

21 世纪应用型本科系列教材

数字电子技术基础

主 编 申忠如

副主编 谭亚丽



西安交通大学出版社
XI'AN JIAOTONG UNIVERSITY PRESS

内容简介

本书是高等院校工科专业应用类教材，可作为电子信息类、计算机类、自动化类等专业的教材，也可供从事相关工作的工程技术人员参考。

21世纪应用型本科系列教材

数字电子技术基础

主 编 申忠如

副主编 谭亚丽



西安交通大学出版社
XI'AN JIAOTONG UNIVERSITY PRESS

西安

内容简介

本书是作者多年从事数字电子技术基础方面的教学与科研工作实践,参照应用型本科人才培养要求,注重专业基础与专业应用的时代改革要求,突出因材施教的教学法研究的需要而编写的。内容包括:数制与码制、数字逻辑基础、集成门电路及其参数、组合逻辑电路分析与设计、集成触发器、时序逻辑电路的分析与设计、脉冲的产生与整形电路、数/模转换和模/数转换和半导体存储器。本书可作为高等学校电气信息类、仪器仪表类、电子信息学科类及其相近专业,本、专科学生“数字电子技术基础”教材和教学参考书,也可作为相关工程技术人员的参考书。

图书在版编目(CIP)数据

数字电子技术基础/申忠如主编. —西安:西安交通大学出版社, 2010.8
ISBN 978-7-5605-3440-4

I. ①数… II. ①申… III. ①数字电路-电子技术
IV. ①TN79

中国版本图书馆 CIP 数据核字(2010)第 024073 号

书 名 数字电子技术基础
主 编 申忠如
责任编辑 任振国 王 欣

出版发行 西安交通大学出版社
(西安市兴庆南路 10 号 邮政编码 710049)

网 址 <http://www.xjtpress.com>
电 话 (029)82668357 82667874(发行中心)
(029)82668315 82669096(总编办)

传 真 (029)82668280
印 刷 陕西丰源印务有限公司

开 本 787mm×1092mm 1/16 印张 17.25 字数 413 千字
版次/印次 2010 年 8 月第 1 版 2010 年 8 月第 1 次印刷
书 号 ISBN 978-7-5605-3440-4/TN·119
定 价 28.00 元

读者购书、书店添货、如发现印装质量问题,请与本社发行中心联系、调换。
订购热线:(029)82665248 (029)82665249
投稿热线:(029)82664954
读者信箱:jdgly@yahoo.cn

版权所有 侵权必究

前言

近年来,电子技术的飞速发展,新器件的不断涌现,促使教学内容必须不断更新和增加,然而由于课时有限,不可能面面俱到。为了保证重点,学有所得,各高校经过几十年的努力,出版了不少的优秀教材,这些教材使用广泛且经过数次修订不断完善,深受广大师生的欢迎。西安交通大学城市学院是应用型“三本”独立院校。近年来一直选用上述教材,发现对三本院校来说,应该有更适合自己的教材,为此,学院教学指导委员会在认真学习教育部的相关文件和调研的基础上,对课程设置和课程大纲的整合进行了专门教改立项。在此基础上,组织了以专业教师和教电子技术的基础课教师相结合的编写小组,确立了编写本书的基本思路。

1. 保证本科的人才培养规格;突出应用型人才的培养特色:“保证基础,突出实践,加强能力。”

2. 鉴于在现代工程应用系统设计中,大都是使用集成化的各种功能模块通过接口组成系统的设计,在教材的内容选取上,对功能模块的原理采用简单的电路予以实现,并由此引出外特性。重点引导学生掌握使用功能模块的功能表,通过接口组成应用系统的设计方法。

3. 为了突出应用型人才的培养特色,在各章中,前有内容提要,后有小结并配有难易程度和数量均适中的思考题和习题;在保证基础够用的前提下,增加了功能模块的使用实例。

4. 在附录 A 中,简单地介绍了半导体二极管、三极管和增强型 MOS 管,使得该教材既可采用“先数字后模拟”,又可采用“先模拟后数字”的教学体系。

5. 在配套的实验指导书中,除了验证性的基本实验外,增加了设计性实践环节,从而培养学生发现问题和解决问题的能力。在课程设计中引入 CPLD 的应用设计,引导学生了解电子技术的新发展。在附录 B 中介绍了在 CPLD 设计中,常用的硬件描述语言 VHDL 基础。

