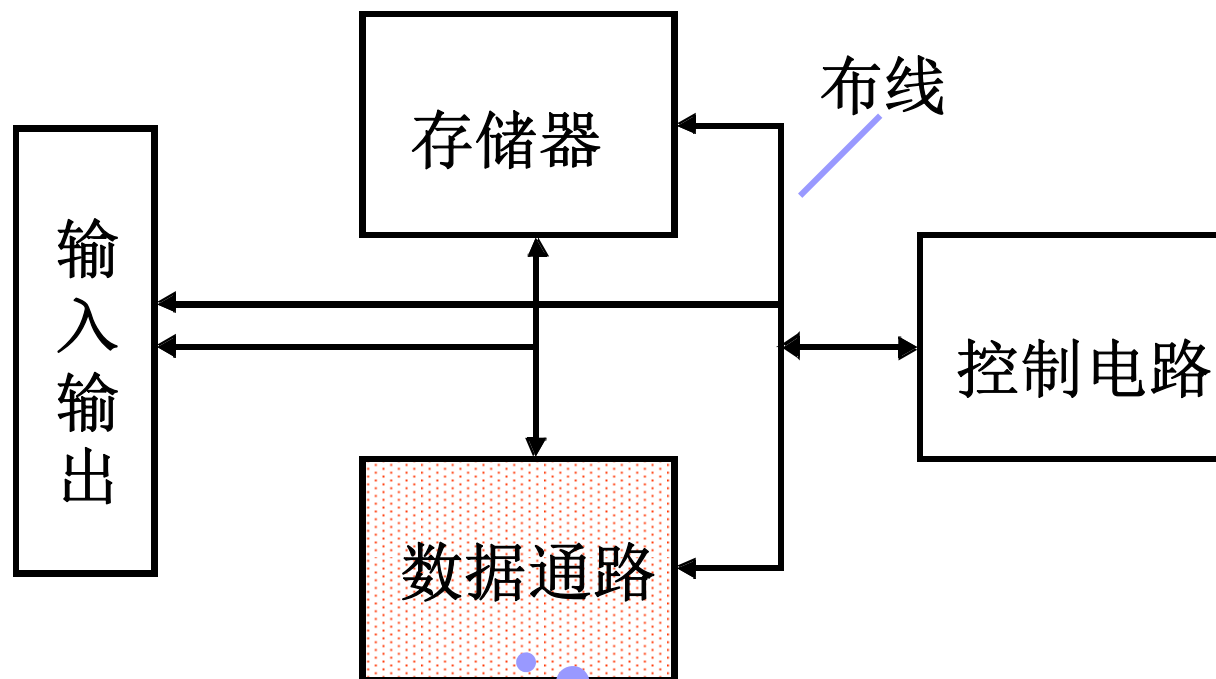


半导体 集成电路

学校：西安理工大学
院系：自动化学院电子工程系
专业：电子、微电
时间：秋季学期

第10章MOS逻辑功能部件

一般的数字处理器



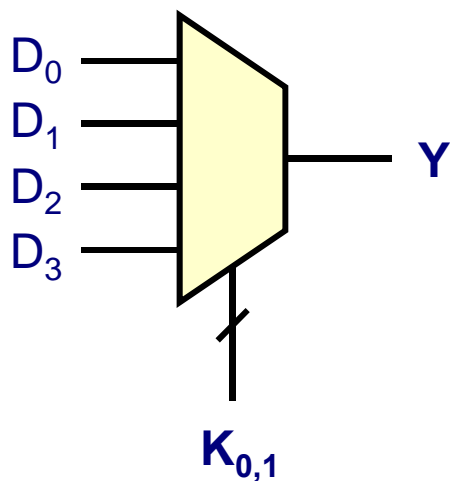
进行数据计算（包括算术运算和逻辑运算）



内容提要

- 多路开关
- 加法器和进位链
- 算术逻辑单元
- 移位器
- 乘法器

一、多路开关

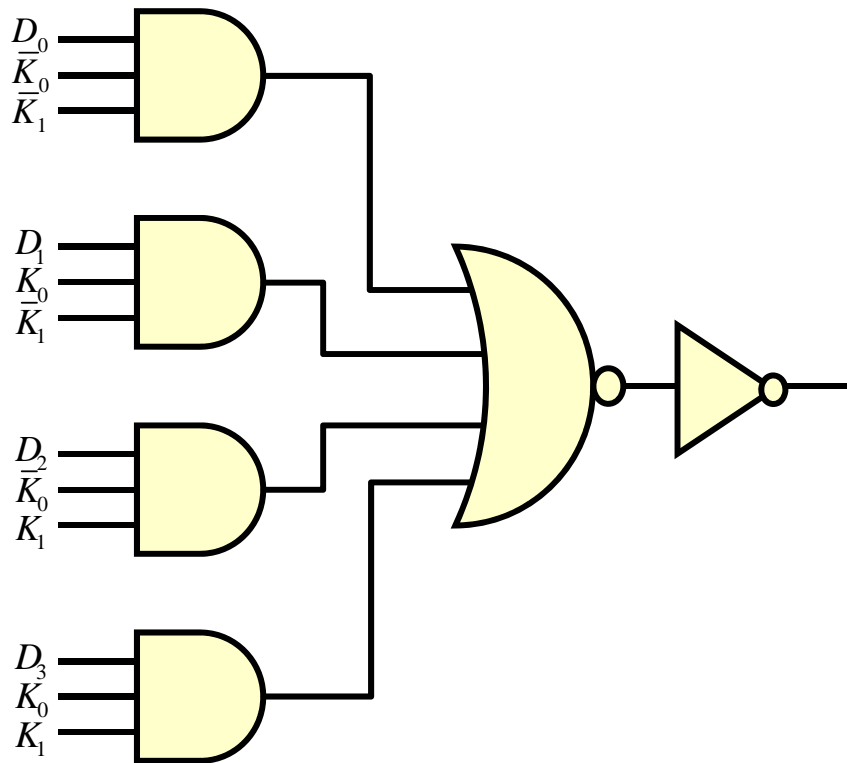


K_1	K_0	Y
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

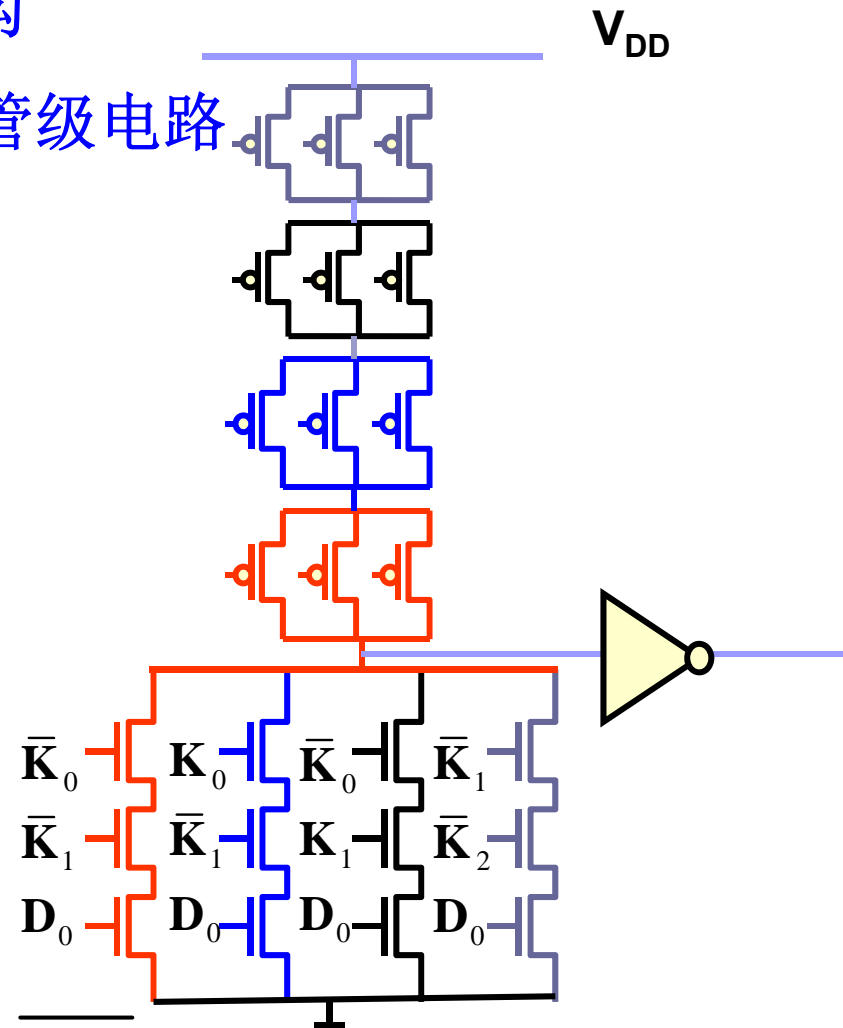
$$Y = \overline{K_0} \overline{K_1} D_0 + K_0 \overline{K_1} D_1 + \overline{K_0} K_1 D_2 + K_0 K_1 D_3$$

