University of Toronto

Faculty of Applied Science and Engineering

Department of Electrical and Computer Engineering

Midterm Examination

ECE 241F - Digital Systems
Wednesday October 10, 2007, 6:00 – 7:30 pm

Duration: 90 minutes
Examiners: S Brown, J. Rose and B. Wang

ANSWER ALL QUESTIONS ON THESE SHEETS, USING THE BACK SIDE IF NECESSARY.
1. No calculator and no cellphones are allowed.
2. The number of marks available for each question is indicated in the square brackets [].
3. There are two extra blank pages at the end of the test for rough work.

AID ALLOWED: The Course Textbook, **Fundamentals of Digital Logic with Verilog Design, 1st or 2nd Edition.**

Last Name: **SOLUTIONS**

First Name: _____

Student Number: _____

Please Indicate Which Lab Section You Are In
Monday Morning [ ]
Monday Afternoon [ ]
Tuesday Afternoon [ ]
Friday Morning [ ]

Total Available Marks:

Question	1	2	3	4	5	6	7	8	Total
Marks Available	10	8	10	10	9	10	8	8	73
Marks Achieved									

[10] Q1. In this question you are to determine the programming of an FPGA to implement two logic functions. The logic block of the FPGA, illustrated as a box in the picture on the next page, is a 3-input lookup table, or 3-LUT. A 3-LUT is a special kind of gate that can implement *any* 3-input logic function – that is, it can perform any function possible in a 3 variable truth table. This is a common type of logic gate in modern FPGAs, including the one used in the lab.

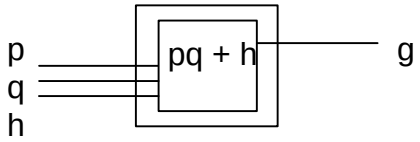
You are to determine the programming information of the FPGA illustrated on the next page to make it implement the following four-input (inputs a, b, c and d) two-output (outputs f1 and f2) function:

f1 = abc + cd

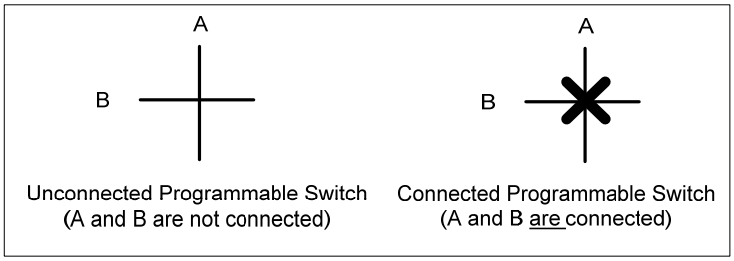
f2 = abc + ad

Your answer should be given by writing on to the picture. We have provided two copies of the picture on the following two pages – use the first one for your rough work, and put your final answer on the second one.

To show what the function of each 3-input lookup table is in your answer, you should write its logic function inside the box. For example, the 3-input lookup table below is "programmed" to implement the function g = pq + h (the picture shows the inputs p, q and h and output g for clarity, but this need not be part of your answer):



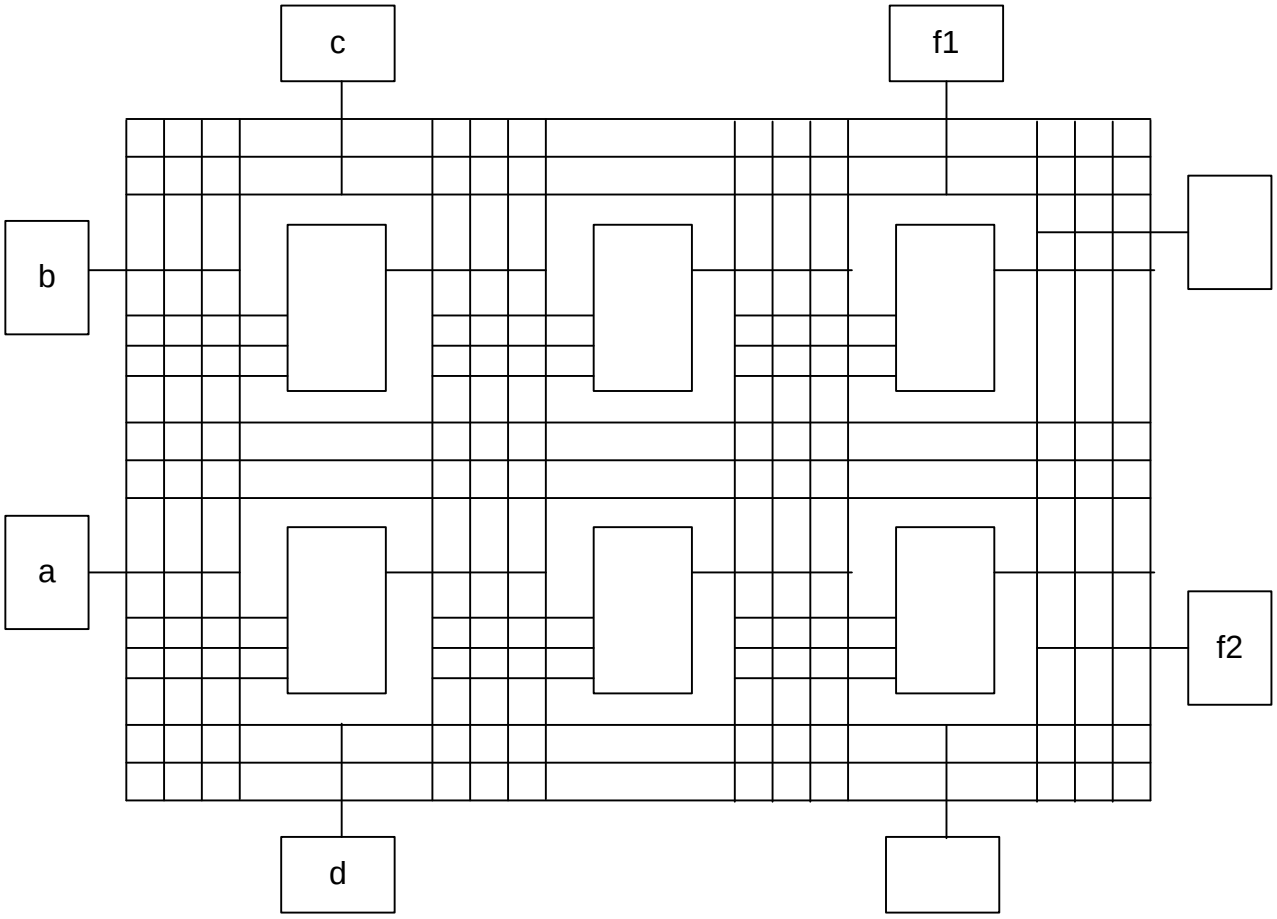
The pictures on the next two pages show the FPGA to be programmed. Assume that every crossing of two wires can be programmably connected with a programmable switch. In your answer, to show that a programmed switch should connect the two crossing wires, draw an X at the intersection of the two wires, as illustrated below:



Note that the inputs (**a, b, c**, and **d**) and outputs (**f1** and **f2**) are already positioned on the pins – **you must use these positions**. Your solution should use the smallest number of three-input lookup tables (3-LUTs) possible. You are **not** required to minimize the number of routing switches used.

