

國立交通大學

電信工程學系

碩士論文

室內火警感測網路之設計與實作

Design and Implementation for an
In-Building Fire Emergency System based on
Wireless Sensor Network



研 究 生：江宗嶽

指 導 教 授：張仲儒 教授

曾煜棋 教授

中華民國九十四年一月

室內火警感測網路之設計與實作

Design and Implementation for an
In-Building Fire Emergency System based on
Wireless Sensor Network

研 究 生	：	江宗嶽	Student	：	Tsung-Yueh Chiang
指 導 教 授	：	張仲儒 教授	Advisor	：	Chung-Ju Chang
		曾煜棋 教授			Yu-Chee Tseng

國立交通大學

電信工程學系

碩士論文

A Thesis Submitted to
Department of Communication Engineering
College of Electrical Engineering and Computer Science
National Chiao-Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Communication Engineering

January 2005

Hsinchu, Taiwan, Republic of China

中華民國九十四年一月

室內火警感測網路之設計與實作

研究生：江宗嶽

指導教授：張仲儒 博士

曾煜棋 博士

國立交通大學電信工程學系（研究所）碩士班

摘 要

隨著科技與時代的進步，促使無線網路的相關技術逐漸被運用於許多重要的領域，尤其是近來備受矚目的「無線感測網路」。將龐大數量的感測器部署於目標環境當中，這些微小的電子裝置均配備著感測、運算與通訊的功能，彼此之間可透過合作與協調的方式，共同達成整體任務，藉此以增進人類生活上的便利與舒適性。在本篇研究論文中，我們將針對室內建築環境的消防應用情境，設計出一套智慧型的火警感測網路系統—「PHOENIX」，本系統運用前瞻性的無線感測網路技術，使得在佈局上不僅能達到簡易與美觀的優點，同時亦能兼顧安全性的考量，其主要提供的服務項目包括：室溫偵測、災情回報以及逃生導引等三大功能，改良現有以實體線路為基礎的建築消防系統，更能夠有效地降低火警災變所造成的傷害影響；我們將採用美國加州柏克萊大學所研發的微塵感測器硬體平台（Mote-MICA2），結合嵌入式微型作業系統（TinyOS）來實作此套系統，相關於系統細部的架構配置、協定流程與效益評估將一併呈現於本文當中。

關鍵字：無線感測網路、室內火警系統、室溫偵測、災情回報、逃生導引。

Design and Implementation for an In-Building Fire Emergency System based on Wireless Sensor Network

Student : Tsung-Yueh Chiang

Advisor : Prof. Chung-Ju Chang

Prof. Yu-Chee Tseng

Department (Institute) of Communication Engineering
College of Electrical Engineering and Computer Science
National Chiao-Tung University



ABSTRACT

Due to the progress of technologies, wireless networks have been widely applied to many important domains, such as wireless sensor networks. The objective of sensor networks is to detect context information of the surrounding environment. Each sensor node has sensing, computing, and communicating capabilities. Sensors cooperate with each other to achieve this goal. In this thesis, we design an intelligent fire emergency system called PHOENIX for in-building, fire-fighting applications based on wireless sensor networks. The system has three major functionalities, including temperature reporting, emergency reporting, and emergency guiding. Because of its wireless communication capability, the vulnerability of wire-line systems is greatly improved. The system is developed based on the Mote-MICA2 hardware platform released by the University of California at Berkeley using a tiny operating system (TinyOS). The thesis reports in details how the PHOENIX system is implemented and operated.

Keywords: Wireless sensor network, Fire emergency system, Thermometric detection, Emergency report, Survival guidepost.

誌 謝

誠摯地感謝曾煜棋教授兩年來的指導與鼓勵，並提供良好的研究環境及充足的實驗設備，讓我得以順利完成此篇論文並取得碩士學位。

此外，也由衷地感謝實驗室裡一起工作的研究同儕們，感謝你們在我創作此篇論文時的參與討論與提供意見。

最後，感謝家人對我的期許及關懷，尤其是母親對我多年的教養及栽培，僅以此論文獻給我的母親。

在此向你們獻上我最誠摯的謝意。



江宗嶽謹識於
國立交通大學電信工程研究所
中華民國九十四年一月

論文目次

中文摘要	i
英文摘要	ii
誌謝	iii
目錄	iv
表目錄	vi
圖目錄	vii
第壹章、緒論	1
第一節、研究背景	2
第二節、研究動機與目的	3
第三節、研究方法與流程	4
第四節、論文結構	5
第貳章、相關文獻	6
第一節、感測器之組成構造與硬體平台	7
第二節、微型作業系統 — TinyOS	10
第三節、無線感測網路之架構及其應用	13
第四節、室內火警感測網路之相關技術探討	15
第參章、室內火警感測網路「PHOENIX」之服務功能	20
第一節、系統綜觀與環境佈署	20
第二節、功能項目—「室溫偵測」之設計與研究	21
第三節、功能項目—「災情回報」之設計與研究	25
第四節、功能項目—「逃生導引」之設計與研究	33
第五節、研究結語	35

第肆章、室內火警感測網路「PHOENIX」之系統實作.....	36
第一節、通訊協定堆疊與訊息封裝.....	37
第二節、系統架構與運作流程.....	42
第三節、系統實際操作與示範.....	47
第四節、效益評估.....	52
第伍章、未來展望與結論.....	56
第一節、未來研究方向.....	56
第二節、本文研究貢獻.....	58
陸、參考文獻.....	59



表目錄

表(一)、網路初始化程序。	23
表(二)、鄰居偵測程序。	24
表(三)、交錯展延樹之演算法 (A)。	28
表(四)、交錯展延樹之演算法 (B)。	30
表(五)、逃生導引程序之示範圖例。	34
表(六)、訊息封裝格式。	40
表(七)、訊息列表總覽。	42
表(八)、感測器之顯示燈號與表示意義。	47
表(九)、感測器之傳輸實驗表。	53
表(十)、感測器之感測實驗表。	54



圖目錄

圖(一)、感測器之構造組成圖。.....	8
圖(二)、微塵感測器之硬體結構圖。.....	9
圖(三)、訊息通訊元件之圖解範例。.....	12
圖(四)、系統部署圖。.....	21
圖(五)、完整圖 (K5) 之四棵交錯展延樹範例。.....	27
圖(六)、交錯展延樹演算法之執行結果。.....	30
圖(七)、鄰居偵測程序之圖解範例。.....	31
圖(八)、區域性廣播之圖解範例。.....	32
圖(九)、PHOENIX 系統之通訊協定堆疊。.....	38
圖(十)、協定堆疊模組之連接關係圖。.....	39
圖(十一)、系統實際示範圖。.....	42
圖(十二)、PHOENIX 系統之資料與控制流路示意圖。.....	43
圖(十三)、PHOENIX 系統之元件配置圖。.....	44
圖(十四)、二階段式回報機制流程圖。.....	46
圖(十五)、PHOENIX 火警系統之圖形介面。.....	49
圖(十六)、PHOENIX 火警系統之執行實例。.....	50
圖(十七)、PHOENIX 火警系統之實際操作圖。.....	52

第壹章、緒論

隨著微機電整合、無線網路及嵌入式處理技術的迅速提昇，促進了微小的電子裝置能夠具備感測、計算與通訊等多樣性功能，將這些微型感測器恣意地部署於觀察環境當中，其所形成的網路稱之為「無線感測網路（Wireless sensor network）」[1][2][3]。這是一種以應用趨勢為導向的網路型態，主要的任務是透過感測器在其感應範圍內偵測週遭環境或特定目標，並將所搜集到的資訊經由無線傳輸方式，回報到基地台，透過系統提供進階的服務功能，進而提升人類的各項生活品質。

目前亦有許多無線感測網路的相關應用[4][5][6]正在持續地發展中，實際的應用範疇相當地廣泛，涵蓋了商業、軍事、教育與工業等諸多領域。事實上，無線感測網路提供了一種有效取得環境資訊的途徑，而在普及運算所探討的議題裡，令應用系統能夠動態地感知環境資訊便是一項極為重要的題材，系統透過感測器所蒐集的資訊，交由基地台進行進階的判斷、處理，讓系統利用這些訊息得以自我調整與維護。我們可預知在不久的未來裡，伴隨著這項新興科技的引領，勢必將對人類生活的影響層面與日俱增，促使人類進入一個真正的環境感知運算（pervasive context-aware computing）時代。

在本篇論文當中，我們將探討如何運用無線感測網路來感知環境中的資訊，並且設計一套適用於室內建築物的火警感測網路，嘗試取代現行諸多弊病的有線傳統消防監測系統。在以下將深入介紹本系統之初，本章將闡明我們研究這項主題的背景、動機以及研究範圍。

第一節、研究背景

近年來國際間「無線感測網路」的相關研究備受矚目，更引起了學界與業界的廣泛討論，其應用範圍無遠弗屆，涵蓋了國防、工業、醫療及保全等各種層面，適當地運用這種能夠自我組態的網路模式，不僅能大幅降低以往對於人力資源的需求，並且能高效率地增加工作效率。

我們的研究背景乃立基於目前已趨成熟的無線隨意網路（Wireless ad-hoc network）[7]，隨意網路是無線感測網路技術的前身，它們皆為無基礎設施（infrastructure-less）的網路型態，具有許多相通的特性，但感測網路的建構條件更為嚴謹，使得許多現今通行於隨意網路上的協定無法直接套用於感測網路上，因此在設計感測網路上至今仍存在著許多嶄新的挑戰，歸納兩種網路主要的差異性如下：

1. 網路範疇不盡相同

無線感測網路的節點數量通常較無線隨意網路超出數十倍甚至百倍，而且感測網路的節點密度普遍較高，這點將使得許多利用廣播通訊方式為基礎所發展出來的隨意網路協定將不再適用於感測網路。

2. 缺乏共同識別碼

由於無線感測網路的節點數量龐大，因此每一個網路節點或許沒有唯一性參考指標，用來作為網路中的共通識別碼（identifier），這點使得無線感測網路的繞徑協定設計不同於無線隨意網路。

3. 裝置能力明顯不同

感測器的硬體設計以成本、體積與耗電為主要考量，因此在無線感測網路中，裝置節點本身所配備的電源、記憶體以及運算能力均受到極大的限制，這些限制相對於無線隨意網路中的行動主機來說，受限制的幅度則輕緩許多。

以無線隨意網路領域中著名的繞徑協定為例，譬如 DSR[8]與 AODV[9]這兩套網路協定便無法直接套用於無線感測網路上，一般而言感測網路常為顧及電源與效能等考量因素，會在資料回報前，執行一些加工處理的程序，其中最關鍵的技術包括有資料彙整 (data aggregation) [10]與資料融合 (data fusion) [11]兩種，但是 DSR 與 AODV 等協定均不能允許資料在彙整或融合後繼續執行，使得無法直接運用於無線感測網路上。

縱使目前在無線感測網路的設計上，仍須面臨許多的困難與挑戰尚待解決，但我們深信這些技術問題必然能夠在大家的共同努力之下，被逐一克服，屆時無線感測網路必能挾以其本身所具備的體積、廉價等諸多優勢，成為領導主流的先驅技術之一。

第二節、研究動機與目的

追求舒適與無憂無慮的生活環境，一直是現代科技努力的目標。伴隨著電機與資訊產業的蓬勃發展，將無線感測網路應用於人類的日常生活中，不再只是電影情節中的虛擬場景，亦非富豪所能獨享。目前在國際間，已經有許多著名的研究團隊正在積極地開發有關無線感測網路的相關技術與各種應用[4][5][6][12][13]。有鑑於此，本研究利用部署於室內的無線感測網路，開發出適用於火警消防的應用系統。

傳統的建築火警系統中，主要是以溫度計或煙霧探測器來偵測所在建築環境內是否發生火災，一旦祝融興起，便立即啟動對應的滅火設備 (fire extinguisher) 來執行預先設定的消防措施，請求就近的消防單位與予支援。在這樣的系統中，所有的探測器是依靠鋪埋於建築物內的實體線路與控管中心進行連結，探測器的運作是彼此獨立的，系統亦不提供長期偵測與紀錄室溫的功能；無疑地，系統的擴充性受到所在環境極大的限制，對於線路的維護也是一項棘手的問題。相較之下，無線感測網路擁有的諸多優點

便脫穎而出，感測器微小的體積可以無礙於空間美觀，無線通訊的形式使得系統佈置相當簡易，透過適當的協定規劃使得感測器，能夠自我組態形成消防互聯網路，提供室溫偵測與傳報功能，充分地發揮出智慧型急難管理的目標。

第三節、研究方法與流程

本研究論文的主題為「室內火警感測網路」，我們首先探討其本質特徵及各項關聯技術，詳盡地執行整理與歸納的工作，確認研究對象的構成模式與規律性，再進而提出系統架構與組成元件的設計概念，透過實驗模擬與實作示範的方式達到對研究對象的完整掌握。在研究進行時，我們所採用的方法，包含以下幾個步驟：

一、文獻資料整理

藉由國際間曾經被發表或刊登的期刊與論文集當中，整理出相關的文獻並透過書面資料的記載，了解無線感測網路的發展現況與研究近程，進而探討各項相關的技術及其優缺點，歸納出所參考的文獻資料，確實掌握研究方向與核心技術，以利於系統之規劃、設計。

二、系統建構與分析

首先分析整體系統所必須注意的所有考量因素，包括：省電、效率與回報機制，透過此步驟思考面對於種種現實問題的解決之道，進而設計出實用的室內火警感測網路系統。

三、系統效能評估

驗證本系統的各项參考數據，反覆查驗以確認系統是否符合預期目標，若有缺陷損及預期之目標，則重新思考找尋原因，進而修正系統以達成預定目標。

四、雛形系統實作

在資源與時間有限的條件下，為求驗證系統於實際面的可行性，本研究著手實作雛形系統，結合柏克萊大學所設計的感測網路端點硬體平台，與微型作業系統，並輔以適當的理論進行測試與調校，增進系統的穩定性。

第四節、論文結構

本篇研究論文之架構共區分為五個章節：第壹章為緒論，闡明本研究之背景、動機與研究方法，第貳章為相關文獻，介紹感測器的組成構造、硬體平台與微型作業系統，並歸納出目前無線感測網路的架構與應用方向，在第參章當中，則針對於室內火警的實際情境提出一套消防應用系統，分別按照三大主要服務功能：室溫監測、災情回報與逃生導引，逐一探究並加以討論，第肆章利用加州柏克萊大學所開發的 Mote-MICAII 感測器裝置結合 TinyOS 來實作本網路系統，同時將實作上的設計考量一併呈現，於第五章中則是本研究論文的結論與未來發展方向，以供為後續研究之參考。

第貳章、相關文獻

在著名影星湯姆克魯斯所主演的一部好萊塢電影「關鍵報告」中，描述未來的年代裡，人類的生活將無時無刻地受到監控，在每一道房門或關卡處皆會有一具智慧型攝影機，透過瞳孔、手紋辨識程序確認身分後，將資料傳回到監控中心，監控中心可以隨時取得城市裡所有位置的地理座標，片段中更令人嘖嘖稱奇的就是那群自動緝捕主角的蜘蛛型機器人，每隻蜘蛛都是獨立的個體，不僅可以在行動間進行偵測，當遭遇特殊狀況時還可以聚集起來，相互支援與協調。雖然虛構的劇情中對於未來世界的描述非常誇張，但可預知在伴隨著無線感測網路等相關技術的提昇，或許到那個年代時像「伊拉克自由行動」¹ (Operation Iraqi Freedom) 這類的任務，有可能在最短時間內達成。

無線感測網路是由數目龐大的無線感測器所組成，這項原本遙不可及的科技，現今配合著微機電整合系統 (MEMS) 及奈米科技 (Nano science) 等技術，確實能夠得以實現。早在三年前，由美國國防部高等研究計劃總署 (DARPA) 協同加州柏克萊大學 (UC Berkeley) 所發展的「智慧微塵 (Smart Dust)」[12] 計畫當中，提出了一種微型的無線感測硬體平台 (Mote) [14]，研究人員在這個硬體平台上實踐了微小電子裝置可配備微電腦、無線通訊與感測模組的目標，每個感測器就類似於微塵一般，透過支援嵌入式處理的微型作業系統 (TinyOS) [15] 即可操控。

下面我們將透過本章相關的文獻整理，進一步地了解何謂微型感測裝置的組成構造以及運行於其上的微型作業系統，同時歸納出目前無線感測網路的架構與現行應用。

¹ 西元 2003 年 3 月份，美國派遣特種部隊進入伊拉克境內蒐集情報，搜索攻擊目標，此次的攻伊行動定名為「伊拉克自由行動」(Operation Iraqi Freedom)。

