数字逻辑设计基础

上册

罗朝杰编著

数 字 逻 辑 设 计 基 础

上 册

罗朝杰 编著
王蔚亢 校

人 民 邮 电 出 版 社

内 容 提 要

本书系统地介绍了硬件逻辑设计的基础理论和设计方法。全书分上下两册。上册包括一至五章，主要讨论逻辑代数和各种组合逻辑电路，内容有：逻辑代数基础，算术运算基础，门电路，组合逻辑设计基础，组合逻辑单元电路设计。下册包括六至十章，主要讨论开关理论和各种时序逻辑电路，内容有：触发器设计，时序逻辑设计基础，时序逻辑部件设计，半导体存贮器，数字系统设计。

本书重点介绍利用中小规模集成电路进行逻辑设计的原理和方法，关于大规模集成电路（如*RAM*和*ROM*）的应用也有所介绍。

本书可供从事数字技术的工程技术人员及大专院校有关专业师生参考。

数字逻辑设计基础
上　册
罗朝杰　编著
王蔚亢　校
•
人民邮电出版社出版
北京东长安街27号
河北省邮电印刷厂印刷
新华书店北京发行所发行
各地新华书店经售
•
开本：787×1092 1/16　1982年9月 第一版
印张：18 12/16 页数：150　1982年9月河北第一次印刷
字数：469 千字　印数：1—12,500册
统一书号：15045·总2616—有5259
定　价：2.05元

前　言

随着计算技术和数字技术的发展，逻辑设计出现了两个明显的分枝，即硬件逻辑设计和软件逻辑设计。软件逻辑是由硬件支持下的程序设计来实现的。近几年来，软件逻辑的研究和应用发展很快，可以想象得到，在计算机和微处理机十分普及的时候，软件逻辑将是进行逻辑设计的主要手段。不过，即便到了那时，硬件逻辑设计也是不可缺少的，这是因为硬件逻辑的操作速度快，灵活性较大，对于规模较小的系统来说，硬件逻辑的成本低得多。就目前条件而论，硬件逻辑设计在数字技术中还占着相当重要的地位，熟练掌握硬件逻辑设计的理论和技巧，对一个从事硬件设计的人员来说是十分必要的。

本书系统地介绍了硬件逻辑设计的基础理论和设计方法，主要讨论如何使用中小规模集成电路进行逻辑设计，至于大规模集成电路的应用也作了一般的介绍。

布尔代数、近代开关理论和离散数学，是进行逻辑设计的理论基础。在本书的第一、四、七章中，详细讨论了上述学科中与逻辑设计直接有关的部分内容。可以认为，这些内容是一个专业设计人员必备的最低限度的理论。为使理论分析和工程实际相结合，我们尽可能从工程应用的角度来讨论这些纯理论问题，并使之自成体系。这样做虽然会失去理论上的严密性，但却因为有了明确的物理概念而使读者更容易理解和接受。

近代数字工程中，将会用到各种各样的单元电路，新电路也在不断出现，由于篇幅所限，不可能对它们一一进行讨论。本书第五、八、九章中，仅对一些常用的，具有代表性的逻辑电路作了详细的讨论。若能掌握这几章中所涉及到的单元电路，设计一个一般用途的数字设备是不太困难的。此外，在第十章中，还对系统级逻辑设计所存在的问题，进行了深入的讨论。

第三章是专门为初学者编写的，它不属于逻辑设计的范围，对于已有门电路知识或只对逻辑设计感兴趣的读者，可以跳过不读。

近几年来，我国各大专院校已纷纷开设“数字电路和逻辑设计”这门课程。为了弥补教材在广度和深度方面的不足，作者根据教学实践和科研工作的需要，参阅了国内外大量图书资料，编写了这本介绍硬件逻辑设计的参考书，仅供从事硬件逻辑设计的科技人员和大专院校师生参阅。

本书在内容叙述上力求深入浅出，通俗易懂，使初学者便于自学。在讨论设计方法时，力求物理概念清楚，设计步骤明确，使读者能够自行设计出所需的单元电路和数字系统。除第三章外，各章均附有思考题和习题。

