数字系统设计

华东理工大学电子与通信工程系
主讲：木昌洪
Email：changhongmu@ecust.edu.cn

2.1 逻辑代数基础

逻辑代数（布尔代数）是按一定的逻辑关系进行运算的代数，是分析和设计数字电路的数学工具。

逻辑是指事物的因果关系，或者说条件和结果的关系，这些因果关系可以用逻辑运算来表示，也就是用逻辑代数来描述。

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。

第二章 数字逻辑基础

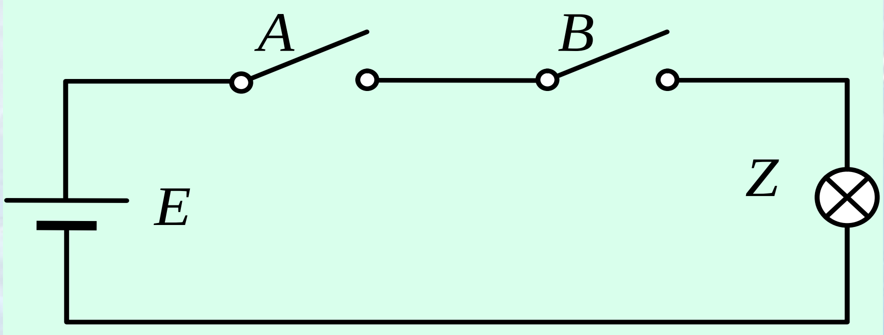
2.1.1. 逻辑变量和基本逻辑运算

1、与逻辑（与运算, AND）

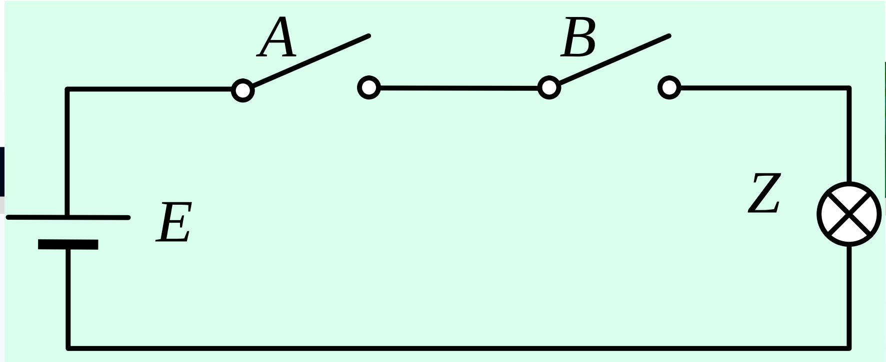
与逻辑的定义：仅当决定事件（Z）发生的所有条件（A，B，C，…）均满足时，事件（Z）才能发生。表达式为：

