

Due: Friday, September 21 at 9:00 pm.

Submission: Please hand in a (tidy) paper copy at the beginning of class.

1. Use Karnaugh maps, and perhaps some additional algebra, to find minimal expressions for the following truth tables.

a	b	c	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

a	b	c	d	f
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

a	b	c	d	f
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

Turn page over for more problems...

2. Give an algebraic expression and a truth table that describes the circuit shown in problem 29 (p. 158) of the Null and Lobur text.

3. An encoder is a circuit that has  $2^n$  input lines, exactly one of which will be “on” at any given time. The circuit also has  $n$  output lines, that together represent a binary number indicating which input line is activated.

For example, consider an encoder with 4 input lines and 2 output lines. We can label the input lines  $x_0, x_1, x_2, x_3$ , and the output lines  $out_0$  and  $out_1$ . Then when  $x_1$  is activated,  $out_0$  (the lower-order output bit) should be 1, and  $out_1$  should be 0.

Provide each of the following for the 4:2 encoder just described.

- a truth table,
- a Karnaugh map and a *simplified* sum-of-products expression for each output line,
- a circuit diagram.

*Hint:* Use “don’t care” conditions in your Karnaugh maps.

4. Consider the function given by the following truth table. Provide a Karnaugh map, a sum-of-products expression, and a circuit diagram that uses only NAND gates to implement this function.

a	b	c	d	f
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

*Hint:* Convert your sum-of-products expression, via DeMorgan’s Laws, into an expression using only NAND and NOT operators. Then draw your circuit based on that expression.