# 微型计算机软件资料汇编

第三册

机械工业部计算中心
合肥工业大学微型机应用研究所 编译

GRAPHICS

muLISP

rDBMS

CP/M

机械工业部仪表局情报室
《仪表工业》编辑部

# 微型计算机软件资料汇编

## 第三册

机械工业部计算中心
合肥工业大学微型机应用研究所 编译

机械工业部仪表局情报室
《仪表工业》编辑部

# 编 译 出 版 说 明

本资料汇编收集了近期从国外引进的微型计算机软件，包括CP/M操作系统及其支持程序、高级程序设计语言、数据库管理系统和应用软件包，可以在Zilog Z80系列、Intel 8080系列微型机上使用，并已在H/Z89微型机上验证。

收集在资料汇编中的有：

微型机操作系统CP/M2.2；

小型关系数据库管理系统CONDOR SERIES/20；

高级语言 COBOL-80，PASCAL/MT+，FORTRAN-80，MBASIC，PL/I-80，C，muLISP；

编辑和字处理系统；

分类/合并程序；

库存管理程序；

图形软件包；

远程终端仿真程序等

这些资料大部分以使用手册的形式提供，可作为微型机用户手册，也可供计算机系统软件和应用软件人员以及大专院校有关专业师生学习参考。

本资料汇编由机械工业部仪器仪表工业局组织，机械工业部计算中心和合肥工业大学微型机应用研究所编译，刘运基、康兴钨审校，并请旅美学者赵鉴芳教授指导审定。在编译出版过程中，得到许多同志的大力协助，谨在此表示谢意。

由于编译者水平所限，难免有错漏之处，敬请读者指正。

本资料汇编共分六册，由机械工业部仪表局情报室《仪表工业》编辑部陆续出版。

# BASIC-80使用手册

# 目　录

## BASIC-80 使用手册

## muLISP/muSTAR-80人工智能系统参考手册

# 第一章　系统介绍和基础知识

本章包括：1.“装配指南”和BASIC-80程序设计语言的基础知识（BASIC-80是适合于8080和Z80微处理器的扩充内容最丰富的BASIC语言之一）；2. BASIC-80对硬件及系统软件的各项要求；3.面向用户的BASIC-80工作环境的说明。

## 第一节　装配指南

此节供Microsoft MBASIC-80解释程序和BASIC编译程序之用

### 1　磁盘目录

你接收到的磁盘应载有下列文件：

1) Microsoft MBASIC-80解释程序源盘：

MBASIC · COM

PI · BAS

MBASIC · COM是BASIC解释程序，它具有的命令和功能在本参考手册中有详细的介绍。PI · BAS是一个用BASIC语言写的实例程序，它用于计算π的值。PI · BAS将有助于熟悉解释程序的工作过程。

2) Microsoft MBASIC编译器源盘Ⅰ：

BASCOM · COM

BASLIB · REL

BASIC编译器存放在文件BASCOM · COM中，它具有各种命令和功能，在“BASIC编译器用户手册”中说明。BASLIB · REL是BASIC编译器系统程序库，可以通过库管理程序（即LIB · COM，在编译器源盘Ⅱ中）来修改它。

3) MBASIC编译器源盘Ⅱ：

L80 · COM

M80 · COM

CREF · COM

LIB · COM

PI · BAS

PI · REL

“Microsoft实用手册”的第二部分，定义了MACRO-80汇编器〔M80 · COM〕的使用和操作。在该实用手册的第三部分论述了交叉调用程序CREF · COM，第四部分论述了连接装配程序L80，第五部分论述了库管理程序LIB · COM。

PI · BAS是一个实例程序，用来计算π值，它帮助你学会如何编译、连接和执行一个程序。PI · REL是一个浮动目的文件，它是对PI · BAS进行编译而产生的。

