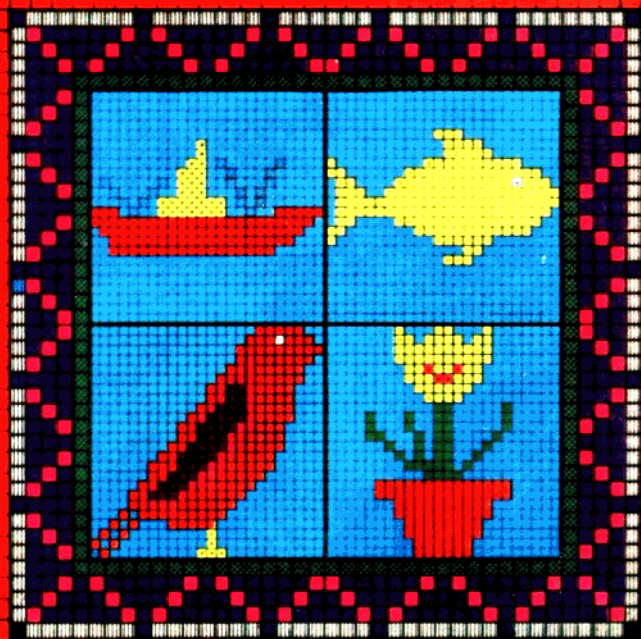


# Apple

## 低解像度繪圖秘笈

時新出版社 蔡文能 譯



# Apple

低解像度繪圖秘笈

蔡文能 譯

時新出版社

APPLE 低解像度繪圖秘笈

---

編 者： 蔡 文 能  
出 版： 時 新 出 版 社  
發 行： 香港中環卑利街 498號二樓  
印 刷 者： 廣 源 印 務 局  
青山道875號工廠大廈

---

定價：港幣

波 文 書 局  
PO WEN BOOK CO.  
Publisher & Bookseller  
Specializing in Chinese Studies  
234 Wanchai Rd. G/FI. E2 H.K.  
P. O. Box 23066, Wanchai H.K.

# 前言

本“秘笈”旨在滿足想迅速學會用電腦繪圖者的需要，讓他們能利用簡單的方法繪出低解像度（low resolution）的電腦圖形。

只要一兩個小時的工夫，您就能在螢光幕上繪出相當複雜的圖形。您不但能了解繪圖方法的細節，而且也可以知道如何保存圖形元件（element），就像保存一個私人的繪圖“程式圖書館”一般。當您的繪圖館庫愈充實後，您便能利用以前的“圖形元件”來組成新的圖樣，設計新的圖形。

透過這本書，您也將學到如何利用藝術字母及數字來顯示像廣告牌大字報的訊息。您也可以利用字幕與圖像交織顯現動態連環故事，也可用來製作新奇的廣告圖像或其它美妙的藝術作品。

本書不僅教您繪圖技巧，還包括了入門者的各種符號圖庫、圖形元件、花邊圖、背景圖等等，真可說是一本包羅萬象的“秘笈”。

這是有創意的秘笈，您可以隨心所欲地擴充您既有的創意！

## 簡 介

本書是日前出版的“低解像度繪圖簡介”(Introduction to Low Resolution Graphics)之續集。在該書中，作者為大家介紹了有關低解像度繪圖的基本概念及一些簡單應用，在大多數的家用級微電腦上大都能使用。所謂低解像度(Low Resolution; 低解析度)意思是說在整個螢幕上的“點”大約只有一千點上下(想想看，此時的圖形是不是要遠遠地看比較像)，故稱為低解像度。

本書是專為APPLE家用電腦而寫的。為什麼？因為目前APPLE II幾乎是最流行的家用電腦，不論美國、日本、臺灣、歐洲均是如此。光是臺灣，製造APPLE仿製品者至少超過一百家，連小教授2號、小神通、寶教一號、統一微電腦等也均是“與APPLE通用”。APPLE不但具低解像度繪圖能力，還具有高解像度繪圖能力；而且還有彩色。這使得用者更能發揮創造力。本書特別針對APPLE II的特性而寫，使APPLE的繪圖能力發揮得淋漓盡致。

