

## 再談巴士

雖然許多讀者做完第 6 章的所有實驗就覺得夠了，不再做進一步的界面設計與開發，但有些人可能有志於為自己的計算機系統設計一套永久性的特殊界面電路。本章主要即針對這些讀者所寫。在這一章之內，我們將介紹如何利用 Apple 計算機所提供之特色，設計各種特殊的界面電路。

若您希望設計一套一用再用的界面電路，則您一定不想將之製作於免焊的麵包板上。因為，麵包板電路佔用空間，看起來雜亂不堪，而且經常在最差的時刻分解了。另一種方式是將界面電路製作成永久形式，使其能置於 Apple 的殼子內，不受損壞。

Apple 計算機設計時，設計工程師們早已發覺將來用者一定有志於再擴充系統，以使計算機能接上各種標準的週邊以及非標準的電路。因此，Apple 計算機在主印刷底板後面均留有八個母邊緣接點，以便幾個重要的信號能輕易為需要者所用。由於前面所介紹的界面就插入其中一現有的“槽”內，因此，事實上您已用過了部份這些信號。

幾個槽分別由 0 至 7 編號，其中除了第 0 號槽已由製造

商特別留作計算機之特別擴充用外，其餘幾個您都可以用。外面已有許多廠商專賣正好能插入這些槽內的界面，這些界面皆可毫不費力地插入這些槽內。

第 5 章我們曾介紹了一些常用的界面信號：位址巴士信號、資料巴士信號、與一些控制信號。但這七個可用的界面邊界接點上，還有一些其它很有用的信號。這些信號即如表 7-1 所列。

由於資料巴士與位址巴士您都熟透了，因此，這裡我們就不再重複。底下，我們介紹一些其它在界面上非常重要的信號。

## 7-1 界面控制信號

### I/O SELECT

表 7-1 Apple 巴士信號及說明

接 脚	名 稱	說 明
1	$\overline{\text{I/O SELECT}}$	邏輯 0 信號，計算機選取 $C_{n}00H \sim C_{n}FFH$ 時， $n$ 號槽的這個信號動作，在 $\phi_0$ 時動作。0 號槽沒有。(10)*
2~17	A15 ~ A0	加緩衝之位址巴士線。(5)

18	R / $\bar{W}$	加緩衝之讀 / 寫控制信號。 。(2)
19	SYNC	視訊時序同步信號。只有 7 號槽有。(?)
20	$\overline{I/O}$ STROBE	邏輯 0 信號，計算機選取 C800H ~ CFFFH 時， 所有槽的這個信號均動作， 於 $\phi_0$ 時動作。(4)
21	RDY	6502 微處理器之準備好 控制輸入。
22	DMA	直接記憶體存取控制輸入。
23	INT OUT	接至鄰接槽之插斷菊環信 號。
24	DMA OUT	接至鄰接槽之 DMA 菊環 信號。
25	+ 5 伏特	+ 5 伏特電源接點。最高 供應各板 500mA。
26	GND	系統接地點。
27	DMA IN	接至鄰接槽之 DMA 菊環 信號。
28	INT IN	接至鄰接槽之插斷菊環信 號。
29	$\overline{NMI}$	6502 晶片之不可罩蓋插 斷輸入。將處理器引導至 03FBH 處的副程式。

30	$\overline{\text{IRQ}}$	6502 晶片之可罩蓋插斷輸入。插斷副程式之位址在 03FE 與 03FF 處。
31	$\overline{\text{RES}}$	輸入 / 輸出線。接邏輯 0 時，Apple 重置。界面可監聽或產生重置。
32	$\overline{\text{INH}}$	接邏輯 0 時，所有內部的 ROM 都禁能。
33	- 12 V	- 12 伏特電源接點。總供應電流 200mA。
4	- 5 V	- 5 伏特電源接點。總供應電流 200mA。
35	COLOR REF	這個 3.580MHz 彩色參考信號唯有 7 號口才有。 ( ? )
36	7 M	標準的 7.159 MHz 參考信號。(2)
37	$Q_3$	標準的 2.046 MHz 參考信號。(2)
38	$\phi_1$	標準的 1.023 MHz 微處理器時序信號。(2)
39	USER1	邏輯 0 輸入。接邏輯 0 時，所有內部輸入 / 輸出設備都禁能。
40	$\phi_0$	標準的 1.023 MHz 微處