第一節、感測器之組成構造與硬體平台

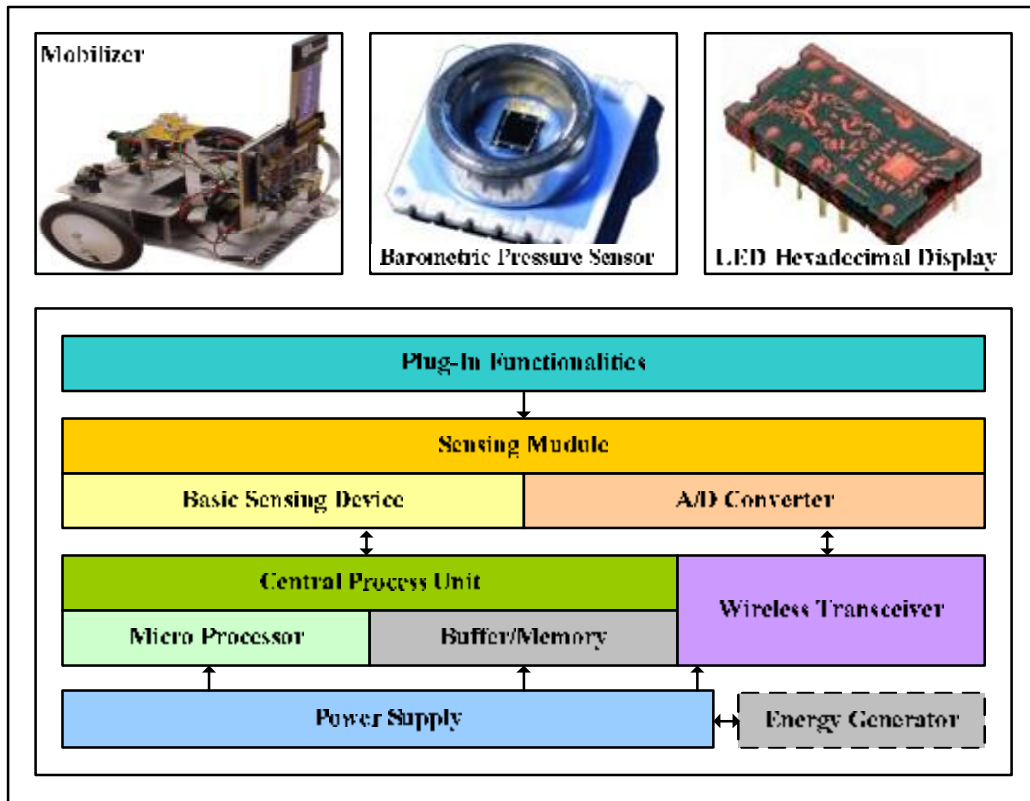
本節中分為兩個部份，首先將概觀性地介紹微型感測器的硬體構造，再針對於本篇研究論文中系統實作時所採用的微塵（Mote）感測器硬體特別說明。

(一) 感測器之組成構造及其功能

一般而言，感測器的構造組成及其功能[1]，如下頁圖(一)所示，主要包含四個部份：中央處理器（central processing unit）、無線收發器（wireless transceiver）、感測模組（sensing module）和電源供應器（power supply）。針對於不同的應用，感應器本身亦能外掛（plug-in）其他附加元件，例如：磁性方位器（magnetometers）、壓力計（pressure transducer）及陀螺儀（inertial measurement system）等設備。

感測器的主要功能就是偵測所在環境的相關參數及變異程度，當感測單元偵測到環境資訊時，會先透過類比／數位訊號轉換器，將類比訊號轉換成數位資料，再交由中央處理器執行進階處理，中央處理器除了擁有運算的功能外，亦附有一個小型的記憶體來暫存運算結果，此外由於鄰近的感測器之間可能會同時進行相同的任務，因此中央處理器亦需要能夠彼此協調、共同合作處理資料，而運算過後的結果將透過無線收發器回報到控制中心，電源供應器則是感測器當中最基礎的設備，它主要負責供應裝置所有的使用電量。另外，感測器可外掛全球定位系統模組（GPS）、震度探測器（SEISMIC）、加速度計（accelerometer）或是其他特殊應用可能所需的感測模組，藉以輔助感測器進行更精確的量測用途。

現今藉由微機電整合的技術提升，使得一顆感測器的實際尺寸已逐漸縮小，甚至如同一枚硬幣般大小，僅單板上便能擁有上述各項功能，這使得感測器運用的幅度大為增廣。

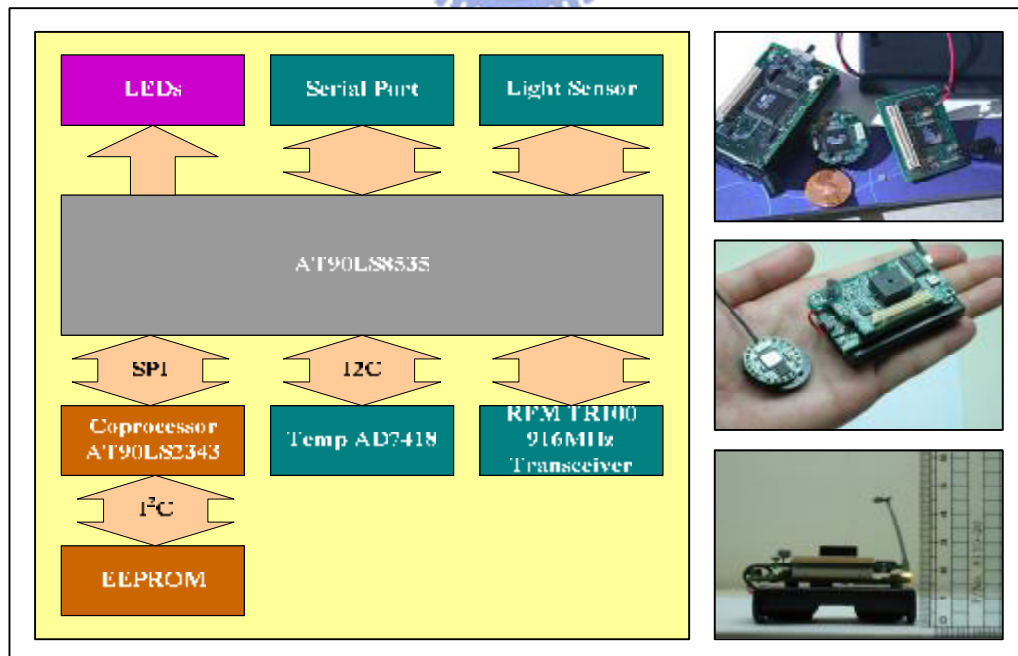


圖(一)、感測器之構造組成圖。

(二) 微塵感測器之硬體平台

由美國加州柏克萊大學 (UC Berkeley) 所研發的微塵感測器終端硬體平台，請參照下頁圖(二)，它採用 ATMEL 公司所製造的晶片 AT90LS8535 做為中央處理器，系統具有 8 K 位元組的可程式快閃記憶體與 544 位元組的隨機存取記憶體，採用精簡指令集設計的哈佛架構 (Harvard Architecture)，提供 16 位元的定址轉換空間；系統針對嵌入式應用提供三種省電模式：Idle、Power-down 以及 Power-save；當系統處於 Idle 模式時，會自動停止執行所有的工作排程；當系統處於 Power-down 模式時，僅剩下非同步 (asynchronous) 中斷和基礎的周邊設備尚可正常工作；倘若系統處於 Power-save 模式時，大致上運作行為與 Power-down 模式無異，但會多一組非同步計時器，以維持週期性的常態工作。微塵感測器的射頻電路採用 RF Monolithic 公司的 TR1000 通訊模組製成，這個模組可提供

916.50MHz 的操作頻率以進行短距離的無線通訊功能，其最高傳輸率可達到每秒 115.2 K 位元組，使用 3 伏特的操作電壓，可調整訊號強弱與敏感度，亦支援睡眠管理模式的設計。溫度感測單元則是 Analog Device 的 AD7418 晶片，為內建 A/D 轉換器、溫度感測器、時脈震盪器的整合晶片，利用 I²C 傳輸介面將數位訊號輸入至中央處理器。此外，系統裡有一個 EEPROM 記憶單元，中央處理器會透過 SPI 介面與八位元的協同處理器 ATMEL AT90LS2343 連接，由協同處理器去管理大小為 256K 位元組的 EEPROM 模組 24LC256。這裡值得一提的是，每項元件都具有低耗電的設計，而整體運作在全效模式下所使用的電壓為 3 伏特，以及 19.5 毫安培的電流量，粗略估計為 60 微瓦，當 LED 顯示燈全部開啟時，則增至 100 微瓦，如果系統在 Idle 模式下，則使得耗電量下降到 3.1 安培。目前這一套系統，透過業界的 Crossbow Technology[14]公司負責生產、製造，同時這家公司也持續地以專業的微機電整合技術，提供更多樣的周邊硬體，使得無線感測網路的應用更為廣泛，諸如壓力計、液晶顯示器等等。



圖(二)、微塵感測器之硬體結構圖。

第二節、微型作業系統 — TinyOS

隨著微機電整合技術與奈米技術的迅速發展，使得一個極小的電子裝置能同時擁有多樣化的功能，進而對人類產生卓越的貢獻。相對地，在這些微型設備上的軟體元件，亦需要極具巧妙的構思設計，而「微型作業系統 (TinyOS)」 [15] 正是一套專為小型記憶體的通訊裝置所設計之作業系統，最早的 1.0 版本是由柏克萊大學於 2002 年 10 月份所發表釋出，並宣告原始碼的版權為大眾公有版權 (General Public License)，發行至今透過廣大的社群共同參與並互相討論，已經新增了許多功能，目前最新的版本為 1.1.9，且能夠支援 Linux、WindowsXP 等數種電腦作業平台。

TinyOS 的架構是建構於元件基礎 (component-based) 和事件驅動 (event-driven) 兩大觀念，在開發的過程中，所有的程序在設計時皆需要從元件與事件的角度上來進行思考；基本上，TinyOS 是利用一個以元件為基礎的模型，來描述硬體與網路組成元件，每一個硬體裝置被描述成一個元件，統稱為「Graph of Components」，加上一個排程器 (Tiny Scheduler) 而建構成整個系統。

開發者可透過「NesC」程式語言來開發自訂的通訊演算規則，這是專為需要透過 TinyOS 平台來操作的嵌入式設備所設計的一套程式語言，它的語法類似於 C 語言，讓開發者可以迅速地適應它，進而迅速地專注於事件與系統元件的處理；TinyOS 本身亦提供了一套完整現成的函式介面給撰寫 NesC 的開發者所使用，可降低開發時的複雜度，大幅提增系統的開發期程。

在撰寫 NesC 語言的同時，有下列四項重要的觀念，必須謹記在心：

1. 資料框架 (Frame)

資料框架是用來描述該元件所需用到的記憶體資源。值得注意的是，一般在撰寫 C 程式時，會習慣用動態宣告的方式，等到執行階段 (run-time) 時再實際配置所需要的記憶體空間，但是在 TinyOS 的開發環境中，必須採用靜態配置的方式，讓程式於編譯連結階段即可把記憶體空間配置完成，而這種方式亦是在開發嵌入式系統的過程中常被運用的重要觀念。

2. 命令處理器 (Command Handler)

這裡所謂的命令處理，如同 SendMsg、RecvMsg 函式等，系統會將呼叫者 (caller) 的參數放在元件內部的資料框架中，然後接收來自於低階的硬體命令，即所謂的「command」，一旦系統接獲這些命令時，即查詢在元件中有註冊過的處理器，處理器本身其實就是一種 callback 函式，它會等待系統的中斷呼叫並執行適當的行為。

3. 事件處理器 (Event Handler)

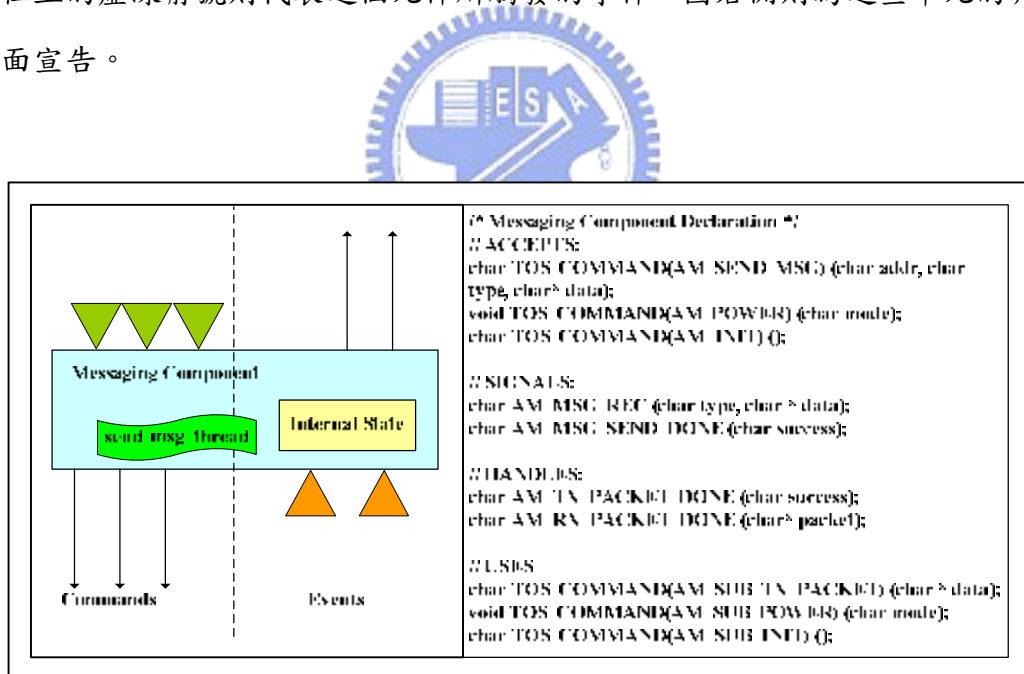
每一個事件是由硬體直接或間接觸發，例如計時器或收發單元的中斷訊號，當事件被觸發時，事件處理器會立即將此事件發生的相關資訊，暫存於元件內的資料框架中，然後進行後續的處理，後續處理程序可能會呼叫預設的 Task 或是執行低階 Command，同時在處理的過程中，可能會觸發到其他或更高階的事件，這些事件將被置入排程器佇列中依照優先權先後被執行。為了避免命令和事件的循環關係造成系統陷入死結或迴圈，TinyOS 規定所有的命令不能觸發任何事件。

4. 工作描述 (Task)

Task 中描述了最基本的工作程序，系統中的所有 Task 會透過工作排程器來安排執行，排程的概念類似於一般常見的分時多工作業系統，在 TinyOS 中工作排程器使用簡單的佇列結構來儲存 Task，並以先進先出 (FIFO) 方式來執行，如果應用上需要更複雜的處理，也可以修

改或增加其他排程演算法進去。Task 在執行期間可以呼叫低階的 Command 或觸發高階的 Event，或者安排元件內部的其他 Task 執行，執行的順序是依照系統設定的優先權而定，Task 的優先權低於事件的觸發。至於工作與電能控制方面，TinyOS 亦提供了睡眠模式的功能，倘若目前工作排程器的佇列內所有 Task 皆已經完成，系統將會進入睡眠模式，此時只有週邊的中斷事件可以喚醒系統。

下圖(三)為一個訊息通訊元件的例子，這個元件包含了內部的狀態描述 (internal state)、一組工作描述 (圖中倒三角形部分)、一組處理單元 (三角形部分)，左下角由上往下的箭號表示這個元件所用到的命令，而由下往上的虛線箭號則代表這個元件所觸發的事件，圖右側則為這些單元的介面宣告。



圖(三)、訊息通訊元件之圖解範例。

第三節、無線感測網路之架構及其應用

當初加州柏克萊大學所提出的「Smart Dust (智慧微塵)」網路計畫，是由美國國防部高等研究計劃局 (DARPA) 所支助的研究計畫案，起初的構想是由一個軍事上的應用開始，所想像的情境是在戰場上，我軍欲監控敵軍的戰情狀態，此時便可派遣無人駕駛的輕型自動飛機，上頭載著數以千計的微型感測器，當飛機抵達欲監控之區域時，將這些感測器直接灑下。這些具備感測與通訊能力的電子設備，其密度和外型理想上就類似於灰塵，俟塵埃落定於陸地時，自我組態形成所謂的「無線感測網路」，執行感測環境的任務，當我方需要敵情資料時，只需指派飛機掠過監測區上空以收集網路所探測到的資料，將這些敵情資訊帶回基地加以分析即可，藉以達到「知己知彼百戰百勝」的戰略目標。

由於無線感測網路是一種以應用為導向 (mission-centric) 的網路型態，針對不同的應用需求使得無線感測網路有許多種不同的架構，大致上，我們可依據感測器是否具備行動的能力，將網路分為靜態 (static) 與動態 (dynamic) 等兩種形式。對於靜態形式的無線感測網路，感測器是不會移動的，這類型的網路可以針對整體通訊系統設定最佳化運作模式，進而讓整體系統的工作效能得以提升。相對於動態形式的無線感測網路而言，由於感測器本身具備行動的能力，使得網路拓撲變動頻繁，通常要針對這樣的情形設計出最佳化演算法的困難度大幅增加，況且動態的網路需要經常再次進行最佳化安排程序，反而滯礙了網路本身的發展空間，或者浪費更多資源，一般來說，動態的無線感測網路僅會針對區域性進行最佳化調整。

無線感測網路的應用層面相當地廣泛，目前這項技術已經被預測為繼個人電腦技術之後，另一項更能直接融入人類生活的新興科技，下面概要地將其應用分為八大類，並條列其各個層面的應用性及潛力：

1. 應用於國防工業：

將無線感測網路部署於敵軍陣營中，避免實際派遣士兵前往勘查，不僅避免人員傷亡，還能夠有效地提升偵搜效率，感測器網路本身亦提供長時間的操作以提供監控戰場的功能；另外，可在我方的後勤裝備或軍火倉之中，加以部署感測器網路，以利掌握所有資源現況。

2. 應用於學術研究：

可將大量的感測器佈署於森林、海洋當中，藉以偵測生物活動、洋流、板塊等變化，例如將微型感測裝置寄放於鳥巢內，提供觀察候鳥習性、生態研究等等。

3. 應用於環境保護：

於各地下水道、工廠廢水排放口，設置能偵測化學污染之感測器，可立即檢測出受污染的程度，於平時亦可週期性紀錄所有污染源之化學成分、濃度等數值，避免這些環工人員親自進入這些危險受污區域。

4. 應用於醫療保健：

將無線且微小如晶片一般大小的感測器，適當地植入人體皮膚表層，或特定臟器附近，負責監控各項重要生理數據，避免凌亂繁多的線材阻礙患者行動，同時亦可發揮遠端照顧的功能。

5. 應用於數位家庭：

近來「數位家庭」、「家庭網路」的口號大為興起，利用與啟動器連接的感測器佈署於家中，讓主婦們在社區上課仍能經由網際網路執行許多家事，或當主人開車接近屋子時便自動地打開車庫等功能。

6. 應用於商業行為：

無線感測器網路也有許多適用於業界的應用，例如車廠可透過感測器網路監控車輛，賣場則用之來監控商品的失竊等等。或者如傳統辦公室的空調系統是中央控制，因此常使得室內某區域之間的溫差甚鉅，可利用無線感測網路監控各個角落的溫度狀況，智慧判斷並調整至適

