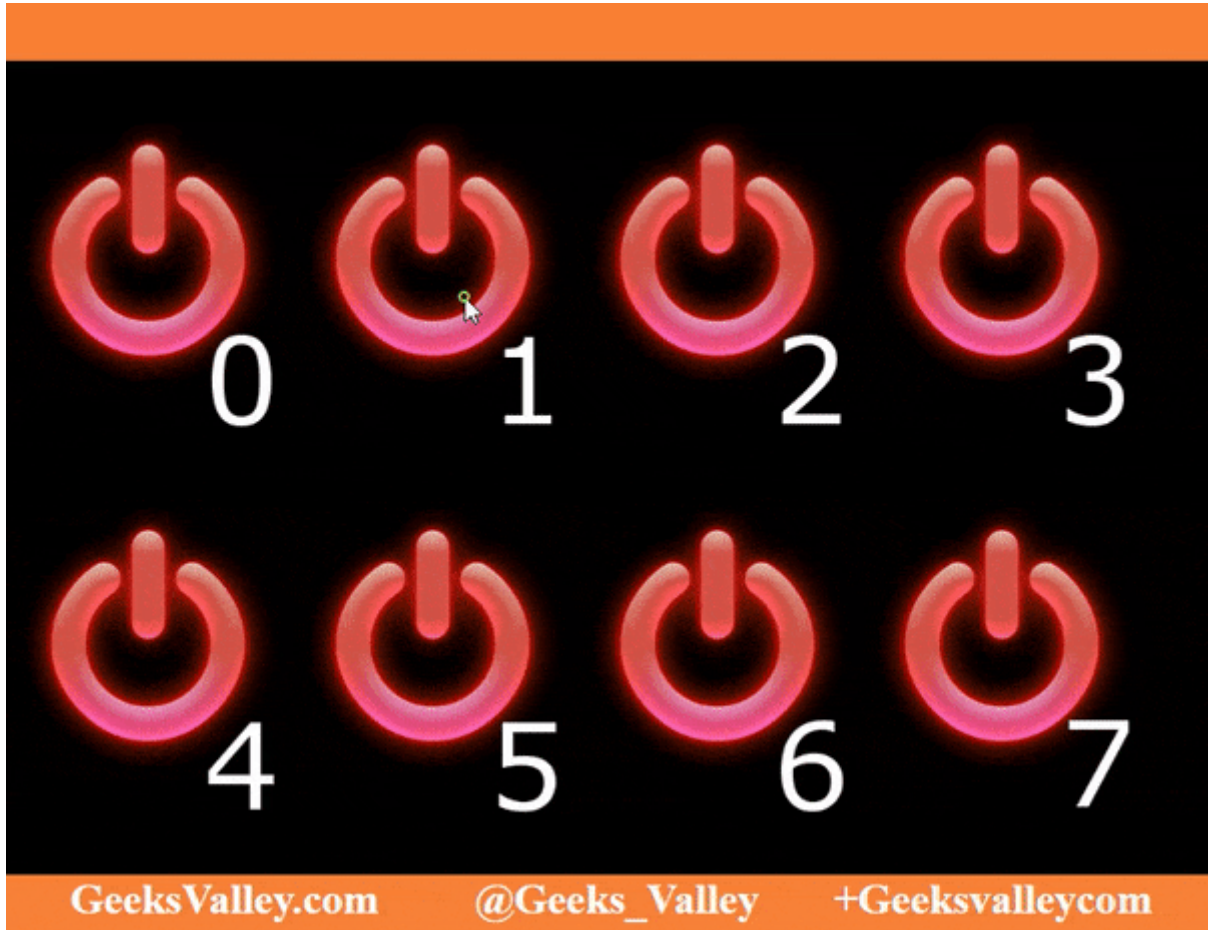


## نظام التحكم ومراقبة الإضاءة عن بعد

نظام التشغيل الآلي والتحكم للمنزل لا يربط كل الأجهزة التكنولوجية في منزلك ببعض وحسب، بل يربطك أنت بمنزلك. لقد صممنا نظام متكامل يمكنه التفاعل مع الأجهزة الكهربائية (الضوء) لمراقبتها و التحكم بها عن بعد لتجعل من حياتك أكثر متعة وراحة خاصة مع الأشياء التي تستخدمها كل يوم ضمن روتينك اليومي.

