



应用型高等学校“十三五”规划教材

SHUZI DIANLU YU LUOJI SHEJI

数字电路与逻辑设计

主编 陈超原 张 静 单财良



华中科技大学出版社
<http://www.hustp.com>

应用型高等学校“十三五”规划教材

数字电路与逻辑设计

主 编 陈超原 张 静 单财良

副主编 陈振云 位 磊

华中科技大学出版社

中国·武汉

2019年4月

内 容 简 介

本书共 8 章和 4 个附录,分别为数字电路理论基础、逻辑门电路、组合逻辑电路、触发器、时序逻辑电路、存储器和可编程逻辑器件、脉冲波形的产生与整形电路、数/模及模/数转换技术、EDA 技术、Quartus Prime 17.1.0 软件使用简介、Verilog HDL 设计初步和部分习题参考答案等。

数字电路与逻辑设计是高等院校电子信息类各专业的一门重要的专业基础课。本书的特点是遵循“深入浅出、保证基础、精选内容、加强概念、联系实际”的原则,将数字电路与逻辑设计的理论与实际相结合,在论述重要概念与分析、设计方法后,通过大量的应用实例,让学生既能够掌握理论知识,又能够解决实际问题,以此帮助学生理解和巩固所学理论及知识。本书可以作为高等院校电子信息类各专业的教材,也可以作为相关专业工程技术人员的参考书。

图书在版编目(CIP)数据

数字电路与逻辑设计/陈超原,张静,单财良主编. —武汉:华中科技大学出版社,2019.8
ISBN 978-7-5680-5178-1

I. ①数… II. ①陈… ②张… ③单… III. ①数字电路-逻辑设计-教材 IV. ①TN79

中国版本图书馆 CIP 数据核字(2019)第 156630 号

数字电路与逻辑设计

Shuzi Dianlu yu Luoji Sheji

陈超原 张 静 单财良 主编

策划编辑:谢燕群

责任编辑:朱建丽

封面设计:原色设计

责任校对:李 琴

责任监印:徐 露

出版发行:华中科技大学出版社(中国·武汉)
武汉市东湖新技术开发区华工科技园

电话:(027)81321913
邮编:430223

录 排:武汉市洪山区佳年华文印部

印 刷:武汉华工鑫宏印务有限公司

开 本:787mm×1092mm 1/16

印 张:18.5

字 数:447 千字

版 次:2019 年 8 月第 1 版第 1 次印刷

定 价:45.00 元



本书若有印装质量问题,请向出版社营销中心调换
全国免费服务热线:400-6679-118 竭诚为您服务
版权所有 侵权必究

前 言

数字电路与逻辑设计是电子、电气、电信、通信、自动化、计算机等电子信息类专业一门重要的专业基础课。为适应现代电子技术快速发展的需要,我们根据多年的教学经验和实践积累,在系统介绍数字电路基本知识的基础上,同时对现代逻辑设计工具(EDA 技术)、Verilog 硬件描述语言等做了介绍,并以实际应用为目的编写了此书。

全书共 8 章和 4 个附录:第 1 章为数字电路理论基础,重点介绍了数制与码制、逻辑运算关系、逻辑函数及其化简方法、卡诺图等;第 2 章为逻辑门电路,重点介绍了半导体的开关特性、TTL 集成门电路的内部结构及使用特点;第 3 章为组合逻辑电路,重点介绍了组合逻辑电路的一般分析方法和设计方法及几种典型的组合逻辑器件的应用;第 4 章为触发器,介绍了常用触发器的类型及功能;第 5 章为时序逻辑电路,介绍了时序逻辑电路的一般分析方法和设计方法,并重点介绍了寄存器和计数器集成电路的应用;第 6 章为存储器和可编程逻辑器件,介绍了 ROM、RAM、PAL、GAL、FPGA 等器件的结构和应用;第 7 章为脉冲波形的产生与整形电路,重点介绍了 555 定时器的功能及由其组成的单稳态触发器、多谐振荡器、施密特触发器电路和应用;第 8 章为数/模及模/数转换技术,介绍了 DAC 和 ADC 的基本原理及典型集成电路;附录 A 为 EDA 技术,介绍了 EDA 基本知识;附录 B 为 Quartus Prime 17.1.0 软件使用简介;附录 C 为 Verilog HDL 设计初步,介绍了 Verilog HDL 程序的基本结构和特点;附录 D 为部分习题参考答案。根据不同专业的具体情况,教学时可以对本书进行内容节选。

