



● 微计算机程序库丛书

李礼贤 译

IBM PC BASIC 程序100例

[美] 杰 弗 布 列 茨 著
约翰克拉克·克莱格

电子工业出版社

12

1

172
57.11

IBM PC

BASIC 程序 100 例

〔美〕杰弗布列茨 著
约翰克拉克·克莱格 译
李 礼 贤

電子工業出版社

006869

内 容 简 介

IBM PC是目前国内外十分流行的一种微计算机。该书收集了约100个可直接在IBM PC上运行的BASIC语言程序。它涉及到教育、绘图、数字计算、通信、事务应用、辅助设计及游戏等各个方面,取材广泛,内容丰富。每个程序都附有简单的说明,介绍其功能及有关的程序设计技巧。这是一本既有实用性,又有参考价值的读物。适合于广大IBM PC用户及有关微计算机教学人员阅读。

100 READY--TO--RUN PROGRAMS AND SUBROUTINES FOR THE IBM PC

JEFF BRETZ

JOHN CLARK CRAIG

TAB BOOKS INC

1983

JS10 38

IBMPC BASIC程序100例

〔美〕杰弗布列茨

著

约翰克拉克·克莱格

李 礼 贤

译

责任编辑:王惠民

※

电子工业出版社出版(北京市万寿路)

新华书店北京发行所发行 各地新华书店经售

沈阳铁路局锦州印刷厂印刷

※

开本: 787×1092 1/16 印张: 17.125 字数: 395.3千字

1986年1月第1版

1986年1月第1次印刷

印 数: 30100 册

定价: 3.50元

统一书号: 15290·83

序 言

书中的大多数程序都是装入内存就可以运行的,但编写本书的目的还是在于介绍一些程序设计技巧。书中每一个程序都有各自的显示技术,技巧和概念。例如,在这些程序中有各种各样的数据输入方法,它们以多种不同的技巧,使用INKEY\$,INPUT,LINE INPUT和其它函数,来进行程序设计。

为了使读者能学习到书中所介绍的技术,这些程序是以公开的,通俗易懂的形式编写的。如果你的IBM PC内存容量有限的话,可以考虑把注释行删掉,也可以将多个语句放在同一行中,把程序重新改造一下。读者可以看出这些程序是通俗易懂的,因为IBM PC BASIC语言是一种很好的程序设计语言;选取这些程序清单时又是经过一番考虑的。这里采用的Microsoft的BASIC版本有突出的优点,它允许使用较长的变量名,使得程序清单容易读懂。例如变量名“DATA.POINTER”比“P2”更能说明变量本身的含意。虽然较长的变量名需要较多的键盘输入时间,但确是一种值得推荐的方法。这样的程序不仅简洁,易读,更重要的是容易使人看明白。

本书中的程序是利用彩色作图插件板进行开发的。其中许多程序是采用SCREEN 0输出方式的,这些程序只需稍加修改或不加修改就能与IBM的高分辨率显示器配合工作,有许多程序利用了IBM PC很强的作图功能,这些程序不能与一般的IBM监视器协同工作。

在开发这些程序时我们得到了许多启发,IBM PC的确是一个很出色的产品,用它来进行工作是很方便的,我们希望这些程序能帮助读者从IBM PC得到乐趣。

目 录

第一章	日历、时钟和恒星时间	(1)
§ 1	日历	(1)
§ 2	时钟	(8)
§ 3	恒星时间	(12)
第二章	教育训练和信息处理	(17)
§ 1	编码的能手	(17)
§ 2	频率	(26)
§ 3	地球同步卫星的定向	(29)
§ 4	欧姆定律	(37)
§ 5	功率计算器	(38)
§ 6	拼单词	(39)
第三章	游戏和其它娱乐程序	(43)
§ 1	迷宫	(43)
§ 2	身体机能的节律	(46)
§ 3	打苍蝇	(51)
§ 4	找单词	(56)
§ 5	登月舱着陆	(59)
§ 6	电传打字机	(67)
§ 7	音响游戏	(68)
§ 8	捉妖怪	(69)
§ 9	堵截游戏	(73)
第四章	作图技巧和演示技巧	(78)
§ 1	高分辨率的作图法	(78)
§ 2	魔棒	(84)
§ 3	图案设计	(85)
§ 4	图案演示	(88)
§ 5	横梁	(109)
§ 6	万花筒	(110)
§ 7	绘画	(111)
§ 8	球体	(112)
§ 9	辐轮	(115)
§ 10	星形	(116)
§ 11	隧道	(118)
§ 12	魔杖	(119)
第五章	数 学	(121)
§ 1	复数	(121)
§ 2	分数	(126)

