

A Smart Vehicle Parking Monitoring and Management System Using ANPR Camera Using Android Phones

Submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering
Degree in ELECTRONICS AND COMMUNICATION ENGINEERING

by

B UMA KARTHIK (39130080)

M SATYA SAI (39130264)



DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

SCHOOL OF ELECTRICAL & ELECTRONICS

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHISALAI,
CHENNAI - 600119**

APRIL - 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICAITON ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of B Uma Karthik (39130080) ,M Satya Sai (39130264) who have done the Project work as a team who carried out the project entitled "A Smart Vehicle Parking Monitoring and Management System Using ANPR Camera Using Android Phones" under our supervision from November 2022 to April 2023.

Internal Guide

Ms. F.V. Jaya Sudha, M.E., (Ph.D)

Head of the Department

Dr. T. RAVI, M.E., Ph.D.,

Submitted for Viva voce Examination held on 19-04-2023

Internal Examiner

External Examiner

DECLARATION

We, **B UMA KARTHIK (39130080)** and **M SATYA SAI (39130264)** and hereby declare that the Project Report on “**A SMART VEHICLE PARKING MONITORING AND MANAGEMENT SYSTEM USING ANPR CAMERA USING ANDROID PHONES**” done by us under the guidance of **Ms.F.V.Jaya Sudha M.E., (Ph.D)** at Sathyabama Institute of Science and technology is submitted in partial fulfilment of the requirements of the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

DATE: 19-04-23

PLACE: Chennai

SIGNATURE OF THE CANDIDATES

1. 
2. 

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. N.M. NANDHITHA M.E., Ph. D, Dean**, School of Electronics, **Dr. T.RAVI M.E., Ph.D.**, Head of the Department of Electronics and Communication Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Ms.F.V.JayaSudha M.E.,(Ph.D)**,for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my phase-2 project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Electronics and Communication Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Nowadays parking is a critical issue in a smart city. Due to parking problems, traffic problems are increasing, the proposed smart parking system implemented using the Android Application that's provided users with an easy way of booking parking slots through an application. Given system avoid the problem of traffic conjunction in commercial areas that unnecessarily consumes time, this paper provides an easy reservation system for parking. In this application, the user can view various parking slots and check for the availability of slots. Whenever a user books a particular slot, it will be marked red and all the available slots will be green. Booking can be done through credit card/net banking. This application also provides an additional feature of canceling the booked slot within 20 minutes from the time of booking. If the user fails to reach the destination on time, then the reservation will be canceled and the payment is refunded. On successful payment, a parking number is sent to the user's email or to his mobile number for further inquiry. Hence this application reduces the user's effort and time of searching the parking slot and also avoids the conjunction of traffic using the internet of things.

TABLE OF CONTENTS

CHAPTER NO.	TOPIC	PAGE NO.
	ABSTRACT	v
	LIST OF TABLES	viii
	LIST OF FIGURES	ix
	LIST OF ABBREVIATIONS	x
1	INTRODUCTION	
	1.1 RELAVANCE OF PROJECT	1
	1.2 EMBEDDED SYSTEM	3
	1.3 BRINGING SOFTWARE AND HARDWARE TOGETHER FOR EMBEDDED SYSTEM	5
	1.4 APPLICATIONS	6
2	LITERATURE SURVEY	
	2.1 LITERATURE SURVEY	7
	2.2 IMPLEMENTATION FLOW	12
3	SCOPE OF THE PROJECT	
	3.1 METHODOLOGY	14
	3.2 OBJECTIVE	16
4	PROPOSED METHODOLOGY	
	4.1 BLOCK DIAGRAM FOR PROPOSED	19
	4.2 ADVANTAGES OF PROPOSED SYSTEM	20
	4.3 HARDWARE REQUIREMENTS	20
	4.4 FEATURES OF ARDUINO	23
	4.5 POWER SUPPLY	27
	4.6 RECTIFIER	28
	4.7 CAPACITORS	28
	4.8 VOLTAGE REGULATORS	29
	4.9 WEB CAMERA	30
	4.10 ULTRASONIC SENSOR	31
	4.11 LCD	33
	4.12 NODE MCU	35
	4.13 SOFTWARE REQUIREMENT	37
5	RESULT AND ANALYSIS	
	5.1 RESULT	41
6	CONCLUSION AND FUTURE SCOPE	

6.1 CONCLUSION	43
6.2 FUTURE SCOPE	43
REFERENCES	44

LIST OF TABLES

CHAPTER NO.	TABLE NO.	TABLE NAME	PAGE NO.
4	Table 4.1	ARDUNIO UNO TECHNICAL SPECIFICATION	26
4	Table 4.2	LCD PIN DESCRIPTION	35

LIST OF FIGURES

FIGURE NO.	FIGURE NAME	PAGE NO.
1.1	CAR PARKING VACANCY DETECTOR	2
1.2	LINK TO OTHER SYSTEM	3
1.3	BLOCK DIAGRAM OF EMBEDDED	3
1.4	FLOW OF BURNING SOURCE CODE TO PROCESSOR	5
3.1	PARKING ZONE STATUS DISPLAY	13
3.2	NUMBER PLATE DETECTION	14
3.3	PARTITION SYSTEM AND DISPLAY	15
3.4	ARCHITECTURE OF STREET PARKING SYSTEM	16
3.5	ANPR CAMERA FOR CAR PARKING SYSTEM	17
4.1	BLOCK DIAGRAM PROPOSED METHOD	19
4.2	TYPES OF ARDUNIO	20
4.3	MICROCONTROLLER	21
4.4	ARDUINO UNO	22
4.5	ATMEGA328	23
4.6	ARDUINO UNO PINOUT	24
4.7	CIRCUIT OF POWER SUPPLY	27
4.8	CIRCUIT OF RECTIFIER	28
4.9	CAPACTIOR	29
4.10	7805 VOLTAGE REGULATOR WITH PINOUT	29
4.11	7812 VOLTAGE REGULATOR WITH PINOUT	30
4.12	WEB CAM	30
4.13	SENSOR DETECTING	32
4.14	FRONT VIEW AND BACK VIEW	34
4.15	NODEMCU PINOUT	36
4.16	ARDUONO INTERFACE	38
4.17	COM 5	40
5.1	PARKING SYSTEM MODEL	42

LIST OF ABBREVIATIONS

CHAPTER NO.	ABBREVAITIONS	FULL FORM	PAGE NO.
4	GND	GROUND PINS	25
4	SPI	SERIAL PERIPHERAL	25
4	IOREF	INPUT/OUTPUT REFERENCES	25
4	AREF	ANALOG REFERENCE	25
4	TWI	TWO WIRE INTERFACE	25
4	PWM	PULSE WIDTH MODULATION	25
4	SRAM	STATIC RANDOM ACCESS MEMORY	26
4	LCD	LIQUID CRYSTAL DISPLAY	33
4	COM5	COMMUNICATION PORT	40

CHAPTER 1

INTRODUCTION

The project entitled smart parking system is to manage all the parking facilities for a user. With the recent growth in the economy and due to the availability of low-price cars in the market, every average middle-class individual can afford a car, which is a good thing, however, the consequences of heavy traffic jams, pollution, less availability of roads and spot to drive the motor car. One of the important concerns, which is to be taken in accounting, is the problem of parking those vehicles. Though, if there is space for parking the vehicle but so much time is squandered in finding that exact parking slot resulting in more fuel intake and not also environment-friendly. It will be a great deal if in some way we find out that the parking itself can provide the precise vacant position of a parking slot then it'll be helpful not limited to the drivers but also for the environment. Initially, when the user is about to enter the location, the LCD displays the number of empty and filled spots and when the user is with their vehicle near the parking detect sensor, he/she would be thrown a notification on the mobile app of the parking slot number, where they should park their vehicle. The main important benefit of a smart parking system is its advanced technology. It follows the latest technologies and concepts to assure profitable outcomes. The design and implementation of smart parking are very easy to supervise and manage. This system can be easily handled by the staff members because of its well-organized structure

1.1 Relevance of the project

The main important benefit of a smart parking system is its advanced technology. It follows the latest technologies and concepts to assure profitable outcomes. The design and implementation of smart parking are very easy to supervise and manage. This system can be easily handled by the staff members because of its well-organized structure.

In recent research, in metropolitan cities, the parking management problem can be viewed from various angles such as high vehicle density on roads. This

results in annoying issues for the drivers to park their vehicles as it is very difficult to find a parking slot. The drivers usually waste time and effort in finding parking spaces and end up parking their vehicles and finding a space on the street which further leads to space congestion. In the worst case, people fail to find any parking space, especially during peak hours and the festive season.



