8

Problem No. 1 (10 points)

Part (a) By using binary code, at least how many bits would be needed to encode the combination of your gender (M, F), your year (freshman, sophomore, junior, senior, graduate), and the first two-digit of your Student ID (assuming each digit is a decimal number)?

Answer: 11 bits.

Show your work below for full credit.

M F (0, 1)
Year (00, 01, 10, 11)
ID decimal (- - - -, - - - -) up

1 + 2 + 4 + 4

Part (b) An electronic cylinder *iCylinder* measures liquid capacity by one 3-digit mixed-radix number system (*gallon*, *quart*, *pint*). The relationships between three capacity units of this number system are: 1 *gallon* = 4 *quarts*, 1 *quart* = 2 *pints*. Assume that radix for digit *gallon* is sixteen.

(b.1) Using binary code, at least how many bits are needed to encode the largest number of *pints* that can be represented by this number system?

Your Answer: 7 bits.

15 gallons 3 quarts 1 pint

3 quarts = 6 pints

15 gal × 4 quart / 1 gal × 2 pint / 1 quart

(b.2) How many *pints* are represented by a reading of $X = (11, 2, 1)$?

Your Answer: 93 pints.

Show all your work below for full credit:

11 6gal × 4 quarts / 1 gal × 2 pints / 1 quart = 88 pints

2 quarts = 4 pints 1 pint

120 + 6 + 1
127

Part (c) A binary number 10010010 is stored in a computer memory.

128 64 32 16 8 4 2 0

(c.1) If it is used to represent time in seconds of a car parking meter, how many seconds a car has parked in the spot?

Your Answer: 146 seconds.

2 + 16 + 128

Show all your work below for full credit:

(c.2) If it is used to represent the temperature of a car thermometer in integer degrees in Celsius, what is the temperature inside a car?

Your Answer: 146 Celsius.

-110° C in 2's comp

Show all your work below for full credit:

15

Problem No. 2 (15 points)

Part (a) Name the following 2-input switching functions using primitive logic functions:

(a.1) The output of f_1 is 1 **if and only if** both inputs are 1. It is AND function.

(a.2) The output of f_2 is 1 if the inputs **are different**. It is XOR function.

(a.3) The output of f_3 is 1 **if no more than** one input is 1. It is NAND function.

(a.4) The output of f_4 is 1 **only when** both inputs are 0. It is NOR function.

(a.5) The output of f_5 is 1 **if at least** one input is 1. It is OR function.

(a.6) The output of f_6 is 1 if number of 1's in inputs **are even**. It is XNOR function.

Part (b) The logic designer Logik Luv plans to design one single two-input (x, y) "multi-function" logic module with two-output (g_1, g_2) to implement these switching functions in Part (a). To do so, he introduces two additional inputs (a, b) that decide the output functions g_1 and g_2 as follows:

a	b	g_1	g_2
0	0	x'	y'
0	1	f_1	f_2
1	0	f_3	f_4
1	1	f_5	f_6

(b.1) Filling in the following 2-D truth table for the functions g_1 and g_2. Place values of g_1 and g_2 in each cell **in order of** (g_1, g_2). $g_1\ g_2$

		(x,y)			
		00	01	10	11
(a,b)	00	11	10	01	00
	01	00	01	01	10
	10	11	10	10	00
	11	01	10	10	11

(Continued on the next page)

(Problem No. 2 - Continue)

(b.2) Write the *minterm* expression for $g_1(a, b, x, y)$ in compact form:

$g_1(a, b, x, y) = \sum m$ {0, 1, 7, 8, 9, 10, 13, 14, 15}

(b.3) The *switching* expression for m_{14} is $abxy'$.

(b.4) Write the *maxterm* expression for $g_2(a, b, x, y)$ in compact form:

$g_2(a, b, x, y) = \prod M$ {1, 3, 4, 7, 9, 10, 11, 13, 14}

(b.5) The *switching* expression for M_{14} is $a' + b' + x' + y$.

$a' + b' + x' + y$

(End of Problem No. 2)

Problem No. 3 (10 points) 10

Your high school buddy BB Frank is interviewed for an internship position at a startup *Cookle Electronics*. One of his interview questions is the tabular minimization using the Quine-McCluskey algorithm for the following 4-input switching function:

$$f(a,b,c,d) = \sum m(0,1,2,5,6,7,8,9,10,14)$$

BB Frank has completed the first step and the Prime Implicant Chart is shown below. You have to help him identify the essential prime implicants and then write the minimal AND-OR (sum-of-product) expression for $f(a,b,c,d)$.

