

普通高等教育“十五”国家级规划教材配套参考书

数字电子技术基础 学习指导书

卫桦林 主编

卫桦林 郭顺华 编



高等教育出版社
HIGHER EDUCATION PRESS

普通高等教育“十五”国家级规划教材配套参考书

数字电子技术基础 学习指导书

卫桦林 主编
卫桦林 郭顺华 编

高等教育出版社

内容提要

本书是为杨志忠主编的“十五”国家级规划教材《数字电子技术基础》编写的学习指导书。内容包括《数字电子技术基础》教材各章的基本要求,重点难点,内容提要及例题分析。目的在于帮助学生在课程学习中开拓思想,提高学习效率,以便更好掌握教材内容。

本书可供普通高等教育的电气、电子、通信、计算机、自动化等专业的学生使用,也可供有关工程技术人员和自学人员参考。

图书在版编目(CIP)数据

数字电子技术基础学习指导书/卫桦林主编. —北京:高等教育出版社, 2004.7

ISBN 7-04-014548-0

I. 数… II. 卫… III. 数字电路-电子技术-高等学校-教学参考资料 IV. TN79

中国版本图书馆 CIP 数据核字 (2004) 第 050803 号

策划编辑 韩 颖 责任编辑 章浩平 封面设计 于 涛 责任绘图 朱 静
版式设计 王艳红 责任校对 金 辉 责任印制 孔 源

出版发行 高等教育出版社
社 址 北京市西城区德外大街 4 号
邮政编码 100011
总 机 010-82028899

购书热线 010-64054588
免费咨询 800-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>

经 销 新华书店北京发行所
印 刷 北京铭成印刷有限公司

开 本 787×960 1/16
印 张 10.75
字 数 190 000

版 次 2004 年 7 月第 1 版
印 次 2004 年 7 月第 1 次印刷
定 价 13.60 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

前 言

本书是为配合南京工程学院杨志忠教授主编的《数字电子技术基础》教材而编写的学习指导书，内容包括《数字电子技术基础》教材各章的基本要求、重点难点、内容提要 and 例题分析。

本书归纳了教材各章的基本要求，重点和难点，并着重对各章的基本内容作了简要概括，使学生在阅读本书后，对教材各章内容有一个整体的认识和理解，达到提纲挈领的目的。本书针对各章的重点，选择典型例题进行分析，其中部分题目是往年一些国内重点高校的研究生入学试题，目的在于启发学生思路，灵活运用所学知识，提高解题能力和应试水平。

本书由卫桦林、郭顺华和华沙同志编写。华沙编写了第7章。郭顺华编写了第3、第8、第9章，其余各章由卫桦林编写。卫桦林任主编，负责全书的组织和统稿。章忠全副教授担任主审，对全书进行了认真仔细的审阅，提出了许多宝贵建议。杨志忠教授在本书的编写中也给予了热情的支持和帮助，在此表示衷心的感谢。本书中的大部分插图由卫羽佶、卫羽倩负责绘制，罗中燕同志帮助收集整理资料，在此一并致谢。

由于编者水平有限，时间仓促，书中难免存在错漏和不妥之处，恳请各位读者批评斧正。

编 者

2004年3月

目 录

第 1 章 绪论	1
§ 1.1 基本要求	1
§ 1.2 重点难点	1
§ 1.3 内容提要	1
§ 1.4 例题分析	8
第 2 章 逻辑代数基础	11
§ 2.1 基本要求	11
§ 2.2 重点难点	11
§ 2.3 内容提要	11
§ 2.4 例题分析	19
第 3 章 集成逻辑门电路	27
§ 3.1 基本要求	27
§ 3.2 重点难点	27
§ 3.3 内容提要	27
§ 3.4 例题分析	36
第 4 章 组合逻辑电路	40
§ 4.1 基本要求	40
§ 4.2 重点难点	40
§ 4.3 内容提要	40
§ 4.4 例题分析	55
第 5 章 集成触发器	69
§ 5.1 基本要求	69
§ 5.2 重点难点	69
§ 5.3 内容提要	69

