

高等院校选用教材

电子信息类

# 数字集成电路教程

浙江工业大学《电子学》教研组

● 龙忠琪 贾立新 编著



科学出版社

437

## 内 容 简 介

本书是为面向 21 世纪素质教育而专门组织编写的,可作为高等院校电子技术课程数字电路部分的选用教材。

全书共 13 章,分为 6 个部分:数字电路分析设计基础、组合逻辑电路、时序逻辑电路、数-模和模-数转换电路、逻辑电路的机助设计、基础练习题与思考题。其中删除了落后的电路内容,重点介绍高速和超高速 CMOS、BiCMOS 及 LSTTL 数字集成电路芯片的原理、应用和基本理论方法,包括 PLD 器件的原理和应用、数字电路 CAD 设计技术等。本书理论和实践并重,内有大量集成芯片应用实例。

### 图书在版编目(CIP)数据

数字集成电路教程/龙忠琪等编著,-北京:科学出版社,2001

ISBN 7-03-009123-X

I. 数… II. 龙… III. 数字集成电路-高等学校-教材 N. TN431.2

中国版本图书馆 CIP 数据核字(2001)第 03478 号

科学出版社 出版

北京东黄城根北街 16 号

邮政编码:100717

新蕾印刷厂 印刷

科学出版社发行 各地新华书店经销

\*

2001 年 5 月第 一 版 开本:787×1092 1/16

2001 年 5 月第一次印刷 印张:18 1/2

印数:1—4 500 字数:420 000

定价:27.00 元

(如有印装质量问题,我社负责调换〈环伟〉)

# 前 言

本教程是高等院校电子技术课程数字电路部分的通用教材,是为面向 21 世纪素质教育而专门组织编写的。与以往教材相比,本教程在内容选取上有三个较大的变化:

(1)从以普通 TTL 电路为主转为以 CMOS 电路为主,LSTTL 电路为辅,尤其加强了高速、超高速 CMOS 和 BiCMOS 电路的内容。

(2)从以分立集成电路模块介绍为主转向同可编程逻辑器件(PLD)并重。

(3)从用中小规模固定功能模块设计数字系统的传统方法为主转向同数字电路机助设计(CAD)并重。

在编写过程中,注意处理了以下几个关系:

(1)既要体现内容的先进性,又要强化基础理论的学习。一方面,内容尽量紧跟电子技术日新月异发展的步伐,反映新工艺、新器件和新的理论方法,甚至部分内容要紧扣最新发展方向和趋势,以拓宽读者的知识空间和激发学生的创新意识;同时,书中涵盖了实际应用中常用的基本概念和基础理论方法,以强化基础理论和基本能力的培养,使读者有一个坚实的理论基础背景,并使本教程具有较强的生命力。

(2)为体现内容的实用性和实践性,器件尽量选用有代表性的实际产品,设计举例尽量采用实例,以缩小理论教学与实际应用之间的空间距离和过渡过程。

(3)内容丰富、信息量大是本教程的一个特点,但同时注重简明、精练,可读性好、易于讲授,是本书始终遵循的编写宗旨。所以其中多数内容讲解深入、细致、易懂,并伴有大量例子;在讲授风格上,淡化传统的解释式、分析式的传授模式,尽量采用启发式、设计式的诱导模式,着重培养读者的形象思维和提高学生举一反三的能力。另一方面,从素质教育出发,有的内容简略介绍,甚至点到为止,以给学生留有一定的思考空间。详略有别是本教程的另一个特点。

此外,还注意了以下几点:

(1)注重视觉效果和形象教学。部分电路芯片既给出了电路图和电路符号,又给出了封装信息或产品样品实物照片,对电路图中需要突出显示的部分特意加了阴影。

(2)教材中全部采用现行国家标准电路图形符号(GB4728)。同时,给出了部分电路的简化功能符号。

(3)习题分为基础练习题和思考题两部分。基础练习题主要用于巩固所学知识,达到基本要求;思考题是从研究生入学试题、实际科研课题以及国内外资料中精心挑选或专门设计的,难度有所提高,主要用于锻炼学生的独立思考能力、综合运用所学知识的能力和创造性思维能力。习题解答将随即制成光盘,盘内同时备有本教程绪论课的多媒体课件和数字电路试题库,库内有精选试题近千个,可自动生成难、较难、中、易四种试卷,适用于各个层次、各个阶段的考教分离试卷生成。

