SMART PARKING SYSTEM USING ESP8266

INDEX

ABSTRACT

The purpose of smart Parking System is that to with growing popularity of Smart Cities, there is always a demand for smart solutions for every domain. The IOT has enabled the possibility of Smart Cities with it's over the internet control feature. A person can control the devices from anywhere in the world by just using a smartphone or any internet connected devices. There are multiple domains in a smart city and Smart Parking is one of the popular domain in the Smart City.

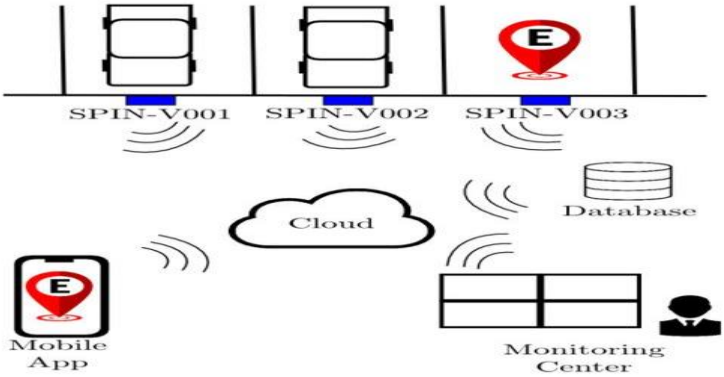
The Smart Parking industry has seen a number of innovations such Smart Parking Management System, Smart Gate Control, Smart Cameras which can detect types of vehicle, ANPR (Automatic Number Plate Recognition), Smart Payment System, Smart Entry System and many more. Today similar approach will be followed and a smart parking solution will be built which will use an ultrasonic sensor to detect vehicle presence and trigger the gate to open or close automatically. The ESP8266 NodeMCU will be used here as the main controller to control all the peripherals attached to it.

In this IOT Smart Parking System, we will send data to webserver for looking up the availability of space for vehicle parking. Here we are using firebase as Iot database to get the parking availability data. For this we need to find the Firebase host address and the secret key for authorization.

INTRODUCTION

With growing popularity of Smart Cities, there is always a demand for smart solutions for every domain. The IoT has enabled the possibility of Smart Cities with it's over the internet control feature. A person can control the devices installed in his home or office from anywhere in the world by just using a smartphone or any internet connected devices. There are multiple domains in a smart city and Smart Parking is one of the popular domain in the Smart City.

The Smart Parking industry has seen a number of innovations such Smart Parking Management System, Smart Gate Control, Smart Cameras which can detect types of vehicle, ANPR (Automatic Number Plate Recognition), Smart Payment System, Smart Entry System and many more. Today similar approach will be followed and a smart parking solution will be built which will use an ultrasonic sensor to detect vehicle presence and trigger the gate to open or close automatically. The ESP8266 NodeMCU will be used here as the main controller to control all the peripherals attached to it.



Smart Parking systems typically obtains information about available parking spaces in a particular geographic area and process is real-time to place vehicles at available positions .It involves using low-cost sensors, real-time data collection, and mobile-phone-enabled automated payment systems that allow people to reserve parking in advance or very accurately predict where they will likely find a spot. When deployed as a system, smart parking thus reduces car emissions in urban centers by reducing the need for people to needlessly circle city blocks searching for parking. It also permits cities to carefully manage their parking supply Smart parking helps one of the biggest problems on driving in urban areas; finding empty parking spaces and controlling illegal parking. This implies M2M technologies aims rightness/safety as well as convenience.

COMPONENTS REQUIRED:

- ESP8266 NodeMCU
- Ultrasonic Sensor
- DC Servo Motor
- IR Sensors
- 16x2 i2c LCD Display
- Jumpers

SOFTWARE REQUIRED: Arduino IDE.

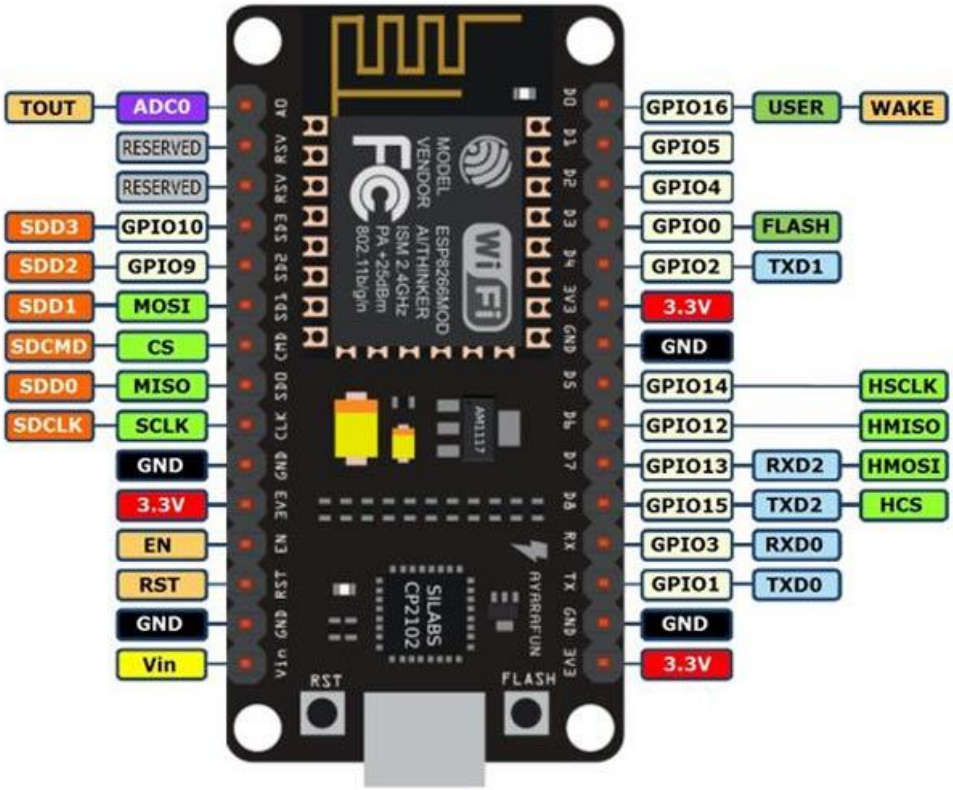
DESCRIPTION ABOUT COMPONENTS:

ESP8266 NodeMCU:-

NodeMCU is an open-source Lua based firmware and development board specially targeted for IoT based Applications. It includes firmware that runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module.

