《数字电路与微机系统》

《数字电路》部分

王志

南京大学

一. 逻辑代数的基本概念、公式和定理

1. 基本和常用逻辑运算
2. 公式和定理

二. 逻辑函数的化简方法

1. 逻辑函数的标准与或式和最简式
2. 逻辑函数的公式化简法
3. 逻辑函数的图形化简法
4. 具有约束的逻辑函数的化简

三. 逻辑函数的表示方法及其相互之间的转换

1. 几种表示逻辑函数的方法
2. 几种表示方法之间的转换
3. 用Multisim进行逻辑函数的化简与变换

1.1 基本和常用逻辑运算

- 三种基本逻辑运算
  - 与（and）
  - 或（or）
  - 非（not）

一、三种基本逻辑运算

1. 基本逻辑关系举例

（1）电路图

图 1.1.1 中所示电路，是反映**与**、**或**、**非**三种基本逻辑关系最简单的例子。

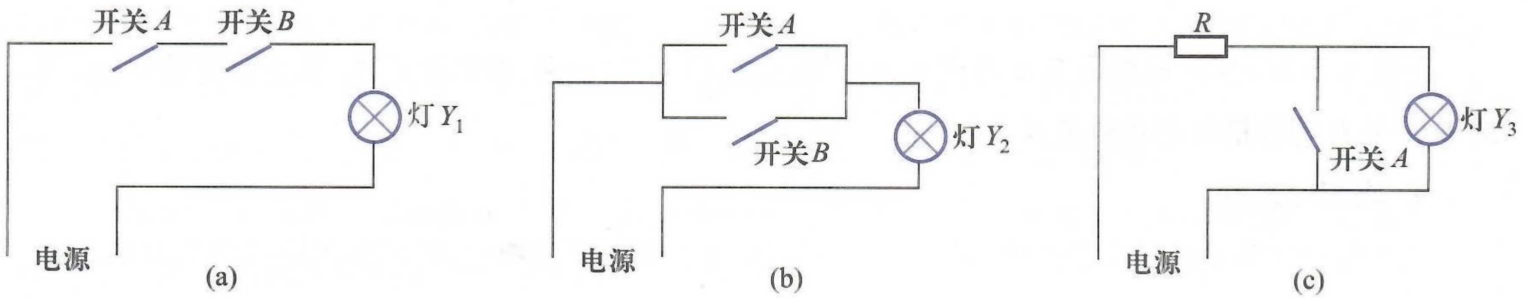


图 1.1.1　基本逻辑关系电路举例

（a）与逻辑关系　（b）或逻辑关系　（c）非逻辑关系

根据电路中的有关定理，可以很容易地列出表 1.1.1 所示的功能表。

表 1.1.1　图 1.1.1 所示电路的功能表

开关 A	开关 B	灯 Y_1	灯 Y_2	灯 Y_3
断开	断开	灭	灭	亮
断开	闭合	灭	亮	亮
闭合	断开	灭	亮	灭
闭合	闭合	亮	亮	灭

1.1 基本和常用逻辑运算

- 逻辑真值表，简称真值表
  - 经过设定变量和状态赋值之后，便可以得到反映开关状态与点灯亮灭之间因果关系的数学表达式
  - 设定变量：用英文字母表示开关和电灯的过程，例：开关A, B，灯 Y_1, Y_2, Y_3
  - 状态赋值：用0和1分别表示开关和点灯有关状态的过程，也称变量取值，例：用0表示开关断开和灯灭，用1表示开关闭合和灯亮
  - 列真值表

A	B	Y_1	Y_2	Y_3
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

1.1 基本和常用逻辑运算

- 与逻辑关系
  - 当决定一件事情的各个条件全部具备时，这件事情才会发生
  - 例：只有开关 A 与开关 B 都闭合时，灯 Y_1 才会亮；对于灯 Y_1 亮这件事来讲，开关 A、开关 B 闭合是与的逻辑关系
- 或逻辑关系
  - 当决定一件事情的各个条件中，只要有一个具备，这件事情就会发生
  - 例：只要开关 A 或开关 B 闭合，灯 Y_2 就会亮；对于灯 Y_2 亮这件事来讲，开关 A、开关 B 闭合是或的逻辑关系
- 非逻辑关系
  - 非就是反，就是否定
  - 例：当开关 A 断开时，灯 Y_3 亮，闭合时反而会灭；对于灯 Y_3 亮这件事来讲，开关闭合是一种非的逻辑关系

基本逻辑运算

- 与运算：$Y_1 = A \cdot B$
  - 读作“Y_1 等于 A 与 B”
  - 与运算和算术中的乘法运算很相似，又称逻辑乘法运算，“Y_1 等于A 乘 B”
  - 书写时表示与或者乘的符号 $\cdot$ 常省略
- 或运算：$Y_2 = A + B$
  - 逻辑加法运算，读作“Y_2 等于A 或 B”，“Y_2 等于A 加 B”
- 非运算：$Y_3 = \bar{A}$
  - 逻辑非运算，逻辑反运算，读作“Y_3 等于A 非”，“Y_3 等于A 反”

逻辑变量与逻辑函数

- 逻辑变量
  - 取值十分简单，在二值逻辑中，只有0和1两个取值
  - 0和1没有数值大小，所表示的是事物相互对立又联系着的两个方面，即两个状态

- 逻辑函数
  - 输入逻辑变量 A, B，输出逻辑变量 Y_1, Y_2, Y_3
  - 如果输入逻辑变量的取值确定之后，输出逻辑变量的值也被唯一确定了，那么称 Y 是$A, B, \ldots$ 的逻辑函数
  - 一般情况下，常用真值表描述变量取值和函数值之间的对应关系，穷举法

$$Y = F(A, B, \ldots)$$

常见的逻辑运算

- 由与、或、非三种基本运算构成的一些复合运算

与非运算	$Y=\overline{A\cdot B}$
或非运算	$Y=\overline{A+B}$
与或非运算	$Y=\overline{A\cdot B+C\cdot D}$
异或运算	$Y=\bar{A}\cdot B+A\cdot\bar{B}$

基本和常用逻辑运算的逻辑符号

- 基本和常用逻辑运算的应用十分广泛，是构成各种复杂逻辑运算的基础
- 有实现这些运算的称之为门电路的逻辑电路存在，是组成各种数字电路的基本单元
- 由美国国家标准协会（ANSI）和电气与电子工程师协会（IEEE）于1984年制定，于1991年进行补充和修订

