VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELGAUM

MINI-Project Report On

Project Title

"ARDUINO AUTOMATED CAR PARKING SYSTEM"

Submitted in partial fulfillment of requirement for the Bachelor of Engineering
in
Electronics & Communication Engineering
By

MITHUN KUMAR JN	**USN:1VE18EC054**
PRAJWAL MS	**USN:1VE18EC070**
SUMANTH GOWDA V	**USN:1VE18EC092**
YOGESH J	**USN:1VE18EC103**

Under the Guidance of

ARJUN KUMAR G B
Assistant professor
Department of E&CE

Department of Electronics &Communication Engineering
Sri Venkateshwara College of Engineering
Vidyanagar, Bangalore-562157

2021

SRI VENKATESHWARA COLLEGE OF ENGINEERING

(Affiliated by A.I.C.T.E and approved by V.T.U)

CERTIFICATE

This is to certify that the Mini Project work prescribed by the Visvesvaraya Technological University, Belagavi entitled **"ARDUINO AUTOMATED CAR PARKING SYSTEM" was carried out by MITHUN KUMAR J N (1VE18EC054), PRAJWAL M S(1VE18EC070), SUMANTH GOWDA V(1VE18EC092), YOGESH J(1VE18EC103)** are Bonafede students of 6th Semester, Electronics & Communication Engineering, Sri Venkateshwara College of Engineering. This is in partial fulfilment for the award of Bachelor of Engineering in VTU, Belagavi during academic year 2021.

It is certified that all corrections/suggestions indicated for mini project internal assessment have been incorporated in the report. The mini project report has been approved as it satisfies the academic requirements in the respect of mini project work prescribed for the said degree.

Signature of the Guide	Name & Signature of the HOD	Signature of the Principal
Arjun Kumar G B	**Dr. Shivashankar**	**Dr.Nageshwar Guptha.M**
Assistance Professor	Professor & Head of Dept	Principal, SVCE
Dept. E&CE, SVCE,	Dept. E&CE, SVCE,	Bengaluru.
Bengaluru.	Bengaluru.	

TABLE OF CONTENTS

ABSTRACT

The aim of the smart car parking system is to reduce the human effort, traffic and to automate the car parking areas. Here we discuss a project which presents a miniature model of an automated car parking system that can regulate and manage the number of cars that can be parked in a given space at any given time based on the availability of parking spot. Automated parking is a method of parking and exiting cars using sensing devices. The entering to or leaving from the parking lot is commanded by an Android based application. We have studied some of the existing systems and it shows that most of the existing systems and it shows that most of the existing systems aren't completely automated and require a certain level of human interference or interaction in or with the system. The difference between our system and the other existing systems is that we aim to make our system is that we aim to make our system as less human dependent as possible by automating the cars as well as the entire parking lot, on the other hand most existing systems require human personnel (or the car owner) to park the car themselves.

Chapter 1

INTRODUCTION

1.1 OVERVIEW

Drivers searching for parking are estimated to be responsible for about 30% of traffic congestion in cities. Historically, cities, businesses, and property developers have tried to match parking supply to growing demand for parking spaces. It has become clear, though, that simply creating more parking spaces is not sufficient to address the problem of congestion. New approaches using smart parking systems look to provide a more balanced view of parking that better manages the relationship between supply and demand. Figure 1.1 shows the car parking system.

Figure 1.1: Car parking system

Smart parking can be defined as the use of advanced technologies for the efficient operation, monitoring, and management of parking within an urban mobility strategy. The global market for smart parking systems reached $93.5 million, with the United States representing 46% market share, and offering a strong growth opportunity for companies offering services in the United States and overseas. A number of technologies provide the basis for smart parking solutions, including vehicle sensors, wireless communications, and data analytics. Smart parking is also made viable by innovation in areas such as smartphone apps for customer services, mobile payments, and in-car navigation systems. At the heart of the smart parking concept is the ability to access, collect, analyse, disseminate, and act on information on parking usage. Increasingly, this information is provided in real-time from intelligent devices that enable both parking managers and drivers to optimize the use of parking capacity.

1.2 PROBLEM STATEMENT

With increase in the population, number of vehicles increases and due to unmanaged parking, it leads to many problems. In centre cities, people face difficulties as increasing number of vehicles creates congestion, wastage of space, wastage of time, traffic problems, car napping, car vandalism and many other difficulties. Smart parking system can reduce these difficulties and help to find the parking area wisely.

1.3 OBJECTIVES

The objectives of the project are as follows: -

- To Design and build up a prototype of an automated car park system.
- To learn how to control the prototype system for automated parking.
- To acknowledge how to program Arduino and make it works on any system.
- To Guide the driver to find an parking space more convenient and efficient using IOT.

1.4 CHAPTER SUMMARY

This work introduces about the smart parking system and brief about the project. Here we have discussed the problem statement and the objective of the project which we are going to work through this project.

Chapter 2

LITERATURE SURVEY

Smart cities use infrastructure, innovation, and technology for adapting the changes brought about by the overpopulation of cities and thus contribute to the reduction of carbon dioxide emissions, decreasing energy consumption, and promoting economic, social, and environmental development. One of the first steps to be implemented to upgrade the traditional cities to become smart cities is smart parking. Many cities around the world have already started implementations of smart parking projects, making life easier. Intelligent parking helps drivers to efficiently and effectively searches for parking spaces through information and communication technology.[1]

The Smart Parking System is designed by making use of some IOT supportable hardware's such as raspberry pi, aurdino boards etc. here we focusing on less power consumption and more performance device so raspberry pi is the suitable microcontroller for our implementation. And NOOBS installer is loaded into the storage device of microcontroller. This installer which consists of various hardware supportable operating systems such as mac os, tiny os, open Elec, Raspbian os etc. where these operating systems which basically consumes less power.[2]

Smart parking using Iot technology helps to designs and develops a real smart parking system which provides information for vacant spaces and also helps the user to locate the nearest availability. This paper uses a computer vision to detect vehicle number plate in order to enhance the security. The user can pay for the parking space prior to the entry of the car through mobile payment. Thus, insuring the reservation of the parking. The user is notified about the parking location, number of slots available and all other relevant information. The paper uses efficient algorithms and techniques for extracting license plate text. An algorithm operates on the ultrasonic sensor detection of the vehicle entering into the parking slot and calculates the minimum cost for the user.[3]

Parking is a service that is quite dated in the transport industry and it is thought to have evolved specifically for different generations. The initial parking system, in which there were not many vehicles, was articulated through the annual space renting model. However, with time, and as the number of vehicles in cities and towns continued to increase, just as the number of urban dwellers, the need for urban planning became apparent.

With this, one area that required special attention, in order to ensure that vehicles entering urban areas do not cause traffic congestions as well as increase the harmful impacts they have on the environment, was urban parking. For this reason, the concept of controlled parking was borne. First, before the emergence of the smart parking system, cities relied on electronic parking services that involved the use of parking meters that were not wholly automated. This created loop-holes in the collection of fees and the auditing process. However, with technological advancement, there has been a notable terminal evolution, which has seen the emergence of smart parking systems. The present smart parking system provides the automation of different parking services, allowing consumers to navigate the entire parking experience independently—from the parking occupancy status to ticketing, parking, and fee settlement.[4]

2.1 CHAPTER SUMMARY

By the above study of the existing work, we are able to use various technologies in each work, combine them with the parking system and overcome all the drawbacks as seen in this extensive study. Hence, we would like to conclude by saying that by doing the extensive literature survey we have gained enormous knowledge on already existing research.

Chapter 3

METHODOLOGY

3.1 IMPLEMENTATION

The smart parking system consist of Arduino microcontroller, infrared sensors, servo motor and LCD display as the main components. Arduino is use as the main microcontroller for the project. The microcontroller will be program as a counter use to count the number of cars entering and exit an indoor parking space. This system uses infrared sensors to detect cars that enter or leave the parking space. Besides that, there are two automatic gates at the entrance and at the exit way that will be controlled by the Arduino microcontroller. The servo motor act as a gate, when the vehicle appears near the Entry gate that is Ir sensor 1 the sensor detects the car and inform the Arduino to open the gate. Here Ir sensor detect the car as well as count the car entered and exit the parking area so that it can be easy to calculate the remaining parking area. This can reduce the man resource and automate the car parking system. For more advance we can use Node Mcu and connect it to some user interface so that the driver can get to know the parking slot wisely using the App and can park his car safely, this reduces the traffic in the city. Here we have implemented using both hardware and software simulation too. Figure 3.1 shows the implementation of car parking system using Arduino uno and Ir sensors along with lcd to display the status.

