新理念

21世纪高等学校新理念教材建设工程

数字电子技术基础学习指导

辽宁工业大学电子信息工程教研室　编

东北大学出版社
Northeastern University Press

新理念

21世纪高等学校新理念教材建设工程

数字电子技术基础学习指导

辽宁工业大学电子信息工程教研室　编

东北大学出版社

·沈　阳·

图书在版编目（CIP）数据

数字电子技术基础学习指导 / 辽宁工业大学电子信息工程教研室编. —沈阳：东北大学出版社，2007.7

（21 世纪高等学校新理念教材建设工程）

ISBN 978-7-81102-428-9

Ⅰ. 数…　Ⅱ. 辽…　Ⅲ. 数字电路—电子技术—高等学校—教学参考资料　Ⅳ. TN79

中国版本图书馆 CIP 数据核字（2007）第 115191 号

出 版 者：东北大学出版社

地址：沈阳市和平区文化路 3 号巷 11 号

邮编：110004

电话：024—83687331（市场部）　83680267（社务室）

传真：024—83680180（市场部）　83680265（社务室）

E-mail：neuph @ neupress.com

http：//www.neupress.com

印 刷 者：沈阳市第六印刷厂书画彩印中心

发 行 者：东北大学出版社

幅面尺寸：184mm × 260mm

印　　张：13.875

字　　数：364 千字

出版时间：2007 年 7 月第 1 版

印刷时间：2007 年 7 月第 1 次印刷

责任编辑：牛连功　　　责任校对：舞　燕

封面设计：唐敏智　　　责任出版：杨华宁

ISBN 978-7-81102-428-9　　　定　　价：15.00 元

《数字电子技术基础学习指导》编委会

主　编　鲁宝春　王景利

　　　　　刘　毅　关维国

副主编　金晓龙　孟丽囡

　　　　　王亚君　李光林

　　　　　吕　娓　李宝国

前　言

本书由辽宁工业大学出版基金资助出版。

电子技术基础是电类各专业的一门重要技术基础课．伴随着电子技术日新月异的发展，此课程涉及面越来越广、实践性越来越强．对步入电学领域的初学者，常感到此课程内容抽象复杂，尤其对基本概念、基本电路的工作原理和基本分析方法感到较难理解，甚至无从下手，缺少系统的概念和分析、解决问题的能力．为解决上述问题，使学生更好地学习电子技术基础课，我们根据多年的教学实践经验，编写了电子技术基础（数字部分）课程的辅导教材，以满足广大读者学习和复习的需要．

本书共9章，各章内容包括基本要求、内容摘要、难点释疑、典型例题分析、自我检查题及参考答案、习题及详细解答六部分．基本要求部分意在使读者明确本章的教学重点、学习和考试的要求．内容摘要部分是对本章内容的基本概念、基本原理和基本分析方法的高度总结和归纳．典型例题分析部分是有针对性地精选出具有代表性的例题进行解析，以便对课程内容的重点、难点有深刻的理解，并达到举一反三的目的．最后按照教学基本要求，给出了自我检查题和习题，同时附参考答案，希望通过自测，检查学习效果．

本书的编写分工为：第一章，金晓龙；第二章，吕娓；第三章，李光林；第四章，王亚君；第五章，第八章，王景利；第六章，关维国；第七章，孟丽囡；第九章，刘毅．

由于时间仓促，书中可能存在错误、疏漏之处，我们诚恳地欢迎广大读者批评指正，并将意见反馈给我们，在此谨向热情的读者致以诚挚的谢意．

编　者

2007年3月

目　录

□ 第一章　逻辑代数基础

基本要求

数字电子技术这门课程既有丰富的理论性，又有很强的实践性，同时还具有相对完整性．学习的目的是解决实际应用中的逻辑问题；采用的思想是研究逻辑问题的规律，用逻辑函数来描述、设计合理的电路．

