]

Lastly, another case study developed a project titled 'Monitoring of Occupant States in Autonomous Vehicles using Capacitance-Sensing Imaging' where they used a capacitive-sensing mechanism as its pressure sensor. This paper proposes a new technology combines capacitive sensing and machine learning to identify and recognise occupants in real-time. The system's main component is a capacitance-sensing mat that covers both the seat base (SB) and backrest (BR) of a standard car seat. The mat's sensor circuitry continually analyses capacitance fluctuations and generates grayscale capacitance-sensing images (CSIs). CSIs show real-time fluctuations in capacitance induced by occupant location and posture on the mat. After validating the classifier's performance, the system is deployed to identify and recognize occupants in real time. The block diagram of the system using pressure pads as the pressure sensor is as shown in Figure 2.9 [24].

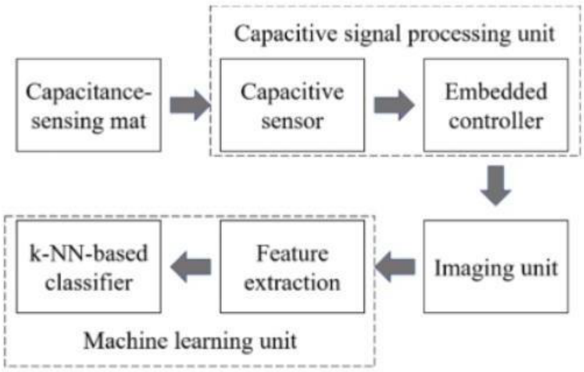

Figure 2.9: Block Diagram of Occupancy Detection System using Capacitive-Sensing Mat Sensor [24]

2.5.1 Summary of Seat Occupant Detection in Vehicles using Pressure Sensor

Table 2.3: Summary of Seat Occupant Detection in Vehicles using Pressure Sensor

Journal Author	Maepa and Moeti [22]	Sharma and Sawant [23]	Kumar et al. [24]
Pressure sensors used for seat occupant detection	Force Sensitive Resistor (FSR) sensor	Pressure pads sensor	Capacitive-sensing mat sensor
Advantages	• Accurate detection. • Real-time monitoring. • User-friendly. • Affordable.	• Cost-effective. • User-friendly. • Low power consumption. • Utilize various sensing technologies.	• Low power consumption. • Inexpensive. • East to integrate into existing vehicle seats. • Real-time monitoring available.
Disadvantages	• Limited sensing range. • Sensitive to temperature changes. • Complex calibration.	• Limited precision. • Limited sensing range. • Sensitive to environmental conditions.	• Sensitive to environmental factors. • Calibration challenges. • Limited pressure range.

Based on Table 2.3 where the previous case studies have been reviewed and summarized, there are a few criteria involving the sensors used for occupant detection system that are suitable to be used for this project for child detection system. From the research that has been done, journal [22] used FSR sensor, journal [23] used pressure pad sensor and journal [24] used capacitive-sensing mat sensor. All three journals are able to detect pressure successfully although each and every one of them have their own pros and cons. When comparing all three journals' sensors, the FSR sensor is chosen as it is the most appropriate one to be used although it has limited sensing range, but for seat occupant purposes to detect the absence of driver and presence of children, it is sufficient to be used which journal [22] utilizes the sensor in their project.

2.6 IoT Module Technologies

The internet of things, or IoT, is a network of interconnected devices that communicate and share data with one another and the cloud. IoT devices are often incorporated with technology such as sensors and software, and they can include mechanical and digital machinery as well as consumer products. The Internet of Things allows data to be sent over a network without the need for human-to-human or human-to-computer exchanges [25].

As for the previous case studies, there are several types of sensors that were used to develop the system. Firstly, one of the previous case studies was a development of a project titled 'Child Presence Detection and Alerting System in an Unmanned Car' where they used a Global System for Mobile Communications (GSM) module with Bluetooth. This project was proposed where it can even be developed to alert humans about their unsupervised pets in a car. The suggested idea aims to decrease tragic incidents from happening which can be reduced by utilizing Raspberry Pi 4 and a Bluetooth module to notify the owner if a child or pet is left alone. In the proposed work, they used speech detection to identify the child's scream using an SVM model, and the Bluetooth status of the parent's phone is checked to see if the parent is out of range. If Raspberry Pi detects that a parent has left their child in the car, an alarm message is delivered to the parent's phone via a Python application called Twilio. The block diagram of the system using GSM module with Bluetooth is as shown in Figure 2.10 below [26].

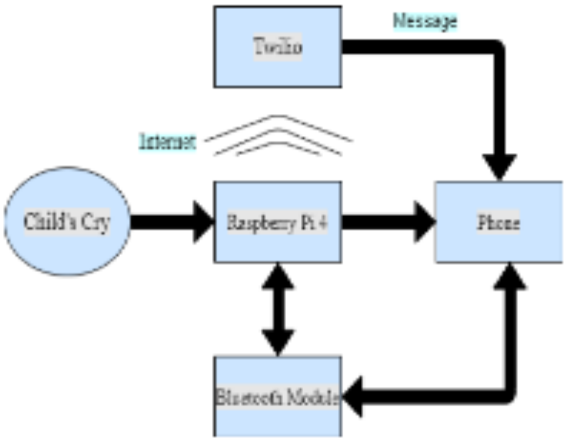

Figure 2.10: Block Diagram of IoT Module using GSM with Bluetooth Module [26]

Secondly, in another case study, they developed a project titled 'Design of Tracking System for Kids' where they used GPS system with Blynk application. Create a wearable gadget that can be utilized to monitor a lost child within a certain distance using the Internet of Things (IoT) to solve the problem. This technique might be extremely valuable in the world of security. The project focuses on creating a wearable GPS-based device and a Blynk app to track youngsters. This will make it easier for parents to find their children. In addition, the technology would allow users to follow children remotely via a mobile network. This will also provide the user with the latitude and longitude coordinates, as well as the child's current location. The system is made up of two primary parts: the NEO6M GPS receiver and the NodeMCU module. The latter includes a built-in GPS receiver that permits tracking the children. The block diagram of the system using GPS module with Blynk application is as shown in Figure 2.11 [27].

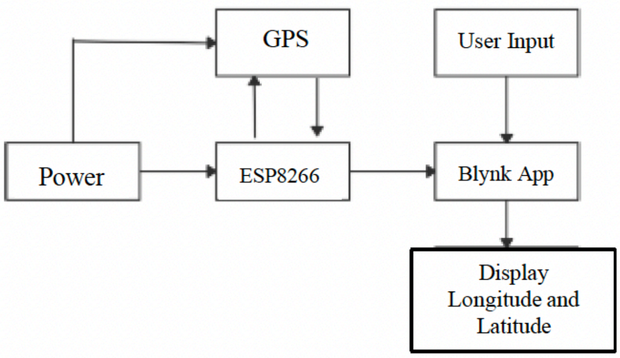

Figure 2.11: Block Diagram of IoT Module using GPS Module with Blynk Application [27]

Lastly, another case study developed a project titled ‘Intelligent Wi-Fi based Child Presence Detection System’ where they used Wi-Fi for IoT purposes This study discusses the first-of-its-kind intelligent CPD system based on commodity Wi-Fi. The proposed CPD system utilizes a statistical electromagnetic wave model to extract information from all multipath components. It includes a motion target detector for detecting awake/moving children, a stationary target detector for detecting sleeping children based on breathing rate, and a transition target detector based on a Naive Bayes Classifier with multipath profile features. The real-time testbed demonstrates that the proposed system achieves a detection rate of > 99.34% and a false alarm rate of < 4.38%, independent of the child's position or mobility state. The suggested system, built around 2.4/5GHz Wi-Fi, integrates seamlessly with existing in-car Wi-Fi systems and requires little CPU and memory resources, making it a viable option for CPD applications. The block diagram of the system using Wi-Fi as IoT module is as shown in Figure 2.12 below [28].

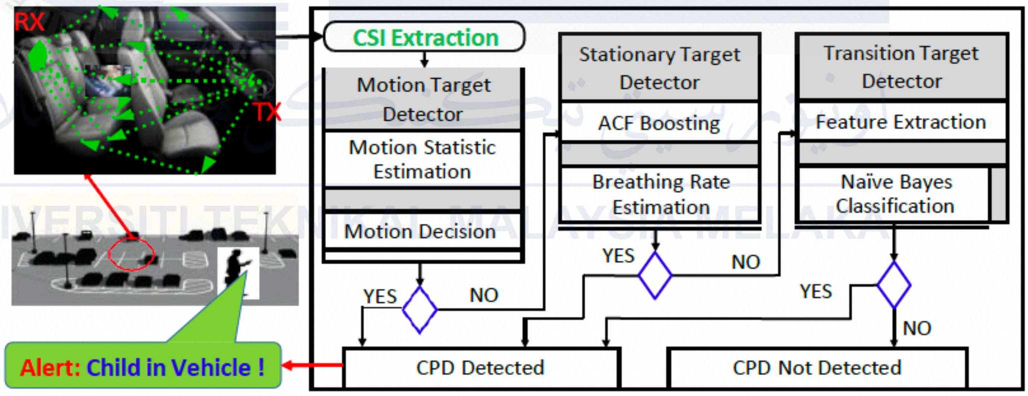

Figure 2.12: Block Diagram of IoT Module using Wi-Fi [28]

2.6.1 Summary of IoT Module Technologies

Table 2.4: Summary of IoT Module Technologies

Journal Author	**Bharadwaj K S et al. [26]**	**Shamuri and Alias [27]**	**Zeng et al. [28]**
IoT module technologies	Global System for Mobile Communications (GSM) module with Bluetooth	GPS with Blynk application	Wi-Fi
Advantages	• Widely-available and easily accessible. • Offers good quality voice call and less deterioration. • East to integrate with Arduino. • Has SMS and phone call feature. • Energy efficient.	• Provides real-time monitoring of sensor data. • Supports notification widgets to send alerts to phones. • Supports a wide range of hardware platforms including Arduino.	• Provides real-time monitoring of a child's presence. • Continuously track presence of child while Wi-Fi is within range of network. • Can send alerts and notifications. • Able to integrate with Arduino.
Disadvantages	• Vulnerable to hacking. • Interferes with certain electronics. • Slower data speeds compared to newer technologies.	• Relies on internet connectivity. • Dependent on cloud servers where the functionality of system could be impacted. • Limited free tier restrictions for widgets.	• Relies on a stable Wi-Fi connection. • Signals affected by interference or environmental conditions. • Higher overall cost. • More power consumption.

Based on Table 2.4 where the previous case studies have been reviewed and summarized, there are a few criteria involving the IoT module technologies used for occupant detection system that are suitable to be used for this project for child detection system. From the research that has been done, journal [26] used GSM module with Bluetooth, journal [27] used GPS with Blynk application, and journal [28] used Wi-Fi as its IoT module. All three journals are able to use the IoT module features successfully as intended although each and every one of them have their own pros and cons. When comparing all three journal's sensors, the GSM module is chosen as it is the most suitable one to be used as it does not rely on internet connectivity like the other journals. Although it has slower data speeds compared to newer technologies, it is sufficient to be used for the IoT purposes as it can send alerts by sending SMS or phone call to the parents, which journal [26] utilizes in their project.

2.7 Summary overall

To summarize the overall research projects that have been reviewed, there are several methods that can be implemented and enhanced which can be used for this project. The first criteria involve the most vital part of the project which is the microcontroller that is used to process data from the sensors. There are many types available in the market that is used for the same purpose as this project, however Arduino based microcontroller is the most common one as it is an open-source platform where the libraries are easily acquired. Moreover, the Arduino microcontroller can be easily integrated with GSM module for IoT purposes and internet connectivity through serial communication.

Furthermore, selecting suitable sensors is also one of the crucial parts of this project as it can help to achieve the project's objectives successfully. For sensing pressure and motion, there are various types of sensors that can operate as such in the market. From the research, the FSR sensor and PIR sensor are the most suitable one to be used for pressure sensing and motion sensing respectively. Both sensors are easy to use, readily available and is suitable for this project for detecting the absence of the driver and the presence of the child.

Lastly, this project also needs to have IoT module for it to be able to transfer data remotely to the user for real time monitoring and for notification purposes. The GSM module is chosen for this project as it can be used for those functions and does not need a stable network to be able to send and receive information. This system which combines all these components will be able to detect unattended children in vehicles in real-time and respond quickly to address the situation.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter summarizes and discusses the methods and techniques used to achieve the objectives for the development of smart detection system for unattended children in vehicles. It also covers the flow chart of the project to show the idea on how to implement the project by following a sequence of operations in order to achieve the objectives successfully.

To achieve the first objective which is to design a system that combines multiple sensors and data sources for reliable detection, the system was designed based on the selection of appropriate sensors and microcontrollers to obtain the desired output from the system. Then, the second objective is to develop an alert mechanism that informs guardians or emergency responders promptly, where the system must have GSM module so that it can connect with IoT. Lastly, to achieve the last objective which is to validate the system's accuracy and reliability in various conditions, the system was tested in various conditions and environments and data was collected in order to obtain high accuracy so that the project operates efficiently with low error.