雖說針對APPLE，不過本書所論之基本概念及原理亦適用於其它各型家用微電腦，尤其是能力類似APPLE者；因此也可供其它機器的使用者參考之用。

在這本小集子中，我們將介紹繪圖形的一般方法，以及用低解像度圖形來表達物體及圖案。程式不用複雜的組合語言或機器語言，反之，完全使用最通俗的BASIC語言。

介紹過一般方法後，我們提供了大約100個的繪圖程式館庫，包括背景圖、物品、各式圖形、格式等等。所以，有了這本秘笈，讀者立可派上用場。利用各種背景、物品、圖形、格式等，使用者便可從中選取一些加以組合成各式各樣的圖案。

然而，更重要的是讀者更能學到如何擴充自己的“圖形館庫”。也許讀者也可為本書提供一些新花樣，以便日後再版時加入“圖形館庫”中。本書提供的通用方法，最重要的一點是只要用此法弄出來的圖案彼此均可拿來互相組合使用。

記住，練習是最好的學習方式，光說不做是沒有用的，所以，立刻開始，將本書中的範例程式搬入您的電腦中，試試其圖形，並試著繪出自己的圖形及字樣，與其他同好交換心得，共享其樂！

# 目 錄

|                           |    |
|---------------------------|----|
| 前言.....                   | i  |
| 簡介.....                   | v  |
| 第 1 章 一個萬用的低解像度繪圖副程式..... | 1  |
| 第 2 章 繪圖資料庫.....          | 9  |
| 第 3 章 產生字符集.....          | 51 |

## 第一章

### 一個萬用的低解像度繪圖副程式

# A General Purpose Low Resolution Drawing Routine

在“低解像度繪圖簡介”(Introduction to Low Resolution Graphics)一書中，作者曾經介紹一個副程式，只要給予任意兩點即可繪出一條低解像度的直線。理論上，由此我們即可繪出任何我們所要的圖形，而你所要做的只是提供所要畫出之線的兩端點即可。

然而，就實用的觀點而言，這種“土法煉鋼式”的方法仍是有些許的不便。即使是個小圖案，如果要在程式中寫下如“X = 這”以及“Y = 那”，則也要寫好多指述行，這是多麼沉悶的工作！而在鍵盤上敲進去更是一件令人昏倒的事，縱使是一位打字老手，在字母和數字之間換來換去也是一件很不愉快的事（註：各種鍵盤數字鍵的位置常不相同）。

儘管如此，但這個概念卻是非常簡單而有用。任何圖案均可想像成由某種顏色的線（或點）組成。易言之，要繪圖，只要用適當的顏色，在顯示幕上畫上必要的線即可，不必去弄高深的數學（當然數學是有用的），你所要做的只是指出螢幕上每一線段的起點及終點。

我們一定要找個折衷的辦法——在維持這方法的簡單性之下，使輸入（打字）的工作減至最少。

事實上，圖中有個簡單的關係可被用來減少要“指明”的“點”，那就是——在多數情況下，一線的終點即為另一線的起點！因此，如果我們“記住”最後畫過的那點（將會成為下一線的起點），則在繪圖過程中幾乎可以省掉一半的

描述點位置之工作（因為有些線會被其它的線隔開）。

使用 DATA 指述可以省去很多麻煩，如此就不必再說了一大堆的“X = 這”及“Y = 那”。放在 DATA 指述行再用 READ 指述便能——將它們讀入電腦，如此一來，說不定程式可縮短到原來的五分之一呢！

