

高雄市高英高級工商職業學校

Kao Ying Industrial Commercial Vocational High School

專題製作報告



作品名稱:新,套圈圈

組長:盧張仕叡

組員:林威任

指導老師:蔡忠憲

中華民國 108 年 5 月

目錄

壹、摘要.....	1
貳、研究動機與目的.....	2
一、研究動機.....	2
二、研究目的.....	2
參、理論探討.....	3
一、Arduino UNO 板.....	3
(一) 介紹.....	4
(二) 特色.....	4
(三) 軟體.....	5
二、四位數七段顯示器.....	6
(一)介紹.....	6
(二)構造.....	7
(三)控制.....	7
(四)原理.....	8
(五)七段顯示器分辨與區別.....	8
三、微動開關.....	9
(一)一般用語.....	9
1. 微動開關.....	9
2. 有接點.....	9
3. 接觸形式.....	9
4. 額定值.....	9
5. 樹脂黏著 (模壓端子).....	9
6. 絕緣阻抗.....	9
7. 耐電壓.....	9
8. 接觸電阻.....	9
9. 耐振性.....	10
10. 耐衝擊性.....	10
(二) 微動開關的組成、構造.....	10
四、麵包板.....	11
(一)介紹.....	11
(二)構造.....	11
(三)分類.....	12
1. 無焊麵包板.....	12
2. 單麵包板.....	12
3. 組合麵包板.....	12
肆、研究方法.....	13

一、	流程圖.....	13
二、	結構圖.....	14
三、	製作材料.....	15
	(一)使用材料.....	15
	(二)使用器材.....	15
四、	程式碼.....	15
五、	製作過程.....	17
伍、	研究結果.....	18
陸、	結論.....	19
	一、討論.....	19
	二、心得.....	19
柒、	參考資料.....	20

圖目錄

圖 1 夜市場常見的套圈圈.....	1
圖 2 ArduinoUNO.....	5
圖 3 Arduino 軟體介面.....	6
圖 4 七段顯示器.....	6
圖 5 七段顯示器的排序.....	7
圖 6 七段顯示器所有的狀態.....	7
圖 7 微動開關的構造.....	10
圖 8 長針微動開關.....	11
圖 9 無焊麵包板.....	11
圖 10 組合麵包版.....	12
圖 11Arduino 編寫程式.....	17
圖 12 電路製作.....	17
圖 13 按鈕測試.....	17
圖 14 微動開關測試.....	17

表目錄

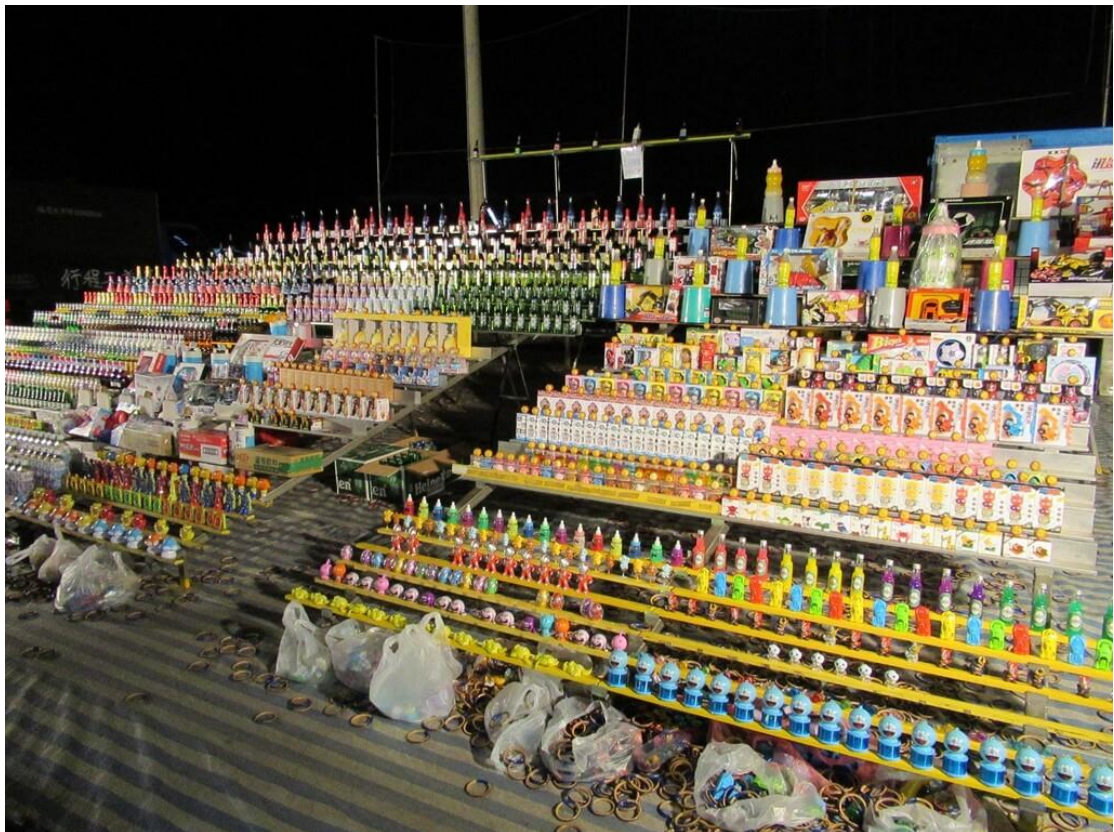
表 1 Arduino 型號規格表.....	4
表 2 七段顯示器對照表.....	8
表 3 材料表.....	15
表 4 使用器材.....	15

