SMART PARKING SYSTEM USING IoT

A PROJECT REPORT

Submitted By

ANNAPURNA SIRISHA D 312212104011

ARCHANA B 312212104012

JAYASHREE S 312212104036

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

SSN COLLEGE OF ENGINEERING

KALAVAKKAM 603110

ANNA UNIVERSITY :: CHENNAI - 600025

April 2016

ANNA UNIVERSITY : CHENNAI 600025

BONAFIDE CERTIFICATE

Certified that this project report titled **"SMART PARKING SYSTEM USING IoT"** is the *bonafide* work of "**ANNAPURNA SIRISHA D (31221210411)**, **ARCHANA B (312212104012)**, and **JAYASHREE S (312212104036)**" who carried out the project work under my supervision.

DR. CHITRA BABU
HEAD OF THE DEPARTMENT
Professor,
Department of CSE,
SSN College of Engineering,
Kalavakkam - 603 110

DR. T. T. MIRNALINEE
SUPERVISOR
Professor,
Department of CSE,
SSN College of Engineering,
Kalavakkam - 603 110

Place:
Date:

Submitted for the examination held on. . . . . .

Internal Examiner **External Examiner**

ACKNOWLEDGEMENTS

We thank GOD, the almighty for giving us strength and knowledge to do this project.

We would like to thank and deep sense of gratitude to our project guide **Dr. T. T. MIRNALINEE**, Professor, Department of Computer Science and Engineering, for her valuable advice and suggestions as well as her continued guidance, patience and support that helped us to shape and refine our work.

Our sincere thanks to **Dr. CHITRA BABU**, Professor and Head of the Department of Computer Science and Engineering, for her words of advice and encouragement and we would like to thank our project Coordinator **Dr. S. SHEERAZUDDIN**, Associate Professor, Department of Computer Science and Engineering for his valuable suggestions throughout this first phase of project.

We express our deep respect to the founder **Dr. SHIV NADAR**, Chairman, SSN Institutions. We also express our appreciation to **Dr. S. SALIVAHANAN**, Principal, for all the help he has rendered during this course of study.

We would like to extend our sincere thanks to all the teaching and non-teaching staffs of our department who have contributed directly and indirectly during the course of our project work. Finally, we would like to thank our parents and friends for their patience, cooperation and moral support throughout our life.

Annapurna Sirisha D **Archana B** **Jayashree S**

ABSTRACT

Due to proliferation of vehicles, India is facing a new problem nowadays. With the families getting smaller and the total number of motor vehicles exceeding the total number of heads per family, the parking scenario is woefully falling short of the current requirements. Any small or large space found on the road is used as a parking slot. This narrows the width of the road and leads to traffic problems. Hence it is very important to find a proper solution to accommodate more number of vehicles and also to reduce the time in finding a free space especially at crowded areas like shopping malls, airports , railway stations or any office parking area. This may include more man power to regulate the traffic at this increasing rate. It is beneficial to use smart system for parking rather than depending on human power.The smart parking system is automated system that optimises parking space usage by accurately sensing vehicle occupancy in real time. This provides a smart solution to the most time spent in searching for a parking slot as the machine chooses the closest parking slot to the vehicle entrance thereby reducing cost of fuel and time.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The Parking system is an area allocated to park the vehicles in an organised manner.Parking is one of the major problem faced by almost all drivers. Inefficient allocation of space, lack of knowledge about parking availability,and dangerous zones all greatly contribute to parking problems.A Smart Parking System (SPS) is used to assist drivers in finding and paying for parking slots[4]. SPS provides the most efficient solution to the parking problem[2]. The SPS has a capacity to become a one-stop driving and parking solutions provider for frustrated drivers[7].

1.1 Motivation

Number of motor vehicles exceeds the total number of heads per family.In order to accommodate more number of vehicles and also to reduce the time in finding a free space especially at crowded areas, there is a need for a digitalized parking system.This type of parking system helps to reduce the man power and provides a dynamic and user friendly solution. This proposal is a support to the Digital India Group, a vision of Digital India started by honourable Prime Minister Narendra Modi that encourages innovative ideas and practical solutions.The smart parking system is digitalised system that optimises parking space usage by accurately sensing vehicle occupancy in real time. To reduce the cost of fuel and time.

1.2 Problem Statement

Smart Parking System is the need of the hour. There have been ideas proposed to address this problem using technologies like WSN, Ultrasonic detectors, Image processing of surveillance cameras. To design a Smart Parking System(SPS) that facilitates the parking system in urban areas in an organised and efficient manner using Infrared sensors,RFID for unique user identification, automatic payment system, LCD displays. The system design should reduce the congestion in parking slots and also to reduce fuel and time consumption by allocating a vacant parking slot to one vehicle at a time using the parking unit occupancy sensors. Automated payment system should calculate the amount with the arrival time, departure time and prehistoric data.

1.3 Scope of the project

This particular project takes into account of the Indian scenario and is designed with many constraints such as less fuel consumption, utilizing the entire parking slots,time required in finding the vacant slot. It provides more cost effective, feasible, safer and reliable solution compared to the existing system.By installing SPS the average revenue increases and the economical payback period is less which will increase the overall profit of the system. This will result in effective installation maintenance and monitoring of the system.Smart Eye sensors can be used for accurate sensing and cloud can be used for storage. Mobile apps can be used to guide and provide various services to the user. We can embed Google Maps into mobile application so that it helps the user to see the directions through

Maps to reach the correct parking slot.GPS(Global Position System) and Navigation helps the user to track the route from current position to allocated slot.

1.4 Organisation of the Report

We discuss about the literature survey in chapter 2, which talks about the parking system solutions and also the various experiments and research activities conducted in this field. Chapter 3 talks about our proposed system, its architecture and its working.Result analysis is presented in chapter 4. Chapter 5 consists of our projects conclusion and possible future works.

CHAPTER 2

LITERATURE SURVEY

Some of the existing parking system had proposed solutions in various ways to solve the issues addressed by the drivers. Almost all issues confronted by a driver during parking has been addressed by one or the other technology but this smart parking system has not been implemented in India because of the expensiveness of the project and technologies involved. A smart parking system involving cost effective technologies and also to adapt the Indian scenario is the main objective of this project.

In this section we review the literature on existing and proposed parking lot applications based on WSN.

Lee, Yoon and Ghosh proposed a hybrid approach [7] For an intelligent parking system using a combination of ultrasonic and magnetic sensors. They demonstrated promising results through various real world experiments and showed that these hybrid solutions are more practical and accurate. Though the main goal was to count the number of vehicles on each floor and provide a cheap and accurate solution, the scope of their work restricts itself to vehicle detection using WSN than providing a smarter parking management solution.

D-Systems Project [2] in his paper presented various issues for a reliable WSN system using magnetic sensors. As part of this project an implementation of a car-park management system using a tiered architecture is detailed using magnetic sensor boards. of dynamic robust routing, delayed retransmissions, etc.

Jatuporn et al. [4], proposed optical WSN as a vehicle counting system in smart parking garages. Their proposed solution is to add another optical sensor head to the wireless sensor node. Two sensor heads were used to classify the objects of different sizes thereby distinguishing and identifying different vehicles.

Vanessa W.S. Tang, in his paper[8], described a WSN-based intelligent car parking system. In the system, wireless sensors were deployed into a car park field, with each parking lot equipped with one sensor node, which detects and monitors the occupation of the parking lot. The status of the parking field detected by sensor nodes is reported periodically to a database via the deployed wireless sensor network and its gateway. The database can be accessed by the upper layer management system to perform various management functions, such as finding vacant parking lots, auto-toll, security management, and statistic report. But this system has not addressed the efficient way to find the vacant slots and payment system.

Ramneet Kaur [6], in his paper the parking system is implemented using Finite State Machine modeling. The system has two main modules i.e. identification module and slot checking module. Identification module identifies the visitor. RF module is used to transmit and receive slot availability information. Slot checking module checks the slot status. These modules are modeled in HDL and implemented on FPGA. A prototype of parking system is designed with various interfaces like sensor interfacing, stepper motor and LCD. Again the most shortest vacant slot is not identified and the directions were also not given.

Mala Aggarwal[5], in his paper has described the Automatic Car Parking that enables the parking of vehicles-floor after floor and thus reducing the space used. Here any number of cars can be parked according to requirement. These make the

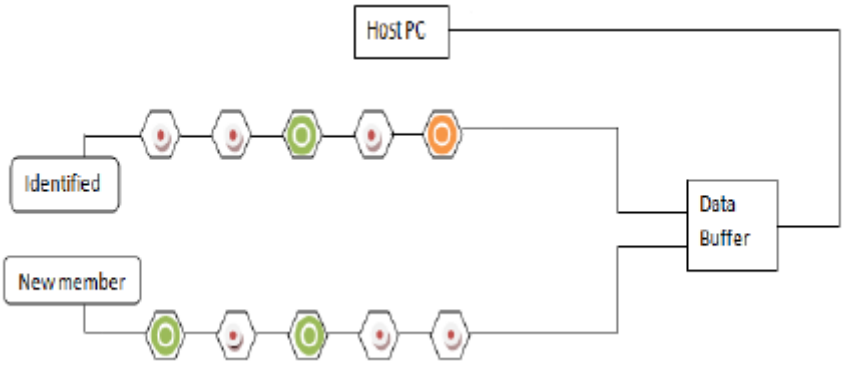


FIGURE 2.1: Block diagram of the parking system

system modernized and even a space-saving one. This idea is developed using AVR Microcontroller. Here program is written according to this idea using AVR ATMEGA 16 microcontroller. Mathematical modeling is also done to identify the least car parking space available among the difference parking places in a city. But one major drawback with the automated parking system is that it requires big lifting machines which consumes more fuel and power which is not cost effective.

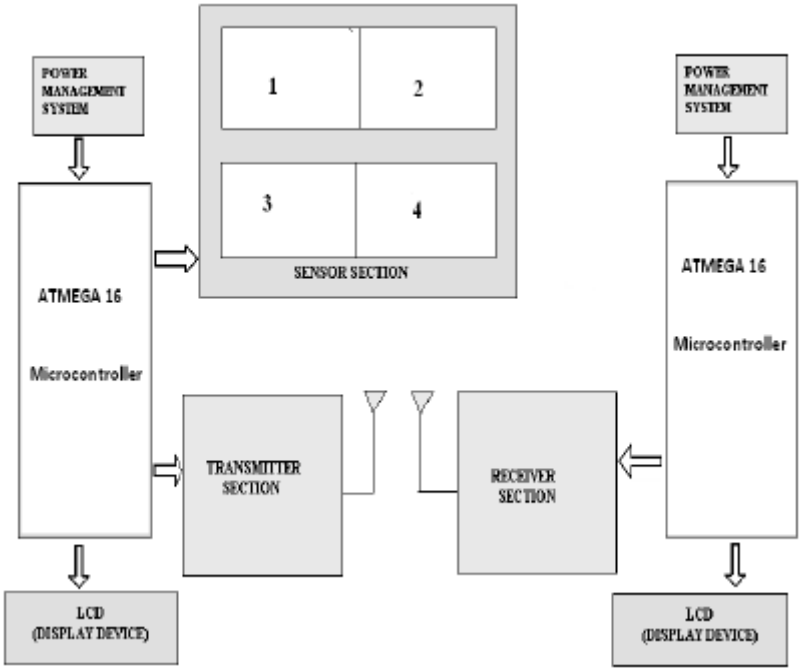


FIGURE 2.2: Automatic car parking system

This paper also shows that the pre existing security surveillance (CCTVs) will be used as a sensing nodes to identify vacant parking space. The captured image will be processed through the AVR Microcontroller and the processed data will be transmitted via ZigBee to a central computer to store and update the occupancy status of available parking space vacancies in the database. Here are two LCD displays, one is for transmitter section and another is for receiver section. A display is provided at the ground floor which is basically a counter that displays number of cars in each floor. It informs whether the floors are fully filled with the cars or is it having place in a particular floor or not. There is facility of lift to carry the car to up and down. Movement of Lift is controlled by dc motor.

