

数字逻辑设计

高翠芸

School of Computer Science

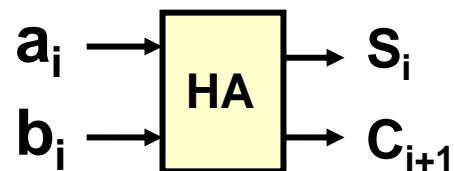
gaocuiyun@hit.edu.cn

加减法器和OC门

- 半加器
- 全加器
- 多位加法器
- 全减器
- OC门

半加器 (Half Adder)

功能：对两个1位二进制数执行相加运算

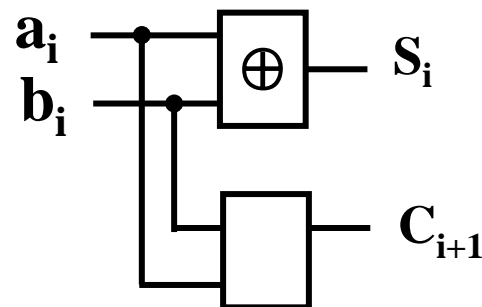


$$S_i = a_i \oplus b_i$$

$$C_{i+1} = a_i b_i$$

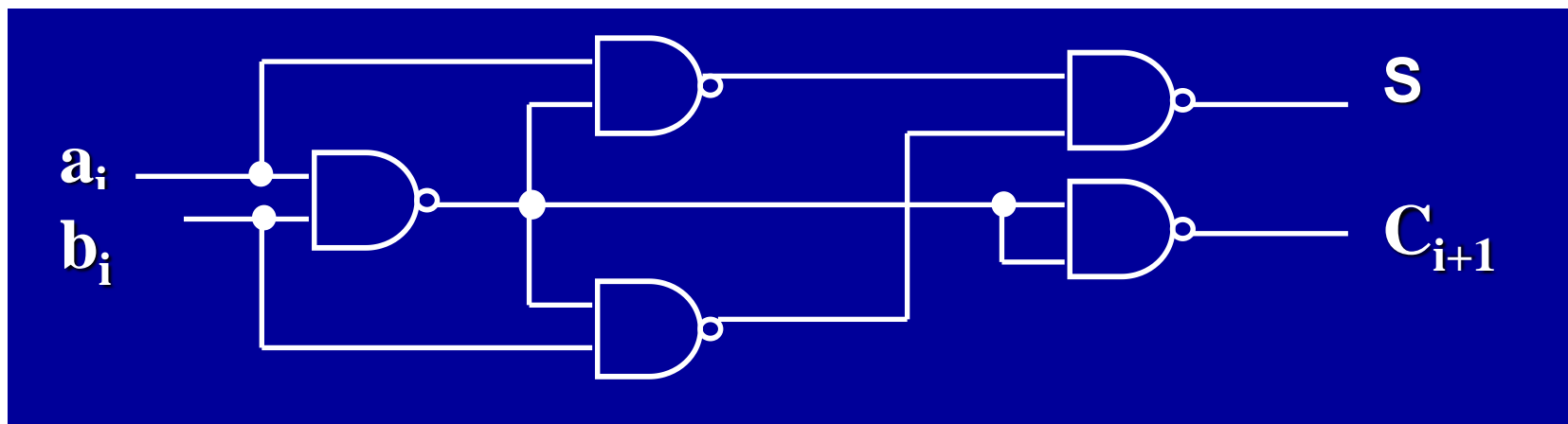
真值表

a_i	b_i	S_i	C_{i+1}
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



利用单一逻辑门与非门实现半加器

$$\left\{ \begin{aligned} S_i &= \bar{a}_i b_i + a_i \bar{b}_i = \bar{a}_i b_i + a_i \bar{b}_i + a_i \bar{a}_i + b_i \bar{b}_i \\ &= a_i (\bar{a}_i + \bar{b}_i) + b_i (\bar{a}_i + \bar{b}_i) = a_i \overline{a_i b_i} + b_i \overline{a_i b_i} \\ &= \overline{a_i a_i b_i} \overline{b_i a_i b_i} \\ C_{i+1} &= \overline{\overline{a_i b_i}} \end{aligned} \right.$$



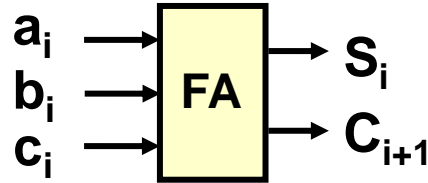
全加器 (Full Adder)

$$\begin{array}{r} 1\ 0\ 1\ 1\ \dots\dots\dots A \\ 1\ 1\ 1\ 0\ \dots\dots\dots B \\ +\quad\quad\quad 0\ \dots\dots\dots C_i \\ \hline \dots\dots\dots S_i \end{array}$$

$$A = a_3 a_2 a_1 a_0 = 1011$$

$$B = b_3 b_2 b_1 b_0 = 1110$$

全加器表示



a_i	b_i	C_i	S_i	C_{i+1}
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

	$b_i c_{i-1}$				
		00	01	11	10
a_i					
0		0	1	0	1
1		1	0	1	0

	$b_i c_i$				
		00	01	11	10
a_i					
0		0	0	1	0
1		0	1	1	1

The second table for C_{i+1} has colored circles highlighting the 1s: a blue circle around the 1 in row 1, column 3; a green circle around the 1 in row 0, column 4; and an orange circle around the 1 in row 1, column 5.

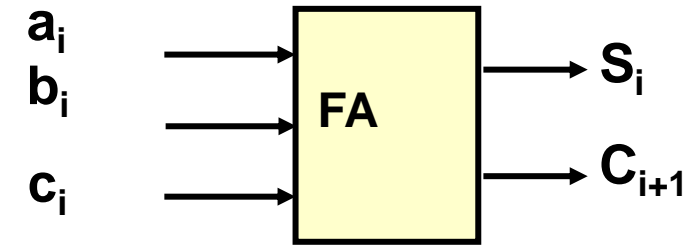
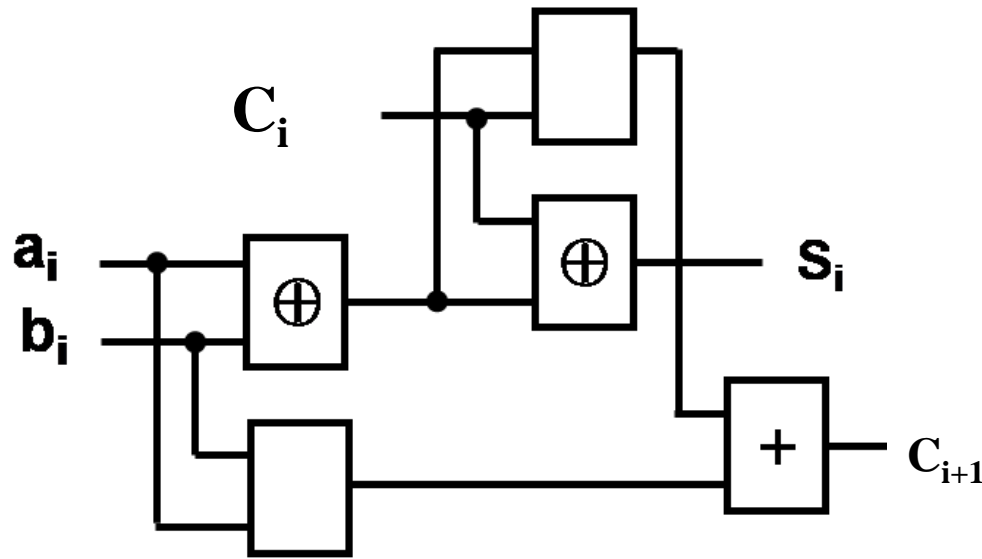
$$\begin{aligned}
 S_i &= \bar{a}_i \bar{b}_i c_i + a_i \bar{b}_i \bar{c}_i + a_i \bar{b}_i c_i + a_i b_i c_i \\
 &= (\bar{a}_i \bar{b}_i + a_i b_i) c_i + (a_i \bar{b}_i + a_i \bar{b}_i) \bar{c}_i \\
 &= (\overline{a_i \oplus b_i}) c_i + (a_i \oplus b_i) \bar{c}_i \\
 &= a_i \oplus b_i \oplus C_i
 \end{aligned}$$

$$C_{i+1} = (a_i \oplus b_i) C_i + a_i b_i$$

全加器逻辑表示

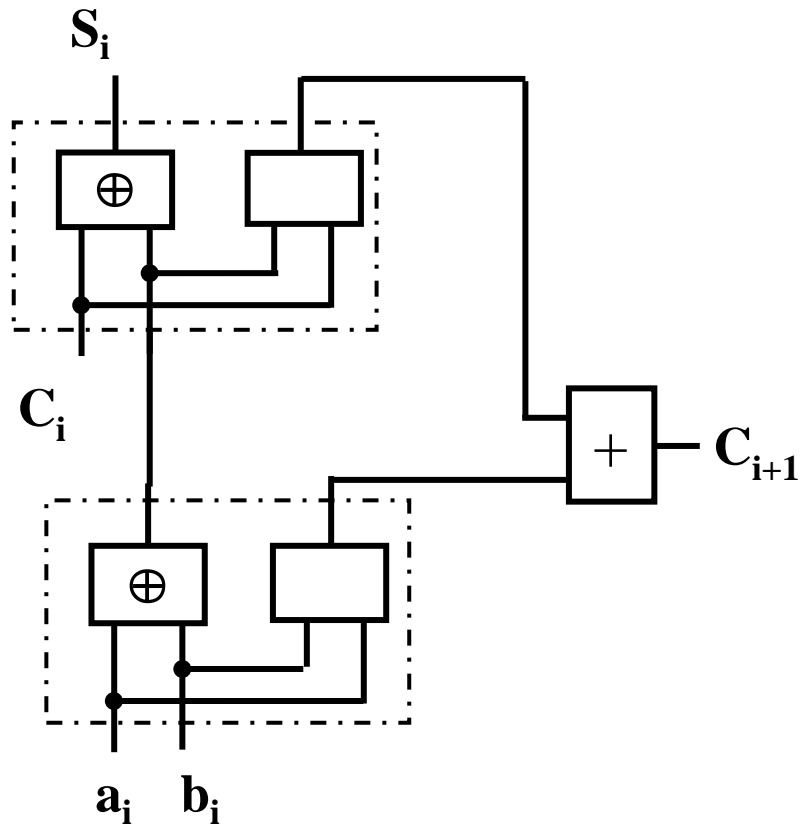
◆ solution 1:

$$\begin{cases} S_i = a_i \oplus b_i \oplus C_i \\ C_{i+1} = (a_i \oplus b_i) C_i + a_i b_i \end{cases}$$



