

[Arduino 基礎入門應用]

作者: 徐昀湏 蔡潤光



1.Arduino 簡介

它可以讀取大量的開關和感測器信號,並且可以控制電燈、電機和其他各式各樣的物理設備;Arduino 也可以開發出與 PC 相連的周邊裝置,能在運行時與 PC 上的軟體進行通信。Arduino 的硬體電路板可以自行焊接組裝,也可以購買已經組裝好的模組,而程式開發環境的軟體則可以從網上免費下載與使用。

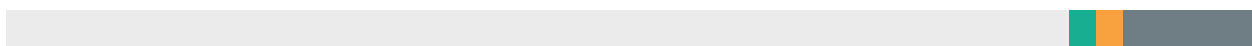
接下來,我們要為你重頭講解 Arduino 的硬體結構與設幾理念。

我們為你介紹的是 Arduino 人們最常使用的一種開發板 Uno 型號,這種開發板設計簡單功能強大,而且價格親民。無論是學習程式設計還是,電路開發都是一個好選擇。

當然 Arduino 還有需多種類,每一種都有它的特性和其特殊功能。各位如果好其可上 Arduin 官網上查詢<https://www.arduino.cc>



Arduino 維開源硬件,所以你可以依據 Arduino 官方網站上所提供的電路圖及規格圖打造屬於自己的開發板。

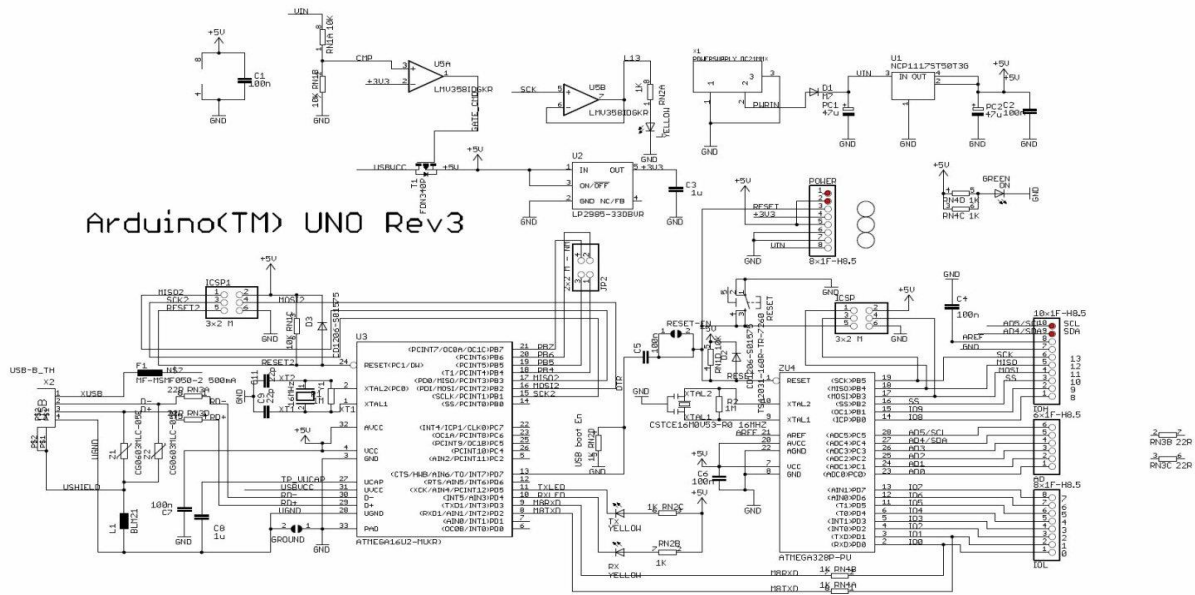


The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

Revision 2 of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode.

Revision 3 of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.