P.Vijay Kumar [9] in his paper the system shall be able to graphically display real time information related to the availability of parking lots to the users and would also enable users to reserve parking lot from remote locations. WSN subsystem mainly deals with monitoring of parking status. This subsystem detects the status of parking space with hybrid sensing techniques and transmits status information through RF. It also receives commands from parking management subsystem to carry out various procedures. The subsystem internally consists of four major modules which include sensing, routing, dissemination and status modules.

Hongwei Wang [3] in his paper designed and implemented a prototype of Reservation-based Smart Parking System (RSPS) that allows drivers to effectively find and reserve the vacant parking spaces. By periodically learning the parking status from the sensor networks deployed in parking lots, the reservation service is affected by the change of physical parking status.

Amin Kianpisheh [1] in his paper Ultrasonic sensors transmit sound waves between 25 kHz and 50 kHz. They use the reflected energy to analyze and detect

the status of a parking space. Ultrasonic waves are emitted from the head of an ultrasonic vehicle detection sensor every 60 milliseconds, and the presence or absence of vehicles is determined by time differences between the emitted and received signals. Ultrasonic sensors can be used for counting vehicles and assessing the occupancy status of each parking space [10]. Despite the low cost and easy installation of ultrasonic sensors, they do have some disadvantages, particularly sensitivity to temperature changes and extreme air turbulence.

The ideas discussed above focus on technologies such as WSN, Finite State Machine Modelling, Automated car parking using Zigbee, AVR Microcontrollers and Reservation based parking systems. These techniques doesnt fit into the Indian scenario and are not user friendly, cost effective. Also the vacancy is not filled efficiently and dynamically. The users are also not uniquely identified. Thus our SPS addresses all the above discussed problems

CHAPTER 3

PROPOSED SYSTEM

3.1 Smart Parking System

For the proposed idea, IR sensors are placed at each parking slot to obtain the status of the parking slots in the parking area.The status is with regard to whether the slot is occupied or vacant. This sensor data is then transmitted wirelessly to the DSS. The Java Application then retrieves the sensor data from the DSS and uses this information for allocating the slot efficiently.RFID (Radio Frequency Identification) tag is used to uniquely identify the user and use this information for communicating with the user directly. A Mail is sent to provide the directions to the allocated slot,payment and parking information.LCD Display placed at the entrance of the parking which is used to display the directions to the user and available vacant slots at any instant.

3.2 Architecture Diagram

The architecture diagram of the proposed system is shown in fig 3.2. The system's architecture includes wireless sensor network, RFID module, Decision Support System(DSS), ESP8266 and an LCD Display to display directions received from DSS to to user.

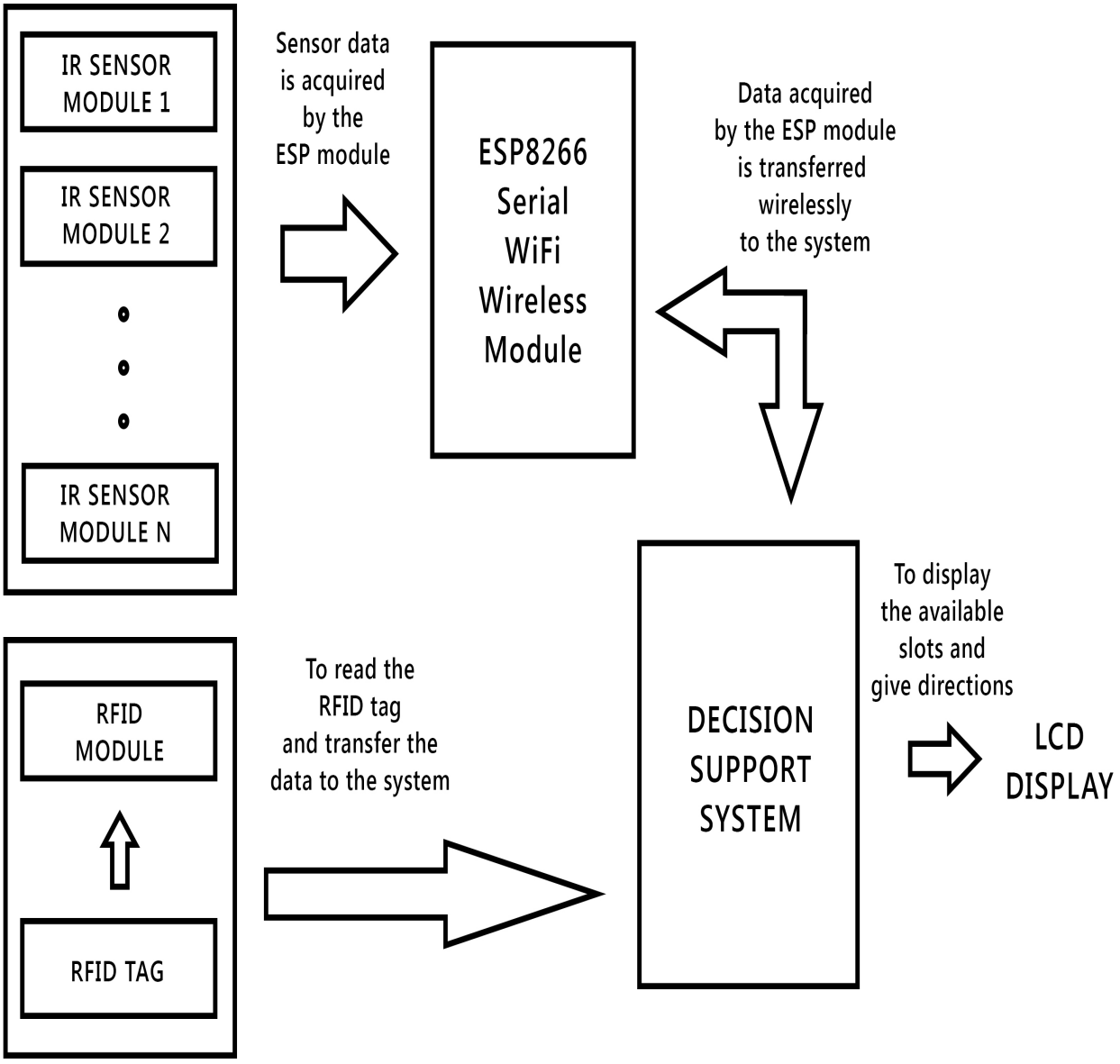


FIGURE 3.1: Architecture Diagram

3.3 Algorithm for the proposed system

Begin

1. User enters the parking system.
2. Read the RFID tag.
3. Transmit the RFID information to the Java application.
4. ESP module reads Sensor data.
5. ESP transmits Sensor data to TCP Server over WiFi.
6. Update data to database.
7. Transfer control to Scheduler module.
8. Allocate the closest vacant parking slot to the user using Shortest path algorithm.
9. Determine directions to the slot.
10. Update the user and slot information in the database.
11. Transfer the control to controller module.
12. Facilitate the instructions to the LCD Display.

End

3.4 Flowchart of the Proposed System

The work flow of the proposed system is depicted in the Fig 3.2.It starts with the user entering the parking area and ends with system communicating with the user via E-mail.

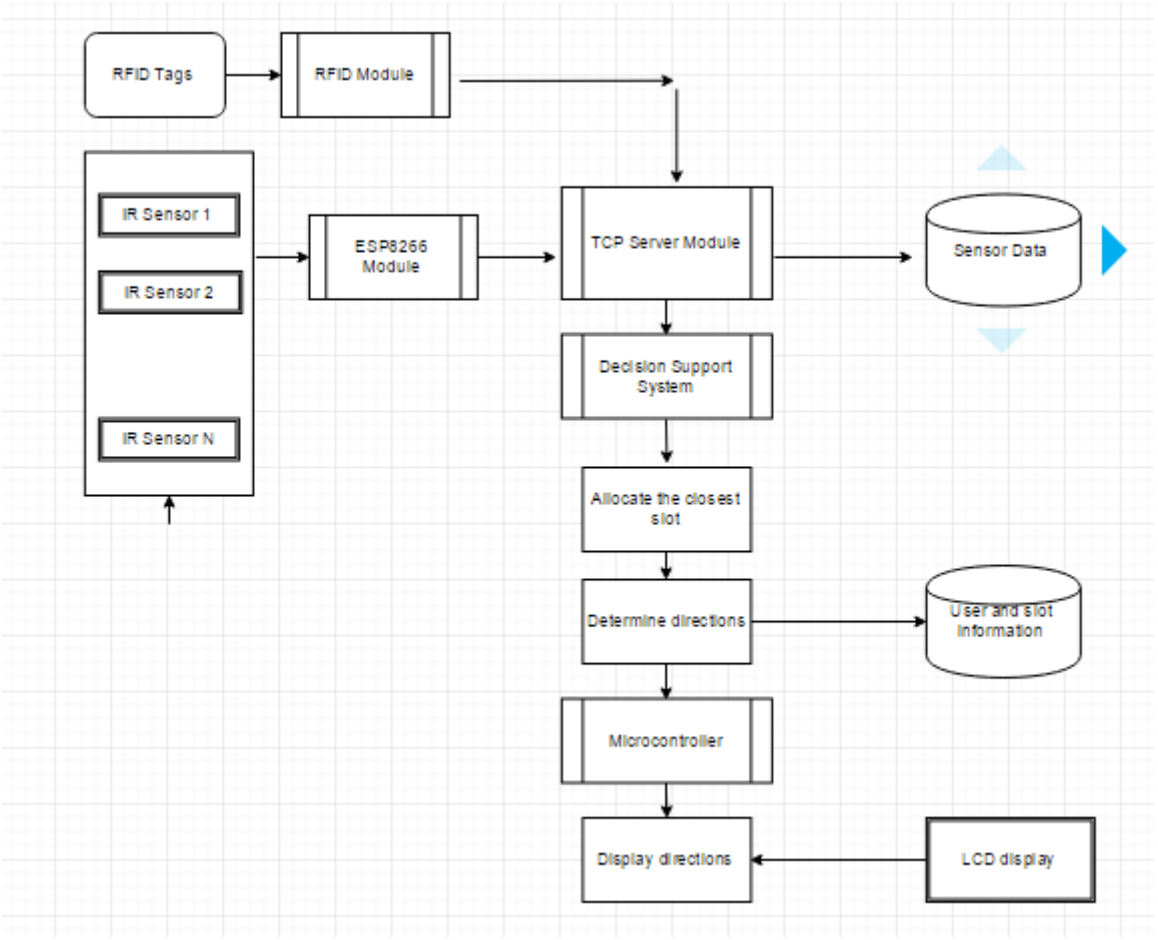


FIGURE 3.2: Flowchart of the Proposed System

3.5 Wireless Sensor Network(Collection of Nodes)

The two main components of the system is shown in Fig3.3 and 3.4,IR Sensor module and ESP8266 a wireless transceiver.

3.5.1 IR Sensor Module

An IR Sensor module is a sensor that transmits and receives infra-red rays when a surface or object is detected. It consists of IR transmitter, receiver, IC LM358 and resistors.When some object comes in the (IR) infrared range, the IR waves hits the

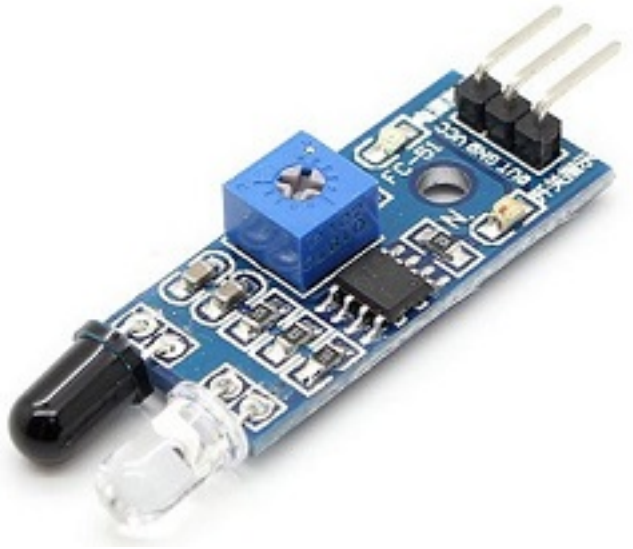


FIGURE 3.3: IR Sensor Module

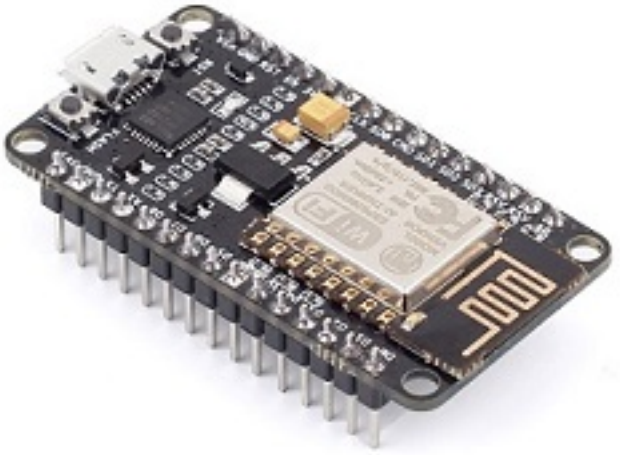


FIGURE 3.4: ESP8266

object and comes back at some angle. The Photo diode next to IR LED detects that IR infrared rays which got reflected from the object and hence works as a proximity sensor. IC LM358 is responsible for converting the analog signal from the transmitter and receiver into a digital output. In this case, an LED connected to the 1st pin of LM358 must glow as the IR transmitter emits IR rays and IR receiver receives the reflected ray.The Op-Amp (IC LM358) compares the two inputs and

provides the output which is higher.

