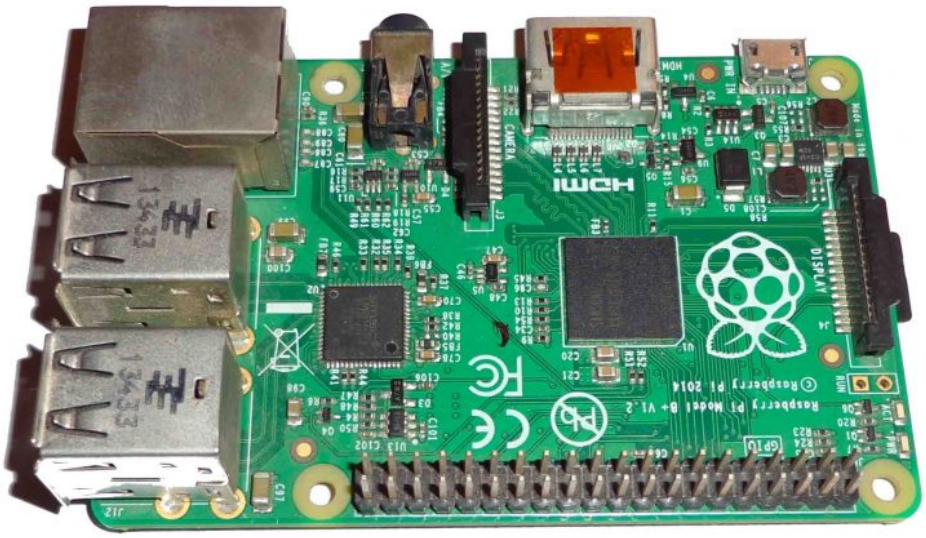
زر إغلاق الراسبري باي

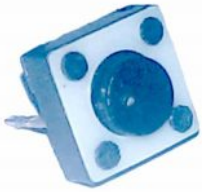
فى هذا الدرس سنتعلم كيفية عمل زر لإغلاق الراسبرى باى بشكل آمن بدون الإضطرار لإغلاقها من خلال أمر الإغلاق ولا فصل مصدر الطاقة عنها.

المكونات المطلوبة



راسبری پای

سلك توصيل female \ female



زر ذو طرفين press button

مكواة لحام

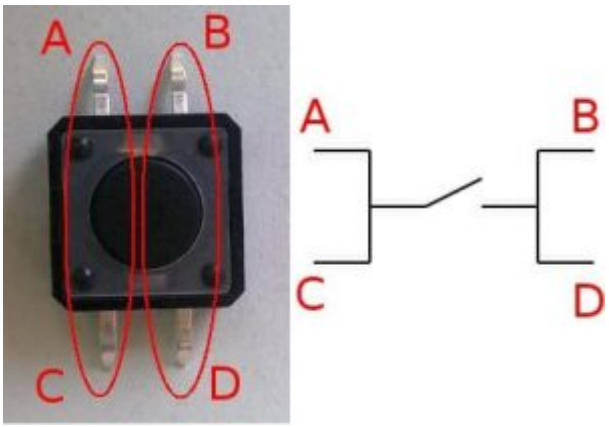
قصدير لحام

توصيل الدائرة

نقوم بقص سلك التوصيل لنصفين متساويين فى الطول ثم نقوم بقطع جزء من الغلاف البلاستيكي الحامي للسلك حتى يظهر جزء مناسب من السلك فنستطيع لحامه فى زر الضغط.

نقوم بلحام طرفيى السلكين فى طرفى رز الضغط كالتالي.

إذا كان زر الضغط من ذوى الأربعة أطراف فعليك ملاحظة الصورة التالية وأن تقوم بلحام طرفى الأسلاك فى النقطتين A,B أو C,D.



كل ما عليك فعله الآن هو إختيار أحد أرجل التوصيل GPIO المناسبة والتى ستختلف طبقاً لنوع الراسبرى باى المستخدمة، في حالتي هذه أستخدم راسبرى باى موديل B+ فقمت بإختيار أقرب رجل توصيل لأحد رجول الأرضي GND وكانت هى الرجل GPIO7نا للا وبإمكانك إختيار ما تريد.