全加器逻辑表示 (2)

◆ solution 2



$$S_i = a_i \oplus b_i \oplus C_i$$

$$C_{i+1} = (a_i \oplus b_i) C_i + a_i b_i$$

$$\begin{cases} S_i = a_i \oplus b_i \\ C_{i+1} = a_i b_i \end{cases}$$

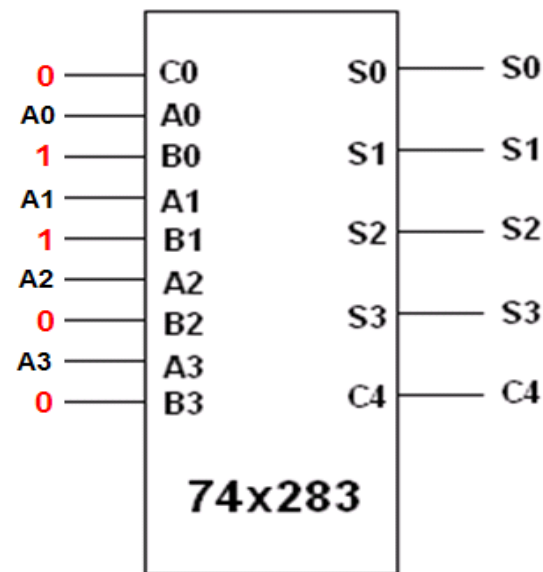
全加器的应用

典型芯片

- 74LS82: 2-bit adder
- 74LS283: 4-bit adder

二进制数 $A_3 A_2 A_1 A_0$	余三码 $S_3 S_2 S_1 S_0$	二进制数 $A_3 A_2 A_1 A_0$	余三码 $S_3 S_2 S_1 S_0$
0 0 0 0	0 0 1 1	1 0 0 0	1 0 1 1
0 0 0 1	0 1 0 0	1 0 0 1	1 1 0 0
0 0 1 0	0 1 0 1	1 0 1 0	×
0 0 1 1	0 1 1 0	1 0 1 1	×
0 1 0 0	0 1 1 1	1 1 0 0	×
0 1 0 1	1 0 0 0	1 1 0 1	×
0 1 1 0	1 0 0 1	1 1 1 0	×
0 1 1 1	1 0 1 0	1 1 1 1	×

应用——余3码产生器



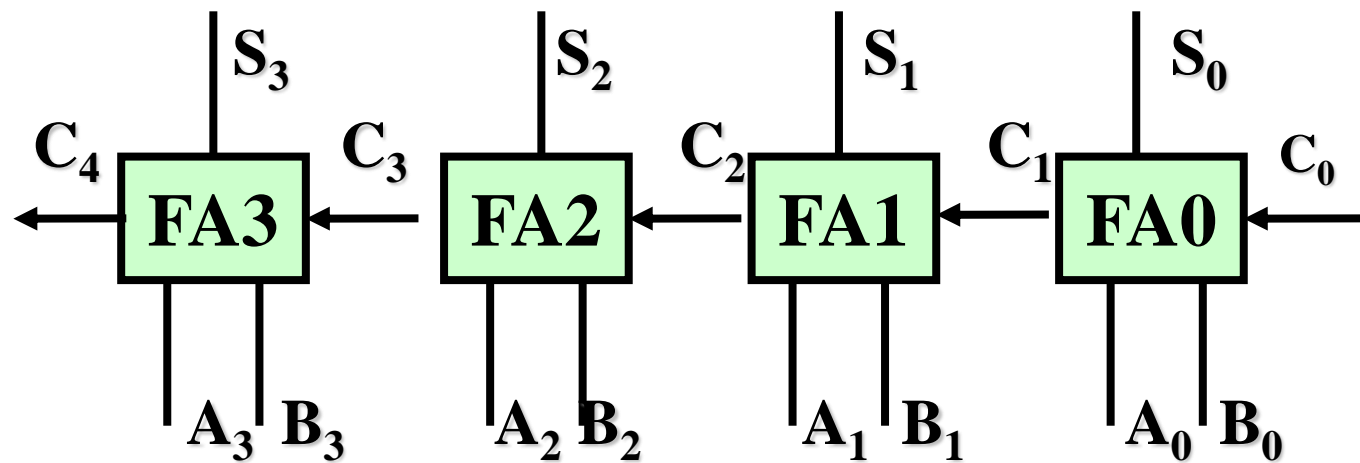
$A_3 A_2 A_1 A_0$: 输入 8421 BCD码

$S_3 S_2 S_1 S_0$: 输出余3 码

$$S = A + 0011$$

4位并行加法器

(1) 串行进位



$$S_i = a_i \oplus b_i \oplus C_i$$

$$C_{i+1} = (a_i \oplus b_i) C_i + a_i b_i$$

- 缺点：串行进位，运算速度慢
- 优点：线路简单
- 关键：进位形成时间
- 解决方案：改串行进位为并行进位

4位并行加法器

P83

(2) 超前进位


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

$$A = A_3 A_2 A_1 A_0 = 1011$$

$$B = B_3 B_2 B_1 B_0 = 1110$$

$$C_{i+1} = P_i C_i + G_i \quad \text{——进位迭代公式}$$

$$P_i = A_i \oplus B_i$$

$$G_i = A_i B_i$$

$$C_1 = P_0 C_0 + G_0$$

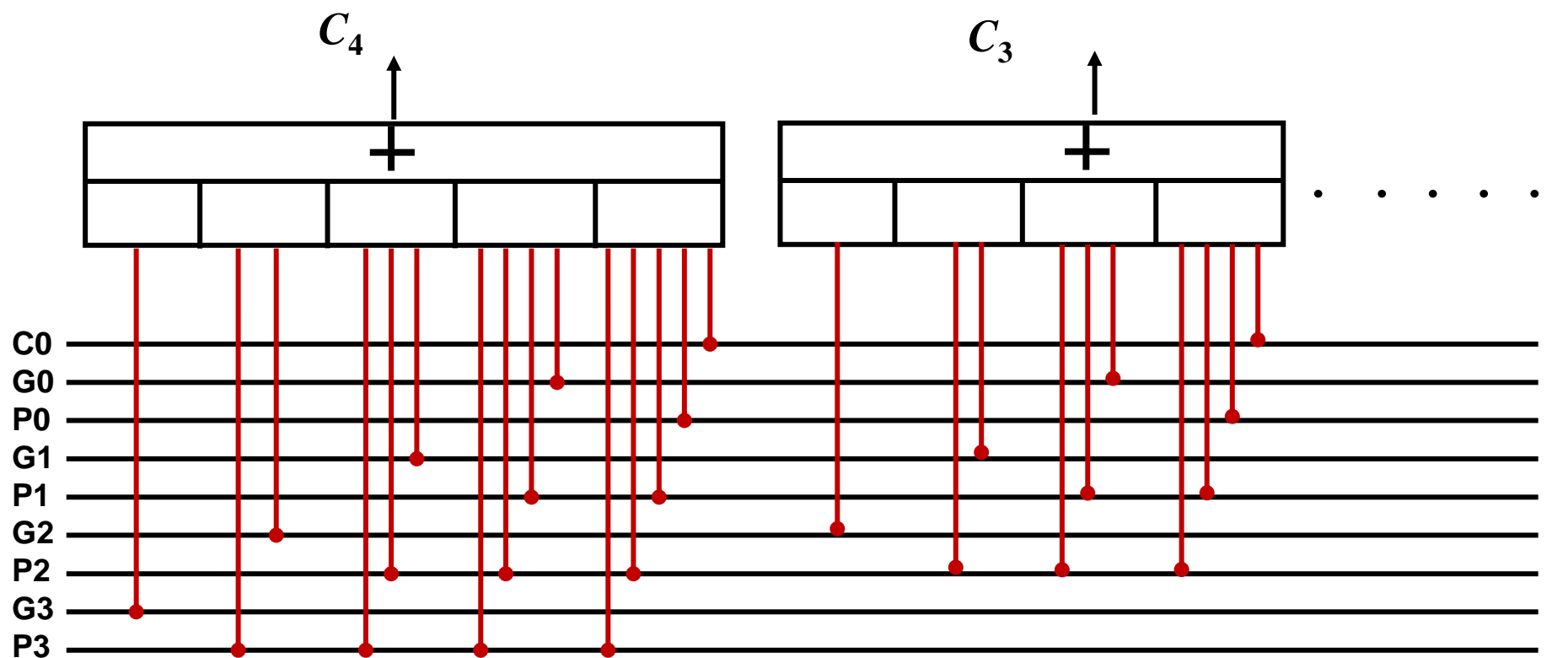
$$C_2 = P_1 C_1 + G_1 = P_1 P_0 C_0 + P_1 G_0 + G_1$$