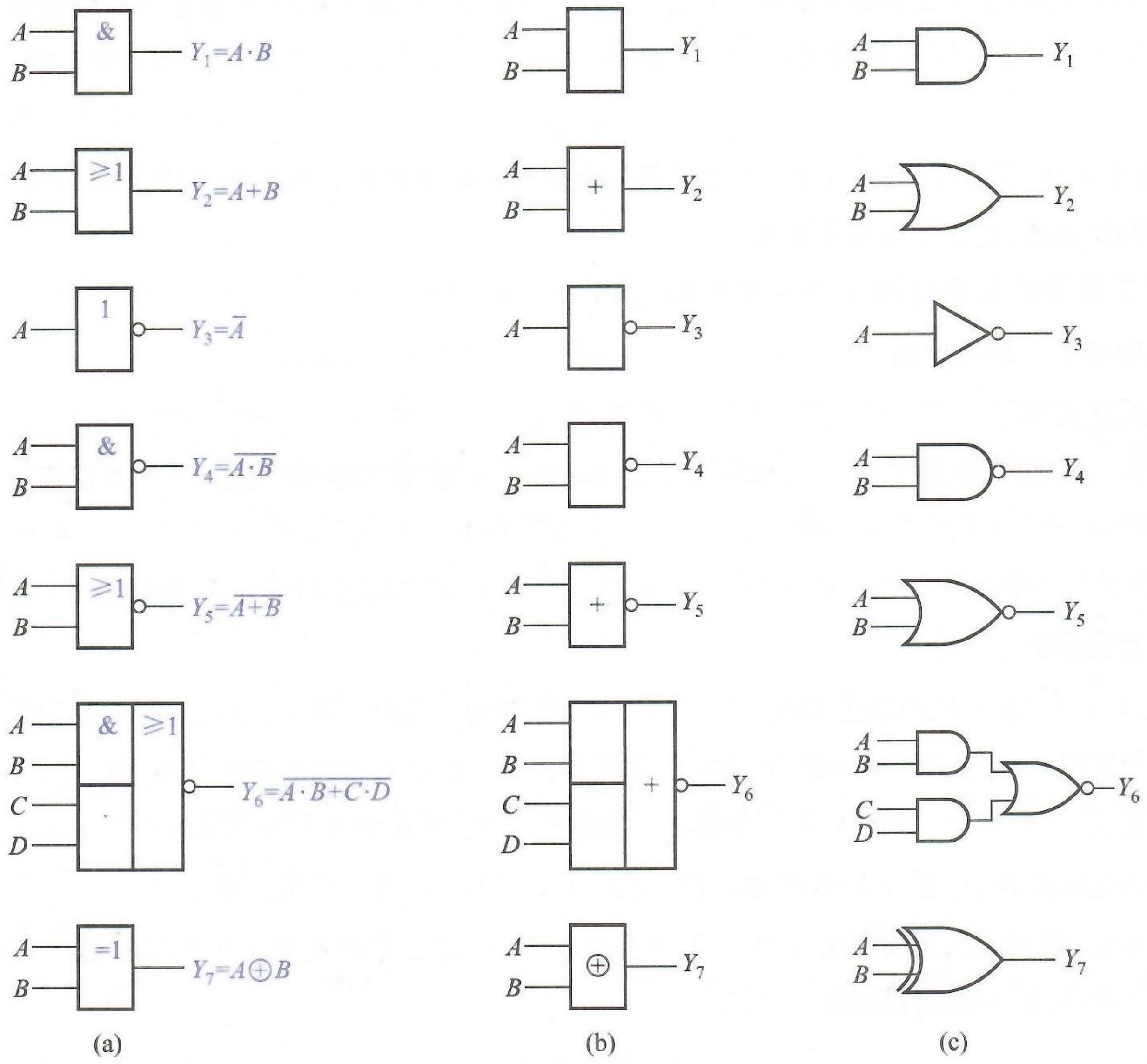


图 1.1.2 七种运算的逻辑符号

(a) 矩形符号 (b) 曾用符号 (c) 特定外形符号

一. 逻辑代数的基本概念、公式和定理

1. 基本和常用逻辑运算
2. 公式和定理

二. 逻辑函数的化简方法

1. 逻辑函数的标准与或式和最简式
2. 逻辑函数的公式化简法
3. 逻辑函数的图形化简法
4. 具有约束的逻辑函数的化简

三. 逻辑函数的表示方法及其相互之间的转换

1. 几种表示逻辑函数的方法
2. 几种表示方法之间的转换
3. 用Multisim进行逻辑函数的化简与变换

公式和定理

- 公理：常量之间的关系，逻辑代数中的基本运算规则
  - 人为规定的，即与逻辑思维的推理一致，又与人们已经习惯了的普通代数的运算规则相似

$0 \cdot 0 = 0$	$1 + 1 = 1$
$0 \cdot 1 = 0$	$1 + 0 = 1$
$1 \cdot 1 = 1$	$0 + 0 = 0$
$\bar{0} = 1$	$\bar{1} = 0$

公式和定理

- 变量和常量的关系

$A \cdot 1 = A$	$A + 0 = A$
$A \cdot 0 = 0$	$A + 1 = 1$
$A \cdot \bar{A} = 0$	$A + \bar{A} = 1$

公式和定理

- 与普通代数相似的定理

交换律	$\mathrm{A} \cdot B = B \cdot A$	$\mathrm{A} + B = B + A$
结合律	$(\mathrm{A} \cdot B) \cdot C = \mathrm{A} \cdot (B \cdot C)$	$(\mathrm{A} + B) + C = A + (B + C)$
分配律	$\mathrm{A} \cdot (B + C) = \mathrm{A} \cdot B + A \cdot C$	$A + B \cdot C = (A + B) \cdot (A + C)$

- 逻辑代数的一些特殊定理

同一律	$A \cdot A = A$	$A + A = A$
德・摩根定理	$\overline{A \cdot B} = \bar{A} + \bar{B}$	$\overline{A + B} = \bar{A} \cdot \bar{B}$
还原律	$\bar{\bar{A}} = A$	

- 定理证明 – 真值表 $A+B\cdot C=(A+B)\cdot(A+C)$

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

- 定理证明 – 真值表 $\overline{A \cdot B} = \bar{A} + \bar{B}$

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

公式和定理

- 关于等式的三个重要规则：代入规则、反演规则、对偶规则
- 代入规则
  - 在任何逻辑等式中，如果等式两边所有出现某一变量的地方，都代之以一个函数，则等式依然成立
  - 在推导公式中用处很大，将已有等式中某一变量用任意一个函数代替后，就得到了新的等式，从而扩大了等式的应用范围

例：已知 $\overline{A\cdot B}=\bar{A}+\bar{B}$，若用$Y=A\cdot C$ 代替等式中的A，根据代入规则，等式依然成立，故

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

公式和定理

- 关于等式的三个重要规则：代入规则、反演规则、对偶规则
- 反演规则
  - 对于任意一个函数表达式Y，将Y中所有的“·”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Y的反函数$\bar{Y}$
  - 意义在于，利用它比较容易地求出一个逻辑函数的反函数

若 $Y=\bar{A}\cdot\bar{B}+C\cdot D+0$	则 $\bar{Y}=(A+B)\cdot(\bar{C}+\bar{D})\cdot 1$
若 $Y=A+\overline{B+\bar{C}+\overline{D+\bar{\bar{E}}}}$	则 $\bar{Y}=\bar{A}\cdot\overline{\bar{B}\cdot C\cdot\overline{\bar{\bar{D}}\cdot E}}$

- 关于等式的三个重要规则：代入规则、反演规则、对偶规则
- 反演规则
  - 运用反演规则时，不是一个变量上的反号应保持不变
  - 运算符号的优先顺序：先算括号，再算乘积，最后算加法；例：$Y = \bar{A} \cdot \bar{B} + C \cdot D$，应先算$\bar{A} \cdot \bar{B}$和$C \cdot D$，然后将两者相加，应写成$\bar{Y} = (A + B) \cdot (\bar{C} + \bar{D})$，而不是$\bar{Y} = A + B \cdot \bar{C} + \bar{D}$

若 $Y = \bar{A} \cdot \bar{B} + C \cdot D + 0$	则 $\bar{Y} = (A + B) \cdot (\bar{C} + \bar{D}) \cdot 1$
若 $Y = A + \overline{B + \bar{C} + \overline{D + \bar{E}}}$	则 $\bar{Y} = \bar{A} \cdot \overline{\bar{B} \cdot C \cdot \overline{\bar{D} \cdot E}}$

公式和定理

- 关于等式的三个重要规则：代入规则、反演规则、对偶规则
- 对偶规则
  - 对偶式：对任何一个逻辑式Y，将其中的“·”换成“+”，“+”换成“·”；“0”换成“1”，“1”换成“0”，则可得到一个新的逻辑式Y^*（或Y'）
  - 若两个逻辑式相等，则它们的对偶式也相等
  - 优先顺序：括号优先，先乘后加
  - 意义在于若一个公式成立，则其对偶公式也成立

$0 \cdot 0 = 0$	$1 + 1 = 1$
$0 \cdot 1 = 0$	$1 + 0 = 1$
$1 \cdot 1 = 1$	$0 + 0 = 0$
$\bar{0} = 1$	$\bar{1} = 0$

同一律	$A \cdot A = A$	$A + A = A$
德·摩根定理	$\overline{A \cdot B} = \bar{A} + \bar{B}$	$\overline{A + B} = \bar{A} \cdot \bar{B}$

若干常用公式

$A\cdot B+A\cdot\bar{B}=A$	
$A+A\cdot B=A$	• 在一个与或表达式中，如果一个乘积项是另一个乘积项的因子，则这另外一个乘积项是多余的
$A+\bar{A}\cdot B=A+B$	• 在一个与或表达式中，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
$A\cdot B+\bar{A}\cdot C+B\cdot C=A\cdot B+\bar{A}\cdot C$	• 在一个与或表达式中，如果两个乘积项中，一个包含了原变量A，另一项包含了反变量$\bar{A}$，而这两项其余的因子都是第三个乘积项的因子，则第三个乘积项是多余的，称之为冗余项
$\overline{A\cdot\bar{B}+\bar{A}\cdot B}=\bar{A}\cdot\bar{B}+A\cdot B$	• 两个变量异或，其反就是它们的同或；反之，两者同或的反就是他们的异或

一. 逻辑代数的基本概念、公式和定理

1. 基本和常用逻辑运算
2. 公式和定理

二. 逻辑函数的化简方法

1. 逻辑函数的标准与或式和最简式
2. 逻辑函数的公式化简法
3. 逻辑函数的图形化简法
4. 具有约束的逻辑函数的化简

三. 逻辑函数的表示方法及其相互之间的转换

1. 几种表示逻辑函数的方法
2. 几种表示方法之间的转换
3. 用Multisim进行逻辑函数的化简与变换

2.1 逻辑函数的化简方法

- 逻辑函数的化简
  - 逻辑函数的表达式越简单，实现它的电路也越简单
  - 不仅经济，可靠性也得到提高
- 公式化简法
  - 用逻辑代数中的公式和定理进行化简
- 图形化简法
  - 卡诺图

逻辑函数的标准与或式和最简式

- 在数字电路中，用集成电路实现逻辑函数时，有些情况下用的是标准与或式，但一般情况下多是函数的最简表达式，或某种化简形式
  - 逻辑函数的表达式越简单，实现它的电路也越简单
  - 不仅经济，可靠性也得到提高
- 标准与或式
  - 任何逻辑函数都可以表示成为最小项之和的形式，即标准与或式
  - 任何逻辑函数，都是由函数中变量的若干个最小项构成的

最小项的概念

- $Y = F(A, B, C)$
  - 三个变量可以构成8个标准形式的乘积项：
    $\bar{A}\bar{B}\bar{C}, \bar{A}\bar{B}C, \bar{A}B\bar{C}, \bar{A}BC, A\bar{B}\bar{C}, A\bar{B}C, AB\bar{C}, ABC$
- 这8个乘积项共同的特点
  - 1. 每个乘积项都有3个因子
  - 2. 每一个变量都以原变量或者反变量的形式，作为一个因子在乘积项中出现且仅出现一次
  - 这样的乘积项就称为最小项

最小项的概念

- 对于n个变量，如果P是一个含有n个因子的乘积项，而且每一个变量都以原变量或者反变量的形式，作为一个因子在P中出现且仅出现一次，那么就称P是这n个变量的一个最小项
  - n个变量一共有2^{n}个最小项
  - 一个变量：$\bar{A}, A$
  - 两个变量：$\bar{A}\bar{B}, \bar{A}B, A\bar{B}, AB$
  - 三个变量：$\bar{A}\bar{B}\bar{C}, \bar{A}\bar{B}C, \bar{A}B\bar{C}, \bar{A}BC, A\bar{B}\bar{C}, A\bar{B}C, AB\bar{C}, ABC$

