Smart Car Parking System

Ishan Bajpai[1], Arcot Abid Ali[2], Aditya tewari[3]

[1,2,3]Department of Electronics and Communication Engineering,
Vellore Institute of Technology,
Katpadi, Vellore, India

[1]ishan.bajpai2016@vitstudent.ac.in,
[2]arcotabid.ali2017@vitstudent.in,
[3]aditya.tewari2017@vitstudent.ac.in

Under the guidance of
Professor Noor Mohammed V.

School of Electronics Engineering,
VIT University, Vellore.

ABSTRACT

As the efficient of internet of Things (Iot) is rising and is alter for remote observation of the surrounding parameters and various stuffs with the utilization of sensors that acquaint for wireless sensing of real time data and transfer them into the specified kind and facilitate to forward the perceived data across the network cloud via Internet Connection. Here, the project work deals with internet service that would be a generous open API service that act as variety for the vary of sensors to look at the perceived data at cloud level and composite a special feature of porting the perceived data using a channel ID and browse API key that is assigned by services and prepared to trace data value.

In recent times the concept of smart cities has gained great popularity of Iot in order to maximize the productivity and reliability of urban infrastructure.

Problems such as, traffic congestion, limited car parking facilities and road safety are being addressed by Iot. In this paper, we present an Iot based cloud integrated smart parking system. The proposed Smart Parking system consists of an on-site deployment of an Iot module that is used to monitor and signalize the state of availability of each single parking space. A mobile application is also provided that allows an end user to check the availability of parking space and book a parking slot accordingly.

1. INTRODUCTION

Car parking is a major issue in modern congested cities of today. There simply are too many vehicles on the road and not enough parking spaces. This problem increases the traffic congestion on roads and increases the risks of road accidents. Also, drivers don't have prior information of the empty parking slots thus leading to inefficient utilization of the parking resources.

We demonstrate the use of IOT based parking management system that allows for efficient parking space utilization using IOT technology. This allows users to check for available parking spaces online from anywhere and avail hassle free parking. Thus, the system solves the parking issue for cities and get users an efficient IOT based parking management system. Smart parking makes the search for parking space easier and convenient for driver.

Our project aims to detect the number of empty parking slots and send the information over the Internet to the smart parking applications which can be accessed by the drivers using their smartphones, tablets, and in car navigation systems. The parking system will be fully automatic which will make use of IR technology to sense if a vehicle has arrived on gate for automated gate opening and toll collection using servo motors to simulate as gate openers. Thus, our parking system would be able to work without any human intervention.

Sensors are used for each parking slot to detect whether the slot is empty or not, and this information is aggregated by local controller and then sent over the Internet to database.

2. TECHNICAL SPECIFICATIONS

This section describes the various technical specifications related to the project.

2.1 Hardware specifications

1. Arduino UNO (At mega 328).

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, 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. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm centre-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and VIN pin headers of the POWER connector. The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

Figure 1:- *Arduino UNO(AT MEGA 328)*

2. Node MCU Wi-Fi Module (ESP8266).

NodeMCU is an open source development board and firmware based in the widely used ESP8266 -12E Wi-Fi module. It allows you to program the ESP8266 Wi-Fi module with the simple and powerful LUA programming language or Arduino IDE.
With just a few lines of code you can establish a Wi-Fi connection and define input/output pins according to your needs exactly like Arduino, turning your ESP8266 into a web server and a lot more. It is the Wi-Fi equivalent of Ethernet module. Now you have internet of things (iot) real tool.
With its USB-TTL, the NodeMCU Dev board supports directly flashing from USB port. It combines features of WIFI access point and station + microcontroller. These features make the NodeMCU extremely powerful tool for Wi-Fi networking. It can be used as access point and/or station, host a webserver or connect to internet to fetch or upload data.

Finally, programmable Wi-Fi module.
Arduino-like (software defined) hardware IO.
Can be programmed with the simple and powerful Lua programming language or Arduino IDE.
USB-TTL included, plug & play.
10 GPIOs D0-D10, PWM functionality, IIC and SPI communication, 1-Wire and ADC A0 etc. all in one board.
Wi-Fi networking (can be used as access point and/or station, host a web server), connect to internet to fetch or upload data.
Event-driven API for network applications.
PCB antenna.

