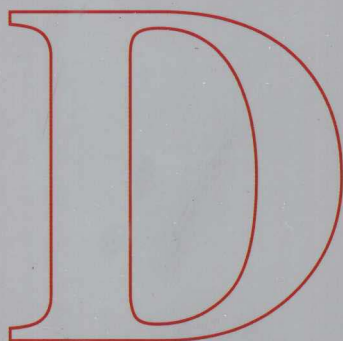


21世纪高等学校计算机**专业**实用规划教材

数字逻辑 与VHDL逻辑设计

盛建伦 编著



清华大学出版社

21 世纪高等学校计算机**专业**实用规划教材

数字逻辑与VHDL逻辑设计

盛建伦 编著

清华大学出版社
北京

内 容 简 介

本书是根据计算机类专业教学的需要编写的,既考虑到计算机专业对数字逻辑课程的要求与其他电气信息类专业的不同,也考虑到与计算机组成原理等后继课程的衔接。全书内容包括逻辑代数基础、门电路、VHDL 语言基础、组合逻辑电路、触发器、时序逻辑电路、用 VHDL 设计逻辑电路、半导体存储器、可编程逻辑器件、脉冲波形的产生与整形、数/模与模/数转换等内容。本书系统地介绍了用 VHDL 设计组合逻辑、触发器、寄存器和时序逻辑的方法。本书的重点内容有丰富的例题和习题,便于自学。

本书可作为计算机科学与技术、网络工程、软件工程等专业的教材,也可供有关专业的工程技术人员参考。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

数字逻辑与 VHDL 逻辑设计/盛建伦编著.--北京:清华大学出版社,2012.8

(21 世纪高等学校计算机专业实用规划教材)

ISBN 978-7-302-29064-3

I. ①数… II. ①盛… III. ①数字电路—逻辑设计 ②硬件描述语言—程序设计 IV. ①TN79
②TP312

中国版本图书馆 CIP 数据核字(2012)第 129696 号

责任编辑:魏江江 赵晓宁

封面设计:傅瑞学

责任校对:焦丽丽

责任印制:杨 艳

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京市密东印刷有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:16.25 字 数:399 千字

版 次:2012 年 8 月第 1 版 印 次:2012 年 8 月第 1 次印刷

印 数:1~3000

定 价:28.00 元

出版说明

随着我国改革开放的进一步深化,高等教育也得到了快速发展,各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度,通过教育改革合理调整和配置了教育资源,优化了传统学科专业,积极为地方经济建设输送人才,为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是,高等教育质量还需要进一步提高以适应经济社会发展的需要,不少高校的专业设置和结构不尽合理,教师队伍整体素质亟待提高,人才培养模式、教学内容和教学方法需要进一步转变,学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007年1月,教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》,计划实施“高等学校本科教学质量与教学改革工程(简称‘质量工程’)”,通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容,进一步深化高等学校教学改革,提高人才培养的能力和水平,更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中,各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势,对其特色专业及特色课程(群)加以规划、整理和总结,更新教学内容、改革课程体系,建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上,经教育部相关教学指导委员会专家的指导和建议,清华大学出版社在多个领域精选各高校的特色课程,分别规划出版系列教材,以配合“质量工程”的实施,满足各高校教学质量和教学改革的需要。

本系列教材立足于计算机专业课程领域,以专业基础课为主、专业课为辅,横向满足高校多层次教学的需要。在规划过程中体现了如下一些基本原则和特点。

(1) 反映计算机学科的最新发展,总结近年来计算机专业教学的最新成果。内容先进,充分吸收国外先进成果和理念。

(2) 反映教学需要,促进教学发展。教材要适应多样化的教学需要,正确把握教学内容和课程体系的改革方向,融合先进的教学思想、方法和手段,体现科学性、先进性和系统性,强调对学生实践能力的培养,为学生知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点,保证质量。规划教材把重点放在公共基础课和专业基础课的教材建设上;特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现教学质量和教学改革成果的教材。

