Exam 2

Last Name, First Name

May the Schwartz be with you!

UF's NaviGator AMS and team.

Please read carefully.

Instructions:

- *Turn off all **cell phones** and other **noise making devices** and put away **all electronics**.*
- *Show **all** work on the **front** of the test papers. **Box** each answer. If you need more room, make a clearly indicated note on the front of the page, "**MORE ON BACK**", and use the back. The back of the page will **not** be graded without an indication on the front.*
- *You may not use any notes, HW, labs, other books, scratch paper, or calculators.*
- *This exam counts for **20**% of your total course grade.*
- *Read each question **carefully** and **follow the instructions**.*
- *The point values for problems may be changed at prof's discretion.*
- *You must pledge and sign this page in order for a grade to be assigned.*
- ***CLEARLY** write your name at the top of **this** test page (and, if you remove the staple, all others). Be sure your exam consists of **11** distinct pages. Sign your name and add the date below. (If we struggle to read your name, you will lose points.)*
- *For anything undefined, if necessary, state a reasonable assumption.*
- *Failure to follow the below rules will result in **NO** partial credit*
  - *The **base** (radix) of all number should be indicated with a **subscript** or **prefix**.*
  - *Truth tables, voltage tables, and timing simulations must be in **counting** order.*
  - *Label the inputs and outputs of each circuit with activation-levels.*
  - *For each mixed-logic circuit diagram, label inputs of **each** gate with the appropriate logic equations.*
  - *For K-maps, label **each** grouping with the appropriate equation.*
  - *Labels inside of parts must be provided whenever it can be confusing, e.g., they **MUST** be specified for MUXes and Decoders, but not for NANDs and ORs.*
  - *For each circuit design, equations must **not** be used as replacements for circuit elements.*
  - *Boolean expression answers must be in **lexical order**, (i.e., /A before A, A before B, & D_3 before D_2).*
  - *When asked to provide a numerical answer, provide a single number only, e.g., 37.9, **NOT** expressions like $3^7 x \sqrt[4]{37}$, or fractions, like 37/42. Provide the proper number of significant figures.*

Good luck & Go Gators!!!

PLEDGE: On my honor as a University of Florida student, I certify that I have neither given nor received any aid on this examination, nor I have seen anyone else do so.

SIGN YOUR NAME _____ DATE (5 April 2018) _____

Regrade comments below: Give page # **and** problem # and reason for the petition.
.
.
.
.
.
.
.
.

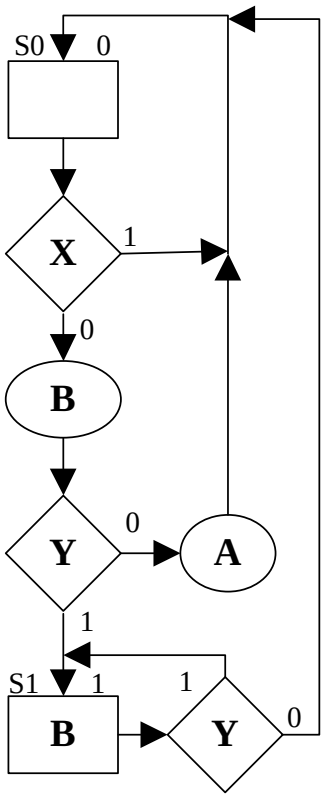
Page	Available	Points
2-3	17	
4	15	
5-6	22	
7	12	
8-10	19	
11	15	
TOTAL	100	

Exam 2

Last Name, First Name

[17%] 1. In this problem you will ultimately design a circuit to implement this ASM. Use a **JK-FF** for the **least significant state bit**, and **T-FFs** for any **other** FF's that you might need. The signals **X** and **A** are **active-low** and **B and Y** are **active-high**. An active-high signal **Start** should **asynchronously** move the machine to state **S1**.

3 min



(7%) a) Complete the next-state **truth** table. You may **not** need all the rows and/or columns. Use wild cards (*) where possible.

7 min

(3%) b) Find the required VHDL equations (**no need** to simplify) **for determining A and the JK-FF inputs**. (You do **NOT** have to find the other equations.)

