

# Apple 界面

55X04 | 2602

至此，您可能急欲知道下列幾點：

1. Apple 實際如何將資訊傳至輸入 / 輸出設備？
2. 輸入 / 輸出設備實際如何與計算機之運算取得同步？
3. 每一個別的輸入 / 輸出設備如何辨認或選取？
4. 輸入 / 輸出設備如何將其資料置於資料巴士上，且其實際如何由資料巴士獲取資料？

由於了解這幾個問題您就能了解到微電腦界面的基礎，因此，這幾個問題十分重要。這些問題將在本章以及其它章回答。為了加深您的印象，我們亦設計了一些實驗，讓您能親自動手做做。

本章將介紹一些數位電路的例子。作者假設讀者已能看懂邏輯電路圖符號，並且熟悉一些較平常的 SN 7400 系列之 TTL 電路。

## 2 - 1 輸入/輸出設備之位址解碼

在能開始討論輸入 / 輸出設備與計算機間之實際資訊傳輸前，我們必須先了解辨認或選取每一個別輸入 / 輸出設備的電路與信號。當然，選取輸入 / 輸出設備有許多種方法，我們將探討其中數種方法。不過，由於個別差異與特殊狀況存在，因此，我們並不能每一種方法都作介紹。

當您以兩個一般用途之輸入 / 輸出命令—— PEEK 與 POKE ——其中之一，寫程式教計算機做資訊傳輸時，6502 處理器一定會產生某些信號，使資料的流動獲得同步。此時此地，我們主要關心的就是位址巴士線的用途。6502 有 16 條位址巴士線可選取記憶位置或輸入 / 輸出設備。您應還記得，PEEK 或 POKE 指令各自都含有一用以辨認被選定之記憶位置或輸入 / 輸出設備的十進位址。當然，Apple 計算機本身無法區別記憶位置與輸入 / 輸出口。

## 2 - 2 設備選取

計算機所使用的每一輸入 / 輸出設備都必須能認得其自己的設備位址。由於 PEEK 與 POKE 命令均使用 16 位元之位址，每一輸入 / 輸出設備必須隨時監聽（注意）著這 16 條位址線（ A15 ~ A0 ），看看有無自己的位址出現（

亦即，自己有無被選取）。輸入 / 輸出設備之電路有三種方法可監聽位址巴士上是否出現某一個特定位址：

1. 閘控 ( gating ) —— 測知許多邏輯信號的某一種特定組合。
2. 解碼 ( decoding ) —— 這是一種更具彈性的閘控方法，其可測知數個位址。
3. 比較 ( comparing ) —— 將位址巴士信號與一已知位址相比，直至相同為止。

當然，您亦可使用這三種技巧之組合，同時，以上這每一種方法亦皆有許多變化。接著，我們將分別舉例說明以上所列這三種基本位址解碼方法。

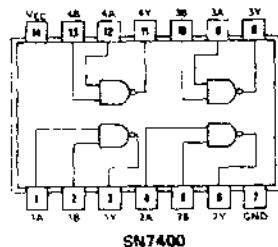
## 2 - 2 - 1 以邏輯閘作位址解碼

在以邏輯閘作位址解碼時，位址必須事先已知，您才能知道邏輯閘要如何組合。這裡，我們以設備位址

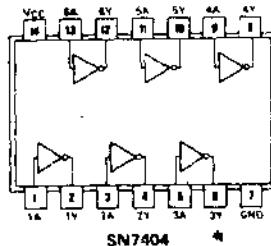
$101010001110111_2$  ( $43255_{10}$ ) 作例子。由於二進寫法顯得冗長且繁瑣，因此，您或許覺得寫成十六進制 A8F7H 較好一點。此外，由於NAND / AND 閘乃最基本的邏輯閘，因此，我們將以這種電路組成我們所要的解碼電路。

