

TP31 / 60

AppLE II

BASIC 语言、磁盘操作系统及应用



华南师范大学计算机科学研究室印

一九八四年三月

说 明

为目前普及微型计称机的需要，我们编写了这份教材，供一般的微型计称机讲习班之用。由于时间匆促，来不及认真校对，错误之处，请指正。

李冠英 张毅忠 郑存陆

1984. 3

目 录

第一章 概 述

1.1—1.8

第二章 APPLE II 的结构

2.1—2.12

第一节 键盘

第二节 荧光屏显示

第三节 主机

第四节 常用外部设备

第五节 外部设备控制卡

第三章 APPLE II 的使用方法

3.1—3.6

第一节 开机和关机

第二节 键盘操作方法

第三节 APPLE II 两种BASIC语言系统的转换

习 题

第四章 APPLE SOFT BASIC概述及其基本成分 4.1—4.21

第一节 怎样使用 NEW、RUN 及 LIST

第二节 基本符号、常量及变量

第三节 函数

第四节 表达式

习 题

第五章 程序的删改

5.1—5.6

第一节 删除程序

第二节 插入程序行

第三节 修改程序

第六章 赋值语句和输入输出语句

6.1—6.22

第一节 输出语句——PRINT 语句

第二节 赋值语句

第三节 键盘输入语句

- 第四节 GET语句 —— 立即输入
- 第五节 盘常昇语句和读常昇语句
- 第六节 恢复语句 —— RESTORE
- 第七节 荧光屏控制语句
- 第八节 CALEAR、REM、END和POKE 语句
- 第九节 打印机输出
- 第十节 使用卡式录音机存取程序
- 习题

第七章 转移、条件转移与循环语句

7.1 — 7.28

- 第一节 GOTO 语句 —— 无条件转移
- 第二节 "CTRL-C"命令 —— 中止程序执行
- 第三节 STOP 语句 —— 程序暂停
- 第四节 逻辑运算, 条件转向语句
- 第五节 ON... GOTO 语句 —— 选择转向语句
- 第六节 FOR... NEXT 语句 —— 循环语句
- 第七节 ONERR GOTO 与 RESUME 语句 —— 出错转移与处理后返回。
- 第八节 INVERSE 与 NORMAL语句 —— 变换字符的黑白显示
- 第九节 FLASH 语句 —— 黑底白字和白底黑字交替闪烁
- 第十节 SPEED语句 —— 控制输出字符的速度
- 第十一节 TRACE语句 —— 显示执行的语句行号
- 习题

第八章 数组 (ARRAYS)

8.1 — 8.18

- 第一节 数组的概念
- 第二节 DIM 语句 预先存储空间
- 第三节 二维数组
- 习题

第九章 子程序及其转子语句

9.1 — 9.11

第一节 GOSUB 语句——转子程序

第二节 RETURN 语句——结束子程序并转回主程序

第三节 POP 语句——消掉一个子程序返回位置

第四节 子程序嵌套 (Nested subroutines)

第五节 ON... GOSUB 语句——选择转子程序

习题

第十章 图形与声音

10.1 — 10.24

第一节 低分辨率图形

第二节 高分辨率图形

第三节 APPLE II 的声音

习题

第十一章 机口语言的监控系统

11.1 — 11.13

第一节 进入和退出监控系统

第二节 监控系统的功能

第三节 存储口区间资料的存取

第四节 反汇编和机口语言程序的执行

第五节 其它监控命令

第十二章 磁盘操作系统 (DOS) 与文件

12.1 — 12.57

第一节 文件的基本概念

第二节 文件名称 (FILE NAMES)

第三节 磁盘操作系统

第四节 在程序中使用 DOS 命令

第五节 文件结构

第六节 顺序文件 (Sequential Files)

第七节 随机文件 (Random Access Files)

习题

第十三章 汉字系统

13.1 13.14

第一节 汉字系统简介

第二节 CPLUS 汉字卡操作

第三节 APPLE SOFT BASIC语言在汉字卡中的执行

第四节 汉字卡的子程序和软开关

附 录

附录A 美国国家标准码 (ASCII CODE)

附录B APPLE SOFT的保留字及其代码

附录C APPLE SOFT摘要

附录D 错误讯息 (ERROR Messages)

