

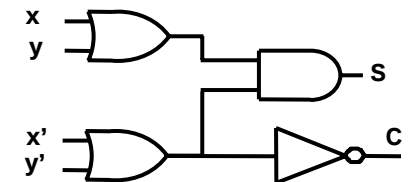
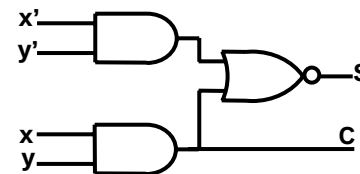
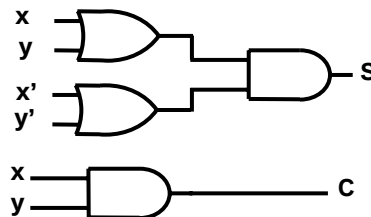
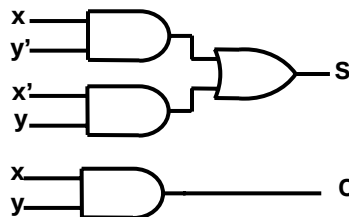
# Half Adder

- Design a logic circuit to add two bits and produce a sum bit and a carry bit. Two inputs and two outputs are needed. Let us call the inputs  $x$  and  $y$ , and the outputs  $S$  and  $C$ .

| $x$ | $y$ | $S$ | $C$ |
|-----|-----|-----|-----|
| 0   | 0   | 0   | 0   |
| 0   | 1   | 1   | 0   |
| 1   | 0   | 1   | 0   |
| 1   | 1   | 0   | 1   |

$$\begin{aligned}
 S &= x \oplus y = x'y + xy' \\
 &= (x'y' + xy)' \\
 &= (x+y)(x'+y')
 \end{aligned}$$

$$\begin{aligned}
 C &= xy \\
 &= (x' + y')' \\
 S &= (C + x'y')'
 \end{aligned}$$



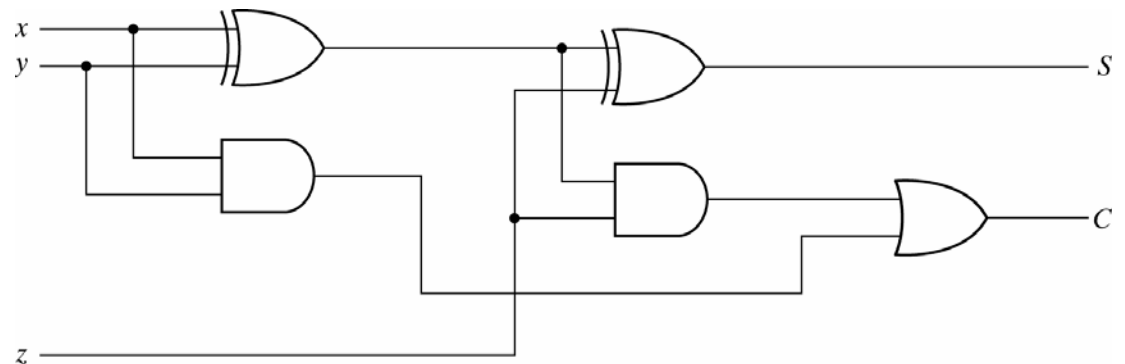
# Full-Adder

- Design a logic circuit that adds two bits and a carry in bit and produce a sum bit and a carry out bit. Three inputs and two outputs are needed. Let us call the inputs  $x$ ,  $y$  and  $z$ , and the outputs  $S$  and  $C$ .

| $x$ | $y$ | $z$ | $S$ | $C$ |
|-----|-----|-----|-----|-----|
| 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   |

$$\begin{aligned} S &= \Sigma(1,2,4,7) \\ &= x'y'z + x'yz' + xy'z' + xyz = (x'y' + xy)z + (x'y + xy')z' \\ &= (x \oplus y)'z + (x \oplus y)z' = (x \oplus y) \oplus z \end{aligned}$$

$$\begin{aligned} C &= \Sigma(3,5,6,7) \\ &= x'yz + xy'z + xyz' + xyz = x'yz + xy'z + xy \\ &= (x'y + xy')z + xy = (x \oplus y)z + xy \end{aligned}$$

