PROJECT REPORT
ON

Self Parking System

Carried Out at

CENTER FOR DEVELOPMENT OF ADVANCED COMPUTING

ELECTRONICCITY, BANGALORE

UNDER THE SUPERVISION OF

Dr. Vishal J. Rathod

Submitted By

Madhusudhan Amudala (220350130004)

Nikhil Yadav (220350130005)

Priyanka Arora (220350130007)

Himanshu P. Raipure (220350130008)

Suryansh Dhakare (220350130010)

PG DIPLOMA IN EMBEDDED SYSTEM AND DESIGN
C-DAC, BANGALORE

<u>Candidate's Declaration</u>

We hereby certify that the work being presented in the report entitled **Self Parking System**, in partial fulfillment of the requirements for the award of PG Diploma Certificate and submitted in the department of PG-DESD of the C-DAC Bangalore, is an authentic record of our work carried out during the period, August 2022 to September2022 under the supervision of **Mr. Shrikrishna S Chippalkatti**, C-DAC Bangalore. The matter presented in the report has not been submitted by us for the award of any degree of this or any other Institute/University.

(Name and Signature of Candidate)

- Madhusudhan Amudala (220350130004)
- Nikhil Yadav (220350130005)
- Priyanka Arora (220350130007)
- Himanshu P. Raipure (220350130008)
- Suryansh Dhakare (220350130010)

Counter Signed by

____

ACKNOWLEDGMENT

We avail this opportunity to express our sincere thanks to our esteemed guide **Dr. Vishal J. Rathod,** for splendid guidance, authentic supervisor, cooperation and moral support throughout this project.

We would like to express our profound gratitude to **Mr. Shrikrishna S Chippalkatti** Training coordinator for creating required facilities to complete this project.

We would like to express our sincere thanks to our staff and management of **C-DAC** for giving all support and suggestions to complete our project.

We acknowledge here out debt to those who contributed significantly to one or more steps. We take full responsibility for any remaining sins of omission and commission.

Student's Name

- Madhusudhan Amudala (220350130004)
- Nikhil Yadav (220350130005)
- Priyanka Arora (220350130007)
- Himanshu P. Raipure (220350130008)
- Suryansh Dhakare (220350130010)

CERTIFICATE

This is to certify that this is a bonafide record of project work on the **Self Parking System** done by **Madhusudhan Amudala** (04) **Nikhil Yadav** (05) **Priyanka Arora** (07) **Himanshu P. Raipure** (08) and **Suryansh Dhakare** (10) under the guidance of **Dr. Vishal J. Rathod** Sir in partial fulfillment of the requirement of a Diploma in Embedded System Design at C-DAC Bangalore for the academic session of April 2022.

Mr. Shrikrishna S Chippalkatti
CDAC #68, Electronic City,
Bangalore-560100, India

ABSTRACT

This project involves building a model of a self-parking system, a machine that increases the amount of parking available inside a lot. Technology parks the car in a designated spot once the driver departs the vehicle inside an entrance area. For roughly 66% of tourists, finding parking spaces on weekends or holidays can take additional time. Peak times are busy in stadiums or shopping centers, and customers frequently have trouble finding open seats in these locations. Create an intelligent, user-friendly automated car parking system that decreases congestion and increases parking capacity inside a parking lot to combat this scenario. Technology parks the vehicle once the driver exits the vehicle inside an entrance area.

TABLE OF CONTENT

List of Figures

CHAPTER 1
INTRODUCTION

This project deals with the manufacture of a Prototype of a Self Parking System which is a mechanical device that multiplies parking capacity inside a parking lot. The driver leaves the car inside an entrance area and technology parks the vehicle at a designated area.

1.1 About Project

In this project we make a prototype of underground automated parking system using Arduino UNO board and IR sensors.

1.2 Scope of Project

To develop an intelligent, user-friendly automated car parking system that reduces - manpower and traffic congestion to offer safe and secure parking slots within a limited area.

