



國立勤益科技大學
電子工程系研究所

碩士論文

使用熱電堆的無線式溫度量測裝置之實現

**Implementation of a Wireless
Temperature-measuring Device Using
Thermopile**

研究生：黃威盛

指導教授：蔡錦福 博士

中華民國 一百零一 年 七 月

國立勤益科技大學
研究所碩士班
論文口試委員會審定書

本校 電子工程系 碩士班 黃威盛 君

所提論文 **使用熱電堆的無線式溫度量測裝置之實現**

合於碩士資格水準，業經本委員會評審認可。

論文口試委員會：

召集人：

許書旂

委員：

林水春

蔡錦福

指導教授：

蔡錦福

所

長：



中華民國一〇一年七月

使用熱電堆的無線式溫度量測裝置之實現

學生：黃威盛 指導教授：蔡錦福 博士

國立勤益科技大學電子工程系研究所碩士班

摘要

無線傳感器網路 (Wireless Sensor Network, WSN) 是未來建立智慧型住宅的重要平台。本研究藉著測量物體輻射功率的非接觸式溫度監測，設計便攜式遠端測溫裝置，使用低功耗的微處理器MSP430做為核心的資料處理與數位化的轉換，配合無線傳輸模組ZigBee，使用SMD元件設計印刷電路板，有效縮小電路面積，將其模組化，該設備將作為感測節點的WSN發展。本研究使用熱電堆感測元件作為溫度感測的窗口，測量範圍是從 0°C 到 300°C ，平均誤差在 $\pm 3^{\circ}\text{C}$ ，並增加遠端輻射測量的品質。在這項研究中，圖控軟體LabVIEW被選定為電腦端的開發介面，詳細記錄數據並顯示在個人電腦上。

關鍵字：熱電堆、紅外線、黑體輻射、ZiBee(XBee)、MSP430。

Implementation of a Wireless Temperature-measuring Device Using Thermopile

Student: Wei-Sheng Huang

Advisor: Dr. Chin-Fu Tsai

Institute of Electronic Engineering
National Chin-Yi University of Technology

Abstract

Wireless Sensor Networks (WSN) is necessary for an intelligent house in the future. This paper describes a non-contact thermometer by detecting an object's radiant power. We will attempt to design a portable device for remote measuring temperature and then to develop a sensing node for WSN. A thermopile equipped with a lens was used to implement the performance. The detection range is from 0 °C to 300 °C. The average error is within ± 3 °C. In this study, Laboratory Virtual Instrument Engineering Workbench (LabVIEW) software plays an important role for designing thermistor linearization and data logging. This study may

provide a useful reference for researchers attempting to increase quality of remote radiometry.

Keywords : thermopile, thermistor, infrared, emissivity, blackbody



致謝

首先感謝國科會計畫撥出研究經費，令本實驗室能有更多的實驗器材與環境，也能為台灣的科技創新產業多一點的貢獻。感謝這兩年來指導教授傳授感測與轉換的專業知識和多方面的要求，提升了個人的專業能力與知識，使我在社會工作上更具競爭力。

感謝學長吳元凱與學弟劉以琳，在我研究遇到問題時有共同討論的對象，也感謝我的父母提供我就讀碩士學歷的機會，這兩年來我受益良多，除了學習專業知識，我更學會了參與團隊合作與磨練如何解決問題的能力，未來期待自己能以樂觀的心態積極面對職場、人生。

目錄

中文摘要.....	3
英文摘要.....	4
致謝.....	6
圖目錄.....	10
表目錄.....	13
第一章、緒論	
1.1 研究背景.....	14
1.2 研究動機與目的.....	15
第二章、理論基礎	
2.1 相關研究.....	16
2.2 熱電堆測溫原理.....	17
第三章、硬體設計	
3.1 熱電堆檢測架構.....	20
3.2 電路架構.....	22
3.2.1 環境溫度補償電路.....	22
3.2.2 熱電堆測溫電路.....	24
3.2.3 輻射測溫線性化.....	25
3.3 單晶片微控制器.....	27

3.3.1	Timer 設定.....	30
3.3.2	ADC 設定.....	31
3.3.3	流程圖.....	35
3.4	ZigBee 無線傳輸.....	36
3.4.1	無線傳輸模組.....	38
3.4.2	操作方式.....	39
3.4.3	工作模式.....	40
3.4.4	傳輸簡易測試.....	42
第四章、資料擷取程式設計		
4.1	LabVIEW 簡介.....	45
4.2	資料擷取DAQ 流程圖.....	46
4.2.1	RS232 接收測試.....	48
4.3	溫度測量流程圖.....	49
第五章、實驗平台與結果		
5.1	實驗平台設置.....	50
5.1.1	溫度校準.....	53
5.2	DAQ 資料擷取結果.....	54
5.3	實驗結果.....	57
5.4	討論.....	63

第六章、結論與未來展望

6.1 結論.....64

6.2 未來展望.....65

第七章、參考文獻.....66

第八章、附錄

8.1 MSP430 程式.....69

8.2 LabVIEW 程式.....72

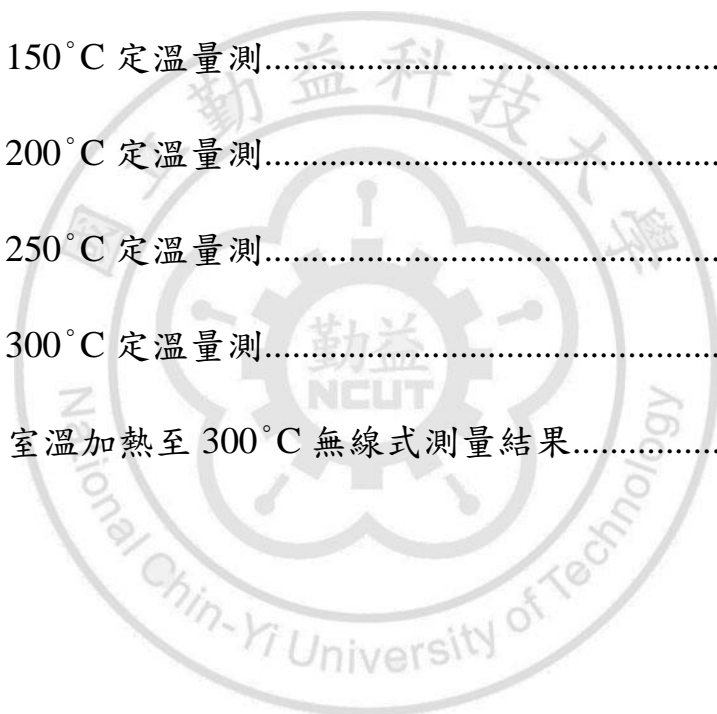


圖目錄

圖 2.1	熱電堆感測器的內部結構.....	17
圖 2.2	熱電堆測溫示意圖.....	18
圖 3.1	信號處理方塊圖.....	20
圖 3.2	TPS 334 L10.6 實體圖.....	21
圖 3.3	頻率響應圖.....	21
圖 3.4	環境溫度補償電路圖.....	22
圖 3.5	線性化解決方案.....	23
圖 3.6	TP 信號放大電路圖.....	24
圖 3.7	AD538 電路圖.....	25
圖 3.8	IAR 開發環境.....	28
圖 3.9	MSP430 內部架構圖.....	29
圖 3.10	工作電壓與頻率關係圖.....	29
圖 3.11	MSP430 內部 Timer 架構圖.....	30
圖 3.12	MSP430 內部 ADC 架構圖.....	31
圖 3.13	ADC10CTL0(ADC10 控制暫存器 0).....	32
圖 3.14	MSP430 程式流程圖.....	35
圖 3.15	ZigBee 工作網路拓樸方式.....	36
圖 3.16	XBee 設定.....	39

圖 3.17	XBee 工作模式.....	40
圖 3.18	點對點模式.....	41
圖 3.19	點對多模式.....	41
圖 3.20	廣播模式.....	41
圖 3.21	XBee 傳輸簡易測試.....	42
圖 3.22	終端機接收設定.....	43
圖 3.23	終端機接收資料圖.....	44
圖 3.24	接收端 XBee 模組實體圖.....	44
圖 4.1	DAQ Assistant 設定.....	46
圖 4.2	DAQ 資料擷取之 LabVIEW 流程圖.....	47
圖 4.3	LabVIEW 函式之 RS232 設定圖.....	48
圖 4.4	溫度測量 LabVIEW 流程圖.....	49
圖 5.1	Gemini R 實體圖.....	50
圖 5.2	MSP430F2012 信號處理電路.....	51
圖 5.3	PCB Layout 圖.....	51
圖 5.4	SMD 元件.....	52
圖 5.5	電路實體圖.....	52
圖 5.6	溫度校準平台設置.....	53
圖 5.7	黑體爐設定 100°C 量測結果.....	54

圖 5.8	黑體爐設定 200°C 量測結果.....	55
圖 5.9	黑體爐設定 300°C 量測結果.....	55
圖 5.10	室溫加熱至 300°C 連續測量結果.....	56
圖 5.11	系統方塊圖.....	57
圖 5.12	50°C 定溫量測.....	58
圖 5.13	100°C 定溫量測.....	58
圖 5.14	150°C 定溫量測.....	59
圖 5.15	200°C 定溫量測.....	59
圖 5.16	250°C 定溫量測.....	60
圖 5.17	300°C 定溫量測.....	60
圖 5.18	室溫加熱至 300°C 無線式測量結果.....	63



表目錄

表 3.1 其他開次方根阻值	26
表 5.1 溫度測量數據	61



第一章、緒論

1.1 研究背景

樂活(Lifestyle of Health and Sustainability, LOHAS)的觀念使「健康和環境責任」成為大家關注的問題。因此，一個能兼具有健康、安全、舒適、方便與節能等功能的智慧型住宅將是未來的生活模式。而智慧型住宅必備的功能就是防火與保全。在自動消防系統中，一間智慧型住宅必須取得一些重要的資訊，如溫度、溼度、空氣品質、照明度和能源消耗等，所以偵測屋內的溫度變化是必要的[1]。

無線感測網路(Wireless Sensor Networks, WSN)為一新的信息收集平台，其使用大量自動的感測節點[2]，IEEE 802.15.4 標準提供這些小尺寸和低功耗設備節點間的無線連接[3]，最近物聯網(Internet-of-Things, IOT)，提供了一個全球性智能化的生活網路，這些網路可使用在許多應用上，如自動消防系統、家庭環境的智能燈光控制...等，未來通過 WSN 與 IOT 的應用，智慧型住宅的建造可以更輕鬆。

1.2 研究動機與目的

一些研究人員已經研究使用焦電型和熱電堆測量溫度[4-8]，但沒有探討在實際環境中使用熱電堆以遠端測量溫度[9-10]，本研究專注於紅外線技術的非接觸式測量方法的研究，其目的是使用熱電堆實現溫度測量，測量範圍為 0°C 到 300°C ，以便提供預防火災用途。本研究採用無線的通訊方式，其中ZigBee是由IEEE 802.15.4與Zigbee Alliance組成，分別制訂硬體與軟體標準，它是一種低傳輸速率(250kbps)、短距離(50~100m)、低消耗功率、架構簡單的技術。Zigbee/IEEE802.15.4協定以低功率(low-power)、低價位(low-cost)為目的的無線個人區域網路(Wireless Personal Area Network, WPAN)發展為WSN所用[11]，搭配佈署於家中的感測器藉以整合空調、門禁、機電、火警功能等架構，本研究結果是要結合煙霧感測...等模組化電路後，形成一個完整的防火系統，使智慧型住宅具有防火的保護功能。

第二章、 理論基礎

2.1 相關研究

常見的溫度感測元件可以分為四大類型：電阻型、焦電型、感溫積體電路與熱電偶。電阻型是利用電阻隨溫度變化的方式測量溫度[12-13]，最常用的是使用鉑，在 0°C 時電阻為 100Ω 的元件(Pt100)；焦電型是利用表面電荷隨溫度變化的原理，測量其電荷強度以反應溫度[14-15]；感溫積體電路是利用半導體材料之導電易受溫度影響的特性；熱電偶是使用兩種不同性質的金屬端點接觸，利用熱電效應測量溫度[16-17]，當熱電偶兩端有溫度差時，即會產生一個電動勢，反之，外加一個電壓時兩端也會產生溫度差。

本研究使用結合了熱電偶和電阻型的溫度感測器。而將熱電偶串聯或者並聯形成堆狀即稱為熱電堆，熱電堆內因溫度變化產生的電動勢比單個熱電偶所產生的大得多，所以對測量溫度的靈敏度相對的提高許多。

2.2 熱電堆測溫原理

熱電堆感測器(Thermopile sensor, TP)是由幾個微型熱電偶串聯而成的熱電堆與熱敏電阻(Thermistor, TH)所組成，如圖 2.1 所示。

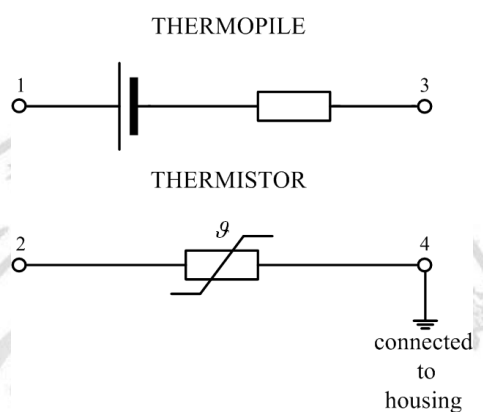


圖 2.1 熱電堆感測器的內部結構

根據 Planck 推導出的公式，以描述黑體(Black body)輻射與熱力學溫度之間的關係如公式 2-1 所示。

$$M_{e,\lambda}(\lambda, T) = \frac{2\pi hc^2}{\lambda^5 \left(e^{\frac{hc}{\lambda kT}} - 1 \right)} = M_{q,\lambda}(\lambda, T) \frac{hc}{\lambda} \quad (2-1)$$

其中 $M_{e,\lambda}(\lambda, T)$ 為黑體的光譜輻射度 ($W / (m^2 \cdot \mu m)$) ; λ 為波長 (μm) ; T 為黑體的絕對溫度 K ; h 為 Planck 常數 ($h = 6.626196 \times 10^{-34} J \cdot s$) ; k 為 Boltzmann 常數 ($k = 1.38054 \times 10^{-23} J / K$) ; c 為真空中的光速 ($c = 2.9979 \times 10^8 m / s$) 。將 Planck 公式從 0 到 ∞ 的波長範圍進行積分就可以得到 Stefan-Boltzmann 定律。對於表面溫度 T_o 的灰體 (Grey body) , 其總輻射功率密度 $M_e(T)$ 如公式 2-2 所示。

$$M_e(T) = \eta \sigma T_o^4 \quad (2-2)$$

η 是物體的放射率， σ 是 Stefan-Boltzmann 常數。因此，通過測量其輻射功率，無須與物體接觸，就能測量物體的表面溫度。藉由 TH 得以測量所在環境的溫度；而出現在感測器的視野角 (Field of view, FOV) 內的物體表面溫度與環境溫度的差異，會使 TP 產生一個輸出電壓，如圖 2.2 所示。

