通向研究生之路系列丛书

数字电子技术

常见题型解析及模拟题

主编　王公望

编者　岳　怡　谢松云

西北工业大学出版社

(陕)新登字 009 号

【内容简介】 本书是根据国家教委制定的高等工业学校电子技术基础课程与脉冲数字电路课程的教学基本要求及硕士研究生入学考试的基本要求编写的辅导教材。

全书分为两部分。正文部分主要内容是：逻辑代数基础、门电路、触发器、组合逻辑电路、时序逻辑电路、大规模集成电路、数—模与模—数转换。各章均按重点与难点、例题精选及习题的格式编写。附录部分收录了国内几所重点高等院校近两年硕士研究生入学试题共 6 套，并提供了两套模拟试题，以供读者参考。另附有各章习题的参考答案，便于自学。

本书可供有志攻读硕士研究生的考生作为复习参考书，也可作为大学本科学生学习数字电路课程的辅助教材。

通向研究生之路系列丛书
数字电子技术
常见题型解析及模拟题
主　　编　王公望
责任编辑　傅高明
责任校对　钱伟峰
*
©1998　西北工业大学出版社出版发行
（邮编：710072　西安市友谊西路 127 号　电话：8493844）
全国各地新华书店经销
陕西省富平印刷有限责任公司印装
ISBN 7－5612－1011－6/TN·60
*
开本：787×1092 毫米 1/16　印张：14.875 字数：357 千字
1998 年 4 月第 1 版　　1998 年 9 月第 2 次印刷
印数：6001—9 000 册　　定价：19.00 元

前　言

本书是西北工业大学出版社根据国家教委颁布的高等工业学校基础课及技术基础课程教学基本要求和硕士研究生入学考试的基本要求组织编写的《通向研究生之路系列丛书》之一。目的在于为有志攻读硕士研究生的考生，提供一套全面的复习参考书，同时也为大学本科学生在学习有关技术基础课程时提供一套实用的辅助教材。

全书共分七章：逻辑代数基础、门电路、触发器、组合逻辑电路、时序逻辑电路、大规模集成电路、数-模与模-数转换。当今数字电路的核心正在由中小规模电路转向以设计专用集成电路为代表的数字系统设计。考虑到现今教学的实际，本书是按照从工程技术出发，以建立功能电路为主体，辅以模块化设计方法的原则进行编写的。重点在于介绍利用中大规模集成电路相互配合使用，构成任意功能的组合和时序逻辑电路的方法。每章由重点与难点、例题精选和习题三部分组成。重点与难点主要阐述各章的基本内容和要求，为读者提供一条主线。例题精选通过典型例题，按照题意分析—电路分析与综合—讨论的格式描述出一个完整的解题思路，帮助读者提高对数字电路的解题能力，并启迪读者作深一层次的思考。为使读者能检验对各章内容理解、应用的能力，每章均选编有习题，并在附录中给出了部分习题的参考答案。

本书附录中收录了国内几所重点高等院校近两年硕士研究生入学考试试题共6套，并提供了两套模拟试题。由于各院校考硕科目不同，在收录的考题中只选录了《数字电路》部分，以供读者参考。

本书由王公望教授、岳怡副教授和博士研究生谢松云三人合作编写。王公望教授任主编，负责全书的组稿与定稿。在编写过程中得到西北工业大学604教研室全体同志的帮助和支持，并由邵世兰、王南京、齐洁等同志为本书抄稿、画图。在此谨致以诚挚的谢意。对本书选用的参考文献的著作者，我们真诚地致以感谢。

编者

1997年7月

目　录

1

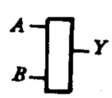
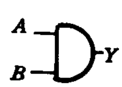
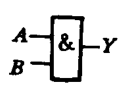
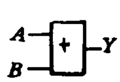
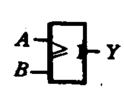
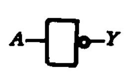
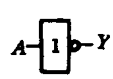
逻辑代数基础

1.1 重点与难点

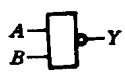
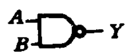
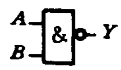
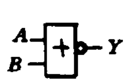
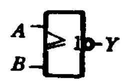
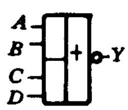
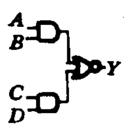
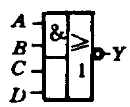
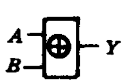
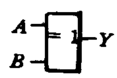
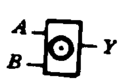
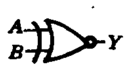
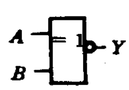
1.1.1 逻辑代数与基本逻辑函数

逻辑代数即是应用于二值逻辑电路中的布尔代数。其特点：一是它的所有变量与函数值仅有两个特征值 0 和 1，具有排中性，它们所表示的是一对互为相反的差异，它的公式、规则、定理与定义均需用二值逻辑的因果关系来理解；二是逻辑代数只有三种基本运算，即与、或、非，对应的即是逻辑与、逻辑或及逻辑非。利用这三种基本运算，则可得出处理实际逻辑问题的各种复合逻辑，如与非、或非、与或非、异或、同或等。实现这些逻辑运算的电路统称为门电路，其逻辑符号、逻辑函数式、输入输出真值表及基本运算规则如表 1.1 所示。

表 1.1 几种常用的逻辑运算

逻辑运算	逻辑函数式	逻辑符号 常用符号		逻辑符号 GB472B 12-85 符号	真值表		基本运算规则
与	$Y=A\cdot B$				$A\ B$ 0 0 0 1 1 0 1 1	$Y=A\cdot B$ 0 0 0 1	$A\cdot A=A$ $A\cdot\overline{A}=0$ $A\cdot 1=A$ $A\cdot 0=0$
或	$Y=A+B$				$A\ B$ 0 0 0 1 1 0 1 1	$Y=A+B$ 0 1 1 1	$A+A=A$ $A+\overline{A}=1$ $A+1=1$ $A+0=A$
非	$Y=\overline{A}$				A 0 1	$Y=\overline{A}$ 1 0	$\overline{\overline{A}}=A$