1．了解这门课程的用途、特点，掌握基本概念，能够完成数制转换、补码计算等．

2．掌握基本逻辑运算，包括功能、逻辑符号．

3．掌握逻辑代数的公式、定理．

4．掌握用逻辑函数描述逻辑问题的方法，最小项的含义和性质，逻辑函数的标准形式．

5．掌握公式法、卡诺图法化简，具有无关项函数的化简．

内容摘要

一、概　述

(一) 数字量和模拟量

数字量：变化在时间上和数量上都是不连续的．(存在一个最小数量单位 Δ)

模拟量：数字量以外的物理量．

模拟电路：用连续的模拟电压或电流值来表示信息．

数字电路：用一个分散的电压序列来表示信息．

(二) 数制和码制

1．数制：表示数量的规则．

(1) 每一位的构成方法．

(2) 从低位向高位的进位规则．

$$D = \sum k_i n^i \quad k \in (0,\ 1,\ \cdots,\ n-1)$$

若整数部分为 n 位，小数部分为 m 位，则

$$i = n-1,\ n-2,\ \cdots,\ 0,\ -1,\ -2,\ \cdots,\ -m$$

2．码制：表示事物的规则．

目前，数字电路中都采用二进制和基于二进制的八进制、十六进制和二-十进制．

表示数量时称二进制．

表示事物时称二值逻辑．

(三) 算术运算和逻辑运算

数字电路能够完成智能处理，这些是通过算术运算和逻辑运算实现的．当二进制代码表示数量时要进行算术运算，而如果表示逻辑状态时要进行逻辑运算．

引入补码，可以通过一个加法器实现加法和减法运算，而乘法可以用加法和移位来实现，除法可以用减法和移位来实现，这样，二进制的加、减、乘、除运算都可以通过加法器来实现．

原码：用二进制数的最高位表示符号位，正数为0，负数为1，其余表示数值部分(也叫数值位)．

正数的补码与原码相同．负数的补码，符号位不变，数值位逐位求反再加1．

注意 采用补码运算时，得到的结果是补码，如果是正数可以直接得出，如果是负数还要求一次补．

二、逻辑代数中的三种基本运算

(一) 基本运算

与：条件同时具备，结果发生．表达式为

$$Y = A \text{ AND } B = A \& B = A \cdot B = AB.$$

或：条件之一具备，结果发生．表达式为

$$Y = A \text{ OR } B = A + B.$$

非：条件不具备，结果发生．表达式为

$$Y = \overline{A}.$$

三种基本运算分别如图 1.1 所示．

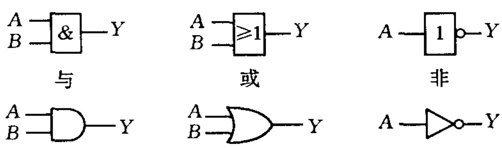


图 1.1

(二) 几种常用的复合逻辑运算

如图 1.2 所示．

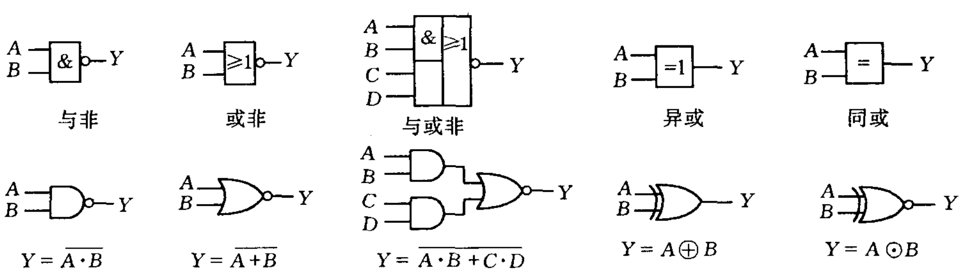


图 1.2

三、逻辑代数的基本公式和常用公式

(一) 基本公式

表 1.1　基本公式

序号	公式	序号	公式
1	$0A=0$	10	$\overline{1}=0$；$\overline{0}=1$
2	$1A=A$	11	$1+A=1$
3	$AA=A$	12	$0+A=A$
4	$A\overline{A}=0$	13	$A+A=A$
5	$AB=BA$	14	$A+\overline{A}=1$
6	$A(BC)=(AB)C$	15	$A+B=B+A$
7	$A(B+C)=AB+AC$	16	$A+(B+C)=(A+B)+C$
8	$\overline{AB}=\overline{A}+\overline{B}$	17	$A+BC=(A+B)(A+C)$
9	$\overline{\overline{A}}=A$	18	$\overline{A+B}=\overline{A}\,\overline{B}$