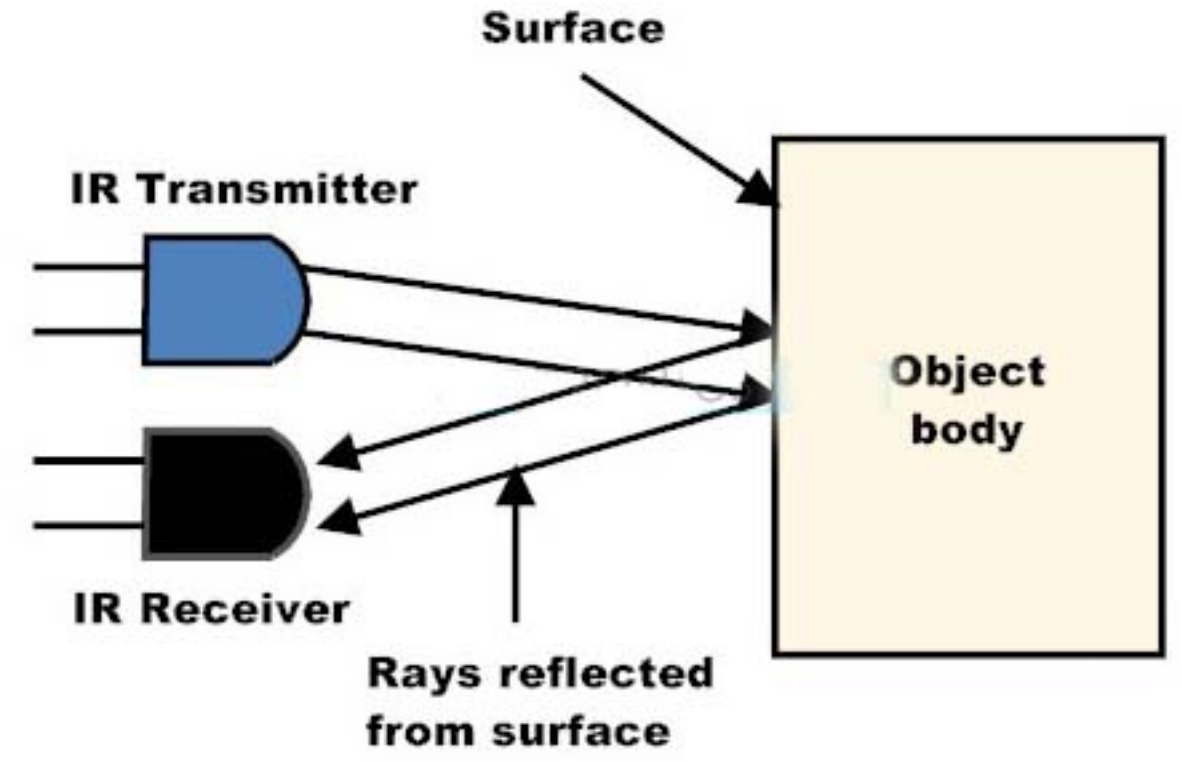


FIGURE 3.5: IR sensing of the object

3.5.2 ESP8266

ESP8266 is a tiny chip with an embedded ARM microprocessor and a WiFi stack all miniaturized into one tiny surface mount chip.
Feature set

- 802.11 b / g / n
- WiFi Direct (P2P), soft-AP

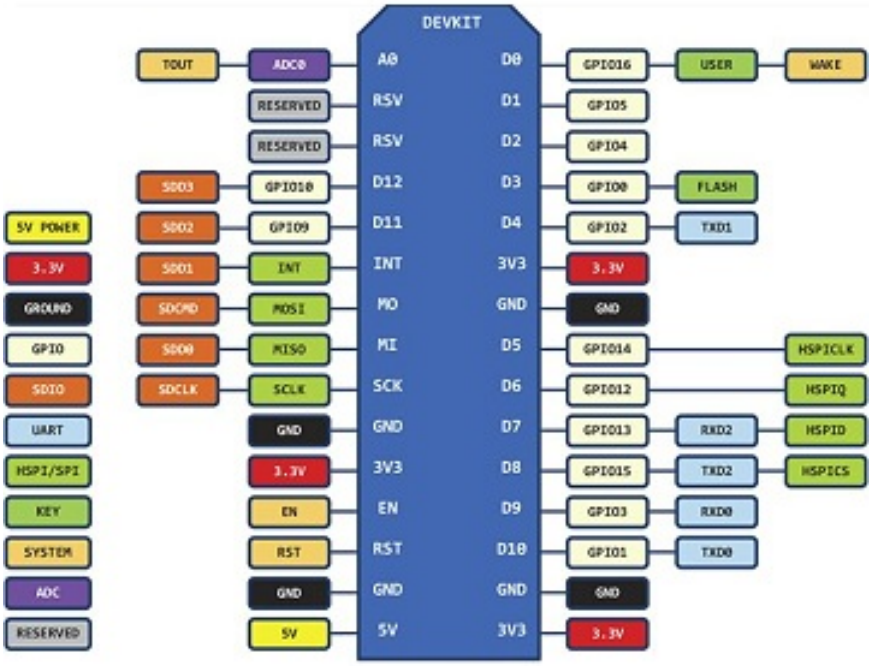


FIGURE 3.6: ESP8266 Pin Diagram

- Built-in TCP / IP protocol stack
- Built-in TR switch, balun, LNA, power amplifier and matching network
- Built-in PLL, voltage regulator and power management components
- 802.11b mode + 19.5dBm output power
- Built-in temperature sensor
- Support antenna diversity
- Off leakage current is less than 10uA
- Built-in low-power 32-bit CPU: can double as an application processor
- SDIO 2.0, SPI, UART

- STBC, 11 MIMO, 21 MIMO
- A-MPDU, A-MSDU aggregation and the 0.4 Within wake 2ms,connect and transfer data packets
- Standby power consumption of less than 1.0mW (DTIM3)

The ESP8266 WiFi Module is a self contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your WiFi network. The ESP8266 is capable of either hosting an application or offloading all WiFi networking functions from another application processor. This module has a powerful enough on-board processing and storage capability that allows it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development upfront and minimal loading during runtime.
The ESP can also perform complex operations like running a program.This can be achieved by flashing it with the other available firmwares.One such firmware is the NodeMCU firmware available online. Upon flashing with NODEMCU firmware, the ESP is now able to run LUA programs. The VCC and GND are used to power the chip, voltage ranging between 5V with a maximum consumption of 300mA.The RX and TX pins are used to communicate or program the ESP8266 using serial communication.GPIO0 and GPIO16 are General Purpose Input Output pins which can be used for input output purposes.
The ESP can work in two modes, ie Station and AP. In Station mode it connects to an existing access point.In Access point mode, it creates an access point for other devices to connect with it.Here we use our device in station mode and connect it with an access point by specifying its ssid and password. Then we create a TCP connection.This esp acts as a TCP client. Then we connect to the

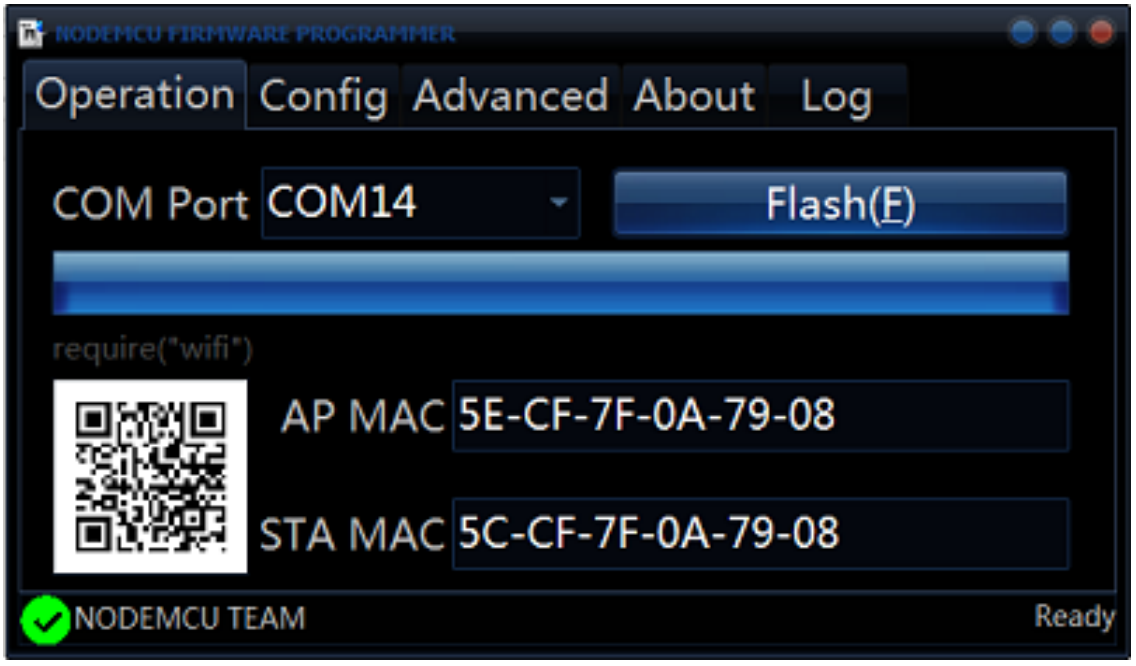


FIGURE 3.7: NodeMCU Flasher

TCP server by specifying its IP address and port. ESPlorer IDE is used to write the code in LUA programming language and to burn the code into ESP module. This IDE is explicit for ESP8266. LUA is a programming language to communicate and command the ESP module. Following are the code snippets to perform certain operations.

3.5.3 LUA code for opening and writing in the init.lua file

file.remove (” init.lua”)

file.open (” init.lua”,”w”)

file.write ([[print (”hi ”)]])

file.close ()

node.restart ()

3.5.4 LUA code for connecting to a WiFi AccessPoint and connecting to a TCP-server

```
wifi.setmode(wifi.STATION)
wifi.sta.config (”ssid”,”password”)
conn=net. createConnection (net.TCP , 0)
conn:connect (6789 ,”192.168.1.2”)
```

3.5.5 LUA code to send data over WiFi

```
conn:send (”sensor data is,”.. s1 ..” ,” .. s2 .. ”n”)
conn:send (”sensor data is,”.. s4 ..” ,” .. s5 .. ”n”)
```

3.6 RFID(Radio Frequency Identification)

Radio-frequency identification (RFID) uses electromagnetic fields to automatically identify and track tags attached to objects. The tags contain electronically stored information. Unlike a barcode, the tag need not be within the line of sight of the reader, so it may be embedded in the tracked object. RFID is one method for Automatic Identification and Data Capture (AIDC).

3.6.1 RFID Reader Module

An RFID reader is a network connected device (fixed or mobile) with an antenna that sends power as well as data and commands to the tags.

