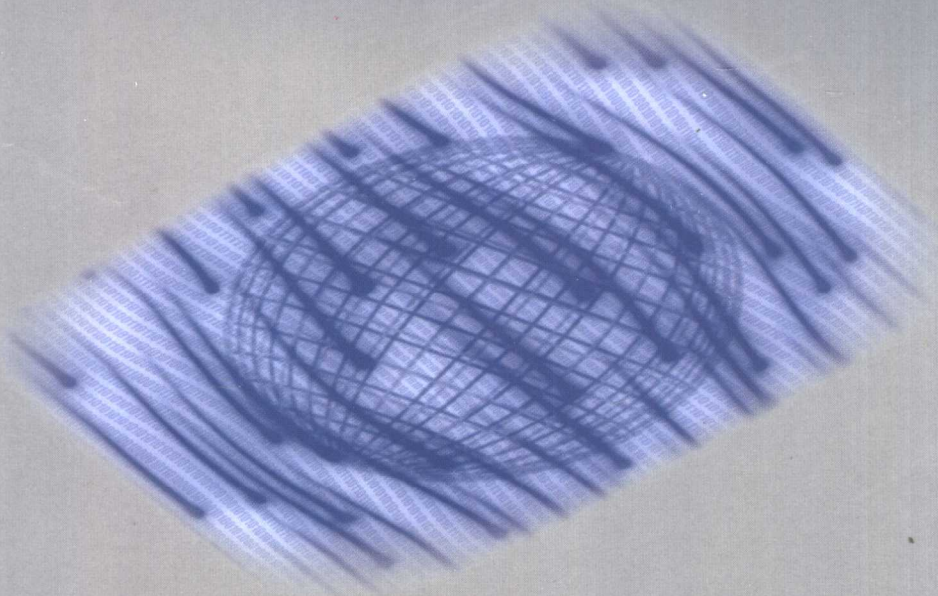


21 世纪信息通信系列教材

数字电路设计与数字系统

SHUZI DIANLU SHEJI YU SHUZI XITONG

刘培植 等编著



2-43



北京邮电大学出版社
www.buptpress.com

数字电路设计与数字系统

刘培植 等编著

北京邮电大学出版社

·北京·

内 容 简 介

本书系统地介绍了数字电路的基础知识,组合和时序电路的分析、设计方法,使读者对数字系统的构成及描述有较深入的了解,达到在具有较坚实的数字电路和数字系统理论知识的基础上,独立使用可编程逻辑器件、其他中小规模器件和计算机辅助分析、设计工具进行逻辑设计的能力。同时,本教材也为后续计算机原理及接口、数字通信等专业课程提供了必要的基础知识和概念。

本教材注重前后学习内容的连贯性,在对一般数字电路分析和设计基础理论讲述的基础上,给出了较多的分析设计实例,并根据数字电路和数字系统设计技术的发展,强调新技术的使用及分析问题和解决问题能力的培养。

本教材可以作为电子、信息类专业本科生教材和教学参考书,也可作为相关专业工程技术人员的参考和培训教材。

图书在版编目(CIP)数据

数字电路设计与数字系统/刘培植等编著. —北京:北京邮电大学出版社,2005

ISBN 7-5635-0994-1

I. 数… II. 刘… III. ①数字信号—电路设计—教材②数字系统—教材 IV. ①TN79②TP271

中国版本图书馆 CIP 数据核字(2004)第 134769 号

出 版 者:北京邮电大学出版社(北京市海淀区西土城路 10 号)

邮编:100876 电话:62282185 传真:62283578

电子信箱:publish@bupt.edu.cn

经 销:各地新华书店

印 刷:北京通州皇家印刷厂

印 数:1—5000 册

开 本:787 mm×1092 mm 1/16

印 张:22.25

字 数:537 千字

版 次:2005 年 2 月第 1 版 2005 年 2 月第 1 次印刷

ISBN 7-5635-0994-1/TP·155

定 价:32.00 元

如有印装质量问题请与北京邮电大学出版社发行部联系

前 言

随着计算机技术和数字技术的发展,现代电子设备已经从单纯用模拟电路实现大范围地向数字化转变。目前,大多只在模拟信号采集、微弱信号放大、高频大功率输出等局部电路采用模拟电路,其余部分广泛采用数字技术及数字处理电路。因此,对数字电路的分析与设计,成为电子工程技术人员必备的专业基础知识。

