

淡江大學電機工程學系碩士班(通訊與電波組)

碩士論文

指導教授：詹益光 博士

共同指導教授：曾憲威 博士

智慧型家電節能管控系統

Energy-Efficiency Management

System of Smart Appliance

研究生：陳志銘 撰

中華民國 103 年 6 月

誌謝

在這兩年的碩士生活中，能夠順利的完成論文研究，我要謝謝許多人。

首先我要謝謝詹益光老師，不管是在學業上或是生活態度上都教導我需多。在學業上，詹老師傳授了需多專業知識，透過這些知識讓我能夠解決在研究上的問題。在生活態度上，我從老師身上學到積極樂觀的態度，去應對生活中的大小問題。第二個我要感謝的是李揚漢老師。李老師同樣在學業與工作態度讓我受益良多。在研究的路途上，李老師提出了許多天馬行空的概念及想法，讓我的研究路途充滿了驚奇與有趣的經驗。另外李老師的工作態度也讓我受益良多，以積極的態度面對工作事務，越困難或越煩雜的工作就要越積極去面對。第三個要感謝的是曾憲威老師，曾老師在我的研究路途中幫助許多，讓我的研究過程少走了許多彎路。

接下來我要感謝實驗室中一起研究的好夥伴，感謝博士班學長們曜哥、羅董、良祐的研究指點與幫助。感謝一起打拼的小李、狒狒哥、嘖嘖、史記、于德、哲遠、立楷、三色、振嘉、明中、師叔、廷瑋，讓我的研究生生活過得精采萬分。雖然在研究過程中被某人銃康了不少次，讓我不爽許久。但還是感謝有你們，一起奮鬥一起玩樂。另外還要感謝其他實驗室的大隻、花形、開智、大頭、老胡、泡芙、金屬馬修、朝棟、蟹堡、小五十、星星等許多人，一起度過這兩年生活。最後我要感謝我的家人，謝謝爸爸

媽媽，讓我順利讀完碩士班。感謝有他們的全力支持與幫助，讓我研究過程平順許多，謝謝你們。



論文名稱：智慧型家電節能管控系統

頁數：44

校系(所)組別：淡江大學 電機工程

學系(研究所)通訊與電波組

畢業時間及提要：102 學年度 第 2 學期

碩士學位論文提要

研究生：陳志銘

指導教授：詹益光 博士

曾憲威 博士

論文提要內容：

現今時代中，許多高科技家電問世，讓人們的生活過得十分便利。但越高科技的家電相對消耗的能源也就越多。因此能源消耗速度加快，能源問題也越來越常被各界提出討論。近幾年，國內更是因為發電廠問題而爭論不休。除了能源來源問題，如何去節約能源也變成一個重要課題。

針對節省能源的目的，本論文設計了智慧家電節能管控系統。這系統透過智慧家庭的概念，使用低速率無線個人區域網路 ZigBee，對家中各個家電設備進行互相連接。並且使用近年來流行的 Android 系統，在其系統下設計家電節能管控 APP。並將此 APP 安裝在智慧型手機上，成為整個系統的中央控制中心。之後將中央控制中心與 ZigBee 無線傳輸網路使用 USB OTG 傳輸媒介連結，中央控制中心便可透過 ZigBee 操控家電設備。並且本文也使用 WSN (Wireless Sensor Network) 概念。在 ZigBee 無線傳輸網路中，加入 Sensor 進行環境品質數據偵測，並將量測到的數據回傳給中央控制中心。中心可以依據數據進行家電節能管制動作，減少家電設備在不必要時後啟動工作。讓家電達到高效率與節約能源的目的。

關鍵字：智慧家電、Android、WSN(Wireless Sensor Network)、IEEE 802.15.4、ZigBee。

表單編號：ATRX-Q03-001-FM030-01

Title of Thesis: Energy-Efficiency Management System of
Smart Appliance

Total pages: 44

Key word: Smart home, Smart appliances, Android, WSN Wireless Sensor Network),
IEEE 802.15.4, ZigBee.

Name of Institute: Department of Electrical Engineering, Tamkang University

Graduate date: June, 2014

Degree conferred: Master

Name of student: Chih-Ming Chen
陳志銘

Advisor: Dr. Yih-Guang Jan
Dr. Hsien-Wei Tseng

詹益光 博士

曾憲威 博士

Abstract:

In these days, many high technology home appliances are available to make people quite convenience in daily life. On the other hand, high technology home appliances also consume large amount of energy. As energy consumption gets its speed up, it draws people to worry about the energy resources. In recent years, it has also brought in keen and controversy arguments in Taiwan whether to continue or to stop on the construction of nuclear power plant. How to save energy, in addition to find new energy resources, is also an important issue.

For the goal of energy saving it is in this thesis to consider the design of a smart home appliances energy saving management and control system. It is through the smart family concept to interconnect all home appliances through the use of low-rate wireless personal area network (LR-WPAN) ZigBee. It considers the design of APP under the popular Android system for the system. It also installs this APP onto the smart cell phone to make the cell phone as the central control terminal for the whole energy saving and management control system. It uses the transmission medium USB OTG (On-The-Go) to connect the

central control terminal with the ZigBee wireless communication network and then the central control terminal is through the ZigBee network to control the home appliances. In this thesis it also considers the WSN (Wireless Sensor Network) concept to use the sensors that are installed in the ZigBee wireless communication system to monitor the environment quality and then to feedback these measured data to the central control terminal for further analysis. The terminal then bases on the received data to make proper energy saving management and control action to reduce the unnecessary home appliances operation times so as to attain high efficiency and energy saving goals in the running of home appliances network.

表單編號：ATRX-Q03-001-FM031-01



目錄

誌謝	I
中文摘要	III
英文摘要	IV
目錄	VI
圖目錄	IX
表目錄	XI
第一章 緒論	1
1.1 研究背景	1
1.2 研究動機	1
1.3 論文架構	2
第二章 研究方法	3
2.1 文獻探討	3
2.2 Android 系統	4
2.2.1 Android 系統架構	4
2.2.2 Activity 生命週期	6
2.3 USB OTG 傳輸介面	9
2.4 ZigBee 無線傳輸系統	11
2.4.1 ZigBee 協定架構	12

2.4.1.1 實體層(PHY).....	12
2.4.1.2 媒體存取控制層(MAC).....	14
2.4.1.3 網路層(NWK).....	14
2.4.1.4 應用層(APL).....	14
2.4.2 ZigBee 裝置類型.....	15
2.4.3 ZigBee 網路拓樸結構.....	16
第三章 系統架構.....	19
3.1 家電節能管控 APP.....	20
3.1.1 APP 操控介面.....	21
3.1.2 APP 系統架構.....	23
3.1.3 家電節能管控 APP 流程.....	25
3.2 ZigBee 傳輸系統.....	28
3.2.1 ZigBee 協調器(Co-ordinator).....	32
3.2.2 ZigBee 感測節點和家電管控節點.....	34
3.2.2.1 智慧家電控制板.....	35
第四章 系統情境模擬驗證與節能比較.....	36
4.1 情境一：溫、濕度 Sensor 測試.....	36
4.2 情境二：家電設備手動控制測試.....	37
4.3 情境三：使用溫溼度數值進行自動控制測試.....	37
4.4 節能比較.....	38
第五章 結論與未來展望.....	40

5.1 結論.....	40
5.2 未來展望.....	40
參考文獻.....	42



圖目錄

圖 2.1 Android 系統架構圖[10].....	6
圖 2.2 Android Process 生命週期[11].....	8
圖 2.3 USB OTG 傳輸介面.....	9
圖 2.4 以 PC 主機為主之傳統 USB 的架構[12][13].....	10
圖 2.5 USB OTG 點對點架構示意圖[12][13].....	11
圖 2.6 ZigBee 協定架構圖[15].....	12
圖 2.7 ZigBee 各個頻段的通道切割[16].....	13
圖 2.8 ZigBee 網路架構圖.....	15
圖 2.9 星狀拓樸[17].....	17
圖 2.10 樹狀拓樸[17].....	17
圖 2.11 網狀拓樸[17].....	18
圖 3.1 系統架構圖.....	20
圖 3.2 Samsung S3 智慧型手機和 OTG 傳輸線.....	21
圖 3.3 智慧家電節能管控 APP 操作介面.....	22
圖 3.4 家電節能管控 APP 架構.....	25
圖 3.5 APP 執行流程圖.....	27
圖 3.6 FT-6250 ZigBee 基礎開發版.....	28
圖 3.7 ZigBee 無線傳輸軟體架構[19].....	29
圖 3.8 ZigBee 網路建立流程圖[19].....	31
圖 3.9 Coordinator 至 End-Device 之資料傳輸[19].....	32
圖 3.10 ZigBee 傳輸系統流程圖.....	33

圖 3.11 ZigBee 感測節點	34
圖 3.12 智慧家電控制板	35
圖 4.1 溫、濕度 Sensor 測試	36
圖 4.2 家電手動操控測試	37
圖 4.3 使用環境數值進行自動操控	38
圖 4.4 節能測試使用電風扇與消耗功率	39
圖 4.5 使用自動控制模式連續測試	39



表目錄

表 2.1 ZigBee、Bluetooth、Wireless 比較[7][8].....	4
表 2.2 ZigBee 使用頻帶及參數[14]	13



第一章 緒論

1.1 研究背景

在現今時代中，許多高科技家電問世，讓人們的生活過得十分便利。但越高科技的家電相對消耗的能源也就越多。因此能源消耗速度加快，能源問題也越來越常被各界提出討論。近幾年，國內更是因為發電廠問題而爭論不休。除了能源來源問題，如何去節約能源也變成一個重要課題。

節約能源在日常生活中如何去達成，除了最早倡導隨手關燈，到最近政府機關倡導的室內溫度未高於 26 度，避免使用冷氣、空調設備...等節約能源觀念。我們還有其他辦法來進行節約能源嗎？在最近幾年，有專家提出了智慧家庭的概念。智慧家庭是將各種家庭中的電器設備自動化，利用網路系統連結，讓設備發揮出高效率的功能，達成居家安全、環境健康、生活便利的目的。但隨著能源永續發展、節能減碳概念的興起，智慧家庭開始朝節能的目標前進。

1.2 研究動機

智慧家庭在目前市場中有需多產品，但真正在一般家庭中實現的還是少數。目前最成功的案例就是中華電信與遠雄企業合作的遠雄「二代宅」[1]。從開始建造時就規劃了光纖到家、智慧家庭...等，因此可見成功的原因在於線路的布建。透過線路讓設備與設備之間可以進行傳輸資料，但此種方法不適用於已經建造好的房子住家。因為已經建造好的住家線路都已經布建完成，如果要使用智慧家庭產品只能重新布建新的線路，花費成本太高。為此原因本文採用 ZigBee 無線傳輸方式，讓設備可以進行無

線的資料傳輸，這樣可以避免改建住家結構重新布建線路問題。[2]

另外智慧型手機市場成長快速，許多人都有一台智慧型手機，其功能可媲美一台小型電腦。而以往智慧家庭都會有一個中央控制中心，在市面上或是學術文章中大多都是使用桌上電腦或者是嵌入式平台系統，在建設智慧家庭系統時都需要花費購置，因此本文使用智慧型手機當作智慧家電節能管控系統的中央控制中心。並在手機中安裝本文設計的智慧家電節能管控 APP(Application)，再透過 ZigBee 無線傳輸系統連結家電設備傳輸資料。再讓 APP 透過資料自行判斷家電設備是否需要啟動，來避免多餘的能源消耗。[3]

1.3 論文架構

在本論文的章節中，第二章會討論到目前文獻中與市面上的智慧家庭的建構方法，並且找出適合方法並介紹。而第三章會說明智慧型家電節能管控系統的架構，說明每一個區塊使用的設備及元件，以及說明使用何種方法將整個區塊結合起來，最後詳細說明系統的動作流程。第四章會敘說智慧家庭節能管控系統在不同的情境下，會執行不同的功能。第五章會對本論文做個結論，並且探討此系統可以加以改善的區塊。

