



應用 IoT 與 MEMS 於 坡地監測系統之開發經驗分享

謝佑明／國立臺灣科技大學營建工程系資訊科技組 副教授

臺灣地區多山少平原，無論公共建設或私人建案都必須往山區發展，以滿足民生或工業建設需求。近年來由於全球暖化，短延時強降雨的極端氣候事件出現頻率愈來愈高，其對於山坡地安全性極為不利，因此山坡地監測技術無論在範圍和密度上都有其提升的必要性。物聯網（IoT, Internet of Things）或稱智慧聯網為近十年來一個新興的時髦術語（buzzword），泛指將裝置連接至網際網路，並有眾多控制器、MEMS 感測器、與通訊模組圍繞此概念被開發出來，而時髦的好處是相關產品大量生產而價格低廉，擁有極佳的效能／價格比，具有提升山坡地監測密度的潛力。本文將分享利用 IoT 相關軟硬體技術開發坡地監測裝置的經驗，並期望未來能看到更多土木營建方面的自動化監測設備能加以應用。

緒論

臺灣是個多山的小島，也為全世界平均人口密度前 20 高的國家。如圖 1 所示，全臺灣約一半的人口居住於圖中紅點處，也代表了若是民生經濟若要能持續發展，山坡地的開發與利用勢在必行。然而坡地開發的安全性近年來屢屢由於地震、強降雨、與老化的測試與挑戰，而發生了如小林村與國道 3 號 3.1K 的事故，使得自動化坡地監測技術愈來愈受到重視。而未來所需監測的範圍勢必將愈來愈大，且監測的空間密度需求也將愈來愈高，因此需要引入新技術，使得自動化監測成本能有效降低。

IoT（Internet of Things）一詞傳由 MIT 的 AutoID 中心共同創辦人 Kevin Ashton 於 1999 年所創造^[2]，當時主要是為了發展 RFID（Radio Frequency Identification）標準以協助物流管理，之後逐漸轉變為各式裝置連接至網際網路（Internet）的代名詞並透過網際網路進行資料交換。思科（Cisco, 主要的網路交換器製造商）則認為 IoT 約在 2008 至 2009 之間真正誕生，因為在該時間點上連接至網際網路的裝置數量已經超過了網際網路上人類的數量了^[3]。圖 2 整理了



圖 1 臺灣 90% 人口所居住區域^[1]

Google 的趨勢分析（trends.google.com）數據，由圖中可知全世界 IoT 的熱搜程度在 2014 年後快速上升，而這三年為 IoT 最為熱門的時段。而 IoT 的熱門代表著在無論在感測器（sensors）、微控制器（sensors）、與通訊模組（communication modules）在市場上有著多樣的選擇，且在成本上由於大量生產而能有著低廉的價格，而在開發上也能在全球資訊網（World Wide Web）上找到充份的學習資源與尋求幫助的論壇。

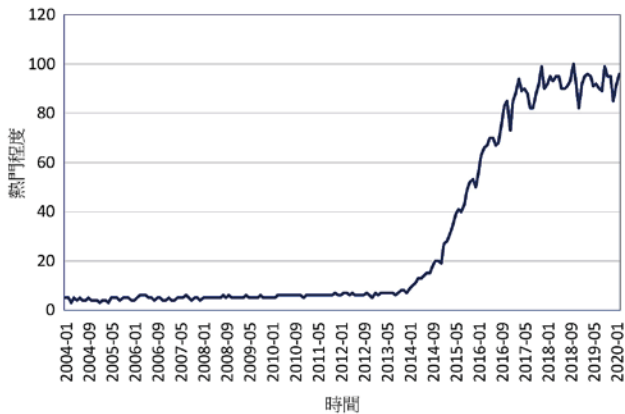


圖 2 Google 趨勢分析上 IoT 搜尋的熱門程度

本文將介紹使用熱門的 IoT 相關技術開發應用於坡地監測的裝置，該裝置具有的功能列於表 1。本文將介紹所選擇使用的硬體、軟體與伺服器端程式開發，以及各種開發中的心得，以期能達到拋磚引玉的效果，看到未來有更多應用 IoT 相關技術的土木營建監測儀器被開發出來並應用於守衛國土安全。

表 1 本文所開發的坡地監測儀器所具有的功能

物理量	說明
振動	以 200Hz 讀取三軸向加速度，並將時間、三向加速度值、與溫度儲存至 MicroSD 記憶卡，16GB 的記憶卡可以記錄約 11 個月的振動資料。
傾斜	依指定時間間隔將三軸向加速度轉換為雙軸向的傾斜度，並透過網路回傳中央伺服器。
溫度	依指定時間間隔讀取當時的儀器所感受到的氣溫。
時間	記錄資料的時間。
—	可透過 ssh 遠端登入更新系統程式，並下載所有監測資料。
—	將 USB 隨身碟插入 USB 埠後會自動將監測資料轉移至隨身碟上，以方便資料下載。
—	透過燈號提示使用者目前的運作狀態
—	每一小時將前一小時所記錄的監測資料進行壓縮，使記憶卡能儲存更長久的監測資料
—	定時將過舊的監測資料刪除，使記憶卡上若太久沒有去下載資料時，仍能記錄新的監測資料

