



Semi & AIOT Coding 智慧物聯-

以 Scratch 連結 **Yabboni** 介紹與操作-Semi Challenge

Date:2025/4/30

Speaker: 國立陽明交大電物碩一 劉容維

Demo Video



Semi Challenge

Scratch

Rabboni

利用Scratch和Rabboni -Semi Challenge

報告人：劉容維

電物所

Contents

γabboni-介紹

γabboni-感測參數介紹

γabboni-操作功能介紹

γabboni-配件介紹

γabboni-軸向定義

γabboni Scratch 連線

γabboni - Scratch UI介紹

γabboni-USB連線

γabboni-藍芽BLE 連線

γabboni-Scratch連線

γabboni-Scratch 範例程
式



APPENDIX

γabboni-其他應用

<https://12u10.lab.nycu.edu.tw/>



IOT: Internet of Things



SENSORS *will be everywhere !!!!*

IMU:
Inertial Measurement Unit

- 加速度 (Accelerometer)
- 角速度 (Gyro)
- 磁力計 (Magnetometer)



半導體
Semiconductor

What is IMU ? Rabboni is an IMU.

Inertial Measurement Unit

慣性的

物體抗拒其運動狀態被改變的性質。

Accelerometer 加速規

測量移動 (加速度)

測量單位時間內速度變化

Gyroscope 陀螺儀

測量轉動 (角速度)

測量單位時間內角度變化

GeoMagnetic 地磁儀

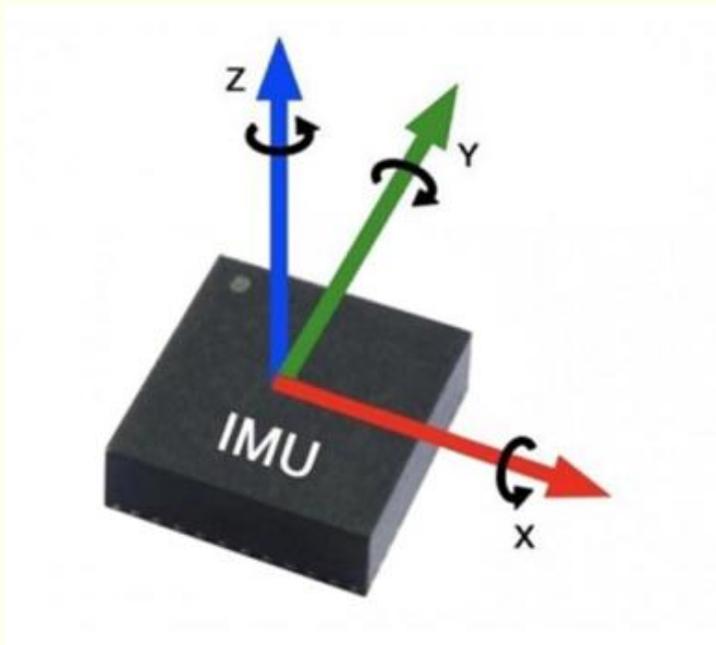
測量地磁方向、大小

可用於定向



半導體
Semiconducto
r

Sensor 入門：聊聊半導體



聊聊 AIOT :
從 Rabboni + Scratch 開始 「貓兔同籠」



入門

資料親子



從 Rabboni  × SCRATCH 開始





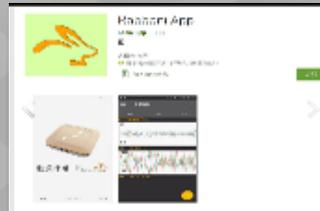
Rabboni is not just a device, It's a platform.

IMU 重力感測



內建六軸重力感測器

Data Extractor 重力感測數據擷取



Android
iOS

API 應用程式介面

- Scratch
- Python
- App Inventor
- Unity

AI Algorithm 演算法開發

- 行動偵測
- 姿態偵測
- 數據分析
- 訊號分析

AIoT 應用程式

- Sports
- Health
- Gaming
- Education



IoT 物聯/雲端介面

- Phone
- Pad
- NB
- Raspberry Pie

EDU 教育資源

- 企業社會責任
- 大學社會責任
- 縣市教育局處合作

r



- yabboni內建六軸重力感測器 (IMU: Initial Measurement Unit)、BLE藍芽傳輸及運算元件
- 可即時傳輸感測讀值並提供取樣頻率及動態範圍
- 配有LED燈，指示rabboni運作狀態及電量顯示



