[CS M51A Winter 17] Midterm exam

Date: 2/16/17

- The midterm is closed books and notes. Tablets and smartphone are not allowed.
- You can use calculators and have up to 2 sheets (= 4 pages) of summary notes.
- Please show all your work and write legibly, otherwise no partial credit will be given.
- **This should strictly be your own work; any form of collaboration will be penalized.**

Name : ____________________

Student ID : ____________________

Problem	Points	Score
1	20	17
2	15	18
3	10	8
4	10	10
5	20	20
6	25	23
Total	100	96

Problem 1 (20 points)

$(a+b)' = a'b'$ $(ab)' = a'+b'$ $x \oplus y = x'y + xy'$

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.

$= (ad' \oplus b')(c+d) + (a'+bc)'cd'$

$= ((ad')'b' + (ad')(b')')(c+d) + (a'+bc)'cd'$ [$x \oplus y = x'y + xy'$ XOR]

$= ((a'+(d')')b' + (ad')b)(c+d) + ((a')'(bc)')cd'$ [DeMorgan's Law and Involution (i.e. $(b')'$)]

$= ((a'+d)b' + (ad')b)(c+d) + (a(b'+c'))cd'$ [DeMorgan's Law and Involution again]

$= (a'b' + b'd + abd')(c+d) + (ab' + ac')cd'$ [Distributivity and Associativity]

$= (a'b'c + a'b'd + b'cd + b'dd + abcd' + abdd') + (ab'cd' + acc'd')$ [Distributivity]

$= a'b'c + a'b'd + b'cd + b'd + abcd' + ab'cd'$ [Complement]

$= a'b'c + b'd(a'+1) + b'cd + abcd' + ab'cd'$ [Absorption]

$= a'b'c + b'd + b'cd + acd'(b+b')$ [Absorption]

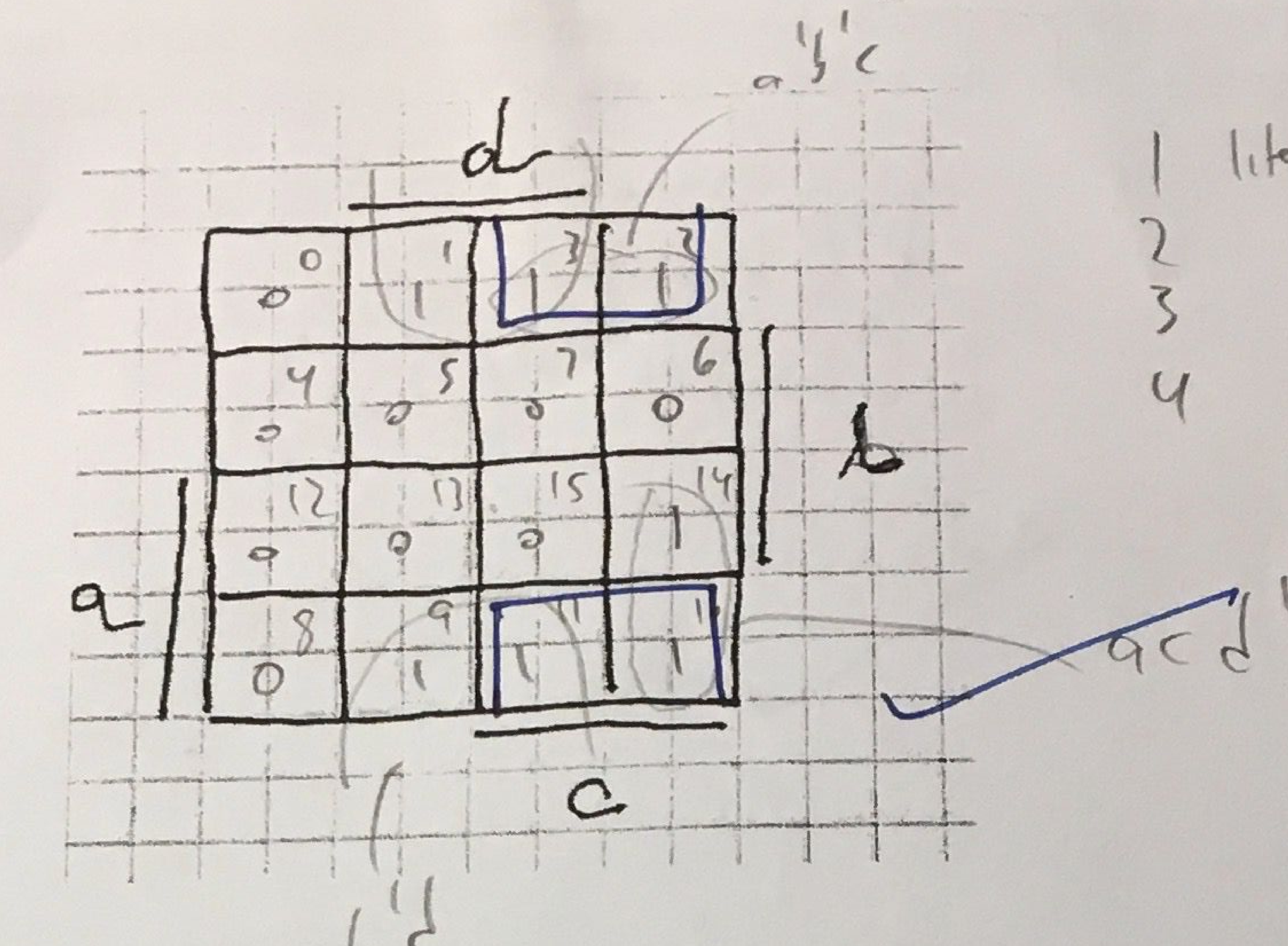
$= a'b'c + b'd(1+c) + acd'$ [Absorption]

