SMART PARKING USING IOT

TEAM MEMBER

Phase -5 Document submission **J.ARUNKUMAR**

Project:**SMART PARKING** **510421106003**

Abstract:

Today many metropolitan areas have seen explosive growth in the number of visitors and patrons due to urban revitalization, extension of transit services into suburban areas, and the general trend toward increased mobility of our society. As a result, there are too many vehicles on the road and insufficient parking spaces. This has led to the need for an efficient parking management system. With the help of a computerized system we can deliver a good service to citizens who wants to park their vehicle into the any on parking management system. In this context, Internet of Things (IOT) uses sensors to connect physical parking space infrastructures with information and communication technologies, where cloud-based smart management services are provided. To implement this concept a mobile based application would be developed. This mobile application will allow an end user to check the availability of parking space and book a particular parking lot accordingly. Each parking lot would be equipped with a control system that enables monitoring of the number of free and occupied parking places and informing users about the parking lot status (open with/without free available parking spaces or closed) Additionally the application would display parking service payment according to parking time duration. Also it will sense if a vehicle has arrived on the gate for automated gate opening. This allows users to check for available parking space online from anywhere for hassle free parking. Thus, the system solves the parking issue.

Keywords— Smart Parking, Internet of Things, Mobile Application, Show available parking lots as per location, Time.

.

System Requirements:

Hardware Requirements:

- Arduino UNO
- Servo Motor
- Anode Display
- Connectors (Male to Male, Female to Female, Male to Female)

Software Requirements:

Arduino Software To be installed from
https://www.arduino.com/
USB Drivers to be installed in the System

Overview of the existing system

Introduction of Problem & Its Related Concepts:

The existing doesn't have the proper security system and the customers need to park theircars should search and then park but whereas in this project it's clear that they will notified
while they are out of the parking slot the maximum cars that could be able to park and ifthe parking slot is full they may wait still the car comes out or else they can move to anyother parking slots next to them. So, this system makes the user to move from outside ofthe parking slot which is better than coming inside and searching and going out. This parking system will provide the best security and it will give the best parking experienceand a flexible move to the customers park their cars or vehicles in this intelligent parkingsystem lots.

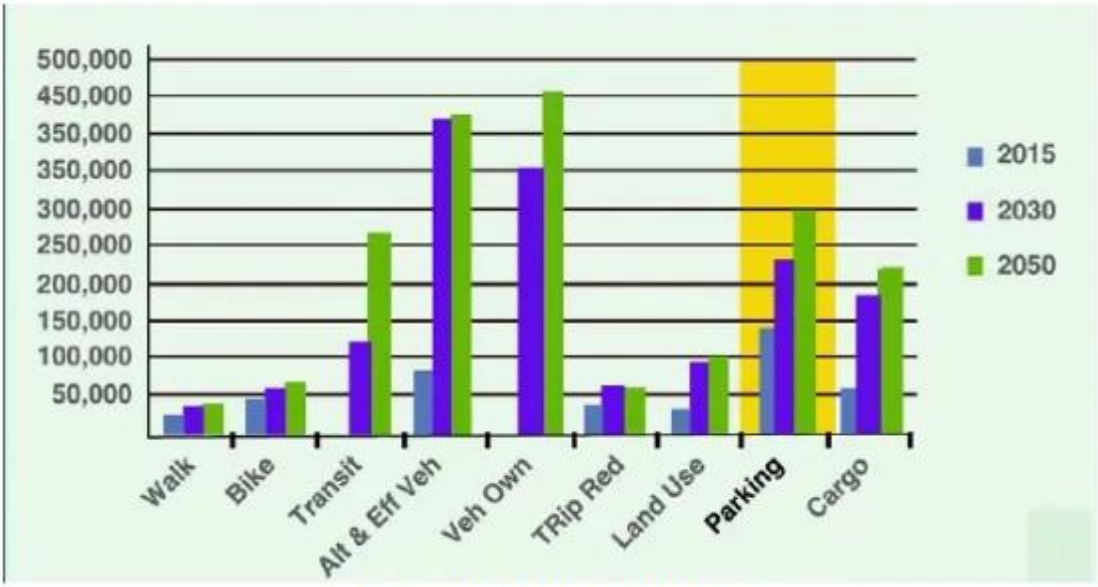


Figure 1 Graph of Comparison

ANALYSIS

Brief IntroductionThe Main Feature:

The parking system has the advantage of working on the basis of the hardware with thisArduino UNO. This System makes the parking system more flexible like when there isloop of n cars to go in only n could go in and the cars exciding than this percentage it willmake the gateway not to be opened. The same thing continues for the next that is the outergateway and the count gets displayed on the Anode Display.

