

计算机等级考试辅导

(二级)

FORTRAN 程序设计

谭浩强 主编
徐士良 赵鸿德 编著

清华大学出版社

目 录

第一章 计算机基本知识	1
1.1 本章要点	1
1.1.1 微型机的硬件与软件系统	1
1.1.2 计算机中数的表示及编码	3
1.1.3 DOS 操作系统概述	7
1.1.4 常用 DOS 命令的使用及常见错误分析	11
1.1.5 微型机的系统维护	23
1.1.6 计算机病毒及其防治	24
1.2 本章难点	25
1.2.1 计算机中数的表示及运算	25
1.2.2 磁盘文件的树状目录结构	27
1.2.3 DOS 命令的正确使用	28
1.2.4 输入输出改向	30
1.2.5 几种打印方式的比较	30
1.3 例题分析	31
1.3.1 选择题	31
1.3.2 填空题	35
1.4 思考题	38
1.4.1 选择题	38
1.4.2 填空题	40
第二章 FORTRAN 程序的结构、书写规则	42
2.1 本章要点	42
2.1.1 FORTRAN 程序的构成	42
2.1.2 FORTRAN 源程序的书写格式	43
2.2 本章难点	44
2.2.1 FORTRAN 程序的模块结构	44
2.2.2 注意正确的书写格式	44
2.3 例题分析	45
2.3.1 选择题	45
2.3.2 填空题	46
2.4 思考题	46
2.4.1 选择题	46
2.4.2 填空题	47
第三章 数据类型及其运算	48

3.1	本章要点	48
3.1.1	常量与变量	48
3.1.2	符号常量及其定义	51
3.1.3	各种运算符及其运算的优先级	53
3.1.4	各种表达式的求值规则	54
3.2	本章难点	57
3.2.1	变量定义的 I-N 规则	57
3.2.2	不同类型数据的混合运算	57
3.2.3	逻辑表达式的运算顺序	58
3.3	例题分析	59
3.3.1	选择题	59
3.3.2	填空题	62
3.4	思考题	64
3.4.1	选择题	64
3.4.2	填空题	65
第四章	最基本的语句	66
4.1	本章要点	66
4.1.1	各种赋值语句	66
4.1.2	表控输入输出	67
4.1.3	格式输入输出	68
4.1.4	STOP 语句与 END 语句的功能	75
4.1.5	DATA 语句	76
4.2	本章难点	77
4.2.1	如何用好格式输入输出	77
4.2.2	DATA 语句的正确使用	80
4.3	例题分析	80
4.3.1	选择题	80
4.3.2	填空题	84
4.4	思考题	86
4.4.1	选择题	86
4.4.2	填空题	88
第五章	选择结构程序设计	90
5.1	本章要点	90
5.1.1	逻辑 IF 语句	90
5.1.2	块 IF 结构	90
5.1.3	块 IF 结构的嵌套	91
5.2	本章难点	93
5.2.1	ELSE IF 实现多路分支	93

5.2.2	要用对 END IF 语句	95
5.2.3	要写对“条件”	95
5.3	例题分析	98
5.3.1	选择题	98
5.3.2	填空题	103
5.4	思考题	108
5.4.1	选择题	108
5.4.2	填空题	112
第六章	循环结构程序设计	116
6.1	本章要点	116
6.1.1	当型循环与直到型循环	116
6.1.2	DO 语句实现循环	119
6.1.3	循环嵌套	121
6.2	本章难点	123
6.2.1	DO 循环中循环次数的计算	123
6.2.2	如何选择合适的循环形式	124
6.3	例题分析	125
6.3.1	选择题	125
6.3.2	填空题	130
6.4	思考题	135
6.4.1	选择题	135
6.4.2	填空题	141
第七章	数组的应用	145
7.1	本章要点	145
7.1.1	数组的定义	145
7.1.2	数组元素的正确使用	147
7.1.3	数组在内存中的存储形式	147
7.1.4	用 DATA 语句给数组赋初值	148
7.1.5	数组的输入与输出	149
7.2	本章难点	153
7.2.1	正确定义数组与使用数组元素	153
7.2.2	多维数组的存储形式	154
7.2.3	二维数组的格式输出	155
7.3	例题分析	156
7.3.1	选择题	156
7.3.2	填空题	162
7.4	思考题	166
7.4.1	选择题	166