随着技术的进步,数字电路和数字系统的设计已由 20 世纪六七十年代的中、小规模集成逻辑部件的设计,发展到目前的大规模和超大规模集成电路以及专用集成电路的设计。集成电路工艺也已从 TTL 电路为主变为以 CMOS 电路为主。

数字逻辑电路的设计方法也在演变和发展,数字系统的设计从传统的单纯硬件设计方法,变为计算机软硬件辅助设计的方法,即电子设计自动化(EDA)和电子系统设计自动化(ESDA)成为现代电子系统设计和制造的主要技术手段。

为适应数字电路与系统分析和设计技术的发展,适应新一代电子、通信人才培养的需要,本书作者在总结多年教学经验的基础上,参考国内外相关资料,在保留传统“数字电路与逻辑设计”系统性的基础上,并根据相关院校教学大纲要求,重新编写本教材。本教材中主要强调数字电路和数字系统设计学习内容的前后连贯性,突出数字逻辑电路的基础理论、分析方法和设计方法的学习,增加了较多的设计实例介绍和应用方面的内容,特别是对可编程逻辑器件(PLD)和硬件描述语言(VHDL)作了相对详细的介绍。为读者独立分析、设计数字电路和数字系统,较快掌握分析设计工具,建立规范有序的思维习惯,提高分析和解决实际问题的能力打下了良好基础。

本书共分 10 章。第 1、2 章内容包括数制与编码、逻辑代数基础及逻辑函数的化简、逻辑门电路及特性,该部分由黄小军编写;第 3、4 章介绍组合电路的分析与设计方法,以及各种触发器的特点和参数,该部分由胡春静编写;第 5、6 章对用中小规模集成时序逻辑电路的分析与设计方法和应用进行了详细

地介绍,该部分由刘培植编写;第7、8章对常用的可编程逻辑器件及硬件描述语言及其应用作了介绍,该部分由郭琳编写;第9、10章介绍了数模变换原理以及数字系统的设计方法,该部分由纪阳编写。

由于作者水平和经验有限,书中难免会有错误和不足,希望读者给予批评指正。

作者

2004年12月1日

目 录

第 1 章 数字技术基础

1.1 数字信号和数字电路	1
1.2 数字数制与编码	1
1.2.1 数制的权和基数	1
1.2.2 数制转换	2
1.2.3 二-十进制编码	4
1.3 逻辑代数基础	5
1.3.1 基本概念	5
1.3.2 复合逻辑运算	7
1.3.3 逻辑代数的基本定律和规则	9
1.4 逻辑表达式的标准形式	11
1.4.1 最小项	11
1.4.2 最大项	12
1.4.3 最大项和最小项之间的关系	12
1.5 逻辑函数的化简	12
1.5.1 代数化简法	13
1.5.2 卡诺图化简法	14
习题	19

第 2 章 逻辑门电路(集成逻辑门电路)

2.1 数字集成电路的特点和分类	21
2.2 晶体管的开关特性	22
2.2.1 晶体二极管的开关特性	22
2.2.2 晶体三极管的开关特性	24
2.3 二极管逻辑门	25
2.4 反相器	27
2.5 TTL 集成逻辑门	30
2.5.1 TTL 与非门工作原理	31
2.5.2 三态输出门	33
2.5.3 TTL 电路使用中注意的问题	34

2.6	ECL 逻辑门	34
2.7	MOS 管开关特性	36
2.7.1	静态特性	36
2.7.2	动态特性	39
2.8	NMOS 逻辑门电路	40
2.8.1	NMOS 反相器	40
2.8.2	NMOS 门电路	43
2.9	CMOS 逻辑门电路	44
2.9.1	CMOS 反相器	44
2.9.2	CMOS 门电路	46
2.10	不同逻辑电平的配合	48
	习题	49

第 3 章 组合电路的分析与设计

3.1	组合逻辑电路的特点	53
3.2	组合逻辑电路的分析	53
3.3	小规模组合逻辑电路的设计	55
3.3.1	由设计要求列真值表	56
3.3.2	逻辑函数的两级门实现	56
3.3.3	逻辑函数的三级门实现	58
3.3.4	组合电路实际设计中的几个问题	61
3.3.5	组合电路设计实例	67
3.4	组合逻辑电路的冒险	73
3.4.1	逻辑冒险与消除方法	73
3.4.2	功能冒险与消除方法	77
3.4.3	冒险消除方法的比较	79
3.4.4	动态冒险	79
3.5	常用的中规模组合逻辑电路与应用	80
3.5.1	集成数码比较器	80
3.5.2	编码器(Encoder)与优先编码器(Priority Encoder)	84
3.5.3	译码器(Decoder)	88
3.5.4	数据选择器(Multiplexer)	96
3.5.5	数据分配器(DMUX)	102
3.5.6	奇偶校验与可靠性编码	104
3.5.7	运算电路	108
	习题	113