(二) 常用公式(见表 1.2)

表 1.2　常用公式

序号	公式
21	$A+AB=A$
22	$A+\overline{A}B=A+B$
23	$AB+A\overline{B}=A$
24	$A(A+B)=A$
25	$AB+\overline{A}C+BC=AB+\overline{A}C$ $AB+\overline{A}C+BCD=AB+\overline{A}C$
26	$A\,\overline{AB}=A\overline{B}$；$\overline{A}\,\overline{AB}=\overline{A}$

四、逻辑代数的基本定理

(一) 代入定理

任何一个含有某变量的等式，如果等式中所有出现此变量的位置均代之以一个逻辑函数式，则此等式依然成立.

运用代入规则可以扩大基本公式的应用范围.

(二) 反演定理

对于任意一个逻辑函数式 F，作如下处理：

1. 若把式中的运算符“·”换成“+”，“+”换成“·”；
2. 常量“0”换成“1”，“1”换成“0”；
3. 原变量换成反变量，反变量换成原变量.

那么得到的新函数式称为原函数式 F 的反函数式 $\overline{F}$.

运用反演规则时应注意两点：

1. 必须保持原函数的运算次序，适当地加入括号. 先与后或.
2. 不属于单个变量上的非号有两种处理方法：一种是该非号保留，而非号下面的函数式按反演规则变换；另一种是引入代入规则，将非号去掉，而非号下的函数式保留不变.

(三) 对偶定理

对于任意一个逻辑函数 F，作如下处理：

1. 若把式中的运算符“·”换成“+”，“+”换成“·”；
2. 常量“0”换成“1”，“1”换成“0”.

所得的新函数式为原函数式 F 的对偶式 F'，也称对偶函数.

求对偶函数时同样需注意保持原式中的运算顺序不变，且它只变换运算符和常量，其变量是不变的，因此一般情况下 $\overline{F}$ 不等于 F'.

对偶规则：如果两个函数式相等，则它们对应的对偶式也相等. 即若 $F_1 = F_2$，则 $F_1' = F_2'$. 使公式的数目增加一倍.

五、逻辑函数及其表示方法

(一) 逻辑函数

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

若以逻辑变量为输入，运算结果为输出，则输入变量值确定以后，输出的取值也随之而定. 输入与输出之间是一种函数关系.

(二) 表示方法

逻辑式：将输入与输出之间的逻辑关系用与、或、非的运算式表示，就得到逻辑式.

逻辑图：用逻辑图形符号表示逻辑运算关系，与逻辑电路的实现相对应.

波形图：将输入变量所有可能的取值和与之对应的输出，按时间顺序排列起来画成时间波形.

卡诺图：将逻辑函数的最小项之和以图形的方式表示出来.

六、逻辑函数的公式化简法

(一) 逻辑函数的最简形式

最简与或：包含的乘积项已经最少，每个乘积项的因子也最少，称为最简的与或逻辑式.

(二) 公式化简法

公式化简法即反复应用基本公式和常用公式，消去多余的乘积项和多余的因子.

七、逻辑函数的卡诺图化简

(一) 逻辑函数的卡诺图表示法

其实质是将逻辑函数的最小项之和以图形的方式表示出来.

以 2^n 个小方块分别代表 n 变量的所有最小项，并将它们排列成矩阵，而且使几何位置相邻的两个最小项在逻辑上也是相邻的(只有一个变量不同)，就得到表示 n 变量全部最小项的卡诺图.

用卡诺图表示逻辑函数：

1. 将函数表示为最小项之和的形式 $\sum m_i$；
2. 在卡诺图上与这些最小项对应的位置上填 1，其余地方填 0.

(二) 用卡诺图化简函数

依据：具有相邻性的最小项可合并，消去不同因子.

在卡诺图中，最小项的相邻性可以从图形中直观地反映出来.

