[CS M51A Winter 17] Solutions for Midterm exam

Date: 2/16/17

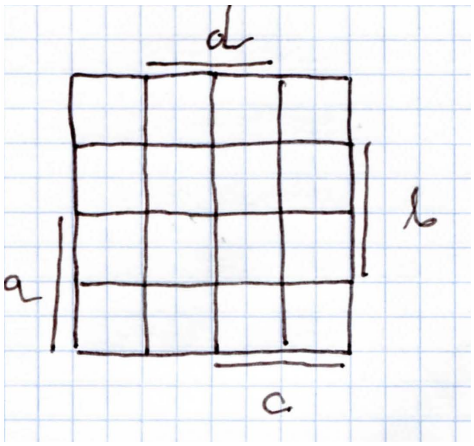
Problem 1 (20 points)

1. (8 points) Using algebraic identities obtain a simplified sum of product for the following switching expression: $E_1(a,b,c,d) = (ad' \oplus b')(c+d) + (a'+bc)'cd'$
   Show each step of your work on a separate line and indicate which identity was used.
   Solution

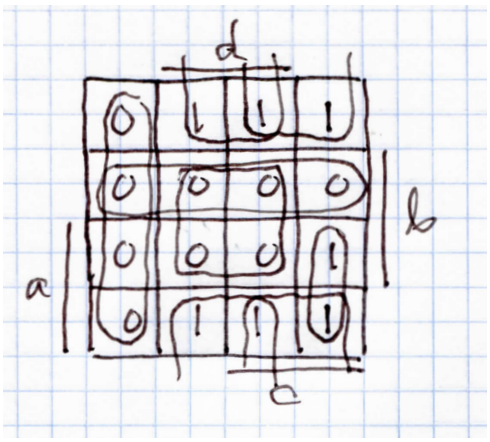
$$
\begin{aligned}
E_1(a,b,c,d) &= (ad' \oplus b')(c+d) + (a'+bc)'cd' && (1)\\
&= (abd' + (a'+d)b')(c+d) + a(bc)'cd' && (2)\\
&= (abd' + a'b' + b'd)(c+d) + a(b'+c')cd' && (3)\\
&= abcd' + a'b'c + b'cd + a'b'd + b'd + ab'cd' && (4)\\
&= abcd' + b'c(a'+d+ad') + b'd(a'+1) && (5)\\
&= abcd' + b'c + b'd && (6)\\
&= c(abd' + b') + b'd && (7)\\
&= c(ad' + b') + b'd && (8)\\
&= acd' + b'c + b'd && (9)
\end{aligned}
$$

(1) → (2) def of XOR; de Morgan
(2) → (3) Distributivity; de Morgan
(3) → (4) Distributivity; P4(ii)
(4) → (5) Distributivity; Complement
(5) → (6) Distributivity
(6) → (7) Distributivity
(7) → (8) Simplification
(8) → (9) Distributivity

2. (4 points) Using a K-map, obtain minimal sum of products and product of sums. Compare the minimal SOP with the SOP in (1).



