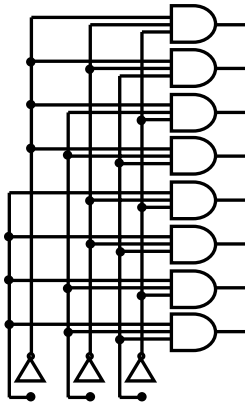
EECC 341, HW #5, Due Mon 4/20/09

1. Compute the total "**equivalent-gate (E.G.)**" cost associated with
   a. A generic n-to-2^n decoder
   b. A generic 2^n-to-1 multiplexer

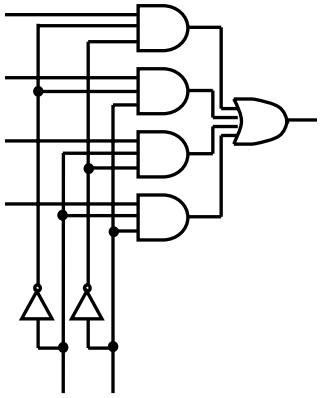
   Recall that, for example, an inverter worth zero E.G., a 2-input AND gate worth 1 E.G, and a 3-input NOR gate worth 2 E.G. Notice that the E.G. cost estimation only applies to fundamental logic gates: Inv, AND, OR, NAND, and NOR.

   Note: A generic decoder or multiplexer is one without ENABLE or any other control input. Below are a generic 3-to-8 decoder and a generic 4-to-1 multiplexer for your reference.

   3-to-8 decoder

   4-to-1 multiplexer

2. Implement a 4-variable function $F=\Sigma_{A,B,C,D}(0,1,4,5,6,9,11,14)$ using
   a. Only NAND gates and inverters
   b. A generic 4-to-16 decoder and possibly ANDs, ORs, & inverters
   c. A generic 3-to-8 decoder and possibly ANDs, ORs, & inverters
   d. A generic 16-to-1 multiplexer and possibly ANDs, ORs, & inverters
   e. A generic 8-to-1 multiplexer and possibly ANDs, ORs, & inverters
   f. In (b)-(e), do you really need AND, OR, & inverter for all cases? Indicate the types of logic gates you really need for each case, and state that whether your statement is true in general, i.e., for any given function F.
   g. Calculate and compare the E.G. costs associated with the designs you have in (a)-(e).
   h. Now suppose you have all the fundamental logic gates, i.e., {AND, OR, Inverter, NAND, NOR}, for you to use. Come up with the best design you could have that minimizes the E.G. cost. What is the cost?

   Note: Show your design steps (if applicable) for step (a)-(e), (h) and draw your circuit diagrams. For (b)-(e), try to use as few extra logic gates (in addition to the decoder/multiplexer) as possible.

3. You do not need to give answers to the following questions in the homework. Please, however, think about these problems and perhaps give yourself some examples.
   - What if you were to implement a given logic function with don't-cares using decoders and multiplexers? How would you simplify the implementation by taking advantages of the don't-cares?
   - What if you were to implement a problem with multiple output signals? Can you use a single decoder to implement multiple-output circuit? Can you do the same with multiplexers?