高等学校教学参考书

数字电子技术基础
重点、难点、题解、试题

侯建军　主编
路而红　娄淑琴　编

高等教育出版社

内容简介

本书是为配合《数字电子技术基础》（侯建军主编，高等教育出版社出版）教材使用而编写的辅助教材。内容包括主教材的基本知识、重点与难点、考核题型与考核重点、综合分析、设计典型题型精解、主教材题解。全书共有九章：数字逻辑基础、逻辑门电路、组合逻辑电路、时序逻辑电路、常用时序集成电路模块及其应用、可编程逻辑器件 PLD、D/A 转换器和 A/D 转换器、脉冲产生与整形和数字系统设计。书末附有模拟试题，可供读者复习时参考。

本书可作为高等院校通信、电子、自动化和计算机等专业学生的辅助教材，也可作为教师的教学手册，还可供相关工程技术人员自学和参考。

图书在版编目（CIP）数据

数字电子技术基础重点、难点、题解、试题/侯建军主编.—北京：高等教育出版社，2005.4
ISBN 7-04-016458-2

Ⅰ.数… Ⅱ.侯… 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（2005）第 013132 号

策划编辑 吴陈滨 **责任编辑** 刘 洋 **封面设计** 李卫青 **责任绘图** 朱 静
版式设计 王 莹 **责任校对** 王 超 **责任印制** 朱学忠

出版发行	高等教育出版社	**购书热线**	010-58581118
社　　址	北京市西城区德外大街 4 号	**免费咨询**	800-810-0598
邮政编码	100011	**网　　址**	http://www.hep.edu.cn
总　　机	010-58581000		http://www.hep.com.cn
		网上订购	http://www.landraco.com
经　　销	北京蓝色畅想图书发行有限公司		http://www.landraco.com.cn
印　　刷	肥城新华印刷有限公司		
开　　本	787×960 1/16	**版　　次**	2005 年 4 月第 1 版
印　　张	22.5	**印　　次**	2005 年 4 月第 1 次印刷
字　　数	420 000	**定　　价**	28.10 元

物料号 16458-00

前　言

本书是为配合《数字电子技术基础》(侯建军主编,高等教育出版社出版)教材使用而编写的辅助教材。在编写过程中,采取了以课程的教学基本要求为基点,某些方面又略高于基本要求的原则,为适合教学改革需要,同时又引入了新技术、新概念和新方法,目的是为适应数字技术飞速发展的需要。

本书内容提供了数字逻辑基础、逻辑门电路、组合逻辑电路、时序逻辑电路、常用时序集成电路模块及其应用、可编程逻辑器件 PLD、D/A 转换器和 A/D 转换器、脉冲产生与整形和数字系统设计的习题约 480 题。每章由基本知识、重点与难点,例题精选,习题解答三部分组成。重点与难点主要阐述各章基本内容的要求,为读者提供了学习各章的方向和目标,同时为学生顺利掌握本课程内容、排除困难给予指导。典型例题是教材中心和重点内容的应用扩展。通过综合分析和设计的典型例题描述了一个完整的解题思路,帮助读者提高对数字电路的综合运用能力。为方便读者复习考试,教材中给出了各章的考核题型与考核比例分数。最后给出数字电子技术课程模拟试题 6 套,供学生全面复习使用。

读者在学习时,应以教材为基础,本书可作为理解、掌握、应用和加深巩固的工具和钥匙。

本书由北京交通大学侯建军任主编,负责全书编写的组织与定稿,其中第一、二章由娄淑琴编写,第三、四、八章由路而红编写,第五、六、七、九章和模拟试题由侯建军编写,北京交通大学张有根对本书进行了审阅。

在本书编写过程中,得到了许多教师的关心和支持,在此致以衷心的感谢。

对本书选用的参考文献的著作者,我们真诚地致以感谢。

虽然教材中的部分内容试用过,但书中难免还有错误和不妥之处,恳请读者批评指正。

编　者

2004.8 于北京

目　　录

第一章　数字逻辑基础

由于数字信号便于存储、处理和传输，数字化已成为当今电子技术的发展潮流，数字系统已经成为人们日常生活中的重要组成部分。本章主要介绍数字电路的基础知识，为读者学习和掌握数字电路的分析和设计奠定数学基础。

