

臺北市立大安高級工業職業學校

電子科專題報告

光觸控時鐘

Optical Touch Control Clock

學生

組長：黃鈴漪

組員：郭祐寧

組員：胡辰瀚

組員：許哲綸

指導老師：王村益 老師

中華民國 107 年 1 月

摘要

發光二極體(Light-emitting diode, 簡稱 LED), 是一種能發光的半導體電子元件, 因能量轉換效率高、壽命長、體積小等優點, LED 的照明及顯示功能已大量運用於生活中, 然而觀察現代人對產品的功能需求, LED 卻只應用於視覺的輸出, 不免為其發展飽和感到唏噓不已, 此時若能進一步的開發、拓展 LED 的使用便成為我們探討的目標。

本專題研究發光二極體最根本的物理現象—光伏效應, 觸控 LED 後手指反射的光線照射回其上, 利用其逆向偏壓產生的微小電流, 結合放大電路與軟體的控制, 判讀電壓的變化量, 達到光觸控的效果, 並使 RGB LED 與時鐘組合, 以提高研究的實用性, 而其中之「光」與「時間」產生的碰撞, 就如同最初日晷的發明一樣, 光觸控時鐘—將兼具時代的潮流的色彩炫麗及科技化的觸控互動, 又不失時鐘最初的單純。

關鍵字：Arduino, 光觸控, 光伏效應

Abstract

Light-emitting diode (LED) is a kind of semiconductor electronic component that can emit light. Due to the advantages of high energy conversion efficiency, long life and small volume, the LED lighting and display function have been widely used in our daily life. However, LED is only applied to the output, can't help feeling pity for its development. So, it's time for us to expand the use of LED.

This project studies the most fundamental phenomenon of light-emitting diode — photovoltaic effect. Touching light emitting diode before the light reflected from the finger back to it. Using the reverse-saturation current from the reverse biasing of a light-emitting diode, combined with an amplifying circuit, the whole system, controlled with our software, can monitor the change in current which ultimately allows us to touch and control through light. Also, we combine RGB LED and clock to enhance the practicality of the study. The optical touch control clock will have the color of the trend of the times, without losing the original simplicity of the clock.

Keywords: Arduino, Optical touch control, Seven-segment display

目錄

目錄.....	III
表目錄.....	IV
圖目錄.....	V
第 1 章 前言.....	6
1-1 專題製作背景.....	6
1-2 專題製作目的.....	6
1-3 預期成果.....	7
第 2 章 理論探討.....	8
2-1 元件介紹.....	8
2-2 電路原理與理論.....	9
2-3 開發環境介紹.....	12
第 3 章 專題設計.....	14
3-1 專題架構圖.....	14
3-2 專題流程圖.....	15
3-3 專題甘特圖.....	16
第 4 章 專題成果.....	16
4-1 實驗過程之問題與解決方法.....	17
4-2 專題成果.....	23
第 5 章 結論與建議.....	25
5-1 結論.....	25
5-2 建議.....	25
參考文獻.....	26
附錄.....	27
設備清單.....	27
材料清單.....	28
詳細電路圖.....	29
完整程式碼.....	30
成員簡歷.....	34

表目錄

表 1 電路偵測比較方式分析	21
表 2 一對多比較-觸控前後數值測量	21

圖目錄

圖 1 SMD5050 外型	8
圖 2 SMD5050 接腳圖	8
圖 3 LM324 外型	8
圖 4 LM324 接腳圖	8
圖 5 MEGA2560 實驗板	9
圖 6 太陽能板發電	10
圖 7 P-N 接面光伏開路電壓、短路電流定義	10
圖 8 太陽能電池典型 V-I 特性曲線	11
圖 9 可見光波長範圍	11
圖 10 ALTIUM DESIGNER	12
圖 11 ALTIUM DESIGNER 錯誤接線	12
圖 12 ARDUINO IDE	13
圖 13 ARDUINO IDE 開發介面介紹	13
圖 14 專題架構圖	14
圖 15 專題流程圖	15
圖 16 專題甘特圖	16
圖 17 三顆 LED 兩級反向放大電路	17
圖 18 三顆 LED 兩級反向放大接線	17
圖 19 三顆 LED 兩級反向放大觸控波形比較	18
圖 20 減法器	18
圖 21 儀表放大器	18
圖 22 電流轉電壓和減法電路	19
圖 23 節點清潔劑雜訊測試	19
圖 24 貼膠帶雜訊測試	20
圖 25 LM324 偵測電路	20
圖 26 光觸控時鐘位置示意圖	23
圖 27 設定小實驗-按下設定	24
圖 28 設定小實驗-設定數字	24
圖 29 顯示部分之計數系統	24

第1章 前言

1-1 專題製作背景

發光二極體(Light-emitting diode, 簡稱 LED), 是一種能發光的半導體電子元件, 因能量轉換效率高、壽命長、體積小等優點, LED 的照明及顯示功能已大量運用於生活中, 然而觀察現代人對產品的功能需求, LED 卻只應用於視覺的輸出, 不免為其發展飽和感到唏噓不已, 此時若能進一步的開發、拓展 LED 的使用便成為我們探討的目標。

時鐘是人類最早發明的物品之一: 古埃及人發現影子的位置和長度會隨著時間改變, 發明了日晷; 同時, 古代中國人也發現水的流動或者燃燒線香都有固定的時間長度, 而發明了銅壺滴漏以及一炷香的計時; 演進到現代的機械鐘錶藉由旋鈕轉動齒輪來更改時間、電子鐘的按鈕調整; 隨著科技進步, 更出現了手機及智慧手錶。由眾多的演變可知, 時鐘在人類世界占有很大的地位, 是個不會退流行的產品, 若將此用於專題上, 便可以大大提高專題的實用性, 同時在時鐘的歷史上留下一筆。

時鐘的創造是為了使人更珍惜寄蜉蝣於天地般短暫的生命, 然而現代人往往被動的接受手機來的時間訊息, 不懂得主動觀察掛在牆上安靜的時鐘。此時拋開手機的繁雜功能——光觸控時鐘, 將兼具時代潮流的色彩炫麗及科技化的觸控互動, 又不失時鐘最初的單純。

1-2 專題製作目的

- 一、利用「光伏效應」及運算放大電路, 實現 LED 發光及偵測光觸控功能。
- 二、當觸控時, 如何利用合適的硬體電路使訊號更穩定。
- 三、撰寫程式, 將光觸控應用於時鐘上。

1-3 預期成果

- 一、利用 RGB LED 達成觸控與顯示功能
- 二、經由差動電路放大後由 Arduino 程式進行偵測
- 三、撰寫具有數數字功能且能設定數字，並依照設定繼續數的程式
- 四、使用 Altium Designer 設計出電路圖並印刷出時鐘電路板

第2章 理論探討

2-1 元件介紹

一、SMD5050

SMD5050 外型採用表面黏著技術(SMT)如圖 1 所示，是一個內部含有紅色、綠色、藍色三顆發光二極體的元件，接腳如圖 2 所示。



圖 1 SMD5050 外型

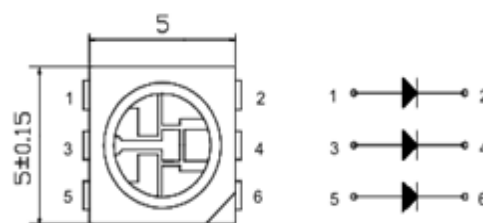


圖 2 SMD5050 接腳圖

(取自：<http://www.snowdragonledhk.com>)

(取自：<http://led-obzor.com>)

因 SMD5050 體積小且封裝內含 RGB 三色的 LED，所以成為我們專題使用的主角—觸控元件。

二、LM324(SMD)

LM324(SMD)是內含四顆運算放大器的積體電路，採用 14 支雙列表面黏著接腳塑膠封裝，其外形如圖 3 所示；它的內部包含四組形式完全相同的運算放大器，除電源共用外，四組運算放大器相互獨立，接腳圖如圖 4 所示。



圖 3 LM324 外型

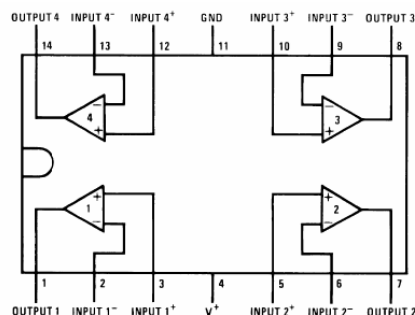


圖 4 LM324 接腳圖

(取自：<https://www.indiamart.com>)

(取自：<http://jugandi.com>)

