数 字 逻 辑

丁 贤 庆

ahhfdxq@163.com

1.5 二值逻辑变量与基本逻辑运算

*逻辑运算：当0和1表示逻辑状态时，两个二进制数码按照某种特定的因果关系进行的运算。

逻辑运算使用的数学工具是逻辑代数。

* 逻辑代数与普通代数:与普通代数不同,逻辑代数中的变量只有0和1两个可取值，它们分别用来表示完全两个对立的逻辑状态。

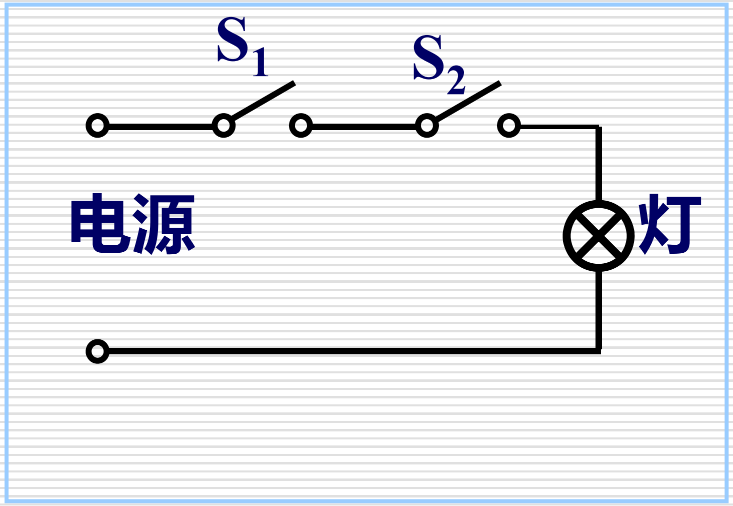
在逻辑代数中，有与、或、非三种基本的逻辑运算。

逻辑运算的描述方式:逻辑代数表达式、真值表、逻辑图、卡诺图、波形图和硬件描述语言（HDL) 等。

1．与运算

(1)　与逻辑:只有当决定某一事件的条件全部具备时，这一事件才会发生。这种因果关系称为与逻辑关系。

与逻辑举例



电路状态表

开关S_1	开关S_2	灯
断	断	灭
断	合	灭
合	断	灭
合	合	亮

1．与运算

与逻辑举例状态表

开关S_1	开关S_2	灯
断	断	灭
断	合	灭
合	断	灭
合	合	亮

逻辑真值表

A	B	L
0	0	0
0	1	0
1	0	0
1	1	1

与逻辑符号

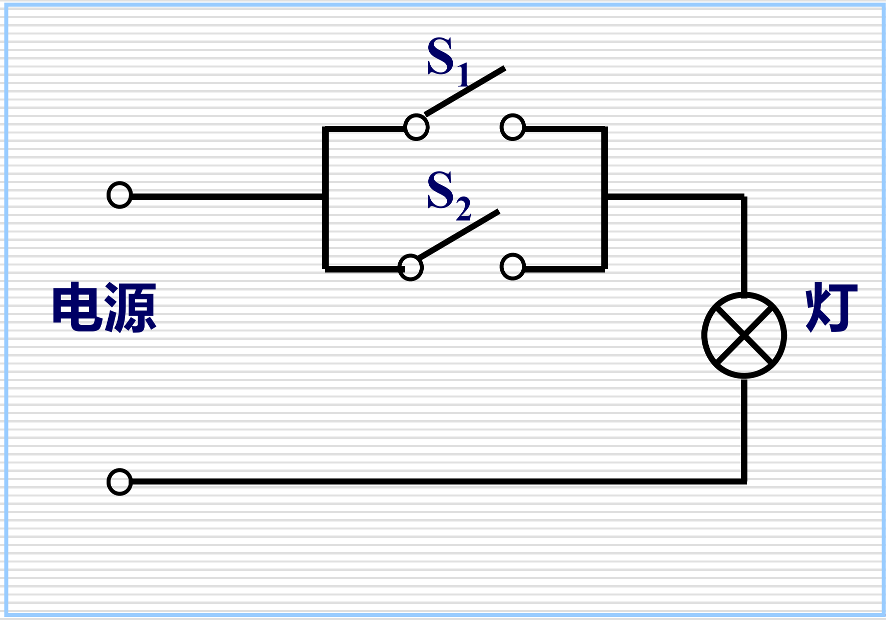
逻辑表达式

与逻辑：$L = A \cdot B = AB$

2、或运算

