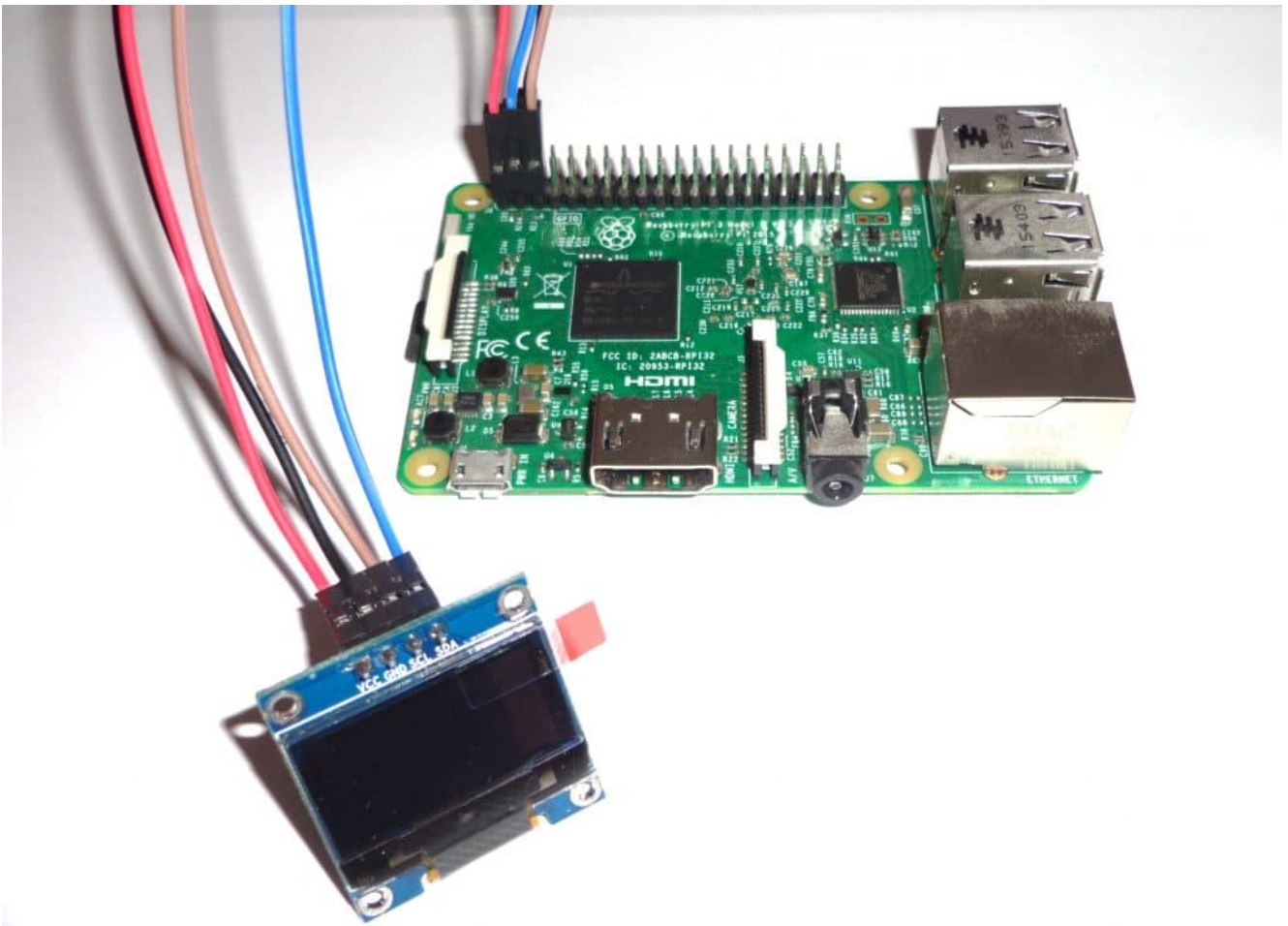


توصيل شاشة OLED على الراسبيري باي

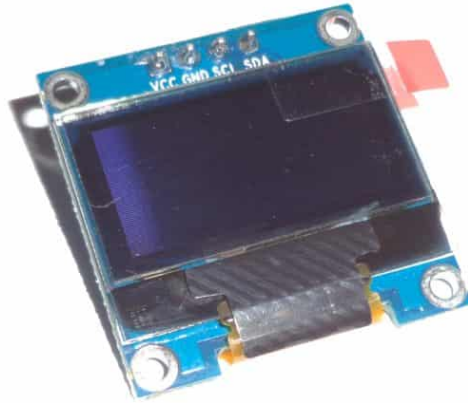
في هذا الدرس سنتعلم كيفية توصيل شاشة OLED عدد نقاطها هو 64x182 ويتم توصيلها على الراسبيري باي عن طريق بروتوكول I2C، حيث يمكننا ذلك من عرض ما نريد وقراءة قيمة أي مستشعر مباشرة على شاشة صغيرة بدون الاضطرار لتوصيل شاشة كبيرة للراسبيري باي ومن أهم مميزات هذه الشاشة هي صغر الحجم ودرجة الوضوح العالية للصورة وإنخفاض إستهلاك الطاقة بدرجة كبيرة.