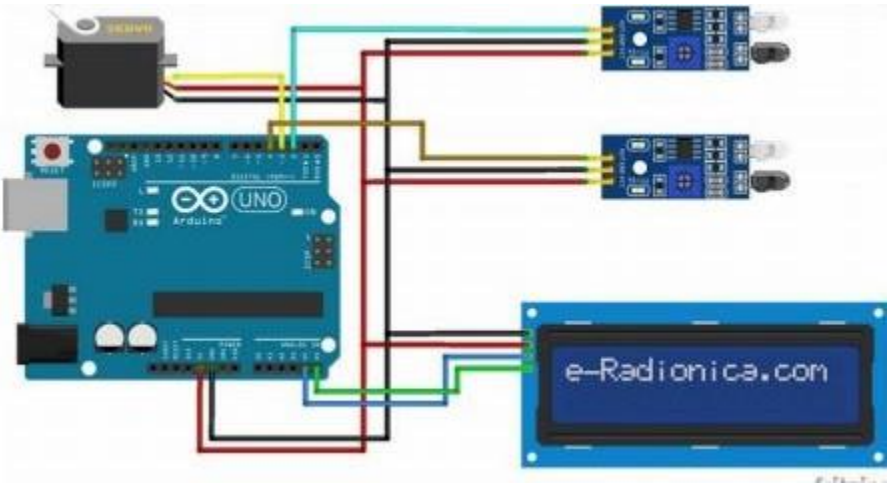

Figure 3.1: Implementation of car parking system using Arduino

3.2 BLOCK DIAGRAM

Arduino Automated Smart car parking system consists of Arduino Uno, Proximity sensor, Servo Motor, Potentiometer and LCD display. Arduino Uno controllers the activity, it takes the input from the sensor and make gate over and close and finally make the output display on the LCD display. The working is as follows according to the block diagram show in figure 3.2.

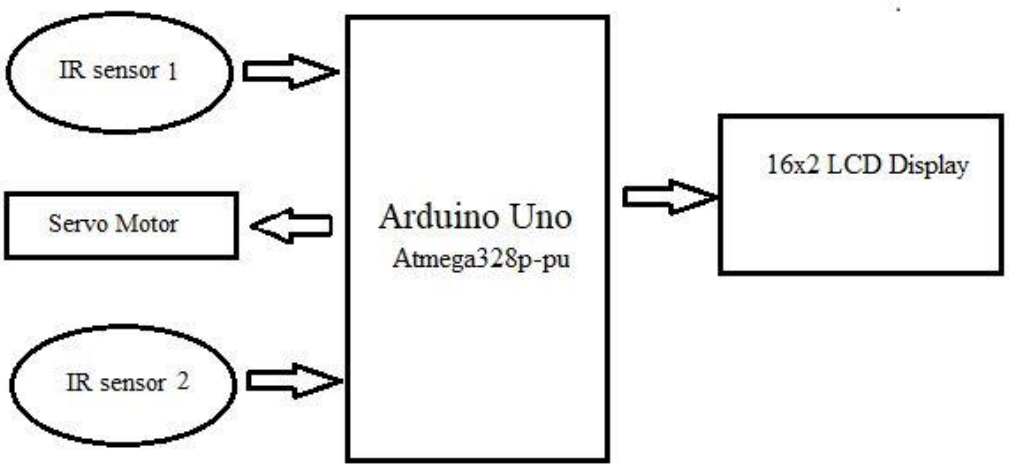

Figure 3.2: Block Diagram

- The proximity sensors such as IR sensor 1 and IR sensor two sensors which detect the car, when the car approaches in its range.
- This sensed massage will be transferred to Arduino Uno, where Arduino Uno count the car and it will compare with the parking slot and then intrust the servo motor to open the gate.
- If the parking slot is available, the gate will open and if the places are fill it will display the massage “Parking full” through LCD display.
- LCD display (16x2) acts as a screen to display the message that is given by the Arduino Uno.

3.3 FLOWCHART

STEPS INVOLVED IN THE PROCESS

The steps involved in the Arduino Automated car parking system is as follow. The projects execution exactly follows this step.

Step-1: The system will TURND ON.

Step-2: Initialize the Arduino Uno I/O pins, LCD display, IR sensor and Servo motor.

Step-3: IR sensors near the Gate that is servo motor detect the vehicle which come in its range.

Step-4: This sensed information will be sent to Uno board and in turn Arduino Uno board will sends the information to servo motor to open the.

Step-5: If the vehicles come in, it will be considered as one car and one parking slot is filled.

Step-6: This parking information that is available slot and number of car parked information will displayed on the LCD screen.

This working flow is shown in the figure 3.3.

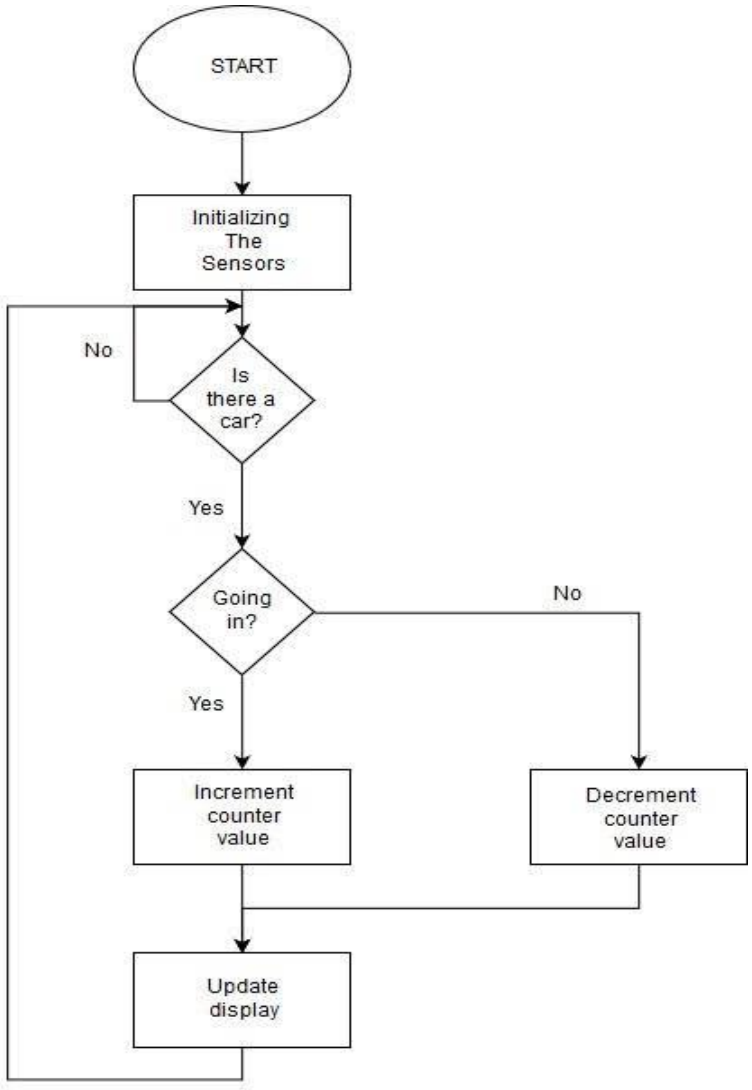

Figure 3.3: Flowchart of Car parking system

3.4 CHAPTER SUMMARY

This chapter deals with key interesting internal block diagram of car parking system. The chapter explains about the connections between every component and required system together for better adjustments of the system to function efficiently and successfully. Here we get to know the Implementation of the project and it circuit diagram.

Chapter 4

HARDWARE AND SOFTWARE COMPONENTS

4.1 HARDWARE COMPONENTS

The implementation of the car parking system requires few hardware components. The hardware components are as follows:

- Arduino Uno
- IR sensor
- LCD display
- Servo motor
- Potentiometer
- Resistor

4.1.1 ARDUINO UNO:

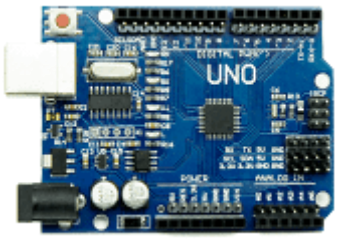

Figure 4.1: Arduino Uno

Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. Arduino is an open-source electronics platform based on easy-to-use hardware and software. It is shown in the figure 4.1 and the pin diagram is shown is figure 4.2.

