

輸入/輸出界面

15/10/2007

前一章，我們已介紹了數種選取與辨認輸入 / 輸出設備的方式，目前，該是實際組成輸入 / 輸出口的時候了。這一章，我們將探討一些使輸入 / 輸出設備能將一位元組之資料送給計算機，以及能接收計算機所送出之位元組資料的實際巴士界面技巧。正如設備選取電路一樣，同樣有許多電路可構成輸入口與輸出口。不過，我們將僅舉少數幾個例子，說明界面的基本原理。

3-1 輸出口

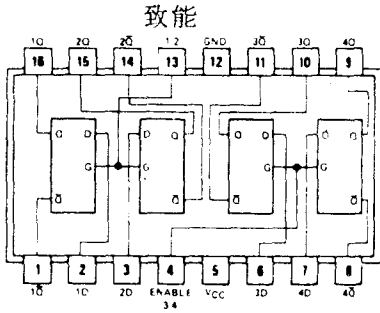
所謂輸出口即為在 BASIC 語言程式之 POKE 命令的控制下，接收計算機所送出之資料位元組的設備或元件。您已經明瞭，在 POKE 指令執行時，巴士上所傳送之資料—— \overline{WR} 脈衝與設備位址——間有一定的時序關係。這在圖 1-4 時已說過。在 Apple 計算機上， \overline{WR} 脈衝的期間大約為 500ns。因此，若我們以 \overline{WR} 脈衝控制將資料由資料巴士傳給輸出

APPLE 界面實驗

設備，透過使用設備選取脈衝，則資料呈現給輸出設備的時間將僅約500ns左右。在這麼短的期間內，接收（亦即，輸出）設備誠難有所作為（亦即，安穩地讀取資料）。為了解決這個問題，每一輸出口都必須備有可自巴士獲取資料，並將資料“保存”至被輸出設備拿走或被另一次資料傳輸“更新”（改變之意）為止的某種電路。

可達成此一功能的電路即稱為鎖住器（latch）。鎖住器會栓住資訊，將之保存至被更新或電源停掉為止。鎖住器 IC 有許多種不同型式，每一種型式均具有不同的控制結構以及資料輸入輸出。本章，我們將介紹其中三種最一般性的：SN7475，SN74175，與 SN74LS373，而不每一種都介紹。圖3-1所示即為這三種鎖住器之接腳圖與功能表。注意，雖然SN7475與SN74LS373都為真正的鎖住器元件，但SN74175實際上則含正反器（flip-flop）。由於SN7475鎖住器晶片內含四個鎖住器電路，且SN74175內含四個正反器電路，因此，組成一個八位元輸出口需有兩個SN7475或兩個SN74175 IC。但SN74LS373由於每一個晶片就含八個鎖住器電路，因此，構成一個八位元輸出口僅需一個晶片就夠了。

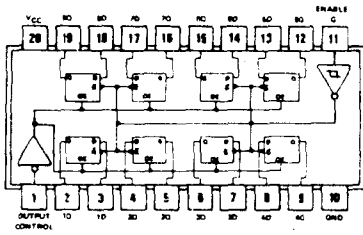
現在，我們先簡單地介紹一下這幾個鎖住器電路的動作原理，好讓您在毫無問題的使用。就以SN7475作例子。您可將SN7475鎖住器想成是“記憶閘”。這點由圖3-1之SN7475的功能表中即可看出。仔細看這個功能表，您會發現，當致能輸入(G)為邏輯1時，“D”輸入所呈現的資料（或邏輯準位）直接通過鎖住器到達“Q”輸出。 \bar{Q} 輸出則為Q



功能表
(每一鎖住器)

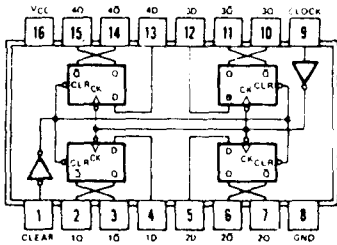
輸入		輸出	
D	G	Q	\bar{Q}
L	H	L	H
H	H	H	L
X	L	Q_0	\bar{Q}_0

H = 高準位 L = 低準位 X = 不定
 Q_0 = 鎖住前之輸出準位



LS373, S373
功能表

輸出控制	鎖住 G	D	輸出
L	H	H	H
L	H	L	L
L	L	X	Q_0
H	X	X	Z



功能表
(每一個正反器)