本书在编写过程中注意循序渐进、难易结合、突出重点、突出实用、侧重能力培养。本书具有以下特色:

(1) 本书体现了一定的基础性和先进性,通过学习,使学生能够具有较为丰富的基础理论和基础知识,使学生具有可持续发展和创新的能力;

(2) 本书内容注重培养学生分析问题和解决问题的能力、综合运用所学知识的能力及工程实践的能力;

(3) 本书在选材和文字叙述上符合学生的认知规律,由浅入深、由简单到复杂、由基础知识到应用举例,并配有丰富的例题和习题。

本书由陈超原、张静、单财良、陈振云、位磊等 5 位老师编写,其中陈超原、张静和单财良任主编,陈振云和位磊任副主编。第 1 章由陈振云编写,第 2、3 章由单财良编写,第 4、5 章由陈超原编写,第 6 章由张静编写,第 7 章由位磊编写,第 8 章由陈振云和陈超原编写,附录 A 至附录 D 由单财良和张静编写。陈超原负责全书的统稿。

本书在编写过程中得到了学部领导俞侃教授的大力支持,电路系胡冬全副教授对全书的编写提出了很多宝贵的意见,编者在此表示衷心的感谢!

编者由于水平有限,书中难免有不足之处,恳请广大读者批评指正。

编 者

2019 年 4 月

目 录

第 1 章 数字电路理论基础	(1)
1.1 数制与编码	(1)
1.1.1 十进制数	(1)
1.1.2 二进制数	(1)
1.1.3 十六进制数	(3)
1.1.4 数制转换	(4)
1.1.5 有符号数的表示方法	(6)
1.1.6 十进制数的二进制编码	(8)
1.1.7 格雷码	(9)
1.2 二进制逻辑变量的电压表示	(10)
1.2.1 用电压表示二进制逻辑变量	(10)
1.2.2 正逻辑与负逻辑	(10)
1.3 基本逻辑运算和基本逻辑门	(11)
1.3.1 逻辑变量和逻辑函数	(11)
1.3.2 与运算及与门	(11)
1.3.3 或运算及或门	(12)
1.3.4 非运算及非门	(13)
1.3.5 复合逻辑及复合逻辑门	(13)
1.3.6 逻辑门符号	(16)
1.4 逻辑代数的基本定律及规则	(17)
1.4.1 逻辑代数的基本定律	(17)
1.4.2 逻辑代数的基本规则	(17)
1.5 逻辑函数的化简	(19)
1.5.1 逻辑函数的表示形式	(19)
1.5.2 逻辑函数的化简方法	(22)
本章小结	(26)
思考题	(26)
习题	(27)
第 2 章 逻辑门电路	(28)
2.1 半导体二极管门电路	(28)
2.1.1 二极管的开关特性	(28)
2.1.2 二极管门电路	(29)
2.2 半导体三极管门电路	(31)