本教程由浙江工业大学信息工程学院电子学教研组集体讨论编写,龙忠琪教授任主编并亲自执键,贾立新副教授任副主编。参加编写工作的还有毛礼武、李如春和浙江大学

的龙胜春等,叶幼慧负责审校。基础练习题、思考题及其解答,题库中的试题选编工作由余佩琼和孙惠英负责,题库软件由王森、赵小江设计,多媒体课件由陈文召、陈锋完成,最后由龙忠琪、贾立新审定。

作 者

2000年12月于杭州

# 目 录

## 前言

第一部分 数字电路分析设计基础	1
1 绪论	1
1.1 数字电路、数字信号与数字系统	1
1.1.1 数字电路与数字信号	1
1.1.2 数字系统	1
1.2 数字电子技术的发展与应用	2
1.3 数字系统中的信息表征	3
1.3.1 数值信息的表示方法	3
1.3.2 非数值信息的表示方法	6
2 逻辑代数基础	8
2.1 概述	8
2.2 逻辑代数中的基本运算	8
2.2.1 与运算	8
2.2.2 或运算	9
2.2.3 非运算	10
2.2.4 复合逻辑运算	10
2.3 逻辑代数中的基本公式	11
2.4 逻辑代数中的基本定理	12
2.4.1 代入定理	12
2.4.2 对偶定理	12
2.4.3 山农定理	13
2.4.4 展开定理	13
2.5 逻辑函数及其表示方法	14
2.5.1 逻辑函数的表达式表示法	14
2.5.2 逻辑函数的真值表表示法	16
2.5.3 逻辑函数的逻辑图表示法	17
2.5.4 逻辑函数的卡诺图表示法	17
2.6 逻辑函数的化简	19
2.6.1 逻辑函数化简的意义及其最简形式	19
2.6.2 逻辑函数的公式化简法	20
2.6.3 逻辑函数的卡诺图化简法	22
2.6.4 任意项及其在逻辑函数化简中的应用	24

* 2.6.5 逻辑函数的列表化简法和 Petrick 算法 .....	25
* 2.6.6 逻辑函数的机助化简 .....	32
2.7 本章小结 .....	36
<b>第二部分 组合逻辑电路 .....</b>	<b>37</b>
<b>3 逻辑门电路 .....</b>	<b>37</b>
3.1 概述 .....	37
3.2 MOS 门电路 .....	37
3.2.1 CMOS 门电路 .....	38
3.2.2 CMOS 门电路的外特性 .....	43
3.2.3 高速和超高速 CMOS 电路 .....	50
3.3 双极型门电路 .....	54
3.3.1 概述 .....	54
3.3.2 LSTTL 门电路 .....	54
3.3.3 LSTTL 门电路的外特性 .....	57
3.3.4 其他 TTL 门电路 .....	63
3.4 BiCMOS 电路 .....	64
3.4.1 BiCMOS 门电路 .....	64
3.4.2 BiCMOS 反相器的外特性 .....	65
3.5 本章小结 .....	68
<b>4 组合逻辑电路的组成及其分析设计方法 .....</b>	<b>69</b>
4.1 组合逻辑电路的组成 .....	69
4.2 组合逻辑电路的分析方法 .....	69
4.3 组合逻辑电路的设计方法 .....	71
4.4 组合逻辑电路中的险象 .....	73
4.4.1 组合逻辑电路中的竞争与险象 .....	73
4.4.2 组合逻辑电路中的险象判别方法 .....	74
4.4.3 组合逻辑电路中的险象消除方法 .....	75
<b>5 常用中大规模组合逻辑电路 .....</b>	<b>76</b>
5.1 编码器 .....	76
5.1.1 二进制编码器 .....	76
5.1.2 BCD 码编码器 .....	80
5.2 译码器 .....	81
5.2.1 二进制译码器 .....	81
5.2.2 MSI BCD 码译码器 .....	84
5.2.3 显示译码器 .....	85
5.3 数据选择器 .....	88
5.3.1 工作原理和电路构成 .....	88