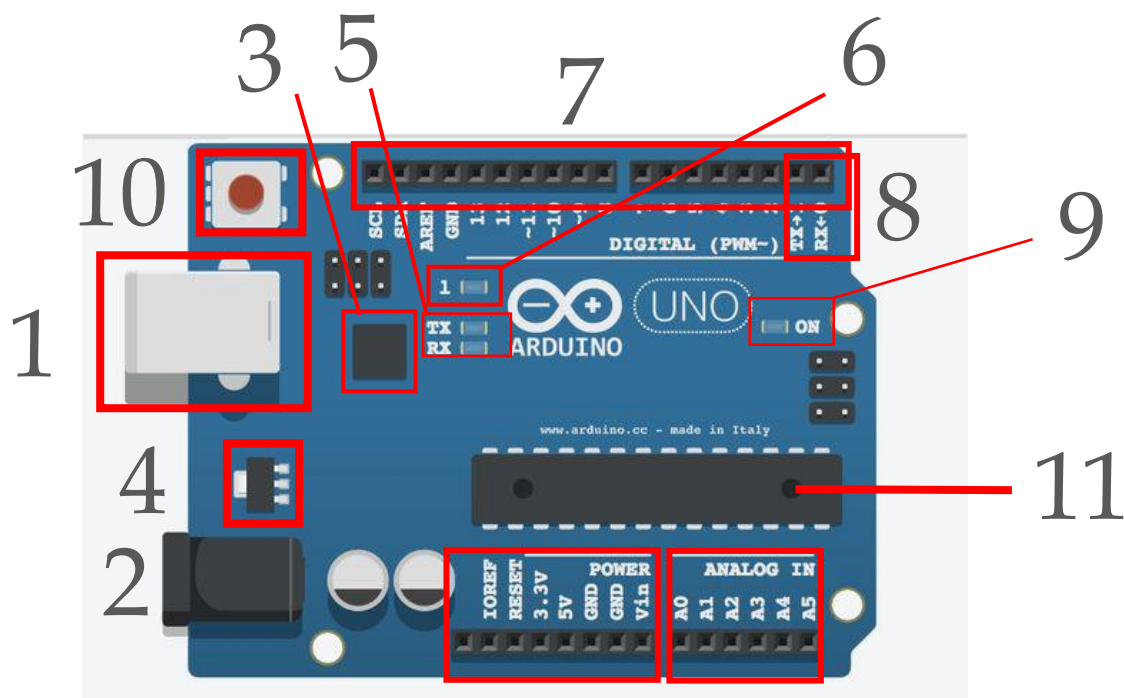
Reference Designs ARE PROVIDED "AS IS" AND "WITH ALL FAULTS. Arduino DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Arduino may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Arduino reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.

ARDUINO is a registered trademark.

以下為基本規格:

微控制器	ATMEGA328
工作電壓	5V
輸入電壓（推薦）	7-12V
輸入電壓（限制）	6-20V
數字 I/O 接腳	14（6 個提供 PWM 輸出）
模擬輸入接腳	6 支
EEPROM	1 KB（ATMEGA328）
震盪速度	16 MHz

腳位與插槽工能介紹:



- 1.支援 USB 接頭傳輸資料及供電(不需額外電源)。
- 2.輸入電壓 **Power Jack**，可以單獨使用，不需連接 USB 電腦時就可以直接接上 5V 變壓器，或是電池直接供電。（所以 1 和 2 只要選選擇一個就可以了）
- 3.**FTDI USB** 晶片，這是 USB 的 Client 端的晶片，透過這個晶片就可以跟電腦溝通連結傳遞資料。
- 4.**Voltage Regulator**:穩壓器，保持電壓的穩定。
- 5.LED 燈：用來顯示傳遞接受的資料 RX 和傳遞出去的資料 TX，如有資料進去跟出去的話這 LED 燈都會閃爍一下，並且這兩個 LED 燈，和區域 7 的接腳 0 和接腳 1 是相連結的。
- 6.LED 燈：接腳 13 是高電壓還低電壓。
- 7.數位接腳:由右到左那他分別是
 - 7-1：數位接腳 0 到數位接腳 13：你注意看一下印刷，有些字前面有～符號，例如說～10，這個意思是當它可以當成數位接腳 10，可以做 PWM 的資料輸出，所以呢這個符號的意思是說在這個板子上面，只有特別的接腳可以做 PWM 的資料輸出。
 - 7-2：GND: 接地接腳。

8.正負極接點。

9.LED 燈：電源顯示燈。

10.重新執行程式按鈕。

Microcontroller: 控制晶片。

電源控制的接腳：

11. [ATMEGA328](#) 晶片。

12. 有 6 支類比輸入腳，標記為 A0 到 A5。

13-1: Vin (電源輸入)

13-2: Gnd (接地)

13-3: Gnd (接地)

13-3: 5V (5V 電源輸出)

13-3: 3V3 (3V 電源輸出)

