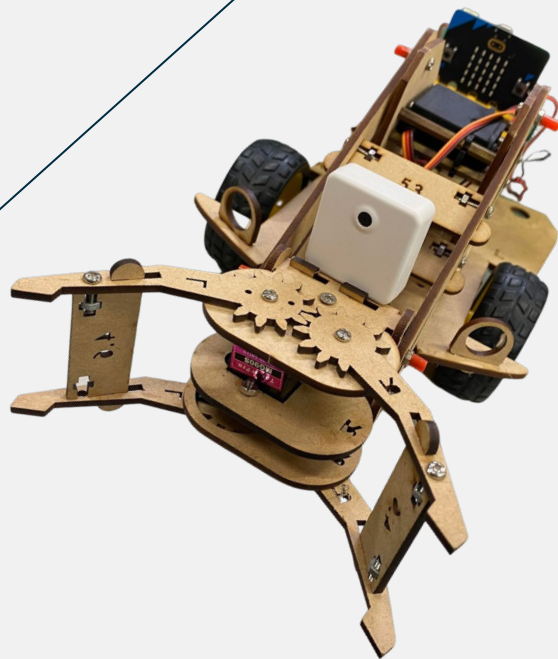


START! AI 智慧小車

自造大師團隊



目錄

CONTENT

01

micro:bit介紹

02

智慧小車組裝

03

程式環境設定

04

micro:bit連接

05

TT直流馬達

06

紅外線感測操作

07

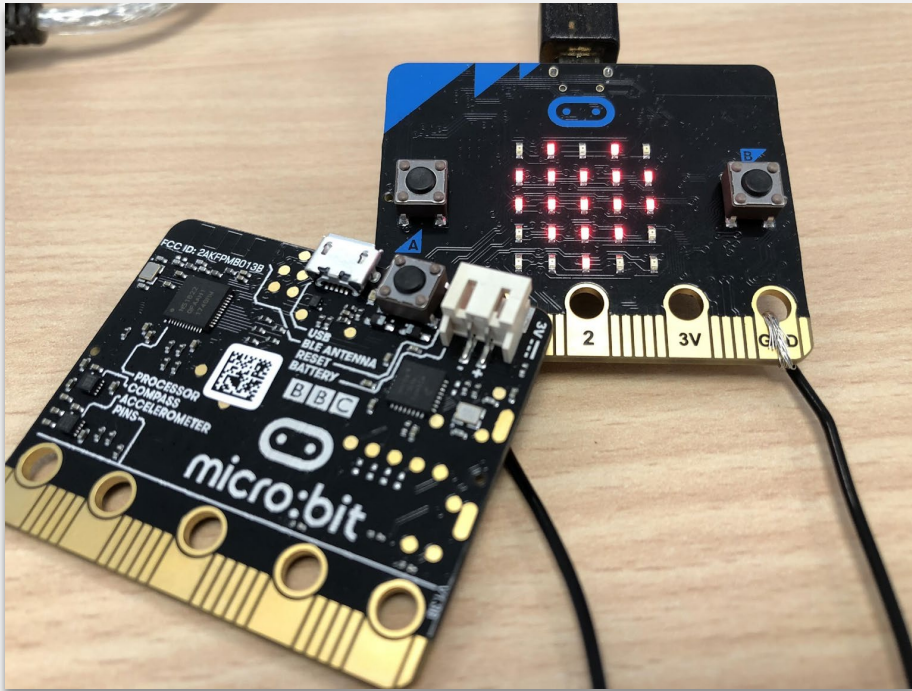
伺服馬達操作

單元一

micro:bit介紹

micro:bit介紹

什麼是micro:bit?

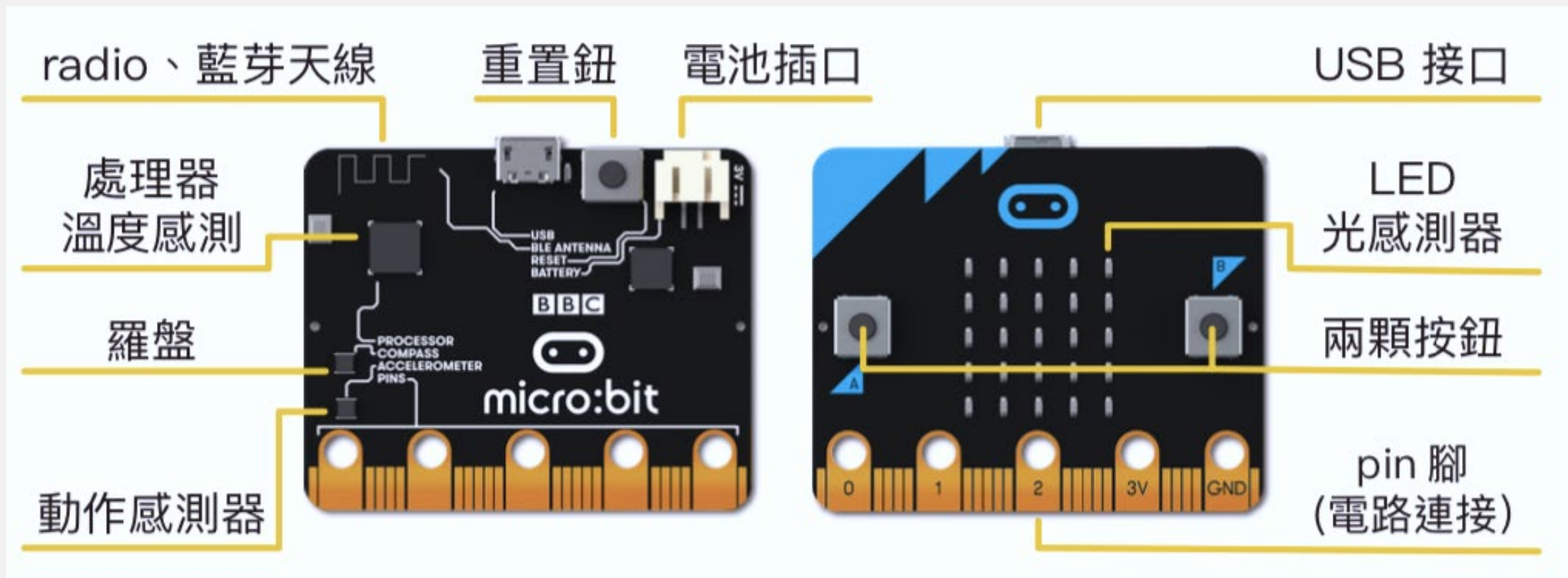


micro:bit 是一款由英國廣播電視公司 (BBC) 設計，專為青少年編程教育的微型電腦開發板。

BBC希望通過micro:bit驅動青少年參與創造性的硬件製作和軟件編程，而不是每天沉浸在各式的娛樂和消費中。

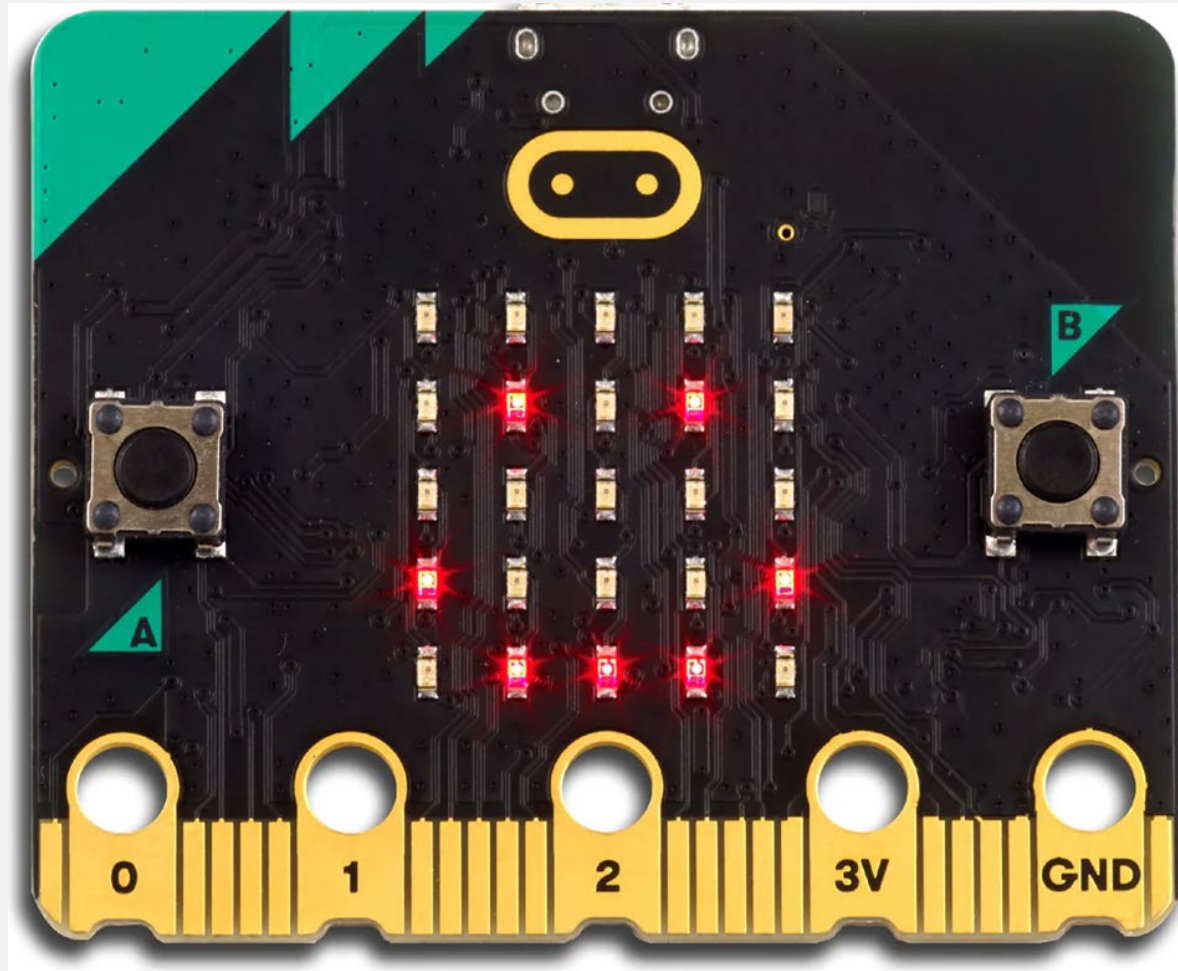
micro:bit介紹

microbit v1.5



micro:bit介紹

最新推出版本microbit v2

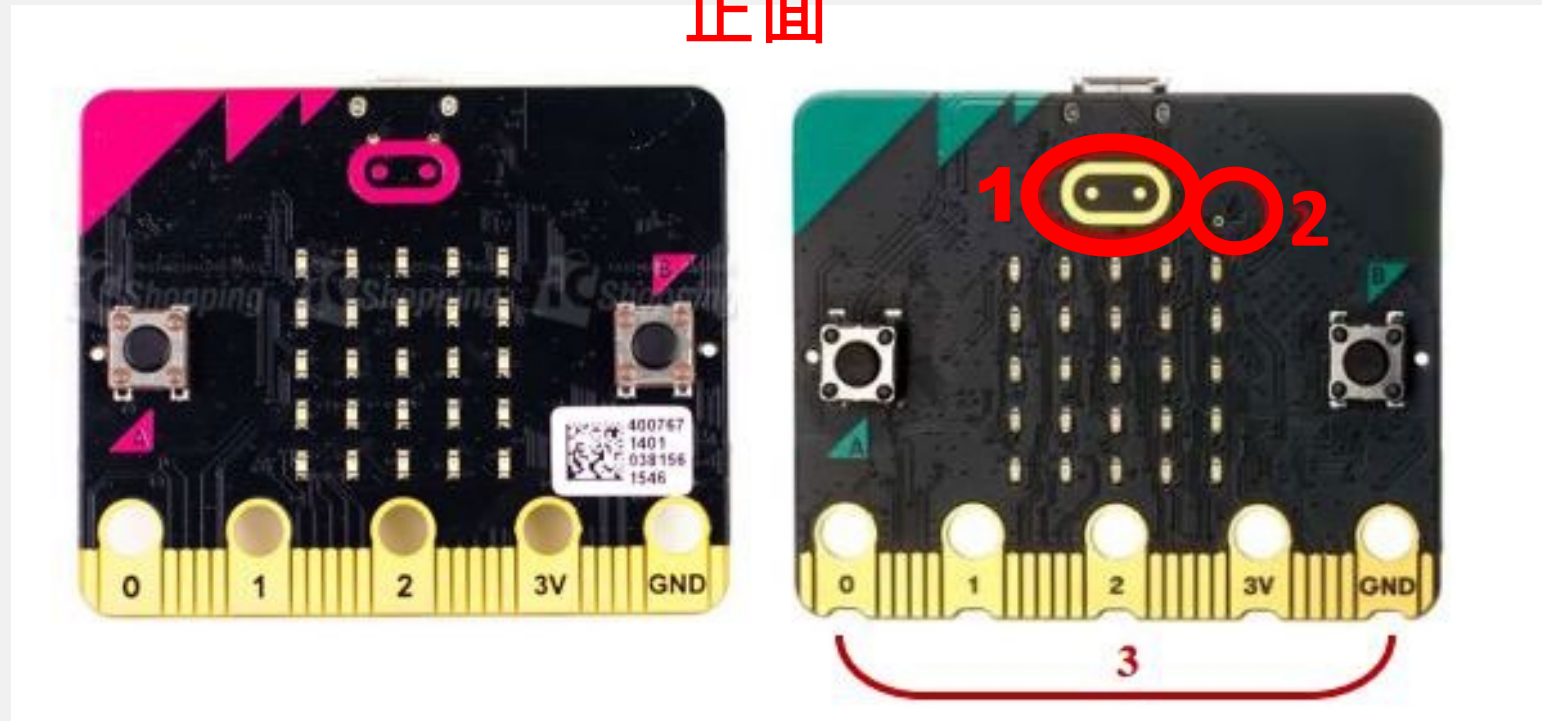


micro:bit介紹

microbit v1.5 與v2的差異

- 1: 電容式觸摸感測器
- 2: 聲音輸入孔與麥克風活動指示燈
- 3: 邊緣連接器的形狀變為凹痕

正面

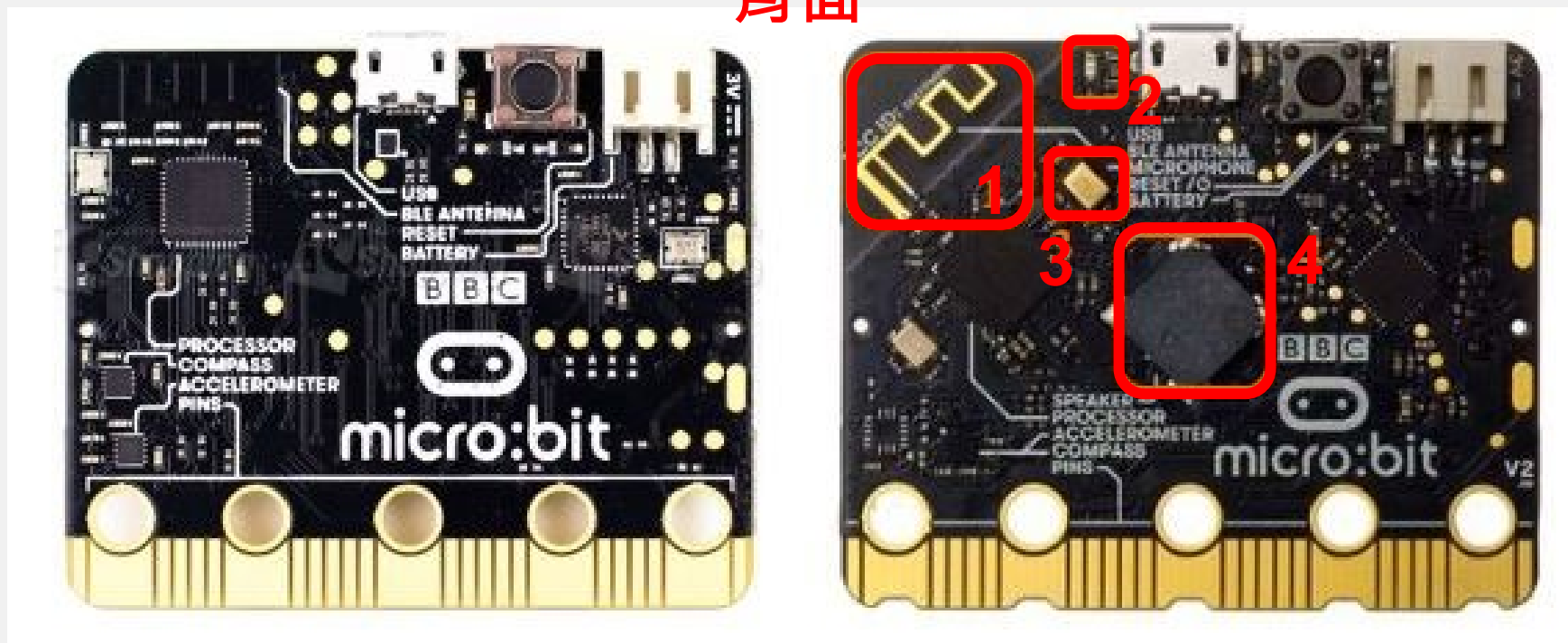


micro:bit介紹

microbit v1.5 與v2的差異

- 1: 鍍銅天線
- 2: 電源狀態燈
- 3: 聲音感測器
- 4: 喇叭

背面



micro:bit介紹

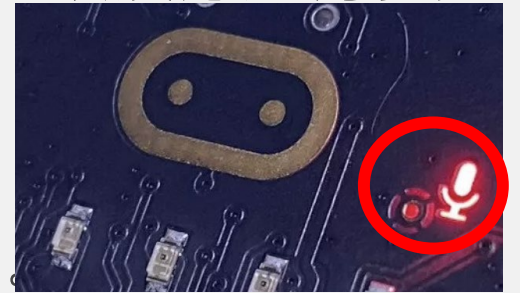
micro:bit v1.5 與 v2 兩個版本硬體規格上的差異

1. **處理器**：由原本的 Nordic nRF51822升級為 nRF52833，**處理效能大幅提升**
2. **記憶體**：內建記憶體從原本的256KB快閃記憶體和16KB RAM，**增加**為512KB快閃記憶體和128KB RAM。
3. **無線通信**：從原本藍牙 4.0 升級為藍牙 5.0，讓**傳輸距離更遠、速度更快**。
4. **喇叭**：增加於板子背面，讓創作者可以即時播放創作的音樂。可以模擬飛行中的大黃蜂，依據運動速度的不同，呈現生動的聲音頻率、速度和音量。

micro:bit介紹

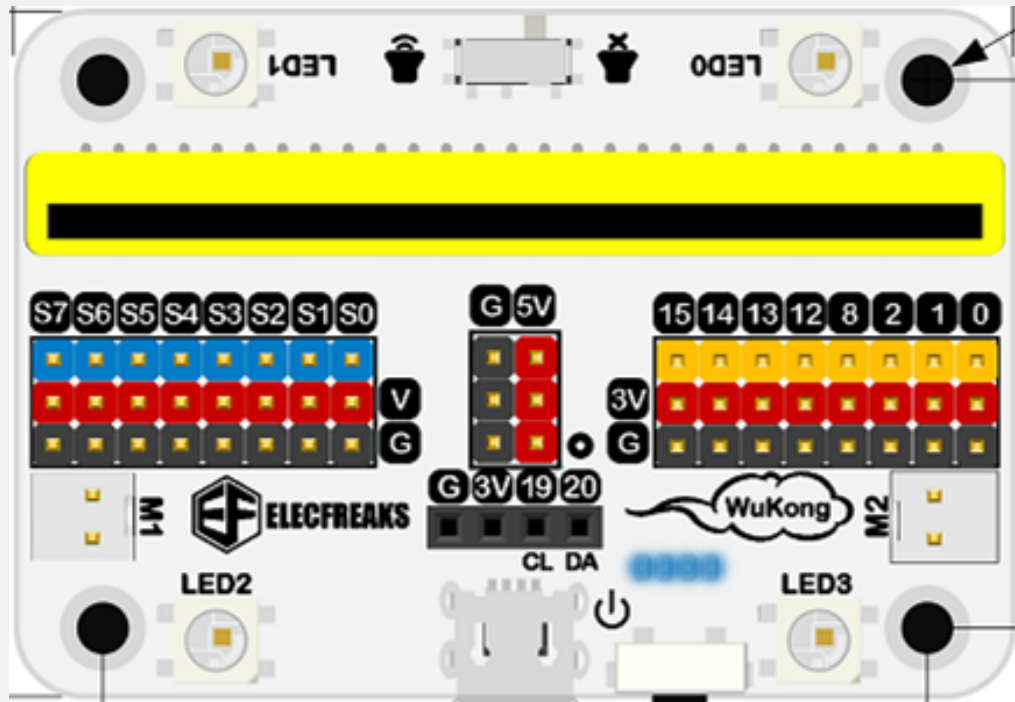
micro:bit v1.5 與 v2 兩個版本硬體規格上的差異

5. **聲音感測器**：增加於板子背面，而正面則是有一個聲音輸入孔，以及旁邊有麥克風活動指示燈，**麥克風使用時該指示燈會亮紅色燈號**。
6. **觸控感測器**：鍍銅處理並設置於正面上方，新版為**電容式觸碰**。
7. **內置的關機 / 睡眠模式**：原本板子背面的**重置按鈕**，新增睡眠與關機模式，透過**長按按鈕**即可關閉電路板電源，或者是轉為睡眠模式。



micro:bit介紹

Wukong 擴充板



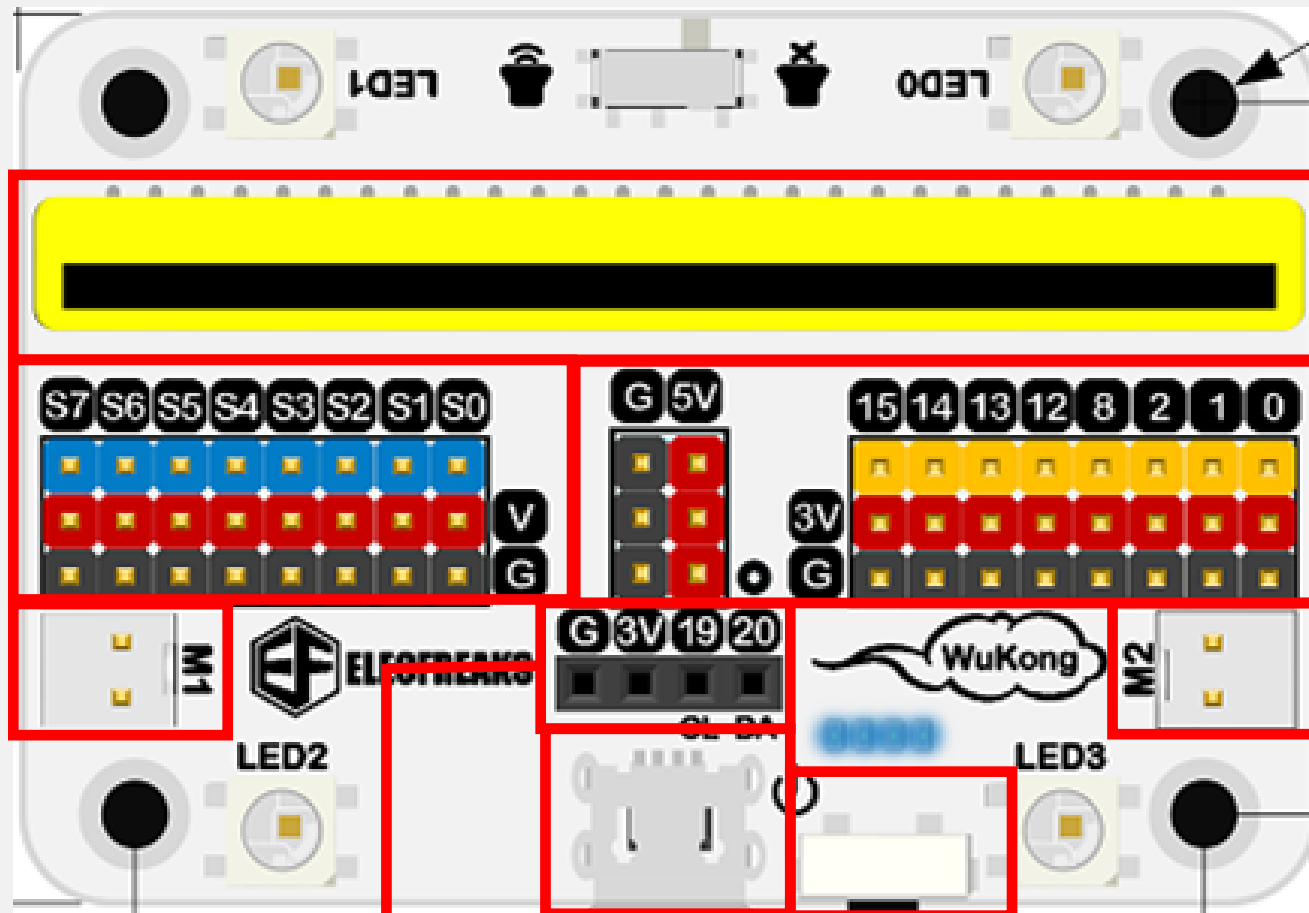
Wukong是一款基於micro:bit而製作的多功能性高整合度擴充板。

內建400mAh鋰離子電池組和板載電源控制系統，它僅需20分鐘即可進行快速充電，充滿電後可一次運行40分鐘以上。

注意！充電時，一定要將micro:bit拔起！

micro:bit介紹

Wukong 擴充板



伺服驅動器連接

馬達驅動連接

I2C排母連接器

電源供應

開關

micro: bit轉接器

8方標準GVS連接埠
和5v連接埠

馬達驅動連接

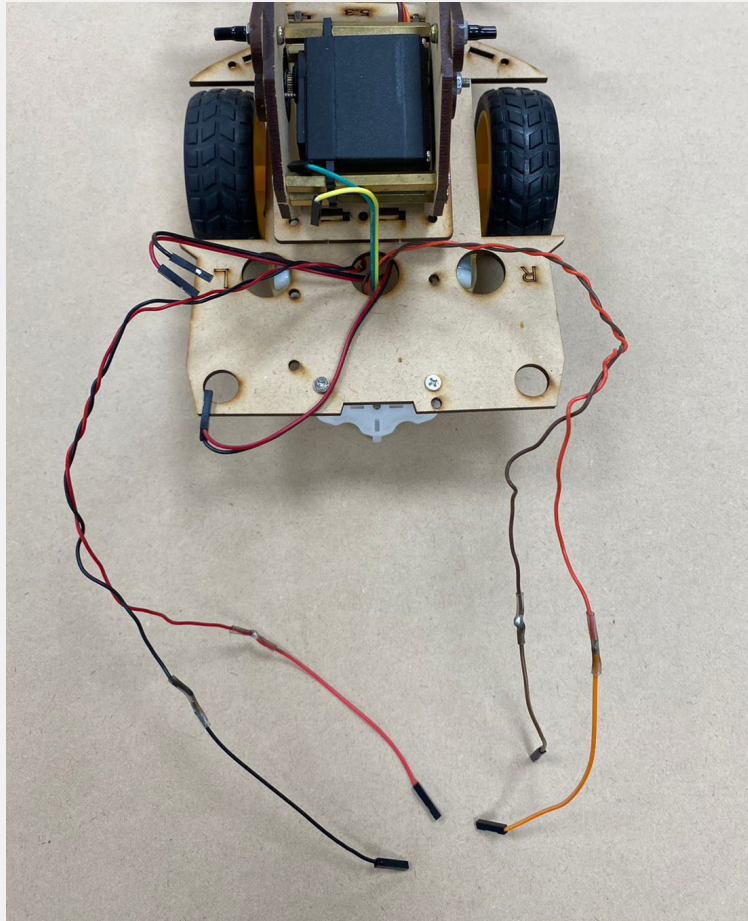
單元二

智慧小車組裝

智慧小車組裝

STEP.

