ECE M16/CS M51A Midterm

[redacted]

TOTAL POINTS

96 / 105

QUESTION 1

1 Question 1-a 5 / 5

✓ **- 0 pts Correct**

- **1 pts** Missing 2x2 PI top/bottom right corner
- **1 pts** Missing bottom row PI
- **1 pts** Missing Upper Center PI
- **1 pts** Missing Lower Center PI
- **1 pts** Missing Right Center PI
- **1 pts** Introducing additional wrong implicants (not PIs)
- **5 pts** Ambiguous circling.

QUESTION 2

2 Question 1-b 5 / 5

✓ **- 0 pts Correct**

- **1 pts** Missed center EPI
- **1 pts** Missed right center EPI
- **1 pts** Missed 2x2 top/bottom-right corner EPI
- **1 pts** Missed bottom row EPI
- **0.5 pts** Missed bottom row EPI but consistent with previous answer
- **1 pts** Mistake made with EPI expression (circle is translated wrong)
- **1 pts** Inconsistent with previous answer (given previous answer the answer given here is wrong)
- **0.5 pts** Missed 2x2 top/bottom right EPI but consistent with previous answer
- **0.5 pts** * + notation error
- **5 pts** No answer
- **2 pts** A should be B, C should be D

QUESTION 3

3 Question 1-c 5 / 5

✓ **- 0 pts Correct**

- **1 pts** Missed 2x2 bottom right corner
- **0.5 pts** Expression is wrong from the circle. (Wrong translation from circle).
- **1 pts** Expression inverses are wrong (dual form is given).
- **1 pts** Unnecessary PI included
- **2 pts** POS is given when SOP is asked.
- **1 pts** Top row PI is missed.
- **1 pts** Left center PI is missed
- **1 pts** Center PI is missed.
- **1 pts** Inverse at the beginning is wrong.
- **5 pts** Should have started with 0s.
- **2 pts** A should be B and C should be D
- **0.5 pts** Click here to replace this description.
- **1 pts** Simplifications are wrong.
- **4 pts** Formed correct POS but nothing else.

QUESTION 4

4 Question 2-a 5 / 5

+ **1 pts** Answered yes.
+ **1 pts** Wrote the correct form
+ **3 pts** Provided proof
- **1 pts** Proof incomplete

✓ **+ 5 pts All correct**

QUESTION 5

5 Question 2-b 6 / 6

+ **1 pts** Wrote XOR to Boolean
+ **2 pts** Expanded the Boolean expression
+ **1 pts** Simplified the expanded expression

✓ **+ 6 pts Got the final DNF**

+ **0 pts** Nothing
- **1 pts** Did not simplify / not DNF / at least one term is wrong

QUESTION 6

6 Question 2-c 5 / 6

+ **6 pts** Got the minimal terms
✓ + **2 pts** Built on the previous answer
✓ + **1 pts** Drew the k-map
✓ + **2 pts** Verified using k-map
+ **0 pts** Nothing
- **1 pts** minor mistake

QUESTION 7

7 Question 2-d 8 / 8

+ **2 pts** Used NOR for other logics
+ **2 pts** Gate diagram for previous logic
✓ + **8 pts** All correct
+ **0 pts** Nothing
- **1 pts** Not minimum / minor issue

QUESTION 8

8 Question 3-a 4 / 4

✓ + **4 pts** Correct
+ **0 pts** Wrong

QUESTION 9

9 Question 3-b 3 / 3

✓ + **3 pts** correct
+ **0 pts** wrong

QUESTION 10

10 Question 3-c 1 / 4

+ **4 pts** correct
+ **0 pts** wrong
+ **2 pts** partially correct from previous wrong answer
✓ + **1 pts** taking 4's complement

