

33.5

Alex Gilbert	904-770-190
--------------	-------------

Name **EEM16/CSM51A (Fall 2017)** SID #

Logic Design of Digital Systems

Prof. Ankur Mehta : mehtank@ucla.edu

Midterm | Wednesday Nov. 1, 2017

Instructions

Do not start working on this exam until you are instructed to do so. Please be sure to read the following instructions completely.

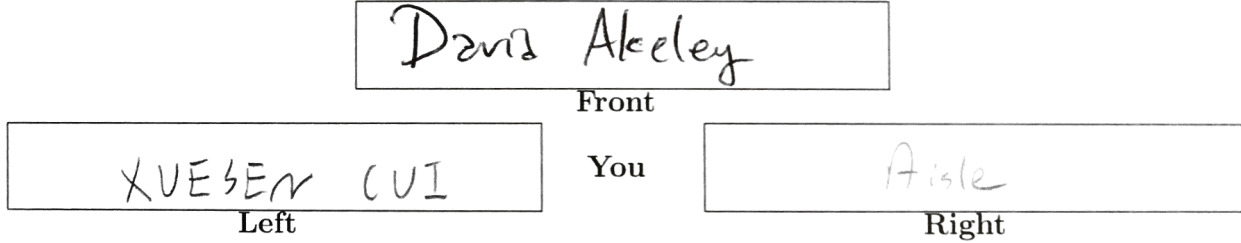
- You are permitted one (1) two-sided letter-sized 8.5"x11" page of notes.
- No electronic devices are permitted. You may leave numerical answers in the form of decimal (base-10) fractions or powers of 2 unless otherwise specified.
- You must show all work.
- You must provide your work and solution in the space provided after every question. There are additional pages at the end of the test if you need more room; clearly note in the space provided after the question if there is content on the extra pages. Anything else written anywhere else on the exam will not be considered.
- Fill out your name and SID# at the top of this page, and your SID# at the bottom left of every other page in this exam packet.
- Do not tear off or unstaple any pages from this exam packet.
- All pages in this packet must be accounted for; if any are missing, the exam will graded a 0.
- According to University policy, cheating will result one or more of the sanctions listed in the UCLA Student Conduct Code. Sanctions for violation of University policies regarding academic dishonesty include suspension or dismissal.

Contents

This exam consists of 5 problems, printed on 8 two-sided pages, including this one.

Seating

Please identify the students sitting near you:



1 Basics (1 point each subproblem, 15 pts total)

1.1 How would you represent the decimal value of -23 using ...

1.1(a). ... 6 bit 2s complement notation in binary?

-23 23 = 010111 -23 = 101000+1
101001

1.1(b). ... 6 bit signed magnitude notation in hexadecimal?

$(-23)_{10} = (010111)_2 = (00010111)_2 = (17)_{16}$

$\therefore \boxed{100017} \times$

~~-0.5~~
 -16² 16¹ 16⁰

1.2 What is the decimal value of 11010110 assuming ...

1.2(a). ... 5.3 fixed point (e.g. xxxxx.xxx), 2s complement notation?

$11010110 = (11010110)_2 \times 2^{-3} = -(00101001+1) \times 2^{-3} = -(00101010)_2 \times 2^{-3}$
 $= -42 \times \frac{1}{8} = -5 \frac{1}{4} = \boxed{-5.25}$ ~~32+2=42~~

1.2(b). ... 4E3 floating point (e.g. S EEE MMMM) notation?

$(-1)^1 \times (1 + (0.110)_2) \times 2^{(5-3)} = -1.375 \times 2^2 = \boxed{-1.375 \times 4}$
 $\frac{1}{4} + \frac{1}{8} = \frac{3}{8} = .375$

1.3 If you were trying to represent the decimal value 13/32, what is the closest value you could represent using ...

1.3(a). ... 5.3 fixed point (e.g. xxxxx.xxx) notation (as a decimal fraction)?

resolution = $2^{-3} = \frac{1}{8} = \frac{4}{32}$ \therefore you could represent $\frac{13}{32} = \boxed{.375}$
~~ie. (0000001)₂~~

1.3(b). ... 4E3 floating point (e.g. S EEE MMMM) notation (as that 4E3 binary value)?