基于你收到的存储载体可能是其他类型，你可以把上述文件复制到一块或多块磁盘中。

### 2. PI · BAS的输出实例

PI BAS程序的输出实例清单于下。注意，由于处理数据所使用的算法不同，因而解释程序和编译程序所产生的结果可能有所区别。

π的近似值——双精度、二项式定理展开式计算（解释程序结果）

| N | 边　数 | 边　长 | 不定近似值 | 过剩近似值 |
|---|---|---|---|---|
| 3 | 8 | 0.76536691188812 | 3.06146764755249 | 4.95931573036713 |
| 4 | 16 | 0.39018064737320 | 3.12144517898560 | 3.87800677621650 |
| 5 | 32 | 0.19603428244591 | 3.13654851913452 | 3.47739260077205 |
| 6 | 64 | 0.09813534468412 | 3.14033102989197 | 3.30237067197655 |
| 7 | 128 | 0.04908246546984 | 3.14127779006958 | 3.22030812114884 |
| 8 | 256 | 0.02454307302833 | 3.14151334762573 | 3.18054350336212 |
| 9 | 512 | 0.01227176748216 | 3.14157247543335 | 3.16096780640274 |
| 10 | 1,024 | 0.00613591633737 | 3.14158916473389 | 3.15125708966375 |
| 11 | 2,048 | 0.00306796119548 | 3.14159226417542 | 3.14641880958168 |
| 12 | 4,096 | 0.00153398059774 | 3.14159226417542 | 3.14400368450104 |
| 13 | 8,192 | 0.00076699029887 | 3.14159226417542 | 3.14279751177684 |
| 14 | 16,384 | 0.00038349514944 | 3.14159226417542 | 3.14219477240231 |
| 15 | 32,768 | 0.00019174757472 | 3.14159226417542 | 3.14189348940372 |
| 16 | 65,536 | 0.00009587385284 | 3.141594`0994263 | 3.14174501554227 |
| 17 | 131,072 | 0.00004793689368 | 3.14159226417542 | 3.14166756506744 |
| 18 | 262,144 | 0.00002396846321 | 3.14159440994263 | 3.14163205998885 |
| 19 | 542,288 | 0.00001198423161 | 3.14159440994263 | 3.14161523485294 |
| 20 | 1,048,576 | 0.00000599211580 | 3.14159440994263 | 3.14160382236958 |

π的近似值——双精度、二项式定理展开式计算（编译程序结果）

| N | 边　数 | 边　长 | 不定近似值 | 过剩近似值 |
|---|---|---|---|---|
| 3 | 8 | 0.76536686473018 | 3.06146745892072 | 4.95931523537420 |
| 4 | 16 | 0.39018064403226 | 3.12144515225805 | 3.87800673496263 |
| 5 | 32 | 0.19603428066912 | 3.13654849054594 | 3.47739256563251 |
| 6 | 64 | 0.09813534865484 | 3.14033115695475 | 3.30237081249040 |
| 7 | 128 | 0.04908245704582 | 3.14127725093277 | 3.22030755454287 |
| 8 | 256 | 0.02454307657144 | 3.14151380114430 | 3.18054396821973 |
| 9 | 512 | 0.01227176929831 | 3.14157294036709 | 3.16096827709498 |
| 10 | 1,024 | 0.00613591352593 | 3.14158772527716 | 3.15125564133382 |
| 11 | 2,048 | 0.00306796037257 | 3.14159142151120 | 3.14641796432625 |
| 12 | 4,096 | 0.00153398063749 | 3.14159234557012 | 3.14400376602075 |
| 13 | 8,192 | 0.00076699037514 | 3.14159257658487 | 3.14279782442605 |
| 14 | 16,384 | 0.00038349519462 | 3.14159263433856 | 3.14219514270746 |
| 15 | 32,768 | 0.00019174759819 | 3.14159264877699 | 3.14189387407905 |
| 16 | 65,536 | 0.00009587379921 | 3.14159265238659 | 3.14174325781772 |
| 17 | 131,072 | 0.00004793689962 | 3.14159265328899 | 3.14166795419967 |
| 18 | 262,144 | 0.00002396844981 | 3.14159265351459 | 3.14163030351872 |
| 19 | 524,288 | 0.00001198422491 | 3.14159265357099 | 3.14161147846025 |
| 20 | 1,048,576 | 0.00000599211245 | 3.14159265358509 | 3.14160206600152 |

