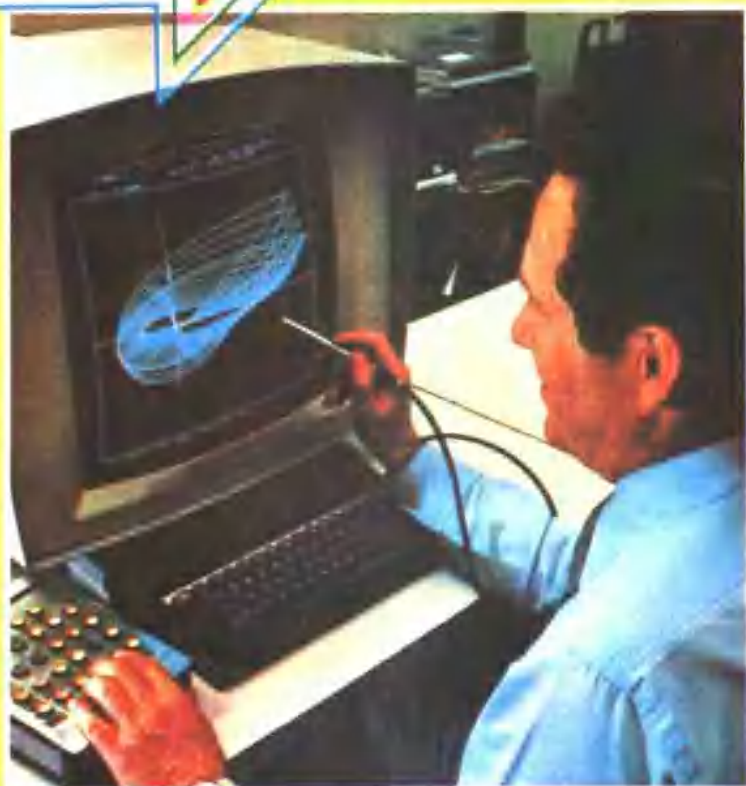


# PET 界面技巧

陳石城 譯



# PET 界面技巧

陳石城 譯

電腦語言中心出版

# PET 界面技巧

---

編譯者：陳 石 城

出版者：電 腦 語 言 中 心

發行者：

九龍彩虹道400號六樓

印刷者：合興隆印刷公司

香港仔宏利工業大廈七樓

---

定價港幣·\$ 30.00

# 序

最近幾年，我們見到小型計算機市場如一般計算機一樣快速地成長，像 TRS-80, APPLE 及 PET 都已經可以買得到。這些小型計算機以低成本提供大量的計算功能，所以它們廣泛地被應用在幾年前還算很昂貴的計算功能上。有許多使用者已經可以自己單獨使用計算機，而一些使用者則發現在計算機加上輔助性的周邊裝置是十分必要的，像印表機，類比轉換器，遙控器等已經廣受應用了。許多人發現他們需要加裝一些裝置到計算機上，但由於缺少介面標準，所以他們面臨了不可避免的負擔——設計並建立自己的介面。

本書有關於 PET 介面技術的問題是要告訴各位一些專用介面的構想，並在 PET 這個系統上來建立起這些介面。我們將採用 BASIC 語言的程式，所以讀者最好能熟悉這種強而有力的程式語言。另外有關設計部份是用 SN7400 系列的標準數位電子元件，所以在接觸這些介面線路之前，至少要了解這些元件究竟是什麼內函。

不像其他計算機只提供一般化的滙流連接器，Commodore 的 PET 計算機具有數種特殊用途的介面連接器，這可以使計算機與實際外界裝置的介面工作變得簡單。

這些連接器提供下面 I/O 口的取用：

- PET 用戶口
- 記憶體擴張口
- IEEE 488 標準介面口

每個口都有特定用途，在本書都將作一番探討。除了從這幾組信號之外，各位可能再也找不到有關PET介面的訊息了。

其中特別有趣的是IEEE-488口；這組信號提供你由IEEE 488標準所定義的口，而且許多計算機廠商及儀器製造商的設備都採用了這項標準。所以這些IEEE 488匹配裝置的介面可以很容易就完成。很多情況，所謂介面只是包含在不同IEEE-488匹配裝置間標準電纜的連接而已，軟體則是用以控制匯流信號線的特殊訊息傳遞。由於有越來越多的裝置可以用IEEE 488匹配I/O口，所以我們將花較多的篇幅在IEEE 488匯流上。也可能用這組匯流來作PET與裝置的介面，即使這些裝置並非使用標準匯流的規格。

