IoT based Car Parking and Monitoring System

SONARGAON UNIVERSITY (SU)

Supervised By

Md. Ferdous Khan

Coordinator & Lecturer
Department of Electrical and Electronic Engineering
Sonargaon University (SU)

Submitted By

Name	ID
Md. Al-Amin	**EEE 1802014015**
Mahabub Alom Monir	**EEE 1802014050**
Md. Sayed Hossain	**EEE 1802014094**
Sre. Uttam Kumar	**EEE 1802014116**
Md. Humaun Kabir	**EEE 1802014120**

Department of Electrical & Electronic Engineering (EEE)
Sonargaon University (SU)
147/I, Panthopath,
Dhaka-1215, Bangladesh.

Date of submission: February 2022

Declaration

It is declared hereby that this project paper or any part of it has not been submitted to anywhere else for the award of any degree.

……………………………… ………………………..

Md. Al-Amin **Md. Humaun Kabir**

………………………… ………………………….

Mahabub Alom Monir **Md. Sayed Hossain**

…………………………

Sre. Uttam Kumar

Under Supervision of

Md. Ferdous Khan
Coordinator & Lecturer
Department of Electrical and Electronic Engineering
Sonargaon University (SU)

Certification

I certify that I have read this project and that, in my opinion, it is fully adequate, in scope & quality as a dissertation for the degree of BSc in Engineering.

Supervisor

....

Md. Ferdous Khan

Lecturer, EEE

Sonargaon University (SU)

Department of Electrical and Electronic Engineering (EEE)

ACKNOWLEDGEMENTS

First and foremost, we would like to thank Almighty Allah for granting our capability and providing me health and an opportunity to finish this work.

We offer our sincerest gratitude to our supervisor, Md. Ferdous Khan, Lecturer, Department of Electrical & Electronic Engineering Sonargaon University for the patient guidance, encouragement and advice he has provided throughout my time as his student. We consider ourselves very fortunate for being able to work with a very considerate and encouraging lecturer like him.

We would also like to express our thanks to the respected **Vice-Chancellor of Su, Professor Dr. Md. Abul Bashar** and also thanks **Head of the SU**, Department of Electrical and Electronic Engineering **Professor Dr. M. Bashir Uddin** for providing me the opportunity to conduct our research in such an academic environment.

Above all, we wish to convey our heartfelt thanks and love towards our parents to whom we owe everything. We are forever indebted to our parents for their understanding, endless support and encouragement when it was most required.

Abstract

This project is the internet of thing base car parking space detection system. Car parking problem is the major problem in our urban cities. This car parking problem increase day to day because of continuous growing number of vehicles. It not just a major problem in our country's urban cities problem but also all over the world's urban cities problem. Internet of Thing base car parking space detection system can solve the problem. Internet of Thing base car parking space detection system that connect to via internet and physical device. Our IOT base car parking space detection system use sonar sensor for detecting the car in the parking area. For control we use Arduino Uno and also use NodeMCU for find the information in our physical device via internet, by the use of blynk system we collect the information about the parking space are available or not. IOT car parking space detection system helps the people to find the parking space. In IOT base car parking space diction system every user has a unique ID and password. In this system will solve the car parking problem.

TABLE OF CONTENTS

Chapter-1 Introduction

Chapter-2 Theory of the project

Chapter-3 Design and Fabrication

Chapter-4 Result and Discussion

CHAPTER-5 Conclusion and Future Works

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Introduction

IOT base vehicle parking area detection device is most important nature in our daily lives. IOT system less human efforts, labours, time and errors. With improve the contemporary technology. Smart phones have end up a necessity for each and every person on this planet. IOT base smart parking house detection device helps to display vehicle-parking slot. IOT base auto parking area detection helps to control parking disaster among vehicles. When they are find, the parking slot at the same time that capability it helps in synchronized parking. This vehicle parking space detection gadget related to the parking vicinity and provide information by means of net about vehicle parking slot. IOT base vehicle parking area detection machine managed the parking slot. It helps the vehicle customers to discover free space in parking slot. It store person time as nicely as their fuel. It assist to obtain parking slot in urban cities, which is very hares. People waste time and fuel in looking out for parking space. Smart parking house detection machine gives data about parking slot. A sonar sensor is use at every slot in parking areas. It tells the parking area availability. Information about the empty and used slot sends over the Google firebase with the aid of internet. Day through day growing vehicle, so it is tough to discover a parking place in urban cities. A short quantity of time and it wasted a lot of fuel in looking an empty parking slot. Overcome from this serious problem, we put in force a IOT base car parking area detection system the place it can tell to person that parking house is reachable or not. If parking slot is full, in any other case want search a new parking region rather of go and search for parking slot. Our city cities, clever auto parking house detection machine will become foremost hassle with upward jostle numbers of vehicles. Generally, it takes more than 5 minutes to discover parking slot area. It will manage the system using IOT based cellular application. Here we used Google firebase to gather facts about parking slot area. Our approach is primarily based on Arduino Nano, Arduino runs with C++ code. We write easy C++ code via Arduino utility and immediately put the Arduino system. It works in accordance to code system keeps song of variety of vehicles in parking slot or not in parking building.

Problem Statement

Day to day population is increase, number of vehicles increases and due to unmanaged parking it leads to many problems. In urban cities, people faces difficulties as increasing number of vehicles creates congestion in roads. so, our valuable time killed by the unmanaged car parking. Lots of fuel burn and creates traffic problems. But our urban buildings car-parking slot is wastage of free space. An emergency patient cannot go to the hospital because of this problem. Our footpath is block and people cannot walk easily on footpath of this problem. This problem also cause some other difficulties.

Objectives

The followings are the objectives of the project to ensure it meets the aim.

- To study about the IOT base car parking space detection system.
- To familiar with the equipment that used in this process.
- Utilize the application in recent circumstances.
- To improve the system with respect of time and need.

Possible outcome

IOT base vehicle parking area detection is one of the most time-honoured and rapid developing smart town solutions across the world. Universities, Airports, City garages and Shopping centres etc. Are just a few subsistence that have begun to recognize the amazing advantages of IOT base vehicle parking area detection system. Many earning source are feasible with IOT base smart parking system. Traffic glide upward jostle to countless motors are required to power around in search of an open parking space, so it decreased site visitors problem and decrease cities accident problem

Methodology

To prepare the project information are collected and analysed form various source and our group discussion among which the following are notable.

- Main information collected form some online research papers.
- Other diagram and short note collected form journal papers.
- Some key information collected from online resources.
- Some information collected from our group discussion.
- And some information collected from our teacher

CHAPTER II

THEORY OF THE PROJECT

Introduction

This chapter includes the total over view of the device. In this chapter we have followed-up the theory of Arduino UNO, Node MCU, IR Obstacle sensor and other components. Here we can know that the total system overview of the projects. And we will also know that how the equipment are working with each other.

Theory

The system architecture of the automatic output appliance can be divided into 3 main Modules. They are:

1. CPU
2. Communication system
3. Control system

The system of IOT base car parking that make parking easier and more sustainable has increased within the past few years. The ability to protect the IOT base car parking has been a latest challenge. The system that provides this ability through the use of efficient and reliable methods of our car parking system. Wireless sensor networking technologies are readily available in our mobile phone devices. It help the car user to get a better yield to find the car parking space and help the traffic system of the country. NodeMCU module connect the system with blynk cloud by the email ID and password via internet. When the car users need to free space for car parking to active the mobile internet and connect the blynk than the car user see the free space for his car parking by the wireless communication. All these devices of the system are connected to the Arduino UNO. NodeMCU is used for communication purpose for the user to system.

Microcontroller

A microcontroller is a compact integrated circuit designed to govern a specific operation in an embedded system. A typical microcontroller includes a processor, memory and input/output (I/O) peripherals on a single chip. Sometimes referred to as an embedded controller or microcontroller unit (MCU), microcontrollers are found in vehicles, robots, office machines, medical devices, mobile radio transceivers, vending machines and home appliances among other devices. Microcontroller is a single chip microcomputer made through VLSI fabrication.

Arduino UNO

Arduino Uno is one of the microcontroller boards manufactured by the Arduino and it is a microcontroller board based on Atmel’s ATmega328P microcontroller. “Uno” means one in Italian and the Uno board is the latest in a series of USB (Universal Serial Bus) Arduino boards which is the reference model for the Arduino platform. The Arduino Uno board has a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, a reset button, 6 analog inputs and 14 digital input/output pins (of which 6 can be used as PWM outputs). It uses the Atmega16U2 programmed as a USB – to - serial converter instead of FTDI USB – to – serial driver chip which was used in all the pre-ceding boards. The board has 32 KB flash memory of which 0.5 KB is used by boot-loader, 2 KB of SRAM, 1 KB of EEPROM and 16 MHz clock speed.

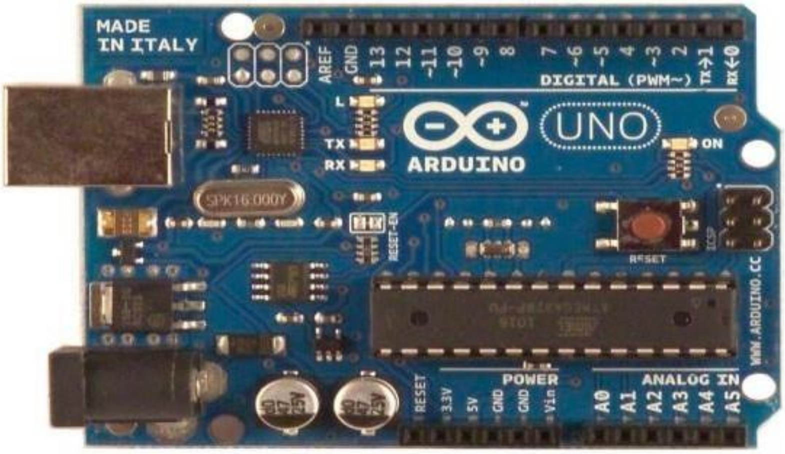


Fig 2.1: Arduino UNO

Reprinted from the Arduino Board Uno Figure 1 shows the Arduino Uno Board manufactured by the Arduino in Italy. It can be powered via a USB connection or with an external power supply.

