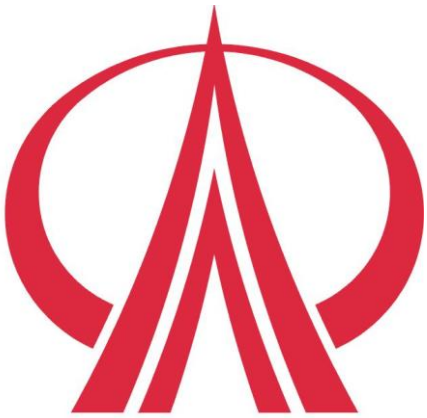
修 平 科 技 大 學
電 機 工 程 系

DEPARTMENT OF ELECTRICAL ENGINEERING
HSIUPING UNIVERSITY OF SCIENCE AND TECHNOLOGY

實務專題報告書

室外智慧農業檢測系統



指 導 老 師： 程兆龍

專題製作學生：

四技電三乙 林建旻 BD104032

四技電三乙 童士綱 BD104072

中 華 民 國 107 年 6 月 25 日

修平科技大學

電機工程系

HSIU-PING UNIVERSITY OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF ELECTRICAL ENGINEERING

指導老師:程兆龍

專題學生:童士綱、林建旻

製作日期:107 年 6 月 25 日

摘要

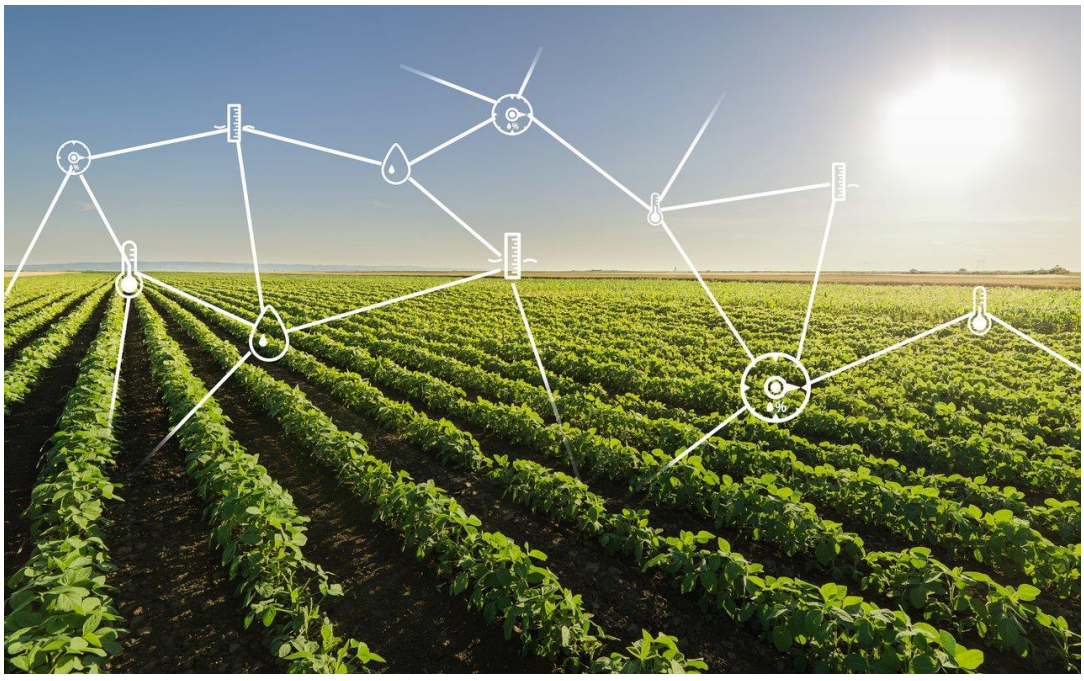
台灣農業由於農村勞動人力老化，農業生產方面逐漸減少，許多問題，例如天然災害環境影響、人力成本高、技術無法傳承，使得現代的年輕人寧可到都市或外地發展，也不願繼續待在家鄉。因此政府推行農業政策，提升農業競爭力。

目錄

第一章 序言

1.1 研究動機

當一個國家沒有基本的農業作支撐，只依靠外國進口糧食的話，可能會產生糧食不足的問題，同時台灣農產品在國外市場之競爭能力，日趨下降，農家所得顯著低於非農家所得， 年輕人不願從事農業非常明顯，近年來政府大力推動精緻農業，導入物聯網、雲端大數據等智慧科技，提升農產品的產量與品質，農業政策必須將過去以提升農業生產力為核心，轉變為將農產品商品化，而以食品加工業帶動農業工業化、企業化經營。智慧農業生產科技範疇中，農產品成長過程中的環境監控技術，是熱門的研究議題。隨著現代科技的快速發展，農業更加重視植物生理資訊的採集和環境的監控。作為影響農作物生長的重要因素，土壤及溫度的濕度控制是必不可少。