各位應了解本書是一本有關這三種I/O口介面技術怎麼作的書。每個介面範例都詳細地解說，並配合完整的軟體和線路。有許多實驗適用於各型具有25×40字母顯示器的PET，其中包含許多改良了ROM的舊PET。

前兩章我們逐步地介紹PET，並由簡單的插線板線路架構來引導各位。這些線路可用標準繞線技術結合起來，於是你可以發現其中的妙趣無窮。

第三章包含許多介面的技術，包括基礎及高等的，它們可用以控制用戶口控制信號。這裏還包括控制用戶口的BASIC指令：PEEK, POKE, 和WAIT。你還會學到用AND, OR, 及NOT運算器來控制I/O裝置，結束了第三章，你應該就十分熟悉用戶口的用法及6522 I/O切片在PET中如何使用了。這一章裏沒有用到組合語言來撰寫程式，當然用組合語言撰寫程式的人應該可以很快地把這些程

式轉換為組合語言的程式。

在 4, 5, 6 三章裏，我們採用另外一種 PET 的介面方法，使用記憶體擴張口。你將會使用許多的介面技術。例如，你會學到各種不同的解碼方法，這些方法可以作選擇不同 I/O 裝置的介面，另外還會學到雙向的 data bus 怎樣把資料傳入或傳出計算機及 I/O 裝置。在第五章，你會看到如何建立一個可以用在記憶體擴張口的簡單線路板。在第六章，有許多實驗可以加強諸位對記憶體擴張口各不同介面的認識。

第七章做了個結論，提供其他介面的資料。例如，PET 可以如何地與非邏輯裝置作介面，而這些裝置都是需要大電壓、大電流的。有關類比轉換器的資料在此一並提及。

第八章包括 PET IEEE 488 匯流的用法，匯流信號也作了詳細的說明。另外還介紹了一種特殊的介面，這種介面用來控制印表機。因為許多人對 IEEE 488 匯流的其他應用甚感興趣，所以我們也提到一種可以由 IEEE 488 匯流作非 IEEE 488 匹配裝置與 PET 介面的一般用途介面線路。這是用另一個微處理機切片來模擬正常 IEEE 488

I/O 裝置出現在匯流上的情形。完整的細節包括組合語言的表列及零件來源附在附錄中。一般用途的“listener / talker”可用在許多應用上，因為它並不特定某一個裝置或某一類介面。

# 目 錄

<b>第一章</b> .....	1
PET 微處理機硬體概論	
位元及位元組    記憶體    周邊裝置    PET 的結構    資料型式	
<b>第二章</b> .....	15
用戶口插線板的架構	
介面電纜    電源供應    建立插線板線路卡    邏輯測試器的接線    邏輯測試器操作的檢驗	
<b>第三章</b> .....	31
用戶口的介面實驗	
概論    用戶口概述    並聯 I / O 實驗插線板上的接線    用戶口位移暫存器概述    插線板上位移暫存器輸入的接線    高等介面專題    可用輸入及輸出線的擴張    數位 - 類比的轉換    用戶口用的數位 - 類比轉換器	
<b>第四章</b> .....	125
記憶體擴張口	
計算機邏輯的輸入和輸出    雙向 Data Bus    栓、三態緩衝器及資料傳遞    PET 的地址解碼	
<b>第五章</b> .....	149
記憶體口 I / O 線路板的架構	
一般架構    電源供應    地址解碼器    輸出口    輸入口    檢驗接線板	

