Homework three
EECS 203
Due 25 April
Prepared by Robert Dick and Russ Joseph

"Mano" is M. M. Mano and C. R. Kime, *Logic and Computer Design Fundamentals.* Prentice-Hall, NJ, fourth ed., 2008.

Please keep track of how long you spend doing this homework assignment. Specifically, how much time is needed to do the problems after studying enough to understand the concepts?

Please note that the assigned homework may not be enough for some people. If the concepts are still a little fuzzy after doing the homework, please take advantage of the other problems in Mano and/or see me for more problems.

Refer to Tables 1.2 and 1.3 in Mano if you need help converting between decimal and binary numbers.

When you can verify an answer, e.g., by using a truth table, check it.

1. **(0 pts.)** Do Mano 4.7, 4.10, 4.19, and 4.25, 1.1, 1.4, 1.7, 1.8, 1.9, 1.13, 5.3, and 5.6.

2. **(0 pts.)** Use the Quine–McCluskey method to find a minimal SOP expressions for the following functions:

   (a) $f(a, b, c) = \sum(5) + d(2, 3, 6, 7)$

   (b) $f(a, b, c) = \sum(2, 4) + d(0, 1, 3, 5, 6)$

   (c) $f(a, b, c) = \sum(1, 4, 5) + d(3, 6, 7)$

3. **(10 pts.)** CMOS review

   (a) Draw the schematics for two CMOS inverters connected in series.

   (b) If the input to the first inverter has been 0 for a long time, and the output of the second inverter is connected to ground, is current presently flowing in the circuit?

   (c) Is the circuit likely to be destroyed? If so, why?

   (d) If the input to the first inverter has been 1 for a long time, and the output of the second inverter is connected to ground, is current presently flowing in the circuit?

   (e) Is the circuit likely to be destroyed? If so, why?

4. **(5 pts.)** Name a function for which a SOP implementation requires an exponential number of literals in terms of the number of input variables, i.e., given

   $$f(i_{n-1}, i_{n-2}, \cdots, i_1, i_0)$$

   find an f for which the required number of literals, l, is related to the the number of inputs, n, in the following way

   $$l(n) \geq 2^n$$

   Hint: Try finding a Karnaugh map that absolutely must have a large number of small prime implicants.

5. **(5 pts.)** Are the number of literals in an implicant directly or inversely correlated with the implicant's size on a Karnaugh map?

6. **(5 pts.)** Use Karnaugh map to find minimal SOP expressions for the following function:

   (a) $f(a,b,c,d) = \prod(4,8,10,13) + d(5,14)$

7. **(5 pts.)** Use a Karnaugh map to find minimal POS expressions for the following functions:

   (a) $f(a,b,c,d) = \sum(0,3,5,7,8,13) + d(9,10,11,12,15)$

8. **(5 pts.)** Design a 3:8 decoder with inputs (i_2, i_1, i_0) and output o using two 1:4 demultiplexers and an inverter. You may make exactly one output high at any time.

9. **(10 pts.)** TGs

   (a) Implement the following function using transmission gates

   $$f(a,b) = \overline{a} + ab$$

   Use control signals a and b to drive TG control lines only, not TG inputs. Remember that TGs are just switches. When the NMOS control input is high, they're closed. When the NMOS control input is low, they're open. The PMOS control input is always the inverse of the NMOS control input. You can start from a switch diagram if that helps. Construct a series-parallel network of TGs routing 1 to the output when appropriate. Then construct a series-parallel network of TGs routing 0 to the output when appropriate. Use only four TGs and two inverters. You might want to simplify $f(a,b)$ first.

   (b) Derive an expression $j(a,b)$ that is 1 if and only if there is a path from 1 to the output.

   (c) Derive an expression $k(a,b)$ that is 1 if and only if there is a path from 0 to the output.

   (d) Prove that a short circuit will not occur by demonstrating that $j(a,b) \cdot k(a,b) = 0$.

   (e) Prove that a High-Z output will not occur by demonstrating that $j(a,b) + k(a,b) = 1$.

   (f) Recall the properties of NMOS and PMOS transistors. If you are permitted to eliminate superfluous transistors, how many NMOS and PMOS transistors does your design require?

10. **(5 pts.)** Assuming access to V_{DD} (1) and V_{SS} (0), show the schematic for an implementation of the following function using only an 8:1 MUX.

   $$f(a,b,c) = ac + \overline{b}$$

11. **(10 pts.)** Assuming access to V_{DD} (1) and V_{SS} (0), show the schematic for an implementation of the following function using only an 4:1 MUX.

   $$f(a,b,c) = ac + \overline{b}$$

12. **(10 pts.)** MUX and DMUX

   (a) Design a 2:1 MUX, $f(i_0, i_1, c)$, using only TGs and NOTs.

   (b) State two dangers when designing with TGs, i.e., state two ways in which an incorrect TG-based design may be *unsafe*.

   (c) Using only TGs, modify your design in (a), so that it will function as a DMUX, $g_0(i,c), g_1(i,c)$. Make sure the DMUX is *safe*.

   (d) Treating your 2:1 MUX as a component (block in the circuit), build a 4:1 MUX.

13. **(10 pts.)** DMUX Logic: Use an 8:1 DMUX and OR/NOR gates to implement the following functions. Try to use as few inputs on your OR/NOR gates as possible.

(a) $f(a,b,c) = \overline{a}\,b\overline{c} + ab\overline{c} + abc$

(b) $f(a,b,c) = (\overline{a} + \overline{b} + c)(a + b + c)$

(c) $f(a,b,c) = \overline{a} + \overline{b}\,c$

14. **(10 pts.)** Encoder: Use Karnaugh Maps to design a three-input priority encoder, $f(i_2, i_1, i_0)$ for which i_2 has the highest priority. Reserve one code for $\overline{i_2}\,\overline{i_1}\,\overline{i_0}$.

   (a) Write the minimized SOP expression for each encoded signal.

   (b) Write the minimized POS expression for each encoded signal. You may start from the same Karnaugh maps used in (a) to save time.

   (c) Based on the expression with fewer literals, derive a schematic for the encoder.

15. **(10 pts.)** Number systems

    (a) Convert 0xBADF00D to binary. Note: This is easier if you convert it to base-16 one digit at a time instead of considering the whole number at once.

    (b) Convert $(302)_8$ to base-10, and back again.

    (c) Convert the following ASCII codes to characters (see page 25 of Mano), using base-2 as an intermediate format: 0x48 0x65 0x58 0x21

    (d) Convert 744 to base-7 and back again.

16. **(5 pts.)** Using the conversion method in class (which, I hope, is clearer than that described in Mano),

    (a) Convert from $(01111)_2$ to Gray code and back again.

    (b) Convert from $(10000)_2$ to Gray code and back again.

17. **(0 pts.)** How much time did you spend on this assignment?

18. **(0 pts.)** Were there any problems that were mostly busy-work? In other words, were there any problems that required a lot of time without helping you to learn new concepts? If so, which ones.

19. **(0 pts.)** Were there any problems for which the lectures, book, and handouts didn't give you enough background information? If so, which ones?