As can be seen in figure 1, pins A0 to A5 are the analog input pins, pins 0 to 13 are 14 digital input/output pins and the pins with a “~” sign can be used as digital pins PWM o can be used as input or output pins by selecting the mode by using the function Pin-Mode() and then using the function digital Read() or digital Write() according to the ne-cessity. Pins 0(RX) and 1(TX) are

used for serial communication while pins 10(SS), 11(MOSI), 12(MISO) and 13(SCK) are used for SPI (Serial Peripheral Interface) communication. In addition to pin 0 and 1, a Software Serial library allows serial communication on any of the Uno’s digital pin.

The microcontroller is a low-power CMOS (Complementary Metal Oxide Semiconductor) 8-bit microcontroller based on the AVR enhanced RISC (Reduced Instruction Set Computer) architecture. The powerful execution of instructions in a single clock cycle leads to the achievement of 1 MIPS per MHz throughputs allowing the designer to optimize power consumption versus processing speed.

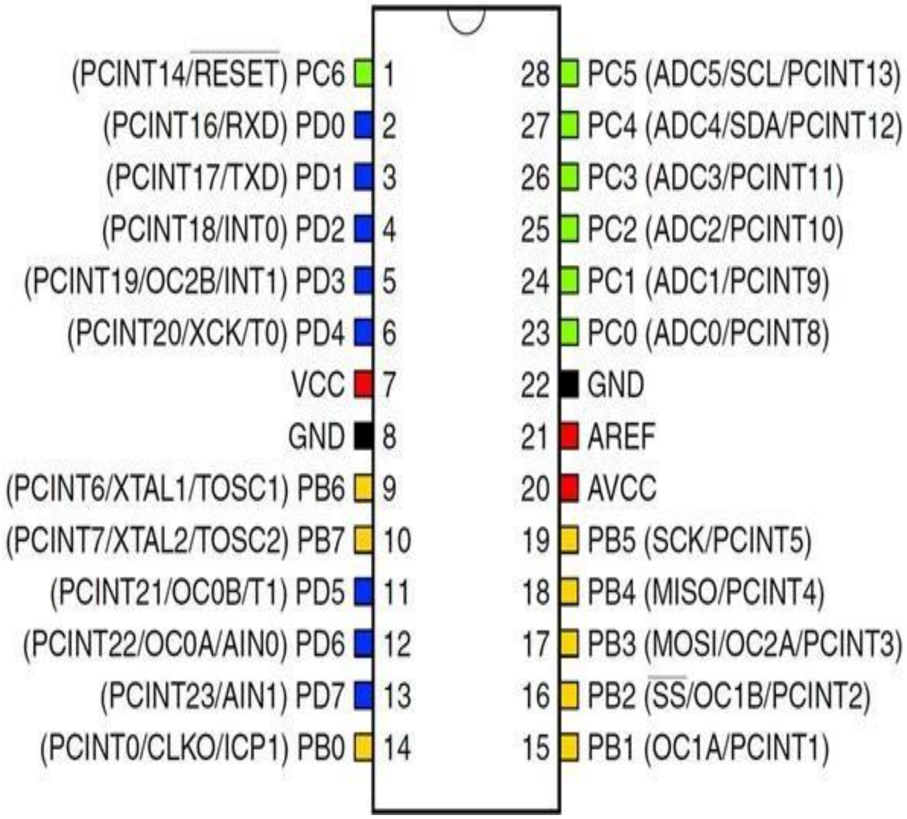


Fig 2.2: ATmega328P Microcontroller

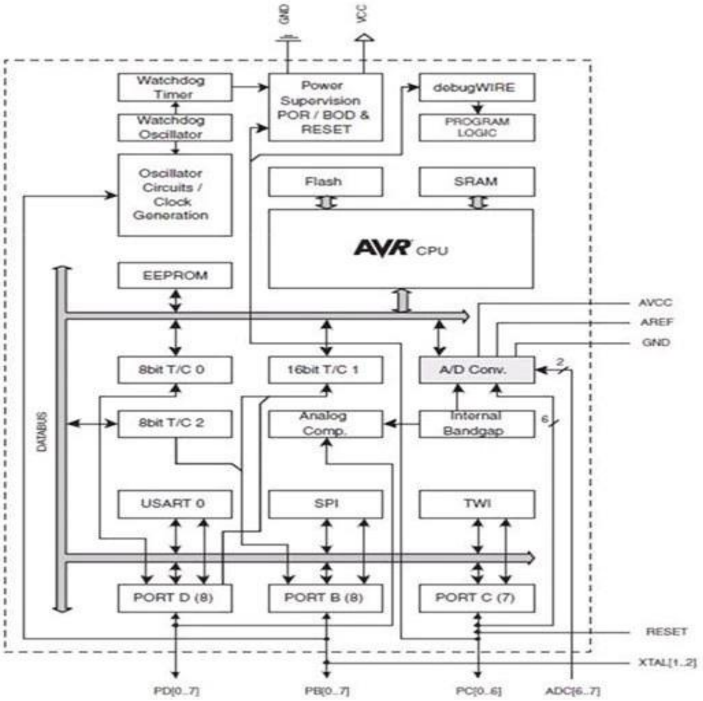


Fig 2.3: ATmega328 Microcontroller Architecture

Reprinted from Datasheet of ATMega328 the internal architecture of the microcontroller is shown in Fig 2.3.3 (02). The central processing unit (CPU) is the brain of the microcontroller which controls the execution of the program. The MCU (Microcontroller unit) consists of 4K/8K bytes of in-system programmable flash with read-while-write capabilities, 256/412/1K bytes EEPROM along with the 512/1K/2K bytes of SRAM. Along with this, the MCU consists of many other features

- 23 general purpose I/O lines and 32general purpose working registers
- flexible timer/counters with compare modes, internal and external interrupts and a serial programmable USART
- A byte-oriented 2-wire serial interface, an SPI serial port, a 6-channel 10-bit ADC (8 channels in TQFP and QFN/MLF packages), a programmable watch-dog timer with an internal oscillator and 5 software -selectable power saving modes.

The five, software selectable, power saving modes are idle mode, Power-down mode, Power-save mode, ADC Noise Reduction mode and the Standby mode. As mentioned in section 2.1.2, the CPU is the brain of the microcontroller which controls the execution of the program.

Therefore the CPU is able to access the memories, perform calculations, control peripherals and handle interrupts. The AVR uses the Harvard architecture with separate memories and buses for program and data to maximize the performance as well as the parallelism. The principle of execution of instructions in the program memory is the single-level pipelining. The concept of pre-fetching the next instruction while executing one instruction enables the instructions to be executed in every clock cycle and the program memory is in the System Reprogrammable Flash memory.

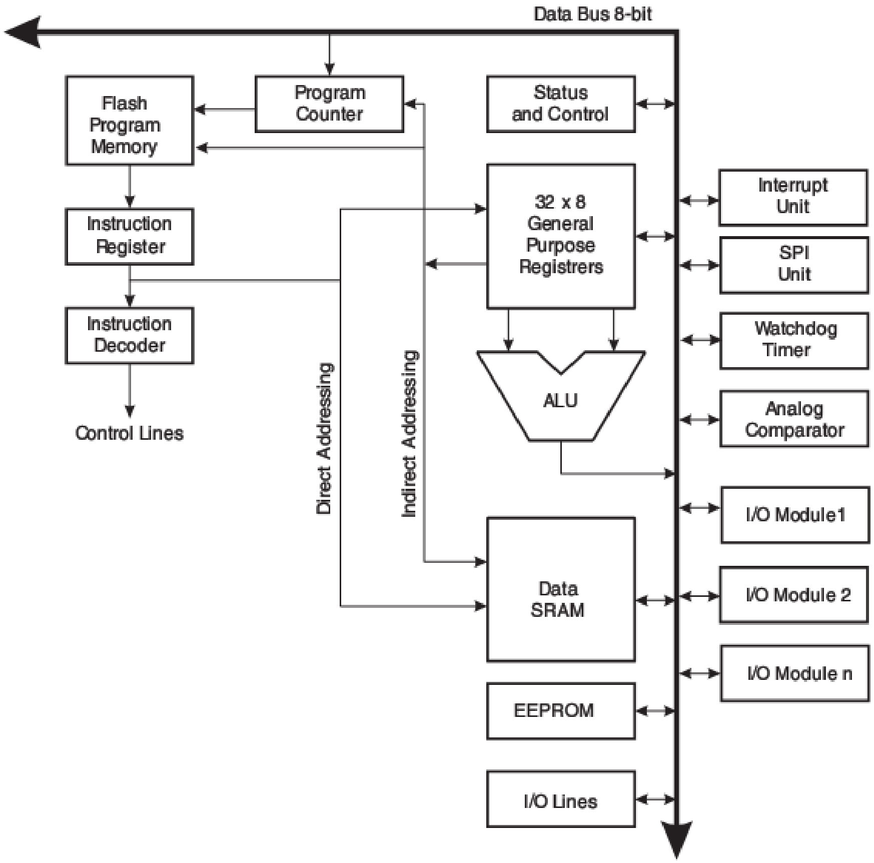


Fig 2.4: Block diagram of the AVR CPU Core architecture

Reprinted from Datasheet of ATMega328, The block diagram of AVR CPU Core architecture is shown in figure 3. The fast-access Register File contains 32 x 8 bit general-purpose working registers with a single cycle access time which results in a single -cycle ALU operation. The arithmetic and logical operations between the registers or between the constant and a register are supported by the ALU. The status register is updated to reflect information about the result of the operation after an arithmetic operation. The boot program section and the application program section are the two main sections of the program flash memory. Stack stores the return address of the program counter during the interrupts and subroutine calls which is allocated in the general data SRAM. The size of the stack is limited by the total size and usage of the SRAM. The data SRAM is accessible through five different addressing modes supported in the AVR architecture while the stack pointer is read/write accessible in the I/O space. The memory spaces in the AVR architecture are all linear and regular memory maps.

