



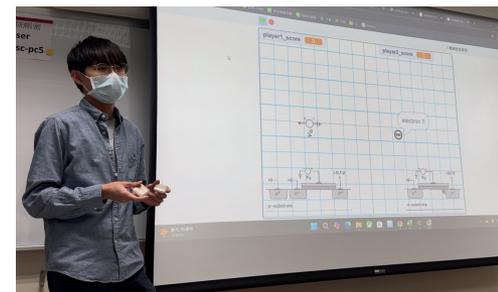
Semi & AIOT Coding 智慧物聯-

電晶體與開關

以 Scratch 連結 **Yabboni** 介紹與操作

Date: 2024 / 11 / 07

Speaker: 姜禮夫



Scratch

Transistor
& Switch

Rabboni

利用Scratch和Rabboni - 電晶體與開關

報告人：姜禮夫

國際半導體產業學院

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

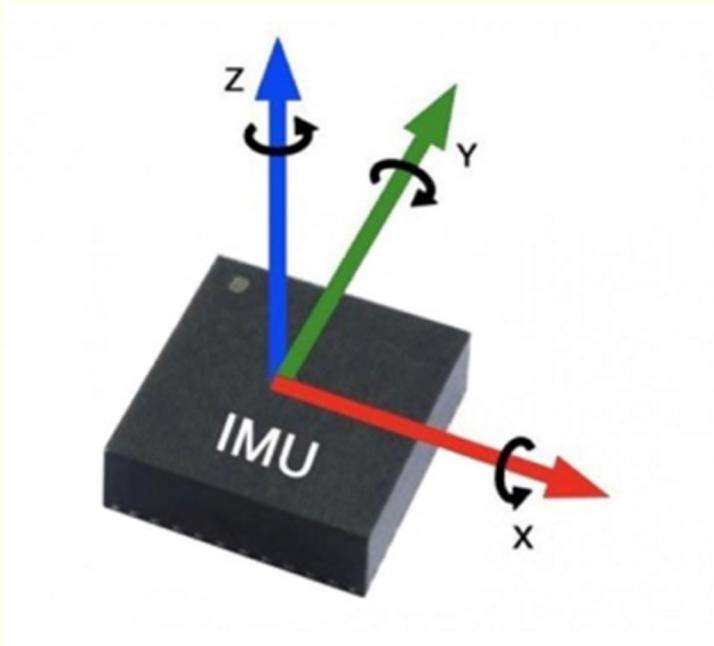


半導體

Semiconducto

r

Sensor 入門：聊聊半導體



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



入門

竹科親子



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

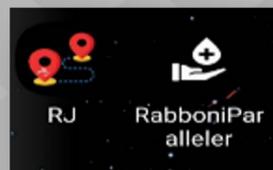
- Scratch
- Python
- App Inventor
- Unity

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



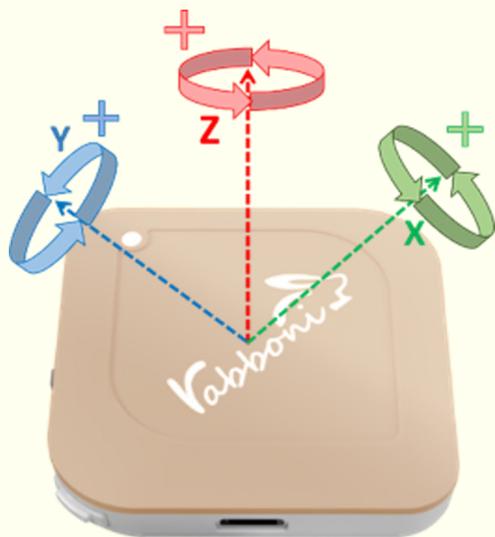
直接看我操作

!



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)

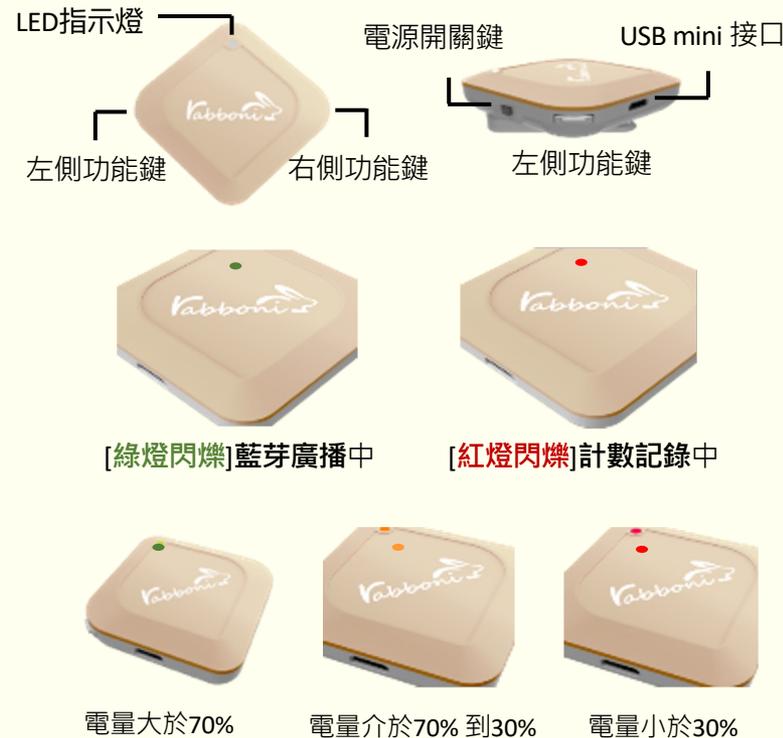
- USB Type A轉接 USB mini線，
可提供傳輸數據以及充電功能。