最小项的性质

- 每一个最小项都有一组且只有一组使其值为1的对应变量取值
- 任意两个不同的最小项之积，值恒为0
- 变量全部最小项之和，值恒为1

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

最小项是组成逻辑函数的基本单元

- 标准与或式
  - 任何逻辑函数都可以表示成为最小项之和的形式，即标准与或式
  - 任何逻辑函数，都是由函数中变量的若干个最小项构成的

例：写出函数 $Y = AB + BC + CA$ 的标准与或式

$$
\begin{aligned}
Y &= AB + BC + CA \\
&= AB(C + \bar{C}) + BC(A + \bar{A}) + CA(B + \bar{B}) \\
&= AB\bar{C} + ABC + \bar{A}BC + ABC + A\bar{B}C + ABC \\
&= \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC
\end{aligned}
$$

最小项是组成逻辑函数的基本单元

- 标准与或表达式是唯一的
  - 一个逻辑函数只有一个最小项之和的表达式，利用逻辑代数中的公式和定理，可以将任何逻辑函数展开或变换成标准与或表达式
- 标准与或表达式也可以从真值表中直接得到
  - 在真值表中挑出那些使函数值为1的变量取值，变量取值为1的写成原变量，为0的写成反变量
  - 这样对应于使函数值为1的每一种取值，都可以写出一个乘积项
  - 把这些乘积项加起来，所得到的的就是函数的标准与或表达式

最小项是组成逻辑函数的基本单元

- 标准与或表达式也可以从真值表中直接得到

$$
\begin{aligned}
Y &= A + BC \\
&= (A + B)(A + C) \\
&= \bar{A}BC + A\bar{B}\bar{C} + A\bar{B}C + AB\bar{C} + ABC
\end{aligned}
$$

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

最小项的编号

- 把最小项对应的变量取值当成二进制数，与之相应的十进制数，就是该最小项的编号
  - $\bar{A}\bar{B}\bar{C}$的对应取值是000，相应的十进制是0，因此编号为0，记为m_0
  - $\bar{A}\bar{B}C = m_1, \bar{A}BC = m_3, ABC = m_7$
  - 一个最小项，只要把原变量当成1、反变量当成0，便可直接得到它的编号
  - 在书写标准与或表达式时，常常约定用注有下标的小写m表示有关的最小项，甚至只用相应编号表示

$$
\begin{aligned}
Y &= \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC \\
&= m_3 + m_5 + m_6 + m_7 \\
&= \Sigma_i m_i \quad (i = 3,5,6,7) \\
&= \sum m(3,5,6,7) \\
&= \sum(3,5,6,7)
\end{aligned}
$$

逻辑函数的最简表达式

- 按照式中变量之间的运算关系的不同，分为五种
  - 最简与或式
  - 最简与非-与非式
  - 最简或与式
  - 最简或非-或非式
  - 最简或与非式

逻辑函数的最简表达式

- 按照式中变量之间的运算关系的不同，分为五种
  - 最简与或式
  - 最简与非-与非式
  - 最简或与式
  - 最简或非-或非式
  - 最简或与非式

逻辑函数的最简表达式：1. 最简与或式

- 乘积项的个数最少，每个乘积项中相乘的变量个数也最少的与或表达式

$$
\begin{aligned}
Y &= AB + \bar{A}C + BC + BCD \\
&= AB + \bar{A}C + BC \\
&= AB + \bar{A}C
\end{aligned}
$$

逻辑函数的最简表达式：2. 最简与非-与非式

- 非号个数最少，每个非号下面相乘的变量个数也最少的与非-与非式
  - 单个变量上面的非号不算，因为已将其当成反变量
  - 在最简与或表达式的基础上，两次取反，再用德·摩根定理去掉下面的反号

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

$$Y = AB + \bar{A}C$$

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

$$= \overline{\overline{AB} \cdot \overline{\bar{A}C}}$$

逻辑函数的最简表达式：3. 最简或与式

- 括号个数最少，每个括号中相加的变量的个数也最少
  - 在反函数最简与或表达式的基础上，取反，再用德 • 摩根定理去掉反号
  - 在反函数的最简与或表达式的基础上，也可用反演规则，直接写出最简或与式

德 • 摩根定理	$\overline{A \cdot B} = \bar{A} + \bar{B}$	$\overline{A + B} = \bar{A} \cdot \bar{B}$

$$Y = AB + \bar{A}C$$

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

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

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

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

逻辑函数的最简表达式：4. 最简或非-或非式

- 非号个数最少，每个非号下面相加的变量个数也最少的或非-或非式
  - 在最简或与表达式的基础上，两次取反，再用德·摩根定理去掉下面的反号

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

$$Y=AB+\bar{A}C$$

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

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

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

逻辑函数的最简表达式：5. 最简与或非式

- 在非号下面相加的乘积项的个数最少，每个乘积项中相乘的变量个数也最少的与或非式
  - 在最简或非-或非式的基础上，用德·摩根定理去掉大反号下面的小反号
  - 在反函数最简与或式基础上，直接取反亦可

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

$$Y = AB + \bar{A}C$$

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

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

总结

只要得到了函数的最简与或式，再用德·摩根定理进行适当变换，就可以获得其他4种类型的最简式

第二章　数字逻辑基础

一．逻辑代数的基本概念、公式和定理

1. 基本和常用逻辑运算
2. 公式和定理

二．逻辑函数的化简方法

1. 逻辑函数的标准与或式和最简式
2. 逻辑函数的公式化简法
3. 逻辑函数的图形化简法
4. 具有约束的逻辑函数的化简

三．逻辑函数的表示方法及其相互之间的转换

1. 几种表示逻辑函数的方法
2. 几种表示方法之间的转换
3. 用Multisim进行逻辑函数的化简与变换

2.2 逻辑函数的公式化简法

- 在与或表达式的基础上，利用公式和定理，消去表达式中多余的乘积项和每个乘积项中多余的因子，求出函数的最简与或式
  - 并项法
  - 吸收法
  - 消去法
  - 配项销项法

2.2 逻辑函数的公式化简法：1. 并项法

- 利用公式$AB+A\bar{B}=A$，把两个乘积项合并起来，消去一个变量

例：化简函数 $Y=ABC+AB\bar{C}+\bar{A}B$ ，求出最简与或式

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

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

$$=B$$

2.2 逻辑函数的公式化简法：2. 吸收法

- 利用公式$A + AB = A$，吸收掉多余的乘积项

例：化简函数 $Y = \overline{AB} + \bar{A}D + \bar{B}E$ ，求出最简与或式

$$Y = \overline{AB} + \bar{A}D + \bar{B}E$$

$$= \bar{A} + \bar{B} + \bar{A}D + \bar{B}E$$

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

2.2 逻辑函数的公式化简法：3. 消去法

- 利用公式$A + \bar{A}B = A + B$，消去乘积项中多余的因子

例：化简函数 $Y = \overline{AB} + AC + BD$ ，求出最简与或式

$$Y = \overline{AB} + AC + BD$$
$$= \bar{A} + \bar{B} + AC + BD$$
$$= \bar{A} + \bar{B} + C + D$$

- 利用公式$AB + \bar{A}C + BC = AB + \bar{A}C$，在函数与或表达式中加上多余项（冗余项），以消去更多的乘积项，从而获得最简与或式

例：化简函数 $Y = A\bar{C} + \bar{B}C + \bar{A}C + B\bar{C}$，求出最简与或式

$$
\begin{aligned}
Y &= A\bar{C} + \bar{B}C + \bar{A}C + B\bar{C} \\
&= A\bar{C} + \bar{B}C + \bar{A}C + B\bar{C} + \bar{A}B \\
&= A\bar{C} + \bar{A}B + B\bar{C} + \bar{B}C + \bar{A}C \\
&= A\bar{C} + \bar{A}B + \bar{B}C + \bar{A}C \\
&= A\bar{C} + \bar{A}B + \bar{B}C + \bar{A}C \\
&= A\bar{C} + \bar{A}B + \bar{B}C
\end{aligned}
$$

2.2 逻辑函数的公式化简法：4. 配项消项法

- 有时不需要配项，直接消去冗余项
  - $\mathrm{AB}+\bar{A}C+BC=AB+\bar{A}C$，常称为 冗余定理

例：化简函数 $Y=\bar{A}B+AC+\bar{B}\bar{C}+A\bar{B}+\bar{A}\bar{C}+BC$

$$Y=\bar{A}B+AC+BC+\bar{B}\bar{C}+A\bar{B}+\bar{A}\bar{C}$$

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

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

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

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

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

一. 逻辑代数的基本概念、公式和定理

1. 基本和常用逻辑运算
2. 公式和定理

二. 逻辑函数的化简方法

1. 逻辑函数的标准与或式和最简式
2. 逻辑函数的公式化简法
3. 逻辑函数的图形化简法
4. 具有约束的逻辑函数的化简

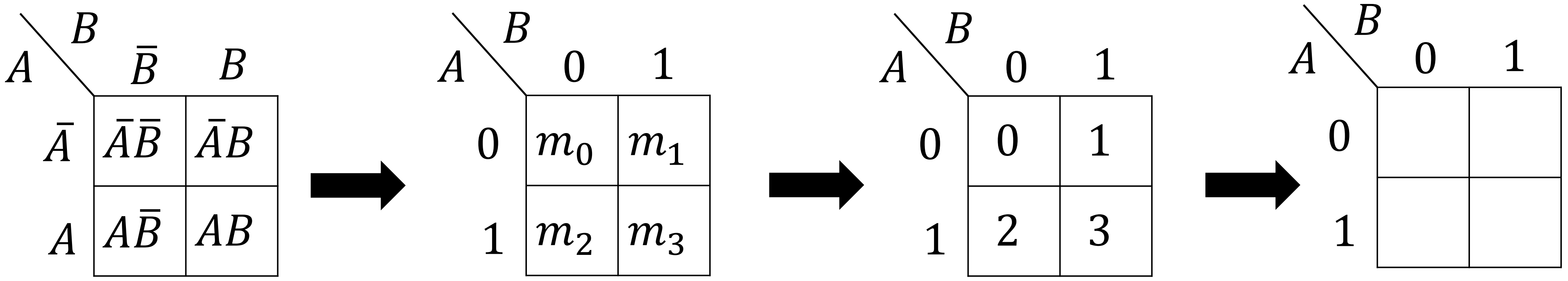
三. 逻辑函数的表示方法及其相互之间的转换

1. 几种表示逻辑函数的方法
2. 几种表示方法之间的转换
3. 用Multisim进行逻辑函数的化简与变换

2.3 逻辑函数的图形化简法

- 用卡诺图化简逻辑函数，求函数的最简与或表达式的方法
  - 有比较明确的步骤可以遵循，结果是否最简，判断起来也比较容易
  - 但是，当变量超过6个以上时，就没有什么实用价值了

