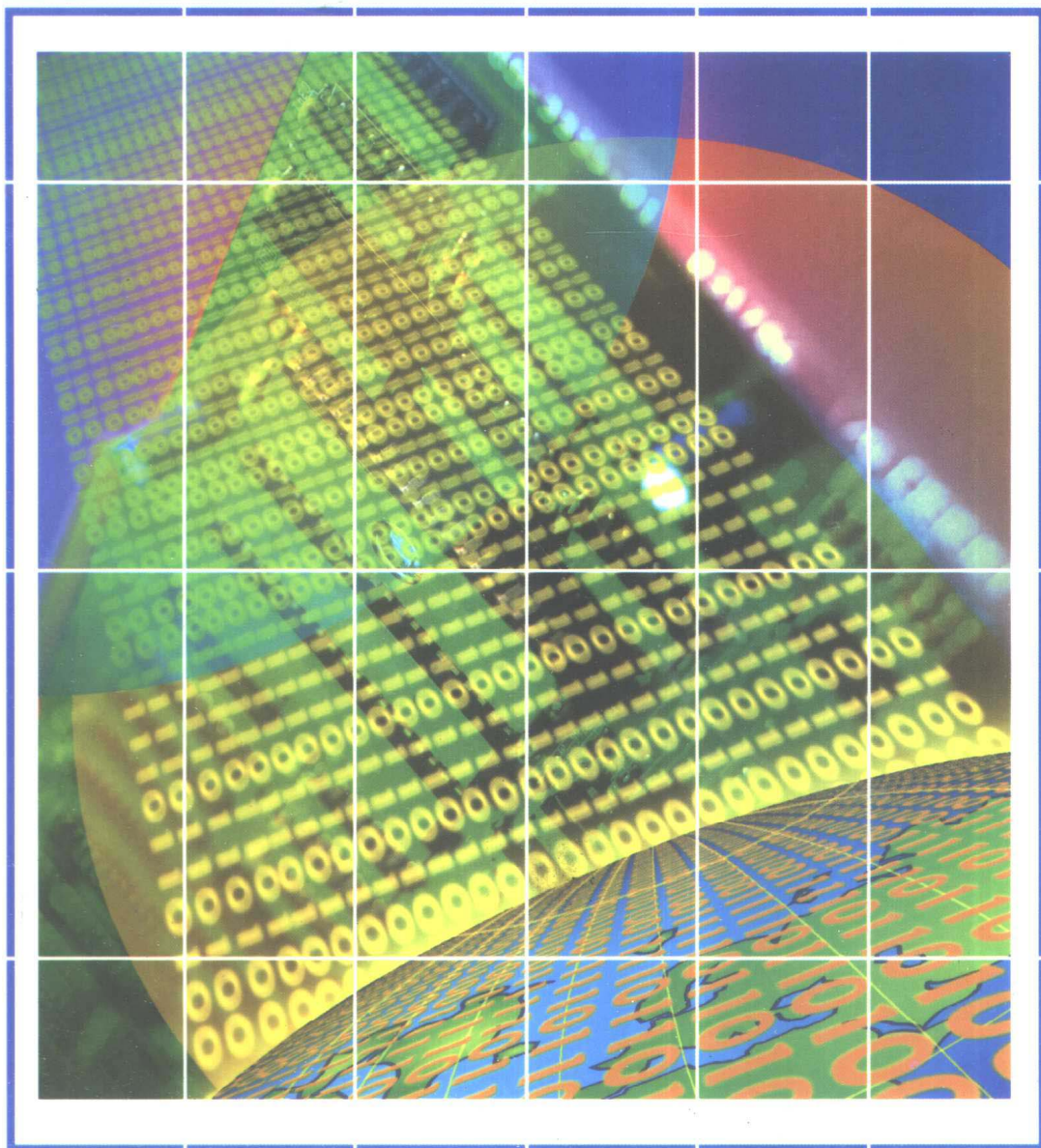


计算机类本科系列教材



数字系统设计基础

侯伯亨 周端 张慧娟 顾新 编著



西安电子科技大学出版社
<http://www.xduph.com>

计算机类本科系列教材

数字系统设计基础

侯伯亨 周 端 编著
张惠娟 顾 新

西安电子科技大学出版社

2000

内 容 简 介

本书是面向 21 世纪的原“数字电路”课程的革新教材,其内容吸收了当前电子电路设计技术的发展潮流,引入了电子设计自动化(EDA)的基本思想,是后续的数字系统设计、ASIC 电路高层设计课程及有关专业课程的必备基础。

