

控制的第一步

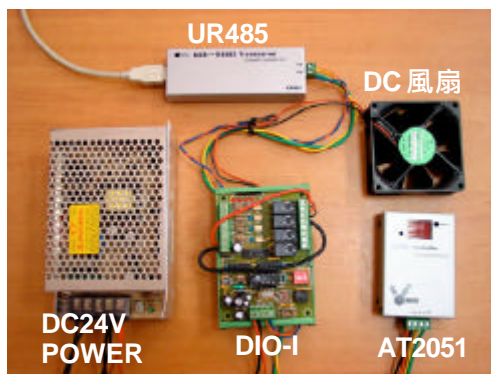
前幾篇文章，我們介紹了 MSComm 的傳送與接收，現在您應該要具備撰寫 PC 端控制程式的基本能力了。在介紹 RS485 的控制方式之前，我們先來建立一些控制的基本概念。

使用 RS485 來進行控制，不外是對其信號傳遞的穩定性與正確性有足夠的信心，然而我們都知道 RS485 無法進行全雙工的運作，也就是無法同時進行傳送及接收的動作，否則會因為資料的碰撞導致不可預期的錯誤產生。

因此，使用 RS485 通訊，我們習慣將傳送與接收的動作以程式區隔開來，以一個中心控制器來掌握所有 RS485 儀器的動作，而這個中心控制器可以是一台 RS485 的 Device，但是通常我們會以 PC 來當做 Controller，因為 PC 可以說是最容易取得且價格不算太昂貴的控制儀器，更重要的是，它的操作較人性化，功能及特性或許不比一般 RS485 儀器強悍，但具備資料庫整合及很好的擴充性，這是使用 PC 控制最吸引人的部分。接下來，我們要以一個簡單的溫度控制程式，介紹 RS485 的控制寫法。

準備器材：

1. AT2051 控制板 一組
2. DIO-I 控制板 一組
3. UR485 轉接盒 一組
4. DC24V 電源供應器 一台
5. DC24V 電扇 一台



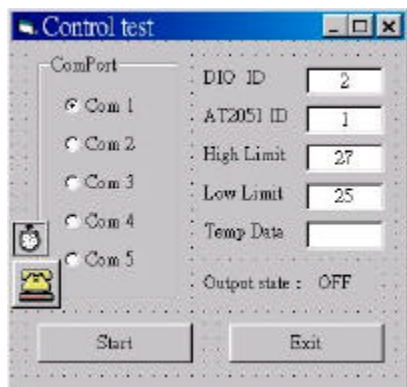
〔圖 1〕器材總覽

動作原理：

當 PC 收到來自 AT2051 的溫度資料，經判斷超過溫度上限時，即送訊號至 DIO-I 控制板，啟動電風扇強制對流以降低溫度；若溫度低於下限值，則將電風扇關閉。

程式內容與解說：

1. 表單的配置狀況



〔圖 2〕表單的配置狀況圖

詳細的設定，可在旗威科技交流網首頁的旗威科技產品應用實例中下載該程式範例，網址是

<http://www.chipware.com.tw/example/control.zip>

在畫面的設計上，我們為了增加程式的靈活度，因此不管是控制板的 ID 也好，溫度判斷的上下限也好，都可以在畫面上直接修改。而且啟動的狀態及量測的溫度值都一起顯示出來，您可以對照一下控制板上的顯示與實際的運作情況吻不吻合。不過這個程式有一些小地方要注意的是：上限值設定的溫度不可以低於下限值，否則在臨界溫度時會一直出現想關卻關不掉的情形，想知道那是什麼情形，自己試試就知道囉！另外也不要再在程式啟動後隨意改變 ComPort 的值，不然也可能會發生錯誤的情形。

控制開啟與關閉的溫度值儘量不要設成一樣，因為當量測的溫度值處於臨界邊緣時，電風扇就會不斷地開開關關，這樣的動作容易使電風扇不斷受到過大的瞬間電流衝擊而造成損壞。對程式設計的觀點來看，這樣的設計容易產生資料持續振盪而不容易達到平衡的情形，因此要儘量避免。

