第 3 章

CHAPTER 3

逻辑代数基础

逻辑代数是布尔代数的一种特例，是分析和设计开关电路的重要数学工具。逻辑代数是研究数字系统逻辑设计的基础理论。本章从应用的角度，主要介绍逻辑代数的基本概念、基本公式和规则，逻辑函数的表示形式、转换与化简。

3.1 逻辑代数的基本概念

作为描述与刻画客观世界的一种数学工具，逻辑代数面对的是离散的数字信号。

“逻辑”一词来自逻辑学，是研究逻辑思维与逻辑推理规律的，它表示事物发生的条件和结果之间的规律，即一种因果关系。在数字逻辑系统中，用 0 和 1 这两个二进制数描述两种相互对立的状态，这种表示方式中不存在中间状态。若我们把条件看作逻辑变量，把结果看作逻辑函数，而逻辑变量和逻辑函数的取值限定在 0 和 1，这样就把一个逻辑问题转化为一个代数问题。这种用代数的方法去研究逻辑问题的科学称为逻辑代数。

虽然有些逻辑代数的运算公式在形式上和普通代数的运算公式雷同，但两者所表示的物理意义有本质区别。逻辑运算表示的是逻辑变量及常量之间逻辑状态的推理运算，而不是数量间的运算。虽然一个逻辑变量的取值只有 0 和 1，只能表示两种不同的逻辑状态，但可以用多个逻辑变量的不同状态组合来表示事物的多种逻辑状态，以处理复杂的逻辑问题。

在数字电路中使用高、低电平表示两种不同的电路状态，它们表示的是一定的电压范围，而不是一个固定不变的电压值。例如，在 TTL 电路中，通常规定高电平的额定值是 3V，低电平的额定值是 0.2V，实际上 2～5V 都为高电平，0～0.8V 都为低电平。

在数字电路中如果用高电平表示逻辑状态 1，用低电平表示逻辑状态 0，称为正逻辑；反之用高电平表示逻辑状态 0，用低电平表示逻辑状态 1，称为负逻辑。两种逻辑之间是可以相互转变的，如无特殊说明，本书采用正逻辑。

3.1.1 逻辑变量与逻辑函数

逻辑代数的变量称为布尔变量或逻辑变量，和普通代数中的变量一样，通常用字母 A，B，C，…表示。和普通代数中变量不同的是，逻辑变量只有两种取值，即 0 或 1。并且，常量 0 和 1 没有普通代数中 0 和 1 的意义，它只表示两种对立的状态，即命题的“假”和“真”，信号的“无”和“有”等。

在普通代数中，函数这个概念是大家所熟悉的，即随着自变量变化而变化的因变量。与普通代数一样，在逻辑代数中，对于 n 个输入逻辑变量 $A,B,C,\cdots$，如果有

$$F=f(A,B,C,\cdots)$$

则称 F 为逻辑函数。逻辑函数与逻辑变量之间的关系称作逻辑函数表达式，简称为逻辑表达式。如果输入逻辑变量 $A,B,C,\cdots$ 等的取值确定了，逻辑函数的值也就被唯一地确定了。必须注意的是，在逻辑代数中，逻辑函数与逻辑变量一样只有两个取值：0 和 1。同样，这里的 0 和 1 并不表示具体的“数”，跟逻辑变量一样只表示两种不同的逻辑状态。任一逻辑函数和其变量的关系，都是由这些变量的与、或、非三种基本运算所决定的，也就是说，不管逻辑函数多么复杂，它都是由相应的输入变量的与、或、非三种基本逻辑运算构成的。

3.1.2　基本逻辑运算及基本逻辑门

在逻辑代数中，变量间有三种基本运算：与、或和非，任何复杂的逻辑运算都可以用这三种基本运算来实现。

1. 逻辑与(AND)运算

逻辑与运算也称为逻辑乘运算。这种运算表明的逻辑关系为当决定一事件的所有条件都具备之后，这个事件才会而且一定会发生。

图 3.1 所示的串联开关电路可用来说明与运算的逻辑关系。

这里的事件是灯亮，而该事件发生的条件则是开关接通。根据图 3.1，如果灯 F 亮，则开关 A,B 必须是全部接通的。如果开关 A 或 B 中有一个没有接通，灯 F 就不会亮。此处，开关接通跟灯亮之间的逻辑关系，就是逻辑与关系。

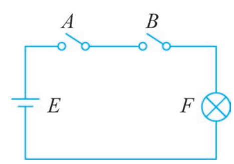

图 3.1　串联开关电路

用逻辑代数表示这种运算，可以表示成

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

式中，A、B 为自变量；F 为因变量；符号“·”为与运算符(也可以用“∧”或者“∩”来表示)，读作“与”，也可读作“乘”，这里的乘指逻辑乘。也可以省略与运算符，直接用 AB 表示逻辑乘。

根据串联开关电路图，可以很容易由 A、B 的取值推出 F 的取值。如果将 A、B 的闭合状态表示为“1”，断开状态表示为“0”；灯 F 亮时用“1”表示，不亮时用“0”表示。可以得出如下结果：

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

将上述关系列成表格，就叫与运算的“真值表”，如表 3.1 所示。当然两个以上的变量进行与运算的时候也可以用类似的情况得到真值表。当变量的个数为 n 时，与运算真值表有 2^n 种情况。

表 3.1　与运算的真值表

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

在数字电路中，实现“与”逻辑关系的电路叫作“与门”。

2. 逻辑或(OR)运算

逻辑或运算又称为逻辑加。这种运算表明的逻辑关系为决定某一事件的多个条件中只要有一个条件具备，这个事件就会发生。

可用并联开关电路来说明或运算的逻辑关系，如图 3.2 所示。

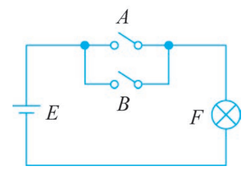

图 3.2 并联开关电路

同上例，这里的事件是灯亮，而该事件发生的条件则是开关接通。根据图 3.2，如果灯 F 亮，则开关 A、B 只需有一个接通即可。只有开关 A 和 B 都断开，灯 F 才会不亮。本例中的开关接通跟灯亮之间的逻辑关系，就是逻辑或关系。用逻辑代数表示这种运算，则可以表示成

$$F = A + B$$

式中，A、B 为自变量；F 为因变量；符号“+”为或运算符(也可以用“∨”或者“∪”来表示)，读作“或”，这里的加指逻辑加。

根据上例所定义的变量取值，可以很容易地根据并联开关电路推出逻辑或的运算：

$$0+0=0$$

$$0+1=1$$

$$1+0=1$$

$$1+1=1$$

或运算的真值表见表 3.2。同样，在多变量的情况下，n 个变量将导致 2^n 种情况。

表 3.2 或运算的真值表

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

在数字电路中，实现“或”逻辑关系的电路叫作“或门”。

3. 逻辑非(NOT)运算

逻辑非运算也就是反运算。它表示了当条件不满足时事件才发生的逻辑关系。

图 3.3 所示的电路可以说明这种非运算的逻辑关系。同样，这里的事件是灯亮，条件是开关接通。由图 3.3 可以看出，当开关 A 接通时灯不亮，但当开关 A 断开时灯反而是亮的。这就说明条件不满足的时候，事件才发生。这样的逻辑关系称作逻辑非运算。

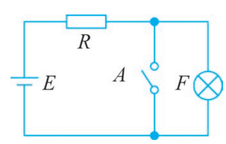

图 3.3 单开关电路

用逻辑代数表示这种运算，则可以表示成

$$F = \overline{A}$$

式中，字母 A 上面的一横表示“非”，也就是“反”的意思，可以直接读作“非”或“反”。同样，可得出在 A 不同的取值下 F 的值如下：

$$\overline{0} = 1$$

$$\overline{1} = 0$$

并据此得出非运算的真值表，如表 3.3 所示。

表 3.3 非运算的真值表

A	F
0	1
1	0

在数字电路中，实现逻辑非运算的电路叫作“非门”。

3.2 逻辑代数的公理、定理及规则

根据逻辑代数中的与、或、非三种基本运算，可以推导出逻辑代数运算的一些基本定律，称为逻辑代数的公理。本节将主要介绍逻辑代数中的公理、定理以及在逻辑运算中十分有用的规则。利用这些公理、定理和规则我们可以有效地化简逻辑函数，更好地分析设计逻辑电路。