(4) 主张一纲多本,合理配套。专业基础课和专业课教材配套,同一门课程有针对不同层次、面向不同应用的多本具有各自内容特点的教材。处理好教材统一性与多样化,基本教材与辅助教材、教学参考书,文字教材与软件教材的关系,实现教材系列资源配套。

(5) 依靠专家,择优选用。在制定教材规划时要依靠各课程专家在调查研究本课程教

材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主题。书稿完成后要认真实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平教材编写梯队才能保证教材的编写质量和建设力度,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校计算机专业实用规划教材

联系人:魏江江 weijj@tup.tsinghua.edu.cn

前 言

本书是专门为计算机科学与技术专业的数字逻辑课程教学编写的。考虑到计算机科学与技术专业对数字逻辑课程的要求与其他电气信息类专业有所不同,本书在编写时本着“强调逻辑、弱化电子,突出集成电路的外部特性而不是其内部电路,着重于用 VHDL 设计计算机的部件和计算机系统而不是用中小规模集成电路构成数字系统”的思想,对内容的取舍有以下安排:

(1) 把如何用硬件描述语言 VHDL 设计逻辑电路与传统的逻辑设计方法有机地融合起来,达到入门快、概念清晰、易学易用的目的。

(2) 在直接后继课程计算机组成原理中有详细讨论的有符号数的编码和运算、ALU 等内容没有包含在本书中,以减少不必要的重复。

(3) 对于在计算机的时序系统中用到的多谐振荡器和单稳态触发器,在计算机接口电路中常用的施密特触发器、数/模转换器和模/数转换器等内容都有简单介绍。

(4) 74 系列和 4000 系列中小规模集成电路的引脚排列等内容没有包含在本书中。

本书可用于 64 学时(含 12 学时实验)教学,也可用于 48 学时(含 8 学时实验)教学。对于 48 学时教学,可采取不讲可编程逻辑电路、脉冲波形的产生和整形电路、数/模与模/数转换电路、TTL 门电路等内容的安排。

若采用本书作为计算机类专业的数字逻辑课程教材,建议实验教学完全放弃传统的实验箱,改用 VHDL 设计和仿真的形式,实现与后续课程计算机组成原理和计算机系统结构的实验教学的无缝对接。

本书的编写有许多新的尝试,加之作者的经验和水平有限、时间仓促,难免存在缺点和错误,殷切希望各方面的读者多提宝贵意见,并将发现的错误和不当之处及早向本书作者反馈,电子邮箱是 jlsheng@163.com。

最后,感谢郭立智对本书的文字编排等提出的宝贵建议和为书稿所做的工作。

盛建伦

2012 年 3 月于青岛

目 录

第 1 章 数字逻辑基础	1
1.1 数制和码制	1
1.1.1 进位记数制	1
1.1.2 不同记数制间的转换	2
1.1.3 二进制数的运算	4
1.1.4 编码	4
1.2 逻辑代数的基本运算	7
1.2.1 逻辑代数的三种基本运算	8
1.2.2 复合逻辑运算	9
1.3 逻辑代数的基本公式和常用公式	10
1.3.1 基本公式和常用公式	10
1.3.2 若干常用公式	11
1.4 逻辑代数的基本定理	12
1.4.1 代入定理	12
1.4.2 反演定理	12
1.4.3 对偶定理	12
1.5 逻辑函数及其表示方法	13
1.5.1 逻辑函数的表示方法	13
1.5.2 逻辑函数的两种标准形式	16
1.5.3 逻辑函数的卡诺图表示法	18
1.6 逻辑函数的公式化简法	20
1.7 逻辑函数的卡诺图化简法	23
1.8 具有无关项的逻辑函数及其化简	25
1.8.1 约束项、任意项和逻辑函数式中的无关项	25
1.8.2 具有无关项的逻辑函数的化简	25
本章小结	27
习题 1	27
第 2 章 逻辑门电路	31
2.1 概述	31

