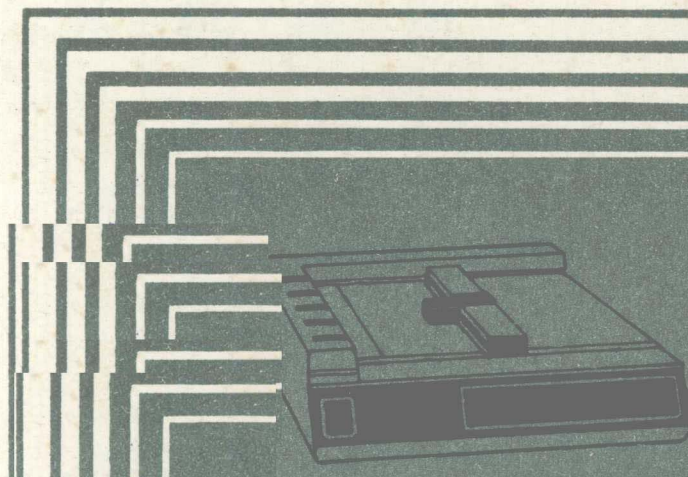


制图课中的 计算机绘图

刘达新编著



BASIC 语言的基本结构

```
10 INPUT A , B2
```

```
20 CD = A * B2/4 - 30
```

```
30 PRINT A , B2:PRINT C
```

```
40 END
```

天津大学
出版社

TH 126
18

制图课中的 计算机绘图

内 容 提 要

本书是为适应国家教育委员会1987年审批的工科各类型制图课程教学基本要求中有关计算机绘图部分的教学需要而编写的。内容包括BASIC语言常用语句及编程方法、图形屏幕显示编程和绘图机绘图编程等。编写中特别考虑到没有学习过BASIC语言且没有使用微型计算机条件的读者的需要。

本书经首届全国制图课计算机绘图教学研讨会讨论并推荐出版。

本书可作为工科大专院校学生学习制图课中计算机绘图部分的教材，也可作为工科职大、业大、中专等校学生的教材，并可供初学计算机绘图的工程技术人员参考。对于没有算法语言基础和微机设备的学校尤为适用。

制图课中的计算机绘图

刘达新 编著

*

天津大学出版社出版

(天津大学内)

天津大学出版社印刷厂印刷

新华书店天津发行所发行

*

开本787×1092毫米^{1/32} 印张：1.125 字数：24千字

1987年10月第一版 1987年10月第一次印刷

印数：1—10000

ISBN 7-5618-0024-X

统一书号：15401·15

定 价：0.30元

目 录

§ 1 微型计算机绘图概述	(1)
(一) 为什么要学习计算机绘图	(1)
(二) 微型计算机绘图系统简介	(1)
§ 2 BASIC 语言简介	(2)
(一) 赋值语句 (LET语句)	(3)
(二) 键盘输入语句 (NPUT语句)	(4)
(三) 数组说明语句 (DIM语句)	(5)
(四) 循环语句 (FOR-NEXT语句)	(5)
(五) 打印 (显示输出) 语句 (PRINT语句)	(7)
(六) 转子程序语句 (GOSUB语句) 和返 回语句 (RETURN语句)	(8)
(七) 终止程序语句 (END语句)	(9)
§ 3 用 BASIC 语言编制程序举例	(10)
(一) 求正六边形角顶坐标值的程序	(10)
(二) 求正三棱锥三视图中各顶点 坐标值的程序	(12)
习题	(15)
§ 4 图形显示简介	(15)
(一) 显示方式和屏幕坐标系	(15)

(二) 常用的几个高分辨率图形显示命令…	(17)
(三) 图形显示编程举例……………	(18)
§ 5 自动绘图机画图简介……………	(22)
(一) SR-6602 绘图机部分常用的绘图命令 (IN、AP、AC、LT、DS、TE) ……	(22)
(二) 绘图机画图编程举例……………	(29)
习题……………	(31)

§1 微型计算机绘图概述

(一) 为什么要学习计算机绘图

计算机绘图是一门新兴学科。近年来微型计算机系统硬件和软件都有很大的改进与提高，微型计算机和个人用绘图机的价格也逐步降低。这些都为使用计算机画图提供了良好的条件。目前计算机绘图已为国内许多企业和科研单位所采用。在国外计算机绘图已被广泛地应用到飞机设计、造船、机械设计、建筑设计、电子线路设计、服装剪裁等许多方面。