輸入		輸出	
清除	時序 D	Q	\bar{Q}
L	X	X	L H
H	↑	H	H L
H	↑	L	L H
H	L	X	Q_0 \bar{Q}_0

圖 3-1 SN7475(上), SN74LS373(中), 與 SN74175(下)
鎖住器晶片之接腳圖與功能表

輸出之反態。而當致能輸入由邏輯 1 變至邏輯 0 時，此時 D 輸入所呈現的準位則為 Q 與 \bar{Q} 輸出所記住或鎖住。圖 3-2 所

示的時序圖正好說明了這些動作。

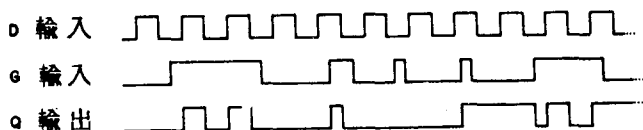


圖 3-2 SN7475鎖住器電路的時序圖

每當“G”輸入變為邏輯1準位，Q輸出即變成“D”輸入之狀態，不論“D”輸入之準位是否正在改變。換言之，當“G”輸入為邏輯1時，“D”輸入之邏輯準位傳遞至“Q”輸出；而當“G”輸入變為邏輯0時，“Q”輸出仍然維持於原來“D”輸入的準位不變。這就是SN7475鎖住器電路的動作原理。SN7475分成兩部份，每一部份均個別獨立動作，彼此互不相干。四個鎖住器電路欲串聯一起動作時，晶片上的兩個閘極輸入可接在一起。當然，幾個鎖住器的輸入與輸出彼此還是互不相干，如此，電路內就有四個不同來源的輸入信號。不過，若彼此欲成縱列動作，則全部四個輸入都必須在同一時間鎖住。

SN74LS373的動作原理與SN7475類同，但其却僅具有一個閘控或致能信號，此外，這個晶片亦僅有Q輸出，而無 \bar{Q} 輸出。SN74LS373另外具有一個輸出控制，不過，當SN74LS373被用作輸出口時，這個控制信號（第1支接腳）通常接地。

SN74175 晶片則含有四個正反器，這四個正反器獲取

並持住於時序脈衝正向緣 (positive-going edge) 時出現於輸入端的資訊 (邏輯準位) 。四個正反器的輸出唯有在這個時候——亦即，時序脈衝正向緣時——才會改變。其它時間，不論時序信號為邏輯 0 或邏輯 1 準位，SN74175 的輸入均不致影響輸出。此即此一正反器元件與其它之鎖住器元件的不同所在。雖然如此，但兩種晶片在計算機界面上的功用則無異。

為了能“清除”正反器的狀態 ($Q = 0, \bar{Q} = 1$)，SN74175 晶片上通常有一個共用的清除輸入 (第 1 支接腳)。每當這個輸入加邏輯 0 準位時，每一個正反器即呈清除狀態 ($Q = 0, \bar{Q} = 1$)。平常，這個清除輸入均接 + 5 伏特 (邏輯 1) 不用。

