

Apple II

檔案處理入門與應用

蔡人育 編著



專業出版社出版

Apple II

檔案處理入門與應用

蔡人育 編著

專業出版社出版

Apple II 檔案處理入門與應用

編著者：蔡人育

出版兼：專業出版社
發行者

九龍大角咀橡樹街836號

承印者：聯基印刷廠

香港仔偉倫工業大廈九樓

定價：H. K. \$ 16.00

序

目前許多人已經使用微電腦，自行設計程式來幫助他處理工作上的業務，而市面上已經出版的微電腦叢書多半只偏重於語言的指令介紹，對於檔案處理都缺乏詳盡的描述，及如何應用的實例。

本書針對此點與讀者之需要，將如何應用檔案結構來做資料處理的技巧，由入門的觀念介紹到實例的應用做一系統化的講解。

本書最後一章指標順序檔 (Index Sequence File) 之應用副程式 (p. 128 ~ p. 129)，為作者個人經驗之應用技巧，已被臺北佳佳科技股份有限公司所採用，現展現於諸讀者之前，供各位參考應用。

因目前微電腦以APPLE II 機器較普遍，故本書採用APPLE BASIC來做實例解說。本書主要是描述檔案的觀念與應用，所以讀者如採用其它的微電腦機器，只要修改例中的部份指令，即可靈活地運用您的檔案來做資料處理了。

蔡 人 育 謹識

目 錄

序.....	I
第一章 資料儲存概念與相關指令介紹.....	1
• DOS 如何將資料儲存於磁碟上.....	1
• DOS 指令介紹.....	2
• 一般立即式命令.....	2
• 磁碟資料檔指令.....	7
• 順序檔指令.....	7
• 隨機檔指令.....	10
第二章 檔案概述.....	15
• 檔案 (File)	15
• 記錄 (RECORD)	15
• 欄位 (Field)	16
• 欄位長度 (Field Length)	17
• 記錄長度 (RECORD Length)	17
• 關鍵欄位 (RECORD KEY)	18
• 一個磁片能夠存放多少筆記錄計算方法.....	18
• 檔案規格設計練習.....	19
第三章 順序檔.....	21

• 順序檔資料存入概念.....	21
• 順序檔資料讀取概念.....	22
• 實例應用.....	23
第四章 隨機檔.....	33
• 概念.....	33
• 新資料加入.....	35
• 列印磁碟檔案.....	39
• 更改磁碟檔案資料.....	41
• 刪除磁碟檔案資料.....	44
• 實例應用.....	54
• 系統概述.....	54
• 庫存檔案規格.....	58
• 新材料基本資料加入.....	59
• 刪除原有檔中之材料資料.....	62
• 更改檔案中之資料欄.....	64
• 進貨處理.....	66
• 出貨處理.....	68
• 庫存報表列印.....	69
• 低於安全庫存量表列印.....	72
• 由螢幕查詢庫存資料.....	74
• 程式串接介紹.....	77
第五章 二分法結構檔.....	79
• 概念.....	79
• 應用副程式使用介紹.....	86
產生一個二分法的空檔.....	86

• 實例應用.....	95
• 產生一個庫存檔.....	97
• 庫存資料查詢.....	105
• 庫存報表列印.....	110
第六章 指標順序檔.....	115
• 概念.....	115
• 應用副程式使用介紹.....	120
• 產生一組指標順序檔的空檔.....	120
• 開檔.....	121
• 關檔.....	122
• 加入新資料.....	123
• 尋找資料.....	124
• 刪除資料.....	126
• 尋找檔案中第一筆資料.....	127
• 尋找檔案中最後一筆資料.....	128
• 尋找下一個 KEY 的資料.....	129
• 尋找上一個 KEY 的資料錄.....	130
• 所有副程式彙整.....	131
• 實例應用.....	133
• 庫存資料新增 / 刪除 / 修改.....	133
• 庫存資料查詢.....	138
• 庫存報表列印.....	144