计算机绘图可使复杂而重复的绘图劳动简单化。它比用手工画图准确得多，甚至可以画出一些用手工无法绘制的复杂图形。

掌握了制图基本知识以后，学习计算机绘图并不困难。作为一名献身于祖国四化大业的中国大学生，对这门新兴学科应有所了解。

本书简要讲述微型计算机绘图系统和 BASIC 语言以及图形显示和绘图机画图等方面的内容。

书中使用 BASIC 语言，采用的硬件是苹果 II 型微型计算机和 SR-6602 型绘图机。

(二) 微型计算机绘图系统简介

微型计算机绘图系统中的硬件一般包括计算机、显示器、

磁盘驱动器、打印机和绘图机等五个部分。如图 1 所示（图中未画出打印机）。

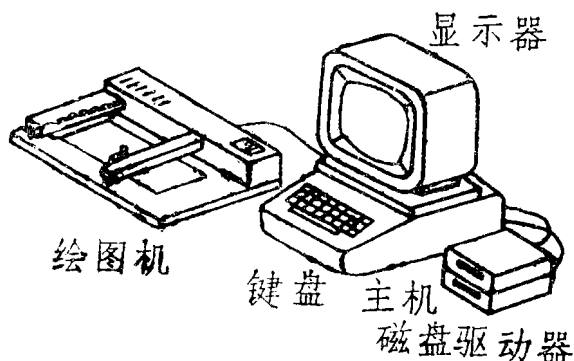


图 1

计算机可以进行数学运算。

显示器在计算机控制下用来显示文字或图形。

磁盘驱动器可在计算机指挥下存储程序和数据，也可向计算机输入程序、数据等各种信息。

打印机可以在计算机的管理下打印出程序或图形。

绘图机可以在计算机控制下画出图形。

键盘是计算机的一个组成部分。它可以把信息输入计算机，是人与计算机进行联系的一种输入工具。

§2 BASIC 语言简介

BASIC 算法语言是计算机常用的高级语言之一。本书仅

介绍与下文画图程序有关的几条语句。

先介绍 BASIC 语言的基本结构。

```
10 INPUT A, B2
20 CD = A * B2 / 4 - 30
30 PRINT A, B2: PRINT CD
40 END
```

上面是用 BASIC 语言编写的程序。

程序要分行写出。每行的开始要写行号（如10、20、…等）计算机运行程序时按行号由小到大逐句执行。

语句中的INPUT、PRINT、END等是BASIC语言中的保留字，不能当作变量名使用。

A、B2、CD等叫做变量名。变量名一定要用英文字母开头，它可以是单一的字母，也可在英文字母后面再跟一个数字或英文字母。

$A \cdot B2/4 - 30$ 叫做表达式，它可包括变量名、常数和运算符等。常见的运算符有：

+（加）、-（减）、·（乘）、/（除）、 $\wedge$ （乘幂）等。

第30行中包括两个语句，它们之间要用“:”号隔开。

（一）赋值语句（LET语句）

格式：

行号 LET 变量名(或数组下标变量) = 数值(或表达式)

句中的“=”号不叫等于号而叫赋值号。

例如：

20 LET C = 6.2832/6（把 2π 弧度的六等分值赋给变量C）

也可以省略语句中的保留字LET，写成：

```
20 C = 6.2832/6
```

又如：

```
20 CD = A * B2/4 - 30
```

把表达式的值赋给变量CD。若 $A = 2$ ， $B2 = 16$ ，则表达式的值为 -22 。也就是把 -22 这个数值赋给变量CD。若用计算机运行第3页中的10~40行程序并给变量A和B2分别赋以2和16，回车后在屏幕上将显示出以下的结果：

```
2           16
-22
```

(二) 键盘输入语句 (INPUT语句)

格式：

行号 INPUT 变量名 或

行号 INPUT “字符串”，变量名

例如：

```
5 INPUT R
```

计算机运行到第5行时，在显示器的屏幕上显示“?”。计算机等待操作者用键盘为变量R输入数值（如键入50）。键入数值后需按RETURN键（回车键），计算机才把刚才键入的数值（50）赋给变量R。

又如：

```
10 INPUT "R = ", R
```

计算机运行到第10行时在屏幕上显示出两个双引号之间的字符串“R =”，它提醒操作者给变量R赋值。这时键入一个

数值，再按 RETURN（回车）键，计算机就把键入的数值赋给变量 R。

（三）数组说明语句（DIM语句）

格式：

行号 DIM 数组名（最大下标）[，数组名（最大下标），…]

例如：

```
30 DIM X(6), Y(6)
```

计算机运行到30行时，定义两个数组X和Y。6是数组内的最大下标。即在X数组中可存放7个数。这七个数分别存放在X(0)，X(1)，X(2)，X(3)，X(4)，X(5)，X(6)等七个数组下标变量之中。X(0)是X数组中第一个数组下标变量，X(6)是第七个数组下标变量。

