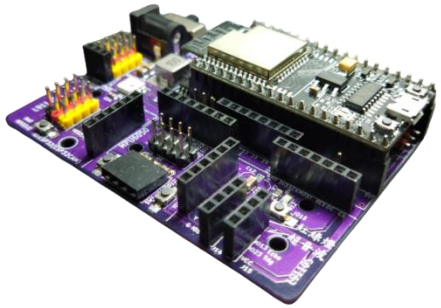
Mason E&D

ESP32 MicroPython 基礎入門與進階物聯網應用

課程大綱



時間	項目	內容
03/15 13:00~16:00	MicroPython 基礎入門	ESP32 介紹及硬體 ESP Gyro擴充板入門操作 - MicroPython開發環境設定與韌體上傳 - microBlock 介面與相關積木介紹 - 完成基本 LED/紅綠燈/按鈕及超音波使用
04/06 13:00~16:00	MicroPython 物聯網應用(1)	馬達進階控制使用及物聯網基本應用一 - Timer / ticks 的使用方式 - L9110S 控制直流馬達 - 伺服馬達 SG90 的使用 - 上傳溫溼度資料至雲端平台
05/10 13:00~16:00	MicroPython 物聯網應用(2)	MicroPython 物聯網進階應用二 - 使用雲端平台遠端控制 LED - 取得雲端資訊: 網路時間/天氣資訊 - 結合 NTP / OLED / Openweather 及圖像顯示 OLED 做個人化的天氣時鐘

上課用的資料： http://gg.gg/esp0315

Mason E&D

Gyro 材料包列表

項目-A	ESP32 含 Gyro 擴充板套件	數量	3/15 上課內容物	待下次上課時發放
1	安信可原廠 nodemcu-32s 38pins	1	V	
2	ESP Gyro 二合一擴充板	1	V	
項目-B	Gyro 教學套件組	數量		
1	0.96 吋 SSD1306 OLED 白字 GND-VCC-SCL-SDA	1		SSD1306 OLED
2	SG90	1	V	
3	L9110S	1		L9110S
4	TT 馬達+輪子	2	V	
5	MPU6050	1		MPU6050
6	YGR 紅綠燈模組	1	V	
7	HC-SR04 超音波模組	1	V	
8	MAX30102 血氧模組	1		MAX30102
9	240x240 ST7789 全彩液晶螢幕	1		ST7789 全彩液晶螢幕
10	無源蜂鳴器	1		無源蜂鳴器
11	DHT-11 溫溼度模組	1		DHT-11
12	整理盒	1	V	

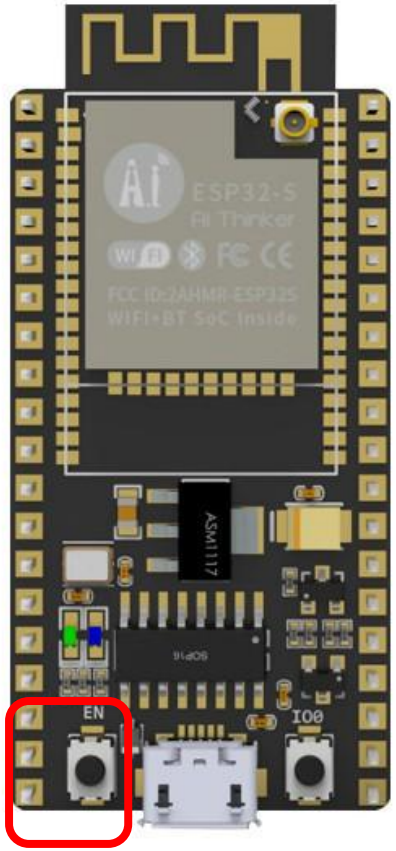
Mason
E&D

ESP32 +Gyro 擴充板(硬體介紹)

https://sites.google.com/view/esp-gyro/

Mason
E&D

ESP32 開發板的功能



重置鈕

ESP32 型號很多種，我採用腳位最多 (38pins)
且大家最推薦的版本 -> 窄版 nodemcu-32s
插入麵包板後，兩邊可留一排腳位通道

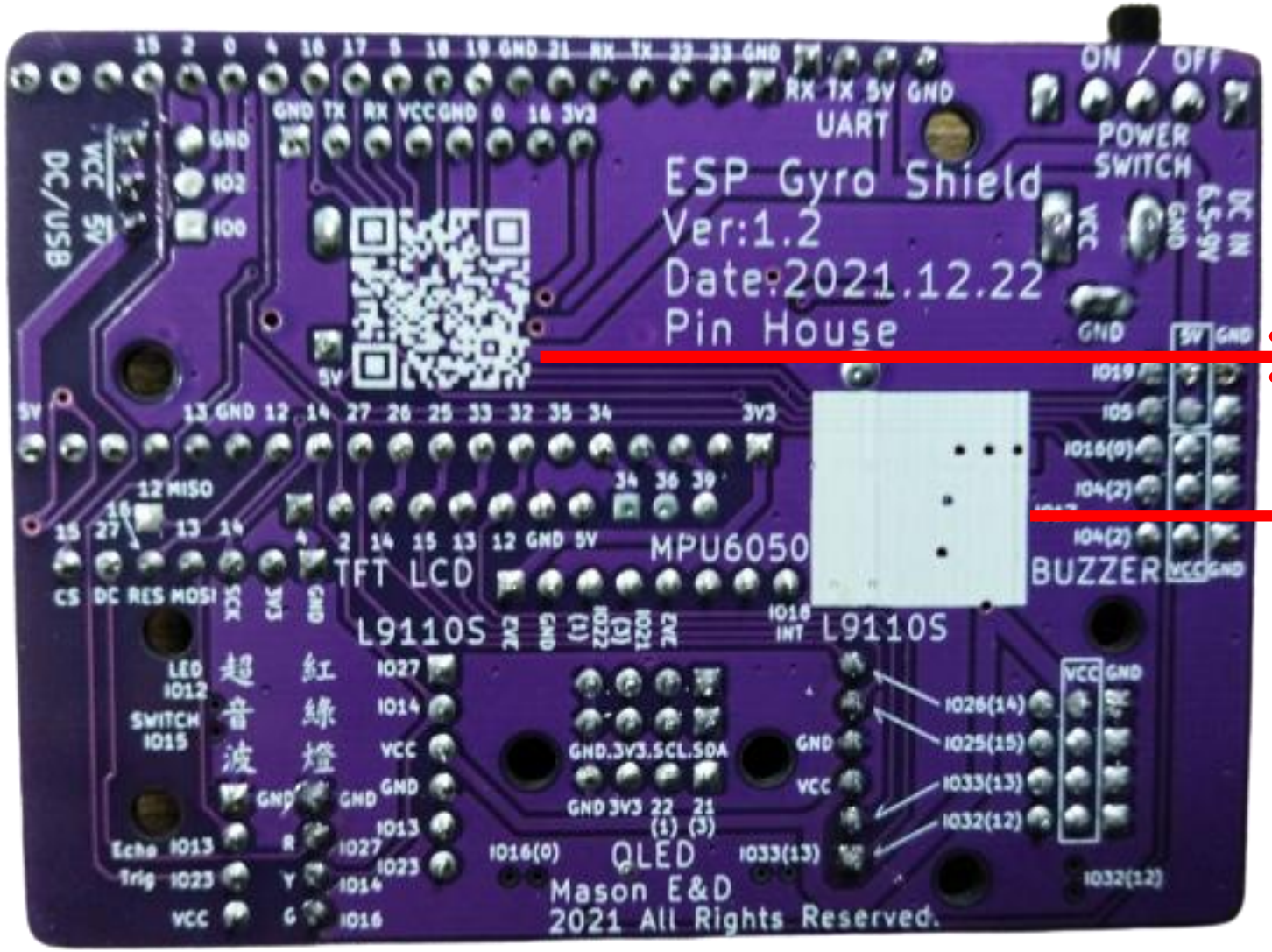
主要功能：

1. 內建 WiFi 2.4G
2. 內建傳統藍牙及低功耗藍牙 BLE
3. GPIO 多達 32根
4. 好用的 I2C (可多組及換腳位)
5. 兩組 SPI -> 接液晶螢幕
6. Flash 4MB
7. 支援眾多的程式語言平台
   - Arduino / MicroPython / C..

使用者廣、網路資源多，為目前最熱門的開發板
TSMC 40nm 制程?
缺點： 腳位標示在背面，看不到

Mason E&D

擴充板相關的使用說明及範例



掃 QR Code 可連至下面網址

https://sites.google.com/view/esp-gyro/

留白底框，方便用簽字筆
編號管理使用

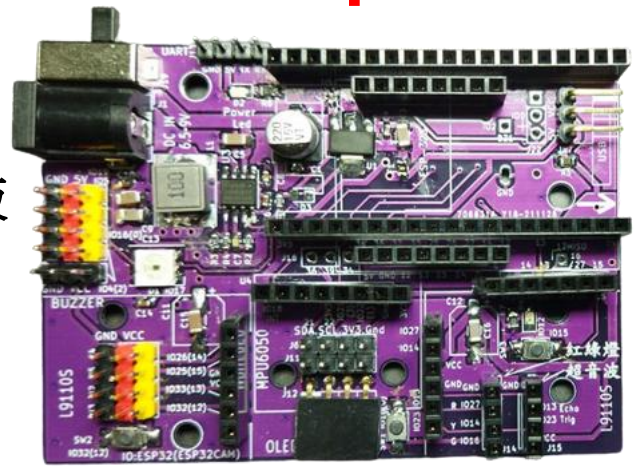
ESP32 +ESP Gyro 擴充板

擴充板使用時注意事項:

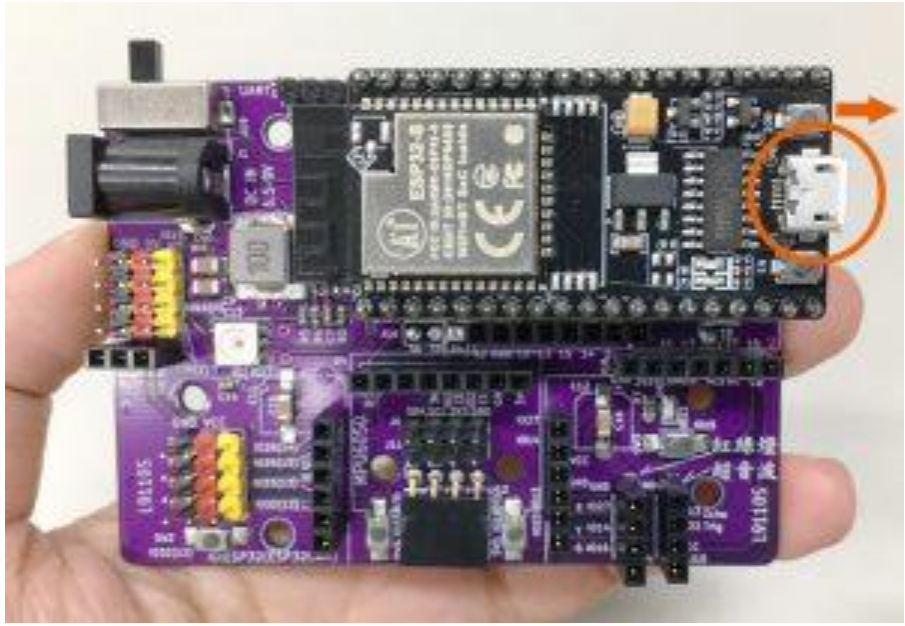
-> 插上開發板時，請注意 usb 那邊請朝擴充版箭頭指的方向，
腳位對準，不要有位移，不然可能會造成開發板損壞