圖 2 - 1 所示即為數種 AND / NAND 閘的接腳圖，且表 2 - 1 為雙輸入 AND 閘與對等 NAND 閘的真值表。由於諸如 SN 7404 之反相器在設備解碼電路上亦很常用，因此，圖 2 - 1 與表 2 - 1 亦分別列出了這種邏輯閘的晶片接脚

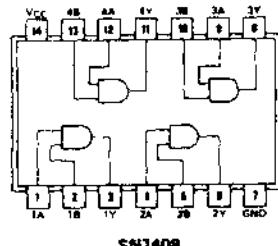
## APPLE 界面實驗



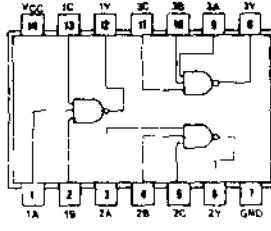
SN7400



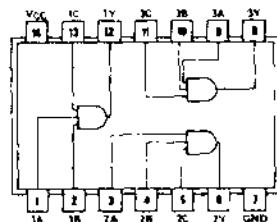
SN7404



SN7408



SN7410



SN74H11

圖 2-1 反相器與各種 AND/NAND 閘的接腳圖

表 2-1 雙輸入 AND 閘與 NAND 閘，以及反相器的真值表

AND 閘		NAND 閘		反相器	
輸入	輸出	輸入	輸出	輸入	輸出
A 0 0	Q 0	A 0 0	Q 1	A 0	Q 1
0 1	0	0 1	1	0	0
1 0	0	1 0	1	1	0
1 1	1	1 1	0		

圖與真值表。無論如何，邏輯 1 均代表高電位（+2.8 至 +5 伏特），而邏輯 0 均代表低電位（0.0 至 0.8 伏特）。NAND 閘功能有 2、3、4、8、與 13 個輸入等多種，而 AND 閘則有 2、3、與 4 個輸入等三種。

由於當所有輸入均為 1 時，AND 閘才輸出邏輯 1，NAND 閘才輸出邏輯 0，因此，我們必須配組 16 位元的二進位址  $1010100011110111_2$ ，使其在 AND 或 NAND 閘的輸入處均能為 1。顯然地，為達此目的，我們必須令輸入值為 0 的位址位元先通過一反相器，然後再加至 AND 或 NAND 閘的輸入。此外，由於沒有 16 個輸入之 AND 或 NAND 閘，因此，我們使用兩個八輸入的 NAND 閘；其中一個八輸入的 NAND 閘解碼上半位址巴士（A15 ~ A8），另一個解碼下半位址巴士（A7 ~ A0）。然後，這兩個 NAND 閘的輸出結果必須“合”（AND）起來，是以，我們將這兩個八輸入之 NAND 閘的輸出，一起加至一雙輸入的 NAND 閘上。由於 NAND 閘在輸出前將結果反相了，因此，我們分別又在兩個八輸入的 NAND 閘輸出處加了一個反相器。這樣的電路結構即如圖 2 - 2 所示。當，而且唯有，所有 16 個輸入呈  $1010100011110111_2$  組合時，這個邏輯閘電路才輸出“邏輯 0”，否則，其它任何時刻（亦即，16 位元輸入之其它組合），此一邏輯閘電路均輸出邏輯 1。因此，在將 16 個輸入分別接至其所對應的位址巴士線後，只要這個邏輯閘電路輸出邏輯 0，我們即可馬上知道，現在計算機所送出的位址是  $1010100011110111_2$ ，這個設備應被選取。