在程序中若没有写入DIM语句，计算机将自动把数组最大下标值定为10，即从0，1，2，…，10，共可存放11个数值。

最大下标除具体数值外，还可以是变量。例如当变量M和N已经取得某一具体数值后，DIM语句可以写成DIM X(M)，Y(N)。

在一个保留字DIM后可以定义一个或多个数组。定义两个或两个以上数组时，数组名之间要加“，”号。

（四）循环语句（FOR-NEXT语句）

格式：

行号 FOR 循环变量名 = 算术表达式1 TO 算术表达式2

[STEP算术表达式3]

(循环体内各语句)

行号 NEXT 循环变量名

例如:

```

10 INPUT "R=";R
20 C = 6.2832 / 6
30 DIM X (6), Y (6)
60 FOR I = 1 TO 6
70 A = I * C
80 X (I) = R * COS (A)
90 Y (I) = R * SIN (A)
120 NEXT I
150 END
    
```

循环语句
 循环体

例中的60-120行是循环语句。第60行FOR语句是循环开始语句,其中I是循环变量,该变量的初值为1(算术表达式1)、终值为6(算术表达式2)、步长(STEP)为1(算术表达式3)。120行是循环终端语句。这两个语句之间的语句叫作循环体(本例中仅有三句,即70到90行)。步长为1时,在循环开始语句(第60行)中可省略不写STEP 1。

程序运行到60行时,先给循环变量I赋初值1,接着把1的值代入到70、80、90行,使变量A和数组下标变量X(1)、Y(1)各得到一定的值。运行到120行时使循环变量I的值增加一个步长,即 $I = I + 1 = 1 + 1 = 2$,并检查I的当前值2是否大于终值6。若不大于6,则返回60行以下,再一次执行循环体内各语句,即以 $I = 2$ 代入70、80、90行,使A和X(2)、Y(2)各得到另一个值。如此循环六次。第六次循环,在计算出X(6)、Y(6)的值后又运行到120行,此时 $I = I + 1 = 6 + 1 = 7$,

并检查I的当前值是否大于终值，当检查出I的当前值7已大于终值6时，程序就不再执行循环体内的语句而去执行NEXT I以后的语句（本例中为150行END语句）。

若步长为2时，此时需在60行的最后加写上STEP 2字样，则每一次循环中 $I = I + 2$ 。这样，在本例中只能循环三次（得到X(1)、Y(1)；X(3)、Y(3)；X(5)、Y(5)共三组X(I)、Y(I)的值），即结束循环。

（五）打印（显示输出）语句 （PRINT语句）

打印语句的格式很多，这里只介绍较常使用的两种。

格式1：

行号 PRINT 变量名(或数组下标变量)[, 变量名, ...]

格式2：

行号 PRINT “字符串”，变量名[, 变量名, ...]

例如：

```
10 INPUT "R=";R
40 X(0) = R:Y(0) = 0
50 PRINT X(0), Y(0)
300 PRINT "AX, AY=", X(0), Y(0)
400 END
```

程序运行到第10行，若给变量R赋值为100，当运行到50行和300行时就在屏幕上显示出：

```
100          0
AX, AY=100    0
```

上一行显示的内容是两个数 100 和 0，它们分别是数组下标变量 X(0) 和 Y(0) 的值。下一行显示的是字符串 AX, AY = (提示点 A 的 X、Y 坐标值) 和 X(0)、Y(0) 的当前值 100 和 0。

(六) 转子程序语句 (GOSUB 语句) 和 返回语句 (RETURN 语句)

为简化程序，在程序中需多次重复运行的一部分语句可以编成子程序，以备主程序调用。编程时经常把子程序编写在程序结束 (END) 语句以后。子程序的最后一句是返回

(RETURN) 语句，子程序被 GOSUB 语句调用。GOSUB 语句与 RETURN 语句在程序中必需成对出现，否则就出现语法错误。

格式：

转子程序语句

行号 GOSUB 行号 (子程序入口行号)

返回语句

行号 RETURN

例如：

```

140 INPUT "L, H = " L, H
142 GOSUB 260
144 L = L * 2: H = H / 3
146 GOSUB 260
150 END
260 C = L + H: G = SQR (C)
270 PRINT G
280 RETURN
    
```

程序说明如下：

第 260 行中后一句 $G = \text{SQR}(C)$ 是把变量 C 的值开平方，并把平方根的值赋给变量 G 。

程序运行到140行，设给变量 L 和 H 分别赋值为10和6，到142行为转子程序语句，其中的260是子程序入口行号。由此转到子程序的入口（260行）并开始执行260到280行的子程序。运行子程序的结果是显示出4（变量 G 的当前值），然后返回到转子程序语句（142行）以下的144行。经144行给 L 、 H 赋新值（ $L = 20$ ， $H = 2$ ）到146行又转到入口为260行的子程序，再次执行这个子程序。执行的结果是显示出4.69041576（变量 G 的当前值）并返回到转子程序语句（146行）以下的150行。150行是END语句，程序运行至此结束。运行上述程序，在屏幕上将显示出以下内容。