Specifications

Microcontroller: Atmel ATmega328

Operating Voltage (logic level):5 V

Input Voltage (recommended):7-12 V

Input Voltage (limits):6-20 V

Digital I/O Pins: 14 (of which 6 provide PWM output)

Analog Input Pins: 8

DC Current per I/O Pin: 40 mA

Flash Memory: 32 KB (of which 2KB used by boot loader)

SRAM : 2 KB

EEPROM: 1 KB

Clock Speed: 16 MHz

Dimensions: 0.70” x 1.70”

Features

• Automatic reset during program download

• Power OK blue LED

• Green (TX), red (RX) and orange (L) LED

• Auto sensing/switching power input

• Small mini-B USB for programming and serial monitor

• ICSP header for direct program download

• Standard 0.1 spacing DIP (breadboard friendly)

• Manual reset switch

Microcontroller IC ATmega328p

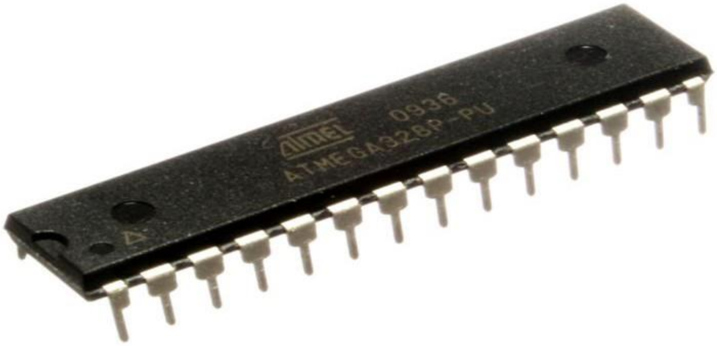


Fig 2.5: Microcontroller IC ATmega 328p.

The high-performance Microchip picoPower 8-bit AVR RISC-based microcontroller combines 32KB ISP flash memory with read-while-write capabilities, 1024B EEPROM, 2KB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, a 6-channel 10-bit A/D converter (8-channels in TQFP and QFN/MLF packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-5.5 volts. By executing powerful instructions in a single clock cycle, the device achieves throughputs approaching 1 MIPS per MHz, balancing power consumption and processing speed.

Power supply, inputs and outputs

Either Arduino is supplied with USB connection or with an external power supply (recommended with 7-12V), outputs are going to have a continuous voltage due to voltage regulators and stabilization capacitors present on the board. These power supply pins are:

- **VIN:** it is the input power supply that will have the same voltage that we are supplying the Arduino with the external power supply

- **5V:** power supply of 5V, this voltage may come from VIN pin anda voltage regulator or from the USB connection.
- **3.3V:** power supply that will provide 3.3V generated by an internal regulator, with a maximum current of 50 mA.
- **GND:** grounding pins

Digital inputs and outputs

Each of the 14 digital pins can be used as an input or output. Besides, each pincan supply or receive a maximum of 40 mA and has a pull-p resistance from 20 to50 kOhm. In addition, some pins have specialized functions such as:

- Pin 0 (RX) and 1 (TX). They are used to receive (RX) and transmit (TX)in TTL serial communication.
- Pin 2 and 3. External interruptions. Pins in charge of interrupting the sequential program established by the user.
- Pin 3, 5, 6, 9, 10 and 11. PWM (pulse width modulation). They form 8output bits with PWM with the function analog Write ().
- Pin 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI(Serial peripheral interface) communication.
- Pin 13. LED. There is a LED connected to the digital pin 13. When thepin value is HIGH, the LED is on, and when the value is LOW, the LED is off.

Analog inputs

Arduino UNO has 6 analogical inputs, from A0 to A5, and each one offers abresolution of 10 bits (1024 states). By default, there is a voltage of 5V, but that rangecan be modified using the pin AREF and using the function analogReference(), where the user introduces the DC external signal he wants to use as reference.

NodeMCU ESP8266 Microcontroller

The ESP8266 is a low-cost Wi-Fi microchip with full TCP/IP stack and microcontroller capability produced by manufacturer Espressif Systems. This small module allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections using Hayes-style commands. However, at first there was almost no English-language documentation on the chip and the commands it accepted. The ESP8285 is an ESP8266 with 1 MiB of built-in flash, allowing for single-chip devices capable of connecting to Wi-Fi.

Advantages of IOT Microcontrollers:

The main advantages of microcontrollers are given.

a) Microcontrollers act as a microcomputer without any digital parts.

b) As the higher integration inside microcontroller reduces cost and size of the system.

c) Usage of a microcontroller is simple, easy to troubleshoot and system maintaining.

d) Most of the pins are programmable by the user for performing different functions.

e) Easily interface additional RAM, ROM, I/O ports.

f) Low time required for performing operations.

Microcontrollers Applications:

Some basic applications of a microcontroller are given below.

1. Used in biomedical instruments.
2. Widely used in communication systems.
3. Used as a peripheral controller in PC.
4. Used in robotics.
5. Used in automobile fields.

NodeMCU Arduino Compatible ESP8266:

This NodeMCU development board is based on the ESP8266, which is a Wi-Fi communication IC built by Espressif. At its core is a microcontroller that runs at a blazing 80 MHz and includes a built-in TCP/IP stack and transceiver, which allows for Wi-Fi communication. This low-cost option can, not only, add Wi-Fi capability to your next project, it can run it. This board is a standalone microcontroller development board that can be easily programmed using the Arduino IDE. Some of the peripherals provided by that ESP8266 include 9 GPIOs, 1 analog input, UART, SPI, and TWI/I2C, and Wi-Fi. It operates at a 3.3V logic level and it can be powered from the USB voltage or AC adapter input. The NodeMCU D1 uses the ESP 8266 microcontroller that is 2 x faster than an Uno, has 160Kbs of Ram compared to the 2K of an Uno and a 100x the amount of Flash memory! And each I/O pin is interruptible!

Technical Specs:

- Supply Voltage (Vin): 5 – 15V
- Logic Voltage: 3.3V
- Wi-Fi, UART, I2C, SPI
- Digital I/Os: 9
- Analog Inputs: 1

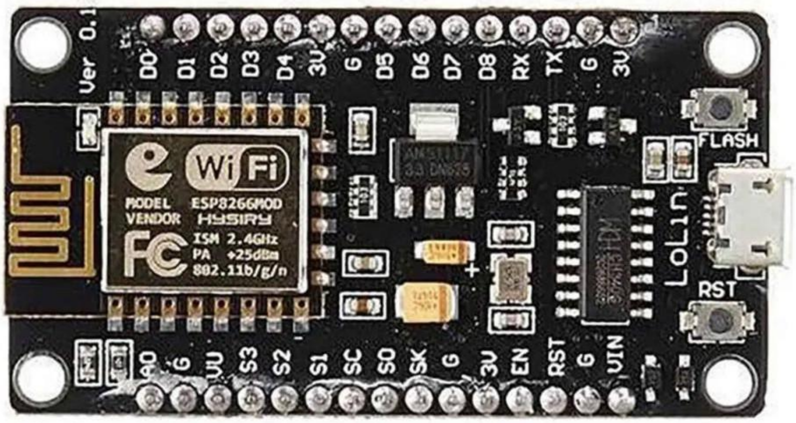


Fig 2.6: NodeMCU microcontroller

NodeMCU Wi-Fi ESP8266 Board

- An Arduino Nano Compatible Wi-Fi board based on ESP8266EX.
- NodeMCU® Wi-Fi ESP8266 Development Board is programmable via Arduino IDE.
- 11 digital input/output pins, all pins have interrupt/pwm/I2C/one-wire supported (except D0)
- 1 analog input (3.3V max input)
- A Micro USB connection
- A power jack, 9-24V power input.
- Compatible with Arduino
- Compatible with nodemcu

Table 2.3.3.1**:** Pin mapping of ESP8266

Pin	Function	ESP-8266 Pin	Microcontroller	ESP-8266EX
TX	TXD	TXD	Operating Voltage	3.3V
RX	RXD	RXD	Digital I/O Pins	11
A0	Analog input, max 3.3V input	A0	Analog Input Pins	1
D0	IO	GPIO16	Clock Speed	80MHz/160MHz
D1	IO, SCL	GPIO5	FLash	4M bytes
D2	IO, SDA	GPIO4	Length	68.6mm
D3	IO,Pull-up	GPIO0	Width	53.4mm
D4	IO,pull-up, BUILTIN_LED	GPIO2	Weight	25g
D5	IO, SCK	GPIO14		
D6	IO, MISO	GPIO12		
D7	IO, MOSI	GPIO13		
D8	IO,pull-down, SS	GPIO15		
G	Gound	GND		
5V	5V	-		
3V3	3.3V	3.3V		
RST	Reset	RST		

*All IO have interrupt/pwm/I2C/one-wire supported (except D0)

Programming:

The NodeMCU has a micro USB for auto programming, alsowe can program it using OTA.