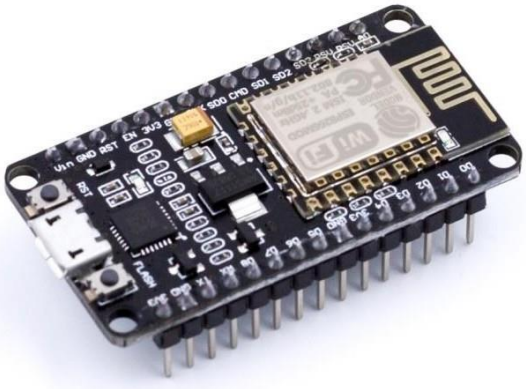

Figure 2:- *Node MCU Wi-Fi Module (ESP8266).*

3. I R Transmitter Receiver Module.

Most remote controls for your TV, cable box, Blu-ray player or other consumer electronic devices use infrared signals. IR is also an inexpensive and effective way to control a variety of maker projects. IR is especially useful in creating assistive technology devices for the disabled, such as mouse and keyboard emulators.
In another tutorial, we describe how to use an open source library called " IRLib" which allows you to decode or encode a wide variety of standard protocols used by consumer electronic devices. This IR breakout board help you get the most out of IRLib. Here is a tutorial on how to use IRLib.
The Transmitter and Receiver Board
This board is also an integral component in an assistive technology device called the "Ultimate Remote" which is documented.
The device consists of 2 infrared LEDs driven by 3 transistors as well as a TSOP38xxx receiver chip and a TSMP58000 learner chip. We've even thrown in some extra ground and power pins that makes it easier to integrate this board into your project.
It can handle 5v or 3.3v power input and the transmit and receive portions of the board can be independently powered at either voltage.
The board is completely open source and there are Eagle CAD files provided as well as links to PCB manufacturers who can create the raw board for you. A complete price list is provided. Most of the parts are available from Adafruit.
The design is flexible and has many options such as including or not including current limiting resistors, the orientation of the receiver chips, and power supply options. Each of these options is described in the assembly instructions of this tutorial.

Figure 3:- *I R Transmitter Receiver Module*

4. Servo motors

A **servo motor** is an electrical device which can push or rotate an object with great precision. If you want to rotate and object at some specific angles or distance, then you use servo motor. It is just made up of simple motor which run through **servo mechanism**. If motor is used is DC powered then it is called DC servo motor, and if it is AC powered motor then it is called AC servo motor. We can get a very high torque servo motor in a small and light weight packages. Doe to these features they are being used in many applications like toy car, RC helicopters and planes, Robotics, Machine etc.

Servo motors are rated in kg/cm (kilogram per centimetre) most hobby servo motors are rated at 3kg/cm or 6kg/cm or 12kg/cm. This kg/cm tells you how much weight your servo motor can lift at a particular distance. For example: A 6kg/cm Servo motor should be able to lift 6kg

if the load is suspended 1cm away from the motors shaft, the greater the distance the lesser the weight carrying capacity.

The position of a servo motor is decided by electrical pulse and its circuitry is placed beside the motor.

Figure 4:- *Servo motor*

2.2 Software Specifications

▯ Arduino compiler (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 3rd 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 tool chain, also included with the IDE distribution. The Arduino IDE employs the program *avrdude* 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.

▯ Blynk mobile app

Similar API & UI for all supported hardware & devices.
Connection to the cloud using:

Wi-Fi
Bluetooth and BLE
Ethernet
USB (Serial)
GSM
Set of easy-to-use Widgets
Direct pin manipulation with no code writing
Easy to integrate and add new functionality using virtual pins
History data monitoring via Super Chart widget
Device-to-Device communication using Bridge Widget
Sending emails, tweets, push notifications, etc.