3.3V	1	2	5V
I2C1 SDA	3	4	5V
I2C1 SCL	5	6	GROUND
GPIO4	7	8	UART TXD
GROUND	9	10	UART RXD
GPIO 17	11	12	GPIO 18
GPIO 27	13	14	GROUND
GPIO 22	15	16	GPIO 23
3.3V	17	18	GPIO 24
GPIO 10 MOSI	19	20	GROUND
GPIO 9 MISO	21	22	GPIO 25
GPIO 11 SCLK	23	24	GPIO 8
GROUND	25	26	GPIO 7
DNC	27	28	DNC
GPIO 5	29	30	GROUND
GPIO 6	31	32	GPIO 12
GPIO13	33	34	GROUND
GPIO 19	35	36	GPIO16
GPIO 26	37	38	GPIO 20
GROUND	39	40	GPIO 21

نقوم بتوصيل الزر السابق الإعداد فى الطرفين رقم 26 و رقم 25 من أرجل التوصيل GPIO كالتالي

الكود البرمجى

فى البداية نقوم بفتح ال Terminal الخاص ب الراسبري باي وكتابة الأوامر التالية أو نقوم بفتح نافذه الـ SSH الخاصة بها من جهاز آخر مربوط معها على نفس الشبكة كما تم شرحة فى الدرس الخامس.

نقوم بكتابة الأمر التالي الذى يقوم بفتح محرر النصوص الشهير Nano ويقوم بإنشاء ملف اسمه off_button_pi.py لكتابة الكود بداخلة بلغة البايثون python.

```
nano off_button_pi.py
```

