

國立臺灣大學生物資源暨農學院生物產業機電工程學系



碩士論文

Department of Bio-Industrial Mechatronics Engineering

College of Bioresources and Agriculture

National Taiwan University

Master Thesis

遠端遙控變量施噴裝置之研製

Development of a Variable-Amount Spraying Device with
Remote Control

戴冠丞

Guan-Cheng Dai

指導教授：葉仲基 博士

Advisor: Dr.-Ing. Chung-Kee Yeh

中華民國 107 年 7 月

July 2018

國立臺灣大學碩士學位論文
口試委員會審定書



遠端遙控變量施噴裝置之研製

Development of a Variable-Amount Spraying
Device with Remote Control

本論文係戴冠丞君（學號 R03631042）在國立臺灣大學
生物產業機電工程學系、所完成之碩士學位論文，於民國
107 年 07 月 13 日承下列考試委員審查通過及口試及格，特
此證明

口試委員：

葉仲基

（簽名）

（指導教授）

吳剛智

黃珮屏

系主任、所長

鄭宗記

（簽名）

誌謝



研究成果的背後總有許多人默默給予協助與鼓勵，首先我想最先感謝的是指導教授葉仲基老師，葉老師在我碩士歷程不順利的情形下給予我這次完成學位的機會，並在農機議題給予我很大的空間去發掘並探討相關之議題，也讓我順利產出這次的成果，真的由衷地感謝老師。

其次要感謝的是在Arduino程式語言上給我諸多協助的思好和朝喻，雖然我曾對程式語言有過接觸，但從事前的諸多不順到最後完成並實際應用的過程中，給予我很多相關的建議與改良方法，進而改出最後的結果，在此也特別謝謝他們。

最後當然也要感謝參與口試的兩位口試委員，吳剛智老師與黃振康老師，以及在我論文卡關時在一旁支持家人與女朋友，在研究所的過程中我也學到很多實驗問題的應對，以及整體規劃與實作上細節的處理，並將其進行討論與整合，再次感謝所有參與本篇論文的老師、同儕與家人，也將本文獻給每位尋找問題並試圖解決答案的觀文者。

摘要



精準農業的議題在近期受到廣泛的討論。然而，至今鮮少有文獻針對具稀釋功能的變量噴霧設備進行探討。本研究之目標將研製一具稀釋功能的變量設備並由液態肥料去檢驗其功能性。藉由一具備Wi-Fi功能之Arduino控制之混合系統的使用，可得到閥開度與流量的關係並進一步決定閥開度與稀釋倍率的關係。研究結果指出當閥開度在24%至34%時呈現一線性趨勢且流量值落在0.17 L/min至0.34 L/min，稀釋倍率在1.5倍至3倍間具有較高的準確度。此外，一款app被研發讓使用者能指定稀釋倍率並透過Wi-Fi傳送資料。總結來說，吾人成功研製一臺精準的、具稀釋功能與Wi-Fi控制的變量噴霧設備。本研究期望有助於未來對於變量應用的討論。

關鍵字：變量施噴、混合系統、Wi-Fi控制

Abstract



The issue of precision agriculture has received considerable critical attention. However, few studies to date have investigated variable-amount spraying device with dilution. The objectives of this research were to develop variable-amount device with dilution and examine its utility with liquid fertilizer. Using an Arduino-controlled hybrid system with Wi-Fi, the correlations between valve travel position and volumetric flowrate were calculated to determine the relationship of valve travel position and dilute factor. Results of this research revealed that a linear trend appear when the valve travel position is within 24% and 34% and the flowrate is within 0.17 L/min and 0.34 L/min, and also showed that dilution factor was highly accuracy within 1.5 and 3.0. Moreover, an app was developed so that users could assign the dilute factor and sent data via Wi-Fi. In conclusion, we have been successful in developing an accurate variable-amount spraying device with dilution and WiFi-controlled. This research expects in contribute to ongoing discussion about dilution in variable-amount application.

Keywords: variable-amount, hybrid system, Wi-Fi controlled

目 錄



口試委員審定書	
誌謝	
摘要.....	ii
Abstract.....	iii
目 錄	
圖目錄.....	vi
表目錄.....	vii
第一章 前言.....	1
1.1 研究背景.....	1
1.2 研究目的.....	3
第二章 文獻探討.....	4
2.1 精準農業概論.....	4
2.2 農藥與液態肥料施噴.....	5
2.2.1 農藥施噴規範.....	5
2.2.2 液態肥料施噴概況.....	7
2.3 變流量噴藥機之分類.....	10
2.3.1 地圖基礎變流量應用.....	11
2.3.2 感測相關之變流量應用.....	11
2.4 Arduino 簡介與農機相關應用.....	12
第三章 材料與方法.....	14
3.1 儀器及待測物.....	14
3.1.1 微控制器與通訊模組.....	14



3.1.2 混合系統.....	16
3.2 控制系統.....	19
3.2.1 控制系統連接.....	19
3.2.2 溶液混合系統.....	22
第四章 結果與討論.....	26
4.1 電動閥之閥開度與流量探討.....	26
4.2 溶液稀釋效益探討.....	30
4.3 App 系統整合.....	33
第五章 結論與建議.....	35
參考文獻.....	36
附錄一 Arduino 與 App 互動之程式碼.....	39
附錄二 App Inventor 開發磚塊圖.....	48

圖目錄



圖 1.1 土壤澆灌.....	2
圖 1.2 葉面施肥.....	2
圖 2.1 精準農業的方法與目的.....	4
圖 2.2 無人機噴灑實體照.....	7
圖 2.3 無人噴霧車.....	13
圖 3.1 Arduino mega 2560 微控制器.....	14
圖 3.2 ESP-01 Wifi 晶片.....	15
圖 3.3 Arduino IDE 介面.....	15
圖 3.4 App 介面.....	16
圖 3.5 山野電動閥外觀.....	17
圖 3.6 流量計(右)與顯示器(左).....	18
圖 3.7 實驗流程.....	19
圖 3.8 資料傳輸路徑.....	20
圖 3.9 全系統線路連接.....	20
圖 3.10 電動閥配線實體照.....	21
圖 3.11 Thingspeak 圖表介面.....	22
圖 3.12 水槽與電動閥連接示意圖.....	23
圖 3.13 流量測試系統連接示意圖.....	24
圖 3.14 流量測試系統連接實體照.....	24
圖 3.15 混合系統連接示意圖.....	25
圖 3.16 混合系統連接實體照(一).....	26
圖 3.17 混合系統連接實體照(二).....	26
圖 4.1 流量與時間關係圖.....	27

圖 4.2 一分鐘流量與閥開度關係圖.....	28
圖 4.3 兩分鐘流量與閥開度關係圖.....	28
圖 4.4 三分鐘流量與閥開度關係圖.....	29
圖 4.5 理論稀釋倍率與實際稀釋倍率關係圖.....	31
圖 4.6 電導度值輸入.....	33
圖 4.7 Thingspeak 輸入值接收.....	33
圖 4.8 Thingspeak 回傳值接收.....	34
圖 4.9 App 端回傳值接收.....	34

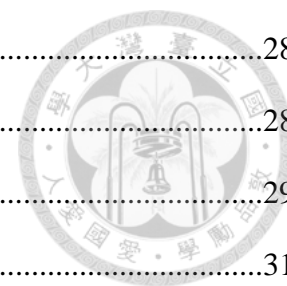


表 目 錄



表 2.1 農產品於不同國家之最大殘留容許量比較.....	6
表 2.2 市售植物營養劑之電導度資訊.....	8
表 2.3 有機肥料之電導度資訊.....	9
表 2.4 部分肥料品目與規格.....	9
表 2.5 變流量施噴應用之研究.....	10
表 2.6 Arduino於農機之相關研究.....	13
表 3.1 山野電動閥之產品參數.....	17
表 3.2 農助旺植物營養液之成分資訊.....	18
表 3.3 部分 AT command 指令與意義.....	21
表 4.1 溶液混合之稀釋倍率與進料、水之閥開度.....	30
表 4.2 一分鐘稀釋倍率與理論值比較.....	32
表 4.3 兩分鐘稀釋倍率與理論值比較.....	32
表 4.4 三分鐘稀釋倍率與理論值比較.....	32

第一章 緒論



1.1 研究背景

變流量施噴系統於精準農業的觀念提倡下日益重要，有別於傳統農業維持耕作的一貫準則，精準農業更重視人為耕作對環境的影響，包含土壤養分維持與污染防治、農藥擴散問題及肥料施用等範疇。然而，現行的農業機具仍未具備足夠的功能實現精準農業的理念，就農藥施噴而言，由於市面上農藥種類繁多，稀釋倍率也不盡相同，在未熟悉及了解使用方式的情況下，導致許多不當施噴的情形，長期下來引發土壤環境污染與農作物變異。

為預防農藥施噴對人體的危害，行政院衛福部有規範農藥殘留容許量標準，限制各農藥對不同作物的施噴濃度，然而農藥濃度無法透過現行的噴藥機控制，僅由調配時的稀釋倍率決定，若未依正確的倍率進行稀釋，很容易有農藥超標的問題，進而在後續危害食用者的健康。

除了農藥噴灑的問題外，近年來液態肥料的興起也慢慢取代傳統的顆粒肥料，原因在於液態肥料對於農作物而言較易被吸收，能在短時間內替作物補充養分，使作物得以提早收成，或得到品質更高的產物。此外，在人力匱乏的情況下，液態肥料對於大規模噴灑（如無人機噴灑）亦有較佳的覆蓋率，但此類肥料同樣存在著稀釋倍率的問題，根據耕作方法、作物種類及肥料種類皆會有不同的稀釋倍率，若濃度太高則會導致肥害，反倒使作物受損。

目前液態肥料的使用方使主要以土壤澆灌(圖1.1)及葉面施肥(圖1.2)為主。土壤澆灌是指將養分溶於水中灌溉於土壤，由作物的根部吸收養分獲得成長；葉面施肥則是將養分直接噴灑在作物的葉面，使養分由葉片氣孔被作物吸收以補足根部吸收效率不彰的問題，但由於葉面組織較為脆弱，液態肥料的稀釋倍率需要更高以防止葉面受損。為此，如何在不同情境下正確調配肥料也成為耕作時相當重要的課題。



圖1.1 土壤澆灌



圖1.2 葉面施噴

1.2 研究目的

液體肥料濃度過高與農藥超標問題多半是源自於使用錯誤的稀釋倍率而導致，為有效提升稀釋倍率的正確性，本研究致力於研發一台具備濃度稀釋功能的裝置，以液態肥料為濃度稀釋的待測物置入水槽內，並加裝電導度計測定其電導度值，再串接上電動閥後以Arduino微控制器進行調控，研究內容將討論電動閥後閥的開度與流量的關係，並探討閥開度對稀釋倍率的影響，以實現現行農業機具無法對濃度進行有效調控的缺陷。

第二章 文獻探討



近年來，以精準農業為核心的噴藥機改良以相當成熟，由不同控制方法進行農藥與肥料的施用也相當多元，文獻探討中將回顧精準農業的意涵，並探討農藥與肥料的使用情況，以及噴藥裝置之演進。

2.1 精準農業概論

精準農業為一透過資訊科技所得到的資訊去改善及決定作物產量之經營策略(美國國家科學研究委員會, 1997)，其內容可分為定位系統、產區辨識、遠端遙控、環境監測、變流量系統、以及資料傳輸(Mondal and Tewari, 2004)。對比傳統農業行為，精準農業更重視數據的收集與分析，並提供相對應的施作量，這些關係經統整後如圖2.1所示。

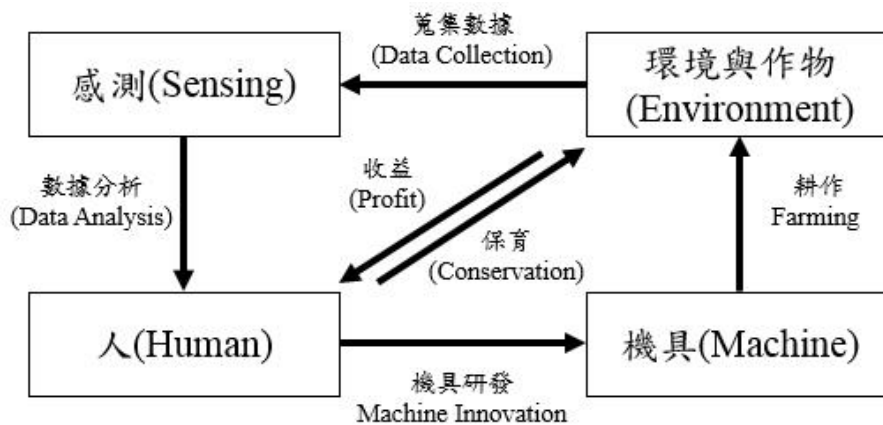



圖2.1 精準農業的方法與目的。

近年來隨著環境保育意識的抬頭，精準農業於環境保育的議題討論越來越多，由於農藥施噴與肥料的使用對土壤及空氣會造成一定程度的影響，故各國法規也針對兩者訂定規範，透過用量限制達到長久之農耕環境保育(Bongiovanni, 2004)。

2.2 農藥與液態肥料施噴



早期農業耕作時為減少及預防病蟲害的發生，發展噴藥機噴灑農藥，近年來該類機具除施噴農藥外，亦有人噴灑液態肥料以提供作物更有效的養分吸收環境，甚至有些農民考量時間成本與人力資源，會將農藥與液態肥料直接混合後一同進行施噴，而各國針對農藥的濃度皆有相當程度的管制與規範，而液態肥料的部分亦有多數研究統整出建議的施噴濃度，為此將由法規與前人研究進行分析。

2.2.1 農藥施噴規範

農藥的用量會因作物種類、氣候與栽種情況而有所不同，然而為了顧及食品安全，各國針對市面上的農藥逐一訂定最大殘留容許量(Maximum Residue Limits, MRL)，以防止過高農藥用量對健康的危害。表2.1中列舉水稻、芒果及茶葉的數種農藥及其容許量為例，可看出各作物所使用的農藥種類相當繁雜，依據病蟲害的種類需要施噴3種以上的農藥，且各農藥之容許量也不盡相同，在複雜的施噴情況下，近年來也不斷有農藥超標的消息傳出。此外，由於各國對於容許量標準有差異，若要出口某項農產品，還要先確認出口國家之相關法定標準，例如：水稻種植所施噴的殺紋寧在臺灣之規範容許量為0.1 ppm，而在歐盟卻僅能有0.05 ppm的殘留；茶葉所使用的可尼丁在臺灣之規範容許量為5.0 ppm，在歐盟僅能有0.7 ppm的殘留。因此若該地生產之農產品是以出口為導向，其農藥容許量需要特別留意。