$$C_3 = P_2 C_2 + G_2 = P_2 P_1 P_0 C_0 + P_2 P_1 G_0 + P_2 G_1 + G_2$$

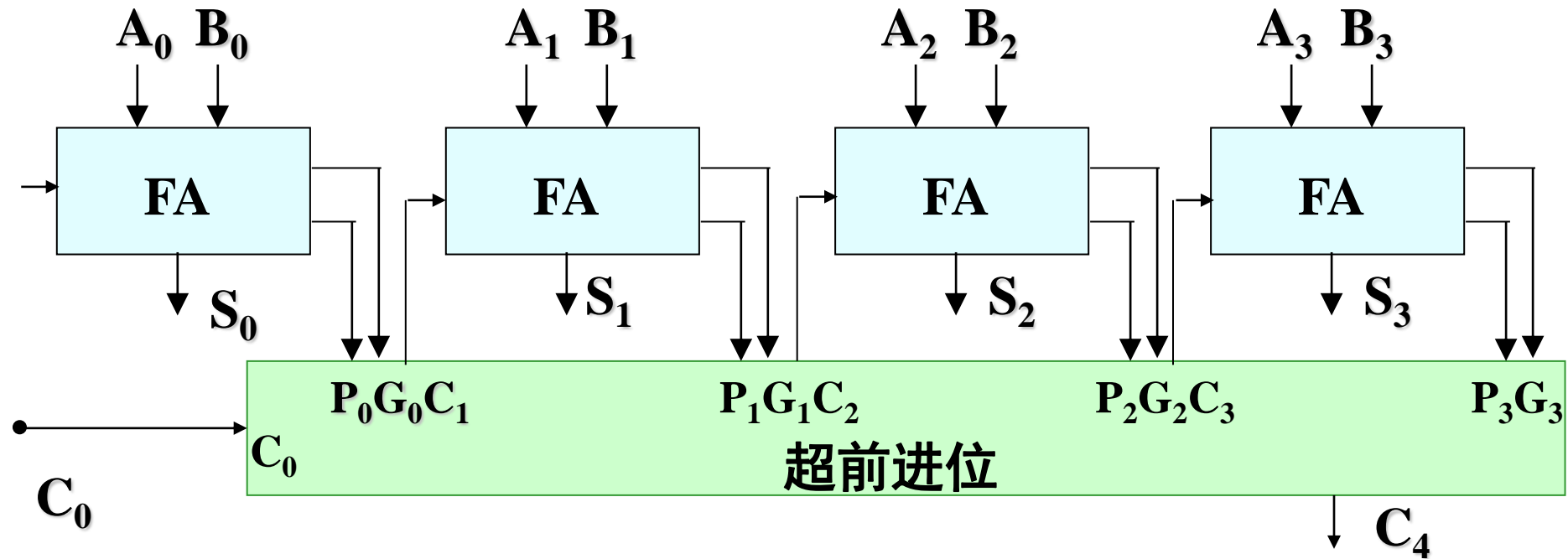
$$C_4 = P_3 C_3 + G_3 = P_3 P_2 P_1 P_0 C_0 + P_3 P_2 P_1 G_0 + P_3 P_2 G_1 + P_3 G_2 + G_3$$

4位并行加法器

(2) 超前进位



4位并行加法器



$$P_i = A_i \oplus B_i \quad G_i = A_i B_i$$

$$C_1 = P_0 C_0 + G_0$$

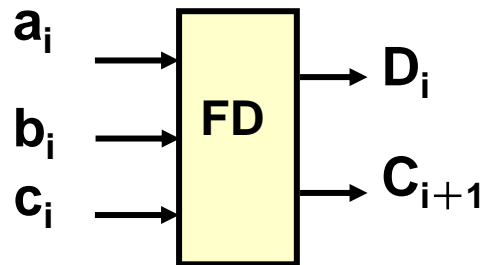
$$C_2 = P_1 C_1 + G_1 = P_1 P_0 C_0 + P_1 G_0 + G_1$$

$$C_3 = P_2 C_2 + G_2 = P_2 P_1 P_0 C_0 + P_2 P_1 G_0 + P_2 G_1 + G_2$$

$$C_4 = P_3 C_3 + G_3 = P_3 P_2 P_1 P_0 C_0 + P_3 P_2 P_1 G_0 + P_3 P_2 G_1 + P_3 G_2 + G_3$$

全减器 (Binary Full Subtractor)

$$\begin{array}{r} 1\ 1\ 1\ 0 \dots\dots\dots A \\ 1\ 0\ 1\ 1 \dots\dots\dots B \\ -\ 0 \dots\dots\dots C_i \\ \hline \dots\dots\dots D_i \end{array}$$



$$A = a_3 a_2 a_1 a_0 = 1110$$

$$B = b_3 b_2 b_1 b_0 = 1011$$

真值表

a_i	b_i	C_i	D_i	C_{i+1}
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

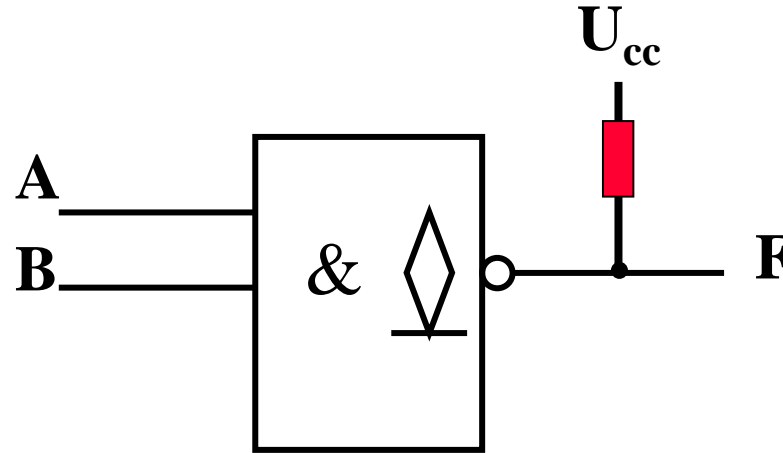
全减器——例

真值表

a_i	b_i	C_i	D_i	C_{i+1}
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

OC门（集电极开路门：Open Collector Gate）

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

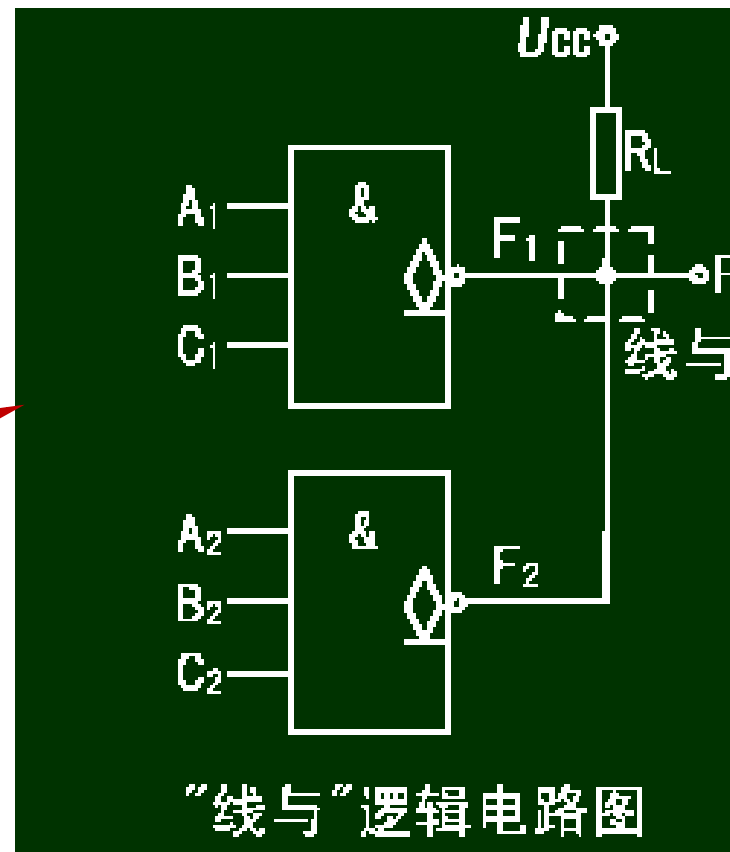


- 几个OC门的输出端可以直接互连：“线与”
- 使用时必须加负载/上拉电阻

OC门&线与

$$F = F_1 \cdot F_2 = \overline{A_1 B_1 C_1} \cdot \overline{A_2 B_2 C_2}$$

不使用OC门，
需要2个与非门、1个与门



小 结

- 半加器
- 全加器
- 多位加法器
- 全减器
- 0C门

Unit 7 组合逻辑元件

- 多路复用器(multiplexers)
- 三态器件(Three-state Buffer)
- 译码器(Decoders)
- 编码器(Encoders)
- 奇偶校验器
- 比较器
- 只读存储器(ROM)
- 利用MSI设计组合逻辑电路

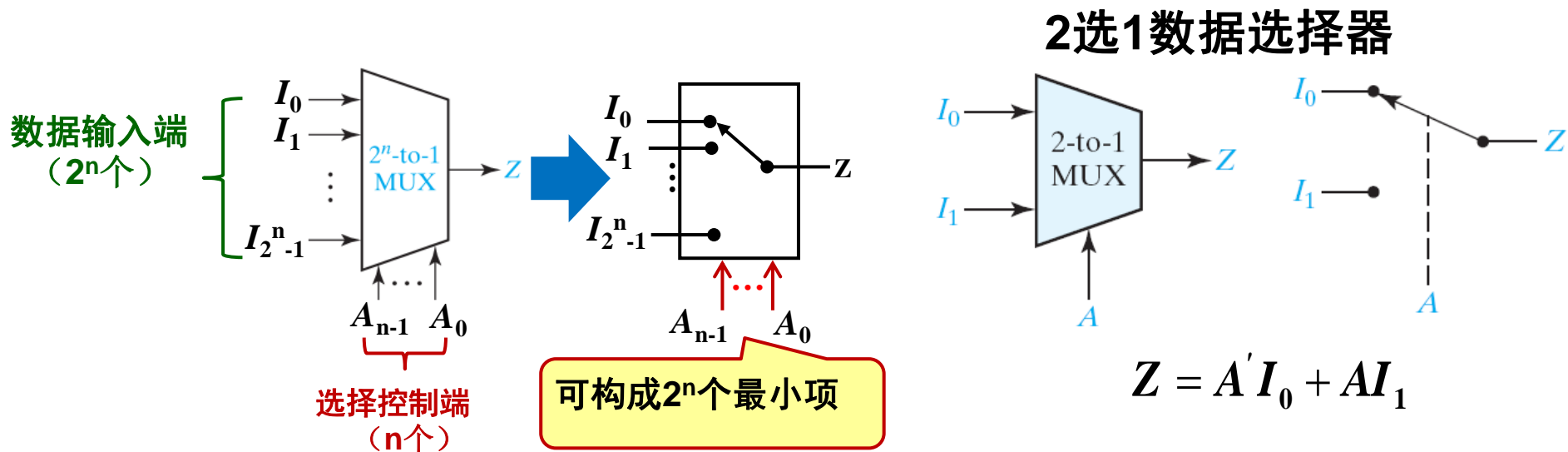
Unit 7 组合逻辑元件

- 多路复用器(multiplexers)
- 三态器件(Three-state Buffer)
- 译码器(Decoders)
- 编码器(Encoders)
- 奇偶校验器
- 比较器
- 只读存储器(ROM)
- 利用MSI设计组合逻辑电路

集成电路的分类

分类	单芯片内集成的逻辑门数量	集成内容	器件封装	需要掌握的内容
小规模 (SSI)				
中规模 (MSI)				
大规模 (LSI)				
超大规模 (VLSI)				