具有內部頻率補償、直流電壓增益高、低輸入偏壓電流、低功耗電流等特點；應用領域包括正、反相放大器、有源帶通濾波器、比較器等電路。

LM324 電源電壓範圍寬、可單電源使用、價格低廉，其中可單電源供電的特性可使專題實驗時，不用隨身攜帶電源供應器，方便行動，因此我們選用其當我們的訊號放大元件。

三、Mega2560 實驗板

Mega 2560 是基於 ATmega2560 的微控制板，具有 54 個數位 I/O 接腳、16 個模擬輸入、4 個 UART、16 MHz 晶體振盪器、USB 跟電源插孔、ICSP 接頭和重置按鈕。只需用 USB 連接或用 AC 電源就能使用，外觀如圖 5 所示。

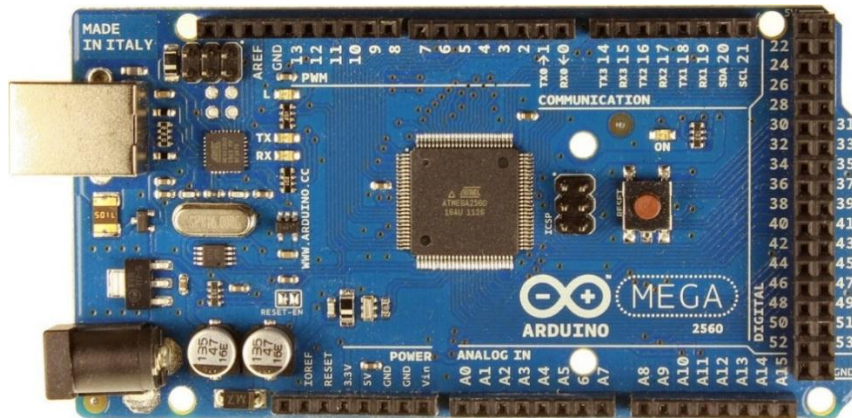


圖 5 Mega2560 實驗板

(取自：<http://taiwanarduino.blogspot.tw/2014/08/arduino-mega-2560.html>)

由於本專題使用三顆 RGB LED 為一段組成七段顯示器，再由七段顯示器發展應用，因電路龐大下，若用 UNO 板將遭遇接腳不足的問題，所以我們使用 Mega2560 當作專題的微控器。

2-2 電路原理與理論

光觸控技術之原理是運用手指遮光，反射光線至發光二極體，產生逆向電壓電流，再經電路轉換穩定後，由微控器讀取，並用程式判斷是否產生觸控，而其中光觸控的基本物理現象可為「光生伏特效應」所解釋。

光生伏特效應 (photovoltaic effect) 是指光照使「不均勻半導體」或「半導

體與金屬結合」的部位之間產生電位差的現象。光伏效應具有廣泛的應用，例如：太陽能板發電，其 P-N 接面示意如圖 6。

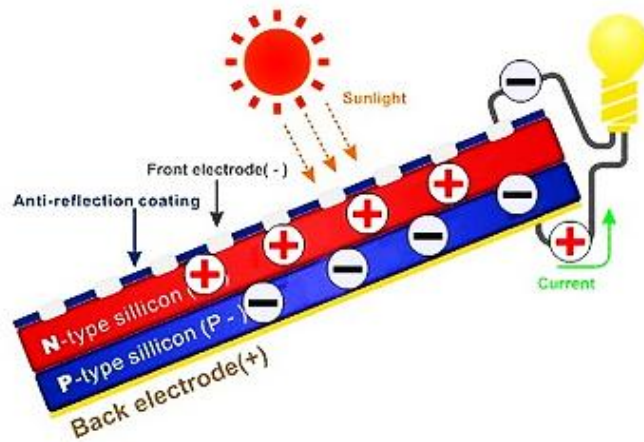


圖 6 太陽能板發電

(取自：<http://solarenergyfactsblog.com/photovoltaic-effect/>)

在初步了解專題之後，我們從 LED 逆向電壓電流測量起，過程中探討了三個部分：

- 一、光觸控的「觸控」到底是要用電流驅動，還是用電壓比較以便微控器的程式讀取？
- 二、當 RGB 三色 LED 偵測時，只會讀取一腳的逆向電壓或電流的變化，該如何選擇偵測腳才能最符合我們需求？
- 三、量測出的電壓電流都非常微小，要使用哪種元件放大訊號？

討論第一部分前，我們必須先知道光伏效應在 P-N 接面之電壓電流定義，如圖 7 所示：短路時的電流 I_{sc} (short-circuit current)稱為光電流 I_{ph} (photocurrent)；開路時的電壓 V_{oc} (open-circuit voltage)稱為光電壓 V_{ph} (photovoltage)。



圖 7 P-N 接面光伏開路電壓、短路電流定義

(取自：<https://electronics.stackexchange.com>)

知道光伏效應產生的電壓電流位置及正負方向定義後，接著看圖 8，我們能看出在不同光線強度下，電流的變化量明顯大過於電壓的變化量，因此本專題選用 LED 的逆向電流進行放大，至於電流不便由微控器直接讀取的部分，就交由電流轉電壓的硬體電路負責。

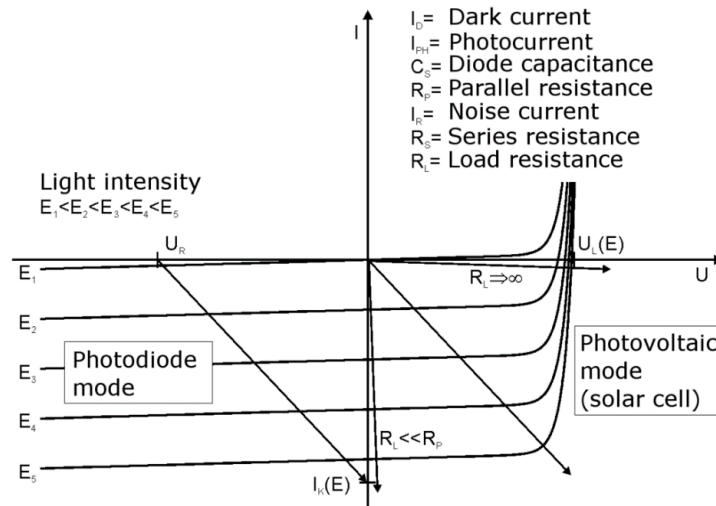


圖 8 太陽能電池典型 V-I 特性曲線

(取自：<https://photo.stackexchange.com>)

第二部分的討論，須提及 RGB 三色之波長範圍，如圖 9 所示，依據高中物理量子力學中的普朗克公式 $E=h\nu$ (註： E 為光子能量， h 為普朗克常數， ν 為光波頻率)，紅光的波長最長，頻率最低，紅光擁有之能量也就最小，當接收能量較強的藍光或綠光時，更容易產生光伏效應之電流，因此我們決定使用紅色 LED 當作偵測端判斷觸控。

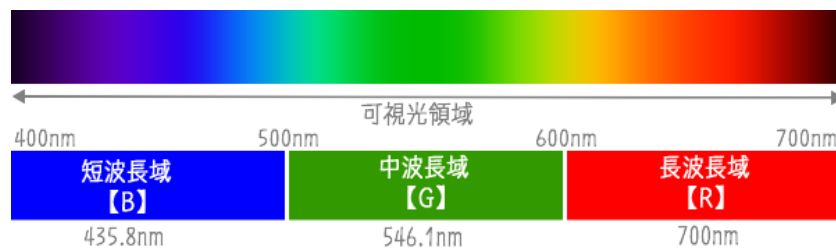


圖 9 可見光波長範圍

(取自：<http://okchallenger.seesaa.net/article/127127592.html>)

第三部分，我們回想高二電子學所學過的放大元件，不外乎就是 BJT、FET、OPA 三種元件，我們朝電路的輸出入阻抗的負載效應、熱穩定度、雜訊干擾、

頻寬大小、增益是否精確等因素進行比較，而仔細考量後發現：BJT 及 FET 在各種組態下不能夠全面性對應到以上提及的五大因素，因此 OPA 的放大電路略勝一籌，為本專題所採用。

2-3 開發環境介紹

一、Altium Designer

Altium Designer 是一個 PCB 設計軟體，用於印刷電路板的電子設計自動化套裝軟體，製作流程為電路繪製、板型設計、電路佈線、電路板製作。



圖 10 Altium Designer

(取自：<https://www.r2download.com/altium-designer-17-1-6-free-download/>)

本專題在實驗過程中，經歷三次佈線：

第一次是線路佈完後，發現佈線規則之線寬若依照預設的 10mil，容易在洗板子時洗到斷路，上網查了如何一次修改線寬的方法，找到 PCB filter 選擇 IsWire，將線全部加粗，然而很多線重疊、變成警示的綠色線，於是只好全部重新佈線。

第二次佈線時，考慮到面積有限，將全部的貼面 IC 放置背面，並讓原本要接到 Mega2560 實驗板上的線接到排針上，避免線路穿越實驗板的接腳與接腳時短路，然而正式洗出來後，卻發現有線沒有接上，如圖 11 所示。