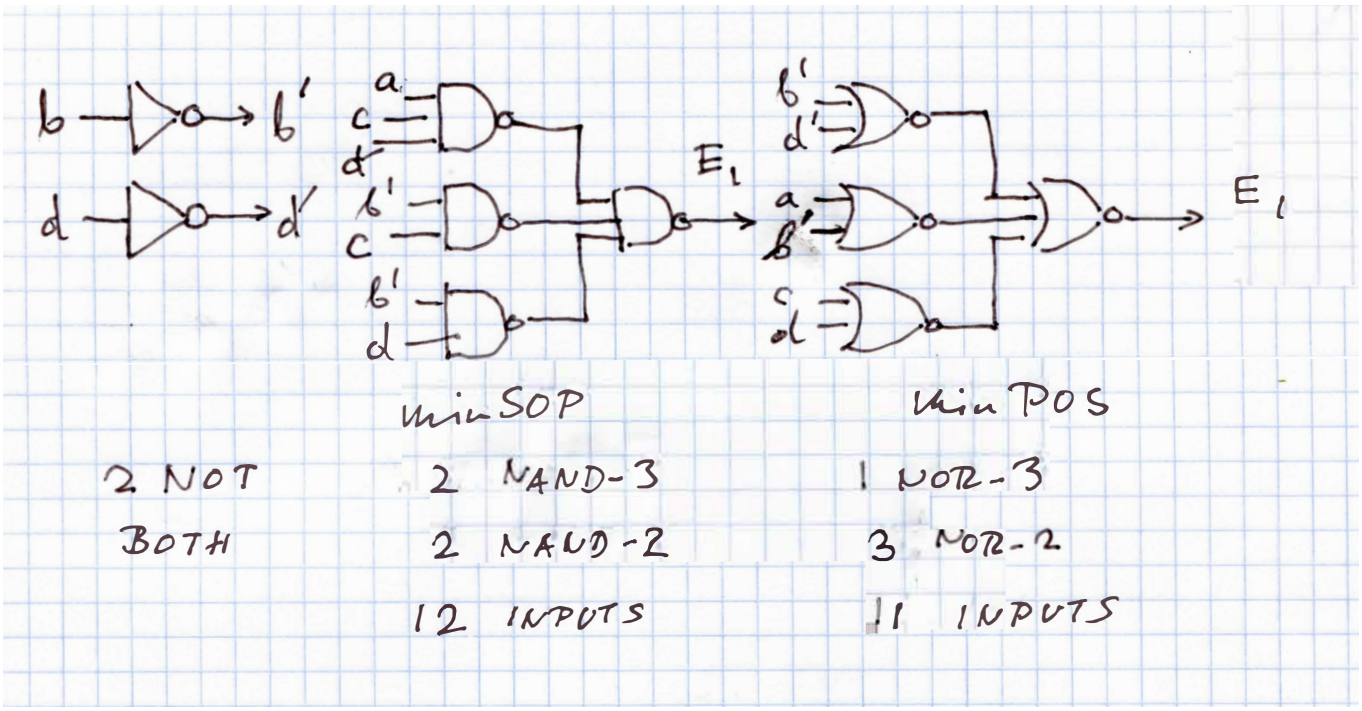
Solution



$E_2 = acd' + b'c + b'd$ same as E_1 min SOP
$E_2 = (b' + d')(a + b')(c + d)$ min POS

3. (8 points) Show implementation of min SOP and min POS expressions using NAND and NOR gates. Inverted inputs are not available, and no constant inputs are allowed. Compare the two networks with respect to the number of gates and the total number of inputs. (You are allowed to use NOT gates.)

Solution