硬體設計

本文所開發的監測儀器之區塊圖如圖 3 所示，其中連線的部份即為製作時需要連接電線的部份，線上的文字代表的是連接線路上所使用的數位介面種類（I2C、One-wire、USB）以及所需供電電壓（3.3V、5.0V），而各區塊所代表的模組或元件整理如表 2，並於後文加以說明並分享一些心得。

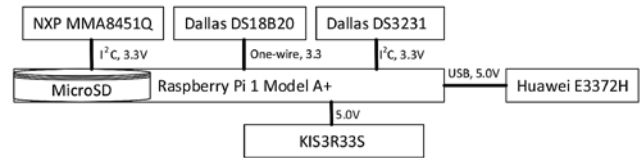


圖 3 本文介紹監測儀器之區塊圖

表 2 監測儀器中的主要元件或模組

類型	品名	簡要說明
控制器	Raspberry Pi Model 1 A+	運行所撰寫之監測程式，由各感測器取得資料並進行處理，並於預設時間間隔透過通訊模組將資料回傳中央伺服器。
感測器	NXP MMA8451	基於微機電技術的數位型三軸加速度計。 加速度之解析度：0.25 mg。 傾斜解析度 0.25/1000、0.014°、或 51 弧秒
	Dallas DS18B20	數位型溫度感測器
通訊模組	Huawei E3372H	USB 介面的 4G 網卡，用以連結至行動網路
儲存裝置	MicroSD 記憶卡	用來記錄收集的監測資料
時鐘模組	Dallas DS-3231	用來保存在系統重置或斷電時真實世界的時間
電源轉換	KIS3R33S	用來將 7-24 伏特的直流輸入電壓轉換為 5 伏特以供應電源給單板電腦

控制器

控制器一直是自動化監測儀器的核心，負責與不同感測器與模組進行溝通與控制，以執行自動化監測的任務。在目前自造者（maker）圈子中，最流行的兩種選擇為 Arduino 與樹莓派（Raspberry Pi）。

Arduino 原為義大利 Ivrea 互動設計學院在西元 2005 年所開發出來^[4]的開發板（development board），讓非電子資工背景的學生能夠很容易地創造出與物理世界互動的裝置，並採取開放原始碼（open source）的方式將其硬體設計、韌體（boot-loader）、與配合的整合式軟體開發環境（IDE, Integrated Development Environment）皆開放出來。也由於其開放的特性，如今可以在市面上買到各種形式與價格的 Arduino 相容開發板。

樹莓派為英國樹莓派基金所開發釋出運行 Linux 作業系統的單板電腦（single-board computer），亦即在一張板子上即擁有個人電腦的完整功能。樹莓派基金會為一英國的非營利組織，其宗旨在於促進電腦科學（computer science）的教育與相關主題^[5]。基金會在西

元 2012 年釋出第一代的樹莓派單板電腦 Raspberry Pi 1 Model B，定價為美金 25 美元，之後又出了另一 20 美元的版本 Model A，其移除了乙太網路與 USB Hub，使得體積更小而且價格更低。之後每一年至一年半都有一新版本上市，而更新版本常伴隨著更強大的效能、新增加的網路功能（WiFi、藍牙）、與略為上揚的價格。

Arduino 與樹莓派兩者的硬體比較列於表 3，其中可以看到樹莓派可提供的效能與記憶體數量遠勝於 Arduino，但伴隨著的是略高的耗電與價格。另外兩點可能很重要的差異，其一是 Arduino 板本身並無記憶卡插槽，故其無法安裝如 MicroSD 記憶卡儲存大量的資料，而樹莓派則必要安裝記憶卡才能運作，但安裝容量愈大的記憶卡，則可在其上儲存愈多的監測資料。另一主要差異是樹莓派沒有 ADC（analog-digital convertor），因此無法直接連結類比（analog）感測器（感測資料是以類比的電壓或電流方式傳出）。然而由於 Arduino 的 ADC 解析度僅為 10-bit，其實不適用於高精度的監測，故若要達到高精度讀取類比感測器的讀值時，無論 Arduino 或是樹莓派都需要另外連結如 ADS1211^[6] 的 ADC 模組，將其訊號轉換為 SPI（Serial Peripheral Interface）或 I²C（Inter-Integrated Circuit）可讀取的數位訊號。而表 3 中的 SPI、I²C、以及 UART（Universal Asynchronous Receiver/Transmitter）都是常見的標準數位介面，數位型（digital）的感測器往往可以使用其中的一種或以上連接至控制器。其中前兩者通常是針對在同一塊電路板上不同晶片之間的溝通，而 UART 則是以導線與不同電路板作較遠距離溝通時使用。USB 相信讀者們並不陌生，它是個人電腦連結週邊設備的標準介面。基本 Arduino 無法連結一般 USB 週邊裝置（網路卡、USB 儲存裝置等），但樹莓派可以。CSI（Camera Serial Interface）則是連接攝影機模組的標準介面。

除了表 3 所列 Arduino 與樹莓派硬體上的差異外，在軟體開發上可用的資源也有著相當大的差異。Arduino 由於是開放原始碼的專案，無論硬體、韌體、或是開發環境上都有網路社群的貢獻與資源可以使用，但是它畢竟是屬於單一硬體架構下的軟體開發，因此它的軟體資源僅限於 Arduino 這個環境。相對而言，雖然樹莓派似乎也是單一硬體架構，但由於其上運行的是完整的 Linux 作業系統，因此即使是其它硬體環境下所開發出來的軟體，只要不涉及特殊硬體，一