介紹

電池容量 充電方式	120mAh 鋰離子充電電池 USB mini 充電
無線傳輸	Bluetooth 4.0 BLE
充電時間	30分鐘
待機時間	5天 (電源開關鍵OFF)
連續使用時間	8 小時
支援作業系統	藍芽：Android USB：系統Windows 7以上



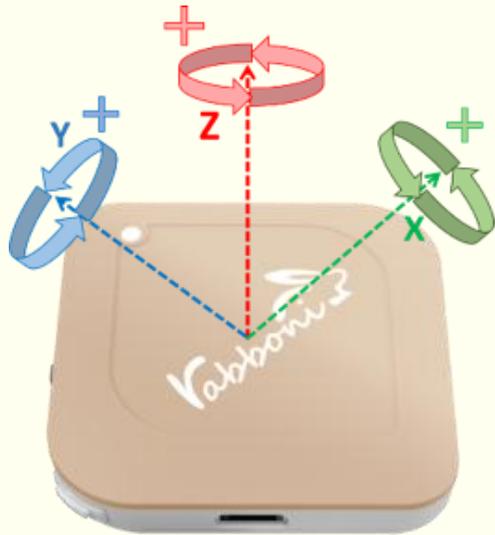
直接看我操作

!



感測參數及軸向介紹

- 直線軸：X/Y/Z 加速度 (Acceleration)
- 環狀軸：X/Y/Z 角速度 (Gyro)



Gyro Full Scale Range	Gyro Sensitivity	Accel Full Scale Range
(°/sec)	(LSB/°/sec)	(g)
±250	65.5	±2
±500	32.8	±4
±1000	16.4	±8
±2000	8.2	±16



Yabboni 配件介紹



Yabboni本體 (正面)



Yabboni本體 (背面)



USB轉接線一條



Yabboni背夾



魔鬼氈手腕帶 (寬2公分、長27.5公分)

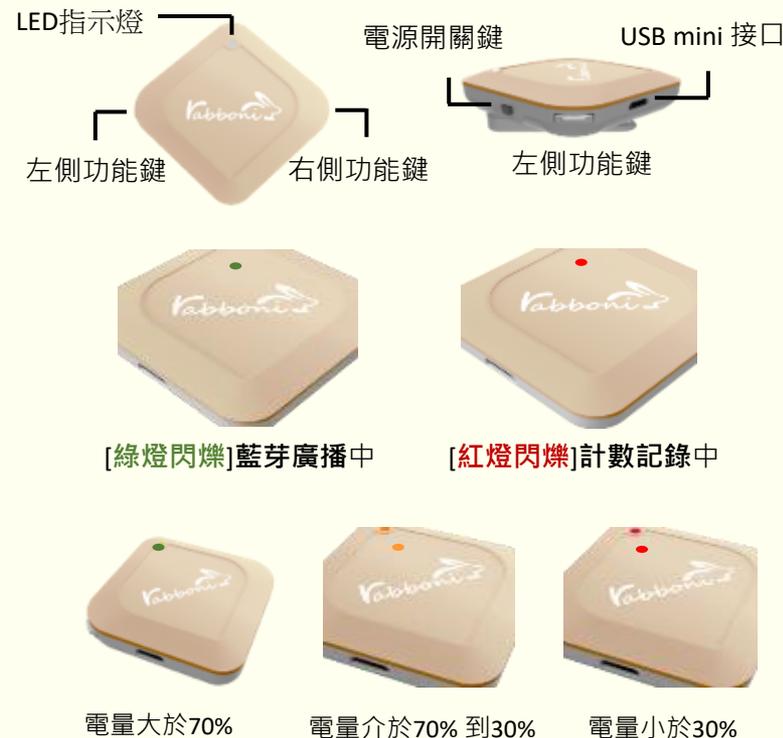
- USB Type A轉接 USB mini線，可提供傳輸數據以及充電功能。

- 提供使用者跑步或行進間Yabboni主體與鞋面穩固結合，確保動作正確偵測



Kabboni 配件介紹

電源開關鍵	單刀開關	On/off 標示
左側功能鍵	(短按1秒)	計數紀錄開始與結束(LED紅燈)
右側功能鍵	(短按1秒)	藍芽廣播開啟・與藍芽裝置配對(LED綠燈)
	(長按5秒)	電量顯示
LED電量指示燈號	(紅)	錄影指示燈、電量小於30%
	(橘)	關機指示燈、電量小於70%
	(綠)	配對指示燈、電量大於70%



