高等学校电子与通信精品教材系列

数字电路与系统设计

Digital Circuits and Systems Design

张顺兴 主 编

黄丽亚
杨恒新 副主编

東南大學出版社

数字电路与系统设计

Digital Circuits and Systems Design

数字电路与系统设计

张顺兴　主　编
黄丽亚
杨恒新　副主编

东南大学出版社
·南　京·

内 容 简 介

本书详细地介绍了数字电路与系统的基本理论及其设计方法。全书内容在参照原国家教委颁布的数字电路课程基本教学要求的基础上，增加了可编程逻辑器件PLD、VHDL语言和数字系统设计等内容，能较好地适应当前数字电子技术的发展要求和当前我国高等院校工科本课程教学内容的实际需要。

全书共12章，内容包括：数制与码制、逻辑代数基础、集成逻辑门、组合逻辑电路、触发器、时序逻辑电路、脉冲信号的产生和变换、D/A和A/D转换、半导体存储器、可编程逻辑器件PLD、VHDL语言、数字系统设计。

本书可作为高等院校电气信息类通信工程、电子信息工程、计算机科学与技术、自动化各专业以及其他相关专业的本科教材，亦可供相关专业的研究生和工程技术人员阅读参考。

图书在版编目(CIP)数据

数字电路与系统设计/张顺兴主编 .—南京：东南大学出版社，2004.8
ISBN 7-81089-696-2

Ⅰ.数… Ⅱ.张… Ⅲ.数字电路—高等学校—教材 Ⅳ.TN79

中国版本图书馆CIP数据核字(2004)第07846号

东南大学出版社出版发行
(南京四牌楼2号 邮编210096)
出版人：宋增民
江苏省新华书店经销 南京邮电学院印刷厂印刷
开本：787 mm×1092 mm 1/16 印张：25 字数：624千字
2004年8月第1版 2004年8月第1次印刷
印数：1—4000册 定价：30.00元
(凡因印装质量问题，可直接向发行科调换。电话：025-83792327)

前　言

本书是根据原国家教委制定的电子技术基础课程教学的基本要求,总结多年教学和实践经验进行编写的。本书的前身是原教材《数字电路与系统》,新编教材在着重介绍数字电子电路的基本理论以及电路分析和设计方法的基础上,进一步完善并增加了PLD、VHDL和数字系统设计的相关内容,能更好地适应数字电子技术的发展趋势。本书可作为高等院校电气信息类通信工程、电子信息工程、计算机科学与技术、自动化各专业以及其他相关专业的本科教材,也可供相关专业的研究生和工程技术人员阅读参考。

本书由张顺兴任主编,黄丽亚、杨恒新任副主编。其中第1,2,5,6,7章及附录由张顺兴编写;第4,8,9,12章由黄丽亚编写(其中第4,8章在原教材作者编写的基础上改编,第12章在原教材第11章中的数字系统设计部分的基础上重新编写);第3,10,11章由杨恒新编写(其中第3章在原教材作者编写的基础上进行改编)。在此,对原教材作者盛利、杨秀侠为本书做出的贡献表示衷心的感谢!

在本书的编写过程中,南京邮电学院电子工程系主任、博士生导师王锁萍教授、副主任蒋国平教授给予了大力支持和具体指导;教务处处长梅杓春教授、副处长刘陈教授、教材科方小玉科长对本书的出版给予了大力的支持和帮助;东南大学出版社张煦编辑为本书的出版作了大量缜密细致的工作;曾使用本书进行教学的南京邮电学院教师周西峰、王厚大、黄学军、赵华、王伟、卢庆利、万佑红、张学军、车晶、戎舟、何艳等为本书的编写提供了宝贵意见。编者在此一并表示衷心的感谢!

