

## RS485 儀器連線程式示範(二)

### Turob C 的程式示範

串列資料的接收函式依然是使用 bioscom()，只要我們傳遞的參數不同，就可以指定是接收模式，由於 PC 大部份的串列接收埠內部有 16 Bytes 的緩衝區，所以我們可以從容地執行其他的程式，有空時才檢查是否有資料進來，不過前提是傳輸速度不能太快，否則還是會有資料串溢位的可能，也就是說：串列資料進來後，還是要儘快進到我們程式的陣列當中，以免被後來的資料覆蓋(over-write)。

在這裡我們要示範的是用詢問(Polling)的方式去檢查串列埠，若一有資料進入後，就直接把該資料顯示在螢幕上，在 DOS 的環境上用這種寫法最容易理解，但是一定要 ”經常” 去呼叫 bioscom() 函式，否則還是會有部份資料被覆蓋的可能。對於 bioscom() 函式還不理解的讀者請看前一週的解說，或是參考 Turbo C 的 Reference Guide 一書中的說明。

#### 485DEMO2.C 內容

```
#include <stdio.h>
#include <conio.h>
#include <ctype.h>

#define PROTOCOL      0xe3
int com;

main()
{
    int m,c;
    int status;
    int id;

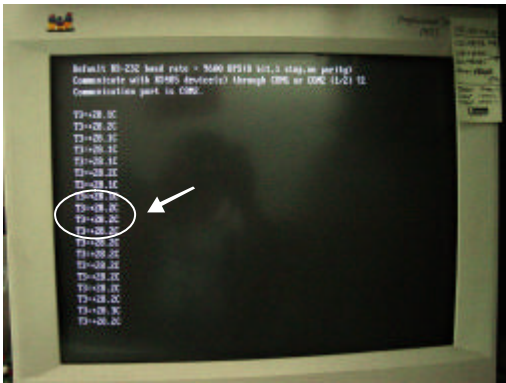
    delay(0);
    dclrscr();
    communication_init();
    m=0;
    id=3;
    while (1)
    {
        status=bioscom(3,0,com) & 0x100;
        if (status)
        { c=bioscom(2,0,com);
          if (c !=0) printf("%c",c);
        }
    }
}
```

```
        if (kbhit()!=0) exit(0);
        m++;
        delay(1);
        if (m==30000) { sio_out(id); m=0; }
    }
}

int sio_out(int data)
{
    bioscom(1,data,com); /* command,data */
    delay(10);
}

int communication_init()
{
    printf("Default RS-232 baud rate = 9600 BPS(8 bit,1 stop,no parity)\n");
    printf("%cCommunicate with RS485 device(s) through COM1 or COM2 (1/2) ?",7);
    scanf("%d",&com);
    printf("Communication port is COM%d.\n",com);
    switch (com)
    {
        case 1: com=com-1; break;
        case 2: com=com-1; break;
        default:com=0;    break;
    }
    bioscom(0,PROTOCOL,com);
    printf("\n");
}
```

假設 AT2051 控制器的 ID 是 3，我們的程式就是送 03H，即 0000 0011B 的二進位串列值，當 AT2051 控制板收受該值後，會在最短的時間內把溫度值回送給 PC，AT2051 控制板事先就把所量到的溫度值化成標準 ASCII 碼，當有人要時就不假思考地將這個 ASCII 字串送出。如果 PC 在送完 ID 碼後就立即進入接收模式，當然就可以在螢幕上看到一連串的溫度值，在 AT2051 控制板回送的資料最後有換行的控制碼(CR+LF)存在，所以我們觀看螢幕上的字串，除了每次都從第一欄位開始顯示，而且 ASCII 的溫度填滿 25 行後，整個螢幕還會上捲，最下面一行則是最近一次的溫度值。在螢幕上我們也可以看到，雖然 AT2051 控制板上只有兩個數字的顯示，但是從串列埠上多送了小數點下一位解析度，所以我們在螢幕上看到的溫度值是攝氏+27.3 度，而不是攝氏+27 度。



〔圖 1〕在螢幕上看到 AT2051 送回的 ASCII 字串



〔圖 2〕我們把 AT2051 控制板掛在書架上，單獨用一個 6V 蓄電池供電

DOS 的環境下也可以改成中斷的方式接收串列資料，但是其困難度較高而且很容易造成當機，我們在此不做示範。當然更方便的方法是在 Windows 下用 VB 的方式來處理，串列通訊就可以很容易處理了，不過，還是隱藏有許多技巧在裡面，除非你親自動手實驗，否則這些知識是無法從一般的書籍當中看到的。

參考資料：

- 1 Turbo C Reference Guide (不知道市面上還有類似的資料)
- 2 旗標 “C 語言實務” 施威銘工作室與林伸茂合著
- 3 旗標 ” I/O 自動控制實務 ” 林伸茂著
- 4 旗標 “單晶片 8051 徹底研究 基礎篇” 林伸茂著
- 5 旗標 “單晶片 8051 徹底研究 實習篇” 林伸茂著



〔圖 3〕Turbo C Reference Guide