13-3: RESET (重新執行程式)

基礎電路注意事項:

Arduino 儲存庫中提供了 [ATmega16U2](#) (或 rev1 和 rev2 板中的 8U2) 固件代碼。

[ATmega16U2 / 8U2](#) 裝有 [DFU](#) 引導安裝控制程序，可通過以下方式激活它：

- 在 Rev1 板上：連接板背面的焊接跳線（在意大利地圖附近），然後恢復到 8U2。
- 在 Rev2 或更高版本的板上：有一個電阻將 8U2 / 16U2 HWB 線拉到地，使其更容易

進入 DFU 模式。

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available in the Arduino repository. The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then rese ing the 8U2.

On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

原文節錄自: <https://www.arduino.cc>

警告:

Arduino / Genuino Uno 有一個可復位的 **polyfuse**，保護開發板的 USB 端避免受短路和過大電流。雖然大多數開發板提供自己的內部保護，保險絲提供了額外的保護層。如果 USB 端口上施加了超過 500 mA 的電流，保險絲將自動斷開連接，直到短路或過載消除。

作者提醒:

並不是每一家製造廠商再其生產的開發板上都有做出完整的短路保護，如果你不確定你的電路是否正確請使用三用電表來測測看試過流，或是自行設計保護電路。

The Arduino/Genuino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed

原文節錄自: <https://www.arduino.cc>

3.Arduino 軟體介紹(下載&上傳程式)

搜尋"Arduino"並進到 Arduino 官網 網址: <https://www.arduino.cc>

The image shows two screenshots of the Arduino website. The first screenshot is the homepage at <https://www.arduino.cc>. The 'Download' link in the navigation bar is circled, and an arrow points to it with the text "按\"下載\"". The second screenshot is the 'Download the Arduino Software' page at <https://www.arduino.cc/en/Main/Software>. A large arrow points to the 'Windows installer' option, with the text "選擇自己的作業系統" above it. The page also displays the Arduino 1.8.0 version information and links to previous releases.

按"下載"

選擇自己的作業系統

Download the Arduino Software

ARDUINO 1.8.0

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. Refer to the [Getting Started](#) page for installation instructions.

Windows installer
Windows ZIP file for non admin install
Windows app [Get](#)

Mac OS X 10.7 Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM

Release Notes
Source Code
Checksums (sha512)

Connect. Collaborate. Create. [Learn more about the Create platform.](#)

