

110新泰創資

2022.1.24

The Fun Theory

<https://www.youtube.com/playlist?list=PL9A7C183D5B9FF6DE>

- 音樂樓梯：<https://youtu.be/SByymar3bds>
- 垃圾桶：<https://youtu.be/qRgWttqFKu8>
- 資源回收：https://youtu.be/zCt_MzsnIUk
- 天橋：<https://youtu.be/sWYFaPDHCqs>
- 跳跳樓梯：<https://www.youtube.com/watch?v=CWwee62DW3U>
- 歡樂腳踏墊：<https://www.youtube.com/watch?v=fLy1mz0IUP0>
- 人行道：<https://www.youtube.com/watch?v=bHLgSfxz6bQ>
- <https://www.youtube.com/watch?v=Ox-9eOc3bQU>
- Youtube搜尋關鍵字：**Fun Theory**、**Arduino project**、**Creative ideas**

The arduino project

- [TOP 10 Arduino projects of 2019](#)
- [5 AMAZING Arduino project DIY 0-7'00''](#)

- [7 INCREDIBLE Projects with a simple FAN](#)
- [Top 10 Arduino Projects 2020 | Mind Blowing Arduino School Projects](#)
- [15 Great Arduino Projects for beginners](#)
- [Top 10 Arduino Projects For Beginners in 2019](#)
- [Top 10 Arduino Projects](#)
- [Arduino project 自動拍球機](#)
- [2016 TOP15 Arduino 創客專案](#)

[2 Creative ideas from Arduino](#)

[3 Creative ideas from Arduino](#)

[Amazing DIY idea with Arduino](#)

康橋國際學校：

[2018 Arduino學生專題](#)

[2017 Arduino學生專題](#)

[2019 Arduino學生專題1](#)

[2019 Arduino學生專題2](#)

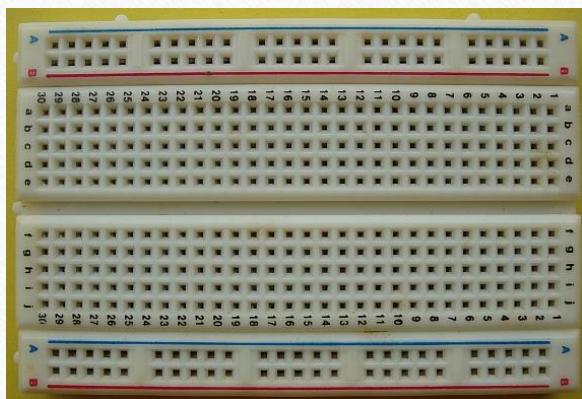
[2019 Arduino學生專題3](#)

上課之前

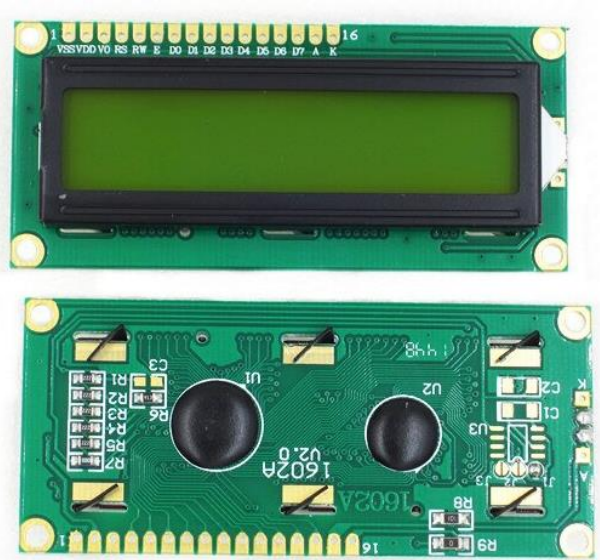
- 外部感應器學習重點在它可以做甚麼？可以應用在哪些方面？至於怎麼接線、怎麼寫程式？老師會幫你解決。