在本书的编写过程中，曾得到西北电讯工程学院和开封市科学技术委员会等单位许多同志的帮助，特别是董有喆、赖茂宏、梅焕文和宫学昌等同志对原稿提出过很多宝贵意见，北京邮电学院王蔚亢同志审阅了本书的原稿，并提出许多修改意见，参加审阅的还有西北电讯工程学院江小安同志，谨借此一并表示衷心的感谢。

对于书中存在的错误或不妥之处，希望读者不吝指正。

罗朝杰

1981.5

目　录

第一章　逻辑代数基础

逻辑代数是研究数字逻辑设计的基础理论。

电子数字计算机和其他数字设备中，大量使用了各种各样的数字逻辑电路。数字逻辑电路的种类繁多，功能各异，有的简单，有的复杂，而且新的电路形式层出不穷。为了弄清它们的个性和共性，熟悉它们的原理和性能，必须掌握逻辑代数这一数学工具。

这一章主要介绍逻辑代数的基本概念和运算规律，以及如何用它来化简逻辑函数。此外，还要介绍一种工程应用上十分有用的逻辑工具——卡诺图。

第一节　逻辑代数和集合代数

在这一节里，我们从数字工程应用的角度来建立一个逻辑代数系统，但是，我们并不想如某些专著那样去论述公理系统的完备性、独立性和不矛盾性，仅从实际应用的需要来确立一套基本定律，提出几个重要定理和运算法则。

一、基本概念

1. 逻辑代数的发展概况

逻辑代数最早叫做数理逻辑，它是从哲学领域中的逻辑学发展而来的。逻辑学是研究逻辑思维和推理的一门学科，为了摆脱冗繁的文字描述，人们使用了一套有效的符号来建立逻辑思维的数学模型，进而将复杂的逻辑问题抽象为一种简单的符号演算。这种逻辑思维和数学符号的巧妙结合，产生了“数理逻辑”这个新的概念。

“数理逻辑”概念是莱布尼兹（*Leibniz*）首先提出来的，乔治·布尔（*G·Boole*）总结了前人的研究成果，于1847年在他的著作中第一次加以系统的论述，可是，直到20世纪初才发展成一门纯数学的分枝学科。“布尔代数”这个名称，就是以布尔的名字命名的。

布尔代数进一步向电子工程领域延伸的结果，便产生了开关代数。“开关代数”是仙农（*Shannon*）1938年提出来的，他运用布尔代数分析开关电路问题时，发现任何由继电器触点组成的开关电路，都能用布尔代数式来进行描述和设计。这给布尔代数找到了一个极其广阔的用武之地，并加速了计算机时代的到来。1946年第一台电子数字计算机的问世，显示出开关代数的重要性，也是布尔代数发展一百年的一个总结。

1952年前后，维奇（*Veitch*）和卡诺（*Karnaugh*）相继提出图解法的概念，丰富和发展了布尔代数。与代数法相比，图解法具有简便、直观和快速等优点，特别是对于解决一些复杂的时序逻辑问题，图解法更显示了它的优越性。

目前，集成电路逻辑门已经取代了机械触点开关，“开关代数”这个术语已较少使用了，为了与术语“数字逻辑设计”相适应，我们把布尔代数叫做逻辑代数。逻辑代数是一种

二值代数系统，随着计算技术的发展，它已经不能满足新的要求了。为了补充它的不足，又提出了一些新的课题，如程序逻辑，多值逻辑，阀值逻辑等，但它们已经超出了本书的范围，本书不再进行讨论了。

2. 逻辑变量

在数理逻辑中，逻辑变量叫做逻辑命题，简称命题。所谓命题，系指一切能够辨其真伪的语句，如

太阳从东方升起来。

雪是黑的。

第一句表达的是科学真理，叫做真命题。第二句显然是错误的，叫做假命题。命题语句必须是陈述句，而且必须满足排中律。排中律指出，每个命题的含义必须或真或假，二者必居其一。任何在含义上似是而非或模棱两可的陈述句，都不能作为命题。

