

QBASIC

语言

徐孝凯 编著

中国科学技术出版社

QBASIC 语言

徐孝凯 编著

中国科学技术出版社

• 北 京 •

图书在版编目(CIP)数据

QBASIC 语言/徐孝凯编著. —北京:中国科学技术出版社,1997
ISBN 7-5046-2402-0

I. Q… I. 徐… II. BASIC 语言—程序设计—教材 IV. TP312

中国版本图书馆 CIP 数据核字(97)第 13058 号

中国科学技术出版社出版

北京海淀区白石桥路 32 号 邮政编码:100081

新华书店北京发行所发行 各地新华书店经售

中国文联印刷厂印刷

※

开本:787 毫米×1092 毫米 1/16 印张:19.75 字数:484 千字

1997 年 11 月第 1 版 1997 年 11 月第 1 次印刷

印数:1—5000 册 定价:25.00 元

前 言

QBASIC 语言是一种简单易学、操作方便、通用性强、结构化程度高的计算机高级程序设计语言,是 DOS 5.0 以上操作系统盘中唯一携带的语言,启动机器后即可直接调入运行和使用。

一台计算机由硬件和软件两个部分组成,硬件就是看得见摸得着的所有物理设备,软件就是机器内部存放着的各种程序的总和。计算机在从开机到关机的整个运行过程中,实际上是在连续不断地、有条不紊地执行着各种系统程序和应用程序。通过学习一种计算机程序设计语言,不但能够了解计算机是如何运行的,而且能够锻炼自己的逻辑思维能力和利用计算机解决实际问题的能力,还能够为学习计算机其他软硬件课程(知识)打下良好的基础。在目前众多的计算机程序设计语言中,应当把 QBASIC 语言作为学习的首选语言。

本书是一本学习 QBASIC 语言的教材。本书在写法上力求循序渐进、条理清楚、层次分明,在内容上力求详略得当、重点突出、丰富实用,在程序设计上力求取材广泛、分析具体、描述简练,每一章后面配有适量的习题和上机题,书后并配有大部分习题的参考解答。本书可作为各级各类学校开设 BASIC 语言课程的更新换代教材,亦可作为计算机爱好者学习 QBASIC 语言的读本。

本书共分十二章,第一章和第二章介绍学习 QBASIC 语言所必需的计算机基础知识和结构化程序设计方法,第三章介绍 QBASIC 语言的组成,为学习以后各章作准备,第四章介绍 QBASIC 语言的上机操作环境,为以后上机输入、编辑和运行程序作准备,第五章介绍 QBASIC 语言的各种数据类型和各种数据表示,为以后学习各种语句和函数作准备,第六章至第八章分别介绍基本输入输出语句、分支语句和循环语句,并进行相应的程序设计,第九章介绍数组的概念及其使用,第十章介绍过程的概念及其调用,第十一章介绍绘图的概念及各种绘图语句,第十二章介绍数据文件的概念及各种文件的使用。

本书不足之处,殷切希望计算机专业的教师和广大读者予以批评指正,本人将十分感激。
联系地址:北京复兴门内大街 160 号 206 信箱,邮政编码:100031。

编 者

1997 年 1 月

目 录

第一章 计算机基础知识	(1)
§ 1-1 计算机中的数制系统	(1)
§ 1-2 计算机中的编码系统	(6)
§ 1-3 计算机硬件系统	(8)
§ 1-4 计算机软件系统	(17)
习题	(28)
上机题	(31)
第二章 结构化程序设计方法	(32)
§ 2-1 用流程图表示三种基本结构	(32)
§ 2-2 用流程图表示算法	(34)
习题	(43)
第三章 QBASIC 语言概述	(45)
§ 3-1 QBASIC 字符	(45)
§ 3-2 QBASIC 单词	(47)
§ 3-3 QBASIC 语句	(49)
§ 3-4 QBASIC 程序	(51)
习题	(53)
第四章 QBASIC 操作环境	(55)
§ 4-1 QBASIC 屏幕的组成	(55)
§ 4-2 QBASIC 菜单	(63)
习题	(73)
上机题	(75)
第五章 QBASIC 数据	(77)
§ 5-1 数据类型	(77)
§ 5-2 常量	(80)
§ 5-3 变量	(83)
§ 5-4 函数	(87)
§ 5-5 表达式	(93)
习题	(99)
上机题	(102)
第六章 简单语句	(104)
§ 6-1 LET 语句	(104)
§ 6-2 PRINT 语句	(106)
§ 6-3 INPUT 语句	(112)
§ 6-4 READ/DATA 语句	(116)
§ 6-5 其他语句	(120)
习题	(123)
上机题	(124)
第七章 分支	(127)
§ 7-1 IF 语句	(127)

