

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



時間提醒藥盒

學生姓名：梁朝輝

陳建智

陳冠文

指導老師：蔡忠憲

中華民國 106 年 03 月

目錄

壹、摘要	1
貳、研究動機	1
一、動機	1
二、目的	2
參、研究方法	3
一、研究流程圖	3
二、研究設備與材料	4
三、系統架構	4
四、研究的材料	5
(一)杜邦線	5
(二)LCD	5
(三)I2C	7
(四)電阻	8
(五)按鈕	9
(六)無源蜂鳴器	9
(七)Arduino Uno 板	10
(八)LED	13
(九)USB 線	13
(十)塑膠板	14
(十一)藥盒	15
五、問題與討論	15
(一)時間判斷	15
(二)變數問題	15
(三)使用者在設定上的困難	15
六、組裝硬體架構	16
肆、研究結果	17
一、傳統藥盒的優缺點	17
(一)傳統藥盒的優點	17
(二)傳統藥盒的缺點	17
二、電子藥盒跟傳統藥盒的比較	17
三、製作過程中遇到的困難	18
(一)討論如何計算時間	18
伍、討論	18
一、創意性	18
二、實用性	18

三、未來展望·····	19
陸、結論·····	19
柒、過程與改造·····	19
一、原版的功能與缺陷·····	19
二、改造後的性能·····	20
三、改造前與改造後的藥盒按鈕功能·····	21
捌、心得·····	22
玖、附錄·····	23
壹拾、參考資料·····	29

圖目錄

圖 1	忘記吃藥的例子圖	2
圖 2	研究流程圖	3
圖 3	系統架構圖	4
圖 4	杜邦線	5
圖 5	LCD 液晶顯示器	6
圖 6	I2C 模組型號與預設位址	7
圖 7	I2C 模組	8
圖 8	電阻	8
圖 9	按鈕	9
圖 10	無源蜂鳴器	9
圖 11	有源蜂鳴器	10
圖 12	Arduino Uno 板	11
圖 13	LED 燈	13
圖 14	USB 傳輸線	14
圖 15	塑膠板	14
圖 16	藥盒	15
圖 17	作品電路圖	16
圖 18	塑膠板裁切完成	16
圖 19	時間提醒藥盒半成品	17
圖 20	原版電子藥盒	20
圖 21	新版電子藥盒	21
圖 22	時間提醒藥盒成品	22

表目錄

表 1	材料表	4
表 2	LCD 接腳介紹表	6
表 3	Arduino 硬體介紹	10
表 4	老人未服藥原因所在	19

壹、摘要

本研究與展望是對現今社會的狀況做一個改善，現在不只健忘的老人需要服用藥物，連工作繁忙的年輕人都有可能必須按時吃藥，不管是哪一個族群都可以利用【時間提醒藥盒】來達到提醒吃藥的作用，有些人也經常會忘記藥盒放置處，這時當時間到時就有可能要找老半天，那如果藥盒會發出聲音那就可以利用聲音來更快速的尋找到藥盒，而【時間提醒藥盒】會有五組提醒時間可以提供設定，當時間到時就會有蜂鳴器發出聲音來提醒，這時跟第幾組提醒時間相對應的LED也會跟著亮，可以提示使用者吃下對的藥物，使用者被提醒後也知道要吃哪一格的藥盒時，就可以按下停止提醒的按鈕讓聲音與LED停止，而有時使用者當下特別忙碌時，就可以按下延後提醒的功能，每按一下就會增加一定的時間，這時就會開始倒數延後的時間，時間一到就會再次響起蜂鳴器以及對應的LED燈。

貳、研究動機

一、動機

最近過年家族團圓時看到許多長輩都要隨身攜帶藥在身上，聊天聊到一半就會看一下時間確認是否時間到了，也會經常性的忘記服用藥物，發現了這些情況之後就有了些想法，於是上網查了些相關資料，發現平常很正常的按時吃藥，對一些患有特殊疾病的人來說卻是非常重要的，平常人生病時才會吃藥，但那些人卻是要每天固定要做的事，而現在的工作環境和時數都非常不理想，導致很多人患有嚴重疾病卻會因為工作繁忙而錯過吃藥時間，平常人錯過了只不過是病情晚一點好而已，但對那些特殊疾病的患者來說有可能就會有嚴重的影響。



37歲廖姓男子(左二)因忘了吃高血壓藥，再加上天氣炎熱，出現小腦中風，幸緊急救回一命，亞大醫院獻花恭賀他康復。(林欣儀攝)

圖 1 忘記吃藥的例子圖

二、目的

- (一) 給健忘的使用者有個安心的提醒裝置。
- (二) 工作場所無法攜帶手機的使用者不用刻意在意時間專心於工作。
- (三) 能夠代替手機 APP 給小孩或老人使用，不用怕損壞或遺失。
- (四) 利用聲音和 LED 指示使用者，讓使用者在對的時間吃下對的藥物。

