DISPLAYING LED PATTERN

- Connect the basic Pi setup
- Connect the LED module to the following physical GPIO pins of the Pi :
- 29,31,33,35,36,37,38,40
- Connect the ground pin of the LED module to ground of Raspberry pi
- Power on the Raspberry Pi

```
import RPi.GPIO as GPIO

import time

GPIO.setmode(GPIO.BOARD)

led1=29

led2=31

led3=33

led4=35

led5=36

led6=37

led7=38

led8=40

GPIO.setup(led1,GPIO.OUT)

GPIO.setup(led2,GPIO.OUT)

GPIO.setup(led3,GPIO.OUT)

GPIO.setup(led4,GPIO.OUT)

GPIO.setup(led5,GPIO.OUT)

GPIO.setup(led6,GPIO.OUT)

GPIO.setup(led7,GPIO.OUT)

GPIO.setup(led8,GPIO.OUT)

GPIO.output(led1,False)

GPIO.output(led2,False)
```

```python
GPIO.output(led3,False)

GPIO.output(led4,False)

GPIO.output(led5,False)

GPIO.output(led6,False)

GPIO.output(led7,False)

GPIO.output(led8,False)

def ledpattern(ledVal1,ledVal2,ledVal3,ledVal4,ledVal5,ledVal6,ledVal7,ledVal8):

 GPIO.output(led1,ledVal1)

 GPIO.output(led2,ledVal2)

 GPIO.output(led3,ledVal3)

 GPIO.output(led4,ledVal4)

 GPIO.output(led5,ledVal5)

 GPIO.output(led6,ledVal6)

 GPIO.output(led7,ledVal7)

 GPIO.output(led8,ledVal8)

def patternOne():

 for i in range(0,3):

 ledpattern(1,0,1,0,1,0,1,0)

 time.sleep(1)

 ledpattern(0,1,0,1,0,1,0,1)

 time.sleep(1)

def patternTwo():

 for i in range(0,5):
```

8

```python
 ledpattern(1,0,0,0,0,0,0,0)

 time.sleep(0.1)
```

```python
    ledpattern(0,1,0,0,0,0,0,0)

    time.sleep(0.1)

    ledpattern(0,0,1,0,0,0,0,0)

    time.sleep(0.1)

    ledpattern(0,0,0,1,0,0,0,0)

    time.sleep(0.1)

    ledpattern(0,0,0,0,1,0,0,0)

    time.sleep(0.1)

    ledpattern(0,0,0,0,0,1,0,0)

    time.sleep(0.1)

    ledpattern(0,0,0,0,0,0,1,0)

    time.sleep(0.1)

    ledpattern(0,0,0,0,0,0,0,1)

    time.sleep(0.1)


def patternThree():

    for i in range(0,5):

    ledpattern(0,0,0,0,0,0,0,1)

    time.sleep(0.1)

    ledpattern(0,0,0,0,0,0,1,0)

    time.sleep(0.1)

    ledpattern(0,0,0,0,0,1,0,0)

    time.sleep(0.1)

    ledpattern(0,0,0,0,1,0,0,0)

    time.sleep(0.1)

    ledpattern(0,0,0,1,0,0,0,0)
```

```python
    time.sleep(0.1)

    ledpattern(0,0,1,0,0,0,0,0)

    time.sleep(0.1)

    ledpattern(0,1,0,0,0,0,0,0)

    time.sleep(0.1)

    ledpattern(1,0,0,0,0,0,0,0)

    time.sleep(0.1)


def patternFour():

  for i in range(0,5):

  ledpattern(0,1,1,1,1,1,1,1)

  time.sleep(0.1)

  ledpattern(1,0,1,1,1,1,1,1)

  time.sleep(0.1)

  ledpattern(1,1,0,1,1,1,1,1)

  time.sleep(0.1)

  ledpattern(1,1,1,0,1,1,1,1)

  time.sleep(0.1)

  ledpattern(1,1,1,1,0,1,1,1)

  time.sleep(0.1)

  ledpattern(1,1,1,1,1,0,1,1)

  time.sleep(0.1)

  ledpattern(1,1,1,1,1,1,0,1)

  time.sleep(0.1)

  ledpattern(1,1,1,1,1,1,1,0)
```