樣可以運行於樹莓派上，也因此樹莓派上可輔助開發的資源，例如程式庫，是遠遠多於 Arduino 的。而進一步地，樹莓派上可以選用多種的程式開發環境，如 C、C++、Python、Node.JS、PHP 等等，但在 Arduino 上選擇就非常受限。

然若耗電量為主要考量時，無論 Arduino 或是樹莓派都不是很理想的選擇，筆者過去指導的研究生使一基於義法半導體（STMicroelectronics）之微處理器 STM32F103C8T6 之便宜開發板（低於新台幣 200 元）並針對其供電電路進行修改，且在韌體開發上使用了處理器的省電模式，可得到含感測器電流消耗 5.4 mA^[7]。

綜合前述因素，包含了較佳的效能、較大的記憶體、可使用記憶卡儲存監測資料、可使用 USB 擴充、以及豐富的軟體開發資源、以及供電無虞的條件下，本文所介紹之監測儀器使用如圖 4 之樹莓派 Model A，作為主控制器。



圖 4 樹莓派 Raspberry Pi 1 Model A+

表 3 Arduino 與樹莓派的硬體比較

	Arduino ^[a]	Raspberry Pi ^[b]
工作時脈	20MHz	700MHz
記憶體（RAM）	32KB	512MB
運行時電流（mA） ^[c]	46.5 ^[d]	100 ^[e]
價格（新台幣） ^[f]	742	1063
記憶卡插槽	無 ^[g]	MicroSD
ADC	10-bit	無
I ² C	有	有
SPI	有	有
UART	有	有
USB	無	有
CSI	無	有
網路功能	另外購買 ^[h]	另外購買 ^[i]

^[a] Arduino Uno Rev. 3

^[b] Raspberry Pi 1 Model A+

^[c] 無連接任何感測器時量測到的電流消耗

^[d] <http://www.gadgetmakersblog.com/arduino-power-consumption/>

^[e] <https://raspi.tv/2014/raspberry-pi-a-how-much-power-does-it-need>

^[f] <https://www.digikey.tw>，其它網路商店可以找到更優惠的價格。

^[g] 可外加記憶卡模組，但需另外購買且會佔用數位介面。

^[h] 主要使用 UART 介面以及 AT 指令集，速度受限於 UART 的速度。

^[i] 可使用如^[g]所述之 UART 介面，也可使用標準 USB 網卡，速度較快。

感測器模組

所謂感測器模組 (sensor module) 如圖 5(a)，為廠商將特定感測器晶片依該感測器所建議之電路焊接於模組之電路板上，並將可能需要連接的接腳引出至電路板上易於連接處 (如圖 5(a) 左側的圓洞)。在選擇感測器模組時，有以下幾個重要因素需要考慮 (1) 使用介面、(2) 模組電路供電電壓 V_{cc} 、(3) 介面電壓以及 (4) 感測器規格。其中使用介面通常與感測器一致，可以感測器的規格書中找到。供電電壓與介面電壓則必需從模組的說明書或規格書找到，介面電壓通常為 5V 或是 3.3V，對應於 Arduino 與樹莓派的數位介面電壓，也有的模組會內含電壓準位轉換模組 (level shift)，使其模組可同時應用於 Arduino 與樹莓派。至於 4 的感測器規格，則依需求加以選擇。為了量測振動訊號，本文介紹之裝置使用了 NXP MMA8451^[8] 這顆基於微機電系統 (MEMS, Microelectromechanical Systems) 技術的數位型三軸加速度規的感測器模組，如圖 5(a)，其透過 I²C 數位介面連接於樹莓派，並使用樹莓派所提供的 3.3V 接腳作為其電源。MMA8451 是在 2011 年時所能找到 14-bit 解析度、低雜訊、便宜、且易取得的三軸加速規，其主要市場其實是應用於智慧型行動裝置如手機、平板電腦等以偵測手機的姿態^[9] 以及是否有動作等的感測器。熟悉訊號量測的讀者可能會覺得 14-bit 解析度並不高，但由於這類應用於消費性電子產品的三軸加速規其雜訊並不低，以 MMA8451 這顆規格書^[8] 上所給出的雜訊為 $99\mu g/\sqrt{Hz}$ ，14-bit 解析度已足夠^[10]，更高的解析度量測的也僅是雜訊的訊號變化而已。而在 2019 年，市場上已經可以找到更低雜訊與更高解析度的數位型三軸加速規模組。而另外應注意的是，數位型三軸加速規除了基本讀取三軸向加速度外，其晶片上往往不僅有將三軸向加速度數位化成讀值，而常常有許多額外的功能或模式，除了基本應設定欲量測的加速度範圍與取樣頻率外，其它尚有許多可能的設定，例如低功耗模式、高解析度模式、低雜訊模式、濾波設定等等。使用時除詳讀基本的規格書外 (例如參考文獻^[8])，也應注意感測器製造商所提供的應用筆記 (application notes, 例如參考文獻^[9,10])，常常可以發現許多有用的資訊。

