



Annual Report
Institute of Space and Plasma Sciences
National Cheng Kung University

2022 年度報告
研究生: 劉致賢
指導教授: 張博宇 博士
日期 2022/1/21

摘要

在 2021 下半年，我設計了一個高壓脈衝產生器（high-voltage pulse generator），做這個的高壓脈衝產生器有兩個用處：第一點主要是配合實驗室現有的設備去產生電漿；第二點是如果配合整流器去組合使用，則可以用於製作大功率高壓直流電源供應器。在這期間中，我同時學習了 Arduino，也練習了如何焊接電路板，還有怎麼去除錯。除此之外，在這段期間也有針對軌道間隙開關中的電極進行拋光處理，也有紀錄電極安裝的詳細過程。本次報告會介紹各項的執行內容。

目錄

一:Arduino 的練習

二:高壓脈衝產生器(High-voltage pulse generator)之製作

三:軌道間隙電極開關安裝流程

四:逃生路線圖之更新

五:未來展望

六:結論

七:附錄

一:Arduino 的練習

此章節有四個小節，分別為：

- (一)Arduino 練習目的 ·
- (二)輸出目標練習 ·
- (三)Arduino 程式 ·
- (四)輸出目標結果 ·
- (五)Summary ·

(一)Arduino 練習目的

練習 Arduino 的目的是希望未來能透過 Arduino 來操控儀器的輸出。參考的書籍為“Arduino 互動設計入門 第四版”，是由杜承翰學長推薦的。書中有蠻多例子的，我主要練習的題目是利用 Arduino 來產生 10kHz 的方波訊號，並利用此訊號來控制 LED 燈。未來我們會利用這個 Arduino 的訊號控制高壓脈衝產生器中的 H 電橋(H-bridge)開關，讓直流電在負載中變成交流電的狀態。

(二)輸出目標練習

我們所要產生的訊號如圖 1-1，透過 Arduino 輸出 2 個 5V 的訊息，訊號波形每 100 us 會重複一次，即 10 kHz 的重現率。在 100 us 內中有四個階段。

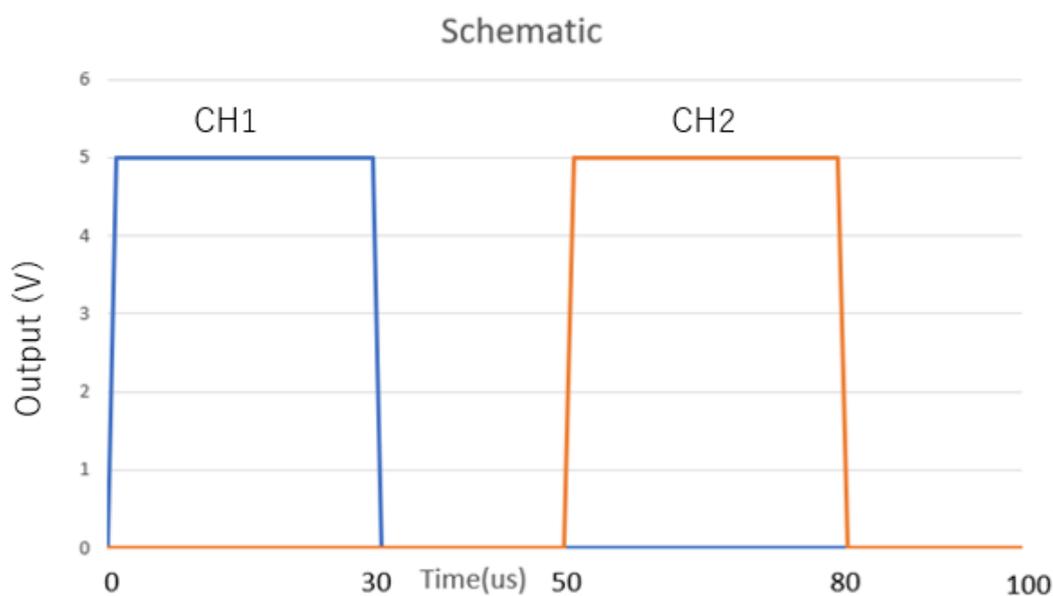


圖 1-1:預期產生之訊號

- (1): 在 0 us~30 us 的時候，訊號 1 送出 5 伏特的訊號，訊號 2 送出 0 伏特訊號。
- (2):在 31 us~50 us 的時候，訊號 1 和訊號 2 皆送出 0 伏特訊號。
- (3):在 51 us~80 us 的時候，訊號 2 送出 5 伏特的訊號，訊號 1 送出 0 伏特訊號。
- (4):在 81 us~100 us 的時候，訊號 1 和訊號 2 皆送出 0 伏特訊號。

(三)Arduino 程式

圖 1-2 是我們控制 Arduino 的程式碼。第一步在第 2 和 3 行定義第九以及第十腳位是輸出的 Pin 腳位，分別輸出訊號一和訊號二。第二步則是在執行階段(loop)訊號開啟(HIGH)和關(LOW)的時間，並利用延遲時間的方式來達成設定 30us 的延遲時間和 20us 延遲時間。延遲的時間是以 us 為單位，所以需要使用 delayMicroseconds 的指令，不能使用以 ms 為單位的 delay 指令配合含小數點的輸入數字。

```
Arduino_test $  
  
1: void setup() {  
2:   pinMode(10, OUTPUT);  
3:   pinMode(9, OUTPUT);  
4: }  
  
5: void loop() {  
6:   digitalWrite(10, HIGH); → 第一階段  
7:   digitalWrite(9, LOW);  
8:   delayMicroseconds(30);  
9:   digitalWrite(10, LOW); → 第二階段  
10:  digitalWrite(9, LOW);  
11:  delayMicroseconds(10);  
12:  digitalWrite(10, LOW); → 第三階段  
13:  digitalWrite(9, HIGH);  
14:  delayMicroseconds(30);  
15:  digitalWrite(10, LOW); → 第四階段  
16:  digitalWrite(9, LOW);  
17:  delayMicroseconds(10);  
18: }
```

圖 1-2:控制 Arduino 的程式碼。

我們發現實際輸出與設定會有差距，我們所做的調整是將階段 2 (line8~10) 和階段 4 (line14~16) 的延遲時間調整為 10 us。原因為每個 Arduino 的品質不一，在延遲時間為 us 等級的時候，Arduino 的訊號延遲時間不會是我們所預期的設定。所以我們將步驟二和步驟四的延遲時間調整為 10 us 時，我們所設定一個週期的時間為 80us，但我們實際會測到的訊號週期 105 us，換算頻率為 9.53kHz。和原本所設計的 10kHz 相差 4.7%，在我們可接受的範圍以內。

(四)輸出結果

接下來要做的是量測 Arduino 所送出訊號的數據分析，我想要量測的數據有；

- 一：Ch1 訊號的時長。
- 二：Ch2 訊號的時長。
- 三：Ch1 一個訊號週期時長。
- 四：Ch2 一個訊號週期時長。

圖 1-3 到圖 1-5 和表 1 是做的測試和數據分析。圖 1-3 為 CH1 和 CH2 的圖例。我們將分析 CH1 和 CH2 訊號時長數據。

我們分析 CH1 和 CH2 訊號時長數據，時間差的計算方式為手動計算，利用示波器得到的 Excel 數據取出在第一個脈衝中 Y 軸大於 3V 的數據點，找出這些數據點中時間最小的點的時間設為起始時間 t_1 。接著，判定 0V 的點的方法，為在第一個脈衝中 Y 軸最後一個大於 3V 的數據點且後一個時間點為小於 3V 的數據點設為截止時間 t_2 ，則訊號週期時長為起始時間-截止時間 t_2-t_1 (us)。

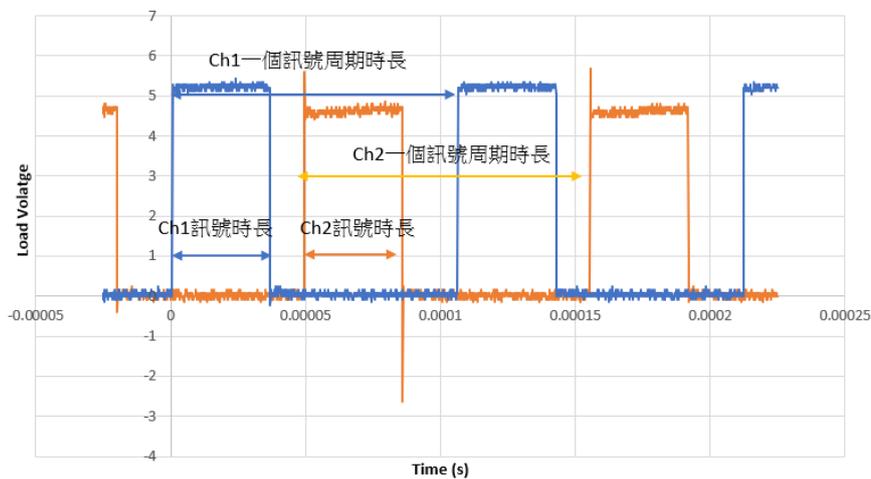


圖 1-3:量測訊號時間長短 CH1 和 CH2 圖例。

量測時做了三組數據，我們從三組數據中取出一組做討論，一組數據分別量測 10 次。從圖 1-4 量測數據統計可得知 CH1 和 CH2 的一個訊號時長大多落在 36.1~36.3 (us)，將大於平均的 110%的數據排除後，再將剩餘訊號數據做平均，得到 CH1 訊號的平均為 $36.20 \pm 0.01(\text{us})$ ，CH2 訊號的平均為 $36.20 \pm 0.01(\text{us})$ 。

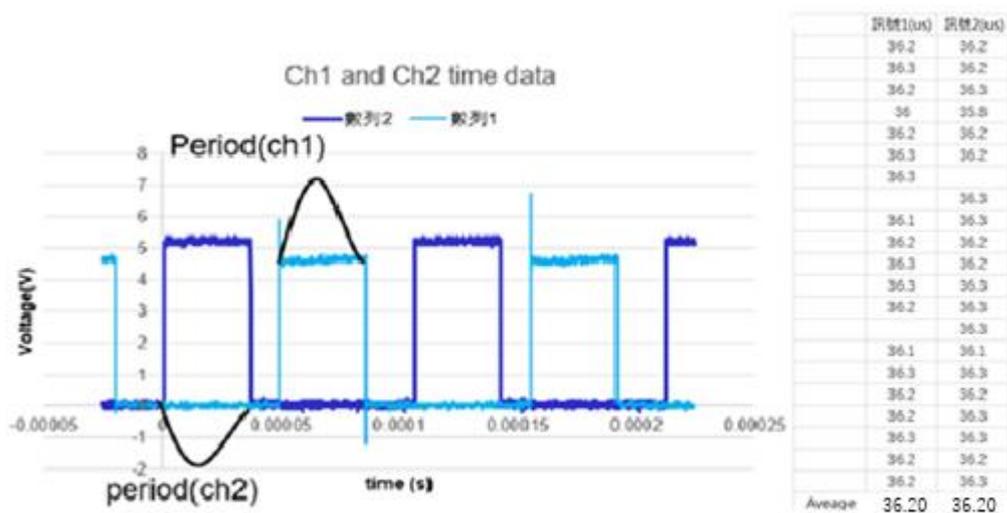


圖 1-4 為 CH1 和 CH2 訊號週期時長分析。

從圖 1-5 量測數據統計可得知 CH1 和 CH2 的一個訊號時長落在 105.7~106.0 (us) ，將大於平均的 110%的數據排除後，再將訊號數據做平均，得到 CH1 訊號得到的平均為 $105.9 \pm 0.2(\text{us})$ · CH2 訊的平均為 $105.9 \pm 0.2(\text{us})$ 。

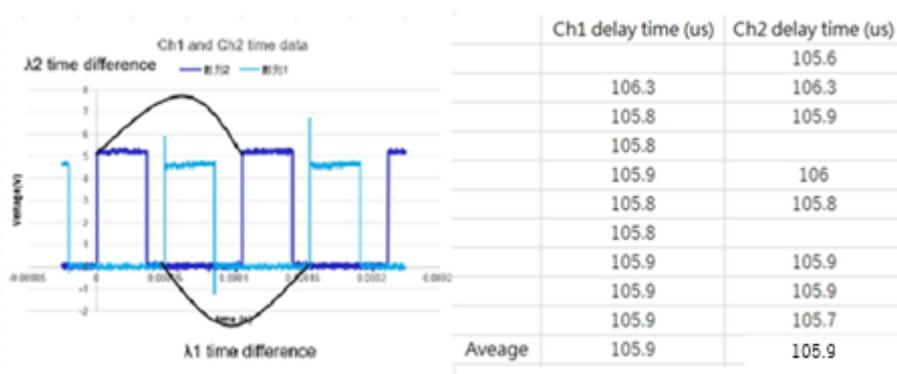


圖 1-5 是分析 CH1 和 CH2 一個週期時長。

量測時總共做了三組數據，一組數據分別量測 10 次，共有三組，量測了 30 組數據。分別有訊號時長和訊號週期長的分析。單位皆為 us，最後表 1 為所有的分析數。

表 1:量測訊號的 CH1 和 CH2 三組所量測數據統計。

		Ch1訊號時長(us)	Ch2訊號時長(us)	Ch1訊號週期時長(us)	Ch2訊號週期時長(us)
第一組	average (us)	36.8	36.31	106.2	107.1
	標準差	0.2	0.04	0.09	0.3
第二組	average (us)	36.6	36.8	106	106.5
	標準差	0.1	0.2	0.02	0.2
第三組	average (us)	36.2	36.2	106.4	106.4
	標準差	0.01	0.01	0.2	0.2

在將表 1-6 所統計三組數據取平均值，即可得到表 2 實驗數據平均值和標準差。

表 2:數據統計平均值和標準差

	Ch1訊號時長	Ch2訊號時長	Ch1訊號週期時長	Ch2訊號週期時長
量測訊號數據平均	36.5	36.44	106.2	106.7
標準差	0.1	0.08	0.1	0.2

(五)Summary

我們利用 Arduino 去產生一個 10k-Hz 的訊號，並將訊號做測試分析，分析出週期時長 CH1 訊號得到的平均為 36.5 ± 0.1 (us)、CH2 訊號得到的平均為 36.4 ± 0.08 (us)、一個週期時長 CH1 訊號得到的平均為 106.2 ± 0.1 (us)、CH2 訊號得到的平均為 106.7 ± 0.2 (us)。未來我們會利用這個 Arduino 的訊號控制高壓脈衝產生器中的 H 電橋(H-bridge)開關，讓直流電在負載中變成交流電的狀態。

二: 高壓脈衝產生器(High-voltage pulse generator)之製作

製作高壓脈衝產生器主要有兩個目的，第一點是可以配合實驗室現有的設備來產生電漿；第二點是配合整流器的組合使用製作大功率的電壓直流供應器。

製作高壓脈衝產生器，主要是由 H 電橋所組成，如圖 2-1. H 電橋由 4 個相同開關 T1、T2、T3、T4 所構成.系統有兩個輸出狀態：

在狀態 1 時:如圖 2-2 所示，T1、T4 導通 (closed)，T2、T3 為斷路 (open) 時，電流會從直流電流供應器 (Power supply) 流向 T1，由左往右經負載，接著流向 T4 最後流回直流電流供應端。

在狀態 2 時:如圖 2-3 所示，當 T1、T4 為斷路 (open) T2 和 T3 為導通 (closed)，電流會從直流電流供應器 (Power supply) 流向 T3，由右往左經負載，接著流向 T2 最後流回直流電流供應器。

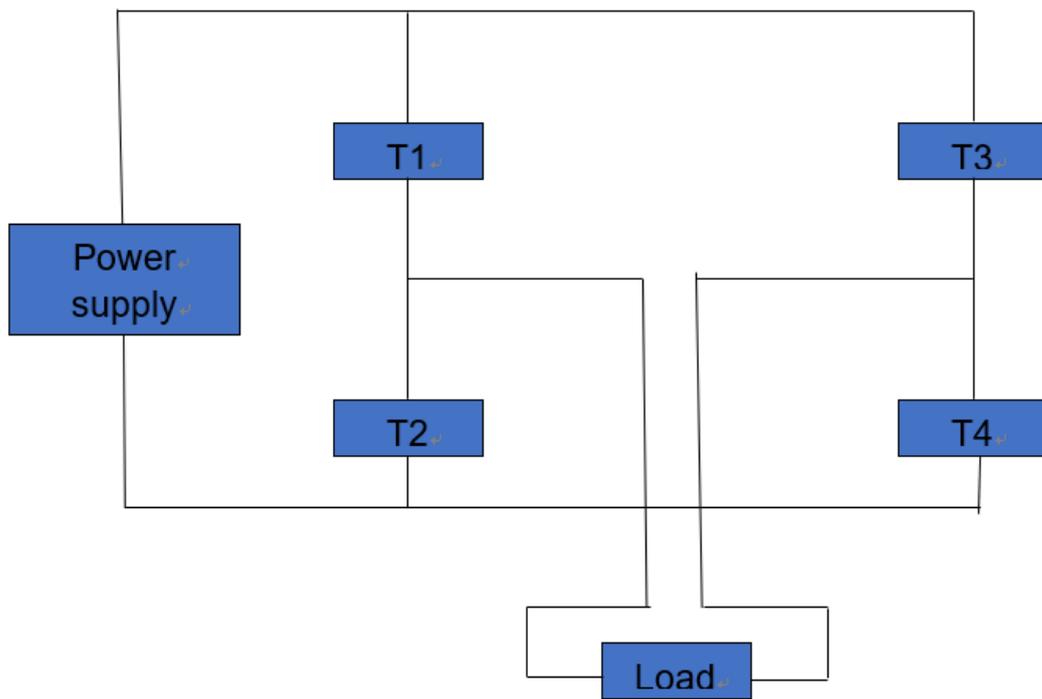


圖 2-1: H 電橋的架構。

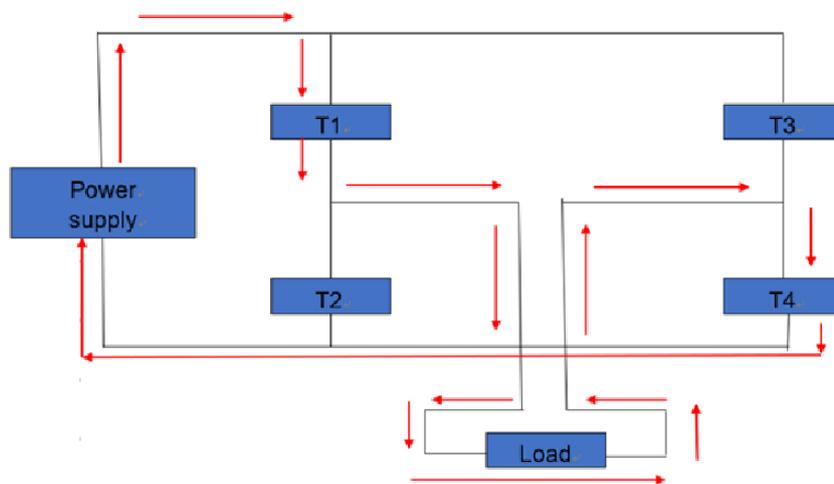


圖 2-2: 當 T2 和 T3 斷開，T1 和 T4 導通時的電流走向(紅色箭頭)

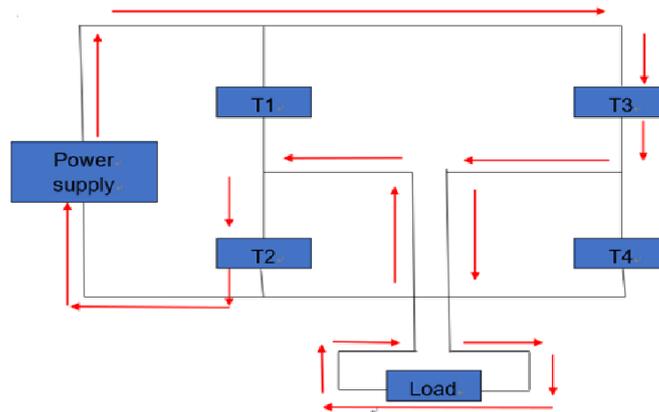


圖 2-3: 當 T1 和 T4 斷開，T2 T3 導通時的電流走向，(紅色箭頭)。

比較圖 2-2 和圖 2-3 可以看到，在狀態 1 和狀態 2 中，流過負載的電流方向相反，我們之後會利用此特性，將直流電流轉成交流電流。如圖 2-4 以示波器中量測負載電壓時可以知道，負載在接收到訊號 1 時會產生藍色線的圖型，當 load 在接收到訊號 2 時會產生橘色線的圖型。兩者訊號進入 load 的方向相反。

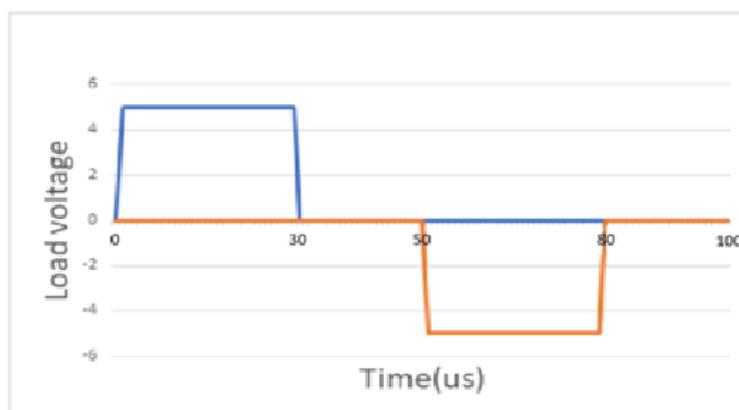


圖 2-4：負載在一個週期訊號時間接收狀態 1 訊號和狀態 2 訊號所產生的圖型。(狀態 1 訊號為藍色訊號，狀態 2 訊號為橘色訊號)

我們使用 IGBT 做為圖 2-1~2-3 的 H 橋架構中的開關。為了驅動 IGBT，我們需要四組 IGBT 驅動器(Driver of IGBT)。然而從圖 2-2 及 2-3 中可以看到 T1 和 T3 的電位會是浮動的，而 T2 和 T4 的電位則是接地，所以沒有浮動電壓的問題。如圖 2-5 所示，我們使用 4 組相同的 IGBT 驅動器(Driver of IGBT)，浮動電壓的部份由 DC-DC converter 承受。

因此，組成高壓脈衝產生器(High-voltage pulse generator)的元件有輸出控制訊號的 Arduino、將電訊號轉成光訊號的光纖發送器 HFBR-1528Z、將光訊號轉成電訊號的光纖接收器 HFBR-2528Z、隔離式的直流電壓轉換器(DC-DC convert)和做為開關的 IGBT。

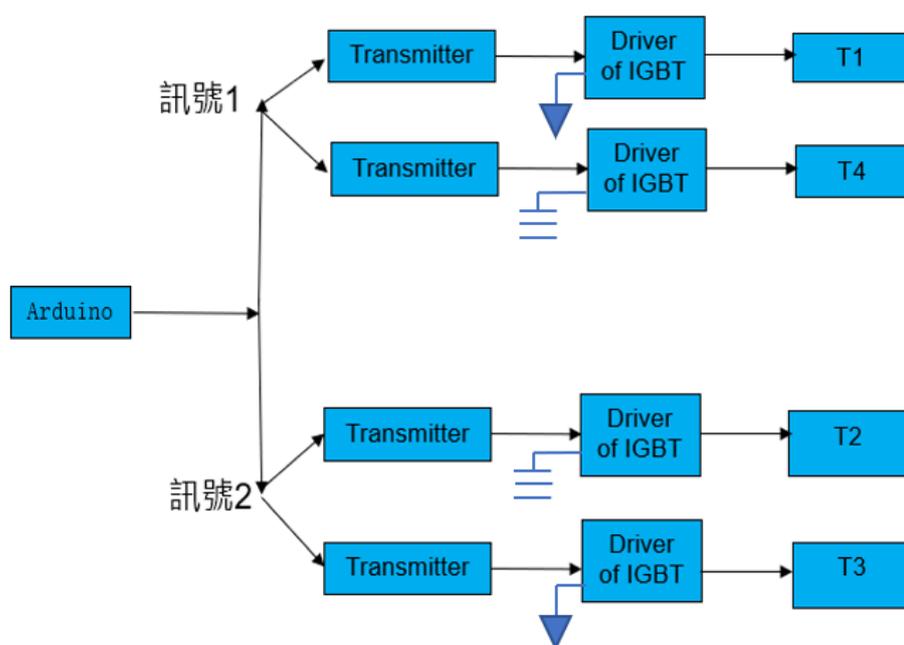


圖 2-5:Arduino 訊號輸出流程，先將訊號傳送給 transmitter，再將

訊號傳遞給 Driver of IGBT，最後傳給開關 T。

系統訊號流程圖如 2-6 所示。Arduino 負責輸出 10KHz 的訊號，透過 Transmitter 中的 HFBR-1528Z 將訊號轉成光訊號後經光纖傳到 Receiver 中的 HFBR-2528Z，HFBR-2528Z 會將光訊號轉成電訊號，再將轉換過來的電訊號傳送給作為開關用的 IGBT。直流電壓轉換器（DC-DC convert）會供應 15 伏特電壓讓系統運作，也提供了浮動電壓所需的絕緣效果。

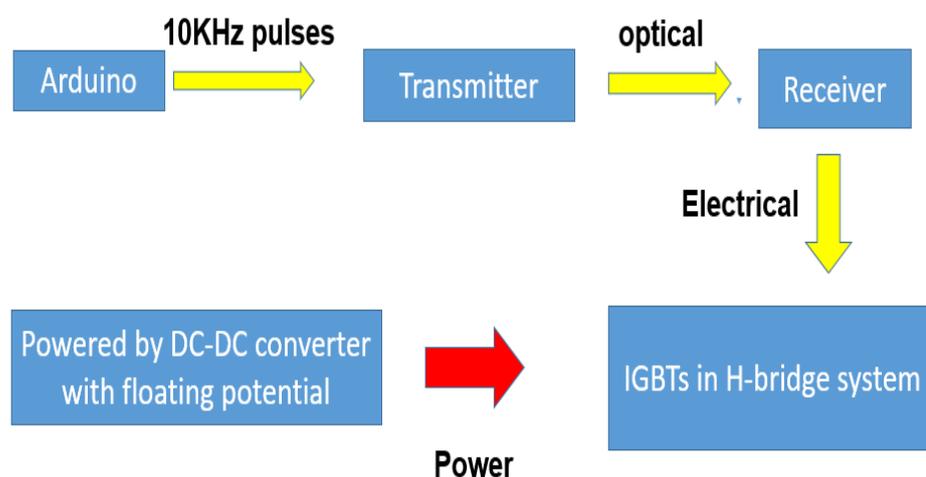


圖 2-6:系統訊號和 power 運作流程

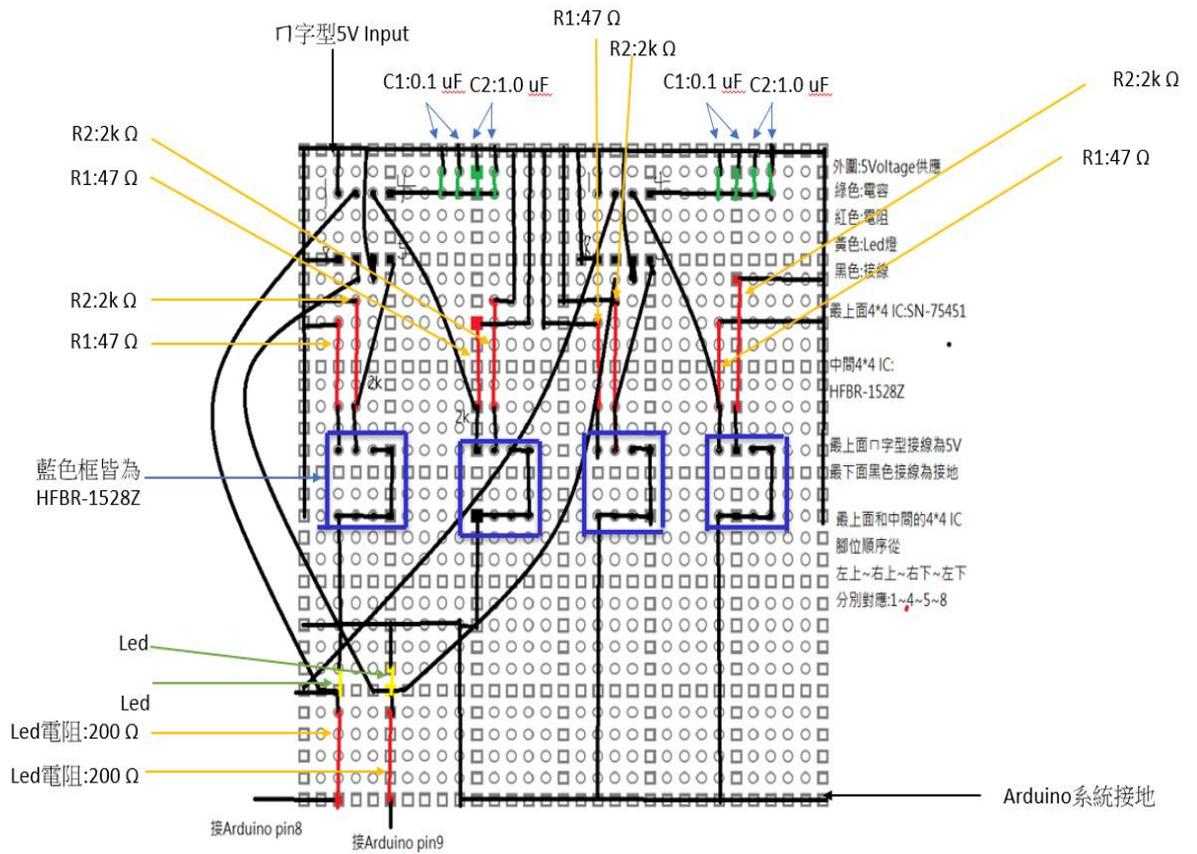


圖 2-8:HFBR-1528z Layout 圖



圖 2-9：Arduino 板實際完成圖

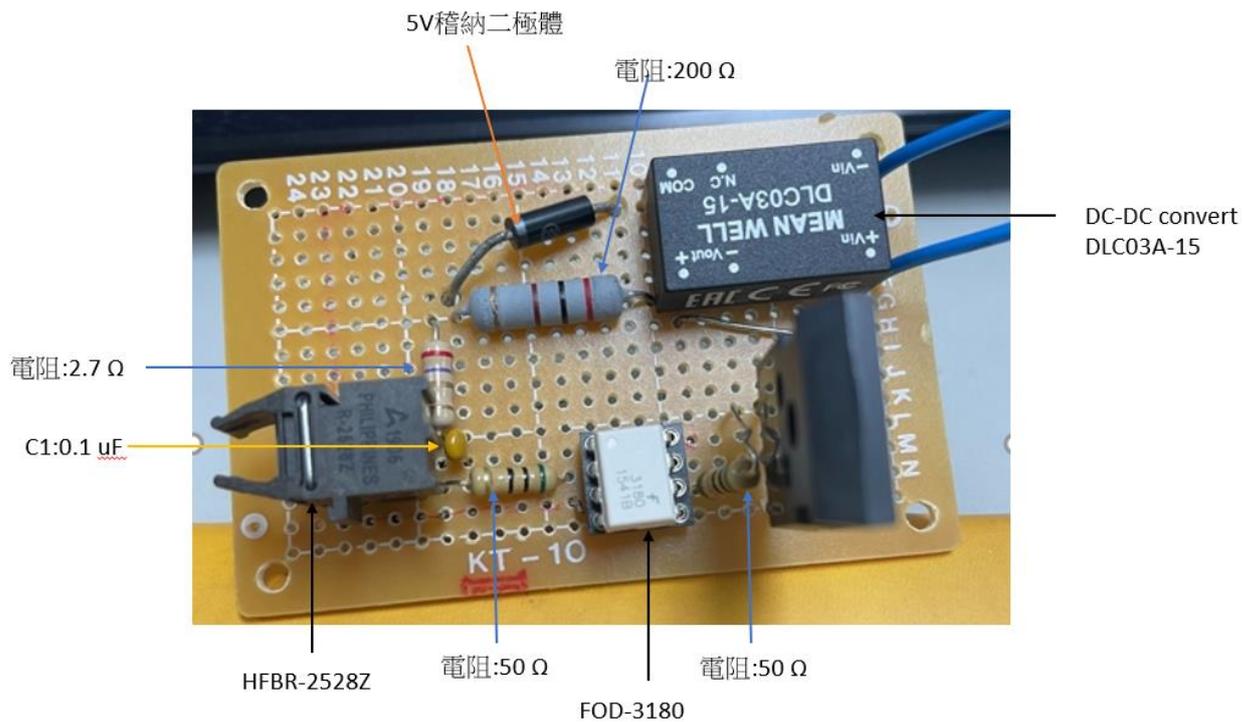


圖 2-11: HFBR-2528Z 實際對照圖 ·

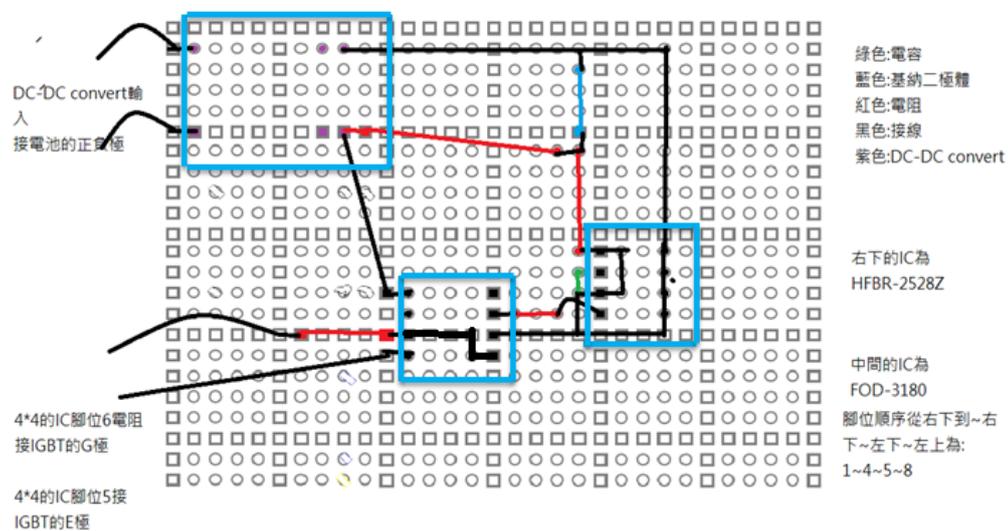


圖 2-12:HFBR-2528z Layout 圖 ·

三:校準夾具設定流程 & 軌道間隙開關安裝流程

(一):校準夾具設定流程

校準夾具是用來控軌道間隙開關的間距的工具。因為我們脈衝功率系統的北翼及南翼所使用的軌道間隙開關間隙設置剛好相反，所以需要調整。校準夾具設定流程夾具放置位置為 3-1 所示(靠近北翼的置物櫃)。

以北翼系統為例，如圖 3-2 為北翼的軌道間隙開關，圖 3-2 左邊半圓柱電極(南側)連接平行傳輸線，系統充電時為接地。右邊半圓柱電極(北側)連接電容，系統充電時為高壓。左邊半圓柱電極和中間觸發電極之間的距離，及右邊半圓柱電容極和中間處發電極之間的距離，兩者之間的比例為 1:2。相反的，南翼使用的軌道間系開關中，兩者的比例為 2:1。

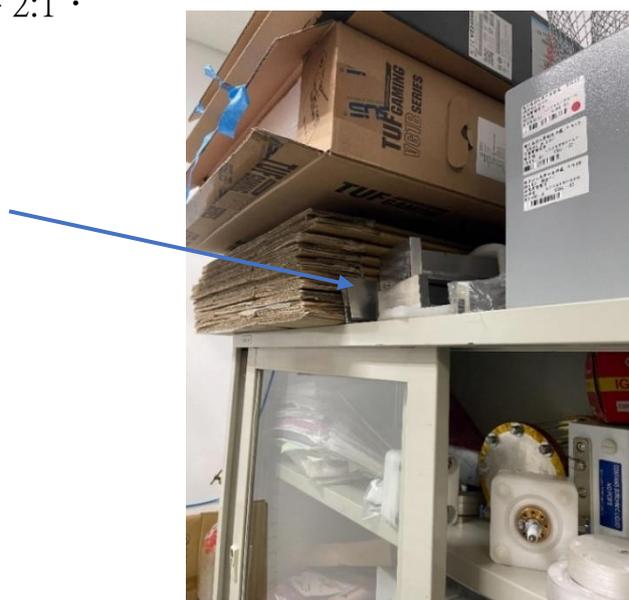


圖 3-1:T 型夾具放置位子.pulsed power 系統北側第一個櫃子上方。

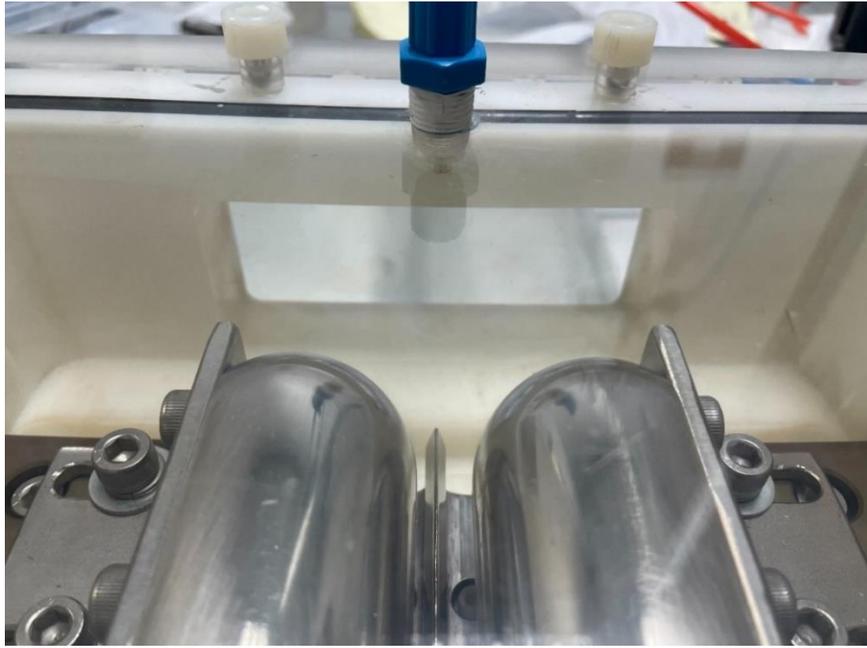


圖 3-2:半圓柱電極和校準夾具的距離比例為 1，右側半圓柱電極(北方)和校準夾具距離比為 2。

設定流程;以北翼為例:

- 1：將圖 3-3 校準夾具的側板拆下。
- 2：如圖 3-4 中，將校準夾具較薄的一側(a 側)朝南放置。
- 3：將校準夾具的側板從西側鎖上。



側板

圖 3-3：校準夾具。

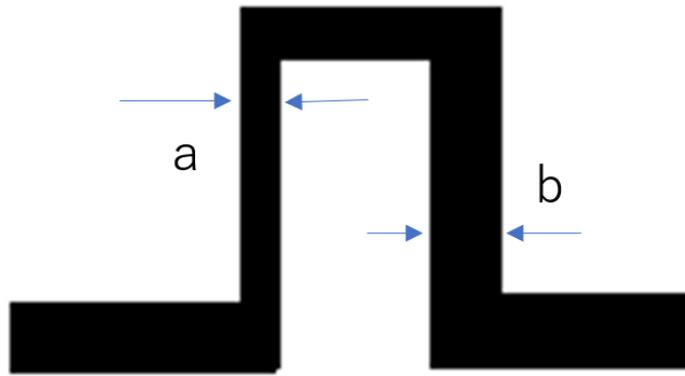


圖 3-4：校準夾具側面示意圖。a 側為較薄一側。b 側為較厚一側。

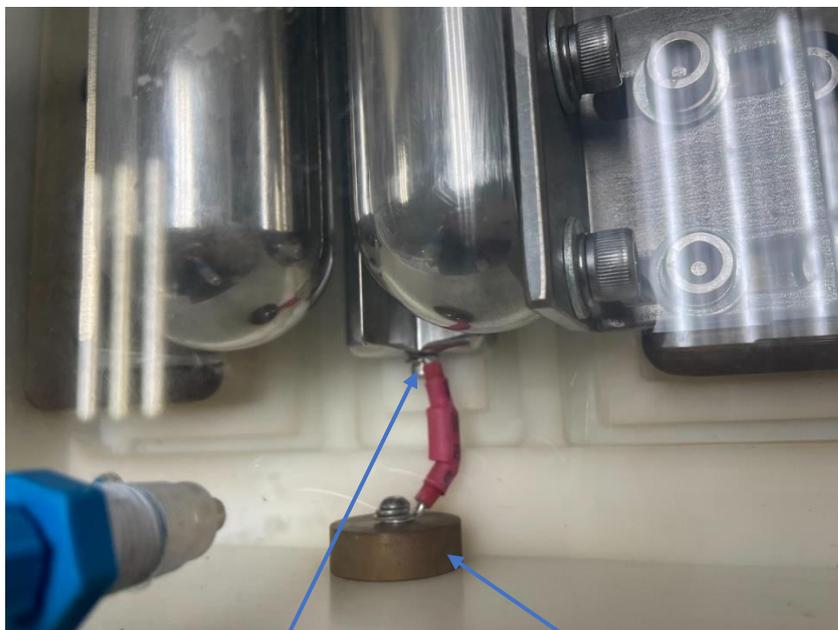
(二):軌道間隙開關安裝流程

使用器材

- 1:乳膠手套 ·
- 2:異丙醇 (Isopropanol IPA) ·
- 3:六角扳手 ·
- 4:校準夾具(T 型金屬板) ·

以北翼的系統為例

- 1：戴乳膠手套 ·
- 2：將軌道間系開關內部用 IPA 擦拭過一次 ·
- 3：將電極導線鎖在(觸發電極)上，如圖 3-5 的箭頭 A 所示 ·



箭頭 A:觸發電極 ·

箭頭 B:觸發輸入電極 ·

圖 3-5(紅色為電極導線) ·

- 4：將觸發電極放置好。
- 5：將觸發電極的螺絲鎖上。
- 6：將電極導線接到觸發輸入電極上，並且鎖好，如圖 3-5 箭頭 B 所示。
- 7：使用三用電表量測電極導線鎖上的兩側，確認觸發電極及觸發輸入電極(圖 3-5 箭頭 B 所示)之間有導通。
- 8：先將圖 3-6 中西側的長方形壓克力板撤下。
- 9：完成第(一)節的夾具設定流程。
- 10：將較準夾具從窗口慢慢推入至底，確認推至最底之後，是否與觸發電極的兩側緊密貼合。

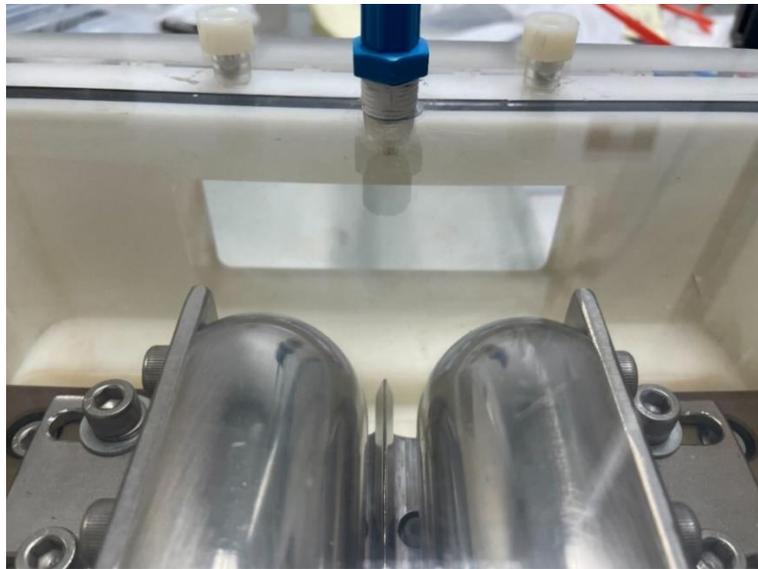


圖 3-6：此圖為北翼軌道間隙開關，西側的長方形窗，用來把校準夾具放進，兩個半圓電極中間的 T 型電極為觸發電極。

- 11：安裝靠近系統側 L 形金屬板，L 形較長的部分放在底部，利用 L

形金屬板最兩旁的螺絲和半圓柱電極鎖上，螺絲只需要鎖上最旁邊兩側，(如圖 3-7 所指的兩對螺絲)。



圖 3-7：半圓形電極與軌道間隙開關下方利用接著的是 L 形金屬板相連，L 形金屬板較長的部分與下方相連，較短的部分接半圓柱電極。

12：將半圓形電極安裝在已放入軌道間隙開關的 L 形金屬板的短邊上，一樣先鎖上半圓柱電極和 L 形金屬板交接的最旁邊兩側螺絲，方便調整。需注意半圓形電極不要撞到已放入的校準夾具，撞到需要重磨半圓柱電極。

13：將半圓柱電極和校準夾具三面貼合，如圖 3-7。順序分別為；

(a)半圓柱電極和校準夾具 A 面貼合。

(b)半圓柱電極和校準夾具 B 面貼合。

(c)確認半圓柱電極和校準夾具兩面貼合。

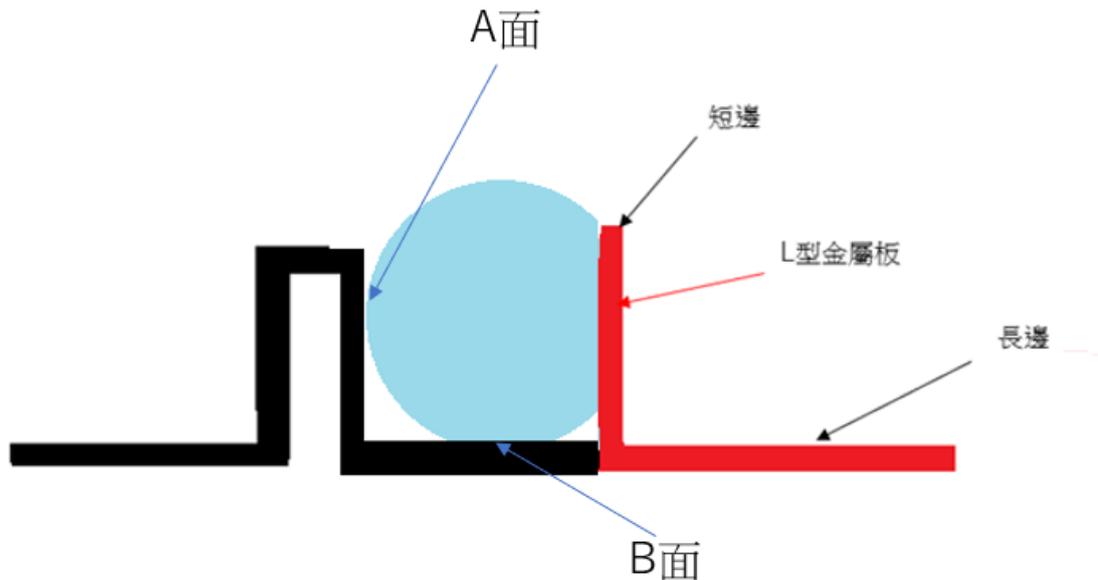


圖:3-7：圖例，半圓柱電極和校準夾具三面貼合。

- 14：貼合後，將軌道間系開關 L 型金屬板的螺絲一一鎖上鎖緊。
- 15：檢查電極和校準夾具是否完全貼合，方式為拿一張 A4 紙，從電極和校準夾具之間放入 A4 紙，確認是否可以插入，如果可以將紙張放入的話，重複步驟 12，直到不能將紙張插入為止。
- 16：重複 11~15 安裝步驟，安裝軌道間系開關上另一側的 L 型金屬板和半圓柱電極。
- 17：校準夾具從西側的開口抽出，放回原位。
- 18：用 IPA 擦拭 L 型金屬板，半圓柱電極，觸發電極和周圍盒內
- 19：將 O-ring 放入軌道間隙關關上方的凹槽。
- 20：將系統上的塑膠蓋子蓋上。
- 21：蓋子上的螺絲放好後並鎖緊，鎖螺絲的一次鎖兩個在對角的螺絲，避免 O ring 被不均勻形的壓縮。