§ 3	函数分析	(134)
§ 4	三维空间的图形	(140)
§ 5	质数	(147)
§ 6	联立方程式	(148)
§ 7	双精度正弦和余弦子程序	(152)
§ 8	矢量	(154)
第六章	调制解调器通信	(160)
§ 1	异步通信接口驱动程序	(160)
§ 2	通信传送程序: 发送和接收	(167)
第七章	程序设计的辅助程序、子程序和实用程序	(176)
§ 1	修改DOS	(176)
§ 2	写信的程序	(179)
§ 3	画圆的程序	(181)
§ 4	延时程序	(183)
§ 5	利用DATA语句建立文本的简单程序	(184)
§ 6	自动启动过程	(186)
§ 7	彩色监视器的调整	(188)
§ 8	填数据	(190)
§ 9	打印输出存储器内容的程序	(192)
§ 10	文件排序	(194)
§ 11	字体—80	(196)
§ 12	信息中心	(198)
§ 13	新的字体	(200)
§ 14	打印输出	(204)
§ 15	旋转字体	(205)
§ 16	转动A字母	(209)
§ 17	子程序和函数	(211)
§ 18	交叉访问	(218)
第八章	小的事务应用程序	(222)
§ 1	支票收簿支决算程序	(222)
§ 2	作图程序	(234)
§ 3	邮件程序	(251)
§ 4	文本编辑	(259)

第一章 日历、时钟和恒星时间

这一章中的程序,利用IBM PC的作图功能和它的内部时钟,建立了三个用于处理第4度(空间),即平常所谓的时间的程序。

§ 1 日 历

在日历程序中,使用了若干与日历有关的子程序。程序的核心部分是根据已知的日期去查找天文上的儒略日(Julian day)数的子程序,和根据已知的儒略日数去查找日期的子程序。这两个子程序还可以确定所找的日子是星期几,适用的日期范围是从1582年起,到3999年。这些子程序还在本书的其它地方加以详细介绍。

总共有三种算法,它们可以用相应的专用功能键来进行选择:按下F1,可以显示出一月的日历单;按下F2,可以根据已知的日期计算出星期几和其它需要的日期;按下F3,可以计算出两个日期之间的天数。当需要使程序结束时,可按下F4。

打入的日期可以采用任何需要的格式。有一个子程序用于分析打入的日期,并计算出它的月、日和年。下面是子程序能够正确解释的几种日期格式:

July 4, 1776 7 / 4 / 1776
4 JUL 1776 07041776

请注意输入项“7 / 4 / 76”是解释为July 4, 1976。如果没有指明哪个世纪,就被认为是第20世纪。

这些子程序的一种有用的功能,是能检查输入的日期是否有效。它首先把日期变换为它的儒略日数,然后再变换为日期。如果变换的结果和原来的日期相符,则日期是有效的。

图1-1到1-4表示出运行日历程序时,屏幕上所能显示出来的某些格式。

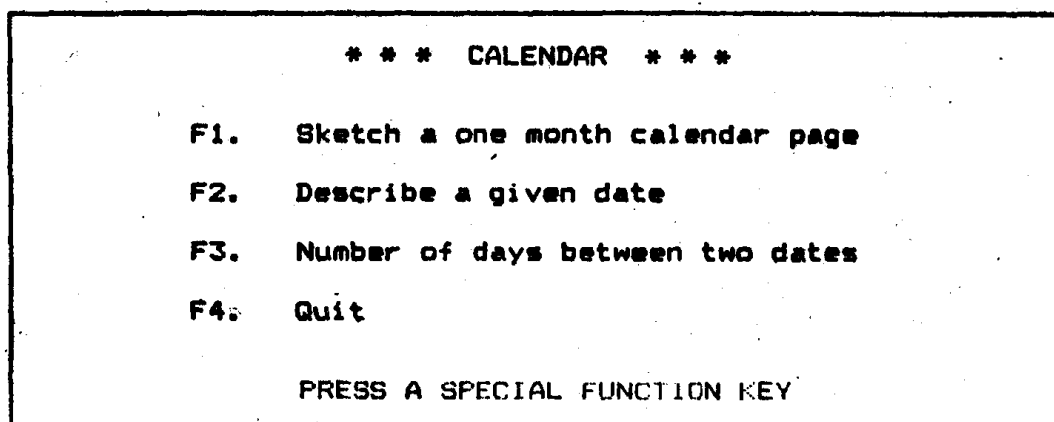


图1-1 在日历程序中可能用到的几个选用项

JULY 1776						
SUN	MON	TUE	WED	THU	FRI	SAT
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

图 1-2 日历程序显示的月的日历单

12 / 25 / 1984 can also be written as DECEMBER 25, 1984.
 The day of the week is TUESDAY.
 It is day number 360 of 1984.
 It is the 31040 day of the century.
 And the astronomical julian day number is 2446060.
 PRESS ANY KEY TO PROCEED

图 1-3 日历程序显示某些日期的结果

Between 7 / 4 / 1776 and 7 / 4 / 1984 there are 75970 days.
 PRESS ANY KEY TO PROCEED

图 1-4 日历程序显示的选用项F 3的结果

```

10 * *****
20 * **      CALENDAR      **
30 * *****
40 *
50 CLEAR
60 SCREEN 0,0,0,0
70 CLS
80 KEY OFF
90 OPTION BASE 1
100 DIM MONTH.NAME$(12),WEEK.DAYS(7)
  
```

```

110 FOR I = 1 TO 12
120 READ MONTH.NAME$(I)
130 NEXT I
140 DATA JANUARY,FEBRUARY,MARCH,APRIL,MAY,JUNE,JULY
150 DATA AUGUST,SEPTEMBER,OCTOBER,NOVEMBER,DECEMBER
160 FOR I = 1 TO 7
170 READ WEEK.DAY$(I)
180 NEXT I
190 DATA SUNDAY,MONDAY,TUESDAY,WEDNESDAY,THURSDAY,FRIDAY,SATURDAY
200 LOCATE 1,29
210 PRINT "* * * CALENDAR * * *"
220 LOCATE 7,1
230 PRINT TAB(20)"F1.    Sketch a one month calendar page
240 PRINT
250 PRINT TAB(20)"F2.    Describe a given date
260 PRINT
270 PRINT TAB(20)"F3.    Number of days between two dates
280 PRINT
290 PRINT TAB(20)"F4.    Quit
300 LOCATE 25,25
310 PRINT "PRESS A SPECIAL FUNCTION KEY";
320 ON KEY(1) GOSUB 470
330 ON KEY(2) GOSUB 1220
340 ON KEY(3) GOSUB 1650
350 ON KEY(4) GOSUB 2010
360 KEY(1) ON
370 KEY(2) ON
380 KEY(3) ON
390 KEY(4) ON
400
410 WHILE QUIT = NOT.YET
420 KEY.BUFFER.CLEAR$ = INKEY$
430 WEND
440 CLS
450 END
460 '
470 ' F1 Subroutine, sketch a month
480 SCREEN 0,0,1,1
490 CLS
500 LOCATE 7,20
510 INPUT "What month ";Q$
520 IF Q$ = "" THEN 1190
530 GOSUB 2150
540 GOSUB 2220
550 MONTH = VAL(Q$)
560 IF MONTH THEN 600
570 FOR I = 1 TO 12
580 IF LEFT$(MONTH.NAME$(I),3) = LEFT$(Q$,3) THEN MONTH = I
590 NEXT I
600 IF MONTH THEN 650
610 LOCATE 8,12
620 PRINT "I don't recognize the month you entered ... try again