2.2	二极管门电路	33
2.2.1	二极管与门	33
2.2.2	二极管或门	34
2.3	CMOS 门电路	35
2.3.1	MOS 管开关电路	35
2.3.2	CMOS 反相器	36
2.3.3	CMOS 与非门和或非门	39
2.3.4	漏极开路的 CMOS 门	40
2.3.5	CMOS 传输门和模拟开关	42
2.3.6	三态输出的 CMOS 门电路	43
2.3.7	CMOS 数字集成电路系列	44
2.4	TTL 门电路	46
2.4.1	三极管开关电路	46
2.4.2	TTL 与非门的工作原理	46
2.4.3	TTL 与非门的电压传输特性	47
2.4.4	TTL 与非门的静态输入特性和输出特性	48
2.4.5	TTL 与非门的动态特性	49
2.4.6	其他类型的 TTL 门电路	50
2.4.7	TTL 集成电路的改进系列	51
2.5	TTL 电路与 CMOS 电路的接口	52
	本章小结	54
	习题 2	54
第 3 章	硬件描述语言 VHDL 基础	57
3.1	概述	57
3.2	库和程序包	58
3.2.1	库	58
3.2.2	程序包	59
3.3	VHDL 的语言要素	60
3.3.1	数据对象	60
3.3.2	数据类型	61
3.3.3	运算操作符	61
3.4	顺序语句	62
3.4.1	赋值语句	62
3.4.2	If 语句	63
3.4.3	Case 语句	65
3.5	并行语句	66
3.5.1	process 语句	66
3.5.2	并行信号赋值语句	67

3.6 设计实体	68
3.6.1 实体	68
3.6.2 结构体	69
3.6.3 层次结构设计	71
本章小结	74
习题 3	75
第 4 章 组合逻辑电路	76
4.1 组合逻辑电路的分析方法和设计方法	76
4.1.1 组合逻辑电路的分析方法	76
4.1.2 组合逻辑电路的设计方法	77
4.2 编码器	79
4.2.1 普通编码器	80
4.2.2 优先编码器	81
4.3 译码器	86
4.3.1 二进制译码器	86
4.3.2 二-十进制译码器	88
4.3.3 用译码器设计组合逻辑电路	90
4.3.4 显示译码器	91
4.4 数据选择器	94
4.4.1 数据选择器概述	94
4.4.2 用数据选择器设计组合逻辑电路	95
4.5 加法器	96
4.5.1 半加器和全加器	96
4.5.2 并行加法器和进位链	98
4.5.3 用加法器设计组合逻辑电路	100
4.6 数值比较器	100
4.6.1 一位数值比较器	100
4.6.2 多位数值比较器	101
4.7 组合逻辑电路中的竞争-冒险现象	102
4.7.1 竞争-冒险现象	102
4.7.2 消除竞争-冒险现象的方法	102
4.8 用 VHDL 设计组合逻辑电路	104
本章小结	110
习题 4	111
第 5 章 触发器和寄存器	114
5.1 概述	114
5.2 锁存器	114

5.2.1	基本 RS 锁存器	115
5.2.2	门控 RS 锁存器	117
5.2.3	D 型锁存器	119
5.3	触发器的电路结构与动作特点	120
5.3.1	脉冲触发的触发器	120
5.3.2	边沿触发的触发器	124
5.4	触发器的逻辑功能及其描述方法	127
5.4.1	RS 触发器	128
5.4.2	JK 触发器	128
5.4.3	D 触发器	129
5.4.4	T 触发器	129
* 5.5	触发器的动态特性	130
5.5.1	基本 RS 锁存器的动态特性	130
5.5.2	门控 RS 锁存器的动态特性	131
5.5.3	主从结构触发器的动态特性	131
5.6	用 VHDL 设计触发器	132
5.7	寄存器	137
5.7.1	数码寄存器	137
5.7.2	数据锁存器	138
5.7.3	移位寄存器	138
5.7.4	用 VHDL 设计寄存器	140
	本章小结	142
	习题 5	142
第 6 章	时序逻辑电路	146
6.1	时序逻辑电路的特点和表示方法	146
6.1.1	时序逻辑电路的特点	146
6.1.2	时序逻辑电路的表示方法	147
6.2	基于触发器的时序逻辑电路的分析	147
6.2.1	同步时序逻辑电路的分析	147
* 6.2.2	异步时序逻辑电路的分析	151
6.3	计数器	153
6.3.1	同步计数器	154
6.3.2	异步计数器	159
6.3.3	移位寄存器型计数器	160
6.4	基于触发器的同步时序逻辑电路的设计	161
6.5	基于 MSI 的时序逻辑电路的分析与设计	170
6.5.1	基于 MSI 的时序逻辑电路的设计	170
6.5.2	基于 MSI 的时序逻辑电路的分析	175

