

(第2版)

数字电子技术

吴小花 © 主编

SHUZI DIANZI
JISHU

广东高等教育出版社
Guangdong Higher Education Press

SHUZI DIANZI
JISHU
SHUZI D

(第2版)

数字电子技术

吴小花 © 主编

内 容 简 介

本书是依据高等职业教育电气自动化技术专业《数字电子技术》教学大纲编写的,以完成项目任务为主线,链接相应的理论知识和技能实训,融“教、学、做”为一体,在编写中特别注重基本技能和应用能力的培养,注重职业素养和创新能力的培养,充分体现了高职课堂教学改革的新理念,更适应新时期高等职业教育的需要。

全书共分8章,内容包括:逻辑代数基础、逻辑门电路、组合逻辑电路、触发器、时序逻辑电路、脉冲波形的产生与整形、存储器与可编程逻辑器件、数/模和模/数转换器。每章都有内容提要、本章小结、实训项目和习题。

本书设置了逻辑事件、四路抢答器、十字路口交通信号灯定时控制系统、大规模数字集成器件、数/模和模/数转换等5个学习情境,编排了包括基本电路的功能测试、应用电路设计实训为内容的18个项目,贯穿全书。

本书可作为高职、高专院校的电子信息、通信、供用电、自动化、电气、机电一体化等电类专业的教材,也可供从事相应工作的技术人员、成人教育、职业培训、自学者参考。

图书在版编目(CIP)数据

数字电子技术/吴小花主编. —2版. —广州:广东高等教育出版社,2014.8
ISBN 978-7-5361-5003-4

I. ①数… II. ①吴… III. ①数字电路-电子技术-高等职业教育-教材 IV.
①TN79

中国版本图书馆CIP数据核定(2013)第236699号

出版发行	广东高等教育出版社
	地址:广州市天河区林和西横路
	邮编:510500 电话:(020) 87554152
	网址: http://www.gdgjs.com.cn
印 刷	广州市穗彩印务有限公司
开 本	787毫米×1092毫米 1/16
印 张	13.5
字 数	293千字
版 次	2010年9月第1版 2014年8月第2版
印 次	2010年9月第1次印刷 2014年8月第3次印刷
定 价	32.00元

再版前言

《数字电子技术（第2版）》是在保持原版的教材内容、理论体系和风格的基础上，根据高职高专教育数字电子技术基础课程教学基本要求，本着“项目引导、任务驱动、教学做一体化”的原则进行修订的。编写本教材第2版的指导思想是：保证基础、精选内容、突出重点、加强应用、便于教学、利于自学。《数字电子技术（第2版）》保持了原版的体系、内容和特点，同时还广泛吸取了使用本教材的师生的意见与建议。本次修订除纠正了原版中存在的错漏符号、图形等不规范的问题外，考虑到许多院校在安排教学计划时先安排上数字电路，再上模拟电路的习惯，如第八章的数/模和模/数转换器，把习题部分关联到模拟电子知识较多以及理论推导繁杂的有关内容删除了。为便于学生自学自测和巩固所学的知识，在书末增加了部分习题的参考答案。

由于编者水平有限，修订后的第2版教材一定还会有许多不尽如人意之处，恳请广大师生和读者批评指正。

编 者

2013年1月

前 言

根据新时期的高等职业教育要由“重视规模发展”转向“注意提高素质”的发展要求,本着“以就业为导向、以能力为本位、以服务为宗旨”的指导思想,我们深入开展了职业教育课程教学改革活动。结合作者多年的教学改革和实践经验,编写了《数字电子技术》。本书以电类专业共同具备的岗位职业能力为依据,遵循学生认知规律,紧密地结合了职业资格考证中对电子技能所做的要求。

本书具有如下几个主要特点:

1. 采用项目教学,内容突出“应用性、技能性和趣味性”。
2. 以完成项目任务为主线,连接相应的理论知识和技能实操,融教、学、做为一体,有利于教学互动。
3. 本书适合实施“边教、边学、边做”的教学方法。