\* \* 表示选讲内容。

5.3.2 MSI 数据选择器 .....	90
5.4 运算电路 .....	92
5.4.1 数值比较器 .....	92
5.4.2 加法器 .....	93
5.4.3 乘法器 .....	96
5.4.4 算术逻辑单元(ALU) .....	96
5.5 只读存储器(ROM) .....	97
5.5.1 固定 ROM .....	97
5.5.2 可编程 ROM、EPROM 和 E <sup>2</sup> PROM .....	98
5.6 MSI/LSI 组合电路芯片的位扩展 .....	100
5.6.1 编码器的位扩展 .....	100
5.6.2 译码器的位扩展 .....	102
5.6.3 数据选择器的位扩展 .....	104
5.6.4 数值比较器的位扩展 .....	105
5.6.5 ROM 的字位扩展 .....	107
5.7 用 MSI/LSI 标准电路芯片设计组合电路 .....	108
5.8 本章小结 .....	115
6 可编程组合逻辑器件 .....	116
6.1 可编程逻辑器件概述 .....	116
6.2 PLA(可编程逻辑阵列) .....	116
6.3 PAL .....	118
6.3.1 PAL 的基本组成 .....	118
6.3.2 PAL 的输出结构 .....	118
6.3.3 用 PAL 设计组合电路 .....	121
6.4 用 FPLA 设计组合逻辑电路 .....	122
<b>第三部分 时序逻辑电路</b> .....	<b>125</b>
7 触发器 .....	125
7.1 RS 触发器 .....	125
7.1.1 基本 RS 触发器 .....	125
7.1.2 同步 RS 触发器 .....	127
7.1.3 主从 RS 触发器 .....	129
7.1.4 用 RS 触发器组成其他功能的触发器 .....	130
7.2 D 触发器 .....	132
7.2.1 维持阻塞 D 触发器 .....	133
7.2.2 主从 D 触发器 .....	134
7.3 JK 触发器 .....	136
7.3.1 主从 JK 触发器 .....	136
7.3.2 边沿 JK 触发器 .....	138
7.4 施密特触发器 .....	139

7.4.1	用门电路构成的施密特触发器	139
7.4.2	施密特触发器的应用	140
7.5	单稳态触发器	142
7.5.1	用门电路构成的单稳态触发器	142
7.5.2	集成单稳态触发器	143
7.5.3	单稳态触发器的应用	144
7.6	555 定时器	146
7.6.1	用 555 定时器构成施密特触发器	146
7.6.2	用 555 定时器构成单稳态触发器	147
7.6.3	用 555 定时器构成无稳态电路	148
7.7	本章小结	149
8	常用 MSI/LSI 时序逻辑电路	151
8.1	时序逻辑电路的组成及功能描述方法	151
8.2	寄存器和移位寄存器	152
8.2.1	寄存器	152
8.2.2	移位寄存器	152
8.2.3	移位寄存器的应用	156
8.3	计数器	157
8.3.1	计数器功能及其组成	157
8.3.2	触发器及时钟电路	157
8.3.3	进位链接及输出电路	159
8.3.4	借位链接及输出电路	161
8.3.5	预置数和清零电路	163
8.3.6	MSI 计数器电路	164
8.4	可读/写存储器	168
8.4.1	随机读/写存储器(RAM)	168
8.4.2	顺序存取存储器(SAM)	171
9	时序逻辑电路的分析与综合	173
9.1	时序逻辑电路的分析	173
9.2	用 SSI 电路芯片设计时序电路	176
9.2.1	穆尔(Moore)型时序电路设计	176
9.2.2	米里(Mealy)型时序电路设计	181
9.3	用 MSI/LSI 电路芯片设计时序电路	183
9.3.1	任意进制计数/分频器的设计	183
9.3.2	信号发生器的设计	185
9.3.3	其他常用时序电路设计举例	190
10	可编程时序逻辑器件	194
10.1	寄存可编程逻辑阵列(RPLA)	194
10.1.1	RPLA 的组成	194