ESP32

Gyro 擴充板



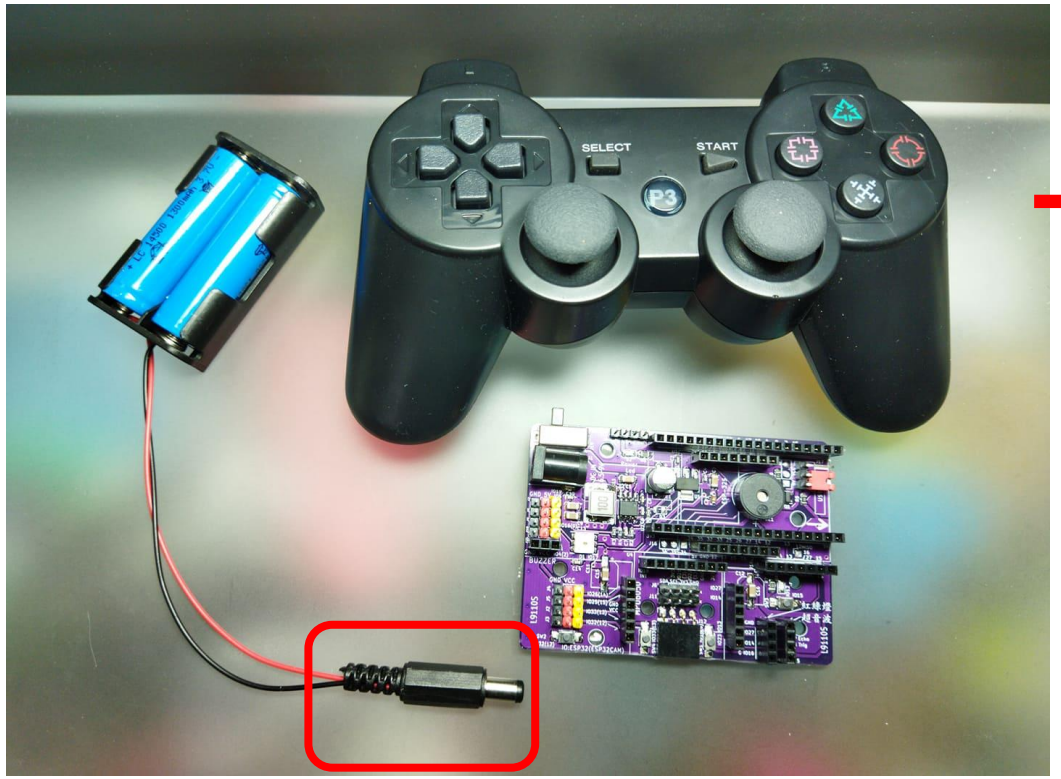
注意箭頭方向



內建 WiFi 及藍牙 BLE，可做物聯網及機器人相關作品

Mason E&D

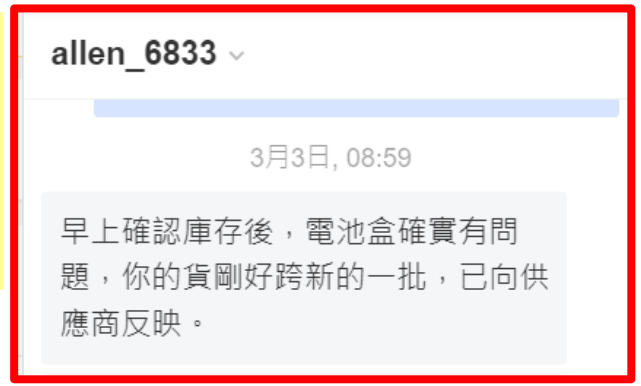
供電方式，可用 14500 或 18650 兩顆串聯 7.4V 的電池盒



無線控制基本上是用手機的 app

而 PS3 手把控制為另外一種客制化應用

外接電池盒時，
請注意只支援
外負內正的標準規範
-> 反向的，會造成零件燒毀



Mason E&D

Gyro 常用腳位表			
ESP32	功能一	功能二	功能三
IO22	I2C - SCL		
IO21	I2C - SDA		
IO19	GVS (SG90)		
IO5	GVS (SG90)		
IO16	GVS (SG90)	紅綠燈- G	按鈕-2
IO4	GVS (SG90)	GVS 母座	
IO26	GVS (SG90)	L9110-A (TT馬達)	
IO25	GVS (SG90)		
IO33	GVS (SG90)	L9110-B (TT馬達)	按鈕-4
IO32	GVS (SG90)		按鈕-1
IO27	紅綠燈- R	L9110-C (TT馬達)	TFT 液晶
IO14	紅綠燈- Y		TFT 液晶
IO13	超音波- ECHO	L9110-D (TT馬達)	TFT 液晶
IO23	超音波- TRIG		
IO17	內建 WS2812		
IO12	TFT 液晶	內建 LED	
IO15	TFT 液晶	按鈕-3	
IO2	內建蜂鳴器		

Gyro 擴充板常用腳位表

I2C 可外接其它模組:
如 OLED / PCA9685 / AI 模組 / 紅外線溫度感測

SPI 可接 TFT 液晶螢幕

GVS 腳位主要可用來接伺服馬達或其它感測器

1. 最多八組伺服馬達 : 機械手臂、四足蜘蛛
2. 最多四組 TT 馬達 : 麥克納姆輪車
3. 兩顆 TT 馬達+四顆伺服馬達 : 遙控手臂車

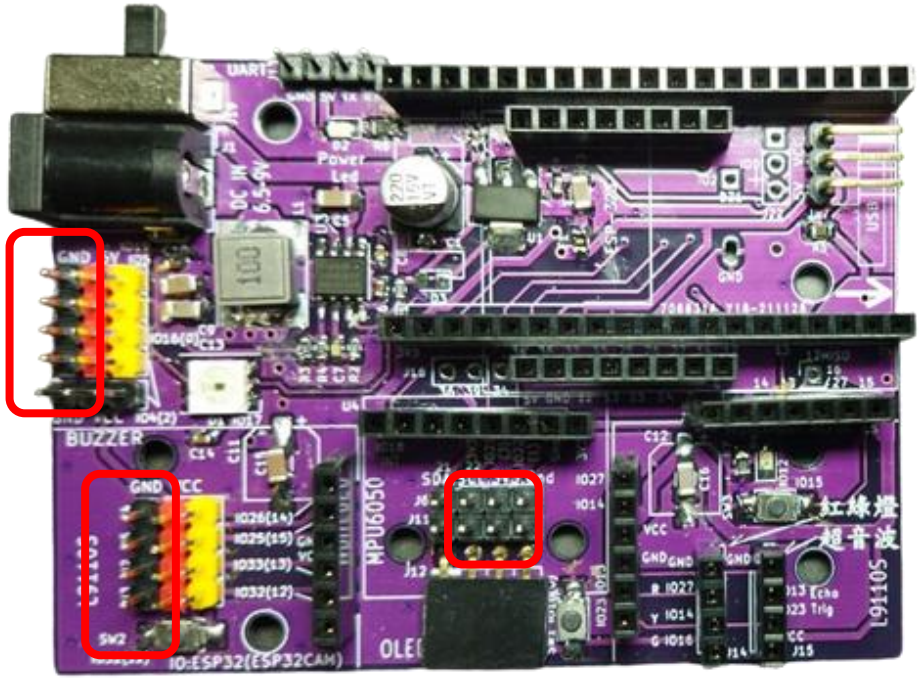
Mason
E&D

使用上需注意事項

使用電源時，需注意短路發生

短路：電源的正負兩極或有電位差的兩條導線【連接】，稱為短路。

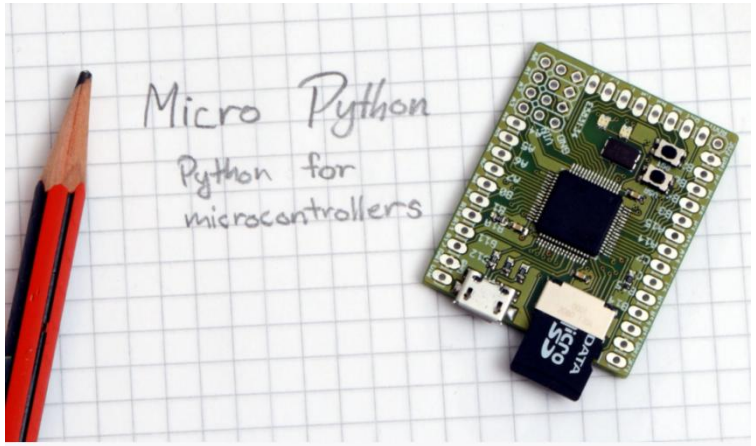
外接電池盒時，
請注意只支援
外負內正的標準規範
-> 反向的，會造成零件燒毀



上圖框起來的地方有正負極，千萬不要用螺絲起子或導體讓兩者”連結”，造成短路
短路瞬間能量釋放，會造成設備損壞.

Mason E&D

MicroPython 簡介



Python 特色? 為什麼要學Python?

-> 程式碼簡潔、易學、易讀、清晰等特性，
-> 有完整的模組 **module** 支援，處理影像、**AI**、大數據很方便
=> 適合作為入門程式語言來學習，是近年最熱門的程式語言之一

但要控制硬體底層的東西，如 **GPIO** 接的燈泡、伺服馬達等設備，反而很困難
還是 **arduino** 以 **C** 為架構的程式語言較方便很多.