建议课时为 64 节,建议与由华南理工大学出版,吴小花、龚兰芳主编的《电子技能训练与 EDA 技术应用》配套使用。

全书内容包括:逻辑代数基础、逻辑门电路、组合逻辑电路、触发器、时序逻辑电路、脉冲波形的产生与整形、存储器与可编程逻辑器件、数/模和模/数转换器。其中标有“*”号的部分建议作为选学内容,可根据学时多少取舍。

本书紧密结合高等职业教育特点,突出理论与实际紧密结合,强调技术应用,淡化电路的内部结构和工作原理,每章都安排了实训项目,通过项目任务的引领,将理论知识融入其中,使

学生能够学以致用，既可以激发学生的学习兴趣，又能培养学生的实操能力。

本书由广东水利电力职业技术学院吴小花编写，编写过程中曾得到李殊骁教授和学院电子教研组同事们的热情支持，在此谨向他们表示衷心的感谢。

由于时间仓促和编者水平所限，书中一定还会有疏漏和错误之处，殷切地期望广大读者给予批评和指正。来信可通过 E-mail 发至 xhua_225@163.com 或 wuxh@gdsdxy.cn。

编 者

2010 年 3 月

目 录

学习情境一 逻辑事件	(1)
第一章 逻辑代数基础	(2)
1.1 概述	(2)
1.2 数制和码制	(3)
1.2.1 几种常用的数制	(3)
1.2.2 不同数制间的相互转换	(4)
1.2.3 常用的二进制代码	(7)
1.3 逻辑代数基础	(8)
1.3.1 逻辑代数中的三种基本运算	(8)
1.3.2 逻辑代数的基本公式和基本定律	(11)
1.3.3 逻辑函数及其表示方法	(13)
1.4 逻辑函数的化简方法	(17)
1.4.1 逻辑函数的公式化简法	(17)
1.4.2 逻辑函数的卡诺图化简法	(18)
本章小结	(26)
◆ 实训项目 信号灯的逻辑控制	(28)
习题	(31)
学习情境二 四路抢答器	(33)
第二章 逻辑门电路	(34)
2.1 概述	(34)
2.2 分立元件门电路	(35)
2.2.1 二极管、三极管的开关特性	(35)
2.2.2 分立元器件门电路	(36)
2.3 复合逻辑门电路	(38)
2.4 TTL 集成门电路	(39)
2.5 CMOS 集成门电路	(42)
2.6 TTL 与 MOS 集成电路的区别及使用注意事项	(43)
本章小结	(44)

◆ 实训项目一 逻辑门电路的基本功能测试	(45)
◆ 实训项目二 由门电路构成的四路抢答器的设计、制作与测试	(48)
习题	(51)
学习情境三 十字路口交通信号灯定时控制系统	(53)
第三章 组合逻辑电路	(54)
3.1 概述	(54)
3.2 组合逻辑电路的分析方法与设计方法	(55)
3.2.1 组合逻辑电路的基本分析方法	(55)
3.2.2 组合逻辑电路的设计方法	(56)
3.3 常用的组合逻辑电路	(58)
3.3.1 加法器	(58)
3.3.2 数值比较器	(61)
3.3.3 编码器	(63)
3.3.4 译码器	(66)
3.3.5 数据选择器及数据分配器	(73)
3.4 组合逻辑电路中的竞争-冒险现象	(75)
3.4.1 竞争-冒险现象及其产生原因	(75)
3.4.2 消除竞争-冒险现象的方法	(75)
本章小结	(76)
◆ 实训项目一 编码器、译码器功能测试	(77)
◆ 实训项目二 九级电压判别器电路设计与制作	(81)
◆ 实训项目三 三人表决器的逻辑电路设计与制作	(83)
习题	(85)
第四章 触发器	(87)
4.1 概述	(87)
4.2 RS 触发器	(88)
4.2.1 基本 RS 触发器	(88)
4.2.2 同步 RS 触发器	(89)
4.3 D 触发器	(91)
4.3.1 同步 D 触发器	(91)
4.3.2 边沿 D 触发器	(91)
4.4 边沿 JK 触发器	(92)
4.5 T 触发器	(93)
4.6 T' 触发器	(94)
4.7 时钟触发器逻辑功能的相互转换	(94)
本章小结	(96)