1.3 System Requirement

1.3.1 Hardware and its specification

(i) Arduino UNO board

Fig 1.1 Arduino UNO

Specification:

- Microcontroller: ATmega328P
- 5V is the operating voltage.
- 7-12V is the recommended input voltage.
- 6-20V input voltage (limit)
- Digital I/O Pins: 14 (of which 6 provide PWM output)
- PWM Digital I/O Pins: 6
- Analog Input Pins: 6
- 20 mA DC current per I/O pin
- 50 mA DC current for 3.3V Pin
- - 32 KB (ATmega328P), of which 0.5 KB is used by the bootloader.
- SRAM: 2 KB (ATMega328P)
- EEPROM: 1 KB (ATmega328P).
- Clock frequency: 16 MHz
- LED_BUILTIN: 13

(ii) Relay Board

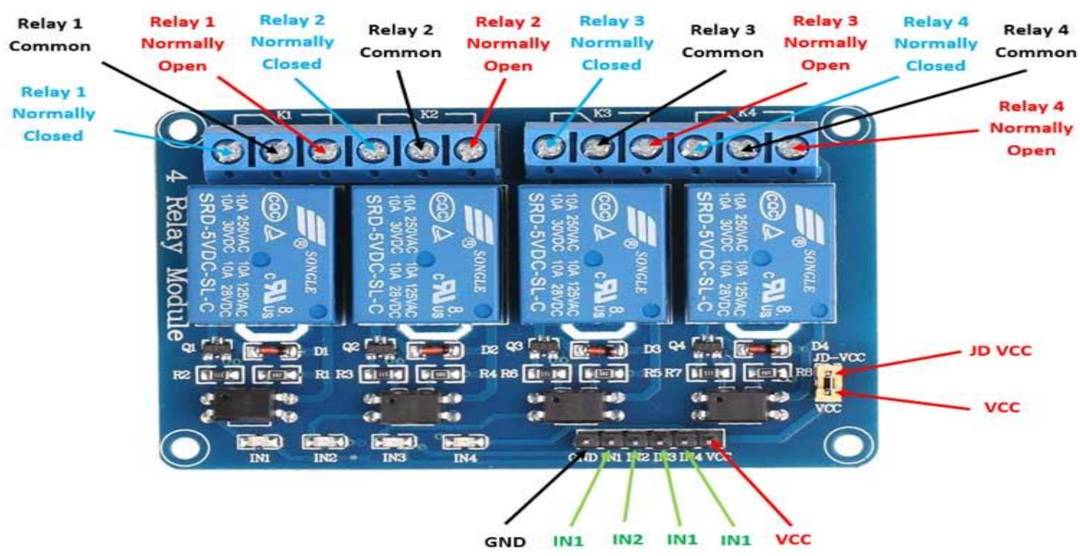

Fig 1.2 Relay board

Specification:

- 4-Channel Relay interface board, and each one needs 15-20mA Driver Current.
- Both controlled by 12V and 5V input Voltage.
- Equipped with high-current relay, AC250V 10A, DC30V 10A.
- Standard interface that can be controlled directly by microcontroller (Arduino, 8051, AVR, PIC, DSP, ARM, ARM, MSP430, TTL logic active low).
- Opto-isolated inputs.
- Indication LED for Relay output status.

(iii) SMPS (Switching mode power supply) :

Fig 1.3 SMPS

Specification:

- Input Voltage: AC 100 - 240V 50 / 60Hz.
- Output Voltage: 12V DC, 2A.
- Output Power :24W (max)
- Protections: Overload / Over Voltage / Short Circuit.

(iv) DC Motors

Fig 1.4 DC Motor

Specification:

- Operating Voltage(V): 12
- Rated Speed (RPM): 200

(v) IR sensors

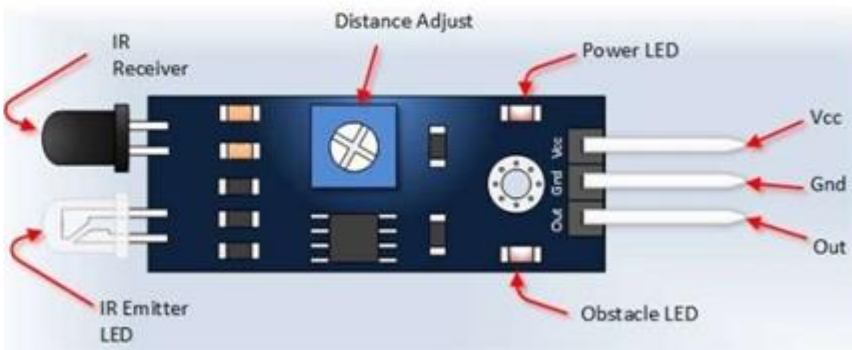

Fig 1.5 IR Sensor

Specification:

- Operating Voltage 3.3 ~ 5 VDC
- Distance Measuring Range 2 ~ 30 CM

1.3.2 Software and its Specification

(i) Arduino IDE	Version: 1.8.19 contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus.
(ii) Microsoft Office	Version 2013
(iii) Draw.io	for Diagrams

CHAPTER 2
BACKGROUND AND RELATED WORK

2.1 Related Work

Previously obtained work on the topic of self-parking is used in context with the ongoing research for the current project.

2.1.1 Internet of things

During the late 90s, Kevin Ashton, a director at AUTO-ID centre at the Massachusetts Institute of Technology (MIT), coined the term Internet of Things (IoT) [5]. Ashton explained that he wanted a system that “empowers computers with their own means of gathering information, so that they can see, hear and smell the planet for themselves, altogether its random glory”.

In other words, IoT will be accustomed link the physical world to the digital world through the internet [5]. As of today, IoT remains developing as more objects are adopting this method [6]. The word “things” within the internet of things will be applied to all or any styles of devices that may be utilized in lifestyle while the words “internet" refers to the wireless connection made through the web [7]. The main objective of IoT is to attach several devices to every other and this may be finished the assistance of various technologies, like frequency Identification (RFID), Wireless Sensor Networks (WSN) and Cloud Computing [6].

Through IoT, individuals should be ready to access and control their devices whenever and wherever they need as long as they're connected to the internet [3]. As an example, in an exceedingly smart home, IoT devices will be able to control the temperature and therefore the light. As mentioned, IoT as a system relies heavily on the internet for it to function and it's therefore highly at risk of security breaches [3] for instance, a tool will be hacked by individuals that ought to not have access to that the particular authorised user can, because of these attacks suffer from severe privacy violations. Therefore, implementation of security measures is crucial to make sure that the privacy of the users is violated. IoT is attacked in several ways supported where the attack occurs. as an example, attacks can either be physical or dematerialized. With physical attacks, the attacker has access to the device itself and might therefore destroy it or temper with it physically [7].

However, security breaches through physical attacks don't seem to be relevant for this thesis and can therefore not be examined or considered throughout the remainder of this paper. Dematerialized attacks, on the opposite hand, don't require physical access. Instead, an attacker can target the software round the device to attack [7], meaning either the network layer, the applying layer, or the transport layer. Attacks on the network layer mean that the attacker aims to disturb the web communication between two devices, as an example by targeting the traditional routing information or the traffic analysis. These forms of attacks can as an example be executed with the assistance of spoofing where the attacker is in a position to swap the transmission of legitimate data to instead transmit malicious data. Attacks on the appliance layer are executed to focus on the end-user software of the IoT devices [7] the often attacked with the assistance of Denial of Service (DoS) attacks which are applying layer are commonly wont to exhaust the available resources of the server, like the memory or the bandwidth of [8]

By exhausting the resources, the client’s legitimate access to the server is going to be denied. Attacks within the transport layer implies that the delivering of information between devices may well be in danger. so as to deliver data between devices of IoT [7], transmission control protocols (TCP) are used. Similarly, to the applying layer, within the previous paragraph, the transport layer may also be attacked with the assistance of DoS attacks which during this case aim to exhaust and monopolize the information delivering function of the protocols. In this project, the most focus are going to be on attacks that occur within the Application layer of IoT, particularly the Message Queuing Telemetry Transport (MQTT) protocol which could be a quite common standard that's implemented on top of the TCP [9].

2.1.2 HTML and CSS

The **Hyper-Text Markup Language,** or **HTML,** is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

2.1.3 Github

GitHub, Inc. is an Internet hosting service for software development and version control, with software feature requests, task management, continuous integration, and wikis for every project. It is commonly used to host open source software development projects. As of June 2022, Hub reported having over 83 million developers and more than 200 million repositories, including at least 28 million public repositories. It is the largest source code host as of November 2021. It provides the distributed version control of Git plus access control, bug tracking,