$$Z = A\ B\ C\ \ldots$$

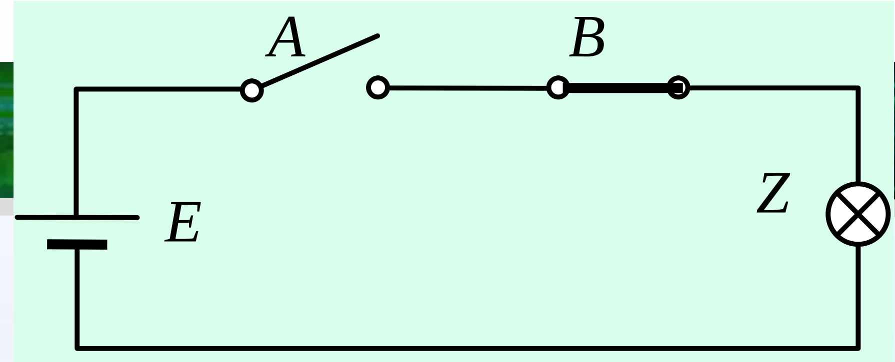
开关A，B串联控制灯泡Z



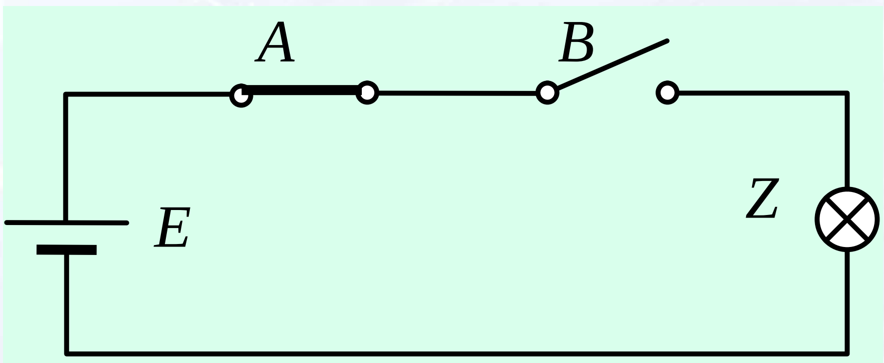
电路图



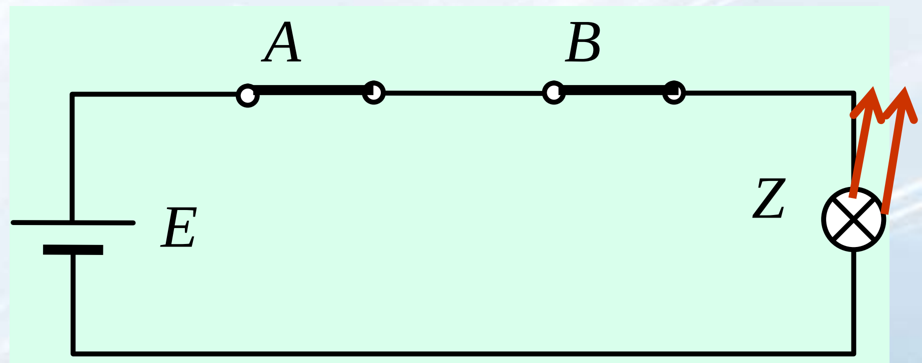
A、B都断开，灯不亮。



A断开、B接通，灯不亮。



A接通、B断开，灯不亮。



A、B都接通，灯亮。

两个开关必须同时接通，灯才亮。逻辑表达式为：

$$Z = A\ B$$

第二章 数字逻辑基础

1、与逻辑（与运算）

将开关接通记作1，断开记作0；灯亮记作1，灯灭记作0。可以作出如下表格来描述与逻辑关系：

功能表

开关 *A*	开关 *B*	灯 *Z*
断开	断开	灭
断开	闭合	灭
闭合	断开	灭
闭合	闭合	亮

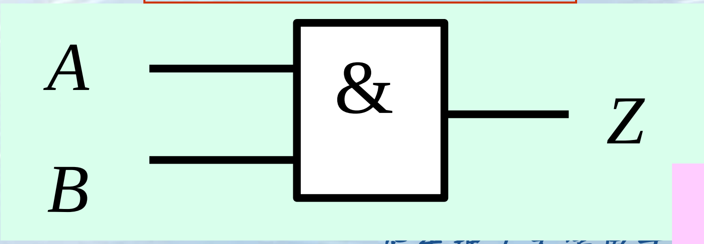
真值表

A	*B*	*Z*
0	0	0
0	1	0
1	0	0
1	1	1

这种把所有可能的条件组合及其对应结果一一列出来的表格叫做真值表。

实现与逻辑的电路称为与门。与门的逻辑符号：

逻辑符号



Z= A B

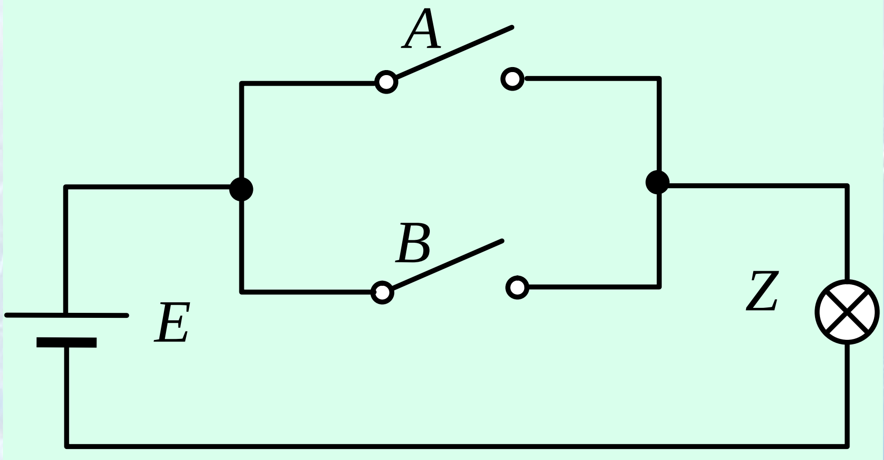
第二章 数字逻辑基础

2、或逻辑（或运算，OR）

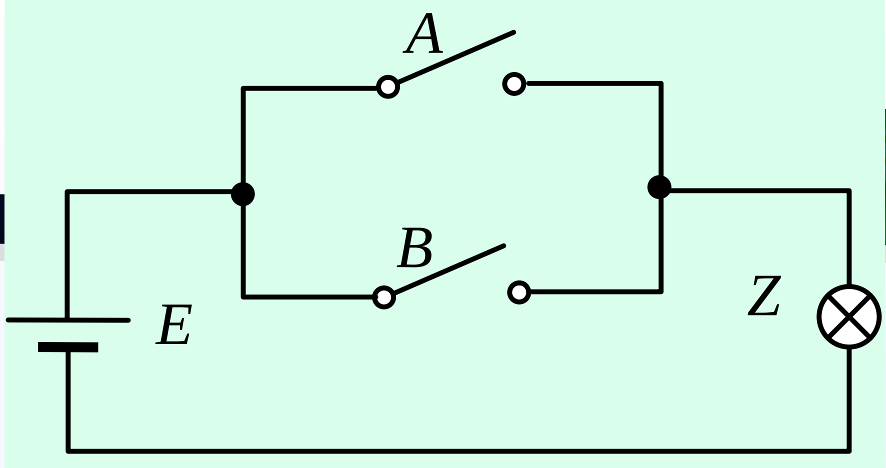
或逻辑的定义：当决定事件（Z）发生的各种条件（A，B，C，...)中，只要有一个或多个条件具备，事件（Z）就发生。表达式为：

