

嘉義市第 37 屆中小學科學展覽會
作品說明書

科別：生活與應用科學（一）科

組別：國中組

題目：亮眼護眼 — 教室 AI 控光系統

關鍵詞：AI 控光系統、硬體結合裝置、等比例縮小模

擬教室

編號：

摘要

本研究以增進照明的護眼及節能為目標，設計出一套整合型的系統達到自動去保護眼睛且減少不必要的電量浪費。將感知用的硬體設備與自行開發的程式作結合，使系統能夠在感知到當下的不同狀態後，做出相對應的調整。可控制的範圍包含：1. 教室內亮度不足時可自動提高亮度；2. 教室內亮度過高時可自動降低亮度；3. 可區域性做獨立的自動調整；可以達到節能護眼的功效。實驗利用 Arduino 及照度模組結合成本裝置，之後直接對應閱讀照度標準，再利用 Arduino 的 PWM 控制方式研究自動調控 LED 照度的方法，改善在閱讀時的光線不足，讓眼睛不受亮度過高或過暗而產生的傷害，亦可以達到節電省電的效果，為任何場所皆需要的裝置系統。

壹、研究動機

常常在陽光充足的情況下，教室裡的燈仍然只能全開時，書的表面都會被光反射，使眼睛感到不舒服，並造成對眼睛的些許的傷害。而在烏雲密布的陰天裡燈光的照度不足時，會使我們在閱讀時使眼睛疲勞且吃力，這同樣會對眼睛造成相當的傷害，因此若能有可以判斷亮度是否充足並做出提醒且自動調整照度的程式裝置，就能讓環境不會因太亮或太暗使眼睛感覺到不舒服了。於是在與老師，討論之後決定使用 Arduino UNO R3 加上能感應照度的模組並連接到電腦上的組合程式來改善我們發現校園所面臨到的問題。

貳、研究目的

- 一、以照度模組連接 Arduino 板測試照度的校正
- 二、自動偵測與開關調控在等比例縮小教室的測試情形
- 三、自動偵測和開關調控系統運用於實體教室中
- 四、測試 Arduino PWM (Pulse Width Modulation) 以不同線路連接 LED 燈管

