Midterm Examination [Duration: 1 hr. 50 min.]

Your Name: *Solutions* Student Id #:

Do not start working on the exam until you are instructed to do so. Please make sure to read the following instructions carefully and in entirety:

1. *You must answer in the space provided for answers after every question. We will ignore answers written anywhere else in the booklet.*
2. *You may use one cheat sheet of a two-sided 8.5"x11" page.*
3. *You may not use any electronic device.*
4. *You must show the intermediate steps in deriving your answer.*
5. *f any sheets are missing from this booklet, you will get zero grade on the entire exam.*

Following table to be filled by course staff only

Problem #	Maximum Score	Your Score	Comments
1	20		
2	20		
3	15		
4	20		
TOTAL	75		

EXTREMELY IMPORTANT

Please do not tear off or remove any pages from this exam booklet. You must <u>return all the pages</u> of this booklet, and failure to do so will void your exam and fetch you zero score for the entire exam.

Problem #1: Quickly Testing the Basics [2*10 = 20 points]

Important Note: each question is worth +2 points for correct answer, and -1 points for incorrect or blank answers. Minimum total score is 0. Only write answers in the boxes, and use the extra sheet at the end of the question for any scratch work.

A. The 5-bit 2's complement number representation of -7 is:

11001

B. 110110, when treated as a 6-bit signed-magnitude number, has a decimal representation of:

- 22

C. You are treating the 8-bit numbers A[7:0] and B[7:0] as unsigned numbers. If you set B[4:0]=A[7:3] and B[7:5]=0, what is the numeric value of B as a function of A?

B = A/8 if / is viewed as integer division, or floor(A/8) if / is real division.

D. What is the maximum number of product terms in a minimal sum-of-products expression with three variables? (*Hint:* think what pattern of 0s and 1s should the K-map have)

Four. The most complicated K-map that doesn't permit simplification is a checkerboard of 1's where 50% of the K-map cells can be 1. So in a 3-input K-map, there are eight cells of which four can be 1 in a checkerboard pattern. Each 1 in the checkboard corresponds to a separate product term since it can't be combined with it's neighbors.

E. What is the minimum number of 2-input NAND gates that would suffice for you to be able to build an implementation of any arbitrary 2-input boolean function?

Five. A function with 2 inputs can be simplified down to just 2 product terms. Of the gates three could be arranged in a tree, computing the sum-of-products of 2 product terms. The other two could be used as inverters to produce negations of the inputs.

F. True or false: A boolean function of N variables with greater than 2^N-1 product terms can always be simplified to an expression using fewer product terms.

True. If there are more than 2^{N-1} product terms, the K-map of the function would be more than half full and there would be 1's in at least two adjacent cells, leading to a possible simplification.

G. The hexadecimal representation for an 8-bit two's complement number is 0xD6. What is its decimal representation?

-42

H. Consider a 3-input combinational device called a *minority gate*. Its output has the logic value that appears at the minority of its inputs: the output will be 0 if 2 or more of the inputs are 1, or 1 if two or more of the inputs are 0. Is the minority gate universal?

Yes. If A, B, and C are inputs and Z the output then a minority gate performs the function $Z = \neg A \neg B \vee \neg B \neg C \vee \neg C \neg A$. If we set $C = 1$ then $Z_{C=1} = \neg A \neg B = \neg(A \vee B)$, which is a NOR2 gate and is known to be universal.

I. An FSM, M, is constructed by connecting the output of a 3-state FSM to the inputs of an 9-state FSM. M is then reimplemented using a state register with the minimum number of bits. What is the maximum number of bits that may be needed to reimplement M?

M has 27 states which require a total of 5 bits in the state register.

J. You connect M N-state FSMs, each have 1 input and 1 output, in series. What's an upper bound on the number of states in the resulting FSM?

Each FSM can in theory be in one of its N states, so an upper bound on the number of states in the combined machine is N^M.