### 3. 磁盘的使用

*磁盘的装入*

打开磁盘驱动器门，将磁盘标签面朝向打开的门，插入磁盘，然后小心地关上驱动器门。

*磁盘的保管*

由于磁盘易于损坏，所以在使用磁盘时必须采取如下的保护性措施：

1) 磁盘不使用时，应将其插入纸套中。

2) 让磁盘远离磁场、磁性纸装订机以及被磁化的剪刀、螺丝起子和大功率电子仪器。因为各种磁场能使磁盘上记录的数据丢失。

3) 更换已损坏或过分陈旧的纸套。

4) 只能在磁盘标签上写字，而且只能使用沾水笔尖，不能使用铅笔或元珠笔，因为它们可能损坏磁盘的表面。

5) 磁盘应远离热源或具有污染性的物质。

6) 磁盘不要暴露在阳光、烟雾或湿度很大的环境中。

7) 不要触摸磁盘表面。因为触摸可能会使数据遭到破坏。

*写保护*

磁盘可以加上写保护，从而使数据不能写入到磁盘中（所有交付的磁盘都具有写保护功能）。一块磁盘写保护的方式，取决于磁盘的大小。

在5.25英寸磁盘的侧边有一个写保护缺口。当缺口用标签或不透明的纸带贴上时，数据就不能再写入到磁盘中。

一块8英寸磁盘的写赋能缺口在盘的侧边。如果写赋能缺口露出，则数据就不能写入到磁盘中。为使8英寸磁盘写赋能，要用标签或不透明的纸带将缺口贴上。

### 4. 制备工作盘

按照CP/M手册中的规定，要制备一个工作盘，首先要接通计算机电源，然后从CP/M源盘I上将CP/M导入内存。

如果有两个或多个相同尺寸的驱动器，那么就可以使用DUP·COM文件来复制MBASIC源盘。如果仅有一个驱动器，则可以按如下方式：

1) 使用程序FORMAT·COM对用来拷贝的空白磁盘进行初始化。

2) 使用PIP·COM程序来复制MBASIC源盘。

注：所有源盘都加上了写保护，以保证有一个精确的软件源本。因此源盘被复制后应存放在一个安全的地方，而将拷贝盘作为日常编程之用。

## 第二节 系统介绍

### 1. 手册简介

本BASIC-80手册是BASIC-80语言的主要参考书。章节是根据功能来编排的。例如，如果需要了解有关字串的资料，那么只要查阅第五章“字串”。

本BASIC-80手册还包括了“装配指南”和一张使用卡片，该指南告诉你怎样拷贝BASIC解释程序工作盘，而使用卡片中列出了许多常用资料，故应把它放在手边以便查找。

**2. 硬件要求**

运行BASIC-80解释程序时，对硬件的要求如下：

1) 8080或Z80微处理器，

2) 具有48K的RAM，

3) 一个软盘驱动器，

4) 一个终端显示器，

5) 可选项——一台硬拷贝设备〔例如行式打印机——译者注〕。

以上是最低要求的硬件结构。我们建议配有多个磁盘驱动器，若计划编制大型程序，那么无疑将需要一台硬拷贝设备。

**3. 系统软件的要求**

BASIC-80解释程序必须在CP/M(2.0或2.0以后的版本)操作系统支持下运行。

**4. 制备磁盘**

BASIC-80解释程序分布在5.25英寸或8英寸磁盘中。"装配指南"中有关于制备工作盘的资料。

除了作拷贝用途之外，一律不要使用BASIC-80源盘，源盘应存放在安全的地方，"装配指南"中还有更多的有关磁盘保管的资料。

**5. BASIC-80的初始化**

BASIC-80解释程序以绝对二进制形式存放在磁盘上，其文件名为MBASIC·COM。BASIC-80可以直接装入内存使用。为装入BASIC-80，可在CP/M提示符后面键入：

```
MBASIC
```

上述命令将把MBASIC装入内存。MBASIC被装入内存后，将显示某些启动信息.该信息类似下列形式：

```
BASIC-80 Rev·5.2
〔CP/M Version〕
Copyright 1977,78,79,80(c) by Microsoft
Created: 11—Aug—80
15430 Bytes free
```