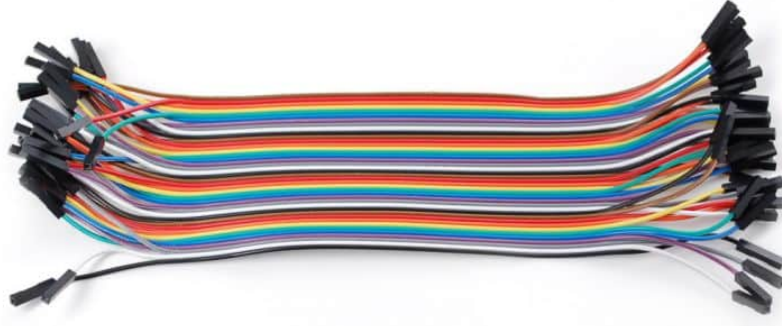
المكونات المطلوبة



راسبيري باي



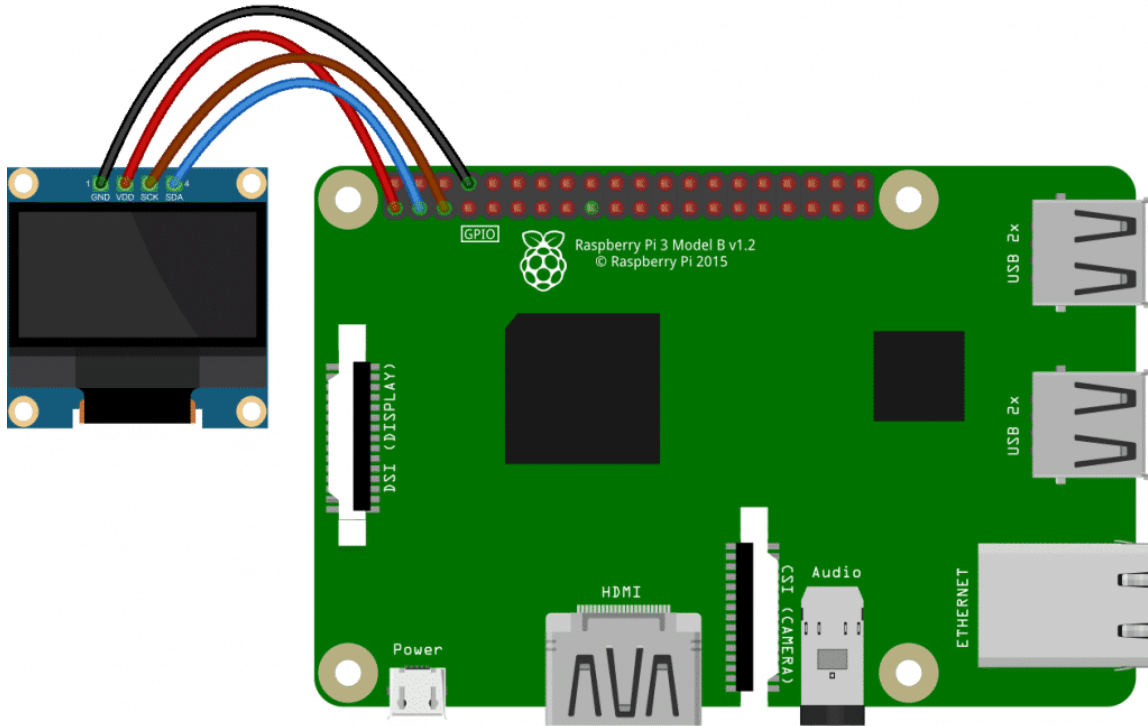
شاشة عرض OLED

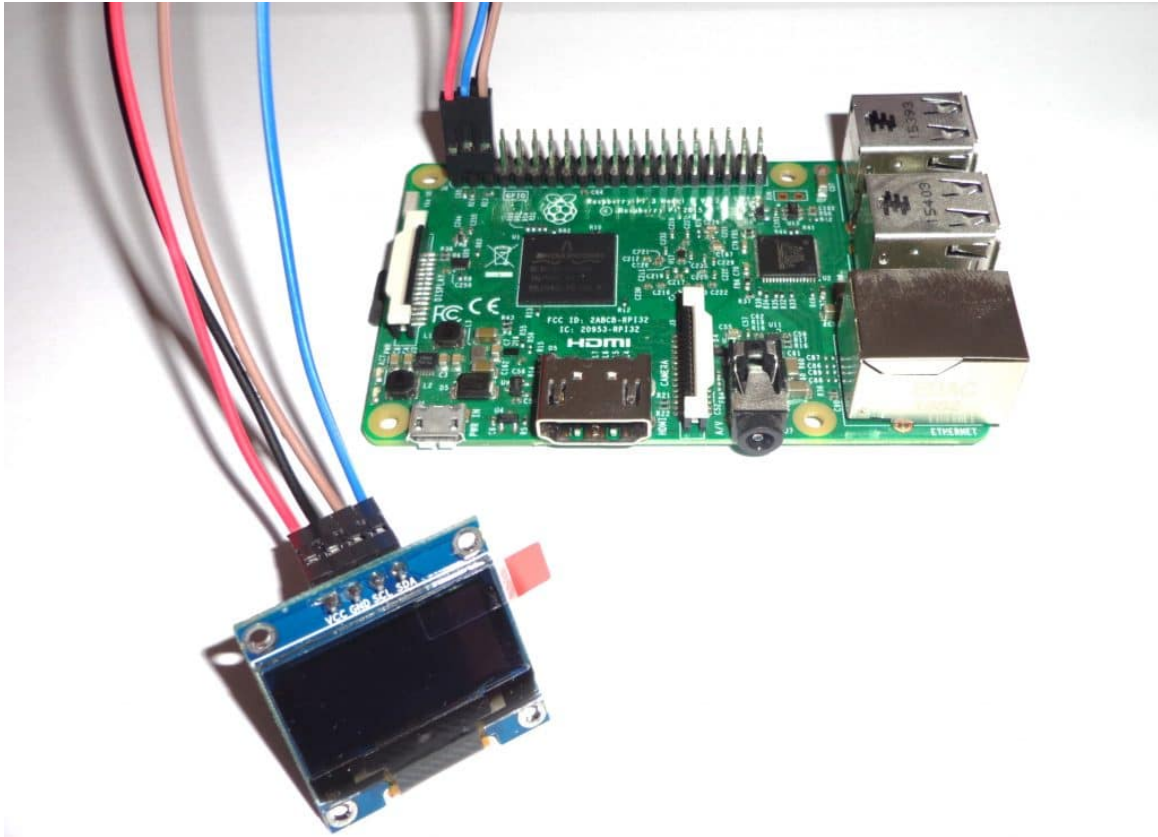


أسلاك توصيل Female / Female jumper

توصيل الدائرة

نقوم بتوصيل الدائرة كما في الصورة التالية ومن الممكن أن نقوم بتوصيلها بإستخدام لوحة التجارب الـ breadboard





