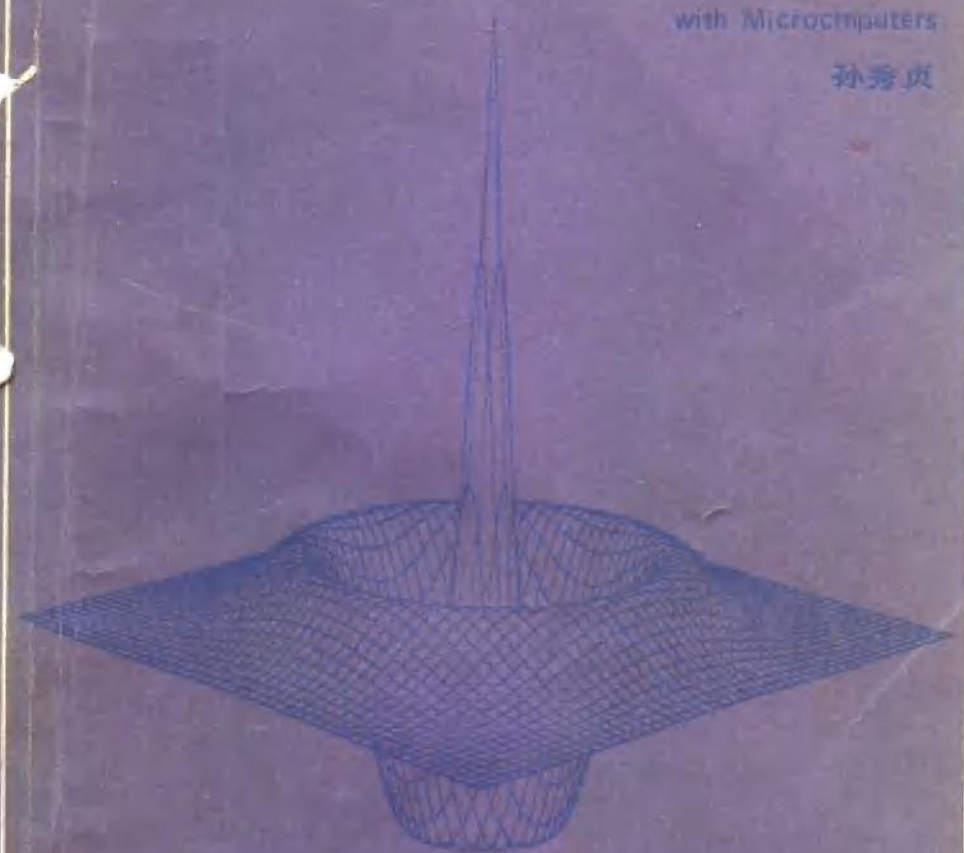


physics Teaching
with Microcomputers

孙秀贞



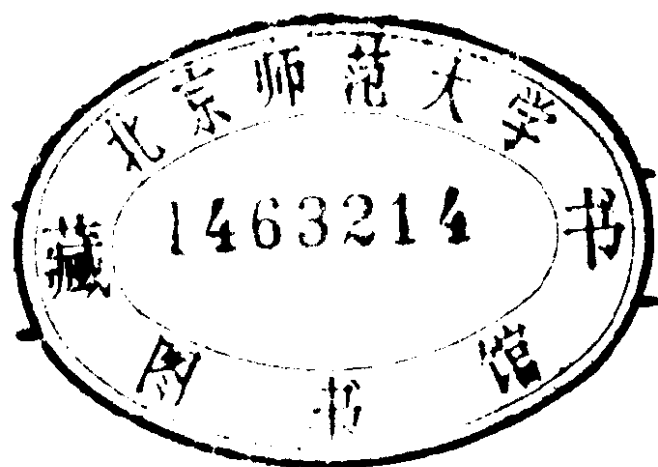
微机在物理教学中的应用

高等教育出版社

微机在物理教学中的应用

孙 秀 贞

J411.4/05



高等教育出版社

内 容 提 要

本书是学习APPLE II BASIC语言并在此基础上学习用APPLE-SOFT的高分辨率编制物理图形程序的教学参考书。

全书分为两篇，第一篇为BASIC语言，共有五章：一、BASIC语言的基础；二、BASIC语句；三、字符串变量；四、BASIC程序的调试与运行；五、彩色作图。第二篇为物理图形程序，包括四十多个完整的物理演示教学程序，遍及物理学中力学、热学与分子物理学、电磁学、光学及近代物理各分支，具备一定的系统性，可以直接应用于大学物理的演示教学。第一篇之后还有习题和习题解答。

