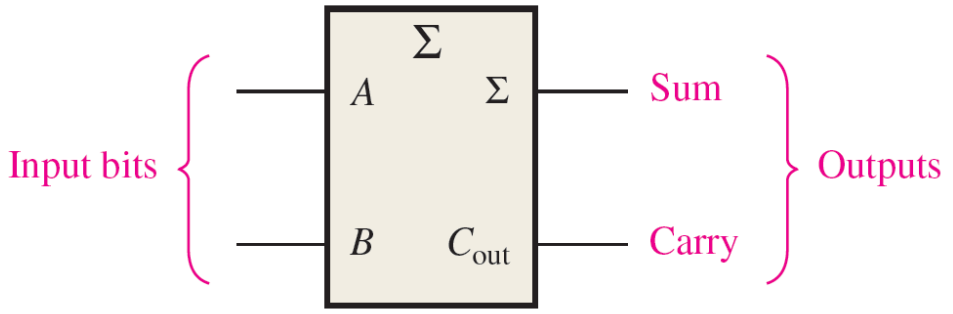
Half Adder

- The half-adder accepts two binary digits on its inputs and produces a sum bit and a carry bit.

A	B	C_{out}	Σ
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

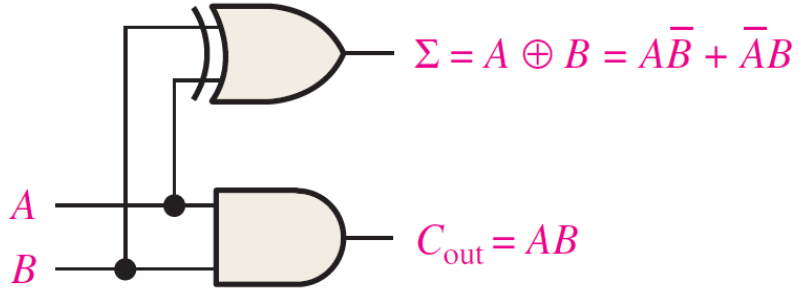


- C_{out} is a 1 only when both A and B are 1s.

$$C_{out} = AB$$

- Σ is a 1 only if A and B are not equal.

