

NAME:

CWID:

For this exam you may

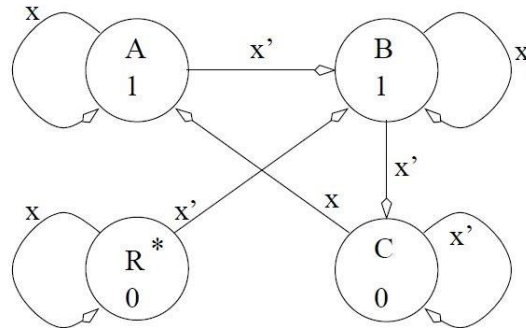
- 2 3x5 cards, front and back

Make sure to:

- Show your work

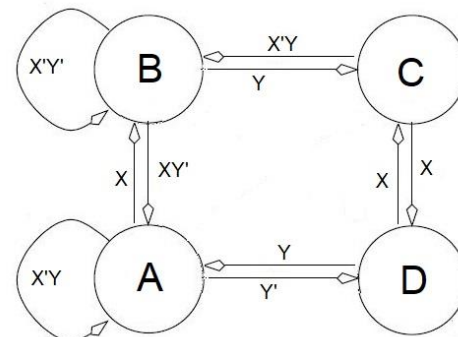
1. (10 pts.) Assume a one's hot encoding of states. Write the memory input equations for state B and C in the following finite state machine. Write the output equation, the outputs are written in each state.

$D_B =$
$D_C =$
$Z =$

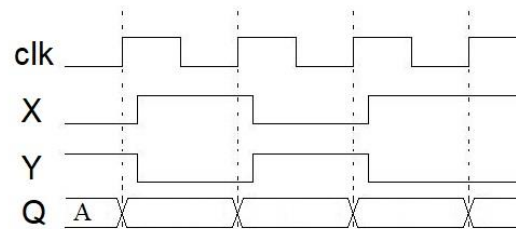
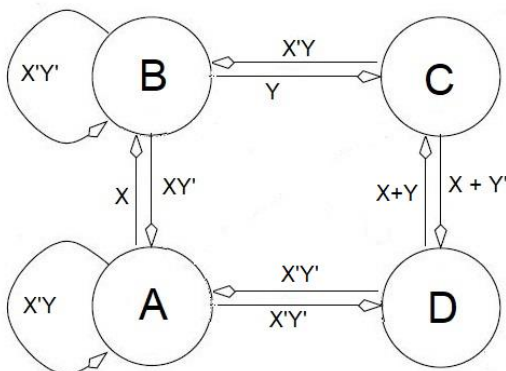


2. (10 pts.) Determine if the transitions leaving each state are complete/unequivocal. Put a yes/no in each cell to indicate the state of the condition.

State	Complete?	Unequivocal?
A		
B		
C		
D		

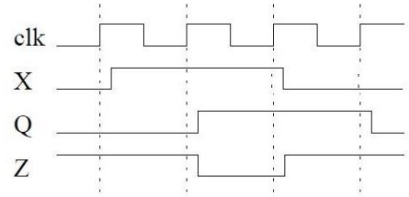
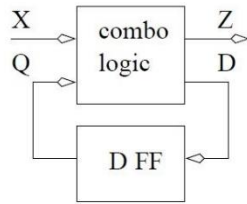


3. (10 pts.) Determine the sequence of states that the following FSM goes through given the inputs provided on the timing diagram. The FSM starts out in state A. Write your answers on the timing diagram.



4. (10 pts.) Complete the truth table that defines the logic inside the box labeled “combo logic” based on the timing diagram given.

Q	X	D	Z
0	0		
0	1		
1	0		
1	1		

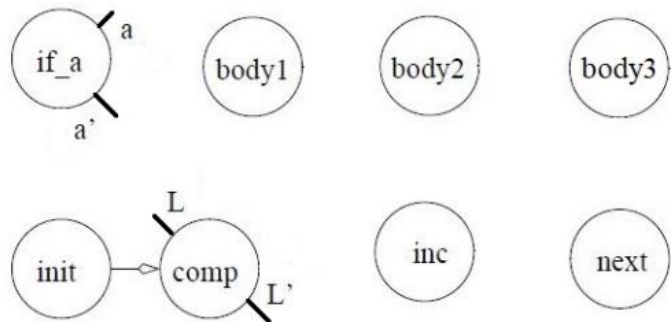


5. (10 pts.) You are building the FSM for the following algorithm. Complete the state diagram by adding arcs. Document your answers in the table below. You may write on the state diagram, but I will ignore anything written there.

```

for (i=0; i<A; i++) {
    if(a) {
        body1;
    } else {
        body 2;
    }
    body3;
}
next;

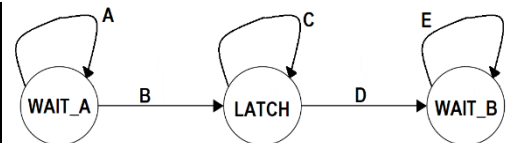
```



Question	Answer
The arc labeled a leaving state if_a goes to state	
The arc labeled a' leaving state if_a goes to state	
The arc labeled L leaving state comp goes to state	
The arc labeled L' leaving state comp goes to state	
The arc leaving state body1 goes to state	
The arc leaving state body3 goes to state	
The arc leaving state inc goes to state	

6. (10 pts.) You are building a state diagram to implement a 2-line handshake where your circuit is the active consumer using the rough state diagram below. Answer the following questions.

