SMART CAR PARKING SYSTEM

PROJECT REPORT

Submitted by

AISHWARYA.S	**Reg.No:13BEC005**
ARAFATH.P	**Reg.No:13BEC017**
DIVYA.R	**Reg.No:13BEC039**
HAMSIKA.S	**Reg.No:13BEC051**

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION

ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE – 641 049

(An Autonomous Institution Affiliated to Anna University, Chennai)

APRIL 2017

BONAFIDE CERTIFICATE

Certified that this project report titled **“SMART CAR PARKING SYSTEM”** is the bonafied work of “**AISHWARYA.S,ARAFATH.P,DIVYA.R,HAMSIKA.S”** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Mr.Timothy Dhayakar Paul M.E.,(Ph.D),	**Dr.K.Malarvizhi M.E.,Ph.D.,**
Assistant Professor	**HEAD OF THE DEPARTMENT**
Electronics & Communication Engineering	Electronics & Communication Engineering
Kumaraguru College of Technology	Kumaraguru College of Technology
Coimbatore.	Coimbatore.

The candidates with Register numbers 13BEC005,13BEC017,13BEC039 and 13BEC051 are examined by us in the project viva-voce examination held on …………………….

EXTERNAL EXAMINER **INTERNAL EXAMINER**

ACKNOWLEDGEMENT

First we would like to express our praise and gratitude to the Lord, who has showered his grace and blessing enabling us to complete this project in an excellent manner. He has made all things in beautiful in his time.

We express our sincere thanks to our beloved Joint Correspondent, **Shri. Shankar Vanavarayar** for his kind support and for providing necessary facilities to carry out the project work.

We would like to express our sincere thanks to our beloved Principal **Dr.R.S.Kumar M.E., Ph.D.,** who encouraged us with his valuable thoughts.

We would like to express our sincere thanks and deep sense of gratitude to our HOD,**Dr.K.Malarvizhi M.E.,Ph.D.,** for her valuable suggestions and encouragement which paved way for the successful completion of the project.

We are greatly privileged to express our deep sense of gratitude to the Project Coordinator **Prof Mr S.Govindaraju M.E**, for her continuous support throughout the course.

In particular, We wish to thank and express our everlasting gratitude to the Supervisor **Mr.Timothy Dhayakar Paul M.E., (Ph.D)**, Assistant Professor for her expert iiiounseling in each and every steps of project work and we wish to convey our deep sense of gratitude to all teaching and non-teaching staff members of ECE Department for their help and cooperation.

Finally, we thank our parents and our family members for giving us the moral support in all of our activities and our dear friends who helped us to endure our difficult times with their unfailing support and warm wishes.

ABSTRACT

Among the challenges that we face in our day to day life one of most unavoidable challenge is parking the car wherever we go. As our need increases our travelling increases but due to drastic increase in usage of vehicles and increase in population we face the tough task of parking our car particularly during busiest hours of the day. During peak hours most of the reserved parking area gets full and this leaves the user to search for their parking among other parking area which creates more traffic. To overcome this problem there is definitely a need for designed parking in commercial environment. To design such parking slot we need to take into the account of reservation of parking slot with optimal parking space which depends on cost and time. Cost function should also combine with the parking cost and proximity to the destination. However here we have designed the time driven sequence method which solves the problem of parking using slot allocation method. This project proposes an system, which is used to implement a prototype of Smart Parking System based on Reservation (SPSR) that allows drivers to effectively find and reserve the vacant parking spaces.

TABLE OF CONTENTS

LIST OF ABBREVATIONS

SCU	Signal Controlling Unit
WIFI	Wireless Fidelity
IR	Infra Red
LCD	Liquid Crystal Display
PC	Personal Computer
LED	Light Emitting Diode
SSH	Secure Shell

LIST OF FIGURES

1.INTRODUCTION:

As the population increased in the metropolitan cities, the usage of vehicles got increased. It causes problem for parking which leads to traffic congestion, driver frustration, and air pollution.When we visit the various public places like Shopping malls, multiplex cinema hall & hotels during the festival time or weekends it creates more parking problem. In the recent research found that a driver takes nearly 8 minutes to park his vehicle because he spend more time in searching the parking lot. This searching leads to 30 to 40% of traffic congestion. In urban areas, one-third of cars which have reached their destination and are circling around looking for a parking space thus leading to problems like pollution and traffic congestion. In a recent survey, researchers have found that in one year, car cruising for parking created the equivalent of 38 times trips around the world, burning 1.7 lakh liter of fuel and producing 730 tons of CO2. So, it is essential to control the air pollution using a robust parking system that will be used for the reservation of parking spot as well as parking spot allocation in on spot resource allocation scenario with the help of android. So if we have parking space information, we can plan for advance booking based on requirement.

2.HARDWARE DESCRIPTION:

2.1 BLOCK DIAGRAM OF SMART CAR PARKING SYSTEM:

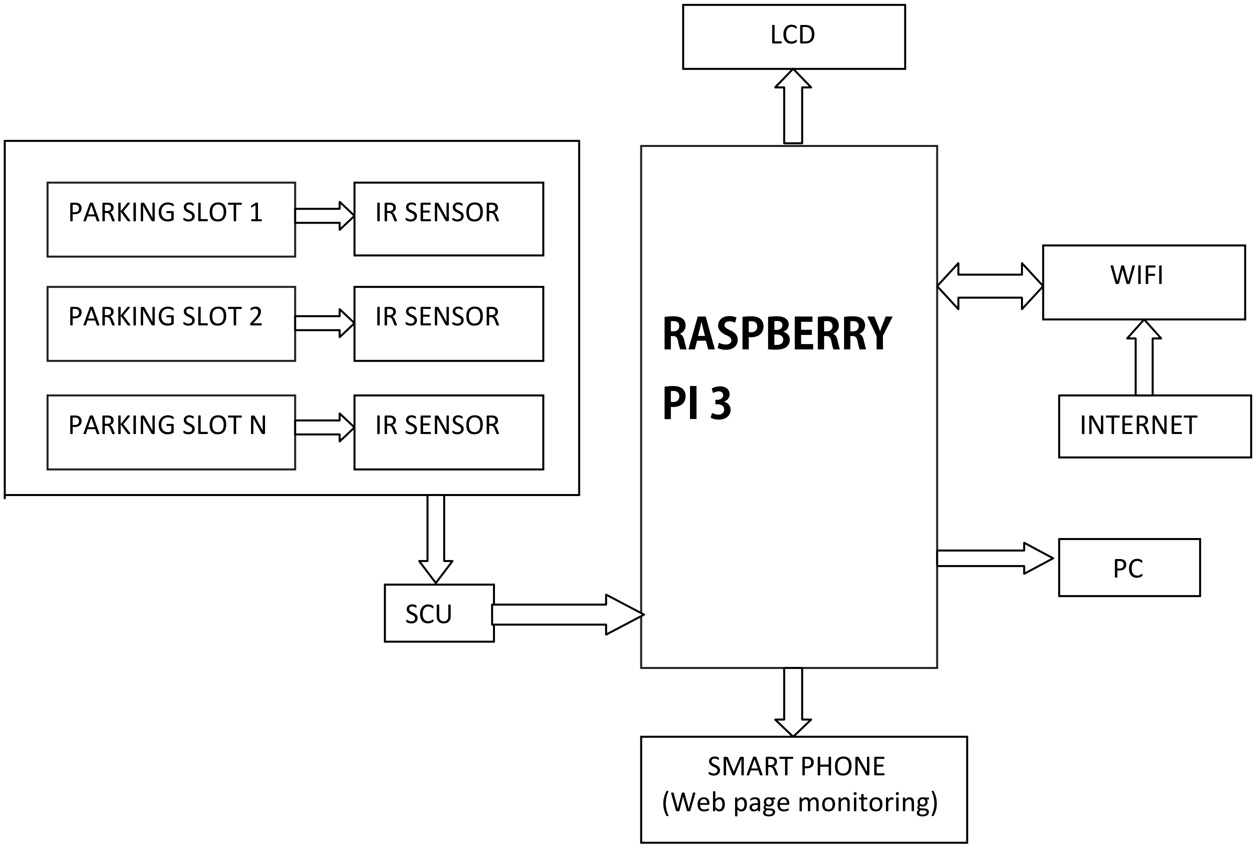


Fig 2.1: Block diagram of smart car parking system.