1. CMOS静态组合逻辑门电路结构

门级电路



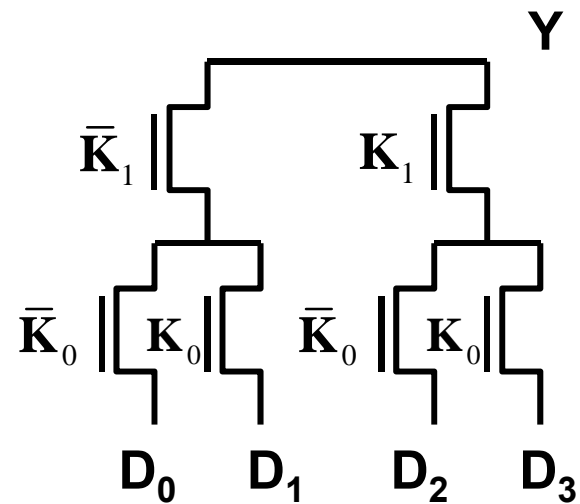
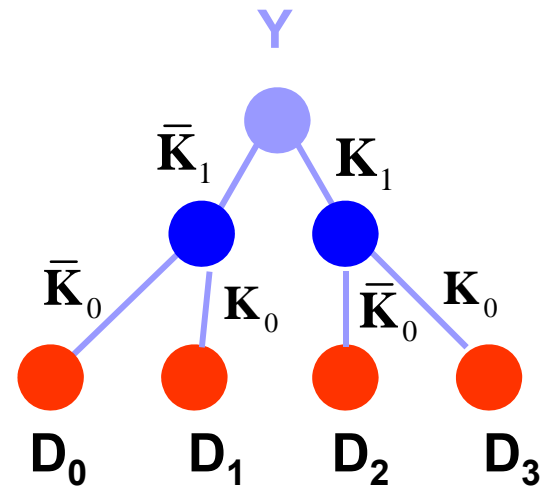
晶体管级电路



$$Y = \overline{K_0} \overline{K_1} D_0 + K_0 \overline{K_1} D_1 + \overline{K_0} K_1 D_2 + K_0 K_1 D_3$$

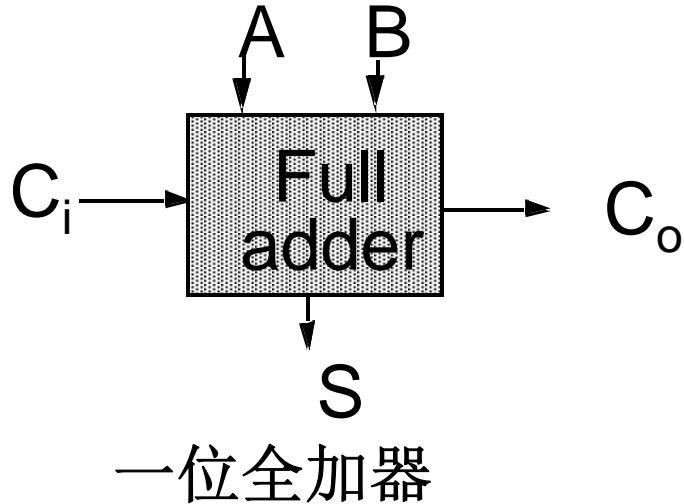
2. 传输门电路结构

K_1	K_0	Y
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3



二、加法器和进位链

1. 定义



A	B	C_i	S	C_o
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$S = A \oplus B \oplus C_i$$

$$= \overline{A}\overline{B}C_i + \overline{A}B\overline{C}_i + A\overline{B}\overline{C}_i + ABC_i$$

$$C_o = AB + BC_i + AC_i$$

令

$$G = AB$$

$$P = A \oplus B$$

$$D = \overline{AB}$$



进位产生信号

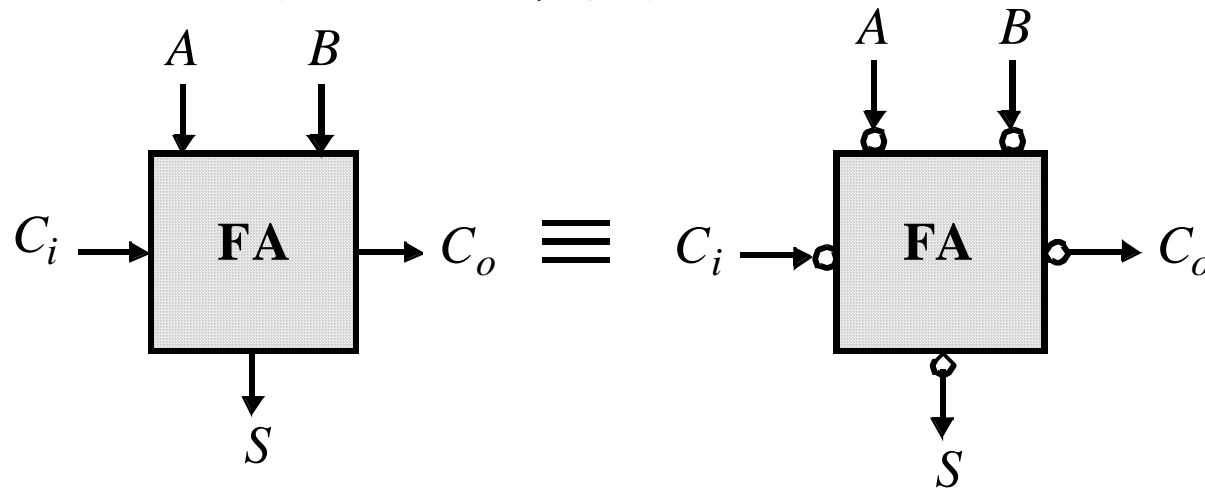
进位传输信号

进位取消信号

$$C_o(G, P) = G + PC_i$$

$$S(G, P) = P \oplus C_i$$

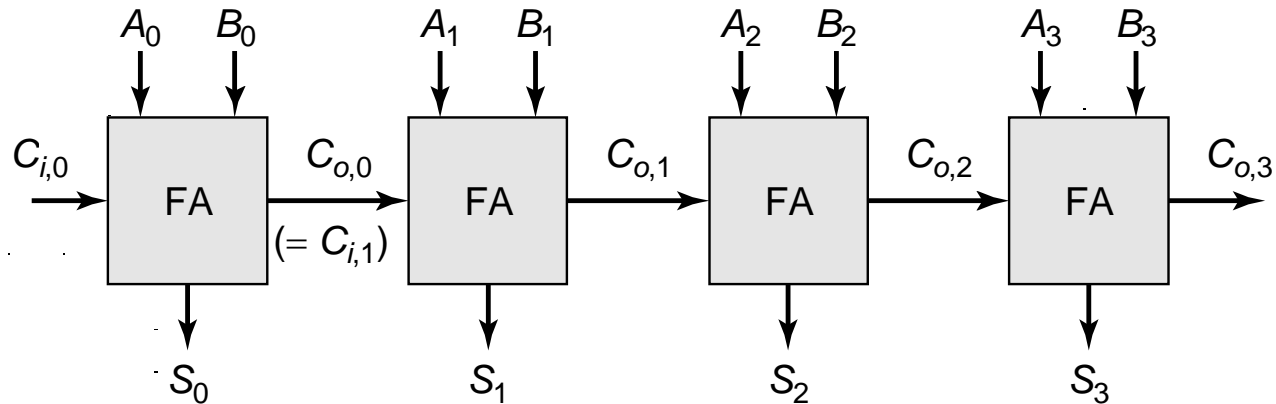
全加器的反相特性



$$\bar{S}(A, B, C_i) = S(\bar{A}, \bar{B}, \bar{C}_i) \quad \bar{A} \oplus \bar{B} \oplus \bar{C}_i = \overline{A \oplus B \oplus C_i} = \bar{S}$$

$$\bar{C}_o(A, B, C_i) = C_o(\bar{A}, \bar{B}, \bar{C}_i) \quad \overline{AB + AC_i + BC_i} = \bar{C}_o$$

逐位进位加法器



最大时延 $t_d = O(N)$

$$t_{adder} = (N-1)t_{carry} + t_{sum}$$

结论:

1. 逐位进位加法器的传播延时与N成线性关系
2. 优化逐位进位加法器的全加器单元时，优化“进位延时”比“和延时”重要

2. 全加器电路设计

(1) 互补静态CMOS组合逻辑电路

变换思路：在不减慢进位产生速度的前提下，让“和”与“进位”产生的子电路之间共享某些逻辑来减少晶体

$$C_o = AB + BC_i + AC_i \quad \text{管数目}$$

$$\begin{aligned} S &= A \oplus B \oplus C_i \\ &= \overline{A} \overline{B} C_i + \overline{A} B \overline{C}_i + A \overline{B} \overline{C}_i + ABC_i \end{aligned}$$

