计算机技术与工程应用基础

王 镭 主编

天津大学出版社

非计算机类工科硕士研究生教学用书

计算机技术及工程应用基础

第一分册

王 镭　　主 编
赵文钦　　副主编

天津大学出版社

内 容 提 要

本书是为非计算机类理工科硕士研究生学位课编写的教材，分四册出版。

第一分册分两篇。第一篇系统介绍了计算机系统的基本知识、数据结构、程序设计技术、软件工程；第二篇介绍了数学模型、数值算法设计、曲线与曲面拟合方法、有限元法和数学模型的应用。

本书可为非计算机类理工科硕士研究生的教材，也可供本科生及工程技术人员参考。

本书编写人员（按姓氏笔画排列）：

王保旗、张崇文、胡人文、赵文钦、柏家球

(津)新登字012号

计算机技术及工程应用基础第一分册

王 镭 主编

*

天津大学出版社出版

（天津大学内）

河北省永清县印刷厂印刷

新华书店天津发行所发行

*

开本：787×1092毫米 1/16 印张：26 1/4 字数：665千字

1991年12月第一版　1991年12月第一次印刷

印数：1—3000

ISBN 7-5618-0307-9

TP·34

定价：11.00元

前　言

本书是根据工科研究生计算机系列选修课教学大纲编写而成的，是非计算机类硕士研究生学位课程的教材。编写本书旨在加强理论基础，着重工程应用，扩大知识领域，使硕士研究生能够在大学本科学习的基础上受到系统性和综合性训练，达到优化硕士研究生知识结构、强化高层次人才的工程逻辑思维和设计能力之目的。因此，对这样一门实用性较强的基础课程，无论从内容的选择和体系的形成，还是从教学时间的安排和教学质量的提高等方面均进行了初步的研究和探索。

随着科学技术的发展，应用计算机解决科学研究、工程技术、企业管理等方面的问题愈来愈显现出重要性。掌握计算机应用技能的人才会具有更强的竞争能力。而研究生的培养既要满足今天的社会需求，亦要适应未来对高层次人才的需要。然而，我们原设置的计算机系列选修课较为分散，系统性也不强。为了更好地发挥学校的教学力量和设备条件的优势，为了充分利用有限的教学时间，我们精选了本书的内容，并注意各部分的衔接和联系，也加强了基础理论知识。

全书由计算机技术基础、数学模型与数值算法设计、微型机检测与控制接口技术、系统仿真与优化和计算机辅助绘图等五篇共三十一章组成。五篇内容既相互独立，又相互联系，并紧密联系工程实际。读者通过学习与实践能够获得较为系统的应用计算机的知识和技术，提高应用计算机的能力。

本书的第一、第二篇是非计算机专业的硕士学位课的必修内容，各专业统一授课。其余的第三、第四、第五篇等可视各专业的情况选修或自学。本书不仅可作为非计算机专业的硕士研究生教材或教学参考书，也可作为计算机专业硕士研究生、科研人员、工程技术人员、企业管理人员和教师的参考书。

王镭教授对全书系统的形成、各篇章的结构布局以及全书的编著内容都给予了具体的指导和订正。

本书第一篇由赵文钦、王保旗编写，第二篇由张崇文、柏家球、胡人文编写，第三篇由何乃文、陈本智、何丕廉编写，第四篇由姬维君、田允沿编写，第五篇由焦法成、胡人文编写。

本书在编写过程中，天津大学计算机工程与科学系许镇宇教授审核了全书的内容并得到天津大学研究生院的支持和帮助。在此一并致谢。

由于水平的限制，错误不妥之处恳请批评指正。

编　者

1990. 10.