第一章

資料儲存概念與相關指令介紹

1-1 DOS 如何將資料儲存於磁碟上

磁 軌

一個 DISK II 5¼" 磁片，共可儲存 104,360 個 BYTES，爲了要能迅速找到其中一個 BYTE，APPLE 的 DOS 將磁片分爲 35 條磁軌（TRACKS），分別編號爲 0，1，2，……，34，每一條磁軌實際上均爲一個同心圓，依序分佈於磁片上。

磁 區

爲了要使某一個 BYTE 更快被找到，DOS 將磁軌又分爲 16 個磁區（SECTOR），每一個磁區內含有 256 個 BYTES。對於某一個 BYTES 而言，如果將它限制在某個磁軌及磁區內，那麼 DOS 只需要找一個磁區就可以了；因此，處理某一個 BYTE 或一串的 BYTES 就幾乎是非常快速了，而如此磁碟區才能真正的發揮效用。

資料存入磁片程序

資料存入或自磁碟中取出，都是由 DOS 控制的，當您將資料寫入一個檔案中時，有幾件事情將會發生：

1. DOS 到磁碟索引檔中尋找這個檔的名稱（在您的程式中，您必給每一個檔一個名字，這一點我們將在稍後描述）。
2. 如果這個檔的名字找到了，DOS 就會從適當的磁區內讀出 256 個 BYTE 的資料，並把它們存在記憶體的一個叫做緩衝區（BUFFER）的區域內，如果名字沒有被找到，或是這個檔的這個磁區未寫入資料，那麼 DOS 將把緩衝區內全部放滿了零。
3. 最多 256 個 BYTE 的資料可以被寫進緩衝區，如果資料不足 256 個 BYTE，那麼寫進緩衝區內時，以前所遺留的資料仍然存在。
4. 如果將剛好 256 個 BYTE 的資料寫入緩衝區內時，那麼緩衝區內的資料被寫回適當的磁區中。
5. 第三及第四步驟將重複地被執行，直到所有要寫入這個檔的資料，都已經被寫到磁片上為止。
6. 所有的資料都已被寫入磁片上之後，DOS 將把磁軌 / 磁區表及索引檔內的記錄做一個修正。

1-2 DOS 指令介紹

1. 一般立即式指令

INIT 指令

在您將資料或程式存入一個空磁片之前，此磁片必須先被設定為

DOS 可接受的格式，設定格式時，任何原先在磁片上的資料將被消除，所以您在設定磁片格式時，必須確定磁片上面的訊息都是您所不再需要保存的。

當您使用 INIT 指令時，必會將目前記憶體內的程式存入磁片內，這個程式在開機時馬上就會自動執行。

```
例：INIT START
      INIT START, D1
      INIT START, D1, S6
      INIT START, D1, S6, V001
```

上述為一典型的例子，“START”為一個程式名字，即執行 INIT 時，記憶體中的程式名字，且設定開機後馬上自動執行“START”這個程式。

“D1”：表磁碟機所在的 DRIVER 位置，可省略。

“S6”：表磁碟機所在的 SLOT 位置，可省略。

“V001”：表將這磁片定為“001”號，如不指定，則電腦自動設為“254”號。

SAVE指令

SAVE 指令將目前記憶體中的資料（即用 LIST 可以顯示出的程式）存入磁片中。

```
例：SAVE PROGRAM1
      SAVE PROGRAM1, D1
      SAVE PROGRAM1, D1, S6
      SAVE PROGRAM1, D1, S6, V001
```

