

國立虎尾科技大學電機工程系
專題製作報告

指導教授：張凱雄 教授

RFID慢跑數據紀錄系統

班 級：四電四甲

參與成員：石峰嘉

陳科澂

余儼庭

張 群

中華民國 一零二年 十月 二十二 日

摘 要

隨著國人健康意識的高漲，加上慢跑的興盛，常在學校操場上常會看見有許多人進行慢跑運動，因而想要設計一套 RFID 慢跑數據紀錄系統並架設於操場，提供使用者紀錄相關慢跑數據並可依據數據自行規劃日後的運動量，避免過量運動造成運動傷害。

本系統架構主要由 Reader、Tag 及 PC 應用端三部分構成，運作時透過跑者身上嵌入 Tag 之手環與 Reader 之間進行 RF 無線傳輸，將 Reader 所接收之資料送至 PC 端資料庫儲存，並透過 PC 端的液晶顯示器來顯示跑者慢跑時的相關數據，此為 RFID 慢跑數據紀錄系統主要功能。

目 錄

摘 要.....	2
目 錄.....	3
圖目錄.....	4
表目錄.....	6
第一章 緒論.....	7
1.1 研究動機.....	7
1.2 研究方法.....	5
第二章 晶片介紹.....	9
2.1 CC430F6137 介紹.....	9
2.2 MAX3223CPWR 介紹.....	10
第三章 系統架構與軟韌硬體設計.....	11
3.1 系統架構.....	11
3.2 流程圖.....	12
3.3 軟韌硬體設計.....	18
3.3.1 硬體.....	19
3.3.2 韌體.....	22
3.3.3 軟體.....	24
第四章 實驗完成結果.....	33
4.1 成品圖.....	33
4.2 操作流程.....	35
第五章 結論與未來展望.....	42
5.1 結論.....	42
5.2 未來展望.....	42
參考文獻.....	43
附錄.....	44

附錄 A.....	44
附錄 B.....	45
附錄 C.....	46
附錄 D.....	48
附錄 E.....	48

圖目錄

圖 2-1:CC430F6137 晶片.....	9
圖 2-2:CC430F6137 內部方塊圖.....	9
圖 2-3:MAX3223CPWR.....	10
圖 2-4:MAX3223CPWR 內部電路圖.....	10
圖 3-1:系統架構圖.....	11
圖 3-2:系統流程圖.....	12
圖3-3:LabVIEW介面流程圖-1.....	13
圖 3-4:LabVIEW 流程圖-2(馬拉松模式).....	14
圖3-5:LabVIEW流程圖-3(馬拉松模式).....	15
圖3-6:LabVIEW流程圖-4(慢跑模式).....	16
圖3-7: LabVIEW流程圖-5(慢跑模式).....	17
圖 3-8: eZ430-Chronos Development Tool.....	19
圖 3-9:READER 之電源電路圖.....	19
圖 3-10:MAX3223CPWR 與 RS-232 之電路圖.....	20
圖 3-11:CC430F6137 晶片與 RF 功能之電路圖.....	21
圖3-12: RSSI 訊號強度與距離示意圖.....	23
圖 3-13:LabVIEW Panel 介面.....	25
圖 3-14:LabVIEW Diagram 介面.....	26

圖 3-15: labview 與資料庫(MySQL)溝通模組.....	27
圖 3-16:使用 SQL 與法之正版.....	27
圖 3-17:使用 SQL 與法之背版.....	28
圖 3-18:進階運用讀取與修改語法之正版.....	28
圖 3-19:進階運用讀取與修改語法之背版.....	29
圖3-20:RS-232模組正版.....	29
圖3-21:RS-232模組背版.....	30
圖 3-22:抓取電腦時間，並做應用.....	30
圖3-23:抓取資料與匯出EXCEL之模組.....	31
圖3-24:換頁模組.....	32
圖3-25:特殊與法排名模組.....	32
圖4-1:READER完成圖.....	33
圖4-2:軟體使用者界面(馬拉松模式).....	33
圖4-3: 軟體管理者界面(馬拉松模式).....	34
圖4-4:軟體使用者介面(慢跑模式).....	34
圖4-5:軟體管理者介面(慢跑模式).....	35
圖4-6:跑者資訊輸入介面-1.....	35
圖4-7:跑者資訊輸入介面-2.....	36
圖4-8:慢跑模式使用者介面.....	36
圖4-9:當TAG接近READER1之情況.....	37
圖4-10:執行初始登錄之動作.....	37
圖4-11:跑者經過READER2之情形.....	38
圖4-12:資料刷新後.....	38
圖4-13:輸入密碼之畫面.....	39
圖4-14:跑者資訊輸入介面-1.....	40
圖 4-15:跑者資訊輸入介面-2.....	40

圖4-16:馬拉松模式使用者介面.....	41
圖4-17:匯出EXCEL檔.....	41
圖 A:RxBuffer 值之觀察室窗.....	45
圖 B:RSSI 之值.....	48

表目錄

表3-1:RSSI公式轉換結果.....	23
----------------------	----

第一章 緒論

1.1 研究動機

隨著國人健康意識的高漲，在學校操場上常會看見有許多人進行慢跑運動，慢跑是個簡單且方便的運動，無須特別的裝備，有雙鞋子以及合適場地即可進行。學校就剛好提供了操場這個適合跑步的場地，但若跑者想知道自己的紀錄必須自行攜帶計錄儀計，對跑者來說似乎不方便，所以我們打算設計 RFID 慢跑數據紀錄系統並架設於操場，使用者只需配戴具有 RF 功能的手環，就可以記錄當日的慢跑的圈數、時間及其他資料，並可依自我能力自行規劃日後的運動量，避免過量運動造成運動傷害，除此之外，學校每年都會在操場舉辦馬拉松比賽，而我們在 RFID 慢跑數據紀錄系統中加上馬拉松模式，供比賽時使用。

1.2 研究方法

本系統主要由 Reader、Tag 及 PC 應用端三部分構成，我們使用德州儀器(Texas Instruments, TI)的 CC430F6137 處理器晶片來實現 Reader 及 Tag，由於它具有低功耗模式以及 RF 無線傳輸功能，很適合用於本系統 Reader 及 Tag 的設計，而 PC 應用端採用美商國家儀器(National Instruments, NI)的 LabVIEW 介面設計軟體來設計使用者及管理員介面，搭配 MySQL 資料庫實現跑者資料及數據存取，並使用 RS232 介面與 Reader 進行資料及數據傳輸，系統運作時透過跑者身上嵌入 Tag 之手環與 Reader 之間進行 RF 無線傳輸，將 Reader 所接收之資料送至 PC 端資料庫儲存，並透過 PC 端的液晶顯示器來顯示跑者慢跑時的相關數據，此為 RFID 慢跑數據紀錄系統主要功能。

第二章 使用晶片

2.1 CC430F6137 介紹

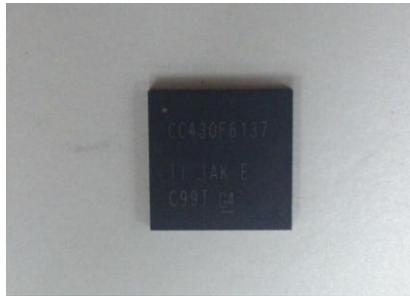


圖 2-1:CC430F6137 晶片

表 2-1:CC430F6137 之主要功能及特性

供電範圍:1.8V~3.6V	兩個通用串行通信接口 z
具低功耗模式： 待機模式：2.0 μ A (LPM3 RTC 模式) 關閉模式 (RAM 保持 LPM4)：1.0 μ A	頻寬：300 MHz to 348 MHz, 389 MHz to 464 MHz 779z to 928 MHz

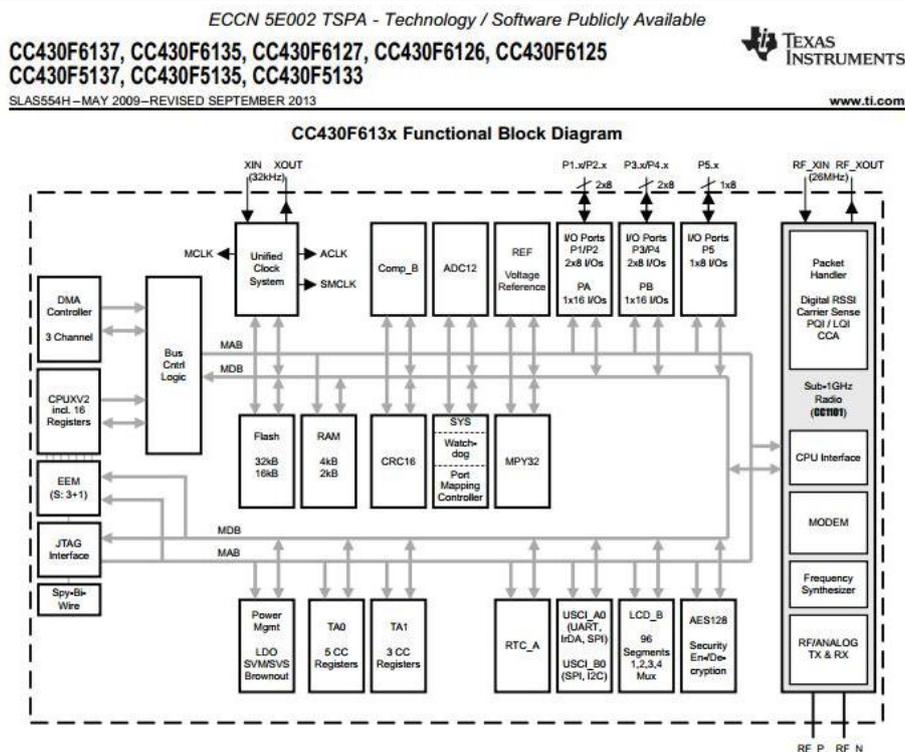


圖 2-2:CC430F6137 內部方塊圖

2.2 MAX3223CPWR 介紹：

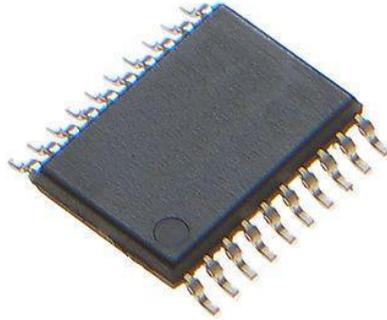


圖 2-3:MAX3223CPWR

電源電流：+/- 60 毫安	電源電壓（最大值）：5.5 V
電源電壓（最小值）：3 V	最大工作溫度：+ 70 C
最小工作溫度：0°C	接口：EIA/TIA-232-F 或 RS-232 或 V. 28
數據速率 0.25 Mbps 的（最小）	

