EXPERIMENT No. Practicals 1

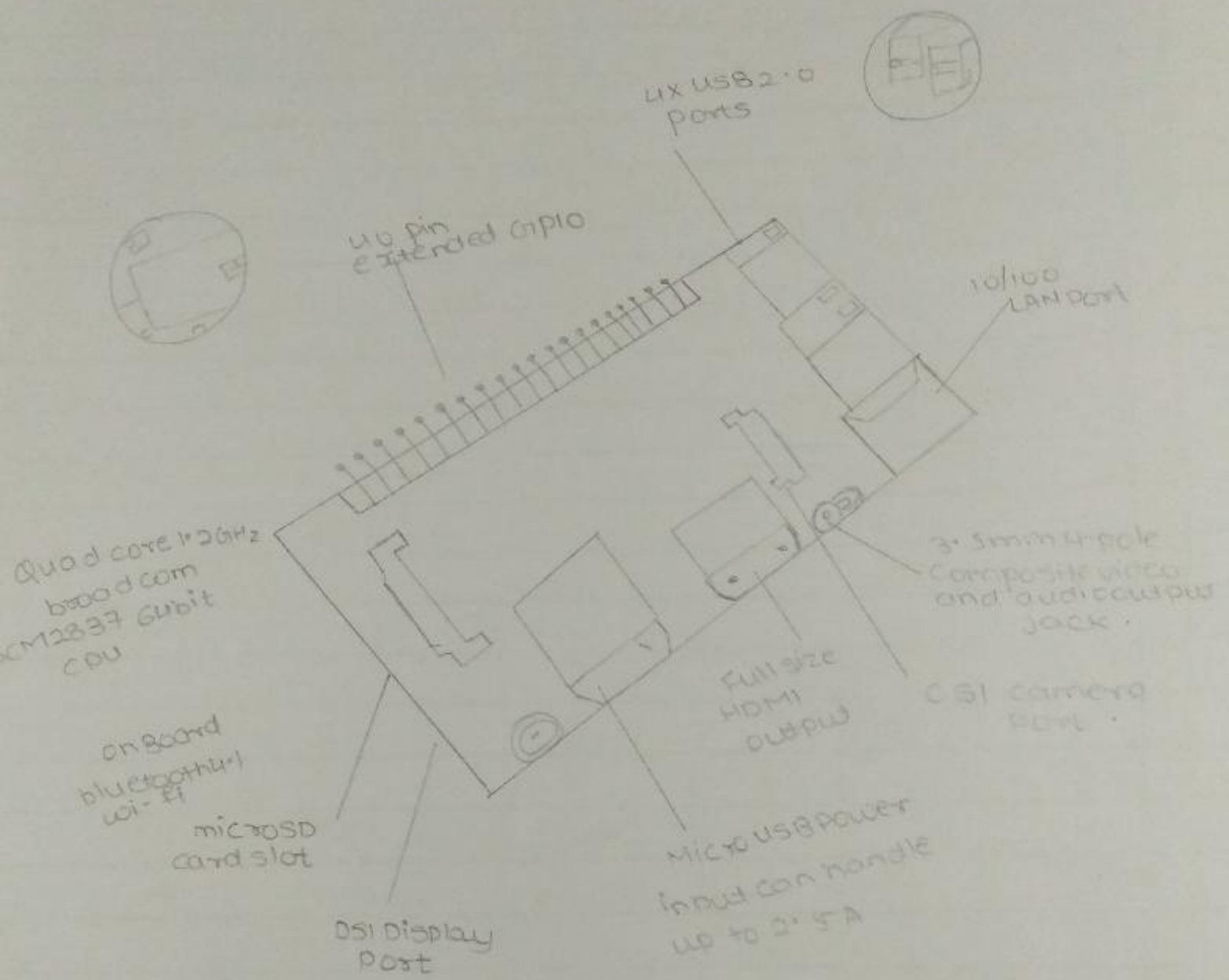
Page No. 1
Date

AIM :-
Starting raspbian Os, familiarizing with raspberry pi components and interface, connecting to ethernet, monitor, USB.

Required Components :-
1. Hardware :-

Particular	Quantity
Raspberry Pi3	1
power Supply 12v/2Amp	1
USB keyboard	1
USB mouse	1
Micro SD card	1
Micro SD USB card reader	1
A Monitor that supports HDMI	1
An HDMI cable	1
An Ethernet cable	1

Output.


4x USB2.0 ports
40 pin extended GPIO
10/100 LAN port
Quad core 1.2GHz broadcom BCM2837 64bit CPU
3.5mm 4-pole composite video and audio output jack
Full size HDMI output
CSI camera port
on board bluetooth 4.1 wi-fi
micro SD card slot
Micro USB power input can handle up to 2.5A
DSI Display Port

2. Software :-
Raspbian, installed via NOOBS.

1) USB ports
2) SD Card slot
3) Ethernet Port
4) Audio Jack
5) HDMI Port
6) Micro USB Power Connector.
7) GPIO ports.

Procedure :-

1) Insert the SD Card into your SD card reader. note the drive letter assigned to the SD card. you can see the drive letter in the right hand columns of windows Explorer.

2) Run the Win 32 Disk Imager utility from desktop or menu.

3) Select the image file and click on write button.

4) Remove SD card and insert in raspberry Pi

) Connect the mouse and keyboard to USB port of raspberry pi.

Connect raspberry pi to HDMI port directly or use HDMI to VGA Connector.

7) Connect the Pi to the internet via ethernet, use an ethernet cable to connect the raspberry pi.

8) Sound will come if it has speakers or connect headphones or speakers to the audio jack if necessary.

9) plug the power supply into a socket and connect it to the micro USB power port.

10) Red light will indicate power on and Green will indicate the booting of the raspberry pi. The Pi will boot up into a graphical desktop.

11) Configuring your Pi:

a) you can control most of your raspberry Pi's settings such as the password, through the raspberry Pi configuration application found in Preferences on the menu.

b) System: In this tab you can change basic system settings of your pi.

1) password.
2) Boot.
3) Auto login.
4) Network at Boot.
5) Splash Screen.
6) Resolution.
7) Underscan.

c) Interfaces.

1) Camera
2) SSH
3) VNC
4) SPI
5) I2C
6) Serial
7) 1-wire
8) Remote GPIO

d) Performance : If you need to do so for a particular project you want to work on, you can change the performance settings of your pi in this tab.