§ 7-2 SELECT CASE 语句	(136)
习题	(142)
上机题	(144)
第八章 循环	(147)
§ 8-1 FOR 循环语句	(147)
§ 8-2 WHILE 循环语句	(157)
§ 8-3 DO 循环语句	(162)
习题	(168)
上机题	(170)
第九章 数组	(172)
§ 9-1 数组的概念	(172)
§ 9-2 数组说明语句	(172)
§ 9-3 数组应用举例	(179)
习题	(191)
上机题	(195)
第十章 过程	(196)
§ 10-1 过程的概念	(196)
§ 10-2 过程的定义	(199)
§ 10-3 变量的作用域	(204)
§ 10-4 过程的调用	(209)
§ 10-5 应用举例	(215)
习题	(222)
上机题	(225)
第十一章 绘图	(226)
§ 11-1 绘图的概念	(226)
§ 11-2 绘图语句	(229)
§ 11-3 窗口和逻辑坐标	(242)
习题	(251)
上机题	(254)
第十二章 数据文件	(255)
§ 12-1 文件的概念	(255)
§ 12-2 有关语句和函数	(256)
§ 12-3 顺序文件	(261)
§ 12-4 随机文件	(271)
§ 12-5 二进制文件	(278)
习题	(280)
上机题	(283)
附录一 QBASIC 运行错误信息表	(284)
附录二 QBASIC 操作键汇总表	(286)
附录三 部分习题参考答案	(289)

第一章 计算机基础知识

计算机,特别是微型计算机已经同人们日常工作、学习和生活密切相关,已经进入到各行各业、各个领域和普通家庭之中,日益发挥着越来越重要的作用,成为人们手中不可缺少的强有力的工具。因此,计算机基础知识已经成为现代社会中每一个人都要学习的基本常识。

计算机并不神秘,它实际上是一种数据处理机,是对数据进行输入、存储、加工、传送和输出的机器。计算机输入的是待处理的原始数据,输出的是经过加工处理的有用信息,就如同任何一台普通机器一样,输入的是原材料或半成品,输出的是经过机器加工的半成品或最终产品。计算机处理的数据包括数值、文字、图像、声音等各种形式的数据。如商品价格、存款金额、职工年龄等能够用数值大小表示的数据都属于数值数据,商品名称、存单帐号、职工姓名等能够用文字或字符表示的数据都属于文字数据,点、直线、圆、矩形等绘图元素都属于图像数据,音频、音调、音高、音长等声音元素都属于声音数据。

一台计算机由输入、输出、存储和运算设备等组成,人们通过输入设备把原始数据输入存储设备中存储起来,然后通过运算设备加工处理数据,产生出用户所需要的有用信息,最后再通过输出设备以适当的数据形式输出出来,供人们使用。

§ 1-1 计算机中的数制系统

计算机中的一切信息,包括命令、程序和数据都是采用二进制代码(即 0、1 序列)表示的。二进制表示,其位数较多,不易书写、阅读和交流,但能够很方便地转换成八进制或十六进制的形式,从而大大地压缩代码长度。二进制、八进制和十六进制同日常采用的十进制既有区别又有联系,它们之间都能够相互转换。

一、不同进位记数制

1. 十进制

日常采用的记数制是十进制,十进制的特点是采用 0,1,2,3,4,5,6,7,8,9 共 10 个记数符号和逢 10 进位的规则。在一个十进制数中,从小数点向左每一位的权值依次为 $10^0, 10^1, 10^2, \dots$, 从小数点向右每一位的权值依次为 $10^{-1}, 10^{-2}, 10^{-3}, \dots$ 。10 称为十进制数的基数,十进制数中每一位的权值就等于基数 10 的相应次幂。一个十进制数的大小等于每一位数字与其权值乘积的累加和。例如,对于十进制数 3047.62 来说,该数的大小,或称作该数的按权展开式为 $3 \times 10^3 + 0 \times 10^2 + 4 \times 10^1 + 7 \times 10^0 + 6 \times 10^{-1} + 2 \times 10^{-2}$ 。

2. 二进制

二进制是计算机系统中使用的记数制,二进制的特点是采用 0 和 1 这两个记数符号及逢 2 进位的规则。在一个二进制数中,从小数点向左每一位的权值依次为 $2^0, 2^1, 2^2, \dots$, 从小数点向右每一位的权值依次为 $2^{-1}, 2^{-2}, 2^{-3}, \dots$ 。2 称为二进制数的基数,二进制数中每一位的权值就等于基数 2 的相应次幂。同十进制数一样,一个二进制数的大小等于每一位数字与其权值乘积的累加和。例如,对于二进制数 1101011.101 来说,该数的大小,即在十进制下的按权展开式为 $1 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$,若按十进制数计算出结果则为 $64 + 32 + 0 + 8 + 0 + 2 + 1 + 0.5 + 0 + 0.125 = 107.625$ 。

由于二进制记数系统只需要使用 0 和 1 这两种数字符号,所以很容易用同一事物的两种不同状态来实现。如可以用电位的高和低、磁性区域的正向磁化和反向磁化、光线的穿透和遮蔽等表示出来。在计算机系统中,正是利用这些事物的两种截然不同的状态来表示二进制数字 0 和 1 的。

