超大规模集成电路基础

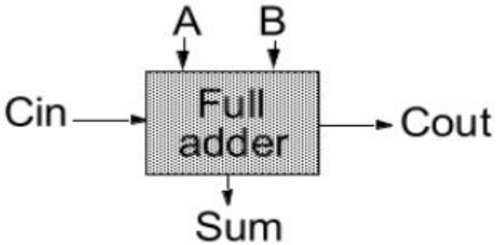
Fundamental of VLSI

第八章 功能设计

二进制加法器

- 时序电路的时间参数

$$S = A \oplus B \oplus C_i$$
$$= A\bar{B}\bar{C}_i + \bar{A}B\bar{C}_i + \bar{A}\bar{B}C_i + ABC_i$$
$$C_o = AB + BC_i + AC_i$$



A	B	C_i	S	C_o	*Carry status*
0	0	0	0	0	delete
0	0	1	1	0	delete
0	1	0	1	0	propagate
0	1	1	0	1	propagate
1	0	0	1	0	propagate
1	0	1	0	1	propagate
1	1	0	0	1	generate
1	1	1	1	1	generate

二进制加法器

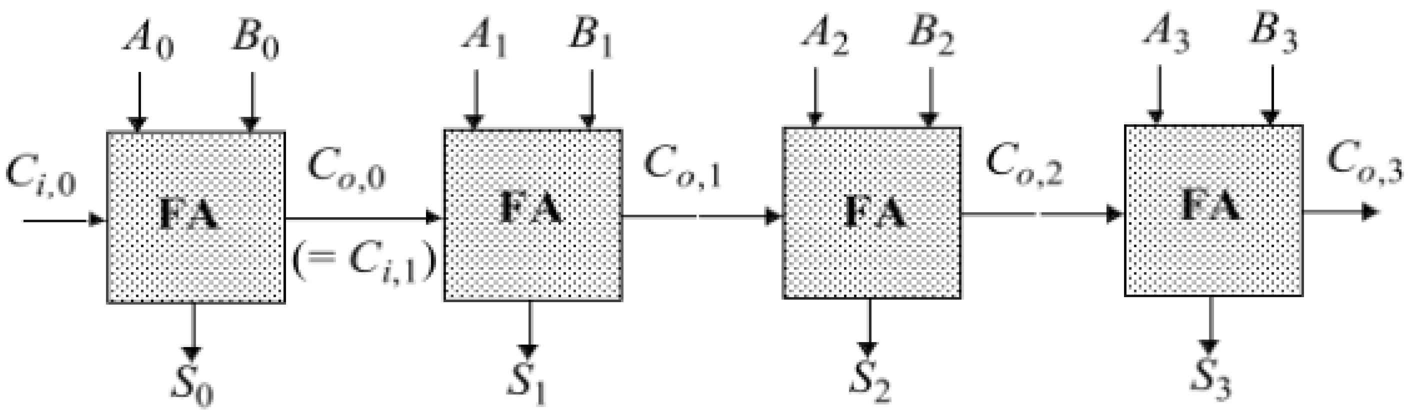
- 产生，取消，传播

$$G = AB$$
$$D = \overline{A}\,\overline{B}$$
$$P = A \oplus B$$

$$C_o(G,P) = G + PC_i$$
$$S(G,P) = P \oplus C_i$$

二进制加法器

- 行波进位加法器



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

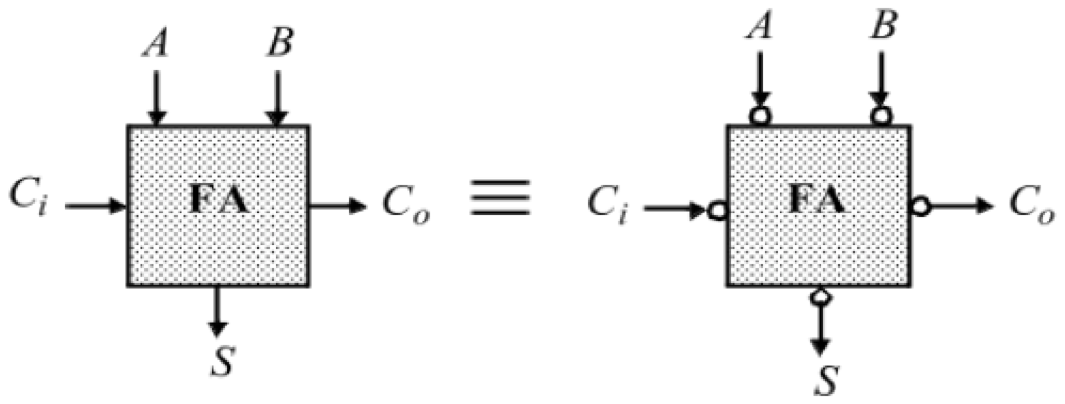
•逐位进位加法器的传播延时与N成线性关系