$$Z = A + B + C + \ldots$$

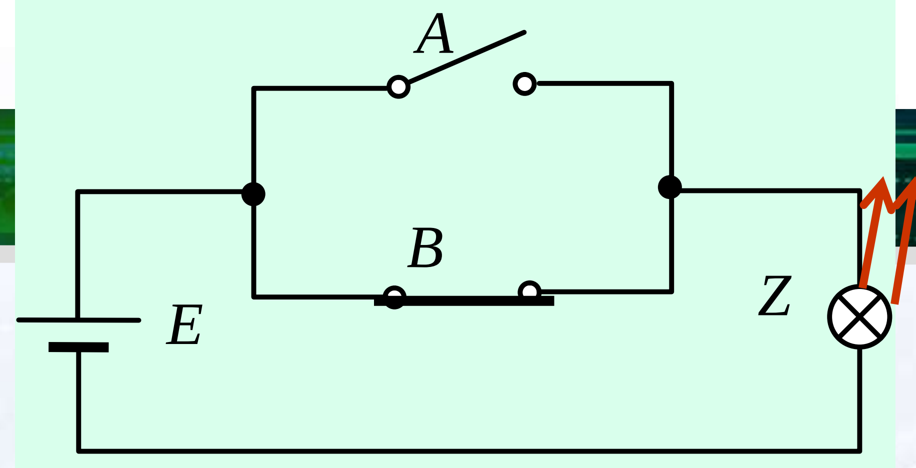
开关A，B并联控制灯泡Z



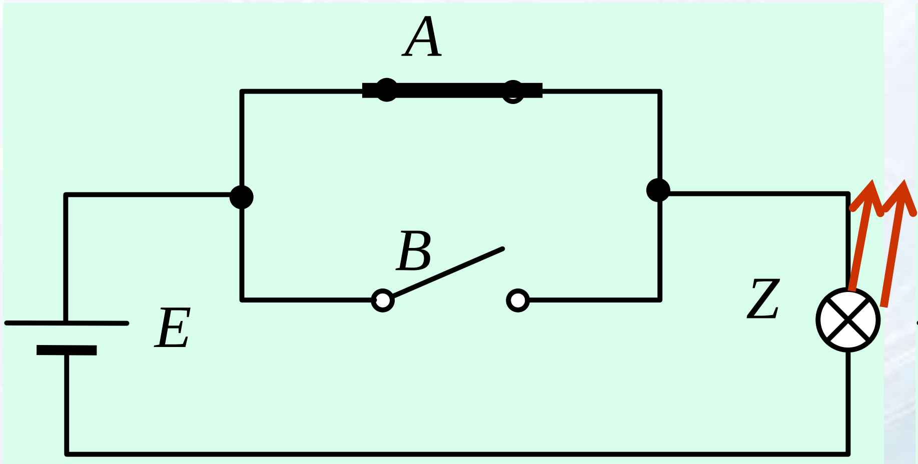
电路图



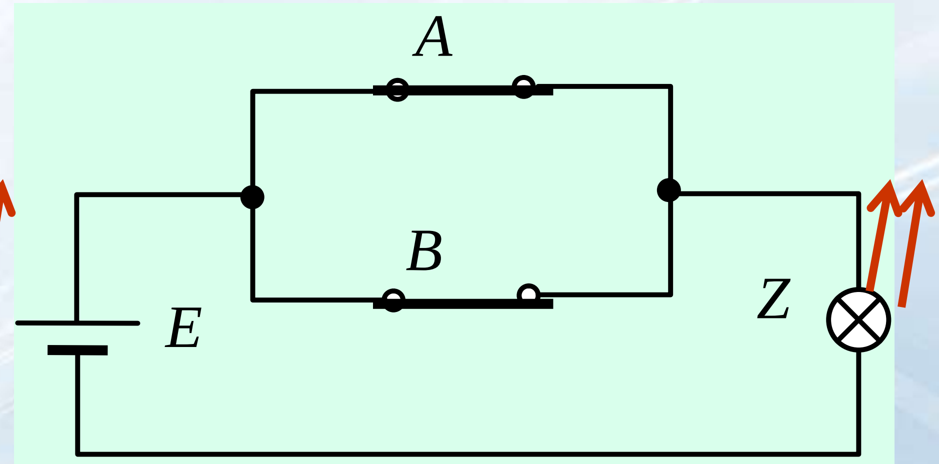
A、B都断开，灯不亮。



A断开、B接通，灯亮。



A接通、B断开，灯亮。



A、B都接通，灯亮。

两个开关只要有一个接通，灯就会亮。逻辑表达式为：

$$Z = A + B$$

2、或逻辑（或运算）

功能表

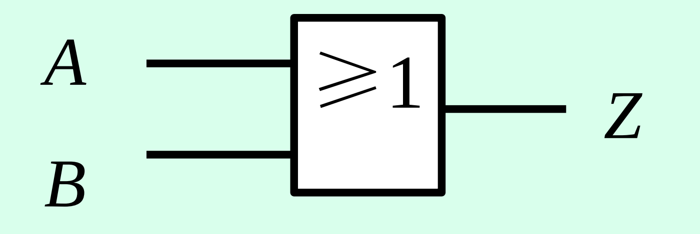
开关 A	开关 B	灯 Z
断开	断开	灭
断开	闭合	亮
闭合	断开	亮
闭合	闭合	亮

真值表

A	B	Z
0	0	0
0	1	1
1	0	1
1	1	1

逻辑符号

实现或逻辑的电路称为或门。或门的逻辑符号：



$Z=A+B$

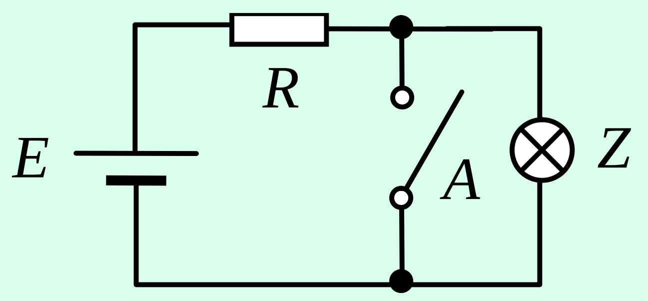
第二章 数字逻辑基础

3、非逻辑（非运算, NOT）

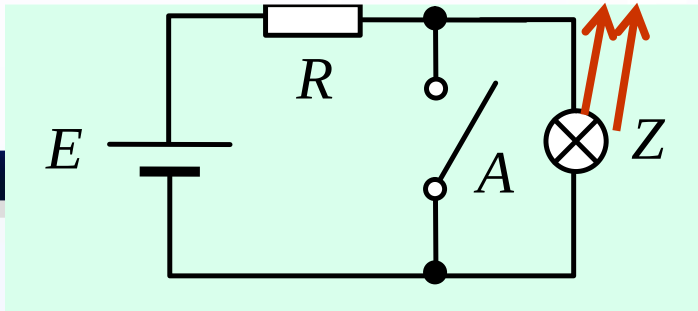
非逻辑指的是逻辑的否定。当决定事件（Z）发生的条件（A）满足时，事件不发生；条件不满足，事件反而发生。表达式为：