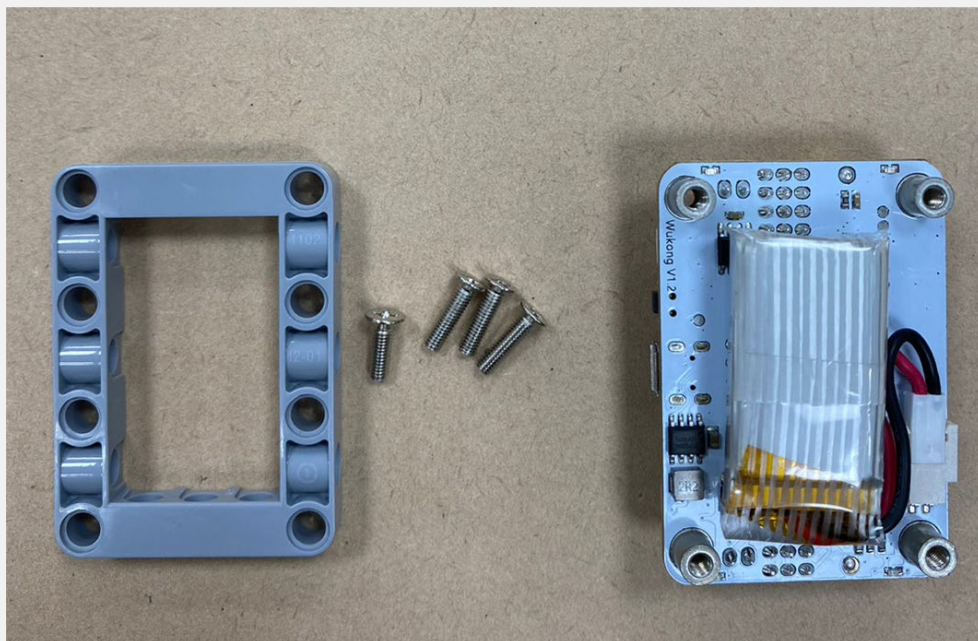
將馬達電線銲上母頭杜邦線 (共4條)



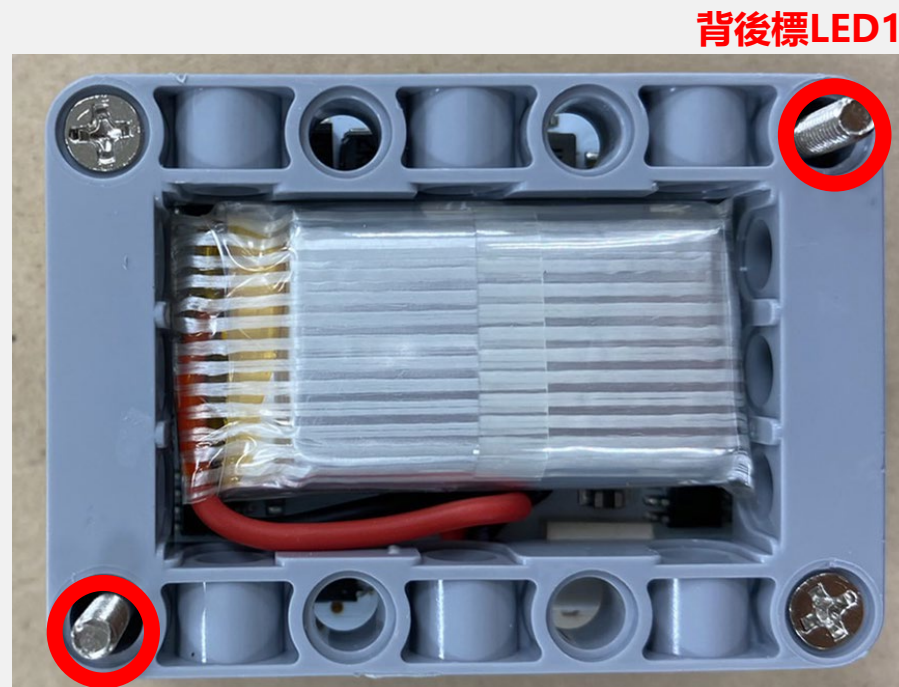
智慧小車組裝

STEP.

將擴充板拆下，並將其中兩顆螺絲反過來，再重新鎖回去



擴充板拆下



背後標LED3

兩顆螺絲反轉、鎖回去

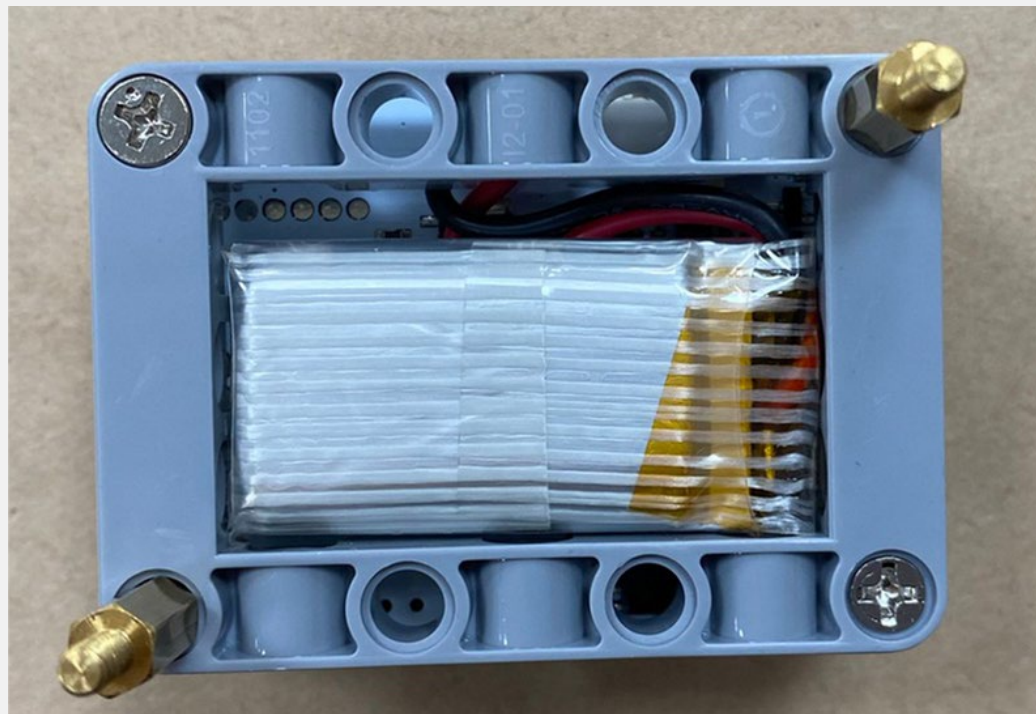
智慧小車組裝

STEP.

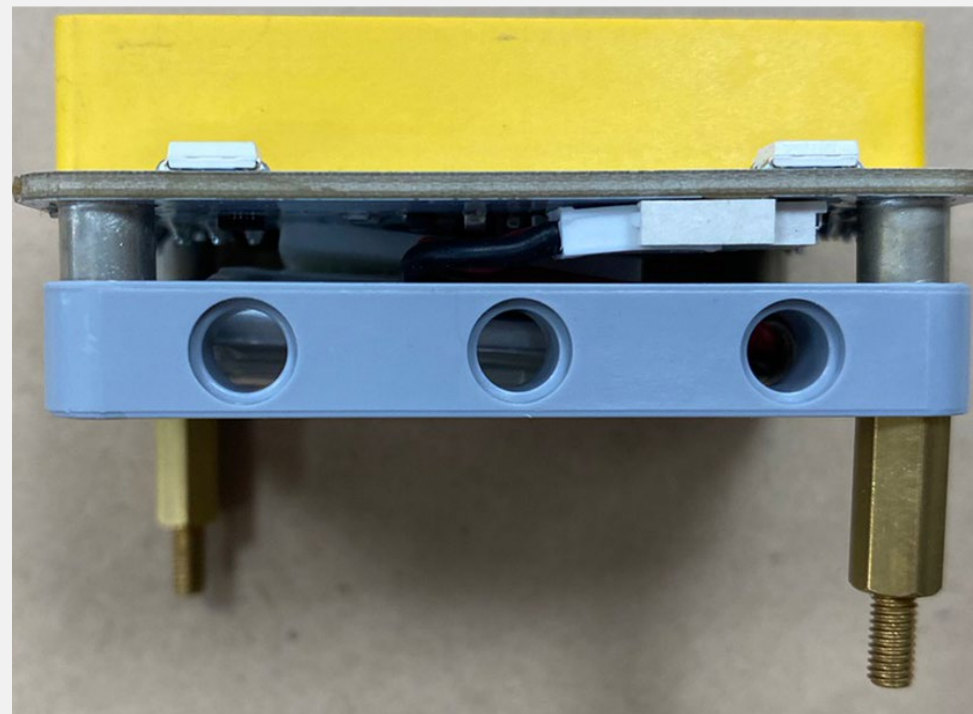
3

將公母銅柱鎖在反過來的兩顆螺絲上

12mm公母銅柱 X 2



仰視圖



後視圖

智慧小車組裝

STEP.

4 將螺絲鎖在左上、右下的母母銅柱

半圓在上

M8,10mm螺絲 X 2



20mm母母銅柱 X 2



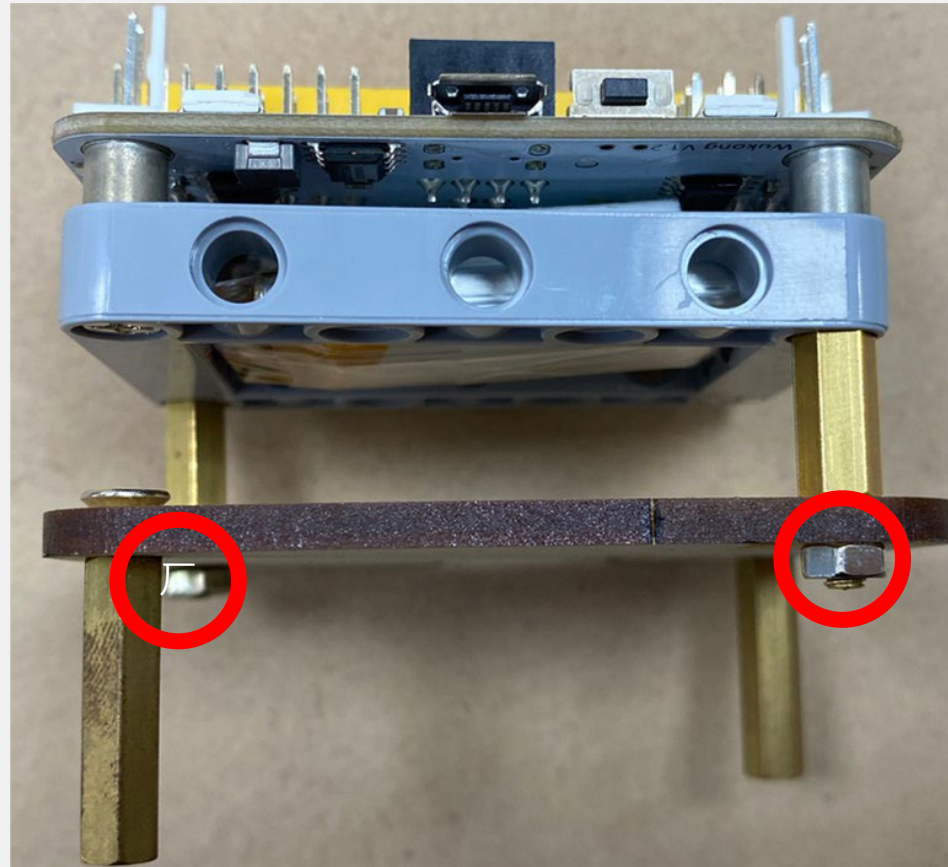
智慧小車組裝

STEP.

5

將擴充板鎖在板子上

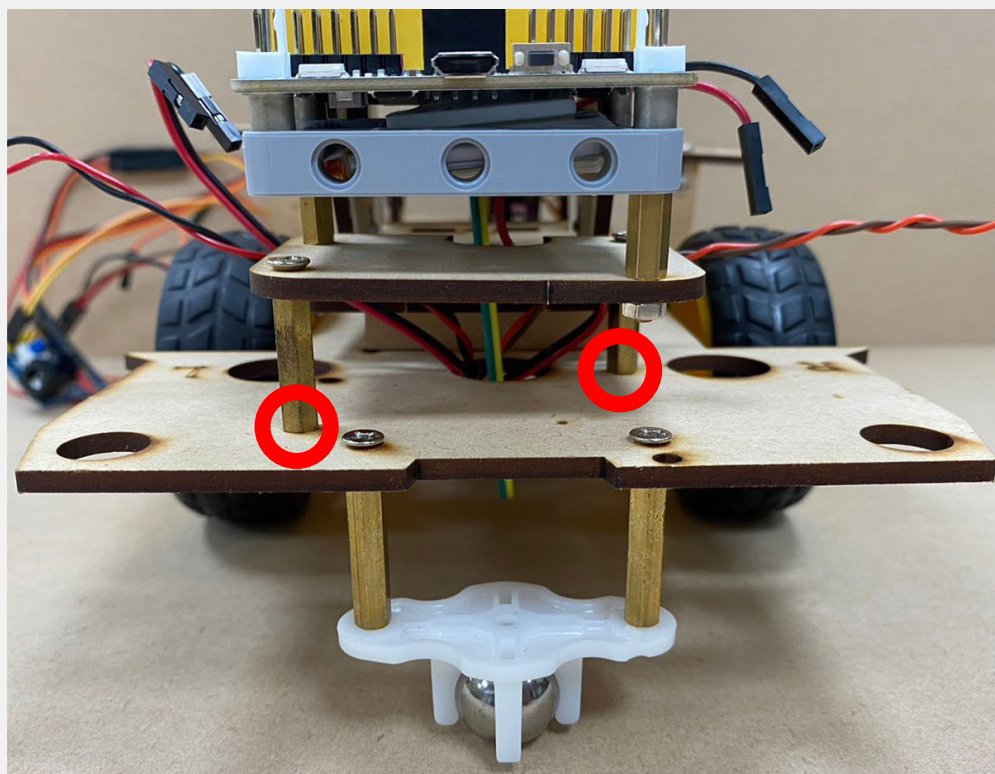
M8止滑螺帽 X 2



智慧小車組裝

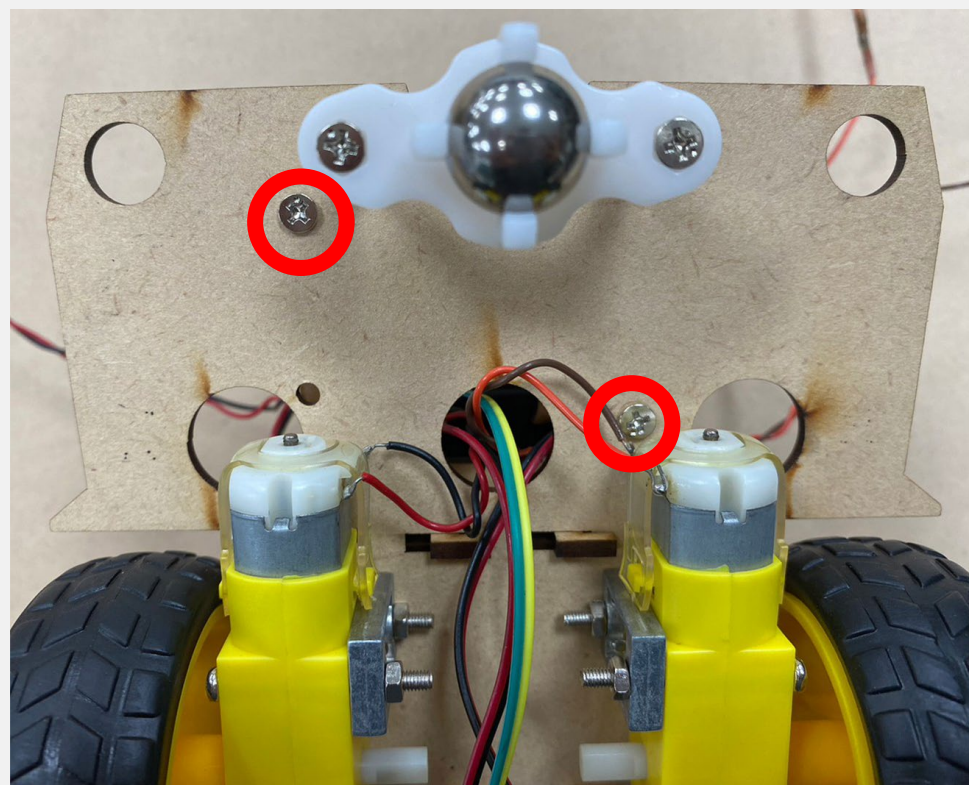
STEP.

6 將擴充板整組鎖在車體



後視圖

M8,10mm螺絲 X 2



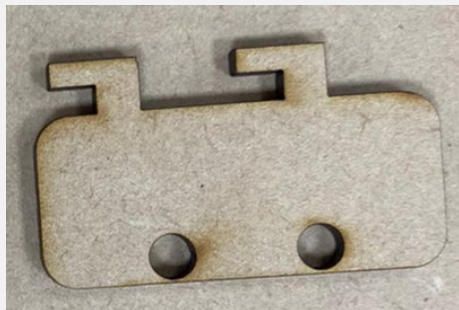
仰視圖

智慧小車組裝

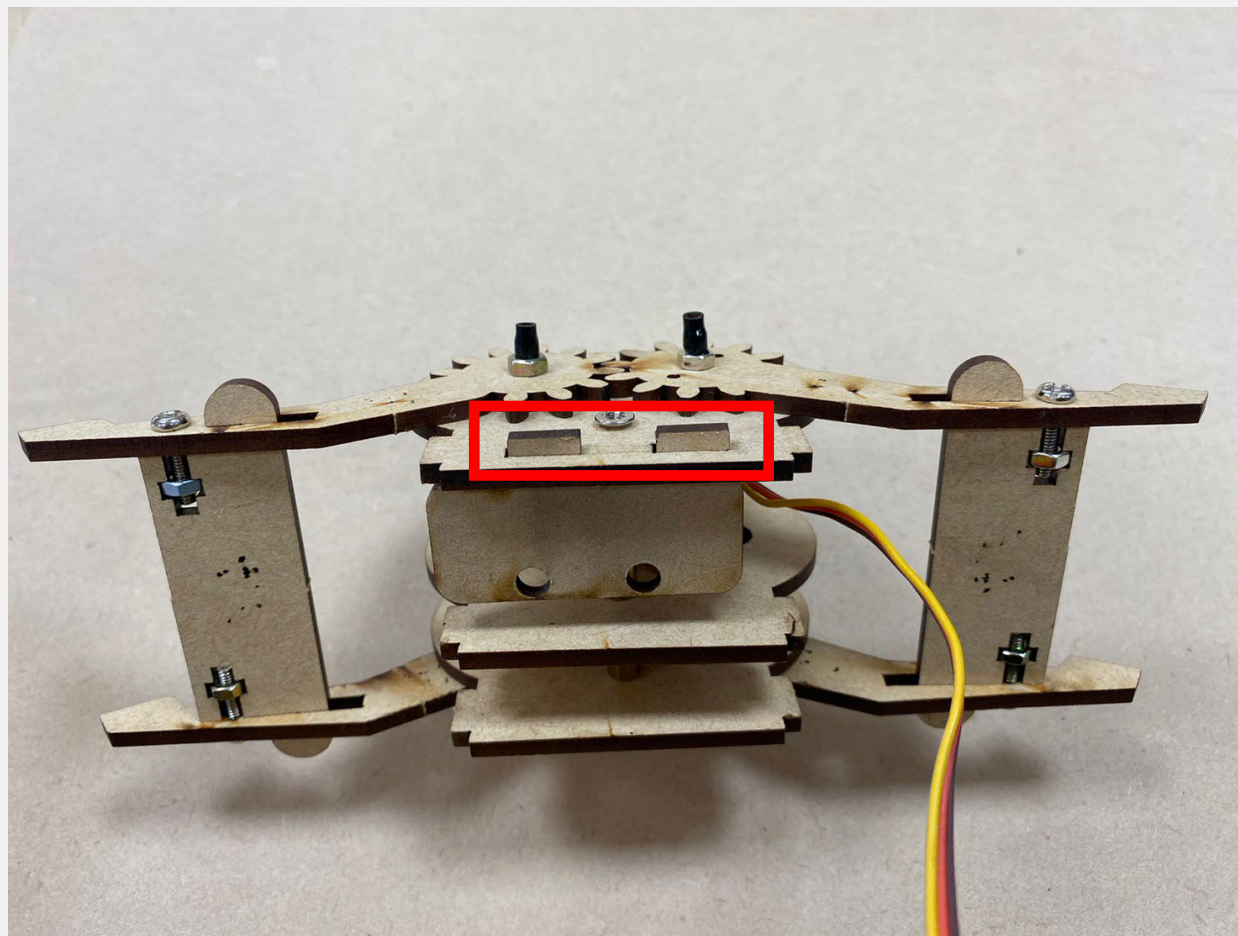
STEP.

7

將小方塊卡在爪子上



小方塊



智慧小車組裝

STEP.

8

鏡頭卡上2根插銷



智慧小車組裝

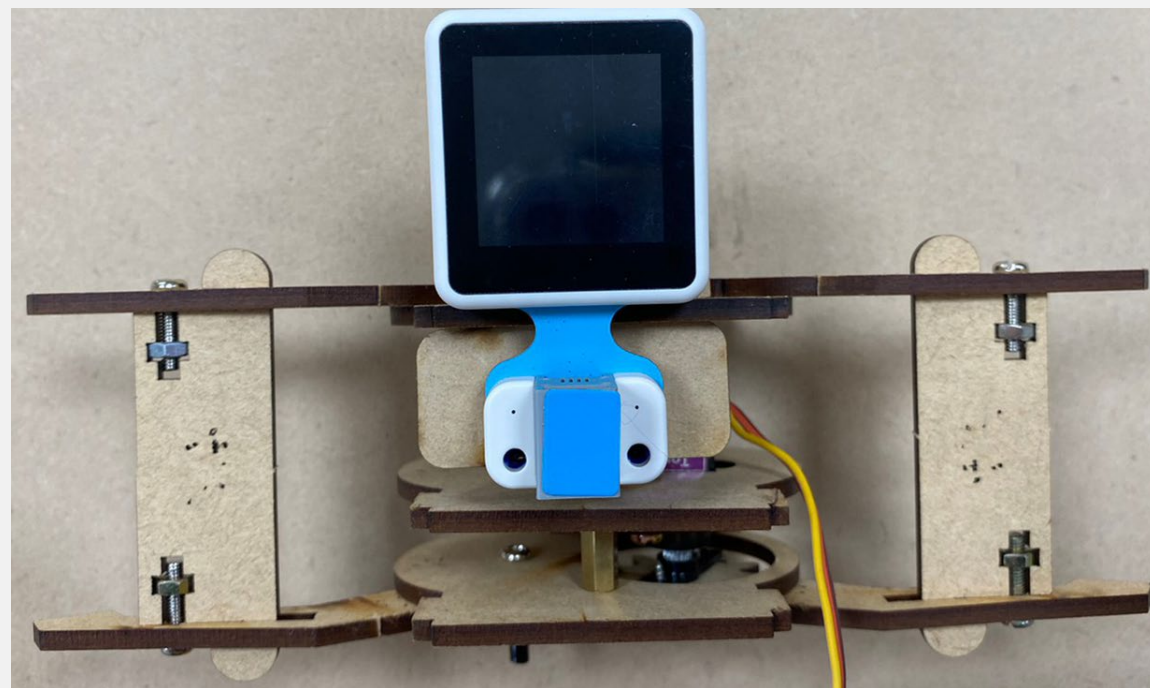
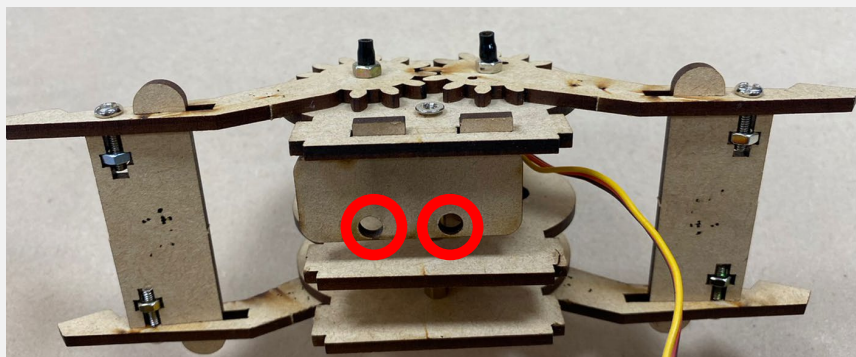
STEP.