3. 八进制

八进制的特点是采用 0,1,2,3,4,5,6,7 共 8 个记数符号和逢 8 进位的规则。八进制通常用来表示对应的二进制数,以减少编码长度。在一个八进制数中,从小数点向左每一位的权值依次为 $8^0, 8^1, 8^2, \dots$, 从小数点向右每一位的权值依次为 $8^{-1}, 8^{-2}, 8^{-3}, \dots$ 。8 称为八进制数的基数,八进制数中每一位的权值等于基数 8 的相应次幂。同样,一个八进制数的大小等于每一位数字与其权值乘积的累加和。例如,对于一个八进制数 2716.34 来说,该数的大小,即在十进制下的按权展开式为 $2 \times 8^3 + 7 \times 8^2 + 1 \times 8^1 + 6 \times 8^0 + 3 \times 8^{-1} + 4 \times 8^{-2}$,计算结果为 $1024 + 448 + 8 + 6 + 0.375 + 0.0625 = 1486.4375$ 。

4. 十六进制

十六进制的特点是采用 0~9、A~F(大、小写均可)共 16 个记数符号和逢 16 进位的规则,其中 A,B,C,D,E,F 分别对应十进制数的 10,11,12,13,14 和 15。十六进制通常也是用来表示对应的二进制数,使得数据的位数比利用八进制表示时更少。在一个十六进制数中,从小数点向左每一位的权值依次为 $16^0, 16^1, 16^2, \dots$, 从小数点向右每一位的权值依次为 $16^{-1}, 16^{-2}, 16^{-3}, \dots$ 。16 称为十六进制数的基数,十六进制数中每一位的权值等于基数 16 的相应次幂。同十进制、二进制、八进制一样,一个十六进制数的大小等于每一位数字与其权值乘积的累加和。例如,对于一个十六进制数 3EC.8 来说,该数的大小,即在十进制下的按权展开式为 $3 \times 16^2 + 14 \times 16^1 + 12 \times 16^0 + 8 \times 16^{-1}$,计算结果为 $768 + 224 + 12 + 0.5 = 1004.5$ 。

表 1-1 给出了十进制整数 0 至 20 所对应的二进制、八进制和十六进制数。

表 1-1

十进制	二进制	八进制	十六进制
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6

7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14

二、不同数制之间的转换

1. 二进制与十进制之间的转换

计算机是采用二进制代码(即数字 0 和 1)序列来存储一切信息的,所以数值的存储、计算和传送都是在二进制环境下进行的,但为了符合人们使用十进制数的习惯,从输出设备输出的运算结果必须是采用十进制形式的,这就需要在输出过程中把二进制数转换为十进制数;同理,从输入设备向计算机输入的数值通常是采用十进制形式的,输入计算机后需要把它转换为二进制数后存储起来。当然,这些转换工作是计算机内部按照一定的算法自动完成的,无需用户过问。如果进行笔算实现二进制数向十进制数的转换,则采用按权展开求和法。例如:

$$\begin{aligned}
 (10110.11)_2 &= 1 \times 2^4 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^{-1} + 1 \times 2^{-2} \\
 &= 16 + 4 + 2 + 0.5 + 0.25 \\
 &= 22.75
 \end{aligned}$$

$$\begin{aligned}
 (11011100)_2 &= 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 \\
 &= 128 + 64 + 16 + 8 + 4 \\
 &= 220
 \end{aligned}$$

由此可知,二进制数 10110.11 对应的十进制数为 22.75,二进制整数 11011100 对应的十进制整数为 220。

十进制整数转换成二进制整数采用逐次除 2 取余法。具体叙述为:第一次用被转换的十进制整数除以 2 后所得余数是对应二进制整数的最低位,即权为 2^0 位,第二次用第一次的整数商除以 2 后所得余数是对应二进制整数的次最低位,即权为 2^1 位,依次类推,直到最后一次商为 0 时止,其所得余数是对应二进制整数的最高位,假定共得到 n 个二进制整数位,则最高位的权为 2^{n-1} 。

2	75	余数	对应二进制位
2	37	1	b_0
2	18	1	b_1
2	9	0	b_2
2	4	1	b_3
2	2	0	b_4
2	1	0	b_5
	0	1	b_6

图 1-1

例如,若把十进制整数 75 转换成对应的二进制整数,则转换过程如图 1-1 所示,其中二进制位 $b_0, b_1, b_2, \dots$ 表示权值依次为 $2^0, 2^1, 2^2, \dots$ 的数字位。

由此可知,十进制整数 75 经转换成二进制数后为 1001011。

把十进制纯小数转换为二进制纯小数采用逐次乘 2 取整法,直到积的小数部分为 0 或达到需要保留的二进制小数位数为止。具体过程可叙述为:第一次用被转换的十进制纯小数乘以 2 后所得乘积的整数部分(即为 0 或 1)是对应二进制纯小数的最高位,即权值为 2^{-1} 位,第二次用第一次乘积的

