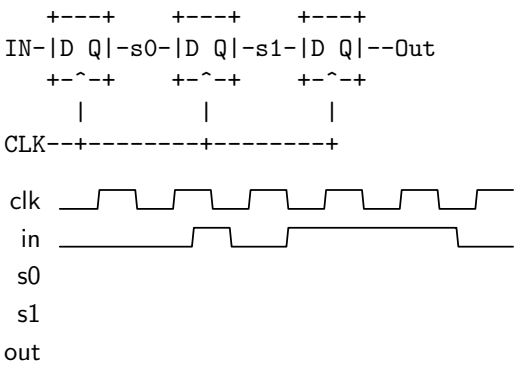
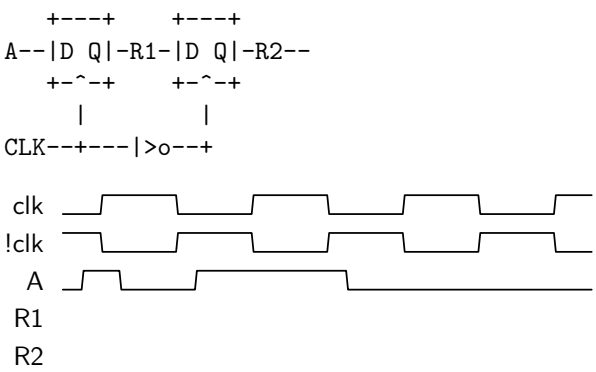
State

1. Fill out the timing diagram for the circuit below:

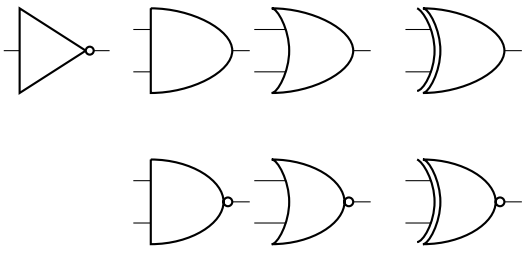


2. Fill out the timing diagram for the circuit below:



Logic Gates

1. Label the following logic gates:



2. Convert the following to boolean expressions:
   (a) NAND
   (b) XOR
   (c) XNOR
3. Create an AND gate using only NAND gates.

4. How many different two-input logic gates can there be? How many n-input logic gates?

Boolean Logic

$1 + A = 1$	$A + \bar{A} = 1$	$A + AB = A$	$(A + B)(A + C) = A + BC$
$0B = 0$	$B\bar{B} = 0$	$A + \bar{A}B = A + B$	
DeMorgan's Law:	$\overline{AB} = \bar{A} + \bar{B}$	$\overline{A + B} = \bar{A}\bar{B}$	

1. Minimize the following boolean expressions:
   (a) Standard: $(A + B)(A + \bar{B})C$

   (b) Grouping & Extra Terms: $\bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C} + AB\bar{C} + A\bar{B}\bar{C} + ABC + A\bar{B}C$

   (c) DeMorgan's: $\overline{A(\bar{B}\bar{C} + BC)}$

Finite State Machine

1. Draw a transition diagram for an FSM that can take in an input sequence one bit at a time, and after each input is received, output whether the number of 1s is divisible by 3. Write out the truth table that the combinational logic block must implement (remember to assign each state a binary encoding). Finally, write the Boolean algebra expressions that implement the FSMs truth table.