

Name Solution_____

Digital Logic I
EE 2720-2
Midterm Examination 10_2
9 November 2011, 14:40–15:30 CST

Exam Rules

Use only a pencil or pen. No calculators of any kind are allowed. Texting is out of the question.

Problem 1 _____ (22 pts)

Problem 2 _____ (22 pts)

Problem 3 _____ (22 pts)

Problem 4 _____ (12 pts)

Problem 5 _____ (12 pts)

Problem 6 _____ (10 pts)

Alias Salla_____

Exam Total _____ (100 pts)

Good Luck!

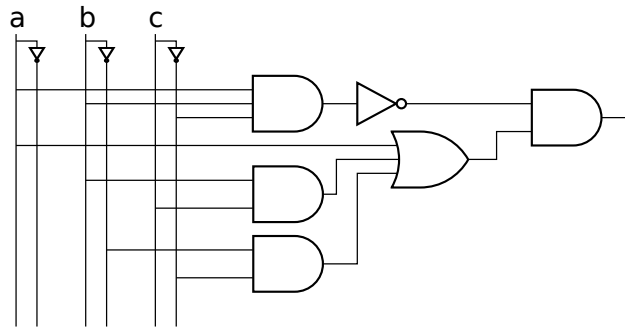
Problem 1: (22 pts) The problems below are based on the following Boolean function:

$$(a + bc + b'c')(abc)'$$

(a) Draw a logic diagram (using AND, OR, and NOT gates) corresponding to the Boolean function. (Do **not** simplify the expression.)

Logic diagram.

Solution appears below.



(b) Write the Boolean function in minterm canonical form. (Show a Boolean expression, not just a list of minterm numbers.) *Hint: For most people directly constructing a truth table would be easier than algebraic manipulation.*

Expression in minterm canonical form.

Solution:

$$\bar{a}\bar{b}\bar{c} + \bar{a}bc + a\bar{b}\bar{c} + a\bar{b}c + abc$$

(c) Write the Boolean function in maxterm canonical form.

Expression in maxterm canonical form.

Solution:

$$(a + b + \bar{c})(a + \bar{b} + c)(\bar{a} + \bar{b} + c)$$

Grading Note: A common mistake is inverting the wrong literals. For example, $(\bar{a} + \bar{b} + c)$ instead of the correct first term, $(a + b + \bar{c})$.

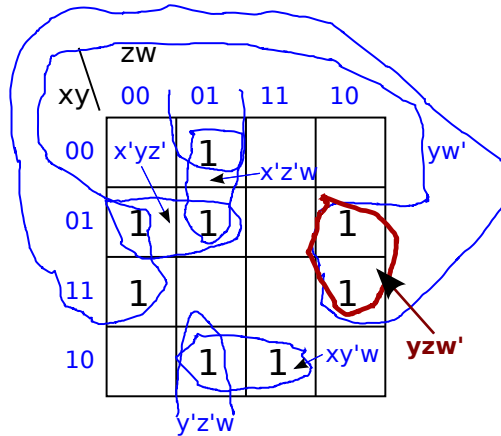
(d) Draw a Karnaugh map for the expression. (Just draw the Karnaugh map, don't use it to simplify the expression.)

Karnaugh map, including variables and row and column numbers.

Solution appears below:

		bc			
		00	01	11	10
a	0	1		1	
	1	1	1	1	

Problem 2: (22 pts) Consider the Karnaugh map below.



(a) Write in the row and column numbers.

Row and column numbers.

Solution appears on the diagram above in blue. (The boxes circled with a heavy red line are a solution to a following part.)

(b) List all of the prime implicants both on the Karnaugh map above, and as a list below.

Prime implicants circled on Karnaugh map.

List prime implicant expressions below.

They are:

(c) In the list of prime implicants above, write an “E” next to each essential prime implicant.

Write an “E” next to essential prime implicants.

Prime Implicants:

- E $y\bar{w}$
- $\bar{x}y\bar{z}$
- $\bar{x}\bar{z}w$
- E $x\bar{y}w$
- $\bar{y}\bar{z}w$

Grading Note: A common mistake was marking an implicant as essential when it really wasn't. Perhaps this was due to the misconception that an implicant that appears in a minimum cost expression must be essential.

(d) Provide an example of an implicant that's neither a prime implicant, nor a minterm. Circle this implicant and show the corresponding Boolean expression. *Grading Note: The original wording of the checkbox item below was slightly different in the original exam.*

Circle implicant that's neither a minterm nor a prime implicant.