合的溫度。

7. 應用於工廠管理：

在現今許多自動化的廠房當中，雖然昂貴的器械得以減低大量的勞工付出，但仍免不了需要部分的人員時時來監控這些器械，以避免差錯發生，倘若以無線感測網路來取代這些人工，則能夠取代這些管理機械的人員。

8. 應用於安全系統：

如本論文中所設計的系統般地，類似的功能例如建築物結構感測網路可即時偵測各角落的震動程度，即早判斷出牆垣圯倒的可能性，更甚之，在地震發生之前，通知內部人員儘速逃離。

目前，國際上有許多著名的相關研究計畫如：NEST[17]、SensoNet[17]等等，這些計畫的主要目標在於開發出適於普及用途之感應器網路平台供給各式各樣的應用來實作與測試。



第四節、室內火警感測網路之相關技術探討

無線感測網路所有的研究幾乎皆以降低能量消耗（energy consumption）為首要目標，然而這樣的要求使得演算法在設計上不應過於複雜，卻得同時兼具良好的執行效能。在我們接下開始研究室內火警感測網路之前，在此我們將相關需要注意的研究方向大致如下：

1. 硬體、架構、實體層(physical layer)以及媒體存取控制層(MAC layer)：

在硬體架構上，感測器本身必須提供多種電源管理模式，在[19]文中曾舉出過感測器有五種適用的睡眠模式，而這五種模式是分別針對中央處理器、記憶體、類比／數位訊號轉換器以及通訊收發器的開關控

制，讓感測器在任何時候抉擇適當的硬體部分來啟動；實體層與媒體存取控制層的協定主要是解決共用頻道的使用權問題，一般而言，感測器在實際傳輸時應該儘量減少傳輸碰撞的機率，換言之，如此可降低因碰撞而必須重新傳送的次數，就實際的運作過程中，無線傳輸所消耗的能量已被證實為感測器當中最主要的能量消耗部份，例如傳送 1 位元至 100 公尺處所需耗費的能量幾乎等於其內部處理器執行 3000 個指令的耗損。在[20][21][22]等文當中，學者們建議使用「分時多工 (TDMA)」，然而 TDMA 有兩個主要的技術瓶頸，首先是需要精確的時間同步 (time synchronization)，然而時間同步卻是另一項相當困難的問題，亦有不少學者提出相關的解決方案，如[21][23]等，但倘若無法精確地達到時間同步的前提，採用 TDMA 仍是無法有效地避免碰撞的發生；其二就是如果感測網路並沒有很高的資料傳送量，將會使得非常多的頻寬資源被浪費。另外一個方向也是利用「睡眠機制」的概念來設計，在 S-MAC[23]中，作者們參照 IEEE 802.11 的 MAC 系統中的睡眠排程 (sleeping schedule)，S-MAC 透過互為鄰居的感測器共同建立一個睡眠排程，讓彼此醒著的時間都能夠交集，不至於因為睡眠使得傳輸中斷，然而此方法也需要經常性同步維護。

2. 網路層 (network layer)：

就網路層來看，在隨意網路中的常見到設計者透過泛洪法 (flooding) 的機制，讓網路節點取得整體資訊，顯然這將有違於感測網路對於減輕耗電負擔的要求，SPIN[24]一文當中，主要是針對泛洪法導致資料被大量重複地傳送的缺點，作者的概念是當網路中節點要傳送資料給它的鄰居時，會先廣播一個稱為超資料 (meta-data) 的資料描述符號 (information descriptor) 給週遭節點，讓鄰居得以對此筆超資料判斷是否曾經接收過，如果未曾接收過則利用預先設計的規則通知傳送超

資料的節點將真正資料傳送給它們，反之則忽略此筆資料以節約發送端實際將資料傳送的電量耗損；另一個著名的方法，直接擴散(directed diffusion) [25]，它是以資料為中心 (data-centric) 的方法，它的特色是每一節點並不以位址識別碼來區別，而是以其所感測到的資料來定址，在此協定中，基地台首先送出一個查詢封包 (query)，將此查詢封包透過網路中其它節點擴散出去，當網路中其他節點擁有滿足此筆查詢的資料時，就會自動地將本身的資料傳回基地台。因為直接擴散協定傳送詢問及感測器回傳資料時皆是區域性地交換資訊 (即鄰近節點間互相交換資訊)，因此這個協定十分適合動態網路架構，且每一個感測器不需儲存太多路由資訊。另外一個適用於動態網路架構的研究，如 GeRaf (Geographic Random Forwarding) [26][27]，GeRaF 假設所有的感測器均知道本身及基地台的地理位置，當傳送資料前，感測器會先廣播自己的位置 (與基地台的位置) 給它的鄰居，如果有鄰居的位置比它還要靠近基地台，這些鄰居將會回應此感測器，如果感測器成功地收到其中一個鄰居的回應，它便將資料傳給這個鄰居，此鄰居再依同樣的方式將資料往前續傳至下一個距離基地台更近的感測器，其優點在於每個感測器只需要知道自已的位置，而此資訊可靠定位裝置得知，因此路徑不需於傳送前先尋找，得以適用於網路架構經常變動的應用。在網路層的設計中，我們亦可以加入睡眠機制的設計，但是如何妥善地安排睡眠期程不至於影響整體感測網路的運作卻是十分重要，因此睡眠排程不能影響感測網路的感測功能，亦即網路感測的涵蓋範圍 (sensing coverage) 不能被降低太多，否則會導致網路無法正常運作。[28]與[29]主要是針對叢集架構的感測網路所提出之睡眠排程演算法，此二方法皆將能量消耗列為首要考量，而感測涵蓋範圍則僅要求高於某個事先設訂的標準，例如達到百分之九十。[28]主要是利用感測器距離叢集領導者的距離來決定進入睡眠的機率，而

[29]則進一步地考慮如何來平衡每一個感測器所消耗的能量以便延長網路運作的生命週期 (life time)。

3. 應用層 (application layer) :

應用層來看,主要有兩個方向得加以思考,一為網路的「容錯程度(fault tolerance)」,對不同的應用,無線感測網路的容錯程度亦有所不同。[30]中,作者提出一個評估感測網路容錯效能的模式,可用來檢驗網路的容錯能力;[31][32]則提出針對資料融合的功能,針對感測器發生故障時,感測網路依然能夠保持正常的感測工作,並將誤差率在一定程度以內的資料送回基地台,作者們結合編碼理論的概念設計資料融合規則,並結合偵測理論 (detection theory) 設計感測器的感測功能,當有若干感測器故障時,具有容錯能力的資料融合規則依然可正確的將結果送回基地台。另一個方向則是指稱「網路安全 (network security)」,受限於感測器有限的記憶體及能源容量,大多傳統網路的保密或安全機制無法直接應用於感測器網路上,尤其是非對稱性的加密系統 (例如 RSA 等) 更加無法運用於感測器網路上。於 SPINS[33]中,作者們提出一個依賴基地台的安全架構,此架構要求每一個感測器與基地台分享一個秘密鑰匙,而感測器之間如果要建立一個安全通道,則需要透過基地台的幫助,這套機制的缺點在於必須將大量的資料傳輸至基地台進而消耗相當多的能量,因此此方法僅適用於基地台將資料安全地廣播給感測器的功能上。[34]中作者提出了隨機前置秘密鑰匙分配的概念 (random key pre-distribution),其方法是將感測器佈置於環境之前,事先將一定數量的秘密鑰匙內存於感測器中,網路開始運作時,感測器會利用本身的密鑰與鄰居建立安全通道,如果兩個相鄰的感測器沒有共同的秘密鑰匙時,他們會利用其他的感應器形成另一個安全路徑,再經由此路徑配對出一個新的雙方共用的秘密鑰

匙，進一步的改進於[35]中提出。基於隨機前置秘密鑰匙分配的概念，[36]大幅改進以上兩方法的安全機制，作者們利用並改進密碼學中 Blom 的鑰匙分配方法後，成功地將其運用於無線感測網路的安全機制上，同時也透過數學分析成功估計此方法的運算複雜度，進而證明安全機制在無線感測器上實現的可能性。由於前面所提之資料融合技術是無線感測網路上最關鍵的技術之一，因此資料融合的安全性便是十分重要的課題，[36]中作者利用證人（witness）機制確認融合後的資料並未遭到負責資料融合的節點竄改過，如此可保證基地台收到的資料是未經偽造的。



第參章、室內火警感測網路「PHOENIX」之服務功能

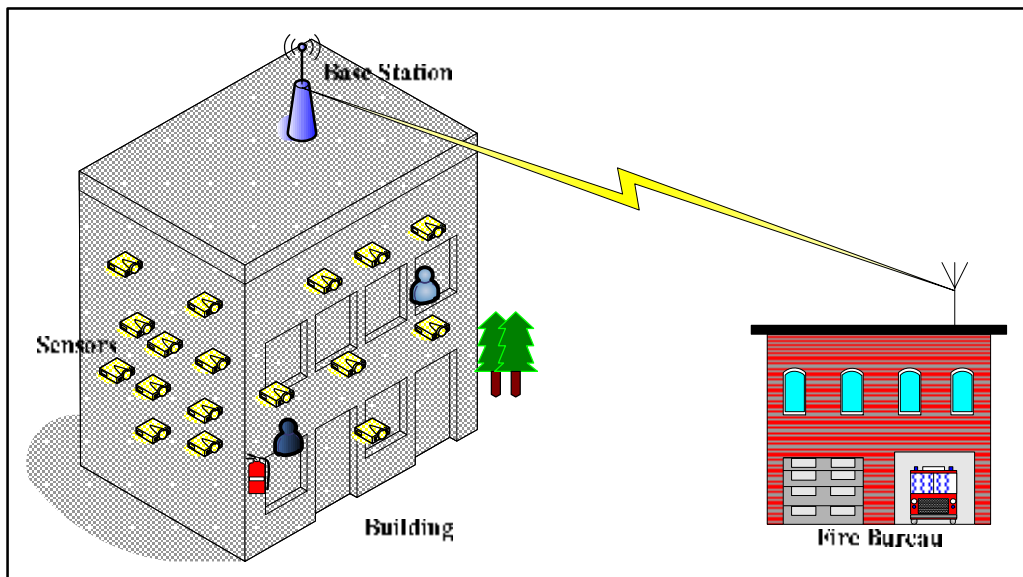
鳳凰亦稱為永生鳥、不死鳥，本系統「PHOENIX」的命名乃源自於古埃及的神話故事，據聞神鳥每間隔五百年會自焚一次，並在颯竄烈火及裊繞濃煙裡結束生命，再從死寂的灰燼當中洗煉獲得重生，變得比原來還要絢爛與茁壯，如此循環不已，故取其「浴火重生」之義。本系統設計之初的主要動機，是希望藉由現今無線感測網路這項前瞻性科技，發展出一套有助益於人類與社稷有實際貢獻的應用系統，針對火災情境，我們所設計的服務功能有：室溫偵測、災情回報以及逃生導引三大主要項目，接下來在本章的內文當中，將以此三大功能項目為主軸，分節詳述並探討其內部運作流程。



第一節、系統綜觀與環境佈署

在深入解說 PHOENIX 系統的三大主要功能項目之前，我們首先概述環境與系統的部署方式。應用的環境主要是設定於一般規模的建築結構體，此類型的建築物在結構上大多清晰整齊，並且依造完善的設計藍圖所建構完成；系統本身是由一部設置於建築物內控制中心的基地台（base station），搭配諸多部署於室內的微型感測器所組成的靜態無線感測網路架構，這些龐大數量的感測器以人工方式逐一被安置於室內所有需要重視的位置，各自在其感應範圍之內監測週遭的溫度指數，由於微型感測器本身礙於有限電量的考量，一般而言其傳輸距離僅介於數公尺至一百公尺之內，為了維持整體網路系統的連通性，系統假設感測器之間的距離不允許超過 25 公尺，系統管理者在擺設每個感測節點時，必須考慮其周圍均至少需要存在一個鄰近節點。

由於目前多數的建築物皆有所謂的安全逃生出口 (emergency exit)，因此在於合理的假設下，基地台端能夠依據建築藍圖掌握所有的安全出口處，同時當系統管理者設置妥定每一個感測器時，可以同時將這些感測器標記於藍圖上，如此便完成系統硬體方面的佈署程序，本系統，俟此步驟完成之後，系統管理者已經全然掌握網路節點與結構的搭配情形，必須將感測器以及逃生出口的位置資訊輸入至基地台內部資料庫中，以便於後續系統運作時，方能夠使充分發揮下面所提及之三項主要功能項目，下圖(四)是系統整個部署完成之後的想像情境圖例。



圖(四)、系統部署圖。

第二節、功能項目—「室溫偵測」之設計與研究

本系統對於室溫偵測的功能，是利用感測器上所提供的溫度感測單元負責偵測裝置本身的週遭環境，在網路連通的前提下，透過繞徑協定將室溫資料回報基地台端來完成。對於無線感測網路而言，由於數據通訊是消耗電源最嚴重的一項因素，因此在多數的實際範例中，為了有效達到節約

電源消耗的目地，感測裝置會設計以短距離的通訊方式來傳輸，一般而言感測器裝置電量消耗的目標被規範於-20dbm 左右[1]。感測網路中，降低電源的消耗主要可以從兩個方向來改進，分就軟、硬的平台的設計而言，在硬體設計的角度上，應該著重於低功率電波收發器的研發，致力發展出相同耗電卻可進行更遠傳輸功能的收發裝置，根據著名的摩爾定律²所述，硬體技術上的里程每隔一年半將成長一倍；另一個方面則是依系統軟體設計來著手，搭配效率更高、失誤率更低的通訊協定機制，避免網路壅塞或碰撞，讓整體效能得以提升。本研究的探討範圍是屬後者，我們從系統整體的觀點，利用網路拓樸的概念來省電，設計良好的拓樸結構能夠降低資料繞徑協定的運算複雜度，進而達成省電功能。

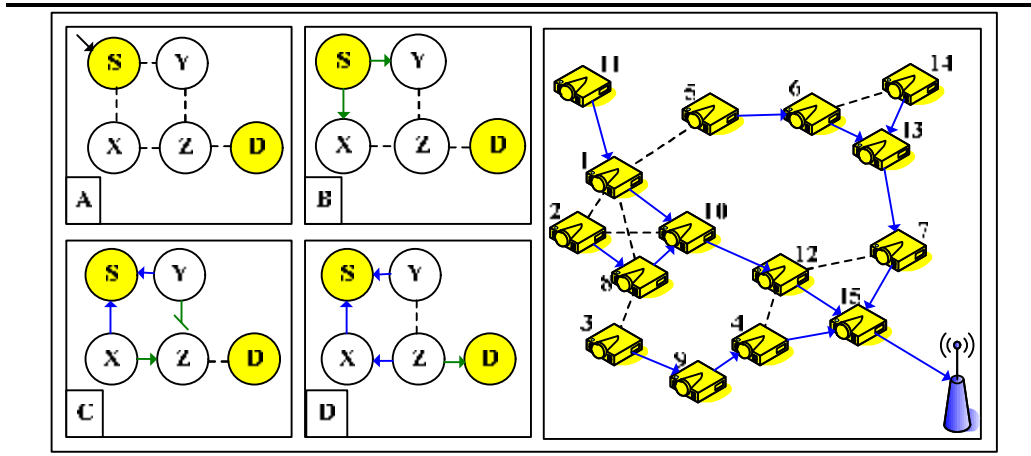
相較於無線隨意網路，無線感測網路有一項極為不同的特點，在隨意網路中，大部分資料的傳輸是屬於點對點的通訊模式，然而在感測網路裡，由於感測資料最終目地是監控中心裡面的基地台，因此是多對一的通訊模式，且多數的應用裡，系統會預設收集資料的基地台固定且唯一，換言之，這項特點讓我們可以容易地掌握感測網路中傳輸的目地端。觀察此項特點，我們設計以圖論當中的「樹狀拓樸 (tree topology)」作為系統通訊的基礎結構，根據圖論中對於樹的定義，每棵樹的組成可分為三種類型：根節點 (root node)、中間節點 (intermediate node) 與葉節點 (leaf node)，鄰近節點之間的關係就類似於親屬關係，除了根節點以外，網路上所有的節點均存在唯一的父節點做為通訊時的上行連結 (uplink)，對於沒有子節點的感測節點被稱為樹的葉節點，其餘的節點則皆屬中間節點，可想而知，普遍來說一棵樹上的葉子數量應該多與根節點，僅次於中間節點；就整體網路所形成的樹狀拓樸而言，倘若樹的成員節點裡包含了所有的網路元件，則稱此樹為一棵「展延樹」。

² 摩爾定律是由英特爾(Intel)名譽董事長摩爾經過長期觀察發現得之，是指 IC 上可容納的電晶體數目，約每隔 18 個月便會增加一倍，性能也將提升一倍。

建構網路中展延樹拓樸的方法，在此可以不失一般性地令基地台作為展延樹的根節點，透過根節點來啟動網路的初始化程序，基地台將發送系統初始化訊息 PHOENIXCMD_BEACON³，位於基地台週遭的感測節點將首先收到此訊息，並將本身的區域變數 *Parent* 設定為發送端的識別碼，俟此接收與設定程序完成後，隨即再依同樣的方式將此初始化訊息傳送其鄰近的其他節點，依此過程持續地重複操作，直到網路上所有的節點都收到資料且建立起與其父節點之間上行連結為止，其程序如下表(一)。

Procedure Network Initiation Process.