续　表

<table>
<tr><th rowspan="2">逻辑运算</th><th rowspan="2">逻　辑
函数式</th><th colspan="3">逻　辑　符　号</th><th rowspan="2">真　值　表</th><th rowspan="2">基本运算
规　　则</th></tr>
<tr><th colspan="2">常　用　符　号</th><th>GB472B
12-85 符号</th></tr>
<tr><td>与
非</td><td>$Y=\overline{AB}$</td><td>
</td><td>
</td><td>
</td><td><table><tr><th>A</th><th>B</th><th>$Y=\overline{AB}$</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table></td><td>$\overline{A\cdot 1}=\overline{A}$
$\overline{A\cdot 0}=1$
$\overline{A\cdot A}=\overline{A}$
$\overline{A\cdot B}=\overline{A}+\overline{B}$
$\overline{\overline{A}\cdot\overline{B}}=A+B$
$\overline{\overline{AB}}=AB$</td></tr>
<tr><td>或
非</td><td>$Y=\overline{A+B}$</td><td>
</td><td>
</td><td>
</td><td><table><tr><th>A</th><th>B</th><th>$Y=\overline{A+B}$</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table></td><td>$\overline{A+0}=\overline{A}$
$\overline{A+1}=0$
$\overline{A+A}=\overline{A}$
$\overline{A+B}=\overline{A}\cdot\overline{B}$
$\overline{A+B}=AB$
$\overline{\overline{A+B}}=A+B$</td></tr>
<tr><td>与
或
非</td><td>$Y=\overline{AB+CD}$</td><td>
</td><td>
</td><td>
</td><td><table><tr><th>$A\ B\ C\ D$</th><th>Y</th><th>$A\ B\ C\ D$</th><th>Y</th></tr><tr><td>0 0 0 0</td><td>1</td><td>1 0 0 0</td><td>1</td></tr><tr><td>0 0 0 1</td><td>1</td><td>1 0 0 1</td><td>1</td></tr><tr><td>0 0 1 0</td><td>1</td><td>1 0 1 0</td><td>1</td></tr><tr><td>0 0 1 1</td><td>0</td><td>1 0 1 1</td><td>0</td></tr><tr><td>0 1 0 0</td><td>1</td><td>1 1 0 0</td><td>0</td></tr><tr><td>0 1 0 1</td><td>1</td><td>1 1 0 1</td><td>0</td></tr><tr><td>0 1 1 0</td><td>1</td><td>1 1 1 0</td><td>0</td></tr><tr><td>0 1 1 1</td><td>0</td><td>1 1 1 1</td><td>0</td></tr></table></td><td></td></tr>
<tr><td>异
或</td><td>$Y=$
$A\oplus B=$
$A\overline{B}+\overline{A}B$</td><td>
</td><td>
</td><td>
</td><td><table><tr><th>A</th><th>B</th><th>$Y=A\oplus B$</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table></td><td>$A\oplus A=0$
$A\oplus\overline{A}=1$
$A\oplus 0=A$
$A\oplus 1=\overline{A}$
$A\oplus\overline{B}=\overline{A\oplus B}$
$A\oplus B\oplus C=A\odot B\odot C$
$A(B\oplus C)=AB\oplus AC$</td></tr>
<tr><td>同
或</td><td>$Y=$
$A\odot B=$
$AB+\overline{AB}$</td><td>
</td><td>
</td><td>
</td><td><table><tr><th>A</th><th>B</th><th>$Y=A\odot B$</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table></td><td></td></tr>
</table>

1.1.2 逻辑代数的基本公式与定理

逻辑代数的基本公式又称为布尔恒等式，分列于表 1.2 中。这些公式反映了二值逻辑的基本思想，是逻辑运算的重要工具，也是学习数字电子电路的必备基础。

表 1.2　逻辑代数定律

序　号	名　称	基　本　公　式	对　偶　式
1	交换律	$A+B=B+A$	$AB=BA$
2	结合律	$A+(B+C)=(A+B)+C$	$A(BC)=(AB)C$
3	分配律	$A+BC=(A+B)(A+C)$	$A(B+C)=AB+AC$
4	0-1 律	$1\cdot A=A$	$0+A=A$
		$0\cdot A=0$	$1+A=1$
5	互补律	$A+\overline{A}=1$	$A\cdot\overline{A}=0$
6	重叠律	$A+A=A$	$A\cdot A=A$
7	对合律	$\overline{\overline{A}}=A$	
8	吸收律	$A+AB=A$	$A(A+B)=A$
		$A+\overline{A}B=A+B$ $\overline{\overline{A}\cdot\overline{\overline{A}B}}=\overline{\overline{A}\cdot(A+\overline{B})}=\overline{\overline{A}\,\overline{B}}=A+B$	$A(\overline{A}+B)=AB$
9	德·摩根定理	$\overline{AB}=\overline{A}+\overline{B}$	$\overline{A+B}=\overline{A}\cdot\overline{B}$
10	包含律	$AB+\overline{A}C+BC=AB+\overline{A}C$	$(A+B)(\overline{A}+C)(B+C)=(A+B)(\overline{A}+C)$

逻辑代数的基本定理：

(1) **代入定理**　任何一个含有变量 A 的等式，如果将所有出现 A 的位置都代之以一个逻辑函数式，则等式仍成立。

(2) **对偶定理**　对于任何一个逻辑函数式 Y，若将其中的"·"换成"+"，"+"换成"·"，1 换成 0，0 换成 1，则得出一个新的函数式 Y'，把 Y' 称为原函数式 Y 的对偶函数式。

