EE 2310 Homework #2 – Digital Logic Circuits and Simplifying Logic with Karnaugh Maps

Name ___________________________________

1. **Given the truth table below, write its Boolean expression in SOP form. Then draw the logic circuit that represents the Boolean expression.**

x	y	z	f
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

2. **Draw the SOP circuit that represents the Boolean expression below.**

$$f = wxy\overline{z} + w\overline{x}yz + wxyz + w\overline{x}y\overline{z}$$

3. **Given the Boolean SOP function shown, simplify using non-Karnaugh-map techniques and draw the simplified SOP circuit.**

$$f = \overline{w}x\overline{y}z + \overline{w}xyz + wx\overline{y}z + wxyz$$

4. For the truth table below, write its Boolean expression in SOP form. Then draw the circuit. Using the techniques learned in class, simplify the circuit (do NOT use a K-map) to a simpler SOP form and draw that SOP circuit.

x	y	z	f
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

5. Write the Boolean expression in SOP form for the truth table to the right. Plot 1's on the Karnaugh Map and simplify the Boolean expression by determining prime implicants. Draw <u>both</u> the original and the simplified circuit.

w	x	y	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

xy

		00	**01**	**11**	**10**
w	**0**				
	1				

6. A logic function can be expressed as $f = \sum_m 0, 1, 8, 9$. Plot the parameters on the K-map below, write the simplified SOP expression, and draw the simplified SOP circuit.

wx \ yz	00	01	11	10
00				
01				
11				
10				

7. For the Karnaugh map at right, <u>write the simplified expression</u>, and draw the simplest possible circuit.

wx \ yz	00	01	11	10
00	1			1
01	1			
11	1			
10	1	1	1	1

8. For the Karnaugh map at right, write the <u>simplified</u> expression, and draw the simplified circuit (only).

wx \ yz	00	01	11	10
00		1		
01		1		
11		1	1	
10		1	1	

9. Consider the K-mapped SOP Boolean function $f = \sum_m 2,6,e$. The input variables operate in a restricted space. Input combinations are limited such that x can never be 0 when w is 1. Map both the 1's in the function and the special "don't care" conditions listed, develop the simplified SOP expression, and draw ONLY the simplified circuit.

yz

wx	00	01	11	10
00				
01				
11				
10				

10. This problem shows that sometimes the POS Boolean expression is the more convenient. For the truth table on the right, fill in the K-map and write both the original and simplified POS expressions, and draw the simplified circuit.

y+z

w+x	00	01	11	10
00				
01				
11				
10				

w	x	y	z	f
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

11. It is desired to multiplex four different input data lines, a-d, onto one output. Five address lines, ("v" [MSB] through "z" [LSB]) control input-to-output selection. The five-bit address can be stated as a hexadecimal number ranging from 0x00 to 0x1f. Input a is MUXed out on address 0x06, b on address 0x0f, c on 0x1b, and d on 0x1e. Draw the MUX circuit below.

12. The truth table for a 1-bit <u>subtract only circuit</u> is shown at the right. This circuit could be used in an n-bit subtractor by tying n of these 1-bit subtractors together. This is a full subtractor, which means that the inputs are x (the bit to be subtracted from), y (the bit that is subtracted from x), and borrow in (bi), the number that results when the adjacent column to the right has a y number bigger than x (or y and x are equal, but there is a borrow from the next-right column). Outputs are difference (D) and borrow out (bo). Fill in the two Karnaugh maps below, simplify, write the SOP expressions for D and bo, and draw the two SOP circuits.

bi	x	y	D	bo
0	0	0	0	0
0	0	1	1	1
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	0
1	1	1	1	1

Difference (D)

bi \ xy	00	01	11	10
0				
1				

Borrow Out (bo)

bi \ xy	00	01	11	10
0				
1				