3.1.1 Flow Chart of the Project

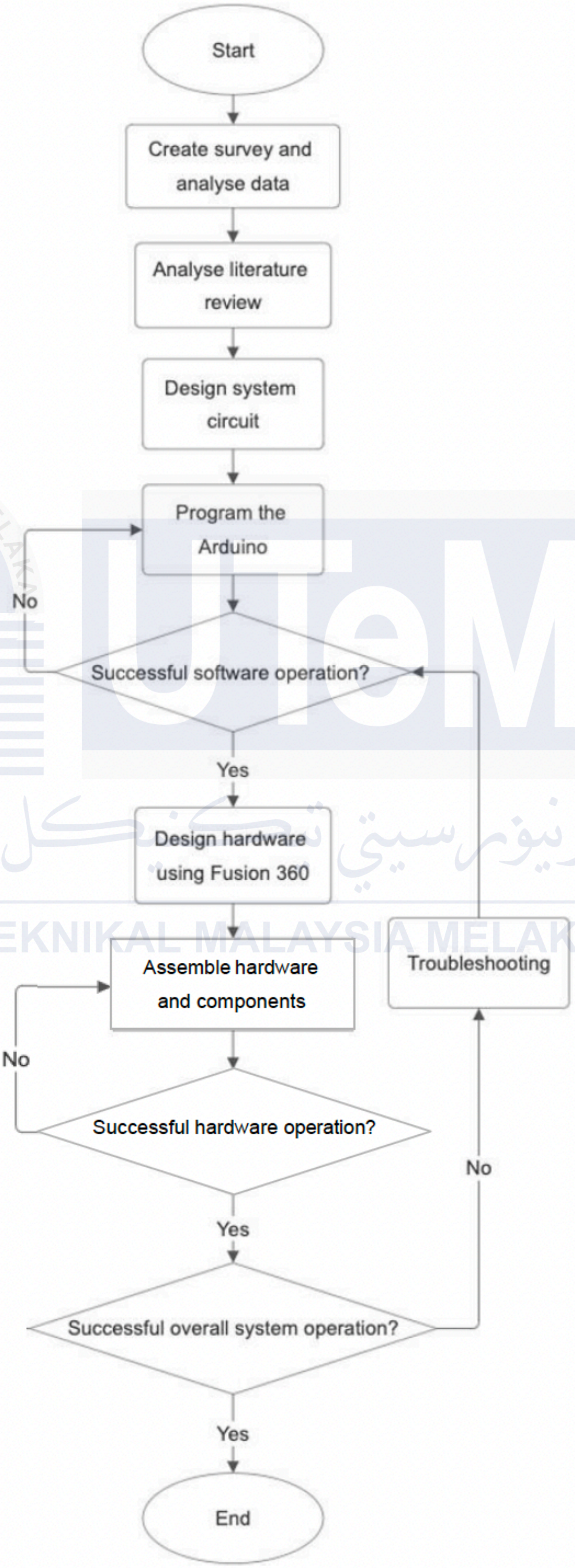

Figure 3.1: Flow Chart of the Project

3.2 Project Overview

Table 3.1: Project Overview

	Experiment	**Objective 1**	**Objective 2**	**Objective 3**
FYP 1	Design the child detection system. • Create survey to find out features needed to be included in the system.	√		
	Experiment 1: Test the FSR sensor.	√		
	Experiment 2: Test the PIR sensor.	√		
FYP 2	Experiment 3: Test the GSM module and buzzer. • Test the developed system with the alert mechanism that can inform guardians and nearby people.		√	
	Experiment 4: Test and validate the system's accuracy and reliability in various conditions.			√

3.2.1 Design the Child Detection System

A survey was created to find out features needed to be included in the system. Procedure for the experiment:

1. The survey for Smart Detection System for Unattended Children in Vehicles was created with the purpose of the survey stated clearly.
2. The questions were constructed clearly that aligns with the objectives of the project.
3. The survey was distributed to car users and the data were collected as shown in the results.
4. The system was designed while taking account the features that the respondents preferred.

5. The components for the system were chosen and the block diagram for the system was created as shown in Figure 3.2.

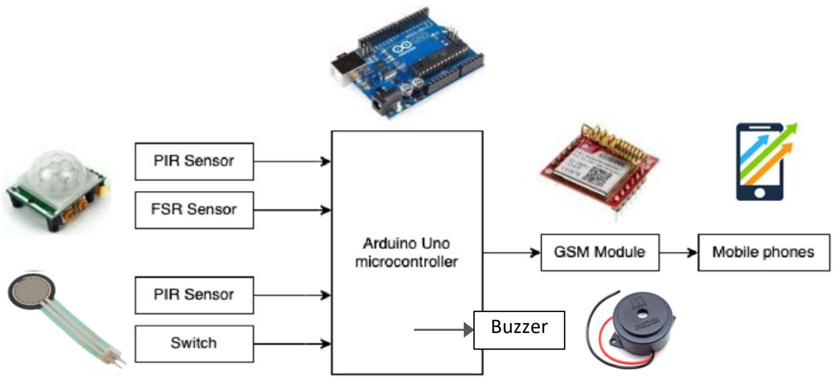

Figure 3.2: Block Diagram of the Child Detection System

3.2.1.1 Components used for the Child Detection System

After conducting research on previous studies, insights have been gained to choose the most suitable components for the child detection system, aligning with the project objectives. This involves designing the system and selecting appropriate components to fulfill the first objective of the project. The selection of components is crucial as it significantly impacts the system's functionality. This chapter will explore various components considered for their performance, specifications, and other relevant engineering factors during the experimental phase. Key components identified for the child detection system include the microcontroller, sensor types, alarm system, IoT system and other various components for the circuit. The components chosen for this project:

a) Arduino Uno as the Microcontroller

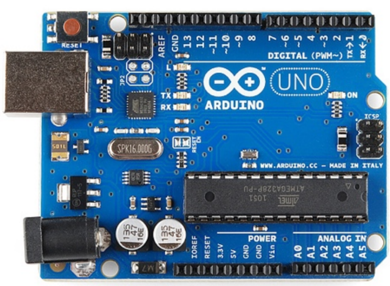

Figure 3.3: Arduino Uno Microcontroller [29]

The microcontroller is a critical component that manages the sequences and processes of the system. By integrating the appropriate programming software and libraries for each type of microcontroller, designers can develop and upload code to the microcontroller, enabling it to process data from the sensors. In this project, the Arduino Uno was chosen to oversee all operations of the child detection system. It is known for its versatility and ease of use, making it an ideal choice for this application. Its robust support community and extensive library of resources ensure reliable performance and ease of troubleshooting. With the Arduino Uno as the microcontroller, the child detection system can efficiently coordinate sensor inputs and trigger necessary actions to ensure safety.

As previously mentioned in the literature review, the Arduino Uno was selected for its compatibility with a wide range of sensors, its open-source nature, and its cost-effectiveness. Additionally, its compact size makes it suitable for simple projects, and most importantly, it excels in real-time data processing.

b) Sensors used for the Child Detection System

Choosing the right sensors is crucial for the system to function efficiently and to achieve the first objective of the project. For this project, various sensors were utilized to detect the presence and absence of both the driver and the child in the car. This allows the system to respond appropriately to different scenarios. The sensors were selected based on their specifications and performance characteristics, ensuring they are well-suited for this project. By carefully matching the sensor capabilities with the project's requirements, we can ensure accurate and reliable detection. Furthermore, the selected sensors enable smooth compatibility with the Arduino Uno, improving the overall integration of the system. Besides that, the system will utilize sensor fusion to accurately identify the presence of a child as it uses two types of sensors to efficiently detect the child, the driver seat only needs one type of sensor to detect their presence and absence.

As highlighted in the literature review, the decision to utilize the force-sensitive resistor sensor and passive infrared sensors was based on their capacity to discern human presence or absence. These sensors offer distinct advantages: the force-sensitive resistor sensor provides real-time feedback on physical pressure variations, while passive infrared sensors offer non-contact detection, minimizing physical interference and enhancing reliability. Additionally, both sensors are known for their cost-effectiveness and ease of integration into electronic systems.

- FSR 402 Force Sensing Resistor Sensor

Figure 3.4: FSR402 Force Sensing Resistor Sensor [30]

A Force Sensing Resistor, often referred to as a Force Sensor or FSR, is a low-cost, simple sensor used to measure weight, squeeze, and physical pressure. The most straightforward method of reading the FSR is to create a voltage divider by combining it with a static resistor. This produces a changing voltage that a microcontroller's analog-to-digital converter can read [30].

- HC-SR501 Passive Infrared Sensor

Figure 3.5: HC-SR501 Passive Infrared Sensor [31]

The passive infrared (PIR) sensor enables the detection of movement by individuals or animals within its detection range. All objects, including the human body, emit heat energy in the form of infrared radiation when their temperature exceeds absolute zero (-273.15 °C). The intensity of this radiation increases with the temperature of the object. As this radiation falls within the infrared spectrum, it remains invisible to the human eye. The PIR sensor is engineered to precisely detect these levels of infrared radiation, making it an efficient tool for motion detection applications.

c) **IoT Module for Child Detection System**

Figure 3.6: GSM SIM800L Module [32]

The IoT module serves as a key component within the child detection system, as it facilitates crucial communication between the system and the guardians. Acting as the system's notification system, the IoT module is responsible for transmitting and messages to the designated guardians, ensuring timely alerts and updates regarding the child's safety and well-being. Through its connectivity capabilities, the module enables seamless communication, allowing guardians to stay informed and responsive to any detected events or emergencies concerning the child. Its integration enhances the system's effectiveness by providing real-time notifications and enabling swift action when necessary, thus significantly encouraging the overall security and reliability of the child detection system.

As emphasized in the literature review, the selection of the GSM SIM800L module was due to many key factors. Its widespread availability and easy accessibility make it a practical choice for integration into various projects. Additionally, its compatibility with Arduino platforms simplifies the integration process, allowing for seamless communication with other system components. Moreover, the module has SMS and phone call features, along with its independence from internet signals, ensures reliable and versatile communication capabilities, making it well-suited for remote monitoring and alerting applications.

d) **Alarm System for the Child Detection System in Vehicles**

Figure 3.7: Piezo Buzzer

The alarm system integrated into the child detection system for vehicles serves a vital role in alerting bystanders when a child is left unattended. Since modifying the car's alarm system to integrate with the current circuit isn't feasible, a piezo buzzer is used as an indicator. It produces a loud sound which effectively draws attention to the situation without requiring extensive vehicle modifications. This proactive approach ensures that nearby individuals are promptly informed, enabling swift action to ensure the child's safety.

e) **LM2596 Voltage Regulator**

Figure 3.8: LM2596 Voltage Regulator

Using the SIM800L GSM Module with the Arduino's 5V output can potentially harm the module due to its operating voltage range of 3.4 V to 4.4 V. To ensure safe operation, it is advisable to use a 2A-rated DC-DC buck converter like the LM2596. Setting the output voltage to 4.0V which is the recommended ideal voltage, provides optimal performance. These converters offer greater efficiency compared to linear voltage regulator modules, making them a preferable choice. The SIM800L GSM module was connected to its respective battery or power source along with the Arduino board, and the module needs to also be connected to the voltage regulator to step down from the battery so that the output from the battery is a stable 4V which is the ideal voltage for the module to operate in. The circuit connection is as shown in Figure 3.9 below [32].

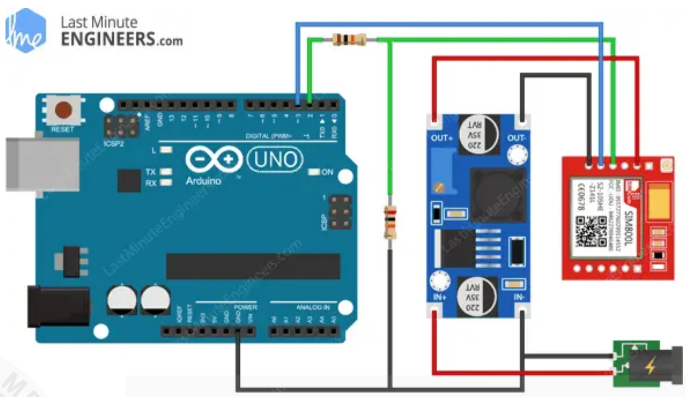

Figure 3.9: SIM800L GSM Module Connection to Arduino Uno and Battery

f) Batteries

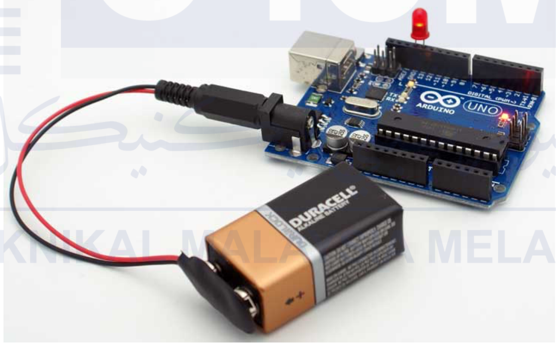

Figure 3.10: 9V Battery

The barrel jack, also known as a barrel connector, serves as the recommended method for powering the Arduino Uno in deployment. It accepts a standard 2.1mm center-positive plug from an adapter supplying 7 to 12 volts, ensuring the Uno receives a consistent 5V supply through its onboard voltage regulator. With a simple configuration where the center pin is positive and the outer sleeve is ground, the barrel connector minimizes the risk of accidental reverse polarity, safeguarding the board from potential damage. Whether for wall adapter or battery operation, the barrel jack provides both a secure mechanical connection and reliable electrical supply, reducing the likelihood of power disruption compared to other connection methods like jumper wires or clips, making it the preferred choice for most applications [33].

Figure 3.11: Lithium-ion 3.7V 2000mAh Rechargeable Battery

To power the GSM SIM800L module, two Lithium-ion batteries with a voltage of 3.7V and a capacity of 2000mAh each as shown in Figure 3.11 are connected in series, totaling 7.4V. However, since the module requires a 4V supply, the voltage needs to be regulated. This is achieved using an LM2596 voltage regulator, which stabilizes the voltage output to 4V. By stepping down the voltage from the combined batteries, the regulator ensures a consistent and reliable power source for the GSM module, optimizing its performance and longevity. This setup effectively addresses the voltage requirement of the module while utilizing the available battery capacity efficiently.

3.2.1.2 Design of the Child Detection System Case

After all the components had been selected, a case had to be designed, and the measurements of the case needed to be suitable for the components to fit inside. The device case has dimensions of 8 cm in height, 16 cm in length, and 12 cm in width, which can accommodate all the components. The 3D model of the casing, along with the components inside it, was created using Fusion 360 software as shown in Figure 3.12 and Figure 3.13 below. Additionally, the casing features a hole on the side to allow the sensors to extend out and be positioned in their respective places in the vehicle.