请注意：系统版本号码、文件创立日期、内存的自由字节数可能与此不同。

当把一个程序文件名附到BMASIC 这个命令串后面时，该程序可以自动运行。例如，如果你想将解释程序装入内存并运行SAMPLE·BAS程序，则可键入下面的命令串：

```
MBASIC␣SAMPLE
```

这里MBASIC和SAMPLE之间的空格是需要的（在本手册中符号␣指明一个空格）。在上面命令中假定文件名的补缺值是·BAS。若指定的文件名没有找到，则将显示"File not found"（文件未找到）的信息，并返回到CP/M命令状态。

## 第三节 基础知识

**1. 操作方式**

解释程序装入内存之后，BASIC-80将显示字符"OK"，这表明BASIC-80已处于命令

工作方式，或称命令状态。

在命令状态中，一旦用回车符RETURN结束指令，BASIC-80解释程序就立即执行，在命令状态中输入的命令和语句前面都不要行号。算术和逻辑运算的结果可以被立即显示并保存下来以备后用，但指令本身在执行以后就丢失了。这种状态对于调试程序和将BASIC-80作为一个计算器以完成快速计算是有用的，该计算一般不需要一个完整的程序。

如果一个程序行由一个行号开始，则BASIC-80认为你想存储该程序行以供以后执行。这称为间接方式或程序方式。存在内存中的程序，输入RUN命令后则被执行。

**2. 行格式**

在BASIC-80程序中，程序行格式如下（方括号中是可选项）：

nnnnn BASIC-80语句〔:BASIC-80语句〕

按照程序员的意图，若干条BASIC-80语句可以放在同一行中，但每条语句之间要以冒号分隔。

一条BASIC-80程序行要以行号开头、回车符结尾，每行最多包含255个字符。

利用终端键盘上的LINE FEED键，可以将一逻辑行分开到若干物理行上去。LINE FEED键允许你在下一个物理行中继续输入同一逻辑行，而不必用回车键RETURN。

**3. 行号**

每一条BASIC-80程序行的起始都有一个行号，行号决定程序行存入内存的次序，也供分支跳转和编辑查找时使用。行号必须在0～65529范围内。在EDIT、LIST、AUTO和DELETE命令中可以使用一个〔·〕代表当前行。

**4. 字符集**

BASIC-80字符集由字母、数字和特殊字符所组成。字母可以是大写字母或小写字母，数字是0～9。

BASIC-80也识别以下的特殊字符和终端键：

| 符号 | 名称 |
|---|---|
| ␣ | 空格 |
| ; | 分号 |
| = | 等号或赋值号 |
| + | 加号 |
| − | 减号 |
| * | 星号或乘号 |
| / | 斜杠或除号 |
| ↑ | 上箭头号或乘幂号 |
| ( | 左圆括号 |
| ) | 右圆括号 |
| % | 百分号 |
| # | 数值符（或镑） |
| $ | 美元符号 |
| ! | 惊叹号 |
| 〔 | 左方括号 |

| | |
|---|---|
| ) | 右方括号 |
| , | 逗号 |
| . | 句号或十进制小数点 |
| ' | 单引号(撇号) |
| " | 双引号 |
| : | 冒号 |
| & | 和 |
| ? | 问号 |
| < | 小于 |
| > | 大于 |
| \ | 右斜杠或整除符号 |
| @ | At符号 |
| _ | 下划号 |
| DELETE | 删除最后输入的一个字符 |
| ESC | 从编辑状态子命令 I 中退出 |
| TAB | 光标跳到下一表格位置，光标每八格一跳 |
| LINE FEED | 光标移动到下一个物理行 |
| RETURN | 结束一行 |
| CTRL | 控制键 |