表2.1 農產品於不同國家之最大殘留容許量比較

作物	農藥種類	臺灣	日本	美國	歐盟
水稻	Hymexazol 殺紋寧	0.10	0.10	0.50	0.05
	Glyphosate 嘉磷塞	0.10	0.10	0.10	0.10
	Paraquat 巴拉刈	0.20	0.10	0.05	0.05
芒果	Pyraclostrobin 百克敏	0.50	0.05	0.60	0.05
	Spinetoram 賜諾特	0.20	0.30	0.30	0.05
	Cyromazine 賽滅淨	0.50	0.50	0.30	0.05
茶葉	Bifenthrin 畢芬寧	2.00	30.0	30.0	30.0
	Dicofol 大克蟎	3.00	3.00	50.0	20.0
	Clothianidin 可尼丁	5.00	50.0	70.0	0.70

最大殘留容許量單位：ppm

除了針對食品安全所制定的規範外，農藥擴散的問題也在環境保育的範疇中被廣泛的討論。其中，Ferrari等人(2003)針對耕作後空氣中的農藥濃度進行檢測，該研究以PELMO模型(Klein, 1997)為主，討論作物揮發與高度、水平距離之關係，其概念可由亨利定律(Henry's Law)與菲克定律(Fick's Law)表示如下：

$$J_v = D \left(\frac{HC_1}{L} \right) \quad (2.1)$$

其中， D 為空氣之擴散係數($cm^2 d^{-1}$)， J_v 為農藥之揮發速率($g cm^{-2} d^{-1}$)， L 為土壤上層之厚度(cm)， H 為亨利常數(-)，而 C_1 為農藥之濃度($g cm^{-3}$)。從理論模型中可知道：農藥擴散的質量通量，和農藥濃度成正相關，顯示若農藥濃度可抑制在某一個範圍值不超標，能夠減少飄散的問題，該篇論文於結論中提到約有22.6%的農藥會經由揮發飄散至空氣中影響空氣品質，且存在著危害附近住戶健康的風險，因此，精確控制農藥濃度亦有助於解決環境保育之問題。

此外，近年來隨著機械自動化的高度發展，耕作行為也逐漸被機器所取代。Spoorthi等人(2017)將無人機操控連接到app上；Yallappa等人(2017)於印度使用無人機進行農藥施噴可得到99%的覆蓋率表現，顯示無人機應用於農耕行為有著相當大的潛力。



圖2.2 無人機噴灑實體照(Yallappa等人，2017)

2.2.2 液態肥料施噴概況

肥料的使用一直以來都是農業行為相當重要的一環，市面上的肥料種類亦相當多元，依成分來源可分為有機肥料與化學肥料；若依外觀區分，則可分為固態肥料與液態肥料。其中，Sutton等人(1986)將各式肥料於不同田地施加並耕種五年進行作物產量之分析，並停止使用六年以觀察土壤肥沃度的表現。實驗結果發現有機肥料的施加效果較化學肥料的產量表現較佳，且使用液態有機肥料的收穫量在停止施用後仍有成長的趨勢。而Zhang等人(2003)更以氮含量相同的情況下比較液態肥料與固態肥料的表現，亦得到相同的結論，顯示液體肥料相對於固體肥料對於提升產量有顯著的效益。然而，液體肥料的施加頻率比起固體肥料更為頻繁，隨著農業人口外移與老化的情況下，使用上相對不易，目前仍以固態肥料為主，液態肥料為輔的方式進行耕作。

而在市面上販售的液態肥料及植物營養液中，蔣永正(2005)做了一份市售產品之化學性質研究，部分肥料之酸鹼值與電導度數據節錄如表2.2所示。其中，大部分植物營養劑若經過正確的稀釋，電導度值多落在2 mS/cm以下，根據美國鹽土實驗室的資料定義：當土壤加入純水使純水飽和時，電導度值大於4 mS/cm，則稱為土壤鹽化，故此範圍內的營養液能正常提供養分給作物。

表2.2 市售植物營養劑之電導度資訊

營養劑名稱	1%之電導度值 (mS/cm)	稀釋倍率	預期實際電導度* (mS/cm)
雙補	4.76	500~1000	0.48 ~ 0.95
高有機營養液	4.75	200~1500	0.32 ~ 2.38
葉面撒布劑	4.43	100~6000	0.07 ~ 4.43
美果多	3.35	200~1000	0.34 ~ 1.68
鮑水丹	2.44	800~1500	0.16 ~ 0.31
果肥1號	5.82	500~1000	0.58 ~ 1.16
果肥2號	2.17	500~1000	0.22 ~ 0.43

*預期電導度之計算式如下：預期電導度=1%電導度值/(稀釋倍率/100)

同樣的結論於21世紀初，王鐘和等人(2002)的著作中亦指出飽和土壤抽出液電導度大於2 mS/cm時，不耐鹽之作物產量會開始受到影響，大於4 mS/cm時大部分作物之生長都會受到抑制。而在實際耕作中，黃子彬(1996)指出施肥量會依據土壤鹽度不同而有所不同，若土壤電導度大於2.5 mS/cm時，則需要針對肥料添加進行改善，且施肥量因隨著土壤電導度而有所調整，可見電導度與施肥量密不可分的關係。

近年來，液態肥料的使用除了使用化學方法以外，針對有機肥料的研究亦相當多元，Romero等人(2013)將廢棄物之濾清液和市售液體肥料進行比較，由電導度分析得到前者有較高之電導度；Kalbani等人(2016)則以牛、雞、及魚類排泄物做為有機肥料，調製成液態後對番茄進行作物成長分析，上述肥料之電導度資訊如表2.3所示。從過往之研究可發現多數有機肥料經不同製程產出後，電導度值普遍偏高，需進行稀釋後才可使用，為此針對電導度的控制便成為土壤鹽度保持相當重要的一環。

表2.3 有機肥料之電導度資訊

有機肥料來源	電導度值(mS/cm)	參考資料
Sewage sludge	0.20	Jambaldorj et al.,2007
Cow Manure	10	Kalbani et al.,2016
Chicken+ cow	10	Kalbani et al.,2016
Chicken manure pellet	9	Kalbani et al.,2016
Agro fish pellet	10	Kalbani et al.,2016
Raw municipal waste compost leachate	8.04	Romero et al.,2013
Digested municipal waste compost leachate	12.10	Romero et al.,2013
Pig slurry	0.16	Sa'nchez et al.,2005

為有效管理多重來源之肥料，我國訂定「肥料管理法」控管各地農戶肥料之使用，除針對來源定義肥料種類外，其主成分(氮、磷、鉀)以及有害物質(重金屬)之含量也有明確規範，肥料種類與限制量節錄至表2.4中。

表2.4 部分肥料品目與規格

品項名稱	性狀	主成分	有害成分
一. 氮肥類			
尿素肥料	固態	全氮 > 45%	二縮脲態氮 < 1%
硫酸胍基尿素肥料	固態	全氮 > 32%	砷含量 < 25 mg/kg
二. 磷肥類			
過磷酸鈣肥料	固態	檸檬酸銨溶性 磷酐 > 30.0%	限制1
腐植酸磷肥料	固態	檸檬酸溶性 磷酐 > 15.0%	限制1(不含汞) 亞硝酸 < 0.20%
三. 鉀肥類			
氯化鉀肥料	固態	氧化鉀 _(aq) > 60.0%	碳酸鈉&硼酸鈉 < 0.50%
碳酸氫鉀肥料	固態	氧化鉀 _(aq) > 45.0%	氯 < 3.0%
四. 次微量要素肥料類			
硼酸肥料	固態	水溶性硼 > 17.00%	
硫酸亞鐵肥料	固態	水溶性亞鐵 > 20.00%	
五. 有機質肥料類			
液態雜項有機質肥料	液態	氮 + 磷酐 + 氧化鉀 > 1%，個別 > 0.1%	

表2.4 部分肥料品目與規格(續)


品項名稱	性狀	主成分	有害成分
六. 複合肥料類			
複合肥料	固態	氮 + 磷酐 + 氧化鉀 > 15.0% , 個別成分 > 1.0%。	限制2
七. 植物生長輔助劑類			
生長輔助劑-腐植酸	固態或液態	腐植酸 > 10.0%	限制3
雜項有機質栽培介質	固態	有機質 > 30.0%	限制3
八. 微生物肥料類			
豆科根瘤菌肥料	固態或液態	固態:1×10 ⁷ 落成數/g 液態:1×10 ⁸ 落成數/mL	限制3
溶磷菌肥料	固態或液態	固態:1×10 ⁷ 落成數/g 液態:1×10 ⁸ 落成數/mL	限制3
生長輔助劑-腐植酸	固態或液態	腐植酸 > 10.0%	限制3
雜項有機質栽培介質	固態	有機質 > 30.0%	限制3

說明：限制1：砷 < 25.0 mg/kg、鎘 < 15.0 mg/kg、鉻 < 150 mg/kg、銅 < 100 mg/kg、汞 < 1.0 mg/kg、鎳 < 25.0 mg/kg、鉛 < 150 mg/kg、鋅 < 250 mg/kg；限制2：砷 < 10.0 mg/kg、鎘 < 0.6 mg/kg、鉻 < 30 mg/kg、銅 < 20 mg/kg、汞 < 0.2 mg/kg、鎳 < 10.0 mg/kg、鉛 < 30 mg/kg、鋅 < 60 mg/kg；限制3：硫氰酸 < 0.07%、氨基磺酸 < 0.07%、二縮脲態氮 < 0.15%、亞硝酸 < 0.30%、砷 < 25.0 mg/kg、鉻 < 150 mg/kg、銅 < 100 mg/kg、汞 < 1.0 mg/kg、鎳 < 25.0 mg/kg、鉛 < 150 mg/kg、鈦 < 200 mg/kg、鋅 < 250 mg/kg；限制4：砷 < 25.0 mg/kg、鎘 < 2.0 mg/kg、鉻 < 150 mg/kg、銅 < 100 mg/kg、汞 < 1.0 mg/kg、鎳 < 25.0 mg/kg、鉛 < 150 mg/kg、鋅 < 250 mg/kg

2.3 變流量噴藥系統之分類

變流量應用(Variable-Rate Application, VRA)在精準農業的範疇中亦佔有一席之地，致力於在控管肥料及農藥等施噴以最大化作物產量。近年來隨著科技的發展，相關的研究與實際應用也隨之普及。Erickson和Widmar (2015)提出一份統計報告指出：於肥料的使用上VRA技術已有超過60%的農戶使用，且預測2018年時包含農藥、施肥與播種等VRA的使用皆會向上成長，顯示出該技術於農業之重要性。VRA於現今農業機械的應用中分為兩類：地圖基礎變流量應用(Map-based VRA)及感測器基礎變流量應用(Sensor-based VRA)，以下將針對此兩類變流量應用進行分析。

2.3.1 地圖基礎變流量應用



地圖基礎變流量應用(Map-Based VRA)是建立在已經建立田野的前提上而實行，使用前須先將田野資訊(如作物產量分布狀況、土壤狀況、地形等)蒐集完成後，依據位置不同使用定位系統控制噴藥裝置並隨位置調節流量，已完成在不同地方改變流量之目的。其中，全球定位系統(GPS)和地理資訊系統(GIS)的使用在此應用中佔相當重要的地位，各國研究也針對定位系統的使用不斷精進與改良。例如：Carrara等人(2004)使用差分全球定位系統(Differential Global Positioning System, DGPS)為主體設計了一款變流量施噴系統，透過電腦傳輸資料，減少了29%之除草劑使用；Mohammadzamani等人(2009)以GPS系統建立數位地圖並將土壤資料匯入後，應用變流量系統控制除草劑之施噴量，最後減少13%之農藥用量。

2.3.2 感測相關之變流量應用

若事先無田野調查之相關資訊，需要透過感測器即時監測並根據結果調整流量，則稱為感測相關之變流量應用(Sensor-based VRA)。其中，感測器的應用範圍大致可分為土壤資訊分析與作物之外觀分析，前者可根據土壤中含有之鹽分或農藥量進行監測；後者則針對作物之高度及顏色判斷其成熟度，例如：Holland和Schepers(2010)將氮含量之分析模型應用到變流量系統，除了節省成本外，也結合GIS系統以進行環境保育；Jeon等人(2010)利用超音波技術量測作物之最高點到土壤距離，藉此判斷作物之高度，且該技術不易受到外在環境(風速、溫度等)。其他地圖相關及感測相關之應用文獻統整如表2.5所示。

表2.5 變流量施噴應用之研究

變流量噴灑應用範疇		參考資料
Map-Based		
桿式噴藥車搭配GPS		曹智鈞和葉仲基, 2008
收割機搭配DGPS		Carrara et. al., 2004
桿式噴藥車搭配RTK-GPS		Esau et. al., 2015
無人機搭配GPS		Spoorthi et. al., 2017
粒狀肥料噴灑裝置搭配DGPS、GIS		Zaman et. al., 2005
Sensor-Based		
Plant Cover Sensor		Dammer and Ehlert, 2006
Ultrasonic Sensor		Jeon et. al., 2010

VRA除了噴灑裝置的改良外，田野調查的成果亦相當重要。Molin等人(2005)使用變流量噴灑裝置測試咖啡豆的栽培效果，發現使用變流量噴灑節省了23%的磷肥料且增加了34%的產量，也證實了這項應用的重要性。

2.4 Arduino簡介與農機相關應用

Arduino是一款單晶片微控制器，藉由程式語言編撰提供良好的互動環境並廣泛應用於不同領域中，例如冷氣的恆溫裝置與汽車的雷達系統，目前該程式編撰軟體以公開共享為基礎，可被應用於任和創作作品中，成為許多控制系統的熱門選擇。

而在農機的相關應用中，Kouchakzadeh及Beigzadeh(2015)為探討背負式噴霧器所產生的震動對使用者的影響，將使用者配戴的各部位加裝感應裝置進行量測並透過Arduino進行資料傳輸；Bugaje等人(2015)則將Arduino與Android app結合控制一台無人噴霧車，並做出試作品進行實際操作(圖2.3)。其他相關應用統整如表2.6所示。



圖2.3 無人噴霧車(Bugaje等人，2015)

表2.6 Arduino於農機之相關研究

農機應用簡介	參考資料
肥料施噴飛行器操控	Ajay et al., 2017
無人噴霧車操控	Bugaje et al., 2015
背負式噴霧機震動監測	Kouchakzadeh and Beigzadeh, 2015
小型果蠅控制器操控	Sarri et al., 2017

