数字逻辑电路

目 录

第一章 数字逻辑基础

第一章 数字逻辑基础

Fundamentals of Boolean Algebar

- 布尔代数 ***Boolean algebra***： 用一种数学运算的代数系统描述人的逻辑思维规律和推理过程。

- 逻辑代数 ***Switching algebra***： 将布尔代数的一些基本前提和定理应用于继电器的分析与描述，称为二值布尔代数，或开关代数。继电器是当时最常用的数字逻辑元件，继电器的接触状态（打开或闭合）用 **0** 或 **1**表示。

- 逻辑代数是二值逻辑运算中的基本数学工具
- 逻辑代数广泛应用于数字系统的分析和设计

数字逻辑电路

第一章 数字逻辑基础

Fundamentals of Boolean Algebar

- 逻辑代数是二值逻辑运算中的基本数学工具
- 逻辑代数广泛应用于数字系统的分析和设计

在现代逻辑分析技术中，逻辑值对应于各种广泛的物理条件：电压的高或低、灯光的明或暗、电容器的充电或放电、熔丝的断开或接通，等等。

不同的计算机逻辑和存储技术中表示位值的物理状态

	表示位值的状态	
技术	0	1
气动逻辑	低压流动	高压流动
继电器逻辑	电路断开	电路闭合
CMOS逻辑	0~1.5V	3.5~5.0V
TTL逻辑	0~0.8V	2.0~5.0V
光纤	暗	亮
动态存储	电容放电	电容充电
非易失的可擦存储器	电子捕获	电子释放
双极只读存储器	熔丝烧断	熔丝完好
磁泡存储器	无磁泡	有磁泡
磁带存储器	磁通朝“北”	磁通朝“南”
聚合体存储器	分子处于状态A	分子处于状态B
只读压缩盘	无凹陷	凹陷
可重写压缩盘	晶态染色	非晶态染色

1.3.1 逻辑代数中的几个概念

1. 逻辑状态 *Logic State*：

当事物的某些特性表现为两种互不相容的状态，即

①某一时刻必出现且仅出现一种状态

②一种状态是另一种状态的反状态

则用符号0、1分别表示这两种状态，称逻辑状态。

即：0 状态 (*0－state*) 和 1 状态 (*1－state*) 状态可互换

一般，0状态——逻辑条件的假或无效，

1状态——逻辑条件的真或有效。

(两种状态无大小之分)

数字逻辑电路

2. 逻辑变量 *Logic Value* ：

用于表示事物的逻辑状态随逻辑条件的变化而变化，取值 **“0”** 或 **“1”** 。

3. 逻辑常量 *Logic Constant* ：

逻辑状态保持不变，取值 **“0”** 或 **“1”**。

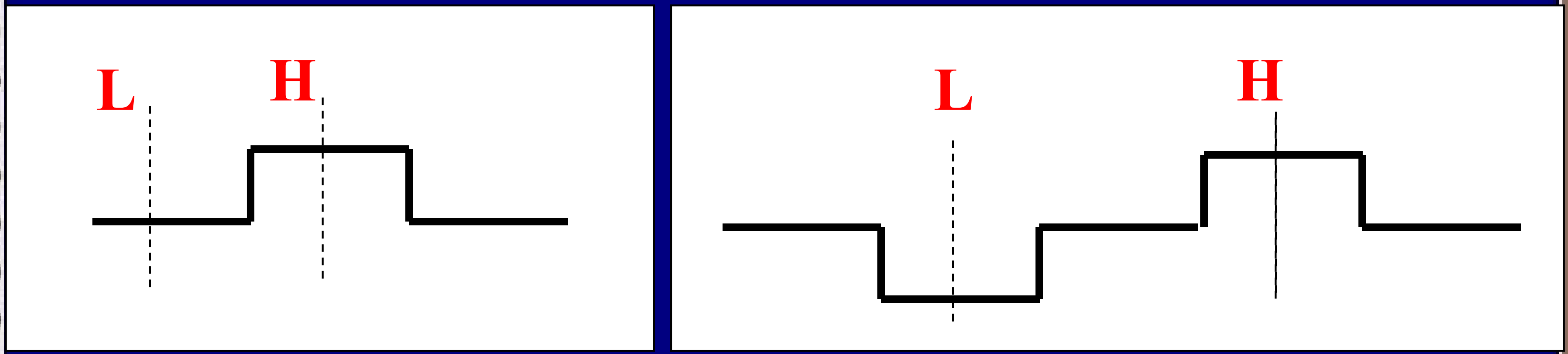
4. 逻辑电平 *Logic Voltage*：

- 在二值逻辑电路(开关电路)中，将物理器件的物理量离散为两种电平：高电平（用 **H** 表示）、低电平（用 **L** 表示）
- 抽象化的高、低电平忽略其物理量值的实际含义，实际上它们是代表着一定范围的物理量。参见下页。
- 在高、低电平之间有一逻辑不确定区，称为“噪音区”。若电平稳定于噪音区称为逻辑模糊，这在逻辑电路中不允许。

表2－1不同工艺器件定义的逻辑电平

工艺	逻辑电平（电源电压为5V）	
	L	H
TTL	0～0.40V	3.0 ～ 5.0V
CMOS	0 ～ 0.80V	2.0 ～ 5.0V

图2-1 脉冲的逻辑电平表示



5. 逻辑约定 *Logic Assumpsit*：

规定　逻辑电平（表示物理器件的输入、输出物理量）

与　逻辑状态（表示物理器件的逻辑功能）

之间的 关系，即逻辑规定（约定）。

这一规定过程称为逻辑化过程。

逻辑约定有两种：正逻辑规定（约定）

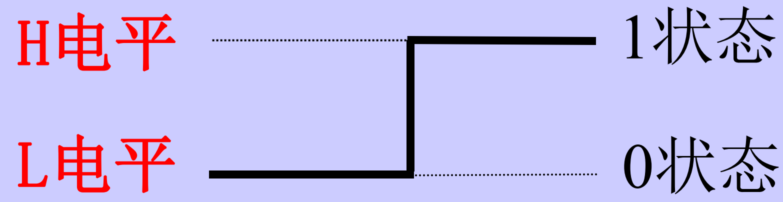
和　负逻辑规定（约定），如下：

约定.　1　高电平　=> 器件 用什么门.

0　低电平

正逻辑规定（约定）

逻辑电平	逻辑状态
L	0
H	1

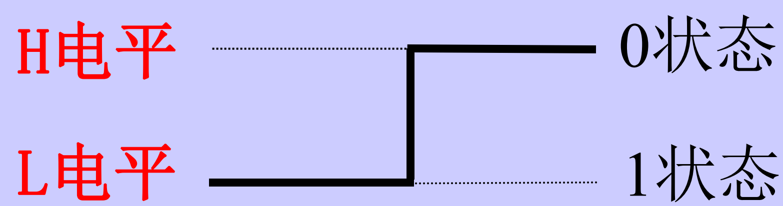


(a) 正逻辑规定（约定）

注：本书均采用正逻辑约定。

负逻辑规定（约定）

逻辑电平	逻辑状态
L	1
H	0



(b) 负逻辑规定（约定）

- 逻辑电路*Logic Circuit*：

　　由实现逻辑变量之间逻辑关系的物理器件所构成的电路称为逻辑电路，即二值逻辑电路。

6. 逻辑代数 *Logic Algebar* ：

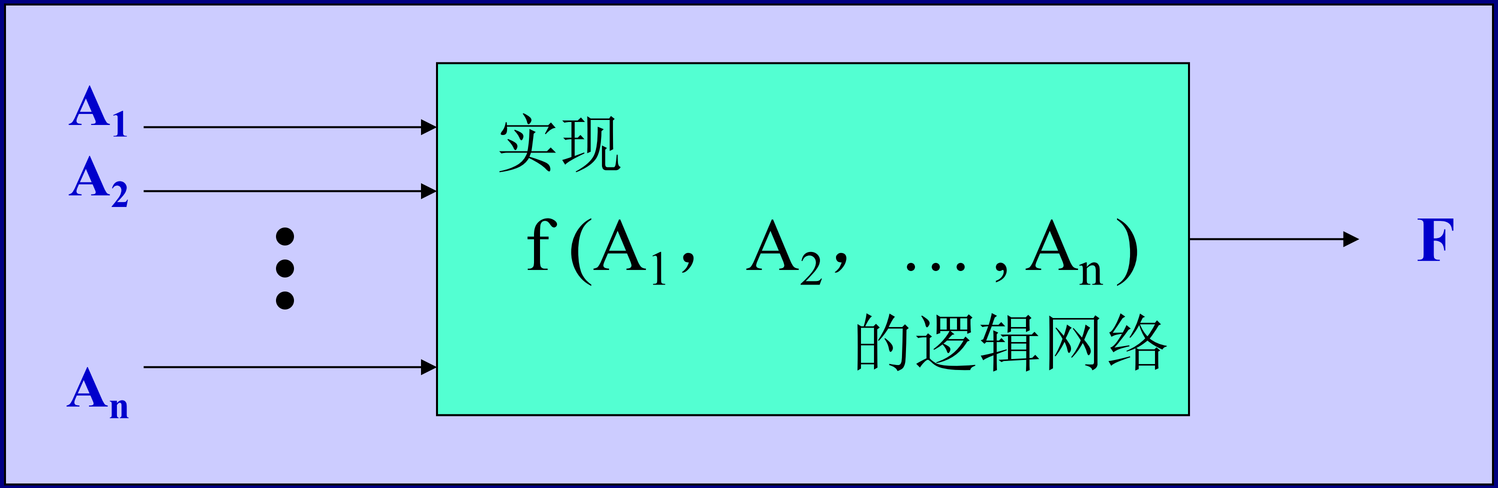
- 用代数形式表现逻辑变量之间的因果关系。
- 用代数运算对这些逻辑变量进行逻辑推理。

因此，逻辑代数是一个集合：逻辑变量集、常量0和1、“与”、“或”和“非”三种逻辑运算。

运算顺序是：“非”最高，“与”次之，“或”最低。

7. 逻辑函数 *Logic Function*： 老板

输入逻辑变量 A_1， A_2， … , A_n；输出逻辑变量F；记为：$F = f(A_1，A_2，\ldots , A_n)$，关系如下图所示：



F = f (A1, A2, …, An)

数字逻辑电路

8. 逻辑函数的表示法 *Representation*： 主要有四种

(1) 真值表（穷举法） *Truth Table* （万能）枚举表，费事，后期挺有用

(2) 逻辑表达式 *Algebraic Forms of Switching Functions* 通法 不快，最直观

(3) 卡诺图 *Karnaugh MAP* （文氏图 *Venn Diagrams 1881*）

(4) 时间图 （信号波形图 ） *Timing*

若出了器件 要另连接

真值表例 2变量 4行.

2^2种

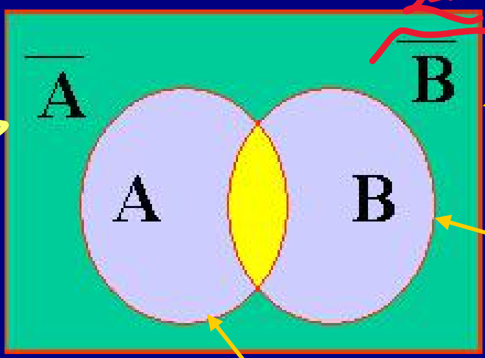
A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

与门

与电路最紧密

表达式例： **F = A · B**

Venn图



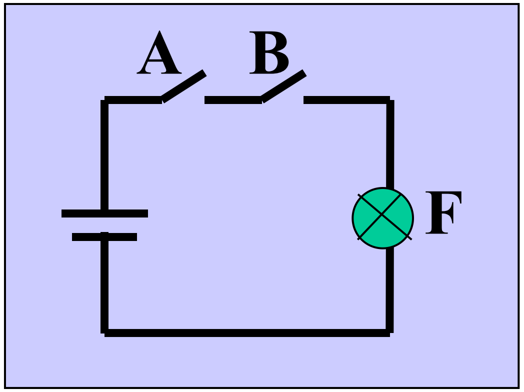
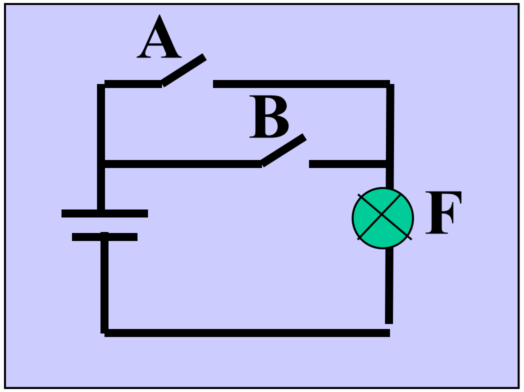
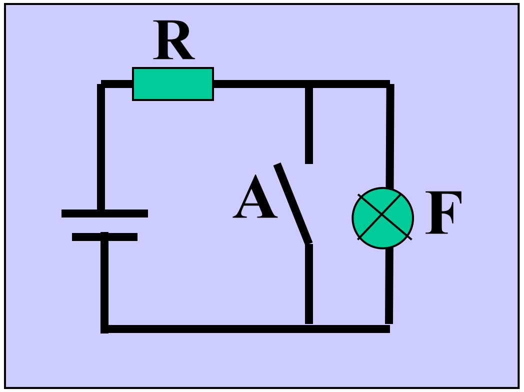
全集 为 1

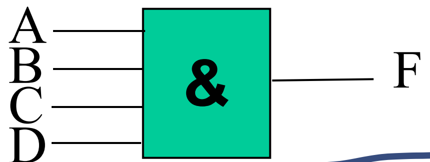
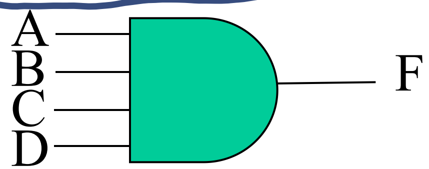
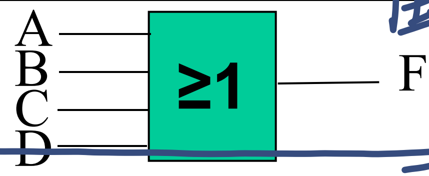
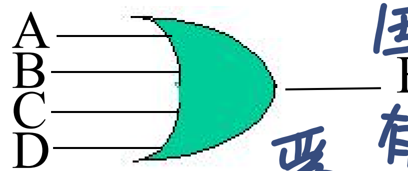
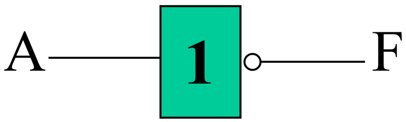
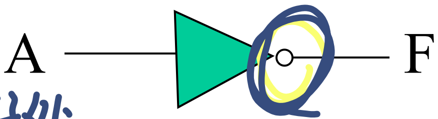
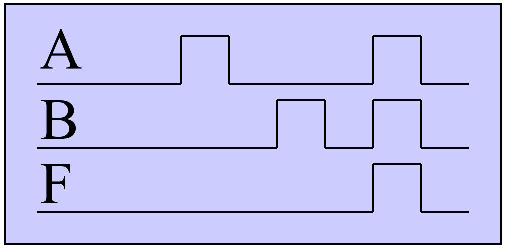
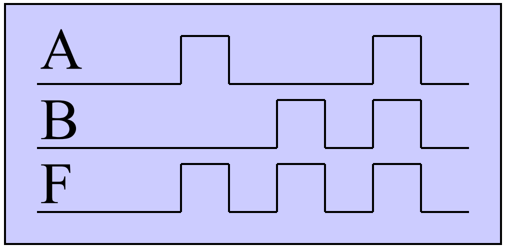
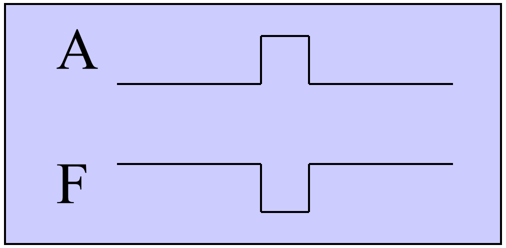
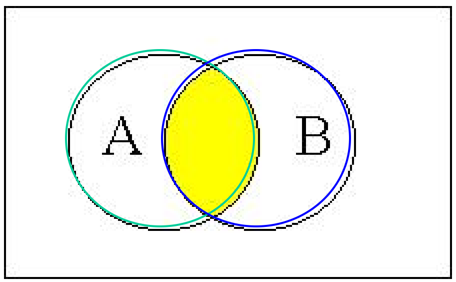
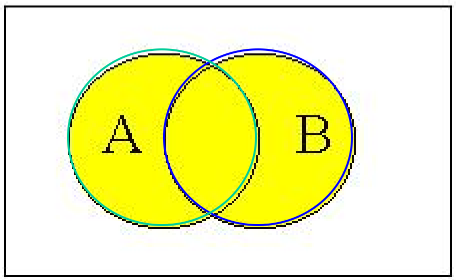
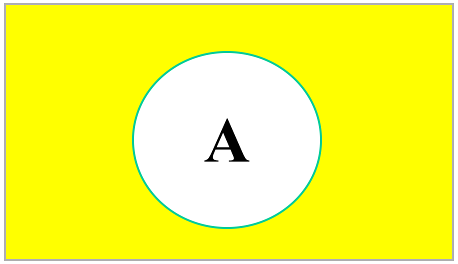
又引入变量B，将已有区域再分别一分为二

只要表达式就对已经

引入变量A，将区域 一分为二

1.3.2 逻辑代数的基本运算

<table>
<tr><th></th><th>“与”运算(逻辑乘)
Logic Multiplication</th><th>“或”运算(逻辑加)
Logic Addition</th><th>“非”运算(逻辑非)
Logic Negation</th></tr>
<tr><td>运算结果</td><td>逻辑积
Logic Product</td><td>逻辑和
Logic Sum</td><td>求补
Complement</td></tr>
<tr><td>示意电路</td><td>
</td><td>
</td><td>
</td></tr>
<tr><td>真值表</td><td><table><tr><th>A B</th><th>F</th></tr><tr><td>0 0</td><td>0</td></tr><tr><td>0 1</td><td>0</td></tr><tr><td>1 0</td><td>0</td></tr><tr><td>1 1</td><td>1</td></tr></table></td><td><table><tr><th>A B</th><th>F</th></tr><tr><td>0 0</td><td>0</td></tr><tr><td>0 1</td><td>1</td></tr><tr><td>1 0</td><td>1</td></tr><tr><td>1 1</td><td>1</td></tr></table></td><td><table><tr><th>A</th><th>F</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table></td></tr>
</table>