目　录

第一篇　计算机技术基础

第二篇 数学模型与数值算法设计

第一篇　计算机技术基础

第一章　计算机系统概论

根据目前科研、生产、管理和数学方面应用计算机的现状，本章将以微型计算机为背景，介绍计算机原理、结构和特点。

1.1　计算机中的数和编码系统

计算机的最基本功能是进行数的计算和处理加工。数在机器中是以器件的物理状态表示的。为了使表示更为方便和可靠，计算机中主要采用二进制数字系统。换言之，计算机只认得二进制数。基于此，所有的字符，包括字母、数字、各种技术符号及汉字等均用二进制编码表示。

1.1.1　进位计数制

日常生活中，人们会遇到十进制、十二进制、十六进制和六十进制等。常用的是十进制。

一、十进制数

一个十进制数有两个主要特点：

（1）它有十个不同的数字符号，即：0、1、2、3、4、5、6、7、8、9。

（2）它逢“十”进位。同一个数字符号在不同的位置（数位）上代表着不同的数值。例如，在数99.9中，小数点前边的第二位、第一位分别代表着9×10^{1}和9×10^{0}，而小数点后的那一个“9”却代表着9×10^{-1}的值。所以这个数可以写成

$$99.9=9\times10^{1}+9\times10^{0}+9\times10^{-1}$$

一般地说，任意一个十进制数A，都可表示为：$A_{n-1}A_{n-2}\cdots A_1A_0\cdot A_{-1}A_{-2}\cdots A_{-m}$，其值为：

$$\begin{aligned}A&=A_{n-1}\cdot10^{n-1}+A_{n-2}\cdot10^{n-2}+\cdots+A_1\cdot10^{1}+A_0\cdot10^{0}\\&+A_{-1}\cdot10^{-1}+A_{-2}\cdot10^{-2}+\cdots+A_{-m}\cdot10^{-m}\\&=\sum_{i=n-1}^{-m}A_i\cdot10^{i}\end{aligned}$$

其中i表示该位数在整个数中相对于小数点前或后的位置，A_i为第i号位的数码，它可以是0—9中的任一个，由具体的数A来确定；m和n为正整数，n为小数点前面的位数，m为小数点后面的位数。式中10被称为计数制的底数（或称为基数）。所以，这是十进制数。

二、二进制数

与十进制类似，二进制数也有两个特点：

(1) 它的数值部分，只用两个符号0和1表示。

(2) 它逢“二”进位。同样，不同位置（数位）上的数码的值是不同的。例如

$$(1001)_2 = 1\times 2^3 + 1\times 2^0 = (8+1)_{10} = (9)_{10}$$

$$(1001\cdot 101)_2 = 1\times 2^3 + 1\times 2^0 + 1\times 2^{-1} + 1\times 2^{-3} = (9.625)_{10}$$

一般地说，任意一个二进制数B，都可以表示为：$B_{n-1}B_{n-2}\cdots B_1B_0\cdot B_{-1}B_{-2}\cdots B_{-m}$，其值为：

$$B = B_{n-1}\cdot 2^{n-1} + B_{n-2}\cdot 2^{n-2} + \cdots + B_1\cdot 2^1 + B_0 2^0$$

$$+ B_{-1}\cdot 2^{-1} + B_{-2}\cdot 2^{-2} + \cdots + B_{-m}\cdot 2^{-m} = \sum_{i=n-1}^{-m} B_i\cdot 2^i$$

其中B_i只能取 0 或 1，由具体的数B确定。n、m为正整数，n为小数点前面的位数，m为小数点后面的位数，2 是进位制的基数，故称为二进制。

三、八进制数

它也有两个主要特点：

(1) 用八个不同的数码符号0—7表示数值。

(2) 逢“八”进位。例如：

$$(327)_8 = 3\times 8^2 + 2\times 8^1 + 7\times 8^0 = (215)_{10}$$

$$(372.135)_8 = 3\times 8^2 + 7\times 8^1 + 2\times 8^0 + 1\times 8^{-1} + 3\times 8^{-2} + 5\times 8^{-3}$$

$$= (250.18164)_{10}$$

所以，任意一个八进制数C可表示为：

$$C = C_{n-1}\cdot 8^{n-1} + C_{n-2}\cdot 8^{n-2} + \cdots + C_1\cdot 8^1 + C_0\cdot 8^0$$

$$+ C_{-1}\cdot 8^{-1} + C_{-2}\cdot 8^{-2} + \cdots + C_{-m}\cdot 8^{-m} = \sum_{i=n-1}^{-m} C_i\cdot 8^i$$

