

第七章

中斷結構

- 7-1 何謂中斷
- 7-2 MCS-51 中斷介紹
- 7-3 MCS-51 中斷的致能與除能
- 7-4 中斷優先權



7-1 何謂中斷

所謂中斷是指當微處理機正在執行一個程式（例如主程式）時，忽然有信號來打斷目前的工作，這時 CPU 必須暫停原來執行的程式，跳去指定的另一個程式（例如中斷服務程式）執行，等到執行完指定的程式之後，再回去繼續執行原來的主程式。

一個完整的微處理機需要有很多的週邊裝置，以 MCS-51 為例，有內部的兩個計時／計數器、串列埠 UART，以及兩個外部中斷要求輸入腳。

採用中斷的方式，則能有效地管理眾多的週邊裝置。CPU 平時執行主程式，當某個週邊裝置需要服務的時候，才會發出中斷要求信號，告訴 CPU 暫時停止目前主程式的工作，去做週邊裝置的工作。

7-2 MCS-51 中斷介紹

MCS-51 提供了 5 個中斷源，在 MCS-51 中任何的中斷產生，都必須經下列三個步驟方才有效。

1. 設定中斷致能暫存器 IE 中的 EA 位元為 1。
2. 設定中斷致能暫存器 IE 中，相關的個別中斷位元為 1。（例如外部中斷的致能位元是 EX0）。
3. 中斷要求旗號提出中斷要求，也就是設定中斷要求旗標為“1”。

下列的表 7-1 是 MCS-51 的各個中斷源所屬的中斷要求旗標及中斷旗標所屬暫存器一覽表。

表 7-1

中斷來源	所屬暫存庫	中斷旗號
外部中斷 0 (INT0)	TCON. 1	IE0
時間中斷 0 (Timer 0)	TCON. 5	TF0
外部中斷 1 (INT1)	TCON. 3	IE1
時間中斷 1 (Timer 1)	TCON. 7	TF1
UART (RXD)	SCON. 0	RI
UART (TXD)	SCON. 1	TI

✧ 計時／計數器中斷

Timer 0 和 Timer 1 的中斷是由 TF1 旗號所產生的，這兩個旗號是在當它們各自的計時／計數器產生溢位時，會由內部自動設定成“1”，並提出中斷要求（Timer 1 在模式 3 時例外）。此時若計時／計數器的中斷已啟用（致能），則 CPU 會跳往自己本身的中斷向量位址處執行，例如：Timer 0 是 000BH，Timer 1 是 001BH，並且將旗號（TF0 或 TF1）清除成 0。

✧ 外部中斷

外部中斷 INT0 和 INT1，我們可以將其規劃成準備觸發動作（偵測輸入中斷信號是否為“Low”）或是規劃成負緣觸發“↓”動作，它是由 TCON 暫存器中之 IEO 及 IE1 兩位元，才是真正提出中斷要求的兩個旗號。當外部中斷發生時，IE0 或 IE1 或被設定為“1”，以便向 CPU 提出中斷要求。

採用準位觸發方式 IEX 會隨著 INTX 準位變化而隨時改變，也就是 $INTX = 0$ 則 $INTX = 1$ ；則 $IEX = 0$ 。因此當 CPU 未能即時處理 IEX 的中斷要求時，若此時 INTX 準位已改變則中斷要求信號將被遺漏。這是採用準位觸發式的缺點。

另外負緣觸發方式則只要檢知 INTX 上的信號由 1→0 時，中斷要求旗號 IEX 就被設定為 1，並且一直維持著 1，直到這個中斷要求被接受為止（去執行中斷服務程式），才會被清除為 0。因此，負緣觸發有記憶外部中斷要求信號的功能，而不會被遺漏。

