The University of New South Wales

School of Electrical Engineering and Telecommunications

ELEC2141: Digital Circuit Design

Final Examination

Session 1, 2011

- Time allowed: 3 hours.
- This paper contains FIVE (5) Questions on 7 total pages.
- Total marks available: 100.
- All questions are of equal value. Part marks are as indicated.
- Answer all questions.
- Answer each question in a separate book.
- Write the question number attempted on the cover of each book.
- The exam paper may be retained by the candidate.
- Authorized examination materials: University approved calculators.
- Any assumptions found to be necessary in answering the questions should be stated explicitly.
- All answers must be written in ink. Except where they are expressly required, pencils may be used only for drawing, sketching or graphical work.

Question 1. (20 marks)

i) Consider the Boolean function:

$$F(A, B, C, D) = (A + B + D)(\bar{A} + B + C) + (B + D)(\bar{B} + \bar{C} + D)A$$

a) Give the truth table representation of *F*. **[2 marks]**

b) Express the function *F* as a Sum-of-Minterms (you may use the "little *m*" short-cut notation). **[1 marks]**

c) Express the function *F* as a Product-of-Maxterms (you may use the "big *M*" short-cut notation). **[1 marks]**

With the aid of a Karnaugh map:

d) List all of the prime implicants of *F*. **[1 marks]**

e) List all of the essential prime implicants of *F*. **[2 marks]**

f) Find a minimal Sum-of-Products expression for *F*. **[1 marks]**

g) Find a minimal Product-of-Sums expression for *F*. **[2 marks]**

h) Using a 4-to-16-line decoder and NOR-gates, draw a decoder-based implementation of *F*. In your design, you may use the standard symbol for a decoder (clearly labeling the inputs and outputs); for the additional logic, you may only use NOR gates with 4 inputs or less. **[3 marks]**

i) For your design in part (h), find the gate-input cost (inverters included) of the decoder, the gate-input cost of the additional logic, and therefore, the total gate-input cost of the complete design. You may assume the decoder is implemented as a cascaded structure, using only 2-input AND gates and inverters. **[3 marks]**

ii) Using Boolean algebra, prove the identity:

$$BC + \bar{A}\bar{B} + \bar{A}\bar{C} = \bar{A} + ABC$$

[4 marks]

Question 2 on the next page.

Question 2. (20 marks)

i) Consider a state diagram of a Mealy state machine that is implemented with ***L*** flip-flops, ***M*** inputs, and ***N*** outputs (where *L*, *M* and *N* are positive integers).

a) What are the minimum and maximum numbers of states in the state diagram? **[2 marks]**

b) What are the minimum and maximum numbers of transition arrows starting at a particular state? **[2 marks]**

c) What are the minimum and maximum numbers of transition arrows that can end in a particular state? **[2 marks]**

d) What are the minimum and maximum numbers of different binary patterns that can be observed on the outputs? **[2 marks]**

ii) Consider the following state transition table, describing a Mealy state machine with two inputs *X* and *Y*, and one output *Z*:

Current State	**Next State, Output Z**			
	XY = 00	**XY = 01**	**XY = 10**	**XY = 11**
A	A, 0	C, 0	B, 0	B, 1
B	D, 0	C, 0	A, 0	B, 1
C	D, 1	B, 0	C, 0	D, 0
D	D, 0	C, 0	D, 0	B, 1

Parts (a) and (b) below can be done in either order. State which part you are attempting first.

a) Convert the Mealy machine to an equivalent Moore machine. Describe the Moore machine using a state transition table. **[4 marks]**

b) Minimize the number of states and derive the reduced state transition table. **[4 marks]**

The Moore machine is to be implemented using D flip-flops and additional logic.