$$\Sigma = A \oplus B$$

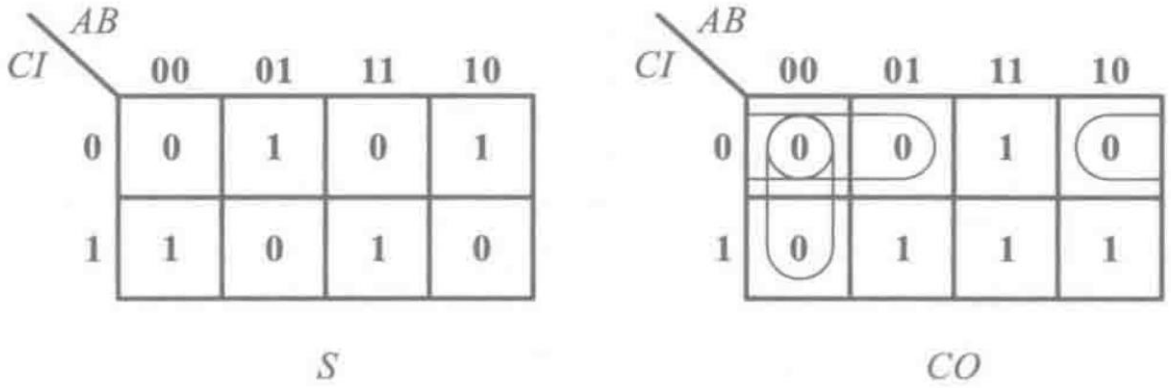


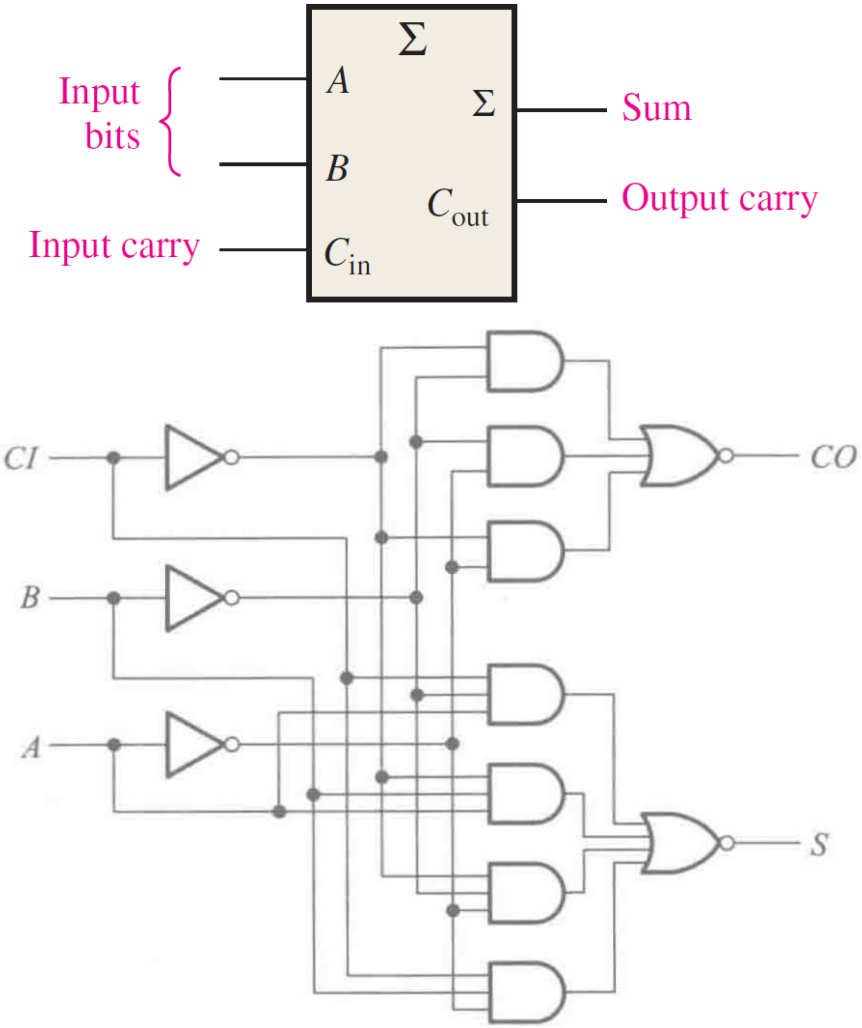
Full Adder

The full-adder (FA) accepts two bits and a carry and generates a sum and a carry.

A	B	C_{in}	C_{out}	Σ
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

$$\begin{cases} S = (A'B'CI' + AB'CI + A'BCI + ABCI')' \\ CO = (A'B' + B'CI' + A'CI')' \end{cases}$$