3.2.1 逻辑代数的公理和基本定理

视频 12

1. 逻辑代数中的公理

0-1 律 $A+0=A$

$A+1=1$

$A\cdot 0=0$

$A\cdot 1=A$

交换律 $A+B=B+A$

$A\cdot B=B\cdot A$

结合律 $A+(B+C)=(A+B)+C$

$A\cdot(B\cdot C)=(A\cdot B)\cdot C$

分配律 $A\cdot(B+C)=A\cdot B+A\cdot C$

$A+B\cdot C=(A+B)\cdot(A+C)$

互补律 $A+\overline{A}=1$

$A\cdot\overline{A}=0$

重叠律 $A+A=A$

$A\cdot A=A$

非非律 $\overline{\overline{A}}=A$

以上公理很容易从开关电路图得到，也可以用真值表证明。

2. 逻辑代数中的定理

吸收律 $A+A\cdot B=A$

$A+\overline{A}\cdot B=A+B$

$A\cdot(A+B)=A$

$A\cdot(\overline{A}+B)=AB$

德·摩根定律 $\overline{A\cdot B}=\overline{A}+\overline{B}$

$\overline{A+B}=\overline{A}\,\overline{B}$

包含律 $AB+\overline{A}C=AB+\overline{A}C+BC$

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

逻辑代数中的定理可以由公理直接推导得出，也可以用真值表证明。下面我们给出分配律公式（加法对乘法的分配律）的推导法证明以及德·摩根定律公式的真值表证明。

【例 3.1】 证明加法对乘法分配律：$A+B\cdot C=(A+B)\cdot(A+C)$。

证明： 此处直接引用其他定理推导。

$$\begin{aligned} A+B\cdot C &= A(1+B+C)+BC && \text{0-1 律} \\ &= A+AB+AC+BC && \text{分配律} \\ &= AA+AB+AC+BC && \text{重迭律} \\ &= (AA+AC)+(AB+BC) && \text{交换、结合律} \\ &= A(A+C)+B(A+C) && \text{分配律} \\ &= (A+B)\cdot(A+C) \end{aligned}$$

【例 3.2】 证明德·摩根定律：$\overline{A\cdot B}=\overline{A}+\overline{B}$。

证明： 将等式左右两侧的真值表画在一起，根据 A、B 的不同取值分别计算真值。可以看出，$\overline{A\cdot B}$ 和 $\overline{A}+\overline{B}$ 的值完全相同，即等式两端相等，如表 3.4 所示。

表 3.4 德·摩根定律证明真值表

A	B	$\overline{A\cdot B}$	$\overline{A}+\overline{B}$
0	0	1	1
0	1	1	1
1	0	1	1
1	1	0	0

视频 13

视频 14

视频 15

视频 16

3.2.2 逻辑代数的基本规则

逻辑代数有三条重要的基本规则，分别是代入规则、反演规则和对偶规则。这些规则在逻辑运算中十分有用。

1. 代入规则

任何一个含有某变量 A 的逻辑等式中，如果将所有出现 A 的地方，都代之以一个逻辑函数 F，等式仍然成立，该规则被称为代入规则。

这是由于任何逻辑函数只有 0 和 1 两种取值，而逻辑变量也只有这两种取值，故代入之后不影响公式的恒等性，由此易知代入规则的正确性。有了代入规则，将已知等式中的某一变量代之以任一函数后，可以得到新的等式。这就扩大了等式应用范围，有助于我们更好地利用逻辑函数这种数学工具。

例如，我们可以任意给定一逻辑等式 $(A+B)\cdot C=AC+BC$，再给定一个函数 $F=A+D$，将出现 A 的地方都用函数 F 来代替，则该等式仍然成立。

$$(A+D+B)\cdot C=(A+D)\cdot C+BC=AC+CD+BC$$

代入规则在公式的推导中有着重要的意义，利用它可以将基本定律中的变量用函数代替，从而扩大了公式的应用范围。例如：推导含有三变量的德·摩根定律。

已知 $\overline{AB}=\overline{A}+\overline{B}$，可以用 $F=CD$ 代替 B，得到

$$\overline{ACD}=\overline{A}+\overline{CD}=\overline{A}+\overline{C}+\overline{D}$$

由此可知三个变量的德·摩根定律也是成立的。进一步可以证明，德·摩根定律对于任意个变量都是成立的。同理，可证明对于任意个变量的交换律、结合律也是成立的。

2. 反演规则

对于任意逻辑函数 F，如果将 F 中的所有“·”变成“+”，“+”变成“·”，“0”变成“1”，“1”变成“0”；将原变量变成反变量，反变量变成原变量，就得到了 $\overline{F}$。这是反演规则。$\overline{F}$ 也被称作 F 函数的反函数。

【例 3.3】 利用反演规则求 $F=A\overline{B}+\overline{C}D+0$ 的反函数。

解：$\overline{F}=(\overline{A}+B)(C+\overline{D})\cdot 1$

【例 3.4】 利用反演规则求 $F=A\cdot\overline{\overline{B}+\overline{\overline{C}D}}$。

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

在应用反演规则时，要注意保持原式的运算顺序。逻辑代数中的运算顺序是：先括号，再逻辑乘，最后是逻辑加。运算时，多于一个变量上的反号也起着括号的作用。

应用反演规则还必须注意两点。

(1) 在求反符号下有两个以上变量时，应用反演规则时求反符号保持不变。例如：$F=\overline{\overline{B}+\overline{\overline{C}D}}$ 的反函数是 $\overline{F}=\overline{B\cdot\overline{C+\overline{D}}}$。

(2) 由反演规则求得的反函数和用德·摩根定律求得的反函数一致。例如：$F=A\overline{B}+C$，利用德·摩根定律求解可以得到 $\overline{F}=\overline{A\overline{B}+C}=\overline{A\overline{B}}\,\overline{C}=(\overline{A}+B)\overline{C}$。同样，对函数 F 利用反演规则，直接得出反函数 $\overline{F}=(\overline{A}+B)\overline{C}$。同时也证明了反演规则的正确性。

3. 对偶规则

对于任意一个逻辑函数 F，如果将其所有的“·”变成“+”，“+”变成“·”，“0”变成“1”，“1”变成“0”，而变量保持不变，所得到的函数式叫作 F 的“对偶式”，记作 F'。F 与 F' 互为对偶式。

例如，$F=A+AB+0$，则其对偶式 $F'=A(A+B)\cdot 1$；$F=AB+\overline{A}C$，则其对偶式 $F'=(A+B)(\overline{A}+C)$。

所谓的对偶规则是指，如果两个逻辑函数式相等，那么，它们的对偶式也一定相等。例如：$F_1=A(B+C)$，$F_2=AB+AC$，显然两者是相等的。

而 $F_1'=A+BC$，$F_2'=(A+B)(A+C)=A+AC+BC=A+BC$，故 $F_1'=F_2'$。并且 $(F')'=F$。

在介绍逻辑代数的公理、定理时，给出了 11 对基本公式。很容易证明，每对公式的左边是对偶式，右边也是对偶式。例如德·摩根定律一对公式的左端 $\overline{AB}$ 和 $\overline{A+B}$ 是对偶式；右端 $\overline{A}+\overline{B}$ 和 $\overline{A}\overline{B}$ 也是对偶式。读者可以自己验证其他公式的对偶式。所以，根据对偶规则，在每对公式中，如果证明了一个成立，则另一个自然成立，无须证明。

求函数的对偶式，与求反函数一样，需要注意运算顺序的问题。

3.3 逻辑函数的表示方法

在给出逻辑函数的定义后，下面讨论逻辑函数的三种表示形式，并给出逻辑函数的标准

形式及表达式之间的转换。

3.3.1 逻辑函数的基本形式

逻辑函数常用的表示方法有逻辑表达式、真值表、卡诺图、波形图、逻辑图和硬件描述语言等。本节只介绍前面四种方法，逻辑图和硬件描述语言将在后面介绍。

1. 逻辑表达式

逻辑表达式是由逻辑变量及或、与、非三种运算符构成的式子，也就是用公式来表示逻辑函数的方法。

例如，要表示这样一个逻辑函数关系：当两个变量 A 和 B 取值相同时，函数取值为 0；否则，函数取值为 1。此函数称为异或逻辑函数，可以用下列逻辑表达式来表示：

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

显然，将 A 和 B 的四种可能取值代入这个表达式，均可验证是正确的。

逻辑函数表达式有“与或”（积之和）表达式和“或与”（和之积）表达式两种基本形式。

2. 真值表

真值表是由逻辑变量的所有可能取值的组合及其对应逻辑函数的取值所构成的表格，这是一种用表格表示逻辑函数的方法。对于异或逻辑函数也可以用表 3.5 所示的真值表来表示。表中列出了两个逻辑变量（A 和 B）所有可能的取值组合（00，01，10，11），并列出了与它们相对应的逻辑函数（F）的值。很容易看出，当 $A=B$ 的时候，$F=0$；当 $A\neq B$ 的时候，$F=1$。此真值表中的变量为两个，真值表项共有 2^2 种组合。可以推得当逻辑函数的变量个数为 n 的时候，真值表由 2^n 个表项组成。显然，随着变量数目的增多，真值表的规模就会变得很庞大。

表 3.5　异或逻辑函数的真值表

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

3. 卡诺图

卡诺图是美国工程师 Karnaugh 于 20 世纪 50 年代提出的。卡诺图是一种平面方格图，由若干小方格构成，每个小方格对应逻辑函数各逻辑变量的一组取值。将对应逻辑函数各逻辑变量所有可能取值组合的全部小方格按一定构造原则构成正方形或长方形，即为卡诺图。构造原则要求卡诺图中相邻两个小方格对应的两组逻辑变量取值相比，只有一个逻辑变量的取值不同，其他逻辑变量取值均相同。