10.1.2	FPLS	195
10.1.3	RPAL	197
10.2	通用阵列逻辑(GAL)	200
10.2.1	GAL 的电路组成	200
10.2.2	OLMC	203
10.2.3	GAL 器件应用举例及使用注意事项	205
10.2.4	ispGAL	208
10.3	FPGA	210
10.3.1	概述	210
10.3.2	FPGA 的结构组成	211
10.4	用 PLD 器件设计时序电路及 PLD 器件的选用	215
10.4.1	用 PLD 器件设计时序电路	215
10.4.2	PLD 器件的选用	215
10.5	本章小结	216
<b>第四部分 数-模和模-数转换电路</b>		217
11	D/A 转换电路	217
11.1	基本 DAC 电路	217
11.1.1	权电流型 DAC	217
11.1.2	倒 T 型 R-2R 网络 DAC	218
11.2	常用 DAC 芯片及其应用举例	219
11.3	DAC 的主要性能参数及芯片选用方法	221
11.3.1	主要性能参数	221
11.3.2	集成 DAC 芯片的选用	222
12	A/D 转换电路	223
12.1	A/D 转换的基本概念	223
12.2	基本 ADC 电路	224
12.2.1	逐次逼近型 ADC	224
12.2.2	双积分型 ADC	225
12.3	常用 ADC 芯片及其典型应用举例	227
12.4	ADC 的主要性能参数及芯片选用	230
<b>*第五部分 逻辑电路的机助设计</b>		233
13	数字电路 CAD	233
13.1	数字电路 CAD 设计流程	233
13.2	数字电路的建模	234
13.2.1	行为模型	234
13.2.2	结构模型	236
13.2.3	混合模型	238
13.3	设计综合	239
13.3.1	设计综合过程	239

---

13.3.2 电路图生成 .....	240
13.4 设计仿真 .....	242
13.4.1 测试输入选择及表征 .....	242
13.4.2 元件延迟模型 .....	242
13.5 模块系统 CAD .....	244
13.6 时序电路 CAD .....	245
13.6.1 设计综合 .....	245
13.6.2 分析验证 .....	248
13.7 PLD 电路 CAD .....	248
13.7.1 常用 PLD 电路 CAD 工具 .....	248
13.7.2 PLD 电路 CAD 设计过程 .....	248
13.7.3 数字电路的 PDL 表征 .....	249
13.7.4 PLD 电路 CAD 设计举例 .....	252
13.8 本章小结 .....	253
<b>第六部分 基础练习题与思考题</b> .....	<b>255</b>
<b>参考文献</b> .....	<b>284</b>

# 第一部分

## 数字电路分析设计基础

### 1 绪 论

**导读:**本章主要讲三个问题:①数字电路、数字信号与数字系统的基本概念;②数字电子技术的发展与应用;③数字系统中信息的表示方法,包括数值信息和非数值信息。

#### 1.1 数字电路、数字信号与数字系统

##### 1.1.1 数字电路与数字信号

电子线路,根据工作信号的不同可分为模拟电路、脉冲电路和数字电路。简单地说,工作于模拟信号之下的电路称为模拟电路,工作于脉冲信号之下的电路称为脉冲电路。在时间上和幅值上连续变化的信号称为模拟信号,如正弦信号、温度信号、录音磁带上的语音信号等。在时间上、幅值上不连续的(或称离散的)信号称为脉冲信号,如脉搏信号、撞击信号、取样信号等。

数字电路是工作在数字信号之下的电路。数字信号是由数字 **0、1** 组成的信号(图 1.1-1),实际上也是一种脉冲信号,如 DVD 碟片上的信号。所以,数字电路也是一种脉冲电路。但是,数字电路不关心脉冲波形的本身,而着重于电路输出状态与输入状态之间的逻辑关系,因此,数字电路也叫做逻辑电路。

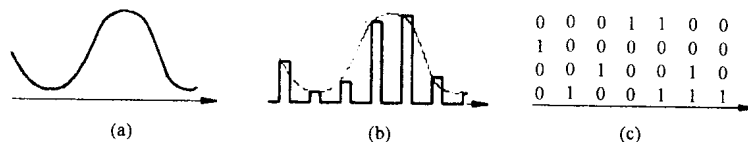


图 1.1-1 模拟信号、脉冲信号和数字信号

(a)模拟信号 (b)脉冲信号(如对图(a)的采样信号) (c)图(b)脉冲的数字信号表示

##### 1.1.2 数字系统

数字系统,是用离散的数字方式表征与处理信息的系统,是一个相互连接的功能模块的集合。所以,数字电路是数字电子计算机和所有数字电子系统的硬件基础,而计算机则是数字电路应用与发展的结晶和典范。因此,学好数字电路课程对于学好后续的计算机等课程具有重要意义。

