UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Universiti Teknikal Malaysia Melaka
Hang Tuah Jaya,
76100 Durian Tunggal,
Melaka, Malaysia.

+606 270 1000
+606 270 1022
www.utem.edu.my

FAKULTI TEKNOLOGI DAN KEJURUTERAAN MEKANIKAL

Tel : +606 270 1184 | Faks : +606 270 1064

Rujukan Kami (Our Ref):
Rujukan Tuan (Your Ref):
Tarikh (Date): 31 Januari 2021

Ketua Perpustakaan
Perpustakaan Laman Hikmah
Universiti Teknikal Malaysia Melaka

Melalui

Dekan
Fakulti Teknologi dan Kejuruteraan Mekanikal
Universiti Teknikal Malaysia Melaka

Tuan

PENGKELASAN TESIS SEBAGAI TERHAD BAGI TESIS PROJEK SARJANA MUDA

Dengan segala hormatnya merujuk kepada perkara di atas.

2. Dengan ini, dimaklumkan permohonan pengkelasan tesis yang dilampirkan sebagai TERHAD untuk tempoh **LIMA** tahun dari tarikh surat ini. Butiran lanjut laporan PSM tersebut adalah seperti berikut:

Nama pelajar: KHAIRUL IQMAL BIN ESA (B092010124)
Tajuk Tesis: DESIGN OF CAR PARKING DISTANCE SENSOR BY USING ARDUINO

3. Hal ini adalah kerana IANYA MERUPAKAN PROJEK YANG DITAJA OLEH SYARIKAT LUAR DAN HASIL KAJIANNYA ADALAH SULIT.

Sekian, terima kasih.

"BERKHIDMAT UNTUK NEGARA"
"KOMPETENSI TERAS KEGEMILANGAN"

Saya yang menjalankan amanah,

NOR FAIZAH BINTI HAMINUDIN
Penyelia Utama/ Pensyarah Kanan
Fakulti Teknologi dan Kejuruteraan Mekanikal
Universiti Teknikal Malaysia Melaka

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

BORANG PENGESAHAN STATUS LAPORAN PROJEK SARJANA MUDA

TAJUK: DESIGN OF CAR PARKING DISTANCE SENSOR BY USING ARDUINO

SESI PENGAJIAN: **2023-2024 Semester 1**

Saya **KHAIRUL IQMAL BIN ESA**

mengaku membenarkan tesis ini disimpan di Perpustakaan Universiti Teknikal Malaysia Melaka (UTeM) dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hak milik Universiti Teknikal Malaysia Melaka dan penulis.
2. Perpustakaan Universiti Teknikal Malaysia Melaka dibenarkan membuat salinan untuk tujuan pengajian sahaja dengan izin penulis.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. **Sila tandakan (✓)

☐ TERHAD (Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia sebagaimana yang termaktub dalam AKTA RAHSIA RASMI 1972)

☐ SULIT (Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)

☑ TIDAK TERHAD

Disahkan oleh:

Alamat Tetap:

NO.132, JLN INDAH 6, TMN PERTAMA INDAH, SUNGAI ABONG 84000 MUAR JOHOR

Cop Rasmi:

NOR FAIZAH BINTI HAMINUDIN
Pensyarah
Fakulti Teknologi Dan Kejuruteraan Mekanikal
Universiti Teknikal Malaysia Melaka (UTeM)

Tarikh: 19/1/2024

Tarikh: 19 JAN 2024

** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh laporan PSM ini perlu dikelaskan sebagai SULIT atau TERHAD.

DESIGN OF CAR PARKING DISTANCE SENSOR BY USING ARDUINO

KHAIRUL IQMAL BIN ESA
B092010124

BACHELOR OF MECHANICAL ENGINEERING TECHNOLOGY (AUTOMOTIVE TECHNOLOGY) WITH HONOURS

2024

Faculty of Mechanical Technology & Engineering

DESIGN OF CAR PARKING DISTANCE SENSOR BY USING ARDUINO

Khairul Iqmal Bin Esa

Bachelor of Mechanical Engineering Technology (Automotive Technology) with Honours

2024

DESIGN OF CAR PARKING DISTANCE SENSOR BY USING ARDUINO

KHAIRUL IQMAL BIN ESA

**A thesis submitted
in fulfillment of the requirements for the degree of
Bachelor of Mechanical Engineering Technology (Automotive Technology) with Honours**

Faculty of Mechanical Technology and Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

DECLARATION

I declare that this Choose an item. entitled “Design of car parking distance sensor by using Arduino” 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 candidature of any other degree.

Signature :

Name : Khairul Iqmal Bin Esa

Date : 12/1/2024

APPROVAL

I hereby declare that I have checked this thesis, and in my opinion, this thesis is adequate in terms of scope and quality for the award of the Bachelor of Mechanical Engineering Technology (Automotive) with Honours.

Signature	:	
Supervisor Name	:	NOR FAIZAH BINTI HAMINUDIN
Date	:	12/1/2024

DEDICATION

I express my heartfelt dedication of this thesis to my beloved parents, Mr. Esa Bin Yusof and Mrs. Zalinda Binti Hassan, who have consistently served as exceptional role models in my life. Their unwavering moral, spiritual, emotional, and financial support have been instrumental in my journey. I am immensely grateful for their constant inspiration and the strength they provide me during times of weakness.

Additionally, I extend this dedication to my steadfast friends, who have been by my side since the beginning, offering unwavering support and encouragement throughout the research process. They have steadfastly believed in my abilities and provided me with the strength to overcome obstacles.

Furthermore, I would like to extend this dedication to my supervisor and groupmate, who have been my pillars of support throughout the entire project duration, standing by me through the highs and lows.

ABSTRACT

The increasing number of vehicles and limited availability of parking spaces have made parking a critical issue in transportation and infrastructure development. The aim is to implement a parking distance control system using advanced technology, such as sensors or cameras, specifically tailored for terrace housing areas. To overcome these challenges, an effective parking distance sensor system is necessary. This system should provide accurate measurements and real-time feedback, optimising sensor placement to accommodate different vehicle dimensions and considering factors like blind spots. The main objective of this project is to design a car parking distance sensor using ultrasonic technology based on the Arduino Uno Rev3 ATmega328P microcontroller with Bluetooth module HC-06 as the connectivity of IoT system. The specific goals include designing IoT systems for the car parking distance sensor, determining the potential distance between different vehicle types and the distance sensor, and conducting comprehensive testing and analysis to ensure the safety and effectiveness of the car parking system. The scope of the research involves identifying potential IoT systems for the car parking distance sensor, designing a questionnaire to gather information on car parking experiences, proposing design ideas for sensor positions and configurations based on questionnaire results, selecting appropriate sensor distances and components, conducting testing on different types of cars (SEDAN, SUV, and MPV), determining suitable sensor positions for all vehicle types, and performing time and range testing using Arduino Uno and ultrasonic sensors. By implementing the designed car parking distance sensor system, this project aims to mitigate congestion and parking problems in residential areas, enhance safety, and contribute to efficient parking space management. Overall, this project addresses the pressing issues related to parking in terraced residential areas, offering an effective distance-sensing solution to improve the parking experience, enhance safety, and optimise the management of parking spaces in housing areas.

ABSTRAK

Jumlah kenderaan yang semakin meningkat dan terhadnya ruang letak tempat letak kereta telah menjadikan isu tempat letak kereta sebagai isu penting dalam pembangunan pengangkutan dan infrastruktur. Matlamatnya adalah untuk melaksanakan sistem kawalan jarak tempat letak kereta menggunakan teknologi terkini, seperti sensor atau kamera, yang khusus direka untuk kawasan perumahan berteres. Bagi mengatasi cabaran ini, sistem sensor jarak tempat letak kereta yang berkesan diperlukan. Sistem ini perlu menyediakan pengukuran yang tepat dan maklum balas secara waktu nyata, mengoptimumkan penempatan sensor untuk menampung dimensi kenderaan yang berbeza dan mempertimbangkan faktor-faktor seperti titik buta. Objektif utama projek ini adalah untuk mereka bentuk sensor jarak tempat letak kereta menggunakan teknologi ultrasonik berdasarkan mikrokontroler Arduino Uno Rev3 ATmega328P dan Bluetooth module HC-06 sebagai pertalian dalam sistem IoT. Matlamat khusus termasuk mereka bentuk sistem IoT untuk sensor jarak tempat letak kereta, menentukan jarak potensial antara jenis kenderaan yang berbeza dan sensor jarak, dan menjalankan ujian dan analisis menyeluruh untuk memastikan keselamatan dan keberkesanan sistem tempat letak kereta. Skop penyelidikan melibatkan mengenal pasti sistem IoT yang sesuai untuk sensor jarak tempat letak kereta, mereka bentuk soal selidik untuk mengumpul maklumat tentang pengalaman tempat letak kereta, mencadangkan idea reka bentuk untuk kedudukan dan konfigurasi sensor berdasarkan keputusan soal selidik, memilih jarak dan komponen sensor yang sesuai, menjalankan ujian pada kenderaan berbeza (SEDAN, SUV, dan MPV), menentukan kedudukan sensor yang sesuai untuk semua jenis kenderaan, dan menjalankan ujian masa dan jarak menggunakan Arduino Uno dan sensor ultrasonik. Dengan melaksanakan sistem sensor jarak tempat letak kereta yang direka bentuk, projek ini bertujuan untuk mengurangkan kesesakan dan masalah tempat letak kereta di kawasan perumahan, meningkatkan keselamatan, dan menyumbang kepada pengurusan ruang letak kereta yang cekap. Keseluruhannya, projek ini mengatasi isu-isu mendesak berkaitan dengan tempat letak kereta di kawasan perumahan berteres, menawarkan penyelesaian pengesan jarak yang berkesan untuk meningkatkan pengalaman tempat letak kereta, meningkatkan keselamatan, dan mengoptimumkan pengurusan tempat letak kereta di kawasan perumahan.

ACKNOWLEDGEMENTS

In the name of Allah, the Most Gracious, the Most Merciful, I humbly express my deepest gratitude. It is through His grace that I have been granted the opportunity to embark on this endeavor and witness personal growth along the way.

I extend my heartfelt appreciation to my Supervisor, Mrs. Nor Faizah Binti Haminudin, for providing me with the chance and platform to participate in this project. I am profoundly grateful for their valuable insights, guidance, unwavering support, and encouragement, which have made this research paper a reality.

Furthermore, I would like to express my gratitude to Universiti Teknikal Malaysia Melaka (UTeM) for equipping me with the necessary resources and facilitating an environment conducive to enhancing my knowledge and skills. UTeM has played a significant role in enabling me to undertake and complete my final year project.

In all sincerity, I am deeply thankful to Allah for His blessings, my supervisor for her guidance, and UTeM for the support throughout this journey.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

CM	-	Centimeter
GPS	-	Global Positioning System
IoT	-	Internet of Things
LCD	-	Liquid Crystal Display
LED	-	Light-Emitting Diode
PWM	-	Pulse-Width Modulation
RoI	-	Return on Investment
SOC	-	Security Operation Center
V	-	Voltage
Vcc	-	Collector supply voltage

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

Parking has become a critical issue of transport, mobility, and infrastructure development. As the number of vehicles continues to increase, the availability of parking spaces has seen rapid development (Aramane, 2021). This trend highlights the important role that parking plays in the lives of vehicle users, especially car owners who prefer owning a car to other forms of transport. Additionally, the country's unpredictable weather conditions, characterised by erratic rainfall, heat waves and thunderstorms, contribute to consumers' decision to rely on cars as their primary form of transportation. The car provides a sense of security to the user and protection to the user during bad weather conditions and is highly guaranteed to reach the next destination carefully. The focus of this project is to address the challenge of parking in terraced residential areas characterised by limited parking spaces where there are many users who have 2 or more cars per house which causes to take other spaces to make their parking spaces which can lead to traffic congestion. Inadequate parking availability exacerbates the problem, with cars often parked in front of houses and exceeding the designated parking limit, encroaching on traffic lanes. In addition, this project implements a parking distance control system that is tailored specifically for terrace housing areas or flat houses and such. The system will use advanced technology, such as sensors or cameras, to monitor parking spaces in real time on app phone. By implementing these systems, users will have better access while also benefiting from the added convenience and protection offered by cars during adverse weather conditions.

1.2 Problem Statement

The difficulty of accurately judging parking distances in tight or crowded spaces poses a challenge for drivers, exacerbated by limited visibility, blind spots, and the absence of advanced camera technology. This problem is further amplified in outdoor residential parking areas, where insufficient designated spaces lead to vehicles encroaching on roadways, causing traffic congestion. To address these issues, there is a need for an effective parking distance sensor system that provides accurate measurements and real-time feedback, enhancing parking safety and efficiency. Optimising sensor placement to accommodate different vehicle dimensions and considering factors like blind spots will ensure the system caters to all vehicle types. By implementing such a solution, drivers can improve their parking experiences, reduce collisions, and promote better traffic flow in densely populated residential areas.

1.3 Research Objective

The main aim of this project is to design a car parking distance device by using ultrasonic based on Arduino Uno Rev3 ATmega328P. Therefore, the objectives are :

a) To design IoT systems for car parking distance sensor by using Arduino Uno.

b) To determine the potential distance between the types of vehicles and the distance sensor and test the position sensor of the design IoT system for three differents types of car (SEDAN, SUV and MPV).

1.4 Scope of Research

The scope of this project are as follows:

1. Identify the potential IoT systems for car parking distance sensor by using Arduino Uno.
2. Designing a questionnaire that relates to the experiences of car parking and distribute the questionnaire to at least 50 respondents to propose at least three sketching ideas for the car parking distance sensor positions, designs and sensors distances, Arduino and others. The proposed selections will be done by using the Pugh method.
3. Determine the position and the location of sensors that suit for all the type of cars.For each type of cars, the range testing are conducted by using measuring tape. Fabricate the car parking sensor on the selected position and installed all components.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Parking distance is the main concern for the drivers when trying to accurately judge the space between their vehicle and obstacles such as wall when parking, especially in tight or crowded parking space can result in accidental collisions or damage to their vehicle or surrounding objects. Narrow or crowded parking spaces exacerbate this challenge, as the space available to move is limited and makes it more difficult to accurately estimate the distance between their vehicle and surrounding objects. Factors such as blind spots, limited visibility due to obstacles, and the size and shape of vehicles contribute to the complexity of judging parking distances. Larger vehicles may have more blind spots and require additional clearance, while smaller vehicles may be easier to maneuver but still face challenges due to limited visibility.

Outdoor parking in terraced housing, flats and condominiums can pose a challenge as vehicles sometimes exceed the designated parking area and encroach on the roadway. This is due to the limited availability of parking spaces and the high demand for parking spaces in densely populated residential areas. As a result, cars are often parked in a way that goes beyond the designated parking zone and extends onto the road. This problem is particularly prevalent in parking areas that are not clearly marked or the available parking spaces are insufficient to accommodate the number of vehicles. In such situations, drivers may use parking spaces along the road or in areas that are not designated for parking, causing vehicles to pass the intended parking area. Intersections with parked vehicles can lead to a number of issues such as narrowing the space available for other vehicles to pass through, potentially obstructing traffic flow, and creating difficulties for pedestrians and other road users.

2.2 Types of parking space in a housing area

Insufficient parking space availability in housing areas, including terrace houses, flat houses, and condominiums, contributes to conflicts and challenges, particularly in terms of illegal parking. Inadequate parking provisions can lead to disputes among residents as compete for limited spots, negatively impacting neighborly relationships. This problem extends to various housing types, including flat houses and condominiums, where the demand for parking often exceeds the available spaces. Insufficient parking options also discourage the use of sustainable transportation alternatives, leading to increased reliance on private vehicles and exacerbating traffic congestion and environmental pollution. Addressing these issues requires efficient parking management strategies, such as implementing parking regulations, promoting alternative modes of transportation, and enhancing enforcement against illegal parking practices. By doing so, housing areas can mitigate conflicts, improve transportation efficiency, and create a more sustainable environment. Understanding the factors and trends behind illegal parking is crucial for effective parking planning and alleviating congestion. Research conducted in Beijing, China revealed that approximately 15% of individuals engage in illegal parking on sidewalks, while 18% park in setbacks (Bram van, Li, & Liu, 2015). Insufficient parking supply leads to illegal double parking, which in turn impacts traffic speeds, enforcement efforts, and air pollution levels (Barone, Giuffrè, et.al, 2013). By studying these causes and patterns, authorities can develop improved parking strategies and regulations, ultimately fostering smoother traffic flow and reducing congestion.

