XOR, XNOR, & Binary Adders

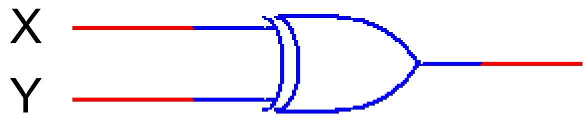
Digital Electronics

XOR, XNOR & Adders

This presentation will demonstrate

- The basic function of the exclusive OR (**XOR**) gate.
- The basic function of the exclusive NOR (**XNOR**) gate.
- How **XOR** and **XNOR** gates can be used to implement combinational logic design.
- How **XOR** gates can be using to design half and full adders.
- How full adders can be implemented with Small Scale Integration (SSI) and Medium Scale Integration (MSI) logic.
- How single bit half and full adders can be cascaded to make multi-bit adders.

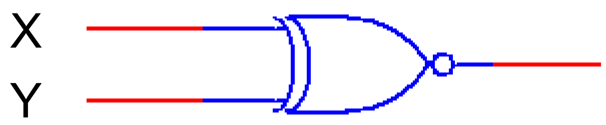
XOR Gate – Exclusive OR



$$Z = \overline{X}\,Y + X\,\overline{Y} = X \oplus Y$$

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	0

XNOR Gate – Exclusive NOR



$$Z = \overline{X}\,\overline{Y} + X\,Y = \overline{X \oplus Y}$$

X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	1

Logic Design with XOR & XNOR

Example

Algebraically manipulate the logic expression for F_1 so that XOR and XNOR gates can be used to implement the function. Other AOI gates can be used as needed.

$$F_1 = X\,\overline{Y}\,Z + \overline{X}\,Y\,Z + \overline{X}\,\overline{Y}\,\overline{Z} + X\,\overline{Y}\,Z$$

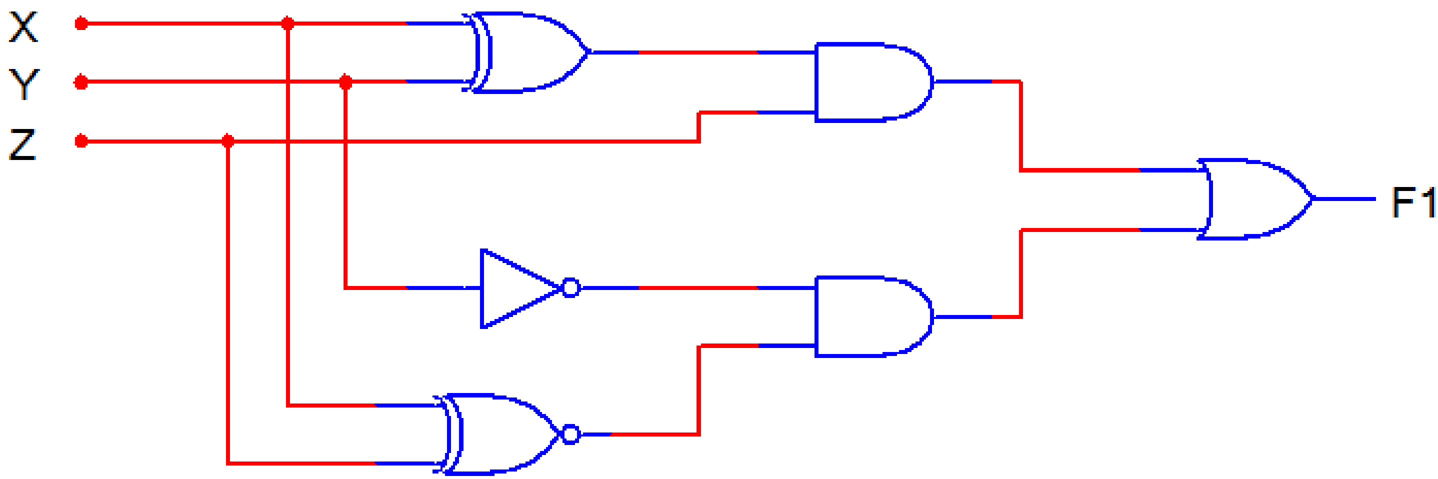
Logic Design with XOR & XNOR

Solution

$$F_1 = X\,\overline{Y}\,Z + \overline{X}\,Y\,Z + \overline{X}\,\overline{Y}\,\overline{Z} + X\,\overline{Y}\,Z$$

$$F_1 = Z\left(X\,\overline{Y} + \overline{X}\,Y\right) + \overline{Y}\left(\overline{X}\,\overline{Z} + X\,Z\right)$$

$$F_1 = Z\left(X \oplus Y\right) + \overline{Y}\left(\overline{X \oplus Z}\right)$$