原函数式 Y 与对偶函数式 Y' 互为对偶函数；两个相等函数式的对偶函数式必相等。

(3) **反演定理**　对于任何一个逻辑函数式 Y，若将其中的"·"换成"+"，"+"换成"·"，1 换成 0，0 换成 1，并将原变量换成反变量，反变量换成原变量，则得出的新的逻辑函数式即为原函数式的反函数 $\overline{Y}$。

(4) **展开式定理**　对于任何逻辑函数都可以对它的某一个变量 x_i 展开成如下的与或式及或与式。

$$f(x_1,x_2,\cdots,x_i,\cdots,x_n)=x_i f(x_1,x_2,\cdots,1,\cdots,x_n)+\overline{x}_i f(x_1,x_2,\cdots,0,\cdots,x_n)$$

$$f(x_1,x_2,\cdots,x_i,\cdots,x_n)=[x_i+f(x_1,x_2,\cdots,0,\cdots,x_n)][\overline{x}_i+f(x_1,x_2,\cdots,1,\cdots,x_n)]$$

1.1.3 逻辑函数式及其化简

逻辑函数表达式反映了实际逻辑问题中输入变量与输出变量之间的因果关系。它可以通

过建立输入输出真值表得出，通常具有两种标准形式：一是最小项之积即标准与或式，二是最大项之积，即标准或与式。这两种标准式又称为范式，是唯一的。若经适当的处理，同一个逻辑函数式可以表示成繁简不同，逻辑运算不同的多种形式，以便采用不同的器件实现其功能。若要得出最简的函数式，则需进行逻辑函数化简，常用的方法一是利用布尔代数进行运算，二是利用卡诺图化简，这是本章的重点要求。此外还有适用于多变量逻辑函数化简的奎恩一麦克拉斯基法和增项消元法，这两种方法通常称为列表法，其化简步骤较为规范，适合于编程运算。

1. 最小项与最大项

(1) 最小项。若有 n 个逻辑变量，它们所组成的具有 n 个变量的与项中，每个变量均以原变量或反变量的形式出现一次，且仅出现一次，则这些与项称为 n 个变量的最小项，记为 m_i。显然 n 个变量必定有 2^n 个最小项，i 表示最小项的序号，在变量按序排列后，所组成的与项中变量以原变量形式出现记为1，以反变量形式出现记为0，这1和0的顺序组成的二进制数对应的十进制值即是 i 值。

例如，A,B,C 三个变量可组成8个最小项，分别记为：$m_0=\overline{A}\overline{B}\overline{C}$，$m_1=\overline{A}\overline{B}C$，$m_2=\overline{A}B\overline{C}$，$m_3=\overline{A}BC$，$m_4=A\overline{B}\overline{C}$，$m_5=A\overline{B}C$，$m_6=AB\overline{C}$，$m_7=ABC$

对于任何一个最小项，只有一组变量取值，且仅有一组变量取值使其逻辑值为1，这组变量取值即是这一最小项的序号 i 值；任意两个最小项之积必为0；对应于输入变量的全部最小项之和必为1；若两个最小项中仅有一个因子不相同，且分别是同一变量的原变量和反变量，则这两个最小项称为相邻最小项。两相邻最小项可以合并为一项，并可消去一个变量。

(2) 最大项。与最小项的定义相仿，若有 n 个逻辑变量，所组成的包含了 n 个变量的或项称为这 n 个变量的最大项，记为 M_i。i 表示最大项的序号，与最小项不同的是，在确定 i 值时原变量取0而反变量取1。

最大项与最小项之间的关系必须认真掌握，它们是：对于 n 个逻辑变量的最大项与最小项，若其序号 i 值相同，则二者互补；若其因子相同，则二者互为对偶。

2. 卡诺图

(1) 任何一个逻辑变量 A，以其原变量 A 和反变量 $\overline{A}$ 代表着同一事物的一对互为相反的状态，因此原变量与反变量就其描述的事物整体，从逻辑上讲各占一半，完全均等。若原变量用1表示，反变量用0表示，这一关系如图1.1所示。

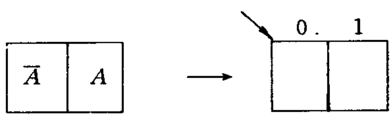


图 1.1　一变量卡诺图

(2) 对于两个逻辑变量 A,B，依其原变量和反变量共有4种不同的组合，代表了事物的4种可能的状态，这四种组合各占整体的1/4，完全均等，如图1.2所示。

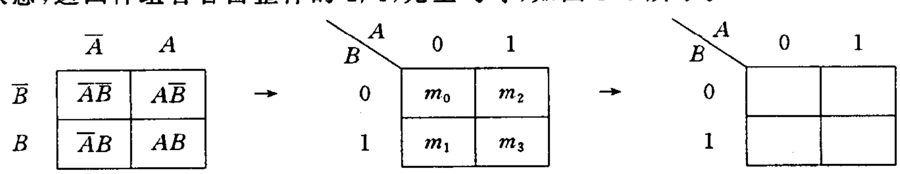
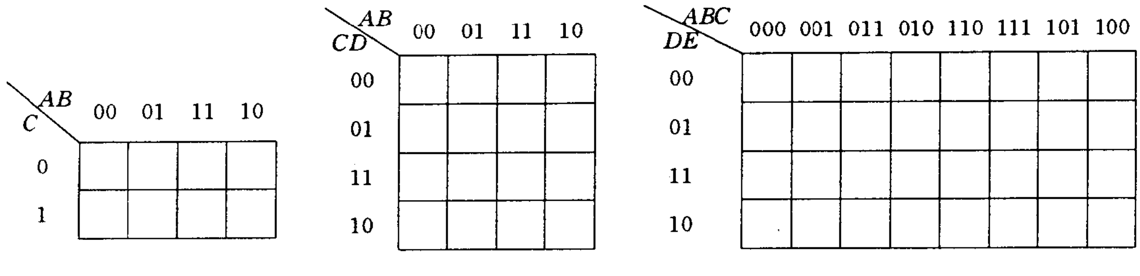