- 逻辑变量的卡诺图
  - 变量卡诺图实际上是一种最小项方块图



变量卡诺图的画法

- 一般画成正方形或矩形
  - 对于n个变量，图中分割出的小方块应有2^n个，每一个小方块表示一个最小项
- 按循环码排列变量取值顺序
  - 循环码的特点是相邻编码之间只有1位码元不同（这是关键，只有这样排列，所得到的最小项方块图才称为卡诺图）
  - 循环码可以很容易地由自然二进制码推导出来

三位二进制码	三位循环码
$B = B_2B_1B_0$	$G = G_2G_1G_0$ $G_i = B_{i+1} \oplus B_i$ $B_3 = 0$

变量卡诺图的画法

- 一般画成正方形或矩形
  - 对于n个变量，图中分割出的小方块应有2^n个，每一个小方块表示一个最小项
- 按循环码排列变量取值顺序
  - 循环码的特点是相邻编码之间只有1位码元不同（这是关键，只有这样排列，所得到的最小项方块图才称为卡诺图）

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

$AB \backslash CD$	00	01	11	10
00				
01				
11				
10				

变量卡诺图的特点

- 用几何相邻形象地表示变量各个最小项在逻辑上的相邻性
  - 在卡诺图中，凡是几何相邻的最小项，在逻辑上都是相邻的
  - 变量取值之所以用循环码，就是为了保证这一特性
- 几何相邻
  - 1. 相接：紧挨着
  - 2. 相对：任一行或一列的两头
  - 3. 相重：对折起来后位置重合
- 逻辑相邻
  - 如果两个最小项，除了一个变量的形式不同以外，其余的都相同，那么这两个最小项就成为在逻辑上是相邻的

变量卡诺图的特点

- 逻辑相邻
  - 如果两个最小项，除了一个变量的形式不同以外，其余的都相同，那么这两个最小项就成为在逻辑上是相邻的
  - 在逻辑上相邻的最小项，是可以合并的

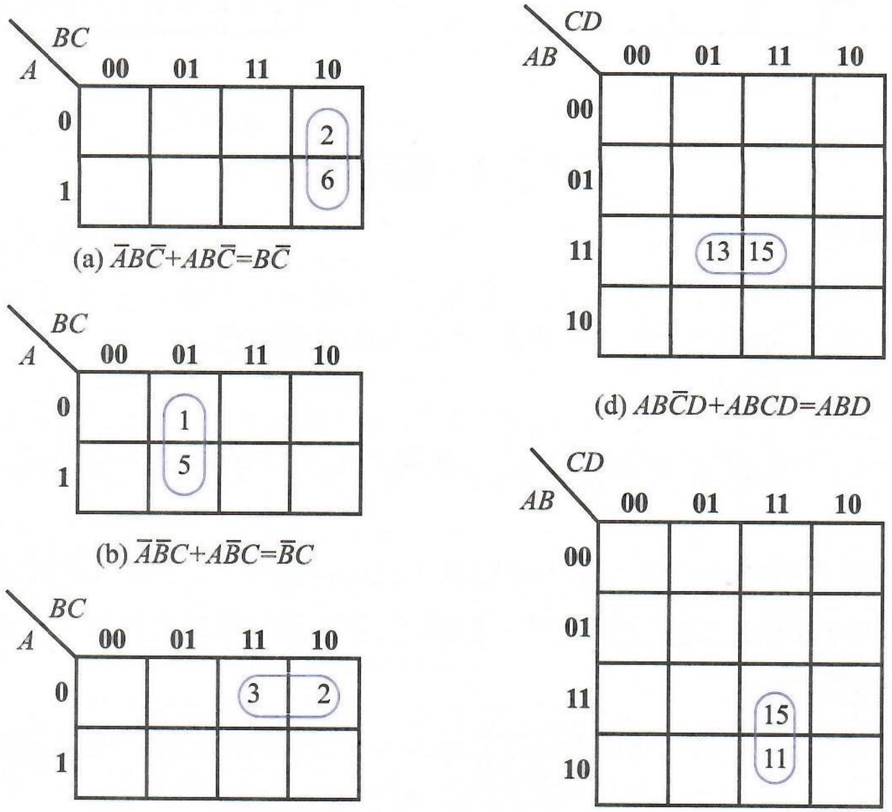
$$AB + A\bar{B} = A$$

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

- 主要缺点：随着变量个数的增加，图形迅速地复杂起来
  - 当变量多于6个时，不仅画图十分麻烦，而且即使画出来了，许多小方块（最小项），是否逻辑相邻也难以辨认，已无使用价值

变量卡诺图中最小项合并的规律

- 凡是几何相邻的最小项均可合并，合并时可以消去有关变量
  - 一般来说，2^n个最小项合并时可以消去n个变量



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

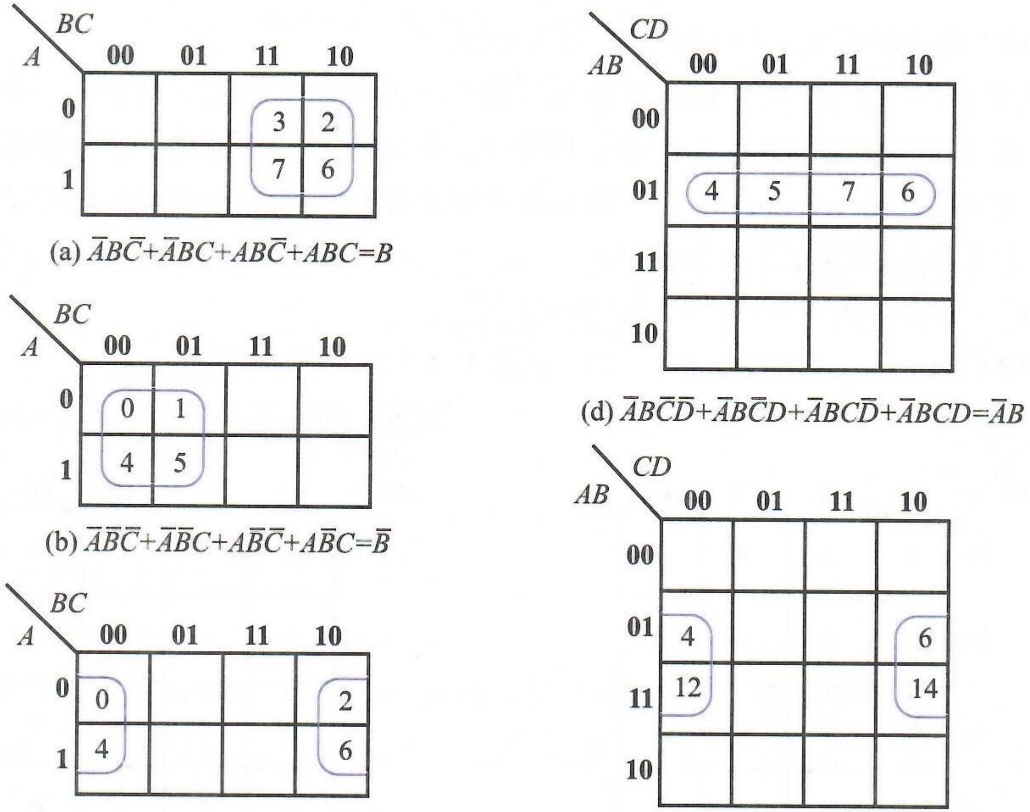
(b) $\bar{A}\bar{B}C+A\bar{B}C=\bar{B}C$

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

(d) $AB\bar{C}D+ABCD=ABD$

(e) $A\bar{B}CD+ABCD=ACD$

图 1.2.3　两个最小项的合并



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

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

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

(d) $\bar{A}B\bar{C}\bar{D}+\bar{A}B\bar{C}D+\bar{A}BC\bar{D}+\bar{A}BCD=\bar{A}B$

