

# 計算機程式設計作業二 報告 機械二 許瀚中 B08502001

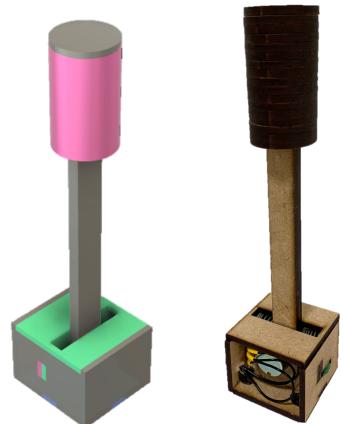
## 選擇題目之動機

根據上次作業製作遊戲的經驗，遊戲角色跳躍的高度是由鍵盤按壓時間決定的，但我想要能達到隨時能以類比的方式調整強弱作為控制的方式，因此這次試著設計能感測按壓力度的搖桿，並以序列埠連線的方式和電腦端的 Unity 溝通。

## 構想解說

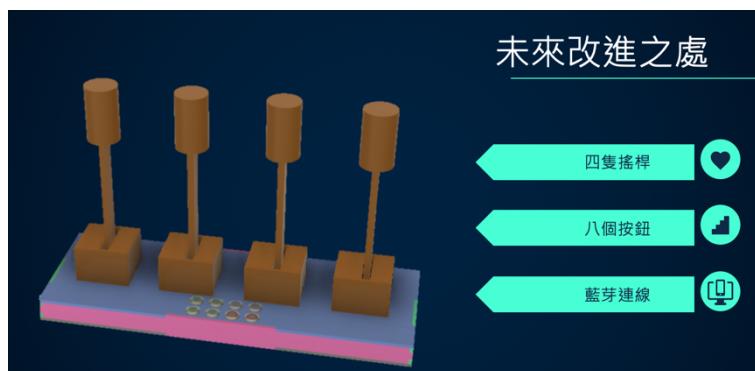
### (一) 搖桿：

搖桿的部分是用 Fusion360 繪圖軟體繪製而成，並藉由雷射切割的方式製作成型，裡面包含了一個可變電阻，使用 Arduino 微控制器讀取可變電阻電壓並與電腦進行序列連線，Arduino 的本身類比輸入有 1/1024 的解析度，因此非常適合用於力度與幅度的感測。



期末的報告預計會用到 4 支搖桿來模擬起重機的操作，實際的機器操作時的速度是會受到搖桿推的力度影響的，因此這次的搖桿製作也算是期末預計構想的提前準備與測試。

### (二) Unity 遊戲：



遊戲規劃：玩家會不斷往前移動，透過鍵盤操作（無力度感測）或是搖桿（有力度感測）來決定玩家向左或是向右的移動幅度。

角色建模：遊戲角色使用 Rhino 繪圖軟體建模，靈感來自 Youtube 影片：<https://youtu.be/072tU1tamd0>

### 製作過程碰到的問題

1. 可變電阻輸出的值在沒觸碰搖桿時仍會浮動：

因為 Arduino 的 ADC 有 1/1024 的解析度，就算可變電阻（搖桿角度）表面上看起來沒在動，電壓的值仍在跳動，導致遊戲的操控變得非常不準。

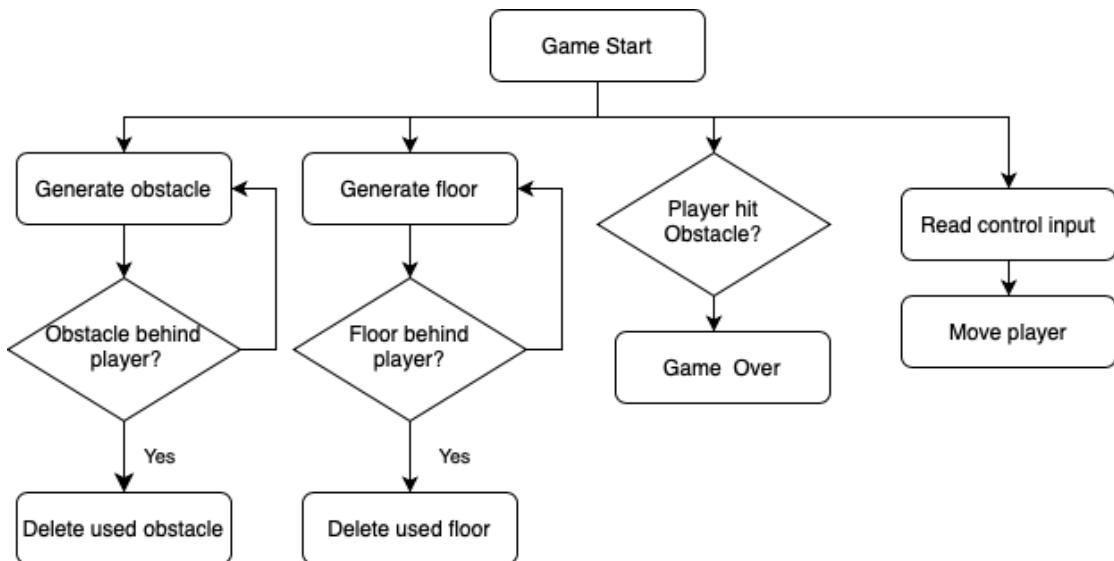
解決方法：在 Arduino 讀取輸入值的程式加入一個模糊的地帶不作出反應（例如：當輸入值介於 0~400 600~1024 時才作出反應，中間搖桿的自然晃動即不會被偵測到。