其中，C_i取0—7中的某个值，这取决于C；n、m为正整数，n为小数点前边的位数，m为小数点后边的位数；8 为基数，故称八进制。

四、十六进制数

它也有两个特点：

(1) 用十六个不同的数码符号0—9和A、B、C、D、E、F表示每一位的值。它与十进制和二进制数之间的关系如表1-1所示。

(2) 逢“十六”进位。在不同数位上的数码表示不同的值。例如：

$$(327)_{16} = 3\times 16^2 + 2\times 16^1 + 7\times 16^0 = (807)_{10}$$

$$(3AB.11)_{16} = 3\times 16^2 + A\times 16^1 + B\times 16^0 + 1\times 16^{-1} + 1\times 16^{-2}$$

$$= (939.0664)_{10}$$

所以，一个任意的十六进制的数D，可以表示为：

$$D = D_{n-1}\cdot 16^{n-1} + D_{n-2}\cdot 16^{n-2} + \cdots + D_1\cdot 16^1 + D_0\cdot 16^0$$

$$+ D_{-1}\cdot 16^{-1} + D_{-2}\cdot 16^{-2} + \cdots + D_{-m}\cdot 16^{-m} = \sum_{i=n-1}^{-m} D_i\cdot 16^i$$

其中，D_i可取0—F之间的某值，这取决于数值D；n、m为正整数，n为小数点前边的位数，m为小数点后边的位数；16为基数，故称十六进制。

表1-1　二进制、十进制和十六进制对照表

十进制数	十六进制数	二进制数	十进制数	十六进制数	二进制数
0	0	0000	8	8	1000
1	1	0001	9	9	1001
2	2	0010	10	A	1010
3	3	0011	11	B	1011
4	4	0100	12	C	1100
5	5	0101	13	D	1101
6	6	0110	14	E	1110
7	7	0111	15	F	1111
			16	10	10000

综合上述几种计数制，可以把它们的特点概括为：

（1）每一种计数制都有固定的基数J，它的每一位可取这J个数值中的不同的数值；

（2）它是逢“J”进位的。因此，每一个数位i对应一个固定的值J^i。J^i被称为该位的“权”。小数点前边的各位的权依次是基数J的非负次幂；而小数点后边各位的权依次是基数J的负次幂。与此相关，若小数点的位置在原数中向前在（左）移一位，则等于该数缩小了J倍；反之，向后（右）移一位，则等于扩大了J倍。

1.1.2　数在不同计数制之间的转换

一、二进制数转换成十进制数

根据二进制的定义，只要将它按权展开相加即可。例如：

$$(111.101)_2 = 1\times 2^2 + 1\times 2^1 + 1\times 2^0 + 1\times 2^{-1} + 1\times 2^{-3}$$
$$= (7.625)_{10}$$

二、十进制整数转换成二进制整数

若把十进制整数215转换成二进制数，即要变成如下的形式

$$(215)_{10} = (K_{n-1}K_{n-2}\cdots K_1K_0)_2$$

问题就是要找到K_{n-1}、K_n、…、K_0的每个值，而这每个值不是0就是1，取决于欲转换的十进制数（例中即215）。根据二进制数的定义

$$(K_{n-1}K_{n-2}\cdots K_1K_0)_2 = K_{n-1}\cdot 2^{n-1} + K_{n-2}\cdot 2^{n-2} + \cdots + K_1 2^1 + K_0 2^0$$

所以

$$(215)_{10} = K_{n-1}\cdot 2^{n-1} + K_{n-2}\cdot 2^{n-2} + \cdots + K_1 2^1 + K_0 2^0$$

