

A班（电阻电路+动态电路）获得一学期学分同学，只能在A班获得另一学期学分
目前A班有杨帆老师开设的平行班

本学期周三晚上习题课：线上

电子电路与系统基础II

理论课第1讲 晶体管数字门电路

李国林

清华大学电子工程系

课程安排

- 按既定计划推进
 - 教学进度计划见网络学堂

- 习题课
 - 每周周三晚上
 - 瞩目：课前**20**分钟微信群和雨课堂通知

晶体管数字门电路 大纲

- 基本逻辑
 - 与，或，非
 - 逻辑运算规则
- 门电路实现
 - 以非门为例，说明用晶体管开关实现门电路
 - BJT, MOSFET, CMOS
 - CMOS门电路
 - 非门、与非门，或非门
- 简单信号处理器
 - 数学运算
 - 加法器例
 - 编码
 - 奇偶校验例：检错
- 总结
 - 数字化趋势
 - 模拟电路的必要性

一、基本逻辑运算

- 逻辑
 - **logic**: 是关于争执 **argument** 的学问
 - 谁对谁错, 孰真孰假?
 - **true or false?**
- 二进制**01**自然对应逻辑**01**

二值逻辑	1	0
正反判断	true	false
	真	假
	正	反
	正确	错误
	同意	反对
	是	否
	许可	不可
	高电平	低电平

布尔逻辑

Boolean logic

真值表

A	not A
0	1
1	0

- George Boole 19世纪中叶定义

- 布尔逻辑中的三种基本运算

- 非: **NOT**

- 反着来, 对着干

- 与: **AND**

- 两个人都同意才可行

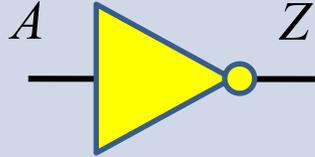
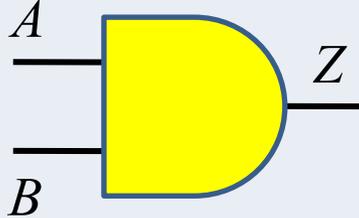
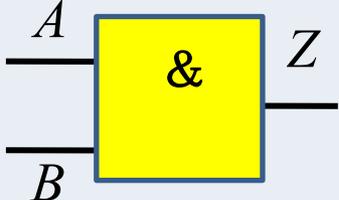
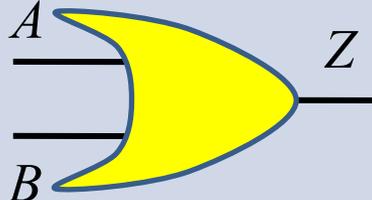
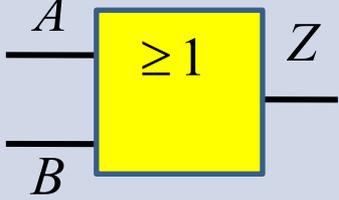
- 或: **OR**

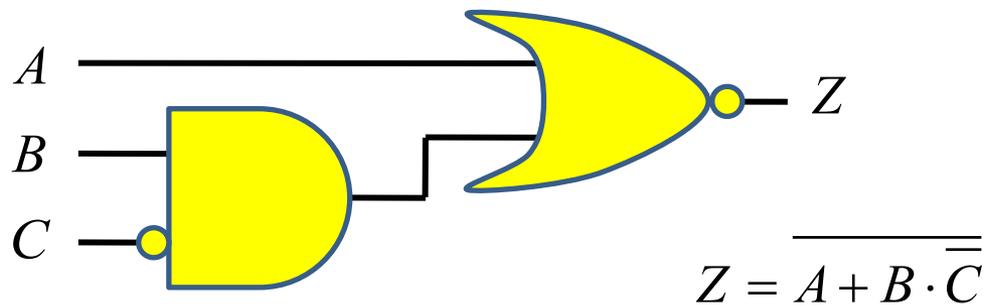
- 两个人中只要有一个同意就可行

A	B	A and B
0	0	0
0	1	0
1	0	0
1	1	1

A	B	A or B
0	0	0
0	1	1
1	0	1
1	1	1

逻辑运算电路符号

逻辑	逻辑表达式	常用电路符号	方块符号
非	$Z = \overline{A}$ $Z = \neg A$ $Z = \text{not } A$	非门 	NOT gate Inverter 
与	$Z = AB$ $Z = A \cdot B$ $Z = A \wedge B$ $Z = A \text{ and } B$	与门 	AND gate 
或	$Z = A + B$ $Z = A \vee B$ $Z = A \text{ or } B$	或门 	OR gate 



由逻辑表达式 获得真值表

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

真值表代表电路功能

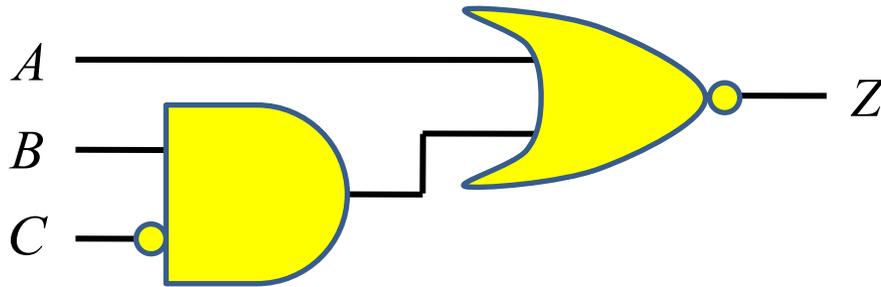
由真值表获得逻辑表达式

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

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

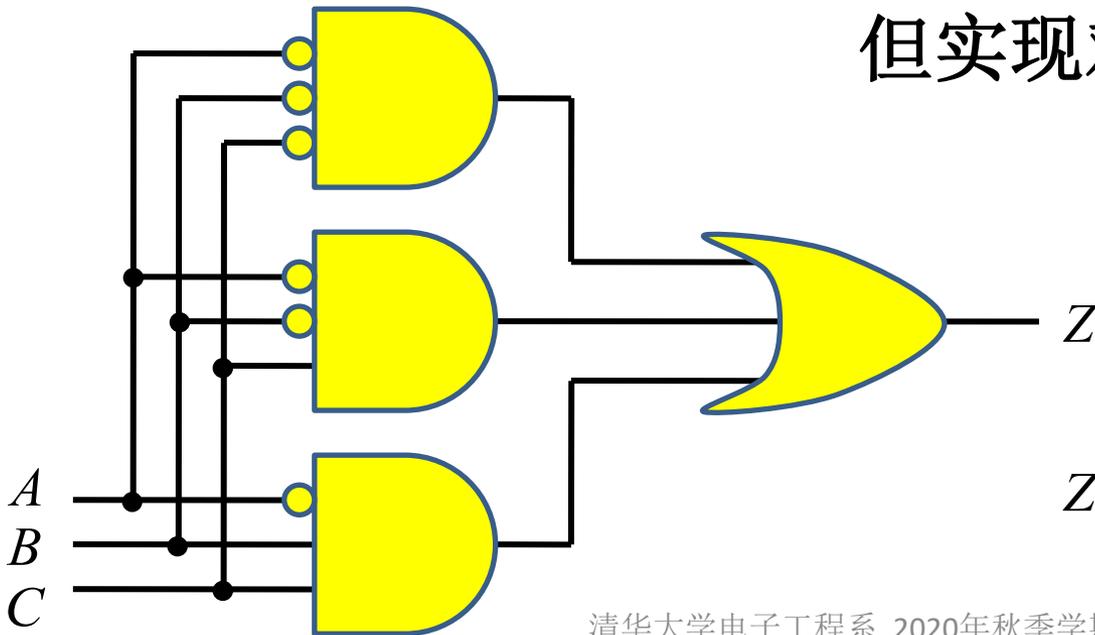
$$Z = \overline{A + B \cdot \overline{C}}$$

如何化简？



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

逻辑运算功能完全一致
但实现难度和成本差距很大



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

逻辑运算的基本规则

$$\overline{\overline{A}} = A \quad \text{双反律: double negation}$$

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

恒等律: identity

$$A + A = A \quad A \cdot A = A$$

幂等律: idempotence

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

湮灭律: annihilator

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

互补律: complementation

交换律: commutativity

$$AB = BA$$

$$A + B = B + A$$

$$(AB)C = A(BC)$$

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

结合律: associativity

分配律: distributivity

$$A(B + C) = AB + AC$$

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

吸收律: absorption

$$A + AB = A$$

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

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

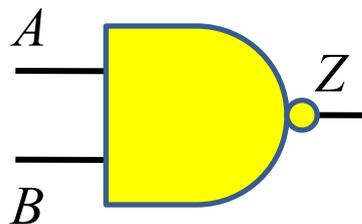
De Morgan's law

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

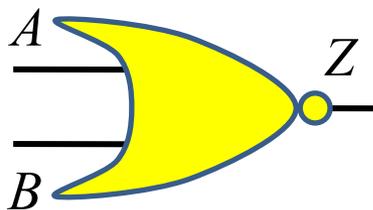
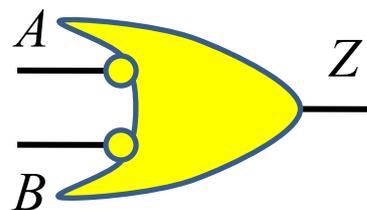
两个逻辑相同，只要它们的真值表相同

De Morgan律的电路符号表述

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

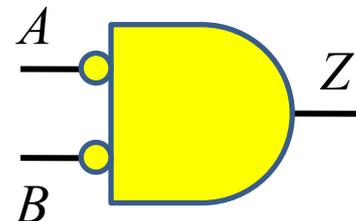


NAND gate
与非门



NOR gate
或非门

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



对De Morgan律的直观理解

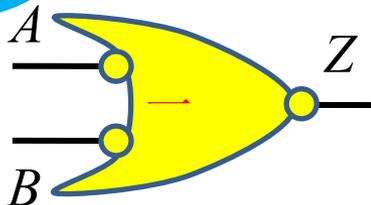
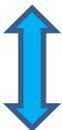
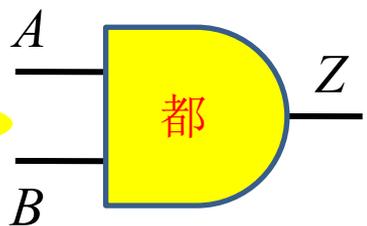
$$\overline{A \cdot B} = \overline{A} + \overline{B}$$

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

两个人都同意则同意

换句话说

两个人中，只要有一个人不同意，则不同意



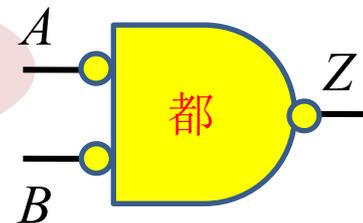
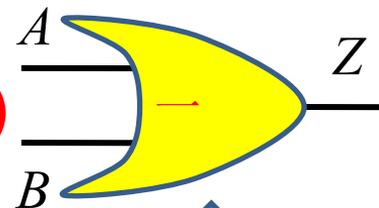
$$\overline{A + B} = \overline{A} \cdot \overline{B}$$

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

两个人中只要有一个同意了则同意

换句话说

两个人都不同意则不同意



逻辑表达式化简

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

?

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

$$\begin{aligned} Z &= \bar{A} \cdot \bar{B} \cdot \bar{C} + \bar{A} \cdot \bar{B} \cdot C + \bar{A} \cdot B \cdot C \\ &= \bar{A} \cdot \bar{B} \cdot \bar{C} + (\bar{A} \cdot \bar{B} \cdot C + \bar{A} \cdot \bar{B} \cdot C) + \bar{A} \cdot B \cdot C \\ &= (\bar{A} \cdot \bar{B} \cdot \bar{C} + \bar{A} \cdot \bar{B} \cdot C) + (\bar{A} \cdot \bar{B} \cdot C + \bar{A} \cdot B \cdot C) \\ &= \bar{A} \cdot \bar{B} \cdot (\bar{C} + C) + \bar{A} \cdot C \cdot (\bar{B} + B) \\ &= \bar{A} \cdot \bar{B} + \bar{A} \cdot C \\ &= \bar{A} \cdot (\bar{B} + C) \\ &= \overline{A + B \cdot C} \end{aligned}$$

幂等律

结合律

分配律

互补律, 恒等律

分配律

De Morgan律

是否是最简式?

不够直观? 最终结果不确定

卡诺图 Karnaugh Maps

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

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

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

卡诺图适用于**3**输入、**4**输入变量的逻辑运算

简单明了

多于**4**变量的逻辑运算化简方法，后续数字逻辑与处理器基础课程中学习

卡诺图训练

AB \ CD	00	01	11	10
00	1	0	0	0
01	1	1	*(不在意)	1
11	1	1	1	1
10	1	1	1	1

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

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

$$Z = \bar{D} + A \cdot C$$

二、门电路的实现

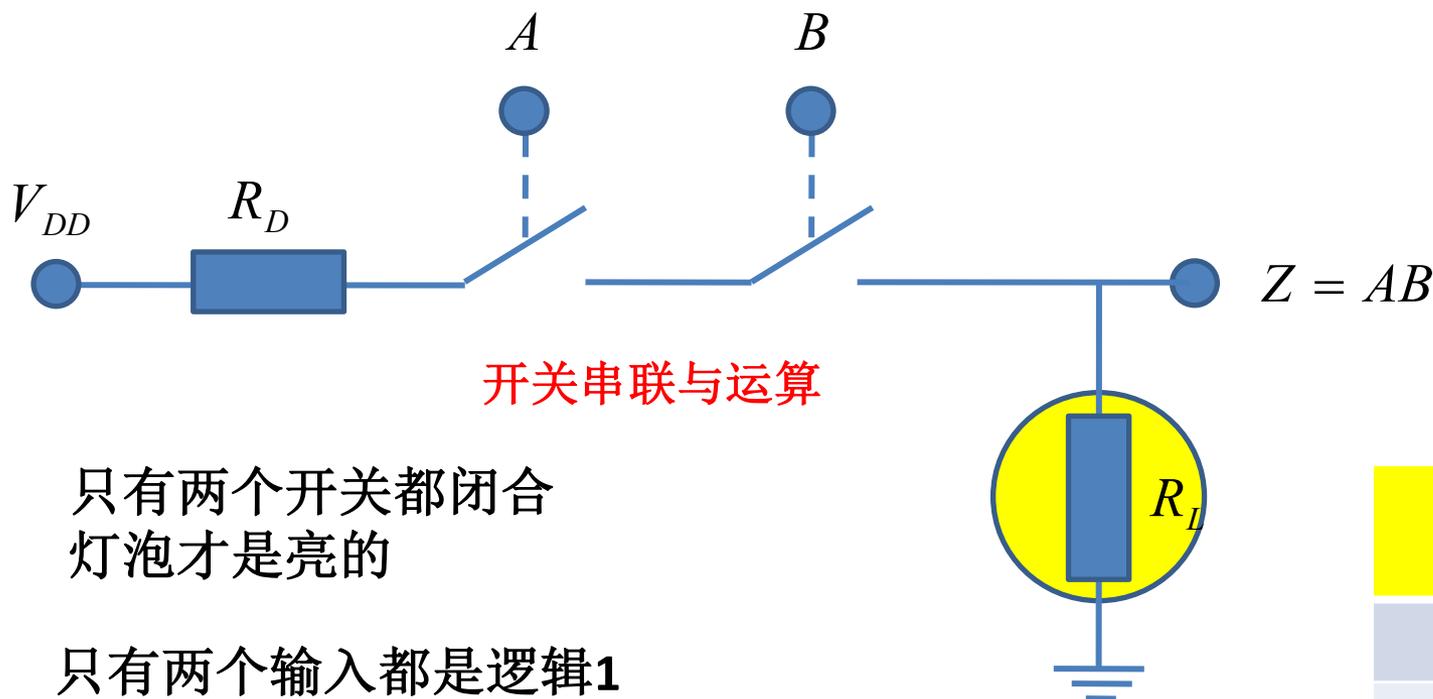
- 逻辑门电路如何实现呢？
 - 逻辑有0、1之分，对应开关的开、关两个状态，因此逻辑门电路可以用开关实现
 - 基本门电路的开关实现
 - 与，或，非
 - 与非，或非

