USAF ACADEMY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

ECE 281 GRADED REVIEW #2
SPRING 2016

Name: _____ **Section:** _____

Academic Security	This examination is not released from academic security until **1700** on **8 April 2015**. Until this time, you may not discuss the examination contents or the course material with anyone other than your instructor.
Integrity	Your honor is extremely important. This academic security policy is designed to help you succeed in meeting academic requirements while practicing the honorable behavior our country rightfully demands of its military. Do not compromise your integrity by violating academic security or by taking unfair advantage of your classmates.
Authorized Resources	1. Any calculator without stored relevant information and which does not convert between number systems.
Instructions	▪ **Show all work for full credit** ▪ Box or circle your final answer. ▪ For all numerical answers, use engineering notation and include units. ▪ Completely label all your diagrams, drawings, graphs, etc. for full credit. ▪ You have **53 minutes** to complete this exam.

Problem	Value	Earned
1	20	
2	20	
3	20	
4	20	
5	20	
Total	100	

This page intentionally left blank

Problem 1 (20 points) *General Knowledge*

a. (2 points) You want to build a 128k x 32 memory array. How many bits must you have for each address?

b. (1 point each) Choose the ***best*** answer for each memory descriptor. Do not use any answer twice.

i. ___ very slow memory
ii. ___ uses rising and falling edge to access data
iii. ___ requires UV voltage to erase its contents
iv. ___ uses 6 transistors per bit cell
v. ___ bit cells can be erased in large blocks

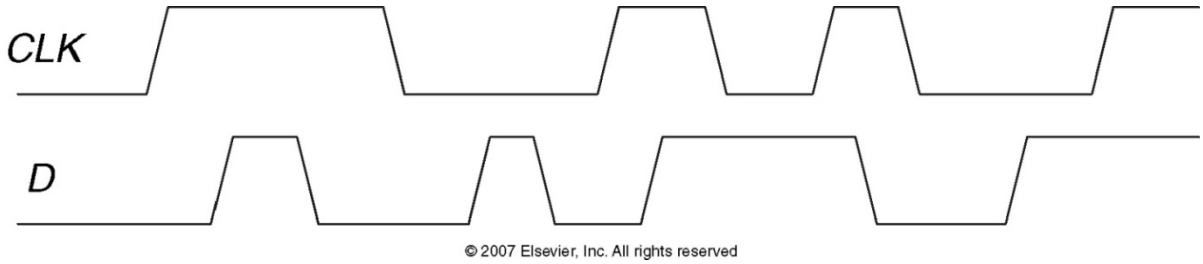
A. PROM
B. EPROM
C. EEPROM
D. Flash
E. DRAM
F. SRAM
G. SDRAM
H. DDR SDRAM

(1.5 points each) Circle the best term for the next two descriptors.

c. Memory with data that is retained when power is lost: RAM ROM

d. Memory that can be created with two-level logic: RAM ROM

e. (10 pts) Given the input waveforms shown below, sketch the output, Q, of a D latch and a D flip flop. Be sure to clearly indicate where certain events line up (dotted lines work great here).



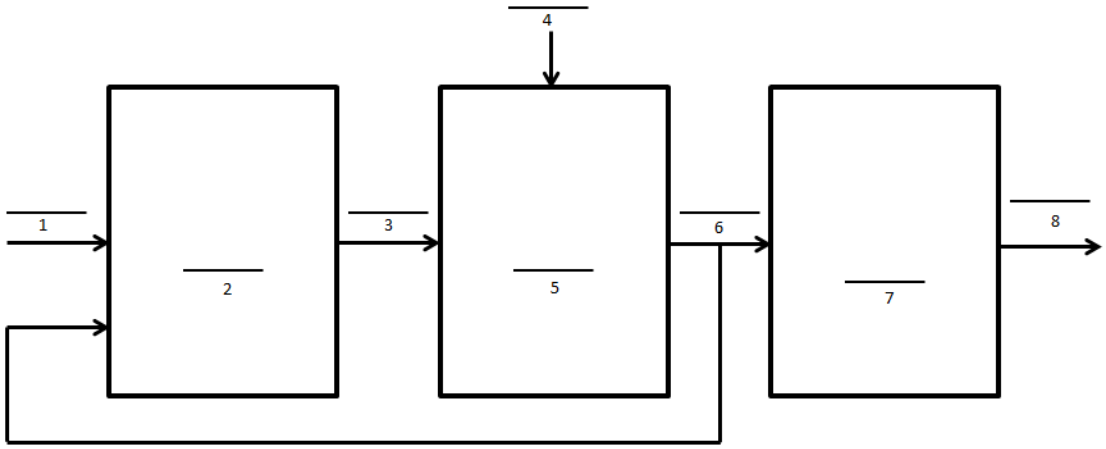
D Latch Q:

D Flip-Flop Q:

Problem 2 (15 points) *Sequential Logic*

a. (8 points) Label the following components on the state machine block diagram below:

A. Userdesignwrapper
B. Clock Signal
C. Outputs
D. Current State
E. Inputs
F. Yellow Box
G. Next-state logic
H. State Memory
I. Output Logic
J. Next State
K. Structure



b. (2 points) Is this a *Mealy* or a *Moore* machine?

c. (2 points) If a state machine has 33 states, what is the minimum number of flip-flops needed for its schematic?

d. (1 point) If a state machine only uses eleven of its sixteen states, what are the remaining states called?

e. (2 points) Describe the mathematical function performed by a **right** shifter. If there is a specific value that is added, subtracted, multiplied or divided, specify it.

Problem 3 (20 points) *FSM Design*

a. (16 pts) You like the combination lock on the ECE 281 classroom so much that you decide to design a similar circuit for your dorm room. You cleverly choose a pass code of 1881, which no one would ever guess. Draw a properly labeled Moore state diagram for your state machine which will detect the sequence 1-8-8-1 and unlock the door. The input, IN, can be any digit from 0-9 and the output is a binary signal, UNLOCK, which goes high when the sequence is detected. Note : If IN = 1, the opposite of that is IN ≠ 1. You don't have to write out every case. Be sure to correctly handle the case where you input a string of 1's followed by 8 8 1 (e.g. 1 1 1 8 8 1)

For example, the following input would provide the associated output:

IN: 5 1 8 8 1 8 8 1 8 8 1 1 8 8 1 3 1 8 1 6

UNLOCK: 0 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0

b. Colleen has written a testbench to help her test VHDL code that computes the function $Y = \overline{A}\overline{B}\overline{C} + A\overline{B}\overline{C} + A\overline{B}C$.

```
A <= '1';  B<= '0'; C <= '0';
  wait for 10 ns;
  assert Y = '1' report "Test 1 failed." severity error;

  A <= '1';  B<= '1'; C <= '1';
  wait for 10 ns;
  assert Y = '1' report "Test 2 failed." severity error;
```

i. (2 points) What console output will she see for this snippet of code testing the two minterms?

ii. (2 points) Did Colleen write the testbench to properly test her code? Explain your reasoning.

Problem 4 (15 points) *Number Systems, Logic Arrays*

a. (4 points) Convert the fixed-point Q3.5 number below to decimal. Show your work.

0x6B

b. (5 points) Add the two following IEEE 754 single-precision floating-point numbers. Circle the correct sum below.

