



MICRO:BIT BLOCKLY LESSONS

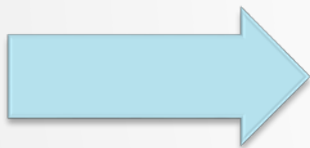
AStar

James Perng 2017.12.14

CONTENTS

1. Micro:bit軟體生態鏈介紹
2. 功能展示短片
3. Blockly 程式課程 (基礎)
4. 行動、平板裝置使用環境介紹
5. 附錄(補充資料)

MICRO:BIT 軟體生態系統介紹



行動應用及遊戲應用開發

App
Inventor

C & C++

Python

雲端及區域連網服務開發

Python

Javascript &
NodeJS

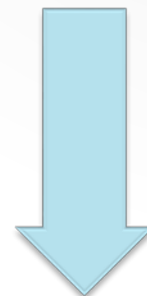


Python

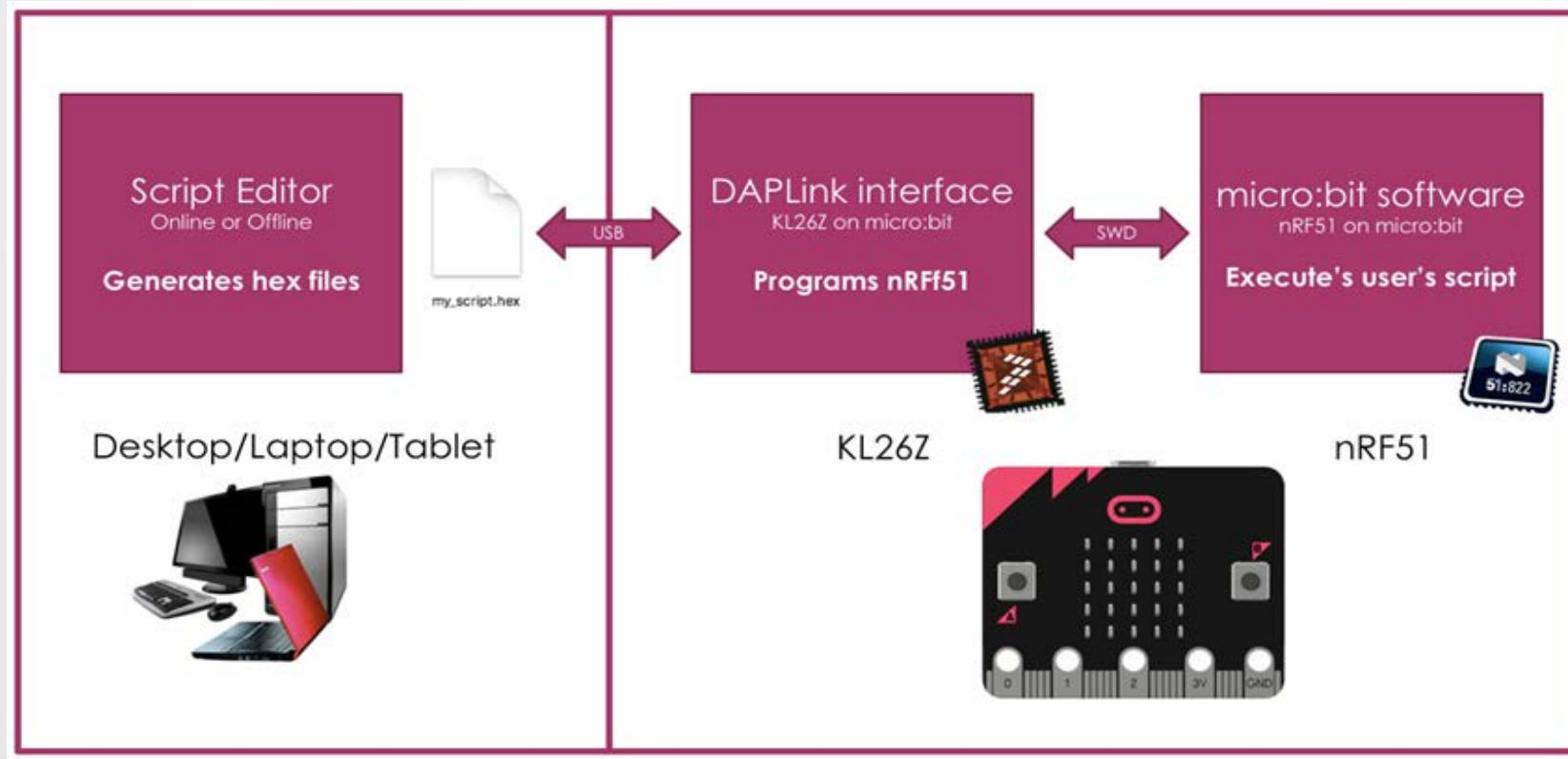
Javascript

Scratch

Blockly



內部機制：

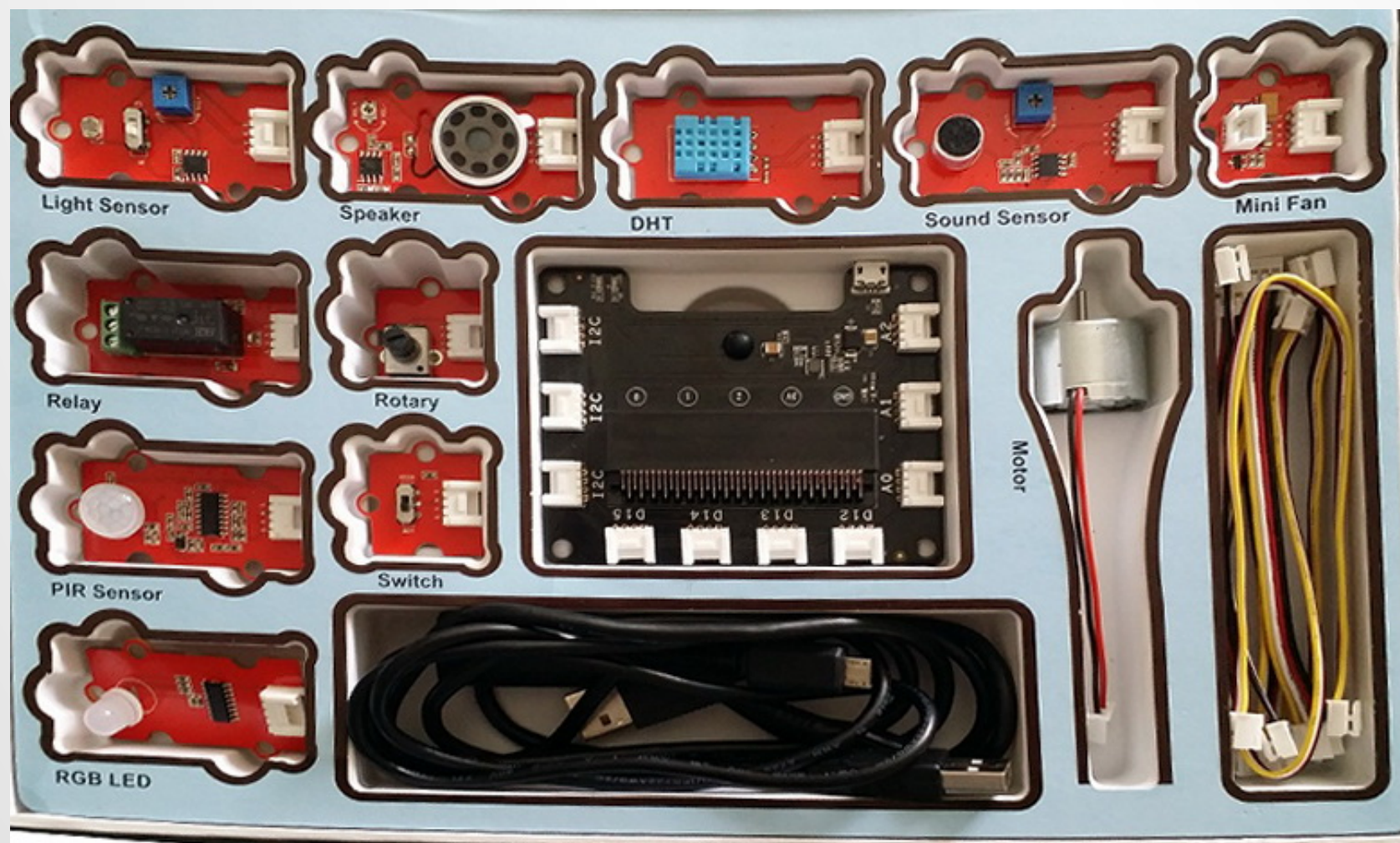


在MICRO:BIT 軟體分成在MICROBIT (TARGET COMPUTER 如上圖右邊)上執行的及主電腦(HOST COMPUTER 如上圖左邊)上兩類：
一般程式寫好後透過USB 轉到MICRO:BIT，整個流程由兩個晶片完成，一是 KL26Z 負責程式刷新(CODE FLASHING)，另一NRF51822則執行完成使用者完成的程式。

功能應用短片

1. 組成多片跑馬燈
2. 自編音樂 (code 6:30~12:34, 聲音14:16~14:32)
3. 藍芽連結控制 (手機、車)
4. 無線連線控制 (螢火蟲)
5. 機器人應用 (樂高_F、五子棋)
6. 加速度計應用 (穩定器、手套)
7. 自走車控制應用
8. 自動澆花裝置
9. 當成搖桿_F 打磚塊遊戲
10. 彩色LED變色控制_F

套件



程式編輯資源介紹

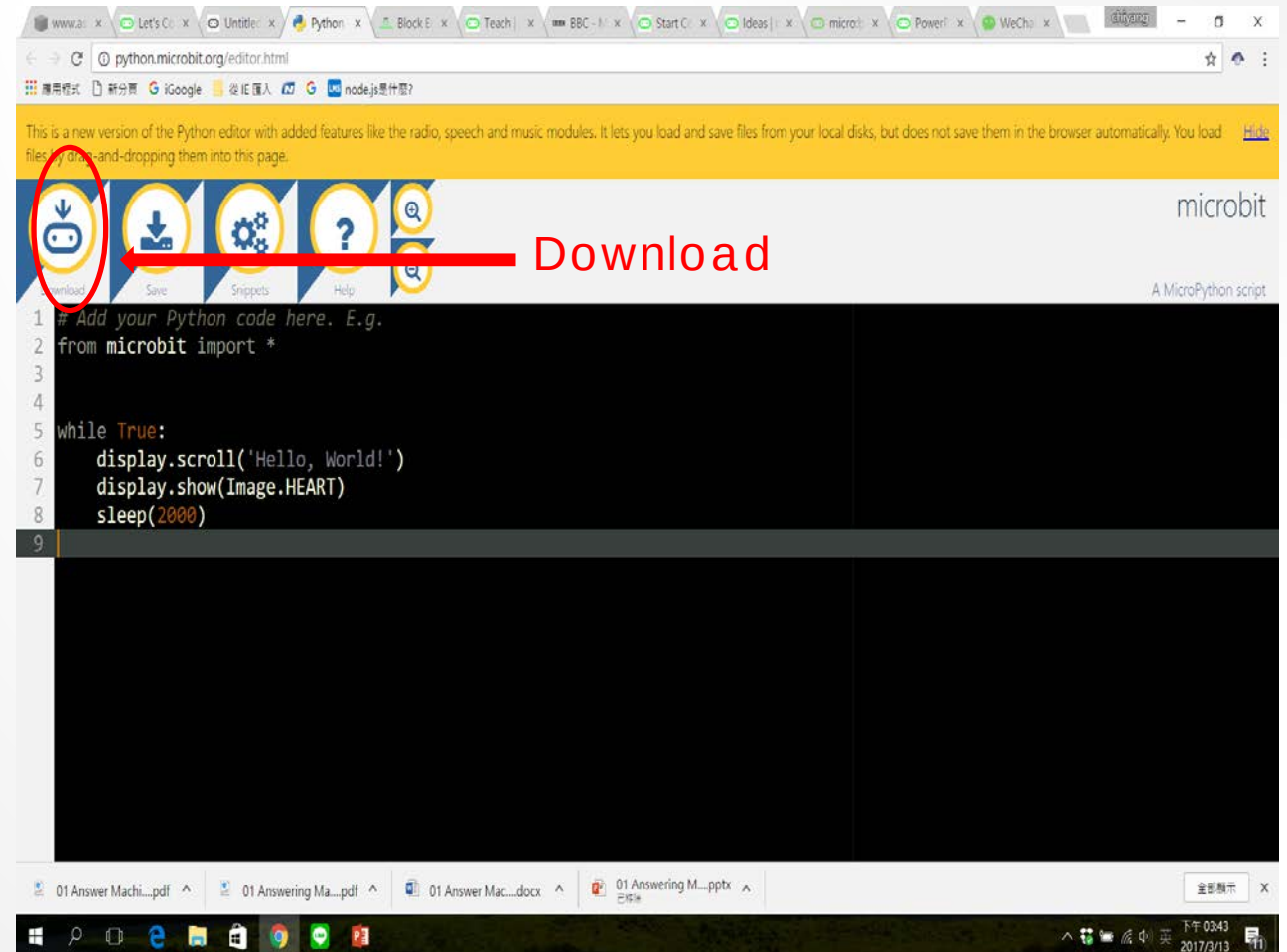
- microbit.co.uk editors based on TouchDevelop backend:
 - [Code Kingdoms Javascript](#)
 - [Microsoft Blocks](#)
 - [Microsoft TouchDevelop](#)
 - [Python](#)
- [mbed Online Compiler](#)
- [Mu offline Python editor](#)
- [PXT for micro:bit](#) (successor to the TouchDevelop-based editors above)
- 全台灣第一本中文書即將開賣：[micro:bit 微型電腦帶著走，使用 Python 輕鬆玩](#) – 作者 陳致中 (含稅、含運、11/8 出貨)

BLOCKLY 程式課程

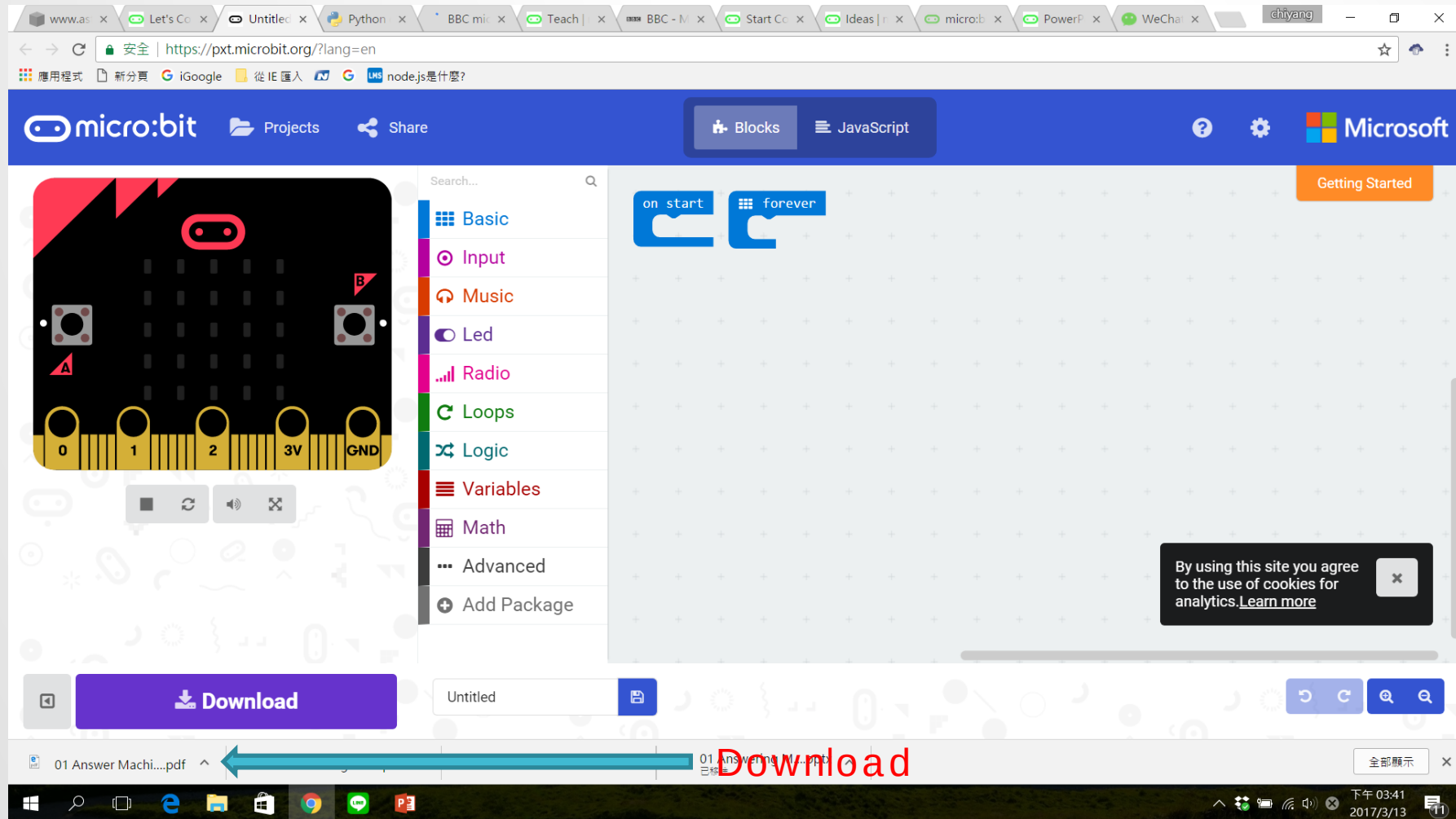
- 從哪邊開始寫程式？
- 編輯環境 & 介面功能介紹
- 多變的LED，好用的無限迴圈
- 剪刀石頭布
- 加入加速度及 if 判斷，我們可以做個micro:bit 骰子
- 幸運餅乾(magic 8)
- 來撥些音樂吧
- 用按鍵做不同的選曲
- 分類帽
- 無線通訊—螢火蟲、投票機
- 溫度、指南針、線上課程練習