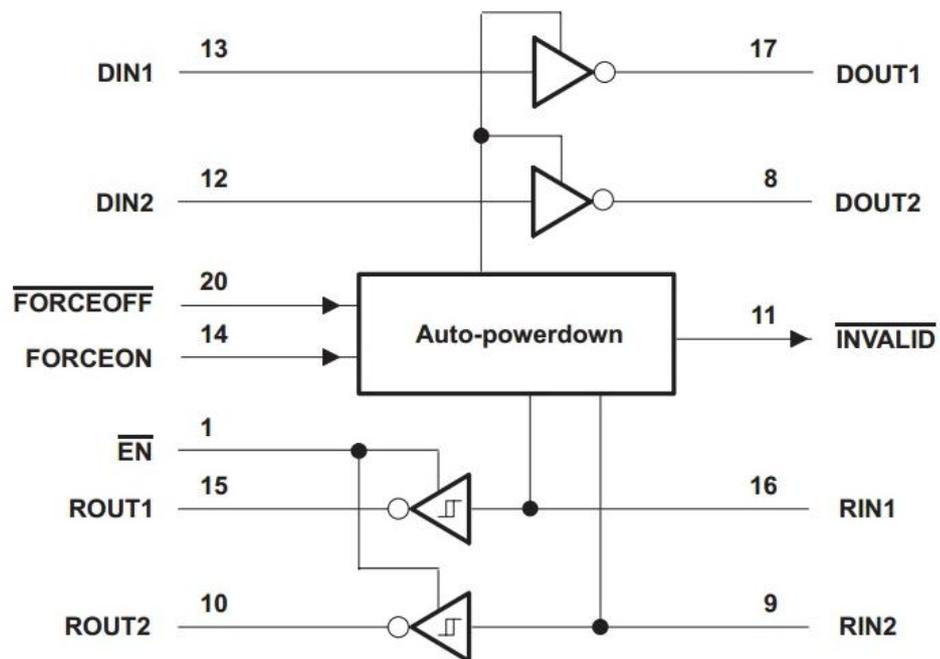


圖 2-4:MAX3223CPWR 內部電路圖

第三章 系統架構與軟韌硬體設計

3.1 系統架構

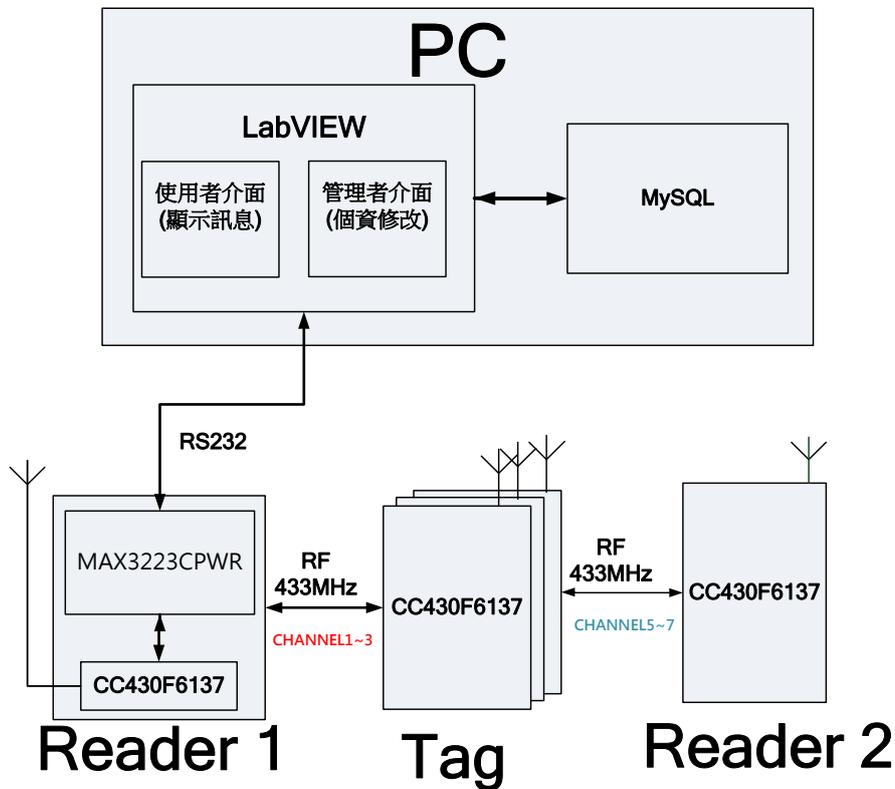


圖 3-1: 系統架構圖

1. PC

(1) LabVIEW

使用者介面，顯示名字、時間、圈數、速率，給使用者觀看。

管理者介面，提供管理者修改資料及維護資料。

(2) MySQL

建立資料庫之使用軟體。

2. Reader

(1) MAX3223CPWR

將 PC 端的 RS232 的電壓準位轉換至 CC430 晶片可使用的電壓。

(2)CC430F6137

讀取 Tag 的編碼。

3. TAG

等待 Reader 的喚醒並傳輸資料給 Reader

3.2 流程圖

3.2.1 系統流程圖

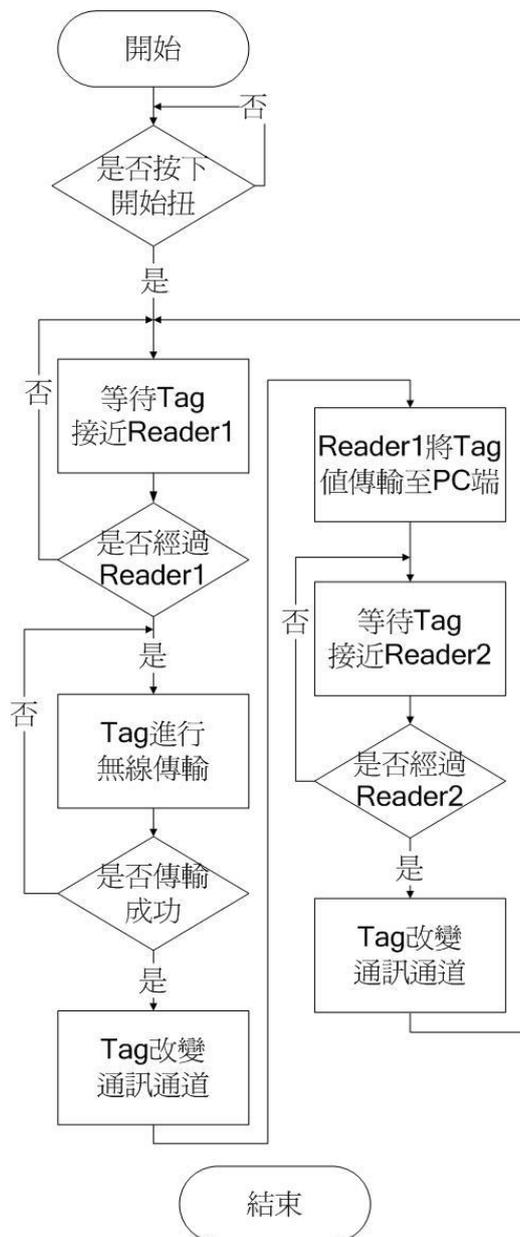


圖 3-2:系統流程圖

如圖3-2所示，一開始READER1會發送訊號給TAG，等TAG接收到訊號後會回傳自身編碼給READER1，當READER1接收到編碼後，TAG將改變通道，之後，READER1會將TAG的值傳輸至PC端進行處理。

