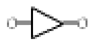
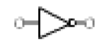
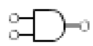
Introduction to Digital Logic Design

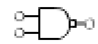
Part 1: Logic Gates

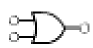
Using the simulator, connect each of the following logic gates to one or two switches and a light bulb so that you can test all possible input combinations. Then complete the truth tables by writing down the output for each set of inputs. We will write all values as either a **1**, which means ON or TRUE or HIGH, or a **0**, which means OFF or FALSE or LOW.

Buffer	
	
Input	**Output**
0	
1	

NOT Gate	
	
Input	**Output**
0	
1	

AND Gate		
		
Upper Input	**Lower Input**	**Output**
0	0	
0	1	
1	0	
1	1	

NAND Gate		
		
Upper Input	**Lower Input**	**Output**
0	0	
0	1	
1	0	
1	1	

OR Gate		
		
Upper Input	**Lower Input**	**Output**
0	0	
0	1	
1	0	
1	1	

NOR Gate		
		
Upper Input	**Lower Input**	**Output**
0	0	
0	1	
1	0	
1	1	

XOR Gate		
		
Upper Input	**Lower Input**	**Output**
0	0	
0	1	
1	0	
1	1	

XNOR Gate		
		
Upper Input	**Lower Input**	**Output**
0	0	
0	1	
1	0	
1	1	

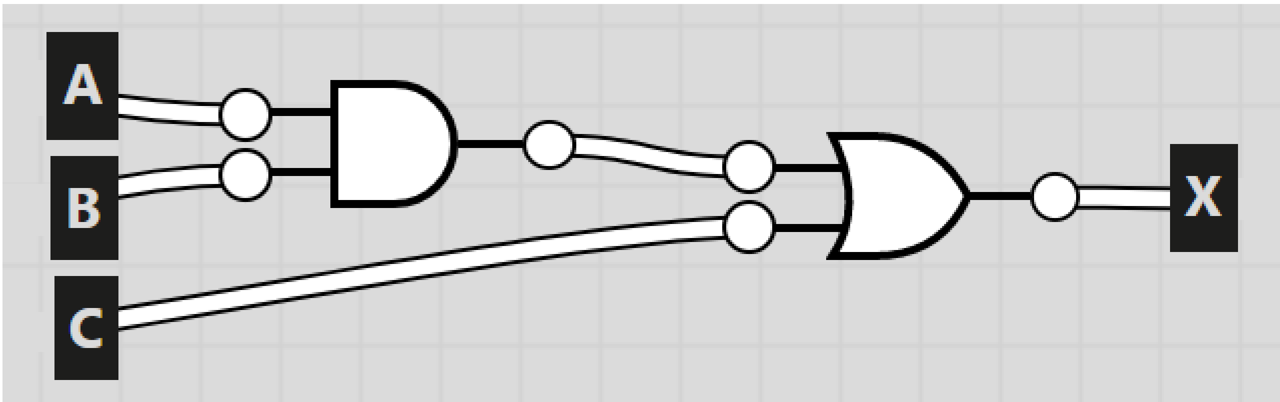
What is the difference between AND and OR gates?

What is the difference between OR and XOR gates?

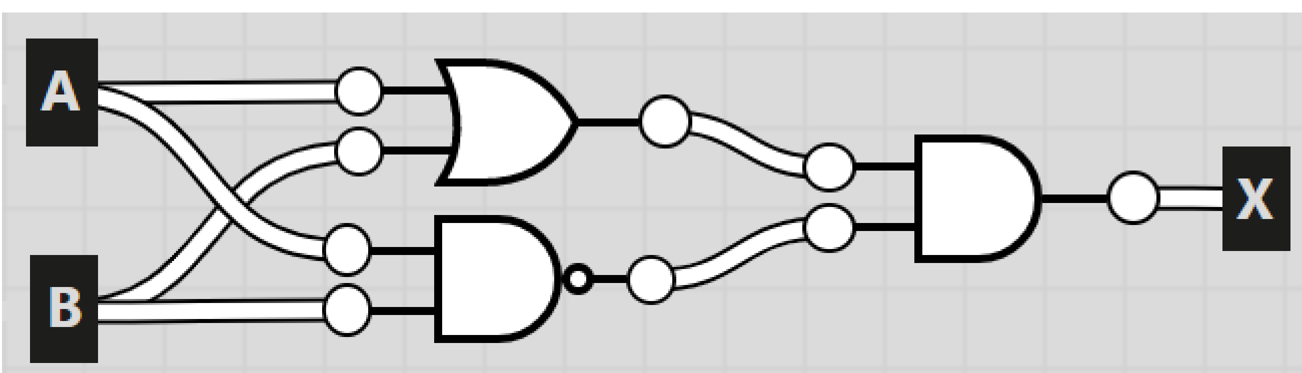
What do you think the small circles on the right sides of the NOT, NAND, NOR, and XNOR gates mean?