本书在涉及集成电路的逻辑符号时,对现行国标符号及国内外曾用、常用的符号都作了介绍,并尽可能采用国标符号(对于中规模器件,为了便于教学,使用了示意性的简化符号,在简化符号中,对低电平有效的信号在其端子上加"。",并在信号名上加非号"－",而且带非号的信号名可视需要写在逻辑框的外部或内部,在国标中,逻辑框内不使用带非号"－"的变量,这与国标不完全一致)。

限于编者水平和成书时间仓促,书中难免有错漏和不足之处,恳盼读者批评指正。

编者

2004年8月

目 录

1 数制与码制

内容提要 本章介绍数制与码制两部分内容。在数制部分首先概述数制中的基本概念,然后介绍各种常用数制及其互相间的转换,最后介绍二进制数的算术运算;在码制部分介绍码制的基本概念以及常用的二进制码和 BCD 码。

1.1 数制

1.1.1 概述

数制是计数体制(即计数的方法)的简称,有累加计数制和进位计数制两种。累加计数制是原始的计数方法,计多大的数要使用与所计数目个数相等的各不相同的计数符号,很不方便。比较方便的计数方法是进位计数制,本章所述数制即指进位计数制。常用的进位计数制有十进位计数制(简称十进制)以及二、八、十六进制。在介绍各种数制之前,首先介绍数制的基础知识。

1) 基本数码

基本数码是指计数制中使用的基本数字符号,简称数码。例如,十进制中的 0、1、2、3、4、5、6、7、8、9 便是。

2) 基数(R)

计数制中所使用的数码的个数称为基数,亦称底数。基数常用R表示,在十进制中 $R=10$。在基数为 R 的数制中,每一个数位上可以使用的数码包括 0 在内共有 R 个,最大数码是 $R-1$,而没有 R,因此计数时当某位数计到 R 时,则在该位记作 0,并向高位进一,即逢 R 进一,故基数为 R 的计数制称为 R 进制计数制。

3) 数位(i)

在由一串数码构成的数中,数码所在的位置称数位。数位的排序用 i 表示,i 的计算以小数点为界,向左依次为第 0 位、第 1 位、……向右依次为第 -1 位、第 -2 位、……例如,在十进制数 123.45 中,3 是第 0 位数,2 是第 1 位数,4 是第 -1 位数等等。

4) 位权(weight)

位权亦称权值。在进位计数制的由一串数码构成的数中,各个数位上的数码所表示的数值的大小不但和该数码本身的大小有关,而且还和该数码所处的数位有关。例如,在十进制数 44 中,十位数 4 表示 4×10^1,个位数 4 表示 4×10^0。可见,不同的数位赋予该位上的数码以不同的表示数的大小的权力。我们把数位上的数码在表示数时所乘的倍数称为该数位的位权。

在 R 进制数中,第 i 位数位的权值用 W_i 表示,$W_i=R^i$。其中 R 是基数,i 是数位的位数。例如,十进制数的第 0 位(个位)、第 1 位、第 -1 位的位权分别为 10^0、10^1 和 10^{-1}。

在由一串数码表示的数中,相邻两个数位中左边数位的位权是右边的 R 倍。

5) 数的表示方式

在进位计数制中数的表示方式有位置记数法、按权展开式、和式 3 种。

(1) 以十进制数 123.45 为例，分别可以表示为

$$N^{①}_{10} = 123.45 \cdots\cdots ① \quad 位置记数法$$

$$= 1\times10^2 + 2\times10^1 + 3\times10^0 + 4\times10^{-1} + 5\times10^{-2} \cdots\cdots ② \quad 按权展开式$$

$$= \sum_{i=-2}^{2} D^{②}_i \times 10^i \cdots\cdots ③ \quad 和式$$

上述①式用一串数码来表示数，称为位置记数法或并列记数法。②式称为按权展开式，这是用多项式来表示一个数，又称多项式表示法。多项式中的各个乘积项由各个数位上的数码加权（加权即乘上权值）构成。③式是把按权展开式表示为∑的形式，称为和式。式中 D_i 表示第 i 位数位上的十进制数码，10^i 为第 i 位的权值。