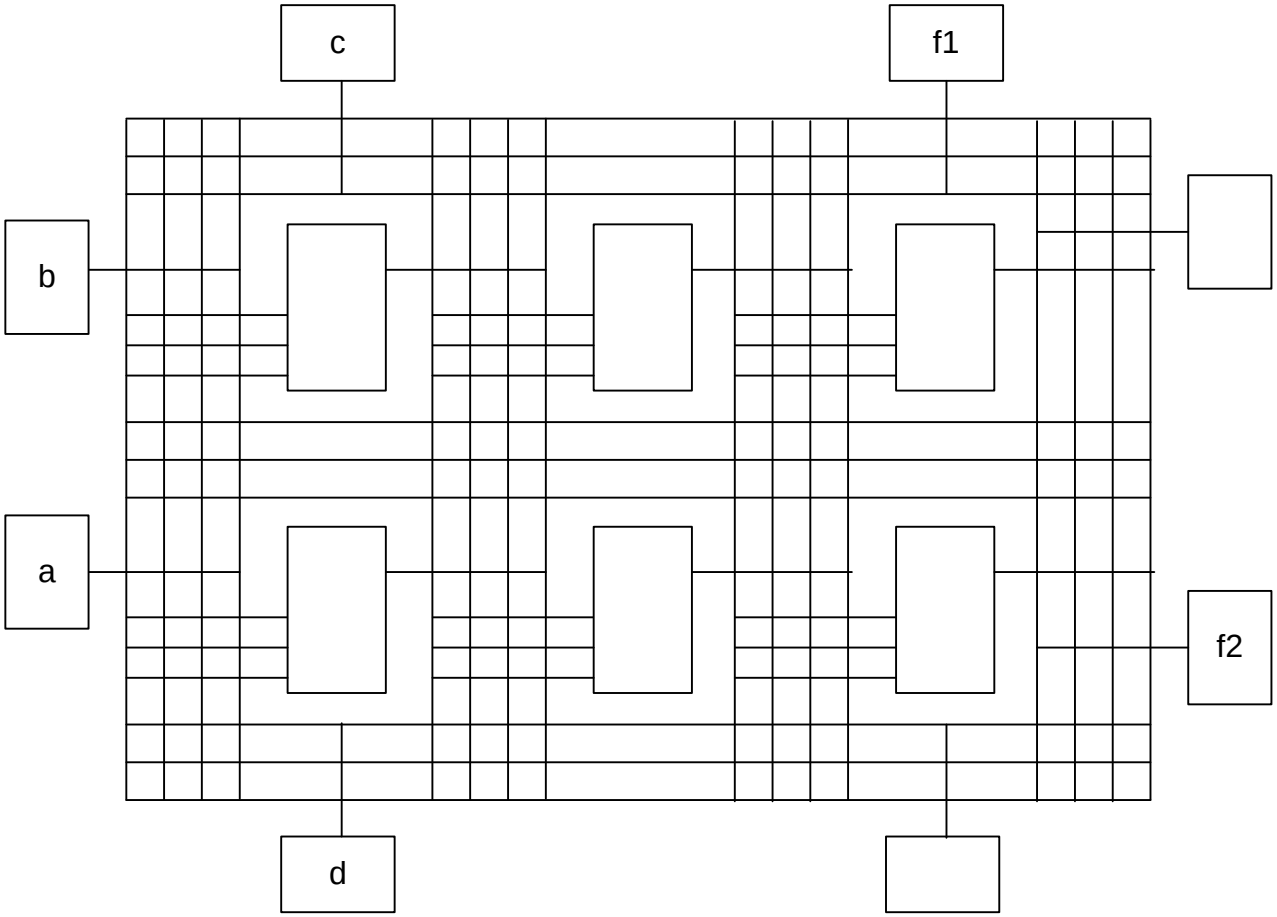
Q1, continued

Place your final answer on this page. There is an exact copy of this picture on the next page that you can use for your rough work. **Be sure to show the programming of both the lookup tables and the programmable routing, as described above.**

