Students have either already taken or started taking this quiz, so be careful about editing it. If you change any quiz questions in a significant way, you may want to consider regrading students who took the old version of the quiz.

Points 39 Published

☐ Show Question Details

Question 1 pts

The answers I give on this exam represent my own work and I have not received from or given aid to another student during the course of this exam. Please upload a file containing your signature and student ID number.

FSM group Pick 1 questions, 10 pts per question

Question A

Consider the following FSM:

It has 2 inputs: w1 and w2. On each arc, the input values are labelled in the format w1w2. It produces a 1 bit output z.

A) Using the following state assignment:
$y_6y_5y_4y_3y_2y_1 = 000001\ (A),\ 000010\ (B),\ 000100\ (C),\ 001000\ (D),\ 010000\ (E),\ 100000\ (F)$

Derive expressions for

$Y_6 = y_2(w_1\bar{w}_2 + w_1w_2) + y_6(w_1\bar{w}_2 + w_1w_2) = y_2w_1 + y_6w_1$

$Y_3 = y_1(\bar{w}_1\bar{w}_2 + \bar{w}_1w_2) + y_6(\bar{w}_1\bar{w}_2 + \bar{w}_1w_2) = y_1\bar{w}_1 + y_6\bar{w}_1$

B) Now, using the following state assignment:

$y_3y_2y_1 = 000\ (A),\ 001\ (B),\ 010\ (C),\ 011\ (D),\ 100\ (E),\ 101\ (F)$

Generate the state table (given below). You may fill in each box with the corresponding state letter (you don't need to write the binary encoding).

PS ($y_3y_2y_1$)	NS ($Y_3Y_2Y_1$)				z
	w_1w_2=00	w_1w_2=01	w_1w_2=10	w_1w_2=11	
000	C	C	B	B	0
001	D	D	F	F	0
010	E	E	B	B	0
011	E	E	B	B	1
100	E	E	B	B	1
101	C	C	F	F	1

Question B

Consider the following FSM:

It has 2 inputs: w1 and w2. On each arc, the input values are labelled in the format w1w2. It produces a 1 bit output z.

A) Using the following state assignment:
$y_6y_5y_4y_3y_2y_1 = 000001\ (A),\ 000010\ (B),\ 000100\ (C),\ 001000\ (D),\ 010000\ (E),\ 100000\ (F)$

Derive expressions for

$Y_6 = y_3(w_1\bar{w}_2 + w_1w_2) + y_6(w_1\bar{w}_2 + w_1w_2) = y_3w_1 + y_6w_1$

$Y_3 = y_1(w_1\bar{w}_2 + w_1w_2) + y_2(w_1\bar{w}_2 + w_1w_2) + y_4(w_1\bar{w}_2 + w_1w_2) + y_5(w_1\bar{w}_2 + w_1\bar{w}_2) = w_1(y_1 + y_2 + y_4 + y_5)$

B) Now, using the following state assignment:

$y_3y_2y_1 = 000\ (A),\ 001\ (B),\ 010\ (C),\ 011\ (D),\ 100\ (E),\ 101\ (F)$

Generate the state table (given below). You may fill in each box with the corresponding state letter (you don't need to write the binary encoding).

PS ($y_3y_2y_1$)	NS ($Y_3Y_2Y_1$)				z
	w_1w_2=00	w_1w_2=01	w_1w_2=10	w_1w_2=11	
000	B	B	C	C	0
001	D	D	C	C	0
010	E	E	F	F	0
011	D	D	C	C	1
100	D	D	C	C	1
101	B	B	F	F	1

Question C

Consider the following FSM:

It has 2 inputs: w1 and w2. On each arc, the input values are labelled in the format w1w2. It produces a 1 bit output z.

A) Using the following state assignment:
$y_6y_5y_4y_3y_2y_1 = 000001\ (A),\ 000010\ (B),\ 000100\ (C),\ 001000\ (D),\ 010000\ (E),\ 100000\ (F)$

Derive expressions for

$Y_5 = y_3(\bar{w}_1\bar{w}_2 + \bar{w}_1w_2) + y_4(\bar{w}_1\bar{w}_2 + \bar{w}_1w_2) + y_5(\bar{w}_1\bar{w}_2 + \bar{w}_1w_2) = \bar{w}_1(y_3 + y_4 + y_5)$

$Y_2 = y_1(w_1\bar{w}_2 + w_1w_2) + y_3(w_1\bar{w}_2 + w_1w_2) + y_4(w_1\bar{w}_2 + w_1w_2) + y_5(w_1\bar{w}_2 + w_1w_2) = w_1(y_1 + y_3 + y_4 + y_5)$

B) Now, using the following state assignment:

$y_3y_2y_1 = 000\ (A),\ 001\ (B),\ 010\ (C),\ 011\ (D),\ 100\ (E),\ 101\ (F)$

Generate the state table (given below). You may fill in each box with the corresponding state letter (you don't need to write the binary encoding).