參、研究方法

一、研究流程圖

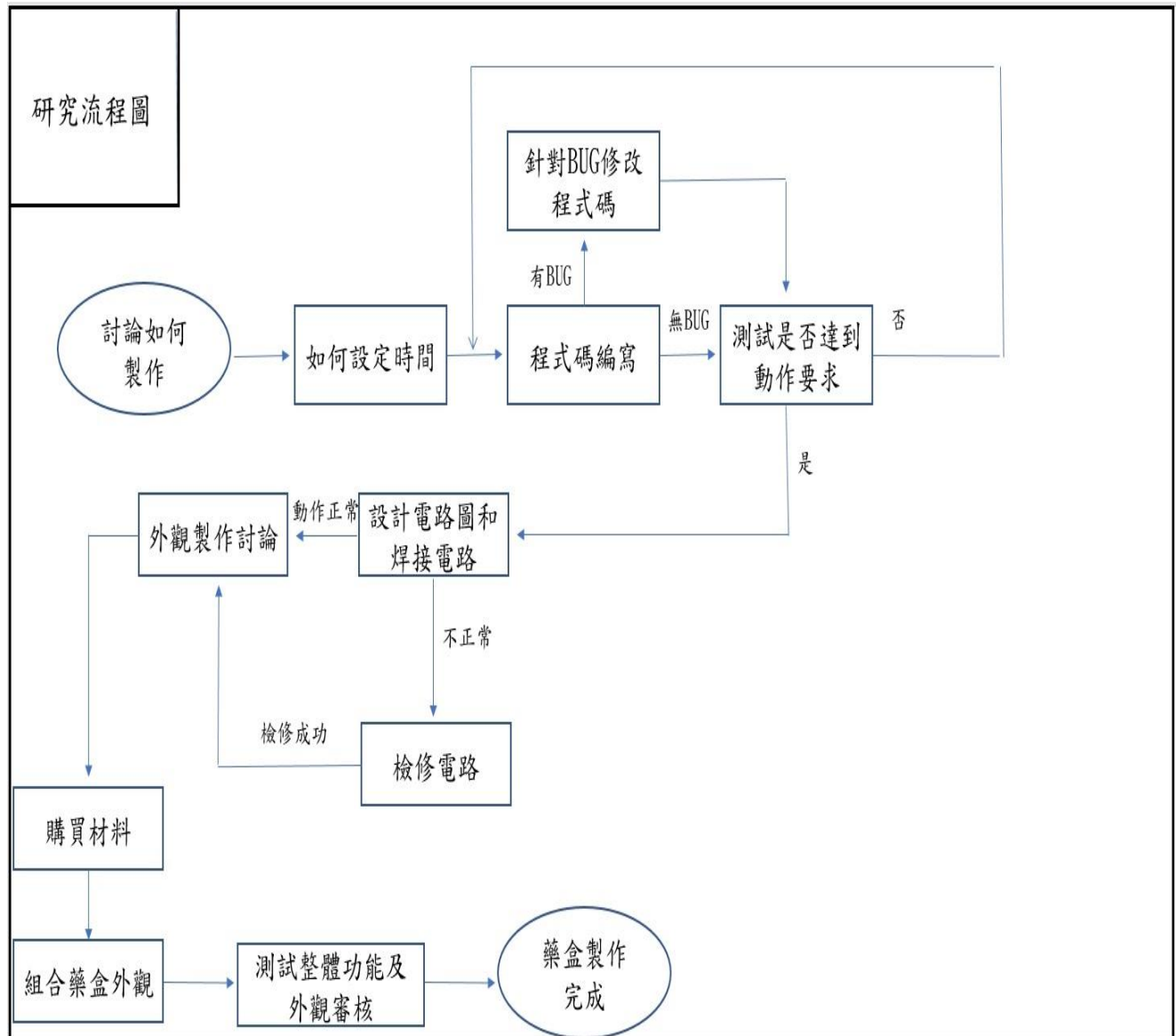


圖 2 研究流程圖

二、研究設備與材料

表 1 材料表

項次	名稱	規格	數量	備註
1	杜邦線		16 條	
2	LCD I2C		1 個	
3	電阻	10KΩ	5 個	
4	九百孔		1 個	
5	按鈕		5 個	
6	蜂鳴器	無源	1 個	
7	塑膠板		2 個	
8	Arduino 板	Uno	1 個	
9	藥盒		5 個	
10	LED		5 個	
11	USB線		1 條	