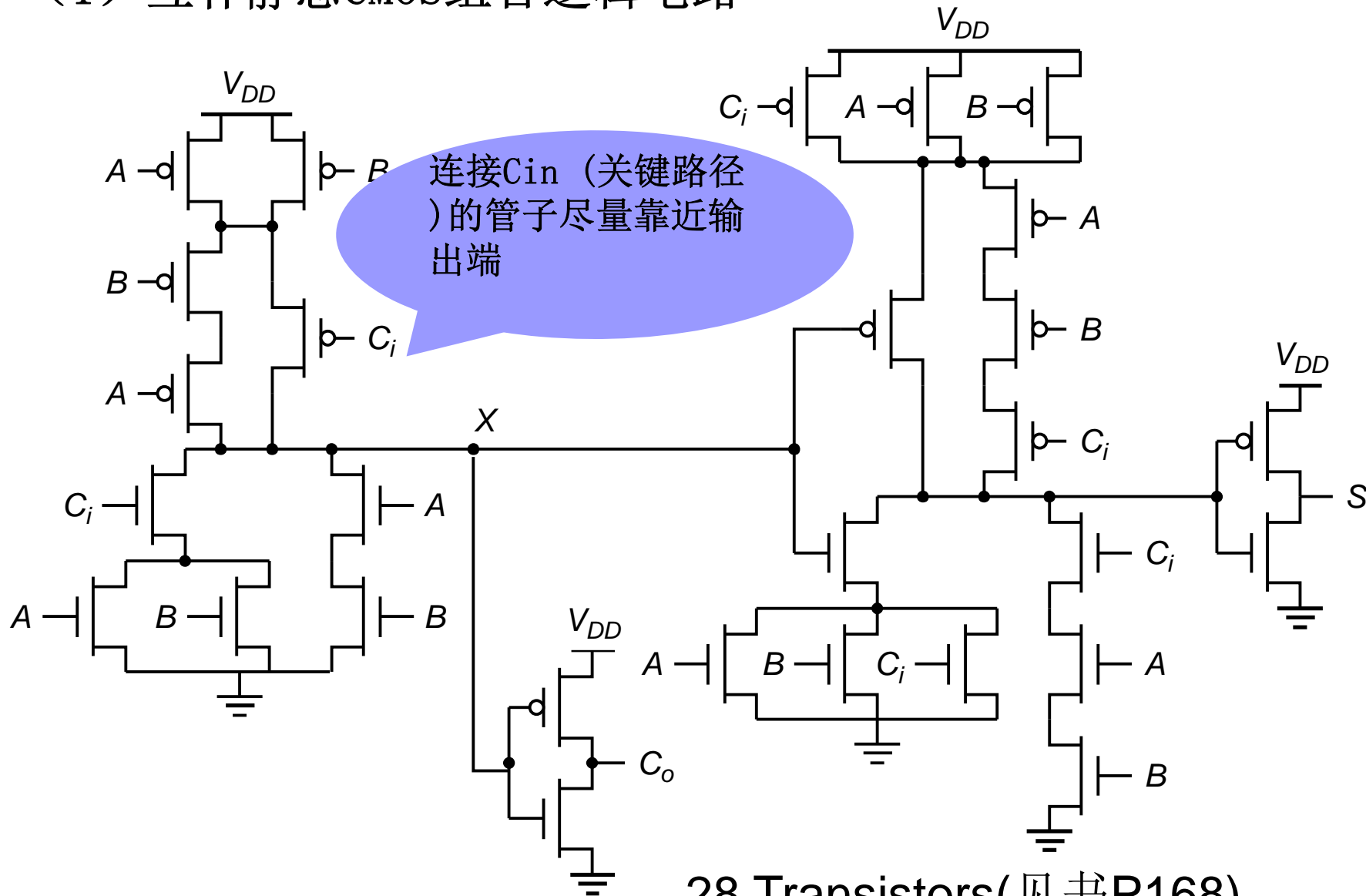


$$S = ABC_i + \overline{C}_o (A + B + C_i)$$

$$C_o = AB + BC_i + AC_i$$

$$S = \overline{C_o}(A + B + C_i) + ABC_i$$

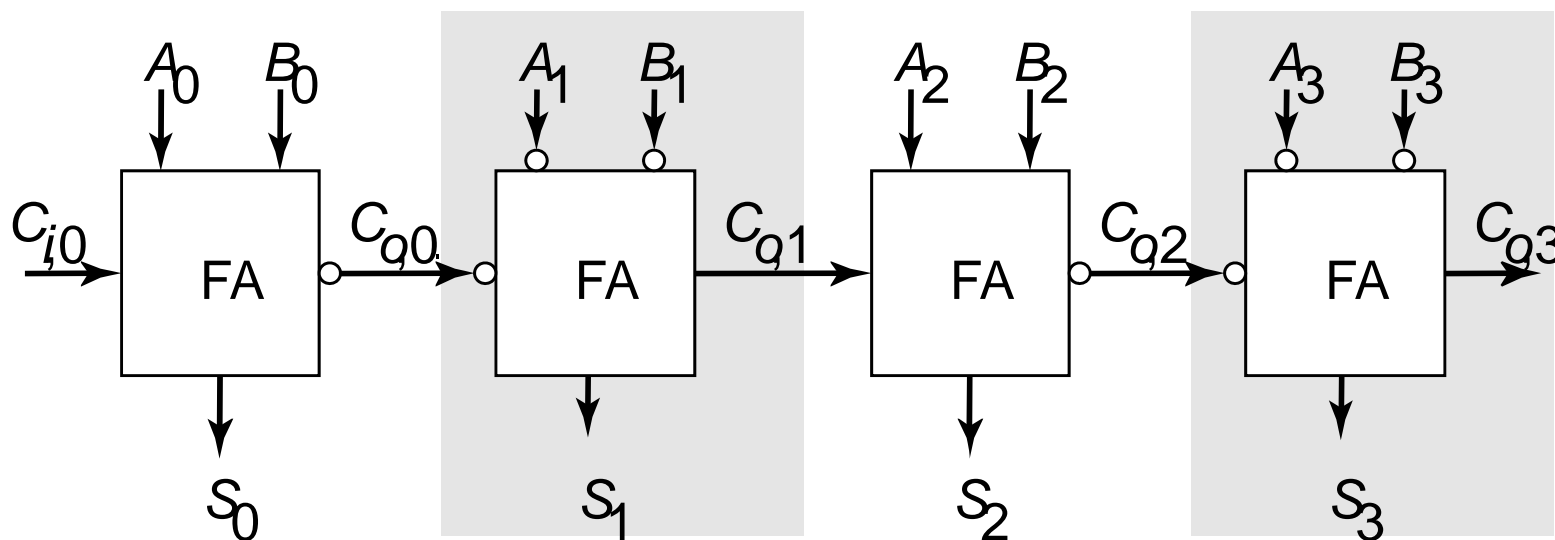
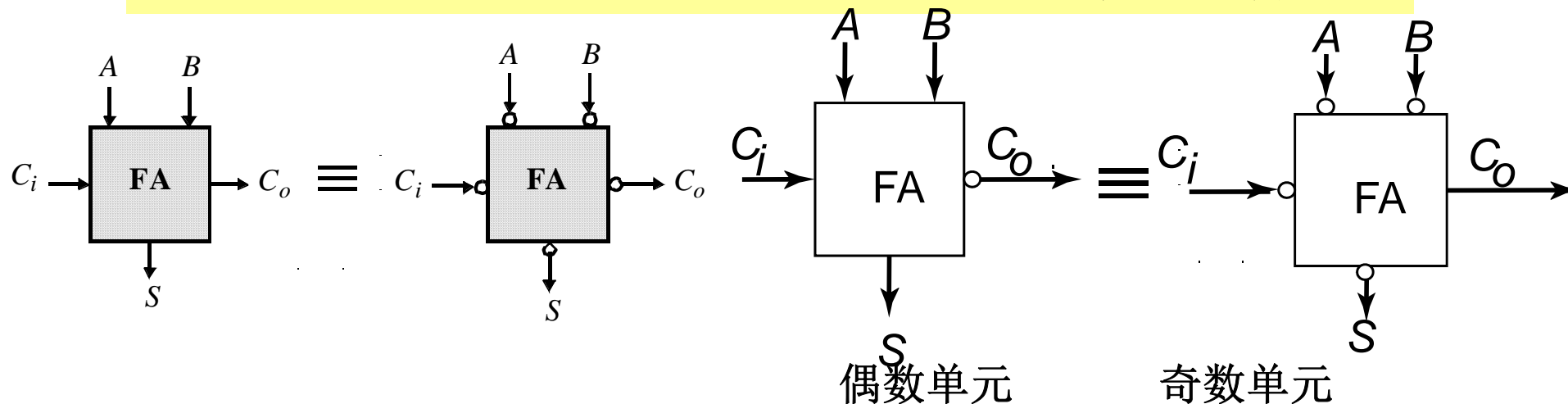
(1) 互补静态CMOS组合逻辑电路