The **NodeMCU ESP8266 development board** comes with the ESP-12E module containing the ESP8266 chip having Tensilica Xtensa 32-bit LX106 RISC microprocessor. This microprocessor supports RTOS and operates at 80MHz to 160 MHz adjusta ble clock frequency. NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power with in-built Wi-Fi / Bluetooth and Deep Sleep Operating features make it ideal for IoT projects.

NodeMCU can be powered using a Micro USB jack and VIN pin (External Supply Pin). It supports UART, SPI, and I2C interface.



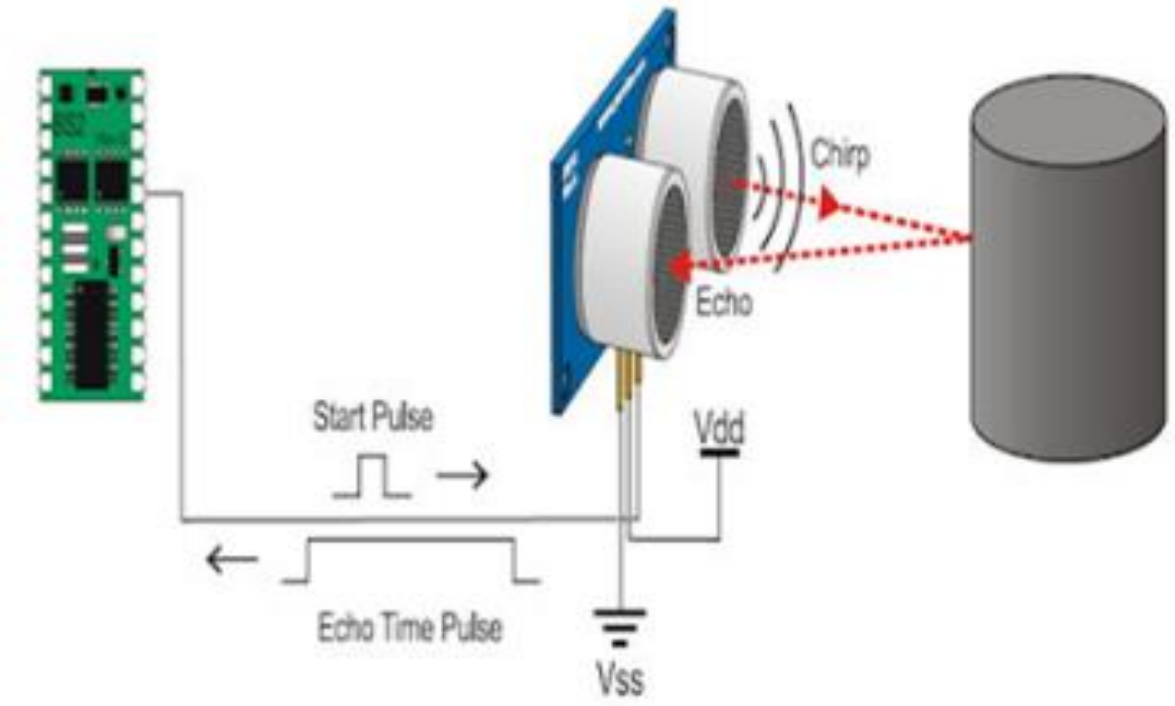
Pin Category	Name	Description
Power	Micro-USB, 3.3V, GND, Vin	**Micro-USB:** NodeMCU can be powered through the USB port **3.3V:** Regulated 3.3V can be supplied to this pin to power the board **GND:** Ground pins **Vin:** External Power Supply
Control Pins	**EN, RST**	The pin and the button resets the microcontroller
Analog Pin	A0	Used to measure analog voltage in the range of 0-3.3V
GPIO Pins	GPIO1to GPIO16	NodeMCU has 16 general purpose input-output pins on its board
SPI Pins	SD1,CMD, SD0, CLK	NodeMCU has four pins available for SPI communication.
UART	TXD0, RXD0,	NodeMCU has two UART interfaces, UART0 (RXD0 & TXD0)

Pins	TXD2, RXD2	and UART1 (RXD1 & TXD1). UART1 is used to upload the firmware/program.
I2C Pins		NodeMCU has I2C functionality support but due to the internal functionality of these pins, you have to find which pin is I2C.

Ultrasonic sensor:

An ultrasonic sensor is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves, and converts the reflected sound into an electrical signal. Ultrasonic waves travel faster than the speed of audible sound.Ultrasonic sensors have two main components: the transmitter (which emits the sound using piezoelectric crystals) and the receiver (which encounters the sound after it has travelled to and from the target).

Humans can detect sounds in a frequency range from about 20Hz to 20 KHz. However, the frequency range normally employed in ultrasonic detection is 100 KHz to 50MHz. Ultrasonic sensors reach up to 8,000 mm (or 8 meters) and thus can be used within most small and medium distances. The ultrasonic sensor (or transducer) works on the same principles as a radar system. An ultrasonic sensor can convert electrical energy into acoustic waves and vice versa. The acoustic wave signal is an ultrasonic wave traveling at a frequency above 18kHz. HC SR04 ultrasonic sensor generates ultrasonic waves at 40kHz frequency.