**5. 控制字**

BASIC-80 中有下列控制字：

| | |
|---|---|
| CTRL—A | 在一行输入时进入编辑状态 |
| CTRL—C | 中断程序执行并返回到BASIC-80状态 |
| CTRL—G | 使终端响铃 |
| CTRL—H | 退格，删除最后键入的字符 |
| CTRL—I | 表格符(TAB)，光标每八位一跳 |
| CTRL—O | 程序运行当中暂停输出，下一个 CTRL-O则重新恢复输出。 |
| CTRL—R | 重新显示当前语句行 |
| CTRL—S | 暂停程序执行 |
| CTRL—Q | 在 CTRL—S 之后，用以恢复程序运行 |
| CTRL—U | 删除当前语句行 |

若要输入这些控制字，则需要在按住CTRL 键的同时按入字母键。例如：要输入CTRL—G，则按住CTRL键不放，再按字母键G。

## 第四节　BASIC-80编程

本节使你了解如何编制BASIC-80程序，并说明 BASIC-80 程序设计中的一些特点。

**1. 装入BASIC-80解释程序**

在使用BASIC-80解释程序之前，必须把它装入计算机内存，它是一个绝对二 进 制 文

件，即该程序可以直接被计算机执行。在按下列步骤操作之前，必须先导入操作系统。如果不了解如何操作，则应查阅有关操作系统手册。

调用解释程序的CP/M文件名是MBASIC·COM，因而要把BASIC-80解释程序装入内存，可在CP/M系统提示符后面键入：

A〉MBASIC

（不要键入A〉，因为它代表 CP/M 系统提示符。并记住按 RETURN 键来结束这一行。）

这里假定MBASIC·COM文件驻留在当前补缺盘中，若该文件没有存放在当前补缺盘中，那么就要先键入驱动器名然后再键入文件名。例如：若A：是当前补缺盘，而 BASIC-80文件驻留在驱动器B：中，则要用下面的命令来装入BASIC-80解释程序：

A〉B:MBASIC

BASIC-80装入内存后，荧光屏上将显示有关启动的信息，可使用的自由存储单元总数以及BASIC-80的文本编号也将被显示。记下可使用的存储单元总数，这对于编制一个复杂的大型程序来说，无疑是一个关键性的信息。

当BASIC-80以上述方式装入内存时，它将对工作环境作下述约定：

不能同时打开三个以上的磁盘文件，

可使用全部有效内存，

随机记录大小为128字节。

当然，你可以通过使用某些开关去改变这些约定。

可被同时打开的磁盘文件的数目范围为0～15，开关/F：用来指定同时打开的文件的最多个数。BASIC-80将在内存中为每一个由开关/F：规定的文件建立一个文件缓冲区，这将减少可使用的存储单元的数量。例如要建立 5 个文件缓冲区，可采用如下命令：

A〉MBASIC␣/F:5

注意！在MBASIC和/F:5间有一个空格，如果未键入这个空格，CP/M系统将认为这个开关是文件名的一部分。

还可以通过开关 /M：来规定 BASIC-80 使用的内存的最高极限位置。在某些场合中，为了预留汇编语言子程序的空间，设置的最高存储位置要处于CP/M BDOS 的位置之下，但希望不要低得太多。在所有场合中最高存储位置都应在BDOS的起始位置之下（BDOS的起始地址在存储单元 6 和 7 之中）。如果/M：开关被省略，则可使用的存储空间最高可达到BDOS的起始位置。

注：文件的数目和最高存储位置，可用十进制、八进制（加前缀＆O）或十六进制（加前缀＆H）数来表示。

通过使用开关/S：，还可以改变随机文件记录的大小。记录大小的补缺值是128字节，一个记录最多为256字节。例如要指定记录大小为200字节，则可以使用以下的命令：

A〉MBASIC␣/S:200

以上三个开关的任何组合都可以用在一条命令行中，例如命令：

A〉MBASIC␣PAYROLL·BAS

表明使用全部存储空间，可同时打开的文件数目为 3 。又如，装入和运行

INVENT·BAC：

A〉MBASIC↵/M:32768 INVENT·BAC

表明可使用低端32K的存储单元，可同时打开3个文件。

BASIC-80解释程序装入内存后，就可以开始编制程序。

**2. 编制BASIC-80程序**

一个BASIC-80程序由若干行语句所组成。语句即是对BASIC-80的指令。每一程序行用一个行号开始，后面跟随一条或多条BASIC-80程序语句。行号指出语句执行的顺序，该顺序也可通过某些语句来改变。