•加法器延时由进位延时决定

进位延迟最坏情况之一

$$A:00000001, \quad B:01111111$$

全加器电路设计考虑

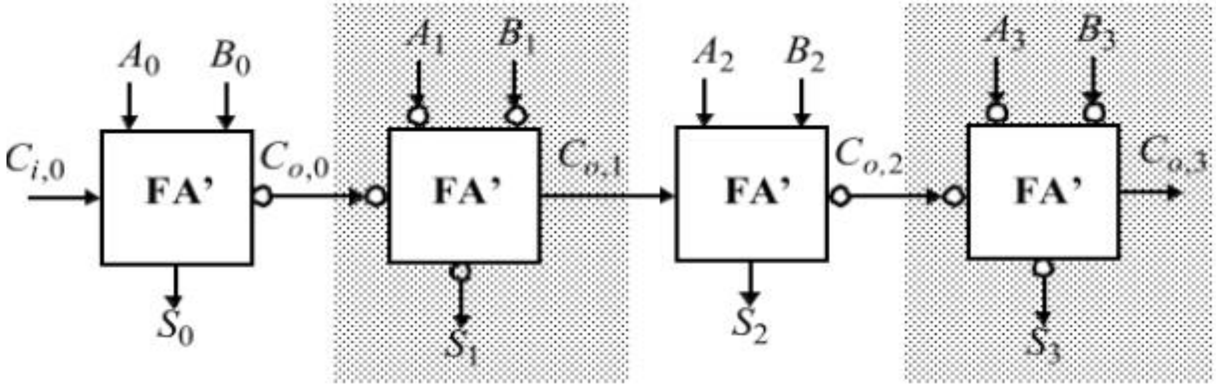
- 加法器的反向特性
  - 把加法器的所有输入反向可以得到反向的输出



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

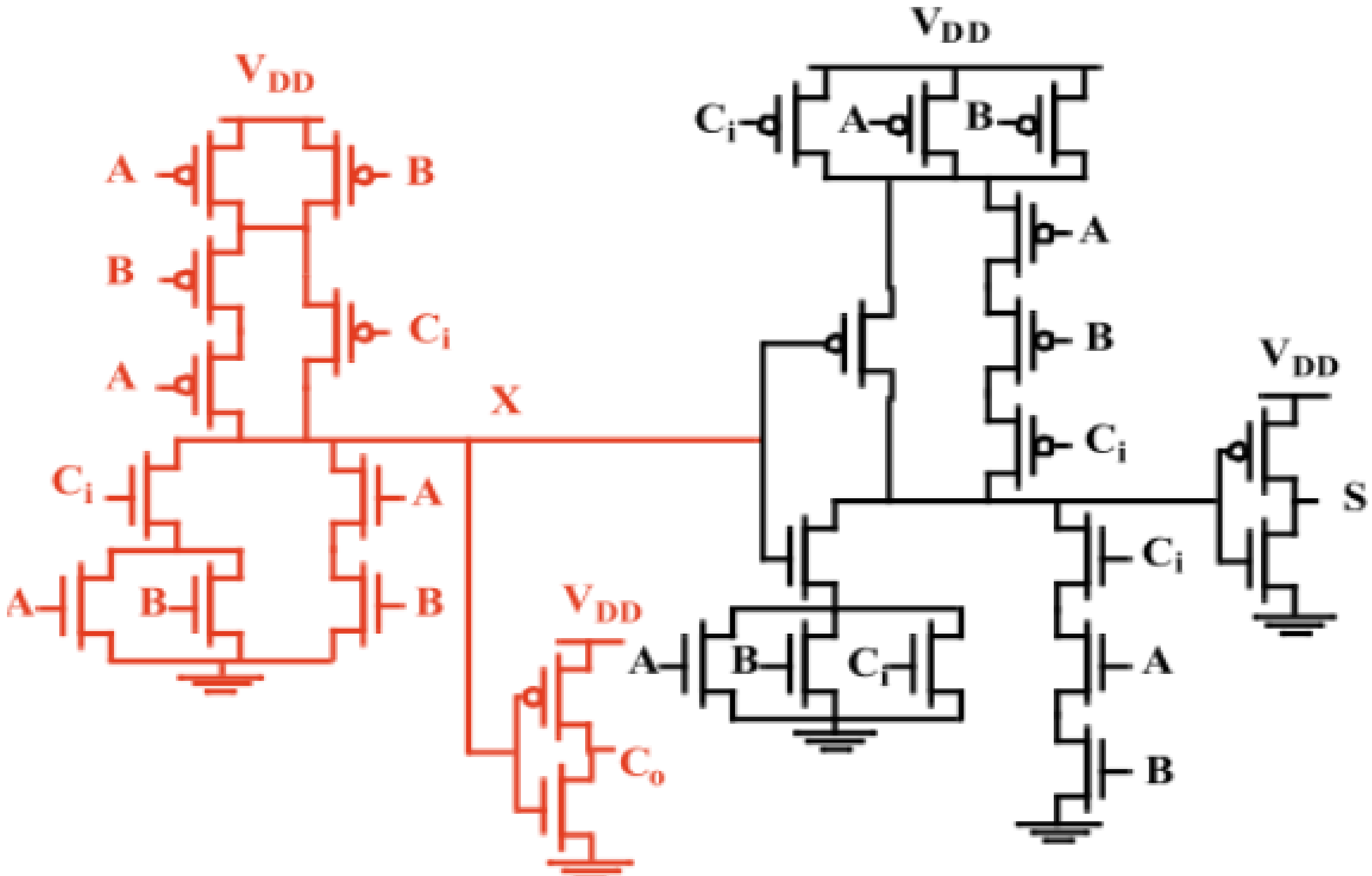
全加器电路设计考虑

- 利用加法器反向特性设计的加法器结构



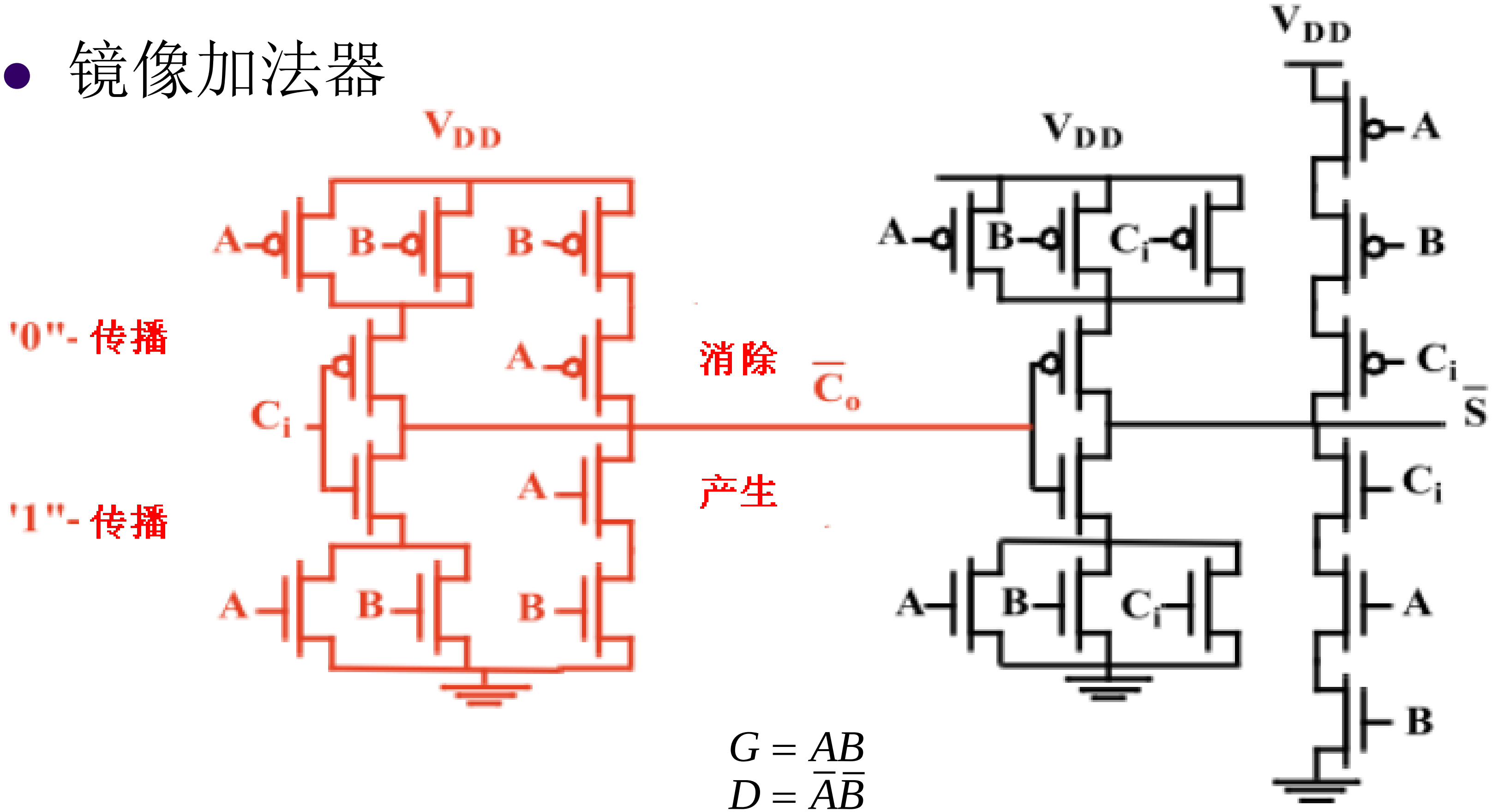
全加器电路设计考虑

- 互补静态CMOS全加器



二进制加法器

- 镜像加法器



$$G = AB$$
$$D = \bar{A}\bar{B}$$
$$P = A \oplus B$$

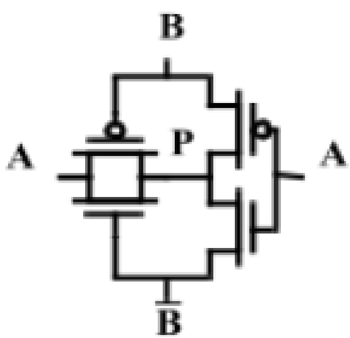
$$C_o(G, P) = G + PC_i$$
$$S(G, P) = P + C_i$$

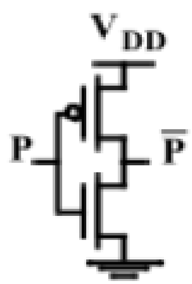
二进制加法器

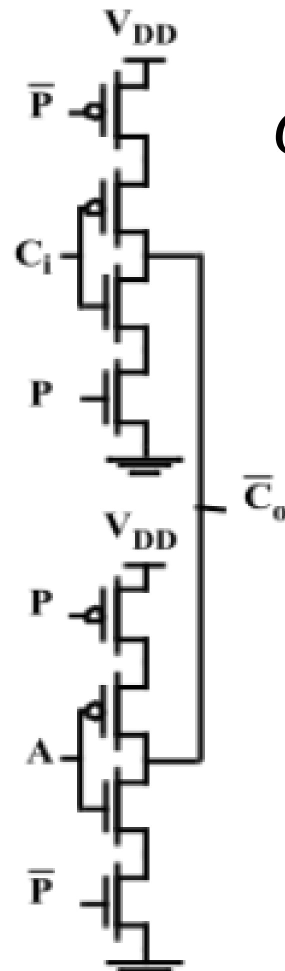
- 传输门型加法器

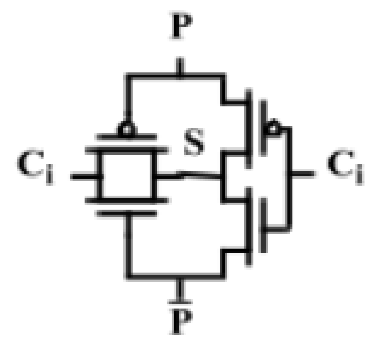
$$G = AB$$
$$D = \overline{A}\,\overline{B}$$
$$P = A \oplus B$$

