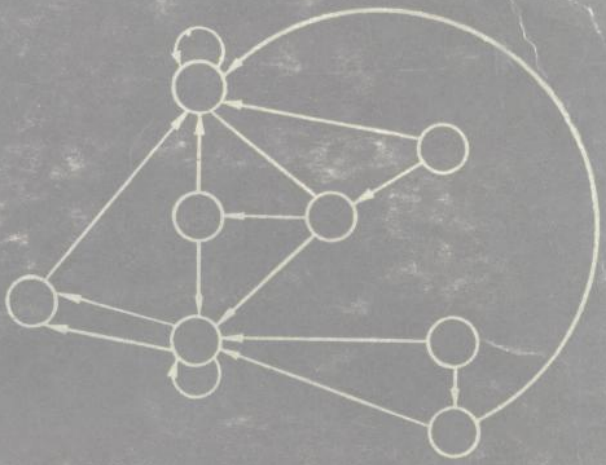

JI CHENG SHU ZI DIAN LU DE LUO JI SHE JI

集成数字电路的逻辑设计

雍新生主编

复旦大学出版社

集成数字电路的逻辑设计

雍新生　主编

复旦大学出版社

内 容 提 要

全书共分七章。前两章为基础部分，第一章介绍分析和设计数字电路的数学工具——开关代数及逻辑函数的化简方法，第二章介绍数字电路的基本部件——门电路的结构及基本参数。第三至五章为全书的核心，分别介绍组合逻辑电路、同步时序电路和异步时序电路的分析和设计方法。第六章介绍数字模块和数字系统的设计原理，列举了一些由中规模电路构成的数字模块。第七章为门阵列逻辑设计，着重介绍 ROM（只读存贮器）和 PLA（可编逻辑阵列）的应用。

本书具有简明实用的特点，可作为大专院校无线电专业的教材及教学参考书，也可供有关工程技术人员自学参考。

集成数字电路的逻辑设计

雍 新 生 主编

复旦大学出版社出版

新华书店上海发行所发行

复旦大学印刷厂印刷

字数421千 开本850×1168 1/32 印张14.75

1987年2月第1版 1987年2月第1次印刷.

印数：1—7,000

书号：13253·048 定价：2.45元

前　言

本书是作者在为复旦大学无线电及微电子专业学生讲授“数字电路与逻辑设计”课程中，总结了各届教学实践经验，经过较大修改和补充后整理而成的。实践证明：在使用本教材的同时，配合适当的脉冲与数字电路实验，能使学生较好地掌握数字电路及数字系统的设计方法。

本书力求做到理论联系实际、由浅入深，通过某些实例的分析，使读者掌握数字电路分析和设计的基本理论。同时，考虑到尽量反映新技术，故对自动逻辑设计和阵列化设计也作了介绍。

全书包括三大部分：前两章介绍有关数字电路的基础知识，即“开关代数和逻辑化简”及“基本门电路”。第二部分是本书的核心，着重介绍数字电路：第三章的“组合逻辑”和第四、五章的“时序逻辑的分析”和“设计方法”。通过具体例题的介绍，归纳出分析和设计的一般步骤。通过前五章的学习，要求读者掌握逻辑电路的具体设计方法并能选用适当的器件来加以实现。第三部分进一步介绍中、大规模逻辑电路的实现。第六章介绍了一些常用数字模块，如运算电路、多路器和计数器等，并介绍了数字系统设计及自动逻辑设计的基础知识。第七章则为阵列化逻辑设计，包括用 ROM 和 PLA 进行逻辑设计的方法，并介绍了大规模集成电路中的门阵列逻辑设计方法。通过这两章的学习，读者可进一步掌握中、大规模数字电路的应用。

全书由雍新生主编，其中第一、二章由邱友瑾编写，第三章由雍新生、邱友瑾合编（由雍新生执笔），第四章、第五章及第六章的一、二、三节由雍新生编写。第六章的第四节及第七章由童家榕编写，他还对第四、五、六章的内容进行了校阅。

