

# Apple 機器語言

張其邦 譯

三葉出版社

# Apple 機器語言

張其邦 譯

三葉出版社印行

# Apple 機 器 語 言

---

編 譯 者：張 其 邦

出 版 者：三 葉 出 版 社

發 行 者：澳門大三巴街富運大廈147號

印 刷 者：金 源 印 刷 公 司  
澳門草街126號地下

---

定 價：H. K. \$ 3 5.00

# 序

這本書的目的是想要對 BASIC 語言有瞭解的 Apple 電腦使用者做設計機器語言程式的介紹，其中由 BASIC 轉變為機器語言的過程是非常簡易的；在這本書的前半段我們使用圖形、發聲等示範程式更顯得有趣味性及實用性，每一個新的指令我們都會做一個解釋，並且對示範程式一步步地解說清楚。

讀者首先由 BASIC 程式中使用 POKE、PEEK 及 CALL 進入並使用機器語言的程式，然後才是 BASIC 的作業系統，使得您可以輸入機器語言程式並執行它。

我們由介紹 BASIC 語言使您能夠很自然地使用 Apple 的監督系統；監督系統允許您直接地輸入、檢查及執行機器語言的程式，這使得編譯 BASIC 敘述所花的時間得以節省。

最後的步驟則是介紹 Apple 的 Mini-Assembler，這個組合語言使您在直接輸入機器語言時所遇到的許多困難得以消除。

由 BASIC 語言逐漸進入機器語言的範疇使您可以利用您已得的知識去發現更深的領域。

您在閱讀本書時將經過四個階段，而機器語言的程式也依照四種不同的方法輸入並執行，每一種方法都將在本書中各別的階段中介紹。

1. **BASIC** 在第一章中介紹，機器語言的程式在 BASIC 語言的控制下，您使用的指令是 POKE、CALL 及 PEEK。
2. **BOS** 在第二章中介紹，機器語言的程式由 BASIC

作業系統控制。

3. SM 在第八章中介紹，機器語言的程式可以由您自己設計並直接由 Apple 的監督系統輸入。
4. MA 在第十一章中介紹，機器語言程式由 Apple 的 Mini-Assembler 組合而成。

BASIC、BOS、SM 及 MA 這些標識都將出現在適當的地方及每章的開始處，使您得以清楚地分辨書中介紹的過程。

# 目 錄

<b>第一章</b>	<b>APPLE SOFT II BASIC的回顧</b> .....	1
	命令.....	3
	設定敘述.....	8
	顯示敘述.....	12
	迴圈敘述及副程式.....	15
	圖形敘述.....	18
	比較敘述.....	21
	數學及邏輯關係的先後順序.....	21
	字串及函數.....	22
	特別重要的幾個 BASIC 敘述.....	24
<b>第二章</b>	<b>BASIC 與機器語言間的聯繫</b> .....	30
	記憶體的使用.....	34
	記憶體圖 ( 硬式的 APPLESOFT 存在 ROM ).....	35
	一個簡單的 BASIC 作業系統.....	40
	完整的 BASIC 作業系統.....	55
<b>第三章</b>	<b>指令碼的格式</b> .....	60
	數字系統.....	62
	計算器.....	69
	記憶體中的指令.....	74
	使用 BASIC 作業系統.....	75

<b>第四章 簡單的圖形</b> .....	94
在螢光幕上繪出一個點.....	95
描在四個角的位置.....	104
畫一條水平線.....	108
繪出垂直線.....	111
繪出長方型.....	116
<b>第五章 顯示文字資料</b> .....	123
顯示出一個字.....	123
新指令的討論.....	129
顯示所有的英文字母.....	138
顯示 ASCII 碼.....	141
執行這個程式.....	145
顯示更多的文字行.....	146
<b>第六章 APPLE的聲音</b> .....	163
對BASIC作業系統的修改.....	165
TONE EXPERIMENT PROGRAM的描述.....	167
執行這個程式.....	170
第二段及第三段如何工作.....	176
新的副程式.....	178
新的指令.....	179
<b>第七章 聲音及圖形更進一步地研究</b> .....	185
輸入及執行這個程式.....	191
使用鍵盤彈奏單音.....	193
程式的描述.....	196