小数部分乘以 2 后所得乘积的整数部分是对应二进制纯小数的次最高位,即权值为 2^{-2} 位,依次类推,直到所得乘积的小数部分为 0 或达到需要保留的二进制小数位数为止。例如,若分别把十进制纯小数 0.75 和 0.36 转换为对应的二进制纯小数,并要求保留 6 位小数,则转换过程如图 1-2(a)和图 1-2(b)所示,其中二进制位 $b_{-1}, b_{-2}, b_{-3}, \dots$ 表示权值依次为 $2^{-1}, 2^{-2}, 2^{-3}, \dots$ 的数字位。

0.75	二进制位	0.36	二进制位
$\times 2$		$\times 2$	
1.50	$b_{-1}=1$	0.72	$b_{-1}=0$
$\times 2$		$\times 2$	
1.00	$b_{-2}=1$	1.44	$b_{-2}=1$
		$\times 2$	
		0.88	$b_{-3}=0$
		$\times 2$	
		1.76	$b_{-4}=1$
		$\times 2$	
		1.52	$b_{-5}=1$
		$\times 2$	
		1.04	$b_{-6}=1$

图 1-2

由图 1-2 可知,十进制纯小数 0.75 和 0.36 分别转换成二进制纯小数并取 6 位小数时,分别为 0.110000 和 0.010111,其中 0.75 被转换后得到的二进制数是精确的结果,而 0.36 被转换后得到的二进制数是近似值,此误差小于最后一位的权值,即小于 2^{-6} ,用分数表示为 $\frac{1}{64}$ 。

若一个十进制数为混合小数,则需将整数部分和小数部分分别转换成对应的二进制数,然后把它们合并起来。例如,若把十进制数 54.3 转换为二进制,则整数部分 54 转换成对应的二进制整数为 110110,小数部分 0.3 转换成对应的二进制纯小数为 0.010011(假定保留 6 位小数),所以十进制数 54.3 对应的二进制数为 110110.010011(保留 6 位小数)。

2. 二进制与八进制和十六进制之间的转换,

二进制的基数为 2,八进制和十六进制的基数分别为 8(即 2^3)和 16(即 2^4),也就是说,八进制和十六进制的基数分别为二进制基数的 3 次方和 4 次方,所以二进制的 3 位对应八进制

的1位,二进制的4位对应十六进制的1位。二进制数转换为八进制数或十六进制数的规则是:从二进制数的小数点向左每3位一组(对于转换为八进制数)或4位一组(对于转换为十六进制数)进行划分,若最高一组不足3位或4位则高位补0,从小数点向右每3位一组(对于转换为八进制数)或4位一组(对于转换为十六进制数)进行划分,若最低一组不足3位或4位则低位补0,然后按组从高到低依次写出对应的八进制数字或十六进制数字即可。

二进制数转换为八进制数举例:

$$(11100110)_2 = (011, 100, 110) = (346)_8$$

$$(101111.10101)_2 = (101, 111.101, 010) = (57.52)_8$$

$$(0.11000110)_2 = (0.110, 001, 100) = (0.614)_8$$

二进制数转换为十六进制数举例

$$(11100110)_2 = (1110, 0110) = (E6)_{16}$$

$$(1011010.011)_2 = (0101, 1010.0110) = (5A.6)_{16}$$

$$(1111111100001000)_2 = (1111, 1111, 0000, 1000) = (FF08)_{16}$$

因为二进制中的3位和4位分别对应八进制和十六进制中的一位,所以把二进制编码(即0、1序列)用八进制或十六进制表示后,可以大大压缩编码的长度。例如,计算机存储器中一个存储单元的地址通常为16位、24位或32位的二进制编码,若把它们表示成八进制编码,则只需要6位、8位和11位,若把它们表示成十六进制编码,则只需要4位、6位和8位。

八进制数或十六进制数转换为二进制数也很简单,只要按位写出对应的3位二进制数或4位二进制数即可。

八进制数转换为二进制数举例:

$$(352)_8 = (011, 101, 010) = (11101010)_2$$

$$(174052)_8 = (001, 111, 100, 000, 101, 010) = (1111100000101010)_2$$

$$(67.43)_8 = (110, 111, 100, 011) = (110111.100011)_2$$

十六进制数转换为二进制数举例:

$$(B7)_{16} = (1011, 0111) = (10110111)_2$$

$$(4D9A)_{16} = (0100, 1101, 1001, 1010) = (100110110011010)_2$$

$$(84.F)_{16} = (1000, 0100.1111) = (10000100.1111)_2$$

3. 八进制和十六进制同十进制之间的转换

八进制数或十六进制数转换成十进制数采用按权展开求和法。

八进制数转换为十进制数举例:

$$(236)_8 = 2 \times 8^2 + 3 \times 8^1 + 6 \times 8^0 = 158$$

$$(2000)_8 = 2 \times 8^3 = 2 \times 512 = 1024$$

$$(42.7)_8 = 4 \times 8 + 2 + 7/8 = 34.875$$

十六进制数转换为十进制数举例:

$$(8F)_{16} = 8 \times 16^1 + 15 \times 16^0 = 143$$

$$(1000)_{16} = 1 \times 16^3 = 4096$$

$$(280.A)_{16} = 2 \times 16^2 + 8 \times 16^1 + 10 \times 16^{-1} = 640.625$$

十进制整数转换为八进制整数或十六进制整数采用逐次除8或除16取余法,其运算过程与十进制整数转换为二进制整数的过程相似,十进制纯小数转换为八进制纯小数或十六进制

纯小数采用逐次乘 8 或乘 16 取整法,其运算过程也与十进制纯小数转换为二进制纯小数的过程相似。例如,若把十进制整数 180 转换为八进制整数和十六进制整数,则转换过程如图 1-3 (a)和图 1-3(b)所示,得到的对应八进制整数和十六进制整数分别为 $(264)_8$ 和 $(B4)_{16}$ 。

