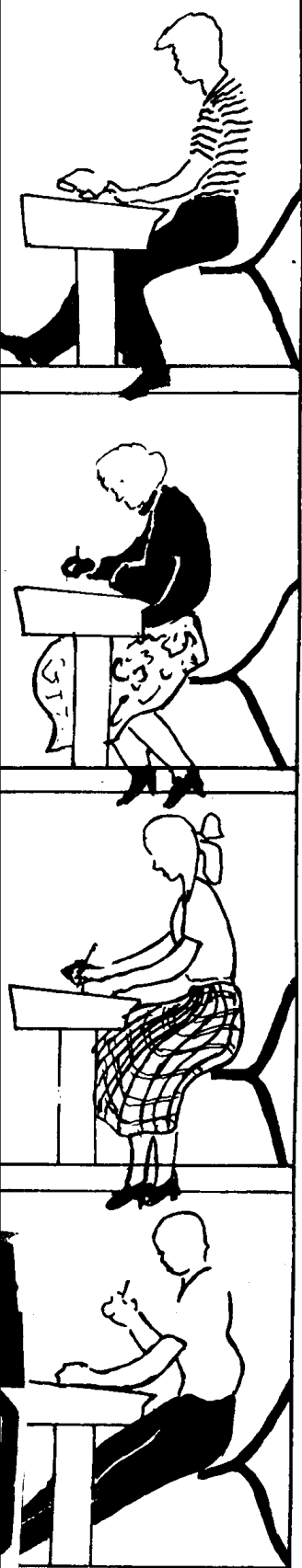


大学计算机应用知识和

应用能力等级考试指南

成都科技大学出版社



大学计算机应用知识和应用能力 等级考试指南

本书编委会

成都科技大学出版社

(川) 新登字 015 号

责任编辑: 刘志刚 刘预知

大学计算机应用知识和应用能力等级考试指南

本书编委会

成都科技大学出版社出版发行

重庆璧山印务有限公司

开本 787×1092 1/16 印张 21.75

1996 年 2 月第 1 版 1996 年 2 月第 1 次印刷

印数 1-5000 册 字数 700 千字

ISBN7-5616-3059-X / TP·134

定价: 18.80 元

目 录

上篇 应用知识

第一章 计算机基础知识.....	(1)
第一节 计算机的发展和它的特点及用途.....	(1)
第二节 计算机中数据、字符的存贮形式与表示.....	(2)
第二章 计算机系统的硬件与软件.....	(11)
第一节 微机系统的基本硬件配置及主要技术指标.....	(11)
第二节 计算机软件系统	(19)
第三章 操作系统.....	(25)
第一节 操作系统概述.....	(25)
第二节 DOS 操作系统	(26)
第三节 UNIX操作系统及使用	(35)
第四节 WINDOWS操作系统及使用.....	(42)
第四章 字表处理软件.....	(48)
第一节 计算机字表处理的含义.....	(48)
第二节 计算机汉字处理和汉字库.....	(49)
第三节 汉字输入法.....	(51)
第四节 字表处理软件的基本功能.....	(60)
第五节 字表处理软件的基本操作.....	(60)
第六节 字表处理软件的使用.....	(62)
第五章 BASIC 语言程序设计	(65)
第一节 BASIC 源程序的结构及其特点	(65)
第二节 常量、变量的运算符、表达式及标准函数.....	(66)
第三节 基本语句.....	(68)
第四节 控制转移语句和控制结构设计.....	(70)
第五节 数组.....	(72)
第六节 字符串操作.....	(74)
第七节 自定义函数与子程序.....	(75)
第八节 磁盘文件的输入和输出.....	(76)
第九节 图形处理.....	(79)
第六章 FORTRAN 语言程序设计	(82)

第一节	FORTRAN 的基本概念	(82)
第二节	基本语句	(88)
第三节	控制语句及控制结构程序设计	(90)
第四节	数组	(96)
第五节	函数和子程序	(97)
第六节	数据联系语句	(101)
第七节	字符信息的处理	(104)
第七章	PASCAL 语言程序设计	(109)
第一节	PASCAL 程序的结构	(109)
第二节	数据类型及其运算	(109)
第三节	PASCAL 语言基本语句	(115)
第四节	选择结构程序设计	(117)
第五节	循环程序设计	(118)
第六节	数组的应用	(120)
第七节	过程和函数	(122)
第八节	动态数据结构	(123)
第九节	文件操作	(125)
第八章	C 语言程序设计	(127)
第一节	C 语言基本结构	(127)
第二节	数据类型及其计算	(128)
第三节	C 语言基本语句及流程的控制	(134)
第四节	数组的定义和引用	(139)
第五节	函数	(140)
第六节	指针类型	(143)
第七节	结构与联合	(148)
第八节	文件操作	(151)
第九章	数据库管理系统 dBASE III	(153)
第一节	数据库的初步知识	(153)
第二节	汉字 dBASE III 基础	(156)
第三节	汉字 dBASE III 的基本命令和基本操作	(163)
第十章	数据库管理系统 FoxBASE +	(187)
第一节	有关数据库 FoxBASE + 的基本知识	(187)
第二节	FoxBASE + 中的常用操作命令	(188)
第三节	FoxBASE + 中的数据运算和函数简介	(191)
第四节	FoxBASE + 程序设计入门	(193)
第十一章	数据库管理系统 Foxpro	(198)