硬體介紹

麵包板



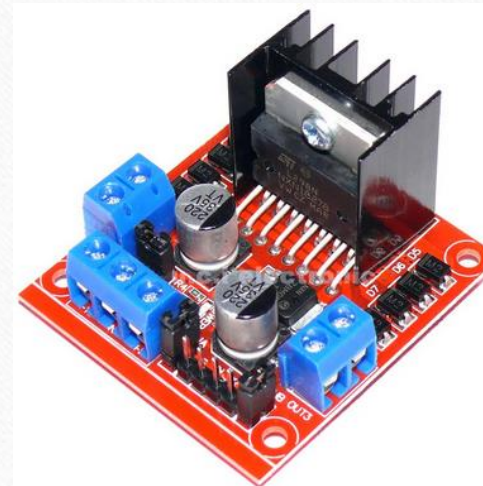
SG90 伺服馬達 舵機



LCD 1602A 16*2 藍底白字
學校這批LCD的配置位址是0x27



HC-SR04 超音波測距模組

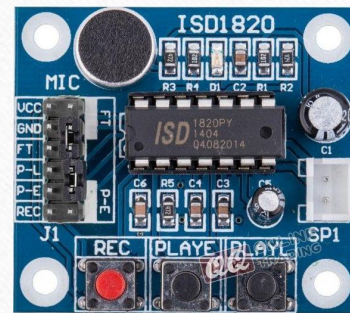


L298N馬達驅動板

硬體介紹

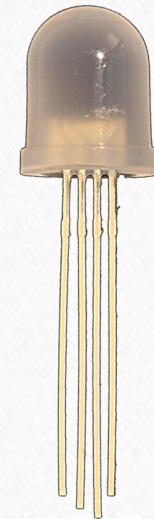


DHT11溫溼度感應器模組

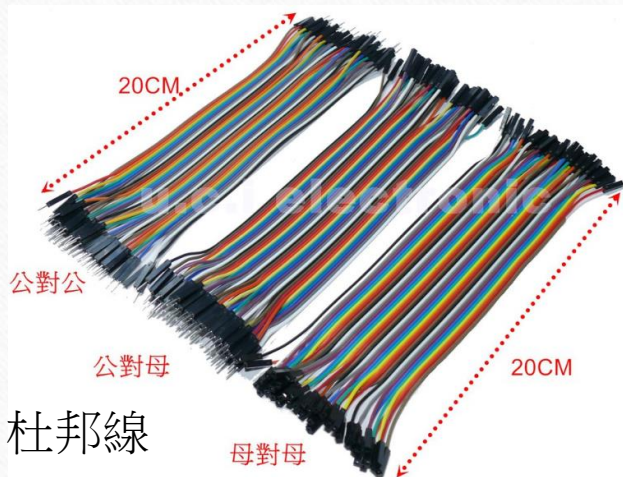


ISD1820簡易錄放音模組

單色LED



三色LED



杜邦線

其他感測器、執行器模組

- 水平感測器
- 溫度
- 聲音
- 顏色
- 人體
- XY搖桿
- 極限開關
- 土壤濕度
- 雨滴
- 步進馬達
- L9110馬達控制器
- 紅光雷射

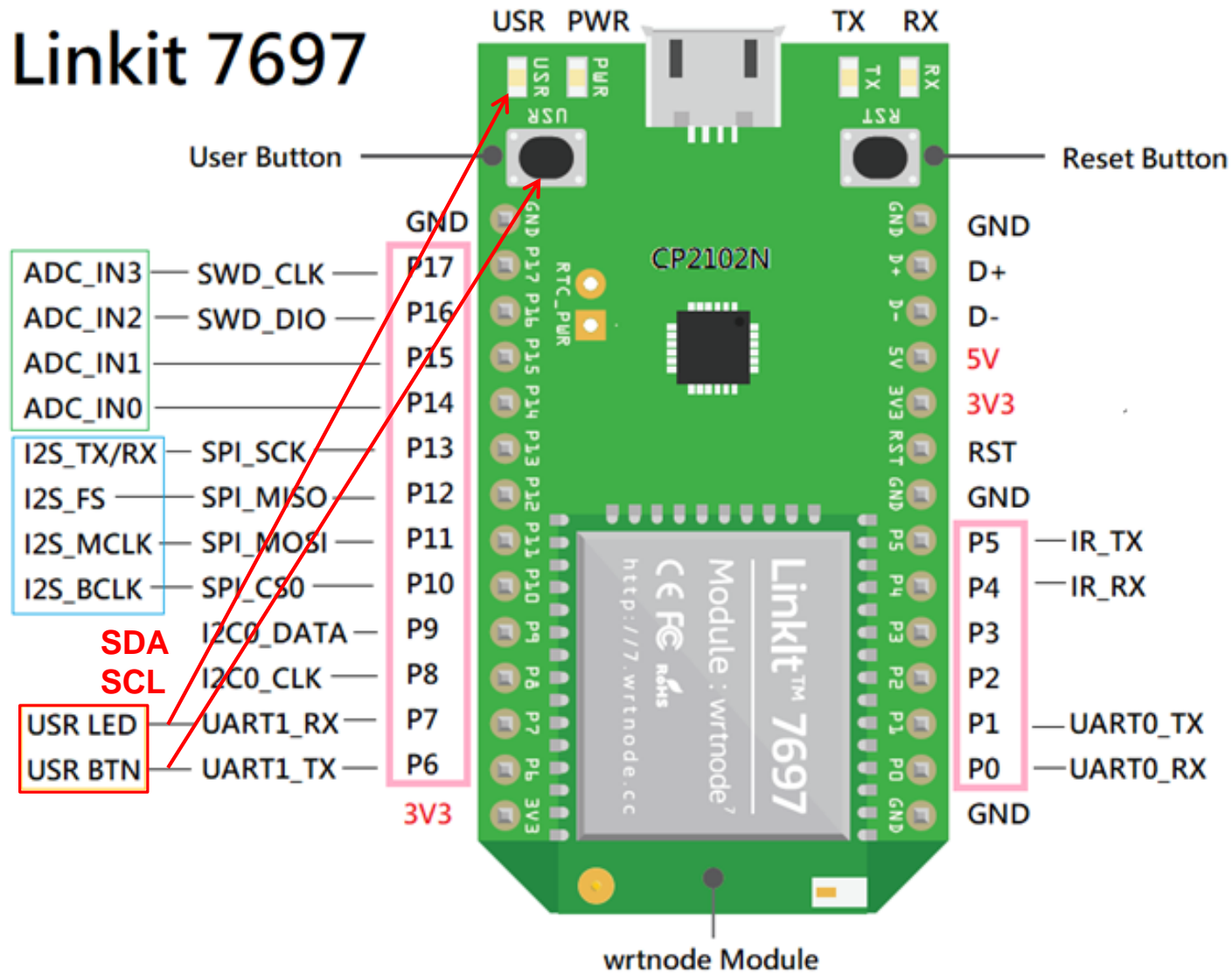
LinkIt 7697 驅動程式下載及安裝

名稱	修改日期	類型	大小
電腦科學LinkIt-設計物聯網應用	2022/1/24 上午 0...	檔案資料夾	
BlocklyDuino3.zip	2022/1/22 下午 1...	zip Archive	715,096 KB
CP210x_Windows_Drivers.zip	2021/2/8 上午 12...	zip Archive	7,001 KB

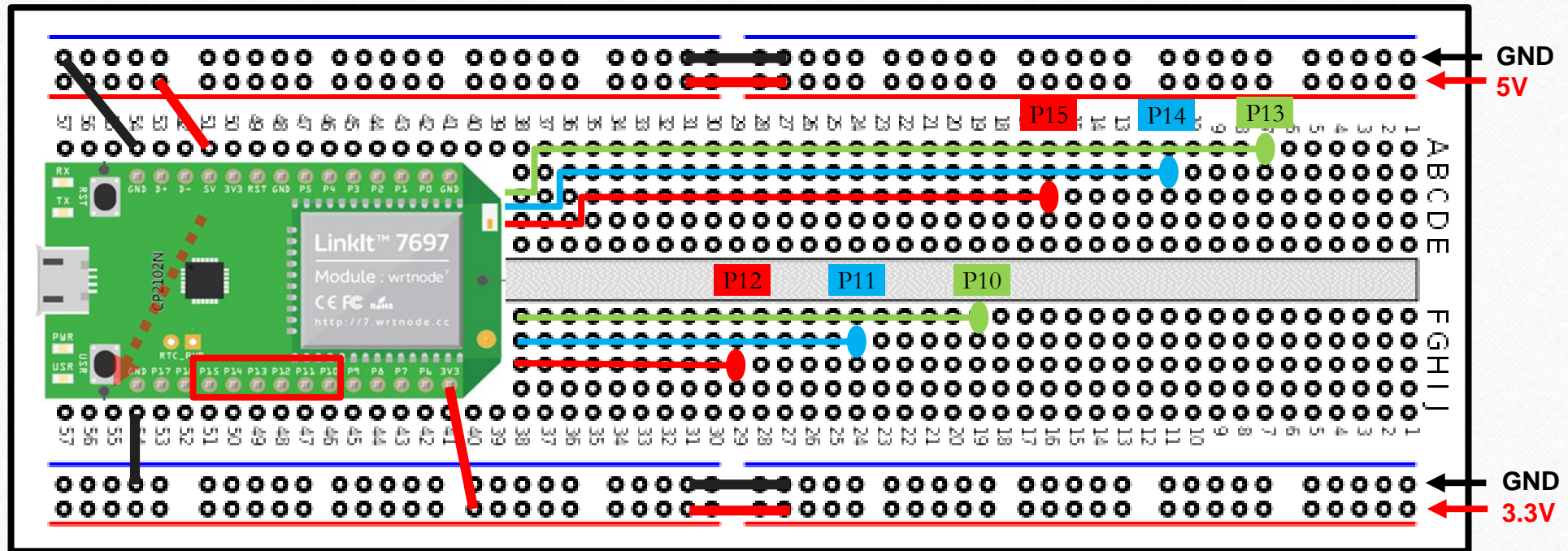
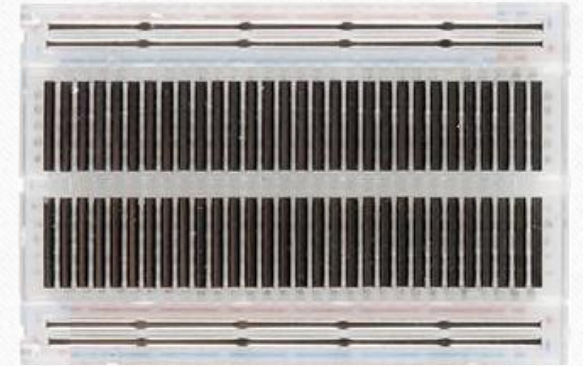
- 7697 需要安裝 USB 驅動程式才能連接到電腦，光碟片內有，僅需安裝1次，以後都可以使用。原安裝程式也可以刪除。
- LinkIt 7697使用Blockly Duino程式，光碟片內的BlocklyDuino3.zip 程式是BlocklyDuino Editor 3.0.312b的版本 + Arduino、ESP32及ESP8266核心函式庫 + 法蘭斯元件。請解壓縮到磁碟機的根本目錄。

注意：請務必將 BlocklyDuino3 資料夾複製到磁碟機的根本目錄，並避免以中文路徑命名而產生的錯誤。

P6、P7為傳送程式資料的腳位，建議上傳程式時不要接感測器，避免出現錯誤，上傳完成後再插上。

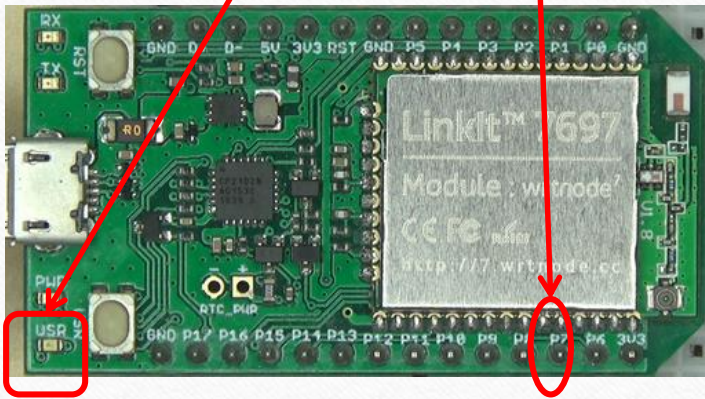


麵包板800孔



第1個小程序

內建LED燈(P7)