§ 5.4 例题分析	76
第 6 章 时序逻辑电路	82
§ 6.1 基本要求	82
§ 6.2 重点难点	82
§ 6.3 内容提要	82
§ 6.4 例题分析	89
第 7 章 脉冲产生与整形电路	109
§ 7.1 基本要求	109
§ 7.2 重点难点	109
§ 7.3 内容提要	109
§ 7.4 例题分析	116
第 8 章 数模和模数转换器	126
§ 8.1 基本要求	126
§ 8.2 重点难点	126
§ 8.3 内容提要	126
§ 8.4 例题分析	132
第 9 章 半导体存储器	136
§ 9.1 基本要求	136
§ 9.2 重点难点	136
§ 9.3 内容提要	136
§ 9.4 例题分析	142
第 10 章 可编程逻辑器件	146
§ 10.1 基本要求	146
§ 10.2 重点难点	146
§ 10.3 内容提要	146
§ 10.4 例题分析	155

第 1 章 绪 论

§ 1.1 基本要求

- (1) 模拟信号、模拟电路与数字信号、数字电路的区别。
- (2) 常用数制及其相互转换。
- (3) 码制与常用编码。

§ 1.2 重点难点

重点：(1) 数字信号的概念，矩形脉冲波形的参数。

(2) 计数制中权的概念，十进制数与二进制数、八进制数、十六进制数的相互转换。

(3) 编码的概念与常用编码。

难点：数字信号的不连续概念。

§ 1.3 内容提要

一、模拟信号、模拟电路和数字信号、数字电路的区别

模拟信号：在时间上和数值上均连续的信号称为模拟信号。最简单的模拟信号是正弦波信号，它是时间的连续函数，同时它的瞬时值在 $-U_m \sim +U_m$ 之间是数值连续的，即可以取到 $-U_m \sim +U_m$ 之间的连续数值。

模拟电路：能够处理(包括产生、放大、传输、运算等)模拟信号电子电路称为模拟电子电路。

数字信号：数字信号是用二进制数表示连续信号在某一时刻瞬时值大小的数字量，它是时间上和数值上均不连续的信号。时间不连续的含义，是它只对连续时间信号在有限个取样点处的瞬时值大小，因此在时间上是不连续的；数值不连续的含义是表示某个样本值的二进制数必须是量化单位 Δ 的整数倍数值(非整数部分被舍去)，因此在数值上是不连续的。

数字信号波形：由于二进制数是由 0 和 1 两个数码组成，在数字电路中，若采用正逻辑定义，低电平用 0 表示，高电平用 1 表示，则数字信号波形就是由一系列对应于时钟信号周期的高电平、低电平组成的矩形脉冲波形。如图 1.3.1 所示， Q 是对应于 CP 的二进制数字信号 01011010 的波形。

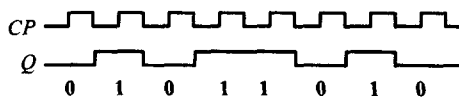


图 1.3.1 数字信号波形

数字信号波形的特点：

- (1) 基本形状为矩形脉冲；
- (2) 矩形脉冲的幅度不变；
- (3) 低、高电平的宽度由具体的二进制数确定。

数字电路：能够处理(包括产生,传输,运算等)数字信号电子电路称为数字电子电路。

二、矩形脉冲信号的主要参数

由于晶体管开关的非理想性，数字信号在实际传输过程中都不是理想的矩形脉冲，为了描述实际矩形脉冲信号的特性，定义了实际矩形脉冲波形的一些特性参数，如图 1.3.2 所示。它们主要有：

- (1) 上升沿时间 t_r ；
- (2) 下降沿时间 t_f ；
- (3) 有效宽度 t_w ；
- (4) 脉冲的幅度；
- (5) 周期 T 和占空比 q