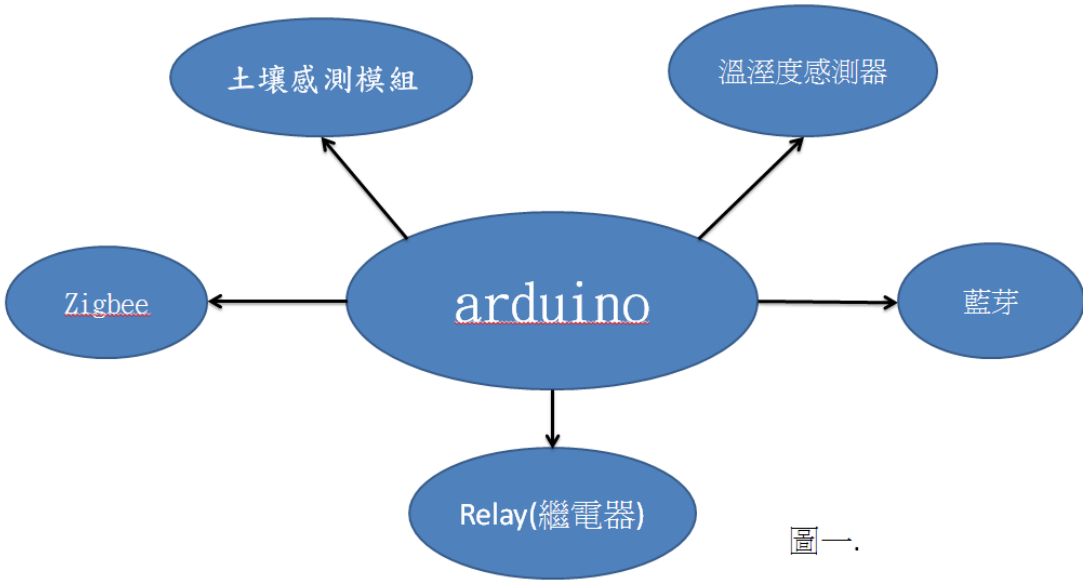


1.2 目的

可透過 PLC/繼電器推動，控制的馬達，以及灑水器進行環境監控提供一個農作物最佳的成長環境。因此發展智慧農業，就必須建置有一個高效、準確的環境監控系統。

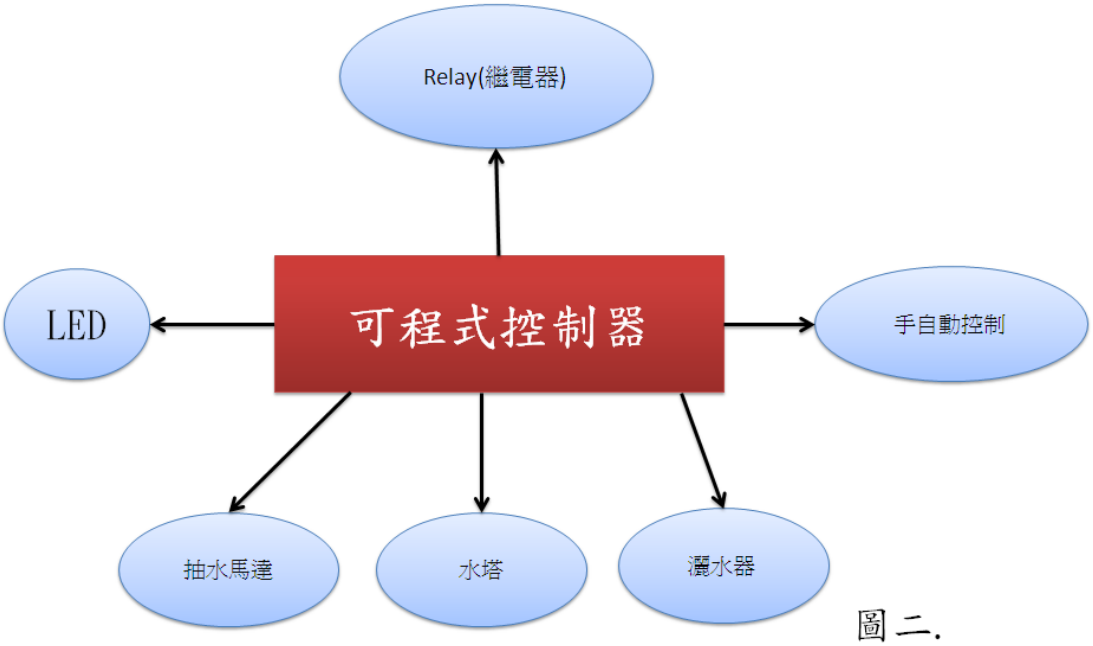
1.3 作品功能與架構圖說明

1. 本專題使用嵌入式晶片、Zigbee 與 BLE 無線技術，選用高功能且價格便宜 ARM Codex M0 STM32 嵌入式系統晶片作為 CPU，結合光照/土壤乾濕度感測器(圖一)，協助農夫”巡田水”工作。數據資料都可以網路閘道器上傳雲端，讓農民充分掌握農田環境與土壤狀況，進一步做農田大數據分析。



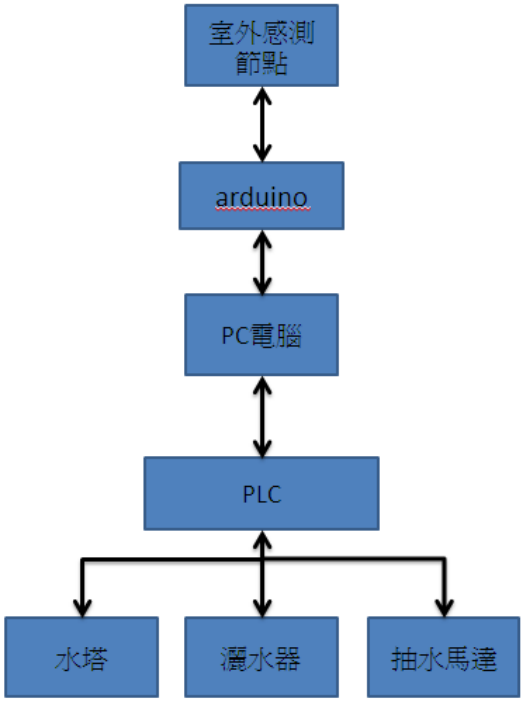
圖一.

2. 監控裝置的控制接點使用 PLC， 利用風扇模擬室外農田抽水馬達、安裝液面感測器至水塔、利用 LED 模擬灑水器。(圖二)



圖二.

3. 整體架構圖



室外農場感測節點系統架構圖

第二章 相關硬體原理介紹與說明

2.1 可程式控制器原理說明與介紹

可程式邏輯控制器（Programmable Logic Controller，簡稱 PLC），一種具有微處理器的數位電子裝置，用於自動化控制的數位邏輯控制器，可以將控制指令隨時載入記憶體內儲存與執行。可程式控制器由內部 CPU，指令及資料記憶體、輸入輸出單元、電源模組、數位類比等單元所模組化組合成。PLC 可接收（輸入）及發送（輸出）多種型態的電氣或電子訊號，並使用他們來控制或監督幾乎所有種類的機械與電氣系統。

隨著不斷的發展，這些當初功能簡單的電腦模組已經有了包括邏輯控制，時序控制、模擬控制、多機通訊等許多的功能，名稱也改為可程式控制器（Programmable Controller）。

在可程式邏輯控制器出現之前，一般要使用成百上千的繼電器以及計數器才能組成具有相同功能的自動化系統，而現在，經過編程的簡單的可程式邏輯控制器模組基本上已經代替了這些大型裝置。可程式邏輯控制器的系統程式一般在出廠前已經初始化完畢，用戶可以根據自己的需要自行編輯相應的用戶程式來滿足不同的自動化生產要求。

現在工業上使用可程式邏輯控制器已經相當接近於一台輕巧型電腦所構成，甚至已經出現整合個人電腦（採用嵌入式作業系統）與 PLC 結合架構的可

程式自動化控制器（Programmable Automation Controller，簡稱 PAC），能透過數位或類比輸入/輸出模組控制機器裝置、製造處理流程及其他控制模組的電子系統。可程式邏輯控制器廣泛應用於目前的工業控制領域。在工業控制領域中，PLC 控制技術的應用已成為工業界不可或缺的一員。

2.2 PLC 內部運作方式

雖然 PLC 所使用之階梯圖程式中往往使用到許多繼電器、計時器與計數器等名稱，但 PLC 內部並非實體上具有這些硬體，而是以記憶體與程式編程方式做邏輯控制編輯，並藉由輸出元件連接外部機械裝置做實體控制。因此能大大減少控制器所需之硬體空間。實際上 PLC 執行階梯圖程式的運作方式是逐行的先將程式碼以掃瞄方式讀入 CPU 中並最後執行控制運作。在整個的掃瞄過程包括三大步驟，「輸入狀態檢查」、「程式執行」、「輸出狀態更新」說明如下：

步驟一「輸入狀態檢查」：

PLC 首先檢查輸入端元件所連接之各點開關或感測器狀態（1 或 0 代表開或關），並將其狀態寫入記憶體中對應之位置 Xn。

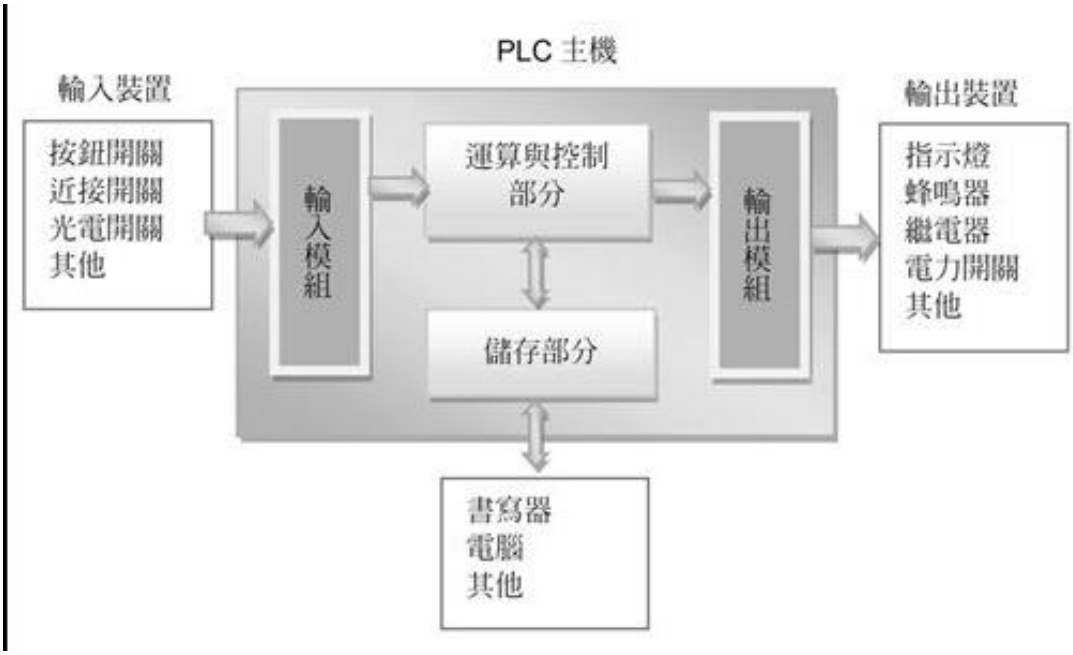
步驟二「程式執行」：

將階梯圖程式逐行取入 CPU 中運算，若程式執行中需要輸入接點狀態，CPU 直接自記憶體中查詢取出。輸出線圈之運算結果則存入記憶體中對應之位置，暫不反應至輸出端 Yn。

步驟三「輸出狀態更新」：

將步驟二中之輸出狀態更新至 PLC 輸出部接點，並且重回步驟一。

此三步驟稱為 PLC 之掃描周期，而完成所需的時間稱為 PLC 之反應時間，PLC 輸入訊號之時間若小於此反應時間，則有誤讀的可能性。每次程式執行後與下一次程式執行前，輸出與輸入狀態會被更新一次，因此稱此種運作方式為輸出輸入端「程式結束再生」。



