**Blink LED**

import RPi.GPIO as GPIO

from time import sleep

GPIO.setmode(GPIO.BOARD)

ledPin = 15 #board pin

GPIO.setup(ledPin, GPIO.OUT)

GPIO.output(ledPin, False)

try:

while True:

GPIO.output(ledPin, True)

print("Led On")

sleep(1)

GPIO.output(ledPin, False)

print("Led Off")

sleep(1)

finally:

GPIO.output(ledPin, False)

GPIO.cleanup()

**Pattern LED**

import RPi.GPIO as GPIO

from time import sleep

GPIO.setmode(GPIO.BOARD)

ledPin1 = 40

ledPin2 = 38

ledPin3 = 36

GPIO.setup(ledPin1, GPIO.OUT)

GPIO.setup(ledPin2, GPIO.OUT)

GPIO.setup(ledPin3, GPIO.OUT)

GPIO.output(ledPin1, False)

GPIO.output(ledPin2, False)

GPIO.output(ledPin3, False)

def ledPattern(ledVal1, ledVal2, ledVal3):

GPIO.output(ledPin1, ledVal1)

GPIO.output(ledPin2, ledVal2)

GPIO.output(ledPin3, ledVal3)

def pattern1():

for i in range(0, 3):

ledPattern(1,0,1)

sleep(1)

ledPattern(0,1,0)

sleep(1)

def pattern2():

for i in range(0, 5):

ledPattern(1,0,0)

sleep(1)

ledPattern(0, 1, 1)

sleep(1)

try:

while True:

pattern1()

pattern2()

finally:

GPIO.cleanup()

**Seven Segment:**

#!/usr/bin/env python

# -\*- coding: utf-8 -\*-

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)

# CLK -> 21 (GPIO) -> BOARD(40)

# DIO -> 20 (GPIO) -> BOARD(38)

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=False)

print ('Continue Python script and tweak Display!')

sleep(5)

Display.ShowDoublepoint(False)

sleep(5)

loops = 3

while loops > 0:

for i in range(0, 10):

Display.SetBrightness(i / 10.0)

sleep(0.5)

loops -= 1

Display.StopClock()

thread.interrupt\_main()

except KeyboardInterrupt:

print ("Properly closing the clock and open GPIO pins")

Display.cleanup()

**GPS:**

'''

GPS Interfacing with Raspberry Pi using Pyhton

http://www.electronicwings.com

'''

import serial #import serial pacakge

from time import sleep

import webbrowser #import package for opening link in browser

import sys #import system package

def GPS\_Info():

global NMEA\_buff

global lat\_in\_degrees

global long\_in\_degrees

nmea\_time = []

nmea\_latitude = []

nmea\_longitude = []

nmea\_time = NMEA\_buff[0] #extract time from GPGGA string

nmea\_latitude = NMEA\_buff[1] #extract latitude from GPGGA string

nmea\_longitude = NMEA\_buff[3] #extract longitude from GPGGA string

print("NMEA Time: ", nmea\_time,'\n')

print ("NMEA Latitude:", nmea\_latitude,"NMEA Longitude:", nmea\_longitude,'\n')

lat = float(nmea\_latitude) #convert string into float for calculation

longi = float(nmea\_longitude) #convertr string into float for calculation

lat\_in\_degrees = convert\_to\_degrees(lat) #get latitude in degree decimal format

long\_in\_degrees = convert\_to\_degrees(longi) #get longitude in degree decimal format

#convert raw NMEA string into degree decimal format

def convert\_to\_degrees(raw\_value):

decimal\_value = raw\_value/100.00

degrees = int(decimal\_value)

mm\_mmmm = (decimal\_value - int(decimal\_value))/0.6

position = degrees + mm\_mmmm

position = "%.4f" %(position)

return position

gpgga\_info = "$GPGGA,"

ser = serial.Serial ("/dev/ttyUSB0") #Open port with baud rate

GPGGA\_buffer = 0

NMEA\_buff = 0

lat\_in\_degrees = 0

long\_in\_degrees = 0

try:

while True:

received\_data = (str)(ser.readline()) #read NMEA string received

GPGGA\_data\_available = received\_data.find(gpgga\_info) #check for NMEA GPGGA string

if (GPGGA\_data\_available>0):