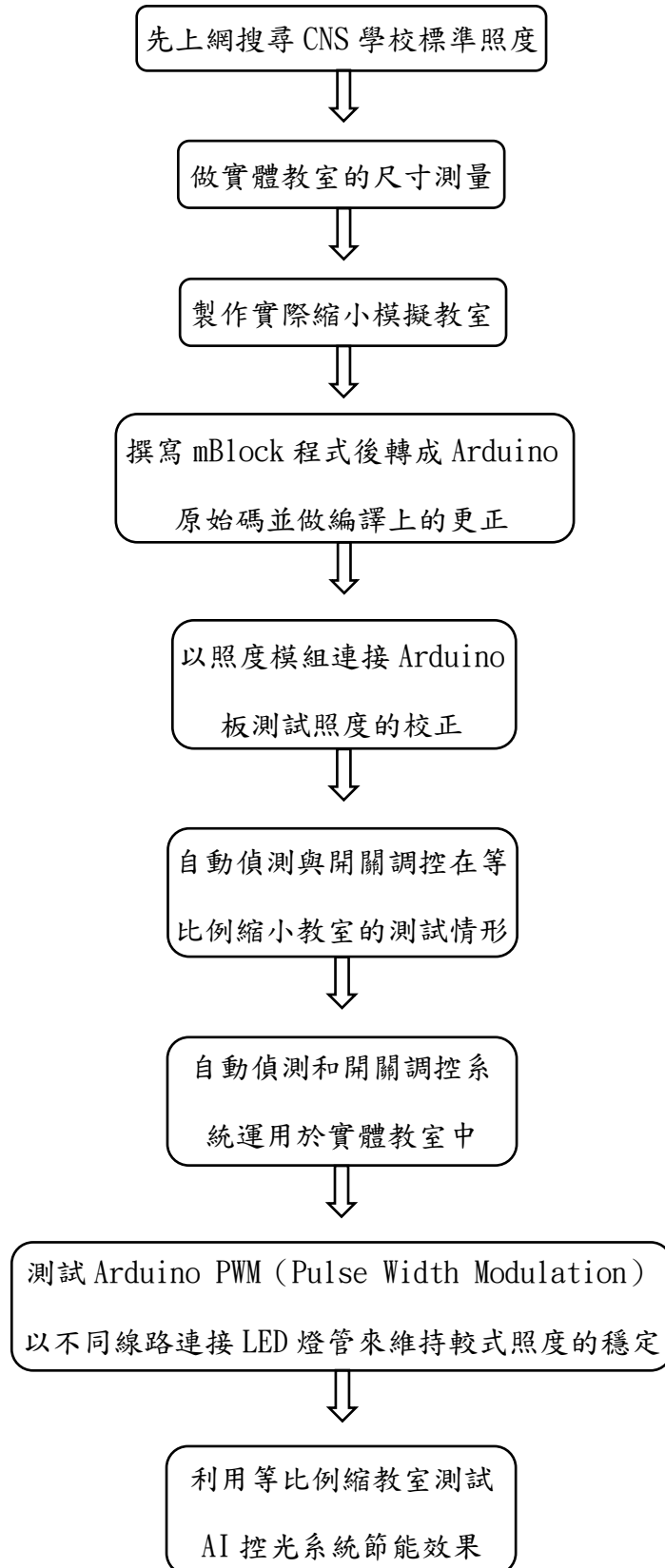
來維持較式照度的穩定

五、利用等比例縮教室測試 AI 控光系統節能效果

參、 研究設備與器材

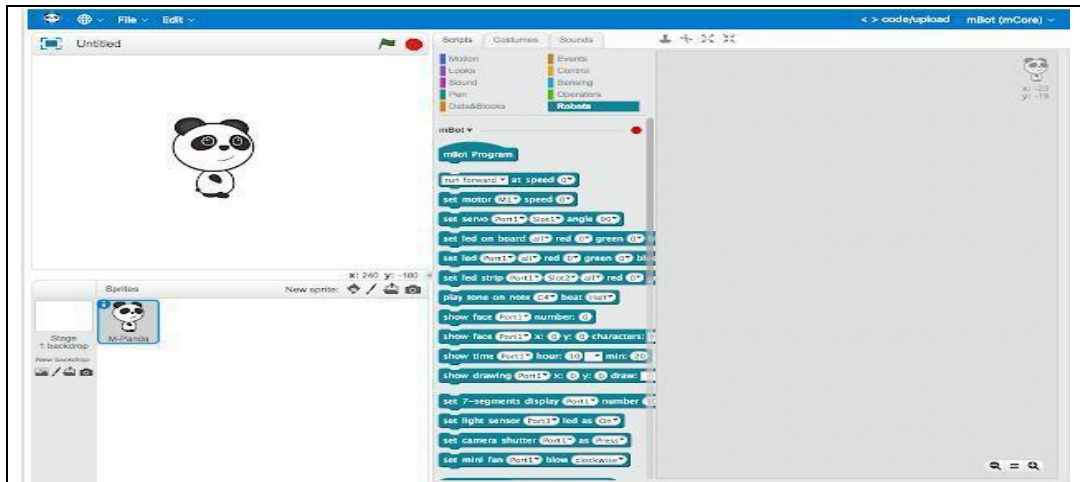
| | | |
|---|---|---|
|  |  |  |
| Arduino UNO R3 | 照度模組 GY-30 BH1750 | 麵包板 |
|  |  |  |
| 照度計 | 電腦 | 1:31 等比例縮小模擬教室 |
|  |  |  |
| LED 燈 | 繼電器 (DC-DC) | Arduino 繼電器 |
|  |  |  |
| 電晶體 | LED 燈管 | 杜邦線 |
|  |  |  |
| 電阻 (1000Ω) | 鱷魚夾 | 相機 |

肆、 研究過程與方法



一、程式的編輯流程：

(一) 以 mblock (MakeBlock 有限公司) 程式編輯軟體依照照度模組 GY-30 (台灣物聯科技有限公司) 的讀取及輸出程式先做編寫，接著再將所設計的自動調光系統寫入程式。



mblock 編輯積木程式

(二) 將在 mBlock 撰寫好的自動調光系統於軟體中在“Arduino 主程式”點選右鍵，選擇“上傳 Arduino 程式”，則 Arduino 程式碼將會顯示在右方，並將其複製。



mblock 程式轉 Arduino 編碼

(三) 將複製下來的程式碼於 Arduino 程式編輯軟體貼上並和 BH1750 的範例程式做合併，再做編譯上的更正。

```
PWM_ | Arduino 1.8.4
檔案 編輯 庫 工具 說明
PWM
#include <Wire.h>
#include <Arduino.h>
#include <SoftwareSerial.h>
#include <BH1750.h> //照度模組一
#include <HL1750.h> //照度模組二

BH1750 middle; //照度模組一
HL1750 side; //照度模組二
double gmax;
double gmin;
double mppm;
double spmm;
double range = 10;

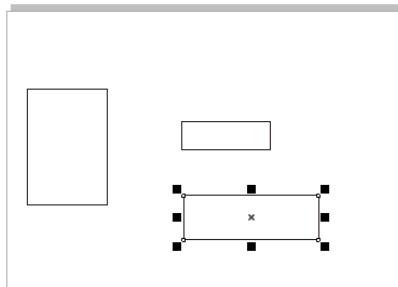
void setup()
{
  Serial.begin(9600);
  Wire.begin();
  middle.begin(); //照度模組一
  side.begin(); //照度模組二
}
```