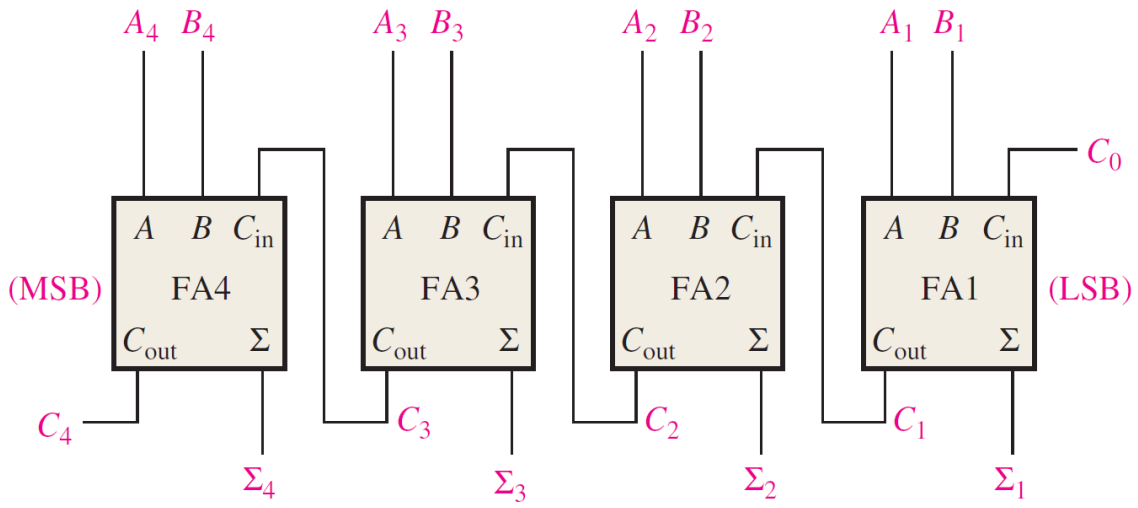


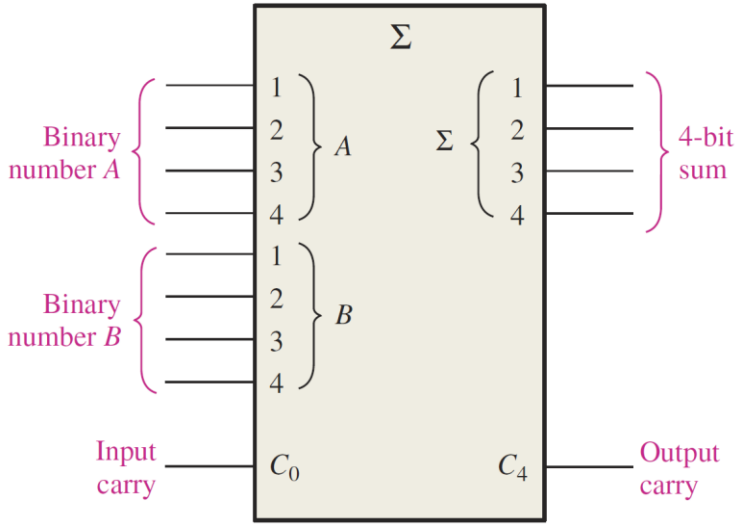


74LS183

Ripple Carry Adder

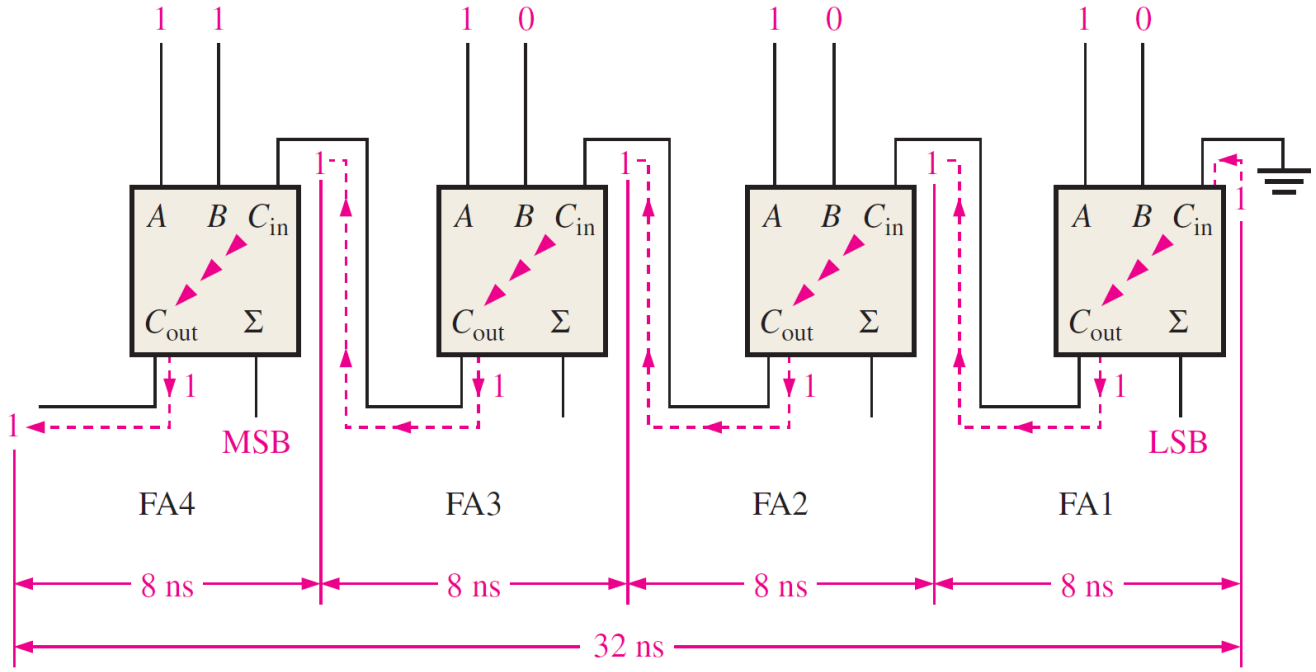
- To add binary numbers with more than one bit, more full-adders are connected