Space for Scratch Work for Problem 1

Problem #2: Combinational Logic Fundamental [4*5 = 20 points]

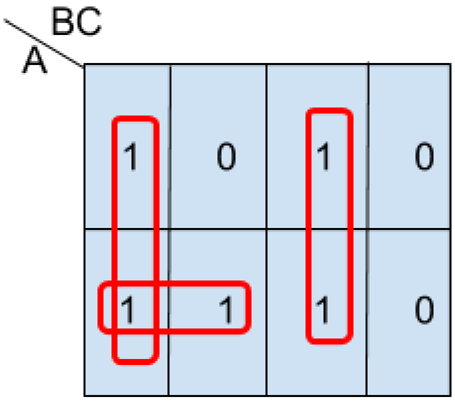
Consider the boolean function defined by the truth table below where A, B, and C are inputs, and F is the sole output.

A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

A. Write a minimal sum-of-product expression for F.

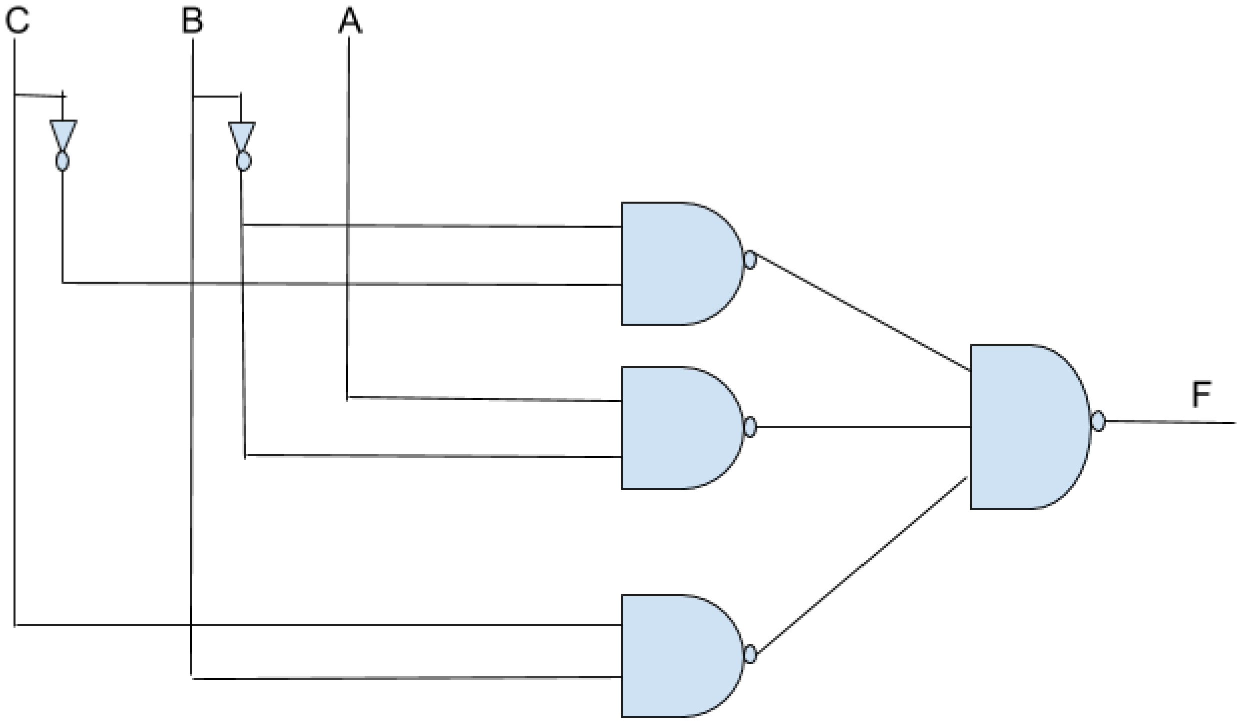
F = $(\neg B \wedge \neg C) \vee (A \wedge \neg B) \vee (C \wedge B)$

Show supporting work below:

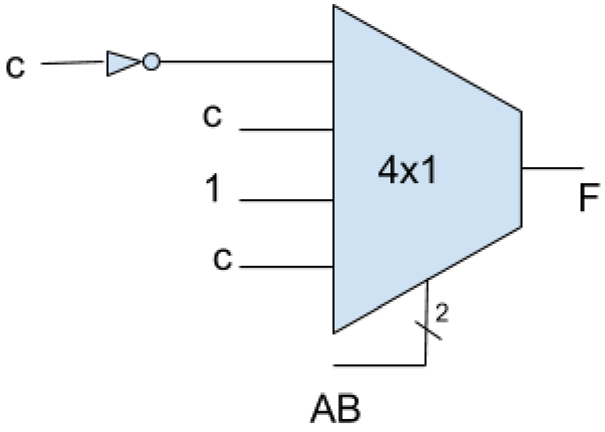


1	0	1	0
1	1	1	0

B. Show a combinational circuit that implements F using only inverters and NAND gates.



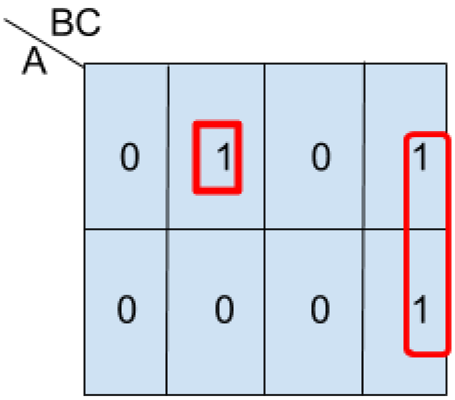
C. Implement F using one 4-input binary-select Multiplexer and inverters.



D. Write a minimal sum-of-products expression for $\neg F$.

$\neg F = (\neg A \wedge \neg B \wedge C) \wedge (\neg C \wedge B)$

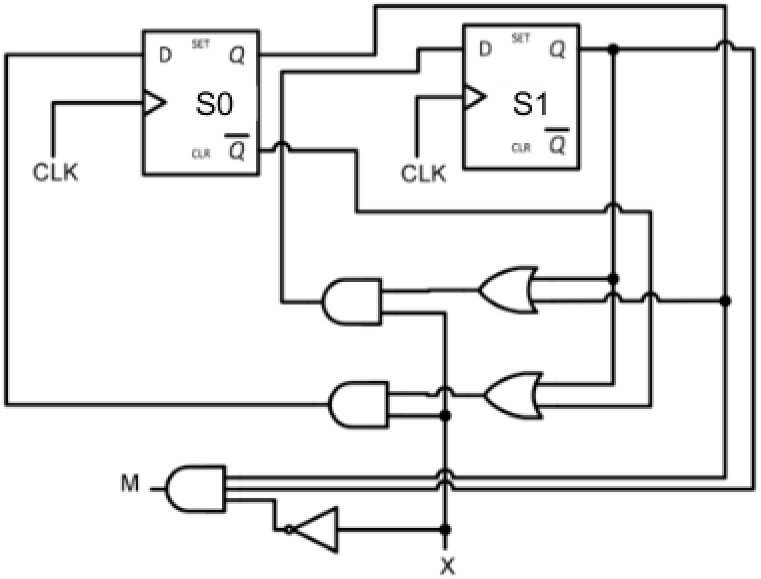
Show supporting work below:



0	1	0	1
0	0	0	1

Problem 3: Sequential Logic Fundamentals [6 + 1.5 + 6 + 1.5 = 15 points]

Consider the following sequential circuit with input X, output M, and two state registers Q0 and Q1.