5 min

Exam 2

Last Name, First Name

(5%) 3 min 1. c) Design the complete mixed-logic circuit diagram, using a JK-FF (and T-FF(s), if necessary) as described previously. Use a block diagram for combinatorial logic (like you did in Lab 5 when you created a graphical symbol in Quartus); label all inputs and outputs as Quartus would when a graphical symbol is created. All **inputs** and **outputs** of your circuit design should be **clearly indicated coming into or out of** the below box. Your design must include the circuitry necessary to **asynchronously** cause the system to go to state **S1** when the **Start** (active-high) signal goes true.

5 min **Inputs** **Outputs**

(2%) 2 min d) What changes are necessary if A is active-high instead of active-low and Y is active-low instead of active-high? Describe any changes necessary in part b and write any new or changed VHDL below.

Exam 2

Last Name, First Name

[6%] 2. Answer the following questions about a state machine designed with one EEPROM, one J-K flip-flop, one T flip-flop, and any other necessary flip-flops of type D.

(3%) 3 min a) What is the size of the EEPROM (addresses × data bits) if the state machine has 4 inputs, 2 outputs, and 7 states? Provide numbers only, i.e, 37 by 9, **NOT** expressions like $3^7 \times \sqrt[4]{37}$.

(3%) 3 min b) What is the size of the EEPROM (addresses × data bits) if the state machine has 3 inputs, 1 outputs, and 15 states? Provide numbers only, i.e, 37 by 9, **NOT** expressions like $3^7 \times \sqrt[4]{37}$.

[4%] 5 min 3. Draw a complete **mixed-logic circuit diagram** to generate the below equation. Do **NOT** simplify. **The circuit must include three switch circuits**, one for an **active-low** input signal **U(L),** one for an **active-high** input **B(H)**, and one other for input **S**.

$$UF = \overline{\overline{S * \overline{U}} + \overline{B}}$$

Use the **minimum number of gates** on only any **single SSI chip** (i.e., a single **SSI** 74HCxxx chip). Draw the switches in their **true** positions. Draw an **LED circuit** for the **active-low** output, **UF(L)**. Do **NOT** draw a layout. Do **NOT** simplify.

[5%] 4. Answer the following short questions/problems.

(2%) 1 min a) In VHDL, what is an architecture?

(3%) 2 min b) Describe, compare and contrast Mealy and Moore outputs. When is one desirable over the other?