如果需要更新PC端資料，則TAG必須經過READER2，當TAG接到READER2訊號後，將會改變回原本預設的通道，並可再與READER1溝通，以此循環。

3.2.2 LabVIEW 介面流程圖

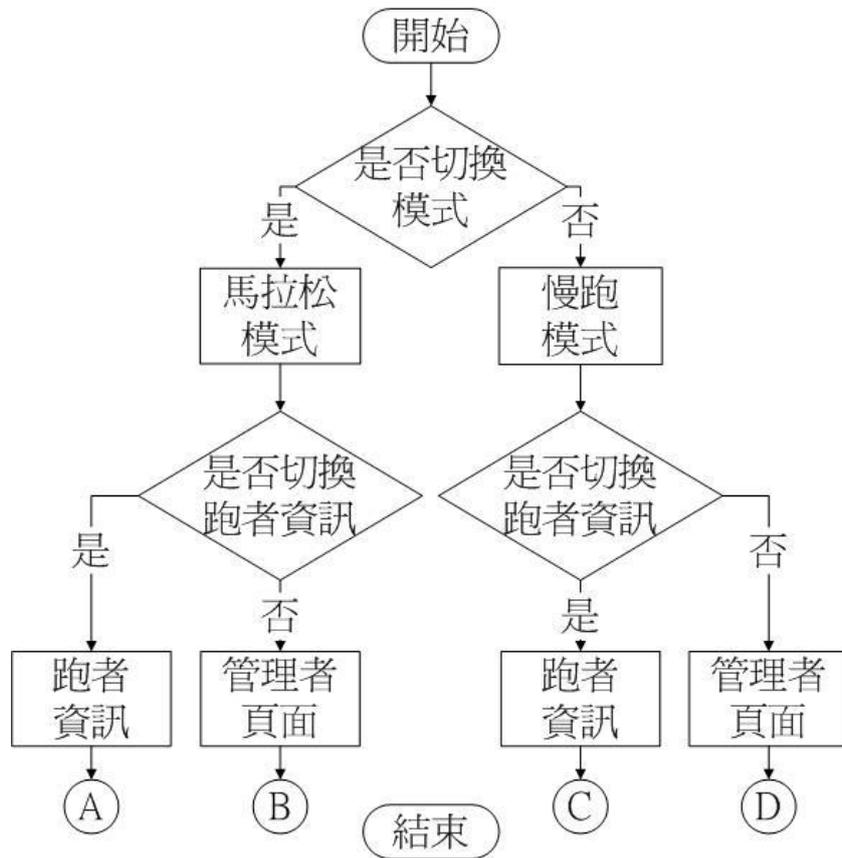


圖3-3:LabVIEW介面流程圖-1

如圖3-3中，當系統啟動時，會先確定是否需要切換模式，是，則切入馬拉松模式；否，則切入慢跑模式。在兩種模式下，都有切換

跑者資訊的按鈕,按下的話,頁面將會切換至跑者資訊;未按下的話,將會顯示管理者介面。

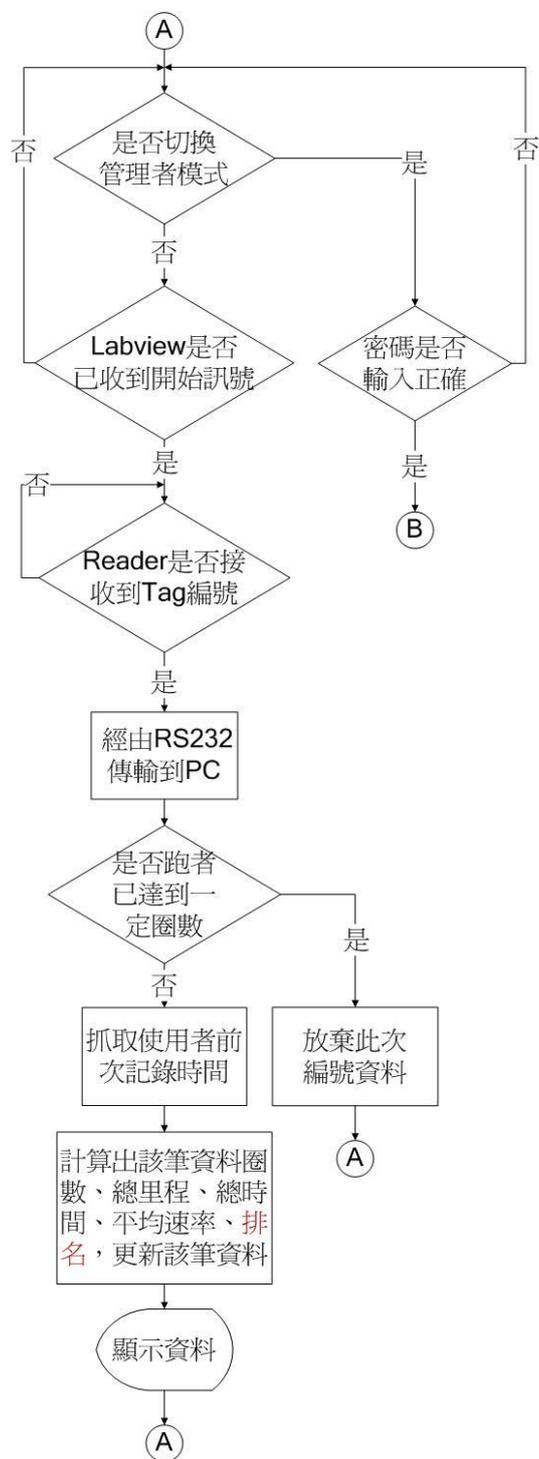


圖3-4:LabVIEW流程圖-2(馬拉松模式)

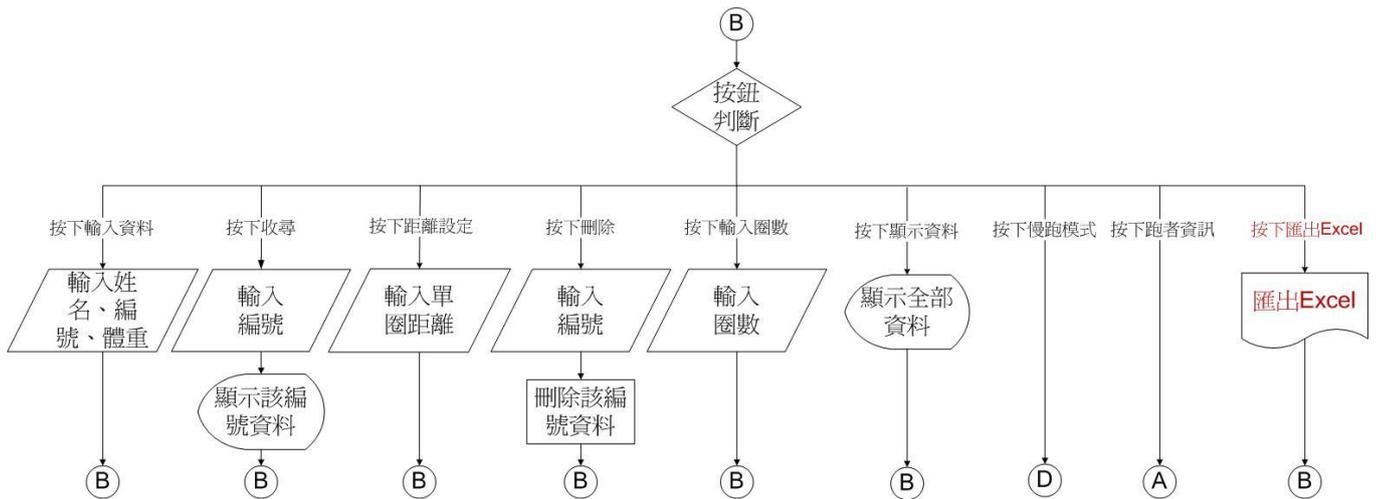


圖3-5:LabVIEW流程圖-3(馬拉松模式)

如圖3-4與圖3-5中，當在馬拉松模式下，READER1等待接收Tag訊號，有接收到Tag訊號，則將Tag內的數值經由RS-232傳輸到PC端；無接收到訊號則繼續等待。

當PC端接收到訊號後，會先判斷是否跑者已達到規定的圈數，否，則抓取使用者前次記錄時間，並計算出圈數、總里程、總時間、平均數率與排名，並顯示在螢幕上；是，則放棄此次編號資料。

在管理者介面下，比賽前可輸入使用者資料、收尋使用者及刪除使用者資料，也可以設定跑道單圈距離，比賽結束後可將比賽結果以Excel檔匯出。

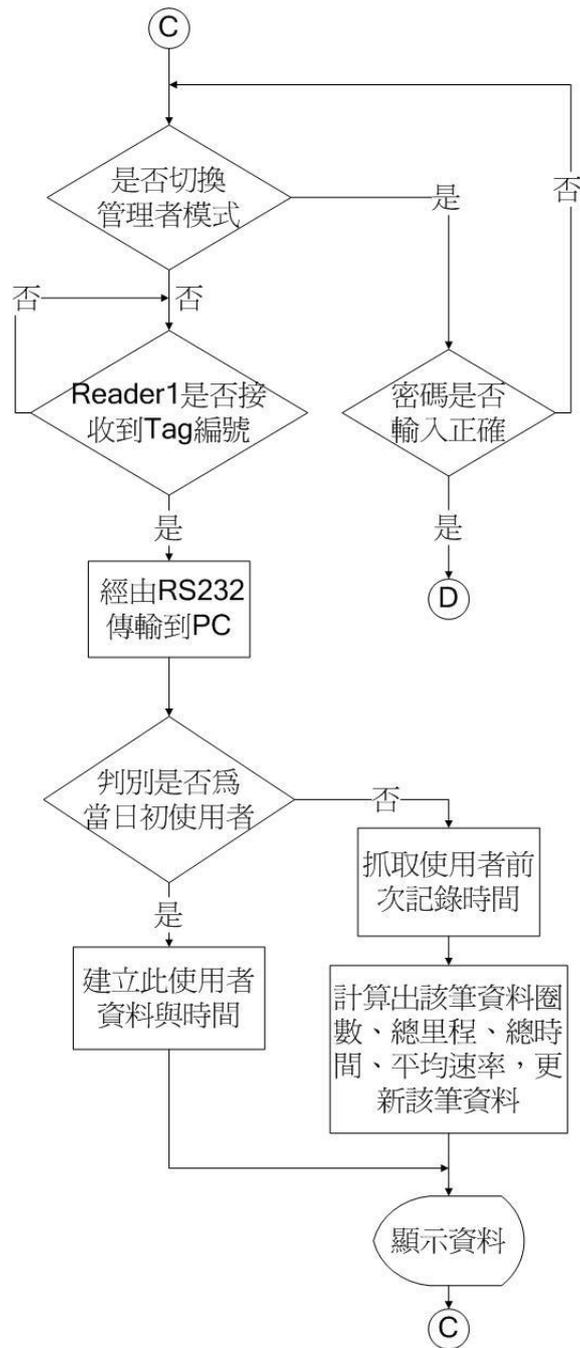


圖3-6:LabVIEW流程圖-4(慢跑模式)

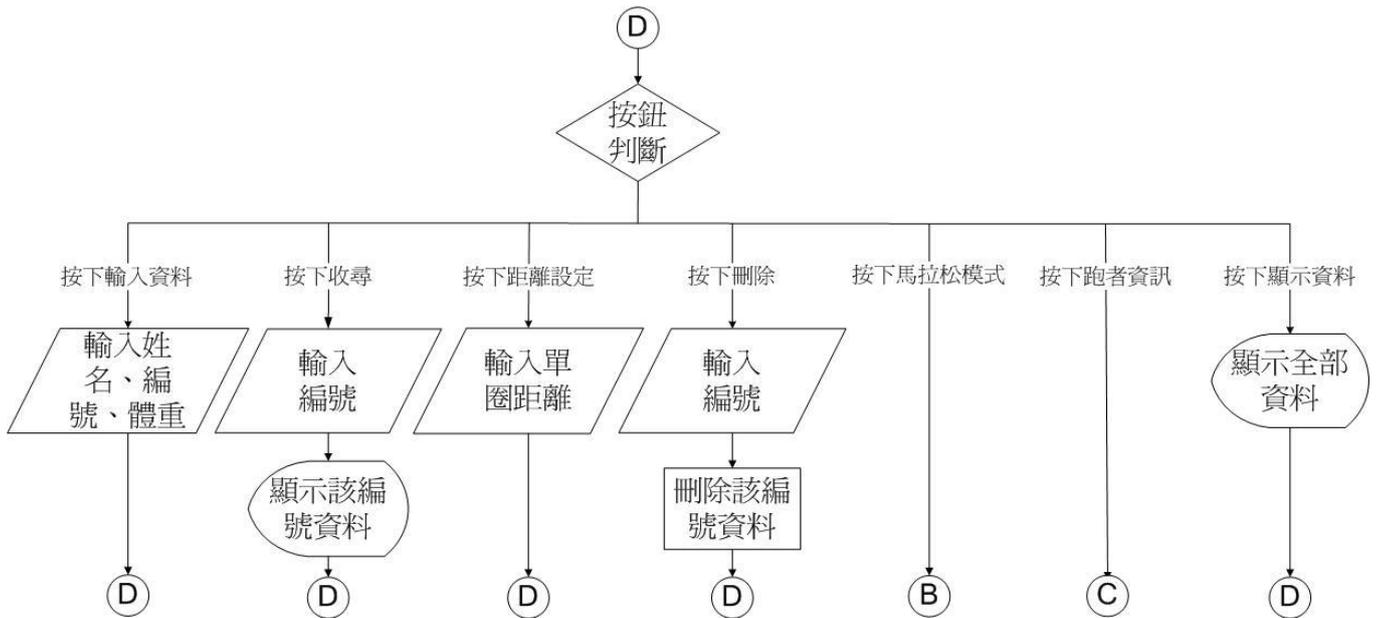


圖3-7: LabVIEW流程圖-5(慢跑模式)