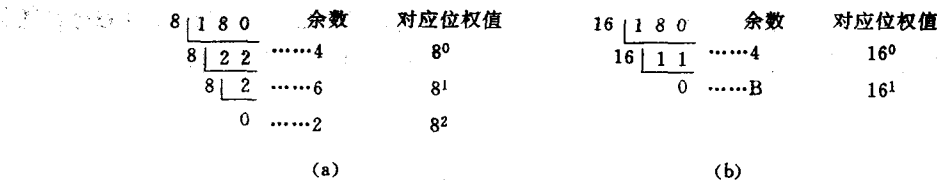


图 1-3

§ 1-2 计算机中的编码系统

在计算机中,不仅数值采用二进制表示,其他一切信息都是以二进制编码的形式表示和存储的。二进制的一位具有 0 和 1 两种编码,可表示事物的两种状态,即用 0 表示一种,用 1 表示另一种。二进制的两位具有 4(即 2^2)种不同的编码,即 00,01,10,11,可表示 4 种不同的状态。二进制的三位具有 8(即 2^3)种不同的编码,即 000,001,010,011,100,101,110,111,可表示 8 种不同的状态。总之,二进制的 n 位具有 2^n 种不同的编码,可最多表示事物的 2^n 种不同的状态。例如,可利用二进制的 5 位,产生 32 种不同的编码,用来表示 32 个不同的字符,或 32 种不同的颜色,或 32 项不同的操作,或 32 个存储单元的地址等。二进制编码的具体含义由软件开发定义。

计算机中的编码丰富多采,下面仅讨论被广泛使用的两种字符编码:ASCII 码和汉字区位码。

一、ASCII 码

在计算机中最常使用的字符编码是 ASCII 码,即美国信息交换用标准代码(American Standard Code for Information Interchange 的字头缩写),它采用二进制的 7 位,对 128 个字符进行了编码,编码范围对应二进制数为 0 至 1111111,对应十进制数为 0 至 127。在 ASCII 码中,十进制值为 0~31 和值为 127 的编码是控制字符编码,每个控制字符代表了每一种控制功能,如控制显示屏光标换行,控制扬声器鸣叫等,其余编码为可显示字符编码,包括大、小写英文字母,数字符号,标点符号,运算符号等。表 1-2 为 ASCII 代码表。

从 ASCII 代码表可以看出,十进制数字符号 0~9,大写英文字母 A~Z,小写英文字母 a~z 都是分别连续编码的,并且数字符号的编码小于大写英文字母的编码,而大写英文字母的编码又小于小写英文字母的编码。

按照 ASCII 代码表对字符的大小排序,可以用来确定两个字符串的大小。对于两个字符串,若长度(即字符个数)相等且对应的每个字符都相同,则认为它们相等;若长度不等且对应的每个字符都相同,则认为较长的字符串大于较短的字符串;若从某个字符起对应不同,则认为字符的 ASCII 码大的那个字符串较大。例如,对于 3 个字符串 ABC、XAB1 和 ABD,则它们按 ASCII 码的排列次序(即从小到大)为 ABC、ABD、XAB1。

表 1-2 ASCII 代码表

ASCII 码	控制字符	ASCII 码	字符	ASCII 码	字符	ASCII 码	字符
0	NUL	32	(Space)	64	@	96	`
1	SOH	33	!	65	A	97	a
2	STX	34	"	66	B	98	b
3	ETX	35	#	67	C	99	c
4	EOT	36	\$	68	D	100	d
5	ENQ	37	%	69	E	101	e
6	ACK	38	&	70	F	102	f
7	BEL	39	'	71	G	103	g
8	BS	40	(	72	H	104	h
9	HT	41	)	73	I	105	i
10	LF	42	*	74	J	106	j
11	VT	43	+	75	K	107	k
12	FF	44	,	76	L	108	l
13	CR	45	-	77	M	109	m
14	SO	46	.	78	N	110	n
15	SI	47	/	79	O	111	o
16	DLE	48	0	80	P	112	p
17	DC1	49	1	81	Q	113	q
18	DC2	50	2	82	R	114	r
19	DC3	51	3	83	S	115	s
20	DC4	52	4	84	T	116	t
21	NAK	53	5	85	U	117	u
22	SYN	54	6	86	V	118	v
23	ETB	55	7	87	W	119	w
24	CAN	56	8	88	X	120	x
25	EM	57	9	89	Y	121	y
26	SUB	58	:	90	Z	122	z
27	ESC	59	;	91	[	123	{
28	FS	60	<	92	\	124	
29	GS	61	=	93	]	125	}
30	RS	62	>	94	^	126	~
31	US	63	?	95	_	127	DEL