数字系统的分析与设计,可分为五个层面:系统层、寄存器传输层、门电路结构层、开关器件如晶体管层以及材料物理层。本书涉及的内容主要是门电路以上层面上的分析与设计。

应当指出,现在许多常用工艺各种功能系列的通用电路模块已经配套齐全,这给设计与实现不同逻辑功能要求的电子系统提供了方便。但是,对于这种用固定功能模块设计的电路系统,用户是不能改变的,其组成的系统相对庞大且难以满足一些专用场合的需求。所以,当前一个迅速发展的趋势是,用计算机和通用可编程逻辑器件代替固定功能模块进行系统设计,用编程控制即可使系统具有不同的逻辑功能,这种“基于芯片”的设计理念与技术正在逐渐取代“基于固定功能模块”的传统设计模式。

## 1.2 数字电子技术的发展与应用

自从1906年世界上第一只电子管诞生,到1946年莫奇特和埃克特发明世界上第一台电子数字计算机Eniac至今,电子技术的发展已经经历了四代:电子管是第一代;1947年由美国贝尔实验室三位科学家肖克利(W. Shockley)、巴丁(J. Bardeen)、布拉顿(W. Braton)发明的晶体管是第二代;1965年第一片集成电路(IC)问世,是第三代;20世纪70年代到80年代初出现了大规模集成电路(LSI)和超大规模集成电路(VLSI),是第四代;现在,已经进入甚大规模集成电路(UVLSI)阶段,但称已经进入第五代还为时尚早。

电子管的诞生,促成了第一台电子数字计算机的问世,而电子计算机的出现,又带来了整个科学技术质的飞跃和革命性变革,被称为20世纪最伟大的科技事件之一。然而,电子管体积大、重量大、耗电多、寿命短、怕震易碎。以Eniac为例,重30多吨、占地1500多平方英尺,用管子31万8千多个,耗电相当于一个中小城镇的照明用电,平均不到一小时出现一次故障。不过,在许多领域中,如无线电广播高频高压大功率发射中,至今仍在使用着电子管。

到了20世纪50年代,电子管逐渐被晶体管所取代,因为晶体管体积小、重量轻、耗电少、寿命长、坚固可靠,晶体管给电子技术和整个工业领域带来了翻天覆地的变化,因此1956年,晶体管的发明者们获得了诺贝尔物理学奖。

20世纪60年代中期,有人把无线电四大元件(R、L、C、晶体管)中的三个同时制作在一块小硅片上,出现了集成电路,从而从分立元件时代过渡到了集成电路时代。需要指出,60年代出现的第一片集成电路是数字集成电路,因为数字集成电路容差大,易于制作,合格率高。由于集成电路的元器件之间不需要人工焊接引线,所以便于封装、体积更小、重量更轻、故障率更低、性能更好、寿命更长、更坚固可靠、使用更加方便,因而得到十分迅速的发展。

到了20世纪70年代,有人不仅把单个元件、单个门电路集成到同一块硅片上,而且把多个单元、子系统等较复杂的电路集成到一起,一个小小的硅片上有成百上千个门电路、数十万只元器件,这就是大规模集成电路。一个几十平方毫米的硅片上可以集成一台甚至多台微处理器及其外围电路、多个小系统、数以千万计的元器件……。可以说,一个崭新的数字时代已经到来。人们常常把每片集成电路中的门电路或等效门电路的数量称为集成度,含有1~10门的为小规模集成(SSI),10~100门的为中规模集成(MSI),100~

10 000 门的为大规模集成(LSI),10 万门以上的为超大规模集成(VLSI)。

现在,随着电子技术的发展,电子技术的应用已经渗透到从国防军工到国计民生的各个角落和各个领域:工业、农业、科技文教、医药卫生、交通运输、航空航天……。而数字电子技术的发展更是日新月异,用途越来越广泛,内容越来越丰富,技术越来越成熟。其应用随处可见,比如你手上戴的电子表、书包内的计算器、实验室中的数字仪表、家庭里的数字彩电、数码相机、市场上的电子秤、马路上的交通管制、工厂里的数控机床,以及数控打火机、程控电话等等,不胜枚举,这些无一不是数字电子技术应用的成果。在许多情况下,完成同样功能,数字电路同模拟电路相比,速度更快、精度更高、工作更稳定、抗干扰能力强、实现更容易、操作更简便、体积更小甚至价格更低廉,信息的处理、存储、故障检测与校正等更加方便。