(2) 对于任意一个 R 进制数 $(N)_R$，可以分别用 3 种方式表示如下：

$$(N)_R = a_{m-1}a_{m-2}\cdots a_1 a_0 a_{-1} a_{-2}\cdots a_{-n} \cdots\cdots ① \text{ 位置记数法}$$

$$= a_{m-1}\times R^{m-1} + a_{m-2}\times R^{m-2} + \cdots + a_1\times R^1$$
$$+ a_0\times R^0 + a_{-1}\times R^{-1} + \cdots + a_{-n}\cdot R^{-n} \cdots\cdots ② \text{ 按权展开式}$$

$$= \sum_{i=-n}^{m-1} a_i R^i \cdots\cdots ③ \text{ 和式}$$

式中 $a_{m-1}, a_{m-2}, \cdots, a_{-n}$—— 分别为第 $m-1, m-2, \cdots -n$ 位上的数码；

R—— 基数；

m、n—— 分别为整数部分的位数和小数部分的位数；

i——数位的序号。

1.1.2 常用数制

1) 二进制

数码：0、1

基数：$R=2$

第 i 位的权值：$W_i = 2^i$

例 表示方式：

$$(101.01)_2 = 1\times2^2 + 0\times2^1 + 1\times2^0 + 0\times2^{-1} + 1\times2^{-2}$$
$$= \sum_{i=-2}^{2} B_i \times 2^i$$

2) 八进制

数码：0、1、2、3、4、5、6、7

基数：$R=8$

第 i 位的权值：$W_i = 8^i$

例 表示方式：

$$(25.6)_8 = 2\times8^1 + 5\times8^0 + 6\times8^{-1} = \sum_{i=-1}^{1} O_i \times 8^i$$

① 脚标 10 表示 N 为十进制数。二、八、十、十六进制数分别用脚标 2、8、10、16 或 B、O、D、H 表示。

② D 表示十进制数码。二、八、十六进制数的数码分别用 B、O、H 表示。

3) 十六进制

数码:0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F

其中数码 A、B、C、D、E、F 依次与十进制数 10、11、12、13、14、15 等值,但 A、B、C、D、E、F 在十六进制中是 1 位数。

基数:$R=16$

第 i 位的权值:$W_i=16^i$

例 表示方式:

$$(12D.23)_{16}=1\times16^2+2\times16^1+13\times16^0+2\times16^{-1}+3\times16^{-2}$$
$$=\sum_{i=-2}^{2}H_i\times16^i$$

表 1.1.1 列出了十进制数 0~20 等数在其他各种数制中的写法。

表 1.1.1 十、二、八、十六进制数对照表

十进制	二进制	八进制	十六进制	十进制	二进制	八进制	十六进制
0	0	0	0	12	1100	14	C
1	1	1	1	13	1101	15	D
2	10	2	2	14	1110	16	E
3	11	3	3	15	1111	17	F
4	100	4	4	16	10000	20	10
5	101	5	5	17	10001	21	11
6	110	6	6	18	10010	22	12
7	111	7	7	19	10011	23	13
8	1000	10	8	20	10100	24	14
9	1001	11	9	32	100000	40	20
10	1010	12	A	100	1100100	144	64
11	1011	13	B	1000	1 111 101000	1750	$3E8$

1.1.3 数制转换

所谓数制转换是指将一种数制的数变换成等值的用另一种数制表示的数。

1) 二进制数(八、十六)→十进制数

将二进制、八进制、十六进制数转换成十进制数只要把原数写成按权展开式再相加即可。

例 1.1.1 分别将$(101.01)_2$,$(74.5)_8$,$(3C.A)_{16}$转换成十进制数。

解 ①$(101.01)_2=1\times2^2+0\times2^1+1\times2^0+0\times2^{-1}+1\times2^{-2}$

$=(5.25)_{10}$

②$(74.5)_8=7\times8^1+4\times8^0+5\times8^{-1}$

$=(60.625)_{10}$

③$(3C.A)_{16}=3\times16^1+12\times16^0+10\times16^{-1}$

$=(60.625)_{10}$

2) 十进制数→二(八、十六)进制数

十进制数转换成二、八、十六进制数只需将整数部分和小数部分分别转换成二、八、十六

进制数，再将转换结果连接在一起即可。

(1) 整数的转换用“除基数取余法” 将十进制数除以目标数制(所要转换成的数制)的基数，得商和余数，取下余数作为目标数制数的最低位数码；将所得的商再除以基数又得商和余数，取下余数作为目标数制数的次低位数码……如此连续进行，直至商为 0 余数小于目标数制的基数 R 为止，末次相除所得的余数为目标数制数的最高位数码。