The block diagram of smart car parking system consists of hardware components such as microcontroller Raspberry pi 3 module, IR sensors, SCU, LCD, WIFI router,PC, smart phone. Each parking slot has IR sensors which is fixed at the middle of the each slots. The sensing range of IR sensors can be varied using the potentiometer in the IR sensors. These are connected to the raspberry pi through SCU. SCU is the signal conditioning unit accepts input signals from the analog sensors and gives a conditioned output of 0-5V DC corresponding to the entire range of each parameter. This unit also accepts the digital sensor inputs and gives outputs in 10 bit binary with a positive logic level of +5V. These outputs are processed in the raspberry pi 3 and the output is displayed in the LCD. The LCD displays 1 if the car is parked in the slot and displays 0 if the car is not parked in the slot. Our android mobiles will be connected to the raspberry pi via WiFi router. PC is connected with raspberry pi through wifi hotspot. A webpage created will be displayed both in the smart phone and PC if it is connected to that wifi hotspot. Pre-locking of slots can be done through our mobile phones by connecting to the same wifi hotspot of raspberry pi. The green box in the webpage indicates free slot,red box indicates that the car is parked in the slot and the yellow box indicates that the slot is pre locked. The pre-locking will stay for 10 seconds, after 10 seconds it will change to unlocked position which means there is a free slot. But in real time applications the delay time can be increased upto 20 minutes.

2.2 RASPBERRY PI:

Raspberry Pi is the credit card seized computer that plugs into your TV and a keyboard,which can be used for many of the things that your average desktop does –spreadsheets,word-processing,games and it also plays high definition video. Pi was introduced as an educational gadget to be used for protyping by hobbyists and for those who want to learn more about programming. It certainly cannot be a substitute for our day to day Linux, Mac or Windows PC.

Pi is based on a Broadcom SoC (System of Chip) with an ARM processor [~700 MHz], a GPU and 256 to 512 MB RAM. The boot media is an SD card [which is not included], and the SD card can also be used for persist data. Now that you know that the RAM and processing power are not nearly close to the power house machines you might have at home, these Pi's can be used as a Cheap computer for some basic functions, especially for experiments and education. The system is using Raspbian OS based on Debian OS running on raspberry pi 2B model which has a configuration of ARM cortex-A7 quad core , 900MHz CPU . It also has 4 USB ports, 40 GPIO pins, Full HDMI port, Ethernet port, micro SD card slot. System has a 16GB SD card of class 10 which is used to mount Raspbian OS. System is designed in such a way that it will use Raspberry Pi as a web server. Raspberry Pi is mounted with a custom designed shield that is used to connect 4 IR sensors using FRC cable. Hardware system consists of 4 IR sensors which are connected to raspberry pi through FRC cables. The system uses 24 GPIO pins to interact with IR sensors and tricolor LEDs. They are connected over a common VCC and GND.

The Raspberry Pi 3 is the third generation Raspberry Pi. Compared to the Raspberry Pi 2 it has:

- A 1.2GHz 64-bit quad-core ARMv8 CPU
- 802.11n Wireless LAN
- Bluetooth 4.1. 1GB RAM
- 4 USB ports,40 GPIO pins,full HDMI port & Ethernet port
- Combined 3.5mm audio jack and composite video
- Camera interface & Display interface
- Micro SD card slot .

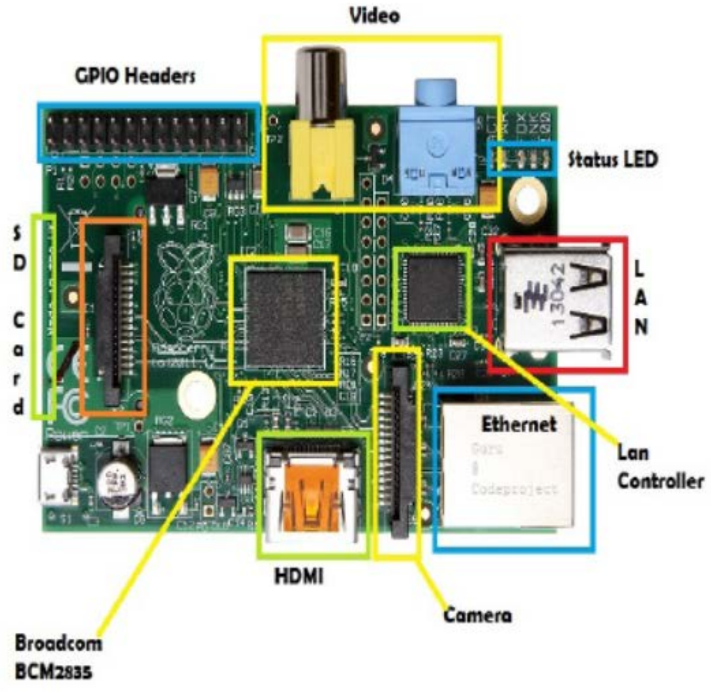
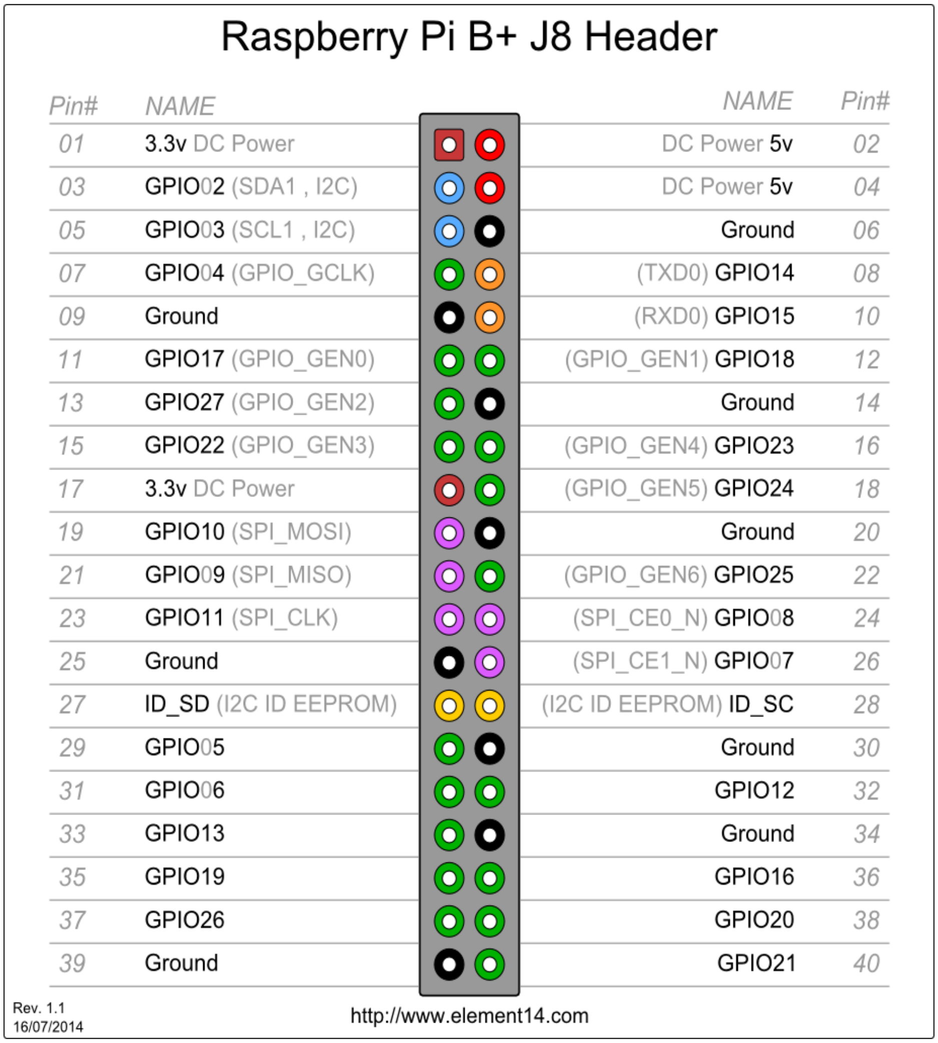


Fig 2.2:Raspberry pi 3 module

2.2.1 PIN DIAGRAM:



Raspberry Pi B+ J8 Header

Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1 , I2C)	DC Power 5v	04
05	GPIO03 (SCL1 , I2C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I2C ID EEPROM)	(I2C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40

Rev. 1.1
16/07/2014

http://www.element14.com

Fig 2.3: Pin diagram of Raspberry pi 3

2.3 IR SENSOR:

Sensors are the devices which senses from the physical world and stimuli on the environment. The system uses IR sensors to detect the vehicle above it. Sensors respond to system to the availability of the vehicle above the IR sensors. If the vehicle is present above the IR sensor, it will return value FALSE and the voltage goes LOW, If the vehicle is not present above the IR sensor, it will return value TRUE and the voltage remains HIGH.An infrared sensor is an electronic device which is used to sense light wavelength of its surroundings by either emitting or detecting infrared spectrum. It is also capable of measuring the heat being emitted by an object and detecting motion. LED will glow with respect to the IR sensor detection. Every IR sensor has a transmitter and a receiver. For every transmitter and receiver, there are total two registers and one more register is present to maintain voltage of LED that will show the status of parking slot to customers.