谢悦华等同志参加了本书的绘图工作，在此一并致谢。

作者水平有限，书中难免有欠妥甚至错误之处，恳请读者批评指正。

一九八五年八月

目　录

第三章 组合电路的分析与综合

第四章 同步时序电路的分析与设计

第五章 异步时序电路的分析与综合

第六章 数字模块与数字系统概述

第七章 阵列化逻辑设计

第一章　开关代数和开关函数及其化简

在介绍数字电路的逻辑设计以前，本章先讨论一些基本问题。主要包括二进制数及不同数制之间的转换，开关代数及其基本定律、开关函数及其化简。

平时我们一般使用十进制的数，而在计算机内部大多是采用二进制，这就产生了不同数制之间的转换问题。采用了二进制数后，数之间的函数关系称为**开关函数**。开关代数是研究用开关函数进行逻辑设计的基本数学工具。本章将概括地说明开关代数的基本运算规则和定理。用这些基本定律可以设计和化简逻辑电路。

除了用代数的方法表示和化简开关函数之外，还可以用几何图形的方法来表示和化简开关函数。用卡诺图表示开关函数的方法称为**卡诺图化简法**，它的特点是明确、直观，尤其适用于变量比较少的开关函数的化简。对于变量较多的函数则适合于用计算机来进行化简，本章对此也将作简单介绍。

第一节　二进制数及不同数制间的转换

平常人们习惯于十进制。如一个十进制数 456.78 可以表示为：

$$456.78=4\times10^{2}+5\times10^{1}+6\times10^{0}+7\times10^{-1}+8\times10^{-2}$$

实际上也可以采用其他数制来表示并进行运算，如八进制、二进制等等。一个八进制数 (135.76)，如用八进制表示，它的值即为：

$$(135.76)_{8}=1\times8^{2}+3\times8^{1}+5\times8^{0}+7\times8^{-1}+6\times8^{-2}$$

二进制数 $(1100101.101)_{2}$，用十进制表示，它的值即为：

$$(1100101.101)_{2}=1\times2^{6}+1\times2^{5}+0\times2^{4}+0\times2^{3}+1\times2^{2}+0\times2^{1}+1\times2^{0}+1\times2^{-1}+0\times2^{-2}+1\times2^{-3}$$

一般地说，对于 r 进制的具有 n 位整数和 m 位小数的数 A_r 可以表示为：

$$A_r=a_{n-1}r^{n-1}+a_{n-2}r^{n-2}+\cdots+a_0r^0+a_{-1}r^{-1}+\cdots+a_{-m}r^{-m} \quad (1.1.1)$$

在数字系统中究竟采用什么数制呢？主要取决于：

(1) 在物理上能够实现，即能用器件的不同物理状态明确表示不同的数字。

(2) 要求尽量简单。在物理上最容易实现的是有两个稳定状态的器件，如开关的"开"和"关"，电灯的"亮"和"暗"，晶体管的"饱和"和"截止"等等，它们都可以用来表示一个二进制数；其中一个状态表示"0"，另一个状态表示"1"。如果要找出具有多个稳定状态的器件，比方说找一个有10个稳定状态的器件以便表示一个十进制数就比较困难了。同时，二进制的运算最简单。我们进行简单的算术运算时，必须要记住两个整数的和及乘积，如对十进制要记住 $\frac{10(10+1)}{2}=55$ 个和与积（九九表）才能运算。对 r 进制就要记住 $\frac{r(r+1)}{2}$ 个和与积。对于二进制，$r=2$，总共只要记得 3 个和与 3 个求积，即：

$$0+0=0 \qquad 0\times 0=0$$
$$1+0=0+1=1 \qquad 1\times 0=0\times 1=0$$
$$1+1=10 \qquad 1\times 1=1$$

因此二进制的运算器及其控制线路显然比其他进制要简单。总之，由于二进制易于用器件来表示，而且运算也简单，所以在计算机中得到广泛采用。

但是由于二进制数写起来很长，人们使用不习惯，往往只在机器内部使用，在输入和输出的地方要实行数制的转换，如在输入端要把十进制数转换成二进制数，输出端要把二进制数转换成十进制数。有时还要实现一些其他数制转换，下面就来讨论一下数制转换问题。

由 (1.1.1) 式可知，对 r 进制的具有 n 位整数和 m 位小数的 A_r，可表示为

$$A_r=a_{n-1}r^{n-1}+\cdots+a_1r^1+a_0r^0+a_{-1}r^{-1}+a_{-2}r^{-2}+\cdots+a_{-m}r^{-m}$$

当我们要把一个十进制数转换成 r 进制数时，主要就是要求出 $a_{n-2}, a_{n-2}\cdots a_{-(m-1)}, a_{-m}$ 等系数。为了计算方便，我们分别求整数部分的系数 $a_{n-1}, a_{n-2}, \cdots, a_0$ 和小数部分的系数 $a_{-1}, a_{-2}, \cdots, a_{-(m-1)}, a_{-m}$。