而由於在過去經驗中，三軸加速規量測到的加速

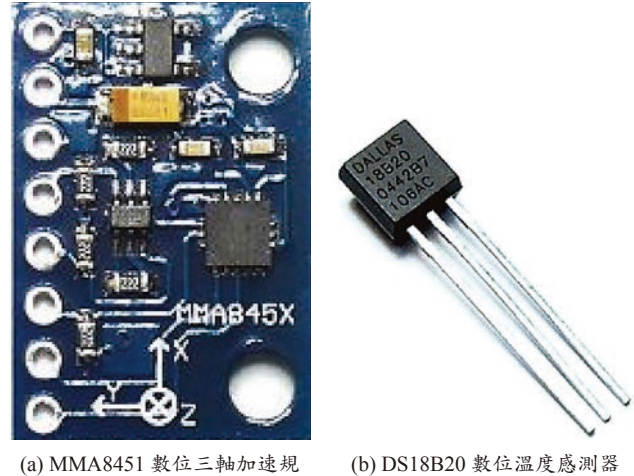


圖 5 所使用感測器模組與感測器 (大小比例與真實不同)

度與溫度^[11] 關係密切，加速度值或是其換算出來的傾斜度皆需經過溫度的補償校正，才能得到較精準的結果。市面上可取得的數位型溫度感測器中，圖 5(b) 之 MAXIM 公司出產的 DS18B20 算是其中價格便宜、接線容易者。但其數位介面很特別，並非如表 3 中所列常見的數位介面，而是使用其獨有的 one-wire 數位介面，但無論 Arduino 或是樹莓派，都已經可以找到現成的程式碼或模組去取得 DS18B20 的溫度讀值。

通訊模組

為了讓所開發裝置能連上網際網路，且配合臺灣於 2018 年底停止 2G、3G 行動網路的服務並全面邁入 4G 行動網路的時代，而選擇使用了華為 E3372H 之 USB 介面支援 4G/LTE 的行動網卡，如圖 6。將其連接上樹莓派後，系統會多出一張網路卡，並可以網頁瀏覽器瀏覽至網址 <http://192.168.8.1> 以控制是否要連線至行動網路並讀取簡訊。然為了達到自動化監測的目的，筆者另外利用了 bash 並配合 Linux 下常見的工具撰寫了幾個不超過 20 行的小程式以控制此行動網卡以達到程式化控制連線狀態、讀取與清除簡訊等目的。



圖 6 華為 E3372H 之 USB 介面行動網卡

儲存裝置

樹莓派為一單板電腦，其必需使用標準的 MicroSD 記憶卡以儲存其作業系統，官方提供的作業系統為 Raspbian，其中最精簡版本安裝完約略佔據 1.8GB 的記憶卡空間，因此若使用 16GB 的記憶卡，大概會剩餘約 14GB 的空間可用來儲存監測資料。

時鐘模組

在一般個人電腦中，常見一 CR2032 的水銀電池，其供電給電腦中的時鐘模組用來保存與維護電腦在關機未通電狀態下的電腦時間，使其與真實世界的時間近乎同步。在樹莓派上並無此裝置，因此樹莓派開機時若沒有網路連線的話，其系統時間會不同於真實時間。為了能將所記錄的振動資料與真實時間連結起來，故需加裝一時鐘模組。

本文介紹之裝置使用了基於 MAXIM 公司的 DS3231 的時鐘模組，如圖 7，其背面為一 CR2032 水銀電池，正面則可看到兩顆晶片，其中大顆的即為 MAXIM 公司的 DS3231^[12]，其為 I²C 介面的晶片，負責記錄目前時間並處理計時更新的邏輯（*e.g.* 閏年、每月份不同天數等等）。小顆的則為一額外的 4KB 非揮發性記憶體晶片 AT24C32，對於使用 Arduino 的使用者來說可能值得加以利用，但由於樹莓派已有記憶卡可長期保存大量資料（遠遠超過 4KB），因此本文介紹之儀器並未加以應用。

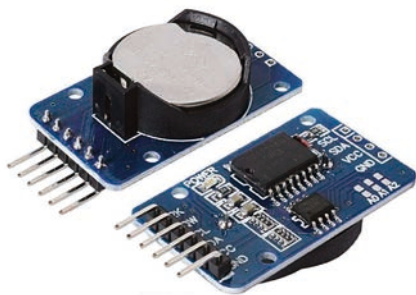


圖 7 DS3231 時鐘模組

電源轉換

在選擇電源轉換模組時，主要考慮 (1) 輸入電壓、(2) 輸出電壓、(3) 最大輸出電流、以及 (4) 轉換效率。

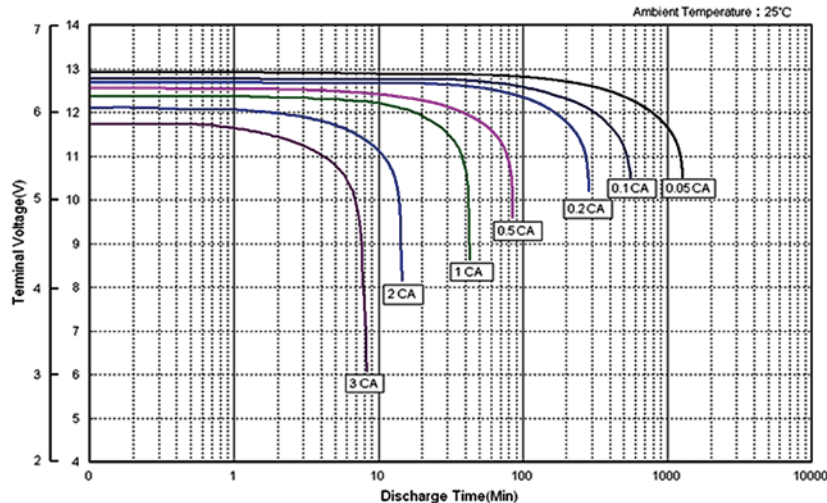
輸入電壓上，一般大地工程自動化監測常使用汽機車電瓶作為其電源，其好處是便宜、量大、穩定性高，但缺點是重量重、體積大（相對於鋰電池）。一般