$= a'b'c + b'd + acd'$

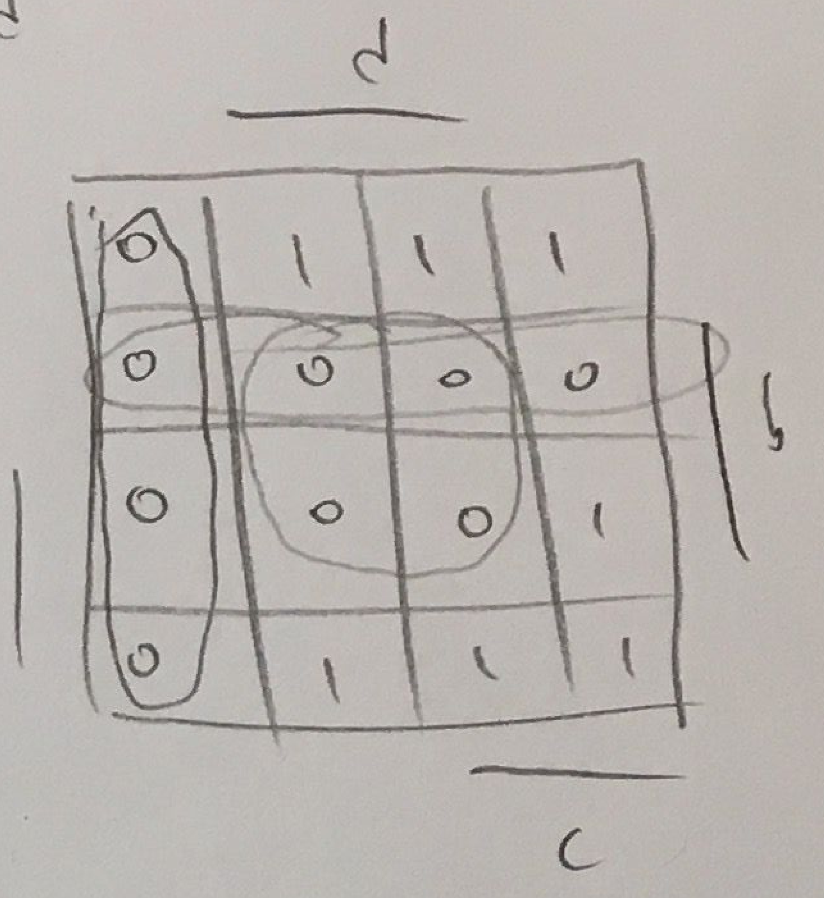
7

1.	$a + b = b + a$	$ab = ba$	Commutativity
2.	$a + (bc) = (a+b)(a+c)$	$a(b+c) = (ab) + (ac)$	Distributivity
3.	$a + (b+c) = (a+b) + c = a + b + c$	$a(bc) = (ab)c = abc$	Associativity
4.	$a + a = a$	$aa = a$	Idempotency
5.	$a + a' = 1$	$aa' = 0$	Complement
6.	$1 + a = 1$	$0a = 0$	
7.	$0 + a = a$	$1a = a$	Identity
8.	$(a')' = a$		Involution
9.	$a + ab = a$	$a(a+b) = a$	Absorption
10.	$a + a'b = a + b$	$a(a'+b) = ab$	Simplification
11.	$(a+b)' = a'b'$	$(ab)' = a' + b'$	DeMorgan's Law

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).



literal	cells
1	8
2	4
3	2
4	1



(3)

Essential Prime Implicants ⇒ $a'b'c$, $b'd$, acd'

Min SOP ⇒ $a'b'c + b'd + acd'$

Essential Prime Implicates ⇒ $(c+d)$, $(b'+d')$, $(a+b')$

Min PS ⇒ $(c+d)(b'+d')(a+b')$

• The min SOP is the same as the SOP I obtained in (1)

Scratch work

$(x+y)' = x'y'$

$(xy)' = x'+y'$

$x \oplus y = x'y + xy'$

$= (ad' \oplus b')(c+d) + (a'+bc)'cd'$

$= [(ad')'b' + (ad')b](c+d) + [a(bc)']cd'$

$= [(a'+d)b' + abd'](c+d) + [a(b'+c')]cd'$

$= (a'b' + b'd + abd')(c+d) + (ab' + ac')cd'$

$= a'b'c + a'b'd + b'cd + b'dd + abcd' + abdd' + ab'cd' + acc'd'$

$= a'b'c + a'b'd + b'cd + b'd + abcd' + ab'cd'$

$= a'b'c + b'cd + b'd(a'+1) + acd'(b+b')$

$= a'b'c + b'cd + b'd + acd'$

$= a'b'c + acd' + b'd(c+1)$

$= a'b'c + acd' + b'd$

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.)