28 Transistors(见书P168)

逐位进位加法器优化目标:使进位通路延迟最小

进位链上的反相器可以利用加法器的反相特性消除

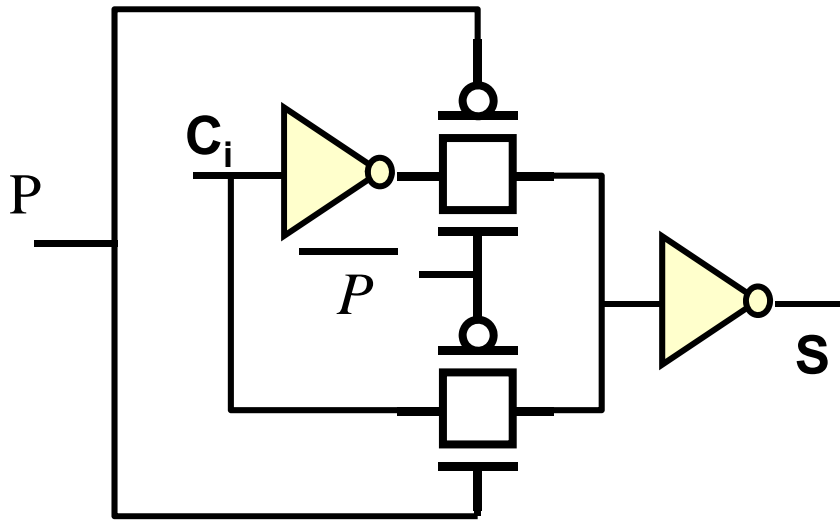


(2) 传输门加法器

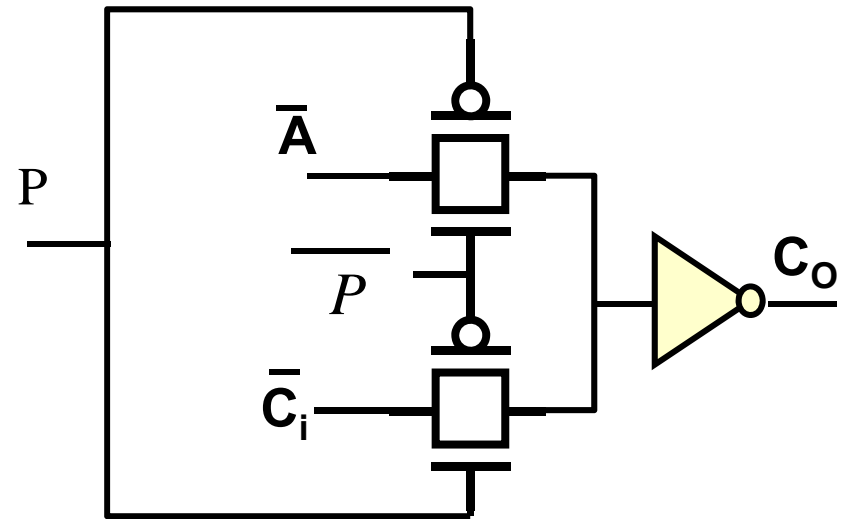
$$S(G, P) = P \oplus C_i$$

$$C_o(G, P) = G + PC_i$$

其中: $G=AB$ $P= A \oplus B$



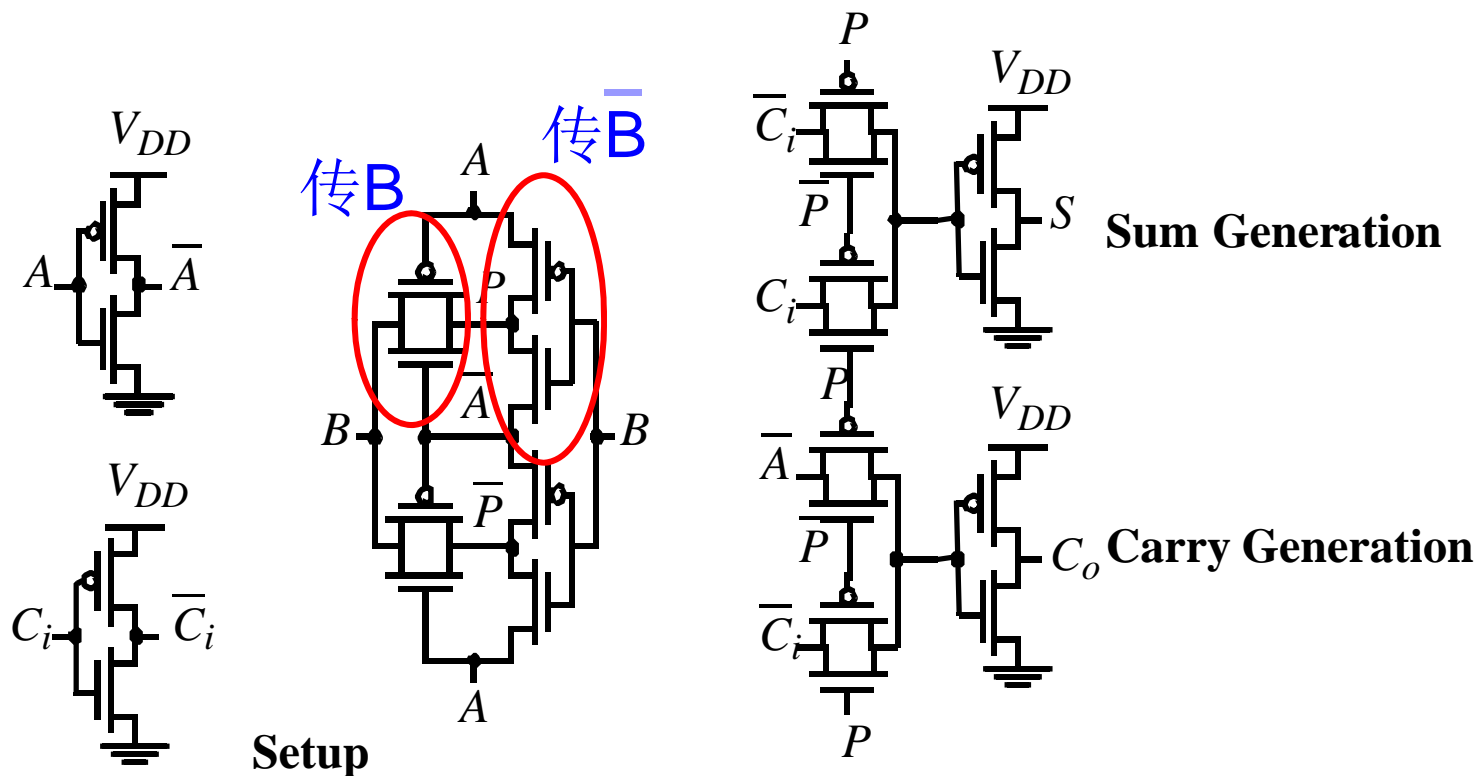
P 为1传 \bar{C}_i , P 为0传 C_i



P 为1传 C_i , P 为0传A或B

门级电路

(2) 传输门加法器



$$P = A \oplus B$$

$$\bar{P} = \overline{A \oplus B}$$

24 Transistors(见书P170)

3.超前进位加法器

采用提前进位办法（**CARRY LOOKAHEAD**）

令

$$G_i = A_i B_i$$

进位产生信号

$$P_i = A_i \oplus B_i$$

进位传输信号

$$C_i = G_i + P_i C_{i-1}$$

$$C_1 = G_1 + P_1 C_0$$

$$C_2 = G_2 + P_2 G_1 + P_2 P_1 C_0$$

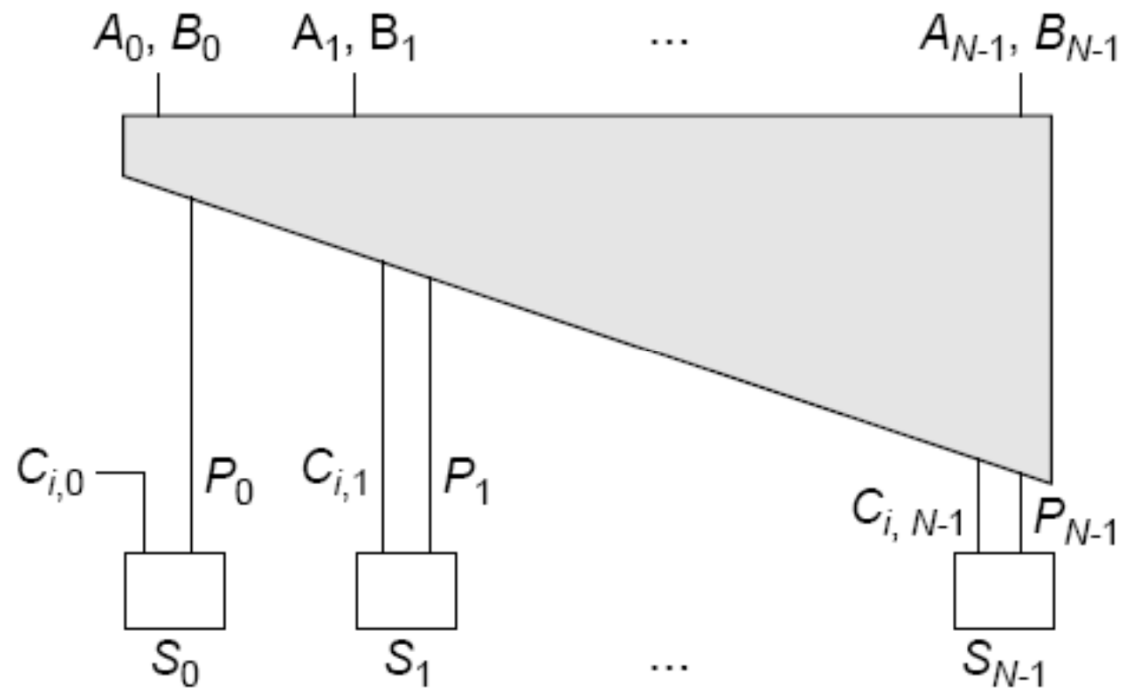
$$C_3 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 C_0$$