Show an expression for the implicant.

On the diagram above the implicant is circled with a **thick red line**, on the right of the Karnaugh map.

(e) Based on the Karnaugh map show a minimum-cost expression for this logic function.

Minimum-cost expression.

The minimum cost expression is:

$$y\bar{w} + x\bar{y}w + \bar{x}\bar{z}w$$

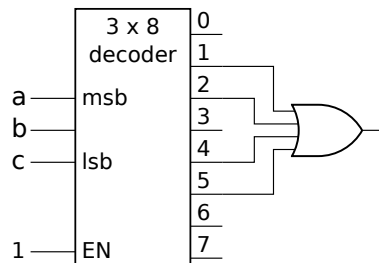
Problem 3: (22 pts) Consider the Boolean function below:

$$ab' + b'c + a'bc'$$

(a) Use a 3×8 decoder plus whatever logic gates are needed to implement this function.

Implement using 3×8 decoder and gates.

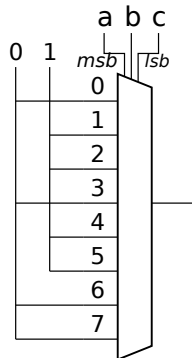
For a decoder implementation one must identify the minterms. One easy way to do that is to construct a truth table. The minterms are $m_1, m_2, m_4,$ and m_5 . Since there are three variables ($a, b,$ and c) a 3×8 decoder is needed. The decoder outputs corresponding to these minterms are connected to an OR gate, as is done in the following diagram.



(b) Use an 8-input multiplexer to implement this function.

Implement using an 8-input multiplexer.

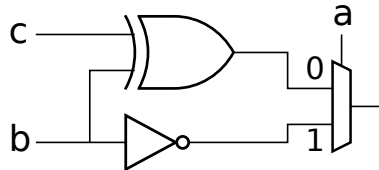
As with the decoder, minterms are used. Set the multiplexer inputs corresponding to the minterms to 1, and set the other inputs to 0. The select inputs are connected to the variables.



(c) Use a multiplexer and additional logic, including possibly exclusive-or gates, to implement this function by performing a Shannon expansion with respect to a (use a as the multiplexer control input). *Hint: it might be easier to eyeball a truth table than to do this by algebraic manipulation.*

Implement using a multiplexer based on a .

When $a = 0$ the function is $b'c + bc' = b \oplus c$. When $a = 1$ the function is $b' + b'c = b'$. These two expressions are used for the multiplexer inputs as is done in the diagram below:

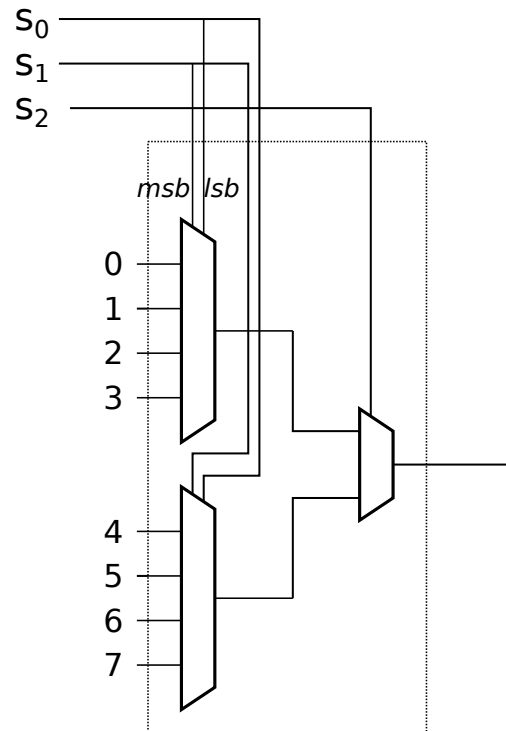


Problem 4: (12 pts) Show how to implement the 8-input multiplexers described below. In each case the three select input bits should be labeled s_2 , s_1 , s_0 , with s_0 being least significant. Label the data inputs 0 to 7.

(a) Implement an 8-input multiplexer using two 4-input multiplexers and a 2-input multiplexer.

Eight-input mux using two 4-input multiplexers.