2.2.1 Terrace house parking space

Terrace houses are known for their dense and compact design, making one of the most densely built forms of landed property development. These houses typically have varying lot sizes, such as 20 feet by 65 feet or 22 feet by 70 feet. The layout of a typical terrace house can be categorised into indoor and outdoor spaces, with the indoor spaces including four bedrooms, three bathrooms, a kitchen, and a dining and living area (A. Abu Bakar et al., 2016). However, the availability of parking space is a critical aspect of terrace houses as can be seen in Figure 2.1. As the interface between the outdoor and indoor areas, the entrance plays a crucial role in providing parking facilities for residents. It serves as the transition point between the public and private spaces, accommodating vehicles and ensuring convenient access to the house. The density of a residential area is a significant factor when considering the availability of parking facilities (Panayappan, et. al, 2007). In terms of parking lot density, it refers to the average distance a driver may need to travel within a residential district. In this context, the crucial role of the entrance becomes evident as it serves as a primary location for providing parking facilities to residents. The proximity and accessibility of parking spaces at the entrance can greatly impact the convenience and ease of parking for residents, especially in densely populated terrace house areas.

Figure 2.1 Parking space of terrace house (H Khan et. al, 2015)

2.2.2 Flat house parking space

In Seoul, the shortage of parking spaces in low-rise residential areas, specifically flat houses, can be attributed to various factors, as mentioned in the previous response. When these areas were initially developed in the 1970s, primarily consisted of single-family houses. However, as time passed, some of these houses were converted into multi-family housing units, resulting in an increased density of residents. With the rise of vehicle ownership and usage from the 1980s onwards, the demand for parking spaces surged dramatically. Unfortunately, the existing infrastructure and design of these low-rise residential areas did not anticipate such a significant increase in vehicles as in Figure 2.2. Consequently, the available parking space within the premises of these flat houses became insufficient to accommodate the growing number of cars (Saehoon Kim, 2018).

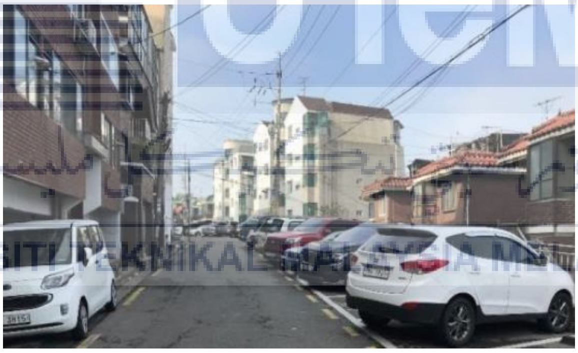


Figure 2.2 Parking space of flat house (Seungwoo Seo, 2018)

In 2004, the Green Parking Project was implemented in Seoul and aimed to address the parking space shortage in Seoul's apartment complexes by removing the walls of individual units. Through the participation of 24,747 apartment units by the end of 2014, a total of 48,867 parking spaces were secured. This approach proved to be relatively cost-effective as it utilised existing private space within the apartments. However, the project has faced challenges in encouraging landowners to participate, leading to a decline in results over the years. To revitalise and enhance the effectiveness of the Green Parking program for apartment complexes in Seoul, suggestions have been made to diversify the plans and introduce self-management systems with adequate support (Shin Jung-in et al., 2011).

2.2.3 Condominium parking space

Condominium parking spaces are specific areas within a condominium complex that are designated for residents to park their vehicles. The availability and allocation of these parking spaces can vary from one condominium to another based on factors such as the size of the complex, the number of units, and local regulations. These parking spaces can take different forms, including open parking lots, covered parking structures, or underground garages. Open parking lots are commonly found in smaller condominiums, while larger complexes often provide covered or underground parking facilities to offer additional protection for vehicles. In some cases, each unit is assigned a designated parking spot, ensuring that residents have their own dedicated space. In other cases, parking spaces may be shared among multiple residents on a first-come, first-served basis, which can lead to challenges in finding an available spot during peak times. Additionally, some condominiums may offer separate parking options for visitors or provide parking permits for residents' guests to ensure convenience and appropriate use of the parking facilities.

Based on research in Taiwan, the most vexing problem concerning condominium parking spaces in Taiwan is the frequent violation of parking regulations. According to a study, 36% of the surveyed management committees found that highly bothersome when residents disregard the established parking rules. In an attempt to address this issue, the Model Condominium Bylaws (2006) include specific provisions regarding parking regulations. However, the study revealed that only 57% of the management committees surveyed had implemented their own parking regulations through condominium bylaws. This suggests that the establishment of comprehensive parking regulations within condominiums in Taiwan remains incomplete or insufficiently enforced (Hun Ren Hsieh, 2009).

Figure 2.3 Condominium parking space (Vista Land, 2021)

Therefore, this project aims to alleviate the problem of parking difficulties in small and crowded areas by leveraging IoT technology. Figure 2.3 show that the increasing number of vehicles and limited parking spaces in terraced residential areas, the implementation of a parking distance control system using advanced IoT sensors or cameras becomes crucial. By providing real-time feedback and accurate measurements, the system assists drivers in accurately judging parking distances in tight spaces. IoT technology enables optimal sensor placement, accommodating different vehicle dimensions and considering blind spots, resulting in an enhanced parking experience, reduced collisions, and improved traffic flow. Via comprehensive testing and analysis, the safety and effectiveness of the car parking system are ensured, making it an effective solution to optimise parking in housing areas.

2.3 Projects focused on sensing applications of IoT

The Internet of Things (IoT) refers to a network of interconnected devices that collect and process data from the physical world. These devices work together to create an ecosystem where applications and services are powered by the data collected (CEBR, 2016; European Commission, 2016). One important application of IoT is in sensing the environment. IoT sensors can be used to measure various environmental factors such as air quality and temperature. By gathering real-time data, these sensors provided better understanding and managing the environment, leading to improved environmental monitoring and decision-making. In agriculture, IoT sensors play a vital role in optimising farming practices. By allowing farmers to monitor soil moisture levels and temperature, providing valuable information for irrigation and fertilisation. By utilising IoT technology, farmers can make more informed decisions and enhance crop health and productivity. Smart buildings and infrastructure also benefit from IoT sensing applications. Sensors placed throughout buildings can monitor energy usage, occupancy levels, air quality, and security. This data helps to optimise energy efficiency, create comfortable environments for occupants, and detect maintenance issues early on. IoT sensing is particularly useful in transportation as well. Sensors in roadways, parking lots, and vehicles collect data on traffic flow and parking availability. This information can be utilised to optimise traffic signals, provide accurate traffic predictions, and guide drivers to available parking spaces, resulting in improved transportation efficiency. The following sub-topic are some of the sensing applications of the IoT.

2.3.1 Global Positions System (GPS) Acquisition

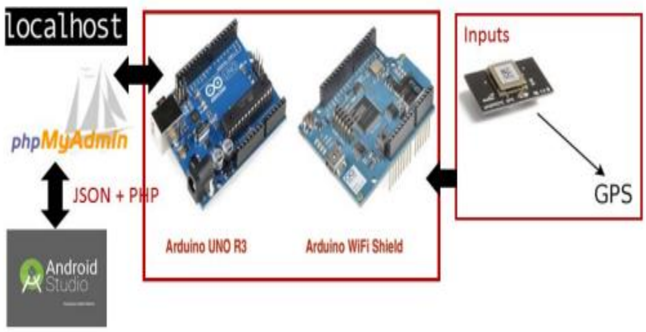


Figure 2.4 GPS Acquisition for IoT system (Salha, 2017)

Figure 2.5 Mobile application to deal with GPS data (Salha, 2017)

In a previous study, the Arduino was equipped with a GPS shield, as shown in Figure 2.4, to obtain GPS data. The GPS data includes the longitude and latitude coordinates of the device's location. This data was then stored in a web-based database for further processing. The collected GPS data facilitated the development of various applications. For example, the GPS data was utilised to create a golf cart finder and navigator for public areas. This application assists users in locating golf cars and navigating through specific areas. Additionally, a lost devices and belongings tracker was developed, as depicted in Figure 2.5. This tracker utilises GPS data to locate lost items such as keys, laptops, bags, bicycles, and more. The commercialisation of this tracking model offers significant utility for tracking and monitoring valuable possessions (Salha M. Alzahrani, 2017). It provides an effective solution for keeping track of important items and minimising the risk of losing.

2.3.2 Weather sensing

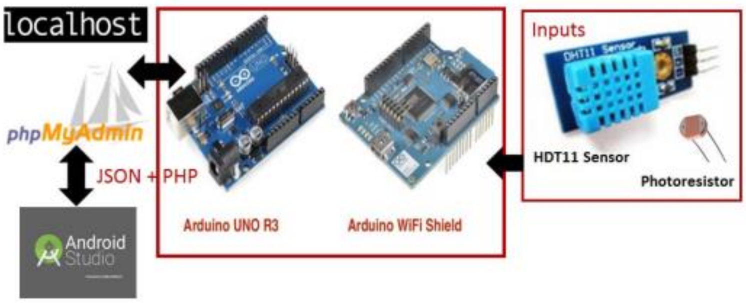


Figure 2.6 Temperature, humidity and light sensing for IoT system (Salha, 2017)

Figure 2.7 Mobile application to deal with weather data and display medical notifications (Salha, 2017)

Weather sensing is another significant application area for IoT systems. This system gathers data from the surrounding atmosphere, including temperature, humidity, pressure, light, and other parameters. Figure 2.6 illustrates a general IoT system designed to collect and transmit weather data to a web server, like the GPS acquisition process described earlier. To measure temperature and humidity, an HDT-11 sensor was employed in the IoT system, providing accurate readings of the environmental conditions. Additionally, a simple photo resistor was used to sense the light intensity in the vicinity. This combination of sensors enables comprehensive weather data collection. The availability of weather data opens numerous possibilities for home automation and control. One developed idea is a smart notify that can be utilised anywhere. Figure 2.7 showcases a mobile application that receives data from the sensors and displays notifications, including general warnings, medical advice, tips, and information about the nearest medical center. It's important to note that the exemplified application in Figure 2.7 was specifically tailored for the Arab community, hence the notifications being reported in Arabic. In the future, this IoT system has the potential to be integrated with screens located in public areas or airports. By displaying real-time weather information, it can assist individuals in making decisions and taking appropriate actions based on the current weather conditions (Salha M. Alzahrani, 2017).

2.3.3 Liquid sensing

Figure 2.8 liquids sensing for IoT system (Alzahrani, 2017)

Figure 2.9 Mobile application to deal with water-level data (Alzahrani, 2017)

A mobile application was developed in conjunction with a smart water bottle to aid individuals in meeting their daily water intake requirements. A study was conducted by Salha and Alzahrani (2017) utilised a smart water bottle, as depicted in Figure 2.8, which was connected to a mobile application, as shown in Figure 2.9. The mobile application calculates the user's daily water goal based on predefined personal parameters, ensuring personalised recommendations. It provides a visual representation of the recommended water intake per day. Furthermore, the application sends timely reminder notifications to the user's mobile device, prompting people to drink water regularly. Simultaneously, the smart water bottle illuminates in synchronisation with the reminder notification, serving as an additional visual cue. This straightforward IoT system offers valuable assistance in helping individuals fulfill their daily water consumption needs, ultimately benefiting their overall health and well-being. By providing personalised water intake goals and sending reminders, it encourages individuals to stay hydrated throughout the day, leading to positive health outcomes (Salha M. Alzahrani, 2017).

2.3.4 Smoke sensing

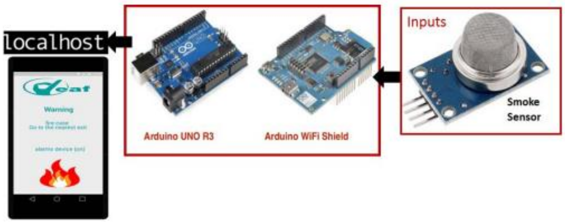


Figure 2.10 Mobile application for smoke detection and alert for deaf (Alzahrani, 2017)

Individuals who are deaf face challenges in receiving auditory warnings during emergency situations. To address this issue, an IoT system was developed, as illustrated in Figure 2.10, which incorporates smoke sensing technology and a smartphone application. The primary objective of this IoT system is to provide timely warnings to deaf individuals during emergencies, particularly in cases of fires. The system utilises smoke sensors that are connected to a smart device application. In the event of an emergency, the application generates light alerts on the IoT device, effectively capturing the attention of the user. Simultaneously, a mobile notification is sent to the user's smartphone, providing additional information such as a map displaying the location of the nearest emergency exit. By combining IoT technology, smoke sensing capabilities, and a smartphone application, this system enhances the safety of deaf individuals during emergency situations. The light alerts serve as visual cues, ensuring that individuals receive critical warnings even in the absence of auditory cues. The mobile notifications and location information further aid individuals in swiftly and safely navigating towards the nearest emergency exit (Salha M. Alzahrani, 2017).

2.3.5 Soil sensing

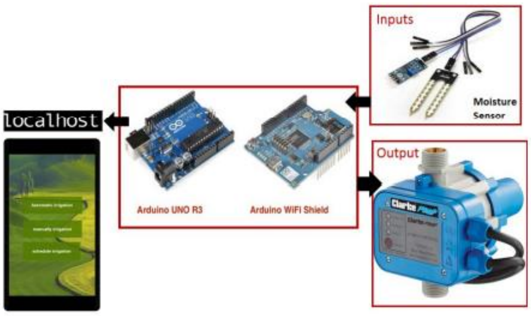


Figure 2.11 Mobile application to deal with moisture data (Salha, 2017)

Figure 2.12 Smart home plants irrigation (Salha, 2017)

The application of IoT-based soil sensing in smart automatic irrigation systems offers an advanced solution. By incorporating a moisture sensor embedded in the soil, this system effectively addresses concerns such as forgetting to water plants or excessive water usage. The moisture sensor, connected to an IoT device, enables the system to detect the level of humidity in the soil and automatically provide an appropriate amount of water. This project significantly simplifies the irrigation process, irrespective of the distance, through its integration with a smartphone application, as depicted in Figure 2.11. The smart irrigation system ensures convenience and efficiency by allowing users to control and monitor the irrigation process remotely. The output of the smart irrigation system, illustrated in Figure 2.12, demonstrates its positive impact on water distribution control, soil saturation, and water conservation. By leveraging IoT technology and soil sensing capabilities, this smart automatic irrigation system optimises water usage, prevents overwatering or underwatering, and promotes the healthy growth of plants. The integration with a smartphone application enhances user accessibility and control, making the irrigation process more efficient and manageable (Alzahrani, 2017).

2.3.6 Distance sensing (ultrasonic)

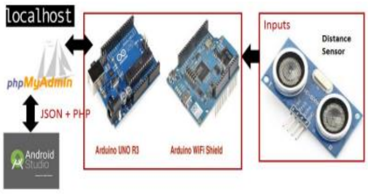


Figure 2.13 Using ultrasonic/ ultrasound sensor for distance measuring (Salha, 2017)

Figure 2.14 Built-in ultrasonic sensors in blood bank refrigerators (Salha, 2017)

Utilising ultrasonic or ultrasound sensors for distance measurement is a valuable application in combination with an IoT system. Distance data obtained from these sensors holds significant potential for various applications. One example is our proposed solution that leverages IoT technologies to inform blood bank authorities and donors about the need for blood donations, as depicted in Figure 2.13 and Figure 2.14. In this solution, ultrasonic sensors are attached to the blood bank refrigerators. This integration helps to monitor the demand for specific blood types through a smartphone application. Additionally, each refrigerator is equipped with an LCD screen that displays the current quantity of different blood types using built-in sensors. This IoT-based system provides comprehensive information to both the medical staff and donors, facilitating efficient coordination and timely response to blood supply requirements. By employing ultrasonic sensors and IoT technology, this solution enhances communication and transparency within the blood bank ecosystem. The real-time data on blood type availability and demand enable proactive decision-making and streamlined blood donation processes. Ultimately, the IoT system's capabilities empower both medical staff and donors to contribute effectively to the vital cause of maintaining an adequate blood supply (Salha, 2017).

2.4 Applications of distance sensor based on car parking systems

Applications of distance sensor based on car parking systems include the development of smart parking systems using IoT technology. These systems utilise distance sensors to detect the presence of vehicles and monitor parking space occupancy in real-time. One such project focuses on the use of NodeMCU 1.0 development board to create a smart parking system that provides guidance to drivers in finding available parking spots. Another project utilises a combination of a 360-degree camera and a Haar-cascade classifier to implement a smart parking guidance system. The distance sensor, along with the camera, helps detect and track vehicles, enabling real-time parking guidance. These applications showcase the practical implementation of distance sensors in car parking systems, aiming to enhance parking efficiency, optimise space utilisation, and improve the overall parking experience for drivers.