设 $A_{r'}$ 为 A_r 的整数部分。则：

$$\begin{aligned}A_{r'}&=a_{n-1}r^{n-1}+a_{n-2}r^{n-2}+\cdots+a_1r^1+a_0\\&=(a_{n-1}r^{n-2}+a_{n-2}r^{n-3}+\cdots+a_1)r+a_0\\&=[(a_{n-1}r^{n-3}+a_{n-2}r^{n-4}+\cdots+a_2)r+a_1]r+a_0\end{aligned}$$

显然 a_0 为 $A_{r'}/r$ 所得的余数，所得商数再除以 r 的余数就是 a_1，第二次得到的商再除以 r，余数是 a_2……，如此等等。

例如，一个整数 $N=(1895)_{10}$ 要转换为二进制，我们可以这样做：

2|1 8 9 5		
2|9 4 7	余 1	$a_0=1$
2|4 7 3	余 1	$a_1=1$
2|2 3 6	余 1	$a_2=1$
2|1 1 8	余 0	$a_3=0$
2|5 9	余 0	$a_4=0$
2|2 9	余 1	$a_5=1$
2|1 4	余 1	$a_6=1$
2|7	余 0	$a_7=0$
2|3	余 1	$a_8=1$
2|1	余 1	$a_9=1$
0	余 1	$a_{10}=1$

$(1895)_{10}=(11101100111)_2$ 这里下标 10，2 分别表示十进制、二进制等。整数十进制变换到二进制可以简称为“除 2 取余”。

再考虑小数部分的转换，设 $A_{r''}$ 为 Ar 的小数部分

$$\begin{aligned}A_{r''}&=a_{-1}r^{-1}+a_{-2}r^{-2}+\cdots+a_{-(m-1)}r^{-(m-1)}+a_{-m}r^{-m}\\&=r^{-1}(a_{-1}+a_{-2}r^{-1}+\cdots+a_{-(m-1)}r^{-(m-1)+1}+a_{-m}r^{-m+1})\\&=r^{-1}[a_{-1}+r^{-1}(a_{-2}+\cdots+a_{-(m-1)}r^{-(m-1)+2}+a_{-m}r^{-m+2})]\end{aligned}$$

只要把 A_r'' 乘以 r，所得整数部分就是 a_{-1} 余下的部分再乘以 r，得到的整数部分是 a_{-2}……，依此类推，可求出 a_{-1} 到 a_{-m} 的全部系数。

例如，有十进制小数 $W=(0.625)_{10}$，要将它化成二进制：

$$0.625\times 2=1.25 \qquad a_{-1}=1$$
$$0.25\times 2=0.5 \qquad a_{-2}=0$$
$$0.5\times 2=1.0 \qquad a_{-3}=1$$
$$W=(0.625)_{10}=(0.101)_2$$

小数十进制变换成二进制的算法可以归结为**"乘 2 取整"**。

如果一个数有整数和小数两部分，只要分别求出整数和小数部分的二进制表示，再合起来就可以了。例如：

$$(1895.625)_{10}=(11101100111.101)_2$$

当一个十进制数要转换成 r 进制的数时，只要把除 2、乘 2 改成除 r、乘 r 就可以了。

一个 r 进制表示的数 A 要转换成十进制，可以直接根据式(1.1.1)计算。

例如，将二进制数 $(11101100111.101)_2$ 化成十进制：

$$\begin{aligned}(11101100111.101)_2=&1\times 2^{10}+1\times 2^9+1\times 2^8+0\times 2^7+1\times 2^6\\&+1\times 2^5+0\times 2^4+0\times 2^3+1\times 2^2+1\times 2^1\\&+1\times 2^0+1\times 2^{-1}+0\times 2^{-2}+1\times 2^{-3}\\=&1024+512+256+64+32+4+2+1\\&+0.5+0.125\\=&1895.625\end{aligned}$$

也可以把式(1.1.1)改写为：

$$\begin{aligned}A_r=&a_{n-1}r^{n-1}+a_{n-2}r^{n-2}+\cdots+a_0r^0+a_{-1}r^{-1}+\cdots+a_{-m}r^{-m}\\=&\{[(a_{n-1}r+a_{n-2})r+a_{n-3}]r+\cdots+a_1\}r+a_0+r^{-1}\{a_{-1}\\&+r^{-1}[a_{-2}+\cdots+r^{-1}(a_{-(m-1)}+r^{-1}a_{-m})]\}\end{aligned}\qquad(1.1.2)$$