只要在决定某一事件的各种条件中，有一个或几个条件具备时，这一事件就会发生。这种因果关系称为或逻辑关系。

或逻辑举例



电路状态表

开关S_1	开关S_2	灯
断	断	灭
断	合	亮
合	断	亮
合	合	亮

2、或运算

或逻辑举例状态表

开关S_1	开关S_2	灯
断	断	灭
断	合	灭
合	断	灭
合	合	亮

逻辑真值表

A	B	L
0	0	0
0	1	1
1	0	1
1	1	1

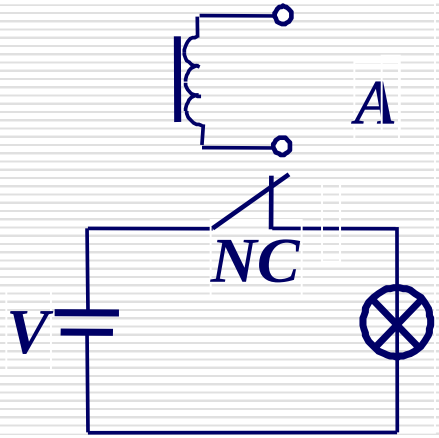
或逻辑符号

逻辑表达式 或逻辑： $L = A + B$

3. 非运算

事件发生的条件具备时，事件不会发生；事件发生的条件不具备时，事件发生。这种因果关系称为非逻辑关系。

非逻辑举例



非逻辑举例状态表

A	灯
不通电	亮
通电	灭

3. 非运算

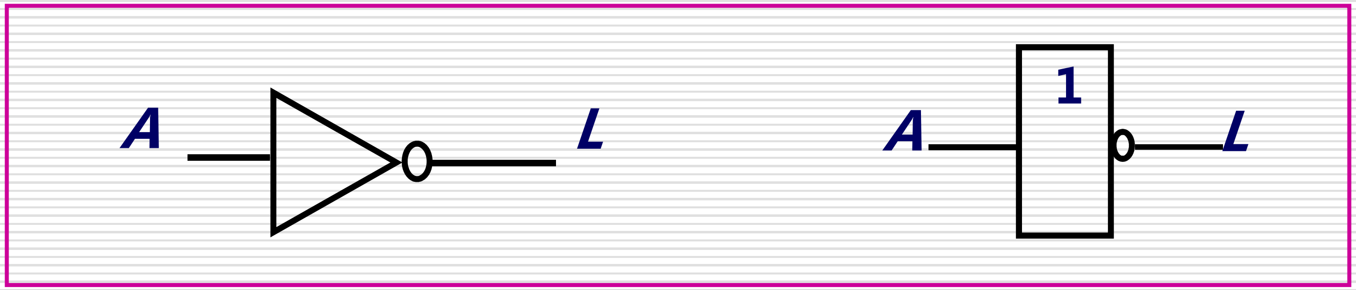
非逻辑举例状态表

A	灯
不通电	亮
通电	灭

非逻辑真值表

A	L
0	1
1	0

非逻辑符号



逻辑表达式

$$L = \overline{A}$$

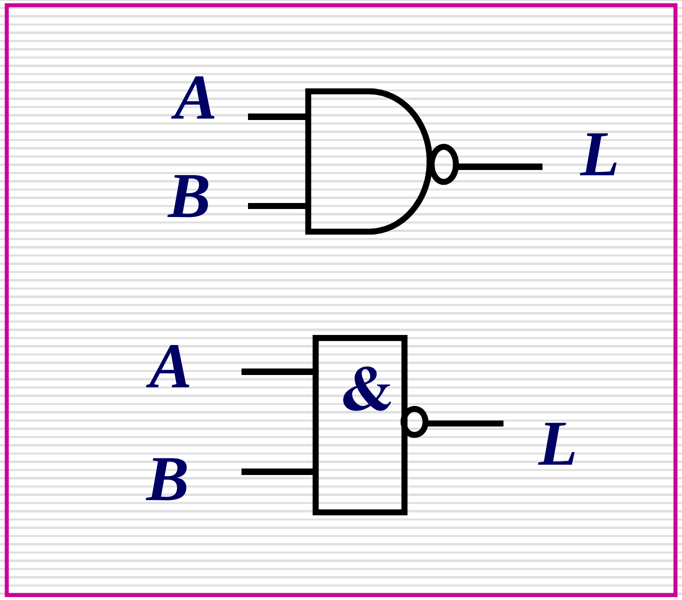
4. 几种常用复合逻辑运算

1)与非运算

两输入变量与非逻辑真值表

A	B	L
0	0	1
0	1	1
1	0	1
1	1	0