从 ASCII 代码表中可以根据任一 ASCII 码查找出对应的字符,也可以根据任一字符查找出对应的 ASCII 码。例如,根据 ASCII 码 53,可查找出对应的字符“5”;根据字符“A”,可查找出对应的 ASCII 码 65。

在计算机内部,存储每个 ASCII 码字符占用一个字节,每个字节含有二进制的 8 位,实际上只需要占用一个字节的低 7 位,空闲的最高位用 0 填补,因为高位补 0 不会改变编码的大小。例如,对于字符“R”,它的 ASCII 码为 82,对应存储字节中的内容为 01010010,用 16 进制表示则为 52H,其中后缀 H 为十六进制数的标记。

二、汉字区位码

计算机不仅能够处理 ASCII 码字符,而且还能够处理汉字。为了实现对汉字进行统一编

码,我国于 1981 年公布了汉字编码国家标准——信息交换用汉字编码字符集基本集。称为 GB2312-80 汉字编码方案,或者称为汉字区位码表。该表中共收录了 6763 个常用汉字和 682 个常用符号,包括各种图形符号、数学符号、西文字符等。对于 6763 个常用汉字又划分为两级,其中把最常用的 3755 个汉字划为一级,把剩余的 3008 个汉字划为二级。对该表中的所有字符共划分为 87 个区,每个区最多 94 个字符,分布情况如下:

1~9 区	常用符号
10~15 区	空闲未用
16~55 区	一级汉字,按拼音顺序排列
56~87 区	二级汉字,按偏旁部首排列

在汉字区位码表中,每个字符对应一个区位码,每个区位码由 4 位十进制数字组成,其中前两位数字为字符所属的区号(即区码),取值范围是 01~87;后两位数字为字符所在区内的位置号(即位码),取值范围是 01~94。例如,区位码为 0283 的字符是“■”,其中区码是 02,位码(即字符“■”在 02 区中的序号)是 83;又如区位码为 1601 的字符是“啊”,其中区码是 16,位码是 01。

计算机采用两个字节存储区位码表中的每个字符,其中第一个字节存放区码,第二个字节存放位码。区位码表中每个字符的区码或位码,其最大值不超过 94,转换为一个字节的 8 位二进制编码后,它的最高位必然为 0,为了使存储区位码字节的内容同存储 ASCII 码字节的内容区别开来,在存储区位码表中的每个字符的区码和位码时,都分别加上了十进制数 160,其对应的二进制数为 10100000,十六进制数为 A0H。在汉字区码和位码的基础上分别增加 160 后得到的十六进制编码称为汉字机内码。例如,“啊”的区位码为 1601,区码和位码对应的十六进制编码为 10H 和 01H,分别加上 A0H 后为 B0H 和 A1H,所以“啊”字符的机内码为 B0A1H,也就是说,所存的第一个字节的二进制内容为 10110000,第二个字节的二进制内容为 10100001。由于汉字机内码中的每个字节的最高位必然为 1,而 ASCII 码中的每个字节的最高位必然为 0,于是,由每个字节的最高位就能够分辨出对应的是区位码字符,还是 ASCII 码字符。

§ 1-3 计算机硬件系统

计算机包括硬件和软件两个系统,硬件就是构成计算机的所有硬设备,软件就是记忆在计算机存储设备中的所有信息。就如同一个人一样,硬件相当于人的躯体,软件相当于记忆在人头脑里的全部知识。

在计算机大家族中,使用最广泛的是微型计算机,一台微型计算机的硬件系统由主机、键盘、鼠标器、显示器和打印机等组成。主机是一个立式或卧式的箱子,箱子内部装有主机板、软磁盘驱动器、硬磁盘驱动器、光盘驱动器等。主机板上插有一个微处理器(MPU)芯片,插有若干个内存储器芯片,带有若干个扩展槽,在有的扩展槽上已经插上了接口卡。微处理器通过主机板上的地址总线、数据总线和控制总线直接同内存储器以及扩展槽上的接口卡相连接,进行信息交换,而接口卡通过电缆线同键盘、鼠标器、显示器、打印机、磁盘驱动器等外部设备相连接。

一、微处理器

微处理器又称为中央处理单元(CPU),它是整个计算机系统的控制和运算中心,它能够存取和执行内存储器中的指令,并能够按照指令的要求对内存储器中的数据进行存取、传送和运算,对外部设备(包括输入输出设备和外存设备等)进行操作控制等。目前常用的微处理器芯片型号有 80286、80386、80486 和 80586 等,多数为美国 Intel 公司的系列产品。这里所列出的芯片,越排向后面的性能越好,速度越快,技术越先进。