Cost (POS) < Cost (SOP), the POS network has one less input.

Problem 2 (15 points)

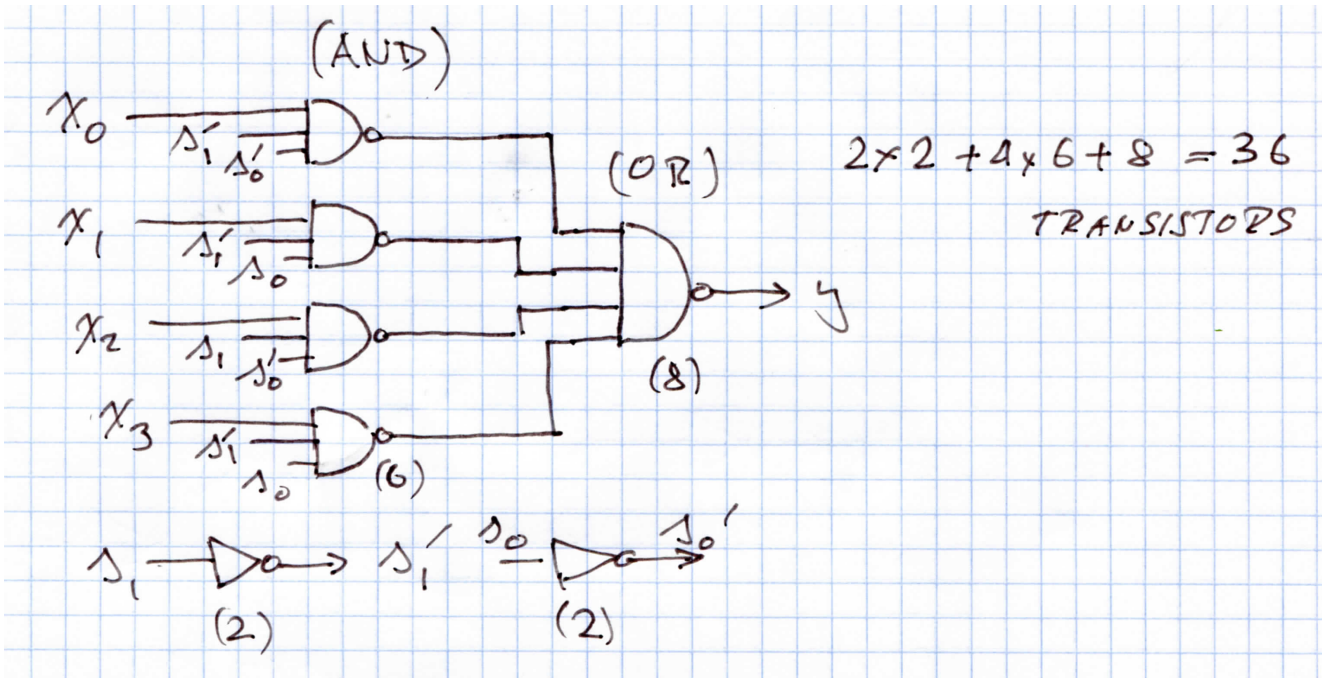
We want to design a gate network to implement a 4-input multiplexer module MUX. This module has four data inputs $\underline{x} = (x_3, x_2, x_1, x_0)$, two select inputs $\underline{s} = (s_1, s_0)$ and the output y, all in binary code. In other words, the output is connected to one of the data inputs determined by the select inputs. Formally, the MUX function is specified as