如何在MICROBIT 寫程式

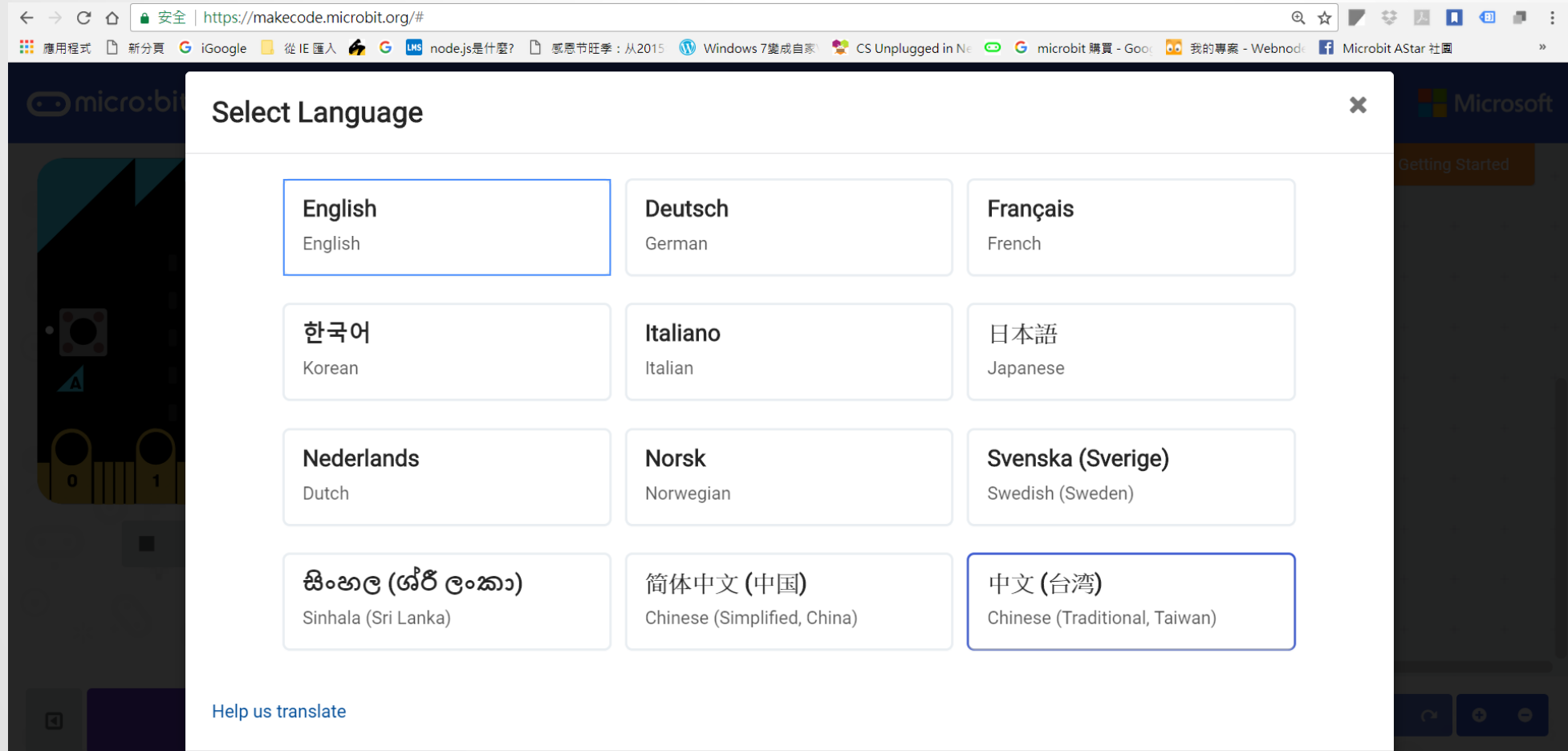
- First step : connect to microbit with Browser (via USB connector and cable)
- Second step : open browser to <http://microbit.org/code/>
- Third step : choose your editor (Java script , Python or MS Block)
- Fourth Step: write your first code and download to your microbit



JAVA SCRIPT BLOCK



HTTPS://MAKECODE.MICROBIT.ORG/



建立/匯入/開啟專案、範例

Blocky & Java
Script code 切換

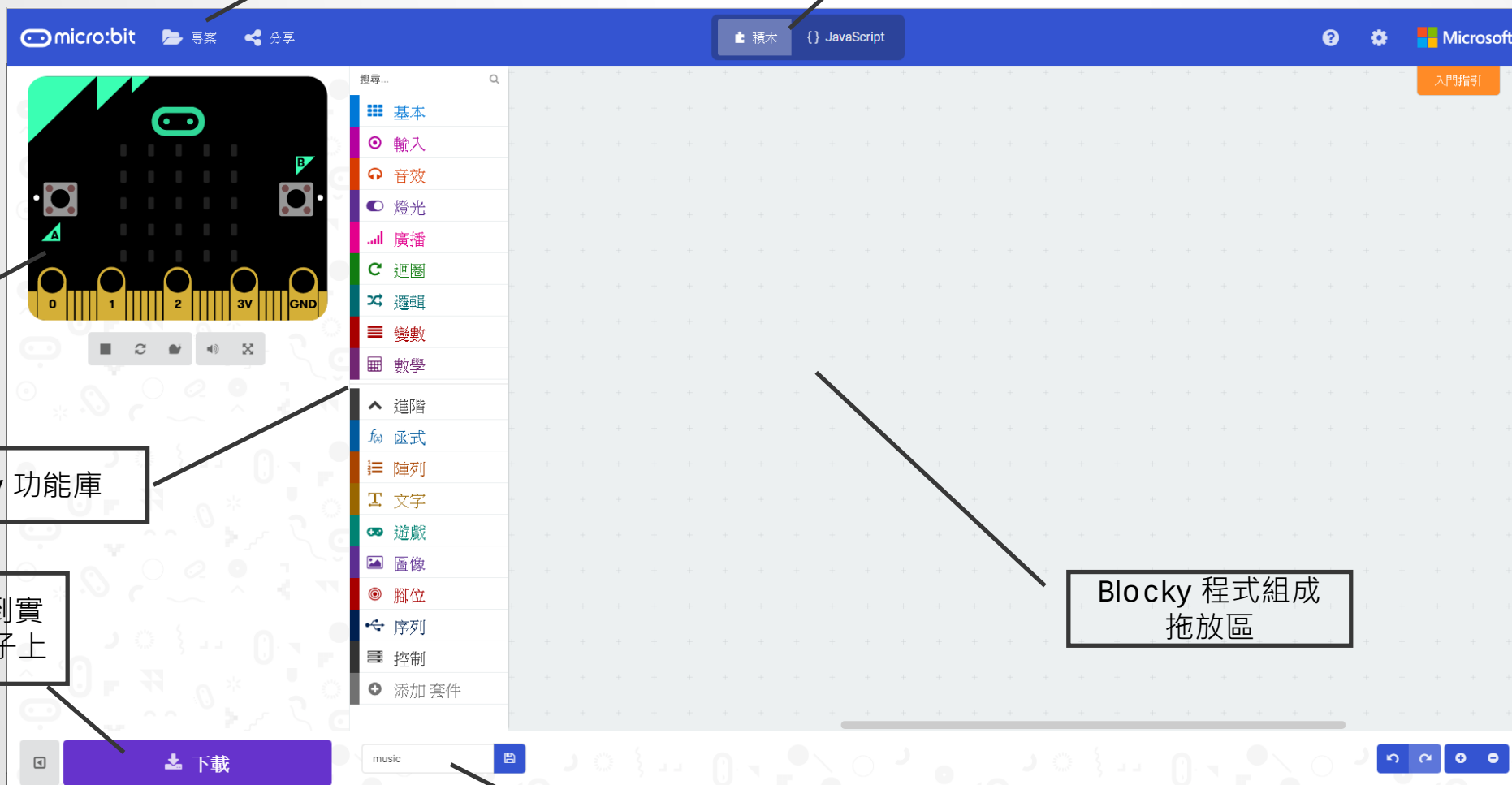
即時模擬器

Blocky 功能庫

將程式編碼傳送到實
際的micro:bit板子上

Blocky 程式組成
拖放區

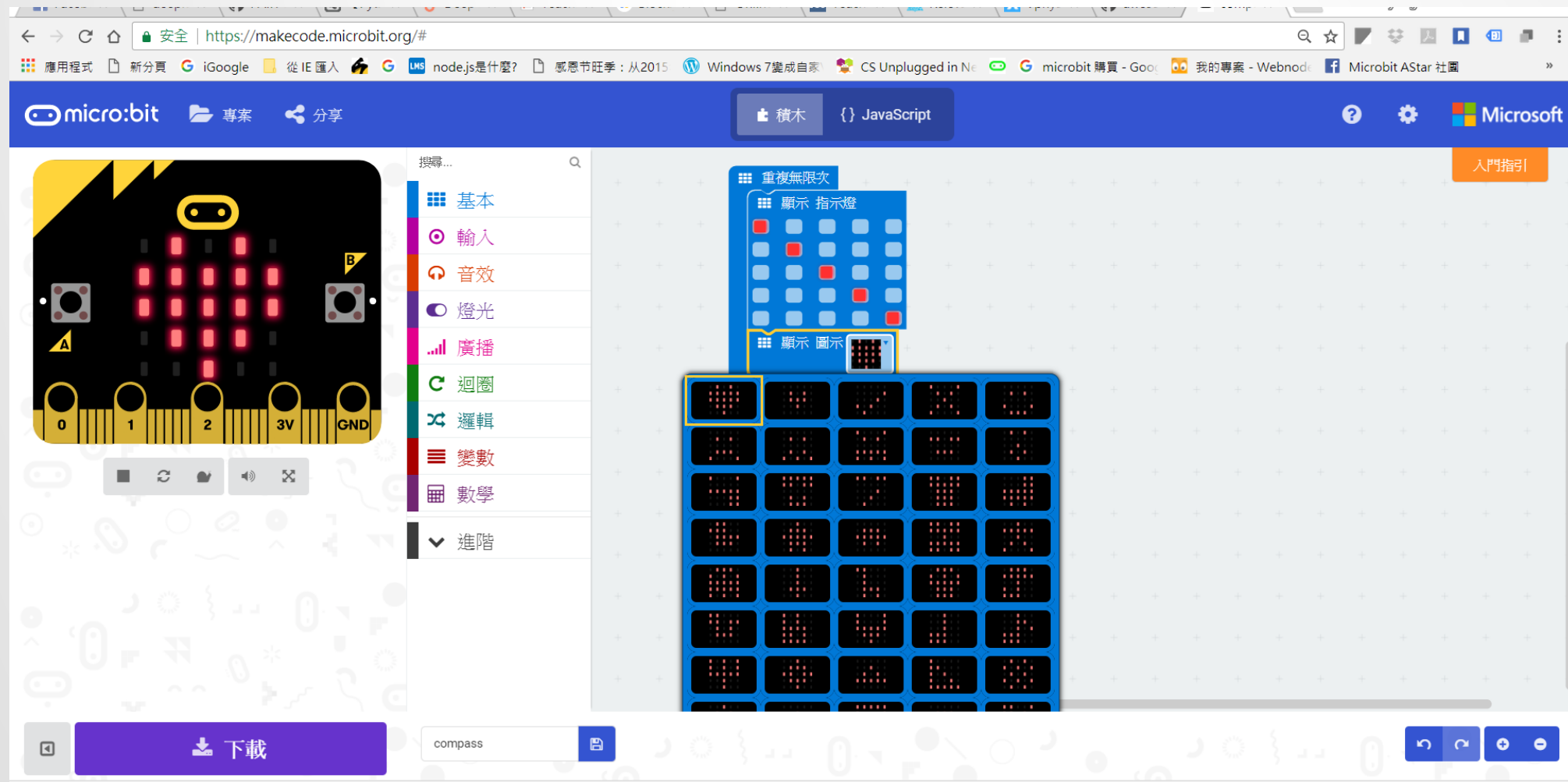
設定專案名稱



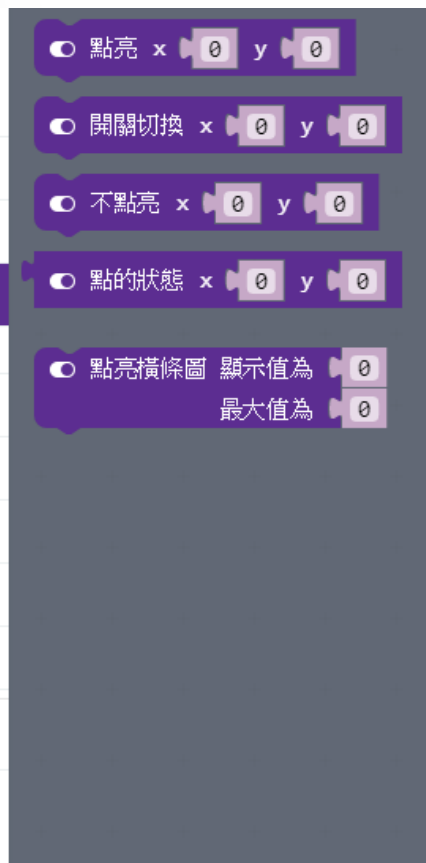
補充概念

- PC base 架構 v.s. 單晶片(SoC、MCU...)架構
- 多執行緒與單執行緒
- 無窮迴圈概念
- 電腦程式概念：宣告、設定在先，判斷、執行在後
- 清楚要電腦做甚麼比怎麼寫程式重要(邏輯 vs. 技巧)
- 流程圖是好工具(目的->解決方法->步驟->調整)

