Midterm Exam

Name (Last, First): Solution

Student Id #:

Student to Left:

Student to Right:

Do not start working until instructed to do so.

1. *You must answer in the **space provided** for answers after every question. We will ignore answers written anywhere else in the booklet. **All pages in this booklet must be accounted** for otherwise it will not be graded.*
2. *You are permitted 1 page of notes 8.5x11 (front and back).*
3. *You may not use any electronic device.*

Following table to be filled by course staff only

	Maximum Score	Your Score
Question 1	15	
Question 2	25	
Question 3	25	
Question 4	35	
Question 5 (EC)	+5	
TOTAL	100	

Question #1 (15 pts)

Consider the following Karnaugh Map for the Boolean function, Y. A blank truth table is provided for your convenience.

AB

CD	"00"	"01"	"11"	"10"
"00"	0	0	1	1
"01"	0	1	0	1
"11"	0	1	0	0
"10"	1	1	X	X

A	B	C	D	Y
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

(a) Circle the prime implicants on the map. (5 pts)
How many prime implicants are there? 5

(b) Write the Boolean (sum-of-product) expression of the *essential* prime implicants of (b) (if any). (5 pts)

EssentialPrimeImplicants = $C\bar{D} + \bar{A}BD + A\bar{D} + A\bar{B}\bar{C}$

4 essential prime implicants

1. $C\bar{D}$
2. $\bar{A}BD$
3. $A\bar{D}$
4. $A\bar{B}\bar{C}$

(c) Express as a minimal sum of product, $\neg Y$. (5 pts)
The K-map is provided for your convenience.

AB

CD	"00"	"01"	"11"	"10"
"00"	0	0	1	1
"01"	0	1	0	1
"11"	0	1	0	0
"10"	1	1	X	X

① $A + C + D$

② $A + B + \overline{D}$

③ $\overline{A} + \overline{B} + \overline{D}$

④ $\overline{C} + \overline{A}$

$(A + C + D)(A + B + \overline{D})(\overline{A} + \overline{B} + \overline{D})(\overline{A} + \overline{C})$ ← Y as product of sums

$$\overline{Y} = \overline{(Y)}$$

$$= \overline{(A + C + D)(A + B + \overline{D})(\overline{A} + \overline{B} + \overline{D})(\overline{A} + \overline{C})}$$

$$\overline{Y} = \overline{A}\overline{C}\overline{D} + \overline{A}\overline{B}D + ABD + AC$$

Question #2 (25 pts)

(a) Is DeMorgan's theorem still true with more than two variables? If so, prove it in the case of three variables x, y and z. (5 pts) Yes ; Multiple ways to prove this (truth table, Venn diagram, 2 variable De Morgan's theorem)

Using De Morgan's theorem with 2 variables: 4 possible cases

① $\overline{x \vee y \vee z}$
$\overline{(x \vee y)} \wedge \bar{z}$
$\bar{x} \wedge \bar{y} \wedge \bar{z}$

② $\overline{x \wedge y \wedge z}$
$\overline{(x \wedge y)} \vee \bar{z}$
$\bar{x} \vee \bar{y} \vee \bar{z}$

③ $\overline{(x \wedge y) \vee z}$
$\overline{(x \wedge y)} \wedge \bar{z}$
$\bar{x} \vee \bar{y} \wedge \bar{z}$

④ $\overline{(x \vee y) \wedge z}$
$\overline{(x \vee y)} \vee \bar{z}$
$\bar{x} \wedge \bar{y} \vee \bar{z}$

- All other combinations of x, y, z work because of the commutative property.

(b) Rewrite the following Boolean equation in (Disjunctive) Normal form. (6 pts)

$$f = \overline{A \oplus B} + \overline{B \oplus C}$$

where $\oplus$ means XOR operation, i.e., $A \oplus B = A\bar{B} + \bar{A}B$

* Note: Fully disjunctive normal form was intended but credit will be given for any valid disjunctive form

