國立臺灣科技大學

電子工程系

嵌入式系統設計實習

溫度計

|  |  |
| --- | --- |
| 指導老師： | 陳武田老師 |
|  |  |
| 實習組別： | 第十三組 |
|  |  |
| 實習組員： | 四電子三甲B9502004葉俊邑 |
|  | 四電子三甲B9502022楊竣宇 |
|  | 四電子三甲B9502028陳敬翔 |

報告日期：民國九十八年三月二十九日

§～catalog～§

[1. 實習目的： 3](#_Toc226102947)

[2. 實習說明： 3](#_Toc226102948)

[3. 實習原理： 3](#_Toc226102949)

[4. 設計內容： 17](#_Toc226102950)

[5. 設備及材料： 19](#_Toc226102951)

[6. 實做及查錯 19](#_Toc226102952)

[7. 實習結果 19](#_Toc226102953)

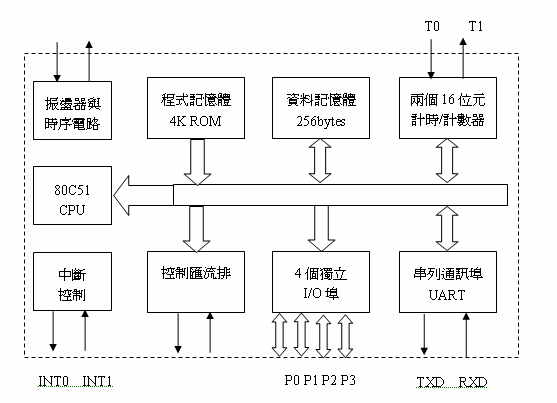
[8. 實習討論 20](#_Toc226102954)

[9. 實習心得 20](#_Toc226102955)

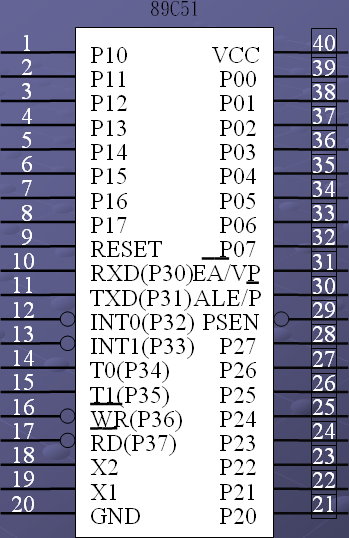
[10. 參考資料 21](#_Toc226102956)

[11. 附件 21](#_Toc226102957)

1. 實習目的：
   1. 了解AD590及ADC0804的使用，並用之設計出數位式溫度計。
2. 實習說明：
   1. 溫度計需具有華氏及攝式的切換功能
   2. 利用四位數掃描式的七段顯示器顯示現在的溫度
3. 實習原理：
   1. 8051功能與結構：
      1. 8051的基本功能特性
         1. 8 位元CPU。
         2. 32條雙向可獨立定址的I/O埠。
         3. 4K程式記憶體(ROM)，外部可擴充至64K；
         4. 128byte資料記憶體(RAM)，外部可擴充至64K
         5. 2個16位元計時/計數器, 5個中斷源，
         6. 全雙工的串列通訊埠(UART)
         7. 具有布林運算能力。
      2. 8051內部結構圖：

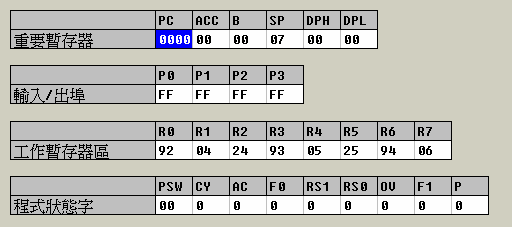


* + 1. 8051接腳電路：



* + 1. 8051內部組成：
       1. 中央處理單元(CPU)。
       2. 內部程式記憶體(ROM)-4KB。
       3. 內部資料記憶體(RAM)-256Bytes。
       4. 振盪與時序電路(12MHZ)。
       5. I/O埠(P0,P1,P2,P3)。
       6. 計時/計數器。
       7. 中斷控制電路。
       8. 串列通訊UART
    2. 一般通用暫存器：
       1. ACC：最重要的暫存器，運算與資料轉移都透過 ACC
       2. PC :程式計數器, 記載著程式下一個待執行指令位址。
       3. B 暫存器：用於乘法，除法指令的輔助暫存器。
       4. PSW 程式狀態字組：記錄程式運作時，CPU各種狀態。
       5. SP堆疊指標：重置(RESET)時，堆疊指標設為07H
       6. DPTR資料指標暫存器16位元暫存器。由DPH，DPL兩個 8位元暫存器組成。
       7. 工作暫存器：共有 RB0、RB1、RB2、RB3四組工作暫存器庫。每個暫存器庫有8個8位元暫存器，分別為R0、R1、R2、R3、R4、R5、R6、R7。

