

## HOMEWORK II

**DUE** : October 8, 2009

**READ** : Related portions of Chapter IV.

**ASSIGNMENT** : There are five questions four of which are from Chapter IV of the textbook.

**Solve all homework and exam problems as shown in class and past exam solutions.**

**1)** Solve Problem 4.6 (b).

You will obtain the minimal SOP expression. **Draw** the minimal circuit, assuming that **double-rail** inputs are available. Remember to write down the postulate or theorem used for the expression simplification, **not** the number of the postulate or theorem.

**2)** Solve Problem 4.7 (h).

The order of the inputs on the truth table is important. The order is (A, B, C, D). You will **show** an output column for each operator (AND, OR, NOT). Note that you will assume there is no NOR gate in the circuit and will **not** simplify the expression in this question.

**Draw** the circuit of the original expression given in the textbook :  $((A + B')' + C)' + D'$ , assuming that **single-rail** inputs are available and with AND, OR, NOT gates. **How many** gate levels does the original expression have ? Then **convert** the original circuit (with AND, OR, NOT) to a **minimal** circuit with only NAND gates, as done in class.

**3)** Solve Problem 4.9 (d).

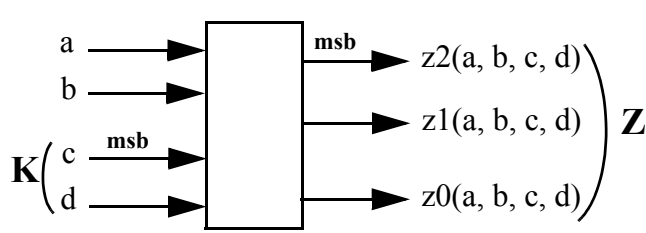
The question is asking you to obtain canonical SOP and POS expressions. **Show** also the missing minterm list. The order of inputs is (W, X, Y).

**4)** Solve Problem 4.10 (e).

**Solve** it for only the canonical SOP expression. Do **not** obtain the canonical POS expression. **Show** the minterm list.

The order of the inputs is (X, Y, Z). The expression in the problem has three terms, the last two of which are identical. **Use** relevant postulates and theorems to handle that. Each term in the expression is not canonical. Thus, they have to be **expanded** to include all the inputs of the function as done in class.

5) Consider the following combinational circuit with **four** inputs and **three** outputs :



K is a 2-bit **2's Complement Binary** number

Z is a 3-bit **2's Complement Binary** number

**Operation Table**

a	b	Operation
0	0	$Z = \overline{K}$
0	1	$Z = \overline{K}^2$
1	0	$Z = K + 1$
1	1	$Z = K - 1$

(i) Obtain the **truth table** of the combinational circuit based on the operation table. In a single sentence can you describe what this circuit does, i.e. its purpose ?

In order for K to have three bits so that it has the same bits as Z, assume that K has an **invisible** third (leftmost) bit whose value is obtained via a sign extension on K. Then perform the necessary operation on K. Name this invisible leftmost bit as "e" and show it on your truth table.

(ii) Then, obtain the **minterm lists** of the outputs from the truth table.

(iii) Then, obtain the **canonical SOP** expression of output **z2** as shown in class.

## RELEVANT QUESTIONS AND ANSWERS

**Q1)** Simplify the following expression by using Switching Algebra :

$$ab\bar{c} + (ab\bar{c} + \bar{a}c)[b(a+c) + \bar{b}c + a\bar{b}\bar{c}]$$

**A1)**

$$= ab\bar{c} + (ab\bar{c} + \bar{a}c)[ab + bc + \bar{b}c + a\bar{b}\bar{c}] \longrightarrow k(m+p) = km + kp$$

$$= ab\bar{c} + (ab\bar{c} + \bar{a}c)[ab + c + a\bar{b}\bar{c}] \longrightarrow k(m+p) = km + kp \ \& \ k + \bar{k} = 1 \ \& \ k1 = k$$

$$= ab\bar{c} + (ab\bar{c} + \bar{a}c)[ab + c + a\bar{b}] \longrightarrow k + \bar{k}m = k + m$$

$$= ab\bar{c} + (ab\bar{c} + \bar{a}c)[a + c] \longrightarrow k(m+p) = km + kp \ \& \ k + \bar{k} = 1 \ \& \ k1 = k$$

$$= ab\bar{c} + ab\bar{c} + \bar{a}c \longrightarrow k(m+p) = km + kp \ \& \ kk = k \ \& \ k\bar{k} = 0 \ \& \ k + 0 = k$$

$$= ab\bar{c} + \bar{a}c \longrightarrow k + k = k$$

**Q2)** Consider the following switching expression :

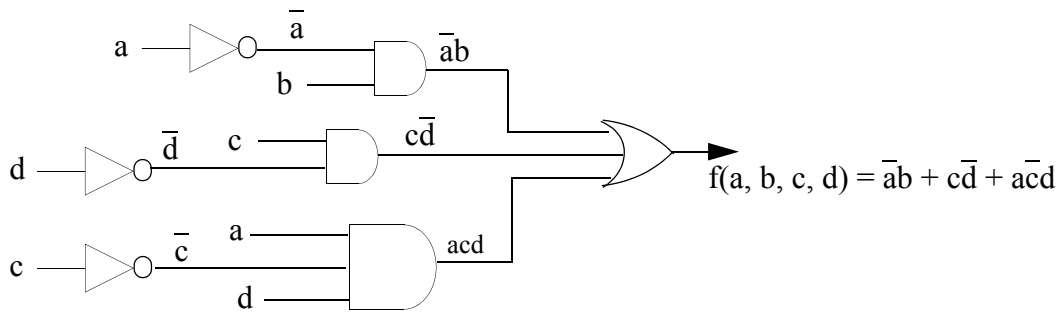
$$f(a, b, c, d) = \bar{a}(bc + b\bar{c}) + (\bar{c} + d) + \bar{a}b\bar{c}d + \bar{c}d(a + ac)$$

Simplify the expression by using Switching Algebra and then draw the corresponding 2-level AND/OR gate network, assuming there are only single-rail inputs.