The parking space is reserved but the vehicle is yet to arrive, the LED glows RED. In another case vehicle is at the entrance of parking lot then LED glows BLUE, to notify the driver the parking slot allocated to him. When a vehicle has already occupied a parking slot the led will glow GREEN. The GREEN LED is used as a positive indicator to showcase proper functioning of IR Sensors. The detecting range of a photoelectric sensor is its "field of view", or the maximum distance the sensor can retrieve information from, minus the minimum distance. A minimum detectable object is the smallest object the sensor can detect. More accurate sensors can often have mimimum detectable objects of minuscule size.

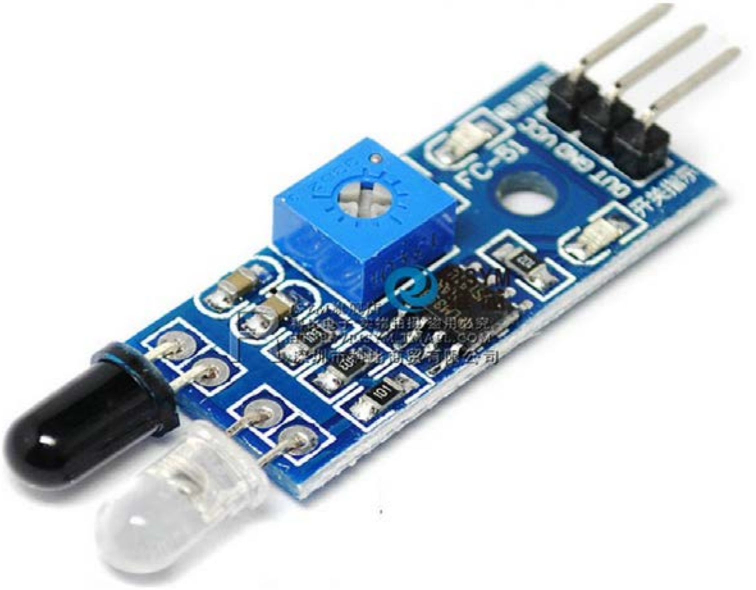


Fig 2.4:**IR sensor**

2.4 SIGNAL CONDITIONING UNIT:

The signal conditioning unit accepts input signals from the analog sensors and gives a conditioned output of 0-5V DC corresponding to the entire range of each parameter. This unit also accepts the digital sensor inputs and gives outputs in 10 bit binary with a positive logic level of +5V. The calibration voltages* (0, 2.5 and 5V) and the health bits are also generated in this unit.

Microcontrollers are widely used for control in power electronics. They provide real time control by processing analog signals obtained from the system. A suitable isolation interface needs to be designed for interaction between the control circuit and high voltage hardware. A signal conditioning unit which provides necessary interface between a high power grid inverter and a low voltage controller unit.

2.5 LIQUID CRYSTAL DISPLAY:

A liquid crystal display (LCD) is a thin, flat electronic visual display that uses the light modulating properties of liquid crystals (LCs). LCs do not emit light directly.They are used in a wide range of applications including: computer monitors, television, instrument panels, aircraft cockpit displays, signage, etc. They are common in consumer devices such as video players, gaming devices, clocks, watches, calculators, and telephones. LCDs have displaced cathode ray tube(CRT) displays in most applications. They are usually more compact, lightweight, portable, less expensive, more reliable, and easier on the eyes. They are available in a wider range of screen sizes than CRT and plasma displays, and since they do not use phosphors, they cannot suffer image burn-in.LCDs are more energy efficient and offer safer disposal than CRTs. Its low electrical power consumption enables it to be used in battery-powered electronic equipment. It is an electronically-modulated optical device made up of any number of pixels filled with liquid crystals and arrayed in front of a light source (backlight) or reflector to produce images in colour or monochrome. The earliest discovery leading to the development of LCD technology, the discovery of liquid crystals, dates from 1888. By 2008, worldwide sales of televisions with LCD screens had surpassed the sale of CRT units

Liquid crystal displays (LCDs) have materials which combine the properties of both liquids and crystals. Rather than having a melting point, they have a temperature range within which the molecules are almost as mobile as they would be in a liquid, but are grouped together in an ordered form similar to a crystal.An LCD consists of two glass panels, with the liquid crystal material sand witched in between them. The inner surface of the glass plates are coated with transparent electrodes which define the character, symbols or patterns to be displayed polymeric layers are present in between the electrodes and the liquid crystal, which makes the liquid crystal molecules to maintain a defined orientation angle.One each polarisers are pasted outside the two glass panels. These polarisers would rotate the light rays passing through them to a definite angle, in a particular direction. When the LCD is in the off state, light rays are rotated by the two polarisers and the liquid crystal, such that the light rays come out of the LCD without any orientation, and hence the LCD appears transparent. When sufficient voltage is applied to the electrodes, the liquid crystal molecules would be aligned in a specific direction. The light rays passing through the LCD would be rotated by the polarisers, which would result in activating / highlighting the desired characters.

The LCD's are lightweight with only a few millimeters thickness. Since the LCD's consume less power, they are compatible with low power electronic circuits, and can be powered for long durations. The LCD's don't generate light and so light is needed to read the display. By using backlighting, reading is possible in the dark. The LCD's have long life and a wide operating temperature range.Changing the display size or the layout size is relatively simple which makes the LCD's more customer friendly. The LCDs used exclusively in watches, calculators and measuring instruments are the simple seven-segment displays, having a limited amount of numeric data. The recent advances in technology have resulted in better legibility, more information displaying capability and a wider temperature range. These have resulted in the LCDs being extensively used in telecommunications and entertainment electronics. The LCDs have even started replacing the cathode ray tubes (CRTs) used for the display of text and graphics, and also in small TV applications.

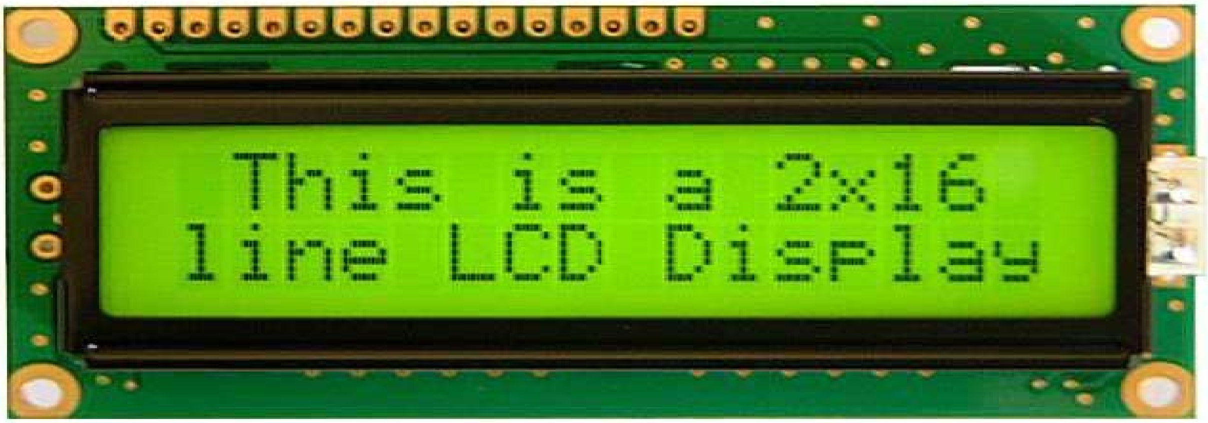


Fig 2.5 LCD display

2.6 POWER SUPPLY:

The power supply should be of +5V, with maximum allowable transients of 10mv. To achieve a better / suitable contrast for the display, the voltage (VL) at pin 3 should be adjusted properly. A module should not be inserted or removed from a live circuit. The ground terminal of the power supply must be isolated properly so that no voltage is induced in it. The module should be isolated from the other circuits, so that stray voltages are not induced, which could cause a flickering display. 1.2GHz quad core processor, WiFi, 1 GB RAM, Bluetooth, and more were present. So the USB ports allows us to connect the same wireless mice, keyboards and storage devices we use with our laptop or desktop PC.