本书的编写工作具体分工如下:谭亚丽编写第 1、3、4 章和附录 A,郭华编写第 7 章和附录 B,谢桦编写第 6、8、9 章,申忠如编写第 2、5 章。全书由申忠如主编,谭亚丽为副主编并协助主编负责编写提纲和统稿工作。在编写过程中,西安交通大学城市学院电气与信息工程系全体教师参与了讨论并提出了宝贵意见。西安交通大学张彦斌教授认真审阅了全稿,并提出了许多修改意见,西安交通大学出版社和学院领导给予了莫大的支持,在此,编者谨向他们致以衷心的感谢。

由于水平有限,一定存在许多不足之处,恳请读者批评指正。

编者

2010 年 2 月于西安交通大学

目 录

第 1 章 数制、数的表示和编码	(1)
1.1 概述	(1)
1.2 几种常用的计数制	(1)
1.2.1 十进制	(1)
1.2.2 二进制	(2)
1.2.3 十六进制	(2)
1.3 数制之间的转换	(3)
1.3.1 二进制数、十六进制数转换为十进制数	(4)
1.3.2 十进制数转换为二进制数、十六进制数	(4)
1.3.3 二进制数与十六进制数之间的转换	(6)
1.4 计算机中数的表示与运算	(7)
1.4.1 数的原码、反码和补码	(7)
1.4.2 补码的运算	(9)
1.5 数字系统的编码	(13)
1.5.1 二-十进制码	(13)
1.5.2 可靠性编码	(15)
1.5.3 字符码与汉字编码	(16)
本章小结	(18)
习题	(18)
第 2 章 数字逻辑基础	(20)
2.1 概述	(20)
2.2 逻辑运算和逻辑门	(20)
2.2.1 基本逻辑运算和基本逻辑门	(20)
2.2.2 复合逻辑运算和复合门	(24)
2.3 逻辑代数的基本定理	(26)
2.3.1 逻辑代数基本定理	(26)
2.3.2 逻辑代数的三条重要规则	(27)
2.4 逻辑函数及其表示方式	(27)
2.4.1 逻辑函数	(27)
2.4.2 逻辑函数常用表示方式	(28)
2.4.3 逻辑函数的两种标准形式	(28)

2.5 逻辑函数的化简	(31)
2.5.1 代数法	(31)
2.5.2 卡诺图法	(32)
本章小结	(37)
习题	(38)
第3章 集成逻辑门电路	(41)
3.1 二极管、三极管开关特性	(41)
3.1.1 二极管的开关特性	(41)
3.1.2 三极管的开关特性	(43)
3.1.3 MOS管的开关特性	(46)
3.2 TTL集成逻辑门	(47)
3.2.1 TTL与非门的工作原理	(48)
3.2.2 TTL与非门的外特性及其参数	(49)
3.3 其他TTL集成门电路	(54)
3.3.1 集电极开路门和输出“线与”	(55)
3.3.2 三态逻辑门和分时“线与”	(56)
3.3.3 使用TTL门电路的几个实际问题	(58)
3.4 CMOS门电路与传输门	(59)
3.4.1 CMOS反相器	(59)
3.4.2 CMOS与非门和或非门	(60)
3.4.3 CMOS传输门	(63)
本章小结	(65)
习题	(66)
第4章 组合逻辑电路的分析和设计	(68)
4.1 概述	(68)
4.2 组合逻辑电路的分析	(69)
4.2.1 组合逻辑电路分析的一般步骤	(69)
4.2.2 组合逻辑电路的分析举例	(69)
4.2.3* 带有负逻辑符号的组合逻辑电路的分析	(71)
4.3 组合逻辑电路的设计	(73)
4.3.1 组合逻辑电路设计的一般步骤	(73)
4.3.2 组合逻辑电路的设计举例	(73)
4.4 常用中规模集成组合逻辑电路	(77)
4.4.1 编码器	(77)
4.4.2 译码器	(82)
4.4.3 多路选择器	(89)
4.4.4 多路分配器	(93)

