

This is a take-home exam. You are allowed to check books, notes, and conduct an internet search. However, you are expected to work on the solution by yourself. No discussion with anyone else is allowed.

There are two parts to the exam. Part A weighs 90 points. Part B weighs 10 points. However, in Part B, the total weight of the problems is higher than 10 points. You are encouraged to earn up to 10 points. In other words, there is no penalty on wrong solutions in Part B, but the maximum points to earn are limited to 10 points.

Please read the following instructions carefully. If you are unclear about any of the questions on the exam, make the most plausible assumption to answer the question. Instructors and proctors will not answer questions on the exam material. **All the Best!**

Part A: (90 pts)

1 True or False

Check if the following statements are True or False. State your choice and use one sentence to explain your choice. (2 pts each)

1. In Boolean algebra, if $x' + y = 1$, and $x'y = 0$ then $y = (x')'$, where x, y are two Boolean variables.
2. In Boolean algebra, if $xy = 1$, we can derive that $x'y + xy' = x' + y'$, where x, y are two Boolean variables.
3. Truth Table: Given a truth table with input x , the Don't Care set is empty, i.e. $D = \emptyset$. Then, its product of sums expression $f_1(x)$ and sum of products expression $f_2(x)$ are equal, i.e. $f_1(x) = f_2(x)$.
4. Truth Table: For a truth table of n switching variables, we can write 3^n different output patterns.
5. Karnaugh Map: In a Karnaugh map of n switching variables, each cell has $n + 1$ neighboring cells.
6. Universal set: $\{XOR, XNOR\}$ is a universal set.
7. Universal set: $\{f(x, y, z) = x + yz\}$ is a universal set.
8. Universal set: $\{f(x, y, z) = x'y + xy' + xz' + x'z\}$ is a universal set (assuming that constants 0, 1 are available).

2 CMOS Circuits

Design a CMOS gate with function $y(a, b, c, d) = [(a(b + cd))']$. Draw the schematic diagram using NMOS and PMOS transistors as the basic elements. (14 pts)

3 Boolean Algebra

Simplify the following two Boolean functions with a goal to minimize the number of literals and state the laws/theorems in your derivation.

1. $f(a, b, c, d, e) = ab' + ac' + bcde + ade$. (10 pts)
2. $f(a, b, c, d, e) = (a + b')(a + c')(b + c + d + e)(a + d + e)$. (10 pts)

4 Karnaugh Map

4.1 Sum of Products Expressions

Use Karnaugh map to simplify function $f(a, b, c) = \sum m(1, 7) + \sum d(0, 2, 5, 6)$.

Write the K-map and list all possible minimal sum of products expressions. No need for the logic diagram. (10 pts)

4.2 Product of Sums Expressions

Use Karnaugh map to simplify function $f(a,b,c) = \sum m(2,5) + \sum d(1,4,7)$.

Write the K-map and list all possible minimal product of sums expressions. No need for the logic diagram. (10 pts)

5 Other Types of Gates

Consider the function $f(x,y,z)$ where $\oplus$ is an Exclusive OR operator and the priority of the operators is AND (first), Exclusive OR, and OR (last):

$$f(x,y,z) = xy' \oplus x \oplus y \oplus (x' + y') \oplus x'y \oplus z.$$

Simplify the function into a minimal sum of products expression using switching function techniques e.g. Shannon's expansion. Show your derivation. Write the solution in a Boolean expression. (20 pts)

Part B: (10 pts)

In Part B, the total weight of the problems is higher than 10 points. You are encouraged to earn up to 10 points. In other words, there is no penalty on wrong solutions, but the maximum points to earn are limited to 10 points. Note that some problems are open, i.e. exact (perfect) solutions may be challenging to derive. Nonetheless, the principles taught in the class can be used to start the exploration.

B.1. Boolean Algebra: For the definition of Boolean algebra in slide #44 of lecture 1, we make the following changes.

- We replace the identity laws (i.e. $a + 0 = a$, $a \cdot 1 = a$) with annulment laws (i.e. $a + 1 = 1$, $a \cdot 0 = 0$).
- For the complement laws, we add a clause that for every element $a \in B$, its complement a' is unique.

The remaining laws are still intact. Use the new definition to prove the identity laws. State the laws (don't use other theorems) in your derivation. (5 pts)

B.2. Binary Counters: An n -bit counter inputs n bits and outputs $\lceil \ln_2 n \rceil$ (ceiling of $\ln_2 n$) bits to count the number of input bits which are true. For example, a 5-bit counter inputs five bits $(a_4, a_3, a_2, a_1, a_0)$, and outputs three bits (y_2, y_1, y_0) as a binary number to count the number of input bits that are true. Thus, if $(a_4, a_3, a_2, a_1, a_0) = (0, 1, 1, 1, 1)$, then $(y_2, y_1, y_0) = (1, 0, 0)$. If $(a_4, a_3, a_2, a_1, a_0) = (1, 0, 1, 1, 0)$, then $(y_2, y_1, y_0) = (0, 1, 1)$.

Use a minimal number of 5-bit counters to implement a 25-bit counter. Note that constants 0,1, and inverters are available and free. Draw your logic diagram. (5 pts)

B.3. Karnaugh Map: A combinational function $f(a,b,c,d)$ has its On-Set $F = \sum m(3,5,8,9,10,12,14,15)$ and an empty Don't-Care set. Implement the function with {AND, XOR, NOT} gates only (no other gates). Assuming that we can have multiple (more than 2) input gates and NOT gates are free. Design with a minimal networks in terms of the total number of AND, XOR gates. Draw a logic diagram to illustrate the design. (5 pts)

B.4. Switching Function: Given a switching function $f(a,b,c,d,e,f,g,h) = (a+b+c)(a'+b'+d)(a'+d'+h)(c+d'+g')(f'+g'+h)(b+c'+e)(d'+e'+f)(e+f'+g)(a+g'+h)(c'+e+h')(b'+f+h)(d+g+h')$. Find an assignment of the inputs so that function f is true. (5 pts)