Part 2: Analyzing Circuits

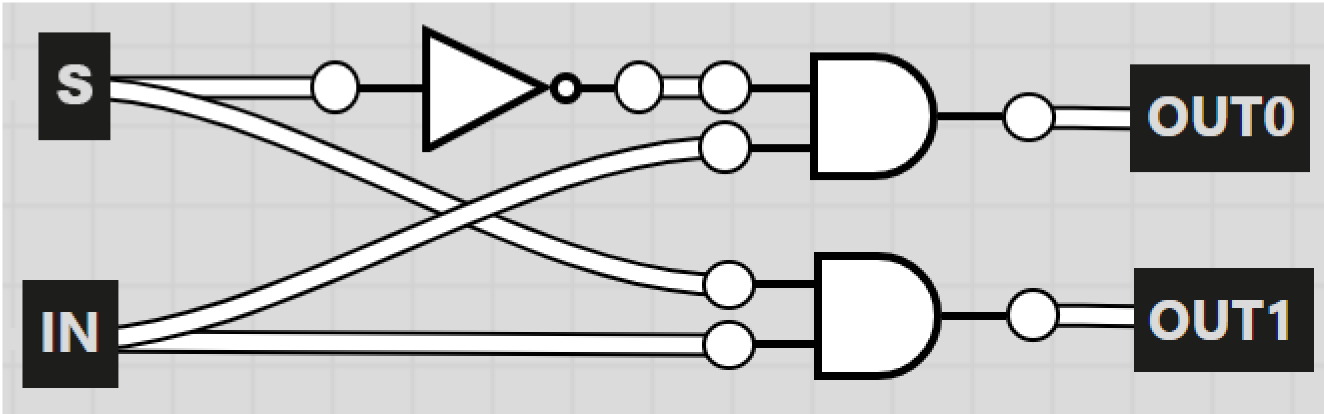
For each of the following circuits, try to complete the truth table using what you just learned about logic gates and your intuition about what happens when logic gates are connected. Intermediate columns have been provided to help make this easier. Then build the circuit in the simulator and record the actual output values to see if you were right.



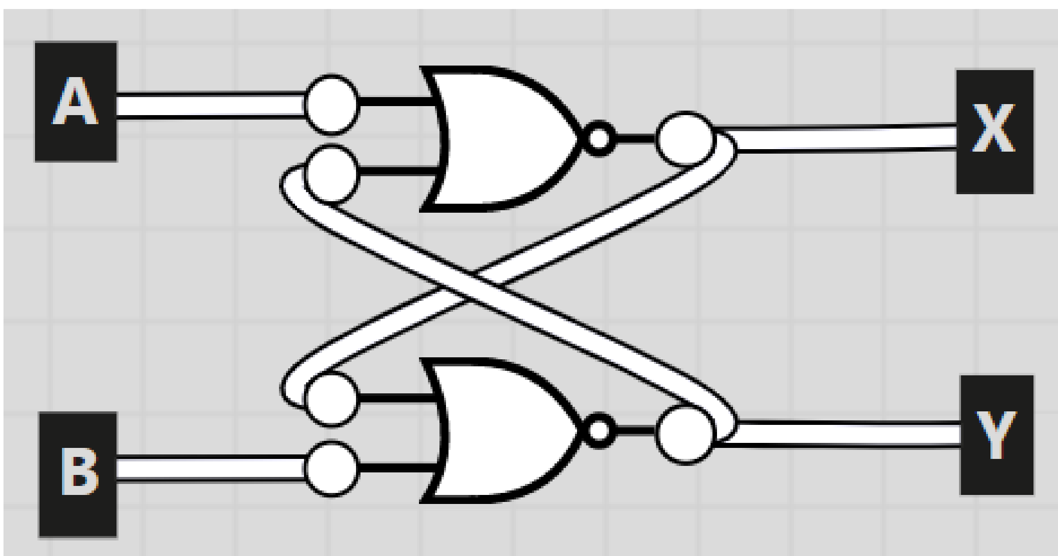
AND Gate w/ Override					
A	**B**	***A AND B***	**C**	**X (Predicted)**	**X (Simulated)**
0	0		0		
0	0		1		
0	1		0		
0	1		1		
1	0		0		
1	0		1		
1	1		0		
1	1		1		



Mystery Gate (???!)					
A	**B**	***A OR B***	***A NAND B***	**X (Predicted)**	**X (Simulated)**
0	0				
0	1				
1	0				
1	1				



1-to-2 Demultiplexer					
S	IN	OUT0 (Predicted)	OUT1 (Predicted)	OUT0 (Simulated)	OUT1 (Simulated)
0	0				
0	1				
1	0				
1	1				



There's something about this circuit that makes it different from the others…

Instead of trying to guess what it will do, go straight to the simulator. In addition to connecting each output to a light bulb, make sure that you connect the output of each NOR gate to one of the inputs of the other NOR gate.

What does it do?

If you had to make up a name for this circuit, what would you choose to call it?

Part 3: Designing Circuits

For each of the following truth tables, try to design a circuit in the simulator that produces the correct outputs for each set of inputs. When you have a working circuit, draw a picture of it below the truth table.

4-Input OR Gate (An OR gate with four inputs instead of two)				
Input A	**Input B**	**Input C**	**Input D**	**Output X**
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Half Adder (Adds together two 1-bit numbers, whatever that means…)			
Input A	**Input B**	**Output S (Sum)**	**Output C (Carry)**
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

OR Gate w/ Enable (An OR gate that can be enabled or disabled by a third input)			
Input EN (Enable)	**Input A**	**Input B**	**Output X**
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Part 4: Final Project

Using the specification below, complete the truth table for this very safe and sophisticated motor controller that includes an emergency stop button and an alarm. Then design a circuit in the simulator that produces the correct outputs for each set of inputs. When you're finished, draw a picture of it below the truth table.

Motor Controller (Specification)					
F (Forward)	R (Reverse)	E (Emergency Stop)	M1 (Motor)	M2 (Motor)	A (Alarm)
OFF	OFF	OFF	OFF	OFF	OFF
ON	OFF	OFF	ON	OFF	OFF
OFF	ON	OFF	OFF	ON	OFF
ON	ON	OFF	OFF	OFF	ON
DOES NOT MATTER		ON	OFF	OFF	ON

Motor Controller (Truth Table)					
Input F	Input R	Input E	Output M1	Output M2	Output A
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			