下篇 应用能力

第一部份：全国计算机等级考试大纲.....	(202)
一、一级考试大纲.....	(203)
二、二级考试大纲.....	(204)
三、三级 A 类考试大纲	(210)
三级 B 类考试大纲	(212)
第二部分：计算机等级考试题型示例及答案.....	(214)
一、一级题型示例.....	(214)
二、二级题型示例.....	(215)
三、三级 A 类题型示例	(228)
三级 B 类题型示例	(230)
四、各级题型示例答案	(234)
第三部分：考试样题.....	(238)
一、四川省第二次计算机等级考试试题.....	(238)
附：试卷答案及评分细则.....	(260)
二、四川省第一、二次计算机等级考试上机考试题.....	(267)
三、普通高校非计算机专业计算机应用知识和能力等级考试题.....	(274)
第四部分：模拟试题.....	(289)
一、计算机基础考试模拟试题.....	(289)
二、模拟试卷.....	(299)
三、上机模拟试题.....	(324)
附录：一、全国计算机等级考试管理规则	(327)
二、全国计算机等级考试考务工作手册.....	(330)
三、全国计算机等级考试考生须知.....	(338)
四、关于全国计算机等级考试报名考试费收费标准的通知.....	(339)
五、全国计算机等级考试机读报名卡填涂说明.....	(339)

第一章 计算机基础知识

第一节 计算机的发展和它的特点及用途

一、计算机的发展和它的特点

随着科学技术的发展,对计算机速度、精度等方面要求越来越高,因此,电子计算机应运而生了。电子计算机是现代科学技术发展的必然产物。

根据计算机采用的电子器件的变更来划分,可以把计算机的发展分为四个阶段,也就是计算机的四代。

第一代从1946年到1957年。这一代计算机主要特点是采用电子管的逻辑电路。体积大、耗电多、使用和维护困难。人们公认的第一台电子计算机是1946年2月在美国问世的,它的名字叫ENIAC(埃尼阿克)。

第二代从1958年至1964年,这一代计算机采用了晶体管逻辑电路。缩小了体积,使用方便,应用范围有所扩大。

第三代从1965年到1970年,它的特点是采用中小规模集成电路器件。计算机技术趋于成熟,开始系列化生产,价格明显降低,应用范围进一步扩大。

1970年以后,计算机进入了第四代,这一代计算机的主要特点是采用了大规模和超大规模集成电路技术。

电子数字计算机的特点:

- (1) 运算速度快。巨型机已达每秒十几亿次。
- (2) 精确度高。一般计算机可以有十几位有效数字
- (3) 具有“记忆”和逻辑判断的能力。计算机不仅能进行计算,而且还可以把原始数据、中间结果、计算机指令等信息存贮起来,以备调用。它还能进行各种逻辑判断,并根据判断的结果自动决定以后执行的命令。
- (4) 计算机内部的操作运算,都是自动控制进行的。

概括起来说,电子计算机是一种以高速进行操作、具有内部存贮能力、由程序控制操作过程的自动电子装置。

这四代计算机基于同一个基本原理,就是以二进制数和程序存贮控制为基础的结构思想。这个思想是由美籍数学家冯·诺依曼(Von Nouman)于1946年最早提出来的,它确立了至今为止的各代计算机的基本工作原理。

正在研制的“第五代计算机”将是一种非冯·诺依曼型的计算机。它采取完全新的工作原理和体系结构。它更接近于人们思考问题的方式,即“推理”方式。这种新型的计算机称为“知识信息处理系统”。其功能从目前单纯的数据处理发展到知识的智能处理。

二、计算机的用途

计算机有以下几方面的应用:

- 1. 科学计算,或称数值计算
- 2. 用于自动控制系统,特别是工业、交通的自动控制

3. 数据处理和信息加工
4. 计算机辅助系统
 - (1) 计算机辅助设计;
 - (2) 计算机辅助制造;
 - (3) 计算机辅助教学。
5. 人工智能 (AI) 方面的研究和应用, 利用计算机模拟人脑的一部分职能。