$y = MUX(\underline{x}, \underline{s}) = x_i \; if \; s = i$, $i = 0, 1, 2, 3$. For example, if $s = 2$, $y = x_2$.

1. Show a sum of products expression for y.
Solution The SOP expression is

$$y = x_0 s_1' s_0' + x_1 s_1' s_0 + x_2 s_1 s_0' + x_3 s_1 s_0$$

2. Implement MUX module using CMOS NAND gates (with fanin as needed) and NOT gates.
Solution
The network is



Optional problem. (10 extra points) Implement MUX module using CMOS transmission gates TG, NOR and NOT gates. A transmission gate TG_i is controlled by signal C_i:

C_i	TG_i
0	on
1	off

Complete the following table defining the values of control variables C_i and the output y:

s_1	s_0	C_0	C_1	C_2	C_3	y
0	0	1	0	0	0	x_0
0	1					
1	0					
1	1					

Show switching expressions for C_i's.

Show the final network. Label all inputs and outputs (external and internal). How many transistors are needed in total?

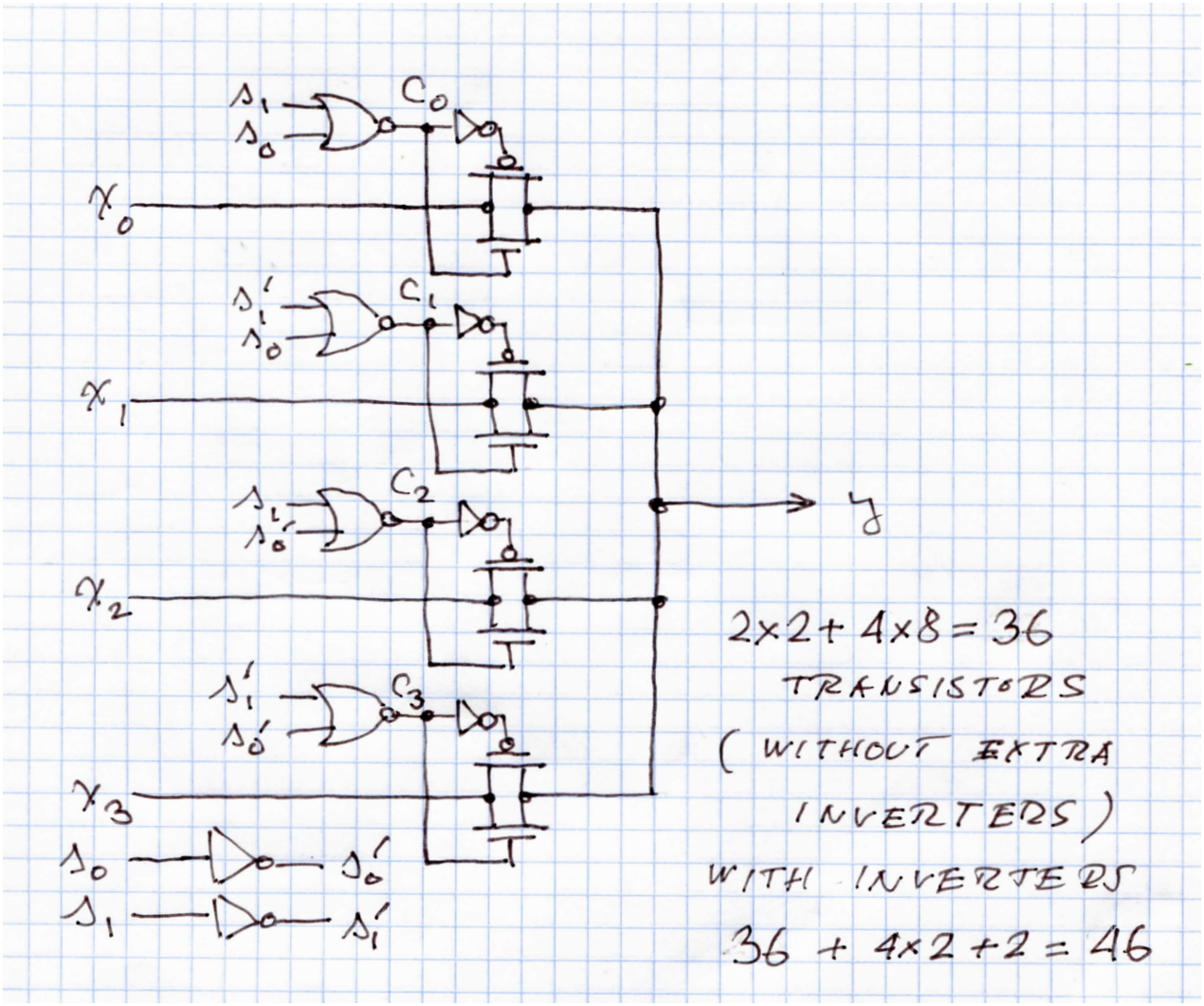
Solution

s_1	s_0	C_0	C_1	C_2	C_3	y
0	0	1	0	0	0	x_0
0	1	0	1	0	0	x_1
1	0	0	0	1	0	x_2
1	1	0	0	0	1	x_3

The expressions for C_i variables are

$$\begin{aligned} C_0 &= s_1' s_0' = (s_1 + s_0)' \\ C_1 &= s_1' s_0 = (s_1 + s_0')' \\ C_2 &= s_1 s_0' = (s_1' + s_0)' \\ C_3 &= s_1 s_0 = (s_1' + s_0')' \end{aligned}$$

The 4-input MUX design with transmission gates is:



Problem 3 (10 points)

1. **(5 points)** A 8-bit vector represents a set of positive integers {0,...,N}. Which of the following coding alternatives provides the largest range? Why? (Give N for each case).

   Solution

   (a) BCD: Max = 10011001 = 99 → The range is 100

   (b) 2421 code: Max = 11111111 = 99 → The range is 100

   (c) Excess-3 code: Max = 11001100 = 99 → The range is 100

   (d) Octal: Max = 11111111 = $377_8 = 255$ → The range is 256

   (e) Binary: Max = 11111111 = $11111111_2 = 2^8 - 1 = 255$ → The range is 256

   Thus, the answer is Octal and Binary.

2. **(5 points)**

   a and b are 12-bit vectors that represent their numbers in the BCD code. a = (100000110101) and the decimal of their sum $a + b$ is $1,800$. What is the bit vector of b? Show all your work.