存檔避免以中文命名

BlocklyDuino Editor F1

Linkit 7697 COM6

積木 XML ARDUINO

數位寫入 Pin 0 - UART 狀態 高電位

數位讀取 Pin 0 - UART

內建 LED 狀態 高電位

讀取內建 LED 狀態

高電位

數位寫入 Pin 0 狀態 高電位

數位讀取 Pin 0

數位讀取 (預設上拉電阻) Pin 0

開發板 7697 初始化 燒錄狀態

重複執行

內建 LED 狀態 高電位

延遲毫秒 1000

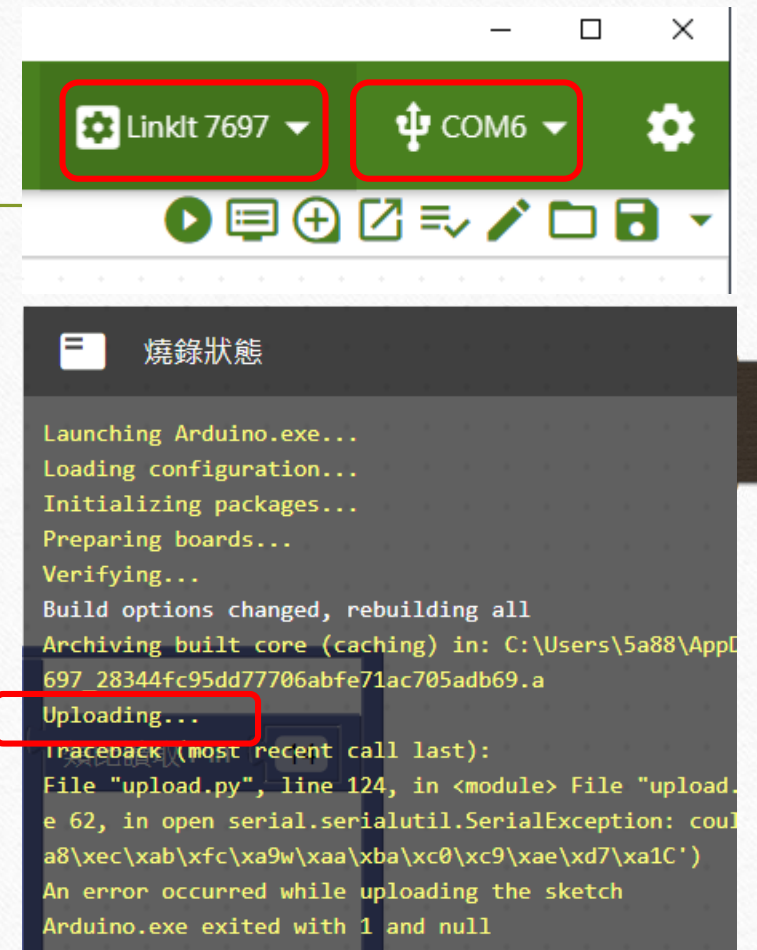
內建 LED 狀態 低電位

延遲毫秒 1000

檔案名稱: [無已載入的文件]

BlockyDuino 燒錄失敗原因

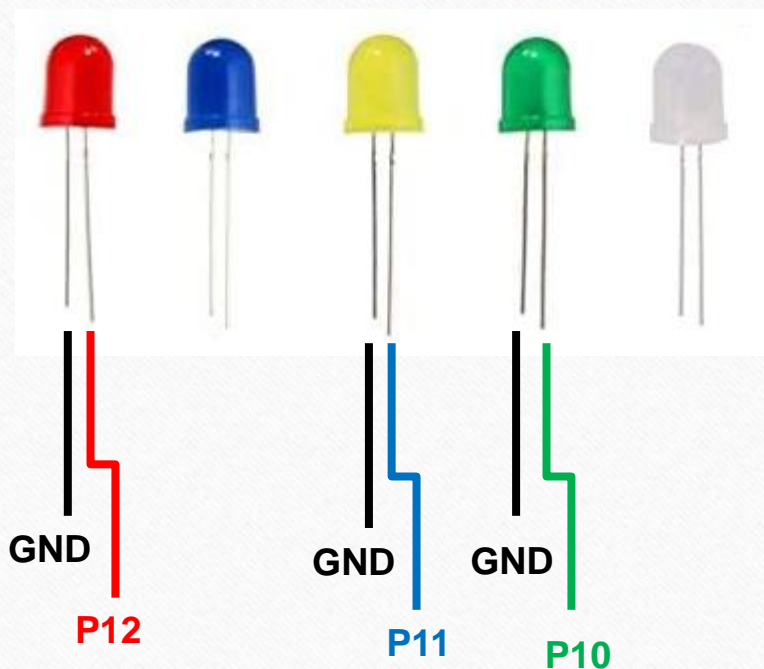
- 未出現Uploading：程式錯誤
多餘程式碼未連接、函數有設定但未使用...
- Uploading後出現錯誤：硬體錯誤
選擇錯誤硬體：Linkit7697、Arduino UNO
選擇錯誤連接埠(例如Com3)或未連接、
連接線接觸不良：部分連接線只能傳送電力、
無法傳送資料或傳輸線品質不良、損壞。



```
燒錄狀態  
Launching Arduino.exe...  
Loading configuration...  
Initializing packages...  
Preparing boards...  
Verifying...  
Build options changed, rebuilding all  
Archiving built core (caching) in: C:\Users\5a88\AppData\Local\Arduino15\packages\arduino\hardware\avr\1.8.10\binaries\697_28344fc95dd77706abfe71ac705adb69.a  
Uploading...  
Traceback (most recent call last):  
File "upload.py", line 124, in <module> File "upload.py", line 62, in open_serial_serialutil.SerialException: could not open port 'COM6': [WinError 10] An error occurred while uploading the sketch  
Arduino.exe exited with 1 and null
```

單色LED

長腳可接P0~P17，短腳接地(GND)



開發板 7697 初始化

- 數位寫入 Pin 10 - SPI 狀態 低電位
- 數位寫入 Pin 11 - SPI 狀態 低電位
- 數位寫入 Pin 12 - SPI 狀態 低電位

重複執行

- 數位寫入 Pin 10 - SPI 狀態 高電位
- 數位寫入 Pin 11 - SPI 狀態 低電位
- 數位寫入 Pin 12 - SPI 狀態 低電位

延遲毫秒 5000

- 數位寫入 Pin 10 - SPI 狀態 低電位
- 數位寫入 Pin 11 - SPI 狀態 高電位
- 數位寫入 Pin 12 - SPI 狀態 低電位

重複 3 次

執行 延遲毫秒 500

數位寫入 Pin 11 - SPI 狀態 低電位

延遲毫秒 500

數位寫入 Pin 11 - SPI 狀態 高電位

- 數位寫入 Pin 10 - SPI 狀態 低電位
- 數位寫入 Pin 11 - SPI 狀態 低電位
- 數位寫入 Pin 12 - SPI 狀態 低電位

延遲毫秒 300

- 數位寫入 Pin 10 - SPI 狀態 低電位
- 數位寫入 Pin 11 - SPI 狀態 低電位
- 數位寫入 Pin 12 - SPI 狀態 高電位

延遲毫秒 2000

呼吸燈

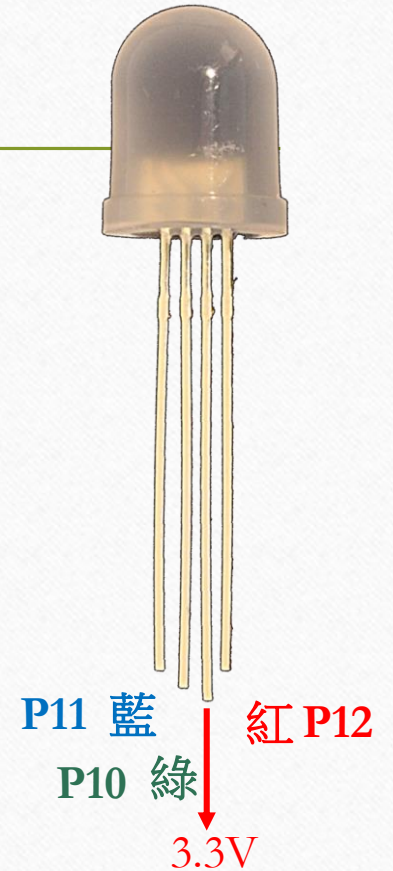
可以增加暫停時間

```
開發板 7697 初始化  
宣告 全域 整數 (int) i 值 0  
重複執行  
  重複 50 次  
  執行  
    類比寫入 Pin 11 數值 (0 - 255) i  
    修改 i 增量 5  
    延遲毫秒 100  
  重複 50 次  
  執行  
    類比寫入 Pin 11 數值 (0 - 255) i  
    修改 i 增量 -5  
    延遲毫秒 100  
  延遲毫秒 500
```

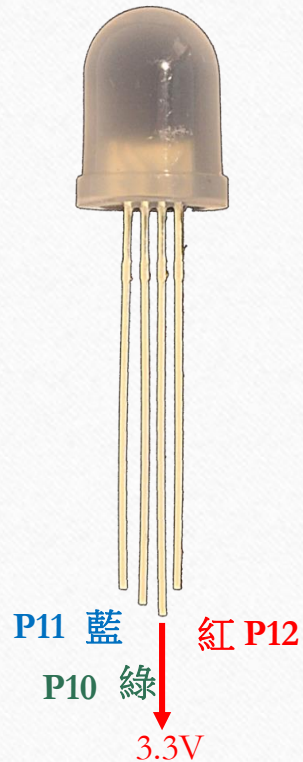
RGB 3色共陽極LED



- 材料包裡面的3色LED是**共陽極**，一般必須在GND端請加限流電阻 220Ω ，否則容易燒壞LED，但是本產品不用。
- **陽極**(老師是接**黃色**電線)腳位需接**3.3V**。
- 因為是**共陽極**，所以當腳位為**低電位**則**亮**燈，**高電位**則**不亮**。