例如，要畫個邊長為 8 單位長度的正方形，如圖 1 所示，則程式可能如下：

```
X1=0:Y1=0:X2=7:Y2=0  
GOSUB 'DRAW LINE'  
X1=7:Y1=0:X2=7:Y2=7  
GOSUB 'DRAW LINE'  
X1=7:Y1=7:X2=0:Y2=7  
GOSUB 'DRAW LINE'  
X1=0:Y1=7:X2=0:Y2=0  
GOSUB 'DRAW LINE'
```

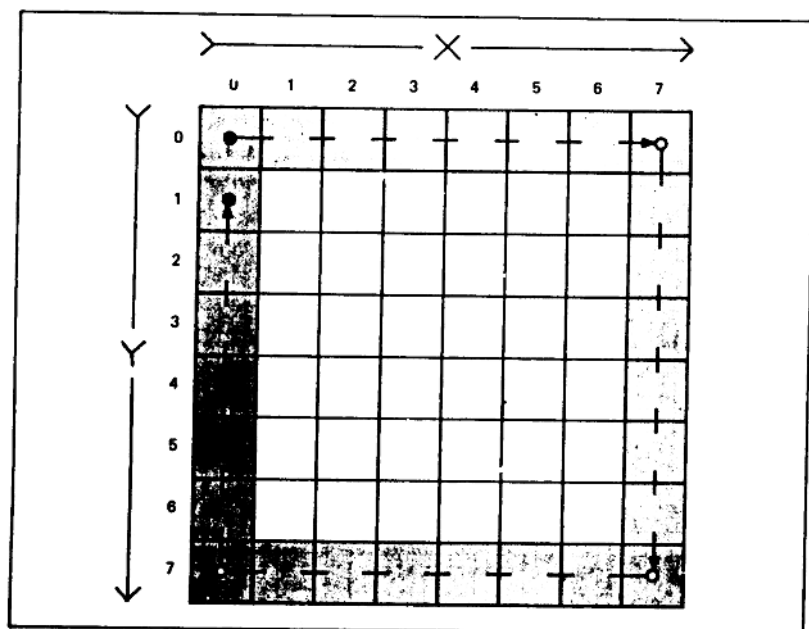


圖 1-1 用四條線段便組成一個正方形。注意每一線段的結束即為另一線段的開始。對於 APPLE 而言螢幕左上角的座標即為 ( 0 , 0 )。

然而如果用 DATA 指述，並假設一線之結束即為另一線之起點（當然畫線的副程式要知道這點），則程式可能只要兩行：

```
GOSUB 'DRAW LINES'  
DATA 0,0,7,0,7,7,0,7,0,0,
```

在 DATA 指述行中最右邊的星號表示畫線結束的特別代碼，例如用 - 1，只要畫圖副程式了解即可。

只要稍加修改，繪圖副程式便可被設計成在 DATA 指述中指明線的顏色，如此每條線的起點均可看出來了。這個副程式如列表 1 所示。

在該程式列表中，真正繪圖副程式始自行號 4000，這程式完全依照 APPLE-SOFT BASIC 的語法，可以在 APPLE 類似的機器上（如小教授 2 號、小神通、金寶島 2 號等）畫出低解像度圖形。此繪圖副程式（4000~4320 行）又再叫用（Call）從 5000 行到 5140 行的畫線副程式。該畫線副程式極為簡單，在低解像度繪圖簡介一書中我們已討論過。

接著讓我們來討論此通用副程式如何工作。首先，行號 4000 將由 DATA 指述中提取（fetch）數據值，並根據該值決定要做任何事？記住，於 APPLE 中，低解像度圖形頁的 X 軸只有 40 點，座標表示為 0 到 39，若值介於 0 至 39 則為正確的 X 軸表示值，由行號 4010 執行後便會跳到 4100 去讀取 Y 座標，否則繼續 4020 行。

在 4020 行可看出，當 X 值介於 40 至 49 時，即表示另一新線段的開始，雖然用 40 到 49 均可以，但通常我們在此書中都用 44 來表示，目的在求易記。