- 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



半導體

Semiconductor

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>	<div style="border: 1px solid red; border-radius: 50%; width: 20px; height: 20px; display: flex; align-items: center; justify-content: center;"> 1 </div> 檢視+
<p>Rabboni 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>	下載 次數：1333
<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介紹與實作教學，並結合感測參數及製程介紹。</p>	下載 次數：938
<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>	下載 次數：262
<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>	下載 次數：247
<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>	下載 次數：366
<p>教材名稱：AIOT Coding 智慧物聯師培分享(中小學) 作者：溫環岸 簡章介紹：本教材為scratch連結rabboni介紹與實作教學，並結合感測參數及製程介紹。</p>	<div style="border: 1px solid red; border-radius: 50%; width: 20px; height: 20px; display: flex; align-items: center; justify-content: center;"> 2 </div> 下載 連結 次數：37



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

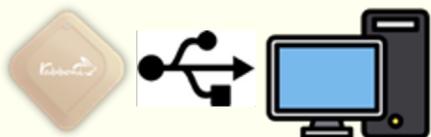
出現選擇連線方式視窗，
可以選擇USB或藍芽連線



SCRATCH 桌面板多連使用說明-USB連線

STEP 4

連結USB與電腦



STEP 5

點擊USB的選項



STEP 6

選擇裝置



STEP 7

選擇 Rabboni – USB HID UART Bridge



STEP 8

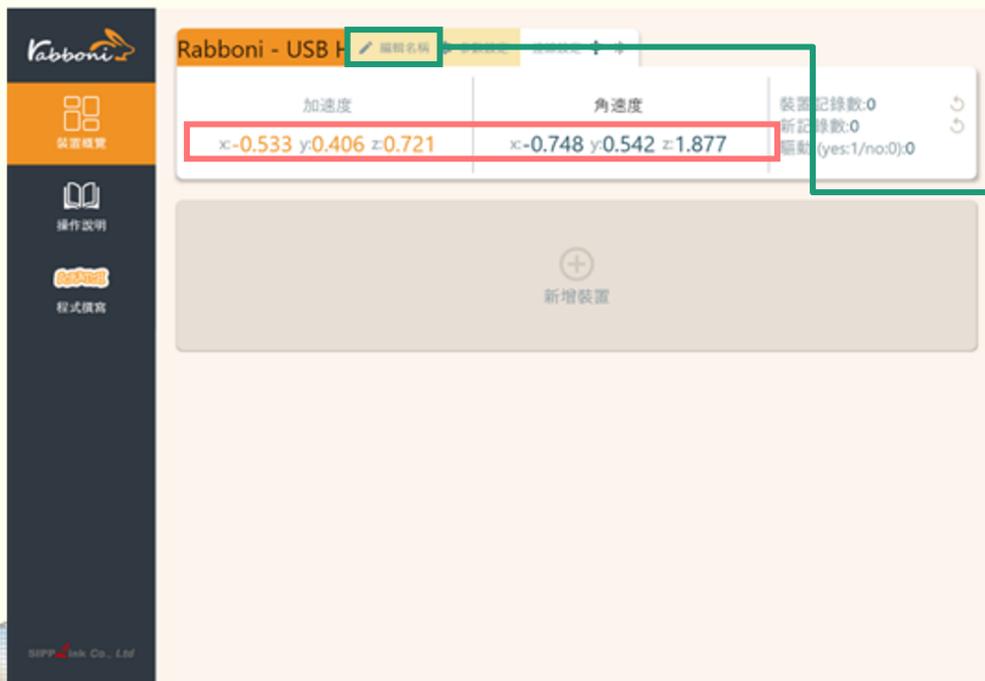
選擇「確認」



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



SCRATCH 桌面板多連使用說明-藍芽連線

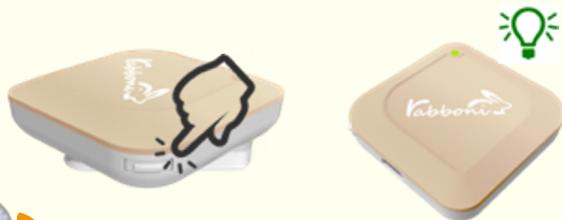
STEP 1

- 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

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



半導體
Semiconductor

Scratch 桌面板多連使用說明-藍芽連線

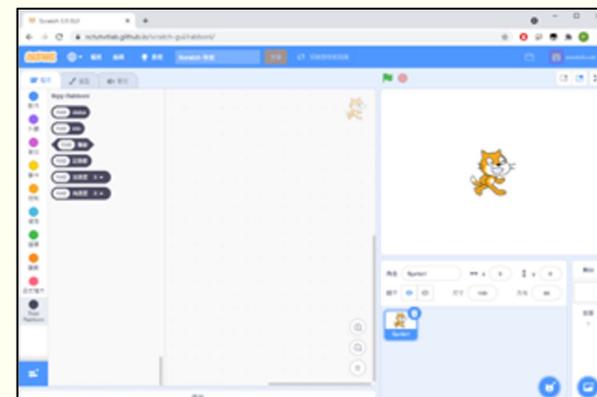
STEP 10

點擊左邊Scratch的ICON



STEP 11

點擊Scratch的ICON，跳轉到瀏覽器



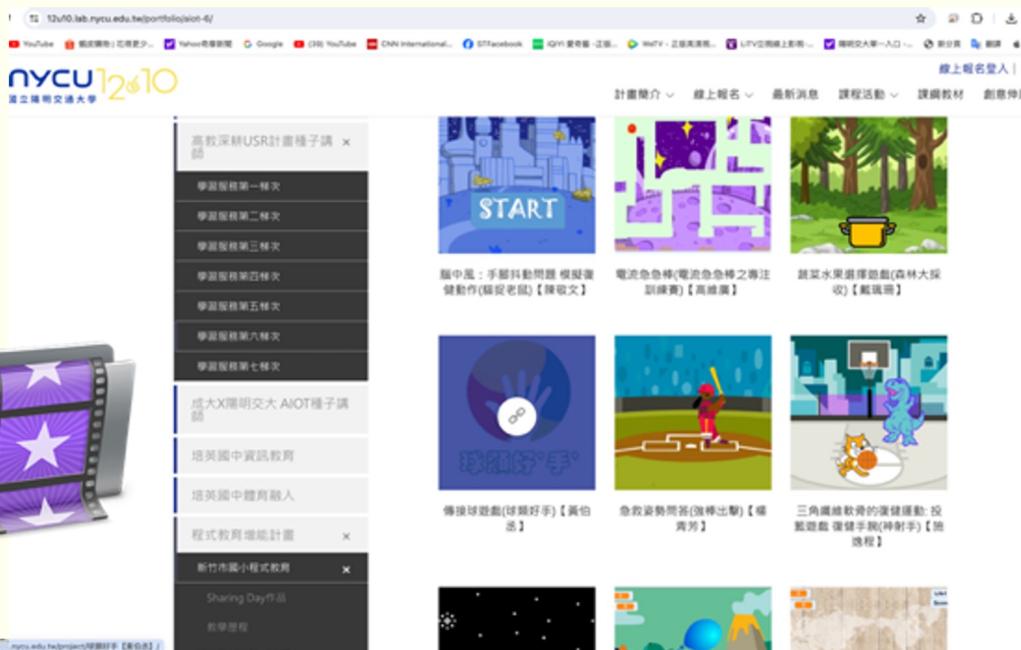
<https://nctutwlab.github.io/scratch-gui/rabboni/>



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/>



創意奔放
SHOW Time!!