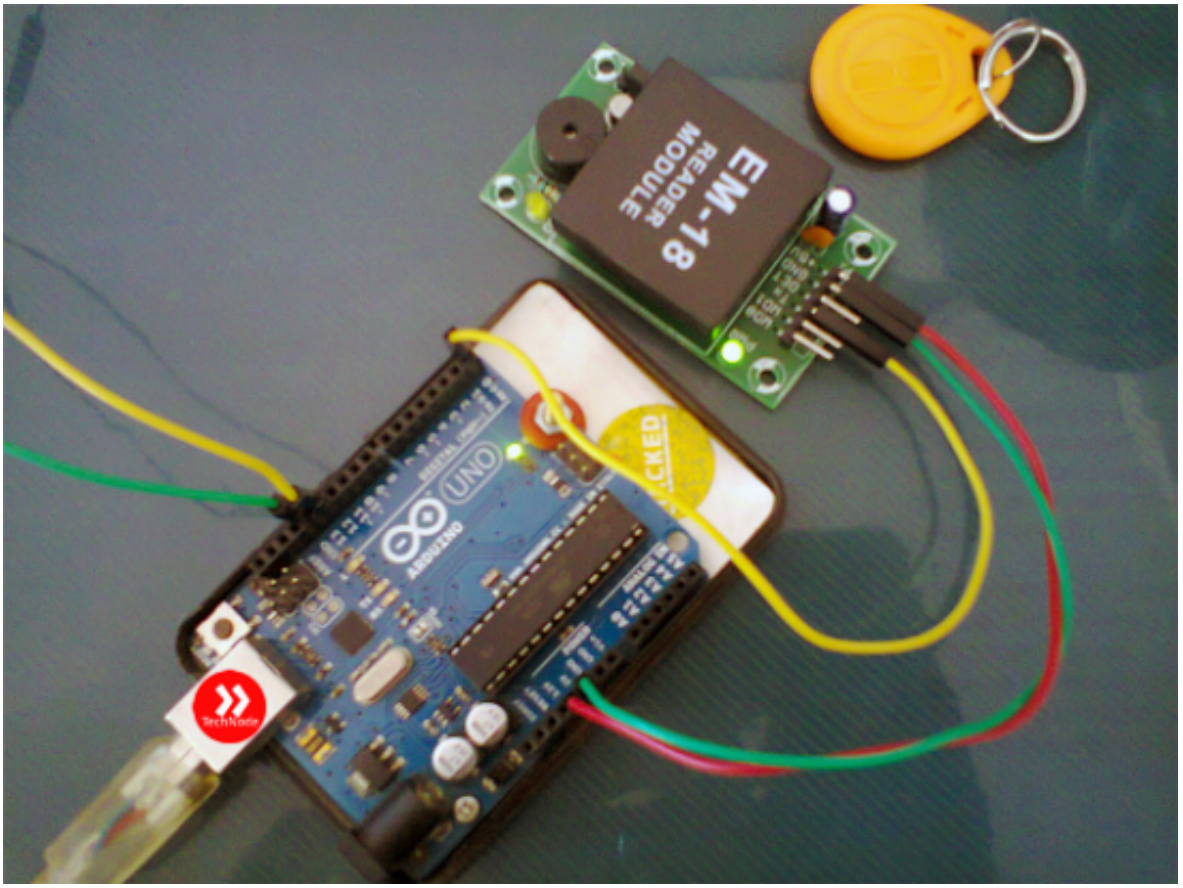


FIGURE 3.8: Interfacing RFID with Arduino

This is a device that provides the connection between the tag data and the enterprise system software that needs the information. The reader communicates with tags that are within its field of operation, performing any number of tasks including simple continuous inventorying, filtering (searching for tags that meet certain criteria), writing (or encoding) to selected tags, etc.

The user carries an RFID Tag which is unique to every user. When he enters the system, he points it to the RFID reader which recognizes the user uniquely and sends this information to the DSS.

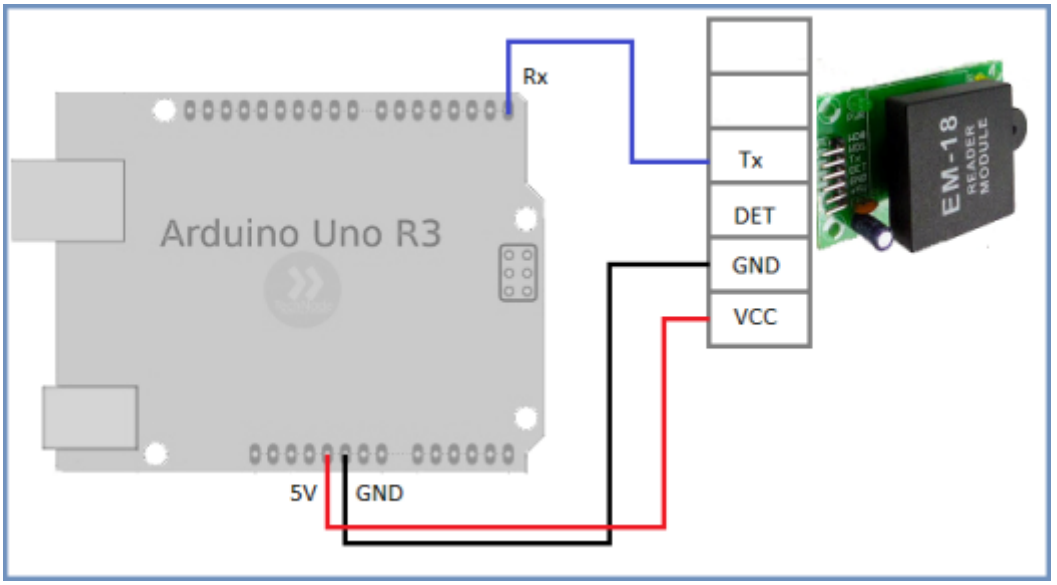


FIGURE 3.9: Connecting RFID with Arduino

FIGURE 3.10: RFID Tag

3.6.2 RFID Tags

An RFID tag is comprised of an integrated circuit (called an IC or chip) attached to an antenna that has been printed. The chip and antenna combo, called an inlay, is then converted and sandwiched between a printed label and inserted into a more durable structure.

The electronic product code (EPC) stored in the tag chip's memory is written to the tag by an RFID printer and takes the form of a 96-bit string of data.

3.6.3 Arduino UNO

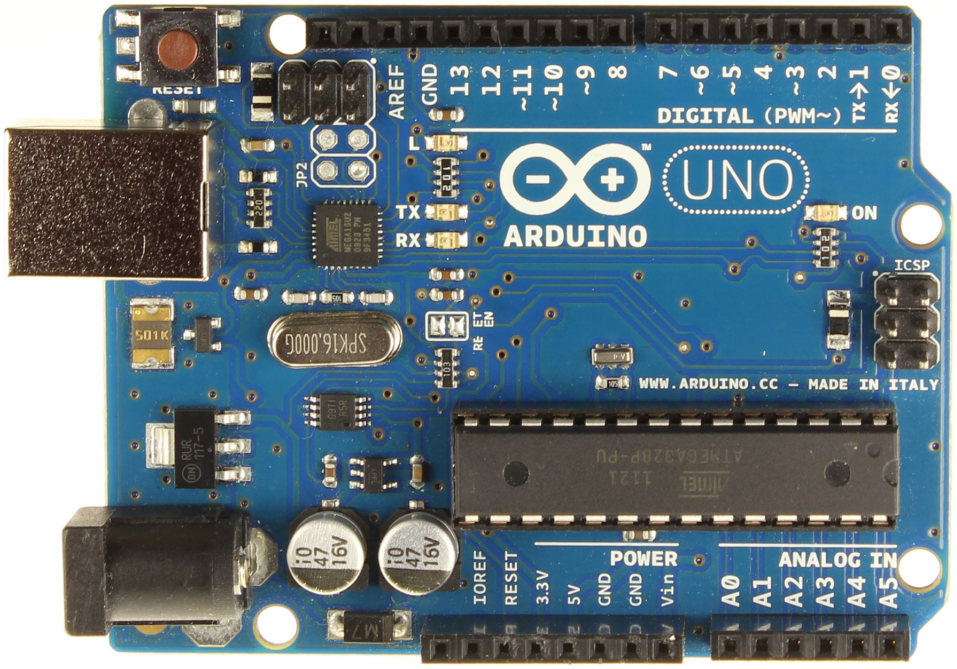


FIGURE 3.11: Arduino UNO

The Arduino UNO is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable (not included) or power it with a AC-to-DC adapter or battery to get started.

This is interfaced with RFID Reader module to detect a RFID Tag and obtain unique user information. This information is sent to the DSS.

Arduino is interfaced with LCD Display to show the direction to the user.

3.7 LCD Display

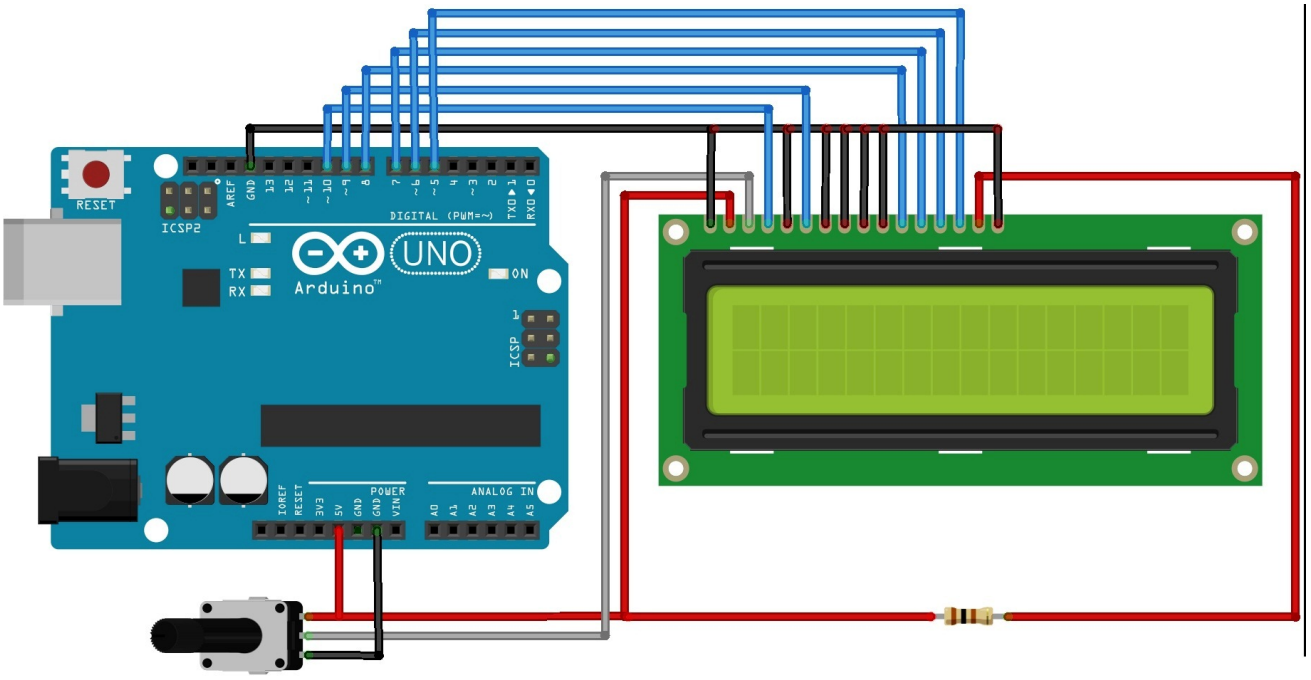


FIGURE 3.12: Interfacing LCD with Arduino

A liquid-crystal display (LCD) is a flat-panel display or other electronic visual display that uses the light-modulating properties of liquid crystals.

LCD can be connected to Arduino to display directions to the user when he enters the parking area. The DSS allocates the parking slot and uses shortest path to give directions to the user.

Interfacing LCD with Arduino is as shown in the following figure

3.8 Decision Support System

The Decision Support System performs four main functions. The data received from the RFID Reader Module is used to uniquely identify the user and updates the user information- RFID tag number, Username, Car number, Email-id in the

database.Secondly, the IR sensor data obtained wirelessly via ESP8266 module is used to update the status array and in the database as either the slots are free or occupied.Thirdly,the user is guided to the closest vacant slot using shortest path algorithm and the corresponding directions are communicated to the user through E-mail and also displayed on the LCD.DSS also calculates and detects the charges for every user during the time of exit.