Try out the new
Arduino Web Editor

ARDUINO SOFTWARE
HOURLY BUILDS

LAST UPDATE
29 December 2016 23:15:31 GMT

ARDUINO 1.0.6 / 1.5.x / 1.6.x
PREVIOUS RELEASES

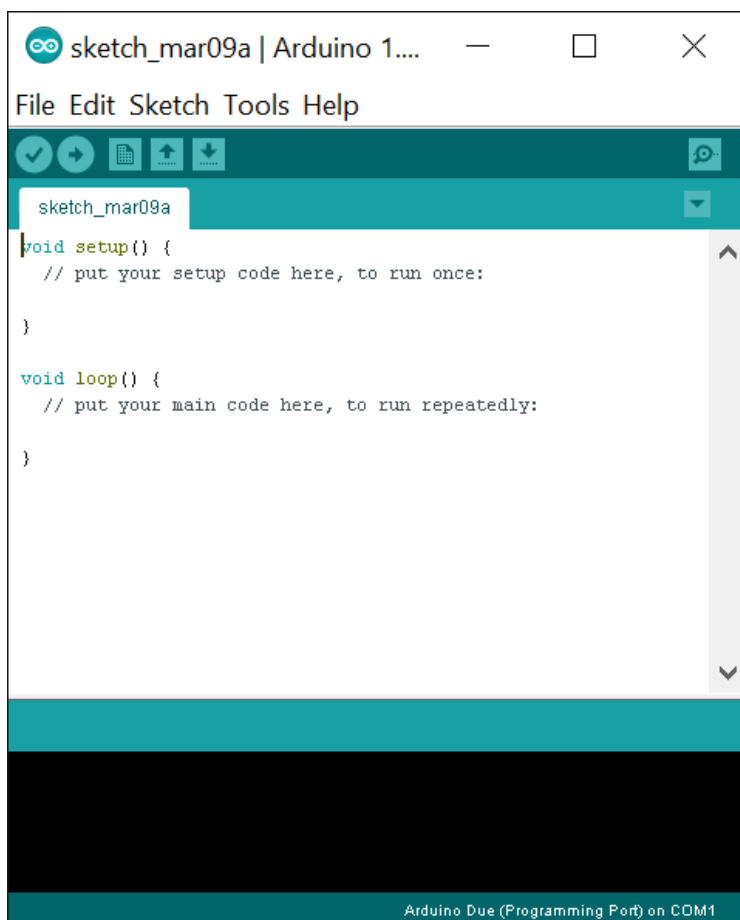


安裝完之後你的 Arduino IDE 基本上就可以使用了。

接下來，要帶大家從程式設計的角度認識 Arduino。

Arduino 程式設計入門

當你打開你的 Arduino IDE 之後，你會看到以下畫面：



那我們稍微介紹一下畫面中的按鈕：



Verity

檢查參數設定或引入程式是否產生錯誤。



Upload

程式進行編譯，將程式碼透過 USB 介面燒錄至 Arduino 控制板。



Save

腳本儲存。



Open

開啟腳本，顯示在同一頁面上。點擊不同腳本便顯示不同腳本。

接下來我們來介紹編寫程式的部分

首先我們會看到這段預設程式碼：

```
void setup()  
{  
  
  
}  
  
void loop()  
{  
  
  
}
```

setup(): 涵式內放的是設定程式的初始環境，像是設定接線的腳位，預設的套件初始質等等。每次值行程式時只會運行一次。

loop(): 涵式內放的是程式主體的應用，它的功能類似於 C/C++ 中的 **main()** 涵式。基本上我們可以把它視為程式主體。

那麼，再兩個涵式前加上的 **void** 意思是無回傳的意思，也就是說這兩個涵式並不會回傳任何值。這是可以被理解的，因為這兩個涵式為整個程式的主體，其它涵式的回傳值幾乎都會回到 **loop()** 涵式中。那麼假如 **loop()** 有回傳值，它並沒有一個接收回傳值的一個位置，這會造成邏輯上的錯誤。

那麼 **setup()** 涵式的作用為定義而不是運算所以它也不需要回傳什麼東西給 **loop()** 涵式。

接下來我們看看一個經典的範例：

範例：

讀取光敏電阻

所需準備的材料：

1. 光敏電阻 *1
2. 電阻：1K 歐姆 棕黑紅 *1
3. 麵包板 *1
4. 單心線 *N
5. Arduino Borad *1

我們先以程式碼的部分為主，先忽略掉所有原件組裝及原理，先直接來看看程式碼的部分。

程式碼：

```
int potPin = 3;
int val = 0;
void setup(){
  Serial.begin(9600);
}
void loop(){
  val = analogRead(potPin);
  Serial.print(val);
}
```

或許有些已經會寫 C/C++ 的同學會好奇為何我們不需要引入任何的標頭，因為在 Arduino IDE 中已經將所有的 library(涵式運算程式碼)植入了。因為 Arduino 的程式結構相對簡單專一所以我們可以用如此便捷的方法專寫程式。當然你也可以自己專寫自己的標頭檔。細節請各位自行上網查詢，當你已經寫好自己的標頭檔或是下載了一個標頭檔，你只需要把它裝進一個資料夾中命名為後扔進 C 槽裡 Arduino 的檔案夾中的 library 中重新啟動你的 Arduino IDE 便可以載入你自己的標頭檔：

```
#include <mylibrary.h>
```

注意:在 Arduino 中大小寫是有差的，Serial.print 不等於 serial.print

輸入與輸出:

已下為比較專業的部分，我們主要要探討電路板上的 I/O 與函數的關係。

首先，數位 I/O 接腳:

14 支數位 I/O 接腳 可以當作 input (輸入)使用，也可以當作 output(輸出) 使用，使用方法是透過 pinMode(), digitalWrite(), and digitalRead() 這幾個函式。這 14 支數位 I/O 接腳，其中幾支腳有特殊的功能:

Serial 通訊	0(RX) 和 1 (TX) 這兩支腳。用來接收(RX)與傳輸(TX) TTL 訊號的序列資料。這兩支腳也連接到 USB Converter 晶片中。
外部中斷	2 和 3 這兩支腳。這兩支腳可以利用外部事件觸發中斷。詳細內容請參考 attachInterrupt() 函式。
PWM	3, 5, 6, 9, 10 和 11 共六支腳。透過 analogWrite() 函式可以提供 8-bit 的 PWM 輸出。
SPI	10 (SS), 11 (MOSI), 12 (MISO) 和 13 (SCK) 這四支腳。這四支腳搭配 SPI Library 可提供 SPI 序列通訊。
LED	13。內建一顆 LED，當 pin 腳為 HIGH 時，LED 打開，當 pin 腳為 LOW 時，LED 關閉。

類比輸入 接腳:

Arduino Uno 有 6 支類比輸入腳，標記為 A0 到 A5，每支腳都可提供 10 位元的解析 (即 1024 種不同的數值)。這些腳位所用的參考電壓預設為 0 到 5V，不過參考電壓也是可以更改的，方法是透過 AREF 腳和 analogReference()

I2C	4 (SDA) 和 5 (SCL) 這兩支腳。透過 Wire library 可以提供 I2C 通訊。
------------	---

其它:

AREF	類比輸入的參考電壓，搭配 analogReference() 函式一起使用。
Reset	當 Reset 腳為 LOW 時，微控制器會重置。

I/O 相關函式集

1. 數位(Digital)

pinMode ()

這個函式提供引入腳位，它的語法為：

pinMode (腳位，模式)

腳位通常為開發板上的 14 個腳位。

模式的部分主要分成 INPUT，OUTPUT 或 INPUT_PULLUP

INPUT_PULLUP 的意思是使內部上拉電阻。

digitalWrite ()

這個函式引入高和低數字引腳。(類似布林函數)

如果該引腳被配置為與輸出 pinMode ()，其電壓將被設定為相應的值：5V（或 3.3V 板 3.3V）為 HIGH 時，0V（地）為低。其值再電路中為 1 或 0。

其語法為：

digitalWrite (腳位，值)

腳位為搭配使用的 pinMode()函數腳位，值為 HIGH 或 LOW。

digitalRead ()

讀取指定的數字引腳的價值，無論是高還是低。

digitalRead (腳位)

腳位:任何你想讀取的腳位皆可。

回傳值為 HIGH 或 LOW

2. 模擬(Analog)

analogRead ()

從指定的模擬引腳讀取值。Arduino 板包含 6 個通道（Mini 和 Nano 上的 8 個通道，Mega 上的 16 個通道），10 位模數轉換器。這意味著它將在 0 和 5 伏之間的輸入電壓映射到 0 和 1023 之間的整數值。這產生讀數之間的分辨率：每單位 5 伏/ 1024 單位或.0049 伏（4.9mV）。輸入範圍和分辨率可用來改變 analogReference ()。

讀取模擬輸入大約需要 100 微秒（0.0001 秒），因此最大讀取速率為每秒 10,000 次。

其語法為：

analogRead(腳位)

作者的話：其實我們很少用的這個涵式

analogReference ()

配置用於模擬輸入的參考電壓（即用作輸入範圍頂部的值）。選項是：

DEFAULT：默認模擬參考值為 5V（在 5V Arduino 板上）或 3.3 伏（在 3.3V Arduino 板上）

內部：一個內置基準，等於在 1.1 伏 ATmega168 的或 ATmega328 和 2.56 伏的 ATmega8 的（*不可用 Arduino 上兆*）

INTERNAL1V1：內置 1.1V 參考（*Arduino 的兆豐僅*）

INTERNAL2V56：內置 2.56V 基準（*Arduino 的兆豐僅*）

外部：適用於 AREF 引腳（電壓 0~5V 只）作為參考。

其語法為：

analogReference (類型)