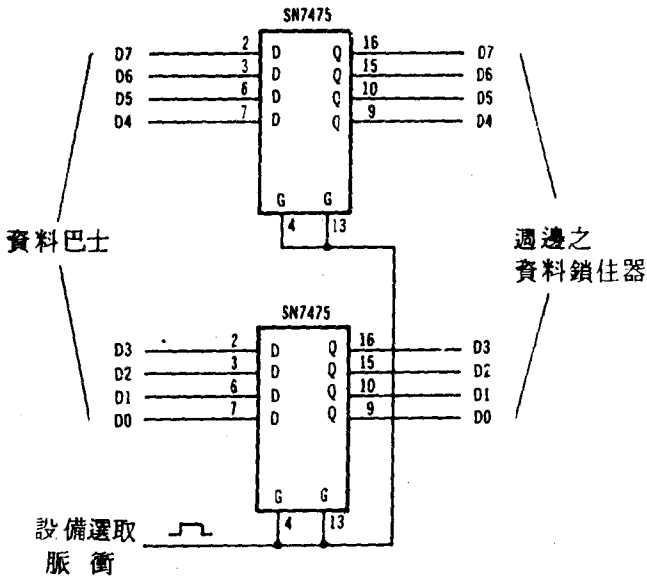


圖 3-3 以兩個 SN7475 鎖住器晶片組成一個輸出口

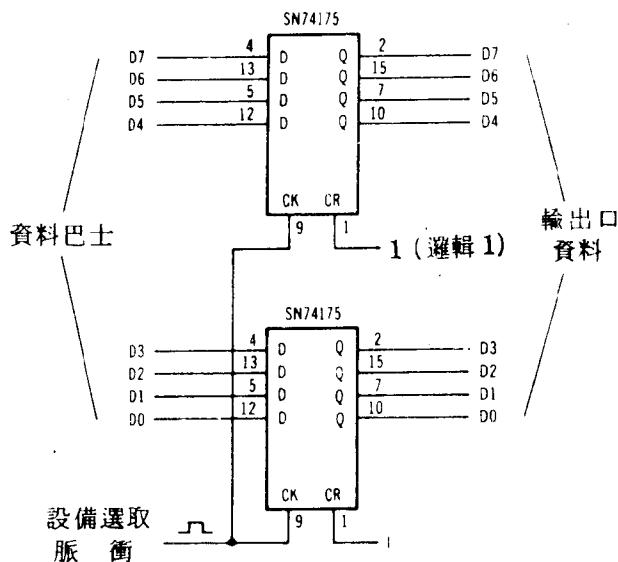


圖 3-4 以兩個 SN74175 鎖住器晶片組成一個輸出口

在執行 POKE 命令時，這三個積體電路之任一者均可用以鎖住 Apple 計算機所輸出之資料。一經正確接在巴士上後，以一輸出設備選取脈衝推動此一鎖住器電路就不是什麼困難的事了。圖 3-3 所示即為一典型的八位元輸出口。在這個電路內，欲教鎖住器電路捕獲且持住 Apple 所輸出的資訊時，您必須加上一準位為邏輯 1 的輸出設備選取脈衝。

圖 3-4 所示則為以兩個 SN74175 鎖住器晶片構成輸出口的情形，這個電路以某種邏輯監視器顯示出晶片所鎖住的資訊。輸出口清除輸入接腳處的“1”代表這些輸入均連至 +5 伏特，或邏輯 1 準位。我們特別以這個“1”代表邏輯準位連接，以有別於電源供給連接——接至電源供給時通常寫成 +5 伏特或 +5V。

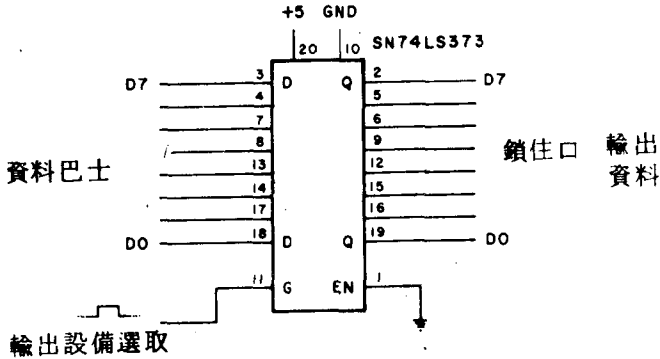


圖 3-5 以 SN74LS373 鎖住器晶片構成一輸出口

圖 3-5 所示則為以一 SN74LS373 鎖住器晶片（內含八個鎖住器電路）構成一個八位元輸出口的情形。使用這種 IC 時，只要一個晶片即可構成一個輸出口。為了使輸出永遠保持致能，輸出控制線（EN）特別接地。同樣地，設備選取電路必須供應一個邏輯 1 準位的輸出設備選取脈衝。這個輸出口一旦正確連接至資料巴士，而且設備選取脈衝亦有來源後，其即可在軟體命令的控制下開始使用。舉個例子而言，POKE 49312, 0 命令執行的結果，計算機將一為零的數值送至位址 49312 的輸出口。若事實上有一個輸出口連接至資料巴士，而且其位址為 49312，則這個零就會出現在這個輸出口上。

例題 3-1 49320 輸出口上之八位元二進計數程式

```

10 FOR N = 0 TO 255
20 POKE 49320,N
30 NEXT N
40 GOTO 10
    
```