数据选择器/多路开关



$$Z = \sum_{k=0}^{2^n-1} m_k I_k$$

控制端最小项 m_k 的序号 K , 指向了第 K 路数据输入端 I_k 。

m_k —— n 个控制变量的最小项

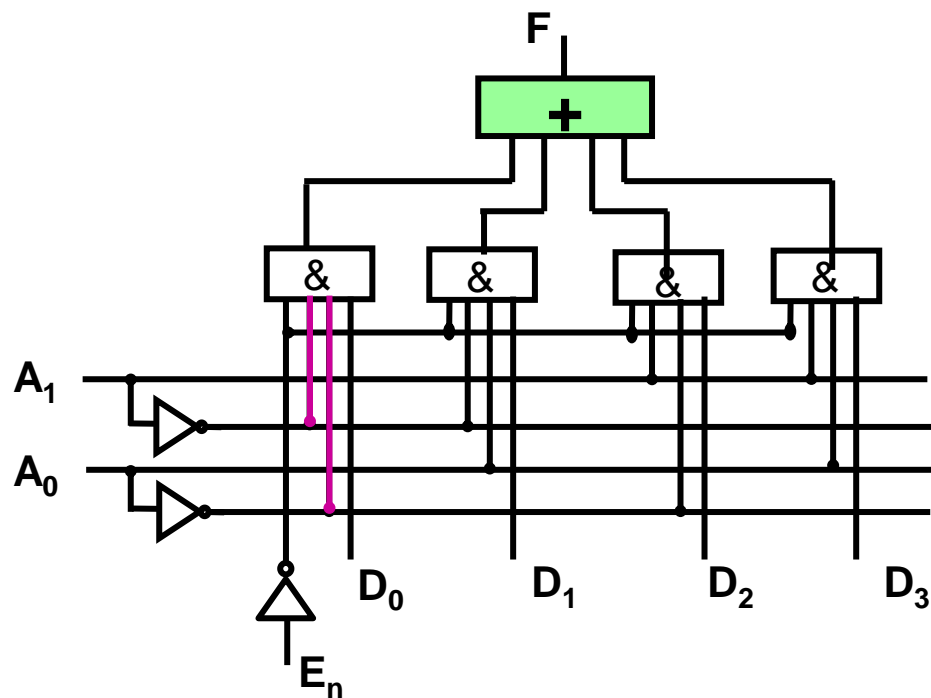
I_k —— 第 k 路数据输入

数据选择器的功能:

- ① 从多路输入中选择一个送往输出端 (2^n 选1);
- ② 选择哪一路输入送到输出端由控制信号决定;

用途: 实现多通道的数据传送;

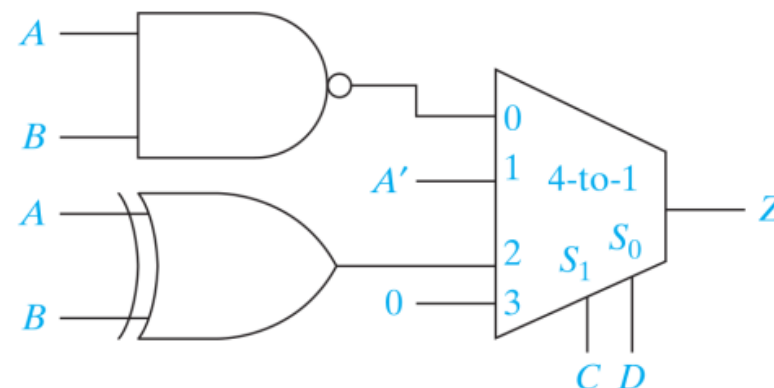
4选1数据选择器



E_n	A_1	A_0	F
1	×	×	0
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3

功能表

■ 典型应用——实现常规逻辑函数



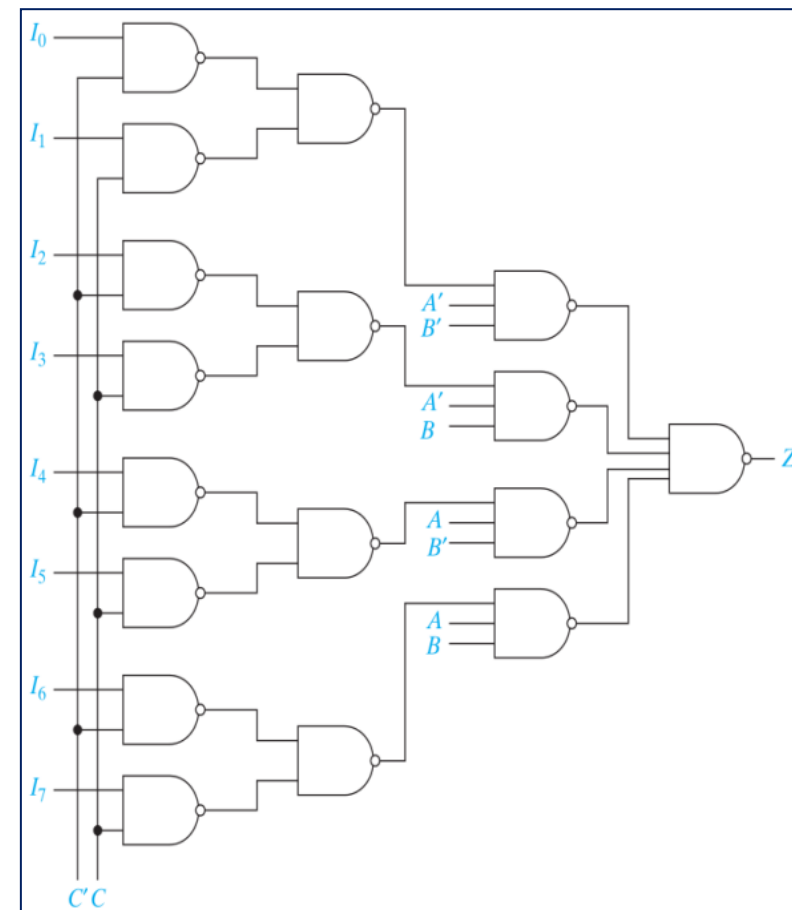
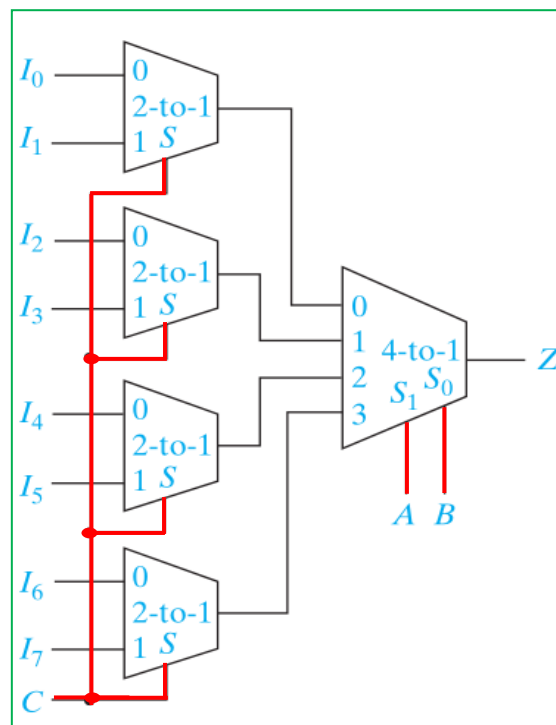
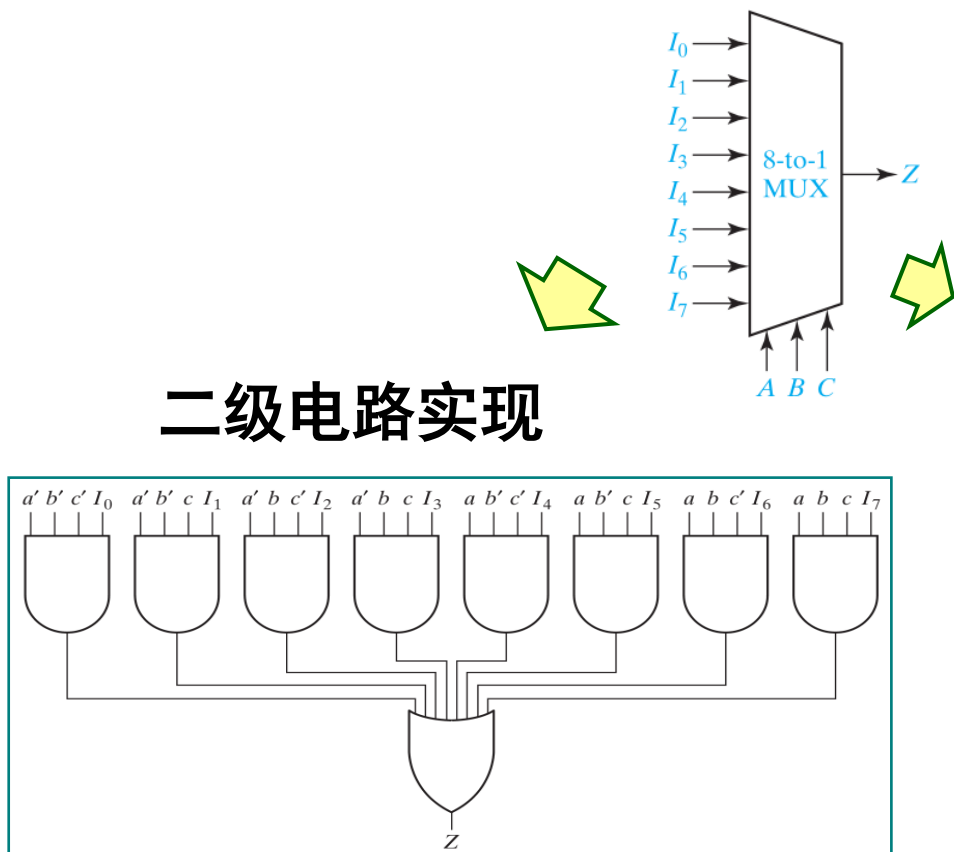
$$F = \bar{E}_n (D_0 \bar{A}_1 \bar{A}_0 + D_1 \bar{A}_1 A_0 + D_2 A_1 \bar{A}_0 + D_3 A_1 A_0)$$