   Solution

   $a = 100000110101 = 835$

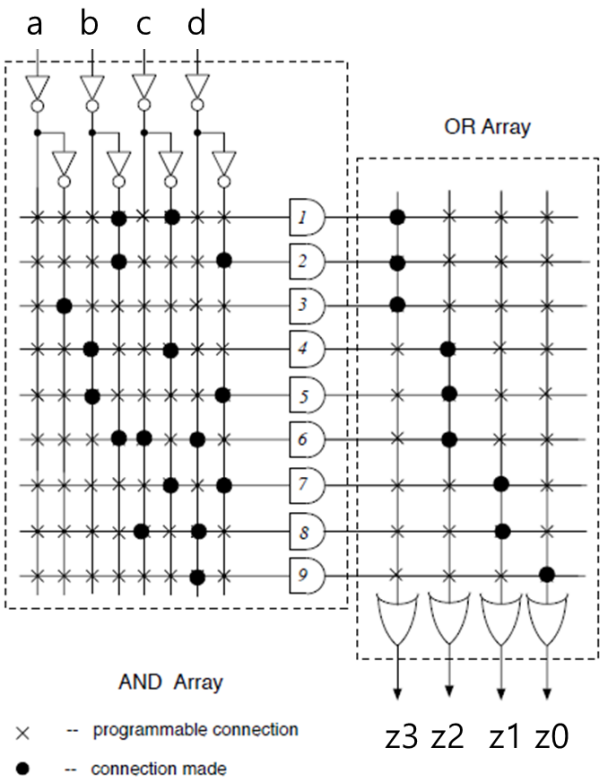
   $1,800 - 835 = 965$

   $965 = 1001\ 0110\ 0101$, so b = (100101100101)

Problem 4 (10 points)

We would like to verify that the PLA implementation shown here implements the following switching functions:

$$z3 = b + bd + ad$$
$$z2 = b'c + d + bc'd$$
$$z1 = cd + d'$$
$$z0 = d$$



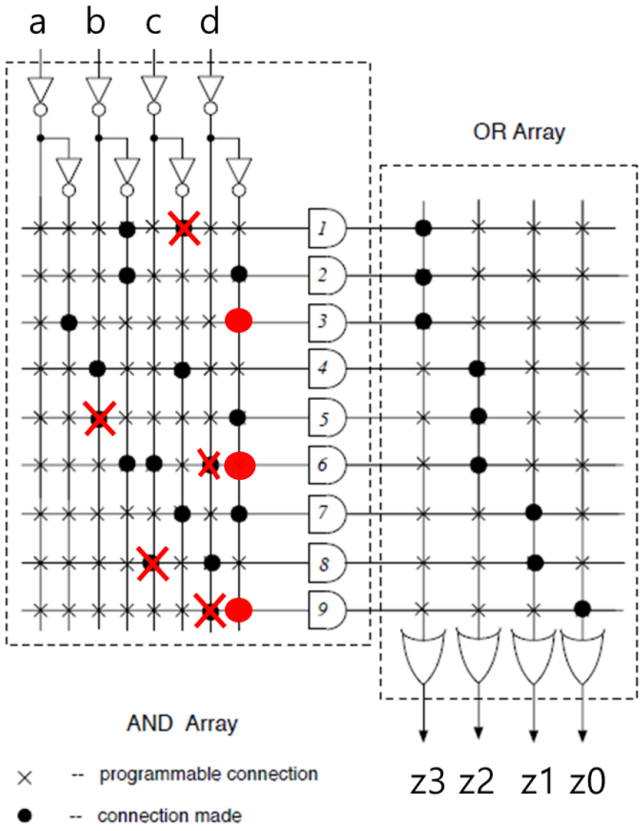
1. **(6 points)** Analyze the PLA shown above and show the output expressions.

   Solution

$$z3 = bc + bd + a$$
$$z2 = b'c + b'd + bc'd'$$
$$z1 = cd + c'd'$$
$$z0 = d'$$

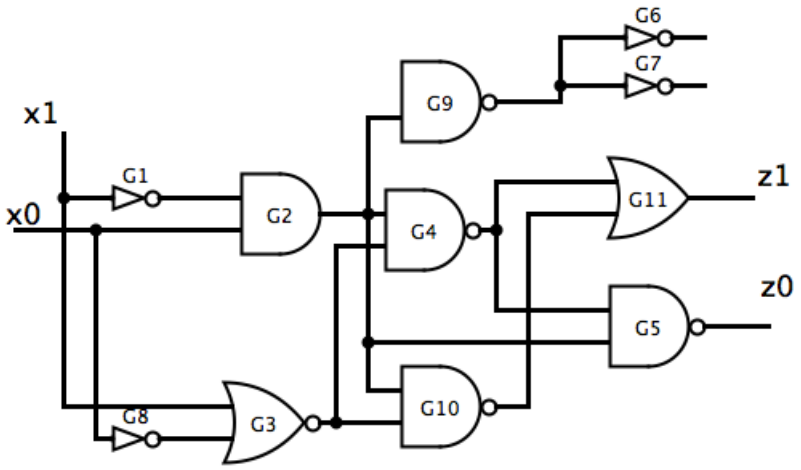
2. **(4 points)** Is the PLA implementation correct? If not, find errors and show the correct implementation (cross out wrong connections and insert correct ones)

