Name: (as it would appear on official course roster)	
UCSB email address: **@ucsb.edu**	**Perm ID Number:**
Lab Section Time:	
Optional: name you wish to be called if different from above	
Optional: name of "homework buddy" (leaving this blank signifies "I worked alone")	

Lab 07: Combinatorial Digital Logic

Assigned: *Thursday, February 27th, 2020*
Due: *Tuesday, March 3rd, 2020*
Points: *80 (normalized to 100)*

- You may collaborate on this homework with AT MOST one person, an optional "homework buddy".
- MAY ONLY BE TURNED ON **GRADESCOPE as a PDF file.**
- There is NO MAKEUP for missed assignments.
- We are strict about enforcing the LATE POLICY for all assignments (see syllabus).

1. In class, we went through the example of splitting a binary addition into single bit additions. This is simplest in terms of design time, but it does not lead to the fastest implementation. Instead, let's create a 2-bit adder as the smallest unit. In this case, there are four bits of input from the two numbers, in addition to the single input carry. There are two result bits and a single output carry. A0 is the least significant bit (the one on the right). inputs: A1, A0, B1, B0, Cin outputs: R0, R1, Cout. For example, if we are performing an addition, it looks like this:

```
      A1 A0
      B1 B0
         Cin
-------------
Cout R1 R0
```

(Question on next page…)

Name:

(as it would appear on official course roster)

a. (8 pts) Fill in this truth table for the function described above

C_{in}	A1	B1	A0	B0	C_{out}	R1	R0

Name:
(as it would appear on official course roster)

2. Given this truth table:

A	**B**	**C**	**D**	**O**
0	0	0	0	**1**
0	0	0	1	**1**
0	0	1	0	**1**
0	0	1	1	**1**
0	1	0	0	**1**
0	1	0	1	**0**
0	1	1	0	**0**
0	1	1	1	**0**
1	0	0	0	**1**
1	0	0	1	**0**
1	0	1	0	**0**
1	0	1	1	**0**
1	1	0	0	**1**
1	1	0	1	**0**
1	1	1	0	**0**
1	1	1	1	**0**

a. (2 pts) Write the <u>unoptimized</u> sum of products equation for the output **O**

b. (4 pts) Draw the K-map, mark it clearly (cleanly and group optimally), and simplify the output function

c. (2 pts) Write the best <u>optimized</u> sum of products equation for the output **O**

Name:

(as it would appear on official course roster)

3. Given this truth table:

A	**B**	**C**	**D**	**O**
0	0	0	0	**0**
0	0	0	1	**0**
0	0	1	0	**0**
0	0	1	1	**0**
0	1	0	0	**1**
0	1	0	1	**1**
0	1	1	0	**0**
0	1	1	1	**0**
1	0	0	0	**0**
1	0	0	1	**0**
1	0	1	0	**0**
1	0	1	1	**0**
1	1	0	0	**1**
1	1	0	1	**1**
1	1	1	0	**0**
1	1	1	1	**1**

a. (2 pts) Write the <u>unoptimized</u> sum of products equation for the output **O**

b. (4 pts) Draw the K-map, mark it clearly (cleanly and group optimally), and simplify the output function

c. (2 pts)Write the best <u>optimized</u> sum of products equation for the output **O**

d. (2 pts) Draw the optimized digital circuit for this function

Name:

(as it would appear on official course roster)

4. For each problem use **Boolean algebra** to simplify the equation. SHOW YOUR WORK, one step per line. You should use only variable names, negation (use `!`), AND (by putting terms next to each other), OR (with `+`), and parentheses. In addition, each line should start with an equal sign (`=`). The first step has been done for you to show how.