Answer:
$$f = \overline{(A\bar{B} + \bar{A}B)} + \overline{(B\bar{C} + \bar{B}C)}$$
$$= (\bar{A}+B)(A+\bar{B}) + (\bar{B}+C)(B+\bar{C})$$
$$= (\bar{A}A + \bar{A}\bar{B} + AB + B\bar{B}) + (\bar{B}B + \bar{B}\bar{C} + BC + C\bar{C})$$
$$= 0 + \bar{A}\bar{B} + AB + 0 + 0 + \bar{B}\bar{C} + BC + 0$$
$$= \bar{A}\bar{B} + AB + \bar{B}\bar{C} + BC$$
$$= (\bar{A}\bar{B}C + \bar{A}\bar{B}\bar{C}) + (ABC + AB\bar{C}) + (A\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C}) + (ABC + \bar{A}BC)$$

$$f = \underline{\bar{A}\bar{B}C + \bar{A}\bar{B}\bar{C} + ABC + AB\bar{C} + A\bar{B}\bar{C} + \bar{A}BC}$$

(c) Simplify f from (b) to a minimum sum-of-products. List which Boolean properties you use at each step of the simplification. Hint: you may use K-map to ***verify*** your answer. (6 pts)

Answer:
$$f = (\bar{A}\bar{B}C + \bar{A}\bar{B}\bar{C}) + (ABC + AB\bar{C}) + (A\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C}) + (ABC + \bar{A}BC) \rightarrow X + X = X \text{ (Idempotence)}$$
$$= \bar{A}\bar{B} + AB + \bar{B}\bar{C} + BC \rightarrow XY + X\bar{Y} = X \text{ (combining)}$$
$$= \bar{A}\bar{B} + AB(C+\bar{C}) + (A+\bar{A})\bar{B}\bar{C} + BC \rightarrow X + \bar{X} = 1$$
$$= \bar{A}\bar{B} + ABC + AB\bar{C} + A\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C} + BC \rightarrow \text{distributive}$$
$$= (\bar{A}\bar{B} + \bar{A}\bar{B}\bar{C}) + (ABC + BC) + (A\bar{C}B + A\bar{C}\bar{B}) \rightarrow \text{commutative}$$
$$= \bar{A}\bar{B} + BC + A\bar{C} \rightarrow \text{combining and absorption}$$

AB \ C	0	1
00	1	1
01	0	1
11	1	1
10	1	0

Should be 3 terms!

$f = \underline{\bar{A}\bar{B} + BC + A\bar{C}}$ (could also be)

should be either: $\bar{A}\bar{B} + BC + A\bar{C}$
or
$\bar{A}C + AB + \bar{B}\bar{C}$

- Solution to find alternative SoP is similar

(d) With only 2-input NOR gates, implement f with a minimal number of gates. Draw the gate diagram. (Note: no complemented inputs are given) (8 pts)

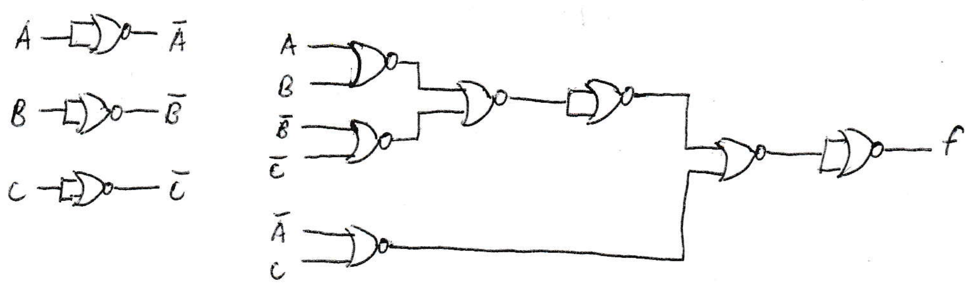
Using the minimum SoP from c)

$$f = \bar{A}\bar{B} + BC + A\bar{C}$$

$$= \overline{(A+B)(\bar{B}+\bar{C})(\bar{A}+C)}$$

$$= \overline{\overline{[\overline{(A+B)} + \overline{(\bar{B}+\bar{C})}]}\,(\bar{A}+C)}$$

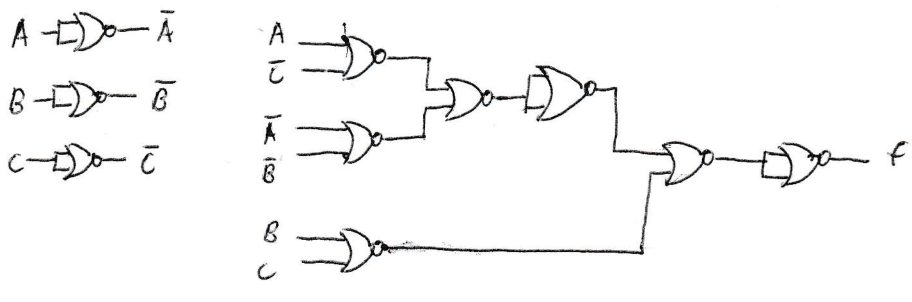
$$= \overline{\overline{\overline{[\overline{(A+B)} + \overline{(\bar{B}+\bar{C})}]}} + \overline{(\bar{A}+C)}}$$