41	<u>DIVICE SELECT</u>	理器時序信號。 $\phi_1$ 之反相。(2)
42-49	D7 ~ D0	邏輯 0 信號，各槽一個。各槽均有 16 個位址使之動作（見表 7-3）。10 加緩衝之資料巴士信號。(1)
50	+ 12V	+ 12 伏特電源接點。總供應電流 250 mA。

由於信號頂上有一橫線，因此，輸入 / 輸出選取信號，I/O SELECT，（第 1 腳）為邏輯 0 動作。剛剛談到的七個現成界面槽（1 號至 7 號）每一個即均具有其各自的輸入 / 輸出選取信號，因此，這個信號可用以選取任何一塊特定的板。當位址巴士線上之位址為  $C_{n}00$  至  $C_{n}FF$ （含）之間時， $n$  號板槽的輸入 / 輸出選取信號即動作。譬如，若 Apple 之位址巴士上所送之位址為  $C5AB$ ，則 5 號槽之輸入 / 輸出選取信號即為邏輯 0。這時，其它口之輸入 / 輸出選取信號即全不會動作。但，有時候這些信號亦全部不會動作。表 7-2 所示即為影響各個輸入 / 輸出選取信號的位址範圍。

輸入 / 輸出選取信號有多種可能之用途。由於其在 Apple 計算機選取一段連續的 256 個位置時動作，因此，這個信號可用作一具有 256 個位址之記憶晶片的致能信號。其同樣亦可作為一可選取 256 個輸入 / 輸出設備之設備

表 7 - 2 輸入/輸出選取之位址分配

界 面 槽	位 址 範 圍	
1	C100-C1FF	49408-49663
2	C200-C2FF	49664-49919
3	C300-C3FF	49920-50175
4	C400-C4FF	50176-50431
5	C500-C5FF	50432-50687
6	C600-C6FF	50688-50943
7	C700-C7FF	50944-51199

選取解碼器的致能信號。這兩個應用分別如圖 7 - 1 與 7 - 2 之方塊圖所示。

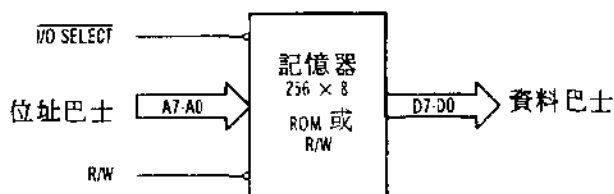


圖 7 - 1 以 I/O SELECT 控制—記憶頁

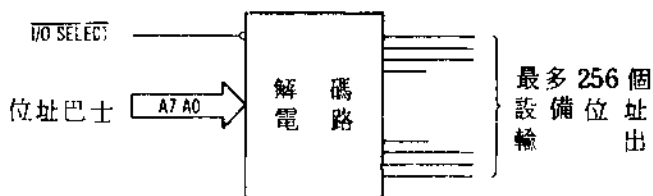


圖 7 - 2 以 I/O SELECT 控制—記憶位址解碼器

您或許要問，既然 Apple 本身已能輕易地選取 48 K 的

記憶位置了，有人還在 Apple 計算機系統加上 256 個位元組的記憶器幹嘛！答案是這樣的，由於組合語言程式效率高，因此，有些應用必須採用一簡短的組合語言常式“推動”界面。在這種情況下，這種“推動常式”（driver routine）即可另置於一塊界面電路專用的唯讀記憶（ROM）晶片上。如此，推動常式即變成整個界面的一部份，而且當界面板（或卡）一插上，這些常式即自動“取入”（loaded），立即可用，而不必像其它常式，再特地由卡帶或磁碟取入。同時，這樣這些常式亦不必佔用任何其它的記憶空間。