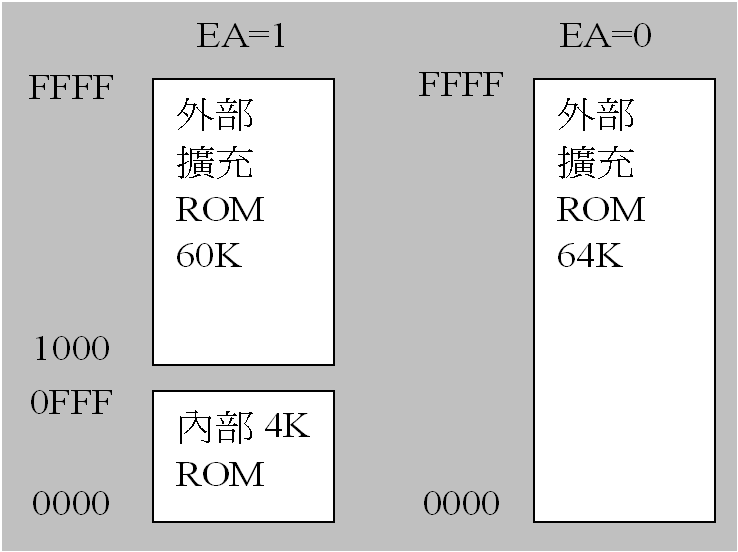
* + 1. 暫存器結構圖：



* + 1. PSW程式狀態字：

|  |  |  |
| --- | --- | --- |
| 位元 | 名稱 | 功能 |
| D7 | CY：進位旗標或借位旗標 | (Carry Flag)進位旗標，用來表示算術指令運算後的結果，其資料的bit 7是否有進位或借位。  加法運算時(ADD)的結果：有進位C=1，沒有進位C=0。  減法運算時(SUB)的結果：有借位C=1，沒有借位C=0。 |
| D6 | AC：半進位旗標或半借位旗標 | (Aux Carry Flag)半進位旗標，用來表示運算後資料的bit 3是否有向bit 4進位或借位。  加法運算時(ADD)的結果：有進位AC=1，沒有進位 AC=0。  減法運算時(SUB)的結果：有借位AC=1，沒有借位 AC=0。 |
| D5 | F0：通用位元 | 可作為一般的讀/寫位元。 |
| D4 | RS1：暫存器庫選擇位元1 | 暫存器庫選擇(Regiser Bank Select)位元1及位元0。  RS1 RS0 暫存器庫選擇  0 0 RB0(位址00h-07h)  0 1 RB1(位址08h-0Fh)  1 0 RB2(位址10h-17h)  1 1 RB3(位址18h-1Fh) |
| D3 | RS0：暫存器庫選擇位元0 |
| D2 | OV：溢位旗標 | (Over)溢位旗標，表示程式經算術或邏輯運算後的結果是否有溢位，若是OV=1，若不是 OV=0。 |
| D1 | - | 空位元 |
| D0 | P：同位元旗標 | (Parity)同位元旗標，表示累積器的內容為奇數個“1”則P=0，偶數個“1”則P=1。 |