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



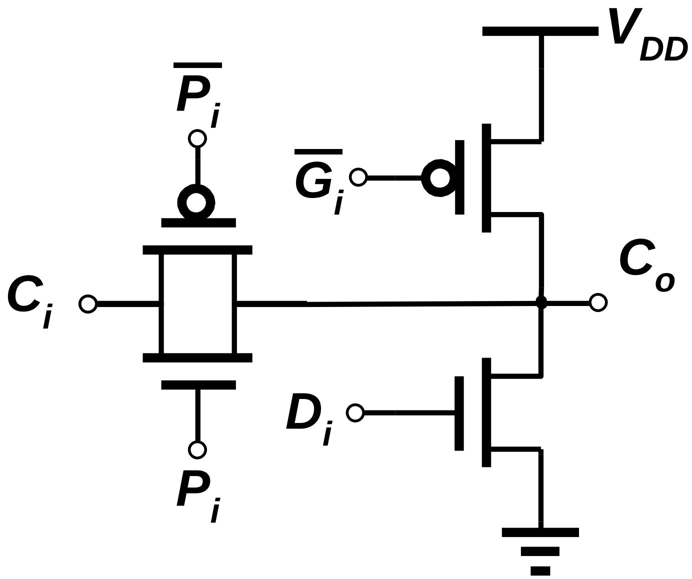


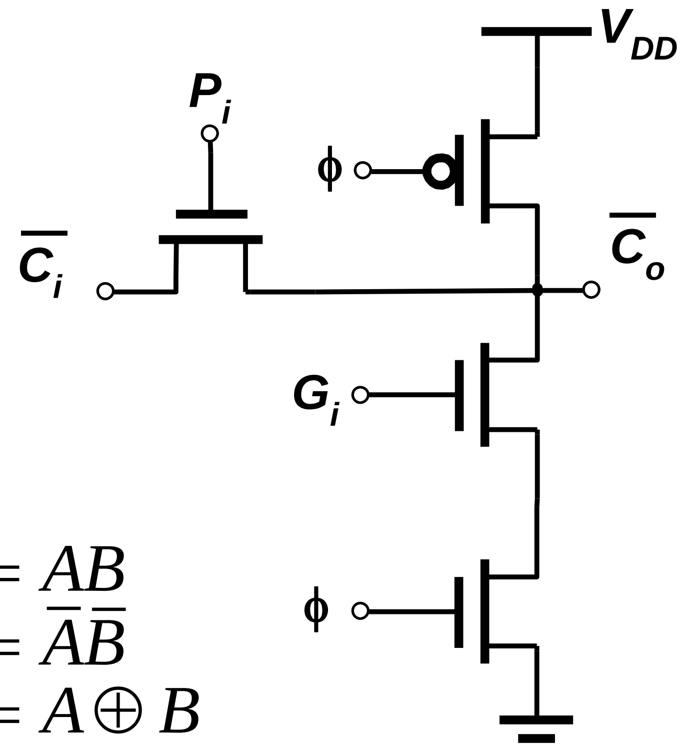




二进制加法器

- 曼彻斯特进位链加法器



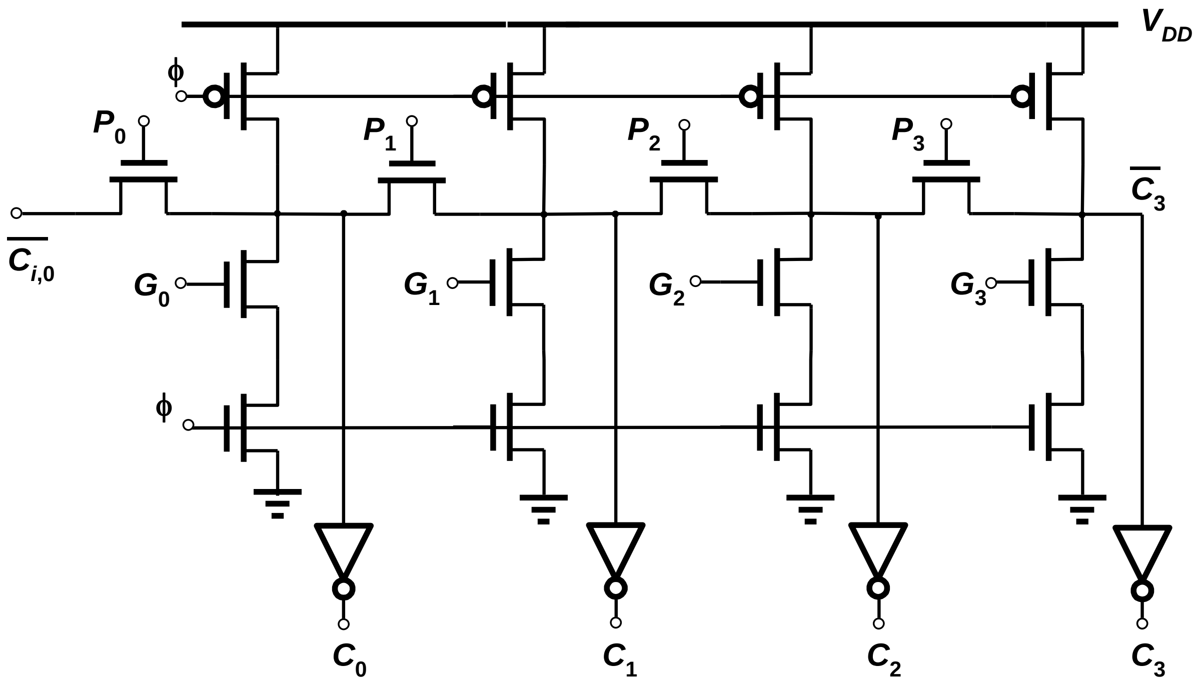


$$G = AB$$
$$D = \overline{A}\,\overline{B}$$
$$P = A \oplus B$$

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

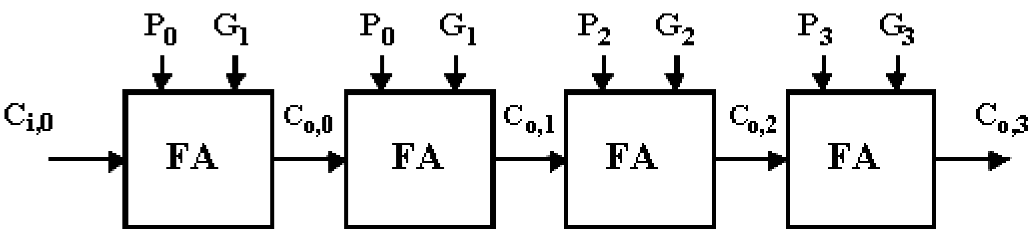
二进制加法器

- 曼彻斯特进位链加法器



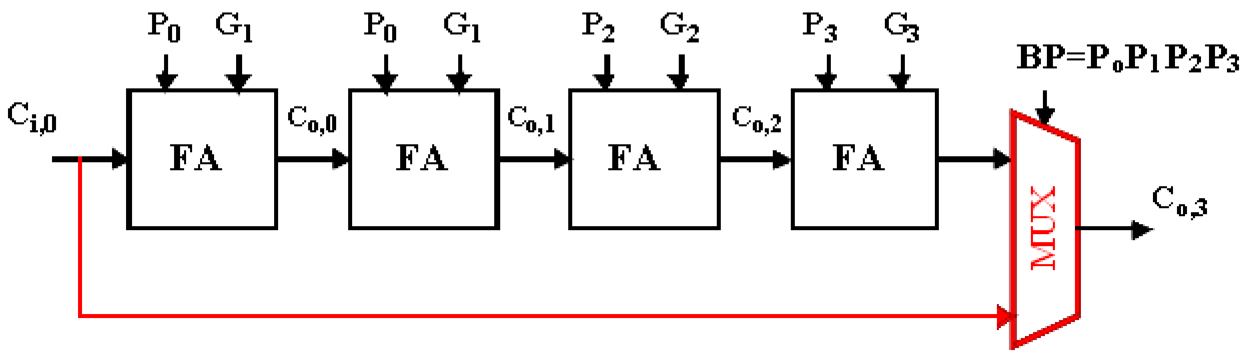
二进制加法器

- 旁路进位加法器



$$G = AB$$
$$D = \overline{A}\overline{B}$$
$$P = A \oplus B$$

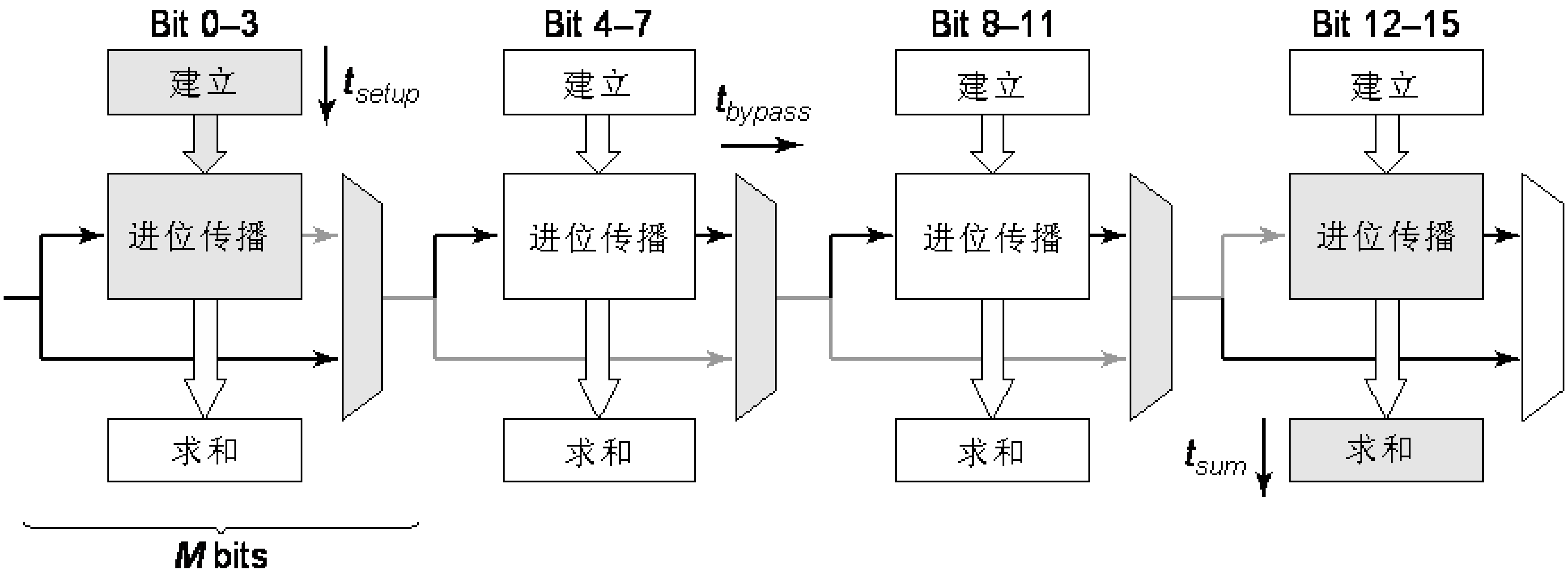
$$C_o(G,P) = G + PC_i$$
$$S(G,P) = P + C_i$$



如果$BP=P_0P_1P_2P_3$，$C_{i,0}=C_{o,3}$

二进制加法器

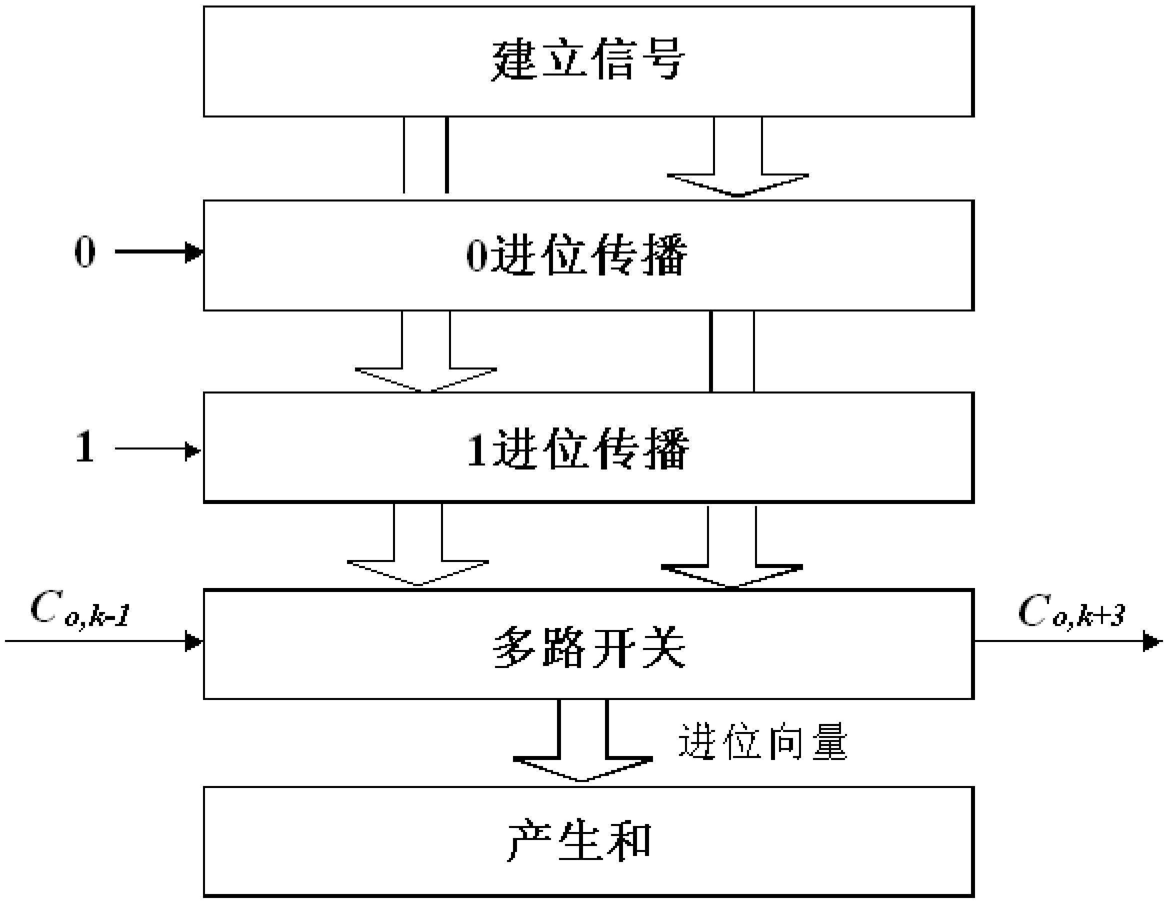
- 旁路进位加法器
  - 关键路径



$$t_{adder} = t_{setup} + (N/M\text{-}1)t_{carry} + Mt_{bypass} + (M\text{-}1)t_{carry} + t_{sum}$$