第三章 材料與方法



3.1 儀器及待測物

本研究中所使用之硬體與軟體部分，將分別以微控制器與混合系統介紹。

3.1.1 控制系統與通訊模組

本研究中所選用之微控制器為Arduino mega 2560(圖3.1)，和該系列入門版Arduino Uno相比，有更多的類比輸入/類比輸出腳位(mega類比輸入有16個，輸出15個；Uno類比輸入有6個，輸出6個)，且在訊號傳遞部分mega提供3組序列通訊接腳腳位，亦多於Uno的1組，在配線上提供更大的應用空間。

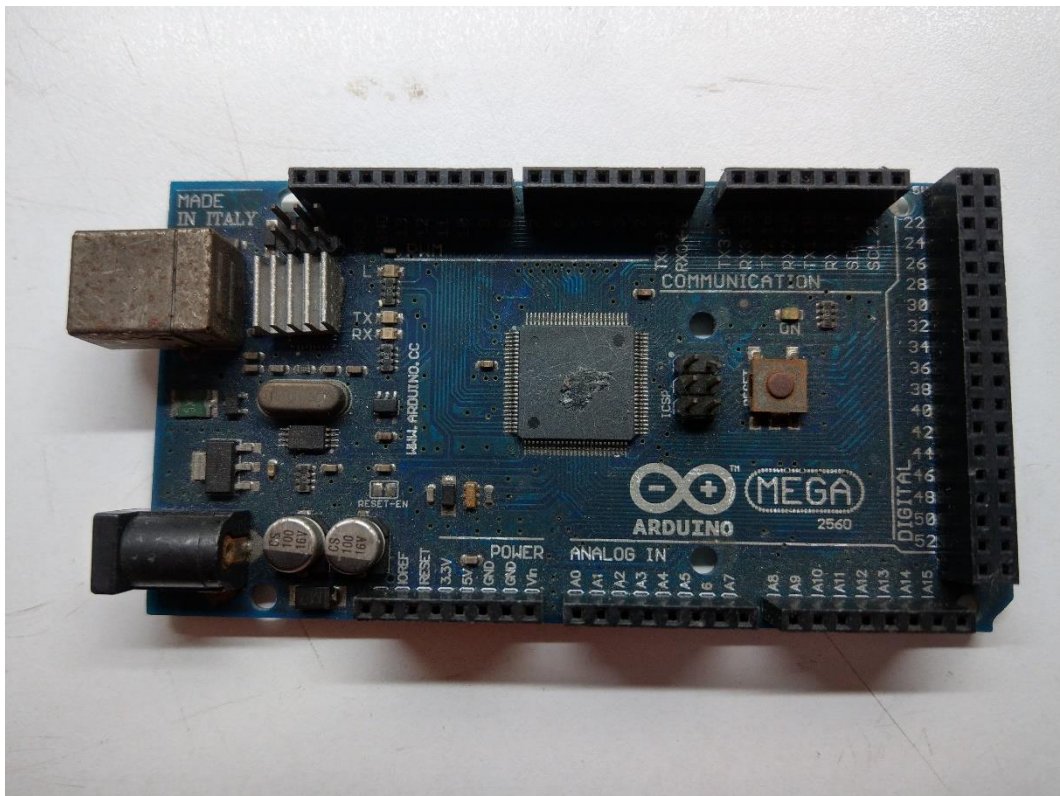


圖3.1 Arduino mega 2560微控制器

而關於Wifi通訊的部分，吾人選用ESP8266 Wifi晶片中之其中一個產品ESP-01(圖3.2)與Arduino對接並進行訊號傳遞，雖然Arduino之相關產品有和Wi-Fi做結合，但成本亦相對較高，故最終選用成本較低之ESP-01進行實驗。

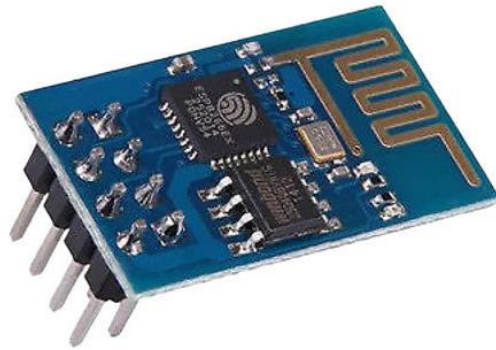


圖3.2 ESP-01 Wifi晶片

軟體部分，Arduino的程式編撰使用Arduino IDE(圖3.3)進行，大部分程式碼以C語言為基礎，而Wi-Fi晶片之通訊動作需由AT command進行設定，兩者皆可在同一編譯環境下進行。

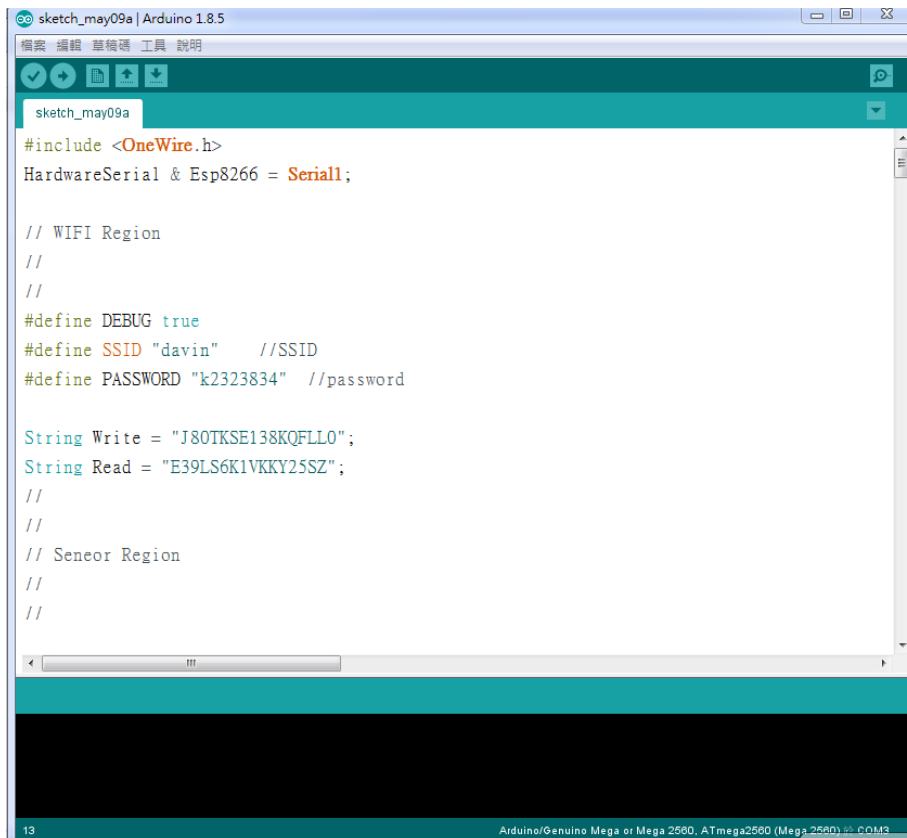


圖3.3 Arduino IDE介面

手機app部分，則使用App Inventor進行開發，主要是希望能透過一款app接收電導度值、閥開度與流量等資訊並進行控制，其介面如圖3.4所示。

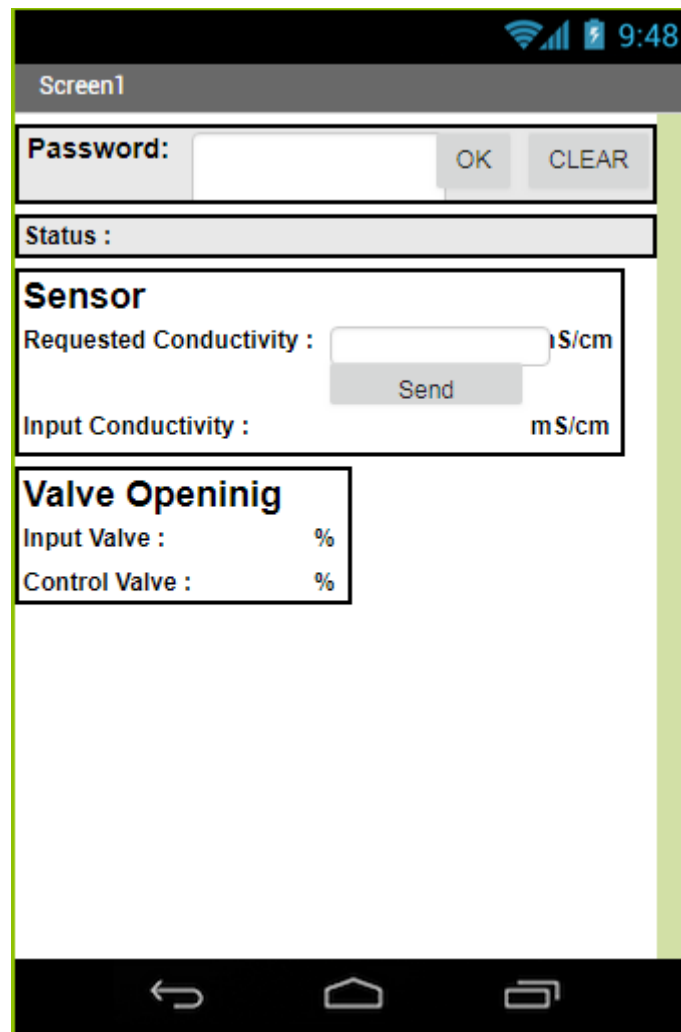


圖3.4 App介面(由Inventor開發)

3.1.2 混合系統

本研究中所做之混合系統由兩個儲水桶、兩個電動閥並聯而成，其中電動閥選用山野OM1電動閥(圖3.5)並搭配比例板控制閥開度，其相關資訊如表3.1所示。儲水桶及PVC管路皆由水電材料行購買。



圖3.5 山野電動閥外觀

表3.1 山野電動閥之產品參數

參數名稱	數值
最大扭力	35 N · m
重量	3 kg
電壓	110V AC
比例板	輸出訊號 4-20mA、2-10V
	輸入訊號 4-20 mA、1-5V、2-10V

本研究中之流量量測部分以自來水進行，流量計選用新政儀器製造之微型流量計(含顯示器)，其外觀如圖3.6所示。而溶液混合部分選用之植物營養液為同興牌農助旺作為稀釋用之液態肥料，其成分與稀釋比例資訊統整如表3.2所示。



圖3.6 流量計(右)與顯示器(左)

表3.2 農助旺植物營養液之成分資訊

品目名稱	混合鉀質肥料
登記成分	水溶性氧化鉀 12.0%
	鹽酸溶性氧化矽 20.0%
	水溶性硼 0.19%
	成分：硫酸鉀、硼酸、二氧化矽
用量	葉面噴灑稀釋1000-1200倍
	土壤灌注稀釋500倍。

3.2 實驗流程與方法



為整合閥開度、流量與稀釋比例等資訊，首先要先了解閥開度和流量之關係，並將其結果進行稀釋比例之換算後由電導度值驗證，最後整合App進行實作，整體流程如圖3.7所示。

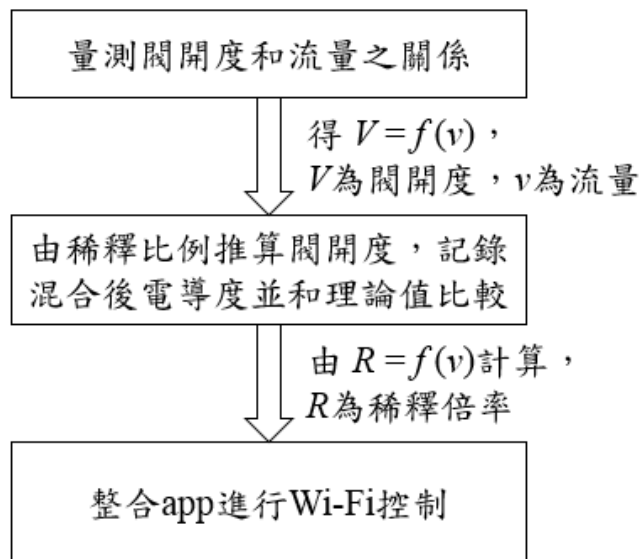


圖3.7 實驗流程

3.2.1 控制系統連接

控制系統由Arduino mega 2560進行控制，負責接收來自Wifi與各個閥體、電導度模組的資訊。app端可輸入出口電導度與流量，上傳到雲端資料庫Thingspeak後，由Arduino接收並進行其他閥體之調控。獲得回傳數值及電導度計等讀值時，將數據上傳到雲端資料庫Thingspeak，再由app透過雲端獲得資料，整個傳遞過程如圖3.8所示。

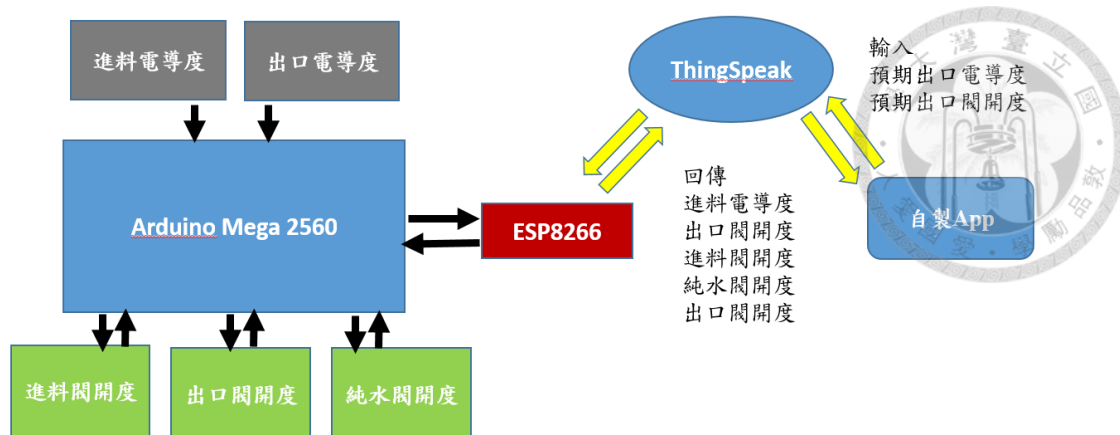


圖3.8 資料傳輸路徑

由於Arduino之類比訊號輸出及輸入皆為電壓值(0到5 V)，但電動閥之訊號輸出為2至10 V，故需串接兩個10 kΩ電阻降壓得到符合範圍之電壓值，整體系統之配線圖如圖3.9所示，電動閥之接線實體圖如3.10所示。