一个逻辑函数的卡诺图就是将此函数在对应逻辑变量一组取值下的逻辑值填入相应的小方格内。利用卡诺图可以表示和化简逻辑函数，将在 3.4.2 节详细介绍。

4. 波形图

如果将逻辑函数输入变量的每一种可能出现的取值与对应的输出值按时间顺序依次排列，就得到了该逻辑函数的波形图，也称为时序图。由于波形图对逻辑函数表示的直观性，使其成为表示和分析电路逻辑关系常用的方法。在一些仿真工具中，常以波形图的形式给

出分析结果，也可通过实验观察波形图来检验所设计逻辑电路的功能是否正确。

图 3.4 所示是异或逻辑函数关系的波形。

以上表示逻辑函数的方法，虽然各有特点，适用于不同场合，但所描述的对象却是相同的。它们之间存在内在的联系，可以方便地相互变换。

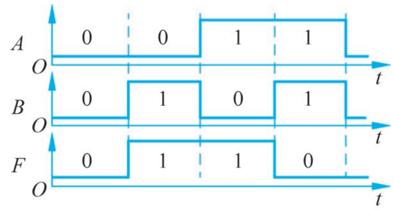

图 3.4 异或逻辑函数的波形

3.3.2 逻辑函数的标准形式

任何一个逻辑函数，其表达式的形式并不是唯一的。本节将介绍逻辑函数的两种基本形式，并在此基础上，进一步介绍逻辑函数基本形式向标准形式的转化。

1. 逻辑函数表达式的基本形式

1）“与或”表达式

所谓“与或”表达式是指在一个函数表达式中，包含着若干个“与项”，其中每个“与项”都可能包含一个或多个以原变量或反变量形式出现的逻辑变量，所有这些“与项”的“逻辑或”就构成了“与或”表达式。

例如，$\bar{B}$、$\bar{A}B$、$AB\bar{C}$ 都是与项，这 3 个与项的逻辑或就构成了三变量的逻辑函数的与或表达式，即 $F=\bar{B}+\bar{A}B+AB\bar{C}$。

2）“或与”表达式

所谓“或与”表达式是指在一个函数表达式中，包含着若干个“或项”，其中每个“或项”都可能有一个或多个以原变量或反变量形式出现的逻辑变量，所有这些“或项”的“逻辑与”就构成了“或与”表达式。

例如，$(A+B)$，$(\bar{C}+B)$，$(\bar{A}+B+C)$均为或项，这 3 个或项的逻辑与就构成了一个三变量的逻辑函数的或与表达式，即 $F=(A+B)(\bar{C}+B)(\bar{A}+B+C)$。

逻辑函数还可以表示成混合形式，这种表示形式既不是“与或”表达式，也不是“或与”表达式。但不论逻辑函数最初给出的是什么形式，它都可以转换为标准的“与或”表达式或“或与”表达式。

2. 最小项及最小项表达式

标准与或表达式又称作最小项表达式，即构成逻辑函数的各个与项都是最小项。下面先介绍最小项的概念。

1）最小项

什么叫最小项？可以从一个简单的例子来看。

现有一个两变量的逻辑函数

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

利用已经介绍过的基本公式，可以得到 F 的下列形式：

$$\begin{aligned} F &= A(\bar{B}+B)+\bar{B}=AB+A\bar{B}+\bar{B}(A+\bar{A})=AB+A\bar{B}+A\bar{B}+\bar{A}\bar{B} \\ &= AB+A\bar{B}+\bar{A}\bar{B} \end{aligned}$$

从这一系列的推导过程中可以看出，同一逻辑函数可以用多种形式来表示，有的比较简单，有的比较复杂，但它们所表示的逻辑意义都是相同的。在这些表达式中有一个最规则的形式，就是推导出来的最后一个表达式：

$$F=\bar{A}\bar{B}+A\bar{B}+AB$$

该表达式是由若干个乘积项之和组成的，该函数的全部逻辑变量（两个）以原变量（A 或 B）或反变量（$\bar{A}$ 或 $\bar{B}$）形式出现在表达式的乘积项中，且变量在一个乘积项中只出现一次。具有这样特点的乘积项称为最小项。我们将最小项的定义作如下规定：

设有 n 个逻辑变量，它们组成的乘积项（“与”项）中，每个变量或以原变量或以反变量形式出现一次，且仅出现一次，这个乘积项称为 n 变量的最小项。

显然，对于 n 个变量，则可以构成 2^n 个最小项。

为了叙述和书写方便，通常用 m_i 来表示最小项。如果各乘积项中的原变量记为 1，反变量记为 0，且当变量顺序确定后，0 或 1 按顺序排列成一个二进制数，则和这个二进制相对应的十进制数就是最小项的下标 i。例如按照上述规定求 $A\bar{B}\bar{C}$ 的下标，由于 A 为原变量，记为 1；B 为反变量，记为 0；C 为反变量，记为 0，从而得到表示下标的二进制数为 100，也就是十进制的 4。故最小项 $A\bar{B}\bar{C}$ 可用 m_4 表示，即 $A\bar{B}\bar{C}=m_4$。表 3.6 给出了三变量全部最小项的真值表。

表 3.6　三变量全部最小项的真值表

ABC 取值	m_0 $\bar{A}\bar{B}\bar{C}$	m_1 $\bar{A}\bar{B}C$	m_2 $\bar{A}B\bar{C}$	m_3 $\bar{A}BC$	m_4 $A\bar{B}\bar{C}$	m_5 $A\bar{B}C$	m_6 $AB\bar{C}$	m_7 ABC	$\sum_{i=0}^{7} m_i$
000	1	0	0	0	0	0	0	0	1
001	0	1	0	0	0	0	0	0	1
010	0	0	1	0	0	0	0	0	1
011	0	0	0	1	0	0	0	0	1
100	0	0	0	0	1	0	0	0	1
101	0	0	0	0	0	1	0	0	1
110	0	0	0	0	0	0	1	0	1
111	0	0	0	0	0	0	0	1	1

2）最小项的性质

从上述讨论及表 3.6 中不难得知最小项的下列三个主要性质。

（1）对于任何一个最小项 m_i，只有一组变量的取值才能使其值为 1。例如：最小项 $A\bar{B}C$ 只有在 A、B、C 的取值分别为 1、0、1 的时候，其值才为 1，其他时候其值均为 0。也就是说，最小项的值为 1 的概率最小，最小项也因此而得名。

（2）任意两个最小项 m_i 和 m_j（$i\neq j$）之积必为 0。例如，$m_3=\bar{A}BC$，$m_6=AB\bar{C}$，两者的乘积为 $m_3\cdot m_6=\bar{A}BC\cdot AB\bar{C}$。容易看出，无论 A、B、C 三个变量的取值是什么，该乘积的结果都为 0。

（3）n 个变量的所有最小项的逻辑和为 1，即 $\sum_{i=0}^{2^n-1} m_i=1$。这是由于对于 n 个变量的任何一组取值，总有一个最小项的值为 1。

3）最小项表达式

任何一个逻辑函数都可以用最小项之和的形式来表示，称为逻辑函数的最小项表达式，也称作标准与或表达式或主析取范式。

借用普通代数中的“$\sum$”符号表示多个最小项的累计“或”运算，圆括号内的十进制数

表示参与"或"运算的各个最小项的项号，它们是各最小项 m_i 的下标值。如两变量逻辑函数 $F=\bar{A}\bar{B}+A\bar{B}+AB$，它由 3 个最小项组成，可以表示成如下形式：

$$F=\bar{A}\bar{B}+A\bar{B}+AB=m_0+m_2+m_3=\sum m^2(0,2,3)$$

可以证明，任何 n 变量的逻辑函数都有一个且仅有一个最小项表达式。若已知某一逻辑函数不是最小项表达式，则可通过反复使用 $x=x(y+\bar{y})$ 而获得最小项表达式。

【例 3.5】 将三变量函数 $F=ABC+\bar{A}C+B\bar{C}$ 展开为最小项表达式。

解：

$$\begin{aligned}F&=ABC+\bar{A}C+B\bar{C}\\&=ABC+\bar{A}C(B+\bar{B})+B\bar{C}(A+\bar{A})\\&=ABC+\bar{A}BC+\bar{A}\bar{B}C+AB\bar{C}+\bar{A}B\bar{C}\\&=m_7+m_3+m_1+m_6+m_2\\&=\sum m^3(1,2,3,6,7)\end{aligned}$$

逻辑函数最小项表达式的三个主要性质如下。

(1) 若 m_i 是 n 变量逻辑函数 $F(A_1,A_2,\cdots,A_n)$ 的一个最小项，则使 $m_i=1$ 的一组变量取值也必定使 F 的值为 1。

(2) 若 F_1 和 F_2 都是 $A_1,A_2,\cdots,A_n$ 的函数，则 $F=F_1+F_2$ 将包括 F_1 和 F_2 中的所有最小项，$G=F_1\times F_2=F_1\cdot F$ 将包括 F_1 和 F_2 中的公共最小项。

