

行政院國家科學委員會補助
大專學生研究計畫研究成果報告

* ***** *
* 計 畫 *
* : 智慧型風力發電監控系統 *
* 名 稱 *
* ***** *

執行計畫學生： 陳柏凱
學生計畫編號： NSC 101-2815-C-147-005-E
研 究 期 間： 101 年 07 月 01 日至 102 年 02 月 28 日止，計 8 個月
指 導 教 授： 林偉川

處理方式： 本計畫可公開查詢

執 行 單 位： 德明財經科技大學資訊科技系

中華民國 102 年 04 月 23 日

國科會大專生參與專題研討計畫繳交報告

計畫編號：101-2815-C-147 -005-E

計畫名稱：智慧型風力發電監控系統

學校：德明財經科技大學資訊科技系

指導老師：林偉川

學生：陳柏凱

摘要

大家所認知的綠色能源不外乎就是太陽能跟風能，現今的市面上的風力發電機大部分都無法即時遠端監控風速發電情況跟蓄電量等等，都需要人員到風力發電機上去收集數據，但如果遇到天氣不佳的時候就顯得相當不方便又危險，因此本研究開發風力發電機組及蓄電模組外，另開發一套相容於智慧型手機系統的應用程式，可以透過智慧型手機即時遠端監控風力發電機的風速風向及蓄電量等以利於研究人員蒐集監控資料。若風力發電裝置發生問題也可透過警報系統即時回報，管理人員也可透過無線網路攝影機，來操作及監控風力發電模組，兼顧了研究人員的操作方便和安全。

關鍵字：風力發電機、智慧型手機、無線網路攝影機、遠端監控、警報系統。

1. 前言

地球暖化日益加重的時代，同時我們開發許多能源興建核能廠火力發電..等。生活經濟一切都仰賴電力的幫助，然而因為聖嬰現象與暖化加劇，天災造成的家園及國家損失越來越嚴重。2011 年 3 月 11 日，日本本州島東北海岸發生 9.0 地震，福島第一核電廠原本設計只能抵擋 7.0 地震，也沒有考慮到會遭遇到這麼大的海嘯衝擊，於是福島第一核電廠因地震及海嘯所造成的核輻射外洩，日本發生如此慘痛的核災與天災，使世界各國汲取教訓，台灣的環保團體也各個跳出來向政府陳情拒絕以核能發電發展低碳能源。環保團體認為，開發再生能源(綠色能源)對抗地球暖化，例如太陽能發電以及風力發電，才是兩全其美的辦法。

現今的綠色能源-風力發電機必須至戶外樓頂上觀測風速計，無法即時掌控風向及風速，若遇上颱風及大雨，除了觀測變得困難外，也可能會對觀測人員造成危險，針對這些問題本論文利用智慧型手機系統開發一個方便即時透過無線網路攝影機監視，及遠端調整風力發電機迎風角度的應用程式，以提升觀測人員的安全性[2][3]。本論文第二節介紹相關技術及研究；第三節列出系統架構設計；第四節為系統實作及測試；第五節為結論以及未來發展方向。

2. 相關研究

2.1 風力發電機研討

風力機主要是藉著空氣流動(即風)轉動葉片以擷取風的動能，進而轉換成有用的機械能或電能。葉片轉子受風吹而轉動，係源於氣動力的作用(包括升力及阻力)對葉片產生轉動扭矩。一般而言，風力機無法利用全部之風能。理論上，葉片氣動轉換之極限效率為59.3%。大多數葉片的轉換效率約在30%~45%之間，經由機電轉換之後，風力機的輸出效率約在20%~40%之間。風力機性能較佳者首推兩葉片及三葉片水平軸高轉速翼型風力機[7]。

2.2 Arduino 控制板

Arduino 是一種微處理器控制板(microcontroller board)，約一個手掌大。使用者可以在 Arduino 上接上各種電子裝置，例如馬達、LED、喇叭、溫濕度感測器、WiFi、藍芽裝置等等。Arduino 採用開放原始碼的軟硬體平台，構建開放原始碼簡單 I/O 介面版，並具有使用類似 Java、C 語言的 Processing/Wiring 開發環境如圖 2.1。Arduino 也可以獨立運作成為一個可以跟軟體溝通的介面，例如與 Adobe Flash、Processing、Max/MSP、Pure Data 等軟體互動，所以用途非常廣泛[11][12]。

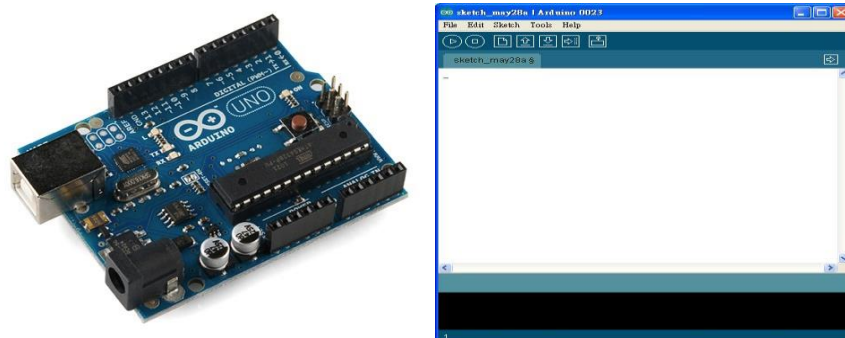


圖 2.1 Arduino 裝置及開發環境

2.3 Wifly RN-131 模組

RN-131 模組是一個獨立的嵌入式 802.11 b/g 無線網路模組如圖 2.2。本研究利用它來提供 Arduino 網路的功能。憑藉其小巧的外型和極低的功耗，RN-131 是很好的行動無線網路。Wifly 模組採用 2.4GHz 的頻道及 TCP/IP 協定。預先裝載軟體簡化了整合模組應用程式的開發的過程，模組只要求四個接腳(Power、TX、RX、GND)來建立一個無線連線，並用一個簡單的 ASCII 命令語言編譯及控制。一旦模組被設定，即可搜尋找到目前所在區域所有的 Wi-Fi 無線網路。