壹、摘要

在平常的晚上，夜市場的食物、生活必需品和小遊戲都開始慢慢開張，其中夜市場小遊戲套圈圈是其中很好玩的遊戲之一，我們腦中在看的同時也想把套圈圈打造一台帶入一點數位遊戲風格，變成一個讓別人看到不一樣的套圈圈。

我們的作品採用了 Arduino UNO 控制板來控制所有的感測器，我們利用微動開關知道圈圈是否有投入柱子上，再利用七段顯示器來確認圈圈有沒有確實投入且顯示分數在七段顯示器上，最後設置一個按鈕來重新開始遊戲。

套圈圈基台有九個柱子，讓玩家拿著圈圈往柱子的方向投擲並嘗試獲得分數，站在一定的距離進行投擲，起始值為十分，若九個柱子都沒有任何一個圈圈進去，十分是基本分數。



(圖 1 夜市場常見的套圈圈)

貳、研究動機與目的

一、研究動機

我們在夜市場都會看到很多遊戲設施給客人們遊玩，其中的小遊戲套圈圈是其中很好玩的遊戲之一，我們腦中在看的同時也想要把套圈圈打造一台帶入一點數位遊戲風格，我們就想到了一個點子，就是把套圈圈改成可以給人看到分數且可以打個交流或趣味競賽。

而且想要玩的時候也不用出去家門找夜市場有沒有套圈圈再裡面，自己簡單的做一台就能有那種感覺。自己訂定規則、遊玩方式、過程中拿到幾分可以拿到甚麼獎勵。

會有這樣的想法是因為我目前看到的套圈圈都是很單純的，所以想要來增加一些數位科技進去來試試看一些新玩法。讓玩家拿著圈圈往柱子的方向投擲並嘗試獲得分數，站在一定的距離進行投擲，起始值為十分，若九個柱子都沒有任何一個圈圈進去，那麼十分就是基本分數。

二、研究目的

- (一)不會耗費成本太多，只需簡易的材料即可。
- (二)增加攜帶方便性，而且只需用行動電源等小型設備來運作。
- (三)讓管理員縮小監控範圍也容易整理，不需頻繁移動也避免遊客投機取巧。
- (四)想要創造大家一起同樂的氣氛，休息時間的時候讓大家可以暫時放下手邊的事情來遊玩套圈圈放鬆一下。

參、理論探討

一、Arduino Uno 板

(一)介紹

Arduino 是一家製作開源硬體和軟體的公司，同時兼有專案和用戶社群，該公司負責設計和製造單板微控制器和微控制器套件，用於構建數位裝置和互動式物件，以便在物理和數位世界中感知和控制物件。該專案的產品是按照 GNU 寬通用公共許可證 (LGPL) 或 GNU 通用公共許可證 (GPL) 許可的開源硬體和軟體分發的，Arduino 允許任何人製造 Arduino 板和軟體分發。Arduino 板可以以預裝的形式商業銷售，也可以作為 DIY 套件購買。

Arduino 電路板設計使用各種微處理器和控制器。這些電路板配有一組數字和類比 I/O 引腳，可以連接各種擴充板或麵包板（封鎖板）和其他電路。這些電路板具有串列埠，包括某些型號上的通用串列匯流排 (USB)，也用於從個人電腦載入程式。微控制器通常使用 C/C++ 程式語言。除了使用傳統的編譯工具鏈之外，Arduino 專案還提供了一個基於 Processing 語言專案的整合式開發環境。

Arduino 專案始於 2003 年，作為義大利伊夫雷亞地區伊夫雷亞互動設計研究所的學生專案，目的是為新手和專業人員提供一種低成本且簡單的方法，以建立使用傳感器與環境相互作用的裝置執行器。適用於初學者愛好者的此類裝置的常見範例包括簡單機器人、恆溫器和運動檢測器。

Arduino 這個名字來自義大利伊夫雷亞的一家酒吧，該專案的一些創始人過去常常會去這家酒吧。酒吧以伊夫雷亞的 Arduin 命名，他是伊夫雷亞邊疆伯爵，也是 1002 年至 1014 年期間的義大利國王。

它使用 Atmel AVR 單片機，採用開放原始碼的軟硬體平台，構建於開放原始碼 simple I/O 介面，並具有使用類似 Java、C 語言的 Processing/Wiring 開發環境。

(表 1 Arduino 型號規格表)

型號	Uno	Leonardo	Due	Mega 2560	Mega ADK
微控晶片	ATmega328	ATmega32u4	AT91SAM3X8E	ATmega2560	ATmega2560
工作電壓	5V	5V	3.3V	5V	5V
輸入電壓	7-12V	7-12V	7-12V	7-12V	7-12V
Digital I/O	14	20	54	54	54
PWM	6	7	12	15	15
Analog input	6	12	12	16	16
Analog output	-----	-----	2	-----	-----
Clock rate	16MHz	16MHz	84Mhz	16MHz	16MHz
Serial port (UART)	1組	1組	4組	4組	4組
External interrupts	5組	5組	可指定所以有可用的pin	6組	6組
SPI	有	有	有	有	有
I2C	1組	1組	2組	1組	1組
Flash memory	32KB	32KB	512KB	256KB	256KB
SRAM	2KB	2.5KB	96KB	8KB	8KB
EEPOM	1KB	1KB	-----	4KB	4KB
面積大小 (cm x cm)	6.9x 5.3	6.9x 5.3	10.2 x 5.3	10.2 x 5.3	10.2 x 5.3
USB Host	-----	-----	-----	-----	有