接著的 4030 行指出若值為 50 到 59 則表示要換繪圖之顏色，在此書中我們用 55 表示，也是為了好記而已。

再下去的 4040 行顯示出任何大於 59 的值均將表示整個圖形繪完了。本書則以 66 來作代表。

再回憶一下，X 值 44 表示新線段要開始，55 表示要換顏色，66 表示圖畫完了。

程式中的 4100 行至 4140 行是用來繪出由 X1, Y1 到 X2, Y2 所形成的直線

，此處是假設  $X_1$ ， $Y_1$  為前一線段之終點，而  $X_2$  即為剛由 4000 行讀入的值  $XP$  加上位移值  $X_0$  而得， $X_0$  是讓繪圖者用以指明圖形所在的位置，稍後會再詳述。至於  $Y_2$  則是由 4100 行讀入的  $YP$  值加上位移  $Y_0$  而得。

由行號 5000 開始的副程式才是真正畫下  $X_1$ ， $Y_1$  到  $X_2$ ， $Y_2$  線段的程式。畫完此線之後，由程式中的 4130 行負責將現在的終點“設定”為下一線的起點即令  $X_1$  變為  $X_2$  值， $Y_1$  變為  $Y_2$  值。接著，程式控制轉向 4000 行，繼續下一線段的繪圖工作。

程式的 4200 行至 4230 行是用來做畫新線段的初始工作，當新線段的起點不是前一線之終點時，例如整個圖的第一筆，或是圖形不連續數都會用上。實際上在此只是畫出一點而已。

再看程式的 4300 行到 4320 行，這是用來指明畫圖之顏色而已，當然，它也是由 DATA 指述中讀入的。

如何繪出完整圖形？很簡單，只要指出要用何顏色，第一線段之起點（ $X$  及  $Y$  座標值）及各線之終點，而這些都經由 DATA 指述來完成。

例如，要畫個深藍色的框框，只要寫出如下的 DATA 行：

DATA 55,2,44,0,0,7,0,7,0,7,0,1,66

這可解釋如下：

第一個值 55 表示下一值為顏色。在 APPLE 之低解像度圖形上有 16 種顏色，分由 0 至 15 表示：

- |                       |                  |
|-----------------------|------------------|
| 0 = 黑色 (Black)        | 8 = 咖啡色 (Brown)  |
| 1 = 紫紅色 (Magenta)     | 9 = 橘紅色 (Orange) |
| 2 = 深藍色 (Dark Blue)   | 10 = 灰色 2 (Gray) |
| 3 = 紫色 (Purple)       | 11 = 淡紅色 (Pink)  |
| 4 = 深綠色 (Dark Green)  | 12 = 綠色 (Green)  |
| 5 = 灰色 1 (Gray)       | 13 = 黃色 (Yellow) |
| 6 = 中藍色 (Medrum Blue) | 14 = 碧綠 (Aqua)   |
| 7 = 淡藍色 (Light Blue)  | 15 = 白色 (White)  |

在此，55 之後的數字為 2，表示深藍色。第 3 個數字為 44，表示新線段

的開始，這通常用於整個圖形剛開始之時，故接著的（0，0）即為起點，也就是第一線段的起點之X及Y值。再接下去的7與0表示第一線段之終點以及第2線段之起始；如此，直到66出現後，整個圖便告完成。

這個圖可以繪於螢幕上的任何地方，只要改變X0及Y0之值即可。例如，要畫在整個畫面大約中央處，則可將X0及Y0之值均定為16即可。

### 【簡單範例】

如果你將程式列表中的4000行到5140行抄入電腦中，並加入如下一些指述行，則便可在螢幕中間繪出一深藍色的框框。

```
10 GR:CALL -936
20 X0=16:Y0=16
30 GOSUB 4000
40 END
..
..
6000 DATA 55,2,44,0,0,7,0,7,7,0,7,0,1,66
```