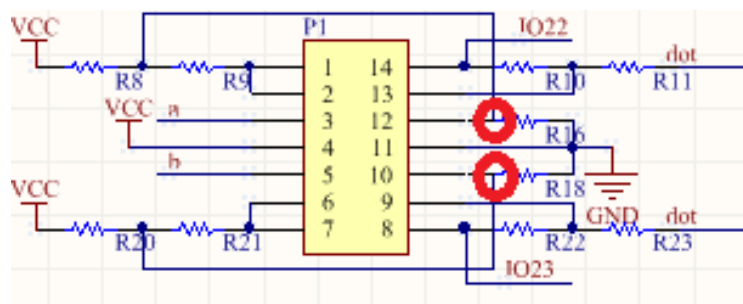


圖 11 Altium Designer 錯誤接線

第三次佈線因為想避免錯誤，線路全部手拉，並把第二次佈線遺漏的內容一併解決。

二、Arduino IDE

Arduino Integrated Development Environment(簡稱 Arduino IDE)，是一個開放原始碼的整合開發環境，可以通過向板子上的微控制器(Atmel AVR 單晶片)發送一組指令來告訴板子該做什麼，其學習門檻較為簡單，不需要電子相關科系的背景，也可以很容易學會 Arduino 相關互動裝置的開發，其圖示如圖 12 所示、開發介面如圖 13 所示。



圖 12 Arduino IDE

(取自：<https://avatars0.githubusercontent.com>)

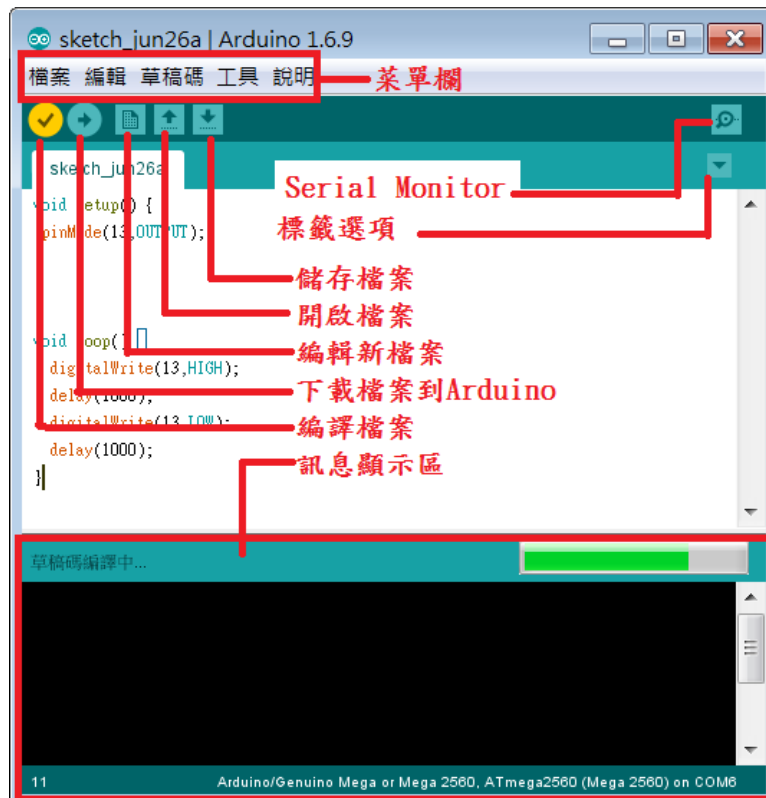


圖 13 Arduino IDE 開發介面介紹

第3章 專題設計

3-1 專題架構圖

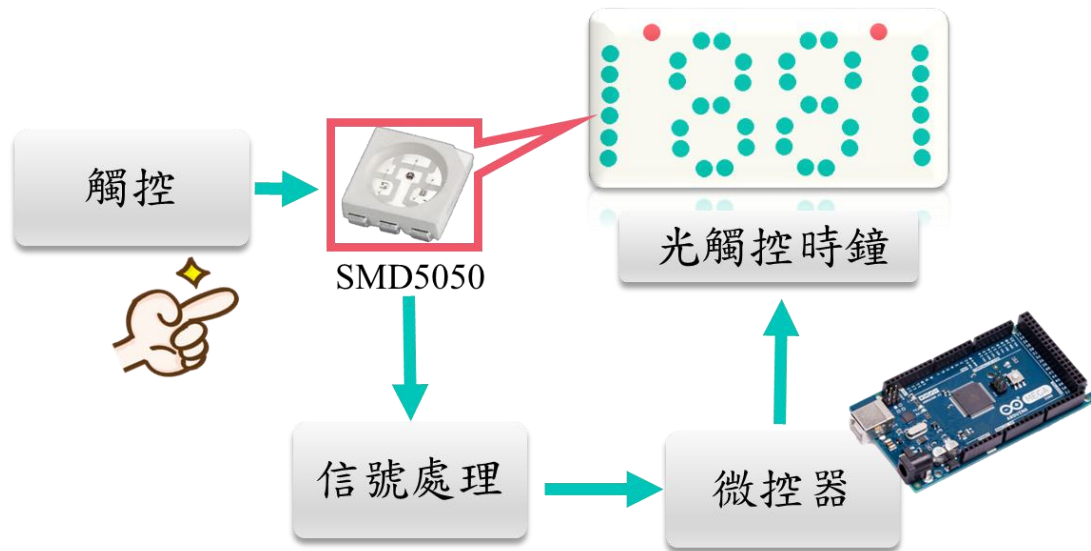


圖 14 專題架構圖

說明：

手指觸控後，反射光線至 SMD5050 的紅色 LED 燈，產生逆向電流，經放大電路等處理，將信號轉成電壓，由微控器 Mega2560 讀取，判斷觸控點，並顯示至光觸控時鐘上。

3-2 專題流程圖

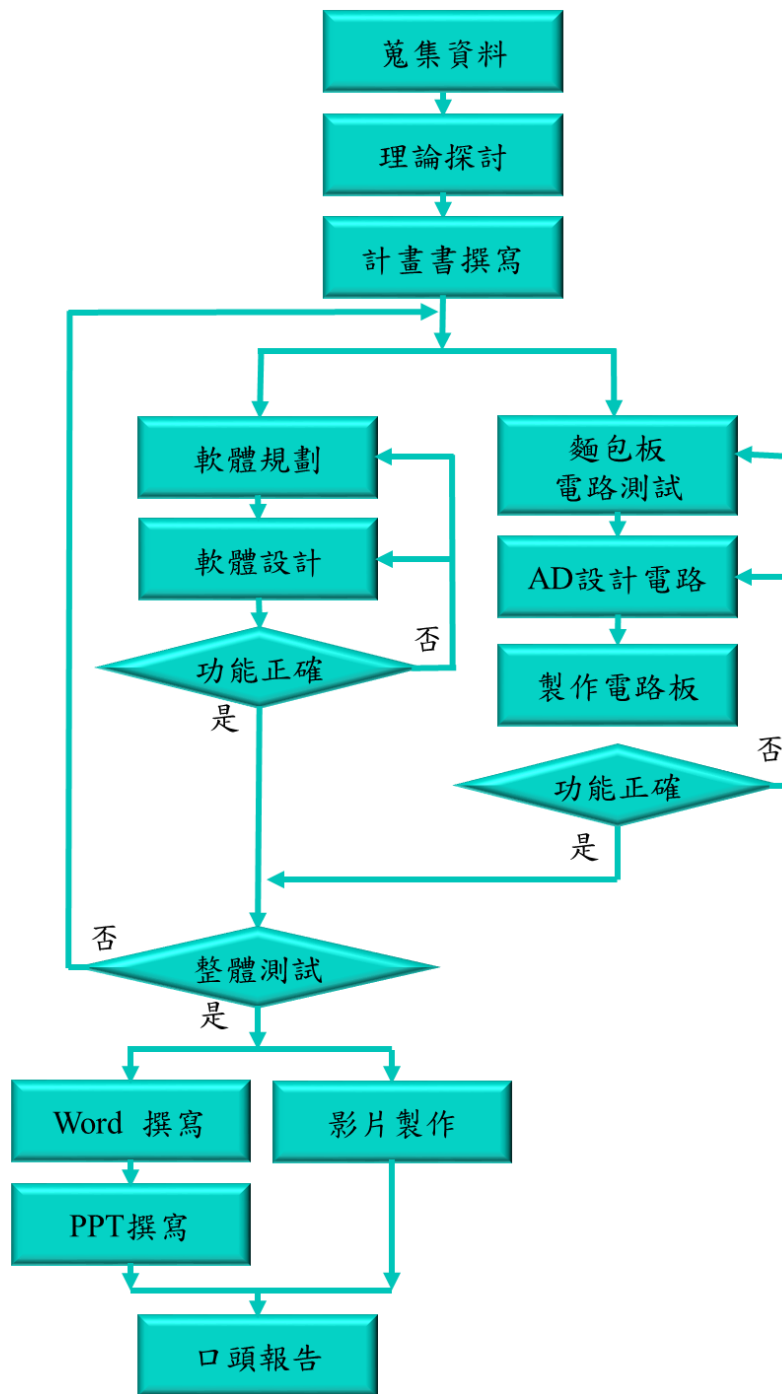


圖 15 專題流程圖

3-3 專題甘特圖

工作項目	週次 (日期)																		負責成員 (座號)註*
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
資料蒐集	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	全體
理論探討	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	全體
計畫書撰寫		■	■																5
麵包板電路測試					■	■						■	■	■	■				4,24,29
Altium Designer 設計電路					■	■	■	■				■	■	■	■				全體
製作印刷電路板												■	■	■	■				4,24
軟體規劃與設計					■	■	■	■	■	■	■	■	■	■	■				5,29
軟體測試					■	■	■	■	■	■	■	■	■	■	■				5,29
整體測試												■	■	■	■				全體
報告撰寫			■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	全體
影片製作														■	■				5
口頭報告							■		■		■		■			■	■	■	全體
預定進度	2	5	9	12	19	26	34	39	46	49	55	62	74	84	96	97	99	100	累積 百分比%