(3) 若逻辑函数 $\bar{F}$ 是逻辑函数 F 的反函数，则 $\bar{F}$ 必定由 F 所包含的最小项之外的其余全部最小项所组成。可用三变量逻辑函数来说明，然后加以推广。

由于 $F(A,B,C)+\bar{F}(A,B,C)=1$，我们也知道 $\sum\limits_{i=0}^{7}m_i=1$。故可知

$$F(A,B,C)+\bar{F}(A,B,C)=\sum_{i=0}^{7}m_i$$

推而广之，则 $F(A_1,A_2,\cdots,A_n)+\bar{F}(A_1,A_2,\cdots,A_n)=\sum\limits_{i=0}^{2^n-1}m_i$

上式表明，一个最小项不在逻辑函数 F 中，就必在其反函数 $\bar{F}$ 中。

3. 最大项及最大项表达式

标准或与表达式又称作最大项表达式。即构成逻辑函数的各个或项都是最大项。下面先介绍最大项的概念。

1) 最大项

对于一个具有 n 个变量的函数的或项，它包含全部 n 个变量，其中每个变量都以原变量或以反变量的形式出现，且仅出现一次，这样的或项称作最大项。最大项的定义同最小项是类似的，例如，$n=2$ 的时候，$A+B$，$A+\bar{B}$，$\bar{A}+B$，$\bar{A}+\bar{B}$ 均为最大项；而 $n=3$ 时，$A+B+C$，$A+B+\bar{C}$ 为最大项，而 $A+B$，$\bar{C}+B$ 不是最大项。同样，与最小项类似，两个变量最多可以组成 4 个最大项；三个变量最多可以组成 2^3 个最大项；n 个变量最多可以组成 2^n 个最大项。

同样地，为了叙述和书写的方便，通常用 M_i 表示最大项。必须注意的是，这里的最大项编号(下标)和最小项编号恰恰相反。即，将或项中的原变量看作 0，反变量看作 1。这里 0 和 1 按变量排列顺序组成一个二进制数，同这个二进制数相对应的十进制数就是最大项

的下标 i。表 3.7 给出了三变量全部最大项的真值表。

表 3.7　三变量全部最大项的真值表

ABC 取值	M_0 $A+B+C$	M_1 $A+B+\bar{C}$	M_2 $A+\bar{B}+C$	M_3 $A+\bar{B}+\bar{C}$	M_4 $\bar{A}+B+C$	M_5 $\bar{A}+B+\bar{C}$	M_6 $\bar{A}+\bar{B}+C$	M_7 $\bar{A}+\bar{B}+\bar{C}$	$\prod_{i=0}^{7} M_i$
000	0	1	1	1	1	1	1	1	0
001	1	0	1	1	1	1	1	1	0
010	1	1	0	1	1	1	1	1	0
011	1	1	1	0	1	1	1	1	0
100	1	1	1	1	0	1	1	1	0
101	1	1	1	1	1	0	1	1	0
110	1	1	1	1	1	1	0	1	0
111	1	1	1	1	1	1	1	0	0

2）最大项的性质

从上述讨论及表 3.7 中不难得知最大项也有下列三个主要性质。

（1）对于任何一个最大项 M_i，只有一组变量的取值才能使其值为 0，而其他组变量的取值使该最大项的值均为 1。例如 $M_2=A+\bar{B}+C$，只有当 A、B、C 的取值分别为 0、1、0 时，M_2 取值为 0，而其他 A、B、C 的取值均使 M_2 的值为 1。也就是说，最大项为 1 的概率最大，最大项也因此而得名。

（2）任意两个最大项的逻辑和为 1。例如，$n=3$ 的时候，$M_4=\bar{A}+B+C$，$M_1=A+B+\bar{C}$，则 $M_4+M_1=\bar{A}+B+C+A+B+\bar{C}=1+B+1=1$。

（3）全部最大项的逻辑积等于 0。这是因为对于一个 n 个变量的任一组取值，都会有一个最大项的值为 0，故 $\prod_{i=0}^{2^n-1} M_i=0$。

3）最小项与最大项的关系

分析表 3.6 与表 3.7，可以看出：

$$\bar{m}_0=\overline{\bar{A}\bar{B}\bar{C}}=A+B+C=M_0$$

$$\bar{m}_1=\overline{\bar{A}\bar{B}C}=A+B+\bar{C}=M_1$$

$$\cdots$$

即 $\bar{m}_i=M_i$ 或者 $m_i=\bar{M}_i$，也就是说同一组逻辑变量相同下标的最小项与最大项是互补的。

4）最大项表达式

如上所述，任何一个 n 变量逻辑函数都可以表示成最小项之和的形式；同样地，也可以用最大项之积来表示任何 n 变量逻辑函数。所谓最大项表达式，就是由给定函数的最大项之积所组成的逻辑表达式。

将已知逻辑函数展开成最大项表达式，需要利用分配律 $A+B\cdot C=(A+B)\cdot(A+C)$将“积之和”表达式（即“与或”表达式）转换为“和之积”表达式（即“或与”表达式）。然后，在该式的各个非最大项的和项中加上它所缺变量的“原”“反”之积，并再次使用分配律，直到把全部和项都变成最大项，便可得到已知函数的最大项表达式。

【例 3.6】 已知逻辑函数 $F=A+\overline{A}BC$，求 F 的最大项表达式。

解：

$$
\begin{aligned}
F &= A+\overline{A}BC \\
&= (A+\overline{A})(A+BC) \\
&= 1\cdot(A+B)(A+C) \\
&= (A+B+C\overline{C})(A+C+B\overline{B}) \\
&= (A+B+C)(A+B+\overline{C})(A+C+B)(A+C+\overline{B}) \\
&= (A+B+C)(A+B+\overline{C})(A+\overline{B}+C) \\
&= M_0\cdot M_1\cdot M_2 \\
&= \prod M^3(0,1,2)
\end{aligned}
$$

利用逻辑函数的最小项表达式，可以用另外一种方法求得逻辑函数的最大项表达式。首先，将函数表达成最小项表达式；其次，找出其反函数中的最小项；最后，用和反函数中最小项相同编号的最大项构成原函数的最大项表达式。下面利用此方法求例 3.6 中给出的逻辑函数的最大项表达式。

【例 3.7】 已知逻辑函数 $F=A+\overline{A}BC$，求 F 的最大项表达式。

解：先求出该逻辑函数的最小项表达式

$$
\begin{aligned}
F &= A+\overline{A}BC \\
&= A(B+\overline{B})(C+\overline{C})+\overline{A}BC \\
&= (AB+A\overline{B})(C+\overline{C})+\overline{A}BC \\
&= ABC+AB\overline{C}+A\overline{B}C+A\overline{B}\,\overline{C}+\overline{A}BC \\
&= m_7+m_6+m_5+m_4+m_3 \\
&= \sum m^3(3,4,5,6,7)
\end{aligned}
$$

故 $\overline{F}=\sum m^3(0,1,2)$，利用最小项表达式求最大项表达式的规则，可以直接得到逻辑函数的最大项表达式 $F=\prod M^3(0,1,2)$，与例 3.6 求得的结果完全一样。

根据最大项表达式的定义，可推得类似于最小项表达式的三个性质，这里不再详述。

3.3.3 逻辑函数表达式的转换

视频 17

在用数字电路实现逻辑函数时，可供选择的数字电路元件主要是各种门电路。常用的门电路有与门、或门、与非门、或非门、与或非门等。为了便于用选定的门电路实现逻辑函数，必须把逻辑函数的最简式转换成与所选门电路一致的形式，这就需要研究不同形式的逻辑函数间转换的问题。本节将介绍相关逻辑函数表达式之间的相互转换。

1. 与或表达式转换为与非-与非表达式

将与或表达式转换为与非-与非表达式只需要对给定的与或表达式两次求反，并对底层非式使用德·摩根定律即可。

【例 3.8】 将逻辑函数 $F=ABC+BC+AB$ 转换为与非-与非表达式。

解：

$$F=\overline{\overline{F}}=\overline{\overline{ABC+BC+AB}}=\overline{\overline{ABC}\cdot\overline{BC}\cdot\overline{AB}}$$

2. 与或表达式转换为或非-或非表达式

当 F 是最简与或表达式的时候，先求 F 的对偶式 F'；再将 F' 变换成与非-与非形式；

最后求 F'的对偶式，还原 F，即 $F=(F')'$。

【例 3.9】 将最简式 $F=AB+A\bar{C}+\bar{A}C$ 转换为或非-或非表达式。

解： 先求 F 的对偶式，得

$$F'=(A+B)(A+\bar{C})(\bar{A}+C)=AC+\bar{A}B\bar{C}+ABC=AC+\bar{A}B\bar{C}$$

再对 F'两次求反，得

$$F'=\overline{\overline{AC+\bar{A}B\bar{C}}}=\overline{\overline{AC}\cdot\overline{\bar{A}B\bar{C}}}$$