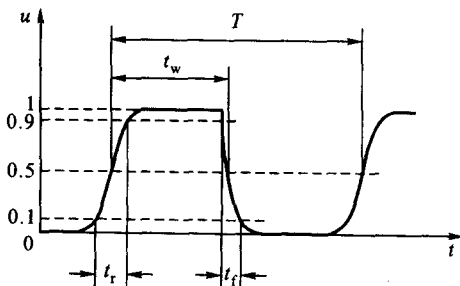


图 1.3.2 矩形脉冲波形的主要参数

$$\text{占空比 } q = t_w / T$$

影响数字电路工作速度的主要参数是 t_r 和 t_f ， t_r 和 t_f 越小，电路工作速度就越快。

三、数制及其相互转换

1. 计数制的基数和权

基数：计数制中计数符号(又称为数码)的个数称为基数，这里用 K 表示。

基数 K 决定了计数制的进位规则, 如十进制数有 $0 \sim 9$ 十个数码, 故 $K = 10$, 进位规则为逢十进一; 二进制数只有 0 和 1 两个数码, 故 $K = 2$, 进位规则为逢二进一; 十六进制数有 $0 \sim 9$ 、 A 、 B 、 C 、 D 、 E 、 F 十六个数码, 故 $K = 16$, 进位规则为逢十六进一。

权: 表征不同数位上数码的值, 称为位权, 简称权。权是基数的整数幂。如十进制数中, 整数部分个位的权是 10^0 , 十位的权是 10^1 , 百位的权是 10^2 , 等等; 小数部分第一位小数的权是 10^{-1} , 第二位小数的权是 10^{-2} , 等等。一般地说, 对于任一个计数制的数 $N_K = a_{n-1}a_{n-2}\cdots a_1a_0.a_{-1}a_{-2}\cdots a_{-m}$, 它们的权从高位到低位依次是 K^{n-1} , K^{n-2} , $\cdots$, K^1 , K^0 , K^{-1} , K^{-2} , $\cdots$, K^{-m} 。

2. 数的按权展开式

根据权的定义, 对于任意 K 进制数 $N_K = a_{n-1}a_{n-2}\cdots a_1a_0.a_{-1}a_{-2}\cdots a_{-m}$, 可以写出以下展开式

$$\begin{aligned} N_K &= a_{n-1}K^{n-1} + a_{n-2}K^{n-2} + \cdots + a_1K^1 + a_0K^0 \\ &\quad + a_{-1}K^{-1} + a_{-2}K^{-2} + \cdots + a_{-m}K^{-m} \\ &= \sum_{i=-m}^{n-1} a_i K^i \end{aligned}$$

对于上式需要说明一点, 权是基数的整数次幂, 各位数码的权进行整数次幂运算后的值都是十进制数。如某一位二进制数的权是 2^3 , 则 $2^3 = 8$ 是十进制数。

3. 数制的转换

数制的转换是把数由一种计数制的等值转换成另一种计数制。

(1) 任意 K 进制数 N_K 转换为十进制数

转换方法: 把 N_K 展开式中的每位数码用等值的十进制数替换后, 计算出的数值就是与 N_K 等值的十进制数。

[例 1.3.1] 将二进制数 **10110** 转换为十进制数。

$$\begin{aligned} \text{解: } (10110)_2 &= 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\ &= 16 + 0 + 4 + 2 + 0 = (22)_{10} \end{aligned}$$

对于十进制数, 其下标可以不写。

[例 1.3.2] 将十六进制数 **2FA** 转换为十进制数。

$$\begin{aligned} \text{解: } (2FA)_{16} &= 2 \times 16^2 + 15 \times 16^1 + 10 \times 16^0 \\ &= 512 + 240 + 10 = 762 \end{aligned}$$

(2) 十进制数转换为任意 K 进制数 N_K

转换分为整数部分转换和小数部分转换。

整数部分转换方法: 除基取余法。取到的最大余数是 $K - 1$, 最小余数是 0 。具体做法是用基数 K 连除十进制数, 每除一次得到一个余数, 直到商为 0 。然后把余数按先得到的余数为低位 (LSB), 后得到的余数为高位 (MSB) 的顺序