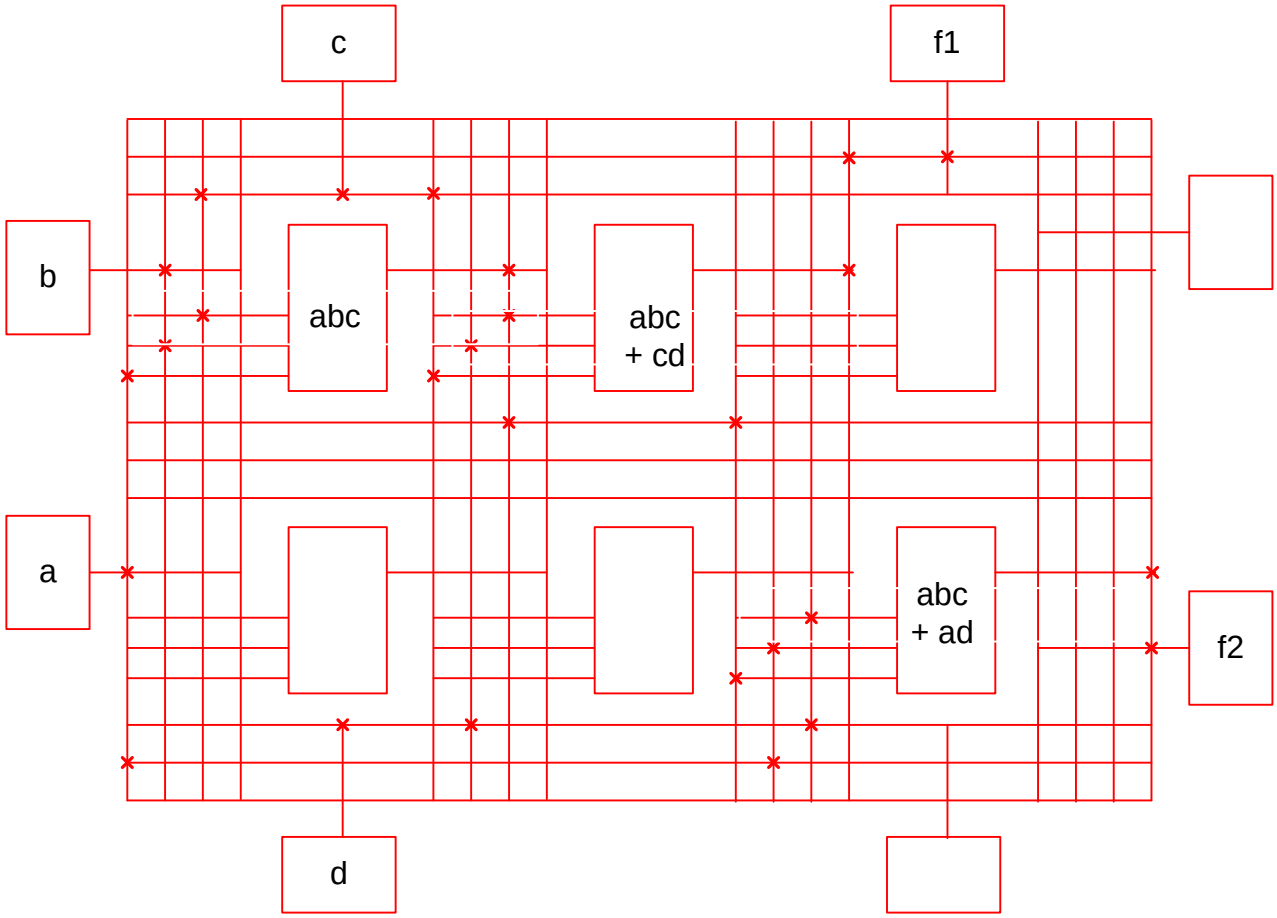


Q1, continued

You can use this picture for your rough work for this question, if you wish.



SOLUTION:



Solution

Marking Scheme:

- For the logic:
  - 2 marks each for the LUT function abc if used just once above.
  - 1 mark for the other 2 logic functions in the 2 LUTs
  - 4 Total
  - -1 for extra LUT used over 3

- For the routing
  - 6 marks for the correct placement and routing (-1 for each error)
  - -2 for not using Xs to indicate wire connection
  - -1 for creating a short circuit

- General
  - Up to 5 marks if only one function was implemented
  - Up to 6 marks if each function was implemented on a separate sheet

- TOTAL 10 Marks

[8] Q2.

A hot air balloon, shown above, is often used as a transportation vehicle. It works by having a gas heater warm the air in the balloon, making the whole vehicle lighter than air, which causes it to rise. When the heater is off, the air in the balloon cools, and causes the balloon to fall. To maintain the balloon at a fixed height, the heater needs to be turned on and off in a controlled manner.

In this question, you will design a digital circuit that turns the heater on and off so as to keep the balloon within two specified height ranges, under the control of a switch, **L**. The inputs to the circuit are three bits $\mathbf{H_2}$, $\mathbf{H_1}$, $\mathbf{H_0}$ that represent the current measured height of the balloon from the ground (determined by a separate altimeter device with these signals as digital outputs), and **L** (which can take on the digital value 0 or 1). The $\mathbf{H_i}$ are coded as given in the table below. When **L** = 0, your circuit should cause the height of the balloon to be kept in the range 201 to 400 metres, and when **L** = 1, the height of the balloon should be kept in the range 501-700 metres.

The output of your circuit, **F**, should be equal to 1 when the heater is to be turned on, and 0 when the heater is to be turned off.

H_2	H_1	H_0	**Balloon Height Range (metres)**
0	0	0	0 - 100
0	0	1	101-200
0	1	0	201-300
0	1	1	301-400
1	0	0	401-500
1	0	1	501-600
1	1	0	601-700
1	1	1	701-800

Answer the questions on the following pages regarding this circuit:

[8] Q2, continued

(a) Determine the Truth Table for the Function F
ANSWER

SOLUTION (Be SURE TO LEAVE ENOUGH SPACE FOR Truth Table when deleting)

L	H_2	H_1	H_0	Balloon Height Range (metres)	Turn Heater On (F)
0	0	0	0	0 - 100	1
0	0	0	1	101-200	1
0	0	1	0	201-300	0, 1 or d
0	0	1	1	301-400	0, 1 or d
0	1	0	0	401-500	0
0	1	0	1	501-600	0
0	1	1	0	601-700	0
0	1	1	1	701-800	0
1	0	0	0	0 - 100	1
1	0	0	1	101-200	1
1	0	1	0	201-300	1
1	0	1	1	301-400	1
1	1	0	0	401-500	1
1	1	0	1	501-600	0, 1 or d
1	1	1	0	601-700	0, 1 or d
1	1	1	1	701-800	0

Marking Scheme:

- 5 marks for a correct truth table. Bonus mark awarded if don't cares are used correctly
- -1 for each incorrect truth table entry

Q2, continued.

(b) Determine the minimal Sum-of-Products expression for F.

ANSWER:

SOLUTION: F = L H2' + H2'H1' + L H1' H0' (WHEN DON'T USE DON'T CARES):

LH2\H1H0	"00"	"01"	"11"	"10"
"00"	1	1	0	0
"01"	0	0	0	0
"11"	1	0	0	0
"10"	1	1	1	1

SOLUTION: F = H2' + L H1' (WHEN USE THE DON'T CARES)

LH2\H1H0	"00"	"01"	"11"	"10"
"00"	1	1	d	D
"01"	0	0	0	0
"11"	1	d	0	D
"10"	1	1	1	1

MARKING SCHEME:

- 3 Marks for correct solution based on the truth table in part (a)
- -0.5 for each non-minimal term

EXCEPTIONS:

- 0 marks if solution simplifies to a constant
- 1 mark if solution simplifies to a single variable, or consists purely of minterms.
- 2 marks if solutions consists of 2 equations – one for L=0 and another for L=1.

[10] Q3. Consider the following logic functions, f and g. They are also represented by the Karnaugh maps shown below (in which "x" denotes a *don't care* value).

$$f(a,b,c,d) = \sum m(0,3,5,7) + \sum d(10,11,12,13,14,15)$$
$$g(a,b,c,d) = \sum m(0,5,6,7,8) + \sum d(10,11,12,13,14,15)$$

c d \ a b	0 0	0 1	1 1	1 0
0 0	1	0	x	0
0 1	0	1	x	0
1 1	1	1	x	x
1 0	0	0	x	x

f

c d \ a b	0 0	0 1	1 1	1 0
0 0	1	0	x	1
0 1	0	1	x	0
1 1	0	1	x	x
1 0	0	1	x	x

g

For each function find the minimum cost logic expression, **in the POS form**. The cost is defined as: cost = # of gates + # of inputs as described in class. Assume that both the complemented and un-complemented input variables are available for free (i.e **do not count the cost of inverters)**.

ANSWER:
SOLUTION:

$$f = \bar{a}\left(b + c + \bar{d}\right)\left(\bar{b} + d\right)\left(\bar{c} + d\right)$$

$$g = \left(\bar{b} + c + d\right)\left(b + \bar{d}\right)\left(b + \bar{c}\right)$$

[4] Marking scheme:

- 2 marks for each correct expression (1 mark for partially correct answer).
- 1 mark each if correct expression, but in SOP form
- -1 if inputs not inverted correctly in POS

Q3, continued.