PLC 內部架構圖

2.3 硬體架構

一般講，PLC 分為箱體式和模組式兩種。但它們的組成是相同的，對箱體式 PLC，有一塊 CPU 板、I/O 板、顯示面板、記憶體塊、電源等，當然按 CPU 效能分成若干型號，並按 I/O 點數又有若干規格。對模組式 PLC，有 CPU 模組、I/O 模組、記憶體、電源模組、底板或機架。不管哪種結構類型的 PLC，都屬於匯流排式開放型結構，其 I/O 能力可按用戶需要進行擴展與組合。PLC 的基本結構框圖如下：

1.電源模組

有些 PLC 中的電源，是與 CPU 模組合二為一的，有些是分開的，其主要用途是為 PLC 各模組的積體電路提供工作電源。同時，有的還為輸入電路提供 24V 的工作電源。電源如果為交流電源通常為 220VAC 或 110VAC，若為直流電源常用的為 24V。

2.中央處理單元

PLC 中的 CPU 是 PLC 的核心，它按 PLC 的系統程式賦予的功能接收並存貯用戶程式和資料,用掃描的方式採集由現場輸入裝置送來的狀態或資料，並存入規劃的暫存器中，同時，診斷電源和 PLC 內部電路的工作狀態和編程過程中的語法錯誤等。進入運行後，從用戶程式存貯器中逐條讀取指令，

經分析後再按指令規定的任務產生相應的控制訊號，去指揮有關的控制電路，與個人電腦一樣，主要由運算器、控制器、暫存器及實現它們之間聯繫的資料、控制及狀態匯流排構成，還有週邊晶片、匯流排介面及有關電路。它確定了進行控制的規模、工作速度、記憶體容量等。

3.記憶體

記憶體主要用於存儲程式及資料，是 PLC 不可缺少的組成單元。PLC 內部會存放撰寫完成編輯的程式指令及資料，通常也可使用 RAM 或 EEPORM 等專用記憶體卡片方式擴充，但擴充能力得依各廠牌與型號有所不同。

4.輸入/輸出單元

PLC 的對外功能，主要是通過各種輸入/輸出模組與外界聯繫的，按 I/O 點數確定模組規格及數量，I/O 模組可多可少，但其最大數受 CPU 所能管理的基本配置的能力，即受最大的底板或機架槽數限制。I/O 模組整合了 PLC 的 I/O 電路，其輸入暫存器反映輸入訊號狀態，輸出點反映輸出閂鎖狀態。

輸入單元是用來連結擷取輸入元件的訊號動作並透過內部匯流排將資料送進記憶體由 CPU 處理驅動程式指令部分。PLC 輸入模組 PLC 系統的架構和輸入模組產品的選擇端視需要被監測的輸入訊號位準而定。

來自不同類型被監測的感測器與流程控制之變量訊號，可以涵蓋從±10mV 至±10V 的輸入訊號範圍。

輸出單元是用來驅動外部負載的介面，主要原理是由 CPU 處理以書寫在 PLC 裡的程式指令，判斷驅動輸出單元在進而控制外部負載，如指示燈、電磁接觸器、繼電器、氣（油）壓閥等。

PLC 輸出模組在工業環境中用來控制制動器、氣閥及馬達等的 PLC 系統類比輸出範圍包括±5V、±10V、0V 到 5V、0V 到 10V、4 到 20mA、或 0 到 20mA 等。

5.通訊

現在 PLC 大多具有可擴充通訊網路模組的功能，簡單的 PLC 以 BUS 纜線或 RS-232 方式通訊連結，較高階的 PLC 會採用 USB 或乙太網路方式做通訊連結。它使 PLC 與 PLC 之間、PLC 與個人電腦以及其他智慧型裝置之間能夠交換資訊，形成一個統一的整體，實現分散集中控制。現在幾乎所有的 PLC 新產品都有通訊網路功能，它和電腦一樣具有 RS-232 介面，通過雙絞線、同軸電纜或光纜，可以在幾公里甚至幾十公里的範圍內交換資訊。當然，PLC 之間的通訊網路是各廠家專用的，PLC 與電腦之間的通訊，一些生產廠家採用工業標準匯流排，並向標準通訊協定靠近，這將使不同機型的 PLC 之間、PLC 與電腦之間可以方便地進行通訊與網路。

PLC 通訊協定規格可分為 RS-232、RS-422、RS-432、RS-485、IEEE 1394、IEEE-488（GPIB），其中 RS-432 最為少見。目前國際中最常用的通訊協定為 MODBUS-ASCII 模式及 MODBUS-RTU 模式，此為 Modicon 公司所制定的通訊協定。PROFIBUS 則為西門子公司所制定。日本三菱電機則推出 CC-LINK 通訊協定。

2.4 程式設計

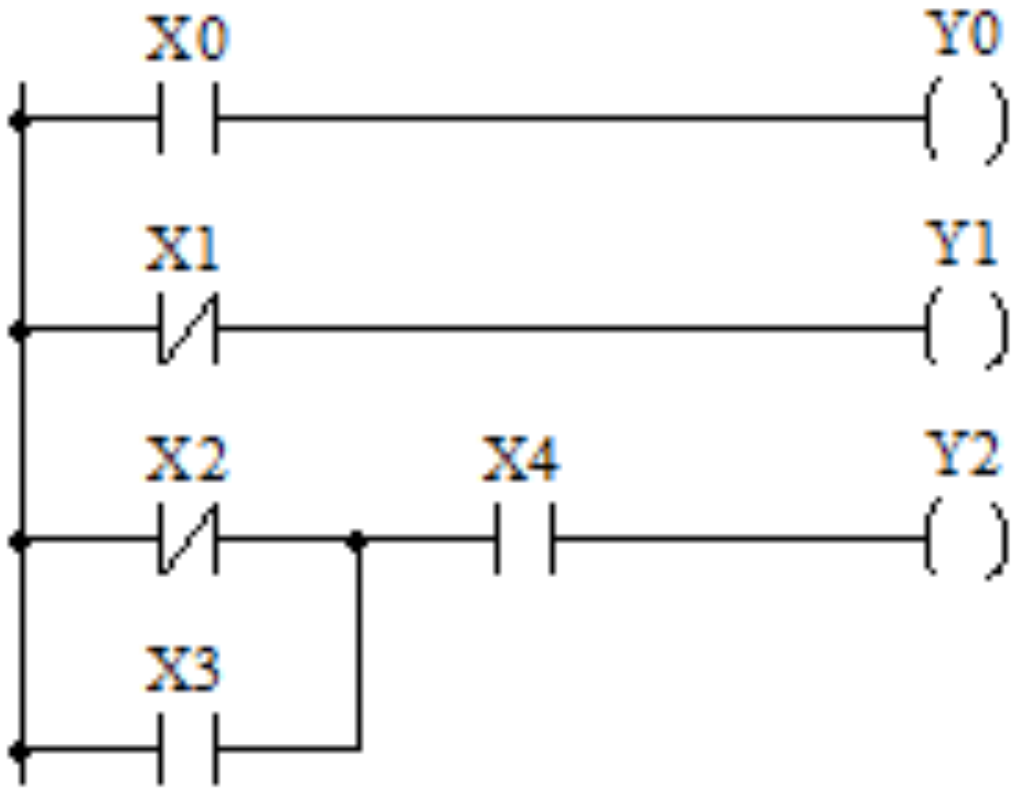
PLC 的編程程式語言與一般電腦程式語言相比，具有明顯的特點，它既不同於高階語言，也不同與一般的組合語言，它既要滿足易於編寫，又要滿足易於調試的要求。目前，還沒有一種對各廠家產品都能相容的編程語言。IEC 61131-3 是一個國際標準，它規範了 PLC 相關之軟體硬體的標準，其最終的目的是可以讓 PLC 的使用者在不更改軟體設計的狀況下可以輕易更換 PLC 硬體。IEC 61131-3 主要是提供了五種程式語言，包含：

1. 指令表（Instruction List，IL 或 Statement List，SL）：類似組合語言的描述文字。由指令語句系列構成，如 Mitsubishi FX2 的控制指令 LD、LDI、AND、ANI、OR、ORI、ANB、ORB、MMP、MMS 與 OUT 等，一般配合書寫器寫入程式，而書寫器只能輸入簡單的指令，與電腦程式中的階梯圖比較起來簡單許多。書寫器不太直觀，可讀性差，特別是遇到較複雜的程式，更難讀；但其優點就是不需要電腦就可以更改或察看 PLC 內部程式。