附录E 有特殊用途的 PEEK 和 POKE 工作单元

附录F RENUMBER 程序的功能

附录G APPLE SOFT 零页使用情况

第一章 概 述

现实世界中，电子计算机的应用非常广泛，特别是微型电子计算机已进入社会各个领域；甚至家庭。在微型计算机中，APPLE II是最常见的一种，因此，本书将以APPLE II作为基本工具，以讲述BASIC语言和它的应用。在所述的有关BASIC语句规定中，大部分语句都适合于其他类型的机型，然而结合了具体的机口，便于结合实践，使问题讲得清楚具体。

本书选材时，是假定他的对象是具有初中文化水平，要求学习使用APPLE II或BASIC语言的读者，在各个问题的说明时，尽可能通俗易懂，多用实例。

本书的编排顺序是，在第一章介绍计算机的一些概念，第二章和第三章主要介绍APPLE II的基本结构和一般使用方法，第四章至第九章主要介绍APPLESOFT BASIC语言，第十章主要介绍用APPLE II显示器显示图形和用扬声器发出声音的方法，第十一章主要介绍APPLE II的监控程序的使用方法，第十二章介绍APPLE II的操作系统，十三章介绍APPLE II的汉卡及其使用方法；十四章列举了一些实用程序。一些参考资料汇集于书本后面的附录中。

下面分别介绍微型计算机的一些概念，以便初步接触计算机的读者，对它有个初步的认识。

一、微型计算机

自从1946年美国宾夕法尼亚大学的莫希莱(John Mauchly)和埃克特(J-Presper Eckert)等人设计成第一台数字电子计算机以来，电子计算机经历了四代：主要元件用电子管的称电子管计算机或第一代计算机，主要元件用晶体管的称晶体管计算机或第二代计算机；主要元件用中、小规模集成电路的称集成电路计算机或第三代电子计算机，主要元件用大规模集成电路的称为第四代电子计算机，由第一代到第四代，计算机的性能不断增加，体积不断缩小，价格不断下降，使各个领域中都可能使用电子计算机。反过来，由于电子计算机被广泛地应用，又促使它和集成电路的不断发后，微型计

机是一种大规模集成电路的产物，由于大规模集成电路工艺的发展，生产集成电路的美国英特尔（INTEL）公司于1971年制成一种叫INTEL 4004的微处理器，这种微处理器包括了能同时对四位二进制数进行运算的运算和控制处理功能的电路，称它为四位微处理器，或简称MPU（Microprocessing unit）。1974年以后，英特尔公司，莫托洛拉（MOTOROLA）等公司研制成INTEL 8080、M6800等八位微处理器和INTEL 8086、M68000等十六位微处理器，现正研制32位微处理器，使目前市场上售的微处理器种类繁多，各有其特点。人们把这种用微处理器作为计算机的中心处理单元（简称CPU）装制而成的电子计算机，称为微计算机。目前市场上售的微型计算机品种繁多，在各个领域得到广泛的应用，APPLE II 微型电子计算机就是其中的一种。

二、二进制数、八进制数和十六进制数

我们最熟悉的记数法是十进制记数法，这种记数法的特点是：

1. 使用十个基数0, 1, 2, 3, 4, 5, 6, 7, 8, 9。
2. “逢十进一”。即每位的位值是它右邻的十倍。对于一个整数来说，从右到左称个位，十位，百位，千位，……。例如“1984”就是“一千九百八十四”。

计算机内部，一般使用二进制数，这种数的记数特点是：

1. 二个基数：0, 1。
2. “逢二进一”。即每位的位值是它右邻的二倍。例如二进制数100, 101, 1011等，是用0和1组成的数。我们不要把二进制数100读成十进制数的“一百”，把101读成“一百零一”。为了不致使二进制数和十进制数相混淆，常常在数字后面加“+”表示该数是十进制数，加“=”表示该数是二进制数。例如100+表示十进制，100=表示100是二进制数。

八进制数和十六进制数

上面读到二进制数由一串“0”和“1”组成，读写都很不方便。为了方便读写等原因，也常常使用八进制数及十六进制数。

八进制数的记数法则是：

- (1) 使用八个基数：1, 2, 3, 4, 5, 6, 7, 0.
- (2) “逢八进一”。每位的位值是它右邻的八倍，可在数后面加“八”，表示八进制数，例如172.4八等。