QUESTION 11

11 Question 3-d 3 / 3

✓ + **3 pts** Correct
+ **0 pts** Wrong

QUESTION 12

Question 3-e 11 pts

12.1 3-e-1 4 / 4

✓ + **4 pts** Correct
+ **0 pts** Wrong

12.2 3-e-2 4 / 4

✓ + **4 pts** Correct
+ **0 pts** Wrong
+ **2 pts** partially correct from previously wrong answer
+ **1 pts** Taking 19's complement

12.3 3-e-3 3 / 3

✓ + **3 pts** Correct
+ **0 pts** Wrong
+ **1 pts** partially correct

QUESTION 13

13 Question 4-a 8 / 8

✓ + **8 pts** Correct
+ **4 pts** Used blocks/non-gate components but correct output
+ **4 pts** Carry wrong
+ **4 pts** Difference output wrong
+ **0 pts** No design
+ **0 pts** Wrong output for carry and difference

QUESTION 14

14 Question 4-b 6 / 6

✓ + **6 pts** Correct
+ **3 pts** Didn't use half-subtractors but correct output
+ **3 pts** carry out wrong
+ **3 pts** difference wrong
+ **0 pts** Wrong/No design

QUESTION 15

15 Question 4-c 7 / 7

✓ + **7 pts** Correct
+ **6.5 pts** Missing first cin but otherwise correct
- **0.5 pts** Missing labels for the bit order (order does matter)
+ **5 pts** Close design but not correct
+ **5 pts** Subtracted cin from all bits
+ **3 pts** Incomplete/Significant design issues but on right track
+ **2 pts** Implemented wrong device

- **0.5 pts** Implemented using half-sub instead of full-sub

+ **0 pts** Wrong/No design

1 Be careful with the order of inputs; good to label them so it's not ambiguous at each block

QUESTION 16

16 Question 4-d **14 / 14**

✓ + **2 pts** **Correct addition function**

✓ + **2 pts** **Correct subtraction function**

✓ + **3 pts** **Correct negation function**

✓ + **2 pts** **Correct multiplication function**

✓ + **3 pts** **MUXs used properly for switching outputs**

✓ + **2 pts** **Completed design/put together properly/correct outputs**

- **0.5 pts** Unclear which wires go where/splitting of wires
- **0.5 pts** Unclear which inputs are which for functions
- **0.5 pts** Wrong codes correspond to functions
- **0.5 pts** Wrong names for adder/subtractor

+ **0 pts** No Design

2 need to add 1

QUESTION 17

17 Question 5 **0 / 5**

+ **5 pts** Correct
+ **4.5 pts** Correct design idea; minor mistake
+ **4 pts** Correct design idea; wrong codes match
+ **3 pts** Correct design idea; not complete/significant mistake
+ **3 pts** Gray-Binary converters wrong
+ **2 pts** Did not handle Gray codes properly/Wrong idea for codes
+ **1 pts** Gray-binary converter only/Fundamentally wrong design

✓ + **0 pts** **No/Wrong design**

Midterm Exam

Name (Last, [redacted]

Student Id #: [redacted]

Student to Le[redacted]

Student to Ri[redacted]

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.

CD \ AB	"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	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	X
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	X
1	1	1	1	0

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

(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

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

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, De'morgans theorem still holds for more than two variables.

Demorgans: $\neg(x \wedge y) = \neg x \vee \neg y$

prove: $\neg(x \wedge y \wedge z) = \neg(D \wedge z) = \neg D \vee \neg z = \neg(x \wedge y) \vee \neg z$

$= \neg x \vee \neg y \vee \neg z$ Proven

call $x \wedge y = D$ plug back in

(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., $A \oplus B = A\bar{B} + \bar{A}B$

Answer: $f = \overline{A\bar{B} + \bar{A}B} + \overline{B\bar{C} + \bar{B}C}$

$= (\overline{A\bar{B}} \wedge \overline{\bar{A}B}) + (\overline{B\bar{C}} \wedge \overline{\bar{B}C})$ Demorgan's

$f = (\bar{A} \vee B) \wedge (A + \bar{B}) + (\bar{B} \vee C) \wedge (B \vee \bar{C})$

