EECS 140/141 Introduction to Digital Logic Design
Spring Semester 2020
Assignment **#3** Due **11 February 2020**

Reading: Sections 2.6, 2.8, 3.1 - 3.3, 3.9.1, 2.7 in Brown/Vranesic

All logic networks on this (and every other assignment) *must* be drawn using a logic template. Points will be deducted for failure to do this!

1. For the the truth table in the next-to-last problem of Assignment 1 *as given in the solutions:*
    a. Express this function as a logic expression in *canonical product-of-sums (CPoS)* form.
    b. Find the "cost" of the CPoS synthesis, and compare it to the answers to problem 10c in Assignment 2 *as given in the solutions* (implementation from word description and CSoP implementation).
2. Problem 2.35, part a, p. 72. For the truth table, put the input variables in the order given in the problem 2.35 statement (x_1, x_0, y_1, y_0). *ALSO,* give the minterm-list form for the logic expression for f (an example of the two possibilities for the min-term list form are given on p. 44 of your book -- you may use either of these possibilities for this problem).
3. For the AND/OR circuit described in lecture, use algebraic manipulation on the canonical product of sums (PoS) form to find a simpler (i.e., lower cost) PoS form. *Hint:* the minimum cost PoS implementation has cost=13, but it is not necessary that you find this simplest implementation; it is only necessary that you find an equivalent PoS implementation that has lower cost than the canonical PoS implementation.
4. Refer to p. 5 of your Lecture 3 skeleton notes for this problem. Show how to implement a 2-input NOR gate as a CMOS gate, similar to what we did in lecture for the 2-input NAND gate in CMOS. Specifically, do the following.
    a. Draw the circuit diagram with inputs V_{x_1} and V_{x_2}, and output V_f. There will once again be 4 transistors, 2 for the Pull Up Network (PUN) and 2 for the Pull Down Network (PDN).
    b. Construct a 4-row table (all combinations of the 2 inputs) with the following columns: V_{x_1}, V_{x_2}, T_{U_k} (one for each Pull Up Network (PUN) transistor in your circuit), PUN (Pull Up Network) for the entire PUN as a whole, T_{D_k} (one for each Pull Down Network (PDN) transistor in your circuit), PDN (Pull Down Network) for the entire PDN as a whole, and V_f. The voltage columns should have a Lo(0) or Hi(1) entry for each row, and the transistor columns and PUN/PDN network columns should have an Open or Closed entry for each row.
5. Problem 3.1, p. 157. In part (b), include the inverters in the transistor count.
6. Show that the XOR network of Figure P3.3 is functionally equivalent to the SoP network shown in Figure P3.1 (used in the previous problem) by finding the truth table for Figure P3.3 In finding the truth table for the XOR network, you *must* include an intermediate column for x_1 XOR x_2, then XOR that column with x_3 to get the output column.
7. Draw (a) a SoP and (b) a PoS synthesis of an XOR gate, using AND/OR/NOT gates.

8. For the logic network below, complete a truth table in which the inputs are listed in the order *a*, *b*, *c*, *d* (left to right). That is, the *d* column toggles 0, 1, 0, 1, etc. moving down the rows.