第 4 章 集成触发器

4.1	时序电路的特点	123
-----	---------------	-----

4.2	触发器的基本特性及其记忆作用	123
4.3	基本 RS 触发器	124
4.3.1	电路结构及工作原理	124
4.3.2	描述触发器(时序电路)的方法	125
4.3.3	基本 RS 触发器的特点	127
4.4	各种钟控触发器的逻辑功能	127
4.4.1	钟控 RS 触发器	127
4.4.2	钟控 D 触发器	128
4.4.3	锁存器	128
4.4.4	钟控 JK 触发器	129
4.4.5	钟控 T 触发器	130
4.4.6	各种触发器之间的转换	131
4.4.7	钟控触发器的缺点	131
4.5	TTL 集成主从触发器	132
4.5.1	基本工作原理	132
4.5.2	主从 JK 触发器的一次翻转	133
4.5.3	异步置 0 置 1 输入	134
4.5.4	主从触发器的特点	135
4.6	集成边沿触发器(Edge_triggered Flip-flop)	135
4.6.1	负边沿 JK 触发器	135
4.6.2	维持-阻塞 D 触发器	136
4.6.3	JK 触发器和 D 触发器的实际产品	137
4.6.4	触发器的逻辑符号	138
4.7	CMOS 触发器	139
4.7.1	CMOS 钟控 D 触发器	139
4.7.2	CMOS 主从 D 触发器	139
4.7.3	CMOS 主从 JK 触发器	141
4.8	集成触发器的选用和参数	141
4.8.1	逻辑功能的选择	141
4.8.2	触发方式的选择	141
4.8.3	触发器的参数	141
	习题	143

第 5 章 时序逻辑电路

5.1	概述	148
5.2	同步时序逻辑电路的分析	150
5.2.1	常用时序电路简介	150
5.2.2	同步时序逻辑电路的分析方法	152
5.2.3	一般同步时序电路分析举例	153

5.2.4	移位寄存器及其应用电路的分析	157
5.2.5	异步时序逻辑电路的分析方法	164
5.3	常用时序电路的设计	166
5.3.1	常用时序电路的设计步骤	166
5.3.2	同步计数器的设计	167
5.3.3	序列信号发生器	171
5.3.4	M序列发生器	173
5.4	一般时序逻辑电路的设计方法	174
5.4.1	一般同步时序逻辑电路的设计方法	174
5.4.2	采用小规模集成器件设计异步计数器	184
	习题	187

第6章 中规模时序集成电路及应用

6.1	中规模异步计数器	197
6.2	中规模同步计数器	199
6.3	中规模计数器的应用	203
6.3.1	中规模计数器构成任意进制计数器	203
6.3.2	中规模计数器的级联	207
6.3.3	计数器用于逻辑设计	210
6.4	中规模移位寄存器	212
6.4.1	中规模移位寄存器的功能	212
6.4.2	中规模移位寄存器介绍	213
6.5	中规模移存器的应用	215
6.5.1	中规模移存器的扩展	215
6.5.2	中规模移存器构成串—并变换器	215
6.5.3	中规模移存器构成并—串变换器	217
6.5.4	中规模移存器构成计数器	218
6.5.5	中规模移存器构成分频器	218
6.5.6	中规模移存器构成序列信号发生器	219
	习题	221

第7章 可编程逻辑器件

7.1	只读存储器	226
7.1.1	ROM可作为一种PLD器件	226
7.1.2	ROM的种类	228
7.2	PLA、PAL、GAL	231
7.2.1	PLA(Programmable Logic Array)	231
7.2.2	PAL(Programmable Array Logic)	232
7.2.3	GAL(Generic Array Logic)	232

7.3	EPLD	234
7.3.1	MAX7000 系列的系统结构	235
7.3.2	MAX7000 系列的 LAB 和 MC	235
7.3.3	MAX7000 系列的 I/O 控制块	237
7.4	CPLD/FPGA	238
7.4.1	FLEX10K 的系统结构	238
7.4.2	FLEX10K 的嵌入式阵列块(EAB)	240
7.4.3	FLEX10K 的逻辑阵列块(LAB)	241
7.4.4	FLEX10K 的逻辑单元(LE)	241
7.4.5	FLEX10K 的快速连线带(FastTrack)	245
7.4.6	FLEX10K 的输入/输出单元(IOE)	246
7.5	FPGA	247
7.5.1	XC4000 系列的可编程逻辑模块(CLB)	248
7.5.2	XC4000 系列的可编程互连资源	249
7.5.3	XC4000 系列的输入/输出功能块(IOB)	251
	习题	252

第 8 章 硬件描述语言 VHDL

8.1	电子设计自动化与硬件描述语言	254
8.1.1	电子设计自动化(EDA)	254
8.1.2	硬件描述语言(HDL)	255
8.2	VHDL 程序的基本结构	255
8.2.1	实体(ENTITY)	256
8.2.2	结构体(ARCHITECTURE)	258
8.2.3	VHDL 的库与程序包	260
8.2.4	配置(CONFIGURATION)	262
8.3	VHDL 的数据和运算符	263
8.3.1	VHDL 中的数据对象	263
8.3.2	VHDL 中的数据类型	265
8.3.3	VHDL 中数据的属性	269
8.3.4	VHDL 中的运算符	270
8.4	VHDL 程序的语句组织结构	271
8.4.1	块(BLOCK)语句结构	271
8.4.2	进程(PROCESS)语句结构	272
8.4.3	子程序(SUBPROGRAM)语句结构	275
8.4.4	元件例化(COMPONENT_INSTANT)语句结构	278
8.5	VHDL 的主要描述语句	282
8.5.1	并行描述语句	282
8.5.2	顺序描述语句	285

8.6 用 VHDL 解决组合逻辑和时序逻辑问题	292
8.6.1 用 VHDL 解决组合逻辑问题	292
8.6.2 用 VHDL 解决时序逻辑问题	293
习题	296
第 9 章 数模和模数转换	
9.1 数模转换器 DAC	300
9.1.1 转换原理和一般组成	300
9.1.2 R-2R 倒 T 型电阻网络 DAC	301
9.1.3 DAC 的主要技术指标	302
9.2 模数转换器	303
9.2.1 模数转换器的基本原理	303
9.2.2 并联比较型 ADC	305
9.2.3 逐次比较型 ADC	307
9.2.4 ADC 的主要技术指标	308
习题	308
第 10 章 数字系统设计	
10.1 数字系统设计概述	309
10.1.1 数字系统基本组成	309
10.1.2 传统数字系统设计方法	310
10.1.3 现代数字系统设计方法	310
10.1.4 现代数字系统的设计流程	310
10.2 ASM 图和 MDS 图	311
10.2.1 ASM 图	311
10.2.2 MDS 图	313
10.2.3 状态图到 MDS 图	314
10.2.4 ASM 图至 MDS 图的转换	315
10.3 数字系统设计实例	316
习题	319
附录 1 国产半导体集成电路型号命名法	321
附录 2 集成电路主要性能参数	323
附录 3 二进制逻辑单元图形符号说明	325
参考文献	343

第 1 章 数字技术基础

本章主要介绍数字技术的基础知识,数字电路中常用的计数制,以及分析与设计数字逻辑电路的理论基础——逻辑代数。

1.1 数字信号和数字电路

自然界中的许多物理量(如时间、距离、温度等)在时间和数值上都有连续变化的特点。它们可在一定范围内取任意实数,我们称这类量为模拟量。工程上常用电压来模拟这些实际的物理量,故称之为模拟信号,其电压在一定范围内是连续变化的。例如正弦变化的交流信号,它在某一瞬间的值可以是一个数值区间内的任何值。处理模拟信号的电路被称为模拟电路(Analog Circuit)。

还有一种物理量,它们的数值大小和每次的增减变化都是某一个最小单位的整数倍,而小于这个最小单位的数值是没有意义的。这一类物理量叫做数字量,把表示数字量的信号称为数字信号。处理数字信号的电路称为数字电路(Digital Circuit)。在数字电路中只有 0 和 1 两种数值组成的数字信号,一个 0 或一个 1 通常称为 1 比特,可以用电位的高低来表示,在电路上很容易实现。

1.2 数字数制与编码

本节从我们习惯的十进制开始,分析、推导各种不同数制之间的转换关系。重点将讨论在数字系统中广泛采用的二进制及其常用编码。

1.2.1 数制的权和基数

数制是进位记数制的简称。记数符号的个数称为基数。

日常生活中常用的是十进制,它有 0,1,2,⋯,9 等 10 个不同的数码,即基数为 10,而各个数码处于十进制的不同数位时,所代表的数值不同。例如:8 921 读做八千九百二十一。八在千位上,读做八千;九在百位上,读做九百;二在十位上,读做二十;一在个位上,读做一。这是一种位置记数法,即把千、百、十、个称为权,8、9、2、1 称为系数,每个系数所处的位置不同,则对应的权值不同。可以将其表示为

$$(8\ 921)_{10} = (8 \times 10^3 + 9 \times 10^2 + 2 \times 10^1 + 1)_{10}$$

其中下脚标 10 代表十进制,它的一般表示为

$$(N)_{10} = (a_{n-1} \times 10^{n-1} + a_{n-2} \times 10^{n-2} + \dots + a_1 \times 10^1 + a_0 \times 10^0 + a_{-1} \times 10^{-1} + \dots + a_{-m} \times 10^{-m})_{10}$$

公式中 n 为整数的位数; m 为小数的位数; a_i 为 $0 \sim 9$ 十个数码中任意一个; 10 称为十进制的基数。

在数字系统中常用二进制, 基数为 2, 每位的权是 2 的幂(见表 1.2.1)。

表 1.2.1 二进制各位的权

二进制位数	权	十进制表示	二进制位数	权	十进制表示	二进制位数	权	十进制表示
13	2^{12}	4 096	7	2^6	64	1	2^0	1
12	2^{11}	2 048	6	2^5	32	-1	2^{-1}	0.5
11	2^{10}	1 024	5	2^4	16	-2	2^{-2}	0.25
10	2^9	512	4	2^3	8	-3	2^{-3}	0.125
9	2^8	256	3	2^2	4	-4	2^{-4}	0.062 6
8	2^7	128	2	2^1	2	-5	2^{-5}	0.031 25

例如

$$(10101.01)_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2}$$

可将进位制的规律推广到任意进位制 R , 表达如下: R 进制有 R 个数码, 以 R 为基数, 逢 R 进 1, 其按权展开式为

$$(N)_R = k_{n-1} \times R^{n-1} + k_{n-2} \times R^{n-2} + \dots + k_1 \times R^1 + k_0 \times R^0 + k_{-1} \times R^{-1} + k_{-2} \times R^{-2} + \dots + k_{-m} \times R^{-m} = \sum_{i=-m}^{n-1} k_i \times R^i$$

1.2.2 数制转换

1. 二进制与八进制及十六进制之间的转换

(1) 二进制数和八进制数之间的转换

八进制的基数是 2 的幂, 因此二进制和八进制的互换非常容易。二进制要转换为八进制时, 只需要将其以小数点为中心, 向两边每三位分成一组, 不足三位时补 0。再把每三位二进制数对应的八进制数码写出即可。

例 1.2.1 将 $(11101.1101)_2$ 转换成八进制数。

解: 二进制数 011 101.110 100

八进制数 3 5 . 6 4

转换结果为 $(11101.1101)_2 = (35.64)_8$ 。

若要将八进制数转换成二进制数, 只要写出每八进制数码对应的二进制数, 依次排好即可。

例 1.2.2 将 $(234.567)_8$ 转换成二进制数。

解: 八进制数 2 3 4 . 5 6 7

二进制数 010 011 100.101 110 111

转换结果为 $(234.567)_8 = (10011100.101110111)_2$

(2) 二进制数和十六进制数之间的转换

用四位二进制数可表示 0~15 十六个数, 所以与八进制类似。只需要将其以小数点为中心, 向两边每四位分成一组, 不足四位时补 0。再把每四位二进制数对应的十六进制数码写出即可。

例 1.2.3 将 $(11101.1101)_2$ 转换成十六进制数。

解: 二进制数 0001 1101.1101

十六进制数 1 D.D

转换结果为 $(11101.1101)_2 = (1D.D)_{16}$

若要将十六进制转换成二进制数, 只要写出每十六进制数码对应的二进制数, 依次排好即可。

例 1.2.4 将 $(AF.26)_{16}$ 转换成二进制数。

解: 十六进制数 A F . 2 6

二进制数 1010 1111.0010 0110

转换结果为 $(AF.26)_{16} = (10101111.00100110)_2$

表 1.2.2 列出了与十进制对应的二进制、八进制、十六进制。

表 1.2.2 与十进制对应的二进制、八进制、十六进制

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	十进制	二进制	八进制	十六进制
0	0000	0	0	6	0110	6	6	12	1100	14	C
1	0001	1	1	7	0111	7	7	13	1101	15	D
2	0010	2	2	8	1000	8	8	14	1110	16	E
3	0011	3	3	9	1001	11	9	15	1111	17	F
4	0100	4	4	10	1010	12	A	16	10000	18	10
5	0101	5	5	11	1011	13	B				

2. 十进制与非十进制之间的转换

(1) 非十进制数转换为十进制数

任何一个数都可用其权展开式表示为

$$(N)_r = \sum_{i=-m}^{n-1} a_i r^i$$

于是将一非十进制数按权展成一多项式, 每项是该位数码与相应权值的乘积, 把项式中的数码和权用等值十进制表示即可。

例 1.2.5 将 $(11010.011)_2$ 转换成十进制数。

$$\begin{aligned} \text{解: } (11010.011)_2 &= 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} \\ &= 16 + 8 + 0 + 2 + 0 + 0 + 0.25 + 0.125 \\ &= (26.375)_{10} \end{aligned}$$

例 1.2.6 将 $(12AF.B4)_{16}$ 转换成十进制数。

$$\begin{aligned} \text{解: } (12AF.B4)_{16} &= 1 \times 16^3 + 2 \times 16^2 + 10 \times 16^1 + 15 \times 16^0 + 11 \times 16^{-1} + 4 \times 16^{-2} \\ &= 4096 + 512 + 160 + 15 + 0.6875 + 0.015625 \\ &= (4783.703125)_{10} \end{aligned}$$

(2) 十进制数转换为非十进制数

① 整数部分的转换

整数部分的转换采用基数除法。所谓基数除法,即用目的数制的基数去除十进制整数,第一次除所得的余数为目的数的最低位,得到的商再除以该基数,所得的余数为目的数的次低位,依次类推,直到商为0,所得的余数为目的数的最高位。

例 1.2.7 将 $(53)_{10}$ 转换成二进制数。

解:

2	53	余数
2	26	1 最低位(LSB)
2	13	0
2	6	1
2	3	0
2	1	1
	0	1 最高位(MSB)

所以转换结果为 $(53)_{10} = (110101)_2$ 。

② 小数部分的转换

小数部分的转换采用基数乘法。所谓基数乘法,即用该小数乘目的数制的基数,第一次乘的结果的整数部分为目的数小数的最高位,其小数部分再乘基数,所得结果的整数部分为目的数小数的次高位,依次类推,直到小数部分为0或达到要求精度为止。

例 1.2.8 将十进制小数 $(0.6875)_{10}$ 转换成二进制数。

解:

$0.6875 \times 2 = 1.3750$	1 最高位
$1.3750 \times 2 = 0.75$	0
$0.750 \times 2 = 1.5$	1
$1.5 \times 2 = 1$	1 最低位

所以转换结果为 $(0.6785)_{10} = (0.1011)_2$

1.2.3 二-十进制编码

编码即用若干特定的二进制编码来表示自然数、字母和符号的过程。这些特定的二进制数码称为字符的代码。这里必须特别指出的是,二进制码不一定表示二进制数。

在计算机中,十进制数除了换算成二进制数外,还可以直接用十进制数进行输入和计算,这种方法就是将十进制的十个数,分别用不少于四位的特定二进制数码表示,我们称为二-十进制编码。

下面介绍几种常用的二-十进制编码:

(1) 8421BCD 码

8421BCD 码是最常用的二-十进制编码。它用 0000~1001 来表示十进制的 0~9。在表示十进制的四位二进制代码中,由高到低权值分别为 8、4、2、1。这种每位二进制有确定权值的编码叫做有权码。

例 1.2.9 $(6)_{10} = (0110)_{8421BCD}$

(2) 2421 码

2421 码也是一种有权码,其四位二进制由高到低分别代表 2、4、2、1。

例 1.2.10 $(6)_{10} = (1100)_{2421}$

(3) 余 3 码

余 3 码也有四位,但每位的权是不固定的,故是无权码。它可以由每个 8421BCD 码加上 $(3)_{10} = (11)_2$ 得到,余 3 码因此而得名。

(4) 余 3 格雷码

余 3 格雷码是无权码,它的特点是在任意两个相邻的数之间(包括 0 与 9)仅有一位不同。

(5) 循环码

循环码的特点是任何相邻的两个码字中仅有一位代码不同,其他代码是一样的。循环码又叫单位距离码,它的这一单位距离性,能避免在码组的转换过渡过程中产生瞬时误码。因此循环码在通信和测量技术中得到了广泛的应用。循环码也是一种无权码,它的编码方案很多,表 1.2.3 列出了其中一种。

以上各种 BCD 编码可以参见表 1.2.3。

表 1.2.3 常见的 BCD 编码

十进制码	8421BCD 码	2421 码	余 3 码	余 3 格雷码	循环码
0	0000	0000	0011	0010	0000
1	0001	0001	0100	0110	0001
2	0010	0010	0101	0111	0011
3	0011	0011	0110	0101	0010
4	0100	0100	0111	0100	0110
5	0101	1011	1000	1100	0111
6	0110	1100	1001	1101	0101
7	0111	1101	1010	1111	0100
8	1000	1110	1011	1110	1100
9	1001	1111	1100	1010	1000

1.3 逻辑代数基础

逻辑代数(Logic Algebra)又名布尔代数(Boolean Algebra),它是按一定逻辑规律进行的代数,是分析逻辑设计的理论基础。本节简单介绍逻辑代数的基本运算和公式,逻辑代数的表示和其化简的方法。

1.3.1 基本概念

1. 逻辑变量

逻辑代数是用来处理逻辑运算的代数。其变量称为逻辑变量,并用大写字母 A、B、C

等表示。它有两种取值,即逻辑0和逻辑1。0和1称为逻辑常量。这种只有两个逻辑变量的逻辑代数是二元布尔代数。本书的逻辑代数是指二值逻辑。

在这里,0和1是表示事物矛盾双方的符号。例如,命题的真假,信号的有无,电位的高低。所以逻辑0和逻辑1本身没有数值的意义。

2. 基本逻辑运算

基本的逻辑运算有与、或、非三种,任何复杂的运算都可由这三种基本逻辑运算来实现。

(1) “与”运算

“与”运算又称为逻辑乘。它表示:当且仅当决定事件发生的条件全部为真时,该事件才为真。“与”逻辑可用如下逻辑表达式表示

$$F = A \times B = A \cdot B = AB$$

“·”表示逻辑乘,可以省去。

图 1.3.1 所示为与门逻辑符号。对于与门,仅当所有的输入都为1时,输出才为1;而只要输入有一个0,输出便是0。

“与”逻辑关系还可以用输入输出逻辑变量的各种取值组合和对应函数值关系的表格形式表示。这种反应输入变量和输出函数值关系的表格称为函数的真值表。假定决定事件具备用“1”表示,条件不具备用“0”表示,则“与”逻辑的真值表见表 1.3.1。

表 1.3.1 “与”逻辑的真值表

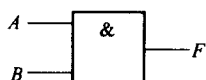


图 1.3.1 与门

A	B	F	A	B	F
0	0	0	1	0	0
0	1	0	1	1	1

(2) “或”运算

“或”运算又称为逻辑加。它表示:当且仅当决定事件发生的条件只要有一件发生,该事件为真。“或”逻辑可用如下逻辑表达式表示

$$F = A + B$$

图 1.3.2 所示为或门逻辑符号。对于或门,当所有的输入至少有一个为1时,输出就为1。

或逻辑的真值表见表 1.3.2。

表 1.3.2 “或”逻辑的真值表

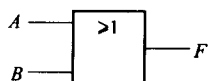


图 1.3.2 或门

A	B	F	A	B	F
0	0	0	1	0	1
0	1	1	1	1	1

(3) “非”运算

“非”运算又称为“反相”运算,或称为“求补”运算。它表示:当且仅当决定事件发生的条件发生时,该事件为假。“非”逻辑可用如下逻辑表达式表示

$$F = \bar{A}$$

图 1.3.3 所示为非门逻辑符号。对于非门,当输入为1时,输出就为0。