22：將西側的長方形壓克力板放置好後，用螺絲一一鎖緊。

四:逃生路線圖之更新

為了避免小型火災發生，用二氧化碳型滅火器可以直接撲滅，減少用泡沫型滅火器機會，使用後清理較方便.所以於 2021 年底時我們購買二氧化碳滅火器放置於實驗室，如圖 4-1，所以需要將二氧化碳滅火器放置位置放入逃生路線中。原版逃生路線如圖 4-2，修正後如圖 4-3。



圖 4-1:左邊為 5 型二氧化碳，放置於高壓測試箱和抽氣櫃之間
右邊為 10 型二氧化碳，放置於無塵手套箱下方。

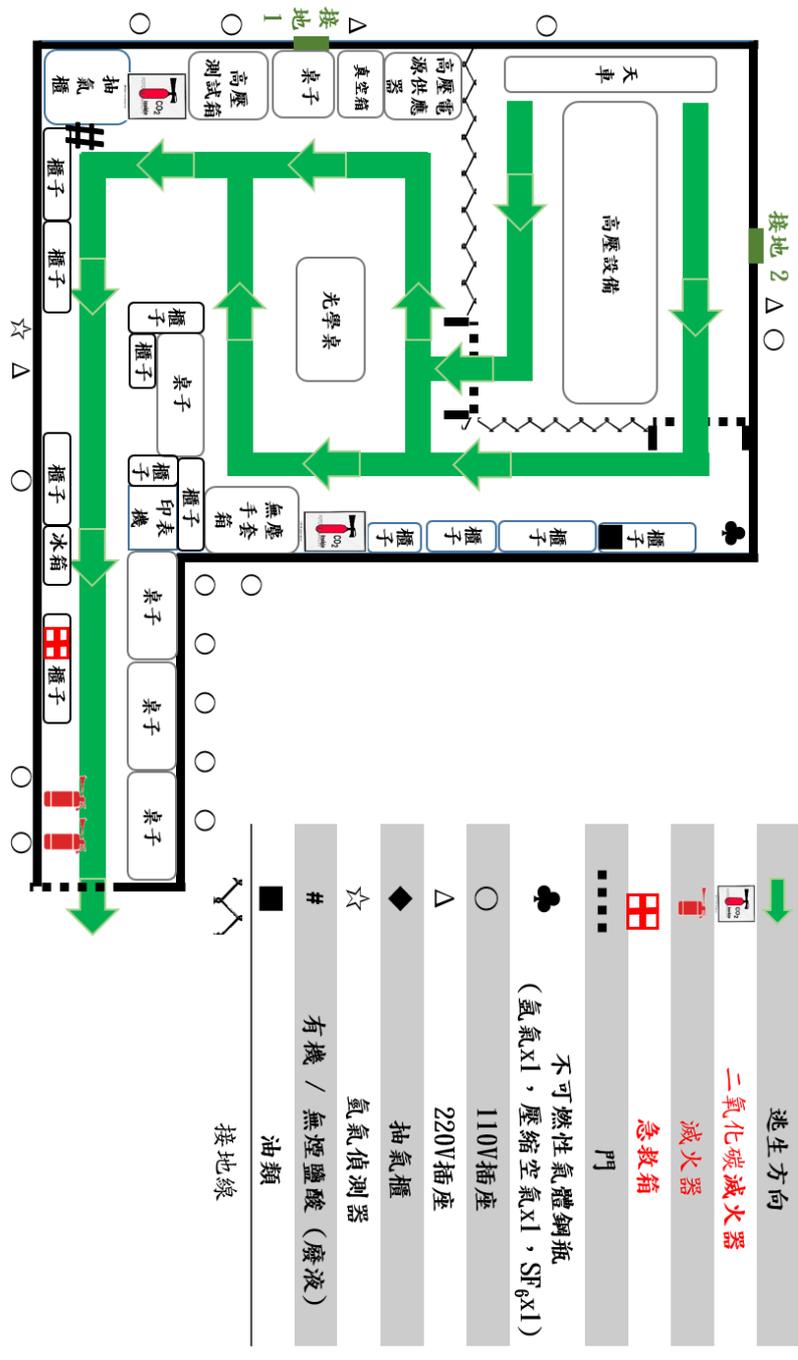


圖 4-3:更新後的逃生路線圖。

5:未來工作

目前高壓脈衝產生器進度剩下 IGBT 的電路設計和穩壓電容以及電阻的購買。未來會開始著手有關實驗室的工作。短期目標在學習 Fusion 360 和碩一的課業上，以便碩二可以專心做自己的工作。

6:結論

我在上學期完成的事項主要有練習 Arduino 的運用，並利用 Arduino 送出訊號到高壓脈衝產生器上，且測量的訊號是在可接受的誤差範圍內。另外在空檔時間內將軌道電極進行拋光工作，且對軌道間系電極製作了安裝流程。在實驗室內添購了二氧化碳滅火器，且重新製作了逃生路線。最後是高壓脈衝產生器的製作，我會在下學期開學前完成。好能銜接下一步工作事項。

附錄一

Arduino code

```
void setup() {
```

```
pinMode(8, OUTPUT);      設定腳位 8 為 Led 1 輸出腳位
```

```
pinMode(9, OUTPUT);      設定腳位 9 為 Led 2 輸出腳位
```

```
}
```

```
void loop() {
```

```
digitalWrite(8, HIGH);    設定 Led 1 為導通
```

```
digitalWrite(9, LOW);     設定 Led 2 為關閉
```

```
delayMicroseconds(30);    設定此次行動的時間為 30us
```

```
digitalWrite(8, LOW);     設定 Led 1 為關閉
```

```
digitalWrite(9, LOW);     設定 Led 2 為關閉
```

```
delayMicroseconds(10);    設定此次行動的時間為 10us
```

```
digitalWrite(8, LOW);     設定 Led 1 為關閉
```

```
digitalWrite(9, HIGH);    設定 Led 2 為導通
```

```
delayMicroseconds(30);    設定此次行動的時間為 30us
```

```
digitalWrite(8, LOW);    設定 Led 1 為關閉  
digitalWrite(9, LOW) ;  設定 Led 2 為關閉  
delayMicroseconds(10);  設定此次行動的時間為 10us  
}
```