“PROGRAM1”為使用者給與的程式名稱，除了以下二規則外，可以使用任何名字：

1. 總長不可超過 30 位字元。

4 檔案處理入門與應用

2 第一位必須非數字。

如果原來磁片中已有“PROGRAM1”這個程式，則原來的程式內容將被新存入的程式取代。

BSAVE 與 SAVE 同，但僅用於二進位檔。

DELETE 指令

DELETE 指令可以將已存入磁片中的程式刪除掉。

例：DELETE PROGRA 1
DELETE PROGRAM1, D1
DELETE PROGRAM1, D1, S6
DELETE PROGRAM1, D1, S6, V001

LOCK 指令

如您想永久保留存入磁片中的程式或資料，不被 DELETE 指令刪除或 SAVE 指令更換，則可用 LOCK 指令將之鎖住。

例：LOCK PROGRAM1
LOCK PROGRAM1, D1
LOCK PROGRAM1, D1, S6
LOCK PROGRAM1, D1, S6, V001

UNLOCK 指令

當您欲更換或刪除一個被鎖住的檔時，可先用 UNLOCK 將之解開，再用 SAVE 指令或 DELETE 指令將之更換或刪除。

例：UNLOCK PROGRAM1
UNLOCK PROGRAM1, D1
UNLOCK PROGRAM1, D1, S6
UNLOCK PROGRAM1, D1, S6, V001

CATALOG指令

CATALOG 指令可以將磁片中所有的程式・資料名稱顯示出來。

例：CATALOG
 CATALOG D1
 CATALOG D1, S6
 CATALOG D1, S6, V001

DISK VOLUME 001

A	003	START
B	113	MAGIC WINDOW
*I	015	APPLE SOFT
T	020	INVENTORY
*A	019	TEDITOR
A	012	ADD/DELETE/MODIFY

“*”表示被 LOCK 的程式或資料。

“A”、“B”、“T”表檔的型式。

數字表示此檔所佔磁片的空間大小，單位為 SECTOR。

每一 SECTOR = 256 BYTES，一個磁片共有 560 個 SECTOR。

數字之後則為此檔之名字。

碼	檔 型 式
A	APPLESOFT 程式檔
B	二進位檔
T	文字資料檔
I	整數 BASIC 程式
R	可被執行的二進位檔

LOAD指令

LOAD 指令由磁片上讀入一個程式，放在記憶體內，

例：LOAD PROGRAM1
LOAD PROGRAM1, D1
LOAD PROGRAM1, D1, S6
LOAD PROGRAM1, D1, S6, V001

如不為程式檔則會發生檔案型式不合的錯誤信號。

BLOAD與LOAD同，但只用於二進位檔。

RUN指令

使用 LOAD 指令之後，按 RUN，則此程式就執行，而此二個步驟可合為一個步驟，即在 RUN 之後加上程式名字即可。

例：RUN PROGRAM1
RUN PROGRAM1, D1
RUN PROGRAM1, D1, S6
RUN PROGRAM1, D1, S6, V001

則“PROGRAM1”這個程式會被 LOAD 進記憶體內，且馬上會自動執行。

BRUN與RUN同，但只用於二進位檔。

RENAME指令

使用 RENAME 指令，可將一個檔名更改成另一個新檔名，代替原來的舊檔名。

例：RENAME PROGRAM1 PROGRAM2
RENAME PROGRAM1, PROGRAM2, D1
RENAME PROGRAM1, PROGRAM2, D1, S6
RENAME PROGRAM1, PROGRAM2, D1, S6, V001

RENAME 指令會將不檢查新的檔名是否已經存在 DISK 中，如果已經存在，則會發生二個同名字的檔。

VERIFY指令

VERIFY 指令可以檢查存入磁片中的檔是否完整，

```
例： VERIFY PROGRAM1
      VERIFY PROGRAM1, D1
      VERIFY PROGRAM1, D1, S6
      VERIFY PROGRAM1, D1, S6, V001
```

如果不完整，則會出現 I/O ERROR 錯誤信號。

2 磁碟資料檔指令

如果要在程式中用 DOS 的指令，則要在每個指令之前加 CHR\$(4)，即 CTL-D。

EX:

```
10 PRINT CHR$(4) ; "CATALOG"
```

順序檔指令

OPEN指令

將一個磁碟的資料檔打開，準備供以後將資料讀出或寫入。

EX:

```
10 D$=CHR$(4);REM CTL-D
20 PRINT D$;"OPEN INVENTORY, D1, S6, V001"
30 END
```

以上例子是將一個叫“INVENTORY”的庫存檔打開，供以後資料之存取，如此檔不在磁片中，則電腦會自動新建立一個空的“INVENTORY”資料檔供您使用。打開後的讀寫頭指標一定指在該檔的第一筆資料，等候讀或寫。