Question	Answer
What arc should be removed?	
What condition should be on the arc labeled A ?	
What condition should be on the arc labeled D ?	
What is the condition on the arc labeled E ?	
What is the output in the LATCH state?	
What is the output in the WAIT_B state?	



NAME:

CWID:

7. (15 pts) Complete the datapath for the following algorithm by filling in the name of the signal associated with the grey boxes in the **table** below. For example, if the signal inside the grey box labeled “A” in the datapath was supposed to be MBR (it’s not because there is no MBR in the problem), you would write “MBR” in the shaded space below the letter A in the **table**. Some notes:

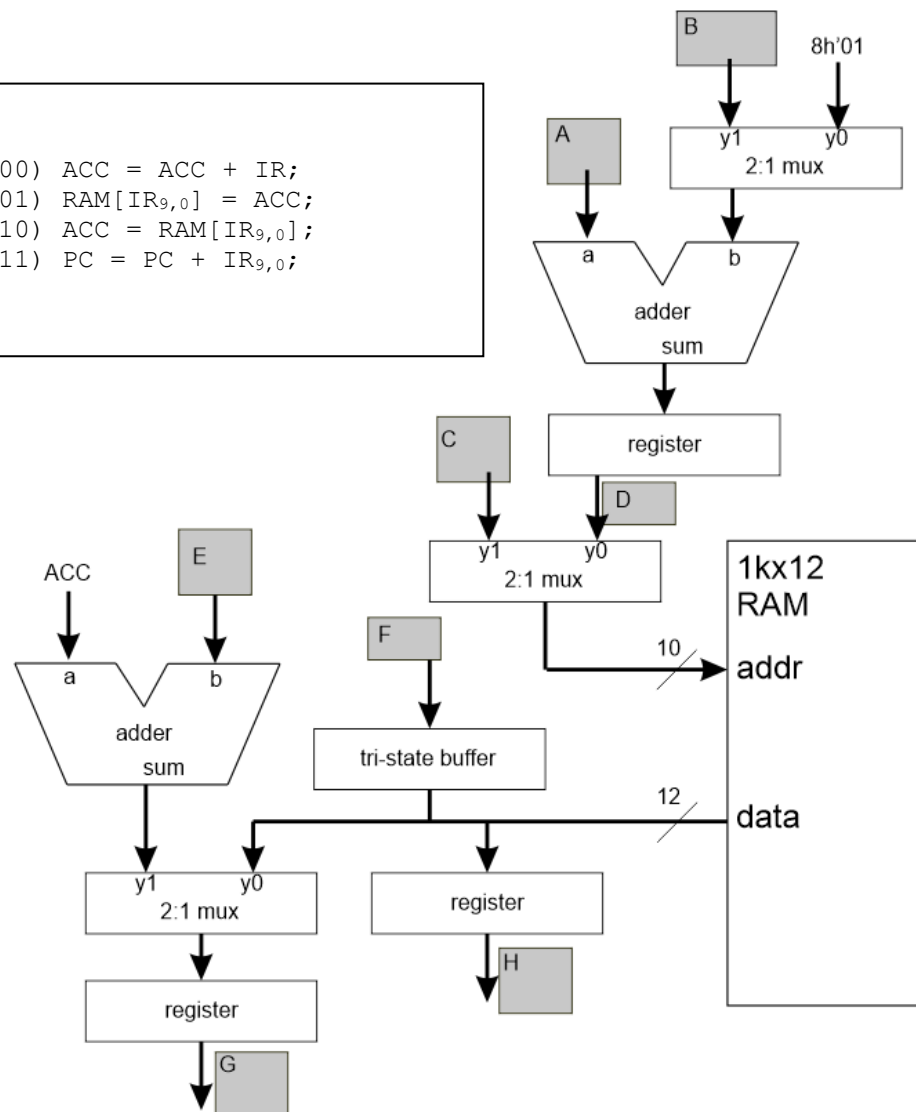
- If you need to use a subvector of a signal use subscripts like those shown in the algorithm.
- Not all the connections in the datapath are explicitly drawn. Some signal names will repeat in the datapath, meaning there is a connection between these wires. As a result, some signal names will repeat in your table of answers.
- You may write on the datapath picture, but I will ignore anything written there.
- Ignore the line numbers in the algorithm – we’ll come back to them later.