* + 1. 8051記憶體介紹：



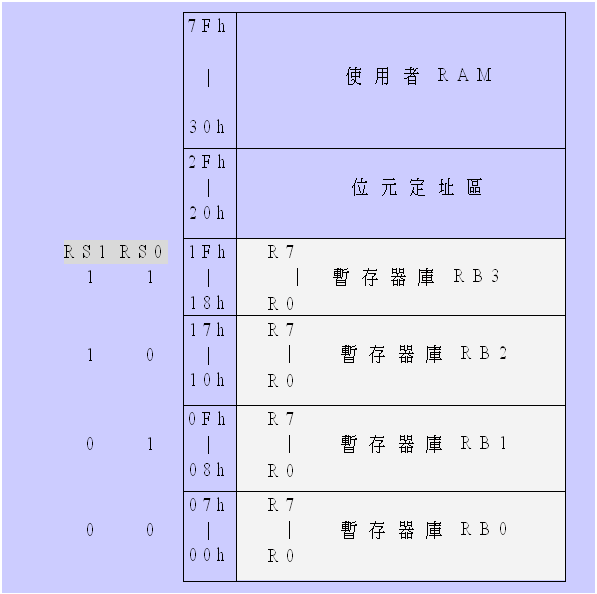
* + 1. 中斷向量位址：

|  |  |  |
| --- | --- | --- |
| 中斷 | 位址 | 功能 |
| RESET | 0000H | 系統重置啟始位址 |
| INT0 | 0003H | 外部中斷INT0向量 |
| INT1 | 0013H | 外部中斷INT1向量 |
| TIMER0 | 000BH | 計時計數中斷TIMER0向量 |
| TIMER1 | 001BH | 計時計數中斷TIMER1向量 |
| TIMER2 | 002BH | 計時計數中斷TIMER2向量 |
| UART | 0023H | 串列埠中斷向量 |

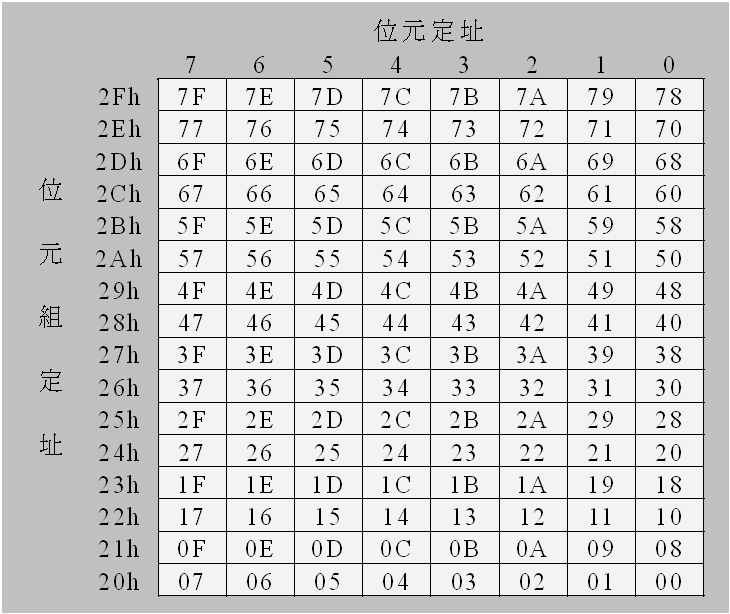
* + 1. 資料記憶體結構：



* + 1. 暫存器庫：



* + 1. 可位元定址區：



* + 1. 中斷向量位址：

|  |  |  |
| --- | --- | --- |
| 中斷 | 位址 | 功能 |
| RESET | 0000H | 系統重置啟始位址 |
| INT0 | 0003H | 外部中斷INT0向量 |
| INT1 | 0013H | 外部中斷INT1向量 |
| TIMER0 | 000BH | 計時計數中斷TIMER0向量 |
| TIMER1 | 001BH | 計時計數中斷TIMER1向量 |
| TIMER2 | 002BH | 計時計數中斷TIMER2向量 |
| UART | 0023H | 串列埠中斷向量 |

* 1. 中斷計時計數與串列通訊：
     1. MCS51的中斷簡介
        1. 單晶片在處理外部輸出入訊號的方式有兩種：
           1. PIO（程式I/O-ProgramI/O）：

透過程式指令讀取I/O埠狀態。但這種輸入方式無法在第一時間立即處理某些具有時效性的外部輸入訊號。

* + - * 1. IIO（中斷I/O-InterruptI/O）：

可立即中斷目前正在執行的程式，並做立即的回應與處理，本章將介紹中斷的原理，型式，與應用。

* + 1. MCS51的中斷型式：
       1. 外部中斷：

8051的外部中斷是指來自晶片硬體接腳INT0、INT1的中斷。這兩個外部中斷的觸發方式有低準位觸發（lowleveltrigger）與負緣觸發（fallingedgetrigger）兩種，可經由TCOM暫存器的ITX（X=0或1）位元設定。

* + - 1. 計時/計數中斷：

當8051的計時/計數器（TIMER0、TIMER1）產生溢位時，溢位旗標TF0（或TF1）會自動設為1，直到CPU跳到對應的中斷向量位址，執行中斷服務程式時，才會自動將TF0（或TF1）清除為1。

* + - 1. 串列埠中斷：

當8051的發射中斷旗標TI或接收中斷旗標RI為1時，會產生串列中斷請求。在中斷服務程式中，必須用指令清除TI與RI，因為硬體不會自動清除這兩個位元。

* + 1. MCS-51的外部中斷：
       1. MCS-51有5個中斷源，2個外部硬體中斷，2個計時/計數中斷，1個串列埠傳輸中斷：
          1. 外部中斷：