4.4.5	加法器	(95)
4.4.6	数值比较器	(97)
4.5	组合逻辑电路中的竞争和冒险	(100)
4.5.1	竞争、冒险现象及其产生的原因	(100)
4.5.2	险象的判断及消除	(101)
	本章小结	(103)
	习题	(103)
第5章	集成触发器	(107)
5.1	概述	(107)
5.2	基本 $\overline{R}\overline{S}$ 触发器与逻辑开关	(107)
5.2.1	基本 $\overline{R}\overline{S}$ 触发器	(107)
5.2.2	电平开关与逻辑开关	(109)
5.3	同步 RS 触发器与数据锁存器	(110)
5.3.1	同步 RS 触发器	(110)
5.3.2	数据锁存器	(112)
5.4	主从 JK 触发器	(113)
5.4.1	主从 JK 触发器原理及功能	(113)
5.4.2	集成主从触发器	(114)
5.5	维阻 D 触发器	(116)
5.5.1	维阻 D 触发器的原理及工作特性	(116)
5.5.2	集成维阻 D 触发器	(117)
5.6	触发器之间的相互转换	(119)
5.7	边沿集成触发器的应用举例	(120)
5.7.1	分频器与计数器	(120)
5.7.2	构成简单的控制电路	(121)
	本章小结	(121)
	习题	(122)
第6章	脉冲的产生与整形电路	(125)
6.1	概述	(125)
6.2	5G555 定时器	(126)
6.2.1	5G555 定时器的工作原理与功能表	(126)
6.2.2	5G555 构成的施密特触发器	(127)
6.2.3	5G555 构成的单稳态触发器	(130)
6.2.4	5G555 构成的多谐振荡器	(133)
6.3	集成施密特触发器	(135)
6.4	专用集成单稳态触发器芯片	(136)
6.5	石英晶体多谐振荡器	(138)

本章小结	(139)
习题	(139)
第7章 时序逻辑电路的分析与设计	(143)
7.1 概述	(143)
7.2 基于触发器的同步二进制计数器的分析与设计	(144)
7.2.1 同步二进制计数器分析	(144)
7.2.2 同步二进制计数器设计	(146)
7.3 基于触发器的同步非二进制计数器的分析与设计	(147)
7.3.1 同步非二进制计数器分析	(147)
7.3.2 同步非二进制计数器设计	(151)
7.4 基于触发器的异步二进制计数器的分析与设计	(157)
7.4.1 异步二进制计数器的分析	(157)
7.4.2 异步二进制计数器的设计	(158)
7.5 基于触发器的异步非二进制计数器的分析与设计	(159)
7.5.1 异步非二进制计数器的分析	(159)
7.5.2 异步非二进制计数器的设计	(161)
7.6 集成计数器	(162)
7.6.1 异步集成计数器	(162)
7.6.2 同步集成计数器	(164)
7.7 任意进制集成计数器设计	(165)
7.7.1 反馈清零法	(165)
7.7.2 反馈置数法	(167)
7.7.3 多片集成计数器的级联	(168)
7.8 集成移位寄存器	(171)
7.8.1 移位寄存器工作原理	(171)
7.8.2 集成移位寄存器举例	(172)
7.8.3 环形移位寄存器	(173)
7.9 时序逻辑电路应用举例	(175)
7.9.1 时序逻辑电路分析	(175)
7.9.2* 时序逻辑电路设计	(176)
本章小结	(178)
习题	(179)
第8章 数/模转换和模/数转换	(183)
8.1 概述	(183)
8.2 数/模转换	(183)
8.2.1 权电阻 D/A 转换器原理	(183)
8.2.2 倒 T 型 D/A 转换器原理	(184)