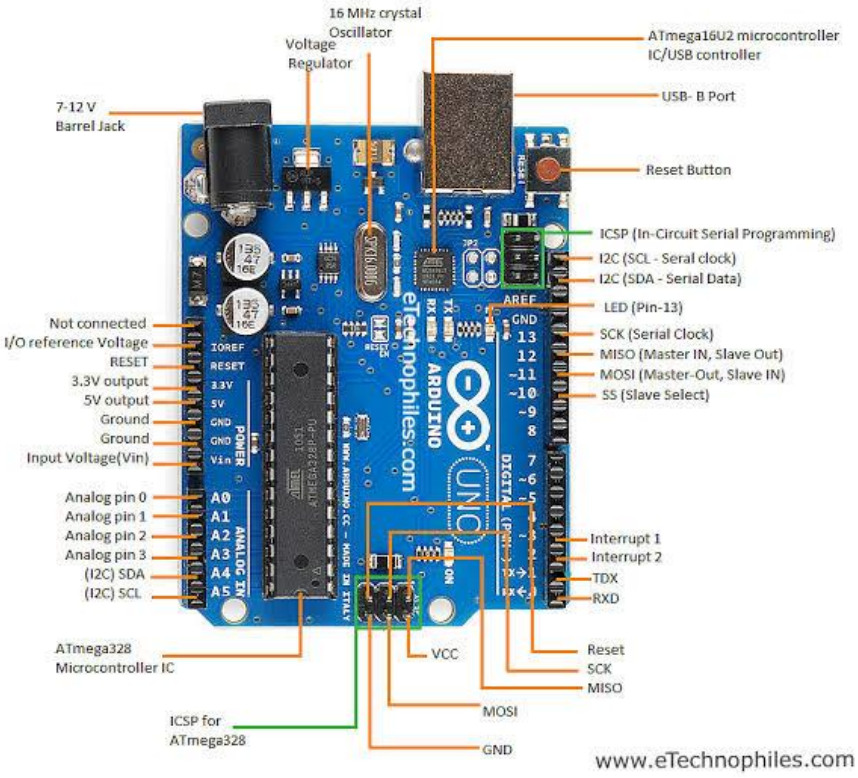

Figure 4.2: Pin diagram of Arduino Uno

4.1.2 IR SENSOR:

Figure 4.3: IR Sensor

A passive infrared sensor is an electronic sensor that measures infrared light radiating from objects in its field of measures infrared light radiating from objects in its field of view. An IR sensor is an electronic device, that emits in order to sense some aspects of the surrounding. An IR sensor can measure the heat of an object as detects the motion. The IR sensor is shown in the figure 4.3.

4.1.3 LCD DISPLAY:

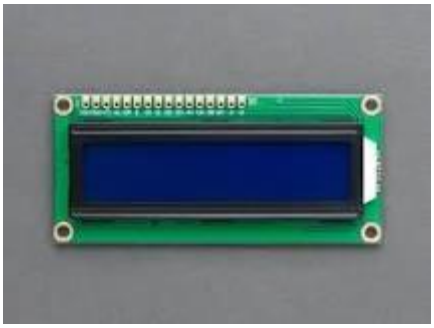

Figure 4.4: LCD Display

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. The 16 x 2 intelligent alphanumeric dot matrix display is capable of displaying 224 different characters and symbols. A 16x2 LCD means it can display 16characters per line and here are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. The 16x2 intelligent alphanumeric dot matrix display is capable of displaying 224 different characters and symbols. This LCD has two registers, namely, Command and Data. The LCD Display is shown in the figure 4.4.

4.1.4 SERVO MOTOR:

Figure 4.5: Servo motor

A servo motor is an electromechanical device that produces torque and velocity based on the supplied current and voltage. A servo motor works as part of a closed loop system providing torque and velocity as commanded from a servo controller utilizing a feedback device to close the loop. The Servo motor is shown in the figure 4.5.

4.1.5 POTENTIOMETER:

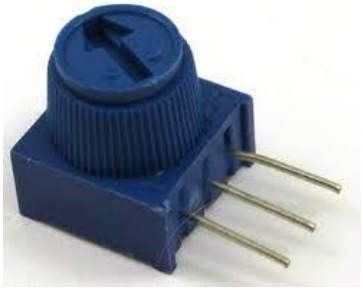

Figure 4.6: Potentiometer

A potentiometer is a manually adjustable variable resistor with 3 terminals. Two terminals are connected to both ends of a resistive element, and the third terminal connects to a sliding contact, called a wiper, moving over the resistive element. The position of the wiper determines the output voltage of the potentiometer. It is shown in the figure 4.6.

4.1.6 RESISTOR:

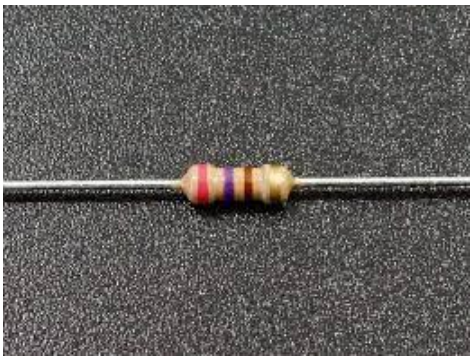

Figure 4.7: Resistor

A resistor is a passive two-terminal electrical component that implements electrical resistance as a circuit element. It is shown in the figure 4.7.

4.1.7 Other Components:

Other components such as Jumper wire and cable. Jumper wire is used to connect the one component to another components and the Cable is requires to dump the code into Arduino Uno board.

4.2 SOFTWARE COMPONENTS

The implementation of the car parking system requires few software components to compile the code and simulate the project. Here we use the open-source platform to compile and simulate the project, such as:

- Arduino IDE
- Tinker Cad

4.2.1 Arduino IDE:

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++. It is used to write and upload programs to Arduino compatible boards, but also, with the help of third-party cores, other vendor development boards. The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main () into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program argued to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware. By default, avrdude is used as the uploading tool to flash the user code onto official Arduino boards.

The Arduino Integrated Development Environment – or Arduino Software (IDE) 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. It connects to the Arduino and Genuino hardware to upload programs and communicate with them. Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension .ino. The editor has feature for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also display errors. The console display text output by the Arduino Software (IDE), including complete error messages and other information. The toolbar buttons allow you to verify and upload programs, create, open and save sketches and open the serial monitor. Figure 4.8 shows the home page of Arduino IDE platform.