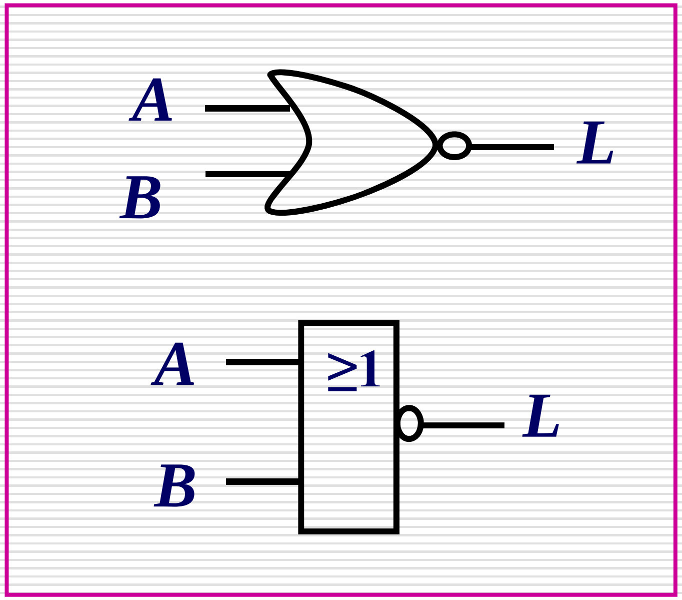
与非逻辑符号



与非逻辑表达式 $L = \overline{A \cdot B}$

2)或非运算

或非逻辑符号



两输入变量或非逻辑真值表

A	B	L
0	0	1
0	1	0
1	0	0
1	1	0

或非逻辑表达式

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

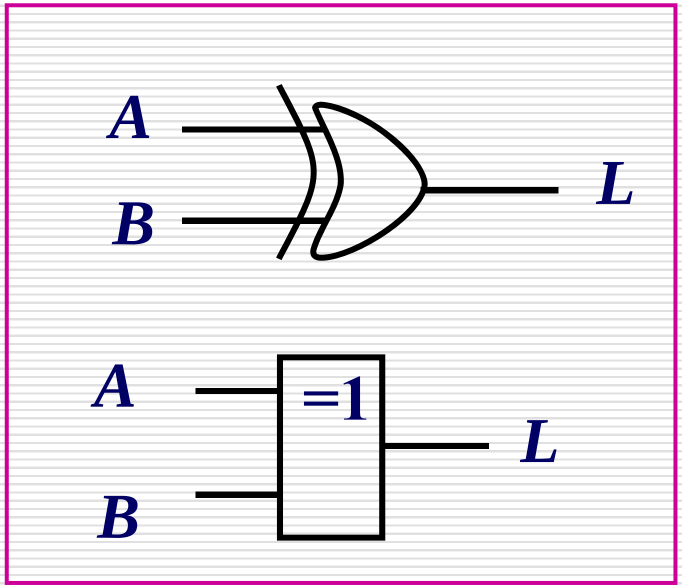
3）异或逻辑

若两个输入变量的值相异，输出为1，否则为0。

异或逻辑真值表

A	B	L
0	0	0
0	1	1
1	0	1
1	1	0

异或逻辑符号



异或逻辑表达式 $L = A \oplus B = \overline{A}B + A\overline{B}$

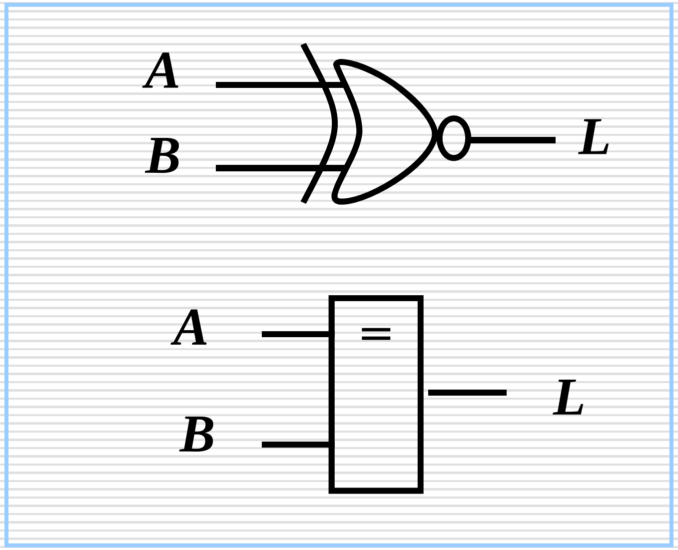
4） 同或运算

若两个输入变量的值相同，输出为1，否则为0。

同或逻辑真值表

A	*B*	*L*
0	0	1
0	1	0
1	0	0
1	1	1

同或逻辑逻辑符号



同或逻辑表达式

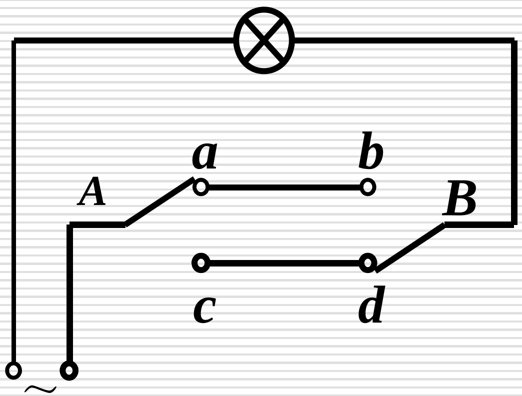
$$L=AB+\overline{A}\,\overline{B}=A\odot B$$