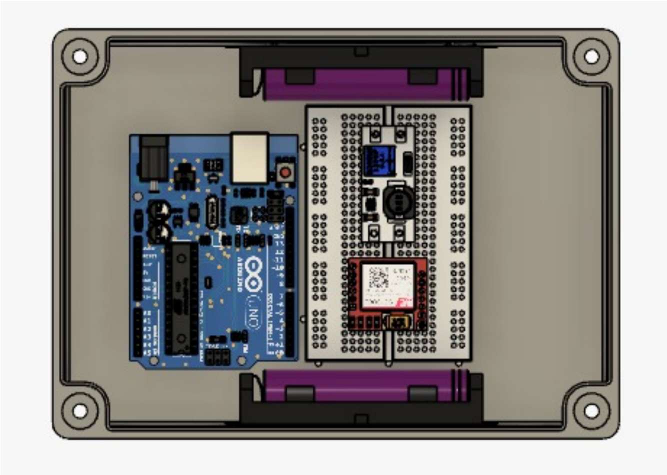

Figure 3.12: Top View of Device Casing with Components using Fusion 360 Software

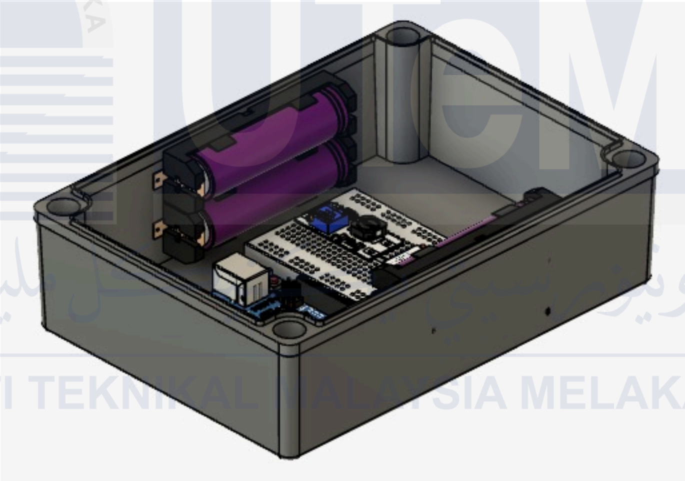

Figure 3.13: Side Angle View of Device Casing with Components using Fusion 360 Software

3.2.2 Experiment 1: Testing the Force Sensitive Resistor (FSR) Sensor

Procedure for testing the FSR sensor:

1. The components were connected as shown in Figure 3.14 below.
2. The programming code for a simple FSR sensor testing was uploaded into the Arduino Uno board using Arduino IDE software.
3. The serial plotter and serial monitor on the Arduino IDE software was opened and the output was observed.
4. The results were tabulated and analysed.

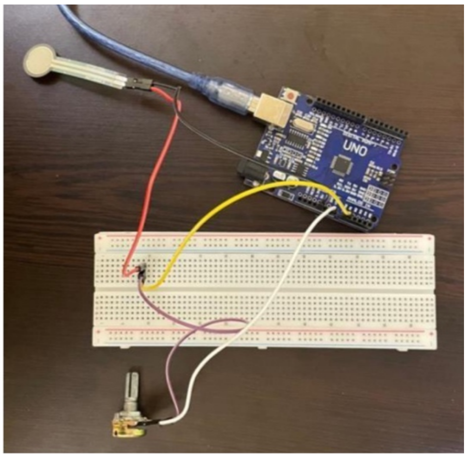

Figure 3.14: FSR Testing Circuit Connection [34]

During the testing of the FSR Baby and FSR Driver sensors, a 2.5kg weight dumbbell was placed on both sensors simultaneously to compare their readings. Despite the identical load, the sensors exhibited slight differences in their outputs. To calculate the error between the FSR Baby and FSR Driver readings, we first determined the average reading for each sensor. After we have calculated the average readings for each of the FSR sensors, the absolute difference was calculated. Then, we can finally obtain the percentage error of the sensors.

Furthermore, it was observed that when a dumbbell with a weight of 2.5kg occupied the seat and placed upon the FSR sensor, the sensor yielded a consistent reading of 313 and above. Since an average baby weight in Malaysia is 2.5kg – 3.5kg [35], then the dumbbell can be used as detection for the baby upon the FSR sensor.

Moreover, the empirical data obtained were served as the cornerstone for establishing the threshold value of 300. Any pressure reading surpassing this threshold indicates the presence of a human occupant, triggering subsequent actions or alerts, such as the activation of safety measures or notifications.

Conversely, pressure readings falling below the established threshold are deemed indicative of inanimate objects occupying the seat. This critical distinction ensures the system's reliability in accurately discerning between human occupants and objects. By employing such a calibrated approach, the system effectively differentiates between scenarios, enhancing its functionality and ensuring appropriate responses tailored to the specific presence detected. Hence, calibration is necessary to obtain the desired weight values since the sensor doesn't provide measurements in kilograms.

The calculation for the sensor errors are as shown below:

- **Average Reading for FSR Driver as shown in equation (3-1):**

$$= \frac{Sum\ of\ (Number\ x\ Weighting\ Factor)}{N} \qquad (3\text{-}1)$$

- **Average Reading for FSR Baby as shown in equation (3-2):**

$$= \frac{Sum\ of\ (Number\ x\ Weighting\ Factor)}{N} \qquad (3\text{-}2)$$

- **Absolute Difference as shown in equation (3-3):**

$$= \left|FSRAverage_{Driver} - FSRAverage_{Baby}\right| \qquad (3\text{-}3)$$

- **Percentage Error as shown in equation (3-4):**

$$= \left(\frac{Difference}{Average_{Reference}}\right) x\ 100\% \qquad (3\text{-}4)$$

An FSR sensor changes its resistance based on the force (or pressure) applied to its surface. The sensor's resistance decreases as the force increases. When used in a voltage divider circuit, the voltage output changes with the applied force, which can then be read by an analog pin on a microcontroller. The calculations are as shown below:

- **Analog Reading to Voltage Conversion:**

Use the formula voltage divider [34] to convert the analog reading to voltage as shown in equation (3-5) below:

$$V_{out} = V_{cc}\ x\ \frac{R_{fixed}}{R_{FSR} + R_{fixed}} \qquad (3\text{-}5)$$

- **The analogRead() function in Arduino converts the analog voltage (0 to 5V) into a value between 0 and 1023. The relationship is shown in equation (3-6) below:**

$$V_{out} = \frac{analogRead\ x\ V_{cc}}{1023} \qquad (3\text{-}6)$$

- **Calculate the FSR resistance as shown in equation (3-7) below:**

$$R_{FSR} = R_{fixed}\ x\ \left(\frac{V_{cc}}{V_{out}} - 1\right) \tag{3-7}$$

- **Since an empirical calibration approach is being used for the testing to make it easier, the calculation is shown in equation (3-8) below which can be simplified according to the calibration values obtained from the testing:**

$$F = W\ x\ g \tag{3-8}$$

$$g = 9.81m/s^2$$

- **Using Empirical Formula:**

  The proportional constant, k can be derived as shown in equation (3-9) below.

$$k = \frac{F}{analogRead} \tag{3-9}$$

- **Using this constant, the force for any analog reading can be calculated as shown in equation (3-10) below:**

$$F = k\ x\ analogRead \tag{3-10}$$

- **Convert Force to Weight in Kilograms as shown in equation (3-11) below:**

$$mass = \frac{F}{g} \tag{3-11}$$

The calculations were used to analyse the results obtained from the experiments and testing of the actual sensor. Since the average mean body weight of a person aged 18 and above is between 50kg to 70kg [36], the sensor was tested on a chair with a person weighing 50kg sitting upon it and the results were observed on the serial monitor of the Arduino IDE software.

The error between the calculated values and the measured values on the serial monitor were calculated by using the following formula as shown in equation (3-12) for error calculation:

$$Error(\%) = \left|\frac{Measured\ Value - Calculated\ Value}{Calculated\ Value}\right| x\ 100\% \quad (3\text{-}12)$$

3.2.3 Experiment 2: Testing the Passive Infrared (PIR) Sensor

Procedure for testing the PIR sensor:

1. The components were connected as shown in Figure 3.15.
2. The programming code for a simple PIR sensor testing was uploaded into the Arduino Uno board using Arduino IDE software.
3. The serial plotter and serial monitor on the Arduino IDE software was opened and the output was observed.
4. The results were tabulated and analysed.

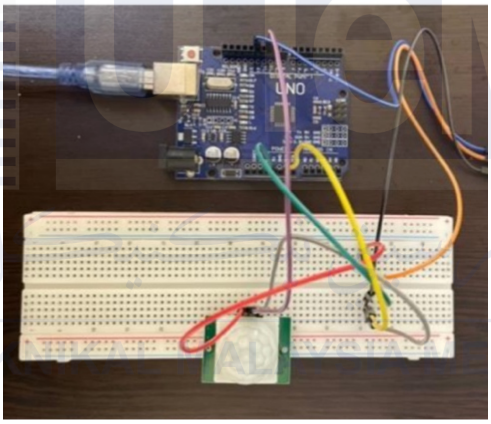

Figure 3.15: PIR Testing Circuit Connection [37]

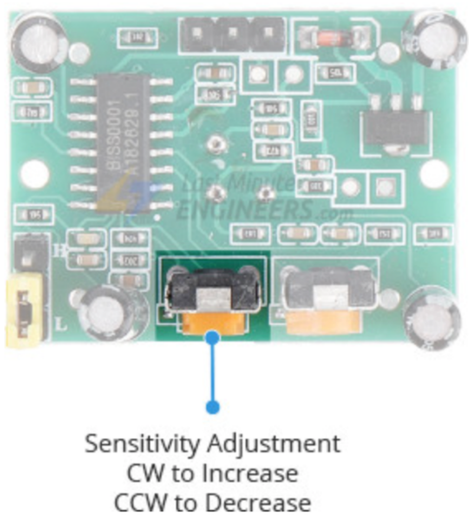

Figure 3.16: Sensitivity Adjustment of PIR sensor [31]

This potentiometer sets the maximum detection range, which can be adjusted from approximately 3 meters to 7 meters (9 to 21 feet). However, the topology of the room can affect the actual range. Rotating the pot clockwise increases the sensitivity and thus the range, and vice versa [31]. For placement in a conventional vehicle to detect a baby in a car seat, a high range is unnecessary. Therefore, the sensor was set to its lowest setting, which is 3 meters, as this is sufficient for use within the vehicle, as shown in Figure 3.16 above.

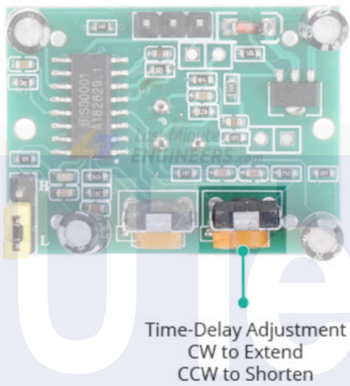

Figure 3.17: Time-Delay Adjustment of PIR sensor [31]

This potentiometer sets how long the output will remain HIGH after motion is detected. It can be adjusted from 1 second to about 3 minutes. Turning the potentiometer clockwise increases the delay, while turning the potentiometer counter-clockwise decreases the delay. Since the movement of a baby is crucial, even slight movements need to be detected to ensure the sensor accurately identifies the presence of a baby in the car [31]. Therefore, the potentiometer is set to the lowest delay setting, the adjustment of the sensor is as shown in Figure 3.17 above.

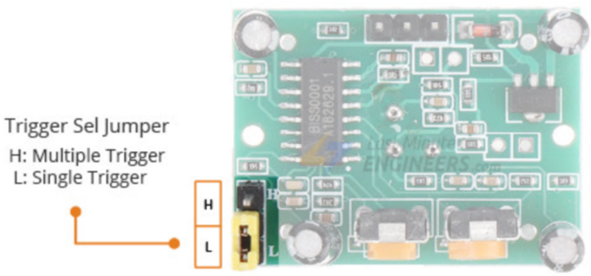

Figure 3.18: Trigger Selection Jumper of PIR Sensor [31]

There are two trigger modes as shown in Figure 3.18 above which is to determine how the sensor reacts to motion detection. In Single Trigger Mode, the system activates a single trigger upon detecting motion, remaining active for a period set by the Time-Delay potentiometer. Subsequent detections are ignored until the output returns to LOW at the end of the delay. In this mode, it's crucial to detect even slight movements, ensuring the system responds promptly to any indication of the baby's activity.

On the other hand, Multiple Trigger Mode activates the output upon detecting motion, staying active for a duration determined by the Time-Delay potentiometer. Unlike the single trigger mode, further detections are not ignored but instead reset the time delay countdown. The system only returns to LOW after a period of no motion. This mode ensures continuous monitoring of the baby's movements without interruptions, providing a vigilant response to any ongoing activity [31].

The Single Trigger Mode operates a system that activates a single trigger upon detecting initial motion, sustaining this alert state for a predefined period determined by the Time-Delay potentiometer. This mode guarantees an immediate response to any movement suggesting the presence of a baby, effectively detecting even subtle movements. It was selected as the optimal choice to ensure the system remains highly responsive to any activity inside the car, crucially avoiding missed triggers and maintaining vigilance.

3.2.4 Experiment 3: Testing the GSM Module and Buzzer

Test the developed system with the alert mechanism that can inform guardians and people nearby. Procedure for testing the GSM module and buzzer:

1. The components were connected as shown in Figure 3.19.
2. The programming code for a simple GSM module and buzzer testing was uploaded into the Arduino Uno board using Arduino IDE software.
3. The serial plotter and serial monitor on the Arduino IDE software was opened and the output was observed.
4. The results were tabulated and analysed.