使用書寫器時，必須注意的是 PLC 指令中輸出有優先次序，其中若有輸出至相同的單元時（如 Y000），輸出的優先次序以位址越大優先次愈越高，一般不容易從書寫器中察覺所輸入的單元。

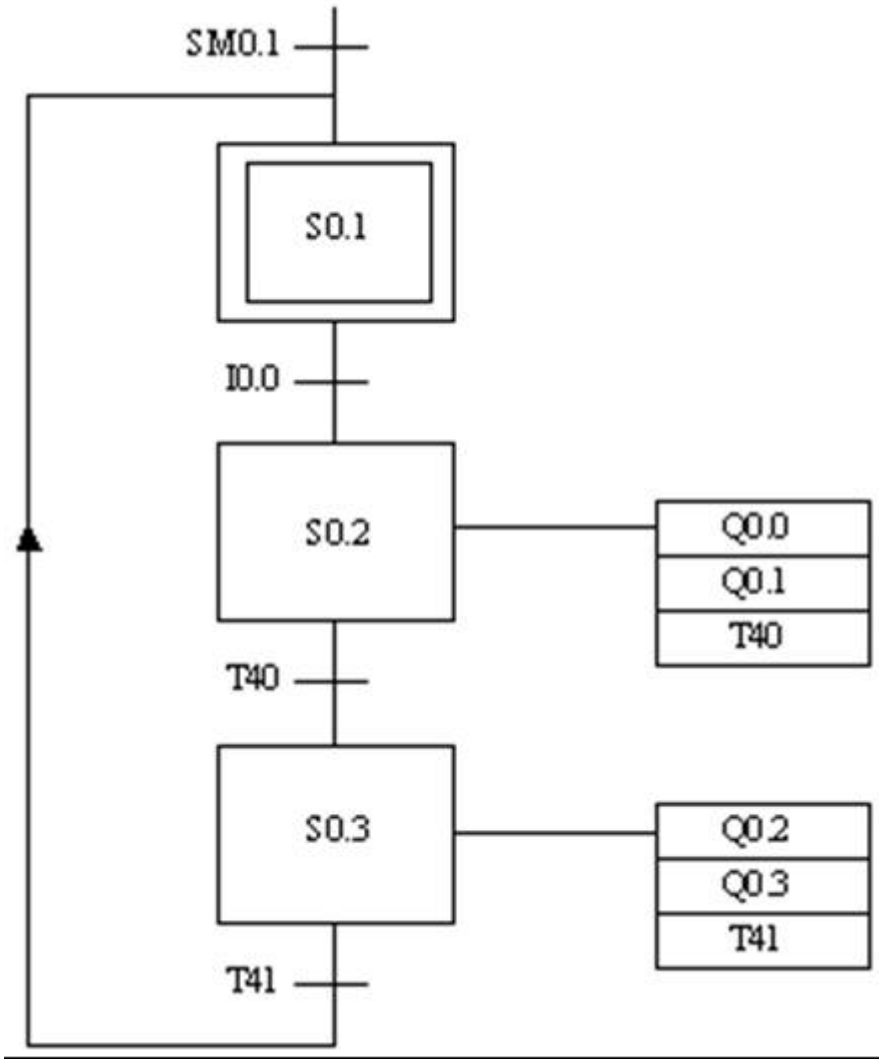
2. 結構式檔案編程語言（Structured Text，ST）：類似 PASCAL 與 C 語言的語法，適合撰寫較複雜的演算法，除錯上也比階梯圖要容易得多。ST 語言類似於程式語言的特性，因此可利用與微電腦及個人電腦相同的程式設計技術進行階梯式語言所難以執行的複雜計算，完成程式的建立。

3. 階梯圖（Ladder Programming，LAD）：類似於傳統上以繼電器控制接觸器的階梯圖，梯形圖是通過連線把 PLC 指令的梯形圖符號連接在一起的連通圖，用以表達所使用的 PLC 指令及其前後順序，它與電氣原理圖很相似。它的連線有兩種：一為母線，另一為內部橫豎線。內部橫豎線把一個個梯形圖符號指令連成一個指令組，這個指令組一般總是從裝載（LD）指令開始，必要時再繼以若干個輸入指令（含 LD 指令），以建立邏輯條件。最後為輸出類指令，實現輸出控制，或為資料控制、流程控制、通訊處理、監控工作等指令，以進行相應的工作。



階梯範例圖

4. 順序功能流程圖（Sequential Function Chart，SFC）：類似於流程設計（Flow Design），流程圖中的步驟組合而完成，主要是規劃動作順序的流程圖，故謂之順序功能流程圖。所謂步序式控制，即是一步一步控制，而這一步與上一步是有關連性的，有順序性的。必須有上一個動作（STL），才會啟動（SET）下一個動作（STL）。



順序功能流程範例圖

5.功能區塊圖（Function Chart Programming，FBD）：以畫電路圖的方式來寫 PLC 程式。常用的程式及迴路可透過 FB（功能區塊）的建立輕易地重複利用。其他一些高檔的 PLC 還具有與電腦相容的 C 語言、BASIC 語言、專用的高階語言（如西門子公司的 GRAPH5、三菱公司的 MELSAP、富士電機的 Micrex-SX 系列），還有用布林邏輯語言、通用電腦相容的組合語言等。

2.5 **Arduino** 原理說明與介紹

Arduino 是一家製作開源電腦硬體和軟體的公司，同時兼有專案和用戶社群，他負責設計和製造單板微控制器和微控制器套件，用於構建數字裝置和互動式物件，以便在物理和數字世界中感知和控制物件。 該專案的產品是按照 GNU 寬通用公共許可證（LGPL）或 GNU 通用公共許可證（GPL）[1] 許可的開源硬體和軟體分發的，arduino 允許任何人製造 Arduino 板和軟體分發。 Arduino 板可以以預裝的形式商業銷售，也可以作為自己動手（DIY）套件購買。

Arduino 電路板設計使用各種微處理器和控制器。 這些電路板配有一組數字和模擬輸入/輸出（I/O）引腳，可以連線各種擴充功能板或麵包板（遮蔽板）和其他電路。 這些電路板具有串行通訊介面，包括某些型號上的通用序列匯流排（USB），也用於從個人電腦載入程式。 微控制器通常使用來自程式語言 C 和 C ++的特徵的方言進行編程。 除了使用傳統的編譯工具鏈之外，Arduino 專案還提供了一個基於 Processing 語言專案的整合式開發環境（IDE）。