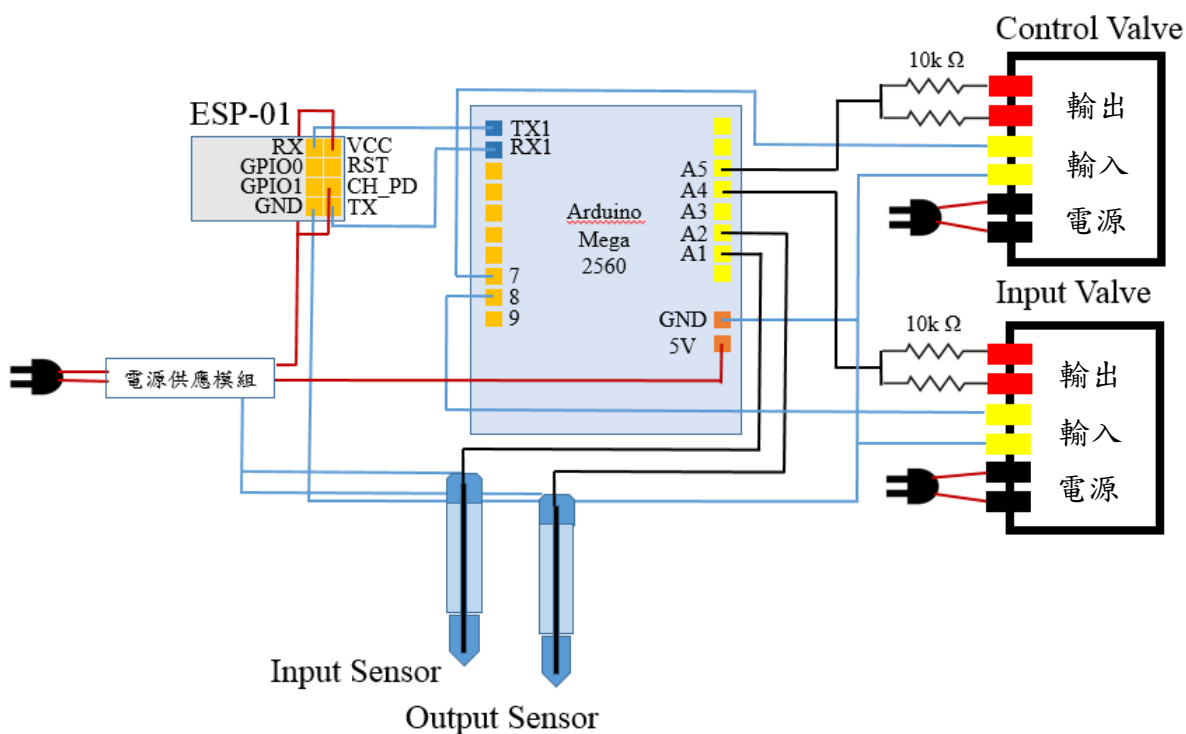


圖3.9 全系統線路連接

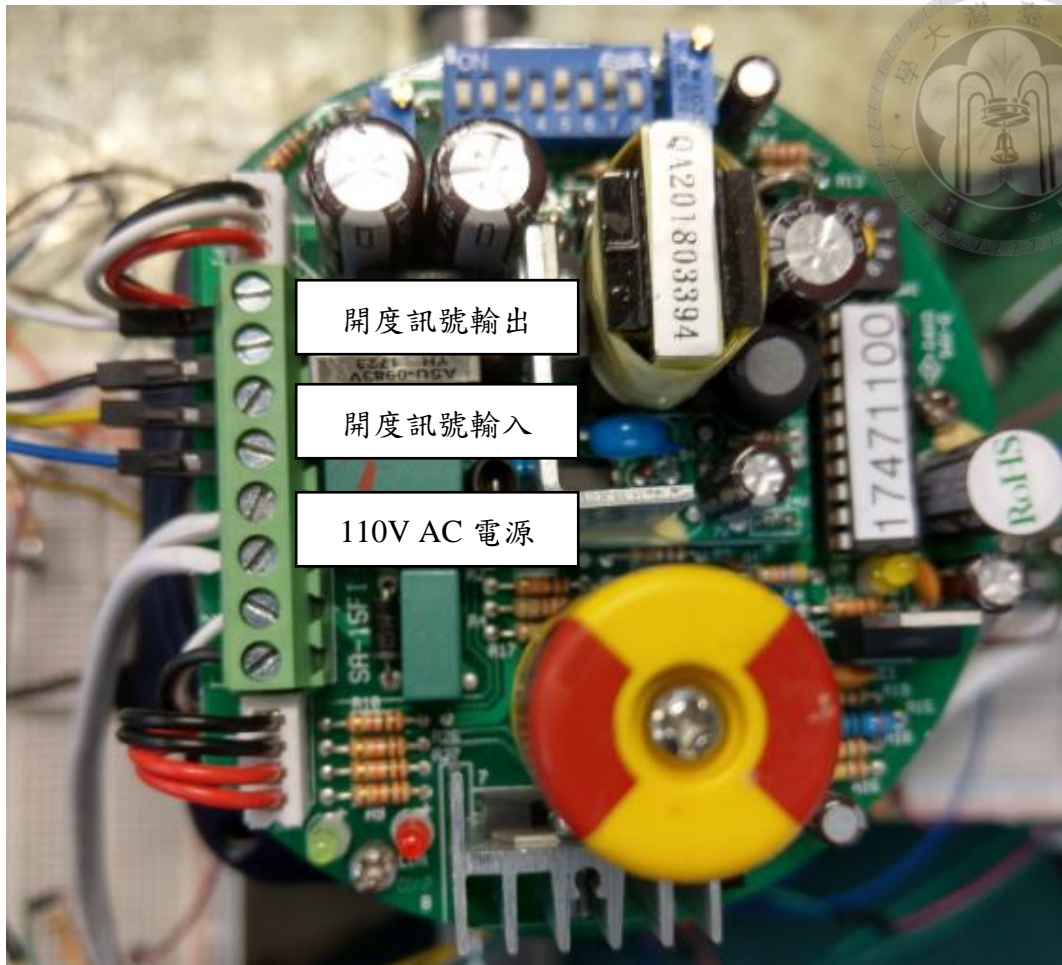


圖3.10 電動閥配線實體照

軟體部分，本研究使用Arduino IDE進行程式撰寫，由於ESP-01 Wi-Fi晶片並不適用Arduino內部的Wi-Fi程式庫，需使用AT command對ESP-01進行操作。本研究中應用到之AT command與指令如表3.3所示，其中<>內的訊息為使用者連接之Wi-Fi基地台及網站資訊。

表3.3 部分AT command指令與意義

指令	動作
AT + RST	重啟晶片
AT + CIOBAUD?	查詢目前鮑率
AT + CWMODE = 1	設定工作模式為網卡模式(連到現有Wifi)
AT+CWJAP=<SSID>,<Password>	連線到指定基地台
AT + CIPSTART = TCP, <IP>, <port>	建立和網站的TCP連線
AT + CIPSEND = <length>	發送數據

建立連線後，Arduino便得以和線上資料庫Thingspeak物聯網進行資料傳輸，該網站提供一個免費的平台給使用者進行資料傳輸，優點是上傳數據的便利性佳，僅需使用對應的API金鑰即可，缺點是每次資料的上傳間隔需等待15秒，對於需頻繁更新數據的使用者有所限制，而在Thingspeak網站上亦可直接讀取圖表示的資訊(圖3.11)。

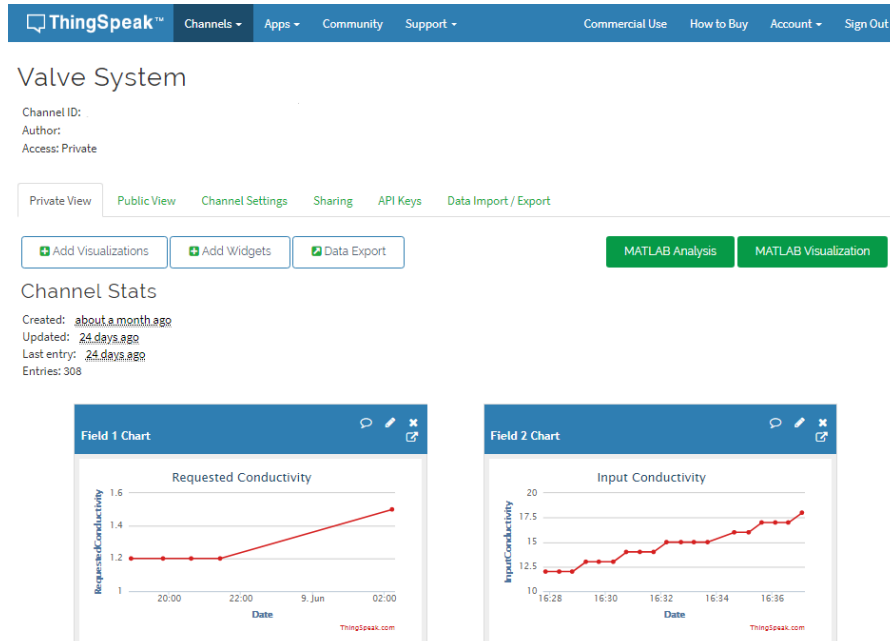


圖3.11 Thingspeak圖表介面

3.2.2 溶液混合系統

混合系統流速之討論，可由白努利定律(Bernoulli's law)和連續方程式進行說明。假設流體為穩態、不可壓縮且忽略摩擦並沿著流線移動，此時將槽底連接管路的部分標記為A點，閘門處標記為B點(圖4.1)，水槽內液面高度為 h ，若已知槽底流速為 v_A ，閘門之流速為 v_B ，可由白努利定律與連續方程式表示為：

$$\frac{1}{2} \rho v_A^2 + \rho g h_A + P_A = \frac{1}{2} \rho v_B^2 + \rho g h_B + P_B \quad (3.1)$$

$$A_A v_A = k A_B v_B \quad (3.2)$$

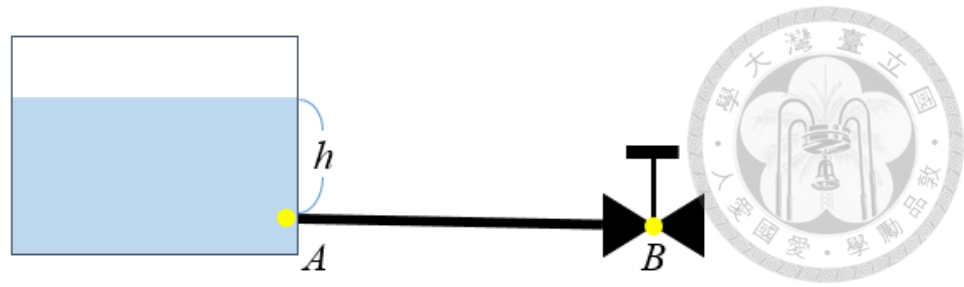


圖 3.12 水槽與電動閥連接示意圖

其中 ρ 為流體密度， h_A 、 h_B 分別為槽底與閥門之揚程， P_A 、 P_B 為槽底與閥門液體所受之壓力強度， A_A 、 A_B 為槽底連接管與閥門全開之截面積， g 為重力加速度， k 為閥開度。由於管路為水平設置（即 $h_A = h_B$ ），若忽略管路磨擦損失，將(2)式代入(1)式可得到：

$$v_A = \frac{1}{\sqrt{\frac{A_A}{kA_B} - 1}} \sqrt{\frac{2(P_A - P_B)}{\rho}} = \frac{1}{\sqrt{\frac{A_A}{kA_B} - 1}} \sqrt{\frac{2\Delta P}{\rho}} \quad (3.3)$$

其中 ΔP 表槽底與閥門之壓力差，由於壓力差主要來自於水槽內的液面高度，且槽底之管路截面積為定值，故將流速以流量表示如下：

$$Q = A_A v_A = \frac{A_A}{\sqrt{\frac{A_A}{kA_B} - 1}} \sqrt{2gh} \quad (3.4)$$

上式中 Q 為管內之流量， h 為水槽之液面高度，由此可推得流量和閥之截面積為正相關($Q \propto k$)，但流量亦受到下列兩因素影響：

1. 水槽內液體高度（實驗過程中液體高度會下降而非定值）
2. 管路的摩擦損失（一般管路、L型、T型與閥門皆有摩擦損失）

為探討溶液混合之流量比，需先建立電動閥開度與流量之關係。將一儲水桶接上一電動閥後，接上流量計(示意如圖3.12所示，實體如圖3.13所示)，注入4公升之自來水，而後調整電動閥開度為20%至100%間，開啟電動閥後記錄一分鐘、兩分鐘及三分鐘後的流量計讀值，並將兩者關係分別作圖討論。