$$C_4 = G_4 + P_4 G_3 + P_4 P_3 G_2 + P_4 P_3 P_2 G_1 + P_4 P_3 P_2 P_1 C_0$$

$$S_i = A_i \oplus B_i \oplus C_{i-1} = P_i \oplus C_{i-1}$$

任何一位的进位输出只由本级和前级的输入信号组成而不必等待逐级传输

超前进位加法器原理框图



超前进位电路图见书P172

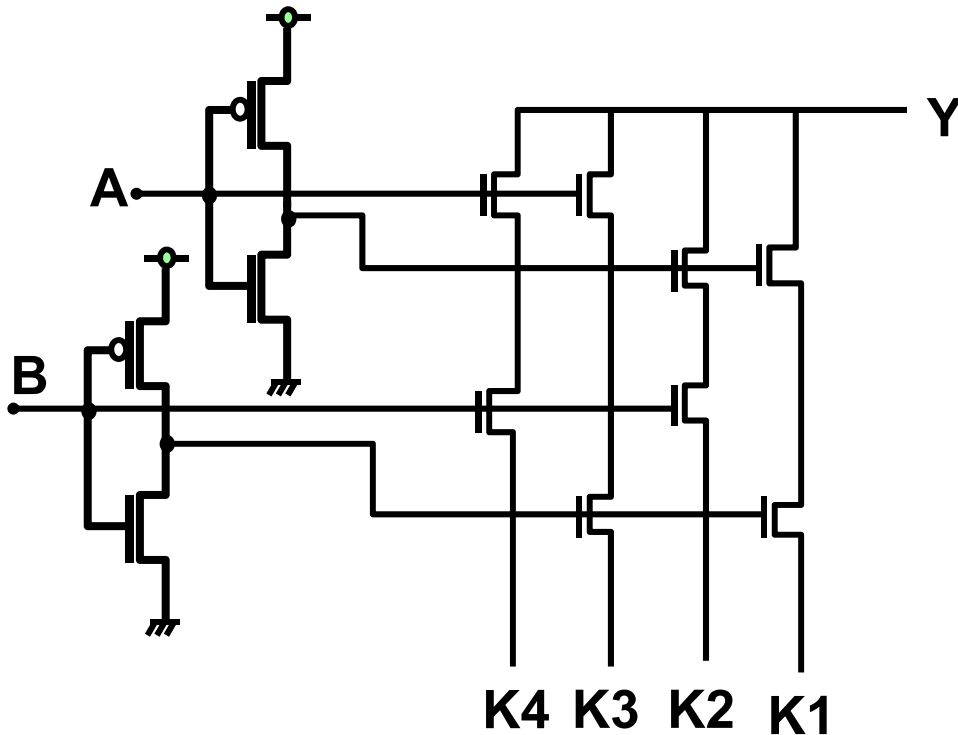


超前进位加法器结论

- 各门的输入端数一般小于等于4
- 位数较多时，四级与四级之间采用逐位加法

三、算术逻辑单元 (ALU)

既能进行算术运算，又能进行逻辑运算的单元

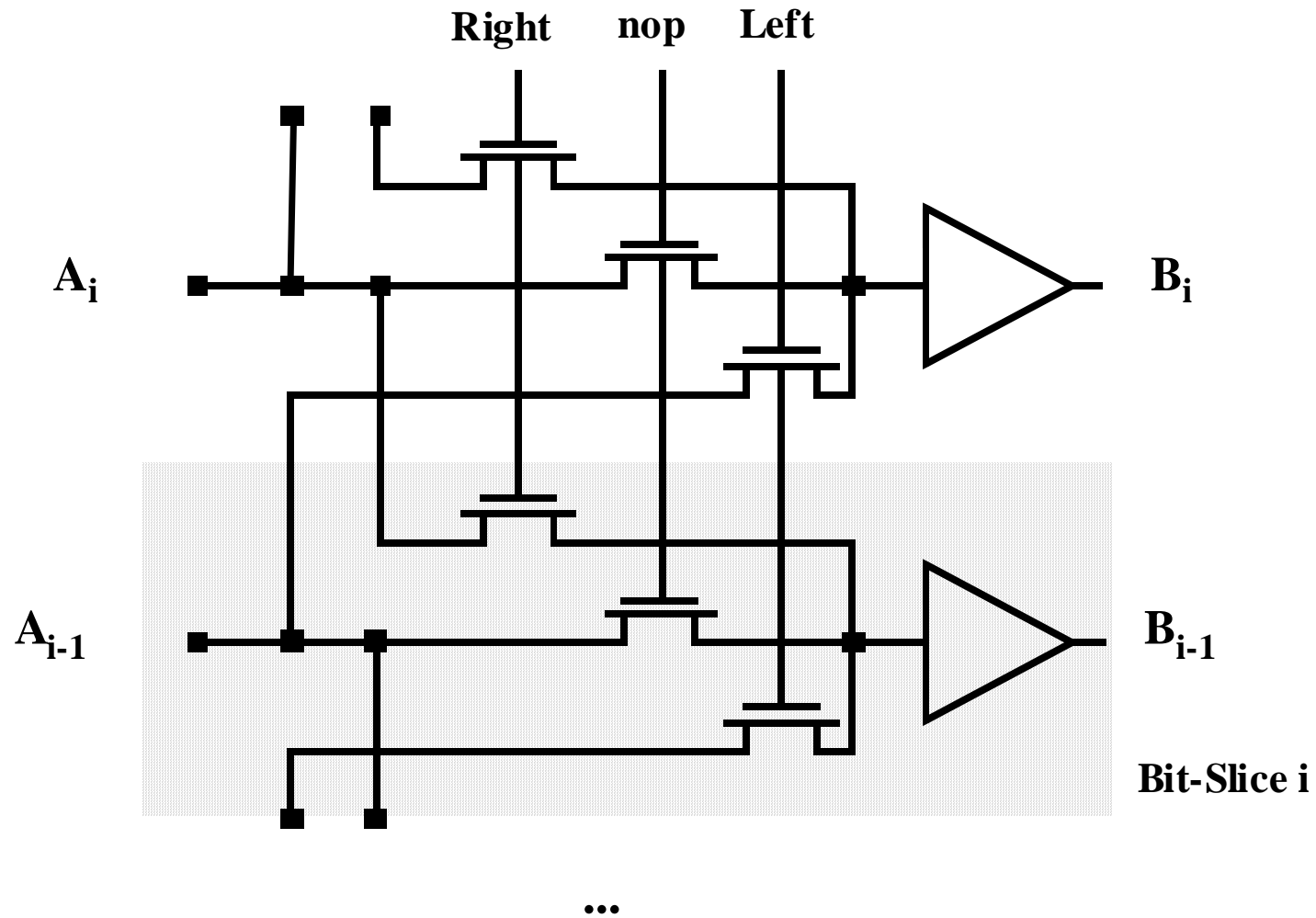


$$Y = ABK_4 + \bar{A}\bar{B}K_3 + \bar{A}BK_2 + \bar{A}\bar{B}K_1$$

K4	K3	K2	K1	Y
0	0	0	0	0
0	0	0	1	$\bar{A} \cdot \bar{B}$
0	0	1	0	$\bar{A} \cdot B$
0	0	1	1	\bar{A}
0	1	0	0	$A \cdot \bar{B}$
0	1	0	1	\bar{B}
0	1	1	0	$A \oplus B$
0	1	1	1	$\bar{A} + \bar{B}$
1	0	0	0	$A \cdot B$
1	0	0	1	$A \oplus B$
1	0	1	0	B
1	0	1	1	$\bar{A} + B$
1	1	0	0	A
1	1	0	1	$A + \bar{B}$
1	1	1	0	$A + B$
1	1	1	1	1