图 1.2　二变量卡诺图

这种反映逻辑变量组合关系的图表称为卡诺图。图 1.3(*a*)、(*b*)、(*c*) 分别给出了三、四、五变量卡诺图的常用形式。



(*a*) (*b*) (*c*)

图 1.3 三、四、五变量卡诺图

3. 卡诺图的特点

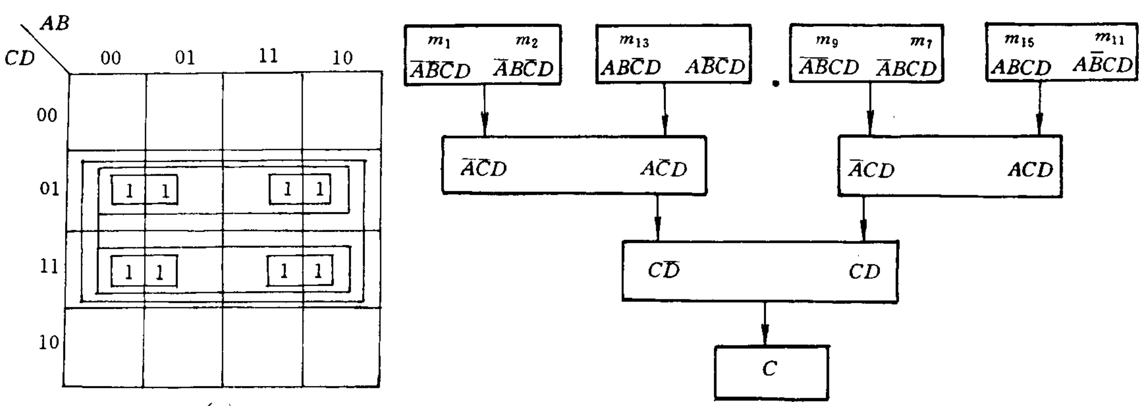
(1) 在卡诺图中每个变量均以原变量及反变量各占其区域的一半，因此 n 个变量的卡诺图包含有 2^n 个小方格，对应着 n 个变量的 2^n 个最小项。

(2) 任意相邻的两个小方格所对应的最小项即是两个相邻最小项，与每一小方格相邻的小方格数随变量数的增加而增加，即相邻小方格数等于变量数。

(3) 在卡诺图上，与逻辑具有区域的公共性，或逻辑具有区域的叠加性，非逻辑具有逻辑否定的含义。这是一个重要的概念，是构成逻辑函数卡诺图的一种基本方法。

(4) 逻辑函数的卡诺图化简方法。利用卡诺图化简逻辑函数的基本思路是合并相邻小方格以进行消元。

例如，某逻辑函数式具有图 1.4(*a*) 所示的卡诺图，则可按图(*b*) 所示的关系处理，消去 A，B，D 三个变量。



(*a*) (*b*)

图 1.4 利用卡诺图合并最小项

在卡诺图上，既可合并由 1 格构成的区域得出原函数式的最简与-或式，也可合并由 0 格构成的区域得出反函数式的最简与-或式，再经反演运算，即可得出原函数的最简或-与式。若是要求将原函数式化简成其它运算形式，例如，与非-与非式，或非-或非式，只需进行适当的逻辑变换就可实现。

4. 具有无关最小项的逻辑函数

(1) 任何一个 n 变量的逻辑函数，总能用 m 个最小项之和的形式来表示，这 m 个最小项即是使该函数式逻辑值为 1 的输入变量的取值。若剩下的 $2^n - m$ 个最小项使函数式的逻辑值为 0，这就表明此函数式与其 2^n 个最小项都有关。这一函数称为完全描述的逻辑函数。

(2) 在某些实际应用中，一个 n 变量的逻辑函数，并不是与它的 2^n 个最小项都有关，而是只与其中的一部分有关，与另一部分则无关系。这一部分无关的最小项并不决定函数的取值，故称为无关最小项，记为 d_i，i 表示序号，确定方法与最小项相同。这类逻辑函数称为包含有无关最小项的逻辑函数，或称为具有约束条件的逻辑函数，也称为不完全描述的逻辑函数。

(3) 无关最小项的出现，通常有两种情况：一是在某些实际问题中，加在逻辑电路上的输入变量的某些取值不可能或不允许出现，这种对于输入变量取值所加的限制称为约束，所对应的最小项称为约束项，它们构成了逻辑函数的约束条件。二是输入变量的某些取值的出现，不会影响逻辑函数的有效取值，即不影响逻辑功能的实现。通常把这些输入变量的取值所对应的最小项称为任意项或随意项。

逻辑函数的约束项及任意项统称为无关最小项，可将它们随意地加到逻辑函数表达式中。在利用卡诺 图化简逻辑函数时，可以把它们用其符号 d 填入图中，并可随意地视为 1 格或 0 格参与化简，而使函数式化简为最简的形式，并不会影响该逻辑函数的实际功能。

1.2 例题精选

$+C\overline{D}$

例 1.1 试利用代数法将函数 $Y = AC + \overline{B}C + B\overline{D} + A(B + \overline{C}) + \overline{A}BC\overline{D} + A\overline{B}DE$ 化简为最简的与-或式。

题意分析 利用代数法化简逻辑函数，实质上就是用逻辑代数（即布尔代数）的基本公式和规则，消去函数式中多余的项和每一项中的多余因子。常用的方法可归纳如下。

(1) 并项法：利用 $A + \overline{A} = 1$，将两项合并为一项，并消去一个变量。

(2) 吸收法：利用 $A + AB = A$，消去多余的项。

(3) 消去法：利用 $A + \overline{A}B = A + B$，消去多余的因子。

(4) 配项法：利用 $A = A(B + \overline{B}) = AB + A\overline{B}$，配项后用来消去多余的项。

在化简复杂的逻辑函数时，往往需要灵活、交替地运用上述方法，才能得到较好的结果。

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

$AC + \overline{B}C + B\overline{D} + \underline{C\overline{D}} + A(B + \overline{C}) + \underline{\overline{A}BC\overline{D}} + A\overline{B}DE =$