2.2 Background

According to the research, the user and the system engage with one other sporadically. Although other parking systems have distinct protocols, awareness of a parking system like the one at Sunway Pyramid, the method is often as follows. As soon as the consumer clicks the machine's button, a ticket will emerge, which he should take before the gate opens. The client will now move to the accessible space. The green light bulb on top of each available lot designates the spots that are still available. The red light over the parking lot signifies either a reserved place or a vehicle parked there. There is a sensor available that will read the card if the user is using a "touch and go" (Malaysian prepaid card), and if there is enough money, the system will deduct the charges. However, there are cash payment devices that the consumer can slip his ticket into, and the machine will read it, estimate the number of hours spent, and calculate the amount the user must pay. The customer must put the amount of money specified by the system into the machine. The ticket will be verified by the system and given 15 minutes, which is ample time for a person to drive their car to the closest exit gate.

A number of technologies have been deployed in the endeavor to solve the parking problem. While some parking solutions are deployed as stand-alone technologies in some situations, in other situations multiple technologies are combined to achieve a given goal.

These technologies include-

(1) Camera-based sensor technology

(2) Infrared sensor technology and

(3) Ultrasonic sensor technology.

1. Camera-based sensor technology Access control systems have been developed and installed in various parking lots. At the airport in Cape Town, an access control point was put in place. It works in such a way that a driver requests for a ticket at the entrance of the parking by pressing a button. When a ticket is issued, the driver is granted access to enter the parking bay [5]. The ticket issued is marked with a code that records the time of entrance to the bay. At the time of exit, the driver has to pay for the time the car spent in the bay. Payment is done by inserting the ticket in another machine which reads the code on the ticket and calculates the amount of time spent. Then a parking fee is charged according to the time spent. Sensors are used at entrance and exit points to keep track of the cars in and out. A display board (variable message sign) is used at the entrance to show whether the bay is full or still has unused slots. This technology helps with the management of payment of use of the parking bays. It also gives users a general idea of the number of available parking slots by displaying a message like “level 1 full”. However, 5 it doesn’t give much help to the driver in finding an exact parking space in the shortest time possible nor allows localization of his car in a given parking spot using for example Radio Frequency Identification (RFID) technology.

2. Infrared sensor technology In Cavendish shopping mall in Claremont, Cape Town, and the access control system is used as well as another system that checks the status of each parking point. The system uses an infrared sensor to detect if a car is parked in a place. A light is fitted above each parking point to show the status. The light operates in the same way as the traffic lights on the road. However the meaning attached to the light colors is slightly different. A red light means that there is a car parked while a green light is used to express a free parking. There

is a section designated for parents with kids. This section has an orange light to show that parking is meant for parents with children. This section also uses the red and green lights to show whether the parking spot is free or used. Another section is reserved for the disabled. It is marked with a blue light. This section also uses the red and green light to show the status of the parking. This system is helpful to the drivers once they go through the entrance. It is relatively easy for the driver to look ahead and figure out where there are green lights in the bay. However, this is only possible for the corridor in which the driver is moving through. The driver has to keep looking until they can land on a green light. However, there is still a likely flaw that a driver may see a green light yet find it red and in-use by the time he reaches the spot.

3. Ultrasonic sensor technology: Siemens AG developed a similar system (SIPARK) that uses ultrasound sensors. These are installed above every single parking to determine whether the parking space is occupied or not [12]. This has been used in more than 70,000 parking spaces. The system is being used with better pay points that accept a wide range of payment options ranging from cash to noncash options such as credit cards. In a report by M. Crowder and C. Michael Walton, the need for intelligent transportation systems (ITS) is examined across a wide range of places [8]. These include business districts, airports, and transit stations. The report emphasizes that university premises are yet a new pressure zone with an increasing need for a parking solution. This report further details the need for real time provision of information to motorists as a remedy to reduction in traffic congestion. Among the systems examined in this report are: Saint Paul Advanced Parking Information system (St Paul APIS) in Minnesota, Phoenix Arizona intelligent parking system, Seattle center APIS, and a few others. In the St Paul APIS [8], static signs are placed in town to direct motorists to where the parking bays are located and the general condition of the road. These signs variable message signs(VMS) have the ability to display different messages as Set by the traffic control staff. Parking could be an expensive process in terms of the money or the time and effort spent. Current studies have revealed that a car is parked for 95 percent of its lifetime and only on the road for the other 5 percent [9].

