

創客基地 - 物聯網課程

Arduino 創客入門與應用

Day1: 2018.06.15

<https://reurl.cc/A1n18>

陳信嘉

shinjia168@gmail.com



上課檔案下載



<https://reurl.cc/1QjQY>

課程重點

物聯網

Arduino 基礎入門

會使用 Arduino

程式設計

程式設計 (coding)

編寫原始程式碼

創客

創意應用及實作

完成創作作品

學會使用 Arduino 及程式



元件使用及程式設計

輸出：LED 的閃爍



輸入：按鈕及旋鈕

多顆 LED (更多)

多種顏色的 LED

各種顯示的變化



創作實際作品

教材參考



FLAG`S 創客·自造者工作坊 10+ 實驗(「Arduino 超入門: 創客·自造者的原力」書+實驗套件)

作者：章奇煒

建議售價：699元

優惠價：**90折 629元(省 70 元)**

色彩：全彩

ISBN：471-294-675-013-5

出版日期：2016/1/26

主題學習

<https://www.flag.com.tw/shop/showbook.asp?bokno=F6789A>

(內容介紹、程式範例下載)

內容章節

第 1 章：Arduino 快速入門

第 2 章：看懂電子電路

第 3 章：Arduino 的數位輸出 / 輸入

第 4 章：Arduino 的類比輸出 / 輸入

第 5 章：用序列埠 (Serial Port) 與 PC 通訊

第 6 章：三色 LED 的控制

第 7 章：LED 排燈

第 8 章：光線感知 - 做一個自動照明系統

第 9 章：溫度感知 - 使用熱敏電阻監測水溫

第 10 章：潮濕感知 - 使一個花草澆水警示系統

第 11 章：使用七段顯示器 - 做一個按鈕計數器

安裝與環境設定

下載 Arduion