（根据 $A + AB = A$，消去 $\overline{A}BC\overline{D}$）

$AC + \underline{\overline{B}C} + B\overline{D} + C\overline{D} + \underline{A(\overline{\overline{B}C})} + A\overline{B}DE =$

（根据 $A + \overline{A}B = A + B$，消去$\overline{B}C$）

$\underline{AC} + \overline{B}C + B\overline{D} + C\overline{D} + \underline{A} + \underline{A\overline{B}DE} =$

（根据 $A + AB = A$，消去 AC 和 $A\overline{B}DE$）

$A + \underline{\overline{B}C} + \underline{B\overline{D}} + \underline{C\overline{D}} =$

（根据 $AB + \overline{A}C + BC = AB + \overline{A}C$，消去 $C\overline{D}$）

$A + \overline{B}C + B\overline{D}$

讨论 本例归结了几种用代数法化简逻辑函数的常用方法。但在具体解题时，并不要求

遵循固定的模式。根据个人对逻辑代数掌握的熟练程度，可几步同时进行，以求快速地得出结果。例如本例的另一种解法是：

$$\begin{aligned}Y = {}& AC + \overline{B}C + B\overline{D} + C\overline{D} + A(B + \overline{C}) + \overline{A}BC\overline{D}A\overline{B}DE = \\ & A(C + B + \overline{C} + \overline{B}DE) + \overline{B}C + B\overline{D} + C\overline{D}(1 + \overline{A}B) = \\ & A + \overline{B}C + B\overline{D}\end{aligned}$$

较之原解法要简捷得多。

例 1.2 试用代数法化简逻辑函数式

$$Y = AB + A\overline{C} + \overline{B}C + \overline{C}B + \overline{B}D + \overline{D}B + ADE(F + G)$$

题意分析 本例给出的逻辑函数式包含有 7 个逻辑变量。因此，化简时重点应放在如何尽可能消去多余的变量，以简化函数式的结构。例如，式中 $ADE(F + G)$ 这一项若能消去，就可删除 E,F,G 三个变量。

解 1

$$\begin{aligned}Y = {}& AB + A\overline{C} + \overline{B}C + \overline{C}B + \overline{B}D + \overline{D}B + ADE(F + G) = \\ & AB + \overline{B}D[1 + E(F + G)] + ADE(F + G) + A\overline{C} + \overline{B}C + \overline{C}B + \overline{D}B = \\ & AB + \overline{B}DE(F + G) + ADE(F + G) + A\overline{C} + \overline{B}C + \overline{C}B + \overline{B}D + \overline{D}B = \\ & AB + A\overline{C} + \overline{B}C + \overline{C}B + \overline{B}D + \overline{D}B = \\ & A\overline{\overline{B}C} + \overline{B}C + \overline{C}B + \overline{B}D + \overline{D}B = \\ & A + \overline{B}C + \overline{C}B + \overline{B}D + \overline{D}B + C\overline{D} = \\ & A + \overline{B}C + \overline{C}B + \overline{B}D + \overline{D}B + C\overline{D} = \\ & A + \overline{B}D + B\overline{C} + C\overline{D}\end{aligned}$$

解 2

$$\begin{aligned}Y = {}& AB + A\overline{C} + \overline{B}C + \overline{C}B + B\overline{D} + \overline{D}B + ADE(F + G) = \\ & (A\overline{\overline{B}C} + \overline{B}C) + \overline{C}B + B\overline{D} + \overline{D}B + ADE(F + G) = \\ & [A + ADE(F + G)] + \overline{B}C + \overline{C}B + B\overline{D} + \overline{D}B = \\ & A + \overline{B}C + \overline{C}B + \overline{B}D + \overline{D}B + C\overline{D} = \\ & A + \overline{B}D + B\overline{C} + C\overline{D}\end{aligned}$$

讨论 通过本例的两种解法可知，对于比较复杂的逻辑函数式，可用不同的公式或方法进行化简，其结果是相同的，但是化简过程有繁与简的差别。解题时，必须善于选择比较精练的方法来完成。

例 1.3 试用代数法将逻辑函数式

$$\begin{aligned}Y = {}& (A + \overline{B} + C + \overline{D})(A + \overline{B} + \overline{C} + \overline{D})(\overline{A} + B + C + D) \\ & (\overline{A} + B + \overline{C} + D)(\overline{A} + \overline{B} + C + D)(\overline{A} + \overline{B} + \overline{C} + D)\end{aligned}$$

化简为最简或-与式。

题意分析 本例给出的逻辑函数式，是一个以最大项之积的形式表示的标准或-与式。化简时，可先用对偶规则将其对偶函数式写出，变成为以最小项之和的形式表示的标准与-或式，进行化简。然后，再对偶回原函数式，即可得出最简的或-与式。

解 由原函数式得其对偶函数式为：

$$\begin{aligned}Y' = {}& A\overline{B}C\overline{D} + A\overline{B}\overline{C}\overline{D} + \overline{A}BCD + \overline{A}B\overline{C}D + \overline{A}\overline{B}CD + \overline{A}\overline{B}\overline{C}D = \\ & A\overline{B}\overline{D}(C + \overline{C}) + \overline{A}D(BC + B\overline{C} + \overline{B}C + \overline{B}\overline{C}) = \\ & A\overline{B}\overline{D} + \overline{A}D\end{aligned}$$