Binary Addition

Carry

Single Bit Addition:

$$\begin{array}{r} 0 \\ +\ 0 \\ \hline 0 \end{array} \qquad \begin{array}{r} 0 \\ +\ 1 \\ \hline 1 \end{array} \qquad \begin{array}{r} 1 \\ +\ 0 \\ \hline 1 \end{array} \qquad \begin{array}{r} {\scriptstyle 1} \\ 1 \\ +\ 1 \\ \hline 1\ 0 \end{array}$$

Multiple Bit Addition:

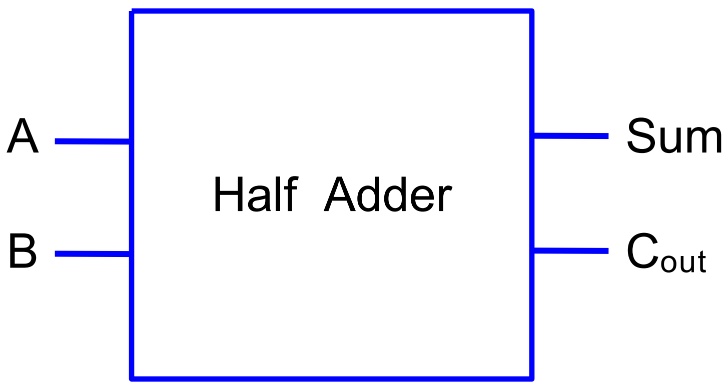
$$\begin{array}{r} 6 \\ +\ 3 \\ \hline 9 \end{array} \qquad \begin{array}{r} {\scriptstyle 1\ 1} \\ 0\ 1\ 1\ 0 \\ +\ 0\ 0\ 1\ 1 \\ \hline 1\ 0\ 0\ 1 \end{array}$$

$$\begin{array}{lcl} C_{out} & & C_{in} \\ 1 & & 1 \\ & 1 & A \\ + & 0 & B \\ \hline & 0 & Sum \end{array}$$