RGB 3色共陽極LED



```
開發板 7697 初始化  
數位寫入 Pin 10 - SPI 狀態 高電位  
數位寫入 Pin 11 - SPI 狀態 高電位  
數位寫入 Pin 12 - SPI 狀態 高電位  
延遲毫秒 1000  
重複執行  
數位寫入 Pin 10 - SPI 狀態 低電位  
數位寫入 Pin 11 - SPI 狀態 高電位  
數位寫入 Pin 12 - SPI 狀態 高電位  
延遲毫秒 1000  
數位寫入 Pin 10 - SPI 狀態 高電位  
數位寫入 Pin 11 - SPI 狀態 低電位  
數位寫入 Pin 12 - SPI 狀態 高電位  
延遲毫秒 1000  
數位寫入 Pin 10 - SPI 狀態 高電位  
數位寫入 Pin 11 - SPI 狀態 高電位  
數位寫入 Pin 12 - SPI 狀態 低電位  
延遲毫秒 1000  
數位寫入 Pin 10 - SPI 狀態 低電位  
數位寫入 Pin 11 - SPI 狀態 高電位  
數位寫入 Pin 12 - SPI 狀態 低電位  
延遲毫秒 2000  
數位寫入 Pin 10 - SPI 狀態 低電位  
數位寫入 Pin 11 - SPI 狀態 低電位  
數位寫入 Pin 12 - SPI 狀態 低電位  
延遲毫秒 5000  
數位寫入 Pin 10 - SPI 狀態 高電位  
數位寫入 Pin 11 - SPI 狀態 高電位  
數位寫入 Pin 12 - SPI 狀態 高電位  
延遲毫秒 3000
```

不亮

綠燈

藍燈

紅燈

黃燈(綠+紅)

白燈

不亮

僅供參考(國高中)：可略過

- LED應串接限流電阻，如果不加限流電阻直接接發光二極體，流過管子的電流會大大超出它的工作電流，導致過熱損壞。
- 本次使用10mm的LED是**共陽極**，最長腳請接**3.3V**。避免接線過於複雜且僅作短時間測試，可不接電阻。**如果要長時間使用建議串接100歐姆Ω電阻。如果接5V，請用150歐姆Ω電阻。如果不清楚LED的基本資料，保險一點就接220歐姆Ω電阻，只是亮度會小一點。**
- LED電阻的算法就是利用歐姆定律Ohm's Law ($R = V / I$)

電阻 = (輸入電壓 - 順向電壓) / 順向電流

功率 = (輸入電壓 - 順向電壓) * 順向電流

順向電壓 -> 啟動電壓

順向電流 -> 額定電流

	55~65Ω		3.3V		5~15Ω		5~15Ω
顏色	紅色	黃色	綠色(普綠)	藍色	綠色(翠綠)		
波長	620-625nm	585-590nm	560-570nm	460-465nm	520-525nm		
亮度	1000-1400mcd	1200-1400mcd	300-400mcd	1000-1200mcd	2000-3000mcd		
電壓	2.0-2.2V	2.0-2.2V	2.0-2.2V	3.0-3.2V	3.0-3.2V		
電流	20mA	20mA	20mA	20mA	20mA		
腳長	28mm	28mm	28mm	28mm	28mm		

ISD1820錄放音模組

- 工作電壓：直流3-5V
- 10秒語音錄放
- REC鍵：錄音按鍵，按住就能錄音，鬆開按鍵停止錄音；
- PLAYL鍵：點動模式放音，按住才放音，鬆開就停止放音；
- RLAYE鍵：觸發模式放音，按一下就播放當前整段的語音；



麥克風



ISD1820錄放音模組

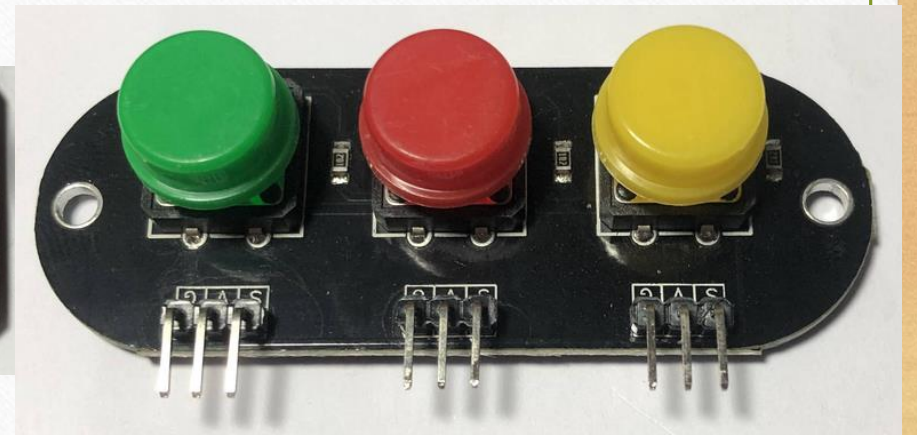
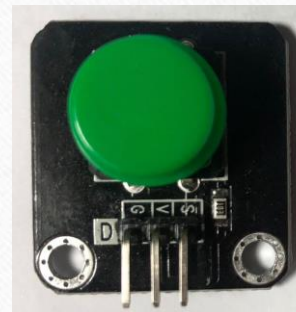
- 只要給P-E腳位一個高電位，就可以觸發模組發出**完整**已錄好的聲音。
- 給P-L腳位一個高電位，就可以觸發模組發出**時間內**已錄好的聲音。時間長度須超過錄音的時間，才能完整播放。



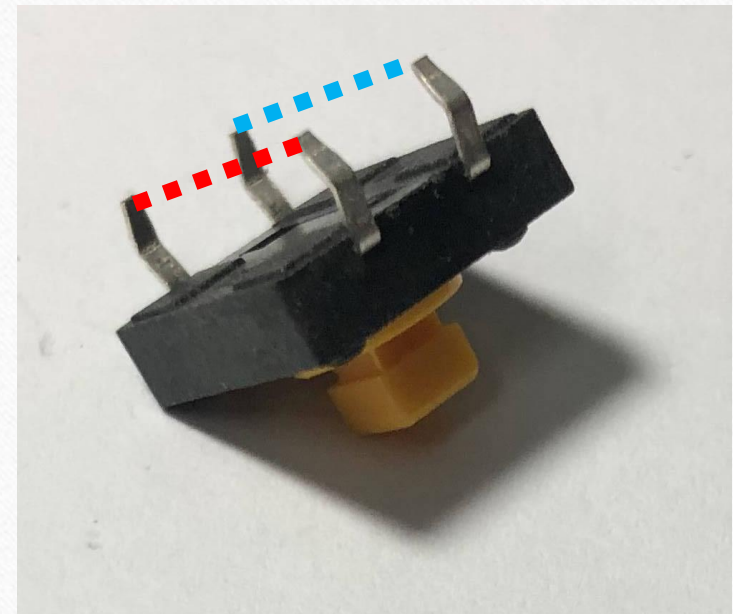
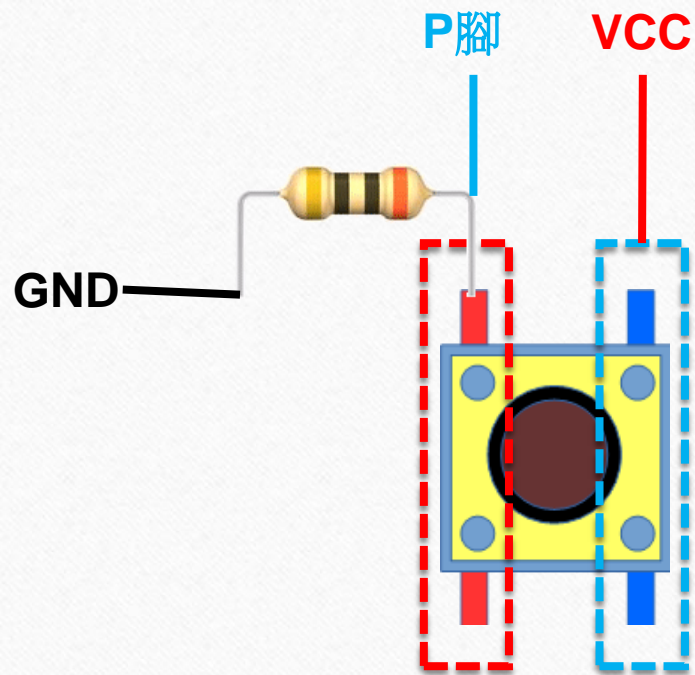
數位值讀取