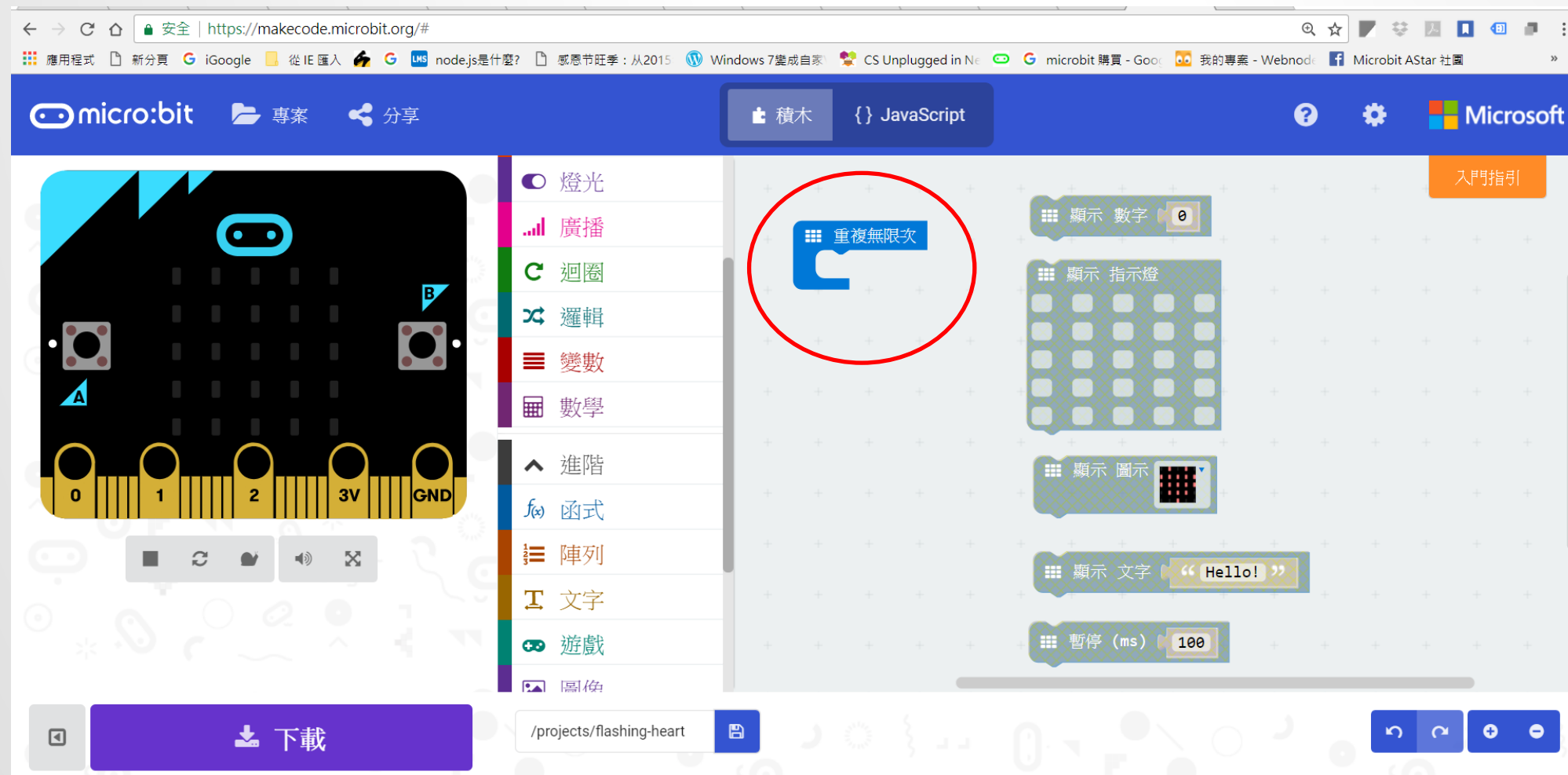
多變的LED



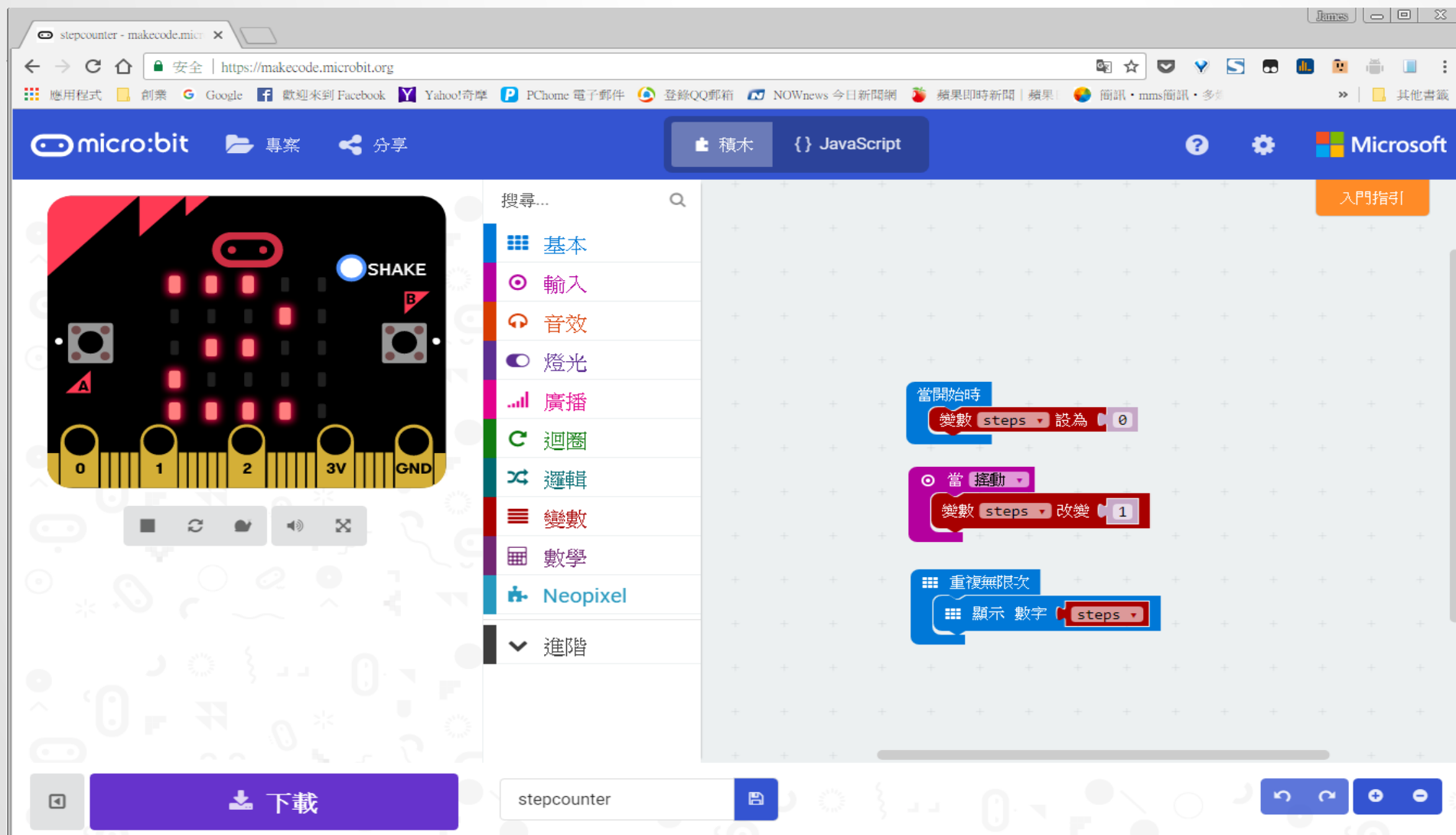
LED顯示功能介紹



好用的無限迴圈



健康計步器範例 – 無限迴圈, SHAKE



LED與邏輯判斷範例 – IF_THEN_ELSE、按鍵

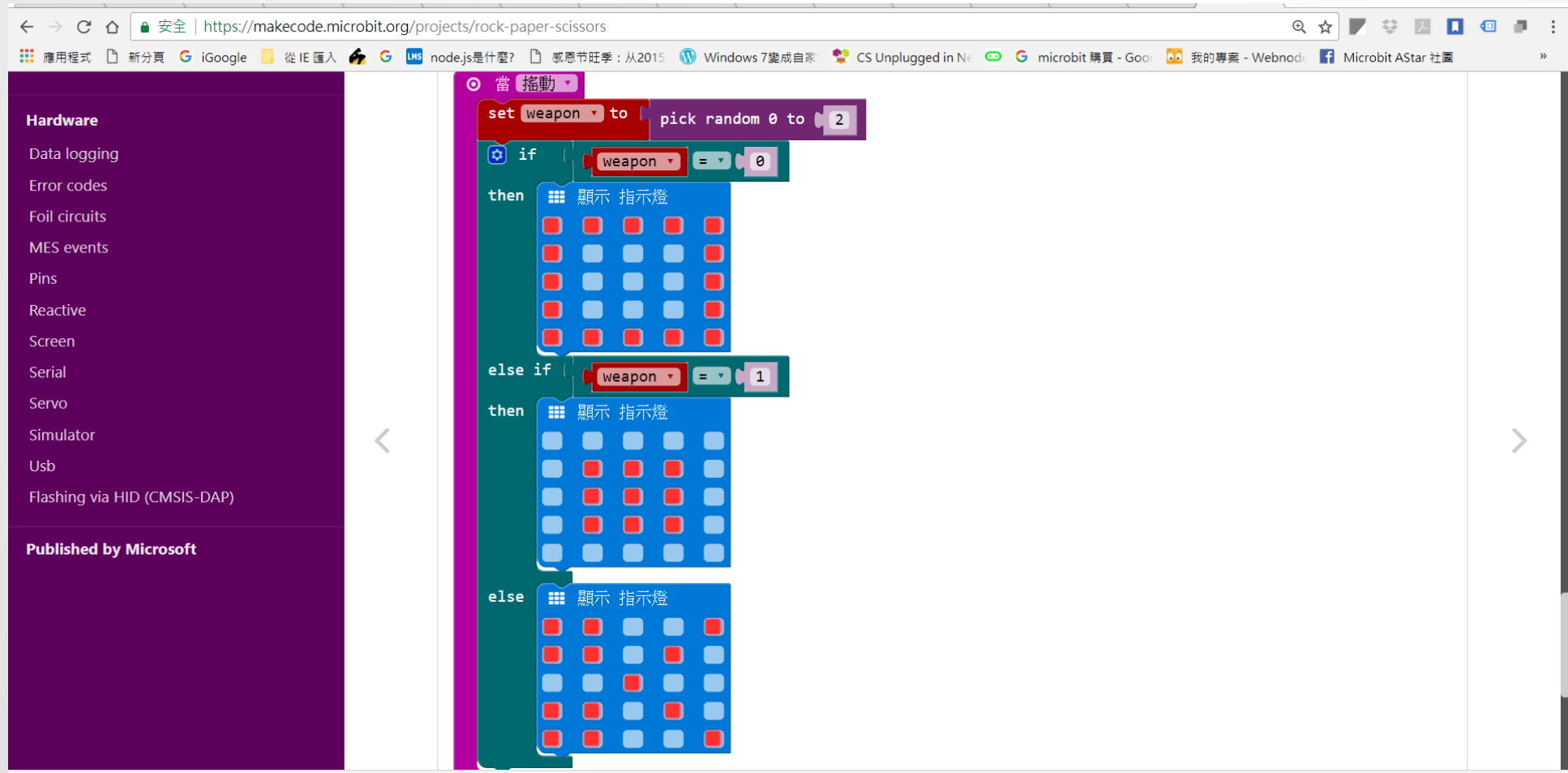
The screenshot displays the Microsoft MakeCode Micro:bit IDE interface. The top navigation bar includes the 'micro:bit' logo, a '專案' (Project) button, a '分享' (Share) button, a '積木' (Blocks) button, a 'JavaScript' button, and a 'Microsoft' logo. The left sidebar features a search bar and a category menu with options: 基本 (Basic), 輸入 (Input), 音效 (Sound), 燈光 (Light), 廣播 (Broadcast), 迴圈 (Loops), 邏輯 (Logic), 變數 (Variables), 數學 (Math), Neopixel, and 進階 (Advanced). The main workspace shows a Micro:bit board on the left with a grid of LEDs. The central area contains a code editor with the following blocks:

- 當開始時** (When started) block containing a **顯示 文字** (Show text) block with the text "Hello World!".
- 重複無限次** (Repeat forever) loop containing two **當 執行** (When executed) blocks, each with a **鍵被按下?** (Key pressed?) block and a **顯示 指示燈** (Show LED) block.
- 暫停 (ms)** (Pause in ms) blocks with a value of 500.
- 清除 屏幕** (Clear screen) blocks.

The bottom of the interface includes a '下載' (Download) button, a text input field containing 'LED-Loop', and a '保存' (Save) button.

剪刀、石頭、布範例 - 變數與亂數

(SCISSORS、STONE、CLOTH)



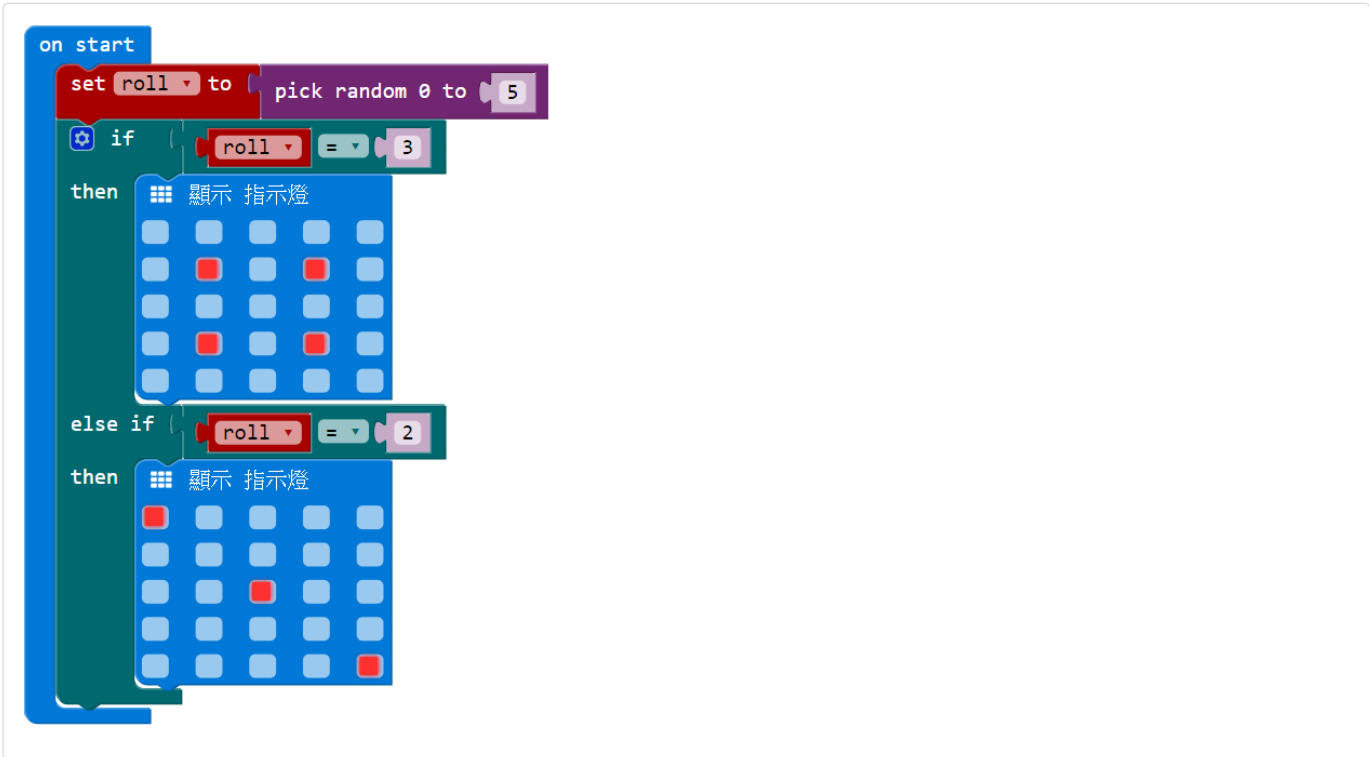
倒數計時 – WHILE、變數設定

The screenshot displays the Microsoft MakeCode micro:bit editor interface. The top blue header bar contains the 'micro:bit' logo, a '專案' (Project) button, a '積木' (Blocks) button, a 'JavaScript' button, and icons for help, settings, and the Microsoft logo. On the left, there is a visual representation of the micro:bit board and a sidebar menu with categories: 基本 (Basic), 輸入 (Input), 音效 (Sound), 燈光 (Light), 廣播 (Broadcast), 迴圈 (Loops), 邏輯 (Logic), 變數 (Variables), and 數學 (Math). The main workspace shows a script for a countdown timer. It begins with a '當 A 鍵被按下' (When A button pressed) event block. This is followed by a '變數 i 設為 9' (Set variable i to 9) block. A green '當' (When) loop block contains three actions: '顯示 數字 i' (Show number i), '暫停 (ms) 100' (Pause 100 ms), and '變數 i 設為 i - 1' (Set variable i to i - 1). The bottom of the interface includes a '下載' (Download) button, a file name input field containing 'CountDown-1', and a '保存' (Save) button. On the far right, there is an '入門指引' (Getting Started) button.

EX: 擲骰子範例(不是完整程式碼)

安全 | <https://makecode.microbit.org/lessons/dice-roll/quiz-answers>

應用程式 新分頁 iGoogle 從 IE 匯入 node.js 是什麼? 感恩节旺季: 从2015 Windows 7變成自家 CS Unplugged in Ne microbit 購買 - Google 我的專案 - Webnode Microbit AStar 社團



The image shows a Scratch code editor with the following code:

```
on start
  set roll to pick random 0 to 5
  if (roll = 3)
  then
    顯示 指示燈
  else if (roll = 2)
  then
    顯示 指示燈
```

The code is written in a block-based language. The 'on start' block contains a 'set roll to pick random 0 to 5' block. This is followed by an 'if' block with the condition '(roll = 3)'. The 'then' block of the 'if' block contains a '顯示 指示燈' (Show indicator light) block. Below the 'if' block is an 'else if' block with the condition '(roll = 2)'. The 'then' block of the 'else if' block also contains a '顯示 指示燈' (Show indicator light) block. The '顯示 指示燈' block is a 5x5 grid of blue squares with red squares indicating the lit positions. For the 'if' block, the lit positions are at (1,2), (2,2), (3,2), (4,2), and (5,2). For the 'else if' block, the lit positions are at (1,1), (2,3), (3,4), (4,5), and (5,5).

幸運餅乾(MAGIC-8)

← → ↻ 🏠 安全 | <https://makecode.microbit.org/lessons/magic-8/challenges> 🔍 ☆

📁 應用程式 📄 新分頁 🌐 iGoogle 📄 從 IE 匯入 📄 LMS node.js是什麼? 📄 感恩节旺季: 从2015 📄 Windows 7變成自家 📄 CS Unplugged in Ne 📄 microbit 購買 - Goog 📄 我的專案 - Webnode 📄 Microbit AStar 社團 »

NeoPixel
File System
Sonar
Bluetooth Temperature Sensor
MAX6675
Bluetooth MAX6675
MIDI
Bluetooth MIDI
Code Dojo Olney
UCL Junk Robot
SRS BitBot

Blocks
On Start
▶ Loops
▶ Logic
▶ Variables
Math
JavaScript blocks
Custom blocks

🔴 當 搖動

- 清除 屏幕
- set randomNumber to pick random 0 to 3
- if randomNumber = 4
 - then 顯示 文字 “ DEFINATELY ”
- else if randomNumber = 3
 - then 顯示 文字 “ TRY AGAIN ”
- else if randomNumber = 2
 - then 顯示 文字 “ YES ”
- else if randomNumber = 1
 - then 顯示 文字 “ NO ”
- else 顯示 文字 “ I DON'T KNOW ”

顯示 數字 8

on start

- 顯示 文字 “ ASK A QUESTION ”
- 顯示 數字 8

📄 📄 ▶ 📄

EXERCISE TIME - 英文版的神奇八號球預設會有下列答案：

1. It is certain (這是必然)
2. It is decidedly so (肯定是的)
3. Without a doubt (不用懷疑)
4. Yes, definitely (毫無疑問)
5. You may rely on it (你能依靠它)
6. As I see it, yes (如我所見，是的)
7. Most likely (很有可能)
8. Outlook good (外表很好)
- 9 Yes (是的)
10. Signs point to yes (標誌指出「是的」)
11. Reply hazy try again (回覆攏統，再試試)
12. Ask again later (待會再問)
13. Better not tell you now (最好現在不告訴你)
14. Cannot predict now (現在無法預測)
15. Concentrate and ask again (專心再問一遍)
16. Don't count on it (想的美)
17. My reply is no (我的回覆是「不」)
18. My sources say no (我的來源說「不」)
19. Outlook not so good (外表不太好)
20. Very doubtful (很可疑)

其中十個答案是肯定(●)，5個是否定(●)，5個是模稜兩可或提示再問一次(●)。

分類帽範例 - 搖動, 按鍵

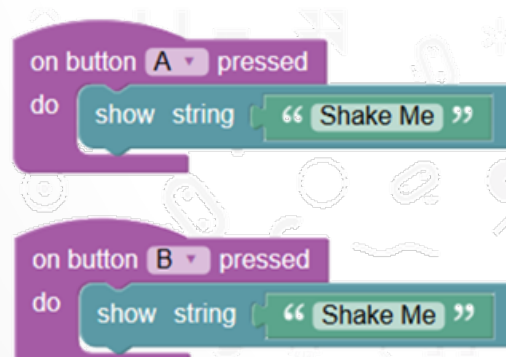
Block Programming

Here is the code to implement the Sorting Hat algorithm:



Resetting the Sorting Hat

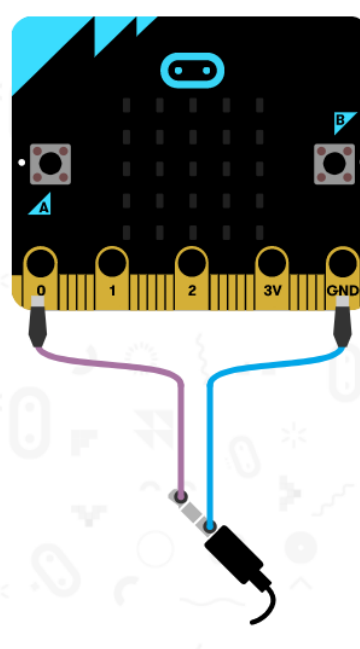
Add the following code to reset the Sorting Hat when either button A or button B is pressed:



來撥些音樂吧 – 內建聲音輸出

安全 | <https://makecode.microbit.org/#>

micro:bit 專案 分享 積木 {} JavaScript



搜尋...