八进制数的基数和二进制数的对立关系是：

八进制数	0	1	2	3	4	5	6	7
二进制数	000	001	010	011	100	101	110	111

因此将一个八进制数写成二进制数，只需将八进制数各位所对应的二进制数写出即可，例如：

八进制数 1 4 2

对应的二进制数 001 100 010 = 1100 01

反之，要将一个二进制数写成对应的八进制数，可用以下方法，对于整数部份，从小数点向左每三位二进制数分成一组，最后一组若不是三位则用0补够，小数部分，从小数点开始向右每三位二进制数分为一组，最后一组不足三位也是用0补足，再把各组所对应的八进制数写出，即是所对应的八进制数。

例如

二进制数

1100.01 可写成 001100.010

对应的八进制数

1 4 2

= 14.2

2. 十六进制数

十六进制数的记数法则如下：

- (1) 使用十六个基数：0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.
- (2) “逢十六进一”。即每位的位值是它右邻的十六倍，一个十六进制数可写成 64_{16} , 42_{16} 等等，有时也可以在数字后面加H表示十六进制如 $64H$, $42H$ 等。本书后面使用十六进制数表示存储器地址时，在数字前面加“\$”号，以表示用十六进制数表示地址，如 $\$6A$, $\$42$ 等。

十六进制数的基数与二进制的对应关系如下表述：

十六进制数	0	1	2	3	4	5	6	7
二进制数	0000	0001	0010	0011	0100	0101	0110	0111
十六进制数	8	9	A	B	C	D	E	F
二进制数	1000	1001	1010	1011	1100	1101	1110	1111

由此，要将一个十六进制写成对应的二进制数，只要将十六数的各位写成对应的二进制数即可，例如：

十六进制数

6A+6

对应的二进制数

0110 1010 = 即 110 10 10 =

反之，要将一个二进制数写成对应的十六进制数，可用下面方法，查数部分，从小数点开始向左每四位二进制数分为一组，最后一组若不足四位，用0补足；小数部分，从小数点开始向右每四位二进制数分成一组，最后一组若不足四位，用0补足，再将各组二进制数的十六进制数写出来，例如：

二进制数

110.1001.01 写成 0110 1010.0100

对应十六进制

6 A . 4 +6

即 6A.4+6

三、字节、字长

(一)、二进制位 (bit)

bit是binary digit 的缩写，它是计算机中度量信息的单位，也是存储资料的最小单位，一个bit能存储一位二进制数“0”或“1”。

(二)、字节 (byte)

字节是用来处理一串二进制数的单位，通常是八位二进制数组成一个byte。因为八位二进制数可以组成00000000, 00000001, ..., 11111111 ($2^8 = 256$) 种不同的状态，足以代表计算机所使用的数字(0, 1, 2, 3, ..., 9)字母(A, B, C, ..., Z)和运算符(+, -, ..., 等)。

(三)、字长 (word length)

电子计算机中，字是计算机处理信息的单元，通常，一个字所包含二进制数的个数，叫做字长。各种计算机的字长可以相同，也可以不相同，如IBM 370的字长是32位，即由四个字节组成，国产系列100计算机的字长是16位，由二个字节组成。

APPLE II 微型计算机的字长是8位，即由一个字节组成，一般来说，字长较大的计算机处理能力较强，因此，计算机字长常常被用来说明计算机功能的标志之一。

(四) 地址、指令

在计算机中，地址是用来标识寄存器、存储单元或外部设备的编号或名字，在指令码中，它规定操作数所在的位置。对于存储单元的地址编号，一般采用八进制数或十六进制数，例如APPLE II的地址分配是(用十六进制)：

0000~BFFF 随机存储单元 RAM 48K byte

C000~CFFF 外部设备 4K byte

D000~FFFF 固定存储单元 12K byte

注：数字“0”写成“O”是为了避免与英文字母中的“O”相混淆。

所谓指令是指计算机执行操作的命令，电子计算机进行各种操作，是指不同的指令进行的，各种不同的计算机具有不同的指令系统，每一条指令都是由一位数值的二进制数码组合而成，分成操作码和地址码两部分，即

操 作 码	地 址 码
-------	-------

一般是操作码放在指令的前面，地址码部分是在指令的后面，每条指令只有一个操作码，但可以有几个地址码，如有两个地址的指令叫二地址指令，三个地址码的指令叫三地址指令，通过下一个问题的讨论，我们还可以进一步了解指令的意义。