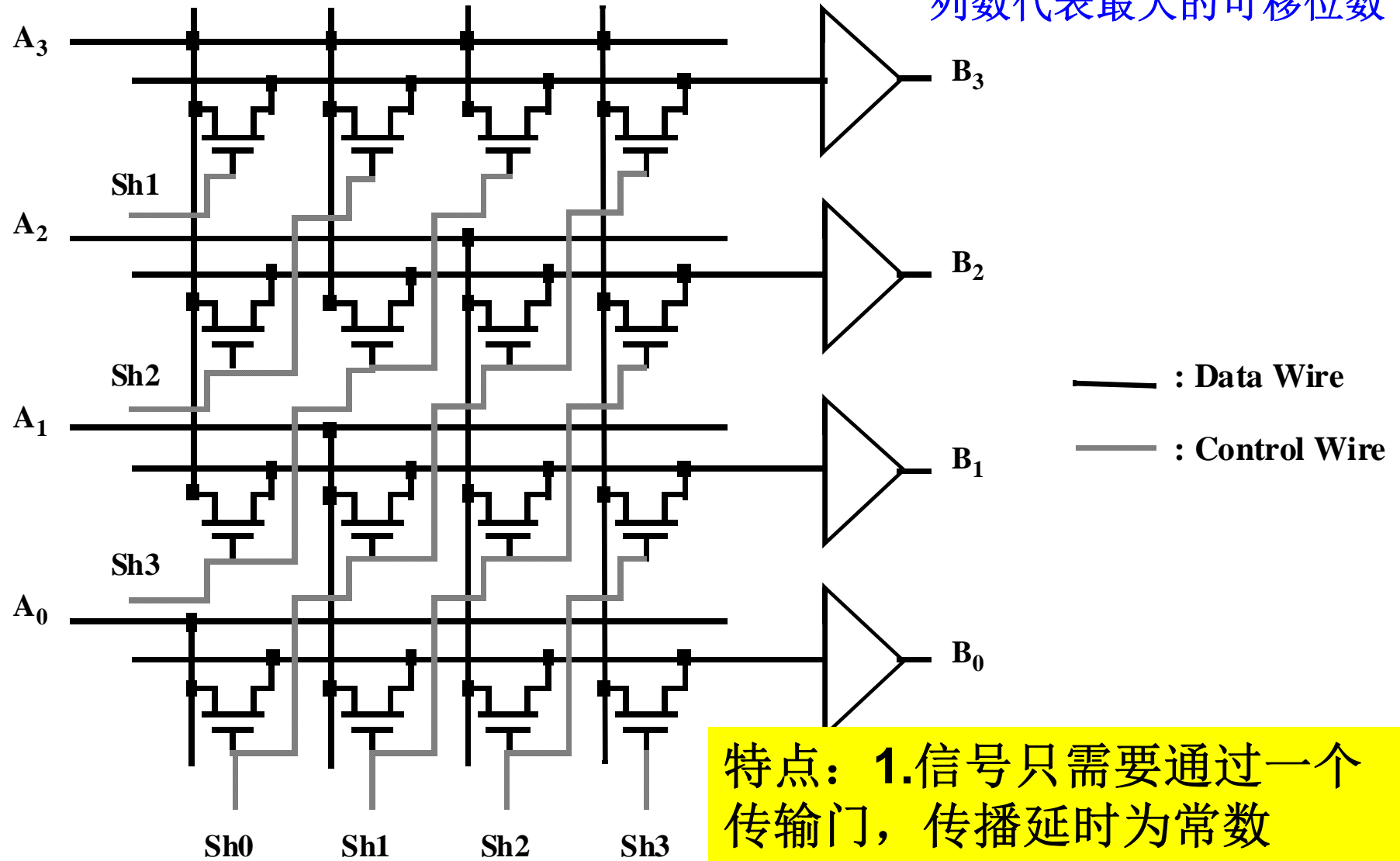
四、移位器

1. 一位可编程移位器



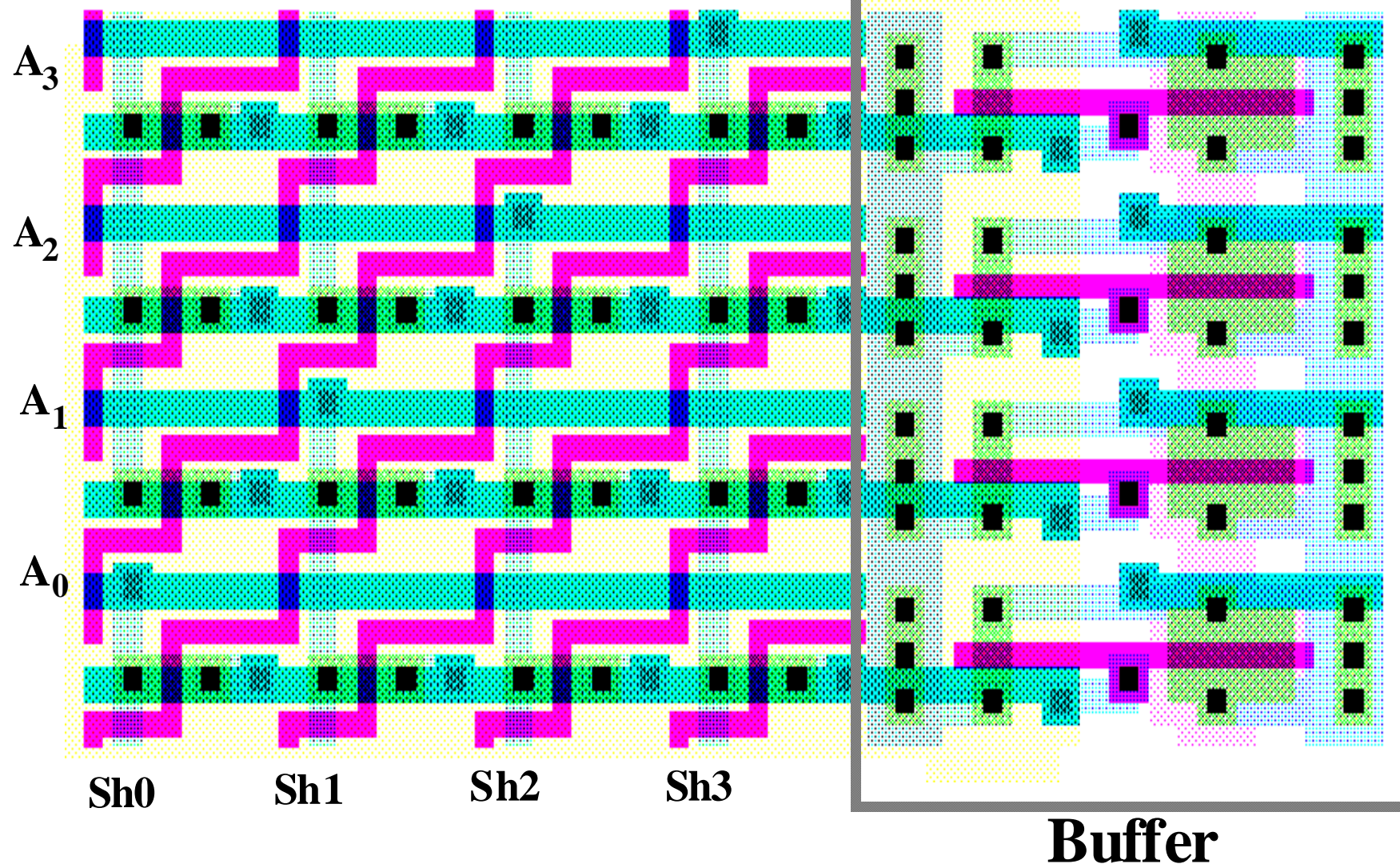
2. 桶式移位器(右移, 符号位A3自动复制)

行数代表字长
列数代表最大的可移位数



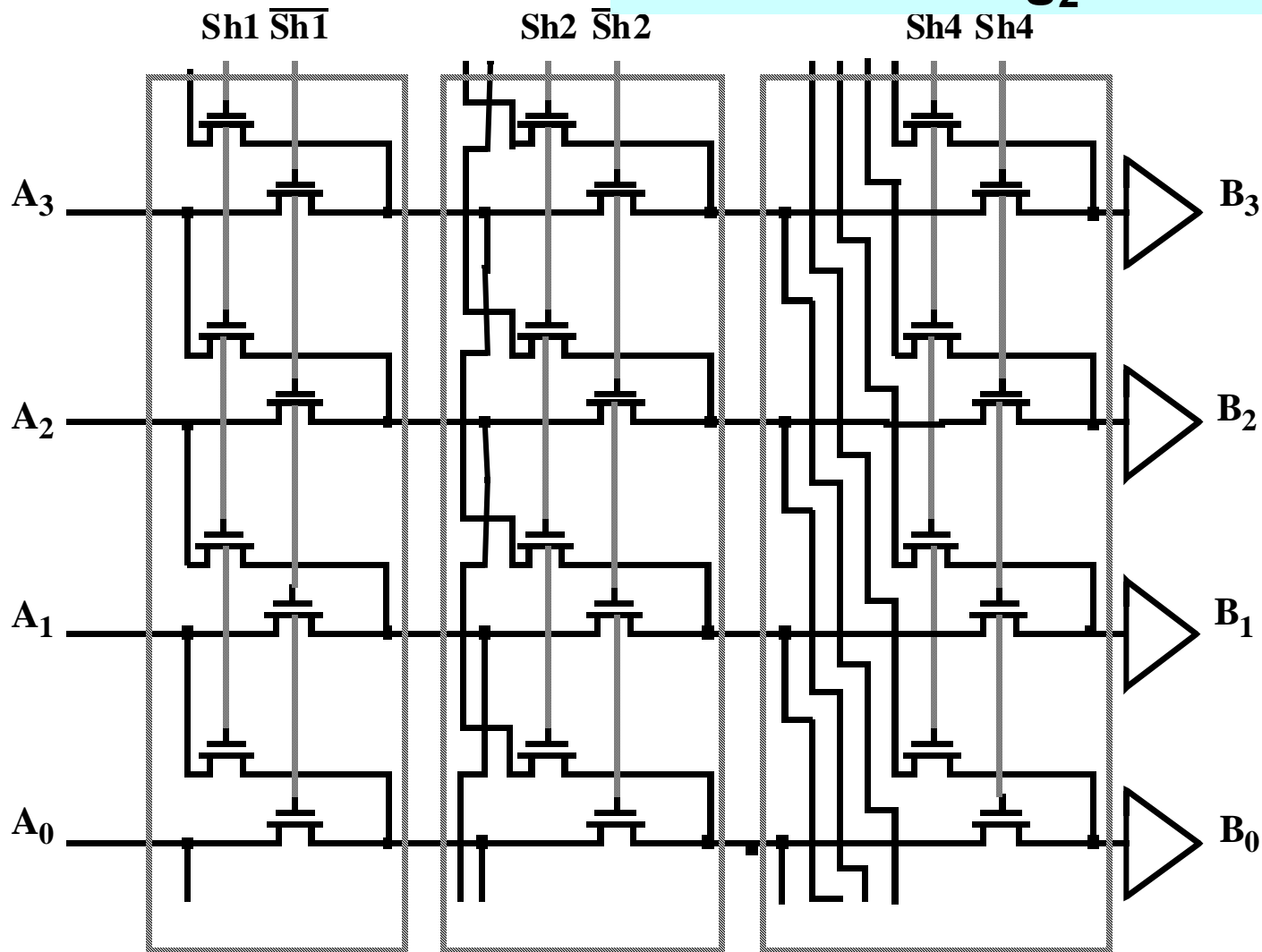
特点: 1.信号只需要通过一个传输门, 传播延时为常数
2.面积主要被布线通道占据