Arduino 專案始於 2003 年，作為義大利伊夫雷亞地區互動設計研究所 Ivrea 的學生專案，目的是為新手和專業人員提供一種低成本且簡單的方法，以

建立使用傳感器與環境相互作用的裝置執行器。 適用於初學者愛好者的此類裝置的常見範例包括簡單機器人，恆溫器和運動檢測器。

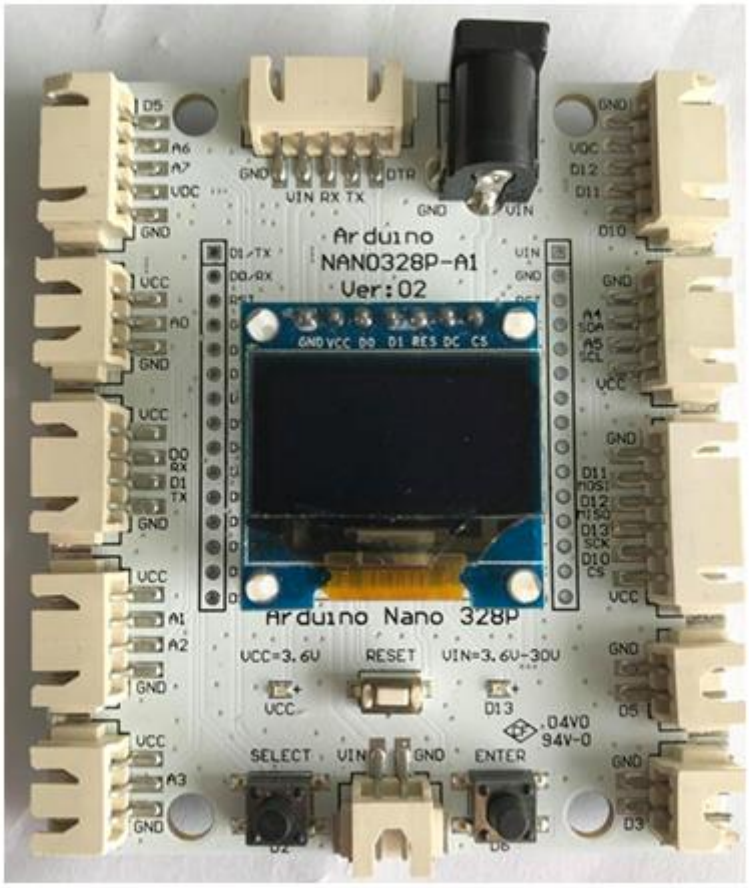
1.特色

- 基於創用 CC 開放原始碼的電路圖設計。
- 免費下載，也可依需求自己修改，但需遵照姓名標示。您必須按照作者或授權人所指定的方式，表彰其姓名。
- 依相同方式分享，若您改變或轉變著作，當散布該衍生著作時，您需採用與本著作相同或類似的授權條款。
- Arduino 可使用 ICSP 線上燒入器，將 Bootloader 燒入新的 IC 晶片。[5]
- 可依據 Arduino 官方網站，取得硬體的設計檔，加以調整電路板及元件，以符合自己實際設計的需求。[6]
- 可簡單地與感測器，各式各樣的電子元件連接，如紅外線、超音波、熱敏電阻、光敏電阻、伺服馬達...等。
- 支援多樣的互動程式，如 Adobe Flash, Max/MSP, VVVV, Pure Data, C, Processing…等。
- 使用低價格的微處理控制器（Atmel AVR）（ATMEGA 8,168,328 等）。
- USB 介面，不需外接電源。另外有提供直流（DC）電源輸入。

2.硬體

原始的 Arduino 硬體是從一間義大利公司 Smart Projects 製造。[7]有些 Arduino 硬體則是被官方授權由美國公司 SparkFun Electronics 和 Adafruit Industries 設計。

Arduino Nano



3.軟體

在 Arduino 上執行的程式可以使用任何能夠被編譯成 Arduino 機器碼的程式語言編寫。而 Atmel 也提供了數個可以開發 Atmel 微處理機程式的整合開發環境，AVR Studio 和更新的 Atmel Studio。

Arduino 範例圖

1.IDE

而 Arduino 計劃也提供了 Arduino Software IDE，一套以 Java 編寫的跨平台應用軟體。Arduino Software IDE 源自於 Processing 程式語言以及 Wiring 計劃的整合開發環境。它是被設計於介紹程式編寫給藝術家和不熟悉程式設

計的人們，且包含了一個擁有語法突顯、括號符合、自動縮排和一鍵編譯並將執行檔燒寫入 Arduino 硬體中的編輯器。

Arduino Software IDE 使用與 C 語言和 C++相仿的程式語言，並且提供了包含常見的輸入/輸出函式的 Wiring 軟體函式庫。在使用 GNU toolchain 編譯和連結後，Arduino Software IDE 提供了一個程式「avrdude」用來轉換可執行檔成為能夠燒寫入 Arduino 硬體的韌體。

2.Sketch

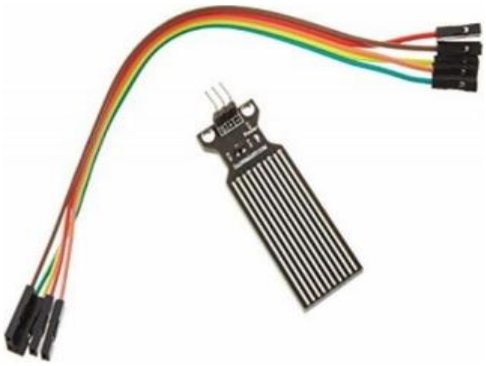
使用 Arduino Software IDE 編寫的程式被稱為「sketch」。[16]一個典型的 Arduino C/C++ sketch 程式會包含兩個函式，它們會在編譯後合成為 main() 函式：

- *setup()*：在程式執行開始時會執行一次，用於初始化設定。
- *loop()*：直到 Arduino 硬體關閉前會重複執行函式放的程式碼。

2.6 zigbee 技術

ZigBee 是 IEEE802.15.4 協議的代名詞。根據這個協議規定的技術是一種近距離、 低複雜度、低功耗、低數據速率、低成本的雙向無線通信技術，主要適合於自動控制 和遠程控制領域，可以嵌入各種設備中，同時支持地理定位功能。ZigBee 的特點主要 有以下幾個方面： (1)低功耗：在低耗電待機模式下，2 節 5 號乾電池可支持 1 個節點工作 6-24 個

月，甚至更長。這是 ZigBee 的突出優勢。相比之下藍牙可以工作數周、WiFi 可以工作數 小時 (2)低速率：ZigBee 工作在 250kbps 的通訊速率，滿足低速率傳輸數據的應用需求 (3)近距離：傳輸範圍一般介於 10～100m 之間，在增加 RF 發射功率後，亦可增加到 1-3km。這指的是相鄰節點間的距離。如果通過路由和節點間通信的接力，傳輸距離 將可以更遠。



zigbee

2.7 藍芽技術

藍牙 4.0 最重要的特性是省電科技，藍芽低功耗(Bluetooth Low Energy, BLE)，極低的執行和待機功耗可以使一粒鈕釦電池連續工作數年之久。此外，低成本和跨廠 商互操作性，3 毫秒連線超低延遲、100 米以上超長距離通訊、AES-128 加密等諸多特 色，可以應用於計步器、心律監視器、智能儀表、感測網路、物聯網等眾多領域，大 大擴展藍牙技術的

應用範圍。 藍牙 4.0 依舊向下兼容，包含經典藍牙技術規範和最高 速度 24Mbps 的藍牙高速技術規範。三種技術規範可單獨使用，也可同時運行。

BLE

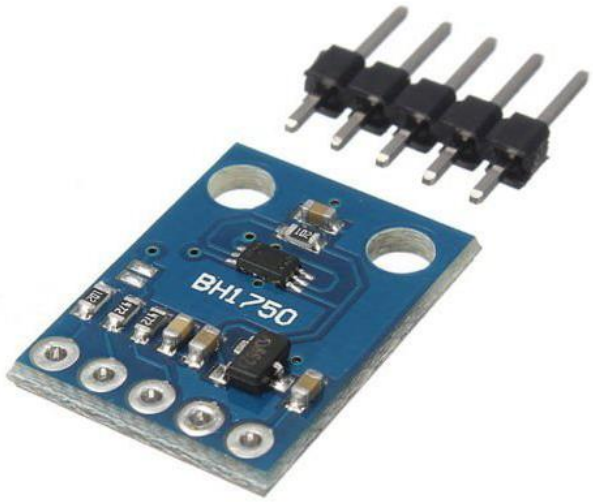
2.8 土壤感測器

土壤感測器為一個簡易水分感測器，利用電阻變化測量水分濕度。土壤越乾燥，輸出電壓越小。越溼潤輸出電壓越大。將本模組插入土中，通過 AD 轉換電壓信號，即可檢測土壤水分可以與 Arduino、小水泵等模組構成自動澆花系統。

土壤感測器

2.9 光照度感測器

檢測亮度輸出模擬量可以設定某個亮度值輸出數字量靈敏度，性能穩定工作電壓 5V，數字量模擬量雙輸出。感測器採用光敏二極體，方向性好，可以感知固定方向的光端口說明：DO 為數字量輸出，AO 為模擬量輸出。光線亮度檢測,光線亮度傳感器，具有方向性，只感應傳感器正前方的光源，用於尋光效果更佳。



光照度感測器

2.10 液面感測器

利用液面感測器，控制水塔之水位，水位過低時可利用手自動抽水控制，使水塔能保持有水狀態。

2.10 圖

第三章 軟體開發

3.1 Visual Basic(簡稱 VB)

Visual Basic（**VB**）是由微軟公司開發的包含環境的事件驅動程式語言。它源自於 BASIC 程式語言。VB 擁有圖形化使用者介面（GUI）和快速應用程式開發（RAD）系統，可以輕易的使用 DAO、RDO、ADO 連線資料庫，或者輕鬆的建立 ActiveX 控制項。程式設計師可以輕鬆地使用 VB 提供的元件快速建立一個應用程式。