与运算的开关实现方案

定义或实现:

逻辑0: 开关断开

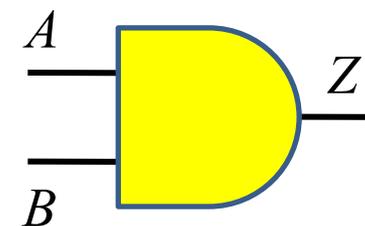
逻辑1: 开关闭合



只有两个开关都闭合
灯泡才是亮的

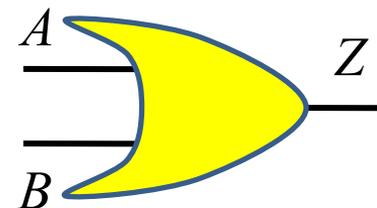
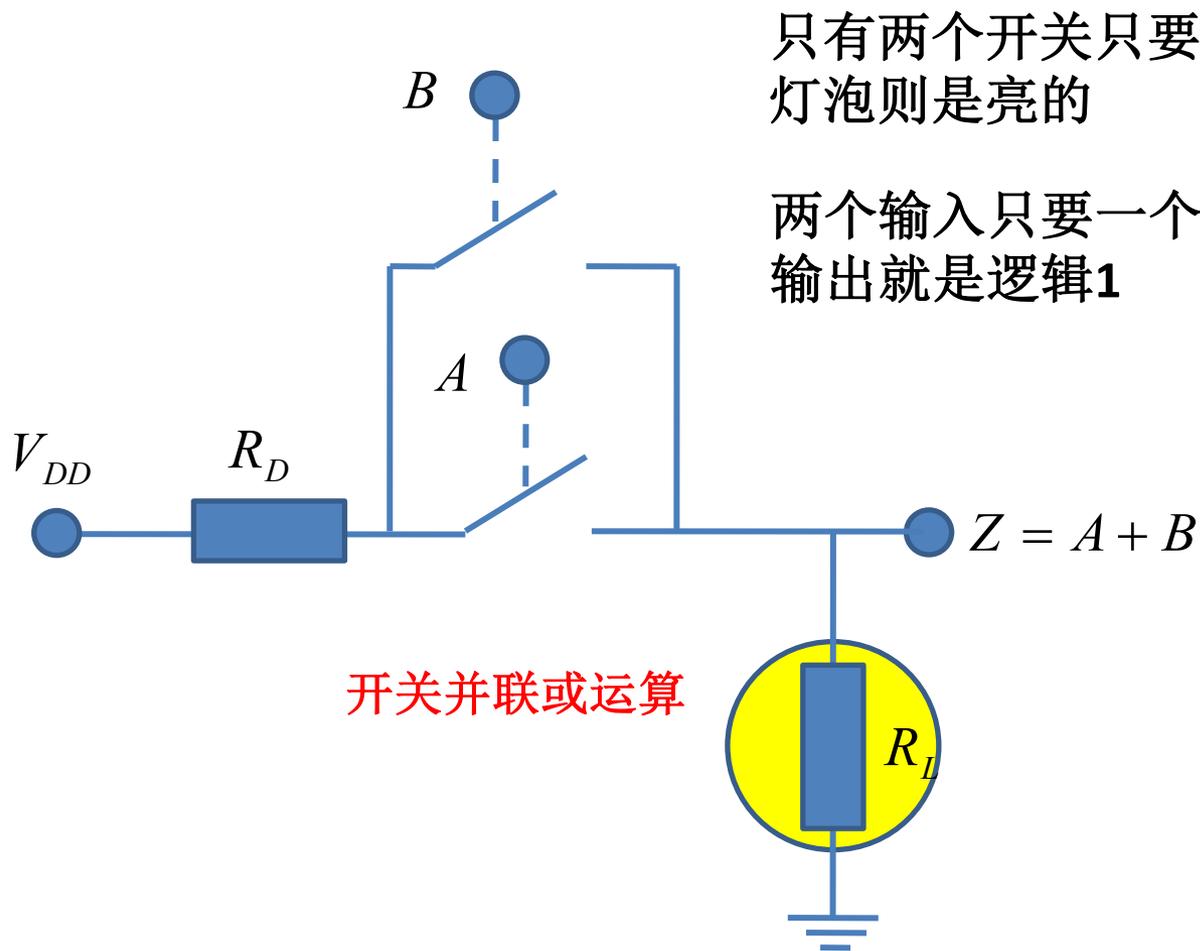
只有两个输入都是逻辑1
输出才是逻辑1