$$Z = \overline{A}$$

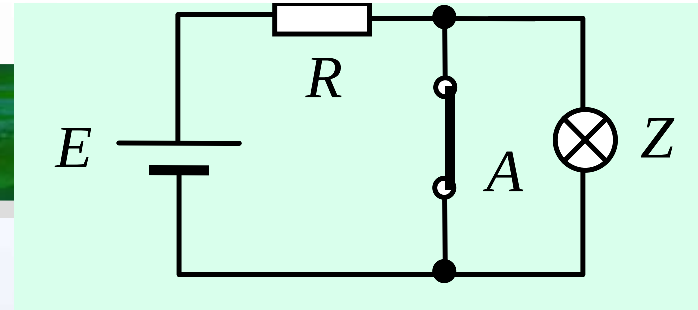
开关A控制灯泡Z



电路图



A断开，灯亮。



A接通，灯灭。

功能表

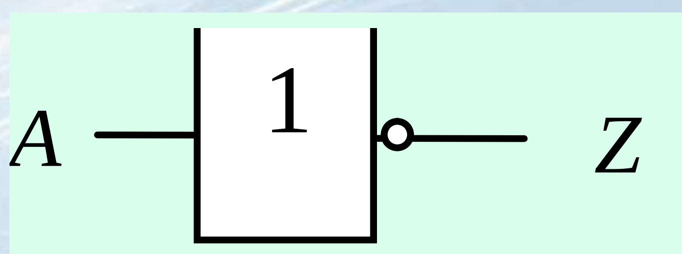
开关 A	灯 Z
断开	亮
闭合	灭

真值表

A	Z
0	1
1	0

逻辑符号

实现非逻辑的电路称为非门。非门的逻辑符号：



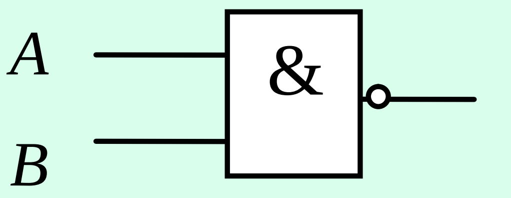
$Z=\overline{A}$

4、几种导出的逻辑运算

（1）与非运算：逻辑表达式为： $Z = \overline{AB}$

真值表

A	B	Z
0	0	1
0	1	1
1	0	1
1	1	0

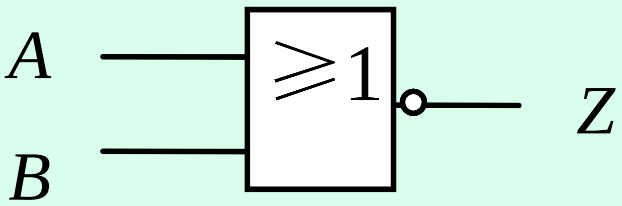


与非门的逻辑符号

（2）或非运算：逻辑表达式为： $Z = \overline{A + B}$

真值表

A	B	Z
0	0	1
0	1	0
1	0	0
1	1	0



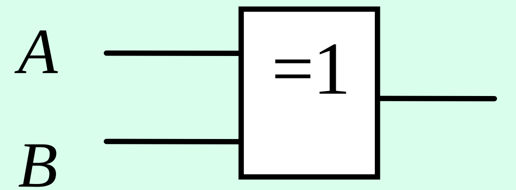
或非门的逻辑符号

4、几种导出的逻辑运算

（3）异或运算：逻辑表达式为：$Z=\bar{A}B+A\bar{B}=A\oplus B$

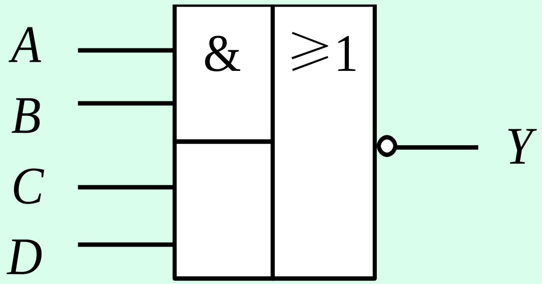
真值表

A	B	Z
0	0	0
0	1	1
1	0	1
1	1	0

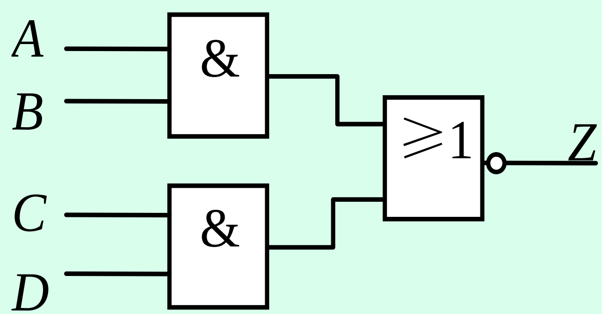


异或门的逻辑符号

（4）与或非运算：逻辑表达式为：$Z=\overline{AB+CD}$



与或非门的逻辑符号



与或非门的等效电路

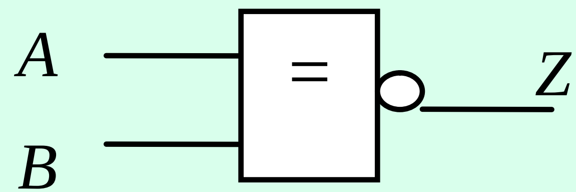
4、几种导出的逻辑运算

（5）同或运算：逻辑表达式为：

$$Z = AB + \overline{A}\,\overline{B} = A\Theta B$$

真值表

A	B	Z
0	0	1
0	1	0
1	0	0
1	1	1



同或门的逻辑符号

2.2.2 逻辑代数的基本规则

1、逻辑函数的建立：

● 逻辑表达式：由逻辑变量和与、或、非3种运算符连接起来所构成的式子。

输入逻辑变量：等式右边的字母A、B、C、D → *自变量*

输出逻辑变量：等式左边的字母Z → *因变量*

● 逻辑函数：如果对应于输入逻辑变量A、B、C、…的每一组确定值，输出逻辑变量Z就有唯一确定的值，则称Z是A、B、C、…的逻辑函数。记为

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

注意：与普通代数不同的是，在逻辑代数中，不管是变量还是函数，其取值都只能是0或1，并且这里的0和1只表示两种不同的状态，没有数量的含义。

2.2.2 逻辑代数的基本定律和规则

1.基本定律

名　称	公 式 1	公 式 2
1—0律	$A\cdot 1=A$ $A\cdot 0=0$ $A\overline{A}=0$	$A+0=A$ $A+1=1$ $A+\overline{A}=1$
还原律	$\overline{\overline{A}}=A$	否定之否定规律
同一律	$A\cdot A=A$	$A+A=A$
交换律	$A\cdot B=B\cdot A$	$A+B=B+A$
结合律	$A(BC)=(AB)C$	$A+(B+C)=(A+B)+C$
分配律	$A(B+C)=AB+AC$	$A+(BC)=(A+B)(A+C)$
反演律（摩根定律）	$\overline{AB}=\overline{A}+\overline{B}$	$\overline{A+B}=\overline{A}\ \overline{B}$
吸收律	$A(A+B)=A$ $A(\overline{A}+B)=AB$	$A+AB=A$ $A+\overline{A}B=A+B$
附加律	$AB+\overline{A}C+BC=AB+\overline{A}C$	

利用真值表很容易证明这些公式的正确性

用基本定律相互证明

❖ 吸收律 $A+\overline{A}B=A+B$

证明：$A+\overline{A}B=(A+\overline{A})(A+B)$

$=1\cdot(A+B)$

$=A+B$

❖ 附加律 $AB+\overline{A}C+BC=AB+\overline{A}C$

证明：$AB+\overline{A}C+BC$

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

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

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

$=AB+\overline{A}C$

2.2.2 逻辑代数的基本定律和规则

2.三个重要规则

（1）代入规则：任何一个含有变量A的等式，如果将所有出现A的位置都用同一个逻辑函数代替，则等式仍然成立。这个规则称为代入规则。

例如，已知等式 $\overline{AB} = \overline{A} + \overline{B}$ ，用函数Z=AC代替等式中的A，根据代入规则，等式仍然成立，即有：