Fig 1.1 Car Parking Vacancy Detector

An embedded system is one kind of a computer system mainly designed to perform several tasks like to access, process, and store and also control the data in various electronics-based systems. Embedded systems are a combination of hardware and software where software is usually known as firmware that is embedded into the hardware. One of its most important characteristics of these systems is, it gives the o/p within the time limits. Embedded systems support to make the work more perfect and convenient. So, we frequently use embedded systems in simple and complex devices too. The applications of embedded systems mainly involve in our real life for several devices like microwave, calculators, TV remote control, home security and neighborhood traffic control systems, etc.

User interface

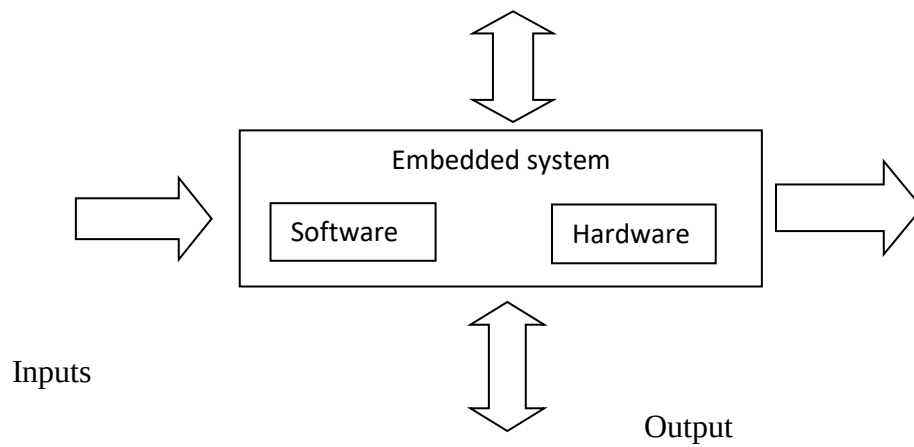


Fig 1.2 Link to other systems

1.2 Embedded system:

Embedded system includes mainly two sections, they are

1. Hardware
2. Software

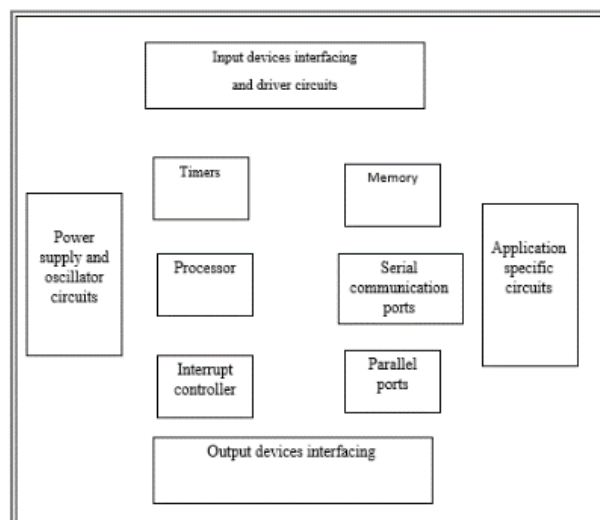


Fig :1.3 Block diagram of embedded system

1.2.1 Embedded System Hardware

As with any electronic system, an embedded system requires a hardware platform on which it performs the operation. Embedded system hardware is built with a microprocessor or microcontroller. The embedded system hardware has elements like input output (I/O) interfaces, user interface, memory and the display. Usually, an embedded system consists of:

- Power Supply
 - Processor
 - Memory
 - Timers
 - Serial communication ports
 - Output/Output circuits
 - System application specific circuits
-
- Embedded systems use different processors for its desired operation. Some of the processors used are Microcontroller, Microprocessor, Digital signal processor **CPU** on a chip, We can attach required amount of ROM, RAM and I/O ports. Large in size, general-purpose.

1.2.2 Embedded System Software

The embedded system software is written to perform a specific function. It is typically written in a high level format and then compiled down to provide code that can be lodged within a non-volatile memory within the hardware. An embedded system software is designed to keep in view of the three limits:

- Availability of system memory
- Availability of processor's speed
- When the system runs continuously, there is a need to limit power dissipation for events like stop, run and wake up.

1.3 Bringing software and hardware together for embedded system

To make software to work with embedded systems we need to bring software and hardware together .for this purpose we need to burn our source code into microprocessor or microcontroller which is a hardware component and which takes care of all operations to be done by embedded system according to our code. Generally we write source codes for embedded systems in assembly language, but the processors run only executable files.The process of converting the source code representation of your embedded software into an executable binary image involves three distinct steps:

1. Each of the source files must be compiled or assembled into an object file.
2. All of the object files that result from the first step must be linked together to produce a single object file, called the re-locatable program.

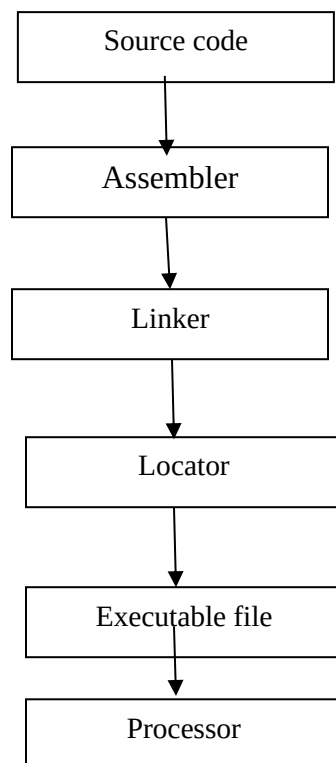


Fig 1.4 Flow of burning source code to processor

1.4 Applications

Embedded systems have different applications. A few select applications of embedded systems are smart cards, telecommunications, satellites, missiles, digital consumer electronics, computer networking, etc.

Embedded Systems in Automobiles

- Motor Control System
- Engine or Body Safety
- Robotics in Assembly Line
- Mobile and E-Com Access

1.4.1 Embedded systems in Telecommunications

- Mobile computing
- Networking
- Wireless Communications

1.4.2 Embedded Systems in Smart Cards

- Banking
- Telephone
- Security Systems

CHAPTER 2

Literature Survey

The following shows a survey done for a smart Vehicle parking System through MQTT broker service automation for with and without internet on Realtime which includes an instant feedback mechanism that drives the exact slot appliances. This system is reliable to integrate on multiple platforms with the help of ESP32 master which serves as standard firmware. Further, the study was extended to multiple operation platforms like Linux, Windows, and IOS are discussed as follows. In this detailed survey, we have learned a lot in stabilizing our concepts from the existing feedback mechanisms that were designed in the previous study.

2.1 Literature Survey

Satyendra K. et.al (2021) presents a step-by-step procedure of a smart home automation controller. It uses IoT to convert home appliances to smart and intelligent devices, with the help of design control. An energy-efficient system is designed that accesses the smart home remotely using IoT connectivity. The proposed system mainly requires, Node MCU as the microcontroller unit, IFTTT to interpret voice commands, Adafruit a library that supports MQTT acts as an MQTT broker, and Arduino IDE to code the microcontroller. This multimodal system uses Google Assistant along with a web-based application to control the smart home. The smart home is implemented with the main control unit that is connected to the 24-hour available Wi-Fi network. To ensure, that the Wi-Fi connection does not turn off, the main controller is programmed to establish an automatic connection with the available network and connected to the auto power backup. The target scaling factor is efficient throughout the multiple platforms switching 9 Slower transit cycles when compared to CoAP, low scalability, and lack of security encryption

Shardha Somani et.al (2021) focuses on a system that provides features of Home Automation relying on IoT to operate easily, in addition to that it includes a camera module and provides home security. The android application basically converts smartphones into a remote for all home appliances. Security is achieved with motion sensors if movement is sensed at the entrance of the house; a notification is sent that contains a photo of the

house entrance in real-time. This notification will be received by the owner of the house via the internet such that the app can trigger a notification. So, the owner can raise an alarm in case of any intrusion or he/she can toggle the appliances like opening the door if the person is a guest. The system uses Raspberry Pi, a small-sized computer that acts as a server for the system. The smart home consists of two modules. Home automation consists; of a fanlight and door controller, and a security module that consists; of a smoke sensor motion sensor, and camera module. Implementation of IR Interface was an excellent Idea for creating universal remote The system Uses Infrared Radiation for data transfer which is much faster than CoAP but could be easily mapped with 38 kHz of regular frequency.