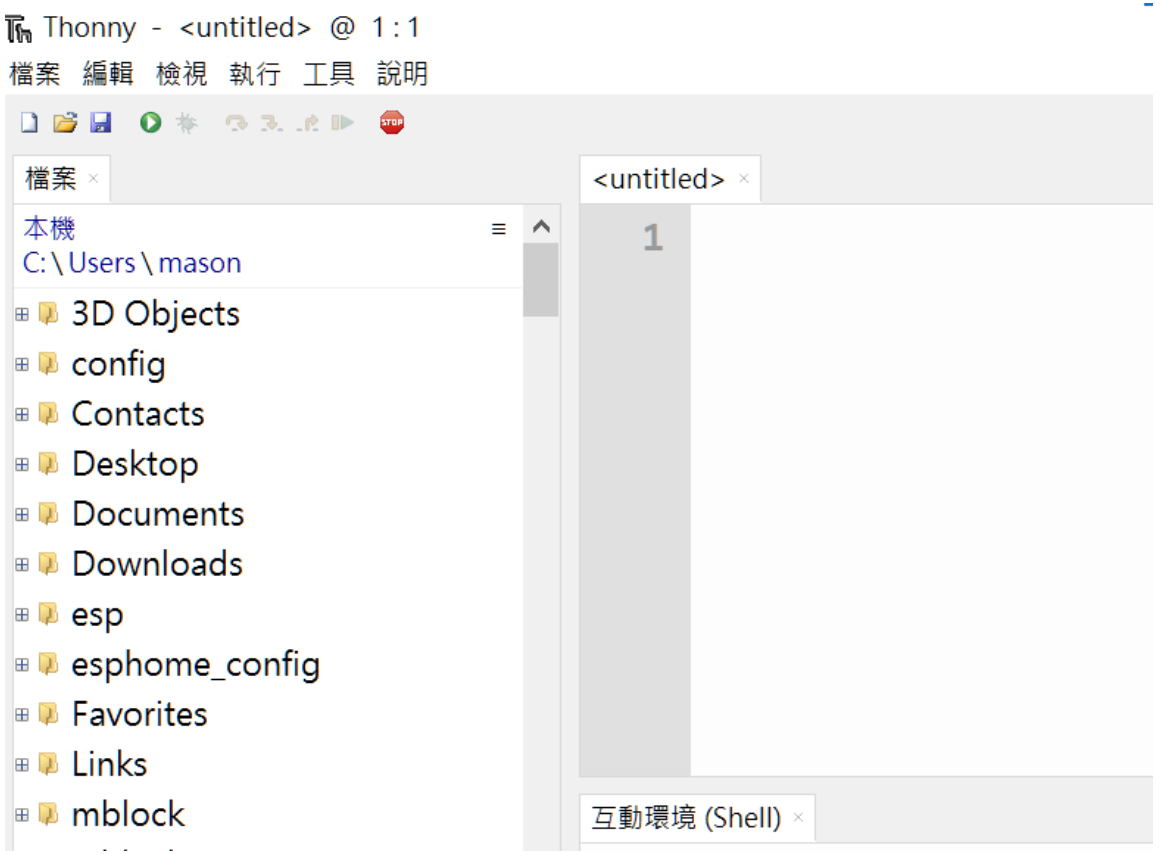
Damien George 一個想法: 能否用**Python** 語言控制單晶片 ，來實現控制硬體設備?

2014 Kickstarter 募資 MicroPython 專案成功，後來很多開發板都支援這個平台
-> microbit / ESP32 / Pi Pico …

Mason E&D

安裝 usb 驅動程式及開發環境

1. 先安裝好 usb 驅動程式 (之前有安裝過的，可省略此步驟)
   - usb_driver/CH341SER.EXE
2. 解開壓縮檔: 離線版 microblock



Mason
E&D

MicroBlock 積木程式開發介面

原始網址：https://microblock.app/
由豐原高中 - 蔡亞柏老師推薦使用
https://www.facebook.com/profile.php?id=100007828750186

- 原先積木只支援 **esp32** 基本的功能及有些 **bug**
- 後續和亞柏老師一起討論和修正相關擴充積木
  更適合 **ESP32** 使用者來使用

未來計畫:
-增加中文化積木介面 **(On-going)**

Mason
E&D

安裝 MicroPython 韌體

Mason E&D

ESP32 安裝 MicroPython 韌體

想像成 **ESP32** 就是一台小型的電腦，要方便使用它，就需安裝一套「作業系統」才能用

韌體可理解成
-> MicroPython 就是一套簡易型的「作業系統」+「**Python** 語法的直譯器」

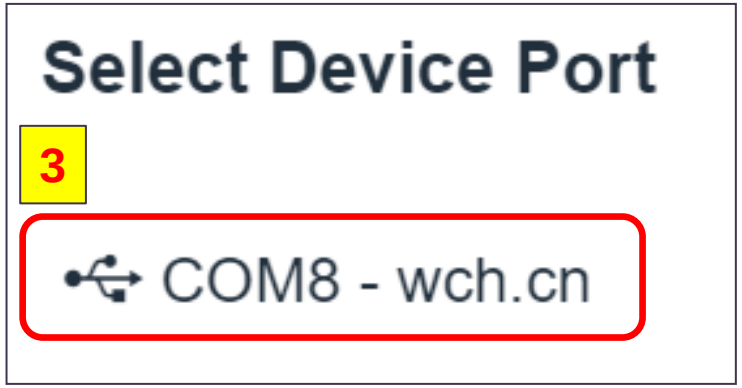
1. 電腦安裝好 **ESP32** 的 **usb driver**
2. 接上 **micro usb** 線至 **ESP32** 上
3. 打開 **microBlock IDE** 選擇
   開發板 -> 更新韌體
4. 選對正確的序列埠，就可以直接上傳韌體

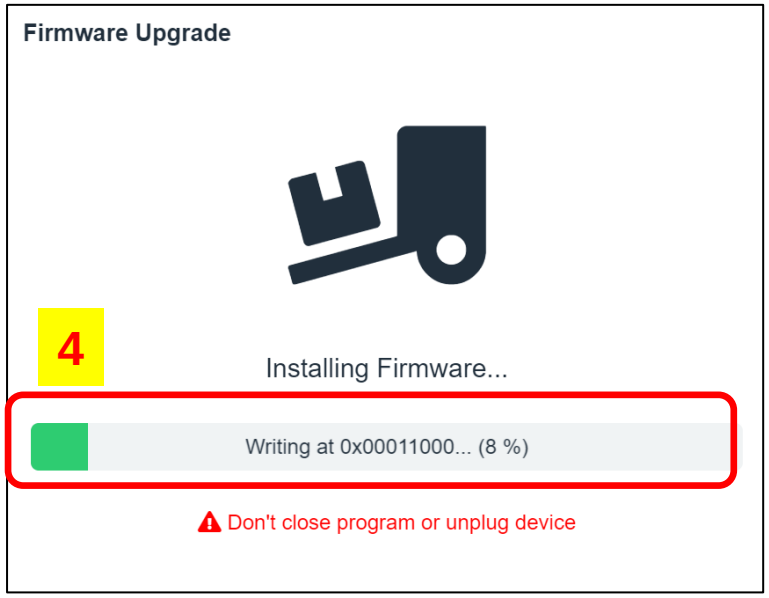
大概約 **3-5** 分鐘，請勿拔掉 **usb** 及亂按板子按鈕

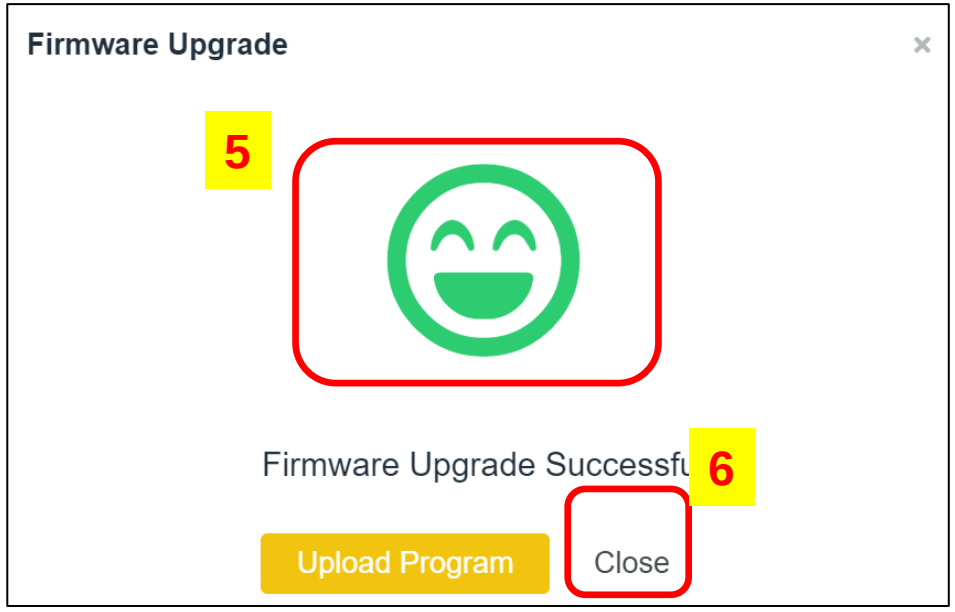
1. 基本如果不燒錄其它韌體，如: **Arduino**
-> 安裝韌體只需燒錄一次
2. 但如果安裝太多程式，就建議重新再燒錄一次即可

Mason E&D

安裝 MicroPython 韌體步驟








Mason E&D
控制命令區
microBlock IDE
檔案 編輯 顯示 開發板 視窗 語言 Help
切換積木/原始碼
重新選擇序列埠
microBlock
Block
Code
Terminal
積木區
Basic
Control
Operators
Variables
程式編輯
digital write 1 to pin 12
wait 1 seconds
digital write 0 to pin 12
wait 1 seconds
REPL 互動區
- 觀看訊息
新專案/命名/儲存/開啟舊檔
ex1_點亮 led 燈
上傳程式鈕
Open project C:\Users\mason\Documents\20220315_class\micropython_ex1-40\ex1_點亮 led 燈.mby at 8:04:42
Board: ESP32 Dev Board
Port: DISCONNECT
Level: Beginner

Mason E&D

Python 基礎語法

- 命名規則
  - 和大多數程式語言一樣，可包含英文字母、數字與下底線，但不能以數字開頭
  - 英文字母區分大小寫小 (A 與 a 是不同的)
- 結尾語句不加分號 (Arduino 程式每行結尾為；)
- 縮排(空白格式)
  - Python 和其他語言最大的不同就是使用縮排來切分程式碼，這和其他語言使用 { } 不同。
- 模組
  - Import , from import
- 註解 單行 # 多行 ‘’’ 或 ###