## APPLE 界面實驗

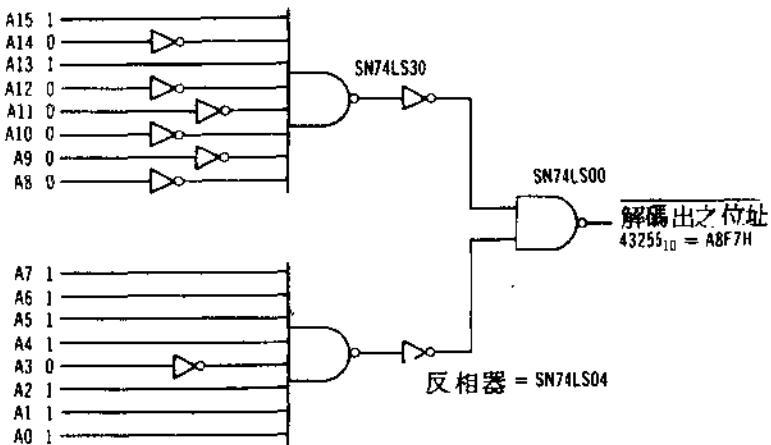


圖 2-2 解碼位址 43255 (= A8F7H) 的邏輯閘電路

這個電路的缺點是，在抵達最後雙輸入之 NAND 閘的輸出前，有些位址信號必須經過四級邏輯閘。由於信號在通過邏輯閘時均會產生時間延遲，因此，這就可能造成某些時序 ( timing ) 問題。不過，由於這個時間延遲實際上非常小，因此，目前我們將暫時忽略它。倘若延遲時間欲再進一步減少，則兩個八輸入之 NAND 閘的輸出可以一 NOR 或 OR 閘加以組合，而不用 NAND 閘。這種方法非常好。NOR 與 OR 閘到處都可買得到，而且在計算機界面上亦用得很廣泛。圖 2-3 所示即為典型 NOR 與 OR 閘的情形，表 2-2 所示則為其真值表。

圖 2-2 所示之邏輯電路圖雖然在解碼單一個位址上很有效，而且耗費便宜，但其却沒有彈性。較有彈性的方法可如圖 2-4 所示。這個電路圖的優點是，每一位址位元是否

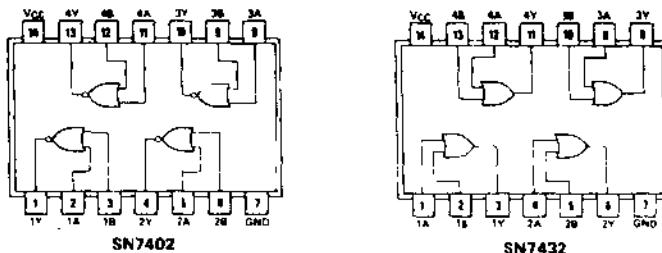


圖 2-3 典型 NOR 與 OR 閘 IC 的接腳圖

表 2-2 雙輸入 NOR 閘與 OR 閘的真值表

NOR 閘			OR 閘		
輸入		輸出	輸入		輸出
A	B	Q	A	B	Q
0	0	1	0	0	0
0	1	0	0	1	1
1	0	0	1	0	1
1	1	0	1	1	1

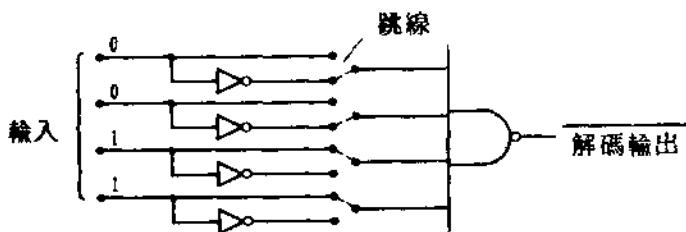


圖 2-4 可以解碼多種不同輸入的四輸入邏輯電路

先通過反相器可以選擇。這樣一來，正如圖 2-5 所示的，整個解碼之邏輯電路就不僅能解碼一個位址。譬如，就圖 2-5 而言，低次的八個位址位元輸入就可為  $01101110_2$ 。若