汽機車電瓶標稱電壓為 12 伏特，但其電壓其實是隨著放電的過程逐漸下降，如圖 7 所示，且與放電電流（圖 8 中的不同 CA）大小相關，而感測器、單板電腦等等電子產品都需要固定且穩定低雜訊的工作電壓輸入，因此需要電壓轉換模組，將電瓶輸出電壓轉換至合適於供給各模組之所需。

在輸出電壓的考慮上，由於樹莓派本身設計上是以 USB 的 5 伏特供電，其板子上已內建降壓模組，將 5 伏特的 USB 輸入電壓降轉為 3.3 伏特與 1.8 伏特供外接感測器與板子上的系統晶片使用，因此前述的感測器、感測模組、與時鐘模組皆直接使用樹莓派所提供的 3.3 伏特作為其電源輸入，我們在設計上只要選擇一電源轉換模組將輸入電壓調節至 5 伏特，並且能供應的電流能滿足所需即可。

在最大輸出電流的考量上，而在所有前述模組或元件中，最耗電者應為 4G 行動網卡，由於其規格上並無說明其所耗費電流，故僅能以最大可能值估算，也就是 USB 2.0 介面規格上提供電壓 5 伏特、電流最大 0.5 安培。雖然樹莓派的規格書上建議使用 5 伏特 2 安培的電源供應器，但 2 安培應該是保守值，如表 3 所列之運行時電流僅為 0.1 安培，並沒有到 2 安培那麼多，應該只有在極端狀況下，如同時撥放影片、進行影片後製、讀寫記憶卡、存取網路等等才有機會到 2 安培。若取用保守值，則電壓轉換模組需能提供 2.5 安培的最大輸出電流。

最後應考量模組轉換效率，轉換效率指的是有多少電能被有效轉換，而用 1 減去轉換效率，即為浪費掉的電能。例如線性降壓（linear regular）模組其轉換效率較差，因其可視為使用一自動調變的可變電阻使輸出電壓降低至目標值，因此其壓降乘以電流的電能會以熱的形式散發出去，而若電流很大時會需要主動散熱以避免降壓模組燒毀，但它所輸出的電壓電流非常的穩定。另一種切換式降壓（switching regular）模組轉換效率較好，可避免電能的浪費，其原理是以極高頻率不斷地開關電路，使其輸出的電壓值平均於目標值，故其壓降不會產生廢熱也不會浪費電能，唯其輸出電壓會有漣波（ripple）的跳動現象，至於其跳動大小應可由降壓晶片的規格書中可以找到。

圖 8 鉛酸蓄電池放電曲線^[13]

軟韌體程式開發

在將前述所有硬體正確連接完成後，接下來是開發程式使其運作。所開發的程式主要依循著表 1 所列之功能，以 C++ 程式語言配合如表 4 的第三方程式庫、Linux 下的工具程式、以及多執行緒（multi-threading）技術實現之。其中，yaml-cpp 是用來讀取以 YAML^[14] 格式所撰寫如表 5 的設定檔，讓設定人員在對監測儀器進行組態設定時可以一般文字編輯器進行修改，並可在設定檔中加入註解。bcm2835 程式庫讓 C、C++ 撰寫的程式得以存取樹莓派使用的系統晶片，以透過 I²C 或 SPI 與不同的感測器溝通，並進行感測器的設定（如取樣頻率、過採樣率、使用低雜訊或省電模式等等）與取得讀數。OpenSSH 則為 SSH（Secure Shell）通訊協定的伺服器端與客戶端工具程式，為 Linux 機器上標準的安全遠端登入使用工具。而由於透過 4G 網路連線至網際網路所取得使用的 IP 位址常常為電信商私有的 IP 位址，無法直接從遠端機器直接連線登入系統，此時可用 OpenSSH 的通訊埠轉接（port forwarding）或稱通隧（Tunneling）

的機制，將裝置的 SSH 伺服器所使用的 22 號通訊埠轉接至中央伺服器上的指定通訊埠（如 22222），之後只要以 SSH 連線至中央伺服器的 22222 號通訊埠，即如同直接連線至裝置上的 22 號通訊埠，以克服無法直接遠端登入電信商私有 IP 位的問題。

而在資料儲存上，系統設計時應同時考量 (1) 批次寫入與 (2) 意外斷電時的資料損失。批次寫入指的是在使用基於快閃記憶體長期資料儲存媒介（long term storage，如記憶卡、硬碟等）不宜將監測資料逐筆寫入，而是應累積一定筆數資料後再進行一次寫入的動作，原因是快閃記憶體的壽命是由其被抹寫次數決定，如果沒有累積足夠量大小的資料寫入，會造成不必要的抹寫而降低壽命。如表 5 中的 save_interval 參數設定會累積 1 分鐘的監測資料量於記憶體中，再一次寫入磁碟中，以期能減少快閃記憶體被抹寫次數。然而若在該 1 分鐘期間發生斷電事件，則該分鐘的監測資料會遺失。故該參數愈大有助於延長記憶卡的壽命，但斷電時會遺失較多的資料。