第二章 研究方法

2.1 文獻探討

智慧家庭，是將家中電器設備透過網路系統進行互相連結。讓家電發揮其整體性高效率的服務功能，並且確保居家安全、居家環境健康與生活便利。

在目前市面上智慧家庭產品或文獻，使用了各種不同的方法，去達到上述的目的。就以國內業者，中華電信的智慧家庭產品來看。他們使用網路來進行設備之間的連結。並且使用了 WSN(Wireless Sensor Network)的方法[4]，使用 Sensor 對環境品質進行量測。再將量測的環境數據透過 Wi-Fi 傳送到嵌入式系統平台(中控中心)。在系統平台收到環境數據後，進行分析數據後可知道環境的狀態。並且視環境需要與否，操控家電啟動或關閉。

而本文透過上述這些概念或方法，去設計智慧型家電節能管控系統。但在中控中心這個區塊，目前大多還是使用桌上型電腦或是嵌入式平台。但為了節省建置成本，本文更改換成近年來流行的智慧型手機。並且因為智慧型手機科技進步快速，功能媲美小型電腦，將其當作中控中心也足夠使用。[5][6]

而在智慧家庭的網路系統中，各家業者使用不同方法進行資料傳輸。而以前面提到的中華電信，則是使用 Wi-Fi 進行資料傳輸。而其他文獻裡則有人使用 Bluetooth 或是 ZigBee。因此我們將這三種無線傳輸方式進行比較與選擇。透過表 2.1，我們選擇 ZigBee 當作系統的傳輸方式。因為

表 2.1 ZigBee、Bluetooth、Wireless 比較[7][8]

Feature(s)	Wi-Fi	Bluetooth	ZigBee
電力持續時間	Hours	Days	Years
複雜度	非常複雜	複雜	簡單
節點數	32	7	64000
連線速度	3 秒以上	10 秒以上	30ms
覆蓋範圍	100m	10m	10m~300m
可延伸性	Roaming possible	no	yes
傳輸速率	11Mbps	1Mbps	250kbps
安全性	SSID	64bit、128bit	128bit AES and Application Layer user defined

ZigBee 的傳輸速率、電池消耗率與擴充性，更加符合本系統的需求。

2.2 Android 系統

Android 是以 Linux 為基礎而設計的行動裝置作業系統，主要使用在行動設備上，例如智慧型手機或平板電腦。目前 Android 系統是由 Google 成立的 Open Handset Alliance (OHA，開放手持設備聯盟) 領導開發，而最新的系統版本為 Android 4.4.2-KitKat。[9]

2.2.1 Android 系統架構

Android 系統是使用 Linux 當作系統核心，而且系統則是使用軟體堆疊(Software Stack)做為架構如圖 2.1。從底層到最上層共有三層：作業系

統(Operation System)、中介軟體(Middleware)以及應用層(Application)，而這三個階層又可細分出五個區塊。

系統作業層中包含了 Linux Kernel 區塊，這區塊是整個系統的核心。其主要的功能是讓系統透過驅動程式驅使硬體工作，讓系統核心可以正常的處理安全(Security)、記憶體管理(Memory Management)、行程管理(Process Management)、驅動程式模型(Driver Model)...等功能。

中介軟體層是在聯繫應用層與作業系統層。在應用層中的應用程式需要執行動作程序時，會向作業系統發出要求。讓作業系統驅動硬體執行動作。但由於應用程式是使用 Java 語言編寫，而作業系統層中的 Linux Kernel 是使用 C 語言，為了讓兩者可以進行溝通。中介軟體必須將應用層所要進行的動作程序，轉換成硬體能了解的動作程序。而負責轉換語言動作的就是函式庫(Libraries)區塊。而在此層中還有另一個區塊，虛擬機器(Android Runtime)，它則是負責管理應用程式的程序時間。

應用層中則是包含了應用程式框架(Application Framework)與應用程式(Applications)區塊。應用程式框架區塊是 Android 為了讓開發者節省編寫應用程式時間，在應用程式區塊與內部系統環境之間設置一個讓程式易於溝通的介面，讓開發者在編寫程式功能時直接呼叫使用，不用再額外撰寫。例如:開發者在應用程式中，想要使用 GPS 定位功能，透過呼叫應用程式框架中的 Location Manager，便可取得經緯度數值。而應用程式(Application)則是包括了以 Java 語言撰寫的應用程式，例如:電子郵件程式、日曆、地圖...等。

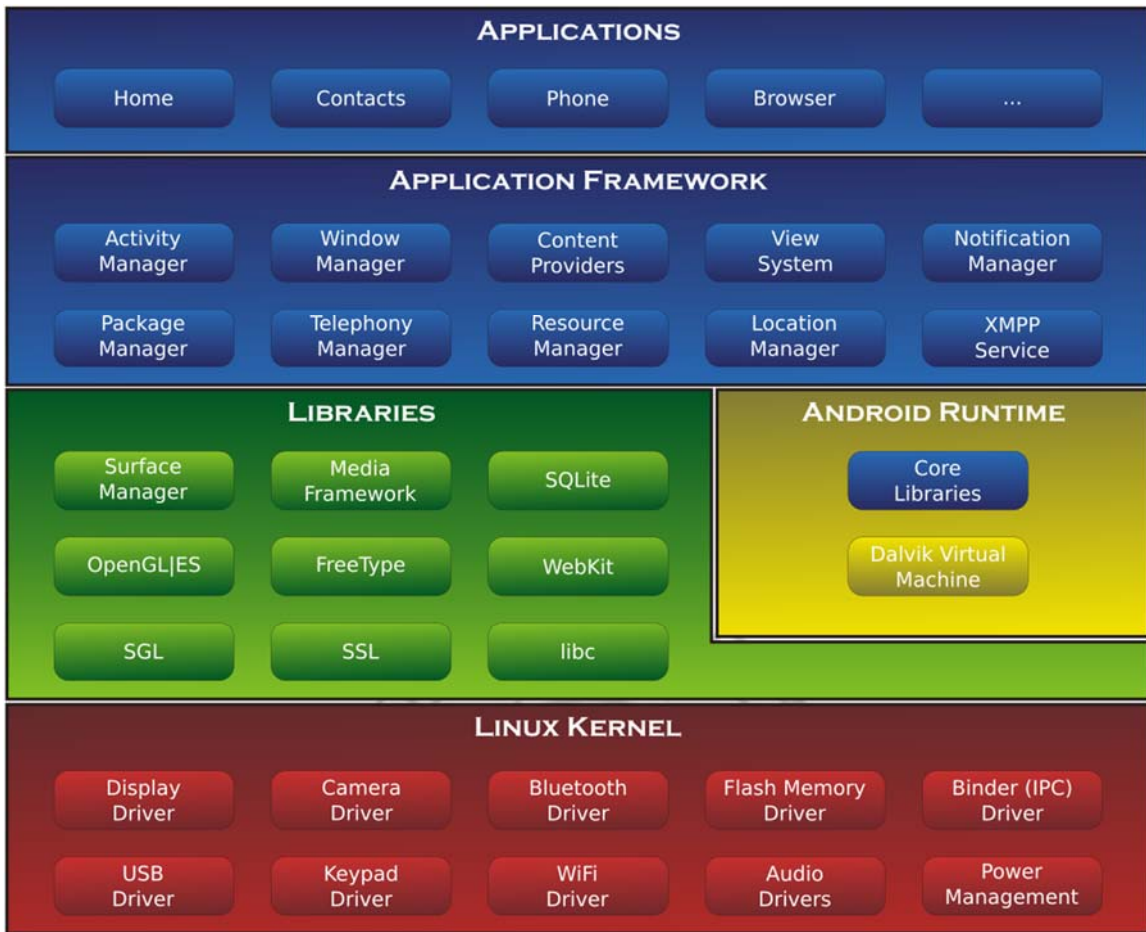


圖 2.1 Android 系統架構圖[10]

2.2.2 Activity 生命週期

Activity 代表的是應用程式的使用者介面。當程式啟動時它便會開始其生命週期，直到使用者或系統關閉應用程式，才會結束它的生命週期。在圖 2.2 中展現了 Activity 整個生命週期。在 Activity 生命週期中，標準的流程是從程式開啟到結束關閉，但在 Android 系統中 Activity 執行時會有可能有突發事件發生，這時 Activity 就有可能被迫中斷，為此 Android 透過數個函式，讓 Activity 可以因應不同的突發事件。下面則介紹這幾種函式。

1. onCreate(): 程式在啟動後，Activity 會向系統要求資源進行註冊元件，進行初始化動作，讓後續函式可以使用元件，完成應用程式功能。
2. onStart(): 元件在註冊完後可以連結到 Application Framework，元件就可以使用 Android 預先寫好的相對應功能。
3. onResume(): 是在當 Activity 元件初始化完成並與 Application Framework 完成連結後，APP 使用者在與程式互動的功能，都是在這函式中進行編寫與調用。
4. onPause(): 當 Activity 被暫停使用，這是程式會呼叫這個函式並且將資料數據保存，等到 Activity 重新再被使用，再將數據重新取回。
5. onStop(): 當 Activity 畫面完全消失在螢幕之中會呼叫此函式。
6. onDestroy(): 當 Activity 被系統或是被使用者關閉，系統會將 Activity 註冊的元件取消，讓資源歸還系統。

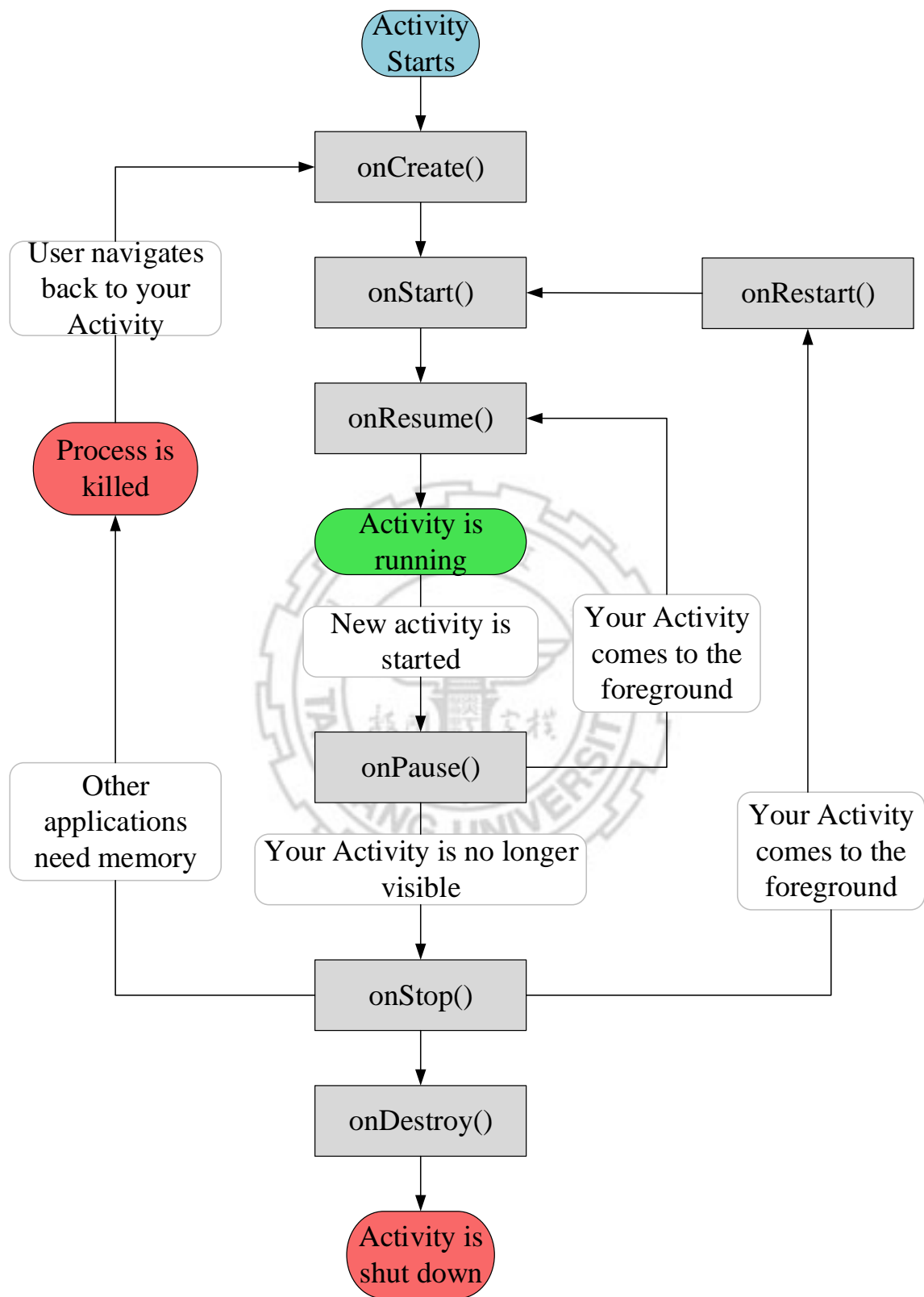


圖 2.2 Android Process 生命週期[11]