編碼後與 Arduino BH1750 範例程式合併

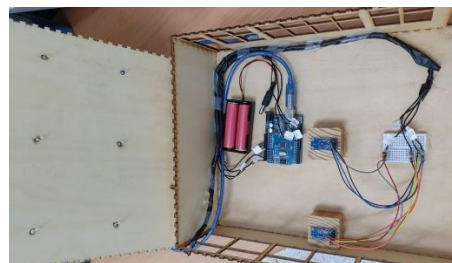
二、等比例縮小模擬教室的製作

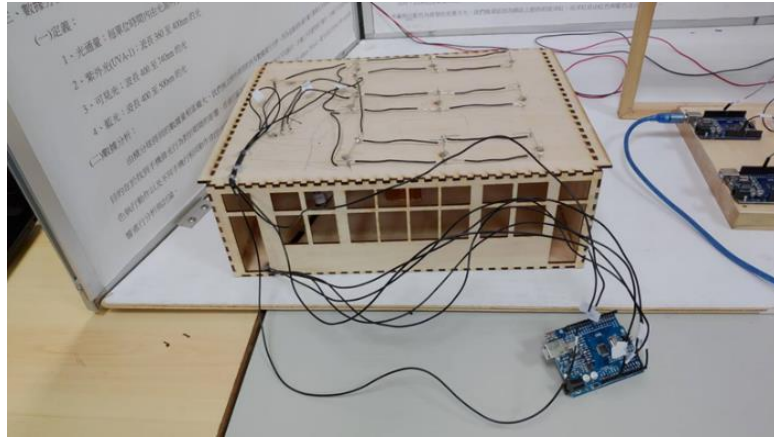
(一) 於實際的教室測量尺寸後選定 1:31 的比例並做四捨五入法至小數一位

(二) 在雷切的繪圖軟體上劃出算好尺寸的教室面形，接著列印出來並組裝



(三) 將 LED 燈、Arduino 板及線路做連接



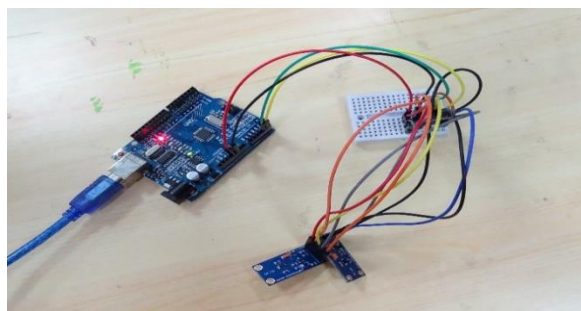


等比例縮小模擬教室成品圖

一、以照度模組連接 Arduino 板測試照度的校正

(一)並開始將其程式修改成 CNS 規定的標準教室閱讀亮度的上限和下限數值

發現由於照度模組能夠直接輸出標準單位 Lux 值所以他的值能夠非常的趨近於照度計的值，且也不會因為延時或重新連接而導致數值的差異。在我們的實驗比較後，由於照度模組的值都和照度計測量到的值差距大都會一致，且在實驗過程中我們使用到的兩個照度模組其一(照度模組 1)的值與照度計均相差約 60Lux，而另一個(照度模組 2)則和照度計均相差約 45 Lux，因此證明其值明顯穩定。



Arduino 連接兩塊照度模組。

二、自動偵測與開關調控在等比例縮小教室的測試情形

(一) 等比例縮小模擬教室的製作

(二) 程式的撰寫按程式的編輯流程編寫

寫好的自動調整開關使亮度能夠因應亮度的不同，但此實驗僅能在開關上做調整，而在標準範圍值內時，則會使燈光閃爍作為告知。我們使用 Arduino 連接兩塊照度模組並分別放在等比例縮小教室的其中一測與中央，當燈高於上限數值時燈就會自動關閉，而低於下限值時燈就會自動開啟，達到自動偵測及控制開關的功能。由於 Arduino 的連接埠只有兩個，所以只能放置於中間，而兩側的部分則只能選一側作為實驗對象(在無特殊情況下兩邊的效果將會相似)，若要將其真正運用至教室的話則可使用兩塊 Arduino 板或修改序列埠