This parking system is generally programed and designed in way in the form of theFOR loop which would be resulting in the repetition of the process still some 'N' numberof times declared in the system as I am using the 8 Segment anode display as my display Iam taking my n as 9 so that it uses a loop and so whenever the parking slot is filled with 9cars it would not open the gateway that is the servo motor in terms of hardware and if it isless than the limit 9 that is 'N' then the gateway opens up and whenever there is an entryinside the gateway the display gets incremented by the flag value over there on the systemand vice-versa that is when the car comes out .

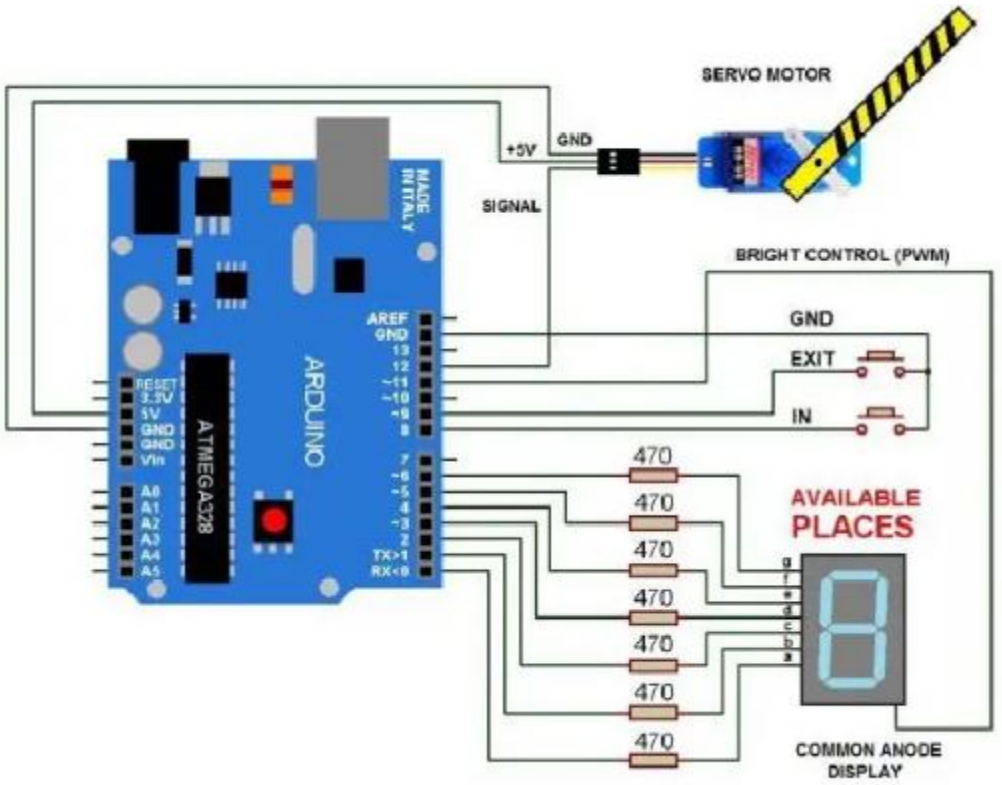
The parking slots it gets decremented byhow many comes out.We are facing the problems of the low parking spaces all around the country and also thereis no proper security because of the salary incrementation for the security persons and thereis lot of consumption of fuel which is also being a big problem as there is less amount offuel available there is more amount price increment and by implementing my project wecan have these things to be control to make our country better and safer place to live andeven the air pollution could be controlled at most 10% of the existing.

Design

Design Overview:

This intelligent parking system has the design of connections of wires that is the JumperWires through which the data is sent to the hardware components connected. The maincomponent used in this project is the Arduino UNO which is a microcontroller which iscontrolled by the program entered in the Arduino IDE which is one of the applicationdeveloped by the Arduino company to make their microcontrollers to have a control. Inthis project, we are using the Anode display and the servo motor. The servo motor is adevice generally which will be able to rotate for about 90 deg. Only but whereas there isanother motor called stepper motor which will rotate up to 180deg. As we need only thegateway so I am using this servo motor which rotates only up to 90deg. Which would usefullike the gateway for the parking slot and next the anode display it is a 7 Segment displaywhich is used to display the numbers by making the bits to be on and off. Since I am usingonly up to 9 cars to be parked inside my parking slot that's why this 7-segment anodedisplay is used. Now, let's see the architecture through which it has been implemented.