2.3 USB OTG 傳輸介面

USB OTG 全名是 USB On-The-Go 如圖 2.3，在 2001 年 12 月 18 日由 USB-IF (USB Implementers Forum)協會公布，但近年來發展起來的技術，主要應用在移動裝置與不同的設備連結，使兩者可以進行資料傳輸。

在以往的 USB2.0 標準中，USB 的架構是以桌上型電腦(PC)為主的星狀拓樸架構如圖 2.4。但這種架構會受限於裝置或設備必須連結 PC，以 PC 為主控端，才能夠讓裝置與裝置進行資料傳輸或通訊功能。而 USB 架構在初期設計時是被定位在電腦與周邊間相連接的介面裝置，為此 USB-IF 在制定 USB 的基本架構選用了主從式架構。將困難複雜的工作交給主控(Host)端，在此為 PC 負責，讓 USB 裝置在工作時只需負責較簡易的動作，讓 USB 使用上較為便利。



圖 2.3 USB OTG 傳輸介面

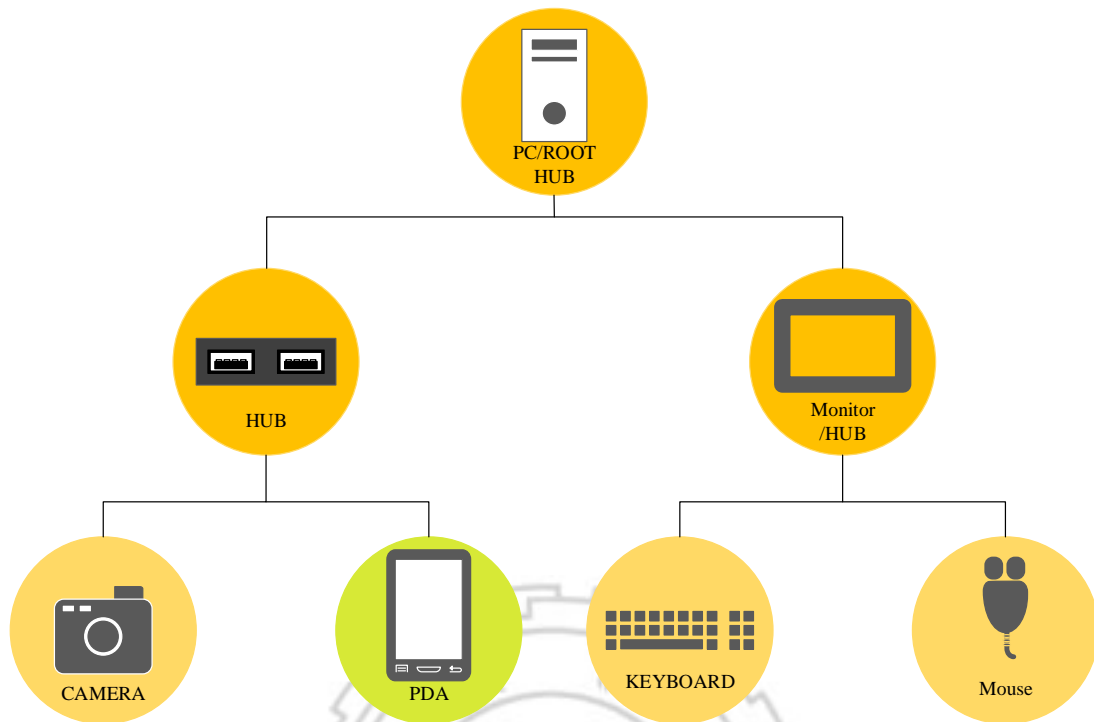


圖 2.4 以 PC 主機為主之傳統 USB 的架構[12][13]

基於上述原因，有許多周邊裝置使用 USB 介面做為連結電腦的媒介。但是現在有越來越多的 USB 裝置出現，隨著移動式的電子產品功能逐漸提升，移動式裝置具備的從屬裝置特性已不符合需求。為此要使得移動裝置跳脫 PC 主機(Host)端，例如：智慧型手機要直接連結隨身碟。但因為原本的 USB 拓樸結構限制了 USB 裝置之間互相連結的可能性。為此 USB-IF 協會使用 USG OTG 規範來解決這個問題。USB OTG 介面具備點對點 (point-to-point) 傳輸的優點如

圖 2.5。

USB OTG 是使用 USB2.0 標準延伸改進出來的規範，因此 USB OTG 還是遵循 USB2.0 規則，但為了達到兩個 USB 裝置可以連結使用，USB

OTG 1.0 規格中新增了雙重腳色裝置(Dual-role device)的概念。這個概念



圖 2.5 USB OTG 點對點架構示意圖[12][13]

是在敘說任何一個 OTG 裝置，可以當主機(Host)端，也可以當從屬(slave)端。因此當兩個 OTG 裝置相連結時，就可由其中一方當作主機端，另一方則是從屬端。這樣就達成 USB 兩個裝置可互相連結的需求。

2.4 ZigBee 無線傳輸系統

ZigBee 是一種低速率(250 kbps)短距離(100 公尺內)的無線傳輸網路協定，底層使用 IEEE 802.15.4 標準規範的實體層與媒體存取層，網路層與應用框架層由 ZigBee Alliance 聯盟制定，應用層則由使用者依照需求編寫功能應用，如圖 2.6。ZigBee 的特色是低速、低耗電量、低成本、支援大量網路節點、支援多種網路拓樸、低複雜度、網路建立快速、可靠、安全。[2][14]

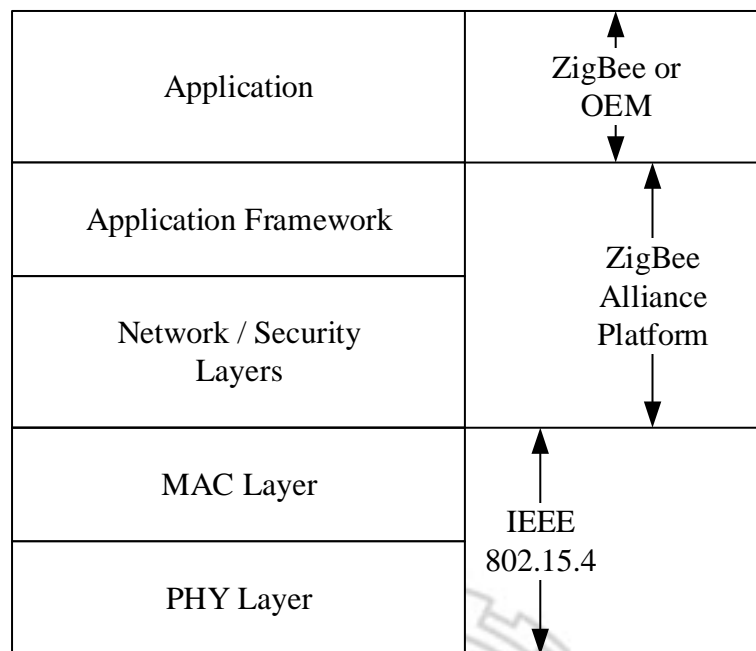


圖 2.6 ZigBee 協定架構圖[15]

2.4.1 ZigBee 協定架構

ZigBee 協定基本架構仍依據 IEEE 802.15.4 標準，並特別針對實體層 (PHY)與媒體訪問控制層制定標準。

2.4.1.1 實體層(PHY)

實體層在制定時，規範了 ZigBee 網路可支援使用三個頻段:2.4 GHz、915 MHz 及 868 MHz，並且定義各個頻段的通道數與傳輸速率如表 2.2。在 2.4 GHz(2400~2483.5 MHz)頻段上如圖 2.7，將其分成了 16 個通道，每個通道頻寬為 2 MHz，並且每個通道中心頻率間隔 5 MHz，而資料，傳輸速率則是 250 kbps。而 915 MHz 則是分配了 10 個通道，速度傳輸率則是 40 kbps。868 MHz 分配了 1 個通道，資料傳輸速率 20kbps。

實體層除了必須遵守上述協定，還必須達到下述功能:

1. 管理無線訊號傳送接收的啟動與關閉。
2. 在當下的通道中做能源的偵測。
3. 利用接收到的封包當作網路連接品質指標，估算網路品質。
4. 利用 CSMA-CA 進行評估可使用的通道。
5. 通道頻率的選擇
6. 資料傳送與接收。

表 2.2 ZigBee 使用頻帶及參數[14]

PHY (MHz)	Frequency band (MHz)	Channel number	Spreading parameters		Data parameters	
			Chip rate (k chip/s)	Modulation	Bit rate (kb/s)	Symbol rate (k symbol/s)
868	868-868.6	1	300	BPSK	20	20
915	902-928	10	600	BPSK	40	40
2450	2400-2483.5	16	2000	O-QPSK	250	62.5

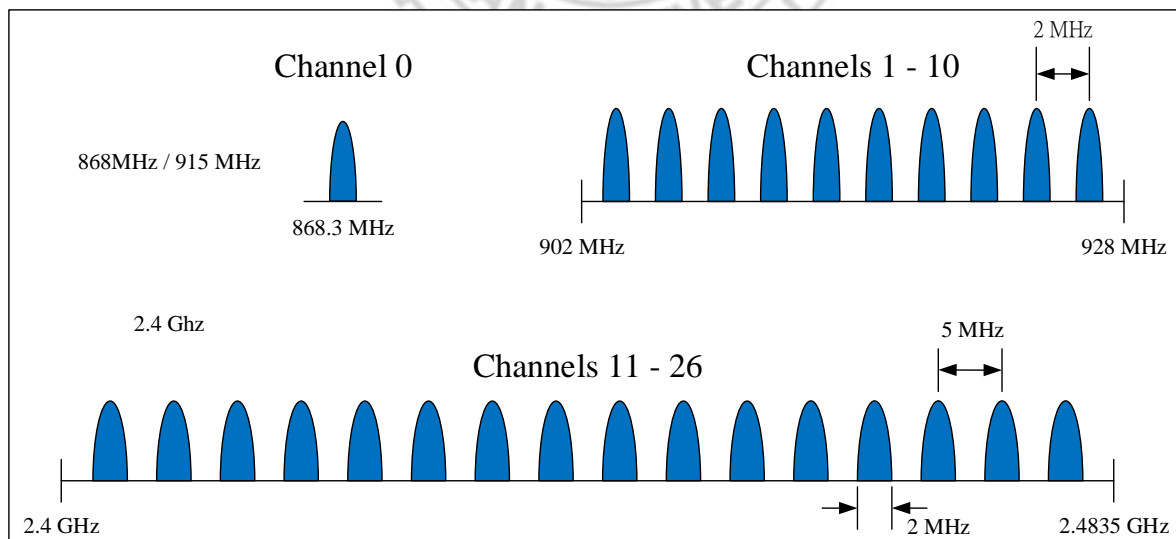


圖 2.7 ZigBee 各個頻段的通道切割[16]