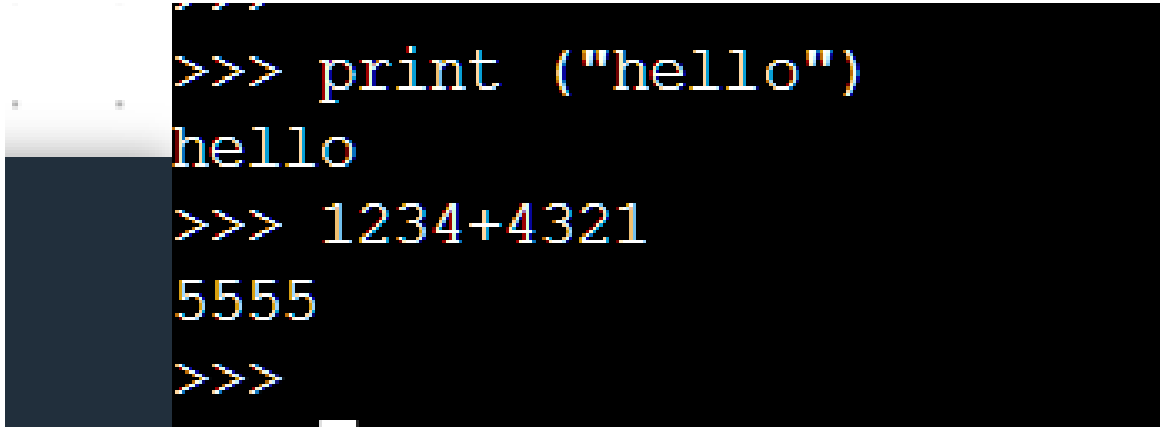
Mason E&D

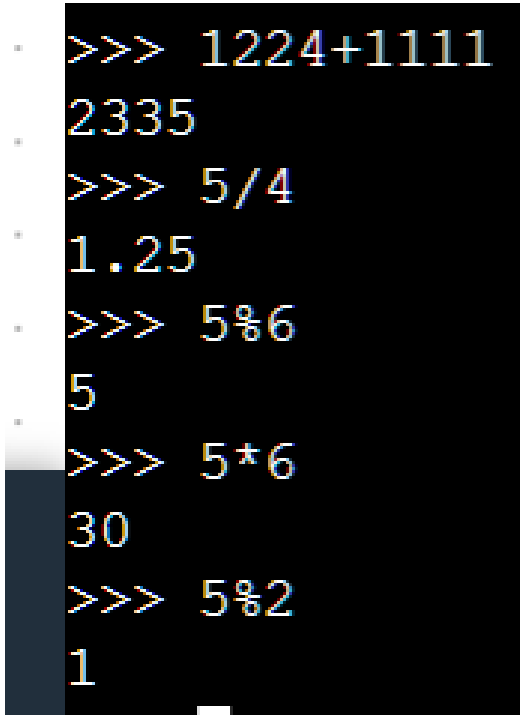
Python 右邊的互動式環境(Shell)

三個 >>> 為開頭

來試看看和 esp32 talk

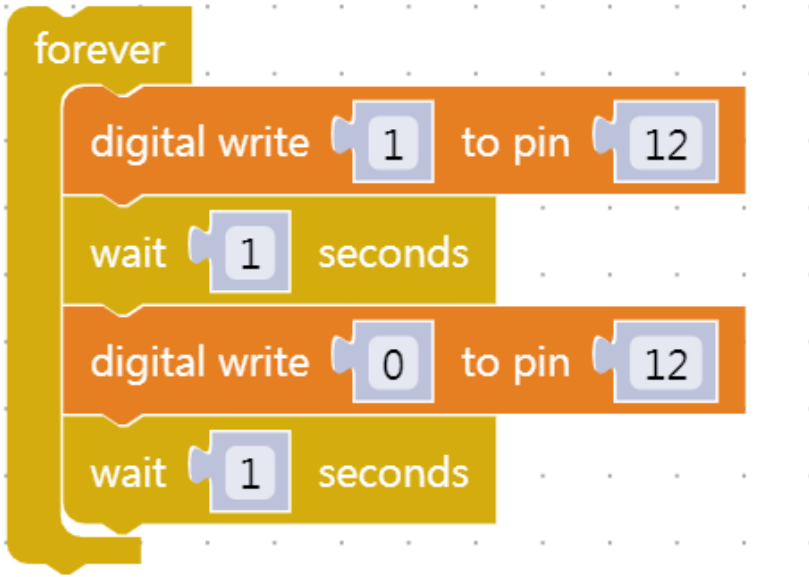
>>> print (“hello”)
hello
>>> 1234+4321
5555
>>>



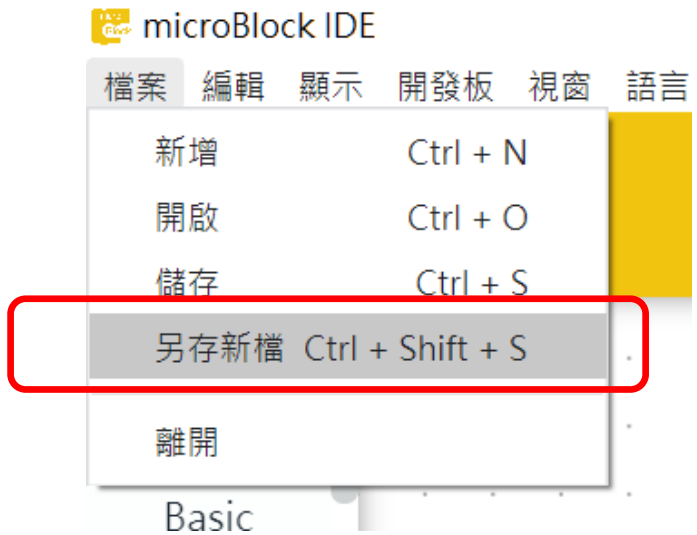


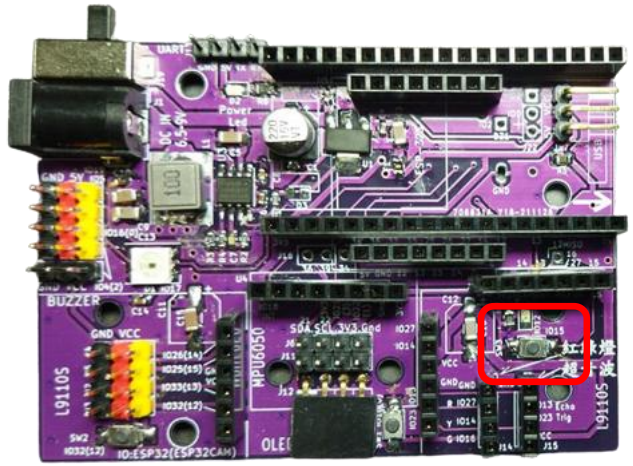
Mason E&D

開始寫第一個程式了
EX1：點亮擴充板上的 LED 燈

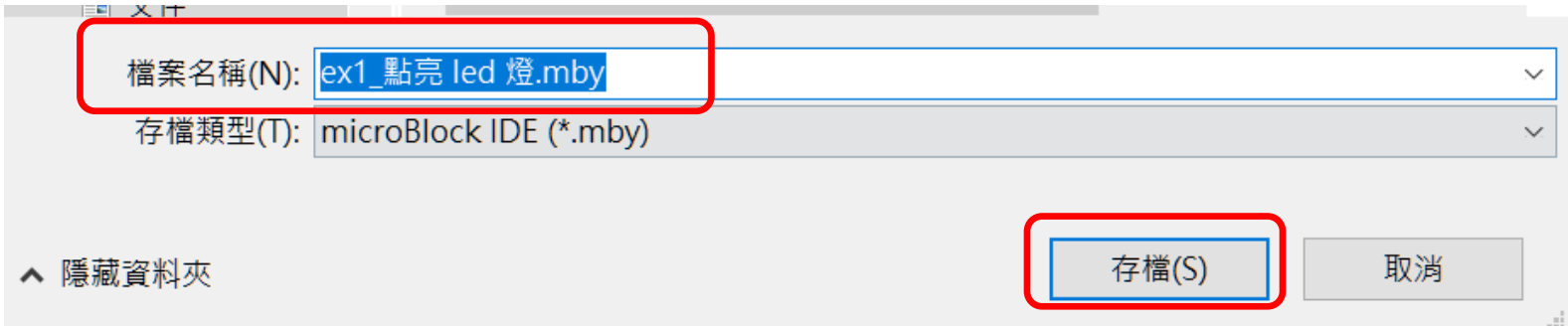


按上傳鍵就完成了



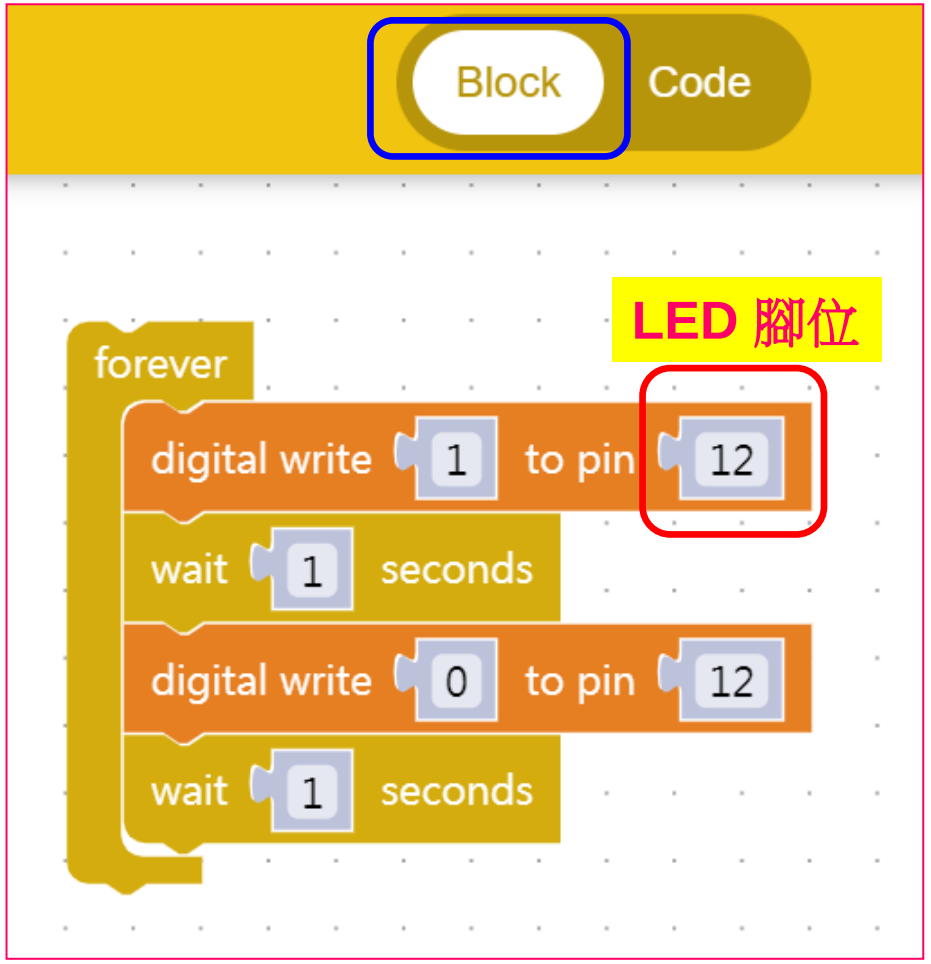


寫上檔名，副檔名為 .mpy



Mason E&D

試著回答下面的問題



Q1 : 如何查看 LED 的腳位?
Q2 : 是寫 “1” 為亮，還是寫 ”0” 為亮?
Q3 : 亮兩秒，暗一秒程式，如何寫?
Q4 : 如何觀看原始的 python code?