2. Arduino 序列連線輸出值的速度太快：

因為 Arduino Loop() 的輸出速度遠快於 Unity Update() 的速度，導致資料堆積沒被讀取，隨著遊戲的執行延遲越來越多。

解決方法：在 Arduino 的程式中加入 delay(20)，Unity 序列連線的部分 setTimeOut 設置為 25 能剛好以正常的速度讀取資料。

### 流程圖



## 程式列表

PlayerMove.cs：控制玩家向前移動的速度、讀取鍵盤、搖桿輸入讓玩家左右移動閃避障礙物、偵測玩家是否碰撞障礙物。

CameraMove.cs：讓攝影機跟隨玩家的後面移動。

FloorMove.cs：隨著玩家往前移動產生新的地面。

FloorRemove.cs：移除距離玩家過遠的地面。

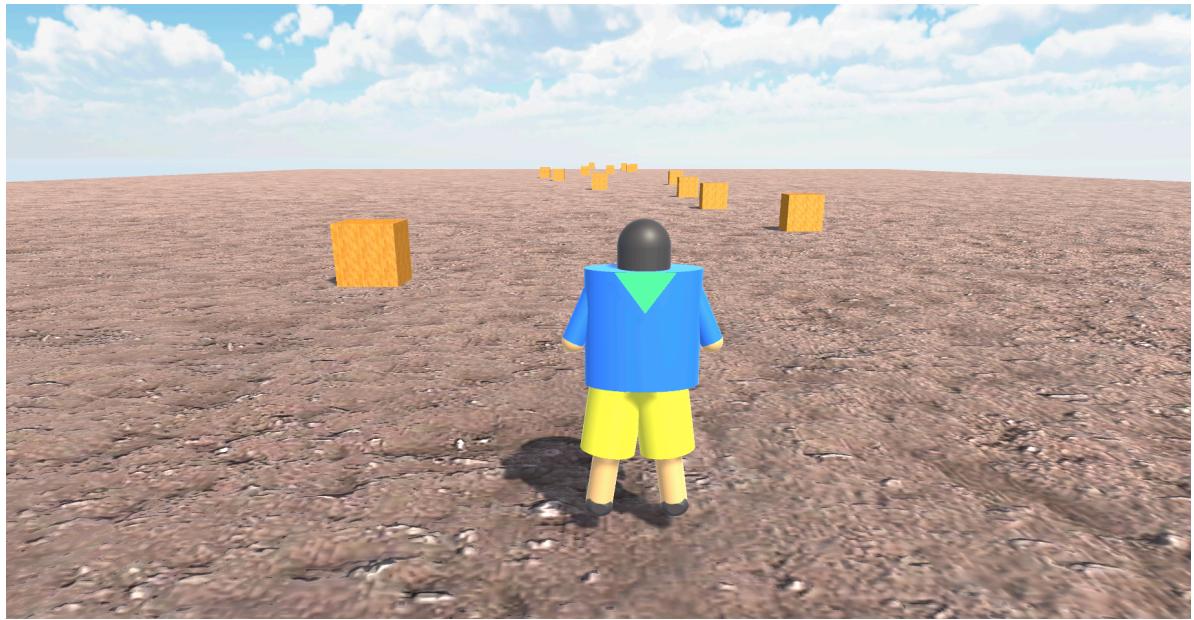
ObstacleRemove.cs：移除距離玩家過遠的障礙物。

ObstacleGenerate.cs：隨著玩家移動隨機產生新的障礙物。

## 程式操作方法

可透過搖桿操控（有道感測）、鍵盤 AD 鍵（左右）或是方向鍵左右操控。

## 程式測試執行結果



- 影片 demo 位於資料夾中 demo.mp4

備註：影片拍攝為螢幕錄影與手機拍攝搖桿動作剪接而成，可能會因為兩段影片時間軸對齊問題有些許延遲。

## 參考文獻

程式語法：<https://docs.unity3d.com/ScriptReference>

Arduino 和 Unity 的連線：

<https://create.arduino.cc/projecthub/raisingawesome/unity-game-engine-and-arduino-serial-communication-12fdd5>

剛體移動：<https://docs.unity3d.com/ScriptReference/Rigidbody-velocity.html>