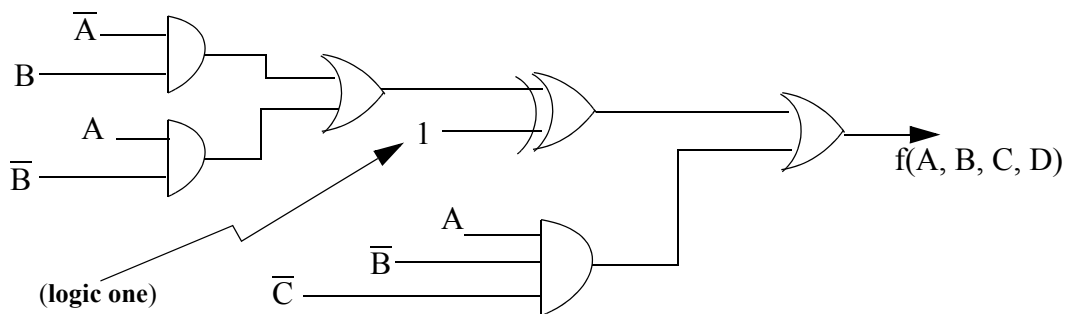
**A2)**

$$\begin{aligned} f(a, b, c, d) &= \bar{a}(bc + b\bar{c}) + (\bar{c} + d) + \bar{a}b\bar{c}d + \bar{c}d(a + ac) \\ &= \bar{a}(bc + b\bar{c}) + (\bar{c} + d) + \bar{a}b\bar{c}d + \bar{a}c\bar{d} \longrightarrow k + km = k \\ &= \bar{a}(b(c + \bar{c})) + (\bar{c} + d) + \bar{a}b\bar{c}d + \bar{a}c\bar{d} \longrightarrow k(m + p) = km + kp \\ &= \bar{a}b + (\bar{c} + d) + \bar{a}b\bar{c}d + \bar{a}c\bar{d} \longrightarrow k + \bar{k} = 1 \ \& \ k1 = k \\ &= \bar{a}b + \bar{c}d + \bar{a}b\bar{c}d + \bar{a}c\bar{d} \longrightarrow (\bar{k} + m) = \bar{k} \bar{m} \ \& \ \bar{k} = k \\ &= \bar{a}b + \bar{c}d(1 + \bar{a}b) + \bar{a}c\bar{d} \longrightarrow k(m + p) = km + kp \\ &= \bar{a}b + \bar{c}d + \bar{a}c\bar{d} \longrightarrow k + 1 = 1 \ \& \ k1 = k \end{aligned}$$

The minimal 2-level AND/OR gate network contains seven gates :

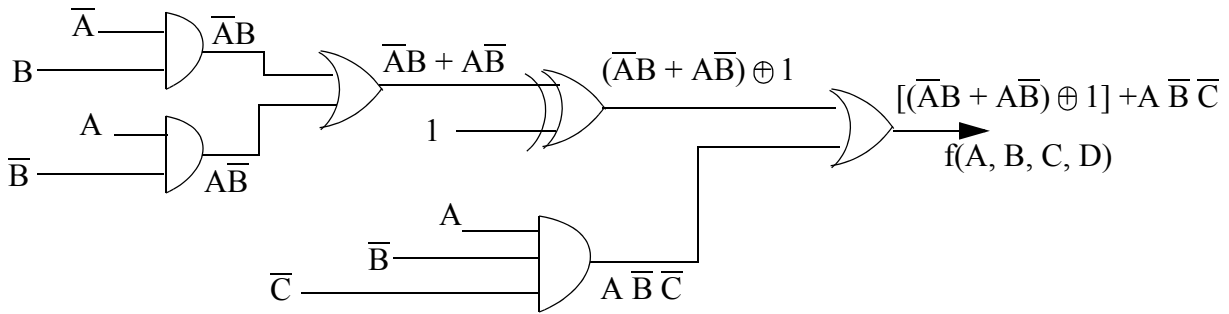


**Q3)** Consider the following gate network :



Simplify the gate network to obtain a minimal SOP expression, by using Switching Algebra.

**A3)** We first write down the expression at the output of each gate to obtain the switching expression :



We then minimize the switching expression :

$$\begin{aligned}
 f(A, B, C, D) &= [(\bar{A} B + A \bar{B}) \oplus 1] + A \bar{B} \bar{C} \\
 &= [k \oplus 1] + A \bar{B} \bar{C} \longrightarrow k = \bar{A} B + A \bar{B} \\
 &= \bar{k} 1 + k \bar{1} + A \bar{B} \bar{C} \longrightarrow k \oplus m = \bar{k} m + k \bar{m} \\
 &= \bar{k} + A \bar{B} \bar{C} \longrightarrow \bar{1} = 0 \ \& \ m 0 = 0 \ \& \ m 1 = m \ \& \ m + 0 = m \\
 &= \overline{(\bar{A} B) + (A \bar{B})} + A \bar{B} \bar{C} \longrightarrow k = \bar{A} B + A \bar{B} \\
 &= \overline{(\bar{A} B) (A \bar{B})} + A \bar{B} \bar{C} \longrightarrow \overline{(k + m)} = \bar{k} \bar{m} \\
 &= \overline{(\bar{A} + B)(A + \bar{B})} + A \bar{B} \bar{C} \longrightarrow \overline{(km)} = \bar{k} + \bar{m} \\
 &= \overline{(A + \bar{B})(\bar{A} + B)} + A \bar{B} \bar{C} \longrightarrow \bar{k} = k \\
 &= A \bar{A} + AB + \bar{A} \bar{B} + B \bar{B} + A \bar{B} \bar{C} \longrightarrow k(m + p) = km + kp \\
 &= AB + \bar{A} \bar{B} + A \bar{B} \bar{C} \longrightarrow k\bar{k} = 0 \ \& \ k + 0 = k \\
 &= \bar{B}(\bar{A} + A\bar{C}) + AB \longrightarrow k(m + p) = km + kp \\
 &= \bar{B}(\bar{A} + \bar{C}) + AB \longrightarrow k + \bar{k}m = k + m \\
 &= \bar{A} \bar{B} + \bar{B} \bar{C} + AB \longrightarrow k(m + p) = km + kp
 \end{aligned}$$

Note that there is **another minimal expression** :  $(\bar{A} \bar{B} + A \bar{C} + AB)$ , which is impossible to notice during the simplification. The fact that the algebra does not enable us to realize there are multiple minimal expressions, is one of main drawbacks of using Switching Algebra for circuit minimization.