圖 16 專題甘特圖

註*：4 號郭祐寧、5 號黃鈴漪、24 號胡辰瀚、29 號許哲綸

第4章 專題成果

4-1 實驗過程之問題與解決方法

實驗問題一：觸控訊號經兩級反向放大後，雜訊跟著放大，無法辨識是否觸控。

實驗解決一：電路改用儀表放大器。

實驗過程：

最初實驗時，我們單純使用兩級的反向放大電路偵測觸控，如圖 17、圖 18 所示，當綠燈輸入 5v 方波，多顆 LED 一起觸控量測出來的方波雜訊太大，無法辨識是否有觸控如圖 19 所示。

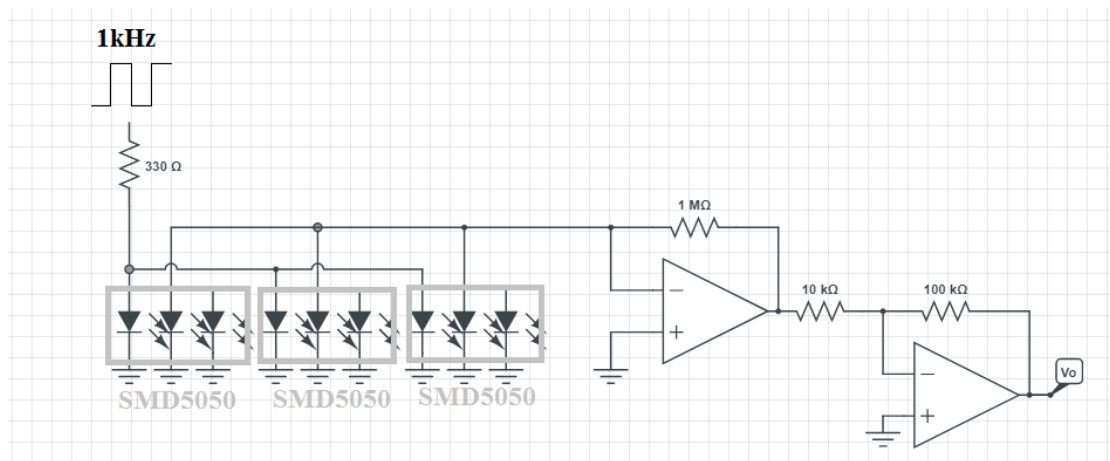


圖 17 三顆 LED 兩級反向放大電路

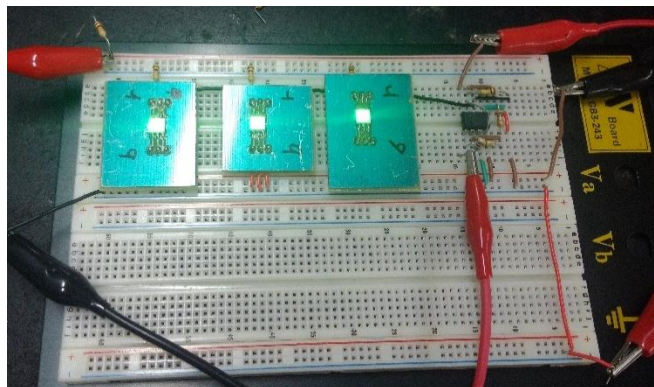


圖 18 三顆 LED 兩級反向放大接線

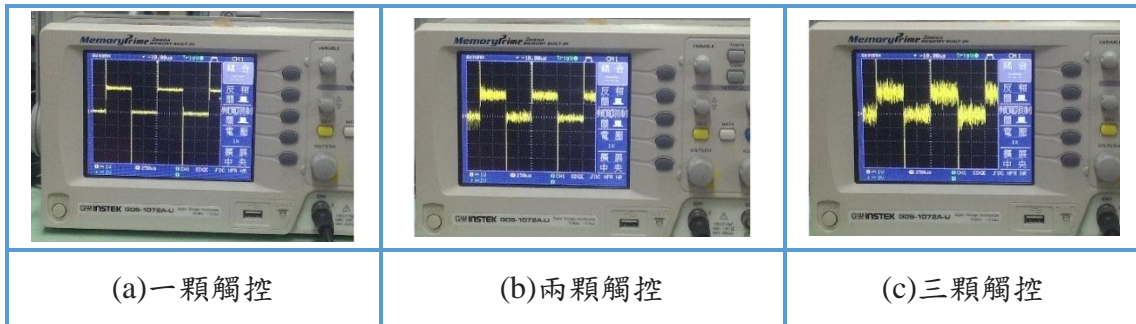


圖 19 三顆 LED 兩級反向放大觸控波形比較

一開始為了解決雜訊，我們使用的是減法器，如圖 20 所示，但因 OPA 為非理想，輸入阻抗不為無限大，導致負載效應，所以我們加上儀表放大器，如圖 21 所示，進一步解決負載效應的問題。

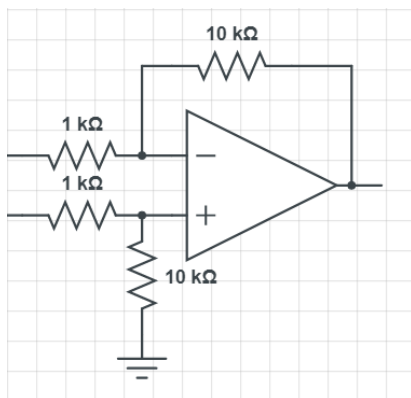


圖 20 減法器

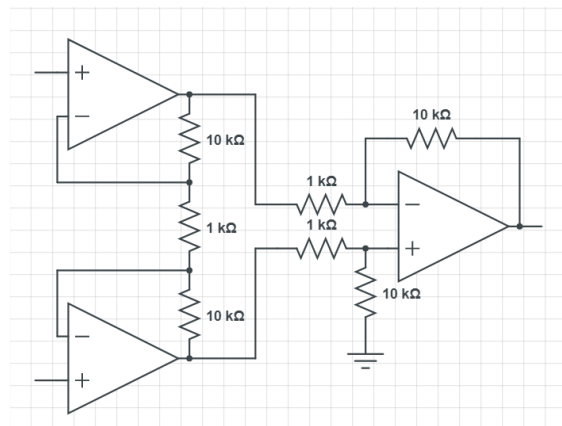


圖 21 儀表放大器

實驗問題二：儀表放大器之放大倍率實際值與計算值不符。

實驗解決二：電路改用电流轉電壓電路再加上減法器。

實驗過程：

電路換成儀表放大器後，卻在量測時，多次發現實際值與計算值不符，於是我們改用电流轉電壓電路再加上減法器，如圖 22 所示，以達成第一級輸入阻抗為 ∞ 、輸出阻抗為 0 的條件。

(輸入阻抗為 ∞ ：因輸入為電流，所以輸入阻抗= ∞ ，才不會負載效應的問題)

(輸出阻抗為 0：因輸出為電壓，所以輸出阻抗=0，才不會有負載效應的問題)

