+


**TRIBHUVAN UNIVERSITY**

**INSTITUTE OF ENGINEERING**

**THAPATHALI CAMPUS**

**A Minor Project Report**

**On**

# IoT Based Smart Parking System

**Submitted By:**

Sandeep Regmi (2072/BEX/333)

Sanjeev Kumar Yadav (2072/BEX/335)

Saugat Tripathi (2072/BEX/337)

**Submitted To:**

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

THAPATHALI CAMPUS

KATHMANDU, NEPAL

November, 2018


**TRIBHUVAN UNIVERSITY**

**INSTITUTE OF ENGINEERING**

**THAPATHALI CAMPUS**

**A Minor Project Report**
**On**
**IoT Based Smart Parking System**

**Submitted By:**

Sandeep Regmi

Sanjeev Kumar Yadav

Saugat Tripathi

**Submitted To:**

Department of Electronics and Computer Engineering
Thapathali Campus
Kathmandu, Nepal

In partial fulfillment for the award of the Bachelor's Degree in Electronics and Communication Engineering

**Under the Supervision of**

Er. Shanta Maharjan

November, 2018

**DECLARATION**

We hereby declare that the report of the project entitled **"IoT Based Smart Parking System"** which is being submitted to the **Department of Electronics and Computer Engineering, IOE, Thapathali Campus**, in the partial fulfillment of the requirements for the award of the Degree of Bachelor of Engineering in **Electronics and Communication Engineering**, is a bonafide report of the work carried out by us. The materials contained in this report have not been submitted to any University or Institution for the award of any degree and we are the only author of this complete work and no sources other than the listed here have been used in this work.

Sandeep Regmi (Roll No: 072/BEX/333) _____

Sanjeev Kumar Yadav (Roll No: 072/BEX/335) _____

Saugat Tripathi (Roll No: 072/BEX/337) _____

**Date:** November, 2018

**CERTIFICATE OF APPROVAL**

The undersigned certify that they have read and recommended to the **Department of Electronics and Computer Engineering, IOE, Thapathali Campus**, a minor project work entitled "**IoT Based Smart Parking System**" submitted by **Sandeep Regmi, Sanjeev Kumar Yadav and Saugat Tripathi** in partial fulfillment for the award of Bachelor's Degree in Electronics and Communication Engineering. The Project was carried out under special supervision and within the time frame prescribed by the syllabus. We found the students to be hardworking, skilled and ready to undertake any related work to their field of study and hence we recommend the award of partial fulfillment of Bachelor's degree of Electronics and Communication Engineering.

_____

Project Supervisor

Er. Shanta Maharjan

Department of Electronics and Computer Engineering, Thapathali Campus

_____

External Examiner

Department of Electronics and Computer Engineering, Pulchowk Campus

_____

Project Co-ordinator

Er. Saroj Shakya

Department of Electronics and Computer Engineering, Thapathali Campus

_____

Er. Kiran Chandra Dahal

Head of the Department,

Department of Electronics and Computer Engineering, Thapathali Campus

November, 2018

## COPYRIGHT

**ACKNOWLEDGEMENT**

We take this opportunity to express our profound gratitude and appreciation to our supervisor Er. Saroj Shakya for his excellent academic and professional guidance throughout this project. We would like to give a special appreciation to Er. Kiran Chandra Dahal (Head of Department) for providing us an opportunity to designate this project.

We are thankful to all the teaching faculty of the Department of Electronics and Computer for their support and help. Lastly, we would like to appreciate the help from our colleagues, our parents and friends for their unwavering encouragement and appreciation.

Sandeep Regmi (Class Roll No.: 072/BEX/333)

Sanjeev Kumar Yadav (Class Roll No.: 072/BEX/335)

Saugat Tripathi (Class Roll No.: 072/BEX/337)

**ABSTRACT**

The major cities lack enough spaces in order to provide a good facility of parking, drivers find parking spaces depending upon the experience and luck. Smart Parking System (SPS) aims to mitigate the current problem of parking management using a prototype of IoT based Smart Parking System using ESP32 and ultrasonic sensor. The occupancy state of the parking space and establishing the connection to the cloud server through status of each parking space are sent using HTTP protocol providing a reliable communication link to be accessed by clients. Using the designing the vacant spaces are denoted by green and the occupied one by red on the user end. With proper scalability the system can eradicate the parking management problem altogether.

*Keywords- IoT, Smart Parking System (SPS), HTTP, ESP32*

**TABLE OF CONTENTS**

**LIST OF FIGURES**

**LIST OF TABLES**

## LIST OF ABBREVIATION

| | |
|---|---|
| CPN | Car Park Network |
| HTTP | Hyper Text Transfer Protocol |
| HTTPS | Hyper Text Transfer Protocol Secure |
| ID | Identification |
| IDE | Integrated Development Environment |
| IoT | Internet of Things |
| MQTT | Message Queuing Telemetry Transport |
| NFC | Near Field Communication |
| RFID | Radio Frequency Identification |
| SPS | Smart Parking System |
| TCP/IP | Transmission Control Protocol/ Internet Protocol |
| Wi-Fi | Wireless Fidelity |
| WWW | World Wide Web |

# 1 INTRODUCTION

The main motto of our program is to minimize the problem of vehicles parking by using sensors to sense the vehicles parked and available free spaces in a parking hall, IoT to make information accessible via internet.