If we take England in 2014 as an example, on average a car was driven for 361 hours a year according to the British National Travel Survey [10] yielding about 8404 hours in which a car would be parked. Now where would you park your car for these very long hours? Cruising for parking is naturally the first problem caused by the increase of car owners globally. On average, 30 percent of traffic is caused by drivers wandering around for parking spaces [1]. In 2006, a study in France revealed an estimation that 70 million hours were spent every year in France only in searching for parking which resulted in the loss of 700 million Euros annually [11]. In 2011, a global parking survey by IBM [2] stated that 20 minutes are spent on average in searching for a coveted spot. With these statistics, we can predict that a great portion of global pollution and fuel waste is related to cruising for parking [12]. Parking spaces are found to be more than plenty in some places and very rare to find in others. Pricing policies had played an important role in the overall parking 10 Literature Review availability for decades [13].

Here comes the important question: do we need to have more parking spaces or do we need better parking management? Numerous countries are working to manage their current transportation systems and road infrastructure to enhance traffic flows, mobility and safety [5]. Emerging from these motivations is the concept of Intelligent Transportation Systems (ITS). ITS are advanced applications applied to transport and infrastructure to exchange information between frameworks for enhanced productivity, safety and environmental performance. ITS vary in technologies applied, from basic management systems; navigation systems, traffic control systems, speed cameras, variable message signs; and to more advanced applications that fuse live information and feedback from other sources, such as Parking Guidance and Information (PGI) systems and Parking Reservation Systems (PRS).

CHAPTER 3

PROPOSED METHOD

3.1 Working Flow

First we have to select whether we have to do parking or un-parking. If we select parking, it checks car availability. If a car is there, it automatically selects the vacant lot using an IR sensor and parks the car. If we select un-parking, then it asks about slot selection and, as per selection, it brings out the car.