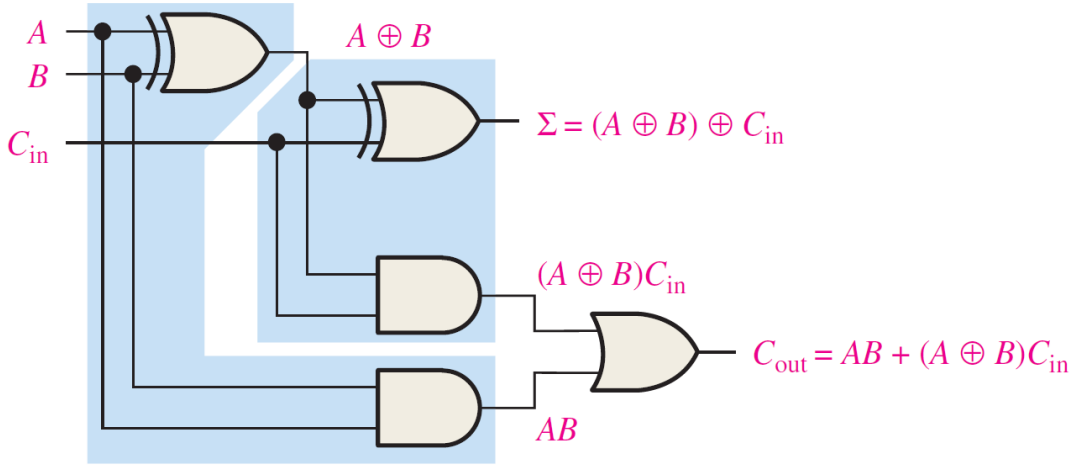
Ripple Carry Adder

- The sum and the output carry of any stage cannot be produced until the input carry occurs.



Full Adder – Another Look

A	B	C_{in}	C_{out}	Σ
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



$$\Sigma = (A \oplus B) \oplus C_{in}$$

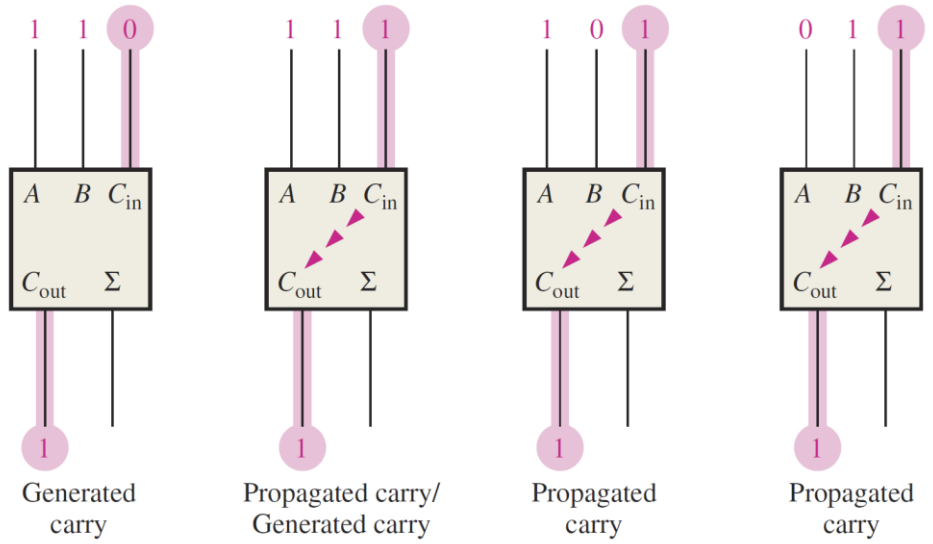
- A carry is generated only when both input bits are 1s.

$$C_g = AB$$

- A carry is propagated when either or both of the input bits are 1s.