## 1.1 Background Introduction

The major cities are facing the traffic problem due to excessive number of vehicles used. The excessive use of vehicles have also created the problem of vehicle parking, people don't get enough space for parking and eventually park vehicles on roadsides which creates traffic jams. Currently, the common method of finding a parking space is manual where the driver finds a space in the street through luck and experience.

This process takes time and effort and may lead to worst case of failing to find a parking space if the driver is driving in a city with high vehicle density. An alternative is to create a parking hall. However, this is not the optimal solution as the parking hall may be very far away from the driver. Use of sensors to check the vacancy of spaces and displaying the status of parking hall via internet can thus minimize the parking problem and reduce the time required for searching the free spaces.

## 1.2 Motivation

As living in the country like Nepal where the lack of proper transport and traffic management is growing and hindering the quality of life of all the citizens living we aimed to find the root cause, one of which turned out to be lack of proper parking management. So to resolve this issue we begin to make a list of all the possible option which in turn may help to mitigate this problems. Thus we opted for the Smart Parking System.

## 1.3 Problem Definition

The number of cars used in major cities has been growing day by day. The increase in density of vehicles creates a serious problem of unmanaged and tedious parking. The major cities lack enough spaces in order to provide a good facility of parking, drivers find parking spaces depending upon the experience and luck. Even when people go for shopping, they spend most

of their time in search of free space to park their vehicles. People nowadays rather prefer to walk to the markets than to take their cars.

The major problems faced by people due to unmanaged parking are:

- Time taken to search for a free space to park is really too much.
- People in hurry often get late to their works while searching for parking spaces.
- There is no guarantee that a person gets a free space to park depending upon the time spend in searching.
- Congested parking causes damages and scratches in the cars.
- In parking halls people often lose the parking slip and are compelled to pay a large amount.

Hence to minimize this problem and to make parking efficient, the IoT based smart parking system is required.

### 1.4 Objectives

- To develop a prototype of IoT based Smart Parking System using ESP32 and ultrasonic sensor.
- To make it easier for drivers to search parking spaces easily via internet.

### 1.5 Scope and Application

If we look at the records of number of vehicles used, it has been growing rapidly since few years. It is obvious that within few years from now, the number of cars used would be very high that it would be impossible to manage the parking by the traditional way.

Hence, to sustain a large number of vehicles running and to minimize the problems of parking, the Smart Parking System can be used. Implementing this system, we can check the condition of free spaces in a parking hall easily by just checking it online. In addition to that, if the number of available free spaces is very less we can make our reservation few minutes earlier using smart phones or tablets online. This would minimize the unwanted time wastage in searching free spaces.

## 1.6 Report Organization

The lack of proper parking management is a growing problem. The use of a current IoT based SPS can eliminate this problem. The use of sensors for the detection of free spaces and listing that to the user can be productive for all. The use of IoT means that any user can access the parking space remotely. Thus although of several limitation of the IoT and the data transfer protocol used it still makes sense to use a Smart Parking System instead of using a traditional method for searching parking spaces.

## 2 LITERATURE REVIEW

Traffic congestion is not only the problem in Nepal but in the whole world. According to TomTom's website, there is always the presence on congestion somewhere in the world at the very moment [1]. The main cause behind these traffic jams is the increase in the number of vehicles the people searching for a parking space. As it accounts for more than 30% of traffic congestion in urban area, the lack of proper parking management is a serious problem [2]. Thus, the need of smart parking is an absolute necessity now and in the near future.

### 2.1 Internet of Things (IoT)

The paradigm shifting technology of the future is Internet of Things. The term coined by Kevin Ashton merely is the combination of two words "Internet" and "Things". The Internet is the networks of the networks of computer network connecting the several devices to form a global communication link among humans to serve billions of people of the whole wide world using internet protocol suit (TCP/IP). It consists of private, public, academic, business, and government networks of local to global scope, linked by a broad array of electronic, wireless, and optical networking technologies [3]. The number of devices connected in IoT is estimated to be in between 6.6 and 20 billion by 2020 [4]. All the smart devices including smart watches, smart wearable, coordinated traffic lights, home automation system and the devices connected to them are all based on IoT.