排列成一行, 就得到 K 进制整数。

小数部分转换方法: 乘基取整法。取到的最大整数是 $K-1$, 最小整数是 0。具体做法是用基数 K 连乘十进制小数, 每乘一次取得 1 位整数, 直到小数变为 0, 然后按先取到的整数为高位小数(小数点后第 1 位), 后取到的为低位小数的顺序排列成一行, 就得到 K 进制小数。若遇到小数乘不尽的情况, 则根据精度要求保留到某一位小数即可。

[例 1.3.3] 将十进制数 139 转换为二进制数和十六进制数。

解: (1) 转换成二进制数, 用除基取余法连除基数 2。即

$$\begin{array}{rcl}
 2 \overline{)139} & & \text{余数} \\
 2 \overline{)69} \quad \cdots & & 1 \quad \leftarrow \text{LSB} \\
 2 \overline{)34} \quad \cdots & & 1 \\
 2 \overline{)17} \quad \cdots & & 0 \\
 2 \overline{)8} \quad \cdots & & 1 \\
 2 \overline{)4} \quad \cdots & & 0 \\
 2 \overline{)2} \quad \cdots & & 0 \\
 2 \overline{)1} \quad \cdots & & 0 \\
 0 \quad \cdots & & 1 \quad \leftarrow \text{MSB} \\
 (139)_{10} = (10001011)_2
 \end{array}$$

(2) 转换为十六进制数。用除基取余法连除基数 16。即

$$\begin{array}{rcl}
 16 \overline{)139} & & \text{余数} \\
 16 \overline{)8} \quad \cdots & & \text{B(11)} \quad \text{LSB} \\
 0 \quad \cdots & & 8 \quad \text{MSB} \\
 (139)_{10} = (8\text{B})_{16}
 \end{array}$$

[例 1.3.4] 将十进制小数 0.78125 转换为二进制小数和十六进制小数。

解: (1) 转换为二进制小数, 用基数 2 连乘小数。即

$$\begin{array}{rcl}
 \text{乘 2} & & \text{取整} \\
 0.78125 \times 2 = 1.5625 & \cdots & 1 \quad \leftarrow \text{小数点后第一位} \\
 0.5625 \times 2 = 1.125 & \cdots & 1 \\
 0.125 \times 2 = 0.25 & \cdots & 0 \\
 0.25 \times 2 = 0.5 & \cdots & 0 \\
 0.5 \times 2 = 1.0 & \cdots & 1 \quad \leftarrow \text{小数点后最后一位} \\
 (0.78125)_{10} = (0.11001)_2
 \end{array}$$

(2) 转换为十六进制小数, 用基数 16 连乘小数。即

$$\begin{array}{rcl}
 \text{乘 16} & & \text{取整} \\
 0.78125 \times 16 = 12.5 & \cdots & \text{C(12)}
 \end{array}$$

$$0.5 \times 16 = 8.0 \quad \cdots \quad 8$$

$$(0.78125)_{10} = (0.C8)_{16}$$

(3) 二进制数和十六进制数、八进制数的相互转换

转换方法：数码等值代换法。具体做法是用4位二进制数与一位等值十六进制数直接进行替换，即实现二进制数与十六进制数之间的相互转换。如果用3位二进制数与1位等值的八进制数直接进行替换，即实现二进制数与八进制数之间的相互转换。

【例 1.3.5】将数 $(1011010.11)_2$ 转换为十六进制数和八进制数。

解：(1) 转换为十六进制数。首先把二进制数以小数点为界，分别向左、向右每4位分为一节，整数高位的缺位和小数低位的缺位补0。然后用等值的十六进制数码替换每一节二进制数即可

$$(01011010.1100)_2 = \underbrace{0101}_5 \underbrace{1010}_{10} \underbrace{.1100}_{12} = (5A.C)_{16}$$