6.6 用 VHDL 设计时序逻辑电路·····	176
本章小结·····	182
习题 6 ·····	182
第 7 章 半导体存储器和可编程逻辑器件 ·····	187
7.1 半导体存储器概述 ·····	187
7.2 只读存储器 ·····	189
7.2.1 掩膜 ROM ·····	189
7.2.2 可编程只读存储器·····	190
7.2.3 可擦除的可编程只读存储器·····	191
7.2.4 快闪存储器·····	193
7.3 随机读写存储器 ·····	194
7.3.1 静态随机读写存储器·····	194
7.3.2 动态随机读写存储器·····	195
7.4 存储器容量的扩展 ·····	196
7.4.1 位扩展方式·····	196
7.4.2 字扩展方式·····	197
7.4.3 字位扩展·····	198
7.5 用存储器设计组合逻辑电路 ·····	199
7.6 可编程逻辑器件简介 ·····	201
7.6.1 概述·····	201
7.6.2 PLD 的分类 ·····	202
7.6.3 可编程逻辑器件的逻辑表示·····	202
7.6.4 通用阵列逻辑·····	206
7.6.5 现场可编程门阵列·····	208
7.6.6 PLD 的编程 ·····	209
本章小结·····	209
习题 7 ·····	210
第 8 章 脉冲波形的产生与整形 ·····	213
8.1 多谐振荡器 ·····	213
8.1.1 环形振荡器·····	213
8.1.2 对称式多谐振荡器·····	214
8.1.3 石英晶体多谐振荡器·····	215
8.2 单稳态触发器 ·····	216
8.2.1 积分型单稳态触发器·····	216
8.2.2 微分型单稳态触发器·····	217
8.2.3 单稳态触发器的应用·····	218
8.3 施密特触发器 ·····	219

8.3.1 电路原理·····	219
8.3.2 施密特触发器的应用·····	220
本章小结·····	222
习题 8·····	222
第 9 章 数/模与模/数转换电路·····	224
9.1 概述·····	224
9.2 数/模转换器·····	225
9.2.1 权电阻网络 D/A 转换器·····	225
9.2.2 倒 T 形电阻网络 D/A 转换器·····	226
9.3 模/数转换器·····	227
9.3.1 模/数转换的基本原理·····	227
9.3.2 直接 A/D 转换器·····	229
9.3.3 间接 A/D 转换器·····	233
本章小结·····	235
习题 9·····	235
附录 A 晶体管和液晶显示器基础·····	238
附录 B 逻辑门的符号·····	245
参考文献·····	246

第 1 章

数字逻辑基础

内容提要：作为全书的基础，本章包括两方面内容：一是计算机和数字系统中常用的进位记数制和码制；二是数字系统分析设计的数学方法——逻辑代数。在扼要介绍了逻辑代数的基本公式、常用公式、重要定理和逻辑函数的表示方法之后，重点讲述了如何运用这些公式、定理和方法化简逻辑函数。

1.1 数制和码制

1.1.1 进位记数制

数制(Number System)有进位记数制和非进位记数制之分。在生活中常用的进位记数制有十进制、二十四进制和六十进制等。

进位记数制就是按进位进行计数。进位记数制有两个要素：基数和权。

基数(Radix)指的是计数所用的不同数码的个数。例如，十进制的基数是 10，有 0, 1, ..., 9 共 10 个不同的数码。计数时是逢十进一。

权(Weight)指的是不同数位上的数码代表的数量级。例如，十进制数在第 n 个数位上的权是 10^n 。

基数为 r 的任意进制数 N 可表示为：

$$\begin{aligned}(N)_r &= k_n \cdot r^n + k_{n-1} \cdot r^{n-1} + \cdots + k_0 \cdot r^0 + k_{-1} \cdot r^{-1} + \cdots + k_{-m} \cdot r^{-m} \\ &= \sum_{i=n}^{-m} k_i \cdot r^i\end{aligned}$$