另外，有時候界面亦會需要一小量的可讀寫（R/W）記憶器作暫時儲存。這時，I/O SELECT 信號亦可用以控制一個單具 256 個記憶位元組的記憶器。

記得，每一個界面槽均具有其各自的 I/O SELECT 信號，且每一個信號均在 Apple 計算機選取一特定記憶“頁”（page）時動作。

## I/O STROBE

輸入 / 輸出攫取信號（I/O STROBE）是一所有界面槽都有的邏輯 0 信號。其為所有界面接點所共用，並不專屬於任一者。每當 Apple 計算機選取 C800H 至 CFFFH（含）之間的任一記憶位址時，這個信號即為邏輯 0。因此，當位址巴士上的位址落於這個範圍（共含  $2048 = 2K$  個記憶位置）時，每一塊板都會收到這個信號。

您可直接以這個信號作記憶晶片與輸入 / 輸出設備之致

## APPLE 界面實驗

能信號，但您亦可將其與某些位址巴士信號，如 A10 ~ A0，一起再經過一次閘控，以肯定這個信號。圖 7-3 之簡單方塊圖所示即為這個信號的用法。這個電路以  $\overline{\text{I/O STROBE}}$

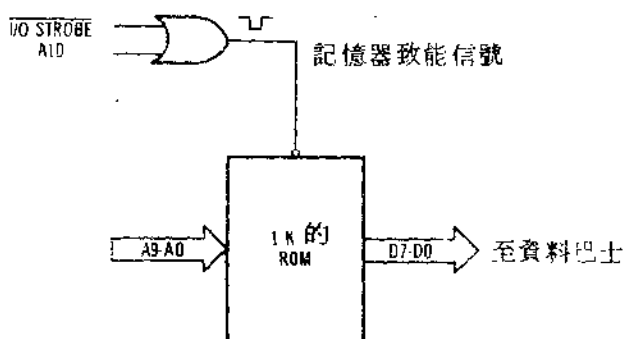


圖 7-3 以  $\overline{\text{I/O STROBE}}$  信號控制 1K 記憶區段

信號選取界面上一個 1K 的 ROM 記憶空間。這個信號另外所能選取的 1K 記憶位址則可分佈於其它界面。不過，作者建議您使用這個信號時還是要多小心，因為，有時候您會發現，製造商已經用了這條線解碼記憶與輸入 / 輸出設備位址，而且其用法完全正如以上所述者。因此，您會發現，您不知該選取您希望加至系統之買來界面好，或選取您自己設計、製作、且裝好的界面好。

## DEVICE SELECT

這個信號是每個界面槽所特有的，而且，如表 7-3 所示的，每個界面槽僅具有 16 個位址。設備選取 ( $\overline{\text{DEVICE}}$



表 7 - 3 設備選取信號之位址分配

界 面 槽	位 址 範 圍	
0	C080-C0BF	49280-49295
1	C090-C09F	49296-49311
2	C0A0-C0AF	49312-49327
3	C0B0-C0BF	49328-49343
4	C0C0-C0CF	49344-49359
5	C0D0-C0DF	49360-49375
6	C0E0-C0EF	49376-49391
7	C0F0-C0FF	49392-49407

SELECT) 信號亦為邏輯 0 動作。由於設備選取信號僅動作於一含 16 個位址之區間內，因此，其用途即較侷限於如圖 7 - 4 所示的輸入 / 輸出設備選取。圖 7 - 4 之電路以設備選取信號推動 ( 致能 ) 一 4 線對 16 線之解碼器。若某一界面僅具有一種功能，而且僅需一個致能信號，則您即可選擇單獨使用設備選取信號，而不必再進一步解碼。只要您記