8.2.3	权电流 D/A 转换器原理	(186)
8.2.4	具有双极性输出的 D/A 转换器	(187)
8.2.5	D/A 转换器的技术参数	(188)
8.2.6	集成 D/A 转换器芯片举例	(189)
8.2.7	D/A 转换器的应用举例	(192)
8.3	模/数转换	(192)
8.3.1	A/D 转换器的基本原理	(193)
8.3.2	并行比较型 A/D 转换器原理	(194)
8.3.3	逐次比较型 A/D 转换器原理	(196)
8.3.4	双积分型 A/D 转换器原理	(197)
8.3.5	A/D 转换器的主要技术指标	(199)
8.3.6	集成 A/D 转换器 ADC0809 芯片	(200)
8.4	采样与保持	(203)
8.4.1	采样与保持概述	(203)
8.4.2	典型的采样保持器集成芯片	(204)
	本章小结	(205)
	习题	(205)
第 9 章	半导体存储器与可编程逻辑器件	(207)
9.1	概述	(207)
9.2	随机存取存储器	(208)
9.2.1	RAM 的电路结构与工作原理	(208)
9.2.2	RAM 的容量扩展	(211)
9.2.3	集成 RAM 举例	(212)
9.3	只读存储器	(213)
9.3.1	ROM 的结构	(213)
9.3.2	ROM 的工作原理	(214)
9.3.3	可编程只读存储器	(216)
9.3.4	ROM 的应用	(217)
9.4	几种常见的可编程逻辑器件简介	(218)
	本章小结	(219)
	习题	(220)
附录 A	半导体器件知识	(222)
A.1	半导体的基本知识	(222)
A.2	半导体二极管	(223)
A.3	双极型晶体三极管	(225)
A.4	绝缘栅场效应管	(228)
A.5	理想运算放大器	(230)

A.6	电压比较器	(231)
附录 B	硬件描述语言 VHDL 简介	(233)
B.1	概述	(233)
B.2	VHDL 的基本结构	(233)
B.3	VHDL 的语言要素	(236)
B.4	VHDL 的运算	(238)
B.5	VHDL 的顺序描述语句	(239)
B.6	VHDL 的进程描述语句	(247)
B.7	时钟信号的 VHDL 描述方法	(255)
B.8	时序电路中复位信号的 VHDL 描述方法	(256)
B.9	数字电路系统设计举例	(258)
	参考文献	(265)

第 1 章 数制、数的表示和编码

二进制是数字电子电路的基础,而十进制是日常生活中人们习惯的计数制。本章首先介绍几种常用的数制及它们之间的相互转换,然后介绍在数字系统中数的几种表示形式——原码、反码和补码,以及补码运算,最后介绍几种常用的数字编码。

1.1 概 述

所谓数制就是指计数体制。在数制中,数字符号、位权和基数称为数制的三要素。其中,数字符号指明不同的数字用不同的符号表示;位权指出了每个数字符号所代表的量值是由符号本身和该符号所处的位置共同决定的;基数是数制的根本,它决定了数字符号的个数、位权的大小以及计数时所遵循进位和借位的规则。在日常生活中,人们习惯用十进制进行计数,而在计时中对分、秒的计数采用六十进制,一天 24 小时即按二十四进制计数,一年 12 个月即按十二进制计数。

在数字电路中常采用二进制计数。为了让机器能识别二进制数的正负,规定符号位为负号“-”时用“1”来表示,为正号“+”时用“0”来表示,这样表示的数就变成了所谓数的原码。在数码的表示中,常用到原码、反码、补码及其相互之间的转换和运算法则等。另外也常用到其他的编码,例如有权码,8421BCD 码;无权码,格雷码等。这些内容将在本章中一一介绍。

1.2 几种常用的计数制

1.2.1 十进制

用 0、1、2、3、4、5、6、7、8、9 十个数字符号的不同组合,根据“逢十进一”的进位方式来表示一个数的方法,称为十进制计数制,简称十进制(Decimal)。如:333.33 和 271.59 等都是十进制数。

它的三个基本要素分别为:

- (1)数符:十进制有十个数字符号,简称数符,分别用 0、1、2、3、4、5、6、7、8、9 表示。
- (2)位权:位权(Weight)为 10^i ($i=\cdots-2,-1,0,1,2,\cdots$),数字符号的不同排列组合构成一个数,每个数符由于所在位置的不同而代表不同的值。
- (3)基数:基数(Radix)为 10,表示十进制允许选用基本数字符号的个数是十个,决定了十进制的计数所遵循进位和借位的规则是逢十进一,即每位计满 10 时向高位进一;在减法中不够减的位向高位借一当十。