1) Overclock.
2) GPU Memory

e) Localisation.

12) This tab allows you to change your raspberry pi settings to be specific to a country or location.

1) Locale : set the language, country and character set used by your raspberry pi.

2) Timezone : set the time zone.

3) Keyboard : change your keyboard layout.

4) Wi-Fi Country : set the wi-fi country code.

13) After starting with raspberry pi for the first time, the welcome to raspberry pi application will pop up and it will guide through the initial set up.

14) Click Next to start the setup.

15) set respective country, Language, and time zone, then click Next again.

16) Enter a new password for raspberry pi and click next.

17) Connect to your wi-fi network by selecting its name, entering the password, and clicking next.

18) Click next let the wizard check for updates to raspbian and install

19) Click done or reboot to finish the setup.

Conclusion:-

Thus raspbian Os was installed, raspberry pi components and interface were studied and implemented, raspberry was also connected to ethernet, monitor, USB.

EXPERIMENT: No. Practical 2

Page No. 6
Date

AIM:-
Displaying different LED patterns with raspberry Pi.

Required Components:-

Particular	Quantity
Raspberry Pi3	1
Power supply 12V/2Amp	1
USB Keyboard	1
USB mouse	1
micro SD card	1
MAX7219	1
Discover board.	1

Procedure:-
step 1:- Connection of Discover development board.

step 2:- Connection of raspberry pi.

step 3:- Switch ON power supply.

step 4:- Login to raspberry pi terminal.
a) Username: pi
b) password: raspberry.

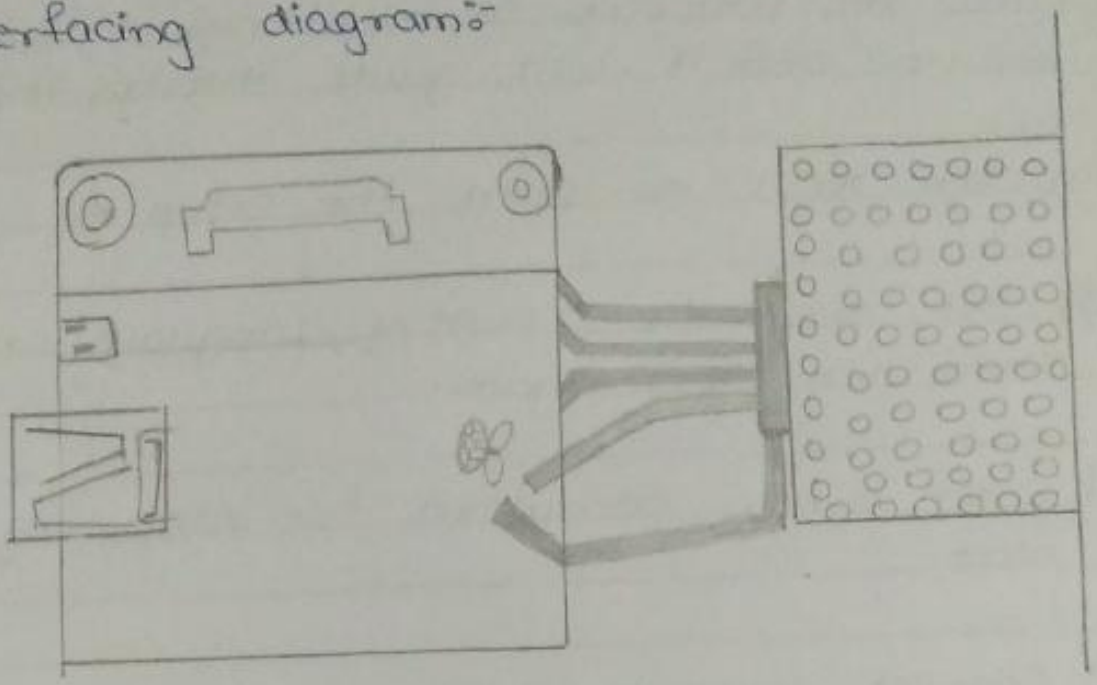
step 5:- Create a new file with an extension .py.

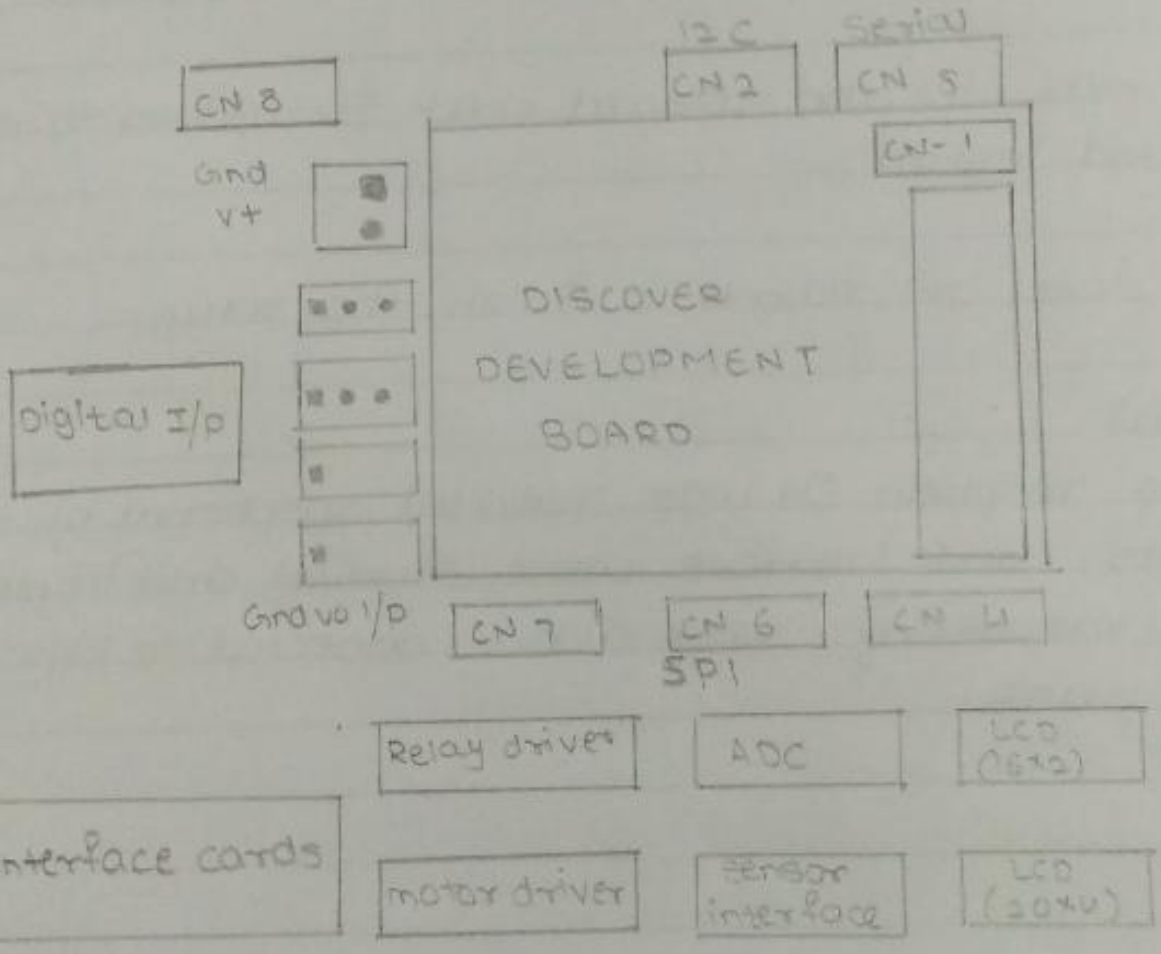
Step 6:- Open the file with python 2 IDLE only.