3 + 3 + 2 + 2 = 10 gates required

Alternatively, ~~f~~ $f = \bar{A}C + AB + \bar{B}\bar{C}$

$$= \overline{\overline{\overline{[\overline{(A+\bar{C})} + \overline{(\bar{A}+\bar{B})}]}} + \overline{(B+C)}}$$



Question #3 (25 pts)

The following 12-bit word can be used to represent different numbers depending on the encoding

1's complement 2's complement (+1)

12b'1110_0110_1101 → 0001_1001_0010 → 0001_1001_0011

$2^8 + 2^7 + 2^4 + 2^1 + 2^0$
$= 256 + 128 + 16 + 2 + 1$
$= 384 + 19$
$= 403$

(a) If the word is 2's complement, what is the corresponding integer? (4 pts) -403

(b) If we convert the word (treated as unsigned) into base-4, what is the represented number? (3 pts)

321231

$\underbrace{11}_{3}\ \underbrace{10}_{2}\_\underbrace{01}_{1}\ \underbrace{10}_{2}\_\underbrace{11}_{3}\ \underbrace{01}_{1}$

(c) If we take answer in (b), extending how we define 1's complement for base-2, write the 3's complement of the base-4 number. (4 pts)

012102

(d) What is this word in Hexadecimal? (3 pts) E6D

(e) In base-20 system, assume each digit is now 00, 01, 02, … 09, 10, 11, …19 (each called a "vigit"). For example, 01,19 is 39 in decimal. Using 3 "vigits":

How would one represent a base-10 integer 1246? (4 pts) 3, 2, 6

1246
-1200 ← $20^2 \times 3$
46
40 ← $20^1 \times 2$
6 ← $20^0 \times 6$

What's the 20's complement representation of -1246 (i.e. the 20's complement of the 1246)? (4 pts) 16, 17, 14 ← +1 19's complement: 16, 17, 13

Using the first vigit as the sign vigit, what is the most positive value in base-10 integer that can be represented? (3 pts) 3999

Max value: 9, 19, 19 (1st vigit still has range 0-19)

$= 9 \times 400 + 19 \times 20 + 19$

$= 3600 + 380 + 19$

$= 3999$

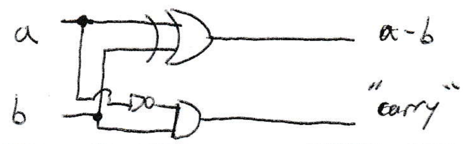
Question #4 (35 pts)

(a) Implement a one-bit "half-subtractor" from gates. The carry-out of this subtractor is 1 when the result is -1. The truth table for this is shown below: (8 pts)

a	b	a - b	"carry"
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

2 methods: ① by inspection of the truth table
② by K-map

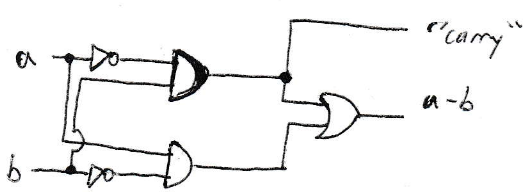
① Notice a-b is True only when a and b are different → XOR gate
"carry" is True only when a=0 and b=1 → $\bar{a}b$



② a-b

a \ b	0	1
0	0	1
1	1	0

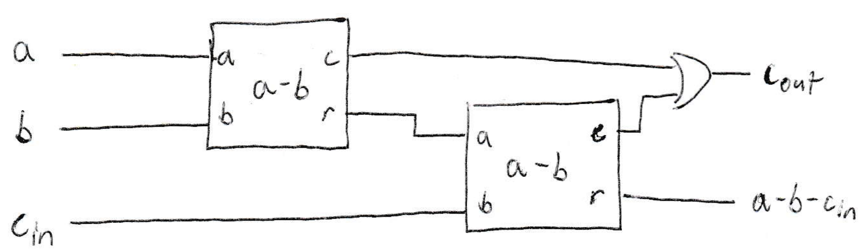
$a-b = \bar{a}b + a\bar{b}$



Either solution is valid

(b) Implement a "full-subtractor" from "half-subtractor" blocks. (6 pts)