Prime Implicant Chart

Prime Implicants	Minterms									
	0	1	2	5	6	7	8	9	10	14
−00−	X	X					X	X		
−0 − 0	X		X				X		X	
− − 10			X		X				X	X
0 − 01		X		X						
01 − 1				X		X				
011−					X	X				

Part (a) The essential prime-implicants in switching expressions are:

b'c' + cd' ✓

Part (b) The minimal switching expression in AND-OR form is:

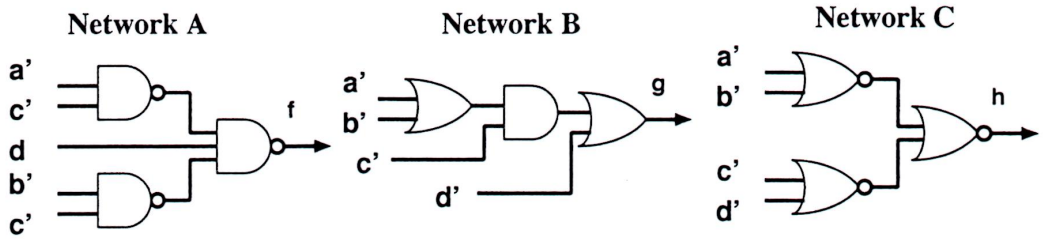
$f(a,b,c,d) =$ b'c' + cd' + a'bd ✓

Show all your work on the *prime implicant chart* above for full credit.

15

Problem No. 4 (15 points)

Three gate networks A, B and C are given below. Tests have shown that two of them are *equivalent*, that is, they implement the same switching function. You are asked to identify the network that <u>is not</u> equivalent.



Part (a) Describe your approach concisely in one sentence.

Answer: Use boolean algebra to reduce networks and find a non equivalent expression.

Part (b) The non-equivalent network is C.

<u>Show</u> all your work below for credit:

$((a'c')'d(b'c')')'$

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

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

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

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

$a'c'+b'c'+d'$

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

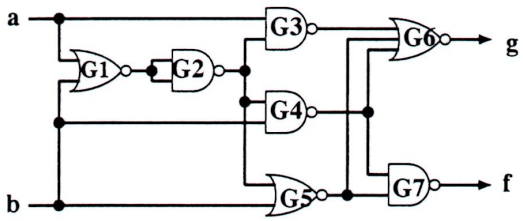
$(a'+b')(c'+d')$

$a'c'+a'd'+b'c'+b'd'$

Problem No. 5 (20 points)

20

Given the gate network below, answer the following questions:



HL

66

Part (a) Assuming that negated variables are available and that NAND and NOR gates have the same delays, identify the *critical path* of the network by listing its gates along the path, starting at the inputs:

G1 → G2 → G4 → G6 ✓

Part (b) Assuming that load factors of all gates equal to 1 and that both outputs f and g have the output load value L, list the output load value of every gate in the *critical path* (e.g., G3: 1):

G1: 2 G2: 3 G4: 2 G6: L ✓

Part (c) Write the expression of the longest network propagation delay T_{pHL} in terms of delays of each gate (You **do not** need to compute the final result but the transition direction at each gate has to be indicated):

66

$T_{pHL} =$ $T_{pLH}(G1) + T_{pHL}(G2) + T_{pLH}(G4) + T_{pHL}(G6)$ ✓

Part (d) Assuming that negated variables are available, find the minimal switching expression of the output f in two-level AND-OR (sum-of-product) form. Show your work below for full credit.

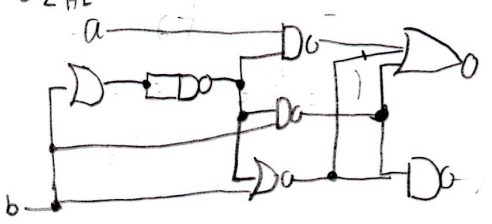
work on back sheet

$f(a,b) =$ a+b ✓.

Your work for Part (d):

66 HL
61 LH
64 LH
62 HL

(Extra space available on the next page)



62 = a+b
G5 = (a+b)' = a'b'
G4 = (ab+b)' =
G7 = (G4 G5)'
64 = (b G2)' G5 = (G2+b)'
G2 = (G1G1)' G1 = (a+b)'
G2 = ((a+b)'(a+b)')'
= ((a+b)' + a+b)
= a+b

G7 (Extra space for Problem No. 5)