在数字工程领域内，研究对象是具体器件的状态，故将“命题”这一术语改为逻辑变量。在数字电路中，凡是能用两种截然不同的状态来描述的电子器件，都可作为逻辑变量来考虑，如机械开关、传输线、指示灯、晶体管或触发器等等。严格说来，应当象命题语句那样来描述一个逻辑变量，例如

信号灯A是亮着的。

传输线B处于高电平。

为了方便起见，以后一律用器件的名称来定义一个逻辑变量，并用大写字母A，B，C，D等来代表它们，简称“变量A”“变量B”等等。

由于排中律的限定，一个逻辑命题只能有两种逻辑值，即逻辑真和逻辑假，简称真值和假值。布尔代数选择了两个最简单的符号0和1来代表这两种逻辑值，令0代表假值，令1代表真值。逻辑变量也用0和1这两个符号来表示器件的两种状态，例如，令$A=1$表示信号灯是亮着的，则$A=0$就表示灯已灭；又如令$B=1$表示传输线处于高电平，则$B=0$表示它处于低电平。显然，似亮非亮的中间状态，不高不低的中间电平，不能用来表示变量的逻辑值。若器件一旦处于这种中间状态或中间电平，则以出现逻辑错误处理。

注意，0和1仅被定义为两种逻辑值，并未赋予它们任何数量的概念，应将它们与自然数0和1区别开。逻辑值0和1仅是两种用来判断真伪的形式符号，无大小、正负之分。

3. 逻辑函数

在数理逻辑中，用联结词将若干基本命题组合成的新命题，称为基本命题的逻辑函数，简称函数。广义说来，任何两个相关的逻辑变量之间，都存在着函数关系。

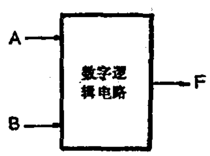

图 1-1 数字逻辑系统

如图1-1所示的数字逻辑系统，输入信号A和B如果是逻辑变量，则输出信号F便是逻辑函数，并将它表示为

$$F=f(A, B)$$

实际上，逻辑函数是描述一个控制过程的数学模型，它能完全表征一个逻辑系统的全部属性和功能。逻辑电路和逻辑函数具有严格的对应关系，这使我们有可能将一个具体的逻辑电路变换为抽象的代数表达式，再对它进行分析研究和加工处理就方便多了。

逻辑函数也只有两种逻辑值 0 和 1，因此它又可以作为逻辑变量去控制别的逻辑电路。

逻辑函数的性质、逻辑函数的化简、逻辑函数的变换、逻辑函数的实现等等，是逻辑设计要讨论的一些重要课题。

4. 真值表

与普通代数的变量相比，逻辑变量是十分简单的，因为它只能有两种取值 0 或 1。对于多变量问题来说，逻辑函数同时受多个变量的控制，一个变量有两种取值的可能，则 n 变量将有 2^n 种不同的取值组合。

二变量有四种不同的取值组合：

00，01，10，11

三变量有八种不同的取值组合：

000，001，010，011，100，101，110，111

表 1-1(a) 真值表格式

ABC	$f(A,B,C)$
0 0 0	a_0
0 0 1	a_1
0 1 0	a_2
0 1 1	a_3
1 0 0	a_4
1 0 1	a_5
1 1 0	a_6
1 1 1	a_7

表 1-1(b) 真值表实例

ABC	F
0 0 0	0
0 0 1	1
0 1 0	1
0 1 1	0
1 0 0	1
1 0 1	1
1 1 0	1
1 1 1	0

一般逻辑问题的变量数目不超过五个，因此，变量的取值组合的总数是有限的，允许我们采用逐点观察法来描述逻辑函数的功能，并将结果用表格形式记录下来，这个用真假值记录的表格叫做真值表。

表1-1(a)是真值表的格式，它由两部分组成。左边一栏列出了变量的所有组合，为了不致漏掉一种组合，各变量的取值按二进制编码形式给出（见第二章第一节），即第一变量(C)的 0 和 1 隔行相间，第二变量(B)的 0 和 1 隔两行相间，第三变量(A)的 0 和 1 隔四行相间，余以此类推。右边一栏为逻辑函数的值，在具体实例中，各种 a_i 为具体的值 0 或 1。表1-1(b)是某个逻辑问题的真值表实例，该问题的功能由 F 列中的 0、1 结构形式给出。例如第一行记录了这样一个事实，即当所有输入变量为 0 时，输出 F 也为 0；又如第二行指出，当 $A=B=0$ 而 $C=1$ 时，输出 $F=1$。