   Example: f(Z,Y,X) = !ZY!X + ZY!X + !YX
   = (!Z + Z)Y!X + !YX
   = Y!X + !YX

   a. (6 pts)f(A,B,C) = !A!BC + A!B!C + !ABC + !AB!C + A!BC

   b. (6 pts) (A,B,C,D) = (A!D + !AC)(!B(C + BD))

   c. (6 pts) f(A,B,C,D) = (!A!C + AD)(B(D + !BC))

Name:

(as it would appear on official course roster)

5. Given this truth table (and note the use of don't-cares, represented by **X**):

A	**B**	**C**	**D**	**O**
0	0	0	0	**1**
0	0	0	1	**X**
0	0	1	0	**0**
0	0	1	1	**1**
0	1	0	0	**X**
0	1	0	1	**0**
0	1	1	0	**X**
0	1	1	1	**0**
1	0	0	0	**X**
1	0	0	1	**0**
1	0	1	0	**1**
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**

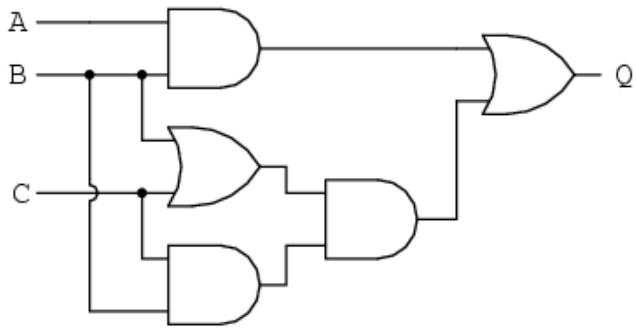
a. (2 pts) Write the unoptimized sum of products equation for the output **O**

b. (4 pts) Draw the K-map, mark it clearly (cleanly and group optimally), and simplify the output function

c. (2 pts) Write the best <u>optimized</u> sum of products equation for the output **O**

Name:
(as it would appear on official course roster)

6. Given the following digital circuit:



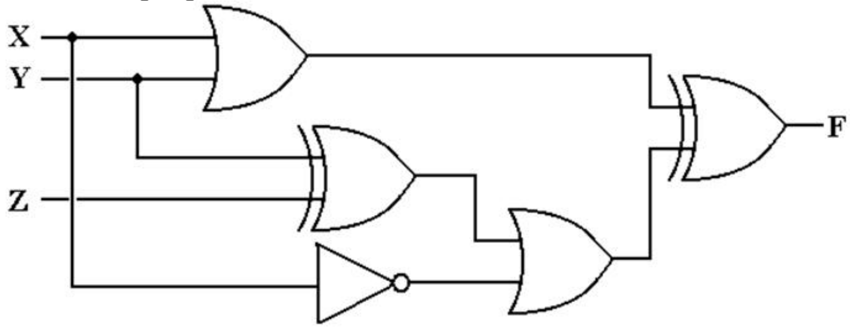
a) (4 pts) Draw the Karnaugh map for this circuit. Show optimal groupings.

b) (2 pts) Write the ***optimized*** logic function that describes the diagram in the form of "sum-of-product".

c) (4 pts) Re-draw the circuit based on your answers in part (b). Bad drawings will lose points!

Name:
(as it would appear on official course roster)

7. Given the following digital circuit:



a) (4 pts) Construct the truth table for this circuit.

b) (4 pts) Using your answer from part (a), write the ***optimized*** logic function that describes the diagram in the form of "sum-of-product". You can use Karnaugh maps (if so, show optimal groupings!) or other techniques, if you like.

Name:

(as it would appear on official course roster)

8. Consider the following C++ function where all variables are single bits:

```
boolean FunctionZ(boolean a, boolean b, boolean c)
{
      boolean x = a & b | c;
      boolean y = ~(x ^ a);
      return y;
}
```

a) (2 pts) Draw the circuit that is a direct representation of this function (without simplifying). Bad drawings will lose points!

b) (3 pts) Use algebraic simplification to simplify this function and write it as a "sum-of-products" format. Start off with:

$$y = \overline{(x \oplus a)}$$

Name:
(as it would appear on official course roster)

c) (5 pts) Given your answer in (b), draw the Karnaugh map for this function output and find an optimal simplification for the function output, **y**. Then, using this, further simplify the function output and prove that: `y = (a & b) + ~(a ^ c)`.