需要记忆	“与”运算(逻辑乘) *Logic Multiplication*	“或”运算(逻辑加) *Logic Addition*	“非”运算(逻辑非) *Logic Negation*
代数式	$F = A \times B = A \cdot B$	$F = A + B$	$F = \overline{A}$
逻辑符号	 	  要	国标  国际 有定好处 
波形图			
文氏图 (F为阴影)			

1.3.3 逻辑代数的基本定理及规则

基本运算： $\overline{1} = 0$ $\qquad$ $\overline{0} = 1$

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

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

$0 \cdot 0 = 0$ $\qquad$ $1 + 1 = 1\ (\neq 10)$

1.3.3.1 布尔代数的基本公理 *Basic Postulates*

公理是基本的假设，是客观存在，无需证明。可以用真值表验证等式成立，当然等式两边也会具有相同的卡诺图。

后期不用化简 FPGA已很顶了

运算的优先顺序：括号，非，与，或。选择即可

或时 1起作用；与时，0起作用

0—1 律 *0 and 1 elements for* + *and* • *operators*

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

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

数字逻辑电路

Commutativity of the + and · operations

交换律 $A + B = B + A$ $A \cdot B = B \cdot A$

结合律 $A + (B + C) = (A + B) + C$

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

x 对 +
与对或 有分配律

Distributivity of the + and · operations

分配律 $A + B \cdot C = (A + B)(A + C)$ △

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

$A\ A + A \cdot B + A \cdot C + B \cdot C$

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

$A(1+B+C) + B \cdot C = A + BC$

例：证明 分配律 $A+B \cdot C=(A+B)(A+C)$ 成立。

用真值表证明，如下：

-把出来，不要一个一个算

A B C	B • C	A+BC	A+B	A+C	(A+B)(A+C)
0 0 0	0	0	0	0	0
0 0 1	0	0	0	1	0
0 1 0	0	0	1	0	0
0 1 1	1	1	1	1	1
1 0 0	0	1	1	1	1
1 0 1	0	1	1	1	1
1 1 0	0	1	1	1	1
1 1 1	1	1	1	1	1

有1则

八种下。等式成立

由此证明 $A+B \cdot C=(A+B)(A+C)$ 成立。 这逻辑证明不考

可以用同样的方法证明 $A \cdot (B+C) = A \cdot B + A \cdot C$ 成立。

互补律 ***Complement*** $A + \overline{A} = 1$ $A \cdot \overline{A} = 0$

重叠律 ***Idempotency*** $A + A = A$ $A \cdot A = A$

对合律 ***involution*** $\overline{\overline{A}} = A$

1.3.3.2 逻辑代数的基本定理 *Fundamental Theorems*

大 不会

画 Venn 可知.

吸收律 *Absorption*

有括先或再与

$A + AB = A$ 吸子集 　　　　 $A \cdot (A + B) = A$

先与后或

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

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

例：证明 $A + \bar{A} \cdot B = A + B$，可以用公理来证明。

右边 $= A + 1 \cdot B$ 　看懂 不用证 　（0—1律）

$= A + (A + \bar{A}) \cdot B$ 　（互补律）

$= \boxed{A + AB} + \bar{A}B$ 　（分配律）

$A(1+B) + \bar{A}B$

$= A + \bar{A}B$ 　（吸收律）

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

$=$ 左边 $= A \cdot B =$ 证明成立

数字逻辑电路

反演律 *DeMorgan's Theorem* (摩根定理)

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

或非 = 非与 反函 与非

证明 $\overline{A\cdot B} = \overline{A}+\overline{B}$ 成立，可以用函数的互补性来证明

AB与 $\overline{A}+\overline{B}$ 互补

设：$X = A\,B \qquad Y = \overline{A}+\overline{B}$

∵ $X + Y = AB + \overline{A}+\overline{B}$

$= \overline{A} + B + \overline{B}$ （吸收律）

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

同功能 比成本

$= \overline{A} + 1$ （互补律）

$= 1$ （0—1律）

反函 往往可以简化

又∵ $X\cdot Y = AB(\overline{A}+\overline{B})$

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

时针

与或式成本低

$= 0\cdot A + 0\cdot B$ (互补律)

$= 0 + 0$ (0—1律)

$= 0$ (基本运算)

∴ X 与 Y 互补，$X = \overline{Y}$，$Y = \overline{X}$。证明摩根定理成立。

数字逻辑电路

N变量的摩根定理：

$$\overline{A_1 + A_2 + \ldots + A_n} = \overline{A_1} \cdot \overline{A_2} \cdot \ldots \cdot \overline{A_n}$$

$$\overline{A_1 \cdot A_2 \cdot \ldots \cdot A_n} = \overline{A_1} + \overline{A_2} + \ldots + \overline{A_n}$$

（此定理证明见代入规则。）

摩根定理的作用：进行函数化简和逻辑变换。

包含律 *Consensus* (也称多余项定理)

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

多余的

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

例：证明 $AB + \overline{A}C + BC = AB + \overline{A}C$ 不要求

左边 $= AB + \overline{A}C + BC \cdot 1$ （0—1律）

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

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

$= AB + ABC + \overline{A}C + \overline{A}CB$ （交换律）

$= AB + \overline{A}C$ （吸收律）

= 右边 证明成立

数字逻辑电路

1.3.3.4 逻辑代数的基本规则 *Basic Formulas*

1. 代入规则： 定理. 吸收 摩根. 多余 （共5）

已知 $f(x_1, x_2, \ldots, x_i, \ldots, x_n) = g(x_1, x_2, \ldots, x_i, \ldots, x_n)$

有任意函数 h，令：$x_i = h$

则 $f(x_1, x_2, \ldots, h, \ldots, x_n) = g(x_1, x_2, \ldots, h, \ldots, x_n)$

依然成立。

证明： ∵ x_i 取值(只有) 0 或 1，使等式成立

又∵ 逻辑函数 h 取值也是仅有 0 或 1

∴ 代入规则成立。

数字逻辑电路

用代入规则证明N变量的摩根定理，如下：

证明N变量的摩根定理：

$$\overline{A_1+A_2+\ldots+A_n} = \overline{A_1}\cdot\overline{A_2}\cdot\ldots\cdot\overline{A_n}$$

∵ $\overline{A_1+X} = \overline{A_1}\cdot\overline{X}$ (摩根定理)

令 $X = A_2 + Y$ 代入

则 $\overline{A_1+A_2+Y} = \overline{A_1}\cdot\overline{(A_2+Y)}$

$= \overline{A_1}\cdot\overline{A_2}\cdot\overline{Y}$ (摩根定理)

令 $Y = A_3 + Z$ 代入

则 $\overline{A_1+A_2+A_3+Z} = \overline{A_1}\cdot\overline{A_2}\cdot\overline{(A_3+Z)}$

$= \overline{A_1}\cdot\overline{A_2}\cdot\overline{A_3}\cdot\overline{Z}$ (摩根定理)

依次类推，则推出 N 变量的摩根定理。

数字逻辑电路

2. 反演规则(香农定理) 必填空

Shannon's expansion theorem

已知原函数 $f(x_1, x_2, \ldots, x_n, 0, 1, +, \cdot)$

则反函数 $\overline{f}(x_1, x_2, \ldots, x_n, 0, 1, +, \cdot)$

$$= f(\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_n, 1, 0, \cdot, +)$$

这是反演律和代入规则的推广使用，是对互补函数的完善说明。

例：$F = A \bullet [\,\overline{B} + (C\overline{D} + \overline{E}) \bullet G\,]$ 优先 顺序不变

先做

则反函数 $\overline{F} = \overline{A} + B \bullet [(\overline{C}+D) \bullet E + \overline{G}\,]$

变量取反. 0/1互换 与或互换

最好不用动，直接反演 如

$A + [\overline{B} + \overline{(C\overline{D}+\overline{E})}\,G]$ 不用动

3. 对偶规则 *Dual expansion theorem* 不用于电路设计

与或式 ⇒ 或与式（比较符合认知 简洁）

已知原函数 $f(x_1, x_2, \ldots, x_n, 0, 1, +, \cdot)$

则对偶函数 $f'(x_1, x_2, \ldots, x_n, 0, 1, +, \cdot)$

$= f(x_1, x_2, \ldots, x_n, 1, 0, \cdot, +)$

不用取反.

结论： 1. $(f')' = f$

2. 若 $f_1 = f_2$

则 $f_1' = f_2'$ 逆射

与或→或与.

1.3.4 逻辑函数的性质

1.3.4.1 复合逻辑

- 与、或、非三种基本逻辑运算组合起来可以实现任何逻辑函数。 逻辑完备组

- 与门、或门、非门三种基本逻辑运算(门)组合起来可以构成实现任何逻辑功能的逻辑电路，称此三门构成了一个逻辑完备组。

- 若实现一个较复杂的逻辑功能，尤其在大规模集成电路快速发展的今天，必须增加门电路的功能，以简化电路。

1. 与非逻辑(NAND)

逻辑表达式为: $F = \overline{A \cdot B \cdot C}$

先有与非门,再有与

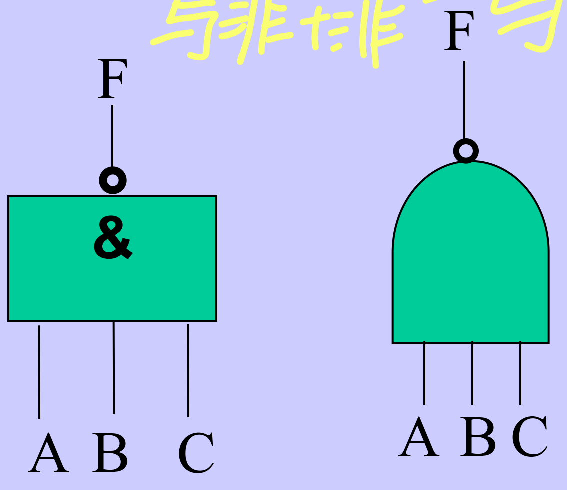
与非逻辑真值表

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

与非门的逻辑符号

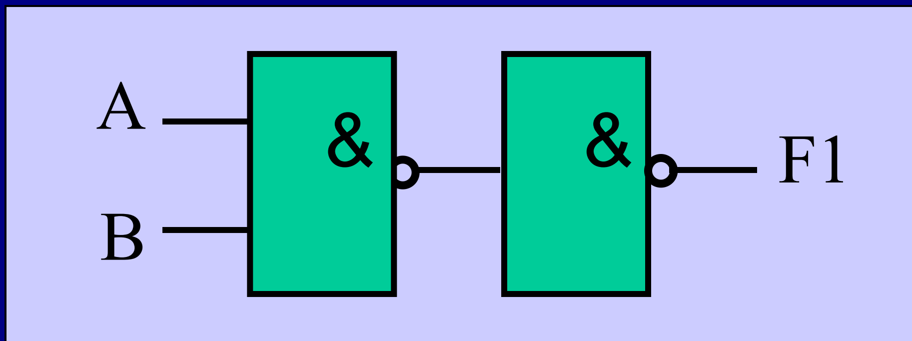
与非门成本更低

与非+非=与

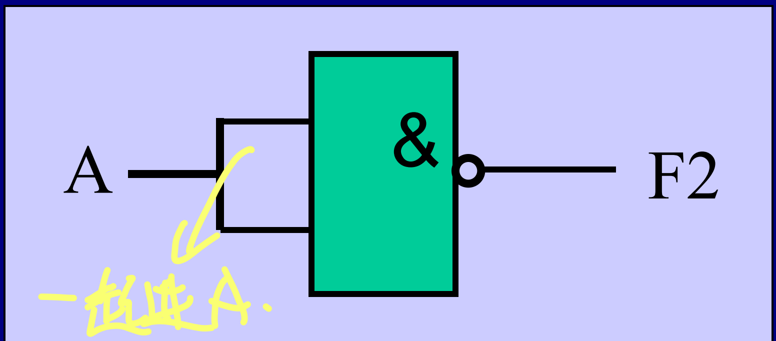


可以用与非门实现三种基本运算：

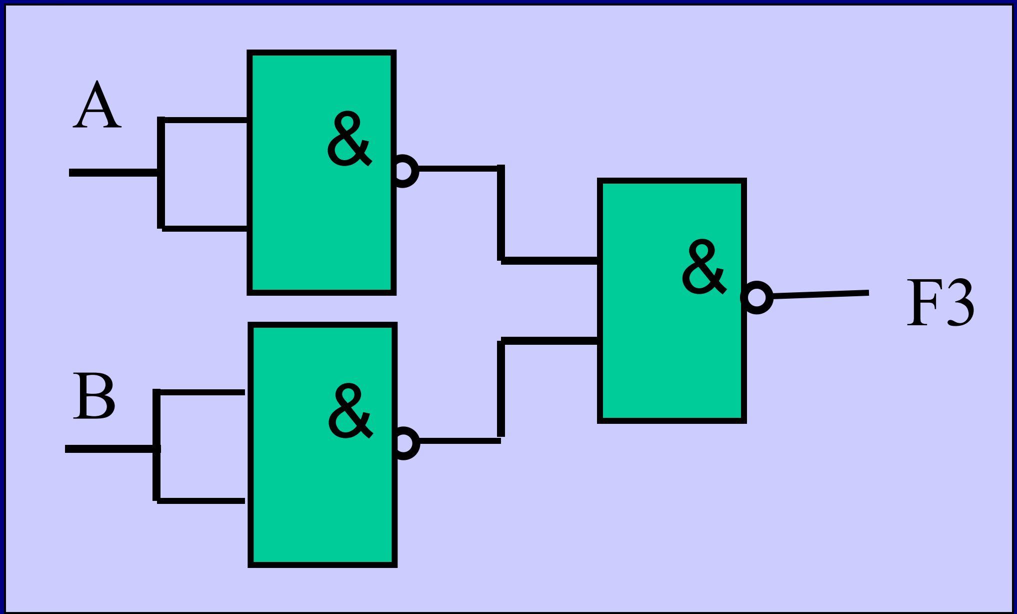
① 与运算 $F1=\overline{\overline{A\cdot B}}$



② 非运算 $F2=\overline{A\cdot A}=\overline{A}$



③ 或运算 $F3=\overline{\overline{A\cdot A}\cdot\overline{B\cdot B}}=\overline{\overline{A}\cdot\overline{B}}=A+B$



实际上比
与比难.

2. 或非逻辑(NOR)

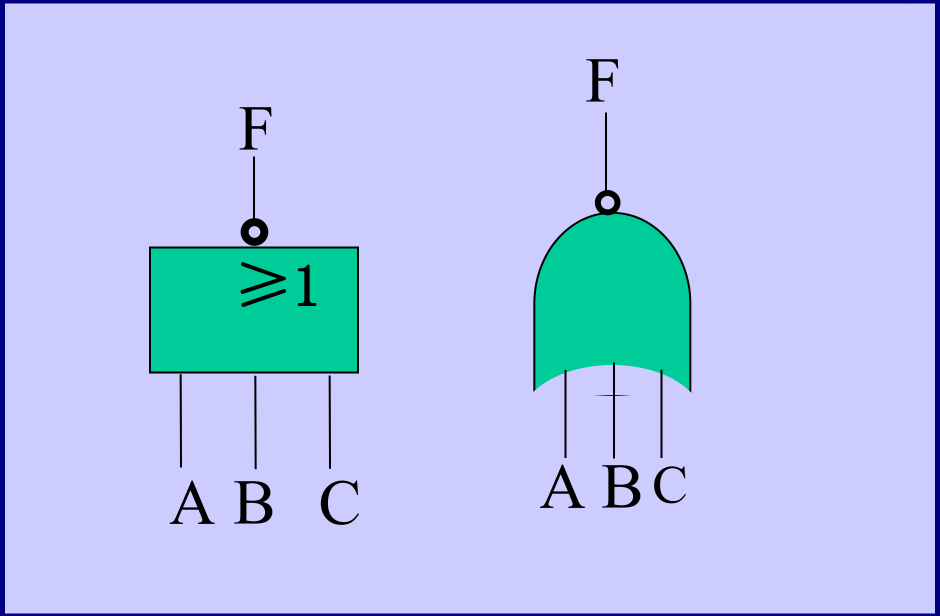
与非更优秀 多用管子 ·

逻辑表达式为: $F = \overline{A + B + C}$

或非逻辑真值表

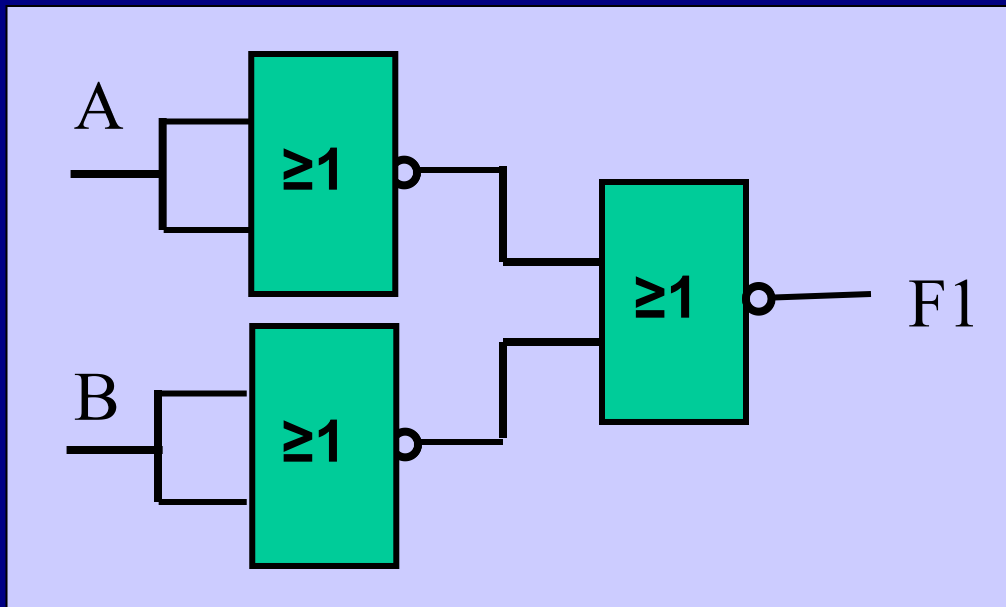
A B C	F
0 0 0	1
0 0 1	0
0 1 0	0
0 1 1	0
1 0 0	0
1 0 1	0
1 1 0	0
1 1 1	0