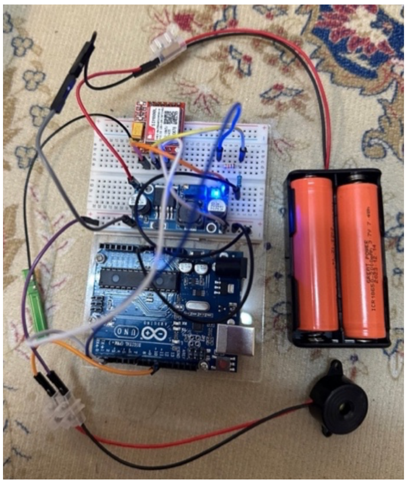

Figure 3.19: GSM Module and Buzzer Testing Circuit Connection [38]

3.2.5 Experiment 4: Test and Validate the System's Accuracy and Reliability in Various Conditions

Procedure for proposed testing the system in various conditions.

1. The components were connected as shown in Figure 3.20.
2. The programming code for the system was uploaded into the Arduino Uno board using Arduino IDE software.
3. The device and the sensors were placed in the car as shown in Figure 3.21.
4. The testing was repeated a few times in various conditions.
5. The results for each condition were tabulated and analysed.

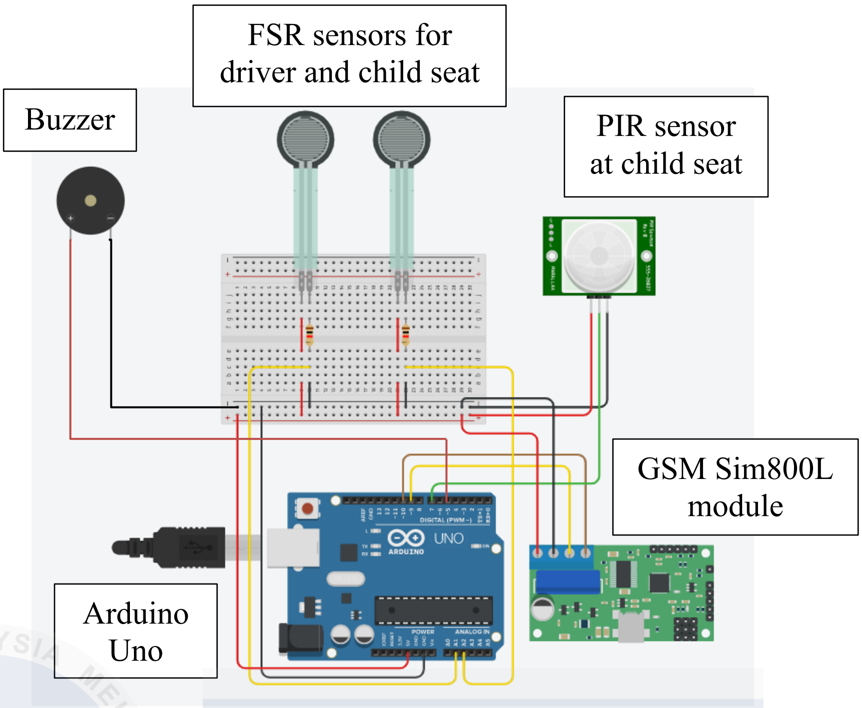

Figure 3.20: System Connection Test

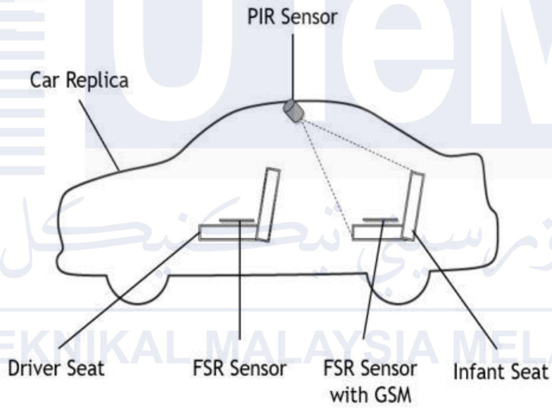

Figure 3.21: The Placement of the Device and Sensors in the Car [39]

The device underwent multiple tests to assess its efficiency and identify recurring errors encountered during operation. Each component's functionality was carefully observed to ensure optimal performance and reliability. These tests were crucial in pinpointing any issues that arose during use, allowing for necessary adjustments and improvements to enhance overall functionality and user experience.

3.3 Summary

To summarize, this chapter presents the methodology in order to develop a new, effective project. The methodology outlined in this project ensures a systematic and comprehensive approach to developing a child detection system. By integrating a cutting-edge sensor technologies and user-friendly interfaces, the project aims to contribute significantly to the prevention of child-related incidents. This methodology serves as a guiding framework, embodying the project's commitment to efficiency, safety, and ethical standards in technological innovation. To ensure the reliability of the child detection system, it is crucial to calculate and minimize errors, specifically false positives and false negatives. For example, if the system frequently mistakes shadows for children (false positives) or fails to detect children in certain positions (false negatives), the algorithms are refined to improve accuracy. Other than that, the sensors errors can also be calculated by taking an average reading for any particular sensor. Then, the absolute difference between these averages were determined. Using the average reading of the sensor as the reference, the percentage error was calculated. This method highlights the inherent variation between the sensors, reflecting typical discrepancies due to manufacturing tolerances and sensor calibration. Regular monitoring and updates ensure the system adapts to various conditions, thereby maintaining high standards of safety and effectiveness.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter contains the results and discussion of experiment 1, experiment 2 and experiment 3 that have been carried out for this semester. The results were obtained and recorded in graph form. Besides that, this chapter also discusses the result for each experiments done to achieve the objectives. The first objective was achieved by designing the child detection system according to the respondents' preferences, as shown in the results. After that, experiment 1 and experiment 2 were designed to meet the first objective which is to design a system that combines multiple sensors and data sources for reliable child detection. As for experiment 3, it is implemented to achieve objective 2 that is to test the developed system with alert mechanism, and experiment 4 was done to achieve objective 3 which is to test and validate the system's accuracy and reliability in various conditions.

4.2 Design and Test the Child Detection System

In this experiment, the child detection system was designed by first creating a survey regarding the features of the system to find out what the car users with children prefer to have in order to reduce the tragic incidents from occurring. The survey was created based on the system's objectives and was relayed to the public where the results were obtained as shown in Figure 4.1 and Figure 4.2.

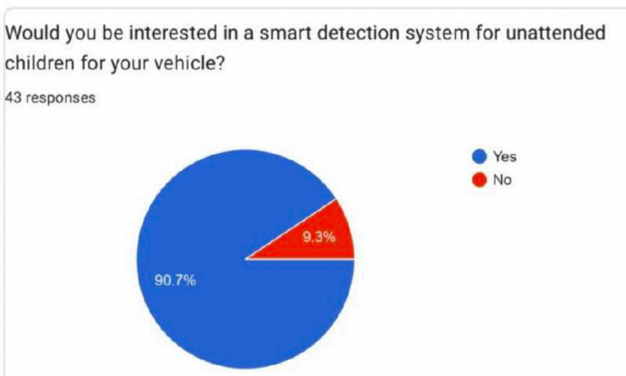

Figure 4.1: Survey Respondents Interest on the System

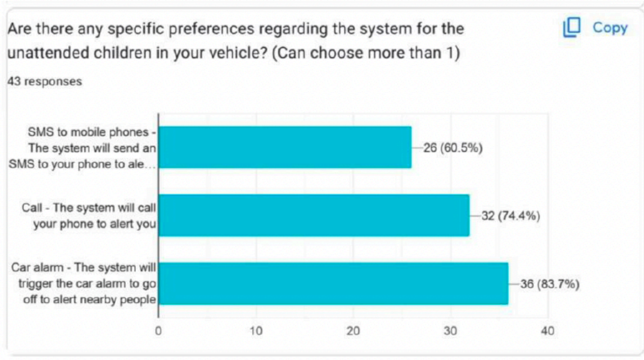

Figure 4.2: Features Preferences of Respondents

In this experiment, the child detection system was designed by first creating a survey regarding the features of the system to find out what the car users with children prefer to have in order to reduce the tragic incidents from occurring. The survey was created based on the system's objectives and was relayed to the public where the results were obtained as shown. From Figure 4.1, it is clear that most of the respondents are interested in the implementation of this system where 90.7% of them are interested, while only 9.3% of them are not. Other than that, from Figure 4.2, most of the respondents preferred the system to have a car alarm feature, however it could not be implemented in this project due to the cost constraints, so a buzzer was used instead as the car alarm indicator. Even so, the other two features that are equally as important are fortunately able to be implemented which satisfies the primary objectives of this project. Figure 4.3 below shows the complete child detection system device, including the case that houses all the components.

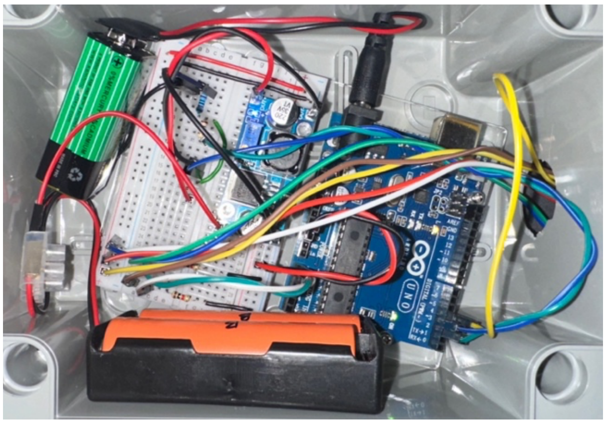

Figure 4.3: Child Detection System Device

4.3 Experiment 1: Testing the Force Sensitive Resistor (FSR) Sensor

In this experiment, the FSR sensors were tested in a controlled environment to monitor the sensitivity and efficiency for it to sense pressure or weight of the driver and the child. This sensor was tested, and the experiment was repeated several times to confirm the results. The reading was observed in the Arduino IDE software and the results that were obtained shows that the sensor is working and can be used for the system. The results were also extracted and tabulated using Microsoft Excel software. The sensor was able to sense when a pressure was placed upon it as shown in Table 4.1 below and also shown in Figure 4.4.

Table 4.1: FSR Sensor Pressure Testing

Sensor Reading	Status	Light Pressure	Medium Pressure
469	- Medium Pressure	200	500
622	- High Pressure	200	500
382	- Medium Pressure	200	500
338	- Medium Pressure	200	500
389	- Medium Pressure	200	500
0	- No pressure	200	500
0	- No pressure	200	500
778	- High Pressure	200	500
540	- High Pressure	200	500
506	- High Pressure	200	500
0	- No pressure	200	500
162	- Light Pressure	200	500
107	- Light Pressure	200	500
365	- Medium Pressure	200	500
240	- Medium Pressure	200	500
659	- High Pressure	200	500
272	- Medium Pressure	200	500
160	- Light Pressure	200	500
19	- No pressure	200	500

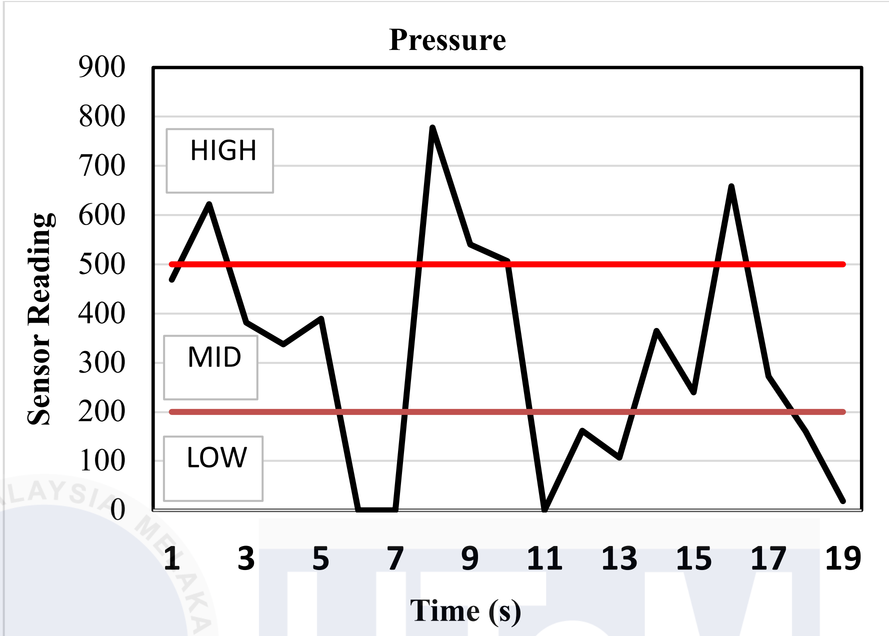

Figure 4.4: Graph of FSR Pressure Testing

The experiment conducted to evaluate the FSR sensor's sensitivity and efficacy in detecting pressure or weight yielded promising results. Through multiple iterations and meticulous testing in a controlled environment, the sensor consistently demonstrated its capability to discern varying pressure levels. Table 4.1 provides a comprehensive overview of the pressure readings obtained during the experiment, showcasing the sensor's ability to differentiate between no pressure, light pressure, medium pressure, and high pressure scenarios.

Furthermore, the extracted data was graphically represented in Figure 4.4, illustrating the relationship between the pressure readings and their corresponding statuses. The graph visually elucidates the sensor's responsiveness to different pressure levels, with distinct clusters representing no pressure, light pressure, medium pressure, and high pressure instances. Such graphical representation enhances the comprehension of the sensor's performance characteristics and facilitates the identification of patterns or trends within the data.