$f = \bar{A}\bar{B} \vee AB \vee \bar{B}\bar{C} \vee BC$ Distributive

$f =$ $(\bar{A} \wedge \bar{B}) \vee (A \wedge B) \vee (\bar{B} \wedge \bar{C}) \vee (B \wedge C)$

(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: $(\bar{A} \wedge \bar{B}) \vee (\bar{B} \wedge \bar{C}) \vee (A \wedge B) \vee (B \wedge C)$

$(\bar{A} \vee \bar{C}) \wedge (\bar{A} \vee \bar{B})$ Distributive

$\wedge (\bar{B}) \wedge (\bar{B} \vee \bar{C})$

Absorption

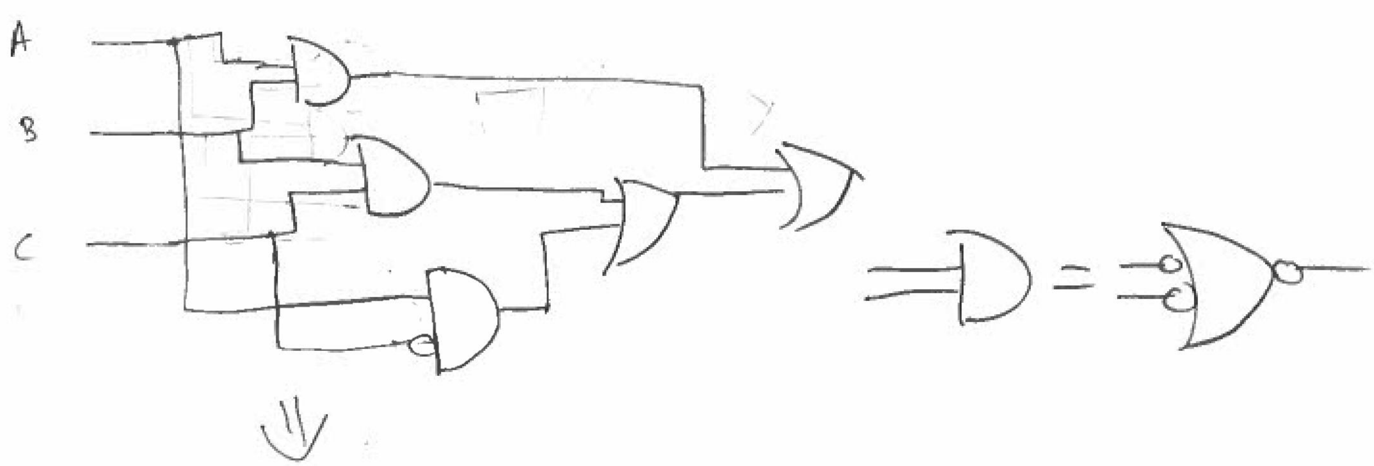
$(\bar{A} \vee \bar{C}) \wedge \bar{B}$ $\vee (\bar{A} \wedge C) \vee \bar{B} \wedge (\bar{B} \vee C)$ Absorption