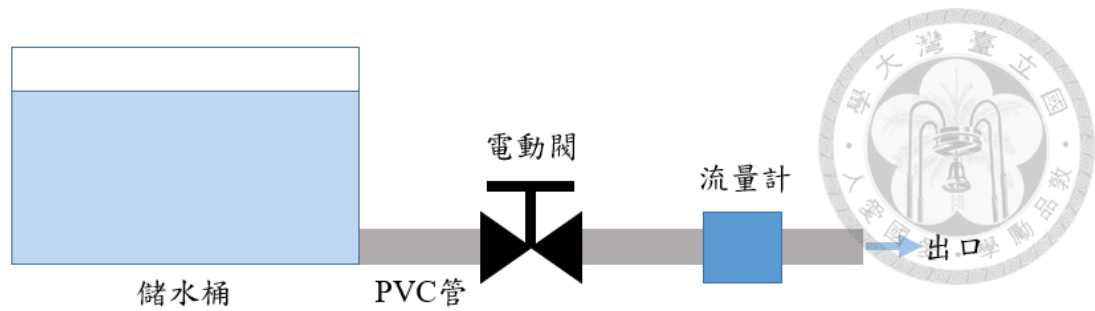


圖 3.13 流量測試系統連接示意圖

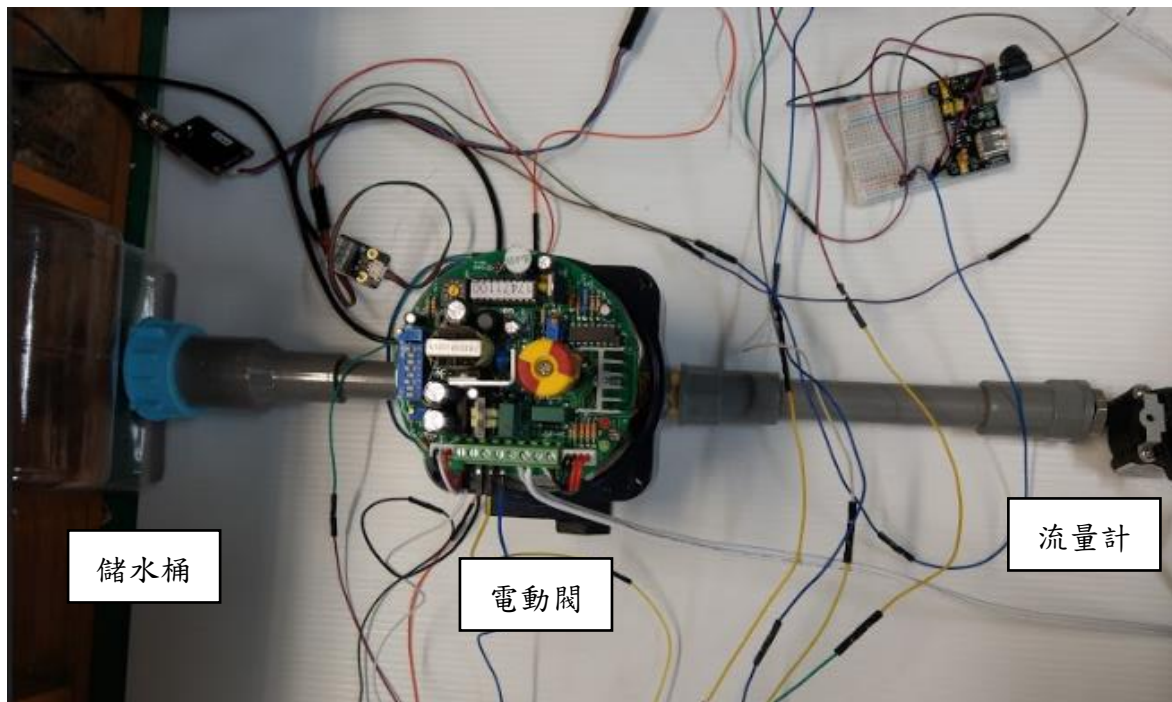
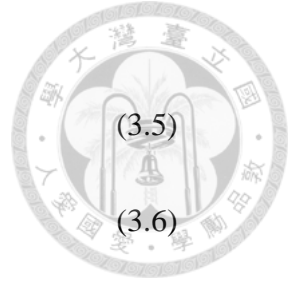


圖 3.14 流量測試系統連接實體照

確認流量與開度之關係後，將兩個儲水桶分別串接電動閥，再串接上T型接頭後導入第三個儲水桶(示意如圖3.14所示，實體如圖3.15與圖3.16所示)，且進料桶與出口端需裝上電導度計。為探討稀釋比例與閥開度之關係，先準備4公升稀釋100倍之農助旺溶液及4公升自來水，分別加入兩儲水桶後，調整進料與水之電動閥開度分別為1.5倍稀釋、1.75倍稀釋、2倍稀釋、2.5倍稀釋及3倍稀釋，並記錄一分鐘、兩分鐘及三分鐘後之出口端之電導度值。開度與稀釋倍率之關係由下列兩算式求得：



$$V_{input} + V_{water} = T \quad (3.5)$$

$$R = \frac{T}{V_{input}} \quad (3.6)$$

其中， R 為稀釋倍率， V_{input} 為進料閥開度， V_{water} 為自來水之閥開度， T 為混合後出口總流量值(定值，需由上一實驗決定)，在指定 R 值的情況可推算出 V_{input} 和 V_{output} 之值。

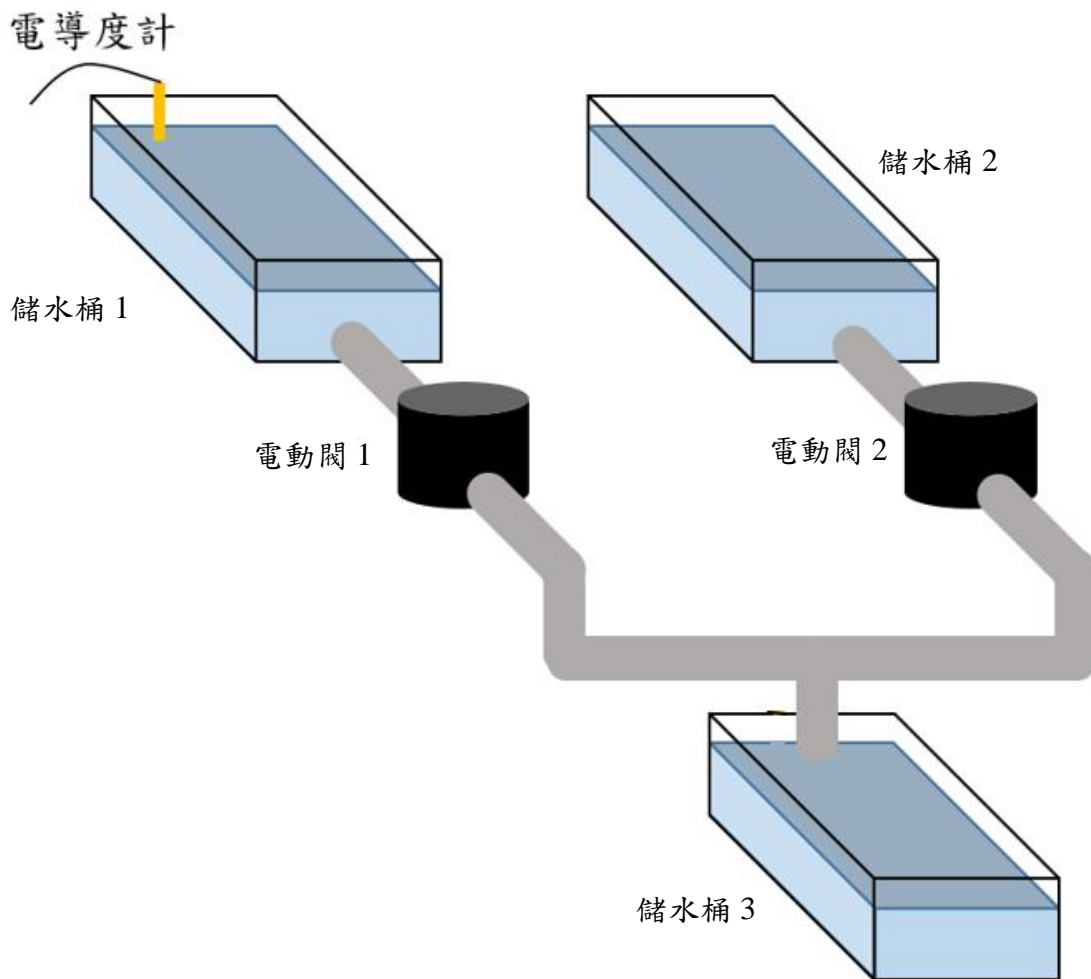


圖3.15 混合系統連接示意圖

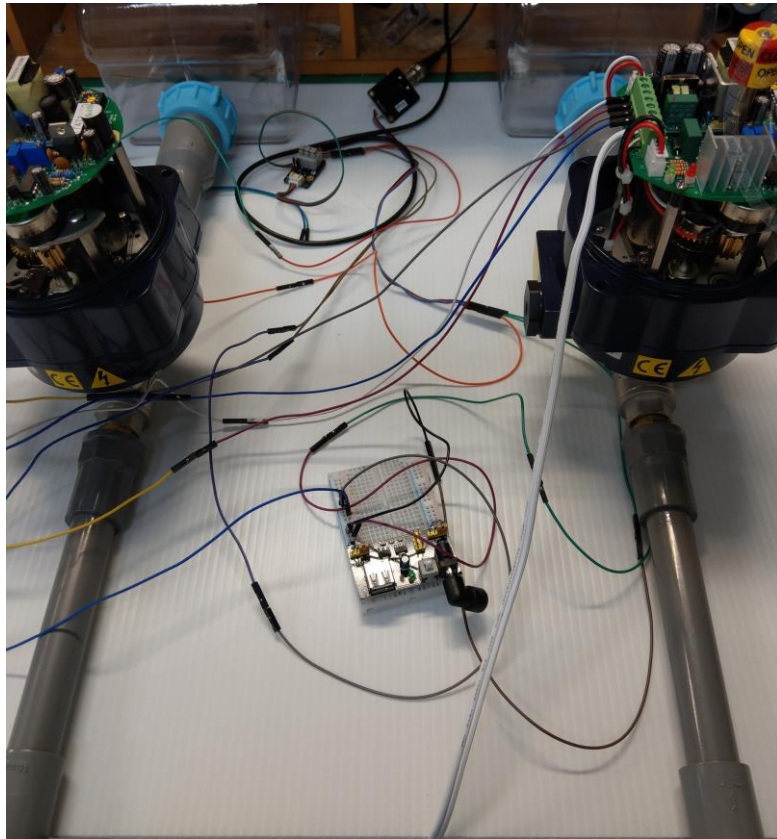


圖 3.16 混合系統連接實體照(一)

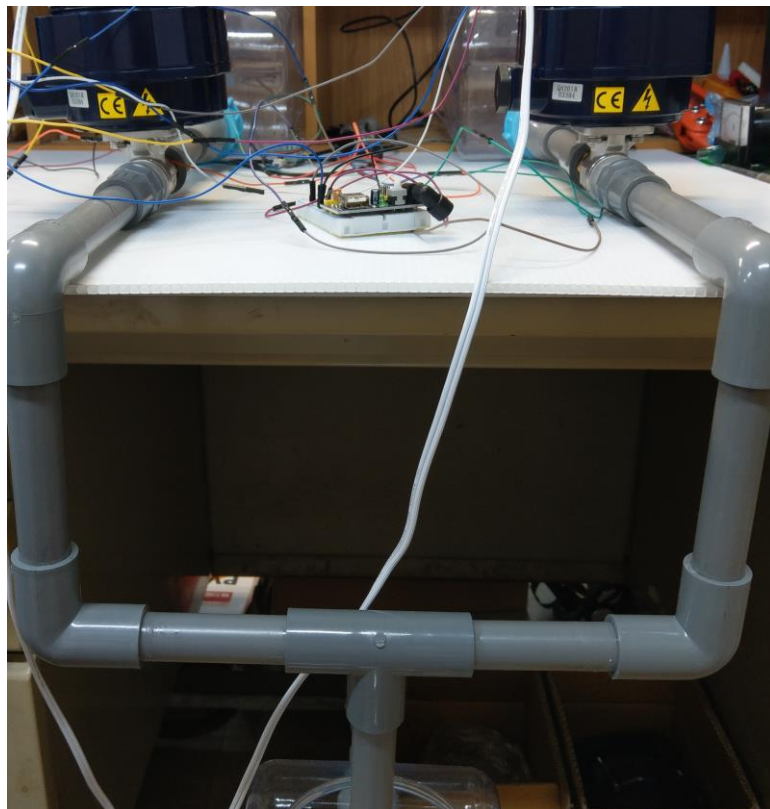


圖 3.17 混合系統連接實體照(二)

第四章 結果與討論



4.1 電動閥之閥開度與流量探討

實際應用中，噴藥機內的液面高度要控制在定值難度較高，原因在於流體一直噴灑時，槽內的液面必定會持續下降，且難以在噴藥的過程中不斷補充液體。另一修正此誤差的方式為在槽底出口端增加一小型幫浦，提供液面下降而減少的壓差，但液面下降量也會隨時間而改變，若非不斷調控幫浦，也很難將壓差控制在固定值，為此吾人並不直接從理論公式中得到流量 Q 及閥開度 k 的關係，而是探討線性函數的結果以利 Arduino 的運算，並為此分別做及流量 Q 與閥開後經過時間 t 以及 Q 與 k 的圖形，結果如圖 4.2、圖 4.3、圖 4.4 與圖 4.5 所示。