例 1.1.2 将$(60)_{10}$分别转换成二、八、十六进制数。

解 ①

$$\begin{array}{r|l}
2 & 60 \\ \hline
2 & 30 \quad \cdots\cdots\text{余 }0\cdots\cdots\text{最低位} \\ \hline
2 & 15 \quad\quad 0 \\ \hline
2 & 7 \quad\quad 1 \\ \hline
2 & 3 \quad\quad 1 \\ \hline
2 & 1 \quad\quad 1 \\ \hline
 & 0 \quad\quad 1\cdots\cdots\text{最高位}
\end{array}$$

所以 $(60)_{10}=(111100)_2$。

②

$$\begin{array}{r|l}
8 & 60 \\ \hline
8 & 7 \quad \cdots\cdots\text{余 }4\cdots\cdots\text{最低位} \\ \hline
 & 0 \quad\quad 7\cdots\cdots\text{最高位}
\end{array}$$

所以 $(60)_{10}=(74)_8$。

③

$$\begin{array}{r|l}
16 & 60 \\ \hline
16 & 3 \quad \cdots\cdots\text{余 }12\cdots\cdots\text{最低位} \\ \hline
 & 0 \quad\quad 3\cdots\cdots\text{最高位}
\end{array}$$

所以 $(60)_{10}=(3C)_{16}$

(2) 小数的转换用“乘基数取整法”

将被转换的十进制小数乘以目标数制的基数 R，取下所得乘积中的整数作为目标数小数的最高位；再将乘积中的小数部分乘以基数 R，取下乘积中的整数作为目标数小数的次高位……如此反复进行，直到乘积的小数部分为 0 或达到所需精度为止。各次相乘所得的整数即可构成目标数的小数，第一次相乘所得的整数为小数的最高位，末次相乘所得的整数为小数的最低位。

例 1.1.3 将十进制数$(0.625)_{10}$分别转换成二、八、十六进制数。

解 ①

$$\begin{array}{lr}
 & 0.625 \\
 & \times \quad 2 \\ \hline
\text{最高位} & (1).250 \\
 & \times \quad 2 \\ \hline
 & (0).500 \\
 & \times \quad 2 \\ \hline
\text{最低位} & (1).000
\end{array}$$

所以 $(0.625)_{10}=(0.101)_2$

②

$$\begin{array}{r}
0.625 \\
\times \quad 8 \\ \hline
(5).000
\end{array}$$

所以 $(0.625)_{10}=(0.5)_8$

③

$$\begin{array}{r} 0.625 \\ \times \quad 16 \\ \hline 3.750 \\ +\ 6.25 \\ \hline (10).000 \end{array}$$

所以 $(0.625)_{10}=(0.A)_{16}$

(3) 剩余误差及转换位数的确定

一个 R 进制 n 位小数的精度为 R^{-n},例如$(0.95)_{10}$的精度为 10^{-2}。采用乘基数取整法将十进制小数转换成二、八、十六进制小数时,可能出现多次相乘后乘积的小数部分仍不为 0 的情况。如果转换取了 n 位,则转换的剩余误差 Δ 小于该 n 位小数的精度,即 $\Delta < R^{-n}$。例如,$(0.95)_{10}=(0.746314631\cdots)_8$,当转换取3位时,可得$(0.95)_{10}=(0.746)_8$,则$\Delta=(0.0003146314631\cdots)_8$,$\Delta<8^{-3}$。

一个 j 位的 α 进制小数,当其转换成 β 进制小数时,如要求转换后保持或不低于原来的精度,其转换成的 β 进制小数应取的最少位数(设为 k 位)计算如下:

由于原精度为 α^{-j},转换后的精度为 β^{-k},根据转换精度的要求应有 $\alpha^{-j}\geqslant\beta^{-k}$,可以求得 $k\geqslant j\dfrac{\lg\alpha}{\lg\beta}$。

k 应取满足上式的最小正整数,即

$$j\frac{\lg\alpha}{\lg\beta}\leqslant k<j\frac{\lg\alpha}{\lg\beta}+1$$

3) 两种任意数制的转换

把一个 α 进制数转换成β 进制数可以利用十进制数作为桥梁:

$$\alpha\text{ 进制数}\xrightarrow{\text{按权展开相加}}\text{十进制数}\xrightarrow{\text{基数乘除法}}\beta\text{ 进制数}$$