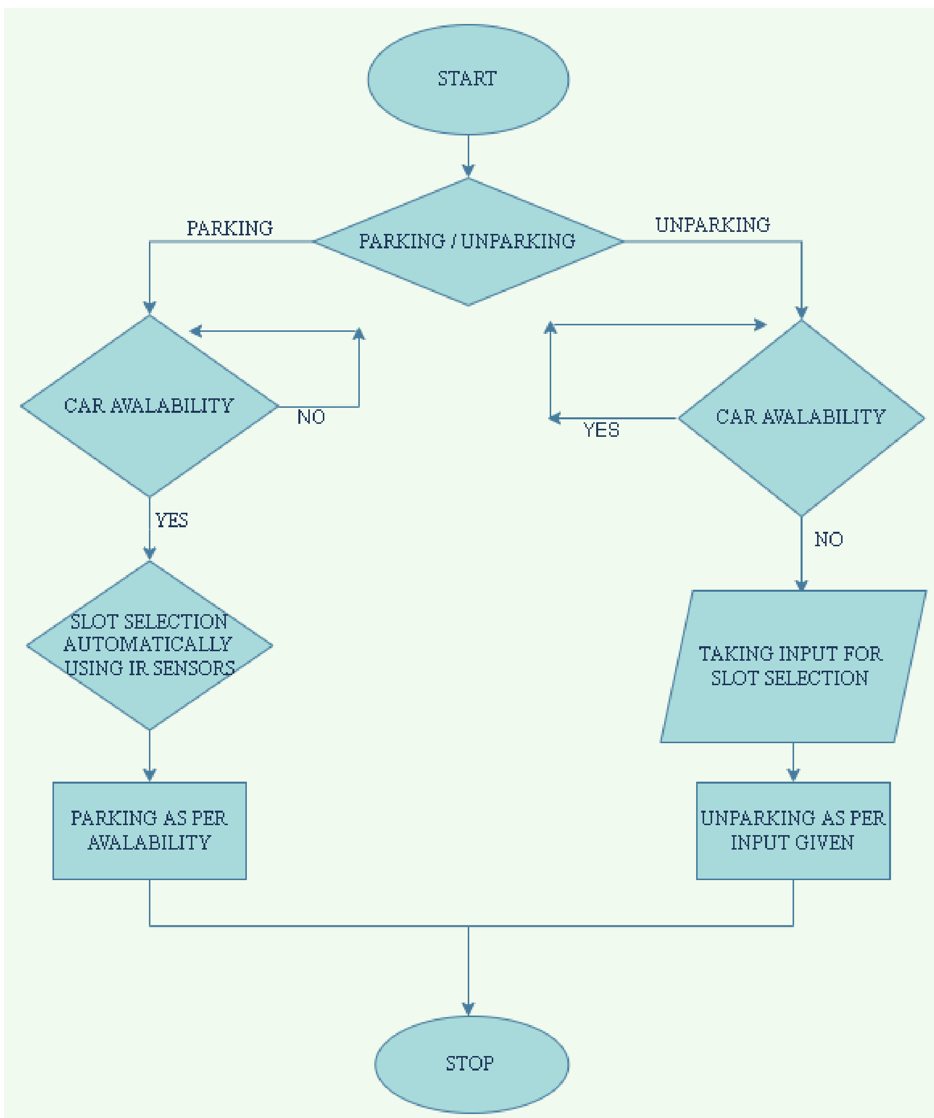

Fig. 3.1 Flow Chart of self Parking System

3.2 IMPLEMENTATION

For implementation, Arduino IDE is used. The code written consists of 390 lines. The code can be found on Github.

For Reference get a look at the github.io link: [18]

3.2.1 Parking and Un-parking code

The parking and un-parking of the vehicle is done at the choice of the owner. The IR sensors check if the vehicle is available and direct it to an empty slot if one is available. While un-parking, the slot is selected by the user where a specific vehicle is parked.

```
void parking()
{
  if ((irs_3() == 1) && (irs_1() == 0 || irs_2() == 0))
  {
    UG_0();

    if (irs_1() == 0)
    {
      UG_1();
      delay(1000);
    }
    else if (irs_2() == 0)
    {
      UG_2();
      delay(1000);
    }
    else
    {
      Serial.println("parking lot is full\n");
    }
  }
}
```

```
void unparking()
{
  if (irs_3() == 0)
  {
    Serial.println("choose the slot to perform 1.slot1 unparking 2.slot2 parking\n");
    while (Serial.available() == 1)
    {
      /*The condition of the empty while loop is Serial.available()==0. When there is no
      input from the user, the Serial.available() function returns a zero value, making
      the condition true. The sketch stays inside the while loop until the user inputs
      something and the Serial.available() returns a non-zero value.*/
    }
    int Choice = Serial.parseInt();

    switch (Choice)
    {
    case 1:
      // unparking code goes here
      Serial.print("unparking slot1 .... \n");
      unparkslot1();
      break;

    case 2:
      // unparking code goes here
      Serial.print("unparking slot2..... \n");
      unparkslot2();
      break;

    default:
      Serial.println("Please choose a valid selection for unparking\n");
    }
  }
  else
  {
    Serial.println("vechicle is dectected cant perform unparking\n");
  }
}
```