[長按右鍵5秒]可以確認電量狀態





Scratch桌面板多連使用說明

安裝安裝.....再安裝



Scratch 桌面板多連程式下載

1. 進入連結：<https://12u10.lab.nycu.edu.tw/downloads/>
2. 如果出現警告，選擇“仍要下載”
3. 選擇“儲存”



下載專區

一般檔案下載

教材內容	
<p>Scratch</p> <p>Scratch 是麻省理工媒體實驗室終身幼稚園組開發的一套電腦程式開發平台，旨在讓程式設計語言初學者不需先學習語言法便能設計產品。</p>	檢視+
<p>Python</p> <p>Python 是種廣泛使用的直譯式、進階程式、通用型程式語言，有吉多·范羅蘇姆創造，第一版釋出於1991年。</p>	檢視+
<p>Rabonni AIoT 程式教育感測裝置申請表</p>	檢視+
<p>Rabboni API</p> <p>文文孟參考資料 API 3.0 含有Scratch,Python,APInventor,Raspberry Pi, Arduino等套件</p>	檢視+



<p>Scratch</p> <p>Scratch 是麻省理工媒體實驗室終身幼稚園組開發的一套電腦程式開發平台，旨在讓程式設計語言初學者不需先學習語言法便能設計產品。</p>	檢視-
<p>教材名稱：Rabboni-scratch 作者：溫環岸 簡章介紹：本教材為scratch連結rabboni介紹與操作教學。</p>	<p>下載</p> <p>次數：1333</p>
<p>教材名稱：Rabboni-scratch + Sensor 作者：溫環岸 簡章介紹：本教材為scratch連結rabboni介紹與操作教學，並結合感測參數及製程介紹。</p>	<p>下載</p> <p>次數：938</p>
<p>教材名稱：10/26 師培Rabboni-scratch + Sensor 作者：朱保銘 簡章介紹：本教材為scratch一對多連結rabboni介紹與操作教學。</p>	<p>下載</p> <p>次數：262</p>
<p>教材名稱：10/28 師培Rabboni-scratch + Sensor 作者：林志威 簡章介紹：本教材為scratch一對多連結rabboni介紹與操作教學。</p>	<p>下載</p> <p>次數：247</p>
<p>教材名稱：11/02 師培Rabboni-scratch + Sensor 作者：游天維 簡章介紹：本教材為scratch一對多連結rabboni介紹與操作教學。</p>	<p>下載</p> <p>次數：366</p>
<p>教材名稱：AIOT Coding 智慧物聯師培中心(中小學) 作者：溫環岸 簡章介紹：本教材為scratch連結rabboni介紹與操作教學，並結合感測參數及製程介紹。</p>	<p>下載</p> <p>連結</p> <p>次數：37</p>



SCRATCH 桌面板多連程式下載

1. 進入連結：<https://12u10.lab.nycu.edu.tw/downloads/>
2. 如果出現警告，選擇“仍要下載”
3. 選擇“儲存”



2. 如果出現警告，選擇“仍要下載”





Scratch桌面板多連使用說明-USB連線



SCRATCH 桌面板多連使用說明-USB連線

STEP 1

解壓縮後，打開資料夾，點擊應用程式開啟



STEP 2

開啟應用程式



點擊新增裝置



!!! 注意 !!!