真值表详尽地记录了逻辑问题的功能，是一种十分有用的逻辑工具，以后要经常用到它。

二、基 本 运 算

虽然实际应用中可能遇到的逻辑问题是千变万化的，但万变不离其宗，它们都可以用三种基本的逻辑运算综合而成。这三种基本的逻辑运算是：

逻辑非——逻辑否定，用“$\overline{A}$”号表示；

逻辑加——加法运算，用“＋”号表示；

逻辑乘——乘法运算，用“×”号表示。

下面分别讨论这三种运算的定义和性质。

1. 逻辑非

对一个变量进行逻辑否定，称为逻辑非运算，它的数学表达式为

$$F=\overline{A}$$

“$\overline{A}$”读作“A非”或“非A”。变量 A 上面的短横线是表示逻辑非的运算符号。我们将 A 叫做原变量，$\overline{A}$ 叫做反变量，A 和 $\overline{A}$ 是一个变量的两种形式。

逻辑非是逻辑代数所特有的一种运算形式。实际上，逻辑非正是排中律的数学模型，它

在逻辑代数中起着相当重要的作用。逻辑非的功能是对变量求反（有时称为求补），不管原变量为何值，进行逻辑非运算之后必得其相反之值，如

$$A=0\text{时}，\overline{A}=1$$
$$A=1\text{时}，\overline{A}=0$$

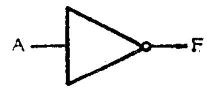

图 1-2 倒相器

由于 0 和 1 意义相反，下列等式必然成立，

$$\overline{1}=0，\overline{0}=1$$

连续进行两次逻辑非运算，相当于没有进行任何运算，即

$$\overline{\overline{A}}=A$$

“$\overline{\overline{A}}$”读作“A非非”，它所表征的正是“否定之否定等于肯定”这一规律。

基本逻辑电路和门的开关作用极为相似，用来控制数字信息的流通，这类逻辑电路统称为门电路或逻辑门（*gate*）。实现逻辑非功能的电路，称为倒相器（反相器）或非门，它的逻辑符号如图1-2所示。

2. 逻辑加

在数理逻辑中，前提和结论之间常常存在着一种简单的**或**关系，只要有一个前提条件具备了，结论就能成立。这种对前提条件的非限定选择，即这个**或**那个，一个**或**多个前提条件具备了都可以，我们把这种逻辑关系称为**或逻辑**。在数字电路中，或逻辑可解释成某种控制过程，参与控制的诸因素中，只要有一个条件具备了就能实现这种控制。

或逻辑又叫做**逻辑加**，两变量的逻辑加可表示如下：

$$F=A+B$$

“$A+B$”读作“A加B”，也可读作“A或B”。式中的“+”是逻辑加的运算符号（有些文献中使用符号“V”或“U”），它仅表示“或”的逻辑功能，没有数量累加概念。

表 1-2

A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

实现逻辑加运算的器件称为或门，它的逻辑符号如图1-3所示。表1-2是或门的真值表，按或逻辑的定义，输入端只要有一个为 1，输出F便为 1，输入端全为 0 时，输出F才为 0，从真值表可得到下列三个等式：*

图 1-3 或门

$$0+0=0$$
$$0+1=1$$
$$1+1=1$$

前两式不难理解，唯最后一式与普通代数不同。1 + 1 = 1，正好说明逻辑加没有数量累加作用。它表示一种逻辑判断，说明对于或逻辑来说，具备多个前提条件与具备一个前提条件所得的结论是一样的。