BASIC-80程序行的格式是：

| 行号 | 语句关键字 | 语句正文 | 行终止符 |
|---|---|---|---|
| 100 | LET | x=x+1 | 〈RETURN〉 |

BASIC-80程序中的每一程序行，必须由行号开始，行号必须在0～65529正整数范围内。BASIC-80行号是一种标志，用来区别程序中各行，因此，每一个行号在程序中必须是唯一的。

在BASIC-80程序中，每一程序行要用一个回车符来结束，即键入键盘上的RETURN键。

可以使用象1、2、3、4这样的连续行号。如：

```
1↵x=1
2↵y=2
3↵z=x+y
4↵END
```

然而，实际应用的是以10为增量的递增行号，这便于在已编好的语句行间插入新的语句。

```
10↵x=1
20↵y=2
30↵z=x+y
40↵END
```

另外一个非常有用的东西是让BASIC-80自动产生行号，这可用AUTO命令来实现。例如，若键入命令AUTO100,10，则BASIC-80会由行号100开始自动编号，每行递增10。你所要做的只是在产生的行号后面键入BASIC-80程序行。

**3. BASIC-80程序的运行**

BASIC-80程序编好后，通常需要去运行这个程序。运行是通过RUN命令来实现的。下面的命令使BASIC-80立即执行内存中的现行程序：

```
RUN
```

这个命令使程序从最低的行号开始连续执行（除非执行的顺序被象GOTO这样的语句所改变）。在RUN命令中也可以规定程序执行开始行号。例如下面的命令使程序从行号100开始执行：

```
RUN↵100
```

RUN命令也可以用于执行一个存在于磁盘上的BASIC-80程序。比如假设ALBUM·BAS文件存储在现行补缺盘上，下面的语句就可使ALBUM·BAS程序投入运行：

RUN"ALBUM"

注意：在上述文件名字串中，没有指定驱动器和文件的扩展名。在这种情况下，则假定是现行补缺驱动器，扩展名是·BAS。

另外，要保证在文件名字串中只使用大写字母。BASIC-80依赖CP/M进行文件处理，而CP/M系统中大多数公用程序不能识别用小写字母写的文件名。因此，以小写字母作为文件名来存储文件显然是不恰当的。因为CP/M不能识别小写字母的文件名，因而也就不能删除或更改这类文件名。虽然小写字母为名称的文件能在BASIC-80中被删除，但在BASIC-80程序语句中使用文件名时，也一定要用大写字母。

这并不是说，使用小写字母作为文件名有原则上的错误，而是使用小写字母作为文件名可能会产生意外的结果。也许有人希望用小写字母作为文件名，以使文件更难被删除或被改名，这样就为一些重要的程序提供了一个额外级别的保护。

**4、 BASIC-80程序的调试**

有时，BASIC-80程序并不象你预期的那样被执行，这通常是由于存在语法错误或逻辑错误。语法错误是易于觉察的，因为BASIC-80解释程序不仅能指出语法错误，而且还指出错误程序行，并自动进入编辑状态；但是，逻辑错误是难于发现的，然而系统提供了若干条语句使得逻辑查错变得较为容易些。

当BASIC-80发现语法错误时，它将在产生的错误行上自动进行编辑状态。此时，可以按L键以显示这一行（L是一个BASIC-80编辑子命令，有关编辑程序的详细说明请参看第九章"编辑"）。

语法错误的产生往往是由于拼错了关键字或使用了不正确的语句结构。请记住BASIC-80解释程序要求所有关键字都用空格分隔这一点。改正语法错误的简便方法主要是查阅参考手册。