合并最小项的原则：

1. 两个相邻最小项可合并为一项，消去一对因子；

2. 四个排成矩形的相邻最小项可合并为一项，消去两对因子；
3. 八个相邻最小项可合并为一项，消去三对因子.
两个相邻最小项合并为一项，消去一对因子的化简步骤为：
1. 用卡诺图表示逻辑函数；
2. 找出可合并的最小项；
3. 化简后的乘积项相加.
（项数最少，每项因子最少）
卡诺图化简的原则：
1. 化简后的乘积项应包含函数式的所有最小项，即覆盖图中所有的 1；
2. 乘积项的数目最少，即圈成的矩形最少；
3. 每个乘积项因子最少，即圈成的矩形最大.

难点释疑

一、代数法化简和卡诺图法化简的联系

卡诺图化简其实质是合并最小项，只不过将最小项按一定规律进行排序，并在 2^n 个最小项中提取公共变量后消去其他变量，以达到简化的目的.

代数法化简逻辑函数式，是运用逻辑代数的定律、定理、规则对逻辑式进行变换，以消除多余的与项和变量. 代数法化简没有普遍适用规律，有时需要一定的经验和熟练的技巧.

卡诺图法化简的实质是合并最小项，以消除多余的变量. 因为卡诺图中的一个小方格就是一个最小项，但是小方格与小方格之间的关系必须是相邻的关系，即卡诺图中的上下、左右、前后小方格的最小项必须保持只有一个变量不同，其余的变量都相同，才能实现 2^n（$n=1, 2, 3, \cdots$，正整数）个相邻方格的最小项合并. 其合并结果是消除 n 个变量. 可见，合并的最小项范围越大，可以消除的变量就越多. 这样做非常直观、便捷.

如果在已知与或表达式的情况下，将该式转换成最小项之和式后，再像卡诺图那样找出 2^n 个相邻最小项进行合并，就很困难了. 可见，代数法化简和卡诺图法化简具有相同的内在关系，只是处理方法不同而已.

二、逻辑函数的不同表示方法之间是如何进行转换的

同一个逻辑问题，可以采用多种方法表示. 而这些描述同一个问题的逻辑表示之间都能实现方便的转换.

（一）由真值表求逻辑函数式和逻辑电路

把真值表中使逻辑函数值为 1 的输入变量组合，写成对应的与项. 若对应的变量取值为 1，则写成原变量；若对应的变量取值为 0，则写成反变量. 然后将这些与项全部“或”起来，就得到了逻辑函数式.

对应于逻辑函数式的反变量，采用非门逻辑符号；与项用与门逻辑符号，多个与项相“或”用或门逻辑符号；将它们按逻辑运算关系连接起来，就能得到实现逻辑要求的逻辑电路.

(二) 由逻辑函数式求真值表

只要把逻辑函数式中所有输入变量按“0” “1”取值，代入所有组合中(2^n——n 是函数的变量数)进行运算，求出相应的逻辑函数值(结果)填入真值表中的相应行即可.

(三) 卡诺图与逻辑函数表达式之间的转换

先将逻辑函数化为最小项之和的形式(即标准与或表达式)，接着画出与函数变量数相对应的卡诺图. 在卡诺图中，凡是与表达式对应的最小项的小方格内填入“1”，其他小方格内填入“0”. 这样便得到了逻辑函数式的卡诺图.

三、具有无关项的逻辑函数如何化简

在化简时，应首先正确认识逻辑变量组合与逻辑结果之间的关系，无关项在一个逻辑函数中的表示方法；知道无关项在一个逻辑函数的化简中，可以当作“1”和“0”处理. 对于逻辑函数中的无关项，可以用几种方法给出. 例如，某逻辑电路的输入信号 $DCBA$ 是 8421BCD 码，由 8421BCD 码的概念可知：变量组合(即最小项) $A\bar{B}C\bar{D}$，$A\bar{B}CD$，$AB\bar{C}\bar{D}$，$AB\bar{C}D$，$ABC\bar{D}$，$ABCD$ 是不会出现的，即不影响 8421BCD 编码结果，所以这些项就是无关项. 在逻辑函数化简时，正因为这些项无关，因此这些项的取值可以认为是“0”，也可以认为是“1”，这由简化程度来决定.