9 將鏡頭的插銷卡進去小方塊的圓孔



+

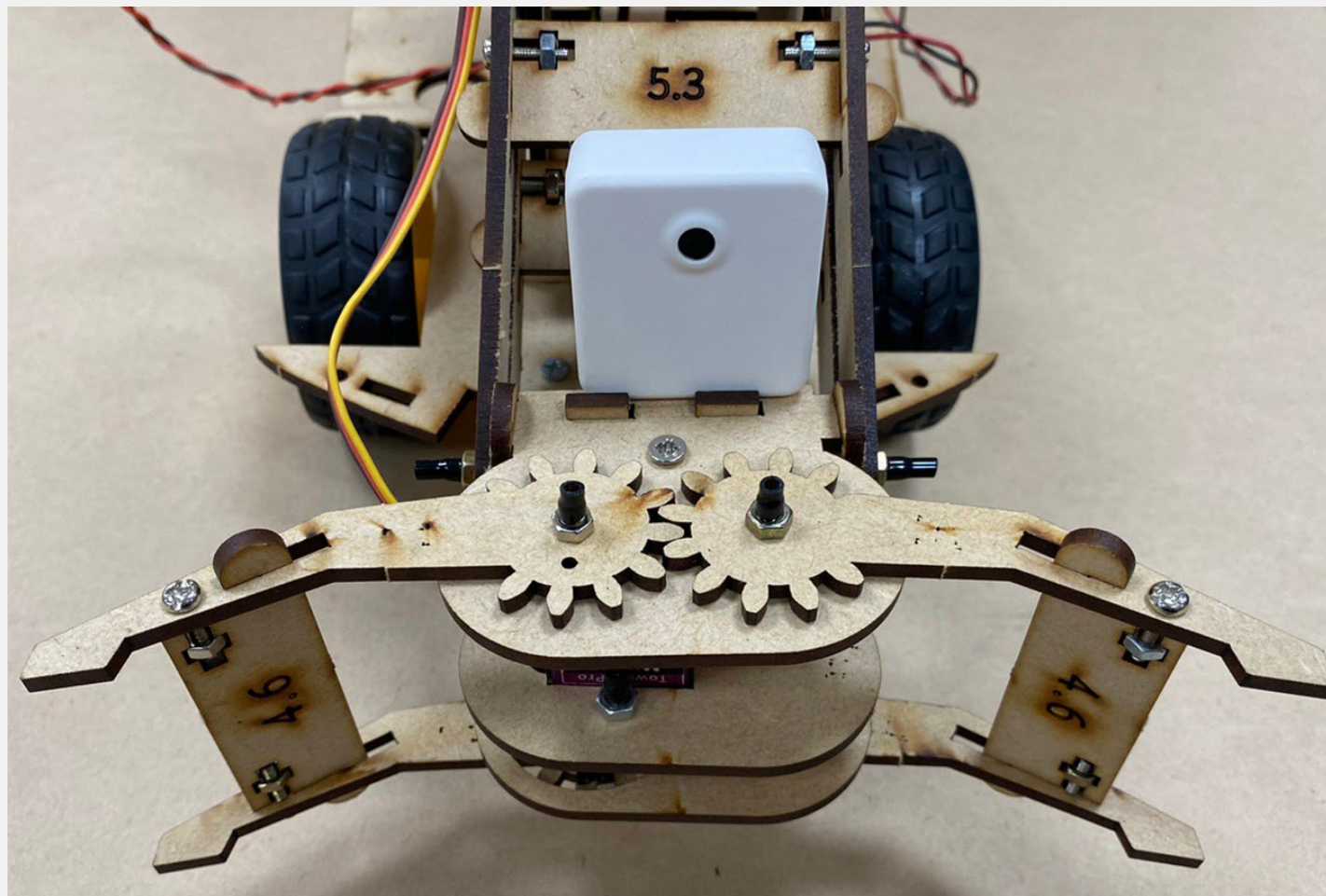
=



智慧小車組裝

STEP.1

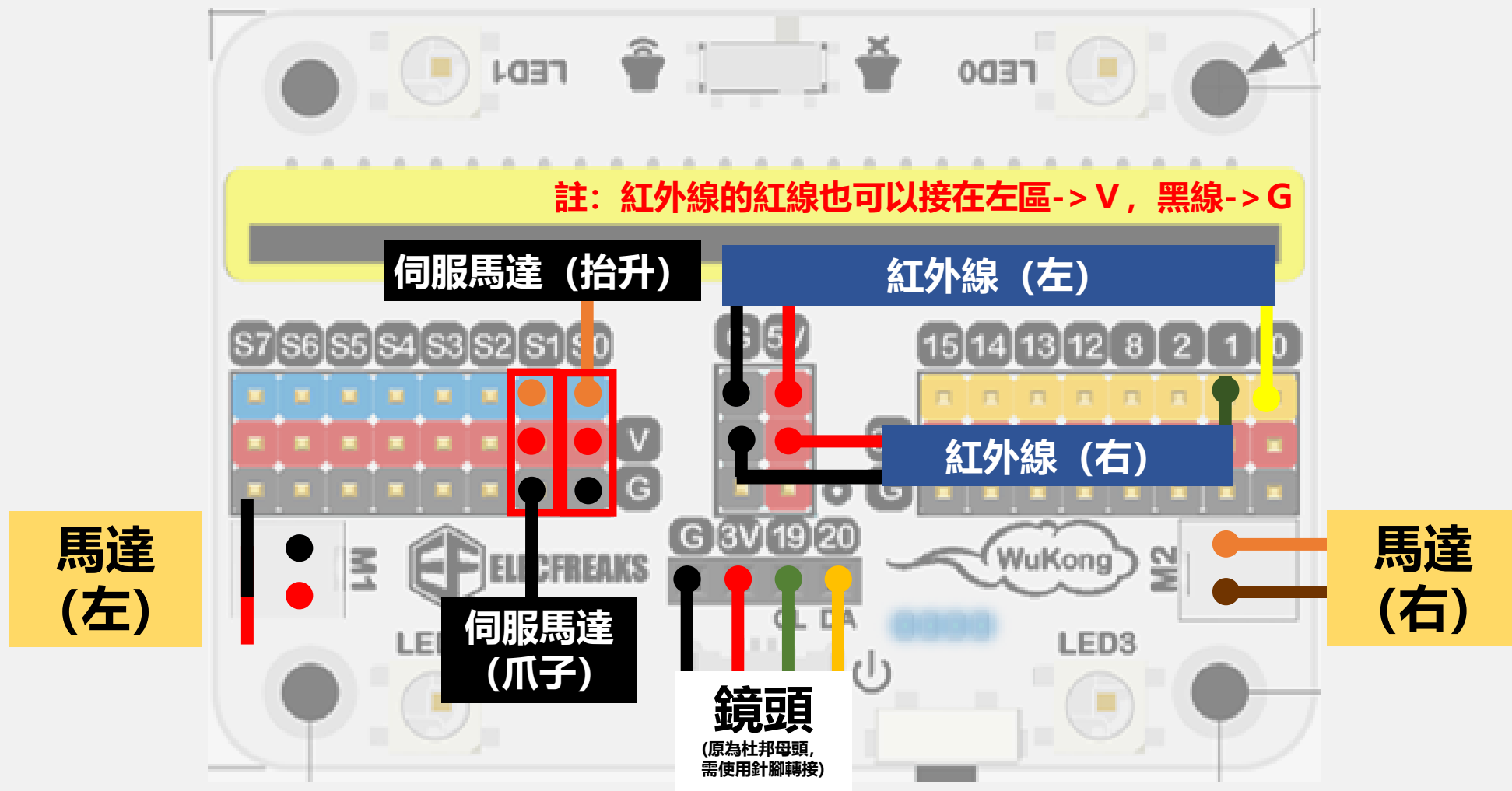
0 將爪子卡回車體



智慧小車組裝

STEP.1

1 接線

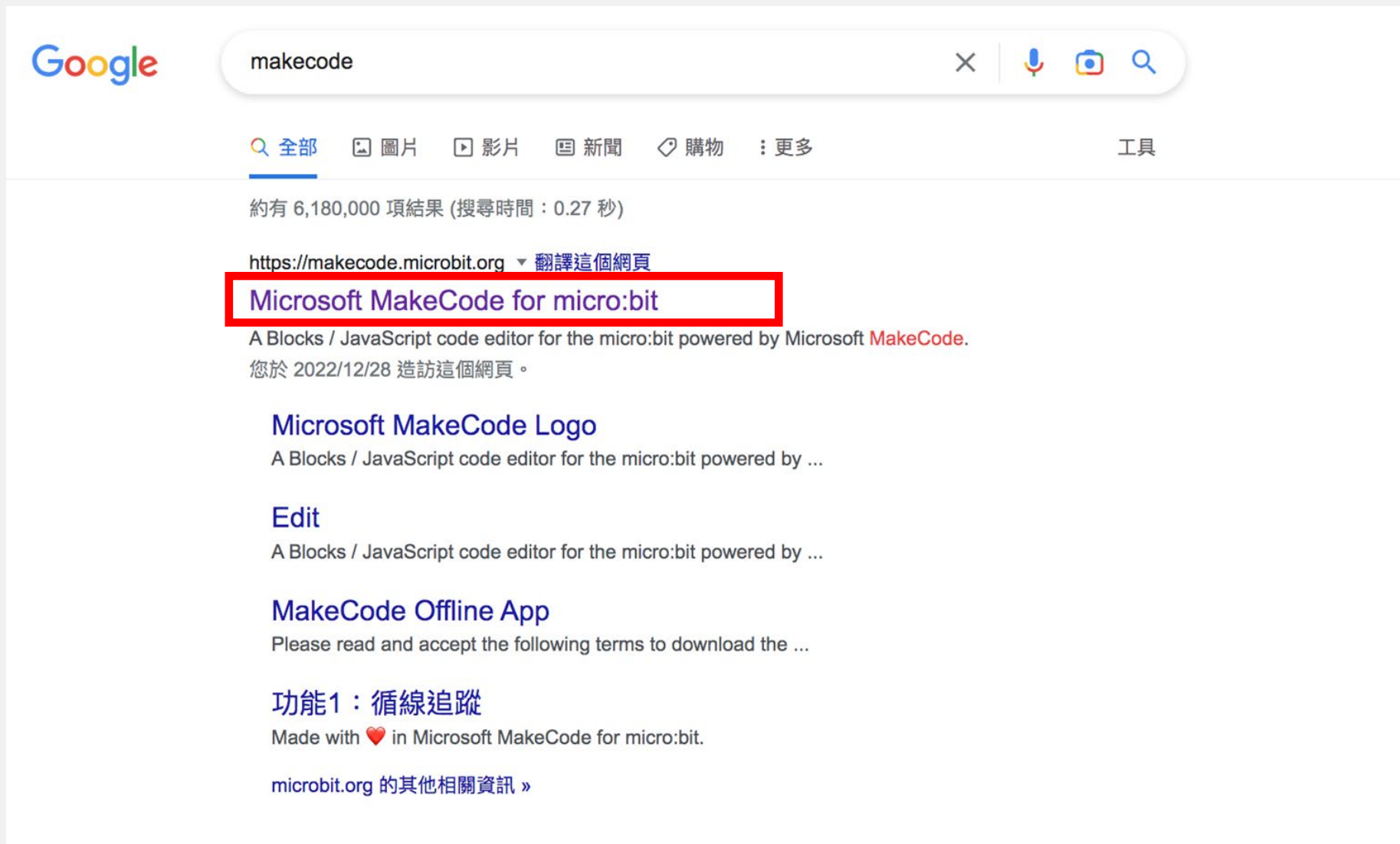


單元三

程式環境設定

程式環境設定

搜尋makecode



The image shows a Google search interface. The search bar contains the text 'makecode'. Below the search bar, there are navigation options: '全部' (All), '圖片' (Images), '影片' (Videos), '新聞' (News), '購物' (Shopping), and '更多' (More). The search results show approximately 6,180,000 results found in 0.27 seconds. The top result is for 'https://makecode.microbit.org' with a sub-heading 'Microsoft MakeCode for micro:bit' highlighted in a red box. Below this, there are several other links related to Microsoft MakeCode, including 'Microsoft MakeCode Logo', 'Edit', 'MakeCode Offline App', and '功能1：循線追蹤'.

Google

makecode

全部 圖片 影片 新聞 購物 更多 工具

約有 6,180,000 項結果 (搜尋時間：0.27 秒)

<https://makecode.microbit.org> 翻譯這個網頁

Microsoft MakeCode for micro:bit

A Blocks / JavaScript code editor for the micro:bit powered by Microsoft **MakeCode**.
您於 2022/12/28 造訪這個網頁。

Microsoft MakeCode Logo
A Blocks / JavaScript code editor for the micro:bit powered by ...

Edit
A Blocks / JavaScript code editor for the micro:bit powered by ...

MakeCode Offline App
Please read and accept the following terms to download the ...

功能1：循線追蹤
Made with ❤️ in Microsoft MakeCode for micro:bit.

[microbit.org 的其他相關資訊 »](#)

程式環境設定

可選擇語言

選擇語言



我的所有專案 [檢視全部](#)

[匯入](#)



程式環境設定

點擊建立新專案

Introduction to the BBC micro:bit

檢視文件

我的所有專案 [檢視全部](#) 匯入

新增專案
建立一個新的空白專案

test1228 5 日前

123 5 日前

教程

New? Start Here!

Flashing Heart

Name Tag

Smiley Buttons

Dice

Love Meter

程式環境設定

命名與創建

Microsoft | micro:bit

Track the soil moisture of your plants!

檢視文件

我的所有專案 檢視全部

test1

新增專案

教程

建立專案 🥰🥰🥰

幫你的專案取個名字。

serial port detection

1.命名

> 程式碼選項

2.點擊創建

創建 (Create) ✓

程式環境設定

點擊擴展



程式環境設定

新增wukong擴展板

← Go Back 擴展

1. 搜尋wukong 🔍

Lights and Display Software 科學 Robotics Gaming Networking

主頁

2. 點擊wukong



wukong
(悟空)micro:bit multifunctional expansion board WuKong

[Learn More](#)



longanbit
The LonganBit (英荔比特, "龙眼"电子积木学习套件) served as a kit for education of AI, containing...

[Learn More](#)

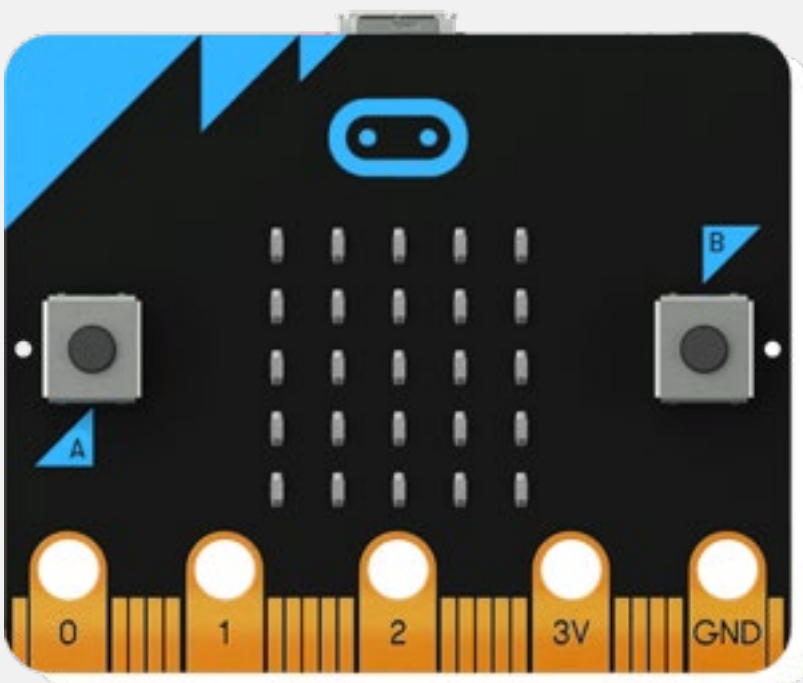
單元四

micro:bit連接

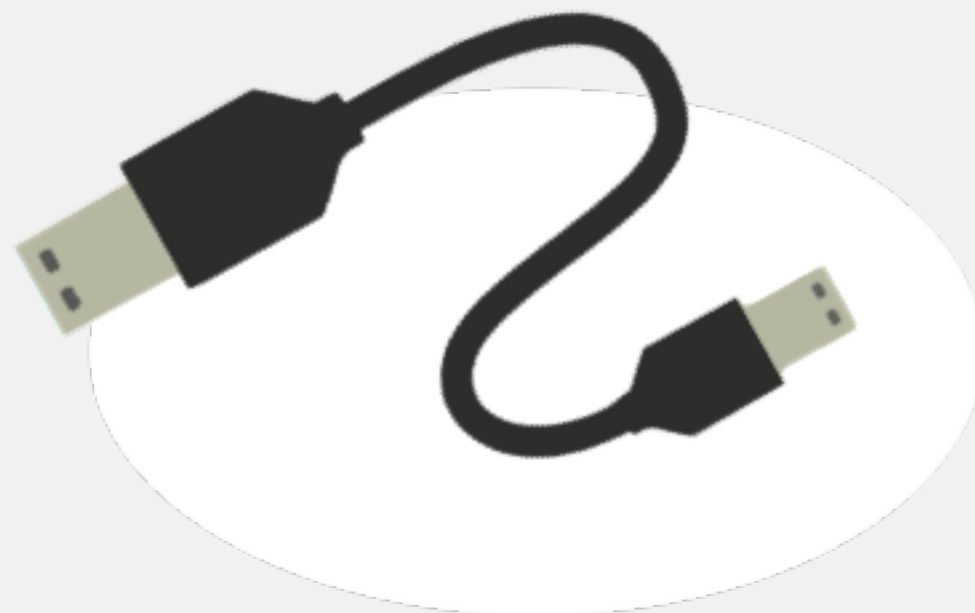
micro:bit連接

注意：上傳程式碼時勿將micro:bit插在擴充板上，以免導致擴充板受損

檢查裝備



主板



Micro USB 連接
線

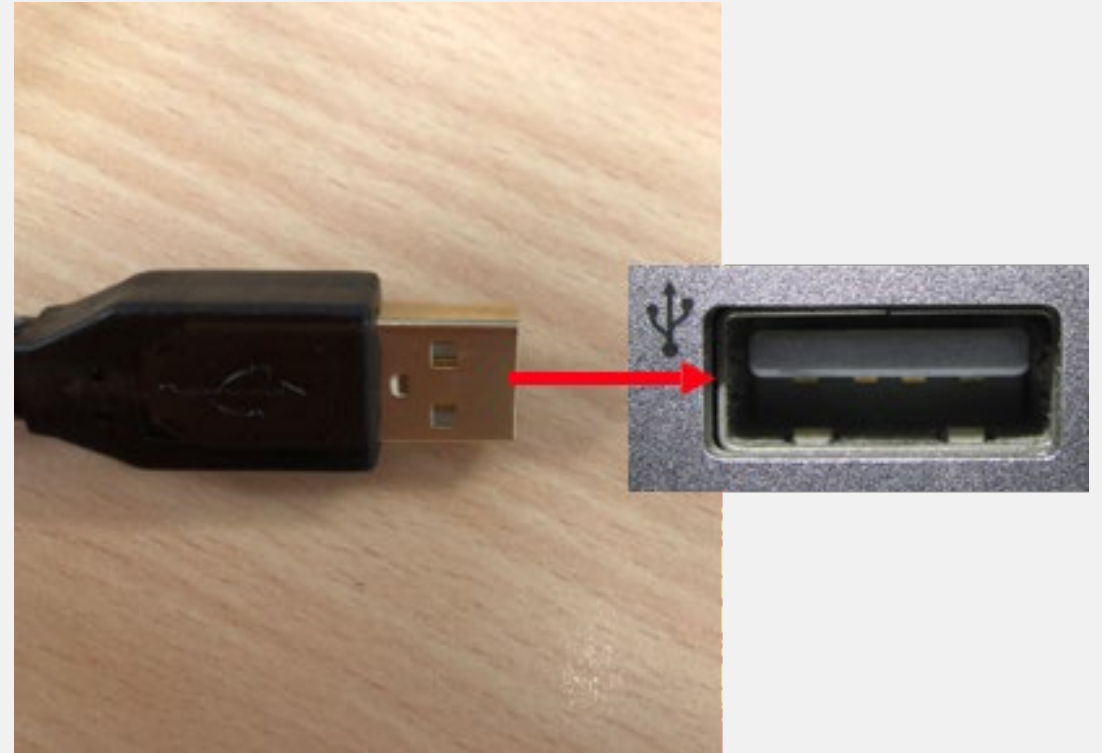
micro:bit連接

注意：上傳程式碼時勿將micro:bit插在擴充板上，以免導致擴充板受損

step1:將連接線接上主板 (注意正反方向)



step2:將 USB 接上電腦



micro:bit連接

注意：上傳程式碼時勿將micro:bit插在擴充板上，以免導致擴充板受損

上傳程式碼到micro:bit

方法一

搜尋...

- 基本
- 輸入
- 音效
- 燈光
- 廣播
- 迴圈
- 邏輯
- 變數
- 數學
- 悟空
- PlanetX_AI-Lens

連接裝置

另1 連接裝置

說明

下載

循跡

1. 點擊...

2. 點擊連接裝置

micro:bit連接

注意：上傳程式碼時勿將micro:bit插在擴充板上，以免導致擴充板受損

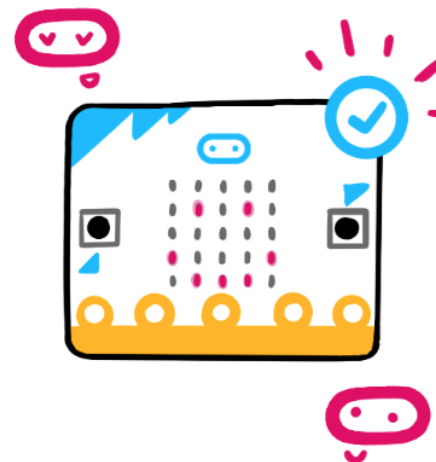
上傳程式碼到micro:bit 方法一

✓ 已連線至 micro:bit



您的 micro:bit 已連結！按下「下載」會自動將您的程式碼複製到您的 micro:bit。

如果您要取消配對此 micro:bit，您可以透過點擊「下載」按鈕旁的 '...' 進行此操作



完成

3.完成連線

micro:bit連接

注意：上傳程式碼時勿將micro:bit插在擴充板上，以免導致擴充板受損

上傳程式碼到micro:bit

方法一

The screenshot displays the Microsoft MakeCode editor for micro:bit. The top navigation bar includes the Microsoft logo, the micro:bit logo, a '積木' (Blocks) button, and a dropdown menu for 'JavaScript'. The main workspace is divided into three sections: a virtual micro:bit board on the left, a block-based programming palette in the center, and a script editor on the right. The palette includes categories like '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), '悟空' (Wukong), 'PlanetX_AI-Lens', '彩虹灯' (Rainbow Light), and '擴展' (Extensions). The script editor shows a sequence of blocks for controlling the micro:bit's LEDs. At the bottom of the interface, a red box highlights the '下載' (Download) button, which is used to upload the code to the physical micro:bit device.

4. 點擊下載

micro:bit連接

注意：上傳程式碼時勿將micro:bit插在擴充板上，以免導致擴充板受損

上傳程式碼到micro:bit

方法一

The screenshot displays the Microsoft MakeCode editor for micro:bit. The top navigation bar includes the Microsoft logo, the 'micro:bit' label, a '積木' (Blocks) button, and a dropdown menu currently set to 'JavaScript'. The main workspace is divided into three sections: a virtual micro:bit board on the left, a central block palette, and a code editor on the right. The block palette lists various categories such as '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), '悟空' (Wukong), 'PlanetX_AI-Lens', '彩虹灯' (Rainbow Light), and '擴展' (Extensions). The code editor on the right shows a JavaScript script with several 'onEvent' blocks. At the bottom of the interface, a dark blue notification bar with a download icon and the text 'Downloaded!' is highlighted with a red rectangular box.

5.表示成功上傳

上傳程式碼到micro:bit

方法二

The screenshot displays the Microsoft MakeCode editor for micro:bit. The top navigation bar includes the Microsoft logo, the 'micro:bit' label, a '積木' (Blocks) button, and a dropdown menu currently set to 'JavaScript'. The main workspace is divided into three sections: a micro:bit board on the left, a block palette in the center, and a script area on the right. The block palette lists various categories such as '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), '悟空' (Wukong), 'PlanetX_AI-Lens', '彩虹灯' (Rainbow Light), and '擴展' (Extensions). The script area contains a sequence of blocks for controlling LEDs. At the bottom of the workspace, a red box highlights the '下載' (Download) button, which is used to upload the code to the micro:bit device.

1. 直接點擊下載

上傳程式碼到micro:bit 方法二

下載完成...



您的程式碼將被下載成 .hex 檔案。您可以透過檔案管理員，將下載好的檔案拖曳至您的 micro:bit 。

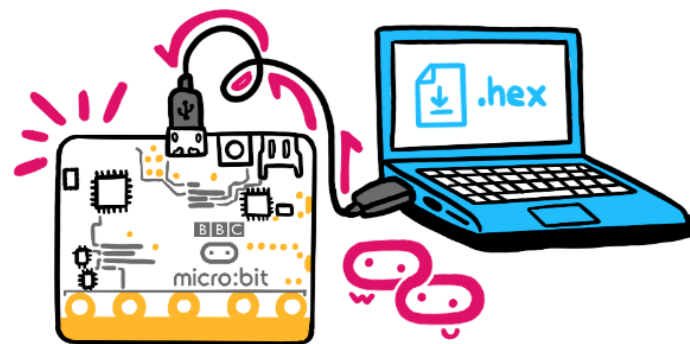
New!