例如:十进制数字 333.33 中各个位置上的 3 所表示的大小就各不相同,最左边的 3 表示

3×10^2 , 而最右边的 3 表示的大小为 3×10^{-2} 。该十进制数按位权可写成如下多项式和的形式

$$(333.33)_{10} = 3 \times 10^2 + 3 \times 10^1 + 3 \times 10^0 + 3 \times 10^{-1} + 3 \times 10^{-2}$$

从上式可以看出, 每个数字符号所表示的数值等于该数字符号值乘以一个与数码所在位置有关的常数, 这个常数即为“位权”, 简称“权”。位权的大小是以基数为底, 数码所在位置的序号 i 为指数的整数次幂, 整数部分为正幂、小数部分为负幂。例如十进制数码的权为 10^i 。在此要注意数码整数部分的位置是从 $i=0$ 开始计算的。

由上例可推广到 N 进制计数制的概念: 基数为 N 的计数制叫做 N 进制计数制, 它包含 $0 \sim N-1$, 共 N 个不同的数字符号, 每个数位计满 N 就向高位进一, 即“逢 N 进一”。

它的三个基本要素分别为:

(1) 数符: $0, 1, 2, \dots, N-1$ 。

(2) 位权: $N^i (i = \dots, -2, -1, 0, 1, 2, \dots)$ 。

(3) 基数: N , 表示 N 进制允许选用基本数字符号的个数为 N 个, 决定了 N 进制在计数时所遵循进位和借位的规则是逢 N 进一, 即每位计满 N 时向高位进一; 借一当 N , 即在减法中不够减的位向高位借一当 N 。

在进位计数制中, 不同数字符号所表示的大小不仅取决于它本身的大小, 还与它所处的位置有关, 相同的数字符号在不同位置表示不同的值, 所以实际上进位计数制采用的是位置表示法。

为了区分各种不同进制的数, 一般在该数的右下角写上基数的阿拉伯数字或者用它们英文单词的第一个字母表示。例如, 十进制数 271.59 可表示为 $(271.59)_{10}$ 、 $(271.59)_D$ 或 271.59D。

任何一个 N 进制数都可以按位权展开的方式写成多项式和的形式, 一个 N 进制数 $(R)_N$ 按权展开的多项式和的一般表达式为:

$$(R)_N = K_{n-1}N^{n-1} + K_{n-2}N^{n-2} + \dots + K_1N^1 + K_0N^0 + K_{-1}N^{-1} + K_{-2}N^{-2} + \dots$$

其中: K_i 为第 i 位的数码, 也称系数, 它的取值为数码 $0 \sim N-1$ 其中之一; N 为基数, N^i 为位权。

1.2.2 二进制

(1) 数符: 二进制(Binary)有两个数符, 分别用 0、1 表示。

(2) 位权: 如 1101.101, 最左边的 1 表示 1×2^3 , 而最右边的 1 则表示 1×2^{-3} , 二进制的位权为 $2^i (i = \dots, -2, -1, 0, 1, 2, \dots)$ 。可将任何一个二进制数展开成多项式和的形式。例如:

$$\begin{aligned} (1001.101)_2 &= (1001.101)_B = 1001.101B \\ &= 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= 8 + 0 + 0 + 1 + 0.5 + 0 + 0.125 = (9.625)_{10} = 9.625D \end{aligned}$$

(3) 基数: 二进制的基数为 2, 表示二进制数字符号有两个, 决定了二进制在计数时所遵循的进位规则是逢二进一, 即在做加法计数时每位计满二时向高位进一; 在做减法计数时不够减的位向高位借一当二。

1.2.3 十六进制

凡是基数为 2^i 的计数制, 从本质上来看都可以认为是二进制计数制, 所以 2^4 即十六进制从本质上讲也是二进制。

(1)数符:十六进制(Hexadecimal)有 16 个数符,分别用 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F 表示。

(2)位权:如 C07. A4H,最左边的 C 表示 12×16^2 ,而最右边的 4 则表示 4×16^{-2} 。十六进制的位权为 $16^i (i=\cdots, -2, -1, 0, 1, 2, \cdots)$ 。可将任何一个十六进制数展开成多项式和的形式。例如

$$\begin{aligned} (C07. A4)_{16} &= (C07. A4)_H = C07. A4H \\ &= 12 \times 16^2 + 0 \times 16^1 + 7 \times 16^0 + 10 \times 16^{-1} + 4 \times 16^{-2} \\ &= 3072 + 0 + 7 + 0.625 + 0.015625 \\ &= (3079.640625)_{10} = 3079.640625D \end{aligned}$$