$2^{-3} \times 1 = \frac{1}{8} = \frac{4}{32}$
0 000 0000 -1

-1.5

1.4 Consider two eight bit signals $X[7:0]$, $Y[7:0]$.

1.4(a). If you consider both as unsigned numbers, what is the numerical relationship of Y as a function of X if you set $Y[3:0]=X[7:4]$; $Y[7:4] = 4'b0000$?

$$X[3:0] \quad \boxed{X = Y \cdot 2^4}_{10}$$

$$X = \quad \quad \quad X = yyy y x x x x$$

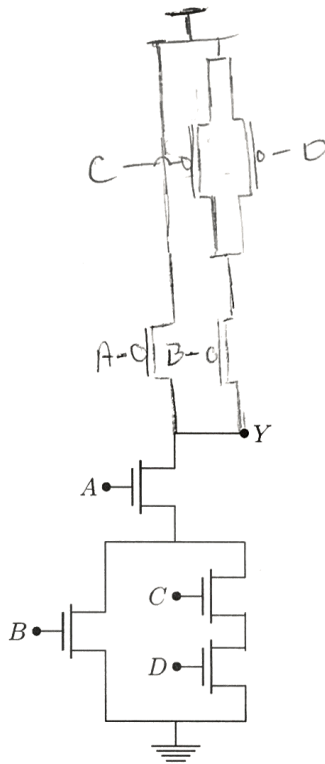
$$Y = 0000 y y y y \quad \quad \quad \checkmark \quad -0.5$$

1.4(b). How would you create that same numerical relationship of Y as a function of X if they were both signed numbers instead?

$$X \quad \quad \quad -)$$

1.5 Consider the given the pull-down network for a CMOS logic gate.

1.5(a). Draw in the pull-up network.

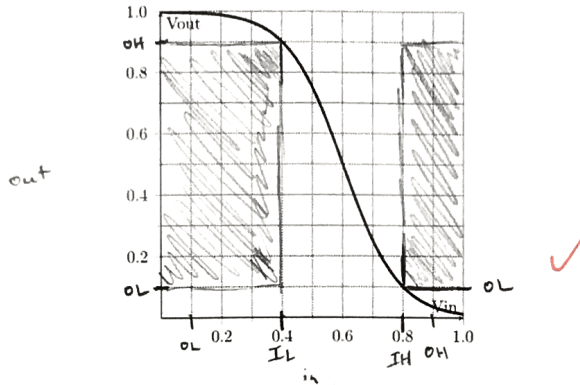


1.5(b). What is a Boolean expression for Y as a function of A, B, C, D ? (You do not need to simplify or expand this expression.)

$$\overline{Y} = A \wedge (B \vee (C \wedge D))$$

$$\boxed{Y = A \wedge (B \vee (C \wedge D))}$$

1.6 Consider the following VTC for a CMOS inverter.



1.6(a). If $V_{OL}=0.1$ and $V_{OH}=0.9$, what is the forbidden zone with the smallest width that we could set for V_{in} ?

Smallest width = 0.2 → b/c curve crosses V_{OL} @ 0.8 input

1.6(b). On the VTC above, shade in the rectangular forbidden region(s) required for any CMOS gate to obey the resulting static discipline.

1.6(c). What is the noise immunity (smallest noise margin) in this static discipline?

$I_H = .8$ $O_H = .9$ ∴ noise immunity = $.9 - .8 = \boxed{.1}$

1.7 Consider a weighted six-sided die with the given probabilities.

i	p_i	bits
1	1/2	$\log_2(1/p_i) = \log_2(2) = 1$
2	1/4	2
3	1/16	4
4	1/16	4
5	1/16	4
6	1/16	4

1.7(a). Complete the table above with the information content associated with each outcome.

1.7(b). What is the entropy of this system?

$H = \sum_{i=1}^6 p_i \log_2(1/p_i) = \frac{1}{2} + (\frac{1}{4} \cdot 2) + (\frac{1}{16} \cdot 4) \cdot 4 = \frac{1}{2} + \frac{1}{2} + 1 = \boxed{2}$

2 Universal gates (8pts)

Consider the three input gate $Y = \text{CCNOT}(X_2, X_1, X_0)$, defined as follows:

$Y = \overline{X_2}$ if X_1 and X_0 are both 1; $Y = X_2$ otherwise.