透過Web USB配對，將程式碼
更快地下載到裝置中！
立即配對



不再顯示



2. 點擊完成

說明

再次下載

完成

上傳程式碼到micro:bit 方法二

3.將檔案拖曳進MICROBIT,完成上傳

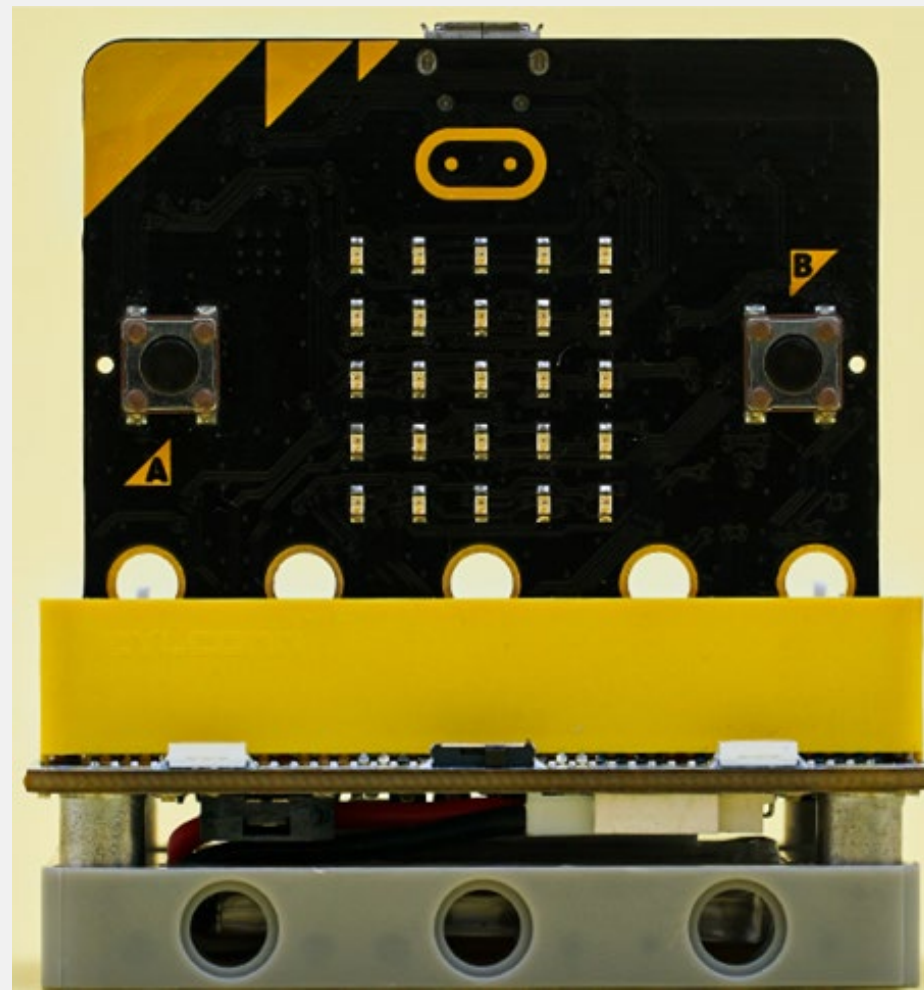
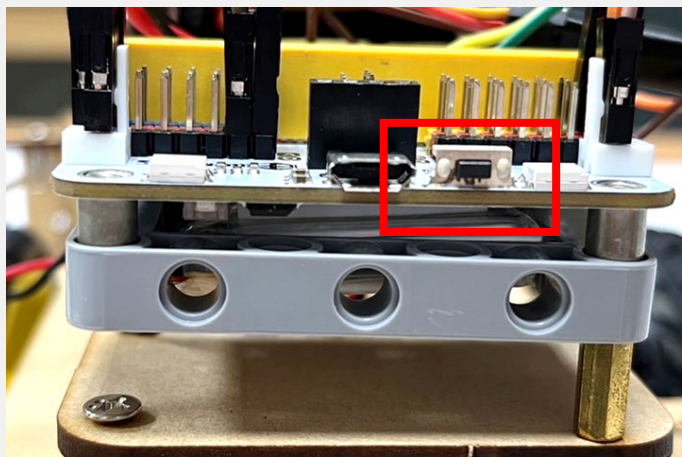


micro:bit連接

注意：上傳程式碼時勿將micro:bit插在擴充板上，以免導致擴充板受損

micro:bit連接擴充板

- ◆ 將micro: bit插入轉接器
(將顯示器朝向寫有wukong字樣的那側)
- ◆ 按一下以打開電源
- ◆ 按兩下以關閉電源



單元五

TT直流馬達

TT直流馬達

車輛前進

當啟動時

设置电机M1的速度为 100 电机M2的 速度为 100

M1(左輪)
M2(右輪)

先不要將程式放到迴圈裡，觀察一下馬達運轉情形。

TT直流馬達

車輛前進

當啟動時

设置电机M1的速度为 100 电机M2的 速度为 100

M1(左輪)
M2(右輪)

為什麼程式沒有放在迴圈卻能不停運轉？

馬達控制與電燈開關相同，是一種「狀態設定」。

打開後便會一直維持著，跟賽車遊戲放開按鍵便會停止不一樣。

TT直流馬達

車輛停止



M1(左輪)
M2(右輪)

車輛停止，將速度設為0



想想看：車輛後退該如何設定？

TT直流馬達

副程式的使用

我們可以將常用的程式碼包裝成一個副程式

The screenshot displays the programming environment for the TT DC Motor. On the left, a sidebar menu lists various categories: 變數 (Variables), 數學 (Math), 悟空 (Wukong), 彩虹灯 (Rainbow Light), 擴展 (Extensions), 進階 (Advanced), 函式 (Functions), 陣列 (Arrays), 文字 (Text), 遊戲 (Games), and 圖像 (Images). The '函式' (Functions) category is selected and highlighted in blue. Below the sidebar, a dark panel titled 'f(x) 函式' (Functions) contains a red-bordered button labeled '建立一個函式...' (Create a function...). Below this are a '返回' (Return) block with a value of 0 and a minus sign, and a section titled 'Your Functions' which lists a function named '呼叫 forward' (Call forward). The main workspace on the right is a light blue grid with the text '建立副程式' (Create sub-program) in large red characters. It shows two examples of function usage: 1. A blue '當啟動時' (When started) block containing a '呼叫 forward' block. 2. A blue '定義函式' (Define function) block for 'forward' containing an orange block that sets the speed of motor M1 to 50 and motor M2 to 50.

TT直流馬達

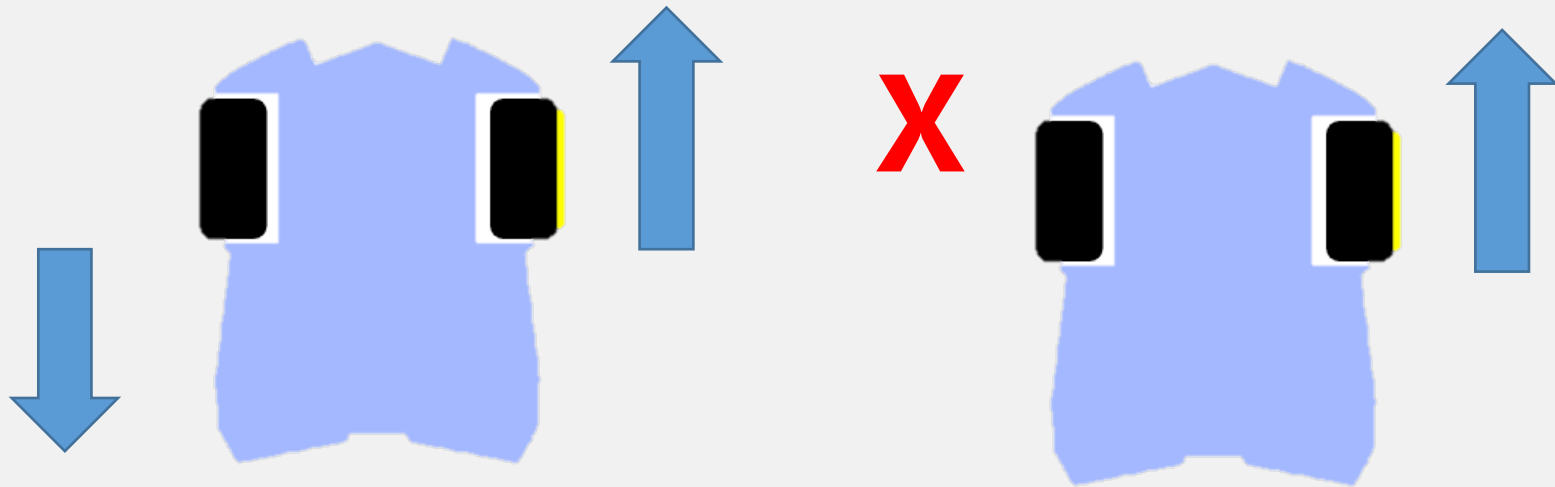
轉彎的方式



原地自轉
turn_0

TT直流馬達

轉彎的方式



原地自轉
turn_0

單輪轉彎
turn_1

TT直流馬達

轉彎的方式



原地自轉
turn_0

X



單輪轉彎
turn_1



差速轉彎
turn_2

TT直流馬達

轉彎的方式

比較一下不同的轉彎方法
有無差異？

```
當啟動時  
  呼叫 forward  
  暫停 3000 毫秒  
  呼叫 turn_0  
  暫停 3000 毫秒  
  呼叫 turn_1  
  暫停 3000 毫秒  
  呼叫 turn_2  
  暫停 3000 毫秒
```

```
定義函式 forward  
  设置电机M1的速度为 100 电机M2的 速度为 100
```

```
定義函式 turn_0  
  设置电机M1的速度为 -100 电机M2的 速度为 100
```

```
定義函式 turn_1  
  设置电机M1的速度为 0 电机M2的 速度为 100
```

```
定義函式 turn_2  
  设置电机M1的速度为 40 电机M2的 速度为 100
```

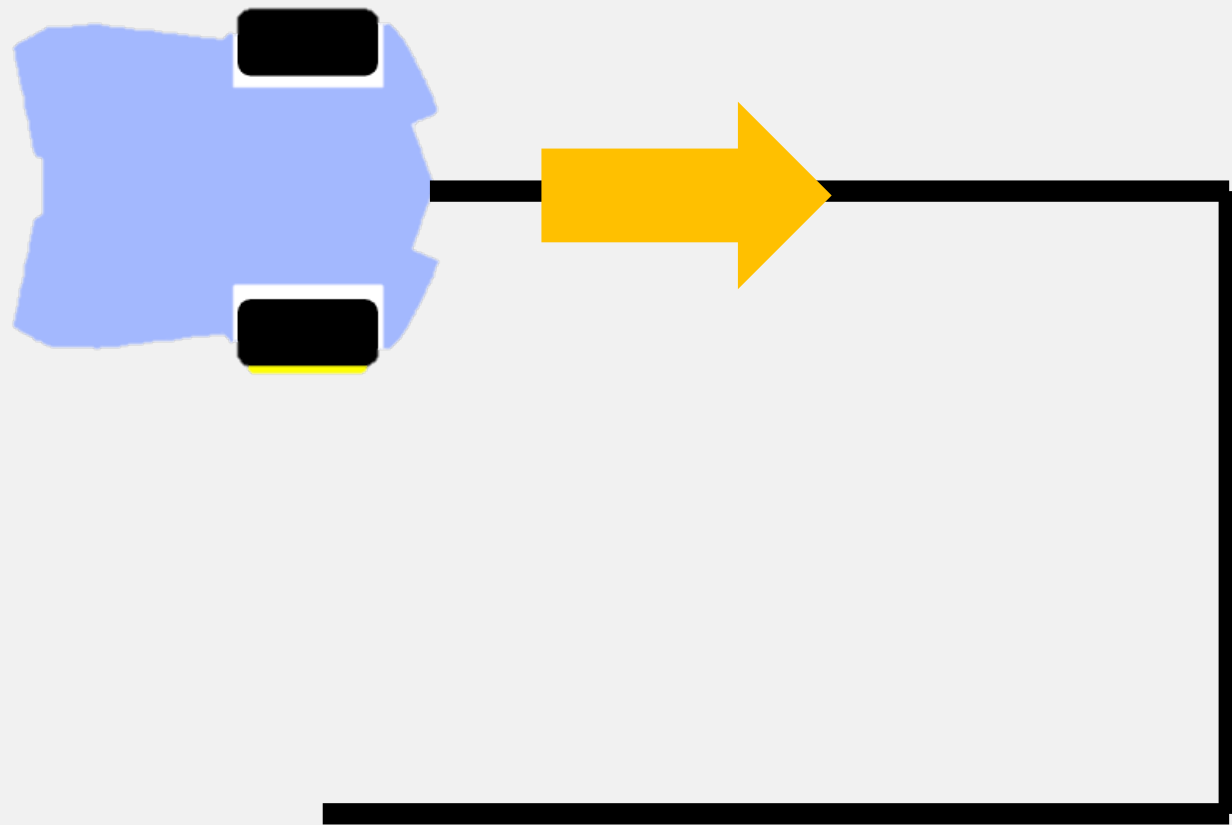
TT直流馬達

活動1：π字轉彎練習(國小)

提示：

試著調整秒數，

讓車輛可以精確地轉90度

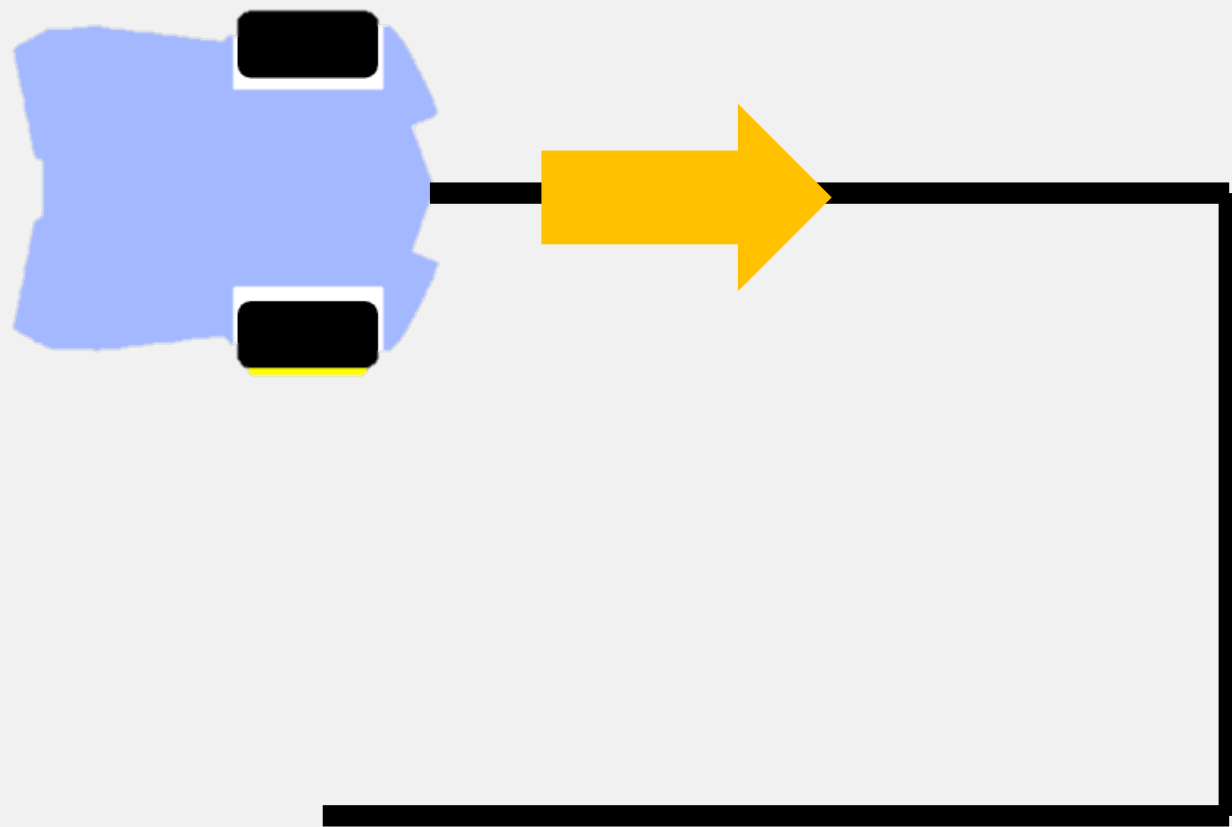


想想看：哪一種方式比較能夠完美地完成90度轉彎？

TT 直流馬達

活動1：π字轉彎練習

```
當啟動時  
  呼叫 forward  
  暫停 3000 毫秒  
  呼叫 turn_0  
  暫停 800 毫秒  
  呼叫 forward  
  暫停 3000 毫秒  
  呼叫 turn_0  
  暫停 800 毫秒  
  呼叫 forward  
  暫停 3000 毫秒  
  呼叫 stop
```

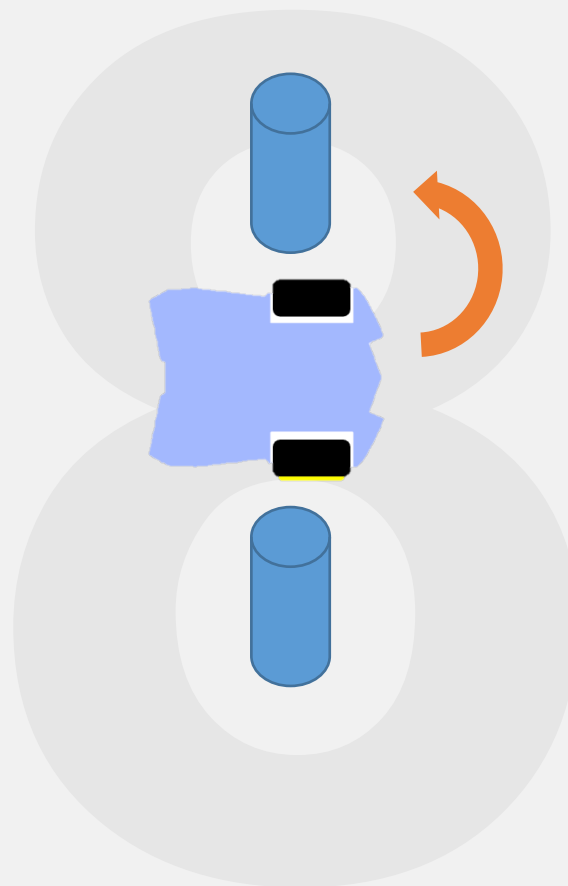


TT直流馬達

活動2：繞8字形

挑戰：

利用差速轉彎在兩個罐子間繞8字

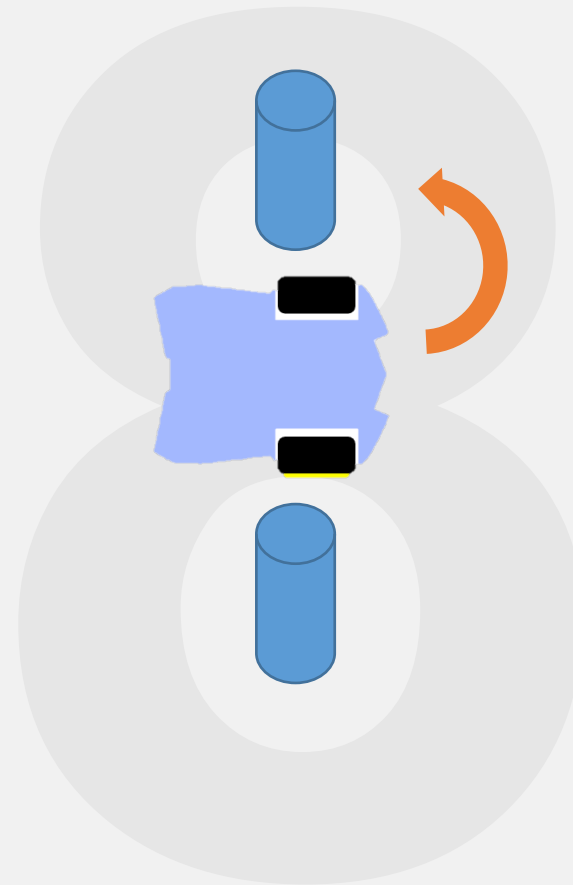


TT直流馬達

活動2：繞8字形

挑戰：

利用差速轉彎在兩個罐子間繞8字



單元六

紅外線感測操作

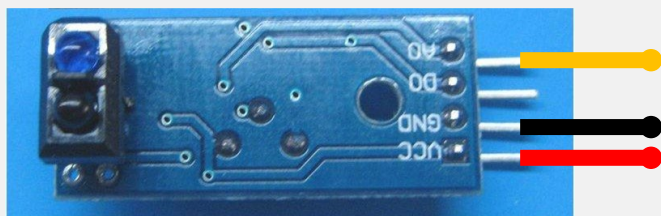
紅外線感測操作

功能介紹

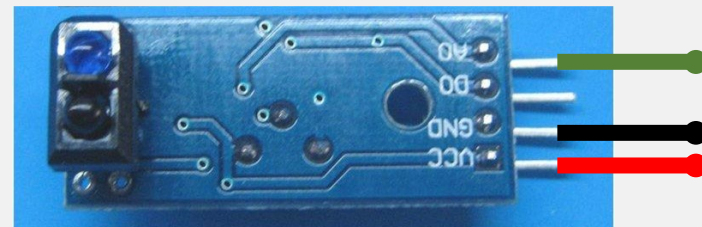
發射紅外線後，讀取反射回來的紅外線強度：

- ◆ 黑色區域：容易吸收光線，反射回來的光線較少。
- ◆ 白色區域：容易反射光線，反射回來的光線較多。

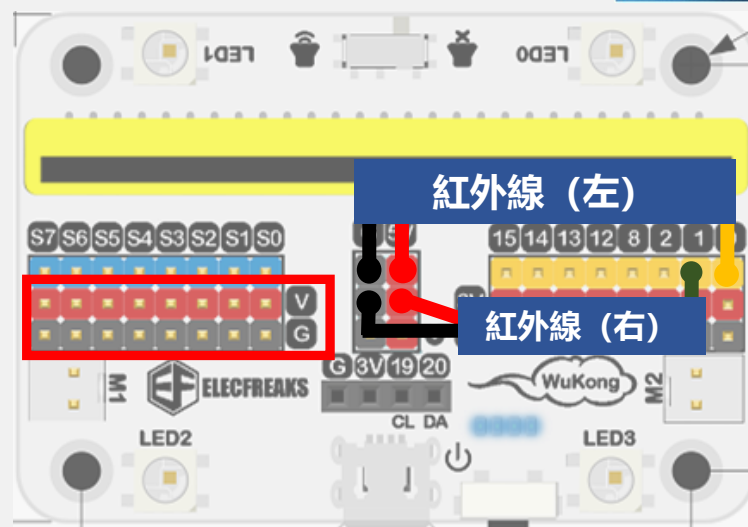
左側感測器接法 (0)



右側感測器接法(1)



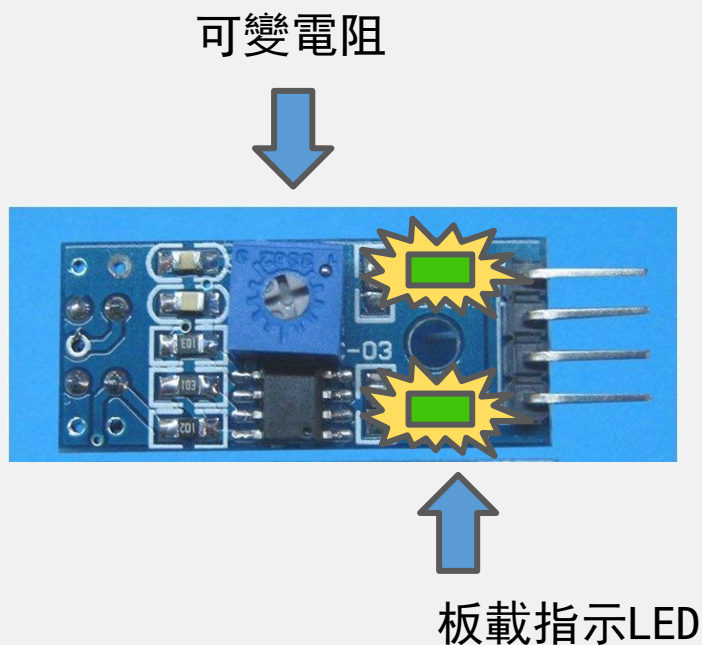
註：紅外線的紅線也可以接在左區->V，黑線->G



紅外線感測操作

補充：紅外線感測模組校正

- ◆紅外線感測模組容易受到場地照明、黑線材質等因素影響，每次使用一定要記得先做以下檢測：
- ◆使用類比模式時，可用程式讀取數值。
- ◆使用數位模式時，可用板載指示燈來觀察是否能判斷黑線，必要時可用板載的可變電阻調整校正數值。



請用手在感測器下方揮一揮
手在下方時：有光線反射，亮兩顆燈。
手放開時：無光線反射，亮一顆燈。

紅外線感測操作

讀取紅外線感測器數值



The image shows a Scratch script on a light blue grid background. The script consists of the following blocks:

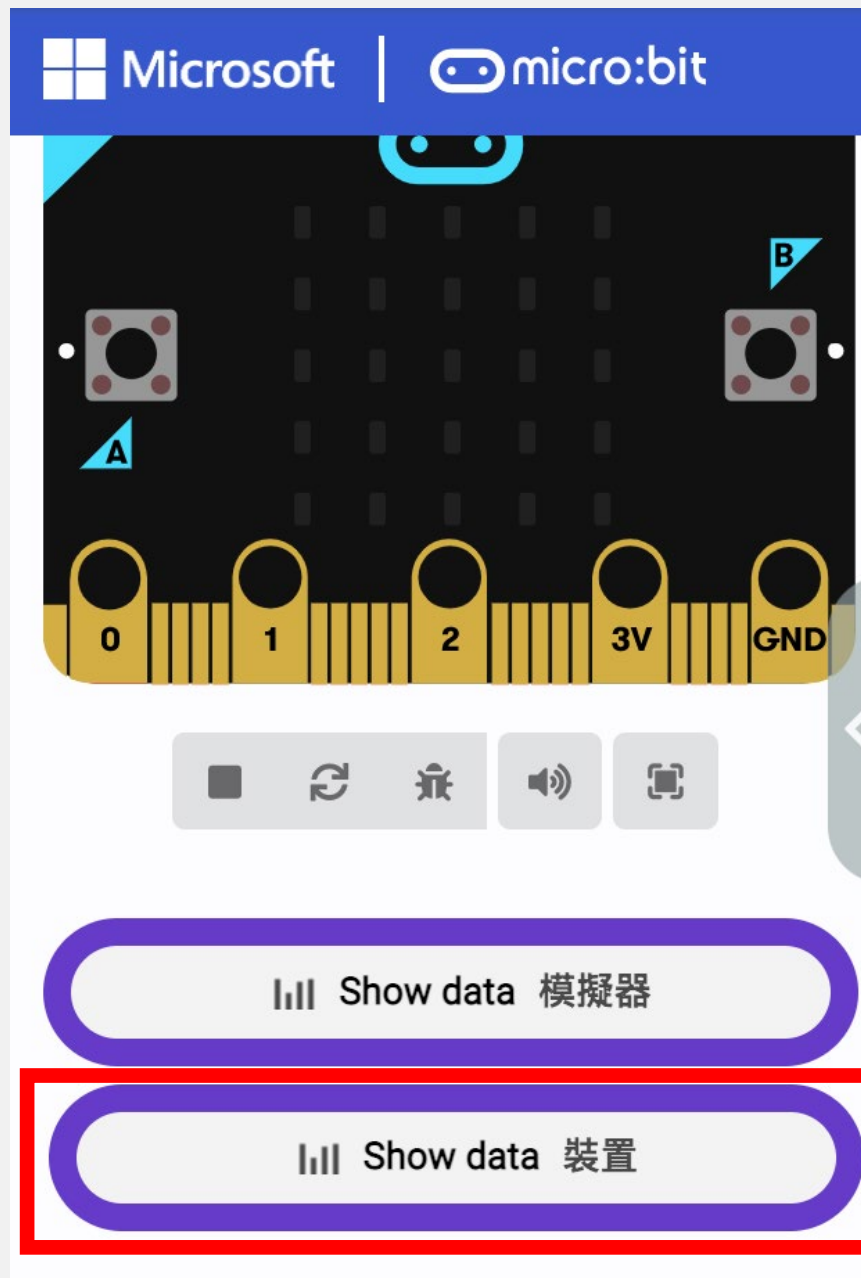
- 當啟動時** (When green flag clicked):
 - 序列 串列傳輸率設為 115200** (Serial: Set baud rate to 115200)
- 重複無限次** (Repeat forever):
 - 序列 寫入文字** (Serial: Write text) block with a **字串組合** (Text concatenation) block containing:
 - "left="** (Text)
 - 類比信號讀取 引腳 P0** (Analog sensor reading: Pin P0)
 - ","right="** (Text)
 - 類比信號讀取 引腳 P1** (Analog sensor reading: Pin P1)
 - 序列 寫入一行文字** (Serial: Write a line of text) block with a **" "** (Text) block.
 - 暫停 1000 毫秒** (Wait 1000 milliseconds)

紅外線感測操作

讀取紅外線感測器數值

上傳完程式後，
將micro:bit（USB線一樣接電腦）
插回擴充板，並打開擴充板的電源。

**打開Show data 裝置
監看紅外線數值**



紅外線感測操作

讀取紅外線感測器數值

Microsoft | micro:bit

積木

JavaScript



裝置



← 返回

當 裝置 發送資料時數值將被記錄

```
left=811,right=318  
left=336,right=316  
left=336,right=315  
left=57,right=51  
left=61,right=52  
left=58,right=50  
left=58,right=47  
left=60,right=51  
left=58,right=48  
left=57,right=47  
left=394,right=975  
left=403,right=992  
left=406,right=971  
left=827,right=1023  
left=823,right=1023
```

白底

黑線

makecode與Ardublockly不同

- 黑線為數值「大」
- 白底為數值「小」

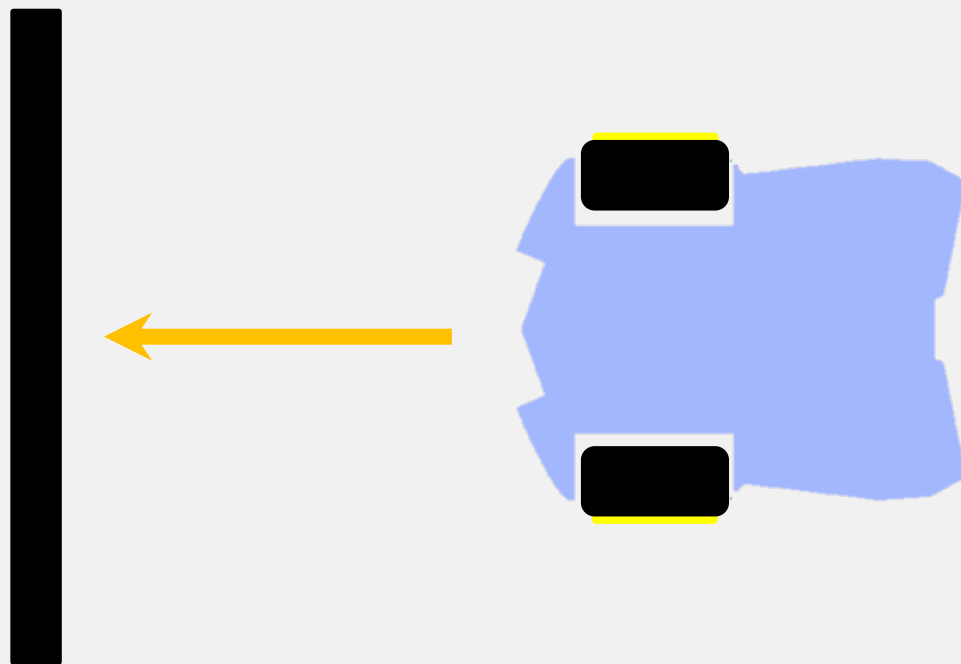
如有原使用Ardublockly的使用者要特別注意喔！

紅外線感測操作

前進遇到黑線停下

練習：

前進直到紅外線感測器偵測到黑線後停下



紅外線感測操作

前進遇到黑線停下



紅外線感測操作

活動3：計算黑線數量

從起點出發
請停在第三條黑線上

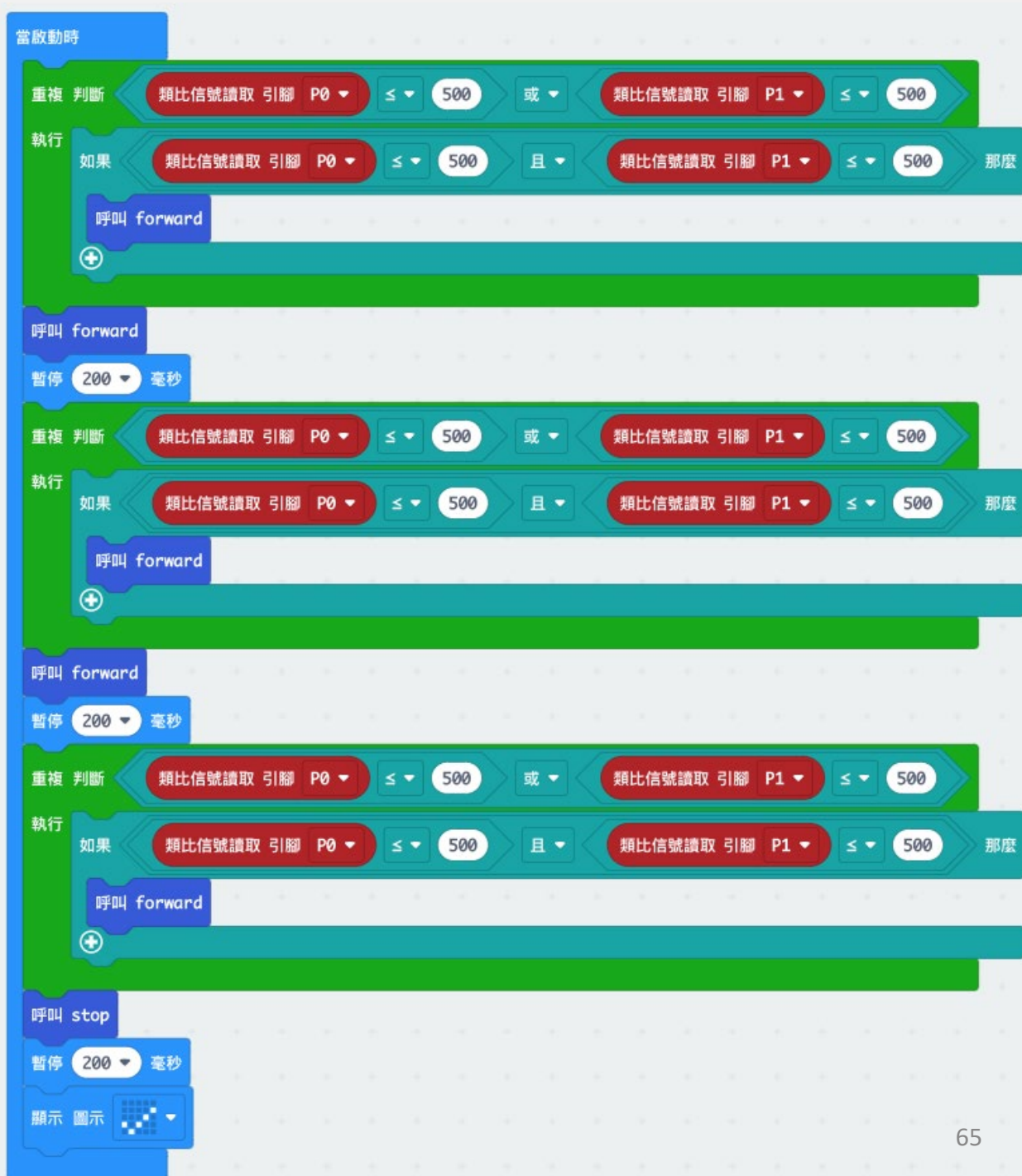
停在這一條



紅外線感測操作

活動3：計算黑線數量

需要視黑線間隔距離調整等待秒數，
若間距太小時容易失敗。



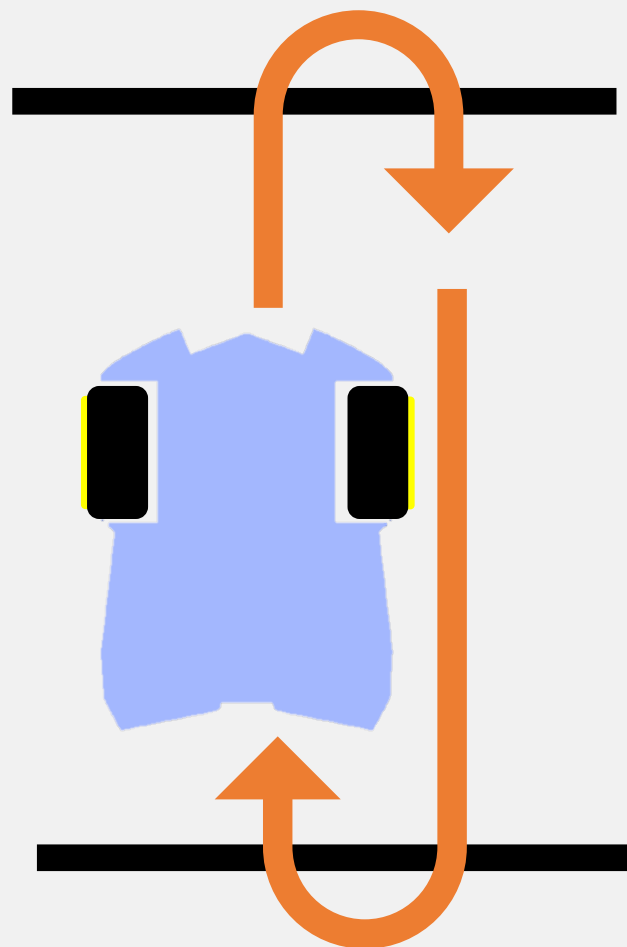
The image displays a Scratch script for an infrared sensor operation. The script is organized into four main sections, each starting with a '當啟動時' (When started) event block. Each section contains a '重複判斷' (Repeat while) block with two conditions: '類比信號讀取 引腳 P0 <= 500' and '類比信號讀取 引腳 P1 <= 500', connected by an '或' (or) operator. Below the repeat block is an '執行' (do) block with an '如果' (if) block containing the same two conditions connected by an '且' (and) operator. The '如果' block contains a '呼叫 forward' (call forward) block. After the '如果' block, there is a '暫停' (wait) block set to 200 milliseconds. The script concludes with a '呼叫 stop' (call stop) block and another '暫停' (wait) block set to 200 milliseconds. A '顯示 圖示' (show icon) block is also present at the end of the script.

紅外線感測操作

活動4：折返跑

練習：

在兩條黑線之間折返跑



紅外線感測操作

活動4：折返跑

練習：

在兩條黑線之間折返跑

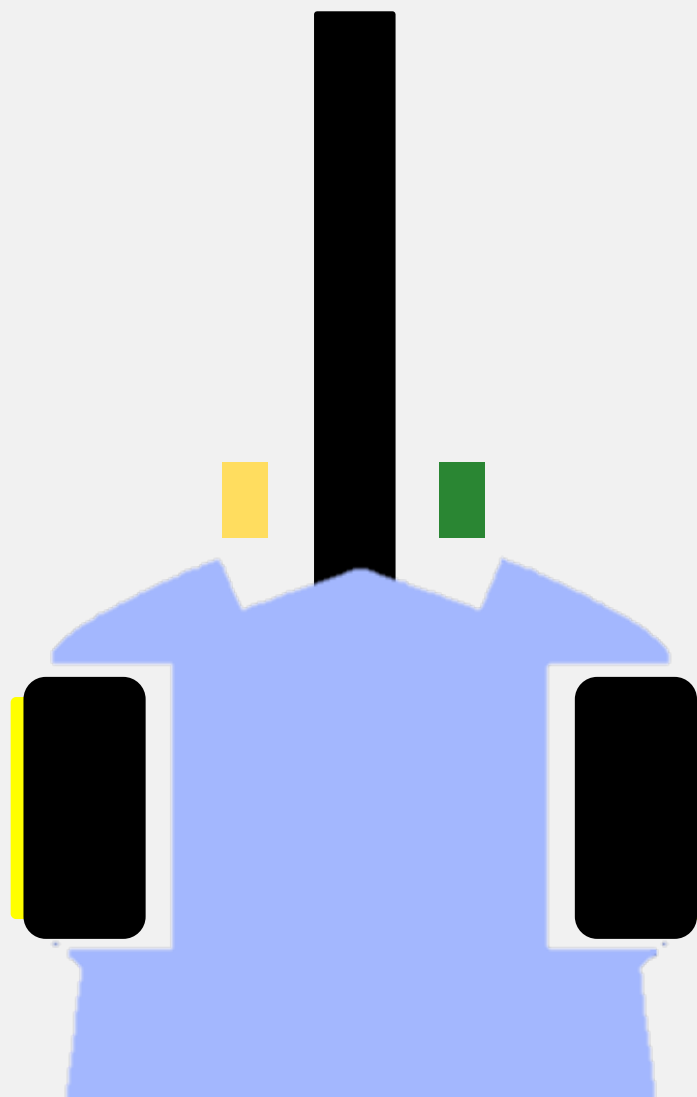


觀察旋轉結果調整等待秒數。

紅外線感測操作

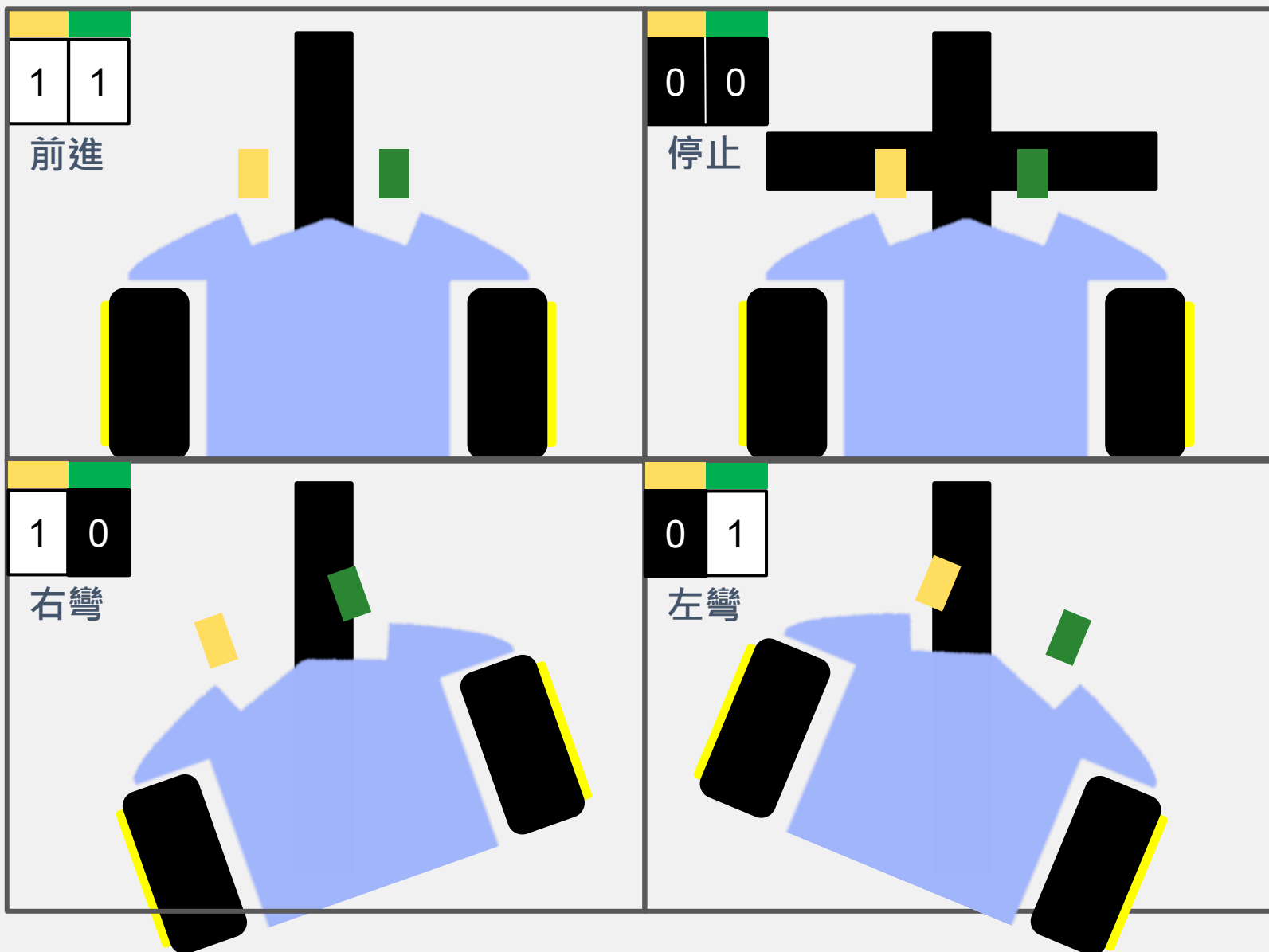
紅外線循跡模式

| | 黃色 | 綠色 |
|----|----|----|
| 前進 | 1 | 1 |
| 左彎 | 0 | 1 |
| 右彎 | 1 | 0 |
| 停止 | 0 | 0 |



紅外線感測操作

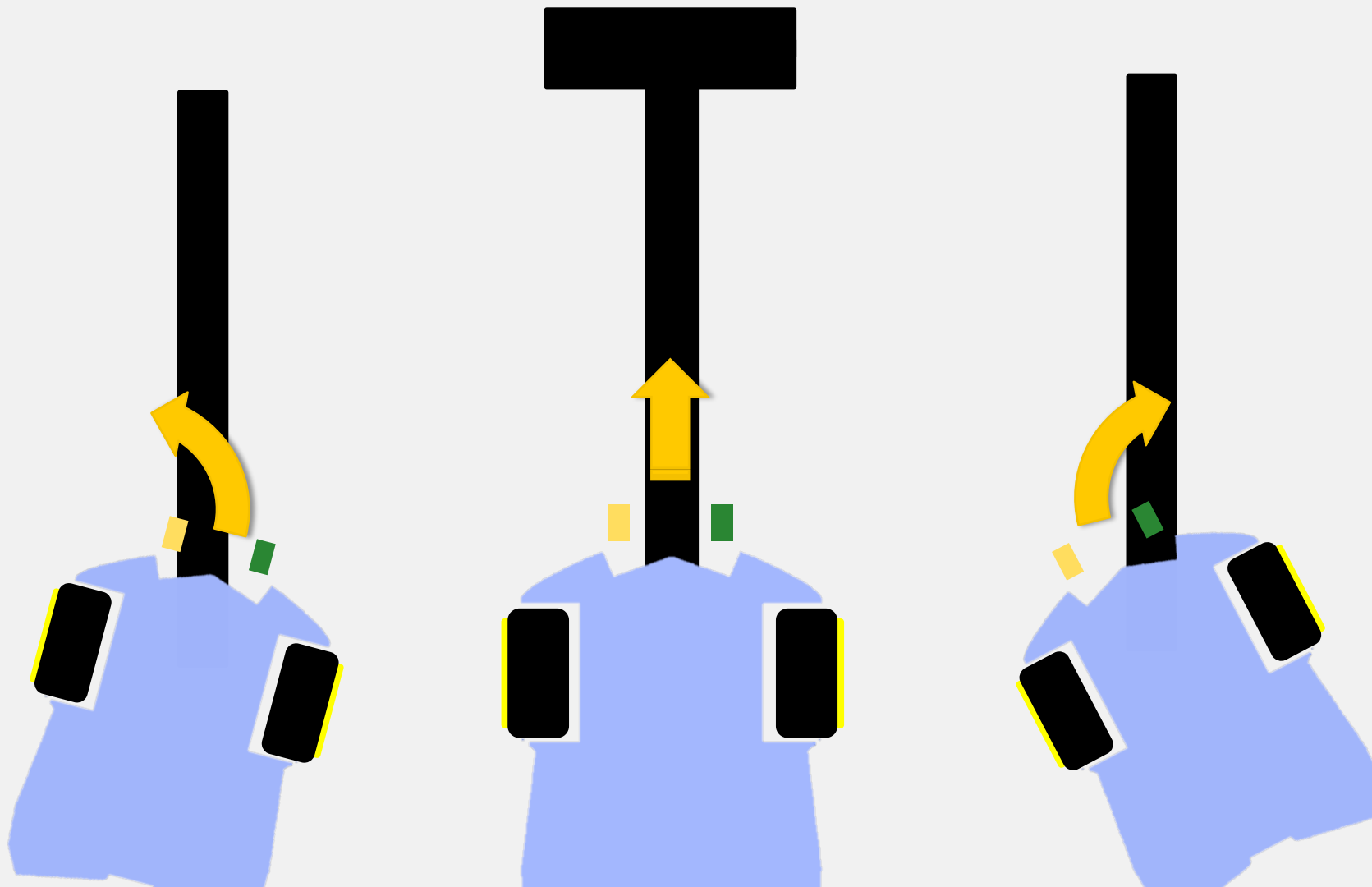
紅外線循跡模式



紅外線感測操作

紅外線循跡模式

口訣：哪邊壓線往哪邊轉



紅外線感測操作

紅外線循跡模式

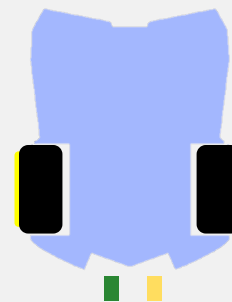
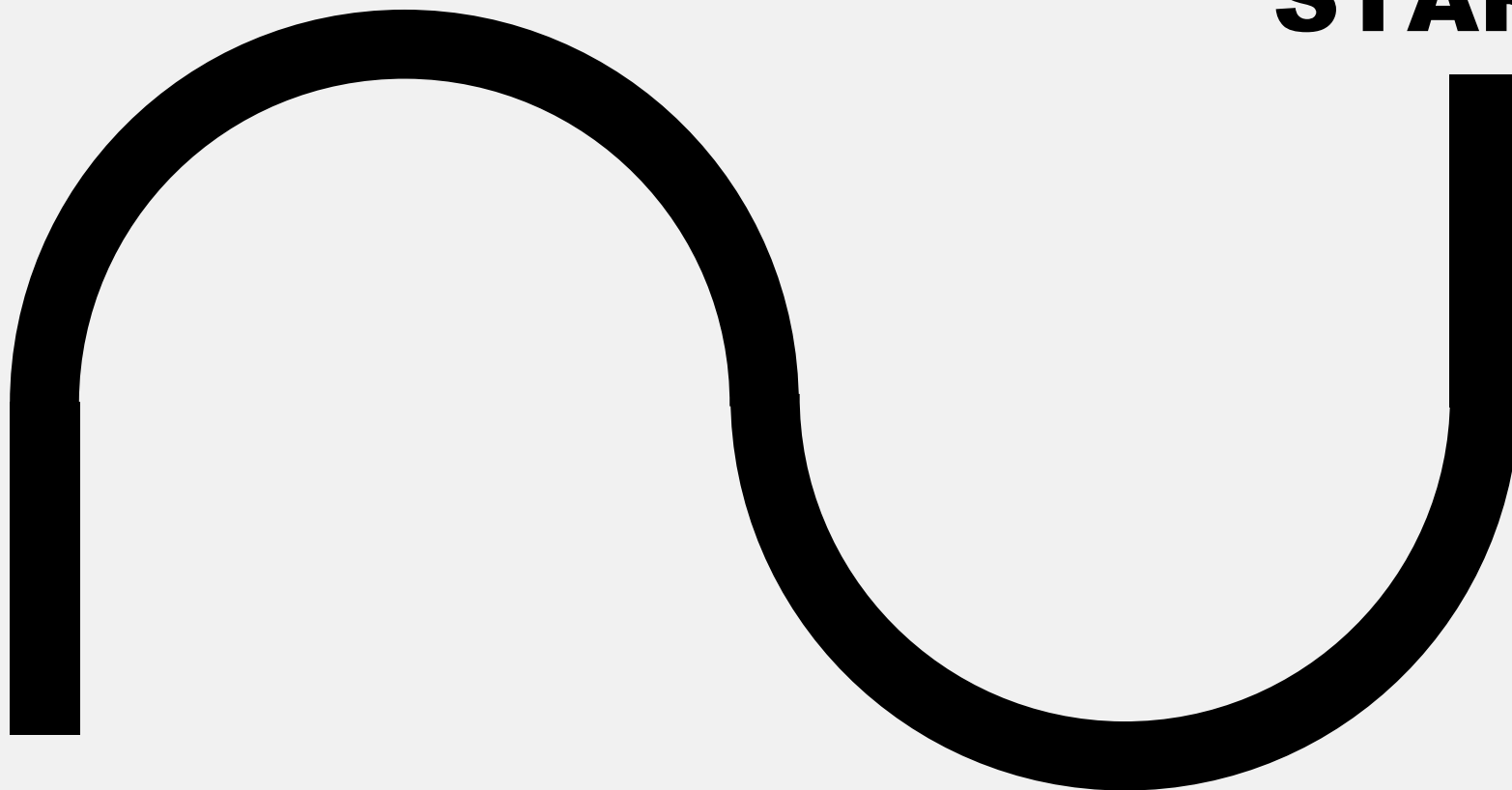
口訣：哪邊壓線往哪邊轉

The diagram shows a Scratch-style block diagram for an infrared line-following program. The program starts with a '當啟動時' (When started) block. It then enters a '重複判斷' (Repeat while) loop with the condition '類比信號讀取 引腳 P0 ≤ 500 或 類比信號讀取 引腳 P1 ≤ 500'. Inside the loop, there are three '執行' (Do) blocks: 1. '如果 類比信號讀取 引腳 P0 ≤ 500 且 類比信號讀取 引腳 P1 ≤ 500 那麼' (If both sensors are below 500, then) followed by a '呼叫 forward' block. 2. '否則如果 類比信號讀取 引腳 P0 ≥ 500 且 類比信號讀取 引腳 P1 ≤ 500 那麼' (Else if the left sensor is above 500 and the right is below 500, then) followed by a '呼叫 turn_left' block. 3. '否則如果 類比信號讀取 引腳 P0 ≤ 500 且 類比信號讀取 引腳 P1 ≥ 500 那麼' (Else if the left sensor is below 500 and the right is above 500, then) followed by a '呼叫 turn_right' block. After the loop, there is a '呼叫 stop' block, a '暫停 1000 毫秒' (Pause 1000 ms) block, and a '顯示 圖示' (Show icon) block.