化简后的原函数为：

$$Y = (Y')' = (A + \overline{B} + \overline{D})(\overline{A} + D)$$

讨论 逻辑函数中的对偶规则有一个重要的推论，即函数与其对偶函数互为对偶函数。因此，在处理用或-与式表示的逻辑函数时，经常用对偶的方法变成与-或式进行处理，然后再对偶得出原函数式。

例 1.4 试用代数法证明：

$$(A + B)(\overline{A} + C)(B + C) = (A + B)(\overline{A} + C)$$

题意分析 证明逻辑等式成立是逻辑代数中的一类题型，本例便是一个典型的例证。与逻辑函数化简一样，证明时可用不同的公式和方法进行。关键仍然是要善于选择比较精练的方法。

解 1

$$\begin{aligned}\text{左式} = &(A + B)(\overline{A} + C)(B + C) = \\ &(A + B)(\overline{A}B + C) = \\ &\overline{A}B + AC + BC = \\ &(A + B)C + (A + B)\overline{A} = (A + B)(\overline{A} + C) = \text{右式}\end{aligned}$$

解 2 令 $F = (A + B)(\overline{A} + C)(B + C)$

$G = (A + B)(\overline{A} + C)$

求 F,G 两函数的对偶函数：

$$\begin{aligned}&F' = AB + \overline{A}C + BC = AB + \overline{A}C \\ &G' = AB + \overline{A}C \\ &F' = G' \qquad \text{则有 } F = G\end{aligned}$$

讨论 本例采用两种方法证明给定的逻辑等式。由于原式是用或-与式形式表示，用对偶规则证明较为简便。对偶规则有一个重要的推论：即两逻辑函数相等，则其对偶函数式必然相等。因此可将等式两边同时对偶成与-或式，然后证明等式成立。

例 1.5 设逻辑函数 $Y = A \oplus B \oplus C$，证明其对偶函数 $Y' = A \odot B \odot C$，且 $Y = Y'$ 是一个自对偶函数。

题意分析 异或逻辑函数从逻辑上定义为两变量的排斥或，同或逻辑函数定义为两变量的异或非。因此，两变量的异或逻辑函数与同或逻辑函数互为反函数。而且可以证明，二者又互为对偶函数。多变量的异或与同或逻辑函数之间的关系较为复杂，通常两个逻辑函数有如下关系成立：

$$A \oplus B \oplus C \oplus \cdots = \begin{cases} 1\ (\text{当取值为 1 的变量个数是奇数时}) \\ 0\ (\text{当取值为 1 的变量个数是偶数时}) \end{cases}$$

$$A \odot B \odot C \odot \cdots = \begin{cases} 0\ (\text{当取值为 0 的变量个数是奇数时}) \\ 1\ (\text{当取值为 0 的变量个数是偶数时}) \end{cases}$$

本例即是推证三变量的异或与同或逻辑函数之间的这种关系。

解 1 因为：

$$\begin{aligned}Y = &A \oplus B \oplus C = \\ &A\,\overline{B \oplus C} + \overline{A}(B \oplus C) = \\ &A\,\overline{B\overline{C} + \overline{B}C} + \overline{A}(B\overline{C} + \overline{B}C)\end{aligned}$$

所以：

$$\begin{aligned}Y' = &[A + \overline{(B + \overline{C})(\overline{B} + C)}][\overline{A} + (B + \overline{C})(\overline{B} + C)] = \\ &(A + \overline{BC + \overline{B}\,\overline{C}})(\overline{A} + BC + \overline{B}\overline{C}) = \\ &(A + \overline{B \odot C})(\overline{A} + B \odot C) =\end{aligned}$$

$$A(B\odot C)+\overline{A}\,\overline{(B\odot C)}=A\odot B\odot C$$

解 2 因为：$Y=A\oplus B\oplus C=$

$$A\oplus(B\overline{C}+\overline{B}C)$$

所以：$Y'=A\odot[(B+\overline{C})(\overline{B}+C)]=A\odot(BC+\overline{B}\overline{C})=A\odot B\odot C$

解 3 因为：$Y=A\oplus B\oplus C=A(B\odot C)+\overline{A}(B\oplus C)$

所以：$Y'=[A+(B\oplus C)][\overline{A}+(B\odot C)]=A(B\odot C)+\overline{A}\,\overline{(B\odot C)}=A\odot B\odot C$

而且：$Y=A\oplus B\oplus C=A\overline{B\overline{C}+\overline{B}C}+\overline{A}(B\overline{C}+\overline{B}C)=$

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

$$(AB+\overline{A}\overline{B})C+\overline{(AB+\overline{A}\,\overline{B})}\overline{C}=A\odot B\odot C$$

因此 $Y=Y'$，即 $Y=A\oplus B\oplus C$ 是一个自对偶函数。

讨论 在逻辑函数的对偶规则中，有一个重要的推论，即一个函数的对偶函数与原函数相等，则称这一函数为自对偶函数。本例给出的函数式 $Y=A\oplus B\oplus C$，即是一个自对偶函数。在推证 $Y'=A\odot B\odot C$ 时，本例采用了三种方法，均是利用异或及同或逻辑的性质及其运算特点进行的。但要注意，本例容易出错的地方是，将原函数写成 $Y=A\oplus B\oplus C=A\oplus(B\oplus C)$；在求对偶式时，把三变量异或关系简单地视为两变量的关系，从而得出 $Y'=A\odot(B\oplus C)$ 的错误结果。

例 1.6 试用公式法证明下列关系成立：

(1) $X_1X_2\oplus\overline{X}_1X_3=X_1X_2+\overline{X}_1X_3$

(2) 若 $X_1+X_2=1$，则有 $X_1\oplus X_2=\overline{X_1X_2}$

(3) 若 $X_1X_2=0$，则有 $X_1\oplus X_2=X_1+X_2$

(4) $(X_1\oplus X_2)\odot X_1X_2=\overline{X}_1\overline{X}_2$

题意分析 逻辑函数均可用与／异或运算来表示。因此，异或运算在逻辑运算中占有特殊的地位。本例给出的一组关系，反映了异或运算和与运算及或运算的内在联系，以及相互转换的条件。