- 1: For Base Station :
- 2: Let *BS* be the base station of network.
- 3: **BS broadcast** *msg* =
(PHOENIXCMD_BEACON, *BS_{id}*, *my_{id}(BS)*);
- 4:
- 5: For all sensors *s_i* in the network **do**
- 6: Initially *Parent* ← *f*;
- 7: **If receive** *msg*(PHOENIXCMD_BEACON, *root_{id}*, *uplink*)
- 8: **If** *Parent* == *f*
- 9: *Root* ← *root_{id}*;
- 10: *Parent* ← *uplink*;
- 11: ***s_i* broadcast** *msg* =
(PHOENIXCMD_BEACON, *root_{id}*, *my_{id}(s_i)*);



表(一)、網路初始化程序。

³ PHOENIX 系統中所有使用的訊息型態請參照表(七)。

在網路上的每一個感測器會週期性地執行所謂的「鄰居偵測程序 (neighbor discovery)」，這項程序是感測器用來收集週遭節點資訊的一個過程，由欲偵測的節點廣播 PHOENIXCMD_DISCOVERY⁴ 訊息至其周圍的鄰居節點，受限於感測器本身的傳輸範圍，僅有那些相對於感測器的距離小於傳輸範圍之內的周圍節點能夠接收到此訊息，在其鄰近節點接收到此訊息之後，主動以 PHOENIXCMD_REPLY⁴ 訊息回覆，以便完成鄰居偵測的程序，如下表(二)。

Procedure	Neighbor Discovery Process.
1:	For every sender s_i :
2:	s_i broadcast $msg=(\text{PHOENIXCMD_DISCOVERY}, my_{id}(s_i))$;
3:	If receive $msg(\text{PHOENIXCMD_REPLY}, sender_{id}, neighbor_{id})$
4:	If $neighbor_{id}$ is not in s_i 's <i>NeighborList</i>
5:	Add $neighbor_{id}$ into s_i 's <i>NeighborList</i> ;
6:	
7:	For all proximal sensors s_j near s_i do
8:	Initially $NeighborList \leftarrow \emptyset$;
9:	If receive $msg(\text{PHOENIXCMD_DISCOVERY}, sender_{id})$
10:	If $sender_{id}$ is not in s_j 's <i>NeighborList</i>
11:	Add $sender_{id}$ into s_j 's <i>NeighborList</i> ;
12:	s_j unicast $msg=(\text{PHOENIXCMD_REPLY}, sender_{id}, my_{id}(s_j))$;

表(二)、鄰居偵測程序。

⁴ PHOENIX 系統中所有使用的訊息型態請參照表(七)。

我們已經介紹網路的初始化程序，倘若網路本身保持完整的連通性，這項程序必然使得每一個感測節點依據距離基地台的遠近關係，依序地被加入網路的展延樹狀拓樸當中，這個初始過後所形成的展延樹狀拓樸稱為「主要展延樹 (primary spanning tree)」。每當感測器依照程序設定 *Parent* 及 *Root* 變數後，它將立即以單一傳播 (unicast) 的通訊方式回報其本身的基本狀態，包含識別碼、電量等資料，同時也會一併回傳其周圍的鄰近資訊 (proximity information)，而鄰近資訊則是透過上述的鄰居偵測 (neighbor discovery) 程序獲得。

經由上述的初始過程之後，所有的訊息流向便可據此主要展延樹為基礎，進而達成系統訊息流通的功能。感測器可將尋常室溫或特定回應資料以己身作為起始的發送端點，依循主要展延樹的樹枝向上回朔至基地台端，根據圖論中的定義而言，樹狀結構本身即是一幅不存在迴路的連接圖 (acyclic connected graph)，這使得我們可以利用這項優點，輕易地設計出具有無迴路 (loop-free) 性質的繞徑回報制度，透過上述的初始過程，每個感測節點會記錄其上行連結的節點識別碼，此方式是讓感應器僅需要沿著主要展延樹的樹枝路徑，將資料單一傳播給其父節點，層層接續傳遞便可完成整個回報的程序。反之，若當基地台端有關於查詢或命令的訊息欲發送至下屬特定的感測器時，可以依源頭繞徑的概念指定路徑，同樣地依照樹枝連接的情形，將命令下達至特定感測器端，這些訊息在主要展延樹之間的傳遞皆屬於單一傳播的通訊方式。

第三節、功能項目—「災情回報」之設計與研究

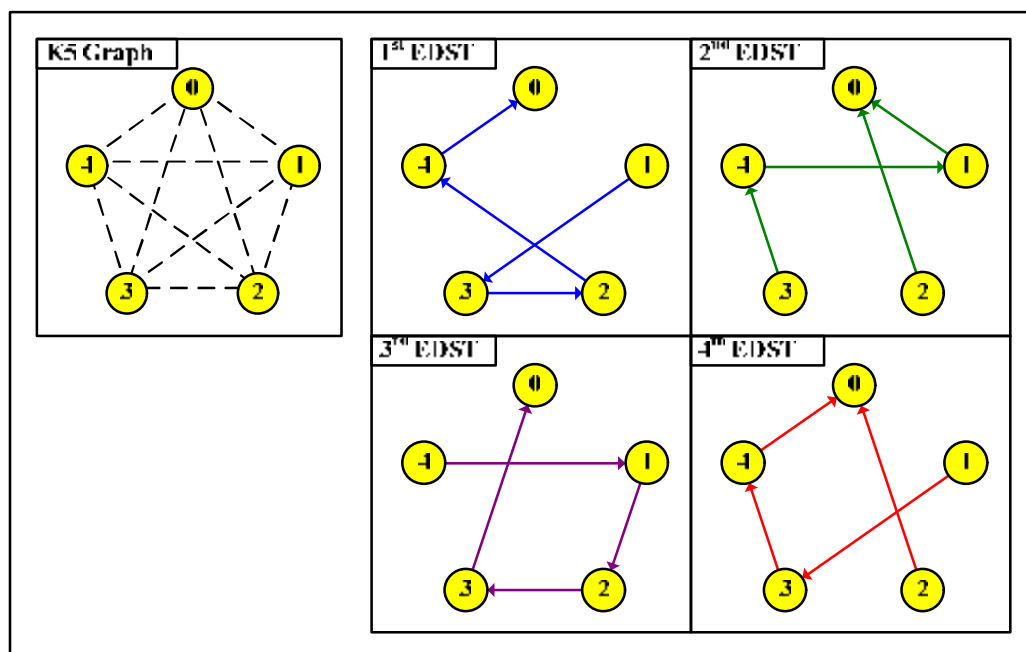
依主要展延樹拓樸作為 PHOENIX 火警系統通訊時的基礎結構，不僅能夠使資料回報更加有效率，並且透過這樣的設計亦能有效地降低繞徑協定的運算複雜度，進而達成節省耗電，藉此以延長整體網路的運作時

程。然而，在真實的世界中永遠充滿著許多不確定的因素，這些環境外在的變因導致實際狀況經常發生改變，以火警災難的實際情境舉例說明，感應器隨時存在著遭受到瞬間火勢的高溫而燒損的機率，加上感測器本身亦有零件故障的可能性，諸如此類的種種因素都將使得主要展延樹拓樸發生斷裂的現象，以至於網路失去連通性。尤其是這些特殊的應用中，系統針對於網路本身的存活程度有著非常嚴格的要求，在此所謂的網路「存活程度 (survivability)」是指當系統面臨意外情況發生時，本身尚且可以完成主要的任務之能力，意即系統對於這些意外情形的抵抗程度。以 PHOENIX 火警系統來說，正當部署於建築物內部的無線感測網路遭逢祝融侵擾時，瞬時高漲的火勢使得部分的感測器失靈，下面我們繼續深入探究本系統該如何面臨在這樣的情形下，依然能夠完成其回報災情的任務。

火災隨時可能發生於建築物內部的任何位置，就主要展延樹拓樸來看，由於網路上所有的節點與基地台之間均僅存在一條可行路徑，假設其中任何一個非葉子的節點出現故障現象時，必然使得這樣的拓樸發生分裂、不連通的情況，此時不僅該節點無法順利作業，亦會一併導致其下屬所有分枝與基地台失聯，遂使整個系統效能受到影響，嚴重影響災情回報功能。針對於此，我們將提出以「交錯展延樹 (Edge-disjoint spanning tree)」的概念改善這樣的情形，增加網路拓樸的存活程度，亦即其網路的連通程度 (connectivity)。

所謂「交錯展延樹」的概念是利用數棵展延樹並存於網路中，用以提升網路的存活程度，系統於平常時候預設以主要展延樹為繞徑的基礎，一旦遭遇危急情形時，其他備份的展延樹 (backup spanning tree) 則會發揮作用，這種利用重複資源以增進系統對於災害抵抗力的論述，曾經在許多容錯式計算的相關研究報告中被指出，主要是利用備份的數棵交錯展延樹來彌補一棵主要展延樹對於系統存活程度的缺陷，但是倘若數棵交錯展延樹彼此之間存在共用的樹枝，我們發現這些共用的連結將限制了系統中提

升存活程度的功效，換言之，我們希望建構出儘量交錯的數棵展延樹，意即它們所共用的連結應該越少越好，在下圖(五)中是 5 個節點所形成的完整圖 (complete graph) 以及其數棵交錯展延樹。



圖(五)、完整圖 (K5) 之四棵交錯展延樹範例。

本篇研究論文中，我們提出一套建構交錯展延樹的演算法，為了便於討論，在此先將整個網路轉換成圖論的數學模型來輔助思考。假設系統存在有 n 個網路節點，這些網路節點將隨機地分佈於二維的歐基里德幾何平面 Ω 上，仝鄰的節點之間能夠利用無線通訊來連結，如果 a 點能夠與 b 點之間進行通訊，則表示 b 點亦可與 a 點進行通訊，這個網路可以用 $G=(V, E)$ 來表達這張圖形，其中 $V=\{v_1, v_2, \dots, v_n\}$ 是由所有感測器節點所形成的有限集合， $|V|=n$ ，無向的邊 (undirected edge) 來表示感應器之間的通訊鏈結，而由所有的無向邊構成的有限集合則為 $E=\{e_1, e_2, \dots\}$ 。

在網路初始過後，感測器網路會建立起主要的展延樹拓撲通訊結構，並利用它來回報本身狀態與鄰近節點的相關資訊，經歷這個步驟之後，可

以使基地台能夠掌握整個網路所構成的連通圖，即 $G=(V, E)$ 。我們所提出的演算法必須能夠依據任意一幅連通圖形，在有效率的時間複雜度內依序計算出 k 棵展延樹，需要注意的是，此演算法將於基地台端進行運算，而計算過後的結果，會依循主要展延樹樹枝路徑以反向的單一傳播方式發送至每一個感測節點。

交錯展延樹的演算法計算過程是採用疊代 (iterative) 的形式，循序完成第 1、2、...、至 k 棵交錯展延樹，其中在第 i 棵展延樹的計算過程中，必須參考先前第 1 至 $i-1$ 棵建立的歷史記錄，最後將所有交錯展延樹的計算結果，存至集合 Ψ 當中，演算法整體程序如下表(三)。

<i>Algorithm</i>	Ψ EDSTs (G, k); Ψ , return the k EDSTs
1:	$\Psi \leftarrow f$;
2:	for ($i=1 \dots k$)
3:	$\Psi \leftarrow \Psi \cup \text{EDST}(G, \Psi)$;
4:	return Ψ ;
<i>Comment</i>	
Ψ	The set of k EDSTs.

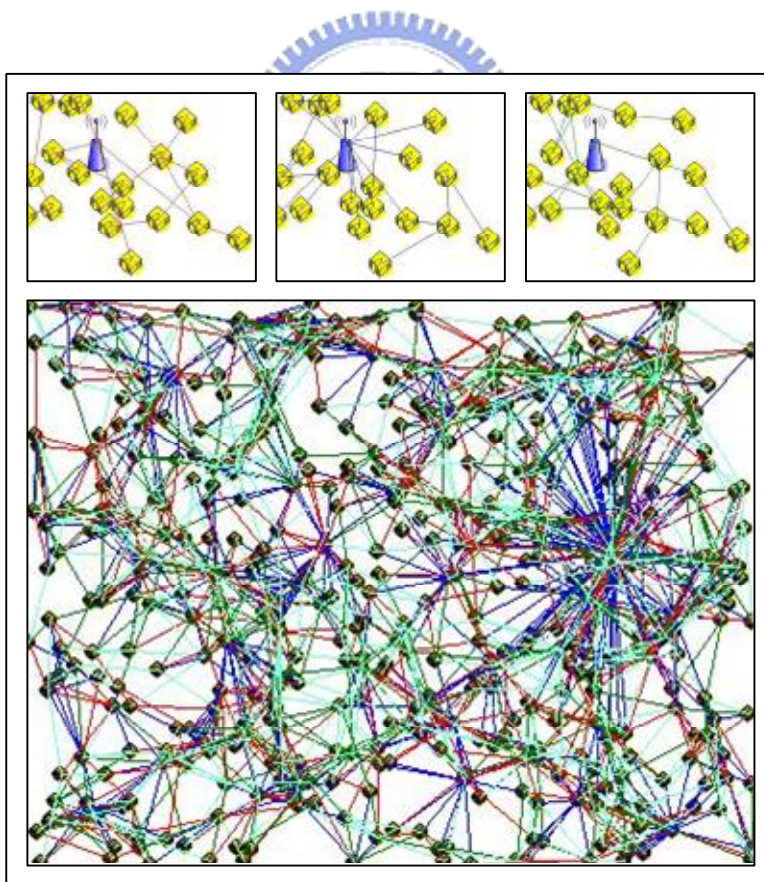
表(三)、交錯展延樹之演算法 (A)。

接著我們針對建立第 i 棵交錯展延樹來說明，第 i 棵樹的建立是以第 0 節點作為樹的根節點，再從與它連接的邊當中，挑選出最合適的其中之一加入第 i 棵展延樹，重複地挑選直至整個程序總共執行了 $n-1$ 次，這代表已經收集了所有的節點並完成展延樹的建構動作。在程序執行的過程中，圖形中的邊線可分為三種類型：treed edge (樹枝)、sprout edge (枝芽) 與 free edge (自由邊)；treed edge 是指那些在挑選過程中已經被納入為展延樹組成元素的邊，sprout edge 是指那些經由篩選過後可能成為 treed edge 的候選邊，而 free edge 則是指仍屬於 $E(G)$ ，但不屬於展延樹的組成邊，在每一輪的挑選過程中，free edges 所形成的組合將包含著 sprout edges。

所謂「挑選」的方式，則必須依據圖形的連結狀況（意即 $E(G)$ ），及其之前所有 $i-1$ 棵展延樹的建構歷程紀錄來決定。舉例說明，兩個邊 e_1 與 e_2 ， e_1 在歷史紀錄中未被挑選過，則我們應該挑選 e_1 為這一階段的 sprout edge，透過這樣的挑選機制可以篩選出在先前的樹尚未使用過的邊，也就是與先前所建立的樹互相交錯的邊，在此階段的執行過程當中，合適的 sprout edges 或許不只有一條，而每一條 sprout edge 皆有兩個端點，根據上述的定義，其中一端屬於目前所建立的第 i 棵展延樹，另外一端屬於原本圖形的 $E(G)$ ，對於這些 sprout edge 而言，靠近樹的端點必定與之連接的 free edges，我們將該端點的 free edge 數量稱之為「free degree」，此時則可依據較大 free degree 為進階的篩選條件，進一步地進行挑選，這個步驟的設計是為了避免第 i 棵展延樹的建立過程，對後續第 $i+1$ 、...、 k 棵展延樹的挑選程序造成過大的影響。倘若經由上述的篩選條件抉擇出來的 sprout edges 數量仍然大於一，則任意選擇之，最後選擇出來的結果邊，即為此階段中最合適的邊；其演算法如下頁表(四)所示，執行示範結果於下頁圖(六)。

<i>Algorithm</i>	T EDST (G); T, return the requested EDST
1:	$T \leftarrow \{v_0 \mid v_0 \in V\};$
2:	// Initially, we pick the vertex v_0 to be the root of i -th EDST.
3:	while ($ V(T) < V $) /* Termination condition. */
4:	$x \leftarrow$ sprout edges with the minimal edge utilization ;
5:	$Y \leftarrow \{v \mid v \in V(x \cap T)\}$ with $h(v)$ is maximal. ;
6:	$T \leftarrow T \cup \{\text{random pick one edges in } x \text{ which ends in } Y\};$
7:	return T;
<i>Comment</i>	
T	The i -th EDST structure.
x	The set of sprout edges.
Y	Treed vertex with maximal free degree.
$h(v)$	Free degree of vertex v .