定义或实现:
灯泡亮: 逻辑1
灯泡灭: 逻辑0



A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1

或运算的开关实现方案

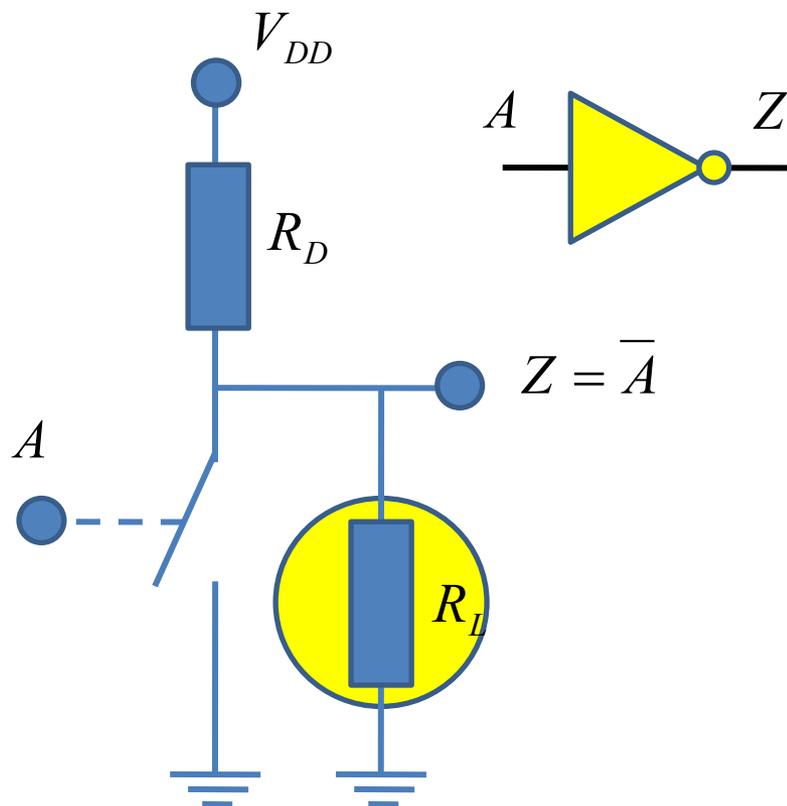


A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

非运算的旁路开关实现方案

Bypass Switch

输入逻辑0：开关断开



$$V_L = \frac{R_L}{R_L + R_D} V_{DD}$$

灯泡亮：输出逻辑1

输入逻辑1：开关闭合

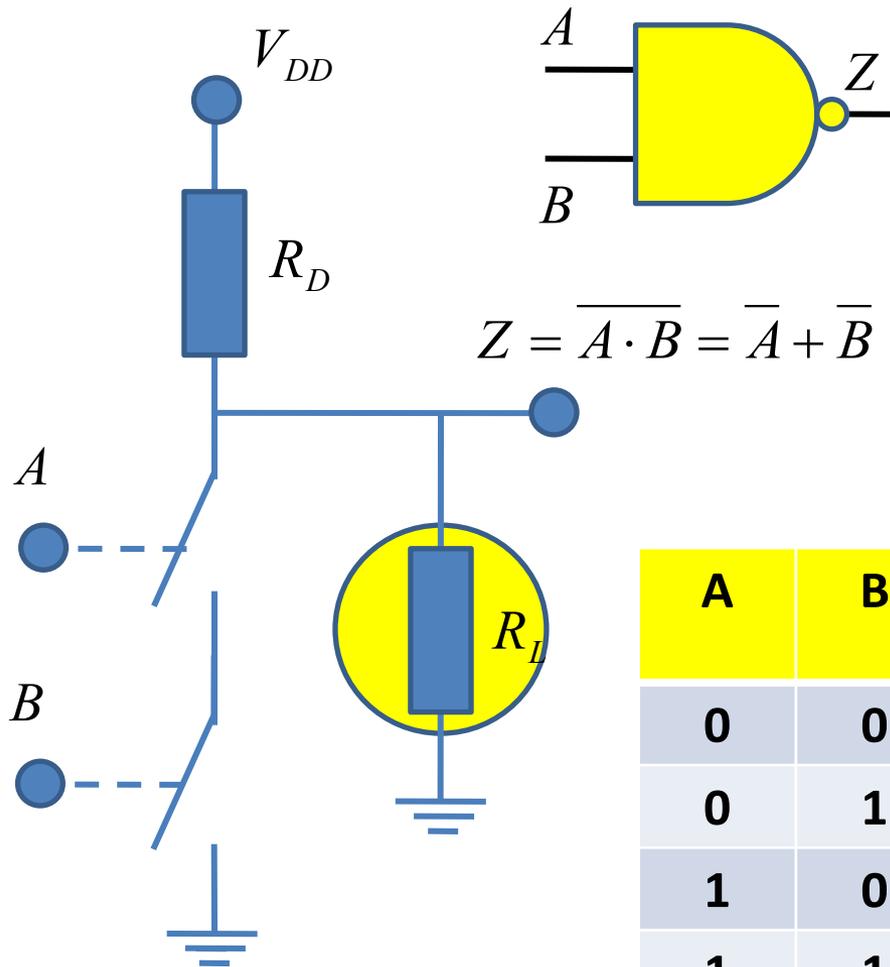
$$V_L = 0$$

灯泡灭：输出逻辑0

开关旁路非运算

A	not A
0	1
1	0

与非运算的旁路开关实现方案



只有两个开关都闭合
灯泡才是灭的

只有两个输入都是逻辑1
输出才是逻辑0

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

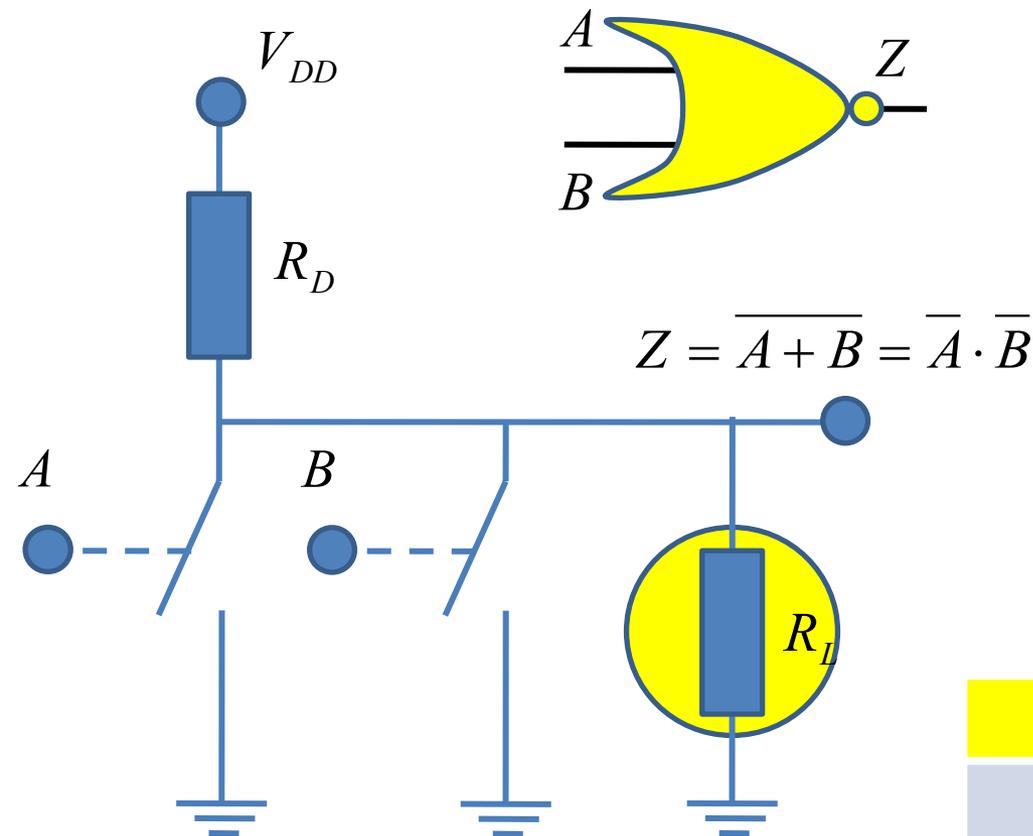
两个开关只要一个是断开的
灯泡则是亮的

两个输入只要一个是逻辑0
输出就是逻辑1

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

A	B	\overline{AB}
0	0	1
0	1	1
1	0	1
1	1	0

或非运算的 旁路开关实现方案



两个开关只要一个闭合
灯泡则是灭的

两个输入只要一个是逻辑1
输出就是逻辑0

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

只有两个开关都断开
灯泡才是亮的

只有两个输入都是逻辑0
输出才是逻辑1

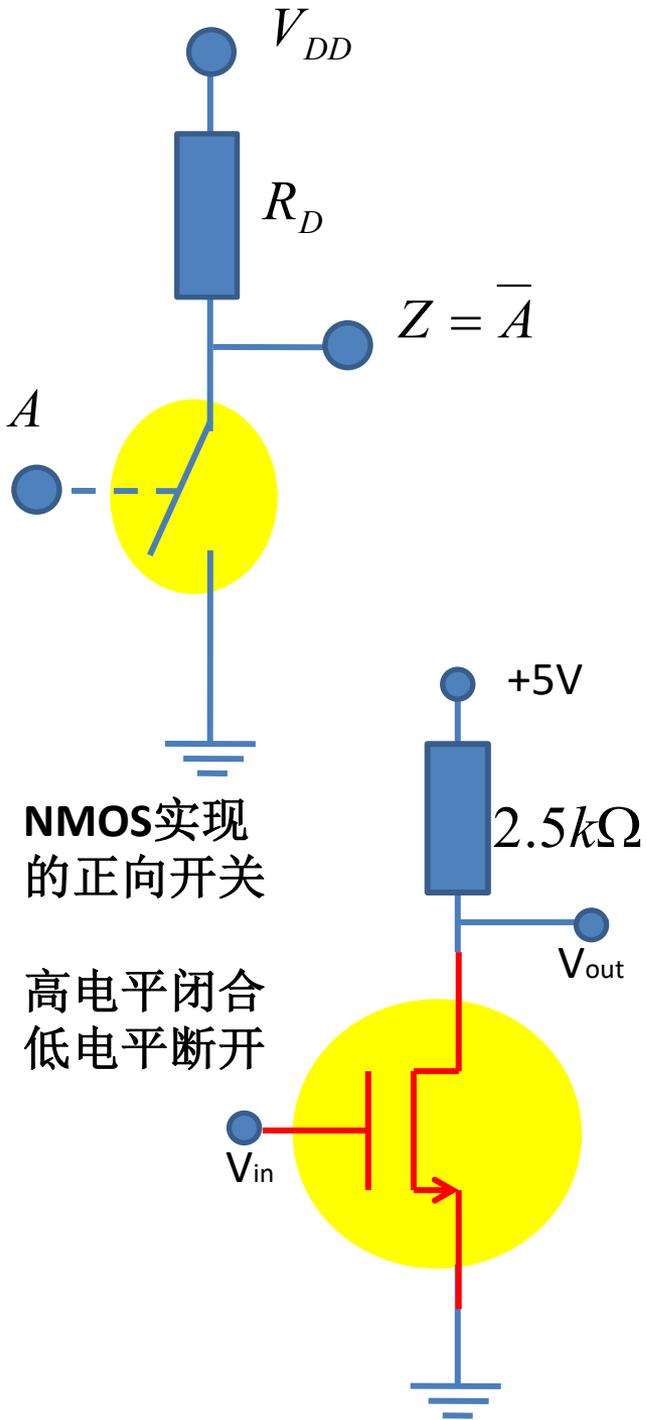
$$Z = \overline{A} \cdot \overline{B}$$

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

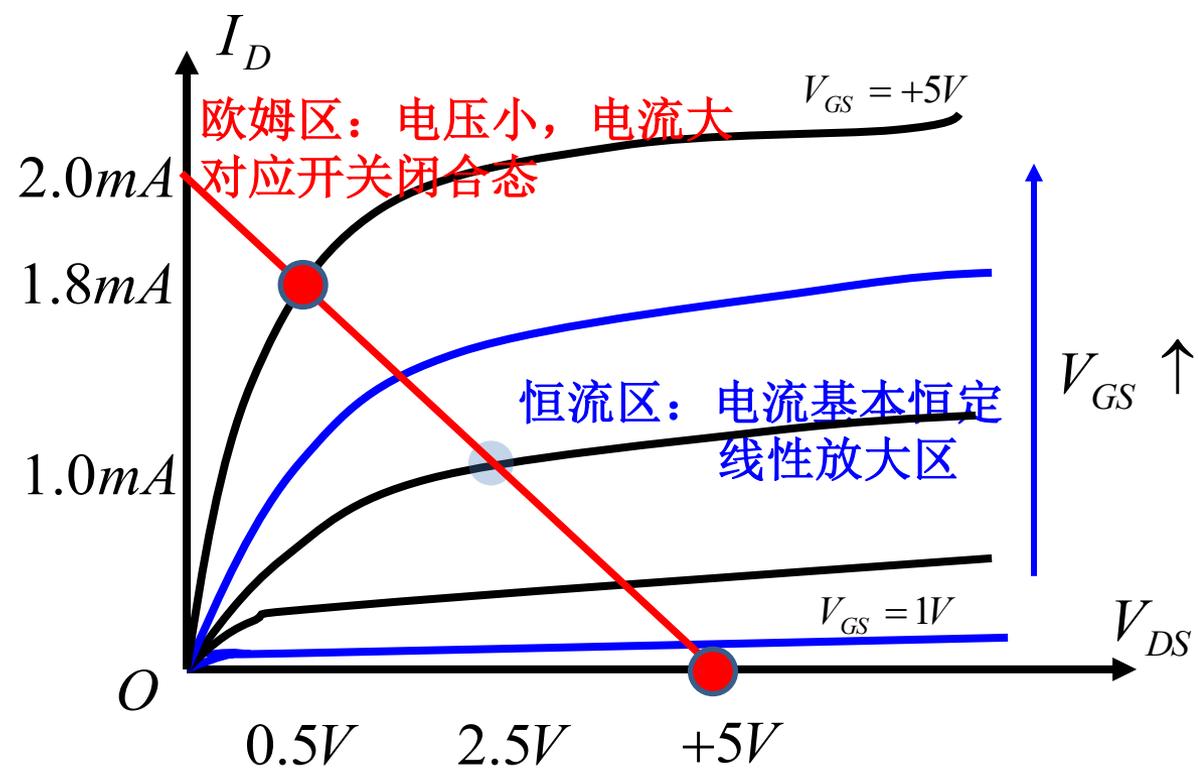
开关如何实现？

- 机械开关不适宜自动控制
- **BJT**有‘饱和导通态’和‘截止态’对应于开关的‘闭合态’和‘断开态’
- **MOSFET**有‘线性电阻区’和‘截止区’对应于开关的‘闭合’和‘断开’
- 用晶体管实现开关是自然的选择

MOSFET开关

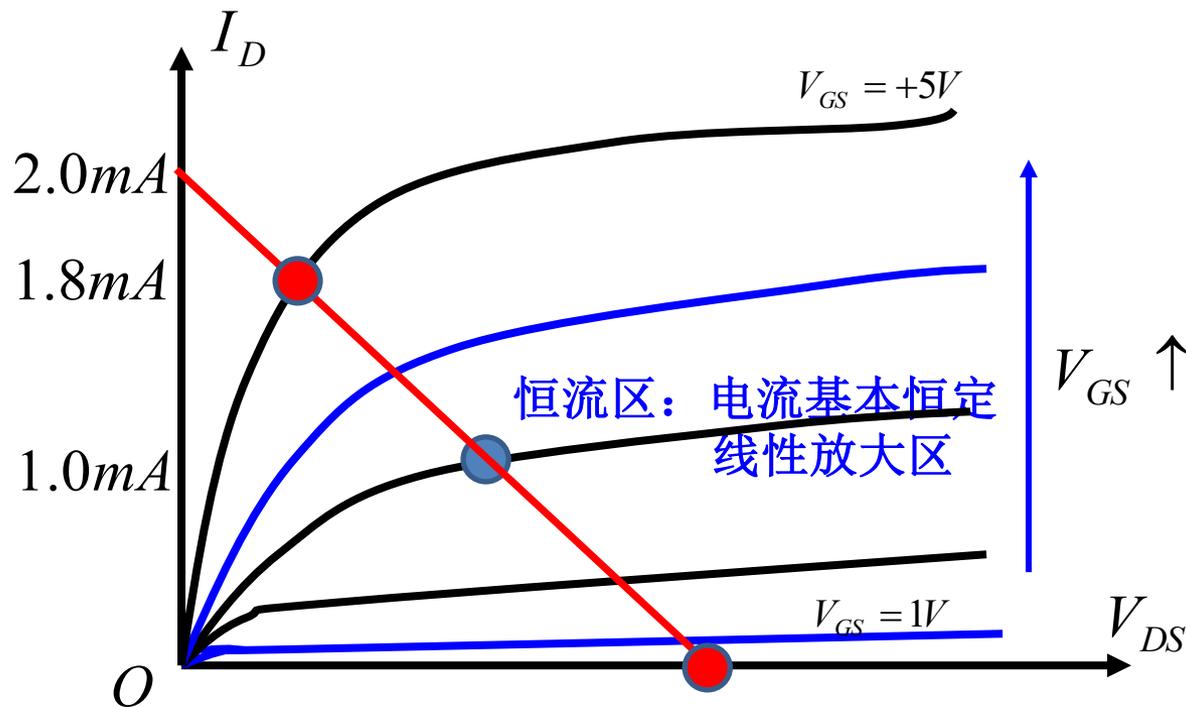
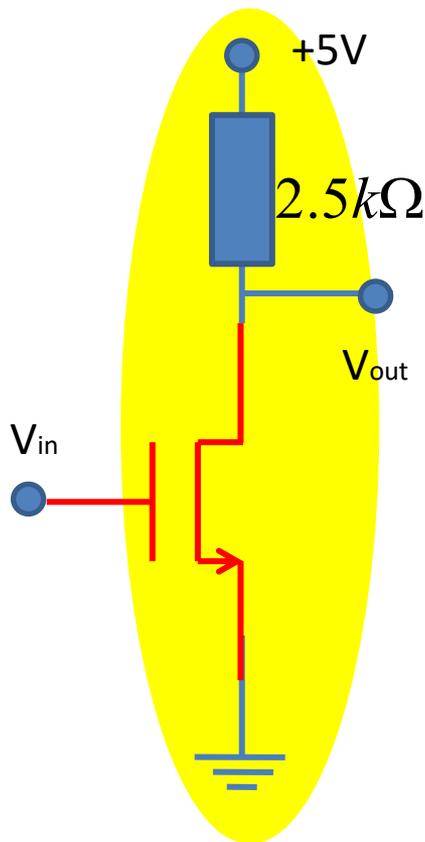


$$V_{out} = \begin{cases} +5V & \text{逻辑1: } V_{in} = \text{低电平(逻辑0)} \\ 0.5V & \text{逻辑0: } V_{in} = \text{高电平(逻辑1)} \end{cases}$$



截止区：电压大，电流小
对应开关断开态

电阻负载的功耗问题

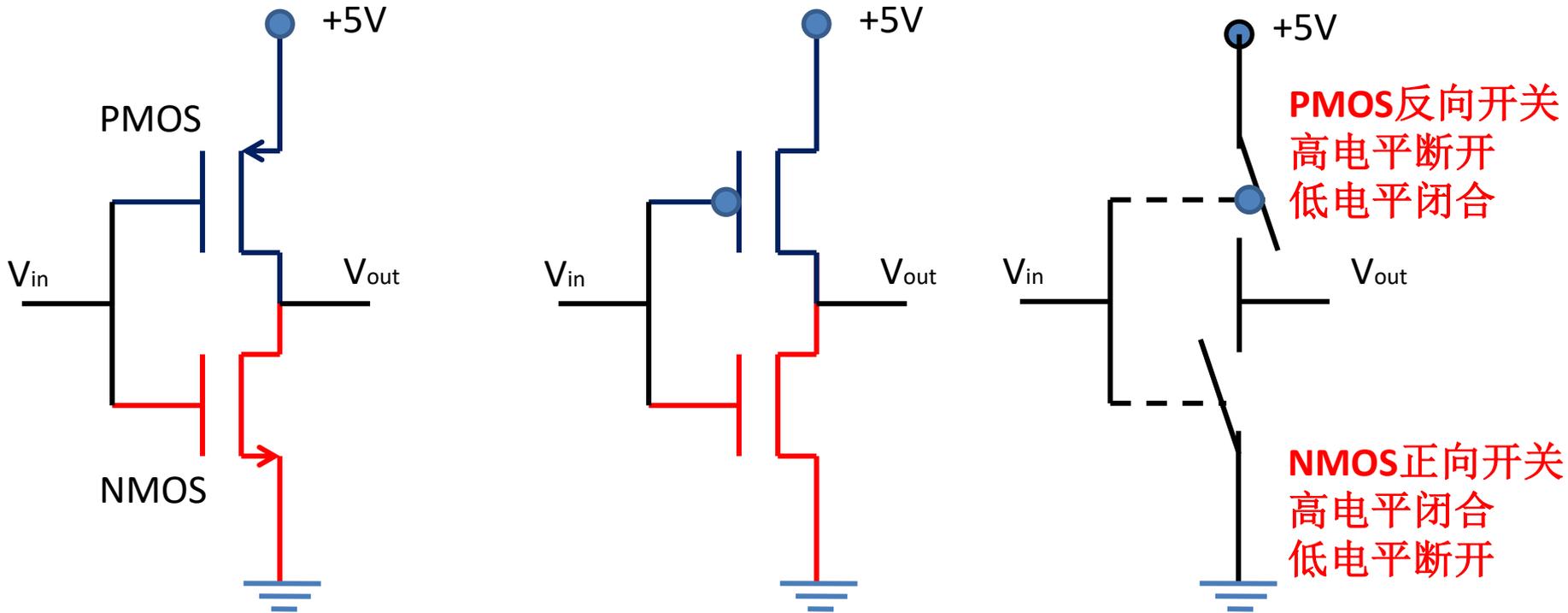


暂不考虑外接负载影响，输出悬空，外接负载为无穷大
开关断开时，晶体管截止，无电流，非门无功耗
开关闭合时，晶体管饱和导通，有大电流流过偏置电阻

$$P_{DC} = V_{DD} I_{DC} = 5V \times 1.8mA = 9mW$$

CMOS非门

Complementary MOS



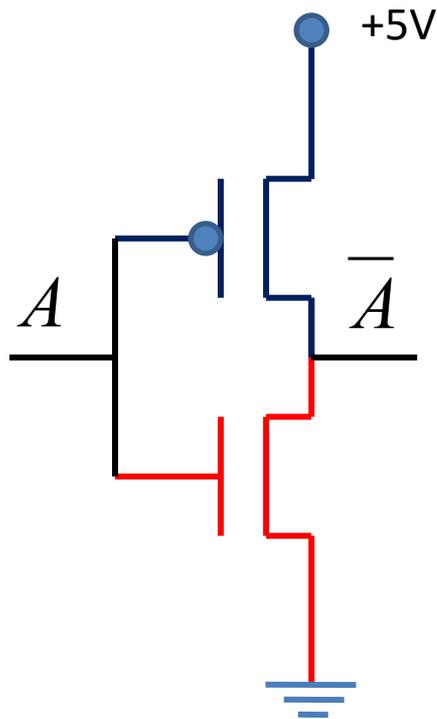
如果开关是理想的：导通 $U=0$ ，关断 $I=0$ ：无论导通或关断， $P=UI=0$