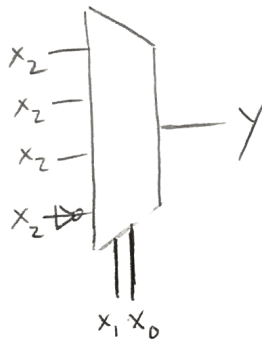
2.0(a). Draw the k-map for this function. (1pt)

	$x_1 x_0$		
x_2	00	01	11
0	0	0	1
1	1	1	0

x_1	x_0	Y
0	0	x_2
0	1	x_2
1	0	x_2
1	1	$\overline{x_2}$

u

2.0(b). Implement this gate, given only the non-inverted inputs, using a single inverter and a 4-1 mux. (3pts)



2.0(c). Is this gate universal? Prove your answer. (4pts)

universal if you can make AND, OR, NOT... or if you can make nand/nor since these are universal

If $x_2 = 0$, $Y = x_1 x_2$ \therefore can implement AND

If $x_2 = x_0$, $Y = \overline{x_0}$ \therefore can implement NOT
and $x_1 = x_0$

If $x_2 = 1$, $Y = \overline{x_1 x_0} = \text{NAND}$ \therefore universal

3 Superfunctions (12pts)

Consider a single valued function of n inputs $Y = f(X) = f(X_{n-1}, X_{n-2}, \dots, X_0)$.

3.0(a). There are M different such functions in total. If $n = 5$, what is M ? (2pts)

$$M = 2^{2^n} = 2^{2^5} = \boxed{32}$$

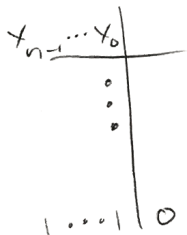
3.0(b). If we label each of the M possible function with a unique (binary) integer, we need m bits for this label. What is m ? (1pt)

$$m = \log_2(M) = \log_2(2^{32}) = \boxed{32}$$

3.0(c). Consider a specific function f_i , which we know can be implemented by a single CMOS gate. If there is a specific input $A = A_{n-1}, A_{n-2}, \dots, A_0$, such that $A_{n-1} = 0$ and $f_i(A) = 0$, what, if anything, can we say about $f_i(A')$, where $A' = 1, A_{n-2}, \dots, A_0$? Why? (2pts)

~~$f_i(A')$ can't be represented by a single CMOS gate~~

3.0(d). We can define a superfunction $g(I, X)$ that takes in an m bit input I identifying the function and the n bit input X , and returns $Y = f_I(X)$ for the specific function f_I labeled by the value I . Come up with the mapping from I to f_I that lets you implement this function with a single mux. For what value I is f_I an n -input NAND gate? (3pts)



$$2^m = M = 2^{2^n} \quad \text{or} \quad m = 2^n$$

$I =$

$$\boxed{I = M}$$

Need is highest for

~~M sets of m input lines: each m input line is a function f_I~~

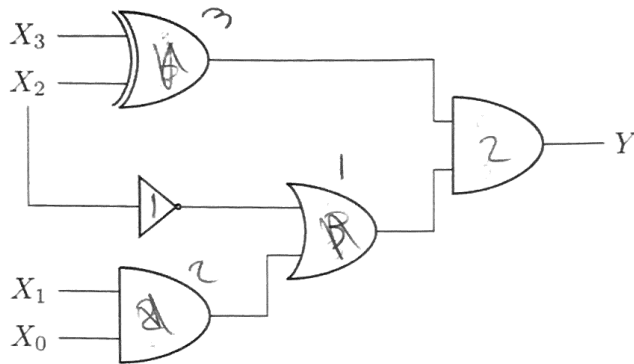
3.0(e). Draw and label this mux-based implementation of $Y = g(I, X)$. (4pts)

-4



4 Boolean gates (8pts)

Consider the following Boolean logic system, with gate delays as in the corresponding table.