In the smart parking system IoT is mainly used to communicate between the several users on a remote basis. In the system proposed by Khanna *et.al*. [5], the parking management rely only on the users input. They created a mobile application platform through which user can book his/her parking space and need to conform their occupancy though the application. Also they used the alarm system to avoid the person who are not genuine. It is cheaper and easier to implement. However, the occupancy state can only be checked using the mobile app and there is the lack of hardware so have to rely on the users input. Also there is no backup plan in the absence or broken internet connection. The booking and the payment is done on hour basis i.e. the booking can be done only on the multiple of hour and user need to pay for a whole hour just to extend some minutes.

In another study [6], the authors Salpietro *et.al.* proposed a Smart Parking System (SPS) that rely on the embedded sensor of the mobile phones to eliminate human from the equation. It makes the system automatic. It makes the system easy to use on the end user however taking the account of several sensors from many mobile phones makes it difficult on the development. Also, it rely only on the mobile phone sensors so if the user doesn't have a phone with them then it can be a problem.

In other study [7], Pham *et.al.* proposed the use of RFID technology. RFID reader counts the percentage of free parking spaces in each car park. They proposed the use of RFID so that the as it is cheaper to implement at a large scale and has more feasibility than other method of SPS. Cloud computing is done so that the user is updated to the real parking time. They used the Car Park Network (CPN), so the multiple parking station can be connected to a same network and the user can choose the one of his liking calculated form the number of parked car and the distance from the user. However, they didn't take payment into consideration. Also, it is a complex system while taking mathematical models into account.

# 3 REQUIREMENT ANALYSIS

For the completion of the project different types of hardware and software were to be required and feasibility study was performed.

## 3.1 Hardware and Software Requirements

### 3.1.1 ESP32

ESP32 integrates Wi-Fi (2.4 GHz band) and Bluetooth 4.2 solutions on a single chip, along with dual high performance cores, Ultra Low Power co-processor and several peripherals. Powered by 40 nm technology, ESP32 provides a robust, highly integrated platform to meet the continuous demands for efficient power usage, compact design, security, high performance, and reliability.

Espressif provides the basic hardware and software resources that help application developers to build their ideas around the ESP32 series hardware. The software development framework by Espressif is intended for rapidly developing Internet-of-Things (IoT) applications, with Wi-Fi, Bluetooth, power management and several other system features. [8]