輸入及執行這個程式·····	201
<b>第八章  APPLE的監督系統</b> ·····	<b>209</b>
處理機的情況暫存器·····	218
減法·····	224
<b>第九章  更多的有效位數及負數值</b> ·····	<b>233</b>
兩個數元組的加法·····	234
兩個數元組的減法·····	238
猜數字的遊戲·····	244
新使用的指令·····	259
新使用的副程式·····	259
<b>第十章  更多監督系統的功能</b> ·····	<b>263</b>
十六進位的加法——立即式·····	263
十六進位的減法·····	267
十進位的算術·····	270
檢查及改變暫存器·····	283
新的指令·····	291
新的監督系統的命令及使用的方法·····	292
<b>第十一章  MINI-ASSEMBLER和定址模式</b> ·····	<b>296</b>
使用MINI-ASSEMBLER·····	298
腳註的位置表示方法·····	305
第二百零頁的腳註·····	306
絕對腳註的位置·····	312
腳註的間接位置·····	317
間接的腳註位置·····	326



<b>第十二章 總 論</b> .....	337
八個數元的乘法.....	338
直接由 BASIC 執行乘法.....	345
使用 BASIC 作業系統作乘法.....	346
使用監督系統作乘法.....	350
使用 MINI-ASSEMBLER 作乘法.....	352
八個數元的除法.....	354
<b>附錄 A-1 BASIC 敘述</b> .....	366
A - 2 機器語言指令.....	367
A - 3 即有的副程式.....	369
A - 4 顯示幕的符號.....	369
<b>附錄 B 負的十進位數字相對應的十六進位值</b> .....	370
<b>附錄 C-1 顯示幕的記憶體位置</b> .....	371
C - 2 顯示幕的 ASC II 碼.....	373
C - 3 低解析度圖形的顏色碼.....	374
<b>附錄 D 6502指令碼</b> .....	375
附表 港、台与内地计算机名词术语对照	B 1

# 第一章

## APPLE SOFT II

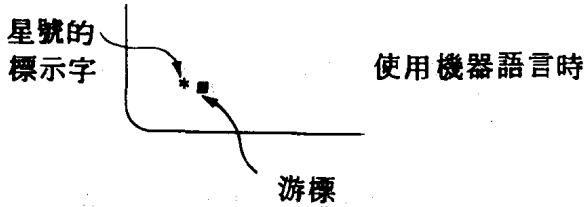
### BASIC 的回顧

#### BASIC

在讀這本書之前有許多事情您必須先完成，我們覺得由於有許多種不同版本的 Apple 電腦，這些工作是必要的。

1. 您已經將硬體方面的連接完成了，如果您尚未做完這個工作，您必須參考 Apple 電腦提供的 reference manuals。
2. 您的機器上具有：
  - a. Applesoft II BASIC 存在可插入的綫路板的 ROM 內。
  - b. 卡上具有選擇 Applesoft II 及整數 BASIC ( Integer BASIC ) 語言的開關。
3. 對您的機器您懂得如何去使用並參考它的手冊。
4. 您必須懂得如何在不同的程式語言中做調換的工作，並使用它們。

Apple 電腦可以瞭解許多種的程式語言，標示字 ( prompt character ) 指出目前 Apple 所能瞭解的語言；星號 ( \* ) 表示您正處於機器語言 ( machine language ) 的情況下，這個語言永遠存在電腦內，您不必從外在的設備輸入它。至於機器語言的監督系統 ( monitor ) 我們將在第八章中介紹。



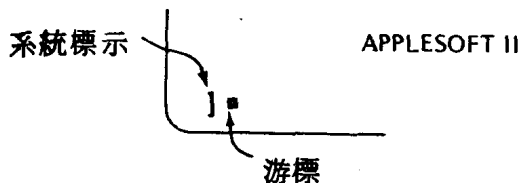
如果您的 Applesoft 存在插入的綫路板的 ROM 內，那麼您也擁有永遠存在 ROM 內的整數 BASIC 語言。至於 ROM 的意義就是“ Read Only Memory ”——僅讀記憶體，只能被“讀”而不能被改變。整數 BASIC 的標示字是“>”，在本書中我們並不討論整數 BASIC，您若想得到較多的資料，您可以參考 Apple II BASIC Programming Manual 。