2. 程式碼的部分

```

Option Explicit
Dim buffer$      ' 字串暫存區
Dim com As Integer ' 序列埠變數
Dim D_ID, A_ID As Integer ' 控制板 ID
Dim H_LMT, L_LMT, TP As Double ' 量測溫度及上下限變數
'-----
Private Sub Command1_Click() ' 控制系統開始運作
    buffer$ = "" ' 清空暫存區
    D_ID = Text5 ' 設定 DIO-I 控制板的 ID
    A_ID = Text1 ' 設定 AT2051 控制板的 ID
    H_LMT = Val(Text2) ' 設定溫度上限
    L_LMT = Val(Text3) ' 設定溫度下限

    Call ComPortOpen ' 開啟序列埠
    MSComm1.Output = Chr(A_ID) ' 送出指令讀取溫度值
    Do
        DoEvents
        buffer$ = buffer$ + MSComm1.Input
    Loop Until Len(buffer$) > 10

    Text4 = Mid(buffer$, 5, 4) ' 將讀取的溫度顯示在視窗上
    TP = Val(Text4) ' 將量測溫度存在溫度變數中

    If TP > H_LMT Then ' 超過上限則啟動風扇
        Call Switch_On
    ElseIf TP < L_LMT Then ' 低於下限則關閉風扇
        Call Switch_Off
    End If

    If Timer1.Enabled = False Then Timer1.Enabled = True ' 啟動定時監控
End Sub
'-----
Private Sub Command2_Click() ' 離開程式
    If (MSComm1.PortOpen = True) Then MSComm1.PortOpen = False
    If Timer1.Enabled = True Then Timer1.Enabled = False
    End
End Sub

```

```

Private Sub Form_Load() ' 表單初始設定
    com = 1 ' ComPort 預設值為 1
    Timer1.Interval = 500 ' 定時監控設定為每秒兩次
End Sub
-----
Private Sub Option1_Click() ' 序列埠設定
    com = 1
End Sub
-----
Private Sub Option2_Click()
    com = 2
End Sub
-----
Private Sub Option3_Click()
    com = 3
End Sub
-----
Private Sub Option4_Click()
    com = 4
End Sub
-----
Private Sub Option5_Click()
    com = 5
End Sub
-----
Sub ComPortOpen() ' 開啟連接埠
    If (MSComm1.PortOpen = True) Then MSComm1.PortOpen = False
    ' 預設為 COM 1
    MSComm1.CommPort = com
    ' 連線速度 9600 baud、無同位檢查、資料位元 8、停止位元 1
    MSComm1.Settings = "9600,N,8,1"
    ' 告訴控制項當使用 Input 時，讀取整個暫存區
    MSComm1.InputLen = 0
    ' 開啟序列連接埠
    If (MSComm1.PortOpen = False) Then MSComm1.PortOpen = True
End Sub
-----
Sub Switch_On() ' 風扇開啟
    MSComm1.Output = Chr(D_ID) ' 送出 DIO-I 板的 ID
    MSComm1.Output = "o3" ' 送出啟動的指令
    MSComm1.Output = vbCrLf ' 送出結束碼
    Label7.Caption = "ON" ' 將啟動狀態顯示在螢幕上
End Sub
-----
Sub Switch_Off()
    MSComm1.Output = Chr(D_ID) ' 送出 DIO-I 板的 ID
    MSComm1.Output = "o0" ' 送出啟動的指令
    MSComm1.Output = vbCrLf ' 送出結束碼
    Label7.Caption = "OFF" ' 將啟動狀態顯示在螢幕上
End Sub
-----
Private Sub Timer1_Timer() ' 定時量測溫度
    Command1_Click
End Sub

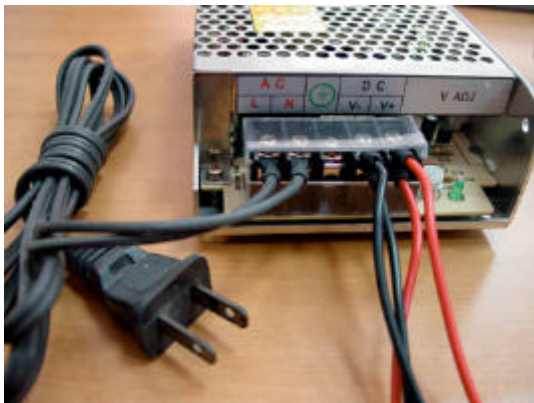
```