求上式 F'的对偶式，得

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

3. 与或表达式变换为与或非表达式

将与或表达式变换为与或非表达式，需先求出逻辑函数的反函数 $\bar{F}$ 的与或表达式，再对 $\bar{F}$ 求反即可。

【例 3.10】 求 $F=AB+A\bar{B}\bar{C}+\bar{A}B\bar{C}+\bar{A}\bar{B}\bar{C}$ 的与或非表达式。

解： 求出函数 F 的反函数表达式

$$\bar{F}=\bar{A}C+\bar{B}C$$

再对 $\bar{F}$ 求反，可得

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

4. 与或表达式变换为或与表达式

将与或表达式变换为或与表达式需要经过如下步骤：先求出其反函数 $\bar{F}$ 的与或表达式；再对反函数 $\bar{F}$ 求反，最后用德·摩根定律进行相应变换。

【例 3.11】 求函数 $F=AC+AD+BC+BD$ 的或与表达式。

解： 先求出函数 F 的反函数 $\bar{F}$，可得

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

对 $\bar{F}$ 求反，可得

$$F=\overline{\bar{A}\bar{B}}\cdot\overline{\bar{C}\bar{D}}$$

用德·摩根定律进行变换，可得

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

5. 或与表达式变换为或非-或非表达式

将或与表达式变换为或非-或非表达式可以通过以下步骤：先求出逻辑函数的对偶式 F'；然后化简对偶式 F'；求出对偶式的对偶式，还原 F，即 $F=(F')'$；对上述 F 两次求反，并对底层非式使用德·摩根定律进行相应变换。

【例 3.12】 将函数 $F=(\bar{A}+\bar{B})(\bar{A}+\bar{B}+D)(A+C)(B+\bar{C})$转变为或非-或非表达式。

解： 求 F 的对偶式 F'并化简，可得

$$F'=\bar{A}\bar{B}+AC+B\bar{C}$$

进一步求 F'的对偶式，可得

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

对 F 两次求反，并用德·摩根定律进行变换可得

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

3.4 逻辑函数的化简

综上所述，逻辑函数的表达式有各种不同表示形式，即使是同一形式的表达式也有繁简之别。对于一个确定的逻辑函数，尽管函数表达式不同，但它们所描述的逻辑功能却是相同的。在数字系统中，实现这些逻辑功能的是逻辑电路。为了降低成本、减少复杂度，需要对逻辑函数化简。把逻辑函数简化成最简形式也称作逻辑函数的最小化。

逻辑函数的最简式是指将一个多变量的逻辑函数以最少的变量连接数和最少的运算符号表示出来，常见的逻辑函数的最简形式有最简与或式和最简或与式。

逻辑函数的最简与或式的定义如下。一个与给定函数等效的"与或"式中，若同时满足：

(1) 该式包含的乘积项最少；

(2) 该式中每个乘积项不能再用变量更少的乘积项来代替，即式中每个乘积项的因子最少。

则此与或式是给定逻辑函数的最简与或式。

最简或与式的定义与之类似，不再详述。

逻辑函数的化简有两种常用的方法，即公式法、卡诺图法。

3.4.1 公式法化简

视频 18

在公式法化简中，常应用并项法、吸收法、消去法和配项法。用公式法化简逻辑函数时，要求熟记并灵活应用逻辑代数中的基本公式、定理及常用公式。

1. 并项法

利用互补律 $A+\overline{A}=1$，将两项合并为一项，并消去一个变量。例如：

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

2. 吸收法

利用常用公式 $A+AB=A$，消去多余的项。例如：

$$F=AB+AB\overline{C}(D+E)=AB(1+\overline{C}(D+E))=AB$$

3. 消去法

利用常用公式 $A+\overline{A}B=A+B$，消去多余因子。例如：

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

4. 配项法

利用 $A\cdot 1=A$ 以及 $A+\overline{A}=1$，配在乘积项上，然后利用上述并项法、吸收法、消去法化简。例如：

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

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

上述化简过程的第四步，分别利用了吸收法和并项法。

或与式的化简方法有两种：其一，直接用公式化简；其二，利用对偶式，将其对偶式（与或式）进行化简，再求对偶式，从而得到最简的或与式。

【例 3.13】 化简或与表达式 $F=\bar{A}(\bar{B}+C+D)(A+B+C)(A+\bar{C}+D)$。

解： 方法一　公式法

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

方法二　利用对偶式

$$F'=\bar{A}+\bar{B}CD+ABC+A\bar{C}D=(\bar{A}+ABC+A\bar{C}D)+\bar{B}CD$$
$$=\bar{A}+BC+\bar{C}D+\bar{B}CD=\bar{A}+(\bar{C}D+\bar{B}CD)+BC$$
$$=\bar{A}+(\bar{B}C+\bar{C})D+BC=\bar{A}+(\bar{B}+\bar{C})D+BC$$
$$=\bar{A}+\overline{BC}D+BC=\bar{A}+D+BC$$

故 $F=(F')'=\bar{A}D(B+C)$

从上述例子可以看出，虽然公式法化简不受变量个数的限制，普遍适用，但化简过程没有固定的规律可循，直观性差，带有一定的试探性，并且难以判断结果是否已最简。为了寻找更简单的方法，人们研究出了卡诺图法化简。

视频 19

视频 20

视频 21

3.4.2 卡诺图法化简

在前述函数的表达方法中提到了卡诺图，利用卡诺图不仅可以表示函数，还能实现逻辑函数的简化，克服公式法直观性差的缺点。本节先介绍卡诺图的构成，然后介绍如何用卡诺图表示函数，以及卡诺图、真值表、表达式之间的转换；最后介绍用卡诺图化简逻辑函数的方法。

1. 卡诺图的构成

卡诺图是一种图形，是由 2^n 个小方格构成的正方形或长方形，n 表示变量的个数。其中每个小方格都对应一个最小项，并且在逻辑上有相邻性的最小项，在几何位置上也会被相邻地排列。相邻性是两个最小项之间只有一个变量互为反变量，其余变量均相同的特性。卡诺图的构成取决于变量的个数，下面分别介绍二变量、三变量、四变量和五变量的卡诺图。

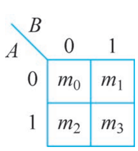

图 3.5　二变量的卡诺图

1）二变量的卡诺图

对于具有两个变量 A、B 的逻辑函数共有 4 个最小项：AB，$\bar{A}B$，$A\bar{B}$，$\bar{A}\bar{B}$。所以二变量的卡诺图应该包含 4 个小方格，如图 3.5 所示。

二变量卡诺图的构成方法是：先将逻辑变量分成两组，一组为 A（图 3.5 中，放在左边）；另一组为 B（图 3.5 中放在上边）。其次，在卡诺图上标注逻辑变量 0、1 的两种取值，用左边的数字 0 表示反变量 $\bar{A}$、数字 1 表示原变量 A，且从上而下地排列；上边的数字 0 表示反变量 $\bar{B}$、数字 1 表示原变量 B，且自左向右地排列。最后，按最小

项等于对应行上和列上的变量相与的原则，将最小项填于相应的小方格内，如：$m_1=\bar{A}B$，$m_2=A\bar{B}$ 等。从图 3.5 中易知，最小项的几何相邻性与逻辑相邻性是一致的。

2）三变量的卡诺图

具有 3 个逻辑变量 A、B、C 的逻辑函数共有 8 个最小项，所以其卡诺图应包含 8 个小方格。依据变量 A、B、C 的不同分组，将有两种形式的卡诺图。

若将逻辑变量 A、B、C 分成 A 一组，B 和 C 一组，其卡诺图如图 3.6(a)所示，而若分成 A 和 B 一组，C 一组，其卡诺图如图 3.6(b)所示。

以图 3.6(a)为例说明其构成方法：首先，把逻辑变量 A 放在左边，逻辑变量 B、C 的与项 BC 放在上边。其次，在卡诺图上标注逻辑变量或表达式的不同取值，用左边的数字 0 表示 $\bar{A}$、1 表示 A；用上边的数字 00 表示 $\bar{B}\bar{C}$、01 表示 $\bar{B}C$、11 表示 BC，10 表示 $B\bar{C}$。注意，数码排列按格雷码的规律，即相邻数码间只有一位不同。最后，按最小项等于相应行上和列上的变量相与的原则，将最小项填写于相应的小方格内。如 $m_2=\bar{A}B\bar{C}$，$m_5=A\bar{B}C$ 等。

A \ BC	00	01	11	10
0	m_0	m_1	m_3	m_2
1	m_4	m_5	m_7	m_6

(a)

AB \ C	0	1
00	m_0	m_1
01	m_2	m_3
11	m_6	m_7
10	m_4	m_5

(b)

图 3.6　三变量的卡诺图

由图 3.6(a)也可以看出，$m_0=\bar{A}\bar{B}\bar{C}$，$m_2=\bar{A}B\bar{C}$，$m_4=A\bar{B}\bar{C}$ 和 $m_6=AB\bar{C}$，虽然在几何上不相邻，但在逻辑上是相邻的，称这种相邻为首尾相邻。