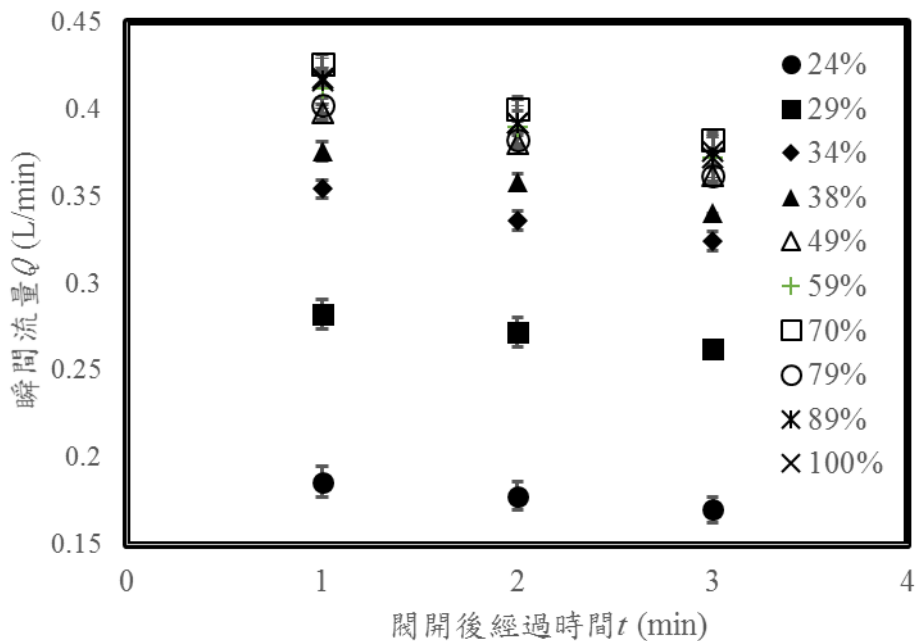


圖 4.1 流量與時間關係圖

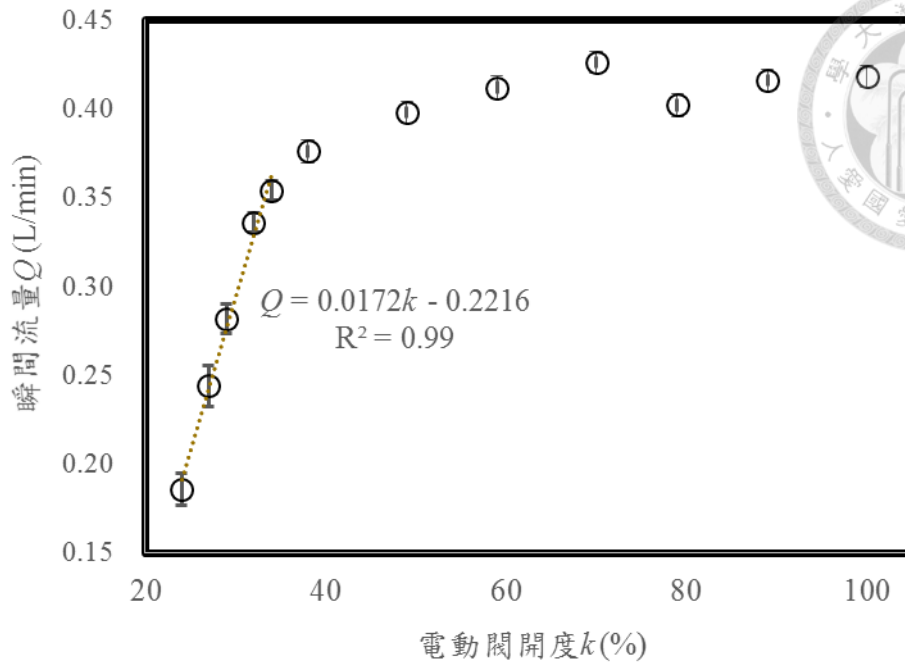


圖 4.2 一分鐘流量與閥開度關係圖

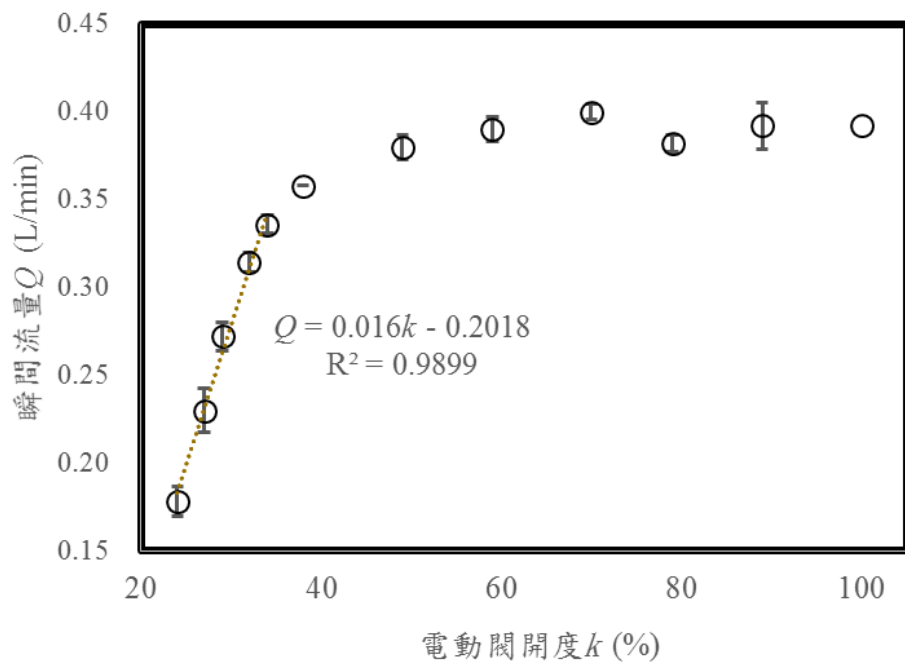


圖 4.3 兩分鐘流量與閥開度關係圖

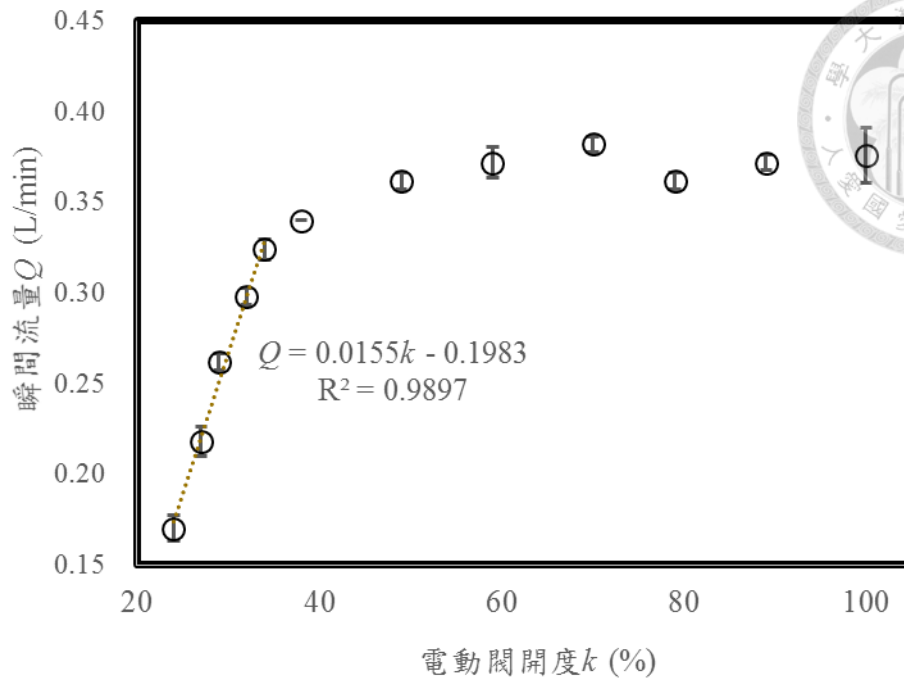


圖 4.4 三分鐘流量與閥開度關係圖

在圖 4.2 中，各開度下之流量值皆隨時間經過而下降，顯示槽內液體量減少時液面高度下降，導致壓差變小而導致流量變慢的情況，此現象符合實驗前的預期；而在圖 4.3 至圖 4.5 在固定時間(1 min、2 min 及 3 min)下討論流量與閥開度的關係，發現閥開度在 24% 至 34% 間和流量之關係有顯著的線性趨勢，而 34% 至 60% 間流量上升幅度逐漸減緩並於 60% 後趨近於定值，此結果和先前的文獻結果相同(廖等人，2003)，同時也顯示無論是否有幫浦存在，皆不會改變閥體對於流量趨勢之影響，代表此趨勢為閥體特徵。

此外，綜合閥開後時間、閥開度與流量三變數，即使閥開度有所不同，但在經過一樣時間後，其流量值仍有線性趨勢，為討論溶液稀釋倍率之問題，將以三個時間點之迴歸直線為基準，確定稀釋倍率並決定流量後推得閥開度，並討論閥開度固定的情況下，稀釋倍率與時間的關係。

4.2 溶液稀釋效益探討

根據 4.1 之實驗結果，閥開度在 24% 至 34% 調整對流量之影響為線性趨勢，此時流量範圍落在 0.17 L/min 至 0.34 L/min 間，由於在此區間的稀釋倍率最大值為 3 (進料流量 0.17 L/min，水流量 0.33 L/min)，最小值為 1.5 倍(即進料流量 0.33 L/min，水流量 0.17 L/min)，故以此為基準將混合後總流量和定為 0.5 L/min，即可得：

$$V_{input} + V_{water} = 0.5 \quad (4.1)$$

$$R_{ideal} = \frac{0.5}{V_{input}} \quad (4.2)$$

其中 V_{input} 為進料流量 V_{water} 為水流量， R_{ideal} 為流量決定之稀釋倍率。根據閥開度與流量之關係，可推得流量與對應閥開度如表 4.1 所示，由於流量較大者隨時間經過，其閥開度值之差異較大，為避免時間因素影響實驗進行，稀釋時閥開度之數值將以 1 min、2 min、3 min 之平均值進行。

表 4.1 溶液混合之稀釋倍率與進料、水之電動閥開度

稀釋 倍率(-)	流量值 (L/min)	一分鐘開度 ¹ (%)	兩分鐘開度 ² (%)	三分鐘開度 ³ (%)	開度平均 (%)	標準差 (%)	
1.50	進料	0.333	32.26	33.45	34.30	34.89	1.02
	水	0.167	22.57	23.03	23.55	22.08	0.49
1.75	進料	0.286	29.50	30.47	31.23	31.23	0.87
	水	0.214	25.34	26.01	26.62	25.74	0.64
2.00	進料	0.250	27.42	28.24	28.92	28.48	0.75
	水	0.250	27.42	28.24	28.92	28.48	0.75
2.50	進料	0.200	24.51	25.11	25.70	24.64	0.59
	水	0.300	30.33	31.36	32.15	32.33	0.91
3.00	進料	0.167	22.57	23.03	23.55	22.08	0.49
	水	0.333	32.26	33.45	34.30	34.89	1.02

註 1. 一分鐘閥開度之計算式為： $Q = 0.0172k - 0.2216$ ，其中 Q 為流量(L/min)， k 為閥開度(%)。

註 2. 兩分鐘閥開度之計算式為： $Q = 0.0160k - 0.2018$ 。

註 3. 三分鐘閥開度之計算式為： $Q = 0.0155k - 0.1983$ 。



溶液實際之稀釋倍率將由進口端與出口端之電導度進行計算，其計算方法如下：

$$R_{real} = \frac{EC_{input}}{EC_{output}} \quad (4.3)$$

其中 R_{real} 表示電導度決定之實際稀釋倍率， EC_{input} 、 EC_{output} 分別表示進口端與出口端之電導度值，若不考慮單一時間之電導度值而取平均值，統整後之結果如圖 4.6 所示。

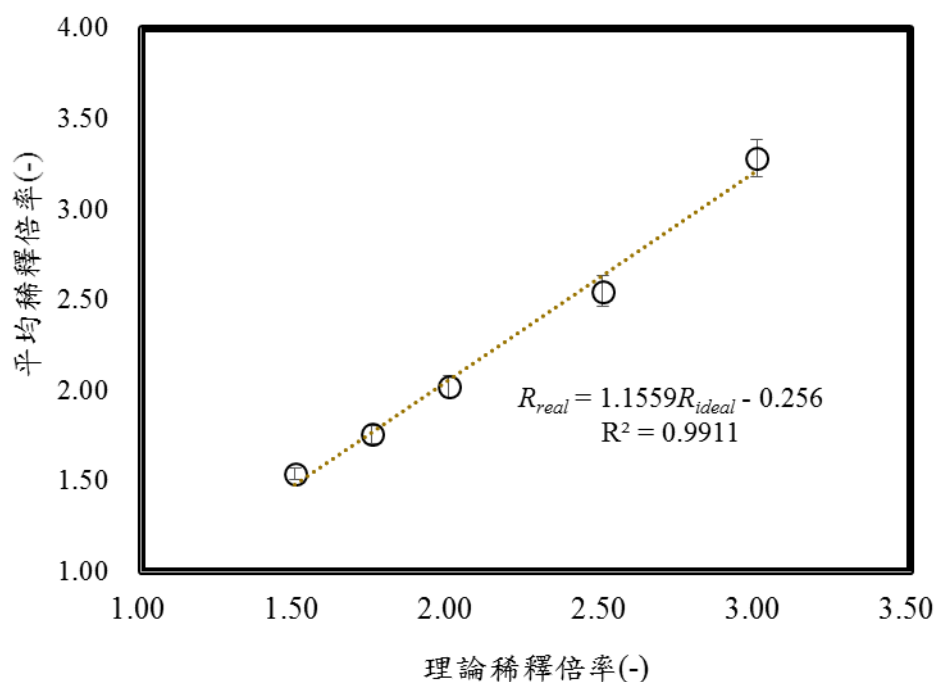


圖 4.5 理論稀釋倍率與實際稀釋倍率關係圖

圖中 R_{ideal} 表示理論稀釋倍率(即表 4.1 中的稀釋比)，若將任意時間之實際稀釋倍率進行統整，可發現理論值和實際值相去不遠，但整體誤差在稀釋倍率越大的情況下有越大的趨勢(稀釋倍率為 3.00 的狀況下較明顯)，代表此種溶液稀釋不宜用在過高的倍率(如 5 倍、10 倍及 20 倍等)。又針對不同時間點理論值與實際值之比較，其結果統整如表 4.2 至表 4.4 所示。



表 4.2 一分鐘稀釋倍率與理論值比較

稀釋倍率(-)	一分鐘稀釋倍率(-)	標準差(-)	誤差百分比(%)
1.50	1.53	0.06	3.87
1.75	1.83	0.10	5.32
2.00	2.03	0.12	5.98
2.50	2.55	0.02	0.62
3.00	3.33	0.55	16.64

表 4.3 兩分鐘稀釋倍率與理論值比較

稀釋倍率(-)	兩分鐘稀釋倍率(-)	標準差(-)	誤差百分比(%)
1.50	1.50	0.05	3.12
1.75	1.83	0.16	8.96
2.00	1.94	0.14	2.35
2.50	2.71	0.02	1.36
3.00	3.18	0.55	7.61

表 4.4 三分鐘稀釋倍率與理論值比較

稀釋倍率(-)	三分鐘稀釋倍率(-)	標準差(-)	誤差百分比(%)
1.50	1.47	0.05	3.38
1.75	1.80	0.16	6.67
2.00	1.97	0.14	3.74
2.50	2.67	0.02	7.41
3.00	3.09	0.55	8.52

由表 4.2 至表 4.4 可發現，相對於平均稀釋倍率，單一時間點的稀釋比例誤差較大，且並無固定趨勢，主要因為儲水桶並無攪拌裝置導致電導度不均，導致電導度起伏較大而測得瞬間之稀釋倍率誤差較大，若是進行實務操作，施噴裝置會附有迴流管，液體理應處於循環流動狀態而非以擴散為主，此時誤差便會下降。此外，由上一小節中可知在閥開度固定的情況下，流量也會有些微變化，亦是造成誤差的原因之一，故此系統雖然準確性高，但誤差較大，除了增加溶液攪拌外，即時調控閥開度亦會是可考慮的選擇。

4.3 App 系統整合

Arduino 程式碼與 App 開發磚塊圖如附件一與附件二所示，由 App 輸入一指定的出口電導度值後(圖 4.6)，該筆數據會上傳到雲端資料庫 Thingspeak(圖 4.7)，並由 Arduino 讀取，Arduino 便會計算稀釋倍率，並由圖 4.5 進行校正後再由式 4.5 及式 4.6 反推流量及開度。