本书共分两篇。第一篇数字逻辑电路设计(第 1 章~第 9 章),与原“数字电路”课程内容大体相同,主要介绍逻辑门、逻辑代数、组合逻辑电路分析与设计、时序电路分析与设计等。第二篇数字系统设计基础(第 10 章~第 16 章),主要介绍 VHDL 硬件描述语言及其逻辑电路设计。本书以数字逻辑电路为线索,介绍了用电原理图和 VHDL 语言设计数字逻辑电路的对应方法和相互关系,并列举了众多实例。另外,还对设计中的一些工程问题分别给予了说明。

本书简明扼要、图文并茂,易读易懂,可以作为大学本科学生的教材,也可以作为一般从事电子电路设计的工程师的自学参考书。

图书在版编目(CIP)数据

数字系统设计基础/侯伯亨等编著. —西安:西安电子科技大学出版社,2000.11

计算机类本科系列教材

ISBN 7-5606-0885-X

I. 数… II. 侯… III. 数字系统-设计-高等学校-教材 IV. TP271

中国版本图书馆 CIP 数据核字(2000)第 33573 号

责任编辑 徐德源

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)8227828 邮 编 710071

http://www.xduph.com E-mail: xdupfxb@pub.xaonline.com

经 销 新华书店

印 刷 西安电子科技大学印刷厂

版 次 2000 年 11 月第 1 版 2000 年 11 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印张 18.5

字 数 435 千字

印 数 1~4 000 册

定 价 19.00 元

ISBN 7-5606-0885-X/TN·0151

*** 如有印装问题可调换 ***

本书封面贴有西安电子科技大学出版社的激光防伪标志,无标志者不得销售。

前 言

“数字电路”课程是电子类和计算机、通信等专业的专业基础课，历来受到各工科类大学的重视，其内容主要介绍逻辑门、逻辑代数、组合逻辑电路设计及时序电路设计等。通过这门课程的讲授，使学生掌握数字电路的基本概念，并能运用所学知识设计简单的组合逻辑电路和时序电路。也就是说，要学生掌握电路级规模的数字电路设计，为后续课程如自动控制、数字信号处理、计算机、通信系统的学习打好基础。

随着计算机技术和微电子技术的发展，ASIC(Application Specific Integrated Circuit)电路设计或单片系统(System on Chip)设计已成为当前电子电路设计的重要课题。为了适应这一新的技术发展变化，所有从事电子电路设计的工程技术人员，在专业知识上必须要上一个新的台阶，即从电路级的局部电路设计向系统级的数字系统设计过渡；从人工画电原理图的设计向形式化电路描述(用硬件描述语言描述电路)和计算机自动化设计过渡。由于当前ASIC电路的顶层设计也可以做到与底层工艺设计无关，因此，数字系统设计已可以从ASIC设计技术中分离出来，成为一门相对独立的学科。该学科内容应主要包括：逻辑代数，硬件描述语言(Hardware Description Language)，组合电路和时序电路设计，数字系统设计方法及相关技术，数字系统硬件、软件协同设计(Hardware、Software Co-Design)等。为顺应这种变化，使学生在本科学习期间掌握数字系统设计的基本技术，应修改原“数字电路”课程的内容，将其扩充成两门课程：“数字系统设计基础”(包括原“数字电路”课程内容)和“数字系统设计”。本教材即是“数字系统设计基础”课程的新编教材，它是后续的专业课及“数字系统设计”课的基础课程。其内容主要包括逻辑门、逻辑代数、组合电路和时序电路设计、硬件描述语言及其逻辑电路设计等。这样的内容安排，既拓展了对数字电路和数字系统的认识，也便于实现数字逻辑电路的计算机辅助教学(CAI)。

本教材共分两篇。第1篇为数字逻辑电路设计，共9章。其内容与原“数字电路”课程教材基本相同，只不过对以往教材中太繁琐的或用处不大的内容进行了删节。第2篇为数字系统设计基础，共7章。其内容主要介绍如何用硬件描述语言(VHDL)描述数字系统，并用相应的软件工具生成硬件电路的基本方法。本书第一篇参考学时为40学时，第2篇为20学时。