Fig. 3.2 Parking and Un-parking

3.2.2 Motor Driving code

Motor drivers are used for two motors which are responsible for moving the vehicle to a specified slot and parking the vehicle in that slot respectively. There are two types of functions, forward (for) and reverse (rev) for controlling the motors.

```
void motor2_for()
{
  digitalWrite(forwards, LOW);
  digitalWrite(backwards, HIGH); /* Activate the relay one direction,
                                  they must be different to move the moto
  delay(300);                    // wait 2 seconds
  digitalWrite(forwards, HIGH);
  digitalWrite(backwards, HIGH); // Deactivate both relays to brake the mo
  delay(300);                    // wait 2 seconds
}
```

```
void motor2_rev()
{
  digitalWrite(forwards, HIGH);
  digitalWrite(backwards, LOW); /* Activate the relay the other direction,
                                 they must be different to move the motor*/
  delay(300);                   // wait 2 seconds
  digitalWrite(forwards, HIGH);
  digitalWrite(backwards, HIGH); // Deactivate both relays to brake the moto
  delay(300);                    // wait 2 seconds
}
```

```
void motor1_for()
{
  digitalWrite(up, LOW);
  digitalWrite(down, HIGH); /* Activate the relay one direction,
                             they must be different to move the mot
  delay(14000);             // wait 2 seconds
  digitalWrite(up, HIGH);
  digitalWrite(down, HIGH); // Deactivate both relays to brake the mo
  delay(1000);              // wait 2 seconds
}
```

```
void motor1_rev()
{
  digitalWrite(up, HIGH);
  digitalWrite(down, LOW); /* Activate the relay the other direction,
                            they must be different to move the motor
  delay(15000);            // wait 2 seconds
  digitalWrite(up, HIGH);
  digitalWrite(down, HIGH); // Deactivate both relays to brake the mo
  delay(1000);              // wait 2 seconds
}
```

```
void unparkslot1()
{
  motor1_for(); // vertical down
   delay(1000);
  motor2_rev(); // horizontal forward
  delay(1000);
  verticalupP(); // vertical up for 1second
  delay(1000);
  motor2_for(); // horizontal backward
  delay(1000);
  motor1_rev(); // vertical Up for starting position
  verticalupP();
  delay(1000);
  motor2_rev(); // horizontal forward
  delay(1000);
  verticaldownP(); // vertical down for 1second delay
  delay(1000);
  motor2_for(); // horizontal backward
  delay(1000);
}
```

```
void verticaldownP()
{
  digitalWrite(up, LOW);
  digitalWrite(down, HIGH); /* Activate the relay one direction, they must
                             different to move the motor */
  delay(1000);              // wait 2 seconds
  digitalWrite(up, HIGH);
  digitalWrite(down, HIGH); // Deactivate both relays to brake the motor
  delay(1000);              // wait 2 seconds
}

void verticalupP()
{
  digitalWrite(up, HIGH);
  digitalWrite(down, LOW); /* Activate the relay one direction, they must
                            different to move the motor */
  delay(1000);             // wait 2 seconds
  digitalWrite(up, HIGH);
  digitalWrite(down, HIGH); // Deactivate both relays to brake the motor
  delay(1000);              // wait 2 seconds
}
```

Fig. 3.3 Motor Driving code

3.2.3 Self Parking System Prototype

The prototype of the self-parking system was made in consideration of the related work done in the field beforehand. The prototype works as per the functionalities discussed above, with some minor mechanical errors, which can be fixed by using adamant resources and tweaks in calculation for motor drivers.