في هذا المشروع، سيتم المراقبة و التحكم بالإضاءة من اي مكان في العالم بإستخدام أي جهاز في المنزل مثل الهاتف الذكي أو الحاسوب أو tablet.



القطع المطلوبة :

الأدوات التي تحتاجها لهذا المشروع :



5mm Red LED

العدد: 8



مقاومة 220 اوم

العدد: 8



لوح تجارب كبير (Full size breadboard)



Raspberry Pi 3 Model B



اسلاك توصيل ذكر/ذكر (Jumper Wires Male Male)



اسلاك توصيل أنثى/ذكر (Jumper Wires Female/male)



Power Supply



Samsung 8Gb Class 6 Microsd Memory Card

## نظرة عامة :

يبدأ النظام من خلال ارسال اوامر المستخدم عن طريق استخدام تطبيق على الهواتف الذكية او صفحة web الى الخادم (server) عبر الانترنت، ومن ثم يقوم الخادم بإرسال هذه الأوامر الى لوحة التحكم (الراسبيري باي ) للتحكم في إيقاف و تشغيل الاضاءه المنزلية.



يمكنك من خلال هذا النظام التحكم بالتوصيلات الكهربائية (Powerstrip)، التحكم تدفئة أو تبريد المنزل، التحكم في تشغيل و اطفاء الالات الكهربائية.

## تصميم لوحة التجارب:

في جزء التوصيل الإلكتروني سيتم توصيل 8 LEDs فقط.الجزء المختلف في التوصيل هو إختيار المنفذ لكل LEDs . في الواقع ، سوف نستخدم في نظام التحكم ومراقبة الإضاءة مكتبة تدعى wiringPi و أرقام المنافذ في هذه المكتبة ليست هي نفس التي على الراسبيري باي.سيتم تسمية المنافذ على الراسبيري باي اعتمادا على مكتبة wiringPi كما هو موضح بالجدول التالي :

| P1: The Main GPIO connector |                 |       |        |       |          |              |
|-----------------------------|-----------------|-------|--------|-------|----------|--------------|
| WiringPi Pin                | BCM GPIO        | Name  | Header | Name  | BCM GPIO | WiringPi Pin |
|                             |                 | 3.3v  | 1 2    | 5v    |          |              |
| 8                           | Rv1:0 - Rv2:2   | SDA   | 3 4    | 5v    |          |              |
| 9                           | Rv1:1 - Rv2:3   | SCL   | 5 6    | 0v    |          |              |
| 7                           | 4               | GPIO7 | 7 8    | TxD   | 14       | 15           |
|                             |                 | 0v    | 9 10   | RxD   | 15       | 16           |
| 0                           | 17              | GPIO0 | 11 12  | GPIO1 | 18       | 1            |
| 2                           | Rv1:21 - Rv2:27 | GPIO2 | 13 14  | 0v    |          |              |
| 3                           | 22              | GPIO3 | 15 16  | GPIO4 | 23       | 4            |
|                             |                 | 3.3v  | 17 18  | GPIO5 | 24       | 5            |
| 12                          | 10              | MOSI  | 19 20  | 0v    |          |              |
| 13                          | 9               | MISO  | 21 22  | GPIO6 | 25       | 6            |
| 14                          | 11              | SCLK  | 23 24  | CE0   | 8        | 10           |
|                             |                 | 0v    | 25 26  | CE1   | 7        | 11           |
| WiringPi Pin                | BCM GPIO        | Name  | Header | Name  | BCM GPIO | WiringPi Pin |