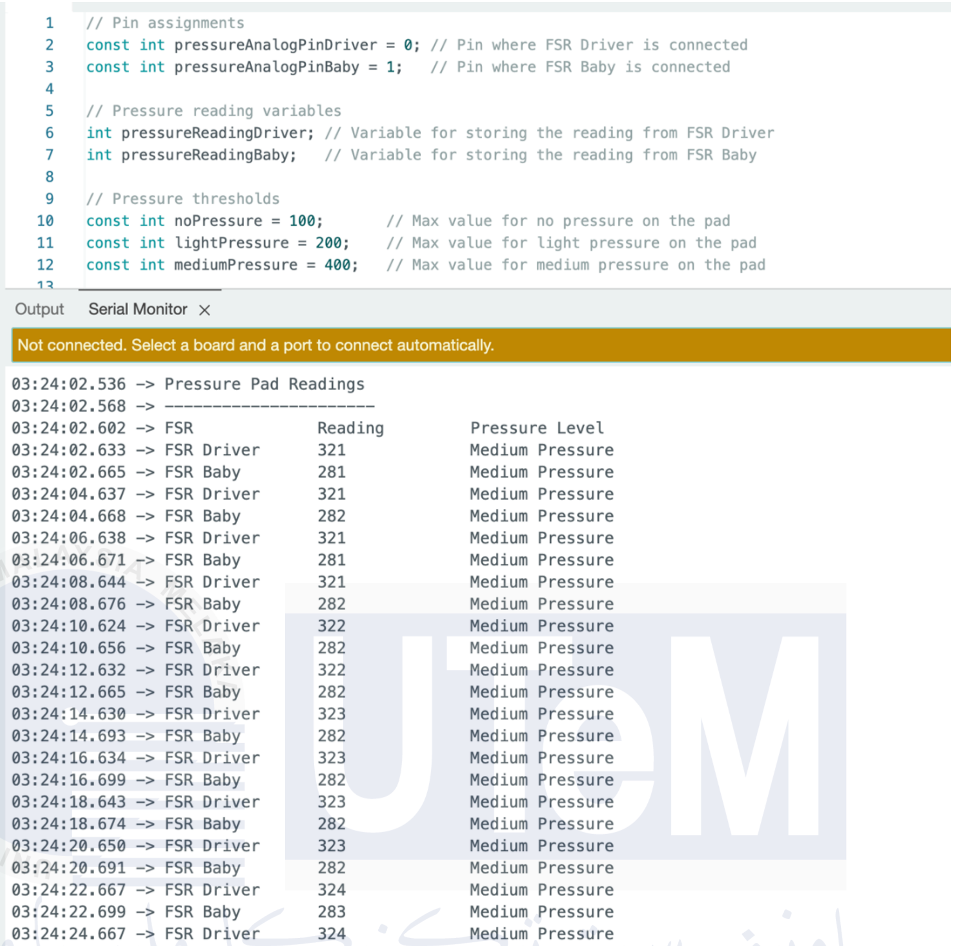

Figure 4.5: FSR Sensor Testing on Arduino Serial Monitor

The FSR sensor for the baby and for the driver was tested and the results were obtained as shown in Figure 4.5 above. It can be seen that there seems to be a slight difference in sensor readings between the FSR Baby and the FSR Driver, as shown in the serial monitor of the Arduino IDE software, is likely due to inherent variations in sensor calibration, manufacturing tolerances, and the distribution of the applied weight. Although both sensors are subjected to the same 2.5kg weight, minor differences in their individual sensitivity and contact area can result in the observed differences in readings.

- **Average Reading for FSR Driver as shown in equation (3-1):**

$$= \frac{(321\ x\ 8) + (322\ x\ 4) + (323\ x\ 4)}{16} = 321.25$$

- **Average Reading for FSR Baby as shown in equation (3-2):**

$$= \frac{(281\ x\ 2) + (282\ x\ 10) + (283\ x\ 4)}{16} = 282.25$$

- **Absolute Difference as shown in equation (3-3):**

$$= |321.25 - 282.25|$$
$$= 39$$

- **Percentage Error as shown in equation (3-4):**

$$= \left(\frac{39}{321.25}\right) x\ 100\%$$
$$= 12.14\%$$

The error between the FSR Baby and the FSR Driver readings, calculated to be 39 units with a percentage error of approximately 12.14%, is relatively small. This indicates that both sensors are still functioning as desired and providing consistent readings under the same load conditions. Such minor discrepancies are expected due to manufacturing tolerances and calibration differences, but they do not significantly impact the overall performance of the sensors. Additionally, the lowest sensor reading for the FSR Baby is 281, suggesting that a threshold of 280 can be used to reliably detect the presence of a baby. Similarly, for the FSR Driver, with the lowest reading at 321, a threshold of 320 can be used to detect the driver. This ensures accurate detection and differentiation between the two sensors' applications.

```
19:27:19.668 -> Pressure Pad Readings
19:27:19.699 -> ----------------------
19:27:19.699 -> Reading          Newton          Kg
19:27:19.732 -> 32               2.52            0.2570
19:27:21.674 -> 64               5.04            0.5139
19:27:23.684 -> 96               7.56            0.7709
19:27:25.699 -> 129              10.16           1.0359
19:27:27.708 -> 161              12.68           1.2929
19:27:29.723 -> 193              15.20           1.5498
19:27:31.704 -> 226              17.80           1.8148
19:27:33.720 -> 258              20.32           2.0718
19:27:35.730 -> 290              22.84           2.3288
19:27:37.710 -> 323              25.44           2.5938
19:27:39.718 -> 323              25.44           2.5938
19:27:41.729 -> 323              25.44           2.5938
19:27:43.741 -> 323              25.44           2.5938
19:27:45.753 -> 323              25.44           2.5938
19:27:47.731 -> 323              25.44           2.5938
19:27:49.747 -> 323              25.44           2.5938
19:27:51.752 -> 322              25.36           2.5857
19:27:53.747 -> 321              25.28           2.5777
19:27:55.758 -> 321              25.28           2.5777
19:27:57.763 -> 320              25.20           2.5697
19:27:59.785 -> 320              25.20           2.5697
19:28:01.787 -> 319              25.12           2.5617
19:28:03.779 -> 318              25.04           2.5536
19:28:05.782 -> 318              25.04           2.5536
19:28:07.792 -> 317              24.96           2.5456
19:28:09.797 -> 316              24.88           2.5376
19:28:11.812 -> 316              24.88           2.5376
19:28:13.793 -> 315              24.81           2.5295
19:28:15.795 -> 315              24.81           2.5295
19:28:17.823 -> 315              24.81           2.5295
19:28:19.820 -> 314              24.73           2.5215
19:28:21.815 -> 314              24.73           2.5215
19:28:23.816 -> 313              24.65           2.5135
19:28:25.815 -> 313              24.65           2.5135
19:28:27.847 -> 313              24.65           2.5135
19:28:29.821 -> 313              24.65           2.5135
```


Figure 4.6: FSR Sensor Readings Conversion

The weight of the dumbbell is 2.5 kg, and the sensor reading is 313, but the weight measured on the serial monitor is 2.5135kg, so the force calculated as shown in equation (3-8) below:

$$F = W \ x \ g$$

$$g = 9.81 m/s^2$$

$$Force \ (N) = 2.5135 kg \ x \ 9.81 m/s^2$$

$$Force \ (N) = 24.66 N$$

- **Using Empirical Formula:**

  Since the sensor reading of 313 and the Force measured on the serial monitor is 24.66N, the proportional constant, k can be derived as shown in equation (3-9) below.

$$k = \frac{F}{analogRead}$$

$$k = \frac{24.66}{313}$$

$$k = 0.0788$$

- **Using this constant, the force for any analog reading can be calculated as shown in equation (3-10) below:**

$$F = k \ x \ analogRead$$

$$F = 0.0788 \ x \ 313$$

$$F = 24.66N$$

- **Convert Force to Weight in Kilograms as shown in equation (3-11) below:**

$$mass = \frac{F}{g}$$

$$mass = \frac{24.66}{9.81}$$

$$mass = 2.51kg$$

- **Calculate error between measured values and calculated values as shown in equation (3-12):**

$$Error(\%) = \left|\frac{Measured\ Value - Calculated\ Value}{Calculated\ Value}\right| x\ 100\%$$

$$= \left|\frac{2.5135kg - 2.5kg}{2.5kg}\right| x\ 100\%$$

$$= 0.54\%$$

The calculation process shows a precise and reliable relationship between the sensor reading, force, and weight. With a dumbbell weight of 2.5 kg and a corresponding sensor reading of 313, the measured weight on the serial monitor is 2.5135 kg. This translates to a calculated force of 24.66 N. Using the sensor reading and the measured force, a proportional constant of 0.0788 is derived. This constant allows for accurate calculation of the force from any sensor reading. Applying it back to the initial reading of 313, the force remains 24.66 N, confirming the formula's accuracy. The consistent conversion of force to weight further validates the reliability of the empirical method used, aligning closely with the expected values.

The pressure pad readings shown in Figure 4.6 indicate a consistent and reliable relationship between sensor readings, force in Newtons, and weight in kilograms. The data demonstrates how sensor readings, recorded at intervals, translate into forces and masses. For example, an initial reading of 32 corresponds to a force of 2.52 N and a weight of 0.2570 kg. This linear relationship is evident throughout the data, with increasing sensor readings producing proportionally higher forces and weights.

Notably, when the sensor reading stabilizes around 323, the force measures approximately 25.44 N and the weight about 2.5938 kg. The readings remain consistent over multiple measurements, highlighting the sensor's stability and the accuracy of the empirical formula used for calculations. The slight variations in readings towards the end of the dataset show a minimal fluctuation, confirming the precision of the measurements. This consistent pattern underscores the effectiveness of the sensor in accurately translating readings into physical forces and weights.

The threshold value of 310 can be used as a reference point for detecting the presence of an average-weight baby in Malaysia, which is approximately 2.5 kg. Since a sensor reading of 313 corresponds to this weight, setting a threshold slightly below this value (at 310) ensures that any reading above 310 will reliably indicate the presence of a baby. This provides a simple and effective way to detect if a baby is on the pressure pad, ensuring accurate monitoring based on the established relationship between sensor readings and weight.

The error between the measured weight of 2.5135 kg and the calculated weight of 2.5 kg is 0.54%, indicating a very high level of accuracy. This small discrepancy confirms the reliability of both the sensor readings and the empirical method used for the calculations, ensuring precise and consistent measurement results. Overall, the results obtained from both the tabulated data and the graphical representation affirm the functionality and reliability of the FSR sensor for integration into the system. The sensor's consistent and discerning response to varying pressure levels underscores its suitability for accurately detecting the presence of occupants, such as the driver and the child, within the vehicle environment, thereby contributing to enhanced safety and monitoring capabilities.

4.4 Experiment 2: Testing the Passive Infrared (PIR) Sensor

In this experiment, the PIR sensor was tested in a controlled environment to monitor the sensitivity and efficiency for it to sense motion or thermal heat of the child. This sensor was tested, and the experiment was repeated several times to confirm the results. The reading was observed in the Arduino IDE software and the results that were obtained shows that the sensor is working and can be used for the system. The results were also extracted and tabulated using Microsoft Excel software. The sensor was able to sense when a human moved in front of it as shown in Table 4.2 and Figure 4.7.

Table 4.2: PIR Sensor Motion Testing

TIME (S)	DATA	STATUS
1	0	Motion stopped
2	1	Motion detected
3	0	Motion stopped
4	1	Motion detected
5	0	Motion stopped
6	0	Motion stopped
7	0	Motion stopped
8	1	Motion detected
9	1	Motion detected
10	1	Motion detected
11	0	Motion stopped
12	1	Motion detected
13	1	Motion detected
14	0	Motion stopped

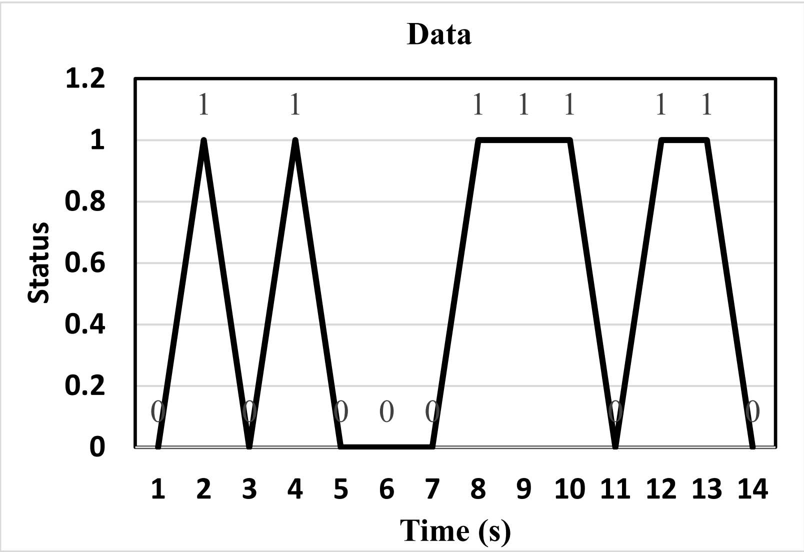

Figure 4.7: PIR Sensor Motion Testing

The experiment conducted to evaluate the PIR sensor's sensitivity and efficiency in detecting motion or thermal heat of the child yielded insightful results. Through systematic testing in a controlled environment and repeated iterations to ensure reliability, the sensor showcased its ability to accurately discern motion presence. Table 4.2 provides a detailed record of the sensor's response over time, indicating instances of motion detection and cessation.

Furthermore, the extracted data was graphically depicted in Figure 4.7, offering a visual representation of the sensor's performance in detecting motion events. The graph illustrates the fluctuation of motion detection status over time, with distinct spikes representing instances of motion detected and plateaus indicating periods of no motion. Such graphical representation facilitates the interpretation of the sensor's responsiveness and aids in identifying patterns or trends in motion detection.

Overall, the results obtained from both the tabulated data and the graphical representation affirm the functionality and reliability of the PIR sensor for integration into the system. The sensor's consistent and accurate detection of motion presence underscores its suitability for monitoring the child's movements within the vehicle environment, thereby contributing to enhanced safety and surveillance capabilities.