A	B	C	D	E	F	G	H

```

1. while(1) {
2.   IR = RAM[PC];
3.   if (IR11,10 == 00) ACC = ACC + IR;
4.   if (IR11,10 == 01) RAM[IR9,0] = ACC;
5.   if (IR11,10 == 10) ACC = RAM[IR9,0];
6.   if (IR11,10 == 11) PC = PC + IR9,0;
7.   PC = PC + 1;
8. } // end while

```

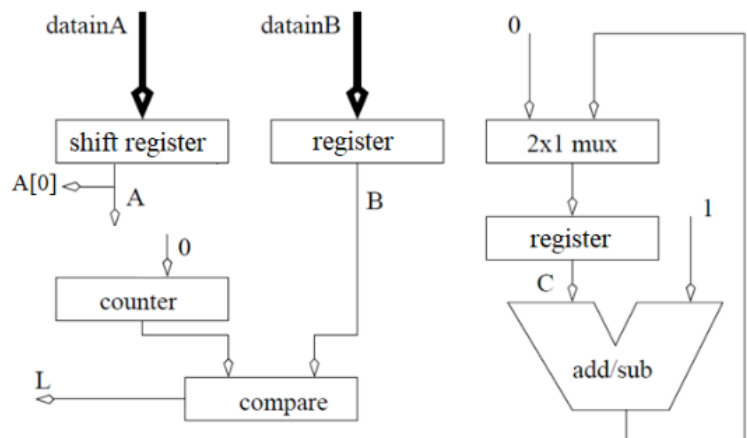
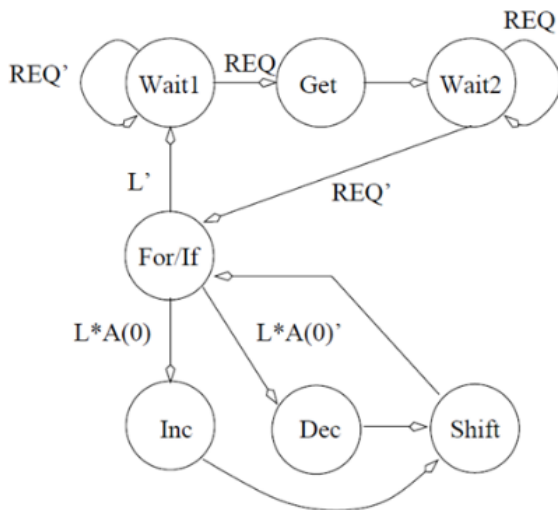


8. (15 pts) This question deals with the construction of a digital circuit to accomplish the task specified by the following algorithm. The datapath and control given below. It is your job to fill out the control word.
- A(0) refers to the LSB of A.
 - A>> 1 refers to A shifted right 1 bit.
 - If a sequential device holds in a particular state, leave the corresponding table cell blank.
 - Initialize the counter and register C in state Get.

```

while(1) {
    ACK = 0;
    while(REQ == 0);
    A = datainA;
    B = datainB;
    ACK = 1;
    while (REQ == 1);
    ACK = 0;
    C = 0;
    for (i=0; i<B; i++) {
        if (A(0) == 1) then
            C = C + 1;
        else
            C = C - 1;
        A = A >> 1;
    }
}

```



State	ACK	shift reg A	register B	register C	2x1 mux	counter	add/sub
	0	00 hold	0 hold	0 hold	0 pass 0	00 hold	0 add
	1	01 shft right	1 load	1 load	1 pass add/s	01 down	1 sub
		10 shft left				10 up	
		11 load				11 load	
Wait 1							
Get							
Wait 2							
For/If							
Inc							
Dec							
Shift							

NAME:

CWID:

9. (10 pts.) Let's look at some of the previous questions more deeply.

- a. Question 8: How many bits are in the status word?
- b. Question 8: How many bits are in the control word?
- c. Question 8: What 2-line handshake role is the outside world playing?
- d. Question 7: List all the lines in the algorithm lines would you need to set the tri-state buffer into a high impedance state.
- e. Question 3: Starting in state A, what input would allow you to visit all four states?

In order to help assure the continued success of the Mines EE program, I would appreciate your evaluation of whether or not this course met its learning objectives. Your answers will have no effect on your grade, your answers will be kept anonymous. Please put a mark in the circle which reflects your opinion on how well each objective was met.

	Strong Agree	Agree	Neutral	Disagree	Strongly Disagree
I understand how to convert numbers from one base to another and how to add and subtract numbers represented in binary and 2's complement form.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I understand how to convert between a truth table, a circuit diagram, boolean expression and a word statement.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I understand how to simplify logic expression into SOP or POS minimal form with or without don't cares.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I understand how adders, comparators, multiplexers and decoders are built and how they operate.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I understand how D,T,SR,JK, latches, clock latches and flip flops are supposed to operate.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I understand how registers, shift registers, counters, tri-state logic and RAMs are built and how they should operate.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I understand how to design Finite State Machines using a Ones Hot encoding.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I understand how to implement complex digital systems using the datapath and control design approach.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>