按鈕開關 (Push Button) 簡稱PB

- 按鈕開關是一個很常見的元件，當我們進行按壓的時候，按鈕上的四隻腳部分或全部會呈現通路的情形，為了避免有短路的發生，所以我們要接一顆電阻(100Ω)進行保護。



四腳按鈕開關，對向兩兩連通，按下開關的當下四支腳互通。



按鈕開關



按住亮，放開不亮

按鈕開關-按1下亮，再按1下不亮

開發板 7697 初始化

宣告 全域 布林 (boolean) LedState 值 假

宣告 全域 布林 (boolean) buttonUP 值 真

重複執行

如果 數位讀取 Pin 10 且 buttonUP = 真

執行

- 設定 LedState 值 LedState 不成立
- 數位寫入 Pin 11 狀態 LedState
- 設定 buttonUP 值 假

否則如果 數位讀取 Pin 10 不成立 且 buttonUP = 假

執行

- 設定 buttonUP 值 真

延遲毫秒 100

方式1

開發板 7697 初始化

宣告 全域 布林 (boolean) P11_State 值 假

重複執行

如果 數位讀取 Pin 10

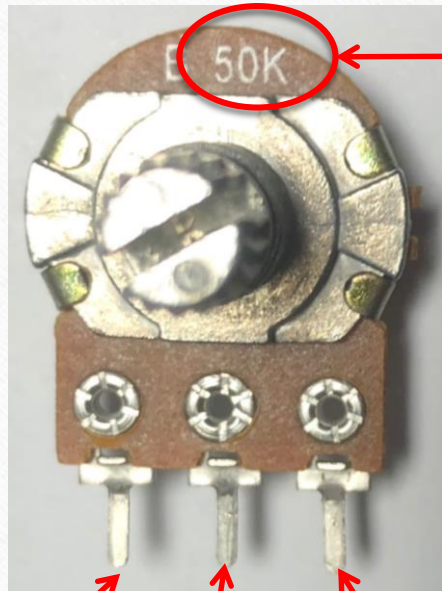
執行

- 設定 P11_State 值 P11_State 不成立
- 數位寫入 Pin 11 狀態 P11_State
- 重複當 數位讀取 Pin 10
- 執行

延遲毫秒 100

方式2

可變電阻 potentiometer



可變電阻最大電阻值：50K Ω

Linkit7697類比回傳值範圍為：0~4095

P腳位

A0→P14

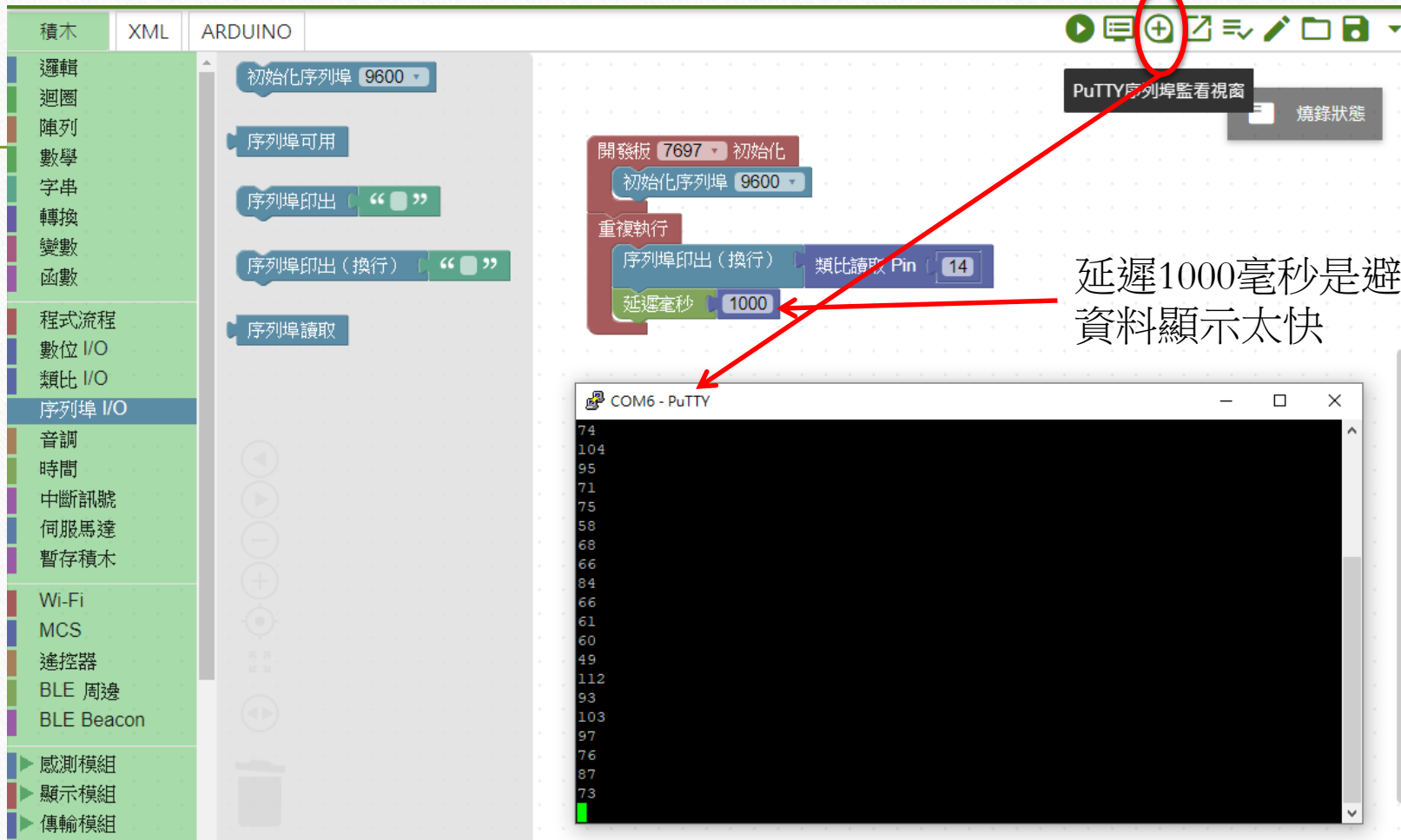
A1→P15

A2→P16

A3→P17

GND 接P腳位 5V

可以利用序列埠I/O來顯示回傳值



The screenshot displays the Arduino IDE interface. On the left, the 'Serial I/O' category is selected in the block palette. The main workspace contains the following code blocks:

- 初始化序列埠 9600
- 序列埠可用
- 序列埠印出 “ ”
- 序列埠印出 (換行) “ ”
- 序列埠讀取

The code is implemented as follows:

```
開發板 7697 初始化  
初始化序列埠 9600  
重複執行  
序列埠印出 (換行) 類比讀取 Pin 14  
延遲毫秒 1000
```

The serial monitor window, titled 'COM6 - PuTTY', shows the following output:

```
74  
104  
95  
71  
75  
58  
68  
66  
84  
66  
61  
60  
49  
112  
93  
103  
97  
76  
87  
73
```

A red circle highlights the '+' icon in the top toolbar, and a red arrow points from it to the '延遲1000' block. Another red arrow points from the text '延遲1000毫秒是避免資料顯示太快' to the same block.

延遲1000毫秒是避免資料顯示太快

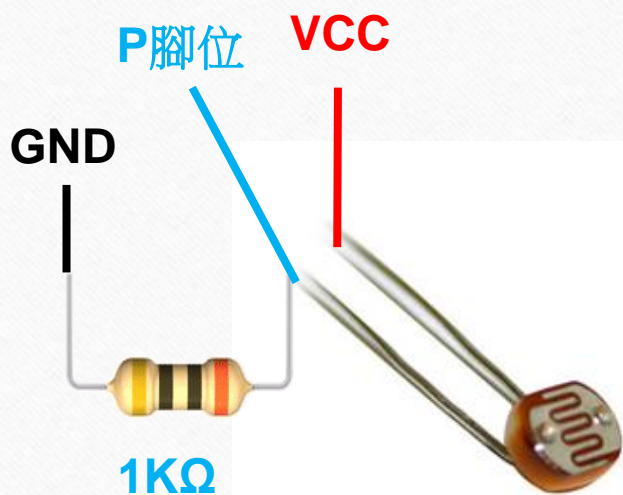
可變電阻控制LED燈



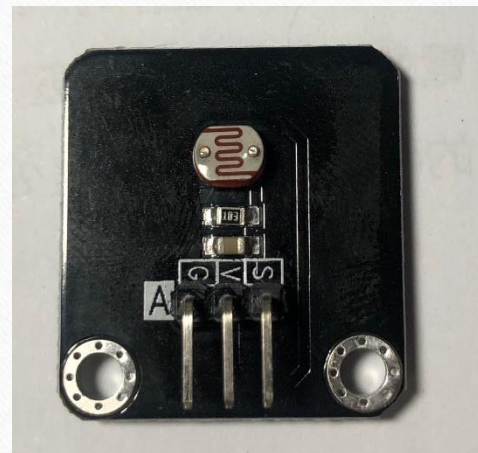
映射：按比例改變
0對應0、4095對應255

光敏電阻

- 光敏電阻是利用光電導效應的一種特殊的電阻。它的電阻和光線的強弱有直接關係。光強度增加，則電阻減小；光強度減小，則電阻增大。利用這個特性，可以用來當作光偵測器的元件。(資料來源：wiki)
- 沒有方向性，VCC、GND相反只是小變大、小變大的差別