```

```

630 BEEP
640 GOTO 500
650 LOCATE 8,12
660 PRINT SPACE$(53);
670 LOCATE 9,20
680 INPUT "What year ";Q$
690 IF Q$ = "" THEN 1190
700 YEAR = VAL(Q$)
710 IF YEAR THEN 760
720 LOCATE 10,12

730 PRINT "I don't recognize the year you entered ... try again
740 BEEP
750 GOTO 670
760 IF YEAR < 100 THEN YEAR = YEAR + 1900
770 IF YEAR > 1581 AND YEAR < 4000 THEN 810
780 PRINT "The year must be in the range 1582 to 3999 ... try again
790 BEEP
800 GOTO 670
810 DAY = 1
820 GOSUB 2300
830 DAYOFWEEK = WEEKDAY
840 TITLE$ = MONTH.NAME$(MONTH)
850 JFIRST = JULIAN
860 MONTH = MONTH + 1
870 IF MONTH > 12 THEN MONTH = 1
880 IF MONTH = 1 THEN YEAR = YEAR + 1
890 GOSUB 2300
900 MONTHDAYS = JULIAN - JFIRST
910 CLS
920 LOCATE 1,37 - LEN(TITLE$) / 2
930 PRINT TITLE$ ; YEAR + (MONTH = 1)
940 DATE = 1
950 ROW = 6
960 COL = DAYOFWEEK * 7 + 10
970 LOCATE ROW,COL - (DATE < 10)
980 PRINT DATE
990 DATE = DATE + 1
1000 IF DATE > MONTHDAYS THEN 1040
1010 DAYOFWEEK = DAYOFWEEK MOD 7 + 1
1020 IF DAYOFWEEK = 1 THEN ROW = ROW + 3
1030 GOTO 960
1040 FOR ROWLINE = 4 TO ROW + 3 STEP 3
1050 LOCATE ROWLINE,15
1060 PRINT STRING$(50,"_");
1070 NEXT ROWLINE
1080 FOR ROW2 = 4 TO ROW + 1
1090 FOR COL2 = 15 TO 65 STEP 7
1100 LOCATE ROW2,COL2
1110 IF ROW2 = 4 THEN PRINT " "; ELSE PRINT "!";
1120 NEXT COL2,ROW2
1130 FOR I = 1 TO 7

```

```

1140 LOCATE 3,7 * I + 10
1150 PRINT LEFT$(WEEK.DAYS(I),3);
1160 NEXT I
1170 BARMESS = 1
1180 GOSUB 2070
1190 SCREEN 0,0,0,0

1200 RETURN
1210 '
1220 ' F2 Subroutine, describe a date
1230 SCREEN 0,0,1,1
1240 CLS
1250 LOCATE 7,7
1260 LINE INPUT "Enter a date ... (any reasonable format) ";CAL$
1270 IF CAL$ = "" THEN 1620
1280 GOSUB 2600
1290 IF YEAR THEN 1340
1300 PRINT
1310 PRINT "Your date is unrecognizable, or isn't a valid date ...
      try again.
1320 BEEP
1330 GOTO 1250
1340 CLS
1350 LOCATE 5,6
1360 BS$ = CHR$(29)
1370 PRINT MONTH;"/";DAY;"/";YEAR;"can also be written as ";
1380 PRINT MONTH.NAME$(MONTH);DAY;BS$;",";YEAR;BS$;". "
1390 LOCATE 7,7
1400 PRINT "The day of the week is ";WEEK.DAYS(WEEKDAY);". "
1410 IF YEAR < 1600 THEN 1590
1420 JULIAN2 = JULIAN
1430 MONTH2 = MONTH
1440 DAY2 = DAY
1450 YEAR2 = YEAR
1460 MONTH = 12
1470 DAY = 31
1480 YEAR = YEAR - 1
1490 IF YEAR < 1582 THEN 1540
1500 GOSUB 2300
1510 YEARDAY = JULIAN2 - JULIAN
1520 LOCATE 9,7
1530 PRINT "It is day number"YEARDAY"of"YEAR2;BS$;". "
1540 YEAR = (INT(YEAR/100) - 1) * 100 + 99
1550 GOSUB 2300
1560 CENTDAY = JULIAN2 - JULIAN
1570 LOCATE 11,7
1580 PRINT "It is the"CENTDAY"day of the century.
1590 LOCATE 13,7
1600 PRINT "And the astronomical Julian day number is";JULIAN2;BS$;". "
1610 GOSUB 2070
1620 SCREEN 0,0,0,0
1630 RETURN