4.5 Experiment 3: Testing the GSM Module and Buzzer

In this experiment, the GSM module and buzzer was tested in a controlled environment to monitor the connectivity and efficiency in sending and receiving alerts to mobile phones. The module and buzzer were tested, and the experiment was repeated several times to confirm the results. The readings were observed, and the results obtained showed that the components worked as intended and from there it was decided that they can be used for the system.

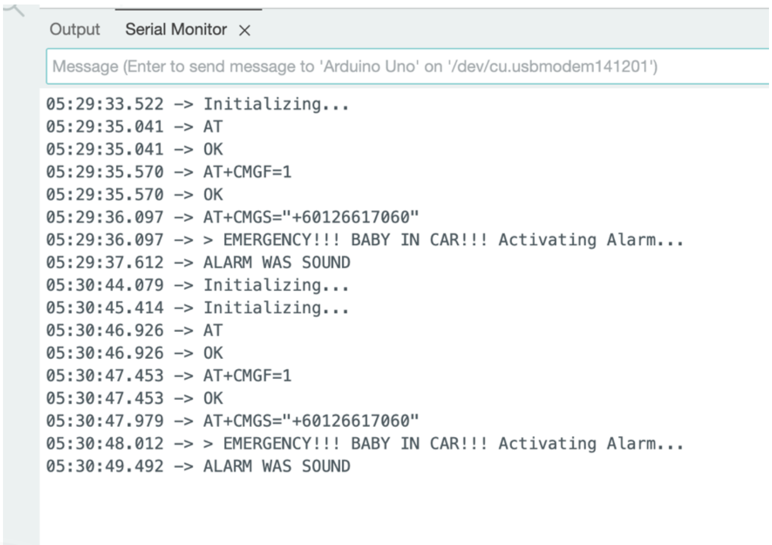

Figure 4.8: Serial Monitor Readings on Arduino IDE

In this experiment, the SIM800L GSM module and a buzzer were tested to evaluate their effectiveness in sending and receiving emergency alerts to mobile phones. The tests were conducted multiple times to ensure accuracy. During each trial, the system successfully initialized, sent an emergency alert message, and activated an alarm as designed.

The Serial Monitor output as shown in Figure 4.8 shows a series of steps where the GSM module initializes, sets the SMS format, and sends an emergency message. Each time the message "EMERGENCY!!! BABY IN CAR!!! Activating Alarm..." was sent, the alarm was triggered immediately after. This consistent pattern of initialization, message sending, and alarm activation confirms that both the communication and alert mechanisms are working correctly.

The results of these tests indicate that the GSM module and buzzer are reliable components for an emergency alert system. Their consistent performance in controlled conditions suggests they can be effectively used in real-world scenarios, such as detecting when a baby is left in a car and promptly notifying the necessary contacts.

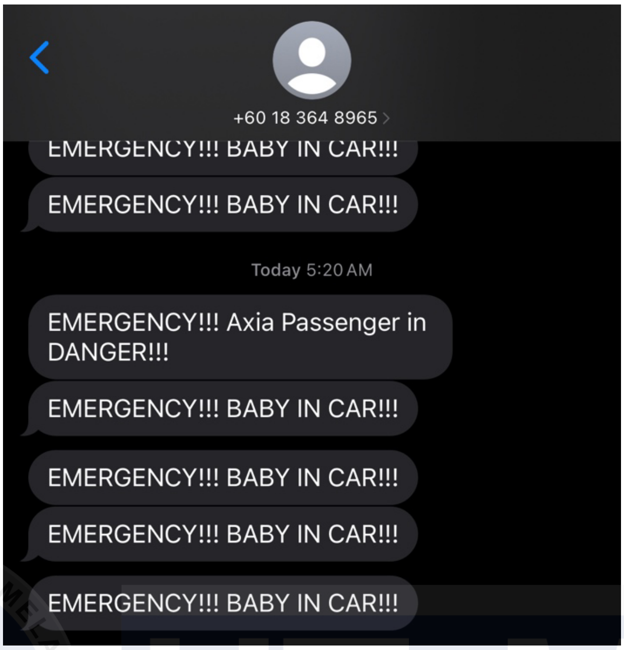

Figure 4.9: SMS Notification sent by GSM SIM800L module

The series of text messages displayed in Figure 4.9 demonstrates the effectiveness of the SIM800L GSM module in sending emergency alerts. The consistent and repeated delivery of the "EMERGENCY!!! BABY IN CAR!!!" messages indicates that the SIM800L module is reliably performing its intended function. This reliability is crucial for emergency situations, confirming that the GSM SIM800L module is a dependable component for real-time alert systems, ensuring immediate notification to the necessary contacts.

4.6 Experiment 4: Test and Validate the System's Accuracy and Reliability in Various Conditions

In this experiment, the system was tested in various conditions to monitor the connectivity and efficiency as a child detection system. This system was tested and the experiment was repeated several times to confirm the accuracy of the system from the obtained results. The readings were observed, and the results obtained showed that the detection system worked as intended which fulfills the objectives of this project.

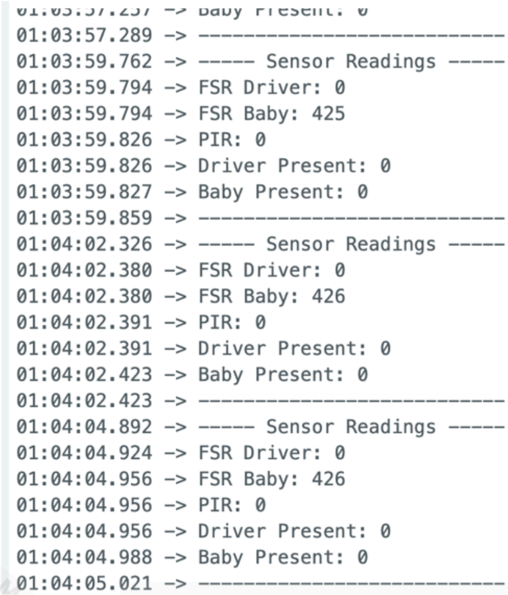

Figure 4.10: Serial Monitor Readings of Child Detection System

The displayed Serial Monitor output shown in Figure 4.10 shows sensor readings from an experiment conducted to test the child detection system. The sensors involved include an FSR (Force Sensitive Resistor) for both the driver and baby, and a PIR (Passive Infrared) sensor to detect movement of the baby.

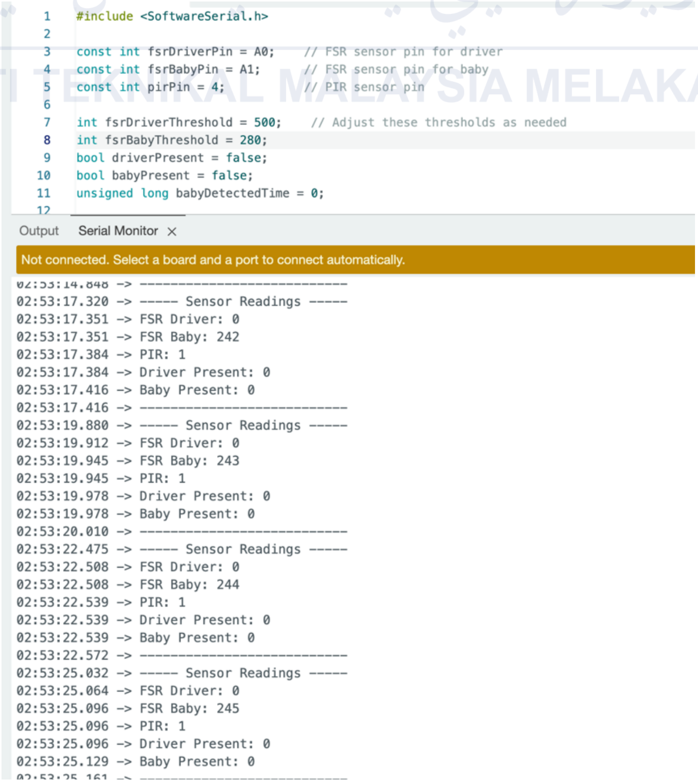

Figure 4.11: Serial Monitor Readings of Child Detection System Testing

The Serial Monitor output shown in Figure 4.11 indicates that the FSR Baby sensor is reading values of 242 and above, and the PIR sensor simultaneously detects movement. However, the "Baby Present" status remains 0 because the threshold for the FSR Baby sensor is set at 280. This threshold is designed to differentiate between a person and an object: if the FSR reading exceeds 280, it indicates the presence of a person; if it is lower, it suggests an object. Consequently, the system does not recognize the presence of a baby because the detected weight or pressure is too low to meet the threshold, which is calibrated for an average baby weight of 2.5 kg.

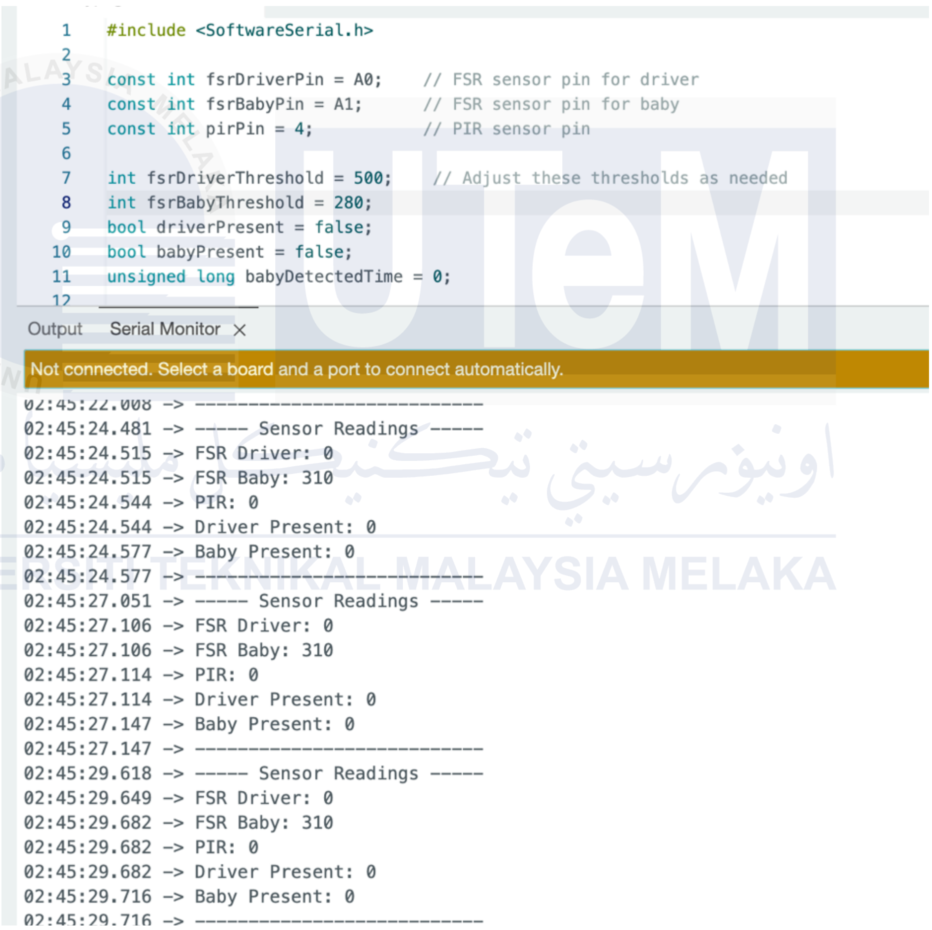

Figure 4.12: Serial Monitor Readings of Child Detection System Testing

The Serial Monitor output shown in Figure 4.12 indicates that the FSR Baby sensor is reading values of 310 and above. However, the PIR sensor is not detecting any movement (PIR sensor reading is 0). As a result, the "Baby Present" status remains 0. The threshold for the FSR Baby sensor is set at 280 to differentiate between a person and an object: if the FSR reading exceeds 280, it indicates the presence of a person; if it is lower, it suggests an object. Since the PIR sensor does not detect any movement, the system concludes that there is no baby or human present, but rather an object.

```
#include <SoftwareSerial.h>

const int fsrDriverPin = A0;    // FSR sensor pin for driver
const int fsrBabyPin = A1;      // FSR sensor pin for baby
const int pirPin = 4;           // PIR sensor pin

int fsrDriverThreshold = 500;    // Adjust these thresholds as needed
int fsrBabyThreshold = 280;
bool driverPresent = false;
bool babyPresent = false;
unsigned long babyDetectedTime = 0;
```

Output   Serial Monitor ×

Not connected. Select a board and a port to connect automatically.

```
02:55:18.558 -> ----- Sensor Readings -----
02:55:18.590 -> FSR Driver: 0
02:55:18.623 -> FSR Baby: 277
02:55:18.623 -> PIR: 1
02:55:18.623 -> Driver Present: 0
02:55:18.656 -> Baby Present: 0
02:55:18.656 -> ---------------------------
02:55:21.152 -> ----- Sensor Readings -----
02:55:21.152 -> FSR Driver: 0
02:55:21.183 -> FSR Baby: 286
02:55:21.183 -> PIR: 1
02:55:21.216 -> Driver Present: 0
02:55:21.216 -> Baby Present: 1
02:55:21.247 -> ---------------------------
02:55:23.709 -> ----- Sensor Readings -----
02:55:23.741 -> FSR Driver: 0
02:55:23.741 -> FSR Baby: 292
02:55:23.773 -> PIR: 1
02:55:23.773 -> Driver Present: 0
02:55:23.806 -> Baby Present: 1
02:55:23.806 -> ---------------------------
02:55:26.272 -> ----- Sensor Readings -----
02:55:26.304 -> FSR Driver: 0
02:55:26.336 -> FSR Baby: 295
02:55:26.336 -> PIR: 1
02:55:26.368 -> Driver Present: 0
02:55:26.368 -> Baby Present: 1
```


Figure 4.13: Serial Monitor Readings of Child Detection System Testing

The Serial Monitor output shown in Figure 4.13 indicates that the FSR Baby sensor is finally reading values above 280, and the PIR sensor simultaneously detects movement. As a result, the "Baby Present" status is set to 1. The threshold for the FSR Baby sensor is set at 280 to differentiate between a person and an object: if the FSR reading exceeds 280, it indicates the presence of a person; if it is lower, it suggests an object. Since both the FSR sensor and the PIR sensor confirm the presence and movement, the system correctly identifies that there is a baby present in the car.