Fig 2.6 Power supply to raspberry pi from laptop

2.7 WIFI:

A Wi-Fi-enabled device, such as a personal computer, video game console, smartphone or digital audio player, can connect to the Internet when within range of a wireless network connected to the Internet. The coverage of one or more (interconnected)access points called hotspots comprises an area as small as a few rooms or as large as many square miles. Coverage in the larger area may depend on a group of access points with overlapping coverage. Wi-Fi technology has been used successfully in wireless mesh networks in London, UK.

Wi-Fi provides service in private homes and offices as well as in public spaces at Wi-Fi hotspots set up either free-of-charge or commercially. Organizations and businesses, such as airports, hotels, and restaurants, often provide free-use hotspots to attract or assist clients. Enthusiasts or authorities who wish to provide services or even to promote business in selected areas sometimes provide free Wi-Fi access. As of 2008 more than 300 city-wide Wi-Fi (Muni-Fi) projects had been created. As of 2010 the Czech Republic had 1150 Wi-Fi based wireless Internet service providers.

Routers that incorporate a digital_subscriber_line modem or a cable modem and a Wi-Fi access point, often set up in homes and other buildings, provide Internet access and internetworking to all devices tuned into them, wirelessly or via cable. With the emergence of MiFi and WiBro (a portable Wi-Fi router) people can easily create their own Wi-Fi hotspots that connect to Internet via cellular networks. Now iPhone, Android, Bada and Symbian phones can create wireless connections. One can also connect Wi-Fi devices in ad-hoc mode for client-to-client connections without a router. Wi-Fi also connects places normally without network access, such as kitchens and garden sheds.

Setting up WiFi in Occidentalis, is also pretty straight forward. We just need to add the name of your wireless network (its SSID) and your password to a configuration file.

Step 1.

Boot the Raspberry Pi without the WiFi adapter plugged in.

Step 2.

Open a Terminal session by clicking on the LXTerminal icon, and enter the following command into it: sudo nano /etc/network/interfaces.

Step 3.

This opens an editor screen of the wifi configuration file we need to change.

Step 4.

When you have finished press [ctrl]x. This will ask if you want to save the modified files. Press 'Y' and then Return to save the file with the same name.

Step 5.

Shut down your Raspberry Pi, plug the WiFi adapter in and start it up again. You should find that the Raspberry Pi connects using the WiFi adapter as it boots up.

3 SOFTWARE USED:

3.1 MOBAXTERM:

MobaXterm is your ultimate toolbox for remote computing. In a single Windows application, it provides loads of functions that are tailored for programmers, webmasters, IT administrators and pretty much all users who need to handle their remote jobs in a more simple fashion.MobaXterm provides all the important remote network tools (SSH, X11, RDP, VNC, FTP, MOSH, ...) and Unix commands (bash, ls, cat, sed, grep, awk, rsync, ...) to Windows desktop, in a single portable exe file which works out of the box. There are many advantages of having an All-In-One network application for your remote tasks, e.g. when you use SSH to connect to a remote server, a graphical SFTP browser will automatically pop up in order to directly edit your remote files. The remote applications will also display seamlessly on the Windows desktop using the embedded X server. MobaXterm proposes an intuitive user interface in order to efficiently access remote servers through different networks or systems.

MobaXterm is undoubtedly a very useful and versatile tool. Functionally, the software is well mannered, and Windows users who want to sniff a little Linux/Unix air can get a good idea of how Linux works. A closer look reveals that anything you can do with MobaXterm can be achieved directly with freely available tools (Cygwin) and Unix tools ported from Cygwin. However, although Cygwin provides a non-Posix environment for Windows, it doesn’t offer a decent terminal, which is one thing Moba-Xterm has going for it.

Admittedly, in pure vanilla Cygwin, you can start an X server automatically and then use xterm, but xterm lacks good-quality fonts, whereas MobaXterm conveniently lets you integrate a font server.The terminal function supports anti-aliasing, fonts, and macros, and a plugin interface allows you to add features; however, these are mainly Unix and X11 tools that were ported from Cygwin.It has a considerable number of plugins up for grabs , and several third-party tools are available as well.ported from Cygwin.