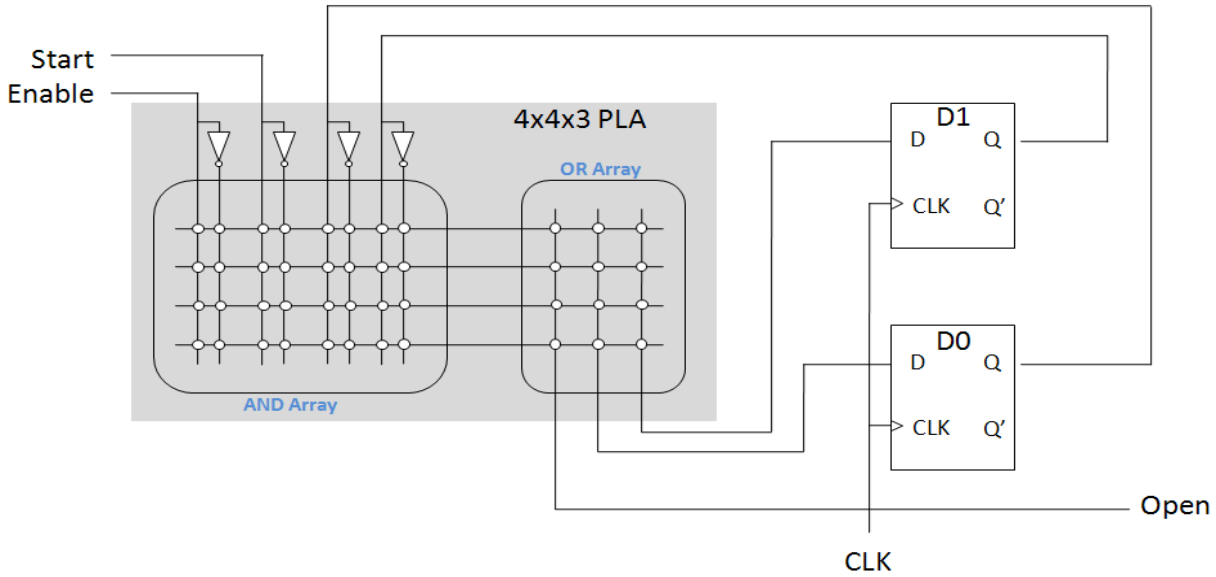
0x432FA000
+ 0x3C210000

a. 0x7F50A000 b. 0x432FA284 c. 0x4350A000 d. 0x4357D142

c. (5 points) You realize that by making use of an appropriately sized PLA, you can greatly simplify your implementation of sequential circuits. Complete your design by using dot notation on the following PLA to implement the given logic functions.

$Q_1^* = Q_1 \cdot Q_0' \cdot Enable$
$Q_0^* = Start \cdot Enable$
$Open = (Q_1' \cdot Q_0) + Enable'$

Hint: Before you begin, label the inputs/outputs of the PLA as Q_1, Q_0, Q_1^*, and Q_0^*



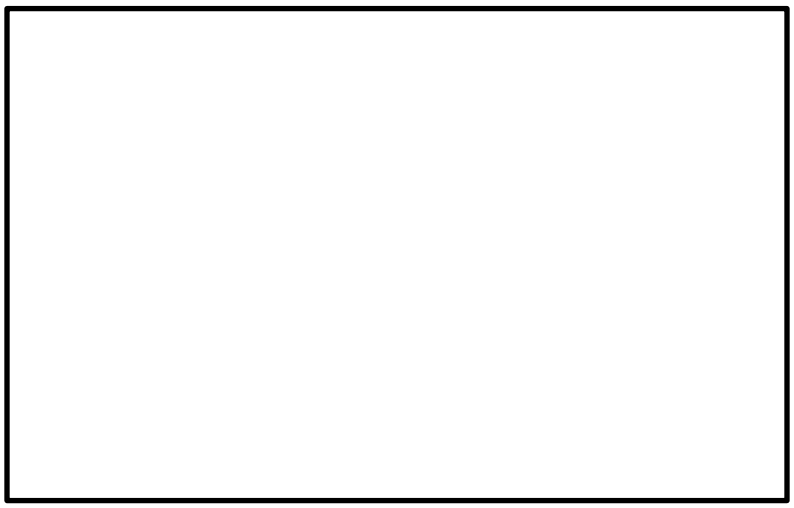
Problem 5 (20 points) *Sequential Building Blocks Application*

Design a 16-bit counter that adds 0x0002 at each rising clock edge. When the counter equals 0x0034, output Z is asserted to light an LED. When the counter equals 0x2006, the counter resets and begins counting again from 0. The circuit has only a clock input. Upon reset, the counter output changes to 0. You may use D flip-flops, adders, and any other digital logic elements we've discussed in class that you may need. Draw and clearly label the diagram to represent such a system.

Problem 6 (10 points) *VHDL*

a. (6 points) Draw the block diagram for the below VHDL module using the provided box as the entity box. Be sure to fully label each component, signal, wire, and bus.

```
entity structural_prog is
   Port ( A : in  STD_LOGIC;
          B : in  STD_LOGIC_VECTOR (2 downto 0);
          X : out  STD_LOGIC;
          Y : out  STD_LOGIC_VECTOR (1 downto 0););
end structural_prog;

architecture Struct of structural_prog is

component TARDIS
          port (key, door, phone   :  in std_logic;
                bigger             :  out std_logic);
end component;

component villain
          port (exterminate, delete :        in std_logic;
                doom, death         :        out std_logic);
end component;

signal s1, s2  : std_logic;
signal b5 : std_logic_vector (1 downto 0);

begin

  Rose: TARDIS
  port map (A, B(1), B(0), s1);

  River: TARDIS
  port map (A, B(1), B(2), s2);

  Dalek: villain
  port map (B(0), s2, X, b5);

  Y <= b5 & s1;

end Struct;
```



b. (1 point each) Circle the correct answer.

T	F	The only time a process can ever run is when the clock changes.
T	F	Blocking statements are evaluated in the order they appear, like normal code.
T	F	One can mix structural and behavioral VHDL programming in a single file.
T	F	A PLA can implement sequential logic.