2.4.1 Car Parking Distance Controller Using Ultrasonic Sensors Based on Arduino Uno

In the study conducted by Paidi and Fleyeh (2017), the design of a car parking sensor addresses issues such as the absence of parking attendants during specific hours and a lack of available attendants. To tackle these challenges, researchers developed an Arduino Uno-based car parking distance controller using ultrasonic sensors. The distance information is processed and combined with the MP3 Shield, triggering specific voice commands at predetermined distances. The prototype method was employed, involving designing, system programming, implementation, and testing to create a functional car parking sensor prototype. The testing concluded that the sensor greatly facilitates parking for drivers by providing accurate distance information and voice guidance. Hence, there is a need for a parking system that simplifies the parking process for car drivers. In this regard, the utilisation of an Arduino Uno device with an ultrasonic sensor (Paidi and Fleyeh, 2017).

Another study used Arduino UNO ATMega 328P which is a microcontroller chip manufactured by Atmel. It is an 8-bit chip based on AVR-RISC architecture and features 32 KB of ISP flash memory with read/write capabilities, 1 KB EEPROM, and 2 KB SRAM. The chip known as ATmega328, is derived from its 32 KB flash memory capacity (Irawan and Yuda, 2019). The study focused on highlighting this feature and its significance. A comprehensive examination of the Arduino UNO module was carried out, with a focus on its user-friendly design (Atmadja and Yosafat, 2014). By highlighting its compatibility with both USB cables and DC-DC adapters, enabling easy connection to a PC (Wahyuni, 2015). In another study, the diverse components of the Arduino UNO module, including 14 input/output pins was investigated by Abdel (2016). Notably, 6 of these pins can be utilised as PWM outputs, while an additional 6 functioning as an analog input. The module is further equipped with a 16 MHz crystal oscillator, USB connection, power jack, ICSP head, and reset button.

The ultrasonic sensor was used which is a device that converts physical quantities, specifically sound, into electrical quantities and vice versa (Lee, 2019). It operates on the principle of sound wave reflection, enabling it to determine the presence (distance) of an object with a specific frequency. Termed "ultrasonic" due to its utilisation of ultrasonic waves, which are sound waves with an extremely high frequency of 20,000 Hz, this sensor operates beyond the range of human hearing. Ultrasonic waves can propagate through solids, liquids, and gases, and their reflectivity on solid surfaces is comparable to their reflectivity on liquid surfaces. However, textiles and foam tend to absorb ultrasonic sound waves.

This project incorporates the hardware design featuring an Arduino microcontroller circuit with an ultrasonic sensor, as illustrated in Figure 2.15. Additionally, Figure 2.16 depicts the Arduino microcontroller with a speaker, highlighting the complete hardware setup and the comprehensive range of hardware configurations are shown in Figure 2.17.

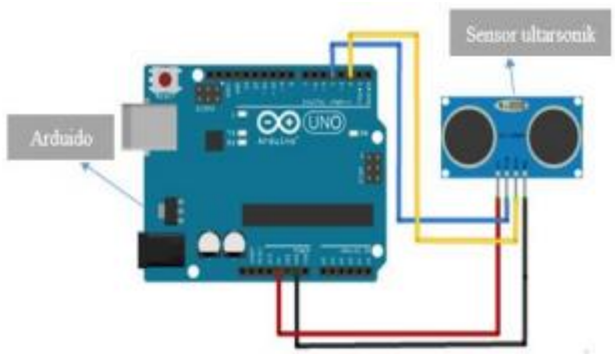


Figure 2.15 Arduino Microcontroller Circuit with Ultrasonic Sensor (Anita, 2021)

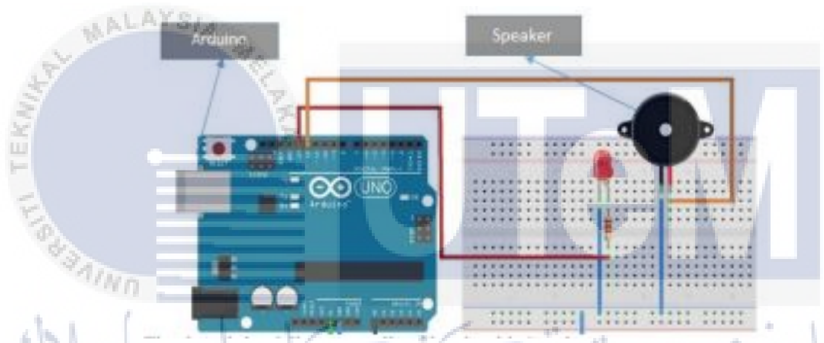


Figure 2.16 Arduino Microcontroller Circuit with Speaker (Anita, 2021)

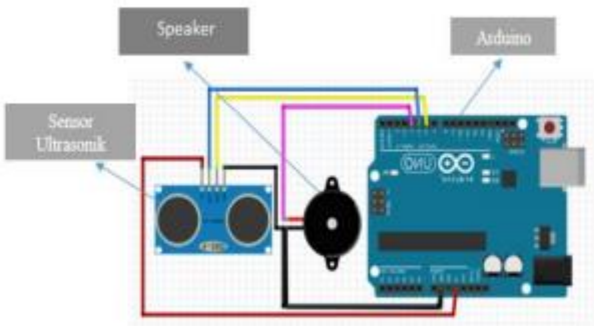


Figure 2.17 The overall series of hardware configurations (Anita, 2021)

The car parking distance controller has been successfully implemented, as demonstrated in Figure 2.18. This implementation involves the integration of various components and technologies to enable efficient and automated parking (Rahmalisa, 2021). By utilising sensors, such as ultrasonic sensors, the system can accurately measure the distance between the car and obstacles, facilitating safe parking maneuvers. The controller, powered by Arduino or a similar microcontroller, processes the sensor data and provides real-time feedback to the driver. This feedback can be in the form of visual indicators, audio signals, or both, allowing the driver to navigate and position the vehicle within the designated parking area with precision. The implementation of a car parking distance controller greatly simplifies the parking process and enhances safety by minimising the risk of collisions or other parking-related incidents.

Figure 2.18 Implementation of a Car Parking Distance Controller (Anita, 2021)

The prototype underwent rigorous testing using the BlackBox Testing method to assess the functionality of the system. This testing approach primarily focuses on evaluating the device's compliance with the specified functional requirements while identifying any potential errors or malfunctions. The tests aimed to uncover any issues related to data structure, performance, or general functionality. The test results, including any errors or discrepancies encountered, have been documented and are presented in Table 2.1. These testing procedures provide valuable insights into the system's performance and help ensure its reliability and effectiveness.

Table 2.1 Testing the system using black box testing (Jenli Susilo, 2021)

No	Procedure	Expected results	Validation
1	If the car approaches the sensor with a distance of 5 cm	Then the sound output emits a reverse sound	Valid
2	If the car approaches the sensor exactly 2 cm	Then the sound output emits a stop sound	Valid
3	If the car approaches the sensor less than 2 cm away	Then the sound output emits a forward sound	Valid

The ultrasonic sensor was subjected to testing at 15 cm distance using a toy car on the prototype. During the test, when the ultrasonic sensor detected the presence of a car within a 15 cm range, it triggered the emission of a reverse sound. The test outcomes, along with visual representations, are displayed in Figure 2.19 (a). These test results demonstrate the sensor's capability to accurately detect objects and successfully activate the corresponding feedback mechanism.

Figure 2.19 Ultrasonic sensor test at different distances. (a) 15 cm (b) 3 cm (c) 2 cm (Jenli,2021)

The ultrasonic sensor underwent testing at a distance of 3 cm and 2 cm using a toy car on the prototype. During the test, when the ultrasonic sensor detected the presence of a car within a 2 cm range and 3 cm, it emitted a sound output as intended. The test results, indicating the successful functionality of the sensor at this distance, are illustrated in Figure 2.19 (b) and (c).

For the analysis of this project, design, and implementation conducted, several conclusions can be derived. Firstly, the parking distance control system designed in this study successfully enables car parking through voice commands emitted by the speaker, enhancing the ease of parking. Secondly, the testing of the parking sensor revealed a disparity in sensor distance between prototype testing and real-world testing (Jenli Susilo and Anita Febriani, 2021).

2.4.2 The development of a smart parking system based on NodeMCU 1.0 using the Internet of Things

In this project, with the increasing number of vehicles on the road, the demand for parking spaces has risen significantly. This poses challenges in efficiently locating vacant parking spots in large parking areas and addressing the issue of illegal parking. To tackle this problem, we propose a smart parking system that utilises the Internet of Things (IoT) as a potential solution. This system aims to streamline parking lot management and assist drivers in easily identifying available parking spaces. The system operates by employing Infrared sensors connected to the ESP12-E (Node MCU) module, which is programmed using the Arduino IDE. Users can access real-time parking space information through a smartphone application. To ensure security and user convenience, registered users are required to log in with a unique code. The effectiveness of this system aligns with the objectives of our research, making it a viable solution for addressing parking-related challenges. Based on the available data, the objective of this study was to design a prototype of a parking monitoring system and develop a parking system based on an existing reference system (Nugrahadi, 2016). The microcontroller chosen for this system was the Node MCU, which was complemented by a Wi-Fi system. The smart parking system was implemented by integrating a mobile application that connects to the cloud. This system effectively provided real-time information to users regarding the availability of parking spaces.

The hardware architecture of this project consisted of three main components. The first component is the NodeMCU (ESP-12), as shown in Figure 2.20. NodeMCU is an open-source software and hardware development environment built around the ESP8266 System-on-a-Chip (SoC). The ESP8266, developed by Espressif Systems, incorporates essential elements of a modern computer, including CPU, RAM, networking (Wi-Fi), and an operating system with an SDK. In this project, NodeMCU was utilised as a microcontroller module with Wi-Fi capabilities, programmed using the Arduino IDE. The specific version of NodeMCU employed was NodeMCU v.1, also known as ESP-12E. Due to limited I/O, two NodeMCU modules were used in this project to achieve the desired project goals.

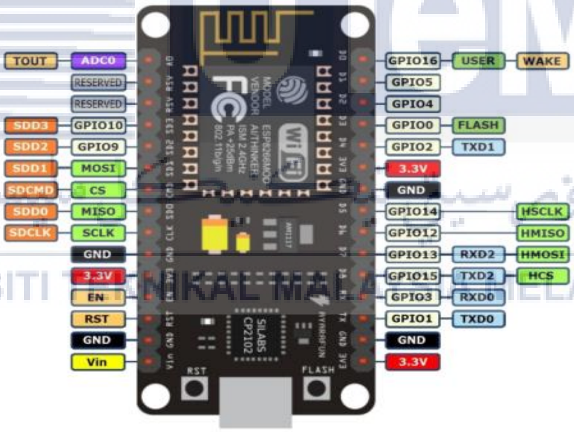


Figure 2.20 NodeMCU 1.0 (Budi, 2017)

In addition, this project employed the infrared (IR) obstacle sensor to detect the availability of parking spaces. When the sensor detects an obstacle blocking its path, it activates an LED to indicate that the parking space is occupied.

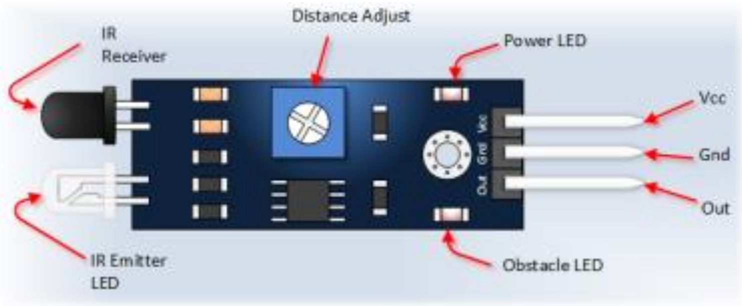


Figure 2.21 Infrared obstacle sensor (Budi, 2017)

Two infrared sensors, as shown in Figure 2.21, are utilised in this prototype for car detection. The logical AND operation is applied to these sensors to determine the availability of parking spaces accurately. This placement strategy with AND logic helps prevent potential misuse of the sensors in the parking spaces.

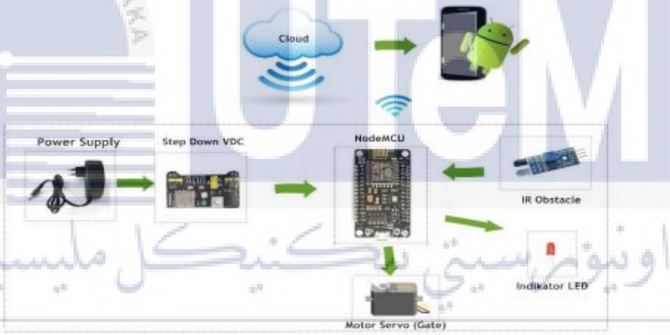


Figure 2.22 Architecture of smart parking system (Budi, 2017)

Besides previous project, Anjari (2018) conducted a project known as the smart parking system architecture, depicted in Figure 2.22, consisting of several components. The IR obstacle sensors are connected to the NodeMCU microcontroller. The NodeMCU chip functions as a self-contained SOC (system on a chip) with an integrated TCP/IP protocol stack, enabling connectivity to a Wi-Fi network. It serves as an intermediary between the sensors and the cloud. The cloud, acting as a database, stores all the parking areas and user records. The NodeMCU transmits the data to Firebase through the Arduino IDE. The mobile application serves as a user interface, allowing users to interact with the system on the Firebase Cloud platform.

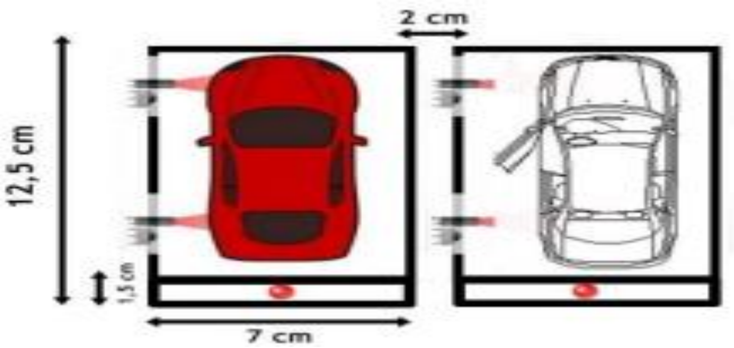


Figure 2.23 Architecture of prototype for parking space dimension (Budi, 2017)

A parking space as illustrated in Figure 2.23, is a designated location for parking vehicles. It is typically demarcated by road surface markings. The size of the space can vary depending on the parking method used, such as parallel parking, perpendicular parking, or angled parking. The dimensions of the parking space, as outlined in the Parking Planning and Operation Manual, are influenced by factors like door openings and vehicle type. The width of the door opening plays a role in ensuring passenger comfort during vehicle entry and exit. In the prototype of the smart parking system, perpendicular parking was chosen as it is commonly employed in car parking lots.

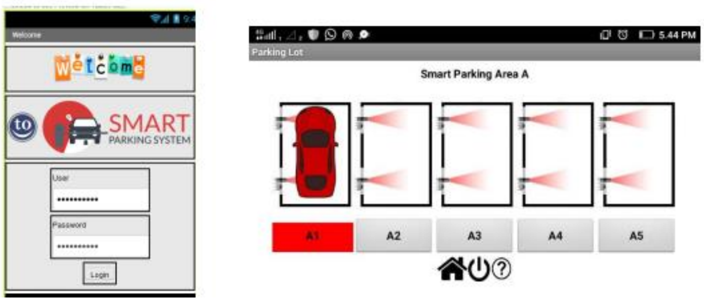


Figure 2.24 The register user interface and display contents of menu (Budi, 2017)

In the depicted prototype shown in Figure 2.24, the parking spaces A1-A5 are monitored for occupancy using IR Obstacle sensors. When a car obstructs the sensor, indicating that the space is occupied, an LED will be activated to provide visual feedback. This smart parking system utilises a Wi-Fi connection to establish communication between the ESP8266 microcontroller and the Cloud Firebase platform. The sensor data is transmitted and synchronised with the application, allowing real-time display of the parking lot status in the mobile application (Anjari, 2018).