Tui-Yi Yang et.al (2017) proposes an optimization of home power consumption based on PLC (Power Line Communication) for an easy-to-access home energy consumption. This also proposes a Zigbee and PLC-based renewable energy gateway to monitor the energy generation of renewable energies. ACS and DDEM algorithms are proposed for the design of intelligent distribution of power management systems to make sure ongoing power supply of home networks. To provide efficient power management the power supply models of home sensor networks are classified into groups viz. main supply only, main supply and backup battery, rechargeable battery power, and non-rechargeable battery power. Devices with particular features are assigned to these groups. It targets establishing a real-time processing scheme to address variable sensor network topologies. Fast and secured since the data encryption is 64bit ratio and it follows MQTT protocol Does not support a large database size, and it has no support for ROLE, COMMIT, and Stored Procedures

Ushar Chaurasia et.al (2018), proposes a system that develops a model to reduce the computation overhead in existing smart home solutions that use various encryption technologies like AES, ECHD, hybrid, etc. these solutions use an intermediate gateway for connecting various sensor devices. The proposed model provides a method for automation with sensor-based learning. The system uses a temperature sensor for development but other sensors can also be used as per requirement. These smart home devices with sensors can configure themselves autonomously and can operate without

human intervention. This work minimizes encryption decryption and focuses on authentication and automation of smart home devices with learning. The system bypasses the local gateway mentioned in the existing system to provide better security for smart home devices and sensor data and save computation overhead. The real-time broker cloud is directly connected with the smart home and manages all incoming and outgoing requests between users and devices. The main purpose to use a real-time broker cloud saves time for cryptographic operations. Bringing AR with Deep Neural schema is an Excellency and this system can be integrated with the 7 senses of the future world. It has a range of support between 3.5Hz to 75Hz which in turn directly matches the neural frequency that potentially messes up sensory organs.

Ish Kool et.al (2020) presents a vision-based machine intelligence system to sense the on/off state of common home appliances. The proposed method of sensing the state of appliances results in a novel home automation system. The accessibility of the suite of devices in the home over a remote network is facilitated by the IP Addressing methods in the IoT. This project uses two boards viz. Raspberry Pi and Intel Galileo Gen 2. The communication between the User devices, Raspberry Pi, and the Intel Galileo boards happens over a wireless network. The UDP protocol is deployed to facilitate the wireless communication of the nodes present in the home automation network. A Pi Cam and a US Logitech camera are attached to the rotating shaft of two different servo motor capture snapshots that are passed as inputs to the Machine Learning based models trained using dlib-C++ to detect the state of the operation of the appliances. The proposed method uses visual modality to automate the appliances, as privacy concerns may emerge while using the images from some specific places, as a counter to this issue, 12 an SPDT switch is added to the Raspberry Pi which when turned off ensures that even if the images are taken from the webcams, they are just passed as inputs to the machine learning models and are not displayed on the website when the users access the website on the server address obtained from Raspberry Pi. High Visual and Artificial Database was integrated which can be a link between and virtual and physical world. A creative contextual database has a created a huge responsive factor Superimposing virtual environmental capacity has no range of lower frequency AR camera support

Vikram N et.al (2015) illustrates a methodology to provide a low-cost Home Automation System (HAS) using Wireless Fidelity (Wi-Fi). This crystallizes the concept of internetworking of smart devices. A Wi-Fi-based Wireless Sensor Network (WSN) is designed for the purpose of monitoring and controlling the environmental, safety, and electrical parameters of a smart interconnected home. The different sections of the HAS are; the temperature and humidity sensor, gas leakage warning system, fire alarm system, burglar alarm system, rain-sensing, switching and regulation of load & voltage, and current sensing. The primary requirement of HAS to monitor and control devices is accomplished using a Smartphone application. The application is developed using Android Studio based on the JAVA platform and the User Interface is exemplified. The primary focus of the paper is to develop a solution cost-effective and flexible in the control of devices and implements a wide range of sensors to capture various parameters. Implementing a local server reduces cognitive allocation in the server. Error Factors are high since it is purely based on local server data.

Mrs. Paul Jasmin Rani et.al (2018) focuses on the construction of a fully functional voice-based home automation system that uses the Internet of Things, Artificial Intelligence, and Natural Language Processing (NLP) to provide a cost-effective, efficient way to work together with home appliances using various technologies such as GSM, NFC, etc. it implements a seamless integration of all the appliances to a central console, i.e., the mobile device. The prototype uses Arduino MK1000, known as Genuine MK1000. The NLP in this project gives the user the freedom to interact with the home appliances with his/her own voice and normal language rather than complicated computer commands. The appliances are connected to the mobile device through an Arduino Board that establishes the concept of the Internet of Things. The Arduino Boards are interfaced with the appliances and programmed in such a way that they respond to mobile inputs. Only a few notable parameters throughout the project The rate of prediction is proportionally less when compared to real-world predictions

Jonathan J. et.al (2017) A new method for augmenting paper documents with electronic information is described that does not modify the format of the paper document in any way. Applicable to both commercially printed documents as well as documents that

are output from PC's, the technique we call Paper-Based Augmented Reality substantially improves the utility of paper. We describe the recognition technology that makes this possible as well as several applications. An implementation on a camera phone is discussed that lets users retrieve data and access links from paper documents to electronic data. Recognition is performed at 4 frames per second on a Treo 700w and support is provided for several user applications, including "clickable paper" – printed web pages whose appearance is unchanged but that can be navigated with a camera phone. Augmented reality with virtual automation was brought to the physical world with the paper content . Linking patches of text will increase a lot of security reasons

Mikko Kytö et.al (2018) proposed Head and eye movement can be leveraged to improve the user's interaction repertoire for wearable displays. Head movements are deliberate and accurate and provide the current state-of-the-art pointing technique. Eye gaze can potentially be faster and more ergonomic but suffers from low accuracy due to calibration errors and the drift of wearable eye-tracking sensors. This work investigates precise, multimodal selection techniques using head motion and eye gaze. A comparison of speed and pointing accuracy reveals the relative merits of each method, including the achievable target size for robust selection. We demonstrate and discuss example applications for 15 augmented realities, including compact menus with deep structure, and a proof-of-concept method for online correction of calibration drift. Accuracy has reached 97% with a simple algorithm 97% precise and accurate but depends purely on real-time saturation and hue point of target images

Subramani Roy Choudri et.al (2021) human life is heading towards a busy schedule it becomes necessary to automate our home appliances. Human error is something that cannot be completely erased. With the busy schedule in hand, there is defiantly a possibility of missing something that may be trivial to us but can result in a catastrophe. For these reasons, Our Project home automation can increase efficiency, security, and reliability. AR has recently evolved for the automation of various electrical appliances by popping virtual objects into the real world. Target Image can be linked to the Virtual world Target Image can be linked-to automation with an augmented

environment but in the case of wet hands, this linking process will not produce tokens to respective server

The current system works with individual sensors in parking slots to update the data Here a lot of Power factor usage and able AMC problems still exist Due to the failure of parking units live status needs to be monitored physically This system is not fully automated due to this semi operable conditions Data are not End to End encrypted there might be a full chance to create collusion

2.2 Implementation flow

Stage 1:

Considering the problems of existing methods and giving solution to that problem by considering the basic requirements for our proposed system

Stage 2:

Considering the hardware requirement for the proposed system

For this we need to select the below components:

1. Microcontroller
2. Inputs for the proposed system (ex: sensors, drivers etc..)
3. Outputs (ex: relays, loads)

Stage 3:

After considering hardware requirements, now we need to check out the software requirements. Based on the microcontroller we select there exists different software for coding, compiling, debugging. we need to write source code for that proposed system based on our requirements and compile, debug the code in that software .

After completing all the requirements of software and hardware we need to bring both together to work our system. For this we need to burn our source code into microcontroller, after burning our source code to microcontroller then connect all input and output modules as per our requirement.

CHAPTER 3

SCOPE OF THE PROJECT

At present some countries have portals through which users can gain information about parking areas via the internet. This system can give users information about parking spaces, but it won't be able to give which parking slot is vacant and occupied. Hence, such a system cannot smartly handle the issue. Car lifts along with an automated robotic system, which automatically takes the car to a particular parking spot as soon as the car enters a platform. This system cannot be installed by medium-scale shopping malls or movie theatres as it can cost them a huge amount. In many public places, the system only shows the availability but it cannot show the exact slot and path to the slot available. Hence, there is a need to smartly find the path to the vacant spot.



Fig 3.1 Parking Zone Status Display

3.1 Methodology

In this project, we are using NodeMCU, IR sensors, and servo motors. One IR sensor is used at the entry and exit gate to detect the car while two IR sensors are used to detect the parking slot availability. Servo motors are used to open and close the gates according to the sensor value. NodeMCU is an open-source IoT platform. 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 term “NodeMCU” by default refers to the firmware rather than the dev kits. The firmware uses the Lua scripting language. The ESP8266 is a low-cost Wi-Fi-enabled microchip with a full TCP/IP stack and microcontroller capability. NodeMCU includes a CPU core, faster Wi-Fi, more GPIOs, and supports Bluetooth 4.2, and low-power Bluetooth.