$L, H = 10, 6$

4

4.69041576

（七）终止程序语句（END语句）

格式：

行号 END

终止程序语句的作用是通知计算机结束程序的运行。

例如：

150 END

§3 用 BASIC 语言编制程序举例

(一) 求正六边形角顶坐标值的程序

由图 2 可见, R 是正六边形外接圆半径, 六个顶点的坐标分别由 $(X1, Y1)$ 、 $(X2, Y2)$ 、 $\dots$ $(X6, Y6)$ 来表示。

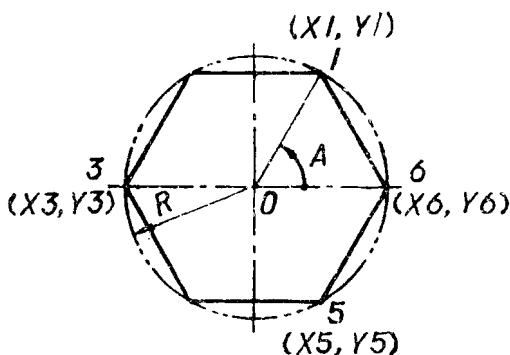


图 2

求各顶点的坐标值, 关键在于定出各顶点和中心 O 的连线与正向水平轴线的各个倾角 A 的数值大小。

下面编制的程序不仅能计算出六个顶点的坐标值, 而且能够把计算的结果全部在屏幕上显示出来。

```
10 INPUT "R=";R
20 C = 6.2882 / 6
30 DIM X ( 6 ) , Y ( 6 )
```

```

60 FOR I = 1 TO 6
70 A = I * C
80 X(I) = R * COS(A)
90 Y(I) = R * SIN(A)
100 PRINT X(I), Y(I)
120 NEXT I
150 END

```

程序说明如下：

第10行要求用键盘给半径变量R赋值。

第20行是把 $2\pi/6$ 的弧度值赋给变量C。

第30行是设定X、Y两个数组，其最大下标均为6，可存放外接圆上六个分点的X和Y坐标值。

第60行是循环开始语句，循环变量I在每次循环中的值是1、2、…6。第70行是计算每一个循环中倾角A的值（由C、 $2 \cdot C$ 、 $3 \cdot C$ 到 $6 \cdot C$ ）。

标准函数COS(A)和SIN(A)中的角度A规定为弧度值。

第80、90行是通过循环语句求出第1、2、…6个角顶的坐标值，它们分别用数组下标变量X(1)，Y(1)；X(2)，Y(2)；…X(6)，Y(6)来代表。

第100行是通过循环语句用屏幕显示X(1)，Y(1)；X(2)，Y(2)；…X(6)，Y(6)（即X1，Y1；X2，Y2；…X6，Y6）的具体数值。

第120行是循环终端语句。给循环变量I赋新值（ $I = I + 1$ ），并检查I的当前值是否大于终值6，以判定是否需要继续循环。

第150行，结束程序运行。

运行上述程序，首先在屏幕上显示“R=”，此时若键入

半径值100，按RETURN（回车）键后，屏幕即显示出各角顶的坐标值。因数值的位数很多，下面只给出经过圆整后的角顶坐标值。

50	86.6
-50	86.6
-100	0
-50	-86.6
50	-86.6
100	0

（二）求正三棱锥三视图中各顶点坐标值的程序

图3中设定坐标原点在图的左上角，X轴向右为正，Y轴

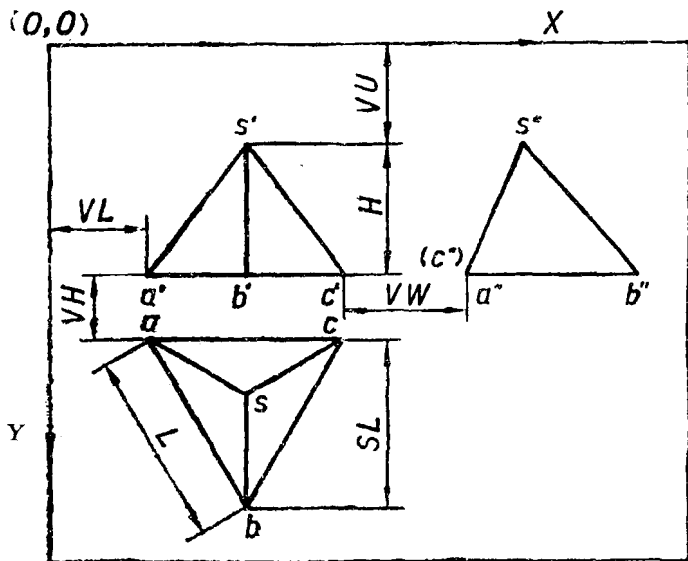


图3