表 4 所使用的第三方程式庫、工具程式、與技術

項目	說明與相連連結
yaml-cpp	用來讀取 YAML 格式的設定檔的程式庫，透過 apt install libyaml-cpp-dev 來安裝使用。 https://github.com/jbeder/yaml-cpp
bcm2835	用來存取樹莓派上的 BCM2835 系統晶片之 I ² C、SPI 等數位介面的 C 語言程式庫，自行由作者網站下載編譯使用。 https://www.airspayce.com/mikem/bcm2835/
OpenSSH	加密存取遠端伺服器的伺服器端與客戶端程式，已預安裝於 Linux 作業系統中 https://www.openssh.com/
lzma	使用 lzma2 演算法壓縮資料檔案的工具程式，已預安裝於 Linux 作業系統中 https://tukaani.org/xz/
usbmount	自動掛載 USB 隨身碟的工具程式，並可自訂掛載完成後要作的動作。
OpenMP	產生多執行緒的程式語言延伸，為大部份現代 C++ 編譯器所支援。 https://www.openmp.org/

表 5 監測儀器設定檔

station_id: 123		
save_interval: 60	# seconds	
temperature_update_freq: 5	# seconds	
send_static_tilt_interval: 300	# seconds	
data_path: data/ # folder to hold gathered data		
days_to_keep: 270	# age of files to keep during cleanup	
cleandisk_interval: 12	# hours between disk cleanup	

而為了方便日後資料的整理，所開發程式會將每 1 個小時的監測資料儲存於不同的檔案之中，並將檔名以年、月、日、時的方式取名，例如 2019 年 4 月 1 日 20 點（晚間八點）0 分 0 秒至 59 分 59 秒的資料皆會儲存於 2019040120.bin 資料檔案中。一小時會產生 72 萬筆的振動資料，每筆振動資料皆包含了時間戳記、三軸向的加速度讀值、以及溫度，其所儲存的二進位檔案大小為 10.6MB。以 16GB 的記憶卡扣除掉作業系統與程式所佔空間後約有 14GB 的空間可供使用，因此可儲存約 56 天的監測資料。為了能在一張記憶卡上儲存更多的監測資料，本研究利用了表 4 所列之 lzma 工具程式將每小時的監測資料檔案進行壓縮，平均下來僅需原來的 16.8% 的儲存空間，也就是每小時的資料僅需約 1.78MB 儲存，因此相同的記憶卡可以儲存將近 11 個月的監測資料。

如表 1 所列，本文所開發之儀器在將 USB 隨身碟插入 USB 埠後，會將記憶卡上的監測資料自動地

轉移到隨身碟上，此功能乃透過 Linux 作業系統下的 usbmount 套件達成。此套件的設定檔在 /etc/usbmount.conf，其中需設定所欲支援的檔案系統，近年來大容量隨身碟需額外加入 EXFAT 或 NTFS 檔案系統的支援。另外在 /etc/usbmount/mount.d 目錄下可放入 bash 腳本檔，以在完成掛載後自動化地執行任務，以本儀器而言即為 (1) 改變樹莓派的燈號（透過寫入資料至 /sys/class/leds/led1/trigger 虛擬檔案達成）讓使用者知道檔案轉移已經開始、(2) 移動檔案至隨身碟上以及 (3) 所有檔案移動完成後自動卸載 USB 隨身碟，並改變燈號讓使用者知道資料已經轉移完成，並可以安全地將隨身碟拔除。

而由表 1 的功能列表、前述的介紹、以及表 5 的設定檔內容，可知所開發的軟硬體有數個不同發生頻率的任務需要處理，包括了每 1 秒鐘讀取時間與三軸向加速度 200 次、每 5 秒鐘更新 1 次目前的溫度資訊、每 300 秒計算 1 次目前的傾斜量並將資料上傳中央伺服器、每 1 小時更換 1 次記錄的資料檔案並把前 1 小時的監測資料檔案進行壓縮、每 12 小時進行一次記憶卡的清理，以確保記憶卡只保存指定期限內的資料。由於有許多不同發生頻率的任務，且每個任務的完成時間不固定，因此使用了表 4 中最後一項的 OpenMP 技術以產生多個執行緒，每個執行緒完成一項任務後休眠至下一次任務發生的時間。而所開發的程式其流程圖如圖 9 所示，其中