本教材第1篇的第1章至第5章由周端编写，第6章至第9章由张惠娟

编写，周端对该部分进行统编。第 2 篇的第 10 章至第 16 章由侯伯亨、顾新编写。侯伯亨对全书进行了统稿。

本教材由江小安担任主审，董秀峰也评审了全稿，参加教材内容讨论和审阅的还有孙肖子、赵建、吴慎华等老师。他们都提出了不少宝贵意见。西安电子科技大学计算机学院副院长武波一直关心和支持本教材的编写工作。另外，还得到了西安电子科技大学出版社领导、编辑及有关人员的支持和帮助。在这里一并表示衷心的感谢。

本教材是面向 21 世纪进行教材革新的一次尝试，其中所涉及到的某些观点和内容安排都有不成熟的地方。另外，由于编者水平有限，错误和不当之处在所难免，殷切希望广大读者给予批评指正。

编 者

2000 年 2 月 18 日

目 录

第 1 篇 数字逻辑电路设计

第 1 章 数制与编码	2
1.1 进位计数制	2
1.1.1 十进制	2
1.1.2 二进制	3
1.1.3 八进制和十六进制	3
1.2 数制转换	4
1.2.1 十进制和非十进制之间的转换	4
1.2.2 用二进制数表示八进制数和十六进制数	5
1.3 编码	7
1.3.1 二—十进制(BCD)码	7
1.3.2 格雷码	8
1.3.3 字符码	8
第 2 章 逻辑门	10
2.1 基本逻辑运算及逻辑门	10
2.1.1 “与”逻辑关系及“与”门	10
2.1.2 “或”逻辑关系及“或”门	12
2.1.3 “非”逻辑关系及“非”门	13
2.2 常用的复合逻辑门	14
2.2.1 与非门	14
2.2.2 或非门	14
2.2.3 与或非门	15
2.2.4 异或门和异或非(同或)门	15
2.2.5 正负逻辑	16
2.3 集电极开路与非门和三状态与非门	17
2.3.1 集电极开路与非门(OC 门)	17
2.3.2 三态与非门	18
2.4 数字电路的分类、性能及器件名称的意义	19
2.4.1 TTL 系列逻辑电路	20
2.4.2 CMOS 系列逻辑电路	22
2.4.3 逻辑门电路使用中的几个实际问题	23
2.4.4 常用的 TTL 逻辑门芯片	25

第 3 章 逻辑代数与逻辑函数化简	27
3.1 逻辑代数的基本定律和规则	27
3.1.1 逻辑代数的基本公式	27
3.1.2 逻辑代数的基本规则	28
3.2 逻辑函数的代数化简法	29
3.2.1 逻辑函数与逻辑图	30
3.2.2 逻辑函数的化简原则及化简方法	30
3.3 逻辑函数的标准形式	32
3.4 逻辑函数的卡诺图化简法	34
3.4.1 卡诺图结构	34
3.4.2 逻辑函数的卡诺图表示法	35
3.4.3 逻辑函数的卡诺图化简法	36
3.4.4 具有无关项的逻辑函数的化简	37
第 4 章 组合逻辑电路的分析与设计	39
4.1 组合逻辑电路的分析	39
4.2 组合逻辑电路的设计	41
4.3 组合电路中的竞争与冒险	43
4.3.1 竞争现象	43
4.3.2 冒险现象	43
4.3.3 冒险现象的消除	44
第 5 章 常见的组合逻辑电路	46
5.1 编码器和译码器	46
5.1.1 编码器	46
5.1.2 译码器	50
5.2 数据选择器	57
5.2.1 数据选择器的工作原理	57
5.2.2 数据选择器的应用	58
5.3 算术运算电路	60
5.3.1 半加器	60
5.3.2 全加器	60
5.4 奇偶校验器	63
5.4.1 奇偶校验	63
5.4.2 奇偶校验电路	64
5.4.3 奇偶校验位的产生	66
第 6 章 触发器	67
6.1 基本 RS 触发器	67
6.1.1 基本 RS 触发器电路组成和工作原理	67
6.1.2 逻辑功能描述	68