- **Libraries for ESP8266 and Blynk**

- **Embedded C language**

3. DESIGN APPROACH AND DETAILS

3.1. Material & Methods

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

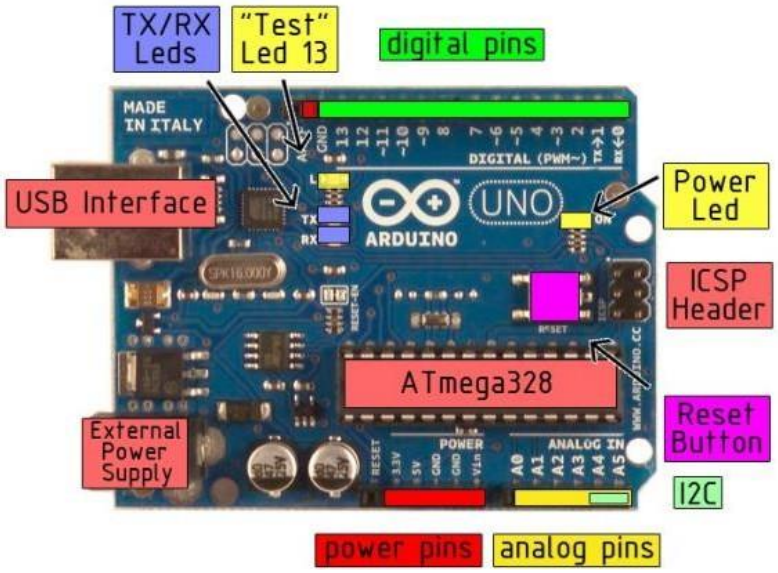

Figure 5:- *Arduino UNO PIN description*

Arduino is the brain for the whole system. It controls and watches over all the components. The ultrasonic sensors will be placed in the parking slots that will encounter the presence of the cars inside the parking slots. One sensor will be placed beside the main entrance of the parking lot.

As soon as the sensors get the presence of a car in front of the entrance, it will send signal to the Arduino chip to check if there is an empty slot inside the parking lot. When Arduino chip acknowledges that there is an empty slot or more then it will send signal to the dc servo motor which will open the main entrance. On the other hand if Arduino chip encounter no empty slots at the time of a car trying to make entrance, the gate will just not open.

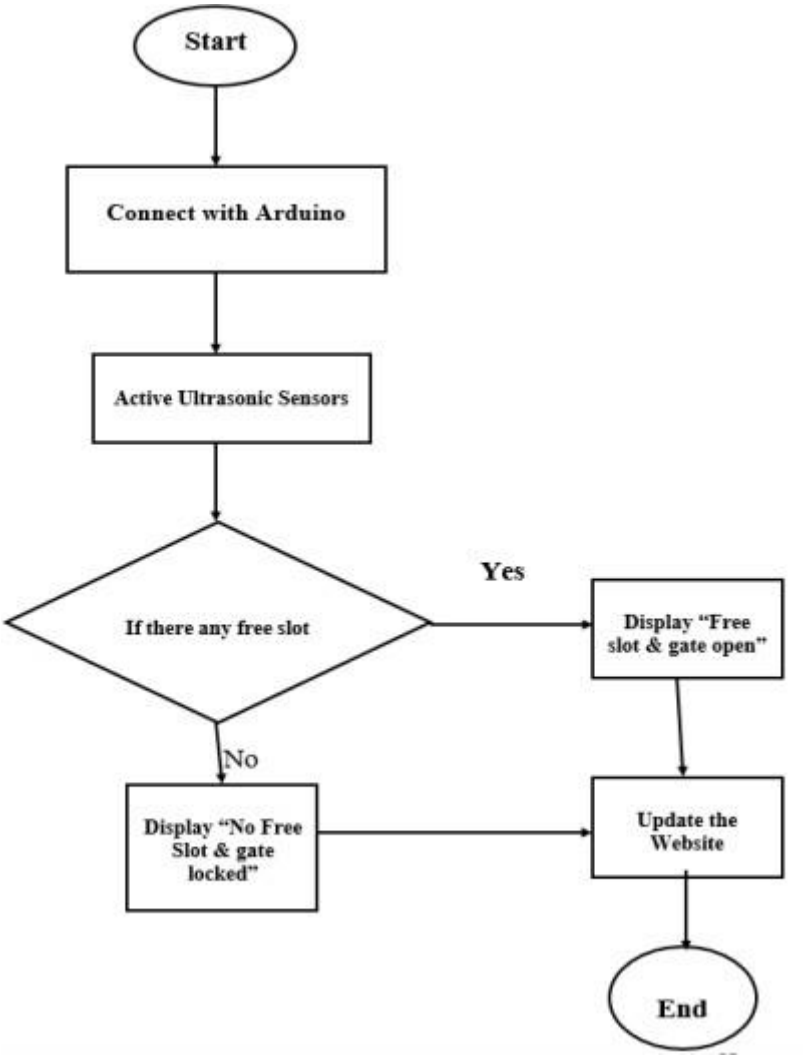

Figure 6:- *FLOWCHART of the system*

To demonstrate the concept, we use IR sensors for sensing parking slot occupancy along with a dc motors to simulate as gate opener motors.

