ECE 327 Solution to Midterm

2016t1 (Winter)

All requests for re-marks must be submitted in writing to Mark Aagaard before 8:30am on Friday March 11.

A random collection of midterms were scanned. Exams that are submitted for re-marking will be verified against this set.

		Total Marks	Approx. Time	Page
Q1	VHDL Semantics	25	15	2
Q2	DFD	25	20	4
Q3	Control Circuitry	25	15	8
Q4	FSM	25	20	9
Totals		100	70	

Q1   (25 Marks) VHDL Semantics
(estimated time: 15 minutes)

For the VHDL program below, calculate the values for the signals c, d, e, and f at 15ns.

NOTES:

1. All of the processes are in the *same architecture*.

2. The signals a, b, c, d, e, and f are declared to be unsigned(15 downto 0).

3. **For full marks you must justify your answer, using text and/or the waveform diagram. You may, but are *not* required to, show a delta-cycle simulation.**

```
process begin
  clk1 <= '0';
  wait for 10 ns;
  clk1 <= '1';
  wait for 10 ns;
end process;

process begin
  a <= (others => '0');
  wait for 10 ns;
  a <= a +    10;
  a <= a +    20;
  wait until rising_edge( clk1 );
  a <= a +   100;
  a <= a +   200;
  wait until rising_edge( clk2 );
  a <= a +  1000;
  a <= a +  2000;
  wait until rising_edge( clk1 );
  a <= a + 10000;
  a <= a + 20000;
  wait;
end process;
```
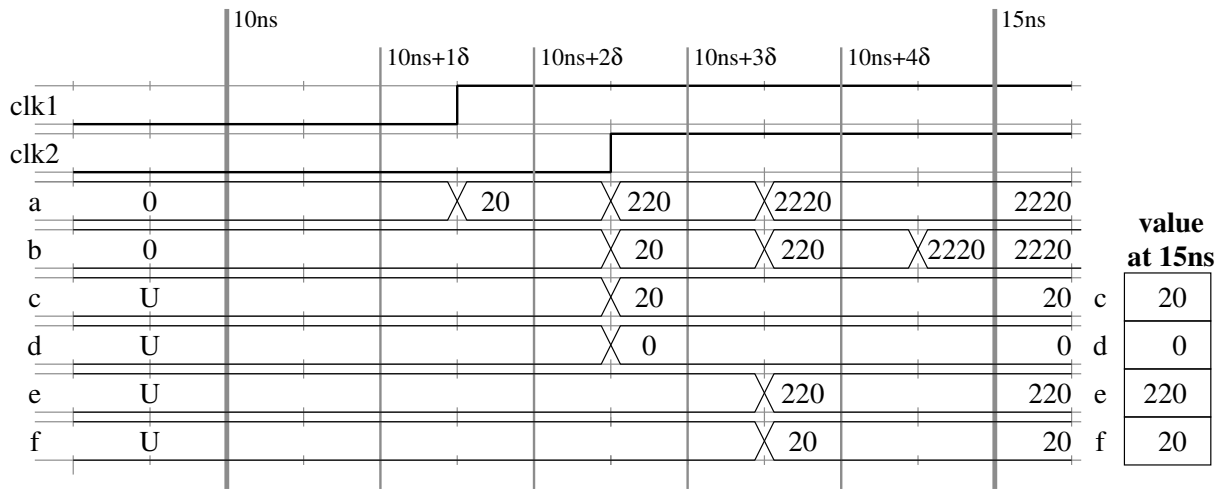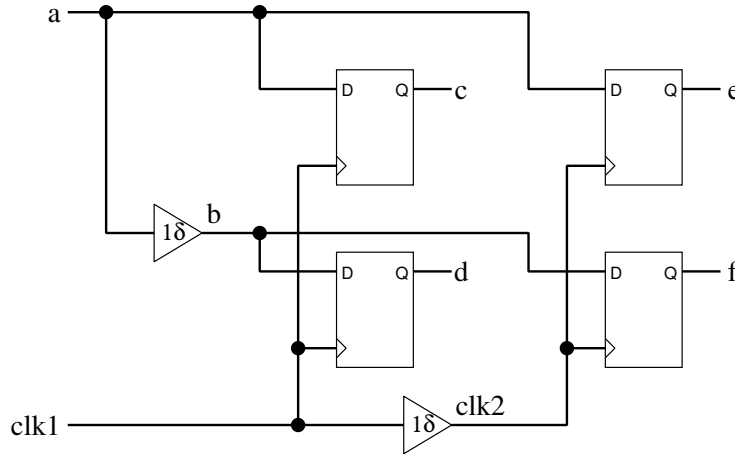
```
clk2 <= clk1;
b   <= a;

process begin
  wait until rising_edge( clk1 );
  c <= a;
  d <= b;
end process;

process begin
  wait until rising_edge( clk2 );
  e <= a;
  f <= b;
end process;
```

Answer:

The schematic is not a necessary part of the answer.