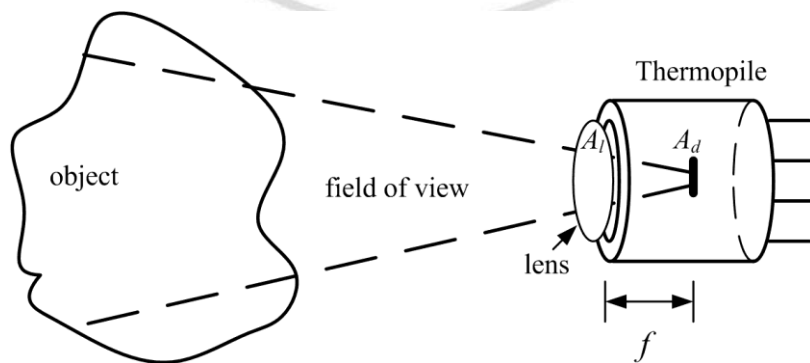


圖 2.2 熱電堆測溫示意圖

其輸出電壓與 TP 感測的輻射功率 Φ_d 成正比，如公式 2-3 所示。

$$\begin{aligned}\Phi_d &= \frac{\sigma(\eta_o T_o^4 - \eta_d T_d^4) \tau A_d A_l}{\pi f^2} \\ &= k(\varepsilon_o T_o^4 - \varepsilon_d T_d^4)\end{aligned}\quad (2-3)$$

τ 是穿透率， f 是焦距， o 、 d 與 l 符號分別代表待測的物體 (object)、偵測器 (detector) 和鏡頭 (lens)。 k 則定義為儀器因子 (Instrument factor)，包含了常數 σ 和參數 η 、 A 和 f 。



第三章、硬體設計

3.1 熱電堆檢測架構

本研究檢測架構如圖 3.1 所示的信號處理方塊圖，分成 TP 與 TH 兩部分經過信號處理電路最後總和信號可以得到溫度信號。使用的感測元件(TPS 334 L10.6) 如圖 3.2 所示，是由 PerkinElmer 公司所生產，為紅外線檢測物體溫度的感測元件，本元件包含一個 TP 和負溫度係數的 TH。TH 在 25°C 時的基準電阻值為 30kΩ，作為環境溫度補償；感測元件 TP 用於測量物體與 TH 之間的溫差，對輻射具有 1μm 到 20μm 的靈敏度。此外感測元件附加了一個光學鏡頭，視野角為 8°。圖 3.3 所示為其頻率響應圖，理想工作頻率為 10Hz 以內。

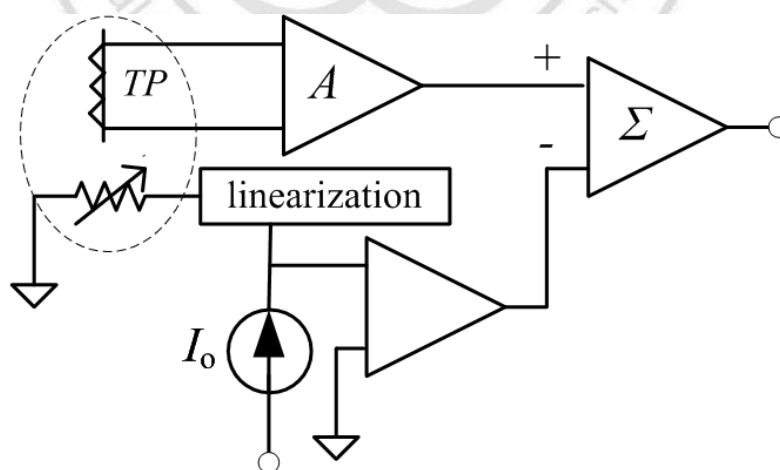


圖 3.1 信號處理方塊圖



圖 3.2 TPS 334 L10.6 實體圖

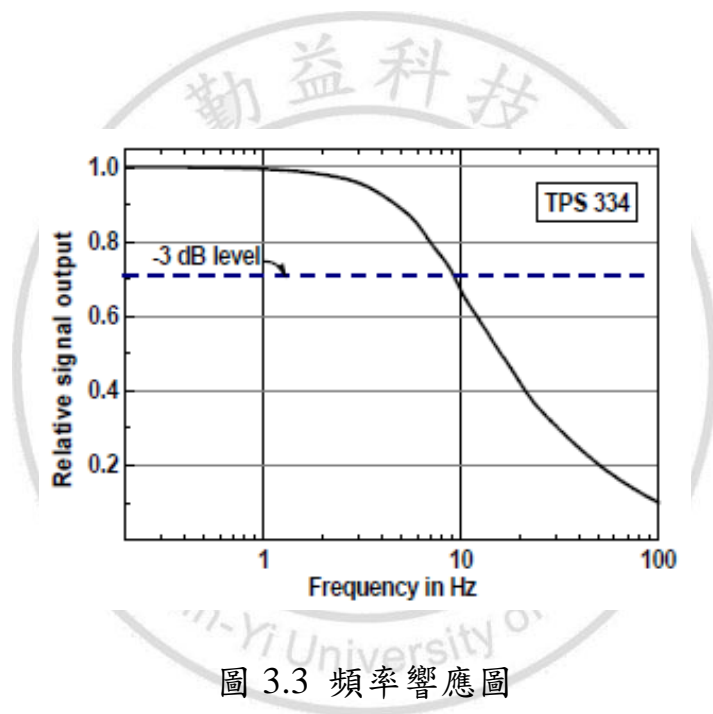


圖 3.3 頻率響應圖

3.2 電路架構

3.2.1 環境溫度補償電路

由於 TH 的阻值變化是非線性的溫度特性，因此，必須經過一個線性化的處理電路，使信號得到線性輸出。基於此，使用一個熟知的方法，以串聯和並聯的組合如圖 3.4 所示，在圖中 R_1 和 R_2 用於線性化 TH[18]。

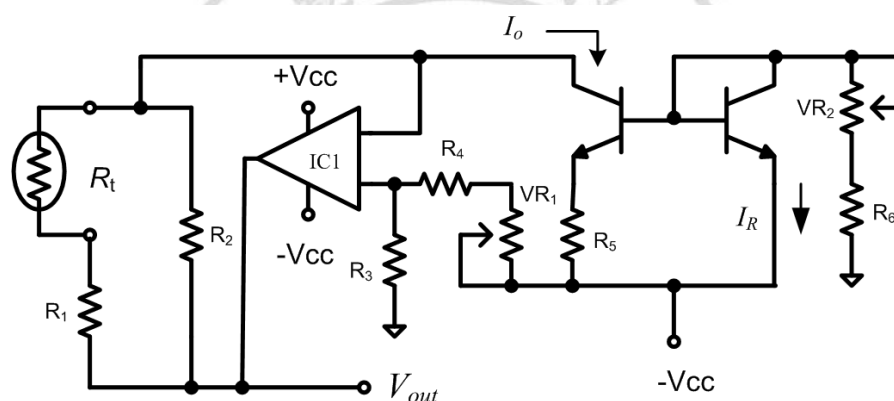


圖 3.4 環境溫度補償電路圖

透過 LabVIEW 軟體設計，由 R_1 和 R_2 以獲得 $-50\Omega/^\circ\text{C}$ 的靈敏度，正如圖 3.5 所示，線性化後的特性曲線幾乎是一直線(白色曲線)，線性的有效範圍從 10°C 到 40°C 並有 0.1°C 內的誤差，此外 Wedlar 電流源電路用於提供定電流 I_0 於線性電路，設置在 $200\mu\text{A}$ 的定電流，從而得到了 $-10\text{mV}/^\circ\text{C}$ 的靈敏度。

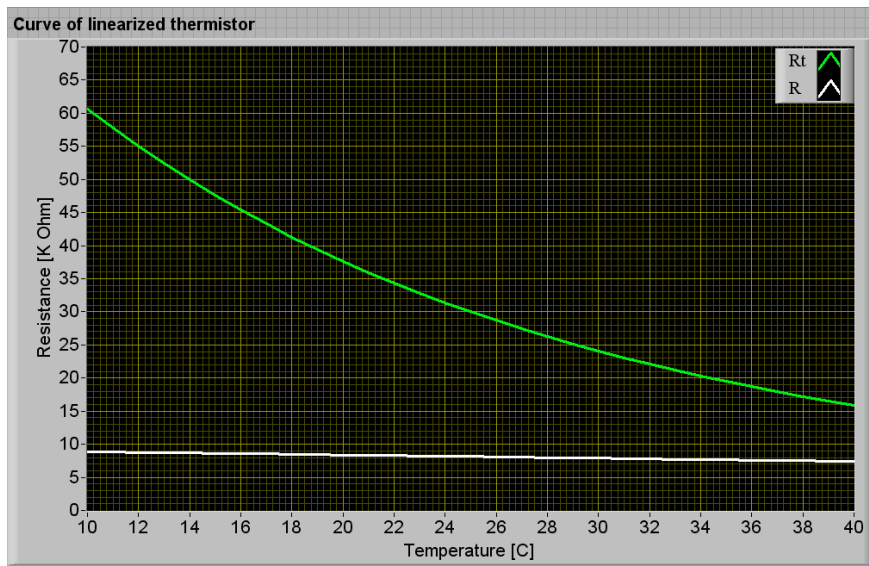


圖 3.5 線性化解決方案($R=(R_t+R_1)/R_2$)



3.2.2 熱電堆測溫電路

經由 TP 測溫後的信號經過第一級濾波後，再由第二級放大，這邊需要反覆的測量並記錄下其輸出電壓，然後計算出所需的 gain 值，並調整其電壓輸出靈敏度為 $10\text{mV}/^\circ\text{C}$ ，電路圖如 3.6 所示。

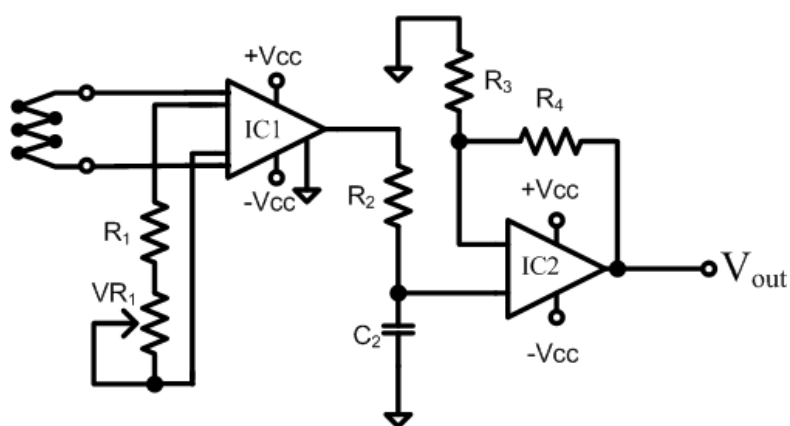


圖 3.6 TP 信號放大電路圖

感測元件使用隔離線與電路做連接，會受隔離線的介面電容 C 影響，如公式 3-1 所示，

$$C = \frac{2\pi\epsilon_r\epsilon_0 L}{\ln\left(\frac{b}{a}\right)} \quad (3-1)$$

其中， ϵ_r 為介質的相對系數， ϵ_0 為真空中的介電系數， L 為隔離線的總長， a 、 b 分別為隔離線的內外徑。隔離線的總長 L 愈大介面電容 C 愈大，導致電容抗 X_c 值下降，分壓後的結果導致整個感測元件的輸出信號過小。

3.2.3 輻射測溫線性化

由公式 2-2 可得知，TP 之輸出電壓與絕對溫度之比為溫度 T 之四次方，由此可知輸出電壓之特性曲線並非線性。因此，經由 TP 之輸出電壓，來測定絕對溫度 T ，必須通過現性化電路如圖 3.7 所示，但沒有理想的物體符合 T^4 光譜響應範圍，故修正為 $T^{4-\delta}$ 。本研究尋求開三次方根電路或開二次方根電路做為線性化的解決方案，如表 3.1 所示。

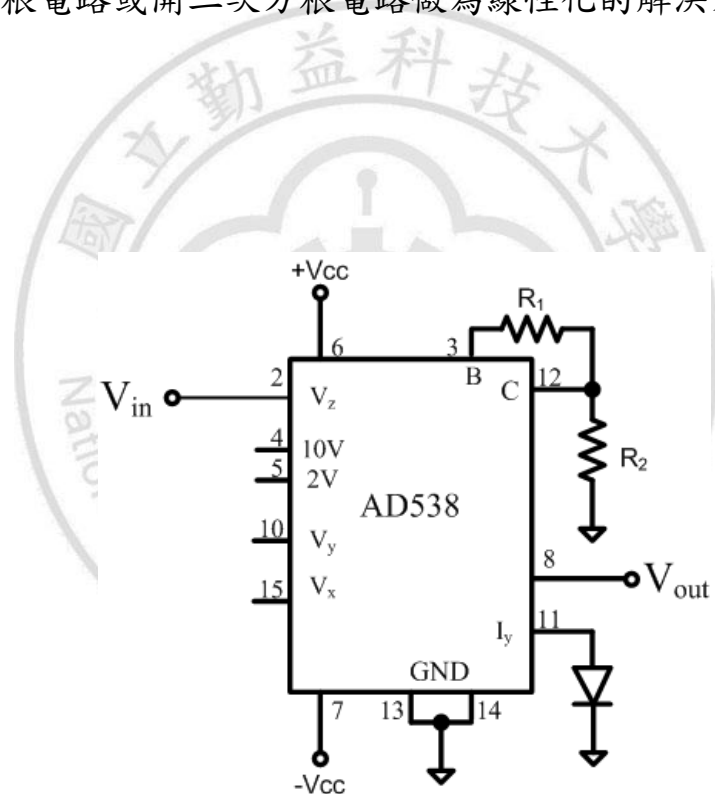


圖 3.7 AD538 電路圖

$$V_{out} = V_y (V_{IN} / V_x)^M \quad (3-2)$$

表 3.1 其他開次方根阻值

	R ₁	R ₂
開二次方根	100Ω	100 Ω
開三次方根	100 Ω	49.9 Ω
開四次方根	150 Ω	49.9 Ω

根據本研究的工作電壓不能驅動 10V 的參考電壓，所以採用的參考電壓為 2V，其第四接腳(參考電壓 10V)必須與第五接腳(參考電壓 2V)接在一起時，此時參考電壓才設定為 2V，此顆 IC 是將信號取對數關係再除以次方根數，最後取反對數可得到開次方根的效果，套入參考電壓後以開四次方根為例，其輸出電壓可以由公式 3-3 更改為公式 3-4 所示。

$$V_{out} = 2(V_{IN} / 2)^{0.25} (V) \quad (3-3)$$

3.3 單晶片微控制器

單晶片微控制器為感測器收集之信號及對外通訊裝置之控制核心，本研究採用由 TI 德州儀器生產之 MSP430 混合訊號處理器(Mixed Signal Processor)。MSP430 系列處理器是 TI 公司於 1996 年推出的一種 16 位元 RISC 超低功耗混合訊號處理器，又稱為「綠處理器」，價格低廉、小型、快速、靈活，是業界使用的控制器中功耗最低的元件，可以快速開發醫療、工業與消費類嵌入式系統。其主要針對實際應用的需求將許多硬體功能整合於內部，可大舉提升資料處理速度，降低硬體規劃空間，減少功率消耗。因此可由控制器有效管控電力分配問題。

本研究中使用此系列的 MSP430F2012，其特徵如下：

- ◆ 電源電壓範圍：1.8 ~ 3.6V
- ◆ 超低功耗：全速執行下耗電流為 220 μ A，睡眠模式下只有 0.5 μ A
- ◆ 2 組捕捉/比較暫存器的 16 位元定時器 Timer_A
- ◆ 內建 10 位元 ADC 支援 8 通道序列轉換
- ◆ 支援 SPI 和 I2C 串列傳輸介面

系統開發環境使用 IAR Embedded Workbench IDE 如圖 3.8 所示，其提供一完全的整合發展方案，包括專案管理員、程式編輯器、編譯工具及 C-SPY Dugger 等，在連續的工作流程，可以建立原始檔案及專案，編輯相關應用，並可在軟體模擬器(Simulator)或硬體上進行除錯。