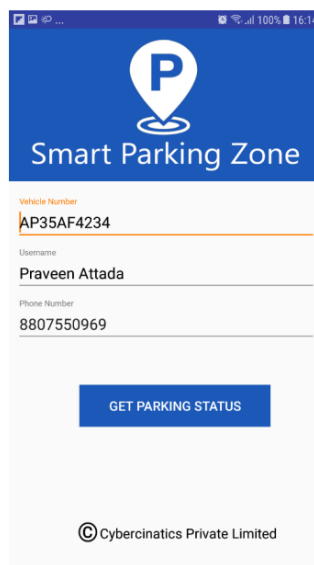
Fig 3.2 Number Plate Detection System

The ESP8266 is a low-cost Wi-Fi-enabled microchip with a full TCP/IP stack and microcontroller capability. NodeMCU includes a CPU core, faster Wi-Fi, more GPIOs, and supports Bluetooth 4.2, and low-power Bluetooth. As soon as the IR sensors get the presence of a

car in front of the entrance, it will send a signal to the NodeMCU to check if there is an empty slot inside the parking lot. When NodeMCU acknowledges that there is an empty slot or more then it will send a signal to the dc servo motor which will open the main entrance. On the other hand, if a NodeMCU encounters no empty slots at the time of a car trying to make an entrance, the gate will just not open. In addition, there will be a website linked with the NodeMCU board to show the number of parking. The idea behind our methodology is very simple, usually, users spend most of their time looking for an empty slot where they can park their vehicle which increases fuel consumption and time wastage. We came up with a new method where we provide the user with an empty slot number where he can park his vehicle without wasting his time for finding one. Similarly, we try to display the start time and end time so that the user can know for what amount of time he has parked his vehicle.



Welcome Screen



App Registration



Parking Zone List

Fig 3.3 Partition System and Display

Various features of the smart parking system are listed below. • Inquiry on availability of parking space and reservation of parking lot • Real-time parking navigation

and route guidance • Vehicle occupancy detection and management of parking lots. Most of the smart parking systems (SPS) proposed in literature over the past few years provide a solution to the design of parking availability information systems, parking reservation systems, occupancy detection and management of parking lot, real-time navigation within the parking facility, etc. However, very few works have paid attention to the real-time detection of improper parking and the automatic collection of parking charges. Thus, this paper presents an internet-of thing (IoT) based E-parking system that employs an integrated component called a parking meter (PM) to address the following issues. • Real-time detection of improper parking • Estimation of each vehicles duration of parking lot usage • Automatic collection of parking charges the E-parking system proposed in this paper also provides a city-wide smart parking management solution via providing parking facility availability information and parking lot reservation system and it is named as parking meter (PM) based E-parking (PM-EP)

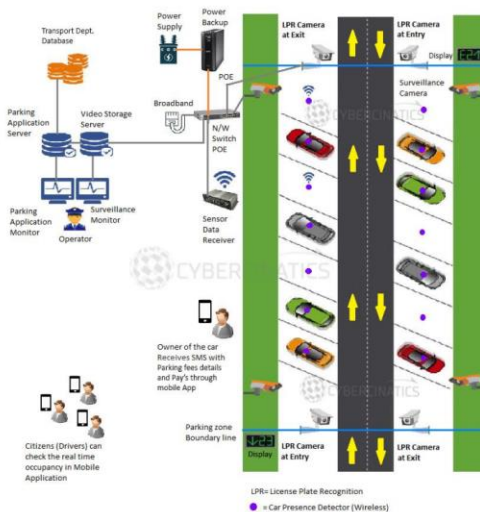


Fig 3.4 Architecture of Street Parking System

3.2 Objective

Smart Parking involves the use of low-cost sensors, real-time data, and applications that allow users to monitor available and unavailable parking spots. The goal

is to automate and decrease time spent manually searching for the optimal parking floor, spot, and even lot. Some solutions will encompass a complete suite of services such as online payments, parking time notifications, and even car searching functionalities for very large lots. A parking solution can greatly benefit both the user and the lot owner. Optimized parking – Users find the best spot available, saving time, resources, and effort. The parking lot fills up efficiently and space can be utilized properly by commercial and corporate entities. Reduced traffic – Traffic flow increases as fewer cars are required to drive around in search of an open parking space. Reduced pollution – Searching for parking burns around one million barrels of oil a day.



Fig 3.5 ANPR Camera for Car Parking System

An optimal parking solution will significantly decrease driving time, thus lowering the number of daily vehicle emissions and ultimately reducing the global environmental footprint. Increased Safety – Parking lot employees and security guards contain real-time lot data that can help prevent parking violations and suspicious activity. License plate recognition cameras can gather pertinent footage. Also, decreased spot-searching traffic on the streets can reduce accidents caused by the distraction of searching for parking. Decreased Management Costs – More automation and less manual activity save on labor costs and resource exhaustion. Enhanced User Experience – A smart parking solution

will integrate the entire user experience into a unified action. Driver's payment, spot identification, location search, and time notifications all seamlessly become part of the destination arrival process.

CHAPTER 4

PROPOSED METHODOLOGY

the architecture and technology stack related to the smart parking system. In this section, we talk about the implementation and working of the system in a real-world scenario. The complete process of booking a parking slot, parking a car in that slot, and leaving the parking area is explained with the help of the following flow chart. We conducted an experiment in order to depict the working of our system at every stage from checking the availability of parking space to actually parking a car in a vacant parking slot. This is done by implementing a smart parking system in the parking area of a shopping mall. Below are the steps that a driver needs to follow in order to park their car using our parking system

BLOCK DIAGRAM FOR PROPOSED METHOD

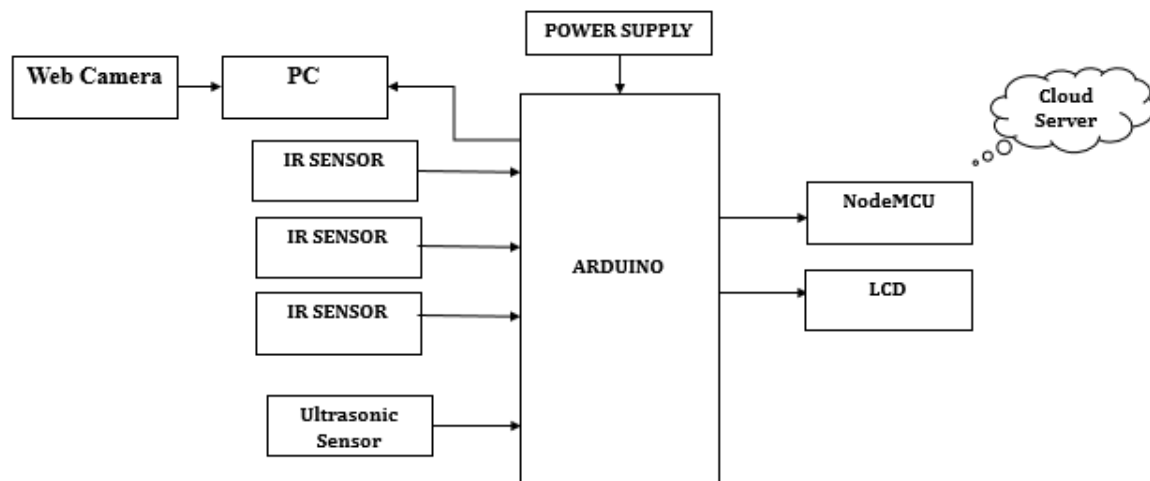


Fig 4.1 proposed method

4.1 Advantages of Proposed System

The concept of the Internet of Things (IoT) started with things with identity communication devices. The devices could be tracked, controlled, or monitored using remote computers connected through the Internet. IoT extends the use of the Internet providing communication, and thus inter-network of the devices and physical objects, or ‘Things’. The two prominent words in IoT are “internet” and “things”. Internet means a vast global network of connected servers, computers, tablets, and mobiles using internationally used protocols and connecting systems. The Internet enables sending, receiving, or communicating information.

4.2 HARDWARE REQUIREMENTS

Arduino Uno is a very valuable addition in the electronics that consists of USB interface, 14 digital I/O pins, 6 analog pins, and Atmega328 microcontroller. It also supports serial communication using Tx and Rx pins. There are many versions of Arduino boards introduced in the market like Arduino Uno, Arduino Due, Arduino Leonardo, Arduino Mega, however, most common versions are Arduino Uno and Arduino Mega. If you are planning to create a project relating to digital electronics, embedded system, robotics, or IoT, then using Arduino Uno would be the best, easy and most economical option.