Gate	t_{PD}	t_{CD}
NOT	1	1
OR	3	1
AND	2	2
XOR	6	3

X_3	X_2	XOR
0	0	0
0	1	1
1	0	1
1	1	0

X_3	X_2	X_1	X_0	n	Y
0	0	0	0	0	0
0	0	0	1	1	0
0	0	1	0	2	0
0	0	1	1	3	0
0	1	0	0	4	0
0	1	0	1	5	0
0	1	1	0	6	0
0	1	1	1	7	1
1	0	0	0	8	1
1	0	0	1	9	1
1	0	1	0	10	1
1	0	1	1	11	1
1	1	0	0	12	0
1	1	0	1	13	0
1	1	1	0	14	0
1	1	1	1	15	0

- 4.0(a). Assume we are able to implement the Boolean gates directly as shown in the diagram above. Given the propagation and contamination delays of each of those gates, what are the propagation and contamination delays of this system? (3pts)

$$t_{PO} = 8 \quad t_{CO} = 4$$

- 4.0(b). If each Boolean gate in the system is lenient, is the entire system of part 4.0(a) lenient? If it is, explain why. If it is not, draw a timing diagram proving that the system can glitch. (3pts)

It is not ~~necessarily~~ lenient
b/c

x_3 _____

x_2 _____

x_1 _____

x_0 _____

~.

- 4.0(c). Complete the truth table above for Y as a function of X_3, X_2, X_1, X_0 . (1pt)

- 4.0(d). Write the function using either minterm or maxterm shorthand, that is, in the form $Y = m(n_1, n_2, \dots)$ or $Y = M(n_1, n_2, \dots)$, where n_1, n_2, \dots are integers. (1pt)

$$m(7, 8, 9, 10, 11)$$

5 Privileged Encoder (17 pts)

Recall that a decoder goes from an n -bit binary encoded input to an m -bit one-hot output. We would like to build an **encoder** to go the other way, generating an n -bit binary encoded integer $Y = Y_{n-1}Y_{n-2} \dots Y_0$ from an m -bit input $X = X_{m-1}X_{m-2} \dots X_0$.

However, since we can't guarantee that the m -bit input will be one-hot, we will need to handle the case when either several or none of the input bits are high. To do so, we will have an additional one-bit input P indicating which direction is privileged, and an additional one-bit output H indicating whether any of the input bits were high:

- If none of the input bits are high, we *don't care* what the n -bit output Y is, but the one bit output $H = 0$.
- If exactly one of the input bits is high, then Y is the index of the input that is high, and $H = 1$.
- If multiple of the input bits are high, and $P = 1$, then Y is the largest index of the inputs that are high, and $H = 1$.
- If multiple of the input bits are high, and $P = 0$, then Y is the smallest index of the inputs that are high, and $H = 1$.

5.0(a). What must n be as a function of m ? (1pt)

$$m = 2^n$$

$$n = \log_2 m$$

5.0(b). Let us first consider the case where $m = 2$. Draw the truth table for outputs Y and H as a function of inputs P and X . (1pt)

P	x_1	x_0	H	Y
0	0	0	0	X
0	0	1	1	0
0	1	0	1	1
0	1	1	1	0
1	0	0	0	X
1	0	1	1	0
1	1	0	1	1
1	1	1	1	1

5.0(c). Use a k-map for $m = 2$ to determine what the *don't care* values should be to minimize a product-of-sums implementation for Y . In this case, what is the minimal Boolean PoS expression for Y ? (2pts)