The functional requirement are the general requirements needed for the functioning of the product under the execution of the products and the software that has been used to makethe system more accurate and makes the hardware components to work properly and easierwith requirements.



5.2.2 Program Implementation With Arduino IDE:

Arduino Pin	Anode Pin
0	7(a)
1	6(b)
2	4(c)
3	2(d)
4	1(e)
5	9(f)
6	10(g)
~11	8/3 (Common Display)

Table 1 – Anode Display Connections

5.2.2 Program Implementation With Arduino IDE:

```
#include <Servo.h>Servo myservo; // create servo object to control a servo
#define ServoM 12 //Connected to the servo motor
.#define Bright 11 //servo library disable PWM on pins 9 and 10.
#define Exit 9 //Pin connected to the EXIT button
.#define In 8 //Pin connected to the IN button
.#define BarLow 177 //Low position of the barrier
.#define BarUp 95 //Up position of the barrier
.#define CAPACITY 8 //Capacity of the parking lot.
#define INTEN 80 //Display intensity %//Pins conections to segments (cathodes)
.#define segA 0
#define segB
1#define segC 2
#define segD 3
#define segE 4
#define segF 5
#define segG 6
//Array with the segments to represent the decimal numbers (0-9).
byte segments[10] = {
// pgfedcba <--- segments
B00111111, // number 0
B00000110, // number 1
B01011011, // number 2
B01001111, // number 3
B01100110, // number 4
B01101101, // number 5
B01111101, // number 6
B00000111, // number 7
B01111111, // number 8
B11101111 // number 9
};void setup ()
```

```
 {myservo.attach(ServoM) ;  // attaches the servo.
pinMode(Exit, INPUT);  // set "EXIT" button pin to input
pinMode(In, INPUT); // set "IN" button pin to input
digitalWrite(Exit, HIGH); // Connect Pull-Up resistor
.digitalWrite(In, HIGH); // Connect Pull-Up resistor.
 pinMode(segA, OUTPUT);
 pinMode(segB, OUTPUT);
pinMode(segC, OUTPUT);
pinMode(segD, OUTPUT);
pinMode(segE, OUTPUT);
 pinMode(segF, OUTPUT);
 pinMode(segG, OUTPUT);
pinMode(Bright, OUTPUT);
analogWrite(Bright, 255 * INTEN / 100)
myservo.write(BarLow);
 //Barrier in the low position
// delay(1000);}int Available = 9;
 // Number of places available.void loop()
{
Display(Available);
if (digitalRead(In) == 0)
{
if (Available != 0)
{
Available-;myservo.write(BarUp);
delay(3000);
myservo.write(BarLow);
}
}
if (digitalRead(Exit) == 0)
{
if (Available != CAPACITY)
{
Available++;myservo.write(BarUp);delay(3000);
myservo.write(BarLow);
```

```
}
}
}
digitalWrite (segA, bitRead (segs, 0));
digitalWrite (segB, bitRead (segs, 1));
digitalWrite (segC, bitRead (segs, 2));
digitalWrite (segD, bitRead (segs, 3));
digitalWrite (segE, bitRead (segs, 4));
digitalWrite (segF, bitRead (segs, 5));
digitalWrite (segG, bitRead (segs, 6));
}
```

RESULT:

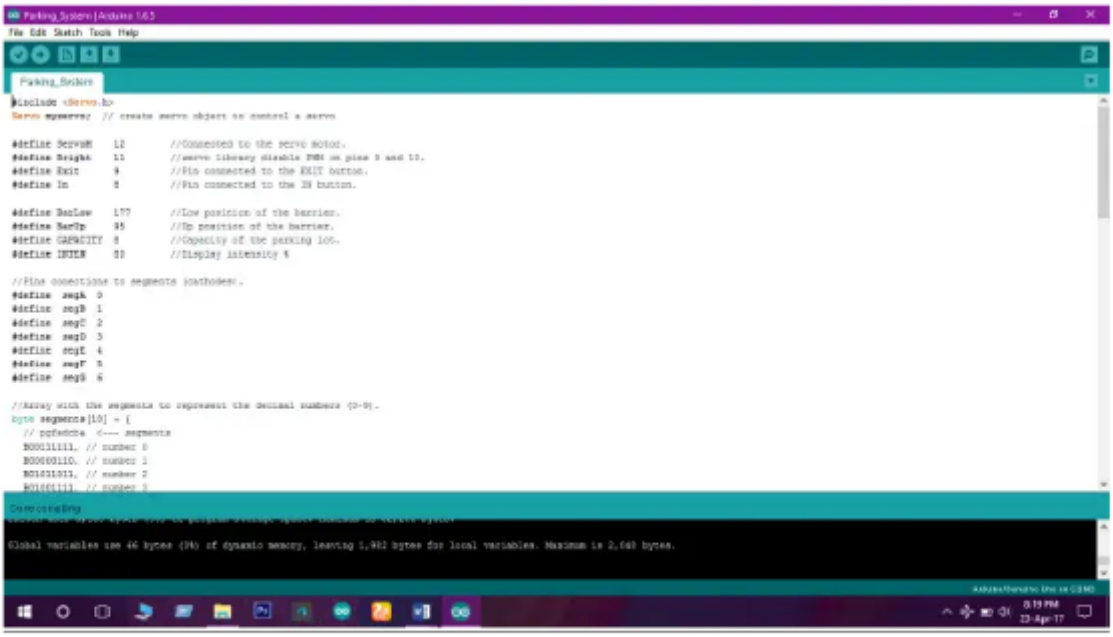
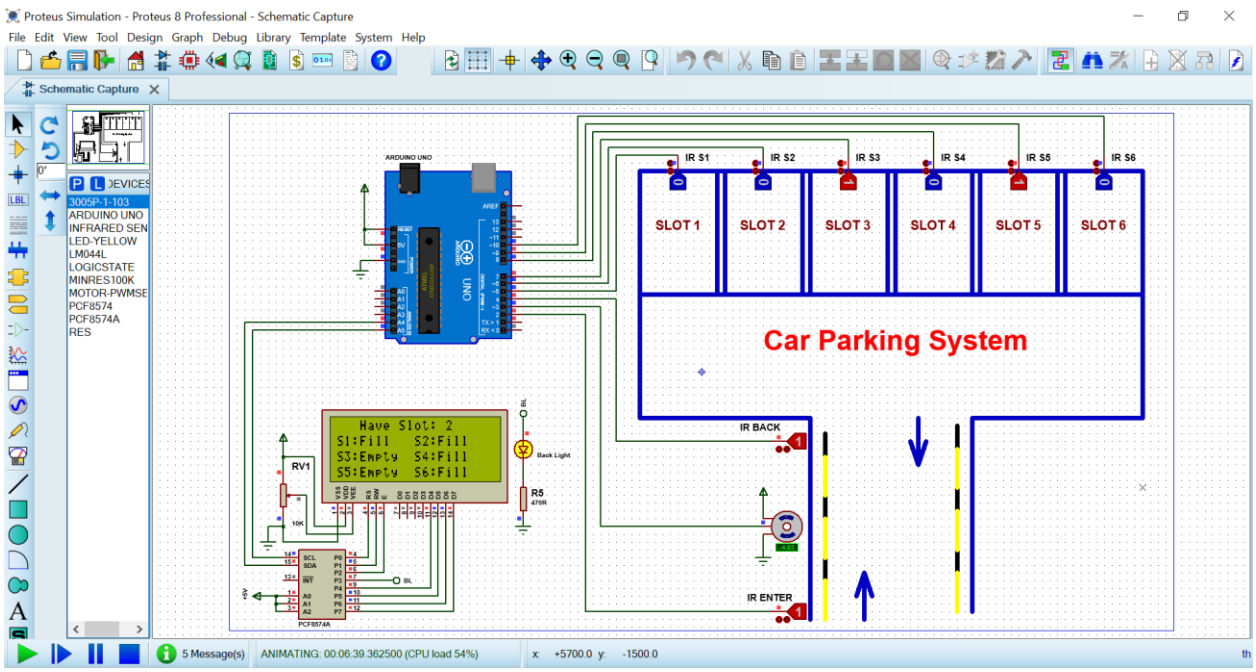


Figure 4 Result Generation

Smart parking in ultrasonicsensor:

Limited parking spaces in certain areas is a common problem for all drivers. A smart parking system helps drivers efficiently find a parking space without the inconvenience of circling blocks, wasting fuel, and wasting valuable time. Our vehicle detection sensors work well to develop this IoT application.