二、存 储 器

存储器(即存储设备)包括内存储器和外存储器。内存储器又称主存储器,简称内存或主存。内存储器是指插在主机板上的若干个存储器芯片,每个存储器芯片同微处理器芯片一样,都是采用大规模和超大规模集成电路生产工艺制造出来的。外存储器简称外存或辅存,它是利用磁盘、磁带、光盘等存储信息的存储器。在外存上存取信息时,是通过电—磁或电—光信号转换来实现的,而在内存上存取信息时只是电信号的传送。内存的存取速度比外存的存取速度快得多,往往高出几个数量级。

内存储器又分为只读存储器(ROM)和读写存储器或称随机存储器(RAM)两种。只读存储器是只能从中读出信息,不能向它写入信息的存储器,它是通过特殊工艺制造出来的,其中的信息不受开关机的影响,能够永久保留下去,因此一般用它来存储系统初始化程序、设备驱动程序、系统数据等一成不变的信息。读写存储器 RAM 是既能够从中读出信息,又能够向它写入信息的存储器,它可以用来存储任何程序和数据。读写存储器是内存储器的主体,一般若不特别指明,所说的内存储器均指读写存储器,所说的内存容量(即内存存储空间的大小)均指读写存储器的容量。

内存储器与微处理器同在主机板上,通过外部总线直接进行信息传送。微处理器运行的程序和运算的数据都直接取自内存储器,运算的结果也直接写入内存储器。微处理器若要运行外存中的程序和使用外存中的数据,都得首先把它们从外存读入内存,然后才能运行和使用。

内存储器 RAM 是集成电路器件,它保存的信息是靠电信号来维持的,一旦关机或重新启动机器(包括冷启动和热启动,冷启动是指打开主机电源开关的启动,热启动是指在机器运行过程中,通过同时按下〈Ctrl〉、〈Alt〉和〈Del〉这 3 个键使机器重新初始化的启动)后,所有信息就荡然无存了,而外存储器保存的信息能够永久持续下去,不会自然消失,当需要时可随时读出,所以对于在内存中建立的、或已经被修改的程序和数据,要及时地利用存盘命令把它们存入到外存中保存起来,以便重新使用。

内存储器由许多存储单元所组成,每个存储单元为一个字节,每个字节包含有 8 个二进制位,可用来存储不多于 8 位的二进制编码,当然若多于 8 位,则可采用多字节存储。内存储器中的每个存储单元都对应一个编号,该编号叫做该存储单元的地址。微处理器就是按照存储单元的地址来访问内存的,每次向内存中写入数据或从内存中读出数据时,都要首先发出地址,然后再发出读写控制信号,最后再把数据写入指定地址的单元或从指定地址的单元读出数据。当向一个存储单元写入新的信息后,该单元原有信息就不存在了,新的信息将一直保存下去,直到再一次被改写为止,但是,当从一个存储单元读出信息后,该单元所保存的信息仍然存在,并

且不管重复进行多少次读出都是如此,所以说,存储器(包括内存和外存)具有非破坏性读出的特点。

存储器的存储容量是指存储空间的大小,计算单位有字节(byte,缩写为 B)、千字节(kilo-byte,缩写为 KB)、兆字节(megabyte,缩写为 MB)、千兆字节(gigabyte,缩写为 GB)等,其中 1KB 等于 2^{10} 即 1024(约 10^3) 字节,1MB 等于 2^{10} KB,也等于 2^{20} (约 10^6) 字节,1GB 等于 2^{10} MB,也等于 2^{30} (约 10^9) 字节。一个微机系统的内存容量可以根据需要配置,通常配置为 1 兆字节至几十兆字节之间。当然,内存容量越大越好,因为越大越有利于提高各种软件,特别是大型软件的运行速度和能力。

三、磁盘和磁盘驱动器

1. 磁盘

磁盘是表面涂有一层磁性材料的扁平圆盘。磁盘表面被格式化后按一定规则划分为许多小区域,每个小区域代表二进制的一位,当小区域被写信号正向磁化时就表示二进制数字 1,反向磁化时就表示二进制数字 0。根据物理学中的剩磁原理,一旦信息被写入到磁盘后,该信息(即每个小区域处于正向磁化或反向磁化的状态)就能够永久地保存下去,直到再一次被改写为止。因此,磁盘就是计算机存储信息的“永久仓库”,计算机软件都是保存在磁盘(或磁带、光盘等外存,这里以叙述磁盘代之)上的,需要运行和使用时,才从磁盘调入内存,而内存只是计算机存储信息的“临时仓库”,用来临时存储从输入设备(如键盘、鼠标器等)输入的程序和建立的文件,或需要运行的外存程序和使用的文件,内存中的信息只有被写入外存后才能永久保存下去。

磁盘分为软盘和硬盘两种。软盘就是单片磁盘,软盘的两面,即正面和反面都能够保存信息。为了保护好软盘表面,免遭破坏,软盘被封装在一个正方形的塑料套内,软盘同封装套成为不可分割的整体。软盘的大小分两种尺寸,一种为 5.25 英寸,另一种为 3.5 英寸,它们的外观分别如图 1-4 中的(a)和(b)所示。

