University of Washington – Computer Science & Engineering

Winter 2024 Instructor: Justin Hsia 2024-01-30

CSE 369 QUIZ 1

Name: _Molly_Model_____

Student ID Number: _1234567_____

Please do not turn the page until 12:30.

Instructions

- This quiz contains 3 pages, including this cover page. You may use the backs of the pages for scratch work.
- Please clearly indicate (box, circle) your final answer.
- The quiz is closed book and closed notes.
- Please silence and put away all cell phones and other mobile or noise-making devices.
- Remove all hats, headphones, and watches.
- You have 20 (+5) minutes to complete this quiz.

Advice

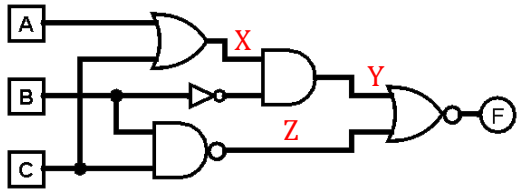
- Read questions carefully before starting. Read *all* questions first and start where you feel the most confident to maximize the use of your time.
- There may be partial credit for incomplete answers; please show your work.
- Relax. You are here to learn.

Question	Points	Score
(1) CL Gates	8	8
(2) K-map	5	5
(3) Waveforms & Verilog	13	13
Total:	**26**	26

Question 1: Combinational Logic Gates [8 pts]

(A) Write out a Boolean expression for the circuit diagram below. *No need to simplify.* Remember to use + (OR), · (AND), and ¯ (NOT) as well as any necessary parentheses to make your answer unambiguous. [2 pts]

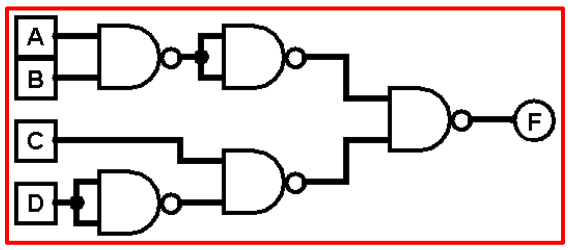
$\mathbf{F = \overline{(A + C) \cdot \overline{B} + \overline{B \cdot C}}}$



$X = A + C$ [0.5 pt]
$Y = X \cdot \overline{B}$ [0.5 pt]
$Z = \overline{B \cdot C}$ [0.5 pt]
$F = \overline{Y + Z}$ [0.5 pt]

(B) Find a minimal implementation of the function below using only **2-input NAND gates**. *We will only accept circuit diagrams.* [6 pts]

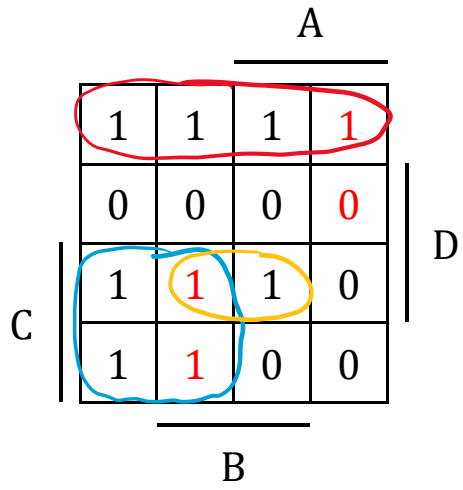
$$F = \overline{AB} + \left(\overline{\overline{C} + D}\right)$$



[2 pt] Valid gate conversion from expression
[2 pt] DeMorgan's applications (either in expression or gates)
[2 pt] Conversion of extra NOTs to NORs

Question 2: Karnaugh Maps [5 pts]

Find the *minimum sum-of-products solution* for the K-map shown below.

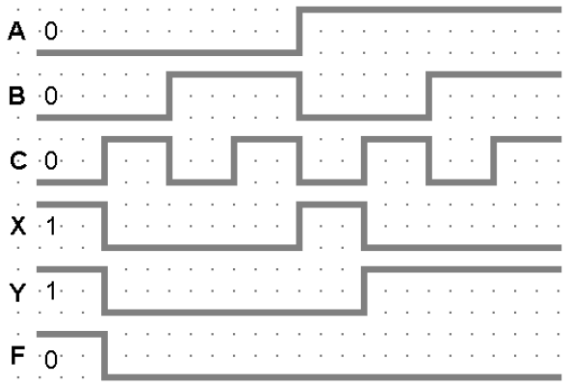


$= \bar{C}\bar{D} + \bar{A}C + BCD$

[2 pt] X choices
[1 pt each] correct term/grouping
[-0.5 pt each] smaller grouping used
[-0.5 pt each] extra grouping included

Question 3: Waveforms & Verilog [13 pts]

(A) Consider the Verilog simulated testbench waveforms shown. Assume all delays are set to 0. If we know that X and Y are outputs of 2-input logic gates, complete the module `Mystery` below. [7 pts]




```
module Mystery (F, A, B, C);
   output logic F;
   input  logic A, B, C;
   logic        X, Y;

   nor G1 (X, B, C);
   // or assign X = ~(B | C);

   xor G2 (Y, A, X);
   // or assign Y = A ^ X;

   and    G3 (F, X, Y);
endmodule
```

(B) A testbench for the `Mystery` module (with inputs `A`, `B`, `C`) is shown below. Which input combinations are NOT currently being tested? You may not need all of the blanks. [3 pts]

```
module Mystery_tb ();
   logic F, A, B, C;

   initial begin
                              #10;  // XXX
      A = 0; B = 0; C = 0;    #10;  // 000
                    C = 1;    #10;  // 001
             B = 1;           #10;  // 011
      A = 1;                  #10;  // 111
                    C = 0;    #10;  // 110
      A = 0;        C = 1;    #10;  // 011
                    C = 0;    #10;  // 010
             B = 0;           #10;  // 000
   end
endmodule
```

Missing combinations:

1. `{A, B, C} = 3'b100;`
2. `{A, B, C} = 3'b101;`
3. `{A, B, C} = 3'b___;`
4. `{A, B, C} = 3'b___;`

(C) Complete the SystemVerilog code below that creates a bus called `my_bus` that is a collection of various DE1-SoC inputs and outputs. Specifically, it should contain the first (*i.e.*, 0th) of each of `SW`, `KEY`, `LEDR`, and `HEX` in that order from most significant to least significant.
Hint: How many bits in a HEX display? [3 pt]

```
_logic [9:0]_ my_bus;

assign my_bus = _{ SW[0], KEY[0], LEDR[0], HEX0 }_____________;
```