<b>第六章</b> .....	169
用接線板的實驗	
邏輯測試器的使用	地址解碼器
口  旗標與決策	輸出  輸入
口介面專題	類比 - 數位轉換器  高等記憶
<b>第七章</b> .....	211
一些便利的介面提示及構想	
二進制電碼的十進制	用數位 - 類比轉換器繪圖
用 DAC 作簡單的類比 - 數位轉換器	從用戶口或記
憶體擴張口來控制 ASCII 型式的資料	與高功率及
非 - TTL 裝置的介面	
<b>第八章</b> .....	229
與 PET 計算機 IEEE 488 口的介面	
IEEE 488 的信號	IEEE 488 與串聯傳輸的差
異  IEEE 488 介面指令及記憶碼	使用 IEEE
488 介面匯流的 BASIC 指令	PET / IEEE 介面的
架構方法	
<b>附錄A</b> .....	285
PET / IEEE 控制器及 Listener / Talker 序列的流程	
圖	
<b>附錄B</b> .....	299
PET / IEEE 一般用途 Listener / Talker 的程式表	
列	
<b>附錄C</b> .....	317
電子零件供應的來源	



# 第 一 章

## PET 微處理機硬體概論

PET 計算機的心臟部份是 6502 微處理機 ( microprocessor )。6502 是 40 根脚的大型積體電路 ( LSI )，它包含了數位計算機所需的一切邏輯線路。其中 PET 所執行的數學及邏輯運算，都由 6502 來完成。如果我們仔細觀察，將發現 PET 它並不只含一個積體電路 ( Integrated Circuit, I.C. )，而是有上打的 I.C. 所組成，其中有些 I.C. 是記憶體 ( memory ) 切片 ( chip )，有些是輔助切片 ( support chip )。而輔助切片則提供 6502 與記憶體、鍵盤 ( key board )，顯示幕以及其他輸出／輸入口 ( I/O port ) 的邏輯介面。這幾年，計算機工業已經感受到 6502 微處理機在 PET 中所造成技術的革新。在十年前，相等的計算能力大概需要數千美元的成本 ( 而且需要上百個 I.C. )。而今 6502 不到 20 美元就可買得到。

## 2 PET 界面技巧

在討論 PET 的介面裝置之前，我們先來了解 6502 是怎樣處理資料，在 PET 中它又扮演著如何的角色。

### 位元及位元組 ( Bits and Bytes )

二進制 ( binary or base-two ) 對計算機來說是很方便的數學方法。二進制中只有 0 和 1 兩個數字，這剛好可以用數位電子學中所定義的兩種狀態 ( 在 TTL 中，邏輯 high 為正 5 伏，邏輯 low 為 0 伏 ) 來表示。一般，高狀態 ( high state ) 表示 1，低狀態 ( low state ) 表示 0。二進制中，由右至左的位數分別表示 1，2，4，8，16，……。其中 2 的乘冪與十進制對應如下：

$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
128	64	32	16	8	4	2	1

例如二進制中 010011 表示有一個 1，一個 2，沒有 4 和 8，一個 16，沒有 32。它相當於十進制的 19 (  $16 + 2 + 1 = 19$  )。二進制中最大的六位數是 111111，相當於十進制的 63，如果要表示更大的數就需要多補一位。在計算機中，二進制的「數」是由一群電子電路來表示，每一個電子電路代表特定的「數位」，而它的狀態則表示這一「數位」是 1 或 0。一般就稱二進制的數位為「位元」 ( bit )，組成一個數的整群位元就稱為字 ( word )。

6502 的字長有 8 個位元，這是說 6502 的運算，大都是一次 8 個位元。8 - 位元的字長通常又叫一個位元組 ( byte )。byte 是由 IBM 首先引用的，因為 IBM 的

計算機用 16-位元或 16-位元以上的字長，但除了某些運算（例如處理 ASCII 型式的資料）之外，它經常只需要一次處理 8 個位元。針對這點，一些特殊指令被併入計算機的硬體部分，以便執行 8 - 位元次單位（subunit）的運算，而一個字裏 8 - 位元的次單位就叫做一個位元組（byte）了。現在雖然大部分微處理機的字長是 8 - 位元，8 - 位元已經構成一個字，但我們仍然延用「位元組」來稱呼 8 - 位元的「字」。

8 個位元限制了我們所能表達的最大數為 255。當要表達更大的數時，就需要兩個或更多的位元組。在寫類似這種多位元組（multibyte）的運算時，必須注意每個位元組到底表示該「數」的那一部分，千萬不要因此混淆了。儘管如此，在 PET 的 BASIC 模擬程式（Interpreter program）中，6502 可以簡易地處理大於 256 的數；而它的計算仍然一次只有 8 個位元。