紅外線感測操作

紅外線循跡

任務：
在S型黑線上做循跡

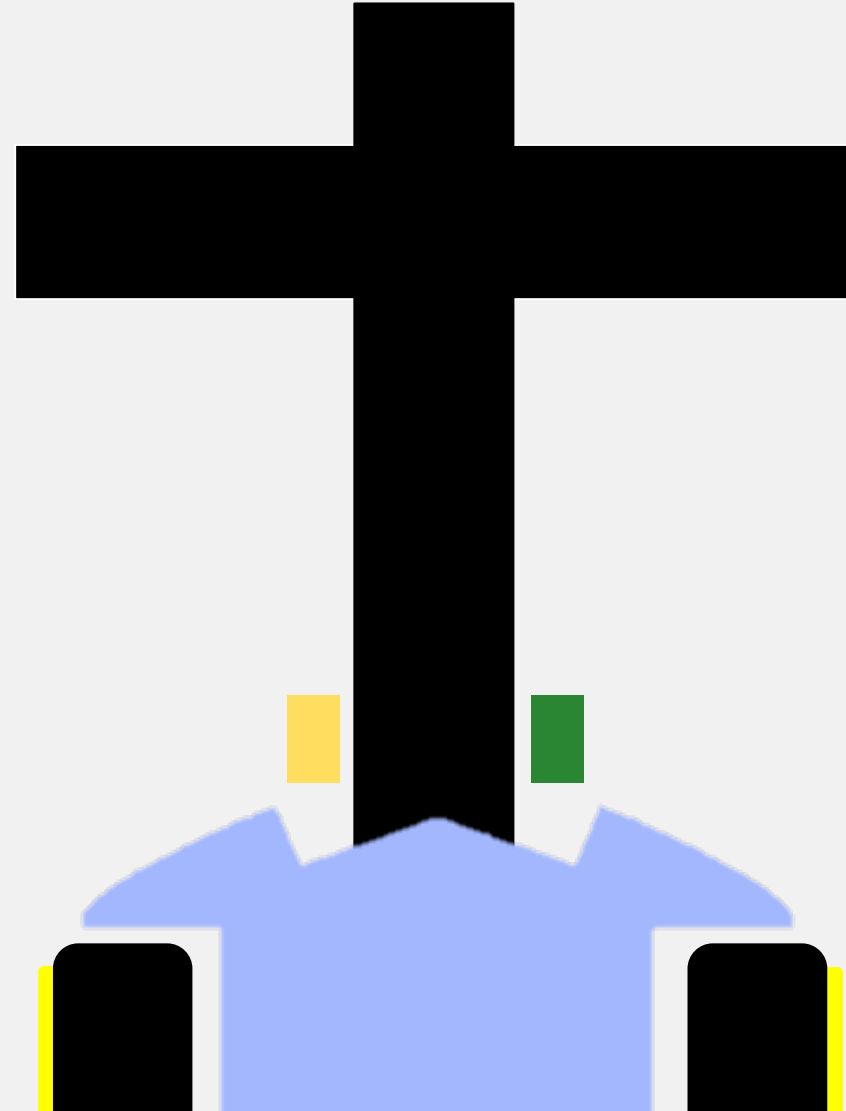


START

紅外線感測操作

活動5：循跡跨橫線

遇到橫線後
強迫走一段路

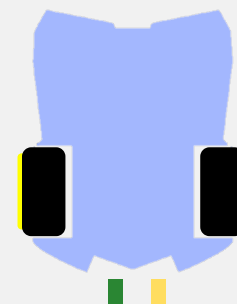


紅外線感測操作

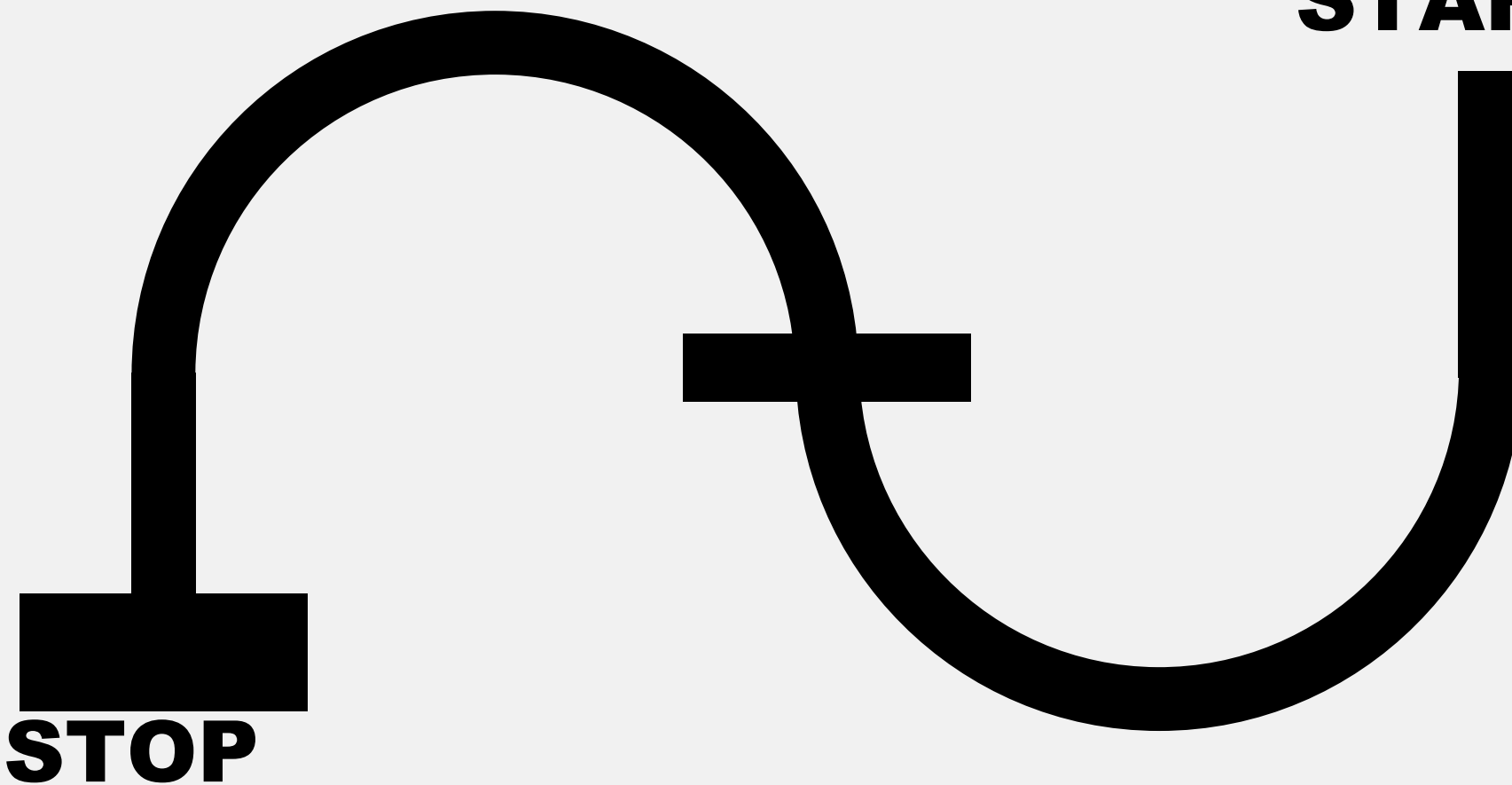
活動5：循跡跨橫線

練習：

在S型黑線上做循跡，必須跨越中段的橫線，並停在最終的寬橫線



START



紅外線感測操作

活動5：循跡跨橫線

當啟動時

重複 判斷

類比信號讀取 引腳 P0 ≤ 500 或 類比信號讀取 引腳 P1 ≤ 500

執行

如果 類比信號讀取 引腳 P0 ≤ 500 且 類比信號讀取 引腳 P1 ≤ 500 那麼

呼叫 forward

否則如果 類比信號讀取 引腳 P0 ≥ 500 且 類比信號讀取 引腳 P1 ≤ 500 那麼

呼叫 turn_left

否則如果 類比信號讀取 引腳 P0 ≤ 500 且 類比信號讀取 引腳 P1 ≥ 500 那麼

呼叫 turn_right

呼叫 forward

暫停 100 毫秒

呼叫 stop

暫停 500 毫秒

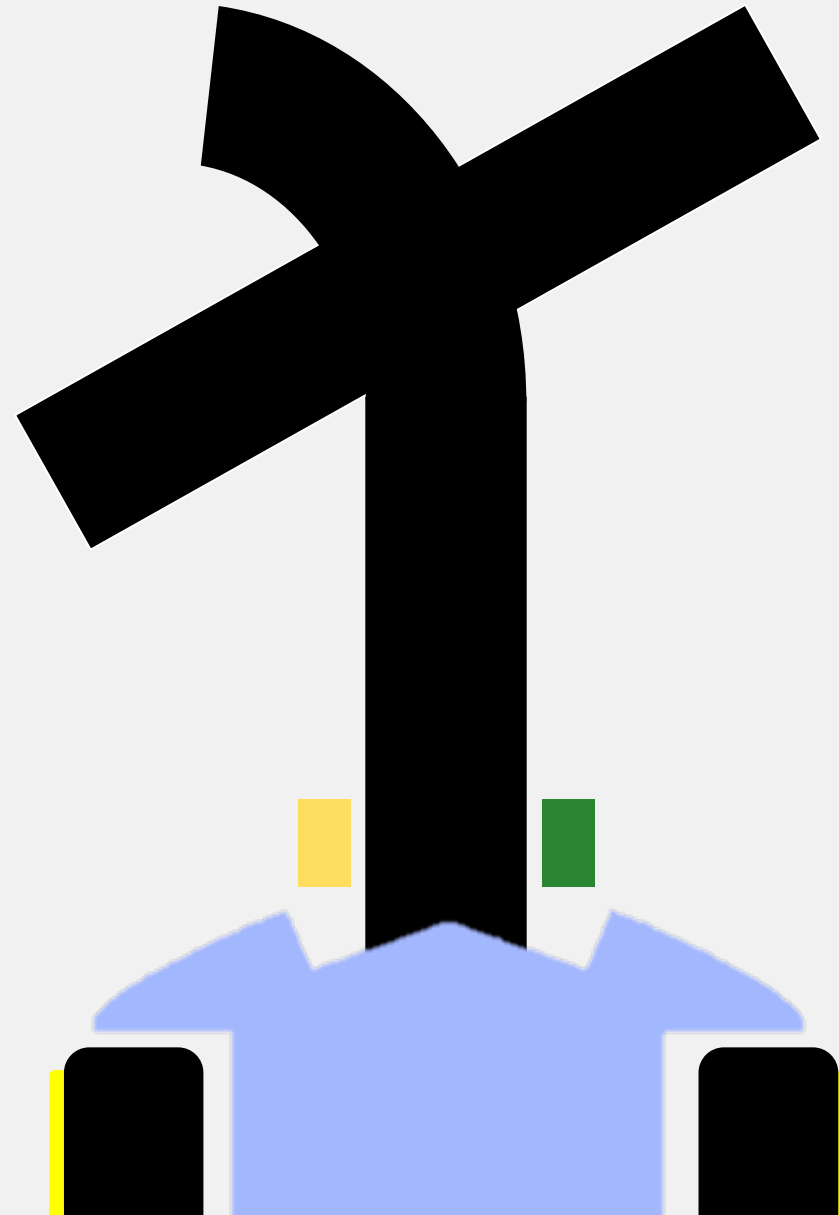
顯示 圖示

紅外線感測操作

特別注意：彎道中遇到橫線...



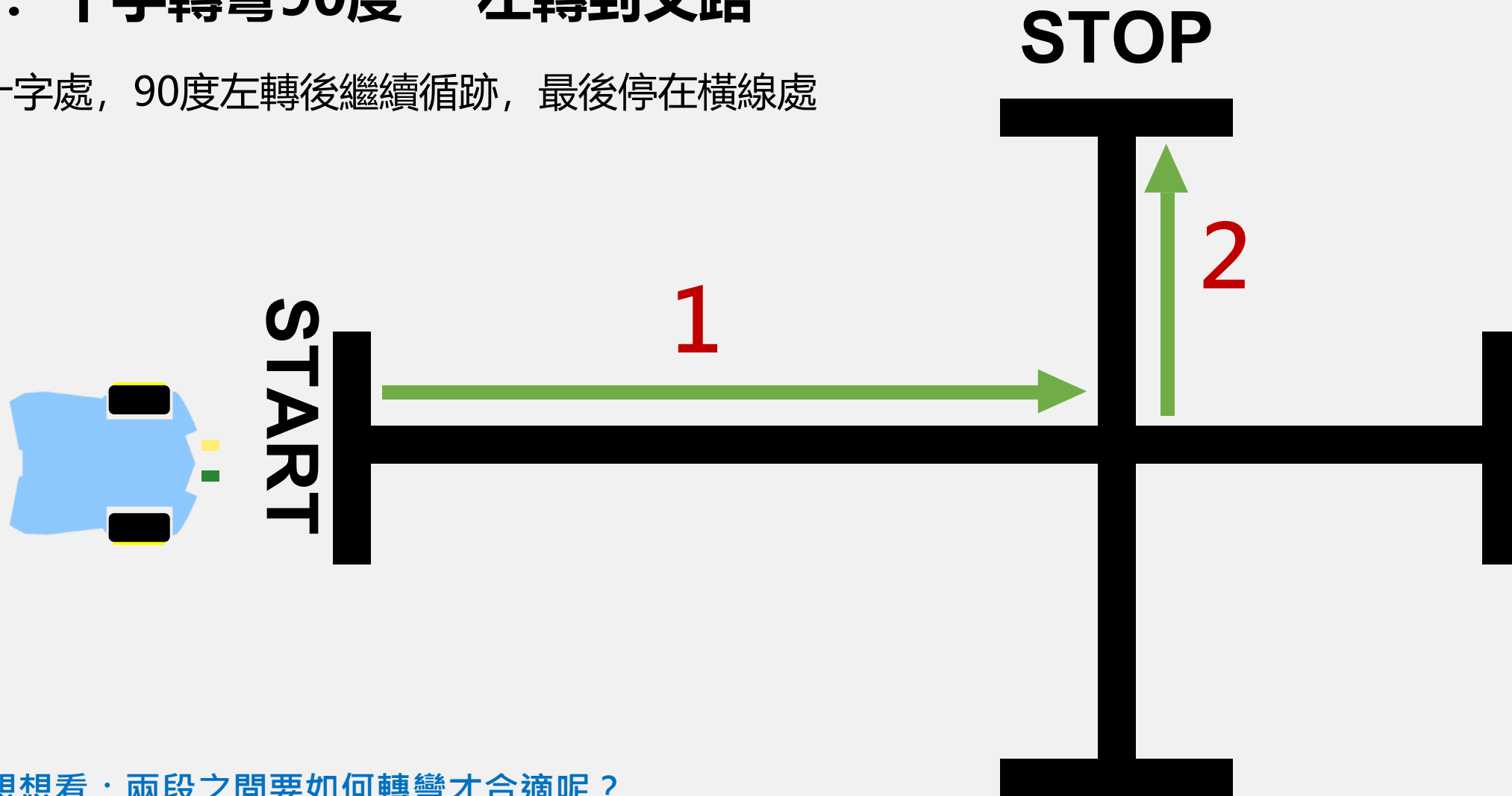
想想看：
仔細調整前進距離避免
感測器脫離黑線...



紅外線感測操作

活動6：十字轉彎90度 – 左轉到叉路

循跡到十字處，90度左轉後繼續循跡，最後停在橫線處



想想看：兩段之間要如何轉彎才合適呢？

紅外線感測操作

活動6：十字轉彎90度 – 左轉到叉路

1 想想看：為什麼這邊需要先**跨越**黑線呢？

2 想想看：為什麼這邊需要先**稍微偏擺**呢？

3 直到左邊的眼睛看到黑色前進行**左轉**

4 **轉彎**成功後循跡直到遇到黑線

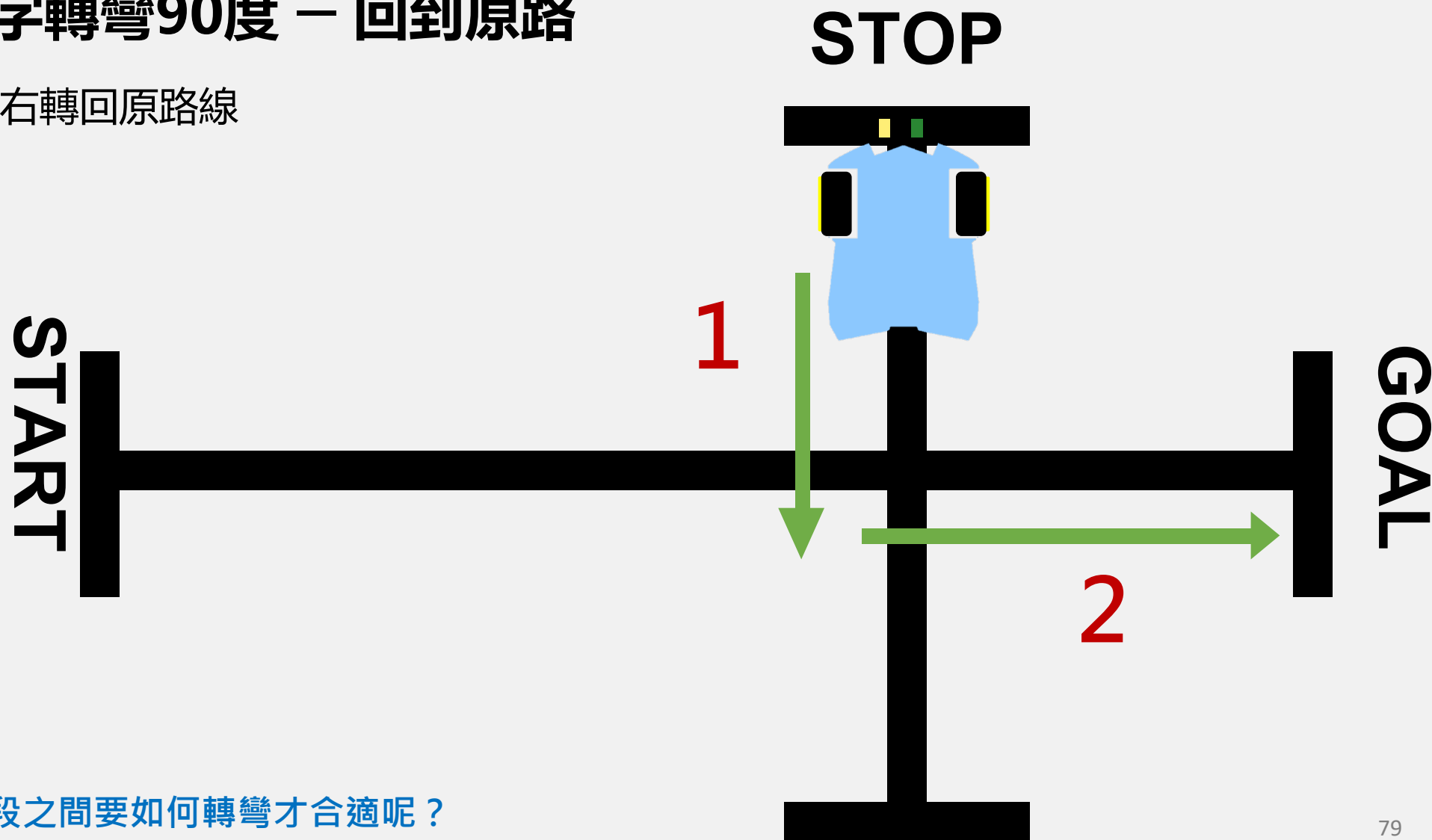
The diagram shows a sequence of blocks for a robot's movement:

- 當啟動時 (When started)
- 顯示 圖示 (Display icon) with a grid icon
- 呼叫 follow&stop (Call follow&stop)
- 呼叫 forward (Call forward)
- 暫停 300 毫秒 (Pause 300 ms)
- 呼叫 turn_left (Call turn_left)
- 暫停 300 毫秒 (Pause 300 ms)
- 重複 判斷 (Repeat loop):
 - 類比信號讀取 引腳 P0 (Analog signal read pin P0)
 - ≤ 500 (≤ 500)
 - 執行 呼叫 turn_left (Execute call turn_left)
- 呼叫 stop (Call stop)
- 暫停 200 毫秒 (Pause 200 ms)
- 呼叫 follow&stop (Call follow&stop)
- 暫停 300 毫秒 (Pause 300 ms)

紅外線感測操作

活動7：十字轉彎90度－回到原路

從橫線處退回，右轉回原路線



想想看：兩段之間要如何轉彎才合適呢？

紅外線感測操作

活動7：十字轉彎90度－回到原路

- 1 想想看：為什麼這邊需要先**後退**呢？
- 2 想想看：為什麼這邊需要先**稍微偏擺**呢？
- 3 直到左邊的眼睛看到黑色前進行**右轉**
- 4 **轉彎**成功後循跡直到遇到黑線



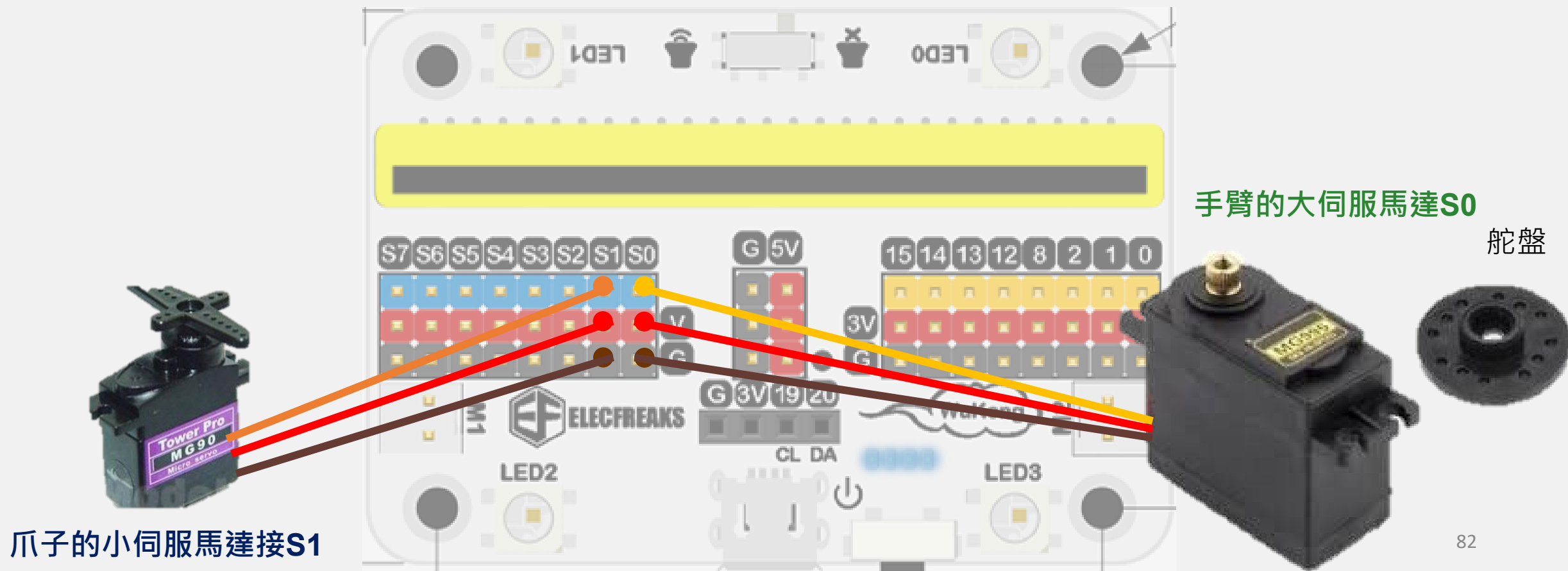
單元七

伺服馬達操作

伺服馬達操作

伺服馬達

- ◆ 精準地控制0度~180度
- ◆ 擴展板上使用S0及S1接腳連接伺服馬達



伺服馬達操作

伺服馬達角度校正

試著寫出右圖的程式，
並依以下步驟進行校正。

→設定伺服馬達角度為90度，

- 1.手臂應抬升至水平
- 2.爪子應張開至水平

若手臂角度不符預期

→拔除伺服馬達電源線

→將舵盤拔下

→調整至正確位置

→接上舵盤

當啟動時

设置 180° 舵机 S0 角度为 90

暫停 1000 毫秒

设置 180° 舵机 S1 角度为 90

暫停 1000 毫秒

※記得預留轉動時間，
一度需6毫秒以上

伺服馬達操作

控制伺服馬達角度

試著調整伺服馬達角度
找出手臂舉起、放下、爪子打開、收合時的角度為何？

提示：

看著伺服馬達軸心時，

角度增加 > 逆時針轉

角度減少 > 順時針轉



| 馬達 | S1 伺服馬達 | | S0 伺服馬達 | |
|----|---------|--------|---------|------|
| 動作 | 打開爪子 | 收合爪子 | 舉起手臂 | 放下手臂 |
| 角度 | 90 | 0 ~ 30 | 90 | 120 |

爪子收合時記得考慮物體大小，不要夾到底。

伺服馬達操作

活動9： 伺服馬達練習

口訣: **Down&Open**、**Close&Up**

預設角度

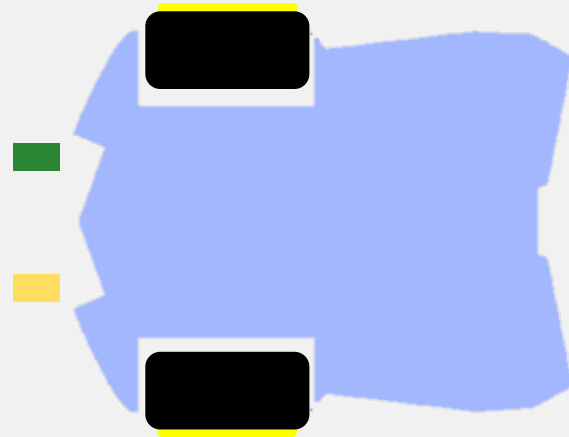
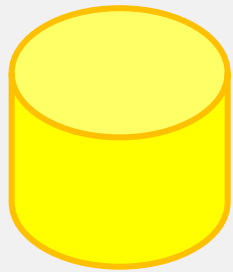
- > 放下手臂
- > 打開爪子
- > 收起爪子
- > 抬起手臂

The image shows two Scratch code blocks for controlling a servo motor. The first block is labeled "當啟動時" (When started) and contains four orange blocks: "设置 180° 舵机 S0 角度为 90" (Set servo S0 to 90 degrees), "暫停 1000 毫秒" (Pause 1000 ms), "设置 180° 舵机 S1 角度为 90" (Set servo S1 to 90 degrees), and "暫停 1000 毫秒" (Pause 1000 ms). The second block is labeled "重複無限次" (Repeat indefinitely) and contains six orange blocks: "设置 180° 舵机 S0 角度为 120" (Set servo S0 to 120 degrees), "暫停 500 毫秒" (Pause 500 ms), "设置 180° 舵机 S1 角度为 90" (Set servo S1 to 90 degrees), "暫停 500 毫秒" (Pause 500 ms), "设置 180° 舵机 S1 角度为 30" (Set servo S1 to 30 degrees), "暫停 500 毫秒" (Pause 500 ms), "设置 180° 舵机 S0 角度为 90" (Set servo S0 to 90 degrees), and "暫停 500 毫秒" (Pause 500 ms).