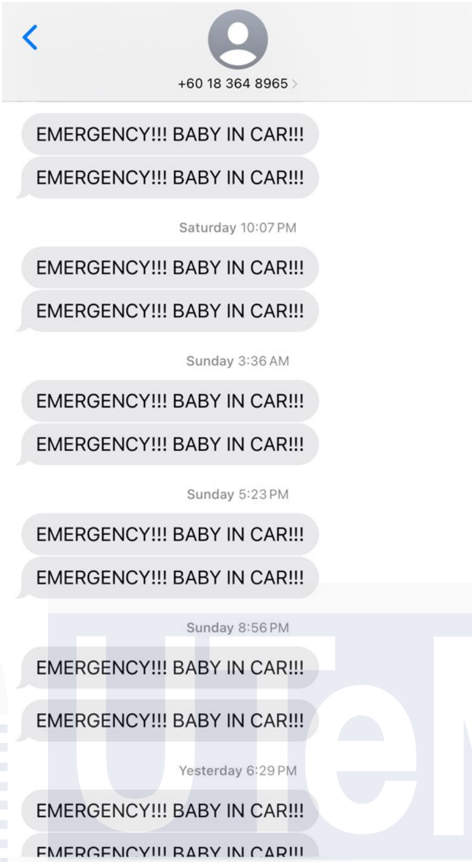

Figure 4.14: SMS Notification sent by the GSM SIM800L Module

The repeated SMS notifications as shown in Figure 4.14 with messages of "EMERGENCY!!! BABY IN CAR!!!" indicate that the SIM800L GSM module was effectively sending alerts as intended. This consistent transmission of emergency messages confirms the system's reliability in detecting a potentially dangerous situation and promptly notifying the designated contacts while also sounding the buzzer as an alarm to alert nearby people. The ability of the system to repeatedly send these alerts demonstrates its robustness and efficiency in real-time emergency communication, fulfilling the key objective of ensuring immediate awareness and response to a child left unattended in a car.

Figure 4.15: The Child Detection System in the Car

The Child Detection System depicted in Figure 4.15 showcases a portable and user-friendly design tailored for easy integration into various vehicles. Its compact size and lightweight nature ensure effortless placement anywhere in the car, whether under seats, in trunk spaces, or other discreet locations. This versatility not only maximizes convenience but also minimizes interference with passengers and vehicle operations.

The system was designed while taking into account the compatibility with different users, can be placed or moved into any conventional cars as it is a portable device, and the system is straightforward to install and operate, requiring minimal setup. It combines advanced sensors like the FSR Baby sensor for weight detection and a PIR sensor for motion, ensuring robust detection capabilities. When the FSR sensor detects a weight exceeding the set threshold of 280 while the PIR sensor senses motion, the system promptly alerts users to the presence of a child in the car, prioritizing safety with reliable functionality.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

As for the conclusion, this project demonstrated the design concept on the smart detection system for unattended children in vehicles. The detection system is developed by utilizing Arduino Uno as the system microcontroller, FSR sensor as to sense the presence of the child and also the absence of the driver, PIR sensor to sense the motion of the child, the buzzer represents the car alarm system to alert nearby people, and the GSM module to send alerts to parents and guardians. The system was designed successfully, and the sensors were tested and proven to operate as intended. The system has the potential to be much more efficient by using more advanced and costly materials as it is an important system that should be used by every car. However, due to cost constraints, the system was designed following the tight budget and some of the components used could be identified as outdated and has a slow processing time. In a nutshell, the system was able to detect the presence of the baby, the absence of the driver, the movement of the baby, send the SMS notifications to the guardians and sound the buzzer to alert nearby people. The objectives of this project involves designing a system that combines multiple sensors and data sources for reliable detection, developing an alert mechanism that informs guardians responders promptly and validating the system's accuracy and reliability in various conditions, and it can be concluded that each and every one of the objectives for this project were achieved successfully.

5.2 Future Works

For future improvements, the system could be enhanced by incorporating additional sensors to provide more comprehensive monitoring and safety features. For instance, adding a thermal sensor that can sense body heat radiation without the need of any contact could help detect the body heat of the baby. This would be particularly useful since the PIR sensor used in this project may not detect a stationary human being. The thermal sensor can accurately measure the infrared radiation emitted by the baby's body, allowing the system to reliably detect the baby's presence even when they are not moving. This added capability would significantly enhance the reliability of the system, ensuring that the baby is always detected, especially if the baby is sleeping when the driver has left them unattended.

To improve power management and ensure the system remains operational during critical times, a rechargeable battery system connected to a charging module could be implemented. This system would allow the device to switch between using the car's electrical power when the car is on and the rechargeable batteries when the car is off. The car's electrical system would charge the batteries through the charging module when the engine is running, ensuring the batteries are always ready for use when the car is off. Also, integrating a small solar panel into the system can provide an additional power source, further enhancing its sustainability. The solar panel would harness sunlight to charge the batteries, ensuring that the system can operate for extended periods without draining the car battery or the rechargeable battery. This setup would ensure that the system operates efficiently, particularly when the car is off, as this is when it is most crucial for the system to function and alert in the absence of the driver.

Moreover, implementing a sleep mode for the system could further conserve battery life. The system could enter sleep mode when both the driver and baby are absent, or when the baby and driver are both present in the car. By reducing power consumption during these times, the system's lifespan would be extended, and the batteries would be preserved for when the driver is absent, which is the critical time for the system to be active. This improvement would ensure the system is always ready to protect the baby while efficiently managing its power resources.

Furthermore, Adding Wi-Fi or Bluetooth connectivity to the system would not only enable communication with a smartphone app or other devices but also provide real-time alerts and notifications to the driver or guardians, thereby offering an additional layer of safety. While the GSM module used in this project can serve this purpose, incorporating more IoT connectivity options can further enhance the system's efficiency and expand its capabilities. By leveraging Wi-Fi or Bluetooth, the system can offer faster and more reliable communication, enabling instant updates and remote monitoring, ultimately ensuring the baby's safety and peace of mind for the caregivers.

Lastly, ongoing research and development could focus on enhancing the system's user interface and control features. Introducing a user-friendly interface, such as a smartphone application or a dashboard display, could empower caregivers with real-time insights and control over the system. This interface could provide detailed information on the baby's status, battery levels, and system alerts, allowing caregivers to respond promptly to any detected risks or emergencies.

By integrating these enhancements, the system can become a reliable safety solution for children left unattended in vehicles. Advanced sensors detect the baby's presence accurately, while Wi-Fi or Bluetooth connectivity provides real-time alerts to caregivers. Solar charging and battery management ensure continuous operation, and an improved user interface enables swift response to emergencies. These upgrades elevate the system's effectiveness, offering caregivers peace of mind during car journeys with their children.

REFERENCES

[1] Sidhutanzeel, ‘importance of cars in life’, Medium. Accessed: Dec. 20, 2023. [Online]. Available: https://medium.com/@sidhutanzeel5/importance-of-cars-in-life-ce90b3aaf3de

[2] ‘The Must Have Car Safety Features in Modern Cars’, Budget Direct. Accessed: Dec. 20, 2023. [Online]. Available: https://www.budgetdirect.com.au/car-insurance/guides/road-safety/car-safety-features-in-modern-cars.html

[3] ‘What is a Smart Sensor and How Does it Work? | Definition from TechTarget’, IoT Agenda. Accessed: Dec. 20, 2023. [Online]. Available: https://www.techtarget.com/iotagenda/definition/smart-sensor

[4] M. Soori, B. Arezoo, and R. Dastres, ‘Artificial intelligence, machine learning and deep learning in advanced robotics, a review’, *Cogn. Robot.*, vol. 3, pp. 54–70, Jan. 2023, doi: 10.1016/j.cogr.2023.04.001.

[5] J. Null, ‘No Heat Stroke’, No Heat Stroke. Accessed: Jun. 21, 2024. [Online]. Available: https://www.noheatstroke.org

[6] M. A. M. ZOKI, ‘Eight-month-old baby dies after being left inside car for 10 hours’, Sinar Daily. Accessed: Dec. 20, 2023. [Online]. Available: https://www.sinardaily.my/article/209935/malaysia/national/eight-month-old-baby-dies-after-being-left-inside-car-for-10-hours

[7] A. Zulkefli, ‘Boy dies after being left for more than 4 hours in principal’s car’, astroawani. Accessed: Dec. 20, 2023. [Online]. Available: https://www.astroawani.com/berita-malaysia/boy-dies-after-being-left-more-4-hours-principals-car-288460

[8] T. N. Alagesh, ‘(Update) Baby dies after being left inside car’, NST Online. Accessed: Dec. 20, 2023. [Online]. Available: https://www.nst.com.my/news/nation/2020/02/566449/update-baby-dies-after-being-left-inside-car

[9] N. M. Z. Hashim, H. H. Basri, A. Jaafar, M. Z. Abd Aziz, A. SALLEH, and A. S. Ja’afar, ‘Child in car alarm system using various sensors’, *ARPN J. Eng. Appl. Sci.*, vol. 9, no. 9, pp. 1653–1658, Sep. 2014.

[10] 'What is a Microcontroller and How Does it Work?', IoT Agenda. Accessed: Dec. 21, 2023. [Online]. Available: https://www.techtarget.com/iotagenda/definition/microcontroller

[11] A. P. Murdan, V. Bucktowar, V. Oree, and M. P. Enoch, 'Low-cost bus seating information technology system', *IET Intell. Transp. Syst.*, vol. 14, no. 10, pp. 1303–1310, 2020, doi: 10.1049/iet-its.2019.0529.

[12] S. Arya and P. Sharma, 'Sensors-Based Automatic Human Body Detection and Prevention System to Avoid Entrapment Casualties inside a Vehicle', in *Machine Learning Adoption in Blockchain-Based Intelligent Manufacturing: Theoretical Basics, Applications, and Challenges*, 2022, pp. 157–169. doi: 10.1201/9781003252009-10.

[13] Brave Learn, 'Arduino Interfacing with PIR Motion Sensor | Brave Learn', Brave Learn. Accessed: Jan. 17, 2024. [Online]. Available: https://bravelearn.com/arduino-interfacing-with-pir-motion-sensor/

[14] M. D. Amin, S. E. A. Rahim, N. S. M. Khairi, M. Z. Ayob, and I. B. Ahmad, 'The Development of Detection System for Bus Passenger Monitoring', vol. 4, pp. 50–57, Dec. 2023.

[15] B. Webmaster, 'How Do Motion Sensors Work: A Guide', Bay Alarm. Accessed: Jan. 18, 2024. [Online]. Available: https://www.bayalarm.com/blog/how-do-motion-sensors-work-a-guide/

[16] D. Jost, 'What is a Motion Sensor? | Fierce Electronics'. Accessed: Jan. 18, 2024. [Online]. Available: https://www.fierceelectronics.com/sensors/what-a-motion-sensor

[17] U. S, A. B. T S R, S. V, and G. B. K, 'IoT based Approach to Estimate the Vacant Seats Available', in *2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA)*, Sep. 2022, pp. 354–358. doi: 10.1109/ICIRCA54612.2022.9985657.

[18] J. Moderl, F. Pernkopf, and K. Witrisal, 'Car Occupancy Detection Using Ultra-Wideband Radar', in *2021 18th European Radar Conference (EuRAD)*, London, United Kingdom: IEEE, Apr. 2022, pp. 313–316. doi: 10.23919/EuRAD50154.2022.9784505.

[19] M. Hämäläinen *et al.*, 'Ultra-Wideband Radar-Based Indoor Activity Monitoring for Elderly Care', *Sensors*, vol. 21, no. 9, Art. no. 9, Jan. 2021, doi: 10.3390/s21093158.

[20] H. Tan and Z. Yang, 'In-Vehicle Occupant Detection Based on Zone Source Feature Using MM-Wave MIMO Radar', in *2021 CIE International Conference on Radar (Radar)*, Haikou, Hainan, China: IEEE, Dec. 2021, pp. 1392–1396. doi: 10.1109/Radar53847.2021.10028477.

[21] 'Pressure Sensors: Types, Use Cases, and Definition', onupkeep. Accessed: Jan. 18, 2024. [Online]. Available: https://www.upkeep.com/learning/pressure-sensor/

[22] M. R. Maepa and M. N. Moeti, 'Smart Occupancy Monitoring System Using Iot Technologies for South African Minibus Taxis', *SSRN Electron. J.*, 2023, doi: 10.2139/ssrn.4663206.

[23] R. Sharma and Y. Sawant, 'Smart Bus: Seat Occupancy Detection System', *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 03, no. 05, pp. 2430–2435, May 2021.

[24] R. P. Kumar, D. Melcher, P. Buttolo, and Y. Jia, 'Monitoring of Occupant States in Autonomous Vehicles Using Capacitance-Sensing Imaging', *IEEE Sens. J.*, vol. 20, no. 21, pp. 12914–12927, Nov. 2020, doi: 10.1109/JSEN.2020.3003312.

[25] 'What is IoT (Internet of Things) and How Does it Work? | Definition from TechTarget', IoT Agenda. Accessed: Jan. 18, 2024. [Online]. Available: https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT

[26] S. Bharadwaj K S, R. A. S, P. A. N, S. Bhat, and S. K. V, 'Child Presence Detection and Alerting System in an Unmanned Car', *Perspect. Commun. Embed.-Syst. Signal-Process. - PiCES*, vol. 4, no. 11, pp. 284–288, Mar. 2021, doi: 10.5281/zenodo.4592826.