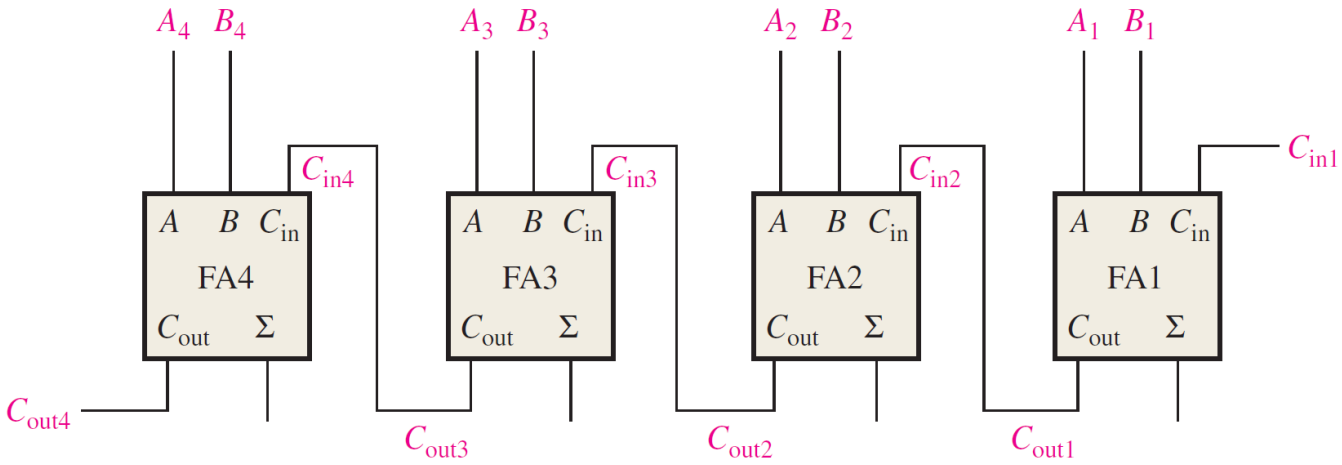
$$C_p = A \oplus B \text{ or } C_p = A + B$$

$$C_{out} = C_g + C_pC_{in} = AB + (A \oplus B)C_{in} = AB + (A+B)C_{in}$$



Look-Ahead Carry Adder

- The look-ahead carry adder produces C_{out} by either carry generation or carry propagation, i.e., $C_{out} = C_g + C_pC_{in}$
- C_g and C_p for each stage are immediately available.



Full-adder 4

$C_{g4} = A_4B_4$
$C_{p4} = A_4 + B_4$

Full-adder 3

$C_{g3} = A_3B_3$
$C_{p3} = A_3 + B_3$

Full-adder 2

$C_{g2} = A_2B_2$
$C_{p2} = A_2 + B_2$

Full-adder 1

$C_{g1} = A_1B_1$
$C_{p1} = A_1 + B_1$

Look-Ahead Carry Adder

Full-adder 1:

$$C_{out1} = C_{g1} + C_{p1}C_{in1}$$

- Based on $C_{out} = C_g + C_pC_{in}$, we can build the look-ahead carry adder
- C_g and C_p for each stage are immediately available.
- It only takes $3t_{pd}$ to get the C_{out}

Full-adder 2:

$$C_{in2} = C_{out1}$$
$$C_{out2} = C_{g2} + C_{p2}C_{in2} = C_{g2} + C_{p2}C_{out1} = C_{g2} + C_{p2}(C_{g1} + C_{p1}C_{in1})$$
$$= C_{g2} + C_{p2}C_{g1} + C_{p2}C_{p1}C_{in1}$$