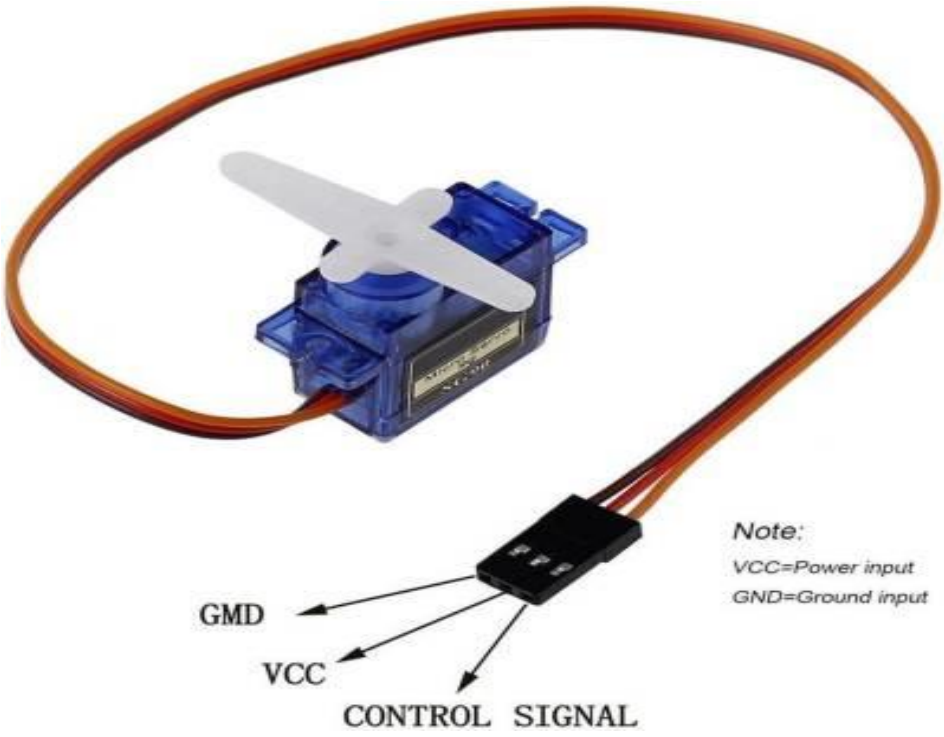
Specifications:-

- Power Supply: 3.3V – 5V
- Operating Current: 8mA
- Working Frequency: 40Hz
- Ranging Distance : 3cm – 350cm/3.5m
- Resolution : 1 cm
- Measuring Angle: 15 degree
- Trigger Input Pulse width: 10uS TTL
- Dimension: 50mm x 25mm x 16mm

DC Servo Motor:-

A type of servomotor that uses DC electrical input to generate mechanical output like velocity, acceleration or position is known as DC servomotor. It is somewhat similar to a normal DC motor. However, there exist some differences between a normal dc motor and dc servomotor.

Basically, DC servomotors of all types are required to be excited individually. This leads to provide linear characteristics between torque and speed.



IR Sensors:-

IR sensor is an electronic device, that emits the light in order to sense some object of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion. Usually, in the infrared spectrum, all the objects radiate some form of thermal radiation. These types of radiations are invisible to our eyes, but infrared sensor can detect these radiations.

The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photodiode . Photodiode is sensitive to IR light of the same wavelength which is emitted by the IR LED. When IR light falls on the photodiode, the resistances and the output voltages will change in proportion to the magnitude of the IR light received.

There are five basic elements used in a typical infrared detection system: an infrared source, a transmission medium, optical component, infrared detectors or receivers and signal processing. Infrared lasers and Infrared LED's of specific wavelength used as infrared sources.

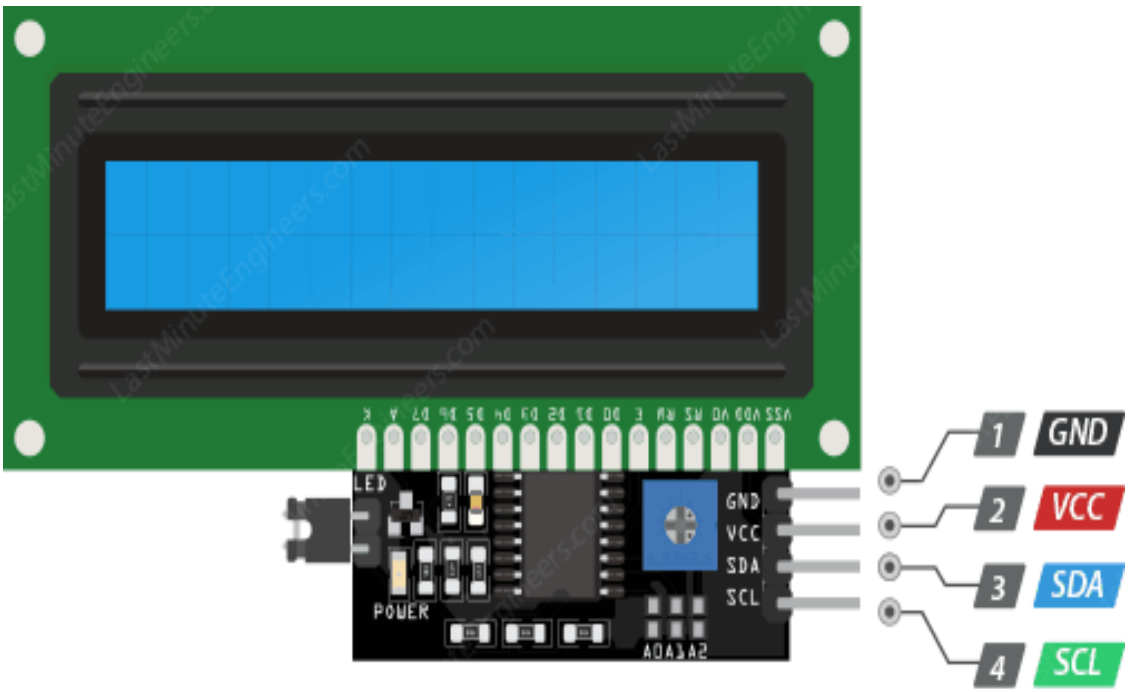
The three main types of media used for infrared transmission are vacuum, atmosphere and optical fibers. Optical components are used to focus the infrared radiation or to limit the spectral response.

16x2 i2c LCD Display:-

16x2 LCD display screen with I2C interface. It is able to display 16x2 characters on 2 lines, white characters on blue background.