↳ 3 inputs: a, b, "carry in"
2 outputs: a-b, "carry out"



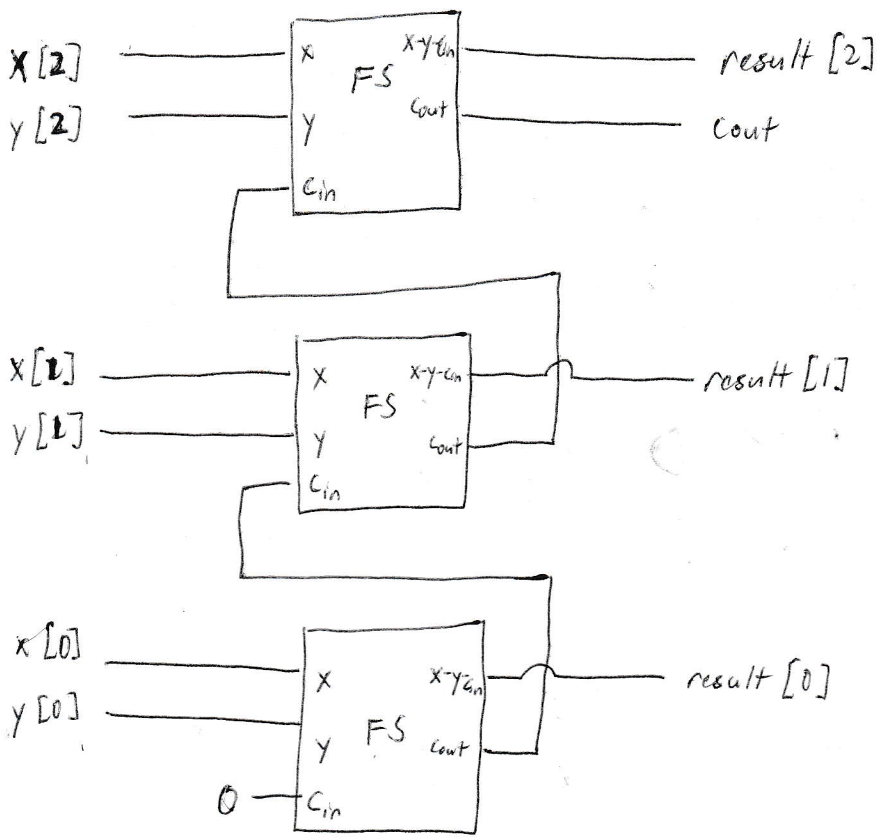
- Just like a full adder from half adders

a	b	c_{in}	a-b-c_{in}	c_{out}
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

(c) Implement a 3-bit "subtractor" from 1-bit "full-subtractor" blocks. (7 pts)

Implementing $X[2:0] - Y[2:0]$ (Index 2 is MSB, Index 0 is LSB)

$= result[2:0]$ with overflow in Cout



* Note: This solution works if inputs are assumed to be unsigned and the outputs are in 2's complement format.