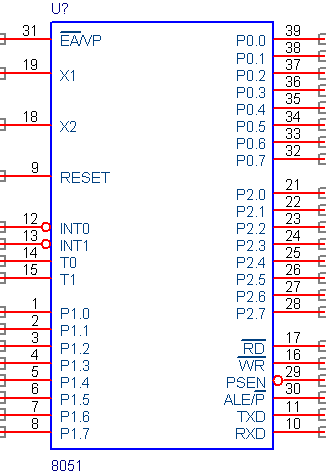
INT0、INT1（P3.2，P3.3）。

* + - * 1. 計時/計數中斷：

T0，T1（P3.4，P3.5）。

* + - * 1. 串列通訊輸出/入：

TXD，RXD（P3.0，P3.1）。



* + 1. MCS-51的中斷功能：
       1. 中斷源的致能方法（使用IE暫存器）。
       2. 外部中斷觸發方式（TCON）。
       3. 中斷優先順序（使用IP暫存器）。
       4. 中斷向量（中斷向量表位址）。
       5. 中斷處理流程。
    2. 中斷致能控制（IE暫存器）：
       1. 中斷源利用中斷致能暫存器（IE暫存器）來控制其致能與禁能。當中斷源被致能時，中斷才能被CPU接受。八位元例如，第0個外部中斷與第1個計時器中斷，IE必須令為#89H=#10001001B，即指令為＜MOVIE，#89H＞，第INT0與第TIMER1中斷。
       2. 當IE小於或等於80H（即IE=#0xxxxxxxB，或IE=#10000000B），所有中斷皆無效（即EA=1，或EX0~EX5皆為0）。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| IE.7 | IE.6 | IE.5 | IE.4 | IE.3 | IE.2 | IE.1 | IE.0 |
| EA | 未用 | ET2 | ES | ET1 | EX1 | ET0 | EX0 |

* + - 1. 功能說明：



* + 1. 外部中斷觸發方式（TCON暫存器）：
       1. INT0、INT1外部中斷的觸發方式有兩種：
          1. 低準位觸發（lowleveltrigger）：

第0外部中斷（INT0），使用CLRIT0指令或IT0=0;

第1外部中斷INT1，使用CLRIT1指令或IT0=0;

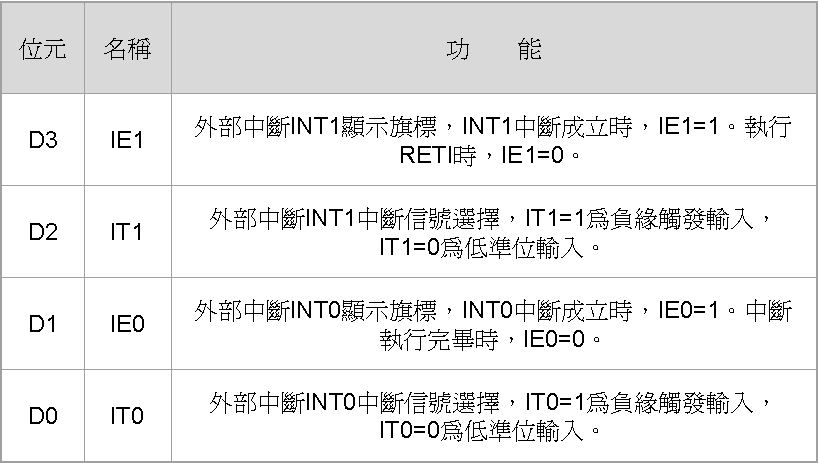
* + - * 1. 負緣觸發（fallingedgetrigger）：

第0外部中斷（INT0），使用SETBIT0指令或IT0=1;

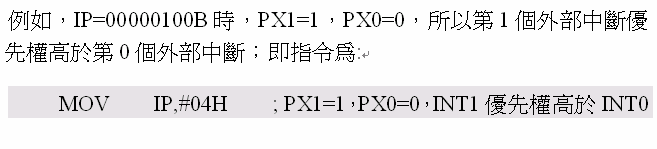
第1外部中斷INT1，使用SETBIT1指令或IT1=1;

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| TCON.7 | TCON.6 | TCON.5 | TCON.4 | TCON.3 | TCON.2 | TCON.1 | TCON.0 |
| TF | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |

* + - 1. 功能說明：



* + 1. 中斷優先順序控制（IP暫存器）：
       1. MCS-51所有中斷源都是使用中斷優先權暫存器（IP暫存器）來控制中斷優先順序。



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| IP.7 | IP.6 | IP.5 | IP.4 | IP.3 | IP.2 | IP.1 | IP.0 |
| 未用 | 未用 | PT2 | PS | PT1 | PX1 | PT0 | PX0 |

