Name: _____

NetID: _____@wisc.edu

Homework 7

CS/ECE 252 Section-2 (MWF 11:00)

Assigned on November 6th

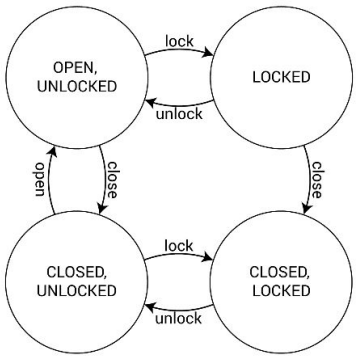
Due on Friday, November 20th by the beginning of class (11 AM)
Neat and legible and writing is preferred, especially for your name and NetID.

1. List the major components of a state machine and give descriptions for each.

(3)

2. For the state machine in the figure below, mention the states transitioned to when the following sequence of inputs is applied every clock. If no option for the state with the input is listed, then the machine stays in that state. The initial state is (CLOSED, UNLOCKED).
lock, open, unlock, open, lock, lock. The first couple states are given.

(4)



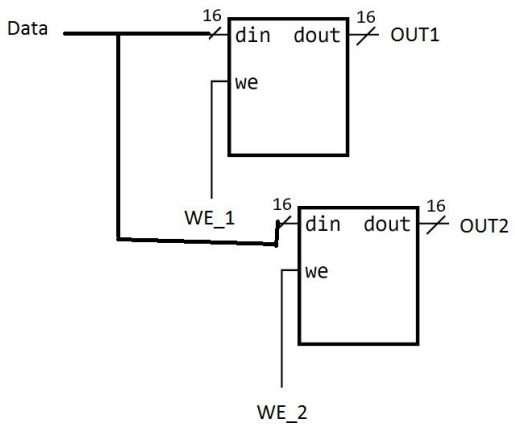
Input	Current State	Next State
lock	**Closed, Unlocked**	**Closed, Locked**
open	**Closed, Locked**	
unlock		
open		
lock		
lock		

3. Why are operations that have data dependencies not allowed in the same state **at the same time**? **(2)**

4. What are auxiliary registers? Can a programmer access them directly? **(2)**

5. What are the five stages of a state machine for a computer's microarchitecture? **(5)**

6. In the following circuit, we have two auxiliary registers wired up as shown. We have three inputs, WE_1, WE_2, and Data. We have two outputs, OUT1 and OUT2. The table below has the inputs filled in over a certain period of time. Fill in the outputs. Notice the inputs happen sequentially from left to right on the table. Assume that OUT1 and OUT2 are initially 0. **(5)**



WE_1	**0**	**1**	**0**	**0**	**1**
WE_2	**0**	**1**	**1**	**0**	**0**
Data	**0**	**1**	**0**	**1**	**0**
OUT1					
OUT2					

7. You are given scores (out of 10) for 4 students. Draw a circuit which can select the score of one among these 4 students, as an output. Clearly mention the bit-width of each bus. **(2)**

8. You want to create a circuit using MUXes, ALU, and wires which has the following definition:
Inputs: **(6)**
$A(A_4A_3A_2A_1)$: 4-bit input
$B(B_4B_3B_2B_1)$: 4-bit input
SAL: Select ALU operation: 2 bit input

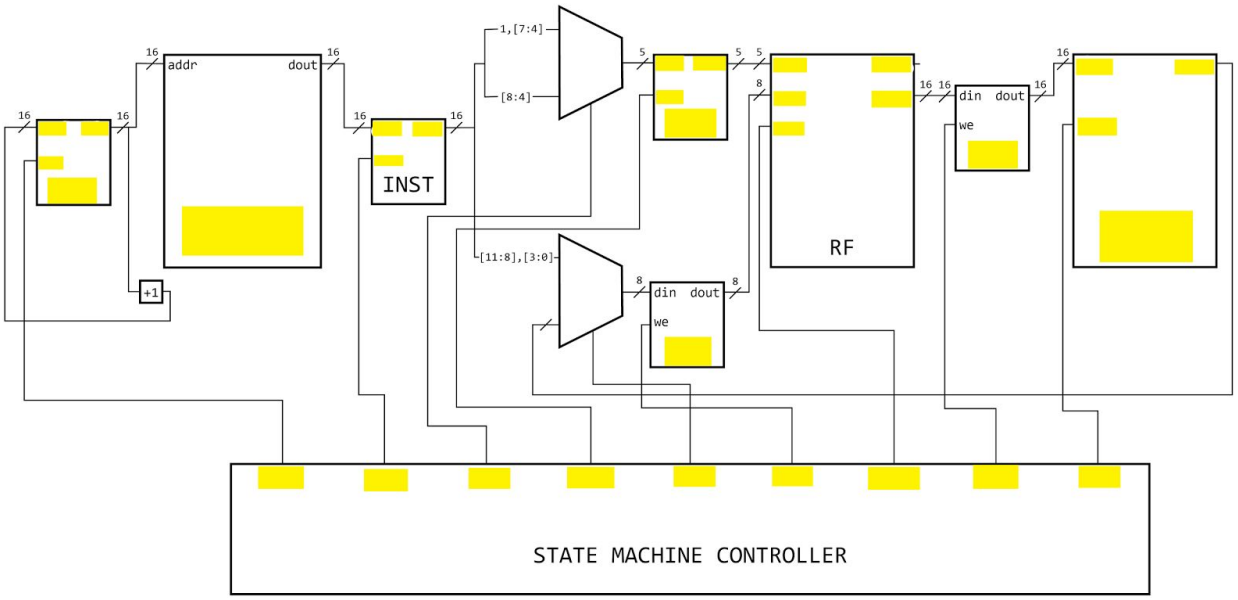
Output: $R(R_4R_3R_2R_1)$: 4-bit output

The operation to be performed by the ALU is selected according to the following table:

SAL	**Operation**
00	Add
01	Subtract
10	bitwise AND
11	bitwise OR

Draw the circuit diagram for this.

9. Below is a diagram for a circuit that can implement the LD and LDI instruction. Write in all the missing labels for ports and components. The missing labels are highlighted. **(2)**



10. In the table at the end of the chapter, several blocks are vacant (without a 0 or a 1). Why is that?
(2)

11. Trace through the instructions below and give the list of states that the program goes through until completion by filling out the following table. The first few states for the first instruction are given to you.To determine the state numbers, reference the state diagram given in the lecture 25 slides (http://pages.cs.wisc.edu/~karu/courses/cs252/fall2015//handouts/lecture/lec-notes-25.pdf, 2nd slide)

(5)

```
ldi r16, 15
ldi r17, -15
add r17, r16
subi r17, 1
```

Cycle	Current State	Instruction/Opcode
0	00000	ldi
1	00001	ldi
2	00101	ldi
3	10010	ldi
4		

12. Trace through the instructions below. For each instruction, walk through the states and give the name of the auxiliary register modified (if any) and its value, and give the control signals that change for that state. You do not need to write if a control signal goes low in the next cycle after it is used (for example, *we* signals). The program starts at PC=0, and the first few lines are given to you.

(10)

```
ldi r16, 15
ldi r17, -15
add r17, r16
breq 1
ld r26, X
subi r17, 1
```

Cycle	Current State	INST	Changed aux registers/values and control signals (if any):
0	00000	ldi	INST = 0xe00f INST_we = 1
1	00001	ldi	REG = 0x10 REG_we = 1 REG_sel = 0
2	00101	ldi	VAL = 0x0f VAL_we = 1 VAL_sel=3
3	10010	ldi	RF_sel = 0 RF_we = 1
4	10011	ldi	