圖 2.2 RN-131 模組

2.4 Android

Android 是基於 Linux 核心的軟體平台和作業系統，早期由 Google，後由開放手機聯盟開發。它採用了軟體堆層 (software stack，又名以軟體疊層) 的架構，主要分為三部分。低層以 Linux 核心工作為基礎，只提供基本功能，其他的應用軟體則由名公司自行開發。Android 是一個免費跨平台的整合開放環境，只要電腦能連上網路，都能隨時下載相關開發工具，開發 Android 應用程式。目前 Android 手機的最新版本為 2.3 版，安裝 Android 的開發環境時，必須包含 JAVA JDK、Eclipse IDE(多用途開發平台)、ADT(Eclipse 的 Android 開發擴充套件(Android Development Tools plugin))、Android SDK(Android 程式開發套件，包含 Android 模擬器(Emulaor))。Android 是以物件導向的概念來撰寫，以 Java 為主要的開發語言，但 Android 並不使用常見的 Java 虛擬器(JVM)，而是使用 Dalvik 虛擬器 [1][4]。

2.5 風速計、風向計研討

風速簡單來說就是空氣因流通而產生風，而風流動的速度就是風速。風速計(風速計是一種測量空氣流速的儀器)，當風吹動風杯(也就是風速計的感應部分)時，使風杯因風吹動而旋轉，帶動儀器內的小型發電機，產生電流，然後再帶動指針而得知風速，儀器在將得知的風速透過 ADC 轉換器轉換成數位訊號傳送給伺服器記錄[5][8]。

2.6 發電馬達(交流發電)

本論文所使用的發電機馬達規格如表 2.1，其實體圖如下圖 2.3。

表 2.1 發電機馬達規格

項 目	規 格
額定線間電壓	18VAC
相數	3 相
額定轉速	3000r/min
旋轉方向	CW/CCW
相對濕度	30%-90%
工作溫度範圍	-10°C- +60°C
電機姿勢	軸伸端全方向
額定轉速	3000r/min
額定電流	≥ 500 mA
輸出直流電壓	18.0 \pm 1.0VDC
輸出交流電壓	18 \pm 1.0VAC
線間電壓差值	0.5V AC
線間電阻	10.0 \pm 1.0 Ω
線間電阻差值	5-35 Ω
絕緣電阻	1M Ω (DC100 兆歐表)



圖 2.3 發電馬達圖

2.7 繼電器

繼電器(Relay)，也稱電驛(如圖 2.4、2.5)，是一種電子控制器件，它具有控制系統（又稱輸入迴路）和被控制系統（又稱輸出迴路），通常應用於自動控制電路中，它實際上是用較小的電流去控制較大電流的一種「自動開關」。故在電路中起著自動調節、安全保護、轉換電路等作用。

我們的作法是當風速計偵測到瞬間陣風達到 15m/s(強風)時，Arduino 就會啟動短路剎車，讓風扇減速。

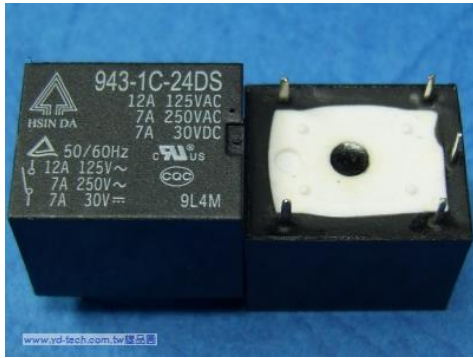


圖 2.4 繼電器

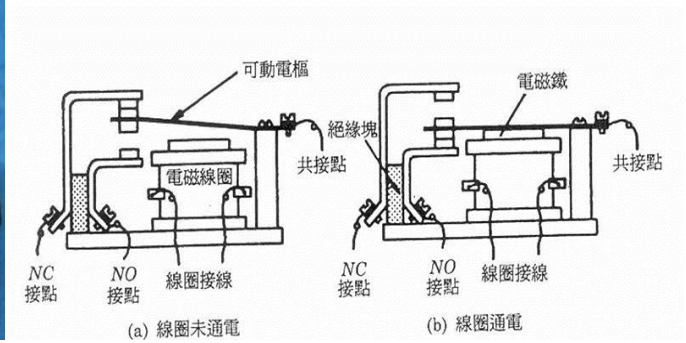


圖 2.5 繼電器內部電路圖

2.8 伺服馬達

此馬達為一個連續旋轉的伺服馬達，規格如表 2.2，實體圖為 2.6:

表 2.2 馬達規格

項目	規格
馬達型號	S35 STD
重量	42 公克
速度(sec/60°)	0.15sec
扭力	2.5cm/kg



圖 2.6 伺服馬達圖

2.9 電子羅盤

電子羅盤 CMPS10 模組是一個傾斜補償羅盤。採用 3 軸磁強計和一個 3 軸加速計和一個強大的 16 位處理器，CMPS10 生產 0-3599 代表 0-359.9 或 0 到 255。測量 X，Y 和 Z 份量的磁場，俯仰和橫滾三個傳感器的輸出被用來計算軸承。CMPS10 模塊需要 3.3 電源- 5V 和吸取的電流名義 25 毫安。有三種方式獲得從模塊軸承。其實體如下圖 2.7。

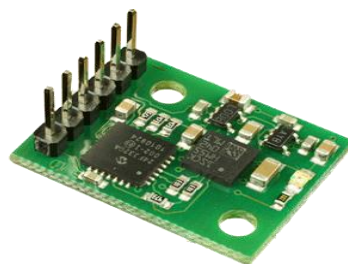


圖 2.7 電子羅盤 CMPS10 模組

3. 系統架構

風力發電機監控系統中包含：風速計監控系統、遠端監視系統、遠端操控系統、警報系統。風力發電機觀測系統作用時，風速計會偵測當時的風速回傳給伺服器，伺服器會依據風速計傳回各個方位的風速來判斷風力發電機是否要轉向來取得更高的轉換效率，如風力發電機需要轉向，伺服器會遙控風力發電機的馬達來轉向。利用 $P=IV$ (功率公式)及 $n=P_{min}/P_{max}$ (效率公式)來計算目前發電的效率顯示在遠端監控系統中。遠端監控者就可以透過 Android 系統來即時監控風力發電機的轉換效率及伺服器的運作是否有問題，如有問題更可以轉由手動調整來確保產品發揮最大功效，當風力發電機發生異常狀況，監視系統會透過伺服器的警報系統（Facebook 機器人）發佈訊息通知使用者系統發生狀況。系統運作架構圖如圖 3.1。