光敏電阻



光敏電阻模組

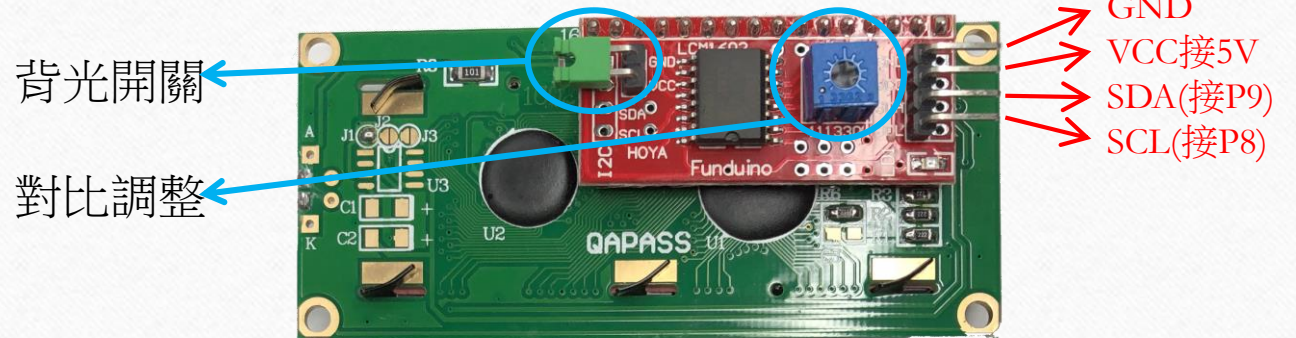
光敏電阻-自動開關小夜燈



程式確認後序列埠可以取消
本例當光敏電阻回傳值小於50
(昏暗)，則LED燈(P10) 打開。

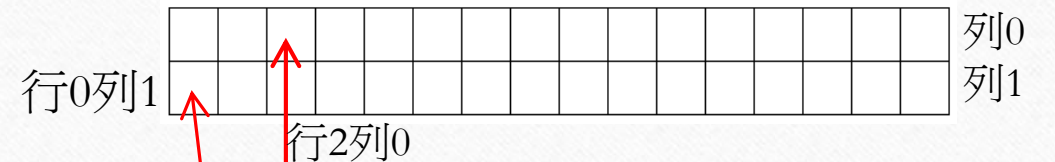
LCD1602顯示模組

- 16表示每列16個字(限英數)
02表示共2列
- SDA接P9
SCL接P8
- 機器原點在**左上角**，
向右向下計算。



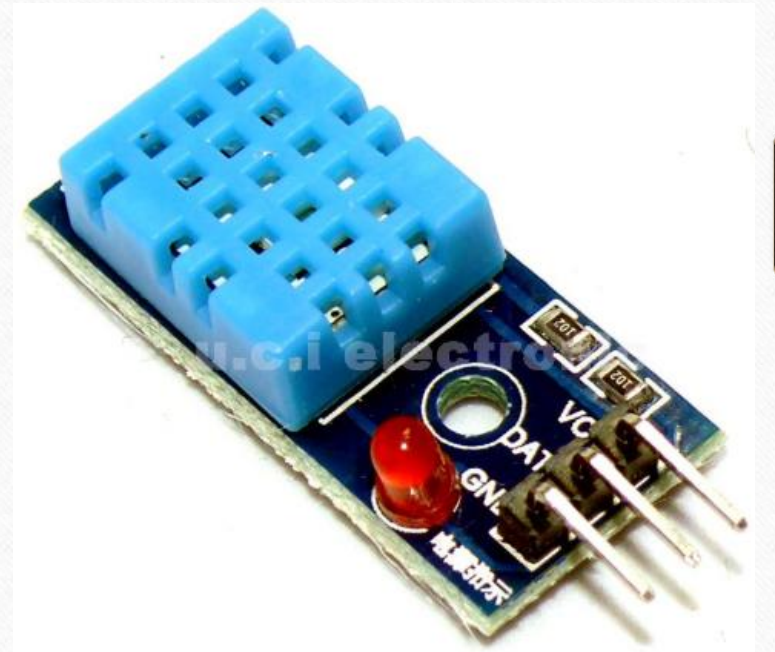
LCD 1602A 16*2 藍底白字
學校這批LCD的配置位址是0x27

LCD1602顯示模組



溫濕度感測器 DHT11模組

- 濕度測量範圍：20---90%RH、濕度測量精度：±5%RH
- 溫度測量範圍：0---50°C、溫度測量精度：±2°C
- 工作電壓：DC5V/3.3V
- 接線VCC → 3.3V/5V電源正極
GND → 電源負極、DATA → 單片機IO口
- **注意：切勿將VCC與GND接反，接反必燒壞！**



溫濕度感測器 DHT11+LCD模組

The screenshot shows the Scratch environment with the following components:

- Left Panel (Library):**
 - 遙控器
 - BLE 周邊
 - BLE Beacon
 - 感測模組
 - 環境感測 (highlighted with a red box)
 - 距離感測
 - 動態感測
 - 顯示模組
 - OLED顯示 (highlighted with a red box)
 - 液晶顯示
 - 彩色燈條
 - 傳輸模組
 - Grove 模組
 - 其他元件
 - 法蘭斯_一般功能
 - 法蘭斯_網頁元素
- Main Stage (Scratch Script):**
 - 開發板 7697 初始化**
 - 初始化 液晶顯示 1602 訊號: I2C I2C位址 0x27
 - 液晶顯示 背光 開
 - 液晶顯示 清除畫面
 - 重複執行**
 - 液晶顯示 設定游標行: 0 列: 0
 - 液晶顯示 顯示 "Temp: "
 - 液晶顯示 顯示 溫濕度計 DHT11 訊號: 40-SPI 量測數值: 溫度 (°C)
 - 液晶顯示 設定游標行: 0 列: 1
 - 液晶顯示 顯示 "Dump: "
 - 液晶顯示 顯示 溫濕度計 DHT11 訊號: 10-SPI 量測數值: 相對濕度 (%)

沒有指定位置，
則接在前資料的後面。

溫濕度感測器 DHT11+LCD模組2

- 邏輯
- 迴圈
- 陣列
- 數學
- 字串
- 轉換
- 變數
- 函數
- 程式流程
- 數位 I/O
- 類比 I/O
- 序列埠 I/O
- 音調
- 時間
- 中斷訊號
- 伺服馬達

轉換成 字串 (String)

- ✓ 字串 (String)
- 字串(二進制)
- 字串(八進制)
- 字串(十進制)
- 字串(十六進制)
- 整數 (int)
- 無號整數 (unsigned int)
- 位元組 (byte)
- 無號數 (word)
- 長整數 (long)

開發板 7697 初始化

- 初始化 液晶顯示 1602 訊號： I2C I2C位址 0x27
- 液晶顯示 背光 開
- 液晶顯示 清除畫面

重複執行

- 液晶顯示 設定游標行： 0 列： 0
- 液晶顯示 顯示 “ Temp: ”
- 液晶顯示 顯示 溫濕度計 DHT11 訊號： 10 - SPI 量測數值： 溫度 (°C) 轉換成 整數 (int)
- 液晶顯示 設定游標行： 0 列： 1
- 液晶顯示 顯示 “ Dump: ”
- 液晶顯示 顯示 溫濕度計 DHT11 訊號： 10 - SPI 量測數值： 相對濕度 (%) 轉換成 整數 (int)

超音波感測器

- ◆ 超音波感測器是由超音波發射器、接收器和控制電路所組成。工作時會從發射端的發射聲波，從接受端接收回音，從2者時間差計算出距離。
- ◆ 超音波因為它的頻率很高，超過是人類耳朵可以接收的範圍，所以人類聽不到它的聲音。
- ◆ 超音波感測器廣泛運用在水位監測，機器人防撞，醫學診斷疾病。

超音波感測器 HC-SR04



腳位：VCC
Trig(P17)
Echo(P16)
GND

超音波+LCD模組

The image shows a block-based programming interface for a microcontroller project. On the left is a sidebar with a tree view of modules: BLE Beacon, 感測模組 (Sensing Modules), 顯示模組 (Display Modules), 傳輸模組 (Communication Modules), and Grove 模組 (Grove Modules). Under 感測模組, there are sub-items: 環境感測 (Environmental Sensing), 距離感測 (Distance Sensing), and 動態感測 (Dynamic Sensing). Under 顯示模組, there are sub-items: OLED 顯示 (OLED Display), 液晶顯示 (LCD Display), and 彩色燈條 (Color LED Strip). The main workspace contains the following code blocks:

- 開發板 7697 初始化** (Board 7697 Initialization) block containing:
 - 初始化 液晶顯示 1602 訊號: I2C I2C位址 0x27 (Initialize LCD Display 1602, Signal: I2C, I2C Address: 0x27)
 - 液晶顯示 背光 開 (LCD Display Backlight On)
 - 液晶顯示 清除畫面 (LCD Display Clear Screen)
- 重複執行** (Repeat Execution) block containing:
 - 液晶顯示 設定游標行: 0 列: 0 (LCD Display Set Cursor Row: 0, Column: 0)
 - 液晶顯示 顯示 " Dist: " (LCD Display Show " Dist: ")
 - 液晶顯示 顯示 超音波測距感應器 HC-SR04P 觸發腳 (Trig) 17 回應腳 (Echo) 16 量測數值: 公分 (LCD Display Show Ultrasonic Distance Sensor HC-SR04P, Trigger Pin (Trig) 17, Echo Pin (Echo) 16, Measurement Value: cm)
 - 延遲毫秒 500 (Delay Milliseconds 500)

The pins 17 and 16 in the sensor block are circled in red.