显然，等式的右边除最后一项K_0外，都包含有2的因子，它们都能被2整除尽。故用2去除十进制数$(215)_{10}$，则其余数即为K_0。

$$2\,\overline{)\,215}$$
$$107 \quad \cdots\cdots \text{余}\ 1 \text{为}\ K_0$$

注意到$(215)_{10}$被2整除且去掉余数后所得的整数商

$$(107)_{10} = K_{n-1}\cdot 2^{n-2} + K_{n-2}\cdot 2^{n-3} + \cdots + K_2\cdot 2^1 + K_1\cdot 2^0$$

而这个等式的右边也是除了最后一项K_1外，都包含有2的因子，都能被2整除。故若用2去除107，其余数必为K_1，即

$$2\,\overline{)\,107}$$
$$53 \quad \cdots\cdots \text{余}1 \text{为}K_1$$

用这样的方法一直继续下去，直到商为0，就可以得到K_{n-1}、K_{n-2}、…、K_1、K_0各值。

$$
\begin{array}{r|l}
2 & 215 \\
2 & 107 \quad \cdots\cdots 余1为K_0 \\
2 & 53 \quad \cdots\cdots 余1为K_1 \\
2 & 26 \quad \cdots\cdots 余1为K_2 \\
2 & 13 \quad \cdots\cdots 余0为K_3 \\
2 & 6 \quad \cdots\cdots 余1为K_4 \\
2 & 3 \quad \cdots\cdots 余0为K_5 \\
2 & 1 \quad \cdots\cdots 余1为K_6 \\
 & 0 \quad \cdots\cdots 余1为K_7
\end{array}
$$

所以

$$(215)_{10}=K_7K_6K_5K_4K_3K_2K_1K_0=(11010111)_2$$

由此可以概括出把十进制整数转换为二进制整数的方法是：用2不断地去除欲转换的十进制数，直至商为0。每次的余数即为二进制数列，最初得到的为整数的最低有效数K_0，最后得到的为最高有效数K_{n-1}。

三、十进制小数转换为二进制小数

若把十进制小数0.6875转换为二进制小数，即

$$(0.6875)_{10}=(0.K_{-1}K_{-2}\cdots K_{-m})_2$$

就要由$(0.6875)_{10}$确定K_{-1}—K_{-m}的值。按照二进制小数的定义，可把上式写成

$$(0.6875)_{10}=K_{-1}\cdot 2^{-1}+K_{-2}\cdot 2^{-2}+\cdots+K_{-m}\cdot 2^{-m}$$

若将上式两边都乘以2，则得

$$(1.375)_{10}=K_{-1}+(K_{-2}\cdot 2^{-1}+K_{-3}\cdot 2^{-2}+\cdots+K_{-m}\cdot 2^{-m+1})$$

显然，等式右边括号内的数是小于1的（因乘以2之前是小于0.5的）。而若两个小数相等，必然是整数部分和小数部分分别相等。故$K_{-1}=1$。剩下的

$$(0.375)_{10}=K_{-2}\cdot 2^{-1}+K_{-3}\cdot 2^{-2}+\cdots+K_{-m}\cdot 2^{-m+1}$$

两边再乘以2，则得

$$(0.75)_{10}=K_{-2}+K_{-3}\cdot 2^{-1}+K_{-4}\cdot 2^{-2}+\cdots+K_{-m}\cdot 2^{-m+2}$$

于是$K_{-2}=0$。如此继续下去，可逐个得到K_{-1}、K_{-2}、…、K_{-m}的值。

$$
\begin{array}{r}
0.6875 \\
\times \quad 2 \\
\hline
1.3750 \quad \cdots\cdots 整数部分是1，为K_{-1}
\end{array}
$$

$$
\begin{array}{r}
0.375 \\
\times \quad 2 \\
\hline
0.750 \quad \cdots\cdots 整数部分是0，为K_{-2}
\end{array}
$$

$$
\begin{array}{r}
0.750 \\
\times \quad 2 \\
\hline
1.500 \quad \cdots\cdots 整数部分是\ ，为K_{-}
\end{array}
$$

$$
\begin{array}{r}
0.5 \\
\times \quad 2 \\
\hline
1.0 \quad \cdots\cdots 整数部分是1，为K_{-}
\end{array}
$$