表(四)、交錯展延樹之演算法 (B)。

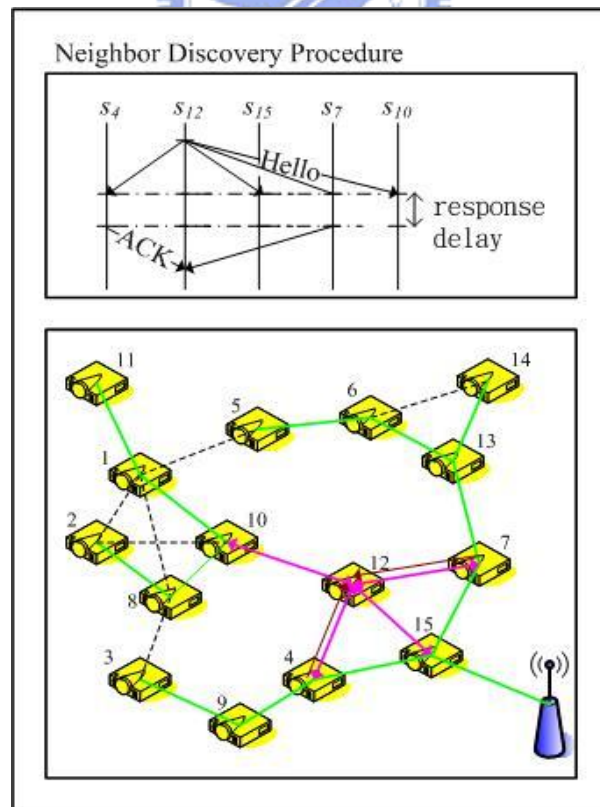


圖(六)、交錯展延樹演算法之執行結果。

演算法的時間複雜度為 $O(k \times n^3)$ ，從矩陣中挑選出適合的 sprout edge 需要花費 $O(n^2)$ ，建立一棵展延樹需要挑選 $n-1$ 次的最適合之 sprout edge，而此步驟則花費 $O(n-1)$ 的時間複雜度，整個演算法共執行 k 次的 EDST 程序，因此需要 $O(k \times (n-1) \times n^2) = O(k \times n^3)$ 的時間複雜度。

交錯展延樹確實能有效地提升網路拓樸的連通性，避免單獨一棵展延樹易於造成斷裂的缺點。接著我們續談如何利用這種特殊的拓樸來完成災情回報的機制，在此首先必須談及感測器判斷火災可能發生的條件。

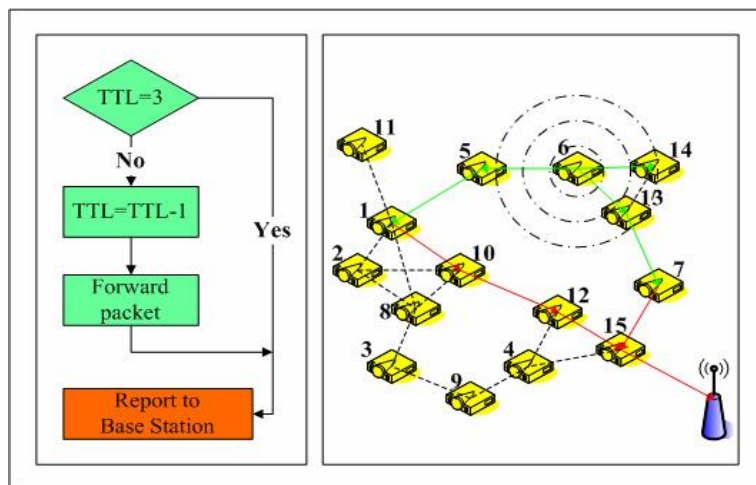
建構於數棵交錯展延樹的拓樸上，本系統的回報方式預設是採用合作方式完成，系統中所有的感測器均會週期性地執行鄰居偵測功能，如下圖(七)，以確保網路連通性，在此系統設計的週期間隔是以秒為單位，因此當鄰居在數秒延遲時間內尚未回覆時，加上感測器本身已偵測到週遭溫度高於危險臨界溫度，隨即判斷目前周圍已經失火了，立即啟動回報機制。



圖(七)、鄰居偵測程序之圖解範例。

當火災發生時，預設的反應路徑是基於主要展延樹拓樸，在順利情狀下，反應訊息可透過其上行節點，以類似室溫回報的流程完成傳報動作，但由於其上行節點亦可能為遭受燒損的受災區之一，使得預設路徑失效，此時我們利用多棵交錯展延樹所形成的拓樸，感測器可以選擇距離基地台最近的上行節點進行回報，與基地台的遠近關係決定於先前於基地台端計算過後的結果，此結果在運算完畢後，將由基地台端以源頭繞徑（source routing）方式逐一發佈，因此每個感測器在無意外的情形下，便能夠掌握其與基地台之間的跳躍數（hop count）。

利用交錯展延樹拓樸所設計的反應機制，概念上是利用在多棵展延樹之間選擇一較適路徑所完成，然而這樣的反應機制並不能夠保證火災緊急訊息能夠確實抵達基地台完成任務，如下圖(八)，因此我們輔以一個簡單的機制來改進我們的回報措施，稱之為「區域性廣播（local broadcast）」，區域性廣播是指透過封包的存活時間（time-to-live，TTL）來限制回報封包散佈至整個網路，假設 TTL 設定為 3，表示災情回報封包由被發送端開始計算，最多僅經過三次的跳躍傳達給較多的鄰近感測器，並請求協助共同向基地台回報，透過這樣的補強措施，可大幅提高災情確實反應的機率。



圖(八)、區域性廣播之圖解範例。

當位於周圍的感測器接獲由災區感測器所發出之回報訊息，在我們的设计當中，我們令位於主要展延樹上屬於災區感測器之子節點轉換至次要的展延樹來傳遞，至於其他非屬災區感測器子節點之感測器則依舊以主要展延樹上所記載的父節點當作是上傳目標，如此的方式來完成我們在本節當中所要求之回報功能。

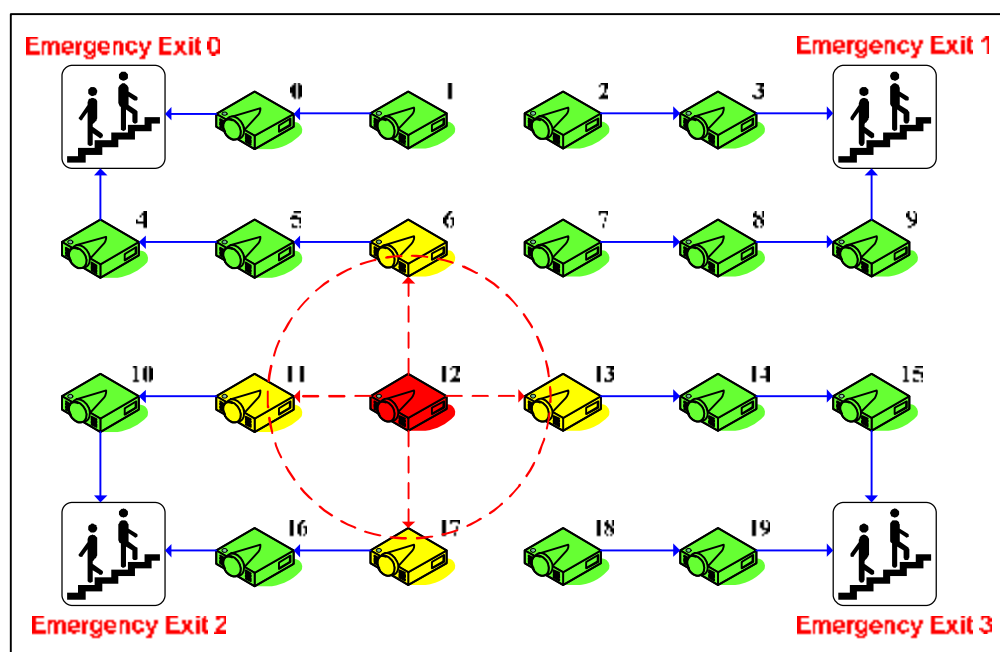
第四節、功能項目—「逃生導引」之設計與研究

繼災情回報功能項目之後，使得系統能夠於第一時間內掌握有關火災事故的實際情形，此時系統將更進一步地提供「逃生導引」的服務功能，希望能透過這項進階的服務功能，讓受災人員能夠迅速地根據感測器裝置導引最正確的逃生方向，儘速脫離火場，將人員傷害減至最低。

假設基地台能夠參照藍圖資訊，掌握整個網路部署於建築物內部的實際情況。透過位於出口處的感測節點啟動逃生導引功能，企圖通知網路上的其他節點利用本身的顯示輸出設備，顯示出目前距離最近的逃生出口處，提供人員最適合的逃生方向；在上一節中，我們曾報告過關於 PHOENIX 系統反應災情的措施，當基地台端在上述的回報措施完成後，將能夠確切地掌握火災區域，此時基地台將即刻依源頭繞徑的路由方式發派緊急命令通知那些位於出口處的感測器裝置，促使這些出口處節點啟動此節中的導引功能。

這個方式是以各個逃生出口為根節點，動態計算出相對於這些逃生出口的逃生展延樹，令基地台掌握建築物中有 n 個緊急出口，正常情形而言，網路中所有連通的節點均可接收到 n 次泛洪機制的訊息，在此泛洪訊息中將包含有由逃生出口起算至任何接收端之跳躍數 (hop count)，因此感測器能夠選擇跳躍數最小的出口處，導引人員逃生，此處選擇最小跳躍數的原因在於表示該逃生路線能夠通往距離最近的出口處。

<i>Procedure</i>	Survival Guiding Process.
1:	For every exit :
2:	Let E be a survival exit.
3:	$SurvivalHopCount \leftarrow 0$;
4:	E Broadcast $msg=(E_{id}, my_{id}(E), SurvivalHopCount)$
5:	
6:	For all sensors s_i in the network do
7:	Initially $SurvivalHopCount_i \leftarrow \infty$;
8:	Initially $SurvivalParent \leftarrow f$;
9:	If $received(e, k, hops)$
10:	If $SurvivalHopCount_i + 1 < hops$
11:	$SurvivalHopCount_i \leftarrow hops + 1$;
12:	$SurvivalParent \leftarrow k$;
13:	s_i Broadcast $msg=(E_{id}, my_{id}(s_i), SurvivalHopCount_i)$;



表(五)、逃生導引程序之示範圖例。

例如上表(五)中的附圖說明，假設第 12 號感應器判斷出有火災情況發生，則該感應器將自己設定為紅色來標示目前此處為危險區域，並廣播通知在其周圍其他感應器，俟週遭感測節點接收到此緊急訊息 PHOENIXEMG_WARNING³ 時，則將自己設定為黃色用以代表警戒區域，

此時系統處於火場正在延燒的時候，藉由上一節的回報機制，緊急訊息將被設法回覆至基地台處，然而當基地台處接獲緊急訊息時，立即以源頭繞徑的方式下達導引命令給四個出口處的感應器，要求四個感應器隨即廣播導引訊號給網路上的所有節點，以發揮導引的功能。

第五節、研究結語

在本章當中，我們以系統的三大主要功能項目為主軸，深入探討其細部的運作流程，在第二章當中，我們提及如何建構主要展延樹以作為室溫資料回傳路徑，完成室溫偵測的功能，接下來，在第三章的內容中，我們呈現出針對單棵展延樹的缺點，所設計的交錯展延樹之建構演算法，並利用多棵展延樹的概念提升網路抵抗意外災難的存活程度，以及利用交錯展延樹達到緊急災難回報的機制，第四節中則是本系統為求更進一步地減低火災對於人員的傷害，而設計的逃生導引功能。透過這三大功能要項，確實改良了傳統的建築火警系統，在下一章當中，我們將延續重點，深入介紹系統實作的過程與操作程序。

第肆章、室內火警感測網路「PHOENIX」之系統實作

根據內政部消防署的統計資料顯示，於民國九十三年元月份迄今，全台灣總共發生三千多起消防事故⁵，而其中建築起火的相關項目共占所有的34%，是所有火警事故中所占比例最大的一部份，換言之，每天至少平均有七次以上的火災事件是發生於建築物內的。然而這些不幸事件歸咎起原因，多半是由人為疏忽所引起的，倘若建築物本身能夠具備適當的防災應變系統，將這些意外發生的機率減至最低，勢必能有效地避免災後對於人員、財物所造成的嚴重傷害。因此現代化的建築規劃在設計之初，除了考量空間格局的美觀與結構穩定性之外，更應該積極思考有關災前預防的安全性議題。

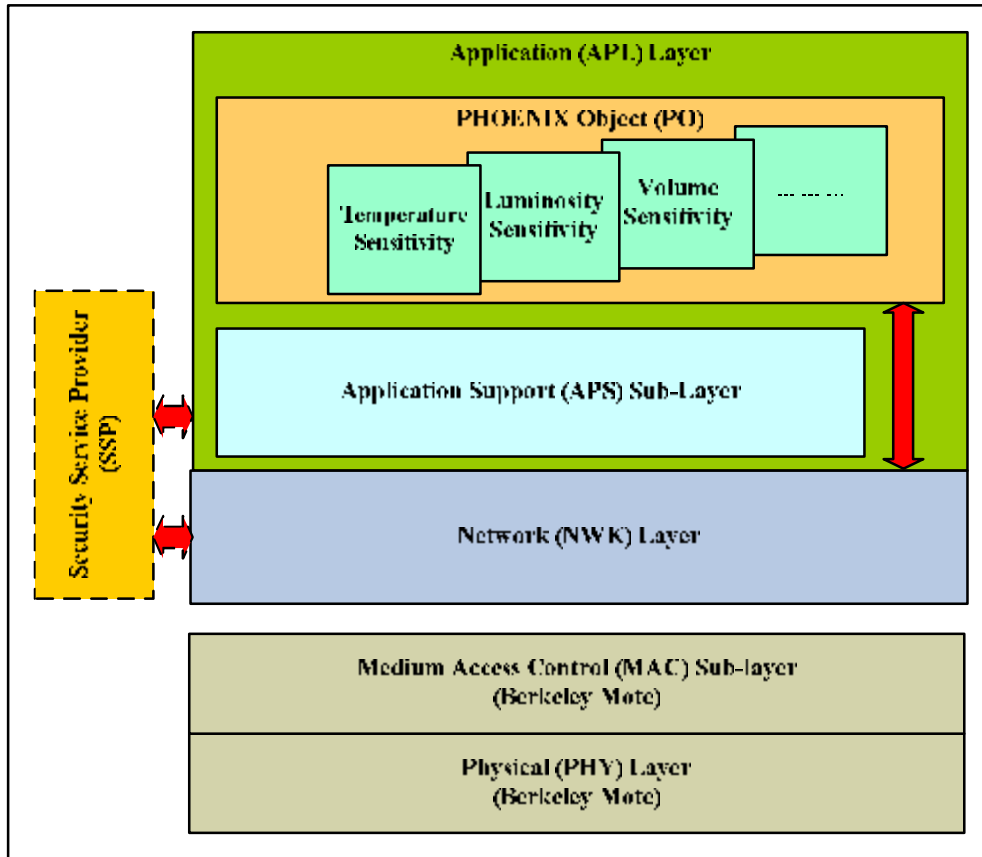
多數的建築物會在室內隔間或走廊通道等處佈置煙霧溫度探測器，用以偵測判斷室內是否可能發生火警事故，然而目前這些探測器的設置是利用鋪埋於牆壁內的實體線路直接與監控中心（管理室）連接，彼此之間乃獨立運作並無互相協調、配合的功能，這使得系統在實際防災的應用上打了折扣，線路本身亦造成維護上的困難，基於種種因素，在本篇研究論文中，我們利用前瞻性的無線感測網路技術為基礎，設計出一套針對於建築物內適用的智慧型防火救援系統，這套系統在佈置上相當地簡易，微型的感測裝置能夠無礙於室內美觀，透過本身具備的通訊功能，進而可自我組態形成一個防火互聯的網路系統。目前主要提供的三大服務項目有室溫偵測、災情回報與逃生導引等，其背後設計理念已於上一章當中詳述，本章將進一步地探討如何實作本系統，我們利用美國柏克萊大學所研發的微塵感測器硬體平台（Mote-MECAII），同時輔以微型作業系統（TinyOS）的嵌入式控制技術，內容著重於系統的架構配置、協定流程與效益評估。

⁵ 請參照內政部消防署全球資訊網，網址：<http://www.nfa.gov.tw/>。

第一節、通訊協定堆疊與訊息封裝

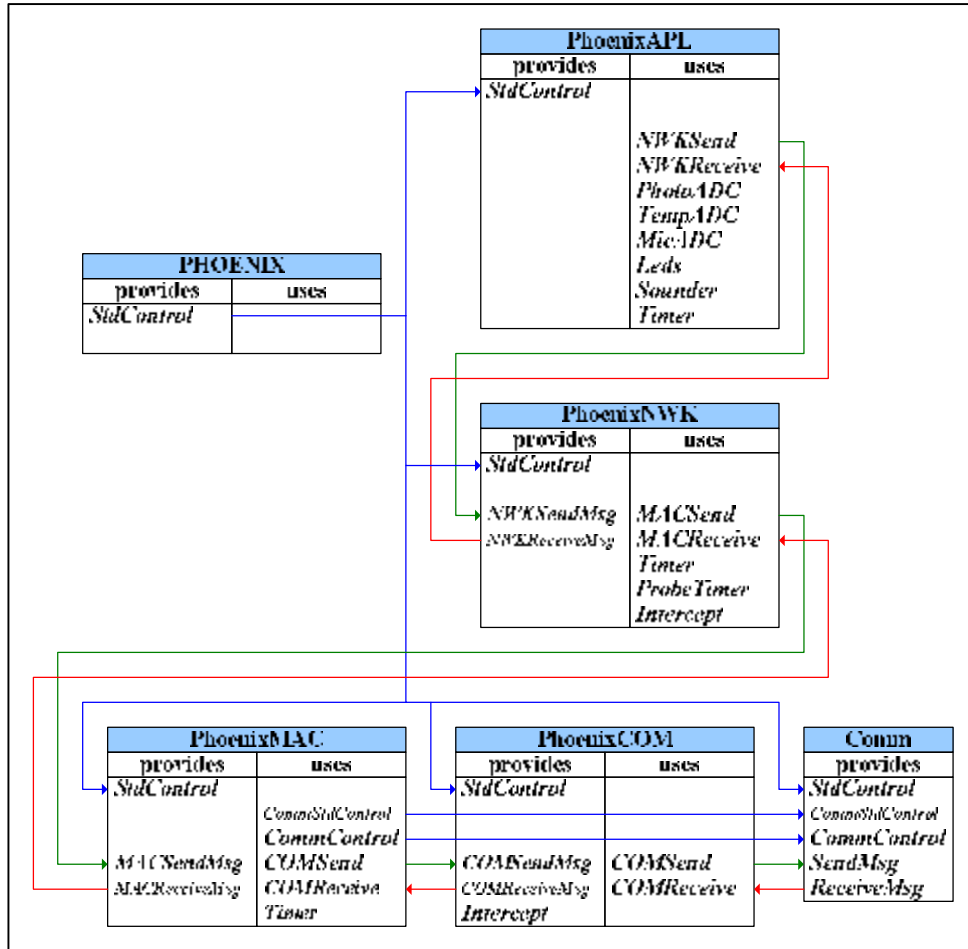
PHOENIX 火警系統是透過基地台以及部署於建築物內部的諸多感測器硬體所共同構築而成的無線感測網路，基地台與感測器均必須遵守共通的無線網路協定 (protocol)，才能彼此交換控制訊息與資料封包。一般而言，在網路協定的描述當中，首先談到的就是「分層 (layer)」概念，這是指網路運作時依照功能項目，將網路諸多程序規劃成不同的模組區塊，每個模組之間互相搭配與支援，進而堆砌成龐大且細膩的系統架構，即所謂「協定堆疊 (protocol stack)」，這項概念是現代化系統設計的一個重要技巧，例如在網際網路上早已盛行許久的 OSI 七層式網路架構就是一個典型的示範，然而礙於無線感測網路對於省電、計算及有限資源等限制考量，我們必須針對本系統量身打造一套合適的協定推疊，下面進而闡述本系統中所設計的推疊架構。