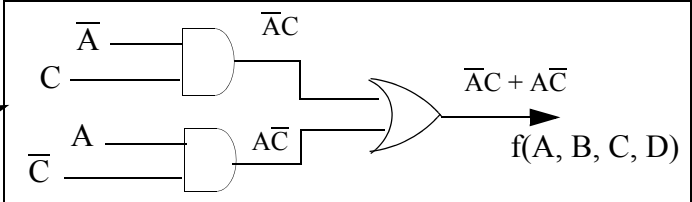
**Q4)** Simplify the following switching expression by using Switching Algebra as shown in class :

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

Then, draw the minimal 2-level AND/OR gate network, assuming that there are double-rail inputs.

**A4)**

$$\begin{aligned}
 &= \overline{(\overline{A C \overline{D}}) + (\overline{B \overline{C}})} (A + \overline{B \overline{D}}) + A \overline{B \overline{C}} + \overline{(A C + \overline{A \overline{C}})} + (A \overline{B} (\overline{C + D})) \\
 &= \overline{(\overline{A C \overline{D}}) (\overline{B \overline{C}})} (\overline{A} (\overline{B \overline{D}})) + A \overline{B \overline{C}} + ((\overline{A C}) (\overline{A \overline{C}})) + (A \overline{B} (\overline{C \overline{D}})) \longrightarrow \overline{(k m)} = \overline{k} + \overline{m} \ \& \ \overline{(k + m)} = \overline{k} \overline{m} \\
 &= (A C \overline{D}) (\overline{B \overline{C}}) (\overline{A} (\overline{B \overline{D}})) + A \overline{B \overline{C}} + ((\overline{A C}) (\overline{A \overline{C}})) + (A \overline{B} (\overline{C \overline{D}})) \longrightarrow \overline{k} = k \\
 &= (A C \overline{D}) (\overline{B} + \overline{C}) (\overline{A} \overline{B \overline{D}}) + A \overline{B \overline{C}} + (\overline{A} + \overline{C})(\overline{A} + \overline{C}) + A \overline{B \overline{C} \overline{D}} \longrightarrow \overline{(k m)} = \overline{k} + \overline{m} \ \& \ \overline{(k + m)} = \overline{k} \overline{m} \\
 &= (A C \overline{D}) (B + C) (\overline{A} (\overline{B \overline{D}}) + A \overline{B \overline{C}} + (\overline{A} + \overline{C})(A + C) + A \overline{B \overline{C} \overline{D}} \longrightarrow \overline{k} = k \\
 &= (A B C \overline{D} + A C \overline{D})(\overline{A} \overline{B \overline{D}}) + A \overline{B \overline{C}} + \overline{A} \overline{A} C + A \overline{C} + C \overline{C} + A \overline{B \overline{C} \overline{D}} \longrightarrow k(m + p) = km + kp \ \& \ kk = k \\
 &= (A B C \overline{D} + A C \overline{D})(\overline{A} \overline{B \overline{D}}) + A \overline{B \overline{C}} + \overline{A} C + A \overline{C} + A \overline{B \overline{C} \overline{D}} \longrightarrow k \overline{k} = 0 \ \& \ k + 0 = k \\
 &= (A \overline{A} B \overline{C} D \overline{D} + A \overline{A} \overline{B} C D \overline{D}) + A \overline{B \overline{C}} + \overline{A} C + A \overline{C} + A \overline{B \overline{C} \overline{D}} \longrightarrow k(m + p) = km + kp \\
 &= A \overline{B \overline{C}} + \overline{A} C + A \overline{C} + A \overline{B \overline{C} \overline{D}} \longrightarrow k \overline{k} = 0 \ \& \ k + 0 = k \ \& \ k k = k \\
 &= A \overline{B \overline{C}} + \overline{A} C + A \overline{C} \longrightarrow k + km = k \\
 &= \overline{A} C + A \overline{C} \longrightarrow k + km = k
 \end{aligned}$$



**Q5)** Consider the following minimal SOP expression :

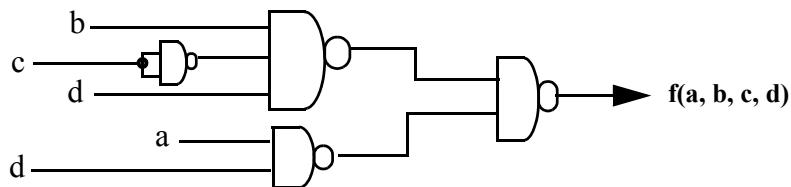
$$f(a, b, c, d) = b \overline{c} d + a d$$

- i) Draw the corresponding 2-level NAND-NAND gate network, assuming **single-rail** inputs and as done **in class**.
- ii) Obtain the canonical SOP expression of the function **algebraically** as done in class.
- iii) Obtain the minterm and maxterm lists of the function.

**A5) i)**  $f(a, b, c, d) = b \overline{c} d + a d$

We know that an SOP expression is implemented by a 2-level AND-OR gate network.

We also know that a 2-level AND-OR gate network is immediately implemented by a 2-level NAND-NAND gate network :



ii)  $f(a, b, c, d) = \overline{bcd} + ad$

$$\begin{aligned}
 &= \overline{bcd}(a + \overline{a}) + ad(b + \overline{b})(c + \overline{c}) && k + \overline{k} = 1 \text{ \& } k1 = k \\
 &= \overline{abcd} + \overline{a}\overline{bcd} + ad(bc + b\overline{c} + \overline{b}c + \overline{b}\overline{c}) && k(m+p) = km + kp \\
 &= \overline{abcd} + \overline{a}\overline{bcd} + abcd + \overline{a}\overline{bcd} + \overline{a}\overline{bcd} + \overline{a}\overline{b}\overline{c}d && k(m+p) = km + kp \\
 &= \overline{abcd} + \overline{a}\overline{bcd} + abcd + \overline{a}\overline{bcd} + \overline{a}\overline{b}\overline{c}d && k + k = k
 \end{aligned}$$

iii)