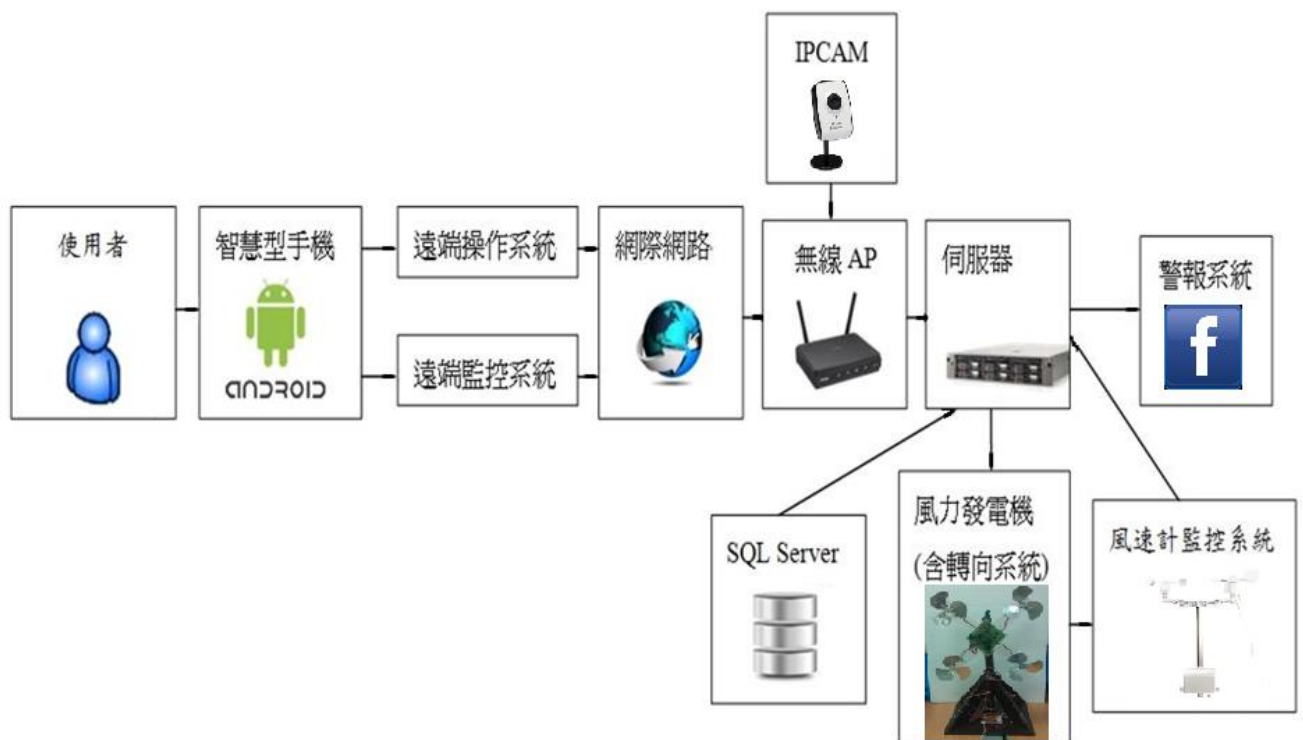


圖 3.1 系統架構圖

我們將本系統子功能主要說明如表 3.1。

表 3.1 系統子功能說明

一、遠端監視功能	透過智慧型手機監控系統程式,就能在搭載 Android 系統的行動裝置上透過 WiFi 網路即時了解風力發電機的狀態。
二、遠端操控功能	透過智慧型手機監控系統程式可以隨時藉由網路控制風力發電機的方向。
三、警報系統	當風力發電機發生異常狀況,會第一時間使用 Facebook 機器人通知使用者系統異常。
四、風速計監控系統	風速計將風速風向等數據傳送給 Arduino,讓監控端可以得知目前的數據。
五、風力發電機轉向系統	當伺服器或監控者要求風力發電機轉向時,轉向系統將會控制馬達把發電機轉向到正確的方位。
六、短路剎車系統	當風速計監測到風速過大的時候,Arduino 就會發送訊號去控制繼電器,讓馬達短路剎車,以免馬達因風速過大燒毀。

3.1 遠端監控系統

遠端監控系統是使用 Android 智慧型手機作業系統的 VideoView 來做呈現,以 RTSP 協定把影像設定成 3gpp 格式直接在手機畫面做 IPCAM 監視器畫面的播放。當警報系統通知使用者系統異常時可即時連線做觀看的動作,以便做最快的異常處理[10]。

3.2 遠端操控系統

Android 智慧型手機其傳輸方法為透過 Socket 傳輸方式連線到伺服器,使用手機觀看風力即時監控系統現況及調整風力發電機的角度。當使用者在有無線網路的地方就可以直接藉由 Android 智慧型手機直接觀看 IPCAM 監視畫面,並調整風力發電機之角度。當警報系統發出警報傳訊息給使用者時,可以即時連線觀看風力即時監控系統的現況,並且馬上做處理。當覺得風力發電機的功率不足時,使用者也可以自行調整風力發電機的角度以提高功率[14]。

3.2.1 Socket 傳輸方式

遠端操控系統主要是透過 Socket 傳輸方式連線到伺服器端,再由伺服器端透過網路經由 Wifly 傳送指令到 Arduino 控制板去更改風力發電機的角度。伺服器接收狀況。伺服器接收狀況如圖 3.2[13]。