第二节 计算机中数据、字符的存贮形式与表示

数制本是人们利用符号来计数的一种科学方法。计数可以有多种方法, 但在计算机的设计与使用中常使用的是十进制、二进制、八进制和十六进制。

冯·诺依曼型计算机中的一切数据都是以二进制形式存贮, 读者可能会问, 为什么不用十进制数, 而要用我们不太习惯的二进制数呢? 其原因有几个: (1) 由于构成计算机的元件都是电子器件, 在技术上用电子器件实现 0-9 数码的十个状态几乎是不可能的, 而实现二进制数码的 0 和 1 两种状态却非常容易, 生活中就有这样的例子, 如灯的“亮”与“不亮”, 开关的“断开”与“闭合”, 晶体管的“截止”与“导通”等。(2) 是二进制数只有两个数, 在计算机内部正确传递与运算可靠性高。(3) 是二进制数运算规律非常简单 (见本节第六点), 这样可使计算机的结构大为简化。

读者可能还会问, 电子器件是怎样表示数的呢? 我们可以用一个十分简单的例子来说明。假如有图 1-1 的 (a), (b), (c) 三组灯, 每组分别为: 一盏灯, 二盏灯, 三盏灯。若灯“亮”用数“1”表示, 灯“灭”用数“0”表示, 请观察它们分别可表示数的范围。

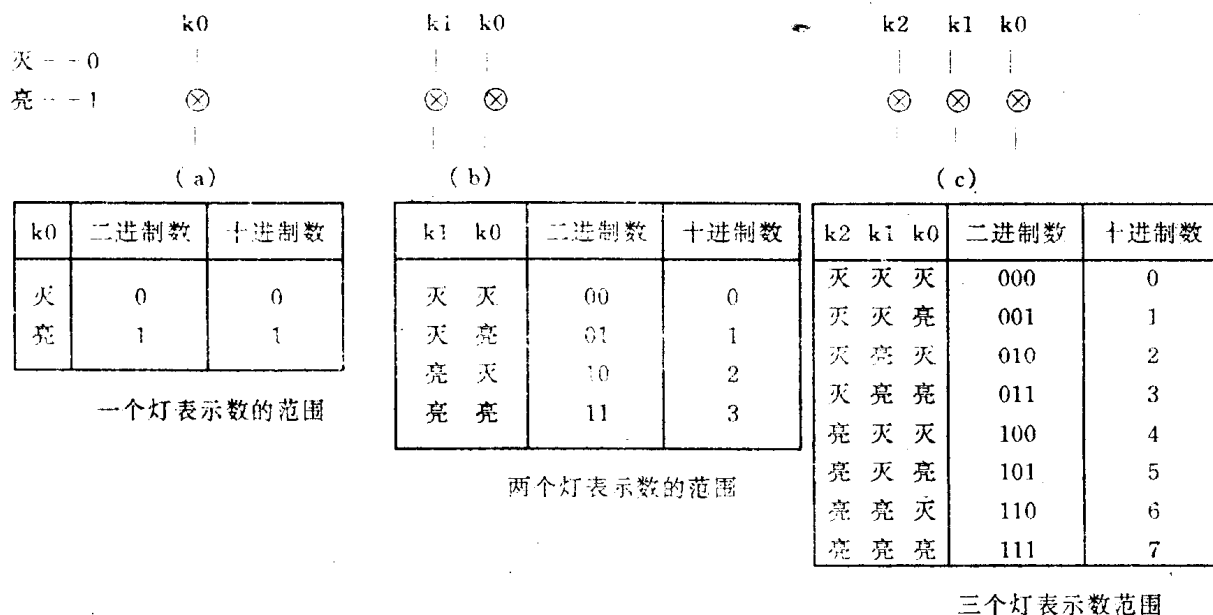


图 1-1 用电子元件表示数

若一盏灯代表一个电子器件, 那么一个电子器件就可以表示 0 和 1 两个数, 最大数是 1; 两个电子器件可以表示 0 到 3 的 4 ($=2^2$) 个数, 最大数是 3; 三个电子器件就可表示 0 到 7 范围的 8 ($=2^3$) 个数, 最大数是 7, 以此类推, 8 个电子器件可表示数的范围是 0 到 255 的 256 ($=2^8$) 个数, 最大数是 255。

计算机中存贮器就是由千千万万个这样具有双态状态的电子线路单元组成，每个单元称为一个“位”（Bit，又称“比特”），这样的“位”在计算机中有几万，几十万，甚至上亿个，位越多说明内存容量（或空间）就越大，可存放的数据就越多。

一、内存的组织形式

为便于内存的管理，象前面所说的那样，常将 8 个二进制位组成的基本存贮单位称为一个“字节（Byte）”，如 00011011，一个数据在内存中存贮时以字节为基本单位。通常一个字符占内存中一个字节空间，而一个整数占 2 个字节空间，一个实数占 4 个字节空间。

如果将内存比着一幢巨大的楼房，那么其中就有成千上万的房间，每个房间可以放下一个八位二进制数，即一个字节。为了在这许多的房间中访问所需的数据，计算机还必须给以字节为基本存贮单位的每一个单元（房间）一个以二进制数编码的地址。通过地址，计算机才能找到相应的单元，才能进行从中取数或存数的操作。通常地址总线的根数越多，存放数据的单元就越多。因为，8 根地址总线可表示的地址范围是（ $2^8 =$ ）256 个房间号，而 20 根的地址总线可表示的地址范围是（ $2^{20} =$ ） 1024×1024 个房间号。