7.4.2	填空题	171
第八章	函数与子程序	175
8.1	本章要点	175
8.1.1	内部函数与语句函数	175
8.1.2	函数子程序的结构与调用方法	175
8.1.3	子例行程序的结构与调用方法	176
8.1.4	形式参数与实在参数之间的数据传递——虚实结合	177
8.2	本章难点	180
8.2.1	函数子程序与子例行程序的比较	180
8.2.2	可调数组的使用	180
8.2.3	正确使用形参变量	182
8.3	例题分析	182
8.3.1	选择题	182
8.3.2	填空题	187
8.4	思考题	192
8.4.1	选择题	192
8.4.2	填空题	195
第九章	数据联系	198
9.1	本章要点	198
9.1.1	公用语句(COMMON 语句)	198
9.1.2	数据块子程序	198
9.2	本章难点	199
9.2.1	无名公用区与有名公用区的区别	199
9.2.2	对数据块子程序的若干规定	202
9.3	例题分析	202
9.3.1	选择题	202
9.3.2	填空题	205
9.4	思考题	206
9.4.1	选择题	206
9.4.2	填空题	208
第十章	字符处理	210
10.1	本章要点	210
10.1.1	字符串和字符型变量	210
10.1.2	字符型数组	212
10.1.3	字符型数据的赋值和运算	212
10.1.4	字符型数据的输入输出	213
10.2	本章难点	215
10.2.1	字符型数据的处理	215

10.2.2	如何输出一个图案.....	217
10.3	例题分析.....	218
10.3.1	选择题.....	218
10.3.2	字符串检索的编程技巧.....	221
10.3.3	翻译密码.....	222
10.3.4	图案输出.....	223
10.3.5	曲线输出.....	225
10.4	思考题.....	226
10.4.1	选择题.....	226
10.4.2	填空题.....	228
第十一章	文件	230
11.1	本章要点.....	230
11.1.1	文件与记录的概念.....	230
11.1.2	文件的打开与关闭.....	231
11.1.3	顺序文件的存取方法.....	232
11.1.4	直接文件的存取方法.....	233
11.1.5	文件的输入与输出操作.....	233
11.2	本章难点.....	234
11.2.1	如何打开一个文件.....	234
11.2.2	有格式顺序文件如何读写.....	235
11.2.3	有格式直接文件如何读写.....	240
11.2.4	无格式顺序文件和直接文件.....	242
11.3	例题分析.....	243
11.3.1	选择题.....	243
11.3.2	填空题.....	245
11.4	思考题.....	247
11.4.1	选择题.....	247
11.4.2	填空题.....	248
第十二章	程序调试与分析	252
12.1	本章要点.....	252
12.1.1	FORTRAN 77 程序的运行环境.....	252
12.1.2	FORTRAN 77 程序的建立、编译、连接和运行.....	253
12.2	本章难点.....	255
12.2.1	程序调试步骤.....	255
12.2.2	FORTRAN 77 程序的动态调试.....	259
12.3	FORTRAN 77 程序的调试举例分析.....	262
12.3.1	FORTRAN 77 程序的上机操作过程.....	262
12.3.2	FORTRAN 77 程序常见错误分析.....	264

第一章 计算机基本知识

1.1 本章要点

1.1.1 微型机的硬件与软件系统

一个完整的计算机系统由硬件系统与软件系统两大部分组成。

一、微型机的硬件系统

1. 中央处理器(又称 CPU)

它是由运算器和控制器两部分组成的。其中运算器负责数据的算术运算和逻辑运算;控制器负责提供控制信号,协调并控制输入输出操作以及对内存的访问。根据 CPU 能同时处理的数据位数,常见的微机有 8 位机、16 位机、32 位机,目前又推出了 64 位的高档微机。CPU 的品质决定了微机系统的档次。

2. 存储器

它用于存放程序或数据信息。存储器中的存储单元是以字节为单位进行编址的。存储器的容量是存储器中所包含的字节数,通常用 KB 或 MB 表示,有时也用 K 或 M 来表示。其中 $1\text{KB}=1024$ 字节, $1\text{MB}=1024\text{KB}$ 。