الكود البرمجي

في البداية نقوم بفتح الـ Terminal الخاص بالراسبيري باي وكتابة الأوامر التالية أو نقوم بفتح نافذة الـ SSH الخاصة بها من جهاز آخر مربوط معها علي نفس الشبكة كما تم شرحه في الدرس الخامس في البداية نقوم بتحديث نظام تشغيل الـ راسبيري باي وتنصيب مكتبة GPIO الخاصة بالتحكم في أطراف الإدخال والإخراج.

```
sudo apt-get update
sudo apt-get install build-essential python-dev python-pip
sudo pip install RPi.GPIO
```

ثم نقوم بتنصيب مكتبات إضافية لتعمل الشاشة على الـ راسبيري باي.

```
sudo apt-get install python-imaging python-smbus
```

ثم نقوم بتحميل ملفات مكتبة SSD1306 للتحكم في الـ OLED والمعدة مسبقاً من Adafruit من خلال الأوامر التالية.

```
sudo apt-get install git
git clone https://github.com/adafruit/Adafruit_Python_SSD1306.git
```

نقوم بتغيير المسار إلى الملف الذي قمنا بتنزيله ونقوم بعمل تنصيب للمكتبة من خلال الأوامر التالية.

```
cd Adafruit_Python_SSD1306
sudo python setup.py install
```

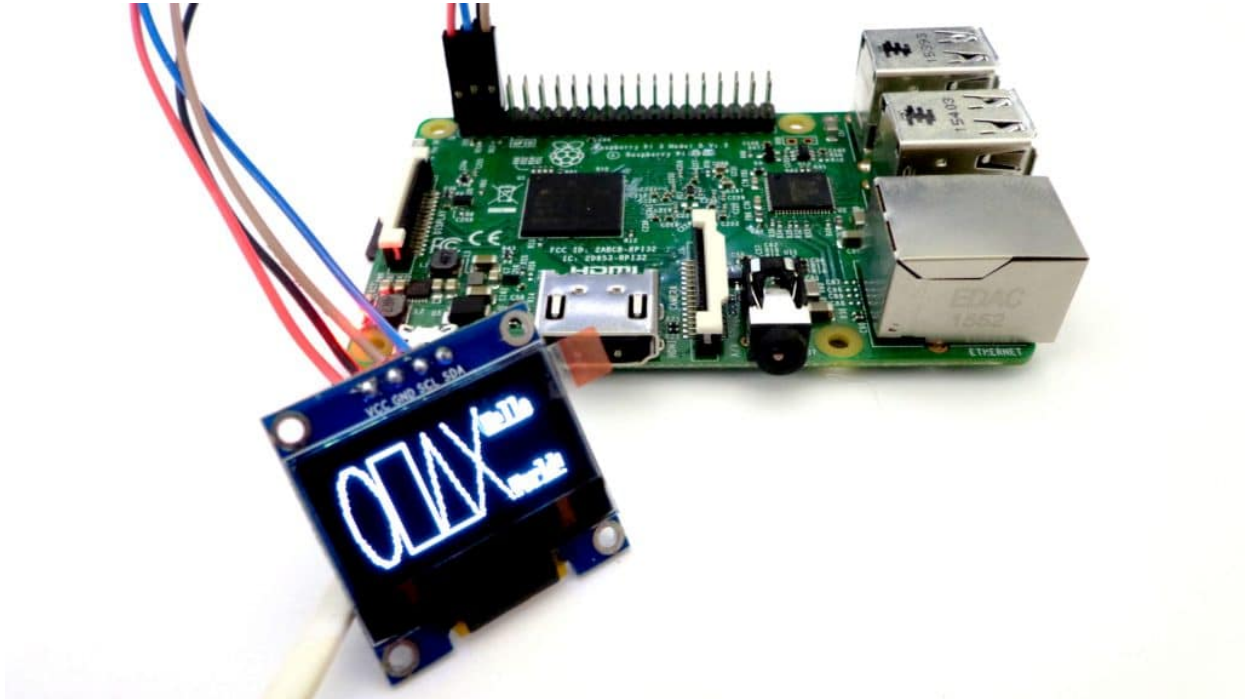
بمجرد الإنتهاء من تنصيب المكتبة تصبح جاهزة مباشرة لإستدعائها من خلال أي برنامج Python عن طريق كتابة الأمر التالي في ملف الـ Python.

```
import Adafruit_SSD1306
```