(二)特色

基於創用 CC 開放原始碼的電路圖設計。免費下載，也可依需求自己修改，但需遵照姓名標示。您必須按照作者或授權人所指定的方式，表彰其姓名。Arduino 可使用 ICSP 線上燒入器，Bootloader 燒入新的 IC 晶片。可依據 Arduino 官方網站，取得硬體的設計檔，加以調整電路板及元件，以符合自己實際設計的需求。可簡單地與感測器，各式各樣的電子元件連接，如紅外線、超音波、熱敏電阻、光敏電阻、伺服馬達等。支援多樣的互動程式，如 Adobe Flash，Max/MSP，VVVV，Pure Data，C，Processing 等。使用低價格的微處理控制器 (Atmel AVR) (ATMEGA 8, 168, 328 等)。USB 介面，不需外接電源。另外有提供直流 (DC) 電源輸入。



(圖 2 Arduino UNO)

(三)軟體

在 Arduino 上執行的程式可以使用任何能夠被編譯成 Arduino 機器碼的程式語言編寫。而 Atmel 也提供了數個可以開發 Atmel 微處理機程式的整合開發環境，AVR Studio 和更新 AtmelStudio。

目前微軟在其 Visual Studio 也有提供 Arduino 的 SDK，在編譯執行上更方便。

而 Arduino 計劃也提供了 Arduino Software IDE，一套以 Java 編寫的跨平台應用軟體。Arduino Software IDE 源自於 Processing 程式語言以及 Wiring 計劃的整合開發環境。它是被設計於介紹程式編寫給藝術家和不熟悉程式設計的人們，且包含了一個擁有語法突顯、括號符合、自動縮排和一鍵編譯並將執行檔燒寫入 Arduino 硬體中的編輯器。

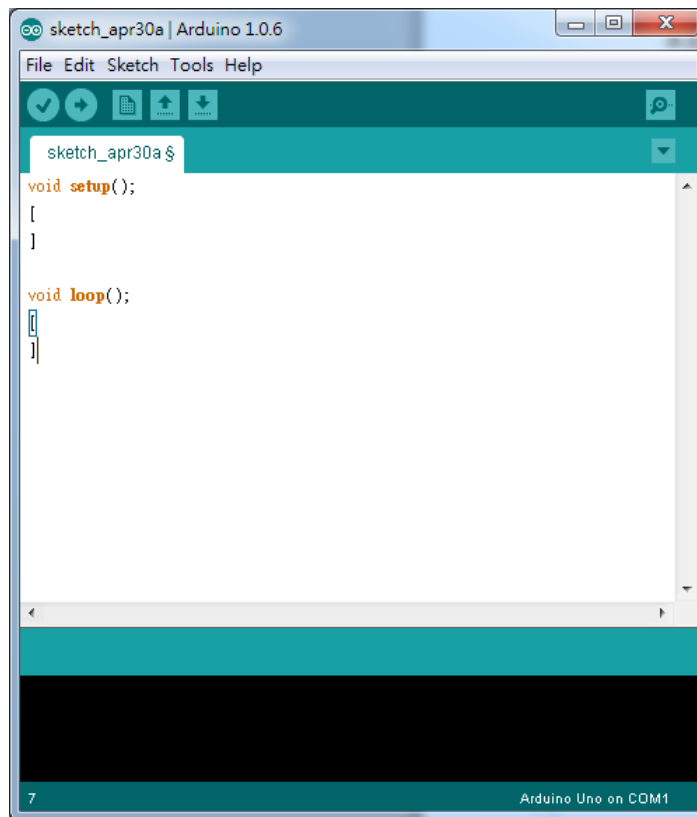
Arduino Software IDE 使用與 C 語言和 C++相仿的程式語言並且提供了包含常見的輸入/輸出函式的 Wiring 軟體函式庫。

在使用 GNUtoolchain 編譯和連結後，Arduino Software IDE 提供了一個程式，avrdude 用來轉換可執行檔成為能夠燒寫入 Arduino 硬體的韌體。

Arduino Software IDE 編寫的程式被稱為「sketch」。一個典型的 Arduino C/C++ sketch 程式會包含兩個函式，它們會在編譯後合成為 main() 函式：

setup()：在程式執行開始時會執行一次，用於初始化設定。

loop()：直到 Arduino 硬體關閉前會重複執行函式放的程式碼。



(圖 3 Arduino 軟體介面)

二、四位數七段顯示器

(一)介紹

為常用顯示數字的電子元件。因為藉由七個發光二極體以不同組合來顯示數字，所以稱為「七劃管」、「七段數碼管」、「七段顯示器」，由於所有燈管全亮時所表示的是「8」，所以又稱「8字管」、「8字顯示器」。

多數七段顯示器還會在右下角附加一個表示小數點的燈管，因此也稱八段管。

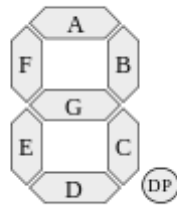


(圖 4 七段顯示器)

(二)構造

一般的七段顯示器擁有八個發光二極體（三橫四縱）用以顯示十進位 0 至 9 的數字外加小數點，也可以顯示英文字母，包括十六進位中的英文 A 至 F（b、d 為小寫，其他為大寫）。