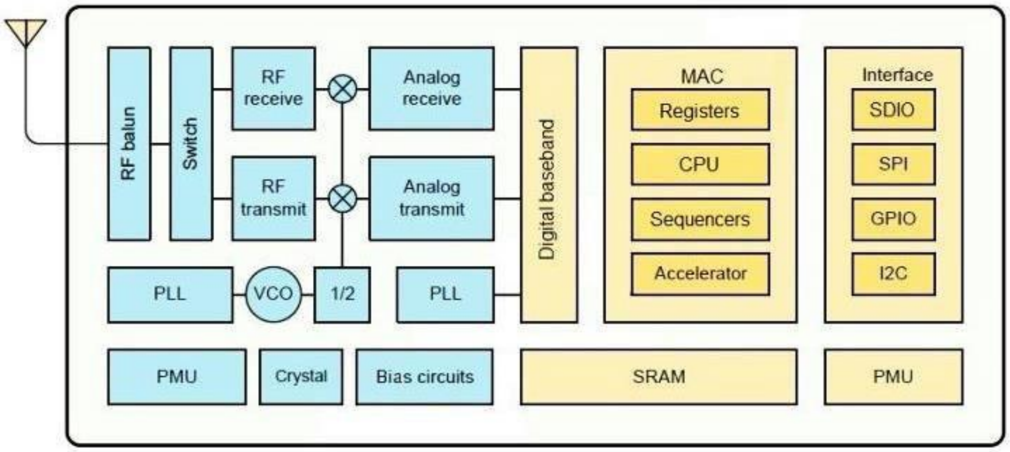


Fig 2.7: NodeMCU microcontroller pin out diagram

Blynk application Fundamentals

There are many apps to control the Wi-Fi communication with our smart phones. In our project we have used Blynk app for controlling our project using our smart phone. Blynk app for iOS and Android is the easiest way to build our own mobile app that work with the hardware of our choice. Blynk Library is an extension that runs on top of the hardware application. It handles all the connection routines and data exchange between our hardware, Blynk Cloud, and our project.

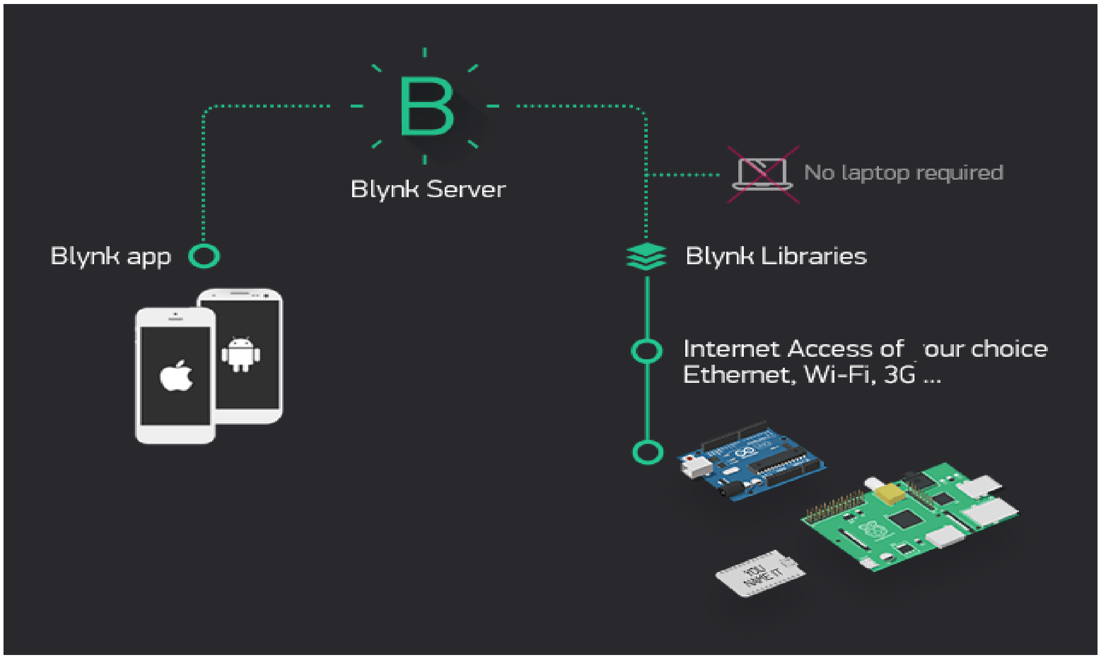


Fig 2.8: Blynk application communication feature

Major features

“Blynk app” 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 our projects using various widgets.

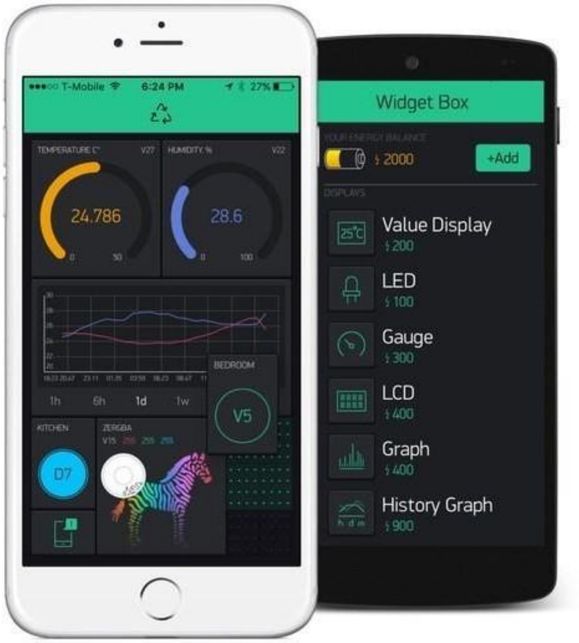


Fig 2.9: Blynk application widget box

IR Obstacle Sensor

Infrared technology addresses a wide variety of wireless applications. The main areas are sensing and remote controls. In the electromagnetic spectrum, the infrared portion is divided into three regions: near infrared region, mid infrared region and far infrared region.

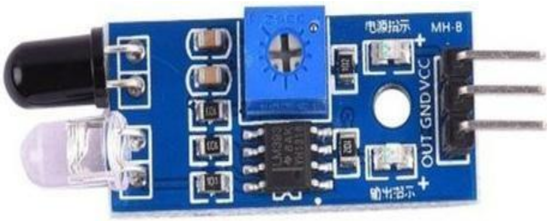


Fig 2.10: Optical Sensor

The wavelengths of these regions and their applications are shown below.

- Near infrared region — 700 nm to 1400 nm — IR sensors, fiber optic
- Mid infrared region — 1400 nm to 3000 nm — Heat sensing
- Far infrared region — 3000 nm to 1 mm — Thermal imaging

The frequency range of infrared is higher than microwave and lesser than visible light.

For optical sensing and optical communication, photo optics technologies are used in the near infrared region as the light is less complex than RF when implemented as a source of signal. Optical wireless communication is done with IR data transmission for short range applications.

An infrared sensor emits and/or detects infrared radiation to sense its surroundings. The working of any Infrared sensor is governed by three laws: Planck's Radiation law, Stephen – Boltzmann law and Wien's Displacement law.

Planck's law states that "every object emits radiation at a temperature not equal to 0^0K". Stephen – Boltzmann law states that "at all wavelengths, the total energy emitted by a black body is proportional to the fourth power of the absolute temperature". According to Wien's Displacement law, "the radiation curve of a black body for different temperatures will reach its peak at a wavelength inversely proportional to the temperature".

The basic concept of an Infrared Sensor which is used as Obstacle detector is to transmit an infrared signal, this infrared signal bounces from the surface of an object and the signal is received at the infrared receiver.

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 can be 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.

Optical lenses made of Quartz, Germanium and Silicon are used to focus the infrared radiation. Infrared receivers can be photodiodes, phototransistors etc. some important specifications of infrared receivers are photosensitivity, detectivity and noise equivalent power. Signal processing is done by amplifiers as the output of infrared detector is very small.

Transformer

A transformer is a passive electrical device that transfers electrical energy between two or more circuits. A varying current in one coil of the transformer produces a varying magnetic flux, which, in turn, induces a varying electromotive force across a second coil wound around the same core. Electrical energy can be transferred between the two coils, without a metallic connection between the two circuits. Faraday's law of induction discovered in 1831 described the induced voltage effect in any coil due to changing magnetic flux encircled by the coil.

Transformers are used for increasing or decreasing the alternating voltages in electric power applications, and for coupling the stages of signal processing circuits.

Basically, transformer are two types.

1. Step up transformer
2. Step down transformer

We have used step down transformer as required for our project. Which is 220v to 9v 600ma step down transformer.

Figure 2.11: step down transformer.

Diode

A diode is a specialized electronic component with two electrodes called the anode and the cathode. Most diodes are made with semiconductor materials such as silicon, germanium, or selenium. Some diodes are comprised of metal electrodes in a chamber evacuated or filled with a pure elemental gas at low pressure. Here we used converted AC into DC using a bridge-wave rectifier that consists of four diodes

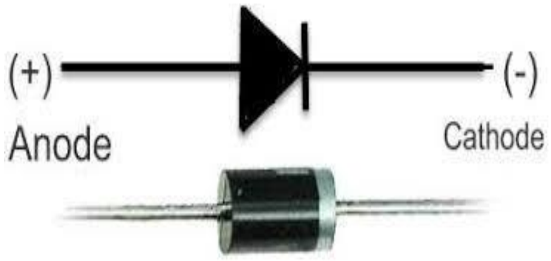


Fig 2.12: Diode.

Full wave rectifier

A Full wave rectifier is a circuit arrangement which makes use of both half cycles of input alternating current (AC) and converts them to direct current (DC). ... This arrangement is known as a Bridge Rectifier. It uses the entire AC wave (Both positive and negative sections). Each diode uses 0.7v when conducting and there are always two diodes conducting.