2.2.1	三极管的开关特性	(31)
2.2.2	三极管反相器	(33)
2.3	TTL 集成门电路	(34)
2.3.1	TTL 逻辑门	(35)
2.3.2	CMOS 逻辑门	(44)
2.3.3	集成逻辑门的使用知识	(50)
	本章小结	(53)
	思考题	(54)
	习题	(54)
第 3 章	组合逻辑电路	(56)
3.1	组合逻辑电路概述	(56)
3.1.1	组合逻辑电路的特点	(56)
3.1.2	组合逻辑电路的逻辑功能概述	(56)
3.1.3	组合逻辑电路的类型、研究方法和任务	(57)
3.2	组合逻辑电路的分析	(57)
3.2.1	组合逻辑电路的分析步骤	(57)
3.2.2	组合逻辑电路的分析举例	(57)
3.3	组合逻辑电路的设计	(59)
3.3.1	组合逻辑电路的设计步骤	(60)
3.3.2	组合逻辑电路的设计举例	(60)
3.4	常用 MSI 组合逻辑电路及其应用	(67)
3.4.1	编码器	(67)
3.4.2	译码器	(74)
3.4.3	数据选择器	(86)
3.4.4	数值比较器	(92)
3.4.5	加法器	(94)
3.4.6	奇偶产生器/校验器	(100)
3.5	组合逻辑电路综合应用	(102)
3.5.1	简单投票系统	(102)
3.5.2	动态扫描显示电路	(103)
3.5.3	点阵显示控制电路	(105)
3.5.4	简单数据传输系统与检错电路	(106)
3.6	组合逻辑电路的竞争与冒险现象	(107)
3.6.1	竞争与冒险的概念及其产生的原因	(107)
3.6.2	消除竞争与冒险的方法	(109)
	本章小结	(111)
	思考题	(111)
	习题	(112)

第4章 触发器	(115)
4.1 基本触发器	(115)
4.2 同步触发器	(117)
4.2.1 同步 RS 触发器	(117)
4.2.2 同步 JK 触发器	(119)
4.2.3 同步 D 触发器	(120)
4.2.4 同步 T 触发器	(121)
4.3 边沿触发器	(122)
4.3.1 主-从 JK 触发器	(122)
4.3.2 主-从 D 触发器	(123)
4.3.3 维持-阻塞 D 触发器	(124)
4.4 不同触发器之间逻辑功能的转换	(126)
4.4.1 D 触发器构成 JK 触发器	(126)
4.4.2 JK 触发器构成 D 触发器	(127)
4.4.3 D 触发器构成 T 触发器	(127)
本章小结	(128)
思考题	(128)
习题	(128)
第5章 时序逻辑电路	(132)
5.1 时序逻辑电路的基本概念	(132)
5.1.1 时序逻辑电路的一般模型	(132)
5.1.2 时序逻辑电路的功能描述方法	(133)
5.2 时序逻辑电路的分析	(134)
5.2.1 同步时序逻辑电路的分析	(134)
5.2.2 异步时序逻辑电路的分析	(138)
5.3 时序逻辑电路的设计	(140)
5.3.1 同步时序逻辑电路的设计	(140)
5.3.2 异步时序逻辑电路的设计	(144)
5.4 寄存器	(147)
5.4.1 数码寄存器	(147)
5.4.2 移位寄存器	(147)
5.5 计数器	(151)
5.5.1 异步二进制加法计数器	(151)
5.5.2 同步二-十进制加法计数器	(153)
5.5.3 集成计数器	(154)
本章小结	(157)
思考题	(158)
习题	(158)

第 6 章 存储器和可编程逻辑器件	(162)
6.1 只读存储器	(162)
6.1.1 ROM 基本结构	(162)
6.1.2 译码与存储阵列	(164)
6.1.3 可编程 ROM	(165)
6.1.4 ROM 应用举例	(166)
6.2 随机存储器	(168)
6.2.1 RAM 的结构和工作原理	(168)
6.2.2 存储容量的扩展	(171)
6.3 简单可编程器件	(173)
6.3.1 可编程器件简介	(173)
6.3.2 PLD 的结构和表示方法	(175)
6.3.3 PLD 分类及设计举例	(177)
6.4 CPLD	(178)
6.4.1 CPLD 的结构框图	(179)
6.4.2 CPLD 的组成部分	(179)
6.5 FPGA	(181)
6.5.1 查找表结构基本原理	(181)
6.5.2 FPGA 结构简介	(182)
本章小结	(185)
思考题	(186)
习题	(186)
第 7 章 脉冲波形的产生与整形电路	(190)
7.1 概述	(190)
7.1.1 理想的矩形脉冲波形	(190)
7.1.2 实际的矩形脉冲波形	(190)
7.2 555 定时器	(191)
7.2.1 555 定时器的简介	(191)
7.2.2 555 定时器的特点	(192)
7.2.3 555 定时器的电路结构与工作原理	(192)
7.2.4 555 定时器电路的应用	(194)
7.3 单稳态触发器	(195)
7.3.1 单稳态触发器的工作特点	(195)
7.3.2 单稳态触发器的分类	(195)
7.3.3 555 定时器构成单稳态触发器	(195)
7.3.4 单稳态触发器的应用	(196)
7.4 多谐振荡器	(198)
7.4.1 多谐振荡器的结构特点	(198)
7.4.2 多谐振荡器的分类	(198)