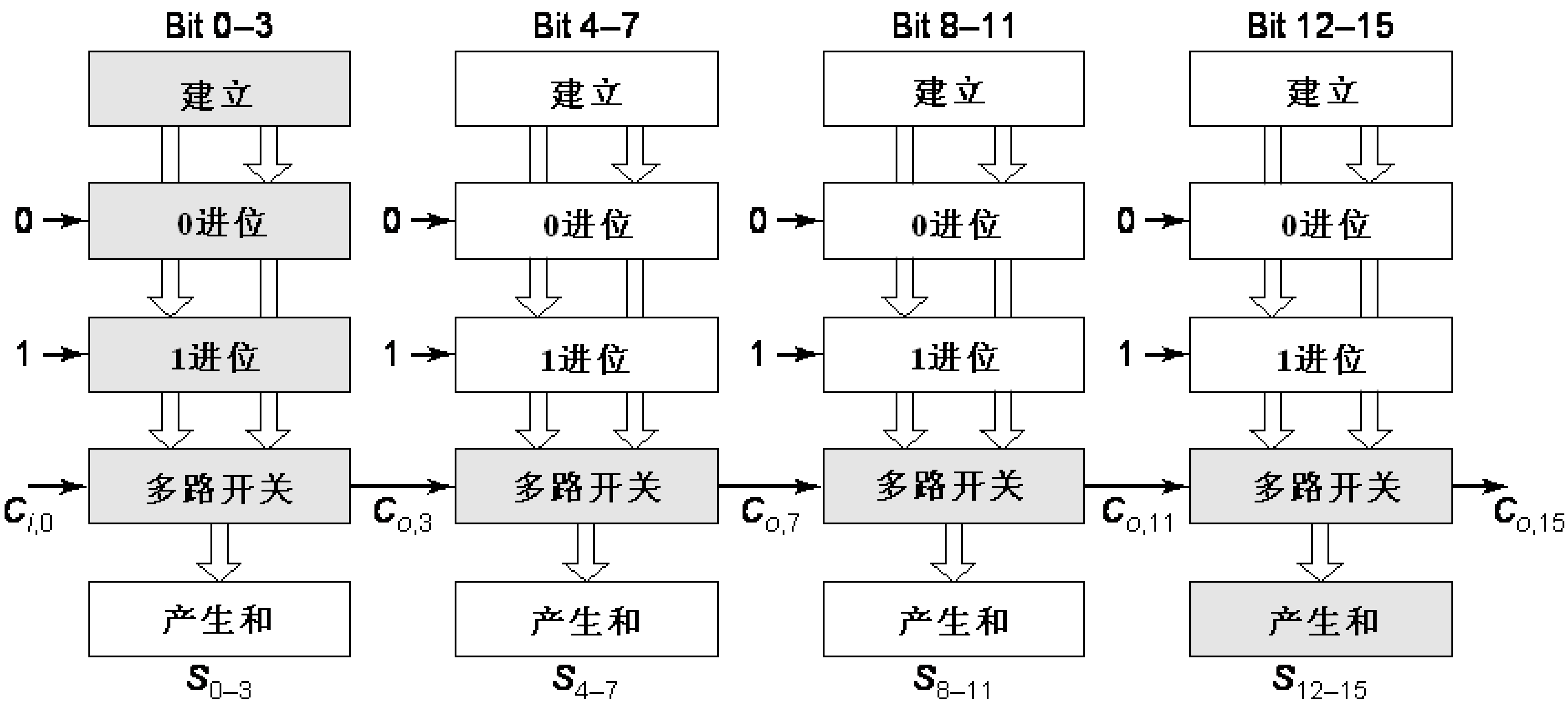
二进制加法器

- 线性进位选择加法器



二进制加法器

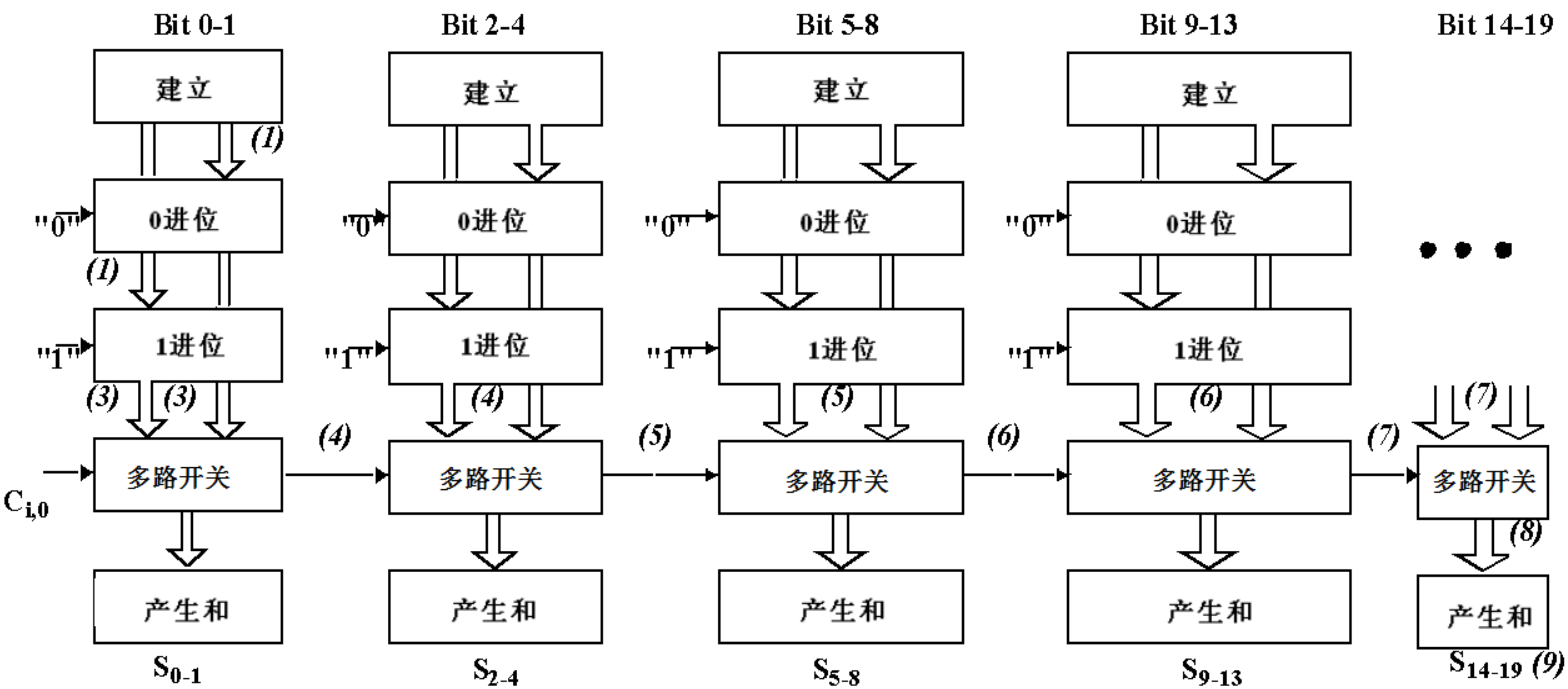
- 线性进位选择加法器
  - 关键路径



$$t_{adder} = t_{setup} + (N/M)t_{carry} + Mt_{mux} + t_{sum}$$

二进制加法器

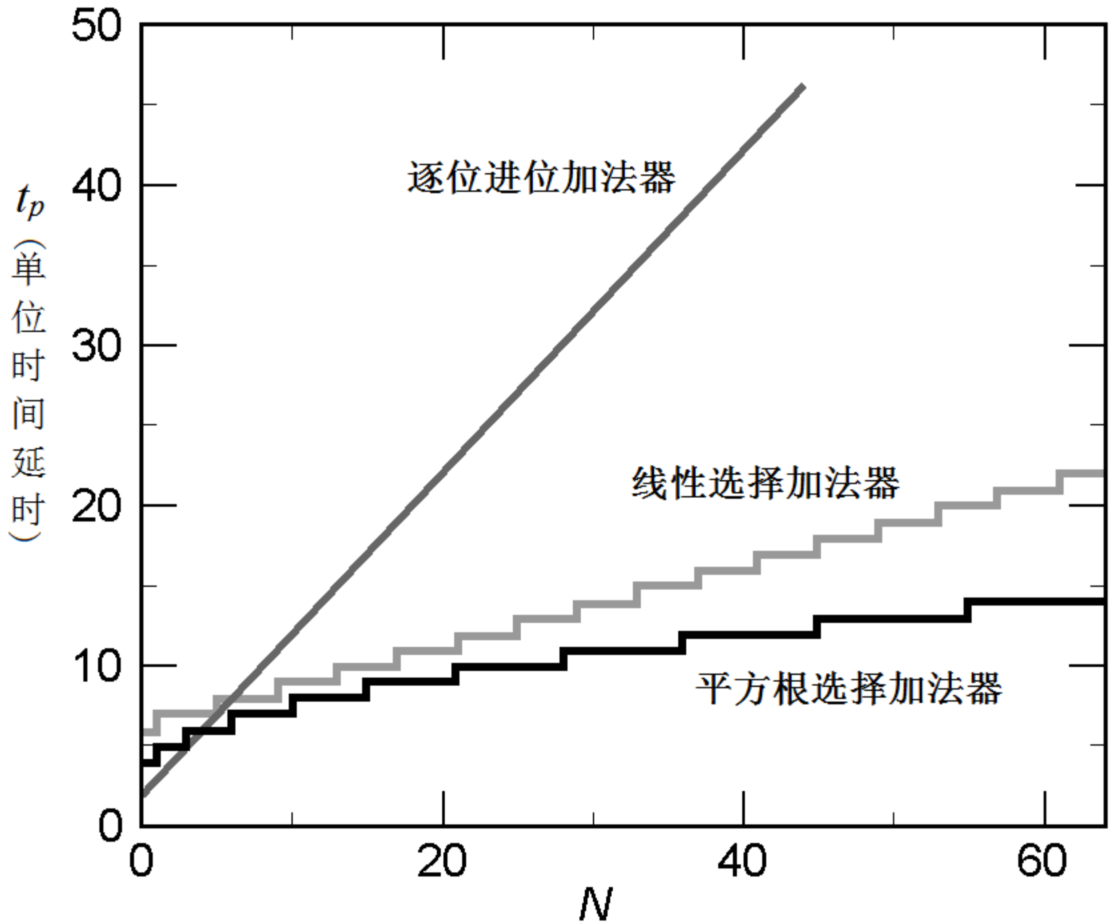
- 平方根进位选择加法器



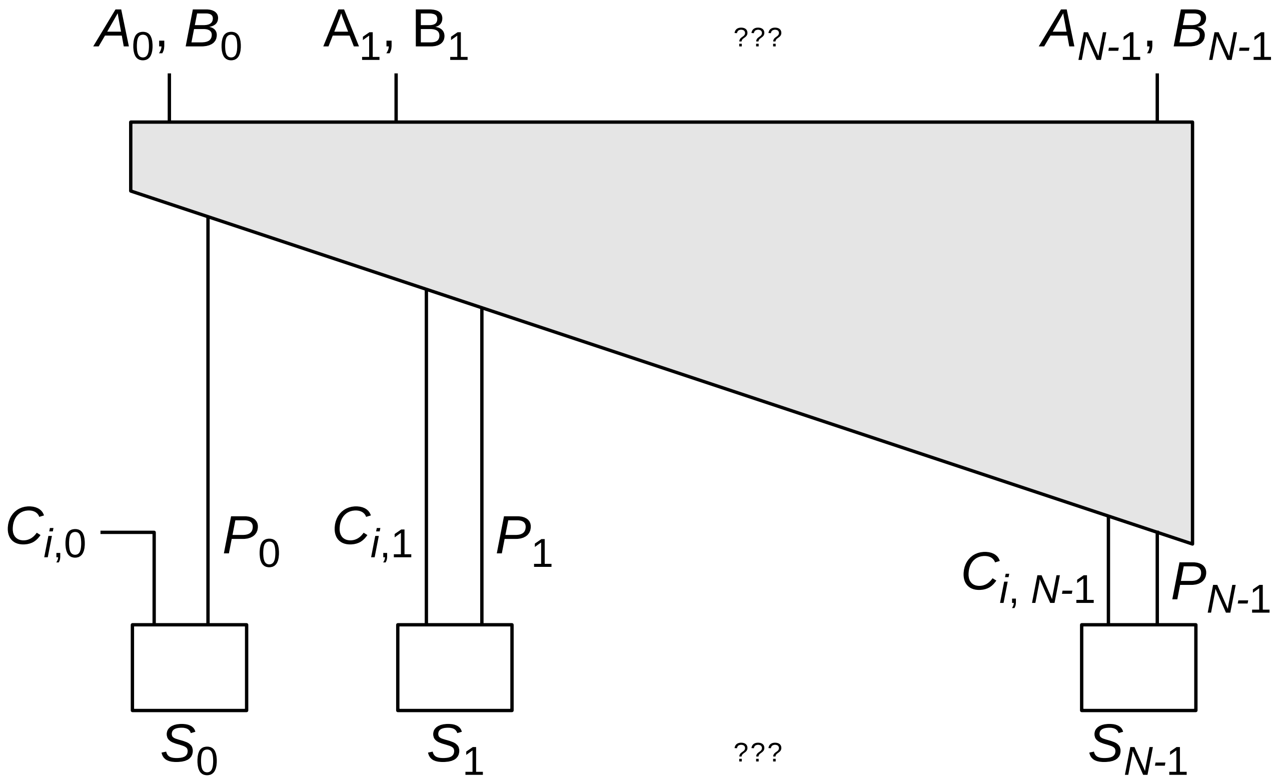
$$t_{adder} = t_{setup} + Pt_{carry} + (2N)^{1/2}t_{mux} + t_{sum}$$

二进制加法器

- 几种进位选择加法器的传播延时比较

