

## مؤشر حالة العمل باستخدام الراسبيري باي

في هذا الدرس سنساعد الأفراد الذين يعملون أو يدرسون عن بعد، في توفير بيئة هادئة خلال أوقات انشغالهم بدون أن يقاطعهم أحد، وذلك باستخدام الراسبيري باي وشاشة كرسالية يتم تعليقها على الباب الغرفة تعمل كأنها مؤشر حالة العمل ، ويظهر عدة حالات اما في اجتماع مشغول متاح وغيرها و يتم التغير بين الحالات باستخدام صفحة ويب تحتوي على عدة خيارات بمجرد النقر على المفتاح يتم طباعة الحالة حتى يتمكن الآخرين من معرفة الانشغال وعدم مقاطعتك.



### المواد والأدوات



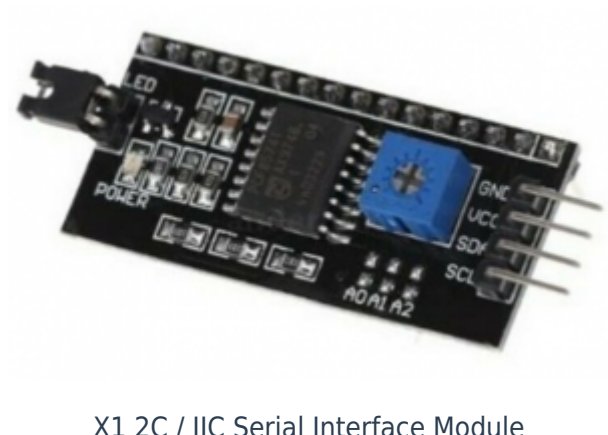
X1 راسبيري باي



X1 ذاكرة 8 قيقا



X1 شاشة كرسالية



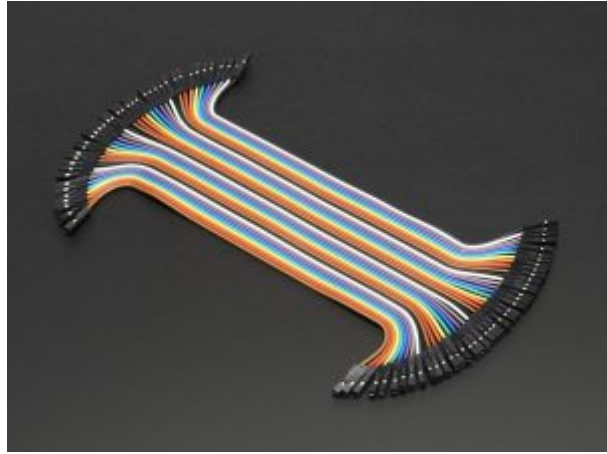
X1 2C / IIC Serial Interface Module



X1 لوحة تجارب



1X محول طاقة 2A5V



X4 أسلاك توصيل F/F

توصيل الدائرة



## الطباعة على الشاشة

أولاً، سوف تحتاج إلى تثبيت نظام الراسبيان على الراسبيري باي إذا لم تكن قد فعلت ذلك قم بالإطلاع على نظام تشغيل الراسبيري باي وبعد تثبيت النظام، يمكنك تهيئة النظام من خلال الرجوع تهيئة نظام التشغيل و يمكنك التحكم بالراسبيري باي عن بعد من خلال الدرس

بالبداية سنقوم بتجربة الطباعة على الشاشة حتى نتحقق من ضبط المقاومة المتغيرة الموجودة على موديول I2C

ابدأ بتحديث النظام من خلال الأوامر التالية التي يتم كتابتها على الشاشة السوداء LXterminal