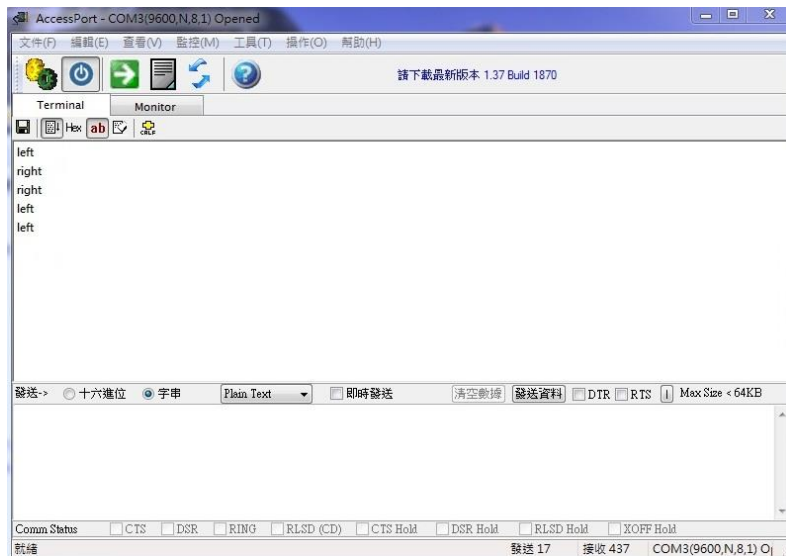


圖 3.2 由 Server 端傳送訊息給 Arduino 控制板畫面

3.3 警報系統

警報系統(Facebook[簡稱 FB]機器人)登入後會開始進入偵測狀態，每 20 秒偵測一次，當有任何異常狀況發生，接收到事件代碼，會透過 SQL Server 去尋找事件代碼所代表的事件來發佈到使用者的 FB 上。而沒接收到事件代碼則是以 20 秒偵測一次。系統實測 10 分鐘每 20 秒發佈一次預計發 30 次，其成功接收率約為 95%。FB 機器人流程圖如圖 3.3 和發佈圖 3.4 和事件代碼圖 3.5[6]。

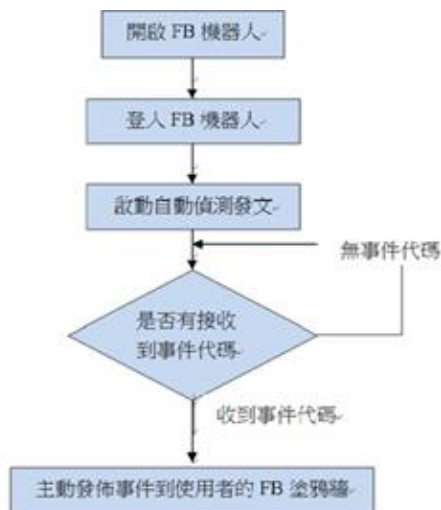


圖 3.3 FB 機器人流程圖



圖 3.4 FB 機器人發佈至塗鴉牆訊息

資料表 - dbo.table1 摘要	
eventsid	printer
1	電源低於40%
2	電源低於60%
3	電源低於80%
4	風力過小發電量不足
5	風力過大啟動防護機制
6	風力發電機沒有轉至該風位
7	風扇異常

圖 3.5 事件代碼對照表

3.4 風速計監控系統

風速計就是當風吹動風杯時，使風杯因風吹動而旋轉，讓內部電路裡的開關閉一次，依照開關的時間，就可以計算出風速了，一次開關的時間換算是 0.67m/sec。風向計裡面有 16 個位置的電阻器，葉片轉動時進行分壓，分壓後的電壓結果就代表不同的方位，其風向換算如下表 3.2，實體圖如 3.6。所以當風力發電機在發電時，系統上的風向計會持續偵測哪個方位的風速較強，將取得的數據透過 FB 機器人傳送給遠端監控的人，讓觀測者可以依取得的數據來調整風力發電機的方位 [5][8]。

表 3.2 風向換算表

Direction (Degrees)	Resistance (Ohms)	Voltage (V=5v, R=10k)
0	33k	3.84v
22.5	6.57k	1.98v
45	8.2k	2.25v
67.5	891	0.41v
90	1k	0.45v
112.5	688	0.32v
135	2.2k	0.90v
157.5	1.41k	0.62v
180	3.9k	1.40v
202.5	3.14k	1.19v
225	16k	3.08v
247.5	14.12k	2.93v
270	120k	4.62v
292.5	42.12k	4.04v
315	64.9k	4.78v
337.5	21.88k	3.43v



圖 3.6 風速計、風向計

3.5 風力發電機轉向系統

風力發電機底部設置了一個伺服馬達，當風速計監控系統，偵測到哪個風向的風速最大時，會傳訊號給 Arduino，讓 Arduino 控制伺服馬達，讓風力發電機轉至最佳方位。

4. 系統實測

4.1 系統電路說明

當交流馬達風扇發電機，轉動產生電能，先進入整流 IC 把交流電轉換成直流電，再進入升壓電路把產生的低電壓提高到 13.75V，因為要防止蓄電池過度充電，所以在蓄電池前有個防過充二極體電路，保護蓄電池。如圖 4.1。因為升壓電路要 3V 左右才可以升至 13.75V，所以我們為了要讓風力發電機能在更小的風速下可以對電池充電，因此我們利用克希荷夫電壓定律的原理將馬達兩兩串聯整流後讓升壓電路提早將電壓升至 13.75V，升至 13.75V 後我們利用克希荷夫電流定律將電流輸出並聯，提升電流量以縮短充電的時間。風速計和風向標的 RJ-11 接頭透過配線盒把 4 個腳位分開連接到 Arduino，其中第 1 腳和第 4 腳是風向標，第 2 腳和第 3 腳是風速計，Arduino 在透過風向標及風速計所提供的數據去判斷，當沒有風時發電機是不會追風的，若風速足以對電池充電時 Arduino 就會控制伺服馬達，轉動風力發電機，當風力發電機轉到差不多的方位時，電子羅盤會再加以校正讓風力發電機的角度誤差在 ± 23 度以內。若風向變化在 ± 23 度以內時，轉向系統則不會啟動以節省轉向系統的耗功。