GPGGA\_buffer = received\_data.split("$GPGGA,",1)[1] #store data coming after "$GPGGA," string

NMEA\_buff = (GPGGA\_buffer.split(',')) #store comma separated data in buffer

GPS\_Info() #get time, latitude, longitude

print("lat in degrees:", lat\_in\_degrees," long in degree: ", long\_in\_degrees, '\n')

map\_link = 'http://maps.google.com/?q=' + lat\_in\_degrees + ',' + long\_in\_degrees #create link to plot location on Google map

print("<<<<<<<<press ctrl+c to plot location on google maps>>>>>>\n") #press ctrl+c to plot on map and exit

print("------------------------------------------------------------\n")

except KeyboardInterrupt:

webbrowser.open(map\_link) #open current position information in google map

sys.exit(0)

'''

ls /dev/ttyUSB\*

ls /dev/ttyUSB0

sudo cat /dev/ttyUSB\*

press ctrl+c

sudo apt-get update

sudo apt-get install gpsd gpsd-clients python-gps

sudo gpsd /dev/ttUSB0 -F /var/run/gpsd.sock

cgps -s

'''

**RFID:**

import RPi.GPIO as GPIO

import time

import serial

GPIO.setmode(GPIO.BOARD)

greenLED=37

redLED=35

buzzer=33

GPIO.setup(greenLED,GPIO.OUT)

GPIO.setup(redLED,GPIO.OUT)

GPIO.setup(buzzer,GPIO.OUT)

GPIO.output(greenLED, False)

GPIO.output(redLED, False)

GPIO.output(buzzer, True)

time.sleep(0.1)

GPIO.output(buzzer, False)

time.sleep(0.1)

GPIO.output(buzzer, True)

time.sleep(0.1)

GPIO.output(buzzer, False)

time.sleep(0.1)

def read\_rfid():

ser=serial.Serial("dev/ttyUSB0")

ser.baudrate=9600

data=ser.read(12)

ser.close()

return data

try:

while True:

id= read\_rfid()

print(id)

if id=="400034E165F0":

print("Access granted")

GPIO.output(greenLED, True)

GPIO.output(redLED, False)

GPIO.output(buzzer, False)

time.sleep(2)

else:

print("Access denied")

GPIO.output(greenLED, False)

GPIO.output(redLED, True)

GPIO.output(buzzer, True)

time.sleep(2)

GPIO.output(greenLED, False)

GPIO.output(redLED, False)

GPIO.output(buzzer, False)

finally:

GPIO.cleanup()

**Camera:**

#import timport ime and picamera library

from time import sleep

from picamera import PiCamera

camera=PiCamera()

camera.resolution=(1080,1920)

camera.start\_preview()

sleep(10)

camera.capture('/home/pi/Pictures/random.jpg')

camera.stop\_preview()

**TELEGRAM:**

import sys

import time

import random

import datetime

import telepot

import RPi.GPIO as GPIO

ledPin = 38

GPIO.setmode(GPIO.BOARD)

GPIO.setup(ledPin, GPIO.OUT)

def on(pin):

GPIO.output(pin, True)

return

def off(pin):

GPIO.output(pin, False)

return

def handle(message):

chat\_id = message['chat']['id']

command = message['text']

print("Got command " + command)

if command == "on":

bot.sendMessage(chat\_id, on(ledPin))

elif command == "off":

bot.sendMessage(chat\_id, off(ledPin))

bot = telepot.Bot('Enter your bot token')

bot.message\_loop(handle)

print("I am listening............")

while True:

time.sleep(10)

#don't run here, run in cmd

#steps: sudo pip install telepot

#python (file path)