$(x+y)' = x'y'$

$(xy)' = x'+y'$

Min SP $= a'b'c + b'd' + acd'$

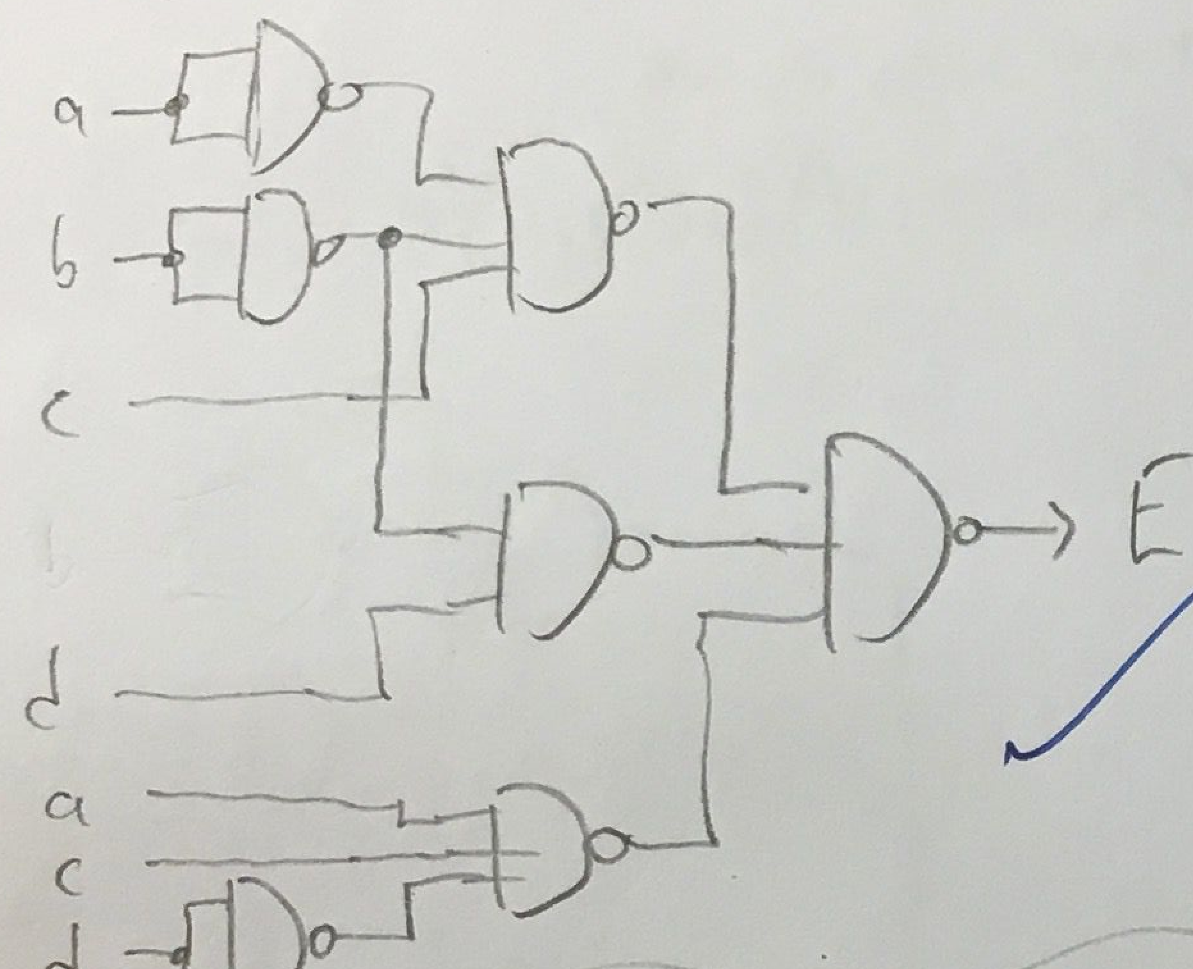
NAND-Implementation $= [(a'b'c + b'd' + acd')']'$