二、数据的二进制表示方法

人类很早就习惯了十指计数的道理，这就是十进制（Dec - Decimal System）计数法。与十进制的逢十进一的进位规则相对应，二进制（Bin - Binary System）就是逢二进一。十进制中参与运算的数据是 0, 1, 2, . . . , 9 共十个数来表示，而在二进制中参与运算的数只用 0 和 1 两个数来表示。因此，在计算机中进行二进制数运算比十进制数简单得多。但应注意对非十进制中的 10 与 100 不能误读作“拾”与“壹百”，而应读作“壹零”“壹零零”。如有一串用二进制表示的一个数或代码 1011011，应读作“壹零壹壹零壹壹”。

在日常生活中，人们经常与各种进制的数打交道，如记录时间可用 12 小时表示半天就是用 12 进制数表示的、24 小时表示一天就是 24 进制的数；60 分钟表示一小时或 60 秒表示一分钟就是用 60 进制的数表示的；用二进制数表示两种状态或一双等概念。在后面的讨论中会知道，计算机中的一个数或一个代码除用二进制数表示外，还常以不同进制形式的数来表示，如八进制、十进制、十六进制。但究竟以十进制表示或以十六进制表示，还是以八进制表示，应视具体情况而定，目的就是为了使使用或阅读方便。

三、二进制数与十进制数的转换

由于人们习惯于十进制，虽然在机内运算采用二进制数，而在输入与输出时仍希望采用十进制。这样用户从显示器上，从打印机上输出数据或从键盘上输入数据时仍与十进制数打交道。因此，在主机与外设之间常常要进行十进制数和二进制数间的转换工作。

一个十进制整数化为二进制数的规则为：“除二倒序取余”。即将十进制数一次又一次地除以 2，将所得的余数从后向前排列即为二进制数。

一个十进制纯小数化为二进制数的规则是：“乘二顺序取整”。即用 2 乘十进制纯小数，取出整数部分后再用 2 乘余下的纯小数部分，又得到新的整数部分，依此类推，直到余下的纯小部分为零或达到所要求的精度为止。再把每次取出的整数部分顺序排列起来，就是与十进制等值的二进制数。

例如将十进制数 15.5625 表示为二进制形式时，可以分别将整数与纯小数部分先分别计算，然后再将其合并。具体计算过程如下：

$$(15)_{10} = (K_n \dots K_1 K_0)_2 \quad (0.5625)_{10} = (0.A_1 A_2 \dots A_m)_2$$

整数部分：除 2 取余的计算过程如下：

2 15		0. 5625	
└───		× 2	
2 7 (1 = K ₀)	↑	1. 1250	取整部分 A ₁ = 1
└───		0. 1250	
2 3 (1 = K ₁)	倒	× 2	
└───	序	0. 2500	取整部分 A ₂ = 0
2 1 (1 = K ₂)	取	× 2	顺
└───	余	0. 5000	序
0 (1 = K ₃)		× 2	取
		1. 0000	整
			↓

得到: $(15)_{\text{Dec}} = (1111)_{\text{Bin}}$ $(0.5625)_{10} = (0.1001)_2$
 $(15.5625)_{\text{Dec}} = (1111.1001)_{\text{Bin}}$

以上括号外的脚注 Dec 表示 Decimal (十进制), 也可用 10 表示, Bin 表示 Binary (二进制), 也可用 2 表示。

所以, 任意一个十进制数要化为二进制数就是将整数部分的“除 2 倒序取余”结果和纯小数部分的“乘 2 顺序取整”的结果组合起来就可以了。即表示如下:

K _n	K _{n-1}	...	K ₁	K ₀	.	A ₁	A ₂	...	A _m
----------------	------------------	-----	----------------	----------------	---	----------------	----------------	-----	----------------

反过来, 要想把一个二进制数转化为十进制数, 只需将二进制数的各位乘以相应的“权”, 然后相加即可。前面十进制数 15 可以表示成:

$$15 = 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$

即是说, 可以将任何一个十进制数 P 化成以 2 为底的指数形式来表示:

$$(P)_{10} = K_n 2^n + K_{n-1} 2^{n-1} + \dots + K_1 2^1 + K_0 2^0 + A_1 2^{-1} + A_2 2^{-2} + \dots + A_m 2^{-m}$$

(1-5-1)

其中 K_n, K_{n-1}, K₁, K₀, A₁, A₂, A_m 系数分别就是 P 在二进制中的表示形式, 2⁻ⁿ 与 2^{-m} 就表示相应位上二进制系数的“权”。如下例:

$$\begin{aligned} (110.101)_{\text{Bin}} &= 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= 4 + 2 + 0 + 0.5 + 0.125 \\ &= (6.625)_{\text{Dec}} \end{aligned}$$