整體來說，PHOENIX 系統的網路協定堆疊可分為四層，如下頁圖(九)中所示，包含了實體層 (Physical Layer)、媒介存取控制層 (Medium Access Control Layer)、網路層 (Network Layer) 以及應用層 (Application Layer) 等四個層面；其中實體層與媒介存取控制層分別負責底層的載波傳輸與媒介控制等基礎通訊功能，搭配專門定義網路訊息交換行為的網路層以及制定系統服務流程的應用層。PHOENIX 系統屬於一種任務導向 (mission-centric) 的應用型架構，在規劃上，我們將應用層再細分為兩項次層次，即物件層 (PHOENIX Object) 與應用支援層 (Application Support Sub-Layer)，裡面描述了所有火警應用時需要參照的元件模組，以及針對物件層所提供的特殊輔助功能模組。



圖(九)、PHOENIX 系統之通訊協定堆疊。

在微型作業系統 TinyOS 上實作所設計的協定堆疊，必須使用一套稱為「NesC」的程式語言，這套程式語言本身是由柏克萊大學所負責維護，具備有類似 C 語言的文法結構，支援客戶端撰寫與編譯程式，下圖(十)為我們所實作的協定堆疊模組配置圖，配合上頁圖(九)中所設計的堆疊架構，可分為四大項模組，由上至下分別為 PhoenixAPL、PhoenixNWK、PhoenixMAC 及 PhoenixCOM，各自對應於上述的四個協定堆疊層次，在圖(十)中亦表現出資料與控制訊號的聯接關係。



圖(十)、協定堆疊模組之連接關係圖。

搭配著協定堆疊的佈局，接著再介紹 PHOENIX 系統中的「訊息封裝 (message encapsulation)」之前，首先得談及在 TinyOS 中所制定的訊息結構 *TOS_Msg*，如下表(六)所示，*TOS_Msg* 訊息結構的大小僅為具有十三個位元組大小，包含一般通訊傳輸時的基本資訊，諸如：*addr* (目的地位址資訊)、*type* (訊息型態)、*length* (訊息長度)、*crc* (錯誤校驗碼) 等等資訊，特別值得一提的是其中一項用來存放 *data* (資料) 的欄位，這個欄位是一個整數型態的指標，指向另外一處長度為 *TOSH_DATA_LENGTH*⁶的

⁶ 依據 TinyOS 中 AM.h 的定義：

```
#define TOSH_DATA_LENGTH 29
```

陣列位置，在 PHOENIX 系統中的所有訊息將利用這個 data 的部份加以封裝，封裝的技巧在於明確地區分出資料所屬的堆疊層次。

Declaration	Structure of TOS_Msg in TineOS Source Code
1:	typedef struct TOS_Msg {
2:	uint16_t addr;
3:	uint8_t type;
4:	uint8_t group;
5:	uint8_t length;
6:	int8_t data[TOSH_DATA_LENGTH];
7:	uint16_t crc;
8:	uint16_t strength;
9:	uint8_t ack;
10:	uint16_t time;
11:	} TOS_Msg;

表(六)、訊息封裝格式。

PHOENIX 系統中訊息封裝主要分為 type、srcaddr、destaddr、hopcnt、seqno 以及 data 等，型別依據所有堆疊層次共分 31 種，如下表(七)所示，其中以標示 CMD、SET、ASK、RPT、EMG 分別表示為命令、設定、詢問、回報與緊急等形態之訊息，主要是以 type 來作為訊息封裝與解封裝時的判斷；srcaddr、destaddr 是紀錄發送來源端的識別碼；每當訊息流路經過一次地跳躍轉傳，hopcnt 欄位會被加一，設計於傳輸時確保收方能無誤地接收資料；seqno 是記載該訊息相對於 srcaddr 節點之發出紀錄，用以避免 destaddr 節點重複接收相同的訊息，也助於了解網路訊息失落的情形。

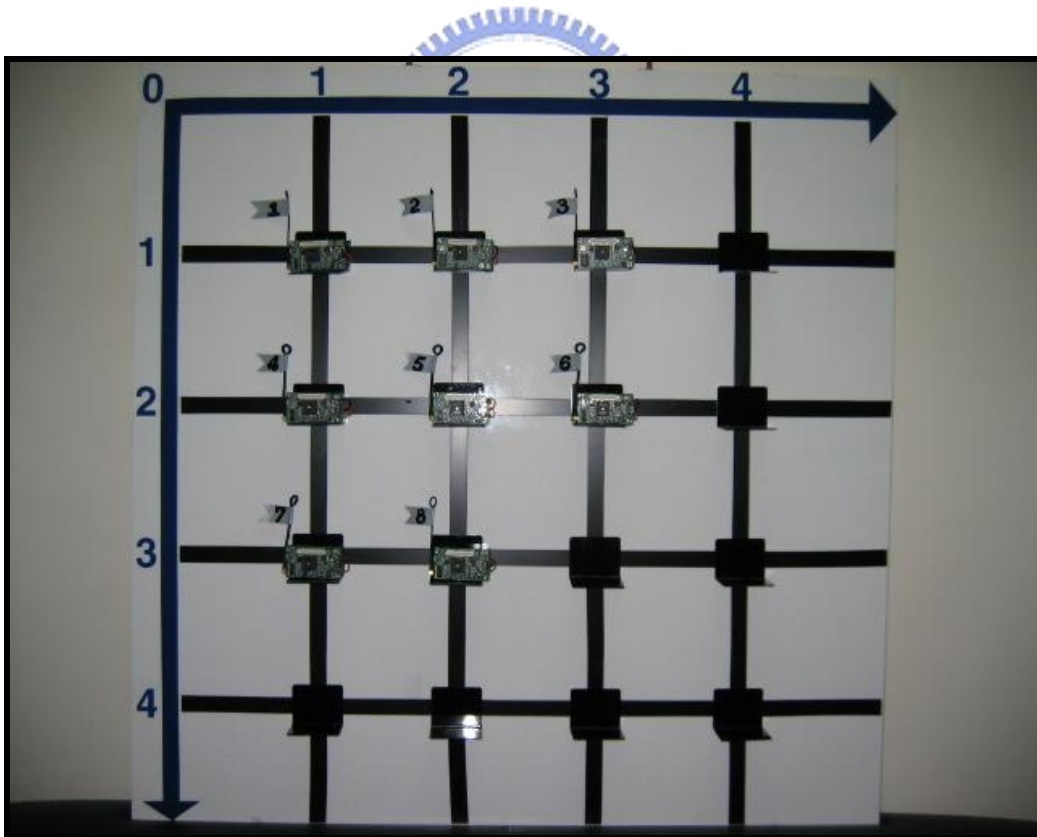
Message Type	Enum.	Comment
Broadcast Message Form Base Station To Sensor		
PHOENIXCMD_RESET_ALL	0	Reset PHOENIX WSN.
PHOENIXCMD_BEACON	1	Initiate PHOENIX WSN.
PHOENIXCMD_TIMESYNC	2	Time synchronize PHOENIX WSN.
PHOENIXCMD_IDLE_ALL	3	Idle PHOENIX WSN.
PHOENIXCMD_ACTIVE_ALL	4	Active PHOENIX WSN.
Unicast Message Form Base Station To Sensor		
PHOENIXCMD_RESET	5	Reset the specified sensor.
PHOENIXCMD_IDLE	6	Idle the specified sensor.
PHOENIXCMD_ACTIVE	7	Active the specified sensor.
PHOENIXSET_PARAMS	8	Set parameters to the specified sensor.
PHOENIXSET_POSITION	9	Set position to the specified sensor.
PHOENIXSET_DATE	10	Set date to the specified sensor.
PHOENIXSET_TIME	11	Set time to the specified sensor.
PHOENIXSET_NBR_DIR	12	Set neighbor direction information to the specified sensor.
PHOENIXSET_EDST	13	Set edge-disjoint spanning trees to the specified sensor.
PHOENIXASK_PARAMS	14	Ask parameters to the specified sensor.
PHOENIXASK_STATUS	15	Ask status to the specified sensor.(including date, time, position, ...etc)
PHOENIXASK_READING	16	Ask reading data to the specified sensor.
PHOENIXASK_NBR_DIR	17	Ask neighbor direction information to the specified sensor.
PHOENIXASK_EDST	18	Ask edge-disjoint spanning trees to the specified sensor.
PHOENIXEMG_GUIDE	19	Initiate the emergency guiding procedure.
Unicast Messages Form Sensor To Base Station		
PHOENIXRPT_PARAMS	20	Report parameters to the specified sensor.
PHOENIXRPT_STATUS	21	Report status to the specified sensor.(including date, time, position, ...etc)
PHOENIXRPT_READING	22	Report reading data to the specified sensor.
PHOENIXRPT_NBR	23	Report neighbor direction information to the specified sensor.
PHOENIXRPT_EDST	24	Report edge-disjoint spanning trees to the specified sensor.
PHOENIXRPT_WARNING	25	Report warning sensing data to the specified sensor.
PHOENIXRPT_DANGER	26	Report danger sensing data to the specified

		sensor.
Broadcast Messages Between Sensors		
PHOENIXEMG_WARMING	27	Inform the warning information to neighbors.
PHOENIXEMG_POST	28	Guiding post the neighbors.
PHOENIXCMD_DISCOVERY	29	Neighbor discovery.
PHOENIXCMD_REPLY	30	Reply the PHONIXCMD_DISCOVERY.

表(七)、訊息列表總覽。

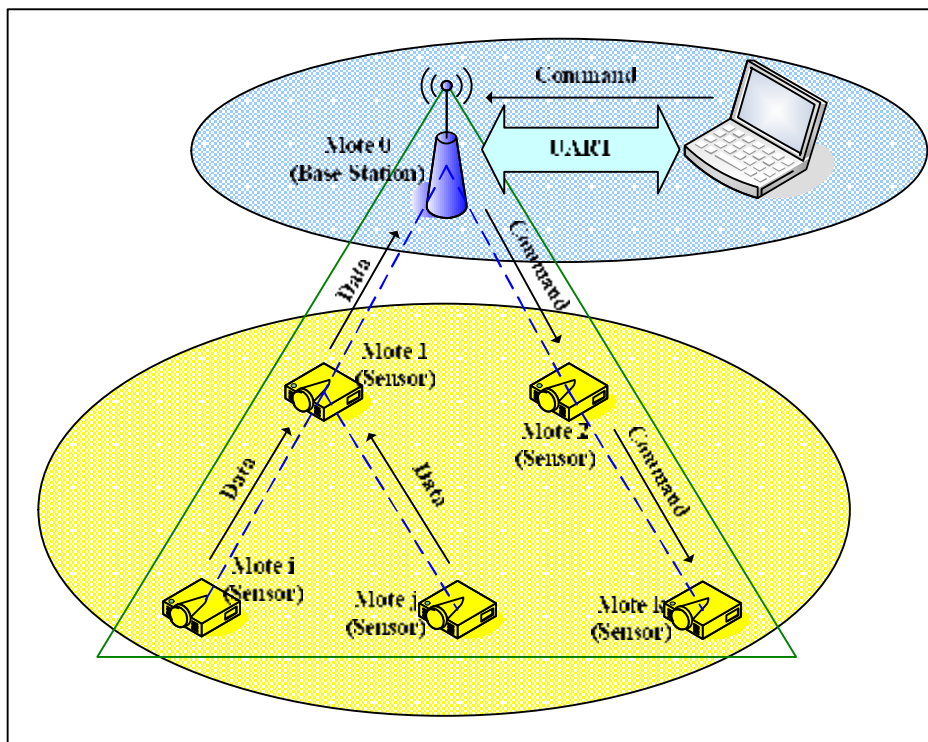
第二節、系統架構與運作流程

整個系統的佈置方式誠如第三章所描述，下圖(十一)是整個系統佈置完成時的實際情境圖。



圖(十一)、系統實際示範圖。

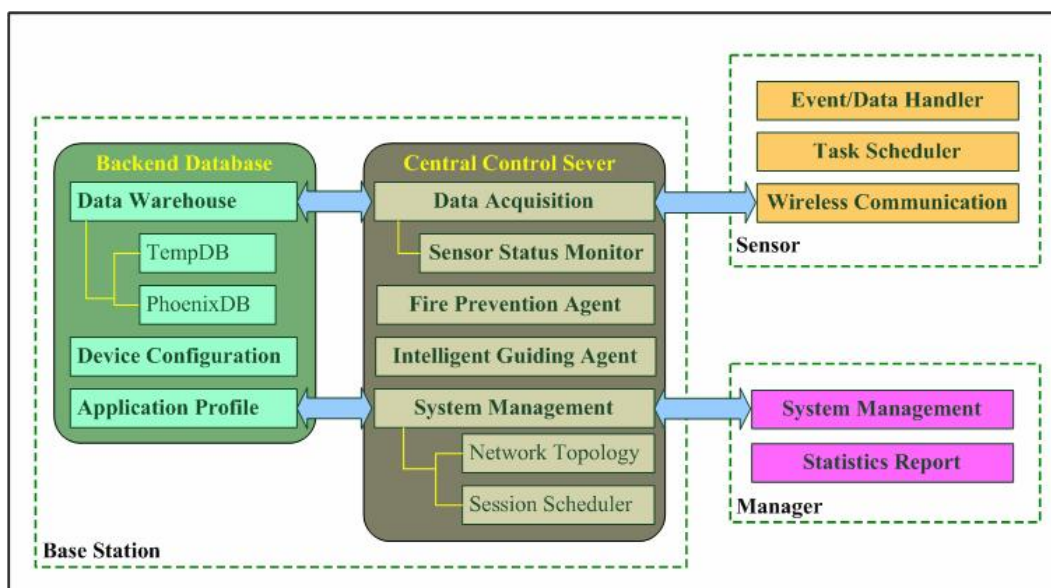
初始過後的感測節點將利用主要展延樹拓撲，回報所蒐集的週遭溫度指數到基地台以供紀錄與觀察，基地台端除了不斷地接收網路所回傳的感測資料外，還必須進階處理這些回報資訊，運算過後的結果可能是系統自我調整或是控制指令，這些進階的結果與判斷亦將利用預設的主要展延樹進行發佈，在下頁圖(十二)中表示出 PHOENIX 系統的控制命令與感測資料流向示意圖。



圖(十二)、PHOENIX 系統之資料與控制流路示意圖。

在 PHOENIX 系統中，基地台端的組成元件配置可依據功能屬性的不同，區分成兩個主要元件，分別為後端資料庫 (Backend Database)、中控伺服器 (Central Control Server)，透過這兩項元件使得基地台得以與系統管理者 (Manager) 以及感測器 (Sensor) 維持聯繫，系統的管理者可透過伺服器存取後端資料庫內所記錄的資料，亦可下達管理指令執行系統維護

的相關程序；系統於平時能夠依造制訂程序自行運作，部分的管理與判斷機制被設計於中控伺服器中定期執行，其中包含感測器狀態監控、建築物溫度回報資料紀錄等功能，系統對於火警事故的緊急處理措施已於上一章當中探討完畢，在中控的伺服器當中將週期性執行這些常態任務以確保系統運行合宜，整體元件的配置圖，參考下圖(十三)所示。



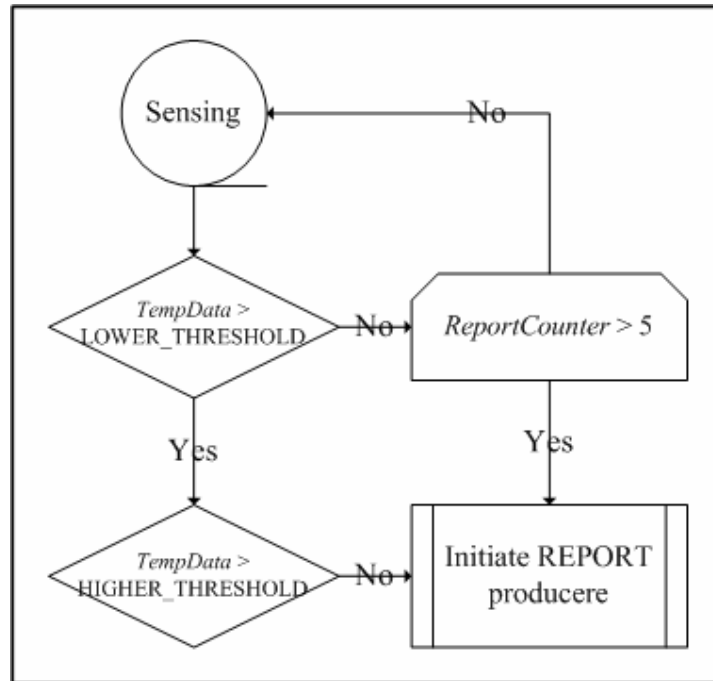
圖(十三)、PHOENIX 系統之元件配置圖。

在圖(十三)中，我們得以窺視到基地台端系統的架構，而此系統我們是以 Java 所撰寫的視窗平台來展現，其中存在幾個獨立的執行緒：火警預防代理 (Fire Prevention Agent, FA)、導引代理 (Intelligent Guiding Agent, IA)，系統維護代理 (System Management, SM)，這幾個執行緒在系統視窗經開啟時便會自動以背景方式執行。

利用泛洪機制建構主要展延樹拓樸，並依此拓樸結構設計了具有無迴路的繞徑協定，提供感測節點回傳溫度、本身狀態等資料時所使用的傳送路徑，這樣的拓樸設計使得繞徑得以於傳送前掌握，進而讓系統本身能夠滿足網路對於省電議題的要求；然而，系統中不斷由感測節點所回傳的溫