安裝

執行主程式

連接 Arduino 板子

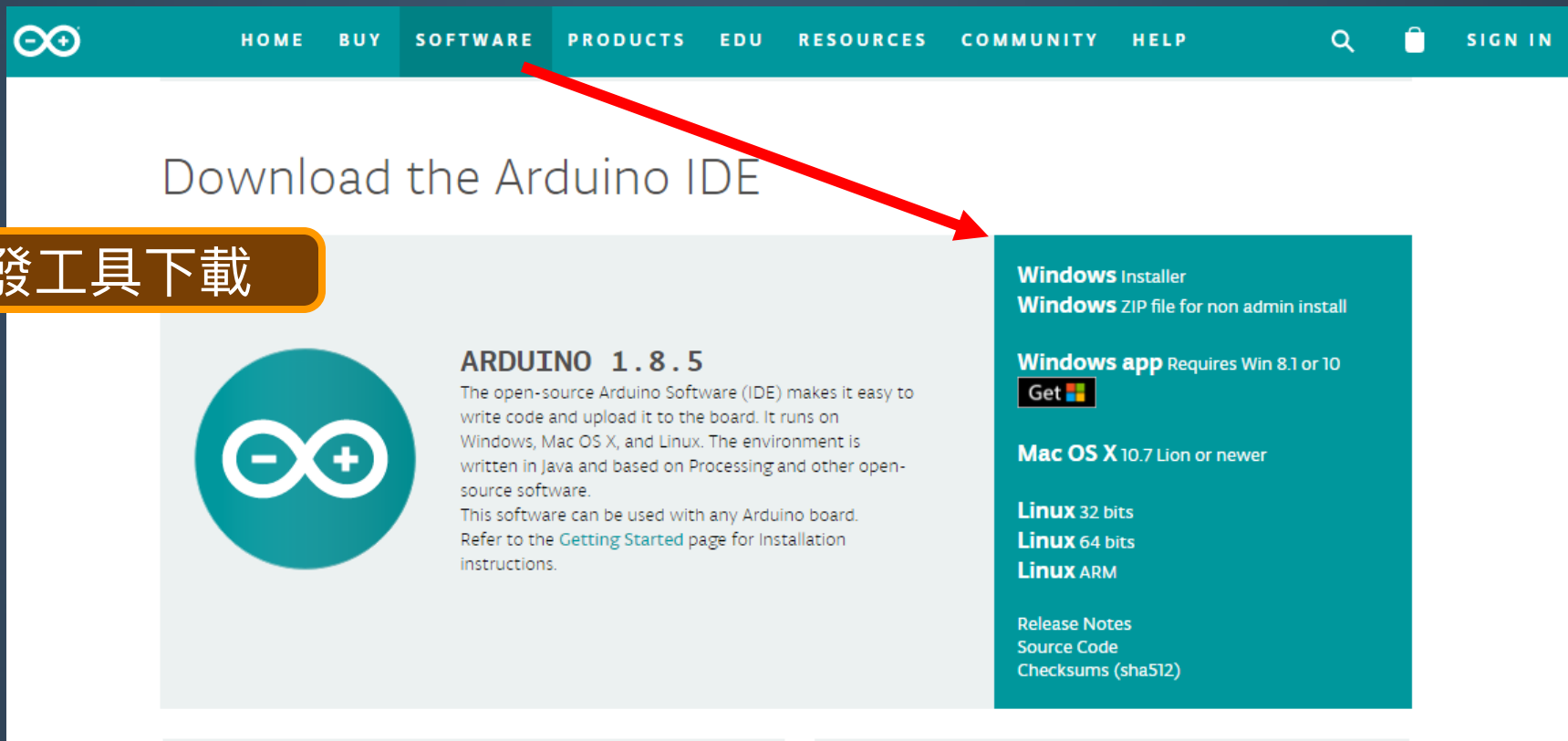
確認驅動程式

執行主程式

設定 板子 及 序列埠

Arduino 官方網站

<https://www.arduino.cc/>



The screenshot shows the Arduino website's 'Download the Arduino IDE' page. The top navigation bar includes links for HOME, BUY, SOFTWARE, PRODUCTS, EDU, RESOURCES, COMMUNITY, and HELP. A red arrow points from the 'SOFTWARE' link to the download options on the right. The main content area features the Arduino logo and the text 'ARDUINO 1.8.5'. Below this, it states: '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 Requires Win 8.1 or 10
[Get](#)

Mac OS X 10.7 Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM

[Release Notes](#)
[Source Code](#)
[Checksums \(sha512\)](#)

開發工具下載

Ch 01: Arduino 快速入門

- 1-1. 購買 Arduino 控制板及周邊零件
- 1-2. 下載 Arduino 程式開發環境
- 1-3. Arduino 程式基本架構
- 1-4. Arduino 的供電