(2) 转换为八进制数，与转换为十六进制数类似，转换如下

$$(1011010.11)_2 = \underbrace{001}_1 \underbrace{011}_3 \underbrace{010}_2 \underbrace{.110}_6 = (132.6)_8$$

本例的逆向转换就是十六进制数、八进制数转换为二进制数。

4. 二进制数的加、减运算

二进制加运算按逢二进一规则运算，减法运算按借一当二规则运算。但在数字电路中要实现二进制数的加、减运算，为避免将电路设计得太复杂，往往采用补码的形式进行二进制数的加、减运算，这样可以利用加法运算电路实现减法运算，从而节省电路的硬件成本。

用加法实现减法运算，就是把减数看作有符号数，然后与被减数相加（类似于代数中的求代数和运算），但是不能采用原码求和，而必须采用补码求和。若求得和为正数，其和就是两数之差的原码；若求得和为负数，其和为两数之差的补码，将结果再求补一次，就得到两数之差的原码。

(1) 有符号二进制数的表示法

有符号二进制数是用符号位表示数的正负的。在二进制数的最高位前面加一位符号位，若符号位为0，则表示二进制数为正数；若符号位为1，则表示二进制数为负数。因此有符号数在书写时的最高位是符号位，其后的各位是数值位。

(2) 二进制数的原码、反码、补码

对于数 $X > 0$ ，如 $X = +1001$ ，则有符号数表示为

$$[X]_{\text{原}} = [X]_{\text{反}} = [X]_{\text{补}} = 01001$$

最高位的 0 表示数值 1001 是正数。正数的原码、反码和补码是一样的。

对于数 $X < 0$ ，如 $X = -1001$ ，则有符号数表示为

$$\begin{array}{ll} [X]_{\text{原}} = 11001 & \text{最高位的 1 表示数值 1001 是负数} \\ [X]_{\text{反}} = 10110 & \text{负数求反码是将数值位各位求反} \\ [X]_{\text{补}} = 10111 & \text{负数求补码是反码加 1} \\ [[X]_{\text{补}}]_{\text{补}} = 11001 = [X]_{\text{原}} & \text{负数补码的补码是原码} \end{array}$$

采用补码运算时，若运算结果为正数，该结果就是原码；若运算结果为负数，该结果是补码，必须再求补一次，才能得到运算结果的原码。另外必须注意，2 个 n 位二进制数相加的和不能大于 $2^n - 1$ ，否则将发生溢出错误。要想避免溢出错误，只有加长二进制数的字长。字长 n 决定了电路允许的最大数值为 $2^n - 1$ 。

(3) 二进制数加、减运算举例

[例 1.3.6] 设 $X_1 = +1001$ ， $X_2 = +0101$ ，试用无符号数和有符号数分别求 $X_1 + X_2$ 、 $X_1 - X_2$ 和 $X_2 - X_1$ 。

解：(1) 无符号数运算

$$\begin{aligned} X_1 + X_2 &= 1001 + 0101 = 1110 \\ X_1 - X_2 &= 1001 - 0101 = 0100 \\ X_2 - X_1 &= -(1001 - 0101) = -0100 \end{aligned}$$

(2) 有符号数运算

① 求 $X_1 + X_2$

$$\begin{aligned} [X_1]_{\text{补}} &= 01001, [X_2]_{\text{补}} = 00101 \\ [X_1]_{\text{补}} + [X_2]_{\text{补}} &= 01001 + 00101 = 01110 \end{aligned}$$