Now, you are to minimize the *total* cost of implementing *both* functions. That is, you should provide expressions for f and g such that the sum of the costs for f and g is minimized. Hint: consider sharing logic gates between the two functions.

c d \ a b	0 0	0 1	1 1	1 0
0 0	1	0	x	0
0 1	0	1	x	0
1 1	1	1	x	x
1 0	0	0	x	x

f

c d \ a b	0 0	0 1	1 1	1 0
0 0	1	0	x	1
0 1	0	1	x	0
1 1	0	1	x	x
1 0	0	1	x	x

g

i) To do so, first determine the logic expressions for f and g to achieve minimum total cost.
ANSWER:

f =

g =

SOLUTION:

$$f = (\bar{b} + c + d)(b + c + \bar{d})(\bar{c} + d)\bar{a}$$

$$g = (\bar{b} + c + d)(b + c + \bar{d})(b + \bar{c})$$

[4] Marking scheme: 2 marks for each correct expression (1 mark for partially correct answers)
4 marks for both correct SOP expressions (1-2 marks for partially correct answers)
-1 for not inverting inputs correctly

$$f = bd + \bar{a} \bullet \bar{b} \bullet \bar{c} \bullet \bar{d} + cd$$

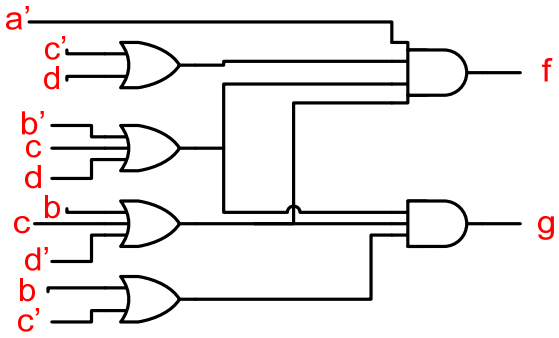
$$g = bd + \bar{a} \bullet \bar{b} \bullet \bar{c} \bullet \bar{d} + bc + a\bar{d}$$

Q3, continued.

ii) Draw the minimum total cost two-level logic circuit using AND and OR gates. Assume that the complemented and un-complemented form of all variables are available as inputs.

ANSWER:

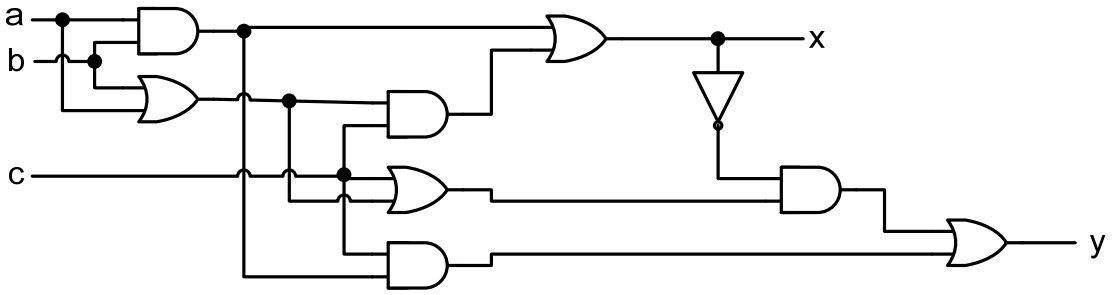
SOLUTION:



[2] Marking scheme: 2 marks for the correct answer
1 mark for partially correct answer
-1 if more than 2 levels of logic

Full marks for correct translation from the answer given in part i)

[10] Q4. Consider the following digital circuit:



a) Determine the truth table of this circuit and put it into the table below:

ANSWER:

a	**b**	**c**	**x**	**y**
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

SOLUTION:

$$x = ab + (a+b)c$$

$$y = \bar{x}(a+b+c) + abc$$

a	**b**	**c**	**x**	**y**
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

[4] Marking scheme:
2 marks for correct x entries (1 mark for partially correct list)
2 marks for correct y entries (1 mark for partially correct list)
If x is wrong and y is correct for incorrect x then 2 marks

b) Determine the logic expression for y in the simplest product-of-sum (POS) form.

ANSWER:

y =

SOLUTION:

c \ a b	0 0	0 1	1 1	1 0
0	0	1	0	1
1	1	0	1	0

$$y = (a+b+c)(\bar{a}+\bar{b}+c)(a+\bar{b}+\bar{c})(\bar{a}+b+\bar{c})$$

[2] Marking scheme: 1 mark for partially correct expression or not in POS form

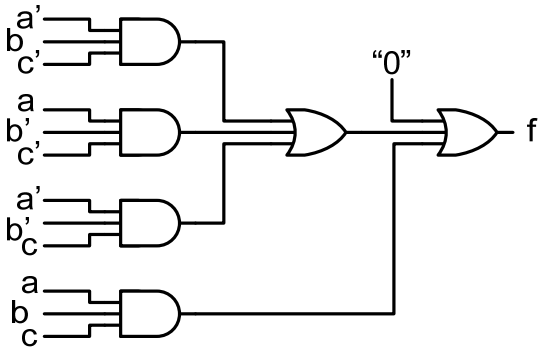
Q4, continued.

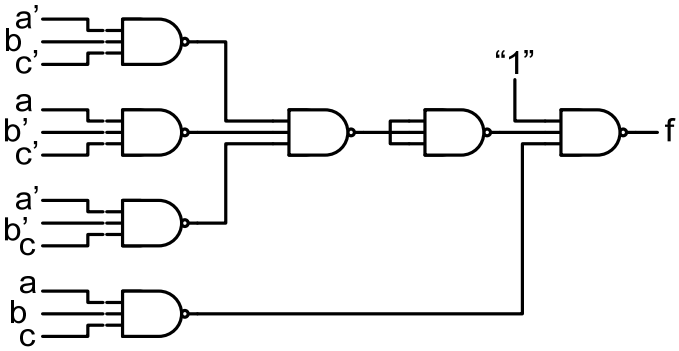
c) Consider the following logic function: $f = \bar{a} \bullet b \bullet \bar{c} + a \bullet \bar{b} \bullet \bar{c} + \bar{a} \bullet \bar{b} \bullet c + a \bullet b \bullet c$ Draw the circuit that implements this function using as few 3-input NAND gates as possible (you can *only* use 3-input NAND gates) Assume that you are allowed to connect inputs of the gates to logic 1 and 0 if necessary, and that both the variables and their complements are available for all input signals.

d)

ANSWER:

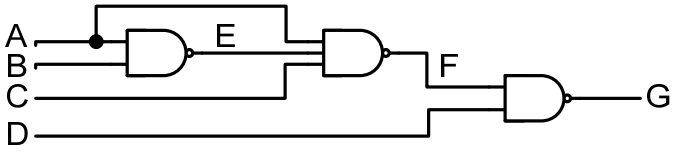
SOLUTION:



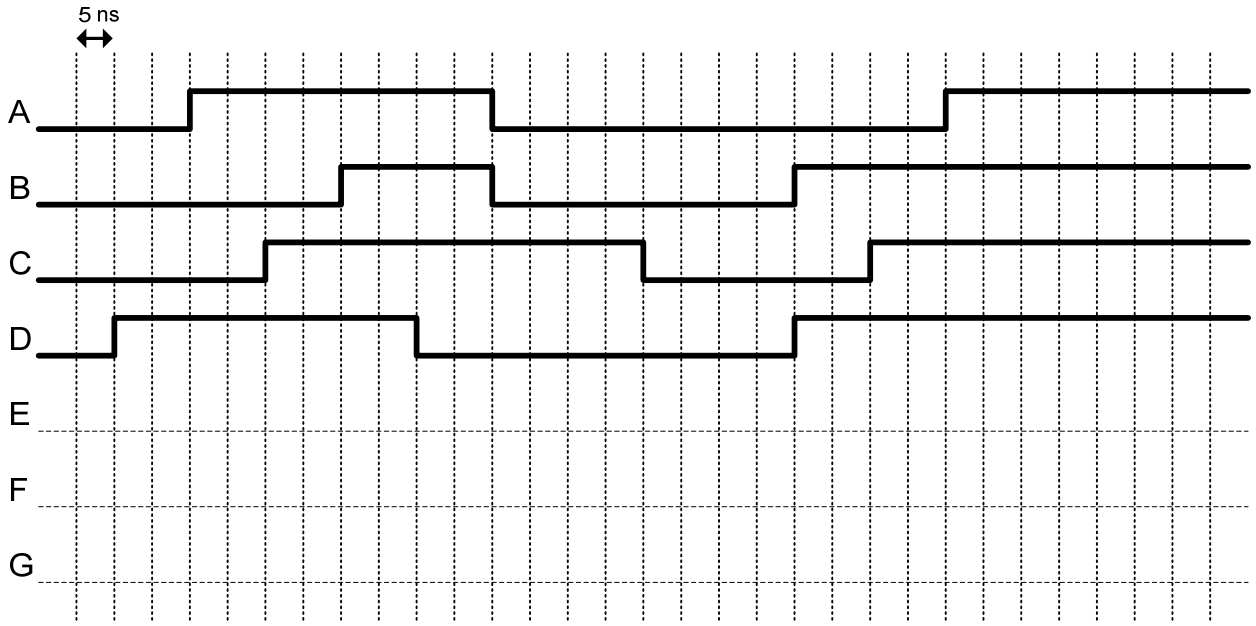


[4] Marking scheme: -1 mark for each gate other than 3-input NAND
-1 mark for each extra gate (more than 7)
-0.5 if gate input is incorrect

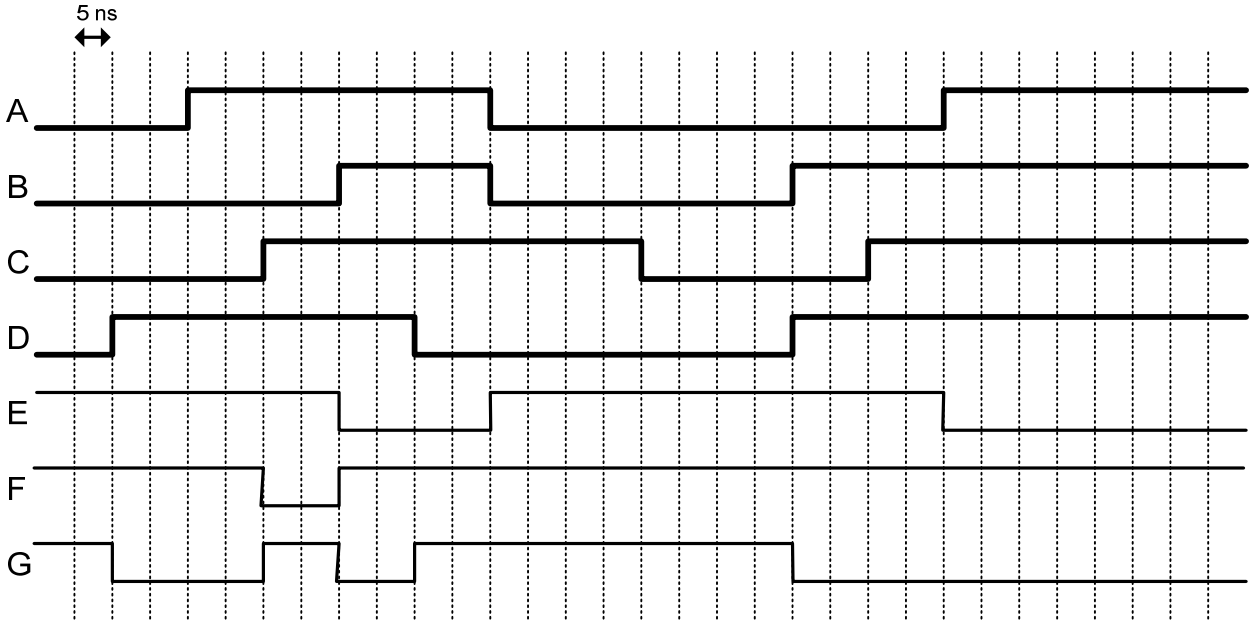
[9] Q5. Consider the following circuit:



a) Draw the <u>functional</u> simulation of wave forms, E, F and G.
ANSWER:



SOLUTION:



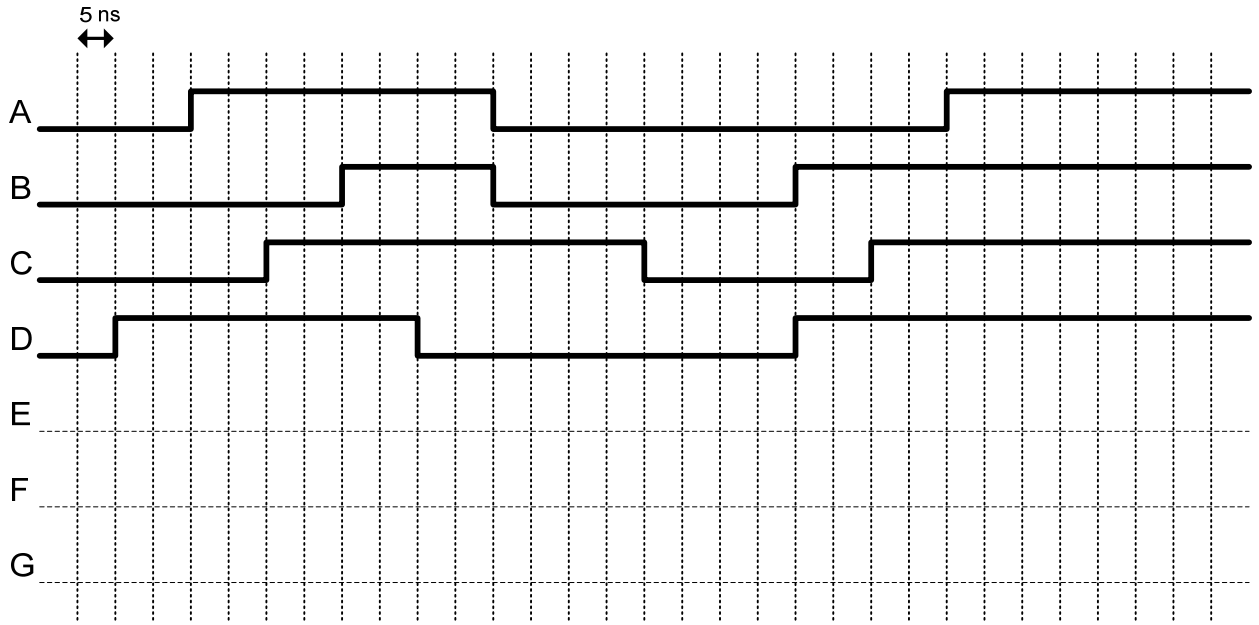
[3] Marking scheme: 1 mark for each correct trace

Q5, continued.