Arduino UNO 控制板介紹

程式下載及安裝

Arduino 板子和電腦相連

開啟範例程式

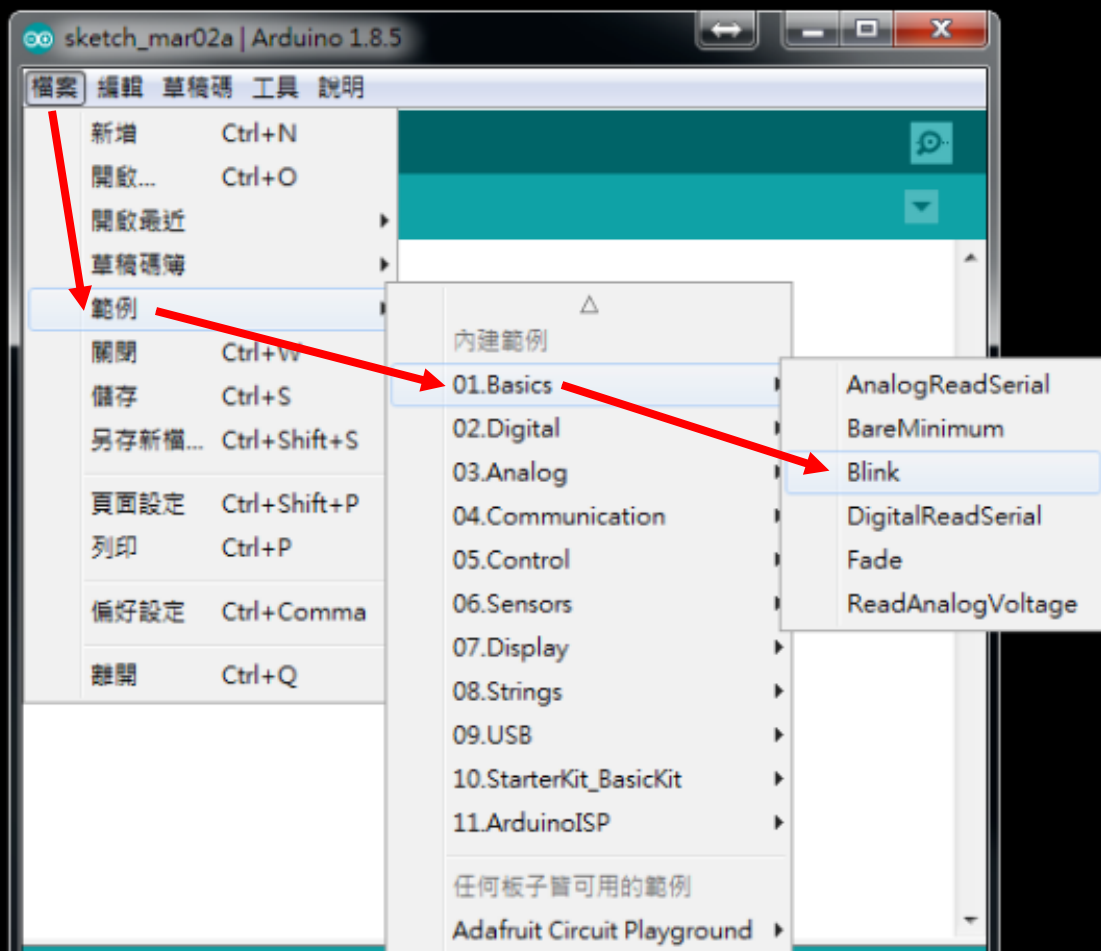
程式上傳 Arduino 及測試

板子：Arduino/Genuino Uno
序列埠：依各人環境而定
(可於裝置管理員中確認)