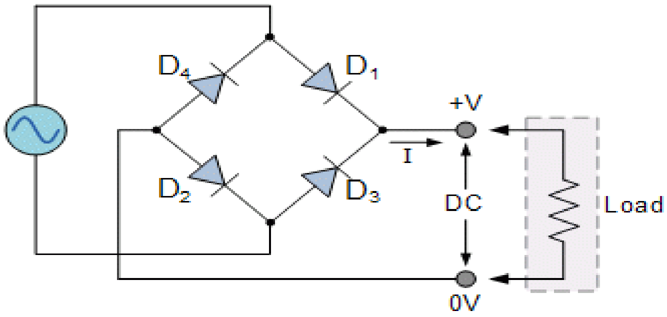


Fig 2.13: Full wave rectifier is a circuit diagram

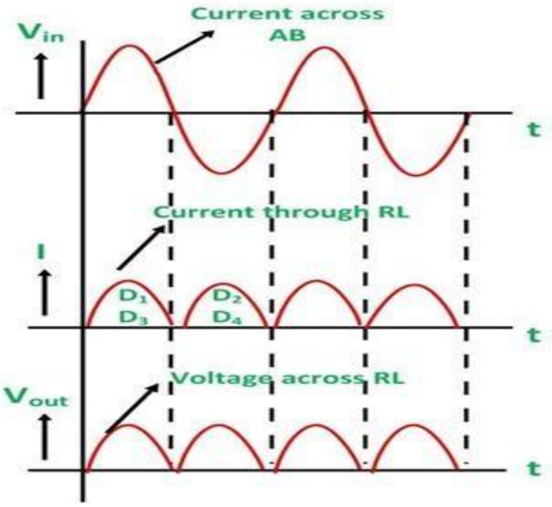


Figure 2.14: Full wave rectifier wave from

LCD Display

A liquid-crystal display (LCD) is a flat-panel display or other electronically modulated optical device that uses the light-modulating properties of liquid crystals combined with polarizers. Liquid crystals do not emit light directly, instead using a backlight or reflector to produce images in color or monochrome. LCDs are available to display arbitrary images (as in a general-purpose computer display) or fixed images with low information content, which can be displayed or hidden, such as preset words, digits, and seven-segment displays, as in a digital clock. They use the same basic technology, except that arbitrary images are made from a matrix of small pixels, while other displays have larger elements. LCDs can either be normally on (positive) or off (negative), depending on the polarizer arrangement. For example, a character positive LCD with a backlight will have black lettering on a background that is the color of the backlight, and a character negative LCD will have a black background with the letters being of the same color as the backlight.

Optical filters are added to white on blue LCDs to give them their characteristic appearance**.**

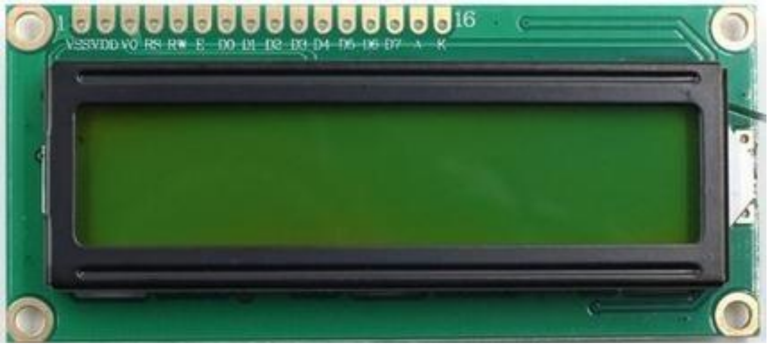


Fig 2.15: LCD Display

Buck Converter Module

A buck converter (step-down converter) is a DC-to-DC power converter, which steps down voltage from its input to its output. The basic operation of the buck converter has the current in an inductor controlled by two switches. In the idealised converter, all the components are considered to be perfect. Specifically, the switch and the diode have zero voltage drop when on and zero current flow when off, and the inductor has zero series resistance. Further, it is assumed that the input and output voltages do not change over the course of a cycle

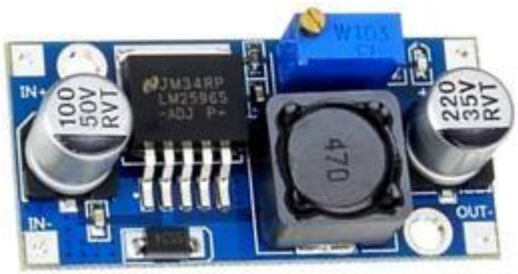


Fig 2.16: DC-DC Buck Converter

CHAPTER III

DESIGN & FABRICATION

Introduction

The implementation of the project is done after simulating the schematic circuit properly. In this chapter, the function of every section in the circuit is investigated with coding and also with physical outlook. To give a proper and clear concept about the operation the entire system is separated into different parts. In this chapter also discusses the working process of the circuits used in various parts with following chart, block diagram and corresponding designed diagram.

Block and Circuit Diagram

This project is basically based on both the arduino uno and NodeMCU. In this project we control our microcontroller by input of IR Sensor. Project contains one motor for control the parking gate.