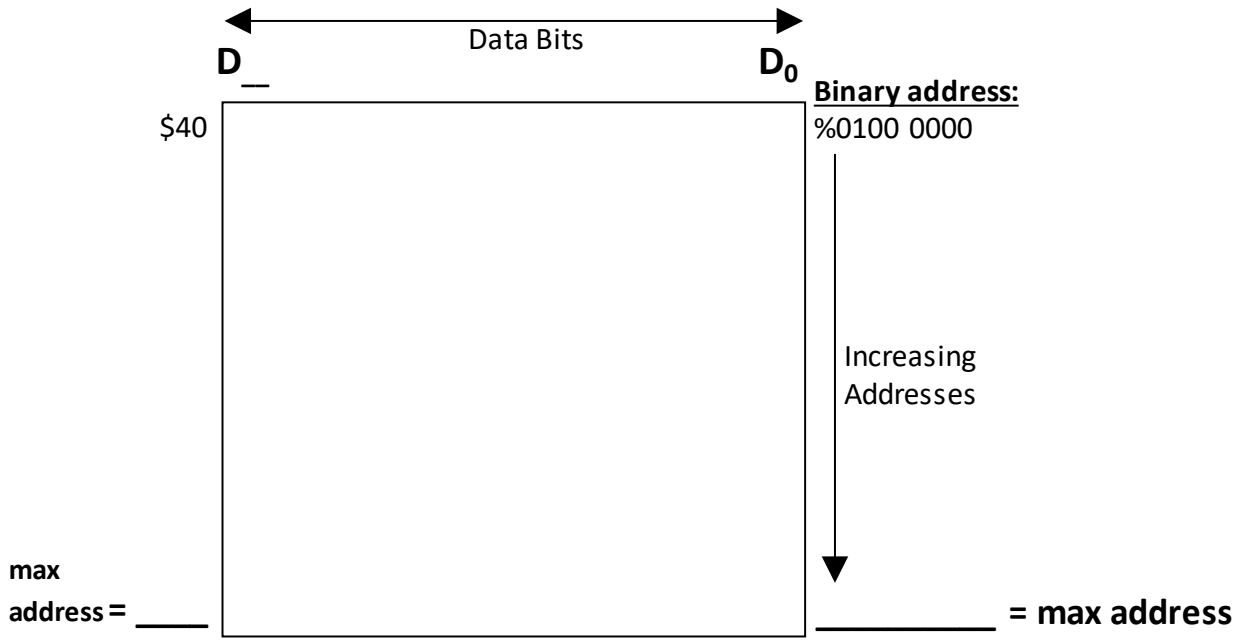
Exam 2

Last Name, First Name

[22%] 5. Given as many of each of the following as needed, design the memory module described below, with an **active-low** chip enable, **CE(L)**: **32×8 FLASH EEPROM**, **64×8 EEPROM**, and **16×4 SRAMs**, The memory module should begin at address **$40 = 40_{16}** with **32x8** of **Flash,** followed immediately by **64×8** of **EEPROM**. The EEPROM should be followed immediately with **16×8** of **SRAM**. Add the **minimum number** of memory devices required. Addresses outside of those required (but within **$0** to **$FF**) must be **unaffected**.

3 min

(%) 6 min a) Draw vertical and horizontal lines in the box below and label each resulting box with the memory type and size, using only the defined types and sizes given above. Also, fill in the subscript on the D (top left) and the maximum address (bottom left and right).



(%) 5 min b) What are the address and data ranges for **each of the** memory components drawn above (in **binary** and in **hex**)?

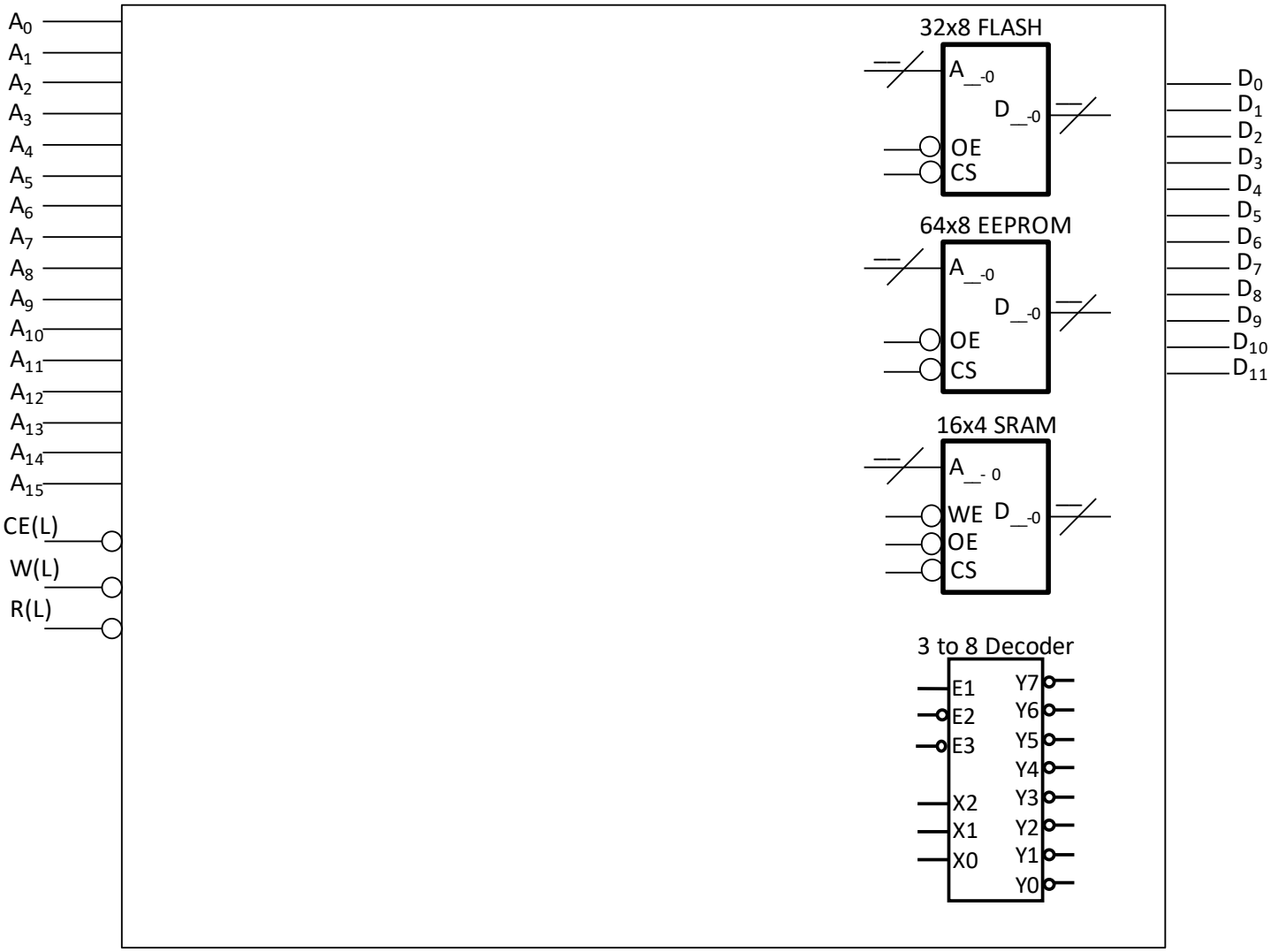
Equations:

Exam 2

Last Name, First Name

(%) 5. c) 8 min

Design the required memory device circuit diagram below. Use the single decoder shown (bottom right) and add the **minimum number** of **additional** memory components and SSI **gates** necessary (but **no** other MSI gates, LSI gates, or PLDs). **Add** memory components and address and data subscripts as needed; **cross out** unneeded address and data pins. **Use labels instead of wires for the design**. **Also**, **write the equations** for each CS at the bottom of the page. Show **all connections** with either labels or wires (**labels are preferred**), just as in Quartus. Don't forget the system's **CE(L), R(L), and W(L)**. Note: R(L) and W(L) are never true at the same time.



Equations:

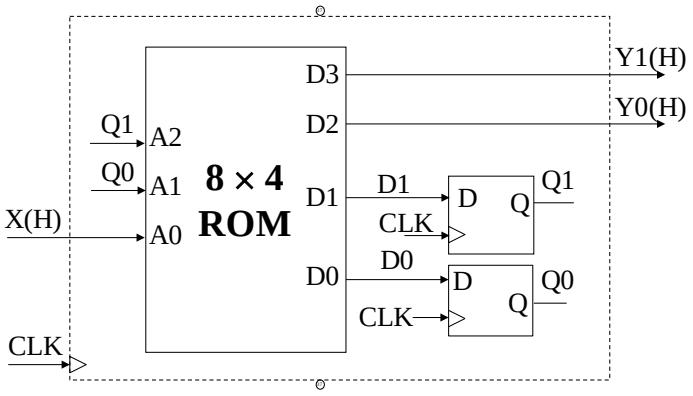
Exam 2

Last Name, First Name

[12%] 6. 14 min

Use the below circuit diagram and the **ROM contents table** below to solve this problem. Note that the addresses and data in the table are in **hexadecimal** [base 16]. (This problem is very similar to a problem in Homework 10. A similar problem was solved in class. Note that two D-FFs are utilized in the below circuit.

Derive the ASM chart for this circuit. Show **ALL** work, i.e., use at least part of the below table. Draw your ASM in the space provided on this page.



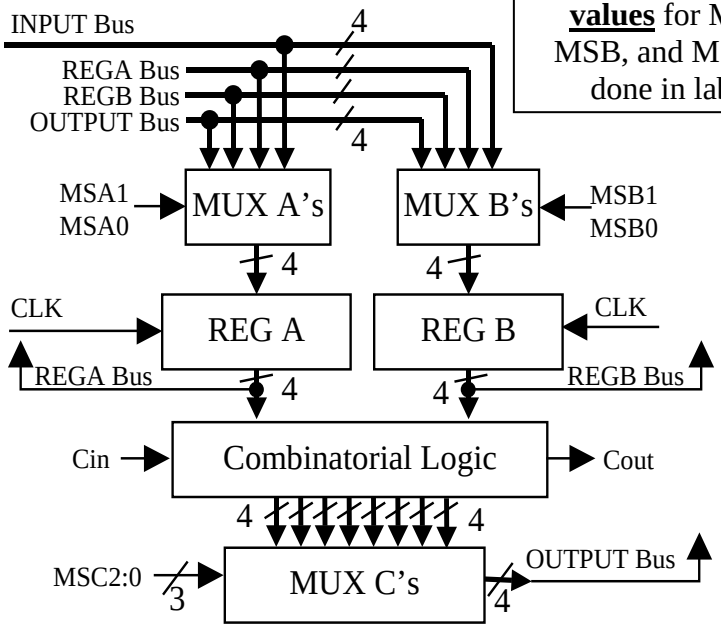
				Contents of ROM (Hex)									
				Addr	Data								
				0	8								
				1	3								
				2	8								
				3	8								
				4	6								
				5	3								
				6	7								
				7	D								

Exam 2

Last Name, First Name

[19%] 7. A block diagram and tables from your Lab 4 are shown below.