microBlock

Block Code

```
1  from machine import Pin
2  from time import sleep
3
4
5  while True:
6      Pin(12, Pin.OUT).value(1)
7      sleep(1)
8      Pin(12, Pin.OUT).value(0)
9      sleep(1)
10
```

Mason E&D

Python 的原始碼

數位輸出：讓擴充板上的內建 LED (IO12) “重覆輪流亮、暗”

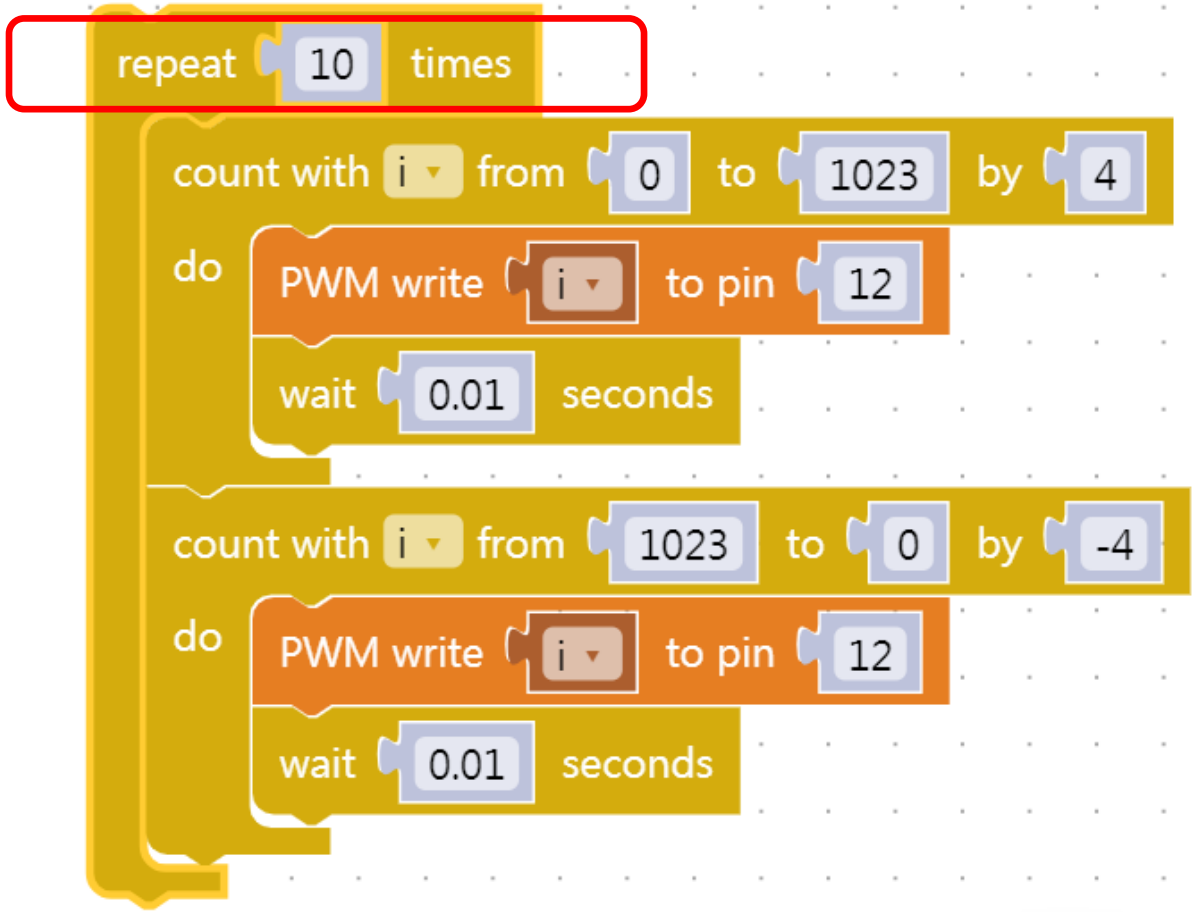
```
from machine import Pin  # 從 machine 模組匯入 Pin 物件
from time import sleep  # 從 time 模組匯入 sleep 物件


while True:    # 一直重覆執行
  Pin(12, Pin.OUT).value(1)  # 設定 IO12 腳位為輸出，並給值為 “1” -> 亮
  sleep(1)                   # 等待 1 秒
  Pin(12, Pin.OUT).value(0) # 設定 IO12 腳位為輸出，並給值為 “0” -> 暗
  sleep(1)                  # 等待 1 秒
```

Mason
E&D

EX2：製作 LED 呼吸燈

利用類比輸出 **PWM** 佔空比的不同，
來改變 **LED** 的亮度，製作出呼吸燈的效果
(由暗漸漸變亮，再漸漸變暗**)**



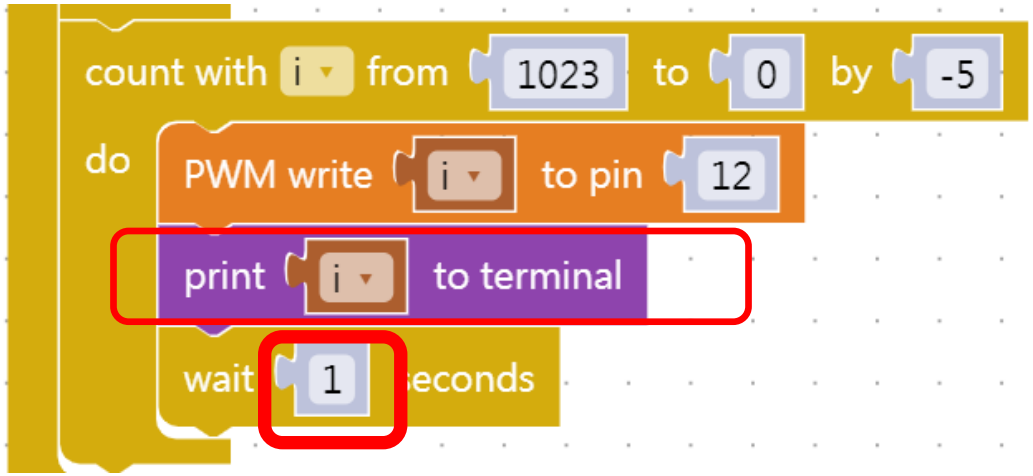
ESP32 為 **10** 位元，可表示 **1024** 階
-> 0-1023

0: 代表最暗
1023：代表最亮

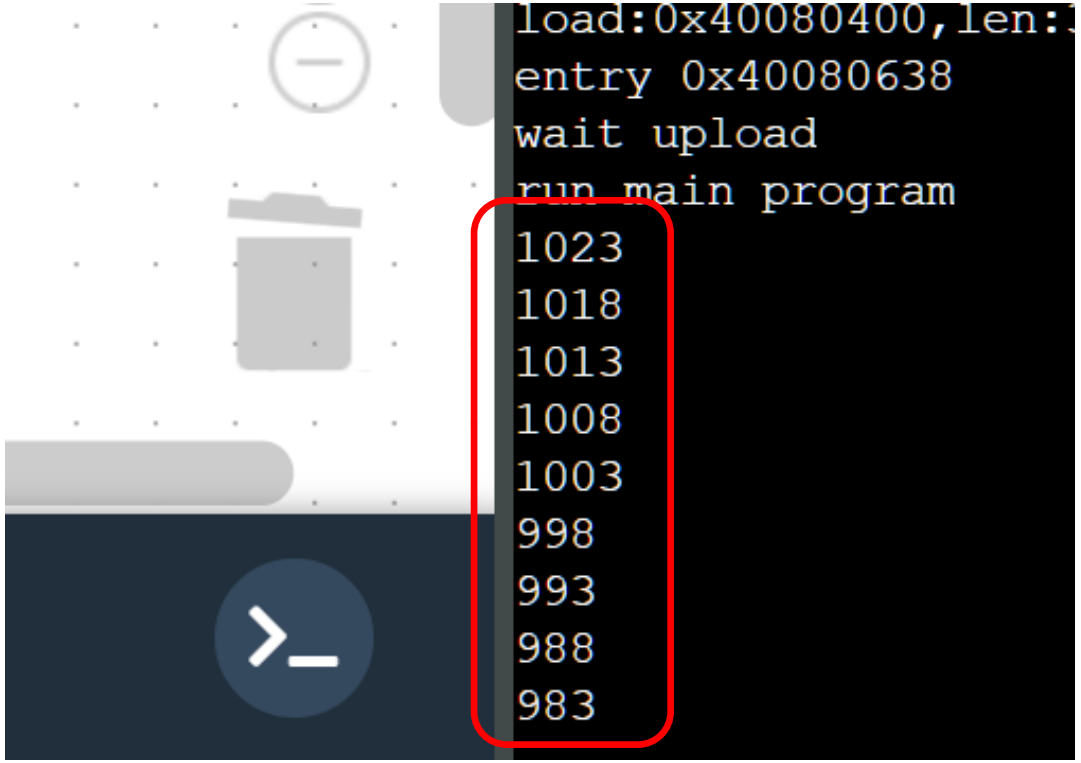
Mason E&D

Q : 如何利用右邊的視窗來"觀察" 值的變化?