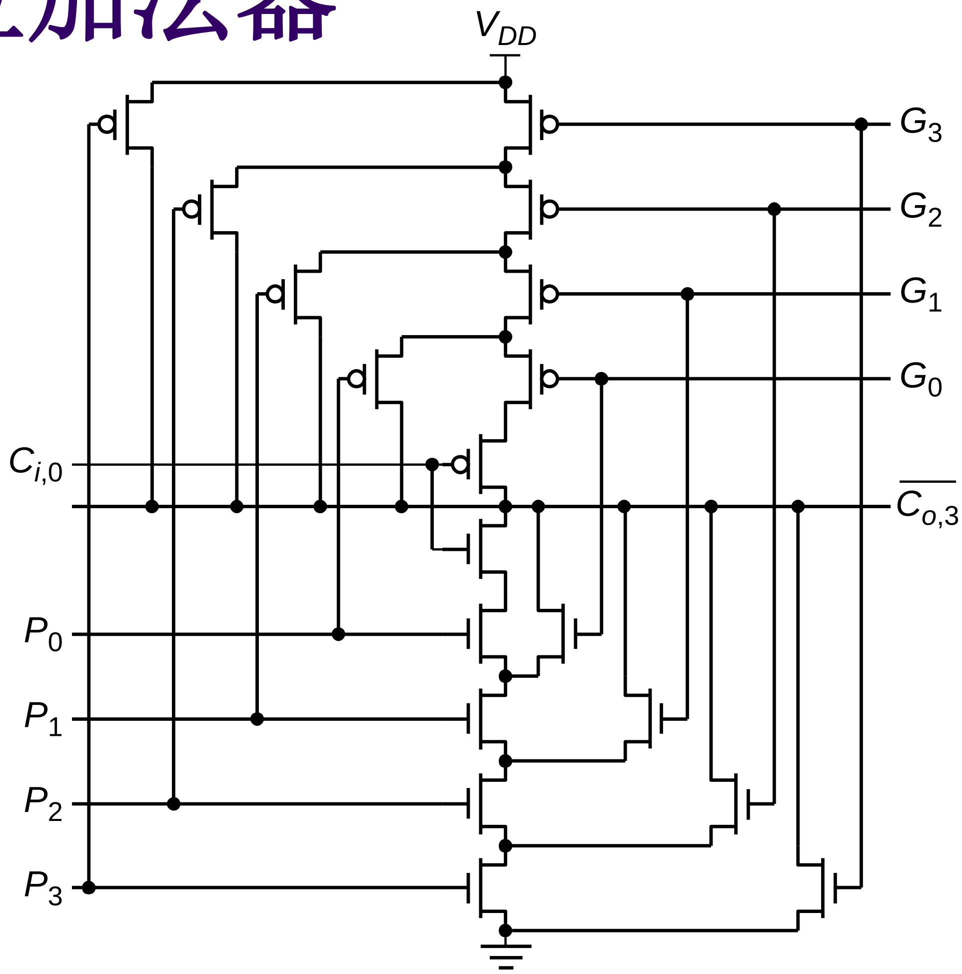


超前进位加法器



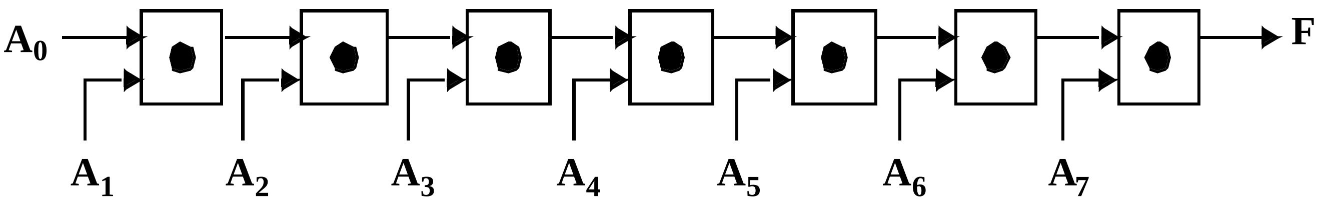
$$C_{o,k}=G_k + P_kC_{o,k-1}=G_k + P_k(G_{k-1}+P_{k-1}C_{o,k-2})$$
$$=G_k + P_k(G_{k-1}+P_{k-1}(\ldots+P_1(G_0+P_0C_{i,0})))$$

超前进位加法器

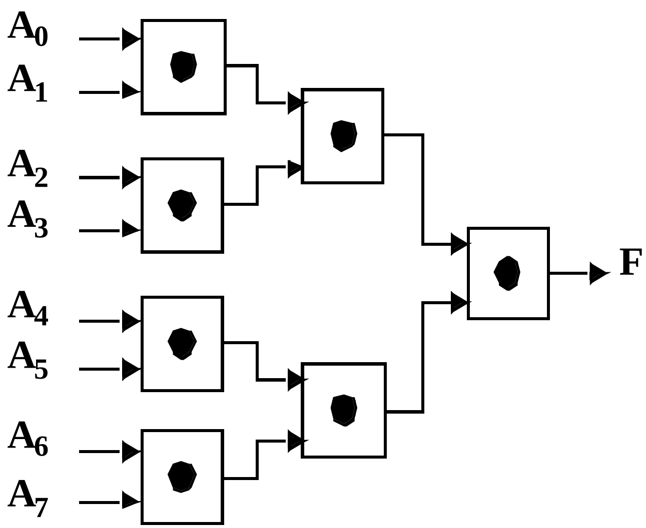


$$C_{o,k} = G_k + P_k C_{o,k-1} = G_k + P_k(G_{k-1} + P_{k-1} C_{o,k-2})$$
$$= G_k + P_k(G_{k-1} + P_{k-1}(\ldots + P_1(G_0 + P_0 C_{i,0})))$$