证明

(1) $X_1X_2\oplus\overline{X}_1X_3=X_1X_2+\overline{X}_1X_3$

$$\begin{aligned}X_1X_2\oplus\overline{X}_1X_3&=\overline{X_1X_2}\,\overline{X}_1X_3+X_1X_2\,\overline{\overline{X}_1X_3}=\\&(\overline{X}_1+\overline{X}_2)\overline{X}_1X_3+X_1X_2(X_1+\overline{X}_3)=\\&\overline{X}_1X_3+\overline{X}_1\overline{X}_2X_3+X_1X_2+X_1X_2\overline{X}_3=\\&X_1X_2+\overline{X}_1X_3\end{aligned}$$

(2) 如果 $X_1+X_2=1$，则有 $X_1\oplus X_2=\overline{X_1X_2}$

$$\begin{aligned}\overline{X_1X_2}&=\overline{X}_1+\overline{X}_2=\overline{X}_1(X_1+X_2)+\overline{X}_2(X_1+X_2)=\\&\overline{X}_1X_2+X_1\overline{X}_2=X_1\oplus X_2\end{aligned}$$

(3) 如果 $X_1X_2=0$，则有 $X_1\oplus X_2=X_1+X_2$

$$\begin{aligned}X_1\oplus X_2&=\overline{X}_1X_2+X_1\overline{X}_2=\overline{X}_1X_2+X_1\overline{X}_2+X_1X_2=\\&X_1(X_2+\overline{X}_2)+X_2(X_1+\overline{X}_1)=X_1+X_2\end{aligned}$$

(4) $(X_1\oplus X_2)\odot X_1X_2=\overline{X}_1\overline{X}_2$

$$(X_1\oplus X_2)\odot X_1X_2=(\overline{X}_1X_2+X_1\overline{X}_2)X_1X_2+(X_1X_2+\overline{X}_1\overline{X}_2)\,\overline{X_1X_2}=$$

$$\overline{X}_1\overline{X}_2\,\overline{X_1X_2} = \overline{X}_1\overline{X}_2(\overline{X}_1 + \overline{X}_2) = \overline{X}_1\overline{X}_2$$

讨 论　本例推证的一组异或运算关系，具有重要的应用价值，应很好地理解和掌握。例如，公式 $X_1X_2 \oplus \overline{X}_1X_3 = X_1X_2 + \overline{X}_1X_3$，说明两个乘积项中只要有一对变量是相反的变量，则它们的异或值和或值相同。因此，可以用简单的或运算代替比较繁杂的异或运算。

例 1.7　试用卡诺图判断下列两组逻辑函数 Y_1 和 Y_2 有何关系：

$$Y_1 = A\overline{C} + B\overline{C} + AB$$

$$Y_2 = \overline{A}C + \overline{B}C + \overline{A}\overline{B}$$

题意分析　卡诺图是反映逻辑变量及其组合关系的图表，任何一个逻辑函数都可在卡诺图中得出特定的表示。本例便是要求用卡诺图表示两个逻辑函数，并比较它们之间的关系。由于函数式是以非标准的形式给出，只要按照与、或、非逻辑在卡诺图上的基本属性，即与逻辑在卡诺图上具有区域的公共性；或逻辑在卡诺图上具有区域的叠加性；非逻辑在卡诺图上具有逻辑否定性，就可填写出相应的卡诺图。

解　根据 Y_1,Y_2 的逻辑表达式画出相应的卡诺图，如图 1.5 所示。

Y_1

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

Y_2

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

图 1.5　例 1.7 函数 Y_1,Y_2 卡诺图

可知，函数 Y_1 和 Y_2 互为反函数。

讨论　卡诺图的用途很广。例如，利用卡诺图可将任一逻辑函数展开成两种标准式，即标准与-或式和标准或-与式；也可利用卡诺图证明两函数相等、互补及进行异或叠加运算；更重要的用途是进行逻辑函数化简。本例要求用卡诺图判别两个函数的互补关系。由解题过程可知，由于避免了用代数法推证时繁杂的逻辑运算，因此更为直观、简便。

例 1.8　试用卡诺图法将逻辑函数

$$Y = \sum m(5,6,7,8,9)$$

化简成最简与-或式及最简与非-与非式。

Y

CD \ AB	00	01	11	10
00				1
01		1		1
11		1		
10		1		

图 1.6　例 1.8 函数 Y 卡诺图化简

题意分析　n 个变量的卡诺图包含了 n 个变量的全部(即 2^n 个）最小项，每一个最小项对应于卡诺图上一指定的小方格。而任意一个 n 变量的逻辑函数都可以由 m 个($0 < m \leqslant 2^n$) 最小项之和的形式表示，这 m 个最小项的变量取值使函数值唯一地为 1。利用卡诺图化简逻辑函数，首先要把逻辑函数填入相应的卡诺图中。本例逻辑函数是由 4 变量构成，并以最小项之和的形式表示，因此，只要选用 4 变量卡诺图，分别在组成函数式的各最小项对应的小方格中填 1(称为 1 格)，其余的小方格填 0(称为 0 格)，就可把逻辑函数表示出来。然后将卡诺图上相邻的 1 格(必须是 2^i 个相邻的 1 格其中 $i = 1,2,3,\cdots$) 合并圈为一块，并把这一圈中既取值为 0 又