结果大于 0，说明该结果就是两数之和的原码。

② 求 $X_1 - X_2 = X_1 + (-X_2)$

$$\begin{aligned} [X_1]_{\text{补}} &= 01001, [-X_2]_{\text{补}} = [-X_2]_{\text{反}} + 1 = 11011 \\ [X_1]_{\text{补}} + [-X_2]_{\text{补}} &= 01001 + 11011 = 00100 \end{aligned}$$

$$\begin{array}{r} 01001 \\ + 11011 \\ \hline 100100 \end{array}$$

↑
自动舍去

运算结果为正数，是因为 $X_1 > X_2$ ，所以 $X_1 - X_2 > 0$ ，运算结果为两数之差的原码。

③ 求 $X_2 - X_1 = X_2 + (-X_1)$

$$\begin{aligned} [X_2]_{\text{补}} &= 00101, [-X_1]_{\text{补}} = [-X_1]_{\text{反}} + 1 = 10111 \\ [X_2]_{\text{补}} + [-X_1]_{\text{补}} &= 00101 + 10111 = 11100 \end{aligned}$$

运算结果为负数,说明这一结果是 $X_2 - X_1$ 的补码,即

$$[X_2 - X_1]_{\text{补}} = 11100$$

而 $X_2 - X_1$ 的原码是

$$[[X_2 - X_1]_{\text{补}}]_{\text{补}} = 10100$$

因为 $X_2 < X_1$, 所以

$$X_2 - X_1 = -0100$$

用有符号数表示就是 10100。

四、码制

码制:码制就是代码,是用 n 位数字符号表示 N 个客观信息量,这种表示又称为编码。如果用二进制数作为所要表示的信息编码,就得到二进制代码,简称二进制码。二进制码的位数 n 和要编码的信息量 N 之间必须满足关系式

$$2^n \geq N$$

1. 十进制数的二进制编码

十进制数的二进制编码就是用 n 位二进制数为 $0 \sim (2^n - 1)$ 个十进制数编码。常用的编码有两种。

(1) 自然二进制码:用 n 位二进制数为与之对应等值的 2^n 个 $(0 \sim 2^n - 1)$ 十进制数进行编码,这样的二进制码称为自然二进制码。它是一种有权码,自然二进制码的加权系数和就是它所对应的十进制数。

(2) 格雷码:格雷码也是用 n 位二进制数为 2^n 个十进制数进行编码,但它是一种无权码。它的编码规则是:相邻两个十进制数的两个代码只有一位互反,其余各位相同,并且首数 0 和尾数 $2^n - 1$ 的两个代码也满足这一规律。表 1.3.1 列出了十进制数 $0 \sim 15$ 的自然二进制码和格雷码。

表 1.3.1 十进制数的二进制编码

十进制数	自然二进制码	格 雷 码	十进制数	自然二进制码	格 雷 码
0	0000	0000	8	1000	1100
1	0001	0001	9	1001	1101
2	0010	0011	10	1010	1111
3	0011	0010	11	1011	1110
4	0100	0110	12	1100	1010
5	0101	0111	13	1101	1011
6	0110	0101	14	1110	1001
7	0111	0100	15	1111	1000

2. 十进制数码(0,1,2,⋯,9)的二进制编码——BCD 码

BCD 码就是用 4 位二进制数为十进制数的十个数码 0~9 进行编码。由于 4 位二进制数共有 16 种取值组合,因此有多种编码方式。常用的有 8421BCD 码, 5421BCD 码, 2421BCD 码, 这些是有权码, 无权码有余 3 码, 格雷码(又称循环码)。表 1.3.2 列出了部分 BCD 码。

表 1.3.2 常用 BCD 码编码表

十进制 数码	十进制数码的 BCD 码				
	8421 码	5421 码	2421 码	余 3 码	格雷码
0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 1 1	0 0 0 0
1	0 0 0 1	0 0 0 1	0 0 0 1	0 1 0 0	0 0 0 1
2	0 0 1 0	0 0 1 0	0 0 1 0	0 1 0 1	0 0 1 1
3	0 0 1 1	0 0 1 1	0 0 1 1	0 1 1 0	0 0 1 0
4	0 1 0 0	0 1 0 0	0 1 0 0	0 1 1 1	0 1 1 0
5	0 1 0 1	1 0 0 0	1 0 1 1	1 0 0 0	0 1 1 1
6	0 1 1 0	1 0 0 1	1 1 0 0	1 0 0 1	0 1 0 1
7	0 1 1 1	1 0 1 0	1 1 0 1	1 0 1 0	0 1 0 0
8	1 0 0 0	1 0 1 1	1 1 1 0	1 0 1 1	1 1 0 0
9	1 0 0 1	1 1 0 0	1 1 1 1	1 1 0 0	1 1 0 1