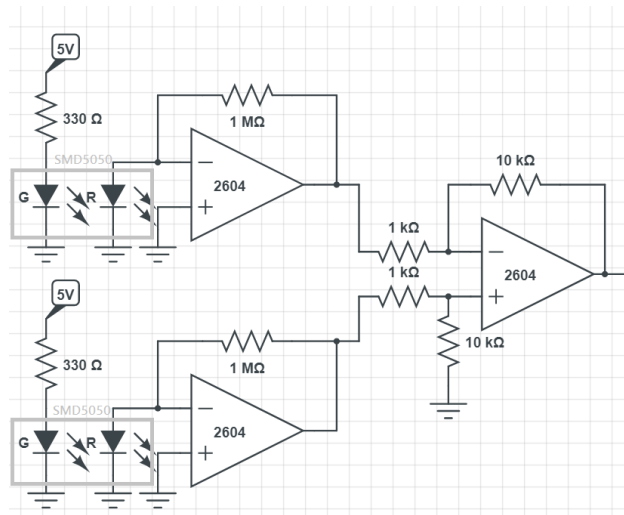


圖 22 電流轉電壓和減法電路

實驗問題三：手指帶有的雜訊強，輸出波形穩定度差。

實驗解決三：在 SMD5050 表面貼上膠帶，進行隔離。

實驗過程：

我們觀察輸出波形的同時，都是在短暫時間內，截取穩定的波形圖，原先打算使用絕緣漆，解決觸控時的手指雜訊，但因經費問題，我們使用實習工廠的節點清潔劑來代替，然而如圖 23 所示，節點清潔劑噴上去隔了一天也沒乾，量測出來問題也沒改善，結論是將其代替絕緣漆，並無法達成抗雜訊功效。

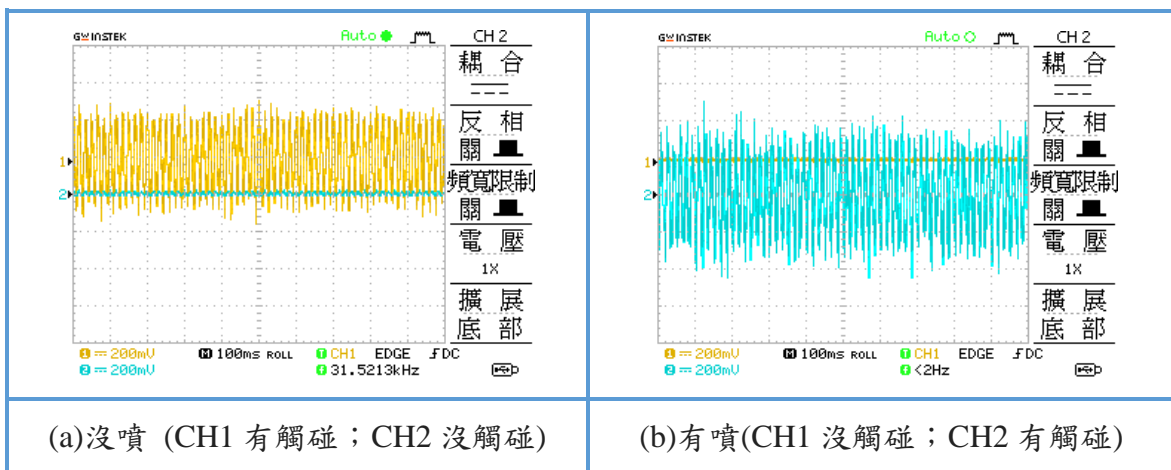


圖 23 節點清潔劑雜訊測試

接著我們嘗試使用膠帶隔絕雜訊，如圖 24 所示，結果比原先穩定許多，故後面的 SMD5050 都被我們在觸控部分貼了膠帶。

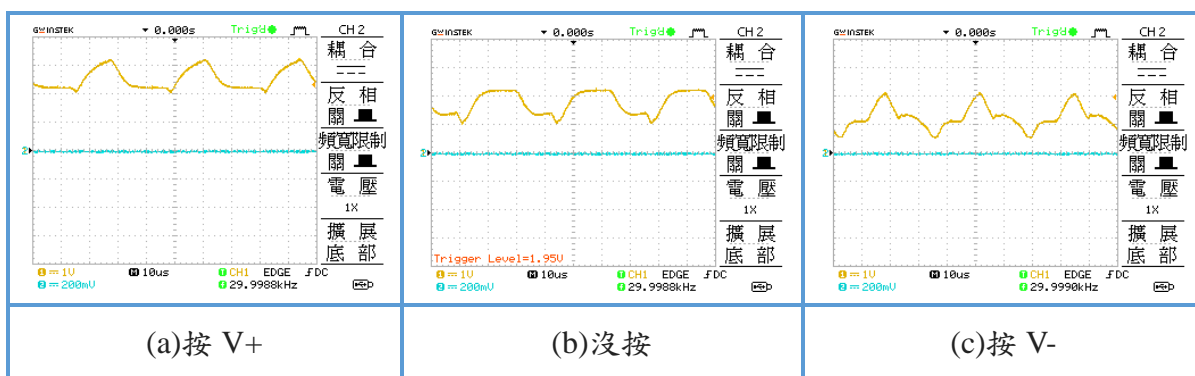


圖 24 貼膠帶雜訊測試

實驗問題四：OPA2604 電源供應正負電壓不方便。

實驗解決四：OPA 改用 LM324，並加上分壓電阻提升電壓直流準位。

實驗過程：

我們考慮到專題的實用性，若使用 OPA2604 將會用到雙電源，且其內部只封裝兩顆運算放大器、單價又高，所以我們替換其為 LM324，而改成單電源的 LM324 的同時，因為觸控輸入端的 V-時，數值會跑到零以下，導致失真，也就是偵測範圍之外，只能使用 V+端，所以在電路上，我們加了一組分壓電阻，如圖 25 所示，於 V+，提升直流準位，讓 V-端能正常使用。

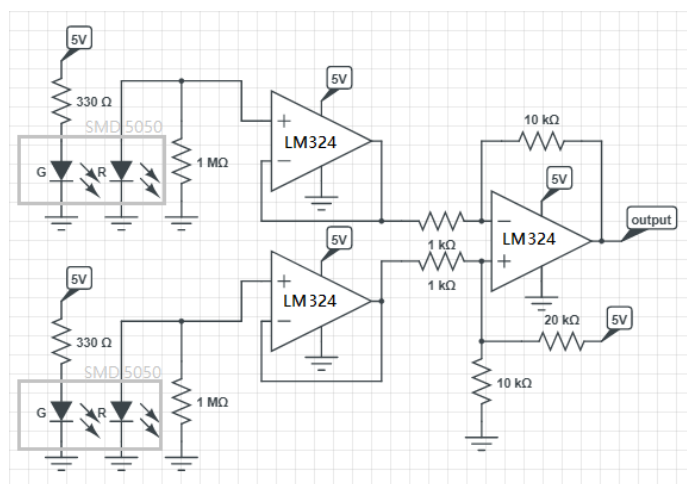


圖 25 LM324 偵測電路

實驗問題五：兩顆燈的比較實驗，如何擴大成我們所需要的完整電路。





實驗解決五：使用一對多的比較方式接線擴大。

實驗過程：

在偵測電路架構完成後，接著是討論偵測點的比較方式：一對多還是兩兩比

較，如表 1 所示，這部分我們思考了一陣子，因為兩種比較方式的優缺點是對半分，詳細一點能考慮到一對多的「一」，可能會無法跟每個點在同一個環境光源進行比較，也就是觸控時，手臂的遮光導致誤判為有觸控，將會非常麻煩；而兩兩比較的部分程式寫起來很麻煩，站在使用者的角度，則是不能同時按兩個相互比較的點，要記住內部線路會變得不易使用… …。最終，我們決定一對多進行比較，並將比較點放 V-，a~g 等顯示點放 V+。

表 1 電路偵測比較方式分析

兩兩比較	 線路簡單
	 比較的兩個不能同時按
比較點	 有複合鍵>>按鈕功能
	 線路複雜

決定了比較方式，我們實際接了代表性的七顆觸控點，用以觀察觸控前後的數值是否可為微控器所判斷並讀取，而實際的數值測量如表 2 所示，數值上能看得出觸控前後的變化，但放大後的幅度不夠統一，在軟體上須取觸控變化最小的為參考值。

表 2 一對多比較-觸控前後數值測量

	a	b	c	d	e	f	g
觸控前	1.3v	2.2v	2.4v	1.5v	0.7v	1.4v	1.4v
觸控後	1.8v	2.9v	3v	2v	1.8v	1.8v	2v

實驗問題六：線路的龐大電路，Altium Designer 佈線困難。

實驗解決六：使用排針隔離要接上 Mega2560 實驗板的線路。

實驗過程：

雖然不能真實變成一個產品的樣子呈現，但也多虧了排針與杜邦線，讓單純的專題實驗很好進行變化，例如：我們買到的 SMD5050 有些藍色、綠色接腳剛好相反，此時接線移動就十分輕鬆。

實驗問題七：洗出來的電路板壽命短，觸控的紅色偵測點實驗幾次就壞掉。

實驗解決七：更換綠色、藍色的限流電阻。

實驗過程：

因為麵包板的偵測一直都很穩定，且真正洗出來的電路與麵包板硬體實驗，並沒有更動的內容，所以依據理論我們沒有去考慮硬體的設計有問題，反而覺得是因為各顆特性不同，所以當紅色不能顯示時，就直接換一顆。