如果 α 和β 都是 2 的正整数幂,则可进行直接转换。例如,当进行二$\rightleftharpoons$八(十六)进制转换时,由于3(4)位二进制数可以用1位八(十六)进制数表示,因此,二进制数表示为八(十六)进制数时可把二进制数以小数点为界分别向左和向右每 3(4)位组成一组,组成时不足 3(4)位时整数部分高位添 0,小数部分低位添 0,使其补足 3(4)位,然后将每组 3(4)位二进制数用 1 位八(十六)进制数表示即可,反之亦然。例如:

$$(0\,\underline{10}\,\underline{101}.\underline{001}\quad\underline{11}0)_2\rightleftharpoons(25.16)_8$$

$$(0\,0\,\underline{10}\quad\underline{1101}.\underline{0100}\quad\underline{1}000)_2\rightleftharpoons(2D.48)_{16}$$

在用上述方法将八(十六)进制数转换为二进制数时需要舍去多余的 0(整数部分最高位的 0 和小数部分最低位的 0)。

需要说明的是,在数字技术中用得最多的是二进制。这是因为二进制中只有 0 和 1 两个数码,很容易用高低电平来表示;如果采用十进制,则需要用 10 个不同的状态来表示十进制中 10 个不同的数码,很不容易。此外,二进制数的运算也比较简单,因此机器数都用二进制数表示。但是用二进制表示的数往往很长,不便于书写和记忆,相对而言,八进制、十六进制数的书写和记忆比较方便,而且和二进制数之间的转换也很方便,因此,又常用八进制和

十六进制数作为二进制数的缩写形式用于书写、输入和显示。

1.1.4 二进制数的算术运算

当两个二进制数用来表示数量的大小时，它们可以进行数值运算，这种运算称为算术运算。二进制数算术运算的法则和十进制数的运算法则基本相同。惟一的区别在于十进制数是逢十进一、借一当十，而二进制数是逢二进一、借一当二。

以两个二进制数 1001 和 0101 为例，其算术运算有：

加法运算

$$\begin{array}{r} 1\,0\,0\,1 \\ +\;\;0\,1\,0\,1 \\ \hline 1\,1\,1\,0 \end{array}$$

减法运算

$$\begin{array}{r} 1\,0\,0\,1 \\ -\;\;0\,1\,0\,1 \\ \hline 0\,1\,0\,0 \end{array}$$

乘法运算

$$\begin{array}{r} 1\,0\,0\,1 \\ \times\;\;0\,1\,0\,1 \\ \hline 1\,0\,0\,1 \\ 0\,0\,0\,0\;\; \\ 1\,0\,0\,1\;\;\;\; \\ +\;\;0\,0\,0\,0\;\;\;\;\;\; \\ \hline 0\,1\,0\,1\,1\,0\,1 \end{array}$$

除法运算

$$\begin{array}{r} 1.1\;\;1\cdots \\ 0101)\overline{1\,0\,0\,1}\qquad \\ -0\,1\,0\,1\qquad \\ \hline 1\,0\,0\,0\;\;\;\; \\ -0\,1\,0\,1\;\;\;\; \\ \hline 0\,1\,1\,0 \\ -0\,1\,0\,1 \\ \hline 0\,0\,1 \end{array}$$

需要说明的是，在数字电路中为了简化运算电路，通常二数相减的运算是用其补码的加法来实现的，乘法运算则用移位和加法两种操作来完成，而除法运算是用移位和减法操作来完成，因此，二进制数的加、减、乘、除都可以用加法电路完成。所以在数字设备中加法器是极为重要的运算部件。

1.2 码制

用文字、符号或数码来表示各个特定对象的过程称为编码，编码所得的每组符号称为代码或码字，代码中的每个符号称为基本代码或码元。在数字电路中通常用二进制数码 0 和 1 构成的代码来表示各有关对象（如十进制数、字符等）。码制是指编码的制式，不同的码制编码时遵循不同的规则。

1.2.1 二进制码

所谓二进制代码是指用二进制数码 0 和 1（这里代码中的码元 0、1 不一定有大小的概念）构成的代码。n 位的二进制代码可以有 2^n 个代码。表 1.2.1 给出了几种典型的二进制代码。