变量与常量的逻辑加，结果如下：

$$A+0=A$$
$$A+1=1$$

由于变量A只有 0 和 1 两种取值，上式的正确性不难知道，把A分别用 0 和 1 代入即得到验证。

同样可以证明一个变量自加多次仍为原变量，

$$A+A=A$$

*由于逻辑代数满足交换律，因此 1 + 0 = 1 和 0 + 1 = 1 可作为一个等式来看待。

$$A+A+A=A$$

它们说明，在或逻辑中，若诸因素相同，只要一个就够了。

图1-4是两个或逻辑的实例。图1-4中两个开关并联控制信号灯L，任何一个开关A或B闭合，L就会亮；当然，两个开关同时闭合，L也会亮。因此，并联开关所实现的是或逻辑。

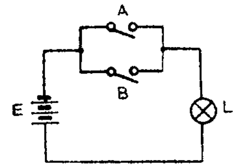

图 1-4 或逻辑实例

3. 逻辑乘

在数理逻辑中，前提与结论之间的兼备关系，称为**与逻辑**。与逻辑指出，必须所有前提条件同时具备，结论方能成立。在控制逻辑中，必须全部条件都具备时才能实现此种控制，也是一种**与逻辑**。

与逻辑又叫逻辑乘，二变量的逻辑乘可表示如下：

$$F=A\times B$$

“$A\times B$”读作“A乘B”或“A与B。式中“×”是逻辑乘的运算符号（有些文献中使用符号“∧”或“∩”），它仅表示“与”的逻辑功能，无数量相乘的概念。我们约定，“×”可用“·”代替，或者把“·”也省掉，如

$$F=A\times B=A\cdot B=AB$$

实现逻辑乘的器件叫做与门，它的逻辑符号如图1-5所示。表1-3是与门的真值表，它和与逻辑的定义完全符合，即输入端只要有一个为0，输出F就是0，全部输入端为1时，输出F才是1。

图 1-5 与门

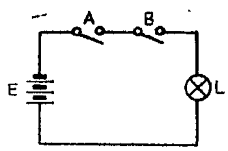

图 1-6 与逻辑实例

表 1-3

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

从真值表可得到下列三个等式：

$$0\times 0=0$$
$$0\times 1=0$$
$$1\times 1=1$$

变量与常量相乘，或者变量多次自乘，结果如下：

$$A\cdot 0=0$$
$$A\cdot 1=A$$
$$A\cdot A=A$$
$$A\cdot A\cdot A=A$$

这些等式都是从定义引出的，它们的正确性无须再作证明。

图1-6的串联开关是与逻辑的一个实例，只有当A和B都接通时，信号灯L才会发亮。

4. 逻辑代数的对偶性

到此为止，我们已定义了五种逻辑符号：

$$0,\ 1,\ \overline{A},\ +,\ \times$$

整个代数系统正是用这五种符号建立起来的（还包括等号和括号）。但是可以证明，由于逻辑非的存在，建立这个代数系统只要三种符号就够了。例如，选择下列三个符号

$$0,\ \overline{A},\ \times$$

来建立这个代数系统时，1 可用$\overline{0}$表示，"+"可用"$\overline{\times}$"表示（非逻辑乘必为逻辑加）。这样仍可得到五个定义所需要的符号，即

$$0,\ \overline{0},\ \overline{A},\ \overline{\times},\ \times$$

的确有人曾经建议使用三种符号的系统，但并未被人们所接受，因为这会增加理解上的负担。现代逻辑代数系统，仍然使用上述五种符号。

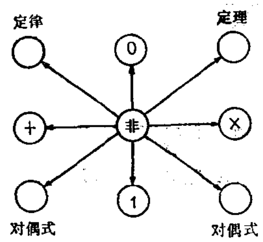

图 1-7 逻辑代数的对偶性

如图 1-7 所示，逻辑代数系统具有很好的**对偶性**，对称中心是逻辑非。任何符号、定律和定理都是成对出现的，通过对称中心联系起来，形成逻辑上的对偶。图1-7 指出，0 和 1 是两种对偶符号，"+"和"×"是两种对偶运算，每个定律和定理都有它的对偶式。