$[(a'b'c)'(b'd')'(acd')']'$

• Note ⇒ x → NAND → x' because $(xx)' = x' + x' = x'$

• I will use this to invert a variable

Min SP



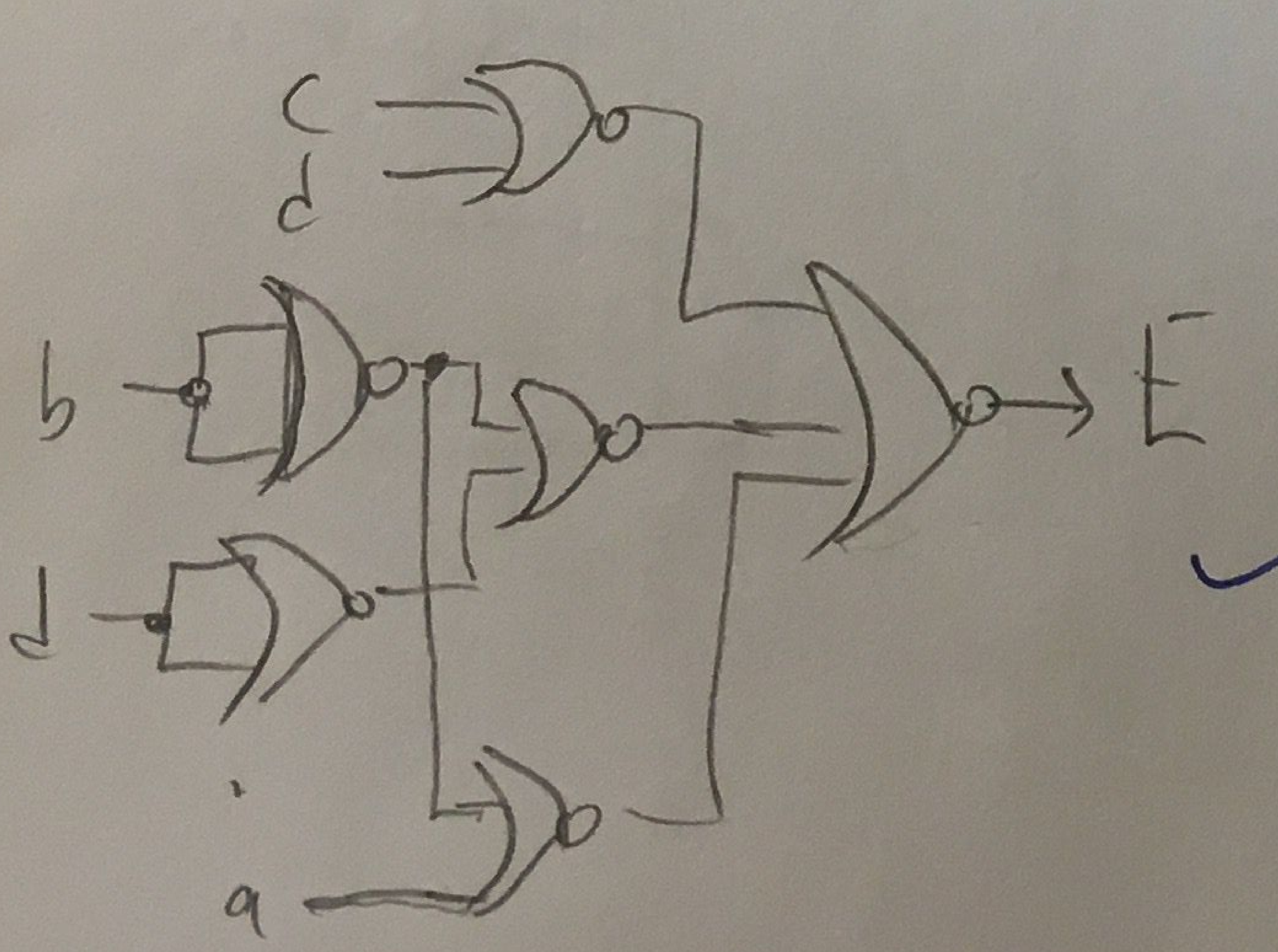
7

Min PS $= (c+d)(b'+d')(a+b')$

$= [[(c+d)(b'+d')(a+b')]']'$

NOR-Implementation $= [(c+d)' + (b'+d')' + (a+b')']'$

• Note ⇒ x → NOR → x' because $(x+x)' = x'x' = x'$



• From my simplified NAND and NOR implementation, there are 7 gates for NAND, and 6 gates in NOR. There are 12 total inputs for NAND, and 11 for NOR. NOR implementation is the more minimal network.

Problem 2 (15 points) 13 + 5

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. The output is connected to one of the data inputs determined by the select inputs. Formally, the MUX function is specified as