对数超前进位加法器



$t_p \sim N$



$t_p \sim \log_2(N)$

对数超前进位加法器

$$C_{o,k}=G_k + P_kC_{o,k-1}=G_k + P_k(G_{k-1}+P_{k-1}C_{o,k-2})$$

$$=G_k + P_k(G_{k-1}+P_{k-1}(\ldots+P_1(G_0+P_0C_{i,0})))$$

$$(G_i+P_iG_j,\ P_iP_j)=(G_i,\ P_i) \bullet (G_j,\ P_j)$$

$$C_{o,0}=G_0+P_0C_{i,0}=(G_0,\ P_0) \bullet (C_{i,0},\ 0)$$

$$C_{o,1}=G_1+P_1G_0+P_1P_0C_{i,0}=(G_1,\ P_1) \bullet (G_0,\ P_0) \bullet (C_{i,0},\ 0)$$

$$= (G_{1:0},\ P_{1:0}) \bullet (C_{i,0},\ 0)$$

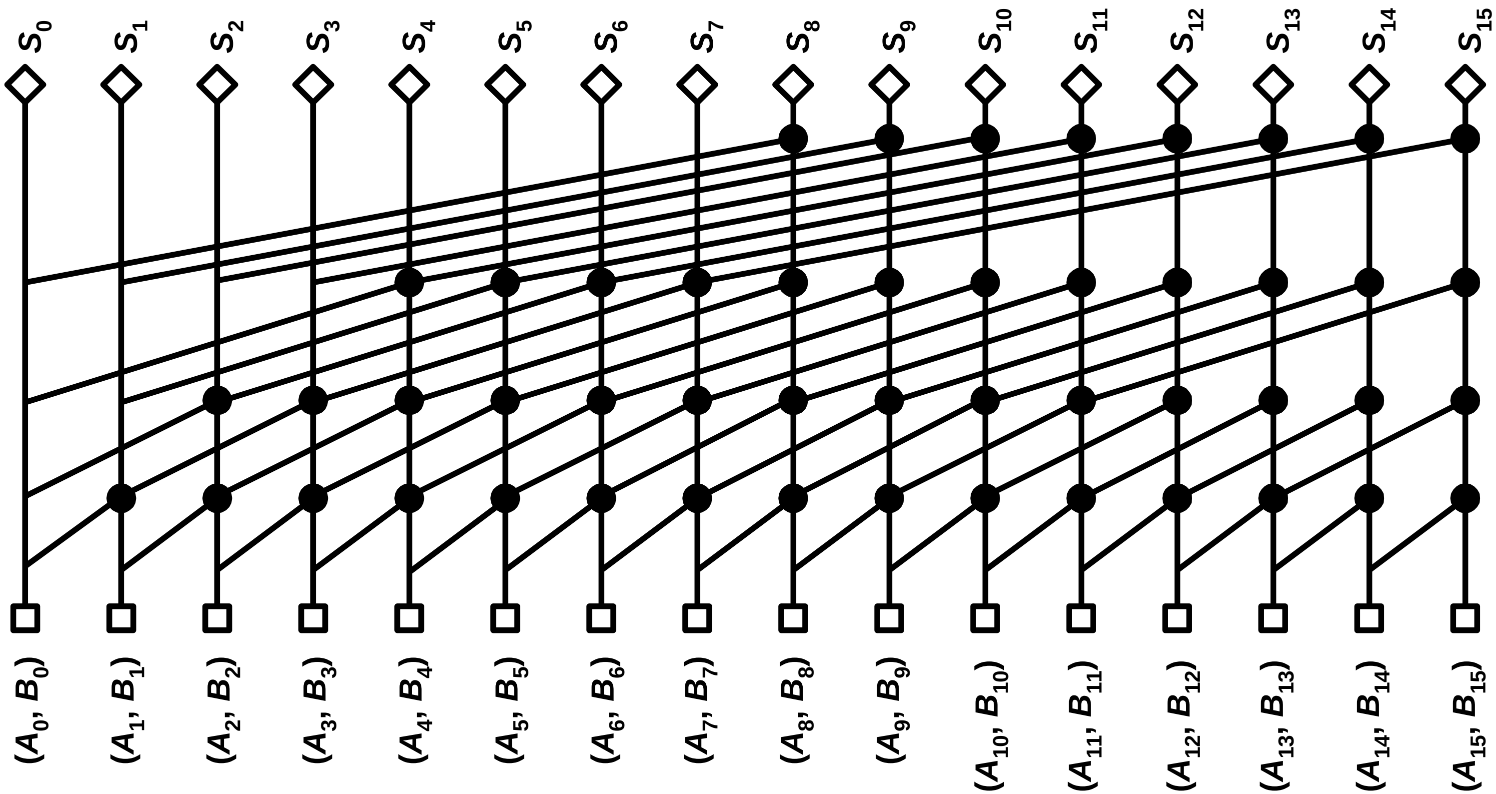
$$C_{o,2}=G_2+P_2G_1+P_2P_1G_0+P_2P_1P_0C_{i,0}$$

$$=(G_2+P_2G_1)+(P_2P_1)(G_0+P_0C_{i,0})$$

$$=(G_2,\ P_2) \bullet (G_1,\ P_1) \bullet (G_0,\ P_0) \bullet (C_{i,0},\ 0)$$

$$=(G_{2:1},\ P_{2:1}) \bullet (G_0,\ P_0) \bullet (C_{i,0},\ 0)$$

对数超前进位加法器



S_0 S_1 S_2 S_3 S_4 S_5 S_6 S_7 S_8 S_9 S_{10} S_{11} S_{12} S_{13} S_{14} S_{15}
(A_0, B_0) (A_1, B_1) (A_2, B_2) (A_3, B_3) (A_4, B_4) (A_5, B_5) (A_6, B_6) (A_7, B_7) (A_8, B_8) (A_9, B_9) (A_{10}, B_{10}) (A_{11}, B_{11}) (A_{12}, B_{12}) (A_{13}, B_{13}) (A_{14}, B_{14}) (A_{15}, B_{15})

乘法器

- 乘法器定义

$$X=\sum_{i=0}^{M-1} X_i 2^i \qquad Y=\sum_{j=0}^{N-1} Y_j 2^j$$

$$Z = X \times Y = \sum_{k=0}^{M+N-1} Z_k 2^k$$

$$=\left(\sum_{i=0}^{M-1} X_i 2^i\right)\left(\sum_{j=0}^{N-1} Y_j 2^j\right)=\sum_{i=0}^{M-1}\left(\sum_{j=0}^{N-1} X_i Y_j 2^{i+j}\right)$$

				1	0	1	0	1	0	被乘数
x						1	0	1	1	乘数
				1	0	1	0	1	0	部分积
			1	0	1	0	1	0		部分积
		0	0	0	0	0	0			部分积
+	1	0	1	0	1	0				部分积
	1	1	1	0	0	1	1	1	0	结果

乘法器

- 部分积的产生
  - 减少非零行数目
    - 波兹编码

$$01111110$$

$$10000\,0\bar{1}0$$

减少“1”的数目

部分积最多情况：1010101010

$$Y = \sum_{j=0}^{(N-1)/2} Y_j 4^j \ \left(Y_j \in \{-2,-1,0,1,2\}\right)$$

$$\begin{aligned} -2 &: \bar{1}0 \\ -1 &: 0\bar{1} \\ 0 &: 00 \\ 1 &: 01 \\ 2 &: 10 \end{aligned}$$

乘法器

- 部分积的产生
  - 波兹编码方法

部分积选择表

乘数位	编码位
000	0
001	+ 被乘数
010	+ 被乘数
011	+ 2 ×被乘数
100	− 2 ×被乘数
101	− 被乘数
110	− 被乘数
111	0