将十进制中 0~9 十个数化为二进制数如表 1-1。

表 1-1

十进制数	化为以 2 为底的指数形式	二进制数	十进制数	化为以 2 为底的指数形式	二进制数
0	0×2^0	0	5	$1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$	101
1	1×2^0	1	6	$1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$	110
2	$1 \times 2^1 + 0 \times 2^0$	10	7	$1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$	111
3	$1 \times 2^1 + 1 \times 2^0$	11	8	$1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$	1000
4	$1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$	100	9	$1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$	1001

四、八进制数

八进制数 (Oct - Octave System) 的进位规则是逢八进一, 参与运算的数是 0, 1, ..., 6, 7 八个数。

由于二进制数书写起来很长、难以记忆、易出错, 加之八进制数和十六进制数与二进制数之间有一种特殊的对应关系, 这就是三位二进制数刚好可以表示从 0 到 7 之间的一个 8 进制数, 而四位二进制数则可以表示一个从 0 到 15 之间的十六进制数, 它们之间的这种对应关系给换算工作带来了极大的方便。所以, 当编写计算机的汇编程序时, 人们常用 8 进制数或 16 进制数来书写或表示机内的一个二进制数。

1、八进制数 (Oct) 与二进制数 (Bin) 间的转换

例: 要将一个二进制数 10011010. 1111 转换为八进制数, 那么先将该二进制数从小数点开始, 分别向左右每三位分成一组, 组间用逗号分隔, 即 10, 011, 010, 111, 100 五组, 每一组代表一个从 0 到 7 之间的数, 因为三位二进制数是不会等于或大于 8 的, $(111)_{\text{Bin}} = (7)_{\text{Dec}}$, 而 $(8)_{\text{Dec}} = 2^3 = (1000)_{\text{Bin}}$ 就需要 4 位二进制数来表示, 即要向前一组数进一位。即可分组为:

10,	011,	010,	111,	100	二进制数
↓	↓	↓	↓	↓	
2	3	2	7	4	八进制数

可得: $(10011010. 1111)_{\text{Bin}} = (232. 74)_{\text{Oct}}$

而将一个八进制数转换为二进制数, 只需将其每一位数分别转为二进制数即可。例如:

7	6	3	2	5	2
↓	↓	↓	↓	↓	↓
111	110	011	010	101	010

可得: $(7632. 52)_{\text{Oct}} = (111, 110, 011, 010, 101010)_{\text{Bin}}$

2、八进制数与十进制数间的转换

八进制数与十进制数间的转换方法跟十进制数与二进制数间的转换基本相同。

要将一个八进制数转换为十进制数只要将前面公式 (1-1) 中以 2 为底的指数形式换为以 8 为底的指数形式即可。即是说, 可以将任何一个十进制数 P 化成以 8 为底的指数形式来表示:

$$(P)_{10} = K_n 8^n + K_{n-1} 8^{n-1} + \dots + K_1 8^1 + K_0 8^0 + A_1 8^{-1} + A_2 8^{-2} + \dots + A_m 8^{-m} \quad (1-2)$$

其中 $K_n, K_{n-1}, K_1, K_0, A_1, A_2, A_m$ 系数分别就是 P 在八进制中的表示形式, $8^{\pm m}$ 就表示相应的“权”。如下例:

$$\begin{aligned}(207. 26)_8 &= 2 \times 8^2 + 0 \times 8^1 + 7 \times 8^0 + 2 \times 8^{-1} + 6 \times 8^{-2} \\ &= 2 \times 64 + 0 + 7 \times 1 + 2 / 8 + 6 / 64 \\ &= 128 + 7 + 0. 25 + . 0938 \\ &= (135. 344)_{10}\end{aligned}$$

反之, 如果将一个十进制整数化为八进制数, 对整数部分的转换规则是: “除 8 倒序取余”。而对纯小数部分的转换规则是: “乘 8 顺序取整”。如将十进制数 29. 6875 转为八进制数, 可如下进行:

8 29	↑	0. 6875	
└───	倒	× 8	顺
8 3 (5 = K ₀)	序	-----	序
└───	取	5. 5000 取整部分 A ₁ = 5	取
0 (3 = K ₁)	余	0. 5000	整
		× 8	

		4. 0000 取整部分 A ₂ = 4	

可得: $(29. 6875)_{10} = (35. 54)_8$ 。

五、十六进制数

十六进制 (Hex - Hexadecimal System) 的进位规则是逢十六进一, 参与运算的数是 0, 1, 2, . . . , 9, 10, . . . 15 共 16 个数。为了便于表示, 十六进制数规定 0-15 中的每个数都只能用一个符号来表示 (不能用两位十进制数来表示)。所以, 0-9 之间的数仍可用原来的符号, 而 10-15 这六个数分别以 A, B, C, D, E, F 来表示, 见表 1. 5. 2。