在工艺上,数字集成电路已从当初的电阻-晶体管逻辑(RTL)、二极管-晶体管逻辑(DTL)发展到了晶体管-晶体管逻辑(TTL)、射极耦合逻辑(ECL)和集成注入逻辑(I<sup>2</sup>L),同时出现了 NMOS、PMOS、CMOS 和 GaAsMOS 等 MOS 数字集成电路,以及 BiCMOS 电路。当前正朝着更加高速、低耗、低电源电压和大密度集成的方向发展。本书将重点介绍高速或超高速 CMOS 电路和 BiCMOS 电路,包括电路原理、系统分析与设计,简单介绍使用渐少的 TTL 电路。

### 1.3 数字系统中的信息表征

计算机和其他数字系统的主要功能是处理信息,因此必须将信息表示成电路能够识别、便于运算或存储的形式。要处理的信息主要有两大类:一是数值信息;二是非数值信息——文字字符或控制符等。下面先介绍数值信息的表示方法——数制及其转换,然后介绍非数值信息的表征——编码。

#### 1.3.1 数值信息的表示方法

##### 1. 数制

在日常生活中,人们最熟悉的是十进制数。但在计算机和数字电路中,常用的是二进制、八进制和十六进制数。数的进制不同,表示形式也不相同。例如一个十进制数  $19_{(10)}$ ,若用二进制来表示则为  $10011_{(2)}$ ,用八进制表示则为  $23_{(8)}$ ,用十六进制表示则为  $13_{(16)}$ ,即

$$19_{(10)} = 10011_{(2)} = 23_{(8)} = 13_{(16)}$$

为了便于比较,表 1.3-1 给出了十进制数中 0~17 的不同数制的表示方法。

众所周知,十进制数是“逢 10 进 1”,因此,个位上有 1 为 1,有 2 为 2……十位上的 1 则代表 10,2 则代表 20……所以,用多项式表示则为

$$19_{(10)} = 9 \times 1 + 1 \times 10 = 9 \times 10^0 + 1 \times 10^1$$

由此推断,一个十进制数  $N_{(10)}$  的一般表达式为

$$N_{(10)} = \sum d_i \times 10^i \quad (1.3.1-1)$$

式中  $d_i = 0, 1, 2, \dots, 9$ , 是  $10^i$  位的系数,  $10^i$  中的 10 称为十进制数的基数。

表 1.3-1 几种常用数制对照

数制	十进制	二进制	八进制	十六进制
基数	10	2	8	16
前 18 个 正 整 数	0	0	0	0
	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	7
	8	1000	10	8
	9	1001	11	9
	10	1010	12	A
	11	1011	13	B
	12	1100	14	C
	13	1101	15	D
	14	1110	16	E
	15	1111	17	F
	16	10000	20	10
17	10001	21	11	
使用字符	0,1,2,3,4 5,6,7,8,9	0,1	0,1,2,3 4,5,6,7	0,1,2,3,4,5 6,7,8,9,A B,C,D,E,F

类似地,二进制数是“逢 2 进 1”,因此个位上有 1 为 1,右起第 2 位有 1 则为 2,第 3 位有 1 则为 4……。显然,同样是 1,在不同位置上所代表的数的大小却不相同,我们则称它们的“权”不同。在二进制数中,从个位算起,各位的权依次为 1,2,4,8(即  $2^0, 2^1, 2^2, 2^3$ )……,所以

$$10011_{(2)} = 1 \times 2^0 + 1 \times 2^1 + 0 \times 2^2 + 0 \times 2^3 + 1 \times 2^4$$