No changes from end of 10ns + 3δ simulation cycle to 15 ns

Marking:

    +1 mark *if answered completely with correct justification*

    -50% *of earned mark if insufficient or incorrect justification*

c

    6 marks 20

    4 marks 220

    3 marks 0

    2 marks 2200

d

    6 marks 0

    3 marks 20

    2 marks 220,   2200

e

    6 marks 220

    4 marks 130,   300,   330

    3 marks 20

    2 marks 2200

f

    6 marks 20

    4 marks 20,   220

    3 marks 0

    2 marks 130,   330,   2200

Q2   (25 Marks) DFD

(*estimated time: 20 minutes*)

Beginning with the data-dependency graph below, design a dataflow diagram, and then perform allocation.
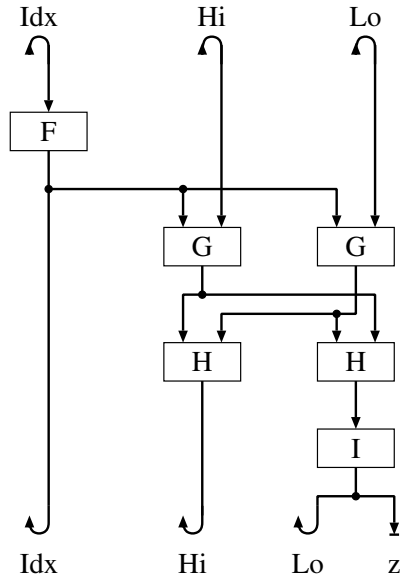
NOTES:

1. Reset initializes Idx, Hi, and Lo. Your dataflow diagram shall *not* show this initialization.

2. Outputs shall be *registered*

3. Optimization goals in order of decreasing importance:

   (a) minimize *area*

       i. number of F components

       ii. number of G components

       iii. number of H components

       iv. number of I components

       v. number of registers

       vi. number of multiplexers

       vii. number of chip-enables

   (b) minimize *latency*

   (c) minimize *clock period*

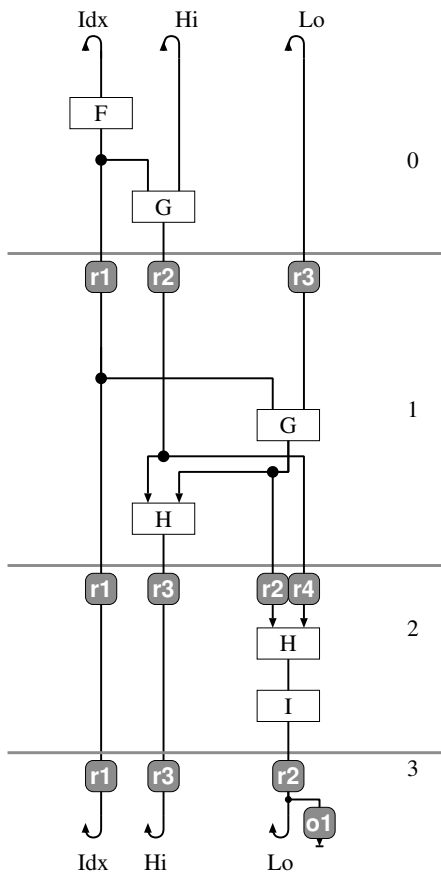4. You shall *not* use any algebraic optimizations,