例題 3-1 的程式即可用以在位址 49320 的輸出口上產生一逐次累增的二進計數。整個計數將循 255, 0, 1, 2, ……… 254, 255, 0, 1 ……… 等等的順序，綿延不斷。稍後在作實驗時，您會再度看到這個程式。

由以上的敘述讀者可看出，輸出口極易構成。絕大多數並行輸入且並行輸出的邏輯元件，只要具有內部鎖住能力，都可用作鎖住器。舉個例子而言，SN 74193 可規劃二進計數器，SN 74LS 194 A 萬用移位暫存器，SN 74198 移位暫存器等等，就都可用作鎖住器。

雖然絕大多數的輸出口均能輕易地以標準規格的積體電路構成，但最近市面上也出現了一些指明專供微電腦使用的新型積體電路，這些電路內部均具有鎖住的功能。內含一個鎖住器之 Signetics NE 5018 八位元數位至類比轉換器晶片就是其中一個例子。

輸出口的典型應用包括下列幾些：

送出資料給印字機

送出資料給顯示幕

控制紅綠燈

送出資料給軟性磁碟

推動模型鐵道之開關

控制一化學程序中之閘門與唧筒

控制繪圖機

送出資料給七段顯示器

控制另一部計算機

有些應用實際使用資訊的數值，有些則僅個別使用每一位元之狀態（開或關，即 1 或 0）。但像印字機之類的設備

則可能兩種都用：既有傳送欲印出資料的口，亦有控制印字機之動作的口。七段 LED 所構成的顯示器，雖然僅算一種“設備”，但其通常必須使用數個輸出口。

3-2 輸入口

輸入 / 輸出設備使用輸入口，以便將資訊傳給計算機。不像輸出口必須能接收並持住計算機於某一時刻置於資料巴士上的資訊，並且必須隨時與資料巴士接在一起，輸入口在不用時，必須能與巴士“斷接”。雖然輸入口必須將邏輯 0 與邏輯 1 送給 CPU，但其却必須配置成當其不被選取時，不致於干擾巴士之使用的情況。

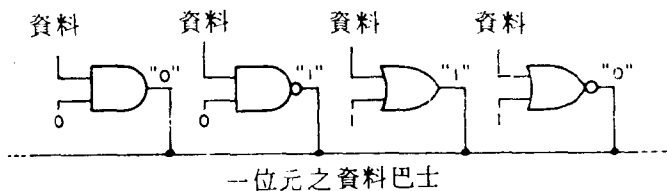


圖 3-6 試著將標準邏輯閘直接連在巴士上

如圖 3-6 所示，由於“未選取”之輸出狀態可能為邏輯 0，亦可能為邏輯 1，因此，我們無法直接以簡單的邏輯閘將資料“置於”資料巴士上。注意到，即使沒有任一個閘被選取或致能，正如以引號括起之邏輯準位所示的，每一邏輯閘之輸出準位還是各不相同。這些準位彼此會“爭著”使用巴士，因而很可能導致燒壞某一個或某幾個晶片。此即清楚

地說明了爲何邏輯閘不能直接用在資料巴士上的原因。

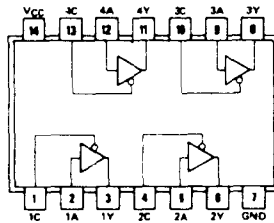


圖 3 - 7 SN74125巴士緩衝器晶片之接腳圖

具有三態 (three-state or tri-state) 輸出的特殊積體電路則可解決此一問題。典型的三態元件爲如圖 3-7 所示的 SN74125 巴士緩衝器。這個晶片內含四個完全一模樣的緩衝器。每一個緩衝器 (輸出準位等於輸入準位) 均具有一額外控制線, 在符號上正好連接至三角形符號的一邊。緩衝器的動作情形是, 在致能時, 輸入端之邏輯準位原樣傳遞至輸出端。不過, 與一般簡單邏輯閘不同的, 當被禁能 (disabled, 亦即, 不動作) 時, 緩衝器之輸出端與巴士 (或其所連接的設備) 形成“斷接”。這就是三態元件的所謂第三態 (另兩種狀態分別爲邏輯 0 與邏輯 1), 這種狀態經常稱爲高阻抗狀態 (HI-Z)。三態元件的連接與斷接速度相當快, 一般均在 20 ns 以內。