除了七段顯示器外，還有十四及十六劃等添加額外斜向筆劃的顯示器；但由於點陣顯示器（英語：Dot-matrix）價格的下跌，這些「多劃管」已基本上被後者取代。

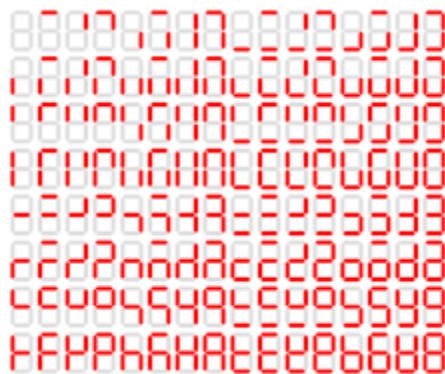


(圖 5 七段顯示器的排序)

(三)控制

七段顯示器分為共陽極及共陰極，共陽極的七段顯示器的正極（或陽極）為八個發光二極體的共有正極，其他接點為獨立發光二極體的負極（或陰極），使用者只需把正極接電，不同的負極接地就能控制七段顯示器顯示不同的數字。共陰極的七段顯示器與共陽極的只是接駁方法相反而已。

七段顯示器已可以特定的集成電路控制，只要向集成電路輸入 4-bit 的二進位數字訊號就能控制七段顯示器顯示；市面上更有 8421-BCD 代碼直接轉為七劃管控制電平的 IC，方便配合單片機使用。



(圖 6 七段顯示器所有的狀態)

(四)原理

共陰極七段顯示器原理其實和 LED 是一樣的，我們控制的接腳如果送電，電流就會經過 LED 和電阻到達接地端，形成電源迴路，於是 LED 就會發光，如果不送電，就會因為沒有電壓落差(壓降)，所以電流不會流過去，因此 LED 不會發亮。

由這七個筆畫的亮與不亮來決定所顯示的數字，例如：1 就是 b、c 要亮，所以我們只要把 D6、D9 送電就行了，所以顯示的每一個數字其實就是這七個筆畫所對應的腳位送不送電的結果。因為每次要顯示一個數字，都要動到七個腳位，因此我們會習慣先做一張對照表，對照表中 1 就是送電，0 就是不送電，對照表如圖：

(表 2 七段顯示器對照表)

Arduino腳位D0~D13	D5	D6	D9	D10	D11	D12	D13
7段顯示器對應筆畫代號	a	b	c	d	e	f	g
7段顯示器顯示的數字	對應腳位控制送電或不送電						
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0
2	1	1	0	1	1	0	1
3	1	1	1	1	0	0	1
4	0	1	1	0	0	1	1
5	1	0	1	1	0	1	1
6	1	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1
E	1	0	0	1	1	1	1

(五) 七段顯示器分辨與區別

七段顯示器主要有兩種規格，分為共陽極與共陰極，共陽極規格必須使用“Sink Current”方式，由單晶片產生低電位的腳位，外部電源流經七段顯示器，再流入單晶片；共陰極規格則使 Source Current 方式，由單晶片輸出高電位的腳位，提供七段顯示器電流，且必須注意單晶片所能提供的電流大小。七段顯示器內部由每節至少 1 個或更多個 LED 晶片組成，在製作過程中 LED 的陽極(P 極)如果全部接在一起，那就是共陽；相反如果 LED 的陰極(N 極)如果全部接在一起，那就是共陰。

一、微動開關

(一)一般用語

1. 微動開關：

具備微小接點間隔及瞬動結構，且以外殼包覆使用指定動作及力量即可執行接點開關動作的接點結構，及外部具備致動器的小型開關。(以下稱為開關)