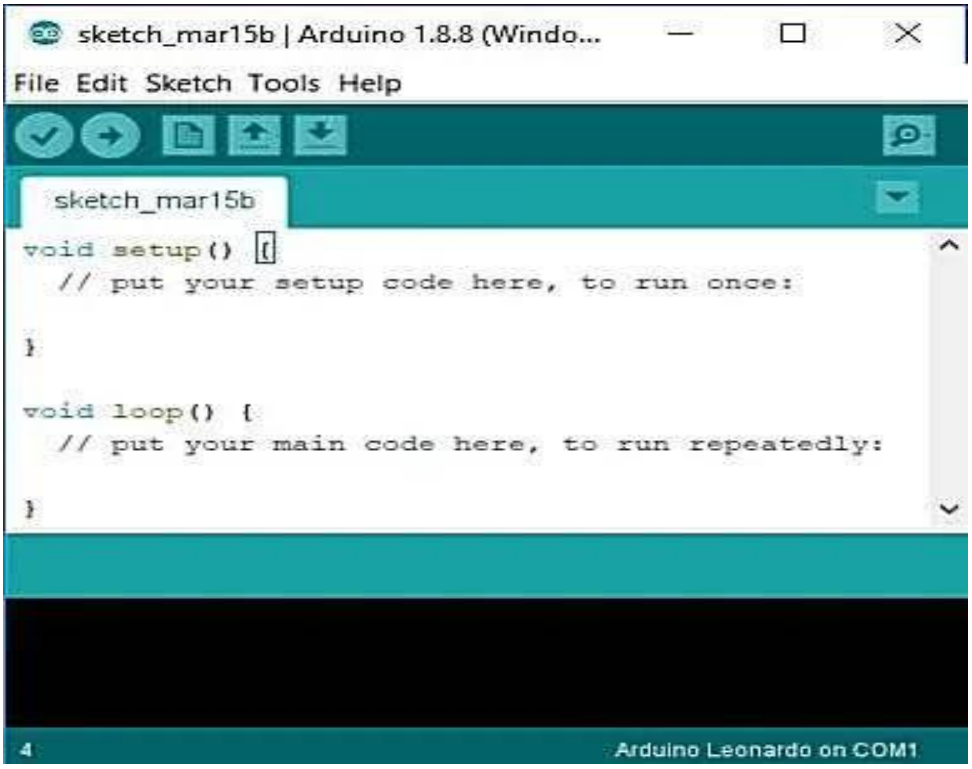

Figure 4.8: Arduino IDE

4.2.2 TINKER CAD:

Tinker cad is a free-of-charge, online 3D modelling program that runs in a web browser. Since it became available in 2011 it has become a popular platform for creating models for 3D printing as well as an entry-level introduction to constructive solid geometry in schools. Tinker cad uses a simplified constructive solid geometry method of constructing models. A design is made up of primitive shapes that are either "solid" or "hole". Combining solids and holes together, new shapes can be created, which in turn can be assigned the property of solid or hole. Combining solids and holes together, new shapes can be created, which in turn can be assigned the property of solid or hole. In addition to the standard library of primitive shapes, a user can create custom shape generators using a built-in JavaScript editor.

Shapes can be imported in three formats: STL and OBJ for 3D, and 2-dimensional SVG shapes for extruding into 3D shapes. Tinker cad exports models in STL or OBJ formats, ready for 3D printing. Tinker cad also includes a feature to export 3D models to Minecraft Java Edition, and also offers the ability to design structures using Lego bricks. Figure 4.9 shows the tinker cad home page.

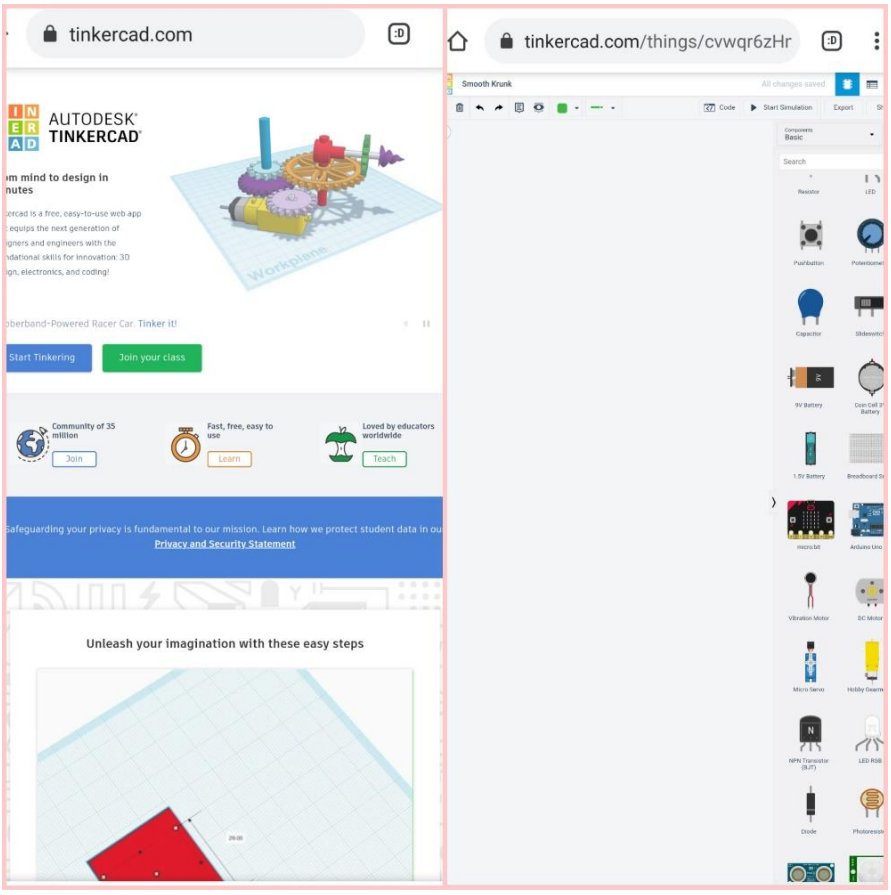

Figure 4.9: Tinker cad

4.3 CHAPTER SUMMARY

The various hardware and software components used for the system are discussed in brief with respect to their specifications and working in this chapter. The different components play an important role in recognizing and sensing the Vehicles and integrating into a complete system to provide a greater number of features.

Chapter 5

RESULTS AND DISCUSSION

We have analysed the working of other, and done extensive literature survey, to understand the basic principles of already existing Arduino Automated Car Parking System. Based on this analysis we were able to detect the faults, or drawbacks in the models built before. With all this knowledge we came up with a perfect idea to overcome few of the faults or defects we had seen in earlier.

5.1 Hardware Output:

Arduino Automated Car Parking System which is also called as smart car parking system since it reduces the human power and parking system is done automatically. Here we have used IR sensors near the gate, and it detect the car and opens the gate as well as it counts the number car entered and exited. Here the Arduino Uno as the total number of parking slots and through this sensor it takes the car entries and display the parking details. We have complied the code using Arduino IDE and uploaded to the Arduino Uno to perform the project objective. Below are the images of Project model (figure 5.3, figure 5.4 and figure 5.5) and Complied code images (figure 5.1 and figure 5.2).