如圖3-6與圖3-7中，當在慢跑模式下，READER1等待接收Tag訊號，有接收到Tag訊號，則將Tag內的數值經由RS-232傳輸到PC端；無接收到訊號則繼續等待。

當PC端接收到訊號後，判斷是否為當日使用者，是，則建立此使用者資料與時間；否，則抓取使用者前次記錄時間，並計算出該筆資料圈數、總里程、總時間、平均數率，並更新該筆資料，顯示在螢幕上。

在管理者介面下，比賽前可輸入使用者資料、收尋使用者及刪除使用者資料，也可以設定跑道單圈距離。

3.3 軟韌硬體設計

本專題將系統架構分成三個部分來做設計，分別為軟體、韌體、硬體。

硬體：READER的製作(採用Altium Designer軟體製作)，洗製電路板(曝光、顯影、鑽孔、焊原件)，與使用eZ430-Chronos Development Tool 作為TAG。

韌體：電腦與READER透過RS-232溝通、TAG與READER之間的溝通，採用Code Composer Studio v5來撰寫程式。

軟體：以LabVIEW製作使用者介面，並透過ODBC((Open Data Base Connectivity)與MySQL資料庫做連結。

3.3.1 硬體

1. TAG:

採用德州儀器的 eZ430-Chronos Development Tool，如圖 3-8。

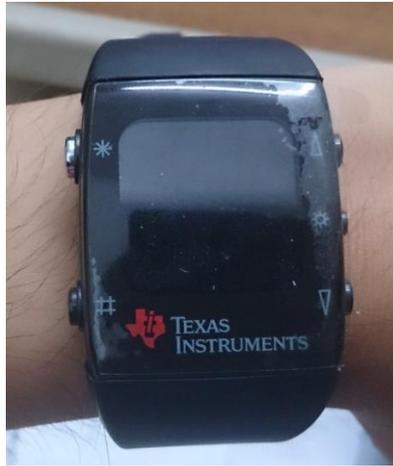


圖 3-8: eZ430-Chronos Development Tool

2. READER:

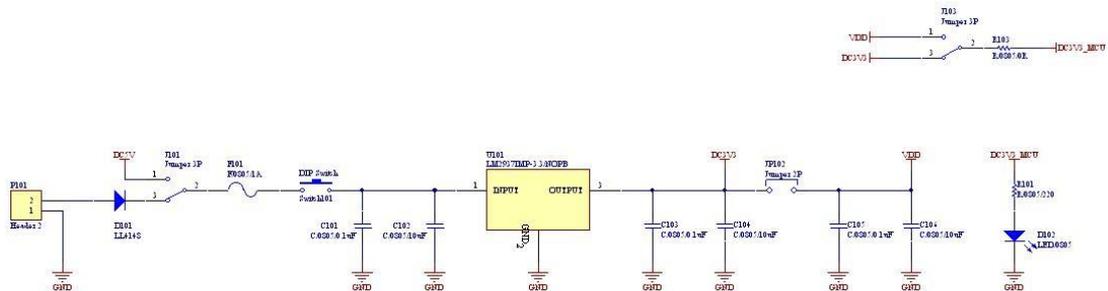


圖 3-9: READER 之電源電路圖

A. LM2937IMP-3.3/NOPB

這是一顆電壓轉換 IC，因為我們所使用的的電路為 DC3.3V 其最大耐壓為 26V，瞬間耐壓為 60V，我們利用此晶片做為外部輸入電源的穩壓晶片。

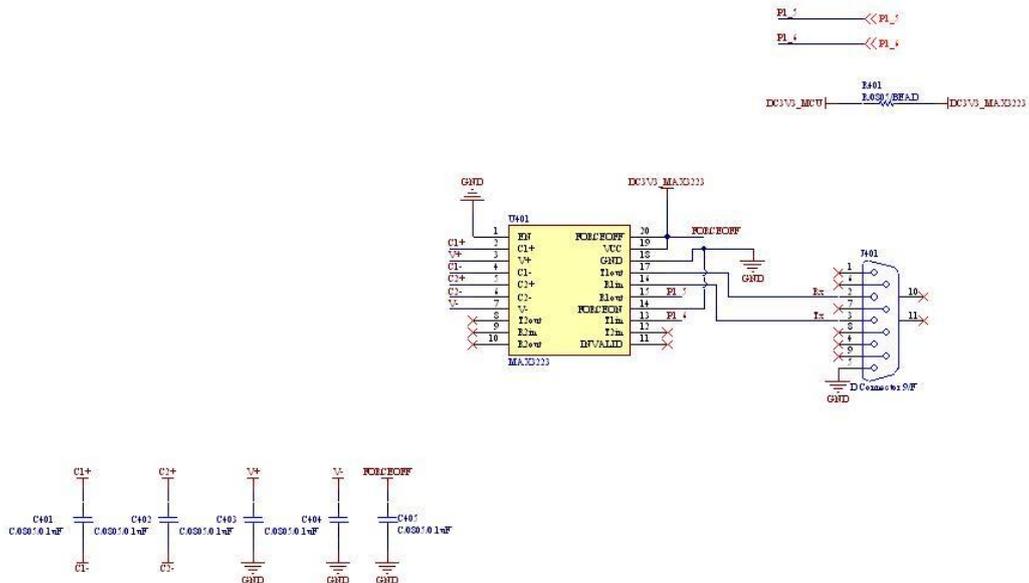


圖 3-10:MAX3223CPWR 與 RS-232 之電路圖

B. M3223CPWR

選擇此晶片是為了讓 CC430F6197 晶片電壓 3.3V 升至 5V，才可與 RS-232 接頭進行溝通。其 Forceoff 為高準位、forceon 為低準位、EN 為低準位時才可進行傳輸，接收時，訊號經由 RS-232 的 3 腳進入 R1in 再經由 R1out 進入 CC430F6137 傳輸時，訊號經由 T1in 進入 MAX-3223 再經由 R1out 進入 RS-232 的 2 腳藉由這樣的方式，將 CC430F6137 接收到的資訊傳遞至電腦進行比較。

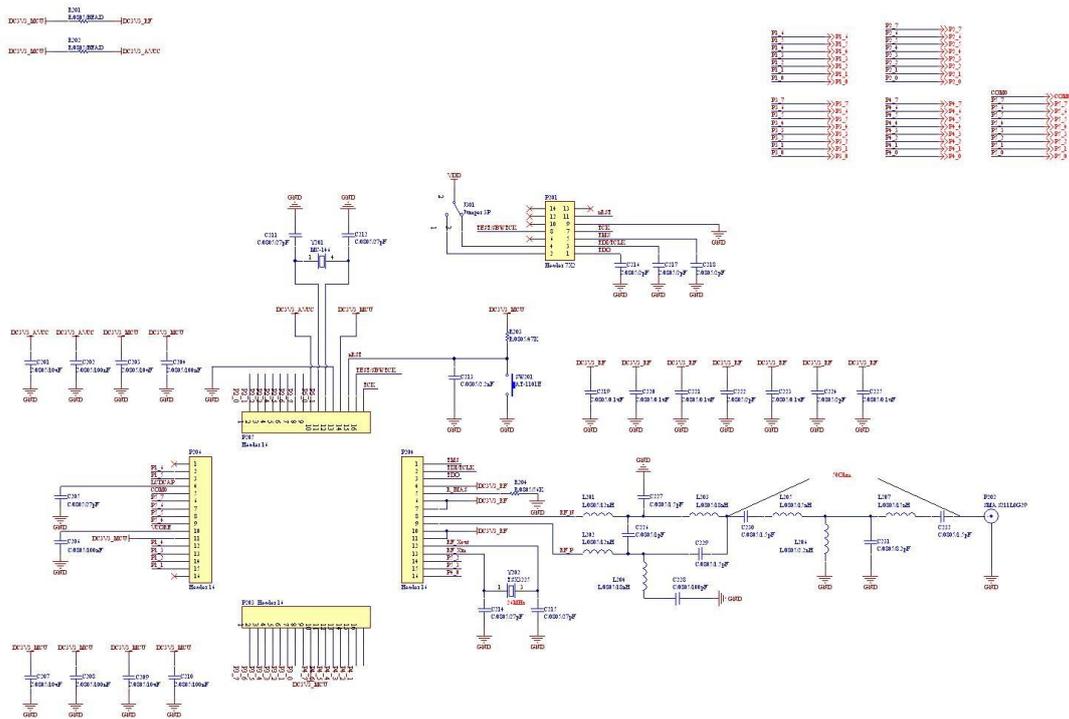


圖 3-11:CC430F6137 晶片與 RF 功能之電路圖

C. CC430F6137

震盪器會先產生 433MHz 的頻率，此時其頻段為 433MHz，愈等待同樣為 433MHz 的 Tag 傳送訊號。

當其透過 RF 接收到 Tag 傳送過來的信號時，判斷是否為可溝通通道，是，將通道內資訊精油 RS-232 傳回電腦；否，則繼續等待接收通道。

3.3.2 韌體

1. 電腦與READER透過RS-232溝通

當READER接收到TAG的編碼，RAEDER會經由RS-232傳輸至電腦做資料的處理。

2. TAG與READER之間的溝通

(1)防碰撞:

為了解決防碰撞問題，我們利用時間去切換不同的通道(CHANNEL)，利用這項技術來達到防碰撞的效果。

經過我們的實驗後，RAEDER傳輸至TAG再由TAG回傳值給READER的時間大概是8ms，但我們發現，每10ms切換一次通道，穩定性較8ms好，因此採用每10ms切換一次通道來實現防碰撞的效果。

(2)RSSI(接收訊號強度):

利用RSSI計算Reader位置與Tag位置之距離變化，訊號強度變化用以準確判別跑者是否經過紀錄點如圖3-12所示，訊號強度迅速變化過程中，當訊號的強度到達我們設定之門檻值時，Reader讀取Tag編碼，並進行運用。