加上對應的上半部，則這麼樣的一個電路就可解碼 65536 個可能位址中的任一個位址。（當然，每一次僅能一個位址。）

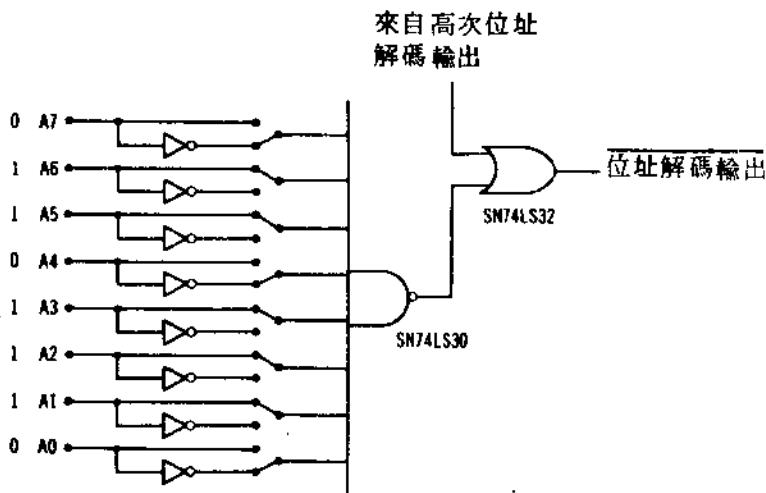


圖 2-5 以可規劃的邏輯閘作設備位址解碼（高次位址部份亦同）

這個可規劃的邏輯閘電路就具有相當的彈性，輸入位址可應界面的需求輕易改變，不過，這種電路僅能選取唯一的一個位址，而且這是一個很嚴重的限制。當同一電路板上同時有數個輸入 / 輸出設備時，每一設備就必須有一各自的位址解碼電路。這種限制可以其它的選取方法加以克服。

很遺憾的，剛剛所示的閘控方式並不足以獨特選取與控制一個輸入 / 輸出設備。您應記得，在前一章討論讀寫 (R /  $\bar{W}$ ) 信號時，我們曾說，R /  $\bar{W}$  信號用以同步來回計算機之資訊流動。因此，若欲正確地使用資料巴士，則輸入 / 輸

出設備亦須使用這個控制信號。在許多 6502 計算機系統的界面上， $R / \bar{W}$  線均負責產生一邏輯 0 的寫入脈衝，而讀取脈衝則另外將  $R / \bar{W}$  信號反相獲得。由於兩個都以邏輯 0 為動作狀態，因此，這樣所產生的兩個控制信號——WRITE ( $\bar{WR}$ ) 與 READ ( $\bar{RD}$ )——在界面電路上就很容易使用。圖 2-6 所示即為這兩個信號的用法。在這個電路內，16 位元閘控電路的輸出分別與  $\bar{RD}$  及  $\bar{WR}$  信號組合，產生兩個控制輸入 / 輸出口的信號。這兩個控制信號分別是經解碼之位址與 WRITE 脈衝的組合，以及經解碼之位址與 READ 脈衝的組合。這每一個閘所產生的輸出脈衝即稱為位址選取脈衝 (address select pulse) 或設備選取脈衝。