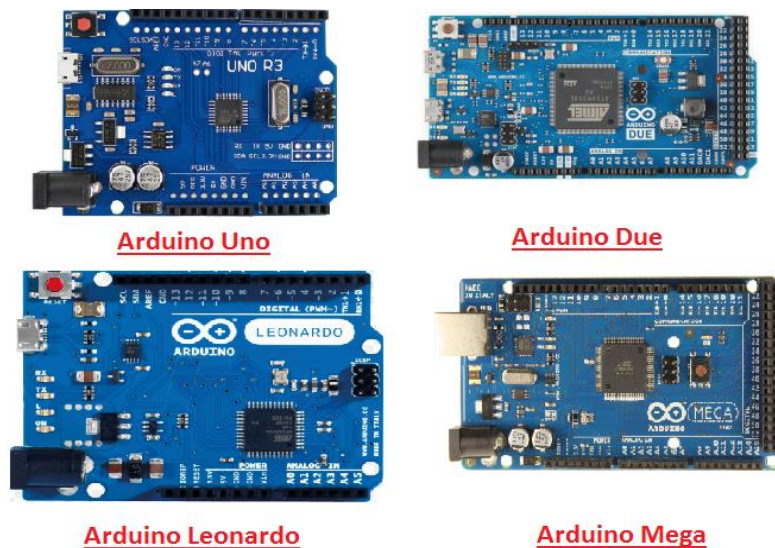


Fig 4.1 Types of arduino

It is an open-source platform, means the boards and software are readily available and anyone can modify and optimize the boards for better functionality.

The software used for Arduino devices is called IDE (Integrated Development Environment) which is free to use and required some basic skills to learn it. It can be programmed using C and C++ language.

Some people get confused between **Microcontroller and Arduino**. While former is just an on system 40 pin chip that comes with a built-in microprocessor and later is a board that comes with the microcontroller in the base of the board, bootloader and allows easy access to input-output pins and makes uploading or burning of the program very easy.



Fig 4.2 Microcontroller

While learning microcontroller requires some expertise and skills. Nevertheless, we can say every Arduino is basically a microcontroller but not every microcontroller is an Arduino.

- **Arduino Uno** is a microcontroller board developed by Arduino.cc which is an open-source electronics platform mainly based on AVR microcontroller Atmega328.
- First Arduino project was started in Interaction Design Institute Ivrea in 2003 by David Cuartielles and Massimo Banzi with the intention of providing a cheap and flexible way to students and professional for controlling a number of devices in the real world.

- The current version of Arduino Uno comes with USB interface, 6 analog input pins, 14 I/O digital ports that are used to connect with external electronic circuits. Out of 14 I/O ports, 6 pins can be used for PWM output.
- It allows the designers to control and sense the external electronic devices in the real world

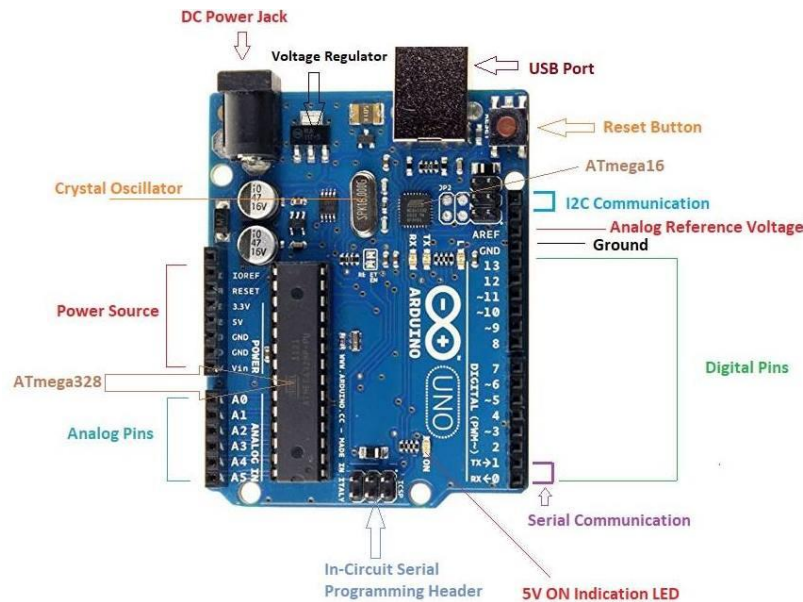


Fig 4.3 Arduino UNO

- This board comes with all the features required to run the controller and can be directly connected to the computer through USB cable that is used to transfer the code to the controller using IDE (Integrated Development Environment) software, mainly developed to program Arduino. IDE is equally compatible with Windows, MAC or Linux Systems, however, Windows is preferable to use. Programming languages like C and C++ are used in IDE.
- Apart from USB, battery or AC to DC adopter can also be used to power the board.
- Arduino Uno boards are quite similar to other boards in Arduino family in terms of use and functionality, however, Uno boards don't come with FTDI USB to Serial driver chip.
- There are many versions of Uno boards available, however, Arduino Nano V3 and Arduino Uno are the most official versions that come with Atmega328 8-bit AVR Atmel microcontroller where RAM memory is 32KB.

- When nature and functionality of the task go complex, Micro SD card can be added in the boards to make them store more information.

4.3 Features of Arduino

- Arduino Uno comes with USB interface i.e. USB port is added on the board to develop serial communication with the computer.
- Atmega328 microcontroller is placed on the board that comes with a number of features like timers, counters, interrupts, PWM, CPU, I/O pins and based on a 16MHz clock that helps in producing more frequency and number of instructions per cycle.

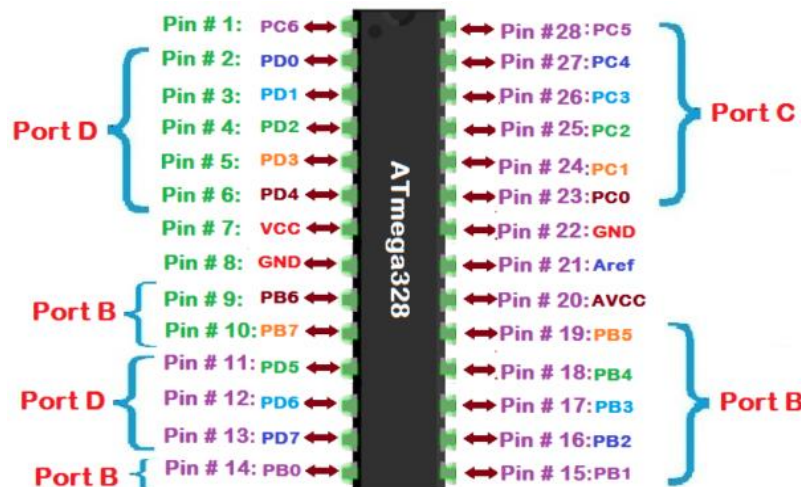


Fig 4.4 Atmega328

- It is an open source platform where anyone can modify and optimize the board based on the number of instructions and task they want to achieve.
- This board comes with a built-in regulation feature which keeps the voltage under control when the device is connected to the external device.
- Reset pin is added in the board that reset the whole board and takes the running program in the initial stage. This pin is useful when board hangs up in the middle of the running program; pushing this pin will clear everything up in the program and starts the program right from the beginning.

-
- The image shows an Arduino Uno R3 board with the following components labeled:
- DC Power jack**: Points to the power input jack.
 - USB Port**: Points to the USB Type-B port.
 - Reset Button**: Points to the circular reset button.
 - Pin Connections**: A legend for the pin headers:
 - Digital Input (Red)
 - Digital Output (Blue)
 - Analog Input (Green)
 - Ground Symbols (Black)
 - Pin Functions**: A table listing the functions for specific pins:

Pin	Function	Power
Digital Pin 0	Serial RX	
Digital Pin 1	Serial TX	
Digital Pin 2	Interrupt 2	
Digital Pin 3	Interrupt 3	
Digital Pin 4	SS	
Digital Pin 5	MCLR	
Digital Pin 6	SS	
Digital Pin 7	SS	
Digital Pin 8	SS	
Digital Pin 9	SS	
Digital Pin 10	SS	
Digital Pin 11	SS	
Digital Pin 12	SS	
Digital Pin 13	SS	
Analog Pin A0	AIN0	
Analog Pin A1	AIN1	
Analog Pin A2	AIN2	
Analog Pin A3	AIN3	
Analog Pin A4	AIN4	
Analog Pin A5	AIN5	
Power Pin GND	GND	Power
Power Pin 5V	5V	Power
Power Pin 3.3V	3.3V	Power

4.3.1 Pin Description

LED. Arduino Uno comes with built-in LED which is connected through pin 13. Providing HIGH value to the pin will turn it ON and LOW will turn it OFF.

24

5V. This board comes with the ability to provide voltage regulation. 5V pin is used to provide output regulated voltage. The board is powered up using three ways i.e. USB, Vin pin of the board or DC power jack.

USB supports voltage around 5V while Vin and Power Jack support a voltage ranges between 7V to 20V. It is recommended to operate the board on 5V. It is important to note that, if a voltage is supplied through 5V or 3.3V pins, they result in bypassing the voltage regulation that can damage the board if voltage surpasses from its limit.