通过本章内容的学习，要求掌握各种不同进制数与二进制码，逻辑代数的基本规律，并能熟练地运用卡诺图化简逻辑函数。

第一节　基本知识、重点与难点

一、基本知识

（一）数制与编码

1. 数制

数制就是进位计数制。数字电路中常用的数制有二进制、八进制、十进制、十六进制，用后缀 B、O、D、H 来区别。

对于任意 R 进制数存在共有规律：

（1）一个确定的基数 R，且逢 R 进一。

（2）有 R 个有序的数字符号和一个小数点，数码 K_i 范围为 $0\sim(R-1)$。

（3）每一个数位均有固定的含意，称权 R^i，不同数位其权 R^i 不同。

（4）进位制数均可写成按权展开式，式中每一项为该位的数码 K_i 和该位的权 R^i 的乘积。

2. 数制转换

一个数从一种进位制表示形式转换成等值的另一种进位制表示形式称为数制转换，其实质为权值转换。相互转换的原则是转换前后两个有理数的整数部分和小数部分必定分别相等。

（1）十进制数与非十进制数转换

整数转换采取除基取余法，小数转换采取乘基取整法。将非十进制数按权展开，然后十进制求和，可得到所对应的十进制数。

（2）非十进制数之间的转换

二进制转换成八进制的原则是从小数点开始，将二进制数的整数和小数部分每 3 位分为一组，不足 3 位的分别在整数的最高位前和小数的最低位后加 **0**

补足,然后每组用等值的八进制码替代,即得目的数。八进制转换成二进制的原则是每位八进制数用3位二进制数代替即可。

二进制转换成十六进制的原则是从小数点开始,将二进制数的整数和小数部分每4位分为一组,不足4位的分别在整数的最高位前和小数的最低位后加**0**补足,然后每组用等值的十六进制码替代,即得目的数。十六进制转换成二进制的原则是每位十六进制数用4位二进制数代替即可。

3. 常用编码

常用的有自然二进制码、格雷码、二-十进制码等。二-十进制编码(Binery Coded Decimal)简称BCD码。它用二进制代码对十进制数的各个数码进行编码。二-十进制编码有很多种,最常用的是8421BCD码(有权码)、余3码(无权码),同一个十进制数所对应的余3码等于所对应的8421BCD码加**0011**(3)。

(二) 逻辑代数与基本逻辑运算

逻辑代数又称开关代数或布尔代数,是按一定逻辑规律进行运算的代数,是分析和设计逻辑电路的基本工具和理论基础,其特点是它的所有变量与函数值仅有两个特征值——逻辑**0**和逻辑**1**,该特征值并不代表数值的大小,仅表示相互矛盾、相互对立的两种逻辑状态。逻辑代数的公式、规则、定理与定义均需用二值的因果关系来理解。

逻辑代数有三种基本逻辑运算,即**与**、**或**、**非**,对应的是逻辑**与**、逻辑**或**和逻辑**非**。利用这三种基本运算,可得出处理实际逻辑问题的各种常用的复合逻辑,如**与非**、**或非**、**与或非**、**异或**、**同或**等。

(三) 逻辑代数的运算公式和规则

1. 基本公式和常用公式

逻辑代数的基本公式和常用公式满足自等律、互补律、分配律及吸收律等等。

2. 三个基本规则

(1) 代入规则

(2) 反演规则

(3) 对偶规则

(四) 逻辑函数的标准形式

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。逻辑函数具有两种标准形式:一是最小项之和(标准**与或**表达式),如果函数的积之和(**与或**)表达式中的每一个乘积项均为最小项,则这种表达式称为标准积之和表达式,也称最小项表达式。一是最大项之积(标准**或与**表达式),如果函数的和之积(**或与**)表达式中的每一个和项均为最大项,则这种表达式称为标准和之积表达式,也称为最大项表达式。