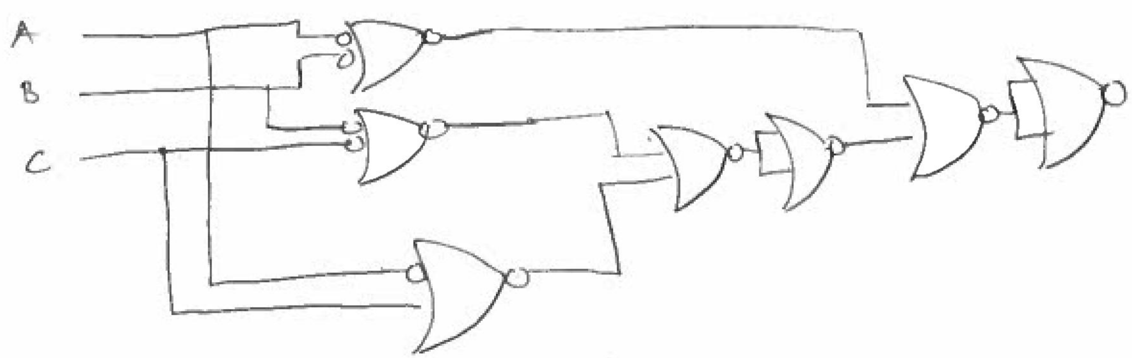
$(A \wedge \bar{C})$

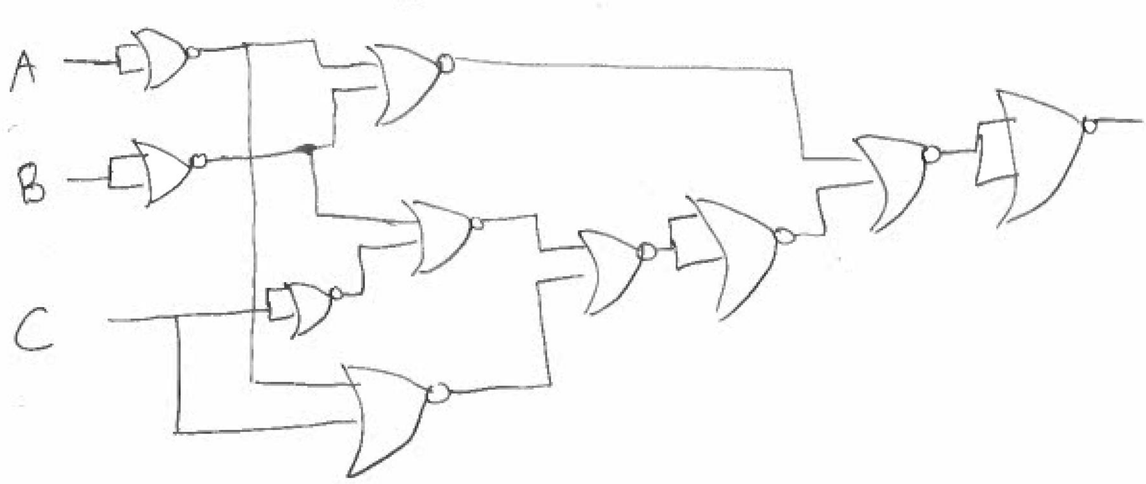
C \ AB	00	01	11	10
0	1	0	1	1
1	1	1	1	0

$f =$ $(A \wedge B) \vee (B \wedge C) \vee (A \wedge \bar{C})$

(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)







10 11 12 13 14 15
A B C D E F

Question #3 (25 pts)

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

12b'1110_0110_1101 −2048 + 1024 + 512 + 64 + 32 + 8 + 4 + 1

(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

(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)

012103

333333
321231
012102
+1

(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

What's the 20's complement representation of -1246 (i.e. the 20's complement of the 1246)? (4 pts) 16.17.14