9

```python
    time.sleep(0.1)
def patternFive():
 for i in range(0,5):
 ledpattern(1,1,1,1,1,1,1,0)
 time.sleep(0.1)
 ledpattern(1,1,1,1,1,1,0,1)
 time.sleep(0.1)
 ledpattern(1,1,1,1,1,0,1,1)
 time.sleep(0.1)
 ledpattern(1,1,1,1,0,1,1,1)
 time.sleep(0.1)
 ledpattern(1,1,1,0,1,1,1,1)
 time.sleep(0.1)
 ledpattern(1,1,0,1,1,1,1,1)
 time.sleep(0.1)
 ledpattern(1,0,1,1,1,1,1,1)
 time.sleep(0.1)
 ledpattern(0,1,1,1,1,1,1,1)
 time.sleep(0.1)


try:
 while True:
 patternOne()
 patternTwo()
 patternThree()
 patternFour()
```

```
    patternFive()
finally:
 GPIO.cleanup()
```

DISPLAY TIME OVER 4 DIGIT 7 SEGMENT DISPLAY

1. Connect the Pin2(5V) of Rpi to Vcc pin of module.

2. Connect Pin 6 (GND) of Rpi to GND of module

3. Connect Pin38 (GPIO20) of Rpi to DIO of module

4. Lastly connect Pin 40 (GPIO21) of Rpi to CLK of module.

https://github.com/timwaizenegger/raspberrypi-examples/tree/master/actor-led-7segment-4numbers

```
from time import sleep
import tm1637
try:
 import thread
except ImportError:
 import _thread as thread
# Initialize the clock (GND, VCC=3.3V, Example Pins are DIO-20 and CLK21)
Display = tm1637.TM1637(CLK=21, DIO=20, brightness=1.0)
try:
 print "Starting clock in the background (press CTRL + C to stop):"
 Display.StartClock(military_time=True)
 Display.SetBrightness(1.0)
 while True:
 Display.ShowDoublepoint(True)
```

```
 sleep(1)

 Display.ShowDoublepoint(False)

 sleep(1)


 Display.StopClock()

 thread.interrupt_main()

except KeyboardInterrupt:

 print "Properly closing the clock and open GPIO pins"

 Display.cleanup()
```

CAPTURING IMAGE WITH RASPBERRY PI AND PI CAMERA

For enabling camera in Raspberry Pi, open raspberry pi configuration using following command,

<mark>sudo raspi-config</mark>

then select Interfacing options in which select camera option to enable its functionality. Reboot

Raspberry Pi.

You can capture an image by just typing a single line command. Open terminal window and type the

command as follows:

<mark>$ sudo raspistill –o /home/pi/Desktop/image.jpg</mark>


```
#import time and picamera library

import picamera

from time import sleep

#create object for PiCamera class

camera = picamera.PiCamera()
```

```python
#set resolution
camera.resolution = (1280, 720)
camera.brightness = 60
camera.start_preview()
#Camera warmup time
sleep(2)
#store image
camera.capture('/home/pi/Pictures/image1.jpeg')
camera.stop_preview()
```

Python Program for Video Recording

```python
import picamera
from time import sleep
camera = picamera.PiCamera()
camera.resolution = (1280, 720)
camera.start_preview()
#start recording using pi camera
camera.start_recording("/home/pi/demo.h264")
sleep(20)
```

49

```python
#stop recording
camera.stop_recording()
camera.stop_preview()
```

OSCILLOSCOPE

VDD – 3.3V (Pin 1)

GND – GND

SDA – SDA (Pin 3)

SCL – SCL (Pin 5)

sudo raspi-config

When the configuration panels open, select interface options, select I2C and click enable.