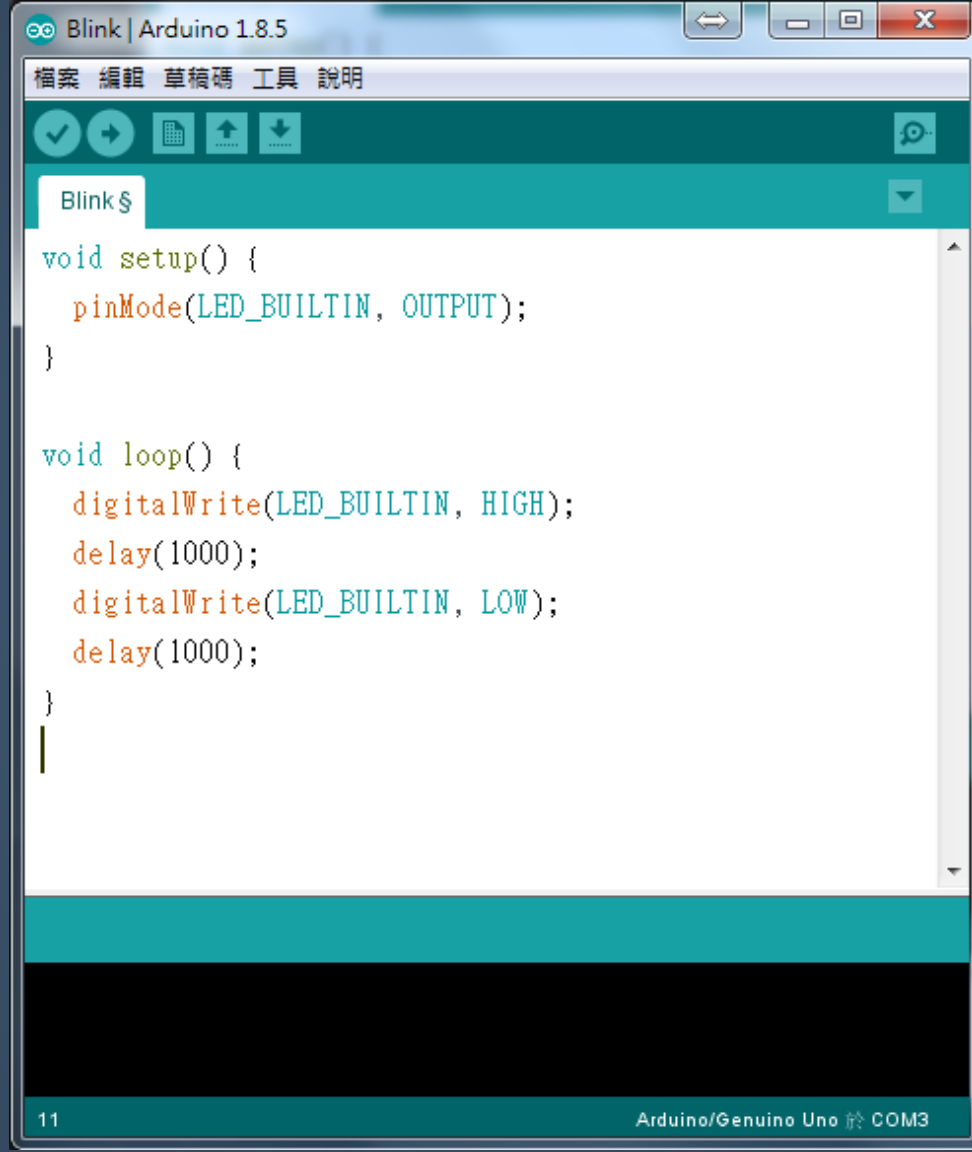
Arduino 程式 (簡化的 C 語言)

```
void setup()  
{  
  // put your setup code here, to run once:  
}  
  
void loop()  
{  
  // put your main code here, to run repeatedly:  
}
```

開啟官方範例 - Blink



程式解說



範例程式 lab1-1

```
void setup() { //"{"表示setup() 函式由此開始
    pinMode(13, OUTPUT); //將第 13 號腳位設為 OUTPUT 模式
} //"}"表示函式到此結束

void loop() { //"{"表示 loop() 函式由此開始
    digitalWrite(13, HIGH); //將高電位輸出到第 13 號腳位
    delay(500); //使程式暫停 0.5 秒, 維持在上一行所執行的狀態
    digitalWrite(13, LOW); //將低電位輸出到第 13 號腳位
    delay(500); //使程式暫停 0.5 秒, 維持在上一行所執行的狀態
} //"}"表示函式到此結束
```

更多練習

- 讓閃礫速度變快
- 讓閃礫節奏有變化
- -----
- 指定顯示腳位為參數
- 還有哪些可以參數化？

Ch 02: 看懂電子電路

2-1. 電壓、電流、電阻

2-2. LED 燈

2-3. 麵包板與單心線

2-4. 電子迴路

2-5. 接地

認識麵包板的線路

認識 LED

LED 的長為正，要注意

電線使用的習慣

紅線為供電，黑線接地

Ch 03: Arduion 的數位輸出/輸入

Lab 3-1. 數位輸出 - 閃爍 LED

Lab 3-2. 數位輸入 - 用按鈕控制 LED

認識 Arduino 的接腳

指定 LED 的閃爍

按壓開關

注意開關的接法

輸入和輸出的程式

按鈕控制 LED 的閃爍

範例程式 lab3-1

```
void setup() {  
    pinMode(7, OUTPUT); //13 改成 7  
}  
  
void loop() {  
    digitalWrite(7, HIGH); //13 改成 7  
    delay(500);  
    digitalWrite(7, LOW); //13 改成 7  
    delay(500);  
}
```