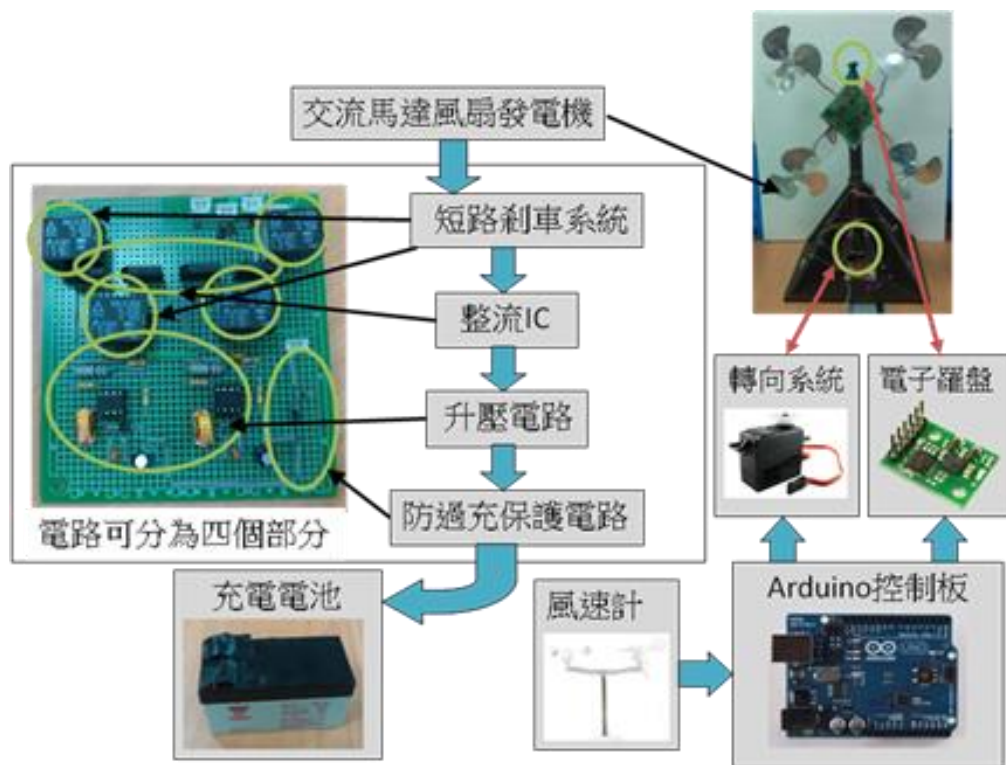


圖 4.1 系統電路結構圖

4.2 整流 IC 及升壓電路及防過充二極體電路

整流 IC 和升壓電路和防過充二極體電路和繼電器的實作整體電路圖，電路圖及控制板實體圖如圖 4.2 及圖 4.3 。

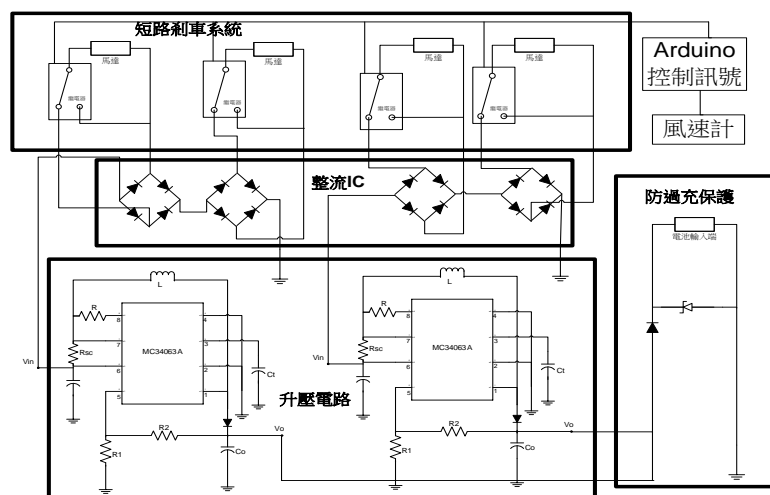


圖 4.2 整流 IC 和升壓電路和防過充二極體和繼電器電路圖



圖 4.3 整流 IC 和升壓電路和防過充二極體和繼電器控制板實體圖

4.2.1 整流 IC

本系統的整流 IC 功能是把交流電轉成直流電，控制板接法如下圖 4.4 及 IC 實體和構造圖 4.5 及 4.6。此 IC 工作原理為利用電橋原理整流如表 4.1。

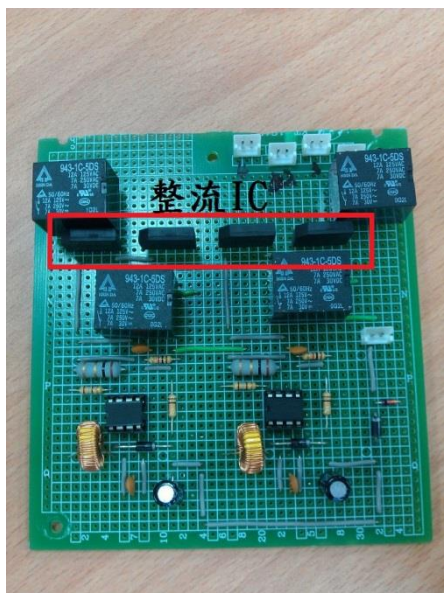


圖 4.4 整流 IC 實體圖



圖 4.5 整流 IC 實體圖

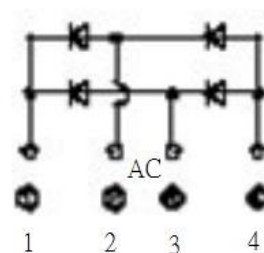


圖 4.6 整流 IC 構造圖

表 4.1 IC 工作原理

IC 接腳與內部構造圖
接腳 1:正極直流電輸出
接腳 2:交流電輸入端
接腳 3:交流電輸入端
接腳 4:負極直流電輸出
IC 特性
1、頻率:中頻
2、小功率
3、交流最大輸入電壓 600V
4、直流最大輸出電壓 600V
5、直流輸出電流 2A