max possible: 25 marks

Answer:
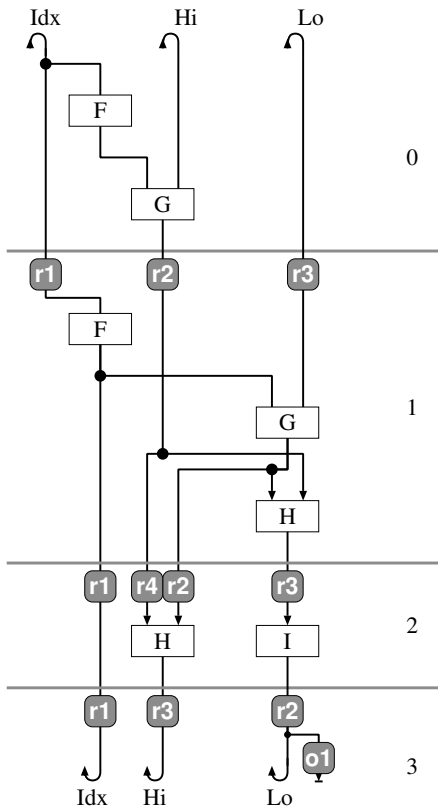
	F	G		H		I	r1	r2	r3	r4
		s1	s2	s1	s2					
0	r1	F	r3	—	—	—	0	G	r2	—
1	r1	F	r3	G	r2	—	F	G	H	r2
2	—	—	—	r4	r2	r3	0	I	H	—
muxes			1				1	1		3
ce							1			1

Latency	3
Clock period	Flop + Max(F+G+H, I)
Number of F	1
Number of G	1
Number of H	1
Number of I	1
Number of regs	4
Number of 2:1 muxes	3
Number of chip-enables	1

max possible: 23 marks

	F	G		H		I	r1	r2	r3	r4	
		s1	s2	s1	s2						
0	r1	F	r3	—	—	—	F	G	r2	—	
1	—	r1	r3	r2	G	—	0	G	H	r2	
2	—	—	—	r2	r4	H	0	I	0	—	
muxes		1		1			1	1			4
ce							1		1		2

Latency	3
Clock period	Flop + Max(F+G, G+H, H+I)
Number of F	1
Number of G	1
Number of H	1
Number of I	1
Number of regs	4
Number of 2:1 muxes	4
Number of chip-enables	2

Idx Hi Lo

F

G

0

r1 r2 r3

G

1

H

r1 r3 r2 r4

H

2

I

r1 r3 r2 o1

3

Idx Hi Lo

max possible: 21 marks

	F	G s1 s2	H s1 s2	I	r1	r2	r3	r4
0	r1	F r3	— —	—	F	G	r2	—
1	—	r1 r3	r2 G	—	0	0	H	G
2	—	— —	r2 r4	—	0	H	0	—
3	—	— —	— —	r2	0	I	0	—
muxes		1	1			2	1	5
ce					1	1	1	3

Latency	4
Clock period	Flop + Max(F+G, G+H, I)
Number of F	1
Number of G	1
Number of H	1
Number of I	1
Number of regs	4
Number of 2:1 muxes	5
Number of chip-enables	3

max possible: 18 marks
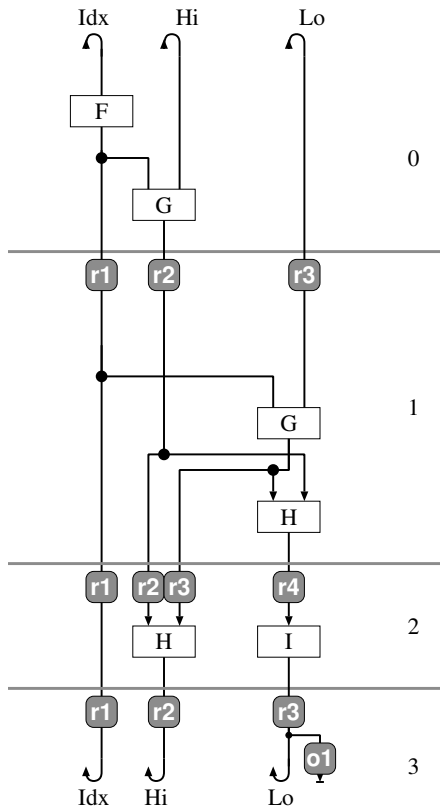
	F	G s1 s2	H s1 s2	I	r1	r2	r3	r4
0	r1	F r2	— —	—	F	G	0	—
1	—	r1 r3	G r2	—	0	0	G	H
2	—	— —	r2 r3	r4	0	H	I	—
muxes		1 1	1 1			1	1	6
ce					1	1	1	3

Latency	3
Clock period	Flop + Max(F+G, G+H, I)
Number of F	1
Number of G	1
Number of H	1
Number of I	1
Number of regs	4
Number of 2:1 muxes	6
Number of chip-enables	3

Marking:

7 marks   *DFD is syntactically and functionally correct*

- *all inter-parcel variables at top of DFD*
- *all inter-parcel variables at bottom of DFD*
- *registered inter-parcel variables*
- *output*
- *correct F, G, H, I connections*

15 marks   *Scheduling and allocation*

- *one instance of each of F, G, H, I*
- *4 registers*
- *register allocation to minimize muxes*
- *register allocation to minimize chip-enables*
- *latency*
- *clock period*

3 marks   *Analysis*

Q3   (25 Marks) Control Circuitry

(*estimated time: 15 minutes*)

Just as another freezing rain storm descends upon Ontario, your manager is called away to an urgent business meeting in Tahiti. You have been given the task of finishing the allocation and the schematic for the dataflow diagram shown below.