伺服馬達操作

延伸練習

在白色區域直走，遇到橫線停下來，並將目標物舉起再放下



伺服馬達操作

延伸練習

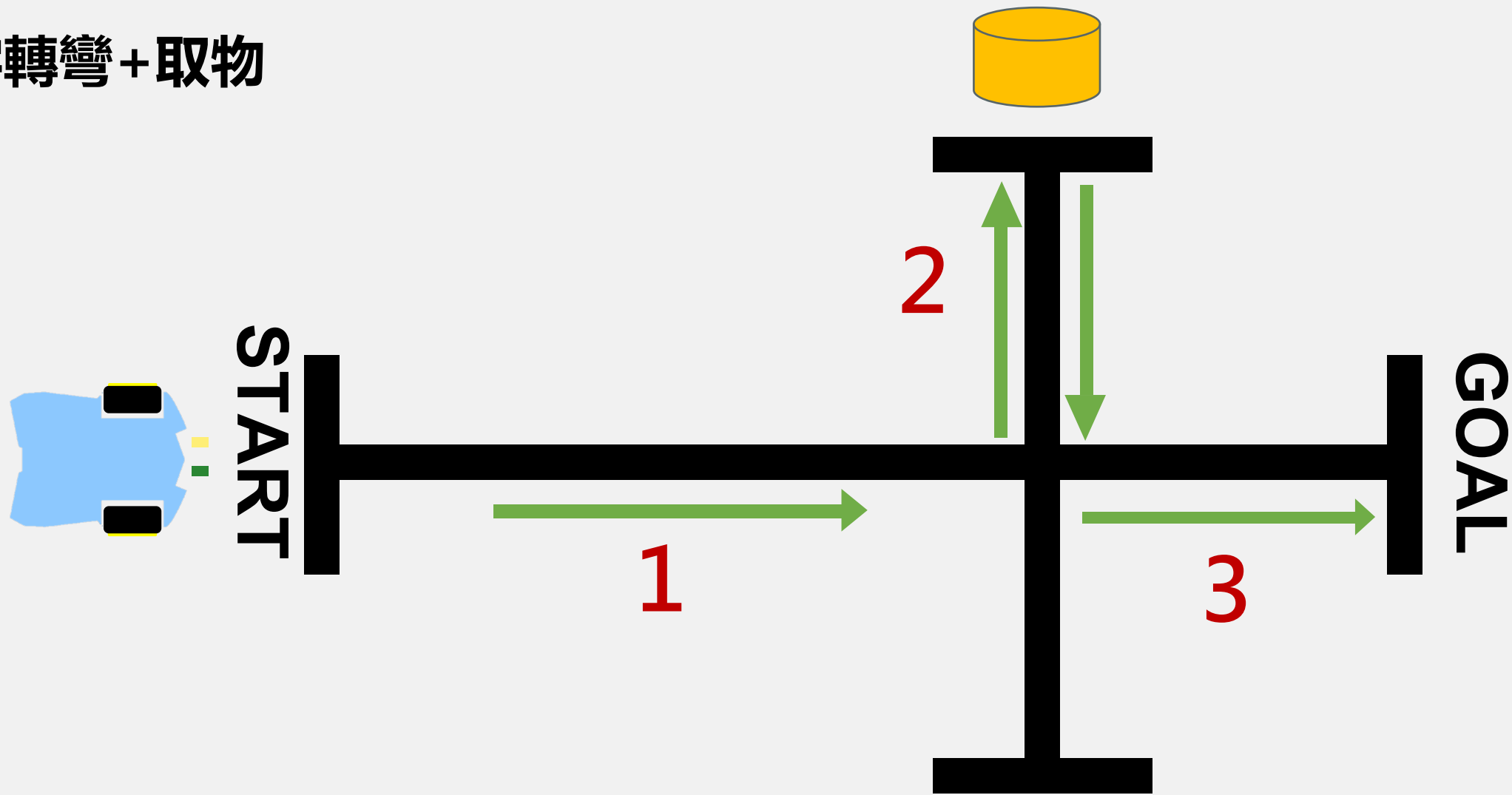
在白色區域直走，遇到橫線停下來，並將目標物舉起再放下。

The diagram shows a sequence of Scratch-like blocks for controlling a servo motor. It starts with a '當啟動時' (When started) block, followed by a '顯示 圖示' (Show icon) block. The sequence then consists of several '设置' (Set) and '暫停' (Pause) blocks for two servos, S0 and S1, with specific angles and 1000ms delays. A '重複 判斷' (Repeat while) loop follows, containing an '如果' (If) block with two conditions: '類比信號讀取 引腳 P0 ≤ 500' and '類比信號讀取 引腳 P1 ≤ 500'. Inside the '如果' block is a '呼叫 forward' (Call forward) block. After the loop, there is a '呼叫 stop' (Call stop) block, followed by more '设置' and '暫停' blocks for servos S1 and S0 with different angles and 1000ms delays. The sequence ends with another '顯示 圖示' (Show icon) block.

一開始就讓爪子打開並放在最底下的角度，在取物時可以避免爪子在運作過程中被目標物干擾。

伺服馬達操作

十字轉彎+取物



想想看：兩段之間要如何轉彎才合適呢？

伺服馬達操作

十字轉彎+取物

```
當啟動時  
顯示 圖示  
呼叫 down&open  
暫停 300 毫秒  
呼叫 follow&stop  
呼叫 forward  
暫停 300 毫秒  
呼叫 X-left  
呼叫 follow&stop  
暫停 1000 毫秒  
呼叫 close&up  
暫停 300 毫秒  
呼叫 back  
暫停 300 毫秒  
呼叫 X-right  
呼叫 follow&stop
```

放下手臂、循跡到十字

左轉到叉路

循跡到橫線

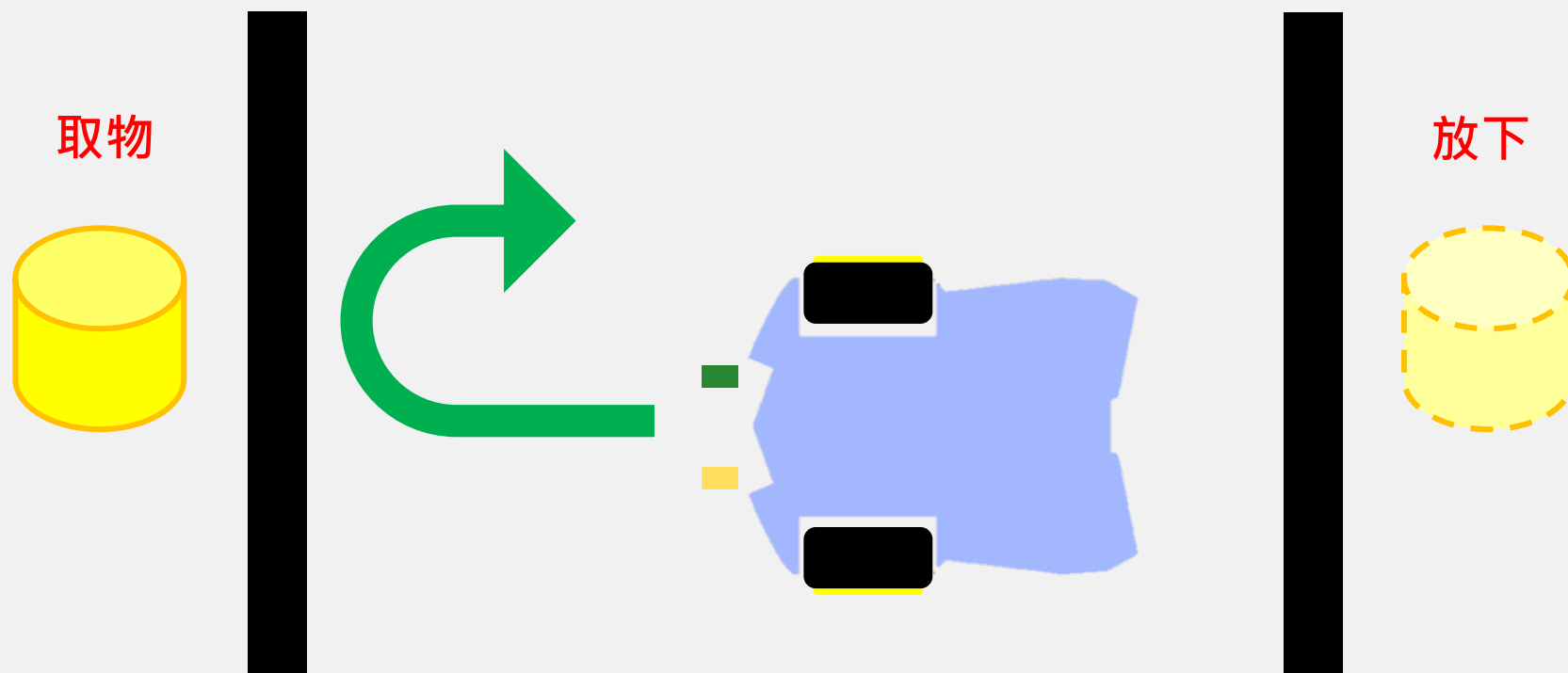
抓起道具

回到原路、循跡至下一關

伺服馬達操作

練習

在白色區域直走，遇到橫線停下來，並將目標物舉起，迴轉後繼續前進，遇到黑線後停下來，並將目標物放下。(提示：兩條黑線間折返跑)



伺服馬達操作

練習

在白色區域直走，遇到橫線停下來，並將目標物舉起，迴轉後繼續前進，遇到黑線後停下來，並將目標物放下。
(提示：兩條黑線間折返跑)

當啟動時

顯示 圖示

呼叫 down&open

可將爪子的一系列動作寫為副程式

重複 判斷 類比信號讀取 引腳 P0 ≤ 500 或 類比信號讀取 引腳 P1 ≤ 500

執行 如果 類比信號讀取 引腳 P0 ≤ 500 且 類比信號讀取 引腳 P1 ≤ 500 那麼

呼叫 forward

呼叫 stop

暫停 1000 毫秒

呼叫 close&up

呼叫 turn_right

原地旋轉90度

暫停 800 毫秒

重複 判斷 類比信號讀取 引腳 P0 ≤ 500 或 類比信號讀取 引腳 P1 ≤ 500

執行 如果 類比信號讀取 引腳 P0 ≤ 500 且 類比信號讀取 引腳 P1 ≤ 500 那麼

呼叫 forward

呼叫 stop

暫停 1000 毫秒

呼叫 down&open

定義函式 down&open

- 設置 180° 舵機 S0 角度为 120
- 暫停 1000 毫秒
- 設置 180° 舵機 S1 角度为 90
- 暫停 1000 毫秒

定義函式 close&up

- 設置 180° 舵機 S1 角度为 30
- 暫停 1000 毫秒
- 設置 180° 舵機 S0 角度为 90
- 暫停 1000 毫秒

單元八

AI影像辨識

smart AI Lens 智能AI鏡頭套件介紹

什麼是smart AI Lens?



1. AI Lens能夠實現人臉識別、圖卡辨識、循線感應，可識別球體和特徵獲取等功能。
2. 螢幕具樂高可相容的孔洞，輕鬆連接樂高，發揮創意。
3. 90度超廣角鏡頭，可識別完整圖像。

smart AI Lens 智能AI鏡頭套件介紹

應用廣泛

圖卡辨識



循線感應



識別球體



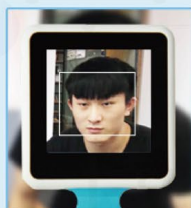
顏色辨識



一鍵學習



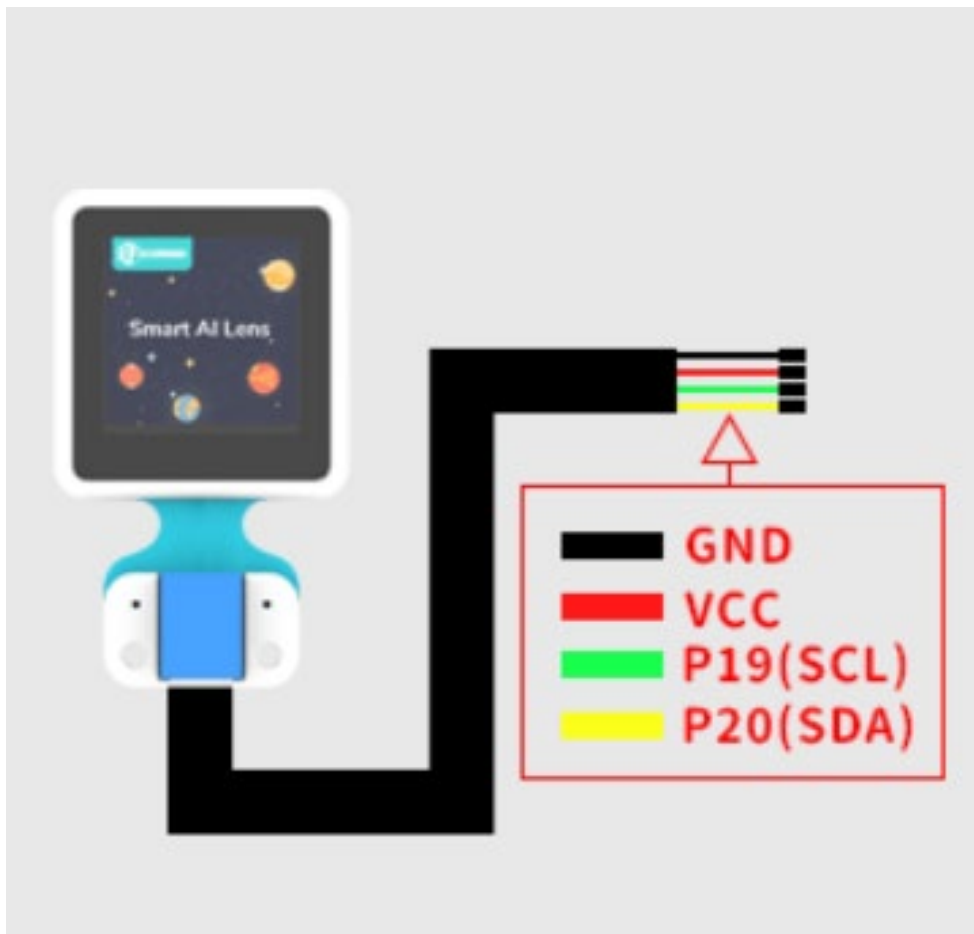
臉部辨識



六種功能，可於不同情況下進行應用。

1. 圖卡辨識
2. 循線感應
3. 識別球體
4. 顏色辨識
5. 一鍵學習
6. 臉部辨識

smart AI Lens 智能AI鏡頭套件介紹



連接方式：RJ11 到杜邦接線

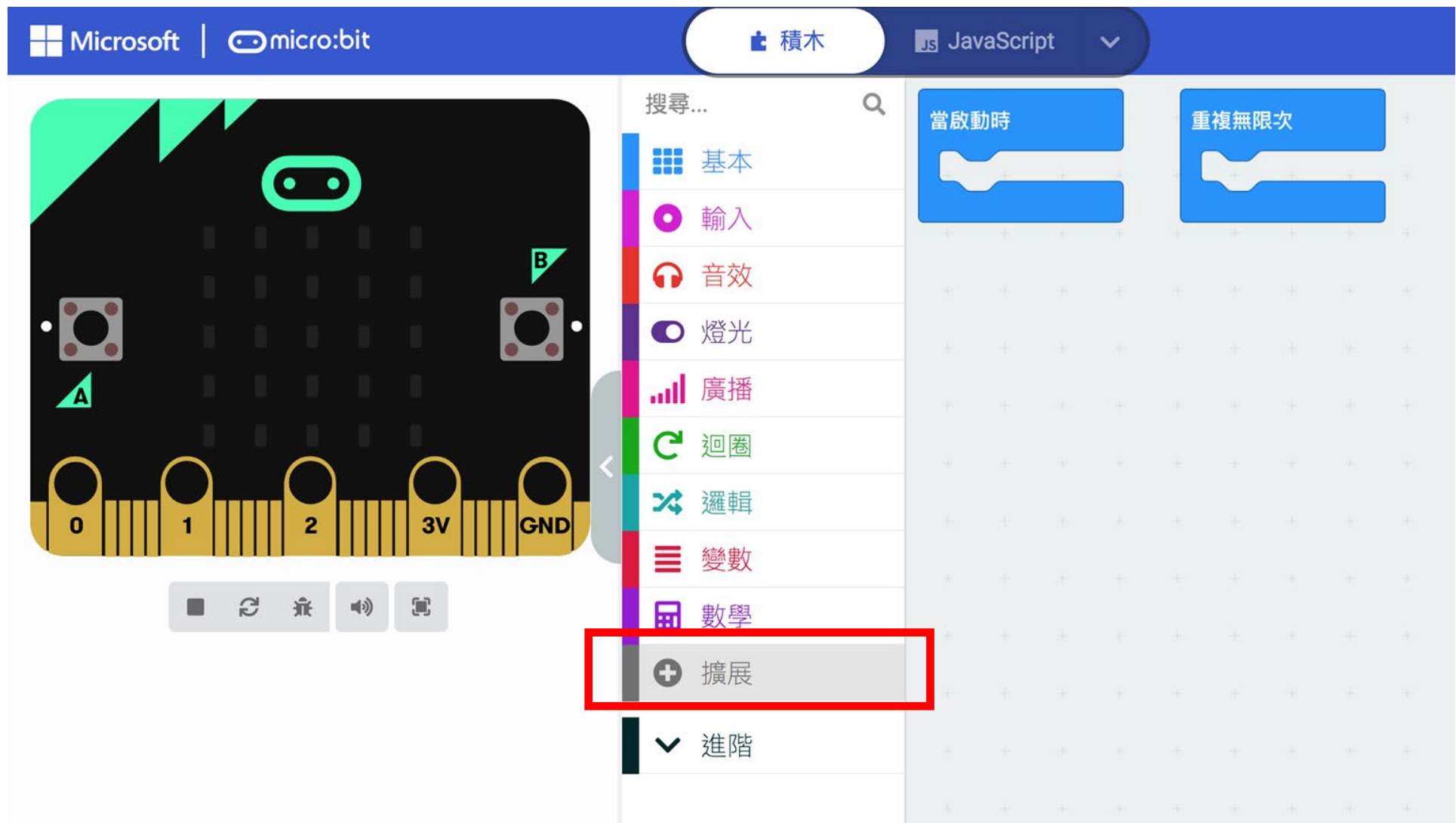
(原為杜邦母頭，需使用針腳轉接至悟空擴充板)

(黑線連接到GND；紅線連接到VCC；綠線連接到P19(SCL)；黃線連接到P20(SDA))

如果鏡頭無法正常運作，請嘗試使用獨立的電源為鏡頭供電。

程式環境設定

點擊擴展



程式環境設定

新增PlanetX-AI



The screenshot shows the Chrome Web Store interface. At the top, there is a blue header with the word "擴展" (Extensions). Below it is a search bar containing the text "PlanetX-AI", which is highlighted with a red box. To the right of the search bar is a magnifying glass icon. Below the search bar are several category buttons: "Lights and Display", "Software", "科學" (Science), "Robotics", "Gaming", and "Networking". In the center of the page, a search result for "PlanetX-AI" is displayed, also highlighted with a red box. The result includes the ELEC Freaks logo, an image of the PlanetX-AI sensor, and the text: "PlanetX-AI (行星X)The micro:bit new sensor series PlanetX(only AI) with RJ11 connection port by ELEC FREAKS...". Below the text is a "Learn More" link.