4.2.2 升壓電路

本系統的升壓電路功能是把馬達產生的低電壓升壓約至 13.75V，控制板接法如下圖 4.7 及控制板實體圖 4.8。

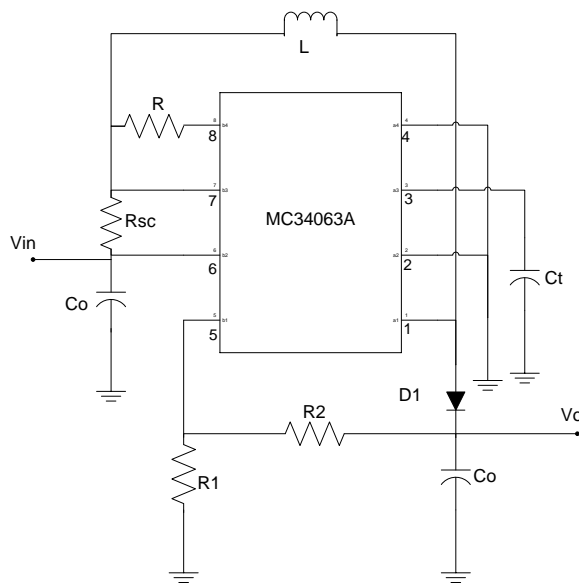


圖 4.7 升壓電路控制板電路圖

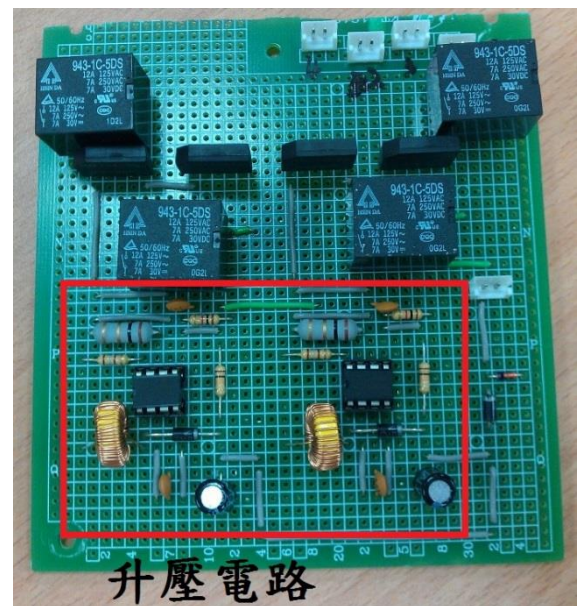


圖 4.8 升壓電路控制板實體圖

零件使用的規格:

$R1=1K\Omega$ 、 $R2=10K\Omega$ 、 $Rsc=1\Omega$ (高功率電阻，以防燒掉)作用是限流

$R=180\Omega$ 、二極體是 1N5819

$Co= 220\mu F$ 16V (把輸出的雜訊濾掉)、 $Ct=300pF$ 、 $L=150\mu F$

4.2.3 防過充二極體電路

本系統的防過充二極體電路功能是二極體防逆向電流，Zener 二極體是避免過充，控制板接法如下圖 4.9 及控制板實體圖 4.10。

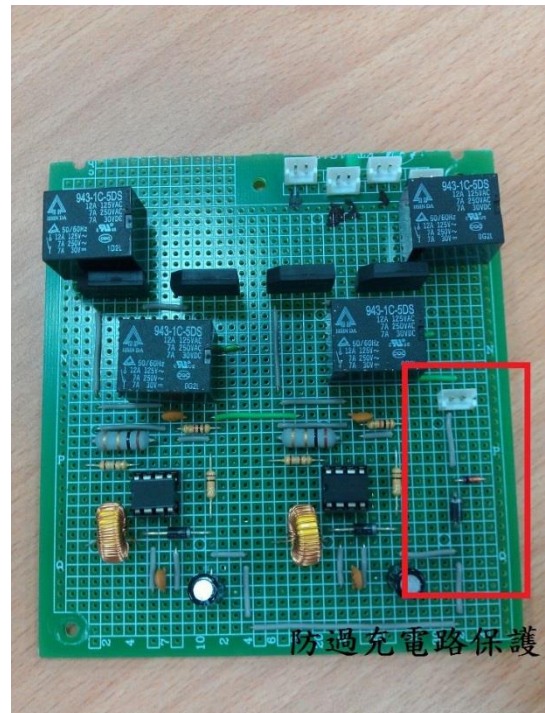
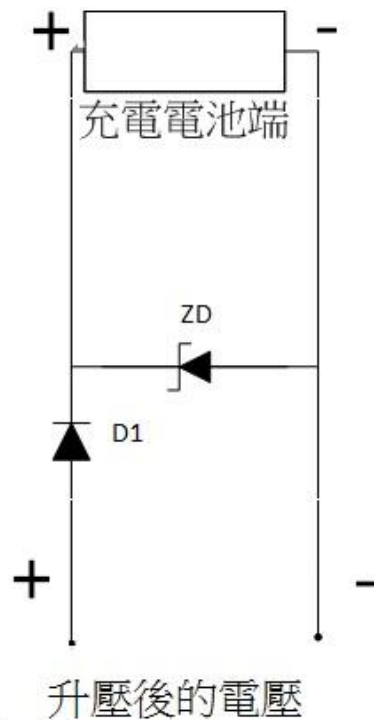


圖 4.9 防過充二極體電路控制板電路圖 圖 4.10 防過充二極體電路控制板實體圖

零件使用的規格: [9]

ZD:13V/1W

D1:1N4001

並聯在電池上，限制電壓不超過 13V 左右。

4.3 風力發電機實作畫面

當有風時，Arduino 會由風向計得知風向從哪裡吹來，Arduino 就會控制伺服馬達調整方位讓風力發電機達到最好的效能。風力發電機實作圖 4.11、4.12。

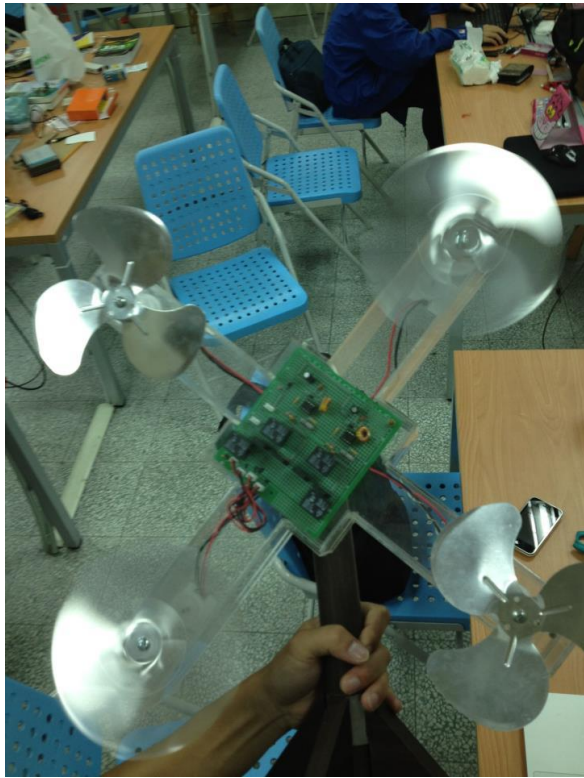


圖 4.11 風力發電機室內實作



圖 4.12 風力發電機室外實作

4.4 風力發電機整體實體圖

我們將連接到 Arduino 的風向標、電子羅盤、伺服馬達、短路剎車所需要的電阻及其他零件跟提供電池+5V 輸出的 7805 這顆 IC 整合到一片電路板如圖 4.13，電路的接線圖如圖 4.15。風速計和風向標的部分，因為套件買來的時候沒有基座，因此我們就設計了穩定的基座，這樣就可以將風速計及風向標站立起來以便系統的作動如圖 4.14[9]。