6.2 钟控触发器	69
6.2.1 钟控 RS 触发器	70
6.2.2 钟控 D 触发器	72
6.2.3 钟控 JK 触发器	72
6.2.4 钟控 T 触发器和 T' 触发器	73
6.2.5 钟控触发器的空翻现象	74
6.3 触发器逻辑功能转换	75
6.3.1 D 触发器转换为其它逻辑功能的触发器	75
6.3.2 JK 触发器转换为其它逻辑功能的触发器	76
第 7 章 时序逻辑电路	78
7.1 时序电路概述	78
7.1.1 时序电路的特点	78
7.1.2 时序电路的分类	79
7.1.3 时序电路的描述方法	79
7.2 同步时序电路分析	81
7.2.1 同步时序电路的分析步骤	81
7.2.2 同步时序电路的分析举例	82
7.3 同步时序电路设计	84
7.3.1 同步时序电路的设计步骤	84
7.3.2 同步时序电路的设计举例	85
7.4 异步时序电路	92
第 8 章 常用时序逻辑器件	94
8.1 计数器	94
8.1.1 同步计数器	95
8.1.2 异步计数器	105
8.2 寄存器	113
8.2.1 基本寄存器	113
8.2.2 移位寄存器	114
8.3 序列信号产生器	118
第 9 章 脉冲产生电路和变换电路	121
9.1 555 定时电路	121
9.1.1 电路组成	121
9.1.2 功能描述	123
9.2 多谐振荡器	123
9.2.1 电路组成	123
9.2.2 工作原理	124
9.3 单稳态电路	126
9.4 施密特电路	128

第 2 篇 数字系统设计基础

第 10 章 数字系统设计概述	130
10.1 传统的系统硬件设计方法	130
10.1.1 采用自下至上(Bottom Up)的设计方法	130
10.1.2 采用通用的逻辑元、器件	132
10.1.3 在系统设计的后期进行仿真和调试	133
10.1.4 主要设计文件是电原理图	133
10.2 利用硬件描述语言(HDL)的硬件电路设计方法	133
10.2.1 采用自上至下(Top Down)的设计方法	134
10.2.2 系统中可大量采用的 ASIC 芯片	135
10.2.3 采用系统早期仿真	136
10.2.4 降低了硬件电路设计难度	136
10.2.5 主要设计文件是用 HDL 语言编写的源程序	137
第 11 章 VHDL 语言描述数字系统的基本方法	138
11.1 VHDL 语言描述数字系统的基本方法	138
11.1.1 VHDL 语言描述电路的基本方法	138
11.1.2 常数、变量、信号所描述的对象	143
11.1.3 数据类型	145
11.1.4 运算操作符	154
11.2 属性(ATTRIBUTE)描述	157
11.2.1 描述时钟边沿的属性 'EVENT	158
11.2.2 描述数据范围的属性 'RANGE 和 'REVERSE-RANGE	158
第 12 章 VHDL 语言的基本设计单元	160
12.1 VHDL 语言的基本设计单元构成	160
12.1.1 最基本的设计单元构成	160
12.1.2 一个完整的设计单元构成	161
12.1.3 设计单元中描述部分的命名规则和注释的标记	162
12.2 构造体的子结构描述	162
12.2.1 BLOCK 语句结构描述	163
12.2.2 进程(PROCESS)语句结构描述	165
12.2.3 子程序(SUBPROGRAM)语句结构描述	166
12.3 库、包集合及配置	172
12.3.1 库	172
12.3.2 包集合	174
12.3.3 配置	175
第 13 章 VHDL 语言构造体的描述方式	178
13.1 构造体的行为描述方式	178