利用Scratch和Rabboni - 電晶體與開關

報告人：姜禮夫

國際半導體產業學院





目錄 / CONTENTS



發想



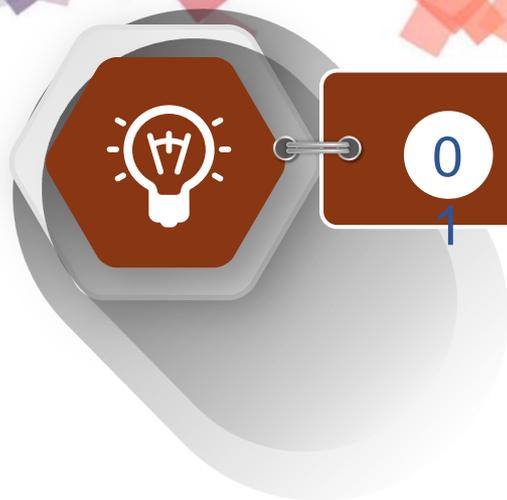
遊戲簡介



遊戲影片

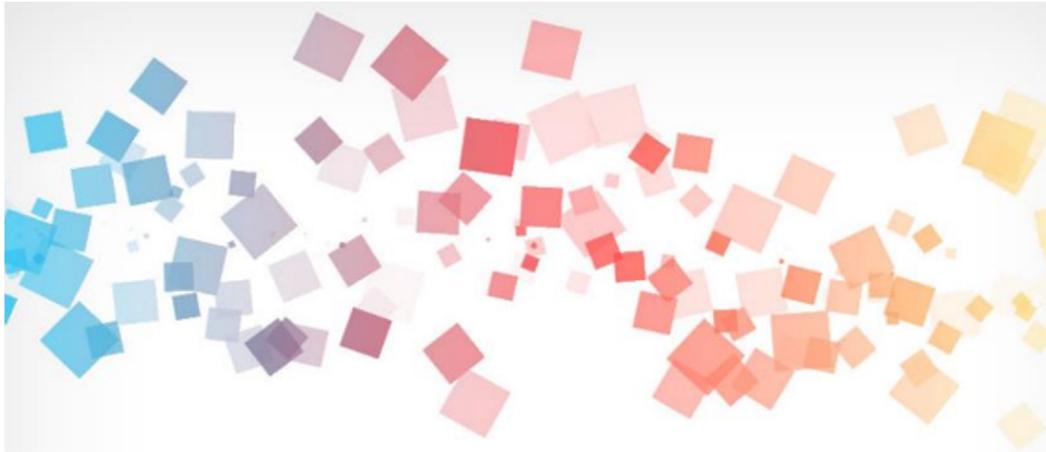


程式介紹



0
↓

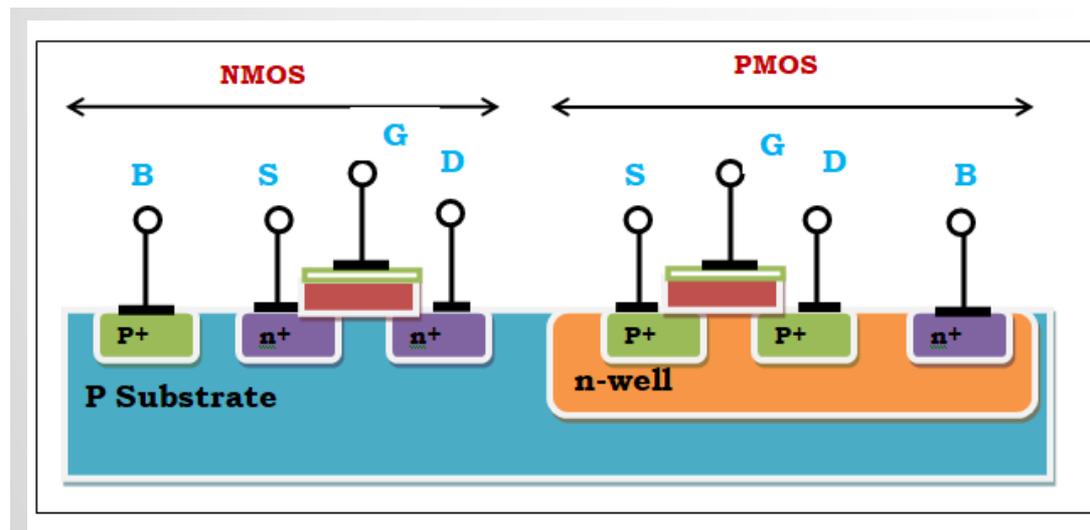
發想



在眾多的半導體元件中，當今我們最常使用的就屬 MOSFET(金屬氧化物半導體場效電晶體)，而他們又常以P/N兩種成對出現使用，構築了我們現今多樣的半導體世界。

製作原因

而我們時常將MOS形容成開關，到底這樣的半導體元件是如何被當成開關操縱，是我想透過這個小遊戲，藉由與Rabboni協作下所傳達的。希望讓大家在遊玩的過程中，學習到基本的半導體元件知識。



圖片來源:<https://medium.com/@alphacode388/cmos-digital-design-fundamental-0e024bee3c89>