用 BCD 码表示十进制数时,就是直接用 BCD 码代换十进制数中的每一位数码。如十进制数 947,用 8421BCD 码表示为

$$947 = (100101000111)_{8421\text{BCD}}$$

§ 1.4 例题分析

[例 1.4.1] 把二进制数 11010.101 转换为十进制数、八进制数和十六进制数。

$$\begin{aligned} \text{解: (1) } (11010.101)_2 &= 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^1 + 1 \times 2^{-1} + 1 \times 2^{-3} \\ &= 16 + 8 + 2 + 0.5 + 0.125 = (26.625)_{10} \end{aligned}$$

$$(2) (11010.101)_2 = \underbrace{011}_3 \underbrace{010}_2 \underbrace{.101}_5 = (32.5)_8$$

$$(3) (11010.101)_2 = \underbrace{0001}_1 \underbrace{1010}_{10} \underbrace{.1010}_{10} = (1A.A)_{16}$$

如果利用十六进制数求加权系数和获得十进制数就会简练些,如

$$(11010.101)_2 = (32.5)_8$$

$$(11010.101)_2 = (1A.A)_{16}$$

$$\begin{aligned}(1A.A)_{16} &= 1 \times 16^1 + 10 \times 16^0 + 10 \times 16^{-1} \\ &= 16 + 10 + 0.625 = (26.625)_{10}\end{aligned}$$

【例 1.4.2】把十进制数 259.875 转换为二进制数。

解：用除 2 取余法和乘 2 取整法分别对整数和小数部分进行转换，可得

$$(259.875)_{10} = (100000011.111)_2$$

对整数部分进行转换时，要用除 2 取余法连除 9 次 2，才能得到结果，其演算很繁琐。可是把 259 写成 $256 + 3$ ，就简单了，因为 $(256)_{10} = 2^8 = (100000000)_2$ ， $(3)_{10} = (11)_2$ ，所以有

$$(259)_{10} = 2^8 + 3 = (100000000)_2 + (11)_2 = (100000011)_2$$

对于小数部分，也可类似写成

$$0.875 = 0.5 + 0.25 + 0.125 = 2^{-1} + 2^{-2} + 2^{-3} = (0.111)_2$$

因此有

$$(259.875)_{10} = (100000011.111)_2$$

这一结果告诉我们，记熟 $2^{10} = 1024$ ， $2^9 = 512$ ， $2^8 = 256$ ，…以及 $2^{-1} = 0.5$ ， $2^{-2} = 0.25$ ， $2^{-3} = 0.125$ ，…这些数，对于数制转换是十分有用的。

【例 1.4.3】把十进制小数 0.69 转换为二进制小数，二进制小数的精度为 2^{-4} 。

解：用乘 2 取整法得

乘 2	取 整
$0.69 \times 2 \approx 1.38$	… 1
$0.38 \times 2 \approx 0.76$	… 0
$0.76 \times 2 \approx 1.52$	… 1
$0.52 \times 2 \approx 1.04$	… 1

因此有

$$(0.69)_{10} \approx (0.1011)_2$$

【例 1.4.4】把数 $(5FB)_{16}$ 转换为二进制数和十进制数。

解：(1) $(5FB)_{16} = \begin{matrix} 5 & F & B \end{matrix} = (010111111100)_2$

$$\begin{array}{ccc} \downarrow & \downarrow & \downarrow \\ 0101 & 1111 & 1100 \end{array}$$

$$(2) (5FB)_{16} = 5 \times 16^2 + 15 \times 16^1 + 12 \times 16^0$$

$$= 1280 + 240 + 12 = (1532)_{10}$$

【例 1.4.5】把数 $(267.25)_8$ 转换为十六进制数。

解：要先把八进制数转换成二进制数，再转换成十六进制数。

$$\begin{array}{ccccccccc}
 (267.25)_8 = & 2 & & 6 & & 7 & . & 2 & & 5 = (010110111.010101)_2 \\
 & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\
 & 010 & & 110 & & 111 & & 010 & & 101 \\
 (10110111.010101)_2 = & \underbrace{1011}_{11} & \underbrace{0111}_7 & . & \underbrace{0101}_5 & \underbrace{0100}_4 = (B7.54)_{16}
 \end{array}$$