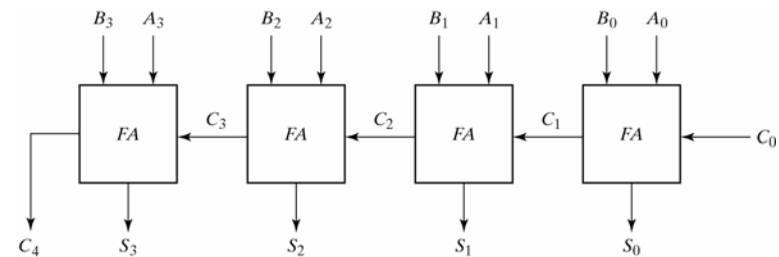


# 4-Bit Binary Adder

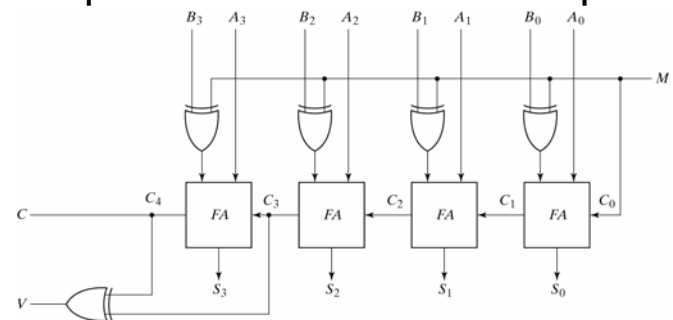
- To add two 4-bit binary number, we proceed as shown in the table. The table shows the role of carry-in and carry-out.

|   |   |   |   |      |
|---|---|---|---|------|
| 0 | 1 | 1 | 0 | Ci   |
| 1 | 0 | 1 | 1 | Ai   |
| 0 | 0 | 1 | 1 | Bi   |
| 1 | 1 | 1 | 0 | Si   |
| 0 | 0 | 1 | 1 | Ci+1 |

4-bit binary parallel adder can be implemented in integrated circuit form by cascading 4 full adders as shown next. The disadvantage of this adder is the possible slowing down of the addition due to the carry propagation time.



The 4-bit parallel adder can be modified to work as 4-bit parallel adder/subtractor by including 4 exclusive-OR gates to provide the 1's complement of B and adding 1 from the M input to make it the 2's complement.



# Carry Propagation

- The carry propagate ( $P_i$ ) and carry generate ( $G_i$ ) variables are shown on the full adder logic circuit. The carries  $C_1$ ,  $C_2$ , and  $C_3$  can be expressed in SOP form as functions of  $C_0$  and the different ( $P_i$ ) and ( $G_i$ ) as follows:

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

$$S_i = P_i \oplus C_i, \quad C_{i+1} = G_i + P_i C_i$$

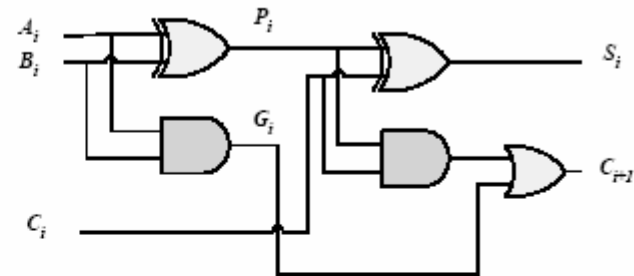
Therefore:

$$C_1 = G_0 + P_0 C_0$$

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

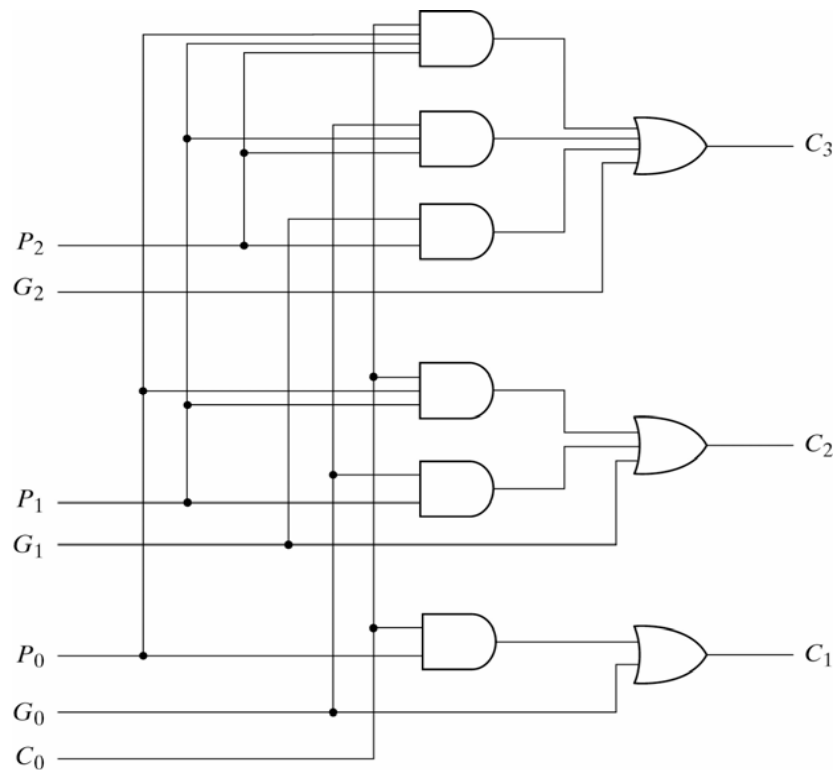
Similarly:

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



# The Look-Ahead Carry Adder

The logic diagram of the look-ahead generator is implemented in a two level form as shown in the following logic circuit.



The 4-Bit adder with the carry look-ahead circuit is implemented as shown in the following circuit.

