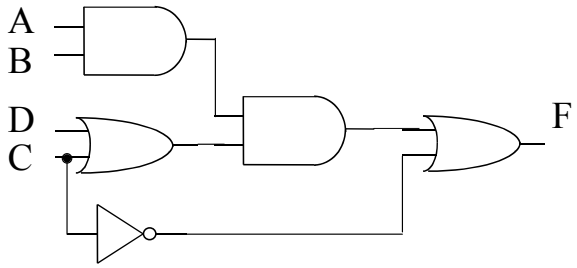
University of California at Berkeley
College of Engineering
Department of Electrical Engineering and Computer Science

EECS 150 R. H. Katz
Fall 2005

Problem Set #2: Programmable Logic
Assigned 7 September 2005, Due 16 September at 2 PM

1. Consider the following multilevel schematic constructed from AND-OR gates.



   a. What is the Boolean algebra equation derived from this schematic?
   b. What is the minimized AND-OR schematic that is equivalent to this schematic? Show your work using a K-map.
   c. Suppose that the schematic is restricted to only using NAND, NOR, and NOT gates. Convert the original schematic to a similar one making use of these gates only.
   d. Suppose that the schematic is restricted to only using NAND gates. Convert the original schematic to a similar one using only NAND gates. NOTE: You may have to add some gates to make the mapping work!
   e. Suppose that the schematic is restricted to only using NOR gates. Convert the original schematic to a similar one using only NOR gates. NOTE: You may have to add some gates to make the mapping work!

2. Consider the following four functions to be implemented in Programmable Logic Arrays:

   $F(A,B,C,D) = \Sigma m(1,4,5,7,9,12,13,15)$
   $G(A,B,C,D) = \Sigma m(1,3,5,6,7,13,14,15)$
   $H(A,B,C,D) = \Sigma m(4,6,12,13,14,15)$
   $I(A,B,C,D) = \Sigma m(1,3,13,15)$

   a. Use K-maps to minimize in sum of products form each of the four functions independently. Write down the Boolean expressions that result.
   b. How many unique product terms span the four functions?
   c. Now minimize the four functions, seeking to find the minimum number of product terms across all four functions simultaneously.

d. How many unique terms does your solution require? Is it a savings compared with your answer to part b?
e. Show how you would program the AND-OR array to implement the four functions using your solution to part c.

3. Programmable logic arrays as presented in class are based on AND-OR two-level logic. It is also possible to implement logic arrays in OR-AND logic. Repeat parts (a) through (d) except where the target is product of sums expressions for the functions F. G. H. I described above.
    e. Which solution is better and why, 2(d) versus 3(d)?

4. Given the function $F(A,B,C) = \Sigma m(1,3,4,6,7)$:
    a. Show how to implement F with an 8:1 multiplexer.
    b. Show how to implement F with a 4:1 multiplexer (HINT: put A and B on the selection inputs).

5. Given the function $G(A,B,C,D) = \Pi M(0,1,3,4,5,10,11)$:
    a. Show how to implement G with a 16:1 multiplexer.
    b. Show how to implement G with an 8:1 multiplexer (HINT: put A, B, and C on the selection inputs).

6. Given the function $H(A,B,C,D) = \Sigma m(1,3,5,7,12,13,14,15)$, show how to implement H with a single 4:1 multiplexer and no other logic.

7. Given the same function from problem 6:
    a. Implement H using a 4:16 decoder as a function generator along with a single large fan-in OR gate.
    b. Implement H using a 3:8 decoder as a function generator along with a single 4-input OR gate.

8. Given the functions F, G, H, I of Problem 2:
    a. How many Xilinx CLBs would it take to implement your answer to Problem 2(a) for all four functions together?
    b. How many Xilinx CLBs would it take to implement your answer to Problem 2(c) for all four functions together?
    c. Considering just the truth tables for the four functions, how many Xilinx CLBs would it take to implement them together?