7.4.3	由 555 定时器组成的多谐振荡器	(199)
7.5	施密特触发器	(200)
7.5.1	施密特触发器的特点	(200)
7.5.2	由 555 定时器组成的施密特触发器	(201)
7.5.3	施密特触发器的应用	(201)
7.6	实例电路分析	(203)
7.6.1	多用途延迟开关电源插座	(203)
7.6.2	基于 555 定时器及 CD4017 计数器的跑马灯	(204)
7.6.3	555 触摸定时开关	(206)
7.6.4	相片曝光定时器	(207)
7.6.5	单电源变双电源电路	(208)
7.6.6	简易催眠器	(208)
7.6.7	电热毯温控器	(208)
	本章小结	(209)
	思考题	(210)
	习题	(210)
第 8 章	数/模及模/数转换技术	(212)
8.1	数/模转换器	(212)
8.1.1	DAC 的基本原理	(212)
8.1.2	电压型 DAC	(214)
8.1.3	电流型 DAC	(216)
8.1.4	集成 DAC	(217)
8.1.5	DAC 的性能指标	(220)
8.2	模/数转换器	(221)
8.2.1	A/D 转换的概述	(221)
8.2.2	A/D 转换的基本原理	(221)
8.2.3	逐次渐近型 ADC	(223)
8.2.4	双积分型 ADC	(224)
8.2.5	并行比较型 ADC	(226)
8.2.6	ADC 的主要技术指标	(227)
8.2.7	ADC0809	(227)
	本章小结	(230)
	思考题	(230)
	习题	(231)
附录 A	EDA 技术	(232)
A.1	EDA 技术概述	(232)
A.2	EDA 技术及其发展	(232)
A.2.1	电子 CAD 阶段	(233)
A.2.2	电子 CAE 阶段	(233)

A.2.3 EDA 阶段	(233)
A.3 数字设计的流程	(234)
A.3.1 设计输入	(235)
A.3.2 综合	(236)
A.3.3 适配(布线布局)	(236)
A.3.4 仿真	(237)
A.3.5 编程配置	(237)
A.4 常用的 EDA 软件工具	(238)
附录 B Quartus Prime 17.1.0 软件使用简介	(239)
B.1 创建新工程	(239)
B.2 建立 Verilog HDL 文件	(244)
B.3 引脚配置	(245)
B.4 项目编译及结果分析	(247)
B.5 下载编程	(248)
附录 C Verilog HDL 设计初步	(249)
C.1 Verilog HDL 简介	(249)
C.1.1 Verilog HDL 的发展经历	(249)
C.1.2 Verilog HDL 的特点	(250)
C.2 Verilog HDL 模块的基本结构	(250)
C.3 Verilog HDL 的语言要素	(252)
C.3.1 基本语法定义	(252)
C.3.2 数据类型	(257)
C.4 Verilog HDL 的行为语句	(260)
C.4.1 过程语句	(261)
C.4.2 块语句	(263)
C.4.3 赋值语句	(264)
C.4.4 条件语句	(266)
C.4.5 循环语句	(269)
C.4.6 编译指示语句	(270)
C.5 Verilog HDL 设计的层次与风格	(271)
C.5.1 Verilog HDL 设计的层次	(271)
C.5.2 结构描述	(271)
C.5.3 行为描述	(276)
C.5.4 数据流描述	(277)
C.5.5 分层次结构电路的设计	(278)
附录 D 部分习题参考答案	(281)
参考文献	(285)