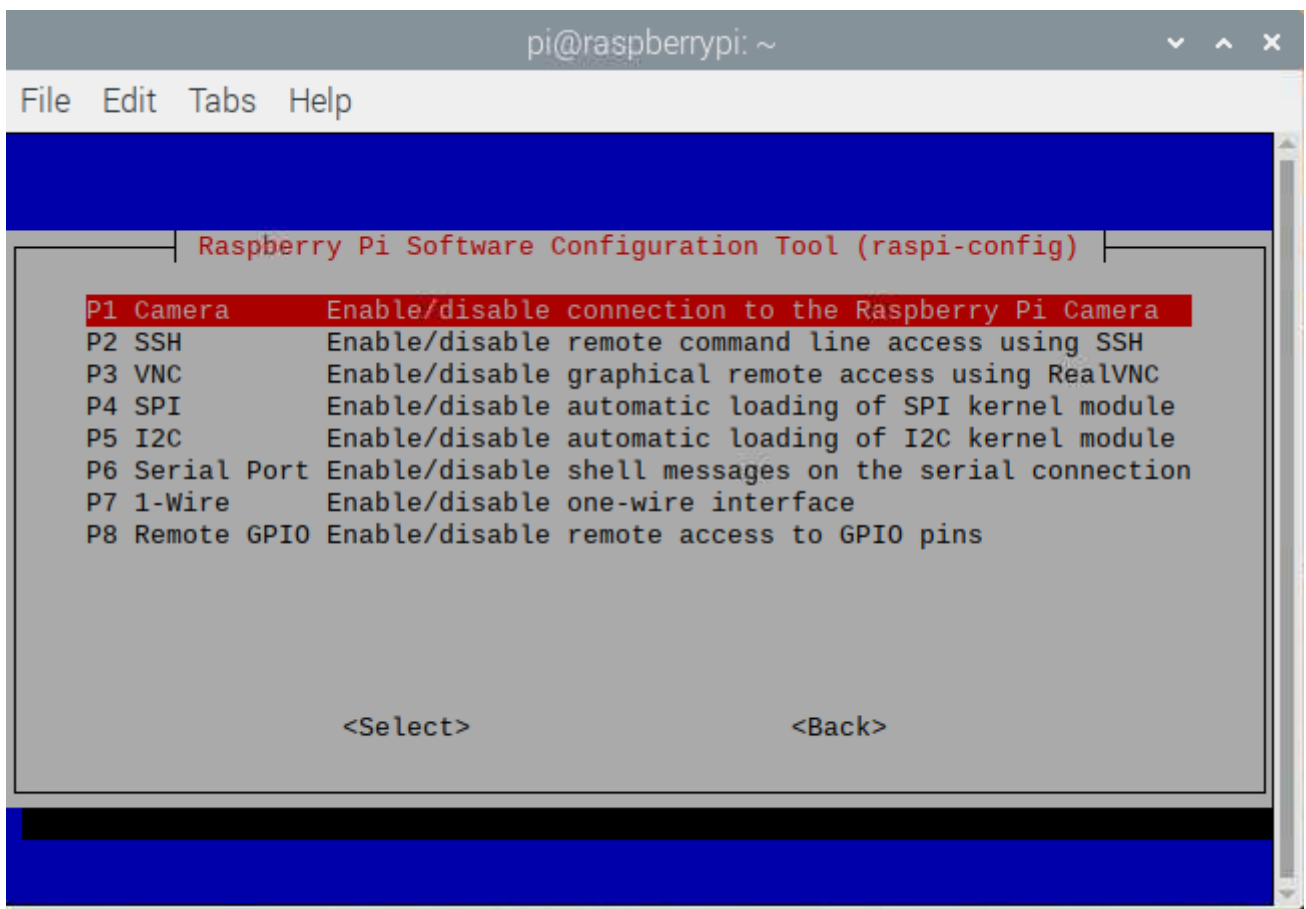
```
sudo apt-get update  
sudo apt-get upgrade
```

قم بتفعيل اتصال i2c من خلال الأمر التالي

```
sudo raspi-config
```



قم بتحريك الأسهم للوصول الى الخيار الثالث "Interface Options"



مرر الأسهم للوصول للخيار الخامس "P5 I2C" و انقر على enter



في البداية سنقوم باعداد ملف خاص بمشروعنا و ذلك من خلال كتابة الأمر التالي

```
mkdir work
```

داخل ملف المشروع سنضيف المكتبة الخاصة بمديول i2c بكتابة الأمر التالي

```
cd work
```