GND. These are ground pins. More than one ground pins are provided on the board which can be used as per requirement.

Reset. This pin is incorporated on the board which resets the program running on the board. Instead of physical reset on the board, IDE comes with a feature of resetting the board through programming.

IOREF. This pin is very useful for providing voltage reference to the board. A shield is used to read the voltage across this pin which then select the proper power source.

PWM. PWM is provided by 3, 5, 6,9,10, 11pins. These pins are configured to provide 8-bit output PWM.

SPI. It is known as Serial Peripheral Interface. Four pins 10(SS), 11(MOSI), 12(MISO), 13(SCK) provide SPI communication with the help of SPI library.

AREF. It is called Analog Reference. This pin is used for providing a reference voltage to the analog inputs.

TWI. It is called Two-wire Interface. TWI communication is accessed through Wire Library. A4 and A5 pins are used for this purpose.

Serial Communication. Serial communication is carried out through two pins called Pin 0 (Rx) and Pin 1 (Tx).

Rx pin is used to receive data while Tx pin is used to transmit data.

External Interrupts. Pin 2 and 3 are used for providing external interrupts. An interrupt is called by providing LOW or changing value.

Table 4.1 Arduino Uno Technical Specifications

Microcontroller	<u>ATmega328P</u> – 8 bit AVR family microcontroller
Operating Voltage	5V
Recommended Input Voltage	7-12V
Input Voltage Limits	6-20V
Analog Input Pins	6 (A0 – A5)
Digital I/O Pins	14 (Out of which 6 provide PWM output)
DC Current on I/O Pins	40 mA
DC Current on 3.3V Pin	50 mA
Flash Memory	32 KB (0.5 KB is used for Bootloader)
SRAM	2 KB
EEPROM	1 KB
Frequency (Clock Speed)	16 MHz

4.3.3 Communication and Programming

Arduino Uno comes with an ability of interfacing with other other Arduino boards, microcontrollers and computer. The Atmega328 placed on the board provides serial communication using pins like Rx and Tx.

The Atmega16U2 incorporated on the board provides a pathway for serial communication using USB com drivers. Serial monitor is provided on the IDE software which is used to send or receive text data from the board. If LEDs placed on the Rx and Tx pins will flash, they indicate the transmission of data.

Arduino Uno is programmed using Arduino Software which a cross-platform application called IDE is written in Java. The AVR microcontroller Atmega328 laid out on the base comes with built-in boot loader that sets you free from using a separate burner to upload the program on the board.

4.4 Power supply

A power supply is a component that provides at least one electrical charge with power. It typically converts one type of electrical power to another, but it can also convert a different Energy form in electrical energy, such as solar, mechanical, or chemical.

A power supply provides electrical power to components. Usually the term refers to devices built into the powered component. Computer power supplies, for example, convert AC current to DC current and are generally located along with at least one fan at the back of the computer case.

Most computer power supplies also have an input voltage switch that, depending on the geographic location, can be set to 110v/115v or 220v/240v. Due to the different power voltages supplied by power outlets in different countries, this switch position is crucial.

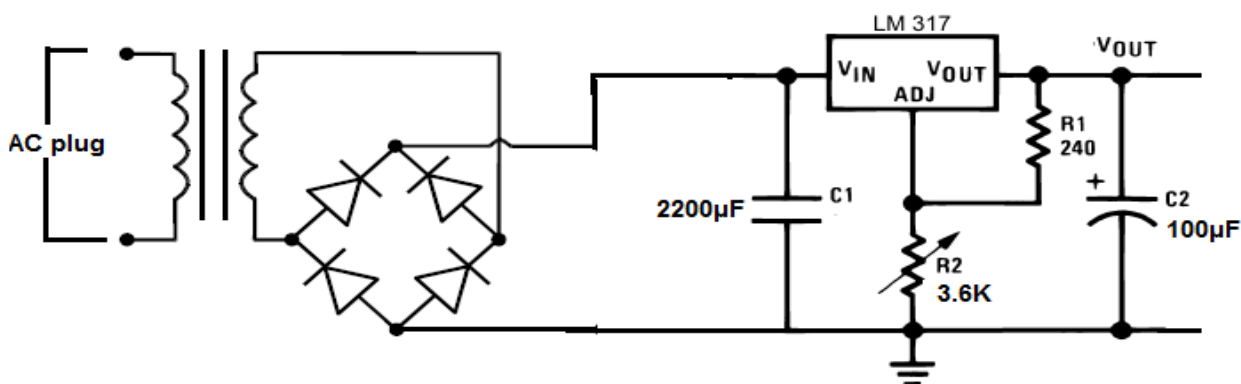


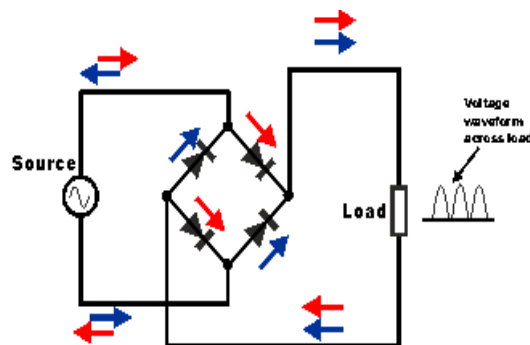
Fig 4.6 Circuit of Power supply

Some basic components used in the supply of power:

4.5 Rectifier

A **rectifier** is an electrical device that [converts alternating current](#) (AC), which periodically reverses direction, to [direct current](#) (DC), which flows in only one direction. The process is known as *rectification*, since it "straightens" the direction of current.

Rectifiers have many uses, but are often found to serve as components of DC power supplies and direct power transmission systems with high voltage. Rectification can be used in roles other than direct current generation for use as a power source.



Circuit of rectifier



Fig 4.7 Rectifier

4.6 Capacitors

Capacitors are used to attain from the connector the immaculate and smoothest DC voltage in which the rectifier is used to obtain throbbing DC voltage which is used as part of the light of the present identity. Capacitors are used to acquire square DC from the

current AC experience of the current channels so that they can be used as a touch of parallel yield.



Fig 4.8 Capacitor

4.7 Voltage regulators

The 78XX voltage controller is mainly used for voltage controllers as a whole. The XX speaks to the voltage delivered to the specific gadget by the voltage controller as the yield. 7805 will supply and control 5v yield voltage and 12v yield voltage will be created by 7812.

The voltage controllers are that their yield voltage as information requires no less than 2 volts. For example, 7805 as sources of information will require no less than 7V, and 7812, no less than 14 volts. This voltage is called Dropout Voltage, which should be given to voltage controllers.

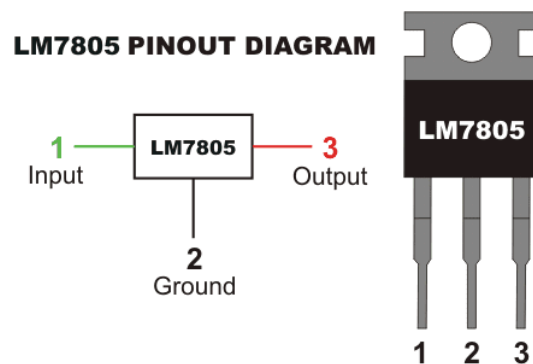


Fig 4.9 7805 voltage regulator with pinout

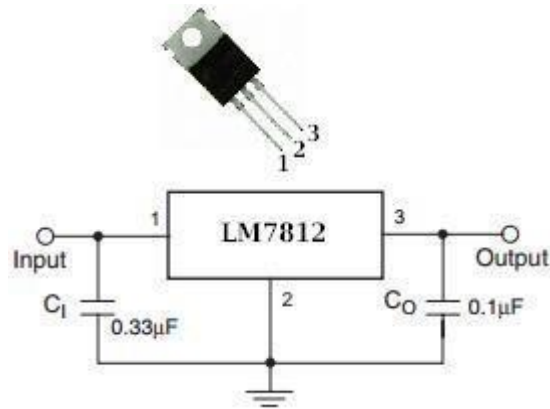


Fig 4.10 7812 voltage regulator with pinout

4.8 WEB CAMERA

A **webcam** is a video camera that feeds or streams an image or video in real time to or through a computer network, such as the Internet. Webcams are typically small cameras that sit on a desk, attach to a user's monitor, or are built into the hardware. Webcams can be used during a video chat session involving two or more people, with conversations that include live audio and video.

Webcam software enables users to record a video or stream the video on the Internet. As video streaming over the Internet requires much bandwidth, such streams usually use compressed formats. The maximum resolution of a webcam is also lower than most handheld video cameras, as higher resolutions would be reduced during transmission. The lower resolution enables webcams to be relatively inexpensive compared to most video cameras, but the effect is adequate for video chat sessions



Fig 4.11 Web cam

4.9 Ultrasonic sensor

An ultrasonic sensor transmit ultrasonic waves into the air and detects reflected waves from an object. There are many applications for ultrasonic sensors, such as in intrusion alarm systems, automatic door openers and backup sensors for automobiles.