直到有一次實驗時，整塊板子紅色偵測燈燒壞、非觸控的燈紅色仍能顯示，我們才驚覺不對勁，討論許久後，發現在麵包板上的觸控只有反射綠光，而洗出來的板子卻加上了藍光，能量約為兩倍，偵測時紅色 LED 逆向衝擊變大，也就更容易燒壞。

於是我們將電阻從 330 歐姆換成 1k 歐姆，結果紅燈不僅沒再壞掉，也剛好解決面板太亮的問題，可說是意外的小收穫。

4-2 專題成果

光觸控時鐘使用說明書：

如圖 26 所示，時鐘開機後，從 00 時 00 分開始計時，同正常時鐘運作，當按下設定點，切換狀態為設定模式，分鐘面板顏色全為綠色，此時可設定上下午顯示點(預設為上午亮黃色，按下後變為下午藍色)，一秒後視為上下午設定完成，此時可設定時間(觸控點如圖 26 的藍圓橘框圓圈)，按下的小時燈亮(其餘小時燈滅)，按下的分鐘切換為藍綠色，十五秒後視為設定完成，若設定有錯誤，回到設定狀態重新設定，若設定正確，由設定的時間繼續計數。

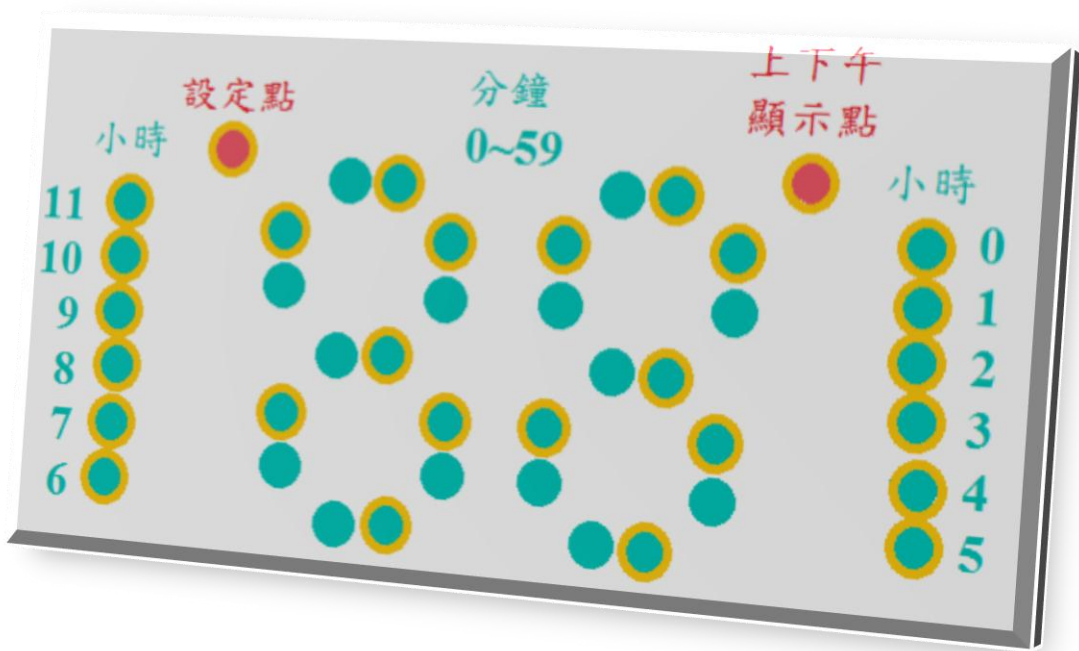


圖 26 光觸控時鐘位置示意圖

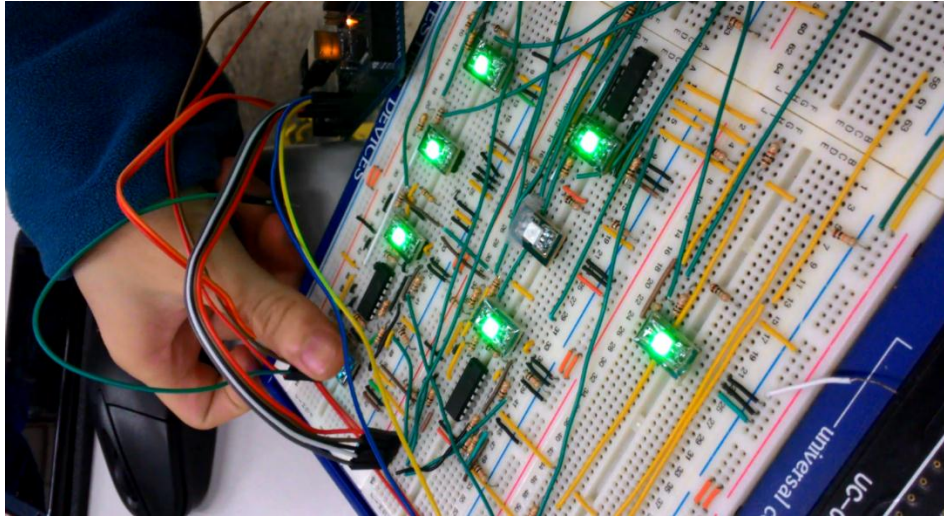


圖 27 設定小實驗-按下設定

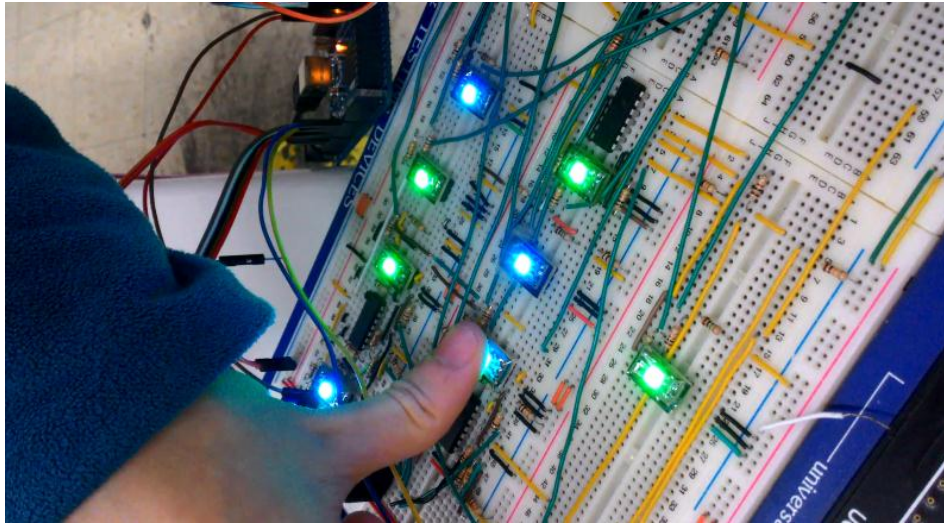


圖 28 設定小實驗-設定數字

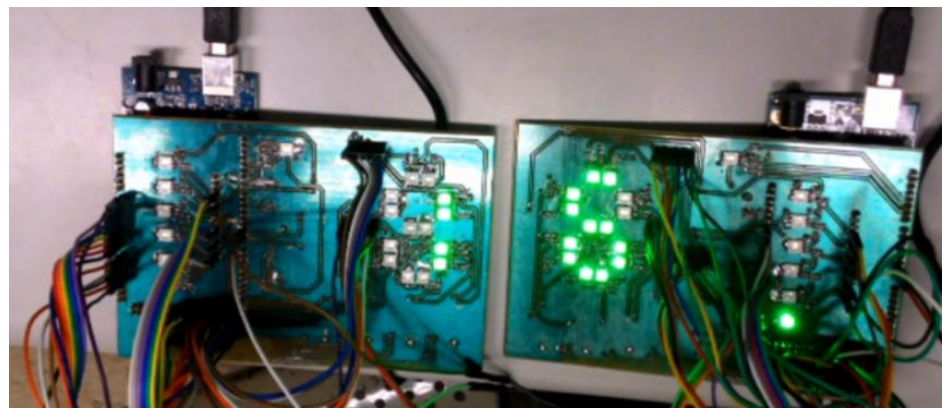


圖 29 顯示部分之計數系統

第5章 結論與建議

5-1 結論

本次因為是使用新的技術嘗試應用於生活中，整個硬體及軟體架構，僅有少少的參考資料，多數要自己重頭實驗，而每次的實際操作對本專題來說都十分重要，故在文中詳細的提及其過程，讓未來的人將此視為借鏡，能夠又更好更快的解決我們所遇到的問題。