[27] M. N. Shamuri and R. Alias, 'Design of Tracking System for Kids', *Evol. Electr. Electron. Eng.*, vol. 2, no. 2, pp. 385–392, Nov. 2021.

[28] X. Zeng, B. Wang, C. Wu, S. Deepika Regani, and K. J. Ray Liu, 'Intelligent Wi-Fi Based Child Presence Detection System', in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2022, pp. 11–15. doi: 10.1109/ICASSP43922.2022.9747420.

[29] 'What is an Arduino? - SparkFun Learn'. Accessed: Jun. 23, 2024. [Online]. Available: https://learn.sparkfun.com/tutorials/what-is-an-arduino/all

[30] ‘In-Depth: Interfacing Force Sensing Resistor (FSR) with Arduino’, Last Minute Engineers. Accessed: Jun. 23, 2024. [Online]. Available: https://lastminuteengineers.com/fsr-arduino-tutorial/

[31] ‘How HC-SR501 PIR Sensor Works & How To Interface It With Arduino’, Last Minute Engineers. Accessed: Jun. 23, 2024. [Online]. Available: https://lastminuteengineers.com/pir-sensor-arduino-tutorial/

[32] ‘In-Depth: Send Receive SMS & Call with SIM800L GSM Module & Arduino’, Last Minute Engineers. Accessed: Jun. 23, 2024. [Online]. Available: https://lastminuteengineers.com/sim800l-gsm-module-arduino-tutorial/

[33] Peter, ‘The Ultimate Guide to Powering Your Arduino Uno Board’, Tech Explorations. Accessed: Jun. 23, 2024. [Online]. Available: https://techexplorations.com/blog/arduino/guide-to-arduino-uno-r3-power

[34] Gus, ‘Arduino Force Sensing Resistor (FSR) - Pi My Life Up’, Pi My Life Up. [Online]. Available: https://pimylifeup.com/arduino-force-sensing-resistor/

[35] Dottie, ‘Small for Gestational Age (SGA)’, PORTAL MyHEALTH. Accessed: Jul. 01, 2024. [Online]. Available: https://myhealth23.primuscore.com/en/small-for-gestational-age-sga/

[36] M. Azmi *et al.*, ‘Body mass index (BMI) of adults: Findings of the Malaysian Adult Nutrition Survey (MANS)’, *Malays. J. Nutr.*, vol. 15, pp. 97–119, Sep. 2009.

[37] S. Campbell, ‘How to Setup Passive Infrared (PIR) Motion Sensors on the Arduino’, Circuit Basics. Accessed: Jan. 22, 2024. [Online]. Available: https://www.circuitbasics.com/how-to-setup-passive-infrared-pir-motion-sensors-on-the-arduino/

[38] ‘Home Security System using GSM’, Hackster.io. Accessed: Jun. 22, 2024. [Online]. Available: https://www.hackster.io/Daini/home-security-system-using-gsm-3f9de4

[39] A. Hazizan, N. A. Md Lazam, and N. I. Hassan, ‘Development of Child Safety Car Alert System Using Arduino and GSM Module’, *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 834, no. 1, p. 012071, Apr. 2020, doi: 10.1088/1757-899X/834/1/012071.

APPENDICES

APPENDIX A FYP 1 AND FYP 2 GANTT CHART

Task	Week													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Selection of project title	■													
Submission of project title	■													
Analyze the problem and objectives		■												
Literature review		■	■	■	■	■	■	■	■	■	■	■	■	■
Experiment 1			■	■										
Proposed methodology					■	■	■	■						
Design prototype								■	■	■	■			
FYP 1 Seminar presentation											■			
Experiment 2											■			
Experiment 3											■			
Report progress and submission										■	■	■	■	■

Task	Week														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Literature review	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Device hardware calibration and testing	■	■													
Experiment 4			■	■											
Experiment 5				■	■										
Testing in car						■	■	■	■	■					
Data analysis										■	■	■	■	■	
FYP 2 presentation															■
FYP 2 report progress										■	■	■	■	■	■

APPENDIX B PROGRAMMING CODE FSR SENSOR TESTING

```
int pressureAnalogPin = 0; // pin where our pressure pad is located
int pressureReading; // variable for storing our reading
int noPressure = 100; // max value for no pressure on the pad
int lightPressure = 200; // max value for light pressure on the pad
int mediumPressure = 400; // max value for medium pressure on the pad

void setup(void) {
  Serial.begin(9600);
  Serial.println("Pressure Pad Readings");
  Serial.println("-----------------------");
  Serial.println("Reading\t\tPressure Level");
}

void loop(void) {
  pressureReading = analogRead(pressureAnalogPin);
  Serial.print(pressureReading);
  Serial.print("\t\t");

  if (pressureReading < noPressure) {
    Serial.println("No Pressure");
  } else if (pressureReading < lightPressure) {
    Serial.println("Light Pressure");
  } else if (pressureReading < mediumPressure) {
    Serial.println("Medium Pressure");
  } else {
    Serial.println("High Pressure");
  }

  // Print the pressure reading again for the Serial Plotter
  Serial.println(pressureReading);

  delay(3000); // wait for 3 seconds
}
```

APPENDIX C PROGRAMMING CODE FSR BABY AND FSR DRIVER SENSOR TESTING

```
// Pin assignments
const int pressureAnalogPinDriver = 0; // Pin where FSR Driver is connected
const int pressureAnalogPinBaby = 1;   // Pin where FSR Baby is connected

// Pressure reading variables
int pressureReadingDriver; // Variable for storing reading from FSR Driver
int pressureReadingBaby;   // Variable for storing reading from FSR Baby

// Pressure thresholds
const int noPressure = 100;       // Max value for no pressure on the pad
const int lightPressure = 200;    // Max value for light pressure on the
pad
const int mediumPressure = 400;   // Max value for medium pressure on the
pad

void setup() {
  Serial.begin(9600);
  Serial.println("Pressure Pad Readings");
  Serial.println("----------------------");
  Serial.println("FSR\t\tReading\t\tPressure Level");
}

void loop() {
  // Read pressure values
  pressureReadingDriver = analogRead(pressureAnalogPinDriver);
  pressureReadingBaby = analogRead(pressureAnalogPinBaby);

  // Display readings for FSR Driver
  Serial.print("FSR Driver\t");
  Serial.print(pressureReadingDriver);
  Serial.print("\t\t");
  Serial.println(getPressureLevel(pressureReadingDriver));

  // Display readings for FSR Baby
  Serial.print("FSR Baby\t");
  Serial.print(pressureReadingBaby);
  Serial.print("\t\t");
  Serial.println(getPressureLevel(pressureReadingBaby));

  delay(2000); // Wait for 2 seconds
```

```
}

// Function to determine the pressure level
String getPressureLevel(int pressureReading) {
  if (pressureReading < noPressure) {
    return "No Pressure";
  } else if (pressureReading < lightPressure) {
    return "Light Pressure";
  } else if (pressureReading < mediumPressure) {
    return "Medium Pressure";
  } else {
    return "High Pressure";
  }
}
```

APPENDIX D PROGRAMMING CODE FSR CONVERSION

```
int pressureAnalogPin = 0; // pin where our pressure pad is located
int pressureReading; // variable for storing our reading
float pressureInKg; // variable for storing the pressure in kg
float pressureInNewton; // variable for storing the pressure in Newtons
const int numReadings = 10; // number of readings to average

int readings[numReadings]; // array to store readings
int readIndex = 0; // index of the current reading
int total = 0; // running total of the readings
int average = 0; // average of the readings

// Constants for force calculation
const float VCC = 5.0; // supply voltage
const float ADC_RESOLUTION = 1023.0; // ADC resolution
const float CALIBRATION_CONSTANT = 0.07875; // derived from updated
calibration

void setup(void) {
  Serial.begin(9600);
  Serial.println("Pressure Pad Readings");
  Serial.println("----------------------");
  Serial.println("Reading\t\tNewton\t\tKg");

  // initialize all the readings to 0
  for (int i = 0; i < numReadings; i++) {
    readings[i] = 0;
  }
}

void loop(void) {
  // subtract the last reading from the total
  total = total - readings[readIndex];

  // read the sensor
  readings[readIndex] = analogRead(pressureAnalogPin);

  // add the new reading to the total
  total = total + readings[readIndex];

  // advance to the next position in the array
  readIndex = readIndex + 1;
```

```
  // if we're at the end of the array, wrap around to the beginning
  if (readIndex >= numReadings) {
    readIndex = 0;
  }

  // calculate the average
  average = total / numReadings;

  // calculate the pressure in Newtons and kilograms
  pressureInNewton = CALIBRATION_CONSTANT * average;
  pressureInKg = pressureInNewton / 9.80665; // Direct conversion from
Newton to kg

  // print the readings in a table format
  Serial.print(average);
  Serial.print("\t\t");
  Serial.print(pressureInNewton, 2); // print the pressure in Newtons with
2 decimal places
  Serial.print("\t\t");
  Serial.print(pressureInKg, 4); // print the pressure in kg with 4 decimal
places
  Serial.println();

  delay(2000); // wait for 2 seconds
}
```

APPENDIX E PROGRAMMING CODE PIR SENSOR TESTING

```
int sensorPin = 10;
int pirState = LOW;

void setup() {
  Serial.begin(9600);
  pinMode(sensorPin, INPUT);
}

void loop() {
  int value = digitalRead(sensorPin);
  pirState = digitalRead(sensorPin); // read input value
  Serial.println(pirState); // print out the state of the sensor
  delay(10);
}
```

APPENDIX F PROGRAMMING CODE GSM MODULE WITH BUZZER TESTING

```
#include <SoftwareSerial.h>

SoftwareSerial mySerial(3, 2);
int buzzerPin = 8; // Define the buzzer pin

void setup()
{
  pinMode(buzzerPin, OUTPUT); // Set buzzer pin as output

  Serial.begin(9600);
  mySerial.begin(9600);

  Serial.println("Initializing...");
  delay(1000);

  mySerial.println("AT");
  updateSerial();

  mySerial.println("AT+CMGF=1");
  updateSerial();

  mySerial.println("AT+CMGS=\"+60126617060\""); // Enter your phone number
here (prefix country code)
  updateSerial();

  mySerial.print("EMERGENCY!!! BABY IN CAR!!!"); // Enter your message here
  updateSerial();

  mySerial.write(26);

  // Activate the buzzer after sending the SMS
  Serial.println(" Activating Alarm...");
  digitalWrite(buzzerPin, HIGH);
  delay(1000); // Buzzer on for 1 second
  digitalWrite(buzzerPin, LOW); // Turn off the buzzer
  Serial.println("ALARM WAS SOUND");
}

void loop()
{
```

```
}

void updateSerial()
{
  delay(500);
  while (Serial.available())
  {
    mySerial.write(Serial.read()); // Forward what Serial received to
Software Serial Port
  }
  while(mySerial.available())
  {
    Serial.write(mySerial.read()); // Forward what Software Serial received
to Serial Port
  }
}
```

APPENDIX G PROGRAMMING CODE COMPLETE SYSTEM

```
#include <SoftwareSerial.h>

const int fsrDriverPin = A0;    // FSR sensor pin for driver
const int fsrBabyPin = A1;      // FSR sensor pin for baby
const int pirPin = 4;           // PIR sensor pin

int fsrDriverThreshold = 500;    // Adjust these thresholds as needed
int fsrBabyThreshold = 280;
bool driverPresent = false;
bool babyPresent = false;
unsigned long babyDetectedTime = 0;

SoftwareSerial mySerial(3, 2);

void setup() {
  Serial.begin(9600);
  pinMode(pirPin, INPUT);

  mySerial.begin(9600);
  delay(1000);
}

void loop() {
  static unsigned long lastPrintTime = 0;
  unsigned long currentTime = millis();

  if (currentTime - lastPrintTime >= 2500) { // Print every 2.5 seconds
    lastPrintTime = currentTime;

    int fsrDriverValue = analogRead(fsrDriverPin);
    int fsrBabyValue = analogRead(fsrBabyPin);
    int pirValue = digitalRead(pirPin);

    // Determine presence of driver and baby
    if (fsrDriverValue > fsrDriverThreshold) {
      driverPresent = true;
    } else {
      driverPresent = false;
    }

    babyPresent = (fsrBabyValue > fsrBabyThreshold) && pirValue;
```

```
    // Print sensor values and presence status in a list form downwards
    Serial.println("----- Sensor Readings -----");
    Serial.println("FSR Driver: " + String(fsrDriverValue));
    Serial.println("FSR Baby: " + String(fsrBabyValue));
    Serial.println("PIR: " + String(pirValue));
    Serial.println("Driver Present: " + String(driverPresent));
    Serial.println("Baby Present: " + String(babyPresent));
    Serial.println("---------------------------");

    // If driver is absent but baby is present, record the time
    if (!driverPresent && babyPresent && !babyDetectedTime) {
      babyDetectedTime = currentTime;
    } else if (driverPresent || !babyPresent) {
      babyDetectedTime = 0;
    }

    // Check if SMS should be sent after 15 seconds
    if (babyPresent && !driverPresent && (currentTime - babyDetectedTime >=
5000)) {
      sendSMS();
    }
  }

  delay(100); // Short delay to improve responsiveness
}

void sendSMS() {
  mySerial.println("AT");
  delay(1000);
  mySerial.println("AT+CMGF=1");
  delay(1000);
  mySerial.println("AT+CMGS=\"+60126617060\""); // enter your phone number
here (prefix country code)
  delay(1000);
  mySerial.print("EMERGENCY!!! BABY IN CAR!!!"); // enter your message here
  delay(1000);
  mySerial.write(26);
}
```