取值为 1 的变量消去，就可达到化简的目的。

解　设给定逻辑函数由 A、B、C、D 4 变量构成。则有：

$$Y = f(A,B,C,D) = \sum m(5,6,7,8,9)$$

卡诺图如图 1.6 所示。

化简后得出最简的与-或式为

$$Y = \overline{A}BD + \overline{A}BC + A\overline{B}\overline{C}$$

经逻辑变换后得出最简的与非-与非式为

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

讨论　本例给出一个重要的关系。若是要把一个逻辑函数化简为最简的与非-与非式，只需用卡诺图将给定函数化简成最简的与-或式，然后将每一个与项（包括单变量）及整个或式都用与非代替就可以了。

例 1.9　试用卡诺图法将逻辑函数

$$Y = \overline{(BD + \overline{A}C)B}$$

化简成最简或-与式及最简或非-或非式。

题意分析　在用卡诺图化简逻辑函数时，在函数的卡诺图中，可合并相邻的 1 格得出原函数的最简与-或式；也可合并相邻的 0 格得出反函数的最简与-或式，然后再求反，即可得出原函数的最简或-与式。经逻辑变换后也可得出函数的最简或非-或非式。

解　给定逻辑函数式的卡诺图如图 1.7 所示。

圈 0 得出反函数的最简与-或式为

$$\overline{Y} = BD + \overline{A}BC$$

将上式求反即可得出函数式的最简或-与式为

$$Y = \overline{BD + \overline{A}BC} = (\overline{B} + \overline{D})(A + \overline{B} + \overline{C})$$

经逻辑变换后，函数式的最简或非-或非式为

$$Y = \overline{\overline{(\overline{B} + \overline{D})(A + \overline{B} + \overline{C}))}} = \overline{(\overline{B} + \overline{D})} + \overline{(A + \overline{B} + \overline{C})}$$

Y

CD \ AB	00	01	11	10
00	1	1	1	1
01	1	0	0	1
11	1	0	0	1
10	1	0	1	1

图 1.7　例 1.9 函数 Y 卡诺图化简

讨论　本例给出一个重要的关系。若要把一个逻辑函数化简为最简的或非-或非式，简便的方法是：在函数的卡诺图上圈 0 得出反函数的最简与-或式，然后将每一个与项（包括单变量）及整个或式均用或非代替，并把原变量变为反变量，反变量变为原变量就可以了。

例 1.10　试用卡诺图法将逻辑函数 $Y = \Pi M(4,5,6,7,9,11)$ 化简成最简或-与式。

题意分析　本例逻辑函数是以标准或-与式，即最大项之积的形式表示，要求用卡诺图法化简成最简的或-与式。常用的方法是利用对偶规则将函数式对偶成标准与-或式，即最小项之和的形式进行化简。得出对偶式的最简与-或式后，再作对偶，即可得出原函数的最简或-与式。

解　设给定逻辑函数由 A,B,C,D 4 变量构成。则有：

$$Y = f(A,B,C,D) = \Pi M(4,5,6,7,9,11)$$

其对偶式为：

$$Y' = f(A,B,C,D) = \sum m(4,6,8,9,10,11)$$

卡诺图如图 1.8 所示。

化简后得出最简与-或式为：

$$Y' = A\overline{B} + \overline{A}B\overline{D}$$

再作对偶，得出原函数式的最简或-与式为：

$$Y = (A + \overline{B})(\overline{A} + B + \overline{D})$$

讨论 本例给出一种利用对偶规则的求解方法，另一种方法是利用反演规则求解。由于最大项与最小项之间，若是序号相同则二者互补，即 $m_i = \overline{M_i}$。因此，本例给出的逻辑函数，按反演规则得出它的反函数式为：

$$\overline{Y} = \sum m(4,5,6,7,9,11)$$

卡诺图如图 1.9 所示。

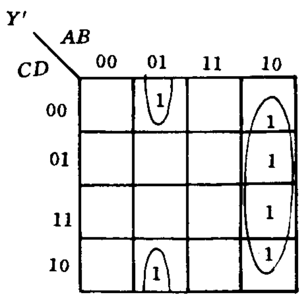


图 1.8 例 1.10 对偶函数 Y' 卡诺图化简

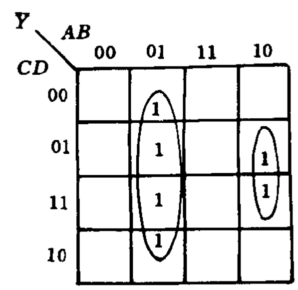


图 1.9 例 1.10 反函数 $\overline{Y}$ 卡诺图化简

化简后得出最简与-或式为：

$$\overline{Y} = \overline{A}B + A\overline{B}D$$

再求反，得出原函数式的最简或-与式为：

$$Y = \overline{\overline{A}B + A\overline{B}D} = (A + \overline{B})(\overline{A} + B + \overline{D})$$

两种方法结果相同。而且适用于用标准的或是非标准的或-与式表示的逻辑函数的化简。

例 1.11 试用卡诺图法将逻辑函数

$$Y = \sum m(5,6,7,8,9) + \sum d(10,11,12,13,14,15)$$

化简成最简与非-与非式。

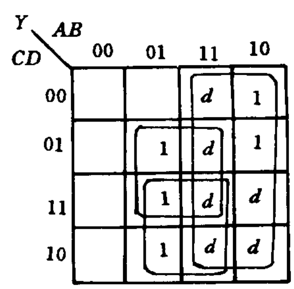


图 1.10 例 1.11 函数 Y 卡诺图化简

题意分析 本例给出的逻辑函数是一种不完全描述的逻辑函数，或称为具有无关最小项的逻辑函数。由于无关最小项对于逻辑函数是一种约束条件，它的含义一是这些最小项不允许出现，二是即使出现也不影响电路的逻辑功能。因此，在用卡诺图化简逻辑函数时，可将无关最小项视为 1 格或 0 格参于合并项中，使逻辑函数得出最简的表达形式。

解 设给定逻辑函数由 A,B,C,D 4 变量构成。则有：

$$Y = f(A,B,C,D) = \sum m(5,6,7,8,9) + \sum d(10,11,12,13,14,15)$$