

組合語言編輯程式活用系列(二)

APPLE II 組合語言編輯程式

TOOL KIT

活用法

徐寶龍 編著



5313
2706

APPLE II 組合語言編輯程式 TOOL KIT 活用法

徐寶龍編著

源 文 書 局
PO WEN BOOK CO.
Publisher & Bookseller
Specializing in Chinese Studies
234 Wanchai Rd. Q/F, EP H.K.
P. O. Box 23066, Wanchai H.K.

版權所有 • 請勿翻印

**APPLE II 組合語言編輯程式
TOOL KIT 活用法**

著者：徐寶龍

出版：萬年出版社

九龍新蒲崗興利工廈9字樓

印刷：萬年出版社印刷部

九龍新蒲崗興利工廈9字樓

定價：港幣H.K.\$17.00

目 錄

第 1 章	概 論	1
1-1	系統說明	1
1-2	編輯程式	2
1-3	組合語言譯碼程式	3
1-4	使用本系統前的準備工作	3
1-5	系統的啟動	4
第 2 章	編輯程式	8
2-1	前言	8
2-2	如何使用編輯程式	9
2-3	命令模式的功能	10
2-4	參數的語法	15
2-5	編輯命令	17
2-6	處理磁片和磁帶的命令	32
2-7	操作命令 (Operating Command)	37
第 3 章	組合語言譯碼程式 (Assembler)	46
3-1	前言	46
3-2	如何使用組合語言譯碼程式	49
3-3	控制譯碼組合程序命令	51
3-4	原始程式的格式	53
3-5	組合語言譯碼程式的假指令	58
3-6	控制列表的假指令	63
3-7	定義資料的假指令	66
3-8	有條件性譯碼組合之假指令	68
3-9	符號表的顯示	70
附錄 A	編輯程式／組合語言譯碼程式使用的記憶體及用途	74
附錄 B	編輯程式顯示的 DOS 錯誤	77
附錄 C	再定位載入程式 (RELOCATING LOADER)	80
附錄 D	組合語言譯碼程式的 DOS 錯誤碼 (DOS ERROR CODES) " OOPS "	83
附錄 E	目的檔的格式	85
附錄 F	標記符號表 (Symbol table) 的格式	88
附錄 G	BASIC 程式的編輯	90

第 1 章

概論

1-1系統說明

Apple II 組合語言譯碼程式／編校程式系統 (Assembler / Editor System) 提供編輯 (Editing) 和將 6502 組合語言程式予以譯碼組合 (Assembling) 的功能。此一系統由三個程式所構成：編輯程式 (Editor)、組合語言譯碼程式 (Assembler) 和命令釋義程式 (Command Interpreter)。使用者可經由命令釋義程式來呼叫編輯程式或組合語言譯碼程式。系統啟動後，命令釋義程式一直在記憶體中。螢幕上任一行的開頭有出現提示字元 (Prompt Character) ' '：'則表示編輯程式也在記憶體中。當使用者透過 ASM 命令，要求系統將使用者的 6502 組合語言程式予以譯碼組合時，系統就會將組合語言譯碼程式從磁碟機讀入，取代在記憶體中的編輯程式，而在編輯緩衝區中經過編輯的檔案也將被清洗掉。一旦譯碼組合的程序完成後

，系統將再度把編輯程式從磁碟機中讀入，取代在記憶中的組合語言譯碼程式。

此系統的使用，需要具有 32K 記憶體之 Apple II 或 Apple II plus 和至少一部磁碟機。如果要經過譯碼組合的組合語言程式相當大時，則需 48K 的記憶體和另外一部來讀取原始檔磁片的磁碟機。

組合語言譯碼程式／編輯程式 系統存於 Applesoft Tool Kit 磁片，此磁片包含 16-Sector DOS，此系統 DOS 檔的結構跟 Applesoft 和 Integer BASIC 具有共通性 (Compatibility)，但無法與 Apple PASCAL 系統共通，Apple PASCAL 有自己的編輯程式和組合語言譯碼程式。