13.1.1	代入语句	178
13.1.2	延时语句	180
13.1.3	多驱动器描述语句	181
13.1.4	GENERIC 语句	183
13.2	构造体的寄存器传输(RTL)描述方式	184
13.2.1	RTL 描述方式的特点	185
13.2.2	使用 RTL 描述方式应注意的几个问题	186
13.3	构造体的结构描述方式	191
13.3.1	构造体结构描述的基本框架	191
13.3.2	COMPONENT 语句	192
13.3.3	COMPONENT_INSTANT 语句	192
第 14 章	VHDL 语言的主要描述语句	194
14.1	顺序描述语句	194
14.1.1	WAIT 语句	195
14.1.2	断言(ASSERT)语句	198
14.1.3	信号代入语句	199
14.1.4	变量赋值语句	199
14.1.5	IF 语句	199
14.1.6	CASE 语句	202
14.1.7	LOOP 语句	207
14.1.8	NEXT 语句	209
14.1.9	EXIT 语句	210
14.2	并发描述语句	211
14.2.1	进程(PROCESS)语句	211
14.2.2	并发信号代入(Concurrent Signal Assignment)语句	212
14.2.3	条件信号代入(Conditional Signal Assignment)语句	213
14.2.4	选择信号代入(Selective Signal Assignment)语句	213
14.2.5	并发过程调用(Concurrent Procedure Call)语句	215
14.2.6	块(BLOCK)语句	216
14.2.7	GENERATE 语句	220
第 15 章	基本逻辑电路设计	224
15.1	组合逻辑电路设计	224
15.1.1	编码器和译码器	224
15.1.2	数据选择器	229
15.1.3	加法器	230
15.1.4	奇偶校验电路	232
15.1.5	三态门电路及总线缓冲器	234
15.2	时序电路设计	237
15.2.1	时钟信号和复位信号的描述	237
15.2.2	触发器和锁存器	238
15.2.3	寄存器	244

15.2.4 计数器及序列信号发生器	245
第 16 章 VHDL 语言程序的仿真、逻辑综合和下载	250
16.1 仿真	250
16.1.1 仿真输入信息的产生	250
16.1.2 仿真模块的编写	252
16.1.3 仿真 Δ	254
16.1.4 不同级别的仿真要求	255
16.2 逻辑综合	256
16.2.1 RTL 描述至未优化的布尔描述的转换	257
16.2.2 布尔优化描述	257
16.2.3 门级映射	258
16.3 VHDL 程序模块的下载	258
16.3.1 下载前的准备	258
16.3.2 配置	259
16.4 可编程逻辑器件介绍	261
16.4.1 可编程逻辑器件分类	261
16.4.2 可编程逻辑器件的几种典型结构	262
16.4.3 FPGA 和 CPLD 芯片介绍	265
习题	269
主要参考文献	286

第 1 篇 数字逻辑电路设计

数字逻辑电路设计篇主要介绍了数字电路的基本概念、基本原理和基本方法。本篇分为组合逻辑电路和时序电路两个部分，其内容共分 9 章。前 5 章介绍了常见的数制和编码，数字电路中所用的基本器件——逻辑门；讲述了数字技术的数学基础——逻辑代数及其化简，组合逻辑电路的分析与设计，常见的组合逻辑电路等。后面 4 章介绍了时序电路的基本单元——触发器；讲述了时序逻辑电路的分析和设计，常用的时序逻辑器件；介绍了脉冲产生电路等。本篇讲述的内容不仅是数字逻辑电路设计的基础，也是学习本书第二篇“数字系统设计基础”所必须要掌握的知识。

第 1 章

数制与编码

数字设备及计算机存在两种不同类型的运算：逻辑运算和算术运算。逻辑运算实际上是实现某种功能，而算术运算是对数据进行加工处理。在日常生活中，人们习惯于十进制数，而在数字系统中多采用二进制，有时也采用八进制和十六进制。

1.1 进位计数制

进位计数制是将数划分为不同的数位，按位进行累计，累计到一定数量后，向高位进位，同时又从零开始。

1.1.1 十进制

十进制是人们最熟悉的一种数制，组成十进制数的数字符号有 0, 1, 2, 3, …, 9，我们称这些数字符号为数码。一种数制的基本特征是它的基数。基数是指数制中表示数值的数码的个数。十进制有 0~9 这 10 个数码，所以它的基数为 10。十进制的计数规则是“逢十进一”。