高中生涯最艱鉅的一項挑戰——專題，讓我們真正從書本中走出來，學習如何自己設計電路、進行實驗、規畫機構、建構程式，相信能力有在實作中漸漸培養，過程中有些遺憾的是，雖然我們從暑假開始就和指導老師做了多次討論，但往往行動力落後，沒能把握時間，硬體部分出來的稍慢，讓軟體沒有足夠的時間進行測試，不過這些過程使我們萃取經驗，同時紮實地豐富了我們高三生活。

5-2 建議

硬體部分，因著我們對 LED 的逆向操作，並非在出廠廠商的品質管制考量範圍內，每顆 LED 的逆向特性不同，得經過實際測量後，依照各顆特性進行調整，不同直流準位才能達到一致性，而硬體電路也會因此不能適用特性差異太大的 LED，這讓我們十分困擾，希望在未來若光觸控技術有所發展，廠商能更注意逆向操作的使用特性，抑或軟體控制能找到解決對策。

機構部分，控制兩塊板子為了增加控制靈活性，電路與 Mega2560 板子使用杜邦排線連接有些零亂，組合面板也略顯陽春，未來可再改變零件擺放位置，或將元件模組化疊成雙層雙面，解決單層雙面面積大的問題。

軟體部分，同機構部分的問題，用了兩塊板子且分開來寫程式，未來可以試著用一支程式控制兩塊板子，甚至只用一塊板子一支程式就完成。另外，在偵測

方面，因 Timer 使用方便，現為我們所使用，之後可考慮用中斷的方式取代 Timer，有觸控時才進行設定部分的觸發，讓程式更有效率、顯示更順暢。

參考文獻

- [1]楊仁元、張顯盛、林家德(2014)：專題製作理論與呈現技巧。新北市：臺科大圖書。
- [2] 梅克²工作室(2014)：Arduino 微電腦控制實習。新北市：臺科大圖書。
- [3] S.O. Kasap(2010)：光電半導體元件。黃俊達等人譯。新北市：全威圖書。
- [4] Cooper Maa(2011)。TimedAction 函式庫簡介。2017 年 11 月 23 日，取自：
<http://coopermaa2nd.blogspot.tw/2011/04/timedaction.html>
- [5] Alexander Brevig(2015).TimedAction Library for Arduino. Retrieved November 23, 2017, from <https://playground.arduino.cc/Code/TimedAction>
- [6]Arduino team(2017). ARDUINO MEGA 2560 REV3. Retrieved December 12, 2017, from <https://store.arduino.cc/usa/arduino-mega-2560-rev3>
- [7]☆故事☆(2011)。Altium Designer～一套電路設計軟體安裝及介紹。2017 年 12 月 12 日，取自：<http://roilyoko.pixnet.net/blog/post/29048596-altium-designer>
- [8] 喬治查爾斯電子電路網(2011)。LM324 四組運算放大器的應用。2017 年 12 月 13 日，取自：<http://gc.digitw.com/Circuit/LM324-Applys.pdf>
- [9] 陸向陽(2015)。【名詞解釋】認識 IDE 整合開發環境。2017 年 12 月 13 日，取自：<https://makerpro.cc/2015/08/what-is-ide/>
- [10] SnapEDA(2017). LM324DG4. Retrieved December 17, 2017, from <https://www.snapeda.com/parts/LM324DG4/Texas%20Instruments/view-part/>

附錄

附錄 1 設備清單

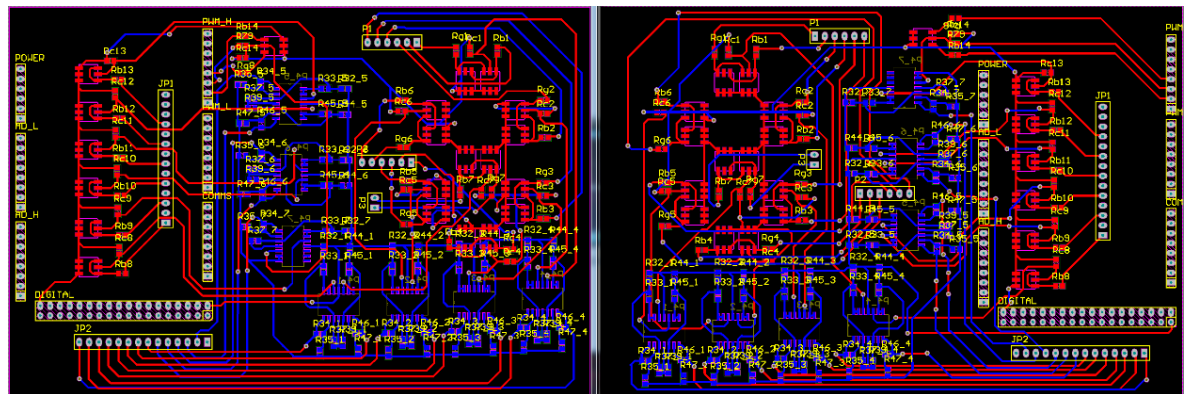
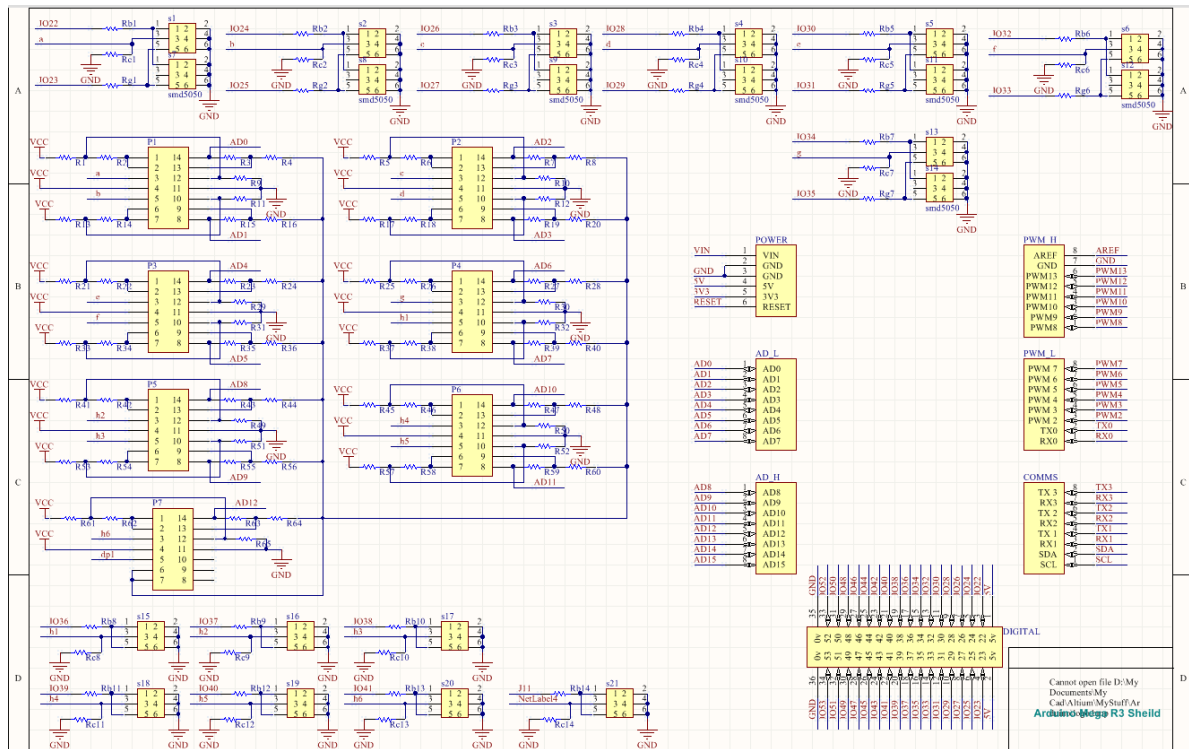
類 別	設備、軟體名稱	應 用 說 明
硬 體	桌上型電腦	撰寫程式語言、製作企畫書、製作簡報、查詢相關知識
	Arduino Mega2560	主控元件之微處理機平台
	黑白雷射印表機	列印實驗電路
	曝光機	曝光電路板
	電路蝕刻機	蝕刻電路板
	鑽孔機	電路鑽孔
	切割機	切割電路板
	烙鐵	焊接零件
	電源供應器	供應測試用電路電源
	示波器	觀察輸出訊號變化
	訊號產生器	產生訊號之儀器
	數位電表	檢測電路
	麵包板	製作實驗電路
軟 體	Arduino IDE	主控元件之微處理機平台
	Altium Designer	規劃電路、設計電路
	Microsoft Word 2013	文書處理

軟體	Microsoft PowerPoint 2013	簡報製作
	Power Director 15	專題影片製作
	Google 雲端硬碟	便於資料存取及彙整