無源蜂鳴器

無源蜂鳴器

- 1.無源內部不帶震盪源，聲音可以發出“DoRaMiFaSoLaSiDo”的效果。
- 2.工作電壓3.3V-5V

有源蜂鳴器

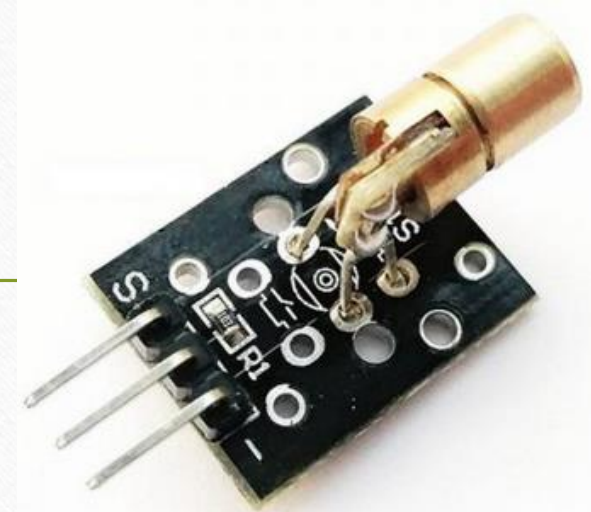
- 1.有源自帶震盪源，只能發出簡單聲音。
- 2.工作電壓3.3V-5V



```
開發板 7697 初始化
重複執行
音調 Pin 14 頻率 C:Do
延遲毫秒 1000
音調 Pin 14 頻率 D:Re
延遲毫秒 1000
音調 Pin 14 頻率 E:Mi
延遲毫秒 1000
停止音調 Pin 14
延遲毫秒 1000
```

紅光雷射模組

- 工作電壓：5V
- 光源波長：650 nm
(人眼睛可看見的範圍312nm - 1050 nm)
- 請勿直射眼睛
- 旋轉頭部可調焦



伺服馬達Servo(舵機)

- 舵機是一種可以控制旋轉角度的馬達，適用於角度不斷變化的控制系統。目前在遙控玩具，如飛機、船、機器人中已經普遍使用。
- 舵機種類多，塑料齒、金屬齒、小尺寸、標準尺寸、大尺寸、可旋轉角度、扭力大小等，依實際需求選購。



DS04-NFC 360°舵機

SG90 180°舵機

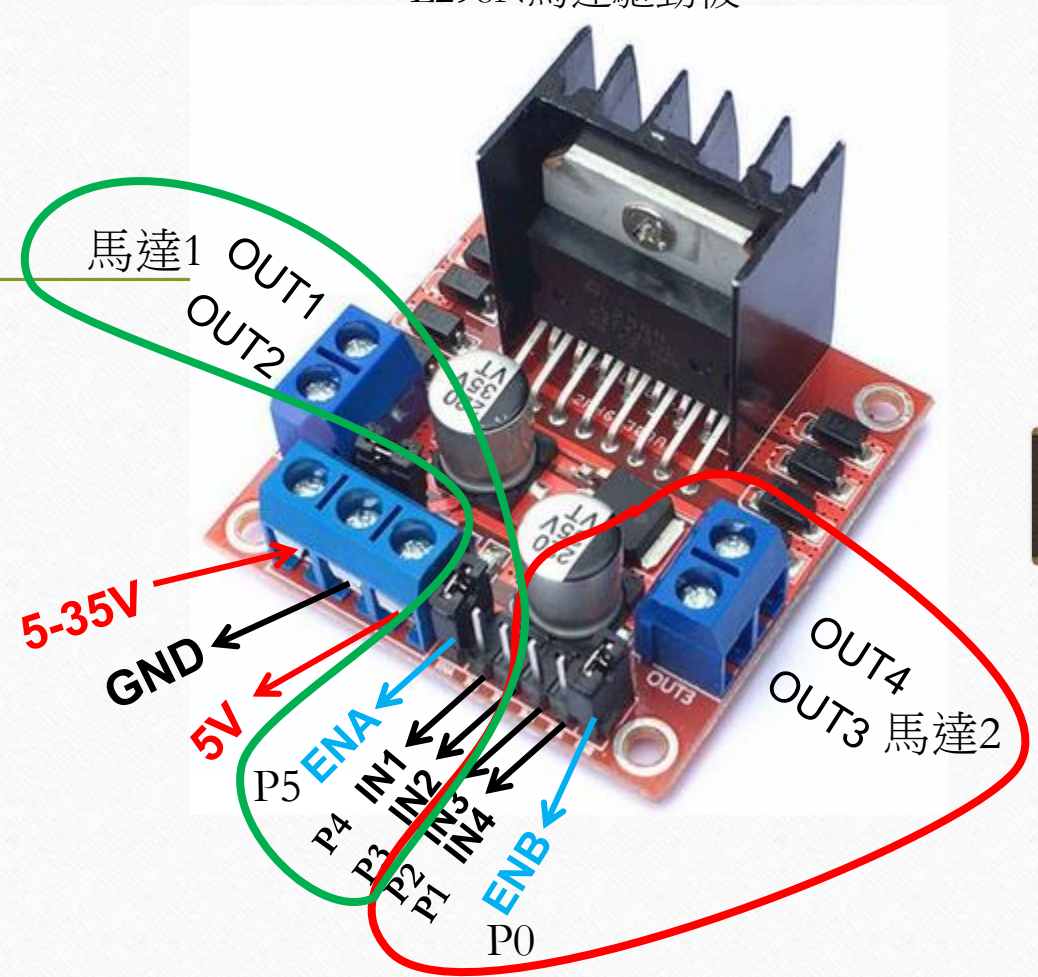
伺服馬達Servo(舵機)

The image shows the Arduino IDE block editor interface. On the left, a sidebar lists various categories, with '伺服馬達' (Servo) highlighted. The main workspace contains several blocks related to servo motor control. A red box highlights the '寫入伺服馬達 Pin 0 - UART 角度 (0 - 180) 0' block. Other blocks include '寫入伺服馬達 Pin 0 角度 (0 - 180) 0', '讀取伺服馬達 Pin 0 - UART 目前角度', '設定伺服馬達 Pin 0 - UART 脈衝寬度 最小值 544 最大值 2400', and '寫入伺服馬達 Pin 0 - UART 脈衝寬度 1500'. On the right, a '重複執行' (Repeat) loop contains a sequence of blocks: '寫入伺服馬達 Pin 6 角度 (0 - 180) 0', '延遲 2000', '寫入伺服馬達 Pin 6 角度 (0 - 180) 90', '延遲 1000', '寫入伺服馬達 Pin 6 角度 (0 - 180) 180', '延遲 1000', and '寫入伺服馬達 Pin 6 角度 (0 - 180) 90'.

L298N馬達控制器

- 驅動電壓：5V~35V
驅動電流：2A(MAX)
- ENA可以輸入PWM信號驅動馬達1的轉速，IN1、IN2高低電位不同可以控制馬達1的方向。
- ENB可以輸入PWM信號驅動馬達2的轉速，IN3、IN4高低電位不同可以控制馬達1的方向。

L298N馬達驅動板



L298N 馬達控制器


利用函數簡化程式

函數 左輪前進  宣告參數

類比寫入 Pin 5 數值 (0 - 255) 255

數位寫入 Pin 3 狀態 高電位


數位寫入 Pin 4 狀態 低電位

函數 右輪前進 

類比寫入 Pin 0 數值 (0 - 255) 255

數位寫入 Pin 1 狀態 高電位


數位寫入 Pin 2 狀態 低電位

函數 左輪後退 

類比寫入 Pin 5 數值 (0 - 255) 255

數位寫入 Pin 3 狀態 低電位


數位寫入 Pin 4 狀態 高電位

函數 右輪後退 

類比寫入 Pin 0 數值 (0 - 255) 255

數位寫入 Pin 1 狀態 低電位


數位寫入 Pin 2 狀態 高電位

函數 左輪停止 

類比寫入 Pin 5 數值 (0 - 255) 0

數位寫入 Pin 3 狀態 高電位

數位寫入 Pin 4 狀態 高電位

函數 右輪停止 

類比寫入 Pin 0 數值 (0 - 255) 0

數位寫入 Pin 1 狀態 高電位

數位寫入 Pin 2 狀態 高電位

積木

XML

ARDUINO

邏輯

迴圈

陣列

數學

字串

轉換

變數

函數

程式流程

數位 I/O

類比 I/O

序列埠 I/O

音調

時間

中斷訊號

伺服馬達

暫存積木

Wi-Fi

MCS

遙控器

BLE 周邊

BLE Beacon

感測模組

顯示模組

 OLED顯示

 液晶顯示

 彩色燈條

傳輸模組

Grove 模組

其他元件

法蘭斯 一般功能

Linkit Remote遙控器

新增 直式 遙控器

名稱 “ Linkit 7697 ”