利用 **print** 印出想看的值 **"i"** 到右邊 **Terminal** 上



等待 1 秒才方便觀看數值的變化



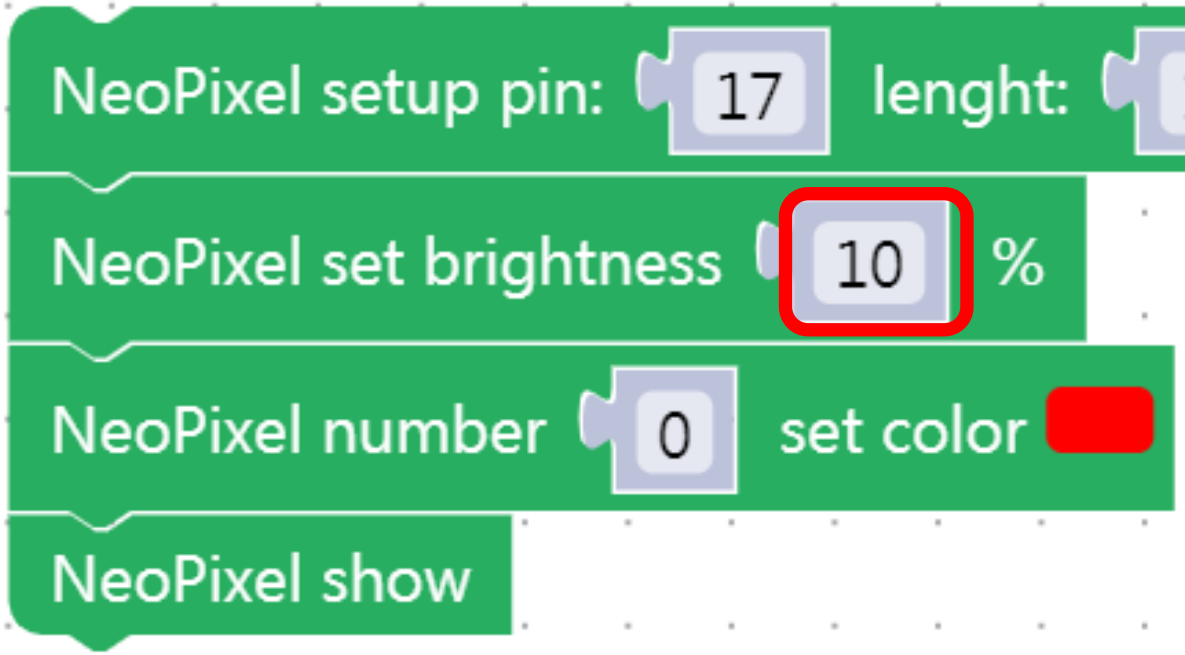
確實如預期的從 1023 每次減 5

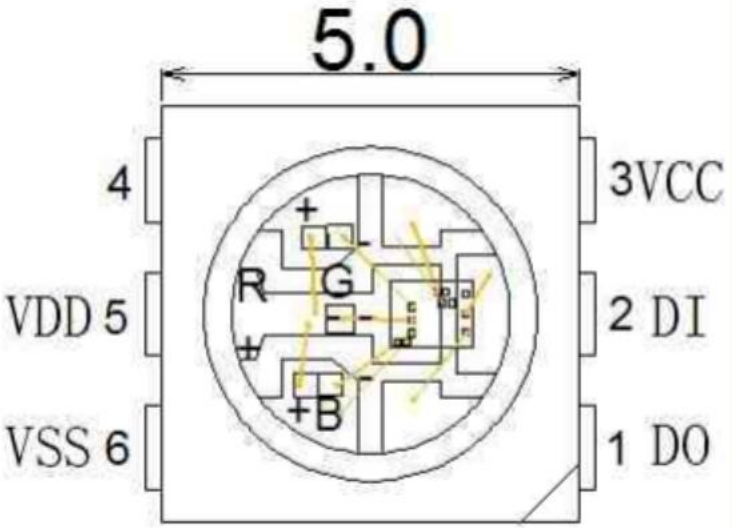
Mason E&D

EX3：點亮內建的 WS2812 全彩 LED

NeoPixel

1. WS2812 腳位在哪？ IO17
2. 利用 NeoPixel 燈條積木來控制 LED 顏色和亮度





1. 建議亮度要設定低一點 10 ~20%
不然有可能會傷害眼睛
2. 裏面其實含三原色 RGB 的 LED組成

Mason E&D

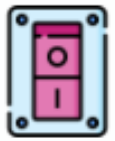
NeoPixel

Q : 試看看寫出 “R->G->B 循環亮五次” 的程式

1. 要記得有 **NeoPixel show** 積木
   才會真正顯示
2. **NeoPixel clear + NeoPixel show**
   會讓 **LED** 關閉，不然會停留在最後的狀態