五. 程序设计语言

人们使用计算机，首先要把想做的事，用人和机都能识别的语言编写程序，计算机执行了这个程序，才能完成预定的目的。程序设计语言是指人们用来编制计算机程序的语言，目前，使用的程序设计语言很多，但从构成来说大致可以分成机内语言；汇编语言和高级语言三类。

(一) 机内语言

机内语言是直接使用机内的指令作为程序设计的语言，机内的指令是计算机的设计者所制定的，一部计算机的所有指令就组成这个机内的指令系统，各种计算机有不同的指令

系统，例如：如果选是二进制码

0000 代表加之操作码

0001 代表减之操作码

0010 代表乘之操作码

0011 代表除之操作码

计算机遇到操作码为0000时做加法，遇到操作码0001就做减法，……。这种直接使用机口指令依附计算机的语言，称为机口语言。假如用二进制码表示的操作码和操作数是：

"+"的操作码 0000

"B"的地址为 0100

"C"的地址为 0101

将B单元的数据加C单元的数据且存入B单元的指令格式为：

操 作 码	第 一 地 址	第 二 地 址
-------	---------	---------

则机口指令可写成

0000 0100 0101

计算机遇到这样的一串二进制码，就知道要进行 $B = B + C$ 的操作，这个操作就是将地址0100内存的数据与存在地址为0101内的数据相加，结果存入0100内。

因此，用机口语言编写程序不但极其繁琐，且容易发生错误。不同的机口，使用的机口语言不同，使用非常不方便。

(二)、汇编语言

汇编语言不是用一连串二进制数代表指令，而是用英文字母及符号代表指令，例如表示上述 $B = B + C$ 操作的一串二进制码：0000 0100 0101

写成 ADD BC

其意义就是将B单元的数据加C单元的数据，结果存入B单元中，人们使用起来就方便得多了。

(三)、高级语言

高级语言比较接近人们的习惯，例如计算 $A = B + C + E$ ，在BASIC语言中（一种高级语言）可以使用语句。

$A = B + C + D + E$

因此，高级语言较易为人们接受，而且通用性较强，即不同的机口，可以使用相同的高级语言，目前常用的高级语言有BASIC。

COBOL, FORTRAN 和 ALGOL 等。

BASIC 语言，叫初学者通用的符号指令代码语言，它的特点是简单，易学，使用方便，并可进行人机对话，极易初学者学习和使用。目前大部分微型计算机都可以使用这种语言。

COBOL 语言，即面向商业的公共语言，它是商业数据处理而设计的计算机语言，主要用于数据处理。

FORTRAN 语言及 ALGOL 语言，它们都适用于描述数值计算过程，是在科学，工程设计技术上，使用最为广泛的一种计算机语言。

六. 流程图

流程图是解决问题步骤和图形表示，它借助各种不同的图形和箭头来表达解决问题的顺序，每一种图形代表一种作业功能，而箭头则表示流程的方向，它可帮助我们设计各种较为复杂的程序，也有助于程序的阅读和检查，根据 GB1526-79 标准规定的信息处理流程图的图形符号和使用规则，将常用的列表如下，见图 (1-1)。

我们用下面的例子说明上述符号的使用方法。

例如某车站收旅客行李托运费，规定每张客票托运行车不超过 50 公斤时，每公斤按 0.15 元计收，如果超过 50 公斤，则超过部分按每公斤 0.25 元计收，假定旅客托运 x 公斤行李，运费 y 的计收方法是：

$$y = \begin{cases} 0.15 * x & x \leq 50 \\ 0.15 * 50 + (x - 50) * 0.25 & x > 50 \end{cases}$$

表示这个计收方法的流程图如图 (1-2) 所示。

七. 美国的信息交换标准代码 (ASCII)

用于代表字符的最常用的字母数字代码是美国信息交换代码，简称 ASCII 码，它是 American Standard Code for Information Interchange 的缩写，代表字母，数字或符号的 ASCII 码可以用八进制数码来书写。这本书后面的附录 A，列举了 APPLE II 键盘的键字所对应的 ASCII 码，它是用十进制数表示的，从表中我们可以看到字母 A 的 ASCII 是十进制数 65，B 是 ASCII 码是十进制数 66……等。