Applesoft II 是 Apple 功能較多的 BASIC 語言，Applesoft 的標示字是“]”，這種 BASIC 可以在下面三種型式中使用：

1. Apple II Plus 系統並具有自動啓動監督系統 ( Autostart Monitor ) 的 ROM 。
2. Applesoft 的介面卡。
3. Apple 的語言系統 ( Language System ) 。

Apple II Plus 系統的 Applesoft II BASIC 存在 ROM 內，因此，Apple 的 Mini-Assembler、浮點程式 ( Floating Point Package ) 及 SWEET-16 編譯程式 ( 存在整數 BASIC 的 ROM 內 ) 就無法在 Apple II Plus 系統中被使用了。

由於我們即將在本書稍後的地方討論 Mini-Assembler，因此我們將把焦點放在 Applesoft II 的 ROM 卡上，而不是在 Apple II Plus 系統上。

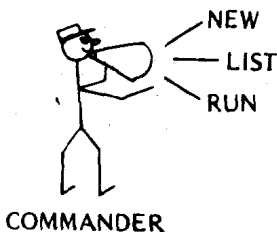


這本書設計的目的在於使您能夠由設計 BASIC 語言的程式而至於設計機器語言的程式，因此這本書假設您對於 Applesoft II BASIC 具有相當的瞭解，我們在這一章中將僅對 BASIC 敘述做一個大略的介紹，如果您對 Applesoft II BASIC 已經有足夠的認識，那麼您可以往下閱讀第二章，否則，您就必須對這一章做一些研究。

雖然我們在這一章中所介紹的並不能涵蓋所有 Applesoft 的功能，但是所有必須具備的敘述及命令都會提供給您的。至於書中介紹的示範程式及練習都必須有 16K RAM 的 Apple 電腦、磁帶機或磁碟機、以及 Applesoft BASIC 語言才能操作。

## 命 令

許多基本的命令在設計、除錯及執行一個程式時都是必須的，我們在這裏討論的是 NEW、LIST、RUN、TEXT、GR、LOAD、SAVE、CONT、TRACE 及 NOTRACE 命令。



NEW —— 這個命令將存在電腦記憶體內的程式全部消除。它不僅將目前的程式清除，並把這個程式所產生的變數也清除乾淨。這個命令在新程式輸入之前要先使用。

範例：

```
10 LET M = 50
20 PRINT M
30 LET M = M+1
40 IF M<60 THEN GOTO 20
50 END
```

NEW

當您打入這個命令並  
按下 RETURN 鍵

哇！所有的東西都不見了！

]

LIST —— 這個命令將目前的程式顯示在螢光幕上，在下面的例子中有幾種不同的型式說明，但都基於電腦中存有程式的假設。

範例：

1. 打入：LIST 並按下 RETURN 鍵。

整個的程式將被顯示，如果這個程式很長，那麼在螢幕被填滿之後將往上捲讓出空間。

2. 打入：LIST 20,100

或者

LIST 20-100 及按 RETURN 鍵。

這個命令會使由行號 20 到 100 的程式行被顯示出來。

3. 打入：LIST -150 並按 RETURN 鍵。

這個命令會將由程式起頭開始到行號 150 結束的程式行顯示出來。

4. 打入：LIST 150- 並按 RETURN 鍵。

這個命令會將由程式行 150 開始到結束的程式列顯示出來。

5. 打入：LIST 150 並按 RETURN 鍵。

這個命令只會將行號 150 的程式行顯示出來。

您可以使用 CTRL S 使得顯示的情形暫時停止，然後又可以使用 CTRL S 使得動作繼續，這使得您可以檢查部份的程式列；顯示的情形可以由 CTRL C 打斷，但是您就無法再由被打斷的地方重新開始顯示的動作了，除非您再使用 LIST 命令時指出停止程式行的行號，才能達到這樣的效果。