数的表示法一般采用位置计数法。在一个数中，每一个数码和数码所在的位置决定了该数的大小，即每一个数码的位置载有该数大小的一个特定的数值，这个数值称为“权”。每个位置的权可以用基数的乘方表示。对十进制来说，基数为 10，所以位权就是 10^0 (个)， 10^1 (十)， 10^2 (百)， 10^3 (千)，…。例如，1998 这个十进制数用位置计数法可以表示为

$$(1998)_{10} = 1 \times 10^3 + 9 \times 10^2 + 9 \times 10^1 + 8 \times 10^0$$

下标 10 表示十进制 (也有用 D 来表示)，即以 10 为基数。包括整数和小数的十进制数的一般形式为

$$\begin{aligned} (N)_{10} &= (a_{n-1}a_{n-2}\cdots a_1a_0.a_{-1}a_{-2}\cdots a_{-m})_{10} \\ &= (a_{n-1} \times 10^{n-1} + a_{n-2} \times 10^{n-2} + \cdots + a_1 \times 10^1 + a_0 \times 10^0 \\ &\quad + a_{-1} \times 10^{-1} + a_{-2} \times 10^{-2} + \cdots + a_{-m} \times 10^{-m})_{10} \end{aligned}$$

式中， a_0 和 a_{-1} 之间的点是小数点， n 和 m 分别是整数的位数和小数的位数。

从计数电路的角度来看，采用十进制是不方便的。因为构成计数电路的基本想法是把电路的状态跟数码对应起来，而十进制的 10 个数码，必须由 10 个不同的而且能够严格区分的电路状态与之对应，这样将在技术上带来许多困难，不利于电路实现。因此，在计数

电路中一般不直接采用十进制。

1.1.2 二进制

二进制数与十进制数的区别在于码数的个数和进位的规律不同。二进制数用两个数码 0 和 1, 并且“逢二进一”。二进制数也采用位置计数法, 每位的权是 2 的幂(见表 1-1)。二进制数的一般形式为

$$\begin{aligned}(N)_2 &= (b_{n-1}b_{n-2}\cdots b_1b_0.b_{-1}b_{-2}\cdots b_{-m})_2 \\ &= (b_{n-1} \times 2^{n-1} + b_{n-2} \times 2^{n-2} + \cdots + b_1 \times 2^1 + b_0 \times 2^0 \\ &\quad + b_{-1} \times 2^{-1} + b_{-2} \times 2^{-2} + \cdots + b_{-m} \times 2^{-m})\end{aligned}$$

下标 2 表示二进制(也有用 B 来表示), 即以 2 为基数。

表 1-1 二进制各位的权

二进制位数	11	10	9	8	7	6	5	4	3	2	1
权	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
(十进制表示)	1 024	512	256	128	64	32	16	8	4	2	1
二进制位数	-1		-2		-3		-4		-5		-6
权	2^{-1}		2^{-2}		2^{-3}		2^{-4}		2^{-5}		2^{-6}
(十进制表示)	0.5		0.25		0.125		0.0625		0.031 25		0.015 625

例 1.1 $(11010.11)_2 = (1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2})_{10}$
 $= (26.75)_{10}$

二进制数的基数为 2, 只有 0、1 两个数字, 电路上只要两个状态与之对应, 便于电路实现。因此, 在数字系统中, 电路运算和处理的数采用的是二进制数。

1.1.3 八进制和十六进制

通常, 二进制数位数很多, 不便书写和记忆, 所以常采用以 2 的幂为基数的八进制数和十六进制数。

八进制的进位规则是“逢八进一”。每位数码用下列 8 个符号之一表示: 0, 1, 2, 3, 4, 5, 6, 7。

一个多位八进制数表示如下:

$$N = (345.67)_8$$

下标 O 表示八进制。其按权展开式为

$$N = 3 \times 8^2 + 4 \times 8^1 + 5 \times 8^0 + 6 \times 8^{-1} + 7 \times 8^{-2}$$

十六进制的进位规则是“逢十六进一”。每位数码用下列 16 个符号之一表示: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F。

一个多位十六进制数表示如下:

$$N = (19AF.EB)_H$$

下标 H 表示十六进制。其按权展开式为

$$N = 1 \times 16^3 + 9 \times 16^2 + A \times 16^1 + F \times 16^0 + E \times 16^{-1} + B \times 16^{-2}$$

1.2 数制转换

由于二进制电路实现容易,八进制和十六进制可以压缩字长,十进制人们十分熟悉,因此,在从事数字电路设计工作中,这几种数制都会用到。不同数制之间可以进行转换,如十进制数转换为二进制数,二进制数转换为十六进制数等。下面介绍这几种数制的转换问题。