2. 有接點：

係指開關之中，被拿來與具備開關特性的半導體開關比較對照，並以接點的機械性開閉執行開關功能之類別。

3. 接觸形式：

根據各種用途所構成之接點的電氣性輸出輸入迴路。

4. 額定值：

一般為開關的特性及性能保證標準值，比方說額定電流、額定電壓等，其前提是必須符合特定條件(負載種類、電流、電壓、頻率等)。

5. 樹脂黏著(模壓端子)：

係指將導線配置於端子部位後，此部位會因為充填樹脂而固化，阻隔暴露的充電部位並提高防滴功能。

6. 絕緣阻抗：

係指非連續端子間、各端子及非充電金屬部位之間、各端子及接地之間的阻抗值。

7. 耐電壓：

於規定的量測位置施加1分鐘高電壓時，不會使絕緣部份被破壞的極限值。

8. 接觸電阻：

係指接點接觸部分的電阻，不過一般包含彈簧或端子部分體阻抗的數值。

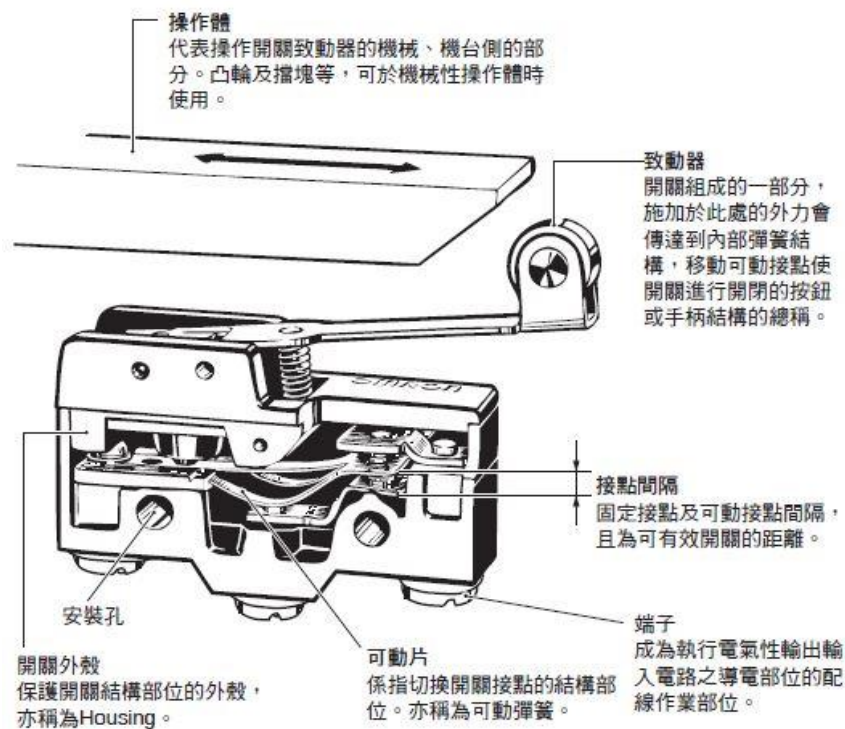
9. 耐振性：

誤動作振動…係指微動開關於使用中，不會使閉路接點斷開超過規定時間，範圍內的振動。

10. 耐衝擊性：

耐久衝擊係指微動開關在運送中或安裝時不會讓各部位產生損傷，可滿足動作特性，在機械性衝擊範圍內的衝擊。
誤動作衝擊…係指微動開關於使用中，不會使閉路接點斷開超過規定時間，範圍內的衝擊。

(二)微動開關的組成、構造



(圖 7 微動開關的構造)

具備著微小接點間隔及瞬動結構外殼包覆使用動作即執行接點開關動作的接點結構，及外部具備致動器的小型關。係指開關之中，被拿來與具備開關特性的半導體開關比較對照並以接點的機械性開閉執行開關功能之類別。根據各種用途所構成之接點的電氣性輸出輸入迴路。



(圖 8 長針微動開關)

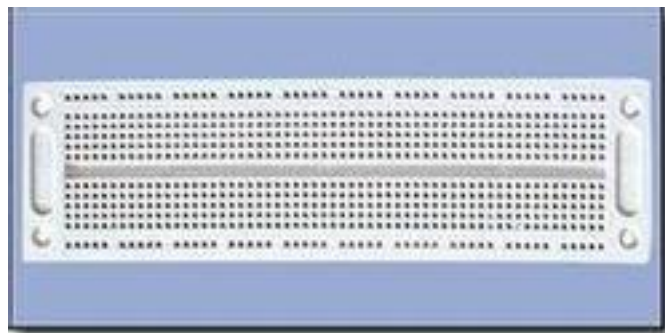
二、麵包板

(一)介紹

麵包板是由於板子上有很多小插孔，很像麵包中的小孔，因此得入孔中時能夠與金屬條接觸，從而達到導電目的。一般將每5個孔板用名，專為電子電路的無焊接實驗設計製造的。由於各種電子元器件可根據需要隨意插入或拔出，免去了焊接，節省了電路的組裝時間，而且元件可以重複使用，所以非常適合電子電路的組裝、調試和訓練。

(二)構造

整板使用熱固性酚醛樹脂製造，板底有金屬條，在板上對應位置打孔使得元件插一條金屬條連線。板子中央一般有一條凹槽，這是針對需要積體電路、晶片試驗而設計的。板子兩側有兩排豎著的插孔，也是5個一組。這兩組插孔是用於給板子上的元件提供電源。母板使用帶銅箔導電層的玻璃纖維板，作用是把無焊麵包板固定，並且引出電源接線柱。



(圖 9 無焊麵包板)

(三) 分類

1. 無焊麵包板

就是沒有作為底座的母板，沒有焊接電源插口引出但是能夠擴展單麵包板的板子。使用時應該先通電。將電源兩極分別接到麵包板的兩側插孔，然後就可以插上元件實驗了(插元件的過程中要斷開電源)。遇到多於5個元件或一組插孔插不下時，就需要用麵包板連線線把多組插孔連線起來。無焊麵包板的優點是體積小，易攜帶，但缺點是比較簡陋，電源連線不方便，而且面積小。不宜進行大規模電路實驗。若要用其進行大規模的電路實驗，則要用螺釘將多個麵包板固定在大木板上，再用導線相連線。

2. 單麵包板

就是有母板作為底座，並且電源接入有專用接線柱，甚至有些能夠進行高壓實驗的還有地線接線柱的麵包實驗板。這種板子使用起來比較方便，就是把電源直接接入接線柱，然後插入元件進行實驗(插元件的過程中要斷開電源)遇到多於5個元件或一組插孔插不下時，就需要用麵包板連線線(也叫麵包線，見參考資料)把多組插孔連線起來。單麵包板的優點是體積較小，易攜帶，可以方便的通斷電源，但缺點是面積小，不宜進行大規模電路實驗。

