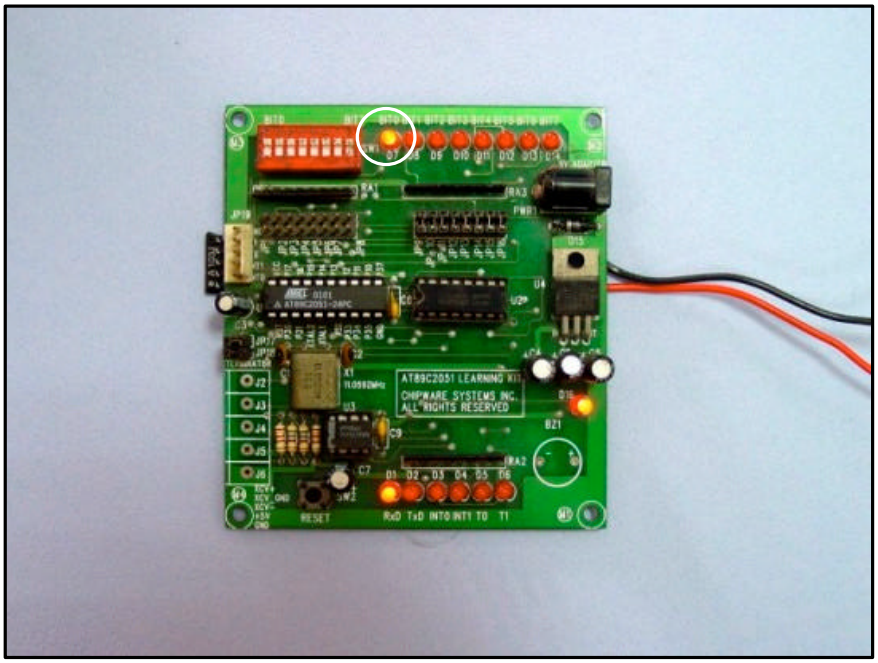
常用指令排行榜(五)

連續看了好久的 JMP，有沒有很累的感覺？本文還要繼續排徊在 JMP 的應用上，不過改用程式範例的方式來介紹，讓您對 JMP 的指令能更容易上手。還記得我們這一連串文章的啟始點是要控制燈號的嗎？我們回到第二篇文章中所提到的程式看看吧！

```
START   MOV     R1,#00H     ;系統啟始時先延遲一小段時間使其穩定
$       DJNZ    R1,$
        MOV     SP,#60H     ;堆疊從 60H 開始（常用設定範圍是 3FH~7FH）
;
LOOP    CLR     P1.0        ;也可以寫成 CLR 90H.0
        CALL    DELAY       ;延遲上一個狀態一小段時間
        SETB    P1.0        ;也可以寫成 SETB 90H.0
        CALL    DELAY       ;延遲上一個狀態一小段時間
        SJMP    LOOP        ;回到 LOOP  再循環
;
DELAY   MOV     R0,#00H     ;延遲時間的副程式
$1      MOV     R1,#00H
$2      DJNZ    R1,$2       ;內迴圈
        DJNZ    R0,$1       ;外迴圈
        RET
```



[圖 1]　利用 AT2051 學習板執行上面程式的情況
P1.0 上的 LED 會不斷閃爍。

旗威科技有限公司　地址：高雄市三民區昌裕街 18-1 號　網址：http://www.chipware.com.tw
技術專線：07-395-5152　技術支援傳真：07-395-5155　E-mail：chipware@chipware.com.tw

這個程式裡有強制跳躍與條件式跳躍最典型的應用，您看出來了嗎？一個是 SJMP 的指令，另一個是 DJNZ，您可能會有這樣的疑問：為什麼要加一個 SJMP 在主程式的最後使其不斷循環？還有，為什麼 DJNZ 可以用來做時間的延遲呢？我們先跳開組合語言的思考模式，探討一下 8051 的硬體架構，接下來的文章要為您介紹 8051 內部運作的一些特色。

當 8051 的系統一開機時，程式的執行一定是從 0000H 的位址開始執行的，如果該晶片是未經程式化過的，那麼裡面的資料應該全部都是 FFH，換句話說，如果您直接把它通電想看它的動作，那麼它所執行的應該是不斷把 R7 的位置一直填上累加器的資料（由查表得知 FFH 為 MOV　R7,A），而 ACC 暫存器的預設值為 00H（由查表得知，該表詳列在基礎篇的附錄 F 中），因此除了 R7 這個暫存器會被不斷填為 00H 之外，應該是看不到任何動作的。問題來了，如果我們的在 R7 裡面存了很重要的數值，可是我們卻忘了在程式的最後加上 JMP 回到程式的起點，那會發生怎樣的事呢？

如果 R7 暫存器裡存放的只是單純的運算數據，那麼了不起應該只是運算的數據不對罷了；但如果 R7 暫存器裡所存放的是 P3 埠的運作指令，而外部所推動的是一個大型的工作平台，那後果就很嚴重了！當然這樣的設計應該還沒放到系統上就應該被測試修正了，這樣的比喻或許有點誇張，但只要不小心，很可能就會造成一些無法彌補的困擾。

回到指令的用法上，JMP 經常會用在程式的無窮迴圈，使系統回歸到某個狀態反覆地執行，以避免上述的狀況產生。執行 DJNZ 需要兩個機械週期的時間，因此在程式裡標示內迴圈 (DJNZ　R1,$2) 的地方會停留 256 x 2 = 512 個機械週期的時間，再加上外迴圈使其停留 (512+2) x 256 = 131,584 個機械週期的時間，假設一個機械週期是 1uS 的時間，那麼 Delay 的時間大約是 0.13S，如果需要 Delay 更久的時間，只要把迴圈數再增加就可以了。

終於把 JMP 的相關指令介紹完了，不過對於學習 Assembler 來說，這才剛剛起步而已，往後我們會陸陸續續把一些進階的用法跟大家做介紹，下次見囉！

旗威科技有限公司　地址：高雄市三民區昌裕街 18-1 號　網址：http://www.chipware.com.tw
技術專線：07-395-5152　技術支援傳真：07-395-5155　E-mail：chipware@chipware.com.tw