三、自動偵測和開關調控系統運用於實體教室中

(一) 程式的撰寫按程式的編輯流程編寫，並開始將其程式修改成 CNS 規定的標準教室閱讀亮度的上限和下限數值

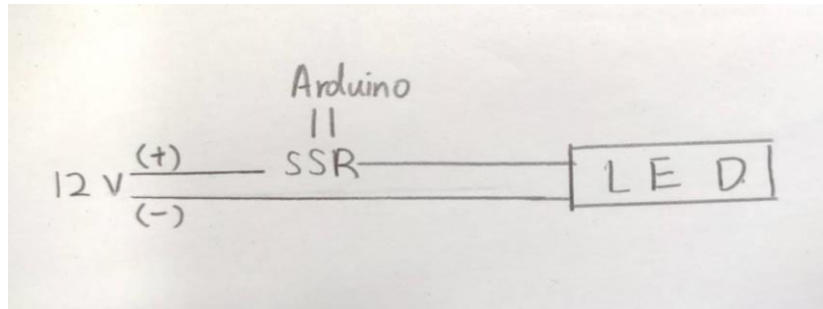
直接使用我們已經寫好的自動調整亮度開關程式使亮度能夠因應實體教室內亮度的不同，之後的方法與實驗步驟二相同，這次則是要控制實體的開關，需額外嘗試連接繼電器及連接 110 伏特的電路，使其能夠正常的依照我們所寫的自動調整亮度開關程式，來為控制教室日光燈的開關做調控。



將火線和地線加上繼電器並連接 Arduino 及照度模組進行實驗。

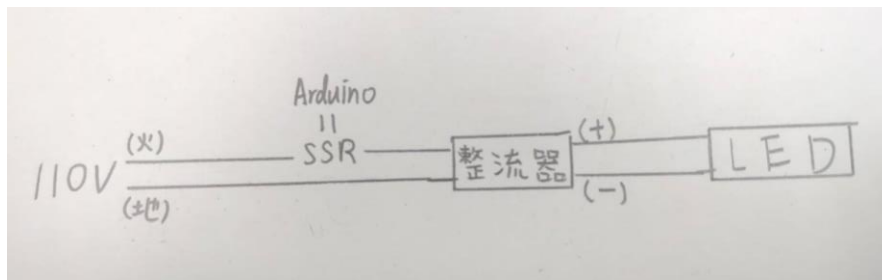
四、將測試 Arduino PWM (Pulse Width Modulation) 以不同線路連接 LED 燈管來維持較式照度的穩定

(一) 我們使用電源供應器輸入的 12V 電連接用 Arduino 控制的繼電器作為控制 LED 燈管的開關，由於繼電器只能控制開關，所以無法使用 PWM 調節照度的功能。



實驗(一)裝置線路圖。

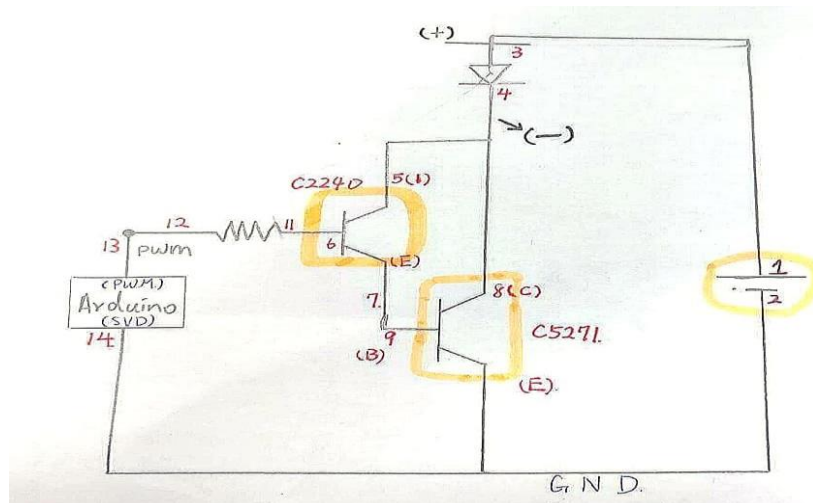
(二) 我們的這個裝置是由插頭輸入的 110V 的電加上繼電器和整流器再連接 LED 燈管所形成的電路，之後用 Arduino 的 PWM 程式使其能夠使用做到能夠調節燈光的方式，但其可調的照度數值範圍很小，並未達到我們所理想的效果。