Two Types of Adders

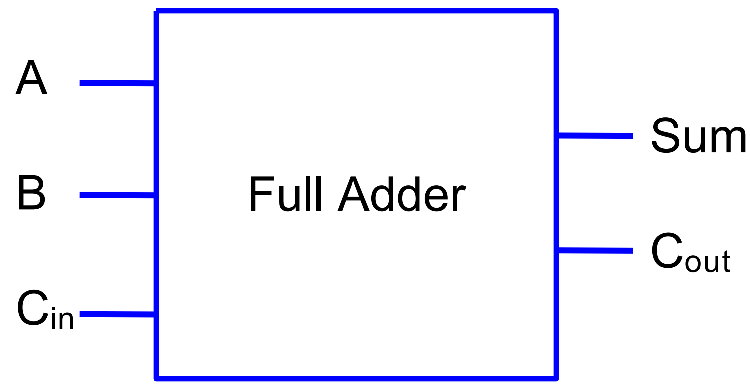
Half Adder

- 2 Inputs (A & B)
- 2 Outputs (Sum & C_{out})
- Used for LSB only



Full Adder

- 3 Inputs (A, B, C_{in})
- 2 Outputs (Sum & C_{out})
- Used for all other bits



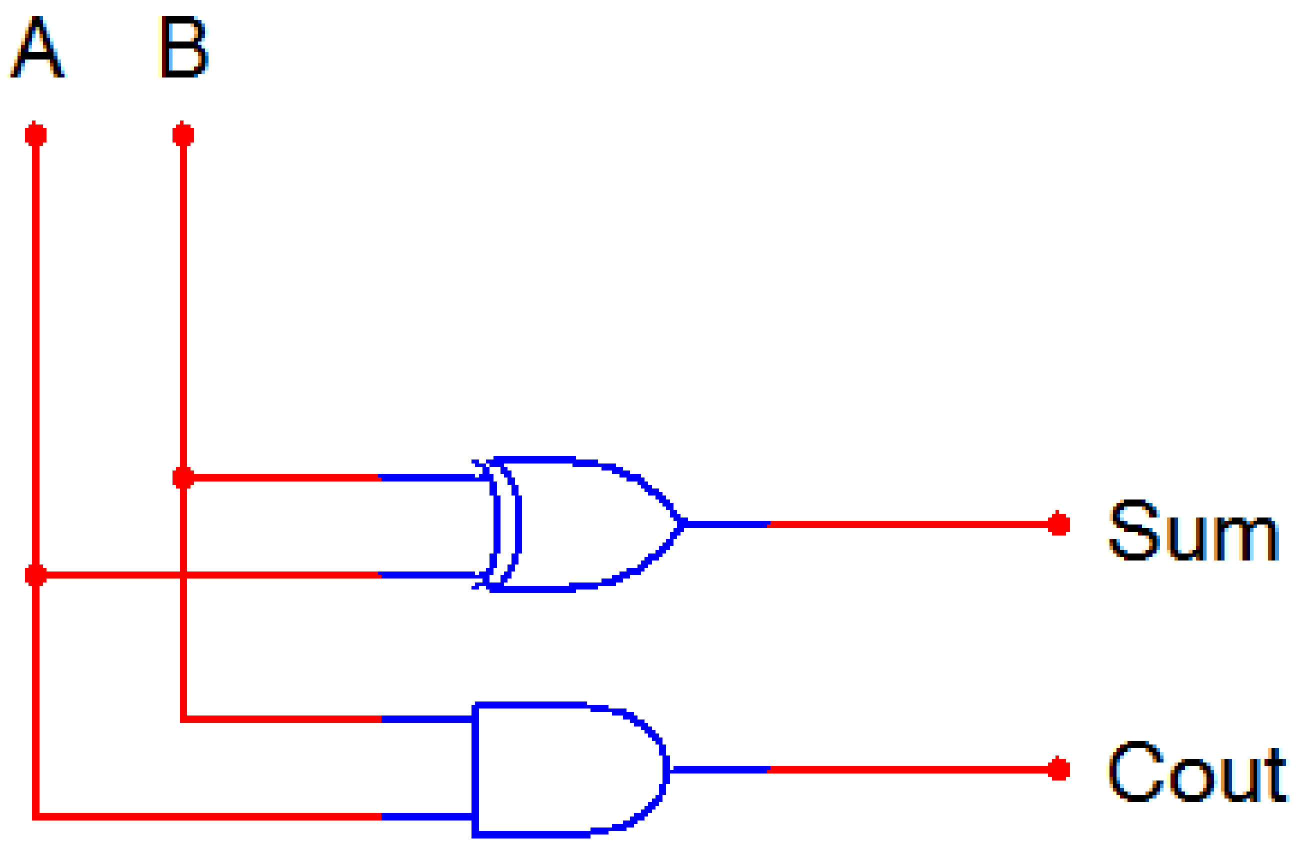
Half Adder – Design

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

$$\text{Sum} = \overline{A}\,B + A\,\overline{B} = A \oplus B$$

$$C_{out} = A\,B$$

Half Adder - Circuit



Full Adder – Design of C_{out}

A	B	C_{in}	Sum	C_{out}
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

	$\overline{C}_{in}$	C_{in}
$\overline{A}\,\overline{B}$	0	0
$\overline{A}\,B$	0	1
$A\,B$	1	1
$A\,\overline{B}$	0	1

$A\,B$

$B\,C_{in}$

$A\,C_{in}$

$$C_{out} = A\,B + B\,C_{in} + A\,C_{in}$$

Full Adder – Design of Sum

A	B	C_{in}	Sum	C_{out}
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

	$\overline{C_{in}}$	C_{in}
$\overline{A}\,\overline{B}$	0	1
$\overline{A}\,B$	1	0
$A\,B$	0	1
$A\,\overline{B}$	1	0

$$Sum = \overline{A}\,\overline{B}\,C_{in} + \overline{A}\,B\,\overline{C_{in}} + A\,B\,C_{in} + A\,\overline{B}\,\overline{C_{in}}$$

K-Mapping did NOT help us simplify . . . Let’s try Boolean algebra.