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

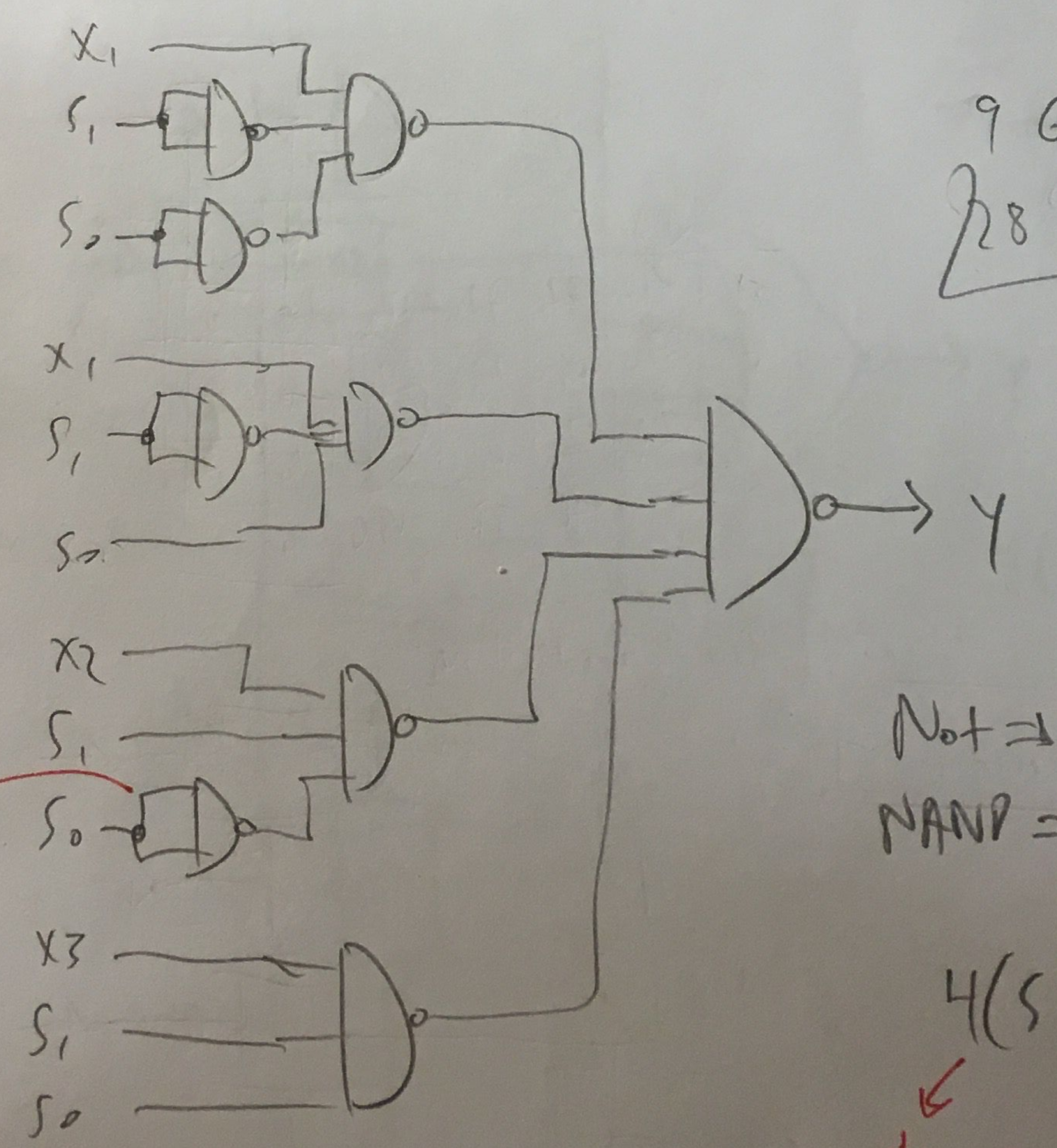
$(x+y)' = x'y'$

1. Show a sum of products expression for y.
2. Implement MUX module using CMOS NAND gates (with fanin as needed) and NOT gates. How many transistors are used?

$y = MUX(x_3, x_2, x_1, x_0, s_1, s_0) = x_0 s_1' s_0' + x_1 s_1' s_0 + x_2 s_1 s_0' + x_3 s_1 s_0$ (+5)

$[(x_0 s_1' s_0' + x_1 s_1' s_0 + x_2 s_1 s_0' + x_3 s_1 s_0)']'$

$[(x_0 s_1' s_0')'(x_1 s_1' s_0)'(x_2 s_1 s_0)'(x_3 s_1 s_0)']'$



9 Gates
28 Transistors

Could have used NOT GATES

Not => 2 transistor
NAND => 4

$4(5) + 2(4) = 28$

depends on fan-in (+2)

+6

Optional problem. (10 extra points) (+5) 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:

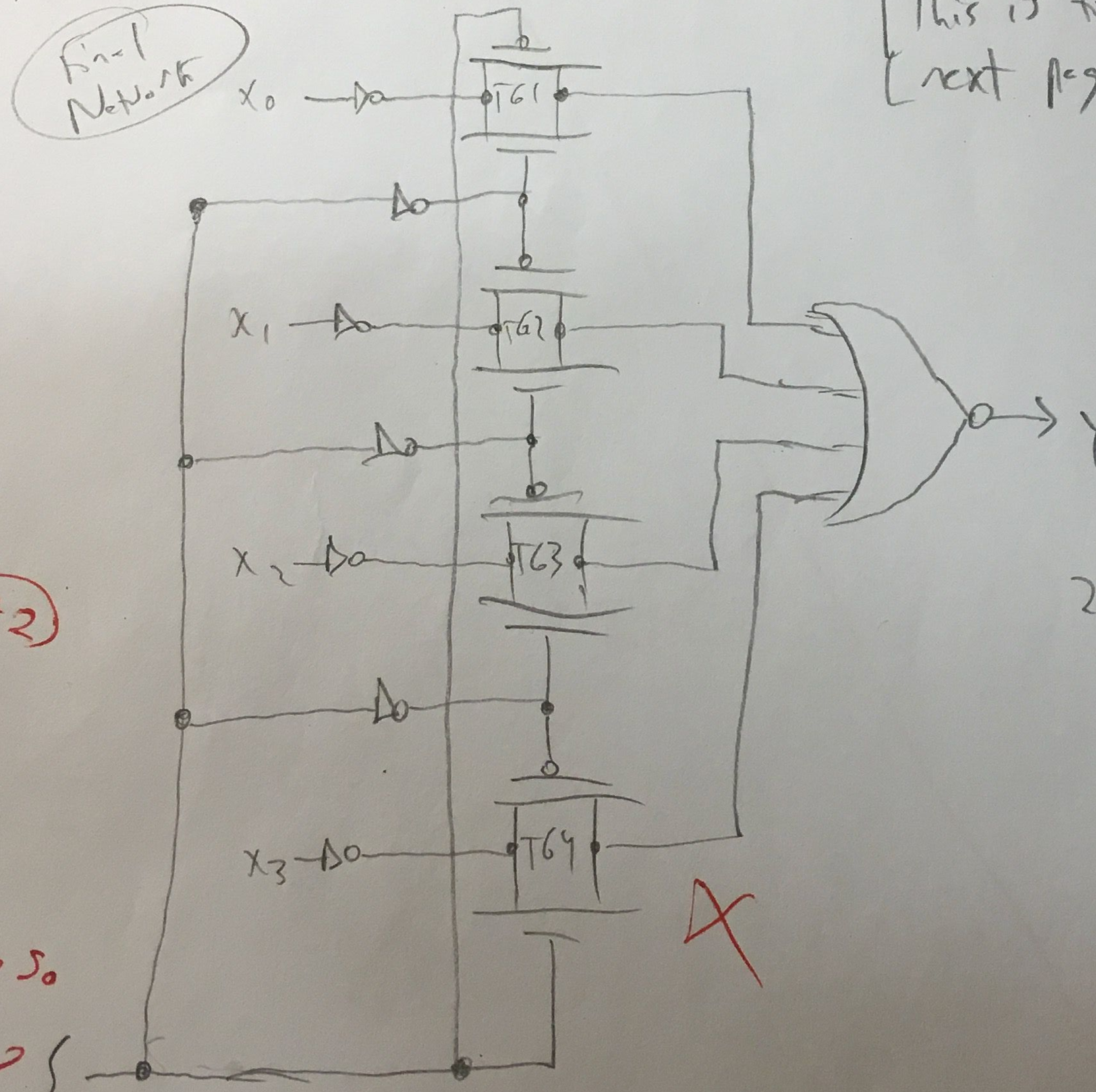
s_1	s_0	C_0	C_1	C_2	C_3	y
0	0	1	0	0	0	x_0
0	1	0	1	0	0	x_1
1	0	0	0	1	0	x_2
1	1	0	0	0	1	x_3

+1 ✓

Show switching expressions for C_i's.