VB 的中心思想就是要便於程式設計師使用，無論是新手或者專家。VB 使用了可以簡單建立應用程式的 GUI 系統，但是又可以開發相當複雜的程式。VB 的程式是一種基於表單的視覺化元件安排的聯合，並且增加代碼來指定組建的屬性和方法。因為預設的屬性和方法已經有一部分定義在了元件內，所以程式設計師不用寫多少代碼就可以完成一個簡單的程式。過去的版本裡面 VB 程式的效能問題一直被放在檯面上討論，但是隨著電腦速度的飛速增加，關於效能的爭論已經越來越少。

第四章 展示系統建置與整合測試

4.1 展示系統架構製作(原硬體呈現)

4.1 圖

4.2 液面感測器測試

利用短路測試液面感測器是否損壞。

4.2 圖

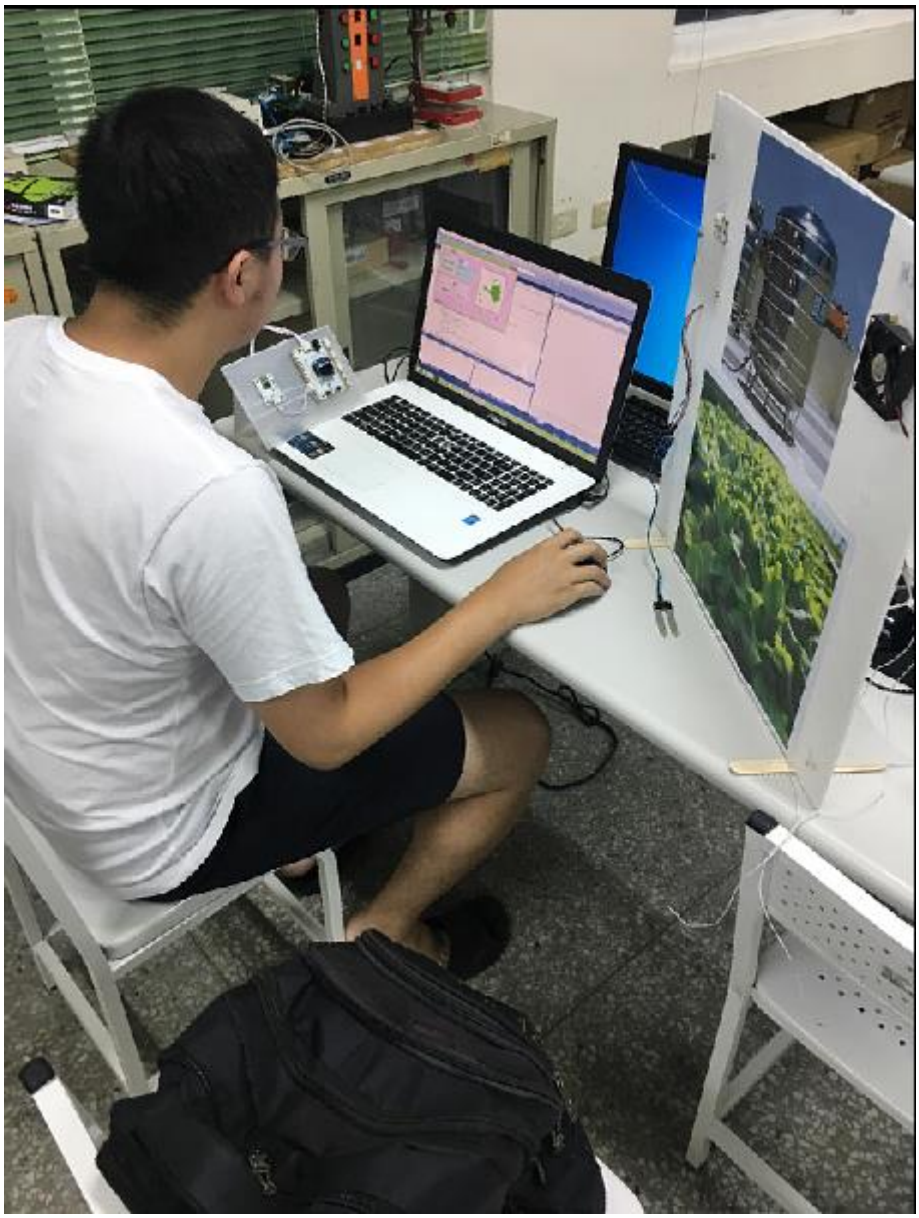
4.3 軟體程式測試

測試軟體是否正常運作，找出 bug 點加以修改。

4.3 圖

4.3 感測器數值

測試感測器傳發與收的數值



4.3 圖

第五章 參考文獻

1. https://zh.wikipedia.org/wiki/Visual_Basic 維基百科

2. https://zh.wikipedia.org/wiki/ZigBee 維基百科

3. https://www.arduino.cc/ Arduino 官方網站

4. https://www.google.com.tw/search?q=%E5%9C%96%E7%89%87&hl=zh-TW&source=lnms&tbm=isch&sa=X&ved=0ahUKEwiT3oO_ibzbAhUKbbwKHWCWBewQ_AUICigB&biw=1024&bih=633 GOOGLE 圖片

6. http://120.109.165.16/iotdevkit/ B217 實驗室

第六章 結論

本作品的監控系統皆提供多功能的感測與控制接點，電源供應，依照農作物的環境需求可彈性配置所需要的環境控制系統，傳至監控主機、手機，提供室外農夫掌握即時的環境監控狀況。農夫的收益提升，吸引更多年輕人的投入，帶動農業生產力升級。而且可提供一般國民健康、高品質的農產品。

第七章 作者簡介

一、基本資料

姓名 :童士綱

性別 : 男

出生日 : 86 年 4月 16 日

Email : a556695279877@yahoo.com.tw

二、學歷

修平科技大學---電機工程系在學

三、經歷

曾當任過物聯網與自慧科技的 TA，及兩年的衛生股長

五、專長

VB程式和焊接

六、證照與競賽證明

硬體裝修(乙級)

硬體裝修（丙級）

工業電子(丙級)

一、基本資料

姓名:林建旻

性別:男

出生日：85 年 10月 10 日

Email ： jcek851010@yahoo.com.tw

二、學歷

光華高工畢業(104級)、修平科技大學---電機工程系在學

三、經歷

曾當任一年的輔導股長、及電機工程系系會總務股長

五、專長

配線、美工

六、證照與競賽證明

室內配線(丙級)

工業配線(丙級)

工業電子(丙級)

附錄 1

1.VB 與程式及介紹

```
Public Class Form1
    Dim SQL AsSQL_TypeDef = New SQL_TypeDef
```

以上為定義 SQL 這個程式

```
 Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        For Each spAs String In My.Computer.Ports.SerialPortNames
    ComboBox1.Items.Add(sp)
      Next
    MySQL_Init(SQL, 20, "127.0.0.1", "root", "123456",
"outdoor_sensor")
      End Sub
```

以上這一段是在說明，我接收到的資料是要傳到

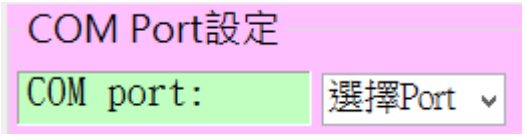
(SQL 這個地方，最多可一次傳送 20 個東西，要傳送的網址為

“127.0.0.1”，帳號為”root”，密碼為”123456”)

```
 Private Sub ComboBox1_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
ComboBox1.SelectedIndexChanged
        SerialPort1.PortName =
ComboBox1.Items(ComboBox1.SelectedIndex)

    SerialPort1.Open()
        Timer1.Enabled = True
    End Sub
```

以上這一段是在說明，ComboBox1 這一樣東西，這一樣東西是負責選擇 COM port 的位子



SerialPort1.Open()

這一段是在說明開啟新的序列來連結

Timer1.Enabled = True

這一段是在說明計時器(Timer1)啟用為真

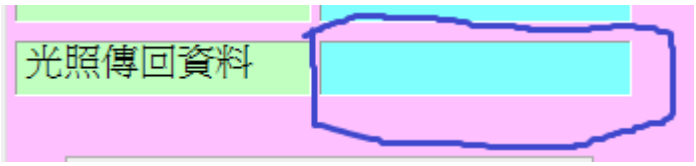
Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Timer1.Tick
Dim RXD$ = SerialPort1.ReadLine()