سنقوم الآن بالذهاب لملف examples داخل مكتبة الـ OLED لنشغل أحد الأمثلة حتى نتأكد من أن الشاشة تعمل بنجاح وذلك كالتالي.

```
cd examples  
sudo python shapes.py
```

سنلاحظ أن الشاشة تعمل وتعرض مجموعة من الأشكال المختلفة وكذلك كلمة Hello World كالتالي.



يجب ملاحظة أن هذه المكتبة تدعم بروتوكول I2C وكذلك SPI وكذلك عدد مختلف من مقاسات الشاشات، فكل ما عليك أن تقوم بتحدد البروتوكول الذي تعمل عليه شاشتك وفي هذه الحالة I2C ومقاس الشاشة 64x128، فنقوم بفتح ملف shapes.py من خلال برنامج nano.

```
sudo nano shapes.py
```

```
# 128x32 display with hardware I2C:  
disp = Adafruit_SSD1306.SSD1306_128_32(rst=RST)  
# 128x64 display with hardware I2C:  
# disp = Adafruit_SSD1306.SSD1306_128_64(rst=RST)  
# Alternatively you can specify an explicit I2C bus number, for example  
# with the 128x32 display you would use:  
# disp = Adafruit_SSD1306.SSD1306_128_32(rst=RST, i2c_bus=2)  
  
# 128x32 display with hardware SPI:  
# disp = Adafruit_SSD1306.SSD1306_128_32(rst=RST, dc=DC, spi=SPI.SpiDev(SPI_PORT,  
SPI_DEVICE, max_speed_hz=8000000))  
# 128x64 display with hardware SPI:  
# disp = Adafruit_SSD1306.SSD1306_128_64(rst=RST, dc=DC, spi=SPI.SpiDev(SPI_PORT,  
SPI_DEVICE, max_speed_hz=8000000))
```

كل ما عليك هو أن تقوم بإزالة علامة # من أمام البروتوكول والمقاس المناسب لشاشتك ووضعها قبل كل المقاسات الأخرى، وفي حالة

```
# 128x64 display with hardware I2C:
# disp = Adafruit_SSD1306.SSD1306_128_64(rst=RST)
```

بإمكانك قراءة المزيد ومراجعة الكود المكتوب في برنامج py فهو سهل وبسيط حيث يقوم برسم مجموعة من الخطوط عن طريق الأمر draw وكذلك طباعة كلمة Hello World مثل التالي.

```
draw.ellipse
draw.rectangle
draw.polygon
draw.line
draw.text((x, top), 'Hello', font=font, fill=255)
```

هناك مجموعة من الأمثلة في ملف examples أحدها يسمى py وهو يقوم بعرض صورة قطة على الشاشة وآخر اسمه animated.py يمكنك من تحريك نص على الشاشة، يمكنك بسهولة قراءة الكود المكتوب والتعديل فيه ليناسب إحتياجاتك.