◆ 实训项目一 触发器功能测试	(97)
◆ 实训项目二 用 D 触发器改进四路抢答器电路实验与实训	(99)
习题	(101)
第五章 时序逻辑电路	(103)
5.1 概述	(103)
5.2 时序逻辑电路的分析方法	(104)
5.2.1 同步时序逻辑电路的分析方法	(104)
* 5.2.2 异步时序逻辑电路的一般分析方法	(108)
5.3 寄存器	(109)
5.3.1 数码寄存器	(109)
5.3.2 移位寄存器	(110)
5.4 计数器	(113)
5.4.1 计数器概述	(113)
5.4.2 二进制计数器	(113)
5.4.3 集成十进制计数器	(118)
5.4.4 实现 N 进制计数器的方法	(119)
本章小结	(121)
◆ 实训项目一 四位简易频率计的设计与制作	(122)
◆ 实训项目二 寄存器功能测试	(128)
◆ 实训项目三 计数器功能测试	(130)
◆ 实训项目四 二位可预置数的减法计数电路的设计与制作	(132)
习题	(134)
第六章 脉冲波形的产生与整形	(137)
6.1 概述	(137)
6.2 555 定时器及其应用	(138)
6.2.1 555 定时器的电路结构与功能	(138)
6.2.2 用 555 定时器构成多谐振荡器	(139)
6.2.3 用 555 定时器构成施密特触发器	(140)
6.2.4 用 555 定时器构成单稳态触发器	(142)
本章小结	(144)
◆ * 实训项目一 555 定时器基本功能测试	(145)
◆ 实训项目二 555 定时器的典型应用	(147)
◆ * 实训项目三 十字路口交通信号灯定时控制系统的设计、安装与调试	(149)
习题	(156)

学习情境四 大规模数字集成器件	(158)
第七章 存储器与可编程逻辑器件	(159)
7.1 概述	(159)
7.2 存储器及其应用	(159)
7.2.1 随机存储器 (RAM)	(159)
7.2.2 只读存储器 (ROM)	(163)
7.2.3 用存储器实现组合逻辑函数	(165)
*7.3 可编程逻辑器件	(167)
本章小结	(170)
◆ * 实训项目 EPROM 构成多路序列信号发生器	(171)
习题	(176)
学习情境五 数/模和模/数转换	(177)
第八章 数/模和模/数转换器	(178)
8.1 概述	(178)
8.2 D/A 转换器	(178)
8.2.1 D/A 转换器的基本原理	(178)
8.2.2 倒 T 型电阻网络 D/A 转换器	(179)
8.2.3 权电阻网络 D/A 转换器	(180)
8.2.4 D/A 转换器的主要技术指标	(181)
8.2.5 常用的集成 D/A 转换器芯片	(182)
8.3 A/D 转换器	(184)
8.3.1 A/D 转换器的基本原理	(184)
8.3.2 并行比较型 A/D 转换器	(186)
8.3.3 逐次逼近型 A/D 转换器	(187)
8.3.4 双积分型 A/D 转换器	(188)
8.3.5 A/D 转换器的主要技术指标	(190)
本章小结	(191)
◆ 实训项目一 加法计数器 D/A 功能测试	(192)
◆ 实训项目二 ADC0809 A/D 功能测试	(194)
习题	(197)
部分习题参考答案	(198)

学习情境一 逻辑事件

逻辑事件在日常生活中随处可见，如家庭照明用开关控制灯泡（管）、体育竞赛项目（如举重）中裁判判决、智力竞赛抢答控制等等。如果用不同的字母代表一个事件的各个环节和事件的结果，用数字表示事件和事件各个环节的状态，那么，我们就可以用字母写出代数关系式来表示事件结果与事件各环节之间的关系，即逻辑关系。