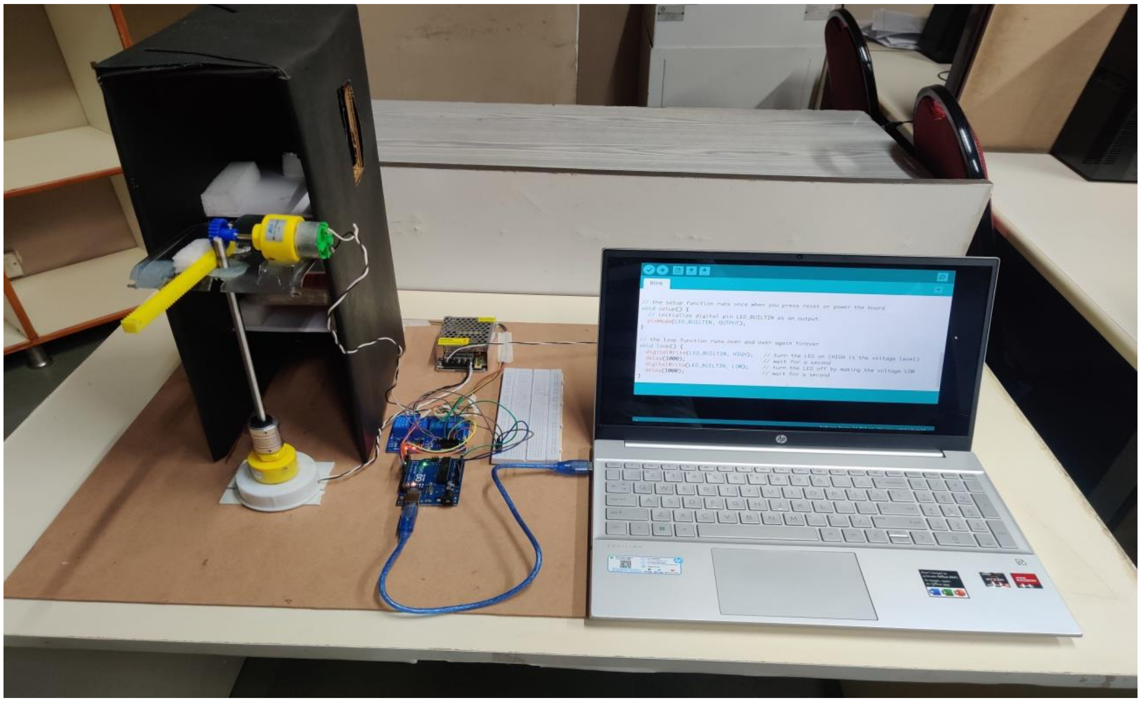

Fig. 3.4 Prototype

CHAPTER 4
LIMITATION AND ENHANCEMENT

4.1 Limitations

1. The structural integrity of the proposed prototype is lacking in some areas.
2. Continuous power supply to SMPS is required as the Arduino may start to execute the code but the motors won't activate, resulting in error.

4.2 Enhancement

1. A proper structure can be implemented using industrial-grade building materials.
2. The owner of the vehicle can get a precise location of where the vehicle is parked and the duration of parking.
3. Vehicle data can be made available to owners and relevant authorities by granting them real-time cloud access.
4. The un-parking can be totally automated using RFID technology.

CHAPTER 5
CONCLUSION

Based on the experiments that have been carried out, it could be concluded that testing carried out using Arduino IDE software was successful. This system provides the parking and un-parking of a car.

In this study, the various types of smart parking system and has been presented. From the various examples of the implementation of the smart parking system being presented, its efficiency in alleviating the traffic problem that arises especially in the city area where traffic congestion and the insufficient parking spaces are undeniable. It does so by directing patrons and optimizing the use of parking spaces. With the study on all the sensor technologies used in detecting vehicles, which are one of the most crucial parts of the smart parking system, the pros and cons of each sensor technologies can be analyzed. Although, there are certain disadvantages in the implementation of visual based system in vehicle detection as described earlier, the advantages far outweighs its disadvantages.

REFERENCES

[1] D. Shoup. Cruising for parking. Transport Policy, 13(6):479–486, 2006.

[2] IBM. IBM global parking survey: Drivers share worldwide parking woes, September 2011.

[3] B. Feeney. A review of the impact of parking policy measures on travel demand. Transportation Planning and Technology, 13(4):229–244, 1989.

[4] D. Schrank, B. Eisele, T. Lomax, and J. Bak. 2015 urban mobility scorecard. 2015.

[5] Katanga Executive Parking. How to use Katanga Valet Parking. Available from: https://www.valet.co.za/

[6] Mthokozisi, M., and Sa,L, S. Wireless Sensor Network Repairing, Honours project, University of Cape Town, 2008.

[7] Mannie, E., Generalized Multiprotocol Label Switching (GMPLS) Architecture, Request for Comments. Available from: https://www.ietf.org/rfc/rfc3945.txt , October (2004)

[8] Crowder, M., and Walton, M. Developing an Intelligent Parking System for the University of Texas at Austin Research Report SWUTC/03/167229-1 Available from: https://static.tti.tamu.edu/swutc.tamu.edu/publications/technicalreports/167229-1.pdf

[9] Rachid, G., and Sebastien, B. Integration of the Driving Philosophers into the Car Park Simulation (January 2005).

[10] Rosen, E.C, Viswanathan,A., and Callon, R. ``Multiprotocol Label Switching Architecture'', Request for Comments. Available from: https://www.ietf.org/rfc/rfc3031.txt (2001);.

[11] A. le Fauconnier and E. Gantelet. The time looking for a parking space: strategies, associated nuisances and stakes of parking management in France. In Proc. European Transport Conference (ETC), September 2006.

[12] Y. Geng and C. Cassandras. New smart parking system based on resource allocation and reservations. IEEE Transactions on Intelligent Transportation Systems, 14(3):1129–1139, September 2013.

[13] D. Shoup. The high cost of free parking. Journal of Planning Education and Research, 17(1):3–20, 1997.

[14] T. Monahan. “War rooms” of the street: Surveillance practices in transportation control centers. The Communication Review, 10(4):367–389, 2007.

[15] Vertically automated rotary parking system – IEEE Xplore Document by M Bhagdev 2017 https://ieeexplore.ieee.org/document/8256571

[16] https://www.researchgate.net/publication/336287016_Vertical_Car_Parking_System

[17] https://www.electrosal.com/product/automatic-vertical-car-parking-system

[18] https://selfparkingsystem.github.io/