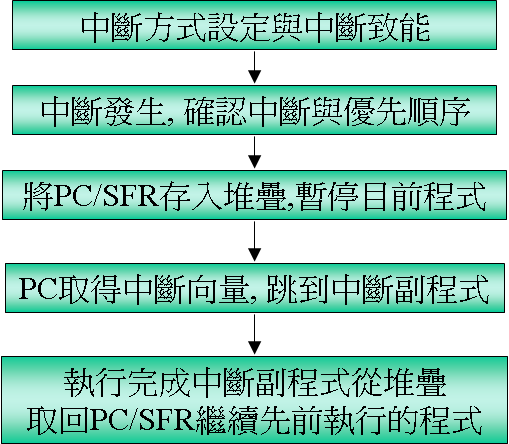
* + - 1. 功能說明：



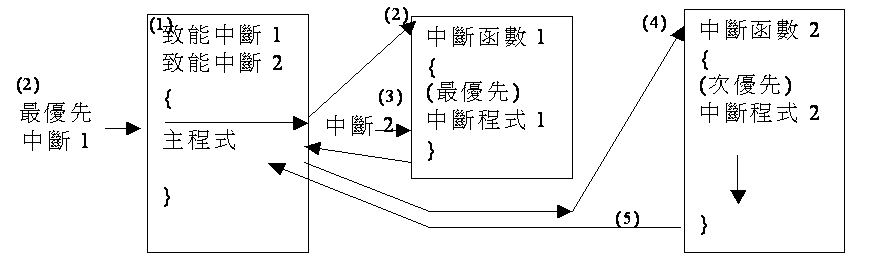
* + 1. 中斷向量位址：

|  |  |  |
| --- | --- | --- |
| 中斷 | 位址 | 功能 |
| RESET | 0000H | 系統重置啟始位址 |
| INT0 | 0003H | 外部中斷INT0向量 |
| INT1 | 0013H | 外部中斷INT1向量 |
| TIMER0 | 000BH | 計時計數中斷TIMER0向量 |
| TIMER1 | 001BH | 計時計數中斷TIMER1向量 |
| TIMER2 | 002BH | 計時計數中斷TIMER2向量 |
| UART | 0023H | 串列埠中斷向量 |

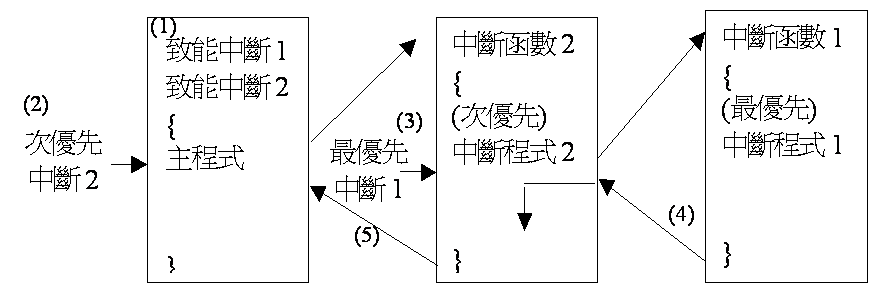
* + 1. 中斷處理流程：



* + 1. 中斷程式的工作方式：
       1. 最優先中斷先輸入：



* + - 1. 次優先中斷先輸入：



1. 設計內容：
   1. 程式碼：

//#include <REGX51.H>

#include <AT89X51.H>

void Delay(int num)

{

int i,j;

for(i=0;i<100;i++)

for(j=0;j<num;j++);

}

main()

{

char segment[10]={0xC0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x98};

char segment2[10]={0x40,0x79,0x24,0x30,0x19,0x12,0x02,0x78,0x00,0x18};

char F = 0x8E;

char C = 0xc6;

char tempt=0;

char scanline[4]={0x0e,0x0d,0x0b,0x07};

unsigned int ad\_data=0;

unsigned char segnum[4]={0,7,8,9};

float ad\_temp=0;

int i=0;

P0=0xFF;// input adc

P2=0x00;// scan

P1=0x00;// scan segment

P3=0x01;// mode

while(1)

{

ad\_temp =15;

//ad\_temp = P0;

ad\_temp = (ad\_temp\*1000)/ 51 ;

if(P3&0x01)

{

ad\_data = ad\_temp;

tempt = C;

}

else

{

ad\_temp = ((ad\_temp \* 9)/ 5 ) + 320;

ad\_data = ad\_temp;

tempt = F;

}

segnum[1] = ad\_data % 10;

segnum[2] = (ad\_data/10)%10;

segnum[3] = (ad\_data/100)%10;