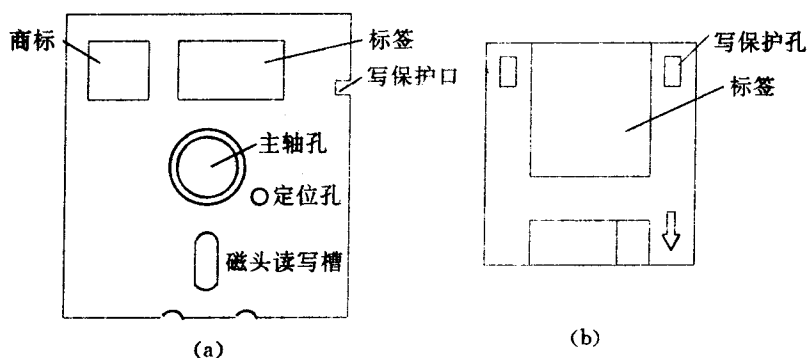


图 1-4

软盘上的标签是一张粘贴纸,当软盘存上信息后,应在空白的标签上填写与所存信息有关的标题,然后把它贴到软盘上。软盘上的写保护缺口和写保护孔是用作写保护的,当写保护缺

口被粘贴条封上或写保护孔没有被背面的滑片关闭时,则表明对该软盘上的全部信息进行了写保护,即只允许从该软盘读出信息,不允许向它写入任何信息,也就是说,具有写保护的软盘变为“只读”盘。如果你试图向写保护的软盘上写入信息,则屏幕上立即显示出“Write Protect Error writing drive X,”错误信息,表明你犯了“写保护”错误。当写保护缺口敞开或写保护孔被背面的滑片关闭时,则表明对该软盘未加写保护,系统既可以从该软盘读出信息,也可以向该软盘写入信息。

对于 5.25 英寸的软盘按存储容量分为高密和低密的两种,每张高密盘的存储容量为 1.2MB,每张低密盘的存储容量为 360KB。早期使用的 5.25 英寸的软盘多为 360KB 的低密盘,现在则多为 1.2MB 的高密盘。对于 3.5 英寸的软盘,按存储容量也分为高密和低密的两种,高密盘的存储容量为 1.44MB,低密盘的存储容量为 720KB,通常使用的 3.5 英寸的软盘多为 1.44MB 的高密盘。在一个软盘上,若带有“HD”或“2HD”标记,则为高密软盘,若带有“DD”或“2D”标记,则为低密软盘。

硬盘又叫硬盘驱动器,它是把多个磁盘片统一按一定间隔安装在一个主轴上,并能够随主轴一起高速旋转,同读写磁头一起密封在一个金属壳内的磁盘存储器。硬盘的存储容量区别很大,档次很多,从几十兆字节、几百兆字节到几千兆字节都有。目前在微机上配置的硬盘一般为几百兆字节到几千兆字节之间。另外,硬盘不像软盘那样有写保护装置,任何时候都既可以从硬盘读出信息,也可以向硬盘写入信息。

2. 磁盘驱动器

在微型计算机的主机箱内通常安装有一个或两个软盘驱动器和一个硬盘驱动器。软盘驱动器是用来驱动软盘读写的设备。每个软驱(即软盘驱动器)都有一个软盘插入槽、一个进出软盘片的开关和一个读写盘指示灯。对于 5.25 英寸的软驱来说,其进出软盘片的开关是一个旋转扳手,当此扳手与插入槽平行时则为开状态,只有在开状态下,才能插入或取出软盘片。当插入时,要使软盘片的正面(即带有商标的一面)朝向扳手,并且手拿贴有标签的一端,从另一端开始向插入槽中轻轻推入,直到完全推进去为止,当取出时,从插入槽中轻轻拉出软盘片即可。当向插入槽中插入软盘片后,要按顺时针方向旋转扳手开关,直到它与插入槽垂直为止,此时的扳手位置为关状态,这样就给驱动器关上了门。对于 3.5 英寸的软驱来说,其进出软盘片的开关是一个按钮,当插入 3.5 英寸的软盘片时,使正面背向按钮开关,并且手拿贴有标签的一端,从另一端开始向插入槽轻轻推入,直到软盘完全进入驱动器并听到咔嚓一声响为止,这表明驱动器已经关好门。从 3.5 英寸的软驱中取出软盘时,首先要把按钮开关按进去,这就打开了该驱动器的门,此时驱动器会把软盘部分地弹出,再用手拉出软盘即可。

当进行软盘操作时,只有当软驱内插入软盘并关好门后,系统才能使用该软驱,否则无法使用。若试图对未插入软盘的软驱或已插入软盘但尚未关上门的软驱进行访问时,则屏幕上显示出:

Not ready reading drive A

Abort, Retry, Fail? _

错误信息,告诉你指定的软驱(A 或者 B)没有准备好,应插入盘并关好门后再打入字母 R 则可继续运行。

当系统对软驱内的软盘正在进行读写操作时,该软驱上的指示灯就点亮,当读写结束后,指示灯就熄灭。只有当指示灯熄灭时,才允许用户打开软驱的门,即对于 5.25 英寸的软驱来