P, x_0	00	01	11	10
0	X	0	0	1
1	X	0	1	1

Let the x_1 's = 0

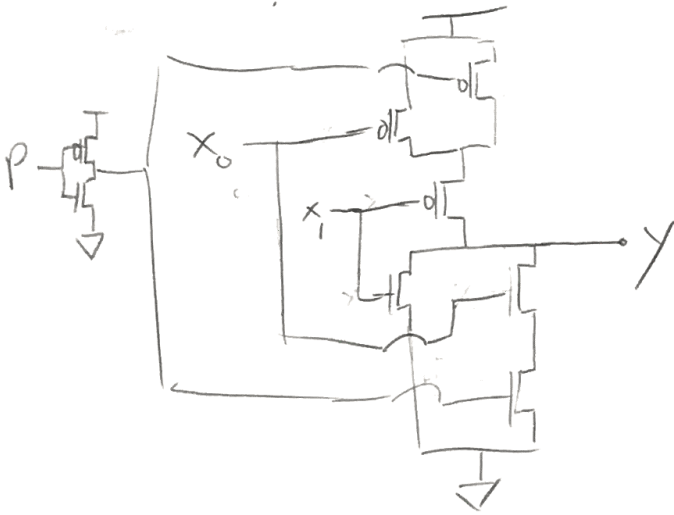
$$\bar{y} = \bar{x}_1 + \bar{P}x_0$$

$$= \bar{x}_1 (\bar{P} + x_0) = \bar{x}_1 (P + \bar{x}_0)$$

$$y = \bar{x}_1 (P + \bar{x}_0)$$

- 0.5

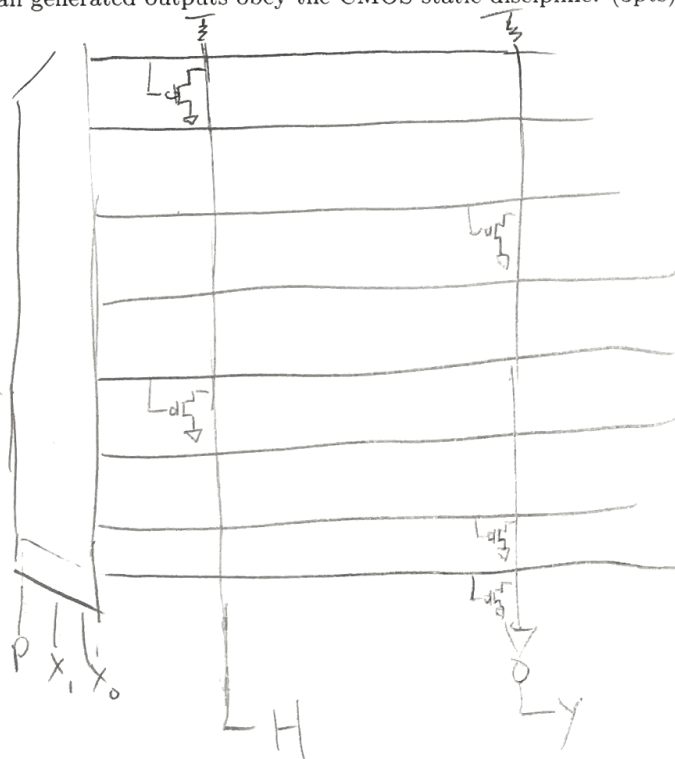
5.0(d). Implement this expression for Y using 3 CMOS gates: two inverters and one 6-transistor gate. (3pts)



$$\bar{y} = x_1 + \bar{P}x_0$$

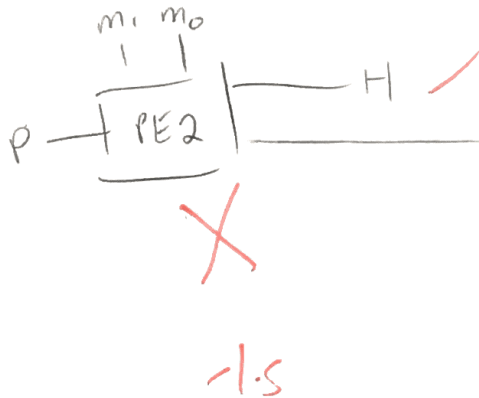
-2

5.0(e). Implement PE2, the privileged encoder for $m = 2$, using a ROM to generate both Y and H from inputs P and X . Ensure that all generated outputs obey the CMOS static discipline. (3pts)



-2.5

- 5.0(f). A **arbiter** generates an m -bit one-hot output from an m -bit input. Use the PE2 block plus any additional CMOS gates to implement the PA2 block, which outputs H as from the PE2 block and A , an $m = 2$ -bit one-hot output that corresponds to the index Y as set from PE2. (2pts)



- 5.0(g). Use two PA2 blocks, plus any CMOS gates or muxes, to generate the PA4 block, which is the privileged arbiter generating an $m = 4$ bit one-hot output A and a one bit output H from $m = 4$ bit input X and one bit input P . (3pts)

-3

5.0(h). Using any CMOS gates or muxes, generate the n -bit result Y from the $m = 4$ -bit output A of the PA4 block.
(2pts)