Boolean Simplification of Sum

$$\text{Sum} = \overline{A}\,\overline{B}\,C_{IN} + \overline{A}\,B\,\overline{C}_{IN} + A\,B\,C_{IN} + A\,\overline{B}\,\overline{C}_{IN}$$

$$\text{Sum} = \overline{A}\left(\overline{B}\,C_{IN} + B\,\overline{C}_{IN}\right) + A\left(B\,C_{IN} + \overline{B}\,\overline{C}_{IN}\right)$$

$$\text{Sum} = \overline{A}\left(B \oplus C_{IN}\right) + A\left(\overline{B \oplus C_{IN}}\right)$$

Let $K = B \oplus C_{IN}$ and substitute

$$\text{Sum} = \overline{A}\left(K\right) + A\left(\overline{K}\right)$$

$$\text{Sum} = A \oplus K$$

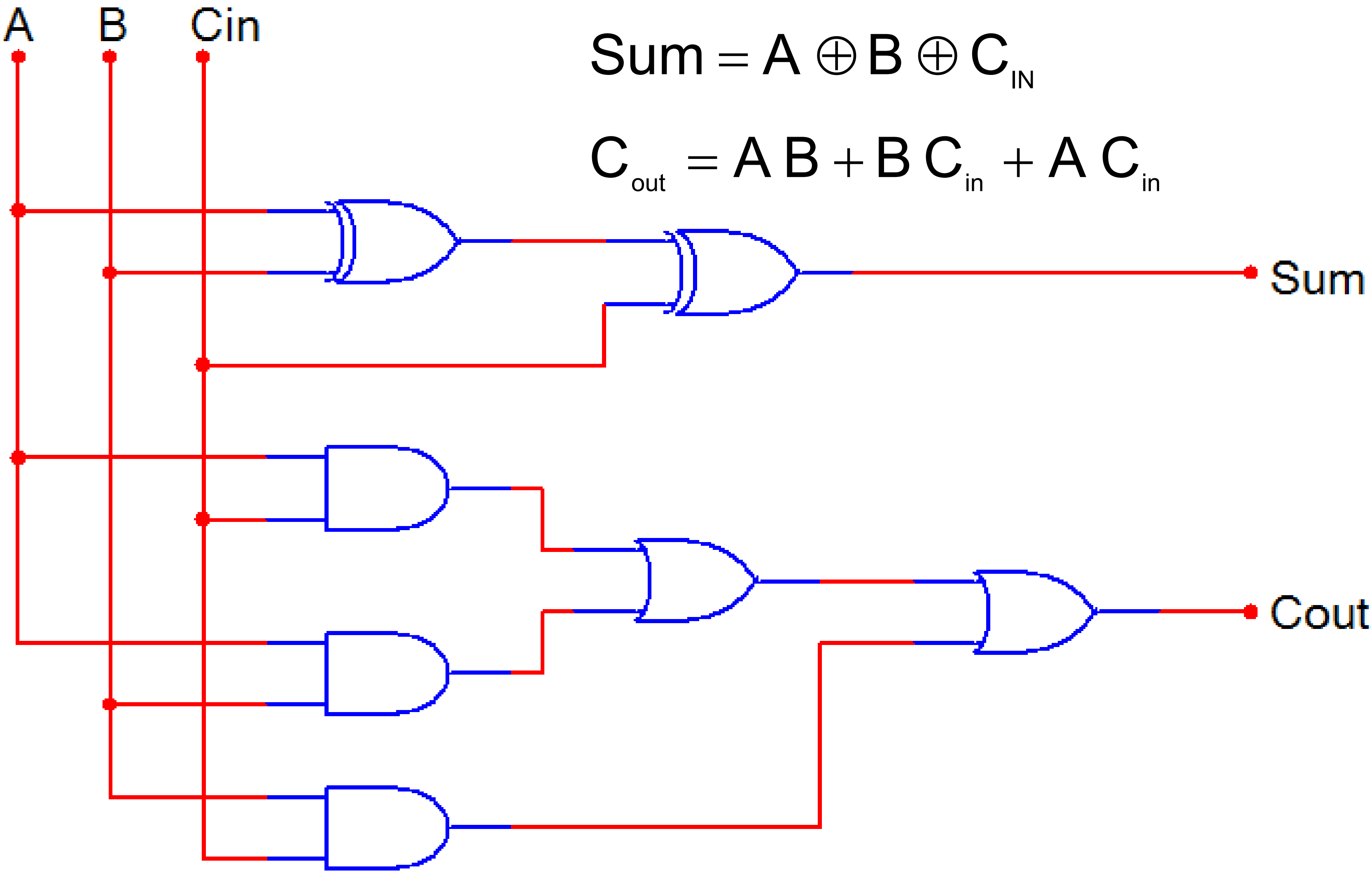
Replacing $B \oplus C_{IN}$ for K

$$\text{Sum} = A \oplus B \oplus C_{IN}$$

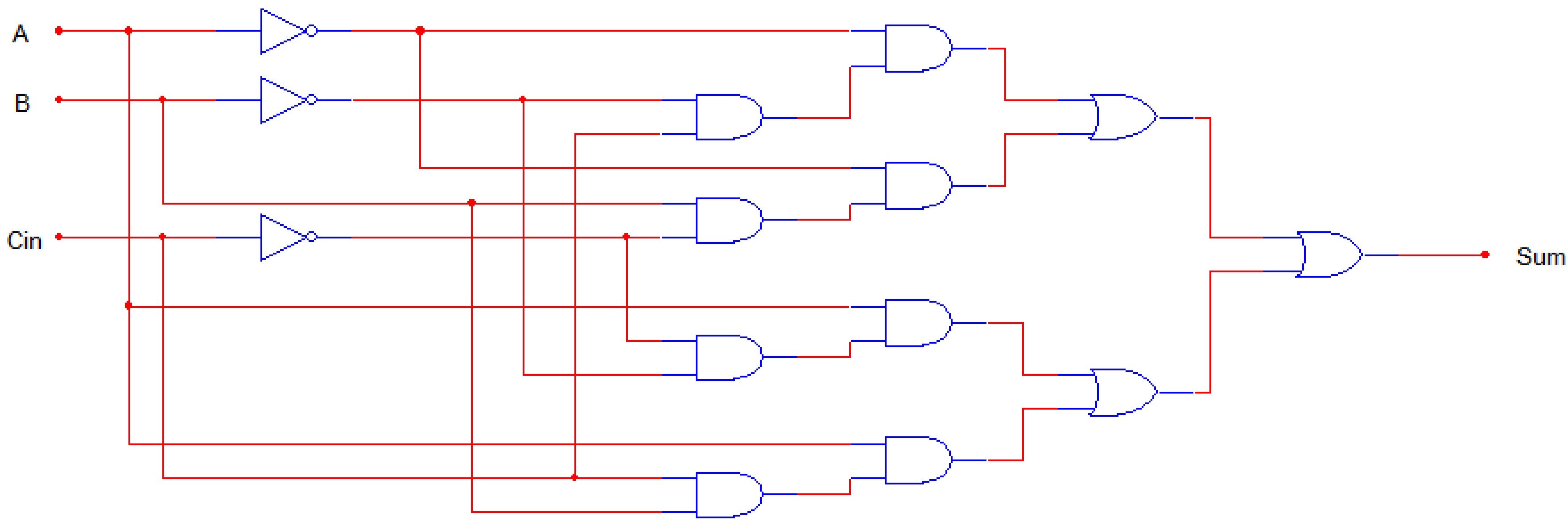
Full Adder - Circuit

$$Sum = A \oplus B \oplus C_{IN}$$

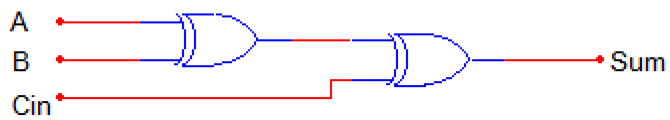
$$C_{out} = A\,B + B\,C_{in} + A\,C_{in}$$