- 基本
- 輸入
- 音效
- 燈光
- 廣播
- 迴圈
- 邏輯
- 變數
- 數學
- 進階

當 A 鍵被按下

- 顯示 指示燈

當 B 鍵被按下

- 演奏 音階 中 C 1 拍

下載 compass

搜尋...

- 基本
- 輸入
- 音效
- 燈光
- 廣播
- 迴圈
- 邏輯
- 變數
- 數學
- 進階

演奏 音階 中 C 1 拍

演奏 音階 中 C

演奏 休息 (ms) 1 拍

播放旋律 dadadum 重複 一次

當音效的 音階播放

中 C

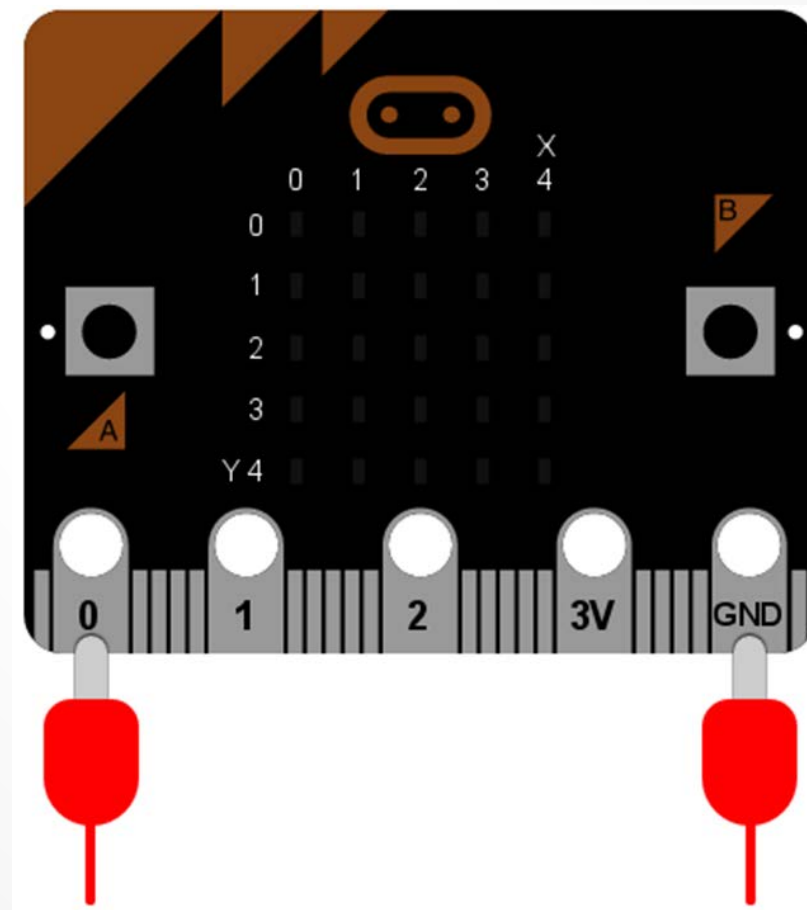
1 拍

演奏速度 (bpm)