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

1024
512
64
32
8
4
1
1645

2048
−1645
403

20
20
400

1200
40
6
1246

19 19 19
3 2 6
16.17.13
+1
16.17.14

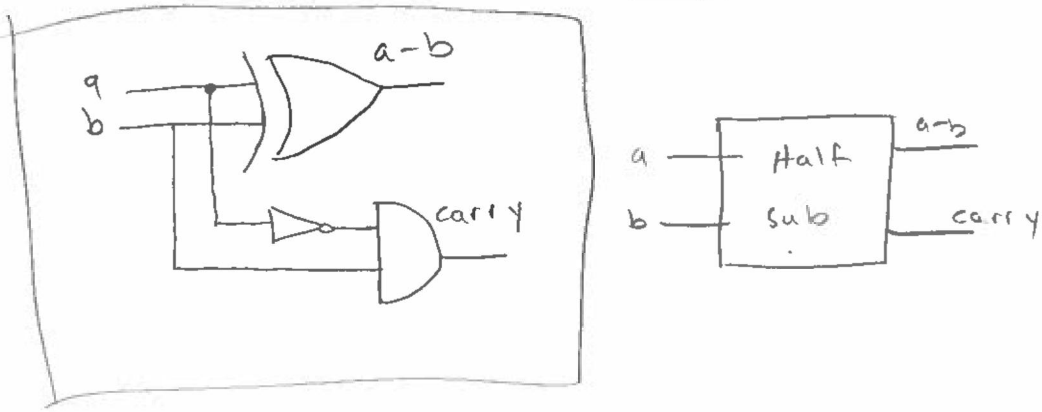
10.0.0

20
20
400

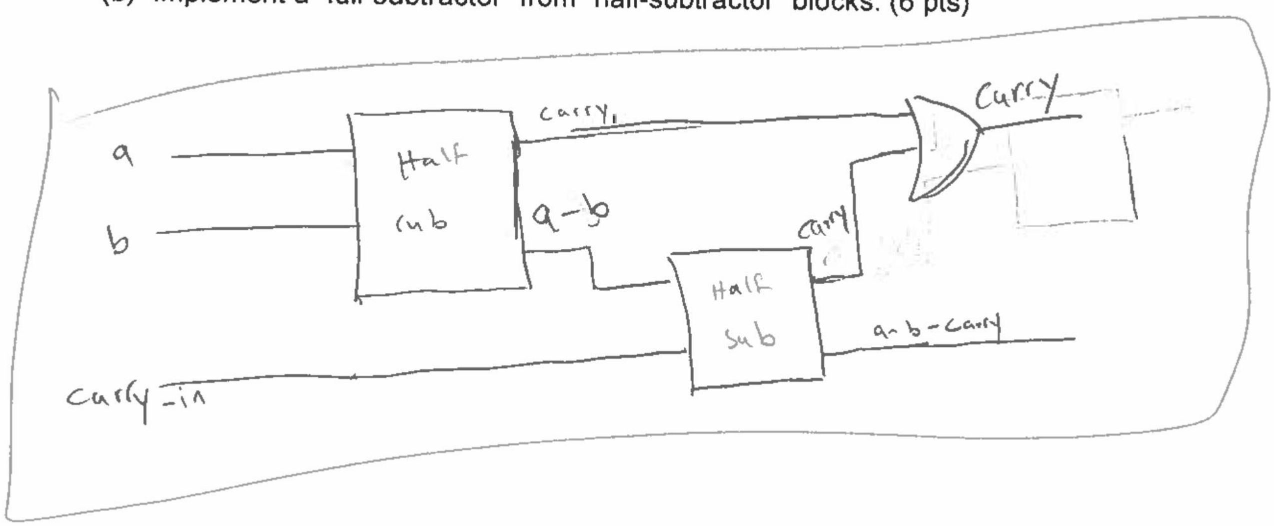
Question #4 (35 pts)

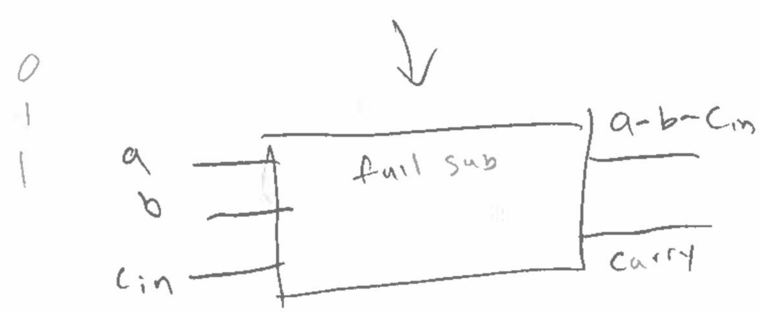
(a) Implement a one-bit "half-subtractor" from gates. The carry-out of this subtractor is 1 when the result is -1. The truth table for this is shown below: (8 pts)