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

图 1.2.4　4 个最小项的合并

变量卡诺图中最小项合并的规律

- 凡是几何相邻的最小项均可合并，合并时可以消去有关变量
  - 一般来说，2^n个最小项合并时可以消去n个变量

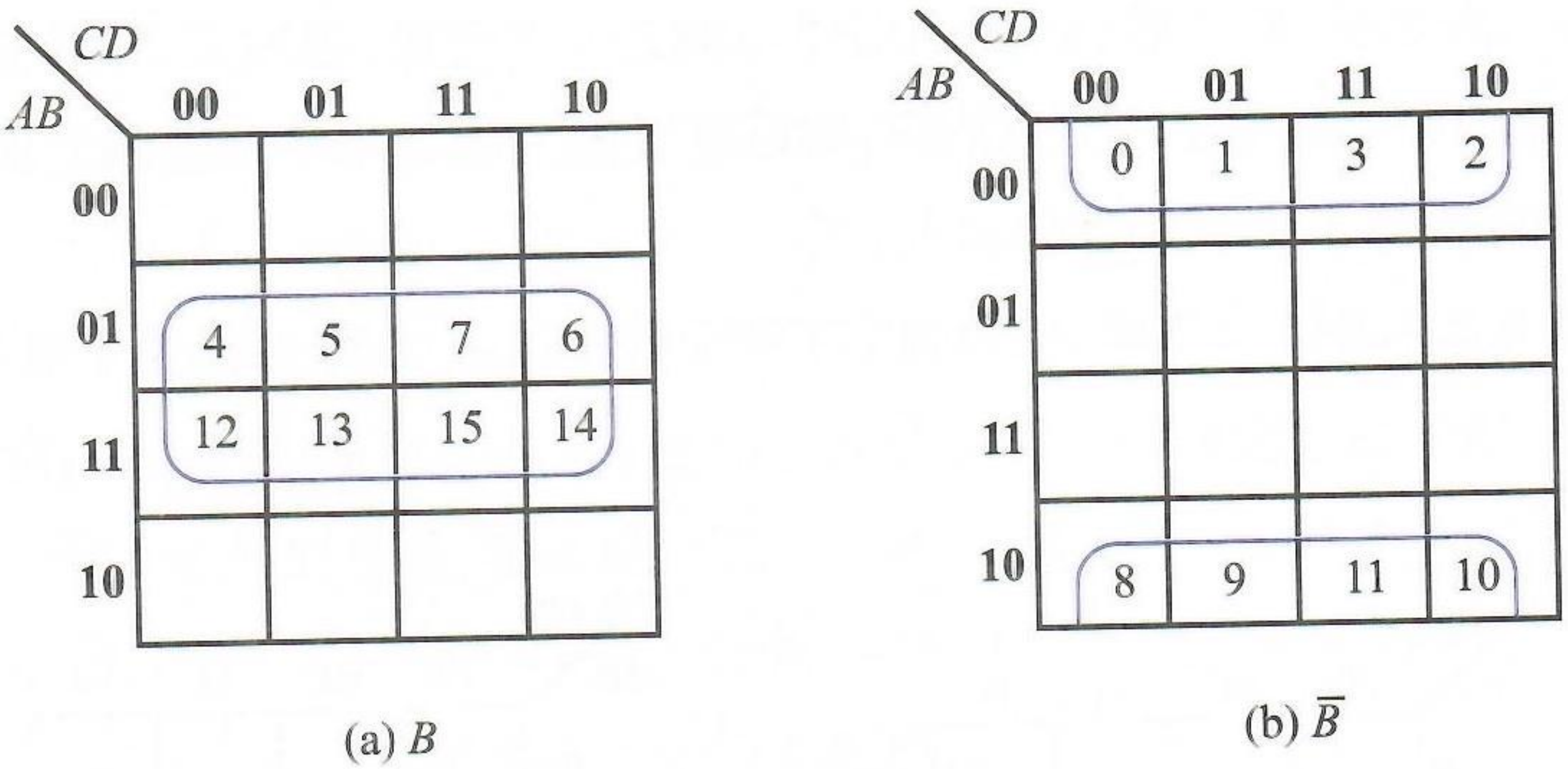


图 1.2.5　8个最小项的合并

逻辑函数的卡诺图

- 在与或表达式基础上，画逻辑函数的卡诺图，可按下列步骤
  - 1. 画出函数变量的卡诺图
  - 2. 在函数的每一个乘积项所包含的最小项处都填上1，剩下的填上0或不填，所得到的就是函数的卡诺图
  - 并不需要将与或式进一步展开成标准与或式

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

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

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

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

逻辑函数卡诺图的特点

- 优点：用几何位置的相邻，形象地表达了构成函数的各个最小项在逻辑上的相邻性，因此可以用来化简逻辑函数，求出函数的最简与或式
- 缺点：当函数变量多于6个时，不仅画起来十分麻烦，而且其优点也不复存在，再无使用价值

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

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

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

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

用卡诺图化简逻辑函数

- 基本步骤
  - 1. 画出逻辑函数的卡诺图
  - 2. 合并逻辑函数的最小项
  - 3. 选择乘积项写出最简与或式
- 合并最小项的原则
  - 必须包含函数的所有最小项，并且保证合并后乘积项的总数最少
  - 相邻的最小项合并时，蕴含函数的最小项数量越多，则合并后乘积项的因子最少
  - 每次合并时，为了消去更多变量，可以重复使用函数的最小项，但是必须保证至少包含1个新的最小项（未被重复使用过），以避免冗余项的出现

用卡诺图化简逻辑函数

- 合并最小项的原则
  - 必须包含函数的所有最小项，并且保证合并后乘积项的总数最少
  - 相邻的最小项合并时，蕴含函数的最小项数量越多，则合并后乘积项的因子最少
  - 每次合并时，为了消去更多变量，可以重复使用函数的最小项，但是必须保证至少包含1个新的最小项（未被重复使用过），以避免冗余项的出现

AB \ CD	00	01	11	10
00		1	3	
01	4	5		
11	12	13		
10			11	10

$Y = \bar{B}CD + B\bar{C} + \bar{A}\bar{C}D + A\bar{B}C$

$\sum m(4,5,12,13) = B\bar{C}$

$\sum m(1,3) = \bar{A}\bar{B}D$

$\sum m(10,11) = A\bar{B}C$

$\sum m(1,5) = \bar{A}\bar{C}D$

$\sum m(3,11) = \bar{B}CD$

- 乘积项$\bar{A}\bar{C}D, \bar{B}CD$包含的最小项都被其他的合并项重复使用过，因此化简结果为：

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

一. 逻辑代数的基本概念、公式和定理

1. 基本和常用逻辑运算
2. 公式和定理

二. 逻辑函数的化简方法

1. 逻辑函数的标准与或式和最简式
2. 逻辑函数的公式化简法
3. 逻辑函数的图形化简法
4. 具有约束的逻辑函数的化简

三. 逻辑函数的表示方法及其相互之间的转换

1. 几种表示逻辑函数的方法
2. 几种表示方法之间的转换
3. 用Multisim进行逻辑函数的化简与变换

2.4 具有约束的逻辑函数的化简

- 约束：用来说明逻辑函数各个变量之间相互制约关系的一个重要概念

例：三八妇女节，某单位包了一场演出，票只发给本单位工作的女同志，以示庆贺，试分析该逻辑问题。

单位	性别	演出票	能否进场	说明
非	男	无	否	
非	男	有		不会出现这种情况
非	女	无	否	
非	女	有		不会出现这种情况
是	男	无	否	
是	男	有		不会出现这种情况
是	女	无	否	
是	女	有	能	

2.4 具有约束的逻辑函数的化简

分析：若用 A、B、C 分别表示单位、性别、演出票，为0时表示非本单位、男同志、无票，为1时表示是本单位、女同志、有票；用Y表示能否进场，为0时不能进场，为1时表示能进场，则真值表为

A	B	C	Y	说明
0	0	0	0	
0	0	1		不会出现
0	1	0	0	
0	1	1		不会出现
1	0	0	0	
1	0	1		不会出现
1	1	0	0	
1	1	1	1	

- ABC的取值只可能出现000,010,100,111，而不会出现001,011,101，因为演出票只发给在本单位工作的女同志
- 这说明 A、B、C 之间有着一定的制约关系，因此称这三个变量是一组有约束的变量
- 由有约束的变量所决定的逻辑函数，称为有约束的逻辑函数

约束项与约束条件

- 不会出现的变量取值所对应的最小项称为约束项
  - ABC的取值只可能出现000,010,100,111，而不会出现001,011,101，因为演出票只发给在本单位工作的女同志，故约束项为$\bar{A}\bar{B}C,\bar{A}BC,A\bar{B}C$
  - 由最小项的性质可知，只有对应变量取值出现时，其值才会为1；约束项对应的是不出现的变量取值，所以其值恒为0
- 由约束项加起来所构成的值为0的逻辑表达式，称为约束条件
  - 约束项的值恒为0，所以约束条件是一个值等于0的条件等式

约束条件的表示方法

- 在真值表中，在对应于约束项的变量取值所决定的函数值处记上叉号（×）
- 在逻辑表达式中，用等于0的条件等式表示
- 在卡诺图中，在约束项处记上叉，以区别于其他最小项

A	B	C	Y
0	0	0	0
0	0	1	×
0	1	0	0
0	1	1	×
1	0	0	0
1	0	1	×
1	1	0	0
1	1	1	1

最小项之和表达式
$\bar{A}\bar{B}C + \bar{A}BC + A\bar{B}C = 0$
标准与或表达式
$\sum d(1,3,5) = 0$
最简与或表达式
$\bar{A}C + \bar{B}C = 0$

$A \backslash BC$	00	01	11	10
0	0	×	×	0
1	0	×	1	0

具有约束的逻辑函数的化简

- 在公式法中的应用
  - 根据需要加上或者去掉约束条件，函数不受影响（约束条件值恒为0）

最小项之和表达式
$\bar{A}\bar{B}C + \bar{A}BC + A\bar{B}C = 0$
标准与或表达式
$\sum d(1,3,5) = 0$
最简与或表达式
$\bar{A}C + \bar{B}C = 0$

$$
\begin{aligned}
Y &= ABC \\
&= ABC + \bar{A}C + \bar{B}C \\
&= C(AB + \bar{A} + \bar{B}) \\
&= C(AB + \overline{AB}) \\
&= C
\end{aligned}
$$

具有约束的逻辑函数的化简

- 在图形法中的应用
  - 在利用函数的卡诺图合并最小项时，可根据化简的需要包含或去掉约束项
  - 合并最小项时，如果圈中包含了约束项，相当于在相应的乘积项中加上了该约束项，其值为0，函数不会受影响