演奏速度改變 (bpm) 20

演奏速度設為 (bpm) 120

外接揚聲器方式



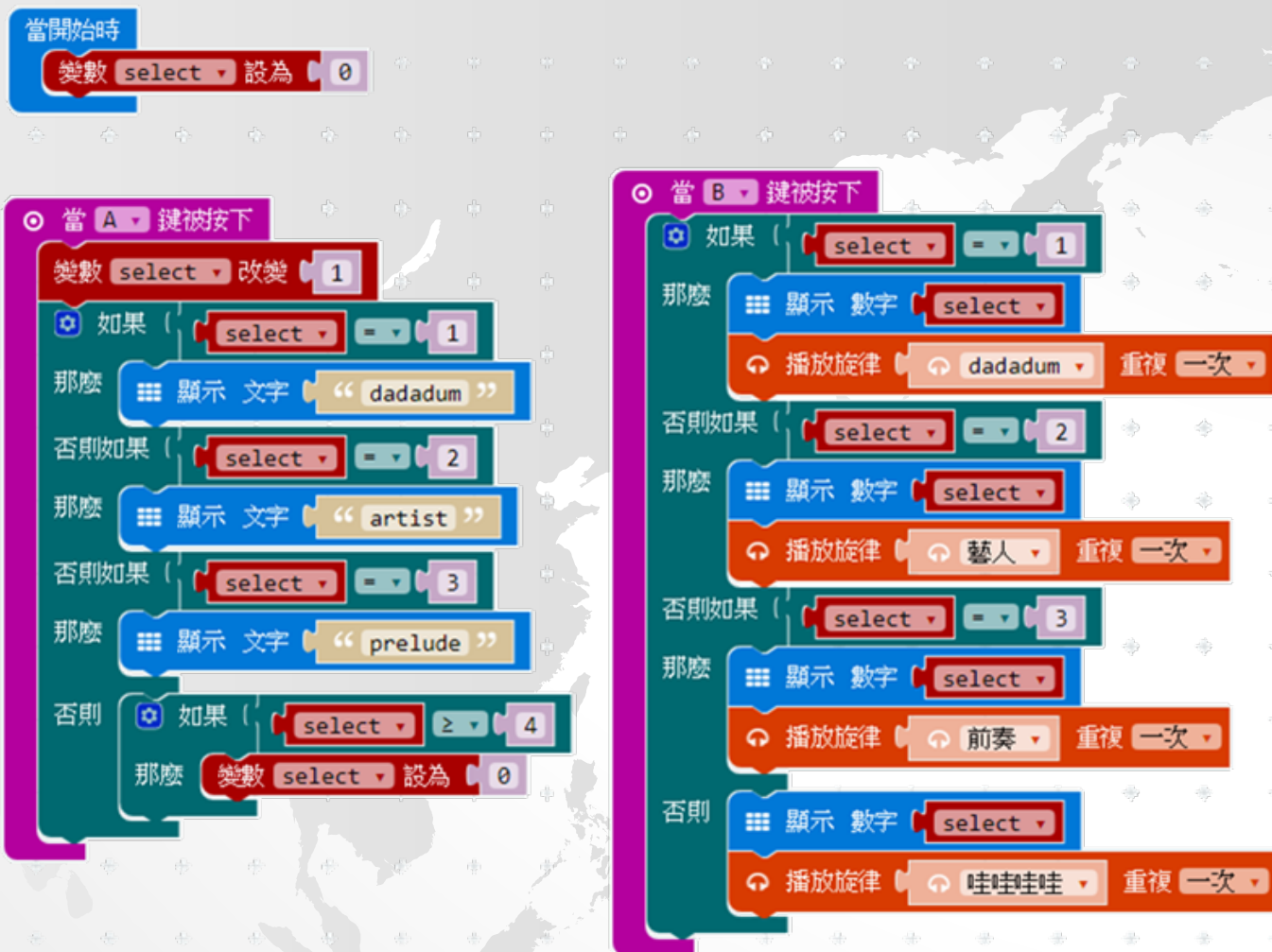
用按鍵選曲撥放

增加按鍵控制做選曲：

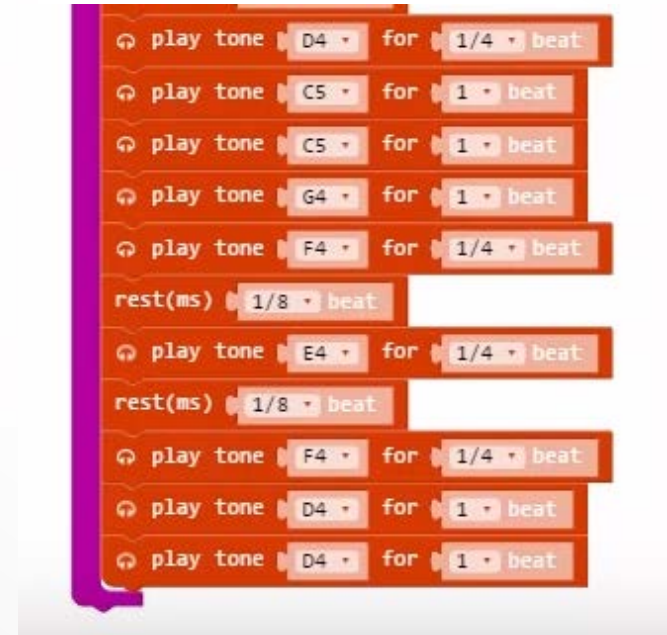
按鍵A被按下時，顯示曲目；

按鍵B被按下時，播放聲音。

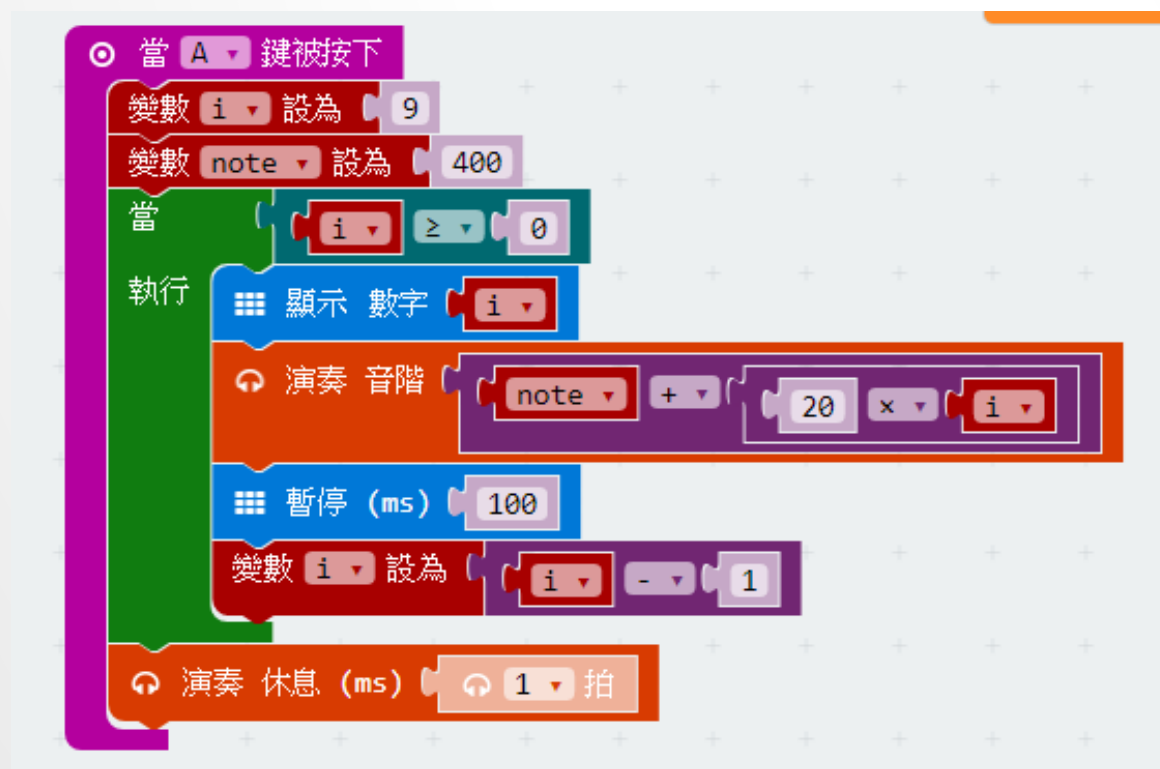
→ 任選3首曲子，
按A顯示曲名，
按B播放。



猜猜是甚麼曲子？

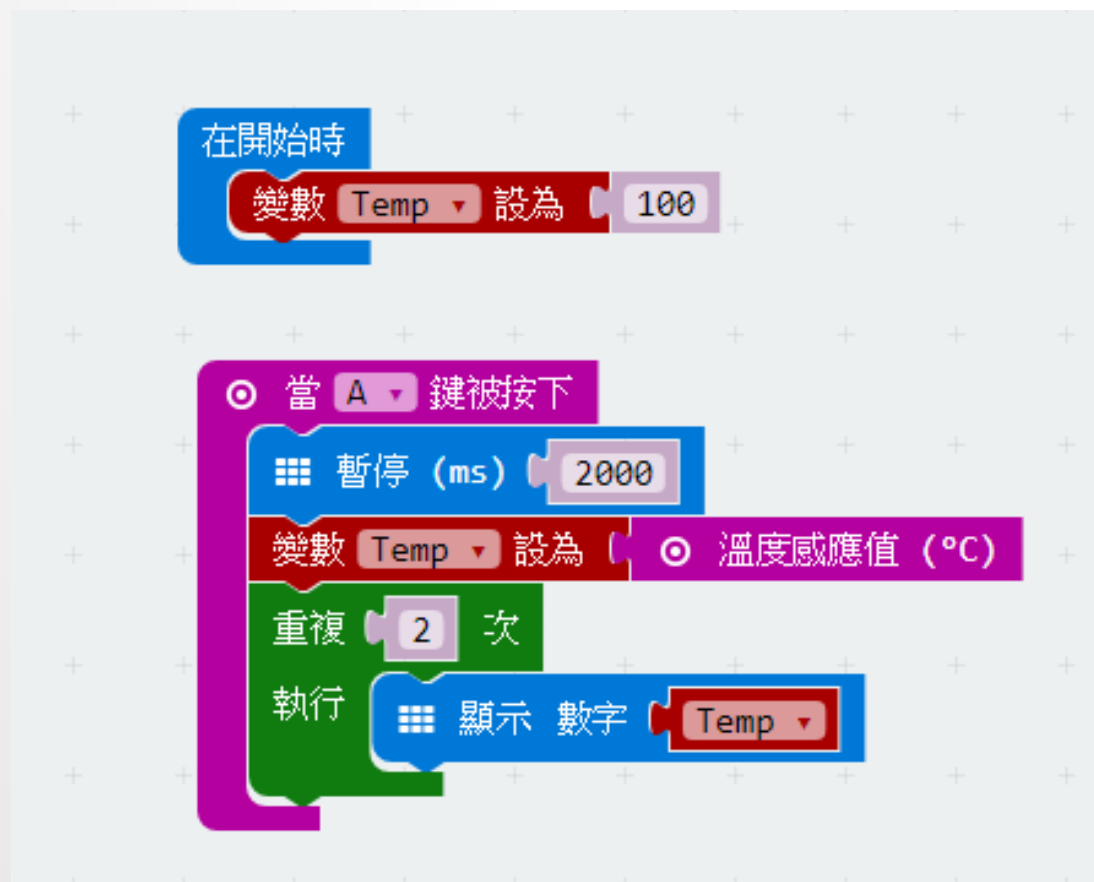


EX: 倒數配上降低音調



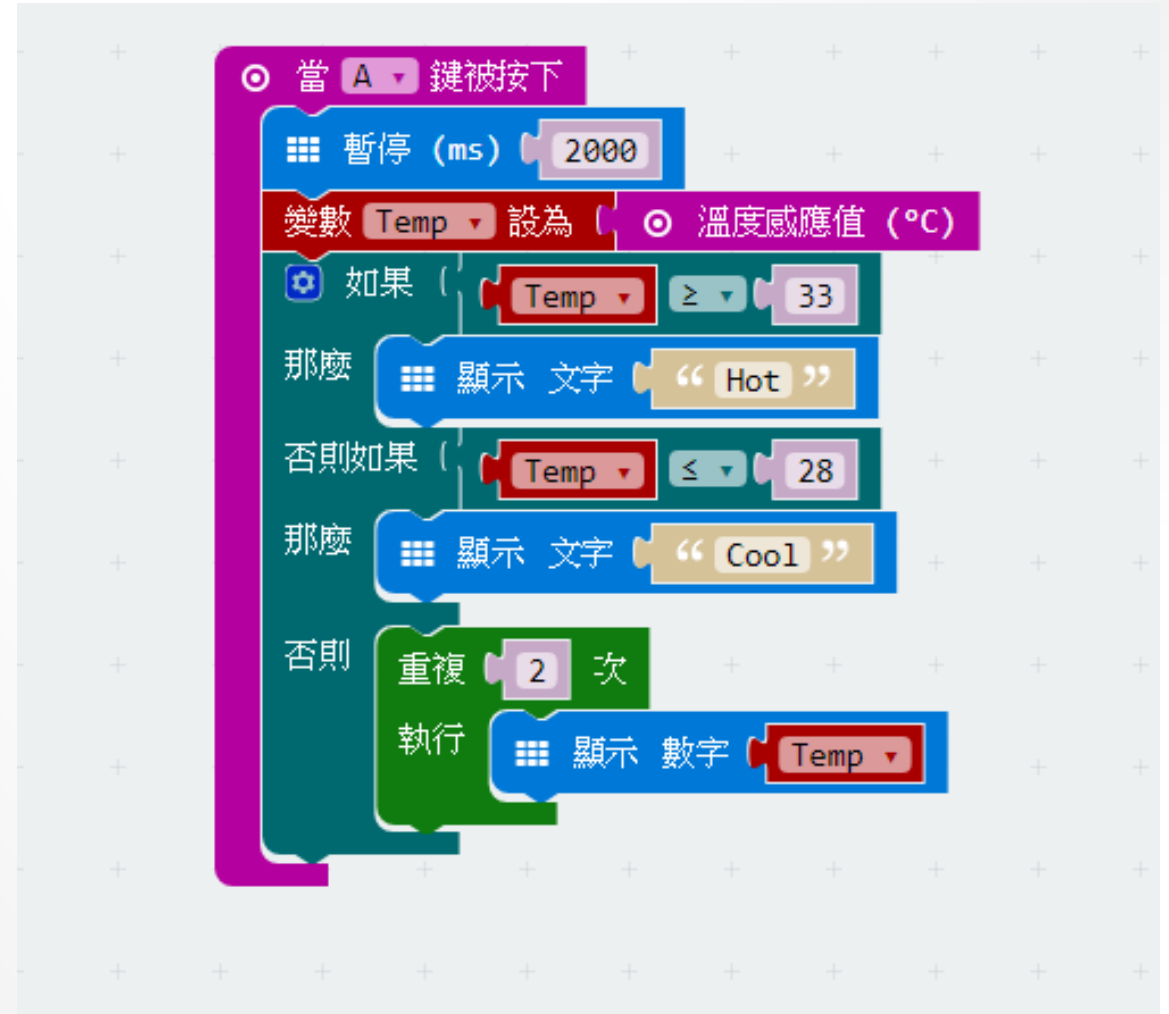
溫度感測器

- 當A鍵按下時讀取溫度值，並顯示出來



EX: 溫度感測判斷

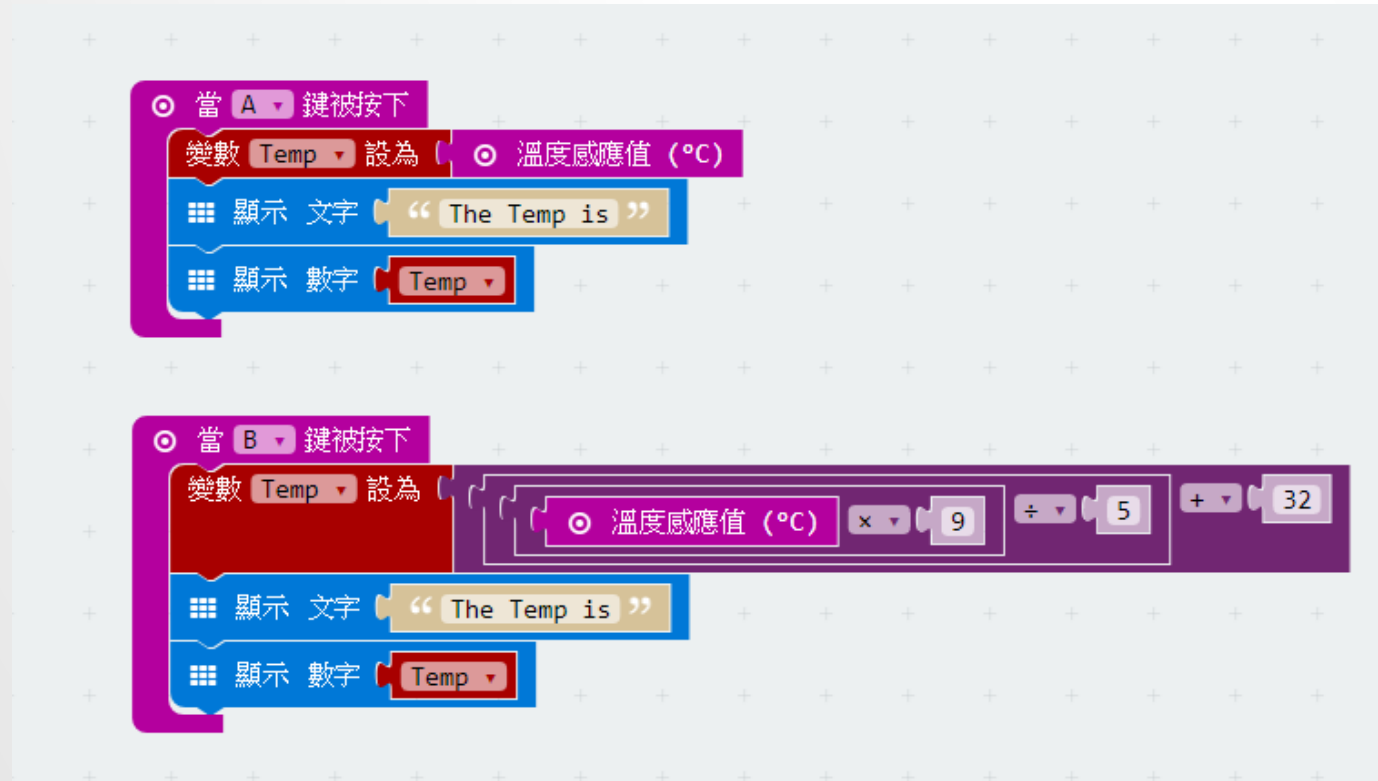
- 感測溫度約大於體溫(33°C)時顯示 "Hot"
- 感測溫度約小於室溫(28°C)時顯示 "Cool"
- 感測溫度在28°C~ 33°C中間顯示溫度



如何顯示華氏溫度？

- 華氏 = 攝氏 $\times(9/5)+32$
- 攝氏 = (華氏-32) $\times 5/9$

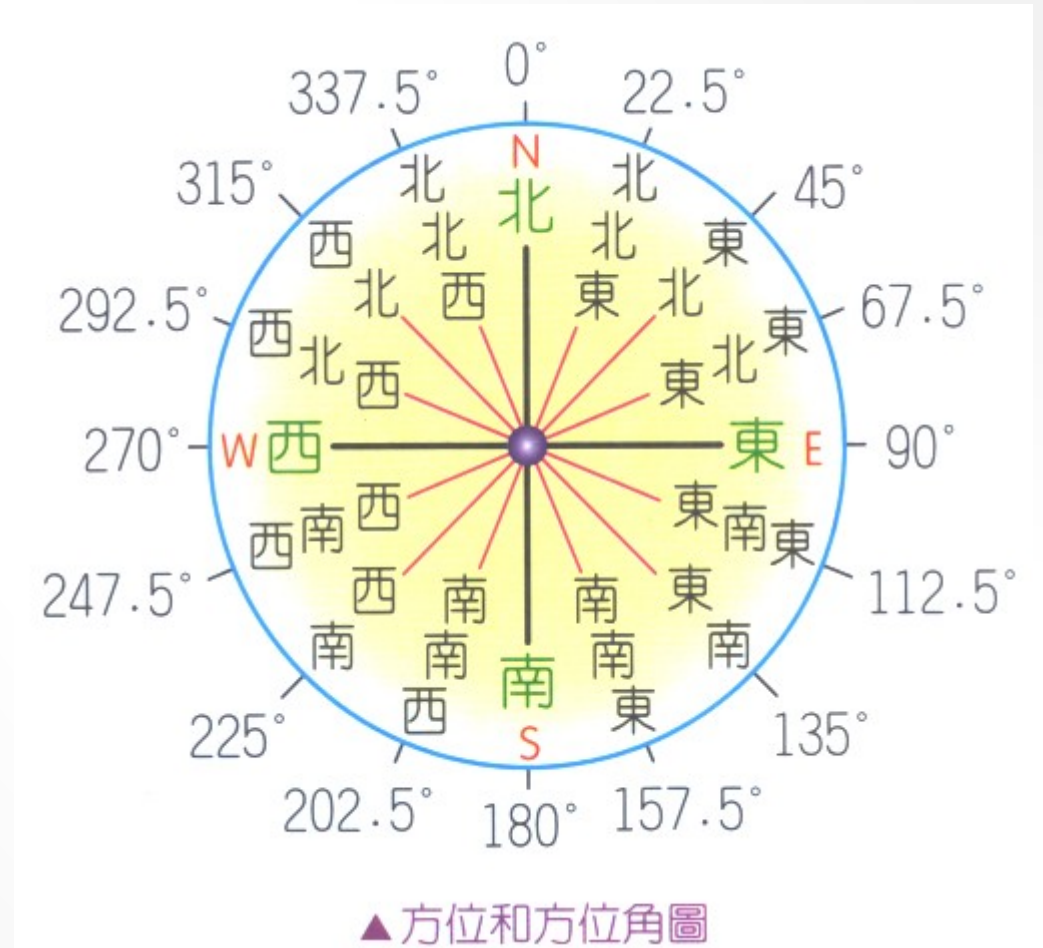
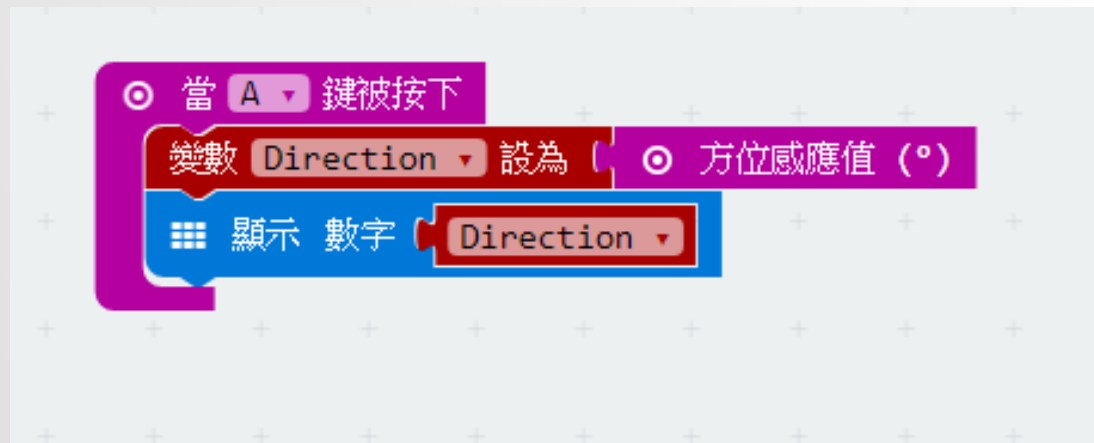
按 A 顯示攝氏，按 B 顯示華氏



留意計算特性!

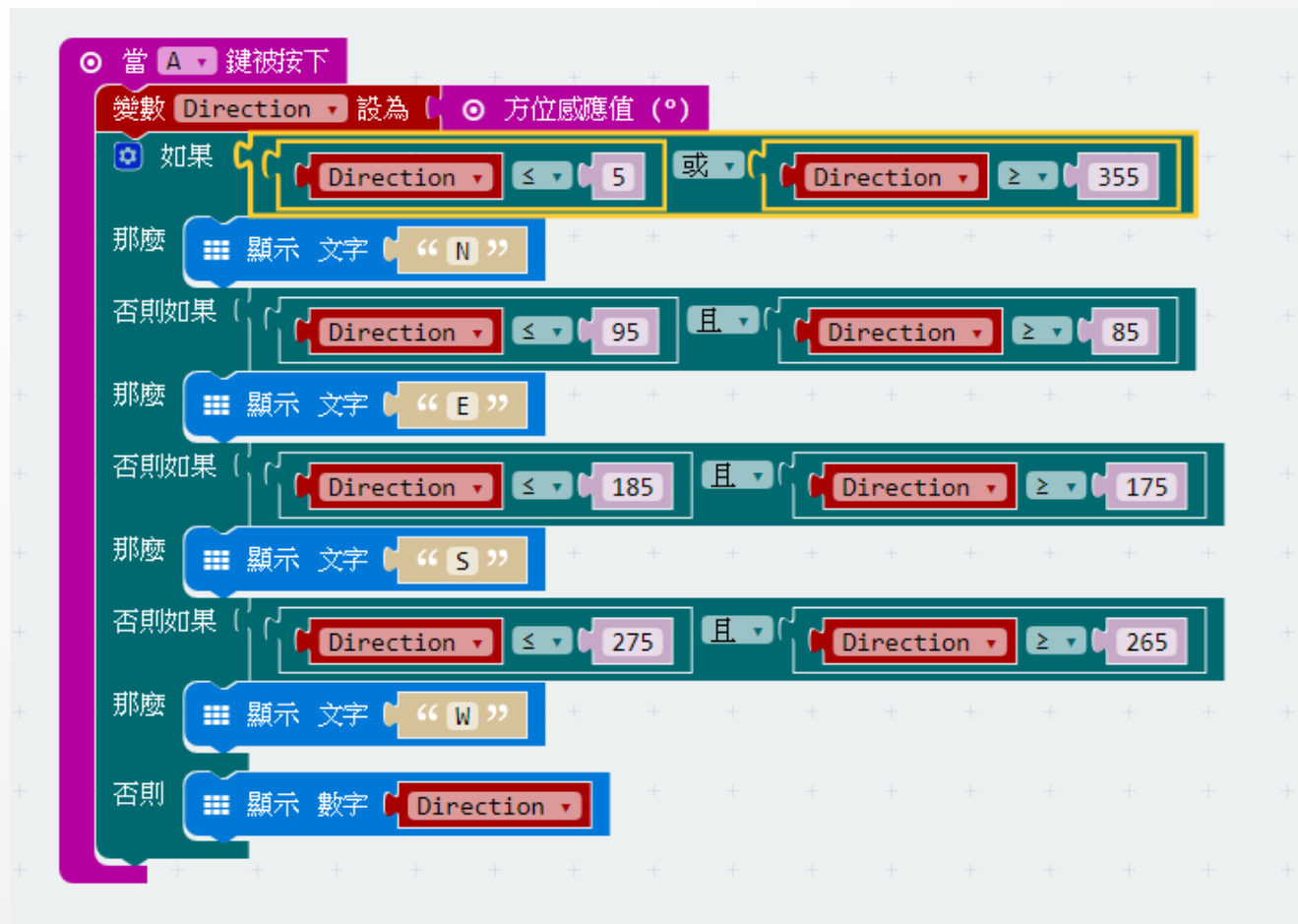
方向感測器

- 按下A鍵時顯示方向值
- 注意校正方式，誤差過大須重新flash code再校正。
- $0^{\circ} \sim 360^{\circ}$ 方位表示法。



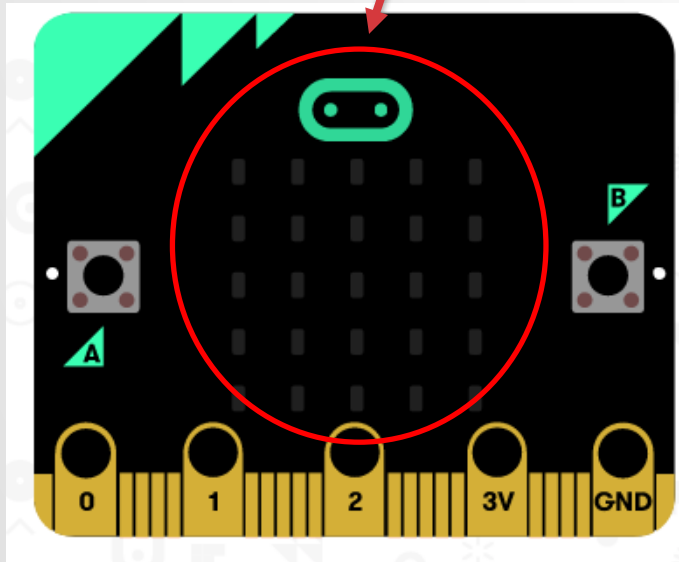
EX: 顯示(S、E、W、N)指南針

- 0度為正北，180度為正南，
- 90度為正東，270度為正西

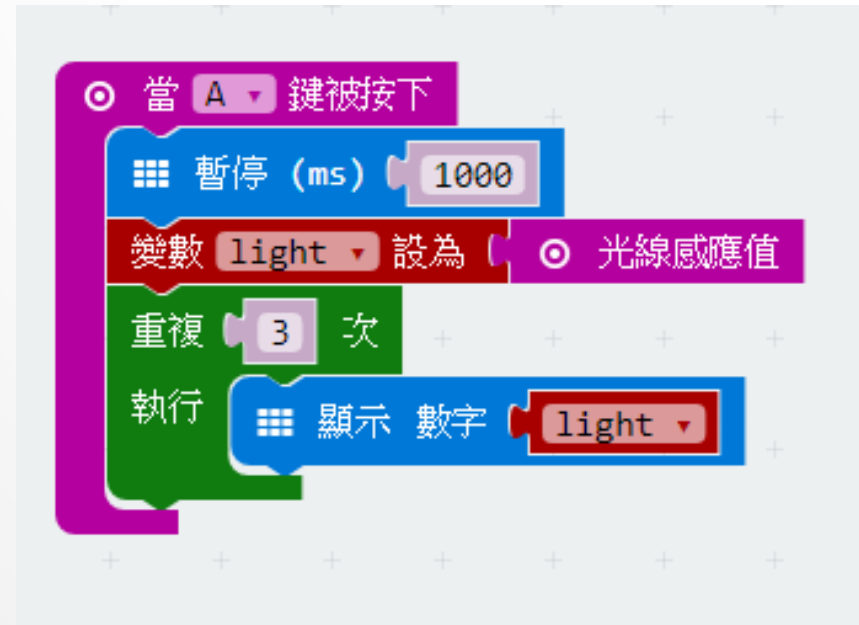


光線(亮度)感測器

感應區



當A鍵按下時顯示偵測到的亮度值，並暫停1秒重複顯示三次。



MICRO:BIT – NETWORK(I)

可以將設備連接在一起以發送和接收消息。這被稱為網絡。把網絡相聯結的網絡被稱為互聯網。互聯網又是所有網絡的互聯網。

要把網絡反映在下面描述的程序中很難。然而，這個項目的美麗之處在於它包含了您需要了解的網絡編程的所有常見特性。它也非常簡單和有趣。

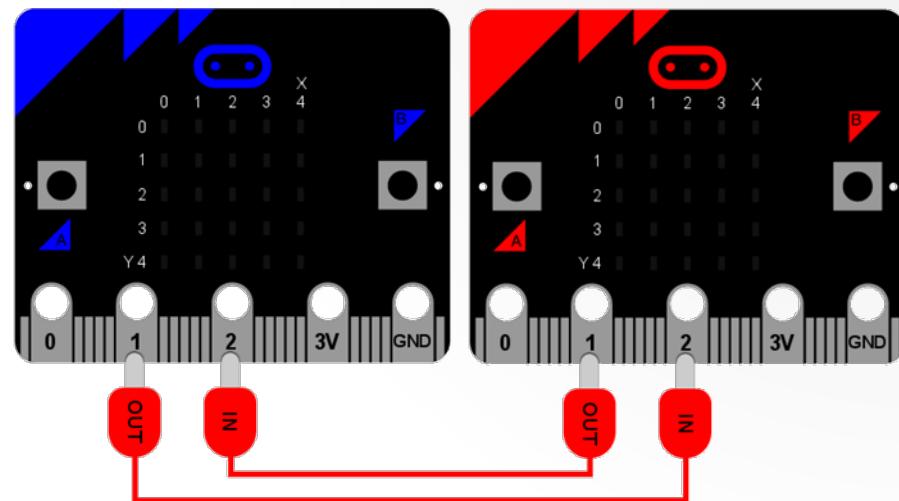
但首先，讓我們設置一個場景... “聯結”