3.8.1 Check for Registered Users

Step 1: Receives RFID Tag number from the RFID Reader Module to uniquely identify the user.
Step 2: The RFID Tag number is checked against the registered users in the database and identifies the user uniquely or identifies as unregistered user and asks him to register.

3.8.2 Optimal slot allocation

Step 1: Receives the IR sensors data via the ESP8266 module in a web browser.
Step 2: Parses the browser content using jsoup. DSS updates the status array and in the database as either zero if the slot is free or as one if the slot is occupied.
Step 3: The user is then guided to the closest vacant slot using shortest path algorithm.
Step 4: The distance from the entrance to each slot is stored in an array.
Step 5: Data from ESP8266 module is obtained and the status of the slots are updated in the status array.

Step 6: Min[distance] to each vacant slot is found and the corresponding slot id- i th row and j th column is allocated.
Step 7: Directions are calculates as follows:
Step 7.1: For i th row, move i steps forward
Step 7.2: For j th column,
Step 7.2.1: if $j <= 5$ turn left and park the vehicle in j th slot
Step 7.2.2: else if $j >= 6$ turn right and park the vehicle in j th slot
Step 8: The directions are then communicated to the registered users through E-mail id and also displayed on the LCD.

3.8.3 Calculating the payment charges

Step 1: Using the RFID Tag number, DSS calculates the charges as (Checkout time - Checkin time)*Cost for each minute + Base amount, where the base amount is a predefined amount for parking .
Step 2: Updates the slot information as free in the database when the user exits.

3.8.4 Functions of the DSS

The DSS is a functional unit of the system which takes the responsibility of allocating the closest vacant slot to the user using shortest path algorithm.

- Finding the vacant slots available at any instant using ESP8266 WiFi module.
- Allocating the closest vacant slot available to the user.

- Directing the user to the allocated slot by giving appropriate directions using LCD Display.
- Obtaining the unique user information using RFID Tag.
- Calculating the charges and intimating the user.

3.9 Pseudo code for the system

3.9.1 Read RFID Tags

- Step 1: User places the RFID tag on the reader module.
- Step 2: RFID Reader recognizes the tag.
- Step 3: User information is passed to DSS.

3.9.2 Data Acquisition

- Step 1: IR Sensors detects the presence of the vehicle and determines the status of the slot.
- Step 2: ESP8266 accepts the sensor data from each sensor.
- Step 3: ESP8266 sends sensor information wirelessly via the embedded WiFi module to the Java Application in the DSS module.

3.9.3 Process data

- Step 1: ESP8266 sensor information is obtained in a web browser through WiFi.
- Step 2: The HTML data of the browser is parsed using jsoup.
- Step 3: The parsed information is used to update the status information of the slots in database.
- Step 4: Read the status information of the slots from the database.
- Step 5: Find and allocate the closest vacant slot to the user based on vacancy of the slots.
- Step 6: Communicate the directions and allocated vacant slot id to the user via e-mail.

3.9.4 Display directions at LCD

- Step 1: Read the instructions given by DSS.
- Step 2: Clear LCD and display the directions given by DSS.

3.9.5 Send mail to user

- Step 1: Retrieve the user email id from database.
- Step 2: Send check-in, slot id, direction, charges information as email to the user.

3.9.6 User Exits

- Step 1: User decides to exit and moves from the parking slot.
- Step 2: User again places the RFID tag on reader module which sends the TagId to the DSS.
- Step 3: DSS recognizes the user and deducts the charges from his balance.

CHAPTER 4

EXPERIMENTAL RESULT AND ANALYSIS

4.1 Implementation Scenario

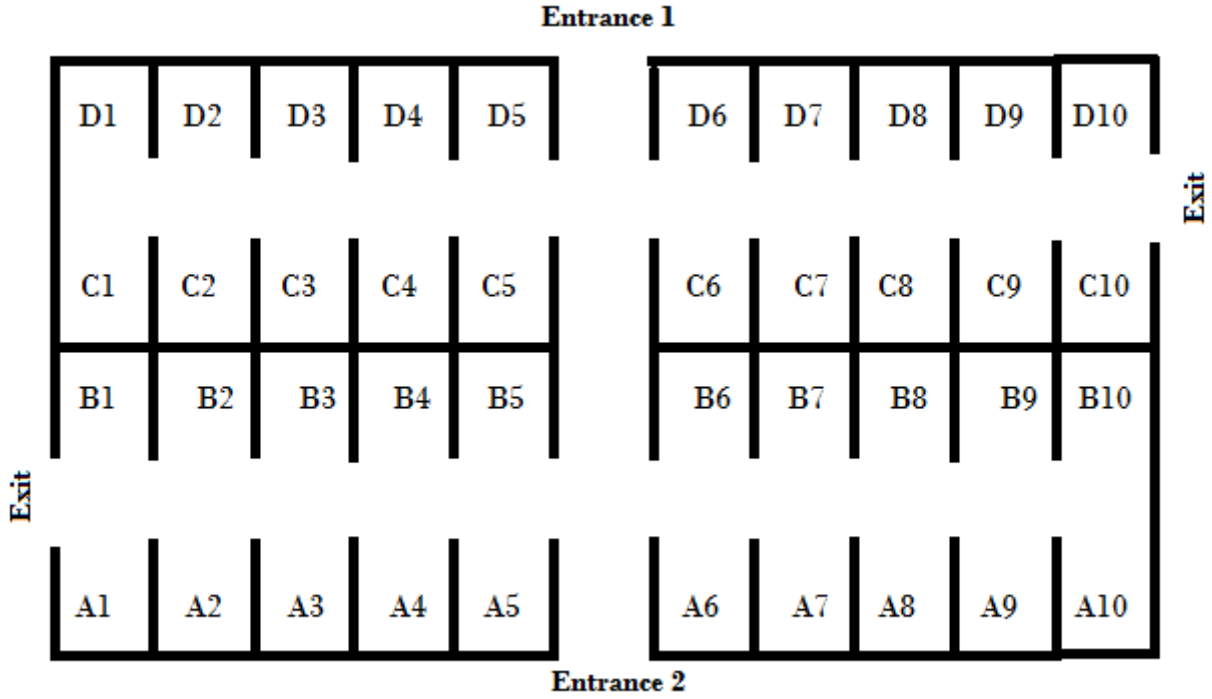


FIGURE 4.1: The layout of the parking slots

The layout considered has 40 slots 4 rows and 10 columns(4X10) with two entrances.There are two separate exits. Each road to the parking system has a lane wide enough to accomodate two vehicles at a time. The closest vacant slot is allocated to the user as he enters,based on the sensor data obtained from the ESP8266 module.The user is then guided to the allocated parking slot by displaying the directions on the LCD display and by sending an E-mail to the

user. While leaving from the parking system, the user checks out using RFID with which the parking charges are deducted and car goes out through the exit.

4.2 Hardware Requirements

- ESP8266 E12 wifi module
- Arduino UNO
- 10 IR Sensor module
- 4 1.5V AA Batteries
- 2 9V Batteries
- 20 X 4 LCD Display
- RFID Reader Module
- 10 RFID Tags(cards)
- Laptop/PC with wireless connectivity, installed with JAVA JDK and JRE
- Breadboard and connecting wires
- Other items like Battery holder,cars,velcro etc

4.3 Software Requirements

- Eclipse IDE

- Arduino IDE
- NodeMCU Flasher
- ESPlorer IDE
- JSoup jar file

4.4 Test Cases

The following are the test cases for the implementation scenario.

- When all the parking slots are free, the nearest slot from the entrance is allocated first which is identified by the shortest path algorithm.
- When one or more slots are occupied, then based on the status of the slots the shortest path algorithm dynamically identifies the nearest slot with respect to the entrance.
- When there is only one vacant slot and there is one vehicle at each entrance, then based on the priorities of the entrances ,the slot is allocated to the vehicle.
- When the SPS is fully occupied and there is no vacant slot, then message will be displayed on the LCD stating that the parking area is fully occupied.

The user enters the parking system.The status of the slots are determined by the IR sensor modules as shown in the Fig 4.2 which is sent to ESP8266.

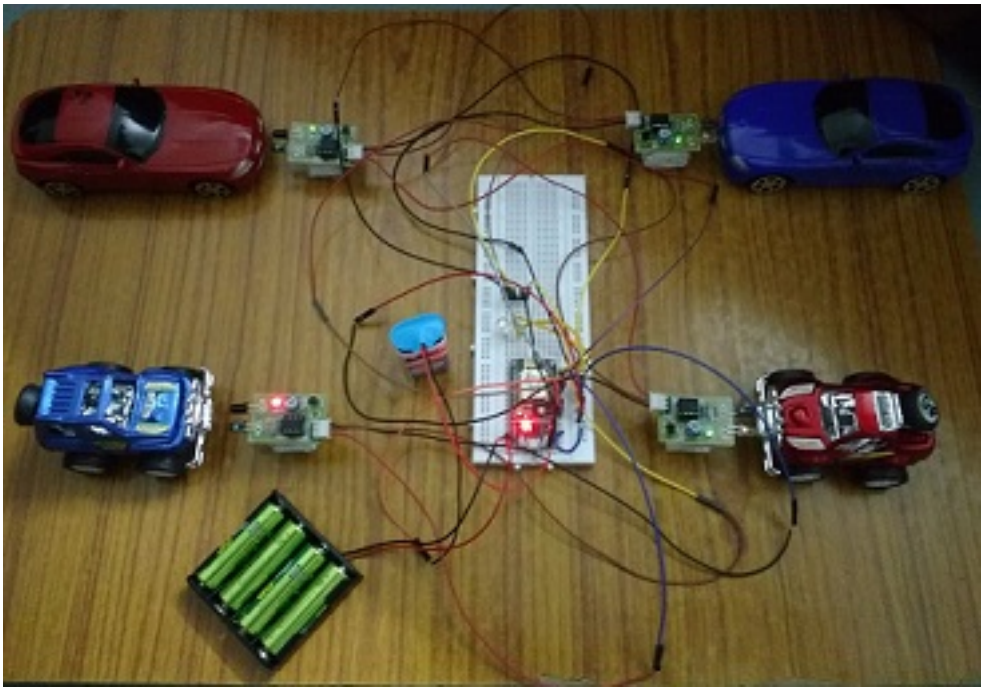


FIGURE 4.2: Status of the parking slots

The ESP8266 Module accepts the sensor information and sends it wirelessly to a web browser.The sensor information transmitted by ESP8266 is shown in Fig 4.3.The sensor information is represented as 1 and 0. 1 represents that the slot is occupied and a 0 represents that the slot is free.

FIGURE 4.3: Status of parking slots obtained in web browser

The snapshot of the database showing the status of the slots in the parking area is shown in Fig.4.4.