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

2.2.2 逻辑代数的基本定律和规则

2.三个重要规则

（2）反演规则：对于任何一个逻辑表达式Z，如果将表达式中的所有“·”换成“+”，“+”换成“·”，“0”换成“1”，“1”换成“0”，原变量换成反变量，反变量换成原变量，那么所得到的表达式就是函数Z的反函数Z（或称补函数）。这个规则称为反演规则。例如：

$$Z = A\overline{B} + C\overline{D}E \quad \longrightarrow \quad \overline{Z} = (\overline{A} + B)(\overline{C} + D + \overline{E})$$

$$Z = A + \overline{B + \overline{C} + \overline{D + \overline{E}}} \quad \longrightarrow \quad \overline{Z} = \overline{A} \cdot \overline{\overline{B} \cdot C \cdot \overline{\overline{D} \cdot E}}$$

2.2.2 逻辑代数的基本定律和规则

2.三个重要规则

（3）对偶规则：对于任何一个逻辑表达式Z，如果将表达式中的所有“·”换成“＋”，“＋”换成“·”，“0”换成“1”，“1”换成“0”，而变量保持不变，则可得到的一个新的函数表达式Z′，Z′称为函Z的对偶函数。这个规则称为对偶规则。例如：

$$Z = A\overline{B} + C\overline{D}E \rightarrow Z' = (A+\overline{B})(C+\overline{D}+E)$$

$$Z = A + \overline{B + \overline{C} + \overline{\overline{D + \overline{E}}}} \rightarrow Z' = A \cdot \overline{B \cdot \overline{C} \cdot \overline{\overline{D \cdot \overline{E}}}}$$

2.2.2 逻辑代数的基本定律和规则

2.三个重要规则

（3）对偶规则

对偶规则的意义在于：如果两个函数相等，则它们的对偶函数也相等。利用对偶规则,可以使要证明及要记忆的公式数目减少一半。例如：

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

$$A(B+C)=AB+AC \quad\longrightarrow\quad A+BC=(A+B)(A+C)$$

注意：在运用反演规则和对偶规则时，必须按照逻辑运算的优先顺序进行：先算括号，接着与运算，然后或运算，最后非运算，否则容易出错。

2. 2. 4 辑函逻数的表示方法

五种表示方法

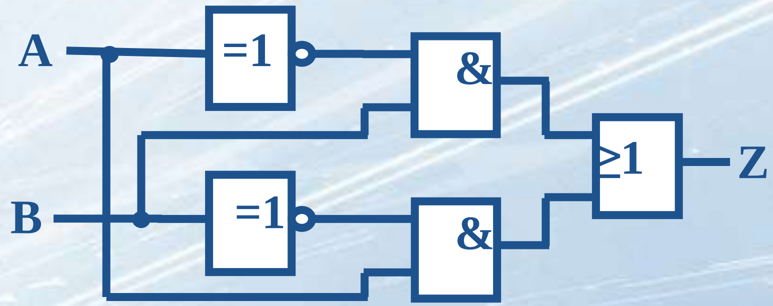
真值表：将逻辑函数输入变量取值的不同组合与所对应的输出变量值用列表的方式一一对应列出的表格。

n个输入变量 ⟶ 2^n 种组合。

逻辑代数式 (逻辑表示式，逻辑函数式)

$$Z = A\bar{B} + \bar{A}B$$

逻辑电路图：



波形图

卡诺图

2.2.4 逻辑函数的表示方法

1、真值表

将输入、输出的所有可能状态一一对应地列出。n个变量可以有2^n个输入状态。

列真值表的方法：一般按二进制的顺序，输出与输入状态一一对应，列出所有可能的状态。

例：设计一个表决(少数服从多数)电路，即当 ABC中两个以上为1时Z为1（*同教材例2.2*）

A	B	C	Z
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

2.2.4 逻辑函数的表示方法

2、逻辑函数式

把逻辑函数的输入、输出关系写成与、或、非等逻辑运算的组合式。也称为逻辑函数式，通常采用“与或”的形式。

- 由真值表写出逻辑函数式：将因变量为1时，自变量与或即可。

例：设计一个表决电路，当ABC中两个以上为1时Z为1。

逻辑函数式为——

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

A	B	C	Z
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

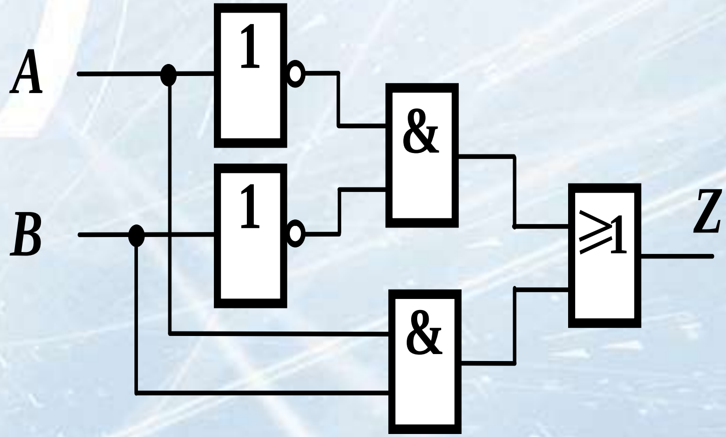
2.2.4 逻辑函数的表示方法

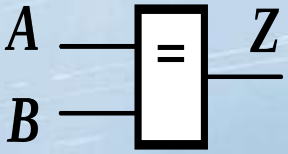
3、逻辑图

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为 $Z=\overline{A}\,\overline{B}+AB$，试画出其逻辑图