قم بتوصيل القطع الإلكترونية كما هو موضح بالصورة :



يتم توصيل الطرف السالب للـ LEDs بمقاومة 220 اوم و الطرف الموجب حسب الجدول التالي:

| WiringPi Pin | BCM GPIO | رقم الـ LED |
|--------------|----------|-------------|
| 0            | GPIO 17  | 1           |
| 1            | GPIO 18  | 2           |
| 2            | GPIO 27  | 3           |
| 3            | GPIO 22  | 4           |
| 4            | GPIO 23  | 5           |
| 5            | GPIO 24  | 6           |
| 6            | GPIO 25  | 7           |
| 7            | GPIO 4   | 8           |

## تثبيت و إستخدام المكتبة Wiring Pi :

مكتبة Wiring Pi هي مكتبة الوصول إلى GPIO على لوحة الراسبيري باي . و انها تسمح لك التحكم بمدخل و مخارج الراسبيري باي من خلال bash script او مباشر من خلال سطر أوامر.

تحقق أولاً ما اذا كانت مكتبة wiringPi مثبتة مسبقاً ام لا ، من خلال نافذة الـ terminal قم بتشغيل الامر التالي:

```
gpio -v
```

اذا لم تقم بتثبيت GIT، يمكنك تثبيتها من خلال الأمر التالي :

```
sudo apt-get install git-core
```

اذا تم ظهور رسالة خطأ هنا ، تأكد من تحديث الراسبيري باي إلى آخر اصدار من Raspbian :

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

للحصول على WiringPi باستخدام GIT، قم بكتابة الأمر التالي:

```
cd
```

```
git clone git://git.drogon.net/wiringPi
```

```
cd ~/wiringPi  
git pull origin
```

سوف تجلب النسخة المحدثة ثم يمكنك إعادة تشغيل البرنامج من خلال الأمر أدناه .  
لبناء/ تثبيت البرنامج :

```
cd ~/wiringPi  
./build
```

## الكتابة و القراءة من المنافذ عن طريق استخدام أوامر المكتبة wiringPi :

الآن يجب أن تكون قادر على إستخدام الأداة، قم بكتابة الأمر التالي على نافذة الـ Terminal :

```
gpio mode 0 out
```

إذا لم يتم ظهور شيء أو رسالة خطأ فقد تم التثبيت للمكتبة بشكل صحيح. في حال تم ظهور رسالة الخطأ "command not found" error" أو شيء من هذا القبيل، تأكد من تثبيت و بناء المكتبة. للقيام بتشغيل و إطفاء الـ LED (Wiring pin 0) ، تحتاج أولاً إلى تعيين الدبوس كمخرج من خلال الأمر التالي :

```
gpio mode 0 out
```

"0" هي رقم المنفذ wiring، و "OUT" لتعيين المنفذ كمخرج . سيتم تشغيل الـ LED من خلال الأمر التالي :

```
gpio write 0 1
```

"0" للدلالة على رقم المنفذ ، و "1" لوضع الدبوس في حالة التشغيل الـ LED (للتشغيل 1 و لإطفاءه 0) . و لإيقاف تشغيلك ببساطة استخدم الأمر :

```
gpio write 0 0
```

وهناك أيضاً أوامر لقراءة GPIO والتي تسمح لقراءة وضع الدبوس. فيمكن من خلاله معرفة حالة الضوء هل هو في حالة التشغيل أو الإيقاف إذا لم تكن على إستطاعه لرؤية الضوء. يمكنك ذلك عن طريق الأمر التالي :

```
gpio read 0
```

"0" لتعيين رقم المنفذ Wiring. هذا الأمر يقوم بترجيع قيمة 1 إذا كان الضوء في حالة التشغيل و قيمة 0 إذا كان الضوء في حالة الإيقاف.