可以想见,一个二进制数  $N_{(2)}$  的一般表达式应为

$$N_{(2)} = \sum b_i \times 2^i \quad (1.3.1-2)$$

式中  $b_i = 0$  或 1,是  $2^i$  位的系数, $2^i$  中的 2 称为二进制数的基数。

注意,式(1.3.1-1)和(1.3.1-2)也适于有小数时的情况,只是小数部分的  $i$  为负数而已。

## 2. 数制转换

在数字系统中,会遇到多种进制的数,所以必须熟悉进制之间的转换。下面主要介绍数字系统中用得较多的十进制数如何转换成二进制数,以及二进制数如何转换成十进制数、八进制数和十六进制数。

### (1) 十进制数转换成二进制数

将十进制数转换成二进制数,可以按照“除 2 求余,至商为 0,余数倒级联”的方法进行。例如,将数  $19_{(10)}$  转换成二进制数,可以按如下所示连续除 2,依次求得余数 11001,倒

级联后即为 10011, 这就是  $19_{(10)}$  所对应的二进制数。

除数	→	2	19	←	被除数		
		2	9	←	商数	余数 1	↑ 余数倒级联
		2	4			余数 1	
		2	2			余数 0	
		2	1			余数 0	
			0			余数 1	

将十进制数转换成八、十六等其他进制数时, 只需将除数值改成 8 或 16 等即可, 口诀仍然是“除基数求余, 至商为 0, 余数倒级联”。

### (2) 二进制数转换成八进制数

二进制数转换成八进制数时, 可以把二进制数由小数点向左每 3 位分为一组(最左一组不够 3 位时用 0 左补齐), 再由左向右依次读取各组所对应的十进制数即可, 例如

$$\underline{10\ 011}_{(2)} = \underline{010\ 011}_{(2)} \quad \text{[用 0 左补齐]}$$

$$= 23_{(8)}$$

二进制数的位数相对较长, 不便阅读和记忆, 使用八进制更方便些。

### (3) 二进制数转换成十六进制数

二进制数转换成十六进制数时, 可以将二进制数由小数点向左每 4 位分为一组(最左一组位数不够 4 位, 用 0 左补齐), 再由左向右依次读取各组数据即可, 例如

$$1\ \underline{0011}_{(2)} = \underline{0001\ 0011}_{(2)} \quad \text{[用 0 左补齐]}$$

$$= 13_{(16)}$$

十六进制数的位数相对较短, 因此在编程和存储数据时多用十六进制数表示。

注意, 在十六进制数中, 每位数都有 16 种状态: 0, 1, 2, ..., 15。其中, 10, 11, ..., 15 分别用 A、B、C、D、E 和 F 表示。

二进制数转换成十进制数, 已在 1.3.1 节中给出示例, 参见式(1.3.1-2), 其基本要领是: “见 1 加(其)权”。

**例 1.3-1** 将十进制数  $246_{(10)}$  转换成八进制数和十六进制数。

**解:**

#### (1) 转换成八进制数

除数	→	8	246	←	被除数		
		8	30	←	商数	余数 6	↑ 余数倒级联
		8	3			余数 6	
			0			余数 3	

即

$$246_{(10)} = 366_{(8)}$$

#### (2) 转换成十六进制数

除数	→	16	246	←	被除数		
		16	15	←	商数	余数 6	↑ 余数倒级联
			0			余数 15=F	

即

$$246_{(10)} = F6_{(16)}$$

**例 1.3-2** 将十六进制数  $9E.78_{(16)}$  转换成等值的十进制数和八进制数。

解:

(1) 将  $9E.78_{(16)}$  转换成十进制数

$$\begin{aligned} 9E.78_{(16)} &= 9 \times 16^1 + 14 \times 16^0 + 7 \times 16^{-1} + 8 \times 16^{-2} \\ &= 158.46875_{(10)} \end{aligned}$$

(2) 将  $9E.78_{(16)}$  转换成八进制数

首先转换成二进制数:

$$9E.78_{(16)} = \underbrace{1001}_9 \underbrace{1110}_E. \underbrace{0111}_7 \underbrace{1000}_8_{(2)}$$

再将此二进制数转换成八进制数:

$$\begin{aligned} 1001\ 1110.\ 0111\ 1000_{(2)} &= \underbrace{010}_2 \underbrace{011}_3 \underbrace{110}_6. \underbrace{011}_3 \underbrace{110}_6 \underbrace{000}_0_{(2)} \quad [3\text{位一组两端补}0] \\ &= 236.360_{(8)} \end{aligned}$$