```

```

1640 '
1650 ' F3 Subroutine, days between dates

1660 SCREEN 0,0,1,1
1670 CLS
1680 LOCATE 7,7
1690 LINE INPUT "Enter one date ... (any reasonable format) ";CAL$
1700 IF CAL$ = "" THEN 1980
1710 GOSUB 2600
1720 IF YEAR THEN 1770
1730 LOCATE 9,1
1740 PRINT "Your date is unrecognizable, or isn't a valid date
      ... try again.
1750 BEEP
1760 GOTO 1680
1770 MONTH3 = MONTH
1780 DAY3 = DAY
1790 YEAR3 = YEAR
1800 JULIAN3 = JULIAN
1810 LOCATE 9,1
1820 PRINT SPACE$(79);
1830 LOCATE 9,7
1840 LINE INPUT "Enter second date ... ";CAL$
1850 IF CAL$ = "" THEN 1980
1860 GOSUB 2600
1870 IF YEAR THEN 1920
1880 LOCATE 11,1
1890 PRINT "Your date is unrecognizable, or isn't a valid date
      ... try again.
1900 BEEP
1910 GOTO 1830
1920 NUMDAYS = ABS(JULIAN3 - JULIAN)
1930 CLS
1940 LOCATE 7,7
1950 PRINT "Between";MONTH3;"/";DAY3;"/";YEAR3;"and";
1960 PRINT MONTH;"/";DAY;"/";YEAR;"there are";NUMDAYS;"days."
1970 GOSUB 2070
1980 SCREEN 0,0,0,0
1990 RETURN
2000 '
2010 ' F4 Subroutine, set quit flag
2020 QUIT = 1
2030 RETURN
2040 '
2050 '
2060 ' Subroutine, wait for user before proceeding
2070 LOCATE 25,28
2080 IF BARMESS = 0 THEN PRINT "PRESS ANY KEY TO PROCEED";
2090 K$ = INKEY$
2100 IF K$ = "" THEN 2090

```

```

2110 BARMESSE = 0
2120 RETURN
2130 '
2140 ' Subroutine, de-space Q$
2150 SP = INSTR(Q$, " ")
2160 IF SP = 0 THEN 2220
2170 Q$ = LEFT$(Q$, SP-1) + MID$(Q$, SP+1)
2180 GOTO 2150
2190 RETURN
2200 '
2210 ' Subroutine, just capitalize Q$
2220 FOR QP = 1 TO LEN(Q$)
2230 CHAR$ = MID$(Q$, QP, 1)
2240 IF CHAR$ < "a" OR CHAR$ > "z" THEN 2260
2250 MID$(Q$, QP, 1) = CHR$(ASC(CHAR$)-32)
2260 NEXT QP
2270 RETURN
2280 '
2290 ' Subroutine, MONTH, DAY, YEAR to JULIAN, WEEKDAY
2300 JULIAN = INT(365.2422 * YEAR + 30.44 * (MONTH-1) + DAY + 1)
2310 T1 = MONTH - 2 - 12 * (MONTH < 3)
2320 T2 = YEAR + (MONTH < 3)
2330 T3 = INT(T2 / 100)
2340 T2 = T2 - 100 * T3
2350 WEEKDAY = INT(2.61 * T1 - .2) + DAY + T2 + INT(T2 / 4)
2360 WEEKDAY = (WEEKDAY + INT(T3 / 4) - T3 - T3 + 77) MOD 7 + 1
2370 T4 = JULIAN - 7 * INT(JULIAN / 7)
2380 JULIAN = JULIAN - T4 + WEEKDAY + 7 * (T4 < WEEKDAY - 1) + 1721060
2390 RETURN
2400 '
2410 ' Subroutine, JULIAN to MONTH, DAY, YEAR, WEEKDAY
2420 T5 = JULIAN
2430 YEAR = INT((JULIAN - 1721061) / 365.25 + 1)
2440 MONTH = 1
2450 DAY = 1
2460 GOSUB 2300
2470 IF JULIAN <= T5 THEN 2500
2480 YEAR = YEAR - 1
2490 GOTO 2460
2500 MONTH = INT((T5 - JULIAN) / 29 + 1)
2510 GOSUB 2300
2520 IF JULIAN <= T5 THEN 2550
2530 MONTH = MONTH - 1
2540 GOTO 2510
2550 DAY = T5 - JULIAN + 1
2560 GOSUB 2300
2570 RETURN
2580 '
2590 ' Subroutine, convert CAL$ to MONTH, DAY, YEAR
2600 Q$ = CAL$
2610 GOSUB 2220