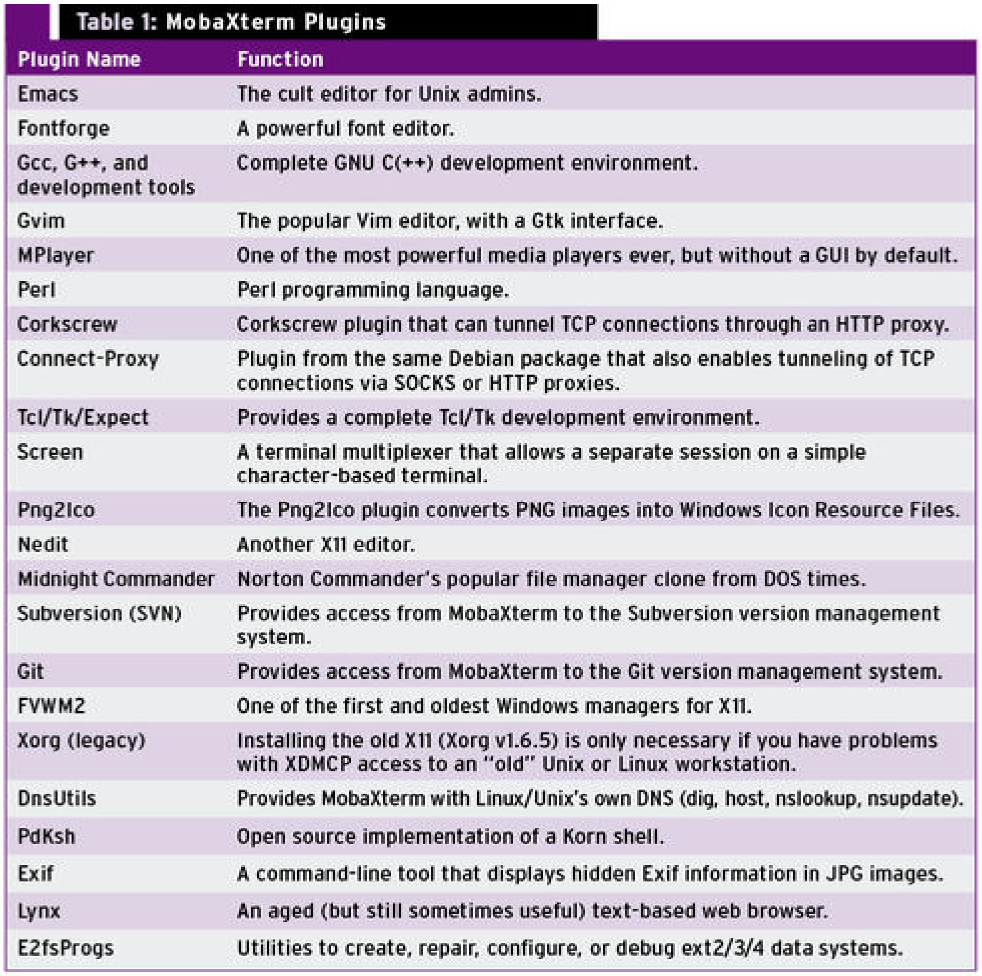


Table 1: MobaXterm Plugins

Plugin Name	Function
Emacs	The cult editor for Unix admins.
Fontforge	A powerful font editor.
Gcc, G++, and development tools	Complete GNU C(++) development environment.
Gvim	The popular Vim editor, with a Gtk interface.
MPlayer	One of the most powerful media players ever, but without a GUI by default.
Perl	Perl programming language.
Corkscrew	Corkscrew plugin that can tunnel TCP connections through an HTTP proxy.
Connect-Proxy	Plugin from the same Debian package that also enables tunneling of TCP connections via SOCKS or HTTP proxies.
Tcl/Tk/Expect	Provides a complete Tcl/Tk development environment.
Screen	A terminal multiplexer that allows a separate session on a simple character-based terminal.
Png2Ico	The Png2Ico plugin converts PNG images into Windows Icon Resource Files.
Nedit	Another X11 editor.
Midnight Commander	Norton Commander's popular file manager clone from DOS times.
Subversion (SVN)	Provides access from MobaXterm to the Subversion version management system.
Git	Provides access from MobaXterm to the Git version management system.
FVWM2	One of the first and oldest Windows managers for X11.
Xorg (legacy)	Installing the old X11 (Xorg v1.6.5) is only necessary if you have problems with XDMCP access to an "old" Unix or Linux workstation.
DnsUtils	Provides MobaXterm with Linux/Unix's own DNS (dig, host, nslookup, nsupdate).
PdKsh	Open source implementation of a Korn shell.
Exif	A command-line tool that displays hidden Exif information in JPG images.
Lynx	An aged (but still sometimes useful) text-based web browser.
E2fsProgs	Utilities to create, repair, configure, or debug ext2/3/4 data systems.

Fig 2.7 Mobaxterm plugins.

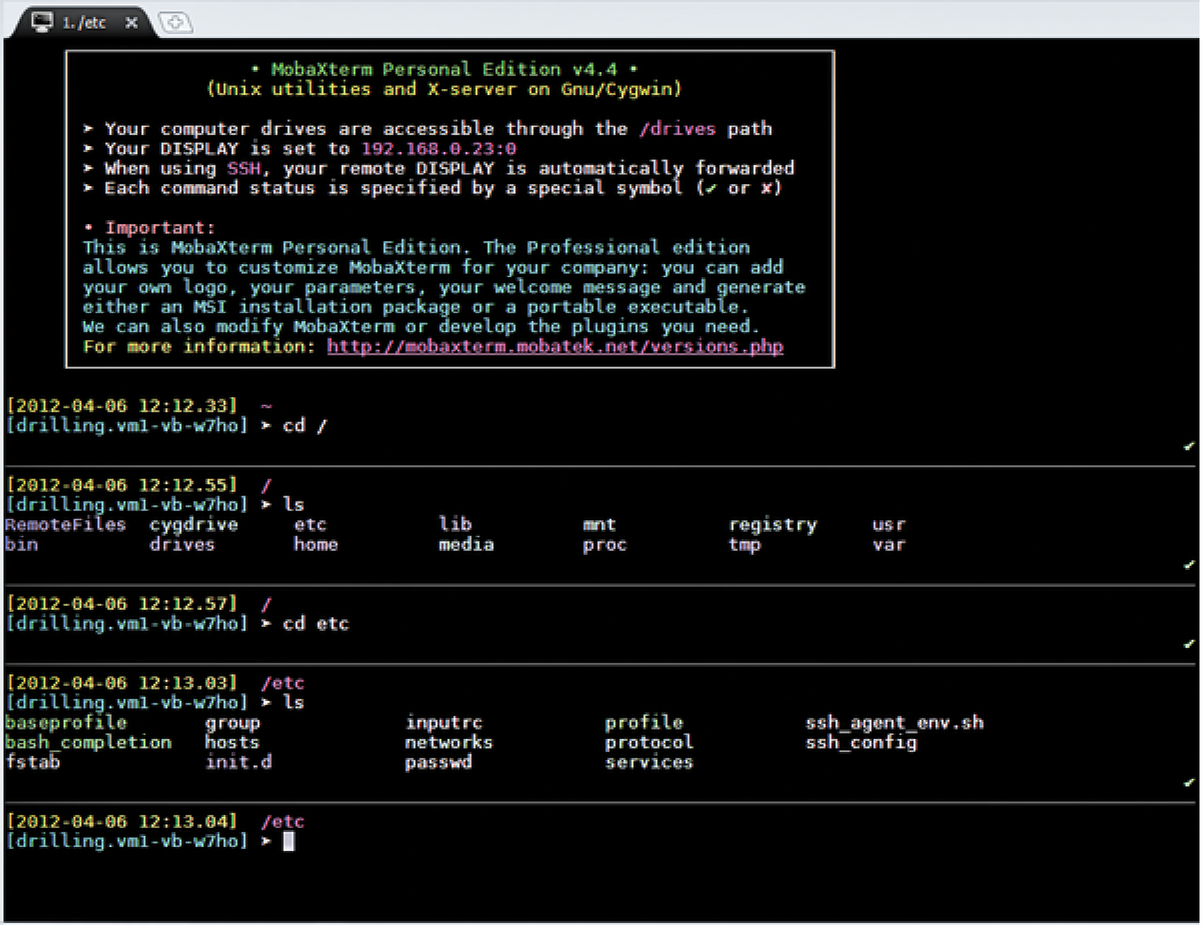
3.1.1 WALKTHROUGH:

When you first start MobaXterm, an alarm might be activated by Windows Firewall. If you then type *allow* , you can define a suitable rule. Then, MobaXterm will show you a local terminal session, including some references to the default configuration. The prompt comes up in the typical Unix manner, showing you <*User>.<Hostname>* . All drives available in Windows (as well as connected network drives) are available in typical Unix style as mountpoints below */drives* . The home screen provides information about the current IP address and the session display number. The Banner with precisely this information can be configured individually in the Pro Version's MobaXterm_Personal_Customizer . Because MobaXterm is based on Cygwin, a complete Linux directory structure is available to Windows administrators interested in the Linux adventure.

The home screen provides information about the current IP address and the session display number. The Banner with precisely this information can be configured individually in the Pro Version's MobaXterm_Personal_Customizer . Because MobaXterm is based on Cygwin, a complete Linux directory structure is available to Windows administrators interested in the Linux adventure.

The GUI is straightforward, even if the toolbar seems to be a little crowded. If you type ssh-add –I the SSH agent will list the keys saved at the time. If you change any settings here, you will need to restart MobaXterm. At the left of the terminal window is a navigation pane with two tabs: *Sessions*and *Tools* . The first tab is used to toggle between active sessions, (MobaXterm is capable of multiple sessions), which can be saved in *Sessions* . In general, the GUI is extremely flexible, and depending on the intended use, a number of views can be accessed through the same menu.
The *Tools* tab lets you access a number of mainly obsolete Terminal games and some useful system and office tools, including a separate editor, *MobaTextEditor* , and a built-in image viewer, *MobaPictureViewer* . The same programs are also available in the Tools menu. For example, *List Running Processes*shows the current task list in a new terminal tab in typical Unix Top style. In the default setting, the X server is identifiable by the green *X server* icon at the top right. If it isn't running, it can be

restarted at any time by selecting X server in the menu. The *List open network ports* system tool (Figure 3) is useful for a quick security check.



```
              • MobaXterm Personal Edition v4.4 •
           (Unix utilities and X-server on Gnu/Cygwin)

   ➤ Your computer drives are accessible through the /drives path
   ➤ Your DISPLAY is set to 192.168.0.23:0
   ➤ When using SSH, your remote DISPLAY is automatically forwarded
   ➤ Each command status is specified by a special symbol (✓ or ✗)

   • Important:
   This is MobaXterm Personal Edition. The Professional edition
   allows you to customize MobaXterm for your company: you can add
   your own logo, your parameters, your welcome message and generate
   either an MSI installation package or a portable executable.
   We can also modify MobaXterm or develop the plugins you need.
   For more information: http://mobaxterm.mobatek.net/versions.php

[2012-04-06 12:12.33]  ~
[drilling.vm1-vb-w7ho] ➤ cd /
                                                                              ✓
[2012-04-06 12:12.55]  /
[drilling.vm1-vb-w7ho] ➤ ls
RemoteFiles  cygdrive    etc         lib         mnt         registry    usr
bin          drives      home        media       proc        tmp         var
                                                                              ✓
[2012-04-06 12:12.57]  /
[drilling.vm1-vb-w7ho] ➤ cd etc
                                                                              ✓
[2012-04-06 12:13.03]  /etc
[drilling.vm1-vb-w7ho] ➤ ls
baseprofile      group            inputrc          profile          ssh_agent_env.sh
bash_completion  hosts            networks         protocol         ssh_config
fstab            init.d           passwd           services
                                                                              ✓
[2012-04-06 12:13.04]  /etc
[drilling.vm1-vb-w7ho] ➤ █
```

Fig 2.8 Mobaxterm commands

3.2 SSH TOOLS:

A feature that some administrators will definitely appreciate is the navigation section on the *SFTP* tab which appears when you click the New session button, choose *<SSH>* , and click Activate the graphical SSH browser when connected .