所以

$$(0.6875)_{10}=(0.1011)_2$$

要注意，在十进制小数转换成二进制小数时，不断用2去乘不一定都能使尾数部分等于零，过程可能会无限制的进行下去。这时只需根据精度要求取足二进制小数部分的位数即可。

由此可概括出十进制小数转换为二进小数的方法是：不断地用2去乘要转换的十进制小数，将每次所得的整数（0或1），依次记为K_{-1}、K_{-2}、…。若乘积的小数部分最后能为0，那么最后一次乘积的整数部分记为K_{-m}。则

$$0\cdot K_{-1}K_{-2}\cdots K_{-m}$$

即为所求的二进制表达式。

但十进制小数，并不都是能用有限位的二进制小数精确表示的。则可根据精度要求取m位，得到十进制小数的二进制表达式。

十进制整数和小数都能分别转换为二进制的整数和小数，所以一个具有整数和小数部分的十进制数，在转换为二进制数时，只要把它分为整数和小数两部分，然后再把它们分别转换为二进制表达式，最后用小数点把这两部分连起来就可以了。

四、任意进制数与十进制数之间的转换

一般地说，任意进制数与十进制数之间的转换，原理和方法跟上述二进制与十进制之间的转换类似。比如，八进制与十进制之间，八进制数转换为十进制数，只要将它按权展开，然后相加即可。十进制转换为八进制，就需将其整数部分与小数部分分开分别进行转换。十进整数转换为八进制整数，只要不断用8去除，每次所得的余数就是八进制的系数，最先得到的是八进制整数的最低有效数，最后得到的是最高有效数；而十进制小数在转换时只要不断用8去乘，每次所得的整数部分，即为八进制小数的系数。最先得到的是八进制小数的最高有效数，最后得到的是最低有效数。

例：将十进制数835.6875转换为八进制数。

首先将整数部分与小数部分分开，分别把它们转换为八进制数：

$$\begin{array}{r|l}
8 & 835 \\
8 & 104 \quad \cdots\cdots 余3为K_0 \\
8 & 13 \quad \cdots\cdots 余0为K_1 \\
8 & 1 \quad \cdots\cdots 余5为K_2 \\
 & 0 \quad \cdots\cdots 余1为K_3
\end{array}$$

所以 $(835)_{10}=(1503)_8$

$$\begin{array}{r}
0.6875 \\
\times\quad 8 \\
\hline
5.5000
\end{array} \quad \cdots\cdots 整数部分是5，为K_{-1}$$

$$\begin{array}{r}
0.5 \\
\times\quad 8 \\
\hline
4.0
\end{array} \quad \cdots\cdots 整数部分是4，为K_{-2}$$

所以 $(835.6875)_{10}=(1503.54)_8$

五、八进制与二进制之间的转换

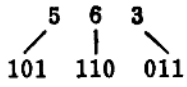
因为$2^3=8$，所以一位八进制数相当于三位二进制数，它们是完全对应的。因此，八进制