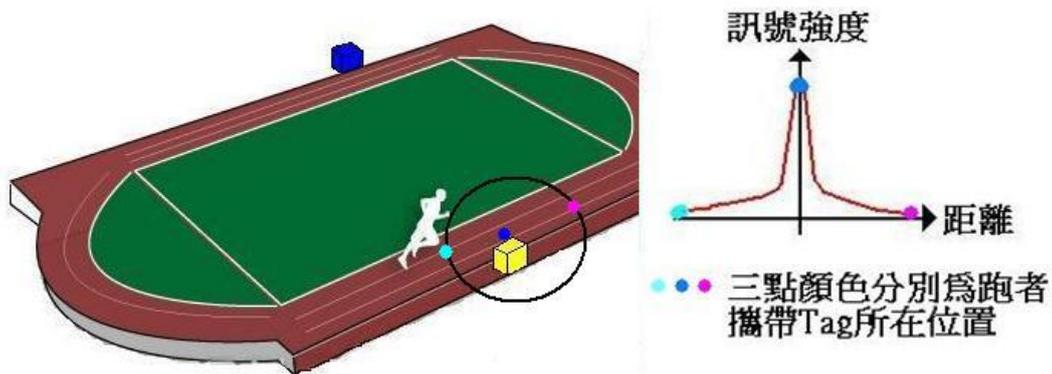


圖3-12: RSSI訊號強度與距離示意圖

另外透過以下兩點公式，我們粗估可得到表3-1的結果

a. $RSSI_dec \geq 128$, then $RSSI_dBm = (RSSI_dec - 256) / 2 - RSSI_offset$

b. $RSSI_dec < 128$, then $RSSI_dBm = (RSSI_dec) / 2 - RSSI_offset$

註: $RSSI_offset$ 根據CC430F6137的Data Sheet 在常溫25度，定電壓在3V之下，
433MHz $RSSI_offset$ 值為74dB

水平距離(cm)	RSSI(dBm)	垂直距離(cm)	RSSI(dBm)
0	-54~-57	0	-52~-53
5	-68~-71	5	-43
10	-74~-77	10	-72~-73
15	-76~-78	15	-69~-71
20	-74~-77	18.5	-70~-72
25	-82~-85		
30	-81~-86		
35	-80~-85		
40	-79~-84		
45	-86~-89		
50	-80~-86		

表3-1:RSSI公式轉換結果

3.3.3 軟體

1. LabVIEW之介紹

LabVIEW為Laboratory Virtual Instrument Engineering Workbench的簡稱，是由Nation Instrument公司，於1986年所發展出的一種圖像程式語言(Graphic Language)，又有人稱之為G語言，它是使用圖像物件函數的方式編輯程式，取代傳統採用文字編輯方式，使得使用者更容易了解程式結構的涵義，也因為如此，它比一般其它的語言容易學習。

對於LabVIEW程式結構而言，有如虛擬儀表(Virtual Instrument)一般，可利用電腦透過網路方式，對遠端的控制物件下達指令。系統的通訊介面包含有GPIB、VXI、PXI、RS-232、RS-485等，可使用視窗作業系統，設定與驅動儀器的通訊介面，亦可支援多種與外部信號連接的方式，例如TCP/IP網路通訊協定、Active X、及DLLs等功能。

LabVIEW在儀器控制與測量方面，也提供相當大的功能，其與儀器的連接介面，可透過GPIB(General Purpose Interface Bus)，如IEEE-488通訊介面。亦可使用儀器所提供較低階的RS-232通訊介面，由區域網路的方式直接與電腦連線，從事訊號的測量、分析、數據儲存、與資料的擷取等功能，提升工作效率與數據資料的準確性。

LabVIEW系統也提供數位與類比的轉換功能，如資料擷取

DQA(Data Acquisition)系統，透過資料及取介面卡，取得類比訊號之後，在轉換為數位訊號，讓一般電腦能夠判讀所擷取的數位訊號。

同樣地，也可以藉由介面的訊號轉換功能，把電腦的指令由數位訊號轉換為類比訊號，來驅動被控制物件，以達到訊號擷取目的。

LabVIEW 原本被廣泛的運用在實驗室中，但現在有許多工廠也使用此系統來將所測量到的資料顯示在電腦螢幕上，用來取代傳統的示波器，在存取及分析資料方面也比傳統示波器方便許多。

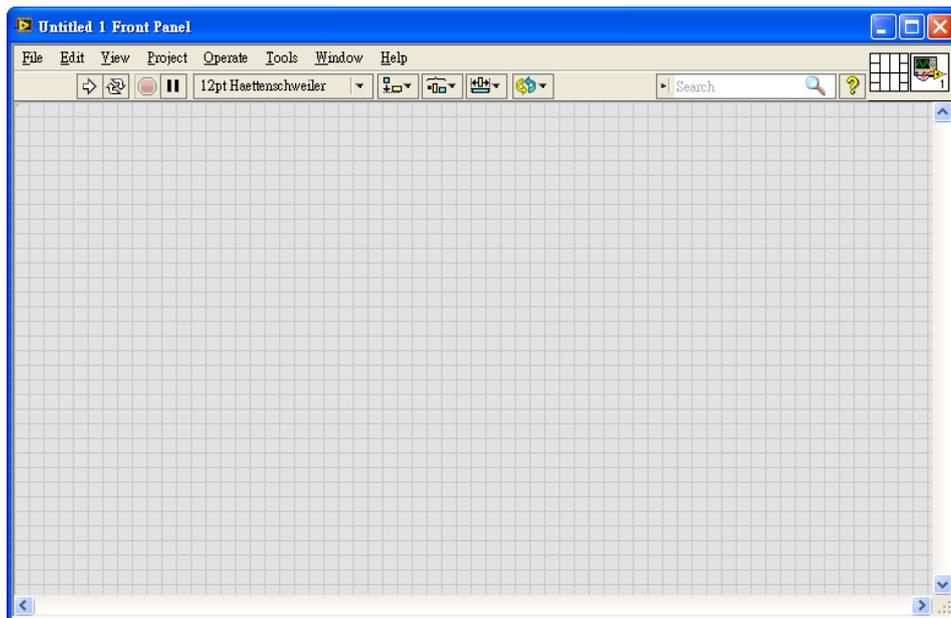


圖 3-13:LabVIEW Panel 介面

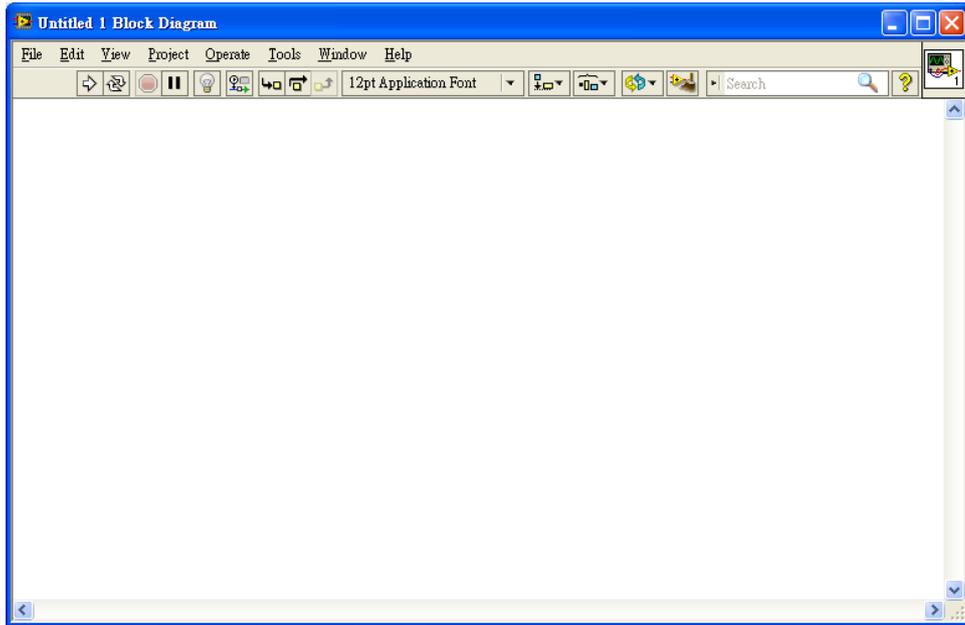


圖 3-14:LabVIEW Diagram 介面

1. LabSQL

LabSQL 是一個免費的、多資料庫、跨平臺的 LabVIEW 資料庫訪問工具包。LabSQL 支援 Windows 作業系統中任何基於 ODBC 的資料庫，包括 Access, SQL Server, Oracle, Pervasive, Sybase 等。

LabSQL 是利用 Microsoft ADO 以及 SQL 語言來完成資料庫訪問，將複雜的底層 ADO 及 SQL 操作封裝成一系列的 LabSQL VIs。利用 LabSQL 幾乎可以訪問任何類型地資料庫，執行各種查詢，對記錄進行各種操作，如圖 3-15。