在 SN74125 內, 每一個緩衝器均有其各自的致能輸入。在資料欲由輸入傳遞至輸出時, 緩衝器的致能輸入必須加邏輯 0。若致能輸入加邏輯 1, 緩衝器之輸出即呈高阻抗 (或斷接) 狀態。另一個類似的電路 SN74126, 接腳則與 SN74125 完全相同。不過, 其致禁能控制則正好相反——

致能輸入加邏輯 1 時，緩衝器致能，加邏輯 0 時緩衝器禁能（呈高阻抗狀態）。我們舉這幾個晶片主要在說明三態元件的動作原理。這些晶片在實際計算機界面電路上很少見，因為，還有其它更好用的元件存在。

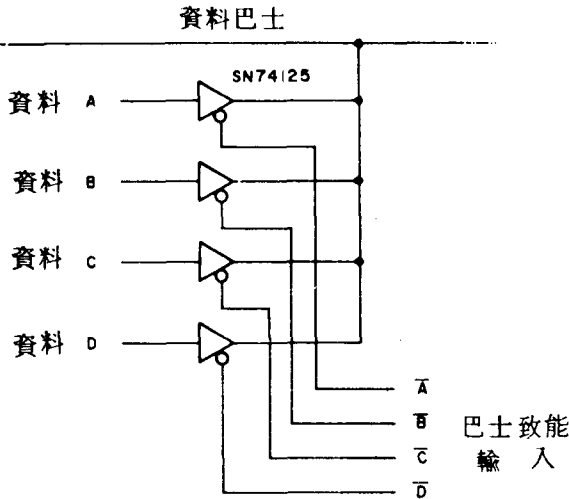


圖 3-8 典型三態巴士，四條線

表 3-1 四線之三態巴士的真值表

致 能				巴 士 內 含
D	C	B	A	
1	1	1	1	無法決定 (所有設備均 HI-Z) 資料 A 資料 B 資料 C 資料 D 不允許
1	1	1	0	
1	1	0	1	
1	0	1	1	
0	1	1	1	
0	0	1	1	
0	0	0	0	

我們舉一個典型的巴士來作說明。於圖 3-8 之電路內，巴士上接了四個一位元的設備。雖然八位元的系統總共需有八條線，但爲了清楚起見，我們僅畫出了單位元的巴士。當四個巴士致能輸入中有任一者加邏輯 0 準位時，其所對應的資料輸入即會通過緩衝器，到達巴士上。我們假設沒有其它設備連接在巴士上，因此，表 3-1 所示的真值表正好適用於這個簡單的巴士電路。

倘若四個緩衝器均未被致能或接至巴士，則除了邏輯閘輸入、記憶器等資料位元的“接收者”外，巴士並未連接至任何東西上，因此，巴士的邏輯值未知。而只要巴士緩衝器之致能輸入有任一者加邏輯 0，這個被選取的緩衝器即會將其資料送至巴士上。注意，不能同時有一個以上的緩衝器被致能，因爲，這樣會造成巴士衝突。

Apple 計算機系統中，所有用以將資訊傳給 CPU 的設備均必須具有三態輸出。是以，即令記憶器亦須具有三態輸出（事實上亦然）。此時，計算機的設計者就必須負責保證，系統中不會同時有兩個或兩個以上的輸入設備被選取。否則，若發生這種複選的情形，計算機即無法正常的作業。

由此可見，用以將資訊傳給計算機的輸入口可輕易地以標準的三態 IC 構成。大部份情況下，我們都使用八個個別的三態緩衝器，每一條線一個。同樣地，大多數情況下，致能輸入亦都並行連接，以使八個緩衝器能將資訊同時送出至巴士上。這個共用的致能輸入有時會做在晶片內，以期僅以一支接腳控制到全部八個位元。