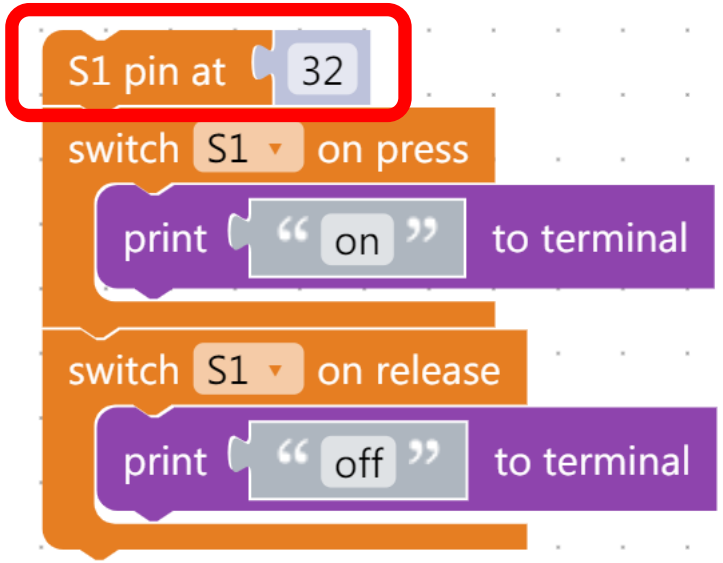
Mason E&D

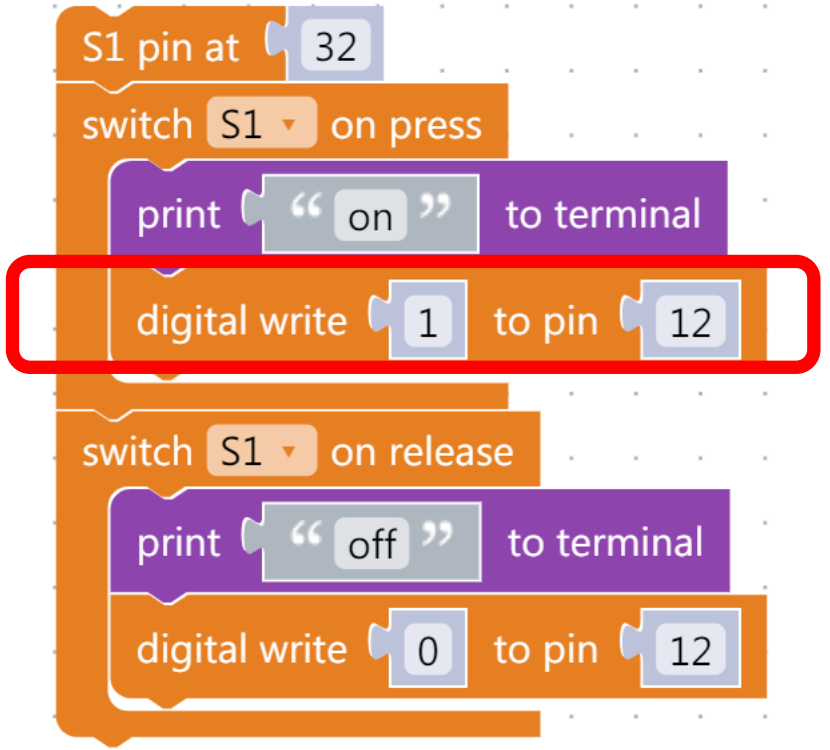
EX4：利用按鈕來控制內建 LED



Switch

1. LED 腳位在哪？ IO12
2. 按鈕腳位在哪？ 左下 SW2: IO32 右中 SW3: IO15
3. 利用 Switch (開關) 積木來控制 LED 亮暗





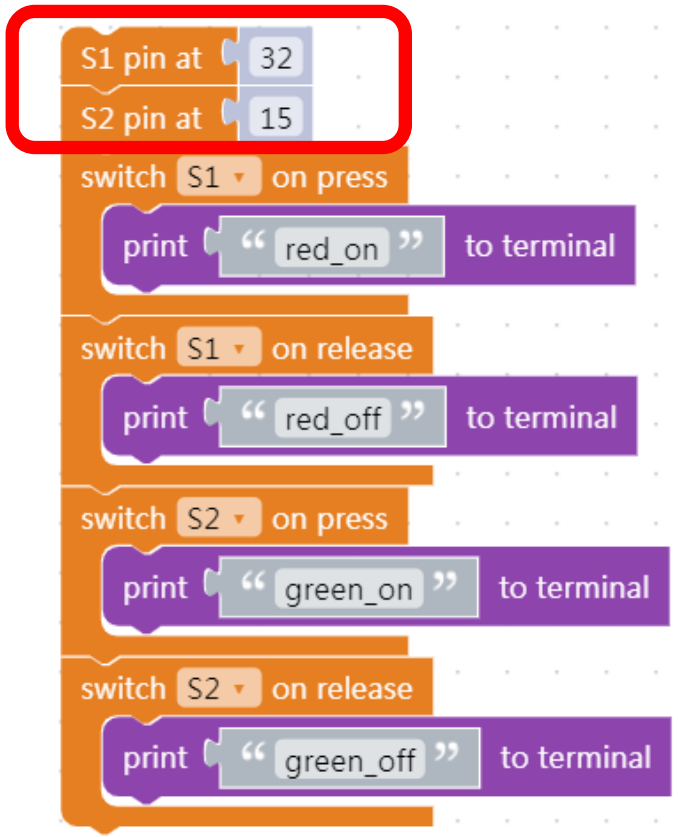
Mason E&D

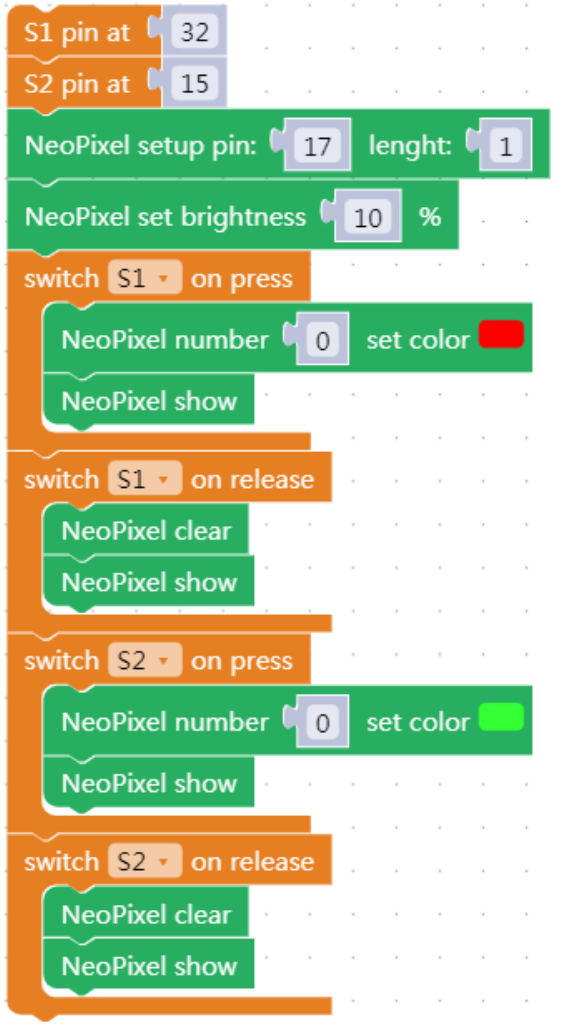
NeoPixel

Switch

EX5：
左按鈕-> 紅燈
右按鈕-> 綠燈

1. WS2812 腳位在哪？ IO17
2. 按鈕腳位在哪？ 左下 SW2: IO32 右中 SW3: IO15
3. 利用 Switch (開關) 積木來控制 WS2812 顏色



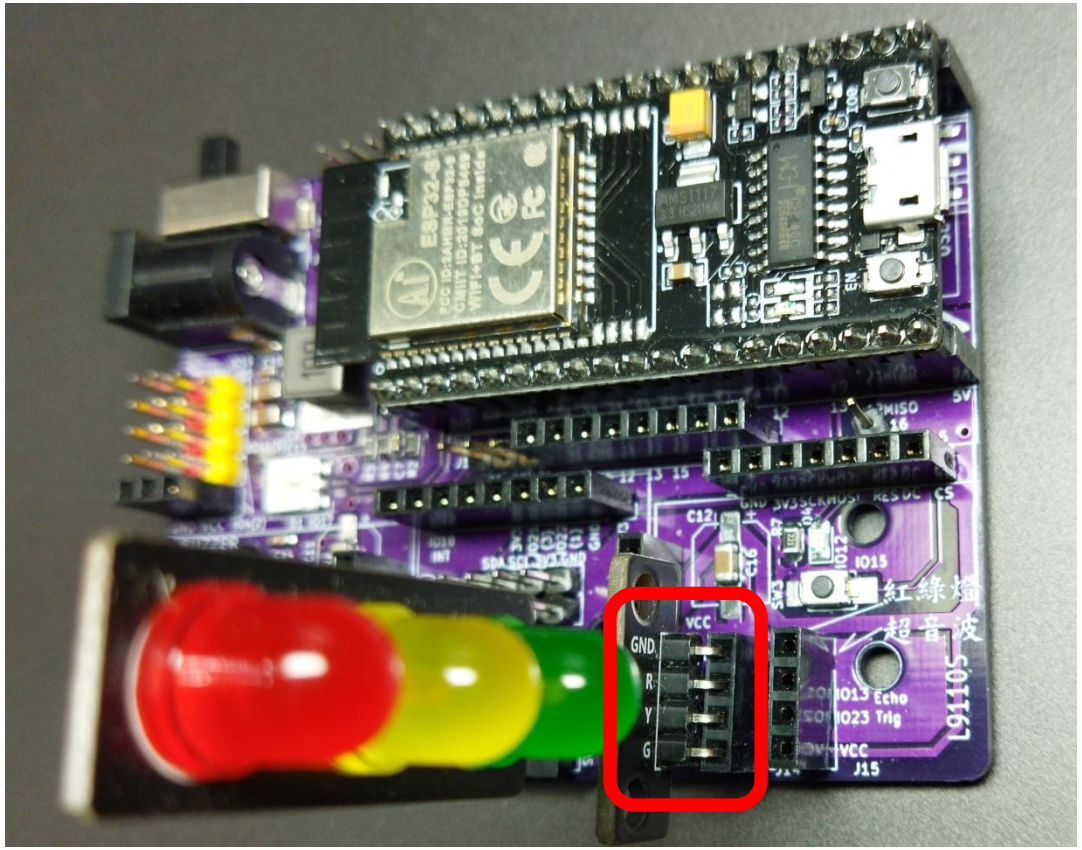


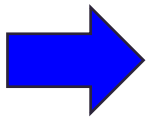
Mason E&D

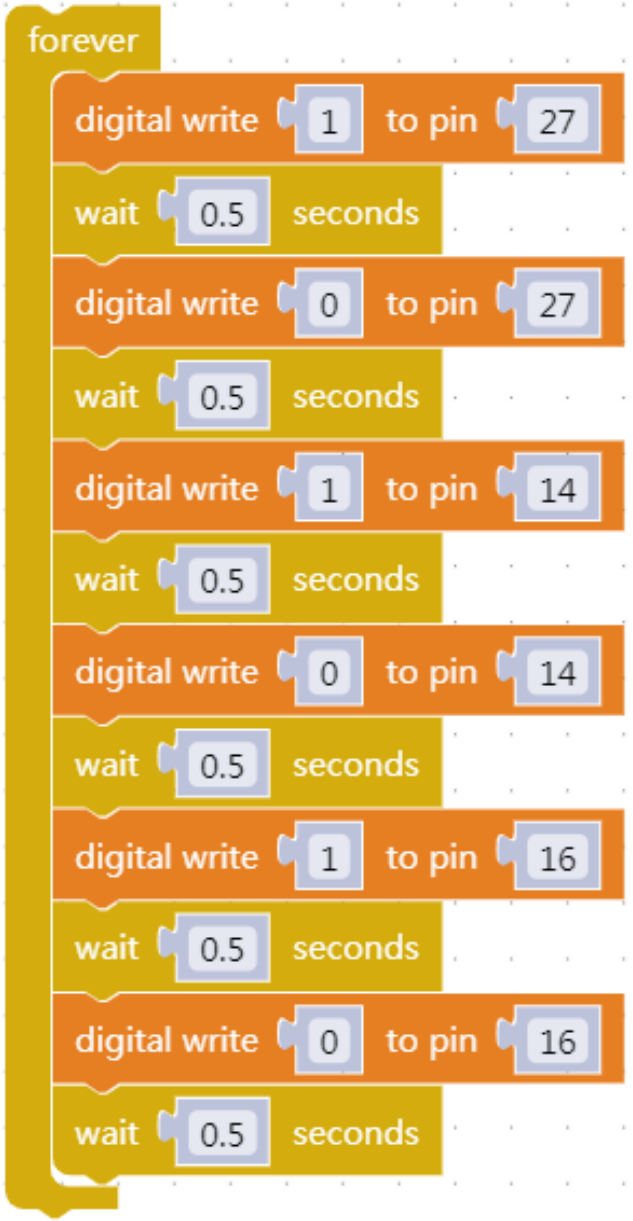
EX6：控制紅綠燈三色模組

1. 三個顏色的腳位在哪？ **R-IO27 Y-IO14 G-IO16**
2. 利用 **EX1** 學的數位輸出方式，即可來控制三顆 **LED**

插拔硬體模組時，請記得先移除電源
且注意正負極方向

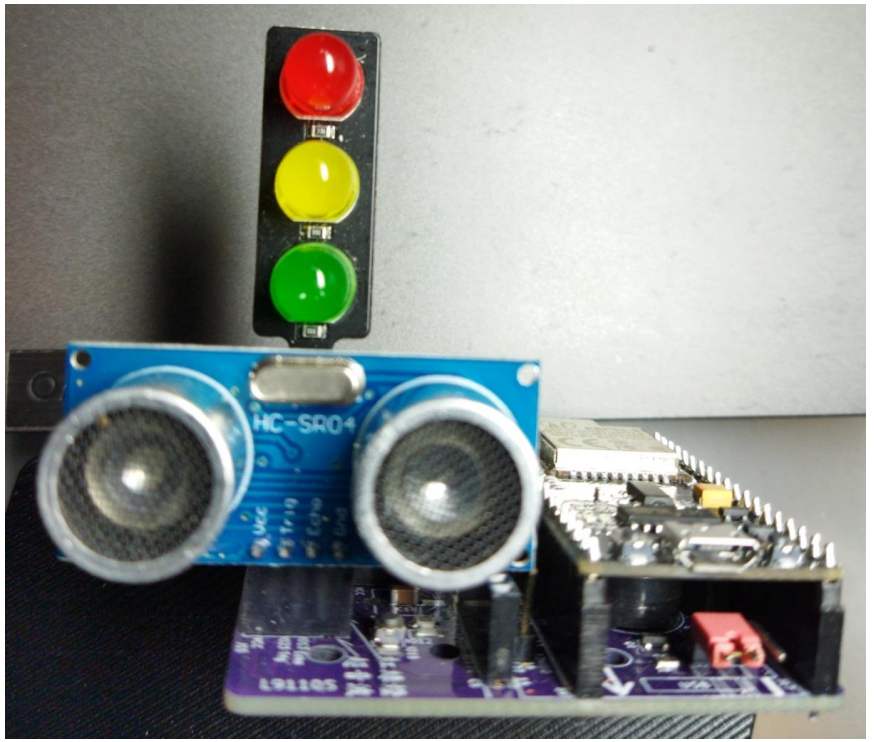






EX7：使用 HC-SR04 超音波模組

1. 腳位在哪? Trig-IO23 Echo-IO13
2. 利用超音波模組積木，即可來測距離



Ultrasonic

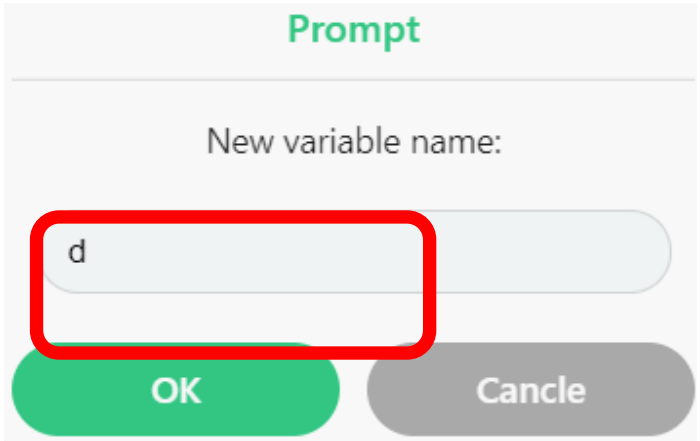
插拔硬體模組時，請記得先移除電源且注意正負極方向

Mason E&D

EX8-1：不同距離顯示不同的燈號

Ultrasonic

1. 三個顏色的腳位在哪? **R-IO27 Y-IO14 G-IO16**
2. 使用變數 **d** 代表量到距離
3. 定義條件，單位 **cm:**
   <8 紅燈，**>8 且 <=15** 黃燈，**> 15** 綠燈