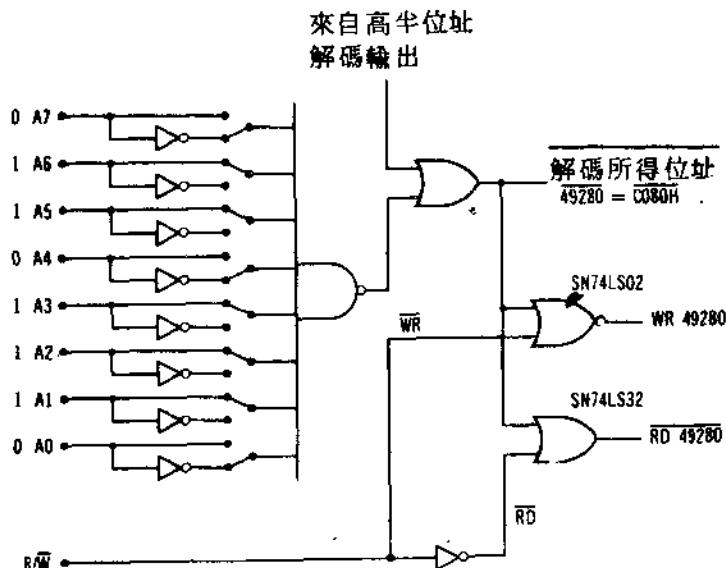


圖 2-6 以  $\bar{RD}$  與  $\bar{WR}$  信號產生設備選取脈衝，以求得同步。

( device select pulse )。說得更通化一點，經解碼之位址受功能脈衝 ( RD 或 WR ) 之控制，產生一設備選取脈衝。在圖 2-6 之電路內，RD 49280 脈衝即可用以控制一輸入口，而 WR 49280 脈衝即可用以控制一輸出口。注意到，WR 49280 上面沒有加一“短槓”。因此，這個控制信號是邏輯 1 時為動作準位，而 RD 49280 則為邏輯 0 時動作準位。注意，若寫成十六進制，則這個例子之輸入 / 輸出口即為 C080H 。

在進一步討論之前，您應了解，所謂讀取作業即將資訊自某一輸入口讀入計算機，而寫入作業即將資訊自計算機送出給某一外部設備。同時，以一個位址控制一個輸入口與一個輸出口亦是相當有用而且適當的。這是因為 RD 與 WR 脈衝永遠不會衝突（任何時刻均僅可能有一者動作），所以，一輸入口與一輸出口共用同一個位址亦不致造成問題。但是，兩個輸入口就不能共用同一個位址，而且兩個輸出口亦不能。事實上，您可能發現，即使一輸入口與一輸出口共用同一位址，但就功能而言，兩者可能毫不相干，而且可用於不同的界面電路上。

這一節所討論的觀念與基本電路都非常重要，其在其它章節會有更進一步的延伸。您務必要了解我們所討論的這些選取設備的信號。記得，我們一直尚未描述輸入與輸出設備像什麼樣，以及其如何動作，這些都將在下一章討論。

## 2 - 2 - 2 以解碼器作位址解碼

經常，以解碼器（decoder）電路取代邏輯閘位址測知電路反而更容易，而且，以解碼器電路取代 NOR 閘設備選取電路有時亦是。為何解碼器這麼好用呢？或許，最好我們應先看看幾種解碼器的樣子以及其如何動作。在研究解碼器時，希望您隨時記得，解碼器只不過是整合在一起，以易於使用的一群邏輯閘罷了。

解碼器電路的規格通常說成 X線對 Y線解碼器，其中，X代表解碼器之二進輸入數，而Y代表可能的輸出個數，或者X輸入所可能呈現的不同二進組合個數。舉個例子而言，若解碼器具有 4 個輸入，則由於 4 個位元可能有 16 種不同的輸出，故這個解碼器即稱為“4 線對 16 線”的解碼器。事實上，後面您會看到，這個規格是一個實際的解碼器。

解碼器的每一個二進輸入都有兩種可能的狀態，亦即，邏輯 0 與邏輯 1。這些輸入之間彼此互不相干。就其僅有兩種可能值而言，輸出亦是二進的，不過，其並不完全獨立。解碼器每次僅有一個獨特（與衆不同之意）的輸出，這個輸出即代表二進輸入的數值或“權重”（weights）。經常，這個獨特的輸出均為邏輯 0，而其它的輸出則均為邏輯 1。

典型的解碼器電路為編號 SN74LS139 的積體電路（英文簡稱 IC）。如圖 2 - 7 所示，這塊積體電路（或稱晶片）事實上含有兩個完全獨立的 2 線對 4 線解碼器。