8选1数据选择器

单一逻辑门实现

数据选择器级联实现

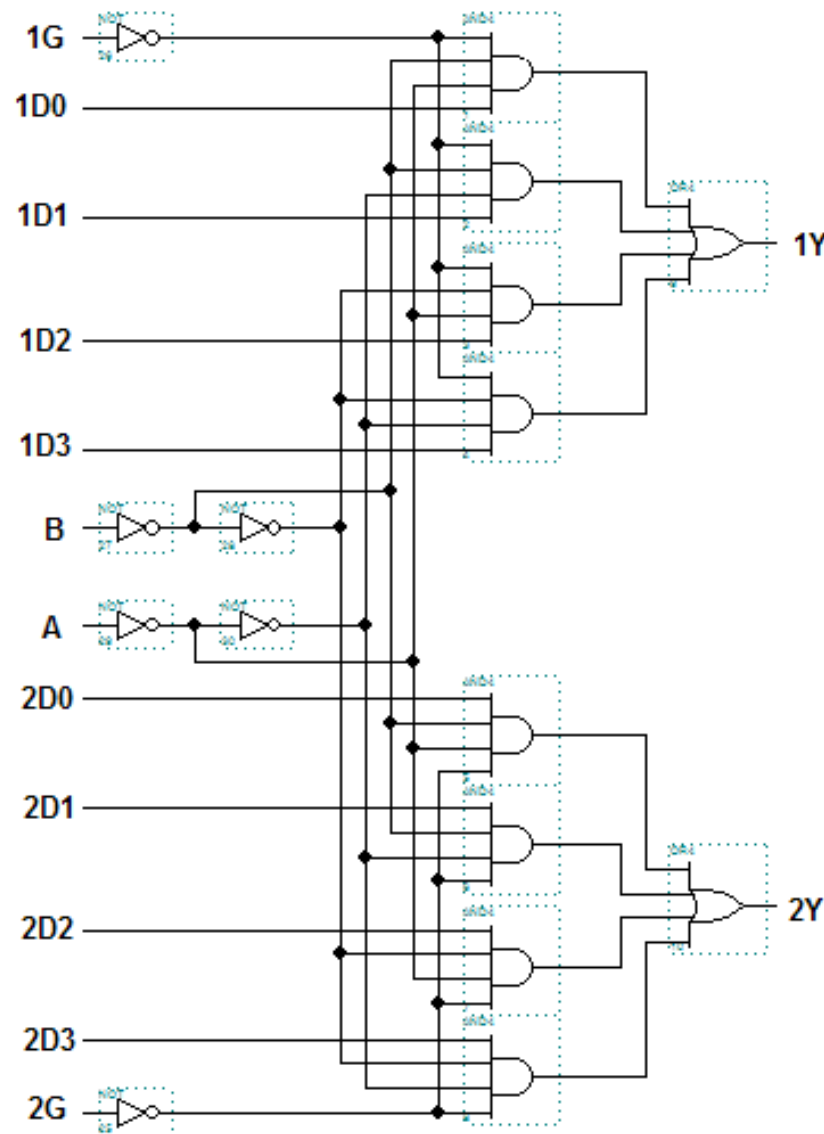
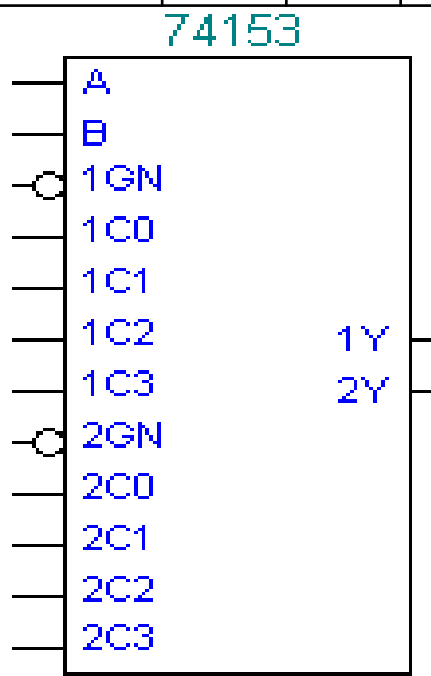
二级电路实现



$$Z = A'B'C'I_0 + A'B'CI_1 + A'BC'I_2 + A'BCI_3 + AB'C'I_4 + AB'CI_5 + ABC'I_6 + ABCI_7$$

双4选1典型器件74LS153

1Gn	2Gn	A	B	1Y	2Y
1	1	×	×	0	0
0	0	0	0	1C ₀	2C ₀
0	0	0	1	1C ₁	2C ₁
0	0	1	0	1C ₂	2C ₂
0	0	1	1	1C ₃	2C ₃



Unit 7 组合逻辑元件

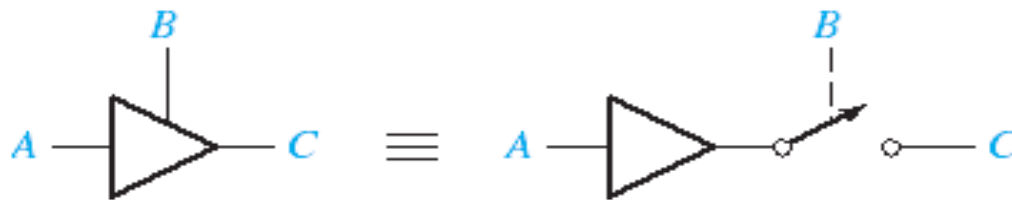
- 多路复用器(multiplexers)
- 三态器件(Three-state Buffer)
- 译码器(Decoders)
- 编码器(Encoders)
- 奇偶校验器
- 比较器
- 只读存储器(ROM)
- 利用MSI设计组合逻辑电路

三态门(Three-State Buffers)

三态——

- 0
- 1
- Z: 高阻态

- 包括三态恒等门、三态非门、三态与非门等，**缓冲器**（驱动门）。
- 用途之一：可用来增强输出驱动能力



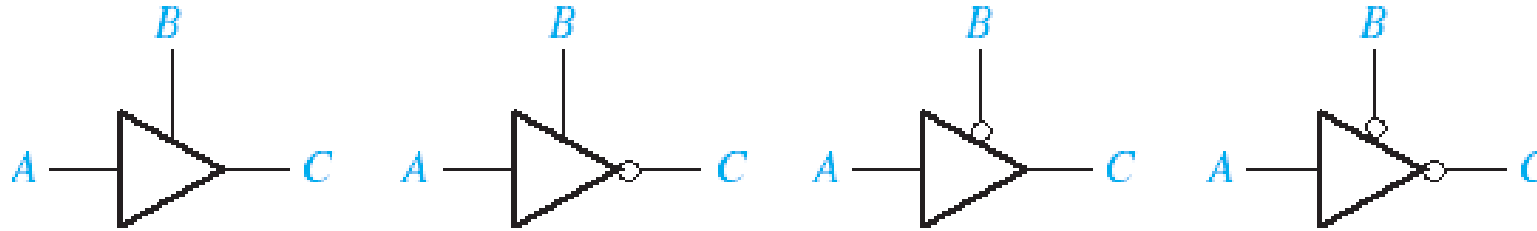
三态门（恒等）

B: 使能端，高电平有效

真值表

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

三态门 (Three-State Buffers)



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

(a)

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

(b)

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

(c)

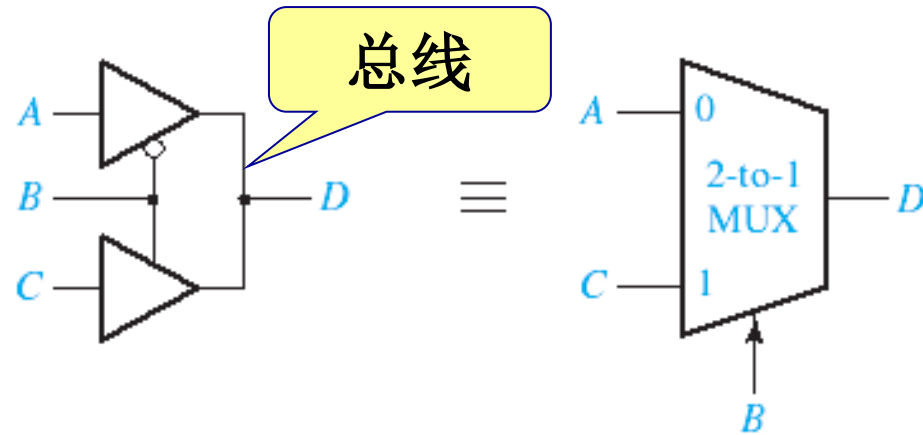
B	A	C
0	0	1
0	1	0
1	0	Z
1	1	Z

(d)

