1. Prove that in 2's complement number system addition overflows if and only if the carry from the sign position does not equal the carry into the sign position. Consider the three cases: adding two positive numbers, adding two negative numbers, and adding two numbers of opposite sign. [10 *pts*]

Solution (Suhyeong)

C_{n+1}: carry from sign position (MSB)
C_n: carry into sign position (MSB)

i) Positive + Positive: because both MSBs are 0, C_{n+1} is always 0. If C_n is 1, MSB becomes 1, so that negative number is resulted in, which is overflow. [+3 *pts*]

ii) Negative + Negative: Because both MSBs are 1, C_{n+1} is always 1. If C_n is 0, MSB becomes 0, so that positive number is resulted in, which is overflow. [+3 *pts*]

iii) Positive + Negative: Because either of MSBs is 1, C_{n+1} is 1 only if C_n is 1, which never leads overflow. (If C_n is 0, C_{n+1} is also 0, which is not overflow as well.) [+4 pts]

2. Algebraically prove or disprove the following identities: [10 *pts*]
   a. $x(y \oplus z) = xy \oplus xz$
   b. $x + (y \oplus z) = (x + y) \oplus (x + z)$
   c. $x(y \equiv z) = xy \equiv xz$
   d. $x + (y \equiv z) = (x + y) \equiv (x + z)$

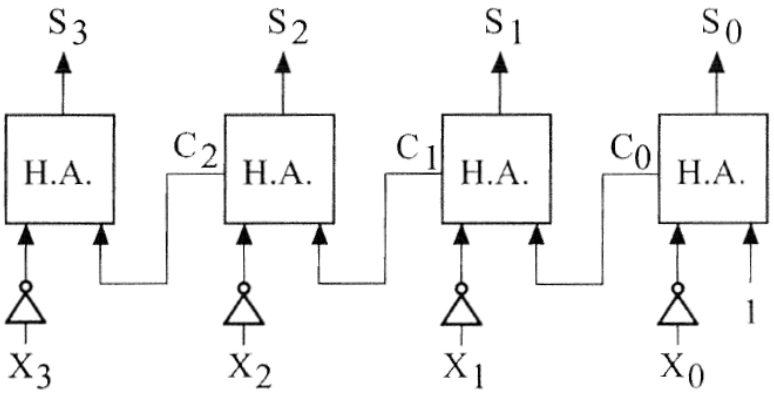
Solution (Cheongwon)

   a. $xy \oplus xz = xy(x' + z') + (x' + y')xz = xyz' + xy'z = x(yz' + yz') = x(y \oplus z)$ [+2.5 *pts*]
   b. For $x = 1$, LHS is 1 but RHS is 0 which are not equal. [+2.5 *pts*]
   c. For $x = 0$, LHS is 0 but RHS is 1 which are not equal. [+2.5 *pts*]
   d. $(x + y) \equiv (x + z) = (x + y)(x + z) + (x + y)'(x + z)' = x + yz + (x'y')(x'z')$
      $= x + yz + x'y'z' = x + yz + y'z' = x + (y \equiv z)$ [+2.5 *pts*]

3. Design a circuit which will find the 2's complement of a 4-bit binary number. Use four half adders and any additional gates. [5 *pts*]

Solution (Suhyeong)

If $S_3S_2S_1S_0$ is 2's complement of $X_3X_2X_1X_0$, $S_3S_2S_1S_0 = \text{Flip}(X_3X_2X_1X_0) + 1$



Scoring criteria

Ambiguity in inputs and outputs of half adders and gates [-2 *pts*]
Use of non-basic logic gates (e.g. MUX) [-2 *pts*]

4. The following prime implicant chart is for a four-variable function $f(A, B, C, D)$.
   a. List the maxterms of f. [5 *pts*]
   b. List the don't-cares of f, if any. [5 *pts*]

	2	3	7	9	11	13
–0–1		×		×	×	
–01–	×	×			×	
--11		×	×		×	
1--1				×	×	×

Solution (Jingon)

$$-0-1 = \{m_1, m_3, m_9, m_{11}\}$$
$$-01- = \{m_2, m_3, m_{10}, m_{11}\}$$
$$--11 = \{m_3, m_7, m_{11}, m_{15}\}$$
$$1--1 = \{m_9, m_{11}, m_{13}, m_{15}\}$$

The first row of prime implicant chart only contains minterms (not don't cares), so m_1, m_{10}, m_{15} are don't cares.