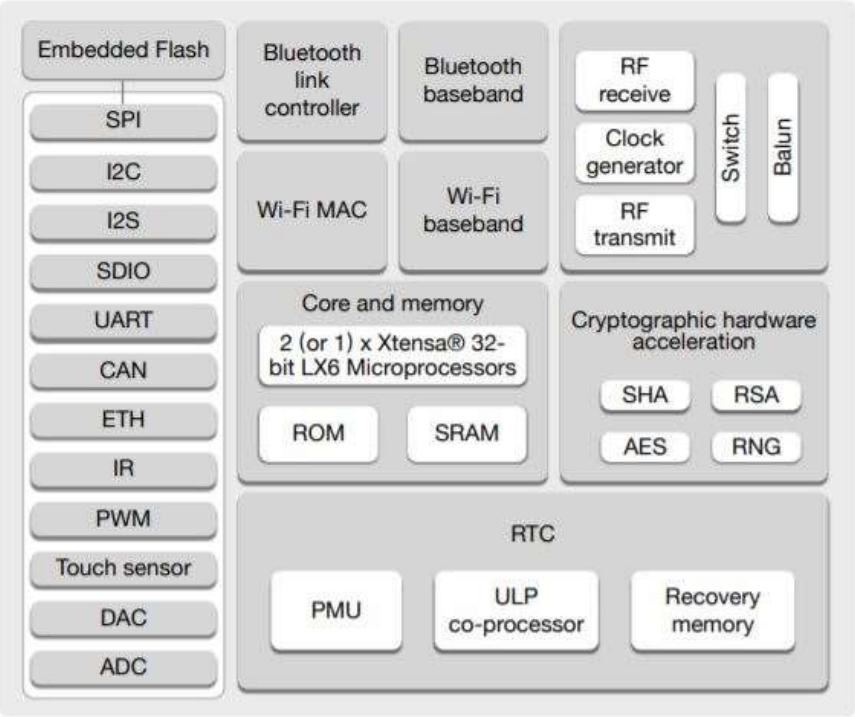


*Figure 3-1 Functional block diagram of ESP32*

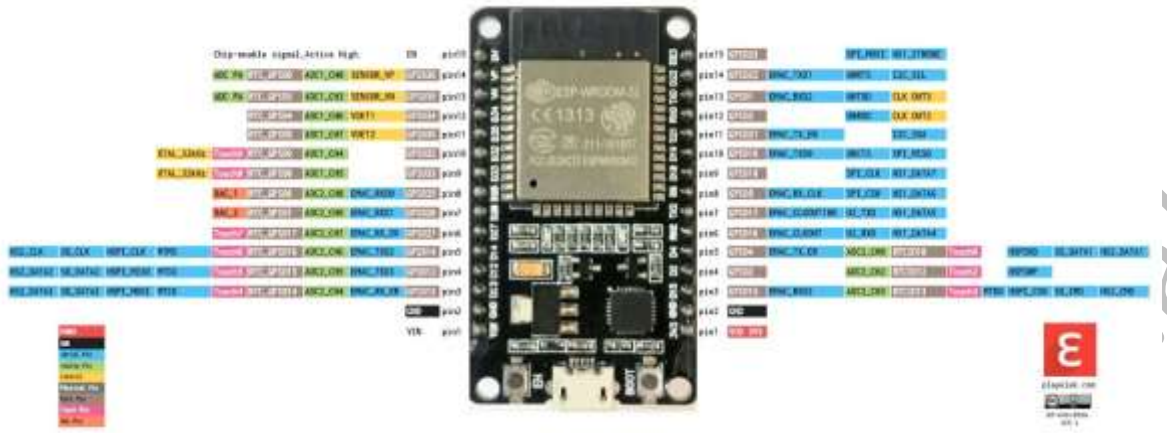

*Figure 3-2 DOIT ESP32 Pinout*

### 3.1.2 Ultrasonic Sensor (HC-SR04)

Ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules includes ultrasonic transmitters, receiver and control circuit. The basic principle of work:

- Using IO trigger for at least 10us high level signal,
- The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.
- If the signal back, through high level , time of high output IO duration is the time from sending ultrasonic to returning. [9]

*Figure 3-3 Ultrasonic sensor*

### 3.1.3 Arduino IDE

The Arduino integrated development environment (IDE) is a cross-platform application that is written in the programming language Java. It is used to write and upload programs to Arduino board or other boards from different manufacturer as it is open source. For our project Arduino IDE was used due to the simplicity of use and code and as there was the availability of the Arduino core of ESP32.

## 3.2 Feasibility Study

The project of "IoT based Smart Parking System" is completed with the help of ESP32 which itself is a powerful processor capable of handling the communication and the processing simultaneously due to dual core processor. As the cost of developing this project is way too cheap so for a small scale the parking management system the same program and some additional ultrasonic sensors could be used as a feasible solution. It uses a HTTP protocol for the transfer of the sensor data so although it may not be secure form of connection but is simple and cheap to implement. But for large scale parking management system multiple ESP32 could be utilized to for a large network of things and using MQTT protocol could be connected to a single webhost. And using the MQTT protocol is more secure and is IoT friendly thus keeps the cost to minimal but rises the complexity. Although the existence of this complexity it is still a feasible solution as compared to other methods.

## 4 SYSTEM ARCHITECTURE AND METHODOLOGY

For our system we have used Arduino IDE for programming the ESP32 and ultrasonic sensor for checking the vacancy of the parking space. First ESP32 connects to the Wi-Fi network specified in the code and reads the sensor data and check the state of parking space and sends the data to the database using HTTP protocol. And on the server side of the program the data is uploaded to the database as per the "upload.php" code and this data is then shown on the website main page for user interaction.