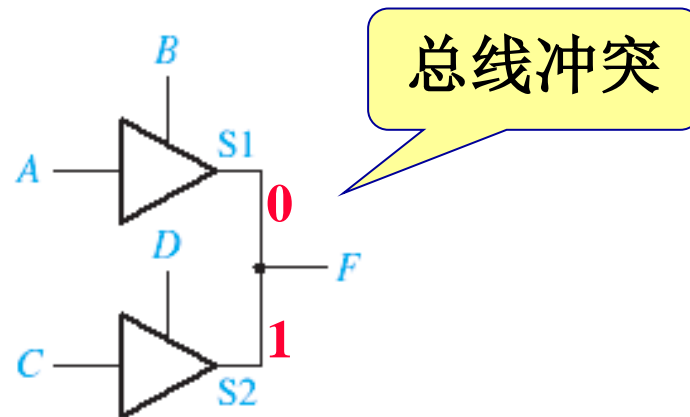
高阻态：电阻很大，相当于开路

高阻态相当于该门同与它连接的电路处于断开的状态。（实际电路中不可能去断开它）

三态门

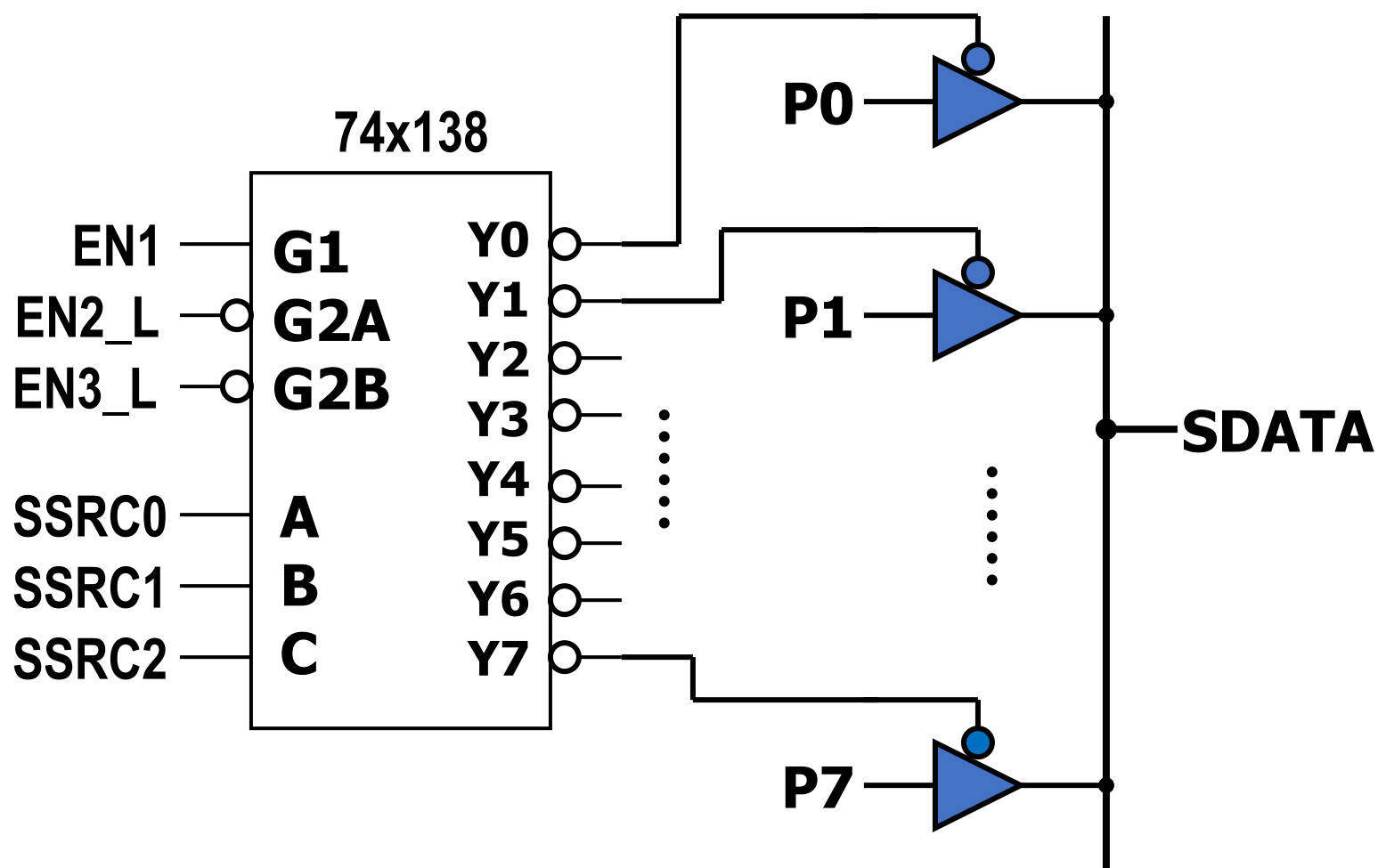


$$D = B'A + BC$$



	S_2			
S_1	X	0	1	Z
X	X	X	X	X
0	X	0	X	0
1	X	X	1	1
Z	X	0	1	Z

三态门

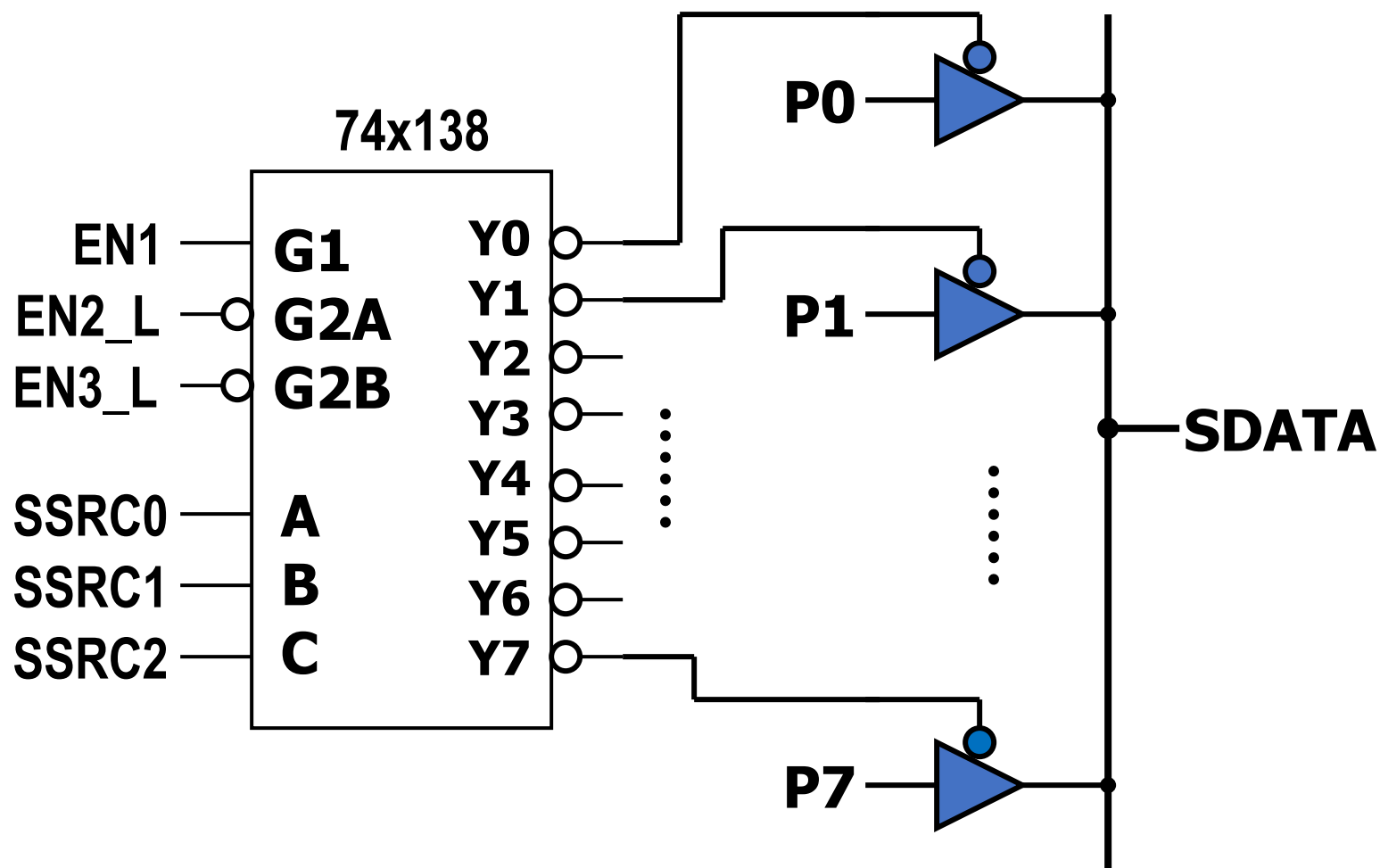


三态器件允许多个信号源共享单个“总线”（同线），但线上每次仅一个器件“谈话”