第1章 数字电路理论基础

二进制数码用0和1表示,八进制数码用0~7表示,十六进制数码用0~9、A~F表示。通过码和权实现数制间的转换。常见的编码有8421 BCD码、余3码、格雷码等。使用与、或、非三种基本运算可以构成逻辑运算的基本定律和各种复杂的逻辑运算表达式。逻辑运算有代入、对偶、反演规则。逻辑函数化简有代数法和卡诺图法。逻辑问题由真值表、函数式、卡诺图、波形图、逻辑图表达。

1.1 数制与编码

人们日常生活中经常使用的数的表示方法是十进制数,也就是用0~9这10个符号表示0~9的数,比9大的数,采取逢十进位的办法。所谓数制就是表示数值所用的数字符号的个数。

此外,我们生活中还经常用到很多其他进制的数,例如,1小时等于60分钟,1分钟等于60秒,采用的都是六十进制数。而一天等于24小时采用的是二十四进制数。

在计算机领域,如果采用我们最习惯的十进制数,就需要10个符号来表示数字0~9,不方便存储和物理实现,因此计算机中采用的是只需要两个符号的二进制数。

我们在本节中以整数为例来讨论各种数制的定义及数制之间的转换等内容。

1.1.1 十进制数

十进制数是用0~9这10个符号表示0~9的数,对于比9大的数,采用逢十进位的办法。例如,十进制数365读作“三百六十五”,3表示300,6表示60,5表示5,即

$$365_{10} = 3 \times 10^2 + 6 \times 10^1 + 5 \times 10^0$$

上式左端的下标10表示365是十进制数。右端的 10^2 、 10^1 和 10^0 称为位权,即 10^2 是百位的权值, 10^1 是十位的权值, 10^0 是个位的权值。10称为十进制数的基数。位权和基数是进位计数制的两个基本概念。任意一个十进制整数可以按位权展开成如下的多项式:

$$a_{n-1} \times 10^{n-1} + a_{n-2} \times 10^{n-2} + \dots + a_0 \times 10^0$$

例 1-1-1 写出十进制数7685的按位权展开的多项式。

解 $7685_{10} = 7 \times 10^3 + 6 \times 10^2 + 8 \times 10^1 + 5 \times 10^0$

1.1.2 二进制数

与十进制数的定义方法相同,二进制数仅有0和1两个数字符号,可以直接表示0和1两个数值,数值2就必须进位,表示成10,读作“壹零”,也就是说二进制数的10意味着:

$$10_2 = 1 \times 2^1 + 0 \times 2^0$$

这样,数值3表示成11。而数值4就必须再进一位,表示成100,以此类推。任意一个

二进制整数,可以按位权展开成如下的多项式:

$$a_{n-1} \times 2^{n-1} + a_{n-2} \times 2^{n-2} + \cdots + a_0 \times 2^0$$

二进制数的基数是2,即逢二进位, $2^{n-1}, 2^{n-2}, \cdots, 2^0$ 是二进制数的位权。在二进制数中,数字通常称为“位”。在8位二进制数中,位权为 2^0 的位称为最低有效位,位权为 2^7 的位称为最高有效位。若将 $a_{n-1} \times 2^{n-1} + a_{n-2} \times 2^{n-2} + \cdots + a_0 \times 2^0$ 的加法按十进制加法去求和,就能求得与二进制数相对应的十进制数,举例如下。

$$1011_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 8 + 0 + 2 + 1 = 11_{10}$$

从式中可以看出,从二进制数转换成十进制数的方法。

例 1-1-2 写出二进制数101111的按位权展开式,并求出相应的十进制数。

$$\begin{aligned} \text{解 } 101111_2 &= 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\ &= 32 + 0 + 8 + 4 + 2 + 1 = 47_{10} \end{aligned}$$

