

天花板隨記

[Home](#)

Arduino筆記

Raspberry Pi筆記

Android筆記

電腦技術

科技新知

生活

TEXT TO SEARCH...

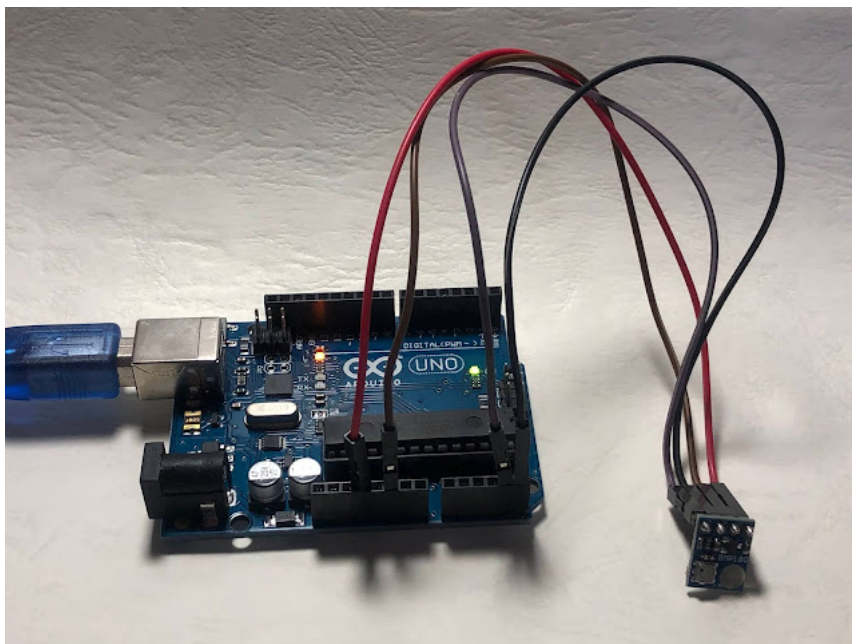
[Home](#) / [Arduino](#) / [Arduino筆記\(70\)：溫度氣壓傳感器模組 BMP180 \(GY68\)](#)

Arduino筆記(70)：溫度氣壓傳感器模組 BMP180 (GY68)

CEILING TSAI 星期一, 9月 30, 2019 ARDUINO

我有好幾個溫度的感測模組，如DHT-11、AM2320、DS18B20等，這幾個是測試溫溼度的感測模組。最近購買一個可以測量溫度跟氣壓的模組BMP180，這個模組跟前先實作的 [筆記\(59\)：大氣壓強度\(高度計\)感測模組](#) **GY-BMP280** 功能一樣具備溫度檢測和大氣監測兩種功能，可以根據測出的溫度和氣壓，計算出晶片所在的海拔高度。BMP280的功能及精度較BMP180佳，在功能上做了一些提升：

- 減小了體積，目前擁有業界最小封裝 $2.0 \times 2.5 \times 0.95 \text{ mm}^3$ 。
- 在測溫的精度上由 0.1°C 精確到 0.01°C 。
- 在海拔 $9000\text{m} \sim 500\text{m}$ 的壓強為 $300\text{hPa} \sim 1100\text{hPa}$
- 氣壓精度由 1pa 精確到 0.16pa 。
- 氣壓傳感器支持SPI和IIC通信接口。
- 運用氣壓計增強GPS定位或者配合IMU傳感器，實現三維導航。
- 傳感器功耗僅有 $2.7\mu\text{A}$ 。
- 提高了解析度和採樣頻率。



以下是維基百科對於氣壓的解釋：

氣壓的國際單位制是帕斯卡（或簡稱帕，符號是Pa），泛指是氣體對某一點施加的流體靜力壓力，來源是大氣層中空氣的重力，即為單位面積上的大氣壓力。在一般氣象學中人們用千帕斯卡（kPa）、或使用百帕（hPa）作為單位。測量氣壓的儀器叫氣壓表。其它的常用單位分別是：巴（bar，1 bar=100,000帕）和公分水銀柱（或稱公分汞柱）。在海平面的平均氣壓約為101.325千帕斯卡（76公分水銀柱），這個值也被稱為標準大氣壓。另外，在化學計算中，氣壓的國際單位是「atm」。一個標準大氣壓即是1 atm。1個標準大氣壓等於101325帕、1013.25百帕和1.01325巴，或者76公分水銀柱。氣壓的地區差別是氣象變化的直接原因之一。

見于气預起的 一個垂西的鐵量

請選取語言

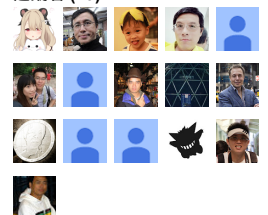
由「

小米情人節發
限量折價券

小米手環 6現省29
再折優惠價\$656
米電動刮鬍刀 五
省300領券再折價
\$1636up

追蹤者

追蹤者 (19)