1.2.1 十进制数和非十进制数之间的转换

下面以十进制数转换为二进制数为例,来说明十进制数转换为其它进制数的转换方法。

一个数可以用十进制数来表示,也可以用二进制数来表示。既然一个数可以用两种不同数制来表示,那么两者之间就必然有一定的转换关系。对于整数,可写成

$$(N)_{10} = b_n \times 2^n + b_{n-1} \times 2^{n-1} + \cdots + b_1 \times 2^1 + b_0 \times 2^0$$

式中, $b_n, b_{n-1}, \cdots, b_1, b_0$ 是二进制数各位的数字。为了得到 $b_n, b_{n-1}, \cdots, b_1, b_0$ 的值,将等式两边除以 2, 得到余数 b_0 , 将所得商再除以 2, 得到 $b_1, \cdots$, 依次连续进行, 直到商为 0, 就可由所有的余数求出二进制数。过程如下, 用短除法。

$$\begin{array}{r} 2 \overline{) b_n \times 2^n + b_{n-1} \times 2^{n-1} + \cdots + b_1 \times 2^1 + b_0 \times 2^0} \cdots \cdots b_0 \\ 2 \overline{) b_n \times 2^{n-1} + b_{n-1} \times 2^{n-2} + \cdots + b_2 \times 2^1 + b_1 \times 2^0} \cdots \cdots b_1 \\ \cdots \\ 2 \overline{) b_n \times 2^1 + b_{n-1} \times 2^0} \cdots \cdots b_{n-1} \\ 2 \overline{) b_n \times 2^0} \cdots \cdots b_n \end{array}$$

例 1.2 将 $(25)_{10}$ 按上述步骤转换成二进制数。

解

$$\begin{array}{r} 2 \overline{) 25} \cdots \cdots \text{余 } 1 \cdots \cdots b_0 \\ 2 \overline{) 12} \cdots \cdots \text{余 } 0 \cdots \cdots b_1 \\ 2 \overline{) 6} \cdots \cdots \text{余 } 0 \cdots \cdots b_2 \\ 2 \overline{) 3} \cdots \cdots \text{余 } 1 \cdots \cdots b_3 \\ 2 \overline{) 1} \cdots \cdots \text{余 } 1 \cdots \cdots b_4 \\ 0 \end{array}$$

所以 $(25)_{10} = (11001)_2$ 。

对于小数, 可写成

$$(N)_{10} = b_{-1} \times 2^{-1} + b_{-2} \times 2^{-2} + \cdots + b_{-(m-1)} \times 2^{-(m-1)} + b_{-m} \times 2^{-m}$$

为了得到 $b_{-1}, b_{-2}, \cdots, b_{-(m-1)}, b_{-m}$ 的值, 将上式两边分别乘以 2, 得到

$$2 \times (N)_{10} = b_{-1} \times 2^0 + b_{-2} \times 2^{-1} + \cdots + b_{-(m-1)} \times 2^{-(m-2)} + b_{-m} \times 2^{-(m-1)}$$

其个位数为 b_{-1} 。将每次十进制数乘以 2 所得积中的个位数去掉, 再进行乘 2, 如此下去, 直到满足误差要求进行“四舍五入”为止, 就可将十进制小数转换成二进制小数。

例 1.3 将 $(0.39)_{10}$ 转换成二进制小数。

解 $0.39 \times 2 = 0.78 \quad b_{-1} = 0$

$$\begin{array}{ll}
 0.78 \times 2 = 1.56 & b_2 = 1 \\
 0.56 \times 2 = 1.12 & b_3 = 1 \\
 0.12 \times 2 = 0.24 & b_4 = 0 \\
 0.24 \times 2 = 0.48 & b_5 = 0 \\
 0.48 \times 2 = 0.96 & b_6 = 0 \\
 0.96 \times 2 = 1.92 & b_7 = 1 \\
 0.92 \times 2 = 1.84 & b_8 = 1 \\
 0.84 \times 2 = 1.68 & b_9 = 1 \\
 0.68 \times 2 = 1.36 & b_{10} = 1 \\
 \dots &
 \end{array}$$

即 $(0.39)_{10} = (0.0110001111\dots)_2$ 。

此例中不能用有限位数实现准确转换。转换后的小数应取多少位合适？实际中常有两种方法，一是指定转换位数，如指定转换 8 位，则 $(0.39)_{10} = (0.01100011)_2$ ；二是根据转换精度确定位数。如此例要求转换精度优于 0.1%，即引入一个小于 $1/2^{10} = 1/1024$ 的舍入误差，则转换到第 10 位时，转换结束。