上面這一段是在定義說(RXD$)要讀取輸入緩衝區 System.IO.Port.NewLine 值之前的內容

If RXD$ <> "" Then

這一段是在說明，如果 RXD$這個值不等於零的時候

If Mid(RXD$, 1, 1) = "L" Then
Label6.Text = Mid(RXD$, 3, RXD$.Length - 2)
End If



這一段是在說明，如果 RXD$收到值之後，(RXD$，的第一個字開始到第一個字結束)他的值是”L”的話，Label6 的變數為(收到的值 RXD$，從第三個位子，並且從減兩個位子開始為輸入的值)

結束如果

```
If Mid(RXD$, 1, 1) = "S" Then
                    Label2.Text = Mid(RXD$, 2, RXD$.Length - 1)
```

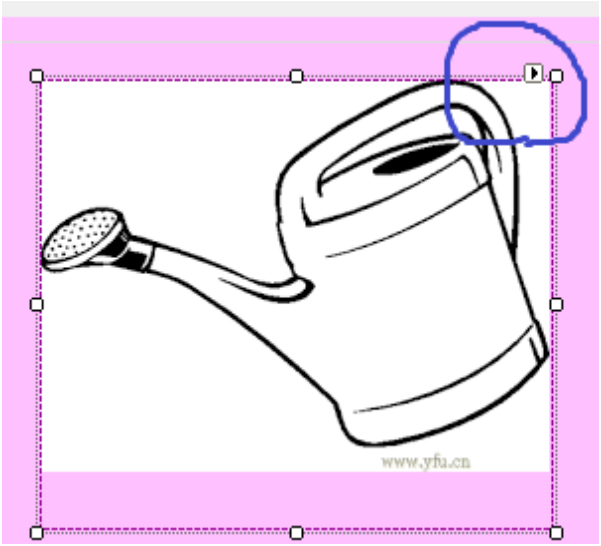
這一段是在說明，如果 RXD$收到值之後，(RXD$，的第一個字開始到第一個字結束)他的值是”S”的話，Label2 的變數為(收到的值 RXD$，從第二個位子，並且從減一個位子開始為輸入的值)


```
If Label2.Text < 1000 Then
                        PictureBox1.Image = My.Resources._111111
                    End If
```

這一段是在說明，如果 Label2 的變數值小於 1000 的話

會叫出 PictureBox1 這個元件裡的 My.Resources. _111111 這一個圖

，My.Resources. _111111 這一個圖是從元件 PictureBox1 加入的



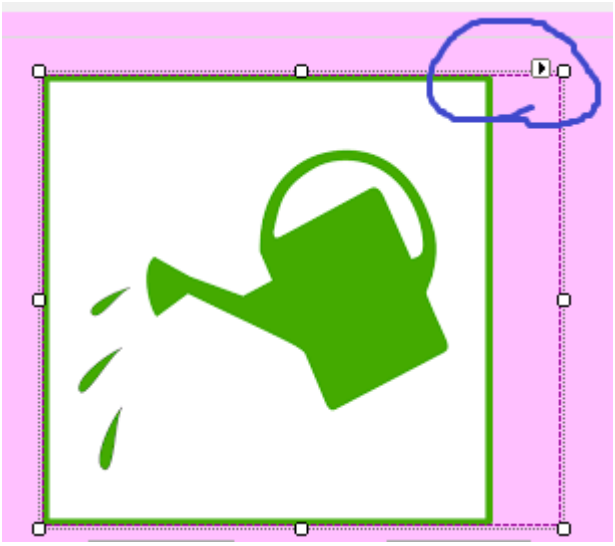
If Label2.Text > 900 Then

PictureBox1.Image = My.Resources. _222222

這一段是在說明，如果 Labe12 的變數值小於 900 的話

會叫出 PictureBox1 這個元件裡的 My.Resources. _222222 這一個圖，

My.Resources. _222222，這一個圖是從元件 PictureBox1 加入的



End If

End If

End If

以上都是為結束如果

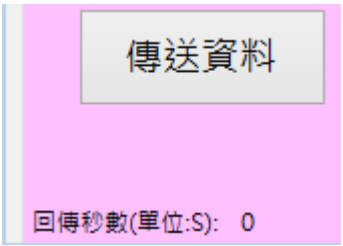
End Sub

Private Sub Button7_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button7.Click

Timer2.Enabled = True

End Sub

當按下 Button7 這一個元件時



Timer2 將會開始計時

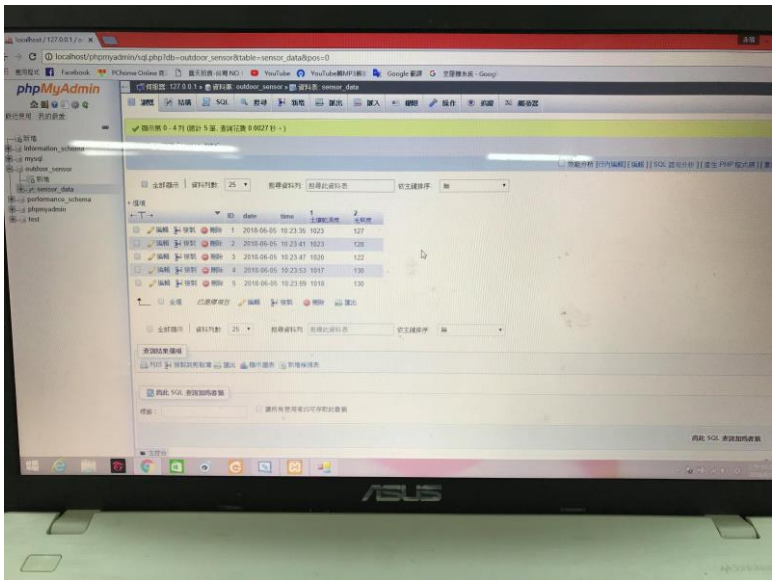
```
    Private Sub Timer2_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer2.Tick
            Label3.Text = Val(Label3.Text) + 1
            If Label3.Text > 5 Then
    SQL.Form = "sensor_data"
    SQL.Column(1) = Label2.Text
    SQL.Column(2) = Label6.Text
    MySQL_Add(SQL, 2)
                    Label3.Text = 0
            End If
        End Sub
```

以上這一段落是在說明，Label3 的變數值等於傳回來的變數值加一

如果傳回來的變數值大於 5 的話，

SQL.Form = "sensor_data"

(傳送到資料庫=”sensor_data”)這個位子



3.1 圖資料庫

SQL.Column(1) = Label2.Text

傳送 Label2 的變數到 Column(1)這一個位子

SQL.Column(2) = Label6.Text

傳送 Label6 的變數到 Column(2)這一個位子

MySQL_Add(SQL, 2)

開始上傳至 SQL，只有兩個資料

Label3.Text = 0
End If　End Sub

將 Label3 的變數歸零，

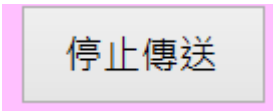
後面一樣是結束如果

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Timer2.Enabled = False
        Label3.Text = 0
    End Sub
```

以上這一段落是在說明

當按下 Button1 這個元件時，中止 Timer2 的計時

並且將 Label3 的變數歸零



```
    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
            PictureBox1.Image = My.Resources._111111
        End Sub
```

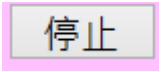
當按下 Button3 個個元件

他會跑出

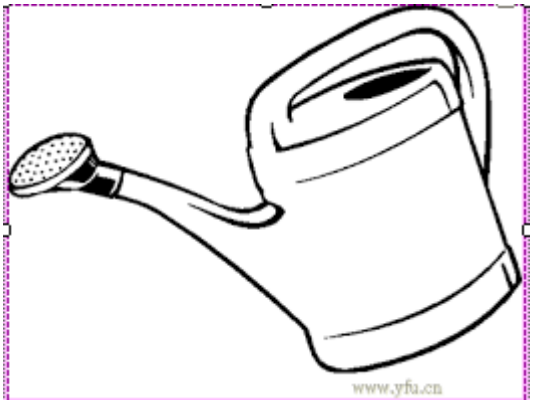
```
    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click