Teacher's Sign.:

Output :-

Interfacing diagram :-





EXPERIMENT: No.

Page No. 7
Date

step 7:- Type and run the program and see the output on 7 segment display board.

Code:-

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
import re
import time
import argparse
from luma.led_matrix.device import max7219
from luma.core.interface.serial import spi, noop
from luma.core.render import canvas
from luma.core.virtual import viewport
from luma.core.legacy import text, show_message
from luma.core.legacy.font import proportional,
CP437_FONT, TINY_FONT, SINCLAIR_FONT, LCD_FONT
def demo(n, block_orientation, rotate, msg):
# Create matrix device.
serial = spi(port=0, device=0, gpio=noop())
device = max7219(serial, cascaded=n or 1, block_Orienta-
tion=block_Orientation,
rotate=rotate or 0)
show_message(device, msg, fill="white", font=proport
ional(LCD_FONT), scroll_delay=0.1)
time.sleep(3)
pass
if name == "main":
try:
text_display = raw_input("Enter message to be display
```

Teacher's Sign. :

EXPERIMENT : No.

Page No. 8
Date

8x 8 matrix = ")
demo (1,0 ,0 , text_display)
except Keyboard Interrupt:
pass
finally:
print "program exit...."

Conclusion:
Thus we have studied and displayed different LED patterns with raspberry pi.

EXPERIMENT No. Practical 3 Page No. 9 Date

AIM :

Displaying time over 4-digit 7-segment display using raspberry pi.

Required Components :-

Particular	Quantity.
Raspberry pi 3	1
power supply 12V/2 Amp	1
USB keyboard	1
USB Mouse	1
Micro SD Card	1
4-Digit 7-segment display module.	1
Discover board.	1

Procedure :-

Step 1 :- Connection of discover development board.

Step 2 :- Connection of raspberry pi.

Step 3 :- Switch ON power supply.

Step 4 :- Login to raspberry pi terminal.

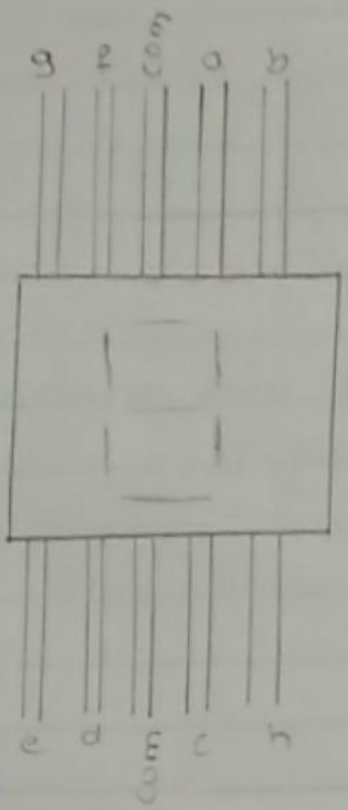
a) username :- pi

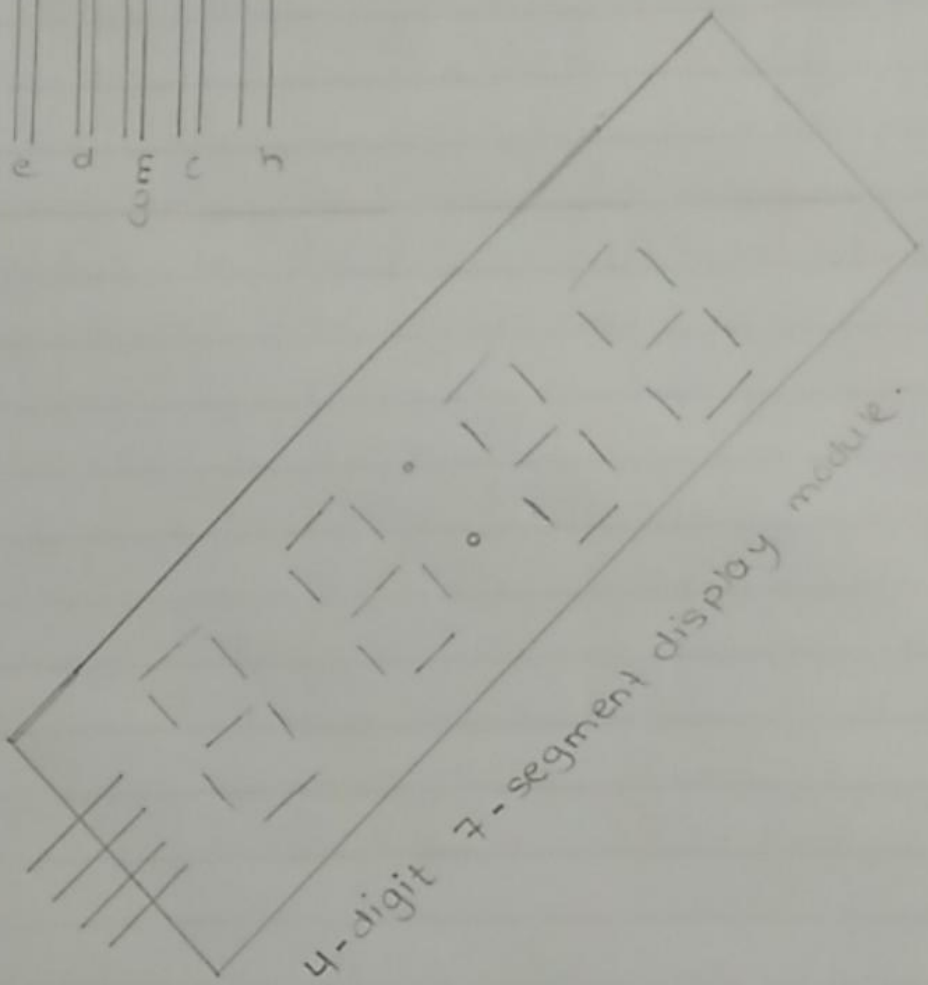
b) password :- raspberry.

Step 5 : Create a new file with an extension .py

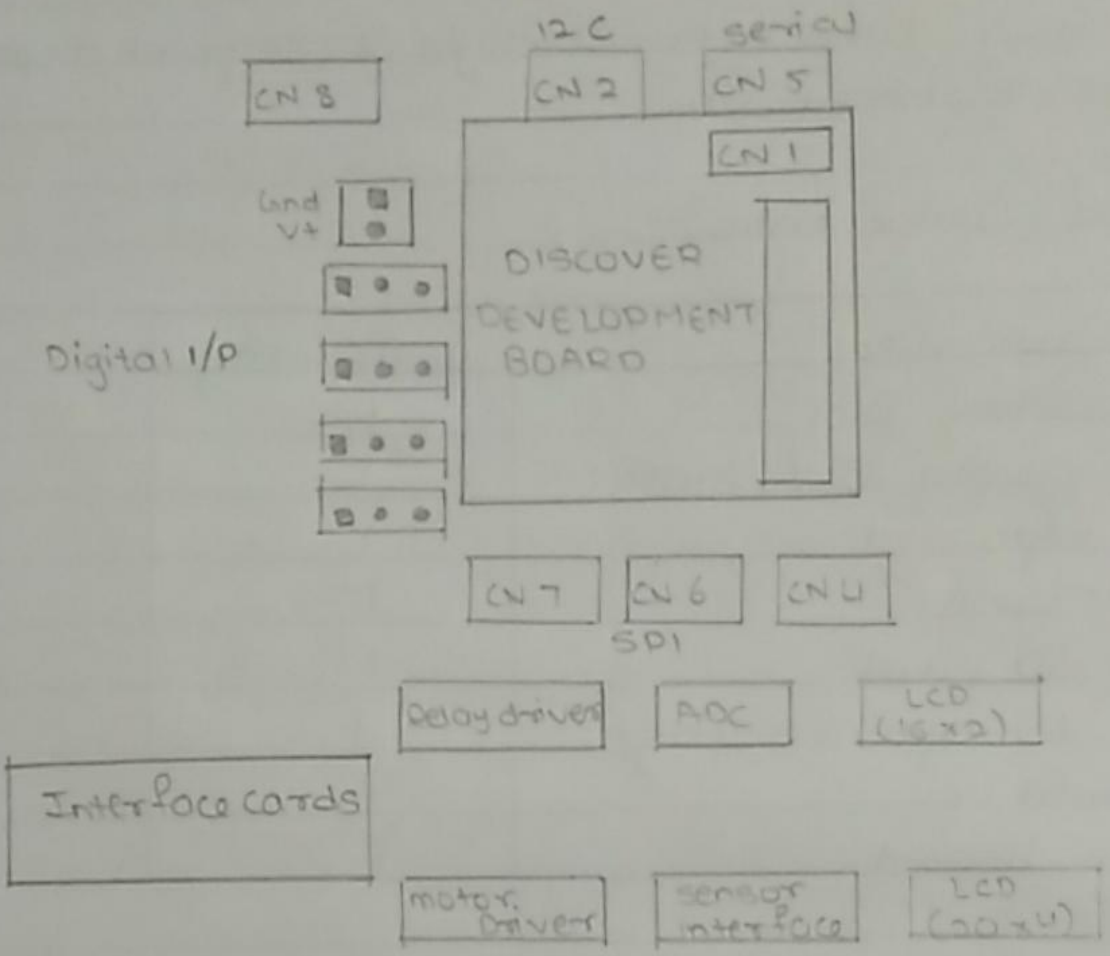
Teacher's Sign :

Output :-





Output :-


I2C
serial
CN 8
CN 2
CN 5
CN 1
Gnd
V+
DISCOVER
DEVELOPMENT
BOARD
Digital I/P
CN 7
CN 6
CN 4
SPI
Delay driver
ADC
LCD
(16 x 2)
Interface cards
motor
Driver
sensor
interface
LCD
(20 x 4)

Step 6:- Open the file with python 2 IDLE only

Step 7:- Type and run the program and see the output on 4-digit 7 segment display board.

Code :-