本书可供高等院校的物理教师和学习大学物理课程的学生以及中等学校的物理教师使用。

责任编辑：陈海平

微机在物理教学中的应用

孙 秀 贞

*

高等教育出版社出版

新华书店北京发行所发行

化学工业出版社印刷厂印装

*

开本850×1168 1/32 印张11.625 字数280 000

1988年3月第1版 1988年3月第1次印刷

印数00 01 — 3 700

ISBN 7-04-000993-5/0.557

定价 2.60 元

序

本书是学习BASIC语言并在此基础上学习用APPLESOFT BASIC的高分辨率编制物理图形程序的入门书。书中给出的大量的、较全面的物理图形程序和图形，可直接应用于大学物理的演示教学，也可供广大读者深入地研究物理教学问题。

全书分为两篇，第一篇为BASIC语言，共分为五章：一、BASIC语言的基础；二、BASIC语句；三、字符串变量；四、BASIC程序的调试与运行；五、彩色作图。第二篇中给出了大学物理中力学、热学与分子物理学、电学和光学以及近代物理学的物理图形程序，所显示的图形既有静止的画面，又有动态的图象，其中某些图形在一般物理教科书中是看不到的。此外，书中有较多的人机对话程序，可供使用者自由的输入各种不同的物理参数，得到用其他手段难以得到的结果，并从对这些结果的比较、分析中，直观形象地、深入地领悟和认识到物理规律。书中包含大量的例题与习题，除少量外大多是作者自己设计的。

本书选用APPLE II机型，APPLESOFT BASIC版本，作者在叙述上力求做到通俗易懂，由浅入深，不要求读者具备高深的数学基础，因而本书对于微型计算机的初学者和具有一定基础的读者都能有所帮助。虽然书中主要讲述微机在大学物理中的应用，但具有中学物理基础的读者，也能从中受益。

在编著过程中，作者曾参考了有关书籍的部分内容；吴庆祥和贾谊明同志曾帮助注释了部分汉字并修改了某些程序，在此一并致以深切的谢意。

由于编写的时间仓促，加上作者水平所限，难免有错误与不当之处，敬请读者批评指正。

作 者

一九八五年三月于福州

目 录

第一篇 BASIC 语言

第一章 BASIC 语言的基础	1
1.1 计算机语言系统.....	1
1.2 BASIC程序的基本结构.....	5
1.3 BASIC语言的基本符号.....	6
1.4 数和变量.....	7
1.5 标准函数和自定义函数.....	11
1.6 表达式.....	17
第二章 BASIC 语句	21
2.1 打印输出 (PRINT) 语句.....	22
2.2 终止 (END) 语句.....	33
2.3 赋值 (LET)语句.....	33
2.4 输入 (INPUT) 语句.....	35
2.5 无条件转向 (GOTO) 语句.....	39
2.6 读数 (READ)语句和数据 (DATA)语句.....	43
2.7 恢复数据 (RESTORE)语句.....	46
2.8 条件转移 (IF...THEN)语句.....	47
2.9 开关 (ON...GOTO)语句.....	62
2.10 循环 (FOR...NEXT)语句.....	64
2.11 数组说明 (DIM) 语句.....	72
2.12 注释 (REM) 语句.....	76
2.13 暂停 (STOP)语句.....	77
2.14 转子(GOSUB)语句和子程序返回 (RETURN) 语句.....	78
2.15 自定义函数 (DEF FN) 语句.....	82