若将具有无关项的逻辑函数表示在卡诺图中，图中填“1”和“0”的小方格分别对应于函数式中的最小项和式中不出现的最小项. 卡诺图中无关项对应处填“×”以示区别.

“×”的小方格可以和“1”格一起包围，此时，在包围圈中的无关项当作“1”对待；“×”的小方格可以不被包围，这时的“×”小方格就当作“0”处理了.

在用表达式化简时，可以将无关项当作“1”写入表达式中，以便和其他项相结合，使表达式化得更加简单些. 如果该无关项对式子的简化无帮助，则当作“0”处理.

以下是用卡诺图化简和用表达式化简的两个例子. 如要求对下列逻辑函数化简：

$$F=f(A, B, C)=\bar{A}C+\bar{A}B+AB\bar{C}\ (\text{无关项为：}A\bar{B}\bar{C}, ABC).$$

用卡诺图化简：

结合“1”方格画包围圈，得 $F=(A, B, C)=B+\bar{A}C$. 这里把“111”格当作“1”处理，而把“100”格当作“0”了. 如图 1.3 所示。

A \ BC	00	01	11	10
0	0	1	1	1
1	×	0	×	1

图 1.3

用表达式化简：

$$F=f(A, B, C)=\bar{A}C+\bar{A}B+AB\bar{C}=\bar{A}C+B.$$

显然，表达式中将添加的 ABC 当作“1”，而将 $A\bar{B}\bar{C}$ 无关项作“0”处理，两者化简的结果完全相同.

四、逻辑函数用最小项之和表示或者用最大项之积表示

(一) 最小项 m

n 变量函数有 2^n 个最小项.

1. m 是乘积项；
2. 包含 n 个因子；
3. n 个变量均以原变量和反变量的形式在 m 中出现一次.

最小项的性质：

1. 在输入变量任一取值下，有且仅有一个最小项的值为 1；
2. 全体最小项之和为 1；
3. 任何两个最小项之积为 0；
4. 两个相邻的最小项之和可以合并，消去一对因子，只留下公共因子.

相邻：仅一个变量不同的最小项.

可将任何一个函数化为最小项之和 $\sum m_i$ 的形式.

(二) 最大项 M

n 变量函数有 2^n 个最大项.

1. M 是相加项；
2. 包含 n 个因子；
3. n 个变量均以原变量和反变量的形式在 M 中出现一次.

最大项的性质：

1. 在输入变量任一取值下，有且仅有一个最大项的值为 0；
2. 全体最大项之积为 0；
3. 任何两个最大项之和为 1.

$$Y = \sum m_i,\quad \overline{Y} = \sum_{k\neq i} m_k,\quad Y = \overline{\sum_{k\neq i} m_k}.$$

可将任何一个函数化为最大项之积 $Y = \prod_{i\neq k} \overline{m_k} = \prod_{i\neq k} M_k$ 的形式.

五、逻辑函数的逻辑相邻性确定

循环码的任何相邻两组代码之间只有一个变量不同，这正好与卡诺图小方格的要求一致.

循环码(即格雷码)具有这样的特征：任何相邻两组代码之间只存在一个不同变量，这一点正好与逻辑相邻性定义(只有一个变量不同的两个与项，逻辑上称其为相邻)一致，因此，在画多变量卡诺图时，按循环码规律就能得到正确的卡诺图.

六、逻辑函数已化到最简

为了便于实现逻辑电路，逻辑函数常用"与或"表达式表示. 因此，是否化到最简主要看与项数目和每个与项所包含的变量数是否最少.

一个与或表达式是否已达到最简，主要看两个方面：一是表达式中与项的数目是否最少了，即表达式中的与项是否不能再合并了；第二是在与项相同的条件下，检查每个与项所包含的变量数是否达到了最少. 因为减少与项可以节省与门个数，减少与项中的变量数可以减少与门的输入端个数.

典型例题分析

例 1.1 将下列二进制数转换为等值的十进制数、十六进制数和八进制数.

(1) $(11001011)_2$；　(2) $(101010.101)_2$；　(3) $(100001.001)_2$.