$\overline{abcd}$ 1101 └──┬──┘ 13	$\overline{a}\overline{bcd}$ 0101 └──┬──┘ 5	$abcd$ 1111 └──┬──┘ 15	$\overline{abcd}$ 1011 └──┬──┘ 11	$\overline{a}\overline{b}\overline{c}d$ 1001 └──┬──┘ 9
--	--	---------------------------------	--	---

$$\begin{aligned}
 f(a,b,c,d) &= \sum m(5,9,11,13,15) \\
 f(a,b,c,d) &= \prod M(0,1,2,3,4,6,7,8,10,12,14)
 \end{aligned}$$

Q6) Consider the following expression :

$$f(a, b, c, d) = \overline{(a + b)} + c(\overline{d(a + a)} + (\overline{bb})) + \overline{c}(ab + \overline{a} + \overline{b})d$$

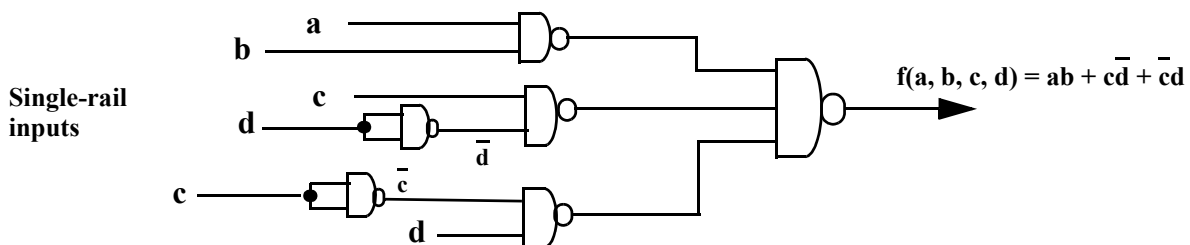
(i) Simplify the expression to obtain the **minimal SOP** expression by using **Switching Algebra** as shown in class.

(ii) Then, draw the corresponding minimal 2-level **NAND-NAND** gate network, by assuming **single-rail** inputs.

A6) i) The simplification to obtain the minimal SOP expression :

$$\begin{aligned}
 f(a, b, c, d) &= \overline{(a + b)} + c(\overline{d(a + a)} + (\overline{bb})) + \overline{c}(ab + \overline{a} + \overline{b})d \\
 &= \overline{(a + b)} + c(\overline{d} \mathbf{1} + \mathbf{0}) + \overline{c}(ab + \overline{a} + \overline{b})d && k + \overline{k} = 1 \text{ \& } k\overline{k} = 0 \\
 &= \overline{(a + b)} + c\overline{d} + \overline{c}(ab + \overline{a} + \overline{b})d && k1 = k \text{ \& } k + 0 = k \\
 &= \overline{(a + b)} + c\overline{d} + \overline{c}(\mathbf{b} + \overline{\mathbf{a}} + \overline{\mathbf{b}})d && k + \overline{k}m = k + m \\
 &= \overline{(a + b)} + c\overline{d} + \overline{c}d && k + \overline{k} = 1 \text{ \& } k1 = k \\
 &= \overline{(a \mathbf{b})} + c\overline{d} + \overline{c}d && (\overline{\overline{k + m}}) = \overline{\overline{k}} + \overline{\overline{m}} \\
 &= ab + c\overline{d} + \overline{c}d && \overline{\overline{k}} = k
 \end{aligned}$$

ii) We know that an SOP expression is directly implemented by a 2-level AND-OR gate network and a 2-level AND-OR gate network can be immediately converted to a 2-level NAND-NAND gate network :



**Q7)** By using a truth table, show if the following two expressions are equivalent :

$$f(a, b, c) = (\bar{a} + ((b+c) a))$$

$$g(a, b, c) = (a + b + c)$$

**A7)**

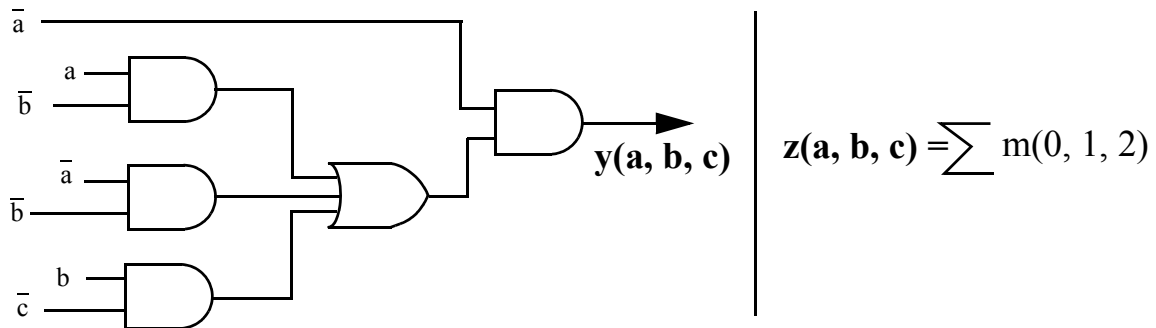
a	b	c	3 ↓ 4 ↓ ( $\bar{a} + ((b+c) a)$ )	1 ↓ 2 ↓ ( $(b+c) a$ )	5 ↓ ( $a + b + c$ )
0	0	0	1	0	0
0	0	1	1	1	1
0	1	0	1	1	1
0	1	1	1	1	1
1	0	0	0	0	1
1	0	1	0	1	1
1	1	0	0	1	1
1	1	1	0	1	1

On the output columns section, there is a **column for each operation**. The order of obtaining the output columns is based on the precedence rules and is shown by the numbered arrows.

Function  $f(a, b, c)$  is column 4. Function  $g(a, b, c)$  is column 5.

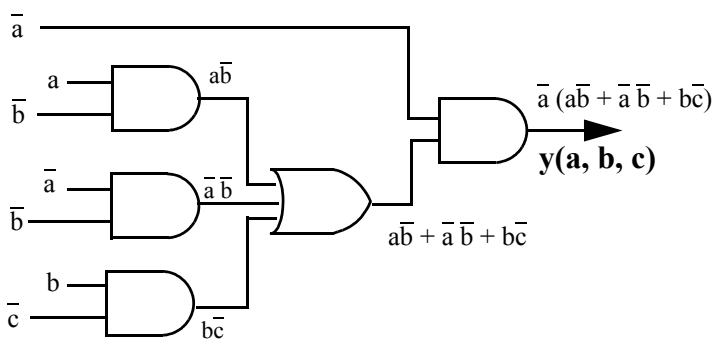
Since the  $f(a, b, c)$  and  $g(a, b, c)$  columns are **not** equivalent, the two functions are **not** equivalent..