(3)基数:十六进制的基数为 16,表示十六进制的数字符号有 16 个,决定了十六进制在计数时所遵循的进位规则是逢十六进一,即在做加法计数时每位计满十六时向高位进一;在做减法计数时不够减的位向高位借一当十六。

在数字电路中,也常用十六进制数来表示二进制数。表 1-1 给出了几种进制间的对应关系。

表 1-1 常用数制对照表

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

1.3 数制之间的转换

同一个数可以用不同的计数体制来表示,根据需要在各种数制之间进行转换。例如在数

字计算机中,它的基本运算和操作可执行代码都是以二进制为基础的,而在日常生活中,多使用十进制,所以各种各样的数据、操作命令等在进入计算机之前必须转化成二进制代码,同时计算机处理完的结果也有必要转换成人们所熟悉的十进制数或是各种文字信息。

1.3.1 二进制数、十六进制数转换为十进制数

任何一个 N 进制数都可以按位权展开的方式写成多项式和的形式,在此,若基数和位权已知,则多项式的和即为该 N 进制数所对应的十进制数。因此将这种转换方法称为多项式求和法。

例 1-1 将二进制数 $(10011.01)_2$ 转换为对应的十进制数。

$$\begin{aligned}\text{解} \quad (10011.01)_2 &= 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} \\ &= 16 + 0 + 0 + 2 + 1 + 0 + 0.25 \\ &= (19.25)_{10}\end{aligned}$$

例 1-2 将十六进制数 $(3B.D)_{16}$ 转换为对应的十进制数。

$$\begin{aligned}\text{解} \quad (3B.D)_{16} &= 3 \times 16^1 + 11 \times 16^0 + 13 \times 16^{-1} \\ &= 48 + 11 + 0.8125 = (59.8125)_{10}\end{aligned}$$

由此可将该方法推广到任意非十进制数到十进制数的转换,即要将一个非十进制数转换成十进制数,只需将该非十进制数按位权展开为多项式,然后求和即可。

1.3.2 十进制数转换为二进制数、十六进制数

十进制数转换为二进制数或十六进制数时,需要将被转换的十进制数分为整数部分和小数部分来分别转换。

1. 整数部分的转换

十进制数转换为二进制数时,因为转换前和转换后的数值必然是相等的,而转换后的二进制数表达式是:

$$b_{n-1}2^{n-1} + b_{n-2}2^{n-2} + \cdots + b_22^2 + b_12^1 + b_02^0$$

可见只要确定系数 b_i 就可求出转换后的数值,所以对整数部分的转换采用除基数 2 取余数的方法,从低位到高位依次获得各位数码的值,直到商等于 0 为止。具体过程是:将整数部分除以基数,取此时的余数作为二进制数的最低位 b_0 ,再将商除以基数,取余数作为二进制数的次低位 b_1 ,依此类推,直到商为 0 时,这时的余数作为二进制数的最高位 b_{n-1} ,这样由余数所构成的序列 $b_{n-1} \cdots b_2 b_1 b_0$ 便是转换成的二进制数。

例 1-3 求 $(25)_{10} = (\quad)_2$

解	$2 \overline{) 25}$	余数	二进制数
	$2 \overline{) 12} \cdots \cdots \cdots$	余 1	最低位 b_0
	$2 \overline{) 6} \cdots \cdots \cdots$	余 0	b_1
	$2 \overline{) 3} \cdots \cdots \cdots$	余 0	b_2
	$2 \overline{) 1} \cdots \cdots \cdots$	余 1	b_3
	0	余 1	最高位 b_4

$$(25)_{10} = (11001)_2$$

同理,十进制整数部分转换为十六进制数时采用除基数 16 取余数的方法,从低位到高位依次获得各数码的值,直到商为 0 为止,此时得到的余数序列 $h_{n-1} \cdots h_2 h_1 h_0$ 便是转换成的十六进制数。