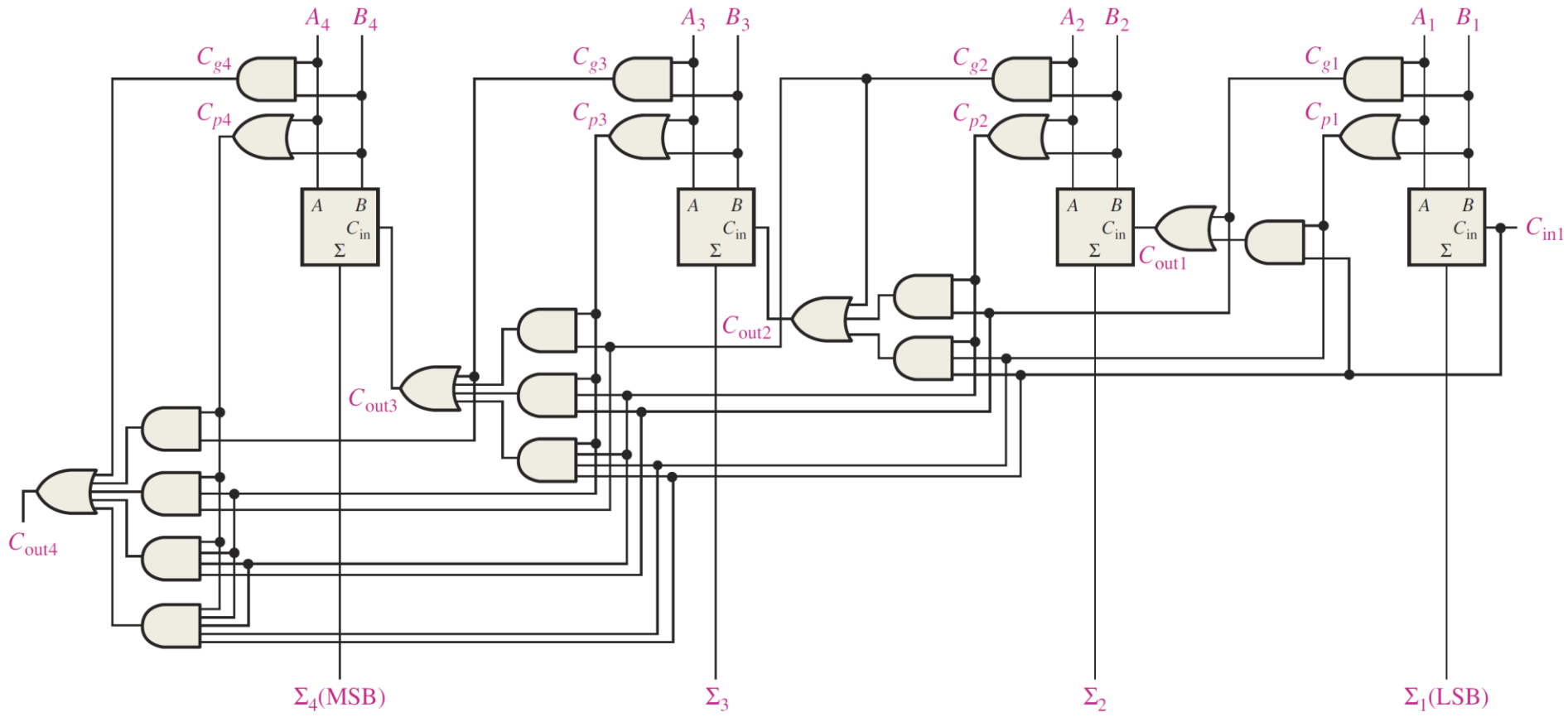
Full-adder 3:

$$C_{in3} = C_{out2}$$
$$C_{out3} = C_{g3} + C_{p3}C_{in3} = C_{g3} + C_{p3}C_{out2} = C_{g3} + C_{p3}(C_{g2} + C_{p2}C_{g1} + C_{p2}C_{p1}C_{in1})$$
$$= C_{g3} + C_{p3}C_{g2} + C_{p3}C_{p2}C_{g1} + C_{p3}C_{p2}C_{p1}C_{in1}$$