包括 PET 在內，大部分的計算機系統由三個基本的部分組成。第一部分是中央處理器（central processor）（這裏用的是 6502），負責計算機一切的數學及邏輯運算。其他的兩部分是記憶體（memory）及輸入／輸出裝置（Input / output device）。

## 記憶體（Memory）

計算機中，記憶體有兩種主要的功能。第一，它儲存了計算機的程式，第二，它可以暫存計算的資料（information）供未來之用。為配合 6502 的 8 - 位元

#### 4 PET 界面技巧

字長，PET 的記憶體是成 8 - 位元的位元組結構，6502 可以隨意取用記憶體的 65,536 個位元組。記憶體的取用，則是將二進制電碼 ( binary code ) 放入 16 條被稱爲 Address bus ( 地址線 ) 的信號線上，每個電碼則選出記憶體中唯一的字。圖 1 - 1 表示 6502 接腳的情形。其中

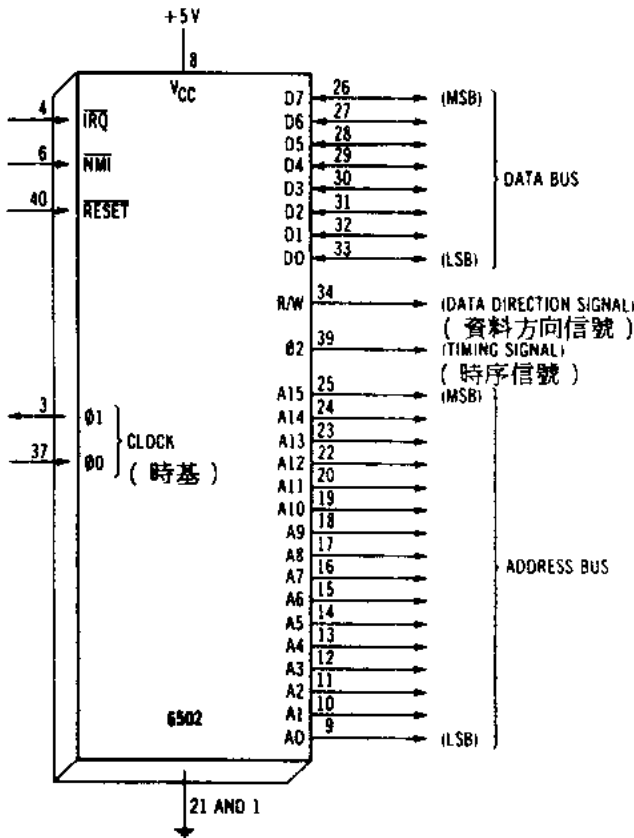


圖 1 - 1 6502 微處理機的主要信號

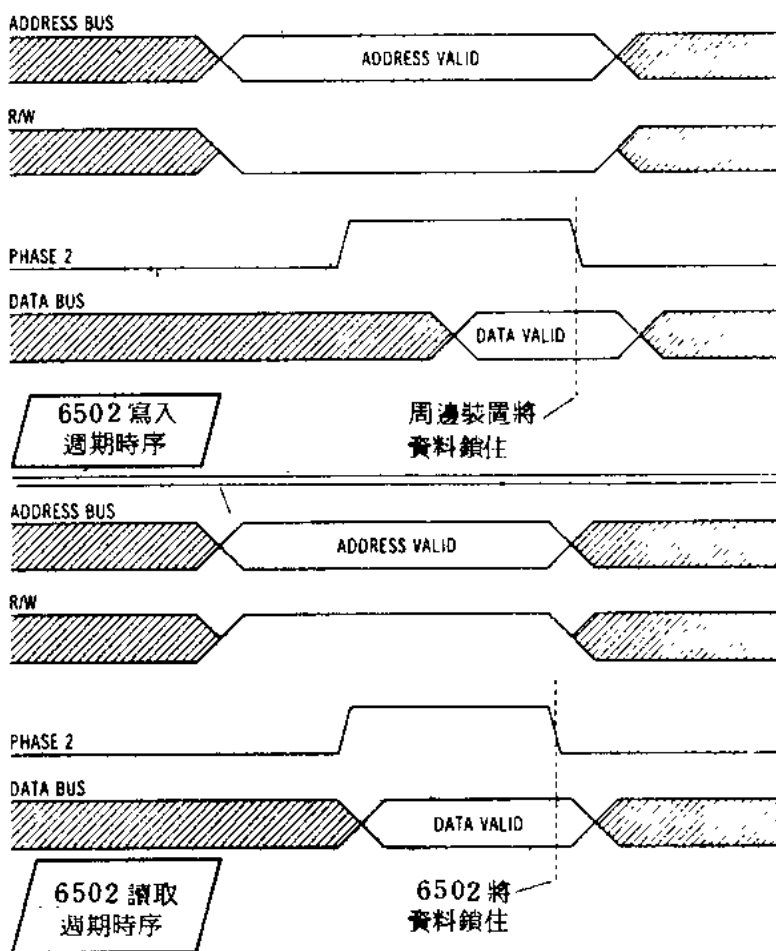
16 條地址線 ( A0 ~ A15 ) 在圖的右邊。65,536 (  $= 2^{16}$  ) 就表示這 16 條地址線所有可能的二進制組合，雖然 6502 可以取用 65,536 個記憶體的字，但將 65,536 個位置完全填滿，似乎並不必要。PET 計算機通常以不用的大小出現 例如 4K, 8K 等等，K 表示  $2^{10}$  (  $= 1024$  )，是記憶體的基本單位。所以，實際上 4K PET 只有 4096 個位元組被填滿，可供儲存程式之用。PET 中記憶體又分為兩類：讀／寫記憶體 ( read / write memory ) 及僅讀記憶體 ( read only memory )。