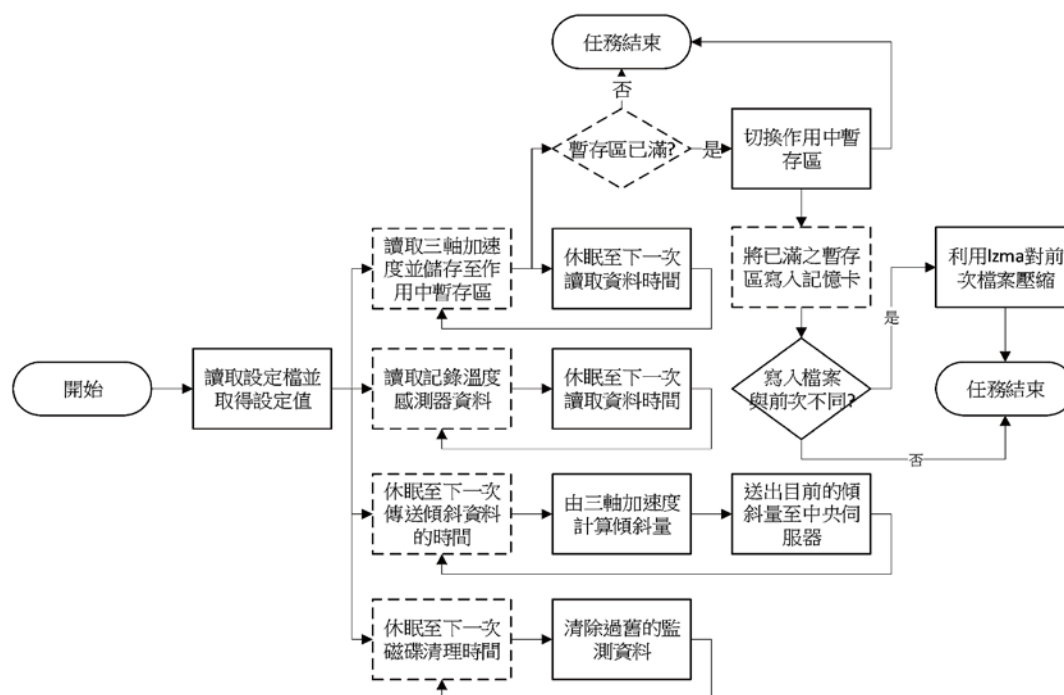


圖 9 所開發監測儀器之主要監測程式流程圖

虛線的部份代表利用 OpenMP 產生出新的任務 (task)，而不同的任務可以獨立而平行地以不同的執行緒執行，由圖 9 中可以看到有四個永不結束的任務，分別讀取加速度、溫度、計算傾斜並上傳伺服器、以及磁碟清理。另外有兩個臨時的任務，一個是當目前的記憶體暫存區已滿時將監測資料批次寫入磁碟、另一臨時任務是在寫入監測資料時，若當前的檔案與前次寫入的檔案不同時，代表前一個小時的監測資料已經寫完至檔案且不會再變動了，故此時可使用 lzma 將其壓縮儲存起來，以節省記憶卡空間。

而在圖 9 的流程中，可看到監測裝置會送出目前的傾斜量至中央伺服器，其乃透過 UDP (User Datagram Protocol) 通訊協定將資料單方向地送往伺服器，換言之在伺服器端也需一程式接收各監測裝置所送出的傾斜資訊。今天有許多新的伺服器程式開發技術，而

Node.JS 是其中的佼佼者。表 6 為伺服器上接收 UDP 封包的程式，其中包含了 (1) 引入 sqlite3 與 dgram 兩模組、(2) 開啟 sqlite 資料庫檔案、(3) 準備監聽 UDP 的 socket、(4) 當收到訊息時呼叫 parse 函式解讀客戶端所送出的監測資料並呼叫 saveDB 函式將監測資料存入 sqlite 資料庫、(5) 開始監聽 12123 通訊埠。注意到表 6 的程式其實為完整的程式，其中僅有 parse 和 saveDB 兩函式的內容沒有列出，由此可知伺服器端程式今日來說可以容易建立。

而資料儲存至中央伺服器後，可開發透過網頁將儲存於資料庫的資料作即時的查詢與呈現。如圖 10 的網頁透過 jQueryUI ^[15] 在網頁上加入一滑條以決定展示資料的起始時間與終止時間、透過 jQuery ^[16] 伺服器取得數據資料、並使用 Google Visualization API ^[17] 將資料以 X-Y 散佈圖的形式進行呈現。

表 6 伺服器接受 UDP 封包的程式

```
let sqlite3 = require('sqlite3');
let dgram = require('dgram');

let db = new sqlite3.Database("data.sqlite", sqlite3.OPEN_READWRITE);
let server = dgram.createSocket('udp4');

server.on('message', function(msg, rinfo) {
  if ( msg.length != 0 ) {
    var info = parse ( msg );
    info.time = new Date ();
    saveDB (info);
  }
});
server.bind(12123);
```