解　(1) $(11001011)_2=(203)_{10}=(CB)_{16}=(313)_8$；

(2) $(101010.101)_2=(42.625)_{10}=(2A.A)_{16}=(52.5)_8$；

(3) $(100001.001)_2=(33.125)_{10}=(21.2)_{16}=(41.1)_8$.

注意　当转换为十六进制数和八进制数时，小数部分不够分组时要补 0.

例 1.2　将下列十六进制数转换为等值的二进制数、八进制数和十进制数.

(1) $(6E)_{16}$；　(2) $(FD.C)_{16}$；　(3) $(9B.A)_{16}$.

解　(1) $(6E)_{16}=(1101110)_2=(156)_8=(110)_{10}$；

(2) $(FD.C)_{16}=(11111101.110)_2=(375.6)_8=(253.75)_{10}$；

(3) $(9B.A)_{16}=(10011011.101)_2=(233.5)_8=(155.625)_{10}$.

例 1.3　将下列十进制数转换为等值的二进制数.

(1) $(145)_{10}$；　(2) $(27.2)_{10}$；　(3) $(0.625)_{10}$.

解：(1) $(145)_{10}=(10010001)_2$；

(2) $(27.5)_{10}=(11011.1)_2$；

(3) $(0.625)_{10}=(0.101)_2$.

例 1.4　将下列十进制数转换为等值的 8421BCD 码.

(1) $(54)_{10}$；　(2) $(87.15)_{10}$；　(3) $(239.03)_{10}$.

解　(1) $(54)_{10}=(0101\ 0100)_{8421}$；

(2) $(87.15)_{10}=(1000\ 0111.0001\ 0101)_{8421}$；

(3) $(239.03)_{10}=(0010\ 0011\ 1001.0000\ 0011)_{8421}$.

注意　BCD 码是一种四位二进制代码，用来特定地表示十进制的十个数码. 要注意的是，当最高位或最低位出现 0 时，不允许省略，必须用四位二进制代码表示每一个十进制数码.

例 1.5　将 2009 个“1”异或起来得到什么结果?

解　偶数个 1 异或结果为 0，奇数个 1 异或结果为 1. 所以将 2009 个 1 异或，结果为 1.

例 1.6　写出下列二进制数的原码和补码.

(1) $(+1011)_2$；　(2) $(-1101)_2$；　(3) $(-00101)_2$.

解　(1) $(+1011)_2$ 的原码和补码都是 01011(最高位的 0 是符号位).

(2) $(-1101)_2$ 的原码是 11101(最高位的 1 是符号位)，补码是 10011.

(3) $(-00101)_2$ 的原码是 100101(最高位的 1 是符号位)，补码是 111011.

例 1.7　求下列逻辑函数 F 的反函数 $\overline{F}$.

(1) $F_1=A\overline{B}+\overline{A}B$；　(2) $F_2=ABC+\overline{\overline{A}+\overline{B}+\overline{C}}$.

解　(1) $\overline{F_1}=(\overline{A}+B)(A+\overline{B})=\overline{A}\overline{B}+AB$；

(2) $\overline{F_2}=(\overline{A}+\overline{B}+\overline{C})\cdot\overline{ABC}=\overline{ABC}\cdot\overline{ABC}=\overline{ABC}$.

注意　使用反演定理时，长非号不变.

例 1.8　求下列逻辑函数的对偶式.

(1) $F_1=(A+B)\cdot(C+D)$；　(2) $F_2=\overline{A+B+\overline{\overline{C}+\overline{DF}}}$.

解　(1) $F_1=AB+CD$；

(2) $F_2=\overline{A\cdot B\cdot\overline{\overline{C}\cdot\overline{D+F}}}=\overline{AB\cdot\overline{\overline{CD}\,\overline{F}}}=\overline{AB}+\overline{CDF}$.

注意　使用对偶定理时，长非号不变.

例 1.9　求逻辑函数 $F=AB+(\overline{A}+B)(C+D+E)$的反函数 $\overline{F}$.

解　$\overline{F}=(\overline{A}+\overline{B})(A\overline{B}+\overline{CD}\,\overline{E})$.