在这个学习情境中，通过设置项目——信号灯的逻辑控制，引入逻辑及逻辑控制的概念。

教学任务：

(1) 介绍数字电子技术的发展与现状，了解数字电子技术的工程应用。

(2) 介绍数字电子技术课程的功能与定位、主要内容、特点。

(3) 基本逻辑事件的表示方法。

(4) 逻辑变量与逻辑函数。

(5) 逻辑函数的化简。

(6) Proteus ISIS 的操作方法。

(7) 培养学生科学严谨的工作作风、认真负责的工作态度，培养较好的心理素质，具有团队合作的精神及良好的职业道德素养。

(8) 培养学生搜查资料的能力。

(9) 培养学生学习电子技术及后续课程的兴趣。

(10) 培养学生的语言表达能力。

学习目标：

(1) 了解数学逻辑的概念，理解“与”“或”“非”三个基本逻辑关系。

(2) 熟悉各种进制及它们之间的转换。

(3) 熟练掌握逻辑代数中的基本公式与常用定律、逻辑问题的描述方法、逻辑函数的化简与变换。

(4) 熟悉 Proteus ISIS 7.1 仿真软件的使用。

(5) 培养学习兴趣和认真做事的态度；培养表述、回答问题等语言表达能力。

教学实施：

教师示范、小组讨论、总结归纳、教师点评。



第一章 逻辑代数基础

逻辑代数是分析和设计数字电路的基本数学工具。本章介绍了有关数制和码制的基本概念,逻辑代数的基本概念、常用公式和定理,逻辑函数的几种表示方法;重点介绍了逻辑函数的公式化简法和卡诺图化简法。

1.1 概 述

电子电路中的电信号分为两大类。一类是数字信号,是指在时间和幅值上都是离散的信号,例如数字显示仪表的显示值、计算机系统中各部件之间传输的信息等。另一类是模拟信号,是指在时间或数值上都是连续变化的信号。例如,从热电偶得到的电压或电流信号就是一个模拟信号,因为被测的温度不会发生突变,所以测得的电压或电流无论在时间上还是数值上都是连续的。习惯上把工作数字信号下的电子电路叫做数字电路,把工作模拟信号下的电子电路叫做模拟电路。

数字电路具有许多优点,如便于高度集成化、通用性强、保密性好、抗干扰能力强、稳定可靠、数字信息便于长期保存等。随着计算机科学与技术的发展,用数字电路进行信息处理的优势日益突出。为了充分发挥和利用数字电路在信息处理上的强大优势,我们可以将模拟信号转换成数字信号,然后送到数字电路进行处理,最后再将处理结果根据需要转换成相应的模拟信号。

数字信号通常用数码形式给出,不同的数码可以用来表示数量的不同大小。在用数码表示数量大小时,仅用一位数码往往是不够的,故经常需要用进位计数制的方法组成多位数码使用。我们把多位数中的每一位构成方法以及从低位到高位进位的规则称为数制。在日常生活中,人们习惯使用十进制,而在数字电路中则多采用二进制,有时也采用八进制和十六进制。

不同的数码不仅可以用来表示数量的大小不同,而且可以用来表示不同的事物或事物的不同状态。当用于表示不同事物时,这些数码就不再具有表示数值大小的含义了,只是不同事物的代号而已,通常把这些数码称为代码,正如在运动比赛中,为便于识别运动员,通常给每一位运动员编一个号码。各选手的号码没有数量大小的含义,只表示不同的运动员而已。为了便于记忆和查找,在编制代码时总要遵循一定的规则,这些规则就称为码制。每个人都可以根据自己的需要选定码制的规则,编制出一组代码。但为了信息交换的需要,必须制定一些大家共同使用的通用代码,例如BCD码、字符码等。