如果一个十进制数既有整数又有小数，应分别转换整数和小数部分，然后将其结果相加。

十进制数转换为八进制数或十六进制数的方法与十进制数转换成二进制数的方法相同。下面通过两个例子来说明转换的过程。

例 1.4 $(357)_{10} = (?)_8$

$$\begin{array}{rcl}
 \text{解} & 8 \overline{) 357} & \dots\dots\dots \text{余数 } 5 \dots\dots\dots o_0 = 5 \\
 & 8 \overline{) 44} & \dots\dots\dots \text{余数 } 4 \dots\dots\dots o_1 = 4 \\
 & 8 \overline{) 5} & \dots\dots\dots \text{余数 } 5 \dots\dots\dots o_2 = 5 \\
 & 0 &
 \end{array}$$

得到 $(357)_{10} = (545)_8$ 。

例 1.5 $(367)_{10} = (?)_{16}$

$$\begin{array}{rcl}
 \text{解} & 16 \overline{) 367} & \dots\dots\dots \text{余数 } 15 = \text{F} \dots\dots\dots h_0 = \text{F} \\
 & 16 \overline{) 22} & \dots\dots\dots \text{余数 } 6 \dots\dots\dots h_1 = 6 \\
 & 16 \overline{) 1} & \dots\dots\dots \text{余数 } 1 \dots\dots\dots h_2 = 1
 \end{array}$$

得到 $(367)_{10} = (16\text{F})_{16}$ 。

1.2.2 用二进制数表示八进制数和十六进制数

1. 二—八进制

八进制的基数是 2 的幂，因此二进制和八进制的互换是非常容易的。

1) 二进制转换成八进制

二—八进制，即用二进制方法表示八进制数。具体做法是将二进制数由小数点开始分别向左和向右每 3 位分成一组，每组便是一位八进制数。由二—八进制转换成八进制是很方便的。

例 1.6 二进制数为 110011101.011。

二一八进制数为 110 011 101.011,

八进制数为 635.3。

如果不能构成三位一组,则在二一八进制整数部分的高位上和小数部分的低位上通过补零来补足 3 位。

例 1.7 二进制数为 10011101.01,

二一八进制数为 010 011 101.010,

八进制数为 235.2。

2) 八进制转换成二进制

八进制数很容易转换成二进制数,先将八进制数转换成二一八进制数,再把二一八进制数转换成二进制数。

例 1.8 八进制数为 345.1,

二一八进制数为 011 100 101.001,

二进制数为 11100101.001。

2. 二一十六进制

十六进制的基数是 2 的幂,二进制和十六进制很容易互换。

1) 二进制转换成十六进制

二进制数由小数点开始,分别向左和向右每 4 位分成一组,每组便是一个十六进制数,这样的表示法叫二一十六进制,即用二进制表示的十六进制数。由二一十六进制转换成十六进制非常容易。

例 1.9 二进制数为 111101000.011,

二一十六进制数为 0001 1110 1000.0110,

十六进制数为 1E8.6。

2) 十六进制转换成二进制

利用二一十六进制,很容易将十六进制数转换成二进制数。

例 1.10 十六进制数为 AF.26,

二一十六进制数为 1010 1111.0010 0110,

二进制数为 10101111.0010011。

3. 二一十进制

为了用二进制表示十进制数中 0~9 这 10 个数码,需要 4 位二进制数。而 4 位二进制数有 16 个状态,因而有 6 个多余状态。以 4 位二进制数表示一位十进制数的数制叫二一十进制。这是用二进制码表示十进制数,简称 BCD 码(Binary Coded Decimal 的缩写)。它具有二进制数的形式,却又具有十进制数的特点。在表 1-2 中,十进制的 10 个数码与自然二进制数一一对应。在这种二一十进制中,从左向右每位的权分别是 8, 4, 2, 1, 所以称为 8421BCD 码。

自然,二一十进制很容易和十进制互相转换。

例 1.11 二一十进制数为 0011 1001 0101,

十进制数为 395。

例 1.12 十进制数为 583,

二一十进制数为 0101 1000 0011。