**Q8)** Determine if the following two functions,  $y(a, b, c)$ ,  $z(a, b, c)$ , are equivalent :



$$z(a, b, c) = \sum m(0, 1, 2)$$

**A8)** We obtain the switching expression for the  $y(a, b, c)$  function. In order to do that we first place the term that corresponds to the output of each gate and the full expression for function  $y(a, b, c)$  :



$$y(a, b, c) = \bar{a} (a \bar{b} + \bar{a} \bar{b} + b \bar{c})$$

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

$$k(m + p) = km + kp$$

$$= \bar{a} (\bar{b} (1) + b \bar{c})$$

$$k + \bar{k} = 1$$

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

$$k1 = k$$

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

$$k + \bar{k}m = k + m$$

$$= \bar{a} \bar{b} + \bar{a} \bar{c}$$

$$k(m + p) = km + kp$$

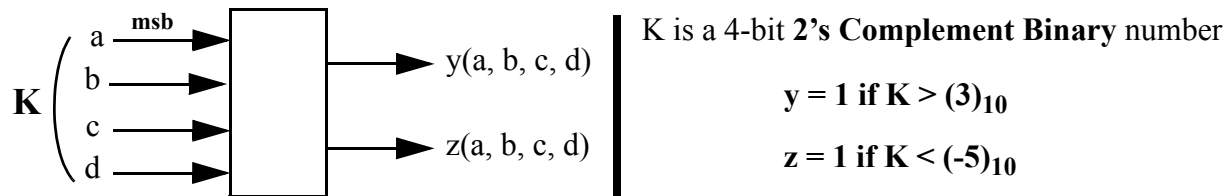
The minimal SOP expression can now be converted to a canonical SOP expression :

$$\begin{aligned}
 &= \bar{a}\bar{b}(c + \bar{c}) + \bar{a}\bar{c}(b + \bar{b}) && k + \bar{k} = 1 \text{ \& } k1 = k \\
 &= \bar{a}\bar{b}c + \bar{a}\bar{b}\bar{c} + \bar{a}b\bar{c} + \bar{a}\bar{b}\bar{c} && k(m + p) = km + kp \\
 &= \bar{a}\bar{b}c + \bar{a}\bar{b}\bar{c} + \bar{a}b\bar{c} && k + k = k
 \end{aligned}$$

The above three canonical product terms correspond to minterms 1, 0 and 2, respectively :

$$\left. \begin{array}{ccc} \bar{a}\bar{b}c & + & \bar{a}\bar{b}\bar{c} & + & \bar{a}b\bar{c} \\ \underbrace{001}_1 & & \underbrace{000}_0 & & \underbrace{010}_2 \end{array} \right\} y(a,b,c) = \sum m(0,1,2) \quad \left| \quad \begin{array}{l} \text{given that} \\ z(a,b,c) = \sum m(0,1,2) \end{array} \right\} y(a, b, c) = z(a, b, c)$$

**Q9)** Consider the following combinational circuit with **four** inputs and **two** outputs :



- (i) Obtain the **truth table** of the circuit based on the textual input/output relationship.
- (ii) Then, obtain the **minterm lists** of the outputs from the truth table.
- (iii) Then, obtain the **canonical SOP** expression of output **y(a, b, c, d)** as shown in class.

**A9)** The truth table and the minterm lists :

K	K	abcd				y	z
		a	b	c	d		
0	0	0	0	0	0	0	
1	1	0	0	0	1	0	
2	2	0	0	1	0	0	
3	3	0	0	1	1	0	
4	4	0	1	0	0	1	
5	5	0	1	0	1	0	
6	6	0	1	1	0	1	
7	7	0	1	1	1	0	
-8	8	1	0	0	0	1	
-7	9	1	0	0	1	0	
-6	10	1	0	1	0	1	
-5	11	1	0	1	1	0	
-4	12	1	1	0	0	0	
-3	13	1	1	0	1	0	
-2	14	1	1	1	0	0	
-1	15	1	1	1	1	0	

**The Minterm lists :**

$$y(a, b, c, d) = \sum m(4, 5, 6, 7)$$

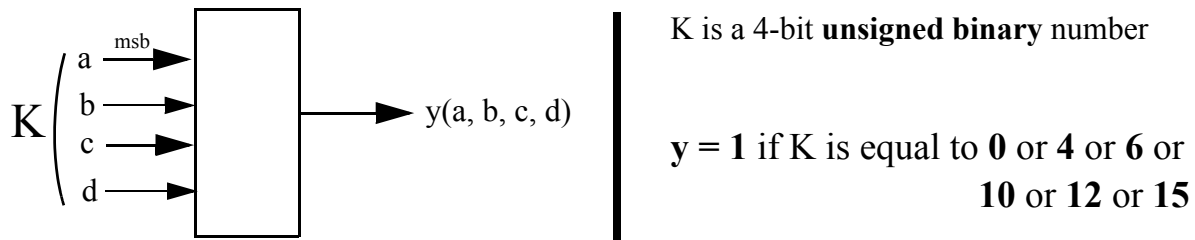
$$z(a, b, c, d) = \sum m(8, 9, 10)$$

**The canonical SOP expression for y(a, b, c, d) :**

$$\begin{array}{c|c|c|c}
 \frac{4}{0100} & \frac{5}{0101} & \frac{6}{0110} & \frac{7}{0111} \\
 \hline
 \bar{a}b\bar{c}\bar{d} & \bar{a}b\bar{c}d & \bar{a}bc\bar{d} & \bar{a}bcd
 \end{array}$$

$$y(a, b, c, d) = \bar{a}b\bar{c}\bar{d} + \bar{a}b\bar{c}d + \bar{a}bc\bar{d} + \bar{a}bcd$$

**Q10)** The black-box view and purpose of a *special purpose comparator* circuit are shown below :



- i) Obtain the truth table of the function  $y(a, b, c, d)$  as done in class.
- ii) Obtain the minterm list of the function as done in class.
- iii) Consider the following minimal SOP expression :

$$f(a, b, c, d) = \bar{a} \bar{d} + a d$$

Prove/disprove if function  $f(a, b, c, d)$  is equivalent to function  $y(a, b, c, d)$  by first obtaining the truth table of  $f(a, b, c, d)$  as done in class and then the minterm list of  $f(a, b, c, d)$  from its truth table.