除十进制(Decimal)外，在计算机系统中常用的进位记数制还有二进制、八进制和十六进制等。不同进位记数制之间可以相互转换。

1. 二进制

二进制(Binary)只有 0 和 1 两个数码，计数时是逢二进一。由于容易找到有两个稳定状态的元件来表示 0 和 1，而要想找到有三个以上稳定状态的元件是非常困难的，因此现代计算机内部的数都采用二进制表示。二进制数的一个位叫做“bit(比特)”。

二进制数可以用在数的后面紧跟一个大写字母 B 的方法表示，这里 B 是代表二进制的标志，不是一位数。也可用把二进制数放在括号里并在括号外加下角标 2(表示基数为 2)的方法表示。例如：

$$(1001.101)_2 = 1001.101\text{B}$$

表 1.1 和表 1.2 给出了部分二进制数不同数位的权与十进制数的对应关系。

表 1.1 二进制整数不同数位的权及其对应的十进制数

小数点前的数位	12	11	10	9	8	7	6	5	4	3	2
权	2^{12}	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2
十进制数	4096	2048	1024	512	256	128	64	32	16	8	4

表 1.2 二进制小数不同数位的权及其对应的十进制数

小数点后的数位	1	2	3	4	5	6	7
权	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}
十进制数	0.5	0.25	0.125	0.0625	0.03125	0.015625	0.0078125
分数	$1/2$	$1/4$	$1/8$	$1/16$	$1/32$	$1/64$	$1/128$

对于很大的二进制数，还可以用 K、M、G、T 等表示。 $1K=2^{10}$ ， $1M=2^{20}$ ， $1G=2^{30}$ ， $1T=2^{40}$ 。

2. 八进制

计算机内部是用二进制做运算的，但是人读写由一长串 0 和 1 组成的二进制数很不方便。随着汇编语言的出现，八进制 (Octal) 被引入计算机系统中。八进制有 0, 1, ..., 7 共 8 个不同的数码，计数时是逢八进一。由于 O 和 0 的书写难以区分，因此用大写字母 Q 代替 O 作为八进制的标志。八进制数的两种表示方法为：

$$(27.3)_8 = 27.3Q$$

3. 十六进制

十六进制 (Hexadecimal) 计数时是逢十六进一。由于人类只创造了 0~9 这 10 个不同的数码，而十六进制需要 16 个不同的数码，因此规定用大写字母 A, B, C, D, E, F 分别代表 10, 11, 12, 13, 14 和 15 这 6 个数码。用大写字母 H 作为十六进制的标志。十六进制数的两种表示方法为：

$$(A4F.8B)_{16} = 0A4F.8BH$$

在计算机系统中采用十六进制当然也是为人提供方便，原因是计算机学科采用了“字节 (Byte)”这一标准单位。一位八进制数是 3 个二进制位，一个字节是 8 个二进制位，用八进制就不方便表示了。而一位十六进制数是 4 个二进制位，两位十六进制数刚好是一个字节。

1.1.2 不同记数制间的转换

不同进位记数制之间相互转换的方法有所不同，根据各转换方法的特点可归纳成 5 大类。

1. 二进制与八进制、十六进制数之间的转换

二进制与八进制、十六进制数之间转换的方法很简单，只需从小数点起向左、右分组按数位对应关系直接写出。

三位二进制数对应一位八进制数，例如：

$$(1011.0101)_2 = (13.24)_8$$

$$(162.24)_8 = (1110010.0101)_2$$

四位二进制数对应一位十六进制数,例如:

$$(11011.01011)_2 = (1B.58)_{16}$$

$$(4A.CF)_{16} = (1001010.11001111)_2$$

2. 任意进制数转换成十进制数

无论什么进制的数都可按下面的公式转换成十进制数:

$$\begin{aligned} (N)_r &= k_n \cdot r^n + k_{n-1} \cdot r^{n-1} + \cdots + k_0 \cdot r^0 + k_{-1} \cdot r^{-1} + \cdots + k_{-m} \cdot r^{-m} \\ &= \sum_{i=n}^{-m} k_i \cdot r^i \end{aligned}$$