2.4.1.2 媒體存取控制層(MAC)

ZigBee 媒體存取控制層的功用在於負責提供 MAC 資訊與管理服務，負責網路信標(beacon)管理、管理保障時槽(guaranteed time slots, GTS)、使用載波檢測多重存取/碰撞避免機制(CSMA/CA)進行封包傳輸、網路信標的同步與非同步。另外 IEEE 802.15.4 在此也規範了支援全功能(Full Function Device, FFD)及精簡功能(Reduced Function Device, RFD)兩類裝置，這兩類裝置可以支援不同的網路拓樸結構，並且擔任協調者(Coordinator)或者是終端設備(End-Device)的腳色。

2.4.1.3 網路層(NWK)

低層的協定是由 IEEE 802.15.4 負責制定，而 ZigBee 聯盟負責制定網路層與應用層，ZigBee 網路層主要控制上層網路的行為，包含網路初始化、建立新的路由、傳送路由資料封包至目的地、配發網路位址給新加入的網路設備、負責尋找與紀錄資料傳輸路徑。並且 ZigBee 網路層中也支援多個網路拓樸，其中包含了星狀、樹狀與網狀網路。

2.4.1.4 應用層(APL)

ZigBee 應用層包含了應用次層(Application Support, APS)、ZigBee 裝置元件(ZigBee Device Object, ZDO)和製造商所定義的應用物件(Application objects)。應用次層提供網路層與應用層之間的介面，維持物件之間的連結表(binding table)，並在連結的裝置之間傳遞訊息，它也維持

了一個應用次層資訊庫(APS Information Base, AIB)。ZDO 的功能包括起始應用程式支援應用子層、網路層以及安全服務等。

2.4.2 ZigBee 裝置類型

在 IEEE 802.15.4 協定中，為了降低成本與電力消耗，依功能需求制訂了全功能裝置(Full Function Device, FFD)和精簡功能裝置(Reduced Function Device, RFD)。FFD 擁有完整的 IEEE 802.15.4 協定功能，具有較高的運算效能與記憶體，但較為耗電需要使用固定電源供電。而 RFD 則是設定了簡略的 IEEE 802.15.4 協定功能，並且採用較低的運算效能與記憶體，因此只需電池供電即可。

在 ZigBee 的網路架構中有三種腳色如圖 2.8，透過這三種腳色可以組織成一個完整的網路。這三種腳色分別是 Co-ordinator、Router 與 End-Device。

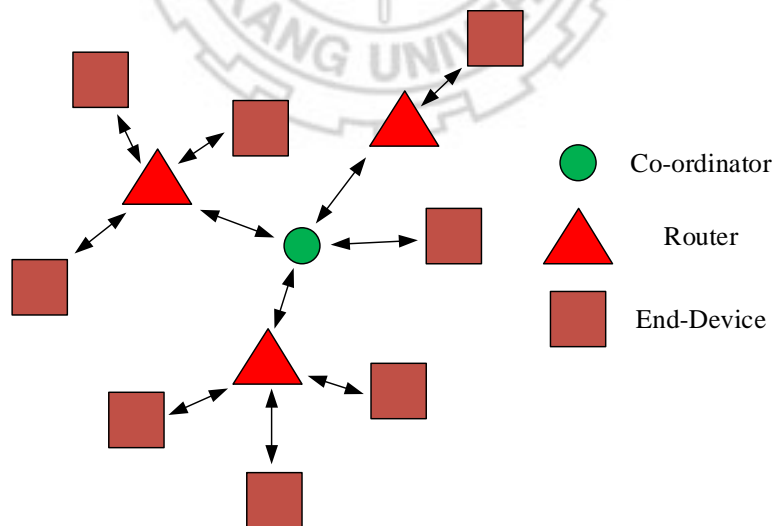


圖 2.8 ZigBee 網路架構圖

1. Co-ordinator 網路協調者:

在建立 ZigBee 網路的過程中，需要一個腳色去發起網路，並搜尋附近裝置加入網路中，而這腳色就由 Co-ordinator 擔任。Co-ordinator 是整個網路的核心，並且為全功能裝置(FFD)。其功能是在一個區域中，設置個人區域網路識別碼(Personal Area Network Identifier, PAN ID)，進行啟動 ZigBee 網路，並且協助建立網路中的安全層以及處理應用層的綁定。

2. Router 路由裝置:

Router 為全功能裝置，功能在於幫助 ZigBee 網路區域中的新裝置，加入網路協助其與其他的終端節點進行通訊。在一般的情況下，路由必須一直維持工作的狀態，保障節點與終端裝置的連線。

3. End-Device 終端裝置:

End Device 為精簡功能裝置(RFD)，其功能為傳輸裝置資料。因此不必去參與維護網路基礎的工作，並且可以選擇休眠節省電源，或喚醒執行工作。

2.4.3 ZigBee 網路拓樸結構

ZigBee 網路透過網路協調器、路由及終端裝置可組成三種網路拓樸架構，分別是星狀拓樸、樹狀拓樸與網狀拓樸。

1. 星狀拓樸:

星狀拓樸是由一個網路協調者與多個終端裝置組織而成如圖 2.9，在這之中所有的終端設備都只會與唯一的網路協調者進行連線，並且透過其進行資料傳輸，將資料轉送到指定的終端節點，其優點在於結構

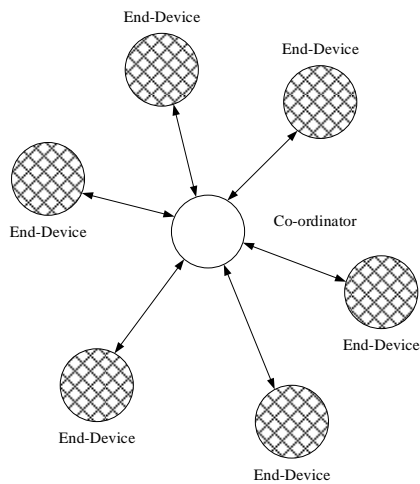


圖 2.9 星狀拓樸 [17]

簡單，建立容易，資料易於同步。

2. 樹狀拓樸

在圖 2.10 中，除了網路協調者與終端節點，在這新加入了路由，有了路由裝置進入網路結構，就可增加網路中的節點，延長網路連線的距離，增加網路規模。

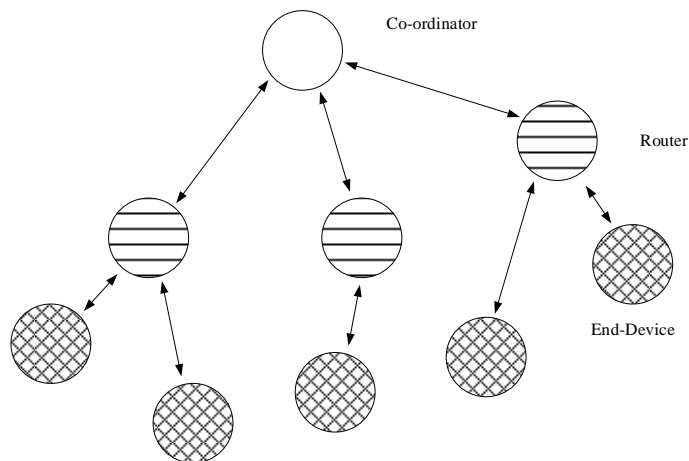


圖 2.10 樹狀拓樸 [17]

3. 網狀拓樸

網狀拓樸是以樹狀拓樸進行改進，在樹狀拓樸中終端裝置必須靠著父節點(Parent node)，才能夠進行訊息傳遞。而網狀拓樸讓路由與路由之間互相連結，這樣訊息在傳遞時，可以選擇最短路徑進行傳輸，減少訊息傳遞時間。另外因為路徑變多，不會因為單一節點故障，而導致訊息無法傳遞，訊息可透過其他路徑傳送到目的地。

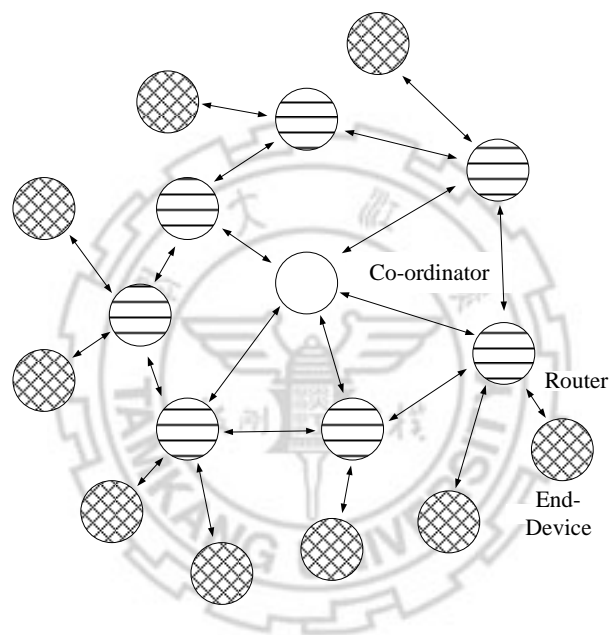


圖 2.11 網狀拓樸[17]

第三章 系統架構

在本文的智慧型家電節能管控系統中如圖 3.1，是使用 Samsung S3 智慧型手機當作系統的中央控制裝置，這個中央控制中心裡安裝了家電節能管控 APP，此 APP 最主要的功能是透過裝置上的 USB OTG 傳輸埠接收經由 ZigBee 傳送的生活環境感測資料，這些資料是由各個 Sensor 感測取得當下的室內生活環境品質數據，家電節能管控 APP 會使用這些數據並且比對人類舒適的生活環境參數，來自動控制家中的電器設備，調整環境品質或者是達到節能的效果。

ZigBee 傳輸與系統主要負責 APP、Sensor、家電設備之間的感測資料與控制訊號傳輸，ZigBee 傳輸系統中有兩個網路裝置角色，Co-ordinator 和 End-Device，Co-ordinator 負責的工作是分析 APP 傳送過來的訊號，判斷訊號是要求 Co-ordinator 進行何種動作，當判斷完成 Co-ordinator 會傳送訊號給指定的 End-Device 進行動作，而 End-Device 負責的工作則是接收控制訊號並且執行動作，例如：當 APP 需要更新環境的溫度數據，會傳送訊號到 Co-ordinator，Co-ordinator 判斷要進行更新環境溫、濕度數據便會傳送控制訊號到 End-Device，End-Device 就會讓 Sensor 進行偵測再透過 End-Device 回傳環境溫度數據。