存储器又分为内存与外存。

内存存储器按其工作方式又可以分为随机读写存储器 RAM 和只读存储器 ROM。RAM 中的数据可以随机地读出和写入,断电后, RAM 中的信息会丢失。ROM 中的数据只能读出而不能写入新的内容,断电后, ROM 中的信息不会丢失。在一般的 PC 机中,内存容量为 640KB,它包括了 RAM 和 ROM,档次较高的微机其内存容量还可以扩充。

内存与 CPU 一起构成了计算机的主机部分。

外存储器又称为辅助存储器。外存的容量一般要比内存大得多,而且可以移动,便于不同系统之间的信息交流。在微机中,最常见的外存储器有磁盘,包括硬盘和软盘。

硬盘的容量要比软盘大得多,目前一般微机所配置的硬盘容量在 60MB 以上。

目前微机上常用的软盘有:容量为 360K 的 5 吋双面双密度(DS,DD)软盘、容量为 1.2M 的 5 吋双面高密度(DS,HD)软盘与容量为 1.44M 的 3 吋双面高密度软盘。5 吋软盘的一侧有一个写保护口,当贴上保护纸后只能读而不能写入。同样,在 3 吋软盘上有一个写保护孔。

在第一次使用磁盘时,必须首先对磁盘格式化。

3. 输入设备

它是外界向计算机传送信息的装置。在微机系统中,最常用的输入设备是键盘。

磁盘机(即磁盘驱动器)也是一种输入设备,它将磁盘上的信息传送到主机中。

4. 输出设备

它的作用是将计算机中的数据信息传送到外部媒介,并转化成某种为人们所认识的表示形式。在微机系统中,最常用的输出设备有显示器和打印机。

显示器和键盘构成了微机系统的控制台。

磁盘机也是一种输出设备,它负责将主机中的信息传送到磁盘上保存起来。

二、微型机的软件系统

通常,软件包括计算机运行所需的各种程序和数据,以及有关的文档。

硬件是软件的物质基础,但硬件能否发挥其作用又取决于软件。

软件一般可以分为系统软件、应用软件和支撑软件。

1. 系统软件

系统软件通常是指管理、监控和维护计算机资源(包括硬件和软件)的一种软件。最常用的系统软件有:

- 操作系统;
- 各种语言处理程序,如汇编程序、编译程序及解释程序等;
- 数据库管理系统,如 FoxBASE 等。

2. 应用软件

应用软件是指利用计算机及系统软件为解决各种实际问题而编制的、具有专门用途的软件。常见的应用软件有:

- 各种字处理软件,如,汉字编辑软件 WORDSTAR、汉字字表编辑软件 CCED 及桌面印刷系统 WPS 等;
- 各种用于科学计算的软件包;
- 计算机辅助制造、辅助设计、辅助教学等软件;
- 各种图形软件等。

3. 支撑软件

支撑软件是指在计算机硬件与系统软件的基础上,用于支援其他软件研制和开发的软件。它的目的是方便用户编制应用软件。

三、微型机的分类及主要性能指标

1. 微型机的分类

• 按字长分

分为 8 位机、16 位机、32 位机和 64 位机。

• 按结构分

分为单片机、单板机、多芯片机与多板机。

• 按用途分

分为工业过程控制机和数据处理机。

2. 微型机主要性能指标

• 字长

- 时钟频率
- 内存容量
- 外部设备配置
- 软件配置

四、微型机的应用领域

微型机的应用领域很广泛,且应用水平也越来越深入。主要应用有以下几个方面:

- 科学技术计算;
- 数据处理;
- 过程控制;
- 计算机通信;
- 计算机辅助设计、辅助制造、辅助测试与辅助教学等。

1.1.2 计算机中数的表示及编码

计算机中的所有信息都是以二进制形式表示的。为了阅读和书写方便,一般又用八进制或十六进制表示。

一、计算机常用计数制

1. 二进制数

在二进制数中,基数为 2。因此在二进制数中出现的数字字符只有两个:0 与 1。每一位计数的原则为“逢二进一”。

要将十进制整数转换为二进制整数可以采用“除 2 取余”法:将十进制数除以 2,得到一个商数和余数,再将商数除以 2,又得到一个商数和余数。这个过程一直做下去,直到商数为 0 为止,每次得到的余数即为对应二进制数的各位数字,但必须注意:最先得到的余数为二进制数的最低位,最后得到的余数为二进制数的最高位。

例如,将十进制整数 69 转换为二进制整数,过程如下:

2 69	余数为 1, 即 $a_0=1$
2 34	余数为 0, 即 $a_1=0$
2 17	余数为 1, 即 $a_2=1$
2 8	余数为 0, 即 $a_3=0$
2 4	余数为 0, 即 $a_4=0$
2 2	余数为 0, 即 $a_5=0$
2 1	余数为 1, 即 $a_6=1$
0	商为 0, 结束

因此, $(69)_{10} = (1000101)_2$

要将十进制纯小数转换为二进制小数可以采用“乘 2 取整”法:用 2 乘十进制小数,其整数部分为 a_{-1} ,再用 2 乘余下的纯小数部分,其整数部分为 a_{-2} 。这个过程一直做下去,直

到余下的纯小数为 0 或满足所要求的精度为止。最后可得到对应的二进制小数 $(0.a_{-1}a_{-2}\dots a_{-n})_2$ 。必须注意,一个十进制小数不一定能准确地转换为二进制小数。

例如,将十进制小数 0.357 转换为二进制小数,过程如下:

$$\begin{array}{r}
 0.357 \\
 \times 2 \\
 \hline
 0.714 \quad \text{整数为 0, 即 } a_{-1}=0 \\
 \\
 0.714 \\
 \times 2 \\
 \hline
 1.428 \quad \text{整数为 1, 即 } a_{-2}=1 \\
 \\
 0.428 \\
 \times 2 \\
 \hline
 0.856 \quad \text{整数为 0, 即 } a_{-3}=0 \\
 \\
 0.856 \\
 \times 2 \\
 \hline
 1.712 \quad \text{整数为 1, 即 } a_{-4}=1 \\
 \\
 0.712 \\
 \times 2 \\
 \hline
 1.424 \quad \text{整数为 1, 即 } a_{-5}=1 \\
 \\
 \vdots
 \end{array}$$

上述过程可以一直做下去,但是不能穷尽。如果要求只取到二进制小数点后第五位为止,则

$$(0.357)_{10} \approx (0.01011)_2$$

对于一般的十进制数,可以分别将整数部分转换与小数部分转换后再组合起来。例如:

$$(69.357)_{10} \approx (1000101.01011)_2$$

2. 八进制数

在八进制数中,基数为 8。因此,在八进制数中出现的数字字符有 8 个:0,1,2,3,4,5,6,7。每一位计数的原则为“逢八进一”。

与二进制数类似,将十进制整数转换为八进制整数可以采用“除 8 取余”法;十进制小数转换为八进制小数可以采用“乘 8 取整”法。

二进制数转换为八进制数采用如下方法:从小数点开始,向前每三位一组构成一位八进制整数位;从小数点开始向后每三位一组构成一位八进制小数位,若最后不足三位二进制位,则应添 0 补足三位。例如:

$$\begin{aligned}
 (69.357)_{10} &\approx (1000101.01011)_2 \\
 &= (1,000,101.010,110)_2 \\
 &= (105.26)_8
 \end{aligned}$$

3. 十六进制数

在十六进制数中,基数为 16。因此,在十六进制数中出现的数字字符有 16 个:0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F,其中 A、B、C、D、E、F 分别表示值 10,11,12,13,14,15。十六进制数中每一位计数原则为“逢十六进一”。

将十进制数转换为十六进制数的方法与二进制数、八进制数类似,其整数部分采用“除 16 取余”法,小数部分采用“乘 16 取整”法。

二进制数转换为十六进制数采用如下方法:从小数点开始向前每四位一组构成一位十六进制整数位;从小数点开始向后每四位一组构成一位十六进制小数位,若最后不足四位二进制位,则应添 0 补足四位。例如:

$$\begin{aligned}
 (69.357)_{10} &\approx (1000101.01011)_2 \\
 &= (0100,0101.0101,1000)_2 \\
 &= (45.58)_{16}
 \end{aligned}$$

同样,也可以很方便地将十六进制数转换为二进制数。

二、计算机中数的表示

数有正有负。在计算机中表示一个数时,总是用最高位表示数的符号,其中“0”表示正,“1”表示负。

在计算机中,小数点位置固定的数称为定点数。通常,计算机中的定点数有两种表示法:

- 小数点默认为在一个二进制数最后一位的后面。这种定点数称为定点整数。
- 小数点默认为在二进制数的最高位(即符号位)后面。这种定点数称为定点小数。

在计算机中,既有整数部分又有小数部分的数称为浮点数。通常,浮点数表示为:

$$P = \pm S \times 2^{\pm N}$$

其中 P, S, N 均为二进制数。S 称为 P 的尾数,表示成定点小数;N 称为 P 的阶码,表示成定点整数。

由此可知,浮点数表示中的各部分又都是定点数。不管是定点整数还是定点小数,由于它们的小数点位置固定,因此在运算时可以不考虑小数点的位置,只要区分是定点整数还是定点小数就可以了。

在计算机中,对于有符号的定点数又有三种表示法:原码、反码与补码。

1. 原码

原码表示中,符号位在最高位,“0”表示正,“1”表示负,其数值部分按一般二进制形式表示。下面是原码表示的例:

$$\begin{aligned}
 (+50)_{10} &= (+0110010)_2 = (00110010)_{\text{原}} \\
 (-50)_{10} &= (-0110010)_2 = (10110010)_{\text{原}} \\
 (+127)_{10} &= (+1111111)_2 = (01111111)_{\text{原}}
 \end{aligned}$$

$$(-127)_{10} = (-1111111)_2 = (1111111)_{\text{原}}$$

在用原码表示时,两个异号数相加时很不方便。

2. 反码

正数的反码是原码本身,负数的反码为该数原码除符号位外各位求反(即“0”变为“1”,“1”变为“0”)。例如:

$$(+50)_{10} = (00110010)_{\text{原}} = (00110010)_{\text{反}}$$

$$(-50)_{10} = (10110010)_{\text{原}} = (11001101)_{\text{反}}$$

$$(+127)_{10} = (01111111)_{\text{原}} = (01111111)_{\text{反}}$$

$$(-127)_{10} = (11111111)_{\text{原}} = (10000000)_{\text{反}}$$

可以验证:一个数的反码的反码是原码本身。

3. 补码

正数的补码是原码本身,负数的补码是该数的反码最后一位加1。例如:

$$(+50)_{10} = (00110010)_{\text{原}} = (00110010)_{\text{补}}$$

$$(-50)_{10} = (10110010)_{\text{原}} = (11001110)_{\text{补}}$$

$$(+127)_{10} = (01111111)_{\text{原}} = (01111111)_{\text{补}}$$

$$(-127)_{10} = (11111111)_{\text{原}} = (10000001)_{\text{补}}$$

同样可以验证:一个数的补码的补码还是原码本身。

在计算机中,加减法运算可以统一化成补码的加法运算,其符号位参与一起运算,结果为补码形式,这是十分方便的。

在上述三种表示中,反码只是作为求补码过程的一种中间形式。为了运算方便,计算机中所有的定点数一般都化为补码表示。

最后必须指出:当数的绝对值超过所用的二进制位数允许表示的最大值时,就会发生溢出,从而造成运算错误。上述的例子中,均用八位二进制位表示一个有符号定点数,因此,用补码表示的数的范围是: $-128 \sim +127$, 如果所要表示的数超过这个范围,就要溢出。如果用十六位二进制位表示数,则用补码表示的数的范围是: $-32768 \sim 32767$ 。

三、字符编码与汉字编码

1. 字符编码

目前微型机中普遍采用的字符编码是 ASCII 码。它是用七位二进制数对 127 个字符进行编码,其中前 32 个是一些不可打印的控制符号。

在 ASCII 编码中,十个数字符号、26 个大写英文字母、26 个小写英文字母是分别连续编码的。因此,如果知道了一个数字符号的 ASCII 码,就可推出其它数字符号的 ASCII 码。同样,对于英文大写字母与小写字母的 ASCII 码也是如此。

2. 汉字编码及字模信息

汉字有两种编码:国标码与机内码。

国标码是“中华人民共和国国家标准信息交换汉字编码”,代号为“GB2312-80”。在国标码的字符集中,收集了一级汉字 3755 个,二级汉字 3008 个,图形符号 682 个,共 7445 个。一个汉字对应一个“区位码”,由四位数字组成,前两位数字为区码(0~94),后两位数