1、二进制数与十六进制数间的转换

将一个二进制数 10011010. 1111 转为十六进制数, 只要先将二进制数从小点开始, 分别每四位分成一组, 组间用逗号分隔, 即 1001, 1010. 1111 三组, 每一组代表一个从 0 到 F 之间的数, 因为四位二进制数是不会等于或大于 16 的, $(1111)_{\text{Bin}} = (15)_{\text{Dex}}$, 而 $(16)_{\text{Dex}} = 2^4 = (10000)_{\text{Bin}}$ 就需要 5 位二进制数来表示, 即要向前一组数进一位。即可分组为:

1001,	1010.	1111	二进制数
↓	↓	↓	
9	A	F	十六进制数

可得: $(10011010. 1111)_{\text{Bin}} = (.9A. F)_{\text{Hex}}$ 。

而将一个十六进制数转换为二进制数, 只需将其每一位数分别转为二进制数即可。例如:

A	8	E	7	C	3
↓	↓	↓	↓	↓	↓
1010	1000	1110	0111	1100	0011

可得: $(A8E7. C3)_{\text{Hex}} = (1010, 1000, 1110, 0111. 1100, 0011)_{\text{Bin}}$ 。

2、十六进制数与十进制间的转换

将一个十六进制数转换为十进制数只需将公式 (1-1) 中以 2 为底的指数形式换为以 16 为底的指数形式即可。即是说, 可以将任何一个十进制数 P 化成以 16 为底的指数形式来表示:

$$(P)_{10} = K_n 16^n + K_{n-1} 16^{n-1} + \dots + K_1 16^1 + K_0 16^0 + A_1 16^{-1} + A_2 16^{-2} + \dots + A_m 16^{-m} \quad (1-2)$$

其中 $K_n, K_{n-1}, K_1, K_0, A_1, A_2, A_m$ 系数分别就是 P 在十六进制中的表示形式, $16^n, 16^{-m}$ 就表示相应的“权”。如将 16 进制数 20AF. A6 转换为十进制数, 可如下进行:

$$\begin{aligned}
 (20AF. A6)_{\text{Hex}} &= 2 \times 16^3 + 0 \times 16^2 + A \times 16^1 + F \times 16^0 + A \times 16^{-1} + 6 \times 16^{-2} \\
 &= 2 \times 4096 + 0 + 10 \times 16 + 15 \times 1 + 10 / 16 + 6 / (16 \times 16) \\
 &= 8192 + 160 + 15 + . 625 + . 0234 \\
 &= (8367. 648)_{\text{Dex}}
 \end{aligned}$$

反过来, 如要将一个十进制数转换为十六进制数, 其整数部分转换规则是: “除 16 倒序取余”; 而小数部分的转换规则是 “乘 16 顺序取整”。如将十进制数 129. 634 转为十六进制数, 可如下进行:

$16 \overline{) 129}$
 $16 \overline{) 8} \quad (1 = K_0)$
 $0 \quad (8 = K_1)$

↑ 倒
 序
 取
 余

0.634
 $\times 16$

 10.144 取整部分 $A_1 = 10 \text{ (A)}$
 0.144
 $\times 16$

 2.304 取整部分 $A_2 = 2$
 0.304
 $\times 16$

 4.864 取整部分 $A_3 = 4$
 0.864
 16

 13.824 取整部分 $A_4 = 13 \text{ (E)}$

顺
 序
 取
 整

所以:

$(129.634)_{\text{Dec}} = (81.A24E)_{\text{Hex}}$

表 1-2

十 进 制 数	二 进 制 数	八 进 制 数	十 六 进 制 数
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