先寫滿足紅燈的條件

Mason E&D

EX8-2：不同距離顯示不同的燈號

Ultrasonic

1. 三個顏色的腳位在哪?
   R-IO27 Y-IO14 G-IO16
2. 使用變數 d 代表量到距離
3. 定義條件，單位 cm:
   <8 紅燈，>8 且 <=15 黃燈，> 15 綠燈

綠燈較不明顯，要從正面來看

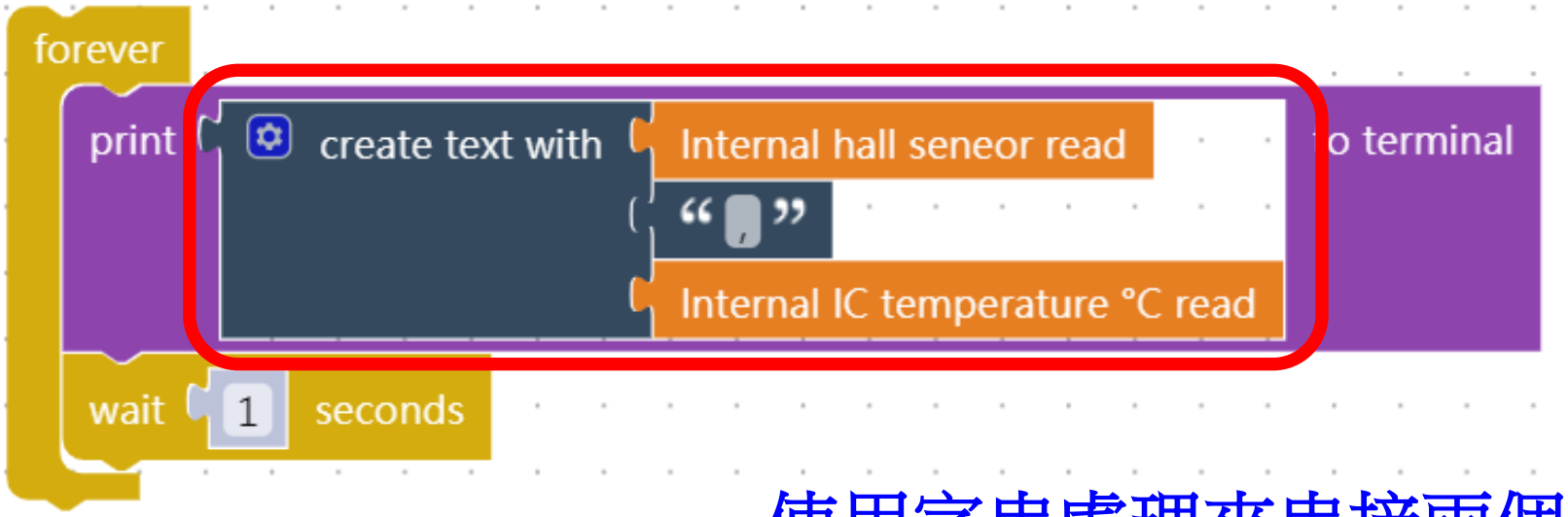
完整的三個條件

Mason E&D

Advanced

EX9：讀取內部的溫度及霍爾感測器

1. 腳位在哪? 皆為 **ESP32** 內建 **sensors**
2. 使用 **print** 來顯示數值



使用字串處理來串接兩個值

可觀察磁鐵 **N** 極和 **S** 極量到的霍爾值是否相同?

Mason E&D

Advanced

EX10 : 讀取內部的 Touch PAD 的值

1. 腳位在哪? IO4/13/15/32/33
2. 使用 print 來顯示數值

只有特定的腳位才有此功能

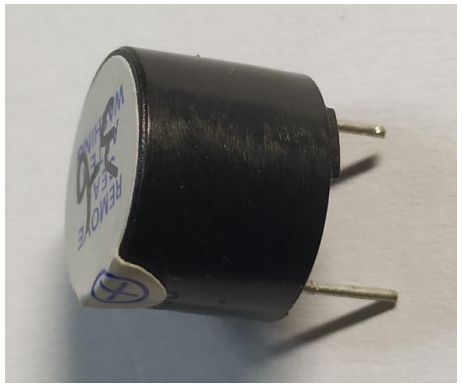
Mason E&D

EX11：
使用內建的無源蜂鳴器，發出聲音

1. 腳位在哪? IO2
2. 使用 **Music** 積木，可播放 **RTTTL** 鈴聲

有源蜂鳴器 -> 有內建震盪源，一上電會發出固定頻率的聲音
無源蜂鳴器 -> 可透過 **PWM** 方式控制其發聲頻率

使用需注意工作電壓及正負極, 常見為 **3.3V** 或 **5V**



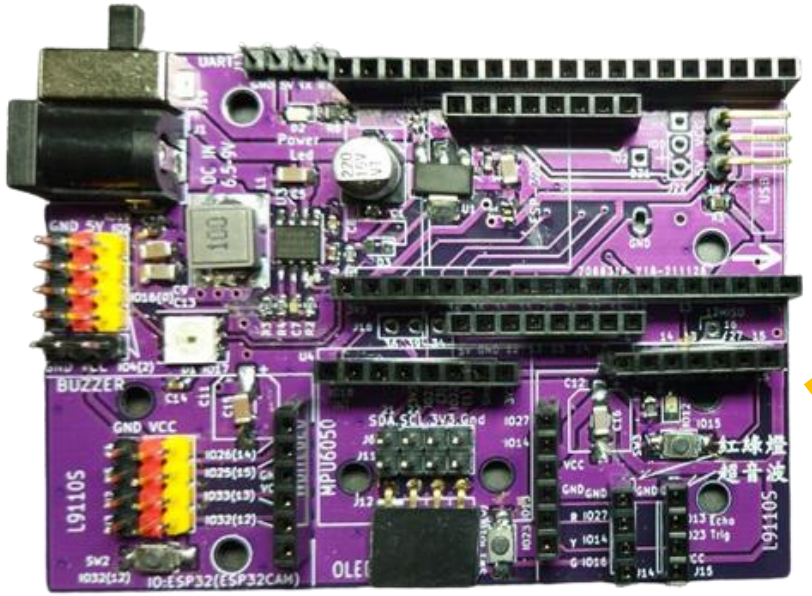
RTTTL格式是之前非智慧型手機的標準的鈴聲格式之一，已經被許多手機所支持 (如: **Nokia**)

Mason
E&D

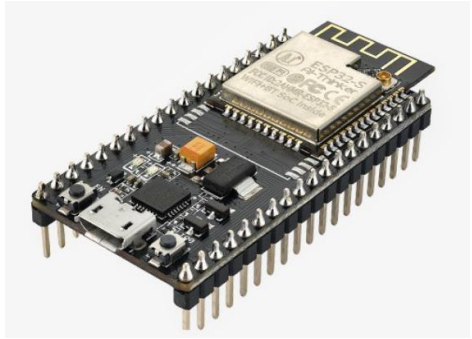
Q & A：相關經驗分享與討論

Mason E&D

分享 Gyro 平台概念

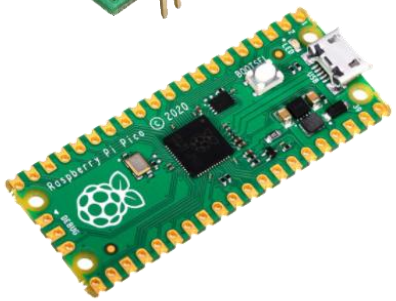


共用 **Gyro** 底板
-> 節省開發的時間和成本

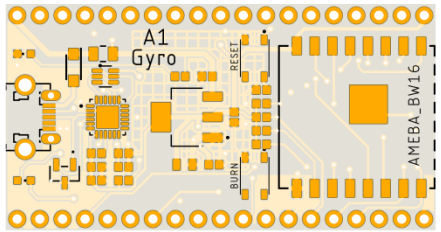


ESP32
ESP32CAM

LinkIt7697-舊有的程式可馬上套用



樹莓派 Pi-Pico
-> 加 WiFi or BT?



Ameba-BW16 模組
- WiFi 5G / BLE 5.0

Mason E&D

Wokwi.com – Esp32-micropython 模擬器

Simulate IoT Projects in Your Browser

https://wokwi.com/arduino/projects/305568836183130690

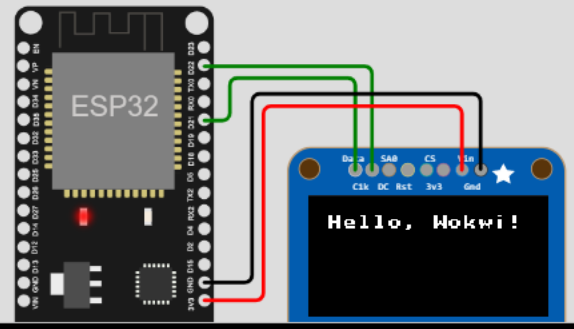
WOKWI SAVE SHARE esp32-micropython-ssd1306 by urish Docs SIGN IN

main.py diagram.json ssd1306.py Library Manager

```
1   from machine import Pin, I2C
2   import ssd1306
3
4   # ESP32 Pin assignment
5   i2c = I2C(0, scl=Pin(22), sda=Pin(21))
6
7   oled_width = 128
8   oled_height = 64
9   oled = ssd1306.SSD1306_I2C(oled_width, oled_height, i2c)
10
11  oled.text('Hello, Wokwi!', 10, 10)
12  oled.show()
13
```

Simulation

00:35.189 99%




```
MicroPython v1.18 on 2022-01-17; ESP32 module with ESP32
Type "help()" for more information.
>>>
```

Mason E&D

課程大綱

時間	項目	內容
03/15 13:00~16:00	MicroPython 基礎入門	ESP32 介紹及硬體 ESP Gyro擴充板入門操作 - MicroPython開發環境設定與韌體上傳 - microBlock 介面與相關積木介紹 - 完成基本 LED/紅綠燈/按鈕及超音波使用
04/06 13:00~16:00	MicroPython 物聯網應用(1)	馬達進階控制使用及物聯網基本應用一 - Timer / ticks 的使用方式 - L9110S 控制直流馬達 - 伺服馬達 SG90 的使用 - 上傳溫溼度資料至雲端平台
05/10 13:00~16:00	MicroPython 物聯網應用(2)	MicroPython 物聯網進階應用二 - 使用雲端平台遠端控制 LED - 取得雲端資訊: 網路時間/天氣資訊 - 結合 NTP / OLED / Openweather 及圖像顯示 OLED 做個人化的天氣時鐘

Thank You!