### 4.1 Block Diagram

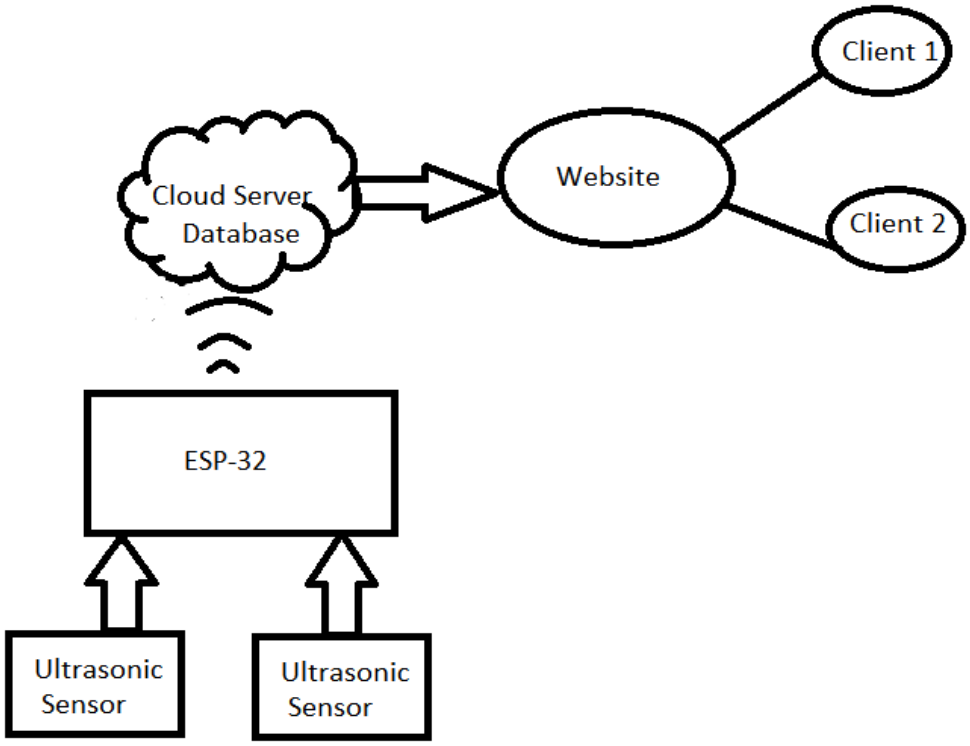


*Figure 4-1 System Architecture*

### 4.1.1 ESP32

ESP32 acts as a brain for the sensor and controlling the sensors reading the data and determining the condition of the parking space and sending the result to the database on the cloud for further processing. It contains built in Wi-Fi capability utilizing which it can connect to a Wi-Fi network and thus to the internet. ESP32 assign a value of '0' for occupied and '1' for the vacant space for each variable of the parking space as per our threshold value (in our project of 70 centimeters i.e. the space with empty space of 70 cm or more is assigned 1 and less than 70 cm is assigned 0). And these values of '0' or '1' on variables are sent to the database using HTTP POST of HTTP protocol.

### 4.1.2 Sensor

These are the small devices that actually senses the environment and send the data to the ESP32. For our project we have used ultrasonic sensor as they provide accurate result and are practically possible to use.

### 4.1.3 Cloud Database

'Upload.php' contains the database information and is used for uploading the data sent by ESP32 using HTTP protocol. In this way, the database store the data send by ESP32 and provides the data to the website for the user interaction.

## 4.2 Data Flow Diagram

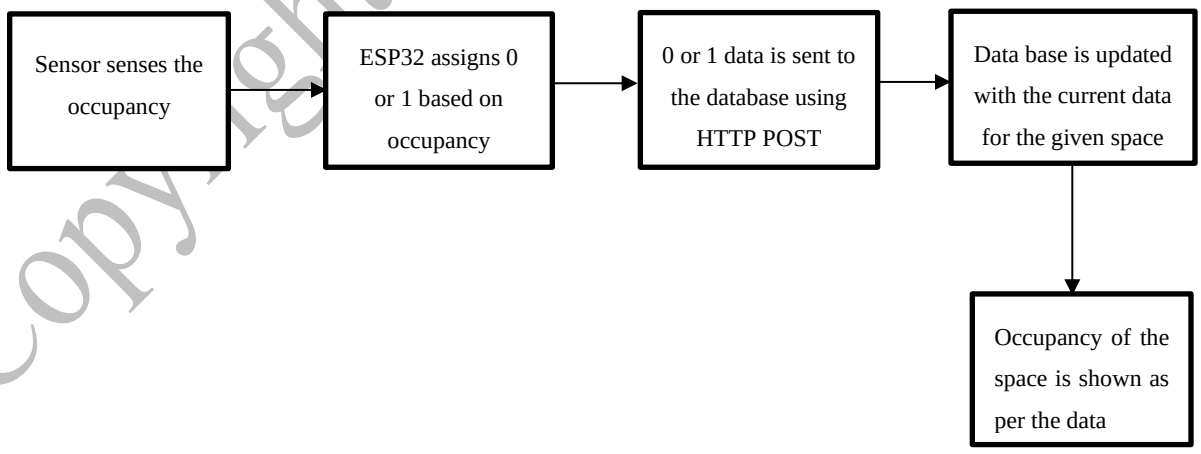


*Figure 4-2 Data flow diagram*

# 5 IMPLEMENTATION DETAIL

Almost all of the methodology described earlier were completed using programing ESP32 and the server side was completed using MySQL for the database and PHP for the backend while CSS was used mainly for designing the website.

## 5.1 Hardware component

The only hardware component used for implementing the project was ESP32, Ultrasonic Sensor HC-SR04, 7805A voltage regulator, 9V battery, battery cap, jumpers and breadboard. All the connection were made as defined on the code for the trigger and echo pin of the ultrasonic sensor. And the power supply was done using the 9V battery using 7805A voltage regulator for +5V dc supply.

## 5.2 Library used for ESP32

### 5.2.1 WiFi.h

WiFi.h library is library file developed specially for Arduino board though they still quite work for ESP32. This library is used for establishing Wi-Fi connection to a certain given SSID. This library allows ESP32 board to connect to the internet. It can serve as either a server accepting incoming connections or a client making outgoing ones. The library supports WEP and WPA2 Personal encryption, but not WPA2 Enterprise. Also note, if the SSID is not broadcast, the board cannot connect. [10]

### 5.2.2 HTTPClient.h

HTTPClient.h library is used to fulfill the HTTP requests like HTTP POST, GET and PUT request to the webserver.