六、二进制的算术运算与逻辑运算

二进制数的运算包括两个方面的运算：一个是算术运算，与十进制数运算相同，不同的是二进制数是逢二进一；另一个是逻辑运算，这是十进制数没有的。

(一) 算术运算

二进制数的算术运算法则十分简单，包括加、减、乘、除四种运算，每种运算只有 3 条。

1、加法运算

运算法则 3 条：(1) $0 + 0 = 0$ ；(2) $0 + 1 = 1 + 0 = 1$ ；(3) $1 + 1 = 10$

例：求二进制数 10101 与 11011 的和。

被加数 10101. (21)₁₀
 加数 11011. (27)₁₀
 进位 +) 1111

和数 110000. (48)₁₀

结果: $(10101)_2 + (11011)_2 = (110000)_2$

2、减法运算

运算法则 3 条: (1)0-0=1-1=0; (2)1-0=1; (3)0-1=1

例: 求二进制数 11001 与 10110 之差。

被减数 11011. (27)₁₀
 减数 10110. (22)₁₀
 借位 -) 1111

和数 00101. (5)₁₀

结果: $(11011)_2 - (10110)_2 = (00101)_2$

3、乘法运算

运算法则 3 条: (1)0×0=0; (2)0×1=1×0=0; (3)1×1=1

例: 求二进制数 1110 与 1101 之积。

被乘数 1110. (14)₁₀
 乘数 ×) 1101. (13)₁₀

1110	部
0000	分
1110	积
1110	

积: 10110110. (182)₁₀

结果: $(1110)_2 \times (1101)_2 = (10110110)_2$

4、除法运算

运算法则 3 条: (1)0÷0=0; (2)0÷1=0 (1÷0 无意义); (3)1÷1=1

例: 二进制数 100110 与 110 相除。

	000110.	商数
除数. . . 110	) 100110.	被除数
	110.	部分积
	0111.	中间被除数
	110.	部分积
	10.	余数

结果: $(100110)_2 \div (110)_2 = (110)_2$ 余数 = $(10)_2$

(二) 二进制的逻辑运算

1、逻辑运算概念: 为了方便地将复杂的逻辑思维与逻辑推理问题用数学模型来处理, 产生了“布尔代数 (Boole algebra)”。布尔代数在电子电路中的应用与扩展, 又产生了“开关代数”。而布尔代数与开关代数又被统称为逻辑代数。逻辑代数研究的是非数值逻辑变量 (其值只有“真”与“伪”之分) 的运算问题。在正逻辑中用“1”代表“真”, 用“0”代表“伪”; 负逻辑表示刚好相反, 此处采用正逻辑。逻辑变量没有大小与正负之分。建立了

逻辑代数后，不论多复杂的逻辑问题都可以用“逻辑非”、“逻辑或”与“逻辑与”三种基本的逻辑运算来描述。

2、逻辑非运算：是对一个逻辑变量 X 进行取反运算，其布尔代数表达式为：

$$F = \bar{X}$$

$\bar{X}$ 读作“ X 非”，其上的横线表示逻辑非，即求反的运算符。若 X 称原变量，那么 $\bar{X}$ 是 X 的反变量，即若 X 的值为“假”，则 $\bar{X}$ 的值就为“真”，相反亦然。所以：

若 $X = 0$ ，则 $\bar{X} = 1$ ，若 $X = 1$ ，则 $\bar{X} = 0$

并且： $\bar{\bar{X}} = X$ ， $X + \bar{X} = 1$ ， $X \cdot \bar{X} = 0$

3、逻辑或运算：是对两个逻辑变量 X ， Y 进行逻辑加的运算。其表达式为：

$$F = X + Y$$

“ $X + Y$ ”读作“ X 加 Y ”或“ X 或 Y ”，“+”不是算术运算中的“加”，而是逻辑运算中的或运算符。逻辑或运算遵守如下的运算规则：

$$0 + 0 = 0 \quad 0 + 1 = 1$$

$$1 + 0 = 1 \quad 1 + 1 = 1$$

以上规则说明，两个同为“真”，或一个为“真”，另一个为“假”的变量进行“或”运算，其结果为“真”；而两个为“假”的变量进行“或”运算时，结果为“假”。所以，当一个逻辑变量与逻辑常量“1”进行“或”运算，其结果一定是“1”，即“真”；而逻辑变量与“0”进行“或”运算时，其结果是逻辑变量本身，即：

$$X + 1 = 1 \quad X + 0 = X \quad \bar{X} + X = 1$$

$$\bar{X} + 1 = 1 \quad \bar{X} + 0 = \bar{X} \quad X + X = X$$

4、逻辑与运算：是对两个逻辑变量进行逻辑乘的运算。其表达式为：

$$F = X \times Y (=X \cdot Y = XY)$$

“ $X \times Y$ ”读作“ X 乘 Y ”或“ X 与 Y ”，“ $\times$ ”不是算术运算中的“乘”，而是逻辑运算中的与运算符。其运算规则为：

$$0 \times 0 = 0 \quad 1 \times 0 = 0$$

$$0 \times 1 = 0 \quad 1 \times 1 = 1$$

以上规则说明，两个同为“真”的逻辑变量进行“与”运算，其结果为“1”；而两个同为“假”或一个为“真”，另一个为“假”的变量进行“与”运算，其结果为“0”，即“假”。显然以下也是成立的：

$$Y \times 0 = 0 \quad Y \times 1 = Y \quad Y \times Y = Y$$

5、多位二进制的逻辑运算：对于多位的二进制运算，以上规则同样适用，只是按位进行逻辑运算而已。如“非”运算是按位求反，若求逻辑变量 $A = 10011$ 的非，则 $\bar{A} = 01100$ ；“或”运算是按位进行逻辑加，若 $X1 = 1001$ 、 $X2 = 1010$ ，则 4 位二进制数的或运算为：

$$X1 = 1001$$

$$X2 = 1010$$

$$X1 + X2 = 1011$$

如“与”运算是按位进行逻辑乘，若 $X1 = 1010$ 、 $X2 = 1000$ ，则 $X1$ 和 $X2$ 的与运算为：

$$X1 = 1001$$

$$X2 = 1010$$

$$X1 \times X2 = 1000$$

七、各种字符在计算机中的表示——ASCII 码

前面介绍了计算机中参与运算数据的二进制数表示。实际上，计算机除了处理数值型数据代码外，还能有效的地处理不是数值型数据的各种符号。这类符号包括 26 英文字母、数字、各种特殊字符以及在屏幕上不可显示和不可打印的能产生特殊效果的字符。为了使计算机能处理这些非数值性数据，就必须用二进制数的编码来表示它们。目前在各类计算机中，尤其是在微型机系统中广泛采用是一种国际上通用的，名叫 ASCII 码（American Standard Code for Information Interchange - 美国信息交换标准代码）编码方案。