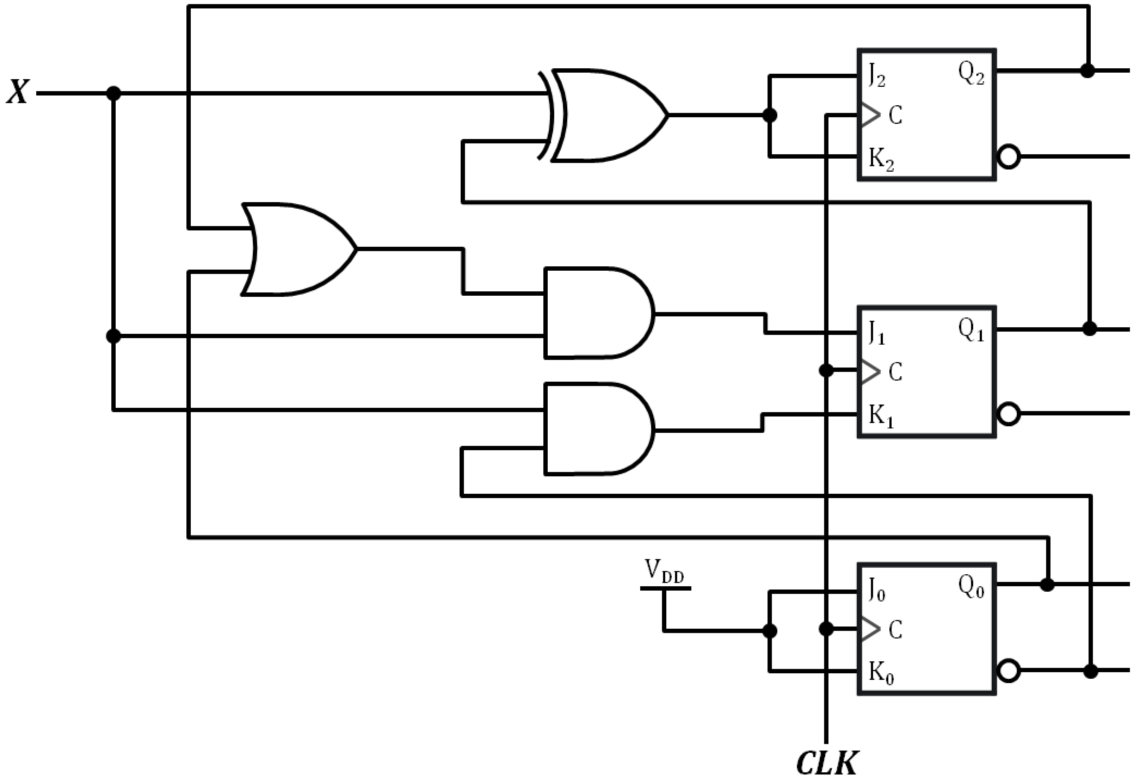
c) Choose and list binary code assignments for the states. **[1 mark]**

d) Derive the flip-flop and output equations. **[3 marks]**

Please turn over - Question 3 on the next page.

Question 3. (20 marks)

Consider the following JK flip-flop based synchronous sequential circuit, with one input *X*:



i) Write all flip-flop input equations. **[3 marks]**

ii) Derive the state transition table. **[10 marks]**

iii) With the states labeled as:

State	$Q_2Q_1Q_0$	State	$Q_2Q_1Q_0$
Z	000	M	100
H	001	U	101
A	010	!	110
B	011	-	111

draw the state diagram (note that the last two characters are the actual state names). **[4 marks]**

iv) With initial state Z, and input sequence 10110011 on *X*, find the sequence of (labeled) states traversed. **[3 marks]**

Question 4 on the next page.

Question 4. (20 marks)

i) The truth table for a 1-bit combinational binary subtractor, analogous to the half adder, computing the difference, *D* = *X* minus *Y*, with *BL* (borrow-from-left), is:

X	Y	D	BL
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

a) Design the logic diagram (circuit) for a 1-bit combinational binary subtractor, comparable to the full adder, with two data inputs *X* and *Y*, a borrow-from-the-right input *BI*, a borrow-request-to-the-left output *BL*, and a difference output *D*. In your answer, show all derivation work. **[8 marks]**

b) Draw a block symbol that represents your design, clearly indicating the inputs and outputs. Place the input/output signals such that the symbol can be easily used as part of a larger multi-bit subtractor design. **[2 marks]**

c) Using multiple instances of the symbol from (b), draw a block diagram for a 4-bit subtractor. **[3 marks]**

d) Assuming signed (2s complement) inputs to the 4-bit subtractor in (c), explain (with the aid of a diagram) what additional logic needs to be added in order to have an underflow indicator signal *UF*. **[3 marks]**

ii) Explain how a simple computer datapath can be modeled as a state machine. In your answer describe what forms the current state, what events may give rise to state transitions, and provide examples of input and output signals. **[4 marks]**

Please turn over - Question 5 on the next page.