實驗(二)裝置線路圖。

(三) Arduino PWM 加上達靈頓電路後連接 LED 燈管，但因反應不夠靈敏會不停地閃爍，因此將其再連接繼電器使反應能更加快速，並且我們改用更小的時間單位百萬分之一秒（程式語法為 `delayMicroseconds()`），之後我們從全開且熄滅時間為零的 `delayMicroseconds(100, 0)` 開始測試，當測試到 `delayMicroseconds(20, 80)` 時還能夠持續亮著而不產生閃爍，但當我們測試

到 delayMicroseconds(19, 81)時，便會開始閃爍，因此我們的 LED 省電調光只能到開 20 微秒並關 80 微秒，接著這樣持續循環而不閃爍，並且也同時發現其可調的數值範圍明顯比上一個實驗還大，所以其結果很符合我們的期待。



實驗(三)裝置線路圖

因為原本的 LED 燈管是串聯的，所以之後我們將其改換成並聯的 LED 燈條實驗，結果我們發現並聯的 LED 所能調整的 Lux 範圍比串聯的來的要更大。由於坊間賣的大部分都是串聯的 LED 燈管，所以並聯的較不容易取得，因此價格也比串聯的較高一些。

五、將完整的自動偵測調光系統運用於等比例縮小的教室中

(一)接續從實驗四的結果後應用在實驗二不足的地方，由於原本的模擬只能到全開和全關，所以此實驗的做法相似於實驗六，其與繼電器相連接後無論大於或小於都不是直接開或關燈，而是一次朝標準的區間值調整固定的值。如五或十，此效果更好於 PWM 的效果。

完整自動操控的縮小教室程式碼：

```
// 函式庫
```

```

#include <Wire.h>
#include <Arduino.h>
#include <SoftwareSerial.h>
#include <BH1750.h> // 加入照度模組一的函式庫
#include <BL1750.h> // 加入照度模組二的函式庫

// 初始宣告
BH1750 middle; // 照度模組一
BL1750 side; // 照度模組二
double gmax; // 最大值代號
double gmin; // 最小值代號
double mpwm; // 中間的 PWM 代號
double spwm; // 側邊的 PWM 代號
double range = 10; // 每次做亮度調整的 range 值 (可依需求調整)

void setup(){ // 啟動設定
  Serial.begin(9600); // 設定通訊速率
  Wire.begin( ); // 啟動 Wire 庫
  middle.begin(); // 照度模組一
  side.begin(); // 照度模組二
  gmax = 120; // 設定接收照度的最大值 (可依需求調整)
  gmin = 100; // 設定接收照度的最小值 (可依需求調整)
  mpwm = 0; // 設定中間 LED 的 PWM 初始值 (可依需求調整)
  spwm = 0; // 設定側邊 LED 的 PWM 初始值 (可依需求調整)
  pinMode(11, OUTPUT); // 設定腳位 11 輸出
  pinMode(10, OUTPUT); // 設定腳位 10 輸出
  digitalWrite(10, 0); // 熄滅腳位 10 的 LED
  digitalWrite(11, 0); // 熄滅腳位 11 的 LED
}

void loop() { // 程式迴圈
  uint16_t mlux = middle.readLightLevel(); // 設定照度模組一的讀取來源
  uint16_t slux = side.readLightLevel(); // 設定照度模組二的讀取來源
  Serial.print("mpwm="); // 通訊顯示字幕
  Serial.print(mpwm); // 通訊顯示中間 LED 的 PWM 值
  Serial.print(" spwm="); // 通訊顯示字幕
  Serial.print(spwm); // 通訊顯示側邊 LED 的 PWM 值
  Serial.print(" mlux:"); // 通訊顯示字幕

```

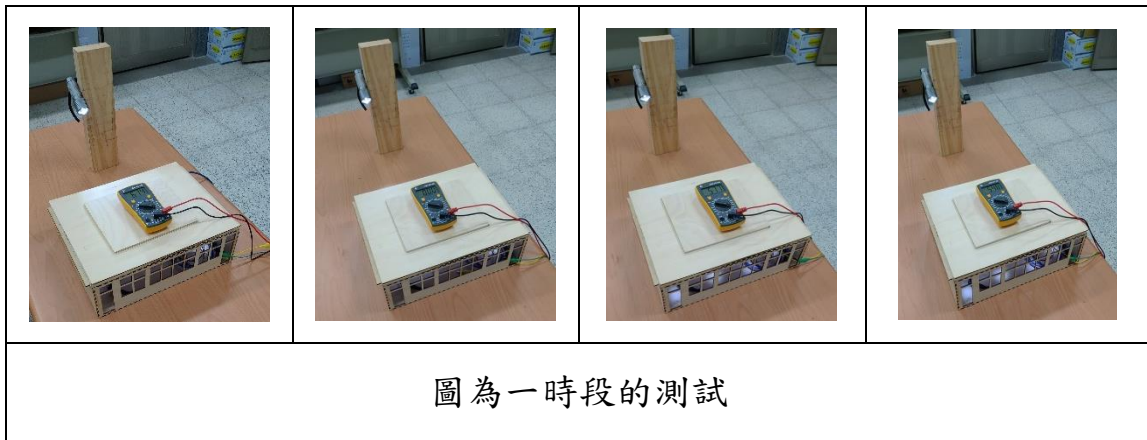
```

Serial.print(mlux);           // 通訊顯示中間讀取的照度值
Serial.print("  slux:");     // 通訊顯示字幕
Serial.println(slux);       // 通訊顯示側邊讀取的照度值
Serial.print("mlux: ");     // 通訊顯示字幕
Serial.print(mlux);         // 通訊顯示中間讀取的照度值
Serial.print("      slux: "); // 通訊顯示字幕
Serial.print(slux);         // 通訊顯示側邊讀取的照度值
delay(1000); // 停一秒
if((mlux) < (gmin)){ // 如果中間讀取的照度小於設定的照度最小值時
    mpwm += range; // 中間 LED 的 PWM 值加上設定的 range 數
    if((mpwm) > (255)){ // 如果中間的 PWM 值大於 255 時
        mpwm = 255; // 設定其 PWM 的值等於 255 (PWM 最大值為 255)
    }
}
if((mlux) > (gmax)){ // 如果中間讀取的照度大於設定的照度最大值時
    mpwm += -range; // 中間 LED 的 PWM 值減掉設定的 range 數
    if((mpwm) < (0)){ // 如果中間的 PWM 值小於 0 時
        mpwm = 0; // 設定其 PWM 的值等於 0 (PWM 最小值為 0)
    }
}
if((slux) < (gmin)){ // 如果側邊讀取的照度值小於設定的照度最小值時
    spwm += range; // 側邊 LED 的 PWM 值加上設定的 range 數
    if((spwm) > (255)){ // 如果側邊的 PWM 值大於 255 時
        spwm = 255; // 設定其 PWM 的值等於 255
    }
}
if((slux) > (gmax)){ // 如果側邊讀取的照度值大於設定的照度最大值時
    spwm += -range; // 側邊 LED 的 PWM 值減掉設定的 range 數
    if((spwm) < (0)){ // 如果側邊的 PWM 值小於 0 時
        spwm = 0; // 設定其 PWM 的值等於 0
    }
}
analogWrite(10, mpwm); // 設定腳位 10 的值等於 mpwm
analogWrite(11, spwm); // 設定腳位 11 的值等於 spwm
}

```

(二) 用等比例縮小模擬教室測量在具有 AI 控光系統和無控光系統的狀態下分別測量所使用的平均電量，並將數值四捨五入至小數點第一位，並計算省電的百

分比。實驗以每小時一次，到實際教室的窗口測量其照度，並至虛擬教室斜過手電筒達到實際照度，並計算電流。

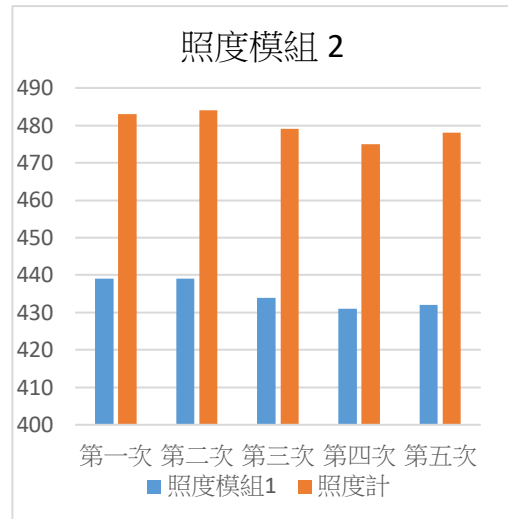
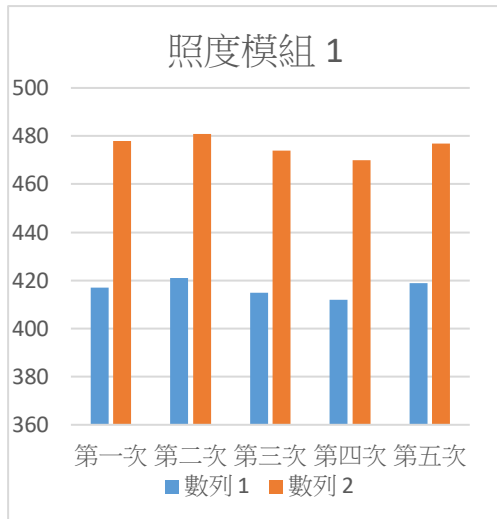


伍、 研究結果與討論

一、以照度模組 GY-30 做為連接 Arduino 測量照度的儀器。

| 測量次數 | 照度模組 1 | 照度計 |
|------|--------|-----|
| 1 | 417 | 478 |
| 2 | 421 | 481 |
| 3 | 415 | 474 |
| 4 | 412 | 470 |
| 5 | 419 | 477 |

| 測量次數 | 照度模組 2 | 照度計 |
|------|--------|-----|
| 1 | 439 | 483 |
| 2 | 439 | 484 |
| 3 | 434 | 479 |
| 4 | 431 | 475 |
| 5 | 432 | 478 |



結果：我們改用 Arduino 板連接(GY-30)，但因序列埠位址只有兩個，因此無法連接第三個。

討論：在無法連接第三個的照度模組上，我們做了討論及驗證實驗，因為一個 Arduino 板給予照度模組的位址只有兩個，因此討論後的解決方法為將 Arduino 板在寫入新的位址即可。

二、模擬等比例縮小教室的自動偵測和開關調控。

當下照度 > 最大照度值 = 關閉照明

當下照度 < 最小照度值 = 開啟照明

照度過高：

| 當下照度 | > | 最大照度值 | = | 關閉照明 |
|------|---|-------|---|------|
| 403 | > | 400 | = | 關閉照明 |
| 404 | > | 400 | = | 關閉照明 |
| 404 | > | 400 | = | 關閉照明 |

照度過低：

| 當下照度 | < | 最小照度值 | = | 開啟照明 |
|------|---|-------|---|------|
| 296 | < | 300 | = | 開啟照明 |
| 299 | < | 300 | = | 開啟照明 |
| 295 | < | 300 | = | 開啟照明 |

結果：我們使用 Arduino 連接兩塊照度模組(GY-30)，分別放在等比例縮小教室的側邊與中央，當亮度高於最大照度值，燈就會自動關閉，當亮度低於最小照度值，便會自動開啟，達到自動控制開關的功能。

討論：在實驗四中我們用等比例縮小的教室將照度模組放入作為測量照度的感應裝置，之後按照其讀出的數值過高或過低來自動判斷簡單的開燈和關燈，但目前尚未完成所需的依照感測到數值後始燈泡做漸漸地條亮或調暗，所以得在實驗 LED 燈泡的調控靈敏性及能夠達成的這項實驗的程式方法。

三、將自動偵測和開關調控系統運用於實體教室中。

結果：我們經過模擬之後，實際到教室實驗，發現開關可以透過 Arduino 控制，也可以自動偵測當下的照度並自行調整亮度。

討論：在實驗五中我們將教室燈接上照度模組後也便能夠像實驗三等比模擬的縮小教室一樣，照度過高便開燈，而照度過低則關燈，但其需額外接繼電器才能達到此效果，而在我們的目標使燈能夠漸亮和漸暗的方面則無法達成，因為學校目前所使用的都是日光燈，因此無法達到 PWM 漸亮或漸暗的效果。

四、將 Arduino PWM 連接 LED 燈管並進行研究及改良。

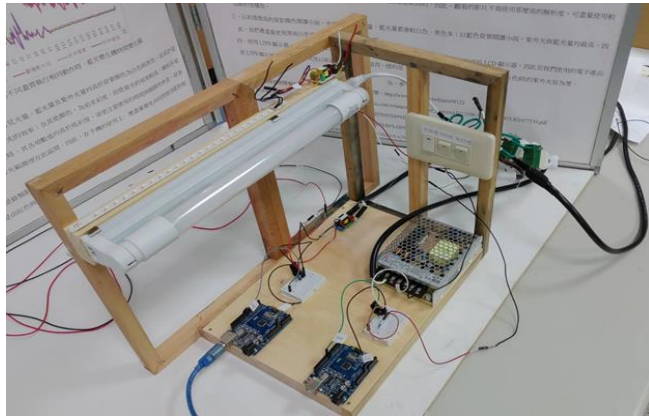
結果：我們使用 Arduino PWM 連接 LED 燈管中原本的變壓器後並接上燈管，但因反應不夠靈敏，因此我們連接繼電器使其反應更加快速並達到使電流能夠更穩定的效果。

| 開與關的時間比(微秒) | Lux 照度計 | 全亮/閃爍/全暗 |
|-------------|---------|----------|
| (100 , 0) | 557 | 全亮 |
| (90 , 10) | 553 | 全亮 |
| (80 , 20) | 543 | 全亮 |
| (70 , 30) | 532 | 全亮 |
| (60 , 40) | 519 | 全亮 |
| (50 , 50) | 504 | 全亮 |
| (40 , 60) | 487 | 全亮 |
| (30 , 70) | 461 | 全亮 |
| (20 , 80) | 426 | 全亮 |
| (10 , 90) | ~ | 閃爍 |
| (0 , 100) | 0 | 全暗 |