$G7 = (G4\ G5)'$

$G4 = (b\ G2)^2$

$G5 = (G2+b)'$

$G2 = (G1\ G1)'$

$G1 = (a+b)'$

$G2 = ((a+b)'(a+b)')'$

$(a'b' + a'b')'$

$= (a'b')'$

$G2 = a+b$

$G5 = (a+b\ +b)'$

$= (a+b)'$

$= a'b'$

$G4 = (b(a+b))'$

$= (ab+b)'$

$= (b(a+1))'$

$= b'$

$G7 = (b'(a'b'))'$

$= (a'b')'$

$= a+b$

$G7 = (G4\ G5)'$

$G4 = (b\ G2)'$

$G5 = (G2+b)'$

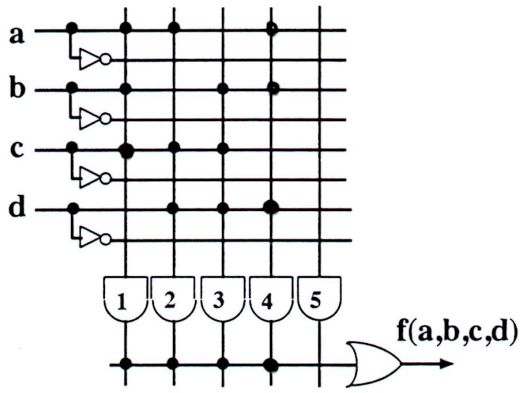
$G2 = (G1\ G1)'$

$G1 = (a+b)'$

(End of Problem No. 5)

Problem No. 6 (15 points)

The novice engineer Goofy has been asked to design a "three-out-of-four" majority circuit using PLAs. The output function $f(a, b, c, d)$ of the circuit is 1 if there are **at least** three 1's in four inputs (a,b,c,d), and is 0 otherwise. He has worked out a minimal solution and implemented his design in the PLA as shown below. Regrettably, he made **one logic single** mistake during the design. As a result, the circuit does not work correctly.

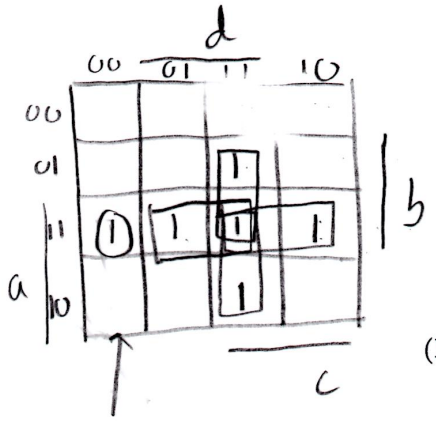


Your tasks in the problem are to help Goofy:

1. Find the mistake. Show all your work in the provided space below.
2. Correct the error by adding and/or removing connections on the original PLA diagram above. Use a "X" for a connection removed and a heavy dot "•" for a connection added. Your correction **must be minimal**.

The mistake is that he included the input of $ab\bar{c}\bar{d}$ which isn't actually a three out of four

First derive switching expression $(ab + acd + bcd)$



To fix the error, we must replace ab with two more essential implicants abd and abc

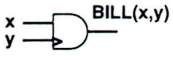
+ 15

(Extra space available on the next page)

error at 11 not true its in input not valid

Problem No. 7 (15 points)

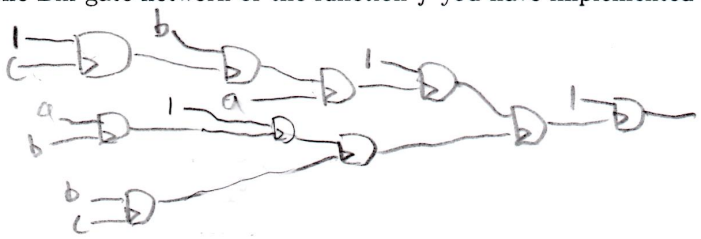
Armed with a solid A in Spring 2015 CSM51A, Bruinie has easily landed a job as a digital design engineer in *WideCom* after graduation. He is motivated to revolutionize the logic design and has invented a new type of logic gate, called *BILL* gates. Its symbol and truth table are given below. He prove the gate is a universal set.



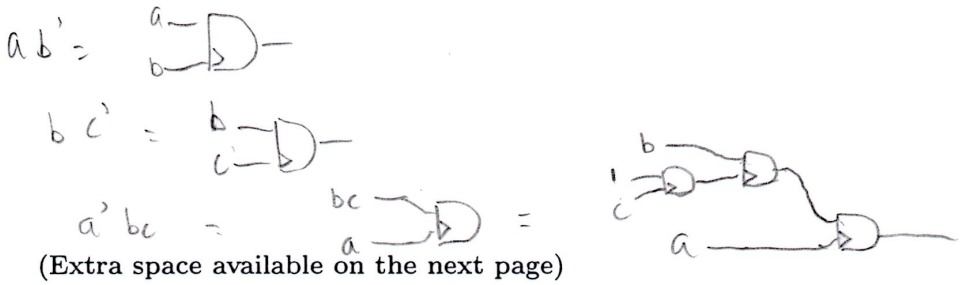
x	y	BILL(x,y)
0	0	0
0	1	0
1	0	1
1	1	0

Using only bill gates and constant 0 and constant 1, you are required to implement the switching function $f(a, b, c) = ab' + bc' + a'bc$.