Accompanied by the rapid development of information processing technology, new fields of application, such as factory automation equipment and car electronics, are increasing and should continue to do so. Using its unique piezoelectric ceramics manufacturing technology developed over many years, Murata has developed various types of ultrasonic sensors which are compact and yet have very high performance. The information contained in this catalog will help you to make effective use of our ultrasonic sensors.

HC-SR04 Sensor Features

- Operating voltage: +5V
- Theoretical Measuring Distance: 2cm to 450cm
- Practical Measuring Distance: 2cm to 80cm
- Accuracy: 3mm
- Measuring angle covered: <15°
- Operating Current: <15mA
- Operating Frequency: 40Hz

HC-SR04 Ultrasonic Sensor - Working

As shown above the HC-SR04 Ultrasonic (US) sensor is a 4 pin module, whose pin names are Vcc, Trigger, Echo and Ground respectively. This sensor is a very popular sensor used in many applications where measuring distance or sensing objects are required. The module has two eyes like projects in the front which forms the Ultrasonic transmitter and Receiver. The sensor works with the simple high school formula that

$$\text{Distance} = \text{Speed} \times \text{Time}$$

The Ultrasonic transmitter transmits an ultrasonic wave, this wave travels in air and when it gets objected by any material it gets reflected back toward the sensor this reflected wave is observed by the Ultrasonic receiver module as shown in the picture below



Fig 4.12 Sensor detecting

Now, to calculate the distance using the above formulae, we should know the Speed and time. Since we are using the Ultrasonic wave we know the universal speed of US wave at room conditions which is 330m/s. The circuitry inbuilt on the module will calculate the time taken for the US wave to come back and turns on the echo pin high for that same particular amount of time, this way we can also know the time taken. Now simply calculate the distance using a microcontroller or microprocessor.

HC-SR04 distance sensor is commonly used with both microcontroller and microprocessor platforms like Arduino, ARM, PIC, Raspberry Pie etc. The following guide is universally since it has to be followed irrespective of the type of computational device used.

Power the Sensor using a regulated +5V through the Vcc and Ground pins of the sensor. The current consumed by the sensor is less than 15mA and hence can be directly powered by the on board 5V pins (If available). The Trigger and the Echo pins are both I/O pins and hence they can be connected to I/O pins of the microcontroller. To start the measurement, the trigger pin has to be made high for 10uS and then turned off. This action will trigger an ultrasonic wave at frequency of 40Hz from the transmitter and the receiver will wait for the wave to return. Once the wave is returned after it getting reflected by any object the Echo pin goes high for a particular amount of time which will be equal to the time taken for the wave to return back to the sensor.

The amount of time during which the Echo pin stays high is measured by the MCU/MPU as it gives the information about the time taken for the wave to return back to the Sensor. Using this information the distance is measured as explained in the above heading.

4.10 LCD

LCD (Liquid Crystal Display) is the innovation utilized in scratch pad shows and other littler PCs. Like innovation for light-producing diode (LED) and gas-plasma, LCDs permit presentations to be a lot more slender than innovation for cathode beam tube (CRT). LCDs expend considerably less power than LED shows and gas shows since they work as opposed to emanating it on the guideline of blocking light.

A LCD is either made with an uninvolved lattice or a showcase network for dynamic framework show. Likewise alluded to as a meager film transistor (TFT) show is the dynamic framework LCD. The uninvolved LCD lattice has a matrix of conductors at every crossing point of the network with pixels. Two conductors on the lattice send a current to control the light for any pixel. A functioning framework has a transistor situated at every pixel crossing point, requiring less current to control the luminance of a pixel.

Some aloof network LCD's have double filtering, which implies they examine the matrix twice with current in the meantime as the first innovation took one sweep. Dynamic lattice, be that as it may, is as yet a higher innovation.

A 16x2 LCD show is an essential module that is generally utilized in various gadgets and circuits. These modules more than seven sections and other multi fragment LEDs are liked. The reasons being: LCDs are affordable; effectively programmable; have no restriction of showing exceptional and even custom characters (not at all like in seven fragments), movements, etc. A 16x2 LCD implies 16 characters can be shown per line and 2 such lines exist. Each character is shown in a lattice of 5x7 pixels in this LCD. There are two registers in this LCD, in particular Command and Data. The directions given to the LCD are put away by the order register. An order is a direction given to LCD to play out a predefined assignment, for example, introducing it, clearing its screen, setting the

situation of the cursor, controlling presentation, and so forth. The information register will store the information that will be shown on the LCD. The information is the character's ASCII incentive to show on the LCD.

Images of LCD Display

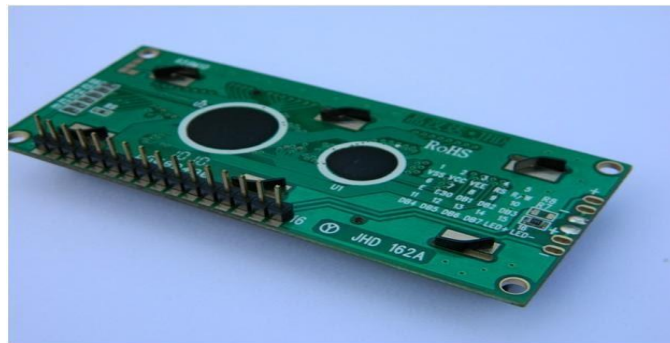


Fig 4.13 LCD Front View and Back View

Table 4.2 PIN DIAGRAM

Pin No	Function	Name
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	V _{cc}
3	Contrast adjustment; through a variable resistor	V _{EE}
4	Selects command register when low; and data register when high	Register Select
5	Low to write to the register; High to read from the register	Read/write
6	Sends data to data pins when a high to low pulse is given	Enable
7	8-bit data pins	DB0
8		DB1
9		DB2
10		DB3
11		DB4
12		DB5
13		DB6
14		DB7
15	Backlight V _{cc} (5V)	Led+
16	Backlight Ground (0V)	Led-

4.11 NodeMCU

NodeMCU is an open-source firmware and development kit that plays a vital role in designing your own IoT product using a few Lua script lines.

Multiple GPIO pins on the board allow you to connect the board with other peripherals and are capable of generating PWM, I2C, SPI, and UART serial communications. The interface of the module is mainly divided into two parts including both Firmware and Hardware where former runs on the ESP8266 Wi-Fi SoC and later is based on the ESP-12 module.

The firmware is based on Lua – A scripting language that is easy to learn, giving a simple programming environment layered with a fast scripting language that connects you with a well-known developer community. And open source firmware gives you the flexibility to edit, modify and rebuilt the existing module and keep changing the entire interface until you succeed in optimizing the module as per your requirements.

- USB to UART converter is added on the module that helps in converting USB data to UART data which mainly understands the language of serial communication.

Instead of the regular USB port, MicroUSB port is included in the module that connects it with the computer for dual purposes: programming and powering up the board.

- The board incorporates status LED that blinks and turns off immediately, giving you the current status of the module if it is running properly when connected with the computer.

The ability of module to establish a flawless WiFi connection between two channels makes it an ideal choice for incorporating it with other embedded devices like Raspberry Pi. NodeMCU comes with a number of GPIO Pins. Following figure shows the Pinout of the board.

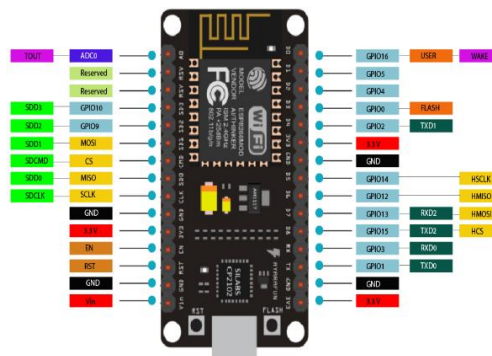


Fig 4.14 NodeMCU Pinout

- There is a candid difference between VIN and VU where former is the regulated voltage that may stand somewhere between 7 to 12 V while later is the power voltage for USB that must be kept around 5 V.

4.12 SOFTWARE REQUIREMENTS

Arduino IDE where IDE stands for Integrated Development Environment – An official software introduced by Arduino.cc, that is mainly used for writing, compiling and uploading the code in the Arduino Device. Almost all Arduino modules are compatible with this software that is an open source and is readily available to install and start compiling the code on the go.

- Arduino IDE is an open source software that is mainly used for writing and compiling the code into the Arduino Module.
- It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process.
- It is easily available for operating systems like MAC, Windows, and Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role for debugging, editing and compiling the code in the environment.
- A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro and many more.
- Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code.
- The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board.
- The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module.
- This environment supports both C and C++ languages.