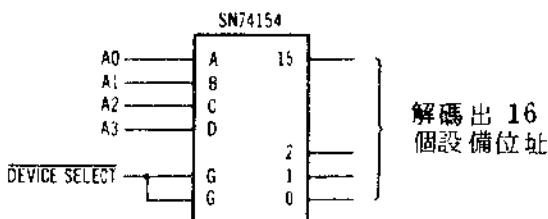


圖 7 - 4 以 DEVICE SELECT 信號推動一輸出  
16 個位址的解碼器

得這樣選取的設備在 16 個不同位址 ( C0n0H 至 C0nFH ) 時均動作，就毫無問題。萬一您決定爾後再作擴充，則這個

信號這樣用法亦限制了您在界面上增加其它功能的能力。

## IRQ 與 NMI

6502 微處理器晶片上有兩個插斷輸入。 $\overline{\text{IRQ}}$  (插斷請求) 輸入為可罩蓋的 (maskable)，其可為軟體步驟所禁能。 $\overline{\text{NMI}}$  (非罩蓋插斷) 輸入則恒可得逞 (動作)。

這兩條插斷輸入線為全部七個界面槽共用，其中， $\overline{\text{IRQ}}$  信號接在 30 腳， $\overline{\text{NMI}}$  信號接至 29 腳。於絕大多數界面電路上， $\overline{\text{NMI}}$  線都特別律定給某一個特定週邊設備，而且無論如何這個輸入信號都要為 6502 微處理器所認可。 $\overline{\text{IRQ}}$  線則由其它許多界面電路所共用。插斷處理常式內必須含有適當的步驟，使計算機能測知其中那一個設備真正提出插斷。在這種情況下，每一個插斷設備都必須具有一個位元的輸入口。6502 微處理器只要讀取這個輸入口的資料，即可得知該插斷設備之旗號狀態，知其是否提出插斷。圖 7-5 所示即為一典型的插斷旗號電路。留意到，旗號是在軟體程式的

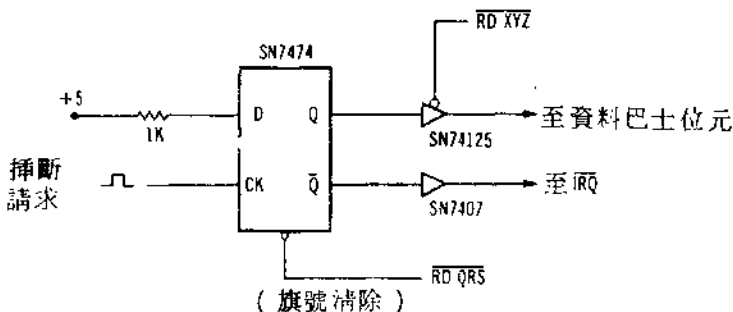


圖 7-5 插斷旗號電路圖

控制下清除的。

假如採用這種“取樣式”，或稱“詢問式”，的插斷，教計算機讀取每一設備的狀態，看其是否提出插斷，則軟體程式內即可建立起一套優先順序（priority）。舉個例子而言，若程式沿著設備 A、B、C……等等的順序讀取各設備的狀態，則設備 A 的優先順序自然就最高，設備 B 次之，設備 C 再次之，如此類推。

幾個界面槽上另外亦有兩個您可能有興趣的插斷線。這兩個信號即為 28 腳之插斷輸入（INT IN）以及 23 腳的插斷輸出（INT OUT）。其特別用以形成界面板與界面板間之“菊環”（daisy chain）插斷結構。如圖 7-6 所示，信號由一個界面板串接至下一個界面板：1 號界面槽的 INT OUT 信號接至 2 號界面槽的 INT IN，而 2 號槽的 INT OUT 信號又接 3 號槽之 INT IN，如此類推。每一