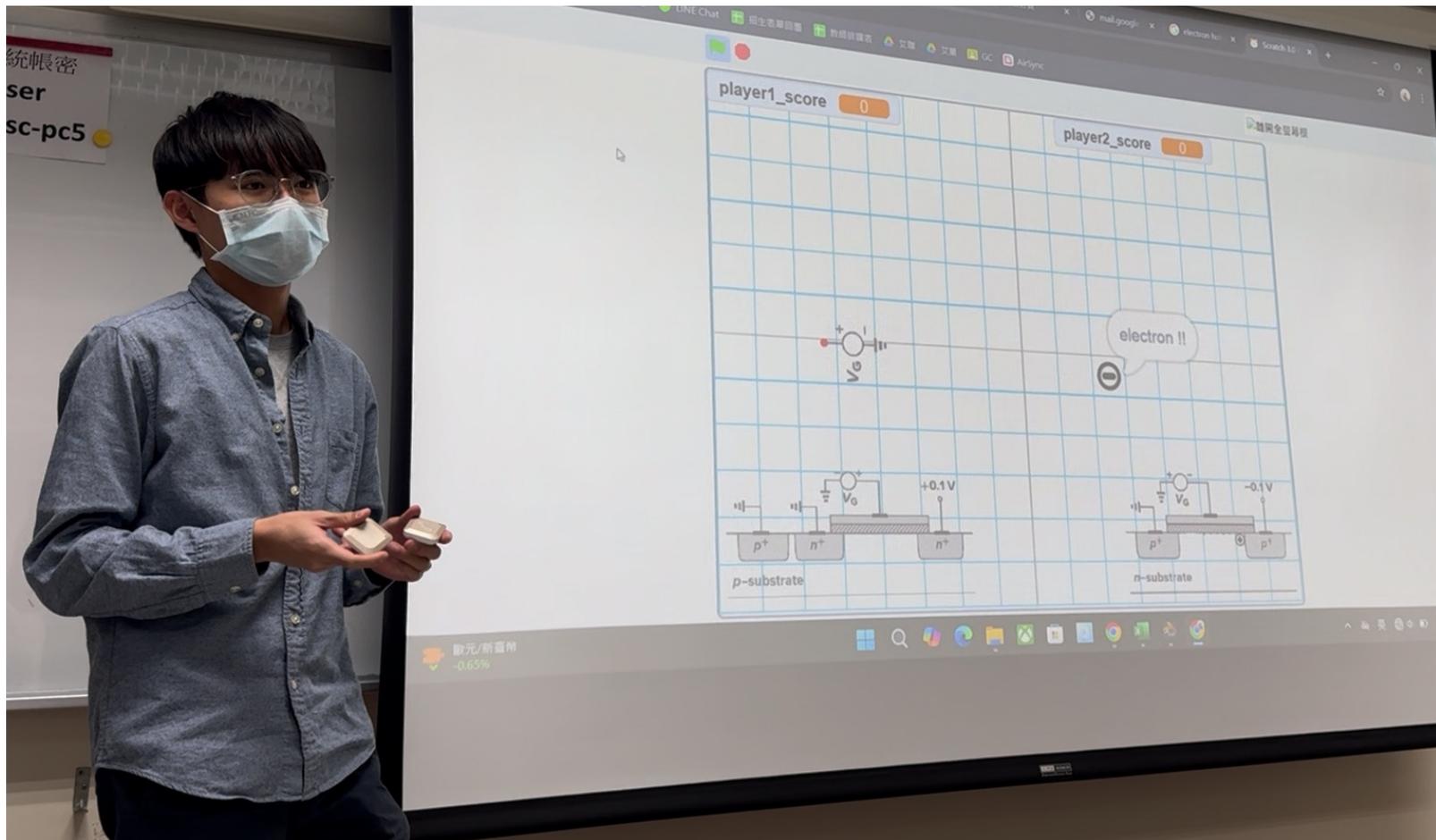
▶▶▶▶ 遊戲玩法簡介

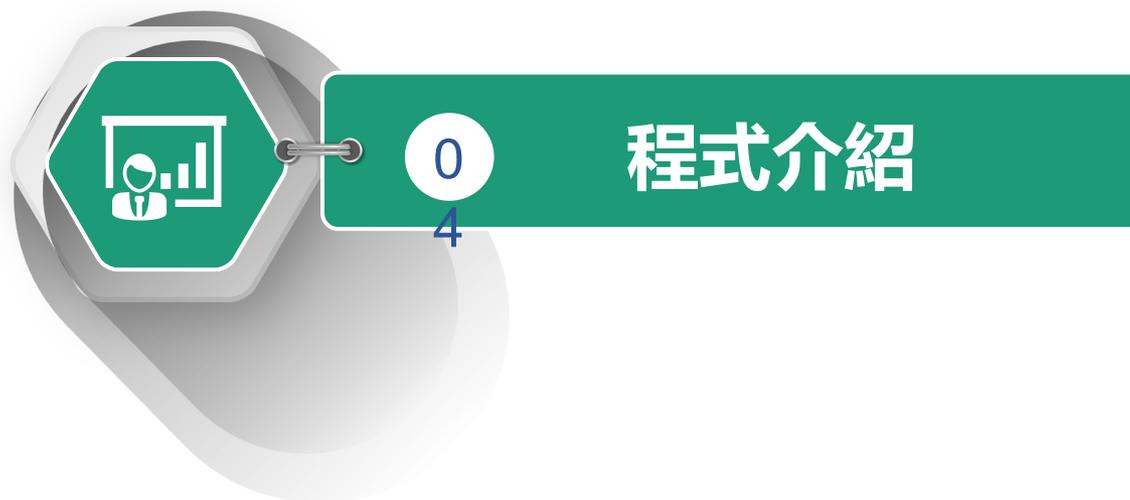


玩家利用rabboni作為搖桿，透過不同方向傾斜、移動的XYZ軸加速度及角速度來操控角色控制訊號來去接觸對應目標的電子或電洞，分數是由總共接觸到的電荷量來決定的。若有一方吃到足量的載子，使元件導通，遊戲就會結束，由那方獲勝