### 讀／寫記憶體 ( Read/Write Memory )

正如它的名稱，我們既可讀取而且可以寫入資料到讀／寫記憶體，這種記憶體通常又叫隨取記憶體 ( random access memory ) 或叫 RAM。雖然這是誤稱 ( 記著，所有的記憶體都可以隨取 )，但是 RAM 的稱呼依舊廣受使用。6502 與記憶體之間的傳遞方式相當簡單。圖 1-1 中顯示 6502 有 8 條資料線 (  $D_0 \sim D_7$  )，資料線 ( data bus ) 是資料進出 6502 的通道 ( 圖中的箭頭表示資料流通的方向 ) 由於資料在這 8 條線上兩個方向都可以流通，我們就用雙箭號來表示。由於資料在同樣一組線上，既可以進入，又可以送出，可能造成混淆。為避免這層困擾，計算機靠 R / W 線 ( read / write line ) 的狀態來控制資料是要送出或送入 6502。假使 R / W 線的狀態為 high ( +5 伏 )，資料就由 data bus 讀進 6502，若 R / W 線的狀態為 low ( 0 伏 )，資料就由 6502 送到 data bus 上。由於 6502

## 6 PET 界面技巧

每秒鐘要完成上百萬個動作，資料傳送時的時序 (timing) 問題就顯得十分重要。精確的時序是由相二時基 (phase-two clock) 所產生的脈衝來解決，由時基來的脈衝告訴記憶體，要在脈衝的那一瞬間來執行資料的傳遞。例如，典型地從記憶體讀取資料的週期 (cycle) 大概是：首先在 16 條 address bus 上放入適當的二進制電碼，來決定



1-2 6502 資料傳遞的時序

6502 所要讀取資料的位置，同時使 R/W 線變為 high 的狀態，準備讀取資料，這時被二進制電碼選中的記憶體切片，就準備好將該 8 - 位元的字送到 6502 的 data bus 上；但實際完成資料傳遞是在相二時基線，從 high 變到 low 的那一瞬間。讀取及寫入的週期，分別用時序圖 (timing diagram) 在圖 1-2 表示。data bus 中資料流通的方向，由 R/W 線的狀態來表示，精確的資料傳遞時序由相二時基的尾端 (trailing edge) 來控制，其中斜線部分，表示資料尚未決定。