$$-2:\bar{1}0$$
$$-1:0\bar{1}$$
$$0:00$$
$$1:01$$
$$2:10$$

$$01111110$$

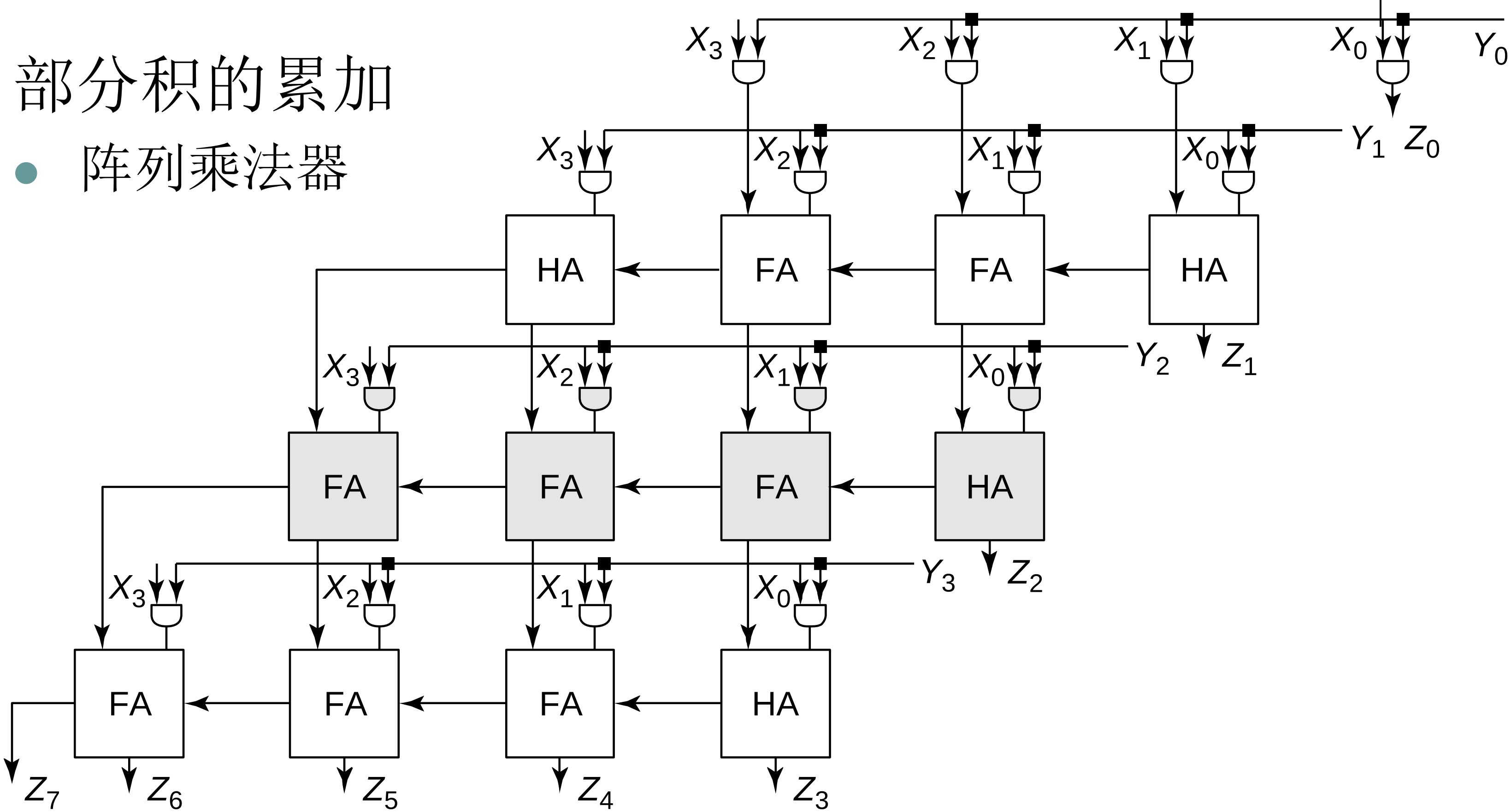
$$01(1),11(1),11(1),10(0)$$

$$011=10$$
$$111=00$$
$$111=00$$
$$100=\bar{1}0$$

$$100000\bar{1}0$$

乘法器

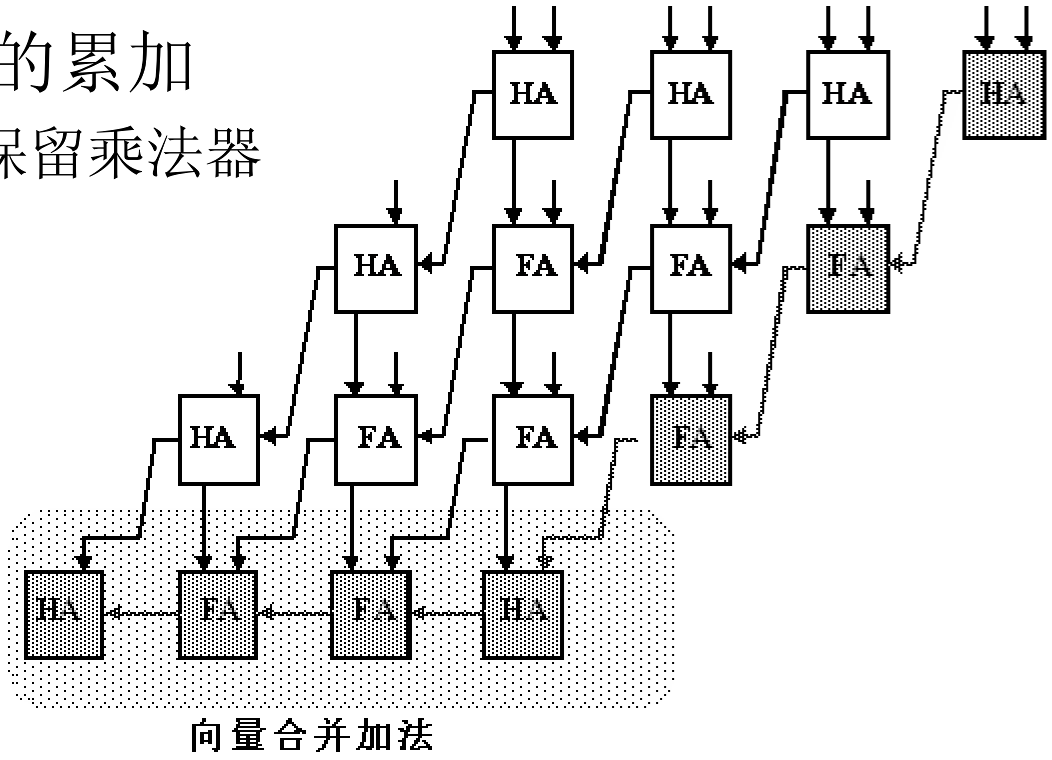
- 部分积的累加
  - 阵列乘法器



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

乘法器

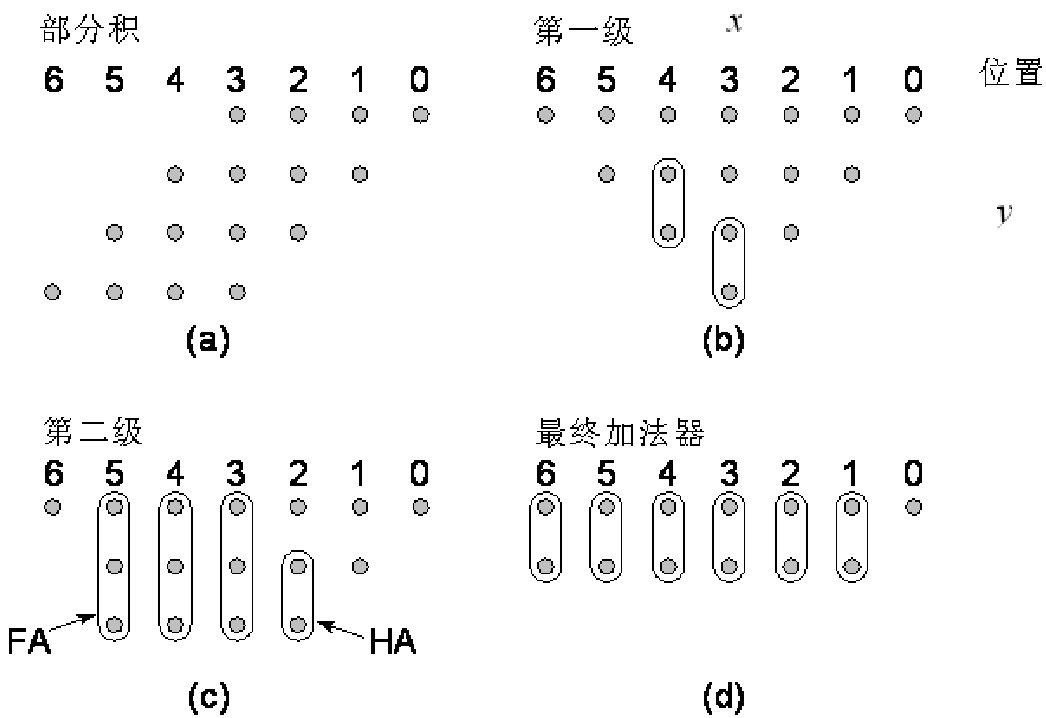
- 部分积的累加
  - 进位保留乘法器



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