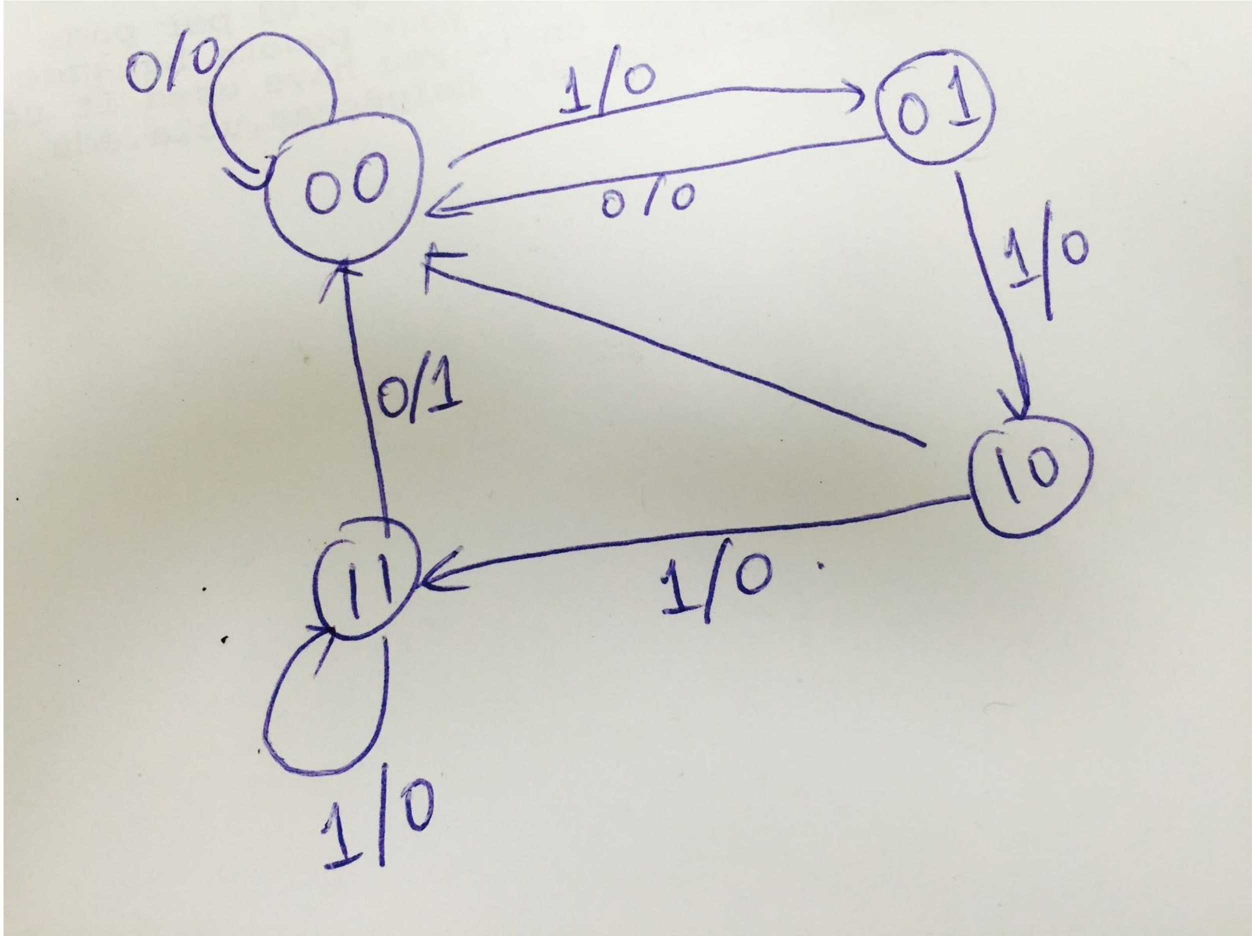
A. Write down Boolean equations for the next state ($S1_{next}$ and $S0_{next}$), and the output M .

$$S1_{next} = (x \wedge (S1_{current} \vee S0_{current}))$$
$$S0_{next} = (x \wedge (S1_{current} \vee \neg S0_{current}))$$
$$M = \neg x \wedge S1_{current} \wedge S0_{current}$$

B. Is this a Meal FSM or a Moore FSM? Justify your answer in one sentence.

Mealy. The output also depends on the input x.