任何一个函数两种标准式中所含最小项和最大项编号是互不重复而相互补充的。但需注意，对于 n 个变量的函数，两式所含编号应为 $0\sim2^n-1$，且互不重复，这些最小项和最大项个数总和为 2^n 个。

（五）逻辑函数的化简

1. 代数法化简逻辑函数

代数法化简的实质是反复运用逻辑代数的运算公式和规则，消去表达式中的多余项和多余因子，以达最简目的。

利用代数法化简对函数变量数目无限制，但方法灵活、技巧性强，需要熟练掌握逻辑代数的基本公式和具有一定的化简经验。

2. 图解法化简函数

图解法又称为卡诺图(Karnaugh Map)法，这种方法的优点是比较直观，可以由图直接写出函数的最简表达式，简化技巧和规律比代数法容易掌握，但它一般运用在五变量以下函数的化简。

卡诺图也是真值表的另一种表示形式，它是将最小项按相邻关系排列的一种方格图形。卡诺图中的一小格对应真值表中的一行，即对应一个最小项，故卡诺图又称真值图。相邻关系是指最小项内所含的变量中只有一个变量互补，反映在卡诺图上为几何位置相邻。

化简步骤可归纳为：

① 将函数填入相应的卡诺图中。

② 按作圈原则将图上填 **1** 的方格圈起来，要求圈的数量少、范围大；每个为 **1** 的格都应被圈入，允许被多个圈重复圈入，但每个圈都必须有新的，即不被其他圈包围的格；这样每个圈均用其对应的乘积项表示。

③ 最后将全部乘积项逻辑加得最简**与或**表达式。

3. 具有无关项的逻辑函数的简化

在实际问题中常会出现两种情况：第一种是输入变量的取值受到限制也称受到约束，它们对应的最小项称约束项。第二种情况是在某些输入变量取值下函数值是 **1** 或 **0** 并不影响整个电路系统的功能，这些变量取值所对应的最小项称为任意项。

约束项和任意项在逻辑函数中统称为无关项。

具有无关项的函数若利用代数法化简是十分困难的，而采用图解法则简单、直观，在填函数的卡诺图时只需在无关项对应的格内填任意符号“∅”、“d”或“×”，根据作圈的需要将这些格可视为 **1** 也可视为 **0**，从而达到简化逻辑函数的目的。

二、重点与难点

重点：

1. 数制

2. 编码

(1) 二－十进制码(BCD 码)

在这种编码中，用 4 位二进制数表示十进制数中的 0～9 十个数码。常用的编码有 8421BCD 码、5421BCD 码和余 3 码。

8421BCD 码是由 4 位二进制数 **0000** 到 **1111** 十六种组合中前十种组合，即 **0000**～**1001** 来代表十进制数 0～9 十个数码，每位二进制码具有固定的权值 8、4、2、1，称有权码。