2.2.4 逻辑函数的表示方法

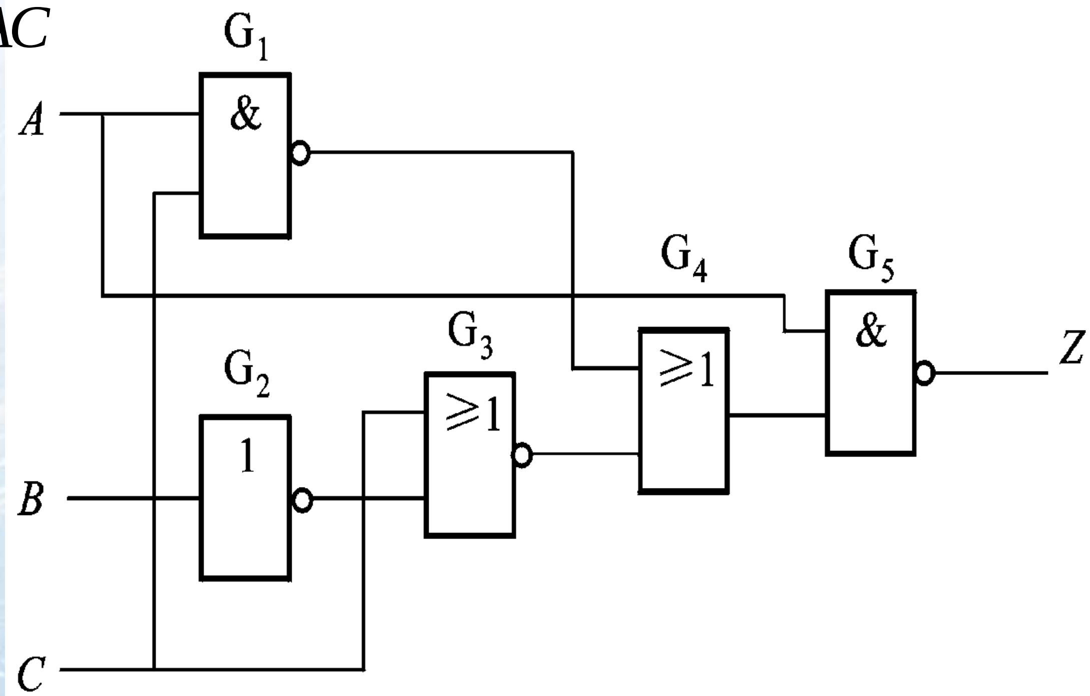
已知逻辑图求逻辑函数式和真值表

例如：写出右图所示逻辑图的逻辑函数式。

解：首先从输入端门电路开始，逐级给每个门标号（**G1~G5**），然后依次写出各个门的输出端函数表达式，分别为：

$\overline{AC}$ $\overline{B}$ $\overline{\overline{B}+C}$ $\overline{\overline{B}+C}+\overline{AC}$

$\overline{(\overline{\overline{B}+C}+\overline{AC})\bullet A}$



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

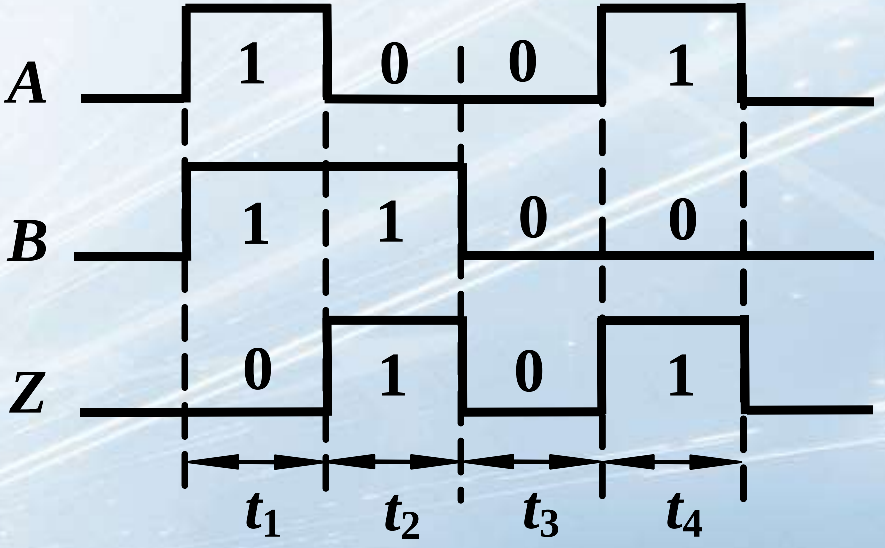
2.2.4 逻辑函数的表示方法

4、波形图

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	Z
0	0	1
0	1	0
1	0	0
1	1	1



2.2.7 逻辑函数的公式化简法

1. 化简的意义与标准

逻辑函数化简的意义：逻辑表达式越简单，实现它的电路越简单，电路工作越稳定可靠。

一个逻辑函数的表达式可以有以下5种表示形式。

（1）与或表达式： $Z=\bar{A}B+AC$

（2）或与表达式： $Z=(A+B)(\bar{A}+C)$

（3）与非-与非表达式： $Z=\overline{\overline{\bar{A}B}\cdot\overline{AC}}$

（4）或非-或非表达式： $Z=\overline{\overline{A+B}+\overline{\bar{A}+C}}$

（5）与或非表达式： $Z=\overline{\bar{A}\bar{B}+A\bar{C}}$

2.2.7 逻辑函数的公式化简法

1. 化简的意义与标准

利用逻辑代数的基本定律，可以实现上述五种逻辑函数式之间的变换。

逻辑函数的最简： 与—或式

（1）乘积项个数最少；

（2）每个乘积项中的变量个数也最少。

$$Z=\overline{A}B\overline{E}+\overline{A}B+A\overline{C}+A\overline{C}E+B\overline{C}+B\overline{C}D$$

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

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

最简与或表达式

2.2.7 逻辑函数的公式化简法

1. 化简方法 (1)并项法

逻辑函数的公式化简法就是运用逻辑代数的基本公式、定理和规则来化简逻辑函数。

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

运用分配律

$$Z_1 = ABC + \overline{A}BC + B\overline{C} = (A+\overline{A})BC + B\overline{C}$$
$$= BC + B\overline{C} = B(C+\overline{C}) = B$$

$$Z_2 = ABC + A\overline{B} + A\overline{C} = ABC + A(\overline{B}+\overline{C})$$
$$= ABC + A\overline{BC} = A(BC+\overline{BC}) = A$$

运用摩根定律

若两个乘积项中分别包含同一个因子的原变量和反变量，而其他因子都相同时，则这两项可以合并成一项，并消去互为反变量的因子。

(2) 吸收法

（１）利用公式Ａ＋ＡＢ＝Ａ，消去多余的项。

$$Z_1 = \overline{A}B + \overline{A}BCD(E+F) = \overline{A}B$$

$$Z_2 = A + \overline{\overline{B} + \overline{CD}} + \overline{\overline{AD}\,\overline{B}} = A + BCD + AD + B$$
$$= (A + AD) + (B + BCD) = A + B$$

如果乘积项是另外一个乘积项的因子，则这另外一个乘积项是多余的。

（２）利用公式Ａ＋$\overline{\text{Ａ}}$Ｂ＝Ａ＋Ｂ，消去多余的变量。

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

$$Z = A\overline{B} + C + \overline{A}\,\overline{C}D + B\overline{C}D$$
$$= A\overline{B} + C + \cancel{\overline{C}}(\overline{A} + B)D$$
$$= A\overline{B} + C + (\overline{A} + B)D$$
$$= A\overline{B} + C + \cancel{\overline{A\overline{B}}}D$$
$$= A\overline{B} + C + D$$