三、系統架構

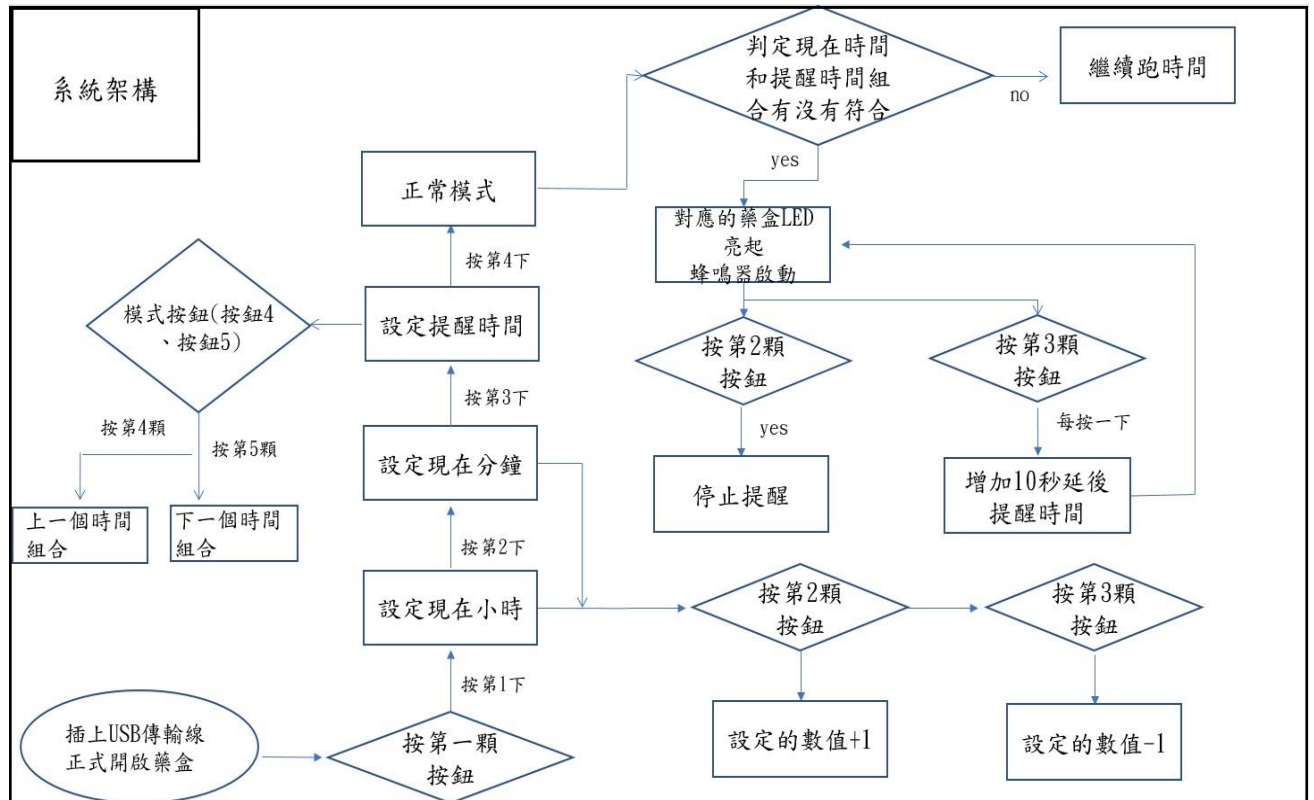


圖 3 系統架構圖

四、研究的材料

(一)杜邦線：

是美國杜邦公司生產的有特殊效用的縫紉線。

電子行業杜邦線可用於實驗板的引腳擴展，增加實驗項目等。可以非常牢靠地和插針連接，無需焊接，可以快速進行電路試驗。

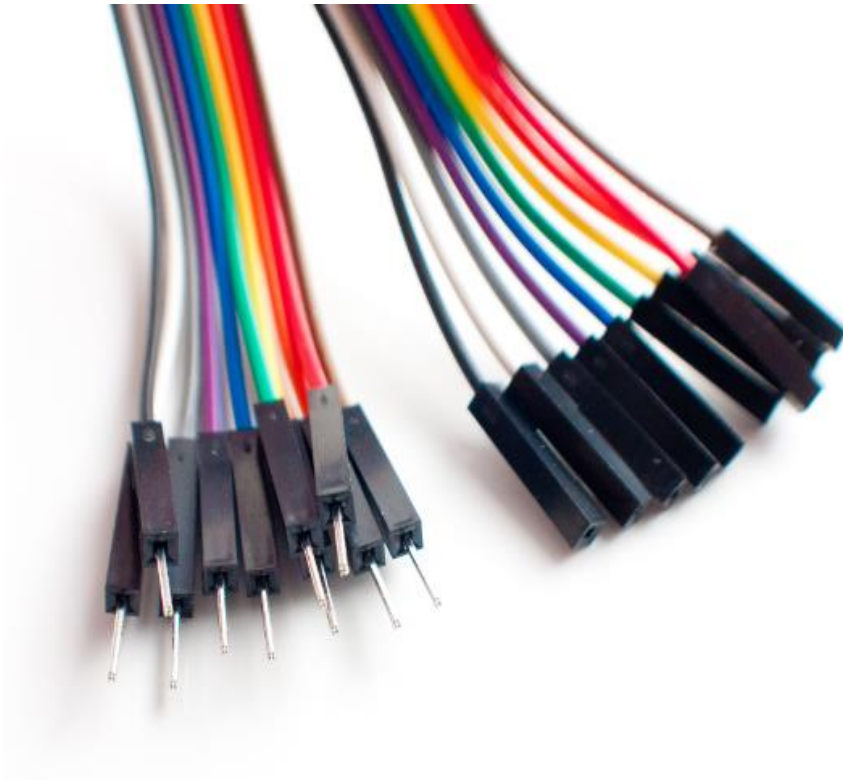


圖 4 杜邦線

(二)LCD：

液晶顯示器（英語：liquid-crystal display，縮寫：LCD）為平面薄型的顯示裝置，由一定數量的彩色或黑白畫素組成，放置於光源或者反射面前方。液晶顯示器功耗低，因此倍受工程師青睞，適用於使用電池的電子裝置。

LCD 總共有 14 支接腳，如果內建背光的話是 16 支，這些腳位的功能整理於下表：

表 2 LCD 接腳介紹表

腳位編號	名稱	說明
1	Vss	接地 (0V)
2	Vdd	電源 (+5V)
3	Vo 或稱 Vee	對比(0-5V), 可接一顆 1k 電阻, 或利可變電阻調整適當的對比
4	RS	Register Select: 1: D0 - D7 當作資料解釋 0: D0 - D7 當作指令解釋
5	R/W	Read/Write mode: 1: 從 LCD 讀取資料 0: 寫資料到 LCD 因為很少從 LCD 這端讀取資料, 可將此腳位接地以節省 I/O 腳位。
6	E	Enable
7	D0	Bit 0 LSB
8	D1	Bit 1
9	D2	Bit 2
10	D3	Bit 3
11	D4	Bit 4
12	D5	Bit 5
13	D6	Bit 6
14	D7	Bit 7 MSB
15	A+	背光(串接 330R 電阻到電源)
16	K-	背光(GND)



圖 5 LCD 液晶顯示器

(三) I2C:

I²C (Inter-Integrated Circuit) 字面上的意思是積體電路之間，它其實是 I²C Bus 簡稱，所以中文應該叫積體電路匯流排，它是一種串列通訊匯流排，使用內送流量備援容錯機制從架構，由飛利浦公司在 1980 年代為了讓主機板、嵌入式系統或手機用以連接低速週邊裝置而發展。I²C 的正確讀法為「I 平方 C」

("I-squared-C")，而「I 二 C」("I-two-C") 則是另一種錯誤但被廣泛使用的讀法。自 2006 年 11 月 1 日起，使用 I²C 協定已經不需要支付專利費，但製造商仍然需要付費以取得 I²C 從屬裝置位址。

標準 LCD 需要比較多的腳位，I2C LCD 可以減少腳位的使用，而且接線也比較簡單。這個 I2C LCD 的特性如下：

1. 16x2 LCD，藍底白字，有背光
2. 走 I2C 介面，設備地址為 0x27
3. 可調整螢幕顯示對比
4. 5V 供電，相容於 Arduino

其 I2C 的接線如下：

1. SDA - 接 Arduino 的 Analog Pin 4
2. SCL - 接 Arduino 的 Analog Pin 5
3. GND - 接 GND
4. VCC - 接 +5V



圖 6 I2C 模組型號與預設位址



圖 7 I2C 模組

(四) 電阻：

在電磁學裡，電阻是一個物體對於電流通過的阻礙能力，其中， R 為電阻， V 為物體兩端的電壓， I 為通過物體的電流。

假設這物體具有均勻截面面積，則其電阻與電阻率、長度成正比，與截面面積成反比。

採用國際單位制，電阻的單位為歐姆 (Ω , Ohm)。電阻的倒數為電導，單位為西門子 (S)。

假設溫度不變，則很多種物質會遵守歐姆定律，即這些物質所組成的物體，其電阻為常數，不跟電流或電壓有關。稱這些物質為「歐姆物質」；不遵守歐姆定律的物質為「非歐姆物質」。電路符號常常用 R 來表示，例： R_1 、 R_02 、 R_{100}



圖 8 電阻

(五) 按鈕:

按鈕也稱為按鍵，是一種電閘（switch，或稱開關），用來控制機械或程式的某些功能。一般而言，紅色按鈕是用來使某一功能停止，而綠色按鈕，則通常可開始某一項功能。按鈕的形狀通常是圓形或方形。



圖 9 按鈕

(六) 無源蜂鳴器:

蜂鳴器是一種一體化結構的電子訊響器，採用直流電壓供電，廣泛應用於計算機、印表機、複印機、報警器、電子玩具、汽車電子設備、電話機、定時器等電子產品中作發聲器件。蜂鳴器也是產生聲音的信號裝置，有機械型、機電型及壓電型。蜂鳴器的典型應用包括報警裝置，定時器。

1. 蜂鳴器的分類:

以下兩張圖雖然都是蜂鳴器，但是存在區別，第一張圖(圖 10)的蜂鳴器可以看到內部有電路板，而第二張圖(圖 11)是澆封起來的。蜂鳴器從驅動方式上來分可以分為無源蜂鳴器和有源蜂鳴器。第一張圖(圖 10)就是無源的，第二張圖(圖 11)就是有源的。

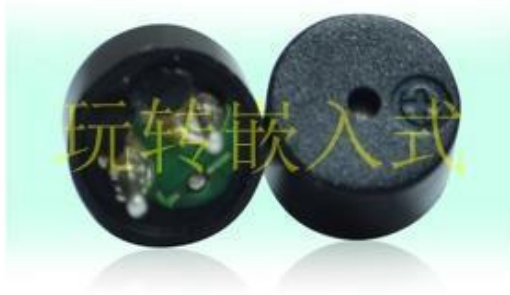


圖 10 無源蜂鳴器



圖 11 有源蜂鳴器

2. 蜂鳴器的驅動方式

無源蜂鳴器是沒有正負之分的，類似於喇叭，只要在兩個腿上加載不同的頻率的電信號就可以實現發聲，根據不同的頻率所發出的聲音也是不一樣的。

有源蜂鳴器是有正負之分的，只需要在兩個腿上加上電壓信號就會發聲，發出的聲音音調單一、頻率固定。

有源蜂鳴器比無源蜂鳴器內部多了振蕩結構，所以有源蜂鳴器在價格上稍微貴一點。

(七) Arduino Uno 板:

Arduino，是一個開放原始碼的單晶片微控制器，它使用了 Atmel AVR 單片機，採用了開放原始碼的軟硬體平台，建構於簡易輸出/輸入 (simple I/O) 介面，並且具有使用類似 Java、C 語言的 Processing/Wiring 開發環境。

表 3 Arduino 硬體介紹

微控制器	ATMEGA328
工作電壓	5V
輸入電壓 (推薦)	7-12V
輸入電壓 (限制)	6-20V
數字 I/O 接腳	14 (6個提供PWM輸出)
模擬輸入接腳	6支
EEPROM	1 KB (ATMEGA328)
震盪速度	16 MHz

1. Arduino 硬體功能說明:

(1) 數位 I/O 接腳:

14 支數位 I/O 接腳 可以當作 input 使用，也可以當作 output 使用，使用方法是透過 `pinMode()`, `digitalWrite()`, `digitalRead()` 這幾個函式。這 14 支數位 I/O 接腳，其中幾支腳有特殊的功能。

2. Arduino 硬體介紹:

我們用下面這張 Arduino Uno 硬體，跟大家介紹一下一般 Arduino 的硬體上面會有哪些東西。

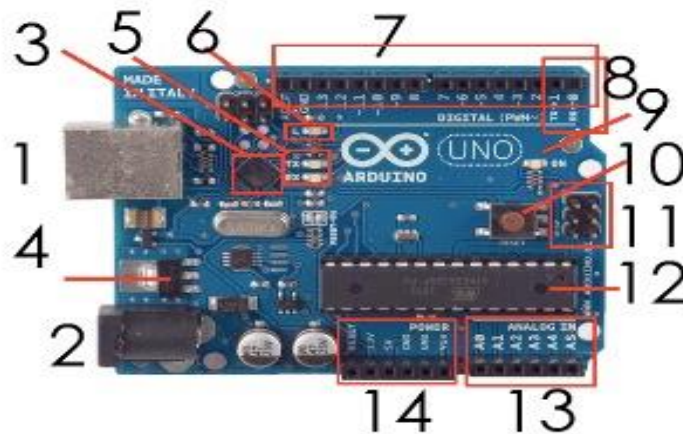


圖 12 Arduino Uno 硬體

- (1) 支援 USB 接頭傳輸資料及供電(不需額外電源)。
- (2) 輸入電壓 Power Jack，可以單獨使用，不需連接 USB 電腦時就可以直接接上 5V 變壓器，或是電池直接供電。
- (3) FTDI USB 晶片，這是 USB 的 Client 端的晶片，透過這個晶片就可以跟電腦溝通連結傳遞資料。
- (4) Voltage Regulator: 穩壓器，保持電壓的穩定。
- (5) LED 燈: 用來顯示傳遞接受的資料 RX 和傳遞出去的資料 TX，如有資料進去跟出去的話這 LED 燈都會閃爍一下，並且這兩個 LED 燈，和區域 7 的接腳 0 和接腳 1 是相連結的。
- (6) LED 燈: 用來顯示區域 7 的接腳 13，是高電壓還低電壓。

- (7) 數位接腳:由右到左那他分別是
數位接腳 0 到數位接腳 13: 你注意看一下印刷, 有些字前面有~符號, 例如說~10, 這個意思是當它可以當成數位接腳 10, 可以做 PWM 的資料輸出, 所以呢這個符號的意思是說在這個板子上面, 只有特別的接腳可以做 PWM 的資料輸出。
GND: 接地接腳。
AREF: AREF 是指模擬參考。他的英文全名是 Analogue REFerence, 它可以讓我們知道 Arduino 的參考電壓。例如, 如果我們要測量的最大電壓範圍 3.3V, 我們那我們就需要有一個很好的 3.3V, 是那我們就可以從 AREF 接腳取得。
- (8) RX<- 0 和 TX -> 1:意思是說如果要做 UART 資料傳遞的話, 想要把資料傳進去 arduino 的話, 可以透過 TX -> 1(接腳1), 那 arduino 如果要做做 UART 資料傳遞出去的話, 就可以把接腳 RX<- 0 (接腳 0) 接到另外一個硬體上面。今天要提醒一下這兩個接腳, 平常他時會跟 USB 連接線連結在一起了, 所以說如果你要把程式燒錄透過 USB 到這個版子上時, 在那個時刻請確定你的接腳 0 和接腳 1 上面是沒有任何接線的, 不然程式燒錄的動作就會失敗喔!
- (9) LED 燈: 電源顯示燈。
- (10) 重新執行程式按鈕。
- (11) ICSP Header: 英文的全名是 In-circuit serial programming 序列燒錄方式, PIC 燒錄器燒錄程式的方法, 程式記憶體為 Flash 的版本使用方便的 ICSP 序列燒錄方式。
- (12) Microcontroller: 控制晶片。
- (13) 類比訊號接腳: a0, a1, a2, a3, a4, a5
- (14) 電源控制的接腳:
- Vin (電源輸入)
 - Gnd (接地)
 - Gnd (接地)
 - 5V (5V 電源輸出)
 - 3V3 (3V 電源輸出)
 - RESET (重新執行程式)

(八) LED:

發光二極體（英語：Light-emitting diode，縮寫：LED）是一種能發光的半導體電子元件，透過三價與五價元素所組成的複合光源。此種電子元件早在 1962 年出現，早期只能夠發出低光度的紅光，被惠普買下專利後當作指示燈利用。及後發展出其他單色光的版本，時至今日，能夠發出的光已經遍及可見光、紅外線及紫外線，光度亦提高到相當高的程度。用途由初時的指示燈及顯示板等；隨著白光發光二極體的出現，近年逐漸發展至被普遍用作照明用途。

發光二極體只能夠往一個方向導通（通電），叫作順向偏壓，當電流流過時，電子與電洞在其內重合而發出單色光，這叫電致發光效應，而光線的波長、顏色跟其所採用的半導體物料種類與故意摻入的元素雜質有關。具有效率高、壽命長、不易破損、反應速度快、可靠性高等傳統光源不及的優點。白光 LED 的發光效率近年有所進步；每千流明成本，也因為大量的資金投入使價格下降，但成本仍遠高於其他的傳統照明。雖然如此，近年仍然越來越多被用在照明用途上。



圖 13 LED 燈

(九) USB 線:

通用序列匯流排（英語：Universal Serial Bus，縮寫：USB）是連線電腦系統與外部裝置的一種串列埠匯流排標準，也是一種輸入輸出介面的技術規範，被廣泛地應用於個人電腦和行動裝置等訊息通訊產品，並擴充功能至攝影器材、數位電視（機上盒）、遊戲機等其它相關領域。

多媒體電腦剛問世時，外接式裝置的傳輸介面各不相同，如印表機只能接 LPT port、數據機只能接 RS232、滑鼠鍵盤只能接 PS/2 等。繁雜的介面系統，加上必須安裝驅動程式並重新開機才能使用的限制，都會造成使用者的困擾。

因此，創造出一個統一且支援易插拔的外接式傳輸介面，便成為無可避免的趨勢。

最新一代是 USB 3.1，傳輸速度為 10Gbit/s，三段式電壓 5V/12V/20V，最大供電 100W，另外除了舊有的 Type-A、B 介面之外，新型 USB Type-C 接頭不再分正反。



圖 14 USB 傳輸線

(十) 塑膠板：

塑膠是指以高分子量的合成樹脂為主要組分，加入適當添加劑，如增塑劑、穩定劑、抗氧化劑、阻燃劑、潤滑劑、著色劑等，經加工成型的塑性（柔韌性）材料，或固化交聯形成的剛性材料。

塑膠最早來自於 1850 年代的英國。自從塑膠被開發以來，各方面的用途日益廣泛。



圖 15 塑膠板

(十一) 藥盒：



圖 16 藥盒

五、問題與討論

(一)Q1. 運算時間的程式會因為不斷的運行，導致按下按鈕時程式無法立即做判斷？

在按下按鈕的時候程式需要時間判斷，而時間運算程式也會跟著同時運行，導致按鈕按下時無法即時的做出反應，因此我們就多新增一個模式，時間運行程式要在此模式下才會正常運行，其餘的設定用模式就不會運行，這樣就可以避免雙方之間造成的衝突，按下按鈕無法即時有反應的狀況也就解決了。

(二)Q2. 提醒時間變數太多？

提醒時間共有五組，每一組又可分為小時與分鐘所以共有 10 個提醒變數，如果都只用變數來做設定的話程式會寫的非常長，編寫判斷時間是否到的程式也會比較麻煩，所以我們利用了陣列，在利用一個變數告訴程式當下要判斷陣列中哪一個提醒時間組合。

(三)Q3. 提醒時間變數過多導致使用者在設定上困難？

因為有 10 個提醒變數，如果只利用一個按鈕切換模式的話，那麼就要從第一組設定到最後一組，這樣使用者在每次設定都必須要按非常多下才可轉換到正常

模式下啟動藥盒的時間運算程式，於是我們就多兩個按鈕讓他們可以上下切換要設定的提醒時間，只要使用者設定好就可以按下第一顆按鈕切換模式，這樣就可以把兩樣功能分開，使用者也不用按那麼多下按鈕切換到正常模式。

六、組裝硬體架構



圖 17 作品電路圖



圖 18 塑膠板裁切完成



圖 19 時間提醒藥盒半成品

肆、研究結果

一、傳統藥盒的優缺點

(一)傳統藥盒的優點

1. 價格較便宜。
2. 藥物區格較多。
3. 因為沒有電子元件，所以不會發出任何吵雜的提醒聲。

(二)傳統藥盒的缺點：

1. 天數與數量太多了，容易造成吃錯藥的情形發生。
2. 要自己注意時間吃藥。
3. 因為沒有聲音提醒，所以容易忘記放置的位置。

二、電子藥盒跟傳統藥盒的比較

(一) 經過以上兩種藥盒的了解在做它們的比對，我們發現各有優缺點，而此專題的構成要素，就是希望能夠中和傳統藥盒和智慧型藥盒的優點。

(二) 傳統藥盒和電子藥盒的比對，我們發現他們都有優缺點，而且這些優點與缺點其實是可以互補的，就以舉例來說：傳統藥盒必須要時常的注意時服藥時間，而電子藥盒的功能卻可以省下這個麻煩，但相同的，傳統藥盒使用上會比電子藥盒來的安靜許多。所以為了達成這個目的，我們想盡辦法讓這兩種藥盒的功能都能夠平均一些。

三、製作中遇到的困難

(一) 討論如何計算時間

1. 一開始我們想說利用 delay 函式讓時間經過一秒變數就會加一，但是用想的就知道會有許多的 BUG 存在，而導致時間會跑不準，所以就決定上網找相關資料開始研究，我們在研究的過程中發現了 millis 這個函式，於是便開始尋找這個語法要如何應用在程式上，好讓我們的時間能每經過一秒就會將變數給加一。

2.

```
if (millis() % 1000 == 0) {}
```

 Millis 因

為計算時是利用毫秒跑的，所以如果用(%)求餘數是否等於 0 來判斷時間是否是經過一秒鐘，這樣的寫法會很容易造成程式來不急進行運算，所以我們想出了一個解決方法。

```
timer=millis()/100;  
if (timer % 10 == 0)
```

先把 millis 的數值除於 100 讓他的數值變小，當 millis 是 150 時則 timer 這個變數就是 1.5 就等同 0.15 秒以此類推，在利用 if (timer % 10 == 0)這一段來判斷他是否經過了一秒，在讓時間的變數進行累加的動作。

伍、討論

一、創意性:

結合市面上的藥盒來設計，體型雖然比較大，但是因為搭配了 Led 以及蜂鳴器等電子元件來配合製作，在加上有延後提醒的功能，讓正在忙的使用者可以稍後再吃藥，效果及功能都比原先的藥盒好。

二、實用性:

利用 LED 搭配蜂鳴器的使用，好讓使用者更有效的服用正確的藥物，不必花任何的時間來確認時間或是藥物的種類，像某一些不能使用 3C 產品的場合，會讓這個作品的功能更加凸顯。就以現在來舉例說明，現在少子化老人也越來越多，能夠照顧老人的人手根本

不夠用，我們就用(表 2)來說明，表示出智慧型藥盒的重要性與價值所在。

表 4 老人未服藥原因所在

原因	發生地點			總計
	在外	家中	花園	
外出一未攜帶藥物	4	0	0	4
沒有提醒吃藥的器具	2	1	1	4
沒有聽到提醒器的聲音	2	0	0	2
忘了	0	3	1	4
和人相處時	0	2	0	2
在緊急中遺漏了吃藥	0	1	0	1
覺得不舒服	0	1	0	1
看電視	0	1	0	1
睡著了	0	3	0	3
總計	8	15	2	25