3 min



Use **proper default values** for MSA, MSB, and MSC, as done in lab 4.

MSA1:0/ MSB1:0	Bus Selected as Input to Combinatorial Logic
00	INPUT Bus
01	REG A Bus
10	REG B Bus
11	Output Bus

MSC	Action
000	REGA Bus to OUTPUT Bus
001	REGB Bus to OUTPUT Bus
010	complement of REGA Bus to OUTPUT Bus
011	bit wise AND REGA/REGB to OUTPUT Bus
100	bit wise OR REGA/REGB Bus to OUTPUT Bus
101	sum of REGA Bus & REGB Bus to OUTPUT Bus
110	shift REGA Bus left one bit to OUTPUT Bus
111	shift REGA Bus right one bit to OUTPUT Bus **without** sign extension

Note that **Cout** is the carry output of the 4-bit adder.

(3%) 5 min a) Assume that you can add a small additional circuit to determine when REGA is **NOT** equal to 3 or 7, i.e., **ANOT3OR7** is true when REGA equals any 0, 1, 2, 4, 5, 6, 8, 9, 10, 11, 12, 13, 14 or 15. Design the simplified (MSOP or MPOS) circuit for this **active-high** output. Appropriately label the inputs and output. Show **ALL** work.

Exam 2

Last Name, First Name

(4%) 5 min 7. b) Calculate **$E** plus **$5**, putting the result in **RegA**. Describe your algorithm (plan) and then what is accomplished in each step. Use the minimum number of clock cycles. Give appropriate values for **ALL** signals. (Your algorithm should still work if all of the numbers are changed.)

#	MSA	MSB	MSC	Input	Cin	RegA	RegB	Output	Cout	RegA+	RegB+	Output+	Cout+
1						0011	0111						
2													
3													
4													
5													
6													

Plan: _____

_____

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____

(4%) 5 min 7. c) In the next three parts of this problem, we will modify the circuit from Lab 4 in order to perform multiple nibble addition, e.g., $37 plus $5F = $96. Draw a simple circuit that can be added to the Lab 4 circuits to "remember" the last carry output (Cout) when a **new** input, **LD_Cout(H)**, is true. Label the stored carry output as **S_Cout(H)**.

Exam 2

Last Name, First Name

(3%) 5 min 7. d) In Lab 4, the carry input (Cin) was obtained from a switch circuit. Instead, in the next two parts of the problem, we will use either GND or S_COUT (the stored carry output from part c) for the carry input. If a new input, **USE_Cout** is true, then use the stored carry output (**S_Cout**) for the carry input; otherwise use GND for the carry input. Draw a simple circuit that can be added to the Lab 4 circuit (**without** the Cin switch circuit); label the output of the below circuit as **Cin**.

(5%) 5 min e) Calculate **$2E** plus **$45**, putting the results in **RegA**. First find the result of adding the least significant nibbles and then the result from adding the most significant nibbles (possibly with carry). Describe your algorithm (plan) and then what is accomplished in each step. Use the minimum number of clock cycles. Give appropriate values for **ALL** signals. (Your algorithm should still work if all of the numbers are changed.) What are the two relevant outputs in **RegA**?

#	MSA	MSB	MSC	Input	LD_Cout	USE_Cout	Cin	RegA	RegB	Output	Cout	S_Cout	RegA+	RegB+	Output+	Cout+	S_Cout+
1								0011	0111			0					
2																	
3																	
4																	
5																	
6																	
7																	
8																	
9																	

Plan: _____

_____

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____
9. _____

Exam 2

Last Name, First Name

[15%] 8. 12 min Complete the timing diagram for **this** ASM. Show small (non-zero) propagation delays. Assume that each of the flip-flops used are rising-edge triggered. This system has an asynchronous reset (**Reset)**. (Shown is a simplified ASM chart for an old UF robot.) Assume that all propagation delays are approximately 10% (i.e., $\frac{1}{10}$ th) of the CLK period. (For example, if the clock period was 100 ns, the propagation delays would be 10 ns.)