余 3 码是由 8421BCD 码加 3(**0011**)得来的，是一种无权码。

(2) 格雷码

格雷码是一种常见的无权码。这种码的特点是相邻的两个码组之间仅有 1 位不同，因而其可靠性较高，广泛应用于计数和数字系统的输入、输出等场合。

3. 逻辑代数基础

(1) 逻辑代数的基本公式与基本规则

逻辑代数的基本公式反映了二值逻辑的基本思想，是逻辑运算的重要工具，也是学习数字电路的必备基础。

逻辑代数有三个基本规则，利用代入规则、反演规则和对偶规则使逻辑函数的公式数目倍增。

(2) 逻辑问题的描述

逻辑问题的描述可用真值表、函数式、逻辑图、卡诺图和时序图，它们各具特点又相互关联，可按需选用。

(3) 图形法化简逻辑函数

图形法比较适合于具有三、四变量的逻辑函数的简化。

难点：

1. 给定逻辑函数，将逻辑函数化为最简

用代数法化简逻辑函数，要求熟练掌握逻辑代数的基本公式和规则，熟练运用四个基本方法——并项法、消项法、消元法及配项法对逻辑函数进行化简。

用图形法化简逻辑函数时，一定要注意卡诺图的循环邻接的特点，画包围圈时应把每个包围圈画得尽可能大。

2. 卡诺图的灵活应用

卡诺图除用于化简函数外，还可以用来检验化简结果是否最简、判断函数间的关系、求函数的反函数和逻辑运算等。

3. 电路的设计

在工程实际中，往往给出逻辑命题，如何正确分析命题，设计出逻辑电路呢？通常的步骤如下：

① 根据命题，列出反映逻辑命题的真值表。

② 根据真值表，写出逻辑表达式。

③ 对逻辑表达式进行变换化简。

④ 最后按工程要求画出逻辑图。

三、考核题型与考核重点

1. 概念与简答

题型 1 为填空、判断和选择。

题型 2 为叙述基本概念与特点。

建议分配的分数为 2 ~ 4 分。

2. 综合与设计

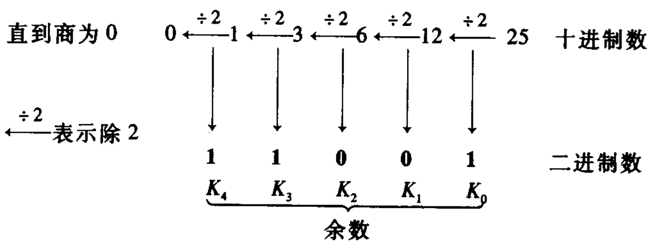
题型为与后续章节内容相结合的综合型题目。

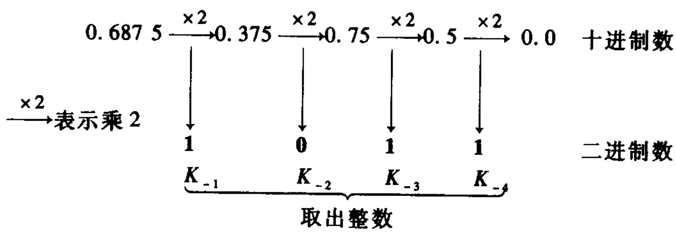
建议分配的分数为 3 ~ 6 分。

第二节　典型题解

例题 1.1　将十进制数 25.687 5 转换为二进制数、八进制数、十六进制数，即 $(25.6875)_D = (\quad)_B = (\quad)_O = (\quad)_H$。

解：





（1）首先将十进制数25.687 5转换成二进制数，分别对整数部分和小数部分进行转换。整数部分采用除2取余法，小数部分采用乘2取整法。

由此可得：$(25.6875)_D=(\mathbf{11001.1011})_B$

（2）将二进制数转换成八进制数，以小数点为界将二进制数3位一组，进行划分，最后每3位用一个等值八进制数代替即可。

$$(\mathbf{11001.1011})_B=(31.54)_O$$

（3）将二进制数转换成十六进制数，以小数点为界将二进制数4位一组，进行划分，最后每4位用一个等值十六进制数代替即可。

$$(\mathbf{11001.1011})_B=(19.B)_H$$

综上可得：$(25.6875)_D=(\mathbf{11001.1011})_B=(31.54)_O=(19.B)_H$

例题1.2　求下列函数的反函数。

（1）$F=\overline{AB+C}\cdot D+AC$

（2）$F=\overline{\overline{A+C}(\overline{BC}+D)}(B+C)+AD$

解：反演规则对任意一个逻辑函数式 F 中的运算符“·”换成“+”，“+”换成“·”；常量 **0** 换成 **1**，**1** 换成 **0**；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 $\overline{F}$。

变换时应注意两点：

① 必须保持原函数的运算次序，适当地加入括号。

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。

由此可得

（1）$\overline{F}=[\overline{(\overline{A}+\overline{B})\cdot\overline{C}}+\overline{D}](\overline{A}+\overline{C})$ 或 $\overline{F}=(AB+C+\overline{D})(\overline{A}+\overline{C})$