PS ($y_3y_2y_1$)	NS ($Y_3Y_2Y_1$)				z
	w_1w_2=00	w_1w_2=01	w_1w_2=10	w_1w_2=11	
000	C	C	B	B	0
001	D	D	F	F	0
010	E	E	B	B	0
011	E	E	B	B	1
100	E	E	B	B	1
101	C	C	F	P	1

Question D

Consider the following FSM:

It has 2 inputs: w1 and w2. On each arc, the input values are labelled in the format w1w2. It produces a 1 bit output z.

A) Using the following state assignment:
$y_6y_5y_4y_3y_2y_1 = 000001\ (A),\ 000010\ (B),\ 000100\ (C),\ 001000\ (D),\ 010000\ (E),\ 100000\ (F)$

Derive expressions for

$Y_5 = y_3(\bar{w}_1\bar{w}_2 + \bar{w}_1w_2) = y_3\bar{w}_1$

$Y_2 = y_1(\bar{w}_1\bar{w}_2 + \bar{w}_1w_2) + y_4(\bar{w}_1\bar{w}_2 + \bar{w}_1w_2) = \bar{w}_1(y_1 + y_4)$

B) Now, using the following state assignment:

$y_3y_2y_1 = 000\ (A),\ 001\ (B),\ 010\ (C),\ 011\ (D),\ 100\ (E),\ 101\ (F)$

Generate the state table (given below). You may fill in each box with the corresponding state letter (you don't need to write the binary encoding).

PS ($y_3y_2y_1$)	NS ($Y_3Y_2Y_1$)				z
	w_1w_2=00	w_1w_2=01	w_1w_2=10	w_1w_2=11	
000	B	B	C	C	0
001	D	D	C	C	0
010	E	E	F	F	0
011	D	D	C	C	1
100	D	D	C	C	1
101	B	B	F	F	1

Simple Processor Pick 1 questions, 8 pts per question

Question

Consider the simple processor introduced in lecture:

You are asked to add a new instruction to the 4 instructions that the processor already has (Load, Mov, Add, Sub). Adding a new instruction requires 3 opcode bits instead of the original 2 as well as some additional modifications.

The new instruction is LoadAndInc Rx, data.

This instruction loads external data and increments that data by 1. The incremented data is stored in Rx.

A) Briefly describe any new hardware that you would need to add to the datapath. Be clear about where in the datapath you would place any new hardware.

Add mux b/t A & Adder w/ input 1

B) Fill in the table below to specify the value of each control signal for each cycle of the new instruction. Assume the instruction is being executed as:

LoadAndInc R2, data

Also include in the table any new control signals your design requires. The table may have more rows and columns than necessary to answer the question. Please include any assumptions you make below your table.

	Cycle 1	2		
Existing Control Signals				
Extern	1	0		
$R_{in}[3:0]$	0000	0100		
$R_{out}[3:0]$	0000	0000		
A_{in}	0	0		
G_{in}	1	0		
G_{out}	0	1		
Done	0	1		
Add/Sub	0	X		
New Control Signals				
new Mux	1	X		

// 0 = Add

// 0 - sel A
1 - sel 1

Assembly Language Pick 1 questions, 6 pts per question

Question A

Consider the following ARM assembly language program:

```
_start: mov r0, #0x80
        lsl r1, r0, #8
        mov r2, #0xab
        orr r3, r1, r2
        str r3, [r0, #0]
        strb r3, [r0, #3]
        ldrsh r4, [r0, #2]
        ldrb r5, [r0, #1]
        cmp r5, r0
        bne else
        eor r6, r4, r3
        b after
else:   eor r6, r3, r0
after:  ror r7, r2, #3
end:    b end
```

Give the full hexadecimal values present in r0 = [r0], r1 = [r1], r3 = [r3], r4 = [r4], r5 = [r5], r6 = [r6] when the processor reaches the "end: b end" instruction. If the value is unknown, write "unknown". If you use any letters in your answer, please use lower case letters (e.g., x, a-f). You may upload any steps or work done to solve this question in the next question.

Question B

Consider the following ARM assembly language program:

```
_start: mov r0, #0x80
        lsl r1, r0, #8
        mov r2, #0xab
        orr r3, r1, r2
        str r3, [r0, #0]
        strb r3, [r0, #3]
        ldrh r4, [r0, #2]
        ldrsb r5, [r0, #1]
        cmp r5, r0
        bne else
        eor r6, r4, r3
        b after
else:   eor r6, r3, r0
after:  ror r7, r2, #3
end:    b end
```

Give the full hexadecimal values present in r0 = [r0], r1 = [r1], r3 = [r3], r4 = [r4], r5 = [r5], r6 = [r6] when the processor reaches the "end: b end" instruction. If the value is unknown, write "unknown". If you use any letters in your answer, please use lower case letters (e.g., x, a-f). You may upload any steps or work done to solve this question in the next question.