字为位码(0~94)。

机内码是指汉字在计算机中的编码。由于区位码中的区码及位码与基本字符的ASCII码有冲突,因此,汉字的机内码与区位码稍有区别。

汉字的机内码占两个字节,分别称为机内码的高位与低位。它们与区位码的关系如下:

$$\text{机内码高位} = \text{区码} + \text{A0H}$$

$$\text{机内码低位} = \text{位码} + \text{A0H}$$

例如,汉字“啊”的区位码为“1601”,区码为16,位码为01,如果分别用十六进制表示,则区码为10H,位码为01H。因此,该汉字的机内码为B0A1H,其中高位为B0H,低位为A1H。

汉字字库是由所有汉字的字模信息构成的。一个汉字字模信息占若干字节,究竟占多少个字节由汉字的字形决定。例如,如果用 16×16 点阵表示一个汉字,则一个汉字占16行,每行有16个点,在存储时用两个字节存放一行上16个点的信息,对应位为“0”表示该点为“白”,“1”表示该点为“黑”。因此,一个 16×16 点阵的汉字占32个字节。同样, 32×32 点阵的一个汉字要用128个字节来存储。其它规格的汉字可以此类推。

在输出一个汉字时,首先根据该汉字的机内码找出对应字模信息在汉字库中的位置,然后取出这些字模信息作为图形在屏幕上显示或在打印机上打印输出。

1.1.3 DOS 操作系统概述

一、DOS 系统的组成及功能

这里所说的DOS操作系统是指美国Microsoft公司为IBM PC微机及兼容机开发的单用户、单任务磁盘操作系统,即MS-DOS。

MS-DOS主要由三个程序模块组成,它们分别是:输入输出系统(IBMIO.COM)、文件管理系统(IBMDOS.COM)与命令处理程序(COMMAND.COM)。其中输入输出系统还包括了直接与计算机硬件打交道的软件模块BIOS。

IBMIO.COM与IBMDOS.COM是DOS系统的两个重要模块,它们是以隐含的方式存放在DOS系统盘上,一般用户看不到这两个文件。

在模块COMMAND.COM中,包括了DOS中所有内部命令的处理子程序。除此之外,它还具有以下功能:

- (1) 对用户输入的DOS内部命令进行解释并执行;
- (2) 对错误中断和键盘中断进行处理;
- (3) 负责将外部命令的命令文件装入内存并执行。

DOS命令分为内部命令及外部命令。内部命令包括在模块COMMAND.COM中,因此是常驻内存的。外部命令存放在磁盘中,只有当要执行外部命令时,才将相应的命令文件调入内存并执行,执行完后,内存中就不再保留该命令文件。

二、DOS 启动

DOS的启动方式有三种:加电冷启动、按主机箱面板上的“RESET”键及按组合控制键Ctrl-Alt-Del热启动。

其中前两种启动过程是一样的,而后一种启动不经过系统自检。

DOS 系统的启动过程如图 1.1 所示。

三、DOS 键盘的使用

在 DOS 状态下,除了一般的字符键外,还定义了一些具有特定功能的键,主要有以下两种。

1. 组合控制键

组合控制键为用户提供了对系统运行进行某种干预的机会,它一般由两个或三个键同时动作组合而成。

DOS 常用的组合控制键如下:

Ctrl-Alt-Del	热启动 DOS 系统。
^ C 或 Ctrl-Break	终止当前操作。
^ P 或 Ctrl-PrtSc	将标准输出同时送到屏幕和打印机。
Shift-PtrSc	在打印机上产生屏幕的硬拷贝。
^ S 或 Ctrl-NumLock	暂停标准输出设备的输出。
^ H 或 BackSpace	退格并删除一个字符。
^ J 或 Ctrl-Enter	结束物理行但不结束逻辑行。

2. 功能键

DOS 功能键又称 DOS 编辑键,用于对 DOS 命令的编辑修改。

F1	复制输入缓冲区中当前位置上的一个字符到屏幕。
F2<字符>	将输入缓冲区中直到指定字符为止的内容复制到屏幕。
F3	将输入缓冲区中当前位置后的所有字符复制到屏幕。
F4<字符>	将输入缓冲区中除当前位置到指定字符之间的所有字符外,以后的字符复制到屏幕。
F5	输入的下一个命令行只存放在输入缓冲区中,而不执行这个命令。

四、DOS 磁盘文件

1. 文件与文件名

互相有关联的信息的集合称为文件。每个文件必须有一个名字,称为文件名。

文件名一般由文件标识符和文件扩展名组成,即文件名的一般形式为:

文件标识符. 扩展名

文件扩展名又称为后缀。

文件标识符可以由 1 至 8 个 ASCII 码字符组成,这些字符可以是:

- 10 个数字符号 0~9;
- 26 个英文字母(大小写等价);
- 特殊符号: \$, #, @, -, ! 等。DOS 规定,在文件标识符中不能出现字符 <, >, \ 和空格等。如果文件标识符中的字符个数超过 8 个,则超过部分无效。

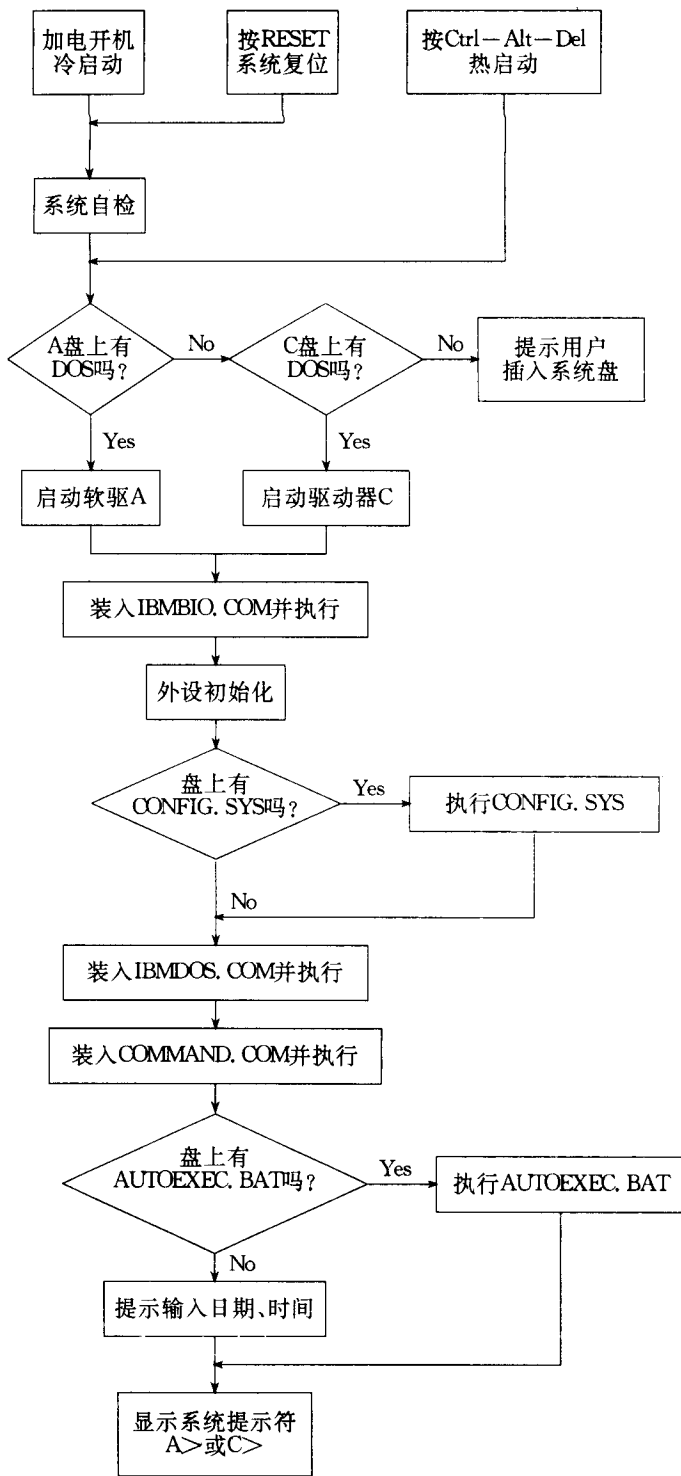


图 1.1 DOS 系统启动过程

文件扩展名必须以小数点开头,后跟 1 至 3 个字符。文件扩展名用以指出文件的类别。

一个文件的文件名可以只有文件标识符而没有文件扩展名。

DOS 系统对有些扩展名的使用有特殊规定。DOS 系统常用的扩展名有:

