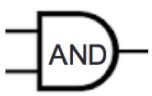
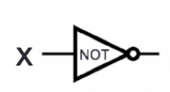
Input		Output
x	y	x AND y
1	1	1
1	0	0
0	1	0
0	0	0

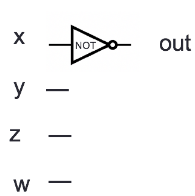
Input		Output
x	y	x XOR y
1	1	0
1	0	1
0	1	1
0	0	0

Input	Output
x	NOT x
1	0
0	1





Example digital circuit:



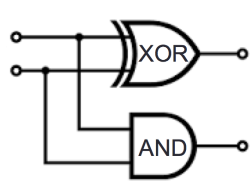
Output when $x = 1, y = 0, z = 0, w = 1$ is _____
Output when $x = 1, y = 1, z = 1, w = 1$ is _____
Output when $x = 0, y = 0, z = 0, w = 1$ is _____

Draw a logic circuit with inputs x and y whose output is always 0. *Can you use exactly 1 gate?*

Fixed-width addition: adding one bit at time, using the usual column-by-column and carry arithmetic, and dropping the carry from the leftmost column so the result is the same width as the summands. In many cases, this gives representation of the correct value for the sum when we interpret the summands in fixed-width binary or in 2s complement.

For single column:

Input		Output	
x_0	y_0	s_0	c_0
1	1		
1	0		
0	1		
0	0		



Draw a logic circuit that implements fixed-width 2 binary addition:

- Inputs x_0, y_0, x_1, y_1 represent $(x_1 x_0)_{2,2}$ and $(y_1 y_0)_{2,2}$
- Outputs z_0, z_1, z_2 represent $(z_2 z_1 z_0)_{2,3} = (x_1 x_0)_{2,2} + (y_1 y_0)_{2,2}$ (may require up to width 3)

First approach: half-adder for each column, then combine carry from right column with sum of left column

Write expressions for the circuit output values in terms of input values:

$z_0 =$ ____

$z_1 =$ ____

$z_2 =$ ____

Second approach: for middle column, first add carry from right column to x_1, then add result to y_1

Write expressions for the circuit output values in terms of input values:

$z_0 =$ ____

$z_1 =$ ____

$z_2 =$ ____

Extra example Describe how to generalize this addition circuit for larger width inputs.