如果您想讓該框框在隔一些時間後拭掉，則只要再加入如下四行即可：

```
40 FOR I=1 to 2000:NEXT I
50 GOSUB 4000
60 END
..
..
6010 DATA 55,0,44,0,0,7,0,7,7,0,7,0,1,66
```

如果讀者想更深入了解此法的奧妙，不妨將下列程式全部敲入您的電腦中，最好能用彩色的顯示器（或連接電視機），執行該程式看看有何效果。

• 現在是否完全了解了？太好了！下一章將給您一大堆非常有用的繪圖館庫，您可以用它來創造您自己的任何圖案。

### 程式列表 3-1

|                    |                                 |
|--------------------|---------------------------------|
| 10 GR : CALL - 936 | 160 X0 = 17:Y0 = 2              |
| 20 X0 = 0:Y0 = 0   | 170 GOSUB 4000                  |
| 30 GOSUB 4000      | 180 X0 = 18:Y0 = 5              |
| 40 X0 = 16:Y0 = 11 | 190 GOSUB 4000                  |
| 50 GOSUB 4000      | 200 X0 = 19:Y0 = 8              |
| 60 X0 = 0:Y0 = 12  | 210 GOSUB 4000                  |
| 70 GOSUB 4000      | 220 END                         |
| 80 X0 = 0:Y0 = 25  | 4000 READ XP: REM GET X COORDI- |
| 90 GOSUB 4000      | HATE                            |

```

100 X0 = 22:Y0 = 32
110 GOSUB 4000
120 X0 = 34:Y0 = 32
130 GOSUB 4000
140 X0 = 1:Y0 = 23
150 GOSUB 4000
4030 IF XP < 60 THEN 4300: REM
      WHEN X BETWEEN 50 AND 59
      CHANGE SEGMENT COLOR
4040 RETURN : REM IF REACH HERE
      THEN FINISHED DRAWING SHAPE

4100 READ YP: REM GET Y COORDI-
      NATE
4110 X2 = X0 + XP:Y2 = Y0 + YP: REM
      SET UP COORDINATES FOR END
      OF SEGMENT
4120 GOSUB 5000: REM DRAW THE
      SEGMENT FROM X1,Y1 TO X2,Y2
4130 X1 = X2:Y1 = Y2: REM MARK
      BEGINNING OF NEXT SEGMENT
4140 GOTO 4000: REM CONTINUE
      DRAWING
4200 READ XP: READ YP: REM GET
      COORDINATES FOR START OF A
      NEW SEGMENT
4210 X1 = X0 + XP:Y1 = Y0 + YP: REM
      ADD IN OFFSET VALUES
4220 PLOT X1,Y1: REM MARK START
      OF SEGMENT
4230 GOTO 4000: REM CONTINUE
      DRAWING
4300 READ C: REM GET COLOR CODE

4310 COLOR= C: REM SET COLOR OF
      SEGMENT
4320 GOTO 4000: REM CONTINUE
      DRAWING
5000 IF X2 > X1 THEN A = 1
5010 IF X2 < X1 THEN A = - 1
5020 IF X2 = X1 THEN 5070
5030 FOR X = X1 TO X2 STEP A
5040 Y = INT (((Y2 - Y1) / (X2 -
      X1)) * (X - X1)) + .5) + Y1
5050 PLOT X,Y
5060 NEXT X
5070 IF Y2 > Y1 THEN B = 1
5080 IF Y2 < Y1 THEN B = - 1
5090 IF Y2 = Y1 THEN 5140
5100 FOR Y = Y1 TO Y2 STEP B
5110 X = INT (((X2 - X1) / (Y2 - Y1)) * .5) + X1
5120 PLOT X,Y
5130 NEXT Y