```

```

2620 CAL$ = 0$
2630 MONTH = 0
2640 DAY = 0
2650 YEAR = 0
2660 FOR I = 1 TO 12
2670 IF INSTR(CAL$,LEFT$(MONTH.NAME$(I),3)) THEN MONTH = I
2680 NEXT I
2690 FOR I = 1 TO LEN(CAL$)
2700 CHAR$ = MID$(CAL$,I,1)
2710 IF CHAR$ < "0" OR CHAR$ > "9" THEN MID$(CAL$,I,1) = ":"
2720 NEXT I
2730 IF INSTR(CAL$,":") THEN 2790
2740 IF LEN(CAL$) <> 6 AND LEN(CAL$) <> 8 THEN 3040
2750 MONTH = VAL(LEFT$(CAL$,2))
2760 DAY = VAL(MID$(CAL$,3,2))
2770 YEAR = VAL(MID$(CAL$,5))
2780 GOTO 2930
2790 VFLAG = 0
2800 FOR I = 1 TO LEN(CAL$)
2810 CALVAL = VAL(MID$(CAL$,I))
2820 IF CALVAL = 0 THEN VFLAG = 0
2830 IF CALVAL = 0 OR VFLAG = 1 THEN 2920
2840 IF MONTH THEN 2870
2850 MONTH = CALVAL
2860 GOTO 2910
2870 IF DAY THEN 2900
2880 DAY = CALVAL
2890 GOTO 2910
2900 YEAR = CALVAL
2910 VFLAG = 1
2920 NEXT I
2930 IF YEAR < 100 AND YEAR > 0 THEN YEAR = YEAR + 1900
2940 IF YEAR < 1582 OR YEAR > 3999 THEN YEAR = 0
2950 IF YEAR = 0 THEN 3040
2960 MONTH2 = MONTH
2970 DAY2 = DAY
2980 YEAR2 = YEAR
2990 GOSUB 2300
3000 GOSUB 2420
3010 IF MONTH2 <> MONTH THEN YEAR = 0
3020 IF DAY2 <> DAY THEN YEAR = 0
3030 IF YEAR2 <> YEAR THEN YEAR = 0
3040 RETURN

```

§ 2 时 钟

这里要介绍的是一种使IBM PC起时钟作用的好方法。最简便的设置时间的方法,是打入一个作为“变量”TIME\$的数值。例如要把时钟拨到10点07分时,可以打入并执行这样的语句: TIME\$=“10:07:00”。

利用个人计算机所提供的几条功能很强的作图命令,程序就能画出时钟的外貌,当时钟表针每走一秒,各种相应的表针都擦掉又重画一次。请注意这些表针不是用直线语句画出的,而是用一种圆语句的专用格式画出的。角度采用负值时,画圆的起始点和结束点都会有直线与圆心连接起来。把起始点和结束点选为同一点时,这个圆画出来就是从圆心辐射出来的一条直线。这个办法可以避免采用直线语句画表针时,需要用复杂计算来定标的麻烦。

另一种技巧,用于把数字画在时钟的表面上,每一个数都先在屏幕的左上角显示出来,然后用GET和PUT命令把数字取来,放到它们应该放的地方去。这个办法可以用于对图案进行标示,或者把一些文字或数字放到屏幕上的任何地方。图1-5表示屏幕上所要显示出来的时钟。在接近时钟程序末了的部分,有两个有用的子程序。其中第1个是用于把TIME\$变换为变量H,M和S。第2个子程序根据H,M和S等变量,以一定的格式来建立一个字符串,并把它赋值给TIME\$。