如果一个乘积项的反是另一个乘积项的因子，则这个因子是多余的。

（3）配项法

（1）利用公式A＝A（B＋B），为某一项配上其所缺的变量，以便用其它方法进行化简。

$$
\begin{aligned}
Z &= A\overline{B} + B\overline{C} + \overline{B}C + \overline{A}B \\
&= A\overline{B} + B\overline{C} + \underline{(A+\overline{A})}\overline{B}C + \overline{A}B\underline{(C+\overline{C})} \\
&= A\overline{B} + B\overline{C} + A\overline{B}C + \overline{A}\overline{B}C + \overline{A}BC + \overline{A}B\overline{C} \\
&= A\overline{B}(1+C) + B\overline{C}(1+\overline{A}) + \overline{A}C(\overline{B}+B) \\
&= A\overline{B} + B\overline{C} + \overline{A}C
\end{aligned}
$$

（2）利用公式A＋A＝A，为某项配上其所能合并的项。

$$
\begin{aligned}
Z &= ABC + AB\overline{C} + A\overline{B}C + \overline{A}BC \\
&= (ABC + AB\overline{C}) + (\underline{ABC} + A\overline{B}C) + (\underline{ABC} + \overline{A}BC) \\
&= AB + AC + BC
\end{aligned}
$$

2.2.7 逻辑函数的公式化简法

（4）消去冗余项法

利用附加律 $AB+\overline{A}C+BC=AB+\overline{A}C$，将冗余项 BC 消去。

$$Z_1 = A\overline{B}+AC+ADE+\overline{C}D$$
$$= A\overline{B}+(AC+\overline{C}D+\cancel{ADE})$$
$$= A\overline{B}+AC+\overline{C}D$$

$$Z_2 = AB+\overline{B}C+\cancel{AC(DE+FG)}$$
$$= AB+\overline{B}C$$

2.2.7 逻辑函数的公式化简法

例：化简函数

$$Z=(\overline{B}+D)(\overline{B}+D+A+G)(C+E)(\overline{C}+G)(A+E+G)$$

解：①先求出Z的对偶函数Z′，并对其进行化简。

$$Z'=\overline{B}D+\cancel{\overline{B}DAG}+CE+\overline{C}G+\cancel{AEG}$$
$$=\overline{B}D+CE+\overline{C}G$$

②求Z′的对偶函数，便得Z的最简或与表达式。

$$Z=(\overline{B}+D)(C+E)(\overline{C}+G)$$

2.2.8 逻辑函数的卡诺图化简法

1. 最小项与卡诺图

(1) 逻辑函数的最小项及其性质

最小项：如果一个函数的某个乘积项包含了函数的全部变量，其中每个变量都以原变量或反变量的形式出现，且仅出现一次，则这个乘积项称为该函数的一个标准积项，通常称为最小项。

例如：3个变量A、B、C可组成8个最小项：

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

2.2.8 逻辑函数的卡诺图化简法

1. 最小项与卡诺图

(2) 最小项的表示方法

最小项的表示方法：通常用符号m_i来表示最小项。下标i的确定：把最小项中的原变量记为1，反变量记为0，当变量顺序确定后，可以按顺序排列成一个二进制数，则与这个二进制数相对应的十进制数，就是这个最小项的下标i。

例如：3个变量A、B、C的8个最小项可以分别表示为：

$$m_0 = \overline{A}\,\overline{B}\,\overline{C}、m_1 = \overline{A}\,\overline{B}C、m_2 = \overline{A}B\overline{C}、m_3 = \overline{A}BC$$

$$m_4 = A\overline{B}\,\overline{C}、m_5 = A\overline{B}C、m_6 = AB\overline{C}、m_7 = ABC$$

（3）最小项的性质：

3变量全部最小项的真值表

A	B	C	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7
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

①任意一个最小项，只有一组变量取值使其值为1。

②任意两个不同的最小项的乘积必为0。

③全部最小项的和必为1。

2、逻辑函数的最小项表达式

任何一个逻辑函数都可以表示成唯一的一组最小项之和，称为标准与或表达式，也称为最小项表达式

对于不是最小项表达式的与或表达式，可利用公式A＋$\overline{A}$＝1和A(B+C)＝AB＋BC来配项展开成最小项表达式。

$$
\begin{aligned}
Z &= \overline{A} + BC \\
&= \overline{A}(B+\overline{B})(C+\overline{C}) + (A+\overline{A})BC \\
&= \overline{A}BC + \overline{A}B\overline{C} + \overline{A}\,\overline{B}C + \overline{A}\,\overline{B}\,\overline{C} + ABC + \overline{A}BC \\
&= \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + \overline{A}BC + ABC \\
&= m_0 + m_1 + m_2 + m_3 + m_7 \\
&= \sum{}_m (0,1,2,3,7)
\end{aligned}
$$

如果列出了函数的真值表，则只要将**函数值为1的那些最小项相加**，便是函数的最小项表达式。

A	B	C	Z	最小项
0	0	0	0	m_0
0	0	1	1	m_1
0	1	0	1	m_2
0	1	1	1	m_3
1	0	0	0	m_4
1	0	1	1	m_5
1	1	0	0	m_6
1	1	1	0	m_7

$\mathbf{m_1 = \overline{A}\,\overline{B}C}$

$\mathbf{m_2 = \overline{A}B\overline{C}}$

$\mathbf{m_3 = A\overline{B}\,\overline{C}}$

$\mathbf{m_5 = A\overline{B}C}$

$$Z = m_1 + m_2 + m_3 + m_5 = \sum{}_m(1,2,3,5)$$
$$= \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\,\overline{C} + A\overline{B}C$$