It is well known that SSH can be used to redirect the X11 screen output of a remotely launched Unix program to a local display (X11 Port Forwarding). SSH tunnels and encrypts the transferred image content and settings. To do this, you need to select the */etc/ssh/sshd_config* file's *X11Forwarding Yes* option server-side. In most Linux distributions, this is the default setting. The option *Enable X11 Forwarding* in the MobaXterm settings must also be selected. After doing both of these things, you will be able to log in to the system as follows:

```
ssh -XC <Username>@<Servername>
```

The -*X* ensures that SSH gets ready to transfer X11 files as soon as a connection is established. The -*C* option enables compression for the transferred screen content. After logging in with MobaXterm in Windows, you will be able to run any graphical X11 program on the remote Linux host; the program's screen content will then immediately be redirected to the Windows screen (one example of this is XEmacs.Obviously this only works for "real" X11 tools. However, you will probably be aware of the huge selection of capable X tools on any modern distribution. Typing *x* -Tab reveals many of these. Additionally, the whole repertoire of SSH functions is available to you with MobaXterm on a Windows computer – including a safer, ad hoc VPN solution in the form of port forwarding.

4.WORKING PROCEDURE:

- The system consists of the components such as IR sensors,raspberry pi and LCD display.
- The sensors are interfaced with raspberry pi and the output which displays whether a car is parked or not is shown in the LCD display.
- The sensors IR1,IR2,IR3,IR4,IR5,IR6 are connected to the pins GPIO02,GPIO03, GPIO04, GPIO17, GPIO27, GPIO22 of raspberry pi respectively.
- The power supply to raspberry pi is given from the PC.
- The OS of raspberry pi used in this system is Jessie. Open the operating system by giving the IP address as 192.168.43.101.
- After the OS gets opened type the python program and save it with extension .py.
- Now open the MobaXterm software where we want to load the main program by giving the following commands:

  - ➢ To load the list of files give the command, ls →Enter
  - ➢ To open the command directory of project code give the command cd\project code→Enter.
  - ➢ To run the main program the command is python main.py→Enter is given.
  - ➢ To stop the program which is running give the command Alt+z.
  - ➢ To exit the window give Halt.

4.1 BOOKING:

In urban areas finding parking space is difficult in rush hours. So here we developed booking system by using this we can prelock the parking slots in advance.We can book the parking slots by using a web portal. The booking procedure explained in the below flow chart. Database provides parking availability information; if there are any vacant slots available then we can book the interested slot. The booking includes the following steps:

Step 1: Open the android phone.

Step 2: Connect to the wifi hotspot named raspberry and type the password as 'jessie123'.

Step 3: Enter the IP address in the browser as '192.168.43.181".

Step 4: The webpage of slots will gets opened.

Step 5: The green box indicated the free slots,the red box indicates parked slots and the yellow box indicates the booked slots.

Step 6: Click on slot1 to book the first slot, the box's colour changes from green to yellow which means that slot has been booked.

Step 7: The slot will be booked for 10s and again it will change to free slots i.e the box will yellow for 10s and after that it will be changed to green if the car has not parked within that time and if the car is parked in that slot it will change to red colour which indicates the parked slot.

4.2 FLOWCHART:

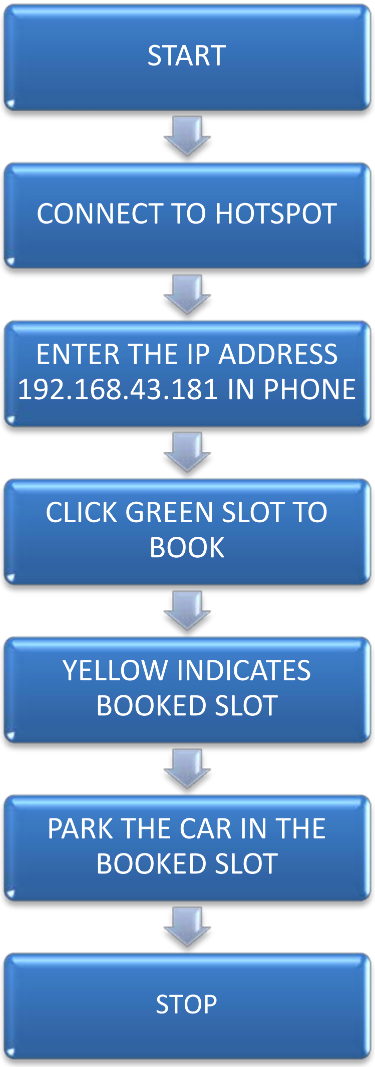

START
CONNECT TO HOTSPOT
ENTER THE IP ADDRESS 192.168.43.181 IN PHONE
CLICK GREEN SLOT TO BOOK
YELLOW INDICATES BOOKED SLOT
PARK THE CAR IN THE BOOKED SLOT
STOP