例 1.10　逻辑函数 $F=\overline{AB}+CD$，求反函数 $\overline{F}$ 和对偶式 F'.

解　$\overline{F}=(A+B)(\overline{C}+\overline{D})$，　$F'=(\overline{A}+\overline{B})(C+D)$.

例 1.11　用公式法化简下列逻辑函数.

(1) $F_1=A\overline{B}+\overline{A}B+A$；　(2) $F_2=A\overline{B}\overline{C}+ABC+A\overline{B}C+AB\overline{C}+\overline{A}B$.

解　(1) $F_1=A\overline{B}+\overline{A}B+A=A(\overline{B}+1)+\overline{A}B=A+B$；

(2) $F_2=A\overline{B}\overline{C}+ABC+A\overline{B}C+AB\overline{C}+\overline{A}B=A\overline{C}(\overline{B}+B)+AC(B+\overline{B})+\overline{A}B$
$=A+B$.

例 1.12　用公式法将函数 $Y=B\overline{C}+AB\overline{C}E+\overline{B}(\overline{\overline{A}\,\overline{D}+AD})+B(A\overline{D}+\overline{A}D)$化简为与或式.

解　$Y=B\overline{C}+AB\overline{C}E+\overline{B}(\overline{\overline{A}\overline{D}+AD})+B(A\overline{D}+\overline{A}D)=B\overline{C}+A\overline{D}+\overline{A}D$.

例 1.13　用公式化将函数 $F(A,B,C)=\overline{A}\overline{B}\overline{C}+\overline{A}B\overline{C}+ABC+\overline{AB}$化简为最简与或式.

解　$F(A,B,C)=\overline{A}\overline{C}+ABC+\overline{A}+\overline{B}=\overline{A}+BC+\overline{B}=\overline{A}+\overline{B}+C$.

例 1.14　用公式法将函数 $F(A,B,C)=(A\oplus B)C+ABC+\overline{A}\overline{B}C$ 化简为最简与或式.

解　$F(A,B,C)=(A\oplus B)C+ABC+\overline{A}\overline{B}C=(A\oplus B)C+(AB+\overline{A}\overline{B})C$
$=C[(A\oplus B)+(\overline{A\oplus B})]=C$.

例 1.15　用公式法将函数 $Y=A\overline{C}+ABC+AC\overline{D}+CD$ 化简为最简与或式.

解　$Y=A(\overline{C}+BC)+C(A\overline{D}+D)=A\overline{C}+AB+AC+CD$
$=A(\overline{C}+C+B)+CD=A+CD$.

例 1.16　用公式法将函数 $Y=A+\overline{B}+\overline{CD}+\overline{\overline{AD}\cdot\overline{B}}$化简为最简与或式.

解　$Y=A+\overline{B}+\overline{CD}+AD+B=1$.

例 1.17　用公式法将函数 $Y=\overline{A}\cdot\overline{C}+\overline{A}\cdot\overline{B}+\overline{A}\cdot\overline{C}\cdot\overline{D}+BC$ 化简为最简与或式.

解　$Y=\overline{A}\cdot\overline{C}+\overline{A}\cdot\overline{B}+BC=\overline{A}\cdot\overline{BC}+BC=\overline{A}+BC$.

例 1.18　用卡诺图化简下列带有约束项函数为最简与或形式.

(1) $F_1(A,B,C,D)=\sum(2,3,6,7,8,10,12,14)$；

(2) $F_2(A,B,C,D)=\sum(0,1,2,3,4,6,8,9,10,11,12,14)$.

解　(1) $F_1=A\overline{D}+\overline{A}C$（见图 1.4(a)）；(2) $F_2=\overline{B}+\overline{D}$（见图 1.4(b)）.

例 1.19　用卡诺图化简下列函数为最简与或形式.