C. Draw the state transition diagram corresponding to this circuit.

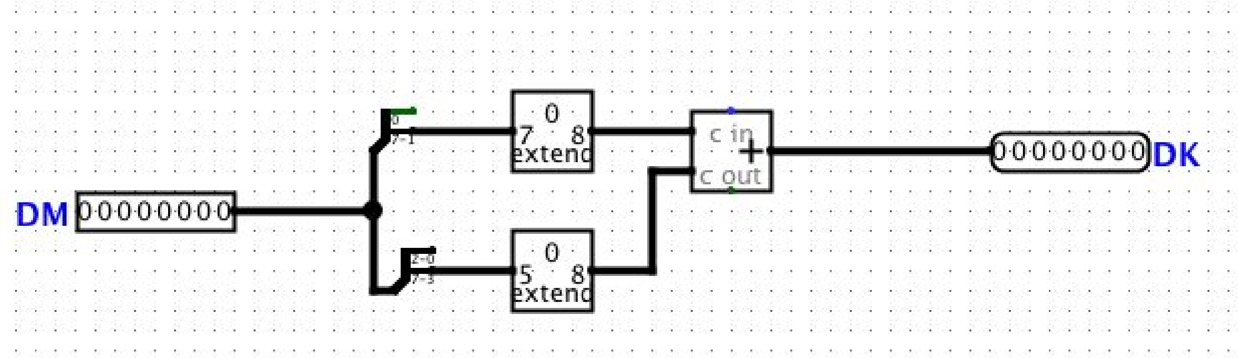


D. What does the circuit do? Describe in simple and plain English. Note that you need to provide a high-level description, e.g. you may say that this circuit detects the pattern <describe the pattern in simple terms>.
Detects the pattern 1110

Problem 4: Miscellaneous [4*5 = 20 points]

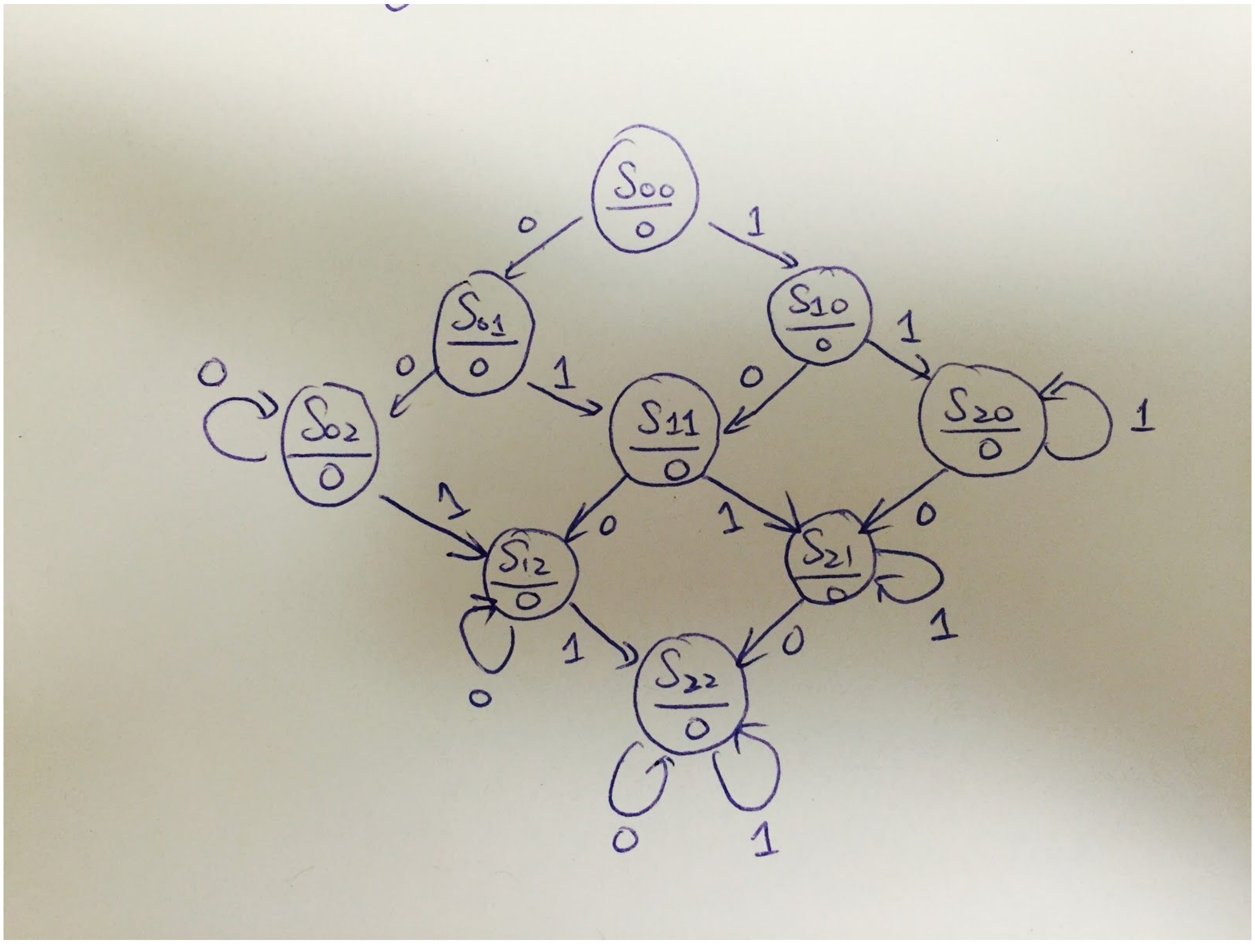
A. One mile is approximately 1.6 kilometers. Design a combinational system which takes as input a distance expressed as in kilometers as a 8-bit unsigned binary integer DK[7:0], and outputs the equivalent distance in miles also as a 8-bit unsigned binary integer DM[7:0]. Your design may use basic logic gates, multiplexers, decoders, encoders, adders, and subtractors. You may not use any other modules (e.g. no multiplier or divider modules are permissible) unless you build them from scratch.