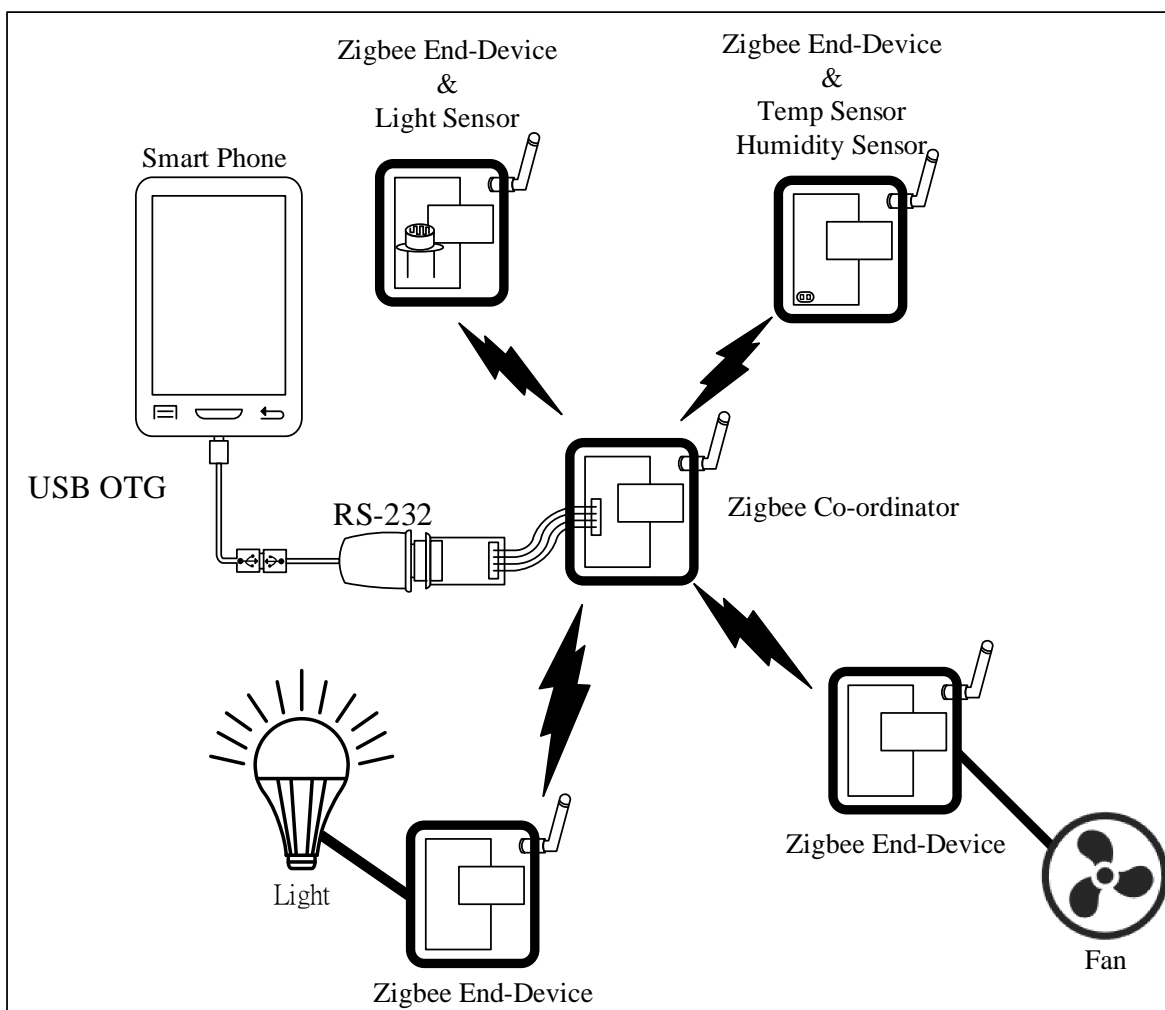


圖 3.1 系統架構圖

3.1 家電節能管控 APP

家電節能管控 APP 是運行在 Android 系統上。因此本文中使用的 Samsung S3 智慧型手機如圖 3.2，使用的系統是 Android 4.1.2 版本。APP 主要的功能整個系統的中控端，透過 ZigBee 無線傳輸系統回傳的環境數據，進行家電操控、調整環境品質與節能目的。為此 APP 必須與 ZigBee 傳輸系統進行連接，而中間的連接媒介是使用 USB OTG 與 USB to RS232，透過此方法 ZigBee 便可將環境數據資料傳送到 APP。



圖 3.2 Samsung S3 智慧型手機和 OTG 傳輸線

3.1.1 APP 操控介面

家電節能管控 APP 在設計的初期，主要的目的是要讓 APP 可以透過環境數據進行操控家電設備，也必須讓使用者清楚知道環境品質數值。因此在設計 APP 介面時，新增了環境數值資訊區塊。這區塊會將接收到的環境數據分類後，會將其顯示在對應的區塊中。除了資訊區塊，介面中還設計了其他兩個區塊，分別是：家電控制區塊與 ZigBee 連接控制區塊。

家電控制區塊是讓使用者可以自行控制家電。在本文中是針對電燈與電風扇進行操控，所以在控制區塊的上半部，有張電燈狀態的圖示。以

圖 3.3 (a)看到的是電燈關閉的狀態，如果透過 APP 開啟電燈，則圖示會變成亮燈狀態的圖示如圖 3.3 (b)。

ZigBee 連接控制區塊的功能是，當使用者將手機與 ZigBee 傳輸系統使用 USB OTG 和 USB to RS232 連接後，透過 Connect 按鈕開啟兩者傳輸通道，開啟通道後 APP 便可向 ZigBee 更新環境數據。



(a)

(b)

圖 3.3 智慧家電節能管控 APP 操作介面

(a) 未連接 USB to RS232 操作介面 (b) 已連接 USB to RS232 操作介面

3.1.2 APP 系統架構

在 APP 的功能中，系統要使用到 USB OTG 與 USB to RS232 進行資料傳輸，為此 APP 需要向 Android 作業系統要求使用 USB 功能。而作業系統層會透過 USB Driver，驅動 USB 硬體進行工作。在函式庫層中使用 libc，其可將應用層中編寫的功能轉換成 C 語言的形式。在應用框架層中使用了 USB Host、USB Manager、USB Device、USB Interface、USB Endpoint、USB Device Connection、USB Request、USB Constants 這 8 種 API。APP 透過這些 API，可以進行管理 USB 裝置、連接及傳輸資料等功能。

1. USB Host API

其功能是讓 APP 在 USB 架構中扮演主控端的角色，讓 APP 負責較複雜工作，如詢問連接的 USB 裝置描述資訊(Device Descriptors)、連接模式資訊(Configuration Descriptors)等相關工作。而連結手機的 USB to RS232 裝置只需負責資料傳輸工作即可。

2. USB Manager

是負責管理加入到 Android 系統上的 USB 裝置，並且與裝置通訊詢問裝置相關資訊。

3. USB Device

代表一個連接的 USB 裝置，其中包含了 USB 裝置的連接訊息、ID 訊息、接口介面資訊。

4. USB Interface

代表 USB 裝置接口，並且設計了一組功能，讓設備可以有一個

或多個接口進行通訊。

5. USB Endpoint

代表 USB 裝置接口上的端點，通常一個 USB 裝置接口有一個或多個端點，讓設備可以進行輸入與輸出的雙向通信。

6. USB Device Connection

代表 USB 裝置上端點的連接到，並且可透過他來進行資料傳輸

7. USB Request

代表一個非同步請求訊息，可透過 USB Device Connection 傳輸

8. USB Constants

USB Constants 定義了 USB 參數，讓 Linux Kernel 參考進行驅動裝置工作。

在應用層中，有家電節能管控系統的主程式(Energy-Efficiency Management APP)與 PL2303driver 副程式[18]。家電節能管控 APP 的主要功能(傳送控制訊號、判斷環境品質...)放在主程式中去執行。PL2303driver 則是幫助主程式進行管理、連接 USB 裝置工作。並且在本文中是使用 PL2303 轉換晶片的 USB to RS232 傳輸線。必須透過副程式進行轉換，程式才能接收到正確的資料，最後副程式會將系統與 USB 裝置的資料傳輸通道，交給主程式進行資料傳輸功能。

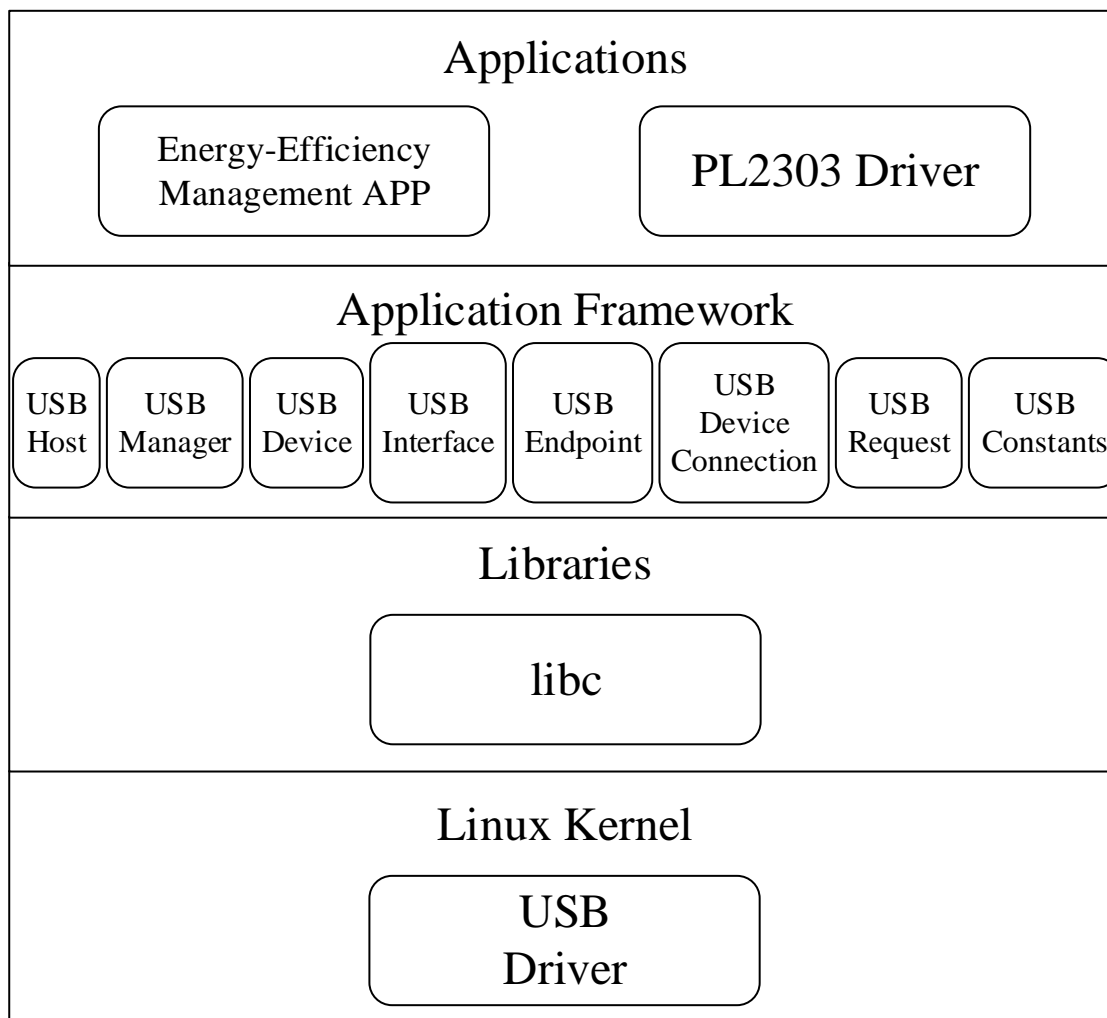


圖 3.4 家電節能管控 APP 架構

3.1.3 家電節能管控 APP 流程

在敘述完 APP 使用的功能與系統架構後，在這節中會說明 APP 在啟動後得執行流程如圖 3.5。

家電節能管控 APP 在啟動後，APP 的 Activity(程式畫面)會跟著執行。緊接著在 onCreate()函式中 Activity 會向系統要求資源，建立畫面中的互動元件與 USB 裝置管理元件。並且將各元件與應用層框架中的 API

互相連結使用。在 onResume()函式中，APP 主要功能都在此編寫調用。在此主程式會先向 PL2303driver 詢問 USB 裝置資訊，並且儲存在記憶體中。當使用者點下連接按鈕，主程式會透過裝置資訊開啟連接通道。當通道開啟後，APP 會每間隔一段時間，會自行對 ZigBee 要求取得環境品質數據，並將其顯示在環境資訊區塊。另外 APP 在開啟自動控制家電節能功能，APP 會依據環境品質數據，與使用者的環境理想舒適數據進行比較。當實際環境數據超過環境理想舒適數據範圍，系統便會發出控制訊息進行控制家電，直到數據回到理想範圍。