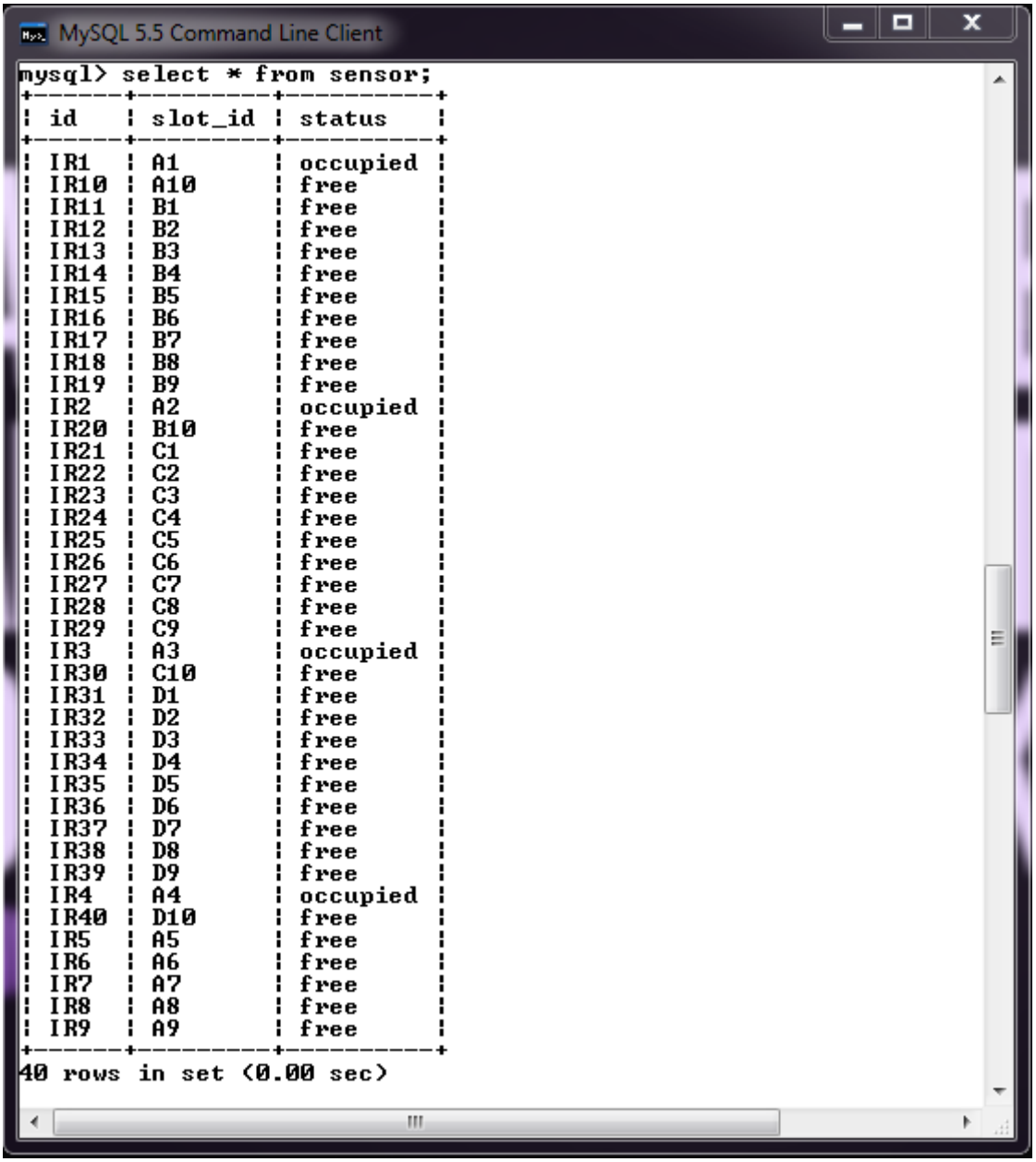
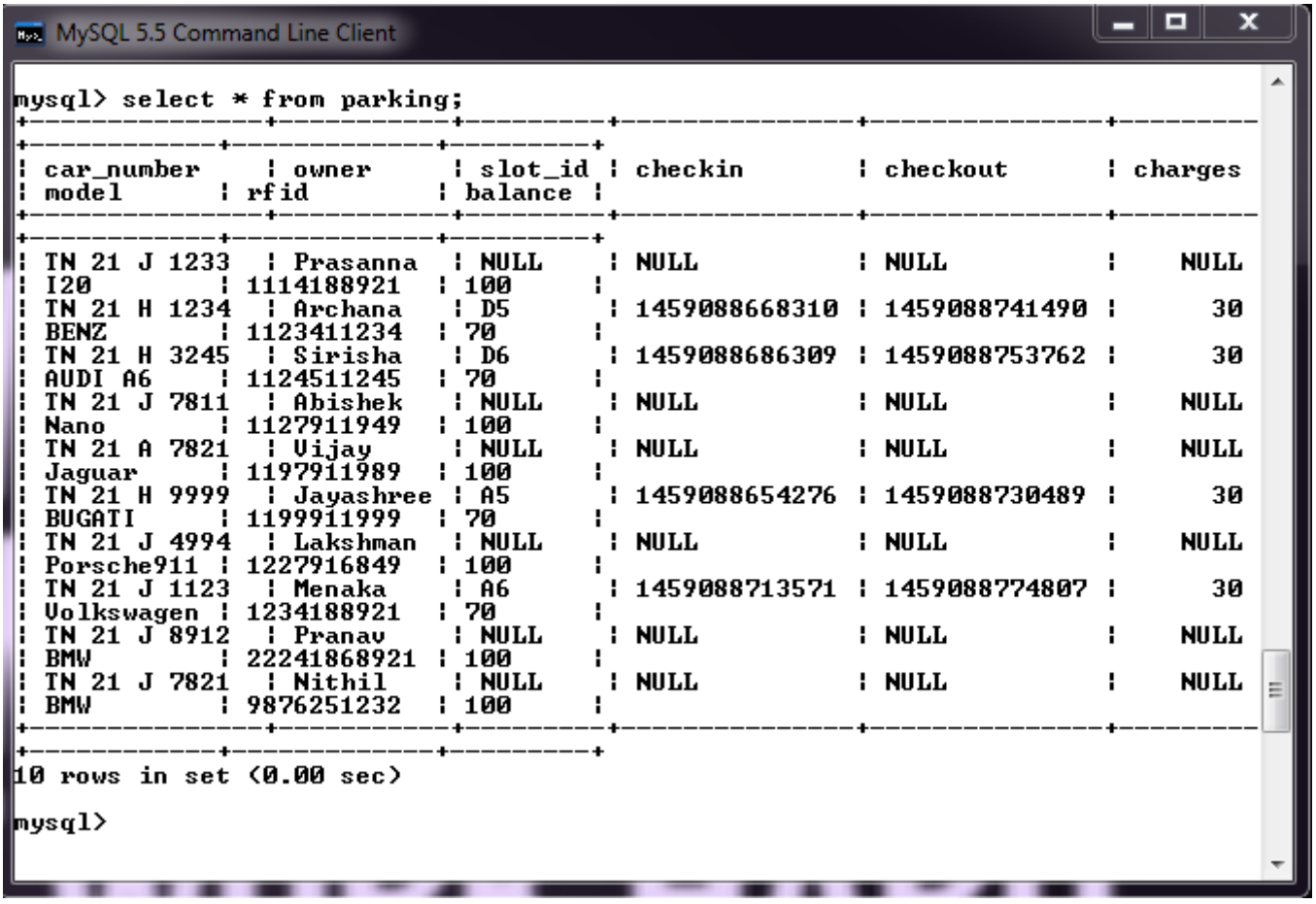


FIGURE 4.4: Status of the slots updated in database

The snapshot of the registered users of the parking area is shown in Fig 4.5. The unique user information is stored in this database and retrieved whenever needed.



car_number	owner	slot_id	checkin	checkout	charges
model	rfid	balance			
TN 21 J 1233	Prasanna	NULL	NULL	NULL	NULL
I20	1114188921	100			
TN 21 H 1234	Archana	D5	1459088668310	1459088741490	30
BENZ	1123411234	70			
TN 21 H 3245	Sirisha	D6	1459088686309	1459088753762	30
AUDI A6	1124511245	70			
TN 21 J 7811	Abishek	NULL	NULL	NULL	NULL
Nano	1127911949	100			
TN 21 A 7821	Vijay	NULL	NULL	NULL	NULL
Jaguar	1197911989	100			
TN 21 H 9999	Jayashree	A5	1459088654276	1459088730489	30
BUGATI	1199911999	70			
TN 21 J 4994	Lakshman	NULL	NULL	NULL	NULL
Porsche911	1227916849	100			
TN 21 J 1123	Menaka	A6	1459088713571	1459088774807	30
Volkswagen	1234188921	70			
TN 21 J 8912	Pranav	NULL	NULL	NULL	NULL
BMW	22241868921	100			
TN 21 J 7821	Nithil	NULL	NULL	NULL	NULL
BMW	9876251232	100			

FIGURE 4.5: Users registered with the parking area

The layout of the java application with check-in, check-out, and update status options is shown in Fig.4.6.

The RFID information and the entrance in which the user has entered during check-in are shown in Fig 4.7. The directions are displayed as shown.

The scenario wherein which the user exits the parking area is shown in Fig 4.8. The charges are calculated based on the check-in and check-out time and intimated to the user and deducted from the card.

The LCD integration with Arduino UNO which is used to display the directions

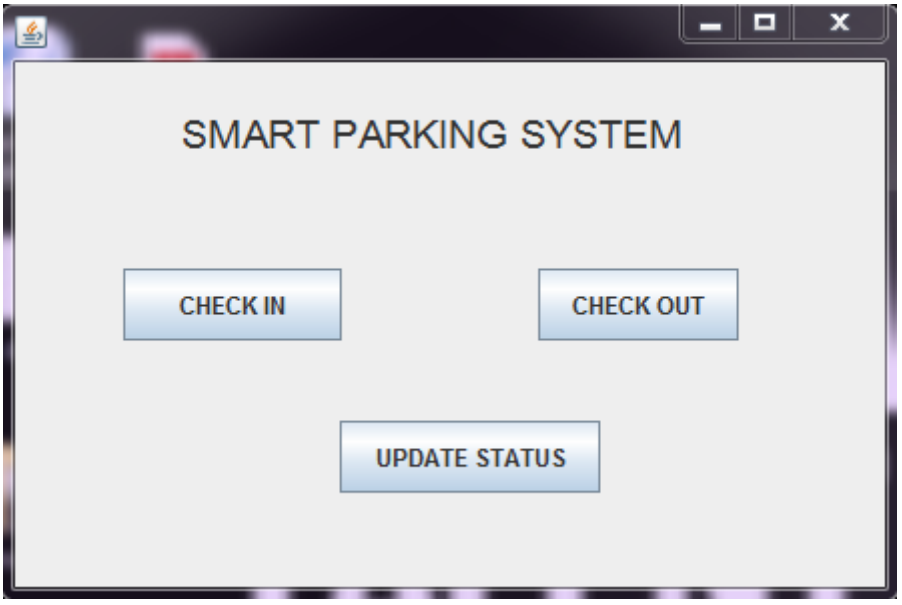


FIGURE 4.6: Smart parking system Java Application

and guide the user to the allocated parking slot is shown in Fig 4.9.

The information intimated by the system to the user regarding the directions to the allocated slot, charges for parking and available balance in the card are sent using e-mail which is shown in Fig 4.10.

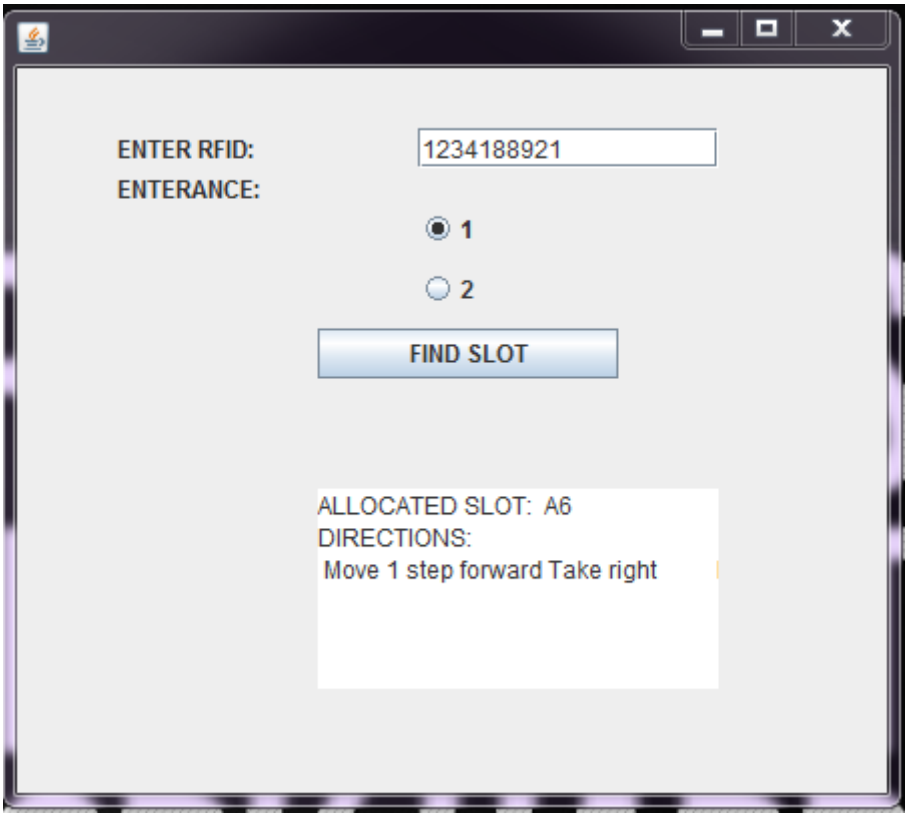


FIGURE 4.7: User enters the parking slot through entrance 1 and a slot is allocated, directions are displayed