```
#!/usr/bin/python
import time
import datetime
from lib import tm1637 as obj
Display = obj.TM1637(2,3,5)
Display.clear()
while(True):
now = datetime.datetime.now()
hour = now.hour
minute = now.minute
second = now.second
Display.clear()
val = [(int(hour/10)),(hour%10),(int(minute/10)),
    (minute%10)]
Display.show(val)
Display.showDoublepoint((second%2))
time.sleep(0.25)
```

Conclusion:-

Thus we have studied and displayed time over 4-digit 7-segment display using raspberry pi.

EXPERIMENT No. Practical 4

Page No. 11
Date

AIM:- Raspberry pi based Oscilloscope

Required Components:

Particular	Quantity
Raspberry pi 3	1
power supply 12V/2Amp	1
USB keyboard	1
USB mouse	1
micro SD card	1
AD9115 ADC	1
Discover board	1
Analog input as per availability	1

Procedure:-

Step1 :- Connection of discover development board.

step2 :- Connection of raspberry pi.

step3 :- Switch ON power supply.

step4 :- Login to raspberry pi terminal.

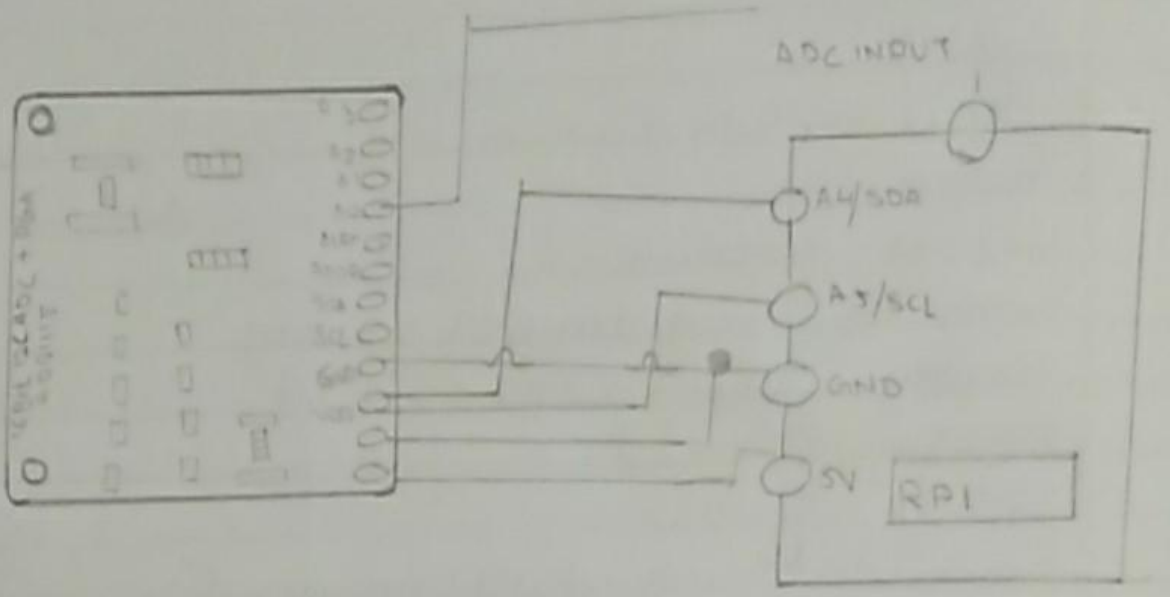
a) username : pi

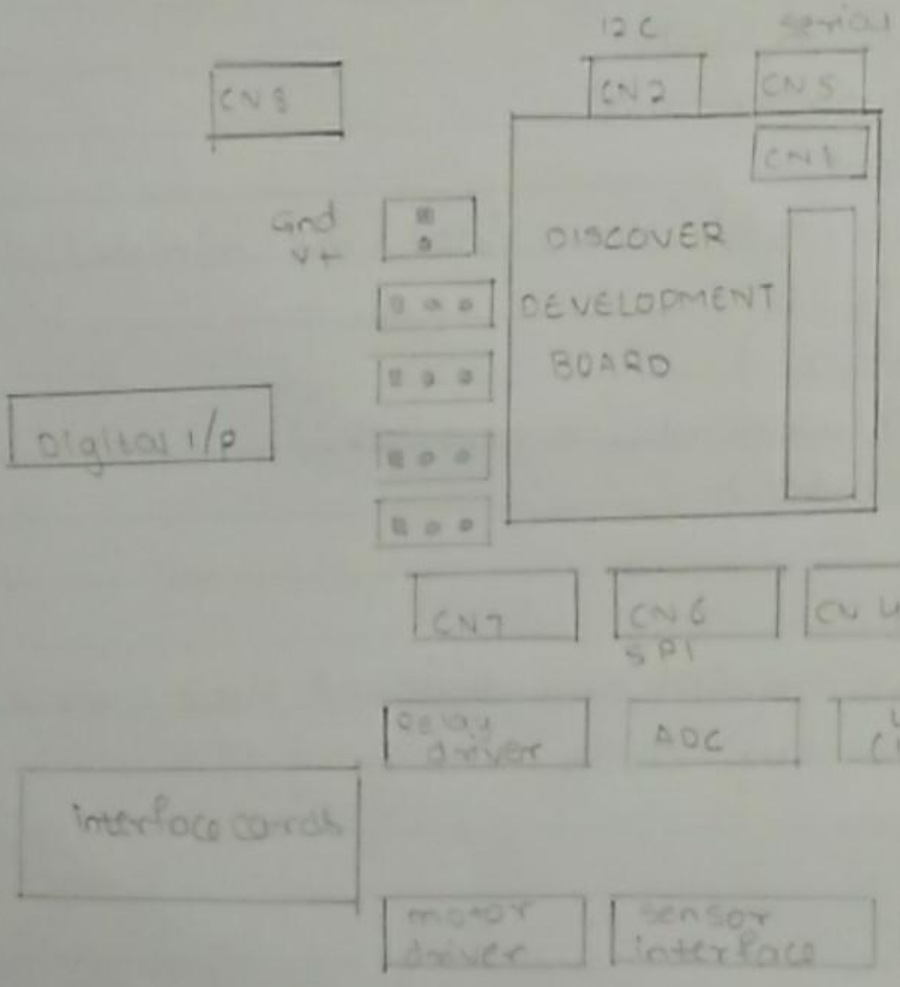
b) password : raspberry

step5 :- Create a new file with an extension '.py

Output :-

Interfacing diagram :-





Step6: Open the file with python 2IDLE only.

Step7: Type and run the program and see the output on terminal.

Code:
```
import time
import matplotlib.pylot as plt
from drawnow import*
import Adafruit_ADS1x15
adc = Adafruit_ADS1x15.ADS1115()
GAIN = 1
val = []
cnt = 0
plt.ion()
# Start continuous ADC conversions on channel 0 using
using the previous gain value. adc.start_adc(0,
gain = GAIN)
print('Reading ADS1x15 channel 0')
# Create the figure function
def makefig():
plt.ylim(-5000, 5000)
plt.title('VSIT_IOT LAB Osciloscope')
plt.grid(True)
plt.ylabel('ADC outputs')
plt.plot(val, 'ro-', label='channel 0')
plt.legend(loc="lower right')
while (True):
# Read the last ADC conversion value and print
it out
```

```
value = adc.get_last_result()
print('channel 0: {0}'.format(value))
# sleep for half a second.
 time.sleep(0.1)
val.append(int(value))
drawnow(make_fig)
plt.pause(.000001)
drawnow(make_fig) cnt = cnt+1
plt.pause if(cnt>50):
  val.pop(0)
```

Conclusion:

Thus raspberry pi based oil Oscilloscope has been studied and implemented.

EXPERIMENT: No.

Practical 5

Page No. 141
Date

AIM:- Control raspberry pi via telegram messenger

Required Components:-

Particular	Quantity
Raspberry pi 3	1
power supply 12V/2Amp	1
USB keyboard	1
USB mouse	1
micro SD card	1
Discover board.	1

Procedure:-

Step 1 :- Connection of discover development board.

Step 2 :- Connection of raspberry pi.

Step 3 :- Switch ON power supply.

Step 4 :- Login to raspberry pi terminal
a) username : pi

b) password : raspberry.

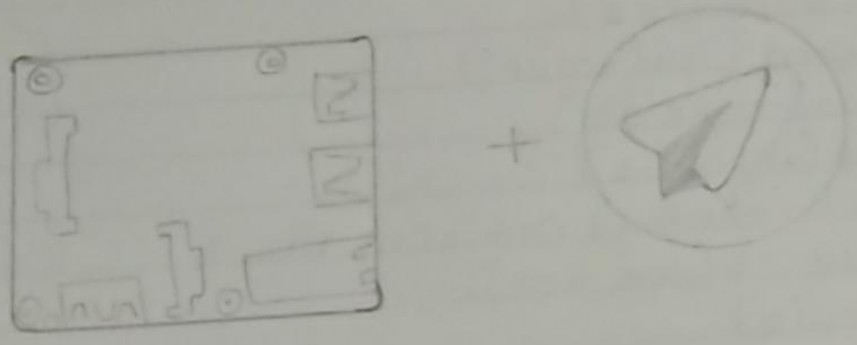
Step 5 :- Installing telegram app on your smart phone [only for Android Smartphone, WIFI or mobile data should be ON]

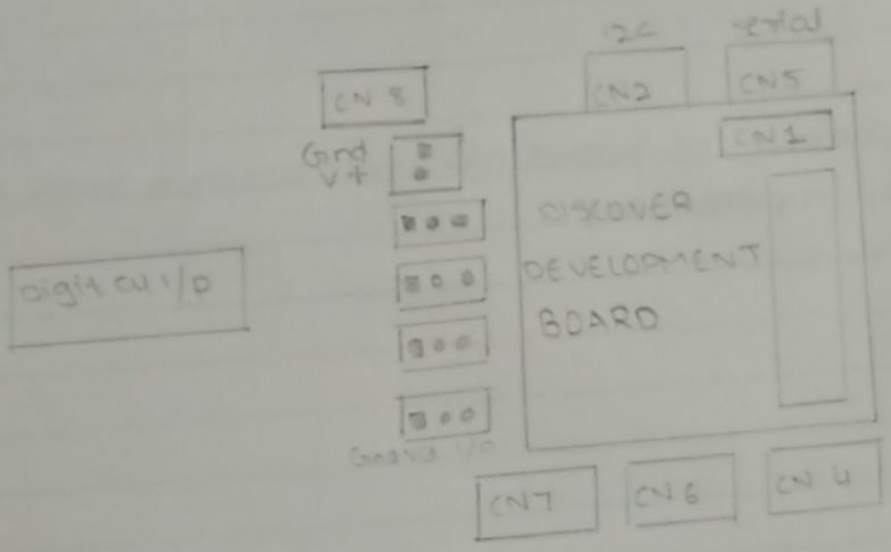
Open "playstore" app in your android mobile.

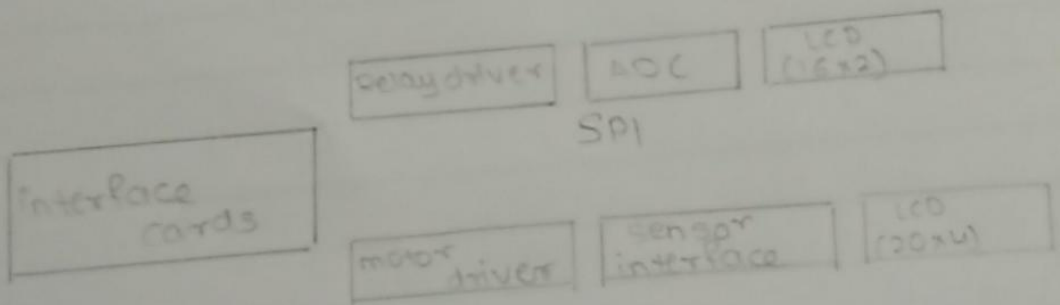
Teacher's Sign.:

Output:

Interfacing diagram:-







Search for "Telegram" in search Option.

Install the "Telegram" app by clicking on install button highlighted in green color.

Open the telegram app in your mobile.

Click on "START MESSAGING" button.

Enter your "mobile number" to register with telegram service.

step6: Now, Configure "Telegram App" for controlling raspberry pi.

step7: program raspberry pi with telegram TOKEN key to control devices connected across relays.

Code: import sys
import time
import random
import datetime
import telepot.
import RPi.GPIO as GPIO

RELAY1 = 20
RELAY2 = 16
FAN = RELAY1
LIGHT = RELAY2

```
GPIO.setwarnings(false)
#to use raspberry pi board pin numbers
GPIO.setmode(GPIO.BCM)
GPIO.cleanup()
# set up GPIO output channel
GPIO.setup(RELAY1, GPIO.OUT)
GPIO.setup(RELAY2, GPIO.OUT)
# your telegram token key variable
telegramBotToken='689381833:AAGBIOLW4rnUWB4nYlih2SB
2RHf3OvMKDqs'
# function to on and off devices
def on(pin):
GPIO.output(pin, GPIO.HIGH)
return "On"
def off(pin):
GPIO.output(pin, GPIO.LOW)
return "off"
def handle(msg):
chat_id = msg['chat']['id']
print str(chat_id)
Command = str(msg['text'])
print 'Receive message from Telegram:%s' % Command
if 'Fan' in Command or 'fan' in Command:
if 'On' in Command:
bot.send message(chat_id, str("Fan" + on(FAN)))
elif 'off' in Command:
bot.sendmessage(chat_id, str("Fan" + off(FAN)))
elif 'Light' in Command or 'Light' in command:
if 'On' in Command:
```

```
bot.sendmessage(chat_id, str("Light" + off(LIGHT)))
bot = telepot.bot(telegram bot Token)
bot.message_loop(handle)
print 'I am listening.....!

while 1:
time.sleep(10)
```

Conclusion:

Thus we have studied and implemented how to control raspberry pi via telegram messenger.

EXPERIMENT: No. Practical 6

Page No. 18
Date

AIM:- Setting up wireless Access point using raspberry pi.

Required Components:

Particular	Quantity
Raspberry Pi 3	1
power supply 12V/2Amp	1
USB Keyboard	1
USB Mouse	1
micro SD Card	1
WiFi USB dongle	1
Discover board	1
Ethernet cable	1

Procedure:-

step1: Install, update and upgrade raspbi

```
sudo apt-get update

sudo apt-get upgrade.
```

step 2: Install hostapd and dnsmasq

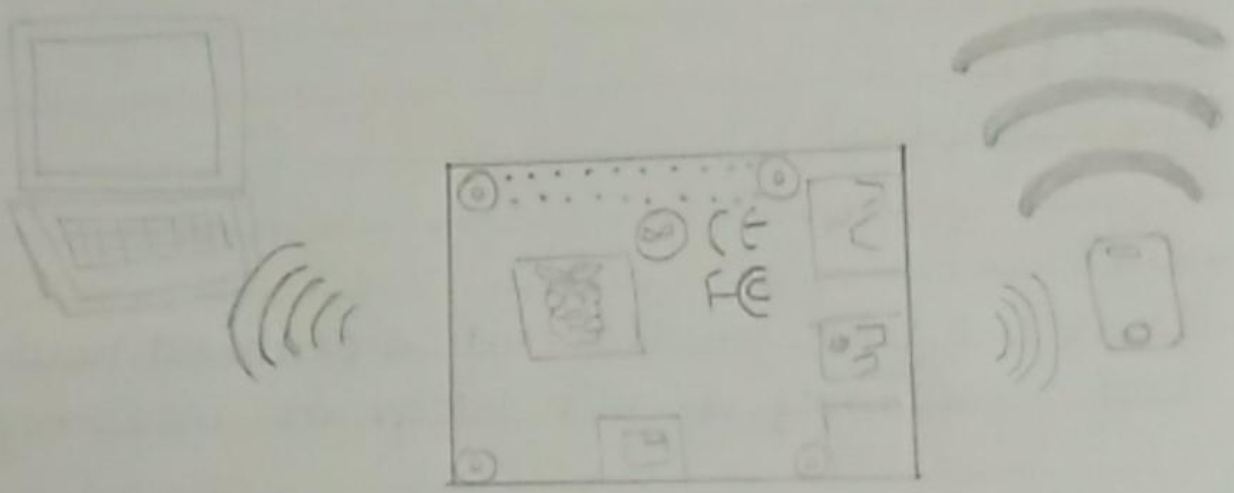
This is done using following commands:

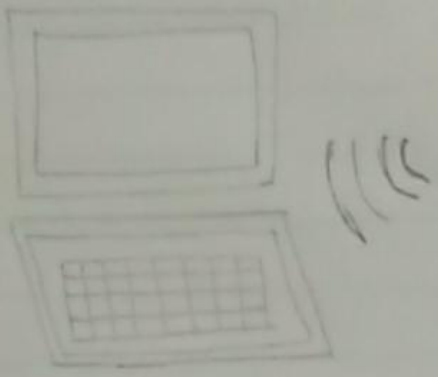
```
sudo apt-get install dnsmasq
sudo apt-get install hostapd.
```

hostapd is the package that let us Create a wireless hotspot using a r raspberry pi and dnsmasq is an easy-to-use DHCP and DNS server. to edit the program

Teacher's Sign.:

Output :





configuration files, disable them from running in the background.

This is done using following commands:

sudo systemctl stop dnsmasq

sudo systemctl stop hostapd.

step 3 :- Configure a static Ip for the wlan0 interface

To edit the configuration file, use the command:

sudo nano/etc/dhcpcd/.conf

At the bottom of config file add the following lines.

interface wlan0

static ip_address = 192.168.4.1/24

step 4 :- Configure the DHCP server (dnsmasq)

step 5 :- Configure the access point host software (hostapd)

/etc/hostapd/hostapd.conf

step 6 :- Set up traffic forwarding

step 7 : Add a new iptables rule

To add IP masquerading for outbound traffic an eth0 using iptables, run the following commands:

sudo iptables -t nat -A postrouting -o eth0 -j MASQUERADE

step 8 :- Enable internet connection

Teacher's Sign

Step 9:- Reboot

Conclusion:

Thus we have studied and implemented the step up of wireless Access point using raspberry pi.

EXPERIMENT No. Practical 7

Page No. 21
Date

AIM:- IOT based web controlled home automation using raspberry pi.

Required Components:-

Particular	Quality:
Raspberry Pi 3	1
Power supply 12v/2Amp	1
USB keyboard	1
USB mouse	1
micro SD card	1
Discover board	1
Relay board	1

Procedure:-

step 1 :- Connection of discover development board.

step 2 :- Connection of raspberry pi

step 3 :- Switch ON power supply

step 4 :- Login to raspberry pi terminal.

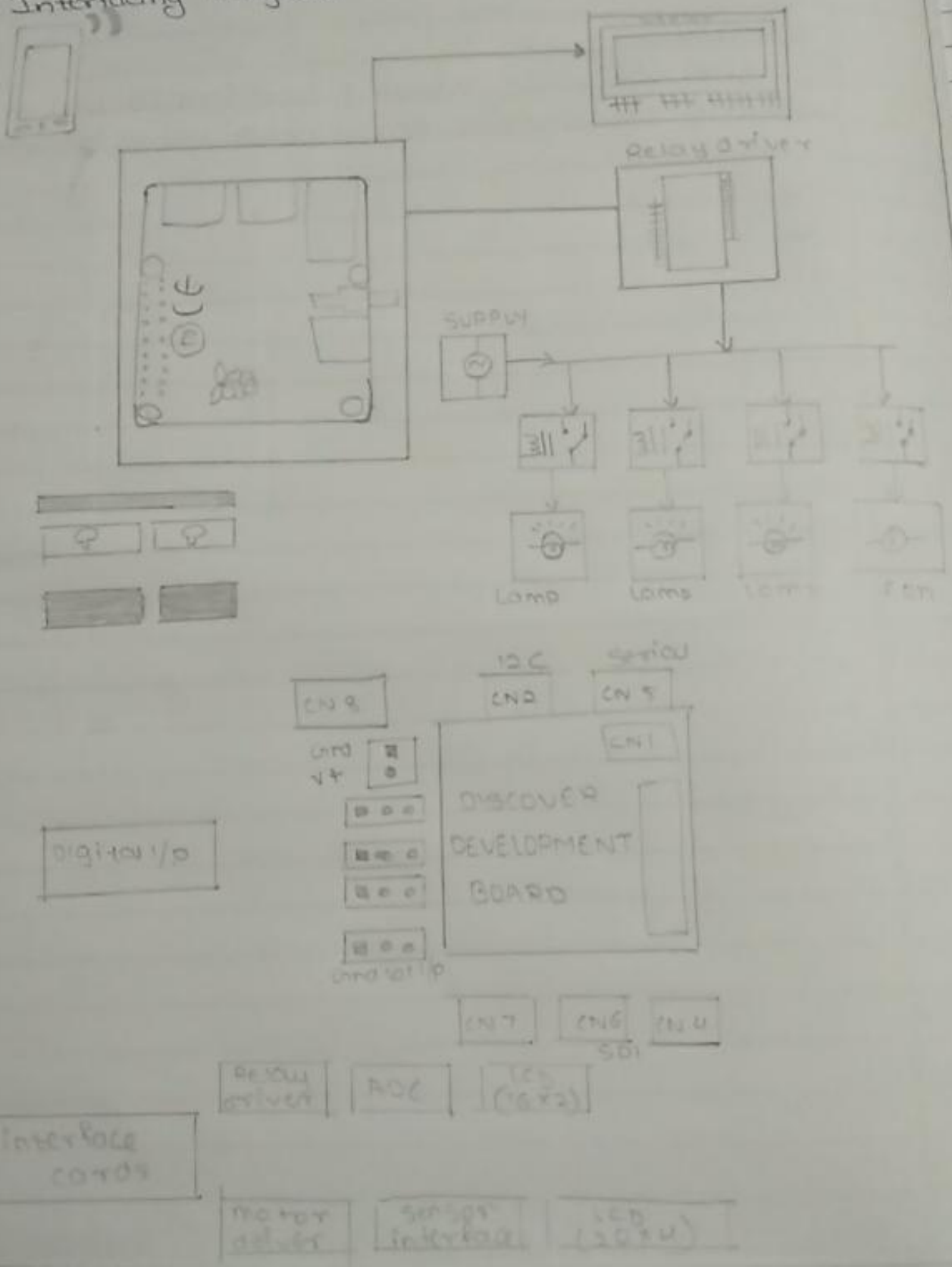
step 5 :- setup blynk app and Create button controls for fan and light as shown.

step 6 :- setup raspberry pi python code.

Teacher's Sign. :

Output :-

Interfacing diagram :-



EXPERIMENT No. Practical 8

Page No. 22
Date

AIM:- Visitor monitoring with raspberry pi and Pi camera.

Required Components

particular	Quantity
Raspberry pi 3	1
power supply 12V/2Amp	1
USB Key board	1
USB mouse	1
micro SD card	1
Discover board	1
Pi camera with CSI connector	1

Procedure:-

step1:- Connection of discover development board.

step 2:- Connection of raspberry pi.

step 3:- switch ON power supply

step4:- login to raspberry pi terminal.

a) username: pi

b) password: raspberry

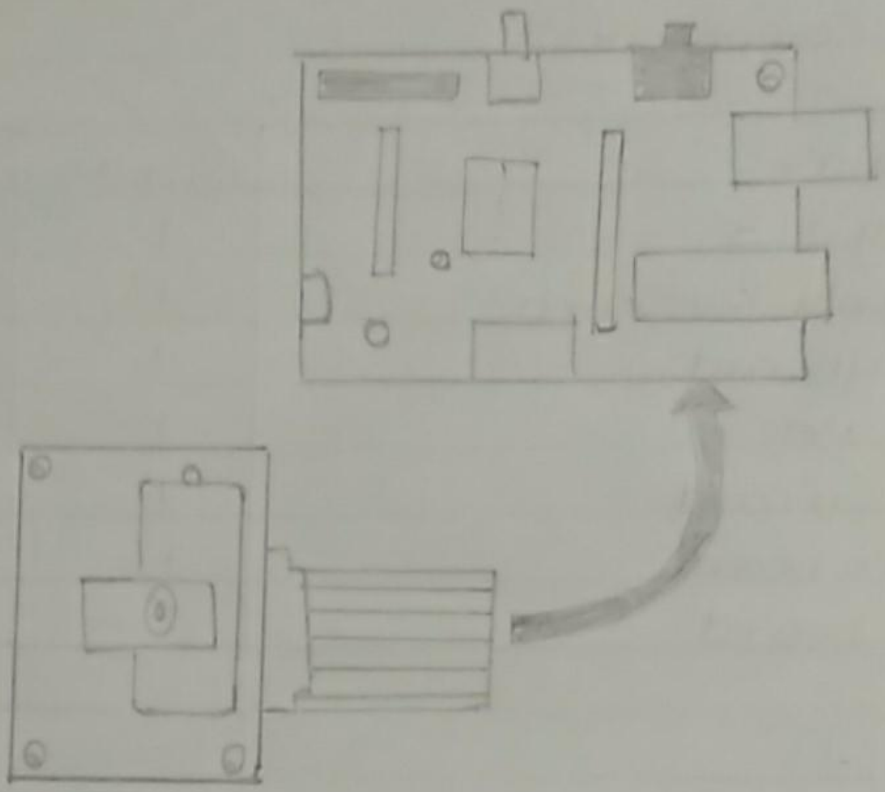
step5:- Install camera library

step6:- Create a new file with an extension .py

Teacher's Sign.

Output

Interfacing diagram:-



step7 :- Open the file with python 2 IDLE only

step8 :- Type and run the program and see the output on terminal.

Conclusion:-

Thus we have studied and implemented visitor monitoring with raspberry pi and pi camera.