### 僅讀記憶體 (Read Only Memory)

PET 的第二種記憶體就叫僅讀記憶體或叫 ROM，計算機只可以從 ROM 讀出資料而不能寫入新的資料。它的好處是可以長久的保存它原來所存的程式或資料，而且可以買得到。在微算機的發展過程，ROM 扮演了相當重要的角色，微處理切片 (microprocessor chip) 和半導體讀寫記憶體，這兩項主要的進步則大大的減低了計算機的成本。直到最近，計算機採用非常昂貴的磁圈記憶體 (magnetic core memory) 即使電源中斷，磁圈記憶仍然可以保留原來的資料，這一點是半導體記憶體所做不到的。像這種計算機一關掉，記憶體的內容就似乎汽化 (evaporate) 了的，我們就稱為具有揮發性 (volatile)。對於使用半導體記憶體的計算機，揮發性記憶造成相當大的困擾。使用磁圈記憶體的計算機，通常它的面板 (front panel) 非常的費力，啟動程式 (initialization program) 就從面板

## 8 PET界面技巧

由人把它放進記憶體，手續十分繁瑣。因為磁圈記憶體沒有揮發性 ( nonvolatile )，一旦啓動程式放進記憶體，它就可以保存在裏面，免去一再放置的繁雜手續。但對於使用揮發性記憶體的計算機而言，在計算機開了以後要工作之前，就需要一再地重覆放置啓動程式的工作。為解決這層困難，ROM 乃應運而生，加入了計算機，用來儲存長久的啓動程式。因為ROM 工作非常好，所以以前那種浪費的面板已經不需要了，現在大部分的微算機，已經免掉從面板上的開關來直接輸入記憶體的裝備了。

PET的ROM並不是只有啓動程式，它還包含了所有BASIC的模擬程式 ( interpreter )，所以當PET 開動後，就有一種高階語言可供使用了。ROM 有許多變型，有些像PET 中所用的，在工廠時程式就已經被寫進去了；有些可以由場 ( field )把程式寫進去，這種就叫可程式僅讀記憶體 ( programmable read only memory )或簡稱PROM。最流行的ROM是EPROM，它可以經由紫外光洗掉，再重新寫入程式。

### 周邊裝置 ( PERIPHERAL DEVICES )

計算機的第三個基本部分是一些提供輸入及輸出的裝置，這些通常叫輸入／輸出裝置 ( I/O device )。假使計算機不能從外界接受資料，或者把結果輸出給外界，顯然它並沒有多大功用。例如，PET 有數種接受輸入資料的方法，像鍵盤 ( key board )，卡帶驅動 ( cassette drive )和IEEE 488 bus。這本書主要是告訴我們，如何把自己



的周邊裝置和 PET 連結起來，我們對計算機的這一點也特別感興趣。

6502 把周邊裝置當記憶體看待，這一點與別的微處理機稍有不同。它把輸入裝置（像鍵盤）變成僅讀記憶體的一型，同樣地，把輸出裝置（如打字機）當作僅寫記憶體（write only memory）。如同別的記憶體，周邊裝置必須賦予一個 16-位元的地址電碼，以便當這個電碼出現在 Address bus 時，可以被辨認出來。介面的邏輯，當適當的地址電碼出現時，就該辨認出來，同時要通知該裝置準備在脈衝到達時，在 data bus 上進行資訊交換。

### PET 的結構（Organization of the PET）

有了以上的認識，我們就可以開始描述 6502 在 PET 中是如何的被安裝。雖然 6502 容許有 64K 的記憶體或周邊裝置，但 PET 通常只填滿其中的一部分。圖 1-3 是 PET 記憶體的映像（map），可以看出記憶空間的上面，填上了 ROM 和 I/O 裝置。下面的部分則填上 RAM，至於中間尚未填上的部分——開區（open），可以填入 RAM，供儲存 BASIC 程式，或加入 ROM，以便行特定的功能，甚至於可以加入 I/O 裝置。

開機時，6502 自動的被重置（reset），重置之後，6502 就開始執行記憶體上面 ROM 所儲存的程式，這個程式首先讓 PET 知道它有多少個 RAM，從 1024 這個位置開始在記憶體的每一個字，依次放入一個電碼，然後再讀取電碼，假使記憶體是好的，那麼將會讀到相同的電碼。以上