第三章 字符串变量	85
3.1 字符串变量的概念.....	85
3.2 字符串变量的输入 (INPUT) 语句.....	86
3.3 字符串变量的输出 (PRINT) 语句.....	87
3.4 字符串变量的赋值 (LET) 语句.....	90
3.5 字符串变量的读数 (READ) 语句和数据 (DATA) 语句.....	91
3.6 字符串的比较.....	92
3.7 子字符串.....	93
3.8 字符串函数.....	94
第四章 BASIC 程序的调试与运行	103
4.1 系统的连接与主机的启动.....	103
4.2 键盘.....	105
4.3 输入程序.....	108
4.4 编辑程序.....	109
4.5 运行程序.....	111
第五章 彩色作图	113
5.1 低分辨率作图.....	113
5.2 高分辨率作图.....	117
习题	139
习题解答	150

第二篇 物理图象及其程序

一、力学部分	160
1. 空气阻力对落体速度与加速度的影响.....	160
2. 无阻尼的抛射体运动.....	170
3. 有阻尼的抛射体运动.....	174
4. 人造卫星按第一、二、三种宇宙速度的运动.....	178
5. 两小球在完全弹性正碰撞前后的动能传递与 两小球质量比的关系.....	183

6. 谐振子势能曲线.....	187
7. 简谐振动、阻尼振动和受迫振动.....	191
8. 共振曲线.....	198
9. 同方向振动的合成.....	202
10. 拍.....	206
11. 李萨如图形.....	210
12. 驻波.....	216
13. 安全抛物线与抛物线族顶点的轨迹.....	221
14. 波的叠加.....	226
二、分子物理学与热力学部分	231
1. 麦克斯韦气体分子速率分布.....	231
2. 在重力场中粒子数按高度的分布.....	235
3. 分子之间的相互作用力和相互作用势能.....	239
4. 理想气体的等温线与绝热线.....	243
5. 真实气体 (CO_2) 的范德瓦耳斯等温线.....	246
三、电磁学部分	250
1. 均匀带电球体和均匀带电球壳 (有一定的厚度) 的场强、 电位移和电势随距离的变化.....	250
2. 亥姆霍兹线圈轴线上的磁感应强度的分布.....	257
3. 电源输出功率、内阻消耗功率随外电阻的变化关系.....	261
4. 纯电阻电路中的电压、电流、功率.....	265
5. 纯电感电路中的电压、电流、功率.....	269
6. 纯电容电路中的电压、电流、功率.....	273
7. R 、 L 、 C 串联电路中的电流、总电压和各分电压.....	277
8. 两点电荷轴线上的场强分布.....	281
9. 电四极子轴线上的场强分布.....	286
10. 真空中两相互平行的无限长直载流导线 的磁感应强度在轴线上的分布.....	291
四、光学部分	297
1. 双缝干涉的光强分布.....	297

2. 单缝衍射的光强分布.....	300
3. 多缝干涉的光强分布.....	304
4. 光栅衍射的光强分布.....	308
5. 光的反射和折射.....	313
6. 光的内全反射.....	317
7. 单层薄膜反射率随膜层光学厚度的变化规律.....	323
8. 方解石的双折射现象.....	328
五、近代物理部分	330
1. 相对论质速关系.....	330
2. 放射性强度随时间的变化规律.....	333
3. 黑体辐射的维恩公式与普朗克公式.....	337
4. 氢原子核外电子的角分布.....	342
5. 氢原子核外电子的径向分布.....	347
附录1 APPLES OF T 中的保留字	353
附录2 DOS 出错信息	356
附录3 APPLES OF T 出错信息	357
附录4 POKE 和 PEEK 命令	360

第一篇 BASIC语言

第一章 BASIC语言的基础

1.1 计算机语言系统

1. BASIC是英文BEGINNER'S ALL-PURPOSE SYMBOLIC INSTRUCTION CODE (初学者通用符号指令代码) 的缩写。BASIC语言是目前国内外被广泛使用的计算机程序设计语言之一。

为了使用计算机进行科学计算、数据处理或过程控制, 必须事先编好程序。所谓程序, 就是根据需解决的问题事先设计好的计算步骤。而表达程序的语言, 必须是计算机能识别的基本语言, 即计算机“语言”。计算机的中央微处理机只会识别二进制数字表示的二进制代码, 这种计算机能接受的代码, 称为机器指令, 而机器指令的集合称为机器语言 (或称指令系统)。机器语言没有明显的特征, 不便于记忆和学习。使用机器语言编写程序是极其繁琐的, 既费力、费时、容易出错, 错了又不容易检查出来, 而且通用性差。因此, 人们就改用助记符代替操作码, 用符号代替地址, 这种语言称为汇编语言。用汇编语言编写的程序比用机器语言执行起来快的多, 但是, 汇编语言仍然与自然语言和数学语言差别很大, 程序设计自动化的程度很低, 通用性也差。为了从根本上解决这些问题, 人们在实践中创造出与计算机指令系统无

关、便于人们掌握、书写的语言,这就是高级语言,如FORTRAN, ALGOL, COBOL, BASIC, PL / 1, PASCAL等语言. 高级语言是一种与数学语言非常接近的语言, 而且它又和计算机的内部结构无关, 人们用高级语言编写的程序称为源程序. 由于高级语言已经是较高级的语言, 而且与数学公式非常接近, 因此, 人们易于记忆与掌握它, 通用性也较强. 但是, 机器是不能直接执行高级语言源程序的, 必须先把它转换成机器指令表示的程序后, 计算机才能执行它. 其一种做法是, 在计算机内事先存放好一套机器语言程序, 当源程序输入计算机后, 它就会充当“翻译人员”完成翻译工作. 同时, 它在边翻译边执行的过程中, 还具备语法检查, 分配存贮单元等功能, 我们称这套机器语言程序为解释程序, 它连同一些辅助性程序合称为解释系统, 本书介绍的APPLESOFT BASIC就采用这种系统. 还有另一种做法, 是用机器语言程序把输入的源程序整个地翻译成用机器指令, 这种手编程序称为编译程序 (例如对FORTRAN等).

对于不同型号的计算机以及不同的编译系统或解释系统来说, 上机操作的方法, 计算结果的输出格式以及对出错信息的规定等等, 一般来说是不同的. 用户可以根据所使用的机器从计算机手册中找到所需要的资料. 本书介绍的是用APPLE II机的单用户BASIC语言, 并着重于BASIC语言在物理学中的应用.

我们所采用的APPLE II机具有两种BASIC语言的能力, 即: 整数BASIC语言 (其系统标示符为“<”)以及APPLE-SOFT语言 (其系统标示符为“□”). 本书主要采用APPLE-SOFT BASIC (扩展浮点式) 版本, 后面常把它简写为BASIC. 它的专用命令, 请查阅附录一.

2. BASIC语言具有以下特点:

(1) 它简单易懂, 其语句、命令与英文中使用的词、以及与

“数学语言”、数学运算符很相近，例如：

END与PRINT在英语及BASIC语言中都是“结束”、“打印”的意思。又如：

在数学语言中有

$\sin x$

$\cos x$

在BASIC语言中有

SIN(X)

COS(X)

可见它很易于记忆，易于掌握。

(2) 它是一种会话式的语言，人们能通过键盘和终端（显示器等）进行“人机对话”。当源程序送入计算机并运行后，计算机能立即检查语句中的句法，若有错误，显示器立即告诉给使用者，提示他马上通过键盘修改程序。例如：

LIST

```
10 LET X = 0
20 LET Y = 5
30 LET Z = Y / X
40 PRINT Z
```

RUN

?DIVISION BY ZERO ERROR IN 30

意思是在第30语句出现分母为零的错误，使用者可以根据这个信息的提示去修改程序。

(3) 它允许在键盘上直接进行计算，而不必编写程序，也可以不写标号（一个没有标号的命令或语句，称为立即执行的命令或语句；反之，则称为延时执行的命令或语句）。例如打入

PRINT (EXP(5)+4)/9 ✓

后，计算机立即能打出结果 16.9347955 来。打入 PRINT SIN(3.14159)+COS(1.5708)✓后，便显示出零来。

(4) 它有字符串处理功能, 因而它不仅能用于科技方面的计算, 而且也便于在小型数据处理及事务管理方面.

(5) 它具有作彩色图象能力的特性, 很适合显示物理学与数学等的图象及设计美术图案等. 此外, 它还可控制喇叭音响, 真是有声有色.

3. 为了说明APPLESOFT程序的结构与规则, 让我们先看一个例子.

例 1-1 一人以初速度 v_0 竖直向上抛出一个球, 若忽略空气的阻力, 则在 t 秒末小球上升的高度

$$h = v_0 t - \frac{1}{2} g t^2 \quad (\text{本书中物理量均采用国际单位制})$$

并在程序中改用大写字母, 以下均同)

LIST

```
10 LET V0 = 12
20 LET G = 9.8
30 LET T = 1
40 LET H = V0 * T - 0.5 * G * T ^
  2
50 PRINT H
60 END
```

上面的程序又可写成:

```
10 V0 = 12 : G = 9.8 : T = 1
20 H = V0 * T - 0.5 * G * T * T
30 ?H
```

上述的第一个程序是用基本的BASIC程序编写的, 下面一个程序是用APPLESOFT编写的, 可以看出APPLESOFT程序还有自己的特殊结构与规则:

(1) 一个APPLESOFT语句行可由多个语句构成, 各语

句间可用冒号分开，构成复合语句，如上述第10语句。

(2) 赋值语句定义符 `LET` 及结束语句定义符 `END` 可以略去不写。

(3) 打印语句定义符 `PRINT` 可用问号 “?” 代替。

(4) `APPLES OF T` 语句若在一行内未写完，计算机会自动写到下一行去，但每一语句最多只允许输入 255 个字符，从第 248 个字符开始有“哔哔”声向使用者发出警告，若超过 255 个字符，就会显示出 “\” 符号。

`BASIC` 语句比较节省内存，因此，它就会占用较多的机器时间，使计算机解题速度大大降低。这种语言用在计算速度要求不太高，题目规模不很大的情况下是比较合适的。

1.2 BASIC 程序的基本结构

1. 一个 `BASIC` 程序是由若干语句集合而构成的，每个语句都让计算机执行某一方面的功能。象例 1-1 中的第 20 语句是赋值语句，将 9.8 赋值给 `G`，50 语句是要求打印出 `H` 来。

2. 一个语句一般可分为三部分

(1) 语句标号或行号：就是每个语句前头的那个数值(例 1-1 中左边从 10 到 60 均为标号)，必须是 0 到 63999 之间的整数，标号数值的间隔可以任意选择，为便于修改和插入新语句，可以 10 为递增量。标号可以不按大小顺序写，但将程序送入计算机后，`BASIC` 解释系统会自动地把源程序中所有的语句按标号从小到大的顺序整理排列好，这种编排整理的能力是计算机编辑功能的一部分。

(2) 语句定义符，也称为 `BASIC` 的关键字 (或保留字)：它规定计算机执行某一特定的功能，如例 1-1 中的 `LET`、`PRINT` 等就是语句定义符。

(3) 语句体：即每个语句定义符后面的部分，即需要执行的具体内容。例 1-1 中第 20 语句 $G = 9.8$ ，就是语句体。

3. 当一个程序送入计算机后，就存入机内，计算机并不立即执行它，使用者必须单独发出运行(RUN)的命令后，计算机才执行该程序，执行的顺序是根据语句标号从小到大一句一句地执行，运行完毕后，程序并不从机内消失，若再打入RUN，机器又重复执行程序。

1.3 BASIC语言的基本符号

各种语言都有自己的基本符号，例如，英语的基本符号是 26 个字母及一些标点符号；日语的基本符号是 50 音图及一些标点符号等。BASIC 程序也是由一些基本符号严格地按照 BASIC 语言的语法规则编排构成的，它的基本符号包括 26 个英文字母、10 个数字 (0 ~ 9) 和一些专用符号及名词，请看表 1-1。

在 BASIC 程序中绝对不允许出现非基本符号，也不允许写得含糊不清，如把数字 0 写成字母 O，否则计算机就不会计算。例如：

计算机不会计算	计算机会计计算
$\sin x$	SIN(X)
$y = v_0 t - \frac{1}{2} g t^2$	$Y = V0 * T - 0.5 * G * T * T$
$u = u_0 \cdot (1 - e^{-t/RC})$	$U = U0 * (1 - EXP(-$ $T / (R * C)))$
	或 $U = U0 * (1 - 1 / EXP$ $(T / (R * C)))$
$\frac{a \cdot b}{c \cdot d}$	$A * B / (C * D)$ (不能写成 $A * B / C * D$)

表 1-1 BASIC 基本符号

字 母	A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
数 字	0 1 2 3 4 5 6 7 8 9
标点符号	.(小数点) ,(逗号) ;(分号) :(冒号) ()(圆括号) "(双引号) ?(问号) ↵(回车换行符) □(空格)
算术运算符	^(乘幂) -(负号) *(乘号) /(除号) +(加号) -(减号) =(把数值赋给变量)
比较符号 (关系运算符)	=(等于) <>或><(不等于) <>(小于)(大于) <=或=<(小于等于) >=或=>(大于等于)
逻辑运算符 (布尔运算符)	NOT(非) AND(与) OR(或)
名 词	LET(让) PRINT(打印) INPUT(输入) READ(读) DATA(数据) STEP(步长) REM(注释) STOP(暂停) END(终止) GOTO(转向) IF(如果) THEN(则) FOR(对于) TO(到) NEXT(下一个) DIM(维数) RESTORE(恢复) GOSUB(转子程序) RETURN(返回) DEF(定义) TAB(制表)

1.4 数和变量

1. 数

- (1) 在BASIC语言中出现的数一般采用10进制数.
- (2) 数包含实数和整数.

整数——不包含小数点及小数部分，它可以是正数也可以是负数.

最小的整数: -32767;

最大的整数: +32767.

实数——包含整数、带小数的数及仅有小数部分的数，它们可以是正数也可以是负数。

实数的取值范围： $-1 \times 10^{38} \sim 1 \times 10^{38}$ 间，在加减运算中，实数可取至 1.7×10^{38} 。绝对值小于 2.9×10^{-39} 的数，APPLE-SOFT把它视为零；大于 10^{38} 的数，APPLESOFT不接受它，并显示溢出信息（OVERFLOW ERROR）。

若未注明正、负号时，被视为正数。

(3) 数的表示形式有定点表示法和浮点表示法两种。

定点表示法是10进制数，指一个数的表示允许使用0~9、小数点、正负号（+号一般可以省略）。例如：

10, 345, 0.67, 3.57, -0.0068, -26

浮点表示法是在定点表示法的基础上增加指数部分 $E \pm ee$ ， ee 表示10的次方， E 表示底数10，若方次为10以内的数，则表示第2个 e 值，第1个 e 为0，但指数不允许带小数， E 前必须有数字。例如：

浮点表示法	定点表示法
$-0.17502E + 3$	-175.02
$3.45E + 2$ (或 $3.45E + 02$)	345
$1E + 1$ (不能写成 $E + 1$)	10
$25E - 5$	0.00025
$2E + 10$	20000000000

(4) 有效数字

有效数字是9位。

① 实数：对一个大于1或小于负1的数而言，只取最左边的9位数字(不是零的数)；若超过9位，则将以后的数四舍五入，此时要使用浮点表示法表示出来。例如：

□ ? 1234567891

$1.23456789E + 9$

□ ? -150000475.75
 -150000476
 □ ? -123456789123456789
 -1.23456789E + 17
 □ ? 90000000.7558
 90000000.8

② 小数：在 +1 与 -1 之间的数同样受到相应的限制，这里的 9 位有效数字从小数点的右边第一位开始算起。例如：

□ ? 0.1234567891
 .123456789
 □ ? -0.00000123456789
 1.23456789E - 6
 □ ? 0.000000000900000007558
 9.00000008E - 10

总之，凡是用 9 位有效数字能准确表示的一律用定点表示法表示；否则就用浮点表示法表示。

2. 变量

所谓变量就是可以改变其数值的量，而常量可以视为它的特殊情况。APPLES OF T 中变量有两种，即简单变量与下标变量。

(1) 简单变量

简单变量可分为实变量、整变量和字符串变量三种，如表 1-2 所列。

上面的变量名中，第一个字必须是字母，它可以用 26 个英文字母中的任一个；第二个字可以是字母或数字或省略。例如：

A, A1, X1, Y3, AD

均可表示实变量的变量名。

变量名仅头两个字符有效：NOT% 与 NO% 是一样的，但