想像一下有個一系列層次的網絡。最底層是最基本的通訊部分：需要某種方式將信號從一個設備到另一個設備。有時這是通過無線電連接完成的，但在這個例子中，我們只需要使用兩根線(如圖所示)。

網絡堆疊(the *network stack*)中的下一層是信號。這通常取決於連接的特性。在我們的例子中，它是簡單的數字開啟和關閉信號通過IO引腳發送下來的電線。

我們可以這樣使用IO引腳：

```
pin1.write_digital(1)      # switch the signal on
pin1.write_digital(0)      # switch the signal off
input = pin2.read_digital() # read the value of the signal (either 1 or 0)
```



MICRO:BIT – NETWORK(II)

通訊協定(Protocol)：我們定義和使用通過計算機網絡傳達消息的協議。計算機需要雙邊同意如何發送和接收消息。也許最著名的協議是由萬聯網(World Wide Web)使用的超文本傳輸協議 (HTTP)。

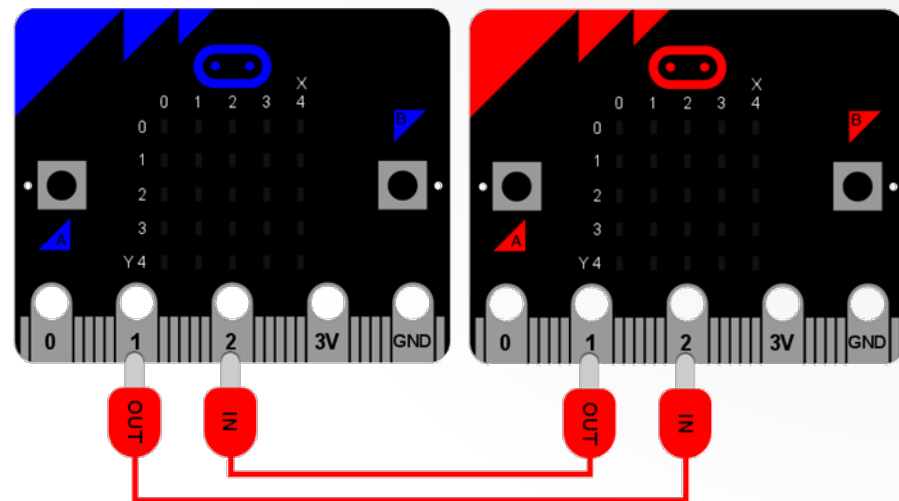
發送消息 (最早期的計算機) 的另一個著名協議是莫爾斯電碼(Morse code)。它定義如何通過長或短持續時間的開/關信號發送基於字符的消息。通常這樣的信號是嗶嗶聲。長持續時間為破折號(-)，而短持續時間是點(.)。通過組合虛線和點莫爾斯定義了發送字符的方式。例如，這裡是標準莫爾斯字母的定義表：

.----	1	..---	2	...--	3	-	4		5	-....	6	--...	7	-----	8	-----	9	-----	0
.-	A	-...	B	-.-.	C	-..	D	.	E	..-	F	--.	G		H	..	I	---	J
-.-	K	...-	L	--	M	-.	N	---	O	...-	P	--.-	Q	...-	R	...	S	-	T
...-	U	-	V	...-	W	-...-	X	-...-	Y	-...-	Z								

這就是一種非常好的網絡堆疊，但您還需要一種與之交互的方式 - 某種形式的應用程序來發送和接收消息。雖然HTTP是有趣的，但大多數人不了解它，並讓他們的網絡瀏覽器處理它 - 萬聯網的底層網絡堆疊是隱藏的 (也應該是要這樣)。

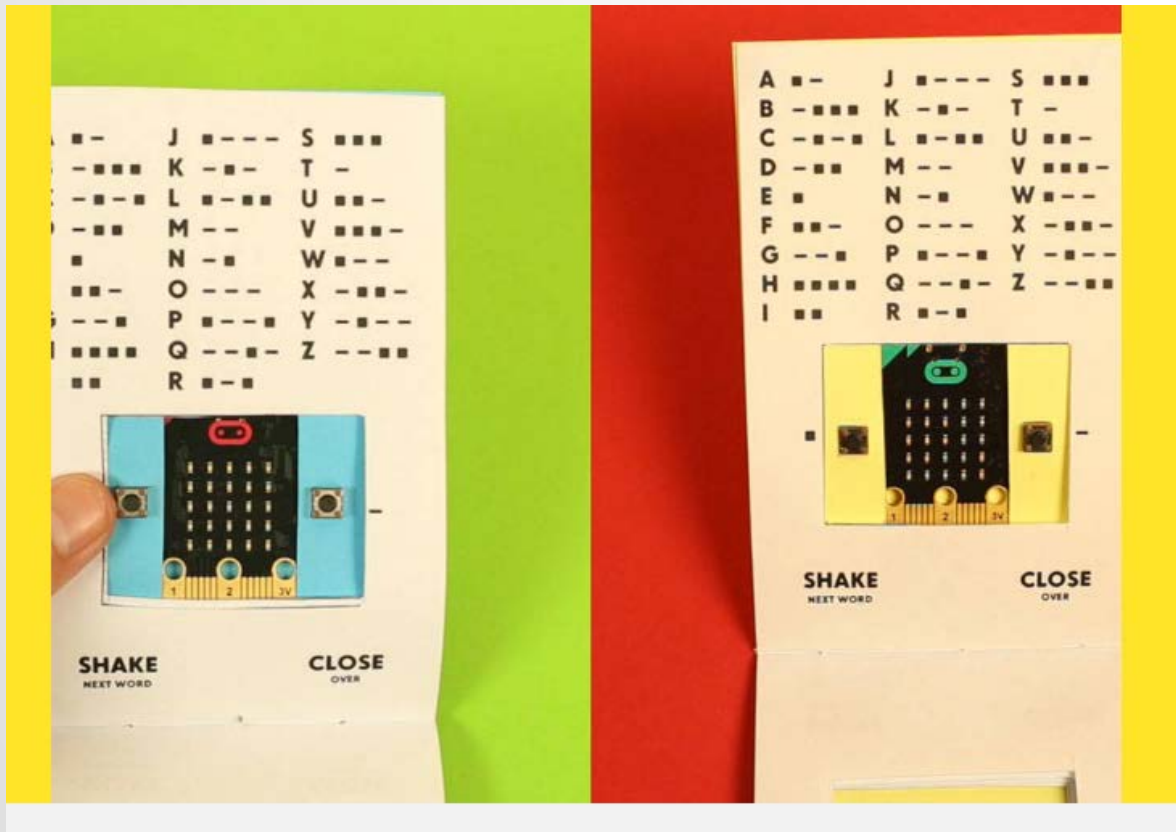
那麼從用戶的角度來看，我們應該為micro:bit寫什麼樣的應用程序？應該如何工作呢？

顯然，要發送消息，您可以輸入點和破折號 (我們可以使用按鈕A)。如果我們想看到我們發送或剛剛收到的消息，我們可以觸發它在顯示器上滾動 (我們可以使用按鈕B)。而這是莫爾斯電碼，如果附上一個揚聲器，我們應該能夠在用戶輸入消息時發出嗶嗶聲作為一種聽覺反饋。



摩斯密碼電話

MICRO MORSE PHONE



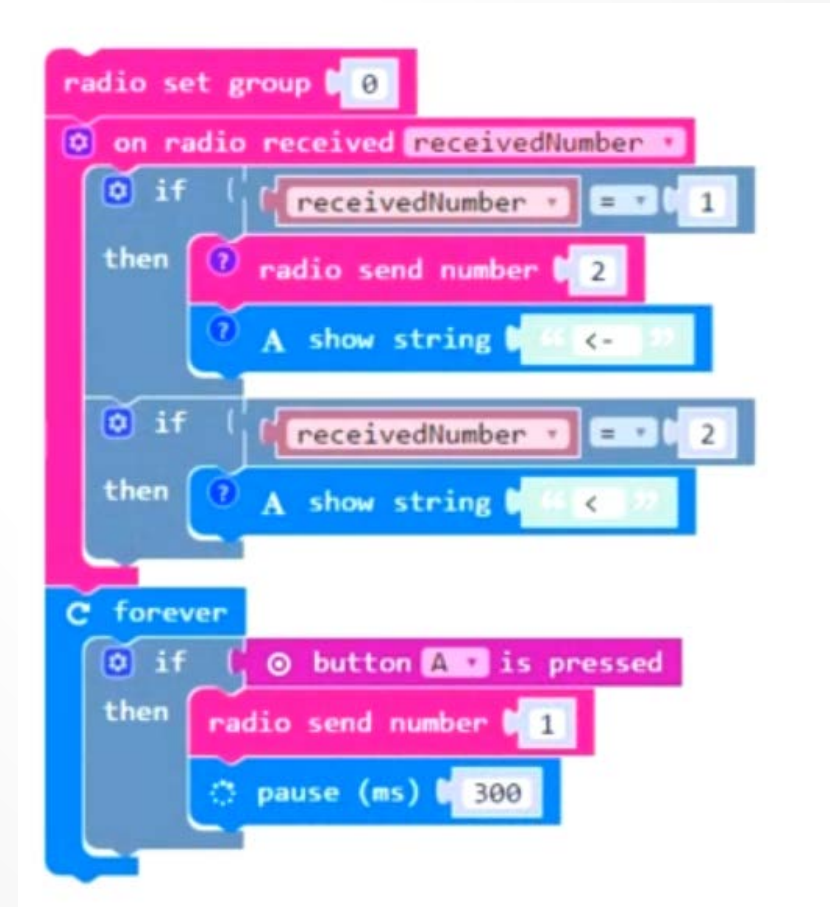
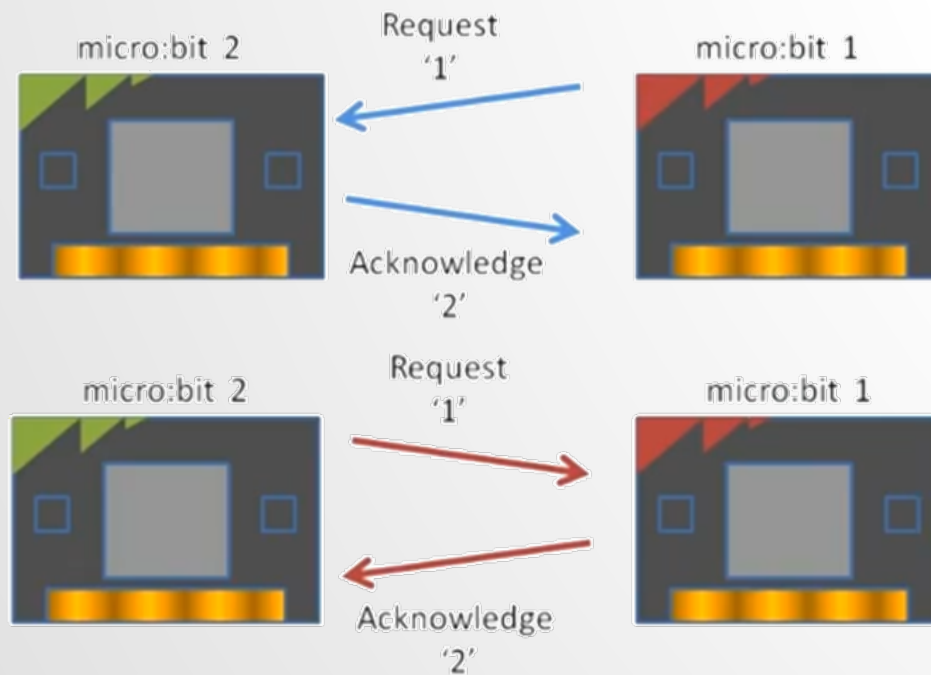
Flash the code

Copy this code into the Mu editor and flash it onto both micro:bits.

```
import radio
from microbit import *
dash = Image("00000:00000:99999:00000:00000")
dot = Image("00000:09990:09990:09990:00000")
word = Image("00900:00090:99999:00090:00900")
over = Image("00900:00090:99999:00090:00900")
radio.on() #turn on radio
while True: #loop forever
    incoming = radio.receive() #check radio signals
    gesture = accelerometer.get_gestures()
    if button_a.is_pressed():
        display.show(dot)
        radio.send(str("dot"))
    elif button_b.is_pressed():
        display.show(dash)
        radio.send(str("dash"))
    elif gesture == "shake":
        display.show(word)
        radio.send(str("word"))
    elif gesture == "face down":
        display.show(over)
        radio.send(str("over"))
    elif incoming == "dot":
        display.show(dot)
    elif incoming == "dash":
        display.show(dash)
    elif incoming == "over":
        display.show(over)
    elif incoming == "word":
        display.show(word)
    sleep(200)
    display.clear()
```

無線通訊範例 - 收到訊號傳送箭頭

- 按A鍵發送訊息給對方
- 收到訊號時顯示箭頭



無線通訊範例—螢火蟲__微軟板

Download this program on as many micro:bit as you can find and try it out in a dark room!

(We've added a [radio set group](#) block to specify which group the firefly will communicate on).



無線通訊範例—改良螢火蟲

若沒收到別人的呼叫時閃方形，
若是收到別人的呼叫時改成閃
心型3次，並也呼叫別人。

被動版

Not work!!!!



無線通訊範例—改良螢火蟲

若沒收到別人的呼叫時閃方形，
若是收到別人的呼叫時改成閃
心型3次，並也呼叫別人。

主動版

在開始時



EX: 投票機

- 在這個項目中分成兩部分，選民與統計者；選民使用micro:bit按鈕投票“是”或“否”，通過無線電發送到統計者的micro:bit。統計者為每個播放器分配一個LED，並根據投票將其打開或關閉。
- 假設選民的按鈕A為否，B為是，則選民程序的工作原理如下：
 - 當按下按鈕A時，通過無線電發送數字0，屏幕上顯示X符號。
 - 當按下按鈕B時，通過無線電發送號碼255，屏幕上顯示Y符號。
- 為了追蹤投票，我們設定無線電傳送設備序列號。（例如：選擇4作為用於通信的群組）



EX: 投票機 – 改良版

- 教師端：

當收到學生的回答，反覆依序顯示：

A -> 選A人數

B -> 選B人數

C -> 選C人數

D -> 選D人數

? -> 答案不確定人數

當翻面送出無線電訊號學生才可作答，並清除記錄。

- 學生端

一開始顯示S代表學生端，無法作答。

當抽到老師端(翻面)通知，左右搖晃板子才會顯示 “A、B、C、D、?” 供選擇，按下A或B鍵送出答案，然後顯示“X” 便無法再作答，除非老師解除鎖定(翻面)。

在開始時

廣播群組設為 99
顯示 文字 "S"
變數 START 設為 -1

當收到廣播 receivedNumber

如果 receivedNumber = 0
那麼 變數 START 設為 0

學生端

重複無限次

如果 START = 0
那麼 變數 xx 設為 加速度感應值 (mg) x
如果 xx < -600
那麼 顯示 文字 "A"
如果 A 鍵被按下? 或 B 鍵被按下?
那麼 廣播發送數字 1
變數 START 設為 -1
顯示 圖示
否則如果 xx < -200
那麼 顯示 文字 "B"
如果 A 鍵被按下? 或 B 鍵被按下?
那麼 廣播發送數字 2
變數 START 設為 -1
顯示 圖示
否則如果 xx < 200

否則如果 xx < 200

那麼

顯示 文字 "C"
如果 A 鍵被按下? 或 B 鍵被按下?
那麼 廣播發送數字 3
變數 START 設為 -1
顯示 圖示

否則如果 xx < 600

那麼

顯示 文字 "D"
如果 A 鍵被按下? 或 B 鍵被按下?
那麼 廣播發送數字 4
變數 START 設為 -1
顯示 圖示

否則

顯示 文字 "?"
如果 A 鍵被按下? 或 B 鍵被按下?
那麼 廣播發送數字 5
變數 START 設為 -1
顯示 圖示

老師端

在開始時

呼叫函式 `clear_score`

廣播群組設為 99

當 螢幕朝下

呼叫函式 `clear_score`

廣播發送數字 0

當收到廣播 `receivedNumber`

如果 (`receivedNumber` = 1)

那麼 變數 `A` 改變 1

否則如果 (`receivedNumber` = 2)

那麼 變數 `B` 改變 1

否則如果 (`receivedNumber` = 3)

那麼 變數 `C` 改變 1

否則如果 (`receivedNumber` = 4)

那麼 變數 `D` 改變 1

否則 變數 `E` 改變 1

重複無限次

顯示 文字 "A"

暫停 (ms) 500

清除 屏幕

計數 `H` 從 0 到 `A` - 1

執行 點亮 x `H` ÷ 5 y `H` ÷ 5 的餘數

暫停 (ms) 3000

顯示 文字 "B"

暫停 (ms) 500

清除 屏幕

計數 `H` 從 0 到 `B` - 1

執行 點亮 x `H` ÷ 5 y `H` ÷ 5 的餘數

暫停 (ms) 3000

顯示 文字 "C"

函式 `clear_score`

清除 屏幕

變數 `A` 設為 0

變數 `B` 設為 0

變數 `C` 設為 0

變數 `D` 設為 0

變數 `E` 設為 0

暫停 (ms) 3000

顯示 文字 "C"

暫停 (ms) 500

清除 屏幕

計數 `H` 從 0 到 `C` - 1

執行 點亮 x `H` ÷ 5 y `H` ÷ 5 的餘數

暫停 (ms) 3000

顯示 文字 "D"

暫停 (ms) 500

清除 屏幕

計數 `H` 從 0 到 `D` - 1

執行 點亮 x `H` ÷ 5 y `H` ÷ 5 的餘數

暫停 (ms) 3000

顯示 文字 "?"