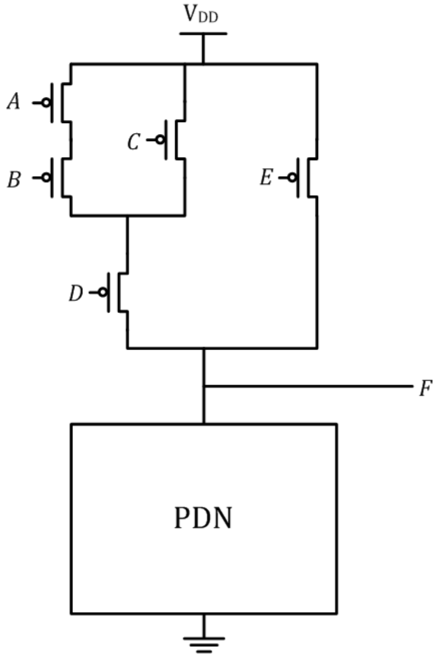
Question 5. (20 marks)

i) Consider the CMOS circuit presented on the right.

a) Draw the corresponding pull-down network (PDN) such that the output *F* always takes one of the two logic values. **[3 marks]**

b) Find a Boolean expression for the function *F*. **[3 marks]**

c) By adding additional MOS transistors, and with the aid of a diagram, suggest a way to convert the circuit for *F* into a 3-state complex gate. That is, introduce an enabling input *EN* such that when *EN* = 1 the output follows the function *F*, and when *EN* = 0 the output behaves as a high-impedance. **[4 marks]**



ii) Consider the following Verilog modules, implementing 2-to-1-line and 4-to-1-line multiplexers, respectively:

```
module mux2(S, I0, I1, Y);
   input S;
   input I0, I1;
   output Y;
   reg Y;

   always @(S) begin
      case (S)
         1'b0: Y <= I0;
         1'b1: Y <= I1;
         default: Y <= 1'bx;
      endcase
   end
endmodule
```

```
module mux4(S, I0, I1, I2, I3, Y);
   input [1:0] S;
   input I0, I1, I2, I3;
   output Y;
   reg Y;

   always @(S) begin
      case (S)
         2'b00: Y <= I0;
         2'b01: Y <= I1;
         2'b10: Y <= I2;
         2'b11: Y <= I3;
         default: Y <= 1'bx;
      endcase
   end
endmodule
```

a) The two modules have a similar bug that will prevent the multiplexers from functioning correctly. Comment on the current behavior of the modules and explain why this behavior is undesirable. Suggest a way to fix the bug. **[3 marks]**

(Continued on the next page...)

Question 5 continues on the next page.

A 16-to-1-line multiplexer can be designed by cascading several instances of the smaller multiplexer modules, as shown in the code below:

```
module mux16(S, I0, I1, I2, I3, I4, I5, I6, I7, I8, I9, I10, I11,
             I12, I13, I14, I15, Y);
   input [3:0] S;
   input I0, I1, I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12,
         I13, I14, I15;
   output Y;

   wire N11, N12, N13, N14, N21, N22;

   mux4 mx11(S[3:2], I#, I#, I#, I#, N11);
   mux4 mx12(S[3:2], I#, I#, I#, I#, N12);
   mux4 mx13(S[3:2], I#, I#, I#, I#, N13);
   mux4 mx14(S[3:2], I#, I#, I#, I#, N14);

   mux2 mx21(S[1], N11, N12, N21);
   mux2 mx22(S[1], N13, N14, N22);

   mux2 mx31(S[0], N21, N22, Y);

endmodule
```

b) Draw the block diagram corresponding to the structural description of the *mux16* module above. **[3 marks]**

The numbers for the input connections have been masked out (denoted `I#` in the four bolded lines). For correct operation of the multiplexer, the output should follow the input with a number corresponding to the select bits (where S[3] is the MSB and S[0] is the LSB).

c) Rewrite the four lines of the Verilog code above, replacing the masked signal names with the correct input signals. Take special note of the order that the select bits are connected in the cascaded structure.

[4 marks]

End of paper.