USB只能連線1個rabboni

藍芽最多同時4個 rabboni

STEP 3

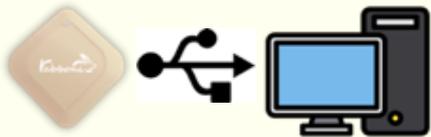
出現選擇連線方式視窗，
可以選擇USB或藍芽連線



SCRATCH 桌面板多連使用說明-USB連線

STEP 4

連結USB與電腦



STEP 6

選擇裝置



STEP 8

選擇「確認」



STEP 5

點擊USB的選項



STEP 7

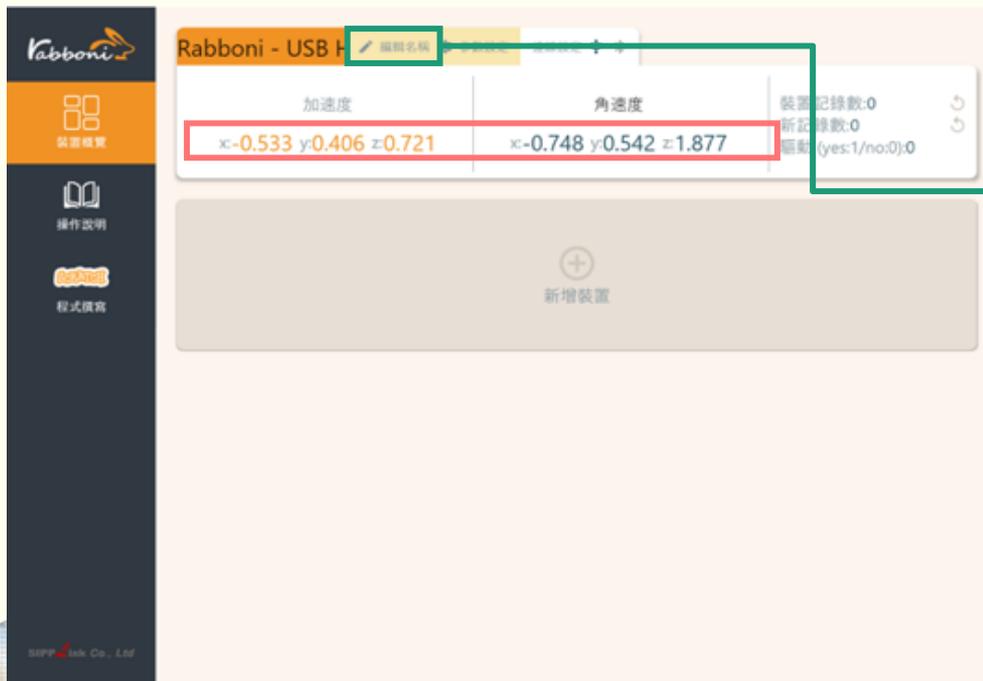
選擇 Rabboni – USB HID UART Bridge



SCRATCH 桌面板多連使用說明-USB連線

STEP 9

數字跳動代表連線成功



STEP 10

可以編輯裝置在電腦上的名稱，
名稱會對應到Scartch裡



STEP 11

按下「確認」後，名稱改變





Scratch桌面板多連使用說明-藍芽連線



半導體

Semiconductor

SCRATCH 桌面板多連使用說明-藍芽連線

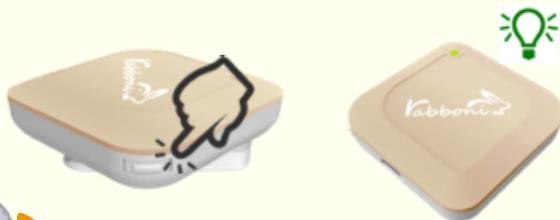
STEP 1

- 電腦沒有藍芽:連結dongle與電腦
- 電腦沒有藍芽:請確認藍芽在4.0-5.1間



STEP 2

短按**右鍵**1秒，開始藍芽連線，綠燈會閃爍直到配對成功。若無配對到手機，會自動於30秒後停止廣播。