1.6 逻辑函数的建立及其表示方法

1.6.1 逻辑函数几种表示方法

1. 真值表表示

楼道灯开关示意图



确定变量、函数，并赋值

开关：变量 A、B
灯 ：函数 L

A、B：向上—1 向下--0
L ：亮---1； 灭---0

逻辑抽象，列出真值表

开关状态表

开关 A	开关 B	灯
下	下	亮
下	上	灭
上	下	灭
上	上	亮

逻辑真值表

A	B	L
0	0	1
0	1	0
1	0	0
1	1	1

2、逻辑函数表达式表示。

逻辑表达式是用与、或、非等运算组合起来，表示逻辑函数与逻辑变量之间关系的逻辑代数式。

例：已知某逻辑函数的真值表，试写出对应的逻辑函数表达式。

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

逻辑真值表

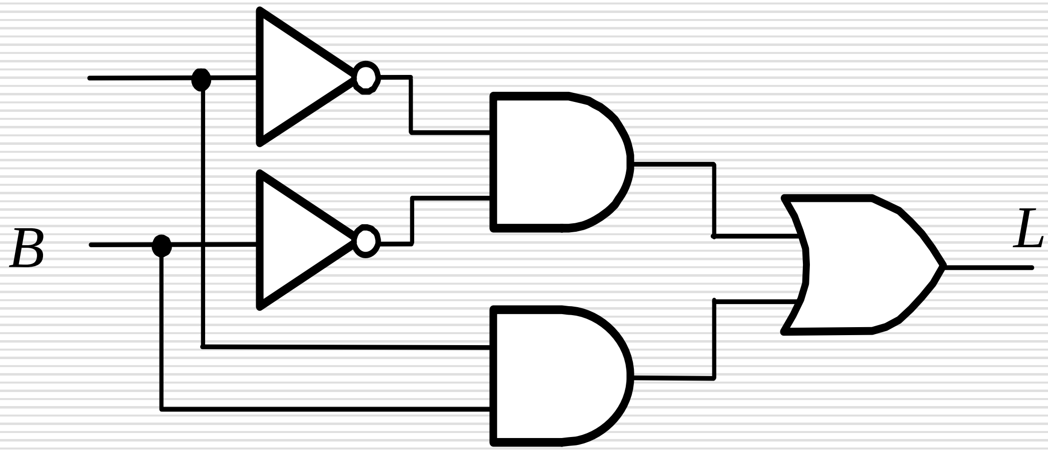
A	B	L
0	0	1
0	1	0
1	0	0
1	1	1

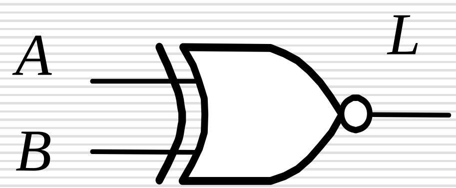
3. 逻辑图表示方法

用与、或、非等逻辑符号表示逻辑函数中各变量之间的逻辑关系所得到的图形称为逻辑图。

将逻辑函数式中所有的与、或、非运算符号用相应的逻辑符号代替，并按照逻辑运算的先后次序将这些逻辑符号连接起来，就得到图电路所对应的逻辑图

例：已知某逻辑函数表达式为 $L = \overline{A}\,\overline{B} + AB$ 试画出其逻辑图



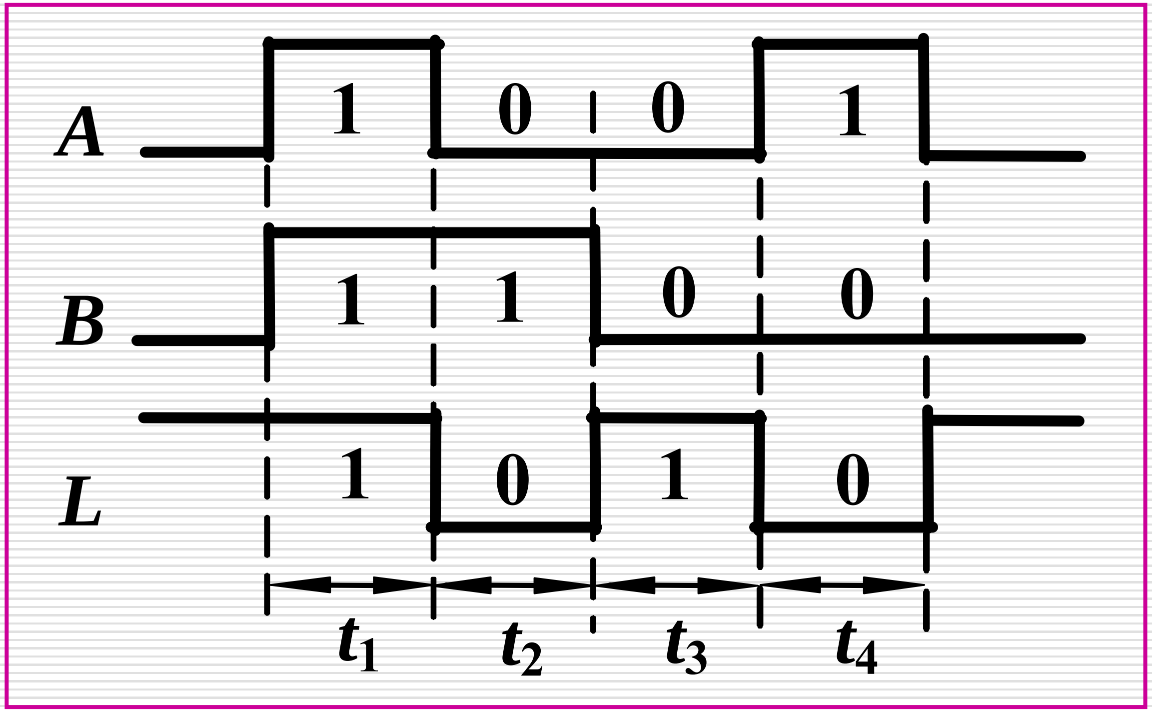


4. 波形图表示方法

用输入端在不同逻辑信号作用下所对应的输出信号的波形图，表示电路的逻辑关系。

真值表

A	B	L
0	0	1
0	1	0
1	0	0
1	1	1



1.6.2 逻辑函数表示方法之间的转换

逻辑函数的真值表、逻辑函数表达式、逻辑图、波形图、卡诺图及HDL描述之间可以相互转换。这里介绍两种转换。

1.真值表到逻辑图的转换

真值表如右表。

转换步骤：

(1)根据真值表写出逻辑表达式

$$L = \overline{A}BC + AB\overline{C}$$