Fig 4.1 Flowchart for booking process

4.3 PROGRAM:

```
import RPi.GPIO as GPIO
import time
#from lcd import *
from lcd_raspi import*
import os
from time import gmtime,strftime
import smtplib
import sys
import datetime
import numpy as np
ir1 = 2
ir2 = 3
ir3 = 4
ir4 = 17
ir5 = 27
ir6 = 22
np.ps=[0,0,0,0,0,0]
_assign = ""

global
now,h,m,s,pre_result,result,sec_assign,c1,c2,c3,c4,c5,c6,c1_y,c2_y,c3_y,c4_y,c5_y,c
6_y
now=0
h=0
m=0
```

```
s=0
pre_result=0
result=0
sec_assign=0
c1=0
c2=0
c3=0
c4=0
c5=0
c6=0
c1_y=0
c2_y=0
c3_y=0
c4_y=0
c5_y=0
c6_y=0

GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.setup(ir1, GPIO.IN, pull_up_down = GPIO.PUD_UP)
GPIO.setup(ir2, GPIO.IN, pull_up_down = GPIO.PUD_UP)
GPIO.setup(ir3, GPIO.IN, pull_up_down = GPIO.PUD_UP)
GPIO.setup(ir4, GPIO.IN, pull_up_down = GPIO.PUD_UP)
GPIO.setup(ir5, GPIO.IN, pull_up_down = GPIO.PUD_UP)
GPIO.setup(ir6, GPIO.IN, pull_up_down = GPIO.PUD_UP)
lcd_init()
time.sleep(0.5)
```

```
lcd_string("    SMART CAR   ",LCD_LINE_1)
lcd_string(" PARKING SYSTEM ",LCD_LINE_2)
time.sleep(1)
lcd_byte(0x01,LCD_CMD)

def pre_clr():
    global now,h,m,s,pre_result
    now = datetime.datetime.now()
    h = now.hour
    m = now.minute
    s = now.second
    pre_result = int(h) * 3600 + int(m) * 60 + int(s)

def now_calc():
    global now,h,m,s,pre_result,result,sec_assign,c1,c2,c3,c4,c5,c6
    now = datetime.datetime.now()
    h = now.hour
    m = now.minute
    s = now.second
    result = int(h) * 3600 + int(m) * 60 + int(s)
    result = result - pre_result

    if sec_assign != result:
        c1 = c1 + 1
        c2 = c2 + 1
        c3 = c3 + 1
        c4 = c4 + 1
        c5 = c5 + 1
        c6 = c6 + 1
        print c1,c2,c3,c4,c5,c6
    sec_assign = result

    #print "sec............... ",result

fo = open('/var/www/html/c1y.txt',"w")
fo.write('0')
fo.close()
fo = open('/var/www/html/c2y.txt',"w")
fo.write('0')
fo.close()
fo = open('/var/www/html/c3y.txt',"w")
fo.write('0')
fo.close()
fo = open('/var/www/html/c4y.txt',"w")
 fo.write('0')
fo.close()
fo = open('/var/www/html/c5y.txt',"w")
fo.write('0')
fo.close()
fo = open('/var/www/html/c6y.txt',"w")
fo.write('0')
fo.close()
pre_clr()
```

```
while (True):
    now_calc()
    if GPIO.input(ir1) == False:
     np.ps[0]=0
    if GPIO.input(ir1) == True:
     np.ps[0]=1
   if GPIO.input(ir2) == False:
  np.ps[1]=0
    if GPIO.input(ir2) == True:
        np.ps[1]=1
  if GPIO.input(ir3) == False:
        np.ps[2]=0
    if GPIO.input(ir3) == True:
        np.ps[2]=1
    if GPIO.input(ir4) == False:
        np.ps[3]=0
    if GPIO.input(ir4) == True:
        np.ps[3]=1
    if GPIO.input(ir5) == False:
        np.ps[4]=0
    if GPIO.input(ir5) == True:
        np.ps[4]=1
    if GPIO.input(ir6) == False:
        np.ps[5]=0
    if GPIO.input(ir6) == True:
        np.ps[5]=1
    _assign = str(np.ps)
    _assign = _assign.replace(',','')
    lcd_string(_assign,LCD_LINE_1)
    lcd_string("CAR PARKING    ",LCD_LINE_2)

   fo = open('/var/www/html/c1y.txt',"r")
    c1_y = fo.read()
    fo.close()
    #print "c1_y = ",c1_y

    fo = open('/var/www/html/c2y.txt',"r")
    c2_y = fo.read()
    fo.close()
    #print "c2_y = ",c2_y

    fo = open('/var/www/html/c3y.txt',"r")
    c3_y = fo.read()
    fo.close()
    #print "c3_y = ",c3_y
  fo = open('/var/www/html/c4y.txt',"r")
    c4_y = fo.read()
    fo.close()
    #print "c4_y = ",c4_y

    fo = open('/var/www/html/c5y.txt',"r")
    c5_y = fo.read()
    fo.close()
    #print "c5_y = ",c5_y
```

```
fo = open('/var/www/html/c6y.txt',"r")
c6_y = fo.read()
fo.close()
#print "c6_y = ",c6_y
#----------car 1--------------------------------
if(c1_y=='0'):
    if GPIO.input(ir1) == True:
        fo = open('/var/www/html/c1.txt',"w")
        fo.write('1')
        fo.close()
        c1=0
    elif GPIO.input(ir1) == False:
        fo = open('/var/www/html/c1.txt',"w")
        fo.write('2')
        fo.close()
        c1=0
if(c1_y=='1') and (c1<10):
    if GPIO.input(ir1) == True:
        fo = open('/var/www/html/c1.txt',"w")
        fo.write('1')
        fo.close()

        fo = open('/var/www/html/c1y.txt',"w")
        fo.write('0')
        fo.close()
        c1=0
    elif GPIO.input(ir1) == False:
        fo = open('/var/www/html/c1.txt',"w")
        fo.write('3')
        fo.close()
if(c1_y=='1') and (c1>10):
    print "File overwrite"
    fo = open('/var/www/html/c1y.txt',"w")
    fo.write('0')
    fo.close()
    c1=0

#----------car 2--------------------------------
if(c2_y=='0'):
    if GPIO.input(ir2) == True:
        fo = open('/var/www/html/c2.txt',"w")
        fo.write('1')
        fo.close()
        c2=0
    elif GPIO.input(ir2) == False:
        fo = open('/var/www/html/c2.txt',"w")
        fo.write('2')
        fo.close()
        c2=0
if(c2_y=='1') and (c2<10):
    if GPIO.input(ir2) == True:
        fo = open('/var/www/html/c2.txt',"w")
        fo.write('1')
        fo.close()
        fo = open('/var/www/html/c2y.txt',"w")
        fo.write('0')
        fo.close()
        c2=0
    elif GPIO.input(ir2) == False:
        fo = open('/var/www/html/c2.txt',"w")
        fo.write('3')
        fo.close()
if(c2_y=='1') and (c2>10):
    print "File overwrite"
    fo = open('/var/www/html/c2y.txt',"w")
    fo.write('0')
    fo.close()
    c2=0

#----------car 3---------------------------------
if(c3_y=='0'):
    if GPIO.input(ir3) == True:
        fo = open('/var/www/html/c3.txt',"w")
        fo.write('1')
        fo.close()
        c3=0
    elif GPIO.input(ir3) == False:

        fo = open('/var/www/html/c3.txt',"w")
        fo.write('2')
```

```
            fo.close()
            c3=0
    if(c3_y=='1') and (c3<10):
        if GPIO.input(ir3) == True:
            fo = open('/var/www/html/c3.txt',"w")
            fo.write('1')
            fo.close()

            fo = open('/var/www/html/c3y.txt',"w")
            fo.write('0')
            fo.close()
            c3=0
        elif GPIO.input(ir3) == False:

fo = open('/var/www/html/c3.txt',"w")
            fo.write('3')
            fo.close()
    if(c3_y=='1') and (c3>10):
        print "File overwrite"
        fo = open('/var/www/html/c3y.txt',"w")
        fo.write('0')
        fo.close()
        c3=0



    #----------car 4--------------------------------
    if(c4_y=='0'):
        if GPIO.input(ir4) == True:

    fo = open('/var/www/html/c4.txt',"w")
           fo.write('1')
           fo.close()
           c4=0
        elif GPIO.input(ir4) == False:
           fo = open('/var/www/html/c4.txt',"w")
           fo.write('2')
 fo.close()
 c4=0
   if(c4_y=='1') and (c4<10):
        if GPIO.input(ir4) == True:
           fo = open('/var/www/html/c4.txt',"w")
           fo.write('1')
           fo.close()

           fo = open('/var/www/html/c4y.txt',"w")
           fo.write('0')
           fo.close()
           c4=0
        elif GPIO.input(ir4) == False:
           fo = open('/var/www/html/c4.txt',"w")
           fo.write('3')
           fo.close()
    if(c4_y=='1') and (c4>10):
        print "File overwrite"
    fo = open('/var/www/html/c4y.txt',"w")
    fo.write('0')
    fo.close()
    c4=0

#----------car 5--------------------------------
if(c5_y=='0'):
    if GPIO.input(ir5) == True:
        fo = open('/var/www/html/c5.txt',"w")
        fo.write('1')
        fo.close()
        c5=0
    elif GPIO.input(ir5) == False:
        fo = open('/var/www/html/c5.txt',"w")
        fo.write('2')
        fo.close()
        c5=0
if(c5_y=='1') and (c5<10):
    if GPIO.input(ir5) == True:
        fo = open('/var/www/html/c5.txt',"w")
        fo.write('1')
        fo.close()

        fo = open('/var/www/html/c5y.txt',"w")
        fo.write('0')
        fo.close()
        c5=0
        elif GPIO.input(ir5) == False:
            fo = open('/var/www/html/c5.txt',"w")
            fo.write('3')
            fo.close()
    if(c5_y=='1') and (c5>10):
        print "File overwrite"
        fo = open('/var/www/html/c5y.txt',"w")
        fo.write('0')
        fo.close()
        c5=0



#----------car 6---------------------------------
    if(c6_y=='0'):
        if GPIO.input(ir6) == True:
            fo = open('/var/www/html/c6.txt',"w")
            fo.write('1')
            fo.close()
            c6=0
        elif GPIO.input(ir6) == False:
            fo = open('/var/www/html/c6.txt',"w")
fo.write('2')
            fo.close()
            c6=0
    if(c6_y=='1') and (c6<10):
if GPIO.input(ir6) == True:
            fo = open('/var/www/html/c6.txt',"w")
```

```
fo.write('1')
            fo.close()
            fo = open('/var/www/html/c6y.txt',"w")
            fo.write('0')
            fo.close()
            c6=0
        elif GPIO.input(ir6) == False:
 fo = open('/var/www/html/c6.txt',"w")
            fo.write('3')
fo.close()
    if(c6_y=='1') and (c6>10):
print "File overwrite"
        fo = open('/var/www/html/c6y.txt',"w")
fo.write('0')
        fo.close()
        c6=0
GPIO.cleanup()
```