S6, D1, V001 三者表此檔所在的磁碟機、地址，可省略。

CLOSE指令

已打開的檔使用完後，要用CLOSE將其關閉，否則往往會使檔內的資料失掉，而且可能其他磁碟上的資料關閉後以後要使用，再用OPEN指令將其打開。

EX:

```
10 D$=CHR$(4):REM CTL-D
20 PRINT D$;"OPEN INVENTORY"
30 PRINT D$;"CLOSE INVENTORY"
40 END
```

以上是將“INVENTORY”檔打開，再關閉。

CLOSE指令不用指定該檔在那一地址，故SLOT、DISK、VOL均可省略，因OPEN時均已定出該檔在那一地址了。

如CLOSE後不加檔案名字，則會將所有檔案關閉。

EX:

```
10 D$=CHR$(4)
20 PRINT D$;"OPEN INVENTORY"
30 PRINT D$;"OPEN CUSTOMER"
40 PRINT D$;"CLOSE"
50 END
```

以上是將“INVENTORY”、“CUSTOMER”兩檔依序打開，再一起關閉。

WRITE指令

用 WRITE 與 PRINT 指令將資料寫入磁片的資料檔中，但寫入之前必須先將該資料檔打開。

EX:

```
10 D$=CHR$(4)
20 PRINT D$;"OPEN INVENTORY"
30 PRINT D$;"WRITE INVENTORY"
40 PRINT "PART-NO-1"
50 PRINT D$;"CLOSE INVENTORY"
60 END
```

上例為 "INVENTORY" 檔打開後，存入 "PART-NO-1" 的資料，再將 "INVENTORY" 關閉。

WRITE 指令不必指定地址，因 OPEN 時已指定該檔所在磁片的地址了。

WRITE 必須與 PRINT 指令合用。

WRITE 只是宣告準備將資料寫入某一檔案中，接著的 PRINT 指令，才是真正執行存入資料的動作。

READ指令

用 READ 與 INPUT 指令將存入磁片中之資料取出，但同樣必須先將該檔打開。

EX:

```
10 D$=CHR$(4)
20 PRINT D$;"OPEN INVENTORY"
30 PRINT D$;"READ INVENTORY"
40 INPUT A1$
50 PRINT D$;"CLOSE"
60 END
```

上例將“INVENTORY”檔打開，並將第一筆資料讀出存入 A1\$ 變數中。

READ 必須與 INPUT 指令合用。

READ 只是宣告準備讀出資料，接下來的 INPUT 才是真正執行讀出資料，存入變數的動作。

READ 不必指定該檔之地址，因 OPEN 時已指定該檔之地址了。

如打開的檔以前沒有存入資料，而此次欲讀出資料時，會因沒有資料可讀而出錯，或是一直讀到該檔的最後時，再讀下去，同樣會發生沒資料可讀而出錯，欲阻止該情況發生，可用 ONERR GO TO 的指令防止。

Ex:

```

5 ONERR GOTO 100
10 D$=CHR$(4)
20 PRINT D$;"OPEN INVENTORY"
30 PRINT D$;"READ INVENTORY"
40 INPUT A1$:REM INPUT FROM DISK
50 PRINT D$ :REM CLOSE DOS COMMAND
60 PRINT A1$:REM PRINT TO CRT
70 GOTO 30 :REM READ NEXT RECORD
100 PRINT D$;"CLOSE INVENTORY"
110 END

```

說明：(5) 先宣告檔案沒資料可讀出錯時就到 (100) 去，將必關閉結束。

(50) 將 (40) 宣告之 DOS 命令結束，否則接下來的 (60) 不會寫到螢幕上，而會寫到磁片中，讀者應特別注意。

隨機檔指令

OPEN指令

與順序檔之 OPEN 相似，但格式稍有不同，必須先指定每筆資