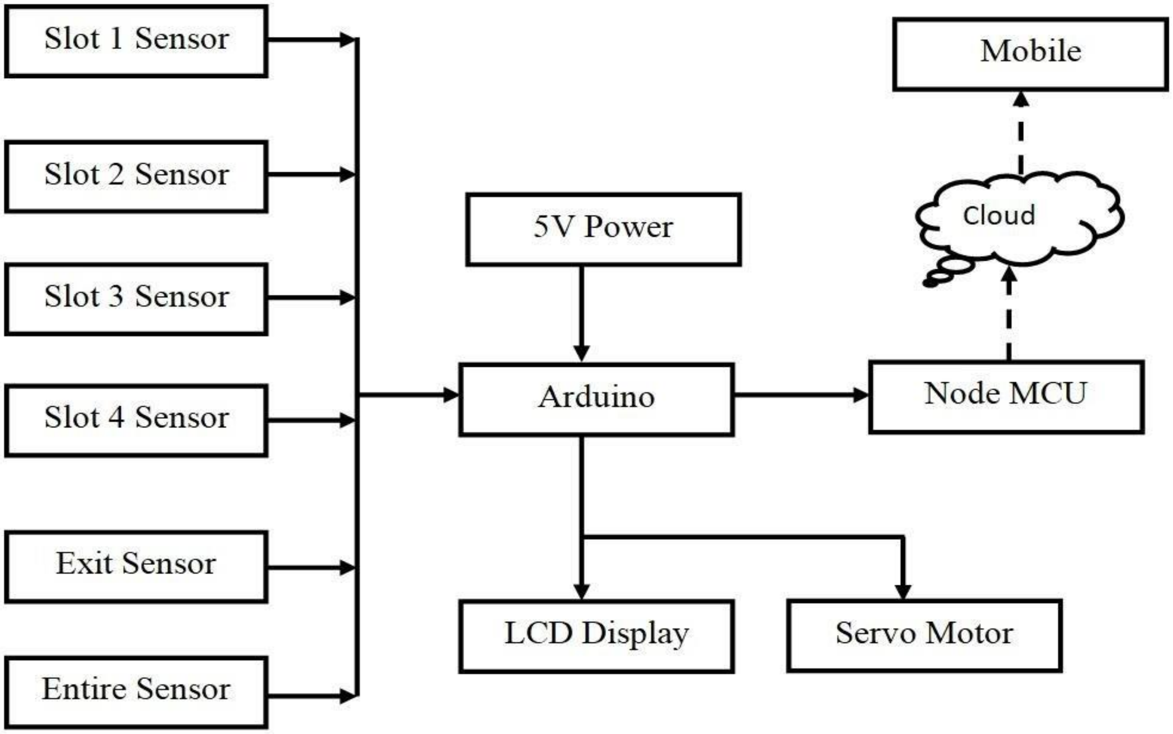


Fig 3.1: Block diagram of project

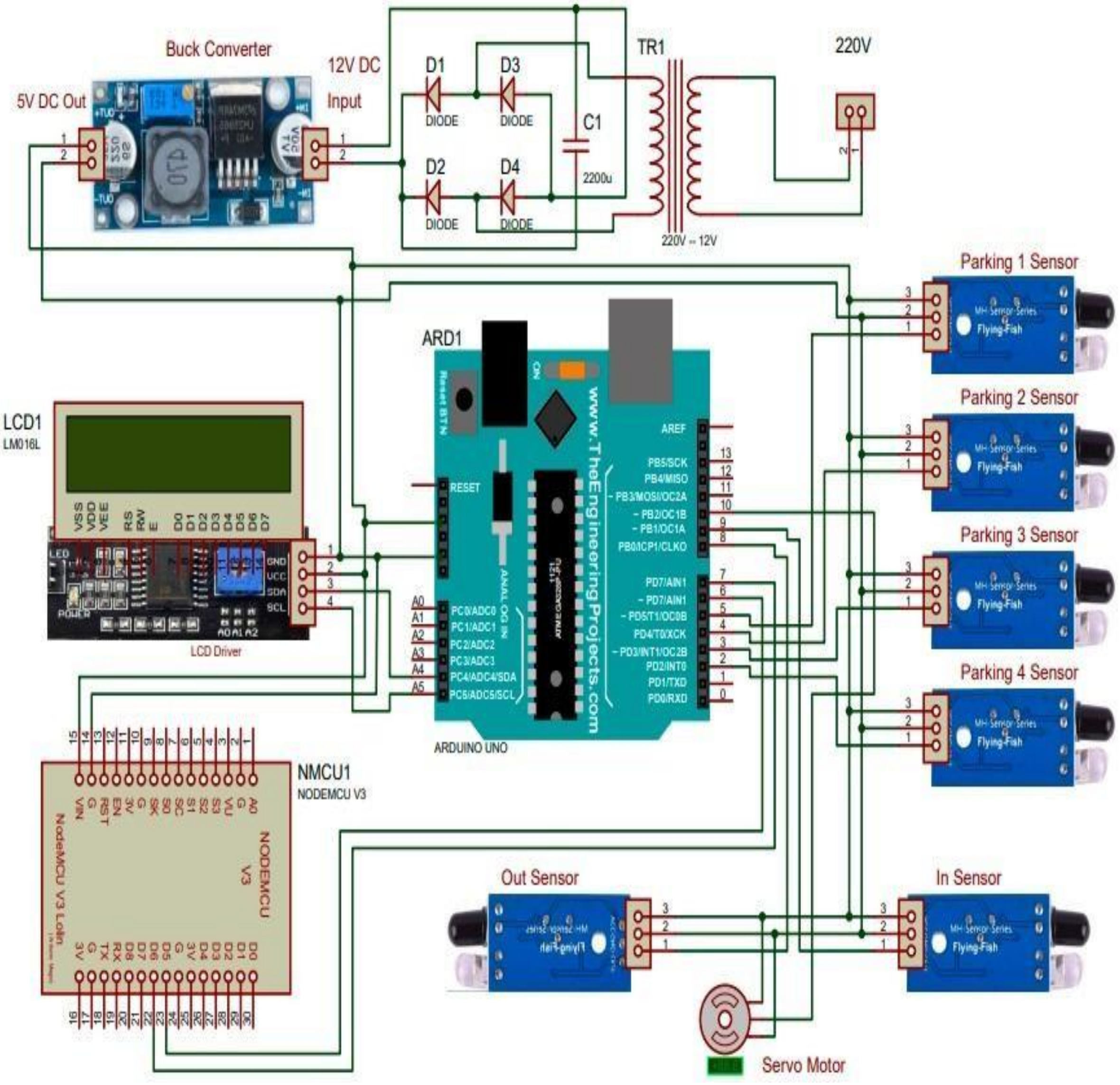


Fig 3.2: Circuit diagram of our project

Hardware Design

The project will help peoples who want to do something or make something with the help of Arduino. The project is designed in two parts, these are the software part and another one is the hardware designing. In this project we have used some basic components like IR (Inferred Ray) sensor as a density detector, LED and Microcontroller.

As a microcontroller we have used Arduino Nano Board here, which is perfect for building a new project or doing anything in the field of robotics or something that is smart to use. To use an Arduino we have to use the Arduino Software which free for all users and can be downloaded from http://www.arduino.cc. The Arduino program is based on C/C++ Programming language. And a huge collection of example is provided in their websites which are also free for all. And the software is one of the easiest software to use.

The hardware design of this project is very simple and efficient also. First of all we have designed the IR sensor. The IR sensor generally works as counter. And counter passes a signal after a significant number of gesture or movement is detected. Then we have designed the whole circuit in the breadboard by the aid of microcontroller. Arduino Nano plank is the perfect choice to make the project more efficient and easy also.

In this control system we have used some basic components to design this circuit. These components are found easily at any electronics shops or markets. For counting we have made a light sensor with some basic components. The output of this circuit is connected to the Yellow, Green and Red lights of Traffic Signal.

The system was designed to be simple and the experimental setup included the prototype model of traffic lights showing lights from the four sides of a junction. The model included traffic lights each on the four sides; depicting red, yellow and green colors. These are designed using colored bulbs of red, yellow and green color. The input signal was given through four different switches to control the timings of four directions.

Working with Arduino Software

First download and install the Arduino IDE for Mac, Linux or Windows to arduino.cc. Windows users also necessity to install a driver. Gather your plank via USB, start the Arduino application and elect Arduino Uno to the tools to plank menu. Bare the design table. Bare Examples: 01. Basics: Blink. Click the toolbar button to upload it to your plank.

The Integrated Development Environment (IDE):

Microcontroller needs software for programming. The Arduino plank has its own integrated development environment (IDE). It is gratis and anybody can download it from its official. That gives Arduino Plank to reach much users and it also helps it to get.

IDE Parts:

a) Compile: Before program "code" can be sent to the board, it needs to be converted into instructions that the board understands. This process is called Compiling.

b) Stop: This stops the compilation process.

c) Create new Sketch: This opens a new window to create news ketch.

d) Open Existing Sketch: This loads a sketch from a file on our computer.

e) Save Sketch: This saves the changes to the sketch.

f) Upload to Board: This compiles and then transmits over the USB cable to our board.

g) Serial Monitor: Until this point when our programs (sketches) didn't work, we just pulled out our hair and tried harder.

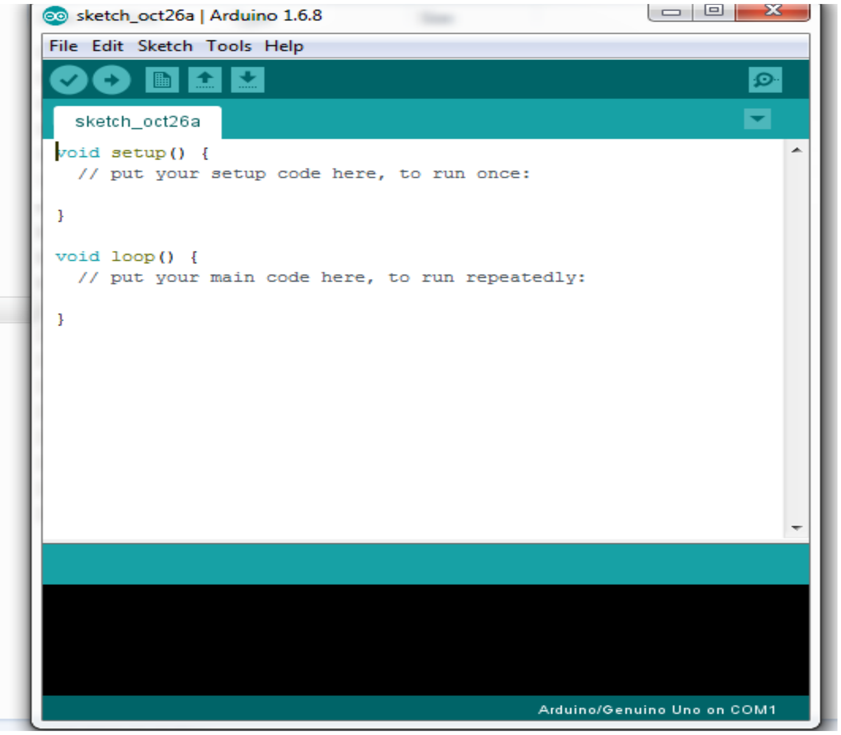
h) Tab Button: This lets you create multiple files in your sketch. This is for more advanced programming than we will do in this class.

i) Sketch Editor: This is where write or edit sketches

j) Text Console: This shows you what the IDE is currently doing and is also where error messages display if make a mistake in typing program.

k) Line Number: This shows what line number your cursor is on.

The smart microcontroller unit named as Arduino nano can be programmed with the Arduino software. There is no any requirement for installing other software rather than Arduino. Firstly, Select "Arduino Uno from the Tools , Board menu (according to the microcontroller on your board).The IC used named as ATmega328 on the Arduino Uno comes pre burned with a boot loader that allows you to upload new code to it without the use of an external hardware programmer.


sketch_oct26a | Arduino 1.6.8
File Edit Sketch Tools Help
sketch_oct26a
Arduino/Genuino Uno on COM1

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Fig 3.3: Programming platform for ARDUINO

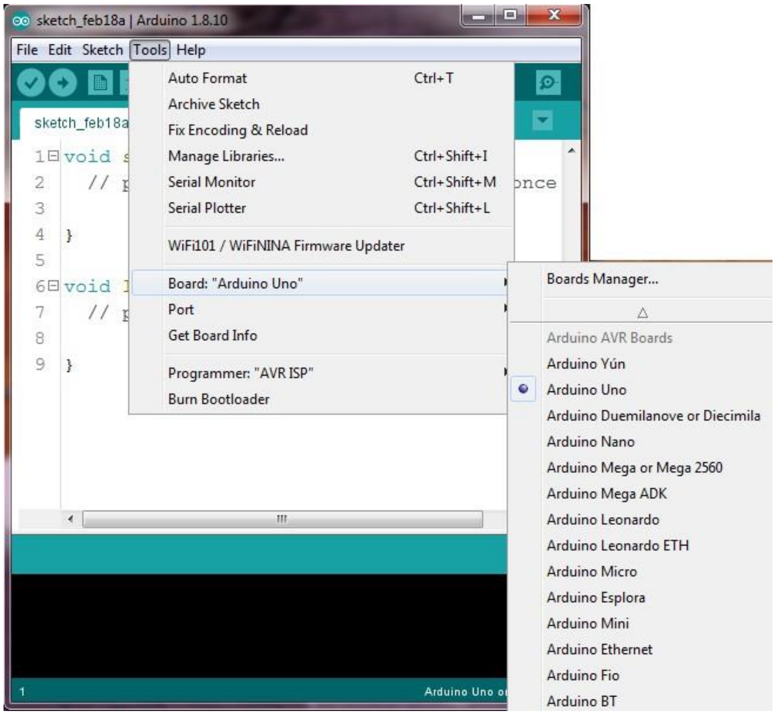


Figure 3.4: IDE configuration for ARDUINO UNO Programmer

Programming software of this line follower is known as ARDUINO-1.6.8. This is open source programming platform**.** The open-source ARDUINO environment makes it easy to write code and upload it to the input/output board. Here we use ARDUINO-1.6.8 platform.