假如不是全部EN线有效，则没有一个三态缓冲器能被使能，此时SDATA上的逻辑值是“未定义”，悬空信号的实际电压值依赖于电路细节。

图 8个信号源共享1根三态总线/同线

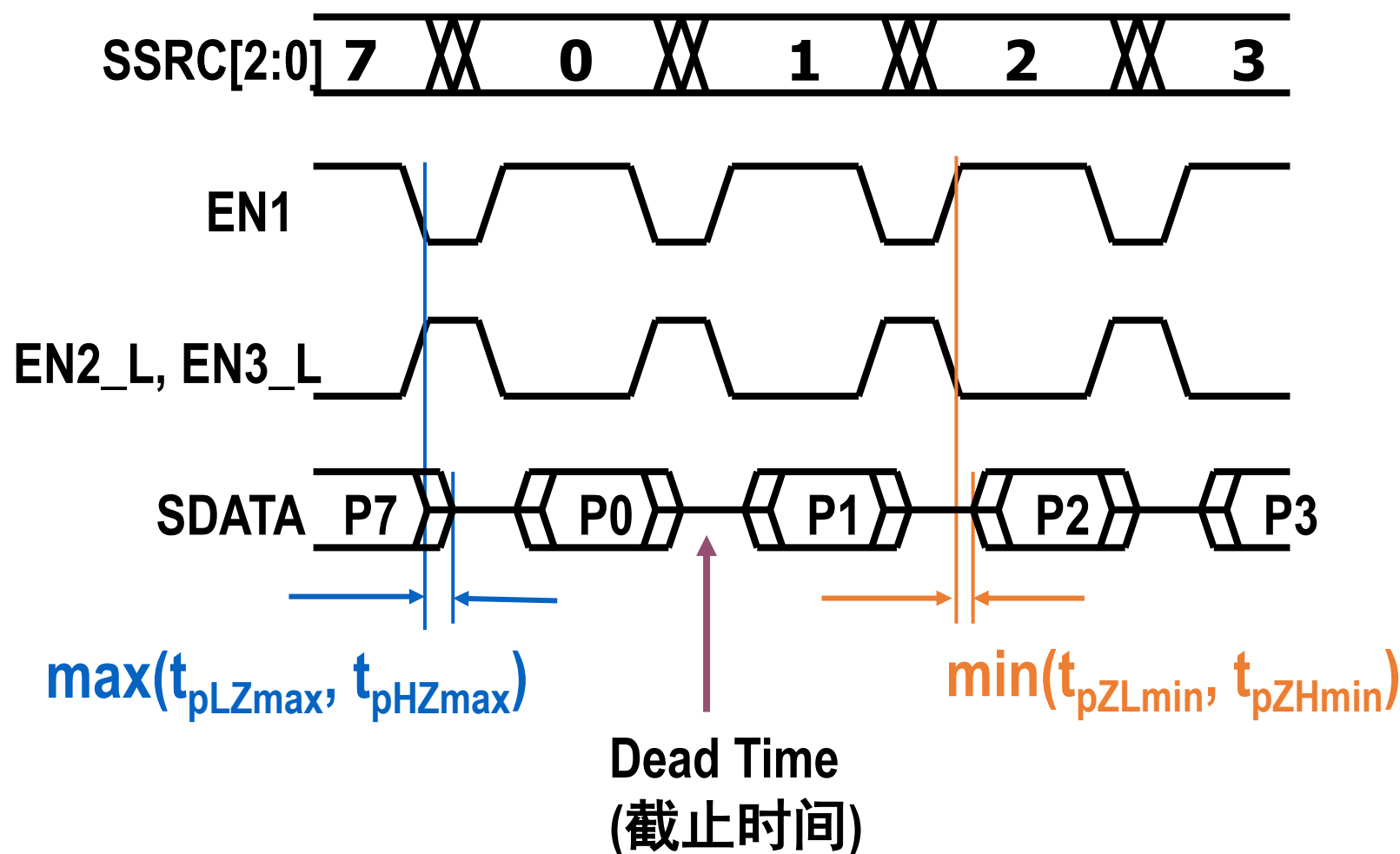
三态门



对典型的三态器件，**进入高阻态比离开高阻态快**。也会使得系统中产生冲突（fighting）

图 8个信号源共享1根三态总线

三态门

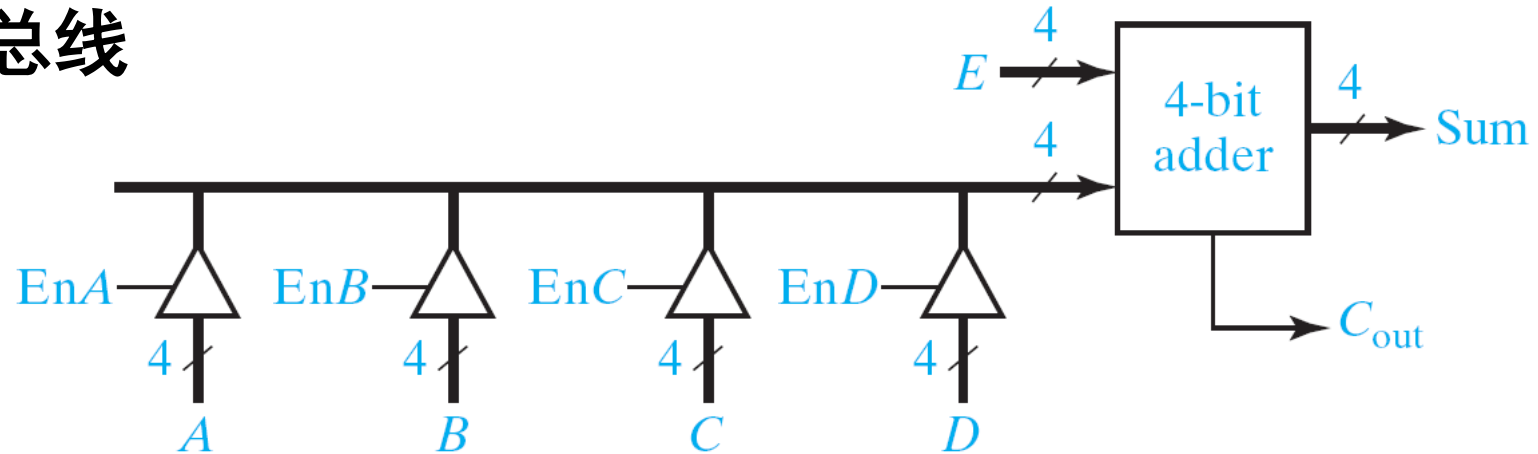


使用三态器件唯一真正安全的方法是**设计逻辑控制**，以保证同线上有一段**截止时间**（dead time），在此期间不应有任何器件驱动同线。

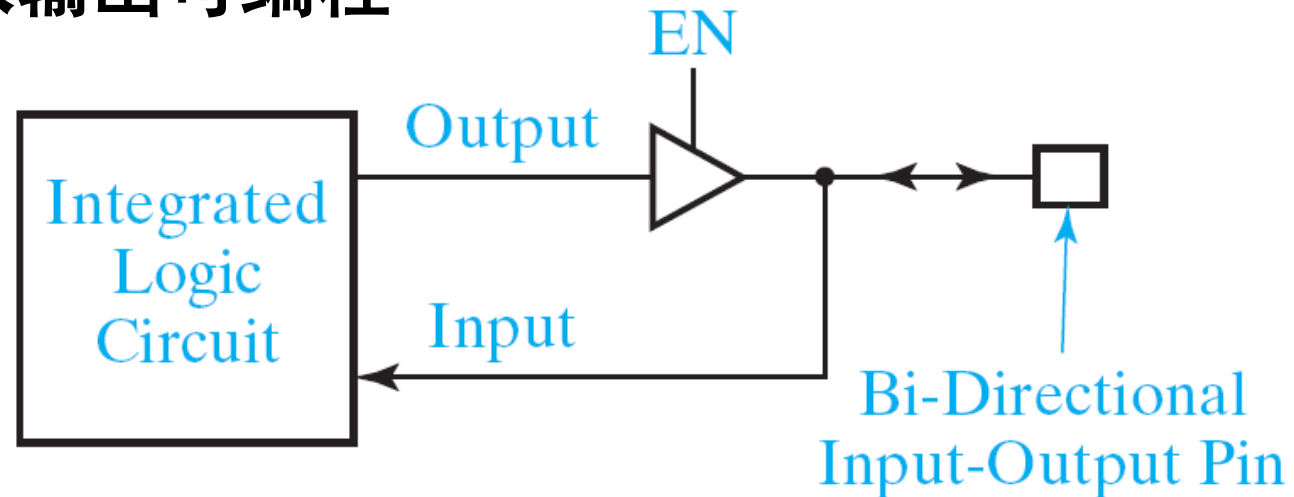
图 三态同线定时图

三态门应用

■ 三态总线

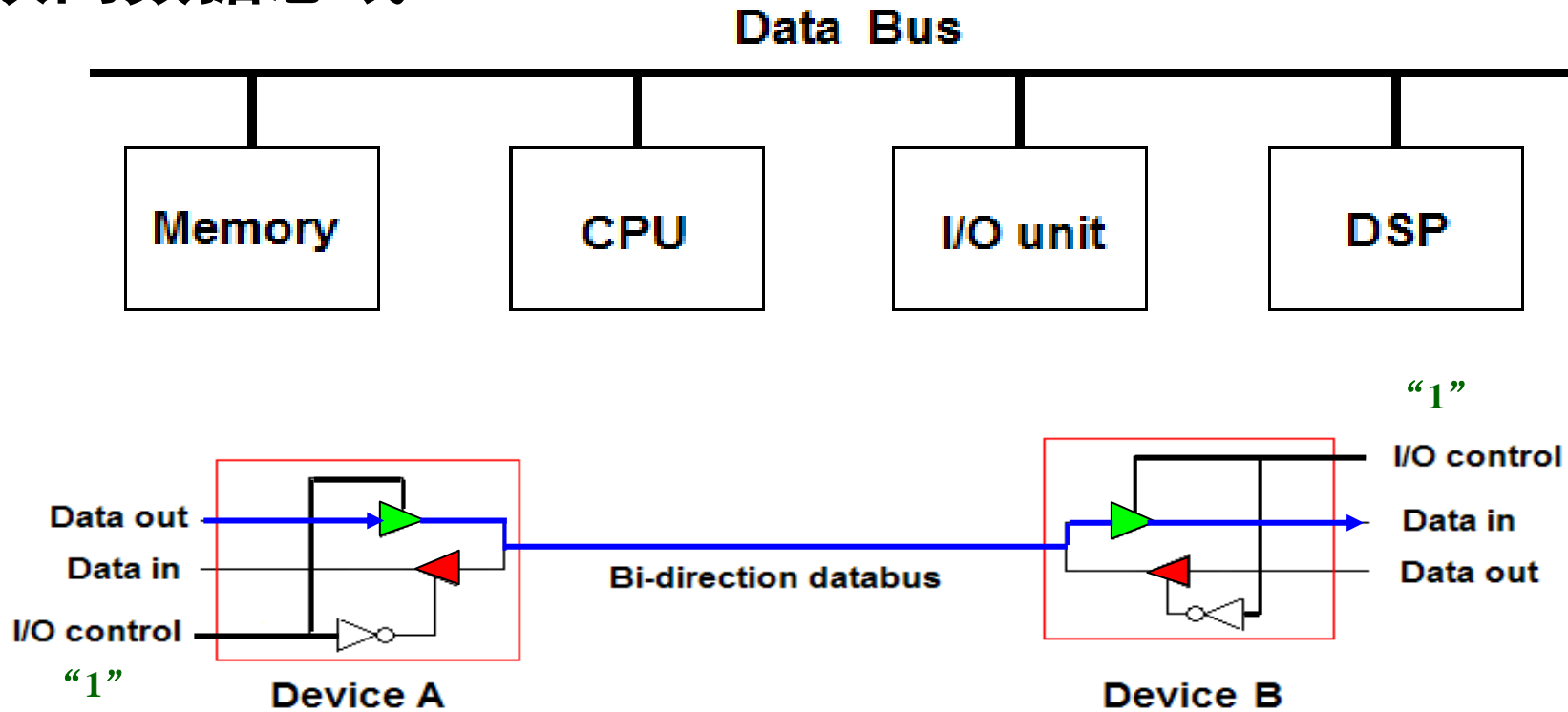


■ 管脚输入输出可编程



三态门应用——续

■ 双向数据总线



三态门应用——续

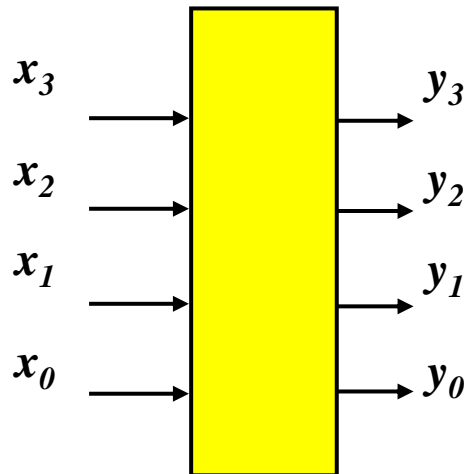
内存里的一个存储单元

- 读写控制线处于**低电位**时，可以写入；
- 读写控制线处于**高电位**时，可以读出
- 但是不读不写，就要用高阻态

三态门的应用

- $X=X_3X_2X_1X_0$ 为8421BCD码，设计一个MOD 5选择电路，要求选择那些能被5整除的数输出。

①真值表(F为控制信号)