此方法就是把基数为 r 的任意进制数 N 按权展开相加。例如:

$$(237)_8 = 2 \times 8^2 + 3 \times 8^1 + 7 \times 8^0$$

$$(1AD)_{16} = 1 \times 16^2 + 10 \times 16^1 + 13 \times 16^0$$

3. 十进制整数转换成二进制整数

要把十进制数转换成二进制数,必须把整数和小数分开,按不同的规则进行转换。

十进制整数转换成二进制整数的方法是除 2(基)取余法。具体方法是将被转换的十进制整数反复除以基数 2,直到商为 0。每除一次,提取余数(1 或 0)作为转换成的二进制数的一个位,最先得到的是最靠近小数点的位。

例 1-1 将十进制数 47 转换为二进制数。

解: 用除 2 取余法。

2	47		余数=1, k_0 位	
2	23		余数=1, k_1 位	
2	11		余数=1, k_2 位	
2	5		余数=1, k_3 位	
2	2		余数=0, k_4 位	
2	1		余数=1, k_5 位	
			0	

转换结果是 $(47)_{10} = (101111)_2$ 。

4. 十进制小数转换成二进制小数

十进制小数转换成二进制小数的方法是乘 2(基)取整法。具体做法是将被转换的十进制小数反复乘以基数 2,每乘一次,提取乘积的整数部分(1 或 0)作为转换成的二进制数的一个位,最先得到的是最靠近小数点的位。

例 1-2 将十进制数 0.37 转换为二进制数,要求精度为 1%。

解: 根据表 1.2,二进制小数第 7 位的 1 等于十进制小数的 0.0078125,小于 1%。因此,本题只要求出 7 位二进制小数就可满足题目要求。

用乘 2 取整法:

$$\begin{array}{rcl}
 & 0.37 & \\
 \times & 2 & \\
 \hline
 & 0.74 & \cdots \cdots \cdots \text{整数部分}=0, k_{-1}\text{位} \leftarrow \text{最靠近小数点的位} \\
 \times & 2 & \\
 \hline
 & 1.48 & \cdots \cdots \cdots \text{整数部分}=1, k_{-2}\text{位} \\
 & 0.48 & \\
 \times & 2 & \\
 \hline
 & 0.96 & \cdots \cdots \cdots \text{整数部分}=0, k_{-3}\text{位} \\
 \times & 2 & \\
 \hline
 & 1.92 & \cdots \cdots \cdots \text{整数部分}=1, k_{-4}\text{位} \\
 & 0.92 & \\
 \times & 2 & \\
 \hline
 & 1.84 & \cdots \cdots \cdots \text{整数部分}=1, k_{-5}\text{位} \\
 & 0.84 & \\
 \times & 2 & \\
 \hline
 & 1.68 & \cdots \cdots \cdots \text{整数部分}=1, k_{-6}\text{位} \\
 & 0.68 & \\
 \times & 2 & \\
 \hline
 & 1.36 & \cdots \cdots \cdots \text{整数部分}=1, k_{-7}\text{位}
 \end{array}$$

转换结果是 $(0.37)_{10} = (0.0101111)_2$ 。

值得注意的是,大部分的十进制小数并不能精确地转换成二进制小数,因而存在转换误差。即十进制小数不一定能用有限位的二进制小数精确表示。这是计算机系统的一个不可消除的误差来源。

利用表 1.1 和表 1.2 可以非常快速地将一些十进制数转换成二进制数。

5. 十进制数转换成八进制、十六进制数

原理上,要把十进制数转换成八进制、十六进制数,整数可采用除基取余法,小数可采用乘基取整法。但是,当基数是 8 或 16 时,做除基取余/乘基取整并不方便。所以,先把十进制数转换成二进制数,再写出八进制、十六进制数可能是更为简便的方法。