3. 組合麵包板

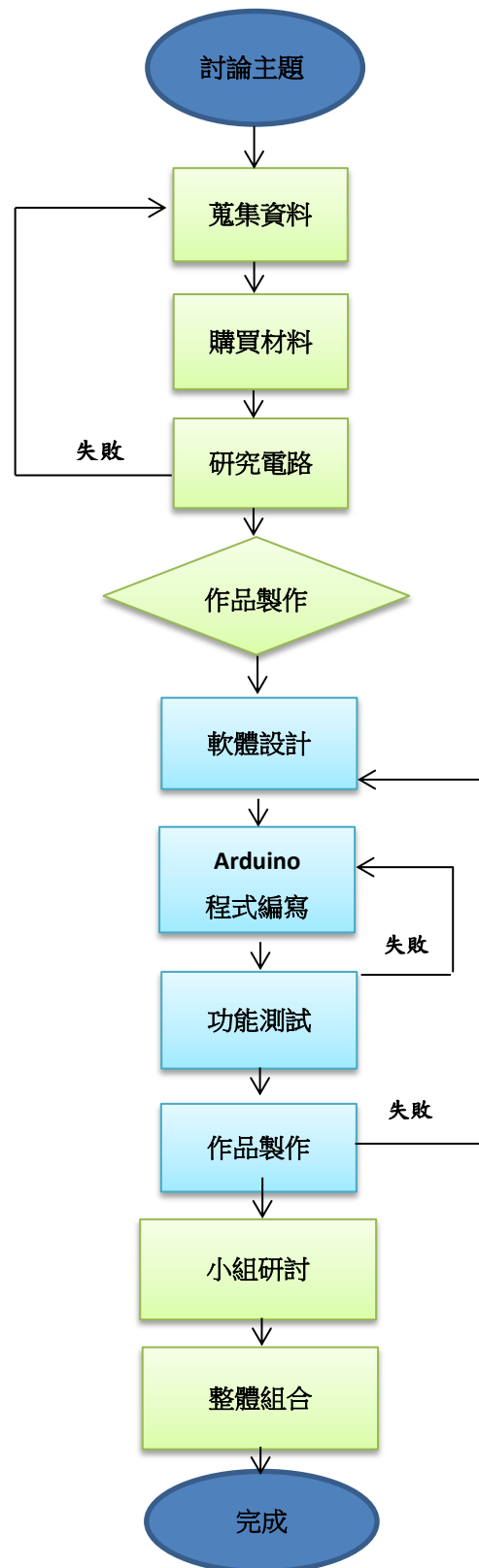
顧名思義就是把許多無焊麵包板組合在一起而成的板子。一般將2-4個無焊麵包板固定在母板上，然後用母板內的銅箔將各個板子的電源線連在一起。專業的組合麵包板還專門為不同電路單元設計了分電源控制，使得每塊板子可以根據用戶需要而攜帶不同的電壓。組合麵包板的使用與單麵包板相同。



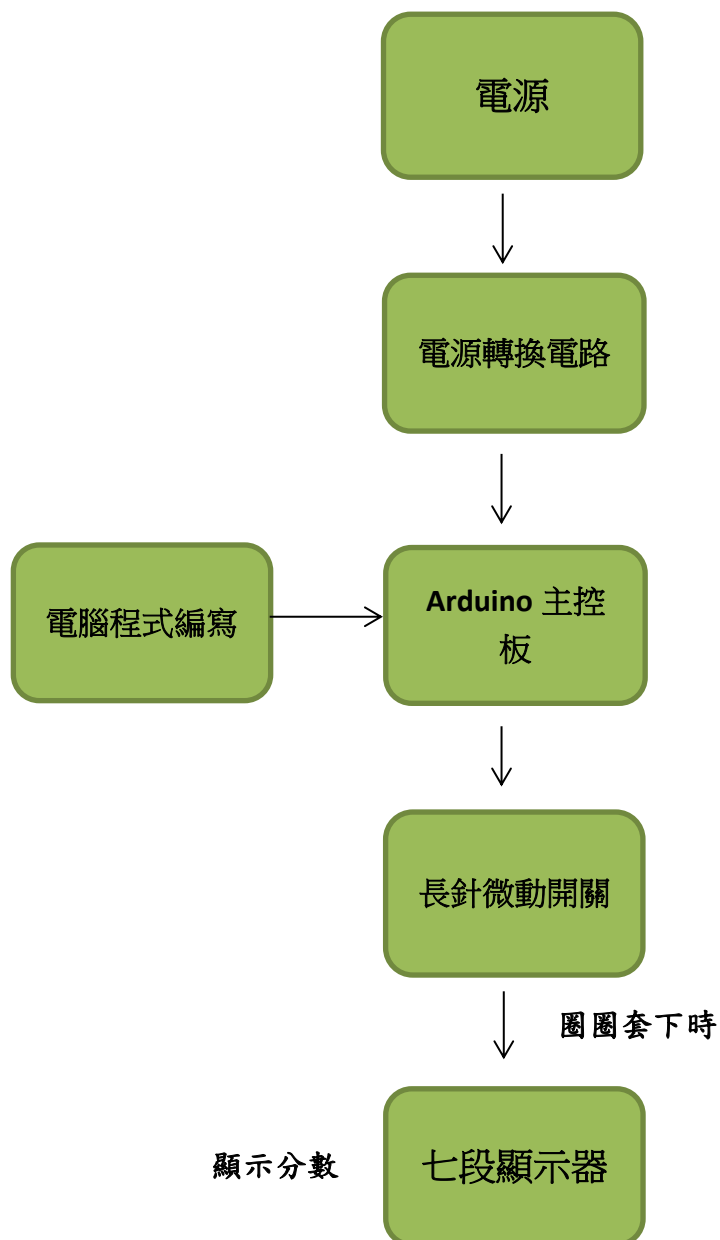
(圖 10 組合麵包版)

肆、研究方法

一、流程圖



二、架構圖



三、製作材料

(一)使用材料

(表 3 材料表)

材料名稱	數量	備註
木板	*1	
竹管	*9	
Arduino Uno	*1	
七段顯示器模組	*1	TM1637
按鈕	*1	
長針型微動開關	*9	
水管	*1	
杜邦線	*1	公公、公母

(二)使用器材

(表 4 使用器材)