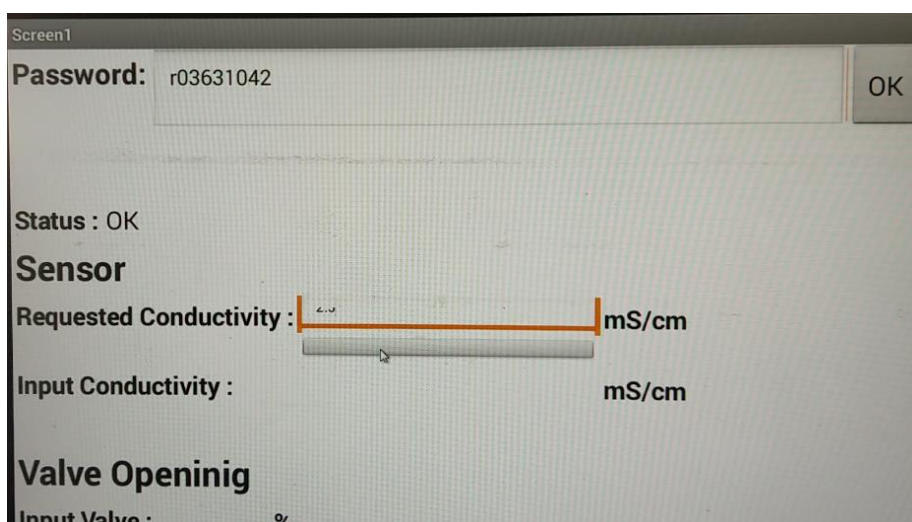


圖 4.6 App 端電導度值輸入

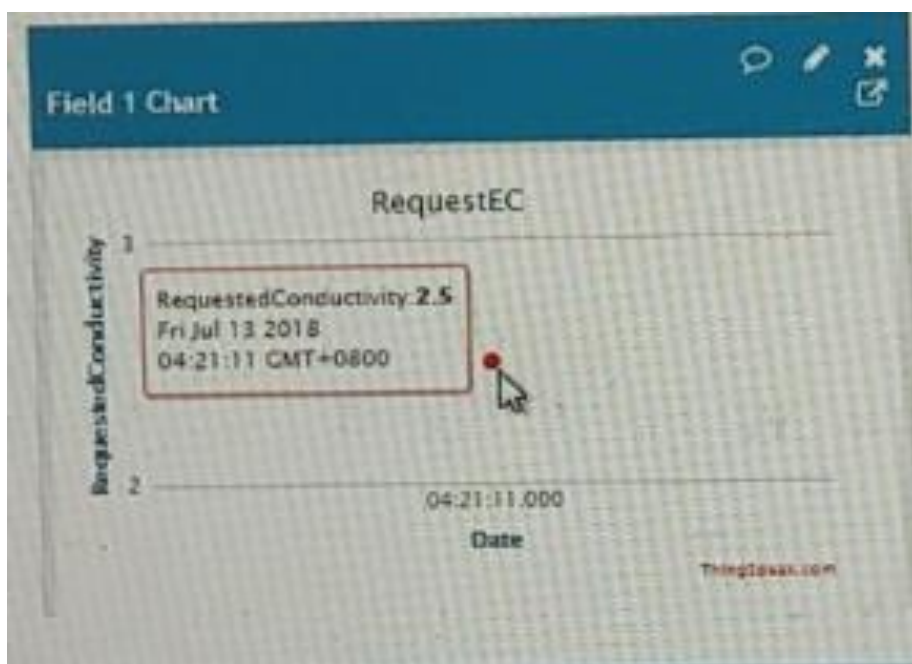


圖 4.7 Thingspeak 輸入值接收

經計算所得之閥開度會由 PWM 訊號傳送至電動閥，而電動閥接收後會將閥調整至指定的開度值，並回傳當前開度資訊至 Arduino 並連同進料之電導度值上傳到 Thingspeak(圖 4.8)，最後閥開度資訊與進料電導度值便會直接呈現在 App 端(圖 4.9)，以實現遠端遙控的目的。

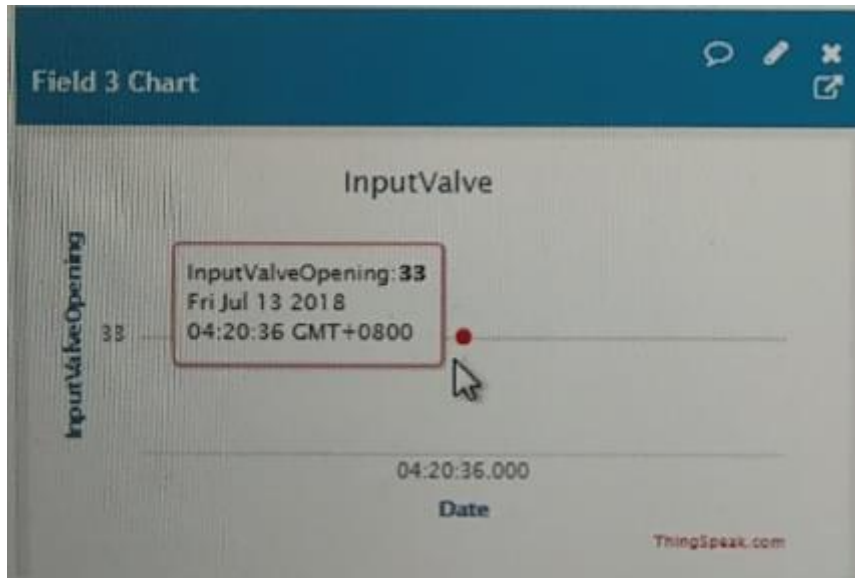


圖 4.8 Thingspeak 回傳值接收

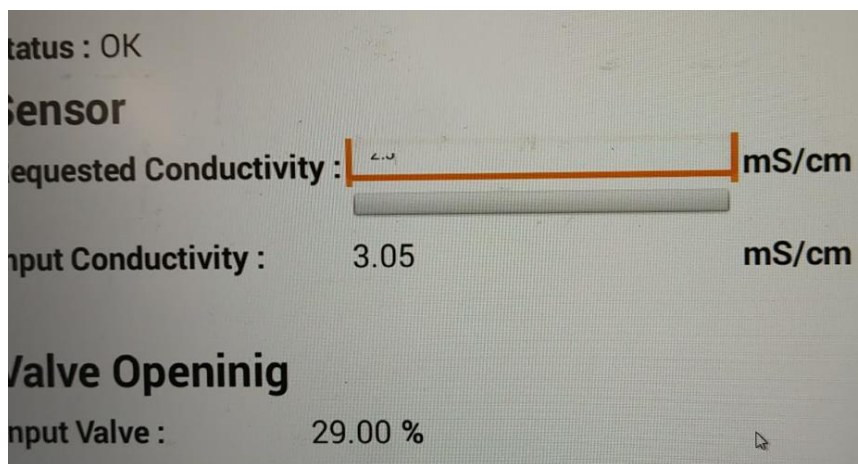


圖 4.9 App 端回傳值接收

第五章 結論與建議



本論文旨在開發一個溶液混合的系統以調整濃度或電導度，在多數文獻探討噴霧機壓力、噴嘴等變流量系統的同時，藉由此研究開發之系統能夠有效避免來自溶液配置的錯誤而造成作物之損傷。

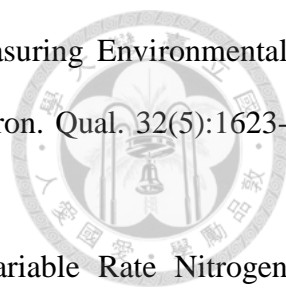
在流量與閥開度的討論中，發現閥開度與流量在 24% 至 34% 呈線性趨勢，此現象和前人試驗結果一致。但本篇研究所得之流量屬小流量(0.5 L/min)，若所應用於動力噴霧機流量較大(30 L/min、50 L/min)的話，除加大管徑外，亦可加上幫浦提升壓降而增加流量。另一方面，由於近期無人機施噴的應用逐漸崛起，若要將變量系統帶到無人機上，就必須考慮裝置之重量。其中，本篇研究所使用的兩顆電動閥為 6 公斤，若含 4L 之水及 4L 之進料達 14 公斤，對於一台載重 20 公斤的農用無人機而言有發展的空間，但受限於臺灣對於無人機的限制目前未能實行，但未來這也是可以發展的方向。

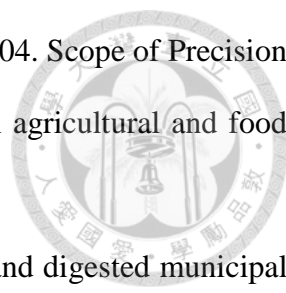
另外，本篇研究亦對於開度與稀釋比例之探討，電動閥相對於電磁閥在調控上具有優勢，故其準確性得以顯現，然而，混合過程中仍需要進行攪拌，才能得到相對精確的數值。為此，若後續有相關之動態測試，可另行比較兩者數值之差異以確認準確度。

針對 app 及 Wi-Fi 控制的部分，ESP-01 晶片發揮龐大的效果，不但能節省成本且功能俱全，唯一的缺點在於通訊設定的命令集和 Arduino 不同，但仍推薦使用 ESP-01 晶片進行更多 Wi-Fi 通訊與互動的實作，以達成遠端操控的目的。

參考文獻

1. 林俊義、廖慶樑、王鐘和編。2002。作物有機栽培102號。行政院農業委員會農業試驗所。臺中。第59-70頁。
2. 曹智鈞、葉仲基。2008。桿式噴藥車變率施噴系統搭配GPS 之實測。台灣農學會報 9(6): 589-603。
3. 黃子彬。1996。宜蘭地區夏季葉菜類設施栽培。花蓮區農業專訊16(3): 4-9。
4. 農藥殘留容許量標準 (民國97年10月21日)。
5. 廖盈達、葉仲基、梁連勝。2003。GPS自動噴藥機之開發-變率施噴系統基礎研究。水稻精準農業體系: 219-230。
6. 蔣永正。2005。市售植物營養劑活性之生物檢測。中華民國雜草學會會刊 26(1): 53-66。
7. 食品衛生法第十一條(2005年11月29日)。
8. Amending Regulation (EC) No 396/2005 of the European Parliament and of the Council as regards Annexes II, III and IV on maximum residue levels of pesticides in or on certain products (2008.08.30).
9. Bongiovanni R. 2004. Precision Agriculture and Sustainability. *Precis. Agric.* 5(4): 359–387.
10. Carrara M., A. Comparetti, P. Febo and S. Orlando. 2004. Spatially Variable Rate Herbicide Application on Durum Wheat in Sicily. *Biosyst. Eng.* 87(4): 387-392.
11. Dammer K. H. and D. Ehlert. 2006. Variable-rate fungicide spraying in cereals using a plant cover sensor. *Precis. Agric.* 7(2): 137-148.
12. Erickson B. and D. A. Widmar. 2015. Precision Agricultural Services Dealership Survey Results. US: CropLife.
13. Esau T. J., Q. U. Zaman, Y. K. Chang, A. W. Schumann and D. C. Percival, A. A. Farooque. 2013. Spot-application of Fungicide for Wild Blueberry Using an Automated Prototype Variable Rate Sprayer. *Precis. Agric.* 15(2): 147-161.

- 
14. Ferrari F., M. Trevisan and E. Capri. 2003. Predicting and Measuring Environmental Concentration of Pesticides in Air after Soil Application. *J. Environ. Qual.* 32(5):1623-1633.
 15. Holland K. H. and J. S. Schepers. 2010. Derivation of a Variable Rate Nitrogen Application Model for In-Season Fertilization of Corn. *Agron J.* 102(5): 1415-1424.
 16. Jambaldorj G., M. Takahashi and K. Yoshikawa. 2007. Liquid Fertilizer Production from Sewage Sludge by Hydrothermal Treatment. Paper presented at International Symposium on EcoTopia Science, Japan.
 17. Jeon H. Y., H. Zhu, R. Derksen, E. Ozkan and C. Krause. 2011. Evaluation of ultrasonic sensor for variable-rate spray applications. *Comput. Electron. Agr.* 75(1): 213-221.
 18. Kalbani F. O. S. A., M. A. Salem, A. J. Cheruth, S. S. Kurup and A. Senthilkumar. 2016. Effect of some organic fertilizers on growth, yield and quality of tomato (*Solanum lycopersicum*). *Int. Lett. Nat. Sci.* 53: 1-9.
 19. Klein M., M. MuÈller, M. Dust, G. GoÈrlitz, B. GottesbuÈren, J. Hassink, R. Kloskowski, R. Kubiak, H. Ressler, H. SchaÈfer, B. Stein and H. Vereecken. 1997. Validation of the pesticide leaching model PELMO using lysimeter studies performed for registration. *Chemosphere* 35(11): 2563-2587.
 20. Mohammadzamani D., S. Minaei, R. Alimardani, M. Almassi, M. Rashidi and H. Norouzpour. 2009. Variable Rate Herbicide Application Using the Global Positioning System for Generating a Digital Management Map. *Int. J. Agric. Biol.* 11(2): 178-182.
 21. Molin J. P., A. V. de Araujo Motomiya, F. R. Frasson, G. Di Chiacchio Faulin and W. Tosta. 2010. Test procedure for variable rate fertilizer on coffee. *Acta Sci., Agron.* 32(4): 569-575.

- 
22. Mondal P., V. K. Tewari, P. N. Rao, R. B. Verma and M. Basu. 2004. Scope of Precision Agriculture in India. Paper presented at emerging technologies in agricultural and food engineering, India.
 23. Romero C., P. Ramos, C. Costa and M. C. Márquez. 2013. Raw and digested municipal waste compost leachate as potential fertilizer: comparison with a commercial fertilizer. *J. Clean Prod.* 59: 73-78.
 24. Sa´nchez M. and J. L. Gonza´lez. 2005. The fertilizer value of pig slurry. I. Values depending on the type of operation. *Bioresour. Technol.* 96(10): 1117-1123.
 25. Spoorthi.S, B. Shadaksharappa, Suraj. S. and V. K. Manasa. 2017. Freyr Drone: Pesticide/Fertilizers Spraying Drone – an Agriculture Approach. Paper presented at International Conference on Computing and Communications Technologies, India.
 26. Tolerances and Exemptions for Pesticide Chemical Residues in Food, 40 C.F.R. 180§ 1971.
 27. Yallappa D., M. Veerangouda, D. Maski, V. Palled and M. Bheemanna. 2017. Development and evaluation of drone mounted sprayer for pesticide applications to crops. Paper presented at Global Humanitarian Technology Conference, USA.
 28. Zaman Q. U., A. W. Schumann and W. M. Miller. 2005. Variable Rate Nitrogen Application in Florida Citrus Based on Ultrasonically-Sensed Tree Size. *Appl. Eng. Agric.* 21(3): 331–335.