Usually, Arduino LCD display projects will run out of pin resources easily, especially with Arduino Uno. And it is also very complicated with the wire soldering and connection. This I2C 16x2 Arduino LCD Screen is using an I2C communication interface. It means it only needs 4 pins for the LCD display: VCC, GND, SDA, SCL. It will save at least 4 digital/analog pins on Arduino. All connectors are standard XH2.54 (Breadboard type). You can connect with the jumper wire directly.

To avoid the confliction of I2C address with other I2C devices, such ultrasonic sensor, IMU, accelerometers, and gyroscope, the I2C address of the module is configurable from 0x20-0x27. And its contrast can be adjusted manually.

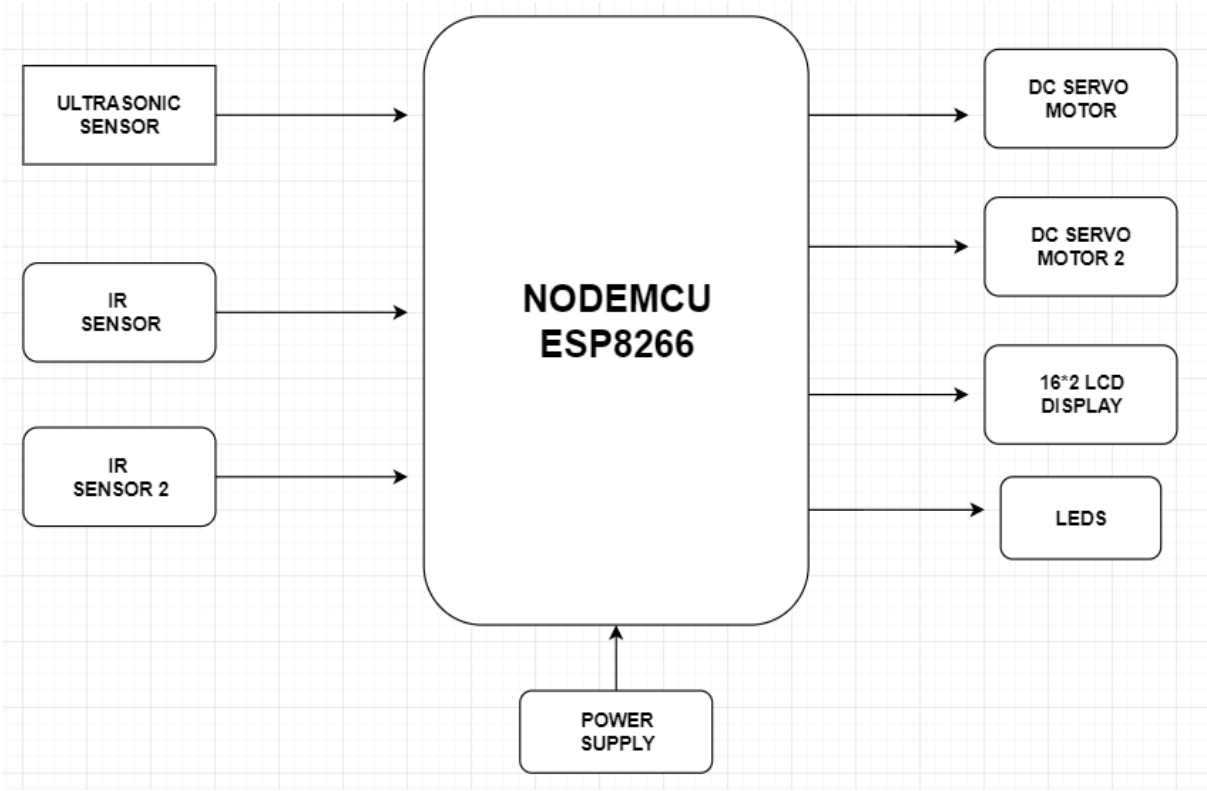


WORKING:-

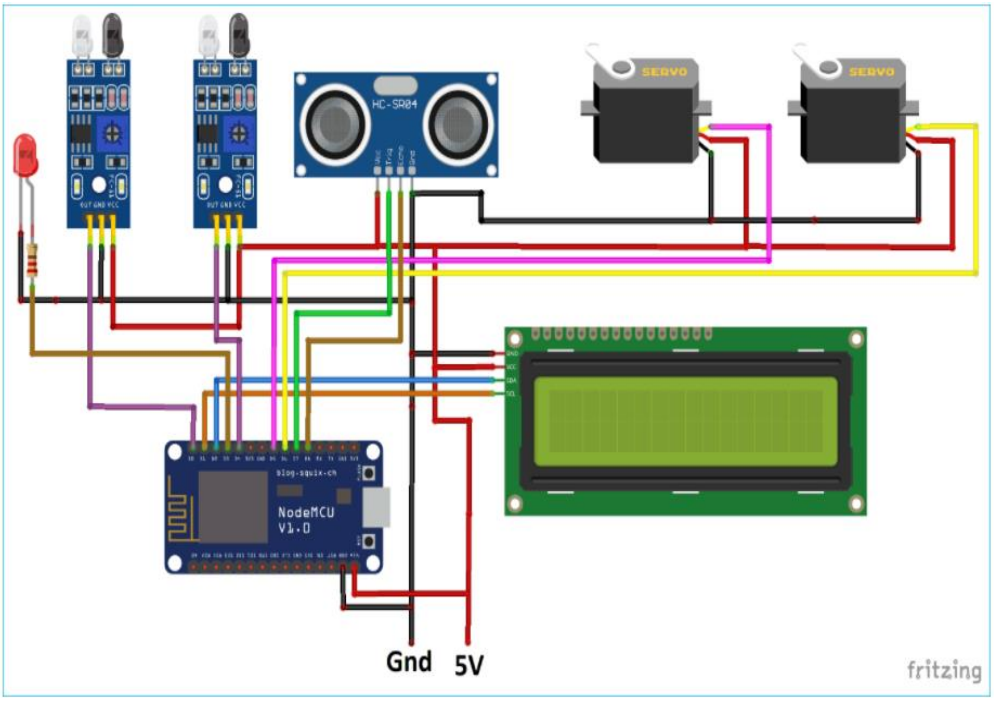
- Circuit diagram for this IoT based vehicle parking system is given below. It involves two IR sensor, two servo motors, one ultrasonic sensor and one 16x2 LCD.
- Here the ESP8266 will control the complete process and also send the parking availability information to Google Firebase so that it can be monitored from anywhere in the world over the internet. Two IR sensors are used at entry and exit gate to detect the presence of car and automatically open or close the gate. IR Sensor is used to detect any object by sending and receiving the IR rays, learn more about IR sensor here.
- Two servos will act as entry and exit gate and they rotate to open or close the gate. Finally an Ultrasonic sensor is used to detect if the parking slot is available or occupied and send the data to ESP8266 accordingly.
- Here the ESP8266 will control the complete process and also send the parking availability information to Google Firebase so that it can be monitored from anywhere in the world over the internet.
- Two IR sensors are used at entry and exit gate to detect the presence of car and automatically open or close the gate. IR Sensor is used to detect any object by sending and receiving the IR rays, learn more about IR sensor here. We have also covered a detailed tutorial on Arduino IR sensor interfacing, which you can check to learn more the working of IR sensor module and how to connect it with the Arduino uno.
- For programming NodeMCU, just plug the NodeMCU to Computer with a Micro USB Cable and open Arduino IDE. The libraries are required for I2C Display and Servo Motor. The LCD will display the availability of Parking Spaces and the Servo motors will be used to open and close the Entry and Exit gates. The *Wire.h* library will be used to interface LCD in i2c protocol. The Pins for I2C in the ESP8266 NodeMCU are D1 (SCL) and D2 (SDA)
- Take the Distance from Ultrasonic sensor. This will be used to detect the vehicle presence in the particular spot. First send the 2 microsecond pulse and then read the received pulse. Then convert it to the 'cm'.