開與關的時間比(微秒)所對應的 Lux 及燈管的亮暗狀況。

| PWM | 照度 | 亮暗情況 |
|-----|-------------|------|
| 255 | 525 | 全亮 |
| 200 | 389 | 全亮 |
| 150 | 315 | 全亮 |
| 100 | 212 | 全亮 |
| 50 | 110 | 全亮 |
| 0 | 7 (室外餘光) | 全暗 |

開與關的 PWM 數值所對應的 Lux 及燈管的亮暗狀況。



實驗裝置成品圖

討論：在實驗六中我們發現一般的 LED 燈管也能夠達到我們所要的依照當下的照度將其漸暗和漸亮的調整教室中的照度能夠達到標準，而雖然其價格高於日光燈，但其在壽命、省電和環保等各方面都比日光燈來的更好

五、將完整的自動偵測調光系統運用於等比例縮小的教室中。

| AI 控光系統 | | 無控光系統 | |
|---------|---------|----------|---------|
| 內側 | 外側 | 內側 | 外側 |
| 27.1 mA | 43.2 mA | 57.7 mA | 54.8 mA |
| 70.3 mA | | 112.5 mA | |

先將 mA 換算成 A 後，在將其乘上每天平均所使用的時間，在得出兩數值後計算其省電百分比：

$$\left\{ \begin{array}{l} \text{AI 控光系統：} 70.3 \text{ mA} \div 1000 \times 60 \times 60 \times 9 = 2277.7 \text{ A} \\ \text{無控光系統：} 112.5 \text{ mA} \div 1000 \times 60 \times 60 \times 9 = 3645.0 \text{ A} \end{array} \right.$$

⇒ AI 控光系統對無控光系統的省電百分比為 37.5 %

結果：我們在模擬教室中放入兩塊照度模組，使其感應當下的照度，並設定一個定值，如果其值高於 200 Lux 時，會自動漸漸按照定值調暗，若其值低

於 300 Lux 時同樣則會自動調漸漸調亮，使其能夠達到我們要的智慧調光效果。

討論：我們發現用和 LED 燈管相似的方法也能將 LED 燈泡規模的教室達到自動調光的效果。所以，若能將其運用至 LED 燈管的實體教室，將能完成本實驗本研究的具體成果

陸、結論

實驗中發現，有 AI 控光系統的教室能夠省下約 37.5 %耗電量，且單獨從 LED 管對傳統燈管燈管的省電百分率來看即可節省約 50%的耗電量 (T8-4 尺)，可以省下非常龐大的電量，除了是節省能源也是很大的降低學校的支出，依這樣所設計的 Arduino 程式及相應模組來對應出依照亮度漸暗或漸亮來自動調控的效果，所以若教室的照明設備能符合則相信期會為一個成功的智慧裝置。

在未來的發展方面，這樣的智慧調光裝置能夠走入各個家庭及我們生活周遭的各種公共場所及環境，其可運用的地方非常多，也期望我們的設計能再進一步做到無線控制，例如透過 Wi-Fi 或藍芽就可以用手機或平板進行遠端控制甚至將其加上能手動的調整設定並裝設於牆壁上，使其更加容易因應不同的需求做適當的調整，相信它一定除了對眼睛的健康有益外，也同時能讓我們的生活更智慧化。

柒、參考文獻

1. Arduino。2018。Home (Arduino 相關器材)。Arduino 股份有限公司。

<https://www.arduino.cc/>

2. CNS 照度標準。2010。Microstone (各讀書或辦公場所照度)。

www.microstone.com.tw/CNS照度標準.html

3. 趙英傑。2015。超圖解 Arduino 互動設計入門。旗標出版股份有限公司。
4. 鄭志鵬。2017。Arduino 實驗儀器設計及程式撰寫練習。宇宙機器人企業有限公司。
5. Arduino UNO 學習 610。2016。照度計模塊 GY-30 。
http://ee543.blogspot.com/2016/12/arduino-uno610gy-30_2.html
6. Arduino。2018。PWM。Arduino 股份有限公司。
<https://www.arduino.cc/en/Tutorial/PWM>