**A10)** Truth tables and minterm lists :

a b c d	$y(a,b,c,d)$	$\bar{a}$	$\bar{d}$	$\bar{a} \bar{d}$	$ad$	$\bar{a} \bar{d} + ad$
0 0 0 0	1	1	1	1	0	1
1 0 0 0	0	1	0	0	0	0
2 0 0 1	0	1	1	1	0	1
3 0 0 1	0	1	0	0	0	0
4 0 1 0	1	1	1	1	0	1
5 0 1 0	0	1	0	0	0	0
6 0 1 1	1	1	1	1	0	1
7 0 1 1	0	1	0	0	0	0
8 1 0 0	0	0	1	0	0	0
9 1 0 0	0	0	0	0	1	1
10 1 0 1	1	0	1	0	0	0
11 1 0 1	0	0	0	0	1	1
12 1 1 0	1	0	1	0	0	0
13 1 1 0	0	0	0	0	1	1
14 1 1 1	0	0	1	0	0	0
15 1 1 1	1	0	0	0	1	1

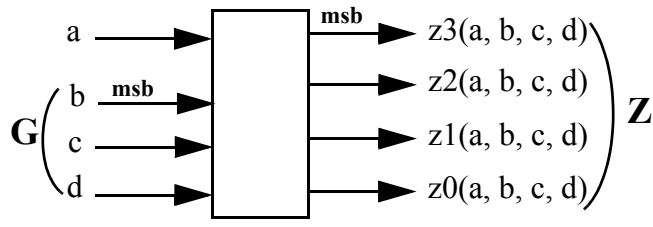
The Minterm list :  
 $y(a,b,c,d) = \sum m(0,4,6,10,12,15)$

The minterm list for  $f(a, b, c, d)$  :  
 $f(a,b,c,d) = \sum m(0,2,4,6,9,11,13,15)$

Since the two functions do **NOT** have identical minterm lists, they are **NOT** equivalent to each other !

**Q11)** Consider the combinational circuit with **four** inputs and **four** outputs below.

- (i) Obtain the **truth table** of the combinational circuit based on the textual input/output relationship.



G is a 3-bit 2's Complement Binary number  
 Z is a 4-bit 2's Complement Binary number

If a = 0 then Z = G - 1  
 else Z = G + 1

In order for G to have four bits so that it has the same bits as Z, assume that G has an **invisible** fourth (leftmost) bit whose value is obtained via a sign extension on G. Then perform the necessary operation on G. Name this invisible leftmost bit as "e" and show it on your truth table.

(ii) Then, obtain the **minterm lists** of the outputs from the truth table.

(iii) Then, obtain the **canonical SOP** expression of output **z3** as shown in class.

**A11)** The truth table, the minterm lists and the canonical SOP expression :

G	Z	G			e	Z			
		a	b	c d		z3	z2	z1	z0
0	-1	<b>0</b>	0	0 0 0	0	1	1	1	1
1	0	<b>1</b>	0	0 0 1	0	0	0	0	0
2	1	<b>2</b>	0	0 1 0	0	0	0	0	1
3	2	<b>3</b>	0	0 1 1	0	0	0	1	0
-4	-5	<b>4</b>	0	1 0 0	1	1	0	1	1
-3	-4	<b>5</b>	0	1 0 1	1	1	1	0	0
-2	-3	<b>6</b>	0	1 1 0	1	1	1	0	1
-1	-2	<b>7</b>	0	1 1 1	1	1	1	1	0
0	1	<b>8</b>	1	0 0 0	0	0	0	0	1
1	2	<b>9</b>	1	0 0 1	0	0	0	1	0
2	3	<b>10</b>	1	0 1 0	0	0	0	1	1
3	4	<b>11</b>	1	0 1 1	0	0	1	0	0
-4	-3	<b>12</b>	1	1 0 0	1	1	1	0	1
-3	-2	<b>13</b>	1	1 0 1	1	1	1	1	0
-2	-1	<b>14</b>	1	1 1 0	1	1	1	1	1
-1	0	<b>15</b>	1	1 1 1	1	0	0	0	0

The Minterm lists :

$$z3(a,b,c,d) = \sum m(0, 4, 5, 6, 7, 12, 13, 14)$$

$$z2(a,b,c,d) = \sum m(0, 5, 6, 7, 11, 12, 13, 14)$$

$$z1(a,b,c,d) = \sum m(0, 3, 4, 7, 9, 10, 13, 14)$$

$$z0(a,b,c,d) = \sum m(0, 2, 4, 6, 8, 10, 12, 14)$$

The canonical SOP expression for z3 :

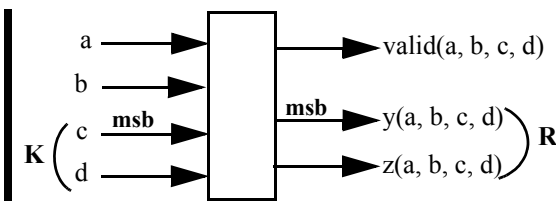
0 0 0 0 0 $\bar{a} \bar{b} \bar{c} \bar{d}$	4 0 1 0 0 $\bar{a} b \bar{c} \bar{d}$	5 0 1 0 1 $\bar{a} b \bar{c} d$	6 0 1 1 0 $\bar{a} b c \bar{d}$
7 0 1 1 1 $\bar{a} b c d$	12 1 1 0 0 $a b \bar{c} \bar{d}$	13 1 0 1 1 $a \bar{b} c d$	14 1 1 1 0 $a b c \bar{d}$

$$z3(a, b, c, d) = \bar{a} \bar{b} \bar{c} \bar{d} + \bar{a} b \bar{c} \bar{d} + \bar{a} b \bar{c} d + \bar{a} b c \bar{d} + \bar{a} b c d + a b \bar{c} \bar{d} + a \bar{b} c d + a b c \bar{d}$$

**Q12)** Consider the combinational circuit with **four** inputs and **three** outputs below.

(i) Obtain the **truth table** of the circuit based on the operation table.