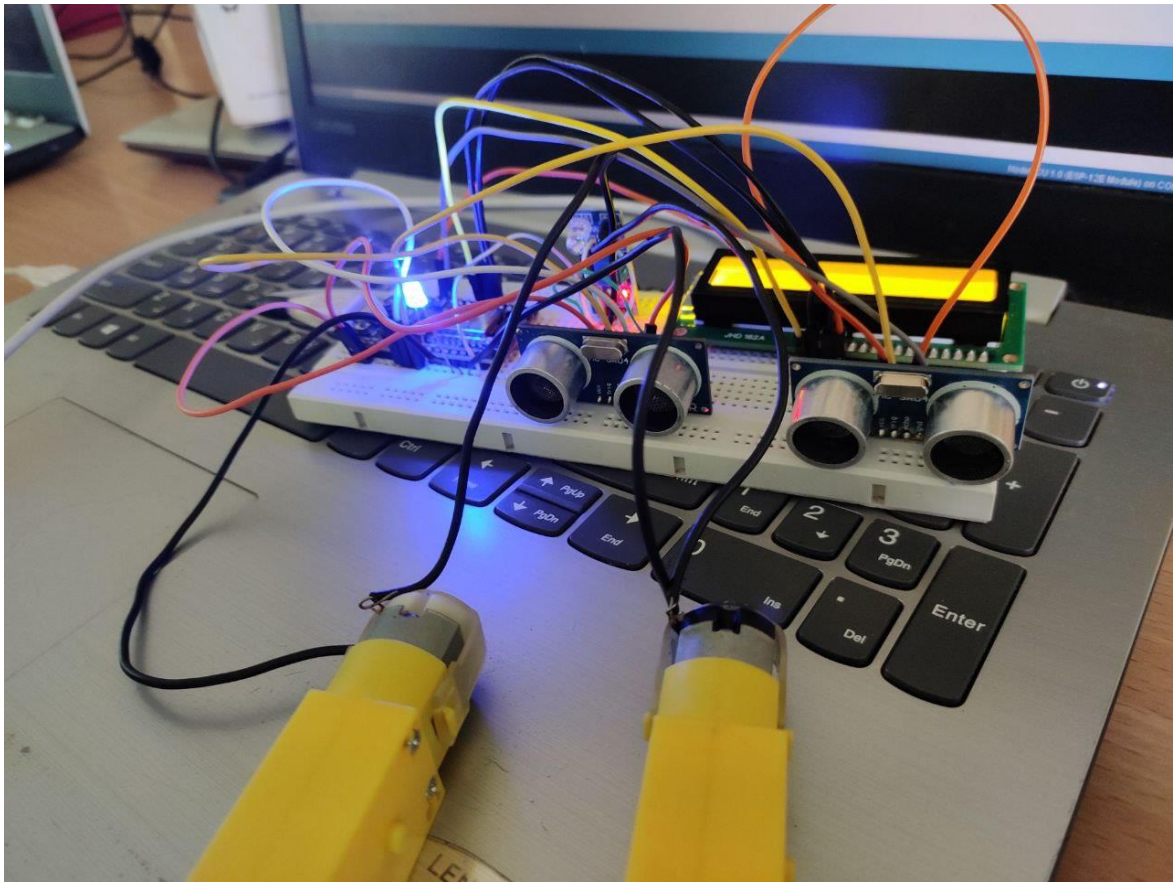
- Digitally read the IR sensor pin as entry sensor and check if it is high. If it is high then increment entry count and print it to 16x2 LCD display and also to serial monitor.
- Check if the car has come to the parking spot and if it has arrived then glow led giving the signal that the spot is full.
- A smart parking system is an architectural framework that comprises different application platforms integrated into embedded systems. For instance, reserved parking Smart Cities 2021, 4 625 spaces allow users to request the application layer, wherein the request will immediately be processed through a network layer. As a way of handling the user request, parking providers are expected to utilize the network layer to process the interaction with the transaction layer, as explained by Kayal and Perros. Finally, the transaction layer's consensus mechanism protocol and the individual parking provider update the distributed ledger.
- In RFID, the vehicle owner must register with the parking owner to receive the RGID tag. This tag is to be applied on the vehicle's windshield, so as the car passes through the entry of parking, the RFID scanner scans the label and deducts the amount for parking from the linked account. With this technology, the process speeds up. The entry gate opens when the sensor detects an RFID tag which allows the car inside the parking area. Simultaneously, the parking counter increments by one. Similarly, the exit gate opens, and the parking counter is decremented.
- IoT is the technology at the core of vehicle tracking platforms. Tools like GPS or OBD sensors help collect location data on a car or a fleet and monitor the occupancy of parking spaces. This information is transferred to the cloud gateway, processed and sent to the network server. The data will be presented to drivers and car company managers in the form of understandable, clear insights.
- Automatic parking systems help reduce the land use for parking and maximize the efficiency of space usage. An automated system is used to move cars up and down to the upper levels of the facility.