切割為寬 3

高 5

新增 字串標籤 “ label1 ” 顯示字串 “ ” 位置(X,Y) 0 0 大小(寬,高) 1 1 顏色 橘色

新增 方形按鈕 “ button1 ” 顯示字串 “ ” 位置(X,Y) 0 0 大小(寬,高) 1 1 顏色 橘色

數位寫入 Pin 11 - SPI 狀態 高電位

新增 圓形按鈕 “ circle1 ” 顯示字串 “ ” 位置(X,Y) 0 0 大小(寬,高) 1 1 顏色 橘色

延遲毫秒 20

新增 開關 “ switch1 ” 顯示字串 “ ” 位置(X,Y) 0 0 大小(寬,高) 1 1 顏色 橘色

新增 滑桿 “ slider1 ” 顯示字串 “ ” 位置(X,Y) 0 0 大小(寬,高) 1 1 最小值 0 最大值 100 初始值 1 顏色 橘色

新增 搖桿 “ joystick1 ” 顯示字串 “ ” 位置(X,Y) 0 0 大小 1 1 顏色 橘色

遙控器連線中

處理手機程式指令

狀態已更新

從 讀取輸入資料

更新 字串標籤 內容 “ ”

Linkit Remote遙控器



初始化

新增 直式 遙控器

名稱 “ Daygo769 ”

切割為寬 4

高 10

新增 字串標籤 “ label1 ” 顯示字串 “ 趙和修的遙控車 ” 位置(X,Y) 0 0 大小(寬,高) 4 1 顏色 橘色

新增 方形按鈕 “ button1 ” 顯示字串 “ 左輪前進 ” 位置(X,Y) 0 2 大小(寬,高) 2 2 顏色 綠色

新增 方形按鈕 “ button2 ” 顯示字串 “ 右輪前進 ” 位置(X,Y) 2 2 大小(寬,高) 2 2 顏色 綠色

新增 方形按鈕 “ button3 ” 顯示字串 “ 左輪後退 ” 位置(X,Y) 0 4 大小(寬,高) 2 2 顏色 橘色

新增 方形按鈕 “ button4 ” 顯示字串 “ 右輪後退 ” 位置(X,Y) 2 4 大小(寬,高) 2 2 顏色 橘色

新增 滑桿 “ slider1 ” 顯示字串 “ 左輪轉速 ” 位置(X,Y) 0 6 大小(寬,高) 2 1 最小值 60 最大值 255 初始值 150 顏色 粉紅色

新增 滑桿 “ slider2 ” 顯示字串 “ 右輪轉速 ” 位置(X,Y) 2 6 大小(寬,高) 2 1 最小值 60 最大值 255 初始值 150 顏色 粉紅色

新增 圓形按鈕 “ circle1 ” 顯示字串 “ 聲音 ” 位置(X,Y) 2 7 大小(寬,高) 1 1 顏色 橘色

新增 滑桿 “ slider3 ” 顯示字串 “ 舵機角度 ” 位置(X,Y) 0 8 大小(寬,高) 4 2 最小值 0 最大值 180 初始值 90 顏色 藍色

Linkit Remote遙控器版面設定

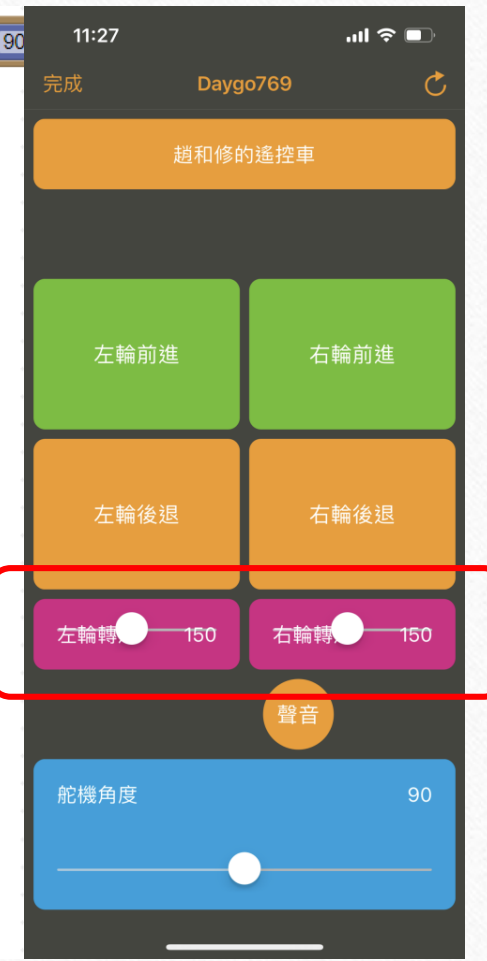


```
新增 滑桿 “slider3” 顯示字串 “舵機角度” 位置(X,Y) 0 8 大小(寬,高) 4 2 最小值 0 最大值 180 初始值 90
? 類比寫入 Pin 0 數值 (0 - 255) 0
數位寫入 Pin 1 狀態 高電位
數位寫入 Pin 2 狀態 高電位
數位寫入 Pin 3 狀態 高電位
數位寫入 Pin 4 狀態 高電位
? 類比寫入 Pin 5 數值 (0 - 255) 0
宣告 全域 整數 (int) V_left 值 150
宣告 全域 整數 (int) V_right 值 150
初始化 液晶顯示 1602 訊號: I2C I2C位址 0x27
液晶顯示 背光 開
液晶顯示 清除畫面
重複執行
處理手機程式指令
如果 slider1 狀態已更新
執行 設定 V_left 值 從 slider1 讀取輸入資料
如果 slider2 狀態已更新
執行 設定 V_right 值 從 slider2 讀取輸入資料
如果 circle1 狀態已更新
執行 如果 從 circle1 讀取輸入資料 1
```

馬達停止

左馬達轉數

右馬達轉數



聲音

```
graph TD
    subgraph Circle1_Logic [circle1 狀態已更新]
        C1_If[如果] --> C1_Read[從 circle1 讀取輸入資料]
        C1_Read --> C1_High[數位寫入 Pin 7 狀態 高電位]
        C1_Read --> C1_Low[數位寫入 Pin 7 狀態 低電位]
    end

    subgraph Button1_Logic [button1 狀態已更新]
        B1_If[如果] --> B1_Read[從 button1 讀取輸入資料]
        B1_Read --> B1_Forward[呼叫 左輪前進]
        B1_Read --> B1_Stop[呼叫 左輪停止]
    end

    subgraph Button2_Logic [button2 狀態已更新]
        B2_If[如果] --> B2_Read[從 button2 讀取輸入資料]
        B2_Read --> B2_Forward[呼叫 右輪前進]
        B2_Read --> B2_Stop[呼叫 右輪停止]
    end

    subgraph Button3_Logic [button3 狀態已更新]
        B3_If[如果] --> B3_Read[從 button3 讀取輸入資料]
        B3_Read --> B3_Backward[呼叫 左輪後退]
        B3_Read --> B3_Stop[呼叫 左輪停止]
    end

    subgraph Button4_Logic [button4 狀態已更新]
        B4_If[如果] --> B4_Read[從 button4 讀取輸入資料]
        B4_Read --> B4_Backward[呼叫 右輪後退]
        B4_Read --> B4_Stop[呼叫 右輪停止]
    end

    subgraph Slider3_Logic [slider3 狀態已更新]
        S3_If[如果]
    end
```



舵機

```
如果 slider3 狀態已更新 執行
  寫入伺服馬達 Pin 6 角度 (0 - 180) 從 slider3 讀取輸入資料
  液晶顯示 清除畫面
  液晶顯示 設定游標行: 0 列: 1
  液晶顯示 顯示 "Angle:"
  液晶顯示 顯示 從 slider3 讀取輸入資料
  液晶顯示 設定游標行: 0 列: 0
  液晶顯示 顯示 "Read to GO!!"
```

```
函數 左輪前進 宣告參數
  類比寫入 Pin 5 數值 (0 - 255) V_right
  數位寫入 Pin 3 狀態 高電位
  數位寫入 Pin 4 狀態 低電位
```

```
函數 右輪前進
  類比寫入 Pin 0 數值 (0 - 255) V_left
  數位寫入 Pin 1 狀態 高電位
  數位寫入 Pin 2 狀態 低電位
```

```
函數 左輪後退
  類比寫入 Pin 5 數值 (0 - 255) V_right
  數位寫入 Pin 3 狀態 低電位
  數位寫入 Pin 4 狀態 高電位
```

```
函數 右輪後退
  類比寫入 Pin 0 數值 (0 - 255) V_left
  數位寫入 Pin 1 狀態 低電位
  數位寫入 Pin 2 狀態 高電位
```

```
函數 左輪停止
  類比寫入 Pin 5 數值 (0 - 255) 0
  數位寫入 Pin 3 狀態 高電位
  數位寫入 Pin 4 狀態 高電位
```

```
函數 右輪停止
  類比寫入 Pin 0 數值 (0 - 255) 0
  數位寫入 Pin 1 狀態 高電位
  數位寫入 Pin 2 狀態 高電位
```



END