上面的程式裡，較為重要的功能都已經模組化，例如序列埠的開啟、風扇啟動、風扇關閉等動作都有單一副程式來掌握動作。程式的運作，是當溫度值超過上限則啟動風扇，低於下限則關閉風扇，最重要的部分是定時量測的部分，相信眼尖的您一定發現，在計時器的動作非常單純，就是每半秒按一次量測按鍵，您或許會有疑問：為什麼不直接把系統的動作寫在 Timer 裡面就好，而是改用 Timer 來按按鍵呢？

還記得我們提到過使用 Timer 的限制嗎？Timer 的動作時間必須小於 Timer 的時間間隔，重點來了：如果在 Timer 裡面加入了 Do Loop 的指令，又加入一大堆的判斷式，那麼會出現怎樣的問題呢？就是動作做不完，動作的時序會有怪怪的反應，這個部分正是我們寫程式所要避免的。記住一個觀念，計時器就單純地計時或計數就好了，除非必要，否則儘量避免讓計時器處理會影響時序的動作。

配線狀況：

1. 將電源供應器上的直流與交流電線先接好。



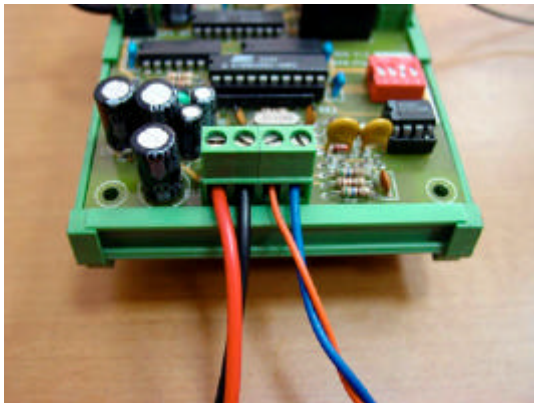
〔圖 3〕電源供應器的配線圖

2. 將 AT2051 控制板的接座依 V+、GND、SIO、/SIO 的順序接上。



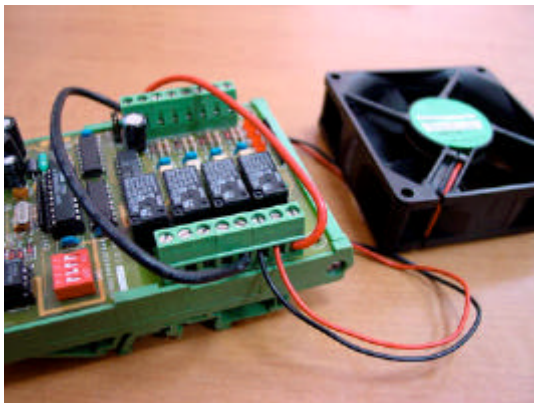
〔圖 4〕AT2051 控制板的配線圖

3. 將 DIO-I 控制板的接座依 V+、GND、SIO、/SIO 的順序接上。



〔圖 5〕DIO-I 控制板的配線圖

4. 利用 DIO-I 控制板上所提供的 DC24V 電源接到 Relay 接點上，並將電扇的正負極依標示接好。



〔圖 6〕風扇的配線圖

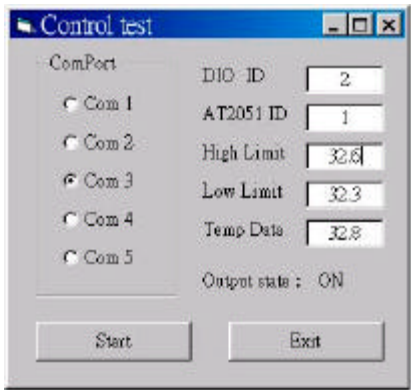
5. 最後將 RS485 串列的線接上，如果您有 RS485 專用線材，那麼只要直接接上就可以了，如果沒有，請記得一定要把 SIO 與/SIO 的傳輸線對絞，以免傳輸訊號無法正確傳遞。