实际MOS开关并不十分理想，但足够接近理想开关，其静态功耗极低

CMOS门电路

PMOS逻辑与NMOS逻辑输出可点接：端口并联

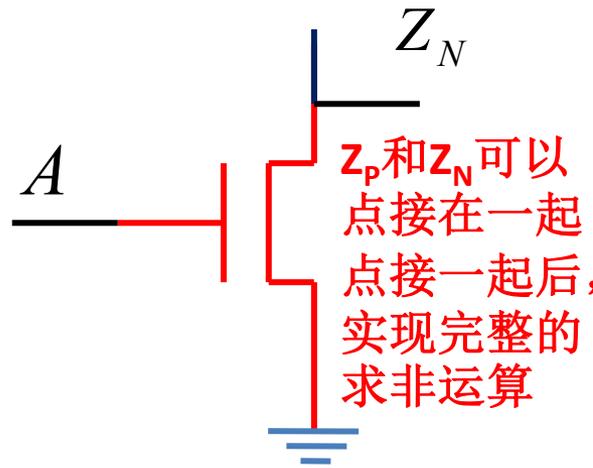
两个逻辑输出可点接并联的前提条件：输出完全一致或一个输出是悬空的



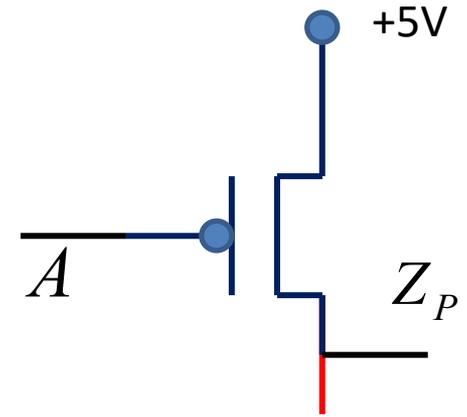
CMOS NOT Gate

PMOS和NMOS，一个饱和导通时另一个截止，消耗功率很小

$$Z_N = \begin{cases} \text{悬浮高阻态} & A = 0 \\ 0 = \bar{A} & A = 1 \end{cases}$$



NMOS正向开关旁路
输入高电平时，输出低电平，求非
输入低电平时，输出悬空

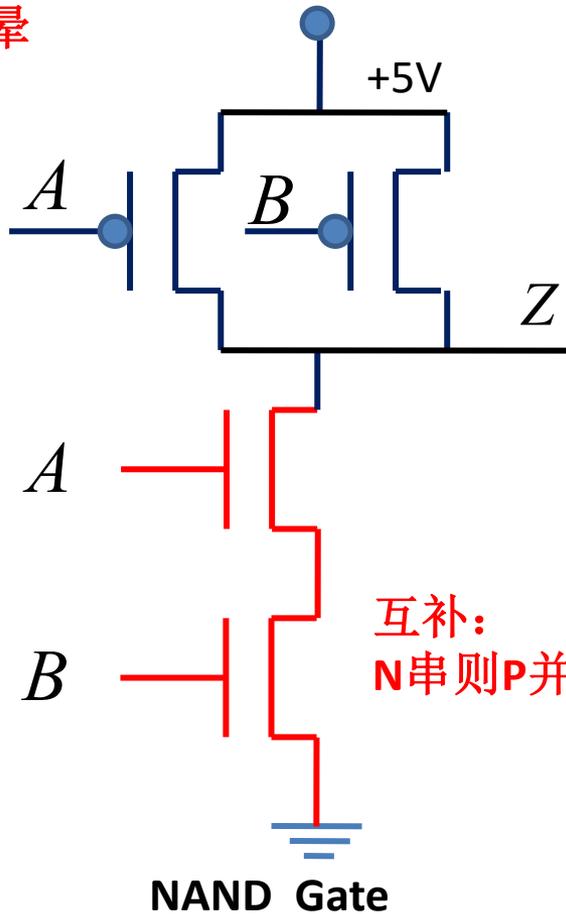


$$Z_P = \begin{cases} 1 = \bar{A} & A = 0 \\ \text{悬浮高阻态} & A = 1 \end{cases}$$

PMOS反向开关
输入低电平时，输出高电平，求非
输入高电平时，输出悬空

CMOS与非门

相同逻辑符号表明它们连在一起，免除连线过多头晕



$$Z_P = \begin{cases} \text{悬浮高阻态} & A \cdot B = 1 \\ \overline{A + B} = 1 = \overline{A \cdot B} & A \cdot B = 0 \end{cases}$$

开关并联或运算，反向开关先求非

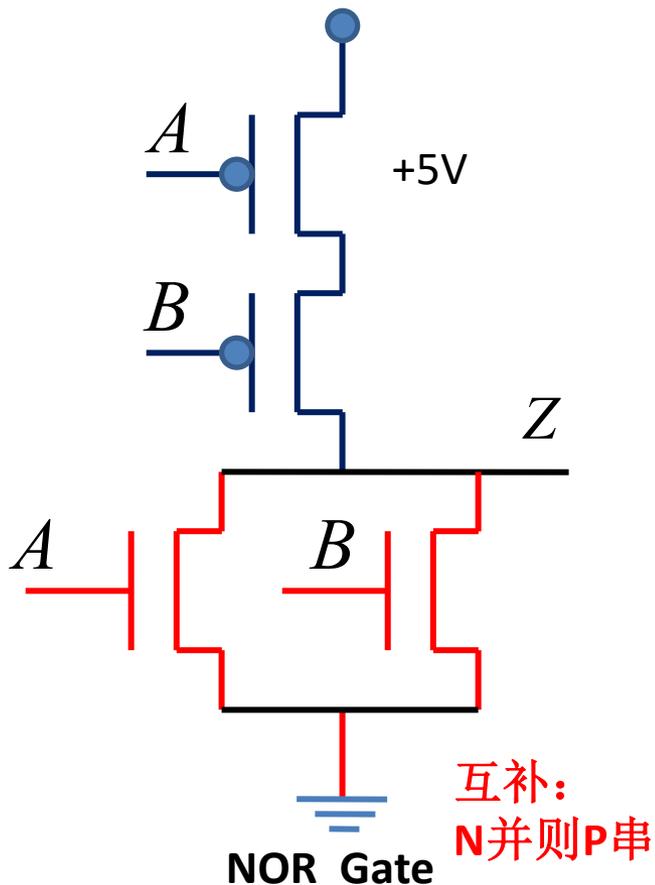
$$Z_N = \begin{cases} \overline{A \cdot B} = 0 & A \cdot B = 1 \\ \text{悬浮高阻态} & A \cdot B = 0 \end{cases}$$

开关串联与运算，旁路开关后求非

$$Z = Z_P \text{ 并 } Z_N = \overline{A \cdot B} = \overline{A} + \overline{B}$$

悬空如弃权默认，由并联电压决定最终逻辑输出

CMOS或非门



$$Z_P = \begin{cases} \overline{A \cdot B} = 1 = \overline{A + B} & A + B = 0 \\ \text{悬浮高阻态} & A + B = 1 \end{cases}$$

开关串联与运算，反向开关先求非

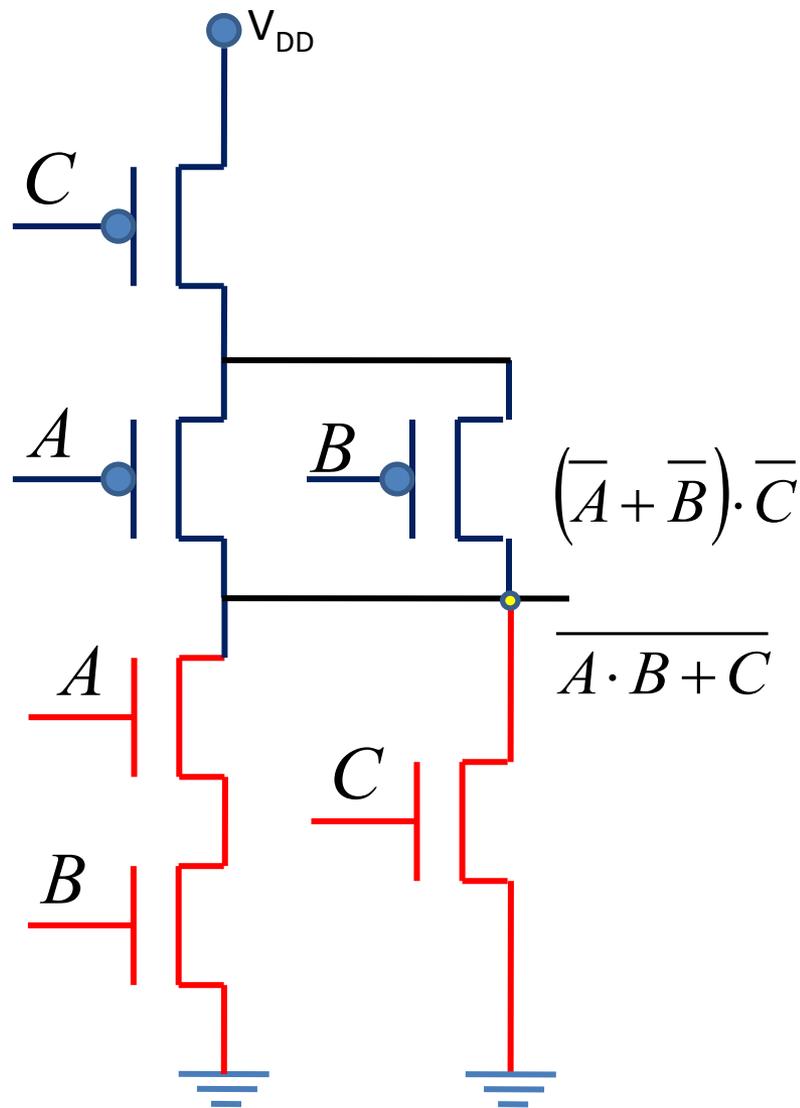
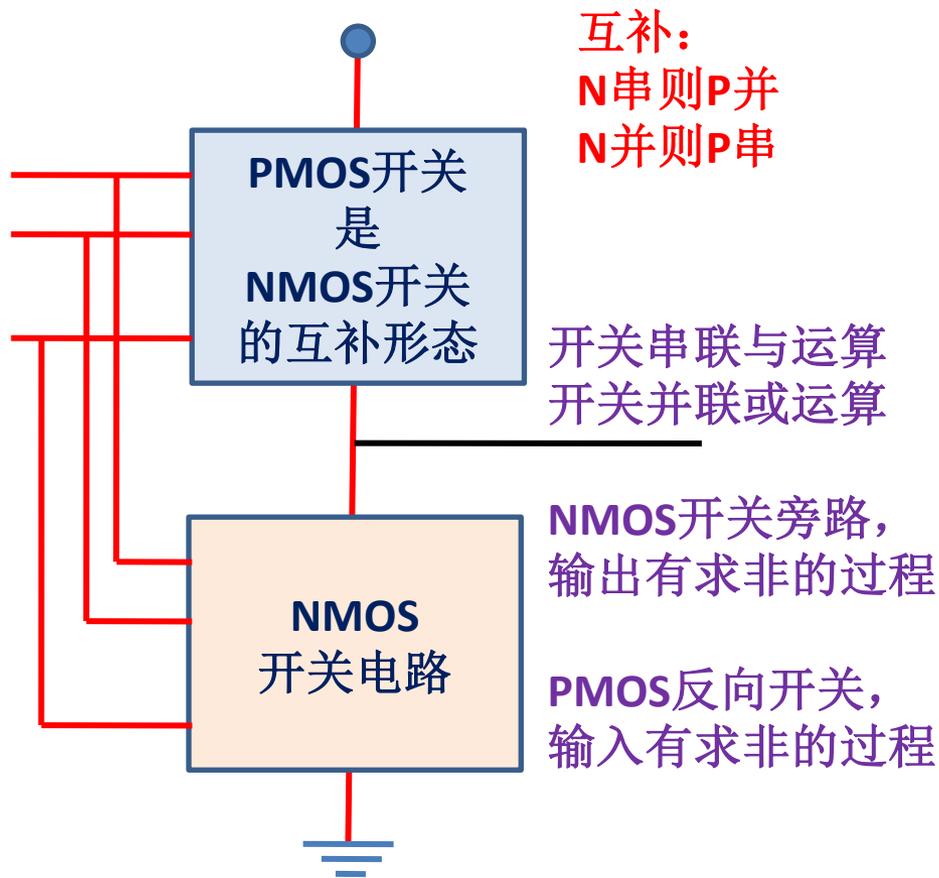
$$Z_N = \begin{cases} \text{悬浮高阻态} & A + B = 0 \\ \overline{A + B} = 0 & A + B = 1 \end{cases}$$

开关并联或运算，旁路开关后求非

$$Z = Z_P \text{ 并 } Z_N = \overline{A + B} = \overline{A} \cdot \overline{B}$$

悬空如弃权默认，由并联电压决定最终逻辑输出

CMOS门电路的一般框架



注意: 上P下N保共源组态, 晶体管不能随意调换位置

三、简单数字信号处理器设计例

- 用基本逻辑门电路搭建简单数字信号处理器
- 两个例子
 - 以二进制加法器的实现例来说明用逻辑运算可实现数学运算
 - 奇偶校验码可实现一定程度的检错

3.1 加法器

- 现欲实现两个8bit二进制数的相加功能，如何实现？

$$\begin{array}{r} \overset{\cdot}{1}\overset{\cdot}{0}\overset{\cdot}{1}\overset{\cdot}{1}\overset{\cdot}{0}100 \\ + 11101101 \\ \hline 110100001 \end{array}$$

$$\begin{array}{r} \overset{\cdot}{1}80 \\ + 237 \\ \hline 417 \end{array}$$

从后到前，一位一位地加，可能需要处理进位

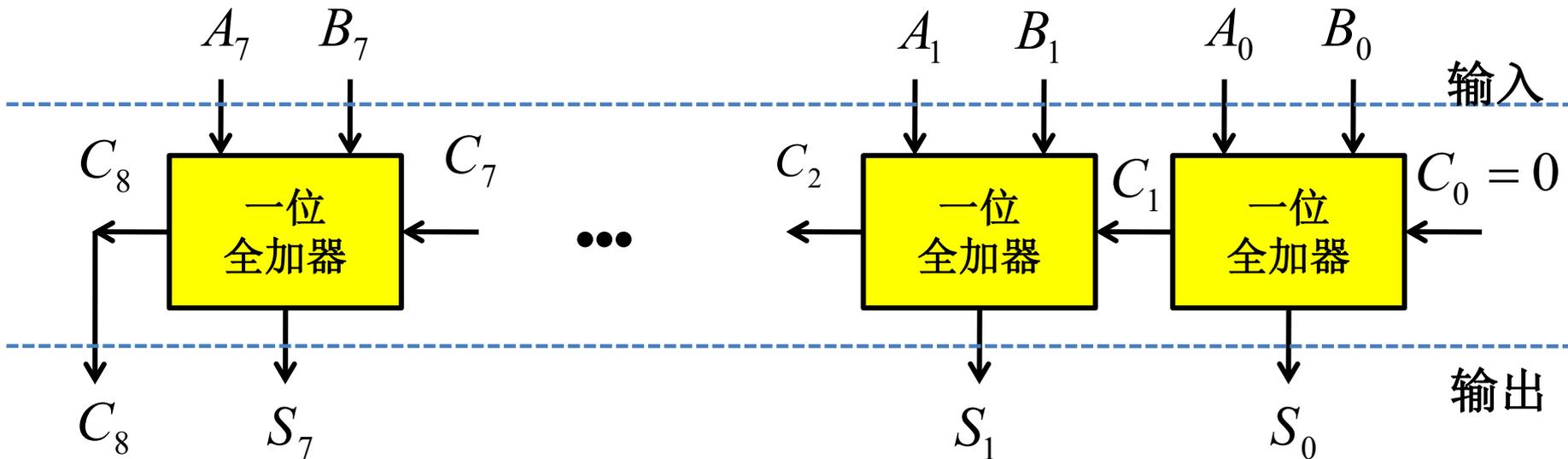
符号运算抽象

	C_8	C_7	C_6	C_5	C_4	C_3	C_2	C_1	0
	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
+	0	B_7	B_6	B_5	B_4	B_3	B_2	B_1	B_0
<hr/>									
	C_8	S_7	S_6	S_5	S_4	S_3	S_2	S_1	S_0