To configure software, we have to use ARDUINO -1.6.8 named arduino.exe

To configure this programmer with computer we need a USB cable then check serial port and select the programmer from Aruino-1.6.8 platform such as,

Project picture

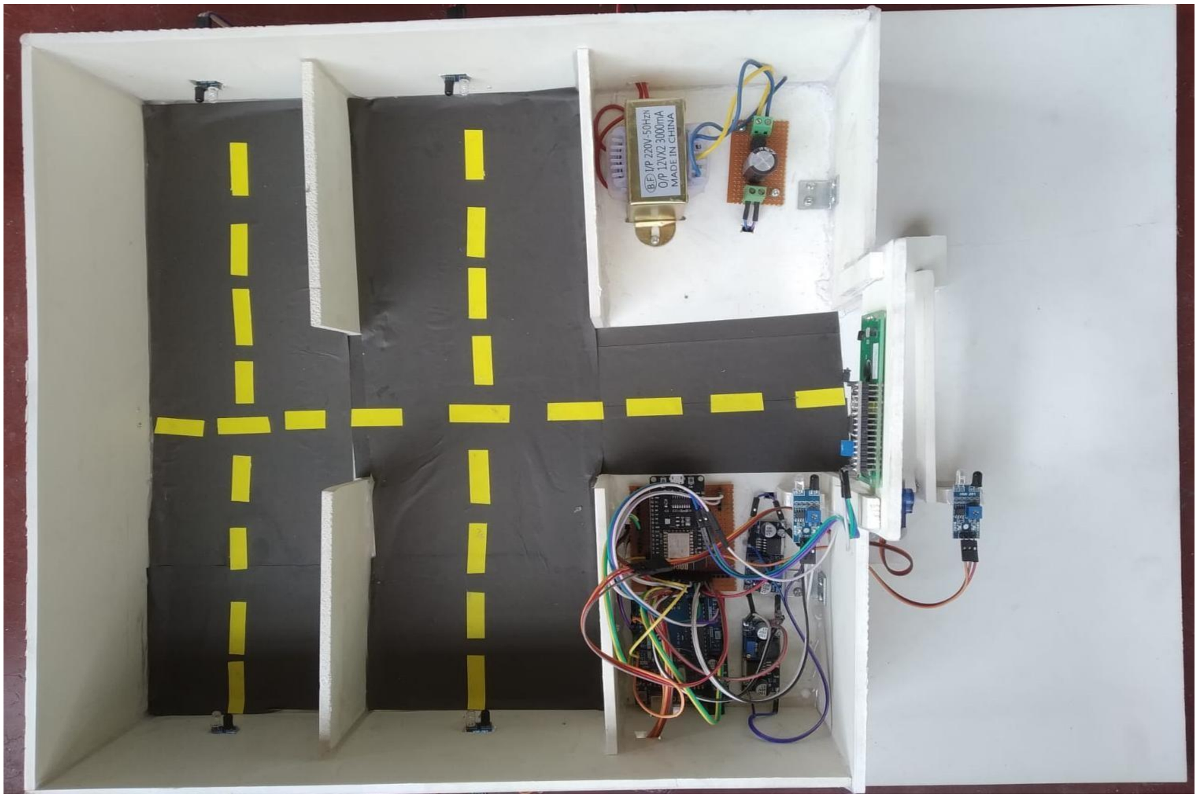


Fig 3.5: Project picture

Summary

This chapter mainly focused on implementation and the design of the system with a block diagram. In this chapter simulation is divided into some parts and operation of each part is discussed briefly. The output of our project shown in that chapter. We also discussed about the research and methodology of our whole project.

CHAPTER IV

RESULT & DISCUSSION

Introduction

In this chapter, we will discuss the outputs and the advantages of our project. There is a variation of coding and variation of using the software in this project. We will also discuss the time delay and efficiency and limitations of this project.

Hardware Result

All the components were connected as per the circuit diagram. The figures below shows the hardware connection and the output obtained.

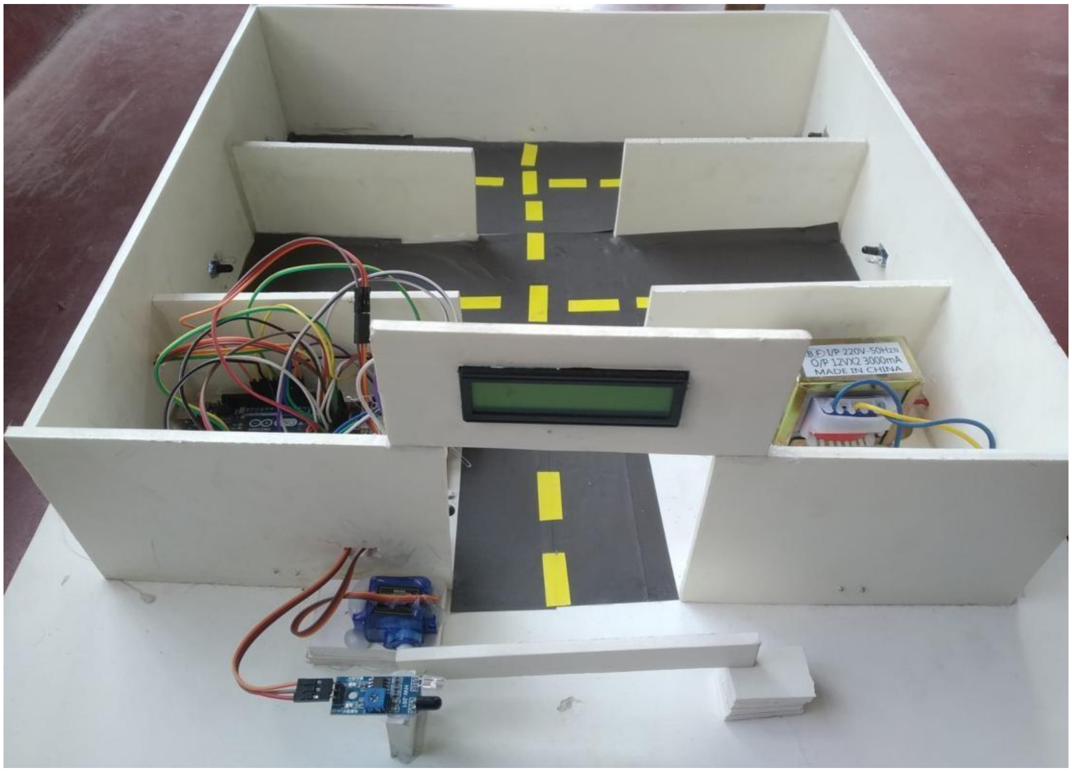


Fig 4.1: Project Top View

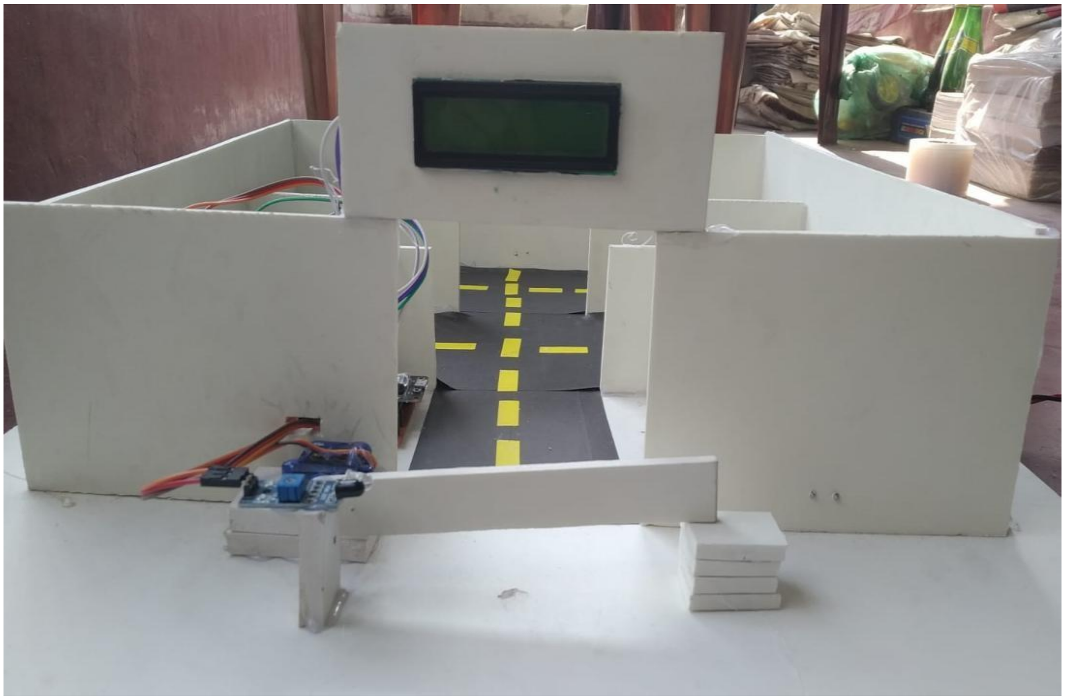


Fig 4.2: Project Front View

Advantage

- It is a real time system and is simple in design.
- It gives accurate Monitoring.
- The system consumes less time to monitor.
- The system is very flexible and economical.

Limitation

- Only four parking lot can be measure.
- Someone need to be check the status periodically.
- Data transfer may hamper due to internet connection.

Application

- It is applicable to reduce road side parking.
- Office buildings

- Shopping Malls
- Hospitals
- Amusement Parks
- Multi-storied house etc.

Costing

Table 4.1: Price list of hardware used at Project

Name	**Quantity**	**Unit Price**	**Total Price**
Arduinoo Uno	1	700	700
NodeMCU	1	500	500
LCD Display	1	550	550
IR Sensor	6	100	600
Servo Motor	1	150	150
Buck converter	1	80	80
Transformer	1	300	300
Cable	1	80	80
Glue	1	50	50
PVC Board	1	450	450
Others	1	1000	1000
		TOTAL COST	**4,460/- Taka**

Discussion

IOT based car parking space detection home control system is capable of reducing the car parking problem in our metropolitan cities. The combination of the hardware and program give the final design of IOT base car parking space detection project. There were some problems occurs in our project . The main problem was programming. It was a major part of this project because it connects by the all equipment and it also connects the cloud server. where we can see the parking slot information. We spend lots of time in this program. Finally we found the actual code in this project. In this project equipment has five part, Arduino Uno, Sonar sensor, NodeMCU, Display, I2C module . I2C module is a LCD display adopter, its connected to the LCD display. Sonar sensor detect the car in parking slot by use of sonar wave. Firstly we test the sonar sensor then test the display by the arduino nano microcontroller. After connected to the NodeMCU by the Arduino nano micro controller and then connected to the Google firebase system. When all the equipment are connected then occurred some connection problem but this was not major, we solved the problem then finally we found our actual output. Pin number is important because it match the program coding.