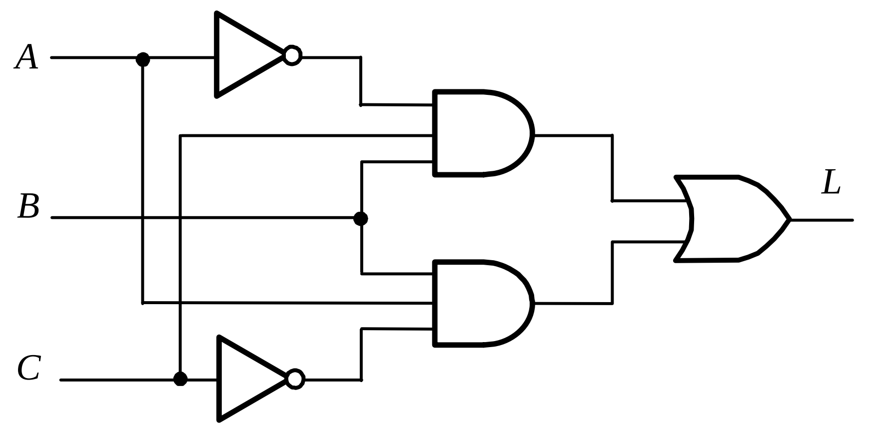
(2)化简逻辑表达式（第2章介绍）

上式不需要简化

A	*B*	*C*	*L*
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

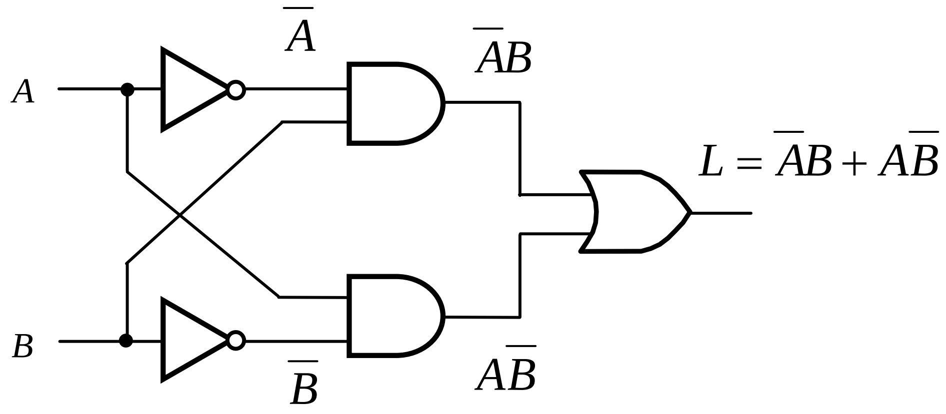
(3)根据与或逻辑表达式画逻辑图

$$L = \overline{A}BC + AB\overline{C}$$



用与、或、非符号代替相应的逻辑符号，注意运算次序。

2. 逻辑图到真值表的转换



转换步骤：

(1)根据逻辑图逐级写出表达式

(2)化简变换求最简与或式

(3)将输入变量的所有取值逐一代入表达式得真值表

A	B	L
0	0	0
0	1	1
1	0	1
1	1	0

第二章 习题

要求：

每个同学自己从第二章的课后习题中选3大题做到作业本中，至于做哪3大题，每个同学自己选择。不作硬性规定。我在检查作业时，只看是否做了3大题。（在你购买的课本上选题做就可以了。）

第 二 章

逻辑代数与硬件描述语言基础

2.1 逻辑代数的基本定理和规则

2.1.1 逻辑代数的基本定律和恒等式

2.1.2 逻辑代数的基本规则

2.1.1 逻辑代数的基本定律和恒等式

1、基本公式

0、1律：$A+0=A \quad A+1=1 \quad A\cdot 1=A \quad A\cdot 0=0$

互补律：$A+\overline{A}=1 \quad A\cdot\overline{A}=0$

交换律：$A+B=B+A \quad A\cdot B=B\cdot A$

结合律：$A+B+C=(A+B)+C \quad A\cdot B\cdot C=(A\cdot B)\cdot C$

分配律：$A(B+C)=AB+AC \quad A+BC=(A+B)(A+C)$

重叠律: $A + A = A$ $A \cdot A = A$

反演律(摩根定理): $\overline{A+B} = \overline{A} \cdot \overline{B}$ $\overline{AB} = \overline{A} + \overline{B}$