SN74LS139 的真值表如表 2 - 3 所示。

## APPLE 界面實驗

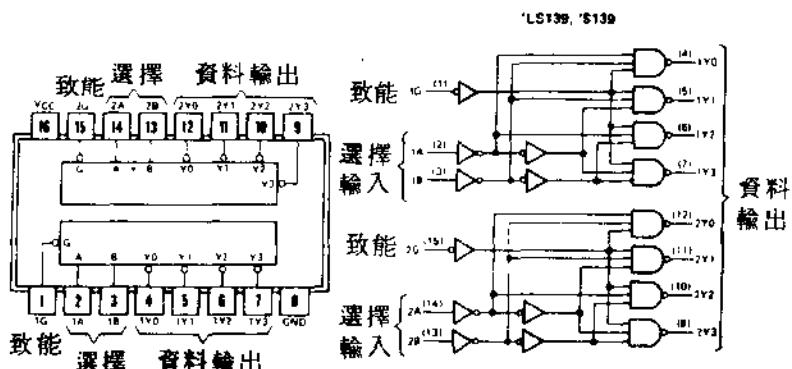


圖 2-7 SN74LS139 解碼器晶片之電路圖與接腳圖

表 2-3 SN74LS139解碼器的真值表

輸 入			輸 出			
致能 G	選擇		Y0	Y1	Y2	Y3
	B	A				
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	L	H	H	L	H	H
L	H	L	H	H	L	H
L	H	H	H	H	H	L

H = 高電位 L = 低電位 X = 不管

當然，表 2-3 所示之真值表對 SN74LS139 晶片內所含的兩個解碼器都有效。絕大多數的解碼器電路都含有一致能（enable）輸入（或稱動作控制輸入），這個輸入可控制整個解碼器的動作與不動作。就 SN74LS139 而言，這個輸入就是 ENABLE 或“G”輸入。您可發現，當“G”輸入為邏輯 1 時，不論輸入 A 與 B 如何，所有輸出均被迫成邏輯 1 狀態。這種情況表示解碼器目前已完全失效。如此

，我們可不必取掉電源而讓整個解碼器失效不動作。

現在，我們來看看一個以 2 線對 4 線解碼器作設備位址解碼的極簡單例子。假設我們僅有少數幾個輸入 / 輸出設備，且 SN74LS139 IC 上的解碼器已足敷使用，則圖 2-8 所示之解碼器電路即為一典型的應用例子。這個電路僅解

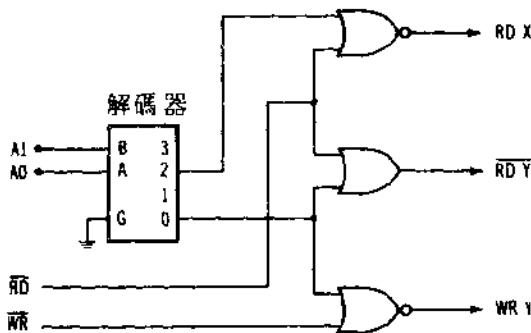


圖 2-8 以 2 線對 4 線解碼器作設備選取

碼位址巴士的其中兩個位址位元，而忽略其它位元。注意，為使解碼器之輸出能正確動作，致能輸入 G 已接地。另外所增加的 NOR 與 OR 閘產生實際的設備選取脈衝。

由於沒有明確的位址，因此，幾個設備選取信號我們分別稱之為 RD X、RD Y、與 WR Y。就圖 2-8 之結構而言，不論 PEEK 命令所使用的位址是 01010101 00000010、00011101 11110110、或 00000000 11111110 [譬如，A = PEEK(21762)]，解碼電路都將產生 RD X 設備選取脈衝，因為，這幾個位址的最低次兩位元  $A_1, A_0$  都等於 10<sub>2</sub>。由於解碼電路中並未解碼  $A_1, \sim A_2$  之位址位元

因此，這幾個位址位元的內含為何均無關。這種設備選取的方式即稱為非絕對 (nonabsolute) 設備選定。所謂非絕對設備選定，即有好幾個位址都可產生選取脈衝。

圖 2-8 的電路解碼 4 個位址，因此可選取 8 個不同的設備，包括 4 個輸入設備與 4 個輸出設備。當然，還必須加上其它的 NOR 或 OR 閘。這種解碼方法雖然沒有多大的彈性，輸入 / 輸出設備的數量不能超過 8 個，但其在小型系統已堪稱夠用。由於其隱含兩個可應用至其它解碼電路的觀念，因此，讓我們再仔細地看看這個電路。