或非门的逻辑符号

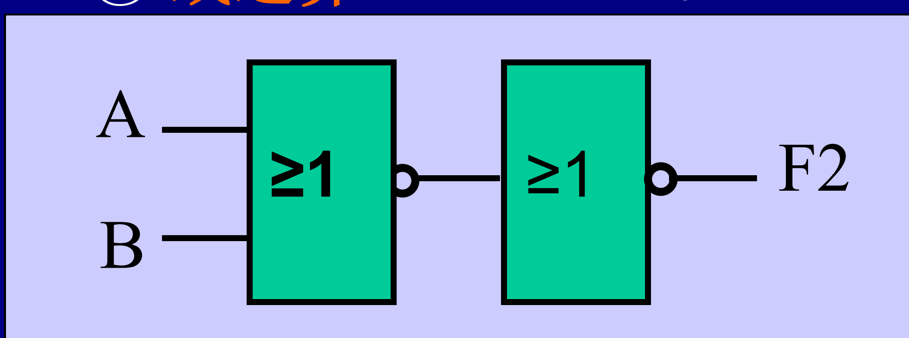


可以用**或非门**实现三种基本运算：

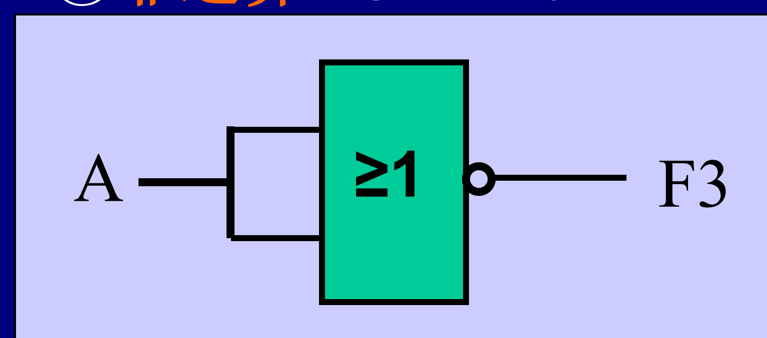
① **与运算** $F1=\overline{\overline{A+A}+\overline{B+B}}=\overline{\bar{A}+\bar{B}}=A\cdot B$



② **或运算** $F2=\overline{\overline{A+B}}$



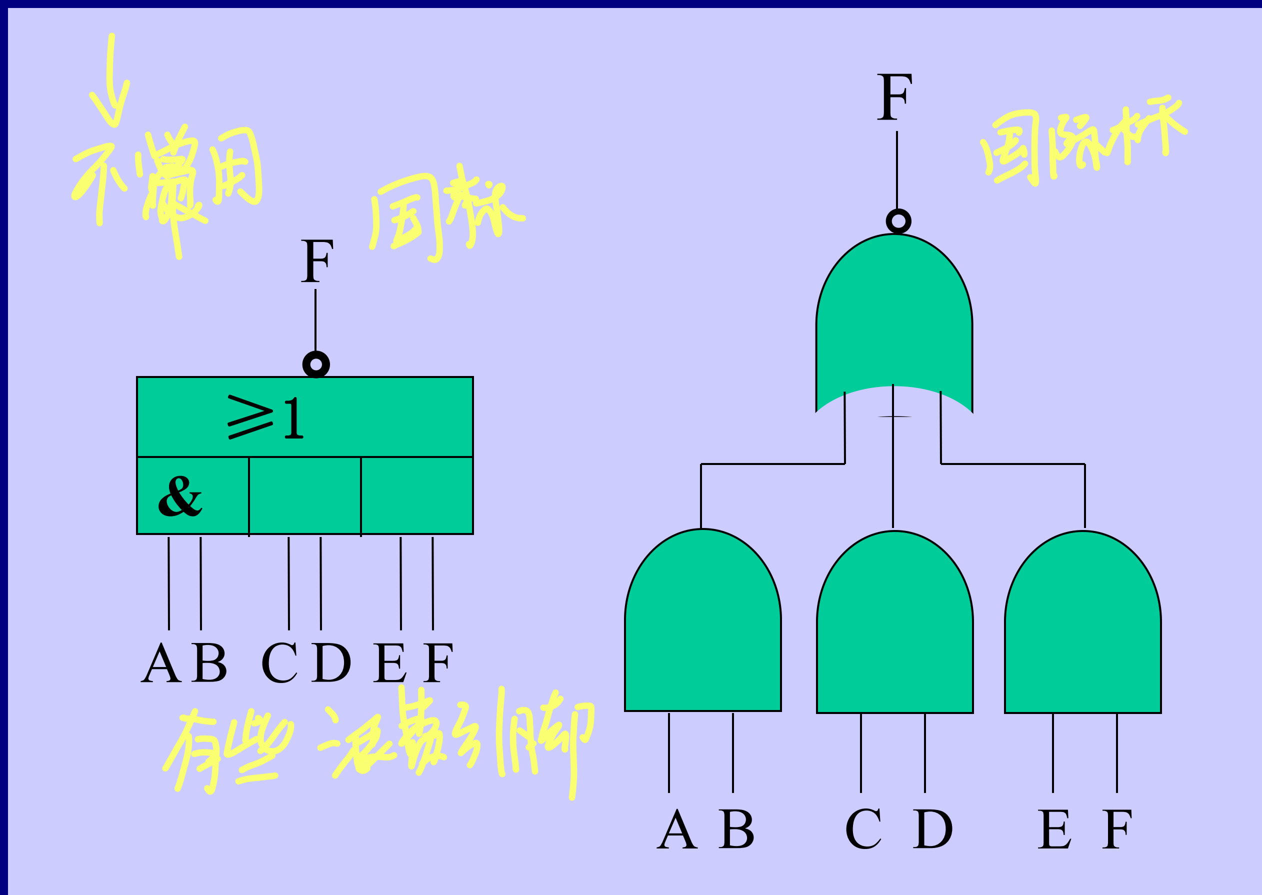
③**非运算** $F3=\overline{A+A}=\bar{A}$



3. 与或非逻辑(AOI)

逻辑表达式为：$F = \overline{AB + CD + EF}$

与或非门的逻辑符号



4. 异或逻辑(XOR)

逻辑表达式为: $F = A \oplus B = \overline{A} \cdot B + A \cdot \overline{B}$

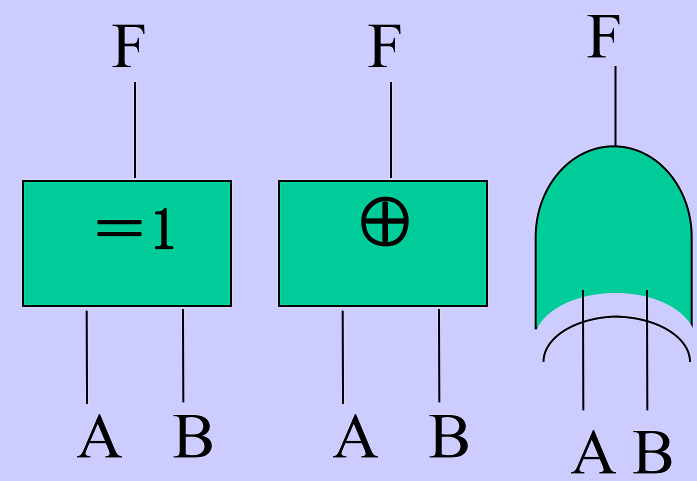
$\overline{A} \cdot B$

异或逻辑真值表

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

长得一样:0

异或门的逻辑符号



AB | $\overline{A}B$ | $A\overline{B}$ |

5. 同或逻辑(NXOR) 小风

逻辑表达式为： $F = A \odot B = \overline{A} \cdot \overline{B} + A \cdot B$

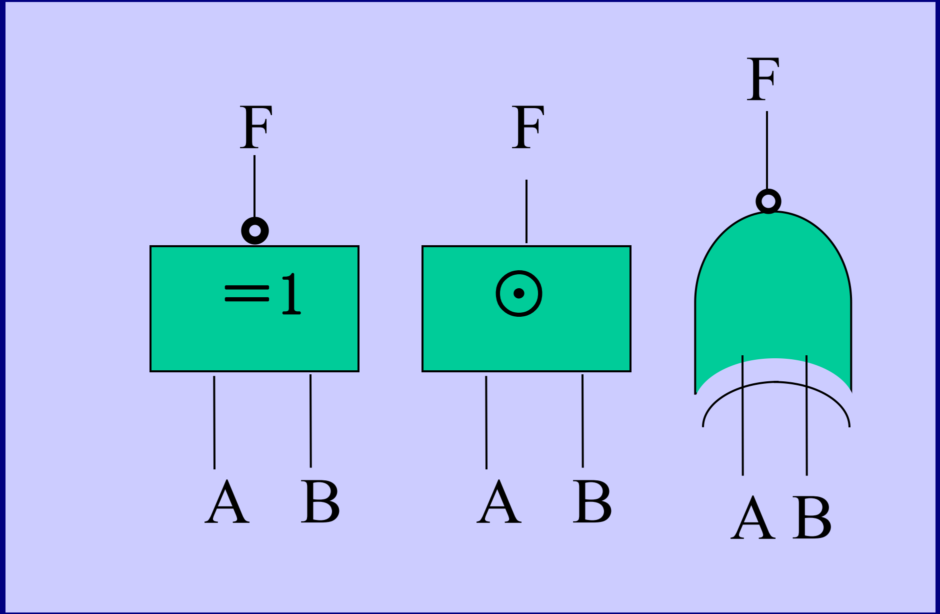
长一样

同或逻辑真值表

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

但不为异或反函

同或门的逻辑符号



异或运算与同或运算的关系 $A\oplus B=\bar{A}B+A\bar{B}$

1. $\overline{A\oplus B}=A\odot B$　　$\overline{A\odot B}=A\oplus B$ 互

2. $(A\oplus B)'=A\odot B$　　$(A\odot B)'=A\oplus B$

3. $A\oplus B\oplus C=A\odot B\odot C$　　互 对偶

例：证明

$$
\begin{aligned}
\overline{A\oplus B} &= \overline{\bar{A}B+A\bar{B}} \\
&= \overline{\bar{A}B}\;\overline{A\bar{B}} \\
&= (A+\bar{B})(\bar{A}+B) \\
&= \bar{A}\bar{B}+AB \\
&= A\odot B
\end{aligned}
$$

证明

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

数字逻辑电路

数字逻辑电路

证明 $A \oplus B \oplus C = \overline{A}(B \oplus C) + A\overline{(B \oplus C)}$ 只关心奇个

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

要么2非 $= \overline{A}(\overline{B}C + B\overline{C}) + A(\overline{B}\,\overline{C} + BC)$

要么0非 $= \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\,\overline{C} + ABC$

001 010 100 111

由上式可知，任意个变量的异或运算，只要输入为 1 的个数是奇数时，输出必为1，即为奇校验逻辑。

同或. 关心0

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

$A \oplus B \oplus C \oplus D$

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

要么3非 $= \overline{A}(\overline{B}C + B\overline{C}) + A(\overline{B}\,\overline{C} + BC)$

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

所以： 1非

当变量为2时，同或运算与异或运算的之间具有互补关系；

当变量为3时，同或运算与异或运算的之间具有相等关系。

由代入规则可以证明：

当变量为偶数时，同或运算与异或运算之间具有互补关系；
当变量为奇数时，同或运算与异或运算之间具有相等关系。
即：

$$A_1 \oplus A_2 \oplus A_3 \oplus \cdots \oplus A_n = \overline{A_1 \odot A_2 \odot A_3 \odot \cdots \odot A_n}$$

n 为偶数

$$A_1 \oplus A_2 \oplus A_3 \oplus \cdots \oplus A_n = A_1 \odot A_2 \odot A_3 \odot \cdots \odot A_n$$

n 为奇数

异或运算和同或运算的基本代数性质

异或

1 起作用

0—1律 (a) $A\oplus 0 = A$ $\qquad A\oplus 1 = \overline{A}$

(b) $A\odot 0 = \overline{A}$ $\qquad A\odot 1 = A$

同或 0 起作用

交换律 (a) $A\oplus B = B\oplus A$

(b) $A\odot B = B\odot A$

分配律 (a) $A(B\oplus C) = (AB)\oplus(AC)$

不用记

(b) $A+(B\odot C) = (A+B)\odot(A+C)$

去括展开

结合律 (a) $A\oplus(B\oplus C) = (A\oplus B)\oplus C$

(b) $A\odot(B\odot C) = (A\odot B)\odot C$

调换律 (a)若 $A\oplus B = C$则 $A\oplus C = B$， $C\oplus B = A$

(b) 若$A\odot B = C$则 $A\odot C = B$， $C\odot B = A$

神奇的异或

由于反码和补码在进行求和运算时，符号位也参与运算，所以不能根据求和时是否有进位而判断是否溢出，因此引入双符号位S_1S_0。

例　求 z = x ? y， 其中 x =＋0111， y =＋1010

则$[x]_{补}$= 000111　　$[y]_{补}$=001010

$[z1]_{补}=[x-y]_{补}=[x]_{补}+[-y]_{补}=$ 000111+110110=111101

$[z2]_{补}=[-x+y]_{补}=[-x]_{补}+[y]_{补}=$ 111001+001010=000011

$[z3]_{补}=[x+y]_{补}=[x]_{补}+[y]_{补}=$ 000111+001010=010001

$[z4]_{补}=[-x-y]_{补}=[-x]_{补}+[-y]_{补}=$

111001+110110=101111

故：判断溢出函数$F=S_1\oplus S_0$

数字逻辑电路

- 逻辑代数， 布尔代数， 开关代数
- **基本概念**： 正逻辑约定
  逻辑函数的4种表示方法
- **基本运算**： 与， 或， 非， 特别留意逻辑符号
- **基本公理**： 注意运算次序(括号、非、与、或），
  或对与的分配律 $A+B\cdot C=(A+B)(A+C)$
- **基本定理**： 吸收律、反演律、包含律

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

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

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

- **基本规则**： 代入规则、反演规则、对偶规则
- **复合逻辑**： 与非、或非、异或、同或

$1 \oplus 0 \oplus 1 \oplus 1 \oplus \cdot$ 模2和

$1 \odot 0 \odot 1 \odot 1 \odot 1$ 有几个"0"/"1"

- 作业

  求反函数， 求对偶函数

1.3.4.2 逻辑函数的基本表达式

Algebraic Forms of Switching Functions

依照逻辑运算的规则，一个逻辑命题可以用多种形式的逻辑函数来描述，而这些逻辑函数的真值表都是相同的。

如： $F = A \oplus B$ 　　$\overline{\overline{AB}\cdot\overline{AB}}$ 标准.

$= \bar{A}B + A\bar{B}$ 　　与或式 ***SOP***

perhaps or

$= (A+B)(\bar{A}+\bar{B})$ 　　或与式 ***POS***

要求.看见这个.反应出来用几个与非门

$= \overline{\overline{\bar{A}B}\ \overline{A\bar{B}}}$ 　　5个与非门 　　与非式

内用一次摩根反演

$= \overline{\overline{(A+B)}+\overline{(\bar{A}+\bar{B})}}$ 　　或非式

$= \overline{\bar{A}\bar{B} + AB}$ 　　与或非式

$= \ldots\ldots$

与2 非1 或2.

数字逻辑电路

1.3.4.3 逻辑函数的标准形式

Canonical Forms of Switching Functions

一个逻辑命题的三种表示法：真值表 →
表达式 → AB.
卡诺图 →

4个情况

三者之间的关系：

①真值表是逻辑函数最基本的表达方式，具有唯一性；

②由真值表可以导出逻辑表达式和卡诺图；唯一.

③由真值表导出逻辑表达式的两种标准形式：

最小项之和 ***The canonical SOP(the Sum Of Products)***

最大项之积 ***The canonical POS(Product Of Sums)***

1. 最小项 *minterm*

设有 n 个变量，它们组成的与项中每个变量或以原变量或以反变量形式出现且仅出现一次，此与项称之为 n 个变量的最小项。

全出现.全与.若干非.

对于 n 个变量就可构成 2^n 个最小项，分别记为 m_i。

其中：下标值 i 的取值为当各个最小项变量按一定顺序排好后，用 1 代替其中的原变量，0 代替其中的反变量，便得到一个二进制数，该二进制数的等值十进制即为 i 的值。

考编号

例如：3 变量的 8 个最小项可以表示为：与最小

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

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

$F = m_0 + m_1 + m_2$

为区别不同 n 值的相同 m_i，可记为：m_i^n

只需有个有1 F就为1

数字逻辑电路

2. 最大项 *maxterm*

设有 n 个逻辑变量，它们组成的或项中，每个变量或以原变量形式或以反变量形式出现且仅出现一次，此或项称为 n 变量的最大项。

对于 n 个变量可以构成2^n个最大项，分别记为 M_i 。

其中：下标值 i的取值规则与最小项中 i的取值规则相反，即将各变量按一定次序排好后，用 0 代替其中的原变量，用 1 代替其中的反变量，得到一个二进制数，该二进制数的等值十进制即为 i 的值。

或项(不带反)

例如，三变量的最大项记为：

$A+B+C=M_0$ $A+B+\overline{C}=M_1$ $A+\overline{B}+C=M_2$

$A+\overline{B}+\overline{C}=M_3$ $\overline{A}+B+C=M_4$ $\overline{A}+B+\overline{C}=M_5$

$\overline{A}+\overline{B}+C=M_6$ $\overline{A}+\overline{B}+\overline{C}=M_7$ $F= M_0 \cdot M_1 M_3 \cdot$

为区别不同 n 值的相同 M_i，可记为： $M_i^{\,n}$ 积.