藍芽連線手機成功後，**綠燈**每10秒閃爍一次

半導體
Semiconductor

STEP 3

點擊「藍芽」的選項



STEP 4

選擇裝置



STEP 5

選擇欲連結rabboni裝置的MAC碼



MAC碼在rabboni的本體背面

STEP 6

選擇「確認」



SCRATCH 桌面板多連使用說明-藍芽連線

STEP 7

數字跳動代表連線成功



STEP 8

可以編輯裝置在電腦上的名稱，
名稱會對應到Scartch裡



STEP 9

按下「確認」後，名稱改變



SCRATCH

桌面板多連使用說明-藍芽連線

STEP 10

新增其他裝置



點擊新增更多裝置



SCRATCH 桌面板多連使用說明-藍芽連線

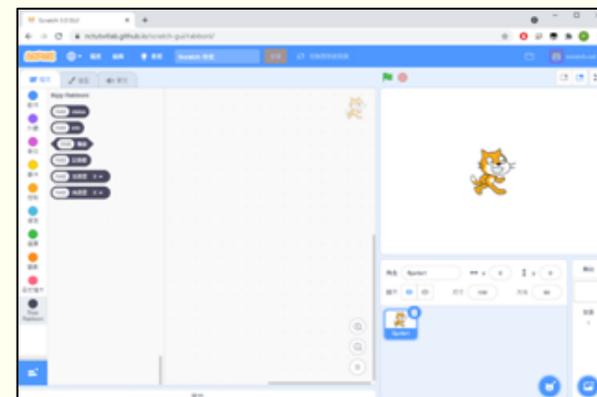
STEP 10

點擊左邊Scratch的ICON



STEP 11

點擊Scratch的ICON，跳轉到瀏覽器



<https://nctutwlab.github.io/scratch-gui/rabboni/>

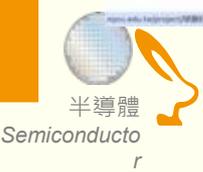
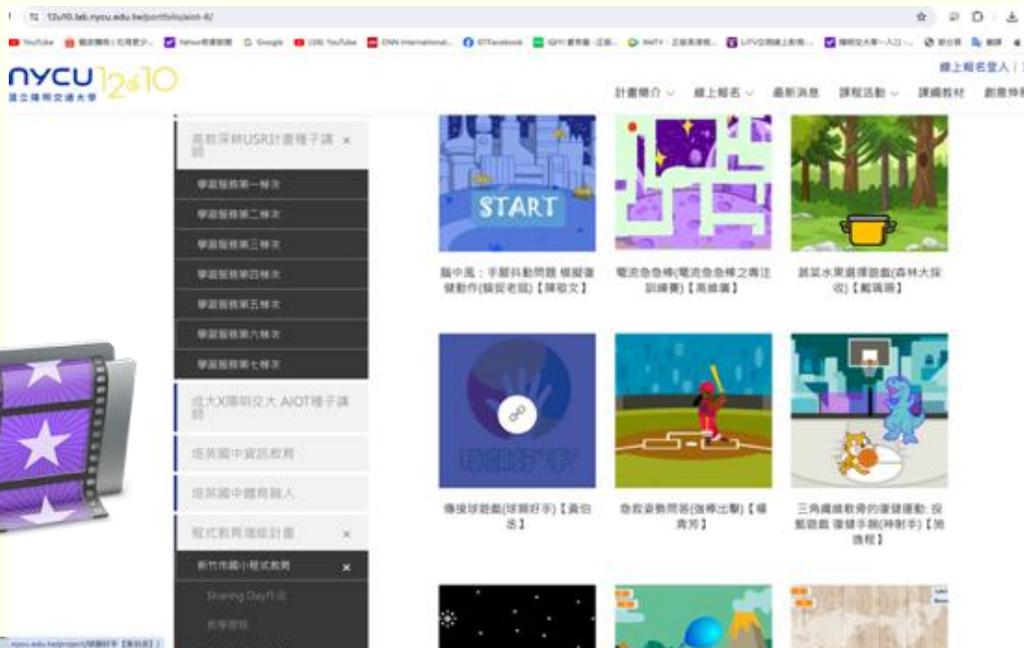


半導體
Semiconductor

程式觀摩:

從別人設計的程式開始: 到『12u10 一定要你贏』網站去逛逛喔

<https://12u10.lab.nycu.edu.tw/portfolio/aiot-6/>





利用Scratch和Rabboni - Semi Challenge

報告人：劉容維

電物所





目錄 / CONTENTS



發想



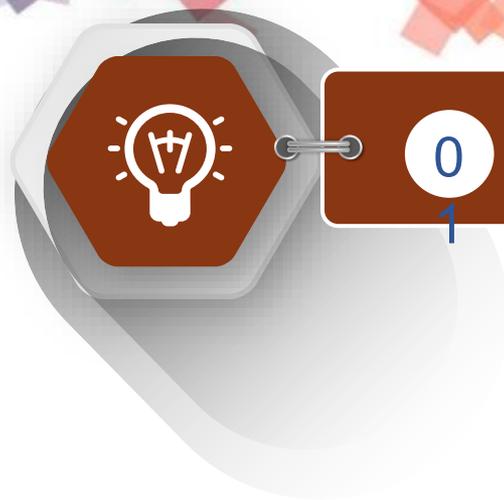
遊戲簡介



遊戲影片

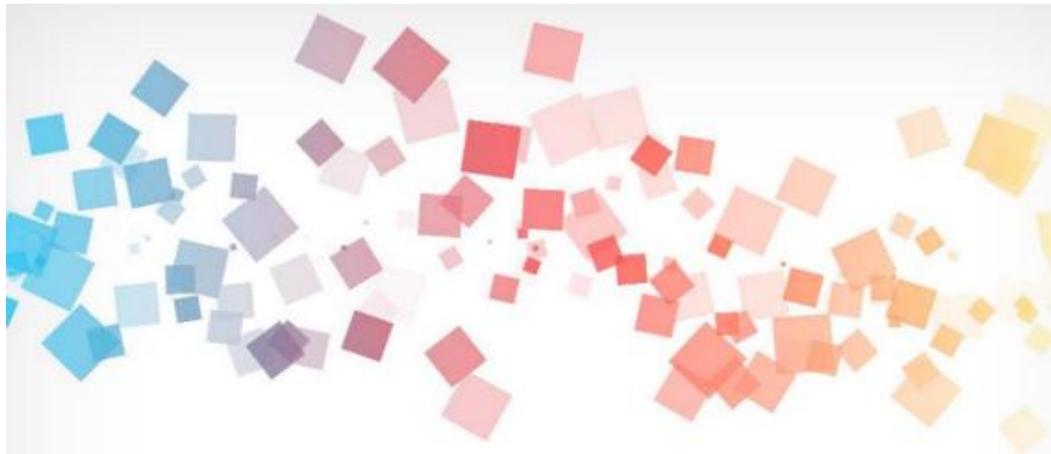


程式介紹



0
1

發想



在生活當中，半導體與我們息息相關，從最常接觸的手機、電視，又或者是微波爐上的顯示面板，都讓我們不知不覺接觸到半導體，也引發了我對當中的興趣，並探討當中最根本原理，也就是材料方面。因此我想藉由設計的Rabboni遊戲來對學生們進行機智問答，同時驗收孩子們是否有專心聽講。

製作原因

在網路或電視看過許多機智問答的益智節目，目的是讓參賽者們能夠在短時間內對一些冷知識或者題目續速做出回應，對此我想出了讓學生們能夠專心上課聽講，並同時擁有娛樂性的半導體挑戰。



圖片來源:[天才衝衝衝](#)



▶▶▶ 遊戲玩法簡介



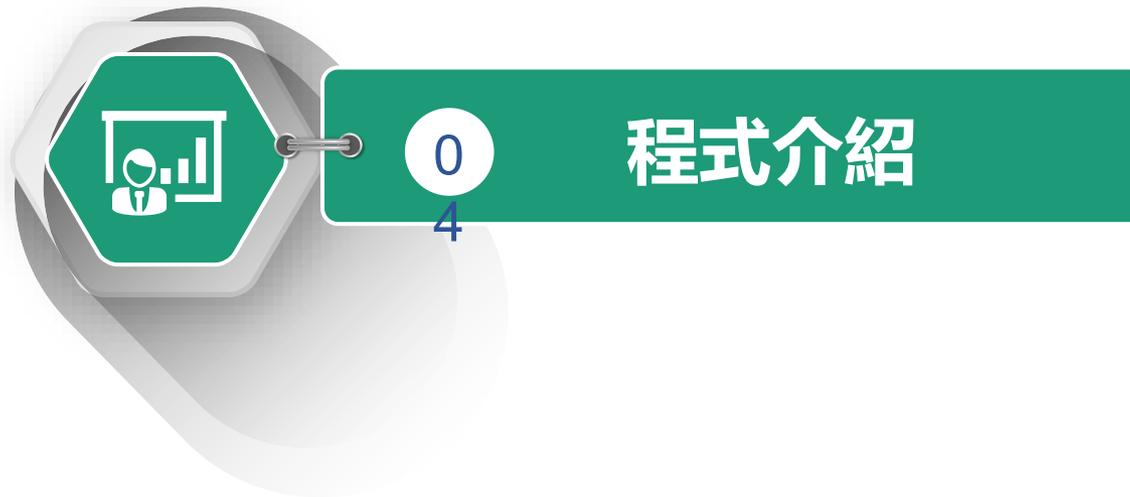
玩家利用rabboni的x軸加速度方向，來操控角色到正確的選項上。每道題目有5秒鐘的思考時間，在站到正確選項的時候，得分數加1，共五題，全對則獲得勝利。