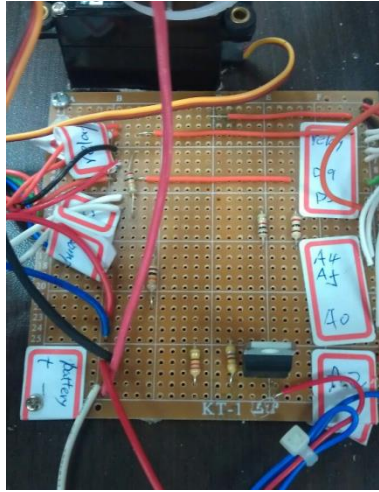


圖 4.13 電路實體圖

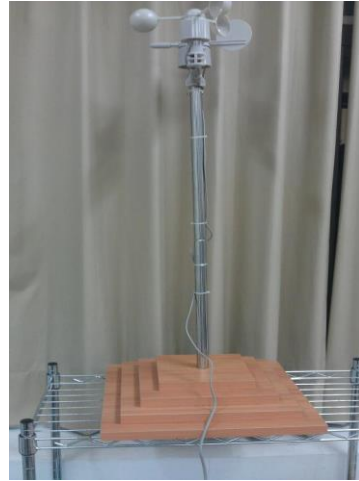


圖 4.14 風力發電機底座

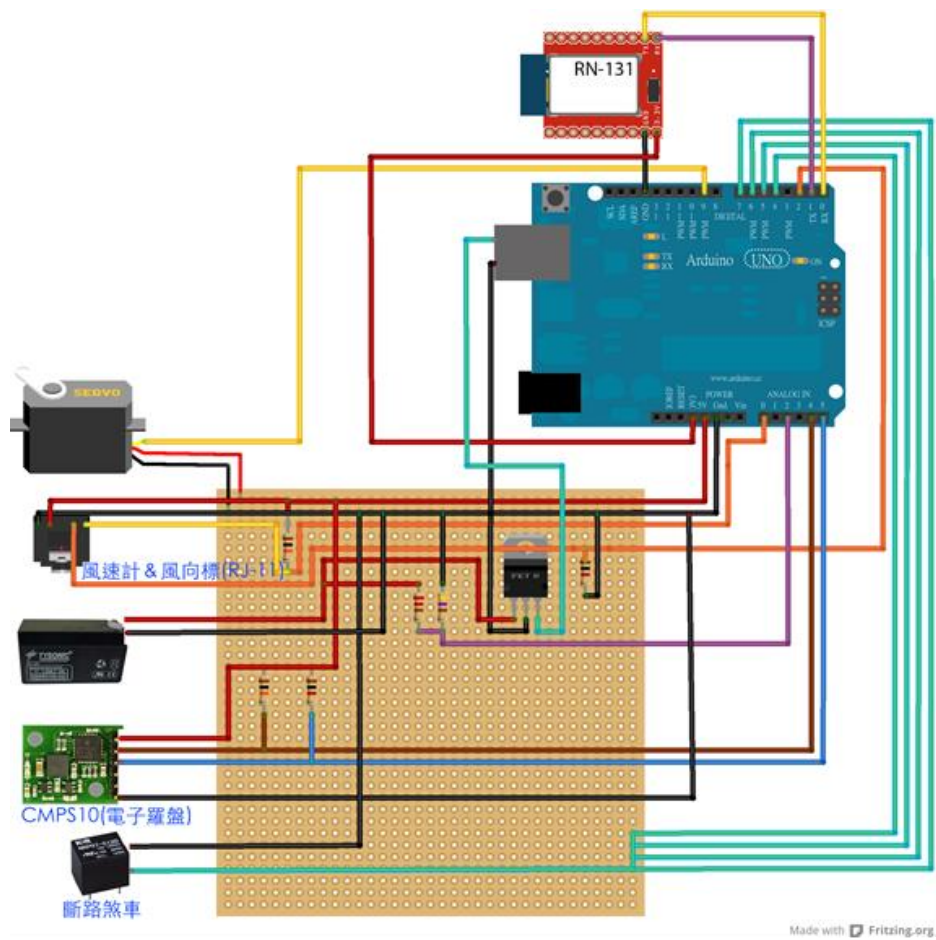


圖 4.15 中心電路示意圖

4.5 電腦及 Android 手機遠端監控實作畫面

使用者可以隨時利用電腦或手機透過網路連到 IPCAM，其解析度如表 4.2。只要有電腦的地方就可以利用網頁查看系統的狀況。我們也在 Android 智慧型手機上開發應用程式，只要使用此應用程式就可以使用智慧型手機隨時隨地監控此系統。若風力發電機發生問題時，使用者也可以使用 Android 手機，透過 Socket 傳遞指令給 Arduino 做角度調整的動作以增加系統的靈活性。遠端監控實作圖 4.17、4.18。