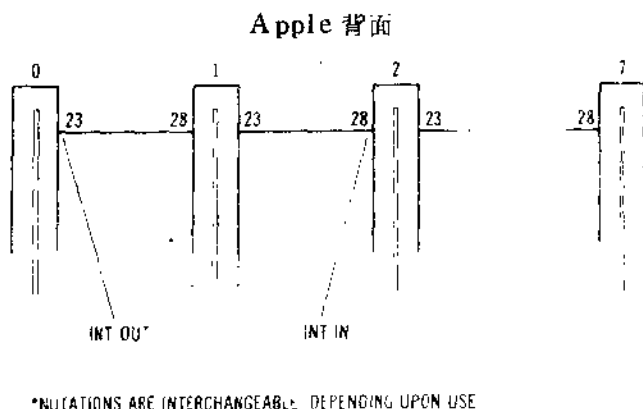


圖 7-6 INT IN 與 INT OUT 巴士信號的接法

個界面槽的 INT IN 或 INT OUT 均僅與隔壁的界面槽連接一次，別不再作其它任何連接。

圖 7 - 7 所示即為一簡單之菊環插斷結構。愈在“下游”的插斷設備，插斷優先順序即愈低，且離 6502 微處理器之 INT 接腳愈遠。在這個電路內，只要有較高優先之插斷設備提出插斷請求，其它任何較低優先之設備所提出的插斷請求即暫時被忽略。等較高優先之設備被服務完，其插斷旗號清除後，這個較高優先之設備即會將閘門“打開”，讓後面優先順序較低的插斷設備“能”將其插斷請求傳給計算機。

正如您可看出的，計算機還是需要有個辦法得知那一個設備正產生插斷，才能選取適當的插斷服務常式 ( interrupt service routine )。由於這種插斷方式十分複雜，因此，我們建議您還是採用圖 7 - 5 所示之插斷旗號電路。在絕大多數情況下，這應該夠用。菊環結構內，界面電路板之間不能有“空”槽，否則，整個 INT IN / INT OUT 電路“環”就會斷掉。插斷至此已討論得差不多了，進一步的細節請您參考“6502 程式設計與界面實驗”一書。

## DMA

DMA 輸入使外部設備能不必經 6502 微處理器而直接選取記憶位置。因此，在這種狀況下，外部設備直接存取了記憶器 ( Direct Memory Access, 簡稱 DMA )。由於有好幾個設備均可請求直接存取記憶器，而幾個界面槽又都有可如 INT IN 與 INT OUT 連接的 DMA IN 與 DMA