擴展

PlanetX-AI

1. 搜尋PlanetX-AI

Lights and Display Software 科學 Robotics Gaming Networking

2. 點擊PlanetX-AI

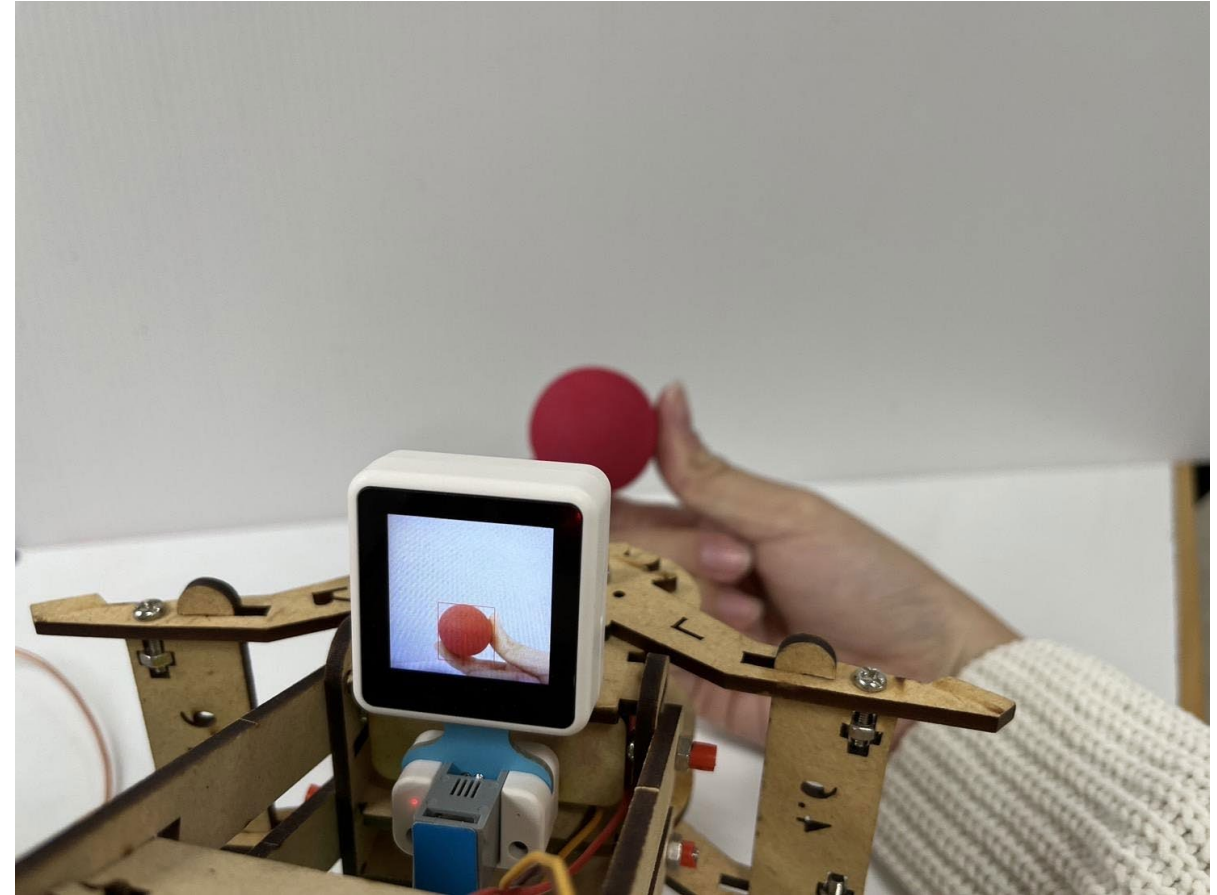
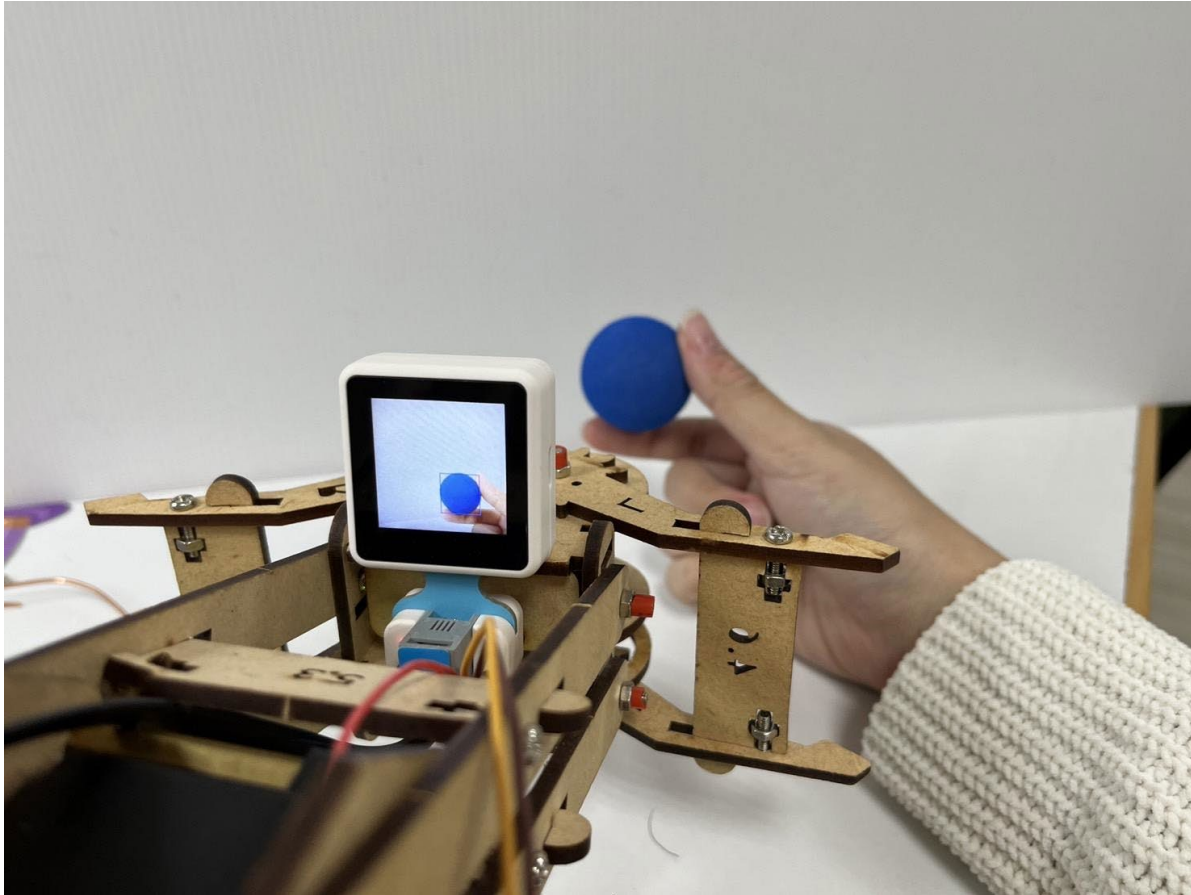
ELEC FREAKS



PlanetX-AI
(行星X)The micro:bit new sensor series PlanetX(only AI) with RJ11 connection port by ELEC FREAKS...

[Learn More](#)

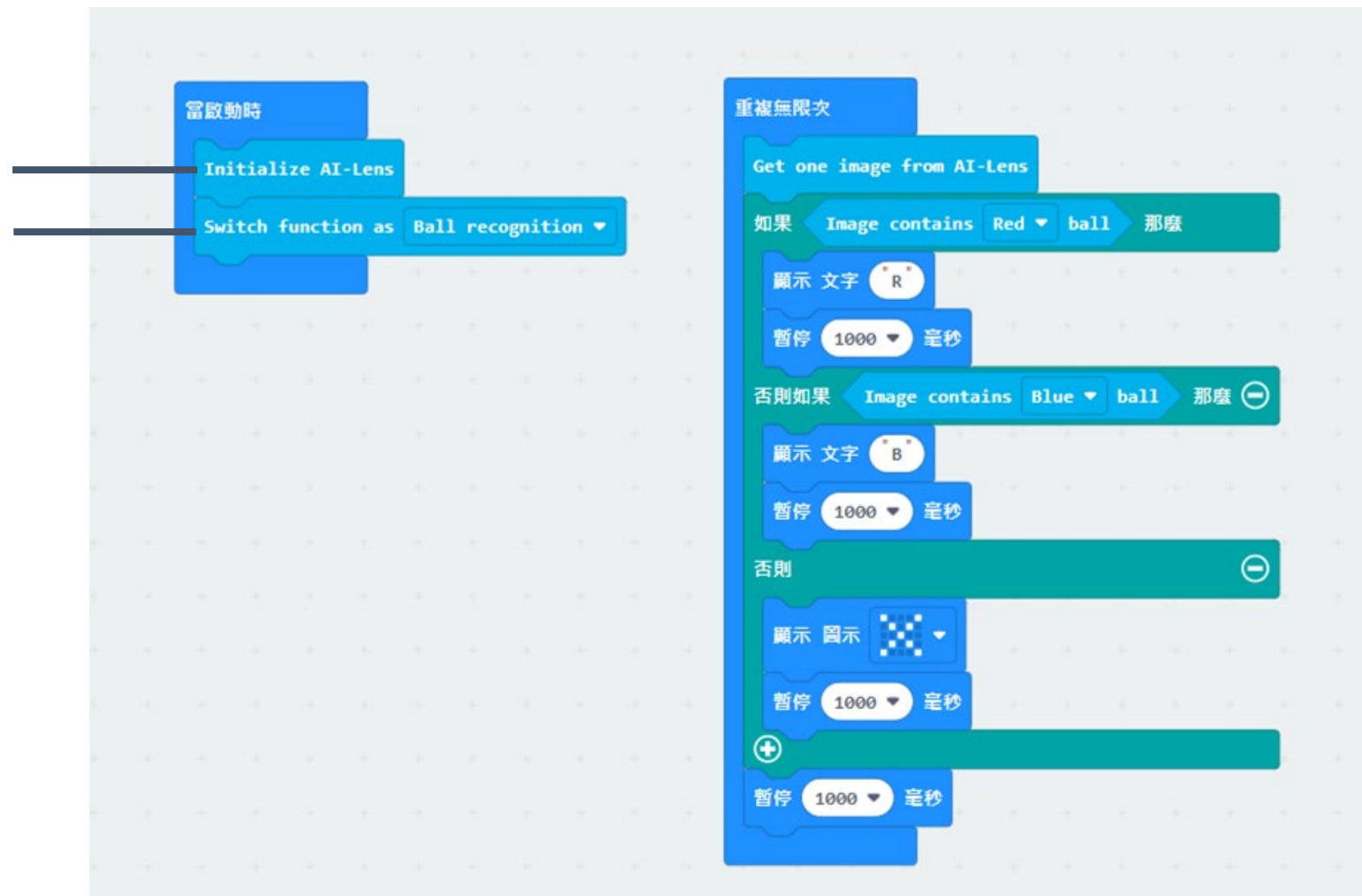
球體辨識



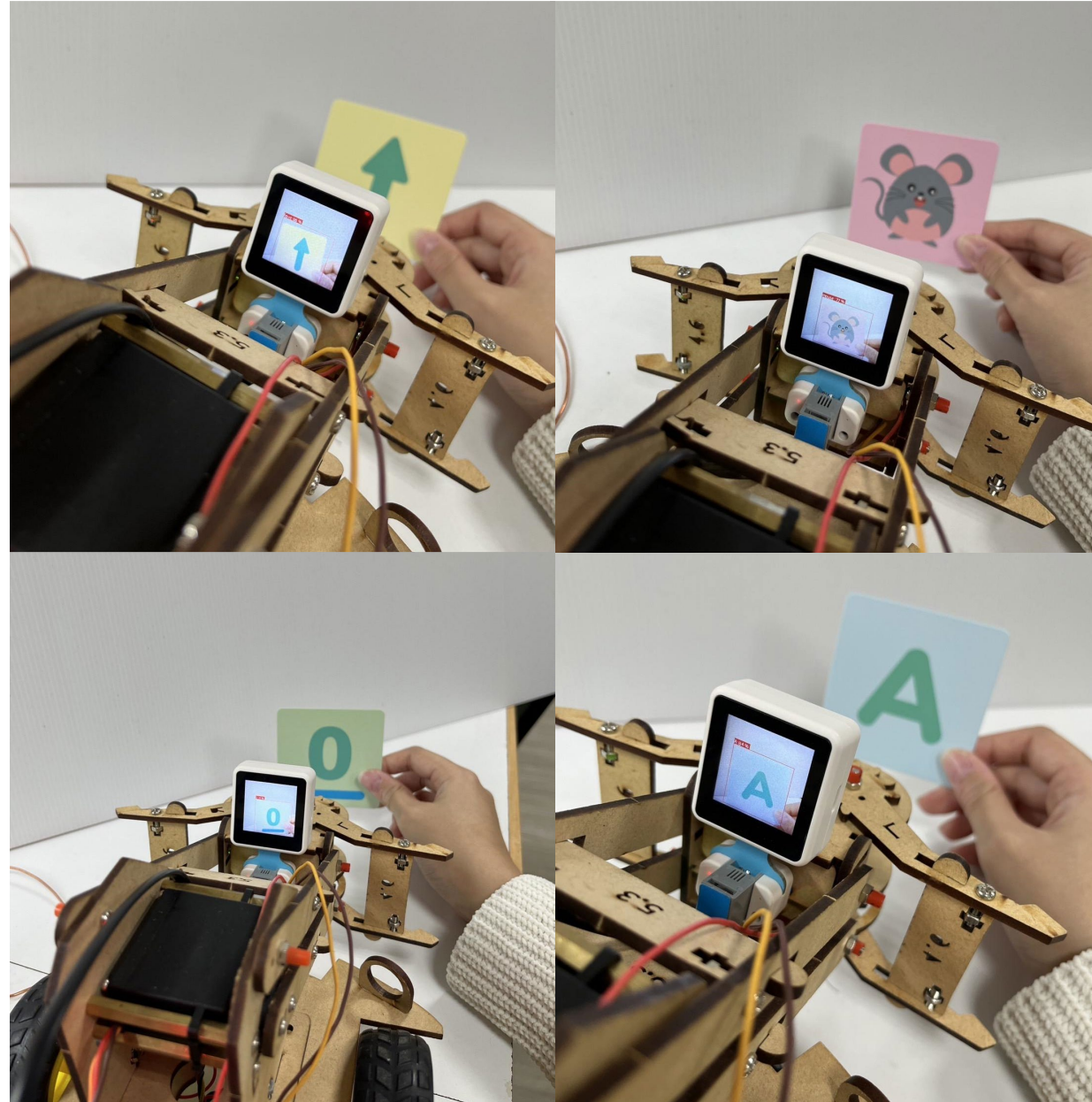
球體辨識

球體顏色僅能辨識「紅」、「藍」

初始化AI鏡頭 設定功能



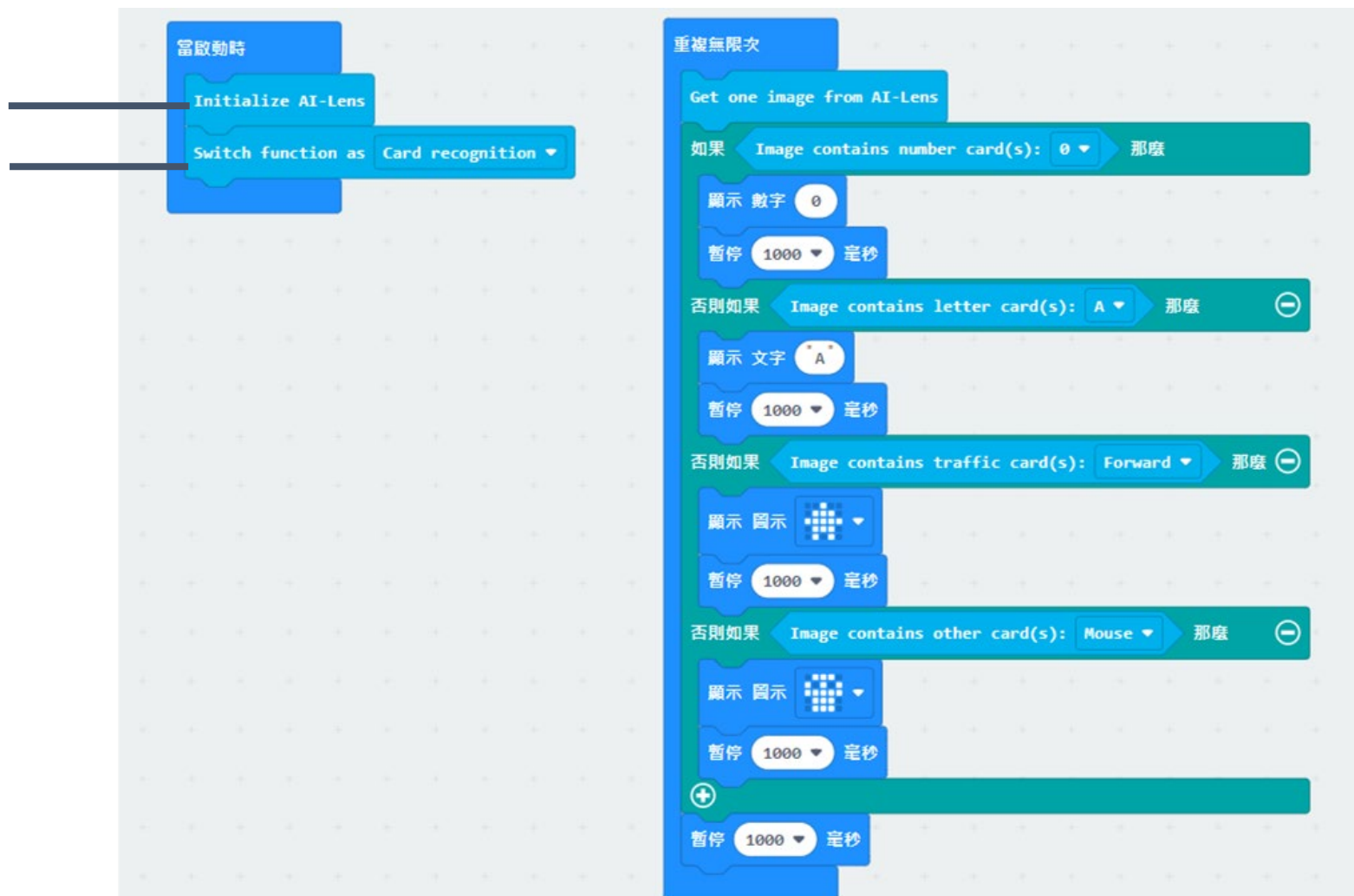
卡片辨識



卡片辨識

數字、英文、方向、圖片四種卡片圖案辨識

初始化AI鏡頭
設定功能



卡片辨識控制移動

練習

請透過卡片前、後、左、右、停控制小車移動。



卡片辨識控制移動

練習

請透過卡片前、後、左、右、停控制小車移動。

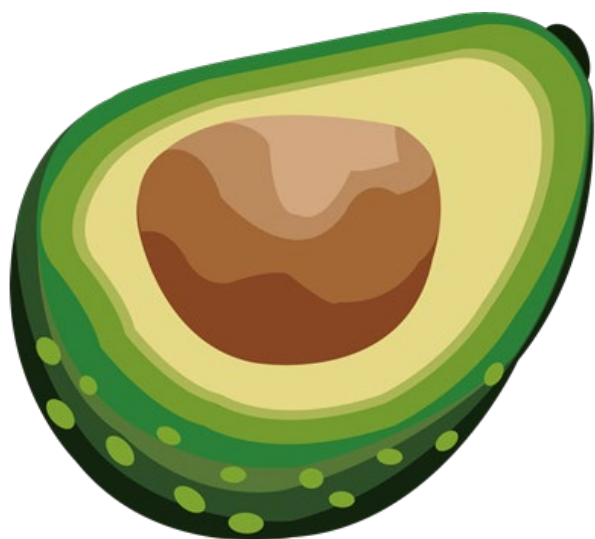
The image displays a Scratch script designed for controlling a robot based on traffic card recognition. The script is organized into three main sections:

- 當啟動時 (When Started):**
 - Initialize AI-Lens
 - Switch function as Card recognition
- 重複無限次 (Repeat Forever):**
 - Get one image from AI-Lens
 - Conditionals and actions for card types:
 - 如果 Image contains traffic card(s): Forward → 那麼: 呼叫 forward
 - 否則如果 Image contains traffic card(s): Back → 那麼: 呼叫 backward
 - 否則如果 Image contains traffic card(s): Turn left → 那麼: 呼叫 left_turn_0
 - 否則如果 Image contains traffic card(s): Turn right → 那麼: 呼叫 right_turn_0
 - 否則如果 Image contains traffic card(s): Stop → 那麼: 呼叫 stop
 - 暫停 100 毫秒
- 定數函式 (Custom Functions):**
 - forward:** 設置电机M1的速度为 100 电机M2的速度为 100
 - backward:** 設置电机M1的速度为 -100 电机M2的速度为 -100
 - left_turn_0:** 設置电机M1的速度为 -100 电机M2的速度为 100
 - right_turn_0:** 設置电机M1的速度为 100 电机M2的速度为 -100
 - stop:** 設置电机M1的速度为 0 电机M2的速度为 0

Learn Object

練習

請訓練以下三個圖案，並成功進行辨識。



Learn Object

當AB同時按下，
儲存此影像為ID1

當A按下，儲存此
影像為ID2

當B按下，儲存此
影像為ID3

當  按下，
完成訓練跳出迴圈

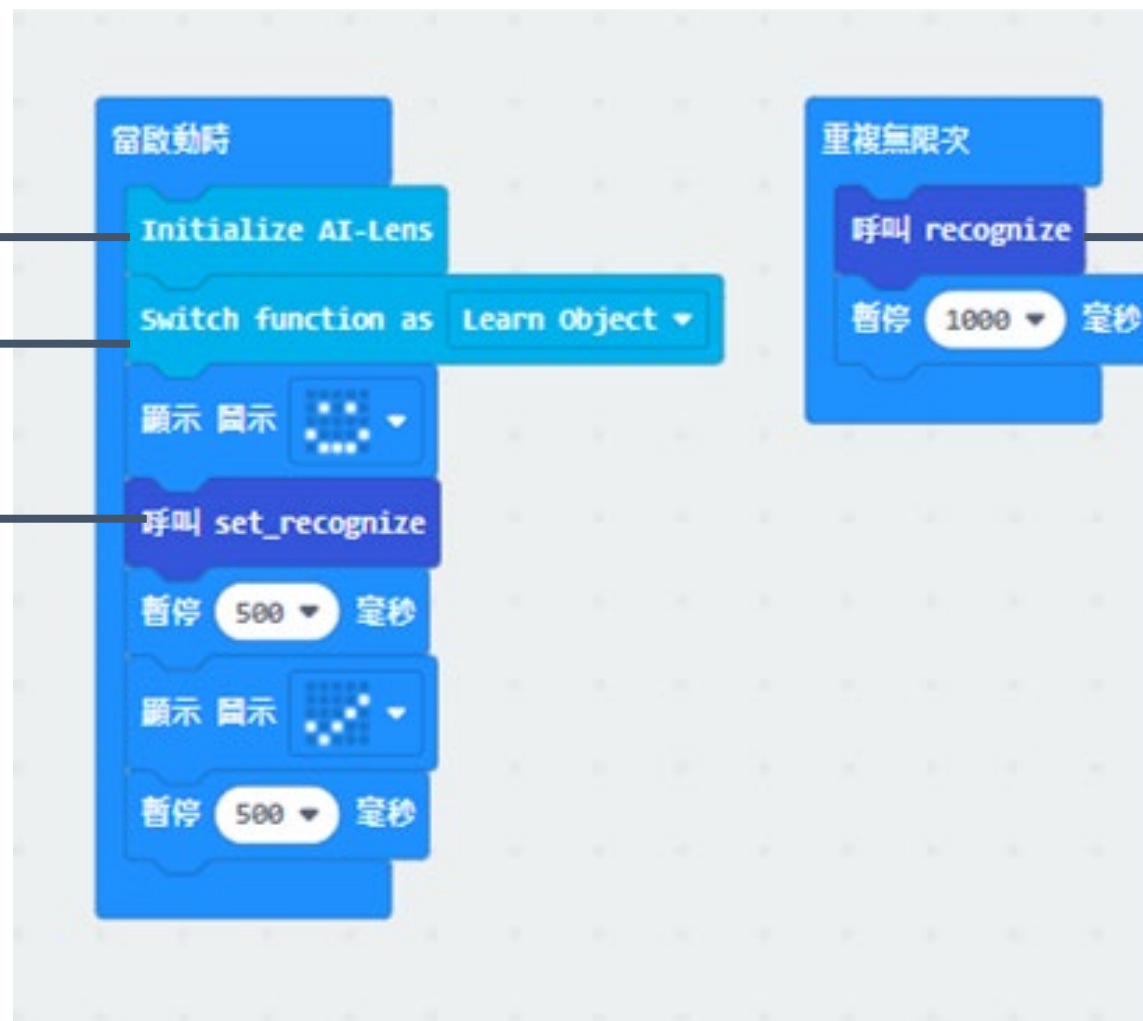


判斷並顯示數字

獲取影像

Learn Object


初始化AI鏡頭
設定功能
圖片訓練



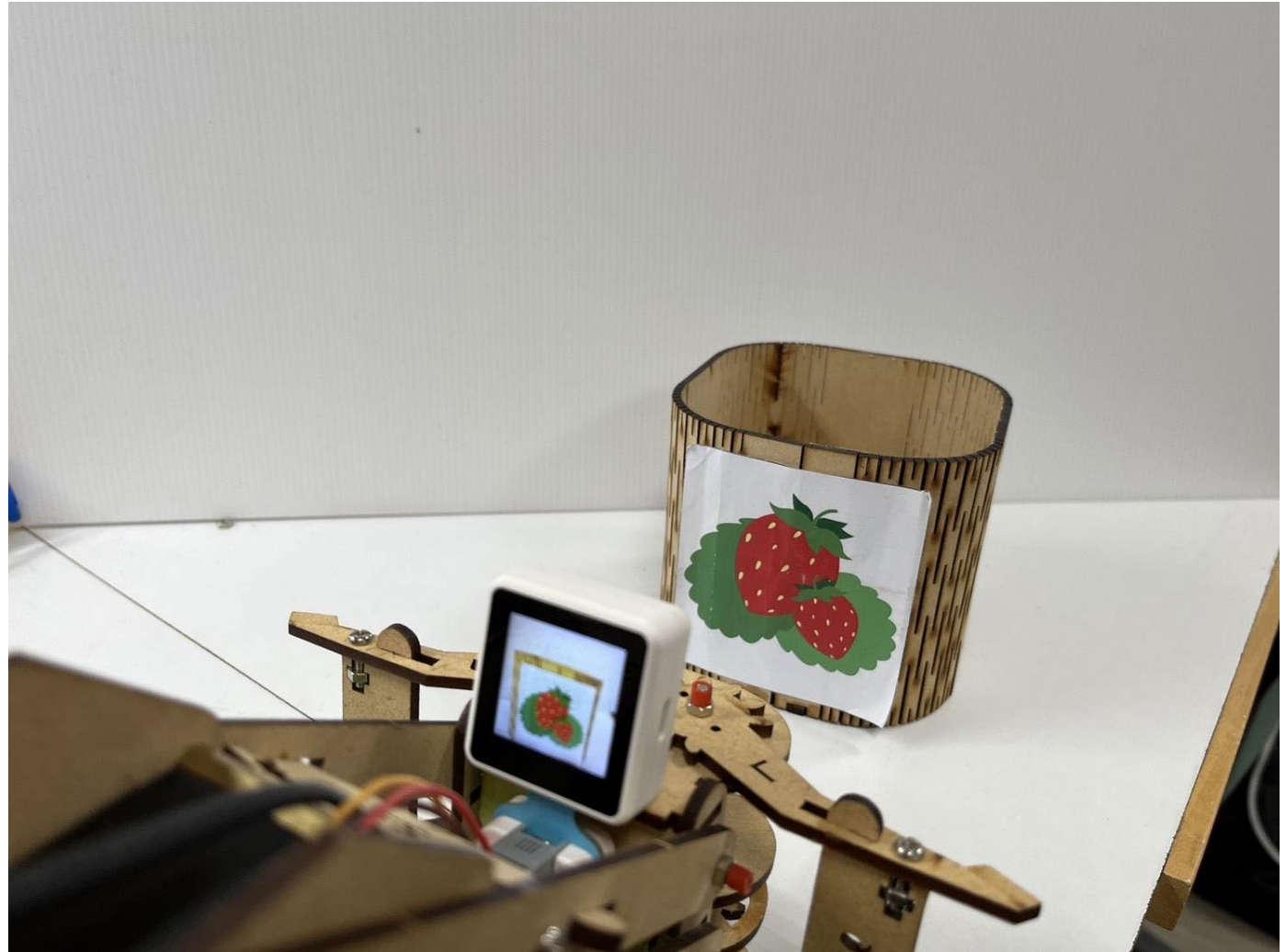
每隔一秒
重複進行辨識

Learn Object

訓練模型步驟：

- 1.同時按下A、 B， 儲存為ID1
- 2.按下A， 儲存為ID2
- 3.按下B， 儲存為ID3
- 4.按下  ， 完成訓練
- 5.開始辨識

1. 每次按下按鈕訓練時，需維持5秒，畫面左下角會呈現5、4、3、2、1，完成後此圖案才完成訓練
2. 此功能辨識時物件內容、大小需盡量與訓練時大小一模一樣，因此盡量放至最大進行訓練，減少雜亂背景



Learn Object

練習

在三岔路終點擺上三個物件，需辨識正確物件並夾回出發點。