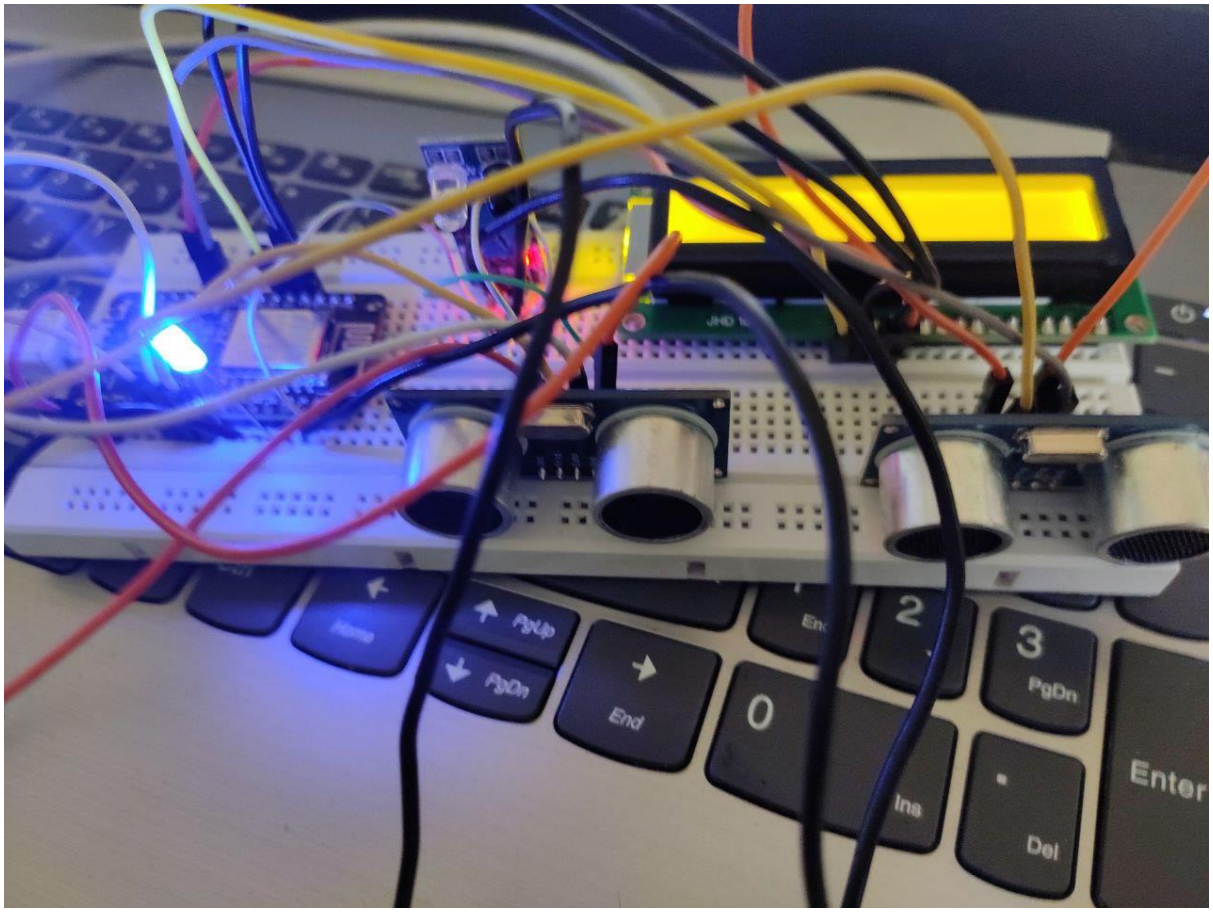
BLOCK DIAGRAM:-

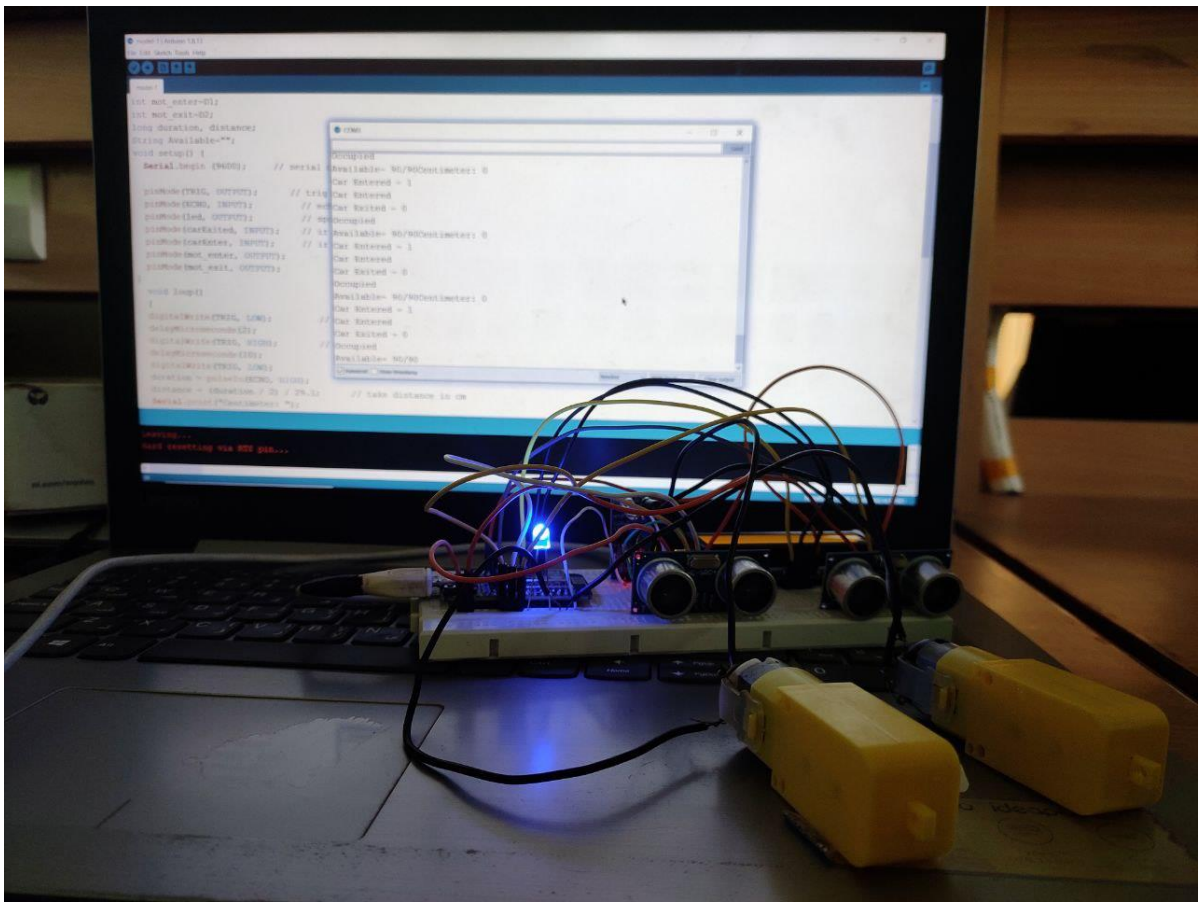


Circuit Diagram:-






PgUp
PgDn
End
0
Ins
2
3
PgDn
Del
Enter

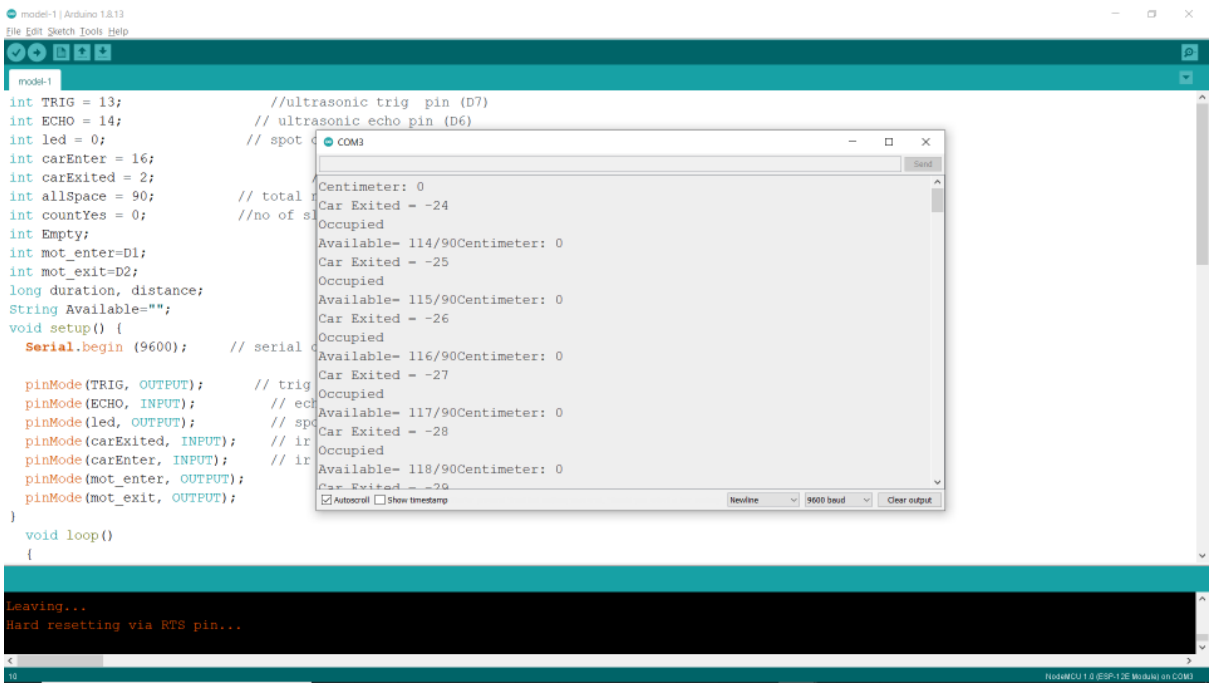


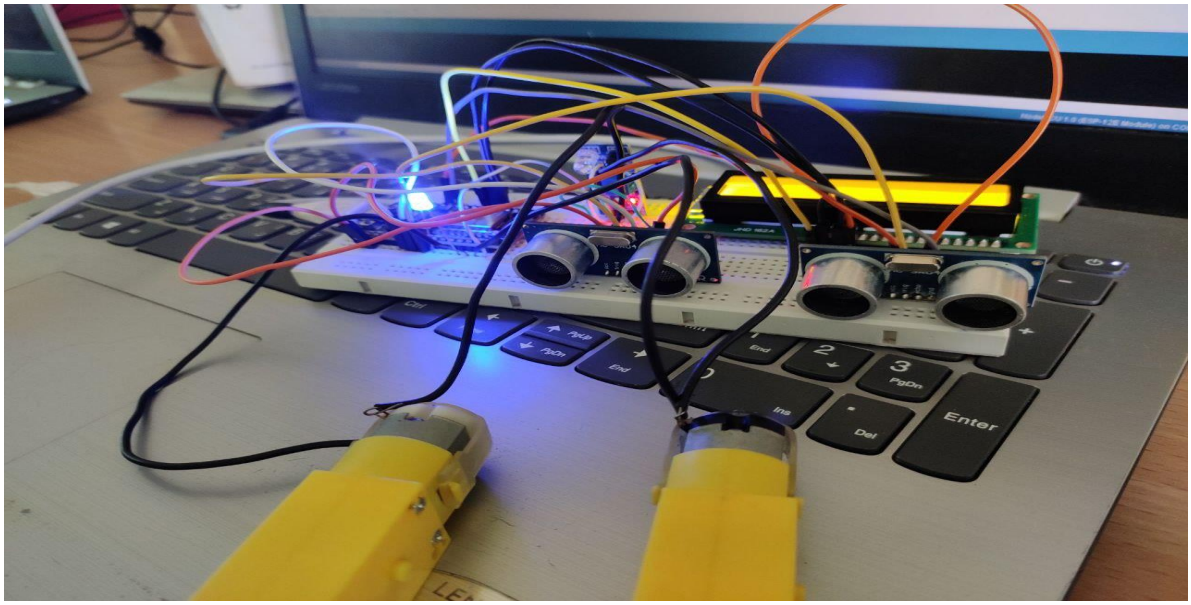
CODE:

```
int TRIG = 13;
int ECHO = 15;
int led = 0;
int carEnter = 16;
int carExited = 2;
int allSpace = 90;
int countYes = 0;
 int Empty;
long duration, distance;
String Available="";
void setup() {
  Serial.begin (9600);
pinMode(TRIG, OUTPUT);
pinMode(ECHO, INPUT);
pinMode(led, OUTPUT);
pinMode(carExited, INPUT);
pinMode(carEnter, INPUT);
}
void loop()
{
  digitalWrite(TRIG, LOW);
 delayMicroseconds(2);
 digitalWrite(TRIG, HIGH);
 delayMicroseconds(10);
 digitalWrite(TRIG, LOW);
 duration = pulseIn(ECHO, HIGH);
 distance = (duration / 2) / 29.1;
 Serial.print("Centimeter: ");
 Serial.println(distance);
 int carEntry = digitalRead(carEnter);
 if (carEntry == HIGH)
  {
    countYes++;
    Serial.print("Car Entered = " );
    Serial.println(countYes );
    delay(2000);
    //cd.clear();
  }
int carExit = digitalRead(carExited);
if (carExit == HIGH&&countYes<60) {
countYes--;
Serial.print("Car Exited = " );
Serial.println(countYes);
delay(2000);
  }
  if (distance < 6) {
       Serial.println("Occupied ");
    digitalWrite(led, HIGH);
```

```
  }
  if (distance > 6) {
    Serial.println("Available ");
    digitalWrite(led, LOW);
  }
  Empty = allSpace – countYes;
  Available = String("Available= ") + String(Empty) + String("/") + String(allSpace);
  Serial.print(Available);
}
```

Output&Results:-





Literature Survey:

Earlier researches and works of various authors on automatic accident detection system are discussed as follows:

Finding a suitable parking space has become a significant concern for people residing in metropolitan cities. The main reason behind this is a lack of parking space. The traditional parking methods cannot be used today as they are ineffective at utilizing space; hence it is vital to find alternative parking systems. Tracking of parking slots is also an essential factor, which can be done using an IoT system integrated with sensors. This paper explores the application of an IoT-based Car parking system. The paper also explores the use of various types of sensors for car parking systems.