**K & R are 2's Complement**  
Binary numbers



a	b	Operation
0	0	$R = K + 1$ ; valid = 0 if overflow
0	1	$R = K - 1$ ; valid = 0 if overflow
1	0	$y = d$ ; $z = c$ ; valid = 1
1	1	$y = d$ ; $z = 0$ ; valid = 1

(ii) Then, obtain the **minterm lists** of the outputs from the truth table.

(iii) Then, obtain the **canonical SOP** expression of output **z(a, b, c, d)** as shown in class.

**A12)** The truth table and the minterm lists :

K		$\overbrace{a\ b\ c\ d}^K$	valid	y	z
0	<b>0</b>	0000	1	0	1
1	<b>1</b>	0001	0	1	0
-2	<b>2</b>	0010	1	1	1
-1	<b>3</b>	0011	1	0	0
0	<b>4</b>	0100	1	1	1
1	<b>5</b>	0101	1	0	0
-2	<b>6</b>	0110	0	0	1
-1	<b>7</b>	0111	1	1	0
0	<b>8</b>	1000	1	0	0
1	<b>9</b>	1001	1	1	0
-2	<b>10</b>	1010	1	0	1
-1	<b>11</b>	1011	1	1	1
0	<b>12</b>	1100	1	0	0
1	<b>13</b>	1101	1	1	0
-2	<b>14</b>	1110	1	0	0
-1	<b>15</b>	1111	1	1	0

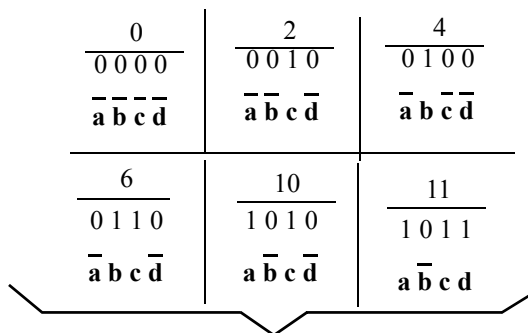
The Minterm lists :

$$\text{valid}(a, b, c, d) = \sum m(0, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15)$$

$$y(a, b, c, d) = \sum m(1, 2, 4, 7, 9, 11, 13, 15)$$

$$z(a, b, c, d) = \sum m(0, 2, 4, 6, 10, 11)$$

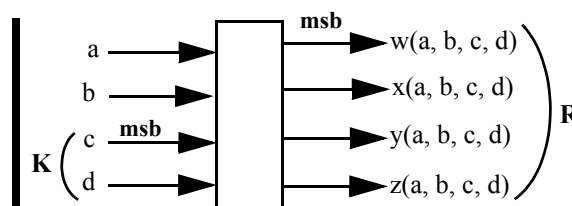
The canonical SOP expression for **z(a, b, c, d)** :



$$z(a, b, c, d) = \bar{a}\bar{b}\bar{c}\bar{d} + \bar{a}\bar{b}c\bar{d} + \bar{a}b\bar{c}\bar{d} + \bar{a}bc\bar{d} + a\bar{b}c\bar{d} + a\bar{b}cd$$

**Q13)** Consider the following combinational circuit with **four** inputs and **four** outputs :

**K & R are 2's Complement**  
Binary numbers



a	b	Operation
0	0	$R = -K$ (negate K)
0	1	$R = 2 * K$ (two times K)
1	0	$R = 4 * K$ (four times K)
1	1	$R = K * K$ (K times K)

(i) Obtain the **truth table** of the circuit based on the operation table. Use **sign extensions** to obtain 4 bits of R from 2 bits of K as done in the homework.

(ii) Then, obtain the **minterm lists** of the outputs from the truth table.

(iii) Then, obtain the **canonical SOP** expression of output  $z(a, b, c, d)$  as shown in class.

**A13)** The truth table and the minterm lists :

K		K				e	f	w	x	y	z
		a	b	c	d						
0	<b>0</b>	0	0	0	0	0	0	0	0	0	
1	<b>1</b>	0	0	0	1	0	1	1	1	1	
-2	<b>2</b>	0	0	1	0	1	0	0	1	0	
-1	<b>3</b>	0	0	1	1	0	0	0	1	1	
0	<b>4</b>	0	1	0	0	0	0	0	0	0	
1	<b>5</b>	0	1	0	1	0	0	0	1	0	
-2	<b>6</b>	0	1	1	0	1	1	0	0	0	
-1	<b>7</b>	0	1	1	1	1	1	1	0	0	
0	<b>8</b>	1	0	0	0	0	0	0	0	0	
1	<b>9</b>	1	0	0	1	0	0	1	0	0	
-2	<b>10</b>	1	0	1	0	1	0	0	0	0	
-1	<b>11</b>	1	0	1	1	1	1	0	0	0	
0	<b>12</b>	1	1	0	0	0	0	0	0	0	
1	<b>13</b>	1	1	0	1	0	0	0	1	1	
-2	<b>14</b>	1	1	1	0	0	1	0	0	0	
-1	<b>15</b>	1	1	1	1	0	0	0	1	1	