器材名稱	數量	備註
桌上型電腦	*1	
Arduino 開發程式	*1	1.6.X 版
手電鑽	*1	
鋸子	*1	
烙鐵	*1	
麵包板	*1	

四、程式碼

```
#include <TM1637Display.h> //載入 TM1637 函式庫
#define CLK 2 //定義七段顯示器模組 CLK 號腳到 2
#define DIO 3 //定義七段顯示器模組 DIO 腳位到 3
TM1637Display display(CLK, DIO); //四位數七段顯示器腳位
define Reset 5 //重置微動開關按鈕腳位
int Button[]={13, 12, 11, 10, 9, 8, 7, 6, 5}; //微動開關按鈕腳位
int fraction=10; //紀錄當前分數
void setup() { //程式初始化
```

```

Serial.begin(9600);           //設定鮑率
display.setBrightness(0x0f); //設定四位數七段顯示器的亮度
for (int i=0;i<9;i++) pinMode(Button[i], INPUT); //設定所有按鈕為輸入
Serial.println(fraction);    //顯示現在分數
display.clear();            //重置四位數七段顯示器
display.showNumberDec(fraction, false); //七段顯示器顯示當前分數
}
void loop() {                //重複迴圈
for (int i=0;i<8;i++){      //判斷八個按鈕哪一個被觸發
  if (digitalRead(Button[i])==1){ //讀取當前按鈕是否被觸發
    if (i==0)fraction=fraction+10; //設定觸發後所更動的分數
    if (i==1)fraction=fraction+10; //設定觸發後所更動的分數
    if (i==2)fraction=fraction+10; //設定觸發後所更動的分數
    if (i==3)fraction=fraction+10; //設定觸發後所更動的分數
    if (i==4)fraction=fraction+10; //設定觸發後所更動的分數
    if (i==5)fraction=fraction+10; //設定觸發後所更動的分數
    if (i==6)fraction=fraction+10; //設定觸發後所更動的分數
    if (i==7)fraction=fraction+10; //設定觸發後所更動的分數
    if (i==8)fraction=fraction+10; //設定觸發後所更動的分數
    Serial.println(fraction);      //顯示當前分數
    display.clear();              //清除四位數七段顯示器的分數
    display.showNumberDec(fraction, false); //顯示四位數七段顯示器的分數
    delay(25); //延遲 0.25 秒
    while (digitalRead(Button[i])==1){ //等待按鈕離開
      delay(10); //延遲 0.1 秒
    }
  }
}
}
if (digitalRead(Reset)==1){ //偵測按鈕按下時是否觸發
  fraction=0; //重置分數
  Serial.println(fraction); //顯示重置後的分數
  display.clear(); //清除七段顯示器內容
  display.showNumberDec(fraction, false); //顯示重置後的分數
  delay(25); //延遲 0.025 秒
}

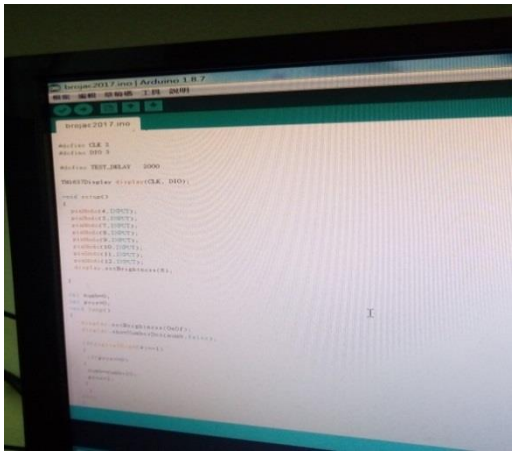
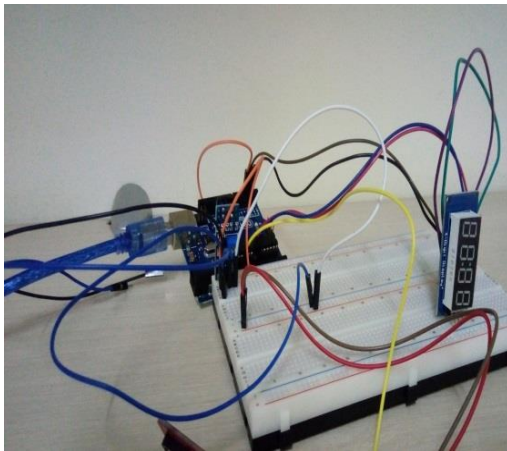
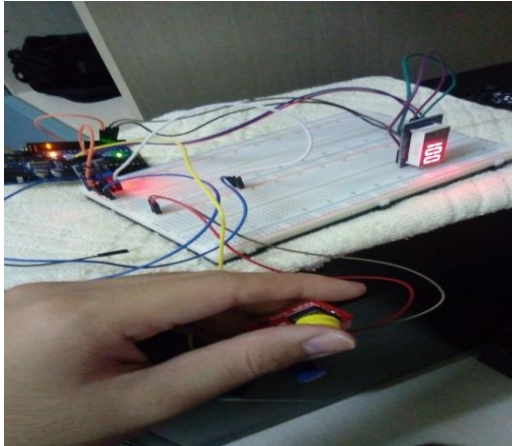
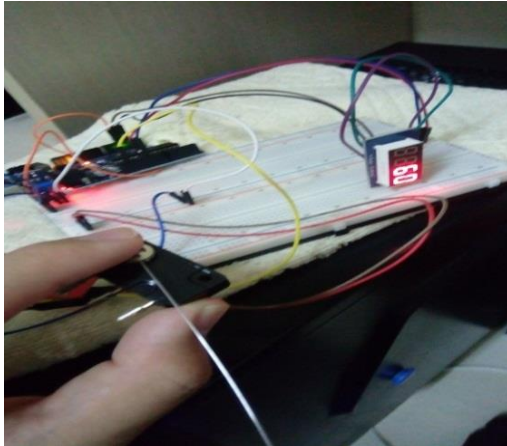
```

```

While(digitalRead(Reset)==1){ //判斷按鈕離開
  delay(10); //延遲 0.01 秒
}
}
}
}