（2）$\overline{F}=(\overline{\overline{\overline{A}\,\overline{C}}+\overline{\overline{B}}+\overline{\overline{C}\cdot\overline{D}}}+\overline{B}\,\overline{C})(\overline{A}+\overline{D})$ 或 $\overline{F}=[\overline{A+C}(\overline{BC}+D)+\overline{B}\,\overline{C}](\overline{A}+\overline{D})$

例题1.3　求下列函数的对偶式。

（1）$F=\overline{\overline{A}B+A\overline{B}}(AC+CD)$

（2）$F=A\cdot\overline{B+\overline{D}}+(AC+BD)E$

解：对于任意一个逻辑函数 F，若把式中的运算符“·”换成“+”，“+”换成“·”；常量 **0** 换成 **1**，**1** 换成 **0**，且保持运算的先后顺序不变，所得的新函数式为原函数式 F 的对偶式 F'。

对偶规则对函数中的原变量、反变量不进行变换，而反演规则包含原变量和反变量之间的变换。和反演规则相同的是，变换过程中原函数的运算的先后顺序均保持不变，且不属于单个变量上的非号保持不变。

由此得

(1) $F' = \overline{(\overline{A}+B)(A+\overline{B})} + (A+C)(C+D)$

(2) $F' = (A+\overline{B\overline{D}})[(A+C)(B+D)+E]$

例题 1.4　证明下列等式成立(方法不限)。

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

证明:通常可运用逻辑代数中的公式和定理对等式进行变换加以证明。证明时可用不同的公式和方法进行,要注意选用比较精练的方法。在变量较少的情况下,也可选用真值表加以证明。

方法一:等式左边 $= (A+B)(\overline{A}+C)(B+C)$

$$\begin{aligned}
&= (A+B)(\overline{A}B+C)\\
&= \overline{A}B+AC+BC\\
&= (A+B)C+(A+B)\overline{A}\\
&= (A+B)(\overline{A}+C)\\
&= \text{等式右边}
\end{aligned}$$

因此等式成立。

方法二:利用对偶规则加以证明

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

两式的对偶函数为

$$F' = AB+\overline{A}C+BC = AB+\overline{A}C$$
$$G' = AB+\overline{A}C$$

由于 $F' = G'$,所以 $F = G$,等式成立。

方法三:利用真值表加以证明,如例题 1.4 表所示。

例题 1.4 表

A　B　C	$(A+B)(\overline{A}+C)(B+C)$	$(A+B)(\overline{A}+C)$
0　0　0	0	0
0　0　1	0	0
0　1　0	1	1
0　1　1	1	1
1　0　0	0	0
1　0　1	1	1
1　1　0	0	0
1　1　1	1	1

从真值表可看出，对于输入变量的所有组合，等式两边的值均相等，因此等式成立。

例题 1.5　将下面的四变量函数用卡诺图化简为最简**与或**式。

$$F_1(A,B,C,D)=\overline{B}C+\overline{B}\,\overline{D}+A\overline{C}D+AB\overline{C}+\overline{A}\,\overline{B}\,\overline{C}D$$

$$F_2(A,B,C,D)=\sum m(1,7,9,10,11,12,13,15)$$

解：卡诺图化简逻辑函数的依据是：含 n 个变量函数的卡诺图中，几何相邻的 $2^i(i=1、2、3、\cdots、n)$ 个小格可合并在一起构成正方形或矩形圈，消去 i 个变量，而用含 $n-i$ 个变量的乘积项标注该圈。