表 4.2 解析度

	手機畫面	電腦畫面
解析度	160X120	640X480
FPS	5/s	30/s
格式	3GP	MPEG4



圖 4.17 遠端手機操控實作畫面

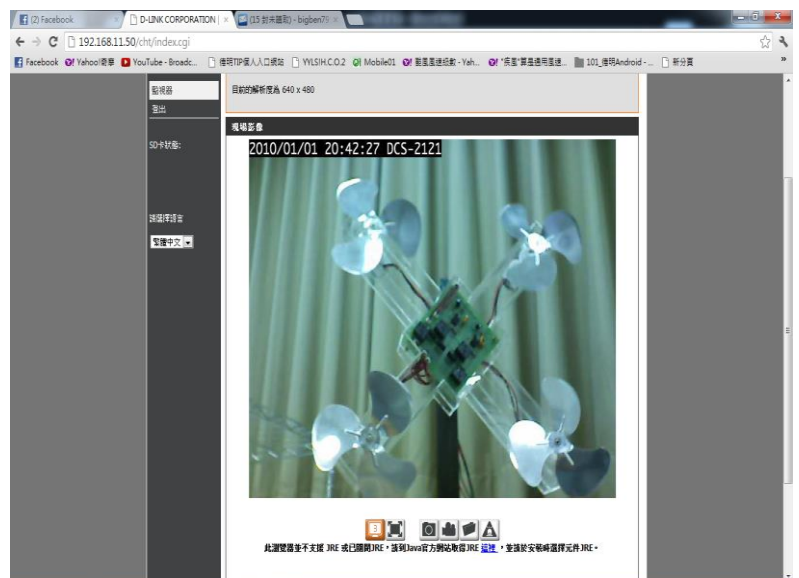


圖 4.18 網路遠端監視實作畫面

4.6 警報系統實作畫面

FB 機器人系統登入後，系統一直處於等待狀態，當本裝置產生異狀時，會觸發 FB 機器人自動發送訊息給使用者。警報系統實作畫面。系統實測 10 分鐘每 20 秒發一次預計發 30 次，其成功接收率約為 95%。警報系統實作如圖 4.19。



圖 4.19 FB 機器人實作畫面

4.7 測試數據

依功率公式 $W=Pt=IVt$ ，假設輸出電壓為 13.75V，蓄電池如圖 4.20 其功率是 $12V \times 1.2A = 14.4W$ ，要求充電時間，理論值為 $14.4W = 75mA \times 13.75V \times t$ ， $t \div 14$ ， $14.4W = 155mA \times 13.75V \times t$ ， $t \div 7$ ，得到了充電時間 t 最久約為 14 個小時，最快約為 7 小時，電流的大小取決於風的大小。以下是室內電風扇系統實測數據如表 4.3。

表 4.3 室內電扇系統實測數據表

電扇風量	理論值	測量值	誤差
弱風(4m/sec)	41mA	38mA~40mA	2%
中風(6m/sec)	57mA	51mA~55mA	3%
強風(7m/sec)	68mA	62mA~66mA	2%

室外實測如圖 4.21，其數據我們測量的結果約為： $14.4W=120mA*13.6V*t$ ， $t\approx 12.5$ ，大概要充電 12.5 小時，才可把電池充飽。

- 若每小時風力發電機系統所得到的功率為
 $13.75V*109mA*59*60=4.8675\text{ kw/h}$
- 調整角度一次時間約耗費 1 分鐘耗功→ $2.88*60=172.8\text{ w}$
- 所以每小時所得到的能量為 → $4867.5-172.8=4.6947\text{ kw/h}$



圖 4.20 蓄電池



圖 4.21 室外測試

5. 結論及未來展望

風速計能即時監測風的大小透過控制板計算風向當變化超過門檻值時，透過轉向系統、電子羅盤將風力發電機轉向風量較大的方位。風力發電機所發的電力可以充電至電池中儲存電力，並自動偵測電池電量情形電池。電池若充飽時，也有保護裝置忽略繼續所充的電力。使用者可透過 Android 手機即時監控風力發電的情形，當發生異常狀況，例如：風速過大、風力發電機異常時，系統自訂啟動 MSN 機器人即時回報，使用者也可透過 Android 手機遠端操控風力發電轉向系統以減少人力資源耗損。

未來可朝向把風力發電機的機身縮小，以提供耗電量的減少，進而提升發電效率，並提供更多的機體保護措施，讓機體能有更完善的保護。

投稿

研討會名稱	主辦單位	論文名稱	研討會日期
中華民國第 33 屆 電力工程研討會	國立臺北科技大學	智慧型風力發電 監控系統	101 年 12 月 7-8 日

比賽

研討會名稱	主辦單位	論文名稱	比賽日期	名次
龍華盃全國綠能 創意與應用專題 製作競賽	龍華科技大學 電機工程系	智慧型風力發 電監控系統	101 年 10 月 26 日	第一名

參考文獻

1. 陳偉華等人，天羅地網系統，德明財經科技大學資訊科技系畢業專題，2011
2. 風力專題，國立高雄應用科技大學，
<http://wais.ee.kuas.edu.tw/energyworld/powerworld/main/chap04-topics/1.reports/EA/wind/3.htm>
3. 莊志偉等人，數位化風向計與風速計之研究，高雄市立中正高工電子科科展
專題，2000，
<http://www.khjh.kh.edu.tw/science40/%E9%AB%98%E4%B8%AD/%E9%AB%98%E4%B8%AD%E6%87%89%E7%94%A82/%E9%AB%98%E4%B8%AD%E6%87%89%E7%94%A82.htm>
4. Android 平台的詳細介紹，2009，
http://www.sogi.com.tw/newforum/article_list.aspx?topic_ID=6072766
5. 風速儀的原理及選型，珠海天創儀器，
<http://www.64817.com/zh-tw/htmlmodel/jswzxx/284529.html>
6. 如何使用 Facebook C# SDK 發佈訊息到塗鴉牆
<http://coding.anyun.idv.tw/2011/04/05/how-to-use-facebook-csharp-sdk-publish-stream-to-wall/>
7. 能源資訊網，
emis.erl.itri.org.tw/_upload/_admin/sepro/27/file/風力發電.doc
8. 風向和風速，http://content.edu.tw/senior/earth/yl_ld/content/6-5/a03.htm
9. 自己 DIY 小風力發電器，2009，

http://club.fansio.com/AllForum/forum_more.asp?ClassID=21&id=1138

10. Android VideoView 如何播放 RTSP，2011，

<http://www.shouyanwang.org/thread-96-1-1.html>

11. 什麼是 Arduino，Arduino.tw 樂園，<http://arduino.tw/index.php>

12. Arduino 介紹，<http://arduino.cc/en/>

13. 黃彬華，Android 2.X 手機程式開發教戰手冊，碁峰出版社，2011

14. [C#]使用 Facebook SDK 開發 Facebook API，

<http://www.dotblogs.com.tw/jimmyyu/archive/2009/11/10/11516.aspx>