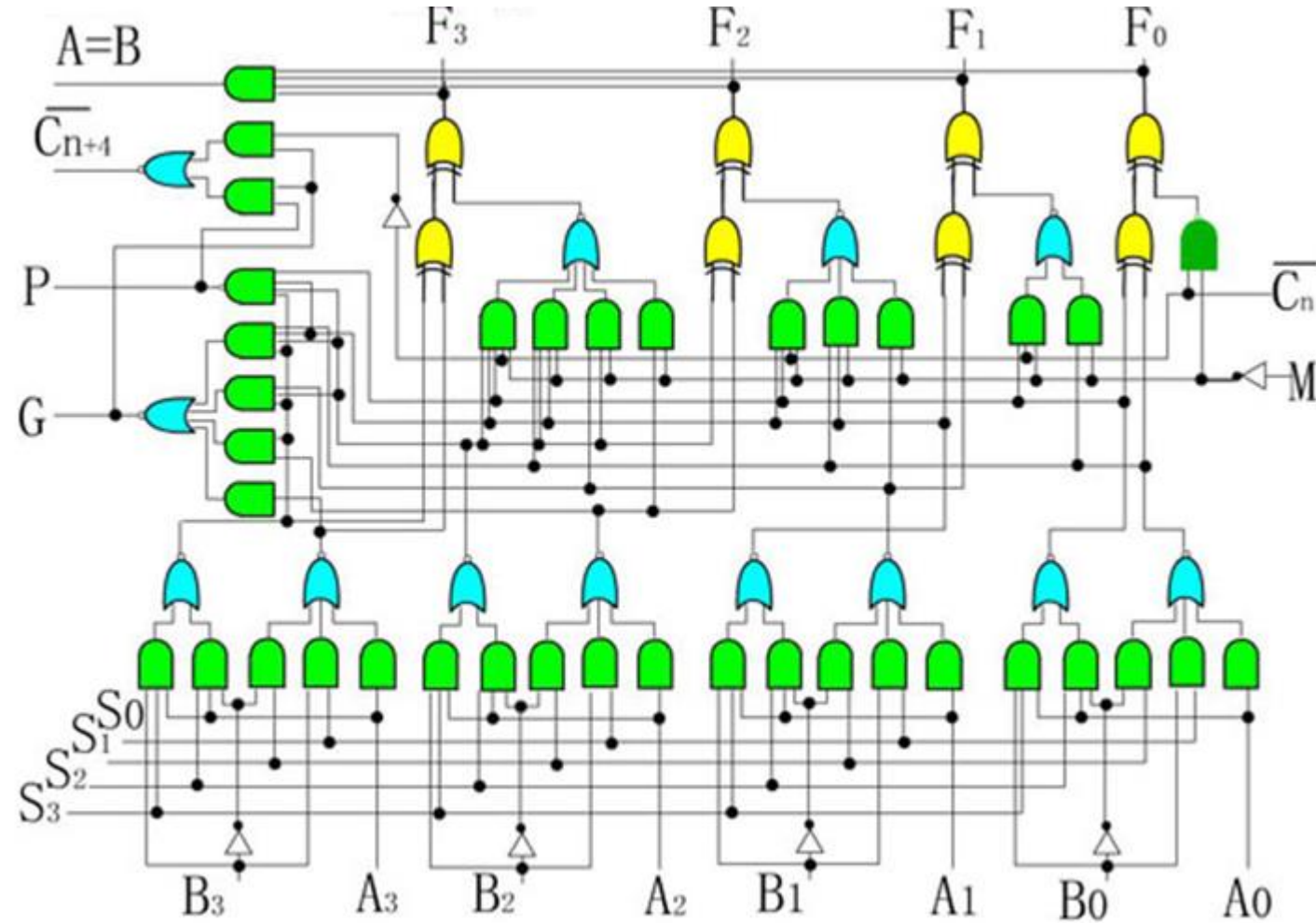


X_3	X_2	X_1	X_0	F	X_3	X_2	X_1	X_0	F
0	0	0	0	1	1	0	0	0	0
0	0	0	1	0	1	0	0	1	0
0	0	1	0	0	1	0	1	0	×
0	0	1	1	0	1	0	1	1	×
0	1	0	0	0	1	1	0	0	×
0	1	0	1	1	1	1	0	1	×
0	1	1	0	0	1	1	1	0	×
0	1	1	1	0	1	1	1	1	×

Unit 7 组合逻辑元件

- 多路复用器(multiplexers)
- 三态器件(Three-state Buffer)
- 译码器(Decoders)
- 编码器(Encoders)
- 奇偶校验器
- 比较器
- 只读存储器(ROM)
- 利用MSI设计组合逻辑电路

ALU



$M=0$, addition

$M=1$, logic operation

三态门

像逻辑门一样，几个独立的三态缓冲器可以封装在单个SSI IC中，他们仍具有独立的使能输入。

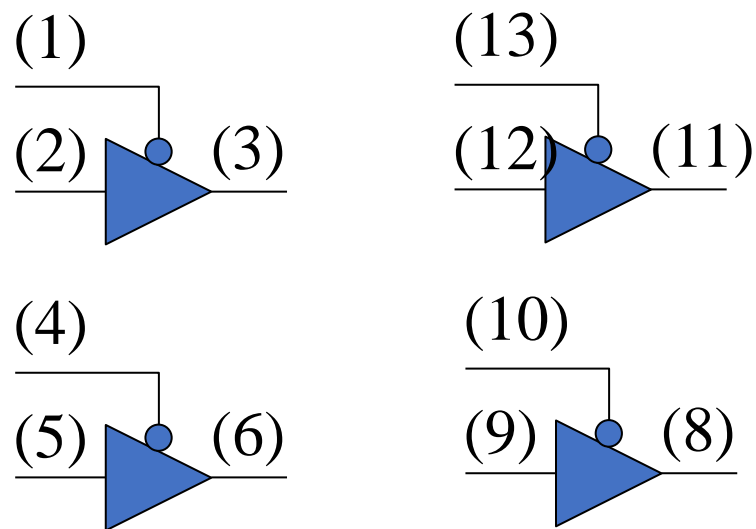
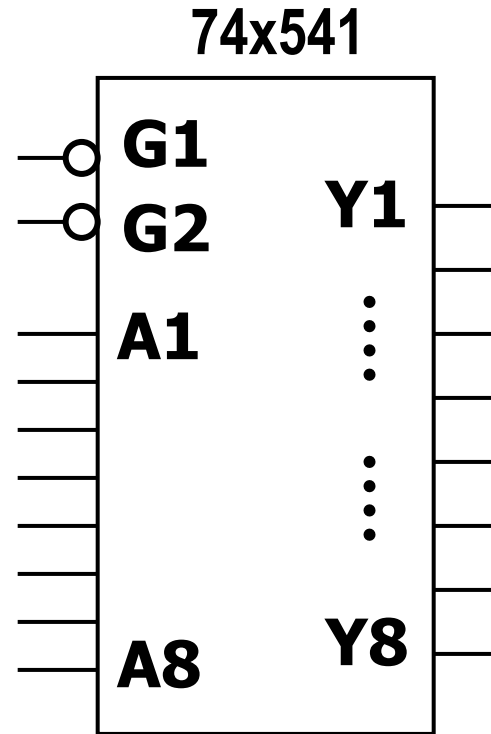


图 74x125三态缓冲器的管脚引线

标准SSI和MSI三态缓冲器

在宽总线应用中为了减少封装尺寸，多数常用的MSI部件包含带有公共使能输入的多个三态缓冲器。



三态缓冲器74x541

标准SSI和MSI三态缓冲器

总线收发器包含三态缓冲器对，每对引脚之间以相反方向连接，所以数据可以双向传输。

总线收发器常用于两个双向总线之间，根据G_L和DIR的状态，控制操作模式。

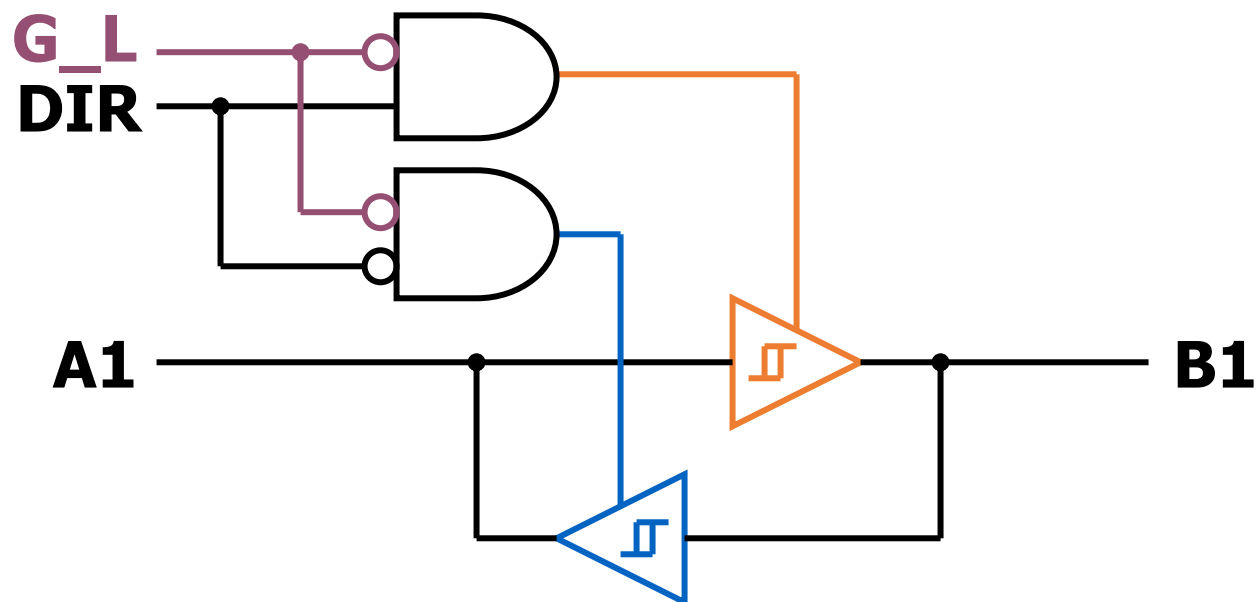


图 总线收发器