The K-map of f as follows:

cd \ ab	00	01	11	10
00	0	0	0	0
01	x	0	1	1
11	1	1	x	1
10	1	0	0	x

a. Maxterms of f: $m_0, m_4, m_5, m_6, m_8, m_{12}, m_{14}$ [+5 *pts*]

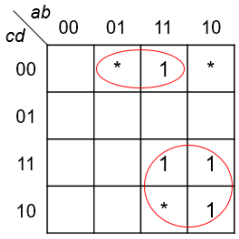
b. Don't cares: m_1, m_{10}, m_{15} [+5 *pts*]

5. Find a minimum two-level, multiple-output AND-OR gate circuit to realize these functions (8 gates minimum). [10 *pts*]

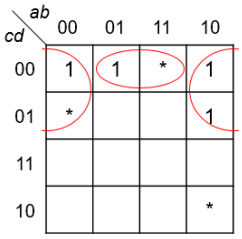
$f_1(a,b,c,d) = \sum m(10,11,12,15) + \sum d(4,8,14)$
$f_2(a,b,c,d) = \sum m(0,4,8,9) + \sum d(1,10,12)$
$f_3(a,b,c,d) = \sum m(4,11,13,14,15) + \sum d(5,\ 9,12)$

Solution (Suhyeong)

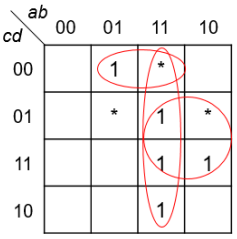
$f_1 = \underline{bc'd'} + ac$
$f_2 = \underline{bc'd'} + b'c'$
$f_3 = \underline{bc'd'} + ab + ad$



f_1



f_2



f_3

By sharing $\underline{bc'd'}$, 8 gates (5 AND and 3 OR gates) with 18 gate inputs [+10 *pts*]

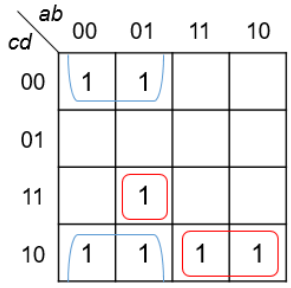
Scoring criteria

Not minimum circuit that includes 8 gates with more than 18 gate inputs [+7 *pts*]
No point for the circuit with more than 8 gates

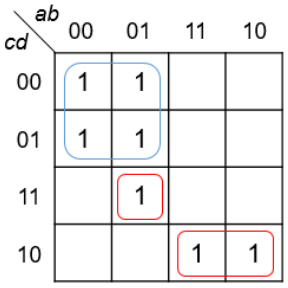
6. (a) Find a minimum two-level, multiple-output NAND-NAND circuit to realize $f_1 = \sum m(0,2,4,6,7,10,14)$ and $f_2 = \sum m(0,1,4,5,7,10,14)$. (b) Repeat for minimum two-level, multiple-output NOR-NOR circuit. [10 *pts*]

Solution (Cheongwon)

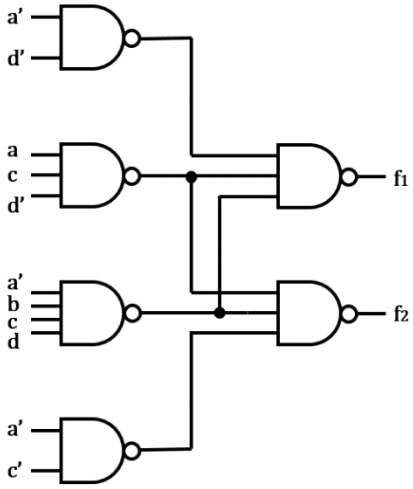
(a) [+5 *pts*]