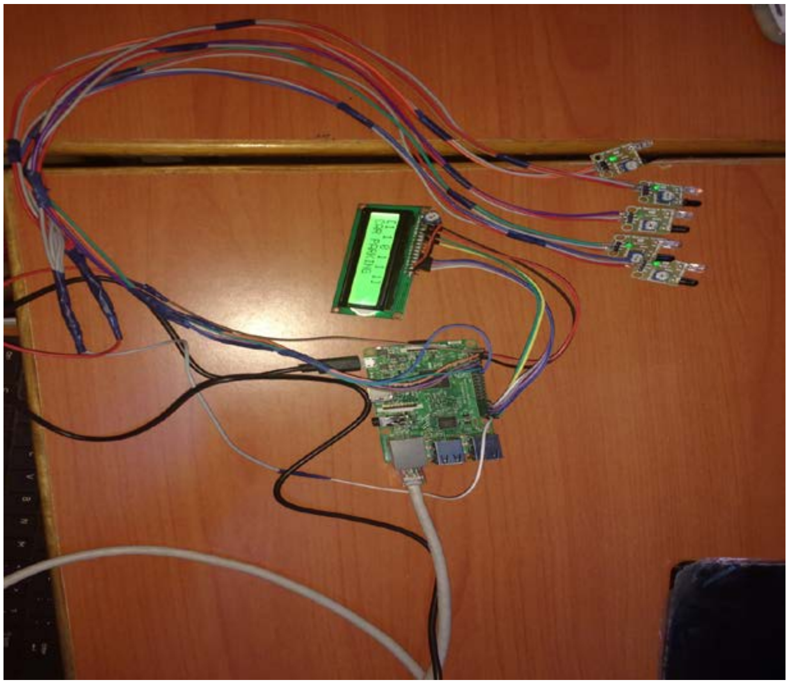
5.ADVANTAGES:

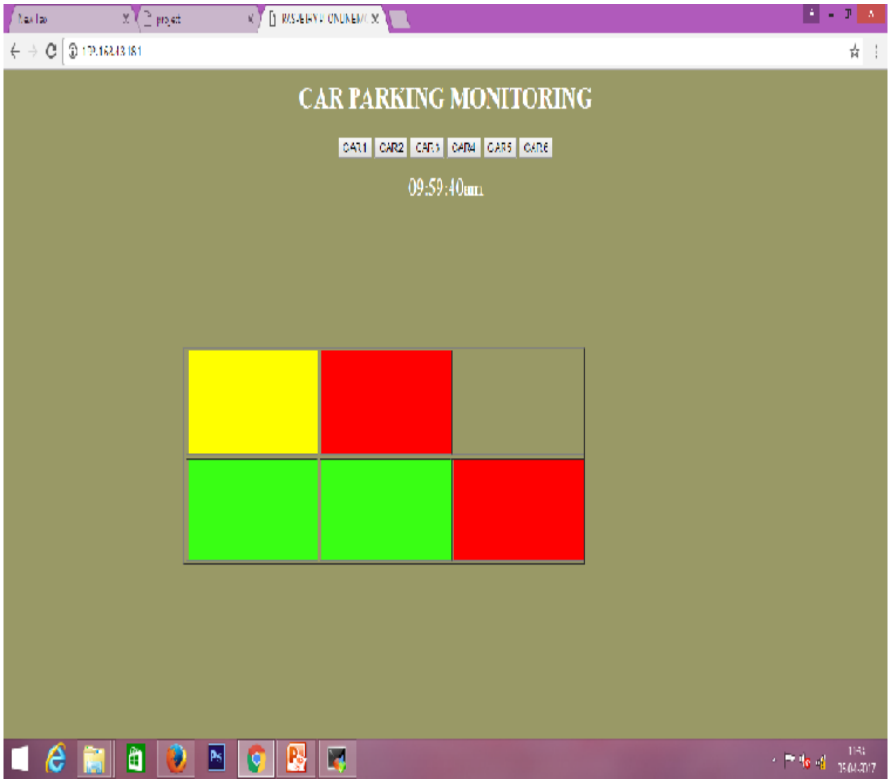
- Guides drivers find available parking spaces near them
- Less number of drivers searching to park, thus reduces the traffic congestion
- Avoids air pollution & global warming.
- Scalable, robust and reliable.
- Reduces the driver stress and improves the urban area.
- Provides tools to optimize the parking space management
- Accurately find out the vehicle occupancy in real time.

5.1 REAL TIME APPLICATIONS:

- In real time applications the IR sensors can be replaced by the ultrasonic sensors where the sensing range of ultrasonic sensor is greater than the IR sensors.
- The waiting time for the booked slots can also be increased for real time applications.

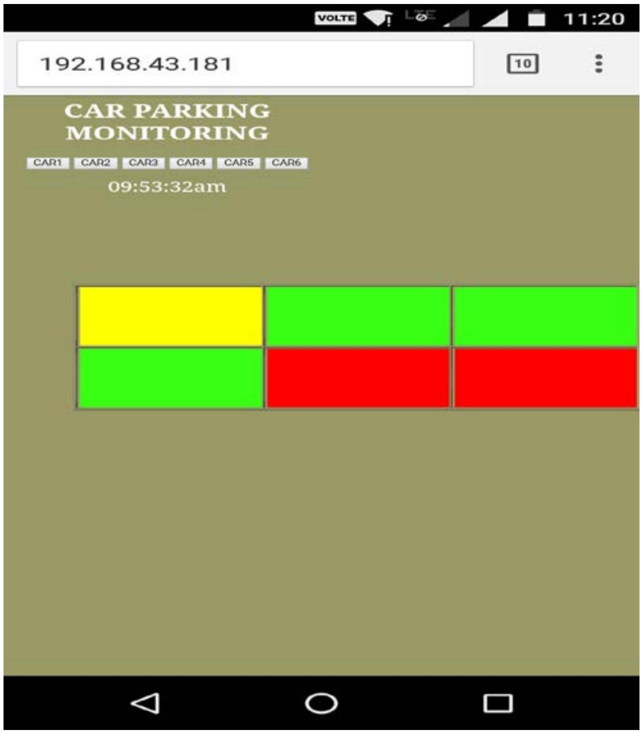
6.IMAGES:





CAR PARKING MONITORING

09:59:40am



CAR PARKING MONITORING

09:53:32am

7.CONCLUSION:

This system explicitly reserves and allocates optimal parking space to drivers, as opposed to simply guiding them to a space that may not be available by the time it is reached.Using the slot allocation method we can book and block our own cheapest and shortest distant parking slot.Thus, this concludes that the system simplifies the context for the researchers for innovating various techniques to administrate and solve the problems faced by drivers on day to day basis.In future, the system can be extended which is not only specific to a private parking like Malls, Company parking, etc. but also can be implemented over various multiple platforms such as public parking also extending the feature by giving parking information based on cost in real time. This will make the management of the parking spaces more efficient, by purging the need of human labor.

8.REFERENCES:

1. Gallo M., D.’Acierno L. and Montella B. 2014. “A multilayer model to simulate cruising for parking in urban areas,” Transp. Policy. Vol. 18, no. 5, pp. 735 744, September.
2. Geng Y. and Cassandras C. G. 2014. “A new “smart parking”system based on optimal resource allocation and reservations,” in Proc. IEEE Conf. Intell. Transp.Syst. pp. 979–984.
3. Hitendra G., Wasnik Askhedkar R. D. and Choudhary S. K. 2013. “Optimal Automatic Car Parking System for Indian Environment” Indian streams research journal Vol. 1, pp.1-4.
4. Hilal Al-Kharusi, Ibrahim Al-Bahadly, “Intelligent Parking Management System Based on Image Processing”, World Journal of Engineering and Technology, 2014, 2, 55-67.
5. 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.
6. 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.