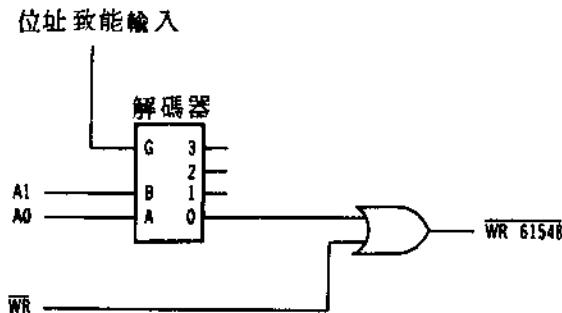


圖 2-9 絕對位址選定所用的解碼器

圖 2-8 之解碼器的致能輸入“G”就直接接地，以使解碼器恒履行解碼功能。這個輸入可使解碼器用於絕對解碼。只要我們使用一個閘控電路，使這個閘控電路唯有在 A15 ~ A2 等位址線出現某一種位元組合型態時，才供應一個致能信號給解碼器之“G”輸入，原來的解碼器電路即可變成絕對選取。而這種閘控電路在前面我們已經討論過了，圖 2

— 5 所示的電路就是一個很好的例子。這個電路即可用以產生以上我們所講的致能輸入。由於 A1 與 A0 兩條位址線已用於解碼器中，因此，其就不必再用作閘控電路的輸入了。圖 2-9 所示即為一個簡單的例子。如圖所示，解碼器之致能輸入來自另一閘控電路。若使用圖 2-5 所示的電路，那 A1 與 A0 的變換開關就不接。

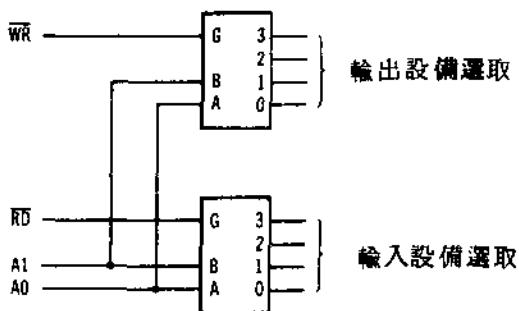


圖 2-10 解碼器之致能輸入與  $\overline{WR}$  以及  $\overline{RD}$  合用，產生設備選取信號

假如圖 2-5 所示之電路加上一解碼 11110000 的上半位址閘控電路，並且 A1 與 A0 兩條最低次位址線不接，則圖 2-9 之解碼電路將僅有在位址輸入為 11110000 01101100 至 11110000 01101111 時才能動作。如此一來，這個解碼電路的 0、1、2、3 等四個輸出將分別對應於 61548、61549、61550、與 61551 等四個設備位址。（這四個位址寫成十六進制則為 F06CH、F06DH、F06EH、與 F06FH。）當然，每一設備選取脈衝的產生都必須加

上額外的 NOR 或 OR 閘（如圖 2-9 所示）。

另一種方法是 SN74LS139 IC 上的兩個解碼器電路都使用，以  $\overline{RD}$  與  $\overline{WR}$  兩個功能脈衝推動（致能）解碼器。如此，位址選取又變成了非絕對性，不過，設備選取閘控則在晶片內完成。圖 2-10 所示即為這種情形。您可發現，產生設備選取脈衝已不再需要 NOR 閘與 OR 閘。雖然這種電路並不一定立即有用，但其却真正說明了以解碼器之致能輸入產生設備選取脈衝的用法。解碼器之閘控或致能輸入可用以產生設備選取脈衝，亦可用作絕對解碼。有時，其亦可兩種