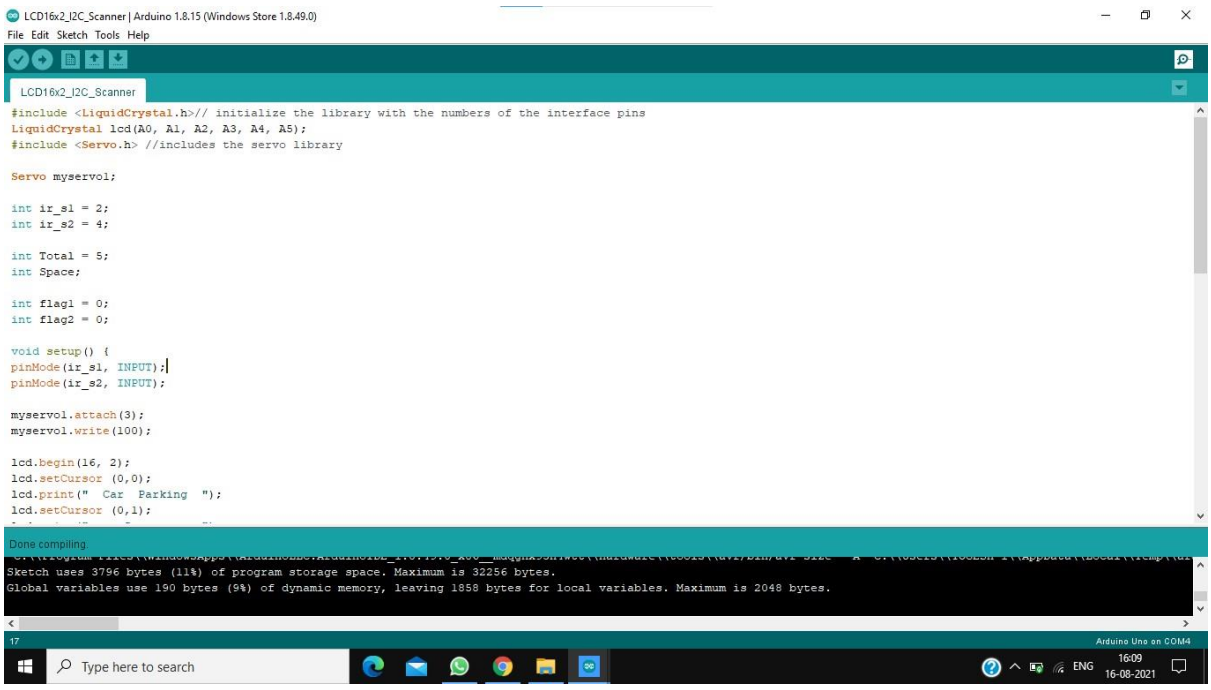

Figure *5.1:* Code compilation using Arduino IDE

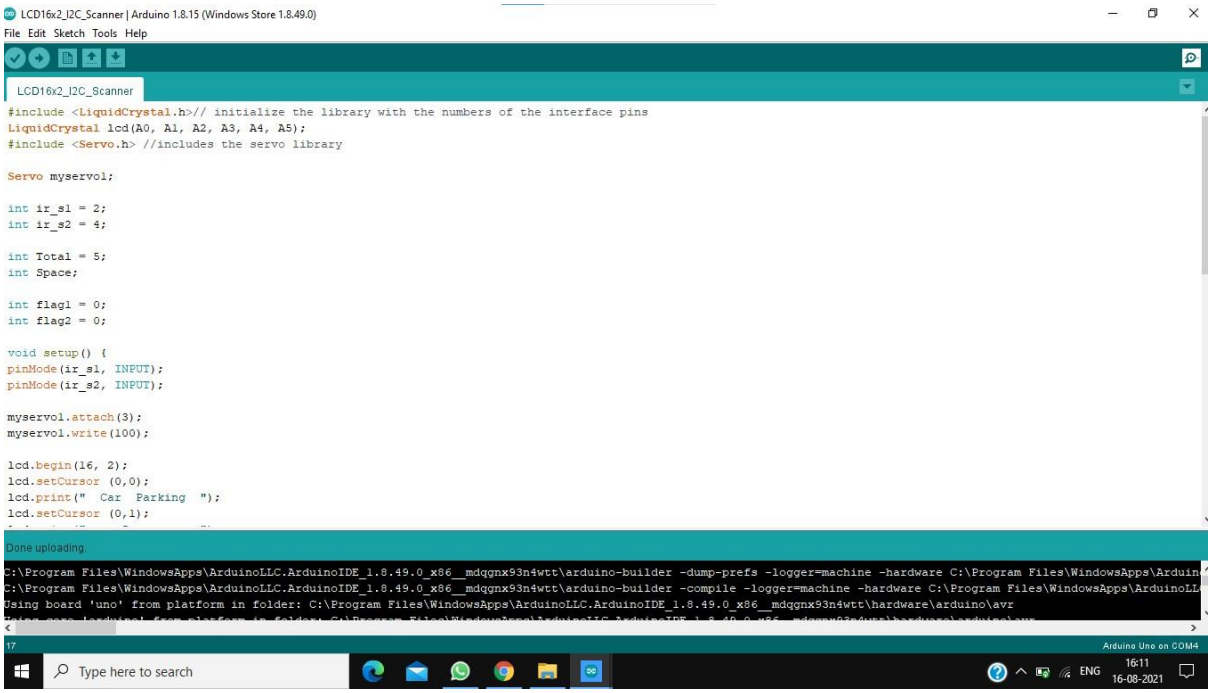

Figure *5.2:* Code uploading using Arduino IDE

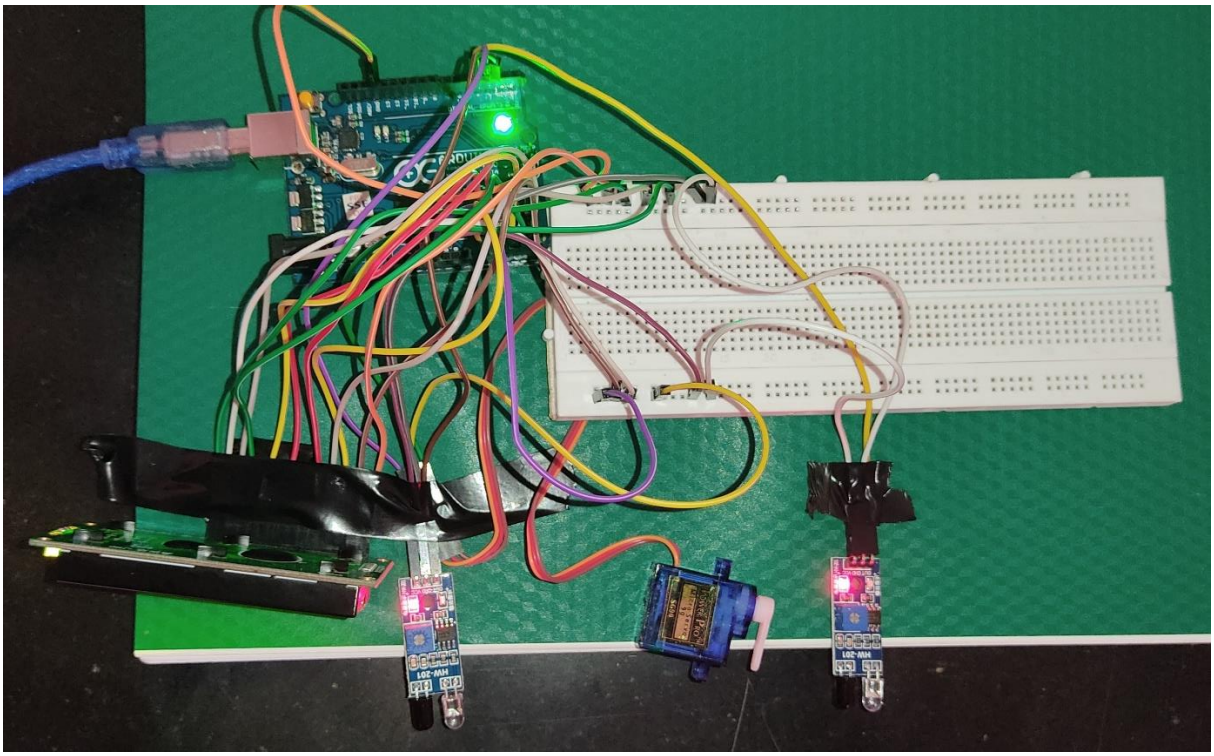

Figure 5.3: Top view of the model

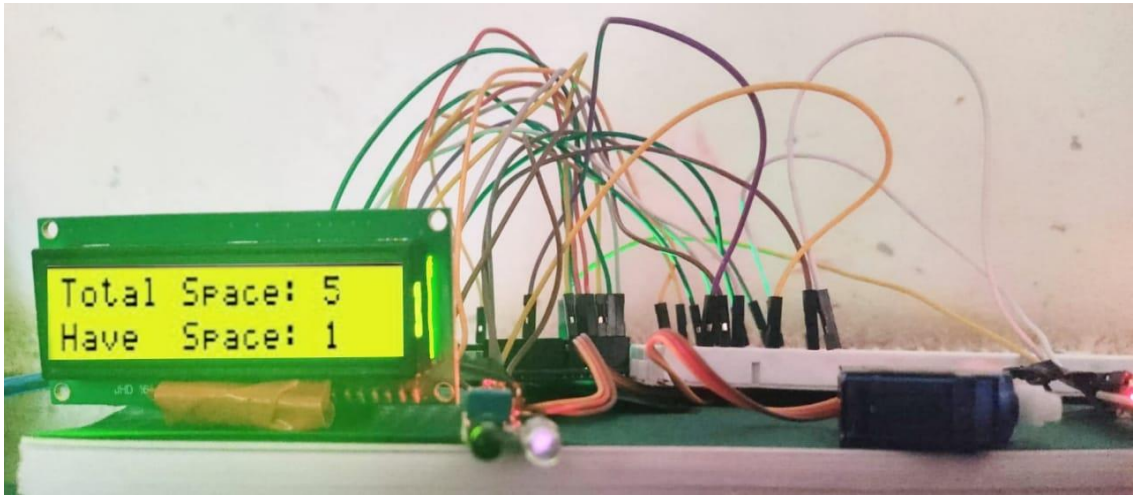

Figure 5.4: Output of the project

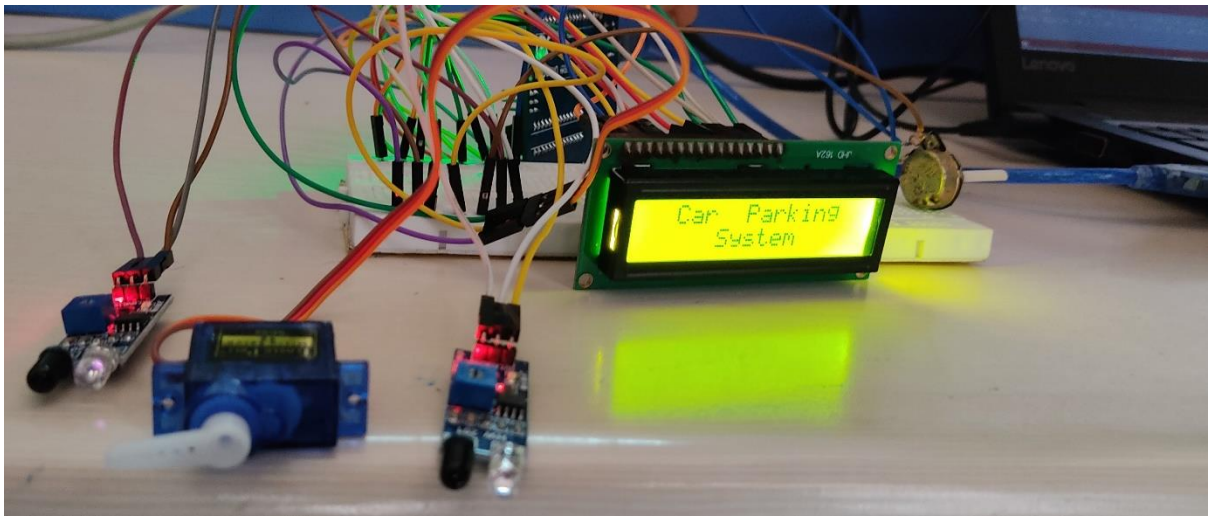

Figure 5.5: Output of the project

CODE:

```
#include<LiquidCrystal.h>
LiquidCrystal lcd(A0, A1, A2, A3, A4, A5);
#include <Servo.h>
Servo myservo1;
int ir_s1 = 2;
int ir_s2 = 4;
int Total = 5;
int Space;
int flag1 = 0;
```

```
int flag2 = 0;
void setup() {
pinMode(ir_s1, INPUT);
pinMode(ir_s2, INPUT);
myservo1.attach(3);
myservo1.write(100);
lcd.begin(16, 2);
lcd.setCursor (0,0);
lcd.print("  Car  Parking  ");
lcd.setCursor (0,1);
lcd.print("     System     ");
delay (2000);
lcd.clear();
Space = Total;
}
void loop(){
if(digitalRead (ir_s1) == LOW && flag1==0){
if(Space>0){flag1=1;
if(flag2==0){

myservo1.write(0);

 Space = Space-1;

}
}else{
lcd.setCursor (0,0);
lcd.print(" Sorry not Space ");
lcd.setCursor (0,1);
lcd.print("    Available    ");
delay (1000);
lcd.clear();
}
}
if(digitalRead (ir_s2) == LOW && flag2==0){flag2=1;
```

```
if(flag1==0){myservo1.write(0); Space = Space+1;}
}
if(flag1==1 && flag2==1){
delay (1000);
myservo1.write(100);
flag1=0, flag2=0;
}
lcd.setCursor (0,0);
lcd.print("Total Space: ");
lcd.print(Total);
lcd.setCursor (0,1);
lcd.print("Have  Space: ");
lcd.print(Space);
}
```

5.2 Software Simulation Output:

Arduino Automated Car Parking System using Tinker cad online simulation tool. Here we have used the Arduino Uno, Lcd display, ultra-sonic sensor, Potentiometer, and servo motor. This software simulation output is similar to hardware output too.

Tinker cad is a free, easy-to-use web app that equips the next generation of designers and engineers with the foundational skills for innovation. Working is as follow: first the ultra-sonic sensor near the gate that is servo motor, sense the car and instruct the Arduino Uno to open the gate. Next the three ultra-sonic sensors are kept near the parking slot, when the car comes in the range of the sensor, ultra-sonic sensor detect the car and it reduce the one parking slot. When car leaves the slots, it updates the parking slot again. This information is shown in the LCD display. Figure 5.5 shows the circuit of the Arduino automated car parking system using tinker cad software. Figure 5.6, 5.7, 5.8 and 5.9 shows the working and output of the Automated car parking system using tinker cad.