Show the Bill gate network of the function f you have implemented below:

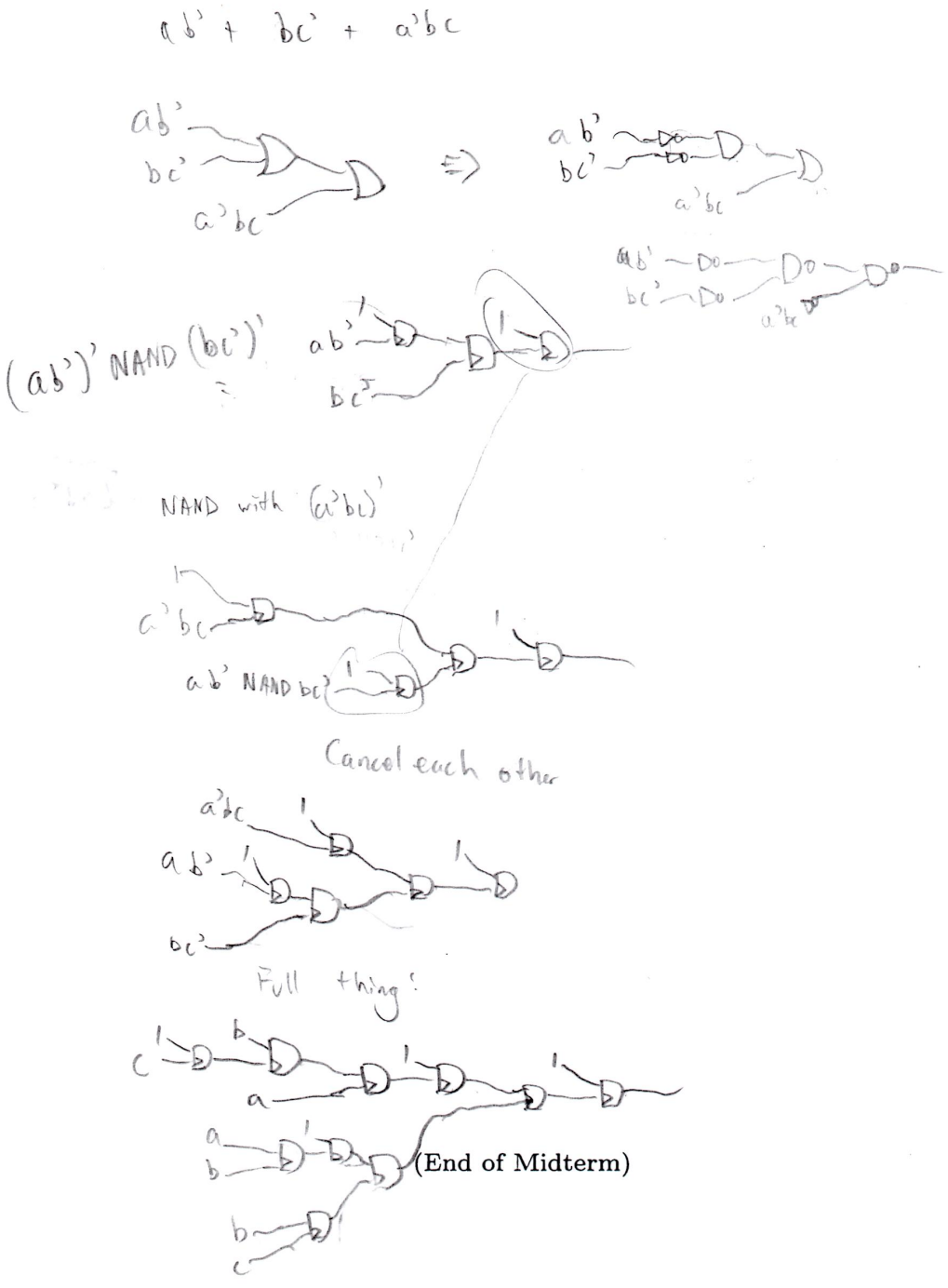


Show all your work below for full credit.



(Extra space available on the next page)

(Extra space for Problem No. 7)



(End of Midterm)