ثم

```
sudo nano i2c_lcd_driver.py
```

قم بنسخ الكود التالي

```
#Caroline Dunn
#this code was modified from:
#https://www.circuitbasics.com/raspberry-pi-i2c-lcd-set-up-and-programming/

# i2c bus (0 -- original Pi, 1 -- Rev 2 Pi)
I2CBUS = 1

# LCD Address
ADDRESS = 0x27

import smbus
```

```

from time import sleep

class i2c_device:
def __init__(self, addr, port=I2CBUS):
self.addr = addr
self.bus = smbus.SMBus(port)

# Write a single command
def write_cmd(self, cmd):
self.bus.write_byte(self.addr, cmd)
sleep(0.0001)

# Write a command and argument
def write_cmd_arg(self, cmd, data):
self.bus.write_byte_data(self.addr, cmd, data)
sleep(0.0001)

# Write a block of data
def write_block_data(self, cmd, data):
self.bus.write_block_data(self.addr, cmd, data)
sleep(0.0001)

# Read a single byte
def read(self):
return self.bus.read_byte(self.addr)

# Read
def read_data(self, cmd):
return self.bus.read_byte_data(self.addr, cmd)

# Read a block of data
def read_block_data(self, cmd):
return self.bus.read_block_data(self.addr, cmd)

# commands
LCD_CLEARDISPLAY = 0x01
LCD_RETURNHOME = 0x02
LCD_ENTRYMODESET = 0x04
LCD_DISPLAYCONTROL = 0x08
LCD_CURSORSHIFT = 0x10
LCD_FUNCTIONSET = 0x20
LCD_SETCGRAMADDR = 0x40
LCD_SETDDRAMADDR = 0x80

# flags for display entry mode
LCD_ENTRYRIGHT = 0x00
LCD_ENTRYLEFT = 0x02
LCD_ENTRYSHIFTINCREMENT = 0x01
LCD_ENTRYSHIFTDECREMENT = 0x00

# flags for display on/off control
LCD_DISPLAYON = 0x04

```

```

LCD_DISPLAYOFF = 0x00
LCD_CURSORON = 0x02
LCD_CURSOROFF = 0x00
LCD_BLINKON = 0x01
LCD_BLINKOFF = 0x00

# flags for display/cursor shift
LCD_DISPLAYMOVE = 0x08
LCD_CURSORMOVE = 0x00
LCD_MOVERIGHT = 0x04
LCD_MOVELEFT = 0x00

# flags for function set
LCD_8BITMODE = 0x10
LCD_4BITMODE = 0x00
LCD_2LINE = 0x08
LCD_1LINE = 0x00
LCD_5x10DOTS = 0x04
LCD_5x8DOTS = 0x00

# flags for backlight control
LCD_BACKLIGHT = 0x08
LCD_NOBACKLIGHT = 0x00

En = 0b00000100 # Enable bit
Rw = 0b00000010 # Read/Write bit
Rs = 0b00000001 # Register select bit

class lcd:
#initializes objects and lcd
def __init__(self):
self.lcd_device = i2c_device(ADDRESS)

self.lcd_write(0x03)
self.lcd_write(0x03)
self.lcd_write(0x03)
self.lcd_write(0x02)

self.lcd_write(LCD_FUNCTIONSET | LCD_2LINE | LCD_5x8DOTS | LCD_4BITMODE)
self.lcd_write(LCD_DISPLAYCONTROL | LCD_DISPLAYON)
self.lcd_write(LCD_CLEARDISPLAY)
self.lcd_write(LCD_ENTRYMODESET | LCD_ENTRYLEFT)
sleep(0.2)

# clocks EN to latch command
def lcd_strobe(self, data):
self.lcd_device.write_cmd(data | En | LCD_BACKLIGHT)
sleep(.0005)
self.lcd_device.write_cmd(((data & ~En) | LCD_BACKLIGHT))
sleep(.0001)

def lcd_write_four_bits(self, data):