雖然有許多晶片都可用以組成輸入口，但其中僅有少數能一般化至符合我們的考慮。我們將使用的兩個主要積體電

路是 SN74365 與 SN74LS244。SN74365亦等於DM8095 (國際半導體公司產品)。這兩個晶片的接腳圖如圖 3-9 所示。

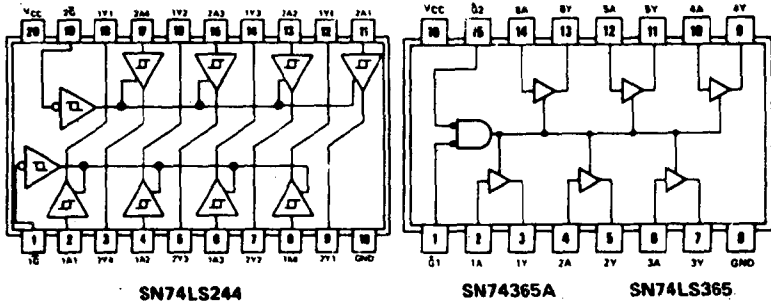


圖 3-9 SN74LS244與SN74365(DM8095)

三態巴士推動器晶片的接腳圖

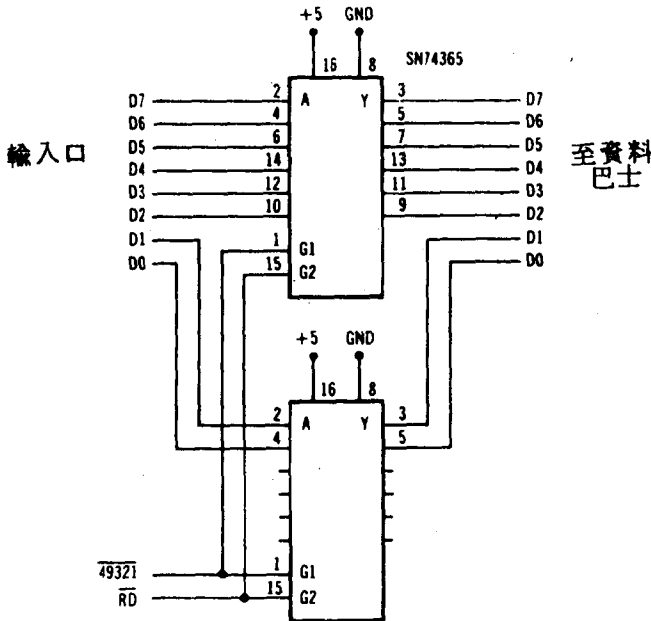


圖 3-10 以SN74365晶片構成的典型輸入口

迅速您會發現，SN74LS244 晶片上含有八個三態緩衝器，而SN74365 晶片上則僅有六個。因此，若我們以SN74365 晶片構成輸入口，我們就必須使用兩個 IC。圖3-10 所示即為一典型的八位元輸入口。就這個構造而言，下面的SN74365 晶片則僅用了兩個三態緩衝器。由於SN74365 內含有控制三態緩衝器之致能的NOR 閘，因此，我們特以之閘控 \overline{RD} 功能脈衝與設備位址 $\overline{49321}$ 。倘若設備選取信號 \overline{RD} 49321 已在界面電路的其它地方產生，則您可將這個信號加至兩個晶片上之其中一個致能輸入，然後將另一個致能輸入接地，或接邏輯 0。這種控制方法即如圖3-11 所示。

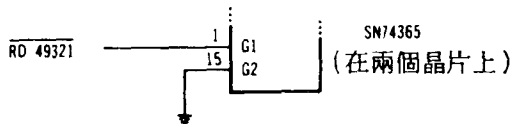


圖 3-11 SN74365三態晶片的另一種控制方法

如例題 3-2 所示，使用這樣一個輸入口，我們可以PEEK 命令將資料值輸入至計算機。

例題 3-2 49321口之資料輸入程式

```
10 A = PEEK (49321)
20 PRINT A
30 GOTO 10
```

就這個例子而言，當 10 號一列之 PEEK 命令輸入一個

八位元的二進值時，這個數值即被轉換成一介於 0 至 255 之間的十進數，然後“印出”在顯示幕上。其功用與下面述句所達成者完全相同：

```
10 PRINT PEEK(49321): GOTO 10
```

使用具有八個（八位元）緩衝器之 SN74LS244 晶片亦可組成一類似的輸入口。這個晶片包含兩組互相獨立的緩衝器，每一組各有四個緩衝器。這兩組緩衝器又分別由兩個致能輸入， $\overline{2G}$ 與 $\overline{1G}$ 所控制。由於 SN74LS244 無內附的 NOR 閘，故設備選取閘控必須設在外面。圖 3-12 所示即為一以