数字逻辑电路

3. 最小项与最大项的性质

例：一个三变量函数 F(A，B，C)，它的真值表及其最小项及最大项的对应关系如下表。

行号	A B C	F(A，B，C)	最小项及代号	最大项及代号
0	000	F(0,0,0)=1	$\overline{A}\,\overline{B}\,\overline{C} = m_0$	$A+B+C = M_0$
1	001	F(0,0,1)=0	$\overline{A}\,\overline{B}\,C = m_1$	$A+B+\overline{C} = M_1$
2	010	F(0,1,0)=0	$\overline{A}\,B\,\overline{C} = m_2$	$A+\overline{B}+C = M_2$
3	011	F(0,1,1)=1	$\overline{A}\,B\,C = m_3$	$A+\overline{B}+\overline{C} = M_3$
4	100	F(1,0,0)=1	$A\,\overline{B}\,\overline{C} = m_4$	$\overline{A}+B+C = M_4$
5	101	F(1,0,1)=0	$A\,\overline{B}\,C = m_5$	$\overline{A}+B+\overline{C} = M_5$
6	110	F(1,1,0)=1	$A\,B\,\overline{C} = m_6$	$\overline{A}+\overline{B}+C = M_6$
7	111	F(1,1,1)=1	$A\,B\,C = m_7$	$\overline{A}+\overline{B}+\overline{C} = M_7$

数字逻辑电路

最小项与最大项具有如下性质：

(1) 对于任意最小项，只有一组变量组合取值可使其值为1；
对于任意最大项，只有一组变量组合取值可使其值为0。
（用Venn图示意出m_i和M_i的区域）

(2) n变量的所有最小项之和必为1，记为：$\sum_{i=0}^{2^n-1} m_i = 1$

n变量的所有最大项之积必为0，记为：$\prod_{i=0}^{2^n-1} M_i = 0$

(3) 任意两个最小项之积必为0，即：$m_i \cdot m_j = 0 (i \neq j)$

任意两个最大项之和必为1，即：$M_i + M_j = 1 (i \neq j)$

由上表可知，最小项与最大项具有如下性质

(4) 同变量数下标相同的最小项和最大项互为反函数

即： $m_i = \overline{M_i}$ $M_i = \overline{m_i}$ 摩根定理

则： $m_i \cdot M_i = 0$ 且 $m_i + M_i = 1$

数字逻辑电路

4. 函数的最小项标准式

逻辑函数被表达成一系列乘积项之和，则称之为积之和表达式(SOP)，或称为与或表达式。

如果构成函数的积之和表达式中每一个乘积项(与项)均为最小项时，则这种表达式称之为最小项标准式(*The canonical SOP*)，且这种表示是唯一的。

卡诺会指掉这些标准式

$B/\bar{B}$

如：$F(A,B,C) = \boxed{AC} + \bar{A}B + BC$

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

吸收 or 分配律

$= m_2 + m_3 + m_5 + m_7$

$= \sum m^3(2,3,5,7)$

写出逻辑函数的最小项标准式的方法：

① 如果给定的函数为一般的与或表达式，可以反复应用公式$X=X(\overline{Y}+Y)$ 代入缺少某变量(Y)的与项中，形成最小项之和的形式。

例：
$$
\begin{aligned}
F &= AC + A\overline{B} + BC \\
&= AC(\overline{B}+B) + A\overline{B}(\overline{C}+C) + (\overline{A}+A)BC \\
&= A\overline{B}C + ABC + A\overline{B}\,\overline{C} + A\overline{B}C + \overline{A}BC + ABC \\
&= A\overline{B}\,\overline{C} + \overline{A}BC + A\overline{B}C + ABC \\
&= \sum m^3(3,4,5,7)
\end{aligned}
$$

老实拆呗，
没什么窍门

例：一个三变量函数 F(A，B，C)，它的真值表及其最小项及最大项的对应关系如下表。

行号	A B C	F(A，B，C)	最小项及代号	最大项及代号
0	000	F(0,0,0)=1	$\overline{A}\,\overline{B}\,\overline{C} = m_0$	$A+B+C = M_0$
1	001	F(0,0,1)=0	$\overline{A}\,\overline{B}\,C = m_1$	$A+B+\overline{C} = M_1$
2	010	F(0,1,0)=0	$\overline{A}\,B\,\overline{C} = m_2$	$A+\overline{B}+C = M_2$
3	011	F(0,1,1)=1	$\overline{A}\,B\,C = m_3$	$A+\overline{B}+\overline{C} = M_3$
4	100	F(1,0,0)=1	$A\,\overline{B}\,\overline{C} = m_4$	$\overline{A}+B+C = M_4$
5	101	F(1,0,1)=0	$A\,\overline{B}\,C = m_5$	$\overline{A}+B+\overline{C} = M_5$
6	110	F(1,1,0)=1	$A\,B\,\overline{C} = m_6$	$\overline{A}+\overline{B}+C = M_6$
7	111	F(1,1,1)=1	$A\,B\,C = m_7$	$\overline{A}+\overline{B}+\overline{C} = M_7$

② 如果给定函数用真值表表示，显然**真值表每一行**变量的组合**对应一个最小项**。如果对应该行的**函数值为 1**，则函数的最小项表达式中**应包含**该行对应的最小项；如果该行的**函数值为 0**，则函数的最小项表达式中**不包含**对应该行的最小项。

例：对应前表中的F(A,B,C)，其最小项表达式应为：

$$F(A,B,C) = m_0 + m_3 + m_4 + m_6 + m_7$$

$$= \sum m^3(0,3,4,6,7)$$

显然 $\overline{F}(A,B,C) = m_1 + m_2 + m_5$

$$= \sum m^3(1,2,5)$$

反函，内做
互补

③ 如果给定函数用卡诺图表示，则卡诺图上的每一块区域对应一个最小项。如果对应该区域的函数值为 1 ，则函数的最小项表达式中应包含该区域对应的最小项；如果该区域的函数值为 0，则函数的最小项表达式中不包含对应该区域的最小项。

最小项与原函数、反函数的关系

对于 n 个变量的函数 F，它共有2^n个最小项，这些最小项不是包含在原函数 F 的最小项表达式中，就是包含在反函数 $\overline{F}$的最小项表达式中。

用逻辑代数可以证明：

$$F(x_1,x_2,x_3,\ldots,x_n)+\overline{F}(x_1,x_2,x_3,\ldots,x_n)=\sum_{i=0}^{2^n-1} m_i=1$$

5. 函数的最大项标准式 不仔细讲.缺什么补什么

逻辑函数被表达成一系列**和项之积**，则称为**和之积表达式(POS)**，或称为**或与表达式**。

如果构成函数的或与表达式中的**每一个和项**均为**最大项**，则这种表达式称为**最大项标准式 (*The Canonical POS)*** ，且这种表示是**唯一**的。

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

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

$$= M_0 \cdot M_2 \cdot M_4 \cdot M_5$$

$$= \prod M^3(0,2,4,5)$$

写出逻辑函数的最大项标准式的方法：

① 如果给定的函数是一般的或与表达式，可以反复应用公式：$X = X + \overline{Y} \cdot Y = (X + Y)(X + \overline{Y})$ 代入缺少某变量(Y)的和项中以形成最大项之积的形式。写成或与式（掌握

例： $F = (A + C)(\overline{A} + B)$

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

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

$$= M_0 \cdot M_2 \cdot M_4 \cdot M_5$$

$$= \prod M^3(0,2,4,5)$$

例：一个三变量函数 F(A，B，C)，它的真值表及其最小项及最大项的对应关系如下表。

行号	A B C	F(A，B，C)	最小项及代号	最大项及代号
0	000	F(0,0,0)=1	$\bar{A}\bar{B}\bar{C} = m_0$	$A+B+C = M_0$
1	001	F(0,0,1)=0	$\bar{A}\bar{B}C = m_1$	$A+B+\bar{C} = M_1$
2	010	F(0,1,0)=0	$\bar{A}B\bar{C} = m_2$	$A+\bar{B}+C = M_2$
3	011	F(0,1,1)=1	$\bar{A}BC = m_3$	$A+\bar{B}+\bar{C} = M_3$
4	100	F(1,0,0)=1	$A\bar{B}\bar{C} = m_4$	$\bar{A}+B+C = M_4$
5	101	F(1,0,1)=0	$A\bar{B}C = m_5$	$\bar{A}+B+\bar{C} = M_5$
6	110	F(1,1,0)=1	$AB\bar{C} = m_6$	$\bar{A}+\bar{B}+C = M_6$
7	111	F(1,1,1)=1	$ABC = m_7$	$\bar{A}+\bar{B}+\bar{C} = M_7$

②如果给定函数用真值表表示，显然真值表每一行变量的组合对应一个最大项。如果对应该行的函数值为 0 ，则函数的最大项表达式中应包含该行对应的最大项；如果该行的函数值为 1 ，则函数的最大项表达式中不包含对应该行的最大项。

例：对应前表中的F(A,B,C)，其最大项表达式应为：

$$F(A,B,C) = M_1 \cdot M_2 \cdot M_5$$
$$= \prod M^3(1,2,5)$$

显然 $\overline{F}(A,B,C) = M_0 \cdot M_3 \cdot M_4 \cdot M_6 \cdot M_7$

$$= \prod M^3(0,3,4,6,7)$$

③ 如果给定函数用卡诺图表示，则函数的最大项表达式可以通过卡诺图得到。

数字逻辑电路

最大项与原函数、反函数的关系

对于 n 个变量的函数 F，它共有2^n个最大项，这些最大项不是包含在原函数 F 的最大项表达式中，就是包含在反函数 F 的最大项表达式中。

可以证明：

$$F(x_1,x_2,x_3,\ldots,x_n)\bullet\overline{F}(x_1,x_2,x_3,\ldots,x_n)=\prod_{i=0}^{2^n-1}M_i=0$$

同一函数的最小项标准式与其最大项标准式的关系：

同一逻辑函数的一种标准式变换成另一种标准式时，互换$\sum m^n$和$\prod M^n$的符号，并在符号后列出原式中缺少的那些数字。而且这两种标准式都是唯一的。

例： $F = \sum m^3(0,2,3) = \prod M^3(1,4,5,6,7)$

证明： $F = \sum m^3(0,2,3)$

则 $\overline{F} = \sum m^3(1,4,5,6,7)$

$$= m_1 + m_4 + m_5 + m_6 + m_7$$

$$F = \overline{\overline{F}} = \overline{m_1 + m_4 + m_5 + m_6 + m_7}$$

$$= \overline{m_1} \cdot \overline{m_4} \cdot \overline{m_5} \cdot \overline{m_6} \cdot \overline{m_7}$$

$$= M_1 \cdot M_4 \cdot M_5 \cdot M_6 \cdot M_7$$

$$= \prod M^3(1,4,5,6,7)$$

- **总结逻辑函数的性质**
- 复合逻辑——逻辑完备性

  与非门的速度最快，代价最低，最常用

  器件的符号

  器件的0、1控制作用

  异或、同或的特点
- 逻辑函数的基本表达式——不唯一

  与或式- 与非式

  简单应用，与或式需要多少个与非门实现
- 逻辑函数的标准形式——最小项之和，最大项之积

  最小项、最大项对下标的规定 会考

  最小项、最大项的性质

  如何写最小项、最大项标准式（从与或式、真值表、卡诺图）

  $F=\sum m(\ )=\prod M(\ )$

  $\overline{F}=\sum m(\ )=\prod M(\ )$ 作业

数字逻辑电路

1.3.5 逻辑函数的化简

Simplification of Switching Expression

一个逻辑函数对应着一个实现其逻辑功能的逻辑电路，当使该函数最简意味着使这个电路也最简。

标准式太杂了 希望用简洁

- 最简逻辑电路：门数最少；门的输入端最少；门的级数最少。

卡诺 只有

- 最简与或式：与项的数目最少；每个与项的变量个数最少。

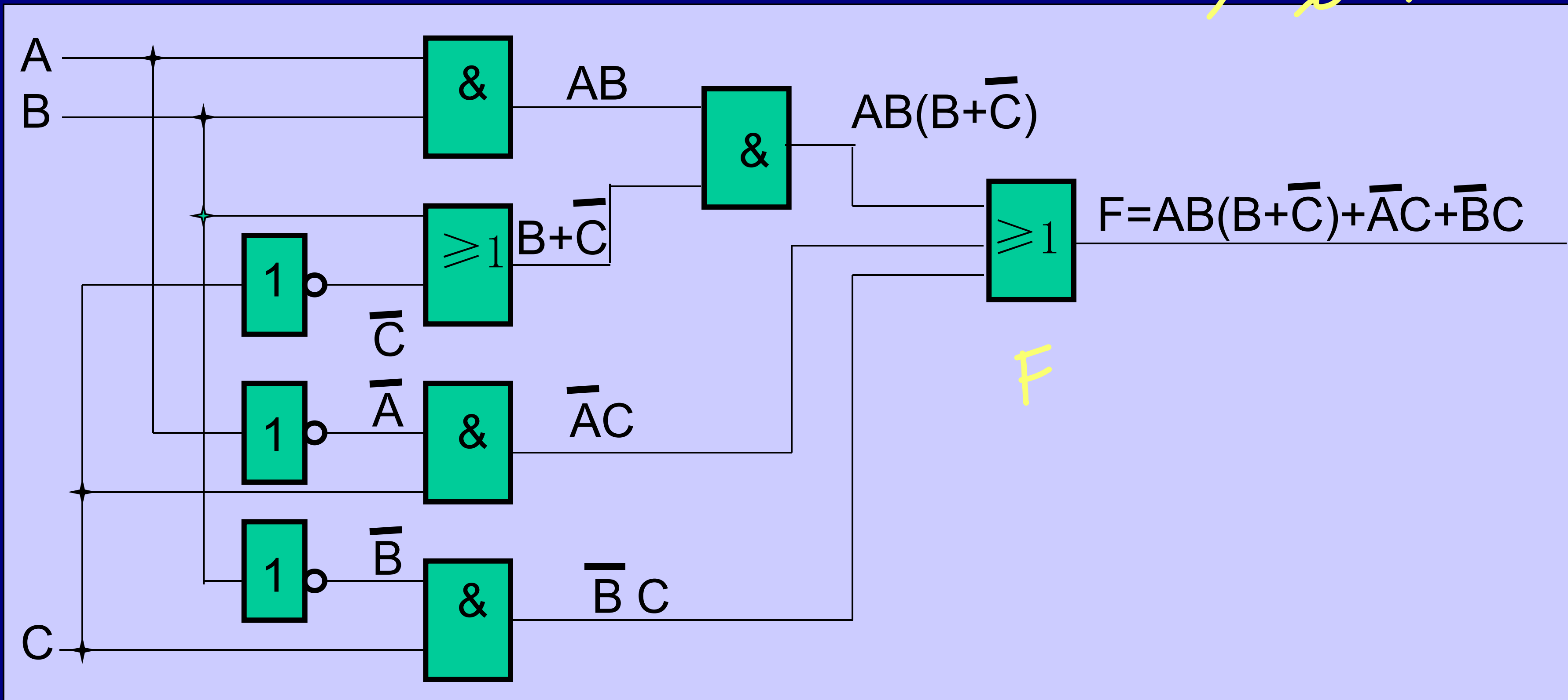
门电级别 后集时间

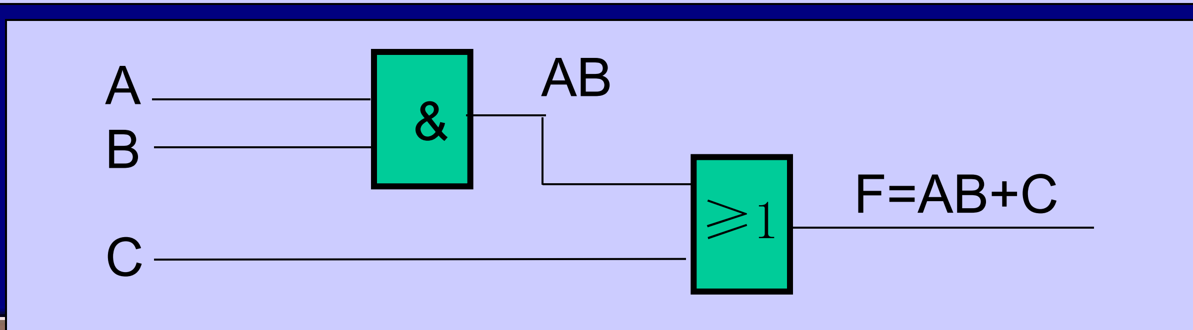
- 最简或与式：或项的数目最少；每个或项的变量个数最少。

级数少

例： $F = AB(B+\overline{C}) + \overline{A}C + \overline{B}C \;=\; AB + C$

对应两种逻辑电路图，如下 必定好





1.3.5.1 代数化简法：

只供观赏

一、与或式的化简

不考试

与或式化简常用的公式主要有四个：

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

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

所谓化简过程就是运用代入规则，把某一子函数看成一个变量，进而应用公式简化，在这一过程中经常需要变换子函数的形式，以便能够应用公式进行简化，最终将函数化简为**最简与或式 *(minimizing SOP)***，常用的方法有如下几种：

1. 吸收法

① 利用公式：$A+AB=A$ 消去多余变量。例：

$$F = AB + AB\cdot(C+D)\cdot E$$
$$= AB$$

② 利用公式：$A+\overline{A}B = A+B$ 消去反变量。例：

$$F = AB + \overline{A}C + \overline{B}C$$
$$= AB + (\overline{A} + \overline{B})C$$
$$= AB + \overline{AB}\,C \quad \rightarrow \text{吸收律}$$
$$= AB + C$$