3）四变量的卡诺图

4 个变量 A、B、C、D 共有 16 个最小项，所以其卡诺图应该包含 16 个小方格，如图 3.7 所示。

四变量卡诺图的构成方法：首先，画出包含 16 个小方格的正方形图形，并将变量 A 和 B 分为一组放在左边，变量 C 和 D 分成一组放在上边。其次，在图形左边自上而下依次标注数字 00、01、11、10，它们分别表示 $\bar{A}\bar{B}$、$\bar{A}B$、AB、$A\bar{B}$；在图形上边自左向右依次排列数字 00、01、11、10，它们分别表示 $\bar{C}\bar{D}$、$\bar{C}D$、CD、$C\bar{D}$。最后，按照最小项等于相应行上和列上的变量相与的原则，将最小项填写于相应的小方格内。

由图 3.7 可知，与三变量相似的是，四变量也具有首尾相邻性。如 m_0 与 m_2，m_0 与 m_8，m_2 与 m_{10} 和 m_8 与 m_{10}。

4）五变量的卡诺图

5 个变量 A、B、C、D、E 共有 32 个最小项，所以其卡诺图应包含 32 个小方格。把五变量中 A 和 B 分为一组，把 C、D 和 E 分为一组的卡诺图如图 3.8 所示。

AB \ CD	00	01	11	10
00	m_0	m_1	m_3	m_2
01	m_4	m_5	m_7	m_6
11	m_{12}	m_{13}	m_{15}	m_{14}
10	m_8	m_9	m_{11}	m_{10}

图 3.7　四变量的卡诺图

AB \ CDE	000	001	011	010	110	111	101	100
00	m_0	m_1	m_3	m_2	m_6	m_7	m_5	m_4
01	m_8	m_9	m_{11}	m_{10}	m_{14}	m_{15}	m_{13}	m_{12}
11	m_{24}	m_{25}	m_{27}	m_{26}	m_{30}	m_{31}	m_{29}	m_{28}
10	m_{16}	m_{17}	m_{19}	m_{18}	m_{22}	m_{23}	m_{21}	m_{20}

图 3.8　五变量的卡诺图

五变量的卡诺图构成方法与四变量类似，这里不再说明。主要强调两点：①五变量的卡诺图是以四变量的卡诺图（四列）右边线为对称轴线，作一个对称图形而构成的。②除了几何相邻、首尾相邻两种情况外，还存在一种所谓重叠相邻的情况，即按对称轴线对折卡诺图、相互重叠的最小项具有逻辑上的相邻性。如，$m_9=\overline{A}B\overline{C}\overline{D}E$ 和 $m_{13}=\overline{A}BC\overline{D}E$，$m_{27}=AB\overline{C}DE$ 和 $m_{31}=ABCDE$ 等。

由三变量的卡诺图的两种形式可知，变量分组方法不同，其卡诺图的形状、最小项的位置也不相同。另外，即使分组方法相同，但分组后的变量的排序位置若不同，其卡诺图也会相应地改变。上述例子是按字母的自然顺序分组变量，并按先图形左边再图形的上边来排列变量的。如四变量时，先把 AB 排在图形左边，再把 CD 排在图形上边。这是一种惯用的排法。也可用其他顺序来安排变量，但无论怎样安排变量，用卡诺图表示时，简化逻辑函数的本质都不会改变。

2. 用卡诺图表示逻辑函数

因为任何一个逻辑函数都可以表示成最小项表达式的形式，所以可利用卡诺图来表示逻辑函数。

1）用卡诺图表示最小项表达式

如果逻辑函数是以最小项的形式给出，则在构成函数的每个最小项相应的卡诺图小方格中填 1，其余的小方格中填 0。小方格中的 1 是指函数中有对应的最小项，而 0 是指函数中不存在该最小项，也可以将小方格中的 1 和 0 看作对应变量不同取值时的函数值。卡诺图中的 0 也可以不填。

2）用卡诺图表示非最小项表达式

如果逻辑函数不是最小项表达形式，可以利用互补律 $A+\overline{A}=1$，先将其变换成最小项表达式，然后用卡诺图表示。

【例 3.14】 试用卡诺图表示函数 $F(A,B,C,D)=\sum m^4(0,3,5,7,11,14)$。

解： 在逻辑函数每个最小项对应的卡诺图小方格中填 1，其余的小方格中填 0，可得到如图 3.9 所示的卡诺图。

【例 3.15】 用卡诺图表示逻辑函数 $F=AB+A\overline{C}$。

解： 先将逻辑函数 F 变换成最小项表达式，即 $F=ABC+AB\overline{C}+A\overline{B}\overline{C}=\sum m^3(4,6,7)$，然后再用卡诺图表示该逻辑函数，如图 3.10 所示。

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

图 3.9 例 3.14 卡诺图

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

图 3.10 例 3.15 卡诺图

如果逻辑函数是一般的与或表达式，也可以不将逻辑函数变换成最小项表达式，而直接用卡诺图表示它。

【例 3.16】　用卡诺图表示逻辑函数 $F=B+A\overline{C}$。

解：根据卡诺图的构成原理，变量 B 对应着卡诺图上 $B=1$ 的那些小方格，共有 4 个小方格；变量 $A\overline{C}$ 对应着卡诺图上 $A=1$，同时 $C=0$ 的那些小方格，共有 2 个小方格。在这些小方格中填 1。因此，可得出相应的卡诺图如图 3.11 所示。

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

图 3.11　例 3.16 卡诺图

事实上，卡诺图、真值表与逻辑表达式之间是可以相互转换的。我们已经介绍了逻辑表达式到真值表的转换，从真值表向逻辑函数转换也十分简单，只需将真值表中逻辑函数 $F=1$ 所对应的最小项求逻辑和，即得到从真值表转换的逻辑函数。同样，我们也可以直接从真值表得到卡诺图，其方法是将真值表中的逻辑函数值(1 或 0)，按对应顺序直接填入卡诺图相应的小方格内。

3. 用卡诺图化简逻辑函数

用卡诺图化简逻辑函数简单、直观，特别适合于四变量以下的逻辑函数的化简。本节先介绍用卡诺图进行函数化简的原理；再介绍合并最小项的规则；最后给出化简逻辑函数的步骤和化简实例。

1）用卡诺图化简逻辑函数的原理

一个函数的最小项表达式是与其卡诺图一一对应的。卡诺图形象地表达了最小项之间的相邻性。即卡诺图中每两个相邻的小方格的最小项只有一个变量互为反变量，其他变量均相同。因此，用卡诺图表示函数时，如有两个相邻的小方格均填 1，则可用相邻性消去一个变量，使函数得以简化。当填 1 的相邻小方格更多时，可以消去更多的变量，使函数更简化。所以，用卡诺图化简逻辑函数的依据是相邻性。

2）合并最小项的规则

利用相邻性合并最小项从而简化函数。下面分几种情况介绍最小项的合并。

（1）两个相邻最小项的合并。

图 3.12 给出了两个相邻最小项合并的各种情况。用一个称作卡诺圈的方圈，把填 1 的相邻小方格圈在一起。

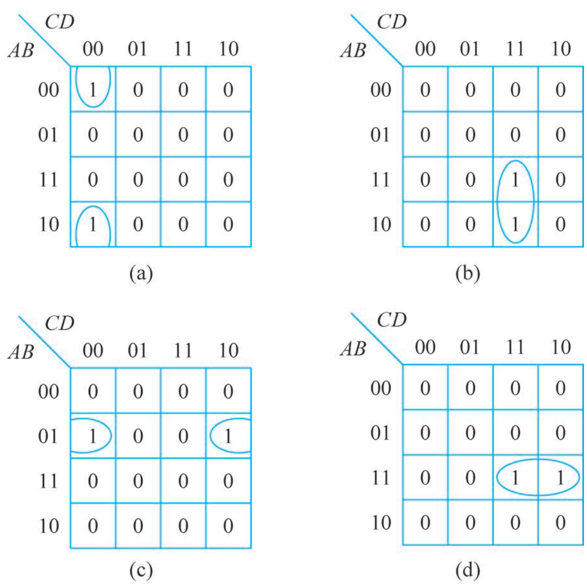

图 3.12　两个最小项的合并

从图 3.12 中可以看出，每个方圈内都包含一个互为反变量的变量：图 3.12(a)是 A 和 $\bar{A}$，图 3.12(b)是 B 和 $\bar{B}$，图 3.12(c)是 C 和 $\bar{C}$，图 3.12(d)是 D 和 $\bar{D}$，它们均可以消去。这样方圈内合并后的与项，图(a)是 $\bar{B}\bar{C}\bar{D}$，图(b)是 ACD，图(c)是 $\bar{A}B\bar{D}$，图(d)是 ABC。