نقوم بكتابة الكود التالي بداخل الملف السابق

```
pi@raspberrypi: ~
  GNU nano 2.2.6            File: off_button_pi.py

#!/bin/ python

import RPi.GPIO as GPIO
import os

gpio_pin_number=7

GPIO.setmode(GPIO.BCM)
GPIO.setup(gpio_pin_number,GPIO.IN, pull_up_down=GPIO.PUD_UP)

try:
   GPIO.wait_for_edge(gpio_pin_number, GPIO.FALLING)
   os.system("sudo shutdown -h now")

except:
   pass

GPIO.cleanup()

                          [ Read 18 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

ثم نقوم بالضغط على Ctrl+O ثم enter ثم ctrl+X وذلك لحفظ الملف السابق.

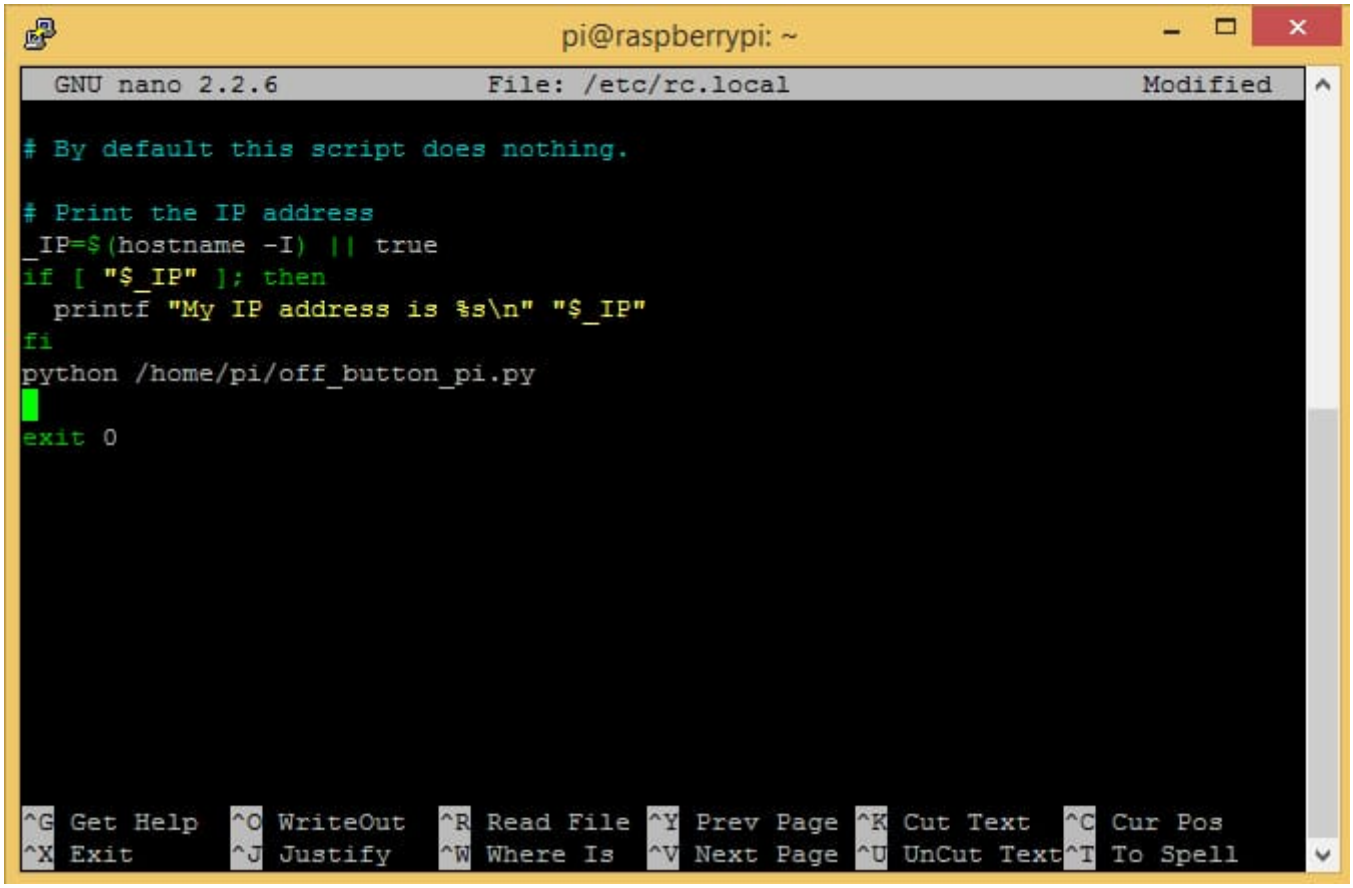
وكما نلاحظ الكود بسيط جداً يعتمد على إستدعاء مكتبتين رئيسيتين وهما GPIO وهى للتعامل مع أرجل التوصيل وجعل الرجل الموصلة على زر الضغط تعمل كطرف إدخال، وكذلك لتشغيل الـ Pull Up Resistor الداخلية فى الراسبرى باى فيمكننا بذلك جعل رجل التوصيل موصلة دائماً على فولت High وعند الضغط فقط على الزر تصبح موصلة بـ Low
ومكتبة os وهى الأهم هنا حيث تمكننا من التعامل مع نظام التشغيل وارسال أوامر له عن طريق كود البايثون فنستخدمها داخل أمر try الذى يتنظر أن يتم الضغط على الزرعن طريق أمر GPIO.wait_for_edge ليقوم بعد ذلك بإرسال الأمر sudo shutdown -h now ليغلق الراسبرى باى بطريقة سليمة.

إذا كنت تريد تغيير رجل التوصيل GPIO كل ماعليك هو تغيّر رقمها فى الملف السابق gpio_pin_number لتصبح تساوى رقم الـ GPIO الأخرى التى تريد إستخدامها.

يبقى لنا خطوة واحده وهى أن نجعل هذا البرنامج يعمل تلقائياً عند فتح الراسبرى باى وذلك بفتح الملف rc.local بمحرر النصوص Nano مع كتابة sudo قبل الأمر وذلك ليعطى للمستخدم صلاحيات الـ super user فيمكنك تعديل هذا الملف.

```
sudo nano /etc/rc.local
```

نقوم بإضافة مسار الملف الذى قمنا بإنشائه فى آخر rc.local قبل exit 0 مع كتابة اسم اللغة المستخدمة فى كتابة البرنامج قبل المسار وهى python كالتالى.



ثم نقوم بالضغط على Ctrl+O ثم enter ثم Ctrl+X وذلك لحفظ الملف السابق.

كل ما تبقى عليك هو عمل reboot للراسبرى باى هكذا.

```
sudo reboot
```

تستطيع الآن الضغط على زر الإغلاق وهو سيقوم بطريقة سليمة وصحيحة بإغلاق الراسبرى باى.