从上面的例子可以看出,为了转换快捷,记住2的各个幂次就会节约大量的时间。除此之外,经常会用到前16个二进制正整数,如表1-1-1所示。

表 1-1-1 前16个二进制正整数

十进制数	二进制数	十进制数	二进制数
0	0	8	1000
1	1	9	1001
2	10	10	1010
3	11	11	1011
4	100	12	1100
5	101	13	1101
6	110	14	1110
7	111	15	1111

与十进制数一样,在二进制整数前面添加0不会影响数值的大小。因此,如果要求二进制数的位数是4,那么在整数前面添加0以补足4位即可。

例 1-1-3 用4位二进制数列出前16个二进制正整数。

解 用4位二进制数表示的前16个二进制正整数如表1-1-2所示。

表 1-1-2 4位二进制数的表示

十进制数	二进制数	十进制数	二进制数
0	0000	8	1000
1	0001	9	1001
2	0010	10	1010
3	0011	11	1011
4	0100	12	1100
5	0101	13	1101
6	0110	14	1110
7	0111	15	1111

1.1.3 十六进制数

由于二进制数的位数一般较多,而且只有0和1两个数字符号,这就给书写和记忆带来了一定的麻烦,比较容易出错。因此,在数字系统中,十六进制数通常作为二进制数的速记符号。

十六进制的定义与前面学过的十进制和二进制的定义完全一致,如表1-1-3所示。

表 1-1-3 十六进制的定义

基 数		符号个数	数 字 符 号	位 权
十进制	10	10	0~9	10^n
二进制	2	2	0,1	2^n
十六进制	16	16	0~9, A~F	16^n

表1-1-3特别强调的是十六进制的符号有16个,包括0~9这10个数字符号和A~F这6个字母符号。前16个十六进制正整数与十进制数、二进制数的对应关系如表1-1-4所示。

表 1-1-4 前16个十六进制正整数与十进制数、二进制数的对应关系

十 进 制 数	二 进 制 数	十 六 进 制 数
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

任意一个十六进制正整数,可以按位权展开成如下的多项式:

$$a_{n-1} \times 16^{n-1} + a_{n-2} \times 16^{n-2} + \cdots + a_0 \times 16^0$$

其中,字母 A~F 与十进制数有着对应关系,在计算时字母 A~F 用相应的十进制数值代替。十六进制数的读法与二进制数的读法相同,即按照从高位到低位的顺序读出相应数字,例如,2A45₁₆ 读作“十六进制数二 A 四五”。

例 1-1-4 写出十六进制数 1C3₁₆ 的按位权展开式,并求出相应的十进制数。

解 $1C3_{16} = 1 \times 16^2 + 12 \times 16^1 + 3 \times 16^0 = 1 \times 256 + 12 \times 16 + 3 \times 1 = 451_{10}$

需要说明的是,二进制数是唯一可以在计算机系统中直接运算的数制,它是数字系统的基础,是除了十进制数之外我们必须熟练掌握的。十六进制数是二进制数的一种书写形式,对书写和记忆而言都比二进制数的容易,因此其应用非常广泛。

1.1.4 数制转换

不同的数制只是以不同的形式来表示数据,例如,十进制数 15₁₀ 表示成二进制数的形式是 1111₂,表示成十六进制数的形式是 F₁₆,这些不同的表示形式在数值上是相等的,即

$$15_{10} = 1111_2 = F_{16}$$

这里再用一个日常生活中比较常见的例子来帮助理解这一概念。假如一个班上有 15 个学生,是用我们已经习惯了的十进制形式来计数的;若以二进制形式计数,则这个班上有 1111 个学生;或者若以十六进制形式计数,则这个班上有 F 个学生。这三种表达形式所表示的都是同一个客观存在,即这个班的学生人数,而学生人数是一个固定值,因此这三种计数形式表示的数值是相等的,只不过我们在日常生活中不习惯使用十进制以外的计数形式而已。