ASCII 码规定用 8 位二进制数（即一个字节）来表示这些字符，其中最高位是奇偶校验位，用于检查信息传递过程中是否出错；而低 7 位用于字符的实际编码，即用 7 位二进制数表示一个字符，一共可以表示 $2^7 = 128$ 种符号。

字符的 ASCII 代码

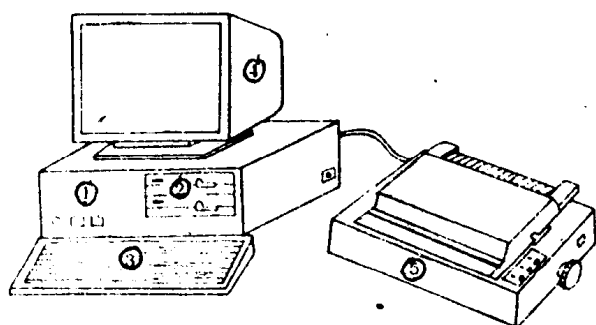
字 符	ASCII 代 码			
	二进制表示	十进制表示	八进制表示	十六进制表示
A	01000001	65	101	41
3	00110011	51	63	33
d	01100100	100	144	64
%	00100101	37	45	25
!	00100001	33	41	21
<	00111100	60	74	3C
+	00101011	43	53	2B
M	01001101	77	115	4D
t	01110100	116	164	74

第二章 计算机系统的硬件与软件

第一节 微机系统的基本硬件配置及主要技术指标

从外观上看，一个典型的微机系统的硬件（Hardware）主要由主机、驱动器、显示器、键盘、打印机五部分组成，这是 PC 机系统的基本硬件配置，如图 2-1 与 2-2 所示。

从图中可见，主机箱内有主机（或主机板）、硬盘驱动器、软盘驱动器、与电源等。硬、软驱动器属于外围设备。所以微机系统硬件是指哪些看得见摸得着的，并能完成一定任务的实际装置的总和。它包括中央处理器、内存储器、输入 / 输出接口、输入 / 输出设备（包括外存）等，从功能上看微机系统可以分为主机与外围设备两大部分，如图 2-3 所示。



IBM-PC 系统硬件配置

① 主机 ② 驱动器 ③ 键盘 ④ 显示器 ⑤ 打印机

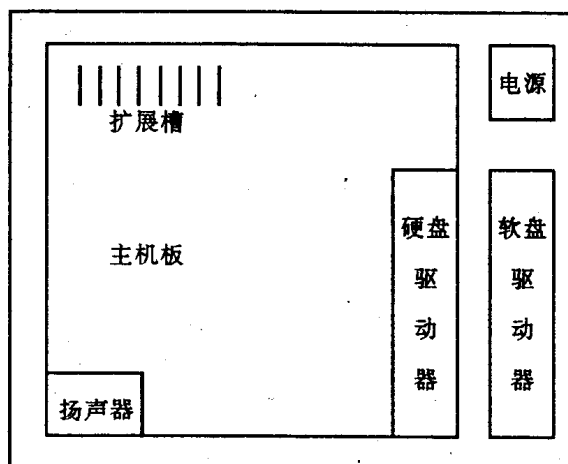


图 2-1 微机系统基本硬件配置

图 2-2 主机箱内各部分示意图

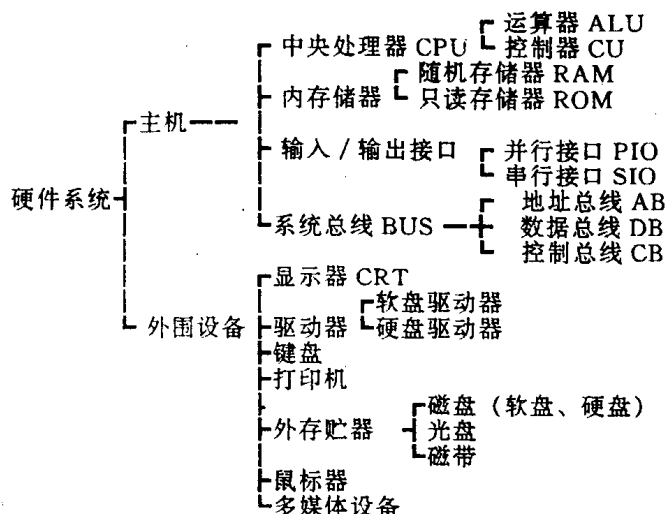


图 2-3 硬件系统的各组成部分的名称