桶式移位器版图

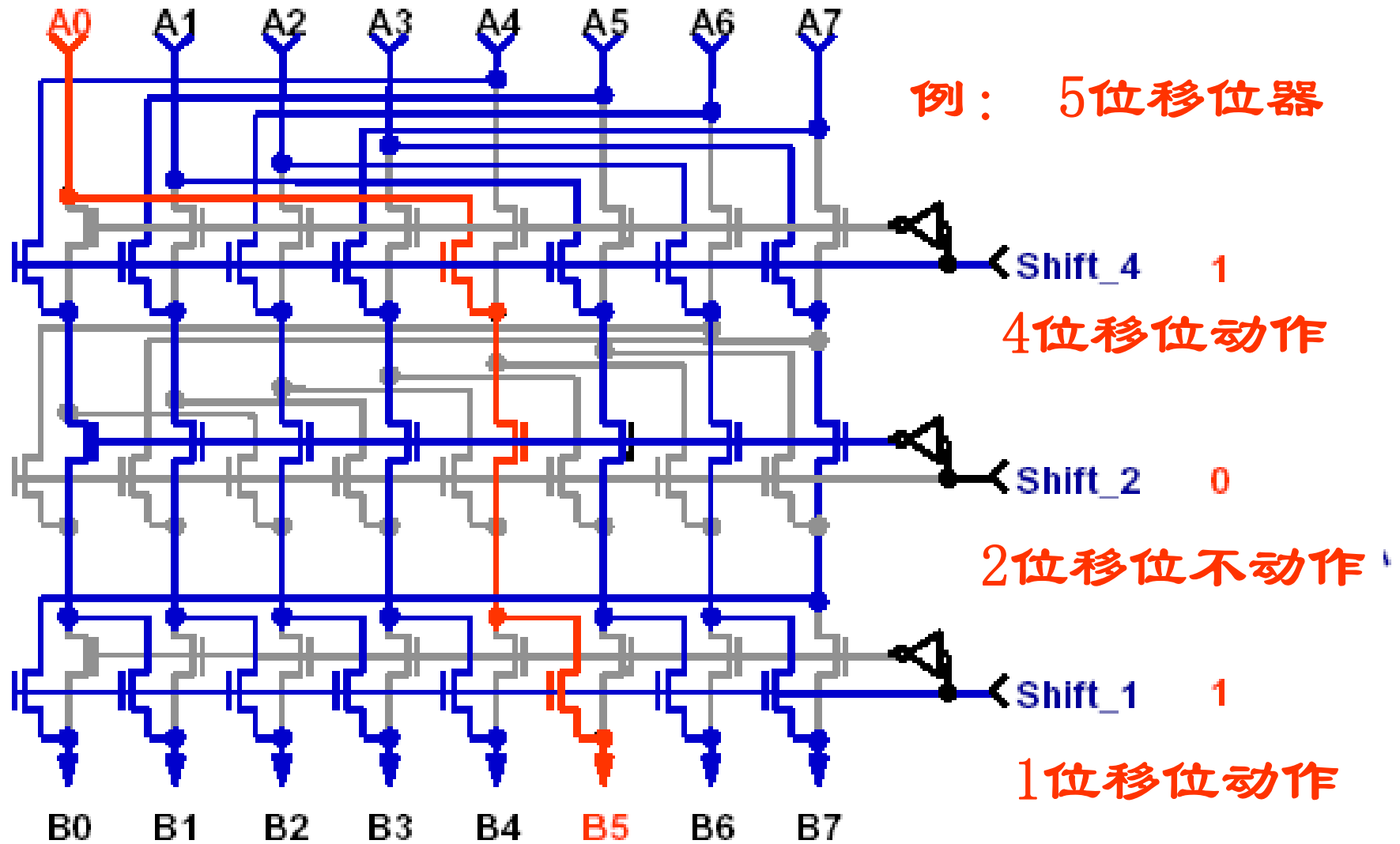


移位数控制为 2^i ，设最大移位宽度为 M 位，则移位级数为 $\log_2 M$ ，速度也取决于 $\log_2 M$

3. 对数移位器



对数移位器工作原理(左移)