IAR Systems 以其高度最佳化之編譯器聞名於世，此套軟體支援標準 C 語言和組合語言，本研究是使用 C 語言來撰寫程式。

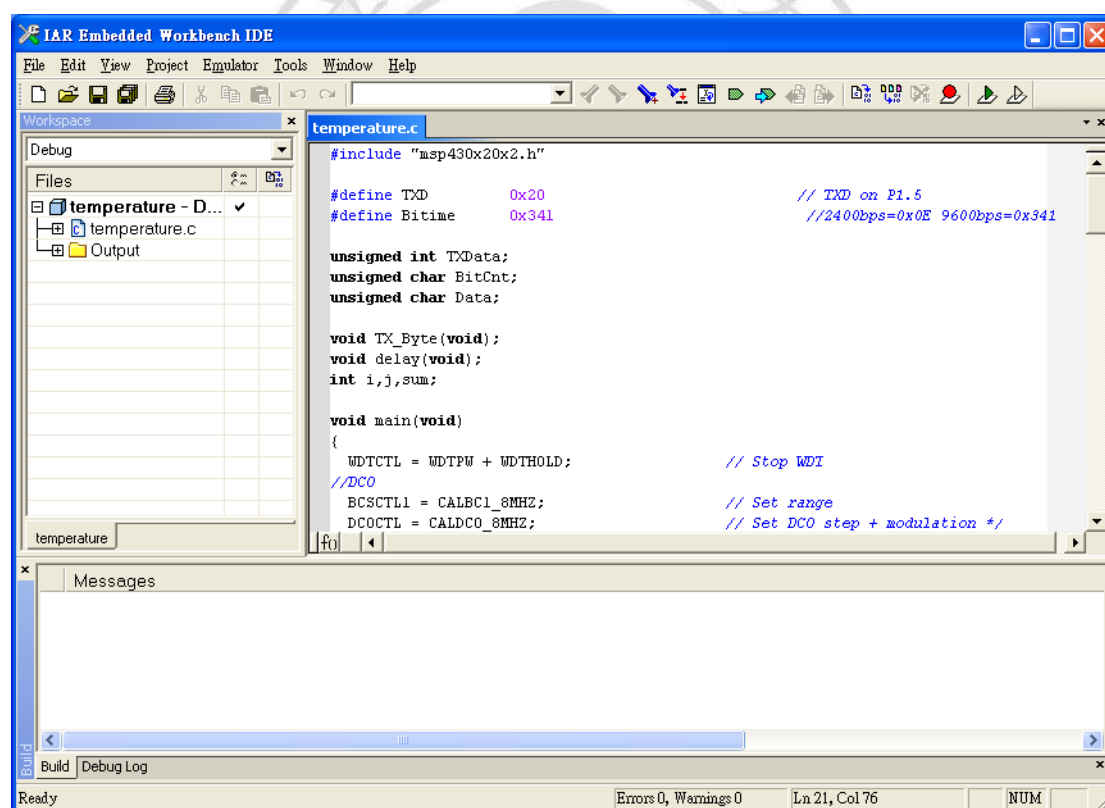


圖 3.8 IAR 開發環境

控制器內部模組如圖 3.9 所示，時脈採用 SMCLK，且圖中並無

UART 傳輸，故利用 Timer 來做軟體 UART 傳輸功能。

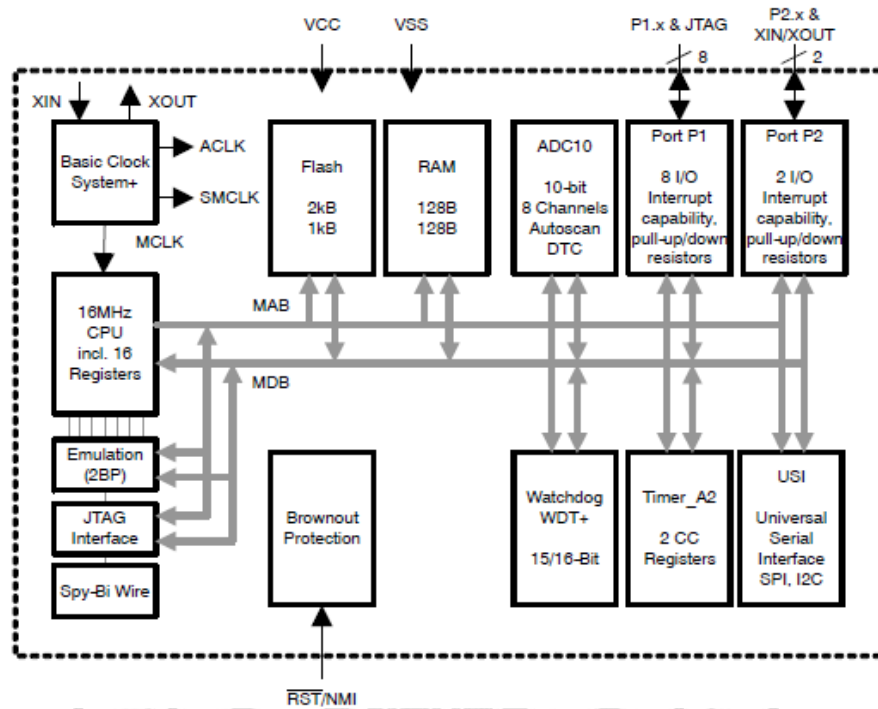


圖 3.9 MSP430 內部架構圖

圖 3.10 為工作電壓與頻率關係圖，本研究採用額定電壓為 3.3V，

選用相對應的石英晶體可使 Timer 誤差降低。

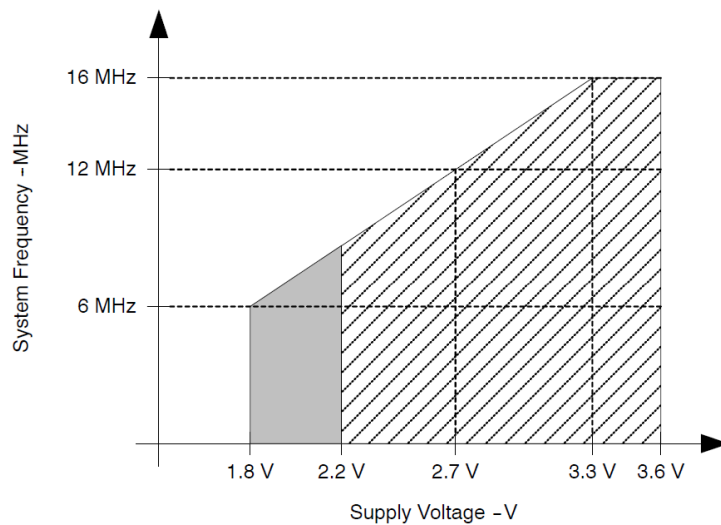


圖 3.10 工作電壓與頻率關係圖

3.3.1 Timer 設定

本研究使用軟體規劃 UART 輸出 Port，設定完 Timer 的相關暫存器，程式會持續執行 Timer，並使用 Timer_A 中斷令資料以 9600bps 從設定好的串列輸出腳送出，如圖 3.11 所示。

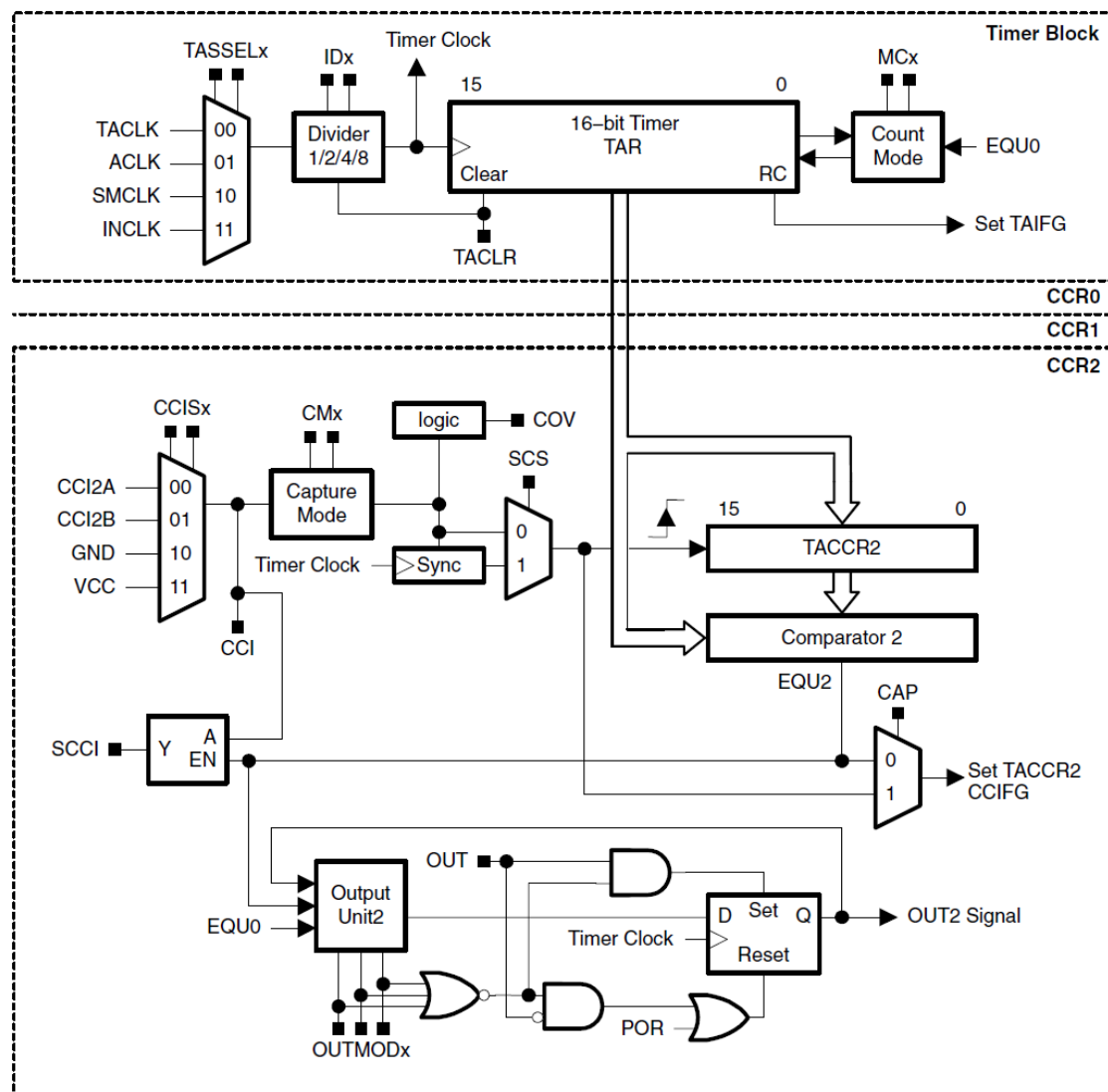


圖 3.11 MSP430 內部 Timer 架構圖

3.3.2 ADC 設定

如圖 3.12 所示，為 MSP430 內建的 10 位元 ADC 架構圖。本系統的額定電壓為 3V，解析度為 $3V/1024 \approx 2.93 \text{ mV/bit}$ ，但是配合 ZigBee 傳輸通訊格式修改為 $3V/255 \approx 11.76 \text{ mV/bit}$ 即可。

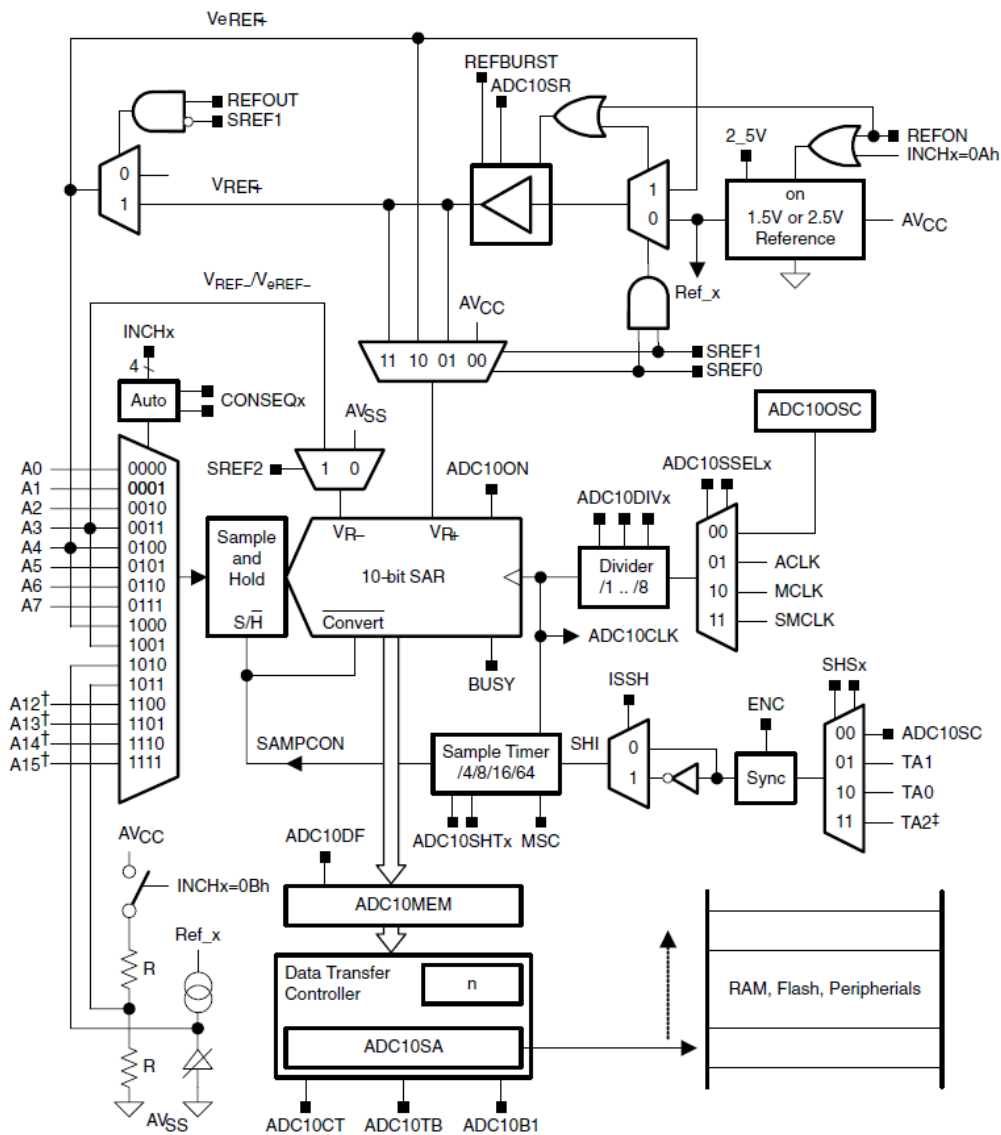


圖 3.12 MSP430 內部 ADC 架構圖

由於 MSP430 的 A/D 為獨立內建模組，因此，只需要在程式開始的時候設定好相關暫存器，讓 A1 執行讀取信號與數位化轉換，等

暫存器將數值轉換完成後，會產生中斷，使用者再由中斷副程式處理 A/D 轉換完的資料即可。

MSP430F2012 程式在撰寫時，其中 ADC10CTL0 是控制 ADC10 轉換的主要功能如圖 3.13 所示，系統所設定的功能包含 SREFx、ADC10SHTx、ADC10SR、REFOUT、REFBURST、ADC10SHTx、ADC10ON、ADC10IE。