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

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

其它常用恒等式

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

$$AB + \overline{A}C + BCD = AB + \overline{A}C$$

2、基本公式的证明 (真值表证明法)

例 证明 $A+\overline{A}\cdot B=A+B$

列出等式、右边的函数值的真值表

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

例：试化简下列逻辑函数 $L=(A+B)(\overline{A}+B)$

$$L = A\overline{A} + AB + B\overline{A} + BB(\text{分配律})$$

$$= 0 + AB + B\overline{A} + B \qquad (A\cdot\overline{A} = 0,\ \ A\cdot A = A)$$

$$= AB + B\overline{A} + B \qquad (A + 0 = A)$$

$$= B(A + \overline{A} + 1) \qquad [AB + AC = A(B + C)]$$

$$= B\cdot 1 = B \qquad (A + 1 = A,\ \ A\cdot 1 = A)$$

2.1.2 逻辑代数的基本规则

1. 代入规则： 在包含变量A逻辑等式中，如果用另一个函数式代入式中所有A的位置，则等式仍然成立。这一规则称为代入规则。

例：$B(A+C)=BA+BC$，

用$A+D$代替A，得

$$B[(A+D)+C]=B(A+D)+BC=BA+BD+BC$$

代入规则可以扩展所有基本公式或定律的应用范围

2. 反演规则：

对于任意一个逻辑表达式L，若将其中所有的与（·）换成或（+），或（+）换成与（·）；原变量换为反变量，反变量换为原变量；将1换成0，0换成1；则得到的结果就是原函数的反函数。