Solution



Problem 5 (10 points)

Calculate the propagation delay $t_{pLH}(z1)$ when $x1$ changes. Assume that $z1$'s load value is 2. Fill in the blanks below with the appropriate values. You don't need to fill all the blanks.



Gate Type	Fan-in	Propagation Delays (ns) t_{pLH}	Propagation Delays (ns) t_{pHL}	Load Factor I
NOT	1	$0.02 + 0.038L$	$0.05 + 0.017L$	1.0
AND	2	$0.15 + 0.037L$	$0.16 + 0.017L$	1.0
OR	2	$0.12 + 0.037L$	$0.20 + 0.019L$	1.0
NAND	2	$0.05 + 0.038L$	$0.08 + 0.027L$	1.0
NOR	2	$0.06 + 0.075L$	$0.07 + 0.016L$	1.0

Solution

Gate name:	G1 $\rightarrow$ G2 $\rightarrow$ G4 $\rightarrow$ G11
Gate type:	NOT $\rightarrow$ AND2 $\rightarrow$ NAND2 $\rightarrow$ OR2
LH / HL:	HL $\rightarrow$ HL $\rightarrow$ LH $\rightarrow$ LH
Output load L:	1.0 $\rightarrow$ 4.0 $\rightarrow$ 2.0 $\rightarrow$ 2.0
Prop. Delay:	$0.05 + 0.017 \times 1.0 \rightarrow$ $0.16 + 0.017 \times 4.0 \rightarrow$ $0.05 + 0.038 \times 2.0 \rightarrow$ $0.12 + 0.037 \times 2.0$

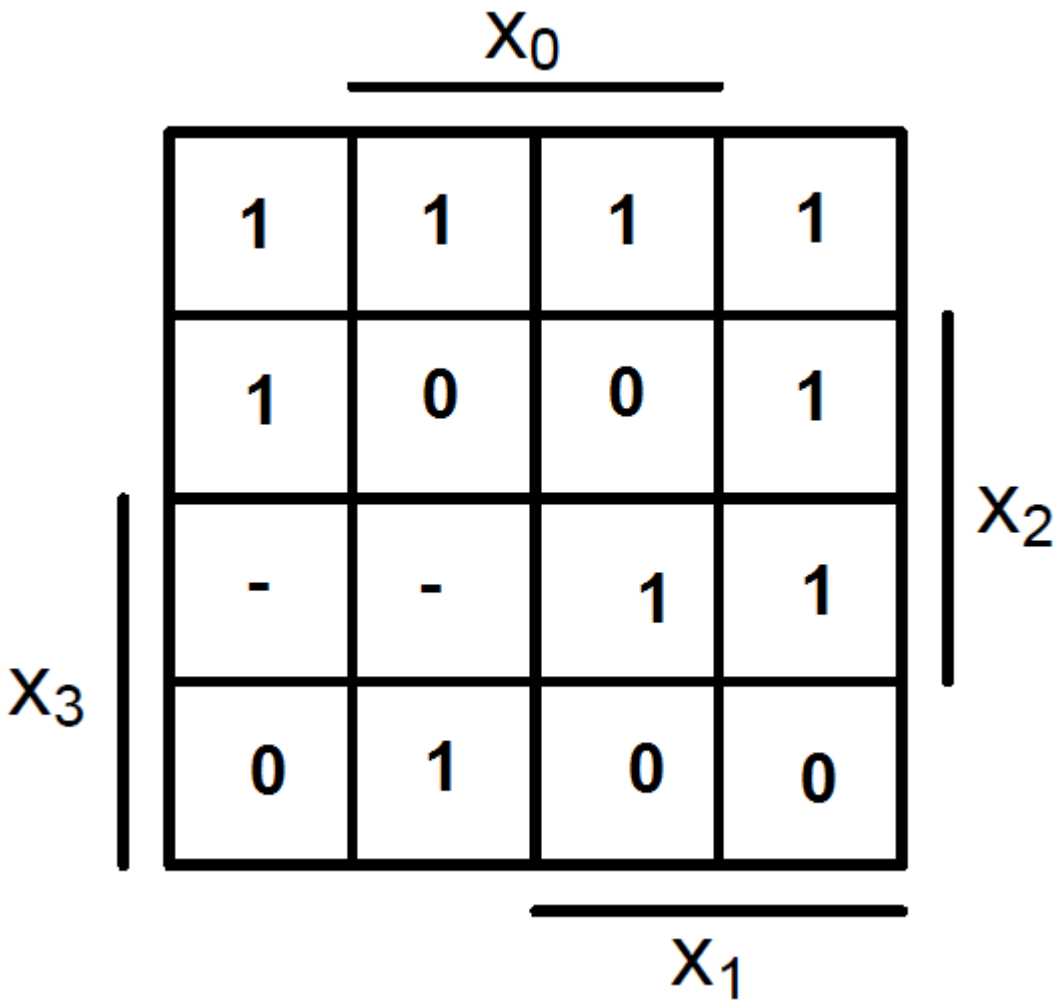
Problem 6 (25 points)