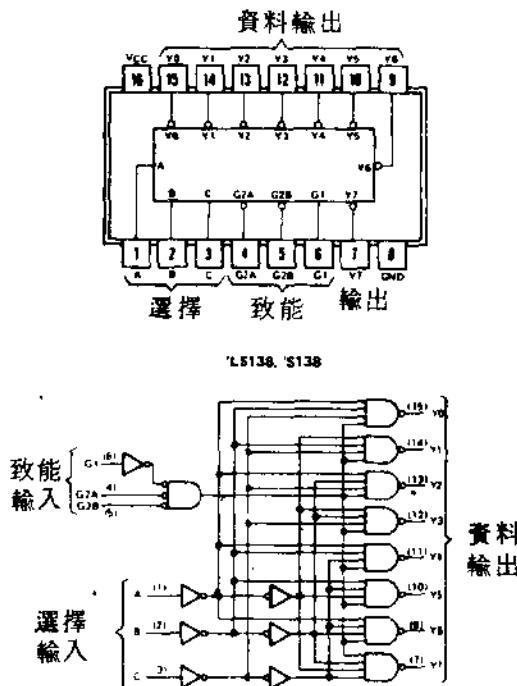


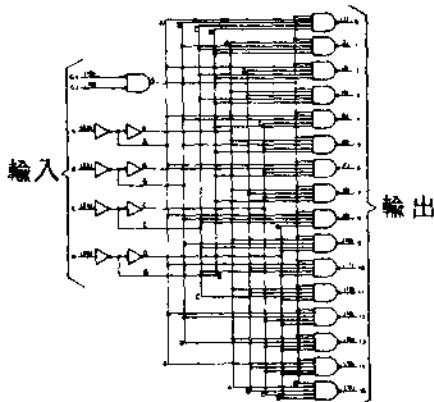
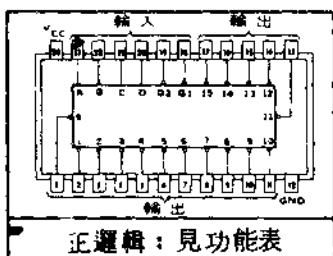
圖 2-11 SN74LS138 解碼器

都用。

### 2-2-3 大型解碼器

在做 Apple 計算機與外部設備的界面工作時，您可能還會用到一些其它的解碼器電路。視您所選用的型態而定，這

功能方塊圖及輸入與輸出電路圖



功能表

端子		端子																			
D1	D2	D	C	B	A	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
L	L	L	X	L	L	L	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H
L	L	L	L	L	L	H	L	H	X	H	H	H	H	H	H	H	H	H	H	H	X
L	L	L	L	H	L	H	H	L	H	H	H	H	H	H	H	H	H	H	H	H	H
L	L	L	L	H	H	H	H	L	H	H	H	H	H	H	H	H	H	H	H	H	X
L	L	L	L	H	L	H	H	H	L	H	H	H	H	H	H	H	H	H	H	H	X
L	L	L	L	H	H	H	H	H	L	H	H	H	H	H	H	H	H	H	H	H	X
L	L	L	L	H	H	H	H	H	H	L	H	H	H	H	H	H	H	H	H	H	X
L	L	L	L	H	H	H	H	H	H	H	L	H	H	H	H	H	H	H	H	H	X
L	L	L	L	H	H	H	H	H	H	H	H	L	H	H	H	H	H	H	H	H	X
L	L	L	L	H	H	H	H	H	H	H	H	H	L	H	H	H	H	H	H	H	X
L	L	L	L	H	H	H	H	H	H	H	H	H	H	L	H	H	H	H	H	H	X
L	L	L	L	H	H	H	H	H	H	H	H	H	H	H	L	H	H	H	H	H	X
L	L	L	L	H	H	H	H	H	H	H	H	H	H	H	H	L	H	H	H	H	X
L	L	L	L	H	H	H	H	H	H	H	H	H	H	H	H	H	L	H	H	H	X
L	L	L	L	H	H	H	H	H	H	H	H	H	H	H	H	H	H	L	H	H	X
L	L	L	L	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	L	H	X
L	L	L	L	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	L	X
L	L	L	L	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	L
H	H	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

H = High Level L = Low Level X = Indeterminate

圖 2-12 SN74154解碼器