顺便指出，由于"+"和"×"是一对平等的符号，不存在"先乘后加"的问题。为了少用括号，我们人为地约定，无括号时一律按"先乘后加"的顺序进行运算。

三、基 本 定 律

为了证明定理和推导公式方便起见，必须确定一套基本定律（在经典著作中称为公理系统）。除了交换律和结合律外，我们确立下列四个基本定律：

吸收律 $\begin{cases} 1+A=1 \\ 0\cdot A=0 \end{cases}$

求补律 $\begin{cases} A+\overline{A}=1 \\ A\cdot\overline{A}=0 \end{cases}$

等幂律 $\begin{cases} A+A=A \\ A\cdot A=A \end{cases}$

分配律 $\begin{cases} A\cdot(B+C)=AB+AC \\ A+BC=(A+B)(A+C) \end{cases}$

每个定律包括两个公式，它们互为对偶式(关于对偶式的定义和求法，后面还要介绍)。吸收律、求补律和等幂律，是从"与、或、非"的定义引出来的，无须再去证明它们，下面仅对它们作些补充说明。

吸收律的两个基本公式为

$$1+A=1,\qquad 0\cdot A=0$$

它们还可以推广到一般形式

$$1+f(A,B,C,\cdots\cdots)=1$$
$$0\cdot f(A,B,C,\cdots\cdots)=0$$

吸收律指出："1加任何数得1，0乘任何数得0。"这里所说的"数"当然是逻辑变量或逻辑函数。换句话说，任何数与1相加或与0相乘时，均被1和0所吸收。

求补律的两个基本公式为

$$A+\overline{A}=1, \qquad A\cdot\overline{A}=0$$

若另一变量$B=\overline{A}$，则说这两变量A和B互补。两个互补的变量不是彼此独立的，实际上是同一变量的两种形式。求补律也适合于两个互补的逻辑函数。求补律指出："若两个变量互补，其和为1，其积为0。"求补律常用来判断两变量或两函数是否互补。

等幂律的两个基本公式为

$$A+A=A, \qquad A\cdot A=A$$

等幂律指出："一个变量多次自加或多次自乘的结果仍为原变量。"这说明逻辑代数不存在倍乘和方幂运算，就是说，$A+A\neq 2A$，$A\cdot A\neq A^2$。

分配律的两个基本公式为

$$A(B+C)=AB+AC$$

$$A+BC=(A+B)(A+C)$$

第一式称为第一分配律，是乘法对加法实行分配，与普通代数一样。现用真值表法来证明它，为此，分别将等号两边的部分作为不同函数处理，

左边：$F=A(B+C)$

右边：$G=AB+AC$

函数F的真值表如表1-4(a)所示，函数G的真值表如表1-4(b)所示，表中还列出了它们的子函数。从这两个真值表可以看出$F=G$，说明第一分配律成立。

表 1-4(a)　F的真值表

A	B	C	$(B+C)$	F
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

表 1-4(b)　G的真值表

A	B	C	AB	AC	G
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

第二式称为第二分配律，是加法对乘法实行分配，普通代数中没有这条分配律，故又称为特种分配律。证明特种分配律的真值表如表1-5所示。

分配律进一步说明加法和乘法是平等的，乘法可以对加法实行分配，加法也可以对乘法实行分配。

这些基本定律得到确认之后，就可以用它们来证明定理和化简逻辑函数。

四、基　本　定　理

仅有一些基本定律是不够的，为了简捷地证明问题或推导公式，还必须提出若干基本定理。使用定理来证明问题可以省时省事，但要求人们熟记这些定理。为了减少记忆的负担，

表 1-5　　证明特种分配律的真值表

A	B	C	BC,	$A+BC$	$(A+B)$,	$(A+C)$,	$(A+B)(A+C)$
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

$A+BC=(A+B)(A+C)$

提出的定理不能太多，要求它们既简单又重要。

逻辑代数中常用的基本定理如下：

吸收定理(Ⅰ) $\begin{cases}AB+A\overline{B}=A\\(A+B)(A+\overline{B})=A\end{cases}$

吸收定理(Ⅱ) $\begin{cases}A+AB=A\\A(A+B)=A\end{cases}$

吸收定理(Ⅲ) $\begin{cases}A+\overline{A}B=A+B\\A(\overline{A}+B)=AB\end{cases}$

摩根定理 $\begin{cases}\overline{A+B}=\overline{A}\,\overline{B}\\\overline{AB}=\overline{A}+\overline{B}\end{cases}$