NOTES:

1. You shall *not* make any changes to the dataflow diagram, except to complete the allocation.

2. The system shall use an ASAP parcel schedule.

3. The variable P shall be initialized to 3.

4. The schematic shall include the control circuitry and the datapath.

5. The control circuitry shall be drawn using Boolean gates and flip-flops (*i.e.*, you may *not* draw a cloud).

6. In the schematics, the label of a datapath component is used as the name of the component's output signal.

7. In the schematic, label all signals to match the dataflow diagram.

8. You shall *not* draw the clock signal.

Finish the allocation

Answer is in black.
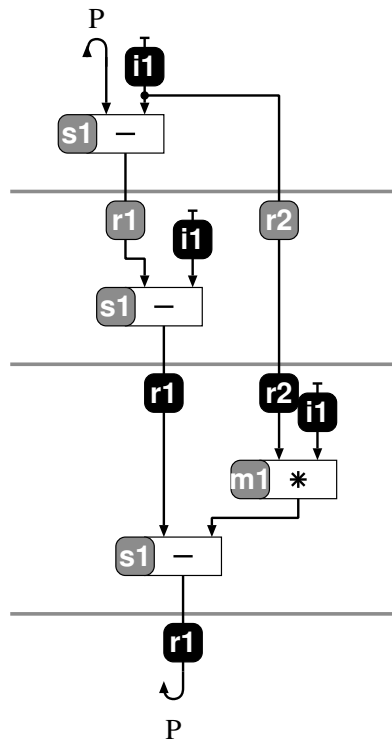
Draw the schematic
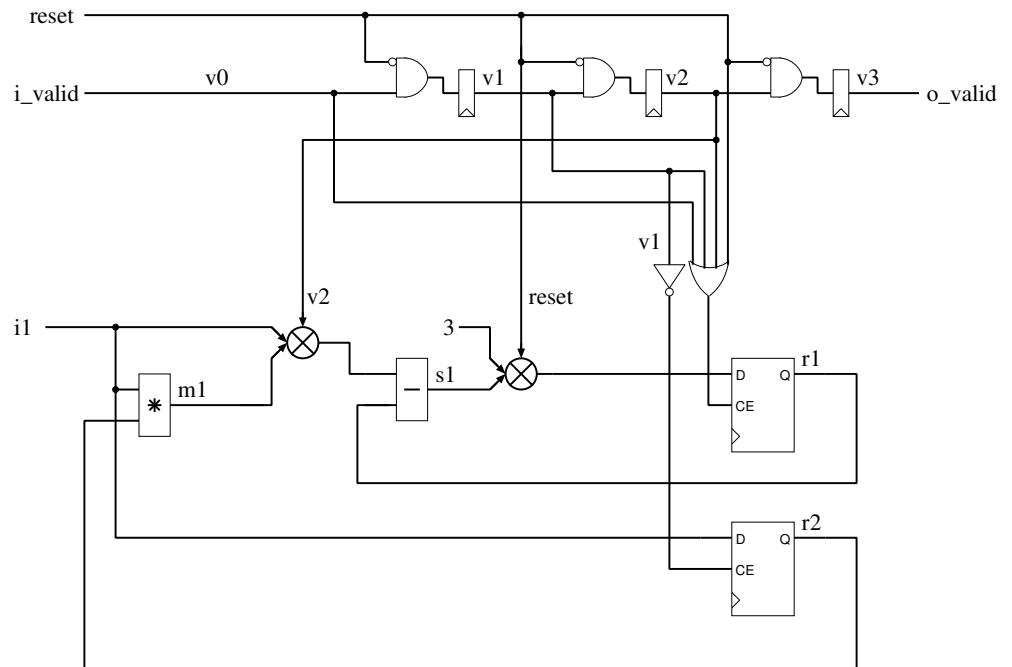
+1 mark i1