系統可分為三種模式 (Mode)：

- 1 命令模式 (Command Mode)
- 2 輸入模式 (Input Mode)
- 3 編輯模式 (Editing Mode)

當系統在命令模式時，則表示命令釋義程式準備接受使用者的命令來執行相關的動作。在輸入模式時，則表示使用者可以經由編輯程式在編輯緩衝區裏建立原始程式。在編輯模式時，使用者可經由編輯程式來編輯或修改在編輯緩衝區裏的原始程式。

1-2 編輯程式

編輯程式是讓使用者用來建立能符合組合語言譯碼程式所要求的格式之原始程式 (Source Program)，或使用編輯命令來編輯修改原始程式。此一編輯程式亦可用來建立 EXEC 檔和用來

編輯或建立 BASIC 程式。有一點須注意的是，它並不是爲了提供文字編輯功能而設計的。它能夠處理磁帶的檔案，且經由 DOS 亦可處理磁碟的檔案。

1-3組合語言譯碼程式

組合語言譯碼程式能提供譯碼組合 (Assembly) 的功能，也就是說，它能夠將使用者所指定的文字檔 (text file) 中之原始程式轉換爲 6502 機器碼 (Machine Code)。它從磁碟中讀取使用者所指定的原始程式，然後將譯碼組合後的結果寫入磁碟中。它僅保留 Symbol table 在記憶體裏面，如果須要話，RLD (Relocation dictionary) 也可保留。因此，此一組合語言譯碼程式不會因記憶空間的關係，而使得譯碼組合的規模受到太大的限制。所以它能夠處理相當大的 6502 組合語言程式。

這個組合語言譯碼程式執行著 Two-Pass 的譯碼組合程序。它亦提供有條件譯碼組合 (Conditional Assembly) 的功能給使用者。組合語言譯碼程式平常並不在記憶體裏，只有在使用者使用 ASM 的命令來要求系統將使用者的原始程式轉換爲 6502 機器碼時，系統才會將它從磁碟機中讀入，一旦工作完畢，它將不存在記憶體裏。

1-4使用本系統前的準備工作

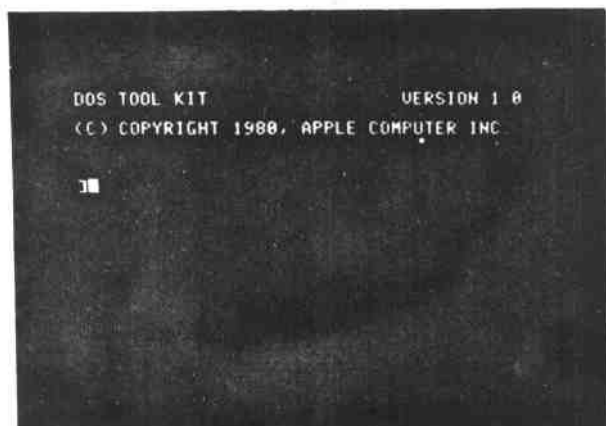
在未熟悉本系統之前，而想嘗試使用編輯程式或組合語言譯碼程式時，最好先將 Applesoft Tool Kit 磁片複製一份，以

防止因操作不當而將 Applesoft Tool Kit 磁片破壞，致使工作無法完成。如果你只有一部磁碟機，那你可利用 Apple Utility 磁片中的檔案複製程式 FID (File Duplicator) 來做拷貝的工作。

這本使用手冊是假設此系統的使用者熟悉 6502 組合語言和 Apple DOS 而寫成的，如果你對於 Apple DOS 不熟悉的話，最好先閱 DOS 手冊，然後再使用本系統。