根据函数填写卡诺图方法：

（1）若已知函数为最小项表达式，则只需将函数中包含的那些最小项对应的格填 **1**，其余格均填 **0**。

（2）若已知函数的真值表，则只需将其真值表中使函数值为 **1** 的那些最小项对应的方格填 **1**，其余格均填 **0**。

（3）若已知函数为一个复杂的运算式，则应首先将其变成**与或**式，再用直接法填写。

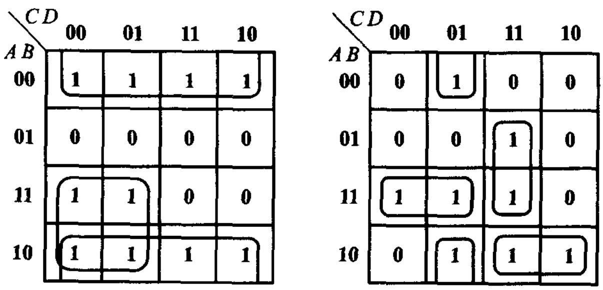
作圈的步骤：

① 首先将所有孤立的单格和只有一个合并方向的格群圈起来。

② 余下的格均有多种合并方向，应用试探法合理作圈，必要时有的格允许被多个圈重复圈入。

③ 图中含 **1** 的格都应被圈入，以防止遗漏乘积项，且每个圈中至少有一格只被圈过一次，否则该圈就是冗余圈。

综上所述，作出函数的卡诺图，如例题 1.5 图所示。



(a) F_1的卡诺图　　　　(b) F_2的卡诺图

例题 1.5 图

然后画包围圈，注意圈的数量要尽可能少、范围要尽可能大。最后写出化简后的逻辑函数表达式。

$$F_1 = \overline{B} + A\overline{C}$$

$$F_2 = AB\overline{C} + A\overline{B}C + BCD + \overline{B}\,\overline{C}D$$

应注意的是：利用卡诺图对逻辑函数进行化简时，最简的结果并不是唯一的。

例题 1.6　将下列函数化为最简**与或非**式。

$$F_1(A,B,C,D) = \sum m(3,5,7,8,12) + \sum d(0,1,10,11,14,15)$$

$$\begin{cases} F_2(A,B,C,D) = AB\overline{C} + AB\overline{D} + \overline{A}BC + AC\overline{D} \\ \overline{B}\,\overline{C} + \overline{B}CD = 0 \end{cases}$$

解：利用卡诺图化简函数为最简**与或非**式时，可考虑在卡诺图中先圈 **0**，再将圈 **0** 所得的函数化为最简**与或**式，然后再在最简式上加反，即得最简**与或非**式。

此法可简称为圈 **0** 加反法。具体解法为：

(1) 画函数的卡诺图

将最小项和约束项（或者函数的约束条件）填入图中。

(2) 合并相邻的最小项

将最小项为 **0** 的方块按卡诺图化简函数为最简**与或**式的方法画圈，尽可能使所画的圈少而大，其中画入圈中的约束项看作 **0**，而没有画入圈的约束项看作 **1**。

函数的卡诺图和具体的画圈法如例题 1.6 图所示。

(3) 写函数的最简**与或非**式

对圈 **0** 的函数求反，即可得原函数的最简**与或非**式。函数的表达式为

$$F_1 = \overline{AD + \overline{A}\,\overline{D}}$$

$$F_2 = \overline{\overline{A}\,\overline{B} + \overline{A}\,\overline{C} + ACD}$$

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

(a) F_1的卡诺图

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

(b) F_2的卡诺图

例题 1.6 图

例题 1.7　已知函数

$$\begin{cases} F_1 = A\,\overline{C}D + \overline{A}B\,\overline{D} + BCD + \overline{A}CD \\ F_2 = \overline{\overline{A}\,\overline{C}\,\overline{D} + BC + A\,\overline{C}\,\overline{D}} \end{cases}$$

求函数 $F_1 \cdot F_2$、$F_1 + F_2$、$F_1 \oplus F_2$。

解：逻辑非的运算即为求反函数，故只要将函数在卡诺图中 **0** 和 **1** 的位置对调一下，即得原函数的反函数。图中若有约束项其位置不变，仍为约束项。

函数在进行**与**、**或**、**异或**的运算时，只要将图中编号相同的方块，按运算规则进行运算，就可求得它们的逻辑**与**、逻辑**或**、逻辑**异或**等函数。

画函数 F_1、F_2 卡诺图，如例题 1.7 图所示。画 F_2 卡诺图时先画出其反函数 $\overline{F_2}$ 的卡诺图，然后将 $\overline{F_2}$ 的卡诺图中 **0** 和 **1** 的位置对调，即例题 1.7 图所示的原函数 F_2 的卡诺图。