1.1.3 二进制数的运算

二进制运算的规则非常简单:

$$\begin{array}{lllll}
 0 \pm 0 = 0 & 0 \pm 1 = 1 & 1 \pm 0 = 1 & 1 + 1 = 10 & 1 - 1 = 0 \\
 0 \times 0 = 0 & 0 \times 1 = 0 & 1 \times 0 = 0 & 1 \times 1 = 1 & \\
 0 \div 1 = 0 & 1 \div 1 = 1 & & &
 \end{array}$$

加法如果有进位,是逢二进一。减法如果不够减,有借位,从高位借来的是 2。

多位二进制数的加、减、乘、除运算的方法与十进制数的加、减、乘、除运算的方法类似,但规则简单得多。运算规则简单就使得完成算术运算的逻辑电路简单,这也是在计算机中采用二进制进行运算的原因之一。读者可以自行练习多位二进制数的加、减、乘、除运算,这里就不给例子了。

1.1.4 编码

在计算机和数字系统中,数是用某种代码表示的。除了表示数以外,代码还用来表示其

他符号、文字或任何特定的对象。由于表示任何对象的代码都是用一串 0 或 1 组成的,看起来与二进制数一样,但其实不是数,因此称为二进制形式的代码,或简称为二进制代码。

编制代码的规则称为“码制”。由若干位代码组成的一个字叫“码字”,一种码制可有若干种码字的组合。 N 位二进制形式的代码有 2^N 个不同的码字,最多可以表示 2^N 个不同的对象。

如果一种码制的每个位都有确定的权,就称这种编码为“有权码”,否则称为“无权码”。

下面介绍几种常用的码制。

1. 二-十进制编码(BCD 码)

在计算机中,十进制数可以某种二进制形式的编码表示,称做“二-十进制编码”,简称 BCD 码(Binary Coded Decimal)。一位十进制数有 0~9 这 10 种状态,需要至少 4 位二进制代码来表示。4 位二进制代码有 $2^4=16$ 个码字,从中任意选 10 个就可以构成一种 BCD 码,所以 BCD 码有多种编码方案。常见的 BCD 码编码方案如表 1.3 所示。其中,8421 码、2421 码和 5211 码属于有权码(恒权码),余三码和余三循环码属于无权码。

表 1.3 常见的 BCD 码

十进制数	8421 码	余三码	2421 码	5211 码	余三循环码
0	0000	0011	0000	0000	0010
1	0001	0100	0001	0001	0110
2	0010	0101	0010	0100	0111
3	0011	0110	0011	0101	0101
4	0100	0111	0100	0111	0100
5	0101	1000	1011	1000	1100
6	0110	1001	1100	1001	1101
7	0111	1010	1101	1100	1111
8	1000	1011	1110	1101	1110
9	1001	1100	1111	1111	1010
权	8421	无	2421	5211	无

8421 码是计算机系统中最常用的一种 BCD 码。由于 8421 码是与自然二进制数的前 10 个数相同,因此又称为 NBCD 码。在计算机中可以直接对 8421 BCD 码做加法运算,这样做的好处是可以避免十进制小数转换成二进制小数的转换误差。

余三码(Excess-3 Code)虽然是一种无权码,但由于其每个码字与 8421 码有简单的对应关系,因此也可以进行加法运算。

观察表 1.3 可以发现,同一个码字在不同的码制中可以代表不同的对象。例如,代码 0100 在 8421 BCD 码中表示 4,在余三码中表示 1,在 5211 码中表示 2。同一个对象在不同的码制中可能用不同的代码表示。例如,数字 7 的 8421 BCD 码是 0111,余三码是 1010,2421 码是 1101。

2. 格雷码

格雷码(Gray Code)是一种无权码,采用绝对编码方式。格雷码可以利用其反射性来构成。一位格雷码:0,1。在后面按相反顺序重复这两个数:0,1,1,0。然后,在前一半的左边增加 0,后一半的左边增加 1,得到二位格雷码:00,01,11,10。用同样的方法可得到三位格雷码:000,001,011,010,110,111,101,100。依此类推,可构成多位格雷码。表 1.4 给出