    PictureBox1.Image = My.Resources._222222
        End Sub
    End Class
```

當按下 Button4 個個元件



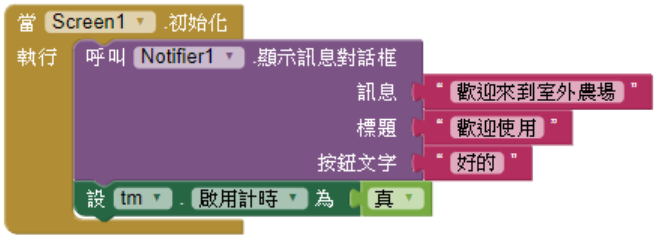
他會跑出



2.App Inventor 2

進入時的介面

左邊為運用到的元件

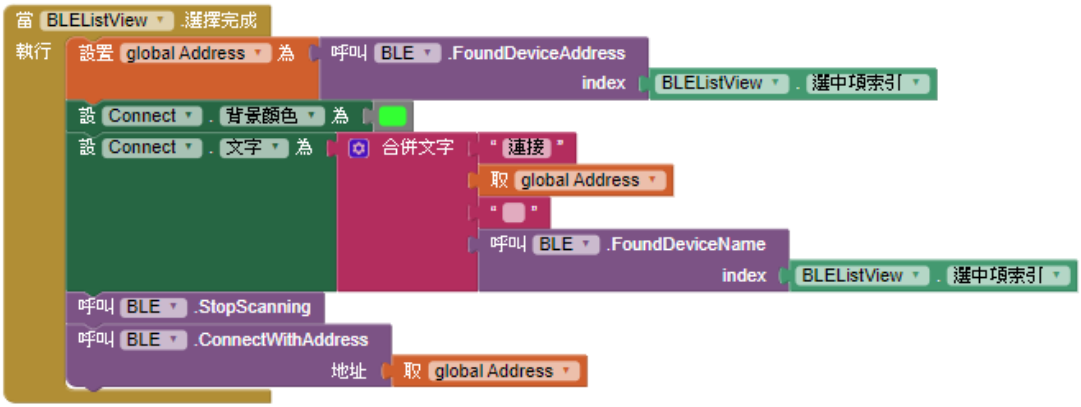


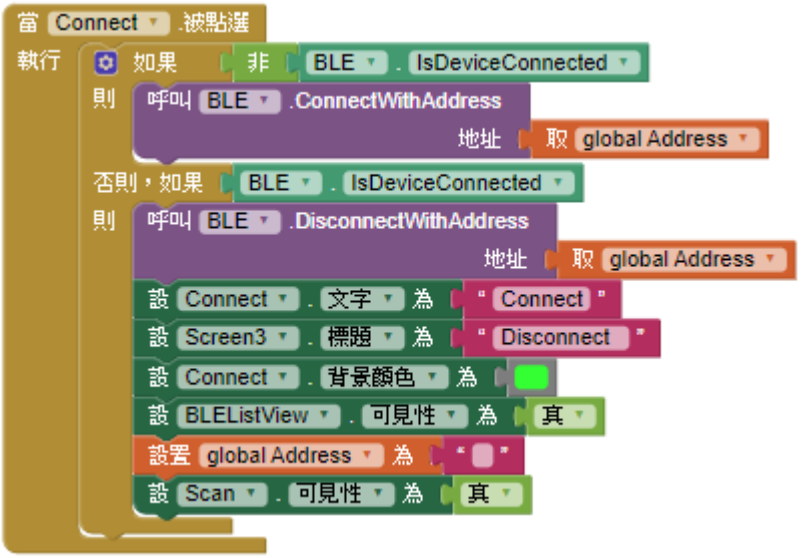
上圖為此介面的程式

下圖為點入介面時的畫面

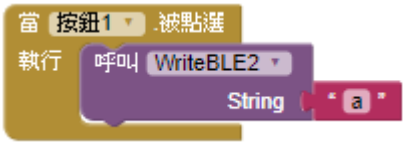
上圖為運用到的元件

以下為程式



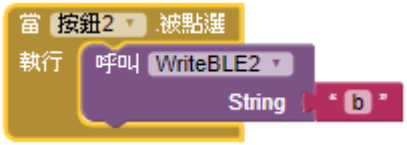


以上是在做藍芽的定義



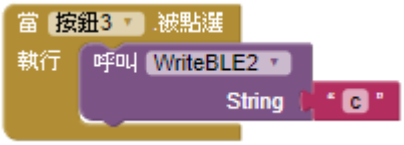
以上是在說明按下按鈕 1

會對 BLE 傳送 a 值



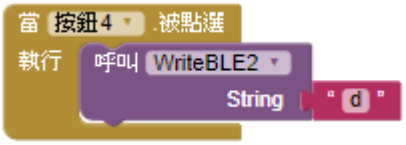
以上是在說明按下按鈕 2

會對 BLE 傳送 b 值



以上是在說明按下按鈕 3

會對 BLE 傳送 c 值



以上是在說明按下按鈕 4

會對 BLE 傳送 d 值

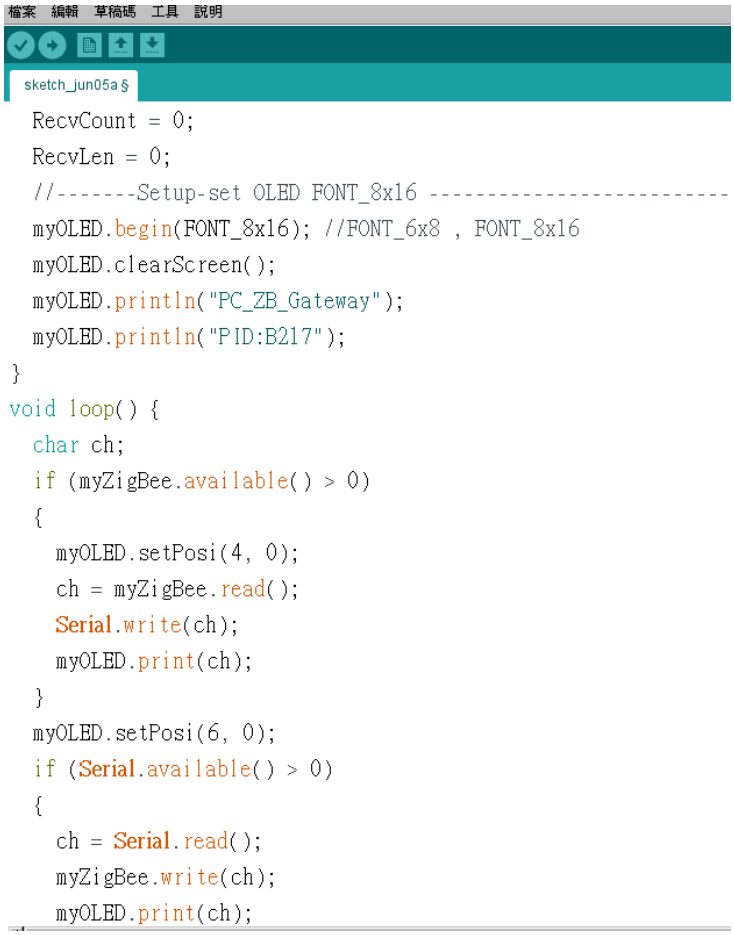
負責收尋藍芽的位子

此段為頁面初始化

初始化 UUID

3. Arduino 程式

1. Zigbee 發射端




```
  RecvCount = 0;
  RecvLen = 0;
  //-------Setup-set OLED FONT_8x16 -------------------------
  myOLED.begin(FONT_8x16); //FONT_6x8 , FONT_8x16
  myOLED.clearScreen();
  myOLED.println("PC_ZB_Gateway");
  myOLED.println("PID:B217");
}
void loop() {
  char ch;
  if (myZigBee.available() > 0)
  {
    myOLED.setPosi(4, 0);
    ch = myZigBee.read();
    Serial.write(ch);
    myOLED.print(ch);
  }
  myOLED.setPosi(6, 0);
  if (Serial.available() > 0)
  {
    ch = Serial.read();
    myZigBee.write(ch);
    myOLED.print(ch);
```

2. Zigbee 接收端

```
sketch_jun05a §
  currentPalette = RainbowColors_p;
  currentBlending = LINEARBLEND;
 //-------Setup-set OLED FONT_8x16 ----------------------------------------
  myOLED.begin(FONT_8x16); //FONT_6x8 , FONT_8x16
  myOLED.clearScreen();
  myOLED.println("ZBTag_LED_RGB_BZ");
  myOLED.println("PanID:B217");
}

// the loop routine runs over and over again forever:
void loop() {
   recv=0;
   String uartRead="";
  if (myZigBee.available())
  {
      uartRead = myZigBee.readString();
      myOLED.setPosi(6,0);
      myOLED.print(uartRead);
      recv=1;
  }
 if (Serial.available())
  {
      uartRead = Serial.readString();
```

3. 藍芽