類型：哪種類型的引用（默認值，內部 INTERNAL1V1，INTERNAL2V56，或外部）。

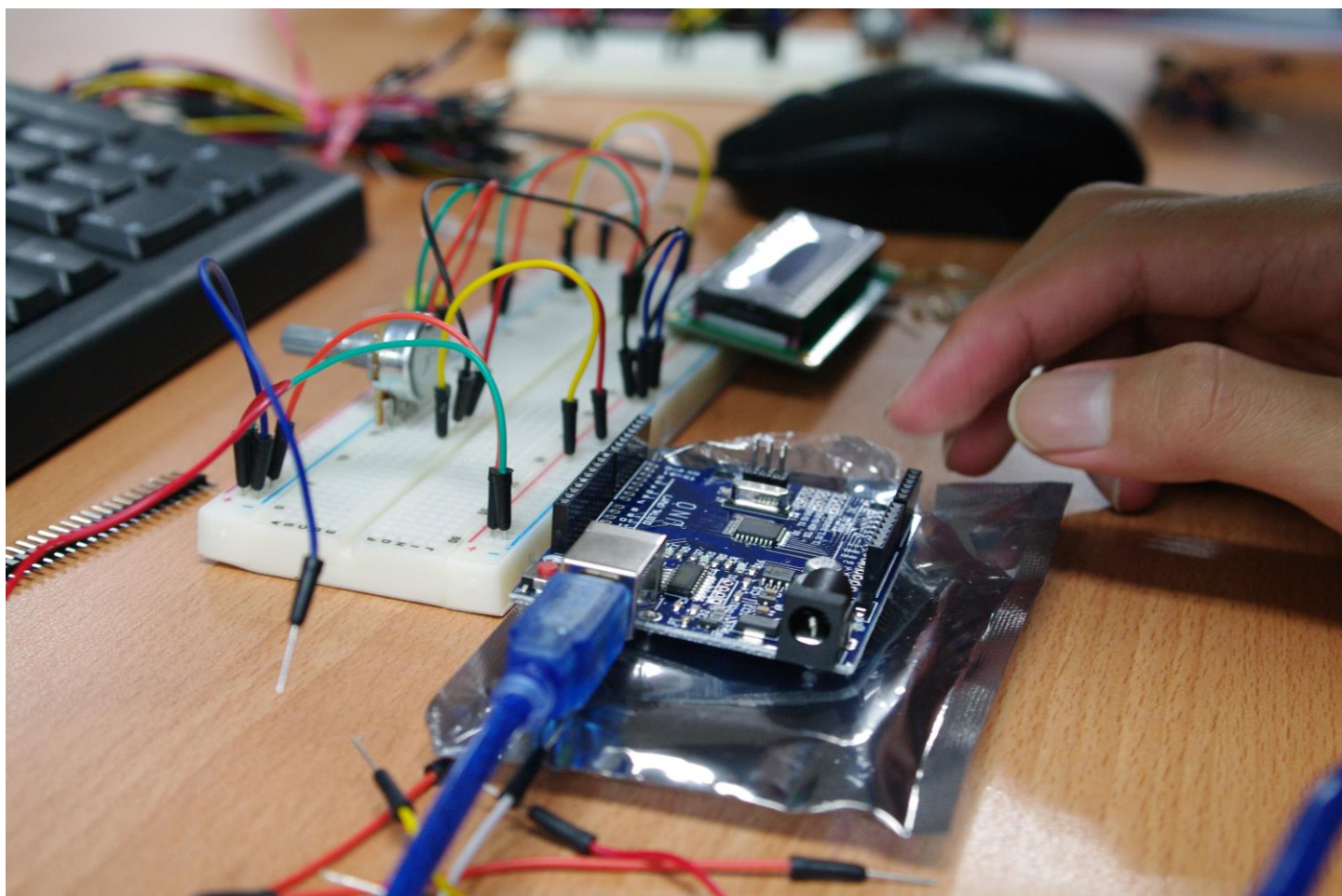
注意：

更改模擬參考值後，analogRead () 的前幾個讀數可能不準確。

警告:

對 AREF 引腳的外部參考電壓不要使用小於 0V 或大於 5V 的任何電壓！如果在 AREF 引腳上使用外部參考，在調用 `analogRead()` 之前，必須將模擬參考設置為 **EXTERNAL**。否則，將會使有效參考電壓（內部產生的）和 AREF 引腳短路，可能會損壞 Arduino 板上的微控制器。

或者，您可以通過 5K 電阻將外部參考電壓連接到 AREF 引腳，從而允許在外部和內部參考電壓之間切換。請注意，電阻將改變用作參考的電壓，因為 AREF 引腳上有一個內部 32K 電阻。兩者用作分壓器，因此，例如，通過電阻器施加的 2.5V 將在 AREF 引腳處產生 $2.5 * 32 / (32 + 5) \approx 2.2V$ 。



這本教學手冊就在此告一段落了，喜望你帶著滿滿的收穫回去~~

http://newsletter.ascc.sinica.edu.tw/news/read_news.php?nid=2782

<https://www.arduino.cc>

更多請看

<https://www.tcsshcsc.org>