2. 合并项法

利用公式： $AB + A\overline{B} = A$ ，两项合并为一项且消去一个变量。例：

$$F = \sum m^4(5,7,13,15)$$

$$= \overline{A}B\overline{C}D + \overline{A}BCD + AB\overline{C}D + ABCD$$

$$= \overline{A}BD + ABD$$

$$= BD$$

3. 消去法

利用公式：$AB + \overline{A}C + BC = AB + \overline{A}C$ 消去多余项。

例：$F = AC + ADE + \overline{C}D$

$$= AC + \overline{C}D + AD + ADE$$

$$= AC + \overline{C}D$$

4. 配项法

当无法发现直接应用公式时，可先增加一些与项，再利用增加项消除多余项，即“先繁后简”。

① 利用公式：$1 = A + \overline{A}$ 增加项数。例：

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

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

不好把握

增加，

2个化简结果

做不出第三种

② 利用公式： $AB+\bar{A}C=AB+\bar{A}C+BC$ 增加项数。

多余

例：

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

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

5. 综合法

在一个函数的化简中同时用几种方法。例：

$$F = AB + A\bar{C} + \bar{B}C + \bar{C}B + \bar{B}D + \bar{D}B + ADE(F+G)$$

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

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

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

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

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

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

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

二、或与式的化简

或与式化简有两种方法：

1. 常规法

常规法化简方式类似于与或式的化简，化简中常用的公式主要有：

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

$$A(A+B)=A$$

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

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

例：$F=A(\underline{A}+B)(\overline{A}+D)(\overline{B}+D)(A+C+E+H)$

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

$$=AD(\overline{B}+D)$$

$$=AD$$

2. 利用最简与或式得到最简或与式

① 二次对偶法

利用对偶规则，先求出F的对偶式F'，再将对偶式F '化简为最简与或式，最后再求一次对偶(F ') ' = F，则得到F 最简或与式。

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

$$F' = AB + A\overline{B} + BC + \overline{C}D + BD$$

$$= A + BC + \overline{C}D + BD$$

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

则 $F = (F')' = A(B + C)(\overline{C} + D)$ 最简或与式

② 二次求反法

利用反演规则，先求出F的反函数 $\overline{F}$，再将反函数 $\overline{F}$ 化简为**最简与或式**，最后再求一次反 $\overline{\overline{F}} = F$，则得到 F 最简或与式。

例： $F = AB + \overline{A}\,\overline{B} + \overline{A}\,\overline{C} + B\overline{D}$

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

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

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

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

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

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

不要求
只有会用
行

- 卡诺图 化简 逻辑函数
  - 一至六变量的卡诺图
  - 卡诺图的构成特点
  - 逻辑函数在卡诺图 上的表示
  - 用卡诺图 化简 逻辑函数的基本原理

1.3.5.2 卡诺图（*Karnaugh MAP*）法(1953)

卡诺图逻辑函数真值表的一种图形表示，利用卡诺图可以有规律地化简逻辑函数表达式，并能直观地写出逻辑函数的最简式。

一、卡诺图的构成

卡诺图是一种平面方格阵列图，下页给出了二变量、三变量、四变量、五变量和六变量的卡诺图。

逻辑区域一分为大

数字逻辑电路

①单变量的卡诺图

0	1
$\overline{A}$	A

分别对应着两个最小项

$$m_0 = \overline{A} \qquad m_1 = A$$

引入变量A，将区域分为两块

②二变量的卡诺图

A

0 $\overline{A}\,\overline{B}$	2 $A\overline{B}$
1 $\overline{A}B$	3 AB

10   11   01

再引入变量B，将区域分为四块

分别对应着四个最小项

$$m_0 = \overline{A}\,\overline{B},\quad m_1 = \overline{A}B,$$

$$m_2 = A\overline{B},\quad m_3 = AB。$$

数字逻辑电路

③三变量的卡诺图

00 01 11 10

		A	A	
0	2	6	4	
1	3	7	5	C
	B	B		

A A 一分为2

相邻最小项可合并

A是B是C是

再引入变量C，将区域分为八块，分别对应着**8个最小项**：

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

$m_4=A\bar{B}\bar{C}$，$m_5=A\bar{B}C$，$m_6=AB\bar{C}$，$m_7=ABC$。

数字逻辑电路

④四变量的卡诺图

Gray码，
01，11 10
相邻

A 右2.

0	4	12	8
1	5	13	9
3	7	15	11
2	6	14	10

C1

D

C

B B'B'

四变量(A、B、C、D)的卡诺图，分别对应着16个最小项：

$m_0=\overline{A}\,\overline{B}\,\overline{C}\,\overline{D}$，　$m_1=\overline{A}\,\overline{B}\,\overline{C}D$，　$m_2=\overline{A}\,\overline{B}C\overline{D}$，　$m_3=\overline{A}\,\overline{B}CD$,

$m_4=\overline{A}B\overline{C}\,\overline{D}$，　$m_5=\overline{A}B\overline{C}D$，　$m_6=\overline{A}BC\overline{D}$，　$m_7=\overline{A}BCD$,

$m_8=A\overline{B}\,\overline{C}\,\overline{D}$，　$m_9=A\overline{B}\,\overline{C}D$，　$m_{10}=A\overline{B}C\overline{D}$,　$m_{11}=A\overline{B}CD$

$m_{12}=AB\overline{C}\,\overline{D}$，　$m_{13}=AB\overline{C}D$,　$m_{14}=ABC\overline{D}$，　$m_{15}=ABCD$

⑤五变量的卡诺图

90%、可 位置

考.

右2

A 右切

B 右2切

0	4	12	8
1	5	13	9
3	7	15	11
2	6	14	10

16	20	28	24
17	21	29	25
19	23	31	27
18	22	30	26

B, E, C, D, C

下切

中切

中切

五变量(A、B、C、D、E)的卡诺图，分别对应着32个最小项。

数字逻辑电路

⑥六变量的卡诺图:有64个最小项

B

C C

F

0	4	12	8
1	5	13	9
3	7	15	11
2	6	14	10

16	20	28	24
17	21	29	25
19	23	31	27
18	22	30	26

E

F

32	36	44	40
33	37	45	41
35	39	47	43
34	38	46	42

48	52	60	56
49	53	61	57
51	55	63	59
50	54	62	58

A

E

D D

卡诺图的构成特点：

(1) 整个卡诺图总是被每个变量逐次地分成两半：原变量，反变量各占一半。任一变量的原变量和反变量所占的区域又被其他变量分成两半。n变量：一个人.n个邻居

			A	A	
	0	4	12	8	
D	1	5	13	9	
D	3	7	15	11	C
	2	6	14	10	C
		B	B		

卡诺图的行和列按照变量的组合标注方法，其变量顺序遵从真值表中变量从左至右的顺序。

卡诺图的构成特点：

(2) n 变量卡诺图有2^n个小方格(或称为单元)，每个小方格内左上角标注小方格名称或号，小方格名称由各行各列所对应的变量字母组成，小方格号的十进制数就是它对应的最小项的下标值；

A

0	4	12	8
1	5	13	9
3	7	15	11
2	6	14	10

D

C

B

卡诺图的构成特点：

(3) 在行和列上**相邻**的小方格，其名称仅有一个变量状态不同。

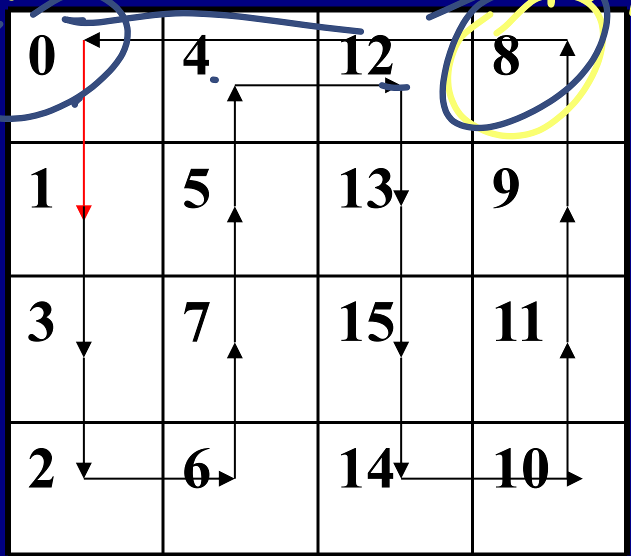
			A	A	
	0	4	12	8	
D	1	5	13	9	
D	3	7	15	11	C
	2	6	14	10	C
		B	B		

十进制数	二进制数	典型Gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

典型格雷码的产生过程

能回0与0邻接就可以生成

可以自己生成格雷



5w种 A_{16}^{16}

模m计数 偶数

数字逻辑电路

十进制数	十进制 Gray码(1)	十进制 Gray码(2)
0	0000	0000
1	0001	0001
2	0011	0011
3	0010	0010
4	0110	0110
5	1110	0111
6	1010	0101
7	1011	0100
8	1001	1100
9	1000	1000
10		
11		
12		
13		
14		
15		

模12

0	4	12	8
1	5	13	9
3	7	15	11
2	6	14	10

模10

0	4	12	8
1	5	13	9
3	7	15	11
2	6	14	10

卡诺图的构成特点：

(4) 若干个小方格将对应一个逻辑函数，即一个逻辑函数可以图示于卡诺图上。

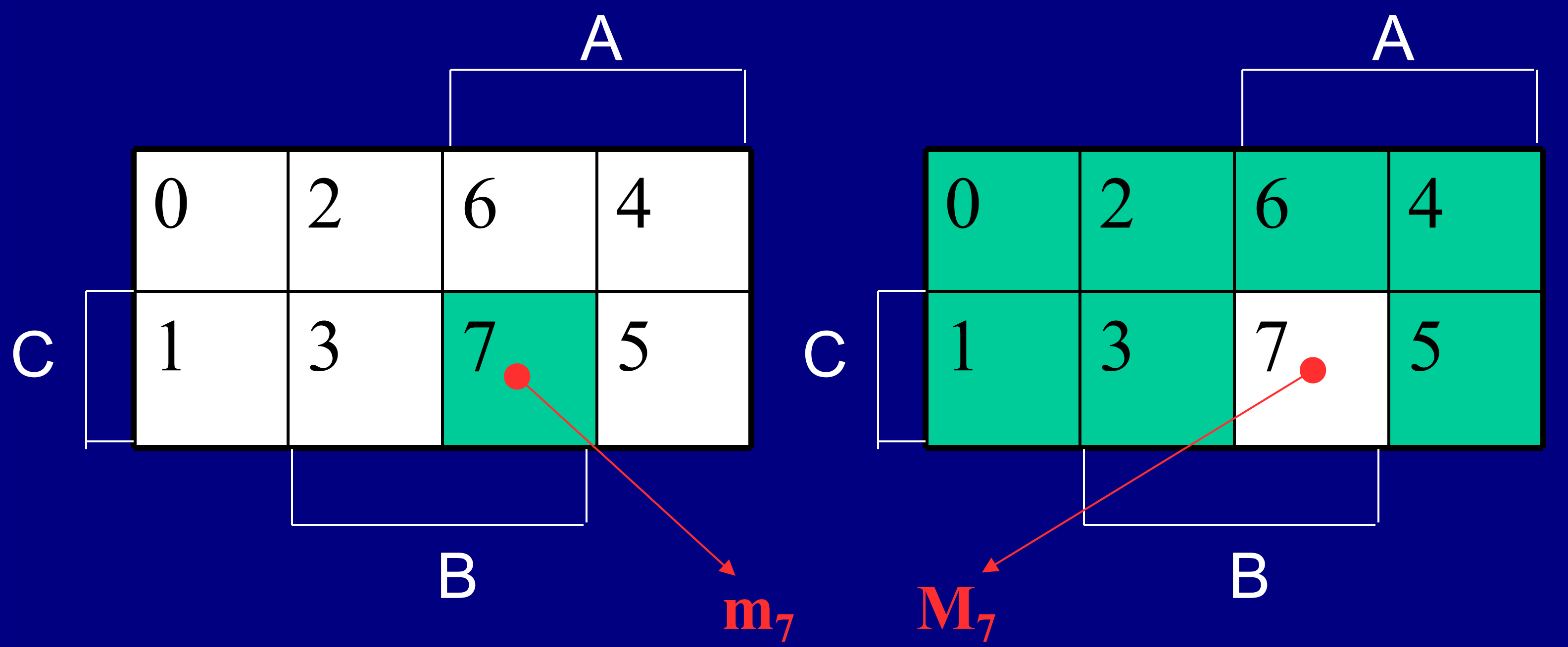
例：$F = m_1 + m_2 + m_5 + m_7$，其真值表和卡诺图标注如下：

行号	ABC	F	m_i
0	0 0 0	0	m_0
1	0 0 1	1	m_1
2	0 1 0	1	m_2
3	0 1 1	0	m_3
4	1 0 0	0	m_4
5	1 0 1	1	m_5
6	1 1 0	0	m_6
7	1 1 1	1	m_7

C \ A B				
	0	2 1	6	4
	1 1	3	7 1	5 1

卡诺图的构成特点：

(5) 除掉某个小方格(即最小项)以外的整个卡诺图区域对应一个最大项，最大项下标值就是应被除去的小方格号。



上述卡诺图构成的特点及最小项与最大项在卡诺图上的表示，可以进一步说明逻辑运算的几何含义：

- 两个函数相“与”，表示两个函数在卡诺图上所占两个区域的公共区域；
- 两个逻辑函数相“或”，表示两个函数所覆盖的全部区域；
- 一个逻辑函数的“非”，就是求该函数覆盖之外的区域。

二、逻辑函数在卡诺图上的表示

① 把给定的逻辑函数化为**最小项标准式**

② 按变量数画出相应**卡诺图**

③ 在对应于最小项标准式中各最小项的小方格内**标以“1”**

④ 所有标有“1”的小方格的**合成区域**就表示该函数。

例：$F_1 = A\overline{C} + ABC + \overline{B}C$

将函数化为标准式，即：

$$F_1 = \overline{A}\,\overline{B}C + A\overline{B}\,\overline{C} + A\overline{B}C + AB\overline{C} + ABC$$

$$= m_1 + m_4 + m_5 + m_6 + m_7$$

$$= \sum m^3\,(1,4,5,6,7)$$

F_1的卡诺图见下页。

$F_1 = \sum m^3\ (1,4,5,6,7)$
的卡诺图表示

AB / C; A

0	2	6 **1**	4 **1**
1 **1**	3	7 **1**	5 **1**

C; B

二、逻辑函数在卡诺图上的表示

2.按变量之间的关系直接画出相应卡诺图，基本方式是与或式填入，先与后或。

所有标有“1”的小方格的合成区域就表示该函数。

例：$F_2 = A\bar{B}C + \bar{A}C + BC$

直接按与、或的几何含义将函数标注到卡诺图上。

与或式填入法、

0	2	6	4
1　1	3　1	7　1	5　1

C.

$\bar{A}C$　　B　BC　　$A\bar{B}C$

例： $F_3 = (A + \overline{B})(\overline{C} + B)(\overline{A} + B)$

求出反函数的与或式，直接按与、或的共同区域将函数标注到卡诺图上，此时小方格应标注“0”，其余的标“1”，则标“1”小方格的集合就是原函数。

才行

对偶没用. 反演

则 $\overline{F_3} = \overline{A}B + C\overline{B} + A\overline{B}$

反函数在卡诺图上标0。

C \ AB	00	01	11	10
0	0: 1	2: 0	6: 1	4: 0
1	1: 0	3: 0	7: 1	5: 0

A

B

C

数字逻辑电路

数字逻辑电路

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

	1
1	

A非B，A B非

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

1	
	1

A非B非 + AB

$$F = A\oplus B\oplus C = \bar{A}\,\bar{B}\,C + \bar{A}\,B\,\bar{C} + A\,\bar{B}\,\bar{C} + ABC$$

奇数个

A

	1		1
1		1	

C

B

数字逻辑电路

不可以化简.

$F = A \oplus B \oplus C \oplus D$ ⟷ $F = A \odot B \odot C \odot D$

互补.

	00	01	11	10
00		1		1
01	1	双1	1	
11		1		1
10	1		1	

单1

卡诺. 规定1的个数3.