Figure 2.25 Real-time database on Firebase platform (Budi, 2017)

The application is synchronised with a real-time database on Firebase, which functions as a cloud platform as shown in Figure 2.25. This setup allows testing the concept of the Internet of Things (IoT), enabling data accessibility from anywhere and at any time through an internet connection. However, the project encountered several constraints that led to a mismatch between the intended goals and the achieved results. These constraints include in the initial project planning, the ESP32 microcontroller was intended to be used due to its higher number of I/O pins compared to NodeMCU. However, due to limited knowledge and the complexity of IoT platforms supporting ESP32, a human error occurred, resulting in the system not aligning with the original plan. The NodeMCU microcontroller has only 10 digital I/O pins, which deviated from the planned system that involved two sensors per parking space. Consequently, the system design had to be adjusted to accommodate the limitations of the microcontroller. These constraints highlight the challenges faced during the project, including limitations in technical knowledge and the need to adapt the system design to the available resources (Budi, 2017).

By the end of this project, the smart parking system developed successfully achieved its intended purpose, which includes providing real-time information about the parking lot's availability through an internet-connected application. This application enables parking users to conveniently search for vacant parking spaces anytime and anywhere, utilising IoT technology as its foundation (Anjari and Budi, 2017).

2.4.3 The Smart Parking Guidance System Using 360 Camera and Haar-Cascade Classifier on IoT System

This research proposed a smart parking guidance system that relied on a modified 360° camera, Raspberry Pi camera module, 360° lens, and Haar-Cascade classifier. The image and video processing were performed using OpenCV and a Python program to detect available parking spaces. Cloud Firebase was utilised to update data, allowing users to access parking space availability via an Android mobile phone, specifically within a closed parking space. This innovative system replaced multiple sensors and cameras found in traditional smart parking systems. Performance analysis showed a remarkable 99.74% accuracy in detecting parking availability, surpassing conventional systems in terms of reliability and cost-effectiveness for parking space guidance.

Figure 2.26 IoT architecture concept (Alsafery and Reiff-marganiec, 2018)

This research comprised several components, including camera technology, a cloud system, and an end-user application. The focus was on developing 360° camera technology as a sensor and integrating it with Raspberry Pi 3 Model B, image processing, a mobile application, and a cloud system, as outlined in the objectives. To simulate the system, a 5-megapixel camera was utilised and modified to provide a 360° view by attaching additional lenses. Any captured image or video underwent processing using a Python program and OpenCV libraries before being transmitted to the Firebase cloud platform and made accessible through the end-user application. This architecture aligns with the concept of the Internet of Things (IoT), as depicted in Figure 2.26. The scope of the IoT market has been expanding rapidly, as indicated by a report from Gartner in 2011 (Sardeshmukh and Ambawade, 2018). The IoT offers great flexibility and efficiency, with its performance largely determined by the algorithms used to correlate and analyse the collected data, particularly the quality of image data in IoT applications (Benita & Chalissery, 2017).

Besides that, cloud computing and IoT are closely intertwined, as involve the sharing and processing of data. Cloud computing, as described by Hall (2010), is a free and shareable architecture that can be improved by communities. It plays a crucial role in the IoT concept. The cloud provides vast storage capacity and can effectively handle real-time activities and notifications for numerous IoT devices (Princy & Nigel, 2015). In this research, cloud computing was employed for the smart parking guidance system. Another solution proposed by Maenhaut et al. (2017) introduced a Raspberry Pi testbed called RPiaaS, which aimed to validate cloud resource management strategies. The testbed provided a user-friendly environment for evaluating novel strategies and transitioning from simulations to larger experimental evaluations. Additionally, Durnyak and Havrysh (2018) utilised computer vision techniques, including Fourier transform, to process input signals from images or videos.

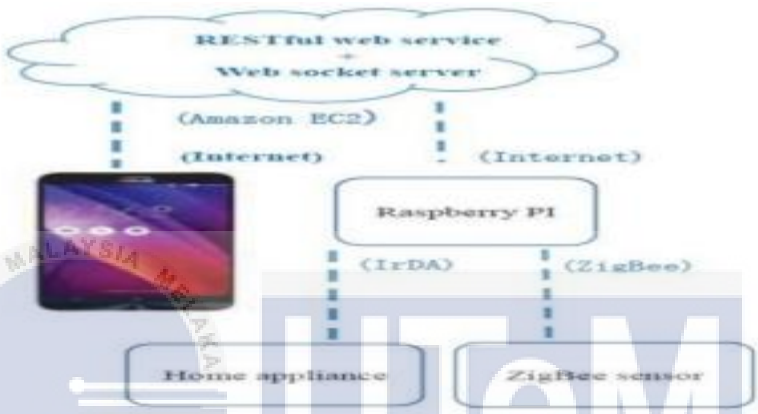


Figure 2.27 Raspberry Pi implemented in Smart Home system (Korea, 2016)

Raspberry Pi is a popular choice among inventors, engineers, and hobbyists for IoT research projects that require cloud connectivity. It’s affordability and open-source nature make it a preferred processor for various applications. The abundance of information, tutorials, and specifications available online further contribute to its appeal for future improvements. Numerous research papers have utilised Raspberry Pi as either the main or supporting component. For example, Collage et al. (2017) and Ojrulwkp et al. (2017) implemented Raspberry Pi in a smart parking system, achieving their objective of cost minimisation. In another study by Korea (2016), Raspberry Pi served as a supporting component in a Smart Home system as Figure 2.27, effectively communicating with the cloud server. Raspberry Pi has also been employed to power cloud servers in applications such as smart sensor networks for buildings (Aktaruzzaman, et. al, 2017), water quality monitoring systems (Lobachev, 2016), and low-cost video transcoding (Barais, Bourcier, & Dion, 2016). In contrast, the Arduino Uno, used in a smart car parking management system, lacks its own operating system, and requires additional shields for features like Ethernet connection, video streaming, and Bluetooth connection, leading to increased costs. Due to its features and affordability, Raspberry Pi was selected as the preferred option for such project. In a study done by Salma et. al (2019) was to investigate and analyse various techniques employed in smart parking guidance systems, with a specific focus on developing a cost-effective process for identifying available parking slots and facilitating reservations. The study utilised simulation as the primary method to accomplish the research objectives. Consequently, the research provides a concise description of the hardware and software systems utilised in the development of the prototype (Salma, et. al, 2019).

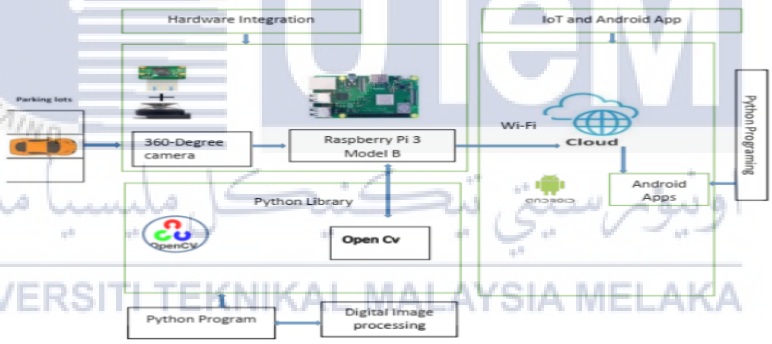


Figure 2.28 Complete process of the smart parking guidance system method (Anjari, 2017)

The system consisted of three stages which are image processing, cloud development, and mobile application development. The primary focus was on image processing, as the camera served as the main sensor for detecting parking space availability. Raspberry Pi was chosen as the prototyping device due to its affordability and capabilities. The entire process is depicted in Figure 2.28. The Raspberry Pi camera was connected to the Kogeto Dots 360° lens converter and attached to the Raspberry Pi port. Image processing was performed using Python and OpenCV. For simulation purposes, the camera was positioned in the center of the parking space to capture a wider view. The video captured by the camera was processed using edge detection to detect changes in pixels, indicating the presence of a car. The data was then sent to the cloud (Firebase) and made available to users through a mobile application. Users could reserve an available parking space, and the Google Maps API was used to estimate the time it would take to arrive at the parking spot.

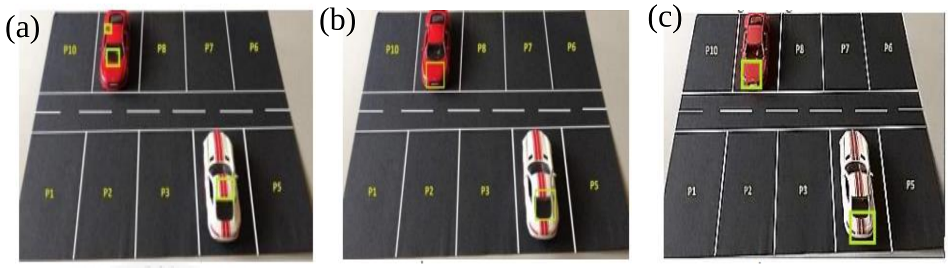


Figure 2.29 Prototype simulation (a) First trial, (b) Second trial, (c) Third trial (Salma, 2019)

Figure 2.29 displays the results of image and video processing. Initially, the video was recorded and then processed using the OpenCV software. However, in later stages, the video was live-streamed, allowing for real-time processing. During the first trial of video processing, the detection accuracy was not very precise. The program identified some unnecessary parts of the car and background. The size, color, and thickness of the rectangles were adjusted within the Python program. The region of interest (RoI) was defined using a trained Haar-cascade .xml file, which was called within the program. This trial utilised 300 negative background images and 150 positive subject images. In the second trial as in Figure 2.29(b), the video was recorded in high lighting conditions. The program accurately recognised the cars with 100% detection accuracy. However, only one detection per car was needed. For this stage, the Haar-cascade utilised 1500 negative background images and 750 positive subject images. The accuracy of the car detection was observed to depend on the number of images used for training the feature detection. In the third trial as Figure 2.29(c), the video was recorded in occluded weather with low light conditions. Despite these challenges, the system was able to accurately detect cars up to 99.49% of the time. It's important to note that these results represent the performance of the prototype system at this stage (Morshidi, 2019).

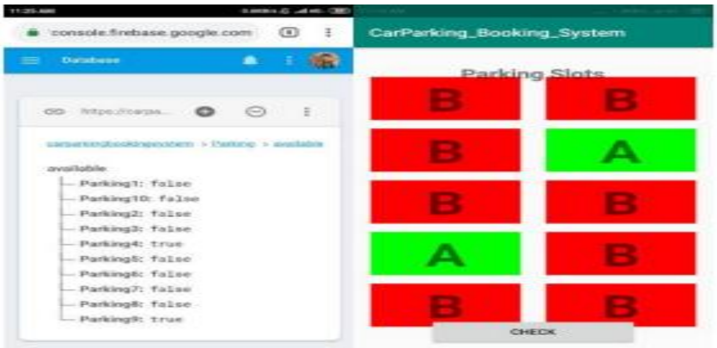


Figure 2.30 Check parking slot and Parking availability information in Firebase (Salma, 2019)

In order to access the parking information as in Figure 2.30, users are required to install the "Car Park Booking System" app on their smartphones. Once installed, they can access the app and engage in various activities. The data retrieved by the app is visually represented by different colors. Available parking lots are displayed in red, while the number of booked lots is indicated in green, as shown in Figure 2.30 (Salma, 2019). Based on the simulation results, it can be concluded that the developed prototype of the 360° smart parking guidance system is reliable and exhibits better detection accuracy compared to other systems. The User Acceptance Test (UAT) demonstrated that this IoT-based monitoring system operates in real time without requiring physical human involvement. Backend users can monitor and interact with the system's real-time data through their devices, such as smartphones, via the Internet. The prototype was tested under two weather conditions: sunny and occluded. In sunny conditions, the accuracy reached 100%, while in occluded weather, there was a slight decrease in accuracy to 99.49%. The accuracy of the system depends on the number of images used for training in the Haar-cascade method, necessitating many positive and negative images with a training ratio of 2:1 (Rashidah Funke Olanrewaju, 2019). Therefore, this research concludes that the implementation of the 360° smart parking guidance system will contribute to creating a smarter environment. Urban drivers will benefit from a more productive situation as can remotely monitor and reserve specific parking lots, saving both time and money by avoiding unnecessary fuel consumption.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The car parking experience has been plagued by various challenges, particularly for vehicles lacking advanced sensing technology. The absence of such technology in car parking systems poses significant safety concerns. Therefore, it is crucial to address this issue by implementing reliable distance sensors specifically designed for car parking. This project aims to ensure the safety and effectiveness of the car parking system by conducting comprehensive testing of the car parking distance sensor. The study focuses on accurately measuring the distance between obstacles and cars, with a particular emphasis on vehicles without advanced technology. The focus of this project is to alleviate congestion and parking problems in residential areas by implementing an effective distance sensing solution. By conducting thorough testing and analysis of the car parking distance sensor, this project endeavors to provide a robust and dependable solution. The aim is to enhance the overall car parking experience, mitigate safety risks, and contribute to the efficient management of parking spaces in housing areas.

3.2 Project Flowchart

Figure 3.1 illustrates the sequence of tasks required to accomplish this project. In the initial stage, the literature study was conducted to gain a deeper understanding of car parking distance sensors in relation to Arduino systems and aimed the objectives of the project as well as identify the problem statement. By formulating survey questions related to the design, cost, and features of the project, a selection process was carried out to determine the appropriate design, sensor, and features. Subsequently, the software circuit was designed and tested through simulations. Following that, the hardware circuit was designed and tested, and the results were analysed and discussed based on the tested outcomes.

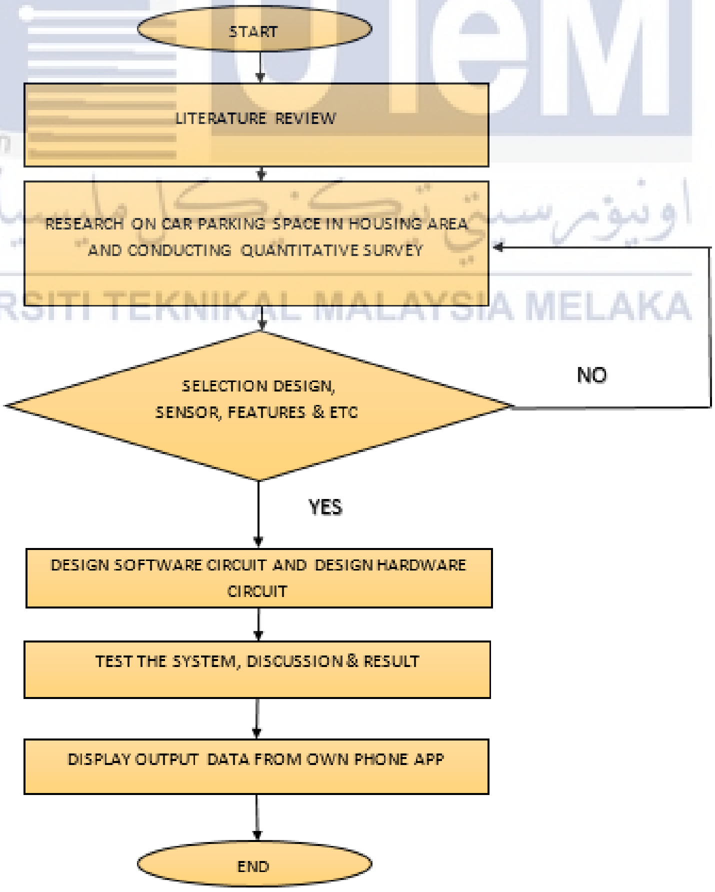


Figure 3.1 Flowchart of this project

3.3 Designed architecture

The project involves the implementation of a sophisticated car parking system using the Arduino Uno ATMega 328P microcontroller with Bluetooth module HC-06 as connectivity to mobile phone application. The primary objective is to efficiently manage parking distance by incorporating the Ultrasonic Sensor HC-SR04 to monitor the distance between obstacles and cars. The core components of this system include parking equipped with the Ultrasonic Sensor HC-SR04, which plays a pivotal role in constantly assessing the distance between the parked cars and potential obstacles. The Arduino Uno ATMega 328P microcontroller serves as the brain of the operation, orchestrating the interactions between the sensor and the parking system. The Ultrasonic Sensor HC-SR04, renowned for its accuracy and reliability, utilizes ultrasonic waves to measure the distance between the sensor and any object within its range. In this context, it precisely gauges the proximity of obstacles to parked cars, offering a real-time assessment of the parking environment.