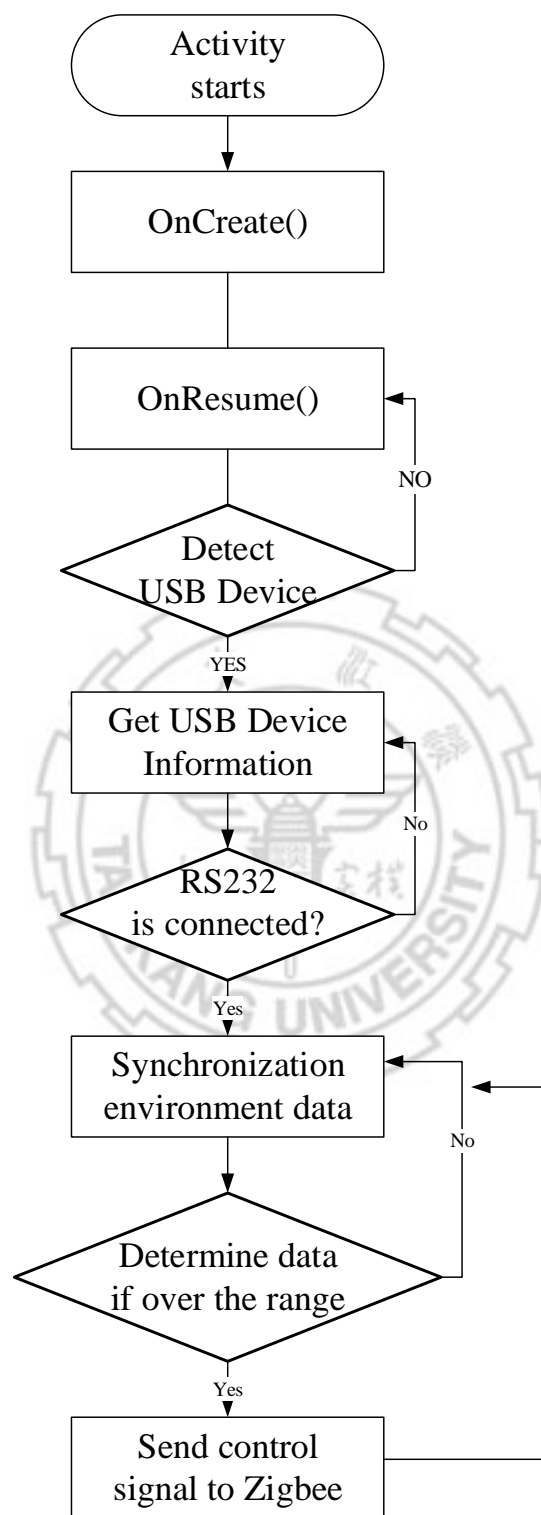


圖 3.5 APP 執行流程圖

3.2 ZigBee 傳輸系統

在 ZigBee 的傳輸系統中，本文使用源鼎科技的 FT-6250 無線基礎開發板如圖 3.6 與 FT-6251 無線感測開發板。這兩種開發版核心是使用 JN5121 模組晶片。其晶片遵循 IEEE 802.15.4 協定設計物理層與媒體存取層，並且經由其 32 位元的 MCU，可控制建立網路架構與資料傳輸。另外在模組板上還有 10 組的雙向 GPIO、UART、ADC、DAC、Comparator 輸出輸入介面。並且根據源鼎科技提供的無線應用開發應用程式，本文只需針對應用層進行功能編寫，即可達到家電節能管控系統的功能需求。

在圖 3.7 中為無線應用開發應用程式軟體架構，應用程式必須通過 802.15.4 的堆疊 API，並且和 IEEE 802.15.4 的堆疊層進行交互，來實現 MCPS/MLME 的請求和確認，消息的標籤與回應。讓 IEEE 802.15.4 堆疊層與底層的硬體以及暫存器互相作動。應用程式將周邊設備 API 與晶片



圖 3.6 FT-6250 ZigBee 基礎開發版

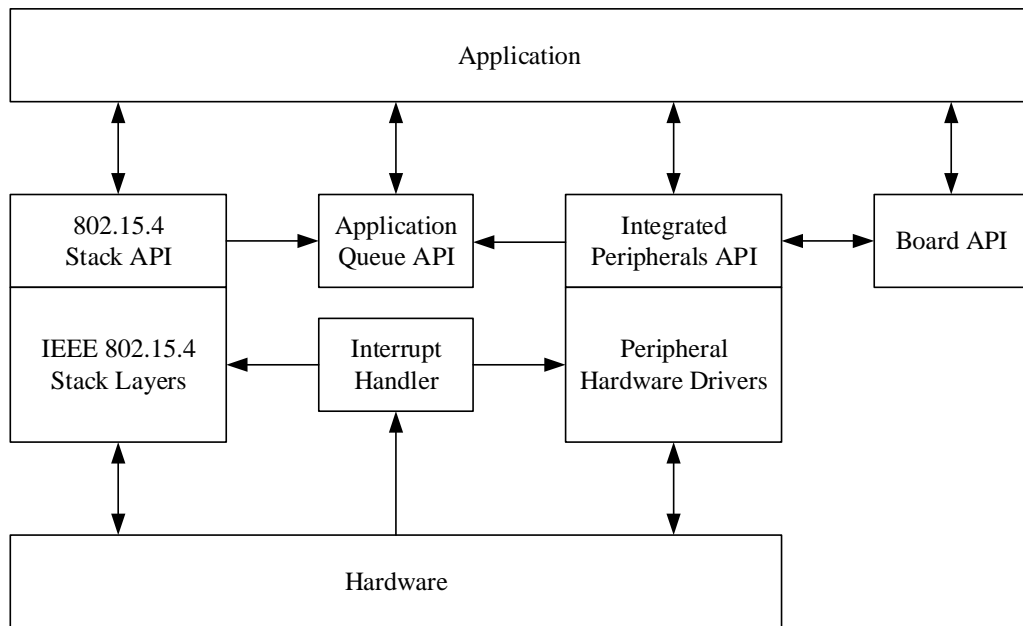


圖 3.7 ZigBee 無線傳輸軟體架構[19]

上的周邊設備進行交互，讓周邊設備 API 可以存取硬體暫存器。另外應用程式也可透過 Board API 進行使用開發板上的感應器，按鈕，LED 等等。在軟體的中間區塊有個中斷控制程序，它可將硬體層產生的各種中斷，通過中斷控制碼轉發給各個架構區塊。

接下來介紹到應用程式的網路建立流程。首先在建立網路前，會對每一個設備(Co-ordinator & End-Device)的 IEEE 802.15.4 堆疊層進行物理層與媒體接收層的初始化(設定使用第 11 至第 26 通道，並且設定成非信標網路)。在初始化完成後，會選擇一個 Co-ordinator 建立成 PAN Co-ordinator，並且對其設定 PAN ID(Personal area network identifier)和 Co-ordinator 短位址。在此 PAN ID 會被作為網路標籤，讓其他節點辨識，是否為節點要加入的網路。接下來 PAN Co-ordinator 必須選擇一個射頻頻率通道來建立網路。在選擇通道的過程中，PAN Co-ordinator 會進行能量掃描檢測來尋找

一個相對乾淨的通道，來建立網路。找到通道後，PAN Co-ordinator 就會開始其工作，等待裝置發送訊息要求加入網路中。而其他的裝置節點在初始化完成後，會在區域內對頻率通道進行掃描動作。當節點裝置在掃描到對應的 PAN Co-ordinator 時，會向 PAN-Co-ordinator 發送加入網路的要求訊息，並等待 PAN Co-ordinator 將它加入到網路中。當 PAN Co-ordinator 將裝置節點加入到網路中，會為裝置節點配置一個 16 位元的短位址，當作在網路中的標籤。最後當網路中出現了 PAN Co-ordinator 和一個裝置節點連結後，網路即可進行資料傳輸。

而資料傳輸有分成 Co-ordinator 向 End-Device 傳送資料與 End-Device 向 Co-ordinator 傳送資料。在 Co-ordinator 向 End-Device 傳送資料中，又有分直接傳輸與間接傳輸。直接傳輸是 Co-ordinator 可以將資料直接發送給 End-Device。End-Device 接收到資料後，會發送確認訊息給 Co-ordinator。但這種傳輸方式會要求 End-Device 隨時都處於資料接收的狀態，必須隨時處於喚醒狀態。間接傳輸是使用另外一種傳輸方式。Co-ordinator 會先將資料保存起來，等待 End-Device 請求讀取資料。用這種方式，End-Device 為了獲得資料必須先發送資料請求訊息。當 Co-ordinator 接收到請求訊息，會先發送訊息告知 End-Device 要傳送資料，在緊接著發送資料給 End-Device。當資料都接收完成，End-Device 會回復訊息給 Co-ordinator 告知資料已經收到。這一種傳輸方式，可讓 End-Device 設備進行休眠，當 End-Device 有需要時再喚醒進行傳輸動作，能讓 End-Device 處於低功耗的狀態下。而 End-Device 對 Co-ordinator 傳輸資料就只有一種，End-Device 向 Co-ordinator 直接發送資料，Co-ordinator 接收到資料後會發送確認訊息。

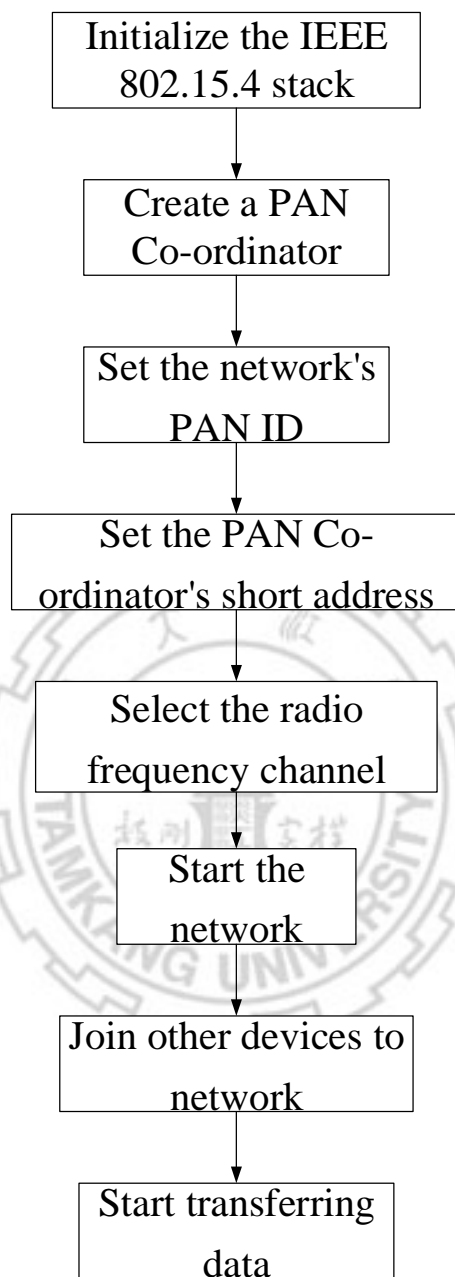


圖 3.8 ZigBee 網路建立流程圖[19]

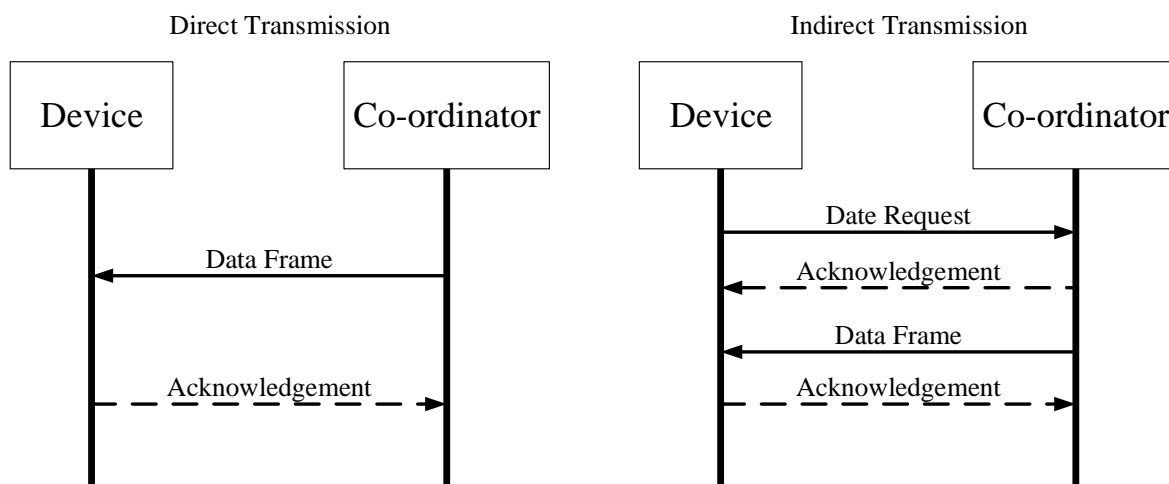


圖 3.9 Co-ordinator 至 End-Device 之資料傳輸[19]