		A	
1		1	
	1		1
1		1	
	1		1

D C B

三、用卡诺图化简逻辑函数的基本原理

1. “相邻” 的判断

(1) 相邻最小项：任意两个最小项中只有一个变量不同（同一变量名但一个为原变量，另一个为反变量），其余变量完全相同，在图上反映的是两个相邻的小方格。

(2) 卡诺图相邻小方格：是指只隔一条边界的两个小方格。

在 n 变量的卡诺图上，每个小方格具有n个相邻的小方格，它们是：

- 具有共同边界的小方格(几何相邻)
- 同一幅卡诺图中分别处于行(或列)两端的小方格 (相对相邻)
- 在相邻两幅卡诺图中，处于相同位置的两个小方格 (相重相邻)

数字逻辑电路

n 变量卡诺图的每一个小方格都有 **n 个相邻块，**

以m_0为例说明如下：

0	1

n=1

2个邻

0	2
1	3

n=2

3个邻居

0	2	6	4
1	3	7	5

n=3

4个

0	4	12	8
1	5	13	9
3	7	15	11
2	6	14	10

n=4

5个邻

√ [illegible]

0	4	12	8	16	20	28	24
1	5	13	9	17	21	29	25
3	7	15	11	19	23	31	27
2	6	14	10	18	22	30	26

n=5

n 变量卡诺图的每一个小方格都有 n 个相邻块，

以m_0为例说明如下：

0	4	12	8	16	20	28	24
1	5	13	9	17	21	29	25
3	7	15	11	19	23	31	27
2	6	14	10	18	22	30	26
32	36	44	40	48	52	60	56
33	37	45	41	49	53	61	57
35	39	47	43	51	55	63	59
34	38	46	42	50	54	62	58

n=6

在真值表上，不易直观地理解最小项的相邻关系，而在卡诺图上则相邻关系一目了然。

2. 画卡诺圈的规则

- 寻找相邻块的目的是为了在图上进行函数化简。
- 任何2^i个（$i \le n$）标1的相邻小方格均可画在一个卡诺圈内；
- 任何2^i个标1的非相邻的小方格不能画入一个卡诺圈内，它们至少画在两个圈内。
- 标1/标0的相邻同维块可画在一个卡诺圈内；
- 标1/标0的非相邻同维块不能画入一个卡诺圈内，它们至少画在两个圈内。

数字逻辑电路

例：五变量的卡诺图

①② ③

2,4,8个

DE \ ABC	000	001	011	010	100	101	111	110
00	0 1	4	12 1	8 1	16 1	20	28	24
01	1	5	13 1	9 1	17	21	29 1	25 1
11	3	7 1	15	11	19	23	31	27
10	2 1	6	14	10 1	18	22 1	30 1	26

① 0维块：$m_7=\overline{A}\,\overline{B}CDE$，是五变量的与项。

② 1维块：由两个相邻的 0 维块构成的一个卡诺圈，

$$m_0+m_{16}=\overline{A}\,\overline{B}\,\overline{C}\,\overline{D}\,\overline{E}+A\overline{B}\,\overline{C}\,\overline{D}\,\overline{E}=\overline{B}\,\overline{C}\,\overline{D}\,\overline{E}$$

$$m_{22}+m_{30}=A\overline{B}CD\overline{E}+ABCD\overline{E}=ACD\overline{E}$$

是四变量的与项。

例：五变量的卡诺图

ABC / DE

0 1	4	12 1	8 1
1	5	13 1	9 1
3	7 1	15	11
2 1	6	14	10 1

16 1	20	28	24
17	21	29 1	25 1
19	23	31	27
18	22 1	30 1	26

③2维块：由四个相邻的 0 维块构成的一个卡诺圈，

$m_8+m_9+m_{12}+m_{13}=\overline{A}B\overline{D}$ 共6个项

$m_0+m_2+m_8+m_{10}=\overline{A}\,\overline{C}\,\overline{E}$

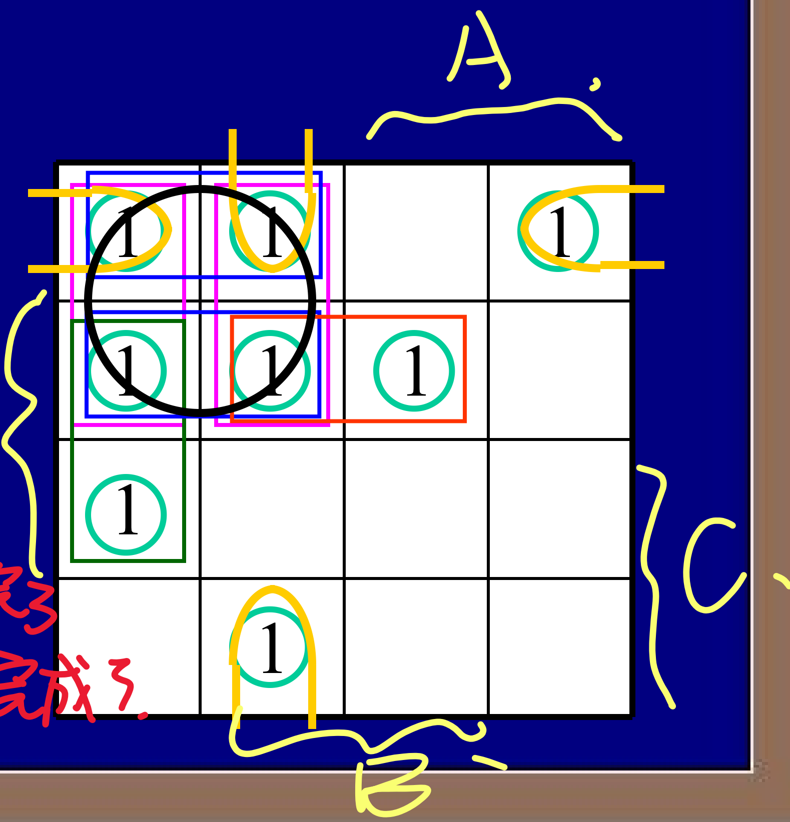
$m_9+m_{13}+m_{25}+m_{29}=B\overline{D}E$ 是三变量的与项。

对 n 变量而言，2^i个$(i\leq n)$标1的相邻小方格所构成的一个卡诺圈表示一个$(n-i)$变量的与项，即：使 2^i 个相邻的最小项被化简为一个$(n-i)$变量的与项。

用卡诺圈化简是有其特点和规律的。

①蕴函项 (*Implicant*):

- 在函数的与或表达式中，每一个与项称为该函数的蕴涵项，它对应着卡诺图中的一个卡诺圈。
- 卡诺圈越大，它所包含的相邻标1小方格越多，则对应此蕴涵项的变量数就越少。
- 卡诺圈既包含某变量的原变量区域，又包括它的反变量区域，则这个变量就不出现在此圈所对应的蕴涵项中。



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

画完了 就完成了.

数字逻辑电路

②质蕴函 (*Prime implicant*)：

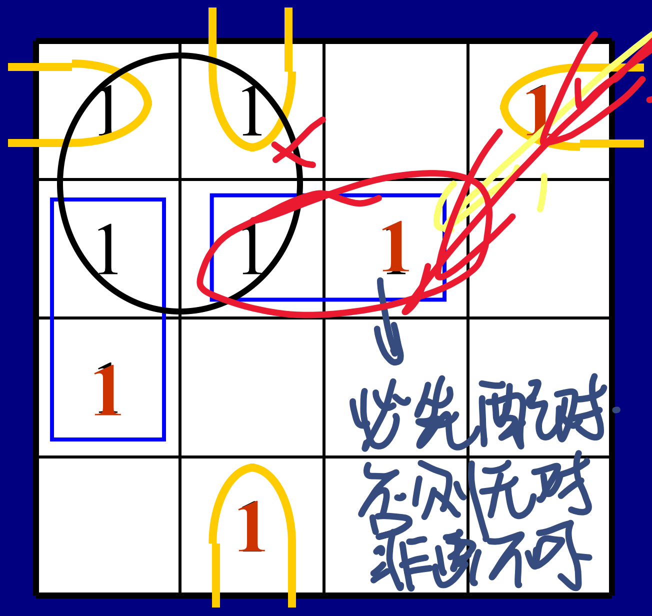
若蕴涵项不是其他蕴涵项的子集，则称为质蕴涵，又称为素项，在卡诺图中称为极大圈。

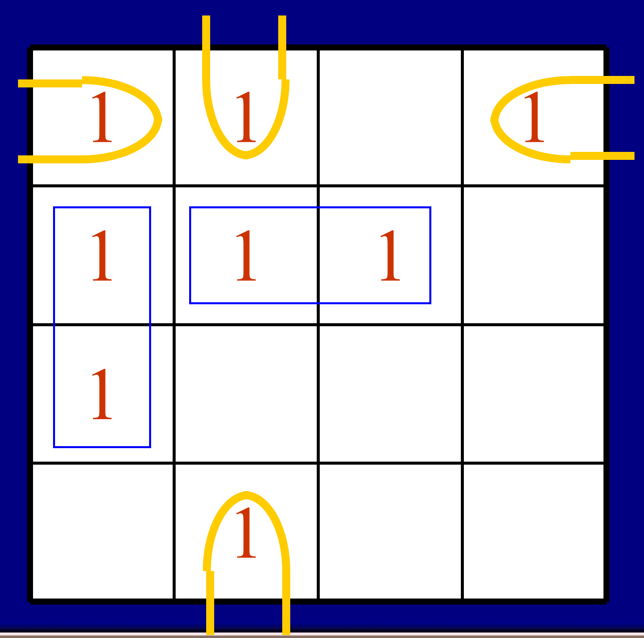
③实质最小项：圈 → 非画不可(不用说

只被一个质蕴涵所覆盖的最小项称为实质最小项。

④必要质蕴函(*Essential prime implicant*)：包含实质最小项的质蕴涵即为必要质蕴涵。

最大圈





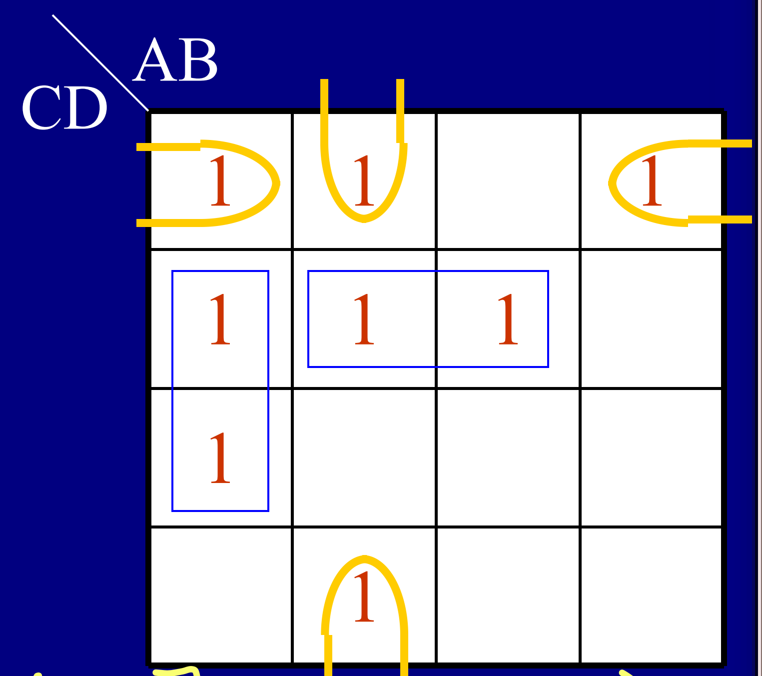
数字逻辑电路

⑤卡诺图上的最小覆盖：

用卡诺圈化简是有其特点和规律的：

挑选**数目最少**的必要质蕴涵，它们**覆盖了**图上**全部标1的小方格**，这就是最小覆盖。

最小覆盖所对应的逻辑表达式**就是最简的表达式**。



如图：函数的最简与或式

如遇大圈，必有单项式 那就求化简

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

四、用卡诺图化简逻辑函数

1. 用卡诺图法化简逻辑函数的基本步骤

(1) 将逻辑函数表示在卡诺图上；

(2) 根据实质最小项确定所有的必要极大圈；

(3) 如果所选出的所有必要极大圈已覆盖卡诺图上全部标1小方格，那么这些必要极大圈的集合就是卡诺图上的最小覆盖；

(4) 如果还有标1的小方格未被上述的必要极大圈覆盖，那么再加上选择最少的极大圈覆盖剩余的标1小方格，即获得最小覆盖；

(5) 写出最小覆盖所对应的逻辑表达式，即最简与或式。

- 总结卡诺图 化简 逻辑函数
  - 一至六变量的卡诺图 ——熟记
  - 卡诺图的构成特点
    - 每个小方格代表一个最小项
    - 相邻小方格只有一个变量不同
    - 卡诺图 上与、或、非等逻辑运算
  - 用卡诺图 表示逻辑函数
    - 真值表填入
    - 与或式填入
    - 或与式填入
    - 特殊函数填入
  - 卡诺图 化简 逻辑函数的方法步骤
    - 找相邻小方格
    - 画卡诺圈 合并相邻小方格——原则是先挑必要质蕴涵

- 注意几个名词
  - **蕴函项**——与项
  - **质蕴函**——最简与项
  - **必要质蕴函**——包含了实质最小项的最简与项（质蕴函）
  - **实质最小项**——只能画到一个卡诺圈中的最小项
  - **最小覆盖**——卡诺圈数目最少，与项数目最少

2. 将逻辑函数化简成最简与或表达式

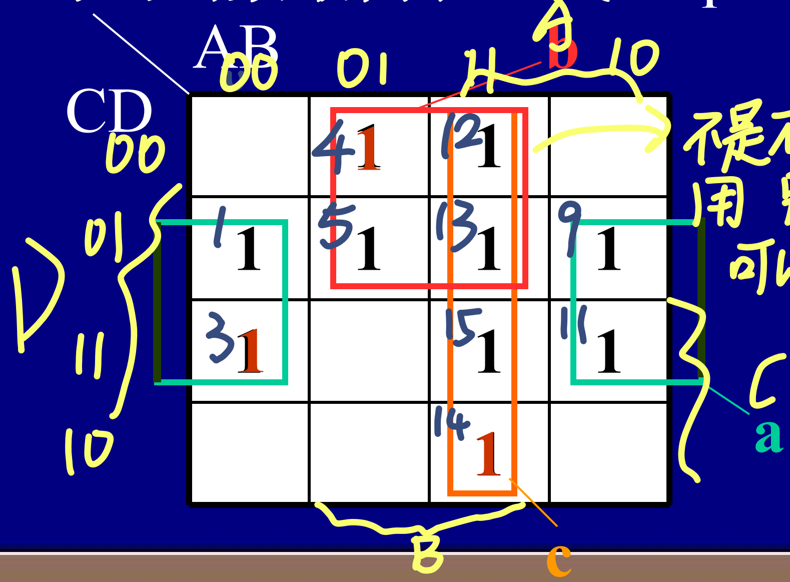
例1 化简 $F_1=\Sigma m^4(1,3,4,5,9,11,12,13,14,15)$

第一步：将函数F_1表示在卡诺图中；

第二步：选择出必要极大圈，它们是 a、b、c，确定所包含的实质最小项分别是 m_3、m_4、m_{14}；

第三步：确定 a、b、c 这三个必要极大圈已覆盖全部标1小方格；

第四步：写出函数最简表达式 $F_1= a+b+c = \overline{B}D+B\overline{C}+AB$



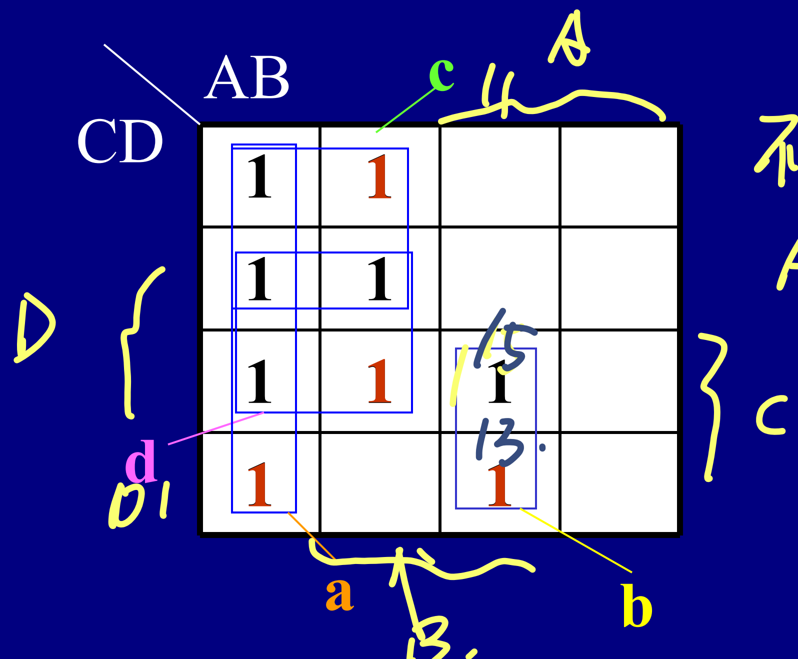
不是不可以用 是没画完 可以继续 但不要落单

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

ABC 不够 往大画.

数字逻辑电路

例2 化简$F_2=\Sigma m^4(0,1,2,3,4,5,7,14,15)$



不能6个. 只能8个.

$ABC+\bar{A}\bar{B}+\bar{A}\bar{C}+\bar{A}D.$

画完了.

期待这个出现

函数最简表达式

$$F_2= a + c + d + b = \bar{A}\bar{B} + \bar{A}\bar{C} + \bar{A}D + ABC$$

最简与或式

例3 化简$F_3=\Sigma m^4(1,5,7,9,11,15)$ 5变量很少见

AB / CD

1	1		1
	1	1	1

AB / CD

1	1		1
	1	1	1

① $F_3=\bar{B}\bar{C}D+\bar{A}BD+ACD$

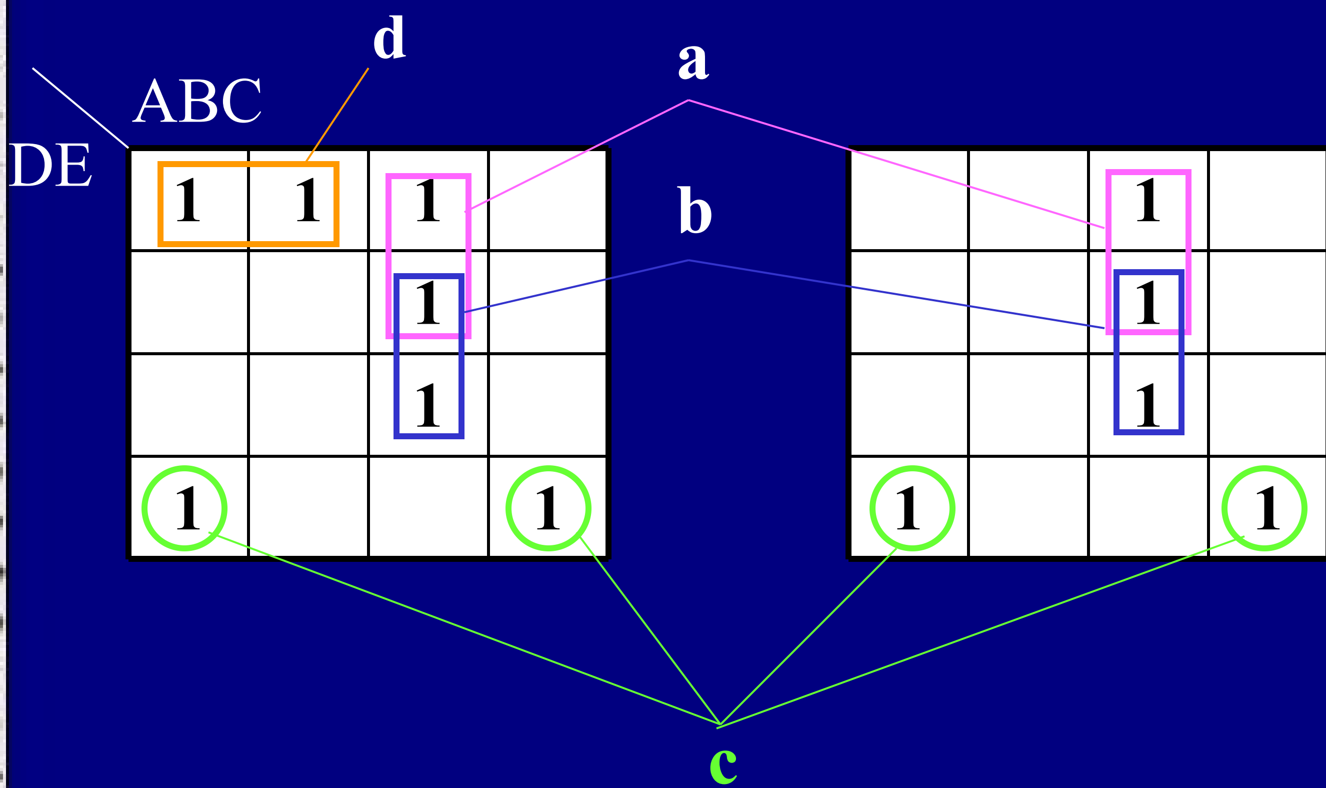
函数F_3的两种表达式，如图①和②所示。

AB / CD

1	1		1
	1	1	1

② $F_3=\bar{A}\bar{C}D+BCD+A\bar{B}D$

例4 化简$F_4=\Sigma m^5(0,2,4,10,12,13,15,18,26,28,29,31)$



函数的最简与或式 $F_4 = a + b + c + d$

$= BC\bar{D} + BCE + \bar{C}D\bar{E} + \bar{A}\bar{B}\bar{D}\bar{E}$

3. 将逻辑函数化简成最简或与式 比较方便电路设计

- 从代数法或与式的化简中已得知，如果**求出反函数的最简与或式**，则按反演规则可得到**原函数的最简或与式**。
- 原函数在卡诺图上**标0小方格**的集合正好是**反函数在卡诺图上的表示**，故：

(1) 按原函数在卡诺图中标0小方格的相邻情况，即可求出反函数的最简与或式；

(2) 将反函数求反，则得到原函数的最简或与式。

例1 化简 $F_1=\Sigma m^4(0,8,9,10,11,12,13,14,15)$

第一步：将F_1表示在卡诺图上，即标1小方格；

第二步：将未填1的小方格均填上0；

第三步：对所有标0小方格选出必要极大圈；

第四步：对所有标0小方格选择最小覆盖，即得到反函数

的最简与或式 $\overline{F_1}= a+b +c = \overline{A}B + \overline{A}D + \overline{A}C$

第五步：对反函数求反，即得到原函数的最简或与式。

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

CD \ AB	00 (+0)	10 (+8)	11 (+12)	01 (+4)	
00	0 1	0	12 1	4 1	+0
10	0	0	14 1	6 1	+2
11	0	0	15 1	7 1	+3
01	0	0	13 1	5 1	+1

b

c

a

先计算或后计算与.