对每一位 i ，都是 $A_i+B_i+C_i$ ，产生和 S_i 以及进位 C_{i+1}

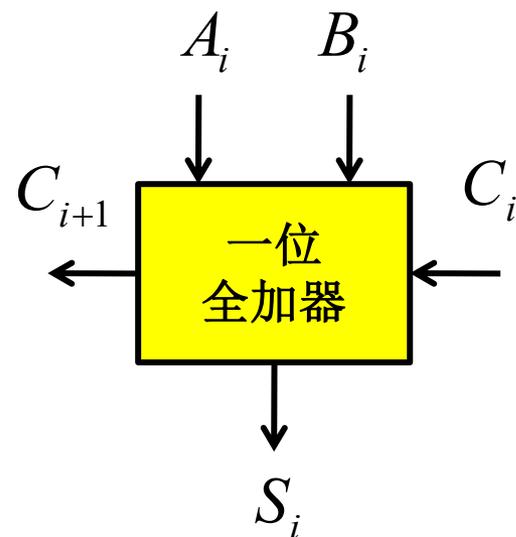
加法器系统结构

- 一位一位地加，同时考虑进位
 - 8个一位全加器级联，即可实现两个8位二进制数的相加功能



如何实现一位全加器

A_i	B_i	C_i	C_{i+1}	S_i
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



$$S_i = \overline{A_i} \cdot \overline{B_i} \cdot C_i + \overline{A_i} \cdot B_i \cdot \overline{C_i} + A_i \cdot \overline{B_i} \cdot \overline{C_i} + A_i \cdot B_i \cdot C_i$$

$$C_{i+1} = \overline{A_i} \cdot B_i \cdot C_i + A_i \cdot \overline{B_i} \cdot C_i + A_i \cdot B_i \cdot \overline{C_i} + A_i \cdot B_i \cdot C_i$$

卡诺图化简

$$S_i = \overline{A_i} \cdot \overline{B_i} \cdot C_i + \overline{A_i} \cdot B_i \cdot \overline{C_i} + A_i \cdot \overline{B_i} \cdot \overline{C_i} + A_i \cdot B_i \cdot C_i$$

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

$$C_{i+1} = \overline{A_i} \cdot B_i \cdot C_i + A_i \cdot \overline{B_i} \cdot C_i + A_i \cdot B_i \cdot \overline{C_i} + A_i \cdot B_i \cdot C_i$$

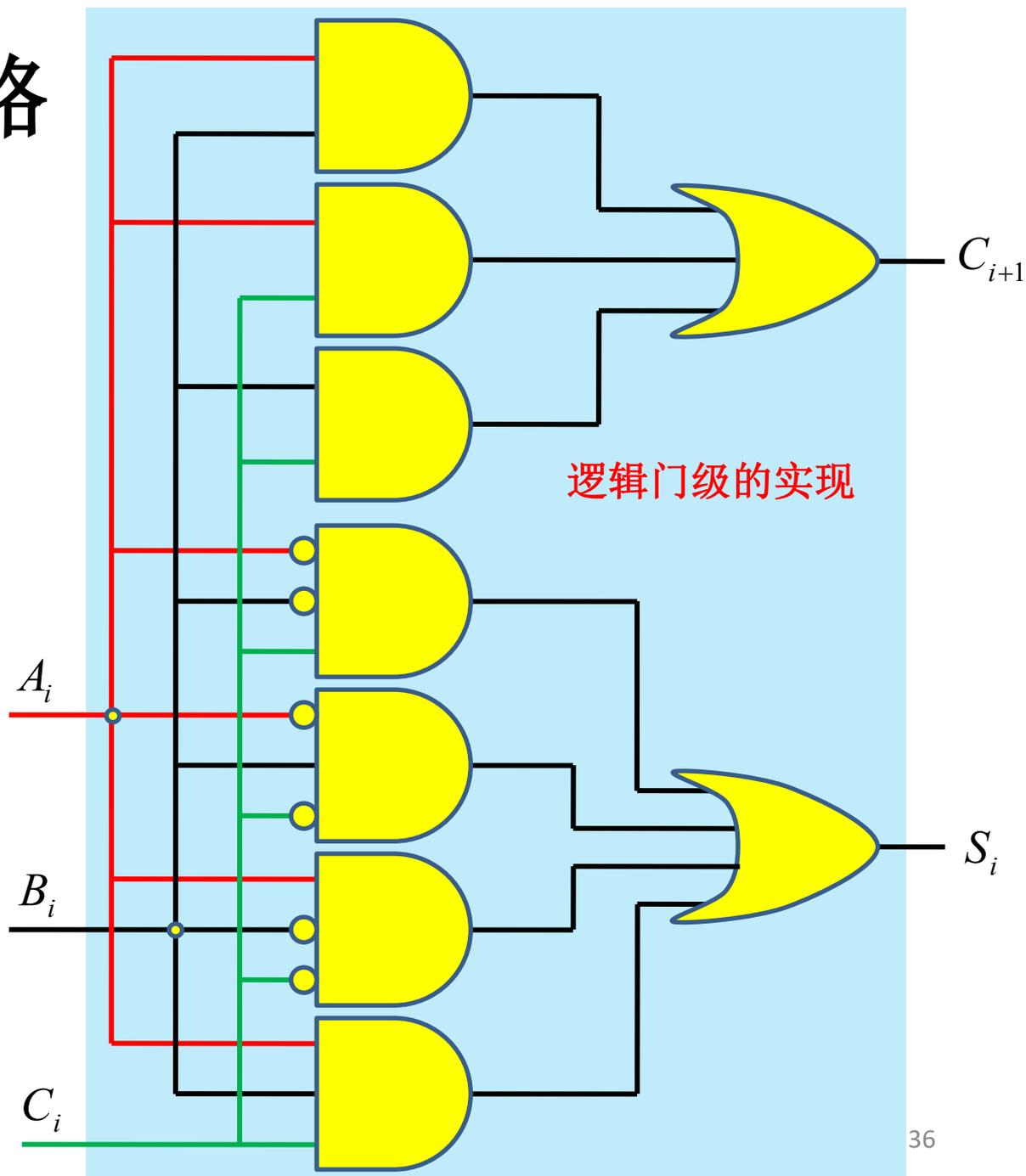
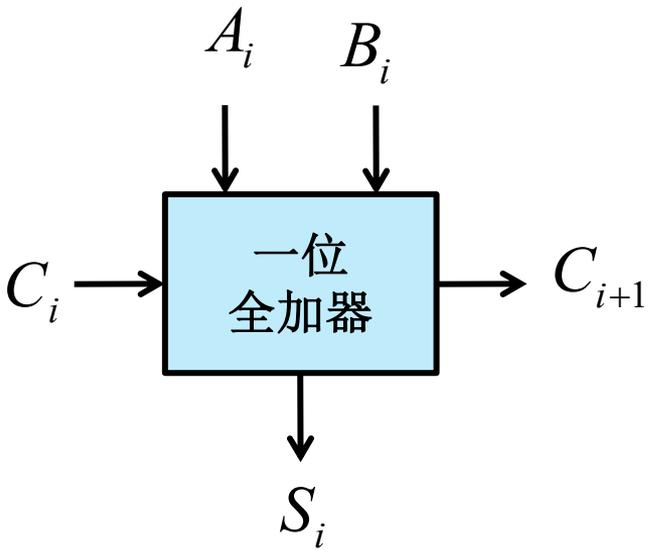
C \ AB	00	01	11	10
0	0	0	1	0
1	0	1	1	1

$$C_{i+1} = A_i B_i + B_i C_i + A_i C_i$$

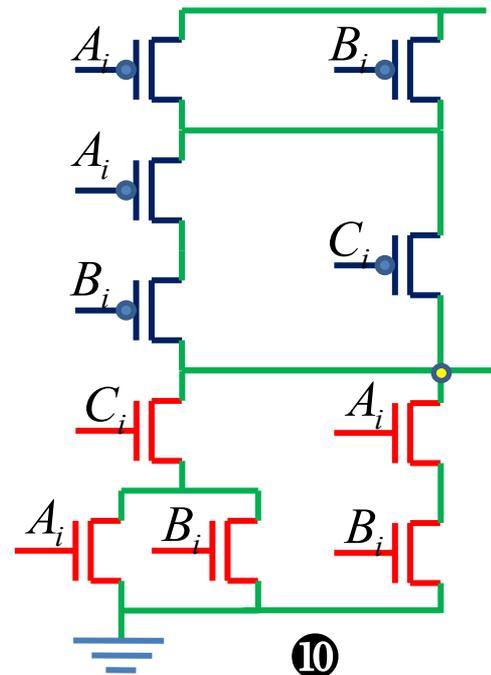
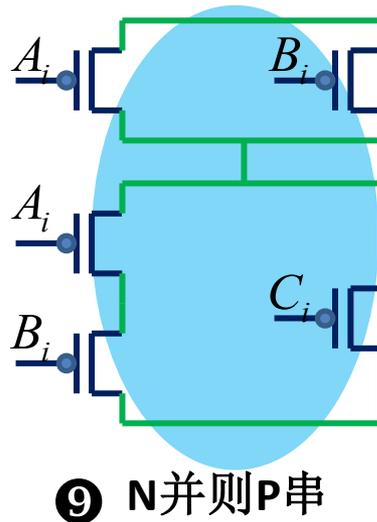
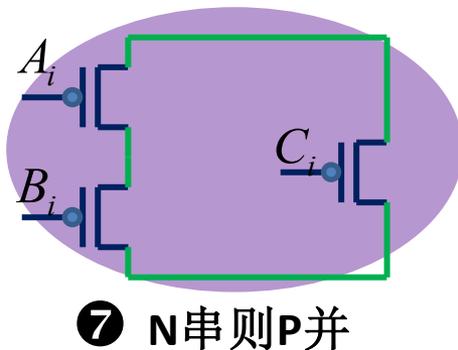
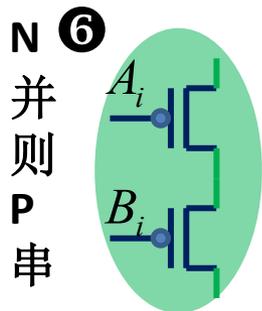
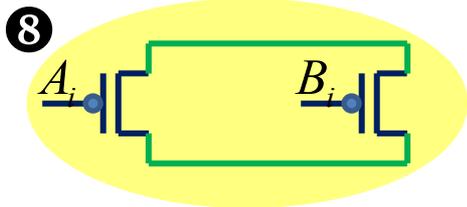
一位全加器电路

$$S_i = \overline{A_i} \cdot \overline{B_i} \cdot C_i + \overline{A_i} \cdot B_i \cdot \overline{C_i} + A_i \cdot \overline{B_i} \cdot \overline{C_i} + A_i \cdot B_i \cdot C_i$$