In the ESP32 code an extern variable Wi-Fi is used which is defined inside of the WiFi.h library. This variable is used to call almost all of the functions inside the library. First of all the Wi-Fi connection is established with the SSID network. And there are two functions defined ‘readSensor’ and ‘readParkingSpace’ which reads the sensor value and determine the occupancy state respectively and is assigned to a variable of the parking space. After this a HTTP connection is established and the data are sent and the site is accessed which load the upload.php code thus uploading the data to the database. Also specifying the header is

necessary for defining which type of content (data) are being sent through the HTTP POST. In our case the content type is ‘application/x-www-form-urlencoded’. And the sensor data is read every 12 seconds.

# 6 RESULT AND ANALYSIS

The output of IoT based smart parking system can be viewed through the our website of any device anywhere in the world i.e. it shows which spaces are vacant and which are occupied.

## 6.1 Output of ESP32 on serial monitor

```
Connecting to the WiFi......
Connecting to the WiFi......
Connecting to the WiFi......
Connected to the WiFi NetworkSaugat WiFi
80:7D:3A:F3:94:F8
192.168.43.208
200



29
0
Parking Space A1 0
Parking Space B2 0
200



192
124
Parking Space A1 1
Parking Space B2 1
200



4
112
Parking Space A1 0
Parking Space B2 1
```

*Figure 6-1 Result seen on serial monitor*

Here on the serial monitor we can see first of all ESP32 connecting to the Wi-Fi network and displaying it's corresponding SSID and IP on the network. And we can see the latest data being sent are 0 and 1 of A1(chamber 1) and B2(chamber 2) respectively. The distance is shown for debugging purposes only. And 200 at the top of each data indicates that an successful connection of HTTP was established and the data was sent successfully.

## 6.2 Result on the webpage

The corresponding result of the data being sent is displayed in webpage. The corresponding data that was sent through the ESP32 is displayed on the webpage showing that the chamber 1 is occupied while chamber 2 is empty.


*Figure 6-2 Result on webpage*

# 7 CONCLUSION AND FUTURE ENHANCEMENT

## 7.1 Conclusion

Our project aim to alleviate the problem of traffic management. Using a cheap sensors and a processing unit to control and send data to the cloud is a simple but a strong solution for a small scale. In the case for a large scale the same concept can be used with some further enhancement on the protocol to combine the multiple system into one cohesive system. Thus our system has solve one of the major issues among the similar system of scalability. In this way the simple parking system could be established that can surely eradicate the parking problem.

## 7.2 Limitation

Limitations are bound to happen in any system as no system can be perfect. The Smart Parking System have several limitation including the security of the data being exchanged. The use of HTTP protocol for the system instead of using HTTPS or more secure protocol decreases the security of the data being transferred and can fall under the trap of hacking. Also a single unit of our system is limited to the number of parking space by the number of GPIO pins of the ESP32 chip i.e.36 pins.

## 7.3 Future Enhancement

- The number of sensors can be added for increasing the system and its accuracy.
- Multiple system can be implemented in the same network to further scale up the system and using some algorithms like Dijkstra's algorithm for finding optimal parking space.
- Using MQTT protocol for future and larger system provides more security and flexibility on the data.
- The webpage could be further enhanced to make it appealing for the eyes.
- Instead of using the free webhost, paid subscription for hosting can provide more reliable system
- The same system could be added using some extra features like booking and automatic payment through accounts using RFID upon leaving the parking complex.

# 8 APPENDICES

## Appendix A: Module specification

| | |
|---|---|
| **Working Voltage** | **DC 5 V** |
| **Working Current** | **15mA** |
| **Working Frequency** | **40Hz** |
| **Max Range** | **4m** |
| **Min Range** | **2cm** |
| **MeasuringAngle** | **15 degree** |
| **Trigger Input Signal** | **10uS TTL pulse** |
| **Echo Output Signal** | **Input TTL lever signal and the range in proportion** |
| **Dimension** | **45*20*15mm** |