ANSWER:
DM = DK/1.6=DK/16*10=DK/16*(8+2)=DK/16*8 + DK/16+DK/16=DK/2+DK/8
DK/2 is right shift by 1 bit, DK/8 is right shift by 3 bits



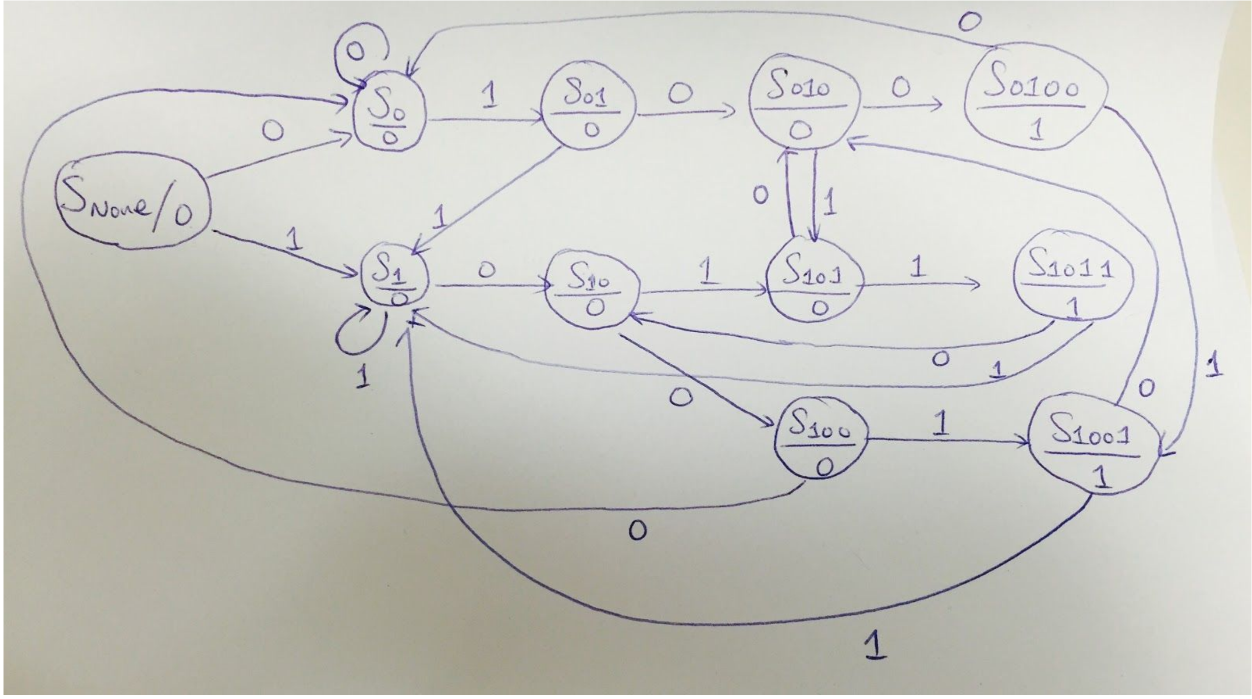
B. A finite state machine has one input IN and one output OUT. The output OUT starts out as 0 but becomes 1 and remains 1 thereafter when at least two 0's and two 1's have occurred as inputs on IN, regardless of the order of appearance. Implement it as Moore FSM. You only need to draw the state transition diagram. (Hint: You can do this in nine states.)