- 理解约束条件化简的实际意义
  - $Y = ABC$：派到剧场的守门人不仅要查票，而且要辨认持票人的单位和性别，很麻烦
  - $Y = C$：只要查票就行，很简单
  - 必须遵守约束条件：事先只能把票发给本单位工作的女同志（否则出现外单位的人，或本单位男）
  - 凡是利用约束条件化简了逻辑函数，就必须遵守约束条件，否则就可能出现错误

$A \backslash BC$	00	01	11	10
0	0	×	×	0
1	0	×	1	0

$$Y = m_7$$
$$= m_1 + m_3 + m_5 + m_7$$
$$= C$$

变量互相排斥的逻辑函数的化简

- 互相排斥的变量
  - 带此类约束：一组变量中，只要有一个变量取值为1，其他变量的值就一定是0

A	B	C	Y	A
0	0	0	0	0
0	0	1	1	
0	1	0	1	
0	1	1	×	不会出现
1	0	0	1	
1	0	1	×	不会出现
1	1	0	×	不会出现
1	1	1	×	不会出现

$A \backslash BC$	00	01	11	10
0	0	1	×	1
1	1	×	×	×

⬇

$$m_4 + m_5 + m_6 + m_7 = A$$

$$m_2 + m_3 + m_6 + m_7 = B$$

$$m_1 + m_3 + m_5 + m_7 = C$$

⬇

最简与或表达式
$Y = A + B + C$

变量互相排斥的逻辑函数的化简

- 互相排斥的变量
  - 带此类约束：一组变量中，只要有一个变量取值为1，其他变量的值就一定是0

A	B	C	Y	A
0	0	0	0	0
0	0	1	1	
0	1	0	1	
0	1	1	×	不会出现
1	0	0	1	
1	0	1	×	不会出现
1	1	0	×	不会出现
1	1	1	×	不会出现

最简与或表达式
$Y = A + B + C$

⬇

	Y
A	1
B	1
C	1

简化形式真值表

带约束的逻辑函数的化简举例

$$\begin{cases} Y = AC + \bar{A}\bar{B}C \\ \bar{B}\bar{C} = 0 \end{cases} \quad \text{(约束条件)}$$

$A \backslash BC$	00	01	11	10
0	×	1	0	0
1	×	1	1	0

$\bar{B}$ $\quad$ AC

最简与或表达式

$$\begin{cases} Y = \bar{B} + AC \\ \bar{B}\bar{C} = 0 \end{cases}$$

第二章 数字逻辑基础

一. 逻辑代数的基本概念、公式和定理

1. 基本和常用逻辑运算
2. 公式和定理

二. 逻辑函数的化简方法

1. 逻辑函数的标准与或式和最简式
2. 逻辑函数的公式化简法
3. 逻辑函数的图形化简法
4. 具有约束的逻辑函数的化简

三. 逻辑函数的表示方法及其相互之间的转换

1. 几种表示逻辑函数的方法
2. 几种表示方法之间的转换
3. 用Multisim进行逻辑函数的化简与变换

3.1 几种表示逻辑函数的方法

- 真值表、卡诺图、表达式、逻辑图、波形图、HDL语言（硬件语言）描述

- 真值表
  - 把变量的各种可能取值与相应的函数值，以表格形式一一列举出来
  - 优点1. 直观明了。在许多数字集成电路手册中，常常都以不同形式的真值表，给出器件的逻辑功能
  - 优点2. 把一个实际逻辑问题抽象成为数学表达形式时，使用真值表是最方便的。一般在数字电路逻辑设计过程中，第一步就是列出真值表，在分析数字电路逻辑功能时，最后也要列出真值表
  - 缺点1. 难以用逻辑代数的公式和定理进行运算和变换
  - 缺点2. 当变量比较多时，列真值表会十分繁琐。为简单起见，在真值表中只列出使函数值为1的输入变量取值

3.1 几种表示逻辑函数的方法

- 真值表、卡诺图、表达式、逻辑图、波形图、HDL语言（硬件语言）描述

- 卡诺图
  - 卡诺图可以说是真值表的一种方块图表达形式。变量取值必须按照循环码的顺序排列，与真值表有严格的一一对应关系，因此也称为真值方格图
  - 优点：用几何相邻形象直观地表示了函数各个最小项在逻辑上的相邻性，便于用来求逻辑函数的最简与或表达式
  - 缺点：只适用于表示和化简变量个数比较少的逻辑函数，也不便于用公式和定理进行运算和变换

3.1 几种表示逻辑函数的方法

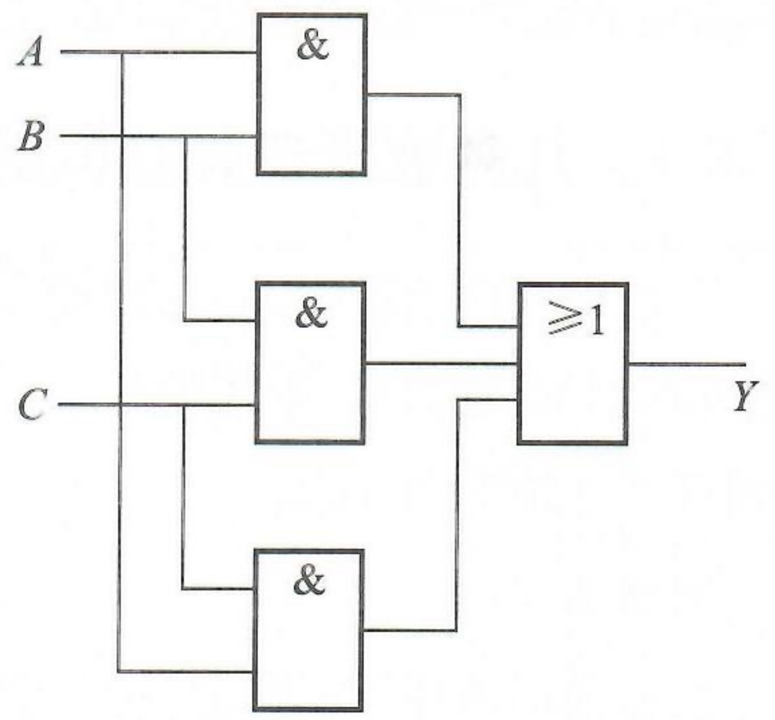
- 真值表、卡诺图、表达式、逻辑图、波形图、HDL语言（硬件语言）描述

- 逻辑表达式
  - 用与、或、非等运算表示函数中各个变量之间逻辑关系的代数式子
  - 优点：书写简洁、方便，可以用公式和定理十分灵活地进行运算、变换
  - 缺点：在逻辑函数比较复杂时，难以直接从变量取值看出函数的值，不如真值表和卡诺图直观

- 真值表、卡诺图、表达式、逻辑图、波形图、HDL语言（硬件语言）描述

- 逻辑图
  - 用基本和常用的逻辑符号表示函数表达式中各个变量之间的运算关系，便能够画出函数的逻辑图
  - 逻辑图与表达式是一一对应的

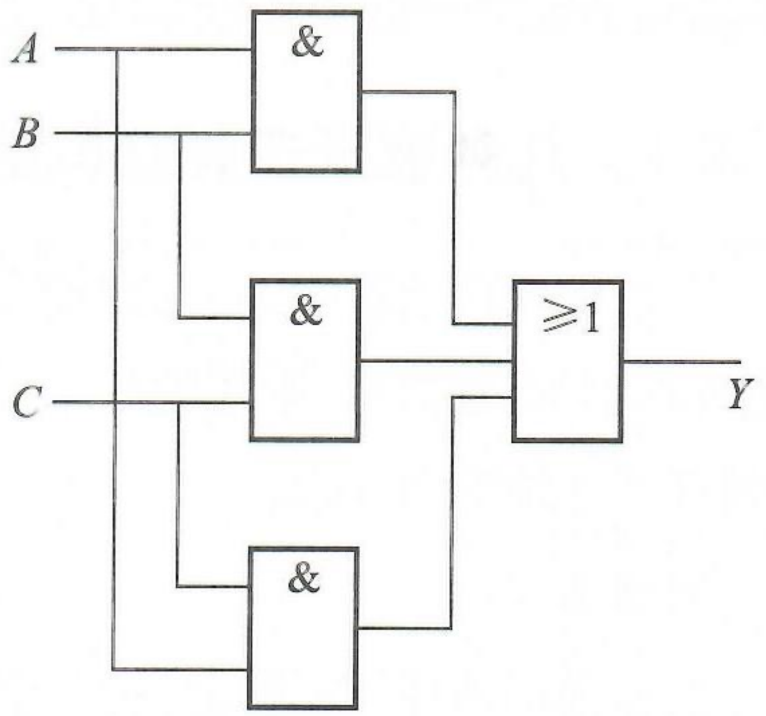
$$Y = AB + BC + CA$$



3.1 几种表示逻辑函数的方法

- 真值表、卡诺图、表达式、逻辑图、波形图、HDL语言（硬件语言）描述

- 逻辑图
  - 逻辑图中的逻辑符号，都有称之为门电路的实际电路器件存在，比较接近工程实际
  - 在工作中，要了解某个数字系统或者数控装置的逻辑功能时，都要用到逻辑图，因为它可以把许多繁杂的实际电路的逻辑功能层次分明地表示出来
  - 在制作数字设备时，常常也要通过逻辑设计，画出逻辑图，然后再把逻辑图变成实际电路
  - 缺点：不能用公式和定理进行运算和变换，表示的逻辑关系不如真值表和卡诺图直观

