**Assignment -3**

Python Programming

|  |  |
| --- | --- |
| Assignment Date | 08 October 2022 |
| Student Name | Akila G |
| Student Roll Number | 953619104003 |
| Maximum Marks | 2 Marks |

**Question:**

Write a python code for blinking LED and Traffic Lights for Raspberry pi

**Solution:**

// Code for Blinking LED

import time

import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BOARD)

led = 5 # GPIO pin number is 5 and name is GPIO3

GPIO.setup(led, GPIO.OUT, initial = 0) # Setup LED and set it initially to OFF

while(True):

GPIO.output(led, GPIO.HIGH) # LED ON

time.sleep(2) # Wait for 2 second

GPIO.output(led, GPIO.LOW) # LED OFF

time.sleep(2)

// Code for Traffic Light

import RPi.GPIO as GPIO

import time

GPIO.setmode(GPIO.BOARD)

YellowLed = 35

RedLed = 33

GreenLed = 37

safeCrossing = 38

button = 11

GPIO.setup(RedLed, GPIO.OUT)

GPIO.setup(YellowLed, GPIO.OUT)

GPIO.setup(GreenLed, GPIO.OUT)

GPIO.setup(safeCrossing, GPIO.OUT)

GPIO.output(RedLed, GPIO.HIGH)

GPIO.output(YellowLed, GPIO.HIGH)

GPIO.output(GreenLed, GPIO.HIGH)

GPIO.setup(button,GPIO.IN,pull\_up\_down=GPIO.PUD\_DOWN)

def cycleLights ():

print ('Traffic: GREEN off, AMBER on')

GPIO.output(GreenLed, GPIO.HIGH)

GPIO.output(YellowLed, GPIO.LOW)

time.sleep(1)

print ('Traffic: AMBER off, RED on')

GPIO.output(YellowLed, GPIO.HIGH)

GPIO.output(RedLed, GPIO.LOW)

time.sleep(1)

print ('Padestrian: Safe to cross on')

GPIO.output(safeCrossing, GPIO.LOW)

time.sleep(5)

print ('Padestrian: Safe to cross flashing')

for flash in range(0, 5):

GPIO.output(safeCrossing, GPIO.HIGH)

time.sleep(0.8)

GPIO.output(safeCrossing, GPIO.LOW)

time.sleep(0.8)

print ('Padestrian: Safe to cross off')

GPIO.output(safeCrossing, GPIO.HIGH)

time.sleep(1)

print ('Traffic: AMBER and RED on')

GPIO.output(YellowLed, GPIO.LOW)

time.sleep(1.5)

print ('Traffic: AMBER and RED off, GREEN on')

GPIO.output(RedLed, GPIO.HIGH)

GPIO.output(YellowLed, GPIO.HIGH)

GPIO.output(GreenLed, GPIO.LOW)

print ('Padestrian button blocked to let traffic flow')

time.sleep(4)

print ('Padestrian button unblocked')

return

def teardown ():

GPIO.output(RedLed, GPIO.HIGH)

GPIO.output(YellowLed, GPIO.HIGH)

GPIO.output(GreenLed, GPIO.HIGH)

GPIO.cleanup()

return

try:

while True:

ButtonPress = False

# Lights start with the green traffic light on

# and the padestrian light off

GPIO.output(GreenLed, GPIO.LOW)

GPIO.output(safeCrossing, GPIO.HIGH)

# Wait until button is presses

print ('Waiting for a padestrian to press the button', end='')

while not ButtonPress:

# Check every 2 seconds for a press

print ('.', end='')

time.sleep(1)

ButtonPress = GPIO.input(button)

print ('\nPadestrian button press detected!')

cycleLights()

except KeyboardInterrupt:

teardown()