1.2 数制和码制

1.2.1 几种常用的数制

数制具有三个基本要素：基、权、进制。

基：数码的个数（也叫做基数）。

权：数码所在位置表示数值的大小（也叫做位权）。

进制：即进位规则，逢“基”进一。

一、十进制（Decimal）

十进制数是以 10 为基数的计数体制。10 个不同的数码是 0, 1, 2, 3, 4, 5, 6, 7, 8, 9。其低位数和相邻高位数之间的关系是“逢十进一”。

任意一个十进制数 N ，都可以用多项式表示为：

$$(N)_{10} = \sum_{i=-m}^{n-1} K_i 10^i \quad (n \text{ 是整数部分的位数, } m \text{ 为小数位数})$$

式中： N 为十进制数；

10^i 为第 i 位的权；

K_i 为第 i 位的系数。

例如： $312.25 = 3 \times 10^2 + 1 \times 10^1 + 2 \times 10^0 + 2 \times 10^{-1} + 5 \times 10^{-2}$ 可以书写成 $(312.25)_{10}$ 或 312.25_D ，下标 D 表示十进制的形式。

二、二进制（Binary）

二进制是目前数字电路中应用最广泛的数制。二进制数是以 2 为基数的计数体制，每位数码的取值仅有 0 或 1 两个可能的数码，每位的权是 2 的幂。它的进位规则是“逢二进一”。

任何一个二进制数 N ，都可以用多项式表示为：

$$(N)_2 = \sum_{i=-m}^{n-1} K_i 2^i \quad (n \text{ 是整数部分的位数, } m \text{ 为小数位数})$$

式中： N 为二进制数；

2^i 为第 i 位的权；

K_i 为第 i 位的系数。

例如： $(1011.011)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3}$ ，也可以书写成 1011.011_B ，下标 B 表示二进制的形式。

三、八进制数（Octal）

八进制数是以 8 为基数的计数体制。8 个不同的数码是 0, 1, 2, 3, 4, 5, 6, 7。它的进位规则是“逢八进一”。

任意一个八进制数 N 都可以用多项式表示为：

$$(N)_8 = \sum_{i=-m}^{n-1} K_i 8^i \quad (n \text{ 是整数部分的位数, } m \text{ 为小数位数})$$

式中: N 为八进制数;

8^i 为第 i 位的权;

K_i 为第 i 位的系数。

例如: $(376.4)_8 = 3 \times 8^2 + 7 \times 8^1 + 6 \times 8^0 + 4 \times 8^{-1}$, 也可以书写成 376.4_0 , 下标 0 表示八进制的形式。

四、十六进制数 (Hexadecimal)

十六进制数是以 16 为基数的计数体制。16 个不同的数码是 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F。它的进位规则是“逢十六进一”。

任意一个十六进制数 N 都可以用多项式表示为:

$$(N)_{16} = \sum_{i=-m}^{n-1} K_i 16^i \quad (n \text{ 是整数部分的位数, } m \text{ 为小数位数})$$

式中: N 为十六进制数;

16^i 为第 i 位的权;

K_i 为第 i 位的系数。

例如: $(3AB.11)_{16} = 3 \times 16^2 + 10 \times 16^1 + 11 \times 16^0 + 1 \times 16^{-1} + 1 \times 16^{-2}$, 可以书写成 $(3AB.11)_{16}$ 或 $3AB.11_H$, 下标 H 表示十六进制的形式。

1.2.2 不同数制间的相互转换

一、二进制数和十进制数的转换

1. 二—十转换

把二进制数转换为十进制数称为二—十转换。在进行转换时, 只要将二进制按位权展开, 然后将所有各项的数值按十进制数相加即可, 例如:

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

2. 十—二转换

把十进制数转换为二进制数称为十—二转换。常用的方法是: 整数部分用“除 2 取余法”, 具体方法是: 除 2 取余, 直至商为 0, 数的高位到低位的排列顺序为由下到上; 小数部分用“乘 2 取整法”; 具体方法是: 乘 2 取整, 取有效位, 小数点后的高位到低位的排列顺序为所取整数的由上到下。