(2) 4 个相邻最小项的合并。

图 3.13 给出了 4 个相邻最小项合并的各种情况。

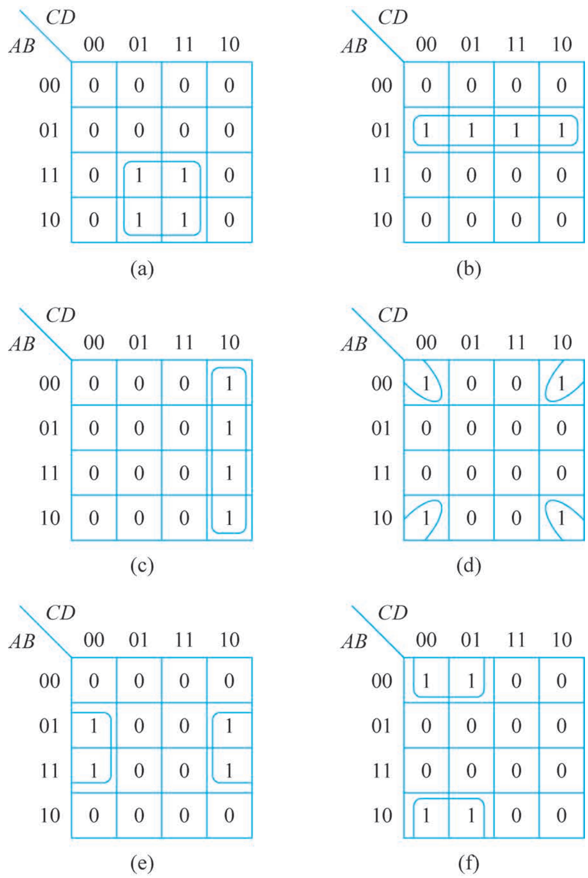

图 3.13　4 个最小项的合并

其中，图 3.13(a)的方圈左边对应着逻辑变量 A 和 B，A 取值不变，B 取值变化；上边对应着逻辑变量 C 和 D，C 取值变化，D 取值不变。所以 4 个最小项的合并，将消去取值变化的 2 个逻辑变量 B 和 C，留下取值不变的逻辑变量 A、D 作为合并后的与项 AD。

消去取值变化的逻辑变量，实际上就是消去卡诺圈内互为反变量的那些逻辑变量。由此不难归纳出最小项的合并规则：消去卡诺圈取值改变的逻辑变量，保留的那些取值不变的逻辑变量的逻辑与就是合并的结果。

由此可以得到图 3.13(a)～图 3.13(f)的合并结果分别为

$$AD,\bar{A}B,C\bar{D},\bar{B}\bar{D},B\bar{D},\bar{B}\bar{C}$$

(3) 8 个相邻最小项的合并。

图 3.14 展示了 8 个相邻最小项合并的各种情况。根据最小项的合并规则，圈有 8 个最小项的卡诺圈，将消去 3 个互为反变量的变量。图 3.14(a)～图 3.14(d)的合并结果分别为 B，D，$\bar{B}$，$\bar{D}$。

综上所述，可以归纳出 n 个变量卡诺图最小项的合并规律。

（1）卡诺圈中小方格的个数必须是 2^i 个，即 2 个、4 个、……，i 为小于或等于 n 的整数。

（2）卡诺圈中的 2^i 个最小项合并，将消去 i 个逻辑变量，其合并结果等于 $(n-i)$ 个逻辑变量的与项。例如，$n=4$ 时，$8=2^3$ 个最小项合并，消去 3 个逻辑变量，留下 1 个逻辑变量。

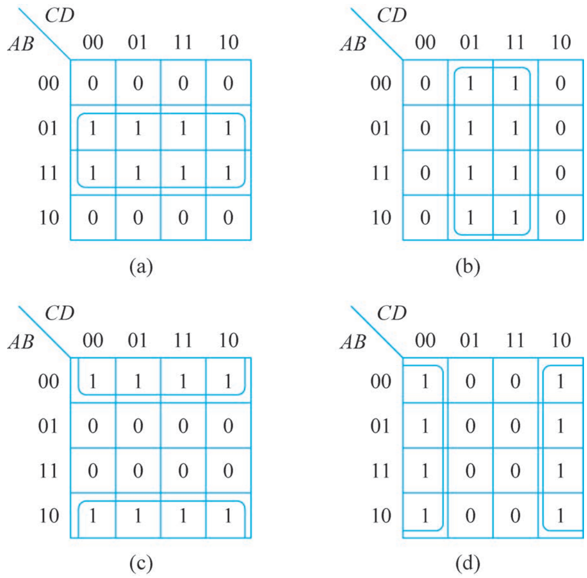

图 3.14 8 个最小项的合并结果

3）用卡诺图化简逻辑函数的步骤

（1）用卡诺图表示所要化简的逻辑函数。

（2）把卡诺图中所有填 1 的小方格用卡诺圈圈起来，将每个卡诺圈中的最小项进行合并。画圈时必须遵守如下原则：

① 每个圈内 1 的个数必须是 2^i 个；

② 每个圈中的小方格可多次被圈，但必须保证每个圈内至少有一个小方格仅被圈一次；

③ 卡诺圈的个数最少；

④ 每个圈应尽量大；

⑤ 所有填 1 的小方格必须圈完。

（3）将合并的与项进行逻辑加。

（4）如果卡诺图中填 0 的小方格比填 1 的少，也可以圈 0 先求得逻辑函数化简的反函数，然后取反求得原逻辑函数的最简与或式。

【例 3.17】 化简逻辑函数 $F(A,B,C)=\sum m^3(0,1,2,4,6,7)$。

解： 用卡诺图表示逻辑函数，如图 3.15 所示。经画圈合并，最后可以得到

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

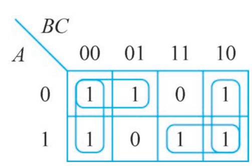

图 3.15 例 3.17 化简逻辑函数

【例 3.18】 化简逻辑函数 $F(A,B,C,D)=\sum m^4(1,5,6,7,11,12,13,15)$。

解： 用卡诺图表示逻辑函数，如图 3.16 所示。

根据图中实线所画的圈，得到逻辑函数最简表达式

$$F = AB\overline{C} + \overline{A}BC + \overline{A}\,\overline{C}D + ACD$$

图中虚线虽然也圈入了 4 个最小项，但这 4 个最小项已全部被其他卡诺圈圈过。因此，该圈中 4 个最小项所合并的与项 BD 为冗余项，不应出现在逻辑表达式中。

【例 3.19】 化简逻辑函数 $F=\overline{A}C+A\overline{B}+BC+A\overline{C}$。

解： 函数 F 的卡诺图如图 3.17 所示。根据图中所画的圈得到逻辑函数最简表达式

$$F = A + C$$

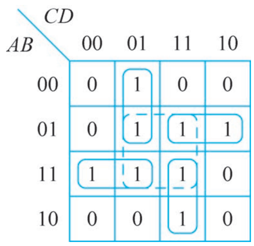

图 3.16　例 3.18 化简逻辑函数

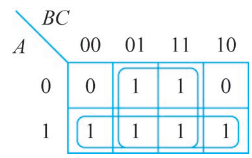

图 3.17　例 3.19 化简逻辑函数

本题卡诺图中填 0 的小方格比填 1 的少，所以也可以用圈 0 的方法先求出逻辑函数的反函数，即 $\overline{F}=\overline{A}\,\overline{C}$。

然后再对反函数取反，得到 F 的最简与或式

$$F = \overline{\overline{A}\,\overline{C}} = A + C$$

从例 3.19 可以看出，根据反演规则，很容易求得逻辑函数最简或与式。此处不再详述。

视频 22

3.4.3　具有无关项的逻辑函数及其化简

在实际的应用中，还有带有无关项的逻辑函数化简。所谓无关项包含约束项和任意项。

基于问题的背景，具体逻辑函数的有些输入变量的取值不可能出现或不允许出现。例如，以电动机工作状态指示电路为例，用逻辑变量 A、B、C 分别表示对一台电动机正转、反转和停止的命令，取 1 时有效；用 F 表示电动机的工作状态，取 1 时表示电动机运行中，取 0 时表示电动机停止运行。因为电动机任何时候只能执行其中的一个命令，所以不允许 A、B、C 中两个以上的变量同时为 1，ABC 的取值只可能是 001、010、100 中的某一种。此处，A、B、C 是一组有约束的变量，通常用约束条件来描述约束的具体内容。当限制某些输入变量的取值不能出现时，可以用它们对应的最小项恒等于 0 来表示。这样，约束条件可以表示为 $\overline{A}\,\overline{B}\,\overline{C}+\overline{A}BC+A\overline{B}C+AB\overline{C}+ABC=0$。将这些恒等于 0 的最小项称为逻辑函数的约束项。

还有一种情况，在某些逻辑函数中，输入变量的某些最小项取值可以是 1 也可以是 0，相应地，该逻辑函数对应的输出值也可以是 1 或是 0，并不影响电路的功能。这些最小项被称作任意项。同样以电动机工作状态指示电路为例。如果电路增加另一个逻辑函数 G，用来表示电动机控制状态，取 1 时表示控制命令出错，启动电机保护程序，取 0 时表示电动机正常工作。当 A、B、C 三个控制变量出现两个以上同时为 1 时，函数 F 表达式中对应的最小项和函数输出等于 1 还是 0 都无关紧要，因为此时函数 $G=1$，已表明控制命令出错，电动机被保护，函数 F 输出的电动机工作状态无效。