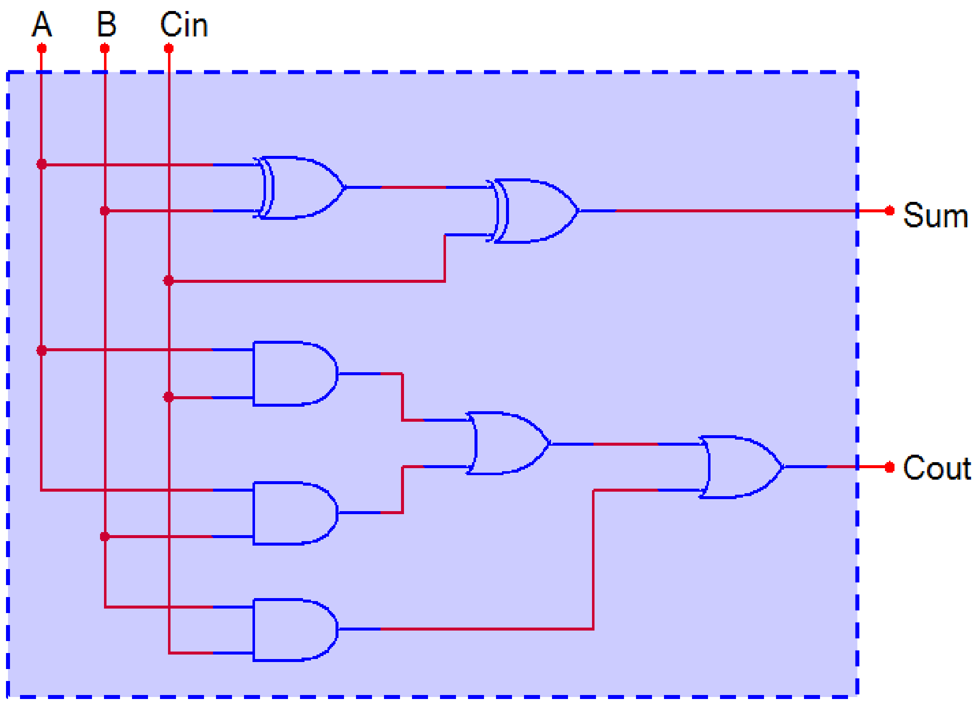
Full Adder: AOI vs. XOR



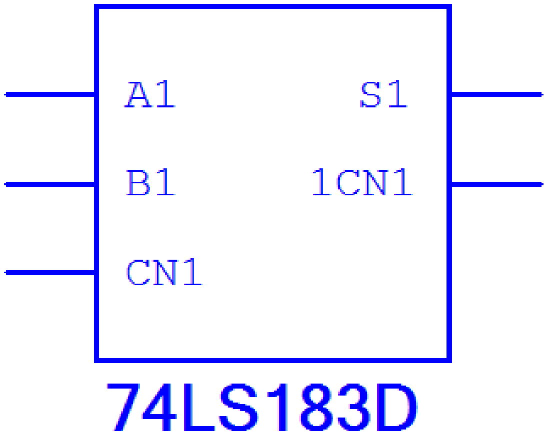
Though XOR gates can be used for implementing any combinational logic design, their primary application is adder circuits. Compare the AOI implementation (above) for the sum function to the XOR implementation (below).