在利用计算机进行计算时，用式(1.1.2)比较方便。

至于两种不同数制 r 和 s 间的转换，例如要把 r 进制的数 N 转化成 s 进制，一般都是 $N_r\rightarrow N_{10}\rightarrow N_s$，即先把 r 进制的 N，化成十进制的

N，再把十进制的 N 化到 s 进制。

第二节　开关代数及其基本定理

开关代数即二元的布尔代数，是逻辑电路分析和设计中的一种基本数学工具。布尔代数是爱尔兰数学家 George Boole (1815-1864) 创立的。最初是用来研究思维规律的，是一种逻辑推理的代数方法。随着数学技术和计算机的发展，二元布尔代数在电路逻辑关系的分析和设计中得到了广泛的应用。因为布尔代数使人们可以用数学的形式来表达电路的逻辑关系并且简化它们。在人们发明了各种物理器件可以实现各种逻辑关系之后，布尔代数就可以和具体的电子线路对应起来，因此利用布尔代数就可以帮助人们分析和设计逻辑电路。

在二元布尔代数中只有两个元素："0" 和 "1" 组成集合。这里 0 和 1 只是一种符号的代表，失去了数量的含义，所以也可以用其他符号来代表，如用"假"和"真"、"高"和"低"等等。但是为了便于和常用代数比较，掌握布尔代数的运算规律，我们还是采用 0 和 1 作为代表。0 和 1 也可以对应于物理上的"开"和"关"两个状态，所以又称**开关代数**。开关**变量**是开关代数中定义的任意变量。显然，它也只有两种取值 0 和 1。

开关代数包括两种元素：0 和 1，以及三种运算：与、或、非，分别记为"·""+""-"。设 X_1, X_2 是两个开关变量，这三种运算定义如表 1.2.1。

表　1.2.1

开关变量		与	或	非	
X_1	X_2	$X_1 \cdot X_2$	$X_1 + X_2$	$\overline{X_1}$	$\overline{X_2}$
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	0	1
1	1	1	1	0	0

与、或是两个或两个以上开关变量之间的运算。对于与运算 $X_1 \cdot X_2$，只有当 X_1 和 X_2 都等于 1 时，$X_1 \cdot X_2$ 才为 1，X_1 和 X_2 中任意一个为 0，或者两个都为 0 时，$X_1 \cdot X_2$ 就是 0。如同乘法书写时可以把乘号省去一样，X_1 和 X_2 的与运算 $X_1 \cdot X_2$ 可以写成 X_1X_2。

对于或运算 X_1+X_2，只有当 X_1 和 X_2 都等于 0 时，X_1+X_2 才为 0，X_1 和 X_2 中任一个为 1 或者两者都为 1，X_1+X_2 就是 1。

非运算经常又称作**求反**，是对一个变量自身进行的运算。当 $X=0$ 时，$\overline{X}=1$；当 $\overline{X}=1$ 时，$\overline{X}=0$。

此外，因为与、或的书写符号和乘号、加号是相同的，习惯上也常常把与运算称作**逻辑乘**，把与运算的结果称作**积**。把或运算称作求**逻辑和**，把或运算的结果称为**和**等等。但是在开关代数中它们有明确意义，不同于普通代数中的求积求和。

为了应用开关代数，下面介绍开关代数中的基本定理。

（一）基本运算定理

定理 1 (a) $X\cdot 0=0$　开关变量和 0 相与，结果必定为 0。

(b) $X+1=1$　开关变量和 1 相或，结果必定是 1。

定理 2 (a) $X\cdot 1=X$　开关变量和 1 相与，结果为其本身。

(b) $X+0=X$　开关变量和 0 相或，结果为其本身。

定理 3 (a) $X\cdot X=X$　开关变量本身相与，结果为其本身。

定理 4 (a) $X\cdot\overline{X}=0$　开关变量和其反变量的与结果为 0。

(b) $X+\overline{X}=1$　开关变量和其反变量的或结果为 1。

定理 5 $\overline{\overline{X}}=X$　开关变量二次求反后仍为它本身。

这些定理给出了一个开关变量和常量 0、1 及它本身进行与、或、非运算的结果。这些定理的证明可以根据与、或、非运算的定义，分别设开关变量为 1 和 0，把它们代入式中，验证在各种情况下等式二边始终相等。同时可以注意到，这些基本定理都是成对地出现，1 和 0 对应，“与”和“或”对应。