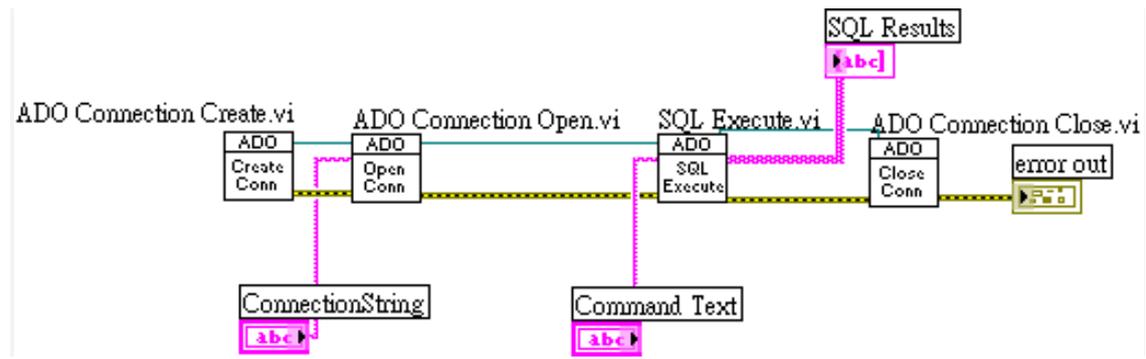


圖 3-15: labview 與資料庫(MySQL)溝通模組

使用 SQL 語法，可以對資料庫做讀取、寫入、修改或是刪除……等，以下為簡易讀取、寫入、刪除的範例，如圖 3-16 與圖 3-17。

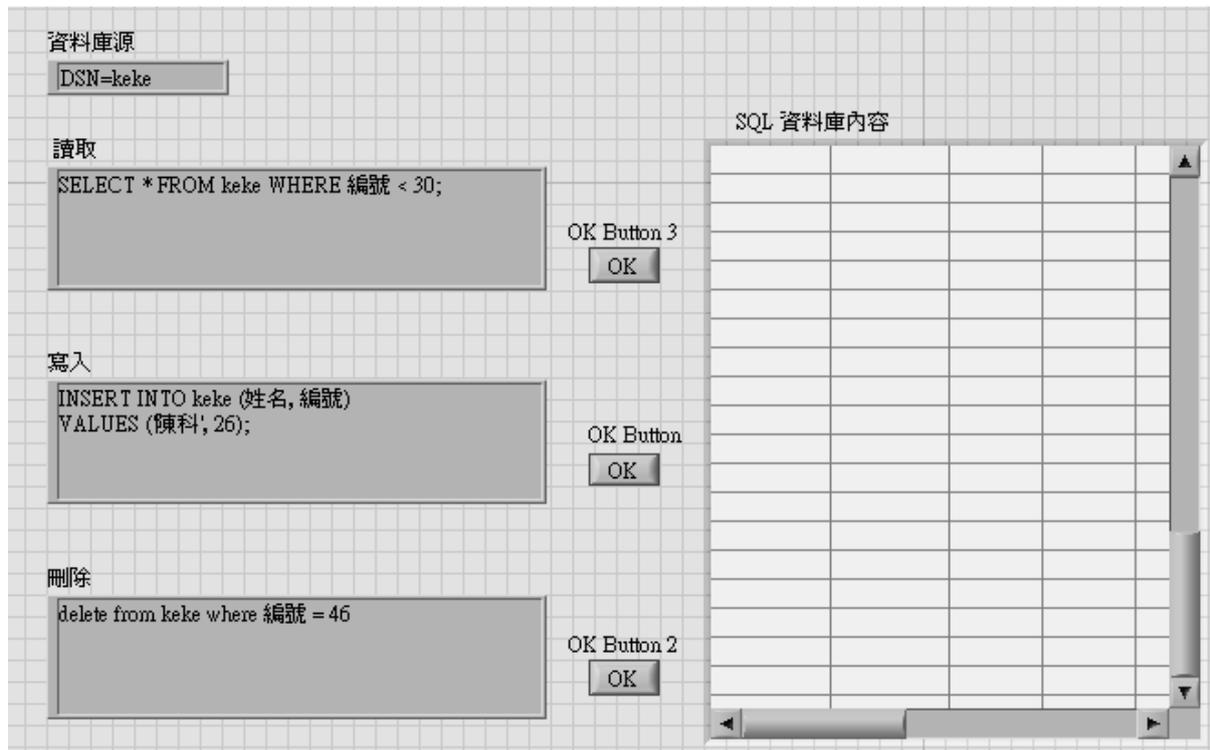


圖 3-16: 使用 SQL 與法之正版

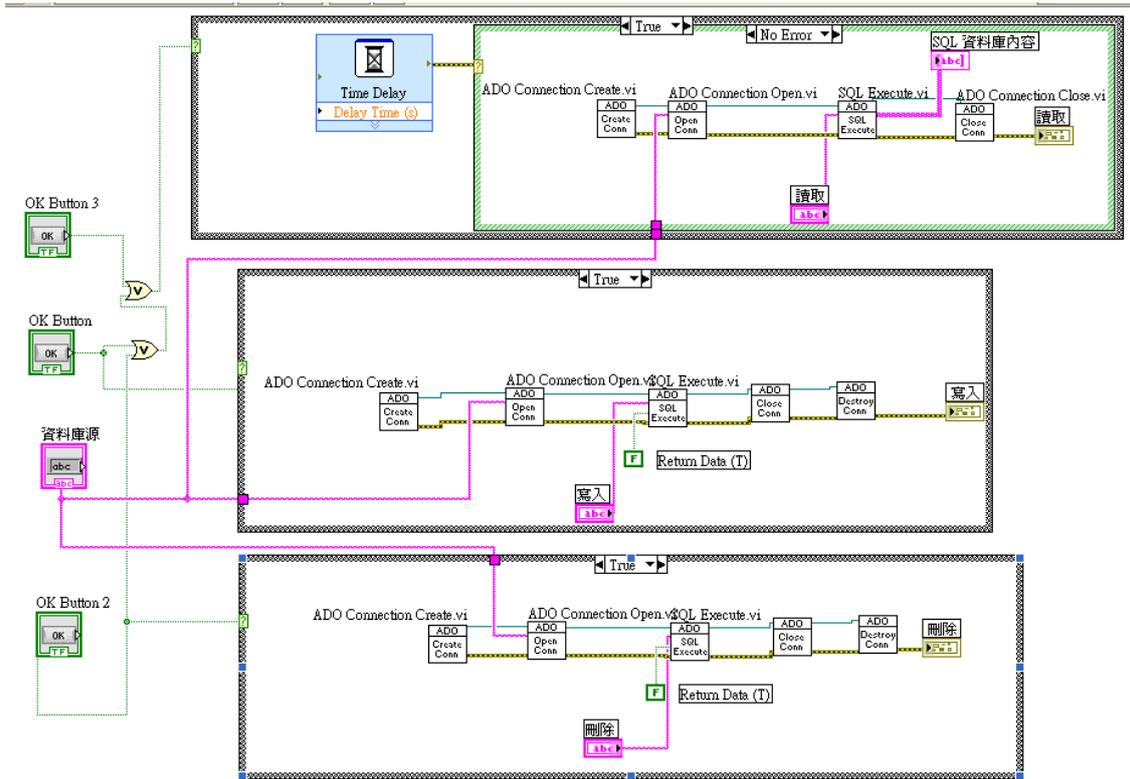


圖 3-17: 使用 SQL 與法之背版

我們可以進階運用讀取與修改的語法，來達到更新資料的功能，

如圖 3-18 與圖 3-19。

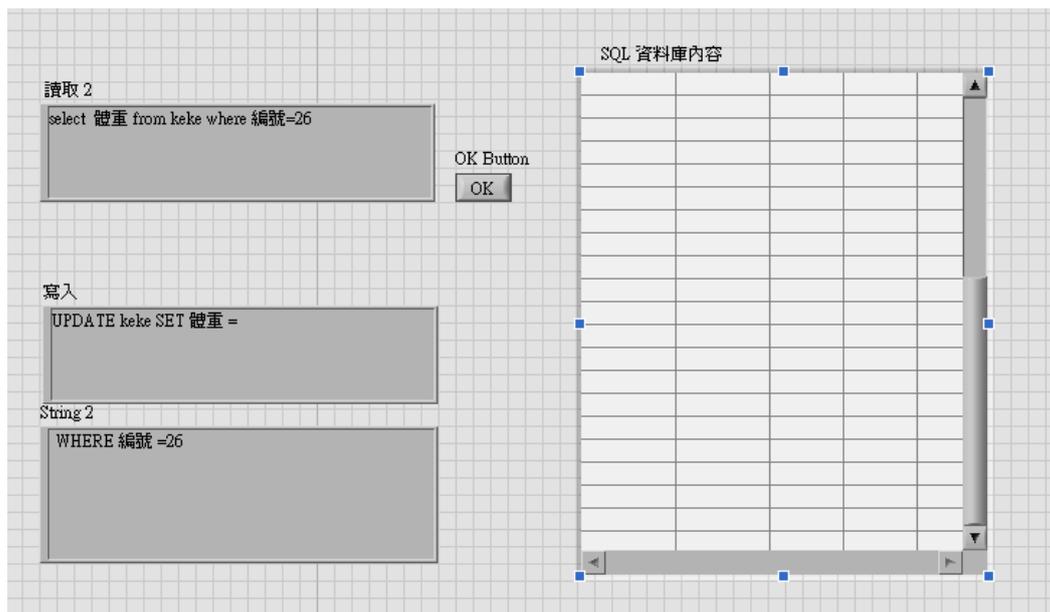


圖 3-18: 進階運用讀取與修改語法之正版

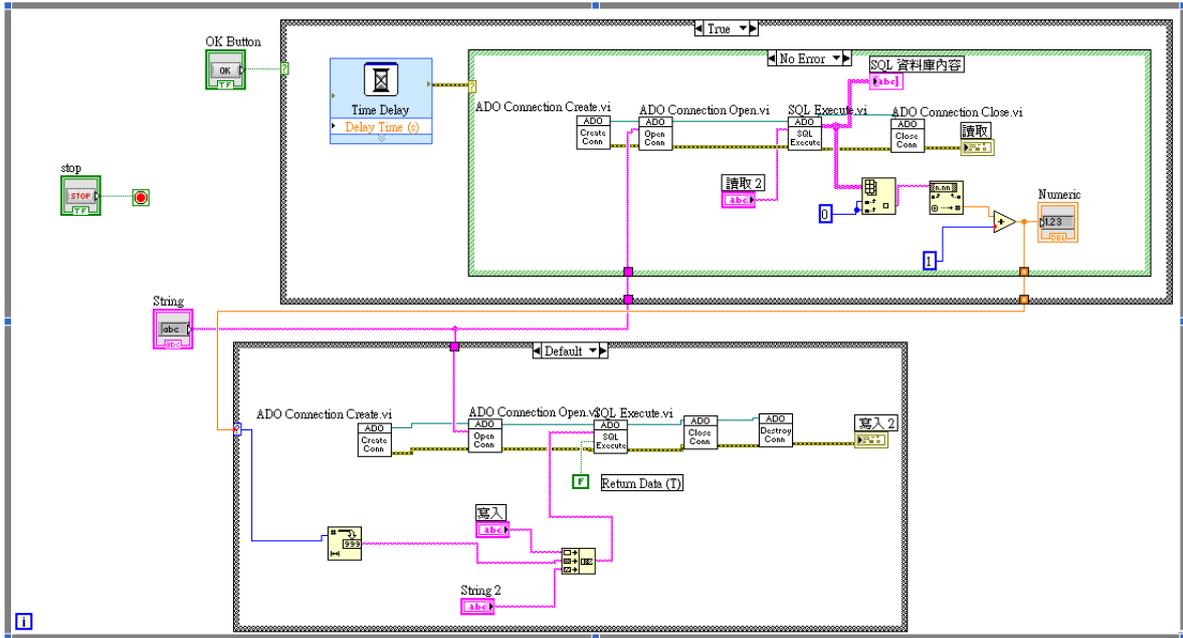


圖 3-19: 進階運用讀取與修改語法之背版

LabVIEWE 內建模組 RS-232 功能，如圖 3-20 與圖 3-21。



圖3-20:RS-232模組正版

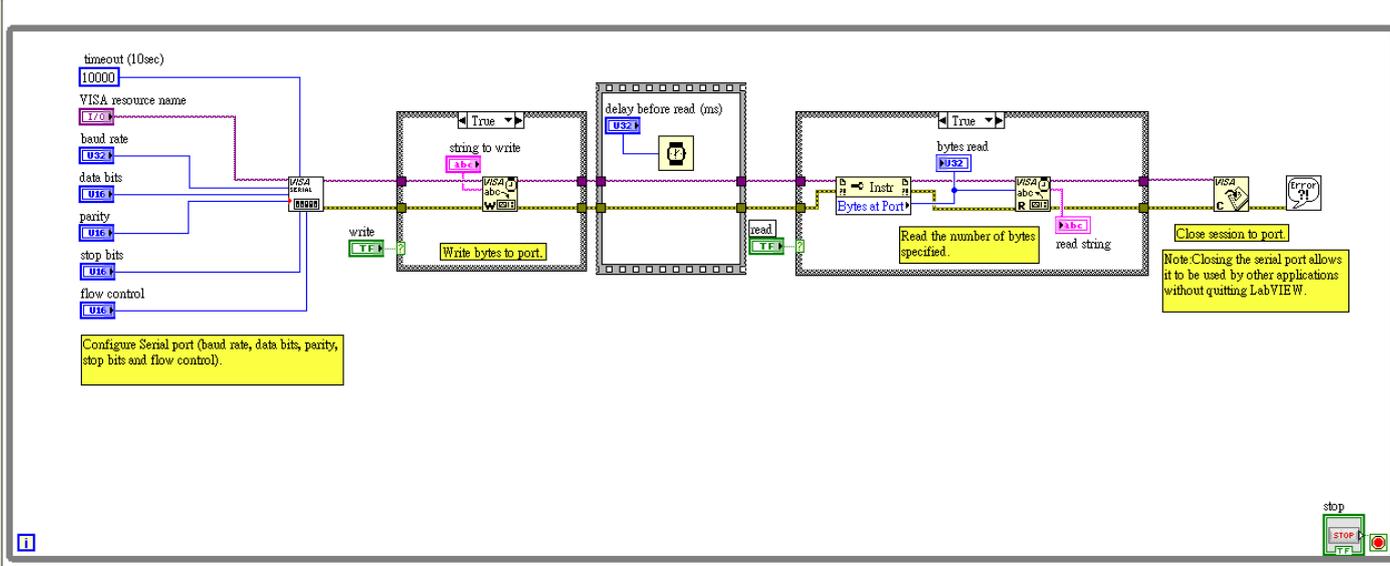


圖3-21:RS-232模組背版

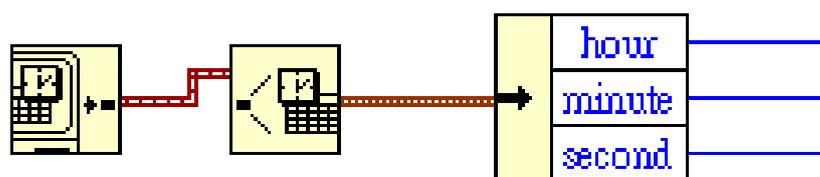


圖 3-22:抓取電腦時間，並做應用。

抓取今日資料，並匯出 Excel 檔，紅圈為匯出 Excel 的模組，黃圈為抓取今日資料的功能，如圖 3-23。

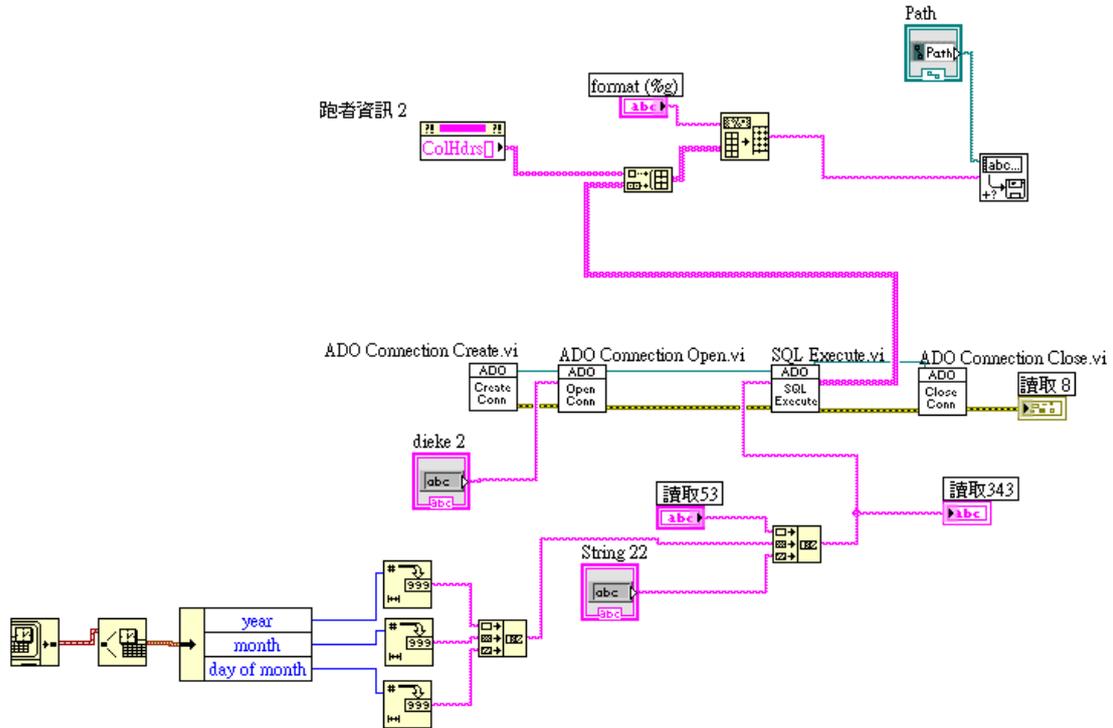