1-5 系統的啟動

將 Applesoft Tool Kit 磁片放入磁碟中，然後啟動 (boot) 磁碟機，啟動後，螢幕會出現：



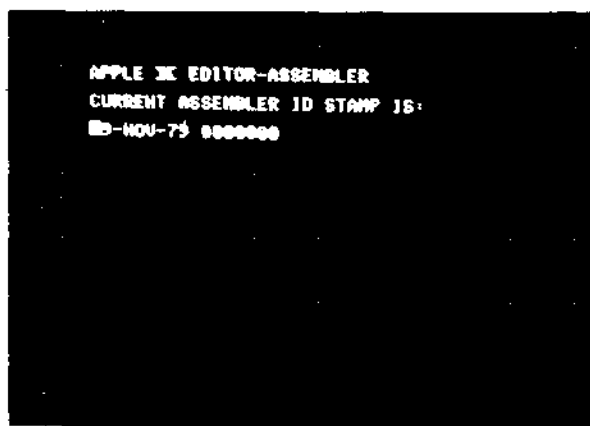
然後從鍵盤按入

RUN EDASM

然後按 RETURN 鍵。如果你的 Apple 只有 Integer BASIC 的話，則須按入

RUN INTEDASM

然後按 RETURN 鍵。螢幕就會顯示出組合語言譯碼程式的 ID Stamp



此 ID Stamp 可依使用者的意思予以更改。直接按 RETURN 鍵，則原先的 ID Stamp 沒有更改。如果使用者希望更改 ID Stamp，則可利用 ← 和 → 鍵來移動游標至欲更改字元的位置，然後按入新的字元以代替原先的字元，一旦將欲更改的字元全部更改完畢後，按 RETURN 鍵，系統會將新的 ID Stamp 存入磁片，代替舊的 ID Stamp，此時螢幕顯示的 ID Stamp 為新的 ID Stamp。（此一 ID Stamp 包含空格，一共可容納 17 個字元，如果更改後的 ID Stamp 超過 17 個字元，則系統僅將前面的 17 個字元存入磁片中，以後顯示出的 ID Stamp 也只有這 17

個字元) 例如, 使用者希望新的 ID Stamp 為

29-NOV-82 #000000

利用→鍵移動游標至原來的 ID Stamp 中字元 7 的位置, 然後按入 82, 再按 RETURN 鍵, 則新的 ID Stamp 將存入磁片, 而且螢幕顯示出新的 ID Stamp。

系統將新的 ID Stamp 存入磁片後, 螢幕會顯示出



螢幕底的左邊有一個字元 “:”, 此一字元稱為提示字元 (Prompt Character), 閃動的底綫 (underline) 為游標。系統顯示在螢幕上任一行的第一個字元為提示字元 “:”, 則表示命令釋義器準備接受命令。當使用者按入命令和 RETURN 後, 命令釋義程式將執行相關動作。

注意: Applesoft Tool Kit 磁片須放在 Driver 1, 如此, 命令釋義程式才有辦法讀到組合語言譯碼程式或編輯程式。

如果 DOS 已經在記憶體中, 而且你準備使用組合語言譯碼

程式或編輯程式時，只要將 Applesoft Tool Kit 磁片放入 Driver 1，然後按 RUN EDASM或RUN INTEDASM 以及 RETURN鍵，這樣即可進入本系統，而不需要重新啓動 (reboot) 磁片中的DOS。如果不想改變 ID Stamp，則按入

BRUN EDASM.OBJ

然後按 RETURN即可。

第 2 章

編輯程式

2-1前言

系統在螢幕上任何一行 (line) 的開頭顯示出提示字元 “ : ” , 則表示命令釋義程式準備接受命令, 也就是說系統處於命令模式。而這時, 編輯程式亦在記憶體中, 隨時可以進行編輯工作。如果按入的命令前面、後面或命令字元串中間有空格存在, 命令釋義程式仍能接受, 因為命令釋義程式會將這些空格予以忽略。這一章所討論的各種命令皆以字元串表示, 字元串中包含有大寫及小寫的字元。字元串是用來表達命令的意義。但命令的輸入並不須要將這整個字元串按入, 而只須按入大寫字元的部分, 小寫字元可以省略。例如, COpy 此命令是要求系統將某一段程式複製一份, 而這命令的輸入只須從鍵盤按入 CO, 而 py 可以省略不按。