度資料居大部分，往往建築物內普遍而言溫度變化量不大，如此使得系統對於相同位置將持續接收到同樣的資訊，針對於此，我們更進一步地闡述本系統中所設計的兩階段回報機制，透過這項設計能夠適當地過濾網路上大量流竄的感測資料封包，這項措施稱為「二階段式回報機制」，在這項進階機制當中，我們設計了兩項抉擇感測資料是否回報的臨界指標溫度：回報臨界溫度與火警臨界溫度，平時感測器依舊維持週期性偵測的常態任務，當有溫度變化事件被觸發時，首先將比對所謂的回報臨界溫度 (LOWER_THRESHOLD)，倘若所偵測之溫度並未超過這個回報臨界常數時，感測器並不會立即回報，同時也意指著目前該感測器所處位置仍處於正常的室溫範圍內，這項系統參數可配合氣候、季節做合理的設定，此外，感測器本身將記錄本身的回報次數，每當未回報次數超過五次時，依然會啟動回報的程序，這裡主要的考量是透過次數提供系統偵測的敏感參數 (sensitivity)，此參數將影響網路運作時間，與資料更新速率等系統效益。

另外，當偵測到的室溫超過回報臨界溫度時，我們以另一項指標參數，即火警臨界溫度 (HIGHER_THRESHOLD)，這是針對事前防範措施而補充設計的，每當溫度參數超過這項臨界指標時，代表週遭環境中出現了任何有可能引發火災的因子，導致偵測溫度過高，此時則必須立即通報基地台，由這兩項臨界溫度作為感測器判斷是否回報的機制，可參照下頁圖(十四)。



圖(十四)、二階段式回報機制流程圖。

在一段執行災情回報程序的過程之後，基地台端將偵測到災情訊息，配合已知的結構藍圖，進而掌握受災區域，此時，基地台端能依據廣度優先搜尋的演算法計算出合理的逃生路線，這些路線同時亦將與系統藍圖的指北方向進行比對，透過夾角幅度判斷出合理的逃生方向，這些導引訊息將發送至網路中目前所有尚可連通的感測節點，藉此導引受災人員正確的逃生策略，在本系統中所設計的導引指示是透過感測器上的預設輸出裝置來顯示正確的逃生方向，由於 Mote 硬體本身目前僅提供三顆發光二極體 (LEDs) 與一個發聲器，因此我們所實作出來的火警雛形系統僅能簡單地顯示表達各種可能的情形，其表達意義如下表(八)。

Composite Expression				Comment
Red Led	Yellow Led	Green Led	Beeper	
Normal Situation				
Dark	Dark	Blinking	Silent	Detect the temperature
Dark	Blinking	Dark	Silent	Battery is low.
Warning Situation				
Dark	Dark	Shining	Beeping	Safe region.
Dark	Shining	Dark	Beeping	Warning region.
Shining	Dark	Dark	Beeping	Dangerous region.
Dangerous Situation				
Blinking	Dark	Dark	Beeping	Direct to North.
Blinking	Dark	Shining	Beeping	Direct to East.
Blinking	Shining	Dark	Beeping	Direct to West.
Blinking	Shining	Shining	Beeping	Direct to South.

表(八)、感測器之顯示燈號與表示意義。

第三節、系統實際操作與示範

本節將深入說明本系統實作過程，系統中分為兩個方向來併行開發，在感測器的操控方面，我們利用微型作業系統 TinyOS 所提供之嵌入式平台，在其上撰寫 NesC 程式碼，所使用的 TinyOS 是 1.1.6 的版本，它包容於 Cygwin 系統底下，讓我們得以在微軟視窗系統中藉以無虞地開發與實現；感測器在實行之前必須透過可程式子板載入我們所設計的程式，這些程式描述著已於前兩節所提及之程序，除了底層藉由 PhoenixCOM 來與感

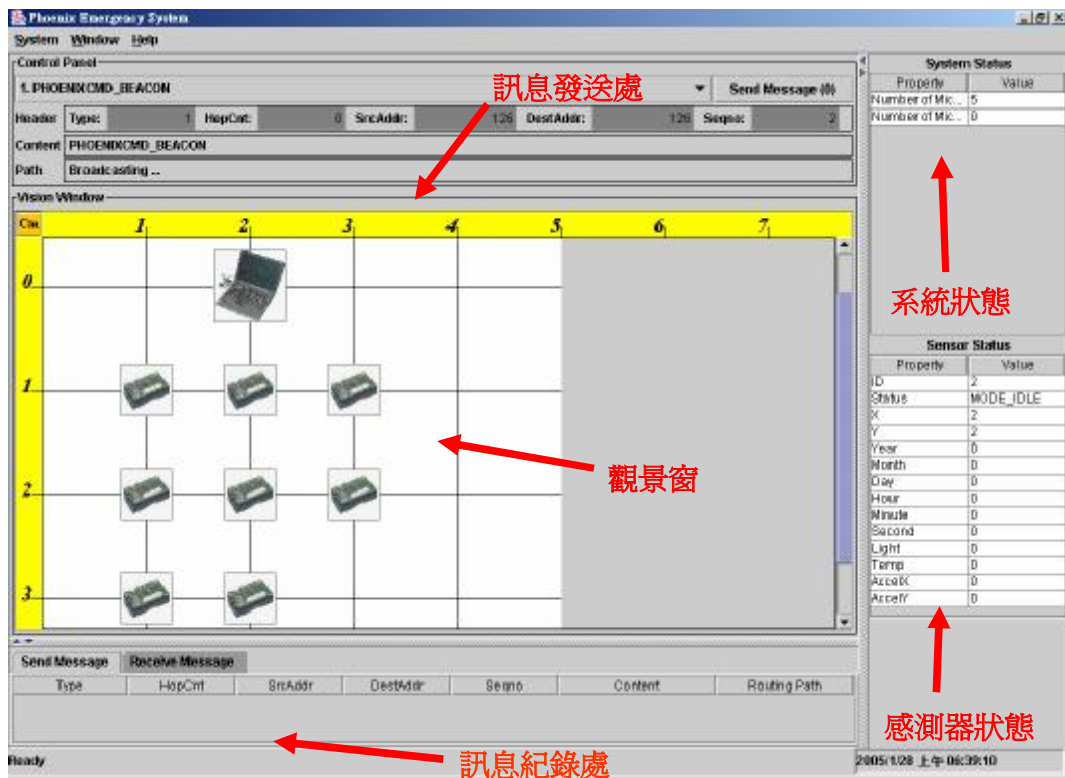
測器聯繫外，其上尚可區分出 PhoenixMAC、PhoenixNWK 以及 PhoenixAPL 三層，前兩層所司網路點對點以及路由等相關工作，最上一層則負責以讀秒計時器來觸發感測器之收集資料功能，在收集資料時，會呼叫 TinyOS 本身對於 Mote 所支援的 ADC 模組來採集光度、溫度、縱向與橫向加速度等值，而這些數值將要求下層的 PhoenixNWK 來遞送，當 PhoenixNWK 接收到由上層而來需要傳送之訊息時，首先會檢查之前再網路初始時所探詢到的父節點，在一般情形下，所有欲回報至基地台端的訊息均是以此程序完成傳送；PhoenixMAC 則是位於 PhoenixNWK 層之下的另一個網路子層，當其上層將來源端、目的端識別碼封裝好後，欲遞送之訊息會來到此處，此時會依造訊息的內容發送至路由資訊所描述的下一站，此時會自動幫訊息內的 hopcnt 欄位加一。

上面所描述為感測器上欲遞送至基地台端之執行過程，而當感測器收到由鄰居節點傳遞而來的訊息時，首先會解出訊息內之 destaddr 欄位，用以判斷是否為本身該收下之訊息封包，在本系統中 destaddr 只有三種可能性：TOS_BCAST_ADDR、TOS_LOCAL_ADDRESS 以及非前兩種，TOS_BCAST_ADDR 為 TinyOS 本身所定義作為廣播用途之識別碼，為常數 0xFF，而 TOS_LOCAL_ADDRESS 則為各類感測器本身之識別碼，這是在載入執行程式前就得為每個感測器分別指定清楚的，如此一來每顆感測器將能接收到各式訊息，其中只有當 destaddr 為前兩種常數時，訊息的進一步內容才會繼續被解析以用作設定或是判斷，若是收到非前兩種識別碼時，則表示此封包只是流經本身，必須再續傳出去乃至正確的目的端點。

在前述時，我們以 destaddr 欄位識別出所接收到之訊息為廣播傳送型態，或者是單一傳送型態；廣播型態的訊息如表(七)中的 0~4 以及 27~30 等，單點傳送型態的訊息如同表中 5~13，當收到這些廣播型態之封包時，除了依訊息內容來調整自身之相關參數外，還將訊息的 hopcnt 欄位累加一，接著便在廣播出去，為求避免感測器之間彼此接收與發送同一廣播訊

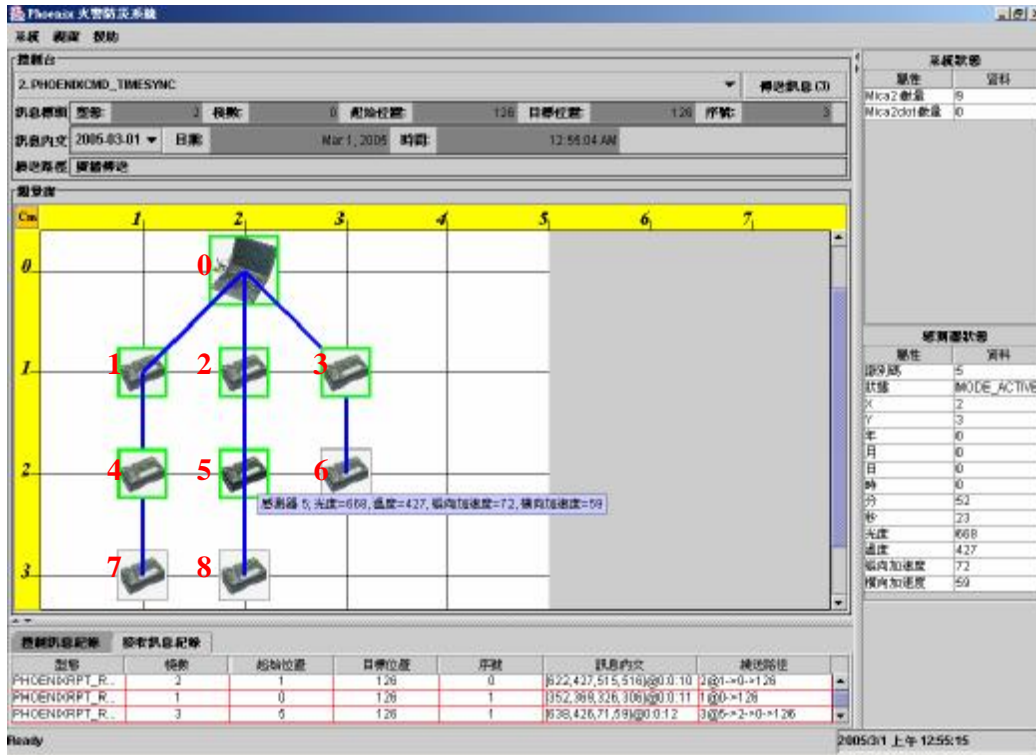
息，我們以 seqno 來判別，具有先前收集過的相同 seqno 欄位之訊息將不會被接收第二次，類似此般的作法亦常見於許多這名協定當中，如同 TCPIP；若當收到單點寄送至此的訊息封包時，此類型封包均為基地台端要求設定某依參數 (SET-)，則再依命令指示之後，以回覆 (RPT-) 20~24 等型態之訊息回覆給基地台端，以利基地台處控管整個網路中，各個感測器之最新即時狀況。

至於基地台端，我們採用昇陽公司 (Sun Microsystems) 之 Java 語言，提供出一套圖形化管理介面，如下圖(十五)所示。



圖(十五)、PHOENIX 火警系統之圖形介面。

在圖(十五)，我們可以透過其中之觀景窗來瀏覽目前網路的各項動態，包括節點數、感測器部署位置，而實際執行起來之畫面實例為下頁圖(十六)所示。



圖(十六)、PHOENIX 火警系統之執行實例。

各個感測器所回報之訊息可由介面中之下方看到，這些訊息會記錄著建築內部各個有佈署感測節點之室溫等資訊；另外，可以透過右方的表格顯示出感測器目前的最新狀態。

在系統端，我們設計了一個獨立的元件 PhoenixCommunication，單獨負責與部署過後的感測器進行通訊，這個元件會自動監聽底層序列阜⁷所回報上來的訊息，並分派至系統的各個主要元件，所接收到的信號的各項 Java Swing 元件在自動反映出即時情境。

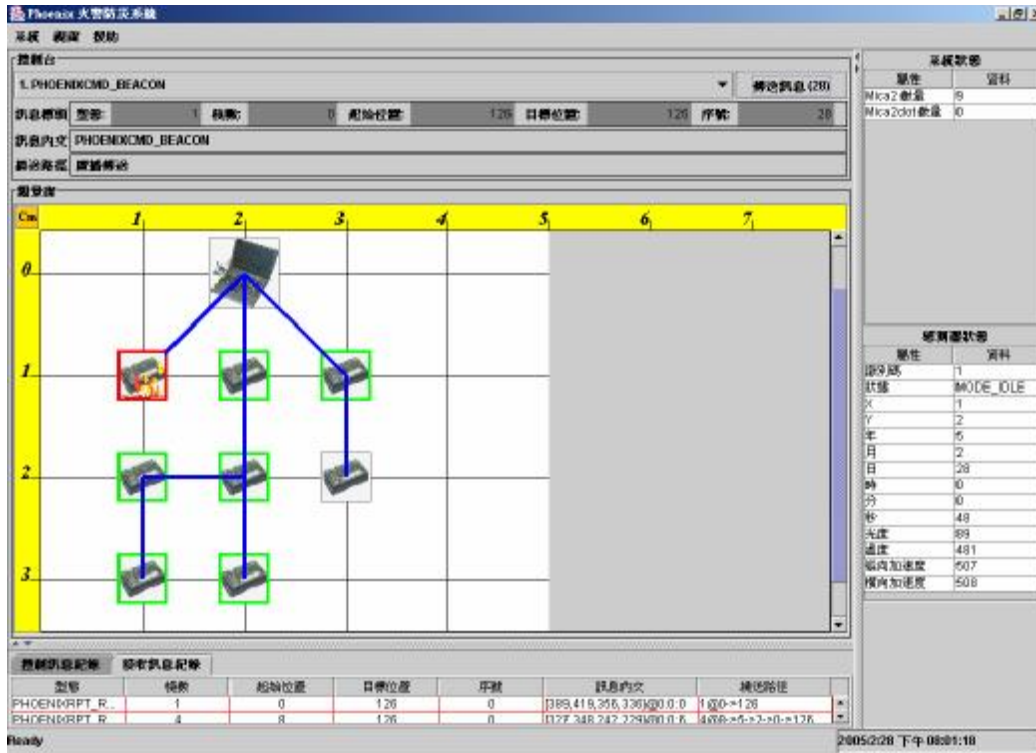
系統除了透過感測器所回傳之定期訊息來掌握感測器最新動態以外，將會隨機發送 ASK 型態的訊息去試探各個感測器，這裡是基於考量到硬體本身的不穩定性，進而由基地台主動查詢感測器。

以上我們已經將一般正常的情形報告妥，依上圖示範的小規模場景，

⁷ 序列阜是指以 RS-232 連線，並設定系統監聽 MOTECOM=serial@57600

我們可以清楚地看到以藍色的線段表示網路連結情狀，我們可利用上一章所談論之利用多棵展延樹之拓撲於基地台端預先計算出來，接著我們發送至各個感測器端，假設系統中預設發送兩棵多重展延樹，此時我們知道本套系統中共存在有三顆展延樹的架構，在平時系統優先以主要展延樹為主要路由依據，然而當火警災變事故發生時，將改以次要的展延樹繼續維持訊息得以繞徑回基地台端，在一般的考量下，誠如第三章所述，我們並不會設定過多的次要展延樹拓撲置網路中，一則是減輕對於訊息過往頻繁的感測網路之負擔，另一則是考量至現實場景中，只有極少的情況允許每一顆感測器彼此均能夠互相通聯，而在這樣的考慮之下，實際感測節點所能探測到的鄰居節點均不至於太多，若我們滋生出過多無謂的展延樹，則勢必有許多連結是被重複到的。

然而，為了在本份論文中方便示範，我們依舊假設目前這五個節點是得以彼此通聯的；首先我們將所有感測器內部對於溫度探測的參數 LOWER_THRESHOLD 設定為 60，HIGHER_THRESHOLD 為 80，在接下來的文字當中，我們實際已適當的熱源接近編號為 1 號之感測器，即刻可由管理介面中看出目前在網路中已有區域著火，如下圖(十七)所示，此時我們可以觀看出 1 號感測器適時地反映在管理台端，而目前網路的連線路由亦重先前的主要展延樹改變為次要展延樹。



圖(十七)、PHOENIX 火警系統之實際操作圖。

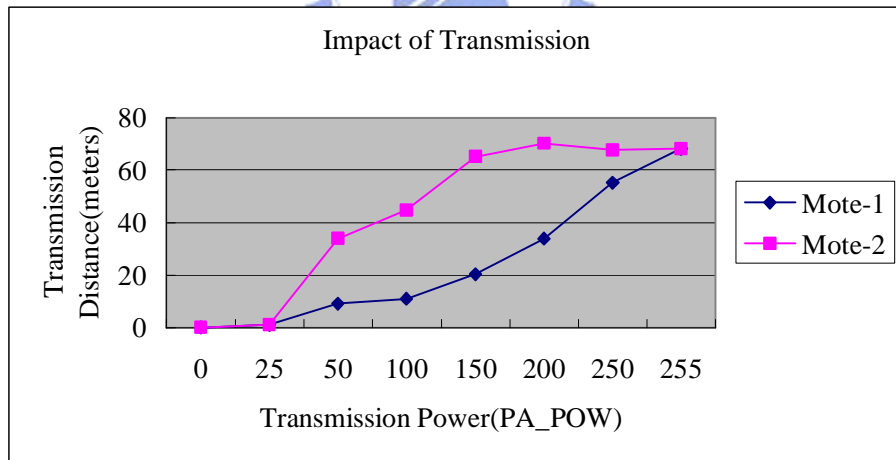
第四節、效益評估

最後我們利用本節來描述在本實作 PHOENIX 火警雜型系統過程中的幾個小實驗結果，以提供給日後對於使用微塵感測器裝置 (MICA2) 來開發其他系統的分享。在多數的情狀下，我們均實際地放置這些 MICA2 motes 來實測，包括於室外草地、建築室內等，然而必須特別談到的是，目前 MICA2 的硬體技術尚且非常不穩定，許多時候我們無法得到預期中的行為產生，這些因素一方面在於其無線電通訊收發不穩定。