【例 1.4.6】用 8421BCD 码表示数 $(1011010)_2$ 。

解：BCD 码是十进制数码的二进制编码，因此必须把二进制数转换成十进制数，才能用 BCD 码表示。

$$(1011010)_2 = 2^6 + 2^4 + 2^3 + 2^1 = (90)_{10}$$

$$(90)_{10} = (10010000)_{8421BCD}$$

【例 1.4.7】设 $X_1 = (7.25)_{10}$ ， $X_2 = (13.5)_{10}$ ，试用有符号数求 $X_1 + X_2$ 和 $X_1 - X_2$ 。

解：(1) 先转换 X_1 、 X_2 为有 5 位整数的二进制数（因 $X_1 + X_2 > 16$ ，故需用 5 位二进制整数记录两数之和）。

$$X_1 = (7.25)_{10} = (00111.01)_2$$

$$X_2 = (13.5)_{10} = (01101.10)_2$$

(2) 求 $X_1 + X_2$ 。

$$[X_1]_{\text{补}} = 000111.01, [X_2]_{\text{补}} = 001101.10$$

$$[X_1]_{\text{补}} + [X_2]_{\text{补}} = 000111.01 + 001101.10 = 010100.11$$

$$(010100.11)_2 = 1 \times 2^4 + 1 \times 2^2 + 1 \times 2^{-1} + 1 \times 2^{-2} = (20.75)_{10}$$

(3) 求 $X_1 - X_2 = X_1 + (-X_2)$ 。

$$[X_1]_{\text{补}} = 000111.01, [-X_2]_{\text{补}} = 110010.10$$

$$[X_1]_{\text{补}} + [-X_2]_{\text{补}} = 000111.01 + 110010.10 = 111001.11$$

$$[X_1 - X_2]_{\text{原}} = [111001.11]_{\text{补}} = 100110.01$$

写成无符号数为

$$-(00110.01)_2 = (-6.25)_{10}$$

第 2 章 逻辑代数基础

§ 2.1 基本要求

(1) 掌握基本逻辑运算和导出逻辑运算及其真值表，熟记它们的逻辑符号。

(2) 掌握逻辑代数基本定律、变换规则及其应用。

(3) 掌握逻辑函数的 5 种表示方法及其相互对应关系。

(4) 熟悉逻辑函数表达式的 5 种基本形式及其相互变换。

(5) 掌握逻辑函数的卡诺图化简法，熟悉逻辑函数的代数化简法。

§ 2.2 重点难点

重点：(1) 常用逻辑运算，逻辑代数基本定律、规则及应用。

(2) 逻辑函数的表示方法及相互对应关系。

(3) 逻辑函数式的变换与化简。

难点：逻辑函数式的变换与代数化简法。

§ 2.3 内容提要

一、逻辑变量的二值性及其意义、逻辑函数及其表示法

1. 逻辑变量

具有两个对立状态的变量称为逻辑变量。如电路中的高电平和低电平；开关的闭合与断开；晶体管的饱和与截止等，都可视为逻辑变量。数字电路中的正逻辑系统，规定用 1 表示高电平，用 0 表示低电平，因此逻辑变量只有 0 或 1 两种取值，故逻辑变量又称为二值变量。逻辑变量取值 0 或 1 仅表示变量的两种对立状态，不具有数值大小的意义。

2. 逻辑函数

逻辑变量之间依一定的对应法则存在着确定的因果关系，这种因果关系称