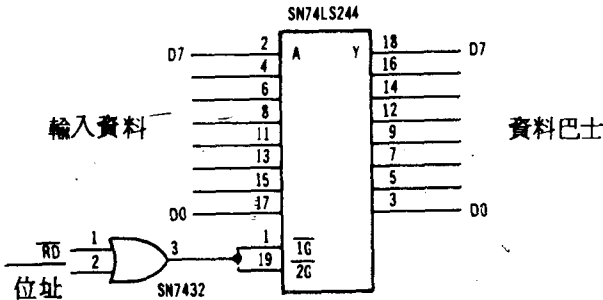


圖 3-12 以 SN74LS244 構成的輸入口

SN74LS244 晶片所構成的典型輸入口。例題 3-2 所示的程式同樣可用於控制將資訊由這個輸入口輸入至計算機。

SN74365 與 SN74LS244 兩者亦皆有接腳完全一一對應，但資料位元通過晶片時却被反相再置於資料巴士上的類

似晶片。這兩個晶片分別是 SN74366 與 SN74LS240。此外，SN74366 亦對等於 DN8096 晶片。一般而言，界面電路中所使用的緩衝器均為不反相緩衝器。

偶而，週邊設備所產生之資訊會超過 8 位元，而這些位元必須都為計算機所讀取。像 12 位元的類比至數位轉換器就是其中一個例子。在這種輸入資訊超過八位元的情況下，我們必須將輸入資訊以八位元為單位分組。就前述之 12 位元轉換器而言，輸入資訊必須分成兩組，一組含八位元，另一組含四位元。同樣地，十六位元值亦需使用兩個輸入口，九位元值亦然。在輸入口之八個位元並非全部使用的情況下，未用的位元通常都接地，使其變為邏輯 0 狀態。倘若這些未用位元的狀態無法決定，則很可能這些位元根本就不做在輸入口電路內。只要適當地使用軟體命令，您同樣可“除去”這些位元。這時，您只要以命令“遮掉”(mask)這些未用位元，使其全變為 0 就可以了。

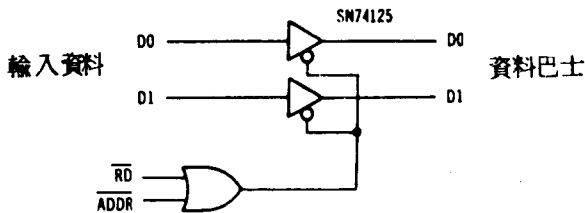


圖 3-13 兩個位元之輸入口

至此，輸入資訊超過八位元的輸入口結構都已定義過了。接著，讓我們來看看如何將自兩個輸入口(圖 3-13)所

輸入的兩個資料位元組重新組合，還原成原來的數值。最低次的八位元由於 Apple 計算機在輸入的過程中即已自動將之轉換成一值介於 0 至 255 之間的十進數，因此，這八個位元沒問題（其值本來就等於一介於 0 至 255 之間的十進數）。不過，較高次的四位元（就 12 位元轉換器的例子而言）就有問題了。這四個位元若單獨考慮，則其將被轉換成一介於 0 至 15 間的十進數，而不是其原有的位置值 256, 512, 等等。不過，很明顯的，由於這四個位元是一個十二位元之資料值的高次四位元，因此，其值剛好差一個 256 的因子。記得，Apple 所輸入的任何八位元值，均被自動轉換成一介於 0 至 255 之間的十進數。

根據以上的討論，Apple 所輸入的兩個數值就很容易能重組成原來的資料。我們只要將高次四位元的值乘以 256，然後再加上低次八位元轉換所得的值，即可獲得原來界面設備所產生的十二位元資料值——為一介於 0 至 4095 間的數值。達成此一作業的完整軟體常式即如例題 3-3 所示。

例題 3-3 12位元輸入的轉換程式

```

10 A = PEEK(49312)
20 B = PEEK(49313)
30 C = (B * 256) + A
40 PRINT C

```

若將所有步驟都寫在一列上，則這個程式就變成下面的單獨一列：