```



```

self.lcd_device.write_cmd(data | LCD_BACKLIGHT)
self.lcd_strobe(data)

# write a command to lcd
def lcd_write(self, cmd, mode=0):
self.lcd_write_four_bits(mode | (cmd & 0xF0))
self.lcd_write_four_bits(mode | ((cmd << 4) & 0xF0))

# write a character to lcd (or character rom) 0x09: backlight | RS=DR<
# works!
def lcd_write_char(self, charvalue, mode=1):
self.lcd_write_four_bits(mode | (charvalue & 0xF0))
self.lcd_write_four_bits(mode | ((charvalue << 4) & 0xF0))

# put string function with optional char positioning
def lcd_display_string(self, string, line=1, pos=0):
if line == 1:
pos_new = pos
elif line == 2:
pos_new = 0x40 + pos
elif line == 3:
pos_new = 0x14 + pos
elif line == 4:
pos_new = 0x54 + pos

self.lcd_write(0x80 + pos_new)

for char in string:
self.lcd_write(ord(char), Rs)

# clear lcd and set to home
def lcd_clear(self):
self.lcd_write(LCD_CLEARDISPLAY)
self.lcd_write(LCD_RETURNHOME)

# define backlight on/off (lcd.backlight(1); off= lcd.backlight(0))
def backlight(self, state): # for state, 1 = on, 0 = off
if state == 1:
self.lcd_device.write_cmd(LCD_BACKLIGHT)
elif state == 0:
self.lcd_device.write_cmd(LCD_NOBACKLIGHT)

# add custom characters (0 - 7)
def lcd_load_custom_chars(self, fontdata):
self.lcd_write(0x40);
for char in fontdata:
for line in char:
self.lcd_write_char(line)

```

ثم حفظ و الخروج من الملف عن طريق Ctrl + S ثم Ctrl + x

لتجربة الطباعة على الشاشة بفتح ملف نص لكتابة الكود (في نفس ملف العمل الذي تم انشاؤه) عن طريق الأمر

```
sudo nano test.py
```

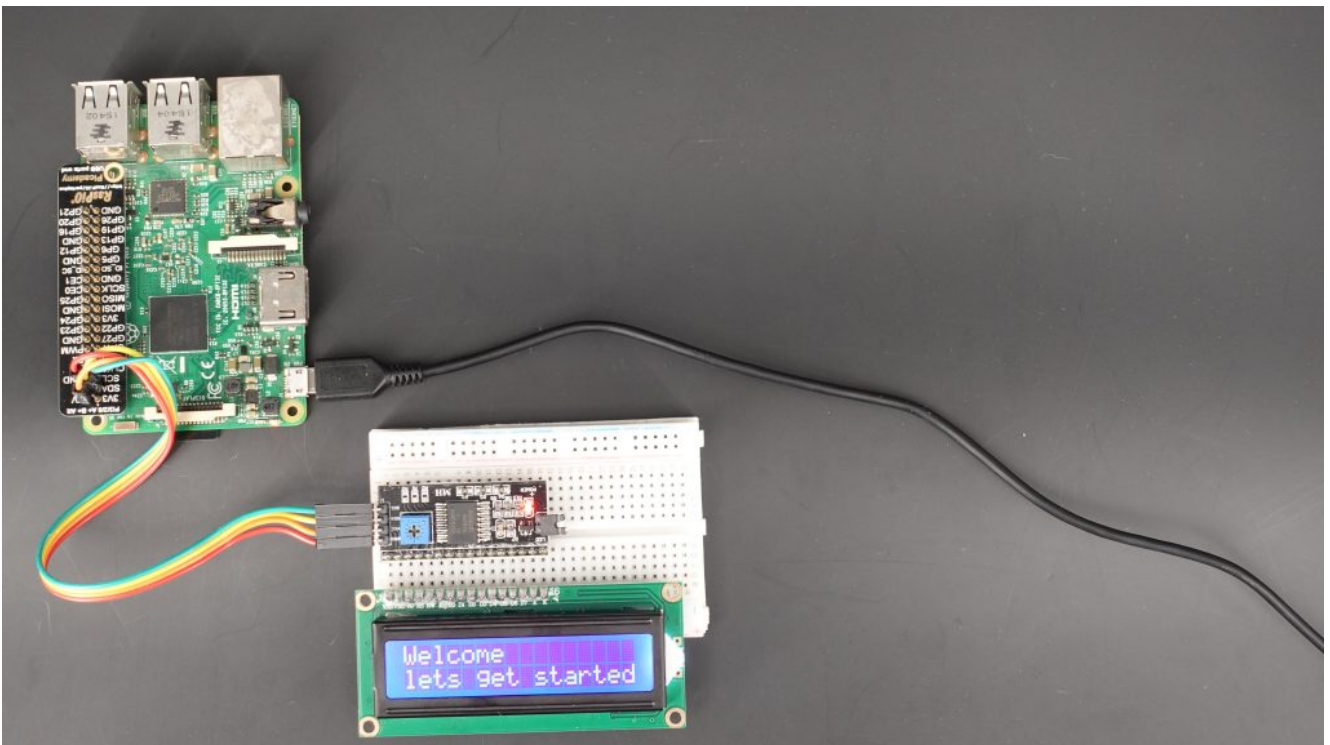
ثم أدخل الكود البرمجي

```
import i2c_lcd_driver #imports functions from i2c_lcd_driver.py file

mylcd = i2c_lcd_driver.lcd()

mylcd.lcd_clear() #clear the LCD screen
mylcd.lcd_display_string("Welcome", 1) #Display text on Line 1 of the LCD screen
mylcd.lcd_display_string("lets get started", 2) #Display text on Line 2 of the LCD
screen
```