وأخيراً مكتبة wiringPi تحتوي على الكثير من الأوامر و الدوال التي تمكنك من التحكم بمداخل و مخرج الراسبييري باي ولكن لن يتم تغطيتها جميعها في هذا المشروع. يمكنك الاطلاع على هذه الروابط إذا كنت مهتم بمعرفة المزيد :

[/http://wiringpi.com/reference](http://wiringpi.com/reference)

[/https://projects.drogon.net/raspberry-pi/wiringpi/the-gpio-utility](https://projects.drogon.net/raspberry-pi/wiringpi/the-gpio-utility)

نظام التحكم ومراقبة الإضاءة عن بعد مع SSH هو جميل ولكن الواجهة ليست سهلة الاستعمال و كتابة الأوامر في كل مرة أمر طويل و مزعج. هذا هو السبب في أننا بحاجة إلى واجهة رسومية لمشروعنا. كما أن برمجة تطبيق لكل نظام تشغيل (IOS, Android, Windows phone ,Mac, Linux, Windows ,....) ستكون طويلة جدا و يتطلب أن تكون على معرفة بعدة لغات مختلفة.

هذا هو السبب لإنشاء صفحة ويب على شبكة النت ، حيث انها ستكون متوافقة مع جميع الأجهزة ، و ستكون بحاجة لمعرفة أربع لغات : HTML,CSS (نمط الصفحة)، PHP (للتفاعل مع الخادم) ، JavaScript (التفاعلات مع المستخدم) . نحن بحاجة لتثبيت خادم الويب على الراسبيري باي. في حالتنا نحن لسنا بحاجة إلى قاعدة بيانات MySQL ، فقط بحاجة إلى خادم HTTP و PHP.

## تثبيت خادم الويب (Web Server) :

أولا قم بعمل تحديث للراسبيري باي :

```
sudo apt-get update
```

قم بتثبيت الخادم (Apache HTTP server) و PHP من خلال الأمر التالي :

```
sudo apt-get install php libapache2-mod-php -y
```

الان قم بالتحقق ما اذا كان الخادم (Server) يعمل، قم بكتابة العنوان IP الخاص بالراسبيري باي على متصفح الويب. اذا كان يعمل ستظهر رسالة "It works" كما هو موضح بالصورة أدناه.



إذا لم يتم مثل ما سبق، قم بالتحقق من العنوان IP الخاص بالراسبيري باي من خلال الأمر التالي :

```
ifconfig
```

أو قم بمحاولة إعادة تثبيت Apache أو إعادة تشغيل الراسبيري باي.

الملف الخاص بواجهة المستخدم :

قم بالدخول إلى ملف /var/www/html/ عبر نافذة الTerminal من خلال الأمر التالي :

```
cd /var/www/html
```

قم بعرض الملفات الموجودة على هذا الملف بإستخدام الامر التالي:

```
ls
```

سيظهر لك ملف واحد يسمى "index.html" هذا الملف يتوافق مع صفحة "It works!". يمكنك حذفه ليتم إنشاء الصفحة الخاصة بهذا المشروع. قم بحذف الملف بإستخدام الأمر التالي :

```
sudo rm index.html
```

وإنشاء ملف آخر يسمى "index.php" :

```
sudo nano index.php
```

ثم قم بكتابة النص التالي داخل الملف:

```
<?php  
phpinfo();  
?>
```

بعد حفظ الملف ، قم بعمل تحديث للمتصفح الخاص بك. يجب أن تشاهد صفحة طويلة مع الكثير من المعلومات حول الخادم و PHP .

