

常用指令排行榜(四)

上次我們所介紹的 JMP 相關指令，是屬於強制跳躍的指令，在這裡我們先稍作一下復習：

指令	性質	使用時機
SJMP	相對位址	256 Bytes 範圍內的跳躍
AJMP	絕對位址	2K Bytes 範圍的跳躍
LJMP	絕對位址	64K Bytes 範圍的長程跳躍

所謂強制跳躍，是不需經過任何判斷，只要執行到這個指令就一定要跳躍的指令，這在一般程式語言的寫法中，一向是很忌諱而儘量不去使用的指令，就像是 GOTO 一樣，這樣的指令往往很容易讓系統不小心陷入某個無窮迴圈中，而且抓不出問題點，然而在組合語言的領域裡，這樣的指令不但需要，而且十分重要，它可以有效節省你的程式空間，減少堆疊的運算，因為如果你用的是 CALL 之類的指令，你就必須騰出足夠的堆疊運算空間，否則很容易因堆疊不足造成系統不斷 RESET 或誤動作頻頻產生的情況。

可是單純的跳躍並不能解決所有的問題，特別是我們需要加入條件式的判斷時，因應而生的條件式跳躍指令就變成了不可不學的重要關鍵，接下來的文章，就要帶您深入條件式跳躍指令的領域。

條件式跳躍的指令很多，而且使用的頻率十分頻繁，它可以分為兩大部分來介紹，第一部分是屬於控制轉移類的指令，它包含 JZ、JNZ、CJNE、DJNZ 等指令；第二部分為布林位元處理指令，它包含 JC、JNC、JB、JNB、JBC 等指令。

控制轉移類的指令，其檢查比對的條件為一個 Byte 的資料，如果條件成立則進行跳躍的動作，如果不成立則略過該指令繼續執行下一個指令，下面是這些指令功能的詳細介紹：

JZ (Jump if Accumlator is Zero): 當累加器為 0 則進行跳躍的動作。

JNZ (Jump if Accumlator is Not Zero): 當累加器不為 0 則進行跳躍的動作。

CJNE (Compare Datas and Jump if Not Equal): 比較兩個暫存器或暫存器與累加器的值，若不相等則跳躍。

DJNZ (Decrement and Jump if Not Zero): 把指定的位址或暫存器的值減 1，若不為 0 則進行跳躍。

筆者將指令所對應的英文意思特別標示出來，用意是為了方便記憶，避免指令不小心用錯，因為所有條件式跳躍的指令都長得很像，可是功能卻不大相同。以上四個指令裡最常用的是 DJNZ，幾乎所有需要迴圈的地方都會用到它；其次是 CJNE，如果要進行資料的比對與除錯，這個指令就相當重要了。

接下來是布林位元處理類的指令，這個部分是 8051 最吸引人的地方，因為它可以對單一個 bit 進行控制，使 8051 單晶片在使用上能有更大的彈性，屬於這個部分的條件式跳躍指令有下列五種：

JC (Jump if Carry is set): 當進位旗標 C 的值為 1 時則進行跳躍的動作。

JNC (Jump if Carry is Not set): 當進位旗標 C 的值不為 1 時則進行跳躍的動作。

JB (Jump if direct Bit is set): 當指定 bit 的值為 1 時進行跳躍的動作。

JNB (Jump if direct Bit is Not set): 當指定 bit 的值不為 1 時進行跳躍的動作。

JBC (Jump if direct Bit is set & Clear it): 當指定 bit 的值為 1 時進行跳躍的動作，並將該 bit 清為 0。

進位旗標的條件判斷最常用在一般的運算上，凡是有藉位或溢位判斷的需要時，JC 與 JNC 就成了不可或缺的指令；而 JB 與 JNB 則用在一般輸出入的判斷上較多。

剛才介紹的所有指令中，判斷的過程都不會影響該 Byte 或該 bit 的值，除了 JBC 與 DJNZ 以外！JBC 除了具備判斷的功能外，還具有歸零的動作，它會影響該 bit 的值，這在使用上要特別小心；而 DJNZ 則是先將該 Byte 先進行減 1 的動作再做判斷，因此也會改變該 Byte 的值。

到此跳躍指令才算介紹完，您可能還會有疑問：不是還有 JMP @A+DPTR 的指令嗎？這個部分，等我們介紹到查表的用法時再來深入探討吧！