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



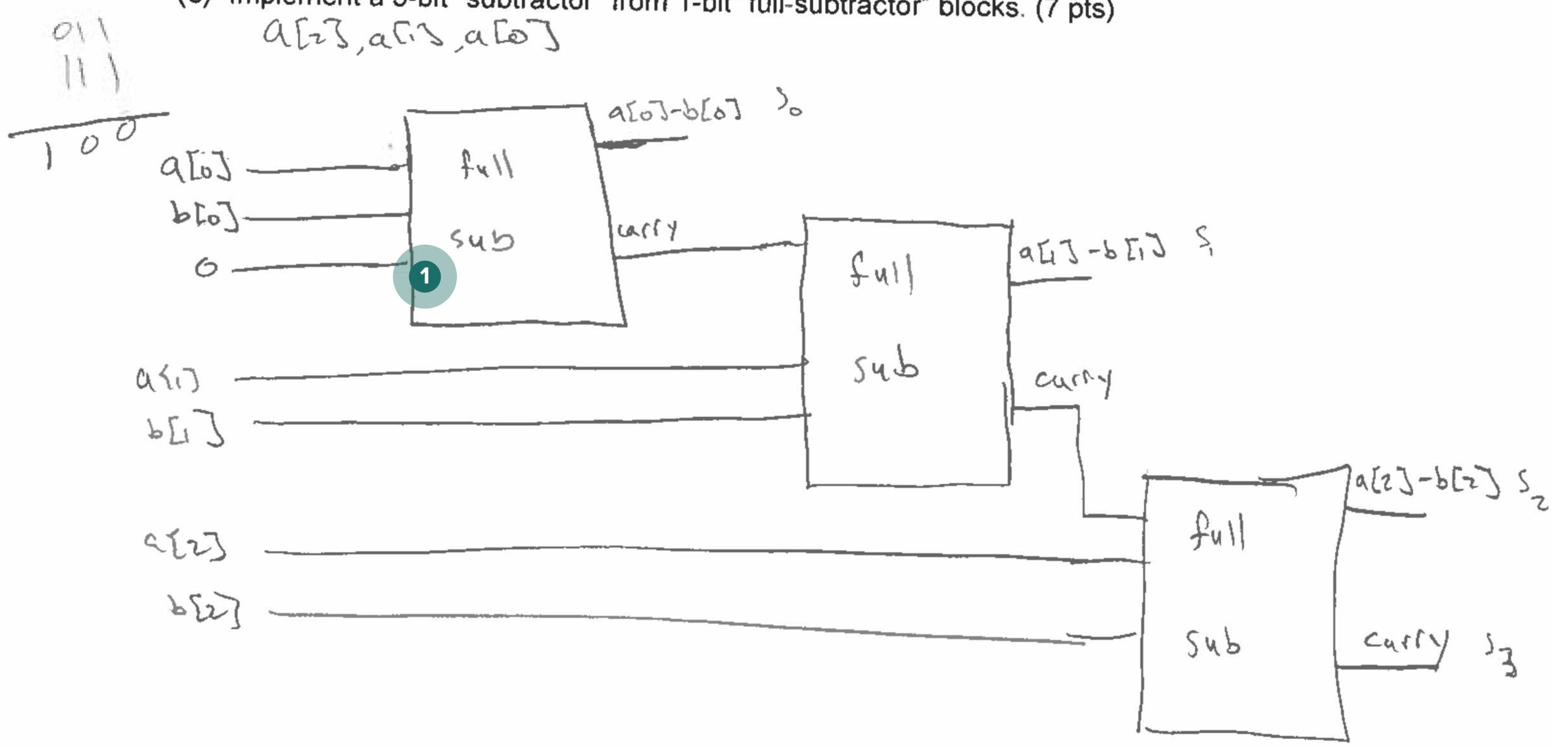
(b) Implement a "full-subtractor" from "half-subtractor" blocks. (6 pts)





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

a[2], a[1], a[0]