+1 mark r1 and r2 in cycle #2

+1 mark r1 in cycle #3

+1 mark no additional allocations

+3 marks i_valid and o_valid

+3 marks valid-bit chain of flip flops

+3 marks valid-bits are reset

+3 marks r1 initialization

+3 marks r1 chip-enable='0' in idle state

+3 marks r2 chip-enable='0' in v1

+3 marks s1 multiplexer

-1 mark each mistake

Q4   (25 Marks) FSM

(estimated time: 20 minutes)

Each of the three state machines on the next page is intended to meet the system description below:

1. The inputs to the system are: start, stop, and a. The output is z.

2. In the first clock cycle, start and stop are both guaranteed to be '0'.

3. The counting shall begin in the clock cycle *after* start='1'.

4. The counting shall continue up to, but *not* including, the next clock cycle in which stop='1'.

5. The current values of start, stop, and a shall affect z in the *next* clock cycle.

6. In the clock cycle *after* stop='1', the output z shall be set to 0 and shall remain 0 until the next sequence of counting begins.

7. The system may put a constraint on the inputs such that there is a *minimum gap* of clock cycles between stop='1' and the next start='1'. Each state machine may have its own value for the minimum gap. The minimum gap may be as small as 0 clock cycles.
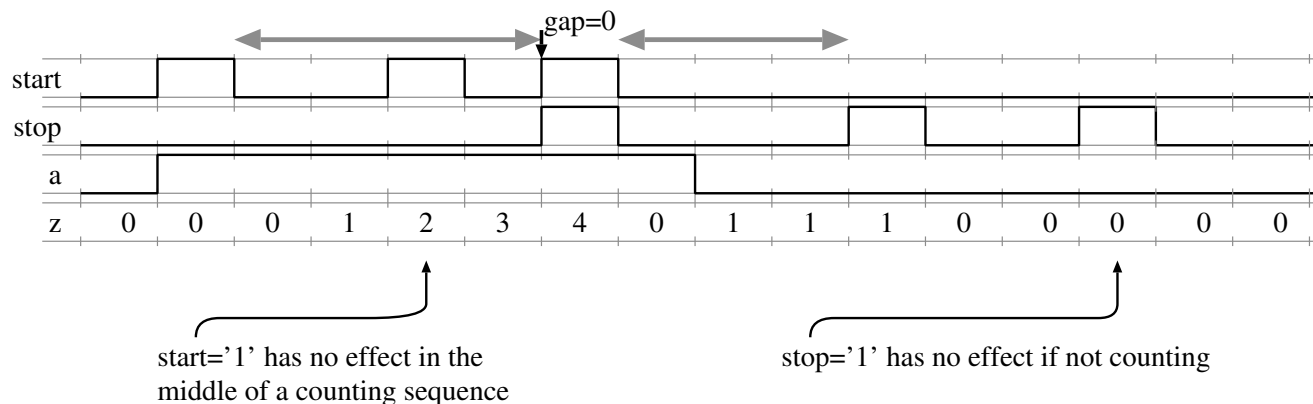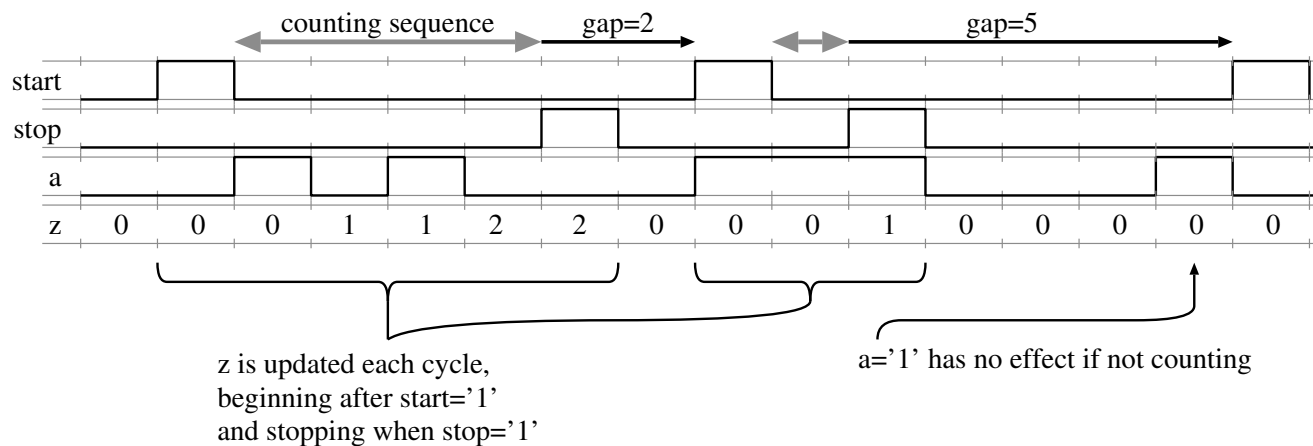

NOTES:

1. If the state machine is correct, answer what the minimum gap value is.

2. If the state machine is incorrect, explain either how the machines behaviour differs from the system description or how the state machine could be modified to fix the incorrect behaviour.