圖3-23:抓取資料與匯出EXCEL之模組

LabVIEW換頁功能是使用迴圈的移位暫存器來保持換頁後的頁面，這裡是將換頁鍵直接做成陣列，偵測哪個按鍵有動作來控制換頁，如圖 3-24。

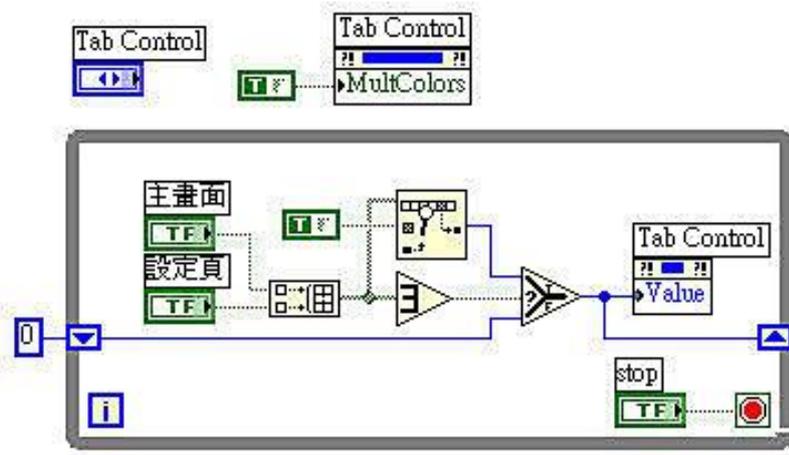


圖3-24:換頁模組

特殊語法排名:

顯示當日的排名、姓名、圈數、總里程、平均速度、總時間、時間、結束時間。排名部分的判斷是先以圈數比較，再以平均速率來決定名次，並以名次先後順序顯示。當圈數和平均速率一樣時，則名次一樣。(例如:當有兩個第一名，下個名次則為第三名。), 如圖 3-25。

讀取53 2

```
SELECT a.rank AS 排名,a.姓名,a.圈數,a.總里程,a.平均速率,
a.總時間,a.時間,a.結束時間 FROM (SELECT 姓名,圈數,總里程,
平均速率,總時間,時間,結束時間, @prev := @curr,@aa:=@bb,
@curr := 圈數,@bb:=平均速率, @rank := IF(@prev = @curr and
@aa=@bb, @rank, @rank+1) AS rank
FROM dieke, (SELECT @curr := null, @prev := null,@aa:=null,@bb:
=null, @rank := 0) s WHERE 年月日 =
```

String 22

```
ORDER BY 圈數 DESC,
平均速率 asc)a;
```

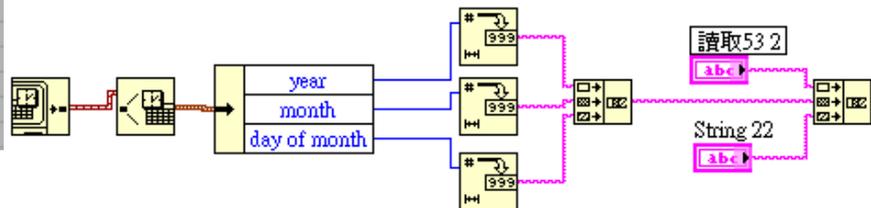


圖3-25:特殊與法排名模組



圖4-5:軟體管理者介面(慢跑模式)