We have used a Node MCU module for internet connectivity and also facilitate as an edge computing device for operating the system and sending the data to the cloud.

The data from the sensors is aggregated by the Node MCU using the cellular mobile network. The data which is collected is displayed on the Blynk application.

We will use DC motors as gates of our parking system. IR sensors will detect vehicles at the opening gate and open the gate.

The slot occupancy will be displayed on the parking application.

The user will be able to pay the parking charges using the same application which will be calculated on basis of the time the vehicle has occupied the slot.

The exit gate will open only when the user makes the payment.

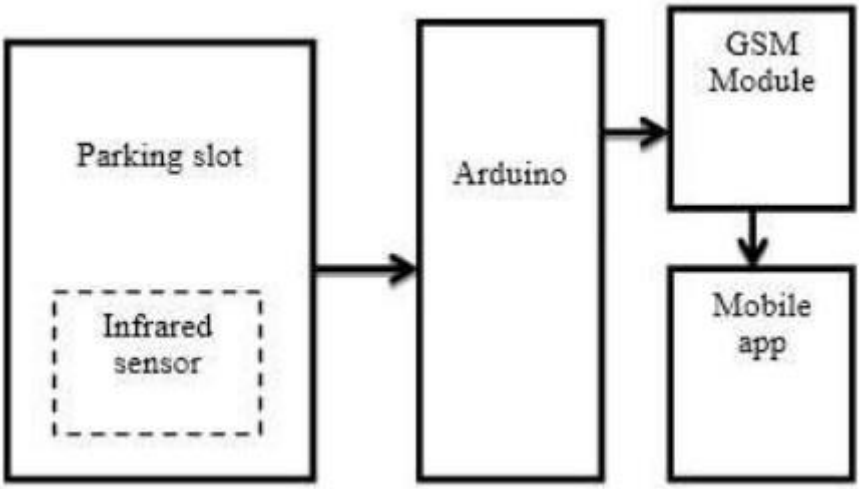

Figure 7:- *BLICK DIAGRAM of circuit*

UNDERSTANDING THE BLYNK APPLICATION AND IT'S USES

Blynk works on the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, visualize it and do many other cool things. There are three major components in the platform:

- **Blynk App** - allows to you create amazing interfaces for your projects using various widgets we provide.
- **Blynk Server** - responsible for all the communications between the smartphone and hardware. You can use our Blynk Cloud or run your private Blynk server locally. It's open-source, could easily handle thousands of devices and can even be launched on a Raspberry Pi.
- **Blynk Libraries** - for all the popular hardware platforms - enable communication with the server and process all the incoming and out coming commands.

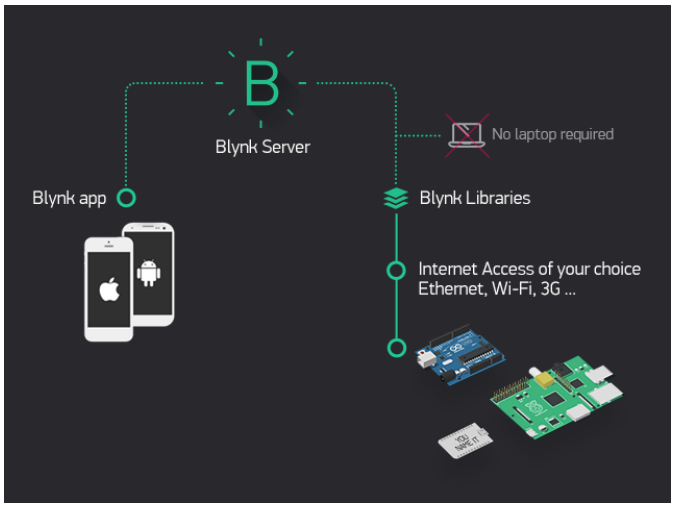

Figure 8:- *BLYNK APP functioning*

4.2 Codes and Standards

The Libraries used in the code are:

- BLYNK_PRINT serial
- Esp8266Wifi.h
- BlynkSimpleEsp8266.h
- Servo.h

Connection procedure of Node MCU to Blynk cloud:

- char auth [] = "ZxvcmGVbUYPqFb6W2B2Ev5tduUdRp9Q3";
- char ssid [] = "ishan";
- char pass [] = "physics2017";

Declaration of variables:

```
bool flag;
bool flag1;
bool e;
WidgetLED led1(V1);
WidgetLED led2(V2);
WidgetLCD lcd(V3);
Servo servo1;
Servo servo2;
int a,b,c,d;
int x;
int y;
void setup ()
{
Serial.begin(9600);
```

```
pin Mode (D0, INPUT);
 pinMode (D1, INPUT); pinMode (D2, INPUT);
 pinMode (D4, INPUT);
 servo1.attach(14);
servo2.attach(12);
servo1.write(0);
servo2.write(0);
Blynk.begin(auth, ssid, pass);
flag=true;
flag1=true;
}
 void loop ()
{
Blynk.run ();
a=digitalRead(D0);
b=digitalRead(D1);
c=digitalRead(D2);
d=digitalRead(D4);
Serial.println(c);
```

Code for entry gate:

```
if(c==0)
{
Serial.print("a"); servo1.write(90);
}
else { servo1.write(0);
}
```

Code for slot 1:

```
if(a==0)
```

```
{
led1.on();
if(flag) { x=millis (); flag=false;
}
}
else
{
led1.off ();
if (! flag) {
lcd.clear();
lcd.print (0,0,"Your parking"); lcd.print (0,1,"cost is Rs ");
lcd.print (12,1, ((mills()-x)/1000) *10); flag=true;
}
}
```

Code for slot 2:

```
if(b==0) {
led2.on(); if(flag1) { y=millis (); flag1=false;
}
}
else
{
led2.off ();
if (! flag1) {
lcd.clear();
lcd.print (0,0,"Your parking"); lcd.print (0,1,"cost is Rs ");
lcd.print (12,1, ((millis()-y)/1000) *10); flag1=true;
}
}
if(d==1) {
```

```
servo2.write(0);
}
}
```

Code for exit gate:

```
BLYNK_WRITE(V4)
{
d=digitalRead(D4); e=V4;
if (d==0 and e==true)
{
servo2.write(90);
}
}
```

5. CONCLUSIONS

After doing study on smart parking project we have found that this system can be introduced in our region and it will be beneficiary in the given context. The main benefits are time and fuel saving which cannot be understated. It can also provide sustainable parking management in an eco-friendly manner. There is less maintenance cost for this system so it is helps the property developer in cost saving. It provides security to the parking ground. It reduces the hassle in parking grounds and traffic jam. It will also encourage Automation Engineering in our country which will make advancement in increasing usage of technology. Given what we know of the growth of our country and number of vehicles entering the roads every day, it's important that the parking systems can keep up with it, because it brings in a lot of growth potential with it along with many opportunities for big revenue increase by implementing simple and efficient systems in parking spaces without much hassle of installation and without the need for adding any extra devices for the users. By overall estimation, it's an efficient and feasible system.

Therefore, we should implement this project and help to develop cities parking systems.

6. REFERENCES

- Saidur Rahman, “IOT based smart parking system,” International Journal of Advances in Computer and Electronics Engineering, January 2019, pp. 11–16, Volume 4, Issue 1.
- Leena Mhatre ; Neha Rai, “Integration Between Wireless Sensor and Cloud”, International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) 2017
- Suha I. Al-nassar, Haraa Raheem Hatem, Jinan N. Shehab, “Design and Implementation of Infrared (IR) Communication System” ,Diyala Journal of Engineering Sciences, Vol. 11, No. 3, September 2018
- Pooja More1 , Diksha Pawar2, Gaurav Mahure3 , Sneha Bansod4 , Ankit Manke5, “Wi-Fi E Switch using Internet of Things”, International Research Journal of Engineering and Technology (IRJET), Volume: 05 Issue: 03 | Mar-2018
- Khondker Shajadul Hasan, Mashiur Rahman, Abul L. Haque,M Abdur Rahman, Tanzil
- Rahman and M Mahbubur Rasheed,“Cost Effective GPS-GPRS