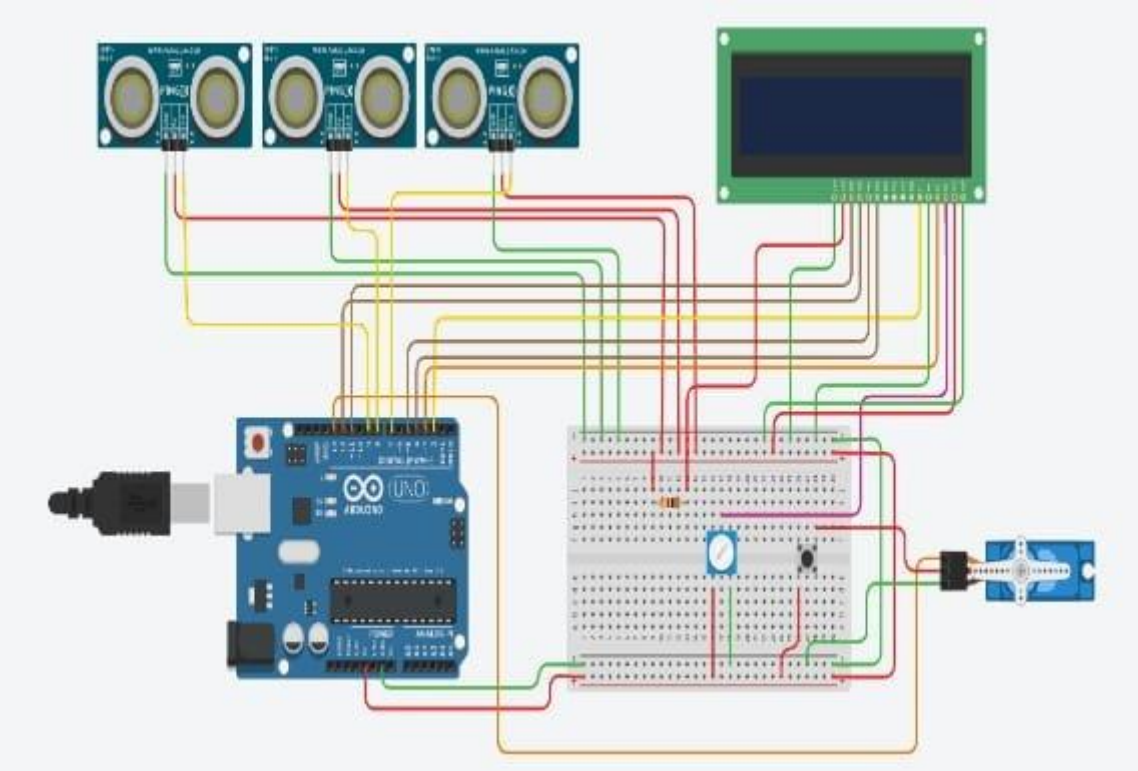

Figure 5.5: Circuit Diagram

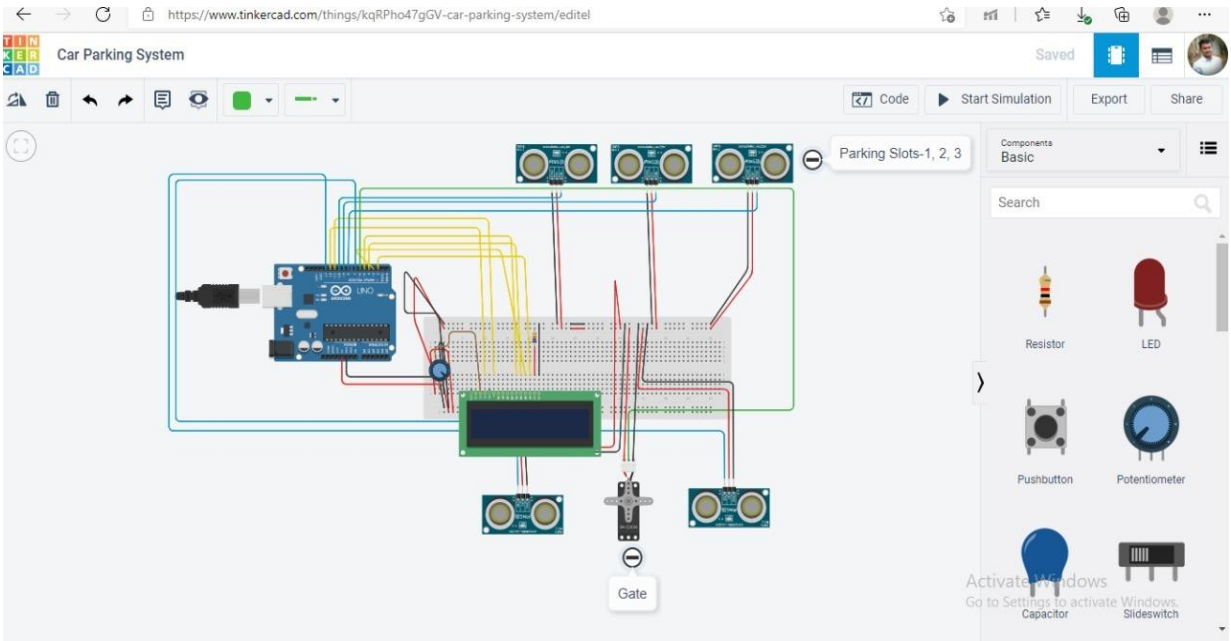

Figure 5.6: Working using tinker cad

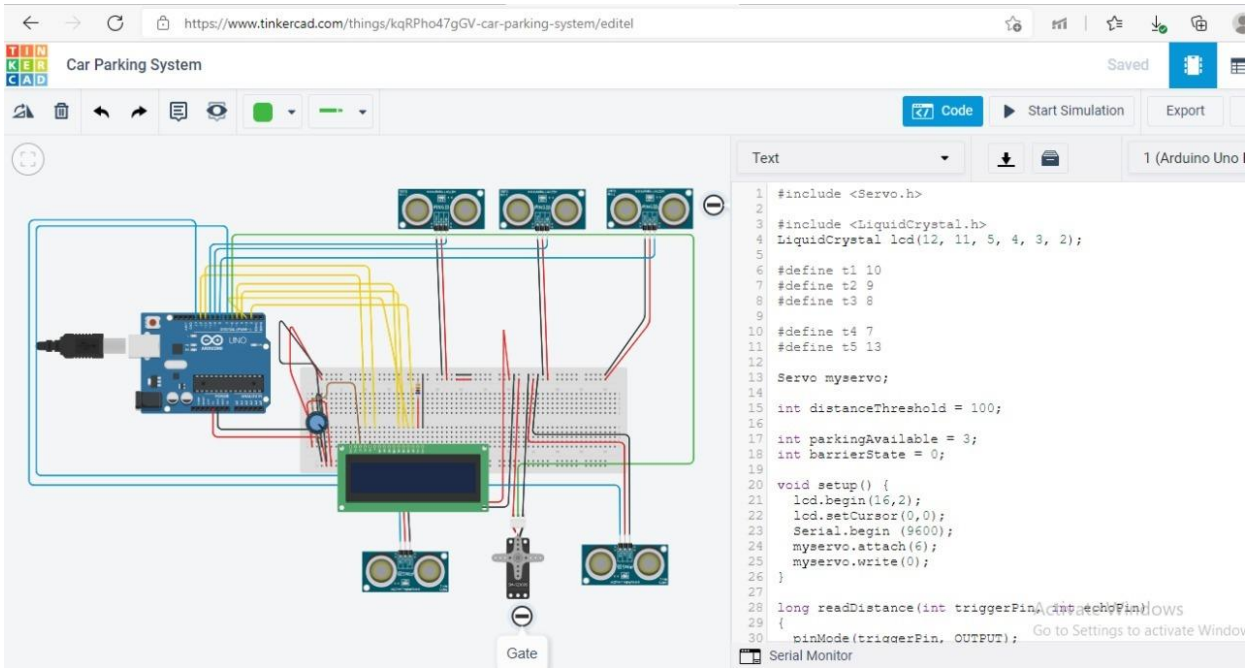

Figure 5.7: Code compilation using tinker cad

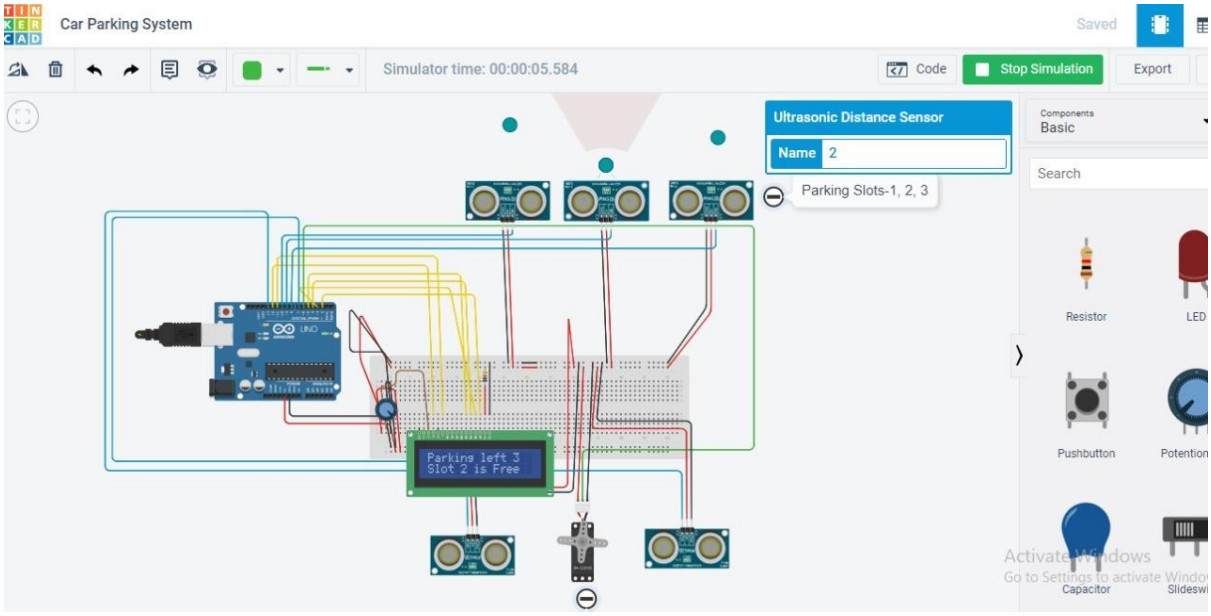

Figure 5.8: Working output

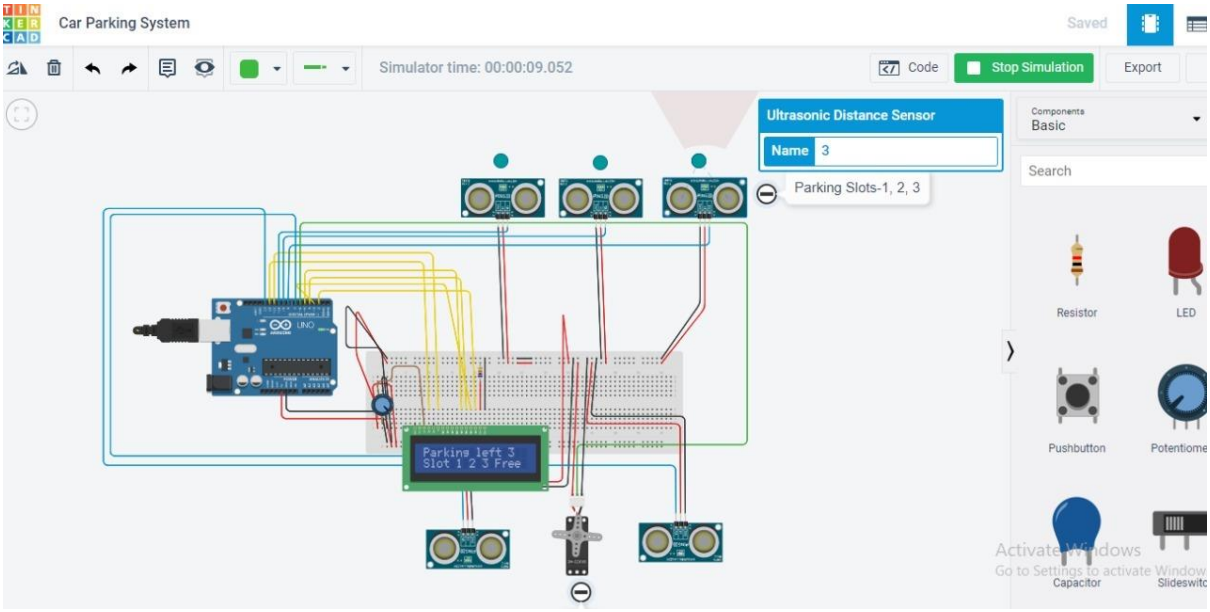

Figure 5.9: Working output

CODE:

```
#include <Servo.h>

#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

#define t1 10

#define t2 9

#define t3 8

#define t4 7

#define t5 13

Servo myservo;

int distanceThreshold = 100;

int parkingAvailable = 3;

int barrierState = 0;

void setup() {

lcd.begin(16,2);
```

```
lcd.setCursor(0,0);
Serial.begin (9600);
myservo.attach(6);
myservo.write(0);
}
long readDistance(int triggerPin, int echoPin)
{
pinMode(triggerPin, OUTPUT);
digitalWrite(triggerPin, LOW);
delayMicroseconds(2);
digitalWrite(triggerPin, HIGH);
delayMicroseconds(10);
digitalWrite(triggerPin, LOW);
pinMode(echoPin, INPUT);
return pulseIn(echoPin, HIGH);
}
void loop()
{
float d1 = 0.01723 * readDistance(t1, t1);
float d2 = 0.01723 * readDistance(t2, t2);
float d3 = 0.01723 * readDistance(t3, t3);
float d4 = 0.01723 * readDistance(t4, t4);
float d5 = 0.01723 * readDistance(t5, t5);
Serial.println("d1 = " + String(d1) + "cm");
```

```
Serial.println("d2 = " + String(d2) + "cm");
Serial.println("d3 = " + String(d3) + "cm");
Serial.println("d4 = " + String(d4) + "cm");
Serial.println("d5 = " + String(d5) + "cm");
if(barrierState == 0)
{
if(d4<100 && parkingAvailable>0)
{
parkingAvailable -= 1;
barrierState = -1;
myservo.write(90);
}
if(d5<100 && parkingAvailable<3)
{
parkingAvailable += 1;
barrierState = 1;
myservo.write(90);
}
}
else if(barrierState == -1)
{
if(d4>=100 && d5<100)
{
barrierState = -2;
```

```
myservo.write(0);
}
}
else if(barrierState == 1)
{
if(d5>=100 && d4<100)
{
barrierState = 2;
myservo.write(0);
}
}
else if(barrierState == -2)
{
if(d5>=100)
{
barrierState = 0;
}
}
else if(barrierState == 2)
{
if(d4>=100)
{
barrierState = 0;
}
```

```
}
lcd.setCursor(0,0);
if(parkingAvailable == 0)
{
lcd.print("Parking Full  ");
}
else
{
lcd.print("Parking left ");
lcd.print(parkingAvailable);
}
if (d1>100 & d2>100 & d3>100){
lcd.setCursor(0,1);
lcd.print("Slot 1 2 3 Free");
delay(500);
}
else if((d1>100 & d2>100)|(d2>100 & d3>100)|(d3>100 & d1>100))
{
lcd.setCursor(0,1);
if(d1>100 & d2>100)
lcd.print("Slot 1 & 2 Free");
else if(d1>100 & d3>100)
lcd.print("Slot 1 & 3 Free");
else
```

```
lcd.print("Slot 2 & 3 Free");
delay(500);
}
else if(d1<100 & d2<100 & d3<100)
{
lcd.setCursor(0,1);
lcd.print("Parking Full   ");
delay(500);
}
else if((d1<100 & d2<100)|(d2<100 & d3<100)|(d3<100 & d1<100))
{
lcd.setCursor(0,1);
if(d1>100)
lcd.print("Slot 1 is Free ");
else if (d2>100)
lcd.print("Slot 2 is Free ");
else
lcd.print("Slot 3 is Free ");
delay(500);
}
delay(100);
}
```

5.3 CHAPTER SUMMARY

This chapter tells us about all the results that have been obtained in the project and also tells us about the complete functioning of the model. The results obtained shows that the model can help the car drivers to look for the car parking and easily park the car which in turn reduces the traffic in the city.

Chapter 6

ADVANTAGES AND APPLICATIONS

6.1 ADVANTAGES

- With this technology advanced security features can be implemented.
- It Enhanced User Experience.
- It Reduce traffic and pollution.
- It increases safety of vehicles.
- This technology decreases management costs.

6.2 APPLICATION

We see that, this new system is an added value to the urban life style. The smart parking reservation system can be implemented in shopping malls, restaurants, theatres etc.

6.3 CHAPTER SUMMARY

The Future use of the smart parking system, Advantages of using the smart parking system and some of the application has been discussed in this chapter.

CONCLUSION