遊戲影片



此為答題成功

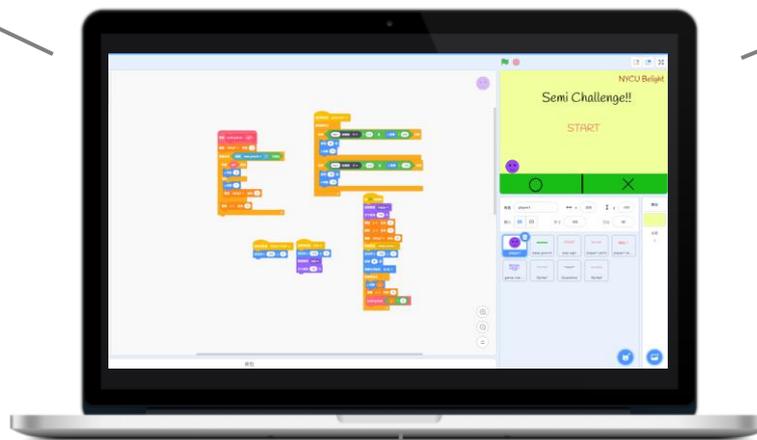


▶▶▶ 程式介紹

遊戲開始
點選start

獲勝或失敗判斷
判斷切換圖片數，讀取成功或失敗

分數
藉由讀取角色位置，判斷是否切換分數，進行加分。



Start1
鼠標點選start，即可開始

角色
透過偵測rabboni的X軸加速度，將腳色移動至正確選項

題目產生
透過調整延遲時間切換題目。

Start1-程式碼

```
當 被點擊
  定位到 x: 0 y: 10
  顯示
  重複無限次
    如果 碰到 鼠標 ? 那麼
      圖像效果 顏色 設為 20
    否則
      圖像效果 顏色 設為 5

當角色被點擊
  如果 Rab1 status = NOT_READY 那麼
    廣播訊息 player1 not ready
  否則
    廣播訊息 game start
    隱藏
```

上方判斷當鼠標碰到產生變色醒目效果，點擊後判斷 Rab 是否連上，如果連接失敗，回傳沒有準備好，有的話發出遊戲開始指令。

```
當 被點擊
  定位到 x: -339 y: 25
  隱藏

當收到訊息 game start
  顯示
  滑行 0.7 秒到 x: 0 y: 25
  等待 3.0 秒
  滑行 0.7 秒到 x: 339 y: 25
  隱藏
  廣播訊息 question
```

此為遊戲規則說明，在接受到滑鼠點擊發出的遊戲開始，滑行道中間，暫停 3 秒供學生觀看，之後滑出畫面後，發出題目指令。

角色-程式碼

```
當 被點擊  
造型換成 happy  
尺寸設為 100 %  
變數 x 設為 0  
變數 y 設為 0  
變數 falling? 設為 10  
廣播訊息 delete clones  
定位到 x: -200 y: 0  
面朝 90 度  
迴轉方式設為 左-右  
重複無限次  
  y 改變 y  
  變數 y 改變 -2  
  touch ground y > 0
```

切換一般角色，並設定角色到初始位置，並且每次接觸到地板不讓他掉到地面以下

```
定義 touch ground up?  
變數 falling? 改變 1  
重複直到 碰到 base ground ? 不成立  
  如果 up? 那麼  
    y 改變 -2  
  否則  
    y 改變 1  
    變數 falling? 設為 0  
變數 y 設為 0
```

定義何為接觸地板，如果碰到地板，改變y分量

角色-程式碼

```
當收到訊息 Player1 Point!
  定位到 x: -200 y: 0

當收到訊息 loss
  定位到 x: -200 y: 0
  造型換成 sad
  尺寸設為 100 %

當收到訊息 game start
  重複無限次
    如果 Rab1 加速度 X > 0.5 且 x 座標 < 230 那麼
      x 改變 10
    如果 Rab1 加速度 X < -0.5 且 x 座標 > -230 那麼
      x 改變 -10
```