編輯程式並沒有將行號 (line number) 存於編輯檔中。

它定義在檔中兩個RETURN間的內容為一行 (line)，而且在每一行的前面加上行號，然後將其顯示於螢光幕上，加上行號主要目的是使編輯工作方便進行。當被編輯檔中有某些的行被刪除，則接在被刪除行後面的行之行號將自動遞減，如果有某些行插入被編輯檔中，則接在插入行後的行之行號會自動地增加。

編輯程式經由Tab命令，提供了16個可調整的表格定位設定 (TAB Setting)，以使得text容易閱讀。編輯程式的輸出裝置為40 Column的螢幕顯示，而命令的輸入是從鍵盤輸入，亦可使用外加的終端裝置，但輸出程式會將每40字列為一行。欲編輯的檔 (Editing File) 存於編輯緩衝區，此一緩衝區的大小跟所使用的Apple有所不同。擁有48K RAM的Apple，其緩衝區的大小為29K，32K RAM的Apple，其緩衝區的大小為13K。以48K的Apple而言，其緩衝區大概可容納700行至1400行的原始程式。典型的6502組合語言，包括註解，一行大約有40個字，而在一個程式裏，有些指令很短，而有些沒有註解。

本系統的編輯程式並不是為了用來做文字處理 (Word Processing) 而設計的。其設計的主要目的是用來發展程式。它亦可用來編輯Applesoft或Integer BASIC程式。讀者可參考附錄G。

2-2如何使用編輯程式

前面已經提過，編輯程式有兩種功能，一為讓使用者在緩衝區建立原始程式，另一為讓使用者來編輯修正在編輯緩衝區中的欲編輯檔或原始程式。

使用者可以利用 2-5 節中的 Add 命令在編輯緩衝區建立新的原始程式。Add 命令的執行，使系統處於輸入模式此時使用者可建立新的原始程式。一旦原始程式建立完畢後，按入 CTRL-D 或 CTRL-Q，可以使系統回至命令模式。這時候，使用者可以使用 2-6 節中的 SAVE 或 TSAVE 命令，將在編輯緩衝區中的原始程式存入磁片或磁帶。如果新建立的原始程式有需要編輯修正時，使用者可以利用 2-5 節中的各種編輯命令來做編輯的工作。

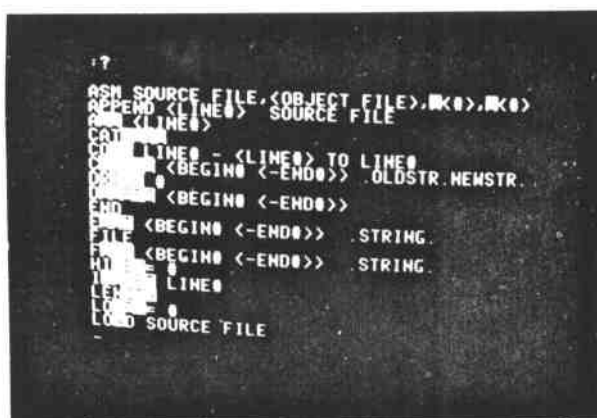
如果使用者所要編輯修正的原始程式並不在編輯緩衝區內，而是存在磁片或磁帶中，在這種情況時，使用者可以利用 2-6 節中的 LOAD 或 TLOAD 命令，將存於磁片或磁帶中的原始程式讀入編輯緩衝區內，然後使用 2-5 節的各種編輯命令來做編輯修正的工作。