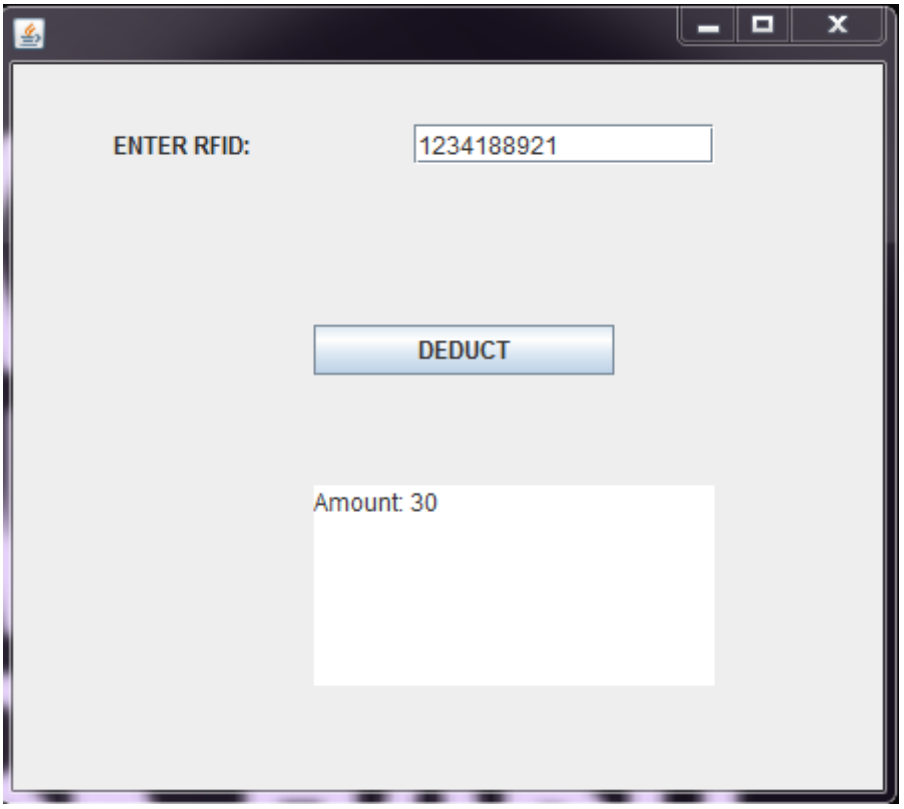


FIGURE 4.8: User exits the parking area and charges are deducted from his account

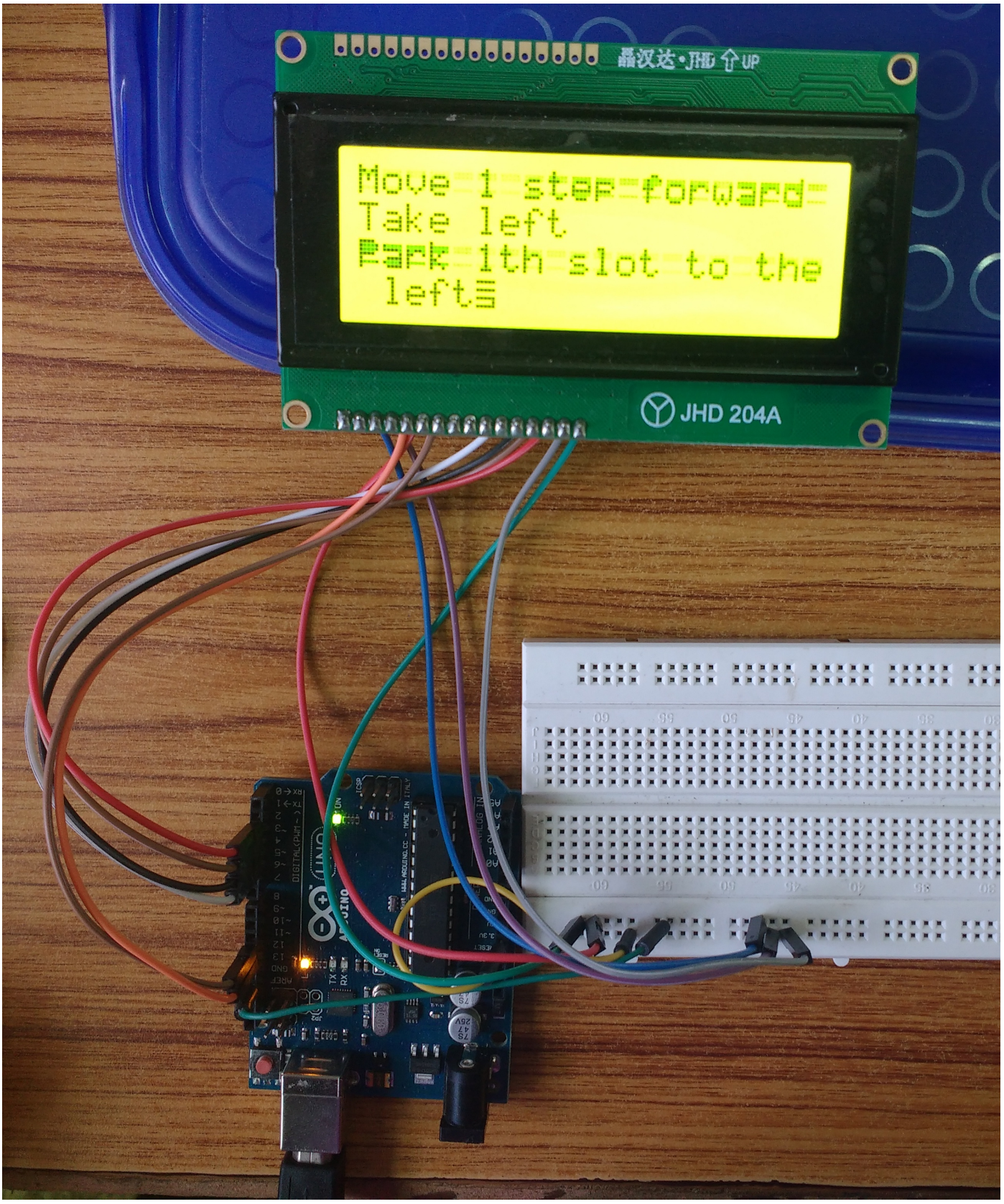


FIGURE 4.9: Directions are displayed through the LCD integrated with arduino UNO

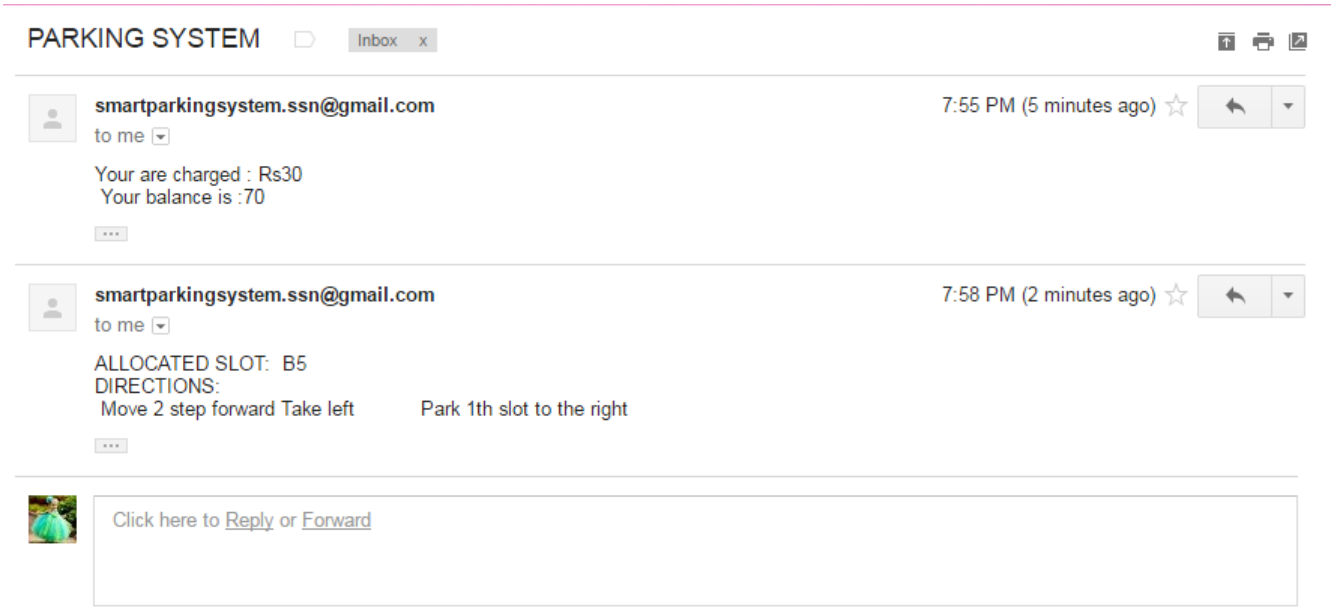


FIGURE 4.10: Directions and the charges are intimated to the user by e-mail

CHAPTER 5

CONCLUSION AND FUTURE WORK

To conclude, we have discussed about our system which was built using cheap components.The system is capable of transferring the sensor data from the multiple sensors situated at the parking slots wirelessly, which is received by a java application and then sent to the decision support system for processing.Algorithm is applied by the decision support system to efficiently allocate a vacant slot to the vehicle.Directions and available vacant slots are displayed in the LCD mounted at the entrance.

By using a smart parking system, human errors can be eliminated completely since it uses a Wifi enabled sensor based technology for dynamic allocation. The allocation uses a smart approach of finding the closest slot from the users entrance. The directions when given in clear manner to the user will also eliminate the chances of confusion.

This particular project takes into account of the Indian scenario and is designed with many constraints such as limited technologies available, less fuel consumption, utilizing the entire parking slots,time required in finding the vacant slot. It provides more cost effective, feasible, safer and reliable solution compared to the existing system.By installing SPS the average revenue increases and the economical payback period is less which will increase the overall profit of the system. This will result in effective installation maintenance and monitoring of the system.

It is important to keep in pace with the technology that is used in this type of system,using SmartEye Sensors for accurately sensing the status of the slots for various sizes of parking slots.This idea can be extended to ramped, multi-storey, on-street and off-street parking. The efficient allocation and usage of the available area is the key to SPS.The data can be stored in the cloud for efficiently processing and storing the processed data.Mobile apps can be used to guide and provide various services to the user. We can embed Google Maps into mobile application so that it helps the user to see the directions through Mapsto reach the correct parking slot. GPS(Global Position System) help the user to track the route from current position to allocated slot.

Appendix A

ESP8266

ESP8266EX offers a complete and self-contained WiFi networking solution; it can be used to host the application or to offload WiFi networking functions from another application processor.When ESP8266EX hosts the application, it boots up directly from an external flash. In has integrated cache to improve the performance of the system in such applications.Alternately, serving as a WiFi adapter, wireless internet access can be added to any micro controller based design with simple connectivity (SPI/SDIO or I2C/UART interface).

A.1 Internal SRAM and ROM

ESP8266EX WiFi SoC is embedded with memory controller, including SRAM and ROM. MCU can visit the memory units through iBus, dBus, and AHB interfaces. All memory units can be visited upon request, while a memory arbiter will decide the running sequence according to the time when these requests are received by the processor.

A.2 External SPI Flash

An external SPI flash is used together with ESP8266EX to store user programs. Theoretically speaking, up to 16 Mbyte memory capacity can be supported. Suggested SPI Flash memory capacity:

- OTA is disabled: the minimum flash memory that can be supported is 512 kByte;

- OTA is enabled: the minimum flash memory that can be supported is 1 Mbyte.Several SPI modes can be supported, including Standard SPI, Dual SPI, DIO SPI, QIO SPI, and Quad SPI.