تحتاج لضبط الشاشة من خلال المقاومة المتغيرة على الميديول لتظهر العبارة التي قمنا بطباعتها



انشاء صفحة الويب

# Status Indicator

This page allows you to select the text of your LCD status indicator.

Status set to Available

- Available
- Busy
- Away
- In a meeting
- On the phone
- Sending emails
- On a video call
- Clear Display

سيتم كتابة الكود البرمجي الخاص بصفحة الويب على بلغة html

داخل ملف سييت تسميته terminals

```
cd work
```

ثم

```
mkdir templates
```

ثم

```
sudo nano lcd.html
```

ثم قم باضافة الكود التالية

```
<!DOCTYPE html>
<!--Section 1 START-->
<html>
<body>

<h1>Status Indicator</h1>

<p>This page allows you to select the text of your LCD status indicator.</p>

<p id="demo"></p>
```

```

<button onclick="availableFunc()">Available</button>
<br>
<button onclick="busyFunc()">Busy</button>
<br>
<button onclick="awayFunc()">Away</button>
<br>
<button onclick="meetingFunc()">In a meeting</button>
<br>
<button onclick="phoneFunc()">On the phone</button>
<br>
<button onclick="emailFunc()">Sending emails</button>
<br>
<button onclick="videoFunc()">On a video call</button>
<br>
<button onclick="clearFunc()">Clear Display</button>

<!--Section 1 END-->

<script>
////////// Section 2 START//////////
var available = "http://0.0.0.0:5000/api/available";
var busy = "http://0.0.0.0:5000/api/busy";
var away = "http://0.0.0.0:5000/api/away";
var meeting = "http://0.0.0.0:5000/api/meeting";
var phone = "http://0.0.0.0:5000/api/phone";
var email = "http://0.0.0.0:5000/api/email";
var video = "http://0.0.0.0:5000/api/video";
var clear = "http://0.0.0.0:5000/api/clear";
////////// Section 2 END//////////
////////// Section 3 START//////////
function availableFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", available);
document.getElementById("demo").innerHTML = "Status set to Available";
xmlhttp.send();
}
function busyFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", busy);
document.getElementById("demo").innerHTML = "Status set to Busy";
xmlhttp.send();
}
function awayFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", away);
document.getElementById("demo").innerHTML = "Status set to Away";
xmlhttp.send();
}
function meetingFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", meeting);
document.getElementById("demo").innerHTML = "Status set to in a meeting";
xmlhttp.send();
}

```

```

}
function phoneFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", phone);
document.getElementById("demo").innerHTML = "Status set to on the phone";
xmlhttp.send();
}
function emailFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", email);
document.getElementById("demo").innerHTML = "Status set to sending to emails";
xmlhttp.send();
}
function videoFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", video);
document.getElementById("demo").innerHTML = "Status set to on a video call";
xmlhttp.send();
}
function clearFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", clear);
document.getElementById("demo").innerHTML = "Clear Status Screen";
xmlhttp.send();
}
////////// Section 3 END//////////
</script>

</body>
</html>

```

لمحة عن الكود البرمجي

```

<!DOCTYPE html>
<!--Section 1 START-->
<html>
<body>

```

سيتم طباعة النصوص التالية في بداية صفحة الويب

```

<h1>Status Indicator</h1>
<p>This page allows you to select the text of your LCD status indicator.</p>