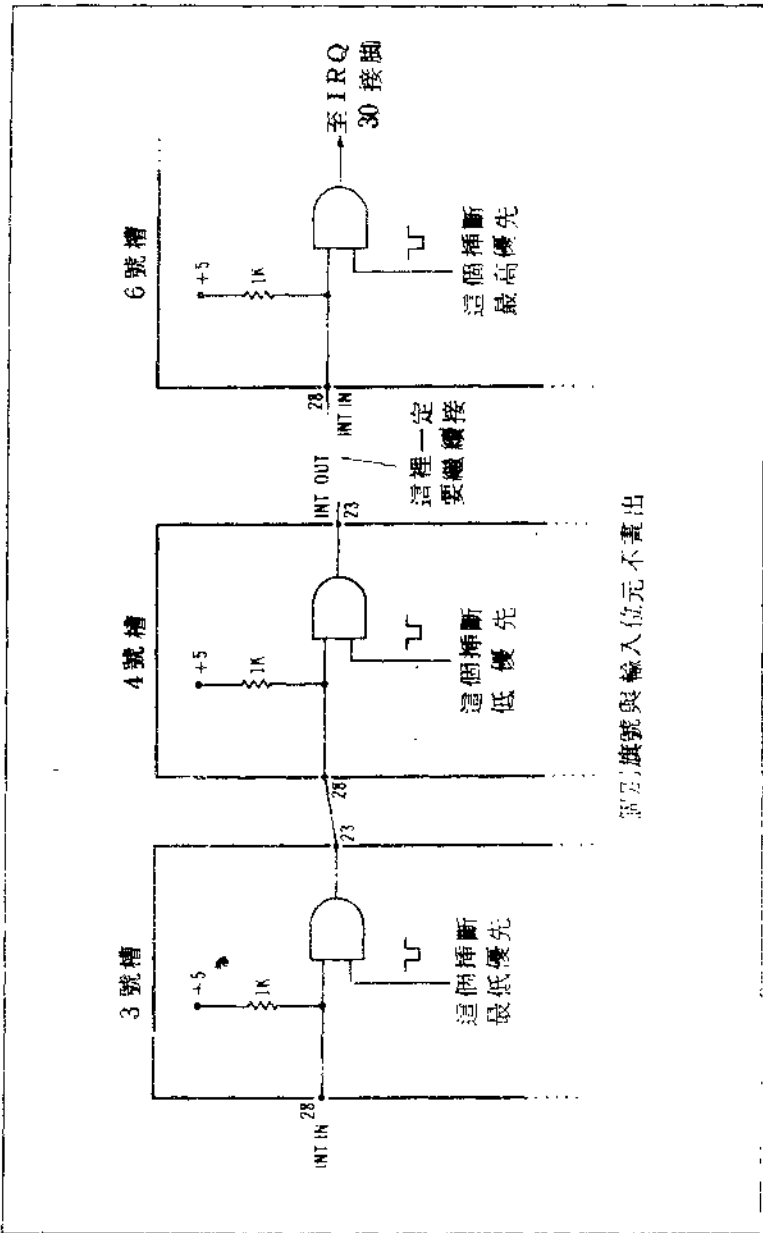


圖 7-7 插斷與輸入位元不畫出

圖 7 - 7 狗環插斷結構

OUT 接脚，因此，各週邊設備之間亦可形成一菊環結構。由於直接存取界面設計並非一件十分簡易的工作，因此，在使用這個特色前，我們建議您能先徹底地了解 6502 微處理器與其有關電路之動作原理。

### RES

第 31 脚之重置信號，RES，實際上是一雙向的信號線。由於每當電源一加上 Apple 重置時，或您按 RESET 按鈕時，這個信號均為邏輯零，所以，您可以這條線重置您的界面電路。只要將這條線接地，您亦可迫使 Apple 變成重置狀態。若您選擇以這條線自界面電路重置 Apple，則您必須以一高電流之集極開路閘或緩衝器，將這條線接地。SN7407 集極開路緩衝器晶片這時就是個適當人選。RES 信號線為所有界面槽共用。

### INH

於 Apple 計算機內，您可以自己所寫的組合語言程式取代儲存於 BASIC 解釋程式 ROM 內的程式。只要將第 32 脚之 INH 線接地，您即可使儲存 BASIC 解釋程式以及監督程式的所有 ROM 都失效，進而以您所寫的程式控制整個系統。由於這種作業已早留有餘地，所以，您很可能用不到此一功能，因為，這樣您就無法存取 Apple 所供應之標準 ROM 內的所有副程式。舉個例子而言，沒有 BASIC 解釋

程式 ROM 內的副程式，您就很難控制顯示幕。但假如您想使用這個功能，則您就需以一集極開路緩衝器晶片將這條線接地。

## USER 1

這個輸入使您能教 Apple 計算機停止產生 I/O SELECT 與 DEVICE SELECT 信號，以致能“關掉”所有的輸入/輸出設備。欲產生此一作業時，這條線必須接邏輯 0。爲了防止不小心用錯了這條線，因此，欲使用 USER 1 信號時，您必須以一條跳線將 Apple 主印刷底板上的兩個焊接點接起，您才有辦法使用這個信號。有關這個信號使用的詳細細節，請您參考“Apple II 參考手冊”。