A.3 Parameters and Pin Details

Categories	Items	Values
WiFi Paramters	Certificates	FCC/CE/TELEC/SRRC
	WiFi Protocles	802.11 b/g/n
	Frequency Range	2.4G-2.5G (2400M-2483.5M)
	Tx Power	802.11 b: +20 dBm
		802.11 g: +17 dBm
		802.11 n: +14 dBm
	Rx Sensitivity	802.11 b: -91 dbm (11 Mbps)
		802.11 g: -75 dbm (54 Mbps)
		802.11 n: -72 dbm (MCS7)
	Types of Antenna	PCB Trace, External, IPEX Connector, Ceramic Chip
Hardware Paramaters	Peripheral Bus	UART/SDIO/SPI/I2C/I2S/IR Remote Control
		GPIO/PWM
	Operating Voltage	3.0~3.6V
	Operating Current	Average value: 80mA
	Operating Temperature Range	-40°~125°
	Ambient Temperature Range	Normal temperature
	Package Size	5x5mm
	External Interface	N/A

A.4 Theory of Operation

Set of high performance, high integration wireless SOCs, designed for space and power constrained mobile platform designers. It provides unsurpassed ability to

Pin	Name	Type	Function
1	VDDA	P	Analog Power 3.0 ~3.6V
2	LNA	I/O	RF Antenna Interface. Chip Output Impedance=50Ω No matching required but we recommend that the n-type matching network is retained.
3	VDD3P3	P	Amplifier Power 3.0~3.6V
4	VDD3P3	P	Amplifier Power 3.0~3.6V
5	VDD_RTC	P	NC (1.1V)
6	TOUT	I	ADC Pin (note: an internal pin of the chip) can be used to check the power voltage of VDD3P3 (Pin 3 and Pin4) or the input voltage of TOUT (Pin 6). These two functions cannot be used simultaneously.
7	CHIP_EN	I	Chip Enable. High: On, chip works properly; Low: Off, small current
8	XPD_DCDC	I/O	Deep-Sleep Wakeup； GPIO16
9	MTMS	I/O	GPIO14; HSPI_CLK
10	MTDI	I/O	GPIO12; HSPI_MISO
11	VDDPST	P	Digital/IO Power Supply (1.8V~3.3V)
12	MTCK	I/O	GPIO13; HSPI_MOSI; UART0_CTS
13	MTDO	I/O	GPIO15; HSPI_CS; UART0_RTS
14	GPIO2	I/O	UART Tx during flash programming; GPIO2
15	GPIO0	I/O	GPIO0; SPI_CS2
16	GPIO4	I/O	GPIO4
17	VDDPST	P	Digital/IO Power Supply (1.8V~3.3V)
18	SDIO_DATA_2	I/O	Connect to SD_D2 (Series R: 200Ω); SPIHD; HSPIHD; GPIO9
19	SDIO_DATA_3	I/O	Connect to SD_D3 (Series R: 200Ω); SPIWP; HSPIWP; GPIO10
20	SDIO_CMD	I/O	Connect to SD_CMD (Series R: 200Ω); SPI_CS0; GPIO11
21	SDIO_CLK	I/O	Connect to SD_CLK (Series R: 200Ω); SPI_CLK; GPIO6
22	SDIO_DATA_0	I/O	Connect to SD_D0 (Series R: 200Ω); SPI_MSIO; GPIO7
23	SDIO_DATA_1	I/O	Connect to SD_D1 (Series R: 200Ω); SPI_MOSI; GPIO8
24	GPIO5	I/O	GPIO5

embed WiFi capabilities within other systems, or to function as a standalone application, with the lowest cost, and minimal space requirement.ESP8266 also integrates an enhanced version of Tensilicas L106 Diamond series 32-bit processor, with on-chip SRAM, besides the WiFi functionalities. ESP8266EX is often integrated with external sensors and other application specific devices through its GPIOs; sample codes for such applications are provided in the

25	U0RXD	I/O	UART Rx during flash programming; GPIO3
26	U0TXD	I/O	UART Tx during flash progamming; GPIO1; SPI_CS1
27	XTAL_OUT	I/O	Connect to crystal oscillator output, can be used to provide BT clock input
28	XTAL_IN	I/O	Connect to crystal oscillator input
29	VDDD	P	Analog Power 3.0V~3.6V
30	VDDA	P	Analog Power 3.0V~3.6V
31	RES12K	I	Serial connection with a 12 kΩ resistor and connect to the ground
32	EXT_RSTB	I	External reset signal (Low voltage level: Active)

software development kit (SDK).

Appendix B

RFID Reader Module

B.1 Introduction

RFID stands for Radio-Frequency Identification. The acronym refers to small electronic devices that consist of a small chip and an antenna. The chip typically is capable of carrying 2,000 bytes of data or less. RFID is a general term that is used to describe a system that transmits the identity (in the form of a unique serial number) of an object wirelessly, using radio waves. This is sometimes referred to as contact-less technology and a typical RFID system is made up of three components: tags, readers and the host computer system. An RFID tag is a tiny radio device that is also referred to as a transponder, smart tag, smart label or radio barcode. The tag comprises of a simple silicon microchip attached to a small flat aerial and mounted on a substrate. The whole device can then be encapsulated in different materials (such as plastic)dependent upon its intended usage. The finished tag can be attached to an object,typically an item, box or pallet and read remotely to ascertain its identity, position or state.

B.2 Basic Device Operation

The reader, sometimes called an interrogator or scanner, sends and receives RF data to and from the tag via antennas. A reader may have multiple antennas that are responsible for sending and receiving radio waves. The data acquired by the readers is then passed to a host computer, which may run specialist RFID software or middleware to filter the data and route it to the correct application, to be processed into useful information.

B.3 Limitations

- Problems with RFID Standards: RFID has been implemented in different ways by different manufacturers; global standards are still being worked on. It should be noted that some RFID devices are never meant to leave their network (as in the case of RFID tags used for inventory control within a company).

- RFID systems can be easily disrupted: Since RFID systems make use of the electromagnetic spectrum (like WiFi networks or cellphones), they are relatively easy to jam using energy at the right frequency. Although this would only be an inconvenience for consumers in stores (longer waits at the checkout), it could be disastrous in other environments where RFID is increasingly used, like hospitals or in the military in the field.

- RFID Reader Collision: Reader collision occurs when the signals from two or more readers overlap. The tag is unable to respond to simultaneous queries. Systems must be carefully set up to avoid this problem; many systems use an anti-collision protocol (also called a singulation protocol). Anti-collision protocols enable the tags to take turns in transmitting to a reader.

- RFID Tag Collision: Tag collision occurs when many tags are present in a small area; but since the read time is very fast, it is easier for vendors to develop systems that ensure that tags respond one at a time.

B.4 Merits

RFID combined with mobile computing and Web technologies provide an effective way for organizations to identify and manage their assets. Mobile

computers,with integrated RFID readers, can now deliver a complete set of tools that eliminate paperwork, give positive proof of identification and prove attendance. Errors are virtually unheard of as this approach eliminates manual data entry. Web based management tools allow organizations to monitor their assets and make management decisions from anywhere in the world.

- Improved Productivity and Cost Avoidance
- Decreased Cycle Time and Taking Costs Out
- Reduced Rework
- Reduced Business Risk Control of Assets
- Improved Security and Service
- Improved Utilisation of Resources
- Increased Revenues
- Exception Management

B.5 Different types of RFID

B.5.1 Active RFID Tags

Active RFID Tags are battery powered. They broadcast a signal to the reader and can transmit over the greatest distances (100+ meters). They can be used to track high value goods like vehicles and large containers of goods.Shipboard containers are a good example of an active RFID tag application.

B.5.2 Passive RFID Tags

Passive RFID Tags do not contain a battery. Instead, they draw their power from the radio wave transmitted by the reader. The reader transmits a low power radio signal through its antenna to the tag, which in turn receives it through its own antenna to power the integrated circuit (chip). The tag will briefly converse with the reader for verification and the exchange of data. As a result, passive tags can transmit information over shorter distances (typically 3 meters or less) than active tags. They have a smaller memory capacity and are considerably lower in cost making them and ideal for tracking lower cost items.

B.6 Features and benefits of RFID over barcodes

Technology	Barcode	RFID	RFID Benefit Example
Capacity			
Line of sight requirement	Required	Not required	No need to orientate scanned items
Number of items that can be scanned	One	Multiple	Very fast inventory count
Automation and Accuracy	Manual read errors and prone to mis-scanning	Fully automated and highly accurate	Error free inventory count
Identification	Only series or type	Unique item level	Targeted recall
Data Storage	Only a meaningless code	Up to several KB	Real time data access in any location

Appendix C

LUA PROGRAMMING LANGUAGE

Lua (loo-, from Portuguese, meaning moon) is a lightweight multi-paradigm programming language designed as a scripting language with extensible semantics as a primary goal. Lua is cross-platform since it is written in ANSI C and has a relatively simple C API. Lua combines simple procedural syntax with powerful data description constructs based on associative arrays and extensible semantics. Lua is dynamically typed, runs by interpreting bytecode for a register-based virtual machine, and has automatic memory management with incremental garbage collection, making it ideal for configuration, scripting, and rapid prototyping.

C.1 Features

Lua is commonly described as a multi-paradigm language, providing a small set of general features that can be extended to fit different problem types, rather than providing a more complex and rigid specification to match a single paradigm. Lua, for instance, does not contain explicit support for inheritance, but allows it to be implemented with metatables. Similarly, Lua allows programmers to implement namespaces, classes, and other related features using its single table implementation; first-class functions allow the employment of many techniques from functional programming; and full lexical scoping allows fine-grained information hiding to enforce the principle of least privilege.

In general, Lua strives to provide flexible meta-features that can be extended as needed, rather than supply a feature-set specific to one programming paradigm.As

a result, the base language is light the full reference interpreter is only about 180 kB compiled and easily adaptable to a broad range of applications.

Lua is a dynamically typed language intended for use as an extension or scripting language, and is compact enough to fit on a variety of host platforms. It supports only a small number of atomic data structures such as boolean values, numbers (double-precision floating point by default), and strings. Typical data structures such as arrays, sets, lists, and records can be represented using Luas single native data structure, the table, which is essentially a heterogeneous associative array.

Lua implements a small set of advanced features such as first-class functions,garbage collection, closures, proper tail calls, coercion (automatic conversion between string and number values at run time), coroutines (cooperative multitasking) and dynamic module loading.

By including only a minimum set of data types, Lua attempts to strike a balance between power and size.

C.2 Example Code

The classic hello world program can be written as follows

```
print(Hello World !)
```

REFERENCES

1. Amin Kianpisheh, Norlia Mustaffa, Pakapan Limtrairut and Pantea Keikhosrokiani, Smart Parking System (SPS) Architecture Using Ultrasonic Detector, International Journal of Software Engineering and Its Applications

2. Barton, J., J. Buckley, B. OFlynn, S.C. OMathuna, J.P. Benson, T. ODonovan, U. Roedig, and C. Sreenan, The D-Systems ProjectWireless Sensor Networks for Car- Park Management, Proceedings of the 65th IEEE Vehicular Technology Conference, Dublin, Ireland, Apr 2007,pp. 170-173.

3. Hongwei Wang and Wenbo He, A Reservation-based Smart Parking System, The First International Workshop on Cyber-Physical Networking Systems

4. IrisNet: Internet-scale Resource-Intensive Sensor Network Service, http://www.intel-iris.net

5. Mala Aggarwal, Simmi Aggarwal, R.S.Uppal ,Comparative Implementation of Automatic Car Parking System with least distance parking space in Wireless Sensor Network, International Journal of Scientific and Research Publications

6. Ramneet Kaur and Valwinder Singh ,2013, Design And Implementation of car parking system on FPGA (Finite State Machine: Parking System) ,International journal of VLSI design and communication systems.

7. Sangwon Lee, Dukhee Yoon and Amitabha Ghosh, Intelligent Parking Lot Application Using Wireless Sensor Networks, Proceedings of IEEE conference, 978-1-4244-2249-4/08

8. Vanessa W.S. Tang, Yuan Zheng, Jiannong Cao,2006,An Intelligent Car park Management System based onWireless Sensor networks, International

symposium on pervasive computing and applications, Hong Kong polytechnic University, China.

9. Vijay Kumar.P ,SIDDARTH T.S, A Prototype Parking System using Wireless Sensor Networks, International Journal of Computer Communication and Information System