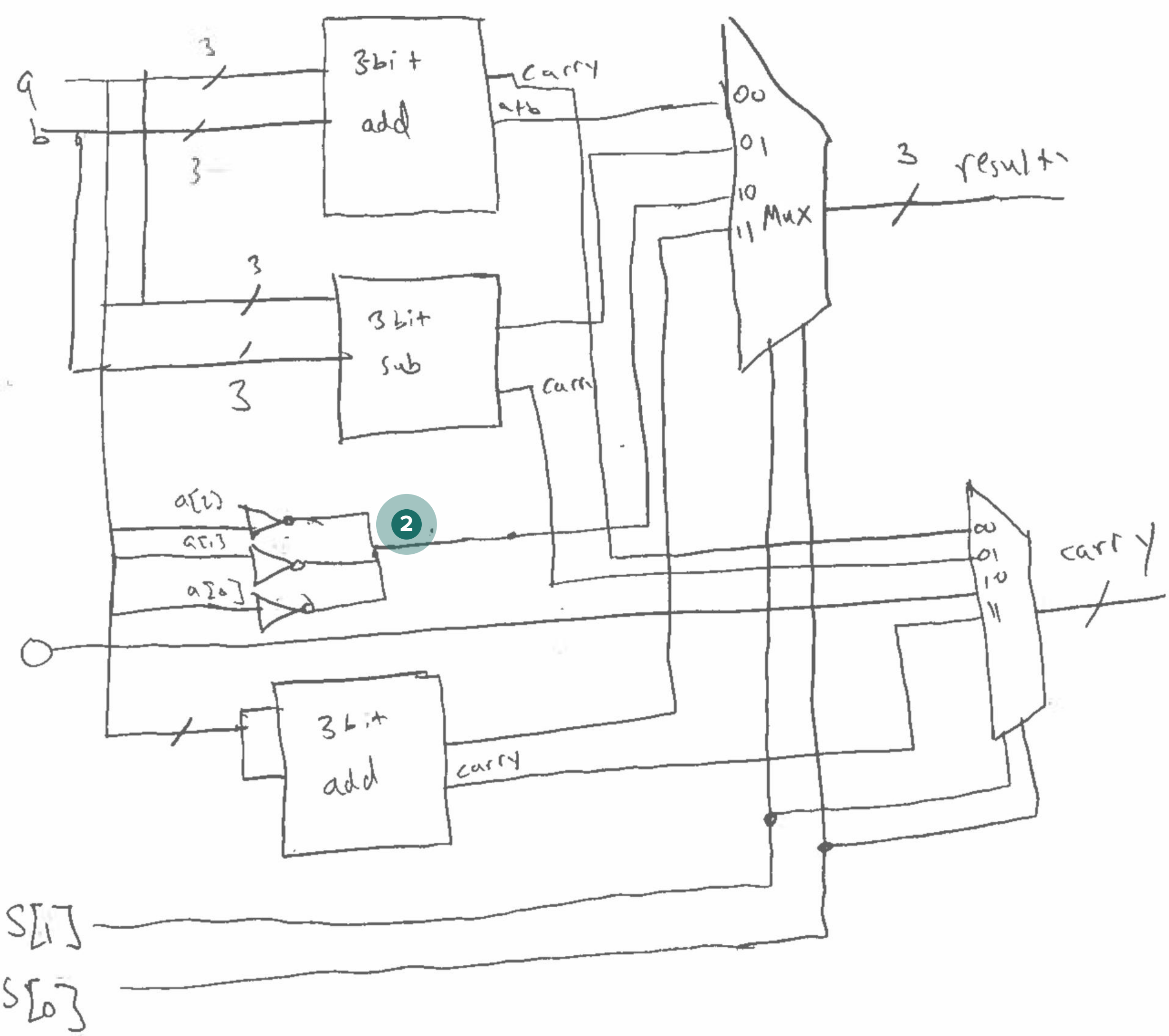


result $= S_3 S_2 S_1 S_0$

(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, implement a 3-bit ALU that can add, subtract, negate one argument, and multiply by 2. 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



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

4 bits