[This is for part II on the next page sorry!]

Final Network



Transistors : 26

Not → 2

Transmission → 2

NOR → 4

$2(7)+2(4)+4 = 26$

(+1) (+2)

Switching Expression

$y = C_0 s_1' s_0' + C_1 s_1' s_0 + C_2 s_1 s_0' + C_3 s_1 s_0$

(+1) ✓

$C_0 = s_1' s_0'$

$C_1 = s_1' s_0$ → not gates

$C_2 = s_1 s_0'$

$C_3 = s_1 s_0$

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

On previous page, sorry!

26 transistors

Problem 3 (10 points)

8

1. (5 points) A 8-bit vector represents a set of positive integers {0,...,N}. Which of the following coding alternatives

unsigned for all

(a) BCD ⇒ 1001|1001 ⇒ 99 (0, ..., 99)

(b) 2421 code (a decimal code) ⇒ 1111|1111 ⇒ 99 (0, ..., 99)

(c) Excess-3 code (a decimal code) ⇒ 1100|1100 ⇒ 99 (0, ..., 99)

(d) Octal ⇒ 11|111|111 ⇒ 377, $(3\cdot8^2)+(7\cdot8^1)+(7\cdot8^0) = 255$

(e) Binary ⇒ 1111|1111 ⇒ FF ⇒ $(15\cdot16^1)+(15\cdot16^0) = 255$, 2^n-1 ✓

provides the largest range? Why? (Give N for each case).

Both octal and binary provide the largest range due to their bit representations. While the other have a max value of 99, octal and binary are able to extend to 255 (2^n-1) where n is the # of bits available due to them being a multiple of radix-2

2. (5 points)
a and b are 12-bit vectors that represent their numbers in the BCD code. a = (1000|0011|0101) and the decimal of their sum $a + b$ is 1,800. What is the bit vector of b? Show all your work.

BCD ⇒ a = 835

a+b = 1800

835+b = 1800

$b = (965)_{10}$ ✓

2|965 → 482 (R1), 2|482 → 241 (R0), 2|241 → 120 (R1), 2|120 → 60 (R0), 2|60 → 30 (R0), 2|30 → 15 (R0), 2|15 → 7 (R1), 2|7 → 3 (R1), 2|3 → 1 (R1), 2|1 → 0 (R1)