不可以直接反演 要过"0"后再作一次反演

4. 同一函数的最简与或式 和 最简或与式 对应的电路比较

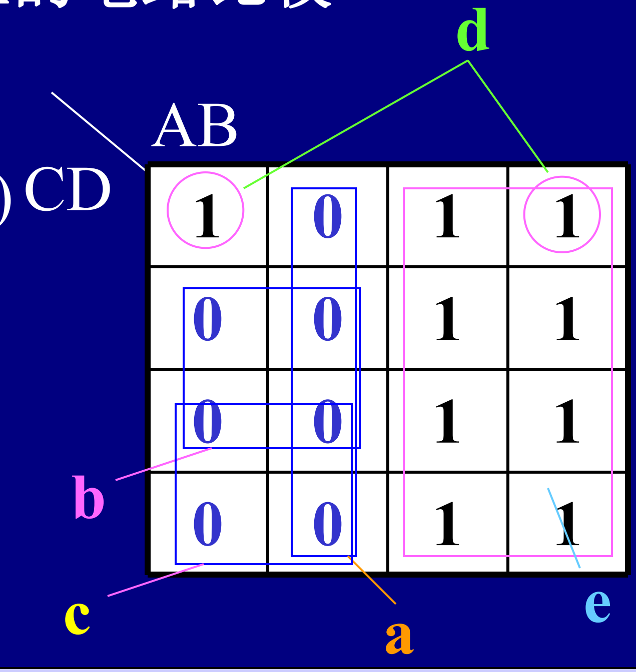
例： F_1的卡诺图表示如右：

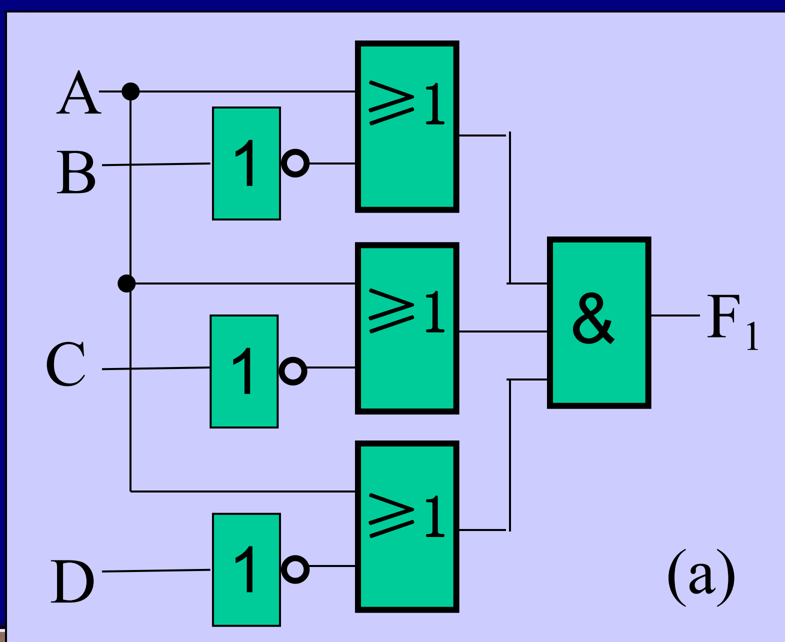
最简或与式：$F_1=(A+\overline{B})(A+\overline{D})(A+\overline{C})$

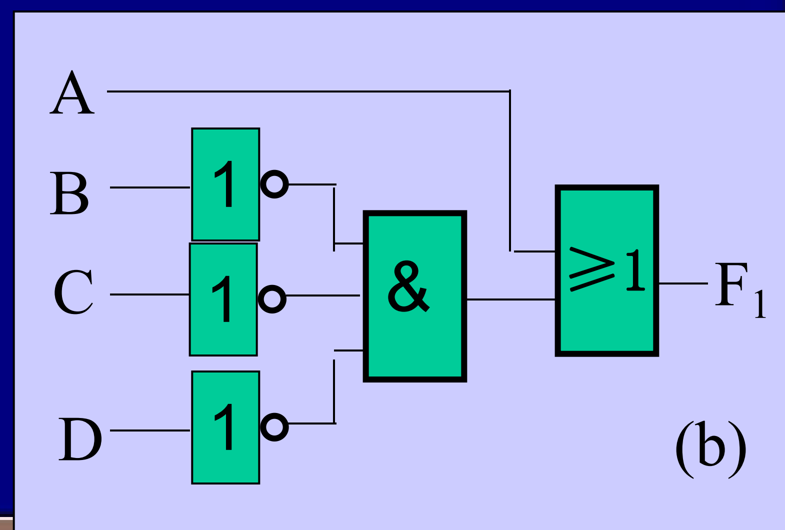
对应的电路图（a）所示；

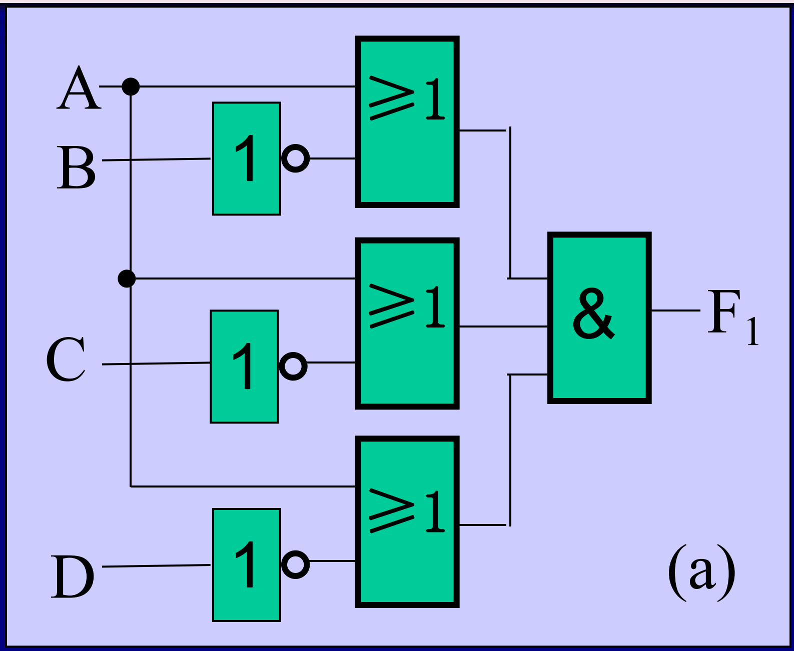
最简与或式：$F_1= A+\overline{B}\,\overline{C}\,\overline{D}$

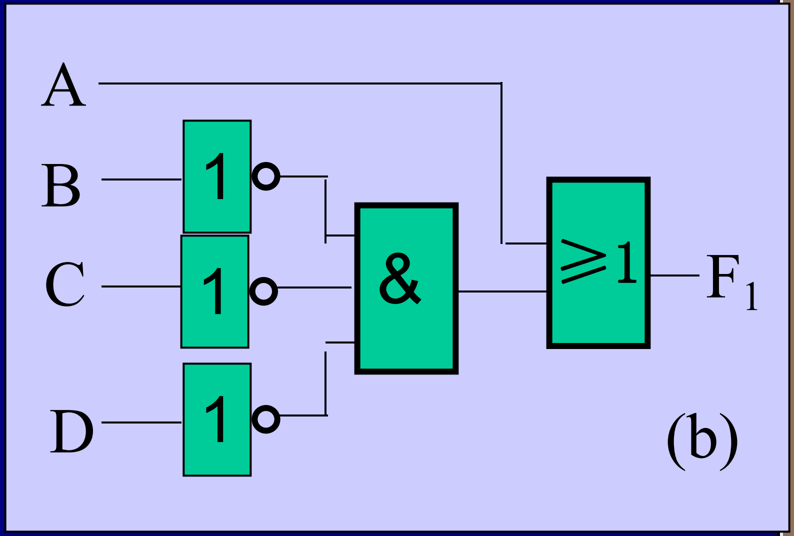
对应的电路图（b）所示。





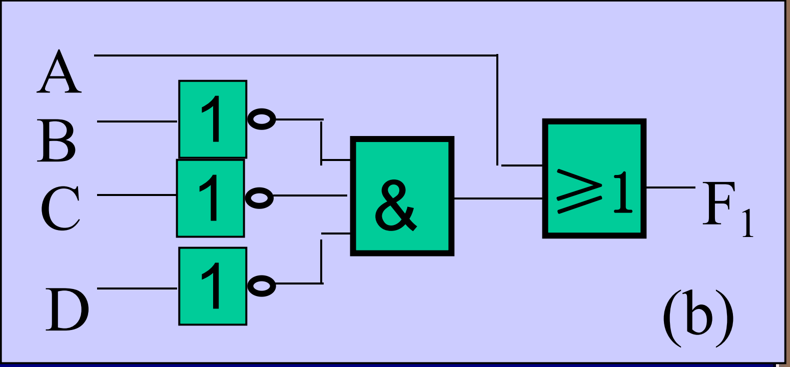






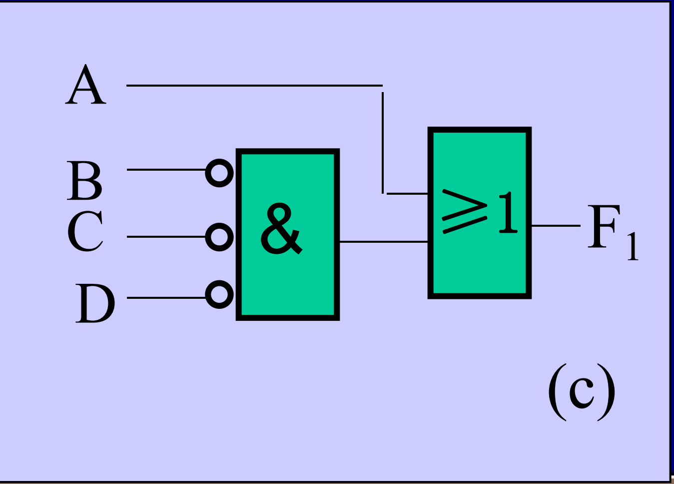
比较上述两种电路

- 图(b)比图(a)少用两个门，如果电路可任选，则应优先选用图(b)电路；
- 这两种电路形式上看都是三级；

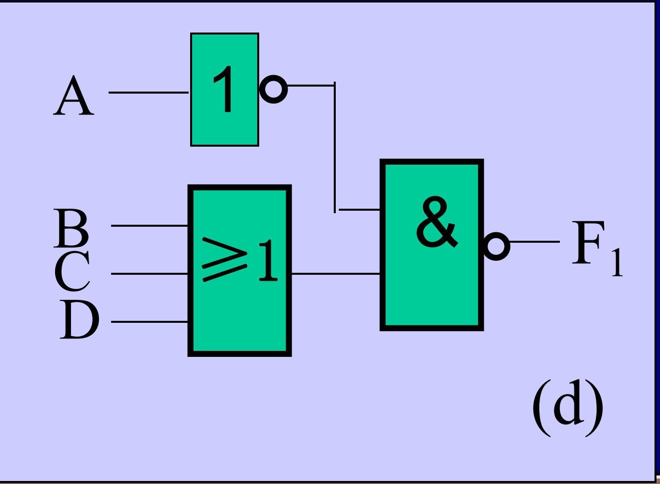


(b)

- 如果能用输入端为非的与门，则图(b)可转换成二级电路，　$F_1 = A + \bar{B}\bar{C}\bar{D}$，如图(c)所示；
- 如果没有输入端为非的与门，可将F_1的与或式变换成或与式，　$F_1 = \overline{\bar{A} \bullet (B+C+D)}$，电路如图(d)所示。



(c)



(d)

- 卡诺图 化简 逻辑函数
  - 化简 成最简 与或式
  - 化简 成最简 或与式

作业

- 关于卡诺图 化简的两个补充问题
  - 有无关项输入时
  - 输入无反变量时

1.3.5.3 利用无关项输入(*don't care input*)简化函数表达式

例如：一位BCD码输入的求偶电路。

分析：① 当输入为偶数时，输出 F 为1，否则输出 F 为0。

② 假设其输入为 $A_8 A_4 A_2 A_1$，在正常情况下，输出表达式可以写为： $F(A_8 A_4 A_2 A_1) = \sum m(0,2,4,6,8)$

③ 函数的最简与或式 $F = \overline{A}\,\overline{D} + \overline{B}\,\overline{C}\,\overline{D}$

A_2A_1 \ A_8A_4	+0	+4 01	+12 11	+8 10
+0	0 1	4 1	12 d	8 1
+1			13 d	
+3			15 d	11 d
+2	2 1	6 1	14 d	10 d

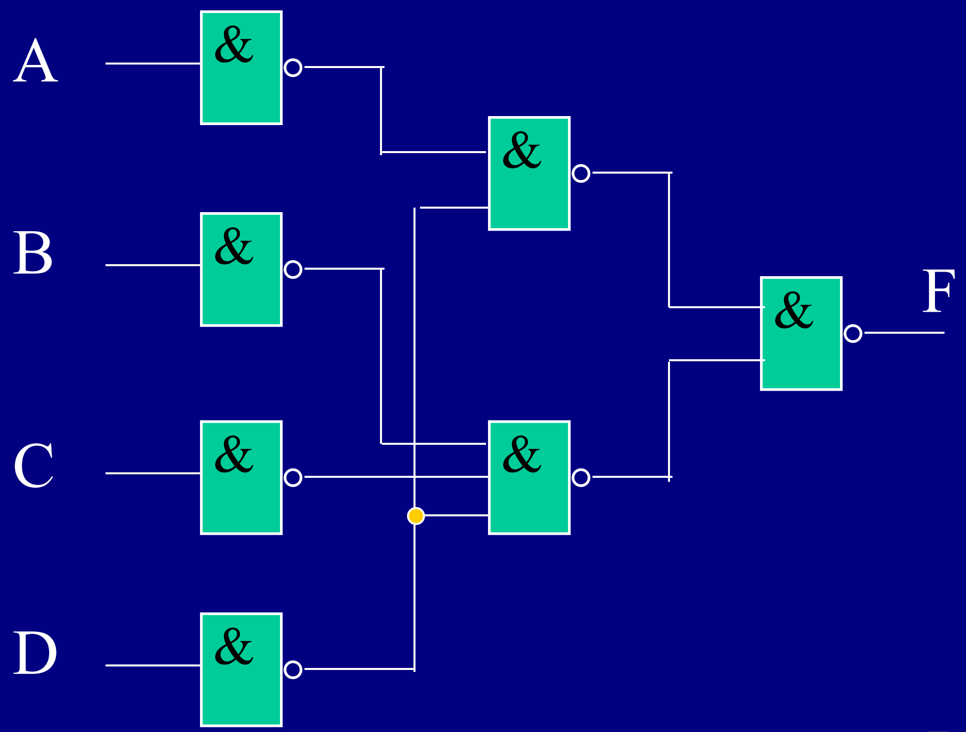
由于最小项m_{10}～m_{15}永远也不会出现，用"d"表示输入组合的无关项，填入卡诺图，表示此类小方格既可表示 1 也可表示 0。

则最简与或式 $F = \overline{D}$ 。

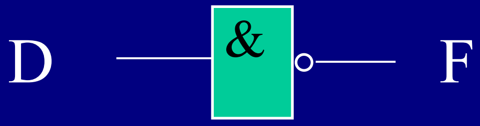
无所谓
你存在与否

无关最小项的未使用和使用的电路比较:

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



$$F=\overline{D}$$



数字逻辑电路

无关项(*don't care terms*) 也称任意项、约束项，构成的函数为不完全给定函数(*Incompletely Specified functions*)

A 后期(带无关项)

定义：当函数输出与某些输入组合无关时，这些输入的组合称为无关项。

产生原因：

①这些输入组合在正常操作中不会出现（即输入具有约束条件）；

②即使这些输入可能出现（即输入不具有约束条件），但实际上输出与它们无关。

作用：当输入出现这些无关组合 **d** 时， **d** 可以随意加入或不加入其对应的函数 F 中（既可使 F 为 1，也可使确F 为 0），并不影响 F 原有的逻辑功能，但为函数 F 的化简提供了帮助。

数字逻辑电路

例1：一个BCD码输入质数检测器。

假设输入为$N_3N_2N_1N_0$，输出表达式可以写为：

$F = \sum m^4(1,2,3,5,7) + \sum d^4(10,11,12,13,14,15)$ 补齐

这里d（...)项列出的即为无关项。

化简的第一步：给出函数的初始卡诺图；

第二步：按前述的方法找出必要质蕴涵，区别是：

- 画覆盖标 1 小方格的极大圈时，应把相邻的 “d” 包含在内（相当于使 d＝1），使画出的极大圈尽可能地大，可减少该质蕴涵的变量数；
- 不圈任何仅包含 d 的圈；
- 不圈任何标 0 的小方格；

N_3N_2 / N_1N_0

		d	
1	1	d	
1	1	d	d
1		d	d

BCD码质数检测器的最简与或式

$$F = \overline{N_2}N_1 + \overline{N_3}N_0$$

数字逻辑电路

例2： $F = \Sigma m^4(4,5,13,15) + d^4(2,3,7,9,14)$

最简与或式

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

CD \ AB				
		1		
		1	1	d
	d	d	1	
	d		d	

例3：一个2421码输入四舍五入判别电路。

$F = \sum m^4(11,12,13,14,15)$

$+\sum d^4(5,6,7,8,9,10)$

最简与或式 $F = A$

CD \ AB				
			1	8 d
		d	1	10 d
		d	1	1
		d	1	9 d

例4：一个余3码输入的偶数电路。

$$F = \sum m^4 (3,5,7,9,11) + \sum d^4 (0,1,2,13,14,15)$$

AB

CD

d			
d	1	d	1
1	1	d	1
d		d	

最简与或式　**F = D**

例5：求如下十进制的Gray码的质数检测电路

CD \ AB				
	0	1	8	9
		2	7	
		3	6	
		4	5	

CD \ AB				
		1		
	d	1	1	d
	d	1		d
	d		1	d

最简与或式 $F = AC\overline{D} + \overline{C}D + \overline{A}B\overline{C} + \overline{A}D$

- 充分利用无关项d帮助化简函数

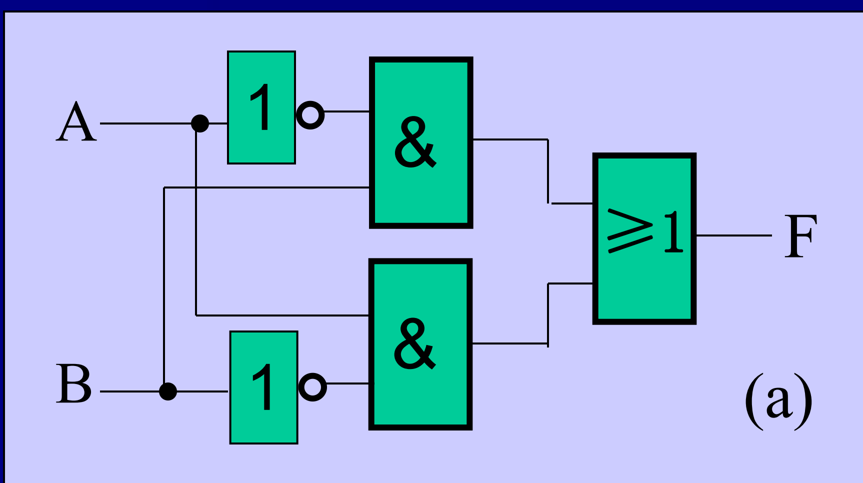
作业

1.3.5.4 输入无反变量的函数的化简

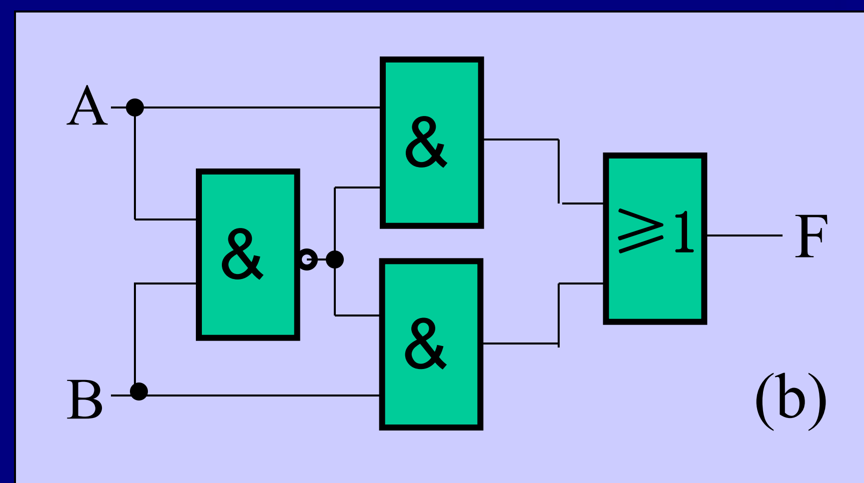
在电路中为减少连线数目，对其外部输入变量只有原变量没有反变量，电路要通过非门来实现反变量。

例： $F = A \oplus B = A\overline{B} + \overline{A}B = A\,\overline{AB} + B\,\overline{AB}$

下图是两式对应的电路，后者可减少一个非门。



(a) F= A⊕B 不共享非门



(b) F= A⊕B 共享非门

对于与或式，共享的门是与非门。

当反变量较多时，如果每个反变量都加个非门太不经济，考虑能否共享非门，即寻找把多个单输入非门合并成一个多输入与非门的方法可以减少非门的个数。

- 化简的出发点是函数已经为最简与或式；
- 化简的主要方法有以下两种：

1. 替代尾因子法

定义：每个与项中原变量部分称为头因子，反变量部分称为尾因子。

特点：把头因子中的任何变量放入任一个尾因子中，该与项不变，即头因子是不变的，尾因子是可变的 。

例：用摩根定律证明 $F = AB\bar{C}\,\overline{DE} = AB\,\overline{AC}\,\overline{DE} = AB\,\overline{BC}\,\overline{DE}$

$= AB\,\overline{ABC}\,\overline{DE} = AB\,\overline{ABC}\,\overline{ADE} = AB\,\overline{ABC}\,\overline{ABDE}$

化简步骤为：

① 把最简式中具有相同头因子的与项合并成一个与项。

$$F = A\bar{B}\bar{C} + BC\bar{D} + A\bar{C}\bar{D} + \bar{B}CD + \bar{A}BC + \bar{A}CD$$ 11个门

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

$$= A\bar{C}\,\overline{BD} + BC\,\overline{AD} + CD\,\overline{AB}$$ 8个门

② 列出最简与或式中所有与项的头因子、尾因子及替代尾因子。

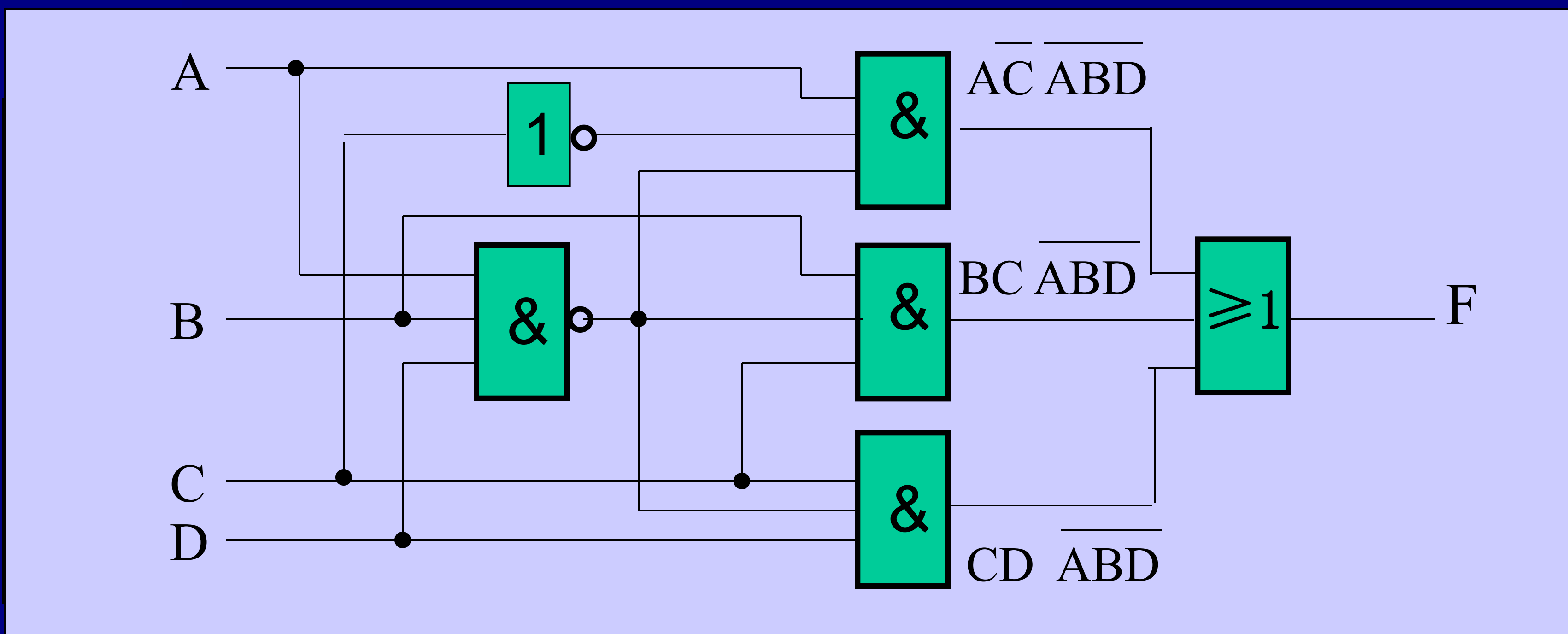
与项	头因子	尾因子	替代尾因子
$A\bar{C}\,\overline{BD}$	A	$\bar{C}$	$\bar{C}$，$\overline{AC}$
		$\overline{BD}$	$\overline{BD}$，$\overline{ABD}$
$BC\,\overline{AD}$	BC	$\overline{AD}$	$\overline{AD}$，$\overline{ABD}$，$\overline{ACD}$，$\overline{ABCD}$
$CD\,\overline{AB}$	CD	$\overline{AB}$	$\overline{AB}$，$\overline{ABC}$，$\overline{ABD}$，$\overline{ABCD}$

③选择共享的替代尾因子，选择的原则如下：

- 替代尾因子共享数尽可能多。
- 在共享数相等时选择最简单的一个。

因此： $F = A\overline{C}\,\overline{ABD} + BC\,\overline{ABD} + CD\,\overline{ABD}$

该电路图如下。



2. 禁止逻辑法

- 任何函数利用不属于它的最小项之非乘之，其逻辑功能不变。即： $F = F \cdot \overline{m_i}$ （m_i不在F中）

  ∵ 不属于F的最小项m_i 均为0

  ∴ $\overline{m_i} = 1$ 故上式成立。

进一步推广 $F = F \cdot \overline{(m_i + m_j)}$ （m_i、m_j均不在F中）

- 任何函数如用属于它的最小项之和的非乘之，则相当于从该函数中扣去了这几个最小项，称禁止逻辑。

例： $F = m_1 + m_3 + m_5 + m_7$

$G = F \cdot (\overline{m_5 + m_7})$

证明

$= (m_1 + m_3 + m_5 + m_7) \cdot (\overline{m_5 + m_7})$

$= (m_1 + m_3 + m_5 + m_7) \cdot \overline{m_5}\,\overline{m_7}$

$= m_1 + m_3$

禁止逻辑（又称阻塞逻辑），上式中$m_5 + m_7$称为

禁止项，G 是 F 被 $\overline{m_5 + m_7}$ 禁止后的函数。

禁止逻辑的几何意义：

$$F = m_1 + m_3 + m_5 + m_7$$

$$G = F \bullet (\overline{m_5 + m_7})$$

$$= m_1 + m_3$$

	00	01	11	10
0				
1	1 1	3 1	7	5

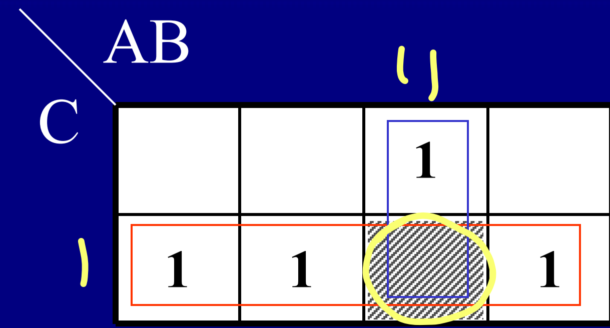
禁止逻辑法化简函数的步骤：

① 把函数中各与项或部分与项共享的标 0 单元画入极大圈；

② 把这些 0 单元作为禁止项，从极大圈禁止掉，原函数保持不变。

数字逻辑电路

例 $F_1 = m_1 + m_3 + m_5 + m_6$



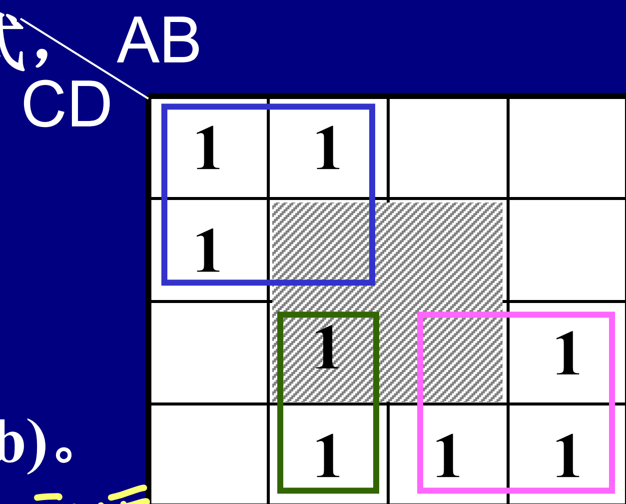
第一步：找出共享的 0 单元m_7作为禁止项；

第二步：画出相应的极大圈；

第三步：函数的与或式 $F_1 = C\,\overline{ABC} + AB\,\overline{ABC}$

例 $F_2 = A\bar{B}C + BC\bar{D} + \bar{A}BC + \bar{A}\bar{B}\bar{C} + \bar{A}\bar{C}\bar{D}$

从卡诺图可知，是最简与或式，电路实现需要**10个与非门**。



如图找出禁止项，画出极大圈，

则 $F_2 = \bar{A}\bar{C}\,\overline{BD} + AC\,\overline{BD} + \bar{A}BC$

实现需要**7个与非门**(书P60图2-37b)。

大框规整也方便，一起挖才有定

数字逻辑电路

3. 输入无反变量的或与式函数的化简

不要求

根据对偶规则：

F —— 给出或与式函数

F'—— 得到与或式函数，寻求共享的与非项

F = (F') '——得到或与式原函数，含有共享的或非项。

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

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

$= AB\,\overline{ABC} + BC\,\overline{ABC} + AC\,\overline{ABC}$

$F = (F')'$

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

AB / C

		1	
	1		1

电路由五个或非门实现(书P61图2-38)。

- 输入无反变量函数的化简
  - 替代尾因子法——代数化简法
  - 禁止逻辑法——卡诺图法

    互通.
    - 共享与非门，禁止项应该尽可能画大
    - 禁止项应选在变量的高区

作业

数字逻辑电路

- 逻辑代数， 布尔代数， 开关代数
- **基本概念**： 正逻辑约定

  逻辑函数的4种表示方法
- **基本运算**： 与， 或， 非， 特别留意逻辑符号
- **基本公理**： 注意运算次序（ 括号、 非、 与、 或），

  或对与的分配律$A+B\cdot C=(A+B)(A+C)$
- **基本定理**： 吸收律、 反演律、 包含律

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

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

  $AB+A\overline{C}+BC=AB+A\overline{C}$
- **基本规则**： 代入规则、 反演规则、 对偶规则

- **总结逻辑函数的性质**
- 复合逻辑——逻辑完备性

  与非门的速度最快，代价最低，最常用

  器件的符号

  器件的0、1控制作用

  异或、同或的特点
- 逻辑函数的基本表达式——不唯一

  与或式- 与非式

  简单应用，与或式需要多少个与非门实现
- 逻辑函数的标准形式——最小项之和，最大项之积

  最小项、最大项对下标的规定

  最小项、最大项的性质

  如何写最小项、最大项标准式（从与或式、真值表、卡诺图）

  $F=\sum m(\ )=\prod M(\ )$

  $\overline{F}=\sum m(\ )=\prod M(\ )$

- 总结卡诺图化简逻辑函数
  - 一至六变量的卡诺图——熟记
  - 卡诺图的构成特点
    - 每个小方格代表一个最小项
    - 相邻小方格只有一个变量不同
    - 卡诺图上与、或、非等逻辑运算
  - 用卡诺图表示逻辑函数
    - 真值表填入
    - 与或式填入
    - 或与式填入
    - 特殊函数填入
  - 卡诺图化简逻辑函数的方法步骤
    - 找相邻小方格
    - 画卡诺圈合并相邻小方格——原则是先挑必要质蕴函

- 卡诺图 化简 逻辑函数
  - 化简 成最简 与或式
  - 化简 成最简 或与式
  - 带无关项输入的化简
  - 输入无反变量的化简

数字逻辑电路

- 卡诺图 化简 逻辑函数
  - 化简 成最简与或式
  - 化简 成最简或与式
  - 带无关项输入的化简
  - 输入无反变量的化简