3.2.1 ZigBee 協調器 (Co-ordinator)

在前一節中，說明源鼎科技提供的無線應用開發應用程式的軟體架構與網路建立流程，而本文只針對應用層進行功能編寫，接下來會描述在 Co-ordinator 與 End-Device 應用層中設計的功能。

在本文的系統架構中，Co-ordinator 除了原本的發起網路、搜尋附近裝置並將其加入網路工作。另外 Co-ordinator 還要負責接收 APP 的控制訊號、回饋環境品質資料與 End-Device 之間的資料傳輸。為了避免 Co-ordinator 誤判接收資料的來源，在接收到資料的同時為其加上來源標籤。之後程序在處理使用資料時，透過標籤即可辨別資料來源是 APP 或者是 End-Device。另外本文在應用層中設計了一個代理人機制，透過代理人機制，ZigBee 傳輸系統會先處理 End-Device 回傳的環境數據，並且將其存放在 Co-ordinator 的記憶體中。當 APP 在詢問資料時，Co-ordinator 在從記憶體中取出並回傳給 APP。而圖 3.10 為 Co-ordinator 與 End-Device 執行流程。

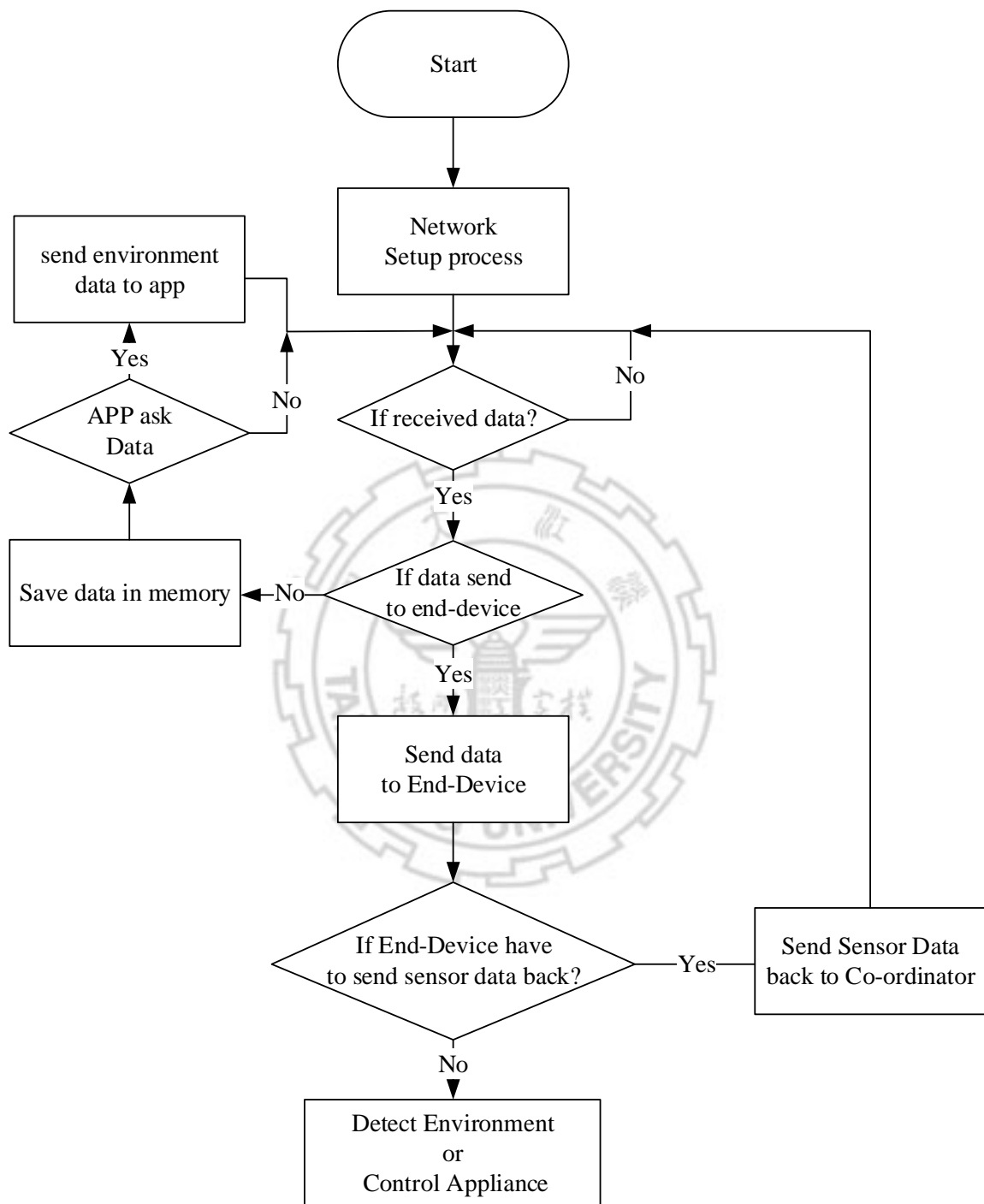


圖 3.10 ZigBee 傳輸系統流程圖

3.2.2 ZigBee 感測節點和家電管控節點

在 End-Device 分成感測端點與家電管控節點。而感測端點如圖 3.11，使用 FT6251 感測開發版。在開發版上有個溫度與濕度感測器，可偵測室內的環境數值。因此當感測端點在接收到更新環境數據指令時，就會開始偵測環境數值，當偵測完成後，會將數值回傳給 Co-ordinator。而家電管控節點和智慧家電控制板一起搭配使用。透過家電管控節點上的 Digital I/O，可輸出高電壓與低電壓，控制智慧家電控制板啟動與關閉。例如當節點接收到啟動家電控制訊號時，控制開發板上的 Digital I/O 會輸出高電位，家電控制板便可形成通路，家電即可進行工作。另外，為了節能的目的是，本文在 End-Device 中設計了一個計時器，這個計時器會計算家電使用的時間，並且回傳給 Co-ordinator。之後 APP 透過使用時間，再與電器的使用瓦數互相計算，便能得知用電量。使用者透過用電量，就能清楚得知節能效果。

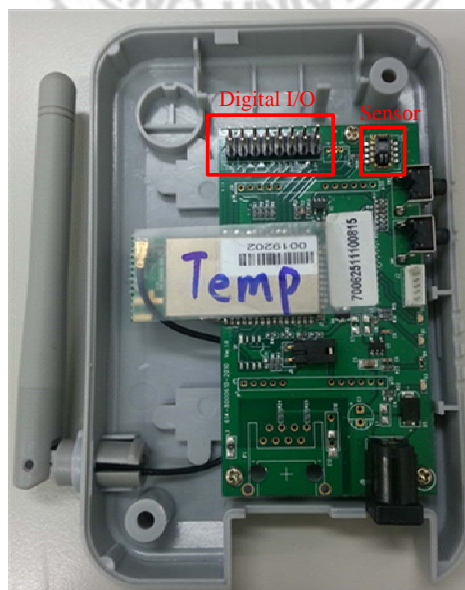


圖 3.11 ZigBee 感測節點

3.2.2.1 智慧家電控制板

為了讓 End-Device 可以控制家電啟動與關閉，在此設計了智慧家電控制板如圖 3.12。智慧家電控制板是使用繼電器、110V 插座、110V 插頭及兩根 pin 腳做成的簡易開關。在電路板上，將插座、插頭與繼電器連接，形成一個迴路，而控制繼電器的通路與段路的輸入腳位，則是與 End-Device 的 Digital I/O 互相連結，透過 I/O 輸出高電位電壓或低電位電壓來進行控制。



圖 3.12 智慧家電控制板

第四章 系統情境模擬驗證與節能比較

在本章節中，為了確保系統在不同的情境下都能正常運作，使用三種不同的情境進行測試，驗證家電節能管控系統在不同的情況下，都能正常工作。

4.1 情境一：溫、濕度 Sensor 測試

在節中，本文會針對溫溼度感測器進行測試。在實驗開始後，先進行常溫測量如圖 4.1(b)。接下來使用手指對感測器進行觸碰動作，用手指緊壓感測器 20 秒如圖 4.1(c)。在手指緊壓過程中，溫度小幅上升，濕度上升較為明顯。

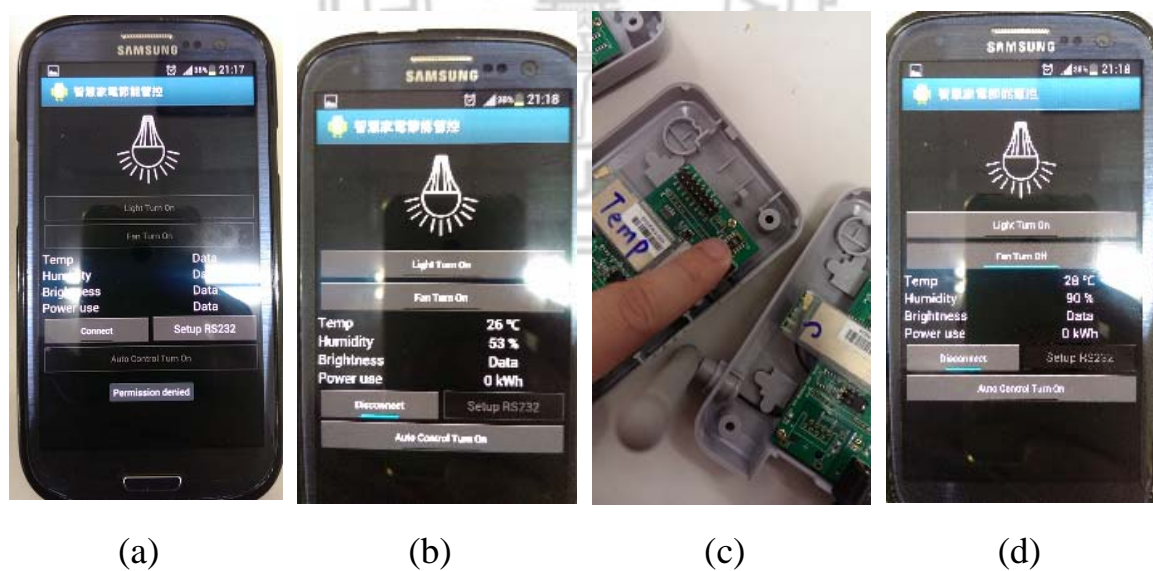


圖 4.1 溫、濕度 Sensor 測試

(a)啟動 APP (b)常溫下溫溼度量測

(c)手指壓住感測器 (d)手指壓住後量測到的數值

4.2 情境二：家電設備手動控制測試

在情境二中，會對系統進行家電操控測試，確認系統能夠正常的驅動家電工作。在實驗的過程中，會透過 APP 進行手動操控家電的動作如圖 4.2，透過 APP 的操控按鈕，點擊按鈕來啟動電風扇與檯燈。