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

(a) F_1的卡诺图

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

(b) F_2的卡诺图

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

$F_1 \cdot F_2$

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

$F_1 + F_2$

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

$F_1 \oplus F_2$

(c) $F_1 \cdot F_2$、$F_1 + F_2$、$F_1 \oplus F_2$ 函数卡诺图

例题 1.7 图

根据卡诺图进行**与**、**或**、**异或**的运算，分别对 F_1、F_2 卡诺图中编号相同的方块进行上述的运算可得例题 1.7 图所示的函数 $F_1 \cdot F_2$、$F_1 + F_2$、$F_1 \oplus F_2$ 的卡诺图。

经化简可得函数 $F_1 \cdot F_2$、$F_1 + F_2$、$F_1 \oplus F_2$ 的表达式为

$$F_1 \cdot F_2 = A\,\overline{C}D + \overline{A}\,\overline{B}CD$$

$$F_1 + F_2 = D + \overline{A}B + \overline{B}C$$

$$F_1 \oplus F_2 = \overline{A}B + \overline{A}\,\overline{C}D + ACD + \overline{B}C\overline{D}$$

例题 1.8　在只有原变量输入情况下实现下列函数。

$$F(A,B,C,D) = \sum m(2,3,4,5,6,7,8,9,10,11,14)$$

解：在某些设计中，只允许原变量输入，此时最简单的办法是加入非门以产生必要的反变量，但这样并非最经济合理，这里以用**与非**门实现电路为例来合理设计。

设计中常用到几个基本关系：

（1）含相同原变量的乘积项可以合并。如：

$$AB\overline{C} + AB\overline{D} = AB(\overline{C} + \overline{D}) = AB\overline{CD}$$

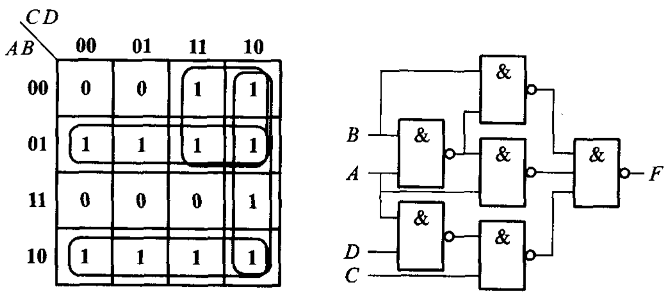
（2）乘积项中的反变量可以用适当的**与非**因子代替。如：

$$AB\overline{C} = AB\overline{C} + AB\overline{B} = AB\overline{BC}$$

同理

$$AB\overline{C} = AB\overline{AC} = AB\overline{ABC}$$

其中，$\overline{BC}$、$\overline{AC}$和$\overline{ABC}$称为$\overline{C}$的代替因子，其意是一个乘积项中的反变量因子可根据需要扩展成为与这一乘积项中的其他一个或几个变量构成的**与非**因子，该乘积项的值仍不会变。选哪个代替因子取代该反变量，要根据其他乘积项中有否相同的代替因子，这样可减少所需**与非**门的数量。首先作卡诺图如例题 1.8 图(a)所示。



(a)　(b)

例题 1.8 图

简化函数得

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

再将前两项 $\overline{A}B$、$A\overline{B}$ 中的 $\overline{A}$ 和 $\overline{B}$ 都用代替因子$\overline{AB}$取代，于是这两项分别等效为 $B\overline{AB}$和 $A\overline{AB}$，而后两项合并为 $C\overline{AD}$，因此

$$\begin{aligned} F &= B\overline{AB} + A\overline{AB} + C\overline{AD} \\ &= \overline{\overline{B\overline{AB} + A\overline{AB} + C\overline{AD}}} \\ &= \overline{\overline{B\overline{AB}} \cdot \overline{A\overline{AB}} \cdot \overline{C\overline{AD}}} \end{aligned}$$

其逻辑电路图如例题 1.8 图(b)所示。