图形



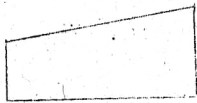
功能
起始或终止



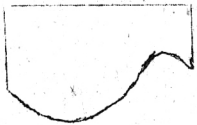
一般处理



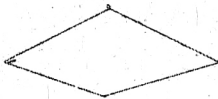
输入或输出



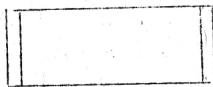
由键盘输入



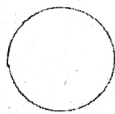
打印输出



测试判断



调用子程序

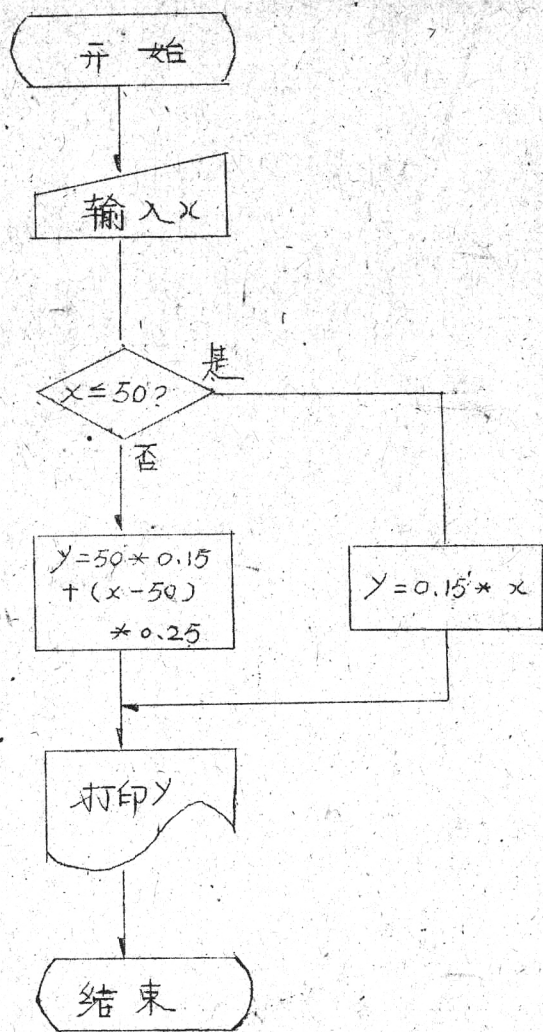


连接



流程方向

图(1-1)



图(1-2)

第二章 APPLE II 的结构

1977年元月，APPLE 的雏形室教研究成功，至今经过了不断的改进。其中功能上的改进，如加上彩色绘图功能，增加录音机上的可靠性等；工业上为了降低生产成本，淘汰旧型 4KRAM 及改进使其适合欧洲电视系统等；特别是它的主机板，有三种不同的产品，1981 年以前采用“NON-RFI 主机板”。1981 年 APPLE 公司为了避免其产品违反 FCC（联邦通讯协会）的规定（由电脑快速交换电路所产生的电磁干扰的最低限度），重新设计了其主机板，称为“RFI 主机板”，产品编号为 820-0044-xx；1983 年 APPLE 公司又推出 APPLE IIe，使其功能增强。目前市场销售的多为“RFI 主机板”，故以下所讲的 APPLE II 是使用以“RFI”主机板那种。

另一方面从整机系统来说，还有一种叫 APPLE PLUS 的计算机，它与 APPLE II 不同之处在于在主机板上固定存贮口中存的 BASIC 语言解释程序不同，APPLE II 主机板上的是查数 BASIC 而 APPLE II PLUS 主机板上的是 APPLE SOFT BASIC（实数 BASIC），目前多数人都采用 APPLE SOFT BASIC，故以它作为书中以后各章讨论的内容。

APPLE II 微型计算机采用积木式结构，其基本部分包括键盘（输入设备），荧光屏显示口（输出设备）和主机，此外，还可以根据使用的需要增加卡式录音机、磁盘机、打印机、绘图板等外部设备，外形如图（2-1）。下面对键盘、荧光屏显示口和主机三个部分加以说明，对其设备也作些有关的简单介绍。

第一节 键盘

键盘是使用者用于输入数据、程序和命令的部分，它类似于英文打字机的键盘，也有一些键是英文打字机的键盘中所没有的，APPLE II 的键盘如图（2-2）所示。这些键的使用将在第三章