### 1.3.2 非数值信息的表示方法

因为数字系统通常用二值器件构成,所以计算机和其他数字系统中的文字字符或控制符等都用二进制数字代码表示,称为编码。应用最广泛的是 ASCII 编码(American Standard Code for Information Interchange),如表 1.3-2 所示,每个控制符、字母字符、算符等都用 7 位二进制代码  $C_6C_5C_4C_3C_2C_1C_0$  表示( $C_7$  是误码校验位),其中数字 0~9 当作字符也用代码表示。例如,字符 SUB(减)的代码为 0001 1010,SP(空格)的代码为 0010 0000,数字 1 的代码为 0011 0001 等。如果要给系统一个指令“SUB 1”(减 1),即可用以下三个代码来表示:

字符	ASCII 码
SUB	0001 1010
空格	0010 0000
1	0011 0001

这样,人们就可以通过键盘向计算机或数字系统输入字母字符、数据或发送指令了。

在数字电路中,十进制数还有多种表示方式,常见的几种示于表 1.3-3 中,用得最多的是 8421 BCD 码。例如数  $369_{(10)}$ ,用 8421 BCD 码表示为 0011 0110 1001<sub>(BCD)</sub>,也就是说,BCD 码的位内是纯二进制编码,位间是十进制编码,故称这种编码方式为二进制编码的十进制码,即 BCD 码(Binary coded Decimal),其权是 8,4,2,1。

表 1.3-3 中还给出了另一种常用代码,称为格雷码,也称循环码。这种码的最大特点是,任意两个相邻码之间只有 1 位不同,即其码距为 1。这种码比长码距码出错的可能性更小,因而更可靠,在逻辑代数和数据测量中有着广泛应用。



表 1.3-2 字符代码——ASCII

$C_6C_5C_4$ $C_3C_2C_1C_0$	000	001	010	011	100	101	110	111
0000	NUL(空,无效)	DLE(数据键换码)	SP(空格)	0	@	P	'	p
0001	SOH(标题开始)	DC1(设备控制 1)	!	1	A	Q	a	q
0010	STX(正文开始)	DC2(设备控制 2)	"	2	B	R	b	r
0011	ETX(本文结束)	DC3(设备控制 3)	#	3	C	S	c	s
0100	EOT(传输结束)	DC4(设备控制 4)	\$	4	D	T	d	t
0101	ENQ(询问)	NAK(否定)	%	5	E	U	e	u
0110	ACK(承认)	SYN(空转同步)	&	6	F	V	f	v
0111	BEL(报警铃响)	ETB(信息组传输结束)	,	7	G	W	g	w
1000	BS(退 1 格)	CAN(作废)	(	8	H	X	h	x
1001	HT(横向列表)	EM(纸尽)	)	9	I	Y	i	y
1010	LF(换行)	SUB(减)	*	:	J	Z	j	z
1011	VT(垂直列表)	ESC(换码)	+	;	K	[	k	{
1100	FF(走纸控制)	FS(文字分隔符)	'	<	L	\	l	
1101	CR(回车)	GS(组分分隔符)	-	=	M	]	m	}
1110	SO(移位输出)	RS(记录分隔符)	.	>	N	^	n	~
1111	SI(移位输入)	US(单元分隔符)	/	?	O	-	o	DEL(删除)

表 1.3-3 几种常用数值代码

十进制数	二进制	8421 BCD 码	2421 BCD 码	5211 BCD 码	余 3 码	余 3 循环码	格雷码
权	8421	8421	2421	5211	无权	无权	无权
0	0000	0000	0000	0000	0011	0010	0000
1	0001	0001	0001	0001	0100	0110	0001
2	0010	0010	0010	0100	0101	0111	0011
3	0011	0011	0011	0101	0110	0101	0010
4	0100	0100	0100	0111	0111	0100	0110
5	0101	0101	1011	1000	1000	1100	0111
6	0110	0110	1100	1001	1001	1101	0101
7	0111	0111	1101	1100	1010	1111	0100
8	1000	1000	1110	1101	1011	1110	1100
9	1001	1001	1111	1111	1100	1010	1101
10	1010						1111
11	1011						1110
12	1100						1010
13	1101						1011
14	1110						1001
15	1111						1000