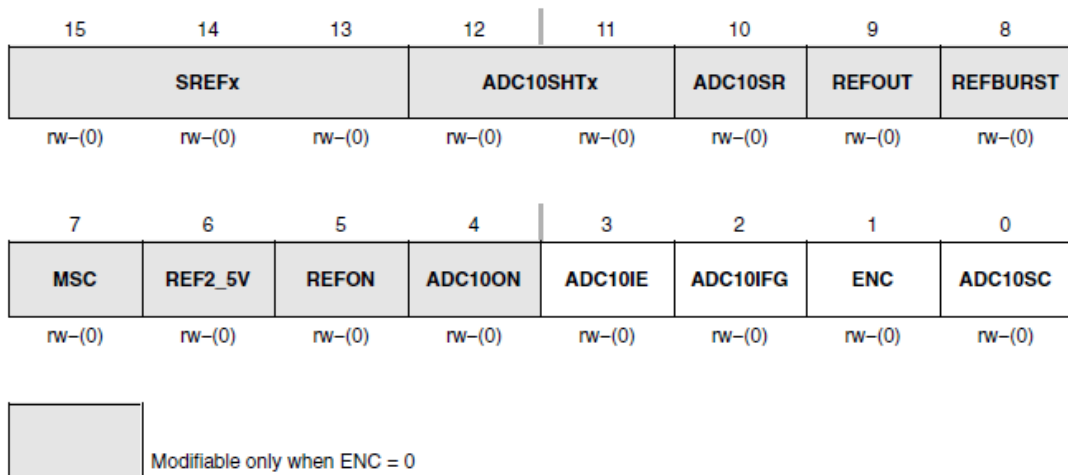


圖 3.13 ADC10CTL0(ADC10 控制暫存器 0)

SREFx	位元	參考電壓的選擇方式位元
	15-13	000 $V_{R+}=V_{CC}$ 和 $V_{R-}=V_{SS}$
		001 $V_{R+}=V_{REF+}$ 和 $V_{R-}=V_{SS}$
		010 $V_{R+}=V_{eREF+}$ 和 $V_{R-}=V_{SS}$
		011 $V_{R+}=Buffered V_{eREF+}$ 和 $V_{R-}=V_{SS}$

- 100 $V_{R+} = V_{CC}$ 和 $V_{R-} = V_{REF-} / Ve_{REF-}$
- 101 $V_{R+} = V_{REF+}$ 和 $V_{R-} = V_{REF-} / Ve_{REF-}$
- 110 $V_{R+} = Ve_{REF+}$ 和 $V_{R-} = V_{REF-} / Ve_{REF-}$
- 111 $V_{R+} = \text{Buffered } Ve_{REF+}$ 和 $V_{R-} = V_{REF-} / Ve_{REF-}$

系統設定 $ADC10CTL0 = SREF_2$ ， $V_{R+} = Ve_{REF+}$ 和 $V_{R-} = V_{SS}$ ，

V_{R+} 為外接引腳的 V_{CC} 參考電壓 3V， V_{R-} 為外接引腳的 V_{SS} 在本文中為 GND。

ADC10SHTx	位元	ADC10 取樣與延遲時間位元
	12-11	00 4 x ADC10CLKs
		01 8 x ADC10CLKs
		10 16 x ADC10CLKs
		11 64 x ADC10CLKs

系統設定 $ADC10CTL0 = ADC10SHT_2$ ，為 8 x ADC10CLKs。

ADC10ON	位元 4	啟動 ADC10 位元
	0	ADC10 啟動
	1	ADC10 關閉

系統設定 $ADC10CTL0 = ADC10ON$ ，為 ADC10 啟動。

ADC10IE 位元 3 ADC10 中斷致能位元

0 停止中斷致能

1 啟動中斷致能

系統設定 $ADC10CTL0 = ADC10IE$ ，為啟動 ADC10 中斷致能。

除了本系統所提到的功能設定外，還有其他常用的功能，例如：
內部參考電壓 REFON、REF_2.5V 為啟動 MSP430F2012 內部的參考電壓，其 0 為 1.5V、其 1 為 2.5V，使用內部的參考電壓可以獲得更高的解析度。以參考電壓 1.5V 與 8 位元傳輸為例，解析度為 $1.5V / 255 = 0.00588235$ ，只要將 ADC10 所需要的功能設定好啟動致能位元 MSP430F2012 會自行 A/D 轉換，轉換完成後便可以從 ADC10MEM 暫存器讀取轉換後的資料。

3.3.3 流程圖

由於電路設定 $10\text{mV}/^{\circ}\text{C}$ 的靈敏度會使溫度在 300°C 的電壓值為 3V ，為了避免飽和，輸入信號的值必需經過衰減再 A/D 轉換，最後由設定好的軟體 UART 傳輸送至電腦端做補償才會得到實際的測量值，如圖 3.14 所示。

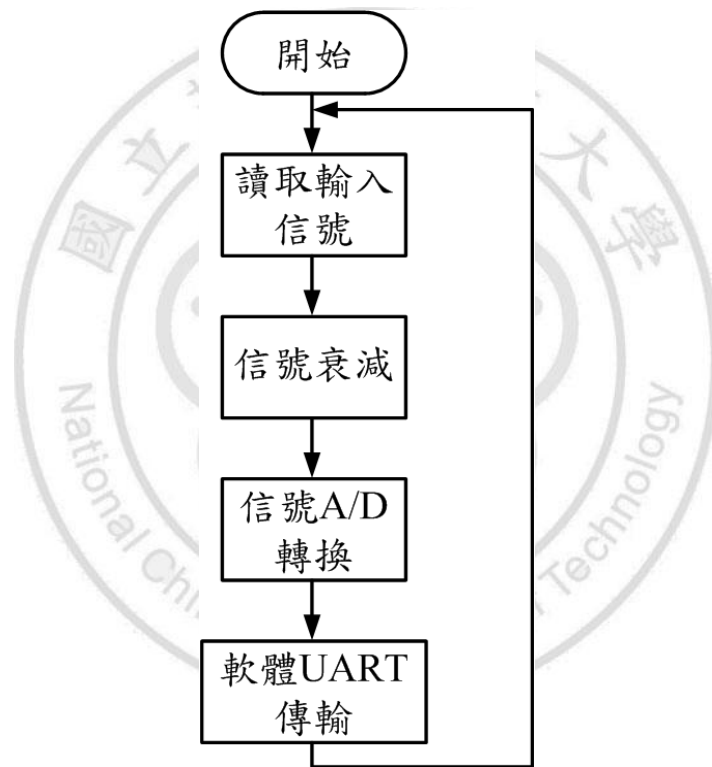


圖 3.14 MSP430 程式流程圖

3.4 ZigBee 無線傳輸

ZigBee 一詞源於蜜蜂(Bee)，蜜蜂看似隨意跳的字形舞，實際是在分享有花與蜂蜜的地方信息給同伴，而 ZigBee 正式採用此種通訊方式。ZigBee 自 1998 年開始發展，並於 2001 年向 IEEE 提案，將 ZigBee 技術納入電氣電子工程師協會(Institute of Electrical and Electronics Engineers, IEEE) IEEE 802.15.4 標準規範中，Zigbee 無線感測網路在規範中就被定義是必須低成本(low cost)、低耗電(low power consumption)、小體積(low volume)、易佈建(low installation cost)、可程式化(programmable)及可動態組成(dynamic building)等優良特性目標，ZigBee 主要是由 IEEE 802.15.4 小組與 ZigBee Alliance 組織，分別制訂硬體與軟體標準。ZigBee 的網路拓樸包含星狀、樹狀及網狀方式，本研究所套用的是星狀網路拓樸，如圖 3.15 所示。

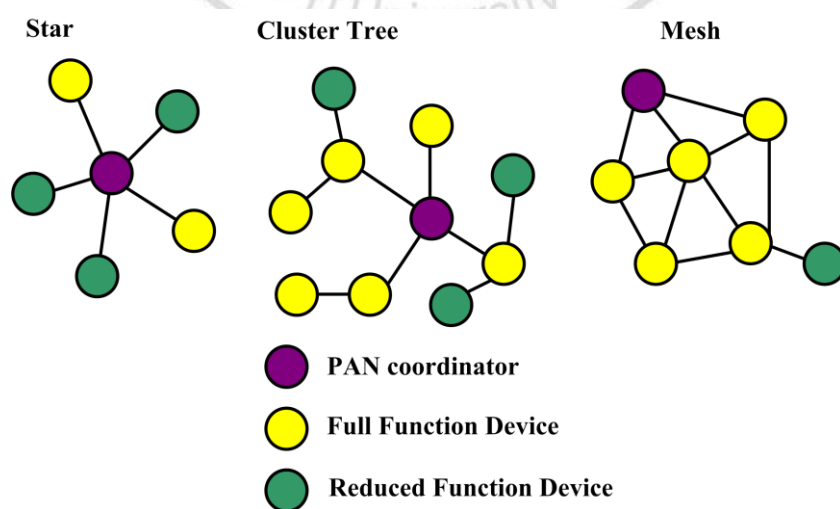


圖 3.15 ZigBee 工作網路拓樸方式

ZigBee 網路的功能運作，與無線感測裝置之佈署以及裝置間自我組建(Self-organization)之網路連結(Connection)有極密切之關係，適合不同大小規模的消費性電子、電腦週邊、工業控制及家用自動化設備...等各方面應用，未來在低速率家庭網路、工業化自動控制及消費性產品將有很大的發展潛力。



3.4.1 無線傳輸模組

本研究使用 Digi 的 ZigBee 無線傳輸模組 XBee。其模組有以下

特點：

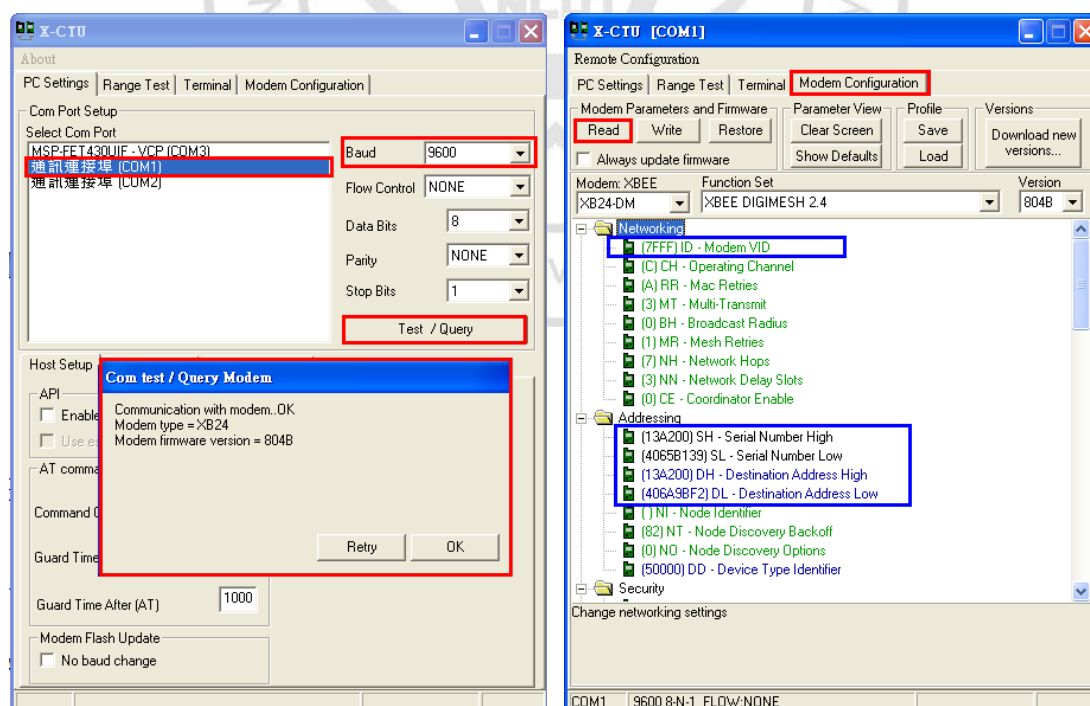
- ◆ 室外無障礙距離：最遠可達 1600 公尺
- ◆ 室內/室外範圍：最遠可到 90 公尺
- ◆ RF 資料傳輸速率：250kbps
- ◆ 發送峰值電流：250mA
- ◆ 睡眠模式電流：小於 50 μ A
- ◆ 接收器敏感度：-100dBm (1%的封包錯誤率)
- ◆ 接收電流：55mA
- ◆ 鮑率：1200bps~230400bps

XBee 無線傳輸模組操作上很簡單，只需要注意傳輸信號需經過 A/D 轉換與相同鮑率的設定便可以使用無線傳輸，接收端的 XBee 模組，其信號也必須經過 MAX232 電路的處理才能輸入個人電腦。

3.4.2 操作方式

Digi 公司提供免費提供 X-CTU 軟體，其用於 XBee 的功能設定，將 XBee 接上電腦 COM 端後，開啟 X-CTU 選擇鮑率以及所連接的 COM or USB，按下測試可確認是否有連線成功，如圖 3(a)所示。

圖 3(b)切換到 Modem Configuration 的索引標籤按下 Read 可以讀取 XBee 內部所有資訊，主要資訊設定在於藍色框線的部分，ID-Modem VID 相當於一個房間號碼，在同一房間中才能互相溝通接收資訊；SH、SL 是各個 XBee 自己的編號 ID，不可修改；DH、DL 則是設定工作模式，設定成欲連線目標的 SH、SL，再按下 Write 即可，如圖 3.16 為點對點模式為本研究中所套用的模式。



(a) 讀取 XBee 狀態

(b) 工作模式設定

圖 3.16 XBee 設定

3.4.3 工作模式

本研究使用 MSP430F2012 做為信號處理的核心，透過軟體 UART 的方式並藉由 XBee 進行資料傳送。傳送的方式，支援點對點 (Point-to-Point) 以及點對多 (Point-to-Multipoint)。若是點對點傳送，就只限於兩點之間的接收，若是廣播模式，則在同一個 ID-Modem VID 網路當中都可以收到各個模組的資料。如圖 3.17 所示，XBee 除了待機狀態外，還有四種模式可以互相切換，其中只有 Command 跟 Sleep 是需要特別設定才會進入此兩種模式，平常當資料收發時會自動進入 TX(Transmit Mode) 與 RX(Receive Mode) 模式。

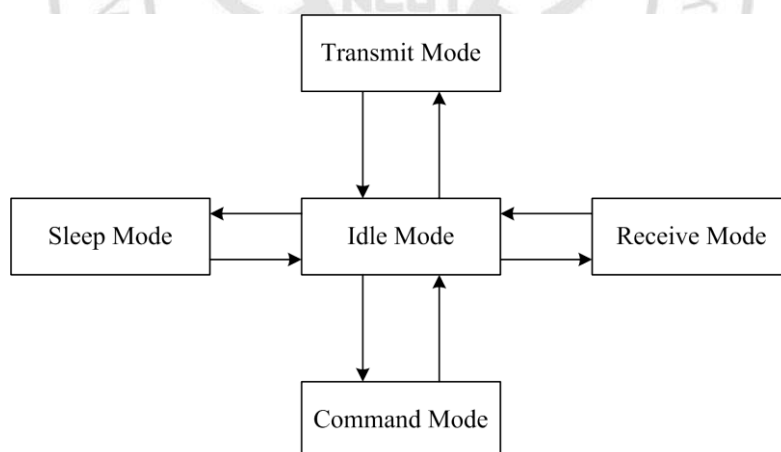


圖 3.17 XBee 工作模式

各個模式基本定義：

一、點對點模式：XBee A 與 C 的 DH、DL 個別輸入 XBee B 與 D 的 SH、SL，如圖 3.18 所示。

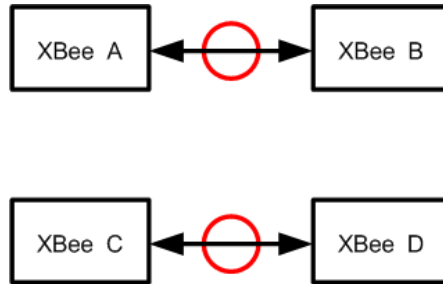


圖 3.18 點對點模式

二、點對多模式：XBee A 的 DH、DL 分別輸入 0 和 FFFF，XBee B 與 C 的 DH、DL 輸入模組 A 的 SH、SL，如圖 3.19 所示。

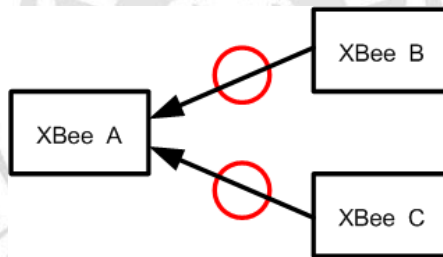


圖 3.19 點對多模式

三、廣播模式：XBee A、B、C、D 的 DH、DL 分別輸入 0 和 FFFF，如圖 3.20 所示。

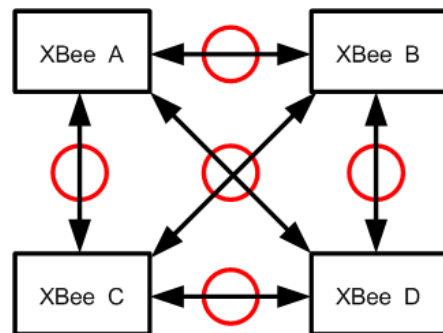
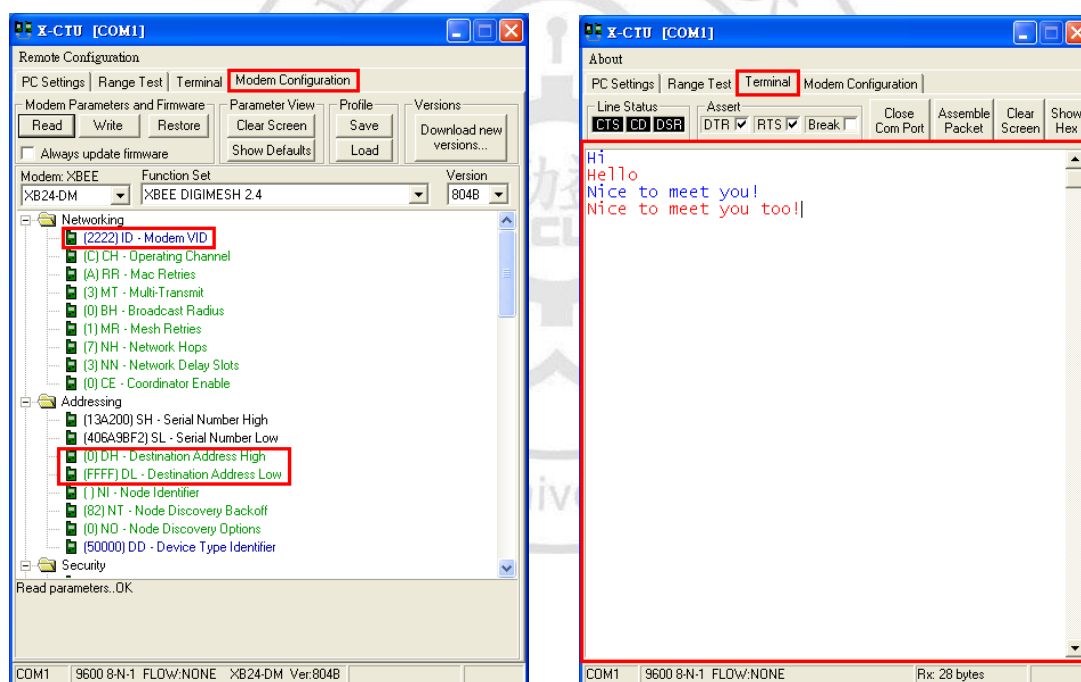


圖 3.20 廣播模式

3.4.4 傳輸簡易測試

為了方便測試 XBee 模組都設定為廣播模式如圖 3.21(a)所示，其中 ID-Modem VID 設定為 2222，DH 為 0，DL 為 FFFF，X-CTU 有內建傳輸測試功能，切換到 Terminal 的索引目錄，紅框內鍵入聊天訊息，就可以透過 XBee 與另一台電腦進行聊天如圖 3.21(b)所示，其中藍色字為使用者 A 傳輸的訊息，紅色字為使用者 B 的傳輸訊息。



(a) 廣播模式設定

(b) XBee 傳輸測試

圖 3.21 XBee 傳輸簡易測試

實際的 XBee 傳輸必須配合一個微控制器進行，所以為了更了解實際的傳輸情形，本研究使用內建有 UART 傳輸的 89S51 微控制器來做個簡易的傳輸練習，因為 89S51 微控制器支援 UART 傳輸，只需下達發射資料的指令配合迴圈的功用便能達到練習 XBee 傳輸的目的，將 89S51 的發射接腳 TX 接至 XBee 的讀取接腳 RX，簡易的測試電路就算完成，最後利用個人電腦內建的終端機接收資料即可，如圖 3.22 所示。



(a)傳輸 Port 選擇

(b)傳輸詳細設定

圖 3.22 終端機接收設定

此外終端機也可以直接用來當作接收端使用，可以幫助確認 XBee 是否有正常工作，如圖 3.23 所示為終端機接收資料圖，圖 3.24 為接收端 XBee 模組實體圖。

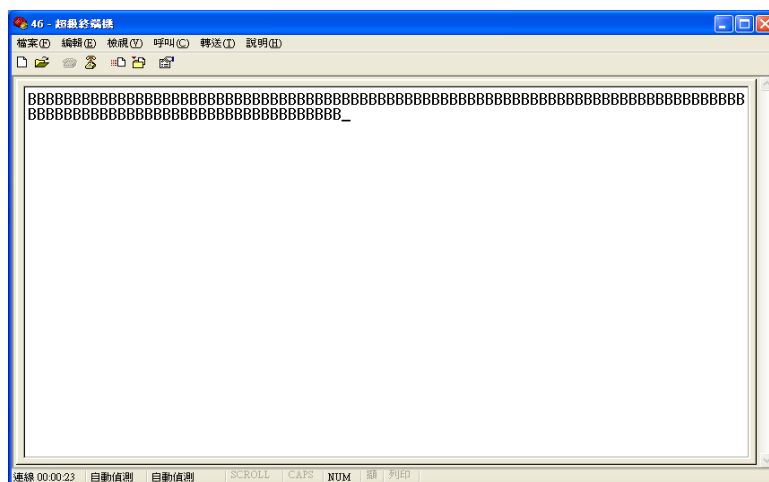


圖 3.23 終端機接收資料圖

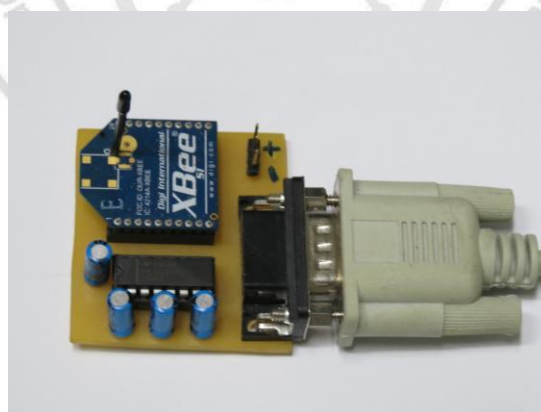


圖 3.24 接收端 XBee 模組實體圖

第四章、資料擷取程式設計

4.1 LabVIEW 簡介

熱電堆溫度監測系統之人機介面是使用 LabVIEW 軟體開發。LabVIEW 是由 National Instrument 公司所發展出的一種圖形化程式介面的發展環境，又稱為 G 語言，其使用圖形式的物件函數取代文字編輯的程式設計方式，讓使用者更清楚了解程式結構的涵義。National Instrument 公司推出一種資料擷取的介面，可以透過這介面讀取電腦外部的信號，只需選定資料擷取介面 I/O 接腳，軟體會自動告知使用者如何接線讀取外部信號，透過 DAQ Assistant 可以設定取樣率，可以依據使用者規劃設定。

LabVIEW 提供了使用者發展測試、量測、控制及自動化系統的完整功能性，其中包含了資料結構、流向控制、迴圈架構與事件管理等一般程式語言的功能。

4.2 資料擷取 DAQ 流程圖

一開始使用 LabVIEW 開發顯示介面時先利用擷取介面卡 NI DAQPad-6015，進行資料的儲存與顯示，未來在將 DAQ Assistant 修改成 RS232 接收可以減少 LabVIEW 程式撰寫的時間。在設定 DAQ Assistant 時需要注意信號輸入的範圍、資料擷取模式與取樣速度，如圖 4.1 所示。

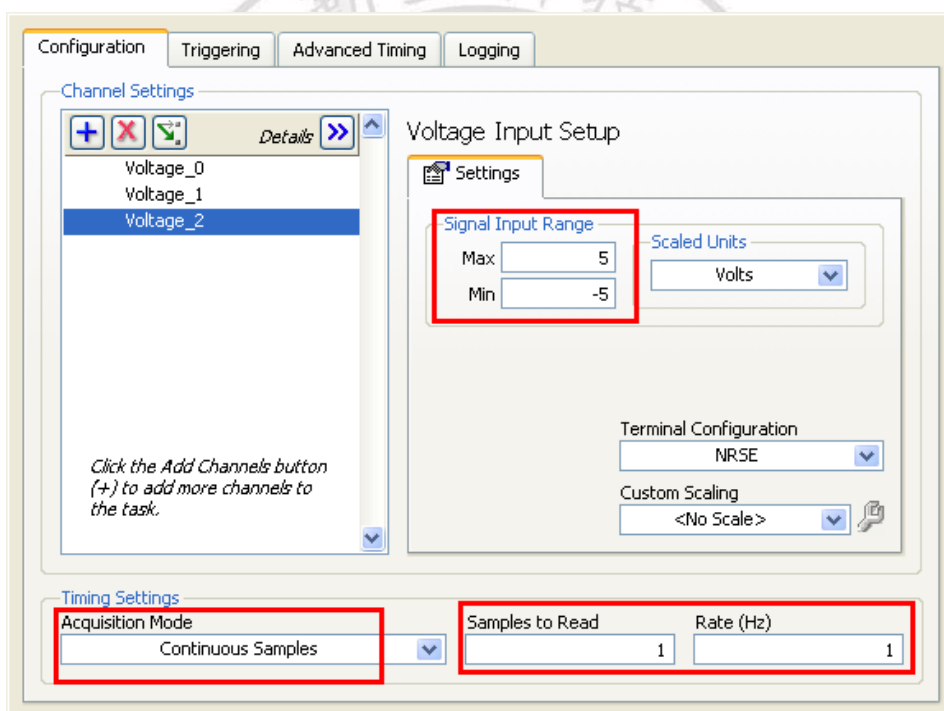


圖 4.1 DAQ Assistant 設定

圖 4.2 為使用 DAQ Assistant 分別擷取 thermistor、thermopile、量測溫度信號之流程圖，使用陣列儲存擷取資料與波形顯示，縮減開發無線傳輸介面時撰寫程式的時間。

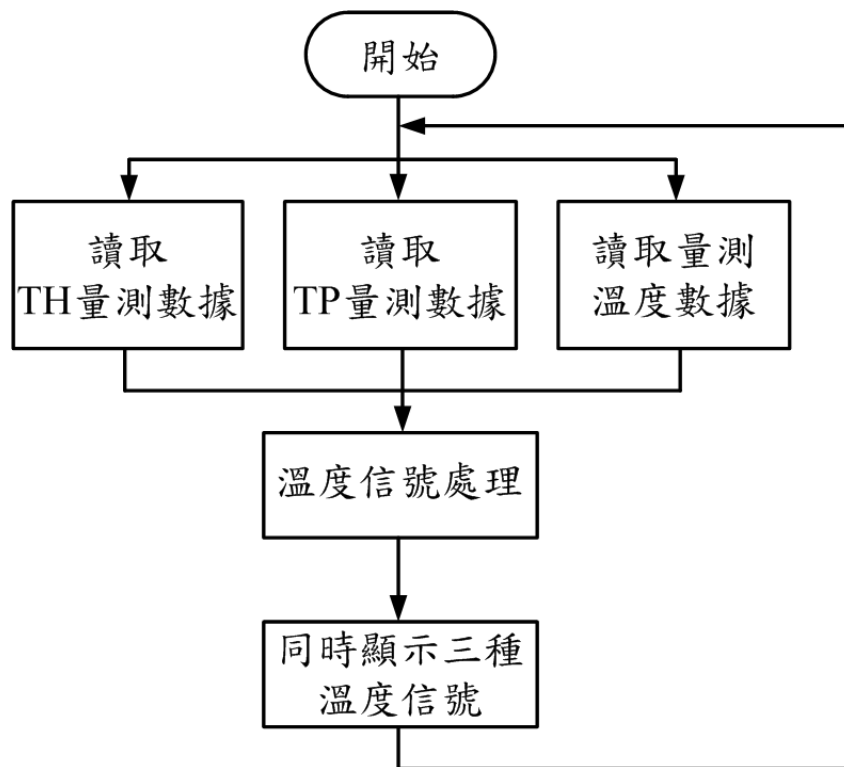


圖 4.2 DAQ 資料擷取之 LabVIEW 流程圖

4.2.1 RS232 接收測試

串列傳輸為一種常用的資料傳送方式，這種通訊方式常用於電腦與電腦之間或與周邊裝置之間，藉由電腦內建的串列埠 RS232 來傳送資訊。使用 LabVIEW 軟體內的串列函式讀取資料，設定好電腦端 RS232 COM Port、鮑率和傳送資料位元就可以接收資料，如圖 4.3 所示。

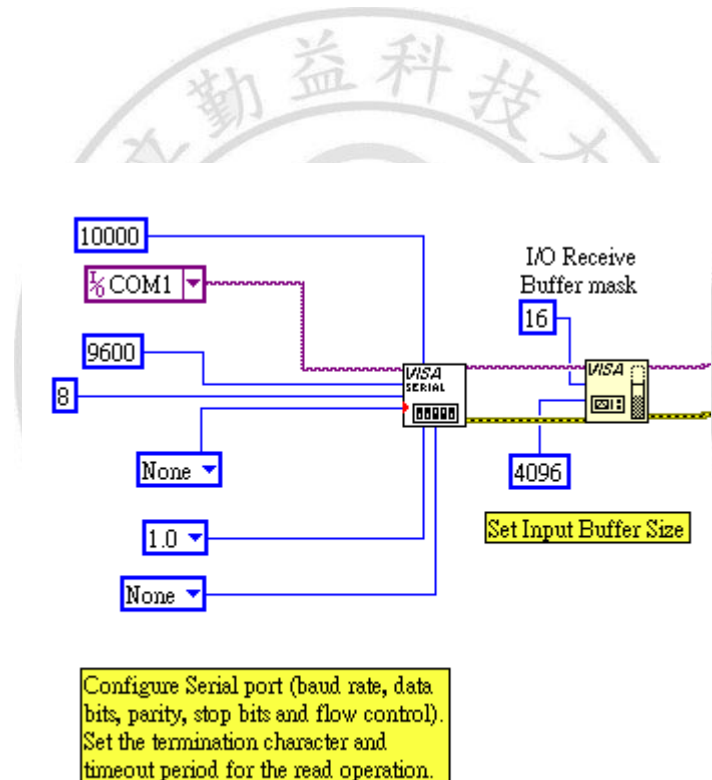


圖 4.3 LabVIEW 函式之 RS232 設定圖

4.3 溫度測量流程圖

LabVIEW 軟體針對溫度測量的設計流程圖如圖 4.4 所示。

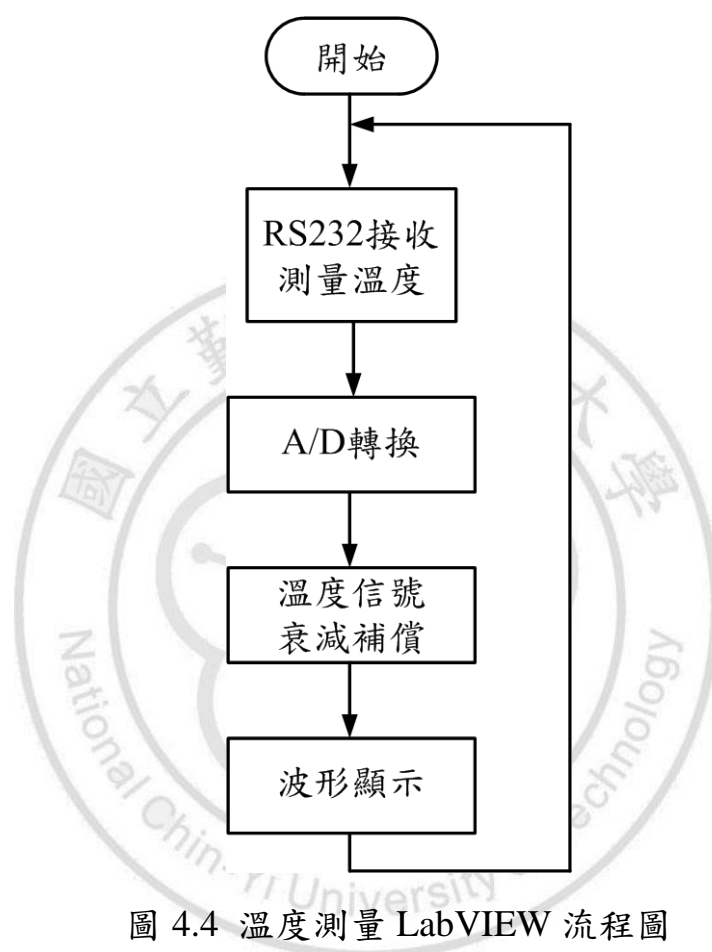


圖 4.4 溫度測量 LabVIEW 流程圖

LabVIEW 透過 RS232 讀取點對點模式的 XBee 接收端所接收的數位信號，經過衰減補償後將其乘上每位元的解析度即可轉換回類比信號，最後顯示波形。

第五章、實驗平台與結果

5.1 實驗平台設置

本研究使用黑體爐 Gemini R model 976 來模擬黑體輻射熱源如圖 5.1 所示，其輻射溫度範圍為 30°C 到 550°C，65mm 的直徑腔，深度為 160mm，發射率大於 0.995，在此要注意感測元件的 D:S 比與 Gemini R 的孔徑大小以確保感測元件有對準黑體爐的輻射腔內，本研究設定的溫度範圍為 0°C ~ 300°C，使用 Gemini R 加熱至 300°C 需要 16 分鐘，散熱所需的時間需要更長，所以散熱的溫度曲線量測不列入本研究中。



圖 5.1 Gemini R 實體圖

本研究使用 Protel 99 SE 作為 PCB Layout 軟體，只要在 Protel 99 SE 中將電路圖畫好，如圖 5.2 所示，就可以將其轉換成 PCB 圖，PCB Layout 時採用雙層佈線規劃，完成圖如圖 5.3 所示，其藍線為底層佈線，紅線則為頂層佈線。

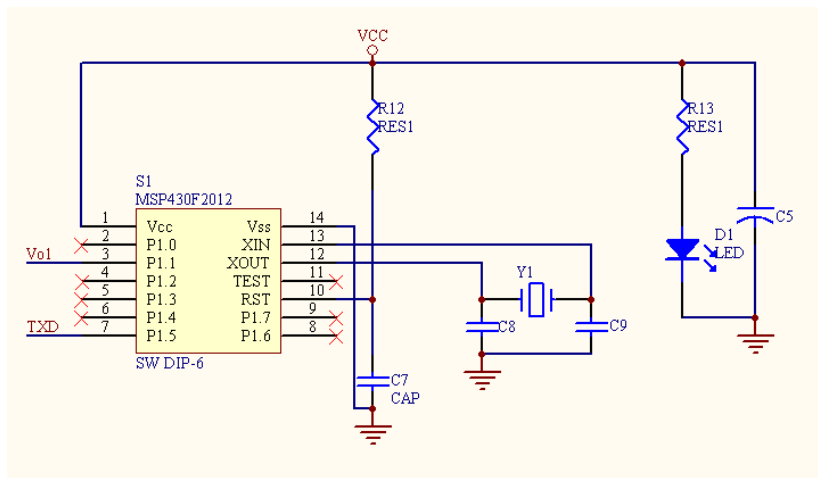


圖 5.2 MSP430F2012 信號處理電路

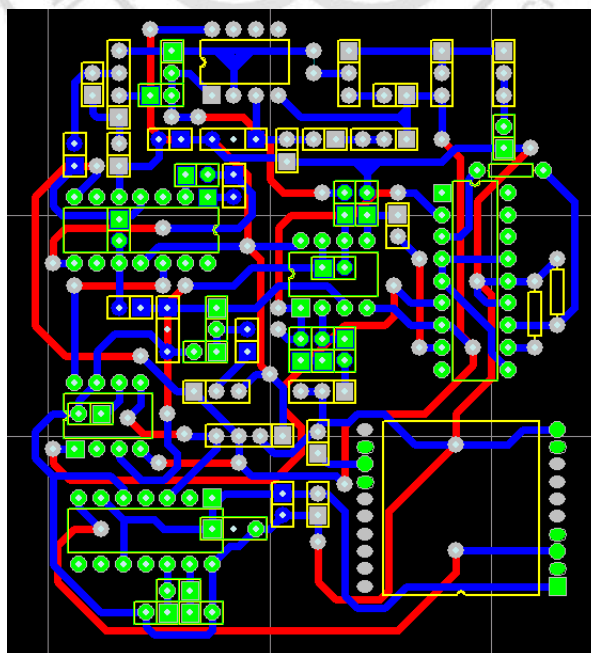


圖 5.3 PCB Layout 圖

電路採用 SMD 元件來縮小 PCB Layout 的面積，如圖 5.4 所示。

實體圖如圖 5.5 所示。

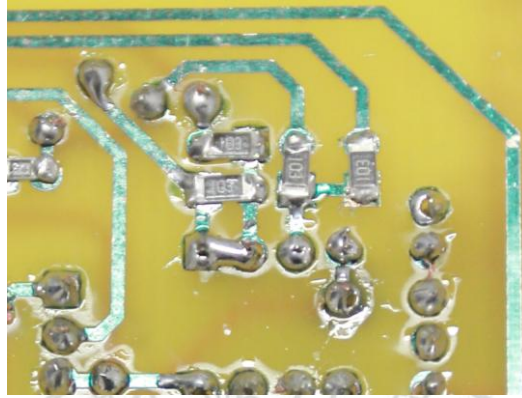


圖 5.4 SMD 元件

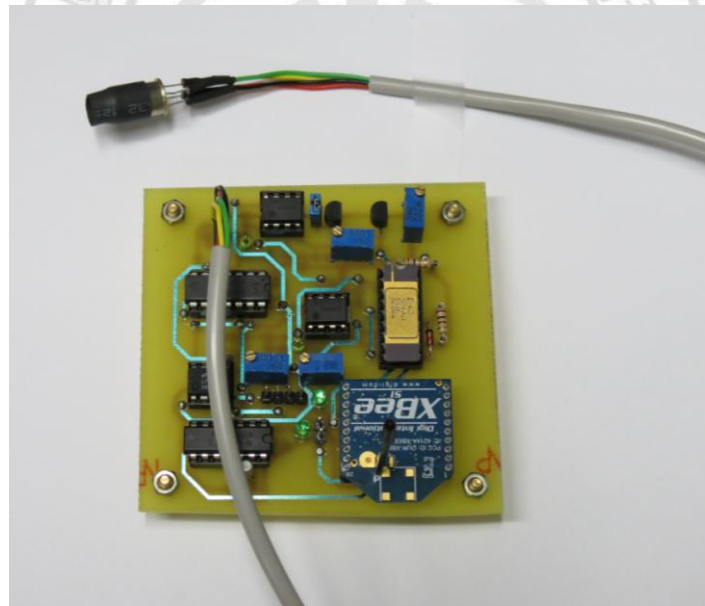


圖 5.5 電路實體圖

5.1.1 溫度校準

如圖 5.6 所示，使用一個近理想黑體輻射源進行反覆的校準電路，感測元件被放置距離黑體輻射源 20cm 處。校準過程中，強調以下三個關鍵點：

1. 如圖 3.4 所示，調整 VR_2 獲得一個恆定流源 I_0 。
2. 如圖 3.4 所示，調整 VR_1 消除偏移電壓，當溫度在 0°C 時，使得其值為 0。
3. 選擇符合溫度範圍的熱電堆放大增益 gain A。



圖 5.6 溫度校準平台設置

5.2 DAQ 資料擷取結果

為了測試系統電路的性能，設定在 100°C 、 200°C 和 300°C 的黑體輻射溫度，感測元件被放置在距離黑體爐 40cm 處，使用 DAQ 資料擷取傳送至個人電腦再由 LabVIEW 軟體顯示，其取樣率為每秒取樣一次。圖 5.7、5.8、5.9 分別顯示三種曲線在設定好的溫度下所測量的結果，其中綠色曲線為環境的溫度由 TH 所測得，藍色曲線為輻射溫度由 TP 偵測，最後白色曲線為輻射溫度與環境溫度的總和。圖 5.7 為黑體爐設定為 100°C 的量測結果，由於靈敏度 $10\text{mV}/^{\circ}\text{C}$ 可得知平均誤差為 $\pm 2^{\circ}\text{C}$ ，但是最大誤差為 8°C 。

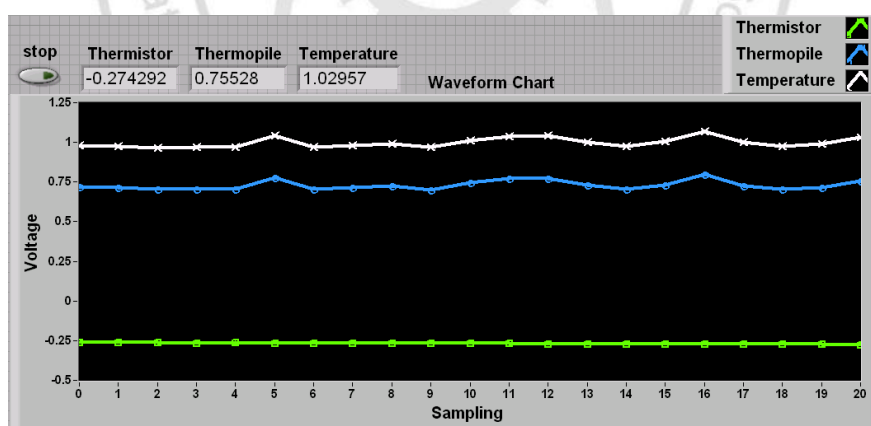


圖 5.7 黑體爐設定 100°C 量測結果

圖 5.8 為黑體爐設定為 200°C 的量測結果，其平均誤差為 $\pm 2^\circ\text{C}$ ，但是最高的誤差為 15°C 以上，圖 5.9 為黑體爐設定為 300°C 的量測結果，平均誤差約為 $\pm 3^\circ\text{C}$ ，但是最大誤差為 18°C。

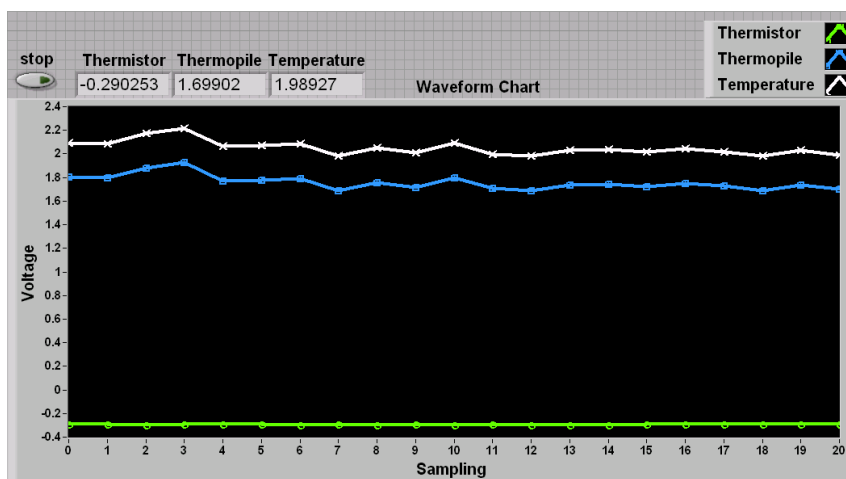


圖 5.8 黑體爐設定 200°C 量測結果

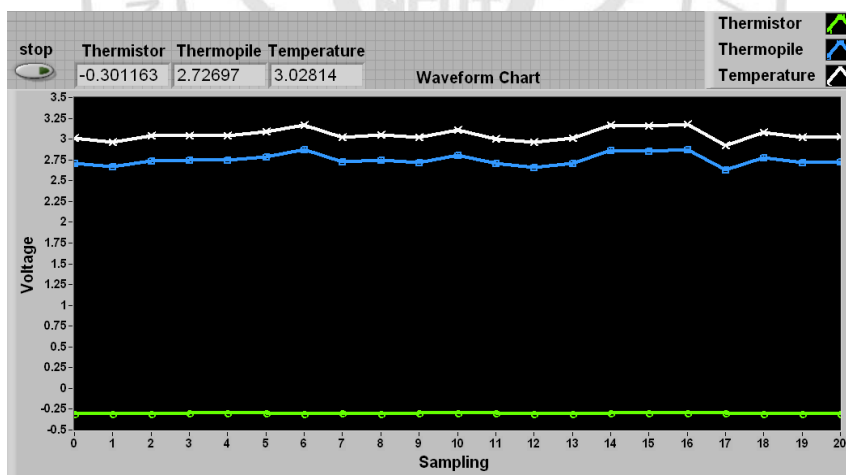


圖 5.9 黑體爐設定 300°C 量測結果

為了測試本文的系統性能，做了一項從室溫加熱至 300°C 的實驗，如圖 5.10 所示，白色曲線顯示黑體爐溫度逐漸升高，在第 75 次取樣點後接近 300°C 。

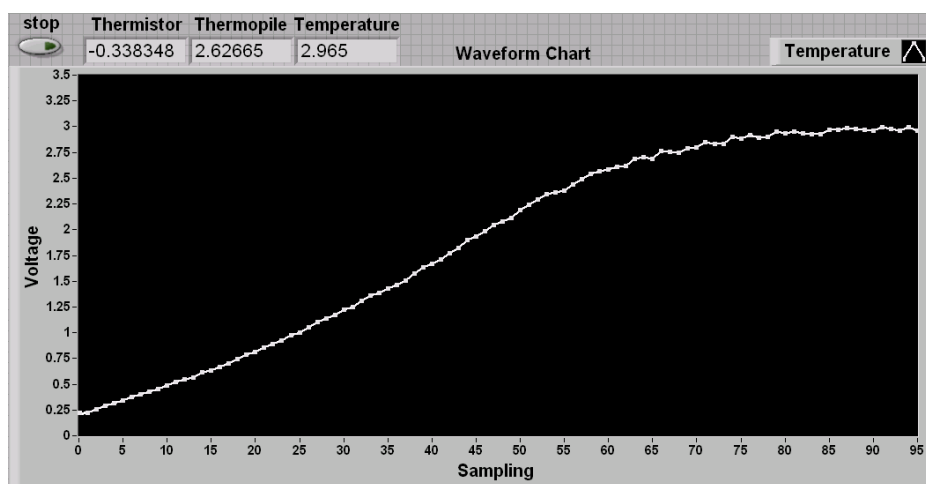


圖 5.10 室溫加熱至 300°C 連續測量結果

根據實驗結果顯示平均誤差可以限制在 $\pm 3^{\circ}\text{C}$ 以下，但溫度測量系統有最大瞬間偏差的缺點，顯然的我們需要一個取平均值的行為來得到更精確的值。為了修正前述誤差，TP 的前置放大加入可以穩定工作的斬波器(Chopper)，以解決飄移(drift)的問題。

5.3 實驗結果

如圖 5.11 所示為本系統方塊圖，感測元件所測得的電壓經由濾波、放大、與線性化電路後，使用 MSP430F2012 微控制器讀取信號並進行數位信號轉換，轉換過後的數位信號將由設定好的軟體 UART 傳輸給 XBee 模組發射出去，接收端的 XBee 模組接收到資料後經由 RS232 傳輸到個人電腦在使用軟體 LabVIEW 作為一個人機界面的顯示，以 9600 鮑率傳送 10 位元資料(8 位元加起始位元與停止位元)配合 LabVIEW 軟體設計的 10 筆信號取一次平均值，可以得到每秒傳送 96 筆信號的取樣率。

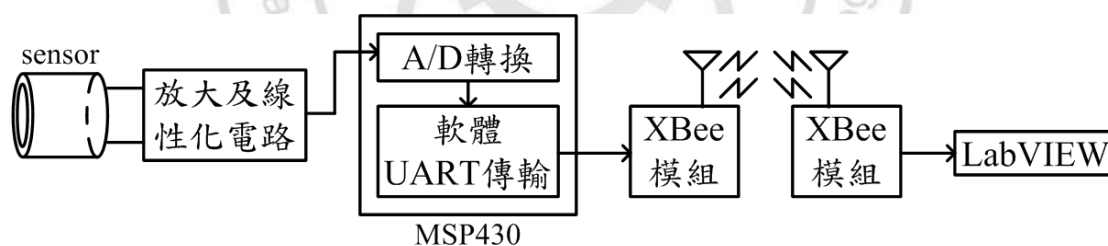


圖 5.11 系統方塊圖

圖 5.12 至 5.17 為 50°C 到 300°C 各別取樣 51 次(圖中陣列之值為 0~49 次取樣結果，第 50 次則顯示在 Temperature)的量測結果，其中各點定溫量測的數據整理在表 5.1，這些實驗數據有助於本文分析溫度曲線的誤差以及了解輻射溫度的特性曲線之線性化。

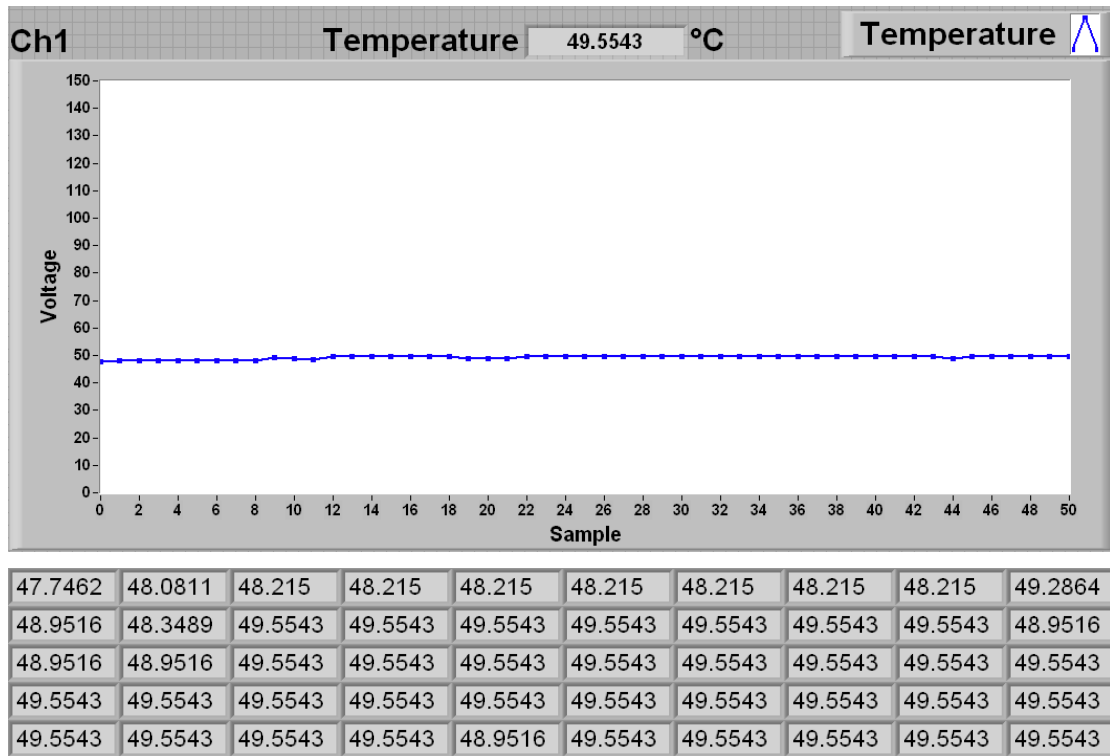


圖 5.12 50°C 定溫量測

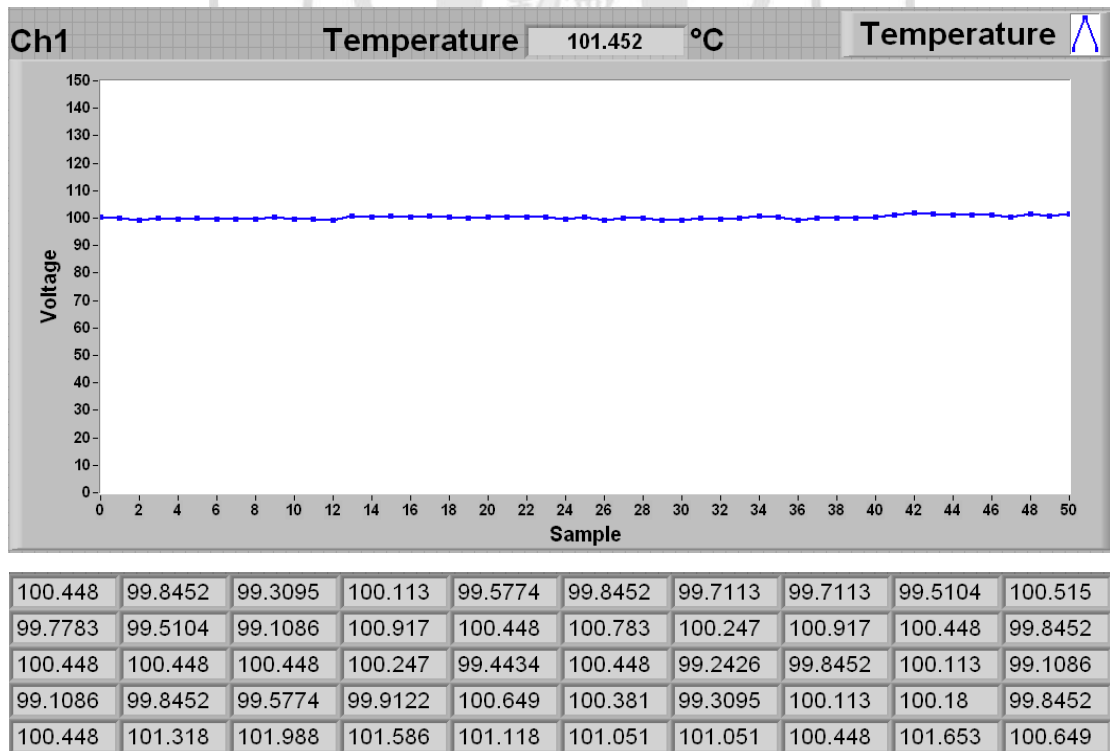


圖 5.13 100°C 定溫量測

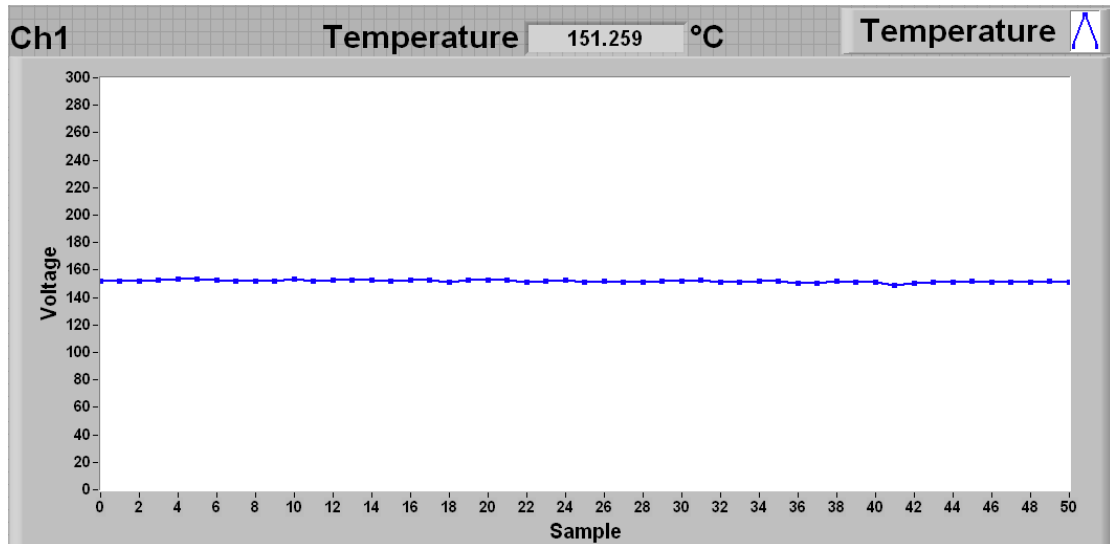


圖 5.14 150°C 定溫量測

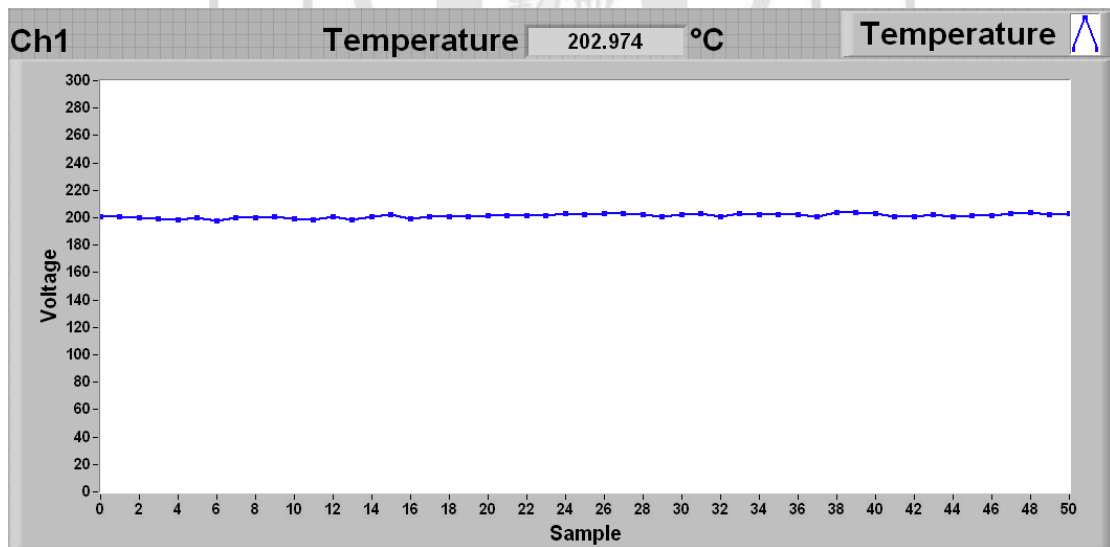


圖 5.15 200°C 定溫量測

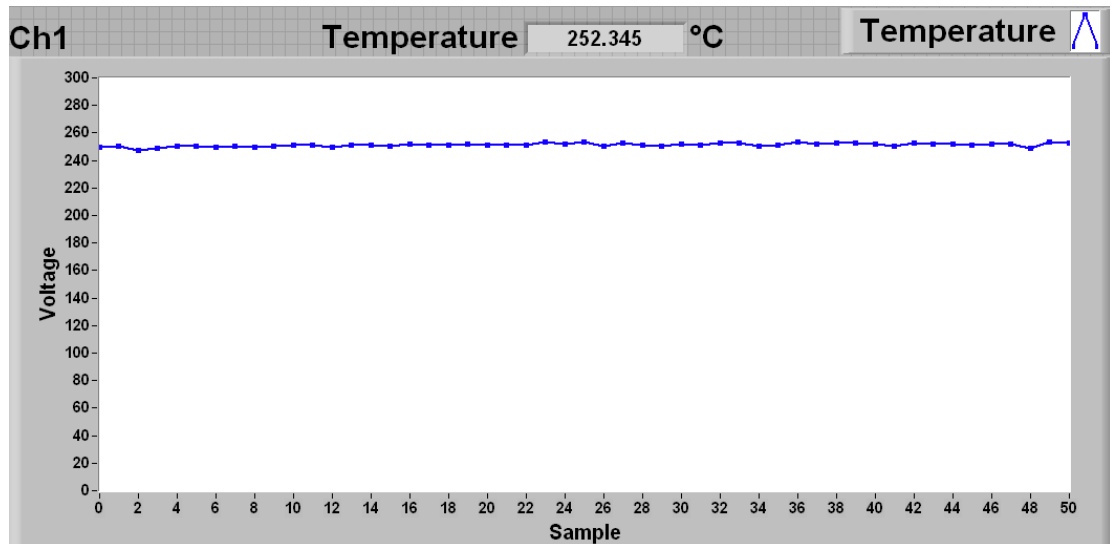


圖 5.16 250°C 定溫量測

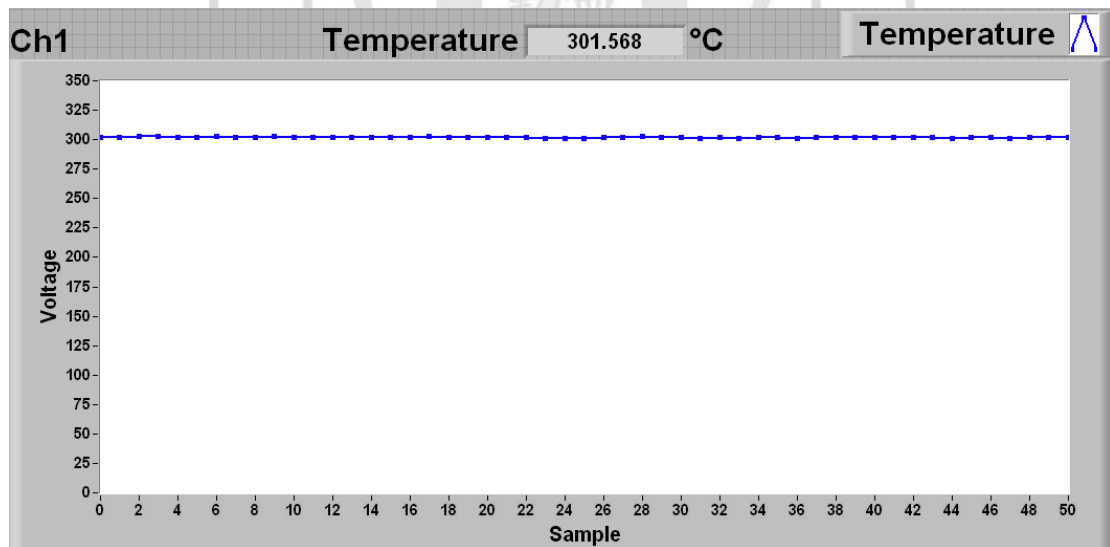


圖 5.17 300°C 定溫量測

表 5.1 溫度測量數據

溫度 取樣序	50 °C	100 °C	150 °C	200 °C	250 °C	300 °C
0	47.7	100.4	152.2	200.3	249.7	302.1
1	48.0	99.8	152.1	200.4	250.2	302.0
2	48.2	99.3	151.7	199.7	247.5	302.8
3	48.2	100.1	152.8	199.2	249.0	302.8
4	48.2	99.5	153.2	198.2	250.2	301.8
5	48.2	99.8	153.1	199.6	250.7	301.5
6	48.2	99.7	152.9	197.9	249.5	302.8
7	48.2	99.7	152.1	199.7	250.5	302.2
8	48.2	99.5	151.9	199.9	250.0	302.2
9	49.2	100.5	151.6	200.5	250.5	302.8
10	48.9	99.7	153.6	199.3	251.1	301.7
11	48.3	99.5	151.9	198.2	251.2	302.2
12	49.5	99.1	152.6	201.0	249.5	301.8
13	49.5	100.9	152.8	198.2	251.1	301.5
14	49.5	100.4	152.9	200.3	251.2	301.6
15	49.5	100.7	151.8	202.2	250.6	302.0
16	49.5	100.2	152.6	198.9	251.6	301.8
17	49.5	100.9	153.0	200.6	251.1	302.8
18	49.5	100.4	151.3	201.1	251.2	302.2
19	48.9	99.8	152.9	200.6	252.0	301.5
20	48.9	100.4	152.6	201.1	251.1	301.5
21	48.9	100.4	152.4	201.8	251.2	301.9
22	49.5	100.4	150.8	201.8	251.2	301.8
23	49.5	100.2	151.8	201.8	253.7	300.9
24	49.5	99.4	152.8	203.0	252.0	301.3
25	49.5	100.4	151.5	202.5	253.3	301.2
26	49.5	99.2	151.9	202.8	250.5	302.0
27	49.5	99.8	150.8	203.3	252.4	301.7
28	49.5	100.1	151.2	202.1	251.3	302.8
29	49.5	99.1	152.1	200.6	250.6	301.8
30	49.5	99.1	151.8	202.0	251.8	301.5

31	49.5	99.8	152.4	203.0	251.2	300.9
32	49.5	99.5	151.2	201.0	252.7	301.6
33	49.5	99.9	151.5	202.9	252.9	301.2
34	49.5	100.6	151.7	202.2	250.7	301.7
35	49.5	100.3	151.6	202.0	250.9	301.7
36	49.5	99.3	150.4	202.1	253.3	300.9
37	49.5	100.1	150.3	201.0	252.1	302.1
38	49.5	100.1	151.6	203.7	253.0	302.1
39	49.5	99.8	150.9	204.0	252.9	301.5
40	49.5	100.4	151.0	203.0	252.1	301.5
41	49.5	101.3	149.1	200.5	250.6	301.5
42	49.5	101.9	150.3	201.1	252.5	301.8
43	49.5	101.5	151.1	202.2	252.1	301.8
44	48.9	101.1	151.1	200.9	251.7	300.9
45	49.5	101.0	151.6	201.3	251.3	302.2
46	49.5	101.0	150.8	201.8	252.2	302.2
47	49.5	100.4	151.2	202.9	252.2	301.3
48	49.5	101.6	150.7	203.5	248.7	302.0
49	49.5	100.6	151.7	202.4	253.2	301.6
50	49.5	101.4	151.2	202.9	252.3	301.5
平均溫度	49.1 °C	100.1 °C	151.7 °C	201.2 °C	251.2 °C	301.8 °C
最大誤差	-2.3 °C	1.4 °C	3.6 °C	4 °C	3.3 °C	2.8 °C
平均誤差	0.9 °C	0.1 °C	1.7 °C	1.2 °C	1.2 °C	1.8 °C
誤差率	1.8%	0.1%	1.13%	0.6%	0.48%	0.6%

根據表 5.1 所示，即使更改為無線傳輸的方式，該系統的平均誤差依然維持 $\pm 3^{\circ}\text{C}$ 以內，其結果顯示本系統的平均測量誤差已達到所預期的範圍內。

如圖 5.18 所示是使用無線傳輸的方式所測得的結果，由室溫持續加熱至 300°C 的連續測量結果經過 A/D 轉換、XBee 無線傳輸後由電腦端的 LabVIEW 顯示，其溫度顯示已修正為正常溫度顯示，最後一次的測量結果為 301.6°C，誤差為 1.6°C。

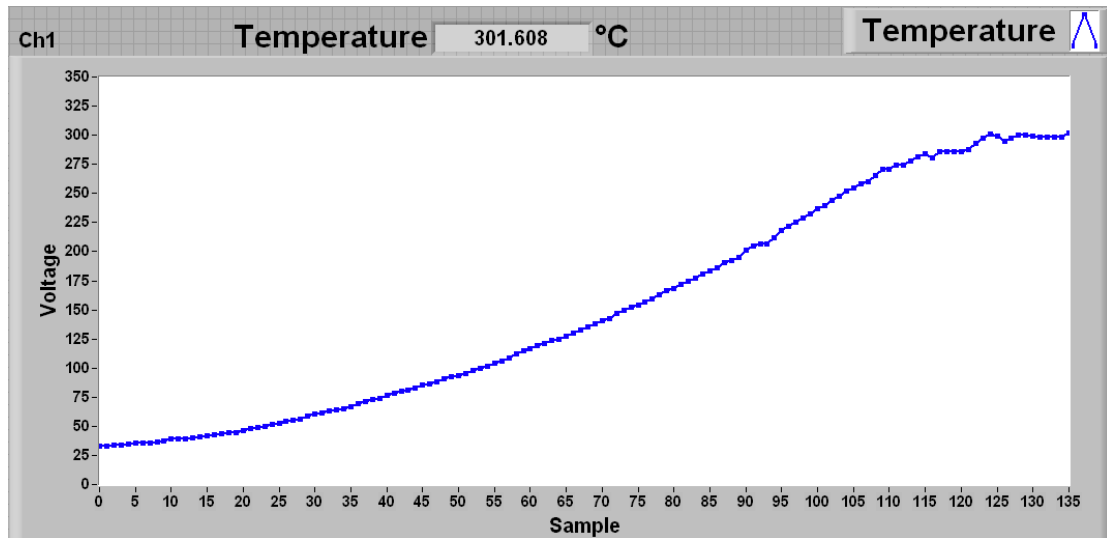


圖 5.18 室溫加熱至 300°C 無線式測量結果

5.4 討論

雖然有關於測量溫度使用焦電型和熱電堆的研究已經有人開發完成，但沒有探討在實際環境中使用熱電堆以遠端測量溫度，本研究專注於紅外線技術的非接觸式測量方法，根據實驗結果本系統可以遠端測量溫度從室溫至 300°C 的溫度範圍，並控制平均誤差在 $\pm 3^{\circ}\text{C}$ 內。

第六章、結論與未來展望

6.1 結論

使用 Thermopile 與紅外線技術的遠端測溫系統已經成功開發完成，實驗結果證明該設備適合遠端量測物體的溫度，只需要在環境中的各點佈署感測節點，搭配無線傳輸模組 Zigbee，就能監測環境中各點的溫度，最後由電腦端顯示，達到監測環境溫度的目的。但是該設備容易受到熱對流的影響，會使該設備產生錯誤的偏差。為了減少誤差，除了要將量測數據均值化(equalizing)外，加入斬波(chopping)功能是必須的，如此就可以得到較準確的數值。

此裝置使用 MSP430 做為感測器信號與無線傳輸模組間的溝通與信號處理的核心，該裝置可以在 WSN 上形成感測節點，如果加上其他感測電路，例如：煙霧感測、紫外線感測...等模組化電路可以形成一個完整的火災偵測系統，使一個智慧型住宅具備防火的保護功能，未來智慧型住宅的建立可以更便利。

6.2 未來展望

目前常用的火災警報器為偵煙式警報器，當火災發生時的濃煙遮斷偵煙式警報器內的光軸，電位改變而啟動警報器，這時火災已經發生，如果可以在微小火苗還在成長時，經由溫度變化的偵測與煙霧偵測的交叉感測在火災還未擴大時發出警報，達到預防火災的目的。

未來的研究方向積極於提升測量品質與穩定性的改善，目前長時間的定溫量測，會因為熱傳導的特性，使與感測元件外殼相連的 TH 測量的值逐漸上升，如果可以間斷性的量測會使本文系統穩定度大幅提升也較不易受環境因素的影響，例如設計一個屏蔽的截面作規律性的遮斷，使 TP 有足夠的時間散熱，可以有效減少熱傳導與熱對流所帶來的影響，鏡頭設計與菲涅爾透鏡的導入也可以有效拉長該設備的遠端量測距離，並且整合電源電路使用一般的鋰電池來驅動整個系統達到輕量化與便攜性的提升也是未來發展的重點。

第七章、 參考文獻

- [1] Y.R. Sivathanu , L.K. Tseng “Fire detection using time series analysis of source temperatures” Fire safety journal 29, Aug 1997, pp.301-315
- [2] Ulema, M., “Wireless sensor networks: architectures, protocols, and management” Systems, Man and Cybernetics (SMC) Information Assurance Workshop, June 2005, pp.356-364
- [3] Howitt, I., Gutierrez, J.A., “IEEE 802.15.4 low rate - wireless personal area network coexistence issues” Wireless Communications and Networking, March 2003, pp.1481–1486
- [4] Has, U., Wassilew, D., “Temperature Control for Food in Pots on Cooking Hobs,” Industrial Electronics, March 1999 , pp. 1030-1034
- [5] C. F. Tsai, M. S. Young, “Pyroelectric Infrared Sensor-Based Thermometer for Monitoring Indoor Objects”, Review of Scientific Instruments, Dec. 2003, pp.5267-5273
- [6] C. F. Tsai, M. S. Young, “Measurement System using Ultraviolet and Multi-band Infrared Technology for Identifying Fire Behavior”, Review of Scientific Instruments, Jan 2006, pp.155-158
- [7] Suman, S., Gaitan, M., Joshi, Y., Harman, G.G., “Wire-Bonding Process Monitoring Using Thermopile Temperature Sensor Advanced Packaging”, IEEE Transactions on Vol. 28 , Dec 2005, pp. 685–693

- [8] Lung-Tsai Li, C. F. Tsai, M. S. Young, "Design and implementation of a system with a multi-element thermopile for monitoring temperature of a plane" Review of Scientific Instruments, Aug 2009, pp.04511-1~8
- [9] Linzmeier, D., Koestler, A., Mekhael, M., Dietmayer, K.C.J., "Simulation of Thermopile IR-Sensors for Automotive Safety Applications", Vehicular Technology Conference, Sept 2006 , pp.3002- 3006
- [10] Wang Ya-xiao, Zhang Wen-Hui, "The Development of Infrared Temperature Sensor Based on Thermopile Detector TPS334 and PGA309", Measuring Technology and Mechatronics Automation, Aug 2009, pp.62-65
- [11] Hong jiakai, Hu bo, "IEEE 802.15.4 compliant RF module design Microwave, Antenna, propagation and EMC Technologies for Wireless Communications", IEEE International Symposium on Volume 2, Aug 2005, pp.910-913
- [12] McCracken, Michael, "Symmetric miniaturized heating system for active microelectronic devices" , Jul 2010, pp.075112 - 075112
- [13] Quilty, J. W., Robinson, " Thermoluminescence apparatus using PT100 resistors as the heating and sensing elements ", Review of Scientific Instruments , Aug 2007 , pp.083905-083905
- [14] Cheng-Hung Tsai, Ying-Wen Bai, "PIR-sensor-based lighting device with ultra-low standby power consumption ", Consumer Electronics, Dec 2011 , pp.1157-1164
- [15] Zappi, P. , Farella, E. , "Tracking Motion Direction and Distance

- With Pyroelectric IR Sensors ”, Sensors Journal, Sept 2010, pp.1486-1494
- [16] Ruppel, F.R., “Modeling a self-calibrating thermocouple for use in a smart temperature measurement system ”, Instrumentation and Measurement, Dec 1990 , pp.313-316
- [17] Qi Zhao , Jiaming Zhang , “Use of a Thermocouple for Malignant Tumor Detection”, Engineering in Medicine and Biology Magazine, Dec 2008, pp.64-66
- [18] C. F. Tsai, Lung-Tsai Li, Chin-Hao Li, and Ming-Shing Young “Implementation of Thermistor Linearization Using LabVIEW” IIHMSP-2009, Sept 2009, pp.530-533
- [19] 孫清華，”感測器應用電路的設計與製作”，全華科技圖書股份有限公司
- [20] 沈建華，”MSP430 系列 16 位超低功耗單片機原理與實踐”，北京航空航天大學出版社

第八章、附錄

8.1 MSP430 程式

```
#include "msp430x20x2.h"

#define TXD          0x20                // TXD on P1.5
#define Bitime      0x341              //2400bps=0x0E 9600bps=0x341

unsigned int TXData;
unsigned char BitCnt;
unsigned char Data;

void TX_Byte(void);
void delay(void);
int i,j,sum;

void main(void)
{
    WDTCTL = WDTPW + WDTHOLD;           // Stop WDT
//DCO
    BCSCCTL1 = CALBC1_8MHZ;             // Set range
    DCOCTL = CALDCO_8MHZ;              // Set DCO step + modulation */
//ADC10
    ADC10CTL0 = SREF_3 + ADC10SHT_2 + ADC10ON + ADC10IE; // ADC10ON, interrupt
    enabled
    ADC10AE0 |= 0x02;                   // PA.1,2 ADC option select
    P1DIR |= TXD;                       // Set P1.0 to output direction
//Timer
    TACCTL0 = OUT;
    TACTL = TASSEL_2 + MC_2;           // SMCLK, up mode
    P1SEL |= TXD;
//Main
    for (;;)
    {

        for(i=0;i<9;i++)
        {
            ADC10CTL0 &= ~ENC;
            ADC10CTL1 = INCH_1;
            ADC10CTL0 |= ENC + ADC10SC;           // Sampling and conversion start
            __bis_SR_register(CPUOFF + GIE);     // LPM0, ADC10_ISR will force exit
            TX_Byte();
            while(BitCnt != 0);
            delay();
        }
    }
}
```

```

    }
}
}

// ADC10 interrupt service routine *****
#pragma vector=ADC10_VECTOR
__interrupt void ADC10_ISR(void)
{

    if(ADC10CTL1 & INCH_1)
    {
        sum += ADC10MEM;                //sum=sum+ADC10MEM
        j++;
        if(j==9)
        {
            Data = (sum/10)/4;
            j=0;
            sum=0;
        }
        __bic_SR_register_on_exit(CPUOFF); // Clear CPUOFF bit from 0(SR)
    }
}

//Function Transmits Character from TXData Buffer
void TX_Byte(void)
{
    BitCnt = 0x0A;
    TXData = Data;
    TXData |= 0x100;
    TXData = TXData << 1;
    TACCTL0 = SCS + OUTMOD0 + CCIE;
    while(TACCTL0 & CCIE);
}

// Timer A0 interrupt service routine
#pragma vector=TIMERA0_VECTOR
__interrupt void Timer_A (void)
{
    TACCR0 += Bitime;
    if(BitCnt == 0)
        TACCTL0 &= ~CCIE;
    else
    {
        TACCTL0 |= OUTMOD2;
        if(TXData & 0x01)
            TACCTL0 &= ~OUTMOD2;
    }
}

```

```
TXData = TXData >> 1;
BitCnt--;
}
}

void delay(void)
{
    int i,j;
    for(i=0;i<=9;i++)
    {
        for(j=0;j<5500;j++)
            ;
    }
}
```



8.2 LabVIEW 程式