约束项和任意项都是逻辑函数式中的无关项，但二者又有区别。

约束项需要人为强行“不让它们出现或加以限制”。在此条件下，可以将约束项写进逻辑函数式中，也可以将约束项从函数式中删掉，都不会影响电路设计的结果，因为通过限制，约束项对应的逻辑变量的取值不会在输入中出现。在用卡诺图化简逻辑函数时，可根据需要，在卡诺图中对应约束项的小方格中填入“1”或“0”。但不难理解，约束项对应的小方格中的“1”只是一个“表象”，它实际上是不存在的，即实际上是“0”。但是，如果限制失败，使约束项客观出现了，而其取值不等于 0，就会导致电路的输出错误。

任意项则不然，任意项无须人为对这些变量的取值进行干预，任意项对应变量的取值可以出现，而此时逻辑函数的输出是 1 还是 0 皆可，并不影响电路的逻辑功能。

在对带有无关项的逻辑函数化简时，如果能够合理利用这些无关项，一般会得到更加简单的化简结果。无关项可随意加到逻辑函数表达式中或不加到逻辑函数表达式中，不影响逻辑函数的实际逻辑功能。但为达到化简逻辑函数的目的，加入的无关项应与逻辑函数尽可能多的最小项(包括原有的最小项和已加入的无关项)具有逻辑相邻性。可以根据具体情况，对无关项进行适当取舍，再用通常的化简方法进行化简。如用卡诺图化简逻辑函数时，某个无关项对应方格中填 1(将该无关项加入逻辑函数式中)还是填 0(逻辑函数式中不包含这个无关项)，应以得到的相邻最小项的卡诺圈最大，且卡诺圈数目最少为原则。

【例 3.20】 化简具有约束的逻辑函数。

$$F=\overline{A}\,\overline{B}\,\overline{C}\,\overline{D}+\overline{A}\,\overline{B}\,\overline{C}D+\overline{A}B\overline{C}D+\overline{A}BCD+A\overline{B}\,\overline{C}\,\overline{D}+A\overline{B}CD+ABC\overline{D}$$

给定约束条件为

$$\overline{A}\,\overline{B}CD+A\overline{B}\,\overline{C}D+AB\overline{C}\,\overline{D}+ABCD=0$$

在用最小项之和形式表示上述具有约束条件的逻辑函数时，也可以写成如下形式：

$$F(A,B,C,D)=\sum m^4(0,1,5,7,8,11,14)+\sum d^4(3,9,12,15)$$

式中，d 表示无关项，d 后面括号内的数字是无关项的最小项编号。

解： 如图 3.18 所示，画出 F 的卡诺图。

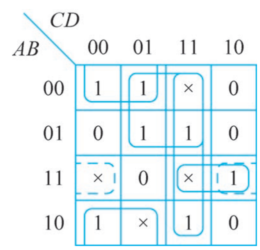

图 3.18　例 3.20 化简逻辑函数

若不利用约束项化简，$F=\overline{A}\,\overline{B}\,\overline{C}+\overline{A}BD+\overline{B}\,\overline{C}\,\overline{D}+ABC\overline{D}+A\overline{B}CD$；

利用约束项，经画圈合并得 $F=\overline{B}\,\overline{C}+\overline{A}D+CD+ABC$；

如果用虚线卡诺圈代替二变量的实线卡诺圈，得 $F=\overline{B}\,\overline{C}+\overline{A}D+CD+AB\overline{D}$。

这两种结果都是正确的。

可见，利用了约束项后，使逻辑函数得以进一步简化。从图 3.18 中可以看出，用实线卡诺圈化简时，为了得到最大的卡诺圈，取约束项 m_3、m_9 和 m_{15} 为 1，而没被圈进去的约束项 m_{12} 被当作 0。

习题 3

3.1　举例说明逻辑函数有哪些基本规则。

3.2　什么是最大项？什么是最小项？最大项、最小项各有哪些性质？

3.3　试用列真值表的方法证明下列异或运算公式。

(1) $A\oplus 0=A$

(2) $A\oplus 1=\overline{A}$

(3) $A\oplus A=0$

(4) $A\oplus\overline{A}=1$

(5) $(A\oplus B)\oplus C=A\oplus(B\oplus C)$

(6) $A(B\oplus C)=AB\oplus AC$

(7) $A\oplus\overline{B}=\overline{A\oplus B}=A\oplus B\oplus 1$

(8) $A\oplus B\oplus A=B$

3.4　用逻辑代数公理和定理证明：

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

(2) $(A\oplus B)\odot AB=\overline{A}\,\overline{B}$

(3) $A\cdot\overline{ABC}=A\overline{B}\,\overline{C}+A\overline{B}C+AB\overline{C}$

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

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

注：$A\odot B=AB+\overline{A}\,\overline{B}=\overline{A\oplus B}$

3.5　写出下列表达式的对偶式：

(1) $F_1=(A+B)(\overline{A}+C)(C+DE)+F$

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

(3) $F_3=\overline{\overline{A\overline{B}}\cdot\overline{C\overline{D}}\cdot\overline{D\overline{A}\,\overline{B}}}$

(4) $F_4=B\overline{(A\oplus B)}+B(A\oplus C)$

(5) $F_5=\overline{\overline{(C\oplus A)}\oplus(B\oplus\overline{D})}$

3.6　写出下列表达式的反函数：

(1) $F_1=((\bar{x}_1x_2+\bar{x}_3)x_4+\bar{x}_5)x_6$

(2) $F_2=S(\overline{W}+I(T+\overline{C}))+H$

(3) $F_3=A(\overline{B}+(C\overline{D}+\overline{E}F)G$

(4) $F_4=\overline{A}B+B\overline{C}+A(C+\overline{D})$

3.7　回答下列问题：

(1) 已知 $X+Y=X+Z$，那么 $Y=Z$ 正确吗？为什么？

(2) 已知 $XY=XZ$，那么 $Y=Z$ 正确吗？为什么？

(3) 已知 $X+Y=X+Z$，且 $XY=XZ$，那么 $Y=Z$ 正确吗？为什么？

(4) 已知 $X+Y=X\cdot Y$，那么 $X=Y$ 正确吗？为什么？

3.8　用公式法化简下列函数：

(1) $F_1=A\overline{B}+AC+BC$

(2) $F_2=A\overline{B}+B+BCD$

(3) $F_3=A+\overline{A}B+AB+\overline{A}\overline{B}$

(4) $F_4=AB+AD+\overline{B}\overline{D}+A\overline{C}\overline{D}$

3.9　将下列函数表示成为“最小项之和”形式及“最大项之积”形式：

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

(2) $F_2=AB\overline{C}+\overline{A}BC+A\overline{B}\overline{C}+\overline{A}\overline{C}$

(3) $F_3=BC+D+\overline{D}(\overline{B}+\overline{C})(AC+B)$

(4) $F_4=\bar{C}B+\bar{A}\bar{B}D+A\bar{B}C\bar{D}+ABCD$

3.10 用卡诺图法化简下列函数,并写出最简**与或**表达式和最简**或与**表达式:

(1) $F_1=(\bar{A}+\bar{B})(AB+C)$

(2) $F_2=\bar{A}\bar{B}+\bar{A}\bar{C}D+AC+B\bar{C}$

(3) $F_3=BC+D+\bar{D}(\bar{B}+\bar{C})(AD+B)$

(4) $F_4(A,B,C,D)=\sum m^4(2,3,4,5,10,11,12,13)$

(5) $F_5(A,B,C,D)=\prod M^4(2,4,6,10,11,12,13,14,15)$

3.11 用卡诺图法化简下列函数,并转化为最简与非-与非式。

(1) $F_1=A\bar{B}C+B\bar{C}$

(2) $F_2=(A+C)(\bar{A}+B+\bar{C})(\bar{A}+\bar{B}+C)$

(3) $F_3=\overline{AB\bar{C}+\bar{B}C\bar{D}}+\bar{A}\bar{B}D$

(4) $F_4=\overline{\overline{C\bar{D}}\,\overline{BC}\;\overline{ABC}\bar{D}}$

3.12 对于互相排斥的一组变量 A、B、C、D、E(即任何情况下,A、B、C、D、E 不可能有两个或两个以上同时为 1),试证明 $A\bar{B}\bar{C}\bar{D}\bar{E}=A$,$\bar{A}B\bar{C}\bar{D}\bar{E}=B$,$\bar{A}\bar{B}C\bar{D}\bar{E}=C$,$\bar{A}\bar{B}\bar{C}D\bar{E}=D$,$\bar{A}\bar{B}\bar{C}\bar{D}E=E$。

3.13 用卡诺图法化简具有无关项的逻辑函数为最简与或形式。

(1) $F_1(A,B,C,D)=\sum m(0,2,7,13,15)+\sum d(1,3,4,5,6,8,10)$

(2) $F_2=C\bar{D}(A\oplus B)+\bar{A}B\bar{C}+\bar{A}\bar{C}D$,约束条件为 $AB+CD=0$

(3) $F_3=(A\bar{B}+B)C\bar{D}+\overline{(A+B)(\bar{B}+C)}$,约束条件为 $ACD+BCD=0$

(4) $F_4(A,B,C,D)=\sum m(2,3,7,8,11,14)+\sum d(0,5,10,15)$