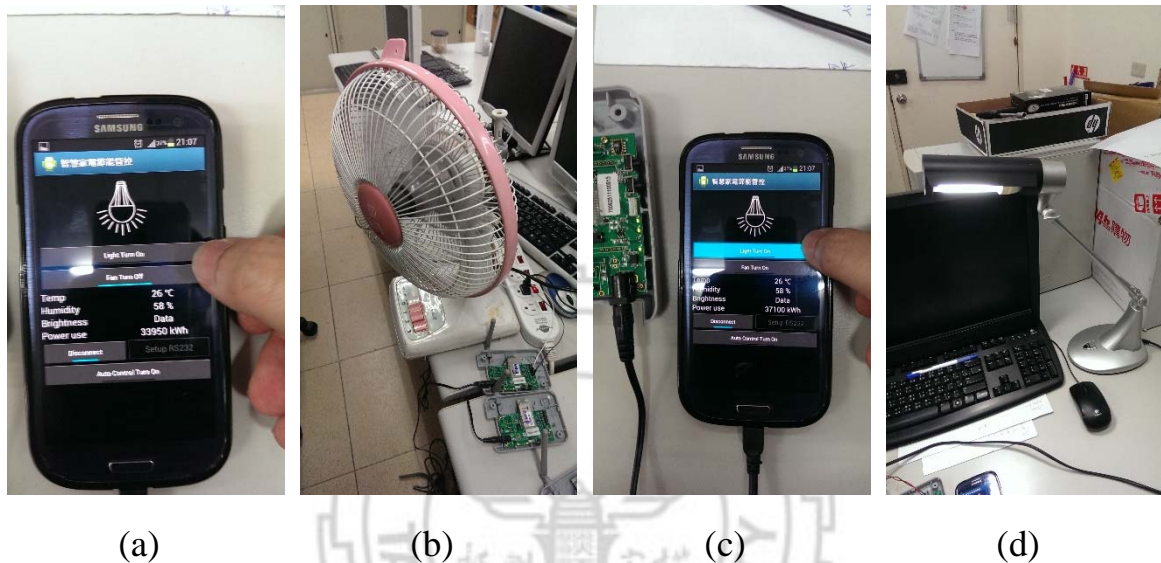


圖 4.2 家電手動操控測試

(a)手動操控電風扇啟動 (b)電風扇正常啟動

(c)手動操控檯燈啟動 (d)檯燈正常啟動

4.3 情境三：使用溫溼度數值進行自動控制測試

在情境三中，會進行家電自動控制測試。透過使用正常數值與不正常數值進行測試。而測試使用的正常數值範圍以人體舒適範圍當作界線，溫度以 26~28 度為舒適界限，濕度低於 60% 為舒適範圍。所以當超過 28 度或者濕度大於 60%，就會啟動電風扇調整環境。在圖 4.3 中，系統啟動自動操控模式，並且先進行量測環境數值。透過人為製造不正常環境數值。系統在偵測到不正常的環境數值後，會開啟電風扇調整環境。

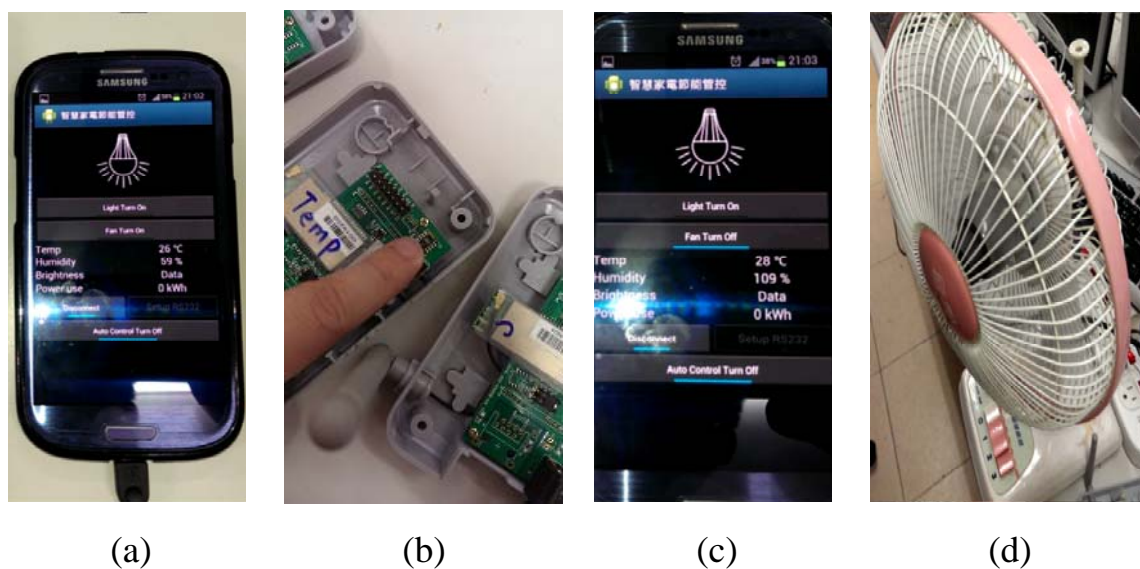


圖 4.3 使用環境數值進行自動操控

(a)正常環境數值 (b)透過人為製造不正常數值

(c)偵測到不正常數值起 (d)APP 自動操控電風扇啟動

4.4 節能比較

在這節中，本文會使用系統進行一整天的連續測試。讓系統在自動操控模式下，依據數值自行操控家電啟動關閉。所計算出來的電力消耗與家電連續開啟一整天電力消耗進行比較。在此使用的家電是電風扇，消耗功率為 45W。另外由於電風扇瓦數太小，這邊以一天用了多少瓦特作比較。

電風扇在用連續運轉一整天後，消耗的瓦特數為 $45W * (24 * 3600)$ 一天的秒數，總共 3888000Ws(單位:瓦·秒)。而使用自動控制所測得的結果如圖 4.5 為 3175245Ws。兩者相差了 712755Ws，換算成用電單位為 0.197 度。相比下來，使用自動控制模式能節省 18% 電力。因此家電節能管控系統，在自動控制模式下能有效的達到節能的效果。



圖 4.4 節能測試使用電風扇與消耗功率

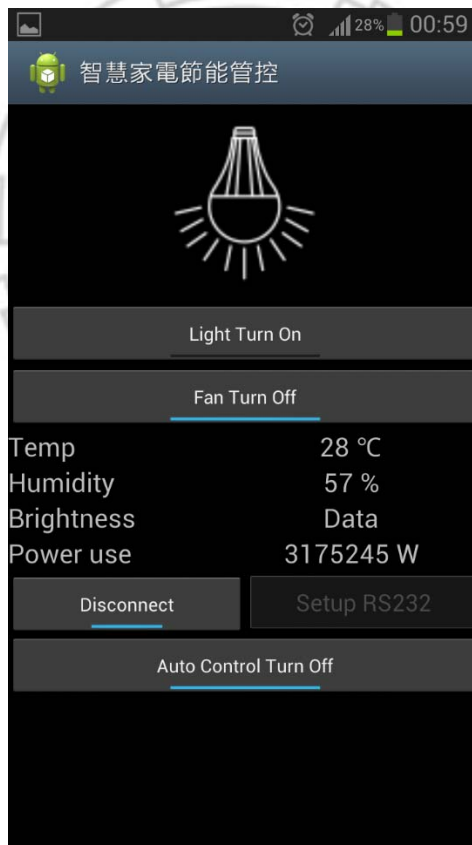


圖 4.5 使用自動控制模式連續測試

第五章 結論與未來展望

5.1 結論

智慧家電節能管控系統，透過使用智慧型手機與 ZigBee 無線傳輸系統，可大幅降低智慧家庭的建置成本。而使用 USB OTG 當作與 ZigBee 之間的傳輸媒介，能夠避免掉以往使用其他傳輸媒介的複雜連線設定，如藍芽必需進行配對連線，Wi-Fi 必需使用密碼連線。而 USB OTG 只須接上手機，按下連接即可開始通訊。而且使用家電節能管控 APP，系統可以自動調節室內環境，保持室內的舒適度與保護使用者的健康。更因為自動控制功能，系統可以避免家電在不需要的時候持續運轉，而消耗過多的能源。

5.2 未來展望

以目前的智慧家電節能管控系統，能夠管控的家電數量並不多。並且家電節能管控 APP 也未加入記錄功能，家電設備的用電資訊在 APP 關閉時就會被移除，無法統整紀錄長期的用電資訊。在未來我們希望能夠增加管控的家電數量，讓家電節能管控系統更加全面化。並且在 APP 中新增紀錄功能，在每次重新啟動 APP 後，能夠接續前一次的用電資訊繼續記錄，整理出一整個月或者一整年的完整資料。而在能量消耗計算，目前系統中沒有可以直接量測電器的電流量設備，而是以估算的方式計算使用電力度數。對使用者而言並不方便。因此希望未來能夠在系統中加入電流偵測設備，讓使用上更加便利，計算能源消耗也可更加精確。

在 ZigBee 傳輸系統中，我們在初期設計時，是以小坪數家庭當為背

景設定，所以 ZigBee 結構是使用了星狀拓樸。但一般家庭中，也有住家是三層樓或四層樓的透天厝住宅。而 ZigBee 在這種環境下就必須更改傳輸結構，讓網路覆蓋率變得更大。並且三層樓或四層樓住宅的空間是往上延伸，因此在布建 ZigBee 時就必須考慮到 Co-ordinator 布建位置。讓每一個樓層的 ZigBee 網路都能互相連結。最後 ZigBee 傳輸系統的整個工作週期中，會有節點裝置停止工作且退出網路的狀況發生。而本文未針對這個問題進行解決。希望在未來能夠設計一個完善的裝置加入或退出網路機制。讓整個 ZigBee 傳輸系統在運作更有效率。



參考文獻

- [1] 智慧生活中華電信智慧家庭，
<http://www.fbblife.com.tw/34244877/article/content.aspx?ArticleID=1706>.
- [2] ZigBee Alliance, <http://www.zigbee.org/>
- [3] RAMLEE, R. A., et al. “Smart home system using android application,” In: *Information and Communication Technology (ICoICT), 2013 International Conference of. IEEE, 2013. pp. 277-280.*
- [4] S. HUSSAIN, S. SCHAFFNER, D. MOSEYCHUCK, “Applications of wireless sensor networks and RFID in a smart home environment,” In: *Communication Networks and Services Research Conference, 2009. CNSR'09. Seventh Annual. IEEE, 2009. pp. 153-157.*
- [5] 高誌謙，“基於 Android 與 ZigBee 之智慧家庭設計與實現，” 電訊工程系，國立高雄海洋科技大學，高雄市，2011.
- [6] D. HAN, J. LIM, “Smart home energy management system using IEEE 802.15. 4 and ZigBee,” *Consumer Electronics, IEEE Transactions on*, 2010, 56.3: 1403-1410.
- [7] LEE, Jin-Shyan; SU, Yu-Wei; SHEN, Chung-Chou. “A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi.” In: *Industrial Electronics Society, 2007. IECON 2007. 33rd Annual*

- Conference of the IEEE*. IEEE, 2007. pp. 46-51.
- [8] ZHANG, Yikui, et al, “A solution for low cost and high performance smart home networking,” In: *Engineering and Industries (ICEI), 2011 International Conference on*. IEEE, 2011. pp. 1-6.
- [9] Android, <http://developer.android.com/sdk/index.html>
- [10] BRADY, Patrick. “Anatomy & physiology of an Android,” In: *Google I/O Developer Conference*. 2008.
- [11] Android Activity, <http://developer.android.com/reference/android/app/Activity.html>.
- [12] USB OTG 規格簡介, <http://www.usblab.idv.tw/usbotg.html>.
- [13] 許永和, *USB 規格與理論剖析*, 儒林出版社, 2009.
- [14] 802.15.4 IEEE Standard for Information technology Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks(LR-WPANs).
- [15] 李俊賢, “無線感測網路與 ZigBee 協定簡介,” *電信國家型科技計畫*, 2006.元月號 77 期.
- [16] ZigBee and Wi-Fi RF Channels, <http://fosiao.com/content/zigbee-and-wifi-rf-channels>.
- [17] Jennic, “IEEE 802.15.4 Wireless Networks User Guide,” JN-UG-3024, Revision 1.1, 2006.
- [18] PL2303 driver, <https://code.google.com/p/indiserver/>

- [19] Jennic, “IEEE 802.15.4 Application Development Reference Manual,”
JN-RM-2024, Revision 1.3, 2007.