（二）交换律、结合律和分配律

定理 6 (a) $X_1\cdot X_2=X_2\cdot X_1$　与运算的交换律。

(b) $X_1+X_2=X_2+X_1$　或运算的交换律。

定理 7 (a) $(X_1+X_2)+X_3=X_1+(X_2+X_3)$　或运算结合律

(b) $(X_1\cdot X_2)\cdot X_3=X_1\cdot(X_2\cdot X_3)$　与运算结合律。

定理 8 (a) $X_1\cdot(X_2+X_3)=X_1X_2+X_1X_3$　与对或的分配律。

即开关变量 X_1 对开关变量 X_2 和 X_3 的或求与，可以先对 X_2 和 X_3 分别求与，然后把二项结果相或。这和普通代数中乘对加的分配律是相似的。

(b) $X_1+(X_2\cdot X_3)=(X_1+X_2)\cdot(X_1+X_3)$ 或对与的分配律。这和普通代数是不同的，开关代数中还有加对乘的分配律，即开关变量 X_1 对 X_2 和 X_3 的与求或，可以通过 X_1 先对 X_2、X_3 分别求或，然后相与得到。

定理 8(a) 以前的几条定理都比较简单，仍可以用设开关变量分别为 0、1 代入验证。从定理 8(b) 开始，我们利用前面的定理来证明新的定理。

对定理 8(b)，左边为

$$
\begin{aligned}
&(X_1+X_2)\cdot(X_1+X_3)\\
&=(X_1+X_2)\cdot X_1+(X_1+X_2)\cdot X_3 && \text{用定理 8(a)}\\
&=X_1\cdot X_1+X_1\cdot X_2+X_1\cdot X_3+X_2\cdot X_3 && \text{同上}\\
&=X_1+X_1\cdot X_2+X_1\cdot X_3+X_2\cdot X_3 && \text{定理 3(a)}\\
&=X_1\cdot(1+X_2+X_3)+X_2\cdot X_3 && \text{定理 2(a)和定理 8(a)}\\
&=X_1+X_2\cdot X_3 && \text{定理 1(b)}
\end{aligned}
$$

定理 9 (a) $X_1\cdot X_2+X_1\cdot\overline{X}_2=X_1$

证明：$X_1\cdot X_2+X_1\cdot\overline{X}_2=X_1\cdot(X_2+\overline{X}_2)=X_1$

(b) $(X_1+X_2)\cdot(X_1+\overline{X}_2)=X_1$

证明：$(X_1+X_2)\cdot(X_1+\overline{X}_2)=X_1+(X_2\cdot\overline{X}_2)=X_1$

定理 9 是定理 8 当 $X_3=\overline{X}_2$ 时的一个特殊情况，用这个定理可以消去 X_2 变量，使表达式简化，是化简逻辑电路所依据的基本定理之一。

(三) 吸收定律

定理 10 (a) $X_1+X_1\cdot X_2=X_1$

证明：$X_1+X_1\cdot X_2=X_1\cdot(1+X_2)=X_1$

该定理说明在一个先求与再求或的“与或”开关代数表达式中，如果一个乘积项是另一个乘积项的因子，则包含这个因子的乘积项是多余的。在化简时可以把它去掉，或者说吸收掉。

(b) $X_1(X_1+X_2)=X_1$

证明：$X_1(X_1+X_2)=X_1\cdot X_1+X_1\cdot X_2=X_1+X_1\cdot X_2=X_1$

和定理10(a)相对应，在一个先求或再求与的“或与”开关代数表达式中，如果一个求和项是另一个求和项的一个加数，则包含这个加数的求和项是多余的，可以去掉。

定理11 (a) $X_1+\overline{X}_1\cdot X_2=X_1+X_2$

证明：$X_1+\overline{X}_1\cdot X_2=X_1+\overline{X}_1\cdot X_2+X_1\cdot X_2$

$=X_1+(\overline{X}_1+X_1)\cdot X_2$

$=X_1+X_2$

它说明在“与或”表达式中，如果一个乘积项的反变量是另一个乘积项的因子，则这个因子是多余的，可以去掉。

(b) $X_1\cdot(\overline{X}_1+X_2)=X_1\cdot X_2$

证明：$X_1\cdot(\overline{X}_1+X_2)=X_1\cdot\overline{X}_1+X_1\cdot X_2=X_1\cdot X_2$