例 1 将 $(58)_{10}$ 转换为二进制数。

解: 用除 2 取余法。

$$\begin{array}{rcll}
 2 \overline{) 58} & \cdots \cdots & \text{余} 0 & \text{最低位} \\
 2 \overline{) 29} & \cdots \cdots & \text{余} 1 & \uparrow \\
 2 \overline{) 14} & \cdots \cdots & \text{余} 0 & \\
 2 \overline{) 7} & \cdots \cdots & \text{余} 1 & \\
 2 \overline{) 3} & \cdots \cdots & \text{余} 1 & \\
 2 \overline{) 1} & \cdots \cdots & \text{余} 1 & \text{最高位} \\
 & & 0 &
 \end{array}$$

把所有余数按箭头方向从高位到低位排列起来便可得到： $(58)_{10} = (111010)_2$ 。

如果一个数既有整数，又有小数部分，则可分别对整数部分和小数部分进行转换，然后合并起来即可。

例 2 将十进制小数 58.8125 转换为二进制小数。

解：从例 1 已经知道 $(58)_{10} = (111010)_2$ 。

$$\begin{array}{rcll}
 0.8125 & & & \\
 \times 2 & & & \\
 \hline
 1.6250 & \cdots \cdots & \text{整数部分}=1, B_{-1}=1 & \text{最高位} \\
 0.6250 & & & \downarrow \\
 \times 2 & & & \\
 \hline
 1.2500 & \cdots \cdots & \text{整数部分}=1, B_{-2}=1 & \\
 0.2500 & & & \\
 \times 2 & & & \\
 \hline
 0.5000 & \cdots \cdots & \text{整数部分}=0, B_{-3}=0 & \\
 0.5000 & & & \\
 \times 2 & & & \\
 \hline
 1.0000 & \cdots \cdots & \text{整数部分}=1, B_{-4}=1 & \text{最低位}
 \end{array}$$

可得， $(58.8125)_{10} = (111010.1101)_2$

二、十六进制数和十进制数的转换

1. 十六进制数转换为十进制数

将十六进制数转换成十进制数的方法和将二进制数转换成十进制数的方法相似，即把想要转换的十六进制数按位权展开相加即可。

例 3 将十六进制数 3F5 转换成十进制数。

解： $3F5_H = 3 \times 16^2 + 15 \times 16^1 + 5 \times 16^0 = 1013_D$

2. 十进制数转换为十六进制数

将十进制数转换为十六进制数，用“除 16 取余”法，即将要转换的十进制数连续除以 16，直到商小于 16 为止，然后把各次余数按最后得到的为最高位，最先得到的为最低位，依次排列起来所得到的数即是所求的十六进制数。

例 4 求十进制数 3256 所对应的十六进制数。

解：用除 16 取余法，把 3256 连续除以 16，直到最后。

16	3256	……余 8	写成十六进制	8	最低位
16	203	……余 11	写成十六进制	B	↑
	12	……余 12	写成十六进制	C	最高位

把所得余数按箭头方向从高到低排列起来便可得到： $(3256)_{10} = (CB8)_{16}$ 。

三、二进制数与八进制数、十六进制数之间的相互转换

1. 二进制数转换为八进制数

由于3位二进制一共有8个状态，而且它的进位输出逢8进1，所以3位二进制数恰好相当于一位8进制数。因此，在将二进制数转换为八进制数时，只要将二进制数的整数部分从低位到高位每3位分为一组并代之以等值的八进制数，同时将小数部分从高位到低位每3位分为一组并代之以等值的八进制数即可。

例5 将 $(001101111010.1011)_2$ 转换为八进制数。

解：二进制 $\underline{001} \ \underline{101} \ \underline{111} \ \underline{010} \ . \ \underline{101} \ \underline{100}$
 ↓ ↓ ↓ ↓ . ↓ ↓
 八进制 1 5 7 2 . 5 4
 $(001101111010.1011)_2 = (1572.54)_8$