✧ 串列埠 UART 的中斷

串列埠之中斷發生在 RI 和 TI 兩位元上，只要兩個旗號中有任何一個位元為 1 時，即可配合 IE 暫存器的 ES 位元來產生中斷致能，則 CPU 將會自動跳往串列埠中斷向量位址（0023H）處執行。要注意的是，CPU 在執行中斷服務程式後，並不會自動清除 RI 和 TI 為“0”，另外還要注意的是，一旦發生中斷跳至串列埠中斷服務程式執行後，由於發送及接收共用一個相量位址（0023H），因此在中斷副程式的開頭，就必須先用指令去判斷究竟是何者提出中斷要求的（判斷 RI 或 TI）。

7-3 MCS-51 中斷的致能與除能

我們可經由程式的設計，對於某些中斷的請求，使它失效（除能 Disable）或者有效（致能 Enable）。IE 中斷致能暫存器就是要用來做這個工作的，它包含有一個禁止所有中斷產生的位元 EA，它可以立即禁止所有中斷的產生（當 EA = 0 時，所有的中斷失效）。它也有控制個別中斷的位元旗標，下列是中斷致能暫存器 IE 的說明。

中斷致能暫存器（INTERRUPT ENABLE REGISTER）

表 7-2

IE	EA	—	ET2	ES	ET1	EX1	ET0	EX0
----	----	---	-----	----	-----	-----	-----	-----

方 法		方 法 說 明
EA	IE. 7	如果 EA=0 時，禁止所有中斷，如果 EA = 1，則各中斷是否被接受，由各自的中斷致能位元加以設定。
—	IE. 6	未使用；保留給將來使用
ET2	IE. 5	致能 Timer 2 溢位或補入之中斷（MCS-52）
ES	IE. 4	致能串埠之中斷
ET1	IE. 3	致能 Timer 1 之中斷
EX1	IE. 2	致能外部中斷 INT1 之中斷
ET0	IE. 1	致能 Timer 0 之中斷
EX0	IE. 0	致能外部中斷 INTO 之中斷

7-4 中斷優先權

MCS-51 的中斷優先權層次共有兩層，即高優先權（High Priority）及低優先權（Low Priority）。每一個中斷源都可經由 IP 暫存器中的位元，規劃為高優先權或低優先權，以設定整個程式的中斷優先權順序。若相對應的位元設定為“1”時，則在高優先權層次。中斷優先權暫存器的說明如下。

中斷優先權暫存器（INTERRUPT PRIORITY REGISTER）

表 7-3

IP	—	—	PT2	PS	PT1	PX1	PT0	PX0
----	---	---	-----	----	-----	-----	-----	-----

方法		方法說明
—	IP. 7	未使用；保留給將來使用
—	IP. 6	未使用；保留給將來使用
PT2	IP. 5	定義 Timer2 之優先權層次（MCS-52）
PS	IP. 4	定義串列埠之優先權層次
PT1	IP. 3	定義 Timer 1 之優先權層次
PX1	IP. 2	定義 INT1 之優先權層次
PT0	IP. 1	定義 Timer0 之優先權層次
PX0	IP. 0	定義 INT0 之優先權層次

一個低層次的中斷源可以被高層次的中斷源中斷，但是不會被同層次的中斷源所中斷，而一個高層次的中斷源不會被任何其它的中斷源所中斷。如果有兩個不同層次中斷源同時要求中斷，此時高層次優先權之中斷源優先被執行。如果兩個相同優先權層次的中斷源，同時提出中斷要求時，則 CPU 將依據內定之輪詢順序（Polling）安排，以決定何者中斷源被接受執行。如果程式設定並沒有設定各個中斷源的優先順序，CPU 即自動以內定中斷優先權來排定中斷優先。

一旦中斷被確認後，硬體會產生一長呼叫“LCALL”動作，CPU 也會把目前兩個 Bytes 程式計數器 PC 推入（PUSH）到堆疊區存放，然後載入中斷源之中斷向量位址，並跳到此位址執行中斷副程式。一直要等到執行 REAT 指令後，才表示中斷結束，再從堆疊區中取回（POP）兩個 Bytes 的返回位址，到程式計數器 PC 中，回到原來主程式被中斷的位址繼續執行。



筆記欄