多余项定理 $\begin{cases}AB+\overline{A}C+BC=AB+\overline{A}C\\(A+B)(\overline{A}+C)(B+C)=(A+B)(\overline{A}+C)\end{cases}$

每个定理包括两个公式，它们互为对偶式。以上提出的五个定理，必须证明之后才能使用，下面就来逐个地证明它们。不过，我们只证明每个定理的第一式，根据对偶原理（后面还要介绍），若第一式成立，第二式必然成立。

1. 吸收定理

吸收定理(Ⅰ)：$AB+A\overline{B}=A$

【证】 左边$=AB+A\overline{B}$

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

$=A$　　(证毕)

吸收定理(Ⅱ)：$A+AB=A$

【证】 左边$=A+AB$

$=A(1+B)$　$(1+B=1)$

$=A$　　(证毕)

吸收定理(Ⅲ)：$A+\overline{A}B=A+B$

【证】 左边$=A+\overline{A}B$　　(分配律)

$=(A+\overline{A})(A+B) \quad (A+\overline{A}=1)$

$=A+B \quad$ (证毕)

应用吸收定理(Ⅰ)和(Ⅱ)，可将两项合并为一项并消去一个变量；应用吸收定理(Ⅲ)，可将某些项中的部分因子消去。以上三个定理在化简函数时是很有用的，由于它们都能消去某个变量或因子，故将它们统称为吸收定理。

2. 摩根定理

二变量的摩根定理为

$$\begin{cases}\overline{A+B}=\overline{A}\,\overline{B}\\ \overline{AB}=\overline{A}+\overline{B}\end{cases}$$

现用求补律证明摩根定理。求补律指出，“若二变量之和为1，其积为0，则它们必然互补。”

【证】 第一式 $\overline{A+B}=\overline{A}\cdot\overline{B}$

令$F=A+B$，$G=\overline{A}\,\overline{B}$，然后将$F$和$G$进行相加和相乘，看其是否互补。

$$\begin{aligned}F+G&=A+B+\overline{A}\,\overline{B} && (\text{吸收定理Ⅲ消去因子}\overline{A})\\ &=A+B+\overline{B} && (B+\overline{B}=1)\\ &=A+1 && (A+1=1)\\ &=1\end{aligned}$$

$$\begin{aligned}F\cdot G&=(A+B)\overline{A}\,\overline{B}\\ &=A\overline{A}\overline{B}+B\overline{A}\overline{B} && (A\overline{A}=B\overline{B}=0)\\ &=0\end{aligned}$$

由于F和G相加为1而相乘为0，则知F和G互补，即$\overline{F}=G$，将F和G代入便得到摩根定理第一式，由此得证。摩根定理第二式也可用同样方式证明。

在逻辑代数中、摩根定理是一个十分重要的定理，它解决了函数求反问题和逻辑变换问题。

3. 多余项定理

多余项定理的两个公式为

$$\begin{cases}AB+\overline{A}C+BC=AB+\overline{A}C\\ (A+B)(\overline{A}+C)(B+C)=(A+B)(\overline{A}+C)\end{cases}$$

这个定理的证明方法是从等式左边推导到右边。

【证】 $AB+\overline{A}C+BC=AB+\overline{A}C$

$$\begin{aligned}\text{左边}&=AB+\overline{A}C+BC\\ &=AB+\overline{A}C+(A+\overline{A})BC\\ &=AB+ABC+\overline{A}C+\overline{A}BC\\ &=AB(1+C)+\overline{A}C(1+B)\\ &=AB+\overline{A}C\end{aligned}$$

多余项定理（第一式）指出，**当某变量（如A）分别以不同形式出现在两项中时，这两项的其余因子组成的第三项（如BC）必为多余项**，可以略去。

若多余项中还有别的因子，它仍然是多余项。因此，多余项定理（第一式）可以推广到