The Arduino Uno ATMega 328P microcontroller takes on the responsibility of processing the data obtained from the Ultrasonic Sensor. It interprets the distance measurements and makes intelligent decisions to optimize parking space utilization. This microcontroller, known for its versatility and programmability, acts as the control center, ensuring seamless communication and coordination within the car parking system. Furthermore, the project aims to provide a user-friendly interface by incorporating a phone app. The Arduino Uno ATMega 328P microcontroller collaborates with the app to display the real-time distance between obstacles and cars. This feature enhances user convenience, allowing individuals to make informed decisions about parking spaces based on accurate and up-to-date information. This project combines cutting-edge technology with user-friendly applications to create an efficient and intelligent car parking system. The integration of the Arduino Uno

ATMega 328P microcontroller and the Ultrasonic Sensor HC-SR04 not only enhances the accuracy of distance measurements but also offers a seamless and intuitive experience for users through the phone app interface.

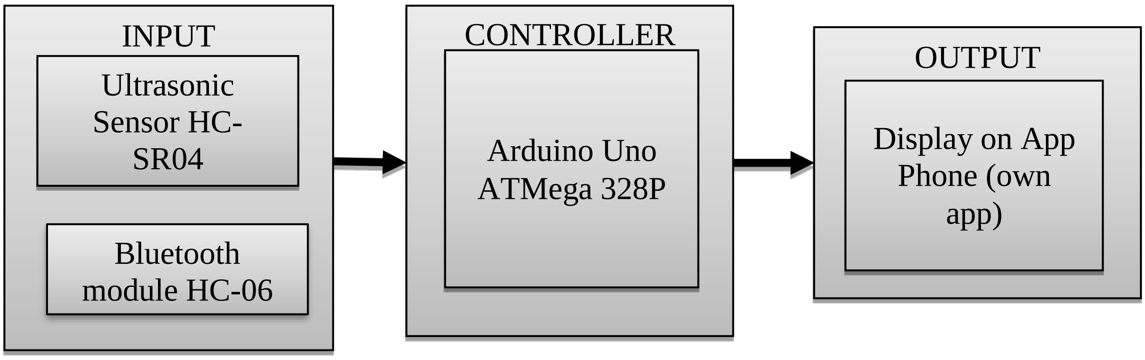


Figure 3.2 Block diagram of car parking distance sensor

3.4 Hardware used in car parking distance sensor with IoT

The car parking distance sensor project utilizes IoT technology and specific hardware components to create an efficient and smart car parking system. The main components include the Arduino Uno ATMega 328P microcontroller, Ultrasonic Sensor HC-SR04, Bluetooth module, and power supply. The project aims to address challenges in car parking by incorporating a distance sensor that measures the distance between obstacles and cars. The Bluetooth module enables wireless connectivity, allowing for real-time data transmission, remote monitoring, and control of the system. The integration of IoT technology and the hardware components enhances the functionality and convenience of the car parking system. It enables real-time monitoring, improves safety, and assists in efficient management of parking spaces.

3.4.1 Board selection

Figure 3.3 Overview of Arduino UNO ATmega328P (Techmakers Innovation, 2019)

In this project, Arduino UNO R3 board is chosen as it is a versatile microcontroller board based on the ATmega328P as in Figure 3.3. It is equipped with various features to support a wide range of projects. In addition to its versatility, the Arduino UNO R3 board offers the advantage of being a cost-effective option for the car parking distance sensor project. Despite its low cost, the Arduino UNO R3 board provides a robust set of features and capabilities that make it suitable for a wide range of applications. Compared to other microcontroller boards on the market, the Arduino UNO R3 board offers excellent value for its price. It provides all the essential functionalities needed for the car parking distance sensor project, including digital and analogue I/O pins, PWM support, and communication interfaces. These features allow for seamless integration of sensors, actuators, and other components required for the system's operation. Moreover, the Arduino UNO R3 board benefits from a large and active community of developers and enthusiasts. This provides extensive documentation, tutorials, and support, making it easier for beginners to get started with their projects. The availability of a vast number of open-source libraries and examples specifically designed for the Arduino platform further enhances its usability and reduces development time. Furthermore, the affordability of the Arduino UNO R3 board enables cost-effective prototyping and scalability. As the project progresses, if additional functionality or connectivity is required, it is relatively easy and inexpensive to expand the system using Arduino-compatible shields or modules.

Table 3.1 specifications of Arduino UNO ATmega328P

Board	**Name**	Arduino UNO R3
	SKU	A000066
Microcontroller	ATmega328P	
USB connector	USB-B	
Pins	**Built-in LED Pin**	13
	Digital I/O Pins	14
	Analog input pins	6
	PWM pins	6
Communication	**UART**	Yes
	I2C	Yes
	SPI	Yes
Power	**I/O Voltage**	5V
	Input voltage (nominal)	7-12V
	DC Current per I/O Pin	20 mA
	Power Supply Connector	Barrel Plug
Clock speed	**Main Processor**	ATmega328P 16 MHz
	USB-Serial Processor	ATmega16U2 16 MHz
Memory	**ATmega328P**	2KB SRAM, 32KB FLASH, 1KB EEPROM
Dimensions	**Weight**	25 g
	Width	53.4 mm
	Length	68.6 mm

By referring to Table 3.1, the board includes 14 digital input/output pins, with 6 capable of generating PWM (Pulse Width Modulation) signals. Additionally, there are 6 analogue input pins for reading analog sensor values. The board also features a 16 MHz ceramic resonator for precise timing. Connectivity options on the Arduino UNO R3 board include a USB-B connector for easy communication with a computer, as well as UART, I2C, and SPI interfaces for connecting to other devices and modules. Powering the board is straightforward, as it can be connected to a computer using a USB cable or powered using an AC-to-DC adapter or battery. The board is designed to provide all the necessary support for the microcontroller, and it includes a built-in LED on pin 13 for convenient visual feedback. The Arduino UNO R3 board consists of the ATmega328P as the main processor, running at a clock speed of 16 MHz. It also features an additional USB-Serial processor, the ATmega16U2, which enables seamless communication between the board and a computer. In terms of memory, the ATmega328P on the Arduino UNO R3 board has 2KB of SRAM (Static Random Access Memory), 32KB of flash memory for storing program code, and 1KB of EEPROM (Electrically Erasable Programmable Read-Only Memory) for non-volatile data storage. The dimensions of the Arduino UNO R3 board are 53.4 mm in width and 68.6 mm in length, with a weight of 25 g. For compatibility, Arduino IDE can easily write, compile, and upload code to the board, providing a user-friendly programming environment. The Arduino CLI offers a command-line interface for advanced users, enabling code compilation and uploading from a terminal. Additionally, the Arduino Web Editor allows editing and developing code directly in a web browser, providing convenience and online project storage. These software tools, including the Arduino IDE, Arduino CLI, and Web Editor, offer compatibility and flexibility, allowing the Arduino UNO R3 board to be programmed both online and offline. This compatibility enhances the accessibility and convenience of working with the board, empowering efficiently develop and deploy the car parking distance sensor system.

3.4.2 Sensor selection

In this project, a single sensor is chosen to be used, which is the Ultrasonic Sensor HC-SR04. The Ultrasonic Sensor HC-SR04 is responsible for detecting the distance between obstacles and cars in the parking area. The sensor operates by emitting ultrasonic waves and measuring the time it takes for the waves to bounce back after hitting an obstacle. This time measurement allows the sensor to calculate the distance between the car and the obstacle accurately. The Arduino software is utilised to interface with and control the Ultrasonic Sensor HC-SR04, enabling precise distance measurements. With the implementation of the Ultrasonic Sensor HC-SR04, the car parking distance sensor system can provide real-time information about the proximity of the car to obstacles.

Figure 3.4 Ultrasonic Sensor HC-SR04 (Robocraze, 2022)

Table 3.2 Specifications of Ultrasonic Sensor HC-SR04

Operating Voltage	**DC 5V**
Operating Current	**15mA**
Operating Frequency	**40KHz**
Max Range	**4m**
Min Range	**2cm**
Ranging Accuracy	**3mm**
Measuring Angle	**15 degree**
Trigger Input Signal	**10μS TTL pulse**
Dimension	**45 x 20 x 15mm**

The HC-SR04 ultrasonic distance sensor as in Figure 3.4 comprises two ultrasonic transducers that perform different functions. By referring to Table 3.2, one of the transducers serves as a transmitter, converting electrical signals into ultrasonic sound pulses at a frequency of 40 kHz. The other transducer functions as a receiver, detecting and listening for the transmitted pulses. When the receiver captures these pulses, it generates an output pulse which the duration is proportional to the distance of the object in front of the sensor. The HC-SR04 sensor offers reliable and non-contact range detection within a range of 2 cm to 400 cm (approximately 13 feet) with an impressive accuracy of 3 mm. Operating at a voltage of 5 volts, the HC-SR04 can be directly connected to an Arduino or any other microcontroller with 5V logic, making it easy to integrate into projects. This sensor provides a convenient and efficient way to measure distances and detect objects without the need for physical contact.

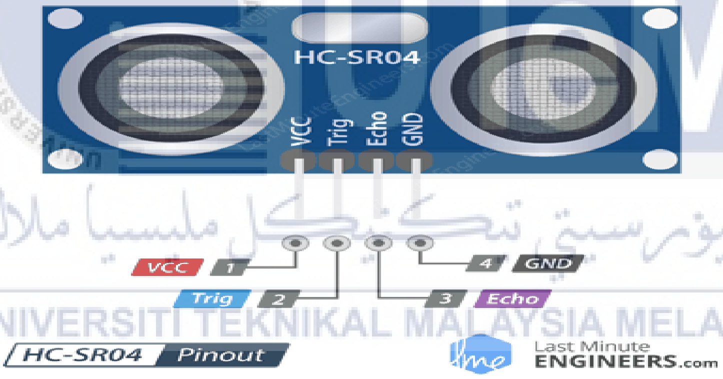


Figure 3.5 Types of pin HC-SR04 ultrasonic (Dejan, 2020)

The HC-SR04 ultrasonic sensor requires a power supply, which can be provided by connecting the VCC pin to the 5V output of Arduino or a compatible power source. The Trigger (Trig) pin is responsible for initiating the transmission of ultrasonic sound pulses. The sensor is triggered when the pin is set to a HIGH state for a duration of 10 µs, which starts the emission of the ultrasonic burst. Figure 3.5 shows that the Echo pin of the sensor goes high when the ultrasonic burst is transmitted and remains high until the sensor receives an echo, at which it goes low. By measuring the duration for which the Echo pin stays high, the distance can be calculated based on the speed of sound. To establish the electrical connections, it is important to connect the GND pin of the sensor to the ground (GND) of the Arduino or the common ground of the system being used. By ensuring proper wiring of these pins and implementing suitable programming logic, effective interfacing between the HC-SR04 ultrasonic sensor and the Arduino or microcontroller can be achieved. This enables the measurement of distances and facilitates distance-based tasks within the scope of this project.

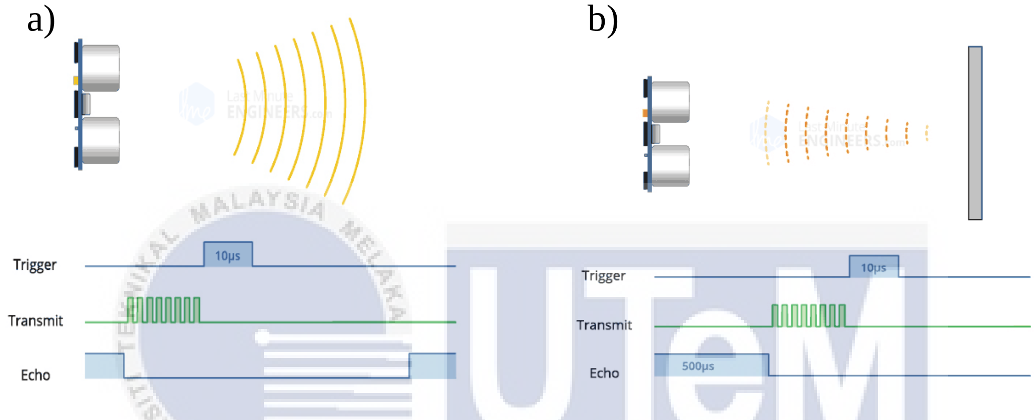


Figure 3.6 The pulse direction a) not reflected back, b) reflected back (Last min engineer, 2020)

The HC-SR04 sensor operation begins by setting the trigger pin to a HIGH state for a duration of 10 µs. As a result, the sensor emits a sequence of eight ultrasonic pulses at a frequency of 40 kHz. This specific 8-pulse pattern is designed to enable the receiver to distinguish the transmitted pulses from any surrounding ultrasonic noise. These eight ultrasonic pulses propagate through the air away from the transmitter, while the echo pin goes HIGH to initiate the reception of the echo-back signal. In Figure 3.6 (a) the event that these pulses are not reflected back, the echo signal eventually times out and transitions to a low state after approximately 38 ms. Consequently, a pulse duration of 38 ms indicates the absence of any obstructions within the sensor's range. However, in Figure 3.6 (b) the transmitted pulses are reflected back, the echo pin promptly goes low upon receiving the echo signal. This generates a pulse on the echo pin, and the width of this pulse varies between 150 µs and 25 ms depending on the time taken for the sensor to receive the echo signal. By measuring the width of the pulse on the echo pin, it is possible to calculate the distance to the reflecting object.

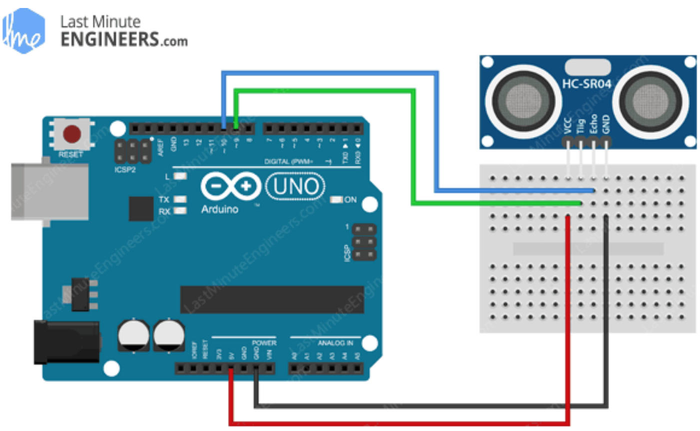


Figure 3.7 Wiring HC-SR04 Ultrasonic Sensor to Arduino UNO (Last min Engineer, 2020)

The connection of the HC-SR04 sensor to an Arduino is a straightforward process. Begin by placing the sensor on the breadboard. The connection was established by linking the VCC pin to the 5V pin on the Arduino and the GND pin to the ground pin. Then, the trig and echo pins are connected to digital pins number 9 and number 10 respectively, as shown in Figure 3.7. Figure 3.8 shows the code example related to this sensor and the output result.

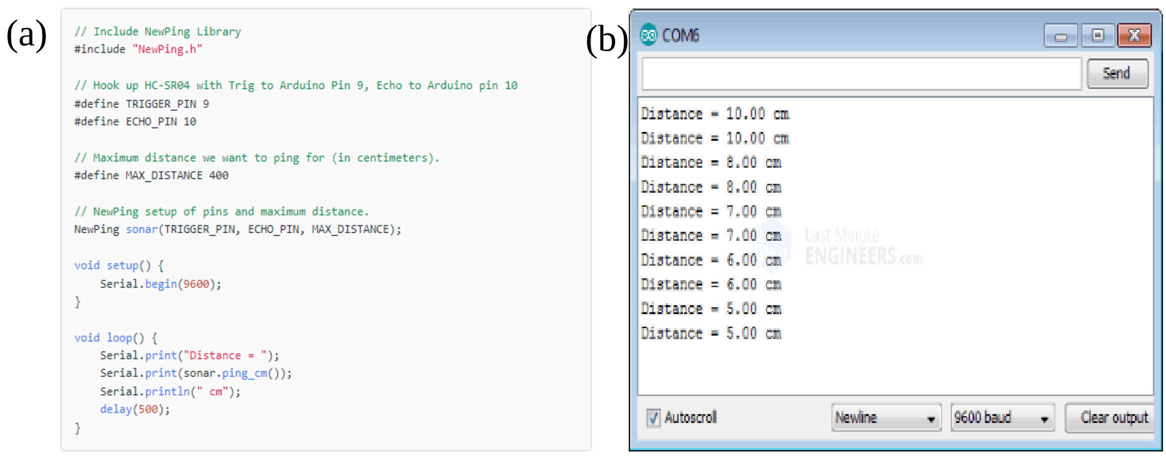


Figure 3.8 (a) Arduino coding example, (b) Output result (Last min Engineer, 2020)

3.4.3 IoT connectivity module

Figure 3.9 Bluetooth module HC-06 (RAMElectronics, 2019)

Figure 3.9 shows a HC-06 Bluetooth module that is commonly used wireless communication component for electronic projects. Operating on Bluetooth 2.0 (Class 2), it supports the Serial Port Profile (SPP) and typically functions at an operating voltage of 3.3V, although it can handle 5V signals. With a communication range of approximately 10 meters (Class 2 range) and an adjustable baud rate (commonly set to 9600 bps by default), the HC-06 utilizes UART (Universal Asynchronous Receiver-Transmitter) for interface. The default name of the Bluetooth device is HC-06, and its default pairing code (PIN) is typically set to 1234 but can be modified. Operating on the 2.4 GHz ISM band, it boasts low power consumption, making it suitable for battery-powered projects. Also HC-06 supports both master and slave modes, though it's commonly configured as a slave device. Its small physical dimensions facilitate easy integration into various projects, and it may feature indicator LEDs to display Bluetooth connection status. Configuration adjustments, such as baud rate, device name, and pairing code, can be made using AT commands. Widely compatible with microcontrollers like Arduino and Raspberry Pi, the HC-06 provides a cost-effective solution for wireless communication.