追蹤





如果要查詢目前所在的相對海拔高度，可以到以下網站，輸入住址後，查詢海拔高度，再將這個網站查得的高度值，修改程式第五行的ALTITUDE變數。網址如下：<http://haiba.qhdi.com/>。



[GY-68 BMP180 溫度氣壓傳感器模組]

BMP180 是Bosch Sensortec 新推出的數字氣壓傳感器，性能非常高，可用於智能手機，平板電腦和運動設備等高級移動設備。像大多數壓力傳感器一樣，BMP180可以測量絕對壓力，這是設備看到的實際環境壓力，隨海拔高度和天氣而變化。GY-68 BMP180 溫度氣壓傳感器模組的規格如下：

- 壓力範圍：300~1100hPa (海拔9000米~-500米)
- 電源電壓：1.8V~3.6V (VDDA)，1.62V~3.6V (VDDD)
- LCC8封裝：無鉛陶瓷載體封裝 (LCC)
- 尺寸：3.6mmx3.8x0.93mm
- 低功耗：5μA，在標準模式
- 高精度：低功耗模式下，解析度為0.06hPa (0.5米)
- 高線性模式下，解析度為0.03hPa (0.25米)
- 含溫度輸出
- I2C介面
- 溫度補償
- 無鉛，符合RoHS規範
- MSL 1反應時間：7.5ms
- 待機電流：0.1μA
- 無需外部時鐘電路



網誌存檔

- 2021 (59)
- 2020 (92)
- ▼ 2019 (79)
 - 十二月 (4)
 - 十一月 (3)
 - 十月 (9)
 - ▼ 九月 (12)
 - Arduino筆記(71)：MPU-60521 三軸陀螺儀+三軸加速計
 - Arduino筆記(70)：溫度氣壓傳感器模組 BMP180 (GY68)
 - Arduino筆記(69)：TTP224 控制LED燈
 - Arduino筆記(68)：用4x4薄碼控制磁閥開關
 - Arduino筆記(67)：使用CP2102 Arduino設定HC-05藍牙模組
 - Arduino筆記(66)：讀取SD卡數據顯示在 ST7735 TFT-LCD顯示器(下)
 - Arduino筆記(65)：Adafruit 程式庫用於顯示器(下)
 - Arduino筆記(65)：Adafruit 程式庫用於顯示器(中)
 - Arduino筆記(65)：Adafruit 程式庫用於顯示器(上)
 - Arduino筆記(64)：ST7735 TFT-LCD顯示器(160 x 128 像素)
 - Arduino筆記：安裝 Arduino 庫(library)



[安裝BMP180 Library]

本實作需要安裝以下程式庫：

- [cyberp/BMP180](#)

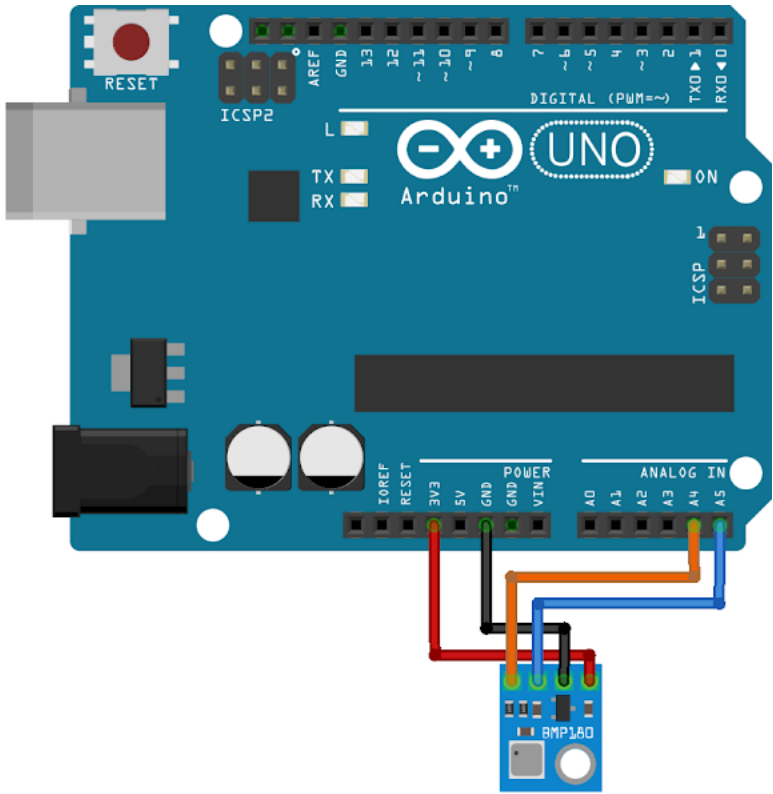
程式庫(Library)安裝方法請參考另一篇文章: [Arduino筆記：安裝 Arduino IDE 程式庫\(Library\)](#)。