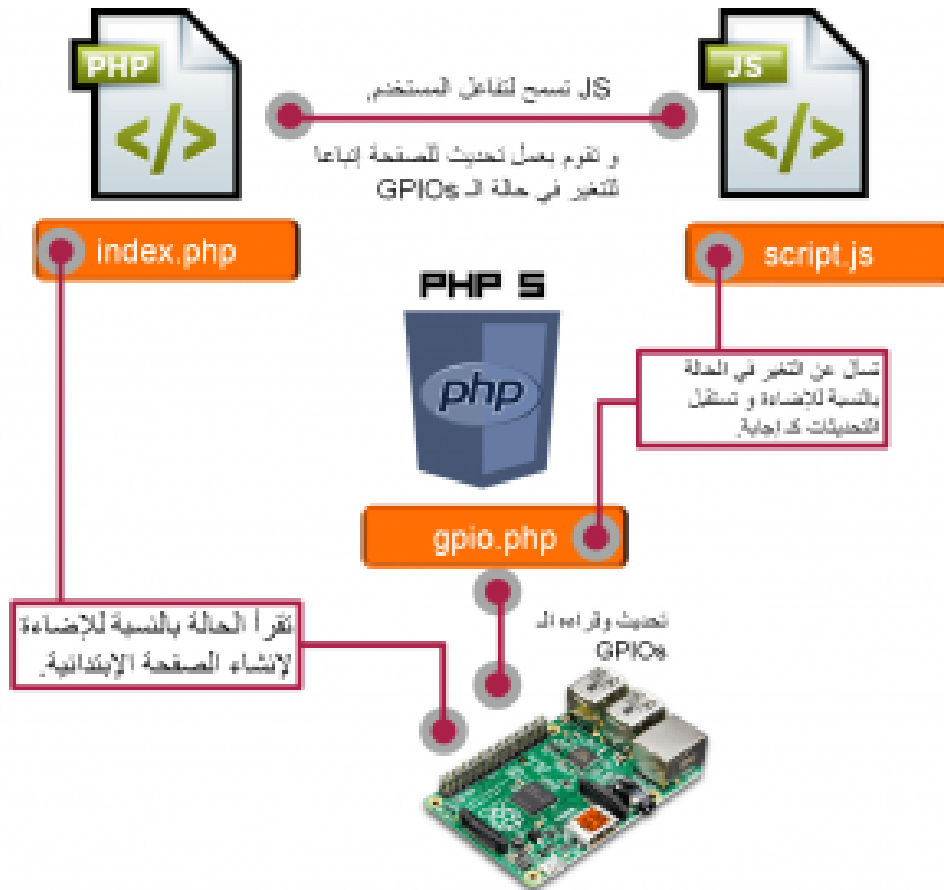
## إنشاء واجهة المستخدم :

يمكننا التحكم بالراسبيري باي من خلال PHP script ولكن لن يكون هناك واجهة لتفعيل المستخدم ، و بالتالي فإننا لن نتمكن من إيقاف و تشغيل الـ LED.

واجهة المستخدم تتألف من صور عبارة عن أزرار تشغيل و إيقاف. الأخضر للدلالة على التشغيل و الأحمر للدلالة على الإيقاف . كل صورة/ زر يتم ربطه مع LED خاص به . لذلك اذا تم النقر على أحد الصور/الأزرار ، سيتم تشغيل أو إيقاف الـ LED المرتبط به و سيتم تغيير الصورة من زر أحمر إلى أخضر او العكس .



سيتم إنشاء واجهة الصفحة باستخدام لغة HTML، و لتفاعلات الخادم وتوليد الصفحة لغة PHP، و JavaScript لإدارة التفاعل مع المستخدم و الرسوم المتحركة للصفحة. ويمكنك استخدام CSS لتخطيط و تنسيق الصفحة مثل الخلفية .



اولا نحن بحاجة لملف "index.php" لإنشاء الواجهة ولتفاعل الخادم . هذه الصفحة هي الصفحة الرئيسية التي تحتوي على 8 ازرار.