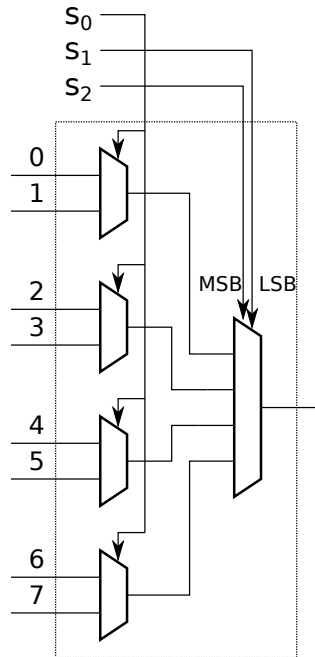
Solution appears below.



(b) Implement an 8-input multiplexer using four 2-input multiplexers and one 4-input multiplexer.

Eight-input mux using four 2-input multiplexers and a 4-input mux.

Solution appears below.



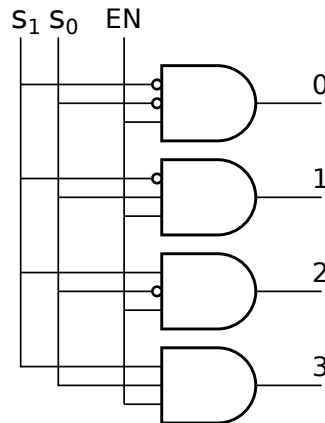
Problem 5: (12 pts) Implement the devices as described below.

(a) Show the logic gates needed to implement a 2×4 decoder, include an enable input. Appropriately label the inputs and outputs.

Logic diagram for a 2×4 decoder, just use gates.

Include logic for enable input.

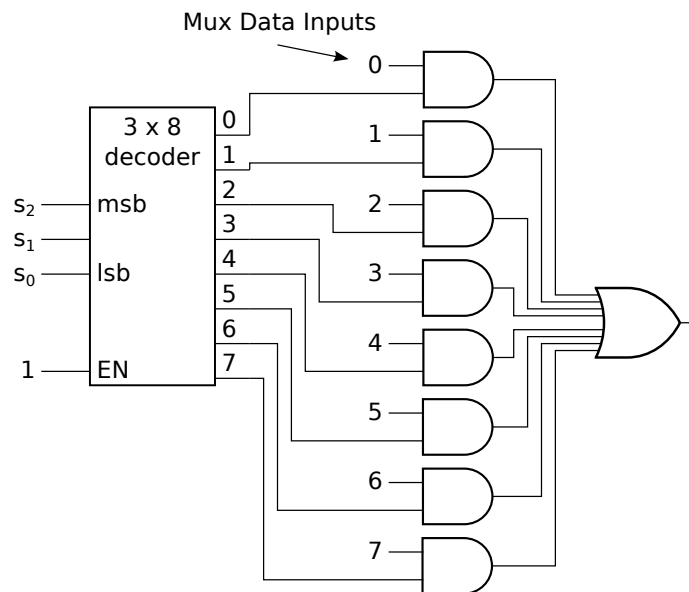
Grading Note: Several students had the correct logic diagram for the decoder, but then OR'd together all of the outputs.



(b) Show how to implement an 8-input multiplexer using a decoder and logic gates. Appropriately label the inputs and outputs.

Logic diagram for an 8-input multiplexer using gates and a decoder.

Solution appears below:



Problem 6: (10 pts) Answer each question below.

(a) Consider five seats, numbered 0 to 4, arranged in a circle and described by Boolean variables i_0 to i_4 . Boolean variable i_0 is true if seat 0 is occupied and \bar{i}_0 is false if the seat is not occupied (no one is sitting in the seat), likewise for i_1, i_2, i_3 , and i_4 .

Write a Boolean expression that's true if at least two people are sitting next to each other and at least one seat is not occupied. (Note: Just write one Boolean expression.) *Hint: This can easily be solved without a truth table.*

Boolean expression.

The solution is:

$$(i_0i_1 + i_1i_2 + i_2i_3 + i_3i_4 + i_4i_0)(\overline{i_0i_1i_2i_3i_4})$$

The terms such as i_0i_1 are for adjacent pairs of occupied chairs. The factor $\overline{i_0i_1i_2i_3i_4}$ forces the expression to be zero when all chairs are occupied.

(b) The statement below is not true. Explain why and correct it.

“By implementing a sum-of-products expression using only NAND gates (in place of AND and OR gates) we expose additional opportunities for simplification.”

Statement is incorrect because ...

... because the two expressions are essentially identical.

The real reason for using NAND gates is ...

... that in some device technologies, such as TTL, NAND gates are faster and less expensive than AND gates.