*Table 8-1 Ultrasonic Sensor HC-SR04 Parameters*

## Appendix B: Schematics diagram

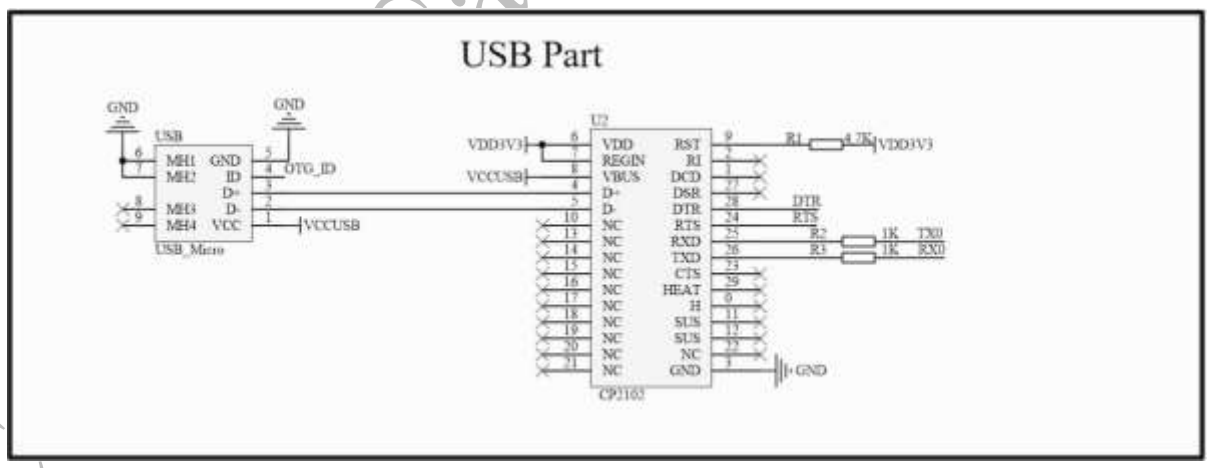


*Figure 8-1 Schematic diagram of USB part*

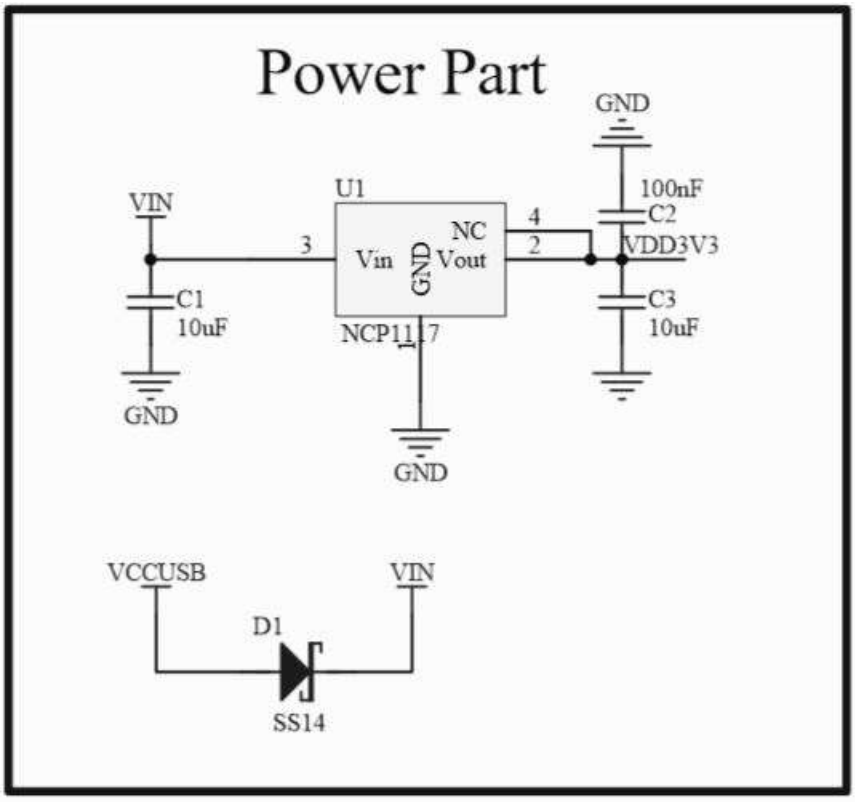


*Figure 8-2 Schematic diagram of power parts*

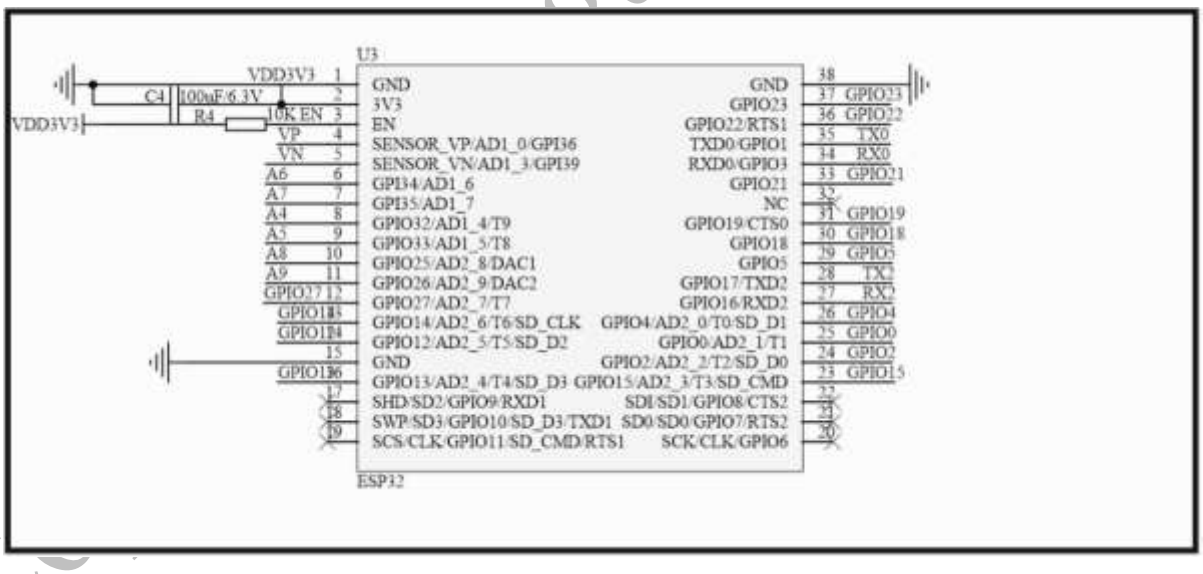


*Figure 8-3 Schematic diagram of ESP32*

**Appendix C: Library Functions**

**WiFi.h library**

- ConnectNoEncryption : Demonstrates how to connect to an open network
- ConnectWithWEP : Demonstrates how to connect to a network that is encrypted with WEP
- ConnectWithWPA : Demonstrates how to connect to a network that is encrypted with WPA2 Personal
- ScanNetworks : Displays all WiFi networks in range
- WiFiChatServer : Set up a simple chat server
- WiFiWebClient : Connect to a remote webserver
- WiFiWebClientRepeating : Make repeated HTTP calls to a webserver
- WiFiWebServer : Serve a webpage from the WiFi shield
- WiFiSendReceiveUDPString : Send and receive a UDP string
- UdpNTPClient : Query a Network Time Protocol (NTP) server using UDP

WiFi class initializes the ethernet library and network settings.

- begin() Begin the Wi-Fi connection
- disconnect() Disconnect the Wi-Fi connection
- config() Configure the network
- setDNS() Set the DNS
- SSID() SSID of the connected network
- BSSID() BSSID of the connected network
- RSSI() RSSID of the connected network
- encryptionType() Type of encryption of the network
- scanNetworks() Scan for Wi-Fi networks
- status() Show the status of the network
- getSocket() Get the socket
- macAddress() MAC address of the device

**REFERENCES**


[1] TomTom, "TomTom," 26 05 2018. [Online]. Available: https://www.tomtom.com/en_gb/trafficindex/list?citySize=LARGE&continent=ALL&country=ALL.

[2] R. Arnott, T. Rave and R. Schob, Alleviating urban traffic congestion, MIT Press, 2005.

[3] S. Madakam, R. Ramaswamy and S. Tripathi, "Internet of Things (IoT): A Literature Review," *Journal of Computer and Communications,* pp. 164-173, 2015.