.COM	命令文件(可执行二进制代码文件)
.EXE	可执行程序文件
.OBJ	目标程序文件
.LIB	库文件
.SYS	系统专用文件
.BAK	备份文件
.DAT	数据文件
.BAT	批处理文件
.BAS	BASIC 源程序文件
.FOR	FORTRAN 源程序文件
.C	C 源程序文件
.PAS	PASCAL 源程序文件
.PRG	dBASE 或 FoxBASE 命令(程序)文件
.DBF	dBASE 或 FoxBASE 数据库文件
.ASM	汇编源程序文件

在文件名中,可以用文件名通配符“*”与“?”来表示一批文件。

文件名中的“*”代表它所在位置起直到符号“.”或空格前的所有字符。

文件名中的“?”代表该位置上的所有可能字符。

2. DOS 设备文件

DOS 系统定义了一些设备文件,它们分别是:

CON	控制台,包括键盘和显示器两部分。从该文件读时表示从键盘输入,向该文件写时表示输出到屏幕。
PRN	连接在第一个并行口上的打印机,该文件只能写而不能读。
LPT1	连接在第一个并行口上的打印机。
LPT2	连接在第二个并行口上的打印机。
LPT3	连接在第三个并行口上的打印机。
AUX	连接在第一个串行口上的通信设备。
COM1	连接在第一个串行口上的通信设备。
COM2	连接在第二个串行口上的通信设备。
NUL	虚设备。

最常用的设备文件是 CON 与 PRN。

3. 目录与路径

文件是存放在磁盘上的,因此,文件也叫磁盘文件。

为了实现对磁盘文件的统一管理,而且又能方便用户对文件的管理与使用,DOS 系

统采用树状目录结构实施对磁盘中所有文件的组织与管理。

DOS 有两种目录:根目录和子目录。

根目录用“\”表示。根目录是在对磁盘进行格式化时就建立。对于一个双面软盘,根目录中最多可以容纳 112 个文件,而对于一个高容量软盘,根目录中最多可以容纳 224 个文件。在根目录中,可以建立下一级的子目录。

每一个子目录必须有一个名字,子目录名的命名规则与文件标识符相同。子目录名没有扩展名。在每个子目录中,可以包含若干文件与再下一级的子目录。子目录由建立子目录命令 MD 建立。

DOS 规定,在同一个目录下,不允许有重名的文件与重名的子目录,而在不同的目录下,文件名可以重名,子目录名也可以重名。

DOS 系统当前的工作盘称为当前盘,当前工作的目录称为当前目录。在对一个文件进行操作时,如果该文件不在当前盘当前目录,则必须指出该文件所在的盘与所在的目录。文件在磁盘上的位置称为文件的“路径”。因此,一个文件名的全称表示为:

[盘符][路径]文件名

其中“盘符”指出该文件所在的磁盘,“路径”指出该文件在磁盘的哪一个目录下。

路径分为绝对路径和相对路径。

绝对路径是由“\”(表示根目录)开头,后跟若干个用“\”隔开的目录名组成,路径中最后一个目录名就是文件所在的目录。用绝对路径可以表示磁盘中任何一个目录下的文件位置。

相对路径不是以“\”开头,而是以当前目录的下一级子目录名开头。但用相对路径不能表示磁盘中所有目录下的文件,它只能表示由当前目录开始的下属各目录分枝上的文件。

在用路径表示文件在磁盘上的位置时,路径中最后一个目录名与文件名之间也要用“\”分隔。例如,A:\DOS\USER\ABC\A.C 表示磁盘 A 上的目录\DOS\USER\ABC 下的一个文件 A.C,其中所用的路径“\DOS\USER\ABC”为绝对路径。又如,USER\ABC\AB.FOR 表示当前盘上的当前目录下的子目录\USER 下的下一级子目录 ABC 下的文件 AB.FOR,其中“USER\ABC”为相对路径。

1.1.4 常用 DOS 命令的使用及常见错误分析

一、文件操作命令

1. 显示文件命令 TYPE(内部命令)

命令格式:TYPE [盘符][路径]文件名

功能:在屏幕上显示指定 ASCII 码文件的内容。

注意事项:

(1) TYPE 命令只能显示一个文件的内容,即在指定的文件名中不能出现通配符“*”和“?”。

(2) TYPE 命令只能显示 ASCII 码文件,对于非 ASCII 码文件,在屏幕上显示出来的是一些不认识的符号。