For the switching function $f(x_3, x_2, x_1, x_0)$, we are given the information below for the dc-set and zero-set.

$$\text{dc-set} = (12,13)$$
$$\text{zero-set} = \text{zero-set of function}$$
$$(x_3 + x_2' + x_1 + x_0')(x_3 + x_2' + x_1' + x_0')(x_3' + x_2 + x_1 + x_0)(x_3' + x_2 + x_1' + x_0)(x_3' + x_2 + x_1' + x_0')$$

1. **(2 points)** Fill out the following K-map.

   Solution The completed K-map is shown:



2. **(3 points)** Which of the given expressions are prime implicants of the function given above? Circle **all** that apply. Do not circle implicants that are not prime.

   Solution
   $x_3'x_0'$, $x_3'x_2'$, x_2x_0', x_3x_2, $x_3x_1'x_0$, $x_2'x_1'x_0$

3. **(3 points)** Write down the complete set of **essential** prime implicants.

   Solution
   x_3x_2.

4. **(3 points)** Write down the minimal sum of products expressions for f. If there are multiple forms of minimal sum of products expressions, you only need to write down one of them.

Solution
$x_3x_2 + x_3{}'x_2{}' + x_2x_0{}' + x_3x_1{}'x_0$
or
$x_3x_2 + x_3{}'x_2{}' + x_3{}'x_0{}' + x_3x_1{}'x_0$
or
$x_3x_2 + x_3{}'x_2{}' + x_2x_0{}' + x_2{}'x_1{}'x_0$
or
$x_3x_2 + x_3{}'x_2{}' + x_3{}'x_0{}' + x_2{}'x_1{}'x_0$

5. **(3 points)** Which of the given expressions are prime implicates of the function given above? Circle **all** that apply. Do not circle implicates that are not prime.

   Solution
   $(x_3{}' + x_2 + x_1{}')$, $(x_3{}' + x_2 + x_0)$, $(x_3 + x_2{}' + x_0{}')$, $(x_2{}' + x_1 + x_0{}')$, $(x_3{}' + x_1 + x_0)$,

6. **(3 points)** Write down the complete set of **essential** prime implicates.

   Solution
   $(x_3{}' + x_2 + x_1{}')$ and $(x_3 + x_2{}' + x_0{}')$

7. **(3 points)** Write down the minimal product of sums expressions for f. If there are multiple forms of minimal product of sums expressions, you only need to write down one of them.

   Solution
   $(x_3{}' + x_2 + x_1{}') + (x_3 + x_2{}' + x_0{}') + (x_3{}' + x_1 + x_0)$
   or
   $(x_3{}' + x_2 + x_1{}') + (x_3 + x_2{}' + x_0{}') + (x_3{}' + x_2 + x_0)$