Step 2: Update the Raspberry pi

sudo apt-get update

sudo apt-get upgrade

Step 3: Install the Adafruit ADS1115 library for ADC

cd ~

sudo apt-get install build-essential python-dev python-smbus git

git clone https://github.com/adafruit/Adafruit_Python_ADS1x15.git

cd Adafruit_Python_ADS1x1z

sudo python setup.py install

import time

import matplotlib.pyplot as plt

from drawnow import *

import Adafruit_ADS1x15

adc=Adafruit_ADS1x15.ADS1115( )

```python
GAIN=1

val=[]

cnt=0

plt.ion()

adc.start_adc(0,gain=GAIN)

print('Reading ADS1x15 channel 0')

def makeFig():

 plt.ylim(-5000,5000)

 plt.title('Oscilloscope')

 plt.grid(True);

 plt.ylabel('ADC outputs')

 plt.plot(val,'ro-',label='Channel 0')

 plt.legend(loc='lower right')

while (True):

 value=adc.get_last_result()

 print('Channel 0: {0}'.format(value))

 time.sleep(0.5)
```

22

```python
 val.append(int(value))

 drawnow(makeFig)

 plt.pause(.000001)

 cnt=cnt+1

 if(cnt>50):

 val.pop(0)
```

Telegram BOT

a) The next step would be to request the Bot Father to create us a new Bot.

b) On the top right corner of the Home screen there will be a search icon, click on it to search for the name "botfather".

c) botfather is a Bot(wizard) which will guide you to create a new bot for you.

d) Click on start and select /newbot as shown in the picture below. Now, the bot will ask for few details like name of your Bot and the user name of the bot.

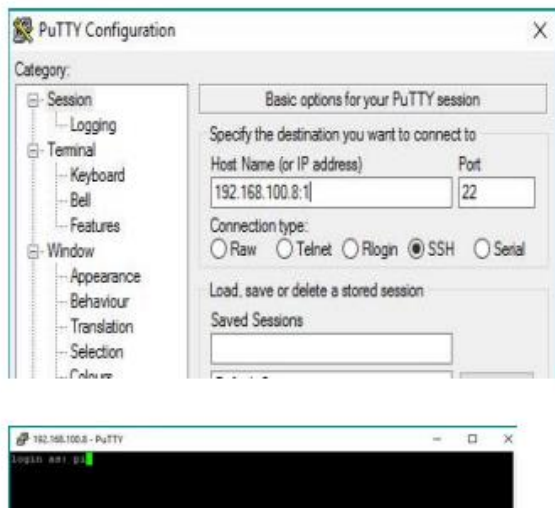e) Fill those details and remember the username as we will needing it in future.

Step3: Getting your token for access

Step 4: Setup Raspberry Pi

Step 5:Install TelegramBot on Raspberry Pi

Step 6: Open Putty

5.2 Connect Pi via SSH

sudo apt-get install python-pip

sudo pip install teleport

import sys
import time
import telepot

```python
import RPi.GPIO as GPIO
#LED
def on(pin):
 GPIO.output(pin,GPIO.HIGH)
 return
def off(pin):
 GPIO.output(pin,GPIO.LOW)
 return
# to use Raspberry Pi board pin numbers
GPIO.setmode(GPIO.BOARD)
# set up GPIO output channel
GPIO.setup(11, GPIO.OUT)
def handle(msg):
 chat_id = msg['chat']['id']
 command = msg['text']
 print('Got command: %s' % command)
28
 if command == 'on':
 bot.sendMessage(chat_id, on(11))
 elif command =='off':
bot.sendMessage(chat_id, off(11))
bot = telepot.Bot('Bot Token')
bot.message_loop(handle)
print('I am listening...')
while 1:
 try:
 time.sleep(10)
 except KeyboardInterrupt:
print('\n Program interrupted')
GPIO.cleanup()
exit()
 except:
print('Other error or exception occured!')
GPIO.cleanup()
```

7.2 Paste your Bot Token here

```python
bot = telepot.Bot('Bot Token')
```

7.3 Run the Code

```python
python telegrambot.py
```

All set, now time to connect the Pi and LED.