$$C_{i+1} = B_i \cdot C_i + A_i \cdot C_i + A_i \cdot B_i$$



N 串则 P 并



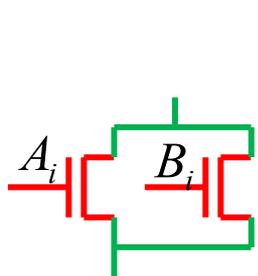
进位逻辑

⑩ $C_{i+1} = B_i C_i + A_i C_i + A_i B_i = (A_i + B_i) C_i + A_i B_i$

晶体管级的CMOS电路实现

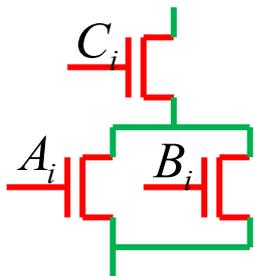
① 或则并

$$A_i + B_i$$



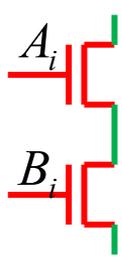
② 与则串

$$(A_i + B_i) C_i$$



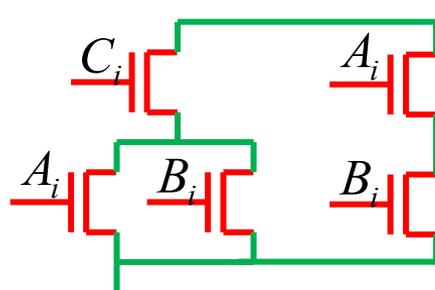
③ 与则串

$$A_i B_i$$



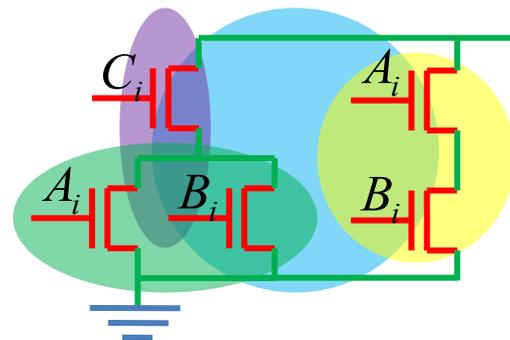
④ 或则并

$$(A_i + B_i) C_i + A_i B_i$$



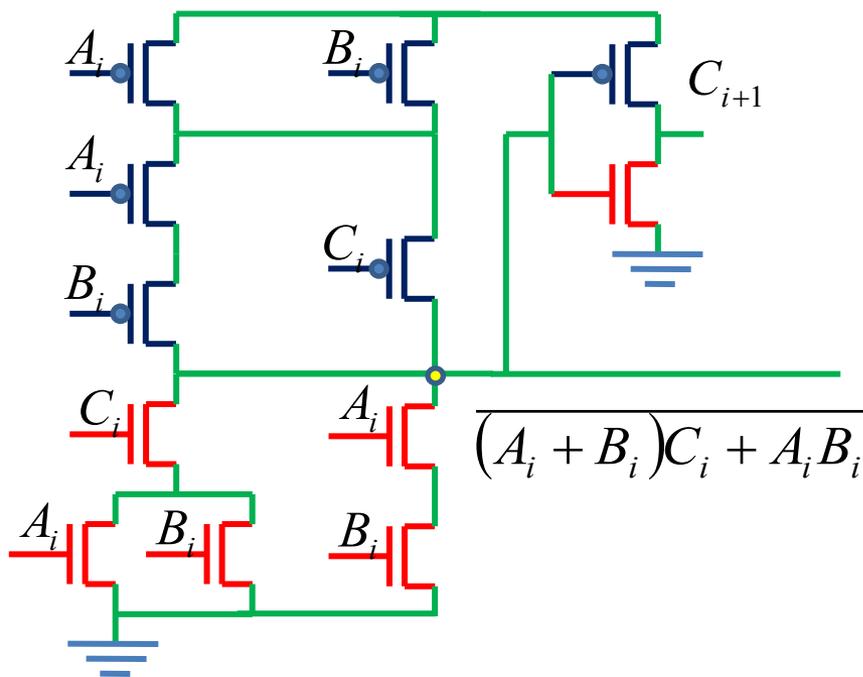
⑤ 旁则非

$$\overline{(A_i + B_i) C_i + A_i B_i}$$



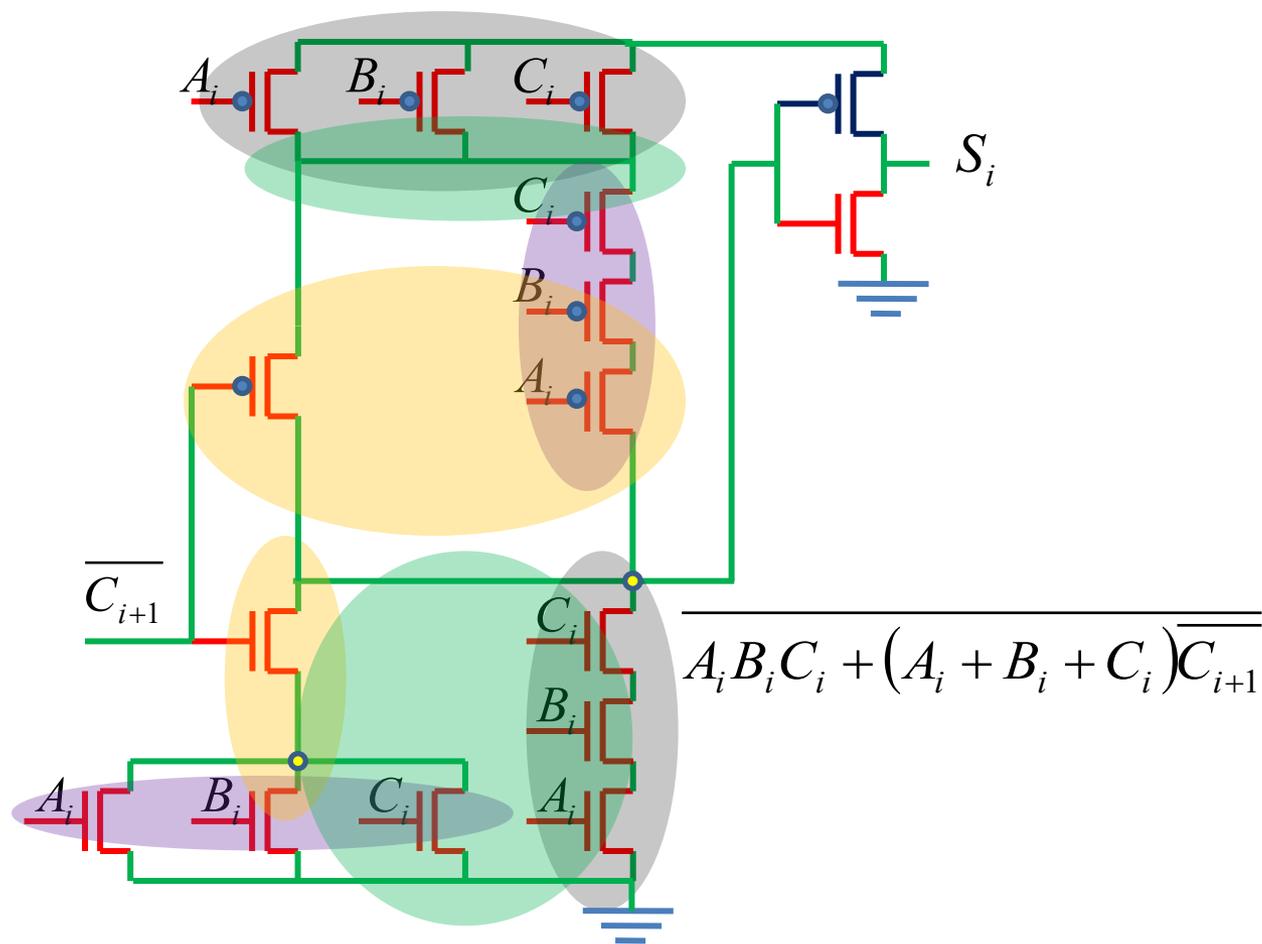
进位逻辑的CMOS电路实现

$$C_{i+1} = B_i C_i + A_i C_i + A_i B_i = (A_i + B_i) C_i + A_i B_i$$

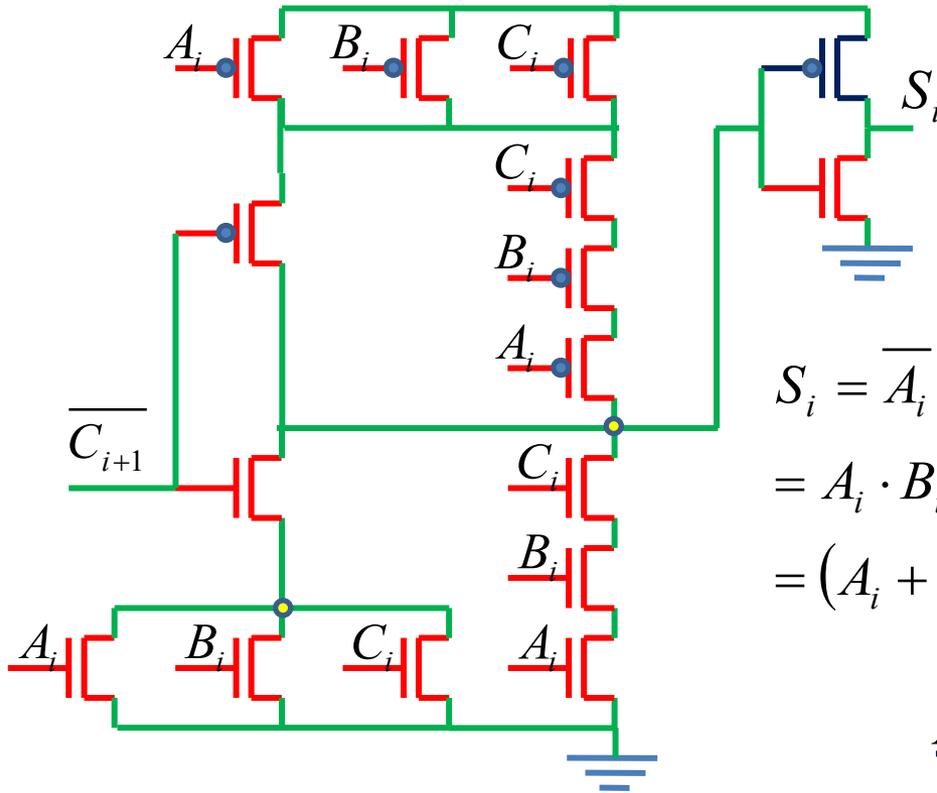


$$S_i = \overline{A_i} \cdot \overline{B_i} \cdot C_i + \overline{A_i} \cdot B_i \cdot \overline{C_i} + A_i \cdot \overline{B_i} \cdot \overline{C_i} + A_i \cdot B_i \cdot C_i = A_i \cdot B_i \cdot C_i + (A_i + B_i + C_i) \cdot \overline{C_{i+1}}$$

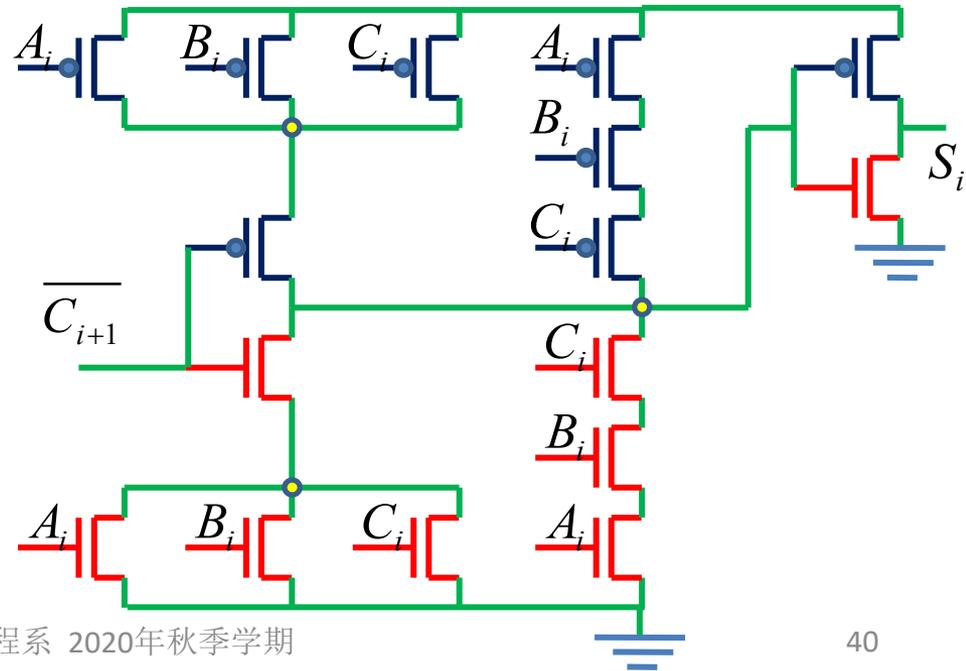
和位逻辑的实现



另一种方式



$$\begin{aligned}
 S_i &= \overline{A_i} \cdot \overline{B_i} \cdot C_i + \overline{A_i} \cdot B_i \cdot \overline{C_i} + A_i \cdot \overline{B_i} \cdot \overline{C_i} + A_i \cdot B_i \cdot C_i \\
 &= A_i \cdot B_i \cdot C_i + (A_i + B_i + C_i) \cdot \overline{C_{i+1}} \\
 &= (A_i + B_i + C_i) \cdot (A_i \cdot B_i \cdot C_i + \overline{C_{i+1}})
 \end{aligned}$$



功能完全一样

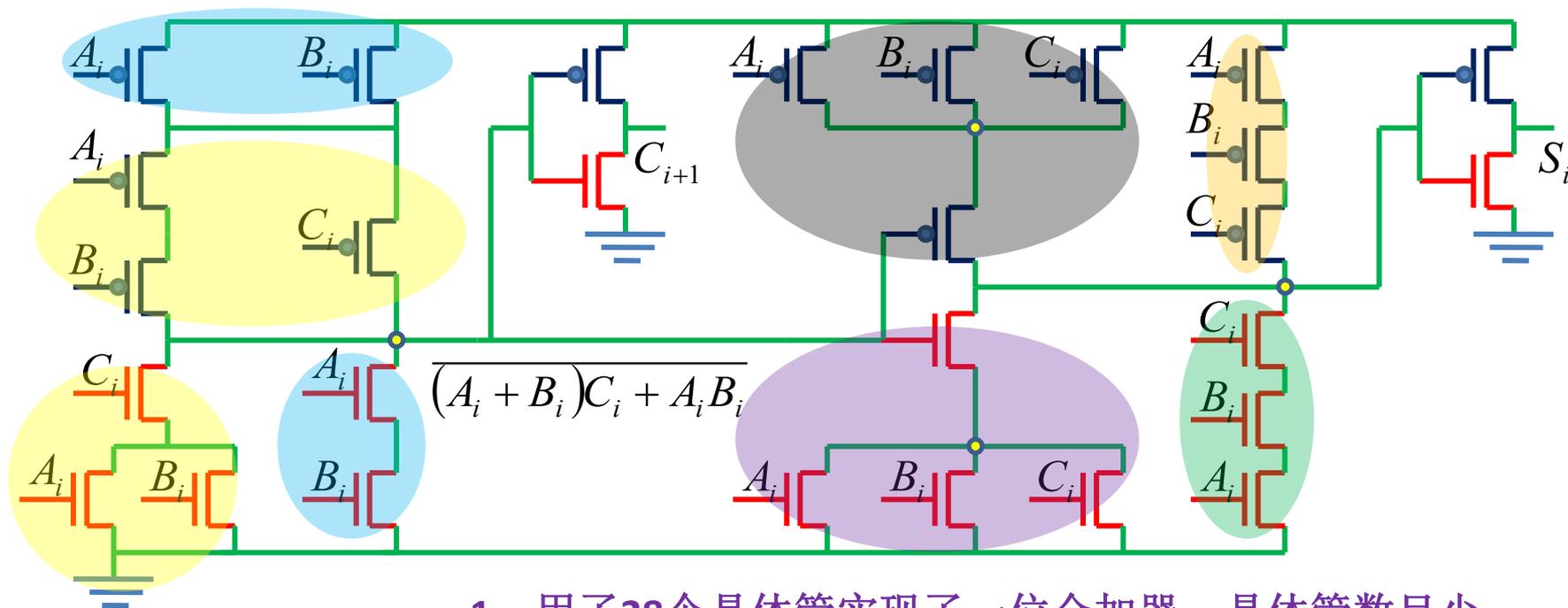
后者更对称，更紧致
结构不具互补特征，但逻辑
功能是互补的，故可点接

一种一位全加器的CMOS实现方案

$$C_{i+1} = B_i C_i + A_i C_i + A_i B_i = (A_i + B_i) C_i + A_i B_i$$

$$S_i = \overline{A_i} \cdot \overline{B_i} \cdot C_i + \overline{A_i} \cdot B_i \cdot \overline{C_i} + A_i \cdot \overline{B_i} \cdot \overline{C_i} + A_i \cdot B_i \cdot C_i = A_i \cdot B_i \cdot C_i + (A_i + B_i + C_i) \cdot \overline{C_{i+1}}$$

$$= (A_i + B_i + C_i) \cdot (A_i \cdot B_i \cdot C_i + \overline{C_{i+1}})$$



- 1、用了28个晶体管实现了一位全加器；晶体管数目少
- 2、输出用反相器，犹如缓冲器，输出逻辑电平更稳定

4.2 奇偶校验可检部分错

- 发送端，对 n 位数据，加1位奇偶校验位，变成 $n+1$ 位数据
 - 规则：偶数个1
- $n+1$ 位数据传输，噪声、干扰等导致误码
- 接收端接收到数据后，进行奇偶校验，满足则认为传输正确，不满足则告警---传输出现错误
 - 不告警未必传输正确，但告警一定出现传输错误
 - 两位同时出错，仍然满足规则，传输错误但没有告警