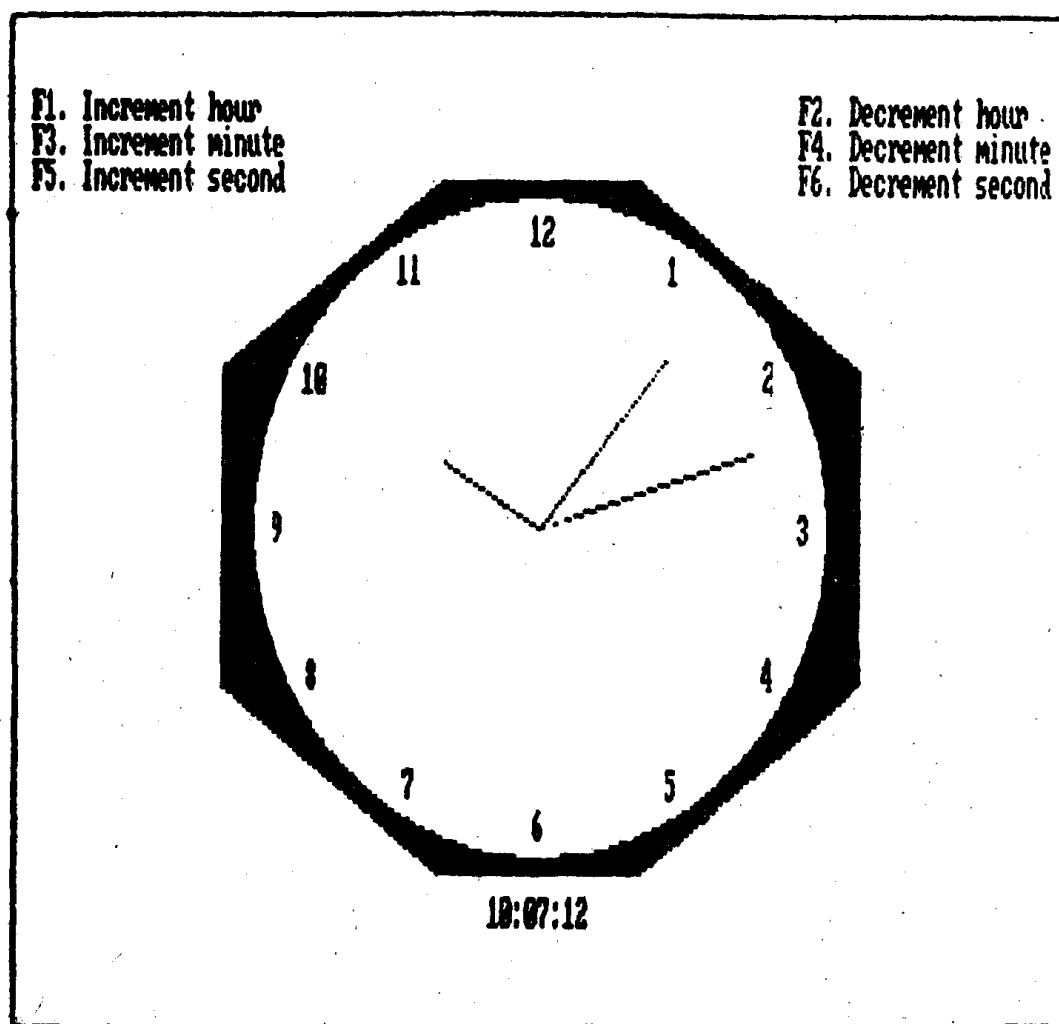


图1-5 时钟程序所画出的时钟表面

```

10 : *****
20 : **      CLOCK      **
30 : *****
40 :
50 CLEAR
60 SCREEN 2

