CSE370: Introduction to Digital Design
Autumn 2001

Homework Set 3

DUE: Halloween, October 31, 2001

Please show *all* of your work. In certain problems, you may be asked to use Design Works. Otherwise, solutions do not have to be typeset, but may be if desired. In any case, your solutions must be legible. Please staple all the pages together. Make it clear which problem is which (especially important for the printouts from Design Works).

1) Follow through the design process to create a logic circuit that detects pairs of 1's in nonadjacent positions of a 4-bit binary word. Specifically, given a 4-bit word ABCD, the circuit output F shall be logic true when (A=1 and C=1) or when (B=1 and D=1) or when (A=1 and D=1). The output F shall be logic false for all other cases.
   a) Draw the truth table and K-map.
   b) Express F in *minimized* sum-of-products form.
   c) Express F in *minimized* product-of-sums form.
   d) Using your expression for F from (b), draw a circuit (logic) schematic for F. You don't have to use DesignWorks here, but may if you wish.
   e) Modify your schematic so that you are using only NAND gates or their equivalents. Do not use unnecessary gates.
   f) Modify your schematic once more so that you are using only NOR gates or their equivalents. Again, do not use unnecessary gates.

2) Katz Exercise 3.13 p.156 (time reponse)

3) Using Katz Exercise 3.16 b and c, p.158: Just demonstrate that all of these functions have hazards: show the case in which a hazard occurs (using a timing waveform, for example), and name the type of hazard (static 1-hazard, etc.)

4) Create a DesignWorks projects for some XOR components.

   a) Draw a DesignWorks schematic for a two-input XOR function with inputs A, B and output F using only NAND gates from the PrimGate library. Use "port in" and "port out" symbols from the Pseudo library to label your inputs and outputs. Turn in this schematic.

   b) Create a block symbol for your XOR component and label it appropriately. Now produce a new schematic with your XOR block on the sheet. Attach a "binary probe" from the Primio library to *F*. Attach a "hex keyboard" from the Primio library to *A* and *B* (note: the bottom wire on the hex keyboard is the least significant bit). Select digits from the keyboard, and verify that your XOR block outputs are correct. Turn in your schematic drawing, with the keyboard and probes.

   c) Draw a schematic for a four-input parity function similar to what you did in part a), but instead of using NAND gates, reuse only instances of your 2 input XOR block from part b). The inputs should be A3, A2, A1, A0 and output F. Note that F should be true if an odd number of the inputs are true. Turn in your schematic.

5) Katz problem 4.11, p.233 (re-engineering with MUXs).