f_1



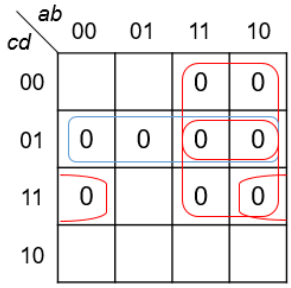
f_2



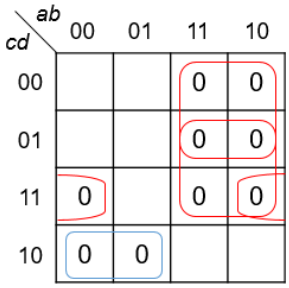
$f_1 = a'd' + a'bcd + acd'$

$f_2 = a'c' + a'bcd + acd'$

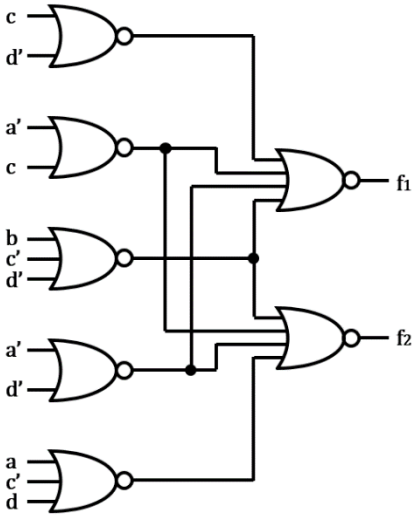
(b) [+5 *pts*]



f_1



f_2



$f_1 = (a' + c)(a' + d')(c + d')(b + c' + d')$

$f_2 = (a' + c)(a' + d')(b + c' + d')(a + c' + d)$

Scoring criteria

Only K-map is correct. [+1 *pt*]

Correct function and circuit diagram but not minimum [+2 *pts*]

Correct function and minimum circuit diagram [+5 *pts*]

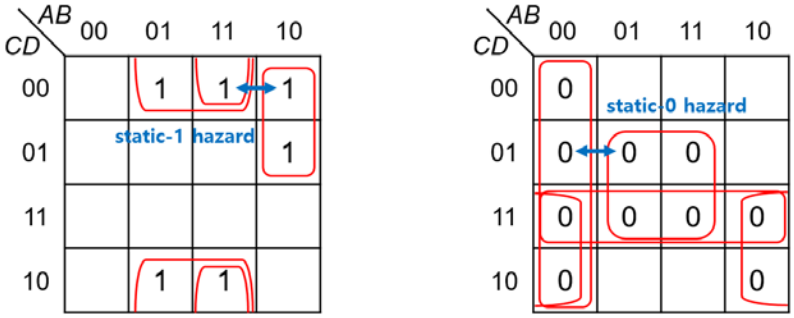
7. Consider the three-level circuit corresponding to the expression f(A, B, C, D) = (A+B)(B'C'+BD'). [15 *pts*]
   a. Find all hazards in this circuit. [10 *pts*]
   b. Redesign the circuit as a three-level NOR circuit that is free of all hazards. [5 *pts*]

Solution (Cheongwon)

a.

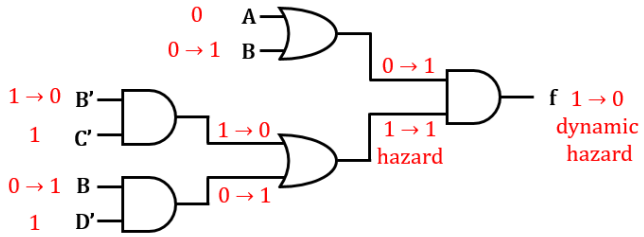
$$f = (A+B)(B'C' + BD')$$
$$= AB'C' + ABD' + BB'C' + BD'$$
$$= (A+B)(B'+B)(B'+D')(B+C')(C'+D')$$