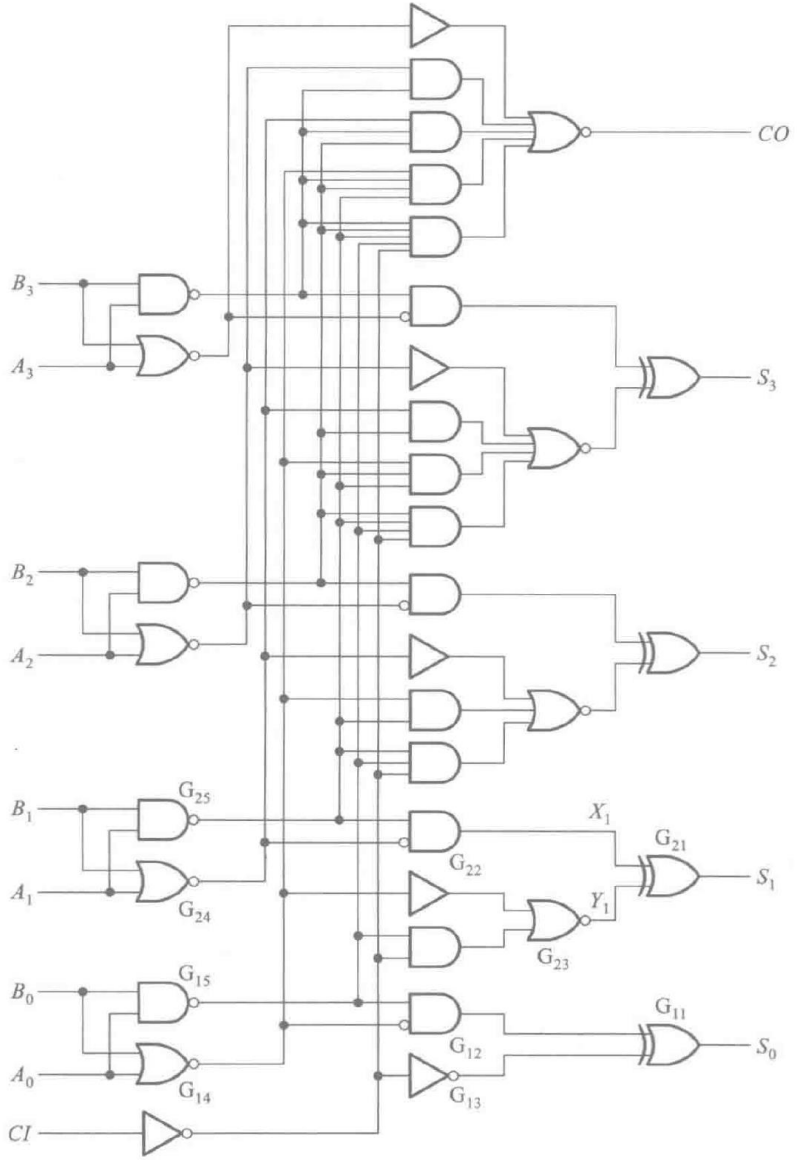
Full-adder 4:

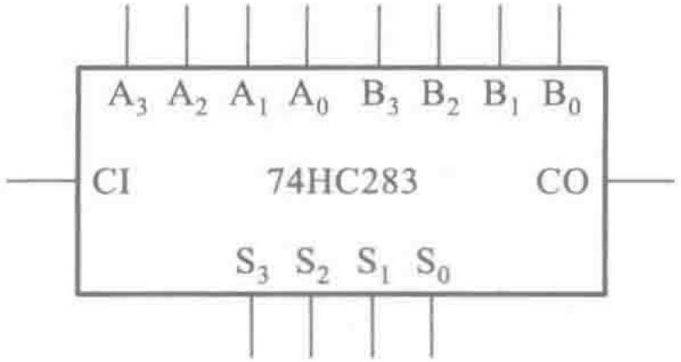
$$C_{in4} = C_{out3}$$
$$C_{out4} = C_{g4} + C_{p4}C_{in4} = C_{g4} + C_{p4}C_{out3}$$
$$= C_{g4} + C_{p4}(C_{g3} + C_{p3}C_{g2} + C_{p3}C_{p2}C_{g1} + C_{p3}C_{p2}C_{p1}C_{in1})$$
$$= C_{g4} + C_{p4}C_{g3} + C_{p4}C_{p3}C_{g2} + C_{p4}C_{p3}C_{p2}C_{g1} + C_{p4}C_{p3}C_{p2}C_{p1}C_{in1}$$

Look-Ahead Carry Adder



Look-Ahead Carry Adder





Reading materials

- Chapter 6 of Floyd book
- Chapter 4 of 阎石 book