附錄 2 材料清單

類別名稱	材 料 名 稱	單 位	數 量	應 用 說 明	備 註
LED	SMD5050	顆	42	感測觸控反射光源、顯示	
IC	LM324	個	14	電流轉電壓電路、放大電路	
電阻器	1K Ω	顆	98	限流電阻(46 顆)、放大電路(52 顆)	
電阻器	10K Ω	顆	52	放大電路、提升直流準位分壓電阻	
電阻器	20K Ω	顆	26	提升直流準位分壓電阻	
電阻器	1M Ω	顆	28	電流轉電壓電路	

附錄 3 詳細電路圖



附錄 4 完整程式碼

```
#define F_CPU 16000000UL //石英晶體震盪器頻率
#include <avr/io.h>      //AVR 設備通用的 IO 定義
#include <TimedAction.h> //引入隱藏 millis()的狀態下達成每 X 毫秒執行某個
Action 的功能

int SensingSource[7] = {A0, A1, A2, A3, A4, A5, A6}; //定義輸入接腳
int InitialSenseValue[7] = {}; //定義初始信號輸入訊號陣列
int SignalSenseValue[7] = {}; //定義觸控信號輸入訊號陣列
const int Delta[7] = {180, 200, 200, 170, 180, 200, 160};
const int GreenPin[7] = {22, 24, 26, 28, 30, 32, 34};
const int BluePin[7] = {23, 25, 27, 29, 31, 33, 35};
const int DpPin[3] = {9, 11};
const char SEG7[10] = {0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x07, 0x7f, 0x6f};
const char SEG[7] = {0x01, 0x02, 0x04, 0x08, 0x10, 0x20, 0x40};

byte Numsignal = 0x00; //定義數字訊號
int Dpsignal = 0;
int Tmin; //定義分鐘
boolean Status = false; //預設顯示

void SetInitialSenseValue(); //宣告副程式偵測初值
void SetSignalSenseValue(); //宣告副程式偵測信號
void Counter(); //宣告副程式計數器
void Reciprocal(); //宣告副程式判斷

TimedAction SetInitialSenseValueAction = TimedAction(1000000,
SetInitialSenseValue);
//建立一個 Action 每 1 秒執行一次 Detect()
TimedAction SetSignalSenseValueAction = TimedAction(20, SetSignalSenseValue);
//建立一個 Action 每 0.002 秒執行一次 Detect()
TimedAction CounterAction = TimedAction(500, Counter);
//建立一個 Action 每 60 秒執行一次 Counter()
TimedAction ReciprocalAction = TimedAction(6000, Reciprocal);
//建立一個 Action 每 60 秒執行一次 Counter()
```

```

void setup() {
  Serial.begin(9600);
  for (int i = 0; i < 7; i++) {
    pinMode(GreenPin[i], OUTPUT);
    digitalWrite(GreenPin[i], HIGH);
    pinMode(BluePin[i], OUTPUT);
    digitalWrite(BluePin[i], LOW);
  }
  for(int i=0; i<3 ;i++){
    pinMode(DpPin[i], OUTPUT);
    digitalWrite(DpPin[i], HIGH);
  }
  delay(50);
  SetInitialSenseValue();
  for (int i = 0; i < 7; i++) {
    digitalWrite(GreenPin[i], LOW);
  }
}

void loop() {
  SetInitialSenseValueAction.check();//讀取偵測值
  SetSignalSenseValueAction.check();//讀取偵測值
  Detect();//計算變動量
  if(Status){
    ShowminB(Numsignal);
    ShowminG(0x7f);
    ReciprocalAction.check();//讀取偵測值
  }
  else{
    if(Dpsignal>6){
      ShowminG(0x7f);
      Status = true;
      Numsignal = 0;
      ReciprocalAction.reset();
    }
    else{
      ShowminB(0x00);
      ShowminG(SEG7[Tmin]);
    }
  }
}

```



```

        CounterAction.check(); //計數
    }
}
}

void Reciprocal() { //確認
    int counter = 0;
    for(int i=0;i<10;i++){
        int seg7 = SEG7[i];
        if(seg7 == Numsignal){
            Tmin = i;
            CounterAction.reset();
            Numsignal = 0;
            Status = false;
        }
        else counter++;
    }
    if(counter = 10){
        ShowminG(0x00);
        Numsignal = 0;
        ReciprocalAction.reset();
        delay(20);
    }
}

void SetInitialSenseValue() { //初始數值
    for (int i = 0; i < 7; i++) { //讀取標準值並存入
        InitialSenseValue[i] = analogRead(SensingSource[i]);
    }
}

void SetSignalSenseValue() { //訊號
    for (int i = 0; i < 7; i++) { //讀取訊號值並存入
        SignalSenseValue[i] = analogRead(SensingSource[i]);
    }
}

void Detect() { // 0~6 abcdefg ; 7~12 = 0~5hr

```

```

Dpsignal = 0;
for (int i = 0; i < 7; i++) {
    if (SignalSenseValue[i] - InitialSenseValue[i] > Delta[i]) {
        Numsignal = Numsignal|SEG[i];
    }
    else if ( InitialSenseValue[i] - SignalSenseValue[i] > 75) {
        Dpsignal = Dpsignal+1;
    }
}
}
}

```

```

void ShowminG(byte dat) { //dat 為 14 位元二進制
    for (int j = 0; j < 7; j++) {
        if (dat % 2 == 1)
            digitalWrite(GreenPin[j], HIGH);
        else
            digitalWrite(GreenPin[j], LOW);
        dat = dat / 2;
    }
}
}

```

```

void ShowminB(byte dat) { //dat 為 14 位元二進制
    for (int j = 0; j < 13; j++) {
        if (dat % 2 == 1)
            digitalWrite(BluePin[j], HIGH);
        else
            digitalWrite(BluePin[j], LOW);
        dat = dat / 2;
    }
}
}

```

```

void Counter() {
    if (Tmin < 9)Tmin ++;
    else {
        Tmin = 0;
    }
}
}

```

成員簡歷

姓 名	黃鈴漪	班 級	電子三乙	
曾 修 習 專 業 科 目	<ol style="list-style-type: none"> 1.基本電學與實習 2.電子學與實習 3.數位邏輯與實習 4.電子電路與實習 5.微處理機與實習 			
參 與 專 題 工 作 項 目	<ol style="list-style-type: none"> 1.資料蒐集 2.理論探討 3.計畫書撰寫 4.Altium Designer 設計電路 5.軟體規劃與設計 6.軟體測試 7.整體測試 8.報告撰寫 9.影片製作 10.口頭報告 			
經 歷 簡 介	<p>工業電子丙級技術士</p> <p>104 學年第 1 學期擔任事務幹事</p> <p>104 學年第 2 學期擔任環保幹事</p> <p>104 學年上下學期擔任物理小老師</p> <p>105 學年上下學期擔任電子學小老師</p> <p>105 學年第 1 學期擔任地理小老師</p> <p>105 學年第 1 學期擔任公民小老師</p> <p>106 學年第 1 學期擔任 A 組實習工場領班</p> <p>106 學年第 1 學期全校優良學生代表</p>			

姓 名	郭祐寧	班 級	電子三乙	
曾 修 習 專 業 科 目	<ol style="list-style-type: none"> 1.基本電學與實習 2.電子學與實習 3.數位邏輯與實習 4.電子電路與實習 5.微處理機與實習 			
參 與 專 題 工 作 項 目	<ol style="list-style-type: none"> 1.資料蒐集 2.理論探討 3.麵包板電路測試 4. Altium Designer 設計電路 5.製作印刷電路板 6.整體測試 7.報告撰寫 8.口頭報告 			
經 歷 簡 介	<p>工業電子丙級技術士</p> <p>104 學年第 1 學期擔任外掃股長</p> <p>106 學年第 1 學期擔任 A 組實習工場材料長</p>			

姓名	胡辰瀚	班級	電子三乙	
曾修習專業科目	<ol style="list-style-type: none"> 1.基本電學與實習 2.電子學與實習 3.數位邏輯與實習 4.電子電路與實習 5.微處理機與實習 			
參與專題工作項目	<ol style="list-style-type: none"> 1.資料蒐集 2.理論探討 3.麵包板電路測試 4. Altium Designer 設計電路 5.製作印刷電路板 6.整體測試 7.報告撰寫 8.口頭報告 			
經歷簡介	工業電子丙級技術士			

姓名	許哲綸	班級	電子三乙	
曾修習專業科目	<ol style="list-style-type: none"> 1.基本電學與實習 2.電子學與實習 3.數位邏輯與實習 4.電子電路與實習 5.微處理機與實習 			
參與專題工作項目	<ol style="list-style-type: none"> 1.資料蒐集 2.理論探討 3.麵包板電路測試 4. Altium Designer 設計電路 5.軟體規劃與設計 6.軟體測試 7.整體測試 8.報告撰寫 9.口頭報告 			
經歷簡介	<p>工業電子丙級技術士</p> <p>104 學年第 2 學期擔任圖書股長</p> <p>106 學年第 1 學期擔任 A 組實習工場工具長</p>			