附件一 Arduino與App互動之程式碼



```
#include <ArduinoJson.h>

#include <OneWire.h>
HardwareSerial & Esp8266 = Serial1;

// WIFI Region
//
//
#define DEBUG true
#define SSID "davin" //SSID
#define PASSWORD "k2323834" //password

String Write = "J80TKSE138KQFLL0";
String Read = "E39LS6K1VKKY25SZ";
//
//
// Sensor Region
//
//
#define StartConvert 0
#define ReadTemperature 1

const byte numReadings = 20; //the number of sample times
byte ECsensorPin = A1; //EC Meter analog output,pin on analog 1
byte DS18B20_Pin = 2; //DS18B20 signal, pin on digital 2
unsigned int AnalogSampleInterval=25,printInterval=700,tempSampleInterval=850;
//analog sample interval;serial print interval;temperature sample interval
unsigned int readings[numReadings]; // the readings from the analog input
byte index = 0; // the index of the current reading
unsigned long AnalogValueTotal = 0; // the running total
unsigned int AnalogAverage = 0,averageVoltage=0; // the average
unsigned long AnalogSampleTime,printTime,tempSampleTime;
float temperature,ECcurrent;

//Temperature chip i/o
OneWire ds(DS18B20_Pin); // on digital pin 2
```



```
//
//
// Valve Region
//
//
const int ValveInReturnPin = A3; // Input Valve
const int ValveControlReturnPin = A5; // Water Valve
const int ValveInPin = 7; // Input Valve
const int ValveControlPin = 9; // Input Valve

float ValveIn = 1;           // input valve open percentage
int ValveInValue = 1;        // input valve
float ValveOut = 100;        // Output valve open percentage
int ValveOutValue = 1;       // Output valve
float ValveCtrl = 1;         // Control valve open percentage
int ValveCtrlValue = 1;      // Control valve
float DiluteRatio = 1;       // Dilute ratio

float ValveInReturn = 1;
float ValveOutReturn = 1;
float ValveControlReturn = 1;
float field1=0;
float SetValue=1.5;
//
//
float TempProcess(bool ch)
{
    //returns the temperature from one DS18B20 in DEG Celsius
    static byte data[12];
    static byte addr[8];
    static float TemperatureSum;
    if(!ch){
        if ( !ds.search(addr)) {
            Serial.println("no more sensors on chain, reset search!");
            ds.reset_search();
            return 0;
        }
        if ( OneWire::crc8( addr, 7) != addr[7]) {
```

```

        Serial.println("CRC is not valid!");
        return 0;
    }
    if ( addr[0] != 0x10 && addr[0] != 0x28) {
        Serial.print("Device is not recognized!");
        return 0;
    }
    ds.reset();
    ds.select(addr);
    ds.write(0x44,1); // start conversion, with parasite power on at the end
}
else{
    byte present = ds.reset();
    ds.select(addr);
    ds.write(0xBE); // Read Scratchpad
    for (int i = 0; i < 9; i++) { // we need 9 bytes
        data[i] = ds.read();
    }
    ds.reset_search();
    byte MSB = data[1];
    byte LSB = data[0];
    float tempRead = ((MSB << 8) | LSB); //using two's compliment
    TemperatureSum = tempRead / 16;
}
return TemperatureSum;
}
//
//
//
boolean connectWiFi()
{
    Serial1.println("AT+CWMODE=1");
    Wifi_connect();
}
//
//
void Wifi_connect()

```





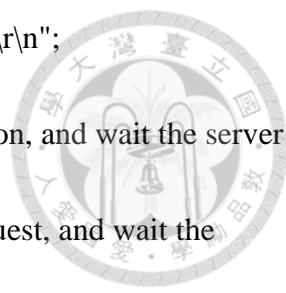
```
{
    String cmd="AT+CWJAP=\"";
    cmd+=SSID;
    cmd+="\", \"";
    cmd+=PASSWORD;
    cmd+="\"";
    sendDebug(cmd);
    Loding("Wifi_connect");
}
//
//
void Loding(String state){
    for (int timeout=0 ; timeout<10 ; timeout++)
    {
        if(Serial1.find("OK"))
        {
            Serial.println("RECEIVED: OK");
            break;
        }
        else if(timeout==9){
            Serial.print( state );
            Serial.println(" fail...\nExit2");
        }
        else
        {
            Serial.print("Wifi Loading...");
            delay(500);
        }
    }
}
void sendDebug(String cmd)
{
    Serial.print("SEND: ");
    Serial.println(cmd);
    Serial1.println(cmd);
}
//
//
```

```

void sendATcmd(char* cmd, unsigned int msDelay, bool dbg) {
    String espMsg="";
    unsigned long timeout = millis()+msDelay;
    Serial.print(">>ESP-01: "+String(cmd)); // debug
    Serial1.print(cmd); // send AT command to ESP-01
    while ( Serial1.available() || millis()<timeout ) {
        while ( Serial1.available() ) {
            espMsg += (char)Serial1.read();
        }
    }
    if ( dbg ) {
        Serial.print(String("<<ESP-01: ") + espMsg);
    }
}
//
//
// query data from ThingSpeak
void queryThingSpeak(char* ipName, char* channelName, String apiKey, float X) {
    String tcpCmd = "AT+CIPSTART=\"TCP\", \"" + String(ipName) + "\",80\r\n";
    String getCmd = "GET /channels/" + String(channelName) +
        "/fields/1/last.json?api_key=" + String(apiKey) +
"&timezone=Asia/Taipei" + "\r\n";
    String cipCmd = "AT+CIPSEND=" + String(getCmd.length()) + "\r\n";
    sendATcmd(tcpCmd.c_str(), 2000, DEBUG); // ask HTTP connection, and wait
the server to ack "OK"
    sendATcmd(cipCmd.c_str(), 2000, DEBUG); // start CIPSEND request, and wait
the server to ack "OK"
    sendATcmd(getCmd.c_str(), 2000, DEBUG); // send GET command
}
//
//
// update data to ThingSpeak
void updateThingSpeak(char* ipName, String apiKey, float inputconductivity, float
inputvalve, float ctrlvalve) {
    String tcpCmd = "AT+CIPSTART=\"TCP\", \"" + String(ipName) + "\",80\r\n";
    String getCmd = "GET /update?key=" + String(apiKey) + "&field2=" +
String(inputconductivity) + "&field3=" + String(inputvalve) + "&field4=" +
String(ctrlvalve) + "\r\n";
}

```





```

String cipCmd = "AT+CIPSEND=" + String(getCmd.length()) + "\r\n";

sendATcmd(tcpCmd.c_str(),2000,DEBUG); // ask HTTP connection, and wait the server
to ack "OK"
sendATcmd(cipCmd.c_str(),2000,DEBUG); // start CIPSEND request, and wait the
server to ack "OK"
sendATcmd(getCmd.c_str(),2000,DEBUG);
}
//
//
void setup() {
// initialize serial communication with computer:
Serial.begin(115200);
// initialize all the readings to 0:
for (byte thisReading = 0; thisReading < numReadings; thisReading++)
readings[thisReading] = 0;
TempProcess(StartConvert); //let the DS18B20 start the convert
AnalogSampleTime=millis();
printTime=millis();
tempSampleTime=millis();
//
//
Esp8266.begin(9600);
sendDebug("AT");
Loding("sent AT");
connectWiFi();
//
//
}

void loop() {
/*
Every once in a while,sample the analog value and calculate the average.
*/
if(millis()-AnalogSampleTime>=AnalogSampleInterval)
{
AnalogSampleTime=millis();

```



```
// subtract the last reading:
AnalogValueTotal = AnalogValueTotal - readings[index];
// read from the sensor:
readings[index] = analogRead(ECsensorPin);
// add the reading to the total:
AnalogValueTotal = AnalogValueTotal + readings[index];
// advance to the next position in the array:
index = index + 1;
// if we're at the end of the array...
if (index >= numReadings)
// ...wrap around to the beginning:
index = 0;
// calculate the average:
AnalogAverage = AnalogValueTotal / numReadings;
}
/*
```

Every once in a while,MCU read the temperature from the DS18B20 and then let the DS18B20 start the convert.

Attention:The interval between start the convert and read the temperature should be greater than 750 millisecond,or the temperature is not accurate!

```
*/
if(millis()-tempSampleTime>=tempSampleInterval)
{
tempSampleTime=millis();
temperature = TempProcess(ReadTemperature); // read the current temperature from
the DS18B20
TempProcess(StartConvert); //after the reading,start the convert
for next reading
}
/*
```

Every once in a while,print the information on the serial monitor.

```
*/
if(millis()-printTime>=printInterval)
{
printTime=millis();
averageVoltage=AnalogAverage*(float)5000/1024;
Serial.print("Analog value:");
Serial.print(AnalogAverage); //analog average,from 0 to 1023
```

```

Serial.print("    Voltage:");
Serial.print(averageVoltage); //millivolt average,from 0mv to 4995mV
Serial.print("mV    ");
Serial.print("temp:");
Serial.print(temperature); //current temperature
Serial.print("^C    EC:");

```



```

float TempCoefficient=1.0+0.0185*(temperature-25.0); //temperature compensation
formula: fFinalResult(25^C) = fFinalResult(current)/(1.0+0.0185*(fTP-25.0));
float CoefficientVolatge=(float)averageVoltage/TempCoefficient;
if(CoefficientVolatge<150)Serial.println("No solution!"); //25^C 1413us/cm<--
>about 216mv if the voltage(compensate)<150,that is <1ms/cm,out of the range
else if(CoefficientVolatge>3300)Serial.println("Out of the range!"); //>20ms/cm,out of
the range
else
{
    if(CoefficientVolatge<=448)ECcurrent=6.84*CoefficientVolatge-64.32;
//1ms/cm<EC<=3ms/cm
    else if(CoefficientVolatge<=1457)ECcurrent=6.98*CoefficientVolatge-127;
//3ms/cm<EC<=10ms/cm
    else ECcurrent=5.3*CoefficientVolatge+2278;
//10ms/cm<EC<20ms/cm
    ECcurrent/=1000; //convert us/cm to ms/cm
    Serial.print(ECcurrent,2); //two decimal
    Serial.println("ms/cm");
}
}
//
//
//Valve Region
//
//
queryThingSpeak("184.106.153.149", 496357, Read, SetValue);
Serial.println(SetValue);
float DiluteRatio = ECcurrent/SetValue;
if(DiluteRatio<1.5)
{
    DiluteRatio=1.5;

```



```

}

ValveIn=1.8534*((0.5/DiluteRatio+0.2018)/0.016)+61.337;
ValveCtrl=1.8534*((0.5*(1-1/DiluteRatio)+0.2018)/0.016)+61.337;

analogWrite(ValveInPin,ValveIn);
analogWrite(ValveControlPin,ValveCtrl);

ValveInReturn = map(analogRead(ValveInReturnPin),204.6,1023,0,100);
ValveControlReturn = map(analogRead(ValveControlReturnPin),204.6,1023,0,100);

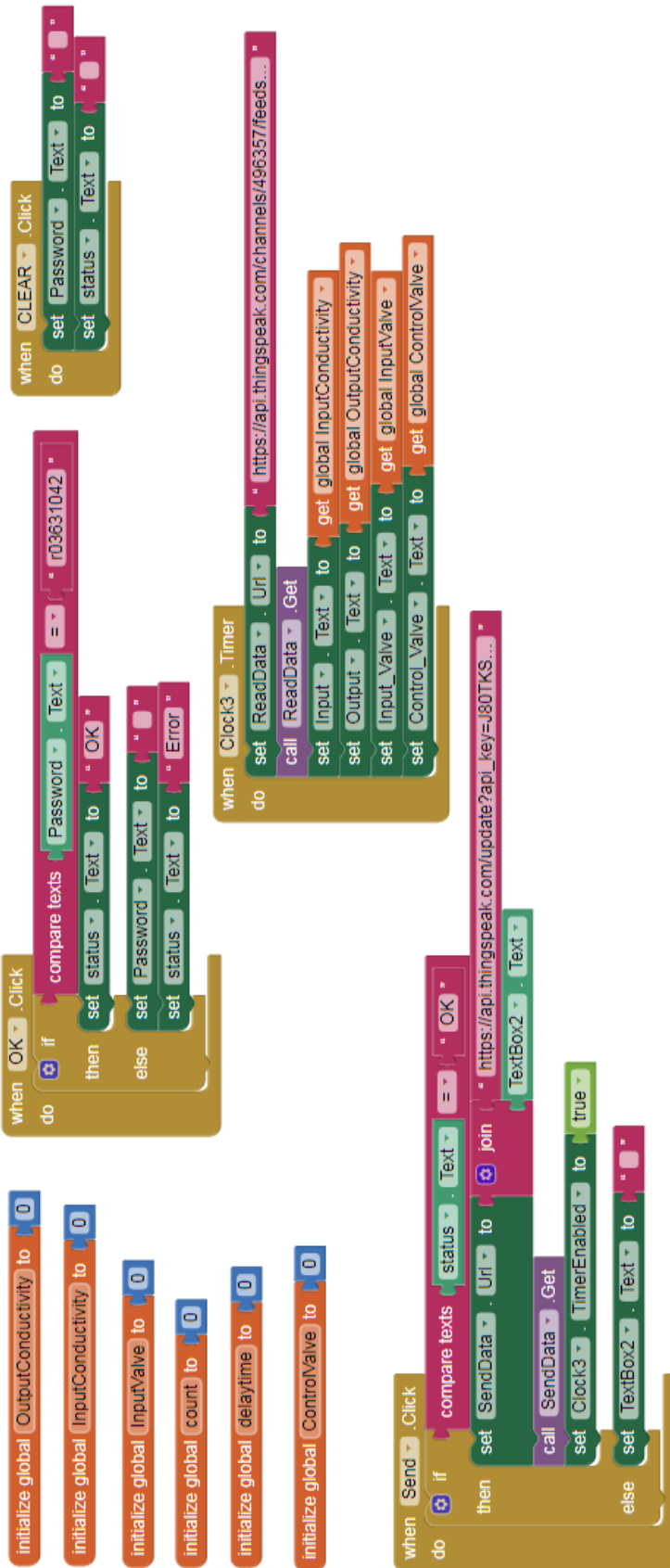
updateThingSpeak("184.106.153.149", Write , ECcurrent, ValveInReturn,
ValveControlReturn);
delay(30000);
}

/*
ch=0,let the DS18B20 start the convert;ch=1,MCU read the current temperature from the
DS18B20.
*/
}

```



附件二 App開發磚塊圖



```

when ReadData . GotText
url responseCode responseType responseContent responseContent
do set Response . Text to get responseContent
if get responseCode = 200
then initialize local json to call ReadData . JsonTextDecode
in jsonText
set global InputConductivity to look up in pairs key "field2"
pairs get json
notFound
set global OutputConductivity to look up in pairs key "field3"
pairs get json
notFound
set global InputValve to look up in pairs key "field4"
pairs get json
notFound
set global ControlValve to look up in pairs key "field6"
pairs get json
notFound

```