0011|1100|0101

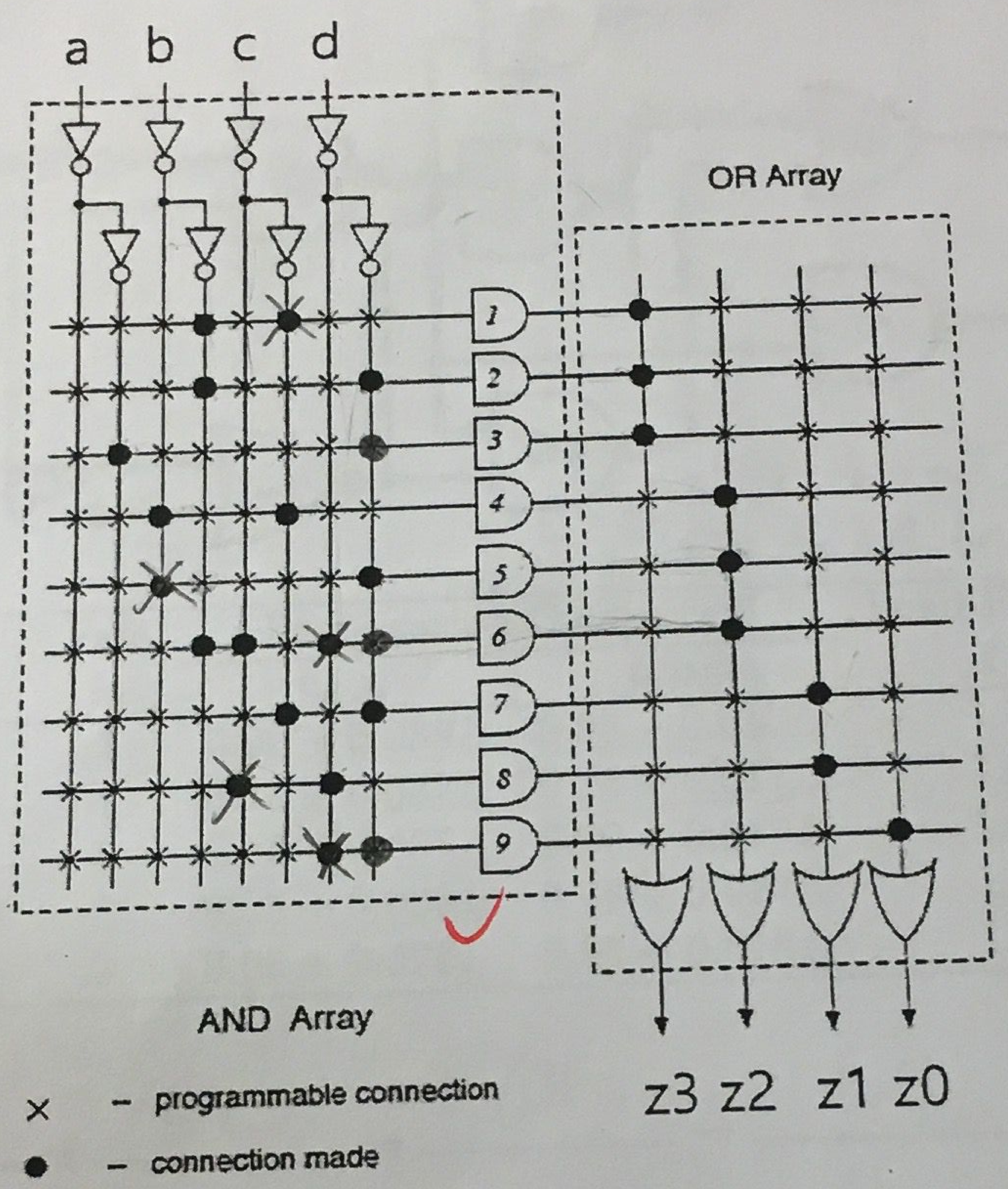
b = (0011 1100 0101) ✗ −2

Problem 4 (10 points)

10

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.

$z_3 = a + bd + bc$
$z_2 = bc'd' + b'd + b'c$
$z_1 = cd' + cd$
$z_0 = d'$

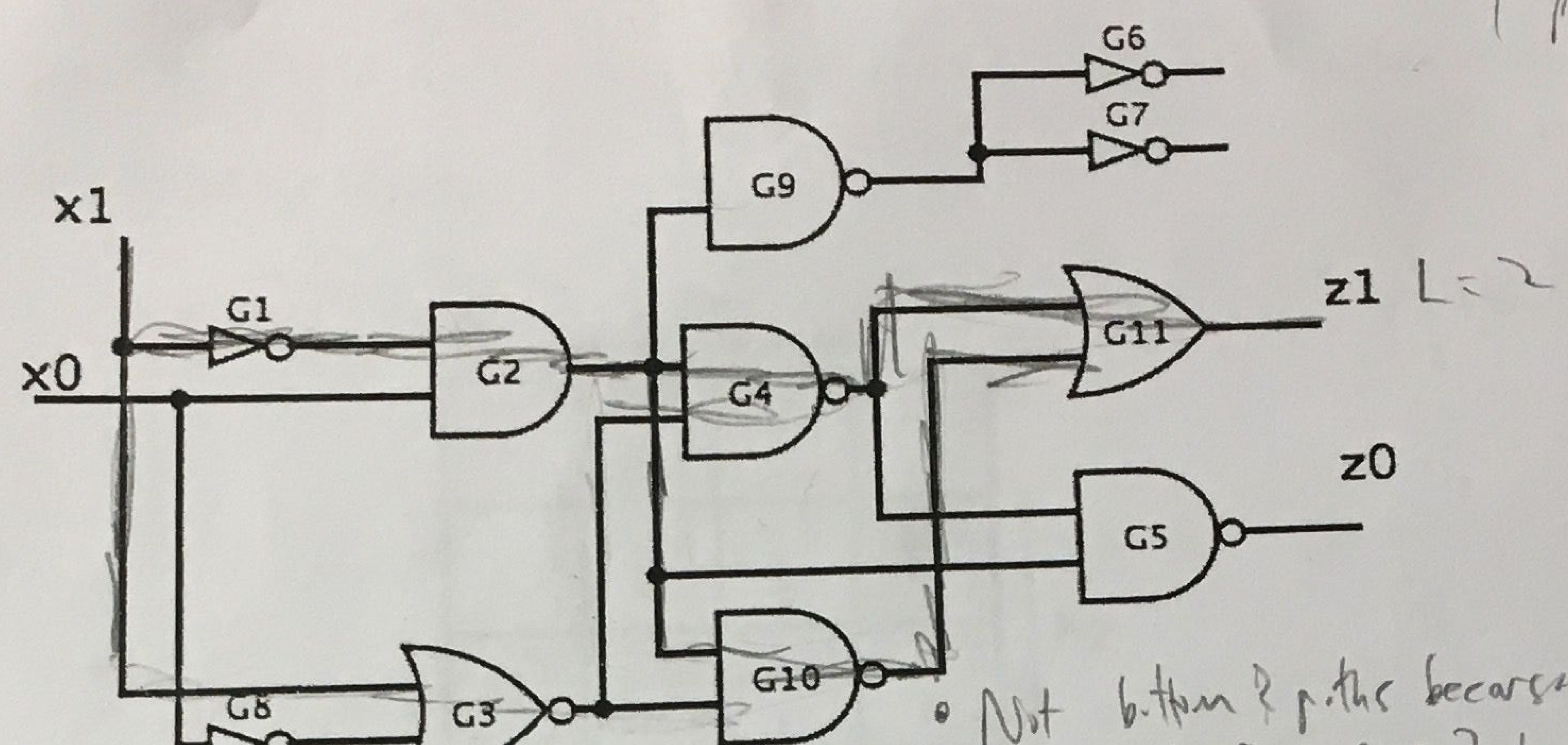
2. **(4 points)** Is the PLA implementation correct? If not, find errors and show the correct implementation (cross out wrong connections and insert correct ones)

No, shown above | connections needed

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.