奇偶校验位的形成

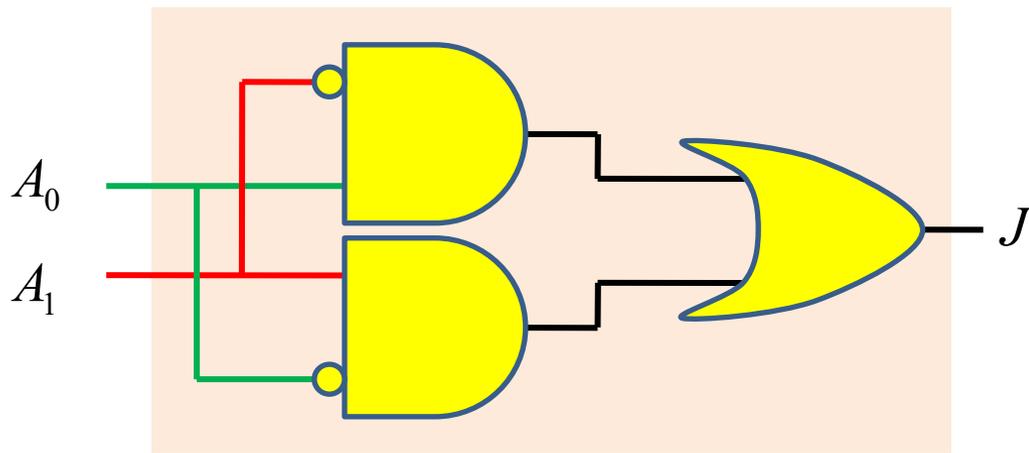
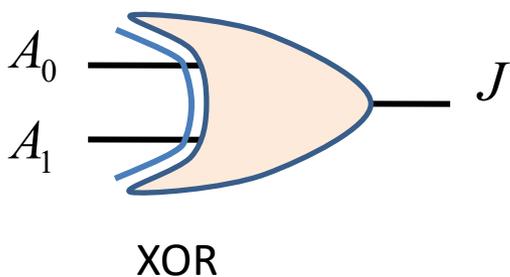
A_0	A_1	J
0	0	0
0	1	1
1	0	1
1	1	0

$$J = \overline{A_0}A_1 + A_0\overline{A_1} = A_0 \oplus A_1$$

异或门:有不同意见则通过, 意见一致则否决
XOR: Exclusive OR

传输数据J-A1-A0

无法卡诺图化简的交错形态



奇偶校验可检奇数位错 偶数位错不可检

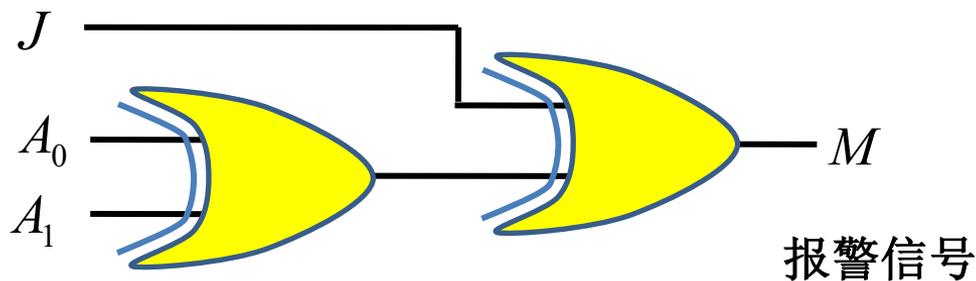
A_0	A_1	J	M
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

无法卡诺图化简的交错形态

$$\begin{aligned}
 M &= \overline{A_0} \cdot \overline{A_1} \cdot J + \overline{A_0} \cdot A_1 \cdot \overline{J} + A_0 \cdot \overline{A_1} \cdot \overline{J} + A_0 \cdot A_1 \cdot J \\
 &= J \cdot (\overline{A_0} \cdot \overline{A_1} + A_0 \cdot A_1) + \overline{J} \cdot (\overline{A_0} \cdot A_1 + A_0 \cdot \overline{A_1}) \\
 &= J \cdot (\overline{A_0 \oplus A_1}) + \overline{J} \cdot (A_0 \oplus A_1) \\
 &= J \oplus (A_0 \oplus A_1)
 \end{aligned}$$

如果校验位正确，则不报警 $J = A_0 \oplus A_1$

$$M = 0$$

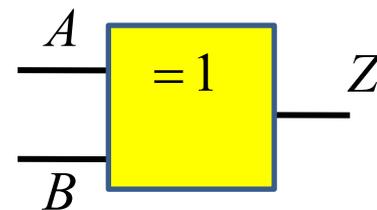
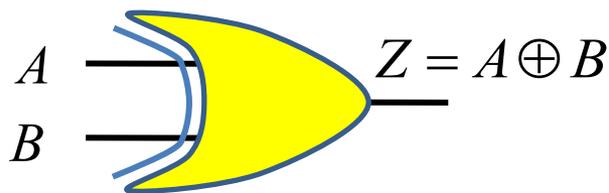


异或门、异或非门

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

异或门：加法和位

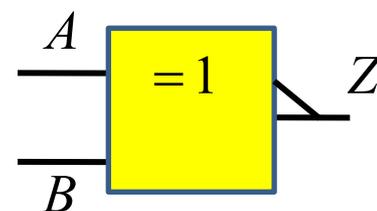
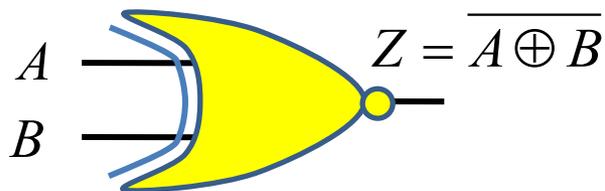
异或门:两个不同则正确, 两个相同则错误
XOR: Exclusive OR



A	B	$\overline{A \oplus B}$
0	0	1
0	1	0
1	0	0
1	1	1

异或非门：简单乘法

异或非门:两个相同则正确, 两个不同则错误
XNOR: Exclusive NOR



数字电路的优势

- 数字电路的优势是可大规模集成
 - 数字电路只需两个状态（容差性强）
 - CMOS电路功耗极低
 - 面积极小
 - 英特尔·酷睿™2四核处理器含有5.8亿多个晶体管
 - 大规模集成数字电路(VLSI: Very Large Scale Integrated Circuit)可实现复杂的运算、处理和控制功能
- 大规模模拟电路很难设计和调试，完成的信号处理功能也相对单一



信息化社会是数字化的

- 数字电路的优势
 - 抗干扰能力强：容差性强
 - 功耗低
 - 面积小
 - 可大规模集成：构建复杂的信息处理系统
- 信息化社会是数字化的

但是，信息系统的构建不可能是 数字电路可以单挑的

- 射频电路/能量转换电路无法数字化
 - 含无线通信：4G/5G通信，可植入医疗设备，...
 - 含能量转换：电源管理系统，放大器，振荡器，传感器和激励器（触摸屏，指纹识别，加速度计，屏幕驱动，音频驱动，...），...
- 即使是数字电路，也需模拟电路的设计技术才能有效处理高速问题
 - 处理器时钟 ~GHz：高频下的寄生电容、寄生电感效应和传输线效应，使得我们必须用模拟电路设计理念去设计数字系统，系统才能正常工作
- 有些问题用模拟电路可能较数字电路更有效，如模糊计算等
- 模拟电路和数字电路是相辅相成的关系
 - 模拟电路面向实际的物理世界进行信息处理，数字电路则是面向数字化信息世界的通道，两者缺一不可构建信息化社会，毕竟世界本身是模拟的，但纯模拟的信息无法传承

本讲小结

- 与、或、非是布尔二值逻辑的基本运算
- 真值表是逻辑运算结果的表格表述
 - 如果两个逻辑表达式的真值表一样，这两个表达式则等价
- 卡诺图是三输入、四输入逻辑运算化简的基本手段
- 二值逻辑运算可以通过开关通断实现
 - 旁路开关后求非，反相开关先求非
 - 开关并联或运算，开关串联与运算
- **CMOS**门电路的**PMOS**开关电路是**NMOS**开关电路的互补，是德摩根律的具体应用
- 各种数字信号处理功能均可通过逻辑运算实现
- 信息化社会是数字化社会，但对于信息处理系统构建而言，模拟电路和数字电路相辅相成缺一不可

作业1：一位全加器

- 证明一位全加器的两个逻辑表达式是成立的
 - 根据逻辑表达式，复画CMOS电路
 - 思考：考察这样的逻辑表达式在CMOS门电路实现上有什么好处？为什么这样表述？

$$C_{i+1} = B_i C_i + A_i C_i + A_i B_i = (A_i + B_i) C_i + A_i B_i$$

$$\begin{aligned} S_i &= \overline{A_i} \cdot \overline{B_i} \cdot C_i + \overline{A_i} \cdot B_i \cdot \overline{C_i} + A_i \cdot \overline{B_i} \cdot \overline{C_i} + A_i \cdot B_i \cdot C_i = A_i \cdot B_i \cdot C_i + (A_i + B_i + C_i) \cdot \overline{C_{i+1}} \\ &= (A_i + B_i + C_i) \cdot (A_i \cdot B_i \cdot C_i + \overline{C_{i+1}}) \end{aligned}$$

真值表一致
或者运用逻辑运算规则证明

思考：很多种表达式代表同一逻辑运算，但哪种是最适宜于CMOS电路实现的呢？

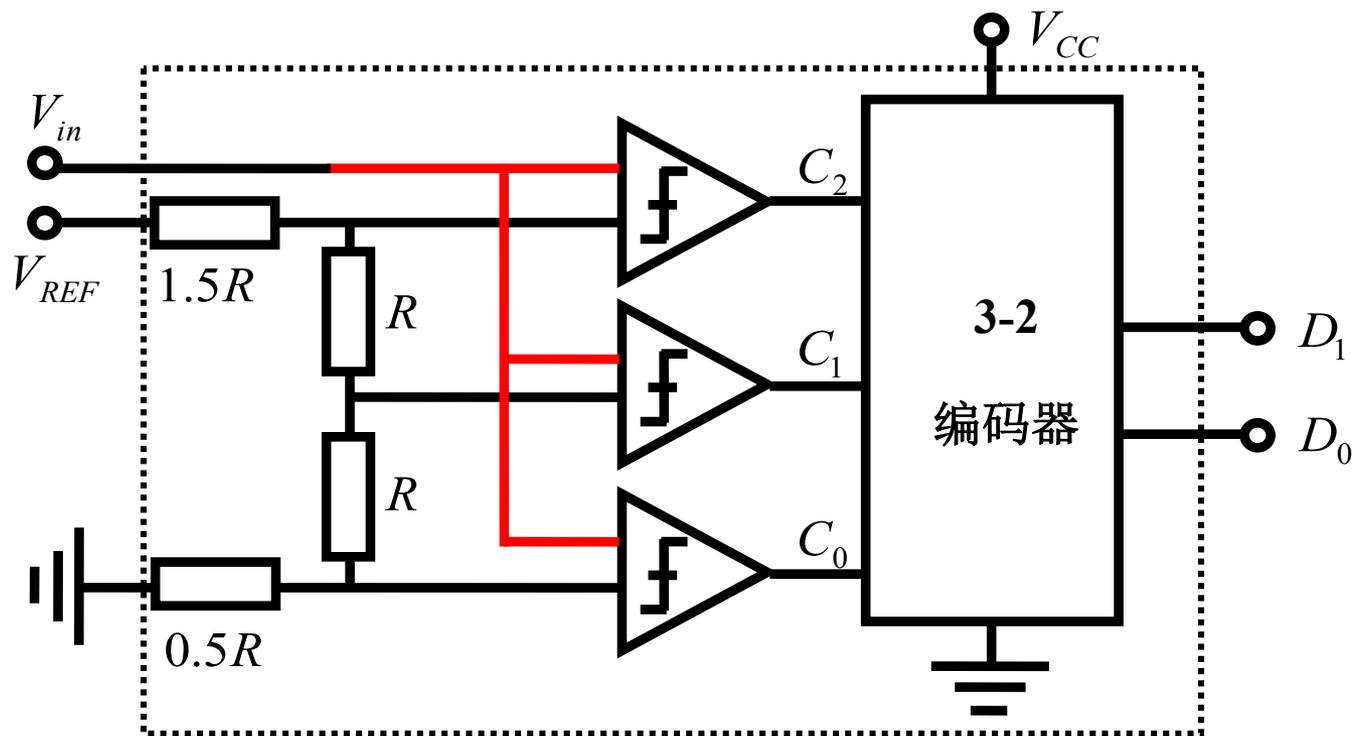
作业2：卡诺图化简

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

化简卡诺图，写出输出Z用ABCD表述的逻辑表达式

用PMOS互补NMOS的CMOS电路形态（上P下N，形式互补）实现这些逻辑运算，画出CMOS晶体管级电路图

FLASH



ADC

模拟输入电压	$C_2C_1C_0$	数字输出码 D_1D_0
	000	00
		01
		10
		11

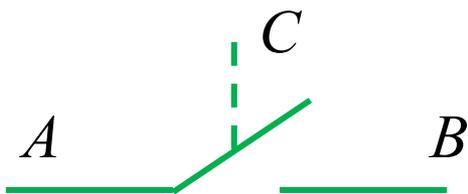
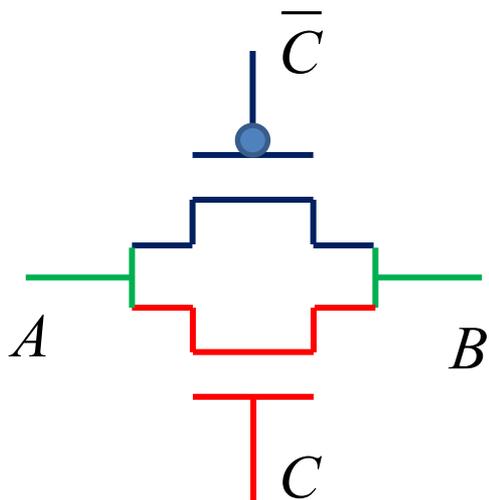
作业3：编码器设计

- 已知flash-ADC的码表如左图
- 设计编码器，实现flash-ADC的正确输出，并画出编码器的CMOS实现方案
 - 用卡诺图进行化简
 - CMOS: 上P下N，形式互补
 - C_2 、 C_1 、 C_0 已经经过电平转换电路，使得逻辑1对应电压 V_{DD} ，逻辑0则对应地电压

$C_2C_1C_0$	数字输出码 D_1D_0
000	00
001	01
011	10
111	11

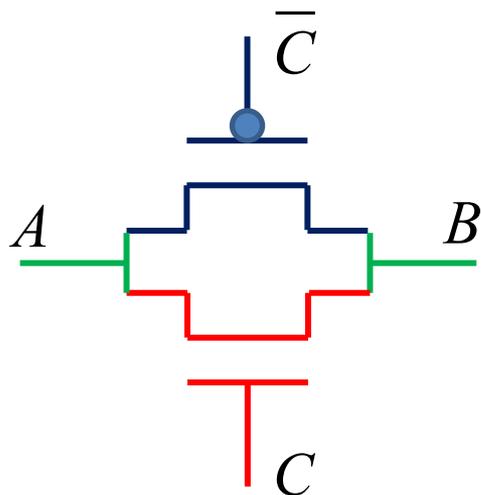
三个输入变量，共8种情况，这里仅有4种情况剩下4种情况真值表中都是*(是0是1不在乎)，因为在实际电路中，这4种情况不会也不应该出现

作业4：信号传输路径上的传输开关

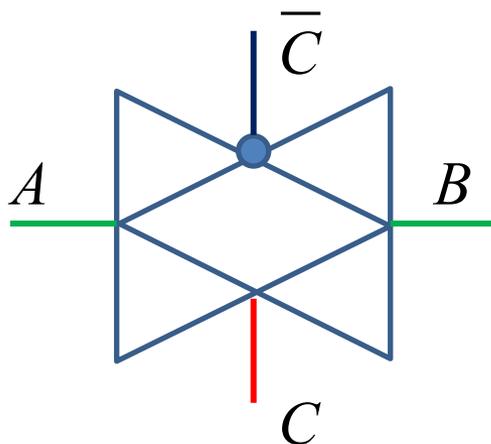


- 图示为经典的传输开关CMOS实现方案
 - 假设A接逻辑1源（输入端），B接负载电阻/电容（输出端）
 - 当C为高电平（逻辑1）时，（NMOS栅极电压高电平，NMOS恒流导通，）PMOS栅极电压低电平，PMOS欧姆导通，等效为开关的闭合状态，负载电压高电平（逻辑1传输）
 - 当C为低电平（逻辑0）时，NMOS和PMOS均截至，输出悬空（负载电阻接地，逻辑0）
 - 假设A接逻辑0源（输入端），B接负载电阻/电容（输出端）
 - 当C为高电平（逻辑1）时，NMOS栅极电压高电平，NMOS欧姆导通（PMOS栅极电压低电平，PMOS恒流导通），等效为开关的闭合状态，负载电压低电平（逻辑0传输）
 - ...