ANSWER:

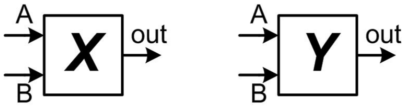


C. Design a state transition diagram for a Moore state machine that takes one input A and has one output X, which should be 1 if, and only if, the last four values of A have been either 1011, 1001, or 0100 (the left most bit represents the oldest value among the four).

ANSWER:



D. Due to the war in the land of Logicia there is a shortage of XOR gates. Unfortunately, the only logic gates available are two weird components called “X” and “Y”. The truth table of both components is presented below – Z represents a High-Z value on the output (i.e. output is not driven, similar to what we saw for tri-state inverters and buffers). Could you help the poor engineers of Logicia build an XOR gate?

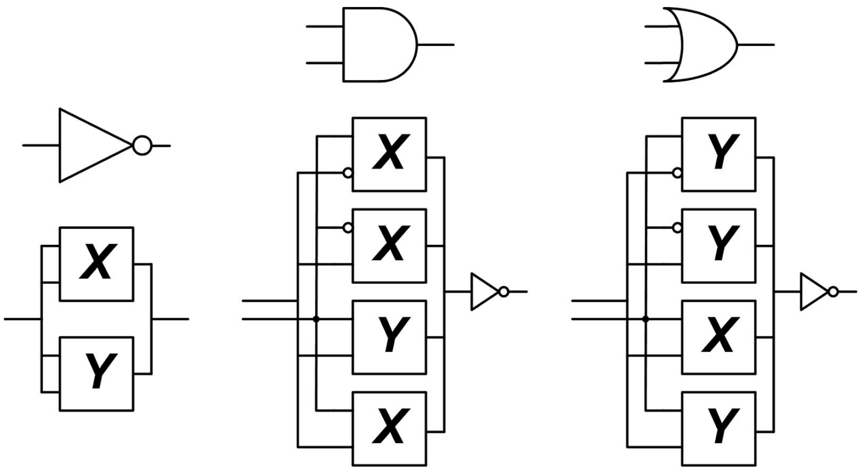


AB	out
00	1
01	Z
10	Z
11	Z

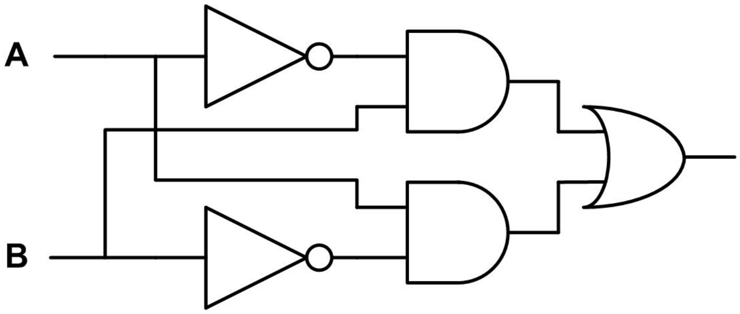
AB	out
00	Z
01	Z
10	Z
11	0

ANSWER:

We can construct NOT, AND2, and OR2 gates as follows:



And the XOR can be constructed as:



A xor B = A’B + AB’

EXTRA SHEET #1

EXTRA SHEET #2

EXTRA SHEET #3