由於使用 I/O SELECT 與 DEVICE SELECT 信號的主要目的即在簡化界面設計，因此，除非您想將系統擴充，接上一些基本系統以外，或可能用到已分配給 I/O SELECT 與 DEVICE SELECT 信號的記憶位址時，您才有可能用到這條線。USER 1 信號在界面接點上的第 39 腳。

## RDY

有時候，我們必須令 6502 微處理器稍爲“延遲”一下，致使輸入/輸出設備或記憶器晶片有足夠的時間讀取資料，並將資料置於資料巴士上。只要將每一界面接點 21 腳上的準備好輸入 (RDY) 接地，我們即可令 6502 微處理器處

於“等候”狀態。這個輸入必須與微處理器的時序取得同步，而且其應於  $\phi_1$  時序為邏輯 1 準位時轉態。由於較早期記憶元件之資料存取速度均無法達到計算機之要求，因此，RDY 輸入均用於較早期的 6502 系統。在這些系統內，只要送出位址資訊，6502 微處理器即必須進至“等候”狀態數個時序週期，一直等到資料備好於資料巴士上為止。除了很特殊的界面外，相信您很少會用到這個信號。

## 時序信號

Apple 上有六個時序信號可作界面用。這些信號分別是  $\phi_0$ 、 $\phi_1$ 、 $Q_3$ 、7M、COLOR、REF、與 SYNC。 $\phi_0$  與  $\phi_1$  為主時序信號，頻率 1MHz。兩個時序信號彼此反相。這些信號主要用以控制外部輸入/輸出作業，使資料於巴士上能正常流通。如圖 5-12 所示， $\phi_1$  信號用以控制外部輸入/輸出設備的  $\overline{RD}$  與  $\overline{WR}$  信號產生。輸入/輸出接點處的 I/O SELECT 與 DEVICE SELECT 信號都已事先與  $\phi_1$  時序信號“閘控”認定過。

$Q_3$  信號則為一頻率 2MHz 的非對稱時序信號；亦即，其並非方波。7M 信號則為一波形為方波，頻率為 7MHz 的時序信號。幾個時序信號均由 Apple 內的主時序電路得來，其彼此間的關係如圖 7-8 所示。有關 6502 之時序關係，請讀者進一步參考 6502 微處理器之數據說明書。

COLOR REF 與 SYNC 信號唯有 7 號界面槽有。

COLOR REF 信號為 Apple 之視頻時序電路所產生的彩色



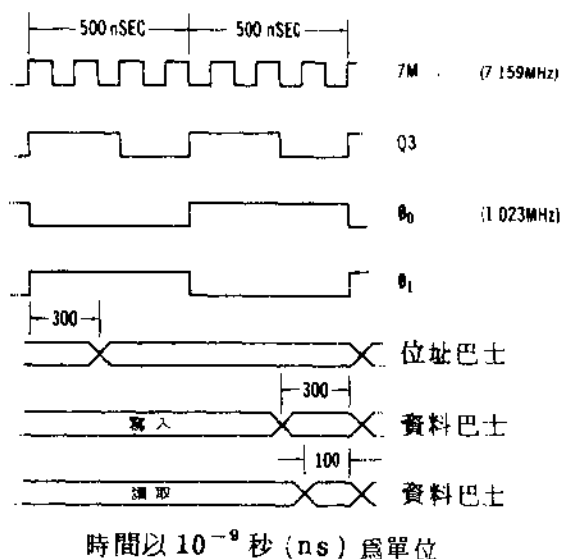


圖 7-8 Apple之各種時序信號的時序圖

參考信號，其頻率為 3.5 MHz。除非您用到視頻控制電路，否則，您將不可能用到這些信號。

## 電 源

界面接點提供了四種標準電壓與接地。這些電壓分別為 +12、-12、+5、與 -5 伏特。由於每一電壓的電流均僅有幾百毫安 ( milliamperes )，所以，您應考慮採用諸如 SN74LS00 族中的低功率界面晶片。