三、未來展望：

未來的科技一定會出現很多新奇的事物，所以我們的作品功能也一定要有新的發展，不然一定會被時間的變化所淘汰，因此我們計劃增新藥盒的功能，比方說：先錄製一些錄音檔，在提醒時間到的時候就撥放這一些錄音檔，就像是真的有人在對你說話一樣，或是說可以一一記錄下吃藥的時間，這樣就可以更好的規畫用藥時間，也能讓醫生能夠清楚的掌握病患吃藥的情形，進行下一次的診斷。

陸、結論

很多人都有吃藥的習慣長年下來就有了提醒吃藥的手機 APP，APP 雖然方便但還是必須攜帶著一個藥盒收納，於是我們就把兩者功能結合在一起，讓單純只是收納藥物用的藥盒也有了提醒的功能，如果使用者當下過於忙碌時也可以利用延後提醒的功能，這樣就可以讓使用者忙完事情之後再來服用藥物，另外還有 LED 提示使用者吃下對應的藥物。

柒、過程與改造

一、原版的功能與缺陷

當初設計藥盒時，純粹是對一般的電子型藥盒設計進行改良，所以有些地方無法完全滿足使用者的需求，我們最初設計時的功能只考慮到，怎麼能夠在準確的時間點提醒眾人服藥，而且原版的功能能夠提醒的時間只有一組，不像我們日後所進行的改造能夠有五組的提醒時間。

對於原版的功能設定，雖然是電子型藥盒的改造，不過就以性能來解釋，它會比市面上的藥盒功能來的更有效率，因為有方便的使用者自訂功能，所以也變的更加客製化。



圖 20 原版電子藥盒

二、改造後的性能

因為是針對原型藥盒的設計，所以一定會比之前的功能還要齊全，相同的程式碼一樣會跟著多起來，但這些不是重點，我們要把原版電子藥盒的功能全面升級，而且我們要的功能一定要比之前的陽春版好。

在此說明新版藥盒的基本功能，其基本功能雖然很像原版的電子藥盒，但是我們並非沒有改變其基本的運作原理，因為它的運作水平沒有達到我們要的水準，我們把原先的提醒時間組合提升到五組，原因是我們發現一般的使用者服藥的時間點，一天不只有一次，這樣一來使用者就可以自訂他想服藥的時間，再來我們也把提醒的功能強化了一番，原先的提醒時間結束時只有蜂鳴器在運作，後來我們想說再加強一下提醒的功能，在提醒時間結束時加入 LED 的特效，這樣就成功加強了他的提醒功能。

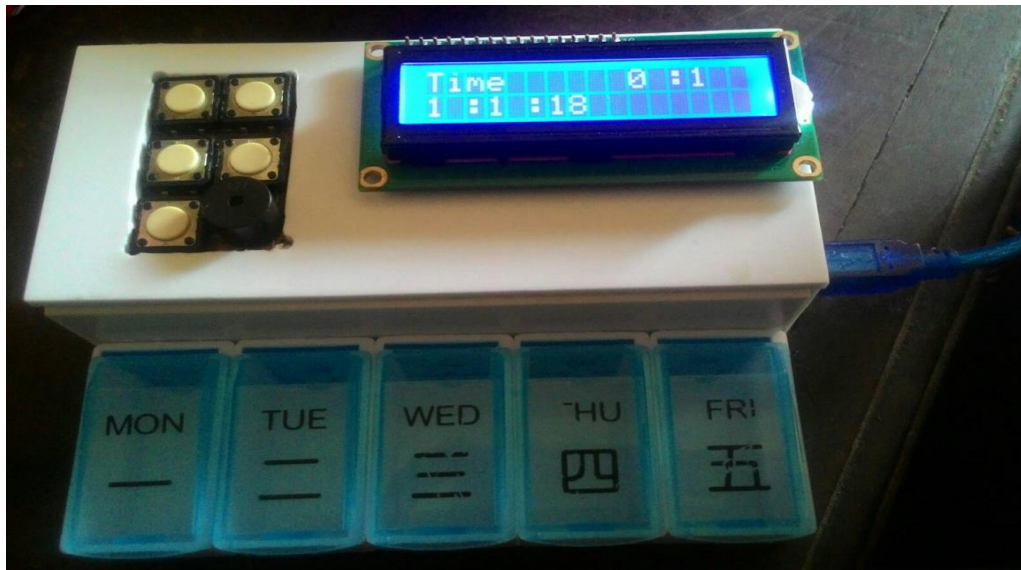


圖 21 新版電子藥盒

三、改造前與改造後的藥盒按鈕功能

(一) 改造前的藥盒總共有 3 顆按鈕，有 5 種的功能，功能分別為：

1. 切換模式
2. 加設定時間的值與暫停提醒
3. 減設定時間的值與延後提醒

(二) 改造後的藥盒總共有 5 顆按鈕，有 7 種的功能，功能分別為：

1. 切換模式
2. 加設定時間的值與暫停提醒
3. 減設定時間的值與延後提醒
4. 上一個時間組合切換
5. 下一個時間組合切換

(三) 由以上兩點可以看到改造前與改造後的功能多加了兩種，雖然只增加兩種，但是卻增加了一項很有用的功能，因為我們增加的功能可讓使用者自行設定幾點幾分該吃哪個藥盒，當時間到了藥盒上面的 LED 便會亮起對應的藥盒，清楚讓使用者知道該吃哪個藥盒，避免重複吃藥的情形發生。

捌、心得

就在一開始得知要做專題，專題是一種必須從無到有的條件下完成的作品，當下我們都覺得很困難，雖然如此，我們仍然必須面對，因為我們知道這一項任務完成後，一定會給我們帶來很大的成長，所以我們就開始分工像拼拼圖一樣一個一個接上去，然後做個測試，一開始的測試難免會有很多的BUG，就像是剛煉出來的劍總是需要石頭磨利，才有它的樣子，程式碼也一樣，剛完成在接上硬體的時候總是會有一些失算的地方，不過我們都有立即修正。

在做完這個提醒時間藥盒的過程中，我們處理了許多的BUG，也查了許多相關的資訊，雖然這是我們第一次做有關Arduino的專題，也雖然在製作的過程中遇到了許多的困難，但是在這當中學到了許多東西，像是Arduino裡面的程式，因為剛開始對Arduino的程式完全不了解也不會應用，而且在寫程式的過程中也會遇到了許多的BUG，所以必須要一直不斷的修改到我們要的功能為止，然後我們就在這當中遇到了一些平常不會遇到的BUG，雖然在做的過程中真的遇到很多的錯誤，但是當成功完成作品的時候，會整個心情雀躍也會有滿滿的成就感。