Our Goal in Parking System OEM Applications:

- Provide a competitively priced product to be integrated into OEM parking systems.
- Simplify Operations
- Maximize Profits

Operation of the Smart Parking System

After the user finishes parking their vehicle, the sensor mote above the parking space transmits the mote ID to the management server. The management server updates the parking state for the corresponding space. Next, the user executes the parking application on their smartphone to save the location of their parked vehicle. The parking application employs the RSSI of the BLE module to recognize the location of the parked vehicle.

The USIM ID of the user's smartphone is then sent to the sensor mote before the sensor mote transmits the received USIM ID and the mote ID to the management server. The management server saves all the information. When users request parking location information regarding their vehicles, the parking application submits the USIM ID to the management server.

In addition, if the user requires the location guidance service to find their parking space, the parking application periodically receives RSSI data from the sensor motes deployed around the user. The RSSI data are then transformed into the distances between the user and the sensor motes. Based on three selected distances, the triangulation method is applied periodically and the current location of the user is recognized so the location guidance service can be provided in real-time.

GRAPH DATA

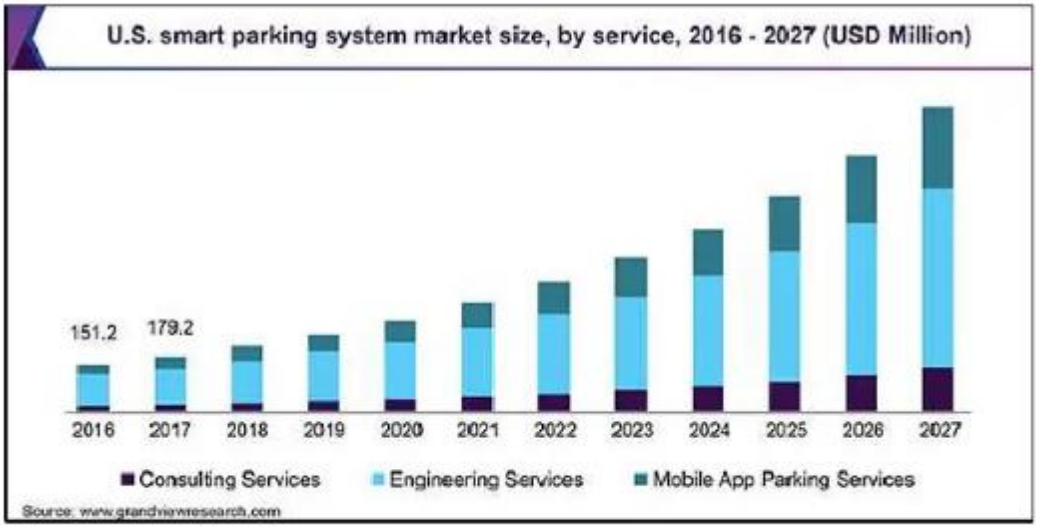


Fig - Estimated Costing Spent By Other Countries

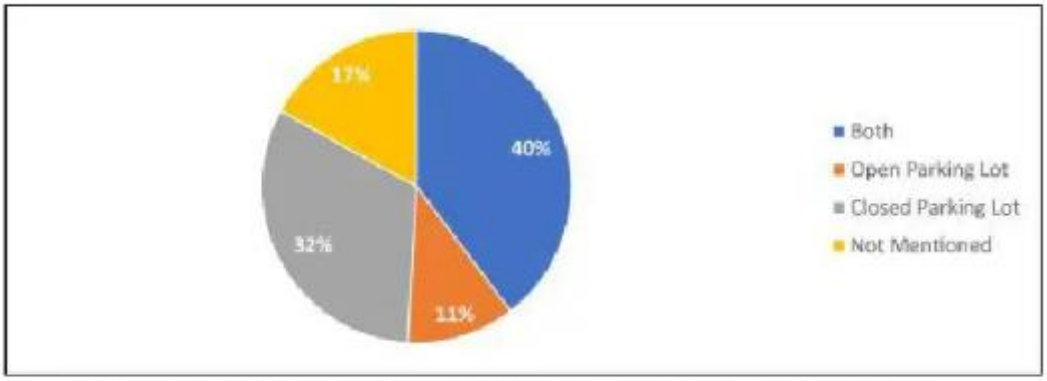


Fig - Usage of camera vision in different types of smart parking systems

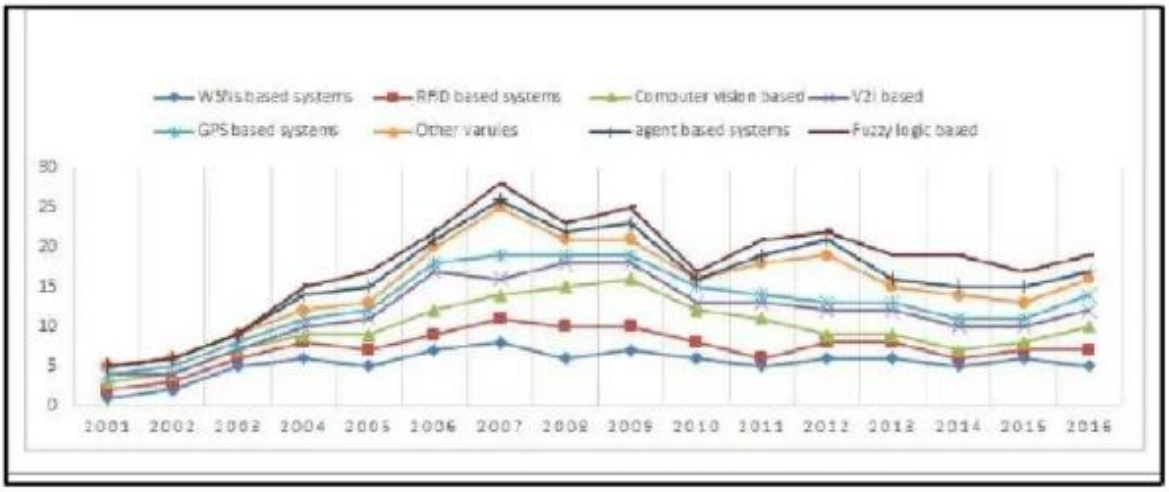


Fig - Different Algorithms Used In Smart Parking Systems Over The Years

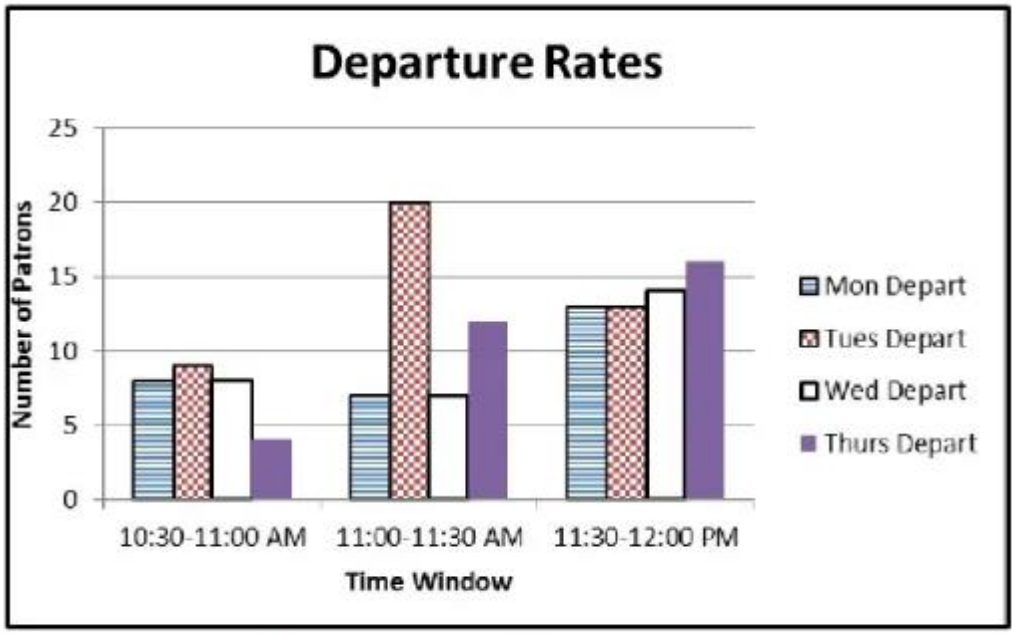


Fig - Data Collected From Test Series

Conclusion

This intelligent parking system which is simple, economic and provides effective solutionto reduce carbon footprints in the atmosphere. It is well managed to access and map thestatus of parking slots from any remote location through the display outside the parkingslots. Thus, it reduces the risk of finding the parking slots in any parking area and also iteliminates unnecessary travelling of vehicles across the filled parking slots in a city. So, itreduces time and it is cost effective also. By implementing this system we could be able tosave the nature and control air pollution and also we could be able to control manpowerthrough this and there will be a great reduction of cost for both the customers and themerchants who make parking slots for their customers.