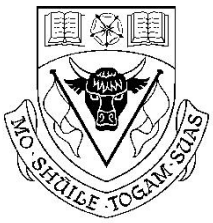

UNIVERSITY OF
CALGARY

ELECTRICAL AND COMPUTER ENGINEERING

ENEL 353 - Digital Circuits

Final Examination

Monday, December 20, 2004
Gold Gymnasium, 12:00 PM - 3:00 PM

Instructions:

- Time allowed is 3 hours.
- The examination is closed-book.
- Non-programmable calculators are permitted.
- The maximum number of marks is 100, as indicated. Please attempt all questions.
- Please use a pen or heavy pencil to ensure legibility.
- If you use more than one examination booklet, please make sure that your name and ID number are on each.
- Where appropriate, marks will be awarded for proper and well-reasoned explanations.

---

1. [**12 marks (3 marks each).**] Consider the decimal numbers 80 and -50. Perform the following operations.

   (a) Express the numbers in 4-digit 10's-complement format, and add them using 10's-complement addition. Convert your answer to standard decimal notation (e.g., +33, -50).

   (b) Express the numbers in 8-bit 2's-complement format, and add them using 2's-complement binary addition. Convert your answer to standard decimal notation (e.g., +33, -50).

   (c) Again using 8-bit 2's-complement format, perform 2's-complement binary subtraction of the second number from the first. Express your answer in decimal form as above. Explain how you may detect overflow, and if overflow has occurred in this case.

   (d) Express the two numbers in 8-bit sign-magnitude format, and convert each to octal and hexadecimal formats.

2. [**6 marks (3 marks each).**] Consider the following function:

$$F = x_1 x_2' + x_2 x_3' + x_1' x_3$$

   (a) Implement the corresponding circuit using 2-input NOR gates only (inverters are not available).

   (b) Implement the circuit using only 2-input AND and 2-input XOR gates (inverters are not available).

3. [**6 marks.**] Consider the block diagram shown in Fig. 1. The input is a four-bit binary number $A = a_3 a_2 a_1 a_0$, where a_3 as the most-significant bit (MSB). Give the truth table for a combinational circuit that encodes this number in two stages, as follows. As shown, the first stage converts A to Gray code $G = g_3 g_2 g_1 g_0$, where g_3 is the MSB. The second stage encodes G to a two-bit binary code $C = c_1 c_0$ using a priority encoder. The highest priority is g_3. The encoder output V is the "valid" bit indicating that one or more bits in the Gray code are equal to 1. Express the functions for the three outputs c_1, c_0, V, in their canonical SOP and POS forms. For the case $V = 0$, use don't cares for c_1 and c_0.

   (Note: it is not necessary to design the circuit; just give the truth table and write the canonical forms.)

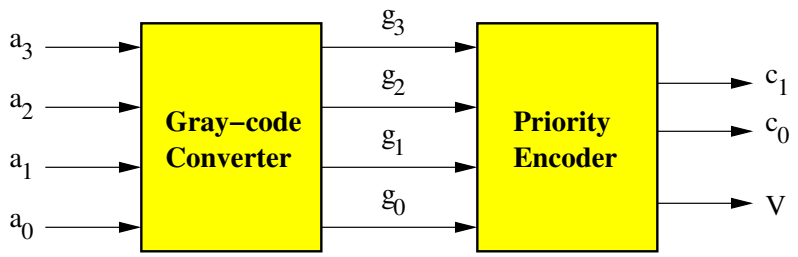

Fig. 1. A combinational encoder circuit

4. [**32 marks total.**] Consider the following three four-variable functions:

$$f_1(w,x,y,z) = \sum(1,3,6,7,8,9,12,13,15)$$
$$f_2(w,x,y,z) = \sum(0,2,4,5,8,10,12,13,14)$$
$$f_3(w,x,y,z) = \sum(0,2,4,5,10,11,14)$$

   (a) [**8 marks.**] Using Karnaugh maps, minimize each of the functions f_1, f_2, f_3, in the most appropriate form; that is, choose between SOP and POS forms such that each of the expressions contain the fewest number of AND and OR operations (inversions aren't counted).

   (b) [**4 marks.**] For f_1 only, draw either a two-level NAND-AND circuit, or NOR-OR circuit, depending on the form of your minimized function in part (a).

   (c) [**8 marks.**] Implement all three functions on a PLA. The outputs of the PLA are available in both "True" (non-complemented) and "Complement" form. Try to make the PLA implementation as simple as possible. You may give either the circuit diagram or the PLA programming table.

   (d) [**4 marks.**] Implement all three functions using a ROM.

   (e) [**8 marks.**] Implement all three functions using 4-to-1 multiplexers and any additional logic gates you may need. Try to make the circuit as simple as possible.

5. [**16 marks total.**] The circuit in Fig. 2 is a synchronous sequential circuit based on D flip-flops (DFFs).