```

```

4010 IF XP < 40 THEN 4100: REM
      IF VALUE LESS THAN 40 HAVE
      VALID PLOTTING POINT
4020 IF XP < 50 THEN 4200: REM
      WHEN X BETWEEN 40 AND 49
      START A NEW SEGMENT
      ,10,39,11,0,11,0,12,39,12,39
      ,13
6005 DATA 0,13,0,14,39,14,39,15
      ,0,15,0,16,26,16,55,12,44,23
      ,19,39,19,39,20,22,20,21,21,
      39,21,39,22,16,22,16,23,39,2
      ,3,39,24,0,24,0,25,39,25,39,2
      ,0,26,0,27,39,27,39,28,0,28

6010 DATA 0,29,39,29,39,30,0,30
      ,0,31,39,31,39,32,0,32,0,33,
      39,33,39,34,0,34,0,35,39,35,
      39,36,0,36,0,37,39,37,39,38,
      0,38,0,39,39,39,55,15,44,10,
      0,21,0,21,1,8,1
6015 DATA 8,2,21,2,44,9,3,19,3,
      ,44,33,2,38,2,38,3,32,3,31,4,
      37,4,44,31,5,35,5,44,0,10,1,
      10,2,11,0,11,44,24,10,28,10,
      27,11,24,11,55,4,44,11,8,13,
      8,44,0,9,15,9,44
6020 DATA 6,10,17,10,18,11,5,11
      ,4,12,19,12,20,13,3,13,2,14,
      21,14,22,15,1,15,0,16,23,16,
      24,17,0,17,0,18,25,18,44,0,1
      9,22,19,21,20,0,20,0,21,20,2
      1,44,0,22,18,22,44,0,23,15
6025 DATA 23,44,0,24,7,24,44,0,
      25,3,25,44,0,26,1,26,0,27,55
      ,3,44,37,12,39,12,39,13,33,1
      3,44,39,14,31,14,44,39,15,29
      ,15,44,39,16,27,16,44,39,17,
      25,17,26,18,39,18,39,19,28,1
      9
6030 DATA 66: REM BACKGROUND #
      1
6050 DATA 55,8,44,0,15,0,7,3,4,
      20,4,23,7,44,4,5,6,7,44,1,16
      ,2,16,3,17,4,17,5,18,6,18,7,
      19,23,19,23,8,7,8,7,18,55,8,
      44,5,5,20,5,21,6,6,6,7,7,22,
      7,55,5,44,1,15
6060 DATA 1,7,2,6,2,15,3,16,3,5
      ,4,6,4,16,5,17,5,7,6,8,6,17,
      44,8,9,22,9,22,10,8,10,8,11,
      22,11,22,12,8,12,8,13,22,13,
      22,14,8,14,8,15,22,15,22,16,
      8,16,8,17

```

5140 RETURN  
6090 DATA 55,7,44,0,0,39,0,39,1  
,0,1,0,2,39,2,39,3,0,3,0,4,3  
9,4,39,5,0,5,0,6,39,6,39,7,0  
,7,0,8,39,8,39,9,0,9,0,10,39  
,11,12,10,12,10,11,44,19,12,  
19  
6090 DATA 10,21,10,21,12,20,12,  
20,11,44,19,17,19,15,21,15,2  
1,17,20,17,20,16,55,3,44,14,  
18,14,15,16,15,16,18,15,18,1  
5,16,55,1,44,17,5,17,3,18,3,  
18,5,66: REM LARGE HOUSE  
6100 DATA 55,8,44,7,15,7,9,8,9,  
8,15,44,6,13,5,13,4,12,55,12  
,44,2,10,5,10,5,11,2,11,44,1  
0,9,13,9,14,8,2,8,1,7,15,7,1  
4,12,5,66: REM CRITTER  
6230 DATA 55,8,44,0,5,5,5,44,1,  
0,1,7,4,7,4,0,55,4,44,2,4,3,  
3,2,3,2,2,1,1,55,11,1,0,3,0,  
3,1,66: REM SMALL FLOWER  
6240 DATA 55,8,44,0,5,5,5,44,1,  
6,1,7,4,7,4,6,55,4,44,2,4,3,  
3,2,3,2,2,1,1,55,11,1,0,3,0,  
3,1,66: REM SMALL FLOWER  
6270 DATA 55,12,44,0,3,7,3,6,2,  
8,0,55,5,6,0,5,0,44,2,2,3,2,  
55,2,44,0,5,1,4,2,5,1,6,44,5