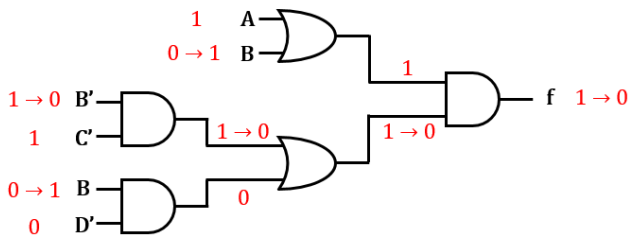
From the K-Map and the BB'C' term



Static-1 hazard: 1100↔1000
Static-0 hazard: 0001↔0101

Potential dynamic hazards (due to BB'C' term): 0000↔0100 and 1101↔1001



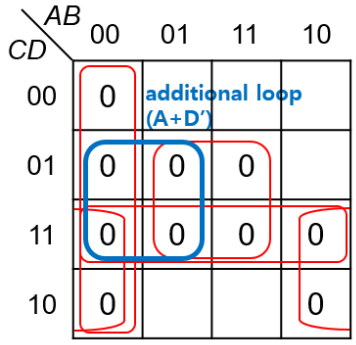


The circuit show that only 0000↔0100 propagates over three paths

Scoring criteria

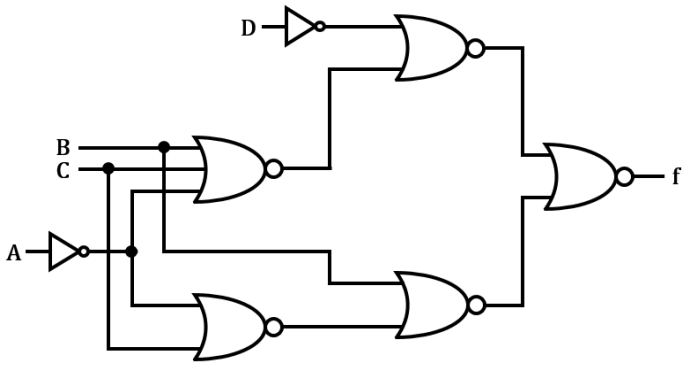
Only K-Map and loop are correct. [+2 *pts*]
For static-1 hazard, additional [+2 *pts*]
For static-0 hazard, additional [+2 *pts*]
For dynamic hazard, additional [+4 *pts*]

b.



$f = (A + B)(B' + D')(B + C')(C' + D')(\boldsymbol{A} + \boldsymbol{D}')$

To implement three-level circuit, f can be multiplied out as
$f = (A + B)(B' + D')(B + C')(C' + D')(A + D') = (A'C + B)(AB'C' + D')$

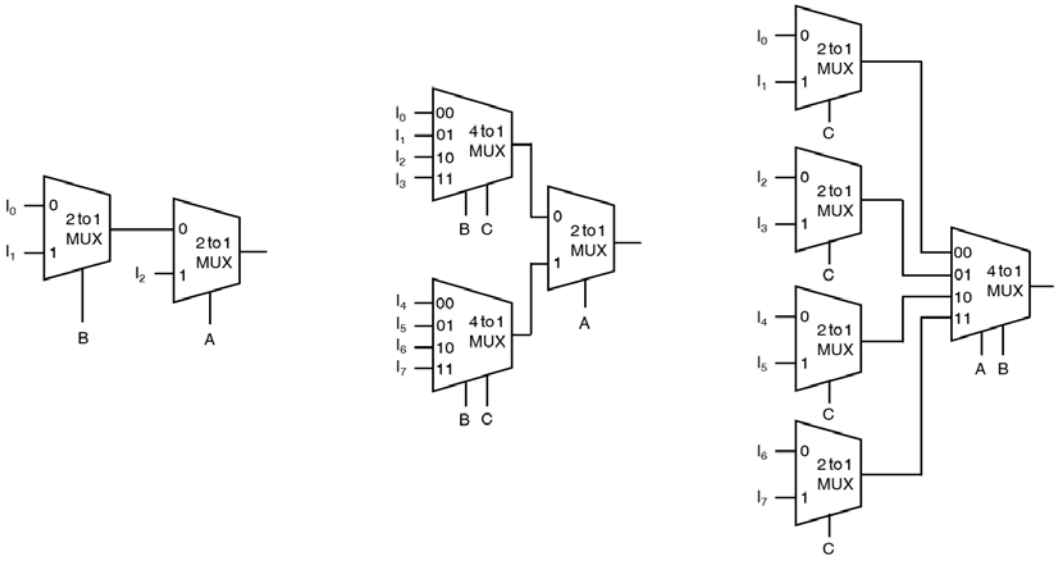


Scoring criteria

Only additional loop and function are correct. [+3 *pts*]
For correct circuit diagram, additional [+2 *pts*]
No point for non-three-level NOR circuit.