数与二进制数之间的转换十分方便。

1.八进制转换为二进制

每位八进制数，可以用三位二进制数表示。

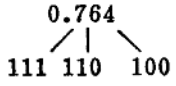
例：$(563)_8$，可分别把5、6、3用三位二进制数表示如下



即

$$(563)_8=(101110011)_2$$

例：$(0.764)_8$的每一位分别用三位二进制数表示如下



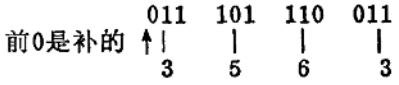
即

$$(0764)_8=(0.111110100)_2=(0.1111101)_2$$

2.二进制转换为八进制

二进制整数转换为八进制时，可以从最低位（小数点之左第一位）开始，向左每三位分为一组，不够三位的用0向左补足三位，然后把每三位二进制数用相应的八进制表示即可。

例：$(11101110011)_2$的转换如下

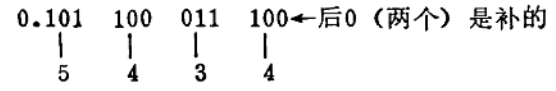


所以

$$(11101110011)_2=(3563)_8$$

二进制小数转换为八进制时，从小数点右面的第一位开始，向右每三位分为一组，最后不足三位的用0向右补足三位，然后把每一组二进制数用相应的八进制表示出来即可。

例：$(0.1011000111)_2$的转换如下



所以

$$(0.1011000111)_2=(0.5434)_8$$

可见，八进制与二进制之间的转换十分方便。在计算机中，数是用二进制表示的，但二进制书写起来太长、易错，通常用八进制（或十六进制）书写。另外，从十进制转换为八进制的计算过程短、方便。所以，在从十进制转换为二进制时，常是先转换为八进制，然后再转换为二进制。

六、十六进制与二进制的转换

在微型计算机中，目前通用的字长为8位，正好可用两位十六进制数表示。

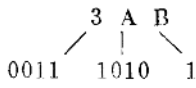
而$16=2^4$，所以，在十六进制与二进制之间也存在着类似于八进制与二进制之间的简单而又直接的联系：用四位二进制数表示一位十六进制数。只要熟悉这个联系，十六进制与二进制之间的转换也十分方便。

1.十六进制转换为二进制

不论是十六进制的整数还是小数，只要把每一位十六进制的数用相应的四位二进制数代替，就可以转换为二进制数。

例：$(3AB)_{16}$可转换为

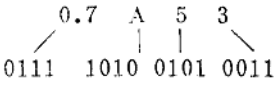
所以



$$(3AB)_{16}=(0011\ 1010\ 1011)_2=(1110101011)_2$$

$(0.7A53)_{16}$可转换为

所以



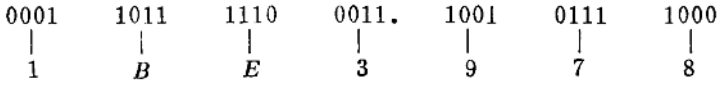
$$(0.7A53)_{16}=(0.0111101001010011)_2$$

2.二进制转换为十六进制

二进制的整数部分由小数点向左，每四位一组，最后不足四位的前面补0；小数部分由小数点向右，每四位一组，最后不足四位的后面补0。然后把每组二进制数用相应的十六进制数代替，即可转换为十六进制数。

例：$(1101111100011.100101111)_2$可转换为

所以



$$(1101111100011.100101111)_2=(1BE3978)_{16}$$①

1.1.3 二进制编码

如前所述，在计算机中，采用的是二进制数，因而，在计算机中表示的数、字母、符号等都要以特定的二进制码来表示，这就是二进制编码。

一、二进制编码的十进制数

因二进制实现容易、可靠、运算简单，所以，在计算机中采用二进制。但是，二进制数不直观，于是在计算机的输入和输出时通常还是用十进制数表示，不过这样的十进制数，要用二进制编码来表示。

一位十进制数可用四位二进制编码来表示。表示的方法很多，较常用的是8421 BCD码，表1-2列出了部分编码。

表1-2 BCD编码表（部分）

十进制数	8421 BCD码	十进制数	8421 BCD码
0	0000	8	1000
1	0001	9	1001
2	0010	10	0001 0000
3	0011	11	0001 0001
4	0100	12	0001 0010
5	0101	13	0001 0011
6	0110	14	0001 0100
7	0111	15	0001 0101

①以后常用数字后面加字母B(Binary)表示为二进制数；用字母O(Octal)表示为八进制数；用字母D(Decimal)或不加字母表示为十进制数；用字母H(Hexadecimal)表示为十六进制数。