在学习过程中,如果遇到了采用某一种计数形式表示的数,我们应该能够根据需要把它等值转换成其他进制的数。

1. 二进制数、十六进制数转换成十进制数

将二进制数、十六进制数转换成十进制数的方法是:将要转换的数写成按位权展开的多项式,计算该多项式的值就可以得到相应的十进制数值按位权展开的多项式。

例 1-1-5 将二进制正整数 10100101₂ 转换成十进制数。

解 $10100101_2 = 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$
 $= 128 + 0 + 32 + 0 + 0 + 4 + 0 + 1 = 165_{10}$

在二进制数转换成十进制数的过程中,凡是 0 的位,不论它的位权值有多大,最终参与加法运算的都是 0,不会影响转换的结果,也就是说,只有 1 的位及该位的位权值决定转换的结果。因此可以直接计算二进制数中的 1 的位权值的和,就可以得到相应的十进制数。本题结果也可以写成:

$$10100101_2 = 2^7 + 2^5 + 2^2 + 2^0 = 165_{10}$$

例 1-1-6 将十六进制正整数 FFFF₁₆ 转换成十进制数。

解 $FFFF_{16} = 15 \times 16^3 + 15 \times 16^2 + 15 \times 16^1 + 15 \times 16^0$
 $= 15 \times 4096 + 15 \times 256 + 15 \times 16 + 15 \times 1 = 65535_{10}$

2. 十进制数转换成二进制数、十六进制数

十进制整数转换成二进制数通常采用除 2 取余法。基本方法是,将被转换的整数除以 2

取余数,商再继续除以2,再取余数,直到商为0为止,然后将余数按照先取的作为低位、后取的作为高位的顺序写出来,就得到转换后的二进制数。

例 1-1-7 将十进制数 105_{10} 转换成二进制数。

解 采用除2取余法。假定转换后的二进制数位用 b 表示, b_0 表示最低位, b_1 表示次低位,以此类推,转化过程如下。

2	105	余1	$b_0=1$
2	52	余0	$b_1=0$
2	26	余0	$b_2=0$
2	13	余1	$b_3=1$
2	6	余0	$b_4=0$
2	3	余1	$b_5=1$
2	1	余1	$b_6=1$
	0		

即

$$105_{10} = 1101001_2$$

十进制小数转换成二进制数的方法是:整数部分与前述相同;小数部分通常采用乘2取整法。基本方法是,将被转换的小数乘以2取整数,余下的小数再继续乘以2,再取整数,直到满足要求为止,然后将小数部分所求的整数按照先取的作为高位、后取的作为低位的顺序写出来,就得到转换后的二进制小数。

例 1-1-8 将十进制小数 0.625 转换成二进制数。

解 0.625×2 取整数 1 高位

0.250×2 取整数 0

0.50×2 取整数 1 低位

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

十进制数转换成十六进制数采用除16取余法。具体运算过程与除2取余法的相同。

例 1-1-9 将十进制数 254_{10} 转换成十六进制数。

解

16	254	余14	$b_0=E$
16	15	余15	$b_1=F$
	0		

即

$$254_{10} = FE_{16}$$

3. 二进制数与十六进制数之间的相互转换

由于 $16=2^4$,因此1位十六进制数相当于4位二进制数,这样十六进制数与二进制数之间的相互转换是十分方便的。

二进制数转换成十六进制数的方法是:从二进制数的最低位开始,每4位分成一组,若

最高位的一组不足 4 位,则在其左边添加 0 以补足 4 位,然后用每组 4 位二进制数所对应的十六进制数取代该组的 4 位二进制数,就可以得到对应的十六进制数。其中,4 位二进制数与十六进制数有着对应的关系。

例 1-1-10 将二进制数 1011101001_2 转换成十六进制数。

解 将二进制数 1011101001 从最低位开始,每 4 位分成一组,由于最高为一组只有 2 位,因此在其左边添加两个 0 以补足 4 位,分组后如下:

二进制数	0010	1110	1001
十六进制数	2	E	9

即

$$1011101001_2 = 2E9_{16}$$

例 1-1-11 将十六进制数 $5C7A_{16}$ 转换成二进制数。

十六进制数	5	C	7	A
二进制数	0101	1100	0111	1010

即

$$5C7A_{16} = 0101110001111010_2$$

注意:转换结果中最高位的 0 可以省略,不影响数值大小,但中间的 0 绝对不可以省略,例如,十六进制数 7 对应二进制数 0111,由于 7 在十六进制数的中间,因此高位的 0 不能省略,也就是说,每个十六进制数位对应的二进制数必须是 4 位。

本题结果也可以写成:

$$5C7A_{16} = 101110001111010_2$$

1.1.5 有符号数的表示方法

前面介绍的二进制数均为无符号数,所有二进制数位均为数值位。然而实际的数值带有符号,既可能是正数,也可能是负数,这样就存在一个有符号二进制数的表示方法问题。

1. 数的符号表示法

为了表示一个有符号数,除了数值位以外还应指定符号位,通常以这个数的最高位为符号位。假定讨论的数为整数。对 8 位有符号二进制整数, D_7 为符号位,并规定 D_7 为 0 表示正数的符号“+”;为 1 表示负数的符号“-”。因此,8 位有符号数的编码格式为:

$$D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0$$

这样,数的符号在机器中也就数字化了,符号位和数值位一起完整地表示有符号的二进制数。一个数在机器中的表示形式称为机器数,而其实际数值称为机器数的真值。

2. 原码表示法

如果正数的符号位用 0 表示,负数的符号位用 1 表示,数值位的编码规则与前面讨论的无符号数编码规则相同,这种表示方法称为原码表示法。

例 1-1-12 有两个有符号的十进制数,用 8 位二进制原码表示 $X_1 = -20$ 和 $X_2 = +32$ 。

解 $X_1 = -20_{10} =$ 符号位 1 数值位 0010100

$X_2 = +32_{10} =$ 符号位 0 数值位 0100000

也可以写成:

$$[X_1]_{\text{原}} = 10010100, \quad [X_2]_{\text{原}} = 00100000$$

其中的下标“原”表示原码。

数0的原码有两种表示形式,即

$$[+0]_{\text{原}} = 00000000, \quad [-0]_{\text{原}} = 10000000$$

8位二进制原码所能表示的数值范围为 $-127 \sim +127$ 。

有符号数的原码表示既简单又易于理解,与真值转换方便,也解决了有符号数的正负符号在计算机中的表示问题,这是它的优点。它的缺点是原码在进行加减运算时非常麻烦,不仅要考虑数的符号,还要考虑数值大小,这使运算器设计非常复杂,因此又引入了有符号数的补码表示法,它可以使正负数的加法和减法运算简化为一种加法运算。

3. 补码表示法

正数的补码与原码相同,因此对于正数 X 而言,下面式子成立:

$$[X_2]_{\text{原}} = [X_2]_{\text{补}}$$

负数的补码可以从原码求得:将负数原码除符号位之外的数位按位取反,即1变0,0变1,再在最低位上加1,就可以得到该数的补码。

例 1-1-13 求十进制数 -105 的补码。

解 $[-105]_{\text{原}} = 11101001$

其原码为11101001,将原码除符号位外各位取反得10010110;在其最低位加1,得到补码10010111,即

$$[-105]_{\text{补}} = 10010111$$

值得注意的是,0的补码只有唯一的形式,符号位和数值位均为0。8位二进制数所能表示的数值范围为 $-128 \sim +127$ 。8位二进制数对应的无符号二进制数、有符号二进制数原码及补码如表1-1-5所示。

表 1-1-5 8位二进制数的表示

二进制数码	无符号二进制数	有符号数原码	有符号数补码
00000000	0	+0	0
00000001	1	+1	+1
00000010	2	+2	+2
...	...	...	...
01111110	126	+126	+126
01111111	127	+127	+127
...	...	...	...
11111101	253	-125	-3
11111110	254	-126	-2
11111111	255	-127	-1