所以我們在做完這次的專題製作後我們不僅學到了平常不會遇到的程式邏輯，心中也會充滿了成就感，而且還成功完成人生中第一次的專題製作。



圖 22 時間提醒藥盒成品

玖、附錄

以下是作品的程式碼

```
1 #include <Wire.h> // Arduino IDE 內建
2
3 #include <LiquidCrystal_I2C.h>
4
5 LiquidCrystal_I2C lcd(0x3F, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE); // 設定 LCD I2C 位址
6
7 int buttonPin1 = 2;int buttonPin2 = 3;int buttonPin3 = 4;int buttonPin4 = 5;int buttonPin5 = 6;
8 int buttonStatus1;int buttonStatus2;int buttonStatus3;int buttonStatus4;int buttonStatus5;
9 int mode;//模式
10 int NowHour;int NowMinute;int Nowseconds;//現在時間
11 int NoticeTime[11]={0,0,0,0,0,0,0,0,0,0,0};//提醒時間陣列
12 int SetMode;int Notice;int Noticecombination;//提醒時間判斷
13 int Toremind=1;//提醒組合判斷
14 int buzzer;//蜂鳴器
15 int Delayed;int DelayedTime;//延後提醒
16 int Startreminder;//啟動提醒
17 int remindTemporary;//提醒暫存
18 unsigned long timer;
19 void setup() {
20     pinMode(buttonPin1,INPUT);
21     pinMode(buttonPin2,INPUT);
22     pinMode(buttonPin3,INPUT);
23     pinMode(buttonPin4,INPUT);
24     pinMode(buttonPin5,INPUT);
25     pinMode(8,OUTPUT);
26     pinMode(9,OUTPUT);
27     pinMode(10,OUTPUT);
28     pinMode(11,OUTPUT);
29     pinMode(12,OUTPUT);
30     pinMode(13,OUTPUT);
31     Serial.begin(115200); // 用於手動輸入文字
32     lcd.begin(16, 2); // 初始化 LCD，一行 16 的字元，共 2 行，預設開啟背光
33
34     // 閃爍三次
35     for(int i = 0; i < 3; i++) {
36         lcd.backlight(); // 開啟背光
37         delay(250);
38         lcd.noBacklight(); // 關閉背光
39         delay(250);
40     }
41     lcd.backlight();
42
43     // 輸出初始化文字
```

```

44   lcd.setCursor(0, 0); // 設定游標位置在第一行行首
45   lcd.print("Hello, user!");
46
47 }
48
49 void loop() {
50   buttonStatus1 = digitalRead(buttonPin1);
51   buttonStatus2 = digitalRead(buttonPin2);
52   buttonStatus3 = digitalRead(buttonPin3);
53   buttonStatus4 = digitalRead(buttonPin4);
54   buttonStatus5 = digitalRead(buttonPin5);
55   if (buttonStatus1 == LOW) //當按鈕1被按下時,按鈕1是切換模式的功能
56   {
57     digitalWrite(9,LOW); //只要按鈕1按下就會關掉所有LED燈
58     digitalWrite(10,LOW);
59     digitalWrite(11,LOW);
60     digitalWrite(12,LOW);
61     digitalWrite(13,LOW);
62     buzzer=0; //蜂鳴器關閉
63     Nowseconds=0; //現在秒鐘歸零,因為不歸零設定新的時間會造成問題
64     SetMode = 1; //設定提醒時間陣列用到的索引值=1
65     mode++; //模式變數加1
66     delay(200);
67     if (mode == 5){mode=1;} //模式有4種,當模式切換成5時回到模式1
68   }
69   if (mode == 1) //模式1
70   {
71
72     if (buttonStatus2 == LOW && NowHour < 24){ NowHour++;} //按下按鈕2且現在小時小於24
73     if (buttonStatus3 == LOW && NowHour > 0){ NowHour--;} //按下按鈕2且現在分鐘大於0
74
75     lcd.clear(); //在LCD上面顯示設定的現在時間
76     lcd.setCursor(0,0);
77     lcd.print("Set Now Hour");
78     lcd.setCursor(0,1);
79     lcd.print(NowHour);
80     lcd.setCursor(2,1);
81     lcd.print(":");
82     lcd.setCursor(3,1);
83     lcd.print(NowMinute);
84
85     delay(200);
86   }
87
88   if (mode == 2) //模式2
89   {
90     if (buttonStatus2 == LOW && NowMinute < 60){ NowMinute++;} //按下按鈕2且現在小時小於60
91     if (buttonStatus3 == LOW && NowMinute > 0){ NowMinute--;} //按下按鈕3且現在小時大於0

```

```

92     lcd.clear();//顯示在LCD上
93     lcd.setCursor(0,0);
94     lcd.print("Set Now Minute");
95     lcd.setCursor(0,1);
96     lcd.print(NowHour);
97     lcd.setCursor(2,1);
98     lcd.print(":");
99     lcd.setCursor(3,1);
100    lcd.print(NowMinute);
101
102        delay(200);
103    }
104
105    if (mode == 3)//模式3時
106    {
107        if (buttonStatus4 == LOW && SetMode < 10)//SetMode為設定提醒時間陣列的索引值
108        {
109            ++SetMode;
110        }
111        if (buttonStatus5 == LOW && SetMode > 0)
112        {
113            --SetMode;
114        }
115
116    if (buttonStatus2 == LOW){//按下按鈕2時加第SetMode個提醒時間陣列
117        ++NoticeTime[SetMode];
118
119
120    }
121    if (buttonStatus3 == LOW){//按下按鈕3時減第SetMode個提醒時間陣列
122        --NoticeTime[SetMode];
123
124    }
125    lcd.clear();//清除LCD
126    switch(SetMode)//判斷設定提醒組合時LCD上該如何顯示
127    {
128        case 0: Notice=1;Noticecombination=1;lcd.setCursor(0,0);lcd.print("NoticeHour"); lcd.print( Notice); break;
129        case 1: Notice=1;Noticecombination=1;lcd.setCursor(0,0);lcd.print("NoticeHour"); lcd.print( Notice); break;
130        case 2: Notice=1;Noticecombination=1;lcd.setCursor(0,0);lcd.print("NoticeMinute"); lcd.print( Notice); break;
131        case 3: Notice=2;Noticecombination=3;lcd.setCursor(0,0);lcd.print("NoticeHour"); lcd.print( Notice); break;
132        case 4: Notice=2;Noticecombination=3;lcd.setCursor(0,0);lcd.print("NoticeMinute"); lcd.print( Notice); break;
133        case 5: Notice=3;Noticecombination=5;lcd.setCursor(0,0);lcd.print("NoticeHour"); lcd.print( Notice); break;
134        case 6: Notice=3;Noticecombination=5;lcd.setCursor(0,0);lcd.print("NoticeMinute"); lcd.print( Notice); break;
135        case 7: Notice=4; Noticecombination=7;lcd.setCursor(0,0);lcd.print("NoticeHour"); lcd.print( Notice); break;
136        case 8: Notice=4; Noticecombination=7;lcd.setCursor(0,0);lcd.print("NoticeMinute"); lcd.print( Notice);break;
137        case 9: Notice=5; Noticecombination=9;lcd.setCursor(0,0);lcd.print("NoticeHour"); lcd.print( Notice); break;
138        case 10: Notice=5; Noticecombination=9;lcd.setCursor(0,0);lcd.print("NoticeMinute"); lcd.print( Notice);break;
139    }