就通訊而言，MICA2 的通訊品質與幾個因素相關，包括有天線的長度，使用較長的天線得以擁有較好的收發訊號品質、電池電量，這點將直接影響了傳輸電量，以及室內各種非可見視線 (non-light of sight) 效益的影響，這使得通訊因為週遭的人員走動，車輛等影響。

根據[14]公司之建議規格書，我們可以將 MICA2 的通訊範圍以程式方

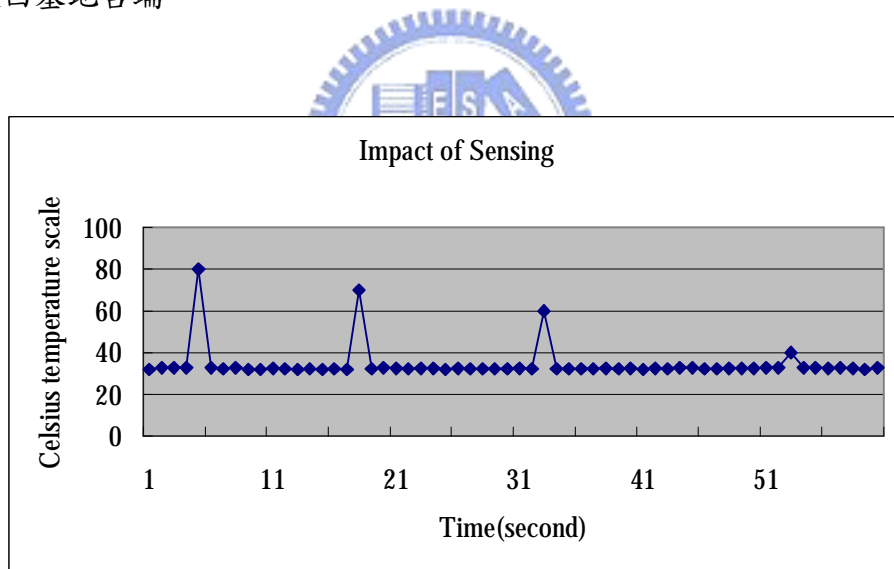
式調整其傳輸電量，總共有 256 種電量階層，簡單的場景設計，我們以兩顆 MICA2 感測器來實驗在空曠的場地中進行這項實驗，其一編號為 1，其二則編號為 2，為避免相互影響，我們將以相同的實驗分別做兩次，以 1 號感測器為實驗組，2 號感測器對照組，令 1 號與 2 號將不斷地發送訊號給 0 號，我們將固定 0 號感測器為測試接收端，當 0 號感測器接收到訊息時，我們將列印於以序列阜連接 0 號的電腦螢幕中，接著我們以步行移動的方式，使得 0 號與 1 號、2 號感測器漸行漸遠，直到收不到訊號時，我們實際量測每種傳輸電量階層所能傳輸之最遠距離，如下表(九)所示，我們可以觀看到對於調整傳輸電量階層所量測出的最遠傳輸距離並非線性，此外 1 號與 2 號分別表現出不同的傳輸範圍，顯然以 2 號感測器得以傳輸至較遠的距離，這項發現令我們得知，即使是相同型號的 MICA2 裝置亦無法保證其有相同的傳輸品質。



表(九)、感測器之傳輸實驗表。

就感測器上所附的各種感測模組而言，我們亦發現到以相同一顆 MICA2 單獨在一個穩定環境內執行簡單收集資料的工作，我們簡單令其每一秒鐘均去採集一次溫度資料並回報，這項實驗只紀錄在一分鐘內的所收

集到溫度資訊，藉以觀察出其感應單元的穩定性，如下表(十)所示，我們可以觀察到，MICA2 本身的感測單元並無法穩定地採集到正確的室溫資料，在一般的情形底下，大部分時間都算是穩定，只是偶爾將會出現採集的溫度激增數倍，這顯然是非正確的數值，目前亦有許多相關的感測器網路之資料彙整 (data aggregation) [10]的研究便是針對這項問題上著手來提供系統層面的解決方案；然而，這些採集到的錯誤訊息都將對本研究中的火警消防系統有所影響，一旦溫度偵測出過高的溫度，將導致感測器發佈危險訊號，針對這項失誤，我們亦得在感測器上每次必須要記憶住前次所採集的溫度資料，在利用這個變數來比對目前所偵測到的資訊是否相差過大，倘若檢驗出是屬於這類型的錯誤時，將會直接忽略掉該筆資訊，而不傳送回基地台端。



表(十)、感測器之感測實驗表。

另一方面，特別值得一提的是，如同前一章所言，MICA2 感測器硬體本身的資料暫存空間過小，其上僅具有 8K 位元組的可程式化快閃記憶體空間，與 4KB 的 SRAM 來存放資料與變數，這使得在開發的階段中，難度上增加了不少；曾經有過在開發的過程中，出現硬體計時器跳動的頻率

不正常，這將導致接下來所有原本該依程式所執行的程序皆亂了次序，歸咎其原因在於程式內某些資料的暫存空間被設定過大，然而這樣的錯誤卻並非是編譯器得以檢查得出來，一旦程式編譯無誤在載入感測器的過程中，由於使用者自訂暫存區過大會擠壓掉原本一些固定要在感測器上執行的基礎程式記憶體區段，但是這樣的錯誤卻著實浪費了不少人力與時間。



第五章、未來展望與結論

隨著無線通訊網路的興起與奈米科技的進步，使得以往體積龐大的電腦及電子裝置可以大幅度縮小，甚至能縮小到如同晶片般大小，也因此，開啟了普及運算的時代。在普及運算的時代中，很重要的特色就是高度的電子化，人們的生活週遭都充斥著電子裝置，所有的裝置將被隱藏在環境當中的某個地方，例如說化身成為服飾的鈕釦或結合在週遭的電器用品之中，普及運算可以讓使用者擺脫目前仍不夠方便的操作模式，未來裡人類可以直接透過語音控制對這些隱式的微型電子設備下達指令，在處理日常生活中的各種事務時，更為便利，甚至還可以取得任何想要的環境資訊，或是與遠方的友人溝通，可以想見在未來的環境中，這些電子裝置所形成的感測器網路對於人類的輔助將愈來愈明顯。

在本篇研究論文當中，我們實際設計了一套火警感測網路系統，這套系統除了提供第參章內文中所談到的三項主要服務功能之外，透過科技進展的角度來看，我們同時也實踐了過去人們所從未想像到的前瞻性應用，下面將本系統於未來尚可研究的方向加以條列整理，以作為後續相關人員參考的依據，最後我們以簡單的文字來闡明在這篇研究論文當中，我們的貢獻之處，以綜整目前的結果。

第一節、未來研究方向

本研究所開發的雛形系統，在功能、服務性、介面上仍有許多有待改進的地方，在未來研究發展的方向有下列幾點：

一、結合更專業的地理資訊系統：

在系統的佈署之初，我們必須以人力手工的方式將所有感測器逐一佈

置於目標建築物內，同時亦將標記這些感測器的位置於結構的對應藍圖之中，待稍後由基地台端設定單一傳輸式指令來通知每顆感測器的實際座標；我們建議本系統可結合目前利用無線網路來實作的定位系統，讓感測器網路可以自行訂定座標，而精確的定位服務有助於導引時期之際，感測器上面所閃爍的方向燈號。

二、提供更多元的服務：

本研究至今仍屬於一雛型系統，距離實際運用於人類生活當中，尚有幾項增值功能有待完成：目前所有系統的狀態，均需要透過基地台端的系統應用程式來呈現，這樣的作法有一明顯的缺點在於住戶無法隨時清楚目前的處境，在未來裡，我們希望將基地台連接網際網路，讓所有的即時資訊都能夠在網路上呈現，而當住戶在外地工作或是旅行時，只要能連上網際網路，輸入正確的帳號、密碼後，便可動態地觀看個人住家的資訊；另外，當有火警發生時，目前的系統中，我們能夠在第一時間之內導引災區內人員儘速逃離，然而，我們希望系統能更進一步地通知在外地工作的住戶，告知其住家目前可能處於火警狀態，而這目標可以透過發送簡訊的方式來實踐，換言之，當建築物目前陷於火警災難之際，基地台可傳送手機簡訊給登記住戶，讓住戶自行決定是否趕忙回到，來搶救財物，減低損失；諸如此類的服務，可以豐富本系統的實用性，提高系統的實際價值。

三、多棵交錯展延樹的運用：

在災情回報的過程中，我們發展了多棵交錯展延樹的網路拓樸來輔助緊急訊息的回報機制，有效地提升系統的容錯程度，然而這樣的機制尚有可進的空間，例如在我們現階段的研究中，需要幾棵展延樹才算足夠，以及當網路節點數量擴張至某一程度時，是否有新的訴求需要

考量，這些問題均需要嚴謹的數理分析與模擬才能得到解答，在下一階段當中，我們將嘗試去分析這套系統的效能，模擬各式網路與建築結構之間的相對關係，來進一步掌握本系統的理论部份。

第二節、本文研究貢獻

本研究所提出之室內火警感測網路，旨在一般建築物內部署許多感測器，透過這些能夠無線傳輸的感測器來監測整個環境，讓基地台的系統能掌管及控制所有的即時資訊，以減低當祝融肆虐時，火場的傷亡人數。為達到這項目標，本系統提供了三項主要功能項目：室溫監測、災情回報以及逃生導引，室溫監測功能可以讓基地台動態觀察建築物內所有部署感測器節點的溫度資訊，統計建築物內各個角落的平均溫度、最高溫度等資訊，以利掌握整棟建築的狀況，災情回報功能讓火災緊急情境發生時，緊急回報的封包不因節點遭受燒損而導致基地台控管失誤，利用多棵展延樹的架構，設計出交錯的效果，來滿足上述的條件要求，適當地修改傳統一棵展延樹的回報繞徑協定，使得交錯展延樹演算法得以實踐，逃生導引功能則在最短時間內，儘速驅散災區內受災人員，令人員依照最短路徑逃去附近的緊急出口處，減低人員傷亡數；另外，本研究採用微塵感測器硬體平台搭配微型作業系統來實作之，我們自行設計合於用途的通訊協定，將上述三大項服務功能實踐於感測器上，自訂訊息封裝的格式與解譯訊息的流程，確實將一套適用於火災場景的感測網路實現於眼前，目前感測網路的實作成果仍屬於相當前瞻的一項技術，我們冀望能透過本研究的微薄貢獻替人類與社稷盡一己棉薄之力。

柒、参考文献

- [1]. I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, “Wireless sensor network: a survey”, *Computer Networks*, vol. 38, pp. 393-422, 2002.
- [2]. M. Weiser, “The Computer for the 21st Century”, *Scientific American*, September 1991.
- [3]. D. Estrin, L. Girod, G. Pottie, M. Srivastava, “Instrumenting the World with Wireless Sensor Networks”, *International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2001)*, Salt Lake City, Utah, May 2001.
- [4]. L. Schwiebert, S. Gupta, J. Weinmann, "Research Challenges in Wireless Networks of Biomedical Sensors", *ACM Mobicomm 2001*, pp. 151-165.
- [5]. A. Mainwaring , D. Culler , J. Polastre , R. Szewczyk, and J. Anderson, “Wireless sensor networks for habitat monitoring”, *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pp. 28-28, 2002.
- [6]. S. D. Glaser. “Some real-world applications of wireless sensor nodes”, *Proceedings of SPIE Symposium on Smart Structures & Materials/ NDE*, San Diego, California, pp. 14-18, 2004.
- [7]. C. Perkins, “Ad Hoc Networks, Addison-Wesley”, Reading, MA, 2000.
- [8]. D. Johnson, D. Maltz, Y-C. Hu and J. Jetcheva, “The dynamic source routing protocol for mobile ad hoc networks,” *Internet Draft, Internet Engineering Task Force*, Mar 2001.
- [9]. C. Perkins, E. Royer, and S. Das, “Ad hoc on-demand distance vector (aodv) routing,” *Internet Draft, Internet Engineering Task Force*, Mar. 2001.
- [10]. Tian He, B. M. Blum, John A. Stankovic, and Tarek F. Abdelzaher, “AIDA: Adaptive Application Independent Data Aggregation in Wireless Sensor Networks”, *ACM Transactions on Embedded Computing System Special issue on Dynamically Adaptable Embedded Systems*, Volume 3, Issue 2, 2004.
- [11]. Lawrence A. Klein, “Sensor and Data Fusion - A Tool for Information and Decision Making”, *SPIE*, 2004.
- [12]. Berkeley WEBS: Wireless Embedded Systems, web site:
<http://www.bfrl.nist.gov/>
- [13]. WINS: Wireless Integrated Network Sensors, web site:
<http://www.janet.ucla.edu/WINS/>

- [14]. Inertial, MEMS Gyro, FAA Certified AHRS, and Smart Dust Wireless Sensors from Crossbow, web site: <http://www.xbow.com/>
- [15]. TinyOS Community Forum. An open-source OS for the networked sensor regime. web site: <http://www.tinyos.net/>
- [16]. Nanoscience Instruments, web site: <http://www.nanoscience.com/>
- [17]. NEST Project at Berkeley, web site: <http://webs.cs.berkeley.edu/nest-index.html>
- [18]. SensoNet Project, BWN-Lab, web site: <http://users.ece.gatech.edu/~weilian/Sensor/>
- [19]. A. Sinha and A. Chandrakasan, "Dynamic Power Management in Wireless Sensor Networks", IEEE Design & Test of Computers, Vol. 18, No. 2, March-April 2001
- [20]. W. B. Heinzelman, P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," IEEE Transactions on Wireless Communications, vol. 1, no. 4, pp. 660-670, Oct. 2002.
- [21]. K. Sohrabi, J. Gao, V. Ailawadhi, G.J. Pottie, "Protocols for self-organization of a wireless sensor network," IEEE Personal Communications, pp. 16-27, Oct. 2000.
- [22]. K. Sohrabi and G. J. Pottie, "Performance of a novel self-organization protocol for wireless ad hoc sensor network," in Proceedings of the IEEE 50th Vehicular Technology Conference, pp. 1222-1226, 1999.
- [23]. W. Ye, J. Heidemann, D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," IEEE INFOCOM 2002, pp. 1567-1576, 2002.
- [24]. W. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks," in Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '99), pp. 174-185, Aug. 1999.
- [25]. C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, F. Silva, "Directed diffusion for wireless sensor networking," IEEE/ACM Transactions on Networking, vol. 11, no. 1, pp. 2-16, Feb. 2003.
- [26]. M. Zorzi, and R. R. Rao, "Geographic Random Forwarding (GeRaF) for ad hoc and sensor networks: energy and latency performance," IEEE Transactions on Mobile Computing, vol. 2, no. 4, Oct.-Dec. 2003.
- [27]. Zorzi, and R. R. Rao, "Geographic Random Forwarding (GeRaF) for ad hoc and sensor networks: multihop performance," IEEE Transactions on Mobile Computing, vol. 2, no. 4, Oct.-Dec. 2003.
- [28]. J. Deng, Y. S. Han, B. Heinzelman, and P. K. Varshney, "Scheduling

Sleeping Nodes in High Density Cluster-based Sensor Networks,”
submitted to the IEEE International Conference on Communications
2004.

- [29]. J. Deng, Y. S. Han, B. Heinzelman, and P. K. Varshney, “Balanced-energy sleep scheduling scheme for high density cluster-based sensor networks,” submitted to the Third International Symposium on Information Processing in Sensor Networks (IPSN’04).
- [30]. G. Hoblos, M. Staroswiecki , A. Aitouche, “Optimal design of fault tolerant sensor networks,” IEEE International Conference on Control Applications, Anchorage, AK, pp. 467-472, September 2000.
- [31]. T.-Y. Wang, Y. S. Han, and P. K. Varshney, “Fault-Tolerant Classification in Multisensor Networks Using Coding Theory,” in Proceedings of the Sixth International Conference of Information Fusion, pp. 772-779, 2003.
- [32]. T.-Y. Wang, P. K. Varshney, and Y. S. Han, “Distributed Fault-Tolerant Classification Fusion Using Error Correcting Codes,” in Proceedings of SPIE, Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications, vol. 5099, pp. 47-57, April 2003.
- [33]. A. Perrig, R. Szewczyk, V. Wen, D. Culler, J.D. Tygar, “SPINS: Security protocols for sensor networks,” Mobile Computing and Networking 2001 Rome, Italy, ACM.
- [34]. L. Eschenauer and V.D. Gligor, “A key-management scheme for distributed sensor networks,” in Proceedings of the 9th ACM Conference on Computer and Communication Security, pages 41-47, Nov. 2002.
- [35]. H. Chan, A. Perrig, and D. Song, “Random Key Predistribution Schemes for Sensor Networks,” 2000 IEEE Symposium on Security and Privacy, May 11-14, 2003, Berkeley, CA.
- [36]. W. Du, J. Deng, Y. S. Han, and P. K. Varshney, “A Pairwise Key Pre-distribution Scheme for Wireless Sensor Networks,” 10th ACM Conference on Computer and Communications Security (CCS2003), Washington DC, Oct. 2003.