التحكم بالإضاءة من خلال ملف PHP :

تنفيذ التطبيق مع كود PHP يمكن القيام به بكتابة دالتين مختلفين : exec و system . أولاً الدالة "system" ، تأخذ هذا الامر متغيرين :

```
system ( string $command, int $return_var )
```

المتغير الأول هو الأمر الذي سيتم تنفيذه، و المتغير الثاني هو الحالة المستردة من تنفيذ الأمر. والمتغير الثاني لا يكون إلزاميا. وهكذا يمكنك إستخدامه إذا كنت بحاجة إلى تنفيذ الأوامر "gpio mode 0 out" او "gpio write 0 1" مثال :

```
<?php
system ( "gpio mode 0 out" );
system ( "gpio write 0 1" );
?>
```

ثم الدالة "exec"، هذه الدالة تقوم بالضبط بنفس العمل للدالة "system" ولكن تقوم بقراءة و تخزين ما طبعه الامر. حيث انه يأخذ ثلاث متغيرات :

```
exec ( string $command, array $output, int $return_var )
```

الإختلاف بالمتغيرات هو مصفوفة \$output. سيتم تخزين ما يتم قراءته خلال الأمر في هذه مصفوفة. وهكذا، يمكنك استخدام هذه الدالة إذا كنت بحاجة إلى قراءة أحد المنافذ مثل الأمر "gpio read 0" وتخزين القيم التي تم استجاعتها. مثال:

```
<?php
exec ( "gpio read 0", $status );
print_r ( $status );
?>
```

سنقوم بهذا المشروع بالتحكم بالمصابيح المربوطة على المنافذ من 0 على 7، يتم إنشاء الثمانية أزرار لأول مره في For loop بإستخدام :

```
exec ( "gpio read ".$i, $output );
```

قم بفتح ملف index.php ، ثم قم بكتابة الكود التالي في الملف و قم بحفظه:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <title>Raspberry Pi Gpio</title>
    <div id="rectangle" style="width:100%;left:0;height:70px;
background:#f88238;"></div>
</head>

<body style="background-color: black;margin:0 auto ; align:middle;">
<!-- On/Off button's picture -->
<?php
$val_array = array(0,0,0,0,0,0,0,0);
//this php script generate the first page in function of the file
for ( $i= 0; $i<8; $i++) {
    //set the pin's mode to output and read them
    system("gpio mode ".$i." out");
    exec ("gpio read ".$i, $val_array[$i], $return );
}
```

```
//for loop to read the value
$i =0;
for ($i = 0; $i < 8; $i++) {
    //if off
    if ($val_array[$i][0] == 0 ) {
        echo ("<img id='button_". $i ."' src='data/img/red/red_". $i .".jpg'
onclick='change_pin (\". $i .\");'/>");
    }
    //if on
    if ($val_array[$i][0] == 1 ) {
        echo ("<img id='button_". $i ."' src='data/img/green/green_". $i .".jpg'
onclick='change_pin (\". $i .\");'/>");
    }
}
?>

<!-- javascript -->
<script src="script.js"></script>
</body>
</html>
```

نحن بحاجة إلى التحقق ما اذا تم النقر على أحد الأزرار من قبل المستخدم . حيث ان JavaScript تفيد لهذا الغرض، سيتم وضعها في ملف منفصل يسمى "script.js" ولكنها لا تزال مدرجة في ملف "index.php" هو مجرد إضافة مستمع (listener) إلى كل من الثمانية أزرار، في كل مرة يتم الضغط على هذه الأزرار ، سيتم استخدام دالة لسؤال ملف gpio.php عن حالة الإضاءة و تستقبل التحديثات كإجابة. و أخير ستقوم الـ JavaScript بتغيير صورة الأزرار إلى الأحمر للدلالة على الإيقاف أو إلى أخضر في حالة تشغيل الإضاءة.

و في الملف الأخير، gpio.php . يحتوي على كود PHP لتشغيل و إيقاف الإضاءة اعتمادا على ما يتم ارسالة من الـ JavaScript .

يمكنك تحميل برمجة كامل المشروع من خلال الرابط المباشر على الموقع [هنا](#) . يمكنك تحميل الملف على الـ راسبيري باي على ملف /var/www/html/ ثم قم بفتح عنوان IP الخاص بالراسبيري باي على متصفح ويب.