暫停 (ms) 500

清除 屏幕

計數 `H` 從 0 到 `E` - 1

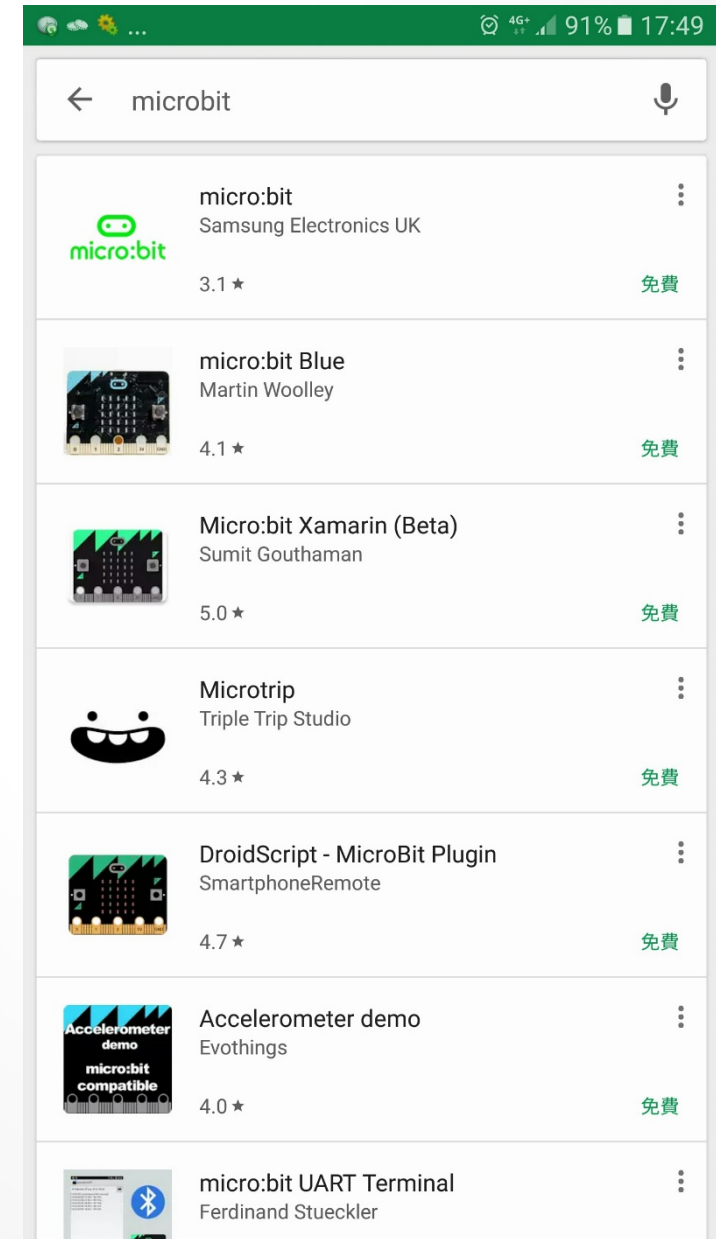
執行 點亮 x `H` ÷ 5 y `H` ÷ 5 的餘數

暫停 (ms) 3000

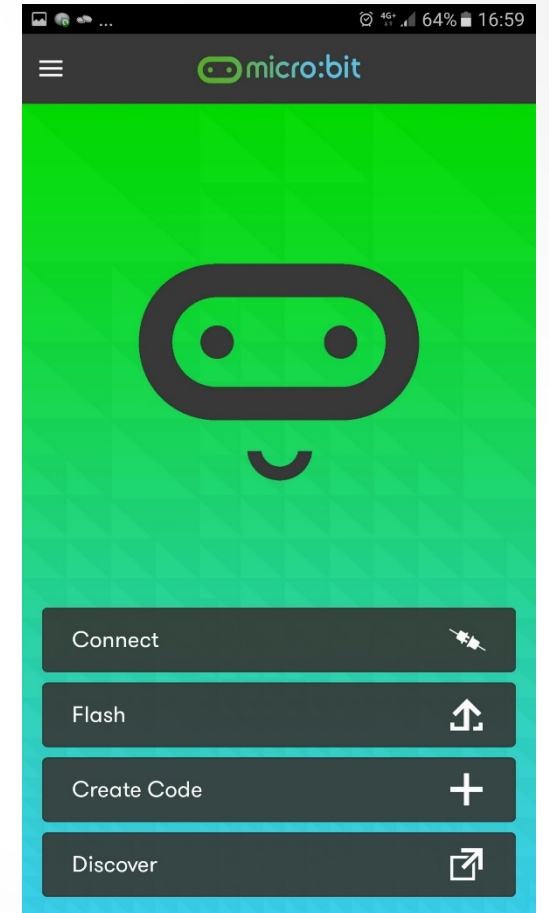
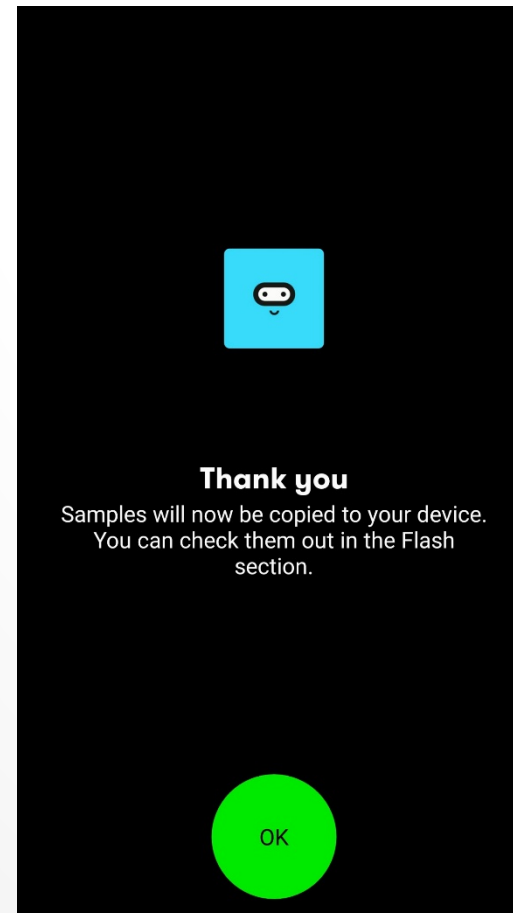
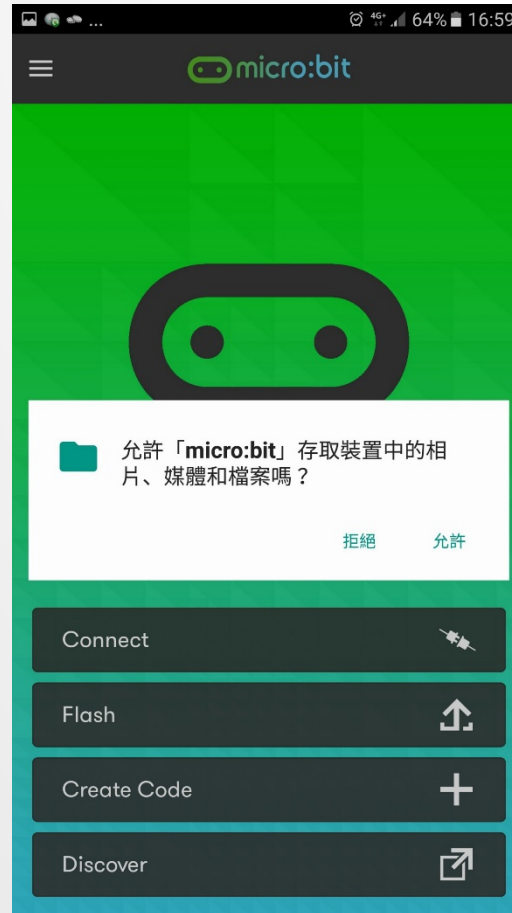
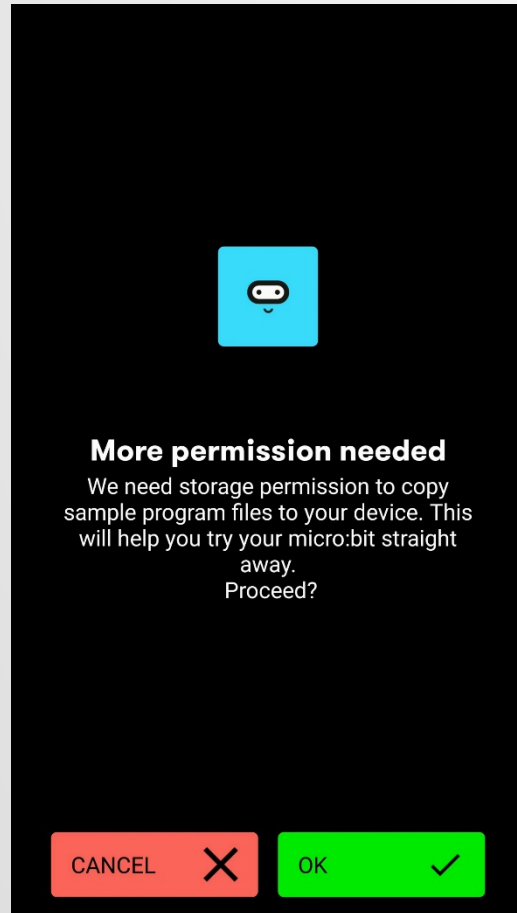
行動裝置安裝與連結

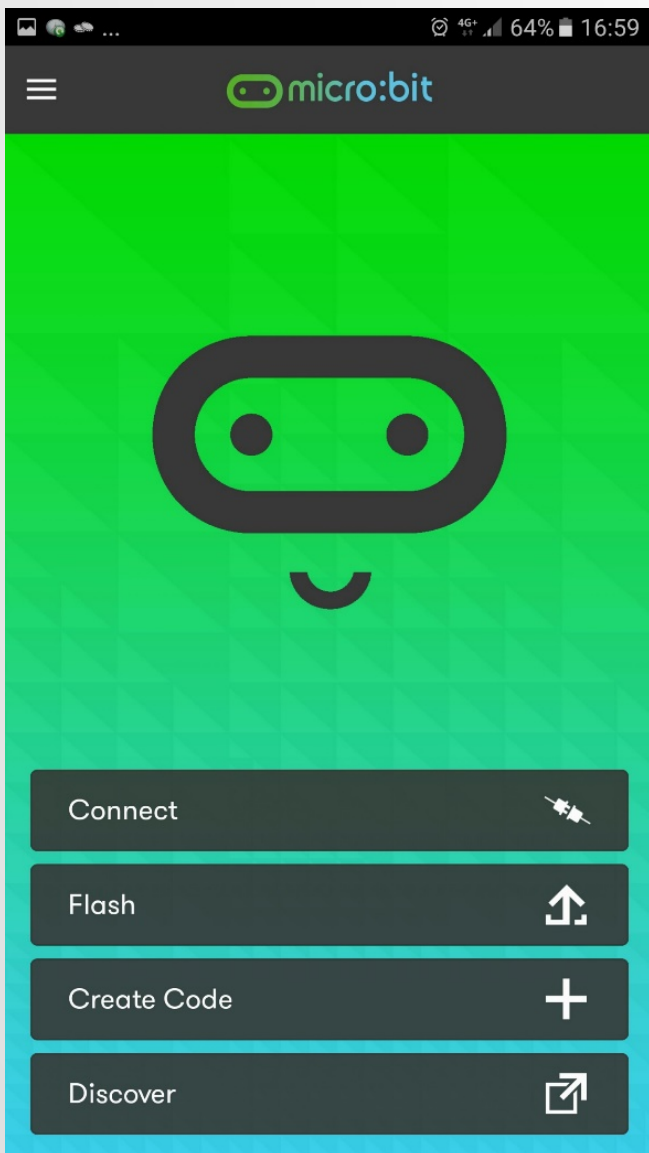
Google play 或是 APP store
搜尋: **microbit**

Google Play
App Store



安裝APP





- **Connection**

將你的micro:bit與 智慧型手機/平板設備 進行配對及連接

- **Flash**

就是從手機平板等設備上傳 程式碼(.HEX文件) 至 micro:bit的地方

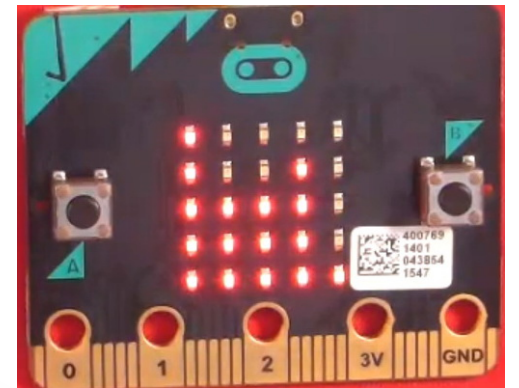
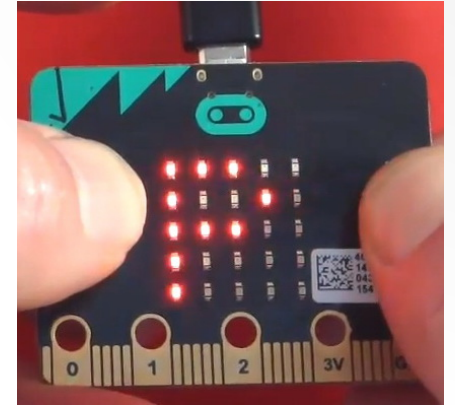
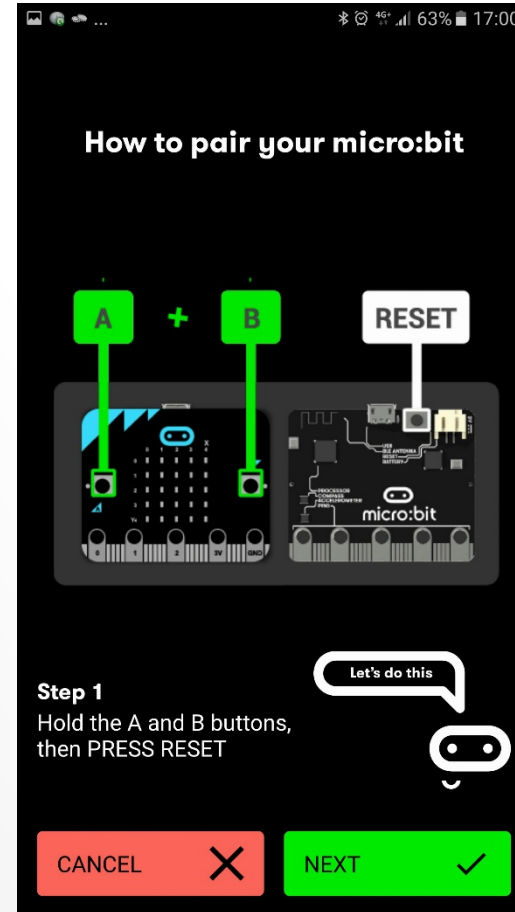
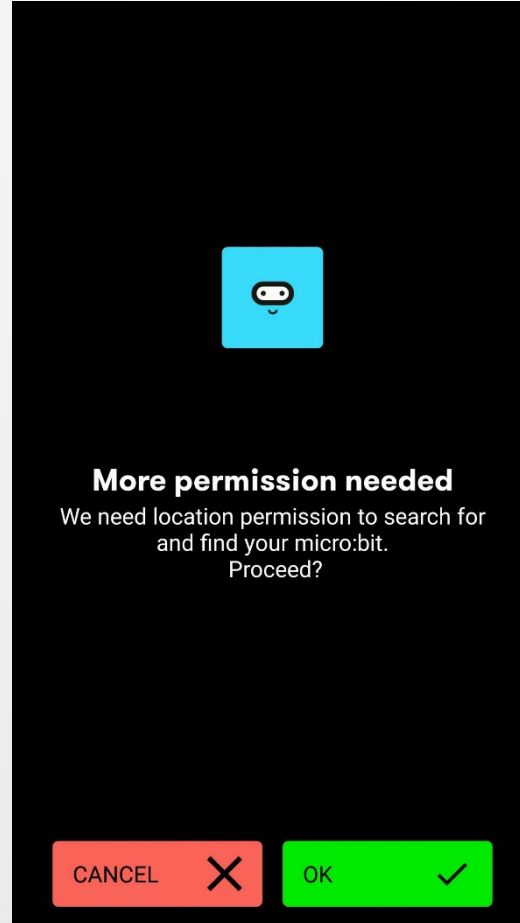
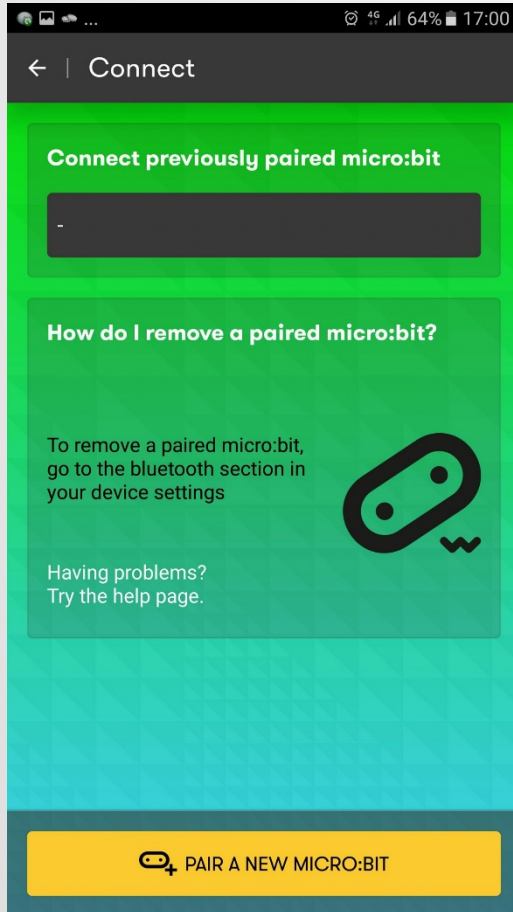
- **Create code**