The Minterm lists :

$$w(a, b, c, d) = \sum m(1, 6, 7, 10, 11)$$

$$x(a, b, c, d) = \sum m(1, 6, 7, 9, 11, 14)$$

$$y(a, b, c, d) = \sum m(1, 2, 5, 7)$$

$$z(a, b, c, d) = \sum m(1, 3, 13, 15)$$

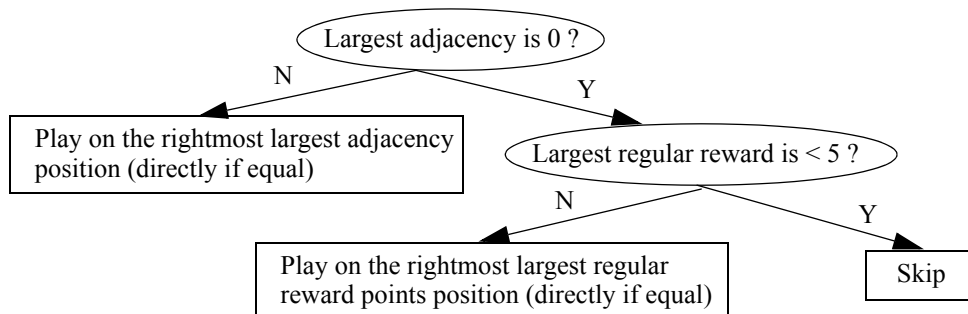
The canonical SOP expression for  $z(a, b, c, d)$  :

$$\begin{array}{c|c|c|c} \frac{1}{0001} & \frac{3}{0011} & \frac{13}{1101} & \frac{15}{1111} \\ \hline \bar{a}\bar{b}\bar{c}d & \bar{a}\bar{b}cd & ab\bar{c}d & abcd \end{array}$$

$$z(a, b, c, d) = \bar{a}\bar{b}\bar{c}d + \bar{a}\bar{b}cd + ab\bar{c}d + abcd$$

Since input K has two bits and output R has 4 bits, we sign extend input K by two bits. These new leftmost two bits of K are “e” and “f” and their values are shown on the truth table above.

**Q14)** Consider the **Ppm** term project. The flowchart of the playing strategy of a machine player is as follows :



Consider the following table that shows the random digit, position displays **before** and **after** the **machine** player plays, whether the random digit is played directly or added, the number of adjacencies, the points earned by the **machine** player and whether the machine player plays again :

RD	Displays Before Play PD3 PD2 PD1 PD0	Displays After Play PD3 PD2 PD1 PD0	D/A	The Adjacency	Points Earned (Decimal)	Machine player plays again
8	C 4 C 4	C (C) C 4	A	2	48	Yes
9	E 5 9 9					
3	F F F F					
1	6 A A 1					
4	E A 4 A					
7	7 7 7 E					

Assume that the code is **44**. The first row shows how the random digit is played by the **machine** player. A circle is drawn on a position if it is played on. The meaning of **D/A** is Direct/Add which is whether the player plays the random digit **directly** on a position or by **adding** to a position. Note that the cases are **independent** of each other. That is, they do not necessarily follow each other with respect to time. **Continue with the remaining rows.**

**A14)**

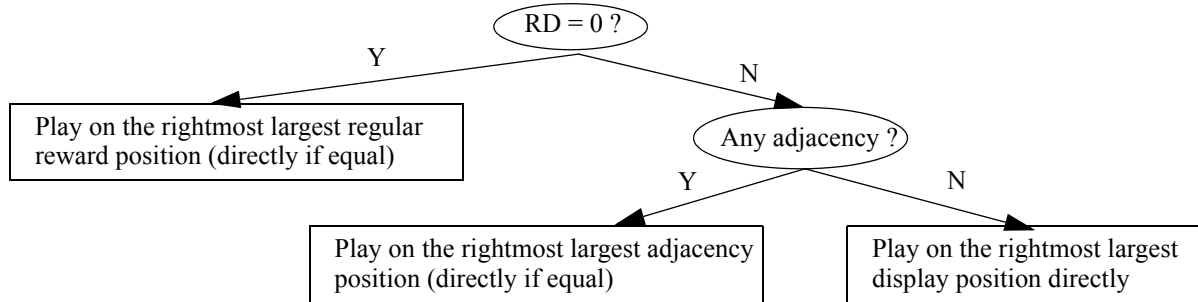
RD	Displays Before Play PD3 PD2 PD1 PD0	Displays After Play PD3 PD2 PD1 PD0	D/A	The Adjacency	Points Earned (Decimal)	Machine player plays again
8	C 4 C 4	C (C) C 4	A	2	48	Yes
9	E 5 9 9	E (9) 9 9	D	2	36	Yes
3	F F F F	F F F F	Skip	Skip	Skip	Skip
1	6 A A 1	6 A (1) 1	D	1	2	Yes
4	E A 4 A	E A 4 (4)	D	1	76	Yes
7	7 7 7 E	7 7 7 (7)	D	3	56	Yes

Note that the machine player does not check for code digits and so misses a large reward point on row 1. The random digit on this row enable it to play a code digit. The machine player earns a large amount of reward points on row 5 by playing code digit accidentally.

**Q15)** Consider the **Ppm** term project. The flowchart of the playing strategy of an imaginary machine player is below.

Consider the table below that shows the random digit, position displays **before** and **after** the **machine** player plays, whether the random digit is played directly or added, the number of adjacencies, the points earned by the **machine** player and whether the machine player plays again.

Assume that the code is **E2**. The first row shows how the random digit is played by the **machine** player. A circle is drawn on a position if it is played on. The meaning of **D/A** is Direct/Add which is whether the player plays the random digit **directly** on a position or by **adding** to a position.



RD	Displays Before Play PD3 PD2 PD1 PD0	Displays After Play PD3 PD2 PD1 PD0	D/A	The Adjacency	Points Earned (Decimal)	Machine player plays again
4	A C 4 3	A C 4 (4)	D	1	8	Yes
2	8 6 2 F					
0	F 1 E 1					
9	E F 5 C					
7	F 9 7 2					
8	2 E 6 2					

Note that the cases are **independent** of each other. That is, they do not necessarily follow each other with respect to time. **Continue with the remaining rows.**

### A15)

RD	Displays Before Play PD3 PD2 PD1 PD0	Displays After Play PD3 PD2 PD1 PD0	D/A	The Adjacency	Points Earned (Decimal)	Machine player plays again
4	A C 4 3	A C 4 (4)	D	1	8	Yes
2	8 6 2 F	8 6 2 (2)	D	1	20	Yes
0	F 1 E 1	(F) 1 E 1	A	0	15	No
9	E F 5 C	E F 5 (5)	D	1	10	Yes
7	F 9 7 2	F 9 7 (7)	D	1	14	Yes
8	2 E 6 2	2 E (E) 2	A	1	254	Yes

The machine player strategy does not check for code digits and so misses to earn code reward points when the random digit is 9 and 0 above. On the other hand, by chance it earns code reward points when the random digit is 2 and 8 above