```

```

70 CLS
80 KEY OFF
90 DIM NUM(6)
100 XCENT = 319
110 YCENT = 99
120 PI = 3.141593
130 DEF FNANG(TIME) = PI / 2 - PI * TIME / 30 - 2 * PI * (TIME > 15)
140 LINE (120,20)-(519,178),1,BF ' draw big white box
150 LINE (XCENT,0)-(0,YCENT),0 ' slice off four corners
160 LINE (XCENT,0)-(639,YCENT),0
170 LINE (XCENT,199)-(0,YCENT),0
180 LINE (XCENT,199)-(639,YCENT),0
190 PAINT (126,20),0 paint out four corners
200 PAINT (512,175),0
210 PAINT (512,20),0
220 PAINT (126,175),0
230 RADIUS = 179
240 CIRCLE (XCENT,YCENT),RADIUS,0 ' draw dark circle inside
250 PAINT (XCENT,YCENT),0 ' darken entire circle
260 FOR D = 1 TO 12 ' put numbers on clock face
270 GOSUB 800
280 XD = XCENT + 165 * COS(FNANG(D*5)) - 11 + 3 * (D > 9)
290 YD = YCENT - 68 * SIN(FNANG(D*5)) - 3
300 PUT (XD,YD),NUM
310 NEXT D
320 LOCATE 1,1
330 PRINT "F1. Increment hour";
340 LOCATE 2,1
350 PRINT "F3. Increment minute";
360 LOCATE 3,1
370 PRINT "F5. Increment second";
380 LOCATE 1,61
390 PRINT "F2. Decrement hour";
400 LOCATE 2,61
410 PRINT "F4. Decrement minute";
420 LOCATE 3,61
430 PRINT "F6. Decrement second";
440 ON KEY(1) GOSUB 860
450 ON KEY(2) GOSUB 920
460 ON KEY(3) GOSUB 980
470 ON KEY(4) GOSUB 1040
480 ON KEY(5) GOSUB 1100
490 ON KEY(6) GOSUB 1160
500 KEY(1) ON
510 KEY(2) ON
520 KEY(3) ON
530 KEY(4) ON
540 KEY(5) ON
550 KEY(6) ON
560
570 WHILE NOT YET.TO.THE.END.OF.ALL.TIME....

```

```

580         WHILE T$ = TIME$
590         WEND
600     T$ = TIME$
610     SECOND2 = SECOND
620     MINUTE2 = MINUTE
630     HOUR2 = HOUR
640     SECOND = VAL(RIGHT$(T$,2))
650     MINUTE = VAL(MID$(T$,4))
660     HOUR = (VAL(LEFT$(T$,2)) MOD 12) * 5 + MINUTE / 12
670     LOCATE 24,37
680     PRINT T$;
690     CIRCLE (XCENT,YCENT),140,1,-FNANG(SECOND),-FNANG(SECOND)
700     CIRCLE (XCENT,YCENT),140,0,-FNANG(SECOND2),-FNANG(SECOND2)
710     IF MINUTE2 = MINUTE THEN 730
720     CIRCLE (XCENT,YCENT),120,0,-FNANG(MINUTE2),-FNANG(MINUTE2)
730     CIRCLE (XCENT,YCENT),120,1,-FNANG(MINUTE),-FNANG(MINUTE)
740     IF HOUR2 = HOUR THEN 760
750     CIRCLE (XCENT,YCENT),70,0,-FNANG(HOUR2),-FNANG(HOUR2)
760     CIRCLE (XCENT,YCENT),70,1,-FNANG(HOUR),-FNANG(HOUR)
770 WEND
780 '
790 ' Subroutine, get a number for putting anywhere
800 LOCATE 1,1
810 PRINT D;
820 GET (0,0)-(22,6),NUM
830 LINE (0,0)-(22,6),0,BF
840 RETURN
850 '
860 ' Key 1 subroutine
870 GOSUB 1230
880 H = (H + 1) MOD 24
890 GOSUB 1290
900 RETURN
910 '
920 ' Key 2 subroutine
930 GOSUB 1230
940 H = (H + 23) MOD 24
950 GOSUB 1290
960 RETURN
970 '
980 ' Key 3 subroutine
990 GOSUB 1230
1000 M = (M + 1) MOD 60
1010 GOSUB 1290
1020 RETURN
1030 '
1040 ' Key 4 subroutine
1050 TEMP$ = MID$(STR$((VAL(MID$(TIME$,4))+59)MOD 60),2)
1060 M = (M + 59) MOD 60
1070 GOSUB 1290
1080 RETURN

```