The development and high growth of the Internet of Things (IoT) have improved quality of life and strengthened different areas in society. Many cities worldwide are looking forward to becoming smart. One of the most popular use cases in smart cities is the implementation of smart parking solutions, as they allow people to optimize time, reduce fuel consumption, and carbon dioxide emissions. Smart parking solutions have a defined architecture with particular components (sensors,communication protocols, and software solutions). Although there are only three components that compose a smart parking solution, it is important to mention that each component has many types that can be used in the deployment of these solutions. This paper identifies the most used types of every component and highlights usage trends in the established analysis period. It provides a complementary perspective and represents a very useful source of information. The scientific community could use this information to decide regarding the selection of types of components to implement a smart parking solution.

The Internet of Things (IoT) has come of age, and complex solutions can now be implemented seamlessly within urban governance and management frameworks and processes. For cities, growing rates of car ownership are rendering parking availability a challenge and lowering the quality of life through increased carbon emissions. The development of smart parking solutions is thus necessary to reduce the time spent looking for parking and to reduce greenhouse gas emissions.

The principal role of this research paper is to analyze smart parking solutions from a technical perspective, underlining the systems and sensors that are available, as documented in the literature. The review seeks to provide comprehensive insights into the building of smart parking solutions. A holistic survey of the current state of smart parking systems should incorporate the classification of such systems as big vehicular detection technologies. Finally, communication modules are presented with clarity.