```

هنا سيتم اظهار الحالة التي تم اختيارها

```

<p id="demo"></p>

```

عرض مفاتيح اختيار الحالة

```

<button onclick="availableFunc()">Available</button>

```

```

<br>
<button onclick="busyFunc()">Busy</button>
<br>
<button onclick="awayFunc()">Away</button>
<br>
<button onclick="meetingFunc()">In a meeting</button>
<br>
<button onclick="phoneFunc()">On the phone</button>
<br>
<button onclick="emailFunc()">Sending emails</button>
<br>
<button onclick="videoFunc()">On a video call</button>
<br>
<button onclick="clearFunc()">Clear Display</button>

<!--Section 1 END-->

```

ارسال البيانات المدخلة إلى الراسبيري باي

يتم كتابة رقم ال IP الخاص بالراسبيري باي بدلا من 0.0.0.0.

```

<script>
///Replace 0.0.0.0 with your Raspberry Pi's internal IP address below

var available = "http://0.0.0.0:5000/api/available";
var busy = "http://0.0.0.0:5000/api/busy";
var away = "http://0.0.0.0:5000/api/away";
var meeting = "http://0.0.0.0:5000/api/meeting";
var phone = "http://0.0.0.0:5000/api/phone";
var email = "http://0.0.0.0:5000/api/email";
var video = "http://0.0.0.0:5000/api/video";
var clear = "http://0.0.0.0:5000/api/clear";

```

اظهار الحالة في رأس الصفحة في خانة (Demo)

```

function availableFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", available);
document.getElementById("demo").innerHTML = "Status set to Available";
xmlhttp.send();
}
function busyFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", busy);
document.getElementById("demo").innerHTML = "Status set to Busy";
xmlhttp.send();
}
function awayFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", away);
document.getElementById("demo").innerHTML = "Status set to Away";
}

```

```

xmlhttp.send();
}
function meetingFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", meeting);
document.getElementById("demo").innerHTML = "Status set to in a meeting";
xmlhttp.send();
}
function phoneFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", phone);
document.getElementById("demo").innerHTML = "Status set to on the phone";
xmlhttp.send();
}
function emailFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", email);
document.getElementById("demo").innerHTML = "Status set to sending to emails";
xmlhttp.send();
}
function videoFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", video);
document.getElementById("demo").innerHTML = "Status set to on a video call";
xmlhttp.send();
}
function clearFunc() {
var xmlhttp = new XMLHttpRequest();
xmlhttp.open("GET", clear);
document.getElementById("demo").innerHTML = "Clear Status Screen";
xmlhttp.send();
}
////////// Section 3 END//////////
</script>

</body>
</html>

```

## برمجة البايثون

سيتم اضافة كود البرمجي داخل ملف (Work) الذي تم عمله لمشروع مؤشر حالة العمل

```
cd work
```

الشكل التالي يمثل توزيع الملف الذي تم انشاؤه



ثم

```
sudo nano status.py
```

ثم نكتب الأمر التالي

```
#!/usr/bin/env python

import i2c_lcd_driver
from time import sleep
from flask import Flask, jsonify, make_response, request, render_template
from datetime import datetime
mylcd = i2c_lcd_driver.lcd()

app = Flask(__name__)
#app.config['SERVER_NAME'] = 'caroline.local'

def currentTime():
    dateraw=datetime.now()
    timeFormat=dateraw.strftime("%-I:%M %p")
    return timeFormat

def switchTH() :
    mylcd.lcd_clear()
    mylcd.lcd_display_string("Welcome to", 1)
    sleep(1)

def switchAvailable() :
    cT=currentTime()
    mylcd.lcd_clear()
    mylcd.lcd_display_string("Status:Available", 1)
    mylcd.lcd_display_string("as of "+cT, 2)
    sleep(1)

def switchBusy() :
    cT=currentTime()
    mylcd.lcd_clear()
```



```

mylcd.lcd_display_string("Status: Busy", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchAway() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("Status: Away", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchMeeting() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("In a meeting", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchPhone() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("On the phone", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchGrading() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("Grading papers", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchEmail() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("Sending Emails", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchVideo() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("On a video call", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchClear() :
mylcd.lcd_clear()
sleep(1)

# API TH