**RUN** —— 這個命令使電腦去執行一個存在記憶體內的程式；所有的變數都將被清為零而程式由具有最低行號的程式行開始執行（除非在 RUN 的後面跟著開始的行號，就如同範例 2 一般）。

#### 範例：

1. 打入：RUN 並按 RETURN 鍵。

程式將由最低行號的程式行開始執行。

2. 打入：RUN 200 並按 RETURN 鍵。

程式將由行號 200 的程式行開始執行。

## 6 APPLE 機器語言

**TEXT** —— 這個命令將螢光幕的顯示設定為文字資料 ( text )，顯示幕的大小為 24 行而每行 40 個字，這是 Applesoft II BASIC 被處理時的正常情況。當您想由顯示圖形的情形返回顯示文字的情形時，您才需要使用這個命令，它也可以在程式中做為一個改變顯示圖形為顯示文字的敘述 ( statement )。

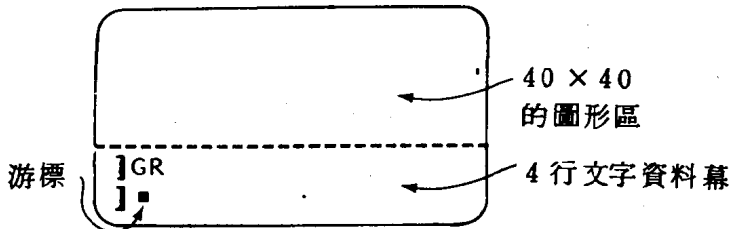
**範例：**

打入：TEXT 並按 RETURN 鍵。

**GR** —— 這個命令將螢光幕的顯示情形設定為低解析度圖形 ( low resolution graphics ) 的顯示。當您使用這個命令時，就有 40 乘 40 個格子供您繪圖，而螢光幕的底色就被清為黑色，同時游標 ( cursor ) 被移往螢幕下端的四行文字資料幕 ( text window ) 的起始處，這時圖形所使用的顏色也被設定為黑色 ( COLOR = 0 )，因此當您想要顯示圖形的時候，您必須設定其他的 COLOR 值。

**範例：**

打入：GR 並按 RETURN 鍵。



**LOAD** —— 這個命令使得電腦將一個程式由磁帶讀入 Apple 的


記憶體內。您必須將磁帶機準備妥當（將磁帶轉到程式的起始處，並且按下 PLAY 鍵），然後您才可以輸入 LOAD 命令。當 Apple 在磁帶上發現資料時，它會發出“嗶”的一聲，而當程式完全存入記憶體內時，您將聽到第二聲的“嗶”，這時候 Applesoft 的系統標示就會出現在螢光幕上。LOAD 命令在執行的時候，您只能使用 RESET 鍵或將電腦關掉才能打斷執行的動作。

**範例：**

將您的磁帶機準備妥當，然後

打入：LOAD 並按 RETURN 鍵。

當結束後，螢光幕將照下面一般地顯示：



```
LOAD
|  █ ← 指出成功地 LOAD 完程式
```

**SAVE** —— 這個命令將存在記憶體內的程式存入磁帶內，您必須在 SAVE 命令被執行前先按下磁帶機上的 RECORD 及 PLAY 鍵；同樣地，您在命令開始及結束時都將聽到“嗶”的一聲。

**範例：**

準備好您的磁帶機，然後

打入：SAVE 並按 RETURN 鍵。

**CONT** —— 如果程式的執行被 STOP、END 或 CTRL C 打斷，那麼您就可以使用這個命令使程式由被打斷的下一



個指令開始繼續被執行，這時並沒有任何的指令或變數被清除掉。如果您的程式被修正過、增加或消除某些程式行，或是由於停止程式的執行而得到錯誤訊息 ( *error message* )，那麼您就不可以使用 CONT 命令。

**範例：**

打入：CONT 並按 RETURN 鍵。

TRACE —— 這個命令在您對程式除錯 ( *debugging* ) 時使用，它會使被執行的程式行的行號顯示在螢光幕上，使您能夠觀察程式的執行是否依照正確的順序。TRACE 的功能可以由 NOTRACE 命令結束。

**範例：**

打入：TRACE 並按 RETURN 鍵。

然後打入 RUN 看看程式執行的順序。

NOTRACE —— 這個命令將 TRACE 的功能結束。

**範例：**

打入：NOTRACE 並按 RETURN 。

當這個程式再被執行時，就不會有行號的顯示了。

## 設定敘述

有許多種的方法可以使資料 ( 數值 ( *numeric* ) 的或是字串 (