引導你回到網站上，寫新的程式或追溯你過去寫過的程式碼

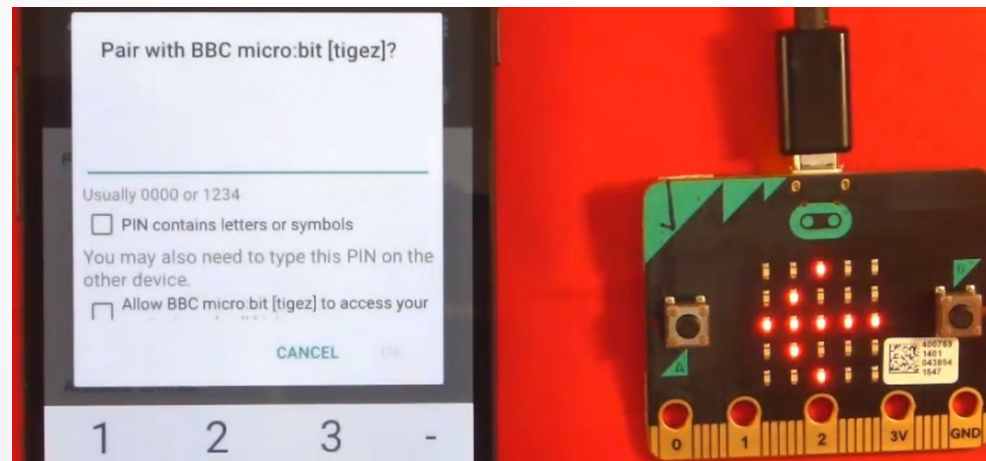
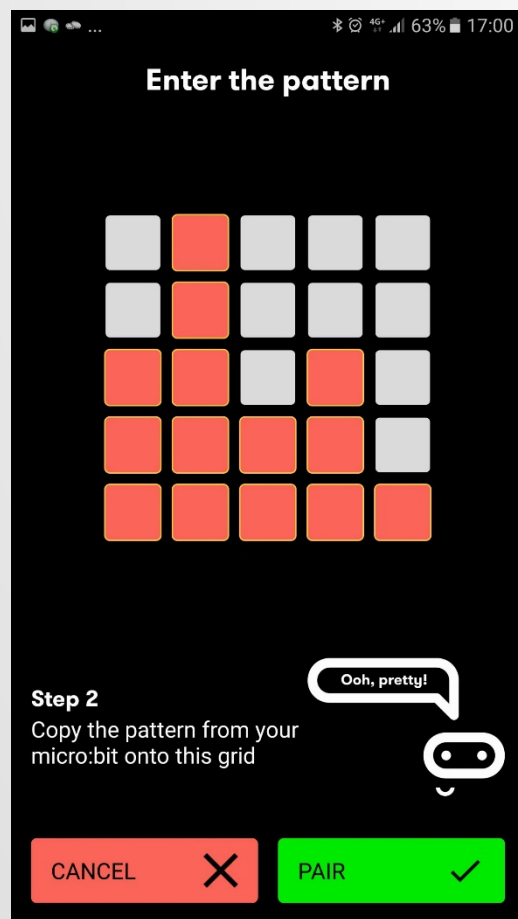
- **Discover**

產生創意的地方(導回官網的IDEA)

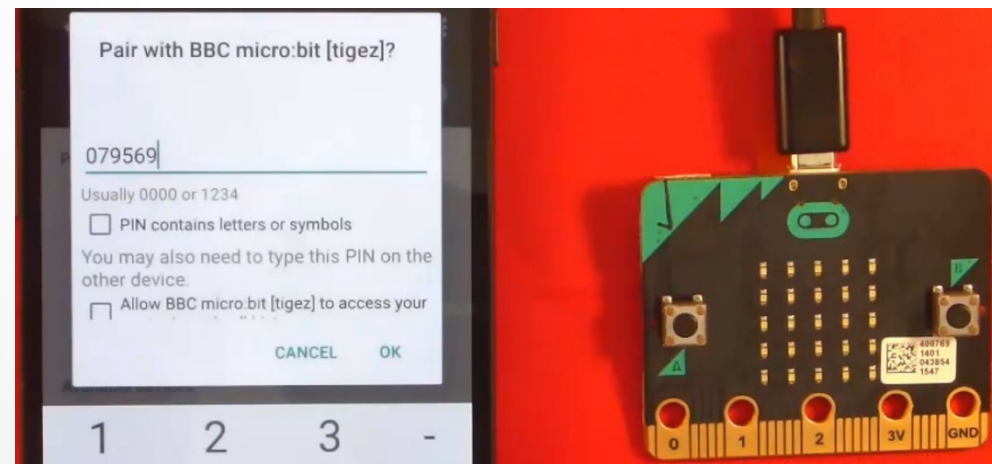
藍芽連結 (進入配對模式 PARING MODE)

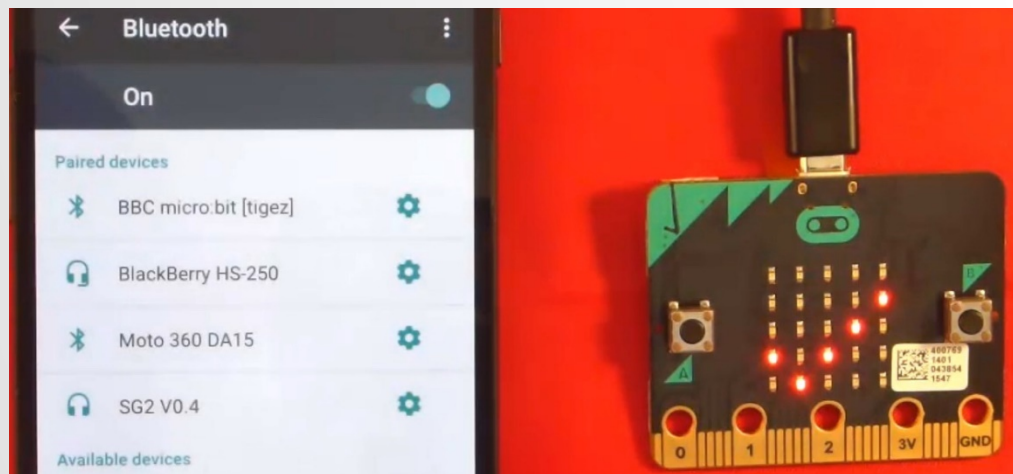


輸入配對pattern

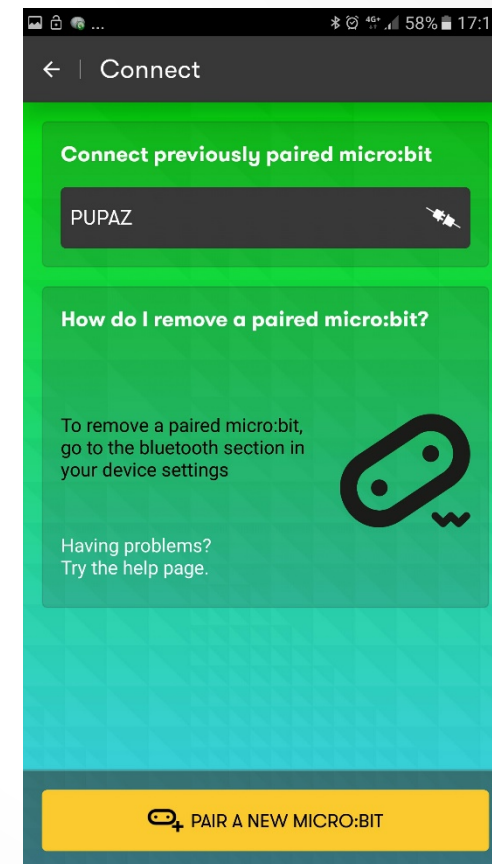
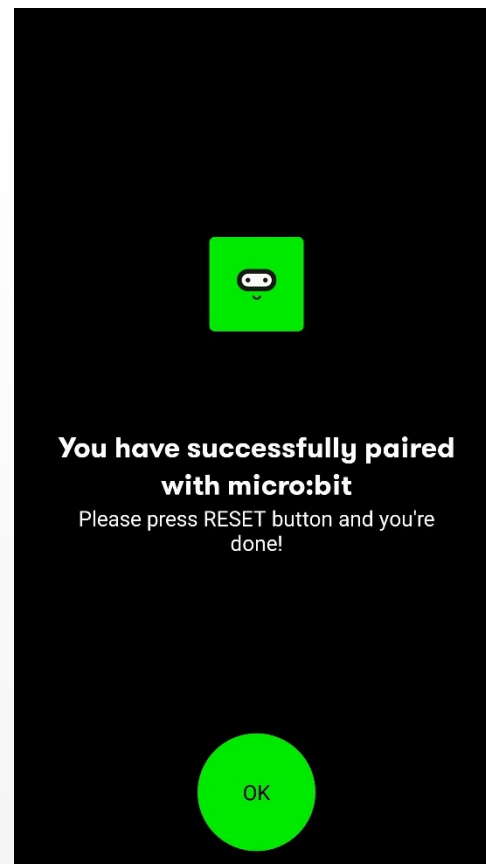


按A顯示pin code
(不一定需要這一步)

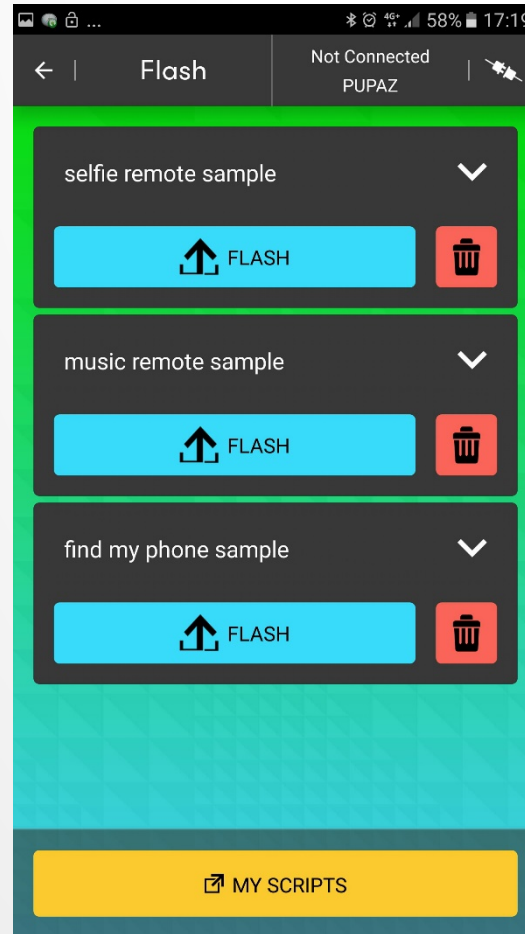
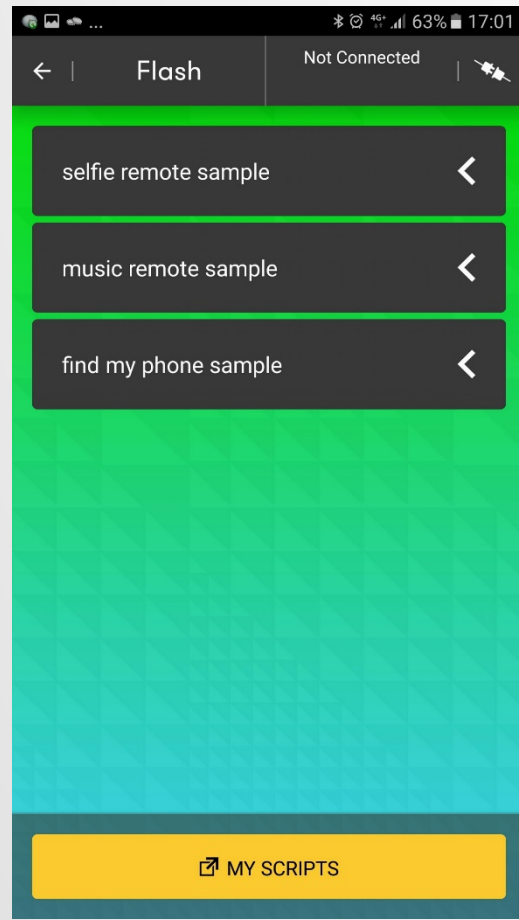




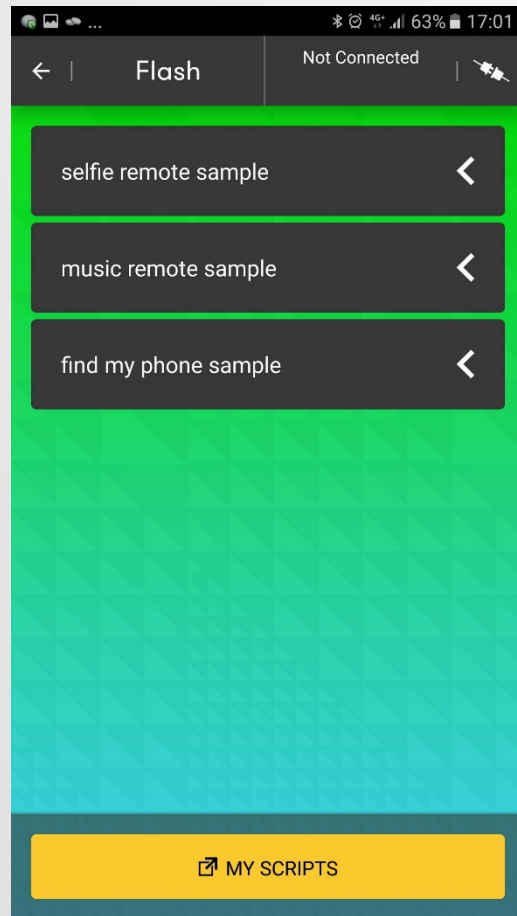
藍芽配對成功，在行動裝置的藍芽設備列表中可找到Micro:bit，可以重新進入配對模式進行連線。



燒錄程式碼



寫程式



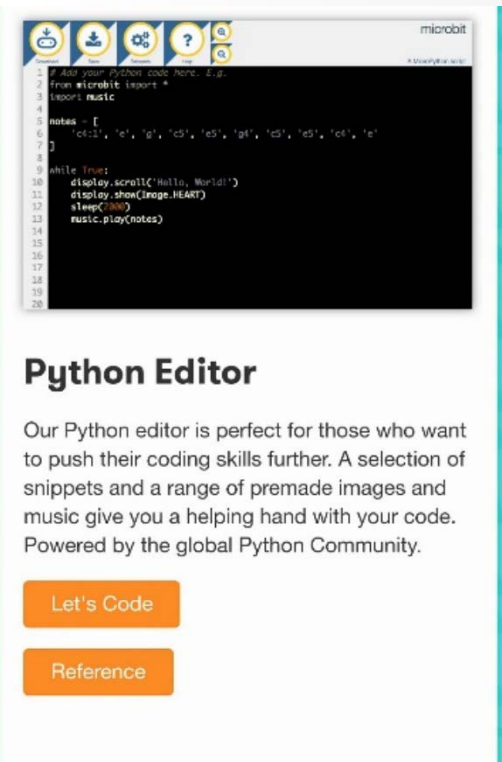
JavaScript Blocks Editor

Micro:bit's new JavaScript editor makes it easy to program your micro:bit in Blocks and JavaScript, along with great new features like peer-to-peer radio. Powered by MakeCode.

Let's Code

Reference

Lessons

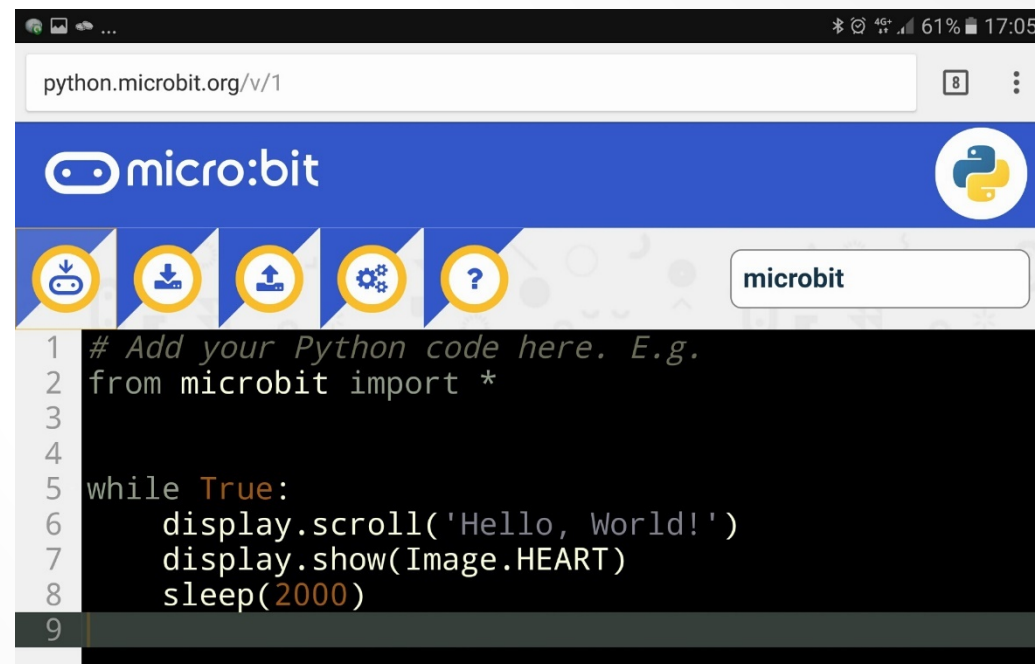


Python Editor

Our Python editor is perfect for those who want to push their coding skills further. A selection of snippets and a range of premade images and music give you a helping hand with your code. Powered by the global Python Community.

Let's Code

Reference



附錄：

線上課程

- [Microsoft 範例課程](#)(由microsoft 為microbit.org 設計的課程)
- [microbit 範例課程](#)(是由micro:bit.org 為microbit 準備的範例課程)
- [Javascript 課程系列介紹](#)
- [Scratch 在 micro:bit 應用介紹](#) (Mac OS)
- [AppInventor 課程系列](#)
- [Raspberry & Micro:bit](#)

台灣MICRO:BIT網路資源

- 臉書社團



- e-Shop



- 部落格



- <https://sites.google.com/site/wenyumaker2/02micro-bit-yan-jiu>
- <http://glglace.blogspot.tw/2017/10/1154-microbit-1.html>
- <http://blog.edumakerlab.org/>
- <http://glophy.com/index.php/2014-02-07-01-06-58/micro-python>