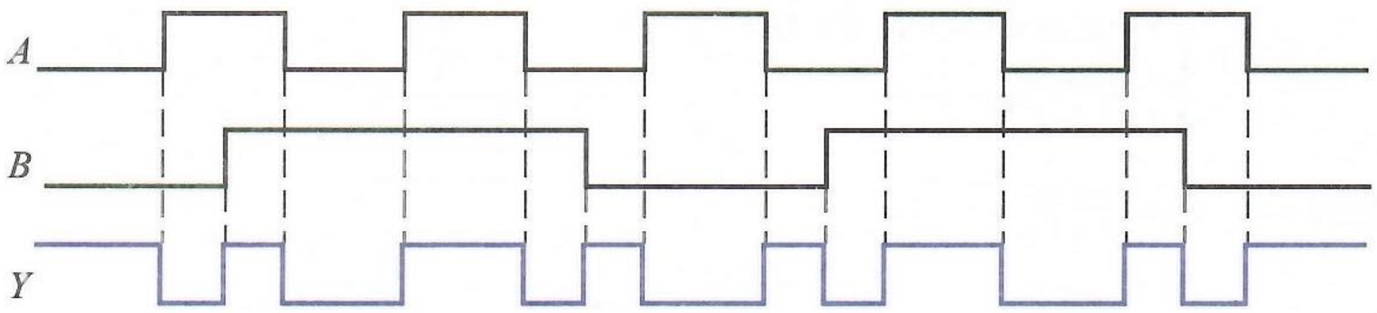


3.1 几种表示逻辑函数的方法

- 真值表、卡诺图、表达式、逻辑图、波形图、HDL语言（硬件语言）描述

- 波形图
  - 在给出输入变量取值随时间变化的波形后，根据函数中变量之间的运算关系、真值表或者卡诺图中变量取值和函数值的对应关系，都可以对应画出输出变量（函数）随时间变化的波形
  - 这种反映输入和输出变量对应取值随时间按照一定规律变化的图形，就称为波形图，也称为时间图

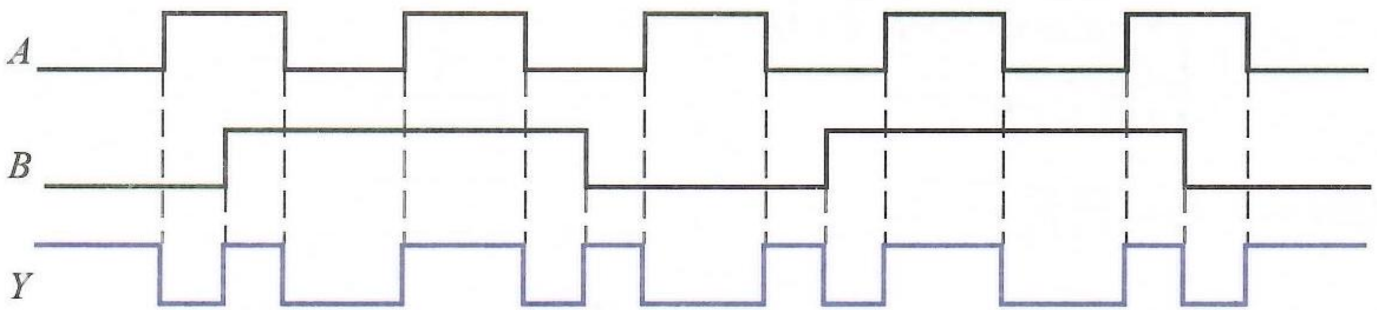
$$Y = AB + \bar{A}\bar{B}$$



- 真值表、卡诺图、表达式、逻辑图、波形图、HDL语言（硬件语言）描述

- 波形图
  - 横坐标是时间轴，纵坐标是变量取值
  - 时间轴相同，变量取值只有0（低）和1（高），所以约定在图中不标出坐标轴
  - 画具体波形时，一定要对应起来画

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



一. 逻辑代数的基本概念、公式和定理

1. 基本和常用逻辑运算
2. 公式和定理

二. 逻辑函数的化简方法

1. 逻辑函数的标准与或式和最简式
2. 逻辑函数的公式化简法
3. 逻辑函数的图形化简法
4. 具有约束的逻辑函数的化简

三. 逻辑函数的表示方法及其相互之间的转换

1. 几种表示逻辑函数的方法
2. 几种表示方法之间的转换
3. 用Multisim进行逻辑函数的化简与变换

3.2 几种表示方法之间的转换

- 最基本：真值表与逻辑图之间的转换

- 由真值表到逻辑图的转换
  1. 根据真值表写出函数的与或表达式或者画出函数的卡诺图
  2. 用公式法或者图形法进行化简，求出函数的最简与或表达式
  3. 根据表达式画逻辑图，有时还要对与或表达式做适当变换，才能画出所需的逻辑图

由真值表到逻辑图的转换

1. 根据真值表写出函数的与或表达式或者画出函数的卡诺图
2. 用公式法或者图形法进行化简，求出函数的最简与或表达式
3. 根据表达式画逻辑图，有时还要对与或表达式做适当变换，才能画出所需要的逻辑图

例：输出变量Y是输入变量A、B、C的函数，当 A、B、C 取值中有奇数个1时 $Y=1$，否则$Y=0$，而且输入变量取值不会出现全为0的情况。

A	B	C	Y
0	0	0	×
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

写表达式

$$\begin{cases} Y=\bar{A}\bar{B}C+\bar{A}B\bar{C}+A\bar{B}\bar{C}+ABC \\ \bar{A}\bar{B}\bar{C}=0 \quad (\text{约束条件}) \end{cases}$$

由真值表到逻辑图的转换

例：输出变量Y是输入变量A、B、C的函数，当A、B、C取值中有奇数个1时$Y=1$，否则$Y=0$，而且输入变量取值不会出现全为0的情况。

写表达式

$$\begin{cases} Y = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC \\ \bar{A}\bar{B}\bar{C} = 0 \quad (\text{约束条件}) \end{cases}$$

⬇

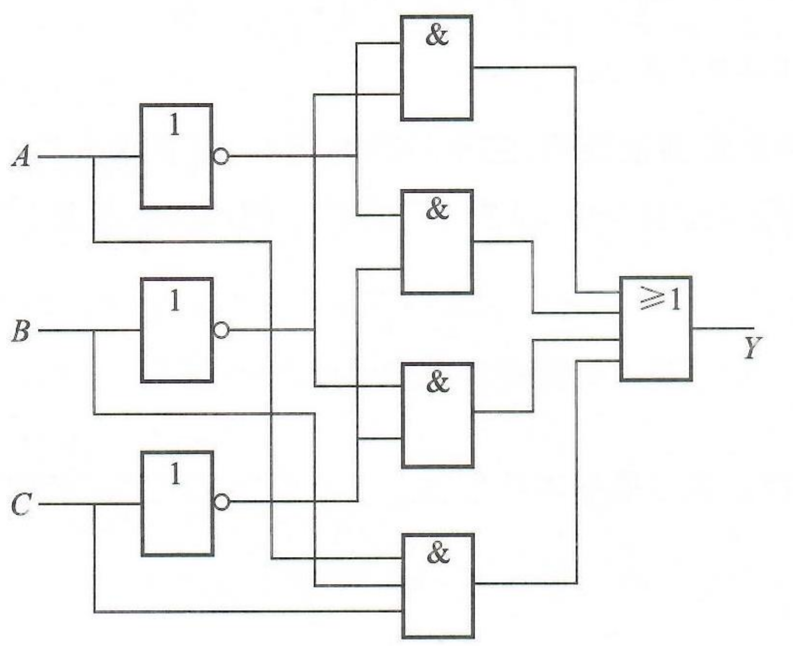
$$\begin{aligned} Y &= \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC + \bar{A}\bar{B}\bar{C} \\ &= \bar{A}\bar{B} + \bar{A}\bar{C} + \bar{B}\bar{C} + ABC \end{aligned}$$

⬇

最简与或表达式

$$\begin{cases} Y = \bar{A}\bar{B} + \bar{A}\bar{C} + \bar{B}\bar{C} + ABC \\ \bar{A}\bar{B}\bar{C} = 0 \end{cases}$$

⬅



用与、或、非运算逻辑符号

由真值表到逻辑图的转换

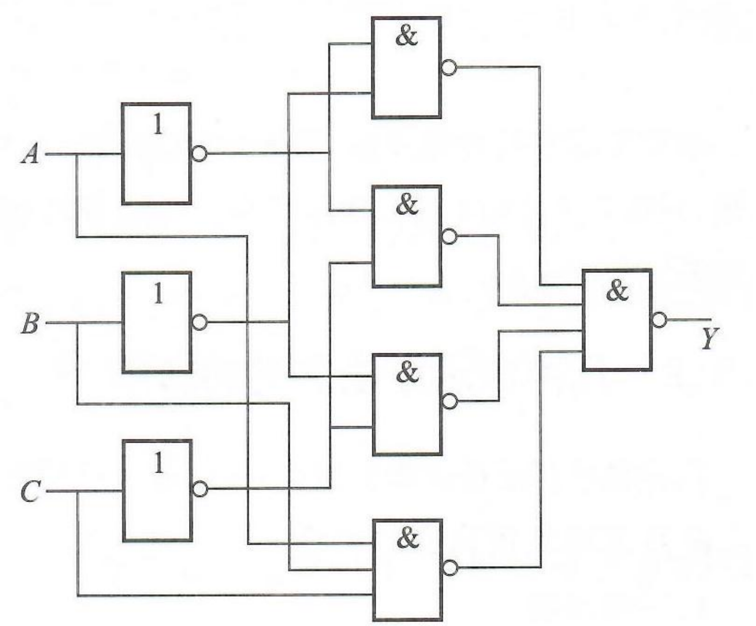
例：输出变量Y是输入变量A、B、C的函数，当A、B、C取值中有奇数个1时$Y=1$，否则$Y=0$，而且输入变量取值不会出现全为0的情况。

最简与或表达式

$$\begin{cases} Y = \bar{A}\bar{B} + \bar{A}\bar{C} + \bar{B}\bar{C} + ABC \\ \bar{A}\bar{B}\bar{C} = 0 \end{cases}$$