4.2 操作流程

A. 慢跑模式:

1. 輸入跑者資訊，如圖4-6及4-7。



圖4-6:跑者資訊輸入介面-1

6. 重複步驟2至步驟5，當跑者結束慢跑時，系統會顯示最後一次經過READER1之資料。

7. 當切換至管理者模式時，必須輸入密碼(如圖4-13)。

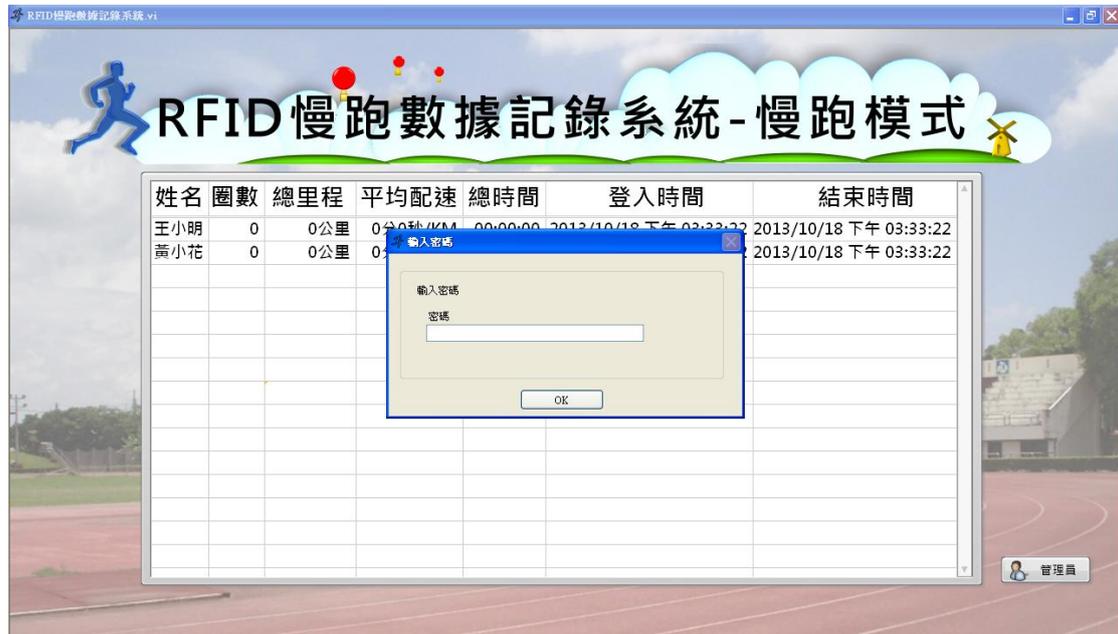


圖4-13:輸入密碼之畫面

B. 馬拉松模式:

1. 輸入跑者資訊，如圖4-14及4-15。



圖4-14:跑者資訊輸入介面-1



圖4-15:跑者資訊輸入介面-2

2. 馬拉松模式使用者介面，比慢跑模式額外增加了排名，如圖4-16。



圖4-16: 馬拉松模式使用者介面

3. 根據我們的設計，馬拉松模式與慢跑模式的整體流程相同，但在此模式下，額外增加排名及匯出EXCEL檔(如圖4-17)。

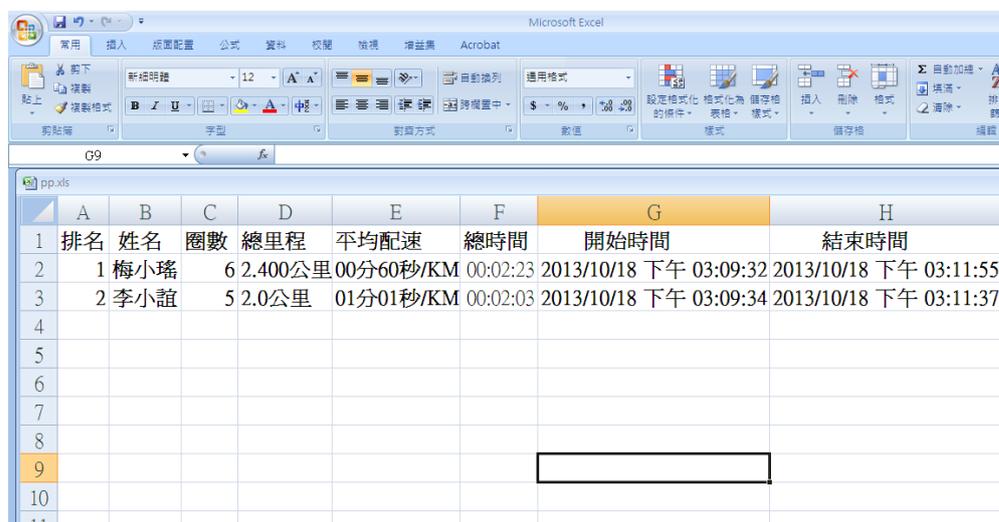


圖4-17: 匯出EXCEL檔

第五章 結論與未來展望

5.1 結論

本系統在實際運用上還差了幾步，首先RF訊號有時會受到干擾導致RSSI瞬間訊號增強 導致我們定位判斷上的錯誤 所以我們以寫程式的方式來改善這種干擾現象，但是，還是不太穩定，這是我們未來必須加以探討如何改善這一點；第二點，我們的資料庫在網路化方面踢到了鐵板，這也是我們必須改善的缺點之一；第三點，由於天線不是指向性天線，所以有時READER無法準確的定位，在未來必須也改善這一點。

5.2 未來展望

- 計算消耗熱量。
- 製作網頁供使用者上網修改個人資訊及查詢個人的運動數據。
- MySQL遠端資料庫連結。

參考文獻

1. IT360技術文章：LabSQL+MySql Step by Step
<http://it360.tw/article/info.asp?TID=10158&FID=165>
2. IT論壇：資料庫：mySQL
http://it360.tw/forum/forum_topics.asp?FID=171
3. 還未出師的小工程師部落格 LabVIEW程式&資料庫
<http://tw.myblog.yahoo.com/liu-yunan/archive?l=f&id=13&page=1>
4. LabVIEW Pro 專業論壇
<http://www.labviewpro.net/>
5. 使用 LabSQL 撰寫資料庫程式 @ 簡單過生活 :: 痞客邦 PIXNET ::
<http://iammic.pixnet.net/blog/post/27977678>
6. 魚的方舟 * 圖形語言的世界* - LabVIEW 工作室 & APP開發研究室
<http://fishark.pixnet.net/blog/category/735962>
7. [SQL]進階之排名Rank/Median/Total/Percent - 芝麻開花 - 博客大巴
<http://flypiaoxu.blogbus.com/logs/165731830.html>
8. LabVIEW討論區AVR與虛擬儀器論壇AVRvi.com
<http://bbs.avrvi.com/thread-htm-fid-8.html>
9. MySQL :: The world's most popular open source database
<http://www.mysql.com/>
10. 與非網電子技術論壇-德州儀器技術社區論壇-MSP430
http://www.eefocus.com/bbs/forum_1077.html
11. 德州儀器官方網站
<http://www.ti.com/ww/tw/>
12. 德州儀器在線技術支持社區
<http://www.deyisupport.com/>

附 錄

附錄A

- RFID 的接收:

我們可以利用內部的副程式 ReceiveOn() 開啟 RFID 的接收功能，當有接收到訊號時會跳至如下的中斷副程式中，此時接收到的值會存在 RxBuffer 的記憶體位子，如下圖 a，[0]與[1]是接收到的值，[2]是 RSSI 強度，[3]是 LQI(連線品質)。

```
#pragma vector=CC1101_VECTOR
__interrupt void CC1101_ISR(void)
{
    switch(__even_in_range(RF1AIV, 32))        // Prioritizing
Radio Core Interrupt
    {
        case 20:                               // RFIFG9
            if(receiving)                       // RX end of packet
            {
                RxBufferLength = ReadSingleReg( RXBYTES );
                ReadBurstReg(RF_RXFIFORD, RxBuffer, RxBufferLength);
                __no_operation();
            }
            else if(transmitting)               // TX end of packet
            {
                RF1AIE &= ~BIT9;
                transmitting = 0;
            }
    }
```

```

else while(1);          // trap
break;
}
__bic_SR_register_on_exit(LPM3_bits);
}

```

Expression	Type	Value
RxBuffer	unsigned char[4]	0x1C26 (Hex)
(x)= [0]	unsigned char	0xFF (Hex)
(x)= [1]	unsigned char	0xFF (Hex)
(x)= [2]	unsigned char	36 (Decimal)
(x)= [3]	unsigned char	154 (Decimal)

圖 A: RxBuffer 值之觀察室窗

附錄 B

● RFID 的傳輸:

我們可以利用內部的副程式如下，Transmit((unsigned char*)TxBuffer, sizeof TxBuffer); 並會跳入對應的中斷副程式中，然後進行 RFID 的傳輸，如下程式碼。

```

#pragma vector=CC1101_VECTOR
__interrupt void CC1101_ISR(void)
{
    switch(__even_in_range(RF1AIV, 32))          // Prioritizing
Radio Core Interrupt
    {
        case 20:                                // RFIFG9
            if(receiving)                        // RX end of packet

```

```

    {
        RxBufferLength = ReadSingleReg( RXBYTES );
        ReadBurstReg(RF_RXFIFORD, RxBuffer, RxBufferLength);
        __no_operation();
    }
    else if(transmitting)          // TX end of packet
    {
        RF1AIE &= ~BIT9;
        transmitting = 0;
    }
    else while(1);                // trap
    break;
}
__bic_SR_register_on_exit(LPM3_bits);
}

```

附錄 C

● RS232 的使用：

我們必須先預設 RS232 的 TX 的腳位與鮑率設定的設定等等的初始設定，String1 為傳輸的文字，並搭配看門口的中斷切換，完成 RS232 的傳輸，範例程式如下。

```

WDTCTL = WDT_ADLY_1000;          //看門口中斷，時間為 1S
SFRIE1 |= WDTIE;
PMAPPWD = 0x02D52;
P1MAP6 = PM_UCA0TXD;            //腳位設定

```

```

PMAPPWD = 0;
P1DIR |= BIT6 ;
P1SEL |= BIT6;
UCA0CTL1 |= UCSWRST;
UCA0CTL1 = UCSSEL_1;
UCA0BR0 = 0x03; //鮑率設定
UCA0BR1 = 0x00;
UCA0MCTL = UCBRS_3+UCBRF_0;
UCA0CTL1 &= ~UCSWRST;
DMACTL0 = DMA0TSEL_17;
__data16_write_addr((unsigned short) &DMA0SA, (unsigned long)
String1);
__data16_write_addr((unsigned short) &DMA0DA, (unsigned long)
&UCA0TXBUF);
DMA0SZ = sizeof(String1);
DMA0CTL = DMASRCINCR_3+DMASBDB+DMALEVEL;
__bis_SR_register(LPM3_bits + GIE); //等待看門狗的中斷

#pragma vector=WDT_VECTOR //看門口中斷副程式
__interrupt void WDT_ISR(void)
{
    DMA0CTL |= DMAEN; // Enable
    __bic_SR_register_on_exit(LPM3_bits);
}

```

附錄 D

- RFID 通道切換:

RFID 除了頻段必須一樣之外，還需配合相同的通道的搭配，才能溝通，所以我們可以利用下列的程式碼，切換通道。(channel 為通道編號)

```
rfSettings.channr=channel;  
WriteRfSettings(&rfSettings);
```

附錄 E

- RSSI 的使用:

當接收到訊號時，(X)=[2]為 RSSI 的值如圖，我們必須經由公式 () 換算成 RSSI_dBm 才能使用，轉換程式碼如下。

Expression	Type	Value
RxBuffer	unsigned char[4]	0x1C26 (Hex)
(*)- [0]	unsigned char	0xFF (Hex)
(*)- [1]	unsigned char	0xFF (Hex)
(*)- [2]	unsigned char	36 (Decimal)
(*)- [3]	unsigned char	154 (Decimal)
(*)- RSSI_Value	int	-56

RSSI_dec

RSSI_dBm

圖 B:RSSI 之值

```
if(buffer[2]>=128)  
    RSSI_Value=(buffer[2]-256)/2-74;  
else  
    RSSI_Value=buffer[2]/2-74;
```