```

```

140
141
142
143 lcd.setCursor(0,1);//判斷玩設定提醒時間當下要如何顯示後顯示
144 lcd.print(NoticeTime[Noticecombination]);
145 lcd.setCursor(2,1);
146 lcd.print(":");
147 lcd.print(NoticeTime[Noticecombination +1 ]);
148 delay(200);
149 }
150
151
152 if (mode == 4)//模式4
153 {
154     timer=millis()/100;//millis是記錄Arduino在開機後經過的毫秒數，因為毫秒運作太快所以先將他除於100縮小數值
155     //例:當millis()跑到150時=1.5=0.15秒以此類推
156     if (timer % 10 == 0)//當timer這個變數能被10整除代表經過一秒
157     {
158         lcd.clear();
159         lcd.setCursor(0,0);
160         lcd.print("Time");
161
162         //時間運算
163         ++Nowseconds;//加現在秒
164
165         if (Nowseconds==60){Nowseconds=0;++NowMinute;if(Delayed==0){buzzer=0;};if(Toremind>9 ){Toremind=1;}}
166         //經過一分鐘後執行以下動作
167
168         if(NowMinute==60){NowMinute=0;++NowHour;}//經過1小時候
169         if(NowHour==24){NowHour=0;}//經過24小時候
170         //判斷提醒時間組合是否有設定
171         if (NoticeTime[Toremind]==0 && NoticeTime[Toremind+1]==0 && Toremind<8 )
172         //判斷提醒時間陣列各組合是否有設定，分有10個陣列每兩個唯一組(小時:分鐘)
173
174         { Toremind+=2;}//如果當下的陣列無設定索引值+2跳到下一組提醒組合
175         //時間判斷
176         else
177         {if (NowHour==NoticeTime[Toremind] && NowMinute==NoticeTime[Toremind+1]&&Toremind<10 )
178         //現在時間與提醒時間吻合時
179
180         {
181             if (buttonStatus3 == LOW)//延後提醒啟動
182             {
183
184                 digitalWrite(9,LOW);
185                 digitalWrite(10,LOW);
186                 digitalWrite(11,LOW);
187                 digitalWrite(12,LOW);

```

```

188     digitalWrite(13,LOW);
189     remindTemporary=Toremind;//陣列索引值暫存
190     buzzer=0;//關閉蜂鳴器
191     Delayed=1;//延後模式啟動
192     DelayedTime+=10;//延後時間+10
193     }
194     if (Startreminder==0)//Startreminder是判斷是否是第一次觸發，不然會根停止提醒功能互相衝突
195     {
196         buzzer=1;
197         Startreminder=1;
198     }
199     if (Nowseconds==59)//經過一分鐘後跳到下一個提醒組合
200     {Startreminder=0; Toremind+=2;}
201     if(buzzer==1){ //如果蜂鳴器有啟動就判斷是第幾個提醒組合併亮起對應的LED
202         switch(Toremind)
203         {
204             case 1: digitalWrite(9,HIGH); break;
205             case 3: digitalWrite(10,HIGH); break;
206             case 5: digitalWrite(11,HIGH); break;
207             case 7: digitalWrite(12,HIGH); break;
208             case 9: digitalWrite(13,HIGH); break;
209         }
210     }
211     }
212     }
213
214     //延後提醒啟動
215     if (Delayed==1)
216     {
217         --DelayedTime;
218         if (DelayedTime==0){
219             buzzer=1;Delayed=0;
220             //延後提醒啟動後的LED顯示
221             switch(remindTemporary)
222             {
223                 case 1: digitalWrite(9,HIGH); break;
224                 case 3: digitalWrite(10,HIGH); break;
225                 case 5: digitalWrite(11,HIGH); break;
226                 case 7: digitalWrite(12,HIGH); break;
227                 case 9: digitalWrite(13,HIGH); break;
228             }
229         }
230     }
231     //暫停提醒
232     if (buttonStatus2 == LOW){
233         buzzer=0;
234         digitalWrite(9,LOW);
235         digitalWrite(10,LOW);

```



```

236     digitalWrite(11,LOW);
237     digitalWrite(12,LOW);
238     digitalWrite(13,LOW);
239     Delayed=0;}
240 //延後提醒啟動和不起動時的右上角顯示
241 if(Delayed==0)
242 {
243     lcd.setCursor(10,0);
244     lcd.print(NoticeTime[Toremind]);
245     lcd.setCursor(12,0);
246     lcd.print(":");
247     lcd.setCursor(13,0);
248     lcd.print(NoticeTime[Toremind+1]);
249 }
250     if(Delayed==1)
251 {
252     lcd.setCursor(10,0);
253     lcd.print(DelayedTime);
254 }
255     lcd.setCursor(0,1);
256     lcd.print(NowHour);
257     lcd.setCursor(2,1);
258     lcd.print(":");
259     lcd.setCursor(3,1);

260     lcd.print(NowMinute);
261 //蜂鳴器啟動
262     if (buzzer==1)
263     {digitalWrite(8,HIGH);delay(50);digitalWrite(8,LOW);delay(50);digitalWrite(8,HIGH);delay(50);digitalWrite(8,LOW);}
264     delay(100);
265 }
266 }
267 }
268

```

壹拾、參考資料

- [1] 天熱忘記吃藥小腦中風
<http://www.chinatimes.com/realtimenews/20161101004165-260405>
- [2] 長輩常常忘記自己吃過藥了沒
<https://www.toplhealth.com/Article/50953>
- [3] 高血壓忘記吃藥
<https://udn.com/news/story/9/2271475>
- [4] 重複使用藥物對身體很傷
<https://read01.com/Ny jxQ.html>
- [5] 高齡者重複用藥
<https://tw.news.yahoo.com/%E9%AB%98%E9%BD%A1%E8%80%85%E9%87%8D%E8%A4%87%E7%94%A8%E8%97%A5-%E5%8C%97%E5%B8%82%E9%A6%96%E5%89%B5%E9%AB%98%E9%BD%A1%E5%8F%8B%E5%96%84%E8%97%A5%E5%B1%80-034751188.html>
- [6] 天冷心肌梗塞 未規律服藥
<http://www.epochtimes.com/gb/17/2/21/n8832106.htm>
- [7] 老人的用藥行為
<http://www.kmuh.org.tw/www/kmcj/data/9108/6.htm>
- [8] 銀髮族生活 6 大危機(服藥危機)
<https://www.toplhealth.com/Article/110/26543>
- [9] 有源蜂鳴器和無源蜂鳴器的區別
<https://read01.com/003d7A.html>
- [10] Arduino 硬體介紹
<http://www.powenko.com/wordpress/?p=4167>