```

```

@app.route('/api/th', methods=['GET'])
def apiTH() :
switchTH()
return jsonify({})

# API switchAvailable
@app.route('/api/available', methods=['GET'])
def apiavailable() :
switchAvailable()
return jsonify({})

# API Busy
@app.route('/api/busy', methods=['GET'])
def apiBusy() :
switchBusy()
return jsonify({})

# API Away
@app.route('/api/away', methods=['GET'])
def apiAway() :
switchAway()
return jsonify({})

# API switchMeeting
@app.route('/api/meeting', methods=['GET'])
def apiMeeting() :
switchMeeting()
return jsonify({})

# API switchPhone
@app.route('/api/phone', methods=['GET'])
def apiPhone() :
global globalLastCalledApi
globalLastCalledApi = '/api/Phone'
switchPhone()
return jsonify({})

# API switchEmail
@app.route('/api/email', methods=['GET'])
def apiEmail() :
switchEmail()
return jsonify({})

# API switchVideo
@app.route('/api/video', methods=['GET'])
def apiVideo() :
switchVideo()
return jsonify({})

# API clear
@app.route('/api/clear', methods=['GET'])
def apiClear() :

```

```

switchClear()
return jsonify({})

@app.errorhandler(404)
def not_found(error):
return make_response(jsonify({'error': 'Not found'}), 404)

@app.route('/')
def index():
#url_for('html', filename='lcd.html')
return render_template('lcd.html')

if __name__ == '__main__':
app.debug = True
app.run(host='0.0.0.0')

```

## شرح الأمر البرمجي

في البداية سيتم استدعاء المكتبات المطلوبة

```

#!/usr/bin/env python
import i2c_lcd_driver
from time import sleep
from flask import Flask, jsonify, make_response, request, render_template
from datetime import datetime
mylcd = i2c_lcd_driver.lcd()

```

تعريف دالة الوقت

```

def currentTime():
dateraw=datetime.now()
timeFormat=dateraw.strftime("%-I:%M %p")
return timeFormat

```

في بداية البرنامج سيتم طبع رسالة ترحيب على الشاشة، وكذلك يتم تحديد السطر الذي سيتم الطباعة عليه

```

def switchTH() :
mylcd.lcd_clear()
mylcd.lcd_display_string("Welcome ", 1)
sleep(1)

```

نعرف 8 دوال للحالات التي قد يختار المستخدم من بينها

```

def switchAvailable() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("Status:Available", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

```

```

def switchBusy() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("Status: Busy", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchAway() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("Status: Away", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchMeeting() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("In a meeting", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchPhone() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("On the phone", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchGrading() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("Grading papers", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchEmail() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("Sending Emails", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

def switchVideo() :
cT=currentTime()
mylcd.lcd_clear()
mylcd.lcd_display_string("On a video call", 1)
mylcd.lcd_display_string("as of "+cT, 2)
sleep(1)

```

نعرف API للحالات الثمانية التي تم تعريفها

```

# API TH
@app.route('/api/th', methods=['GET'])

```

```

def apiTH() :
switchTH()
return jsonify({})

# API switchAvailable
@app.route('/api/available', methods=['GET'])
def apiavailable() :
switchAvailable()
return jsonify({})

# API Busy
@app.route('/api/busy', methods=['GET'])
def apiBusy() :
switchBusy()
return jsonify({})

# API Away
@app.route('/api/away', methods=['GET'])
def apiAway() :
switchAway()
return jsonify({})

# API switchMeeting
@app.route('/api/meeting', methods=['GET'])
def apiMeeting() :
switchMeeting()
return jsonify({})

# API switchPhone
@app.route('/api/phone', methods=['GET'])
def apiPhone() :
global globalLastCalledApi
globalLastCalledApi = '/api/Phone'
switchPhone()
return jsonify({})

# API switchEmail
@app.route('/api/email', methods=['GET'])
def apiEmail() :
switchEmail()
return jsonify({})

# API switchVideo
@app.route('/api/video', methods=['GET'])
def apiVideo() :
switchVideo()
return jsonify({})

# API clear
@app.route('/api/clear', methods=['GET'])
def apiClear() :
switchClear()

```

```
return jsonify({})
```

اكتب ip الخاص بالراسبييري باي في مربع url على المتصفح مع رقم المنفذ ليتم ظهور صفحة الويب كالتالي (0.0.0.0:5000)