最简与非-与非表达式

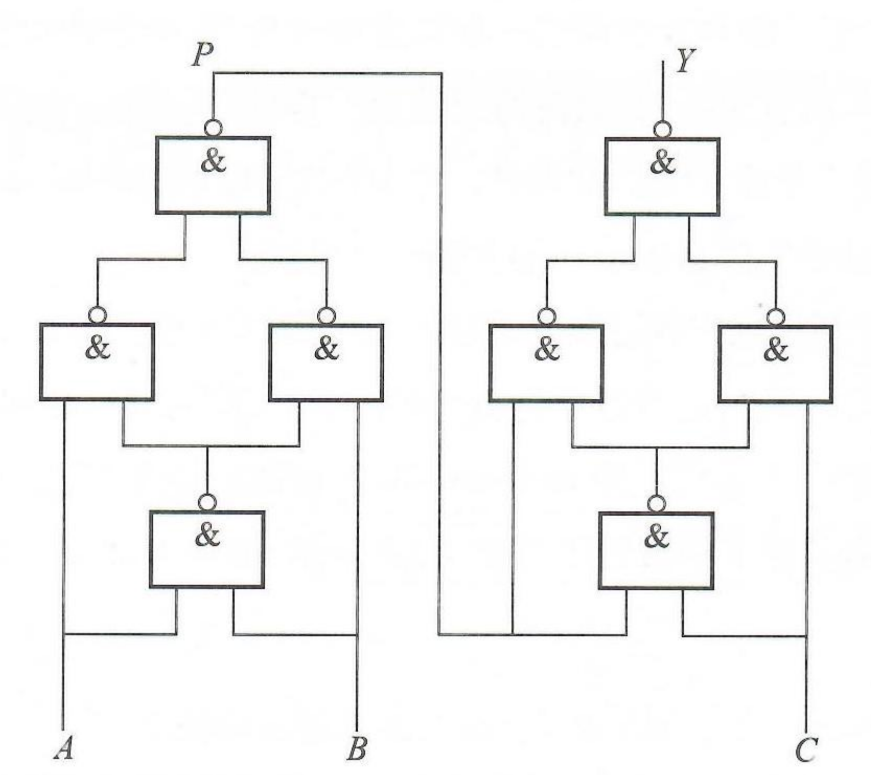
$$\begin{cases} Y = \bar{A}\bar{B} + \bar{A}\bar{C} + \bar{B}\bar{C} + ABC \\ \quad = \overline{\overline{\bar{A}\bar{B} + \bar{A}\bar{C} + \bar{B}\bar{C} + ABC}} \\ \quad = \overline{\overline{\bar{A}\bar{B}} \cdot \overline{\bar{A}\bar{C}} \cdot \overline{\bar{B}\bar{C}} \cdot \overline{ABC}} \\ \bar{A}\bar{B}\bar{C} = 0 \end{cases}$$



用非、与非运算逻辑符号

由逻辑图到真值表的转换

1. 从输入到输出或从输出到输入，用逐级推导的方法，写出输出变量（函数）的逻辑表达式
2. 进行化简，求出函数的最简与或式
3. 将变量各种可能取值代入与或式中进行运算，列出函数的真值表



设中间变量为P:

$$P = \overline{\overline{A\overline{AB}} \cdot \overline{\overline{AB}B}} = A\overline{AB} + \overline{AB}B = A\bar{B} + \bar{A}B$$

$$Y = \overline{\overline{P\overline{PC}} \cdot \overline{\overline{PC}C}} = P\overline{PC} + \overline{PC}C = P\bar{C} + \bar{P}C$$

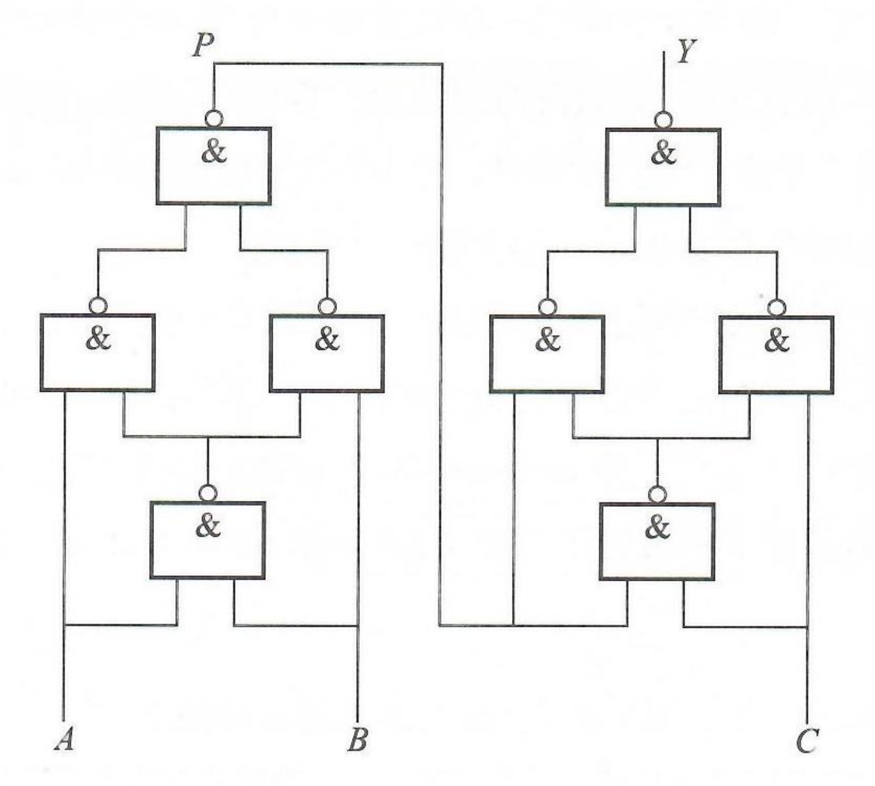
$$= (A\bar{B} + \bar{A}B) \cdot \bar{C} + \overline{A\bar{B} + \bar{A}B} \cdot C$$

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

由逻辑图到真值表的转换

1. 从输入到输出或从输出到输入，用逐级推导的方法，写出输出变量（函数）的逻辑表达式
2. 进行化简，求出函数的最简与或式
3. 将变量各种可能取值代入与或式中进行运算，列出函数的真值表

$$Y=A\bar{B}\bar{C}+\bar{A}B\bar{C}+\bar{A}\bar{B}C+ABC$$



A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

一. 逻辑代数的基本概念、公式和定理

1. 基本和常用逻辑运算
2. 公式和定理

二. 逻辑函数的化简方法

1. 逻辑函数的标准与或式和最简式
2. 逻辑函数的公式化简法
3. 逻辑函数的图形化简法
4. 具有约束的逻辑函数的化简

三. 逻辑函数的表示方法及其相互之间的转换

1. 几种表示逻辑函数的方法
2. 几种表示方法之间的转换
3. 用Multisim进行逻辑函数的化简与变换

3.3 用Multisim进行逻辑函数的化简与变换

- 电子设计自动化（Electronic design automation, EDA）技术
  - 随着半导体技术、集成技术的飞速发展，电子器件的更新速度越来越快，电子电路的复杂程度也日益提高
  - 以计算机为工作平台，融合应用电子技术、计算机技术、智能化技术的最新成果，研制而成的各种计算机辅助设计（Computer aided design, CAD）软件
  - 这些CAD软件可应用于电路性能和参数的分析、运行状态的仿真分析、集成电路芯片的设计、可编程器件的设计以及印制电路板的设计

3.3 用Multisim进行逻辑函数的化简与变换

- 在各类CAD软件中，Multisim是常用的EDA仿真软件
  - 满足对一般的数字电路、模拟电路以及数字-模拟混合电路进行分析和设计的需要
  - 与Ultiroute软件配合使用，可以自动完成从电路原理图输入到印制电路板设计的全部设计过程
  - 用户界面友好、直观，使用非常方便，易学易用

课后自主练习：

使用Multisim软件在逻辑函数的真值表、标准与或式、最简与或式以及逻辑图之间任意进行转换

第二章 小结

- 常用逻辑关系及运算
  - 三种基本逻辑运算：与 、或、非
  - 四种复合逻辑运算：与非 、或非、与或非、异或

真值表	函数式	逻辑符号

- 逻辑代数的公式和定理
  - 是推演、变换和化简逻辑函数的依据，有些与普通代数相同，有些则完全不同，要认真加以区别
  - 这些定理中，摩根定理最为常用

第二章 小结

- 逻辑函数的化简法
  - 化简的目的是为了获得最简逻辑函数式，从而使逻辑电路简单、成本低、可靠性高
  - 化简的方法主要有公式化简法和图形化简法两种

1. 公式化简法：可化简任何复杂的逻辑函数，但要求能熟练和灵活运用逻辑代数的各种公式和定理，并要求具有一定的运算技巧和经验。

2. 图形化简法：简单、直观，不易出错，有一定的步骤和方法可循。但是，当函数的变量个数多于六个时，就失去了优点，没有实用价值。

约束项：（无关项） 可以取 0，也可以取 1，它的取值对逻辑函数值没有影响，应充分利用这一特点化简逻辑函数，以得到更为满意的化简结果。

第二章 小结

- 逻辑函数常用的表示方法

真值表、卡诺图、函数式、逻辑图和波形图。

- 它们各有特点，但本质相同，可以相互转换
- 尤其是由真值表 → 逻辑图 和 逻辑图 → 真值表， 在逻辑电路的分析和设计中经常用到，必须熟练掌握

第二章 作业

教材：《数字电子技术基础简明教程（第四版）》

- 习题：1.8, 1.9, 1.11, 1.15, 1.16

THE END