6070 DATA 22,17,22,18,8,18,55,2  
,44,2,13,2,12,44,2,10,2,8,44  
,5,14,5,13,44,5,11,5,10,55,7  
,44,9,17,9,15,11,15,11,17,10  
,17,10,16,44,9,12,9,10,11,10  
5,6,0,8,0,5,15,5,15,4,0,4,1,  
3,14  
6110 DATA 3,1,3,44,3,2,14,2,13,  
1,4,1,5,0,11,0,55,11,44,8,1,  
44,13,2,44,4,4,44,10,4,44,2,  
7,44,6,7,44,12,7,66: REM  
TREE  
6120 DATA 55,4,44,11,2,12,3,13,  
4,10,4,10,5,15,5,44,3,6,13,6  
,14,7,0,7,0,8,44,3,0,12,8,12  
,9,3,0,3,11,2,12,4,12,44,12,  
10,12,11,11,12,13,12,55,11,4  
,5,6,4,7,5,6,6,66: REM  
SMALL BICYCLE  
6300 DATA 55,5,44,0,0,1,0,3,2,4  
,2,6,0,7,0,66: REM SHALL  
SEAGULL  
6310 DATA 55,5,44,0,0,1,0,3,2,4  
,2,6,0,7,0,66: REM SHALL  
SEAGULL  
6320 DATA 55,5,44,0,0,1,0,3,2,4  
,2,6,0,7,0,66: REM SHALL  
SEAGULL

附表 1-1 程式列表 1-1 中用到的變數參考表，各數字代表程式行號。如果您想修改該程式，則此表相當有用。

```

*****
*
*
* LISTING 1-1
*
*
* --TABLE OF VARIABLES--
*
*
*****
A - X AXIS DIRECTION INDICATOR
5000 5010 5030
B - Y AXIS DIRECTION INDICATOR
5070 5080 5100
C - COLOR CODE
4300 4310
X - X AXIS PLOTTING COORDINATE
5020 5040 5060 5110 5120
X0 - X AXIS INITIAL OFFSET VALUE
20 40 60 80 100 120 140 160 180 200 4110
4210
X1 - X AXIS STARTING COORDINATE
4170 4210 4220 5000 5010 5020 5030 5040
5040 5110 5110
X2 - X AXIS ENDING COORDINATE
4110 4170 5000 5010 5020 5030 5040 5110
XP - X AXIS RELATIVE DATA POINT
4000 4010 4020 4030 4110 4200 4210
Y - Y AXIS PLOTTING COORDINATE
5040 5050 5100 5110 5120 5130
Y0 - Y AXIS INITIAL OFFSET VALUE
20 40 60 80 100 120 140 160 180 200 4110
4210
Y1 - Y AXIS STARTING COORDINATE
4170 4210 4220 5040 5040 5070 5080 5090
5100 5110 5110
Y2 - Y AXIS ENDING COORDINATE
4110 4170 5040 5070 5080 5090 5100 5110
YP - Y AXIS RELATIVE DATA POINT
4100 4110 4200 4210

```

## 第二章

### 繪圖資料館

# A Library of Drawings

利用前一章的繪圖方法，任何人都可以建立一個繪圖資料館。每當你繪出一個圖樣，便可將其存入該館中。以後要再用到相同圖形時，則只要執行該程式即可，至於圖形在螢幕上的位置則可利用  $X0$  及  $Y0$  之值來控制。