2. 八进制数转换为二进制数

在将八进制数转换为二进制数时，只要按原来顺序把每一位八进制数用相应的3位二进制数来代替即可。

例6 将 $(520.371)_8$ 转换为二进制数。

解：八进制 $\underline{5} \quad \underline{2} \quad \underline{0} \quad . \quad \underline{3} \quad \underline{7} \quad \underline{1}$
 ↓ ↓ ↓ . ↓ ↓ ↓
 二进制 $\underline{101} \ \underline{010} \ \underline{000} \ . \ \underline{011} \ \underline{111} \ \underline{001}$
 $(520.371)_8 = (101010000.011111001)_2$

四、二进制数与十六进制数之间的相互转换

1. 二进制数转换为十六进制数

由于4位二进制数恰好有16种状态，而把4位二进制数看作一个整体时，它的进位输出又正好是逢16进1，所以只要从低位到高位将整数部分每4位二进制数分为一组并代之以等值的十六进制数，同时从高位到低位将小数部分每4位二进制数分为一组并代之以等值的十六进制数，即可得到对应的十六进制数。

例7 将 $(001101111010.1011)_2$ 转换为十六进制数。

解：二进制 $\underline{0011} \ \underline{0111} \ \underline{1010} \ . \ \underline{1011}$
 ↓ ↓ ↓ ↓
 十六进制 3 7 A . B
 $(001101111010.1011)_2 = (37A.B)_{16}$

2. 十六进制数转换为二进制数

在将十六进制数转换为二进制数时，只要按原来顺序把每一位十六进制数用等值的4位二进制数来代替即可。

例 8 将 $(9B3A5.28)_{16}$ 转换为二进制数。

解：十六进制 9 B 3 A 5 . 2 8
 ↓ ↓ ↓ ↓ ↓ ↓ ↓
 二进制 1001 1011 0011 1010 0101 . 0010 1000

即： $(9B3A5.28)_{16} = (10011011001110100101.00101000)_2$

1.2.3 常用的二进制代码

将若干个二进制数码 0 和 1 按一定规则排列起来表示某种特定含义的代码，称为二进制代码，表 1-1 中列出了常见的 3 种二进制代码，它们的编码规则各不相同。

一、8421BCD 码

在有权 BCD (Binary Coded Decimal) 码中，每 1 位十进制数均用一组 4 位二进制数码来表示，这 4 位二进制数码中的每 1 位都有固定权，表示固定的数值。

8421 码是十进制代码中最常用的一种 BCD 码。这种编码的优点是 4 位码之间满足二进制的规律；8、4、2、1 是 4 位二进制数所在位的权。将每一位的 1 代表的十进制数加起来，得到的结果就是它所代表的十进制数。

例如： $(01100011)_{8421BCD} = (63)_D$ 。

用 8421BCD 码表示 42609 应为： $(42609)_D = (01000010011000001001)_{8421BCD}$ 。

二、余 3 码

余 3 码的编码规则与 8421 码不同，如果把每一个余 3 码看作 4 位二进制数，则它的数值要比它所代表的十进制数码多 3，故将这种代码叫做余 3 码。余 3 码不是恒权代码。

三、格雷码

格雷码 (Gray Code) 又称循环码，其编码方式有多种，表 1-1 中仅列出其中一种。与普通二进制代码相比，格雷码的最大特点就是相邻两个代码之间只有一位发生变化，即编码中任意两个相邻数对应的代码中只有一位不同，其余各位均相同。这种码又称为单位距离码。单位距离码的最大优点是实现它的逻辑电路可靠性高，应用十分广泛。1.4.2 节中的卡诺图就使用了格雷码。

表 1-1 常见的二进制代码

十进制数	编码种类		
	8421BCD 码	余 3 码	格雷码
0	0000	0011	0000
1	0001	0100	0001
2	0010	0101	0011
3	0011	0110	0010
4	0100	0111	0110
5	0101	1000	0111