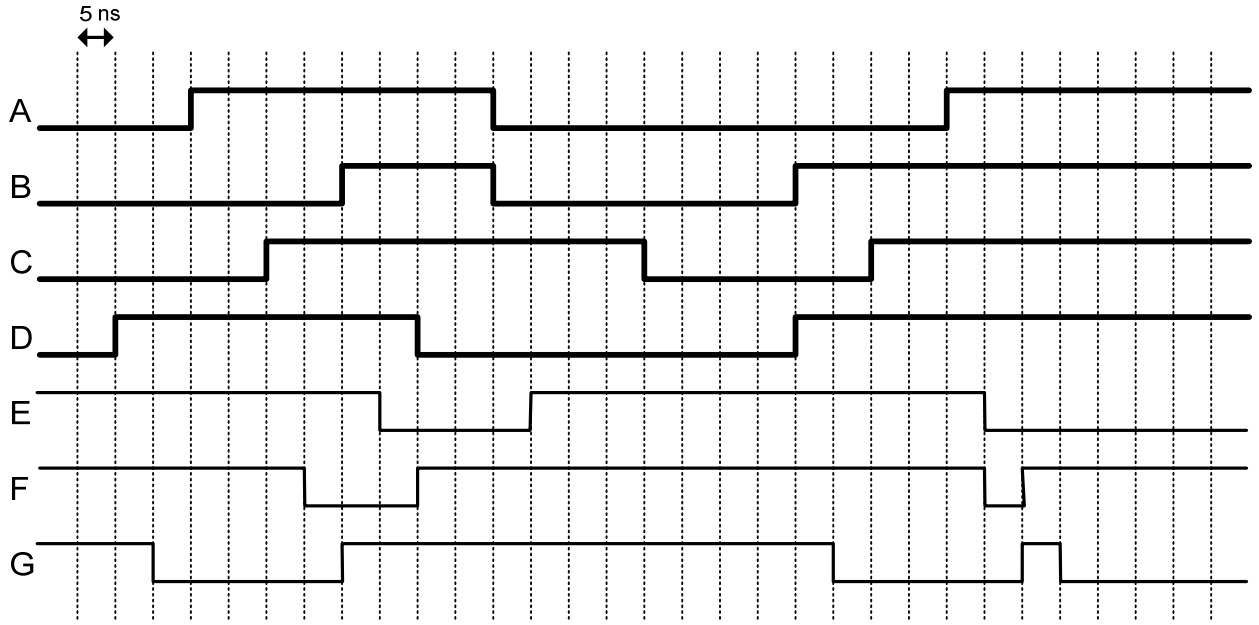
b) Now, assume that each gate has a gate delay of 5 ns, draw the *timing* simulation wave forms, E, F and G.

c)

ANSWER:



SOLUTION:



[6] Marking scheme: 2 marks for each correct trace

[10] Q6.

Bob wishes to express to Alice what ingredients would represent a good Thanksgiving dinner. Bob expresses his opinion in several ways, shown below. Your task is to identify a single group of these expressions that have the same logical meaning. Place an **X** beside each of the equivalent expressions.

____ Thanksgiving dinner will be good if it (has potatoes but does not have rice) or if it (has rice but does not have turkey) or it (has fish but does not have turkey)

___ Thanksgiving dinner will be good if it has turkey or fish

____ Thanksgiving dinner will be good if it (has turkey but does not have fish) or if it (has fish but does not have turkey) or it (has both turkey and fish)

____ Thanksgiving dinner will be good if it (has rice and fish) or (has turkey but does not have fish) or if it (has fish but does not have rice)

____ Thanksgiving dinner will be good if it (has potatoes and turkey but does not have fish) or if it (has turkey but does not have potatoes) or it (has both fish and potatoes) or it (has fish but does not have turkey)

____ Thanksgiving dinner will be good if it (has both fish and rice) or if it (has fish but does not have turkey) or it (has turkey but does not have potatoes) or it (has turkey but does not have fish) or it (has all of turkey, fish, and potatoes)

____ Thanksgiving dinner will be good if it (does not have fish and does not have potatoes) or if it (has both rice and turkey) or it (has both rice and potatoes)

SPACE FOR ROUGH WORK IS GIVEN BELOW AND ON THE NEXT PAGE

Q6, continued. SPACE FOR ROUGH WORK

Q6. SOLUTION

____ Thanksgiving dinner will be good if it (has potatoes but does not have rice) or if it (has rice but does not have turkey) or it (has fish but does not have turkey)

__**X**__ Thanksgiving dinner will be good if it has turkey or fish

__**X**__ Thanksgiving dinner will be good if it (has turkey but does not have fish) or if it (has fish but does not have turkey) or it (has both turkey and fish)

__**X**__ Thanksgiving dinner will be good if it (has rice and fish) or (has turkey but does not have fish) or if it (has fish but does not have rice)

__**X**__ Thanksgiving dinner will be good if it (has potatoes and turkey but does not have fish) or if it (has turkey but does not have potatoes) or it (both fish and potatoes) or it (has fish but does not have turkey)

__**X**__ Thanksgiving dinner will be good if it (has both fish and rice) or if it (has fish but does not have turkey) or it (has turkey but does not have potatoes) or it (has turkey but does not have fish) or it (has all of turkey, fish, and potatoes)

____ Thanksgiving dinner will be good if it (does not have fish and does not have potatoes) or if it (has both rice and turkey) or it (has both rice and potatoes)

Marking Scheme:

2 for each correct answer to a maximum of 10.