在发现语法错误的时候，可以查阅参考手册中的有关章节，这可以利用索引来查找。在发现并纠正了错误之后，应该记住这种错误是如何产生的，并避免重犯类似错误。

BASIC-80解释程序的内部特性使得调试一个BASIC-80程序非常方便。它提供若干条语句帮助对BASIC-80程序进行查错，但你首先要弄清错误的性质。

一个程序中的错误，可能引起一个错误输出的发生，或使程序执行转移到错误语句，计算的结果可能是错误的或无法理解的。程序中的错误还会显示出状态错误标志，所以在要了解程序为什么这样做之前，你必须知道程序目前正在做什么。

可以相信，在绝大多数情况下(99.99%)是程序本身有错误，而BASIC-80解释程序出错是极不可能的。BASIC-80解释程序是适合于8080/z80微处理器的最广泛最透彻的BASIC语言之一，它本身是非常可靠的。所以如果产生了问题，总可以认为是由用户程序中的错误所引起的。

一旦确定了程序正在做什么，就可以着手去弄清为什么程序不能正确地执行。例如假设一个程序分支转移到一个行号，而这个行号却是你所不希望的。系统提供了跟踪标志以跟踪程序的流程，TRON语句用来设置跟踪标志，而TROFF语句用来撤消跟踪标志。

在跟踪程序运行时，行号被依次显示出来，它们被置于方括号([ ])内。最好是打印一张程序清单，以便对照这张清单来进行跟踪。

另外一项重要技术是在程序中设置断点。你可以用STOP语句去暂时停止程序的执行，

然后输入命令去打印每个变量的值，也可以给变量赋予新值，最后你可以用 CONT 命令或GOTO命令让程序继续运行。

虽然可以打印和改变变量的值，但在使用了STOP语句中止执行后，不能改变原来的BASIC-80程序。如果在暂停时修改程序，那么所有已经存入的变量值都将丢失，所有打开的文件都将被关闭。

**5. BASIC-80程序的保存**

当一个BASIC-80程序的编制或调试工作告一段落时，你一定希望将当前程序拷贝到磁盘上，这可以用SAVE命令来实现。SAVE命令的一般格式是：

SAVE"〈文件名〉"

〈文件名〉必须是一个合法的CP/M文件名。如果没有给出驱动器号，则约定为当前补缺驱动器；如果没有给出文件扩展名，则补缺值为·BAS。例如，要想保存程序 GAME·BAS，则可以使用下面语句：

SAVE"C:GAME·BAS"

注意，该文件将保存在驱动器C:上，文件扩展名·BAS可以省略，而由系统提供补缺值。文件名一定要用大写字母。BASIC-80解释程序一般以紧缩二进制形式来保存文件，然而程序也可以ASCII码形式来保存。例如：

SAVE"C:GAME", A

这个命令将文件以ASCII码形式存储在驱动器C:中，其文件名为GAME·BAS。也可以以带保护的形式来存储程序，这样使得程序不能被列出或编辑，这只要在文件名后面附加一个字母P。例如：

SAVE"C:GAME", P

这个文件将以二进制编码形式存储。当带保护的文件以后被运行或装入时，则将不能列出这个程序或对它进行编辑。

**6. BASIC-80程序的装入**

若想在某一个BASIC-80程序中开始下一阶段的工作，必须将它从磁盘调入内存。这可使用LOAD命令来实现。LOAD命令的一般格式为：

LOAD"〈文件名〉"

例如，要想装入PAYROL·BAS程序，则可使用下面的命令：

LOAD"PAYROL"

注意该文件的扩展名被省略了，BASIC-80约定文件扩展名是·BAS。另外，驱动器号也被省略了。在这种情况下，约定为当前补缺驱动器。

文件名规定采用大写字母，这个规定适用于所有包含文件名的字串常量或变量。

同样也可采用LOAD命令来执行程序，这只要附加一个字母R在文件名的后面。例如：

LOAD"PAYROL", R

这种LOAD命令的格式将把程序装入内存并立即执行它，好象输入了RUN命令一样。所有已经打开的文件保持打开，以供刚装入的文件使用。

**7. 在硬拷贝设备中开列BASIC-80程序**

在程序设计中，有时会需要BASIC-80程序的硬拷贝清单。把 BASIC-80 程序送到硬拷贝设备和把程序送到控制台非常相似，这可使用LLIST命令。