4 possible paths



• Not bottom 2 paths because it has one less gate output
• Is top path for upper 2 because it has a higher load factor

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

Gate name: G1 → G2 → G4 → G11 → ____ → ____

Gate type: NOT1 → AND2 → NAND2 → OR2 → ____ → ____

LH / HL: HL → HL → LH → LH → ____ → ____

Output load L: 1.0 → 4.0 → 2.0 → 2.0 → ____ → ____

Prop. Delay: 0.05+0.017(1) → 0.16+0.017(4) → 0.05+0.038(2) → 0.12+0.037(2) → ____ → ____

G1 → G2 → G4 → G11 or G1 → G2 → G10 → G11

✓

0.067 + 0.228 + 0.126 + 0.194 = $t_{pLH}(z_1)$ from x_1 = 0.615 (ns)

• The path above was chosen due to it having the greatest # of gates in correspondence with a higher overall load L than the other 2 paths.

Problem 6 (25 points)

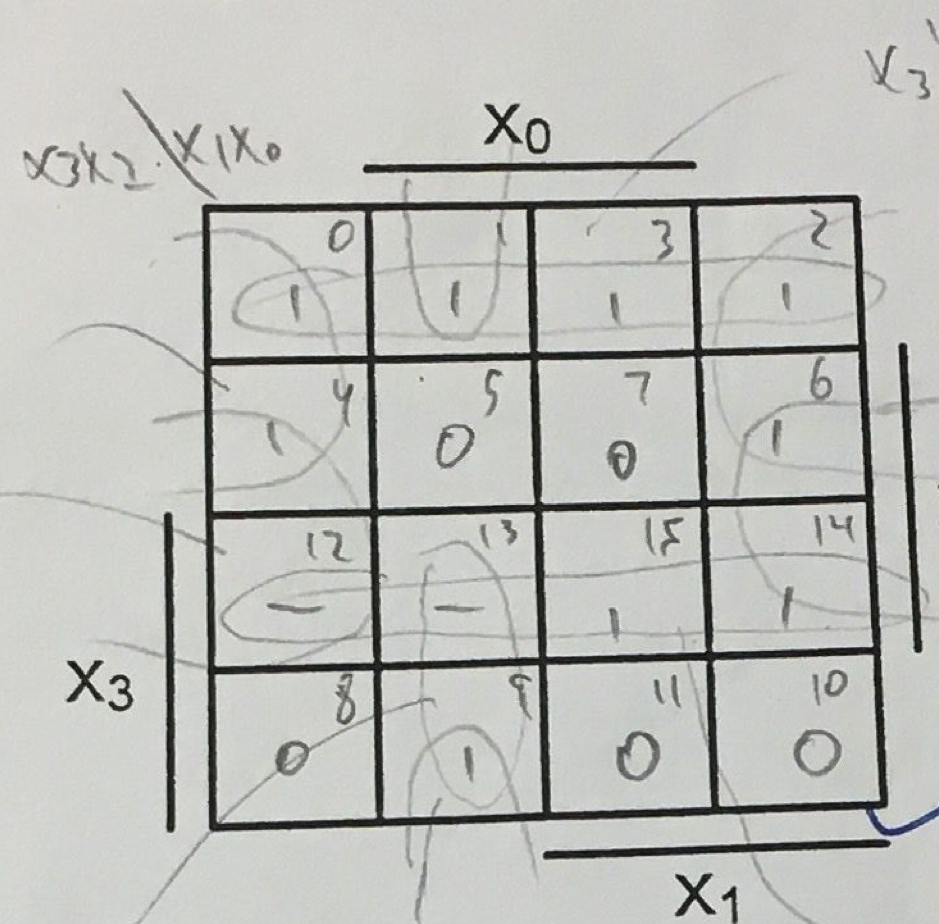
For the switching function $f(x_3, x_2, x_1, x_0)$, we are given the information below for the dc-set and zero-set.

dc-set = (12,13)

zero-set = 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.



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.

(a) x_3x_1
(b) $x_3'x_2'$
(c) $x_3'x_1$
(d) $x_3'x_0'$
(e) x_2x_0'
(f) x_3x_2
(g) $x_3x_2x_1$
(h) $x_3'x_2'x_1'$
(i) $x_2'x_1'x_0$
(j) $x_3x_1'x_0$
(k) $x_3'x_2x_1'x_0$
(l) $x_3'x_2x_1x_0'$

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

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.

5. (~~3~~ 5 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.

(a) $(x_3' + x_2')$
(b) $(x_3' + x_1')$
(c) $(x_3' + x_2 + x_0)$
(d) $(x_3' + x_2 + x_1')$
(e) $(x_3' + x_1 + x_0')$
(f) $(x_3 + x_2' + x_0')$
(g) $(x_3 + x_1' + x_0)$
(h) $(x_2' + x_1 + x_0')$
(i) $(x_3' + x_1 + x_0)$
(j) $(x_3 + x_1' + x_0')$
(k) $(x_3 + x_2 + x_1 + x_0')$
(l) $(x_3 + x_2' + x_1' + x_0)$

5

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

$(x_3 + x_2' + x_0'),\ (x_3' + x_2 + x_1'),\ (x_3' + [\text{illegible}] + x_0)$

2

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.

Min PS = $(x_3 + x_2' + x_0')(x_3' + x_2 + x_1')(x_3' + x_2 + x_0)$

3