因為邊界為230，設定在讀取
加速度變化夠快(換邊)的時候，
改變x方向。並且當最後接收到
失敗，改變成哭臉。

題目產生-程式碼

The image displays three columns of Scratch code blocks. The first column starts with a '當被點擊' (When clicked) event, followed by '隱藏' (Hide), '當收到訊息 question' (When message received), '變數 Current Round 設為 0' (Set variable Current Round to 0), '定位到 x: 0 y: 25' (Go to x: 0 y: 25), '顯示' (Show), '造型換成 1' (Switch costume to 1), '等待 6 秒' (Wait 6 seconds), '定位到 player1 位置' (Go to player1 position), '隱藏' (Hide), and a conditional block '如果 x 座標 > 0 那麼' (If x coordinate > 0 then) containing '廣播訊息 Player1 Point!' (Broadcast message Player1 Point!), '變數 Current Round 改變 1' (Change variable Current Round by 1), '等待 0.5 秒' (Wait 0.5 seconds), '定位到 x: 0 y: 25' (Go to x: 0 y: 25), '顯示' (Show), '造型換成 2' (Switch costume to 2), '等待 6 秒' (Wait 6 seconds), and '定位到 player1 位置' (Go to player1 position). The second column starts with '隱藏' (Hide), a conditional block '如果 x 座標 < 0 那麼' (If x coordinate < 0 then) containing '廣播訊息 Player1 Point!' (Broadcast message Player1 Point!), '變數 Current Round 改變 1' (Change variable Current Round by 1), '等待 0.5 秒' (Wait 0.5 seconds), '定位到 x: 0 y: 25' (Go to x: 0 y: 25), '顯示' (Show), '造型換成 4' (Switch costume to 4), '等待 6 秒' (Wait 6 seconds), '定位到 player1 位置' (Go to player1 position), '隱藏' (Hide), and another conditional block '如果 x 座標 > 0 那麼' (If x coordinate > 0 then) containing '廣播訊息 Player1 Point!' (Broadcast message Player1 Point!), '變數 Current Round 改變 1' (Change variable Current Round by 1), '等待 0.5 秒' (Wait 0.5 seconds), '定位到 x: 0 y: 25' (Go to x: 0 y: 25), '顯示' (Show), '造型換成 Done' (Switch costume to Done), '等待 6 秒' (Wait 6 seconds), and '定位到 player1 位置' (Go to player1 position). The third column starts with '隱藏' (Hide), '等待 6 秒' (Wait 6 seconds), '定位到 player1 位置' (Go to player1 position), '隱藏' (Hide), a conditional block '如果 x 座標 > 0 那麼' (If x coordinate > 0 then) containing '廣播訊息 Player1 Point!' (Broadcast message Player1 Point!), '變數 Current Round 改變 1' (Change variable Current Round by 1), and another conditional block '如果 Current Round < 5 那麼' (If Current Round < 5 then) containing '廣播訊息 loss' (Broadcast message loss).

設定每6秒鐘切換造型(題目)並透過定位角色位置座標，來判斷是否站在正確圈叉。並設定變數current round 來計算答對次數，讓最終判斷是否獲勝時得以判斷次數是否為5次。

分數-程式碼

```
當 被點擊  
隱藏  
當收到訊息 Player1 Point!  
造型換成下一個  
如果 造型 名稱 = 5 那麼  
廣播訊息 Player1 Win!  
當收到訊息 game start  
造型換成 0  
顯示  
定位到 x: -125 y: 100  
當收到訊息 loss  
造型換成 Abby-a  
顯示  
定位到 x: 0 y: 50
```



遊戲開始時切換成0分，並顯示到固定。位置當接收到角色設定時的得分條件，切換造型加一分，當最後計數current round 沒有等於5，傳出的lose在這邊接收，並更改圖案成lose。

獲勝或失敗判斷-程式碼

```
if (x coordinate > 0) then  
  broadcast message: Player1 Point!  
  change Current Round by 1  
if (Current Round < 5) then  
  broadcast message: loss
```

設定變數current round來計算答對次數，讓最終判斷是否獲勝時得以判斷次數是否為5次。

```
when I receive Player1 Point!  
  change sprite to next  
if (sprite name = 5) then  
  broadcast message: Player1 Win!  
when I receive Player1 Win!  
  show
```

如果收到得分5分時，傳出 player1 win，並設定當接收到時，更改圖片成win。



感謝聆聽!