   (a) [**12 marks.**] Using the conventional sequential-circuit analysis and synthesis methods, redesign the circuit using JK flip-flops (JKFFs).

   (b) [**4 marks.**] Suppose that, after redesigning the circuit for JKFFs, you discover that JKFFs are unavailable. Without otherwise changing your new circuit, how would you adapt some other type of flip-flop (your choice) to behave like a JKFF?

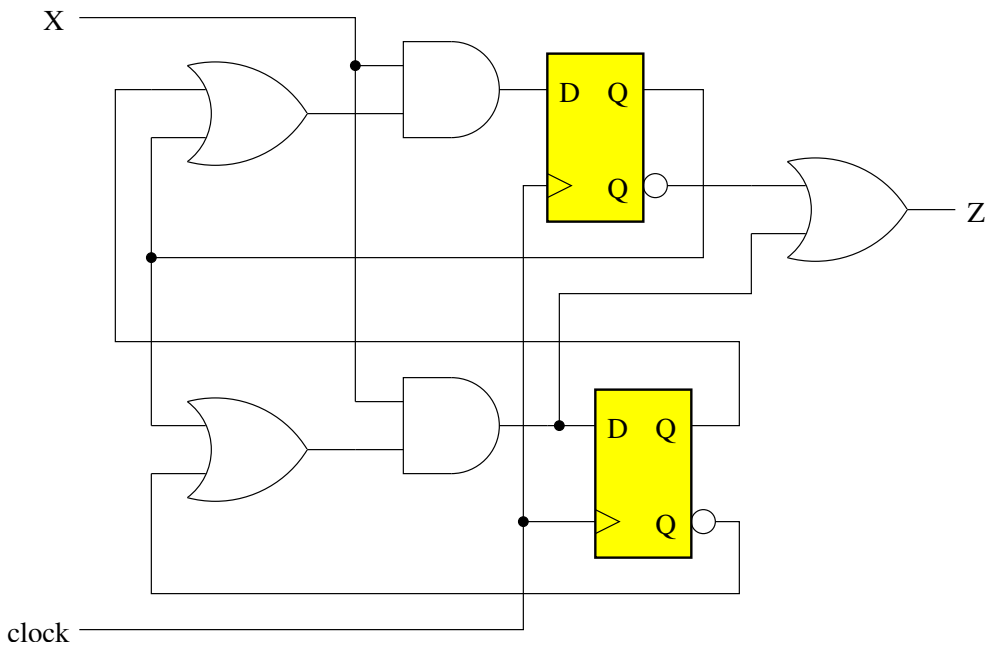

Fig. 2. Redesign this circuit using JKFFs

6. [**14 marks.**] Design a synchronous BCD *down-counter* based on T flip-flops. There is one output y, such that $y = 1$ when the count is zero, otherwise $y = 0$. Use don't-cares for the unused states to simplify your design. Determine the minimized output and flip-flop input equations; it is not necessary to draw the circuit.

7. [**14 marks total.**] Consider the following state diagram problems.

   (a) [**6 marks.**] For the state diagram shown in Fig. 3, write the state table. Minimize the number of states, if possible, and rewrite the state table.

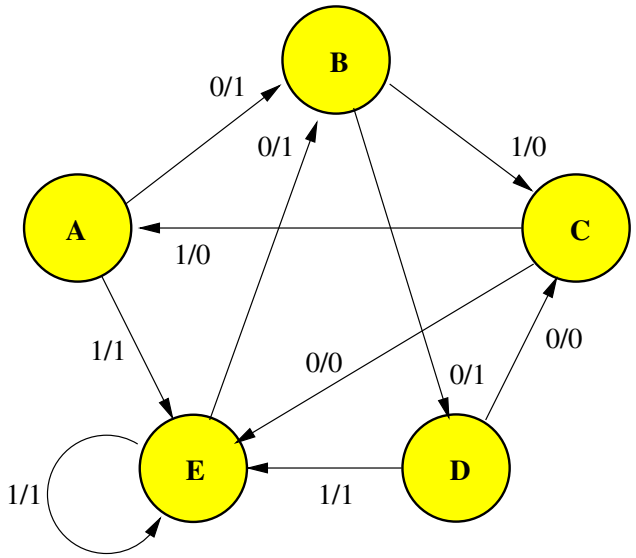

Fig. 3. A state diagram

   (b) [**8 marks.**] Draw a state diagram for a clocked sequential state machine with two inputs, $INIT$ and X, and one Moore-type output Z. The machine's operation is as follows:

   As long as $INIT$ is asserted (i.e., equal to 1), Z is continually 0. Once $INIT$ is negated (i.e., set to 0), Z should remain 0 until either of the following occurs:

   - X has been 0 for two consecutive clock ticks and then 1 for two consecutive clock ticks;
   - X has been 1 for two consecutive clock ticks and then 0 for two consecutive clock ticks.

   Then, Z should go to 1 and remain 1 until $INIT$ is asserted again.

   (Be as messy as you like!)