Exceptions:

If first or last checked then maximum mark is 8.
If both first and last checked then maximum mark is 6

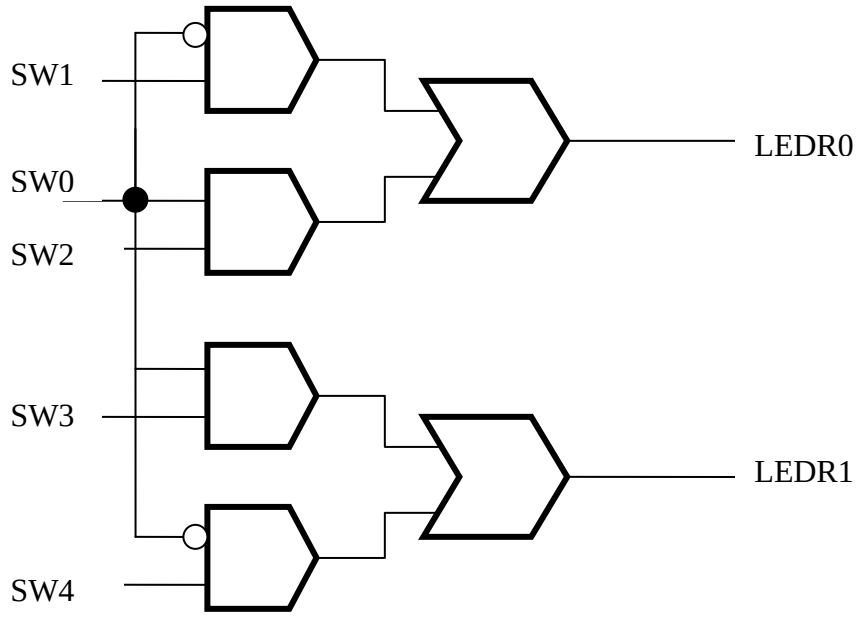
[8] Q7.

Draw a schematic diagram using only AND, OR, and NOT gates that corresponds to the Verilog code given below. Show the inputs SW and outputs LEDR on your schematic.

```
module ver_question (input [0:4] SW, output [0:1] LEDR);

    wire c, x1, x2, y1, y2, f, g;

    assign c = SW[0];
    assign x1 = SW[1];
    assign x2 = SW[2];
    assign y1 = SW[3];
    assign y2 = SW[4];

    subcircuit U1 (c, x1, x2, f);
    subcircuit U2 (~c, y1, y2, g);

    assign LEDR[0] = f;
    assign LEDR[1] = g;

endmodule

module subcircuit (input c, x, y, output f);

    assign f = (!c & x) | (c & y);

endmodule
```

GIVE SCHEMATIC HERE:

SOLUTION



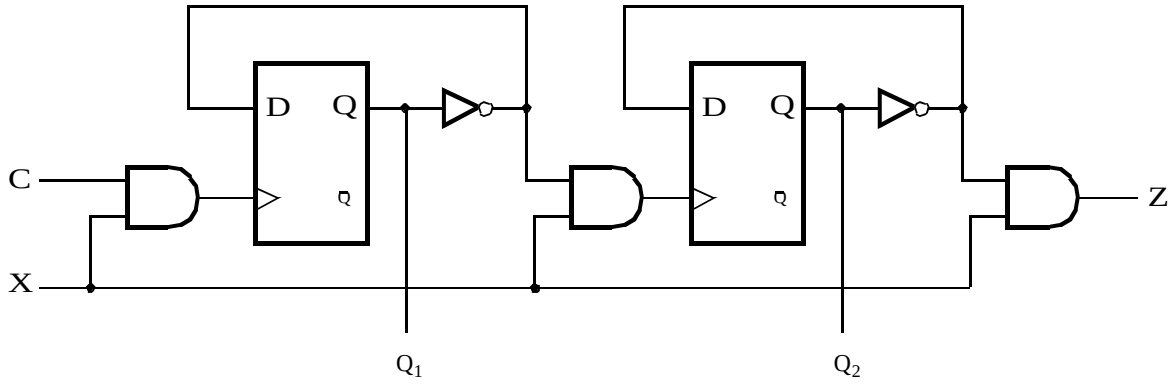
Marking Scheme:

2 for gates in top half, 2 for gates in bottom half, 2 for labels for SW, 2 for labels for LEDR

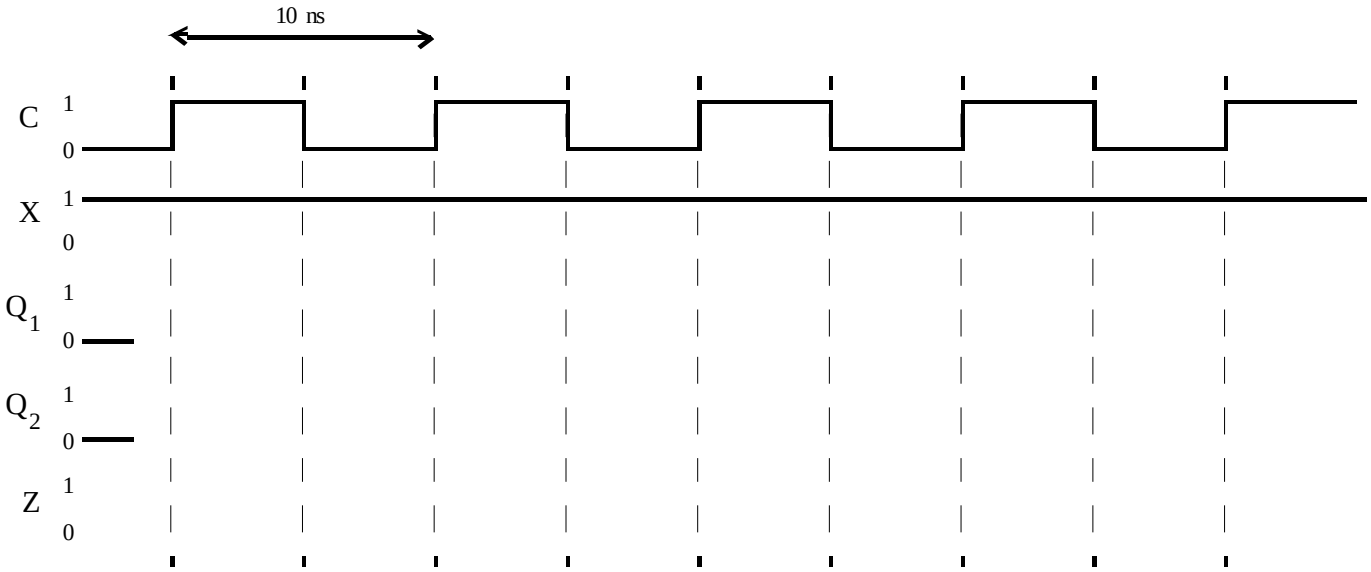
If they use multiplexers, -2

SOLUTIONS SOLUTIONS SOLUTIONS SOLUTIONS SOLUTIONS SOLUTIONS

[8] Q8. Consider the circuit shown below:

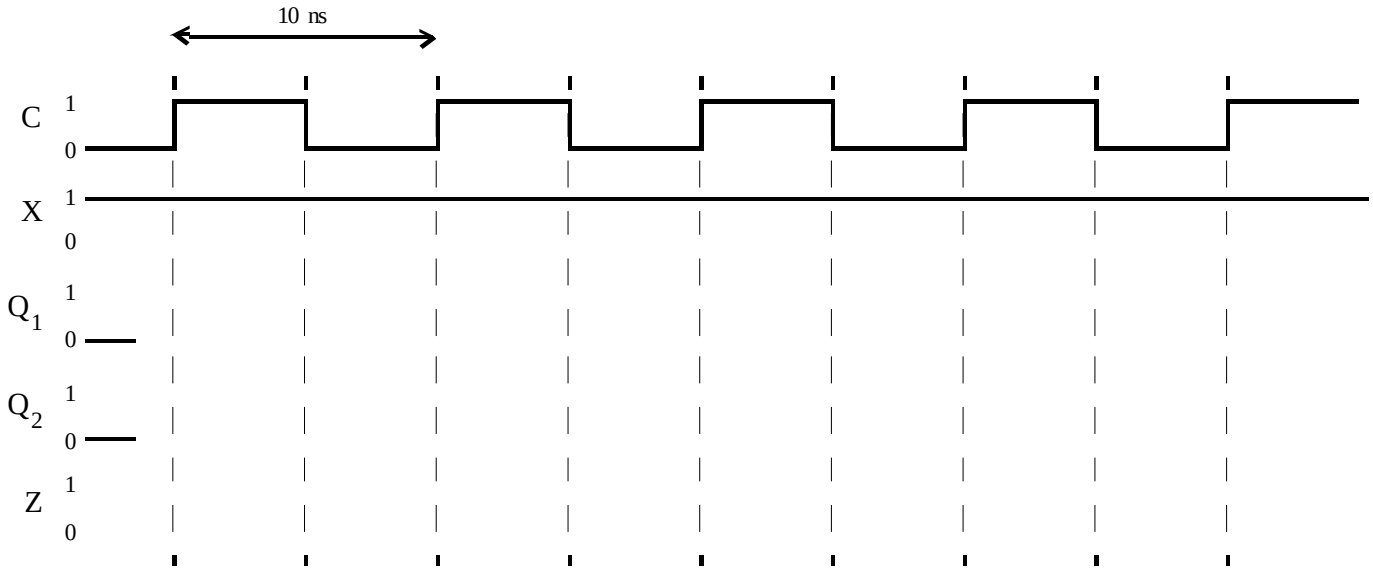


(a) [4] Assuming zero delay time through both logic gates and flip-flops, complete the timing diagram shown below. Note that we assume both flip-flop outputs are 0 to start.



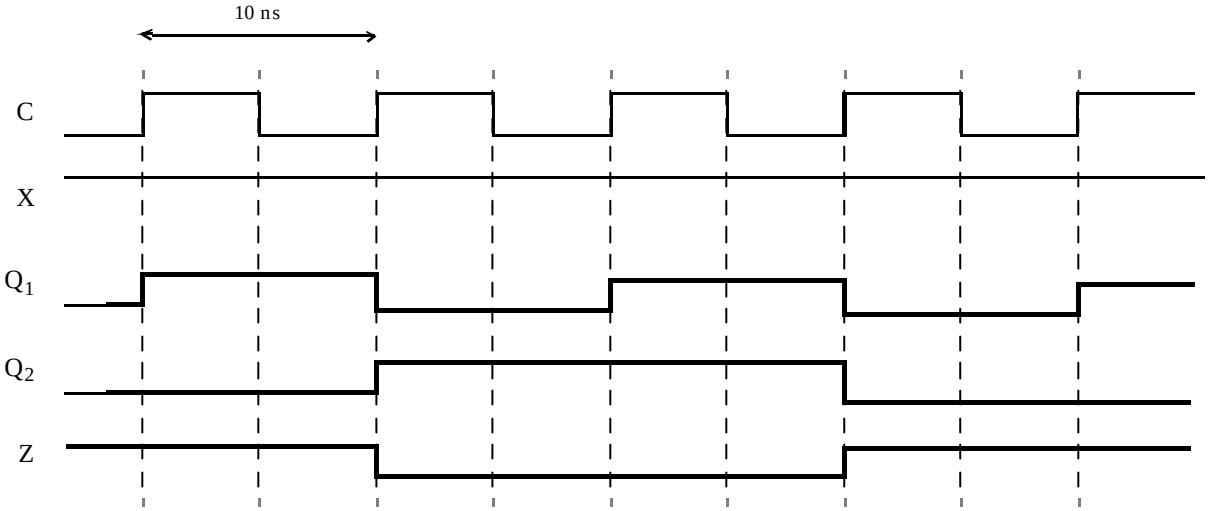
Q8, continued.

(b) [4] Assuming a flip-flop clock-to-Q delay of 3 ns, an AND-gate delay of 2 ns, and a NOT gate delay of 1 ns, complete the timing diagram below. Note that we assume both flip-flop outputs are 0 to start.

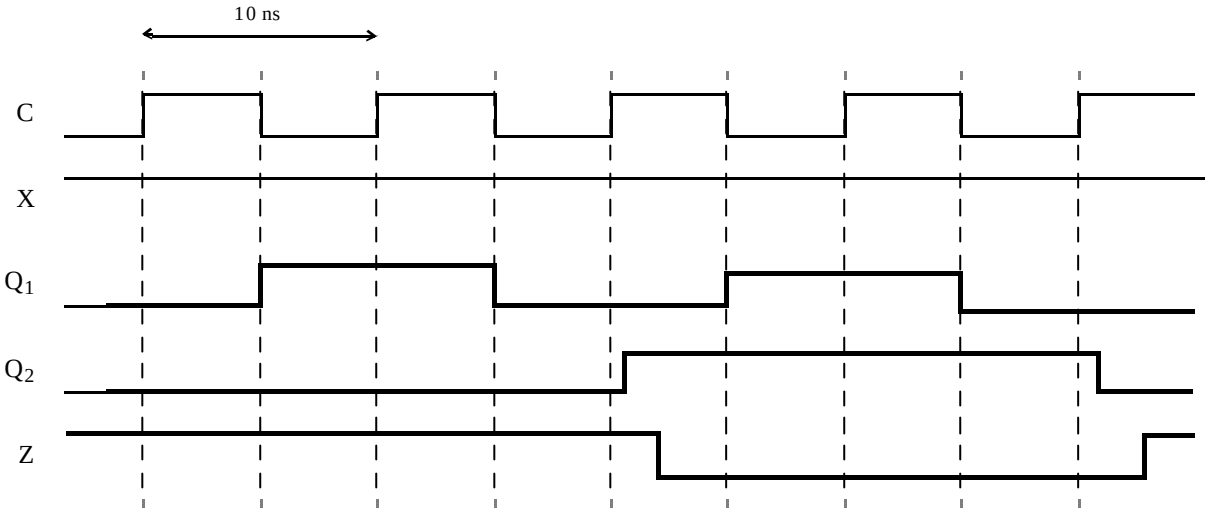


SOLUTIONS SOLUTIONS SOLUTIONS SOLUTIONS SOLUTIONS SOLUTIONS
SOLUTIONS

(a) Assuming zero delay time through both logic gates and flip-flops, complete the timing diagram shown below.



(b) Assuming a flip-flop clock-to-Q delay of 3 ns, an AND-gate delay of 2 ns, and a NOT gate delay of 1 ns, complete the timing diagram below.



Marking Scheme:

+1.5 for Q1
+1.5 if Q2 is correct based on Q1 (which may be incorrect)
+1 if Z is correct based on Q2 (which may be incorrect)