Advantages:-

- Reduced traffic and pollution.
- Easy implementation and management
- Cost-effective solution
- Better parking experience
- Increased Safety and Security
- Increased Protection

Conclusion:-

The system benefits of smart parking go well beyond avoiding the needless circling of city blocks. It also enables cities to develop fully integrated multimodal intelligent transportation systems that don't rely on cars in the first place.

Developing smart parking solutions within a city requires data standardization and management; mobile phone integration; hardware and software innovation; and coordination among various stakeholders (on and off street parking facility owners, business owners, municipalities, transportation authorities, customers, and software developers).

These technical solutions and stakeholders are the same data structures and development groups integral to making a smart phone -enabled, multimodal, fully integrated transportation solution a reality. In effect, the technical enablers and multi-stakeholder coordination effort behind development of a local smart parking solution creates a launch pad toward full transportation system integration.

References:

https://ieeexplore.ieee.org/document/8090284

https://ieeexplore.ieee.org/document/7947615

https://ieeexplore.ieee.org/document/8537282

https://ieeexplore.ieee.org/document/8090284.

https://ieeexplore.ieee.org/document/9562776

https://circuitdigest.com/microcontroller-projects/iot-based-smart-parking-system-using-nodemcu

https://www.youtube.com/watch?v=Qoc6yoct51c

https://www.happiestminds.com/Insights/smart-parking/

https://mobidev.biz/blog/iot-based-smart-parking-system

https://www.smartparking.com/smartpark-system

https://www.researchgate.net/publication/355370817_Literature_Review_on_Parking_System