例2.1.1 试求 $L = \overline{A}\,\overline{B} + CD + 0$ 的非函数

解：按照反演规则，得

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

3. 对偶规则：

对于任何逻辑函数式，若将其中的与（·）换成或（+），或（+）换成与（·）；并将1换成0，0换成1；那么，所得的新的函数式就是L的对偶式，记作 L' 。

例: 逻辑函数 $L=(A+\overline{B})(A+C)$的对偶式为

$$L' = A\overline{B} + AC$$

当某个逻辑恒等式成立时，则该恒等式两侧的对偶式也相等。这就是对偶规则。利用对偶规则，可从已知公式中得到更多的运算公式，例如，吸收律

2.2 逻辑函数表达式的形式

2.2 逻辑函数表达式的形式

2.2.1 逻辑函数表达式的基本形式

1、与-或表达式

若干与项进行或逻辑运算构成的表达式。由与运算符和或运算符连接起来。

$$L = A \cdot C + \overline{C} \cdot D$$

2、或-与表达式

若干或项进行与逻辑运算构成的表达式。由或运算符和与运算符连接起来。

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

通常表达式为混合形式

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

经过变换可转换为上述两种基本形式

2.2 .2 最小项与最小项表达式

1. 最小项的定义和性质

n个变量$X_{1,}\ X_{2,}\ \ldots,\ X_{n}$的最小项是$n$个因子的乘积，每个变量都以它的原变量或非变量的形式在乘积项中出现，且仅出现一次。一般n个变量的最小项应有2^n个。

例如，A、B、C三个逻辑变量的最小项有（$2^3=$）8个，即

$\bar{A}\bar{B}\bar{C}$、$\bar{A}\bar{B}C$、$\bar{A}B\bar{C}$、$\bar{A}BC$、$A\bar{B}\bar{C}$、$A\bar{B}C$、$AB\bar{C}$、ABC

$\bar{A}B$、$A\bar{B}C\bar{A}$、$A(B+C)$等则不是最小项。

2、最小项的性质　　三个变量的所有最小项的真值表

A	B	C	$\bar{A}\bar{B}\bar{C}$	$\bar{A}\bar{B}C$	$\bar{A}B\bar{C}$	$\bar{A}BC$	$A\bar{B}\bar{C}$	$A\bar{B}C$	$AB\bar{C}$	ABC
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

- 对于任意一个最小项，只有一组变量取值使得它的值为1；
- 任意两个最小项的乘积为0；
- 全体最小项之和为1。

3、最小项的编号

三个变量的所有最小项的真值表

			m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7
A	B	C	$\overline{A}\,\overline{B}\,\overline{C}$	$\overline{A}\,\overline{B}C$	$\overline{A}B\overline{C}$	$\overline{A}BC$	$A\overline{B}\,\overline{C}$	$A\overline{B}C$	$AB\overline{C}$	ABC
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

最小项的表示：通常用m_i表示最小项，m 表示最小项,下标i为最小项号。（1---对应原变量。0---对应反变量）

2. 最小项表达式

由若干最小项相或构成的表达式，也称为标准与-或式。

- **为“与或”逻辑表达式；**
- **在“与或”式中的每个乘积项都是最小项。**

例1 将 $L(A,B,C)=AB+\overline{A}C$ **化成最小项表达式**

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

例2 将 $L(A,B,C)=\overline{(AB+\overline{A}\,\overline{B}+\overline{C})\overline{AB}}$ 化成最小项表达式

a.去掉非号 $L(A,B,C)=\overline{(AB+\overline{A}\,\overline{B}+\overline{C})}+AB$

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

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

b.去括号

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

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

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

$$=m_3+m_5+m_7+m_6=\sum m(3,5,6,7)$$

2.2.2 最大项与最大项表达式

1. 最大项的定义和性质

n个变量$X_{1,}X_{2,}\ldots, X_n$的最大项是n个因子或相，每个变量都以它的原变量或非变量的形式在或项中出现，且仅出现一次。一般n个变量的最大项应有2^n个。