例 1-4 求 $(46)_{10} = (\quad)_{16}$ 。

解

$16 \overline{)46}$	余数	十六进制数
$16 \overline{)2}$	余 14	最低位 h_0
0	余 2	最高位 h_1

$(46)_{10} = (2E)_{16}$

在此要注意第一个余数 14 是十进制数,在写结果时要用相对应的十六进制数 E 来代替。

2. 小数部分的转换

对小数部分的转换采用乘基数取整数的方法,从高位到低位依次获得各位数的值,直到小数部分为 0 或满足有效数字(仅最后一位数字是不确定值)时为止。具体方法是将小数部分乘以基数,取此时乘积的整数部分作为二进制数小数部分的最高位 b_{-1} ,再将乘积的小数部分乘以基数,取乘积的整数部分作为二进制数小数部分的次高位 b_{-2} ,依此类推,直到乘积的小数部分为 0 或满足有效数字为止,这时乘积的整数部分作为二进制数小数部分的最低位 b_{-n} ,这样由乘积的整数部分所构成的序列 $b_{-1}b_{-2} \cdots b_{-n}$ 就是转换成的二进制数。

例 1-5 求 $(0.3125)_{10} = (\quad)_2$

解

$0.3125 \times 2 = 0.625$	 整数为 0	最高位 b_{-1}
$0.625 \times 2 = 1.25$	 整数为 1	b_{-2}
$0.25 \times 2 = 0.5$	 整数为 0	b_{-3}
$0.5 \times 2 = 1.0$	 整数为 1	最低位 b_{-4}

$(0.3125)_{10} = (0.0101)_2$

用同样的方法即乘基数 16 取整数的方法可得到十进制小数部分转换成的十六进制数。

例 1-6 求 $(0.3125)_{10} = (\quad)_{16}$

解

$0.3125 \times 16 = 5.0$	 整数为 5
--------------------------	-------------

$(0.3125)_{10} = (0.5)_{16}$

乘积的小数部分可能无法得到 0 的结果,这时应根据有效数字的要求取一定的有效位数。

例 1-7 求 $(0.314)_{10} = (\quad)_2$, 要求保留 6 位二进制有效数字。

解

$0.314 \times 2 = 0.628$	 整数为 0
$0.628 \times 2 = 1.256$	 整数为 1
$0.256 \times 2 = 0.512$	 整数为 0
$0.512 \times 2 = 1.024$	 整数为 1
$0.024 \times 2 = 0.048$	 整数为 0
$0.048 \times 2 = 0.096$	 整数为 0

$(0.314)_{10} = (0.010100)_2$

如果一个要转换的十进制数既有整数又有小数,则应该将整数部分和小数部分分别转换,然后再将转换结果相加得到最终的结果。

例 1-8 求 $(12.625)_{10} = (\quad)_2$

解 将整数部分 12 按照除基数取余数的方法进行转换。

$$\begin{array}{rcll}
 2 \overline{) 12} & & & \\
 2 \overline{) 6} & \cdots \cdots \cdots & \text{余 } 0 & b_1 \\
 2 \overline{) 3} & \cdots \cdots \cdots & \text{余 } 0 & b_2 \\
 2 \overline{) 1} & \cdots \cdots \cdots & \text{余 } 1 & b_3 \\
 0 & \cdots \cdots \cdots & \text{余 } 1 & b_4
 \end{array}$$

$$(12)_{10} = (1100)_2$$

将小数部分 0.625 按照乘基数取整数的方法进行转换。

$$0.625 \times 2 = 1.25 \quad \cdots \cdots \cdots \text{整数为 } 1 \quad b_{-1}$$

$$0.25 \times 2 = 0.5 \quad \cdots \cdots \cdots \text{整数为 } 0 \quad b_{-2}$$

$$0.5 \times 2 = 1.0 \quad \cdots \cdots \cdots \text{整数为 } 1 \quad b_{-3}$$

$$(0.625)_{10} = (0.101)_2$$

$$\text{所以, } (12.625)_{10} = (1100.101)_2$$

由此可将整数部分“除基数取余法”和小数部分“乘基数取整法”应用到十进制数到任意非十进制数的转换中。

1.3.3 二进制数与十六进制数之间的转换

1. 二进制数转换为十六进制数

十六进制数的基数 16 正好是二进制数基数 2 的 4 次幂,从表 1-1 也可看出,4 位二进制数的 16 种组合和十六进制数的 16 个数码是一一对应的关系,由此很容易得到二进制数转换为十六进制数的方法,即用 4 位二进制数表示 1 位十六进制数。具体方法为:将被转换的二进制数以小数点为界,整数部分由右向左按 4 位一组划分,不够 4 位的在高位补 0;小数部分由左向右 4 位一组划分,数位不够 4 位者在低位用 0 补齐,然后将每组二进制数用相应的十六进制数代替即可。