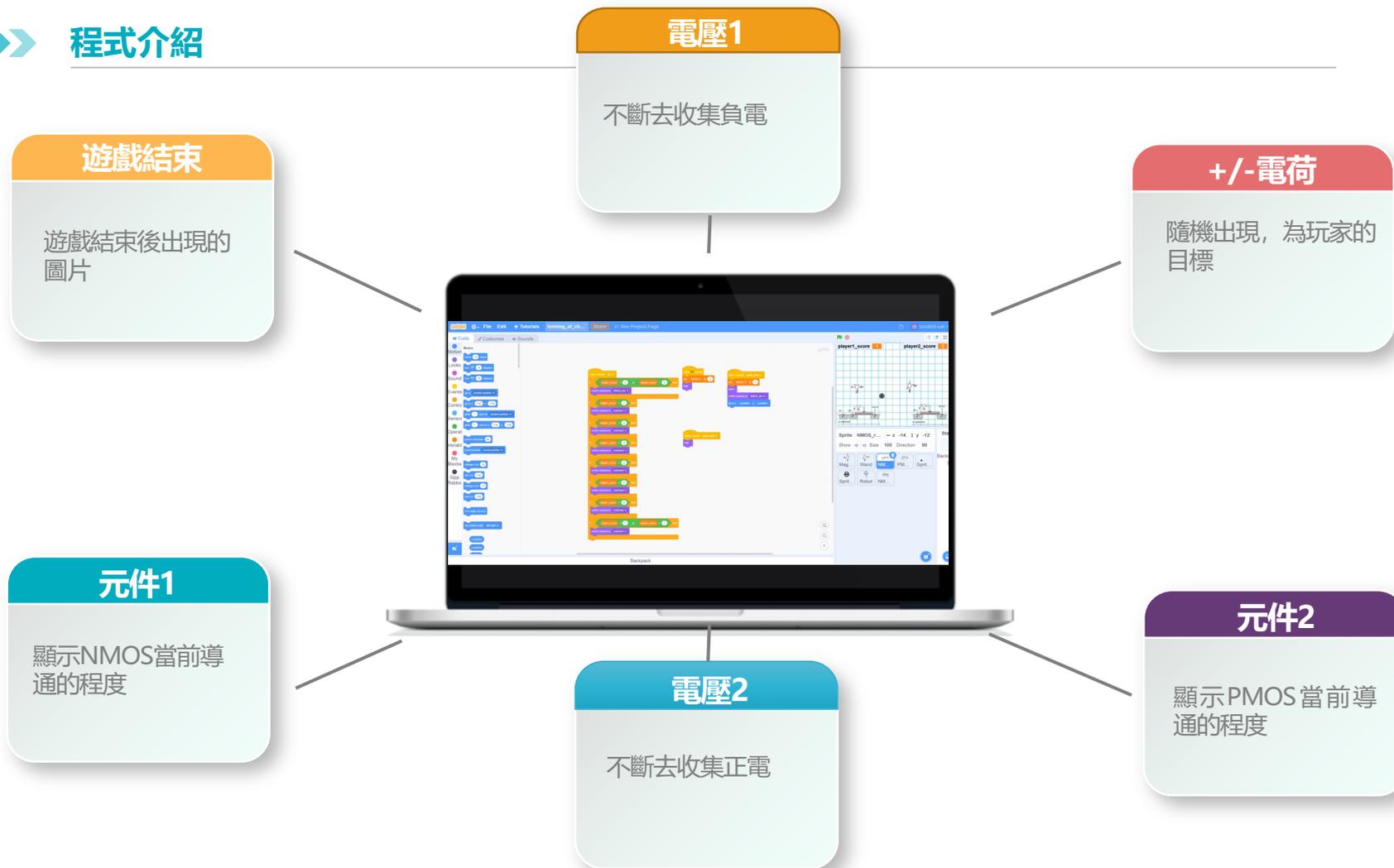


遊戲影片

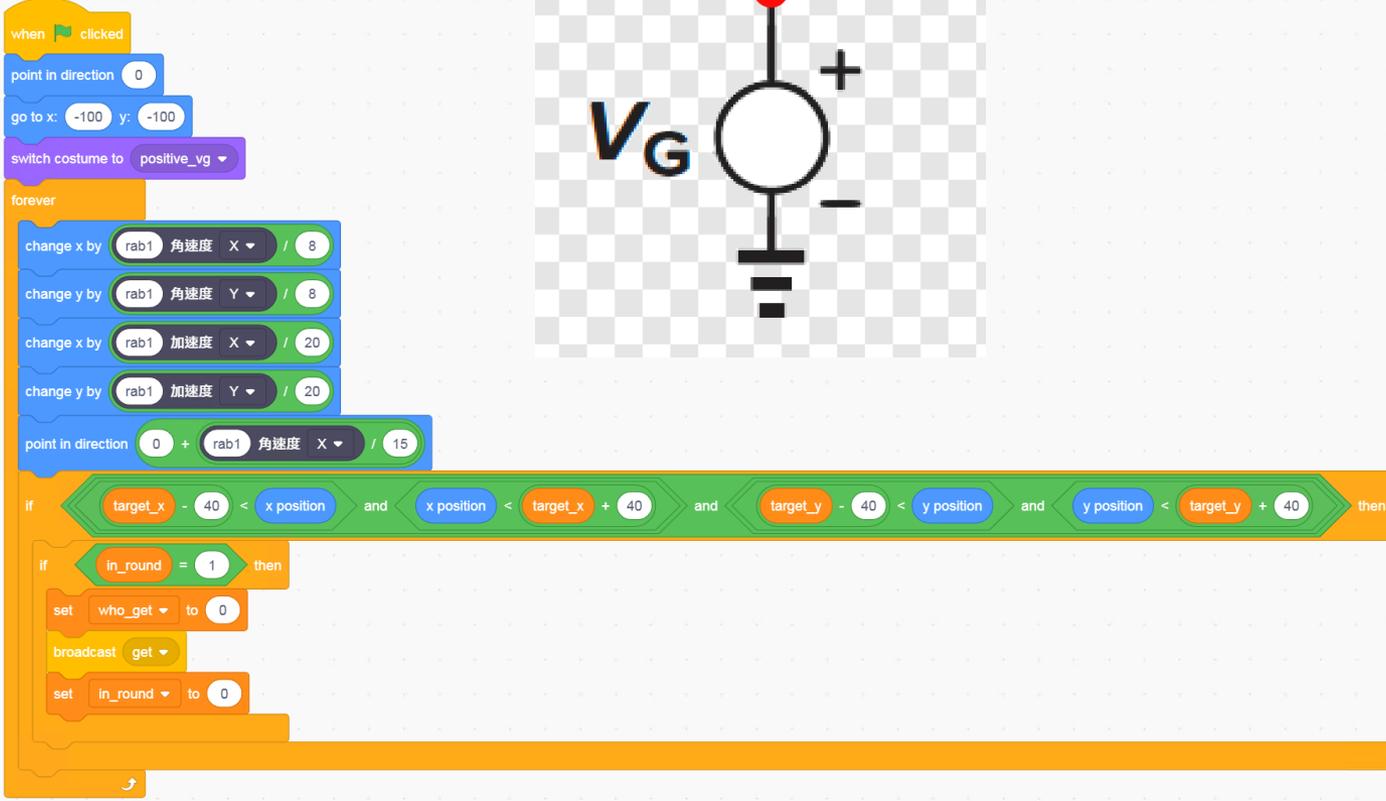




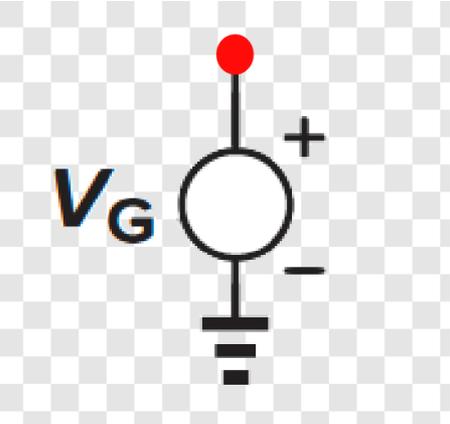
▶▶▶ 程式介紹



電壓1/2-程式碼



The image shows a Scratch script for animating a voltage source. The script starts with a 'when clicked' event, followed by 'point in direction 0', 'go to x: -100 y: -100', and 'switch costume to positive_vg'. A 'forever' loop contains four 'change by' blocks: 'change x by rab1 角速度 X / 8', 'change y by rab1 角速度 Y / 8', 'change x by rab1 加速度 X / 20', and 'change y by rab1 加速度 Y / 20'. This is followed by 'point in direction 0 + rab1 角速度 X / 15'. A large 'if' block checks for a collision: $target_x - 40 < x\ position$ and $x\ position < target_x + 40$ and $target_y - 40 < y\ position$ and $y\ position < target_y + 40$. If true, it sets 'who_get' to 0, broadcasts 'get', and sets 'in_round' to 0.



The diagram shows a voltage source symbol on a grid background. It consists of a circle with a red dot at the top and a battery symbol at the bottom. The label V_G is to the left of the circle. The top terminal is marked with a '+' sign and the bottom terminal with a '-' sign.

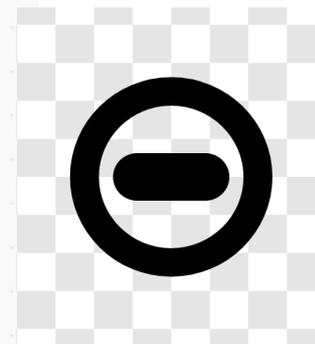
➤➤➤ +/-電荷-程式碼

當玩家控制的游標接觸到目標時對應行為。

```
when I receive get
  if hole = 1 then
    if who_get = 0 then
      change player1_score by -1
    else
      change player2_score by 1