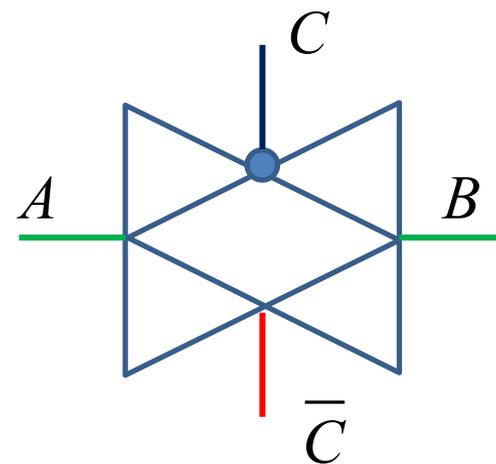
CMOS传输开关符号



双向导通



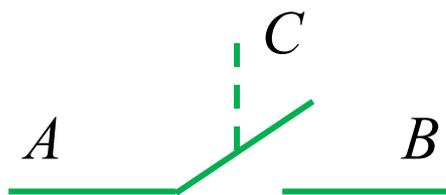
$$B = \begin{cases} CA & C = 1 \\ \text{悬空} & C = 0 \end{cases}$$



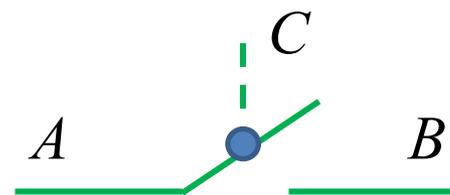
$$B = \begin{cases} \bar{C}A & C = 0 \\ \text{悬空} & C = 1 \end{cases}$$

假设A为输入（A之前电路可等效为源）

B为输出（B之后电路可等效为负载）



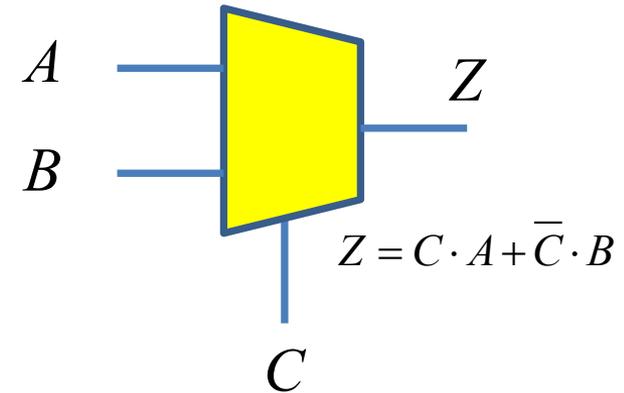
C为1时允许A过去



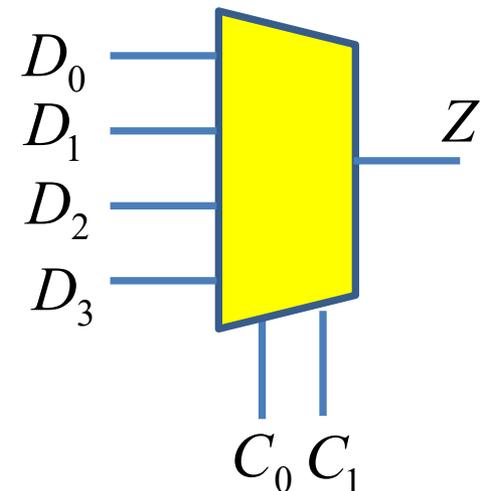
C为0时允许A过去

作业：多路选择器Multiplexer设计

- 4、用CMOS电路实现一个双路选择器，里面采用CMOS传输开关，画出晶体管级CMOS电路图
 - 如果控制端C=1，则传输A
 - 如果控制端C=0，则传输B
- 5、请设计一个4路选择器，画出CMOS实现电路图
 - 如果控制端C₁C₀=00，则传输D₀
 - 如果控制端C₁C₀=01，则传输D₁
 - 如果控制端C₁C₀=10，则传输D₂
 - 如果控制端C₁C₀=11，则传输D₃

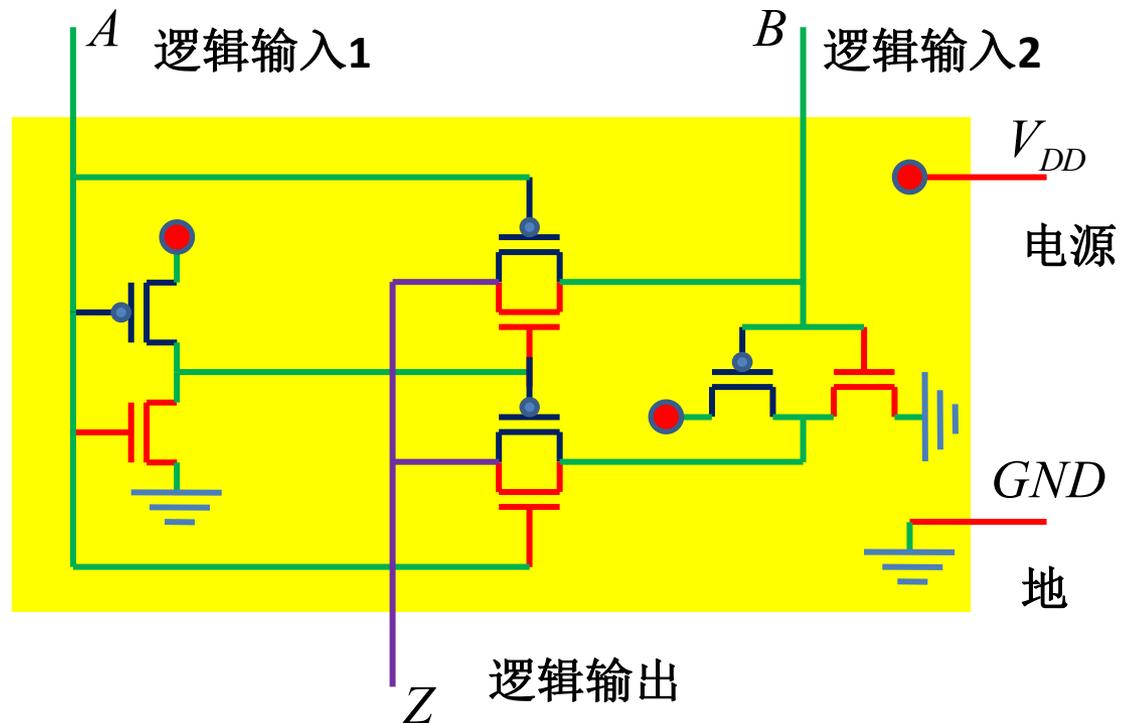
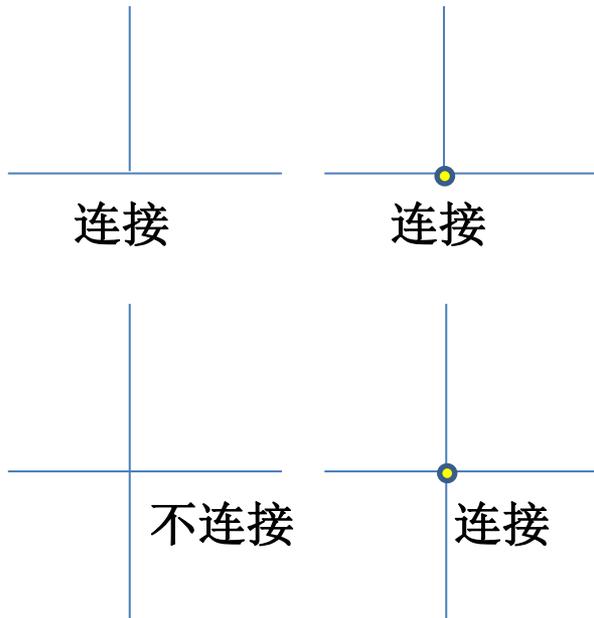


$$Z = \bar{C}_1 \cdot \bar{C}_0 \cdot D_0 + \bar{C}_1 \cdot C_0 \cdot D_1 + C_1 \cdot \bar{C}_0 \cdot D_2 + C_1 \cdot C_0 \cdot D_3$$



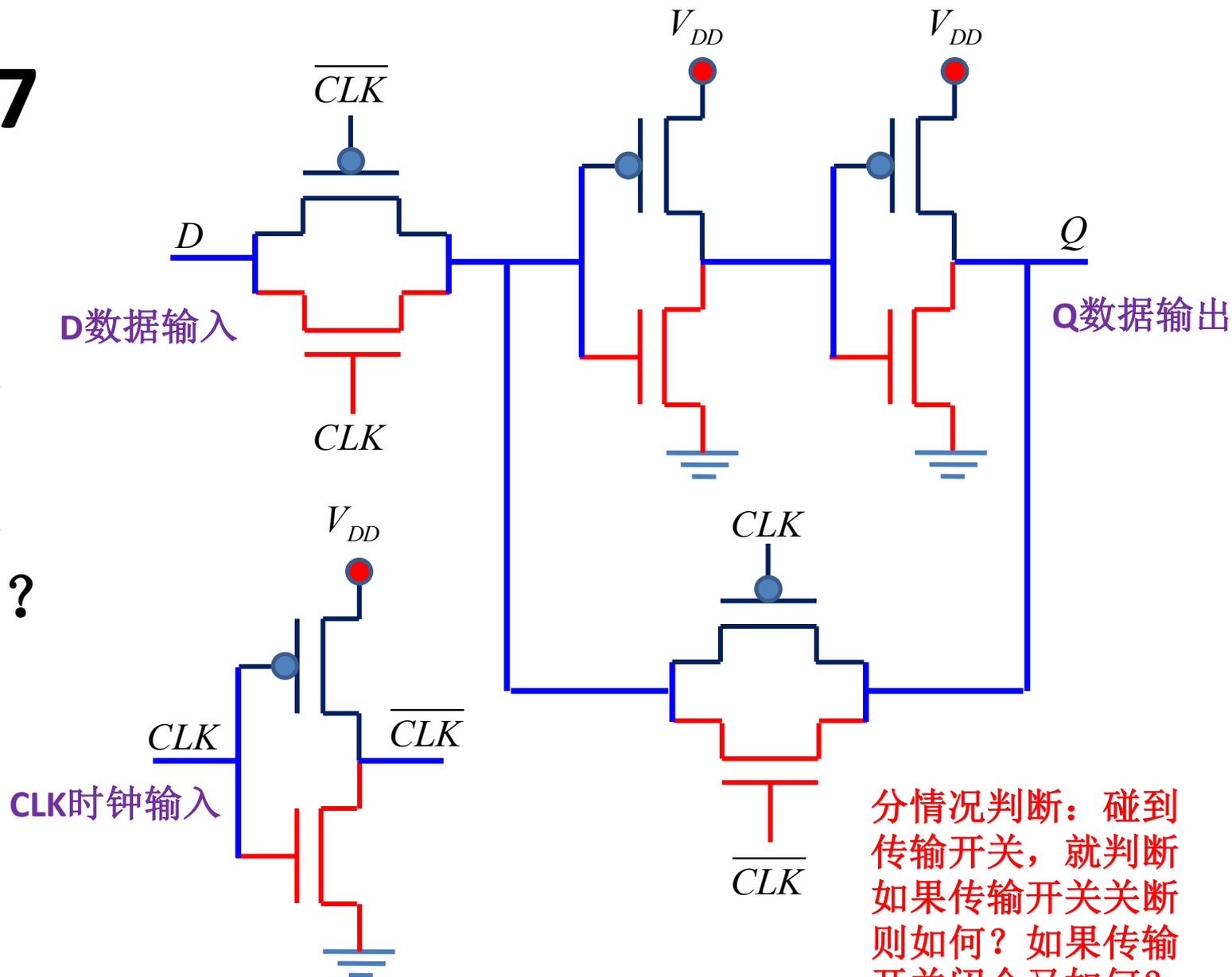
作业6：逻辑运算分析

- 请分析如图所示电路实现的是什么逻辑运算？
 - 给出详尽的分析过程



作业7

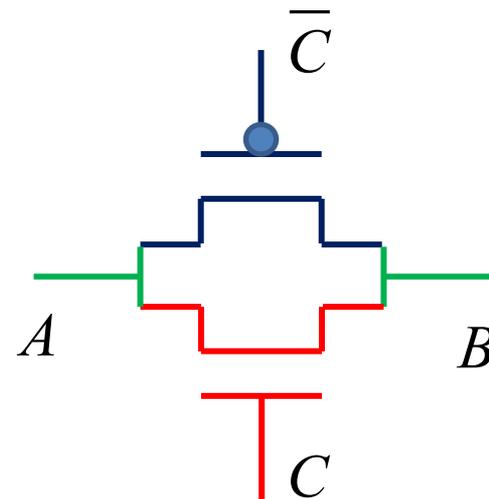
- 尝试分析如图所示数字电路实现什么功能？
 - 给出详尽的分析过程



分情况判断：碰到传输开关，就判断如果传输开关关断则如何？如果传输开关闭合又如何？

CAD仿真作业

- 仿真确认CMOS传输开关比单独NMOS/PMOS晶体管开关更优
 - 自选晶体管参量
 - 研究开关闭合时的静态传输情况
 - 激励源为方波恒压源，在逻辑1电压和逻辑0电压之间变化
 - 负载分两种：电阻负载和电容负载
 - 研究负载波形与激励源波形，分析差异原因
 - 调换激励源和负载位置，说明双向开关的优越性
 - 研究开关的动态特性
 - 激励源分别为逻辑1电平恒压源和正弦波恒压源两种情况
 - 开关控制信号为逻辑1电平和逻辑0电平之间转换的方波信号
 - 研究开关的动态特性
 - 研究开关控制信号的频率多大时，可确保信号的有效传输



PMOS的衬底始终接最高电平
NMOS的衬底始终接最低电平

三种开关：

- 1、单NMOS开关
- 2、单PMOS开关
- 3、CMOS传输开关

两种负载：

- 1、电阻负载
- 2、电容负载