Q4a

Answer:
   Incorrect.

   Explanation: When $a='1'$, the output z is updated in the current *clock cycle. It should be updated in the next clock cycle.*
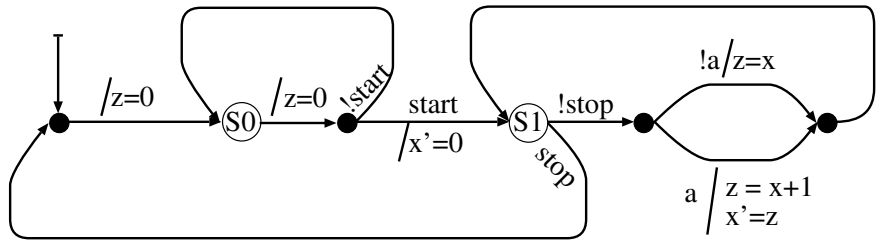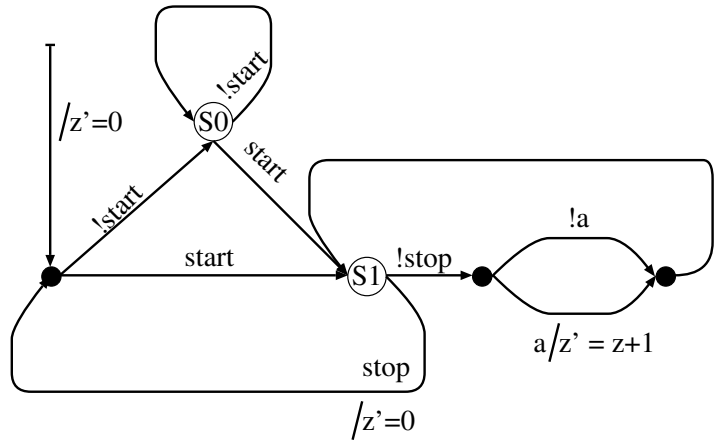

Q4b

Answer:
   Correct. Gap=0.

   Explanation: z is initialized in the first clock cycle and when $stop='1'$. When $stop='1'$, $start$ is sampled in the same clock cycle.
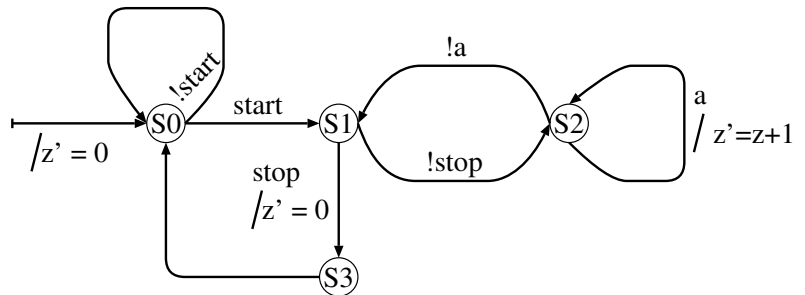

Q4c

Answer:
   Incorrect.

   Explanation (required): The input a is not sampled every clock cycle during the counting sequence.

Marking:

Fsm is correct, student says correct

+3 marks *Correct*

+5 marks *Gap*

5 marks   *correct value*

3 marks   *Off by one*

1 mark   *Off by two*

Fsm is correct, student says buggy

5 marks   *Perceptive observation with correct information*

3 marks   *Significant correct information*

1 mark   *Some correct information*

Fsm is buggy, student says buggy

+3 marks *Buggy*

+5 marks *Justification*

5 marks   *Justification is correct and well written.*

4 marks   *Justification is technically correct, but poorly written.*

3 marks   *Justification is almost correct.*

2 marks   *Significant amount of correct information.*

FSM is buggy, student says correct

5 marks   *Gap is correct value*

3 marks   *Gap is off by one*

1 mark   *Gap is off by two*

+1 mark   *Answered all three questions completely.*