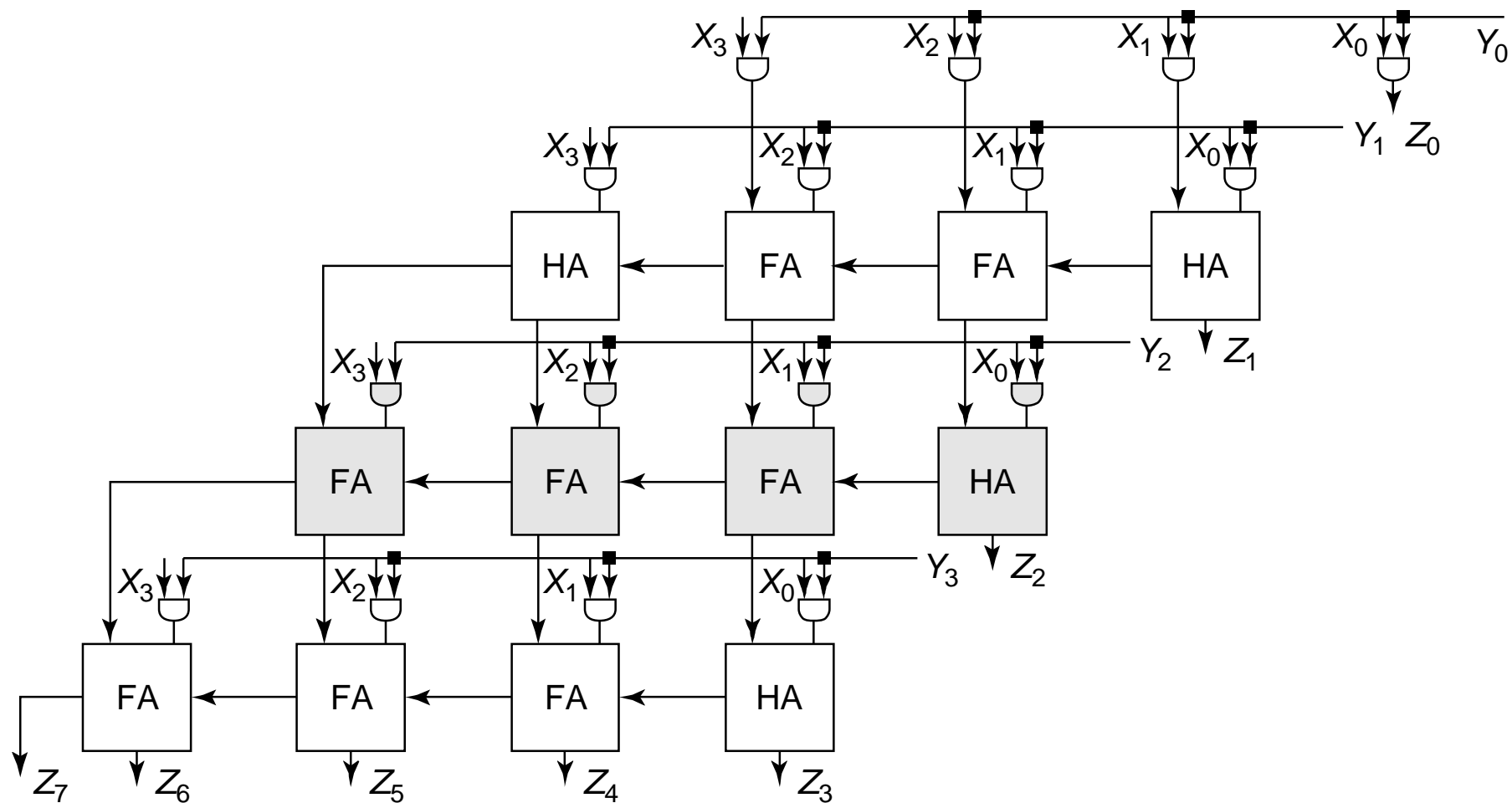




移位器结论

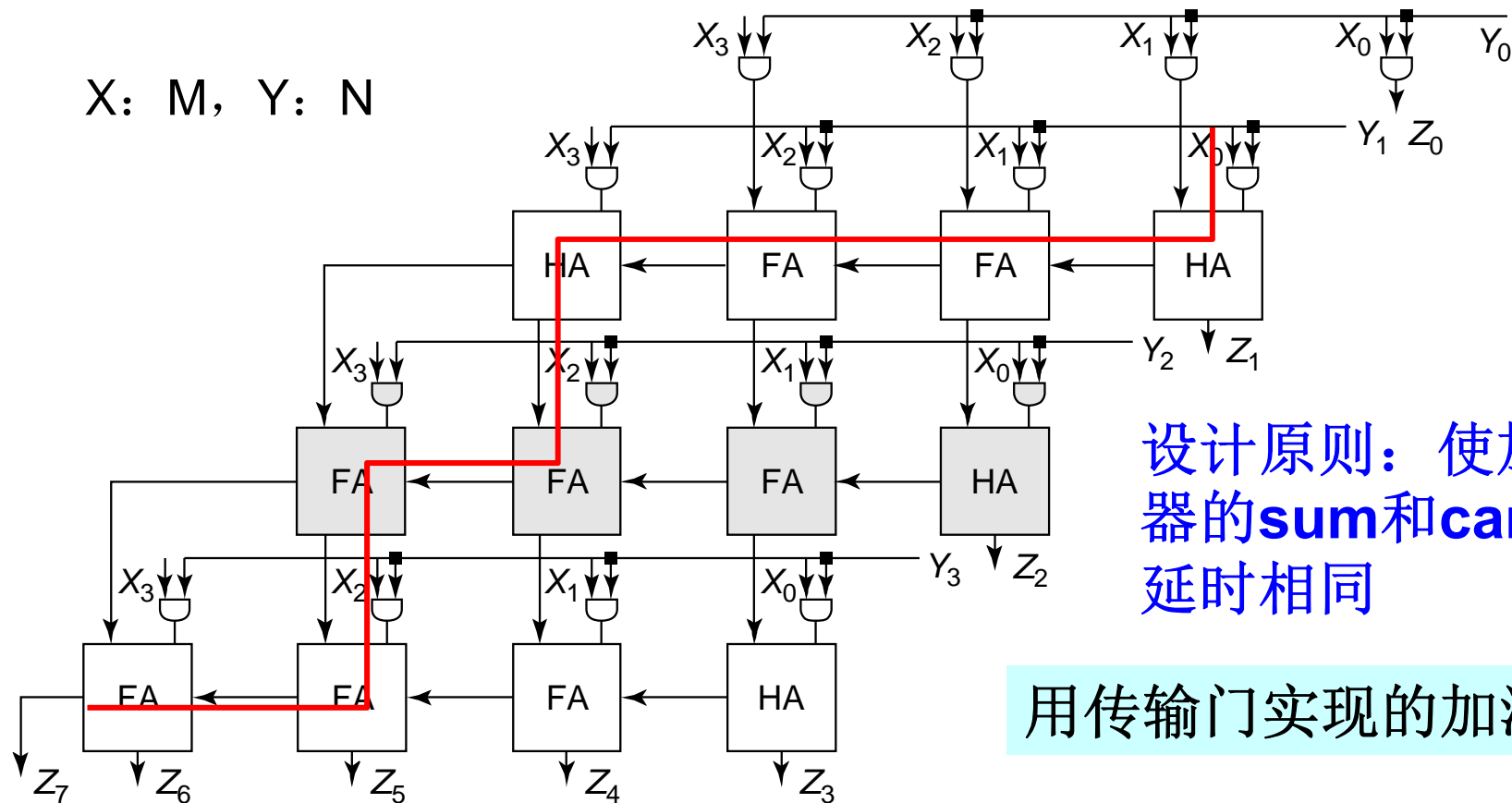
- 对数移位器无需编解码。
- 桶式移位器适合于小型移位器，对于较大规模的移位器，对数移位器更有效（级数比桶式的少）。
- 对数移位器的速度取决于 $\log_2 M$ （ M 为移位宽度），当传输门串联级数较多时，可插入缓冲器提高速度。

2. 并行阵列乘法器



MxN阵列乘法器的关键路径

X: M, Y: N



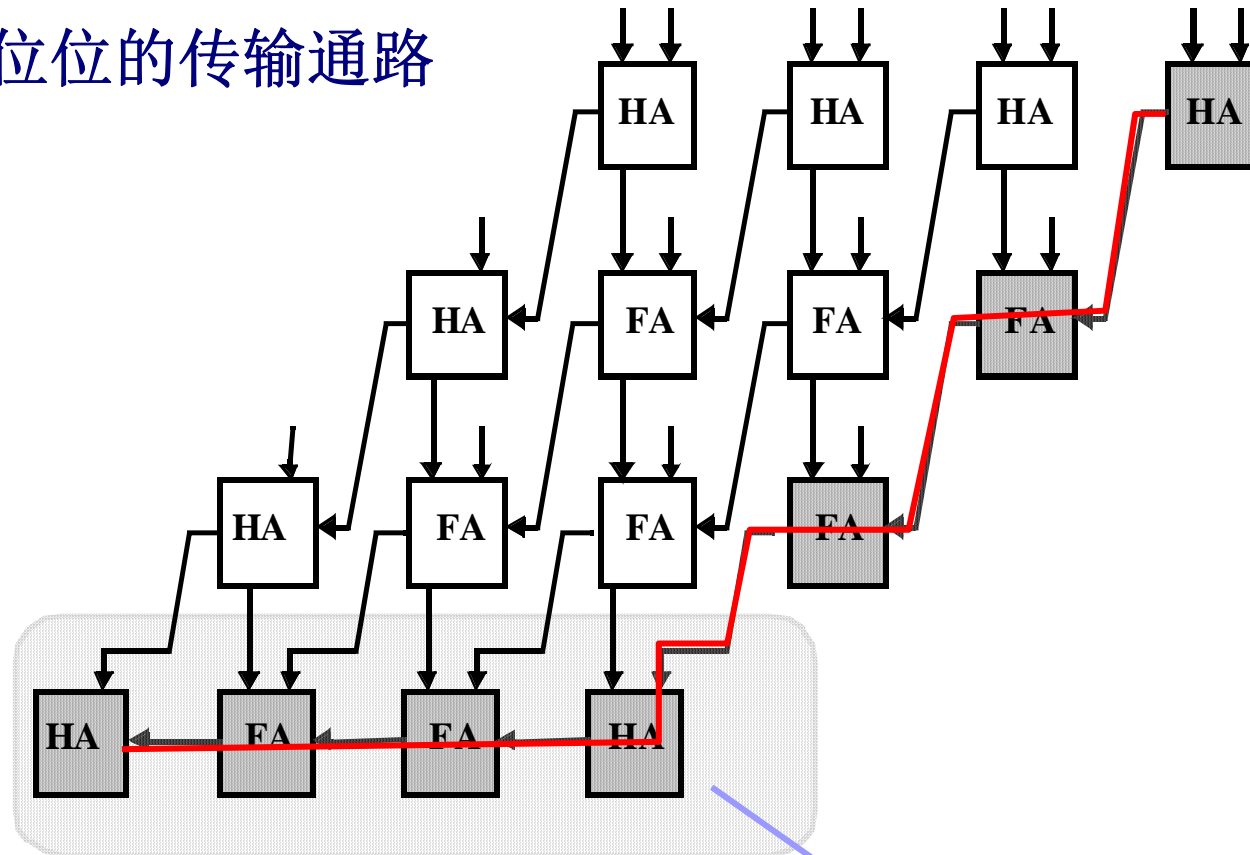
设计原则：使加法器的sum和carry延时相同

用传输门实现的加法器

$$t_{\text{mult}} \approx [(M-1) + (N-2)]t_{\text{carry}} + (N-1)t_{\text{sum}} + t_{\text{and}}$$

3. 保留进位乘法器

改变进位位的传输通路



Vector Merging Adder 向量合并加法器

$$T_{\text{mult}} \approx (N-1)t_{\text{carry}} + t_{\text{and}} + t_{\text{merge}}$$



作业：

P180 9.1