Question 0 pts

Use this question to upload any work done to solve the previous question. This is optional but may be used to award partial marks.

Group Pick 1 questions, 8 pts per question

Question A

Consider the following circuit with two inputs: reset and clock, and four outputs Q3,Q2,Q1,Q0. This circuit is a special type of counter. Note: You can ignore the inverted Q outputs of the flip-flops.

Q_3	Q_2	Q_1	Q_0
0	0	0	1
0	0	1	0
0	1	0	0
1	0	0	0

a) Considering the timing diagram above, give the value of Q3, Q2, Q1, Q0 ***after*** each of the four rising clock edges shown.

b) Write a Verilog module that describes the behaviour of the counter. The module should begin as follows:

```
module onehotcount(reset, clock, Q);
  input reset, clock;
  output reg [3:0] Q;
```

always@(posedge clock)
begin
Q[0] <= reset | Q[3]
Q[1] <= (~reset) & Q[0]
Q[2] <= (~reset) & Q[1]
Q[3] <= (~reset) & Q[2]
end
end module

Question B

Consider the following circuit with two inputs: reset and clock, and four outputs Q3,Q2,Q1,Q0. This circuit is a special type of counter. Note: You can ignore the inverted Q outputs of the flip-flops.

Q_3	Q_2	Q_1	Q_0
1	0	0	0
0	0	0	1
0	0	1	0
0	1	0	0

a) Considering the timing diagram above, give the value of Q3, Q2, Q1, Q0 ***after*** each of the four rising clock edges shown.

b) Write a Verilog module that describes the behaviour of the counter. The module should begin as follows:

```
module onehotcount(reset, clock, Q);
 input reset, clock;
 output reg [3:0] Q;
```

always@(posedge clock)
begin
Q[3] <= reset | Q[2]
Q[2] <= (~reset) & Q[1];
Q[1] <= (~reset) & Q[0];
Q[0] <= (~reset) & Q[3]
end module

Group Pick 1 questions, 6 pts per question

Question A

You are to design a circuit that uses three D-type flip-flops having outputs Q2, Q1, and Q0. The circuit has four inputs: count, resetn, w, and clock. It behaves as follows:

- When count is logic 0, the three flip-flops operate as a shift register. At each positive clock edge, the value of w is stored in the flip-flop whose output is Q2; the value of Q2 (before the positive edge) is stored in the flip-flop whose output is Q1; the value of Q1 is stored in the flip-flop whose output is Q0.
- When count is logic 1, the three flip-flops behave as an up-counter, where Q2 is the most-significant bit and Q0 is the least-significant bit.
- resetn is an active-low asynchronous reset.

Draw the circuit and upload your drawing. You may assume that D-type flip-flops with an active-low asynchronous reset are available; see the symbol below. The flip-flops are positive edge triggered. You may assume a 3-bit adder block is available; see the symbol below.

Question B

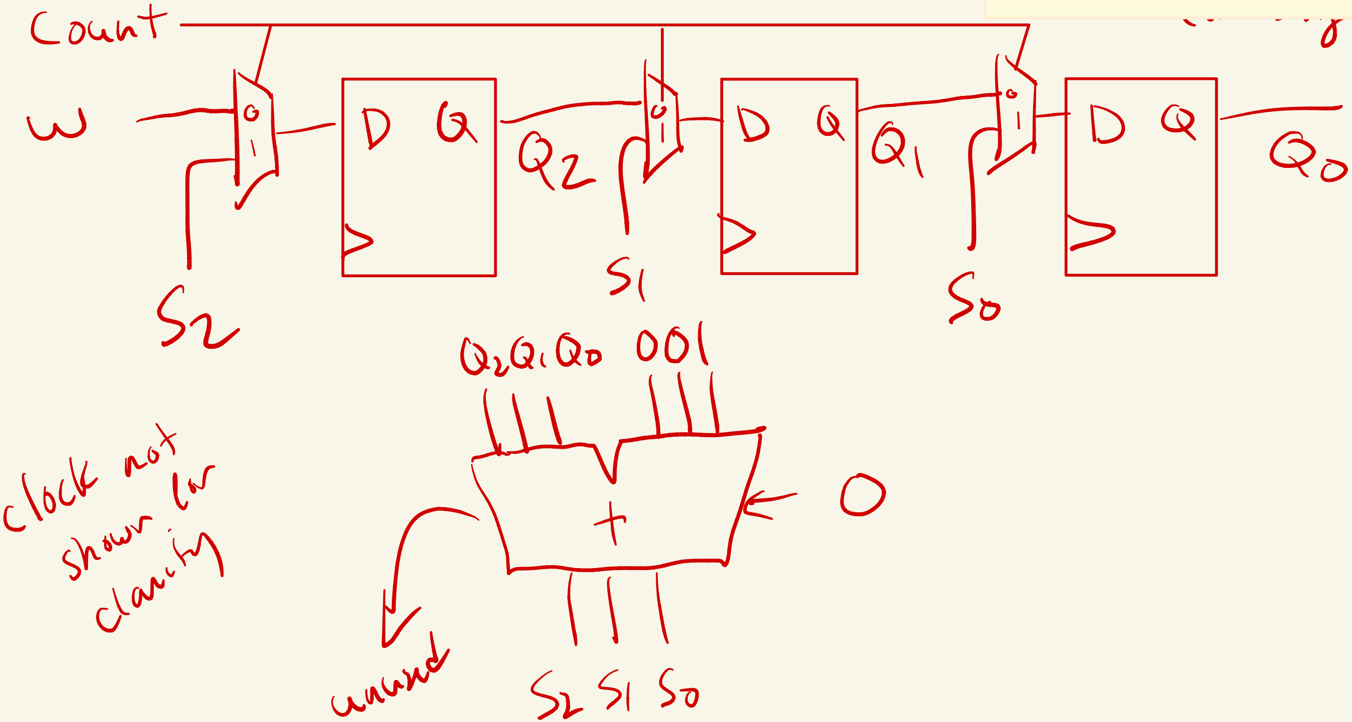
You are to design a circuit that uses three D-type flip-flops having outputs Q2, Q1, and Q0. The circuit has four inputs: shift, resetn, w, and clock. It behaves as follows:

- When shift is logic 1, the three flip-flops operate as a shift register. At each positive clock edge, the value of w is stored in the flip-flop whose output is Q2; the value of Q2 (before the positive edge) is stored in the flip-flop whose output is Q1; the value of Q1 is stored in the flip-flop whose output is Q0.
- When shift is logic 0, the three flip-flops behave as an up-counter, where Q2 is the most-significant bit and Q0 is the least-significant bit.
- resetn is an active-low asynchronous reset.

Draw the circuit and upload your drawing. You may assume that D-type flip-flops with an active-low asynchronous reset are available; see the symbol below. The flip-flops are positive edge triggered. You may assume a 3-bit adder block is available; see the symbol below.

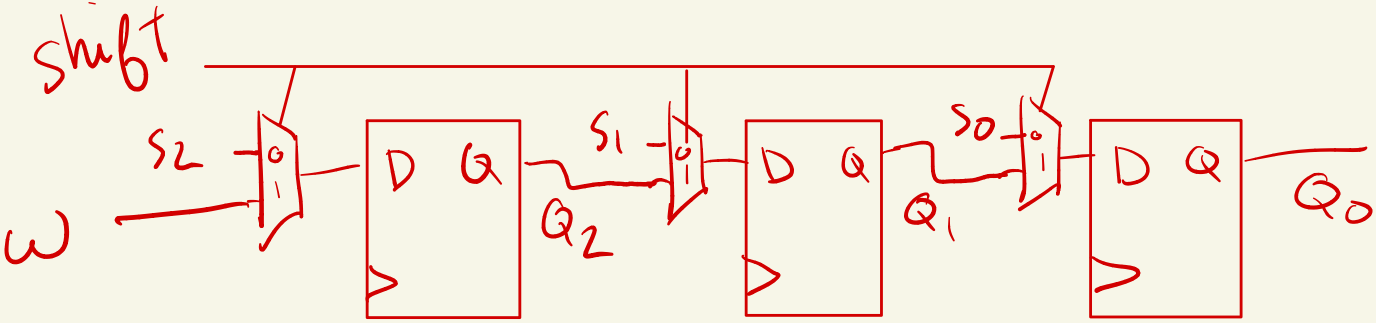
☐ Notify users this quiz has changed

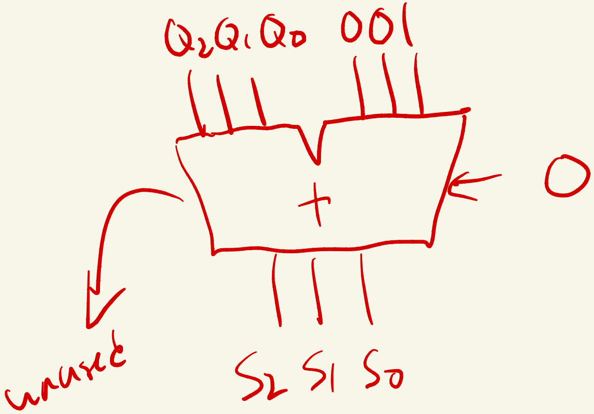
A



clock not shown for clarity

B





clock not shown for clarity