當你建立一適度的繪圖館後，你將發現只要花一點點時間就能繪出一相當生動的圖形（利用該館庫）。

能將一圖輕易地繪於另一圖之上也是此法有趣的一面。只要圖案的大小適中，便能在螢幕上畫出有遠有近的透視圖或風景畫。此法另一個好處在於複雜的組合可以用簡單的圖形組起來；例如，要畫個叢林，只要將一棵樹重複畫它幾次就出來了。

作者在本章中提供了 100 個圖形資料，參見列表 2-1。至於其圖案則在下列各頁中，每個圖正上方的數字即為程式中的行號。因此，若用行號 6000 的數據來畫，則圖形就如前一章的範例，是個小方形。欲將之擺在螢幕任何位置則只消改變  $X0$ ， $Y0$  之值即可，當然這要在召用副程式之前做。例如，令  $X0 = 16$  及  $Y0 = 16$  便會畫在螢幕中央。

作者所提供的圖共分成六大類，如表 2-1 所示，共有 50 個小元件，30 個中型圖案（邊長小於 16 單位長），以及 12 個大元件，另外尚有 4 個全幕花邊、2 個背景圖、和 2 個全像圖。

如何把圖案加到您的繪圖館庫呢？很簡單，依照第一章教的方法，將您的圖

案化為數據即可。一般我們常用方格紙先畫出略圖，用不同的斜影表示要畫不同的顏色。然後所要做的只是將草圖化成“數值”而已。別忘了我們慣用 44 表示新線段要開始了，55 表示要換顏色，而 66 表示畫圖完成。

表 2-1

本章繪圖館庫的圖形共分為六類，行號是對應於程式列表 2-1

| 程式行號      | 類 別           | 圖框大小    |
|-----------|---------------|---------|
| 6000-6490 | 小元件 ( 50 個 )  | 8 X 8   |
| 6500-7090 | 中型元件 ( 30 個 ) | 16 X 16 |
| 7100-7440 | 大型元件 ( 12 個 ) | 24 X 24 |
| 7500-7660 | 全幕花邊 ( 4 個 )  | 40 X 40 |
| 7700-7785 | 背景圖 ( 2 個 )   | 40 X 40 |
| 7800-7894 | 全幕圖形 ( 2 個 )  | 40 X 40 |

將圖“化”成數字並沒有定則，我習慣先由邊開始，先畫出樣子，再將“內部”細細地畫出來。沒有一定的方法能使所要描述的“點”最少，這完全是藝術，要靠天份與經驗，用您最習慣的就可以了。在以後所附各圖中有實線畫出箭頭，那就是作者“化(畫)”的方向，也許您有自己獨到的見解，試試看吧！

有幾點值得注意，這是作者由經驗得來的，想提供列位看官作參考。首先碰到的是該畫個框架就好抑或要填滿？如果圖形要重疊時，只畫框架恐怕容易混淆，所以應慎重選擇。在這個繪圖館庫中的“圖”有些只畫外框，有些則畫實體，在您想組成複雜圖案時應適當選擇。

另外，若您想換顏色則只要改一下顏色代碼即可，甚至您也可以由終端機讀入玩者所要的顏色代碼，這只要稍微修改程式即可。

永遠用同樣的參考座標原點來畫圖有個好處，這使得以後只要提供 X0, Y0 的位移值即可將圖擺於任何地方。在此繪圖館庫之任何圖案均以 ( 0 , 0 ) 為參考點。

如何使用自己的繪圖館庫呢？在程式列表 2-1 中的行號只是供參考而已（與所附圖中號碼對應）。實用上若有大圖案您也許得將其存於檔案中，要繪圖時，從中選取所要的項目組成程式即可。不需用到的就可以不要，以節省記憶體。

因此，從中選取所要的各行後，行號也可適當調整，不必依照原來所列的，