```

五、製作過程

	
<p>(圖 11 Arduino 編寫程式)</p>	<p>(圖 12 電路製作)</p>
<p>我們藉由遊戲的遊玩方式來編輯我們的程式來實現，設計類似籃球投進加分的元素來呈現我們的套圈圈。</p>	<p>寫完程式後，開始組合顯示器、元件，依照程式順序的配合，來接上零件</p>
	
<p>(圖 13 按鈕測試)</p>	<p>(圖 14 微動開關測試)</p>
<p>測試按鈕按下分數歸位是否為正常使用於遊戲結束時按下即可分數重洗。</p>	<p>用來感測圈圈投入開關，七段顯示器配合微動開關的觸碰接收來顯示分數。</p>

伍、研究結果

一、七段顯示器

當圈圈丟進去柱子裡的時候經過開關所發出的輸入信號之段顯示器模組就會藉由開關的觸發後再經過程式碼判斷，就可以計算加分顯示在上面。比賽結束按鈕按下時，把目前顯示的分數重置回十分基本分數。

二、微動開關

微動開關有分三個角，NO、NC 和 C，其中 NO 接負、C 接正、NC 空接只有這樣才能讓微動開關觸發利用 圈圈投入時方便感測採用了微動開關進行與圈圈接觸，在圈圈投進柱子後開關能夠確實判斷到圈圈已進入柱子，但是也是有一定的低機率造成誤判，比如說圈圈沒有投進但是壓到開關的別處、圈圈卡在柱子上、彈到別的柱子誤觸.....等。

三、木頭製作

在多次討論後，決定使用傳統套圈圈的架構概念來組裝，發現用階梯式的話柱子的周圍空間會有些狹窄導致圈圈無法投入的可能，經過數次的修補可以讓圈圈不會被設計上的問題而無法投入。

四、套圈圈的優缺點

(一)優點:

- (1)當圈圈套進去時,與開關接觸可以顯示分數
- (2)在空間上有一定的限制可以讓圈圈投入

(二)缺點:

- (1)雖然有套進去,但是也有彈出或卡住等可能
- (2)在空間上也有圈圈沒有套進柱子誤觸開關造成誤判

陸、結論

一、討論

我們在過程中了解到自己的製作能力和思考上有待加強，像是為甚麼我們沒有想到、怎麼沒注意到可以這樣製作導致花費不必要的金錢。

當中有些部分問題我們透過這次的專題製作我們有學到課程以外的專業知識和校內所學的實習來製作我們的作品。也有很多問題也是我們沒有遇到或沒辦法去解決，可以請教老師來指導我們可以怎麼去解決這類的問題。

在專題製作中我們也知道說一個作品並不是一個人去構想，而是一個團隊一起建構一個作品，每個人的想法和建議何在一起，再運用我們課堂上學的跟團隊裡的想法做通用。而且還可以在過程中發現自己部分能力不足需要稍補強。也更加深了專題製作的意義，也是一種訓練，以後也可能有類似的工作是關於專題製作的元素。

我們在製作一份專題報告中，以前不管是國中國小我們在做的時候都是隨便打打、貼一貼，交出去就好了，可是那只是給我們一些概念去做的，真正的報告是有要求格式、整齊度、詳細、明確，是有一定的要求。所以在這裡也見識到了"報告"在任何的場合都扮演著重要的角色。

二、心得

這次的專題運用自身的創意來做出很好玩的作品，雖然我們的作品跟構造得有些落差或者是做出來的結果跟想像中不一樣，但是學到了我們在專題當中學習到學校之外的知識領域以及內容當中的觀念，在課程裡也學習到了團體合作。製作了這個專題從中間學到很多事情，如何團隊、分工、並一起解決問題。

我覺得這個過程是個很重要的一段時刻，雖然結果的不是很ok但是在製作的時候有吸收到不少的經驗，我相信我能在這當中找到一條能讓我看到成功的道路。

柒、參考資料

- 一、Arduino 介紹，取自於：
<https://zh.wikipedia.org/wiki/Arduino>
- 二、Arduino 軟體，取自於：
<https://www.arduino.cc/en/main/software>
- 三、按鍵計數，取自於：
<https://kknews.cc/zh-tw/digital/8eoebjl.html>
- 四、洪國勝(2019/04)。全民自造與程式設計 使 Arduino。
台灣:泉勝出版有限公司。
- 五、柯博文(2014/08)。Arduino 互動設計專題與實戰。
台灣: 碁峰。
- 六、趙英傑(2016/12)。超圖解 Arduino 互動設計入門。
台灣:旗標。
- 七、Mark Geddes(2015/07)。Arduino 自作專案大百科。
台灣:碁峰。
- 八、Don Wilcher(2014/11)。Make: 簡易 Arduino 專題製作。
台灣:歐萊禮
- 九、郭子榮(2017/08)。木工職人基礎手工具。
台灣:雄獅美術。
- 十、〈木工從新手到高手〉編委會(2014/05)。木工從新手到高手。
台灣:機械工業出版社。