[4] Statista, "Statista," Statista, 2018. [Online]. Available: https://www.statista.com/topics/2637/internet-of-things/. [Accessed 26 05 2018].

[5] A. Khanna and R. Anand, "IoT based Smart Parking System," in *Maharashtra Institute of Technology*, Pune, 2016.

[6] R. Salpietro, L. Bedogni, M. D. Felice and L. Bononi, "Park Here! A Smart Parking System based on Smartphones' Embedded Sensors and Short Range Communication Technologies," in *Internet of Things (WF-IoT), 2015 IEEE 2nd World Forum on*, Milan, 2015.

[7] T. Pham, M. Tsai, N. D.B, D. C.R and D. Deng, "A Cloud-Based Smart-Parking System Based on Internet-of-Things Technologies," *IEEE Access* , vol. 3, pp. 3122-3124, 2015.

[8] "Espressif," [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/latest/get-started/index.html. [Accessed 23 11 2018].

[9] "https://www.mouser.com/ds/2/813/HCSR04-1022824.pdf," Mouser, [Online]. Available: https://www.mouser.com/ds/2/813/HCSR04-1022824.pdf. [Accessed 23 11 2018].

[10] Arduino, "arduino.cc," [Online]. Available: https://www.arduino.cc/en/Reference/WiFi. [Accessed 24 11 2018].

[11] C. Sun, "Application of RFID Technology for Logistics on Internet of Things," Elsevier B.V., 2012.

[12] Microchip Technology Inc., "Microchip," MIcrochip, 01 February 2011. [Online]. Available: https://www.microchip.com/wwwproducts/en/ATmega32. [Accessed 30 May 2018].

[13] R. Want, "An Introduction to RFID Technology," *IEEE CS and IEEE ComSoc*, 2006.

[14] Sunrom Electronics, "Sunrom," Sunrom, [Online]. Available: https://www.sunrom.com/p/mifare-rfid-readerwriter-1356mhz-rc522. [Accessed 30 May 2018].

[15] "Wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/Radio-frequency_identification. [Accessed 30 May 2018].

[16] Microchip, "Microchip," Microchip Technology Inc., [Online]. Available: http://www.microchip.com/mplab/avr-support/atmel-studio-7. [Accessed 30 May 2018].

[17] Espressif, "Espresif," [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf. [Accessed 24 11 2018].