  think positive charge (+)
  change ghost effect by 50
  wait 1.5 seconds
  hide
  clear graphic effects
  set hole to 0

when clicked
  set hole to 0
  hide

when I receive Hole
  set hole to 1
  show
  say hole !!
  go to x: target_x y: target_y
```



當接收到隨機的出現訊號時，移動到指定座標並顯現

元件1/2-程式碼

The image displays a Scratch script for an NMOS transistor simulation. The script is organized into several functional blocks:

- Initialization:** A "when I receive" block for "game_start" sets the "NMOS" variable to 1, shows the object, switches to the "NMOS_raw" costume, and moves to a specific x and y position.
- Click Event:** A "when clicked" block sets "NMOS" to 0 and hides the object.
- Score Logic:** A "when I receive" block for "get" triggers a series of "if" statements. Each "if" statement checks a condition on "player1_score" (e.g., = 0, < 0, = 1, = 2, = 3, = 4, = 5, = 6, = 7, > 7) and, if true, switches to a corresponding costume (costume1 through costume7).
- Termination:** A "when I receive" block for "game_stop" hides the object.

Below the code is a schematic diagram of an NMOS transistor. It shows a cross-section with a p-substrate, p+ source and drain regions, and an n+ gate region. A gate voltage V_G is applied to the gate, and a drain voltage of +0.1 V is applied to the drain. The diagram also illustrates the presence of electrons in the channel region.