How to install Arduino IDE:

You can download the Software from Arduino main website. As I said earlier, the software is available for common operating systems like Linux, Windows, and MAX, so make sure you are downloading the correct software version that is easily compatible with your operating system.

- If you aim to download Windows app version, make sure you have Windows 8.1 or Windows 10, as app version is not compatible with Windows 7 or older version of this operating system.

The IDE environment is mainly distributed into three sections

- **1. Menu Bar**
- **2. Text Editor**
- **3. Output Pane**

As you download and open the IDE software, it will appear like an image below.

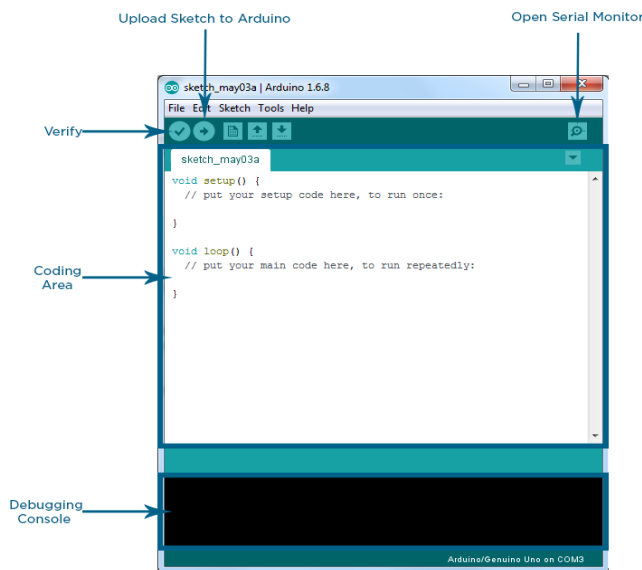


Fig 4.15 Arduino Interface

The bar appearing on the top is called **Menu Bar** that comes with five different options as follow

- **File** – You can open a new window for writing the code or open an existing one. Following table shows the number of further subdivisions the file option is categorized into.

As you go to the preference section and check the compilation section, the Output Pane will show the code compilation as you click the upload button.

And at the end of compilation, it will show you the hex file it has generated for the recent sketch that will send to the Arduino Board for the specific task you aim to achieve.

- **Edit** – Used for copying and pasting the code with further modification for font
- **Sketch** – For compiling and programming
- **Tools** – Mainly used for testing projects. The Programmer section in this panel is used for burning a bootloader to the new microcontroller.
- **Help** – In case you are feeling skeptical about software, complete help is available from getting started to troubleshooting.

The **Six Buttons** appearing under the Menu tab are connected with the running program as follow.

- The check mark appearing in the circular button is used to verify the code. Click this once you have written your code.
- The arrow key will upload and transfer the required code to the Arduino board.
- The dotted paper is used for creating a new file.
- The upward arrow is reserved for opening an existing Arduino project.
- The downward arrow is used to save the current running code.
- The button appearing on the top right corner is a **Serial Monitor** – A separate pop-up window that acts as an independent terminal and plays a vital role for sending and receiving the Serial Data. You can also go to the Tools panel and select Serial Monitor, or pressing Ctrl+Shift+M all at once will open it instantly. The Serial Monitor will actually help to debug the written Sketches where you can get a hold of how your program is operating. Your Arduino Module should be connected to your computer by USB cable in order to activate the Serial Monitor.

- You need to select the baud rate of the Arduino Board you are using right now. For my Arduino Uno Baud Rate is 9600, as you write the following code and click the Serial Monitor, the output will show as the image below.



Fig 4.17 COM 5

The main screen below the Menu bard is known as a simple text editor used for writing the required code.

CHAPTER 5

RESULT AND ANALYSIS

5.1 RESULT:

Implementing an SMART Vehicle Parking Monitoring and Management System using ANPR cameras can be significant. Some of the possible results

Improved parking management: With the ANPR cameras and software system, parking operators can have real-time information on the occupancy and availability of parking spaces. This allows them to better manage the parking area, ensuring that spaces are allocated efficiently and that customers have a positive parking experience.

Increased revenue: By optimizing the use of parking spaces, the Automated Vehicle Parking Monitoring and Management System can help increase revenue for the parking operator. The system can provide insights into peak usage periods, allowing for the implementation of dynamic pricing strategies that can maximize revenue generation.

Improved security: ANPR cameras can provide enhanced security for the parking area by capturing images of vehicles and their license plates as they enter and exit. This can help deter unauthorized access and improve safety for customers and their vehicles.



Fig 5.1 Parking System Model

Reduced labor costs: By automating the parking management process, the need for manual monitoring and management is reduced. This can lead to significant cost savings for the parking operator

Enhanced customer experience: With improved parking management, increased revenue, and enhanced security, customers are likely to have a better parking experience overall. This can lead to increased customer satisfaction and loyalty.

Overall, an SMART Vehicle Parking Monitoring and Management System using ANPR cameras can bring numerous benefits to parking operators and customers alike. The system can improve efficiency, revenue, security, and customer experience, making it a valuable investment for parking facilities of all types and sizes.

CHAPTER 6

CONCLUSION AND FUTURE SCOPES

6.1 Conclusion

The concept of Smart Cities has always been a dream for humanity. Over the past couple of years, large advancements have been made in making smart cities a reality. The growth of the Internet of Things and Cloud technologies have given rise to new possibilities in terms of smart cities. Smart parking facilities and traffic management systems have always been at the core of constructing smart cities. In this Project, we address the issue of parking and present an IoT-based Cloud integrated smart parking system. The system that we propose provides real-time information regarding the availability of parking slots in a parking area. Users from remote locations could book a parking slot for them by the use of our mobile application.

The efforts made in this paper are indented to improve the parking facilities of a city and thereby aim to enhance the quality of life of its people.

6.2 Future Scope:

We used Segregation Algorithm based on Empty slots and number plate tracking. This consumes a high database work so in further cases this history management can be replaced with temporary data Live updating of the system will end once the customer gets his vehicle out of the parking unit This minor error from our proposed system made a huge impact on the memory allocation part Such things must be taken into notice when this project is presented in real-time

REFERENCES

- [1] Faiz Ibrahim Shaikh, Pratik Nirnay Jadhav, Saideep Pradeep Bandarkar, Omkar Pradip Kulkarni, Nikhilkumar B. Shardoor "Smart Parking System Based on Embedded System and Sensor Network", International Journal of Computer Applications (0975 – 8887) Volume 140 – No.12, April 2016 International Journal of Pure and Applied Mathematics Special Issue 171.
- [2] Thanh Nam Pham¹, Ming-Fong Tsai¹, Duc Binh Nguyen¹, ChyiRen Dow¹, And Der-Jiunn Deng² "A Cloud-Based Smart-Parking System Based on Internet-of-Things Technologies", IEEE Access, Received July 24, 2015, accepted August 16, 2015, date of publication September 9, 2015, date of current version September 23, 2015.
- [3] El Mouatezbillah Karbab, Djamel Djenouri, Sahar Boulkaboul, Antoine Bagula, CERIST Research Center, Algiers, Algeria University of the Western Cape, Cape town, South Africa,"Car Park Management with Networked Wireless Sensors and Active RFID",,978-1-4799-8802-0/15 ©2015 IEEE
- [4] Mr. Basavaraju S R "Automatic Smart Parking System using Internet of Things (IOT)", (International Journal of Scientific and Research Publications, Volume 5, Issue 12, December 2015)
- [5] M. M. Rashid, A. Musa, M. Ataur Rahman, and N. Farahana, A. Farhana, "Automatic Parking Management System and Parking Fee Collection Based on Number Plate Recognition.", International Journal of Machine Learning and Computing, Vol. 2, No. 2, April 2012, Published 2014.
- [6] Hilal Al-Kharusi, Ibrahim Al-Bahadly, "Intelligent Parking Management System Based on Image Processing", World Journal of Engineering and Technology, 2014, 2, 55-67.
- [7] X. Zhao, K. Zhao, and F. Hai, "An algorithm of parking planning for smart parking system," in Proc. 11th World Congr. Intell. Control Autom. (WCICA), 2014, pp. 4965_4969.
- [8] L. Mainetti, L. Palano, L. Patrono, M. L. Stefanizzi, and R. Vergallo, "Integration of RFID and WSN technologies in a smart parking system," in Proc. 22nd Int.

Conf. Software., Telecommun. Comput. Netw. (SoftCOM), 2014, pp. 104_110.

- [9] Harmeet Singh, Chetan Anand, Vinay Kumar, Ankit Sharma, "Automated Parking System With Bluetooth Access", International Journal Of Engineering And Computer Science ISSN:2319-7242,Volume 3 Issue 5, May 2014, Page No. 5773- 5775.