MSI Full Adder



SSI - Full Adder



MSI - Full Adder

Cascading Adders – Four Bits

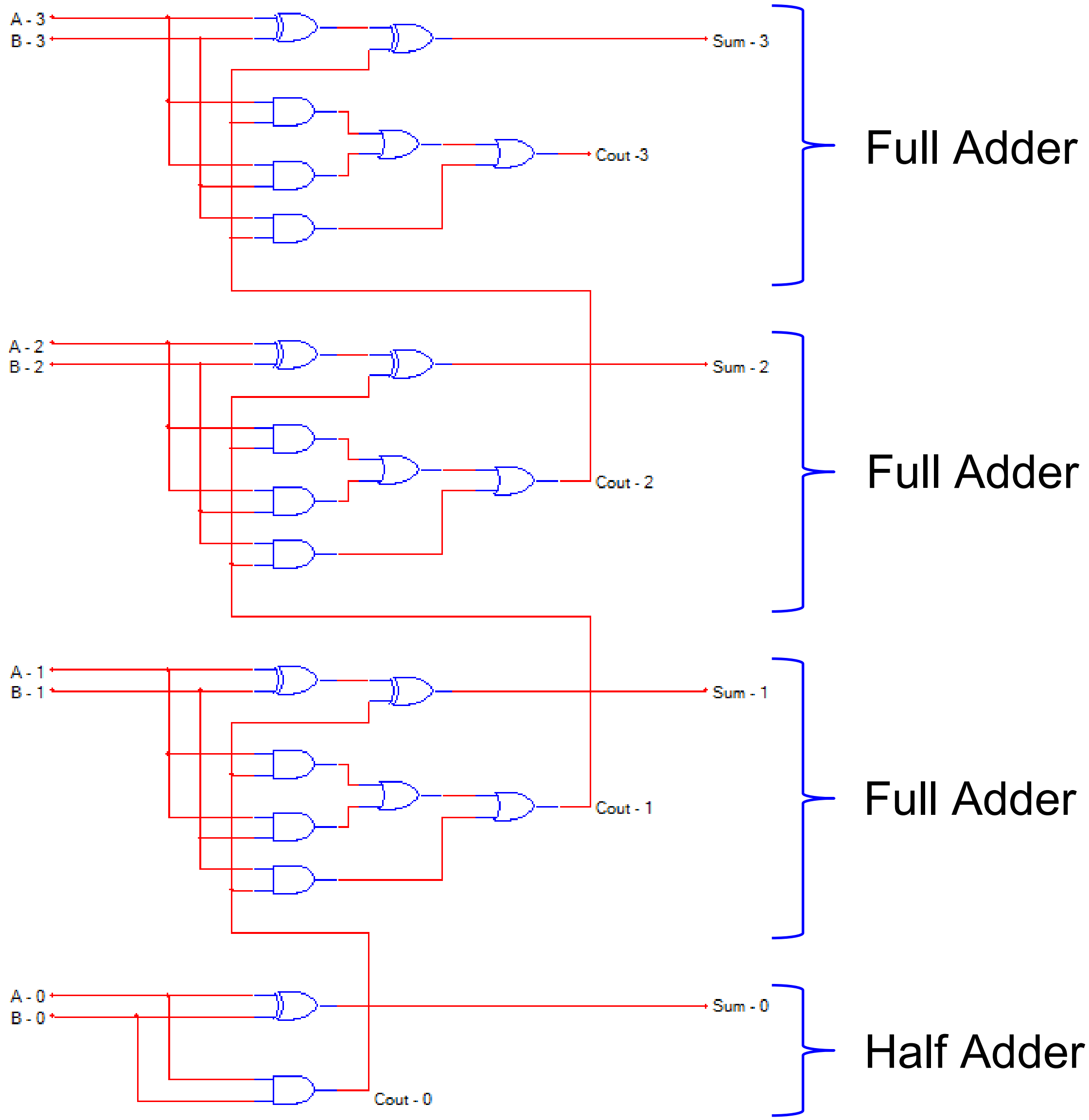
$$\begin{array}{r} 0\,1\,1\,0 \\ +\ 0\,0\,1\,1 \\ \hline 1\,0\,0\,1 \end{array}$$

Example: 6 + 3 = 9

General Form

$$\begin{array}{rcccc} & C_{out_3} & C_{out_2} & C_{out_1} & C_{out_0} \\ & A_3 & A_2 & A_1 & A_0 \\ + & B_3 & B_2 & B_1 & B_0 \\ \hline & S_3 & S_2 & S_1 & S_0 \end{array}$$

Four Bit Adder with SSI Logic



Four Bit Adder with MSI Logic