//segnum[3] = ad\_data /1000;

//display

for(i=0;i<4;i++)

{

P2=scanline[i];

P1=tempt;

if((i==1)|(i==3))

{

P1=segment[segnum[i]];

}

if(i==2)

P1=segment2[segnum[i]];

Delay(1);

}

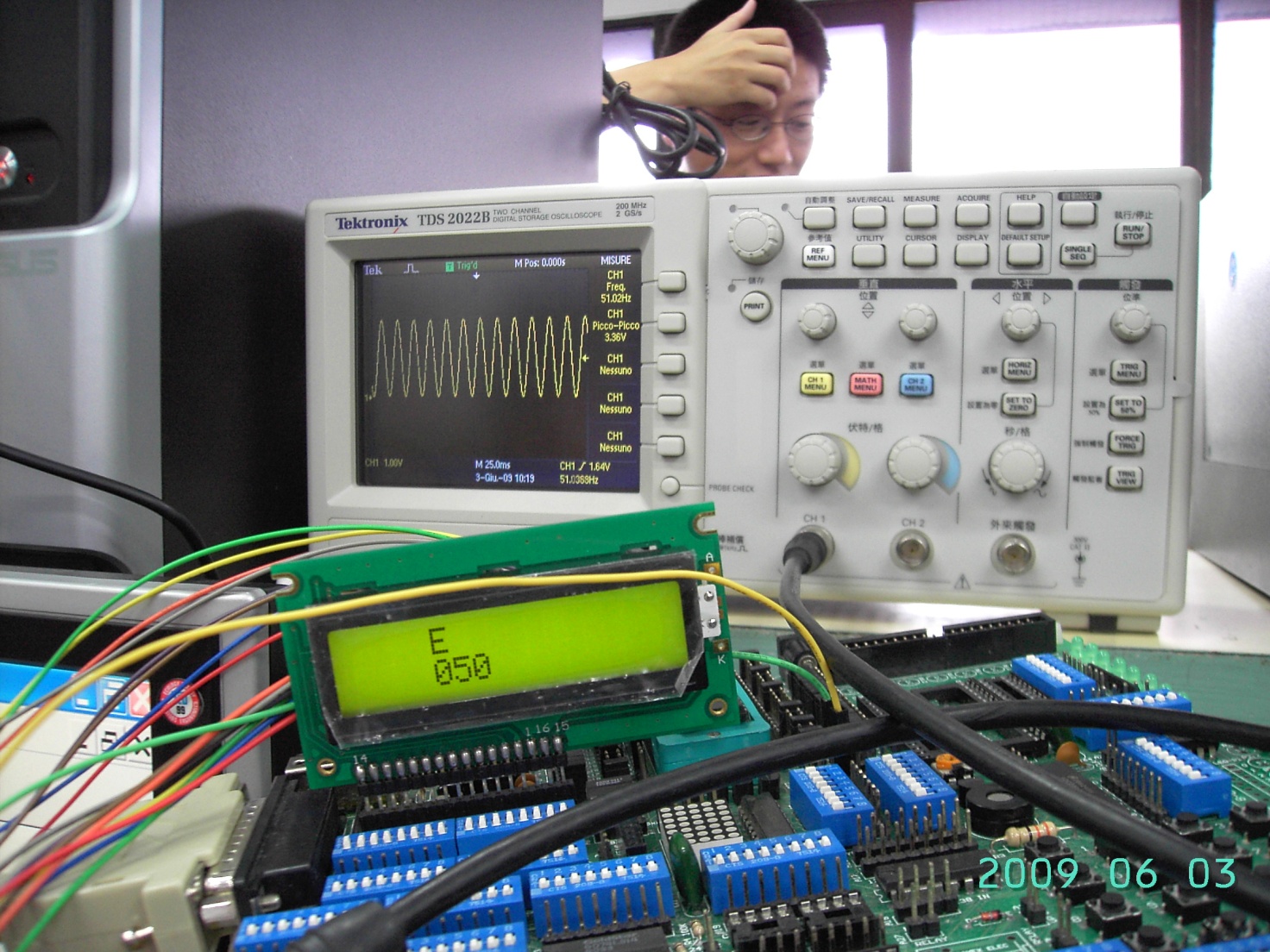
}

}

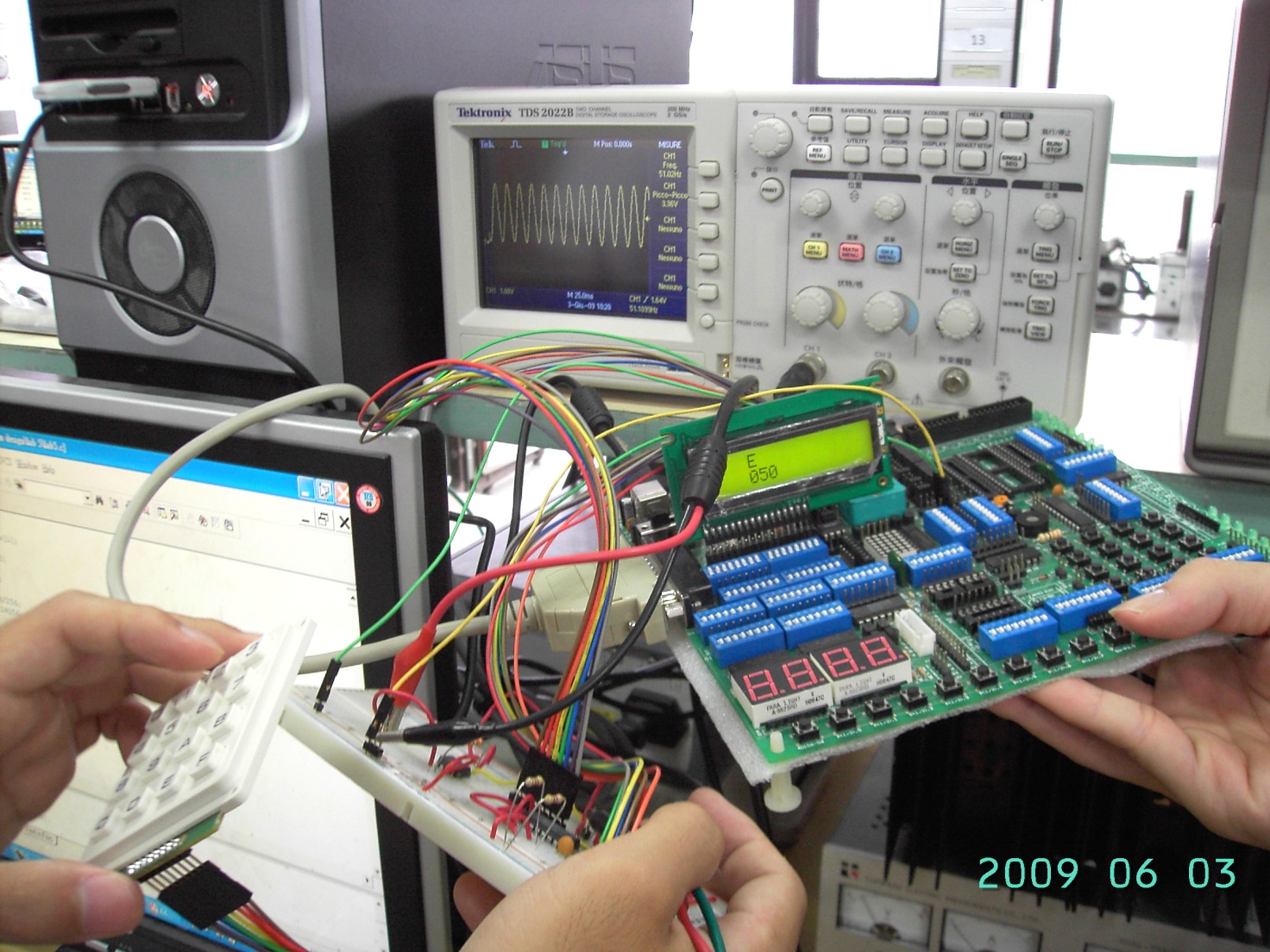
2. 設備及材料：
   1. 德源科技8051及AVR綜合實驗板
3. 實做及查錯

實驗方面可能比較大的問題是溫度轉換的靈敏度吧，條那顆可變電組要非常的仔細，才能合乎目前的溫度，而且為了測試是否能正常的感應溫度，特地去借了一隻銲槍stand by。

1. 實習結果
   1. 50Hz產生結果



* 1. 裝置全景



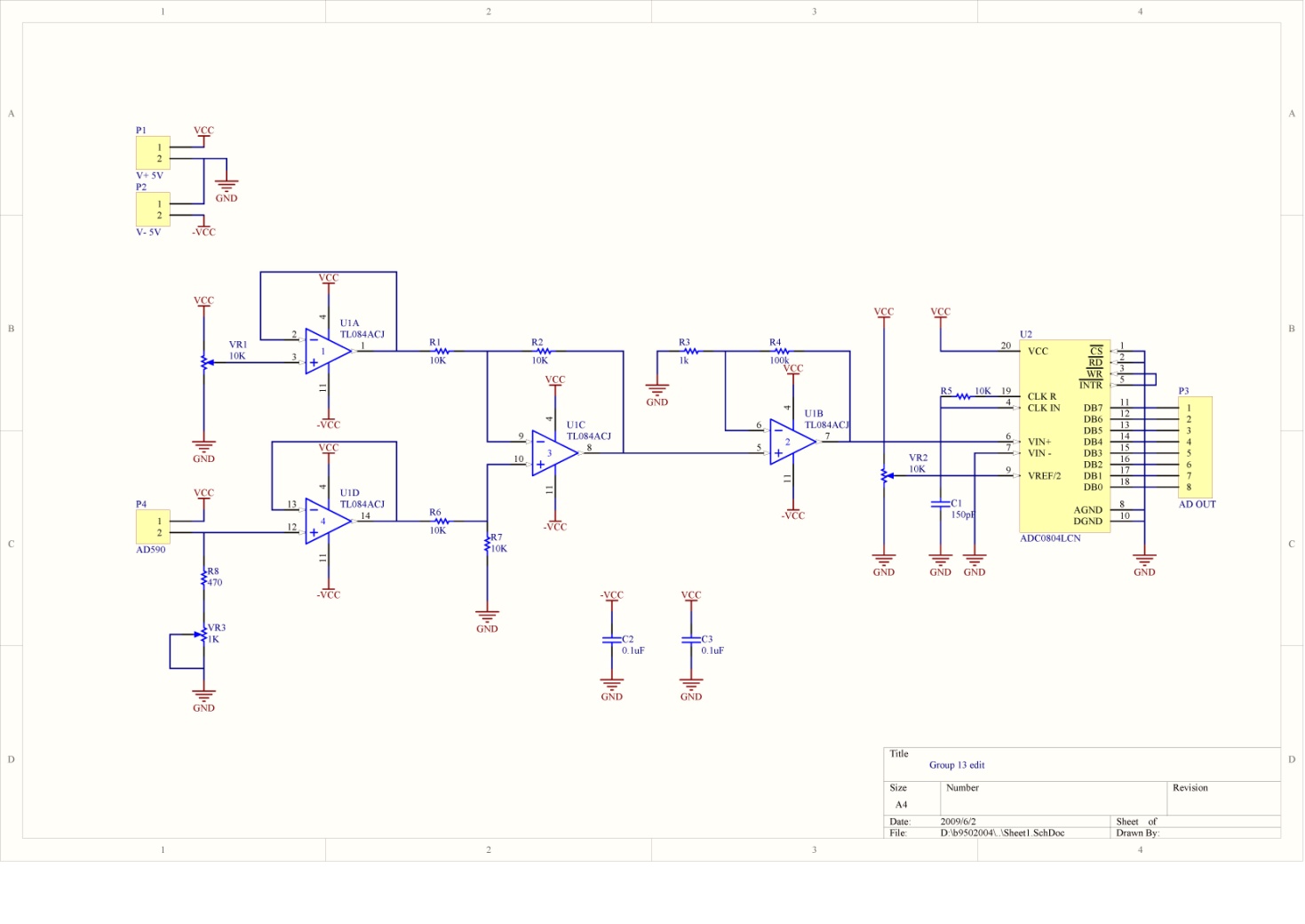
1. 實習討論

由於剛開始的軟體沒有寫得很好，導致溫度到達70幾度就上不去了，被老師看到這點後，我們因此遭受到第一次的駁回，後續我們在軟體上將數值多做一些運算，使其可以衝破到100度以上，才平安過關，所以看清楚老師要的成果也是很重要的。

1. 實習心得

溫度轉換一直都是我們這些初學者很重要的一個實習，老在我們高職的時後就認為AD590是個很神奇的東西，直到現在我們可以如心所欲的去控制它，感覺我們所有人的能力真的有在成長，雖然在別人的眼光來看這是一件小事情，但對我們來說，做出這項成果還是會有一番的成就感，加上老師拿打火機去燒它，那種看著自己的作品，老嚴格的裁判評審下，得到過關的稱號，真的很戲劇化。

2. 參考資料
   1. 楊明豐，碁峰資訊股份有限公司，「8051單晶片C語言設計實務：使用Keil C」
   2. 實習板功能介紹光碟，各單元電路圖／完整電路圖.pdf
3. 附件
   1. 完整電路圖：
      1. 外接電路電路圖：



國立臺灣科技大學

電子工程系

嵌入式系統設計實習

報告結束

感謝老師指導與詳閱