CHAPTER V

CONCLUSION AND FUTURE WORKS

Conclusion

The parking system is quite a challenge in modern days. Since the modern cities, number of cars has been increasing and day to day people are facing bigger problem while trying to manage their cars into a parking slot. This continuity of parking crisis gives rise to new solutions with the help of Internet of things base car parking space detection thus managing car parking systems. The proposed project provides real time information of a car parking slot. In IOT base car parking space detection system can save our lots of things such a time kill, reduce traffic jam, car safety etc. This project is low cost and high efficiency. It will be benefit the urban building house holder, property developer to increase their revenue which will add to the government tax revenue. It is also helping the government by increasing tax revenue. It will also encourage Engineering in our country which will make advancement in increasing usage of technology.

Future works

In this IOT base car parking projects have many future scope. Day by day man will be increase. So, man will be make house , shopping mall, etc and man will be buy vehicles. For this vehicles parking need parking slot. Manually it is not possible so, our dependency will be come IOT base car parking space detection system. It is much more easy and low cost. In future develop the software and create application for parking space detection. Using this app people can find parking space. For our country this project have lots of scope for find future car parking space detection. In future this project will be added Google map so, we can find easily our parking slot.

References

[1] Rico, J., Sancho, J., Cendon, B., & Camus, M. (2013, March). Parking easier by using context information of a smart city: Enabling fast search and management of parking resources. In Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on (pp. 1380-1385). IEEE.

[2] Zheng, Y., Rajasegarar, S., & Leckie, C. (2015, April). Parking availability prediction for sensor-enabled car parks in smart cities. In Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015 IEEE Tenth International Conference on (pp. 1-6). IEEE.

[3] Zhou, F., & Li, Q. (2014, November). Parking Guidance System Based on ZigBee and Geomagnetic Sensor Technology. In Distributed Computing and Applications to Business, Engineering and Science (DCABES), 2014 13th International Symposium on (pp. 268-271). IEEE.

[4] Botta, A., de Donato, W., Persico, V., & Pescapé, A. (2014, August). On the Integration of Cloud Computing and Internet of Things. In Future Internet of Things and Cloud (FiCloud), 2014 International Conference on (pp. 23-30). IEEE.

[5] Ji, Z., Ganchev, I., O'droma, M., & Zhang, X. (2014, August). A cloud based intelligent car parking services for smart cities. In General Assembly and Scientific Symposium (URSI GASS), 2014 XXXIth URSI (pp. 1-4). IEEE.

[6] International Parking Institute, “2012 Emerging Trends in Parking”.

[7] Ballon, P., Glidden, J., Kranas, P., Menychtas, A., Ruston, S., & Van Der Graaf, S. (2011, October). Is there a Need for a Cloud Platform for European Smart Cities?. In eChallenges e-2011 Conference Proceedings, IIMC International Information Management Corporation.

[8] FastPark System website, http://www.fastprk.com.

[9] Chen, S. Y., Lai, C. F., Huang, Y. M., & Jeng, Y. L. (2013, July). Intelligent home-appliance recognition over IoT cloud network. In Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International (pp. 639-643). IEEE.

[10] Dash, S. K., Mohapatra, S., & Pattnaik, P. K. (2010). A survey on applications of wireless sensor network using cloud computing. International Journal of Computer science & Engineering Technologies (E-ISSN: 2044-6004), 1(4), 50-55.

[11] Fox, G. C., Kamburugamuve, S., & Hartman, R. D. (2012, May). Architecture and measured characteristics of a cloud based internet of things. InCollaboration Technologies and Systems (CTS), 2012 International Conference on (pp. 6-12). IEEE.

[12] Han, D. M., & Lim, J. H. (2010). Smart home energy management system using IEEE 802.15. 4 and zigbee. Consumer Electronics, IEEE Transactions on, 56(3), 1403-1410.

[13] Parwekar, P. (2011, September). From Internet of Things towards cloud of things. In Computer and Communication Technology (ICCCT), 2011 2nd International Conference on (pp. 329-333). IEEE.

[14] Rao, B. B. P., Saluia, P., Sharma, N., Mittal, A., & Sharma, S. V. (2012, December). Cloud computing for Internet of Things & sensing based applications. In Sensing Technology (ICST), 2012 Sixth International Conference on (pp. 374-380). IEEE.

[15] Wikipedia website, https://en.wikipedia.org/wiki/MQTT.

[16] Suciu, G., Vulpe, A., Halunga, S., Fratu, O., Todoran, G., & Suciu, V. (2013, May). Smart cities built on resilient cloud computing and secure internet of things. In Control Systems and Computer Science (CSCS), 2013 19th International Conference on (pp. 513-518). IEEE.

[17] Ye, X., & Huang, J. (2011, December). A framework for cloud-based smart home. In Computer Science and Network Technology (ICCSNT), 2011 International Conference on (Vol. 2, pp. 894-897). IEEE.

[18] Zaslavsky, A., Perera, C., & Georgakopoulos, D. (2013). Sensing as a service and big data. arXiv preprint arXiv:1301.0159.

[19] Doukas, C., Capra, L., Antonelli, F., Jaupaj, E., Tamilin, A., & Carreras, I. (2015, January). Providing generic support for IoT and M2M for mobile devices. In Computing & Communication Technologies Research, Innovation, and Vision for the Future (RIVF), 2015 IEEE RIVF International Conference on (pp. 192-197). IEEE.

[20] Tsirmpas, C., Anastasiou, A., Bountris, P., & Koutsouris, D. A new method for profile generation in an Internet of Things environment: An application in ambient assisted living.

[21] Kafle, V. P., Fukushima, Y., & Harai, H. (2015, April). ID-based communication for realizing IoT and M2M in future heterogeneous mobile networks. In Recent Advances in Internet of Things (RIoT), 2015 International Conference on (pp. 1-6). IEEE.

[22] Sarkar, C., Uttama Nambi SN, A., Prasad, R., Rahim, A., Neisse, R., & Baldini, G. (2012). DIAT: A Scalable Distributed Architecture for IoT.

APPENDIX A

PROGRAM CODE

```
#include <Wire.h>
#include<SoftwareSerial.h>
#include <LiquidCrystal_I2C.h>
#include <Servo.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
SoftwareSerial mySerial(6,7);
Servo gate;

int sensor1 = 2;
int sensor2 = 3;
int sensor3 = 4;
int sensor4 = 5;
int sensor5 = 9;
int sensor6 = 8;
int Rx=0;


void setup() {
  Serial.begin(9600);
  mySerial.begin(9600);
  gate.attach(10);
  lcd.init();
  lcd.backlight();
  lcd.clear();
  lcd.setCursor(1, 0);
  lcd.print(" CAR Parking");
  delay(400);
  lcd.clear();
 pinMode(sensor1, INPUT);
 pinMode(sensor2, INPUT);
 pinMode(sensor3, INPUT);
 pinMode(sensor4, INPUT);
 pinMode(sensor5, INPUT);
 pinMode(sensor6, INPUT);
 gate.write(70);

}
void loop() {
int lot1 = digitalRead(sensor1);
int lot2 = digitalRead(sensor2);
int lot3 = digitalRead(sensor3);
int lot4 = digitalRead(sensor4);

int in1 = digitalRead(sensor5);
int out1 = digitalRead(sensor6);
Serial.print(lot1);
Serial.print(lot2);
Serial.print(lot3);
Serial.print(lot4);

Serial.print(in1);
Serial.println(out1);



if (lot1==0)
{
  lcd.setCursor(0, 0);
  lcd.print("L1: **    ");

  Rx=1;
  mySerial.write(Rx);
}

if (lot1==1)
{
  lcd.setCursor(0, 0);
  lcd.print("L1:       ");

  Rx=2;
  mySerial.write(Rx);
}


if (lot2==0)
{
  lcd.setCursor(10, 0);
  lcd.print("L2: **");

  Rx=3;
  mySerial.write(Rx);
}
```

```
if (lot2==1)
{
  lcd.setCursor(10, 0);
  lcd.print("L2:  ");

  Rx=4;
  mySerial.write(Rx);
}


if (lot3==0)
{
  lcd.setCursor(0, 1);
  lcd.print("L3: **    ");

  Rx=5;
  mySerial.write(Rx);
}

if (lot3==1)
{
  lcd.setCursor(0, 1);
  lcd.print("L3:       ");

  Rx=6;
  mySerial.write(Rx);
}



if (lot4==0)
{
  lcd.setCursor(10, 1);
  lcd.print("L4: **");

  Rx=7;
  mySerial.write(Rx);
}

if (lot4==1)
{
  lcd.setCursor(10, 1);
  lcd.print("L4:  ");
  Rx=8;
  mySerial.write(Rx);
}


if (out1 ==0)
{
  lcd.clear();
  lcd.setCursor(2, 0);
  lcd.print("Thank You");
  gate.write(180);
  lcd.setCursor(0, 1);
  lcd.print("Gate Opening...");
  delay(5000);
  gate.write(70);
  lcd.setCursor(0, 1);
  lcd.print("Gate Closing...");
  delay(1000);
}

if ((lot1==1 || lot2==1 || lot3==1 || lot4==1))
{
  lcd.clear();
  lcd.setCursor(4, 0);
  lcd.print("WELCOME");
  lcd.setCursor(0, 1);
  lcd.print("Gate Opening...");
  delay(1000);
  gate.write(180);
  Serial.println(" Gate Open");
  delay(5000);
  gate.write(70);
  lcd.setCursor(0, 1);
  lcd.print("Gate Closing...");
  delay(1000);
}

delay(1000);
}
```