例 1-9 求 $(101110.101101)_2 = (\quad)_{16}$

$$\begin{aligned}
 \text{解} \quad (101110.101101)_2 &= (0010, 1110, 1011, 0100)_2 \\
 &= (2E, B4)_{16}
 \end{aligned}$$

上述方法可以推广到二进制数和基数为 2^n 的 2^n 进制数的转换,即用 n 位二进制数表示一位 2^n 进制数。

例 1-10 求 $(101011.100101)_2 = (\quad)_8$

$$\begin{aligned}
 \text{解} \quad (101011.100101)_2 &= (101, 011, 100, 101)_2 \\
 &= (53.45)_8
 \end{aligned}$$

例 1-11 $(26)_{10}$ 需要用几位二进制数表示?

解 因 $2^4 < 26 < 2^5$, 所以需要用 5 位二进制数表示。

也可以理解为:要想表示 26 种不同的状态或事件,必须用 5 位二进制数的不同组合才能将其区分开来,但其中还有 6 种组合是没有用到的。

2. 十六进制数转换为二进制数

十六进制数转换为二进制数实际上就是上述过程的逆过程,即将被转换的十六进制数中的每个数码都用对应的 4 位二进制数表示即可。

例 1-12 求 $(9A7E)_{16} = (\quad)_2$

解 $(9A7E)_{16} = (1001, 1010, 0111, 1110)_2$
 $= (100110100111110)_2$

上述方法可以推广到基数为 2^n 的 2^n 进制数和二进制数的转换,即将被转换的 2^n 进制数中的每个数码都用对应的 n 位二进制数表示即可。

例 1-13 求 $(6574)_8 = (\quad)_2$

解 $(6574)_8 = (110, 101, 111, 100)_2$
 $= (110101111100)_2$

1.4* 计算机中数的表示与运算

1.4.1 数的原码、反码和补码

前面所提到的二进制数及其运算都没有涉及对负数的运算,为了解决负数的运算,引出数的原码、反码和补码的概念。

1. 原码

因为在数字电路中仅能识别“0”和“1”,而不能识别“-”和“+”符号,所以规定数中的“+”用“0”表示,“-”用“1”表示,一般将数的最高位设为符号位,数的绝对值按照数制间的转换方法转换为对应的二进制数,这种用“0”和“1”表示符号的数的形式,称为原码,也称为数的机器码。

数 X 的原码定义式:

$$[X]_{\text{原}} = \begin{cases} X & (X \text{ 为正数}) \\ 2^{n-1} + |X| & (X \text{ 为负数}) \end{cases}$$

其中, n 为数 X 转换为机器码的位数。

例 1-14 在 8 位机中,将 +1、-1 和 -9 分别用原码的形式表示出来。

解 +1 的原码为:00000001
 -1 的原码为:10000001
 -9 的原码为:10001001。

例 1-15 在 8 位机中,将 +0 和 -0 分别用原码的形式表示出来。

解 +0 的原码为:00000000
 -0 的原码为:10000000。

设两个正数 X 和 Y ,那么它们之间的运算就有 $(\pm X) \pm (\pm Y)$ 8 种情况。用数的原码进行两个正数的加法运算即 $X+Y$ 时,比较直观,且容易转换。但在进行两个异号数的加、减法运算时,直接按照常规的原码进行计算就行不通了。这时必须通过比较这两个数的绝对值大小来确定减数和被减数及结果的符号,因此完成该功能的电路将会相当复杂,另外如判断、比较等又会使运算速度变慢,为此引出数的反码和补码。

2. 反码

数 X 的反码定义式: