通·信·与·控·制·系·列·教·程

数字电路原理 设计与实践教程

DIGITAL CIRCUITS THEORY DESIGN & PRACTICE LECTURES

主 编 王革思

COMMUNICATION & CONTROL

哈尔滨工程大学出版社
Harbin Engineering University Press

通信与控制系列教程

数字电路原理、设计与实践教程

DIGITAL CIRCUITS THEORY DESIGN & PRACTICE LECTURES

主　编　王革思
副主编　解　武
主　审　赵旦峰

哈尔滨工程大学出版社

内容简介

本书在介绍数字电路基本原理、分析与设计方法等基础知识的同时，选择部分代表当今数字电子技术发展水平的新技术和新方法作为教学内容，在保证基本概念、基本理论和基本方法明确的前提下，力求内容精练，重点突出，定位准确，技术先进，有较强的可读性和操作性。本书理论与实践紧密结合，注重培养学生的设计能力、动手能力、创新能力及掌握先进技术能力，学以致用。对于那些学过基本理论而缺乏实践经验的学生和工程技术人员来说，可以快速地提高理论应用水平。

全书分为五大部分，共有15章，主要内容包括数字电路基础、可编程逻辑器件原理和应用、电子设计自动化、数字电路系统设计和工程应用设计实例。附录部分介绍了JDS-1数字电路教学实验箱的组成、原理和使用，以及74系列和4000系列数字集成电路资料，可供设计或实验时参考。

本书可作为普通高等院校电子工程、通信工程、工业自动化、计算机应用技术等信息工程类及相关专业的本科生教材，也可作为教师和工程技术人员的参考书。

图书在版编目(CIP)数据

数字电路原理、设计与实践教程/王革思主编.—哈尔滨:哈尔滨工程大学出版社,2007.4
ISBN 978-7-81073-953-5

Ⅰ.数…　Ⅱ.王…　Ⅲ.数字电路:逻辑电路-高等学校-教材　Ⅳ.TN79

中国版本图书馆CIP数据核字(2007)第019524号

出版发行　哈尔滨工程大学出版社
社　　址　哈尔滨市南岗区东大直街124号
邮政编码　150001
发行电话　0451-82519328
传　　真　0451-82519699
经　　销　新华书店
印　　刷　黑龙江省地质测绘印制中心印刷厂
开　　本　787mm×1 092mm　1/16
印　　张　15
字　　数　326千字
版　　次　2007年4月第1版
印　　次　2007年4月第1次印刷
印　　数　1—3 000册
定　　价　20.00元
http://press.hrbeu.edu.cn
E-mail:heupress@hrbeu.edu.cn

前　　言

21世纪是信息时代，以数字化为基本特征。目前，数字电子技术已广泛应用于电子、通信、计算机、自动控制、广播、电视等各领域，已成为这些领域中的工程技术人员必须掌握的基本理论和技能。集成电路技术的飞速发展，数字系统设计方法的不断推陈出新以及电子设计自动化技术的应用，使设计者自行设计和实现复杂的数字系统已经成为现实。技术的进步和更新必然要求教学内容和形式做出相应的调整。

本书是一部理论与实践紧密结合的教材，适应数字电子技术发展的新形势，同时满足当前教学改革的需要。这本书结合了编者多年的理论教学、实验教学和科研方面的经验，并吸收了其他同类书籍的优点，能够引导学生从学习理论知识向解决实际问题的过渡。本书编写本着学以致用的原则，在介绍数字电路基本原理、分析与设计方法等基础知识的同时，还选择部分代表当今数字电子技术发展水平的新技术和新方法作为教学内容，在保证基本概念、基本理论和基本方法明确的前提下，力求内容精练，重点突出，定位准确，技术先进，有较强的可读性和可操作性。

本书具有以下特点。

1.深入浅出。本书内容力求通俗易懂，层次结构上由浅入深，分析与设计方法灵活多样，集知识性、实践性和趣味性于一体，引导、启发学生的主动性，使学生比较容易接受、掌握和应用。

2.内容实用。本书针对必须熟练掌握的理论知识安排了相应的实例练习，结合实例讲解理论，以深化对理论的理解，使理论源于实践，又进一步指导实践。参考书中的内容，学生可自行完成一定难度的数字电路设计。

3.注重能力培养。本书将可编程逻辑器件和电子设计自动化等先进的数字电路设计技术引入教学实践环节，通过数字电路的设计、仿真及下载过程，既是对学生掌握理论知识水平的综合检验，也是训练和培养学生的设计能力、动手能力、创新能力及掌握先进技术的能力，学以致用。

4.解决工程问题。本书给出的实例均与实际问题有关，部分实例的技术性能已接近或达到工程应用水平。

全书分为五大部分，共有15章，主要内容包括数字电路基础、可编程逻辑器件原理和应用、电子设计自动化、数字电路系统设计和工程应用设计实例。附录部分介绍了JDS－1数字电路教学实验箱的组成、原理和使用，以及74系列和4000系列数字集成电路资料，可供设计或实验时参考。

本书由王革思任主编，解武任副主编。其中，解武编写了第8，11，12章、附录B和附录C；潘大鹏编写了附录A；王革思编写了其余各章。最后由王革思统稿。由赵旦峰教授审阅全稿。

在本书中引用了一些文献的相关内容，在此向这些书的作者致以诚挚的谢意！由于编者水平有限，不足之处在所难免，希望读者提出宝贵意见。

编　者

2007年3月

目　录

数字电路基础

可编程逻辑器件原理和应用

电子设计自动化

数字电路系统设计

工程应用设计实例

数字电路基础

1 逻辑代数基础知识

1.1 数制和码制

1.1.1 数制

在数字电路中经常使用的计数进制有十进制、二进制和十六进制等。这些不同的进制表示不同数制。我们把多位数码中每一位的构成方法以及从低位到高位的进位规则称为数制。

(1)十进制

位的构成:0~9,10个数码。

进位规则:逢十进一。

基数:10。

任何一个十进制数展开式为

$$D = \sum k_i \times 10^i \tag{1-1}$$

例如:

$$(168.75)_{10} = 1\times10^2 + 6\times10^1 + 8\times10^0 + 7\times10^{-1} + 5\times10^{-2}$$

若将式(1-1)中的基数10用任意数 N 取代,可得 N 进制数展开式为

$$D = \sum k_i \times N^i \tag{1-2}$$

在式(1-2)中,k_i 是第 i 位的系数;N^i 是第 i 位的权。若整数部分的位数是 n,小数部分的位数是 m,则 i 由大到小包含从 $n-1$ 到0的所有正整数和从 -1 到 $-m$ 的所有负整数。

(2)二进制

位的构成:0和1,2个数码。

进位规则:逢二进一。

基数:2。

根据式(1-2),任何一个二进制数展开式为

$$D = \sum k_i \times 2^i \tag{1-3}$$

(3)十六进制

位的构成:0~9和A~F,16个数码。

进位规则:逢十六进一。

基数:16。

其中：A ~ F 分别表示数码 10 ~ 15。根据式(1 - 2)，任何一个十六进制数展开式为

$$D = \sum k_i \times 16^i \tag{1-4}$$

1.1.2　数制转换

(1)二 - 十转换

把二进制数转换为等值的十进制数称为二 - 十转换。转换时只要将已知的二进制数按式(1 - 3)展开，然后把所有各项数值按十进制数相加，就可以得到相应等值的十进制数。

例如：

$$(1101.11)_2 = 1\times2^3 + 1\times2^2 + 0\times2^1 + 1\times2^0 + 1\times2^{-1} + 1\times2^{-2} = (13.75)_{10}$$

(2)十 - 二转换

十 - 二转换有两种方法。第一种方法是通过查 2^i 权值表完成转换。2^i 权值见表 1 - 1。

表 1 - 1　常用的 2^i 权值表

i	2^i 权值		i	2^i 权值	
-3	2^{-3}	0.125	5	2^5	32
-2	2^{-2}	0.25	6	2^6	64
-1	2^{-1}	0.5	7	2^7	128
0	2^0	1	8	2^8	256
1	2^1	2	9	2^9	512
2	2^2	4	10	2^{10}	1 024
3	2^3	8	11	2^{11}	2 048
4	2^4	16	12	2^{12}	4 096

例如：

$$\begin{aligned}(367.625)_{10} &= 256+64+32+8+4+2+1+0.5+0.125\\ &= 1\times2^8+1\times2^6+1\times2^5+1\times2^3+1\times2^2+1\times2^1+1\times2^0+1\times2^{-1}+1\times2^{-3}\\ &= (101101111.101)_2\end{aligned}$$

第二种方法是通过乘、除运算法实现转换，即十进制数的整数部分用除法转换，小数部分用乘法转换。这种方法比较麻烦，具体情况可参考有关书籍。

(3)二 - 十六转换

把二进制数转换成等值的十六进制数称为二 - 十六转换。转换时以二进制数小数点的位置为界限，整数部分从低位到高位将每 4 位二进制数分为一组，小数部分从高位到低位也是将每 4 位二进制数分为一组，位数不够补 0，并代之等值的十六进制数，即可得到对应的十六进制数。

例如：将 $(1011100.10101)_2$ 转换成等值的十六进制数。

$$(\underbrace{0101}_{5}\underbrace{1100}_{C}.\underbrace{1010}_{A}\underbrace{1000}_{8})_2=(5C.A8)_{16}$$

(4)十六－二转换

把十六进制数转换成等值的二进制数称为十六－二转换。转换时只需将十六进制数的每一位用等值的4位二进制数代替即可。

例如：

$$(A8F.6D)_{16}=(101010001111.01101101)_2$$

(5)十六－十转换

把十六进制数转换成等值的十进制数称为十六－十转换。转换时只要将已知的十六进制数按式(1－4)展开，然后把所有各项数值按十进制数相加，就可以得到相应等值的十进制数。

例如：

$$(2A.F1)_{16}=2\times16^1+10\times16^0+15\times16^{-1}+1\times16^{-2}\approx(42.94)_{10}$$

此外，要将十进制数转换为等值的十六进制数时，可以先将其转换成二进制数，然后再将得到的二进制数转换成十六进制数即可。

1.1.3 二进制数的表示方式

在数字电路和计算机中使用的二进制数有两种表示形式，一种是无符号的二进制数，另一种是有符号的二进制数。在有符号的二进制数中其最高位作为符号位，正数为0，负数为1，余下的各位表示数值。

有符号的二进制数有三种表示方式：原码、反码和补码。

正数的反码、补码与它的原码相同。负数的反码通过将原码的数值位逐位取反得到，负数的反码的最低位加1可得到负数的补码。

例如：分别将＋89和－89写成它们的二进制数的原码、反码和补码形式。

$$(+89)_{10}=(01011001)_{原}=(01011001)_{反}=(01011001)_{补}$$

$$(-89)_{10}=(11011001)_{原}=(10100110)_{反}=(10100111)_{补}$$

注意，在转换过程中符号位应保持不变。

例1－1 计算$(1001)_2-(0101)_2$

解 方法一：根据二进制数减法运算规则，可知

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

方法二：采用补码加法运算

$$(1001)_2-(0101)_2=(+1001)_2+(-0101)_2=(01001)_{补}+(11011)_{补}$$

$$\begin{array}{rr} & 0\ 1\ 0\ 0\ 1 \\ + & 1\ 1\ 0\ 1\ 1 \\ \hline 舍去\leftarrow 1 & \boxed{0}\ 0\ 1\ 0\ 0 \end{array}$$

符号位

两种方法运算结果一样。因此，利用补码可以将二进制数的减法运算转化成加法运算。此外，乘法运算可以用加法和移位两种操作实现，除法运算可以用减法和移位两种操作实现，因此二进制数的四则运算都可以用加法运算电路实现，大大简化电路的结构。

1.1.4 码制

用 4 位二进制数表示 1 位十进制数 0 ~ 9 这十个状态时要遵循一定的规则,这些规则称为码制。这些 4 位二进制数码组成的代码称为二 – 十制代码(Binary Coded Decimal,简称 BCD 代码)。表 1 – 2 列出了几种常见的 BCD 代码,它们的编码规则各不相同。

8421 码是数字电路中最常用的一种 BCD 代码。这种码中每一位的 1 都代表一个固定数值。因为代码中从左到右每一位的 1 分别表示 8,4,2,1,所以把这种码叫做 8421 码。由于 8421 码的每一位权是固定不变的,它属于恒权代码。

若把每一个余 3 码看作 4 位二进制数,则它的数值要比它表示的十进制数多 3,故而将这种代码叫做余 3 码。将两个余 3 码做十进制加法运算时,若两数之和为 10,正好等于二进制数的 16,于是便从高位自动产生进位信号。此外,0 和 9、1 和 8、2 和 7、3 和 6、4 和 5 的余 3 码互为反码。余 3 码是变权代码。

2421 码是一种恒权代码,它的 0 和 9、1 和 8、2 和 7、3 和 6、4 和 5 也互为反码。

余 3 循环码也是一种变权代码。在余 3 循环码中两个相邻的代码之间仅有一位状态不同,因此,按照余 3 循环码状态接成的计数器,每次状态转换过程中只有一个触发器翻转。对余 3 循环码译码时不会发生竞争 – 冒险现象。

表 1 – 2　几种常见的 BCD 码

编码种类 / 十进制数	8421 码	余 3 码	2421 码	余 3 循环码
0	0000	0011	0000	0010
1	0001	0100	0001	0110
2	0010	0101	0010	0111
3	0011	0110	0011	0101
4	0100	0111	0100	0100
5	0101	1000	1011	1100
6	0110	1001	1100	1101
7	0111	1010	1101	1111
8	1000	1011	1110	1110
9	1001	1100	1111	1010
权	8421		2421	

1.2 逻辑代数基本运算

逻辑代数是描述客观事物逻辑关系的数学方法。只有两种对立逻辑状态的逻辑关系称二值逻辑,例如开关的通与断、灯的亮与灭、事情的正确与错误等。在二值逻辑代数中,用字母表示逻辑变量,逻辑变量的取值只有 0 和 1 两种情况。这里的 0 和 1 已不再表示数值的大小,只代表两种不同的逻辑状态。

1.2.1 与逻辑运算

与逻辑运算真值表见表 1-3。根据表 1-3,可得与逻辑运算公式为

$$Y = A \cdot B \tag{1-5}$$

式中,A,B 是输入变量,Y 是输出变量。与逻辑运算特点:当输入全为 1 时,输出为 1,否则为 0。

1.2.2 或逻辑运算

或逻辑运算真值表见表 1-4。根据表 1-4,可得或逻辑运算公式为

$$Y = A + B \tag{1-6}$$

或逻辑运算特点:只要有一个输入为 1 时,输出为 1,否则为 0。

1.2.3 非逻辑运算

非逻辑运算真值表见表 1-5。根据表 1-5,可得非逻辑运算公式为

$$Y = \overline{A} \tag{1-7}$$

、

非运算也叫做取反运算。

表 1-3 与逻辑运算真值表

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

表 1-4 或逻辑运算真值表

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

表 1-5 非逻辑运算真值表

A	Y
0	1
1	0

1.2.4 常用的复合逻辑运算

(1)与非逻辑运算

$$Y = \overline{A \cdot B} \tag{1-8}$$

可理解成先与后非。与非逻辑运算特点:只要有一个输入为 0 时,输出为 1,否则为 0。

(2)或非逻辑运算

$$Y = \overline{A + B} \tag{1-9}$$

可理解成先或后非。或非逻辑运算特点:当输入全为 0 时,输出为 1,否则为 0。

(3)异或逻辑运算

$$Y = A \oplus B = A \cdot \overline{B} + \overline{A} \cdot B \tag{1-10}$$

异或逻辑运算特点:当两个输入不同时,输出为 1,否则为 0。

(4)同或逻辑运算

$$Y = A \odot B = A \cdot B + \overline{A} \cdot \overline{B} \tag{1-11}$$

同或逻辑运算特点:当两个输入相同时,输出为 1,否则为 0。

异或和同或互为反运算,即

$$A\oplus B=\overline{A\odot B}\quad 或\quad A\odot B=\overline{A\oplus B} \tag{1-12}$$

实现各种逻辑运算的单元电路称为门电路。几种常用的门电路图形符号如图1-1所示,其中非门也叫做反相器。图中上边一行是目前国家标准局规定的符号,下边一行常见于国外一些书籍、资料以及EDA软件上。在各种门电路中与非门和或非门功能最强,可以替代其他类型的门电路。门电路又是组成中规模数字集成电路、大规模数字集成电路的基本单元电路。

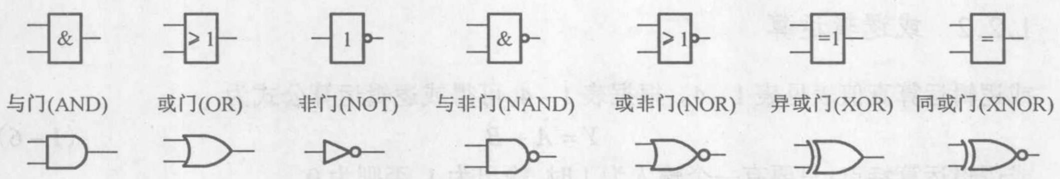

图1-1　几种常用的门电路图形符号

1.3　逻辑函数及其表示方法

从前面介绍的逻辑关系可以看出,当输入变量取值确定以后,输出的取值便随之而定。因此,输出与输入之间存在一种函数关系,这种函数关系称为逻辑函数,写作

$$Y=F(A,B,C,\cdots)$$

1.3.1　逻辑函数表示方法

逻辑函数的常用表示方法有逻辑真值表(简称真值表)、逻辑函数式(又称逻辑式或函数式)、逻辑图、卡诺图和时序图。

例1-2　设计一个奇、偶判断电路,它有三个输入端 A,B,C,一个输出端 Y,如图1-2所示。当 A,B,C 状态为1的个数为偶数时,Y 为1,否则 Y 为0。用逻辑函数的5种形式分别表示该电路。

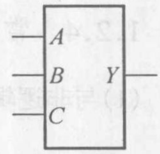

图1-2　奇、偶判断电路管脚排列图

(1)逻辑真值表

将输入变量所有取值的组合($2^3=8$个)及其对应的输出值列成表,即可得表示逻辑函数关系的逻辑真值表,见表1-6。

(2)逻辑函数式

在表1-6中,输入变量取值为1时,用原变量表示,取值为0时,用反变量表示,并将 Y 为1时对应的输入变量的乘积项相加,得到逻辑函数式为

$$Y=\bar{A}BC+A\bar{B}C+AB\bar{C} \tag{1-13}$$

式(1-13)是由几个乘积项相加组成的,把这种形式的逻辑函数式称为与-或逻辑式。

(3)逻辑图

将式(1-13)中的与、或、非运算符号用图形符号表示出来,就可以画出表示逻辑函数关系的逻辑图,如图1-3所示。

表 1－6 奇、偶判断电路逻辑真值表

输 入			输 出
A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

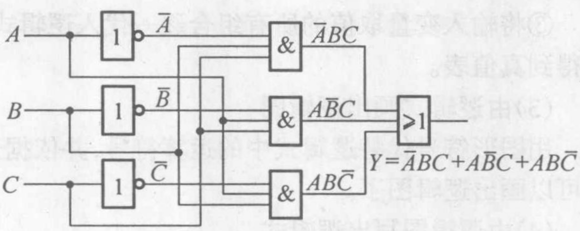

图 1－3 奇、偶判断电路逻辑图

(4)卡诺图

奇、偶判断电路的卡诺图如图 1－4 所示。从图中可以看出有三个小方块里添 1,代表了式(1－13)中的三个乘积项。

(5)时序图

为了便于用动态观察的方法检查数字电路的逻辑功能,输出变量 Y 与输入变量 A,B,C 之间的逻辑函数关系可以用时间波形表示,这种波形图称为时序图。奇、偶判断电路的时序图如图 1－5 所示。

$A \backslash BC$	00	01	11	10
0			1	
1		1		1

图 1－4 奇、偶判断电路卡诺图

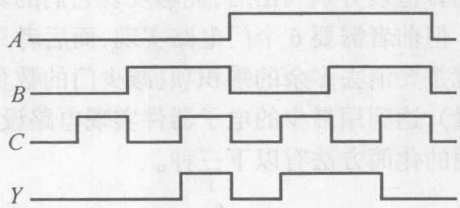

图 1－5 奇、偶判断电路时序图

1.3.2 各种表示方法间的互相转换

通过例 1－2 可以知道同一个逻辑函数可以用五种不同的方法表示,那么这些方法之间是可以互相转换的。经常用到的转换方式有以下几种。

(1)由真值表写出逻辑式

转换的步骤:

①找出真值表中使输出 Y 为 1 的那些输入变量取值的组合。

②每组输入变量取值的组合对应一个乘积项,其中取值为 1 的写入原变量,取值为 0 的写入反变量。

③将这些乘积项相加,即得 Y 的逻辑式。

(2)由逻辑式列出真值表

转换的步骤:

①画出真值表，将已知逻辑式的输入变量列在表的左侧，输出变量列在表的右侧。

②在输入变量列中添上输入变量取值的所有组合，组合数量为 2^n，其中 n 为输入变量的个数。

③将输入变量取值的所有组合逐一代入逻辑式中，得到一组输出变量值并填入表中，即可得到真值表。

(3)由逻辑式画出逻辑图

用图形符号代替逻辑式中的运算符号，并依据运算优先顺序把这些图形符号连接起来就可以画出逻辑图了。

(4)由逻辑图写出逻辑式

从电路的输入端到输出端逐级写出每个图形符号对应的逻辑式，就可以得到该电路对应的逻辑式了。

从以上介绍的几种转换方式可以看出，只要知道逻辑函数的任何一种表示方法，都可将其转换成其余的四种表示方法，其中真值表能够最直观地表示出逻辑函数的功能。

1.4　逻辑函数化简

在进行数字电路设计时，同一个逻辑式可以写成不同的形式。例如，有两个逻辑式

$$Y = \bar{A}BC + A\bar{B}C + ABC \quad 和 \quad Y = AC + BC$$

将它们的真值表分别列出后，能够发现它们的真值表相同，说明两个式子表示的是同一个逻辑函数。但前者需要 6 个门电路实现，而后者只需要 3 个门电路。由此可见，逻辑函数化简的目的就是要消去多余的乘积项（减少门的数量）和每个乘积项中多余的因子（减少与门输入端数量），达到用最少的电子器件实现电路设计的优化。

常用的化简方法有以下三种。

1.4.1　公式法化简

(1)基本公式

化简用的逻辑代数的基本公式有 18 个，见表 1－7。

表 1－7　逻辑代数的基本公式

序号	公　式	序号	公　式
1	$0 \cdot A = 0$	10	$\bar{1} = 0; \bar{0} = 1$
2	$1 \cdot A = A$	11	$1 + A = 1$
3	$A \cdot A = A$	12	$0 + A = A$

表 1－7(续)

序号	公式	序号	公式
4	$A \cdot \bar{A} = 0$	13	$A + A = A$
5	$A \cdot B = B \cdot A$	14	$A + \bar{A} = 1$
6	$A \cdot (B \cdot C) = (A \cdot B) \cdot C$	15	$A + B = B + A$
7	$A \cdot (B + C) = A \cdot B + A \cdot C$	16	$A + (B + C) = (A + B) + C$
8	$\overline{A \cdot B} = \bar{A} + \bar{B}$	17	$A + B \cdot C = (A + B) \cdot (A + C)$
9	$\bar{\bar{A}} = A$	18	$\overline{A + B} = \bar{A} \cdot \bar{B}$

(2)常用公式

化简用的逻辑代数的常用公式有 6 个，见表 1－8。

表 1－8　逻辑代数的常用公式

序号	公　式
19	$A + A \cdot B = A$
20	$A + \bar{A} \cdot B = A + B$
21	$A \cdot B + A \cdot \bar{B} = A$
22	$A \cdot (A + B) = A$
23	$A \cdot B + \bar{A} \cdot C + B \cdot C = A \cdot B + \bar{A} \cdot C$ $A \cdot B + \bar{A} \cdot C + BCD = A \cdot B + \bar{A} \cdot C$
24	$A \cdot \overline{A \cdot B} = A \cdot \bar{B}$;　$\bar{A} \cdot \overline{A \cdot B} = \bar{A}$

例 1－3　利用逻辑代数的基本公式和常用公式化简下列逻辑式

$$Y_1 = ABC + A(\bar{B} + \bar{C})$$

$$Y_2 = AC + A\bar{B} + \bar{B}\bar{C}$$

解　利用基本公式 8，7 和 14，得

$$Y_1 = ABC + A\overline{BC} = A(BC + \overline{BC}) = A$$

因为 Y_2 式中有 C 和 $\bar{C}$，利用常用公式 23，得

$$Y_2 = AC + \bar{B}\bar{C}$$

当已知的逻辑式比较简单时利用公式法对其进行化简，比较方便。但是，当遇到比较复杂的逻辑式时，公式法使用起来就比较麻烦了。

1.4.2　卡诺图化简

在介绍卡诺图化简之前，先介绍一下最小项的概念。

1.4.2.1　最小项

在 n 变量逻辑函数中，若 m 为包含 n 个变量的乘积项，而这 n 变量均以原变量或反变量的形式在 m 中出现一次，则称 m 为最小项。n 变量逻辑函数中最小项的数量共有 2^n 个。

例如：三个变量 A,B,C 可组成 $8(2^3)$ 个最小项，见表 1-9。

为了使用方便，通常用最小项编号来表示某个最小项。使某一最小项为 1 的变量取值组合所对应的十进制数，即为该最小项的编号。例如：当 $\bar{A}BC=1$ 时，即 $ABC=011$，表示的十进制数就是 3，因此将 $\bar{A}BC$ 这个最小项记作 m_3。三变量最小项的编号见表 1-9。

表 1-9　三变量最小项编号表

最小项	使最小项为 1 的变量取值组合			对应的十进制数	编号
	A	B	C		
$\bar{A}\bar{B}\bar{C}$	0	0	0	0	m_0
$\bar{A}\bar{B}C$	0	0	1	1	m_1
$\bar{A}B\bar{C}$	0	1	0	2	m_2
$\bar{A}BC$	0	1	1	3	m_3
$A\bar{B}\bar{C}$	1	0	0	4	m_4
$A\bar{B}C$	1	0	1	5	m_5
$AB\bar{C}$	1	1	0	6	m_6
ABC	1	1	1	7	m_7

从最小项的定义出发可以证明它有如下性质。

①在输入变量的任何取值下只有一个最小项的值为 1。

②全体最小项之和为 1，即 $\sum m_i = 1$。

③任意两个最小项的乘积为 0。

④具有相邻性的两个最小项可以合并成一项并消去一对因子。

若两个最小项只有一个因子不同，则称这两个最小项具有相邻性。例如，$A\bar{B}C$ 和 ABC 两个最小项中仅有一个因子不同，因此它们具有相邻性。这两个最小项或运算时可以合并成一项，消去一对因子 B 和 $\bar{B}$，即

$$A\bar{B}C + ABC = AC(B+\bar{B}) = AC$$

逻辑函数式的标准形式之一就是最小项之和的形式。例如，给定的逻辑式为

$$Y = AC + \bar{A}BC$$

则化成最小项之和形式为

$$Y = AC(\bar{B}+B) + \bar{A}BC = A\bar{B}C + ABC + \bar{A}BC = m_3 + m_5 + m_7 = \sum m_i(i=3,5,7)$$

1.4.2.2　卡诺图

(1)卡诺图定义

将 n 变量的全部最小项用一个小方块表示，并使具有相邻性的最小项在几何位置上也相邻地排列起来，所得到的图形叫做 n 变量最小项的卡诺图。这种表示逻辑函数的方法是由美国工程师卡诺(Karnaugh)首先提出的，因此把这种图形叫做卡诺图。

图 1-6 画出三变量和四变量最小项的卡诺图。图 1-6(a)中的两个小方块，如 m_0 和 m_1，m_1 和 m_5 不仅在几何位置上相邻，同时还具有相邻性。图 1-6(b)画出具有相邻性的两个最小项在几何位置上也相邻的几种情况(可将表中第一行和第四行、第一列和第四列看成