3.5 Simulation setup

For this project, the simulation was established using Tinkercad software, which incorporates all the necessary components for the car parking distance sensor project. Tinkercad software enables the creation of a virtual environment that replicates the real-world conditions and encompasses all the elements required for the functioning of the parking distance sensor in a car. The utilisation of Tinkercad software enables a thorough simulation of the project, guaranteeing accurate representation of all components and their interactions within a virtual environment. This approach allows for testing and validation of the parking distance sensor design's functionality and effectiveness before its physical implementation in the vehicle.

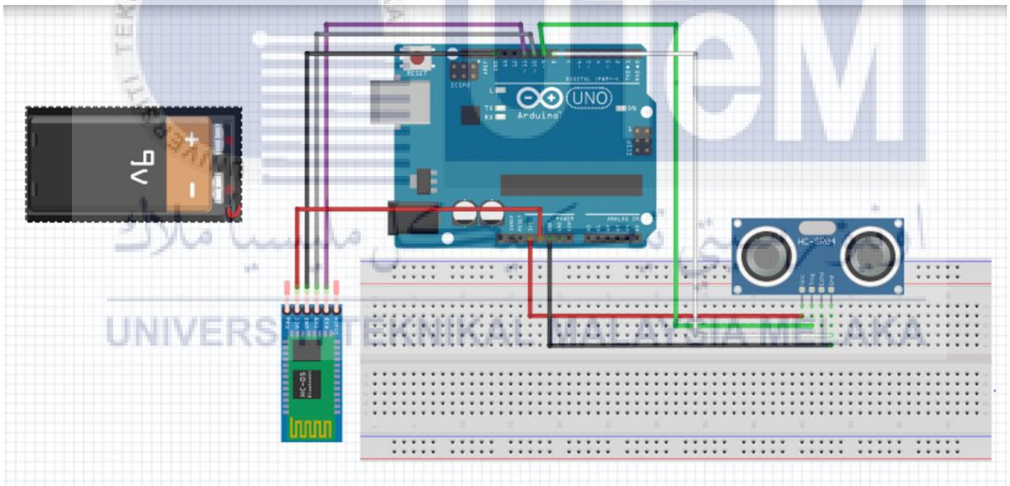


Figure 3.10 software circuit setup for this project systems

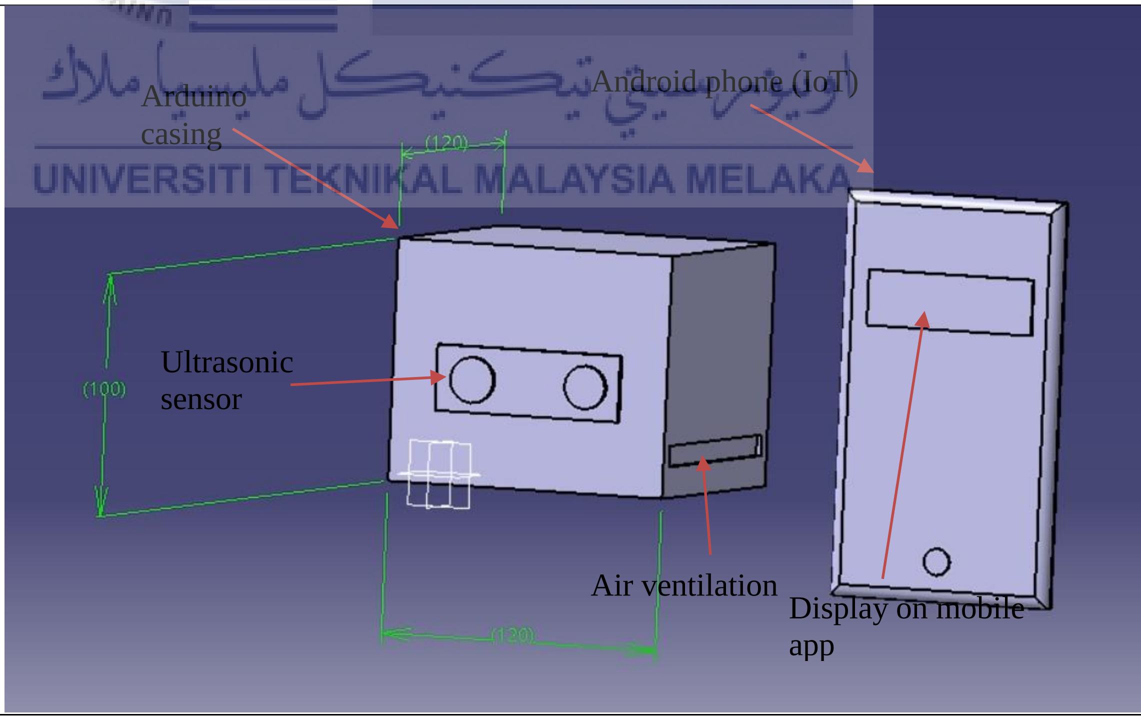
By implementing the configuration as illustrated in Figure 3.10, it becomes possible to conduct tests to determine the appropriate and maximum distances for a car to park in close proximity to an obstacle without any accidents occurring. The setup outlined provides a framework for conducting experiments and gathering data related to the car's parking distance sensor. The objective is to identify the optimal range at which the sensor can effectively detect obstacles and provide accurate display output to the driver. Through this testing process, different distances can be examined to assess the sensor's performance and ascertain the safe zone for parking near obstacles. The results obtained from these tests will inform the final design considerations for the car parking distance sensor, ensuring that it can reliably detect obstacles within the desired range and prevent any potential accidents during the parking maneuver.

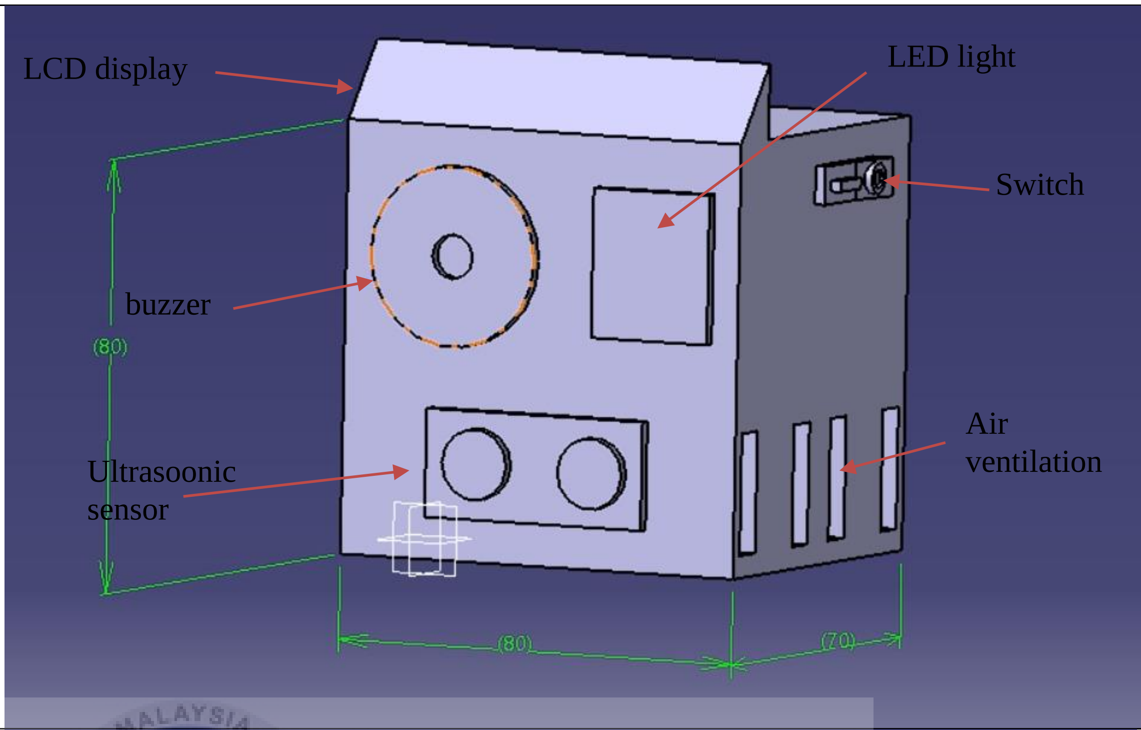
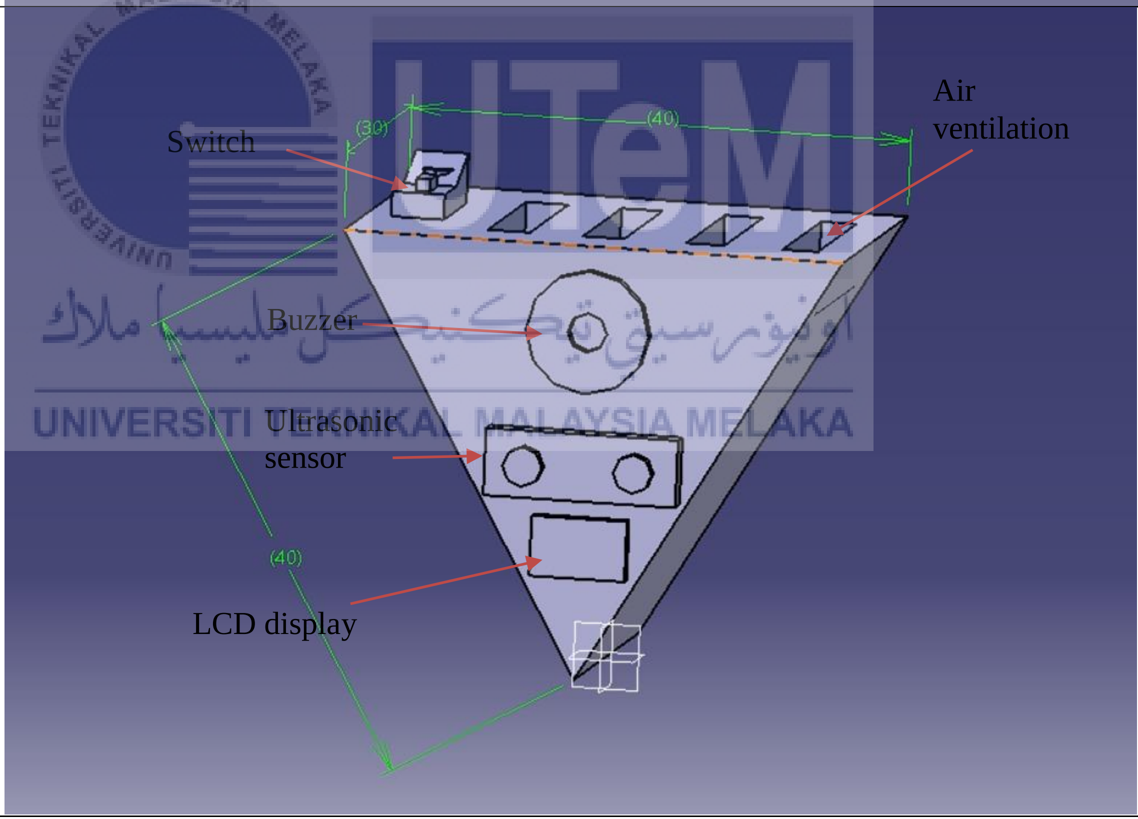
One of the key features of our IoT system is its ability to relay real-time distance information to the driver through a dedicated smartphone application. This app serves as a convenient interface, providing instantaneous feedback on the proximity of obstacles during the parking maneuver. By offering this dynamic and user-friendly display, drivers can make informed decisions and execute precise parking maneuvers with confidence.

3.6 Conceptual design

To ensure the optimal design for this project, several design options were proposed. During the design process, it is crucial to generate multiple designs before selecting the final one. Therefore, this subsection will provide an explanation of the designs that have been proposed. The project offers three design options as in Table 3.2 for the parking distance sensor system. Design A incorporates an IoT based system utilizing an ultrasonic distance sensor, Bluetooth module, and a dedicated smartphone application to display the distance output. Design B features a fixed buzzer and an LED light, with the buzzer emitting audible alerts and the LED light offering visual indication of the distance to the obstacle, synergistically enhancing safety and notifying the driver during the parking process. Design C introduces a visually appealing v-shaped style with a hanged LED light and buzzer, combining modern aesthetics with functionality. The hanged LED light provides clear visual indication of the distance, while the buzzer delivers audible alerts when the car approaches the obstacle, collectively ensuring heightened driver awareness and safe parking.

Table 3.3 Conceptual design comparison

Design	Features
A (IoT system with HC-06 and ultrasonic sensor with phone app)	

<table>
<tr>
<td>B
(Fixed buzzer and LED light position)</td>
<td>
</td>
</tr>
<tr>
<td>C
(V-shaped style with hanged LED light and buzzer)</td>
<td>
</td>
</tr>
</table>

3.6.1 Pugh method

The feedback in response to question 8 referring Appendix B guided the design process, and the selection of the design was determined through the utilization of the Pugh method. To determine the final design, the Pugh method was utilised as a means of evaluating and ranking the various design options. By employing a Pugh method, the rationale behind selecting the final design becomes more robust. Table 3.3 presents the Pugh table specifically used in this project, where weights were assigned on a scale of 1 to 3. Each design option was rated on a scale of -1, 0, and 1. The ratings were multiplied by their respective weights, and the totals were subsequently summarised. Based on the result design A has the highest score of 11.

Table 3.4 Pugh method

CRITERIA	WEIGHT	Design A with adjustable position buzzer and awning display	Design B with fixed buzzer and LED light	Design C is v-shaped style with hanged LED light and buzzer
FLEXIBILITY	2	1	-1	0
SAFETY FEATURE	3	1	0	-1
SIZE	2	0	1	0
SYSTEMS STYLE	2	1	0	1
VENTILATION FEATURE	3	1	0	0
RATE				
1		10	2	2
0		1	3	3
-1		0	-2	-3
SUMMATION		11	3	2

The Pugh Method employs a weight scale of 1, 2, or 3 to assess the importance or priority of each criterion. A weight of 1 signifies relatively lower importance, while a weight of 3 indicates a higher level of significance (Ted Hessing, 2021). This method follows several steps for evaluation. First, identify the criteria that are crucial for evaluating the design alternatives, ensuring their relevance. A score of -1, 0, or +1 was assigned to indicate whether each alternative performs worse, the same, or better than for each criterion. To account for the relative importance of the criteria, apply the weight scale by multiplying the scores of each alternative by the corresponding criterion weight. Calculate the total scores by summing up the weighted scores for each alternative. This provides a quantitative measure of the overall performance of each alternative based on the weighted criteria. Finally, determine the best alternative by comparing the total scores. The alternative with the highest total score is considered the most preferred choice. By utilising this simplified Pugh Method with the weight scale of 1 to 3, a systematic evaluation and comparison of design alternatives can be conducted based on their performance against important criteria. This approach facilitates the decision-making process by providing a structured framework for assessing and selecting the most suitable alternative.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

The focus on this project is to design an IoT systems for the car parking distance sensor that will display the output on the app phone using Arduino coding with connectivity HC-06 Bluetooth module. In order to ensure this project well function, the simulation and analysis was carried out. In this chapter, the final coding for IoT systems for car parking distance sensor has been discussed.