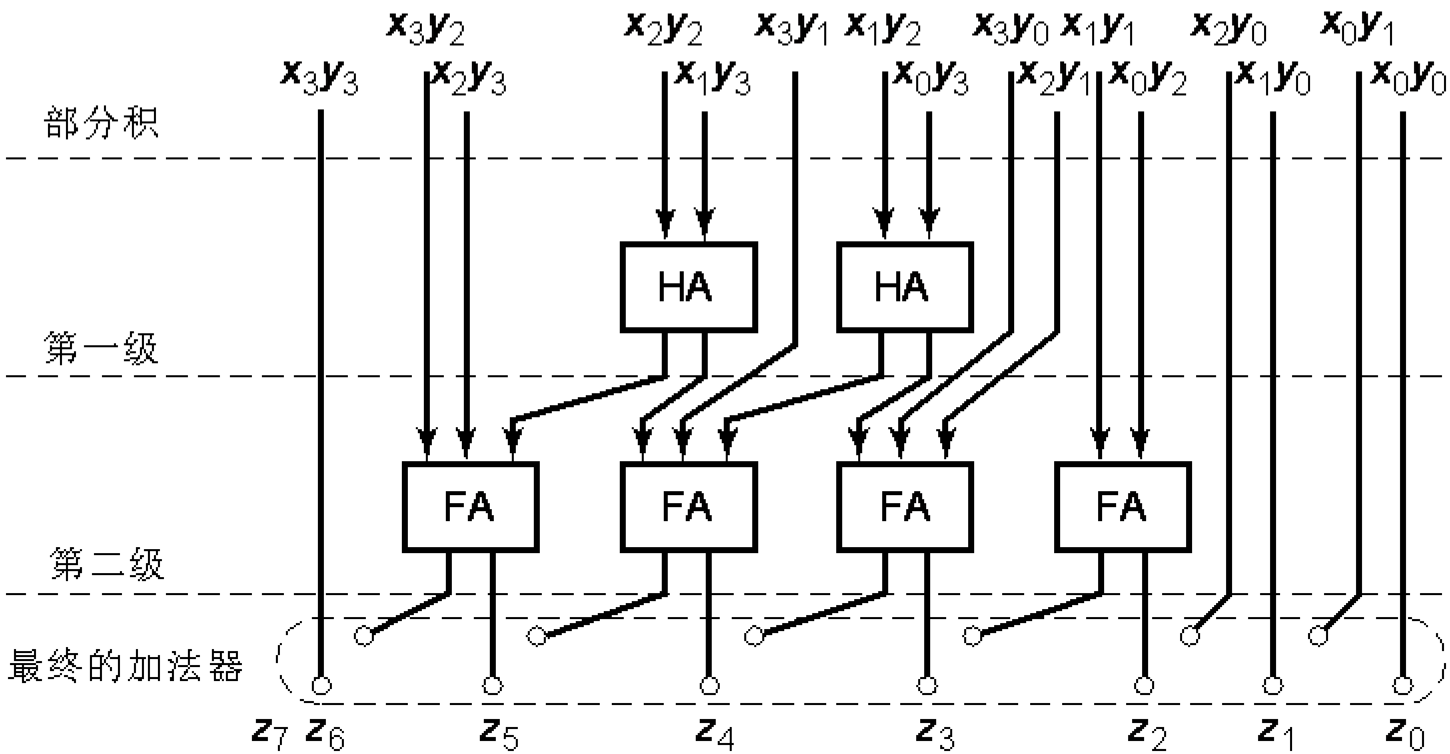
乘法器

- 部分积的累加
  - 树形乘法器（华莱士树）
    - 减少关键路径和所需的加法器单元数目
    - 全加器：3-2压缩器



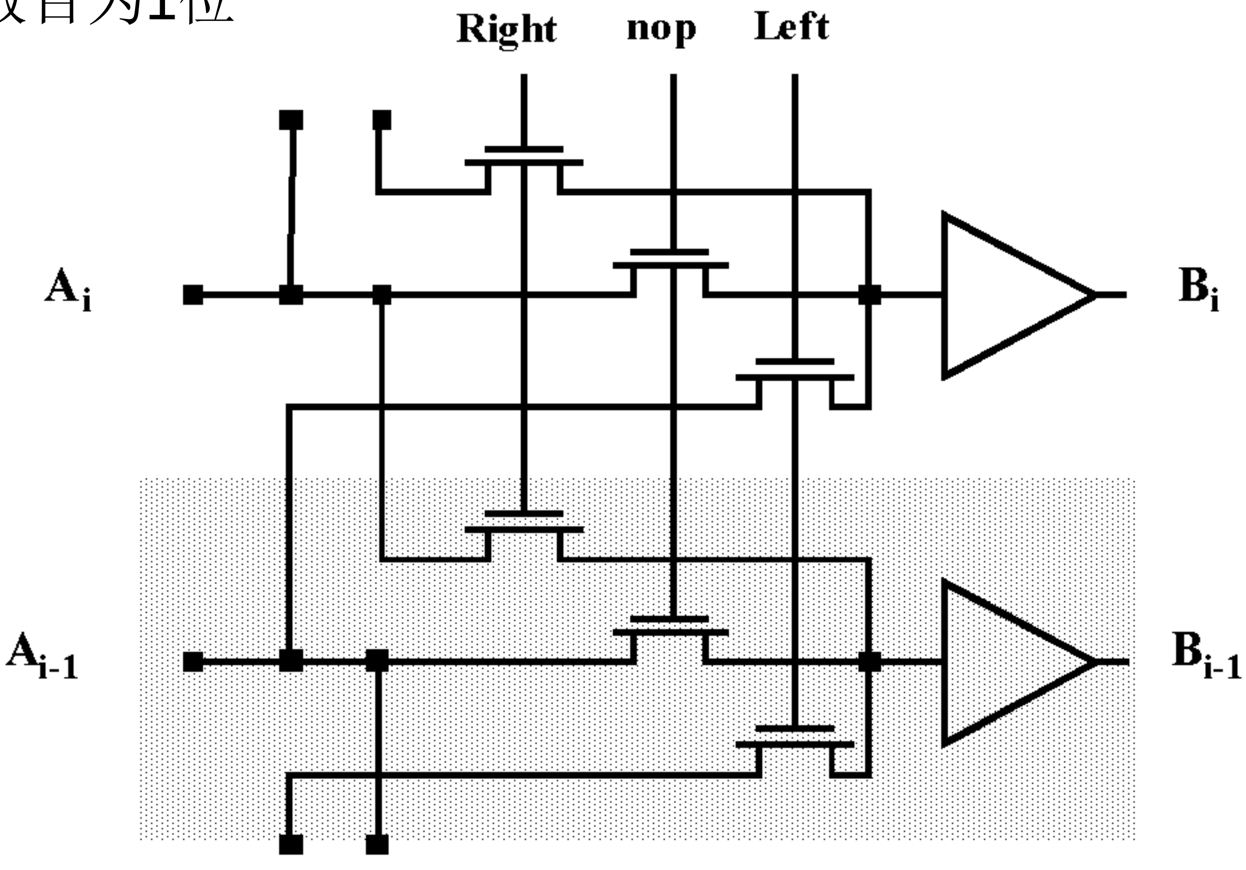
乘法器

- 部分积的累加
  - 树形乘法器（华莱士树）



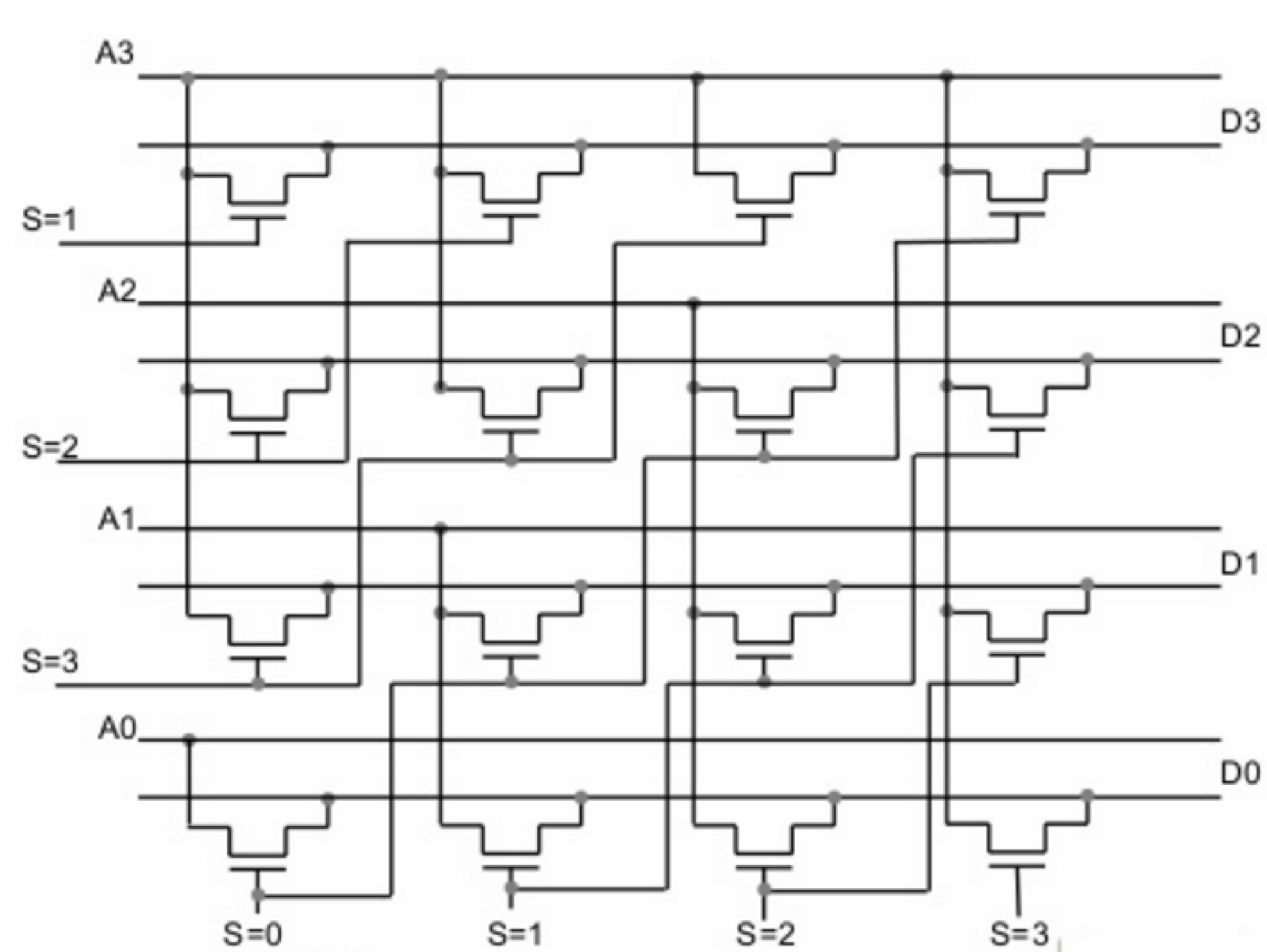
移位器

- 移位数目为1位

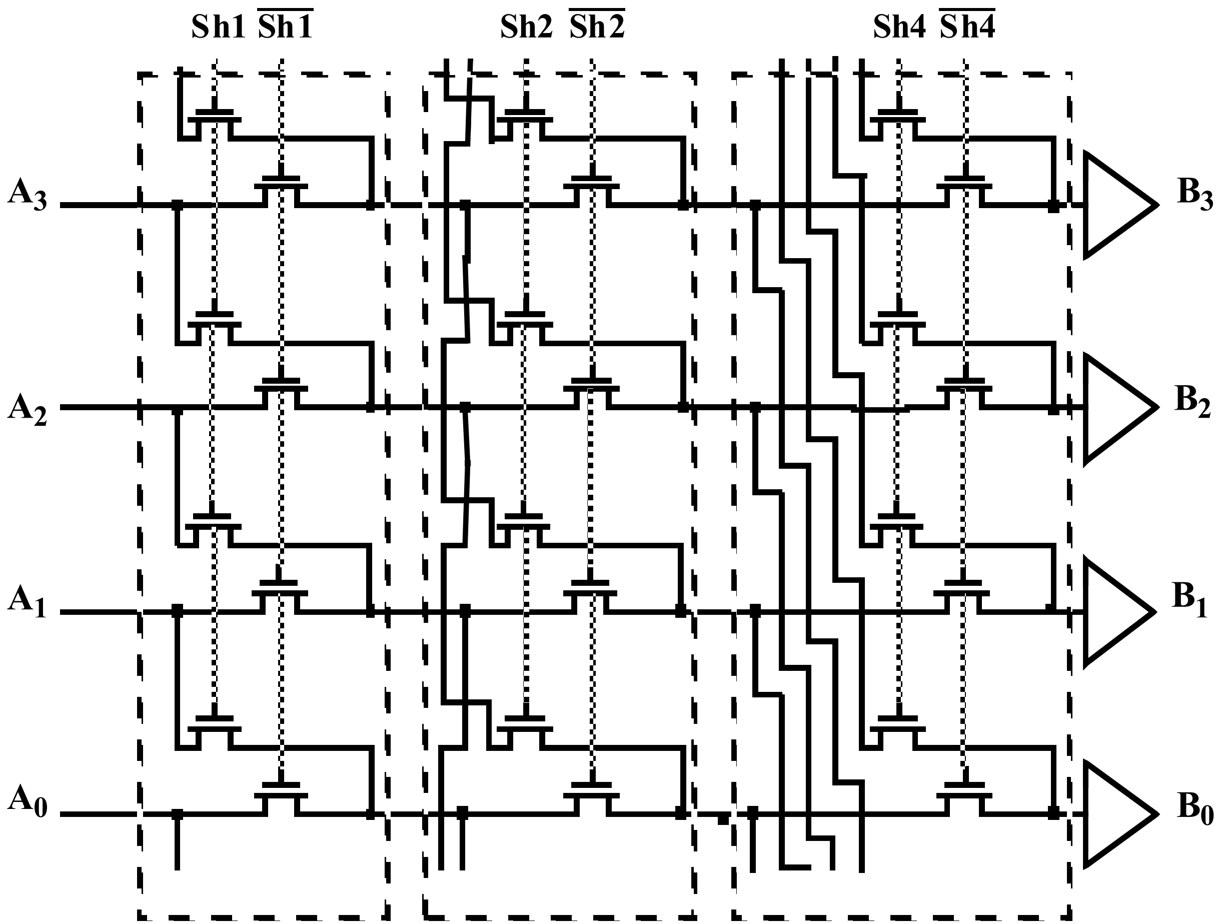


...

桶形移位器



对数移位器



Sh1 $\overline{\text{Sh1}}$
Sh2 $\overline{\text{Sh2}}$
Sh4 $\overline{\text{Sh4}}$
A_3
A_2
A_1
A_0
B_3
B_2
B_1
B_0