背景-程式碼

```
when clicked
  set player1_score to 0
  set player2_score to 0
  set who_get to -1
  set cycle_count to 0
  switch backdrop to Xy-grid-20px
  broadcast Robo
  wait 4.5 seconds
  switch backdrop to Xy-grid-30px
  broadcast game_start

  repeat until (player1_score > 7 or player2_score > 7)
    set continue to 0
    set temp_rand to pick random 1 to 2
    set target_x to pick random -200 to 200
    set target_y to pick random -140 to 140
    wait 1.5 seconds
    if temp_rand = 1 then
      broadcast Hole
    if temp_rand = 2 then
      broadcast Electron
    set in_round to 1
    wait until continue = 1

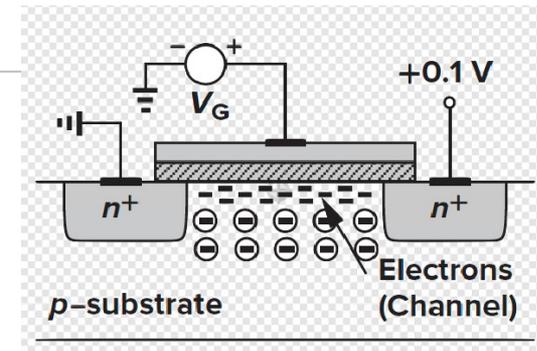
  broadcast game_stop
  switch backdrop to Stripes
  broadcast Channel
```

初始話設定

判斷獲勝條件與決定隨機出現目標

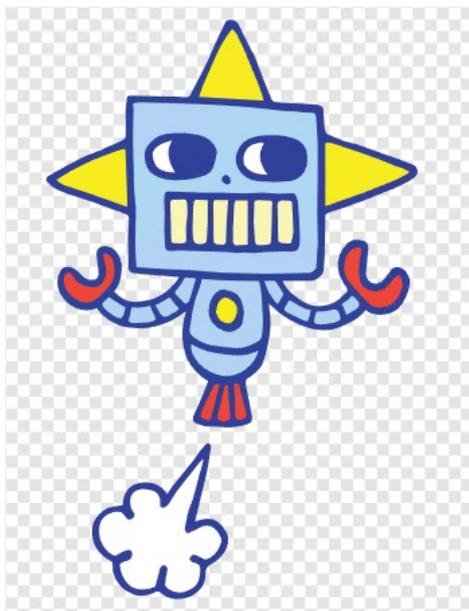
結尾畫面-程式碼

```
when I receive Channel  
go to x: 0 y: 0  
show  
if player1_score > player2_score then  
  switch costume to channel  
  say !!!! Player 1 win !!!!  
  wait 0.5 seconds  
  say NMOS channel formed.  
else  
  switch costume to PMOS_channel  
  say !!!! Player 2 win !!!!  
  wait 0.5 seconds  
  say PMOS channel formed.
```



由何者獲勝顯示不同畫面

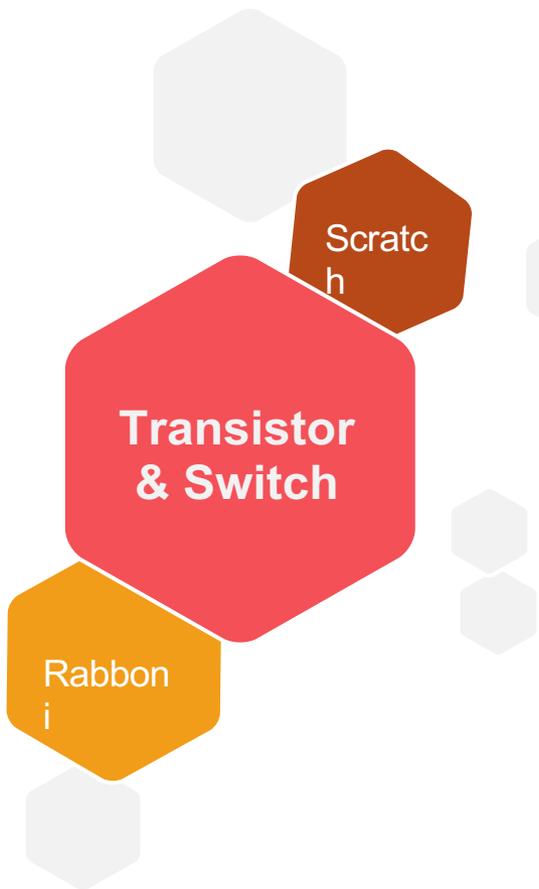
▶▶▶ 遊戲開始的機器人 -程式碼



```
when I receive Robo
go to x: 0 y: 0
show
say Let's pretend we are al transistors.
switch costume to robot-a
wait 0.5 seconds
switch costume to robot-b
wait 0.5 seconds
switch costume to robot-c
wait 0.5 seconds
switch costume to robot-d
wait 0.5 seconds
say Player 1 is NMOS, and Player 2 is PMOS.
switch costume to robot-a
wait 0.5 seconds
switch costume to robot-b
wait 0.5 seconds
switch costume to robot-c
wait 0.5 seconds
switch costume to robot-d
wait 0.5 seconds
hide
clear graphic effects
```



變換造型與說話



感謝聆聽!