4.2 Ratio from real car size to model size

This project involved a testing methodology that utilised a demonstration concept employing miniaturisation. The objective was to accurately replicate the actual dimensions of various car types in a scaled-down model with a ratio of 1:24. This scale transformation allowed for a meticulous representation of the original car dimensions in a reduced form. To illustrate, consider a real car with specific dimensions such as height, length, and width. The process of miniaturization adhered to a conversion scale of 1:24. This implies that each unit in the real-world dimensions was shrunk to 1/24th of its size in the scaled-down model.

Figure 4.1 Actual dimension size for Sedan, SUV and MPV car

Figure 4.1 shows that the actual dimension scale for each type of the car, for the conversion scale, it can be calculated by using equation 4.1

$$\frac{\text{Original size}}{24} \qquad \text{(Equation 4.1)}$$

For a SEDAN car, the scale was u1:24. The height of 1544 mm, length of 5391 mm and width of 1950 mm was converted to 6.4 cm of height, 22.5 cm of length and 8.1cm of width as in Figure 4.2. Then, for the SUV the height was 1570 mm, length of 4360 mm and width of 1830 mm was converted to 6.5 cm of height, 18.2 cm of length and 7.6 cm of width. For MPV, that had height of 1945 mm, length of 5130 mm and width of 1900 mm was converted to 8.1 cm of height, 21.4 cm of length and 7.9 cm of width.

Figure 4.2 example of the sedan car in 1:24 scale from box

4.3 Position of the ultrasonic sensor

Figure 4.3 Measurement of possible sensor location from ground for sedan

For the ultrasonic sensor position, the distance from back rear camera or sensor was measured by using measuring tape. The average distance was calculated to obtained the suitable distance for each type of car. Figure 4.3 shows that the result was 77.4 cm distance from possible sensor location to the ground for sedan car. By using conversion scale from equation 4.1, the converted value was 3.2 cm.

Figure 4.4 Measurement of the possible location sensor from ground for MPV

By referring to Figure 4.4, the result was 89.7 cm distance from possible sensor location to the ground for MPV car and the converted value was 3.7 cm. Lastly, for the SUV car, it was assumed as same as MPV car, which the location sensor for converted scale was 3.7 cm. From all the distances, the average distance for location sensor was calculated to get appropriate location sensor distance for all types of car.

$$Average = \frac{location\ sensor(SEDAN + MPV + SUV)}{3} \qquad \text{(Equation 4.2)}$$

From equation 4.2, the calculated average for sensor location was 3.5 cm. The conversion scale used to implement the testing was 1:24. Hence, 3.5 cm was decided for location sensor or height of the sensor from the ground in miniature scale 1:24 while for real height of location sensor was 85.6 cm.

4.4 Arduino coding & equipment

This section provides a detailed explanation of the coding for the Arduino software, along with a comprehensive breakdown of the circuit diagram. Additionally, modifications to the circuit diagram have necessitated a change in the equipment or tools used for the circuit, and these adjustments are elaborated upon in the discussion.

4.4.1 Arduino coding

The Arduino coding need to be store in Arduino uno board and need to communicate to Bluetooth module for connectivity for distance measurement. In Figure 4.5 shows that the libraries used in this project.

```
#include <SoftwareSerial.h>
#include <NewPing.h>

```

Figure 4.5 The libraries used in this coding

The code incorporates two crucial libraries, which are SoftwareSerial.h and NewPing.h. SoftwareSerial aids in establishing extra communication channels for devices, beneficial for linking with external components. In the car parking project, it may be applied for communication between different parts of the system. The NewPing.h library simplifies the utilisation of ultrasonic sensors, like the HC-SR04, in Arduino projects. It furnishes pre-built functions for precise distance measurements. In this instance, it indicates the utilisation of ultrasonic sensors to gauge distances between cars and obstacles in real-time. Both libraries are pivotal for ensuring effective communication and dependable distance monitoring in the parking system.

```
SoftwareSerial BT(10, 11); // TX, RX respectively
NewPing distanceSensor(9, 8); // TRIGGER, ECHO
```

Figure 4.6 Initialized for Arduino-based project

Figure 4.6 shows lines of code, a communication channels are set up for the Arduino project. The first line, SoftwareSerial BT(10, 11), establishes a communication link between the Arduino and a Bluetooth module on pins 10 and 11, providing a wireless connection. The second line, NewPing distanceSensor(9, 8), configures an ultrasonic sensor on pins 9 and 8, allowing the Arduino to measure distances. This is crucial for assessing the space between the car and obstacles during parking. These lines initialise the communication components (Bluetooth and ultrasonic sensor) in our project.

```
void setup() {
  Serial.begin(9600);
  BT.begin(9600); // Initialize Bluetooth module
}
```

Figure 4.7 Void setup coding used

In the setup part of the code as in Figure 4.7, two essential things are happening. The line Serial.begin(9600); sets up communication with the computer or another connected device via a USB cable. It's like getting the Arduino ready to talk to the computer, and 9600 is the communication speed we've chosen. The line BT.begin(9600); initialises the Bluetooth module, making sure that the Bluetooth communication is set up and ready when our Arduino starts working. In a nutshell, these lines prepare Arduino to communicate with both the computer and the Bluetooth module at the beginning of the program.

```
void loop() {
  int distance = distanceSensor.ping_cm();
  if (distance < 0) {
    distance = 0;
  }

  Serial.println(distance);

  // Print only if distance is greater than zero with "cm" unit
  if (distance > 0) {
    BT.print(distance);
    BT.println(" cm");
  }

  delay(1000); // Delay to avoid flooding the Bluetooth connection with data
}
```

Figure 4.8 The void loop in the arduino coding

In the loop section as in Figure 4.8, the distance is measured using the ultrasonic sensor and stored in the 'distance' variable. If the distance somehow goes below zero, it's set to zero. The distance is then printed to the computer using Serial.println() for monitoring. After that, it checks if the distance is greater than zero. If true, the distance value is sent to the Bluetooth module and displayed with the unit "cm". To avoid overwhelming the Bluetooth connection, there's a delay of 1000 milliseconds (1 second) before the loop repeats, ensuring a steady flow of information.

4.4.2 Arduino circuit and equipment

The Arduino circuit for this project had to be changed due to the addition IoT systems. At first, the circuit is considered complex. However, after some modification of the coding and the circuit achieved a state of specification. The circuit for this project is shown in Figure 4.9 and Table 4.1 shows the equipment for the circuit and the quantity.

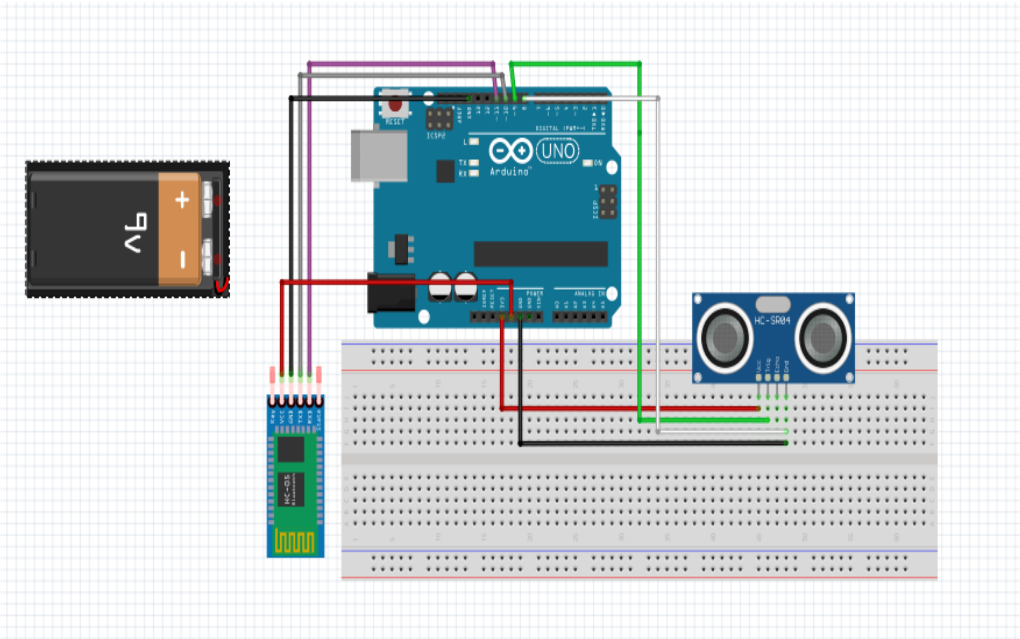


Figure 4.9 Circuit diagram for this project

Table 4.1 Equipment used

NO	EQUIPMENTS	QUANTITY
1	Arduino uno r3	1
2	Ultrasonic sensor hc-sr04	1
3	Bluetooth module hc-06	1
4	Battery 9v	1
5	Cable type-b	1
6	Bread board mini	1
7	Wire jumper	8

4.4.3 Development of phone app (MIT software)

In the implementation of this IoT systems project, the MIT App Inventor as shown in Figure 4.10 was employed for the development of a customised application. This application serves the purpose of presenting real-time distance measurements in centimeters, reflecting the outcomes of the distance measurement process.

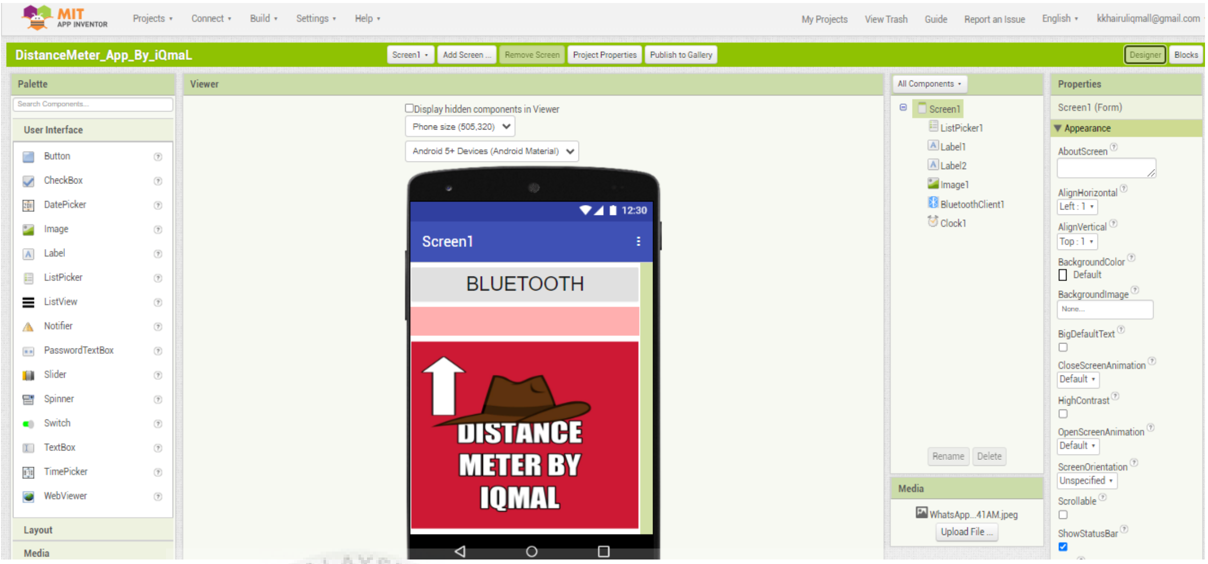


Figure 4.10 Interface of the MIT app inventor

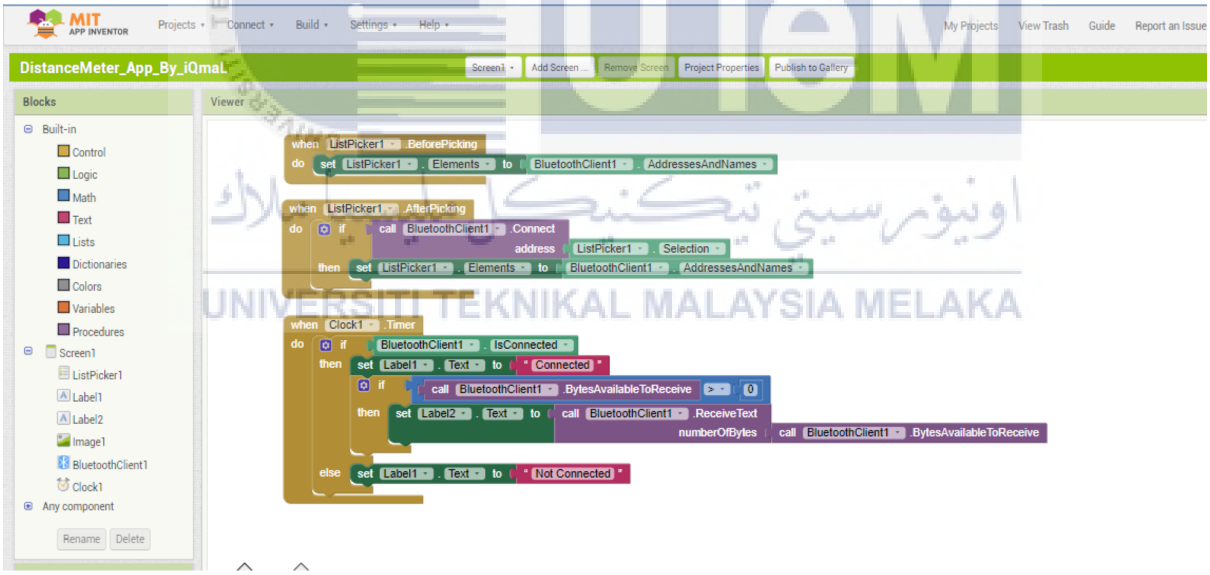


Figure 4.11 Coding blocks for distance meter app

Figure 4.11 illustrates the coding blocks employed in this project to establish a connection between the Bluetooth module HC-06 and the Arduino Uno R3 board. These coding blocks were instrumental in enabling the built application to seamlessly display real-time distance measurements. The integration of these components demonstrates a key aspect of the project's implementation, showcasing the effective utilization of technology to achieve the desired outcome in the form of a functional distance meter on the mobile application interface.

4.5 Test result

The development of IoT distance sensor measurement systems necessitate the ability to identify potential risks in any given scenario, including stringent conditions. The collected results are systematically tabulated for subsequent analysis, aimed at enhancing the efficiency and accuracy of real-time measurements.

4.5.1 Rawr test display on serial monitor arduino ide

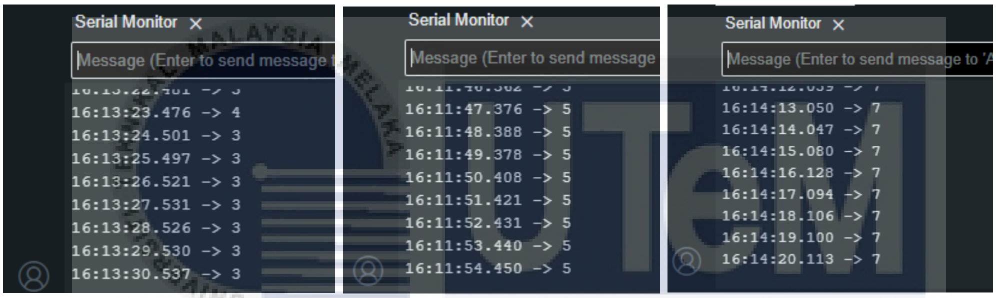


Figure 4.12 Output for 3 differents of distance on serial monitor

Figure 4.13 Example of the raw testing setup

Configuring the setup as depicted in Figure 4.13 reveals three potential output distances 3 cm, 5 cm, and 7 cm as Figure 4.12. A comprehensive analysis is required to determine the most suitable distance that aligns with the project's objectives. Given the emphasis on optimising proximity for efficient parking space utilisation, the distance of 3 cm has been judiciously selected as the most effective and appropriate metric, aligning with the overarching goals of the project.

4.5.2 Miniature test

This test was conducted to evaluate the device's performance across three different distances and with three distinct types of cars. The primary objective was to systematically analyse potential issues, including malfunctions and errors in data organisation. During these tests, a mobile phone was utilised as an output display to showcase real-time distance measurements, adding a dynamic element to the evaluation process.