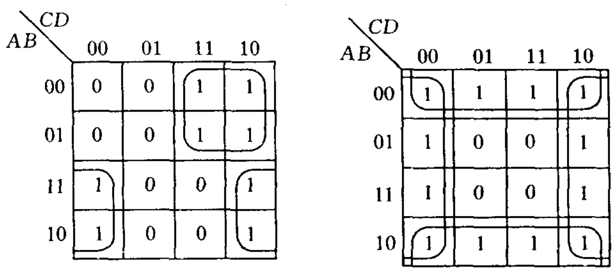
(1) $F_1=\overline{A}B\overline{C}\overline{D}+\overline{A}BC\overline{D}+AB\overline{C}D+ABCD+A\overline{B}C\overline{D}$，
　约束条件：$\overline{A}\overline{B}\overline{C}\overline{D}+\overline{A}\overline{B}C\overline{D}+\overline{A}\overline{B}\overline{C}D+\overline{A}BCD+A\overline{B}\overline{C}\overline{D}=0$；

(2) $F_2=\overline{A}\overline{B}\overline{C}\overline{D}+A\overline{B}\overline{C}\overline{D}+\overline{A}BCD+A\overline{B}CD$，
　约束条件：$\overline{A}\overline{B}CD+\overline{A}\overline{B}\overline{C}D+ABCD=0$.

解　(1) $F_1=\overline{B}\overline{D}+\overline{A}\overline{D}+BD$（见图 1.5(a)）；

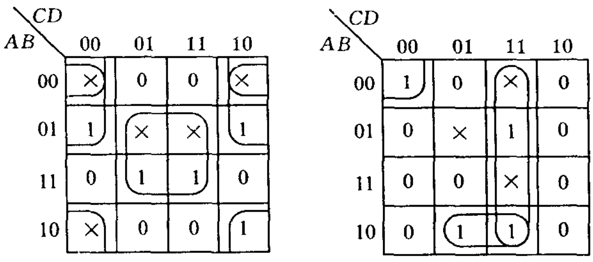
(2) $F_2=CD+A\overline{B}D+\overline{A}\overline{B}\overline{C}\overline{D}$（见图 1.5(b)）.

例 1.20　用卡诺图法将下列逻辑函数化简成最简与或式.



(a)　(b)

图 1.4



(a)　(b)

图 1.5

$F(A, B, C, D)=\bar{A}\cdot\bar{B}\cdot\bar{C}\cdot\bar{D}+\bar{A}\cdot\bar{B}C\bar{D}+\bar{A}B\bar{C}\cdot\bar{D}+A\bar{B}\cdot\bar{C}D$,

约束条件：$\bar{A}\bar{B}CD+\bar{A}BCD+ABCD+ABC\bar{D}+A\bar{B}CD+A\bar{B}C\bar{D}=0$.

解　$F=A\bar{B}+\bar{B}\bar{D}$（见图 1.6）.

例 1.21　用卡诺图法将下列逻辑函数化简成最简与或式.

$Y=C\bar{D}(A\oplus B)+\bar{A}B\bar{C}+\bar{A}\cdot\bar{C}\cdot D$，约束条件为 $AB+CD=0$.

解　$Y=B+\bar{A}D+AC$（见图 1.7）.

例 1.22　用卡诺图将下列函数化简为最简与或式.

$Y(A, B, C, D)=\sum m(0, 1, 2, 4, 5, 9)$，

约束条件：$m_7+m_8+m_{10}+m_{11}+m_{12}+m_{13}=0$.

解　$Y=\bar{C}+\bar{B}\cdot\bar{D}$（见图 1.8）.

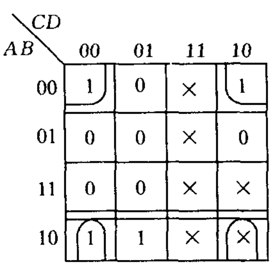


图 1.6

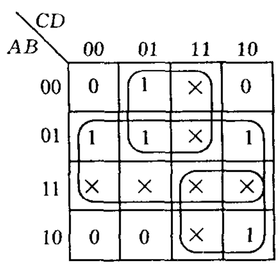


图 1.7

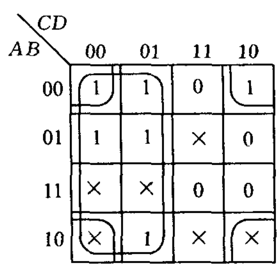


图 1.8