Automation is a step in the right direction for a future fulfilled in the world of transportation. This design provides an effective solution to the common problem discussed. The smart car parking system was designed, fabricated and tested which provided accurate results when the threshold distance was calibrated and the obstruction was detected. The switching of LEDs Display based on the vehicle in the parking space was instantaneous based on no vehicle and vehicle detected. The design is flexible and can be altered based on the space available and can be installed even in tight and constrained space. Based on the number of slots detected a common information board is displayed indicating the count of parking spaces available. It can be concluded that with correct connection of some simple electrical components, it is possible to create an automatic smart car parking system, thus decreasing aimless driving, fuel and time, as well as making the process of parking considerably simpler.

The smart parking system based on IOT concept has been implemented using various sensor circuitry and cloud. It is an efficient system for car parking which prevails traffic congestion. This work is further extended as smart car parking system with automatic entry and exits system also fully automated system using multilayer parking methods. We see that, this new system is an added value to the urban life style. In the current world, where Google cars are introduced and automated driving is growing interest among the people, this will be an improvement of great magnitude. Automated cars can do the parking on their own, if this car parking app is a part of their built-in system. Hence, I believe that these simple concepts, when released into the world, will be one of those revolutionary changes in everyday activities.

FUTURE SCOPE

In future works, this framework can be enhanced by including different applications. For Example, internet booking by utilizing GSM. The driver or client can book their parking area at home or while in transit to the shopping centre. This can diminish the season of the client to seeking the empty frameworks can be added to enhance this framework to distinguish the question and guide the driver or clients speediest. We will attempt to decrease the mechanical structure and attempt to make it eco-friendly.

REFERENCES

- Luis F. Luques-vega and David A. Michel-Torres, "IOT smart parking system", International Journal of Environmental Research and publication, Volume-20, Issue 5, https://doi.org/10.3390/s20051476, February 2020.
- Mr. Basavaraj S R, "Automatic smart parking system using Internet of Things", International Journal of scientific and Research publication, Volume-12, 629-632, December 2015.
- Elakya R, Juhi Seth, Pola Ashritha, R Namith, "Smart Parking System using IOT", International journal of Engineering and Advanced Technology (IJEAT), ISSN:2249-8958, Volume-9, Issue-1, DOI:10.35940/ijeat.A1963.109119, October 2019.
- Can Biyik, Gabriele Pieri and Zaheer Allam, "Smart Parking System", MDPI journal, Volume-4, Issue-1,623–642. https://doi.org/10.3390, January 2021.
- Aashish Joshi, Arni Tharakaram Hariram, K M Vishall Somaiya, Mubashir Hussain, "Smart Car Parking System", IJERT Publication, Volume-09, Issue-09, https://dx.doi.org/10.17577//IJERTV0IS090305, September-2020.
- Suvarna Nandyal, Sabiya Sultana, Sadaf Anjum, "Smart Car Parking System using Arduino UNO", International Journal of Computer Application, Volume-169, issue-1:13-18, DOI:10.5120/ijca2017914425, July 2017
- Geng, Y., Cassandras, C.G. New "Smart Parking" System Based on Resource Allocation and Reservations, IEEE Trans. Intell. Transp. System, Volume-14, Issue-2,1129–1139, January-2013.
- M.Meenaloshini, J.Ilakkiya, S.Nithyasri, "Smart Car Parking System in Smart Cities using IR", International Conference on Computing and Communication Technologies, Volume-3, Issue-1, DOI:10.1109/ICCCT2.2019.8824953, May-2019.