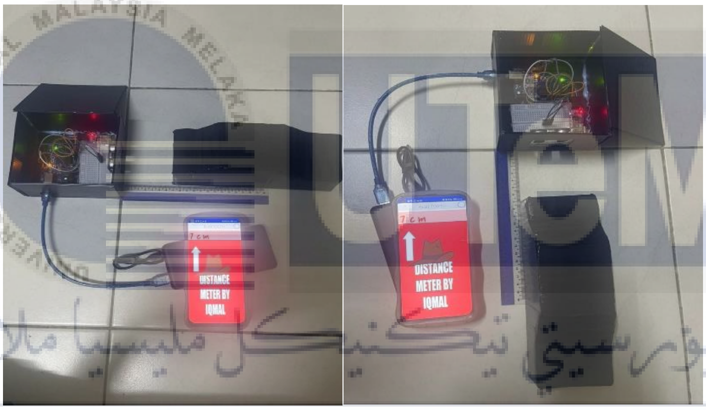


Figure 4.14 Ultrasonic sensor testing with Sedan car in 7 cm

Upon conducting a test with a Sedan car at a 7 cm distance, as illustrated in Figure 4.14, the obtained result indicated a discrepancy in measurement accuracy when compared to the ruler employed for real distance measurements. This observation highlights a deviation from the expected precision in the distance measurement during this particular test scenario.

Figure 4.15 Ultrasonic sensor testing with Sedan car in 5 cm

During the evaluation involving a Sedan car at a 5 cm distance, depicted in Figure 4.15, the acquired result demonstrated favourable measurement accuracy when compared to the ruler utilised for actual distance measurements. This finding underscores a satisfactory alignment with expected precision in distance measurement for this specific test scenario, validating the experimental outcome.

Figure 4.16 Ultrasonic sensor testing with Sedan car in 3 cm

During the assessment, which included the evaluation of a Sedan car at a 3 cm distance, as illustrated in Figure 4.16, the obtained result exhibited highly accurate measurements in comparison to the ruler utilised for actual distance measurements. This outcome reinforces a notable concordance with the anticipated precision in distance measurement for the specified test scenario, thereby substantiating the validity of the experimental findings.

During the evaluation, when a car approached the sensor at 7 cm, the distance measurement was found to be inaccurate, rendering the result invalid. However, when the car approached the sensor at distances of 5 cm and 3 cm, the distance measurements were accurate, establishing the validity of these results. This indicates that the device performs reliably at closer proximities, especially at 5 cm and 3 cm, meeting the intended accuracy criteria for distance measurements in these scenarios as stated in Table 4.2.

Table 4.2 Miniature testing

No	Description	Result	Validation
1.	Car approach to the sensor with distance of 7cm	The distance measurement not accurately	Invalid
2.	Car approach to the sensor with distance of 5cm	Distance measurement was accurate	Valid
3.	Car approach to the sensor with distance of 3cm	Distance measurement was accurate	Valid

The selection of a 3 cm distance was determined to be the most suitable for all types of cars after careful consideration and analysis. It was discussed and established that a closer proximity to the sensor results in a greater front space allowance for the vehicle within the designated section. This decision is grounded in a strategic evaluation of optimal spacing requirements to enhance efficiency and accommodate various car sizes effectively.

4.6 Discussion

The development of IoT distance sensor measurement systems underscore the importance of identifying potential risks in diverse scenarios, even under stringent conditions. To achieve this, systematic tabulation of collected results is conducted, facilitating subsequent analysis aimed at refining the efficiency and accuracy of real-time measurements. In configuring the setup, as illustrated in Figure 4.13, three potential output distances (3 cm, 5 cm, and 7 cm) are identified that was supported by previous study (Susilo, 2021), as shown in Figure 4.12. A comprehensive analysis is essential to determine the most suitable distance aligning with the project's objectives. The decision to judiciously select the 3 cm distance is motivated by the project's overarching goals, with a specific emphasis on optimizing proximity for efficient parking space utilization.

The conducted tests, encompassing various distances and car types, utilised a mobile phone as an IoT display for showcasing real-time distance measurements. This dynamic approach adds depth to the evaluation process, ensuring a thorough examination of potential issues, including malfunctions and errors in data organization. Specifically, tests with a Sedan car at 7cm distance revealed a discrepancy in measurement accuracy compared to the ruler employed for actual distance measurements (Figure 4.14). Conversely, tests at 5 cm (Figure 4.15) and 3 cm (Figure 4.16) distances demonstrated favourable and highly accurate results, respectively, aligning with the anticipated precision for these specific test scenarios.

Table 4.2 further highlights the evaluation outcomes, indicating inaccurate distance measurements at 7 cm but accurate readings at 5 cm and 3 cm distances. The device exhibits reliable performance, especially at closer proximities, meeting the intended accuracy criteria for distance measurements in these scenarios. The selection of a 3 cm distance as the most suitable for all car types is a carefully considered decision, grounded in a strategic evaluation of optimal spacing requirements. This choice aims to enhance efficiency and accommodate various car sizes effectively, ensuring a greater front space allowance within the designated section. Overall, the experimental findings and analyses contribute to a comprehensive understanding of the device's performance characteristics and its alignment with the project's objectives.

CHAPTER 5

5.1 Conclusion

In conclusion, the design and development of the IoT car parking distance sensor using Arduino has been successful. Overall, this system significantly enhances the user experience during parking. The integration of this system simplifies and secures the car parking process through the display on the connected mobile app. Notably, one of the significant challenges faced by drivers is accurately judging parking distances, especially in tight or crowded spaces with limited visibility, blind spots, and a lack of advanced camera technology. However, this project effectively addresses this issue by employing ultrasonic sensors to measure distances between the rear of the car and the sensor, applicable to various car types. Furthermore, the project incorporates safety and IoT features, wherein distance measurements are conveniently displayed on the driver's phone through a Bluetooth module connection. This comprehensive approach not only tackles the challenges associated with parking but also leverages technology to provide real-time information to the driver, promoting safer and more efficient parking experiences.

The system underwent analysis by assessing the ultrasonic sensor's height from the ground, which was then scaled down to 1:24, corresponding to 3.5 cm. Following this, three distances of 3 cm, 5 cm, and 7 cm were tested using both raw testing and miniature tests. The evaluation concluded that a 3 cm distance proved to be the most suitable for all types of cars, making the system well-suited for residential areas, even when positioned outside the house. Additionally, the system design distinguishes itself as the safest among the proposed alternatives. This underscores its reliability and safety features compared to other designs.

5.2 Future work

Here are recommendations for future developments in this project:

a) Enhance the body structure of the IoT car parking distance sensor by incorporating additional features.

b) Expand the functionality of the mobile app beyond simply displaying distance measurements. Include multiple functions and consider integrating audible outputs to enhance user attention.

c) Conduct extensive research on establishing standard measurements for this system and undertake a thorough analysis of distance testing methodologies.

d) Implement improvements to the measurement process, focusing on enhancing real-time measurements and addressing any identified issues for optimal system performance.

REFERENCES

Abdel-Hafez, M. N. (2011, january 1). *A sequential approach for fault detection and identification of vehicles'* . Retrieved from 2011 Fourth International Conference on : https://www.semanticscholar.org/paper/A-sequential-approach-for-fault-detection-and-of-Abdel-Hafez-Nabulsi/6e0709f05e5434f0729f7e12c3c1dc6b02d3dc19

Abou-Senna, H. A. (2020, april 9). . *How does parking availability interplay with the land use and affect traffic congestion* . Retrieved from AbstractStrategic development of parking space is considered urban congestion: https://www.sciencedirect.com/science/article/abs/pii/S221067072030113X

Alzahrani, S. M. (2017, august 23). *Sensing for the Internet of Things and Its Applications*. Retrieved from IEEE: https://ieeexplore.ieee.org/document/8113776

Anita Febriani, U. R. (2021, september 5). *Car Parking Distance Controller Using Ultrasonic Sensors Based On Arduino Uno*. Retrieved from Journal of Robotics and Control (JRC) Volume 2, Issue 5, September 2021: https://pdfs.semanticscholar.org/82fb/e71862a6c29e2a3bd973e7b53339ae05aef8.pdf

Arditi, D. C. (2010, june 20). *Issues and proposed improvements regarding condominium management in Taiwan*. Retrieved from Council for Economic Planning and Development: https://www.sciencedirect.com/science/article/abs/pii/S0197397508000374

Asadi, K. &., & Fitriawan. (1970, January 1). *Designing a car parking helper system in a garage using STM32: Semantic scholar*. Retrieved from 2019 3rd InternationalConference on Electrical, Telecommunication and Computer Engineering (ELTICOM): https://www.semanticscholar.org/paper/Designing-a-car-parking-helper-system-in-a-garage-Nasution-Wahyudi/171390717ab5166bfa6fef58045b989303f764e5

Budi, L. A. (2018). *The Development of Smart Parking System based on NodeMCU 1.0 using the Internet of Things*. Retrieved from IOP Conference Series: Materials Science and Engineering: https://iopscience.iop.org/article/10.1088/1757-899X/384/1/012033/meta

Budil, A. a. (2018, july 1). *IOPscience*. Retrieved from IOP Conference Series: Materials : https://iopscience.iop.org/article/10.1088/1757-899X/384/1/012033/meta

Daniel Ng Chiu Loong, S. I. (2019, august). *Machine vision based smart parking system using IoT*. Retrieved from TELKOMNIKA: http://eprints.utm.my/id/eprint/89967/1/YusmeerazYusof2019_MachineVisionBasedSmartParking.pdf

Ferdinan Ahmad Fakaruddee, K. H. (2022, NOVEMBER). *Development of ASmart Car Parking System by Using Arduino Uno*. Retrieved from BEAM: https://beam.pmu.edu.my/index.php/beam/article/view/13/26

Gede Susrama Mas Diayasa, N. L. (2020). *Progressive Parking Smart System in Surabaya's Open Area Based on IoT*. Retrieved from iopscience: https://iopscience.iop.org/article/10.1088/1742-6596/1569/2/022043/meta

Héctor Espinoza a, G. K. (2020, october). *Estimating the impact of the Internet of Things on productivity in Europe*. Retrieved from sciencedirect: https://www.sciencedirect.com/science/article/pii/S2405844020307805

Kim, S. a. (2018, december). *Investigating problems of vehicle parking in a low-rise residential area in Nangok-dong, Seoul*. Retrieved from IOP Conference Series Earth and Environmental Science: https://www.researchgate.net/publication/329971756_Investigating_problems_of_vehicle_parking_in_a_low-rise_residential_area_in_Nangok-dong_Seoul

M. Venkata Sudhakar a, A. A. (2021, july 4). *Development of smart parking management system*. Retrieved from sciencedirect: https://www.sciencedirect.com/science/article/abs/pii/S2214785321048926

Mendiratta, S., Dey, D., & Sona, D. R. (2017, AUGUST 17). *Automatic car parking system with visual indicator along with IoT*. Retrieved from IEEE EXPLORE: https://ieeexplore.ieee.org/abstract/document/8211549

Midde Ranjit Reddy, G. S. (2018, july). *Smart Vehicular System based on the Internet of Things*. Retrieved from ACM DIGITAL LIBRARY: https://dl.acm.org/doi/10.4018/JOEUC.2018070103

Muhamad Muzhafar Abd Kadir, M. N. (2020, OCTOBER). *IoT based Car Parking Management System using IR Sensor*. Retrieved from RESEARCHGATE: https://www.researchgate.net/publication/350489759_IoT_based_Car_Parking_Management_System_using_IR_Sensor

Salma, R. F. (2019, september). *Smart Parking Guidance System Using 360o Camera and Haar-Cascade Classifier on IoT*. Retrieved from International Journal of Recent Technology and Engineering (IJRTE): https://www.ijrte.org/wp-content/uploads/papers/v8i2S11/B11420982S1119.pdf

Seungwoo Seo, S. K. (2017, Mac 21). *Investigating problems of vehicle parking in a low-rise residential area in Nangok-dong, Seoul*. Retrieved from IOP Conference Series: Earth and Environmental Science: https://iopscience.iop.org/article/10.1088/1755-1315/213/1/012017

Susilo, J. (2021, september). *Car Parking Distance Controller Using*. Retrieved from Journal of Robotics and Control (JRC): https://journal.umy.ac.id/index.php/jrc/article/view/10184/5893

Waheb A. Jabbar, C. W. (2021). *An IoT Raspberry Pi-based parking management system for smart campus*. Retrieved from sciencedirect: https://www.sciencedirect.com/science/article/abs/pii/S2542660521000317

Wan-Joo Park, B.-S. K.-E.-S. (2008, JUNE 6). *Parking space detection using ultrasonic sensor in parking assistance system*. Retrieved from IEEE XPLORE: https://ieeexplore.ieee.org/document/4621296

WINGS, E. (n.d.). *LCD 16x2 Interfacing With Arduino Uno*. Retrieved from ELECTRONIC WINGS: https://www.electronicwings.com/arduino/lcd-16x2-interfacing-with-arduino-uno

APPENDICES

APPENDIX A : Gantt chart

ACTIVITIY	W 1	W 2	W 3	W 4	W 5	W 6	W 7	W 8	W 9	W 10	W 11	W 12	W 13	W 14	W 15
Title selection															
PSM1 briefing															
Briefing on project title with supervisor															
Report writing of literature review															
Correction of literature review															
Report writing of introduction															
Correction of introduction															
Report of survey questionaire															
Correction of survey questionnaire															
Report writing of Methodology															
Correction of Methodology															
Submission of 1st draft report															
Correction of 1st draft report															
Submission of final draft report															

PSM2 Tasks	Weeks													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Equipment Preparation														
Create and install the project														
Project Testing														
Result and Discussion														
Conclusion and Recommendation														
Pre-Submission Report														
Correction														
Final Report Submission														
Presentation														

APPENDIX B : Survey questions

SURVEY ON PARKING EXPERIENCES WITH SENSOR TECHNOLOGY

Development of Car Parking Distance Controller Using Ultrasonic Sensors and IoT Integration

SECTION A: Please answer all the questions.

1. Gender
   - Male
   - Female

2. Age (years old)
   - 18-25
   - 26-40
   - 41-60
   - Above 60

3. How many cars do you have per household?
   - Don't have any car
   - Only 1 car
   - 2-3 cars
   - More than 4 cars

4. What type of car do you own?
   - SEDAN
   - SUV
   - MPV

5. Have you ever encountered difficulties or challenging while parking your car in a narrow and limited residential area?
   - Yes
   - No

SECTION B: Circle the correct numeric response to each question.

NO	QUESTIONS	Survey scale: (1:Strongly Disagree), (2:Disagree), (3:Neutral), (4:Agree), (5:Strongly Agree)				
1	Do you often face problems when parking that only depends on the side mirror and rear mirror without the help of a parking sensor?	1	2	3	4	5
2	Do you agree that having a parking distance controller in your car would improve your parking accuracy and safety?	1	2	3	4	5
3	Do you agree that integrating IoT (Internet of Things) capabilities into the parking distance controller project can enhance its functionality and provide additional benefits to users?	1	2	3	4	5
4	Do you agree that a parking distance sensor is necessary, especially for cars that don't have a reverse parking sensor?	1	2	3	4	5
5	Do you agree that the car parking distance sensor is suitable for all types of cars, regardless of their size or model?	1	2	3	4	5
6	Do you agree that the parking distance controller should provide both VISUAL and AUDIBLE alerts to indicate the proximity to obstacles?	1	2	3	4	5
7	Do you agree that having a wireless connection for the parking distance controller would make it more convenient to use?	1	2	3	4	5
8	Do you agree that the parking distance controller should have a compact and unobtrusive design?	1	2	3	4	5
9	Do you agree that the parking distance controller should be affordable and cost-effective for car owners?	1	2	3	4	5
10	Do you agree that having a warning indicator for low sensor battery or system malfunction in the parking distance controller is important, even if it means increasing the overall cost of the project?	1	2	3	4	5
11	Do you agree that utilizing solar power for parking distance controllers can ensure a dependable and uninterrupted energy supply but may increase the overall construction cost of the project?	1	2	3	4	5

APPENDIX C : Full Arduino coding for distance measurement in cm

```
#include <SoftwareSerial.h>
#include <NewPing.h>

SoftwareSerial BT(10, 11); // TX, RX respectively
NewPing distanceSensor(9, 8); // TRIGGER, ECHO

void setup() {
  Serial.begin(9600);
  BT.begin(9600); // Initialize Bluetooth module
}

void loop() {
  int distance = distanceSensor.ping_cm();
  if (distance < 0) {
    distance = 0;
  }

  Serial.println(distance);

  // Print only if distance is greater than zero with "cm" unit
  if (distance > 0) {
    BT.print(distance);
    BT.println(" cm");
  }

  delay(1000); // Delay to avoid flooding the Bluetooth connection with data
}
```

APPENDIX D : Full coding block from MIT app iventor