在“或与”表达式中，如果一个求和项的反变量是另一个求和项的加数，则这个加数是多余的，可以去掉。

定理12 (a) $X_1\cdot X_2+\overline{X}_1\cdot X_3+X_2\cdot X_3=X_1\cdot X_2+\overline{X}_1\cdot X_3$

证明：$X_1\cdot X_2+\overline{X}_1\cdot X_3+X_2\cdot X_3$

$=X_1\cdot X_2+\overline{X}_1\cdot X_3+X_2\cdot X_3(X_1+\overline{X}_1)$

$=X_1\cdot X_2+X_1\cdot X_2\cdot X_3+\overline{X}_1\cdot X_3+\overline{X}_1\cdot X_2\cdot X_3$

$=X_1\cdot X_2+\overline{X}_1\cdot X_3$

它说明在一个“与式”表达式中，如果两个乘积项中一个包含了原变量X_1，另一个项包含了反变量$\overline{X}_1$，而这二项的其余因子都是第三个乘积项的因子，则第三个乘积项是多余的，可以去掉。

(b) $(X_1+X_2)\cdot(\overline{X}_2+X_3)\cdot(X_1+X_3)$

$=(X_1+X_2)\cdot(\overline{X}_2+X_3)$

证明：$(X_1+X_2)\cdot(\overline{X}_2+X_3)\cdot(X_1+X_3)$

$=(X_1+X_2)\cdot(\overline{X}_2+X_3)\cdot(X_1+X_3+X_2\cdot\overline{X}_2)$

$=(X_1+X_2)\cdot(\overline{X}_2+X_3)\cdot(X_1+X_3+X_2)\cdot(X_1+X_3+\overline{X}_2)$

$=(X_1+X_2)\cdot(\overline{X}_2+X_3))$

在"或与"表达式中，如果两个求和项中一个包含了原变量 X_2，另一项包含了反变量 $\overline{X}_2$，而这二项其他的加数 X_1、X_3 都是第三个求和项的加数，则第三个求和项是多余的，可以去掉。

定理 10～12 都是可以去掉某些项，或者说吸收某些项，使开关代数表达式简化。因此这些定理在化简逻辑电路时很有用。

(四)反演定律

定理 13 (a) $\overline{X_1 \cdot X_2} = \overline{X}_1 + \overline{X}_2$

(b) $\overline{X_1 + X_2} = \overline{X}_1 \cdot \overline{X}_2$

定理 13 又称作**摩根定理**，即对两个变量的与求反可以通过对每个变量分别求反再对这二项求或得到；对两个变量的或求反可以通过对每个变量分别求反，再对这二项求与得到。

它的证明可以设开关变量 X_1、X_2 为各种值，代入验证。见表 1.2.2

表 1.2.2

X_1	X_2	$X_1 \cdot X_2$	$X_1 + X_2$	$\overline{X_1 \cdot X_2}$	$\overline{X_1 + X_2}$	$\overline{X}_1$	$\overline{X}_2$	$\overline{X}_1 + \overline{X}_2$	$\overline{X}_1 \cdot \overline{X}_2$
0	0	0	0	1	1	1	1	1	1
0	1	0	1	1	0	1	0	1	0
1	0	0	1	1	0	0	1	1	0
1	1	1	1	0	0	0	0	0	0

此定理可以推广到几个开关变量

定理 14 (a) $\overline{X_1 \cdot X_2 \cdot \cdots \cdot X_n} = \overline{X}_1 + \overline{X}_2 + \cdots + \overline{X}_n$

(b) $\overline{X_1 + X_2 + \cdots + X_n} = \overline{X}_1 \cdot \overline{X}_2 \cdot \cdots \cdot \overline{X}_n$

此定理可以反复应用定理 13 证明。例如把 $\overline{X_1 \cdot X_2 \cdot \cdots \cdot X_n}$ 可以看成是二个变量 X_1 和 $(X_2 \cdot X_3 \cdots X_n)$ 的乘积

$$\overline{X_1 \cdot (X_2 \cdot \cdots \cdot X_n)} = \overline{X}_1 + \overline{X_2 \cdot X_3 \cdot \cdots \cdot X_n}$$

再把后面这一项看成是二个变量的乘积，如此分解下去。

反演定律使我们可以用与运算代替或运算，反之亦然。这在具体构成电路时往往很有用。以后可以看到，根据这些定理，甚至能用一种形式的电路来实现各种逻辑功能。