8. (a) Show how two 2-to-1 multiplexers (with no added gates) could be connected to form a 3-to-1 MUX. Input selection should be as follows: If AB = 00, select I0. If AB = 01, select I_1. If AB = 1-, select I_2. [3 *pts*]
(b) Show how two 4-to-1 and one 2-to-1 multiplexers could be connected to form an 8-to-1 MUX with three control inputs. [3 *pts*]
(c) Show how four 2-to-1 and one 4-to-1 multiplexers could be connected to form an 8-to-1 MUX with three control inputs. [4 *pts*]

Solution (Jingon)



(a) [3 *pts*] (b) [3 *pts*] (c) [4 *pts*]

Scoring Criteria

Control bits (e.g. 00, 01, 10, and 11 in 4-to-1 MUX)are not specified. [-1 *pt*] (for each problem)

9. Answer the followings.

a. When we derive minimum POS of *f*, we derive minimum SOP of *f'* and take its complement. Explain why this is true (i.e. complement of minimum SOP of *f'* is minimum POS of *f*).

b. In 4-bit 2's complement number system, we use 2^4 to define a negative number (i.e. negative number of 3=0011 is 13=1101 because 3+13=16); in 1's complement, we use 2^4-1 to define a
negative number. Can we use 2^4+1 and 2^4-2 to define a negative number? Explain what happens.

c. For 3-variable K-map, what is the maximum number of essential prime implicants you can have? How about 4-variable K-map?

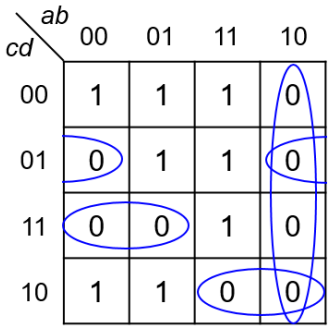
d. AOI gate performs AND followed by OR followed by INV, i.e. it functions $ab + c$. Design the gate using nMOS and pMOS transistors. Use minimum number of transistors.

Solution (Jingon)

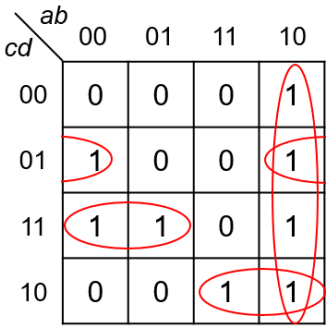
a.

Let's consider the K-map of f'. The K-map of f can be optained by exchanging 0 and 1 from the K-map of f'.

For example,



f



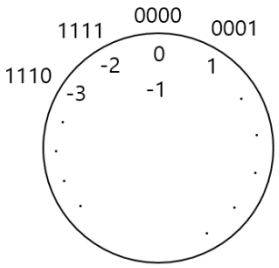
f'

Since SOP of f' and POS of f have the same prime implicants, minimum SOP of f' is minimum POS of f.

b.

We cannot use 2^4+1 system.

If we use 2^4+1 system, N+N' = 2^4+1, we cannot express -1 in 4 bits. [2 *pts*]
(e.g. we need 5 bits for expressing -1)



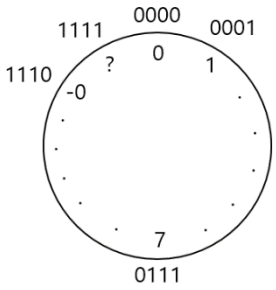
We cannot use 2^4-2 system, N+N' = 2^4-2,

If we use 2^4-2 system, N+N' = 2^4+1, we cannot express $1111_{(2)}$. [3 *pts*]

For example,

$1111_{(2)}$ + $0001_{(2)}$ = (1)$0000_{(2)}$, then $1111_{(2)}$ should be -1

$1111_{(2)}$ + $1110_{(2)}$ = (1)$0001_{(2)}$, then $1111_{(2)}$ should be 1



c.

For 3-variable K-map: 4 [2 *pts*]

1	
	1
1	
	1

or

	1
1	
	1
1	

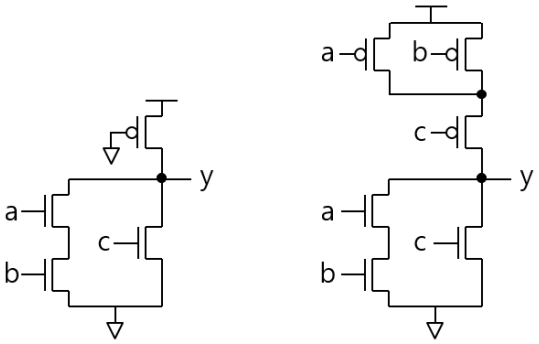
For 4-variable K-map: 8 [3 *pts*]

1		1	
	1		1
1		1	
	1		1

or

	1		1
1		1	
	1		1
1		1	

d.



Scoring Criteria

Question 9-d)

Correct functionality [+1 *pts*].
If # transistors ≤ 6, additional [+4 *pts*].