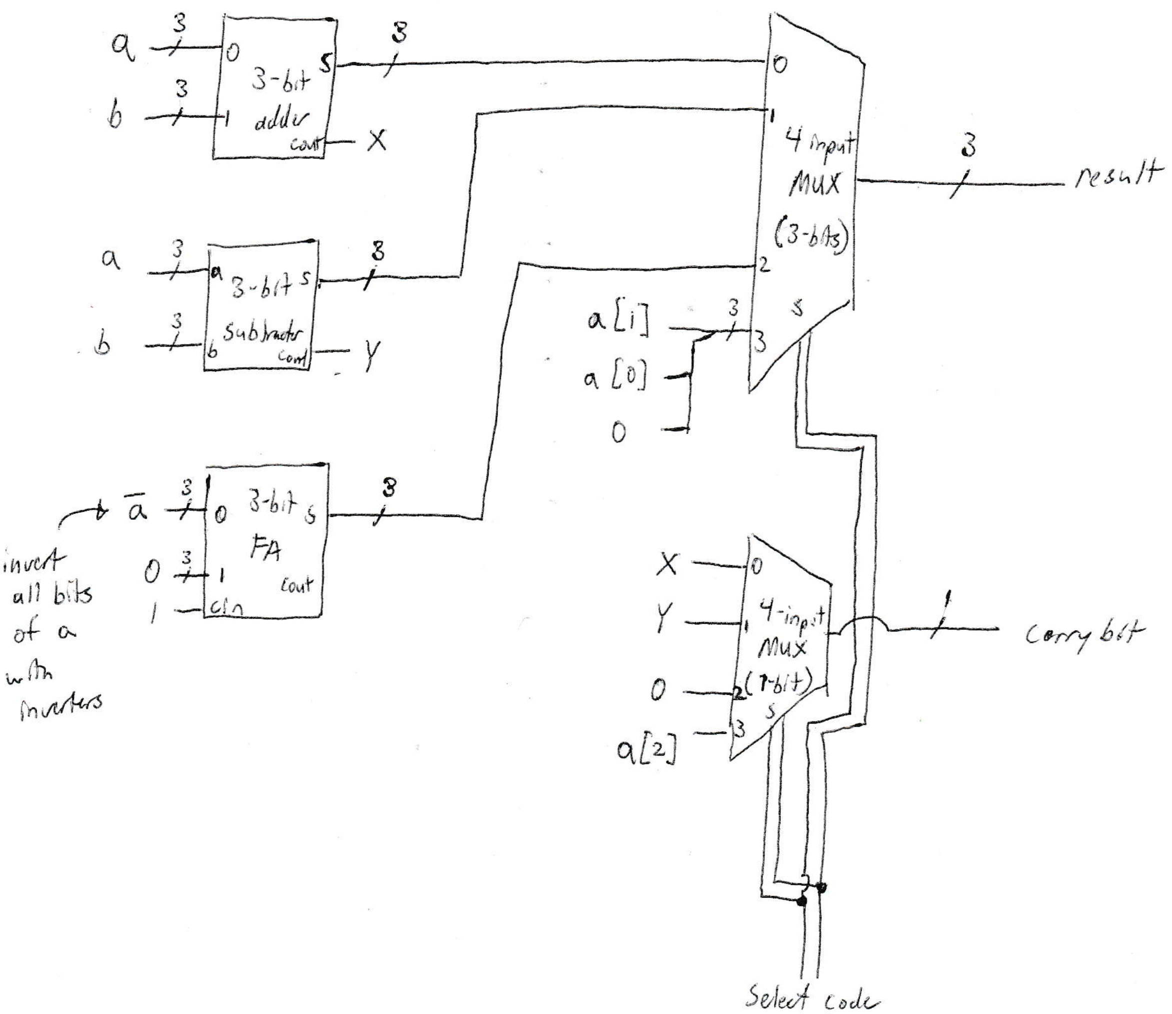
(d) Processors use a block called an ALU (Arithmetic Logic Unit) as part of their processing capability. Here we will implement a very basic ALU with a total of 4 functions, selected by a 2-bit code. Using the building blocks discussed in lecture and the 3-bit subtractor block, <u>implement a 3-bit ALU that can add, subtract, negate one argument, and multiply by 2</u>. The select codes are listed in the table below. Note that there are 3 inputs (3-bit a, 3-bit b, and the 2-bit select code) and 2 outputs (3-bit result and a 1-bit carry). (14 pts)

Hint: Multiplying a number is like shifting the bits to the left and using 0 as the lowest bit. An example: a = 1 = 3'b001 -> 2a = 2 = 4'b0010

Select Code	Result (3-bits)	Carry bit
00	a + b	carry out
01	a - b	carry out
10	-a	0
11	2*a	Product MSB

Adder
"Subtractor"
invert and add 1
change order of wires

- Combine all of these functions with MUXs



* Note: Ideally, -a actually requires 4-bits since -4 → +4 cannot be represented in 3-bits. This is a mistake in the problem and thus the -4 case will not work.

Question #5 (Extra Credit - 5 pts)

Implement a 4-bit Gray code +1 incrementor using building blocks (no gates). The 4-bit Gray codes are shown below.

Decimal Number	Gray Code	Unsigned Int. interpretation
0	0000	0
1	0001	1
2	0011	3
3	0010	2
4	0110	6
5	0111	7
6	0101	5
7	0100	4
8	1100	12
9	1101	13
10	1111	15
11	1110	14
12	1010	10
13	1011	11
14	1001	9
15	1000	8

& increment by 1 by switching to the next uint value
ex: $0 \rightarrow 1$ and $8 \rightarrow 0$ (wrap around)