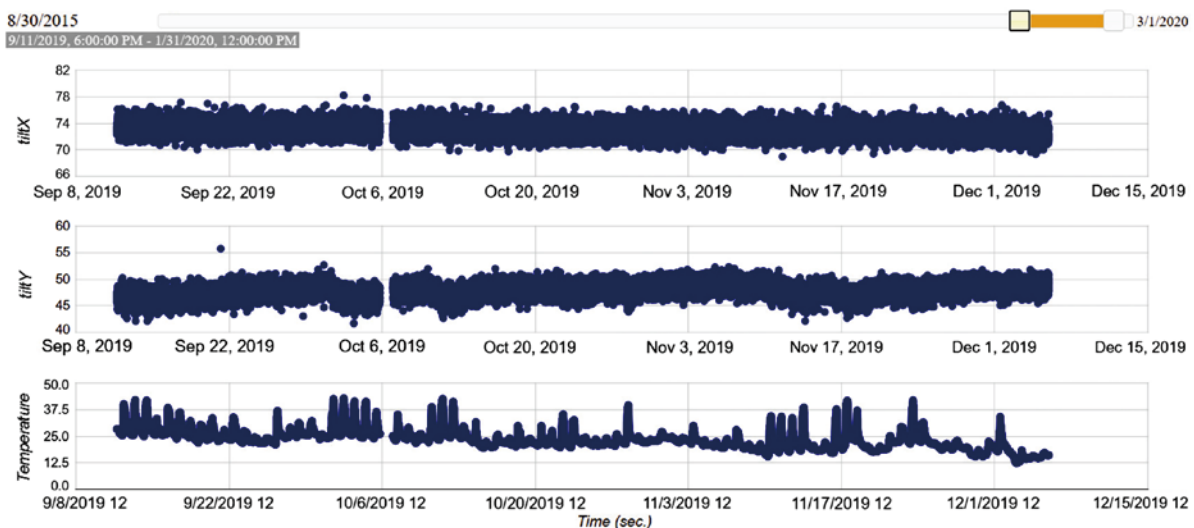


圖 10 透過網頁呈現伺服器上所儲存的監測資料

結語

本文分享了基於 IoT 技術開發的坡地監測儀器，包括其中不同的感測器或模組的選擇考量、其上軟體開發時的設計考量與所使用到的技術、伺服器端的程式開發、以及最終透過網頁呈現資料可用的相關技術。期待本文能達到拋磚引玉的效果，未來能看到更多、更密集、更先進的監測裝置，加入守衛國土家園的行列。

參考文獻

- 林佳賢 (2016), 「你知道台灣 9 成人口住在多少比例的土地上嗎? 4 張圖表看台灣和澳洲的人口分布」, The News Lens 關鍵評論, <https://www.thenewslens.com/article/33862>, 存取於: 2.24.2020。
- MIT AUTO-ID LABORATORY, retrieved Feb. 26, 2020, from: <https://autoid.mit.edu/>
- Evans, D. 2011. "The internet of things: How the next evolution of the internet is changing everything." Cisco Internet Business Solutions Group. Available at: https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf.
- Arduino. (2020, Feb. 21), In Wikipedia, the free encyclopedia. Retrieved Feb. 26, 2020, from <https://en.wikipedia.org/wiki/Arduino>
- Raspberry Pi Foundation (2020, Feb. 21), In Wikipedia, the free encyclopedia. Retrieved Feb. 26, 2020, from https://en.wikipedia.org/wiki/Raspberry_Pi_Foundation
- 彭信華 (2009), 「應用三軸加速規於建物傾斜量測之設計與驗證」, 國立臺灣科技大學營建工程系碩士論文。
- 彭敬鈞 (2016), 「三軸加速規於橋梁結構安全監測系統之研究」, 國立臺灣科技大學營建工程系碩士論文。
- NXP Semiconductors (2017), "MMA8451Q, 3-axis, 14-bit/8-bit digital accelerometer." NXP Semiconductors Data sheet: Technical data, Document Number: MMA8451Q Rev. 10.3, 02/2017, From: <https://www.nxp.com/docs/en/data-sheet/MMA8451Q.pdf>
- K. Tuck (2011), "Embedded Orientation Detection Using MMA8451, 2, 3Q." Freescale Semiconductor Application Note, Document Number: AN4068 Rev. 2, 10/2011, from: http://cache.freescale.com/files/sensors/doc/app_note/AN4068.pdf
- K. Tuck (2010), "How Many Bits are Enough?" Freescale Semiconductor Application Note, Document Number: AN4075 Rev. 1, 09/2010, From: <https://www.nxp.com/docs/en/application-note/AN4075.pdf>
- 王煒剴 (2013), 「以數位式三軸加速規建構傾斜儀之設計與評估」, 國立臺灣科技大學營建工程系碩士論文。
- DS3231, "Extremely Accurate I2C-Integrated RTC/TCXO/Crystal." Maxim Integrated, Inc., Document Number: 19-5170 Rev. 10; 3/15, From: <https://datasheets.maximintegrated.com/en/ds/DS3231.pdf>
- GS Battery, Taiwan, 「小型閥控式(密封式)鉛酸蓄電池的特性」, 台灣杰士電池密閉鉛蓄電池技術手冊, From: <https://www.gs-battery.com.tw/knowhow-technicalmanual-5.php>
- YAML Ain't Markup Language., retrieved Feb. 26, 2020, from: <https://yaml.org/>
- The jQuery Foundation, "jQuery User Interface." retrieved Feb. 26, 2020, from: <https://jqueryui.com/>
- The jQuery Foundation, "jQuery with less, do more." Retrieved Feb. 26, 2020, from: <https://jquery.com/>
- Google, "Google Visualization API Reference." retrieved Feb. 26, 2020, from: <https://developers.google.com/chart/interactive/docs/reference>



中國土木水利工程學會 喜訊報導

陳彥伯局長	榮陞	交通部政務次長
許泰文教授	當選	國立海洋大學校長
薛春明秘書長	榮獲	中國工程師學會 109 年傑出工程師獎
胡宣德教授	榮獲	中國工程師學會 109 年傑出工程教授獎
吳瑞賢教授	榮獲	中國工程師學會 109 年傑出工程教授獎