将真值表中函数值为0的那些最小项相加，便可得到反函数的最小项表达式。

3、卡诺图

（1）卡诺图的构成

将逻辑函数真值表中的最小项重新排列成矩阵形式，并且使**矩阵的横方向和纵方向的逻辑变量的取值按照格雷码的顺序排列**，这样构成的图形就是卡诺图。

每个2变量的最小项有两个最小项与它相邻。

A \ B	0	1
0	m_0	m_1
1	m_2	m_3

2 变量卡诺图

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

3 变量卡诺图

每个3变量的最小项有3个最小项与它相邻。

卡诺图的特点是任意两个相邻的最小项在图中也是相邻的。

（相邻项是指两个最小项只有一个因子互为反变量，其余因子均相同，又称为逻辑相邻项）。

每个4变量的最小项有4个最小项与它相邻

最左列的最小项与最右列的相应最小项也是相邻的。

$AB \backslash CD$	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}

4 变量卡诺图

最上面一行的最小项与最下面一行的相应最小项也是相邻的。

两个相邻最小项可以合并消去一个变量

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

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

逻辑函数化简的实质就是相邻最小项的合并

（2）用卡诺图表示逻辑函数

◆ 逻辑函数是以**真值表或者以最小项表达式**给出：在卡诺图上那些与给定**逻辑函数的最小项相对应的方格内填入1，其余的方格内填入0**。

$$Z(A,B,C,D)=\sum_m(1,3,4,6,7,11,14,15)$$

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

m_3 m_1 m_6 m_4 m_7 m_{14} m_{15} m_{11}

◆　逻辑函数以**一般的逻辑表达式**给出：先将函数变换为与或表达式（不必变换为最小项之和的形式），然后在卡诺图上与每一个乘积项所包含的那些最小项（该乘积项就是这些最小项的公因子）相对应的方格内填入1，其余的方格内填入0。

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

变换为与或表达式

$$Z=\overline{A}\,\overline{D}+\overline{B}C$$

$\overline{A}\,\overline{D}$的公因子

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

$\overline{B}C$的公因子

说明：如果求得了函数Z的反函数$\overline{Z}$，则对$\overline{Z}$中所包含的各个最小项，在卡诺图相应方格内填入0，其余方格内填入1。

(3) 卡诺图的性质

● 任何两个（2^1个）标1的相邻最小项，可以合并为一项，并消去一个变量（消去互为反变量的因子，保留公因子）。

$\cancel{\bar{A}}\bar{B}\bar{C} + \cancel{A}\bar{B}\bar{C}$

$= \bar{B}\bar{C}$

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

$\cancel{\bar{A}}BC + \cancel{A}BC = BC$

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

$A\bar{B}\cancel{C}D + A\bar{B}\cancel{\bar{C}}D$

$= A\bar{B}D$

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

(3) 卡诺图的性质

- 任何4个（2^2个）标1的相邻最小项，可以合并为一项，并消去2个变量。

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

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

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

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

$\overline{A}B$

$\overline{C}D$

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

$B\bar{D}$

$\bar{B}D$

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

BD

$\bar{B}\bar{D}$

(3) 卡诺图的性质

- 任何8个（2^3个）标1的相邻最小项，可以合并为一项，并消去3个变量。

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

B

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

$\overline{\mathbf{D}}$

小结：相邻最小项的数目必须为2^n 个才能合并为一项，并消去 n个变量。包含的最小项数目越多，即由这些最小项所形成的圈越大，消去的变量也就越多，从而所得到的逻辑表达式就越简单。这就是利用卡诺图化简逻辑函数的基本原理。

(4) 卡诺图化简的基本步骤

逻辑表达式或真值表

$$Z(A,B,C,D)=\sum m(3,5,7,8,11,12,13,15)$$

1

卡诺图

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

①圈越大越好，但每个圈中标 1 的方格数目必须为 2^i 个。②同一个方格可同时画在几个圈内，但每个圈都要有新的方格，否则它就是多余的。③不能漏掉任何一个标 1 的方格。

2

2

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

CD

BD

冗余项

$A\overline{C}\,\overline{D}$

合并最小项

3

最简与或表达式

3

将代表每个圈的乘积项相加

$$Z(A,B,C,D) = BD + CD + A\overline{C}\,\overline{D}$$

两点说明：

① 在有些情况下，最小项的圈法不只一种，得到的各个乘积项组成的与或表达式各不相同，哪个是最简的，要经过比较、检查才能确定。

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

$$\bar{A}\bar{B}\bar{C} + \bar{A}BD + \bar{A}C\bar{D} + BC$$

不是最简

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

$$\bar{A}\bar{B}\bar{D} + \bar{A}\bar{C}D + BC$$

最简

两点说明：

② 在有些情况下，不同圈法得到的与或表达式都是最简形式。即**一个函数的最简与或表达式不是唯一的。**

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

$$\bar{A}\bar{C} + \bar{B}\bar{C}\bar{D} + ACD + \bar{A}BD$$

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

$$\bar{A}\bar{C} + \bar{B}\bar{C}\bar{D} + ACD + BCD$$

2.2.9 具有约束项的逻辑函数的化简

1、含约束项的逻辑函数

约束项：函数可以任意取值（可以为0，也可以为1）或不会出现的变量取值所对应的最小项称为约束项，也叫做任意项或无关项。

例如：判断一位十进制数是否为奇数。

A B C D	Z	A B C D	Z	说 明
0 0 0 0	0	1 0 0 0	0	
0 0 0 1	1	1 0 0 1	1	
0 0 1 0	0	1 0 1 0	×	不会出现
0 0 1 1	1	1 0 1 1	×	不会出现
0 1 0 0	0	1 1 0 0	×	不会出现
0 1 0 1	1	1 1 0 1	×	不会出现
0 1 1 0	0	1 1 1 0	×	不会出现
0 1 1 1	1	1 1 1 1	×	不会出现

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

输入变量A，B，C，D取值为0000～1001时，逻辑函数Z有确定的值，根据题意，偶数时为0，奇数时为1。

$$Z(A,B,C,D)=\Sigma_m(1,3,5,7,9)$$

A，B，C，D取值为1010～1111的情况不会出现或不允许出现，对应的最小项属于约束项。用符号“φ”、“×”或“d”表示。约束项之和构成的逻辑表达式叫做约束条件或任意条件，用一个值恒为0的条件等式表示。

$$\Sigma_d(10,11,12,13,14,15)=0$$

含有约束条件的逻辑函数可以表示成如下形式：

$$Z(A,B,C,D)=\Sigma_m(1,3,5,7,9)+\Sigma_d(10,11,12,13,14,15)$$

2、含任意项的逻辑函数的化简

在逻辑函数的化简中，充分利用约束项可以得到更加简单的逻辑表达式，因而其相应的逻辑电路也更简单。**在化简过程中，约束项的取值可视具体情况取0或取1。具体地讲，如果约束项对化简有利，则取1；如果约束项对化简不利，则取0。**

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

不利用约束项的化简结果为：

$$Z=\overline{A}D+\overline{B}\,\overline{C}D$$

利用约束项的化简结果为：

$$Z=D$$

第二章 数字逻辑基础

课后作业：

2.5,3,4,5,9,10,11,16,17,21

Thank You !