〔圖 7〕UR485 的配線圖，線材一定要對絞唷！

程式執行情形：

1. 當量測的溫度超過上限時，電風扇會啟動讓溫度降下來。



〔圖 8〕PC 端顯示啟動時的狀況



〔圖 9〕啟動時 RLY1 和 RLY2 的 LED 會亮

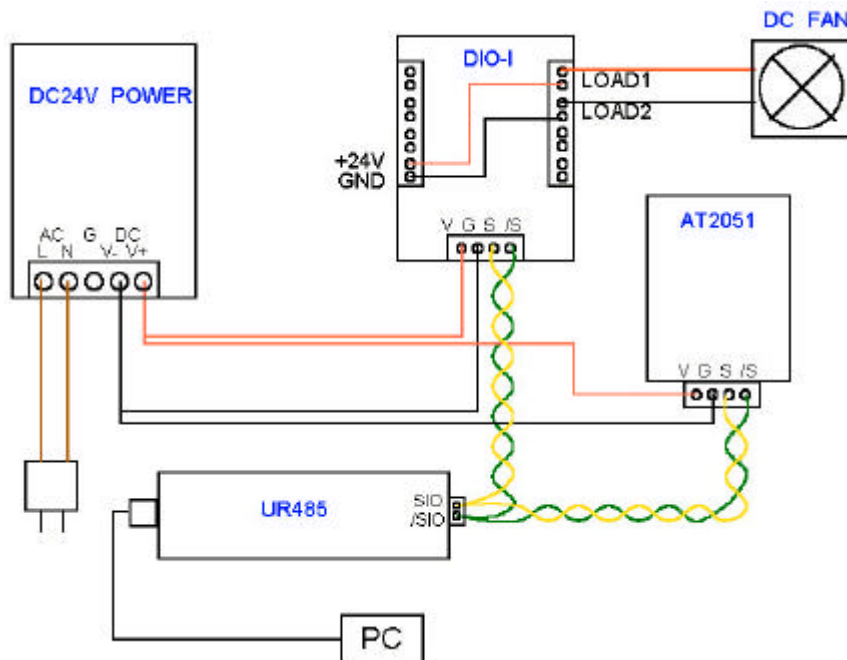
2. 當量測的溫度低於下限時，電風扇會自動關閉。



〔圖 10〕PC 端顯示關閉時的狀況



〔圖 11〕關閉時 RLY1 和 RLY2 的 LED 會關閉



〔圖 12〕配線示意圖

如果您要改成一般家用電扇，請將 Relay 接點上的直流電源供應改成交流電源供應，並確認接線無誤後才通電，以免造成觸電的危險。

問題探討：

1. 程式執行後，AT2051 及 DIO-I 板都沒有任何反應，這是什麼原因？

Ans：請確認您所傳送的 ID 值與控制板的設定相符，且兩控制板的 ID 沒有衝突的情況。如果還是沒有反應，請分別檢查 AT2051 及 DIO-I 是否有損壞，或 RS485 的接線有反接的情形。

2. 程式上溫度的量測與讀取都正確，可是電風扇就是無法正常運作，這是什麼原因？

Ans：這可能是您所使用的 PC 的 CPU 處理速度較慢，在傳送訊號上出了一點問題，建議將 Timer 的時間加長，並在傳送給 DIO-I 板的每個 Byte 之間加入一小段 Delay，這樣應該就可以解決了。另外，請確認您的電扇是一通電就 ON 的裝置，而不是需要其他開關控制的，如果有，請記得先將它打開，不然是一定不會動的。

3. 如果我要控制很多組電扇，可是接點不夠用，且電流量超過控制板 Relay 所能負荷的範圍，那我應該怎麼處理？

Ans：比較好的作法就是用耐電流量小的 Relay 控制耐電流量大的 Relay 的開關狀態，再用耐電流量大的 Relay 控制您所使用的電扇，這樣就可以解決這個問題了。一般來說，使用在工業配電上的 Relay，有耐電流量 100A 以上的規格，您可以到電子材料行詢問看看。

以上是一個 RS485 連線系統完整組成的過程，不過，這只是個開端，下一篇文章，我們要深入在控制 RS485 時所會面臨的一些問題。提示一下，那和本文的問題探討有密切的關聯性唷！