例如，A、B、C三个逻辑变量的最大项有（$2^3 =$ ）8个，即

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

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

1. 最大项的定义和性质

最大项的表示：通常用M_j表示最大项，M 表示最大项,下标j为最大项号。 (1---对应反变量。0---对应原变量)

最大项的性质：

- 对于任意一个最大项，只有一组变量取值使得它的值为0；
- 任意两个最大项的之和为1；
- 全体最大项之积为0。

2. 最小项和最大项的关系

两者之间为互补关系：$m_i = \overline{M_i}$ ，或者$M_i = \overline{m_i}$

例: 逻辑电路的真值表如右，写出最小项和最大项表达式。

最小项表达式：

将 $L=1$ 的各个最小项相加

$$L(A,B,C)=m_3+m_5+m_6$$
$$=\sum m(3,5,6)$$
$$=\overline{A}\cdot B\cdot C+A\cdot\overline{B}\cdot C+A\cdot B\cdot\overline{C}$$

最大项表达式：

将 $L=0$ 的各个最大项相乘

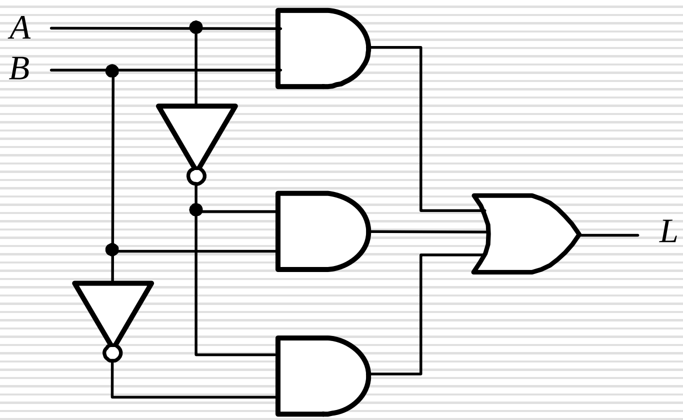
$$L(A,B,C)=M_0\cdot M_1\cdot M_2\cdot M_4\cdot M_7$$
$$=\prod M(0,1,2,4,7)$$
$$=(A+B+C)\cdot(A+B+\overline{C})\cdot(A+\overline{B}+C)\cdot(\overline{A}+B+C)\cdot(\overline{A}+\overline{B}+\overline{C})$$

A	B	C	L	
0	0	0	0	M_0
0	0	1	0	M_1
0	1	0	0	M_2
0	1	1	$1\rightarrow m_3$	
1	0	0	0	M_4
1	0	1	$1\rightarrow m_5$	
1	1	0	$1\rightarrow m_6$	
1	1	1	0	M_7

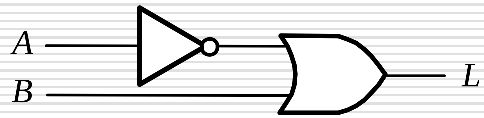
2.3 逻辑函数的代数化简法

2.3 逻辑函数的代数法化简

化简的目的：降低电路实现的成本，以较少的门实现电路。



(a) 标准积之和的电路



(b) 成本最低的电路

图（a）和图（b）的电路逻辑功能相同，但图 （b）电路简单可靠性高，成本低。

2.3.1 逻辑函数的最简形式

逻辑函数有不同形式，如与-或表达式、与非-与非表达式、或-与表达式、或非-或非表达式以及与-或-非表达式等。

将其中包含的与项数最少，且每个与项中变量数最少的与-或表达式称为**最简与-或表达式**。

$$L = AC + \bar{C}D$$ “与-或” 表达式

$$= \overline{\overline{AC} \cdot \overline{\bar{C}D}}$$ “与非-与非”表达式

$$= (A+\bar{C})(C+D)$$ “或-与”表达式

$$= \overline{\overline{(A+\bar{C})} + \overline{(C+D)}}$$ “或非 - 或非” 表达式

$$= \overline{\bar{A}C + \bar{C}\bar{D}}$$ “与-或-非”表达式

2.3.2 逻辑函数的代数化简法

1、逻辑函数的化简

化简的主要方法：

1．公式法（代数法）

2．图解法（卡诺图法）

代数化简法：

运用逻辑代数的基本定律和恒等式进行化简的方法。

并项法: $A+\overline{A}=1$

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