2-3 命令模式的功能

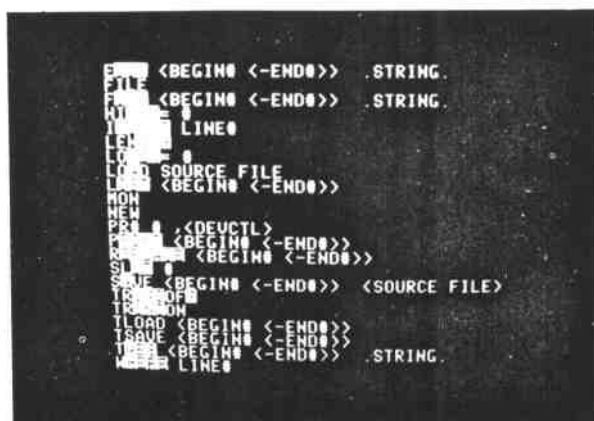
使用者可以使用 Help 命令來要求系統將各種命令及各個命令的語法 (Syntax) 顯示在螢幕上。從鍵盤按入

?

然後按 RETURN，螢幕上會顯示出各種命令及其語法：



全部的命令需要二次螢幕顯示，再按任何鍵，就可將其餘的命令顯示在螢幕上：



顯示在螢幕上的命令字元串中，用黑底白字顯示的字元為必需字

元，亦即，要求系統執行此命令，須要將必須字元按入。以黑字白底顯示的字元為可以省略的字元。例如，DELETE 命令，輸入此命令只須按入必須字元 D，DELETE 可以省略。螢幕僅顯示各種命令後面的參數之一般型式。有關各命令參數之更詳細語法，讀者可查閱下面各節中各個命令的說明。

命令模式有許多功能使得系統的使用更為方便，這些功能敘述如下：

(1) 允許數個命令同時輸入

這一功能允許使用者在一行中按入數個命令。命令釋義程式會將這些命令依照順序儲存起來，每當一個命令執行完畢後，命令釋義程式會取出下一個命令來執行。在按入命令時，每個命令之間需用命令標點 (Command Delimeter) 來分開。此一標點通常設定為冒號“:”，使用者可以改變命令標點為另外一種符號。任何命令語法錯誤將會使命令釋義程式清除儲存著的其他命令。然而，Find, Change, Edit 命令如發生找不到字元串的情形，系統並不將這種情況視為命令的錯誤。在儲存著的命令中，ASM 將是最後被執行的命令，因為，一旦組合語言譯碼程式被讀入時，任何跟在 ASM 後面的命令將會被清除。

Add, Insert, Copy, Delete 和 Replace 命令會使得被編輯程式某些行的行號受到影響而改變。因此，使用者須注意，當會改變行號的命令執行後，而儲存著尚未執行的命令後面之行號參數是否仍為原來使用者欲處理那行之行號，例如：

```
1  LDA  # $ 34
2  STA  POA
3  STA  POB
```

```

4  LDA  POC
5  STA  BUFFER
6  STA  REG

```

我們希望將第3行刪除，然後將第1行COpy至第6行前面，數個命令同時輸入為

```
: D 3 : CO 1 TO 5
```

因為刪除第3行的命令先執行，這時原先的第6行變為第5行，因此COpy的參數行號須為5而不是6。如果先執行COpy的命令後，再執行刪除的命令，則此問題便可以解決。

這時，命令輸入應為

```
: CO 1 TO 6 : D 3
```

(2)命令標點的設定(Command Delimeter Set)

這一功能允許使用者改變命令標點的符號。在使用者沒有改變命令標點之符號時，機定(default)的命令標點符號為“:”，因為冒號一般在6502組合語言程式並沒有使用到。Find, Change和Edit命令所尋找的字元串不可含有與命令標點一樣的符號字元。否則，命令釋義程式會將此符號字元視為命令標點，因此執行的結果會有錯誤產生。所以命令標點設定的功能，使這種錯誤得以避免。

更改命令標點的方法為：按ESC鍵，然後按冒號“:”，再按新的命令標點字元，新的標點就會顯示在螢幕上。如果按入新的命令標點字元為空格，則喇叭會嗶一聲，系統希望你重新按入另一個命令標點字元。如果按入新的命令標點字元為RETURN鍵時，則系統會將目前使用的命令標點字元顯示在螢幕上，然後回到命令模式。任何字元皆可設定為命令標點字元。