範例程式 lab3-2

```
const int button = 4; //宣告 button 為整數常數，並將其值設為 4(數位 pin4)
int btVoltage = 0; //將按鈕電位 (btVoltage) 設為整數變數，初始值為 0

void setup() {
  pinMode(LED_BUILTIN, OUTPUT); //將內建 LED 腳位設為輸出模式
  pinMode(button, INPUT); //將 button 腳位設為輸入模式
}

void loop() {
  btVoltage = digitalRead(button); //讀取按鈕腳位目前的電位值

  if (btVoltage == HIGH){ //若按鈕狀態為高電位
    digitalWrite(LED_BUILTIN, HIGH); //內建 LED 腳位 HIGH, LED亮起
  }
  else{
    digitalWrite(LED_BUILTIN, LOW); //LED 腳位會接收到低電位熄滅
  }
}
```

更多練習

- 結合 Lab3-1 及 Lab3-2
讓按鈕按下後，LED 亮
- 讓按鈕功能相反，按下後 LED 滅，放開後為亮
- -----
- 讓 LED 為閃爍狀態 (按鈕按下時)

Ch 04: Arduion 的類比輸出/輸入

Lab 4-1. 類比輸出 - 使用 PWM 控制 LED 亮度

Lab 4-2. 類比輸入 - 用可變電阻調整 LED 亮度

數位 (digit) 與 類比 (analog)

PWM 模擬成類比

Arduino 的類比腳位

for 迴圈程式

可變電阻的使用

注意『~』符號

3~, 5~, 6~, 9~, 10~, 11~

範例程式 lab4-1

```
const int led = 3; //宣告整數常數 led, 並將其值設為 3

void setup() {
    //setup()內不須寫入程式碼
}

void loop() {
    for (int fadeValue = 0 ; fadeValue <= 255; fadeValue =fadeValue + 15)
        //設定後方 {} 內 fadeValue 變數的初始值、條件、每執行一次的變化量
        {
            analogWrite(led, fadeValue); //將 PWM 調整後的電壓值輸出到指定的 PWM 腳位
            delay(100); //使程式停止在上一行的階段 0.1 秒
        }

    for (int fadeValue = 255 ; fadeValue >= 0; fadeValue =fadeValue - 15)
        //設定後方 {} 內 fadeValue 變數的初始值、條件、每執行一次的變化量
        {
            analogWrite(led, fadeValue); //將 PWM 調整後的電壓值輸出到指定的 PWM 腳位
            delay(100); //使程式停止在上一行的階段 0.1 秒
        }
}
```

範例程式 lab4-2

```
const int variable = A2; //宣告 variable 為整數常數，並將其值設為 A2
const int led = 9; //宣告 led 為整數常數，並將其值設為 9

void setup() {
    //setup()內沒有程式碼
}

void loop() {
    int sensorValue = analogRead(variable);
    //analogRead(variable) 由類比輸入腳位讀取可變電阻的電壓值，
    //並傳回介於 0~1023 的整數值來對應原本的電壓值
    //宣告感測數值變數為整數變數，其值等於 analogRead(variable) 的傳回值。
    analogWrite(led, sensorValue/4);
    //使用 analogWrite() 將 sensorValue 輸出到 PWM 腳位，點亮 LED
    //由於 analogWrite() 函式只接受 0~255 之間的數值，
    //所以必須將 sensorValue 除以 4
    delay(150); //延遲程式執行 0.15 秒
}
```

更多練習

- 使用 `map()` 函式換算
- 利用旋鈕調整 LED 閃爍的快慢
- 利用兩顆旋鈕，一個調整快慢，一個調整亮度