[材料]

- Arduino Uno x 1
- GY-68 BMP180
- 連接線 x 4條

[接線與電路圖]

ARDUINO UNO	BMP180(GY-68)
5V(VCC)	VCC
GND	GND
A4(SDA)	SDA
A5(SDL)	SDL



fritzing

[程式]

在讀取壓力感測值之前，必須先取取溫度讀數，透過startTemperature()和getTemperature()取得溫度值，結果以攝氏度為單位。在取得溫度值後，就可以讀取壓力值。透過startPressure()和getPressure()取得感測值完，結果以毫巴 (mb) 或百帕斯卡 (hPa) 為單位。

如果要測得天氣狀況，則要消除海拔的影響，這要跟他地方公佈的壓力的海拔高度進行比較，使用sealevel()函數時，需要提供測量壓力的已知高度，如果要測量高度，則需要知道壓力在基準點的高度。這可以是平均海平面壓力，或是以前的壓力值，依照測得的結果加或減其准值，這早透過height()函數完成。

- ▶ [五月](#) (3)
- ▶ [四月](#) (4)
- ▶ [三月](#) (9)
- ▶ [2018](#) (11)
- ▶ [2017](#) (50)
- ▶ [2014](#) (20)
- ▶ [2013](#) (11)
- ▶ [2012](#) (11)
- ▶ [2011](#) (57)

標籤

[3d印表機](#) (5) [生活](#) (22) [其他](#) (1)
(3) [音樂/電影](#) (9) [旅遊](#) (51) [網路文章](#) (4) [餐廳推薦](#) (2) [環島](#) (2)
[記](#) (1) [Andriod](#) (25) [A](#)
(124) [Blender](#) (13) [Blogger](#) (
[ESP32](#) (15) [ESP32CAM](#) (3) [E](#)
(21) [FPGA](#) (2) [GPX](#) (2)
[MicroPython](#) (20) [Node-F](#)
[NodeMCU](#) (21) [Pico](#) (16)
[Raspberry Pi](#) (112) [s](#)




```

SFE_BMP180 pressure; // 建立一個物件名為 pressure
#define ALTITUDE 25 // 台灣西部海岸大約的海平面高度

void setup()
{
  Serial.begin(9600);
  Serial.println("REBOOT");

  // 初始設定
  if (pressure.begin())
    Serial.println("BMP180 init success");
  else
  {
    // 初始化錯誤，一般是連接問題
    Serial.println("BMP180 init fail\n\n");
    while(1); // 永久停在這裡
  }
}

void loop()
{
  char status;
  double T,P,p0,a;

  // 每 10秒鐘偵測一次

  // If you want sea-level-compensated pressure, as used in weather reports,
  // you will need to know the altitude at which your measurements are taken.
  // We're using a constant called ALTITUDE in this sketch:

  Serial.println();
  Serial.print("provided altitude: ");
  Serial.print(ALTITUDE,0);
  Serial.print(" meters, ");
  Serial.print(ALTITUDE*3.28084,0);
  Serial.println(" feet");

  // 如果要測量高度，而不是壓力，就需要提供已知的基本壓力

  // 取得溫度測量值

  // Start a temperature measurement:
  // If request is successful, the number of ms to wait is returned.
  // If request is unsuccessful, 0 is returned.

  status = pressure.startTemperature();
  if (status != 0)
  {
    // 等待感測完成
    delay(status);

    // 取得完整溫度測量，感測值存在變數T，函數回傳 1表示成功，0表示失敗

    status = pressure.getTemperature(T);
    if (status != 0)
    {
      // 顯示感測值
      Serial.print("temperature: ");
      Serial.print(T,2);
    }
  }
}

```

[實作結果]

```

COM11

BMP180 init success

provided altitude:  meters, 82 feet
temperature: 27.98 .C,      82.36 .F
absolute pressure: 1002.86 mb,      29.62 inHg
relative (sea-level) pressure: 1005.83 mb,      29.71 inHg
computed altitude: 25 meters, 82 feet

provided altitude:  meters, 82 feet
temperature: 27.96 .C,      82.34 .F
absolute pressure: 1002.86 mb,      29.62 inHg
relative (sea-level) pressure: 1005.84 mb,      29.71 inHg
computed altitude: 25 meters, 82 feet

provided altitude:  meters, 82 feet
temperature: 27.94 .C,      82.29 .F
absolute pressure: 1002.75 mb,      29.61 inHg
relative (sea-level) pressure: 1005.73 mb,      29.70 inHg
computed altitude: 25 meters, 82 feet

provided altitude:  meters, 82 feet

```



• Github: [LowPowerLab/SFE_BMP180](#)

Share on Facebook

Share on Twitter

Share on Google Plus

RELATED POSTS



0 Comments:

輸入您的留言...

發表留言的身分：

Google 帳戶

發布

預覽



較新的文章

首頁

較舊的文章

