First Name	Last Name	First letter of last name

UW Userid

ECE 327 Midterm

2016t1 (Winter)

Instructions and General Information

- 100 marks total
- Time limit: 1 hour and 20 minutes (80 minutes)
- No books, no notes, no computers. Calculators are allowed
- If you need extra paper, request some from a proctor.
- Write neatly.
- To earn part marks, you must show the formulas you use and all of your work.
- **The proctors and instructors will not answer questions, except in cases where an error on the exam is suspected. If you are confused about a question, write down your assumptions or interpretation.**
- **Justifications of answers will be marked according to correctness, clarity, and concision.**

			Total Marks	Approx. Time	Page
Q0	!!Almost Free!!		2	2	2
Q1	VHDL Semantics		25	15	3
Q2	DFD		25	20	4
Q3	Control Circuitry		25	15	6
Q4	FSM		25	20	8
Totals			100	72	

Q0 (2 Marks) !!Almost Free!!

(*estimated time: 2 minutes*)

Q0a (1 Mark) Best part

What is the best part of the course?

Q0b (1 Mark) Most improve

What one thing could be done to most improve the course for the remainder of the term?

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;
```

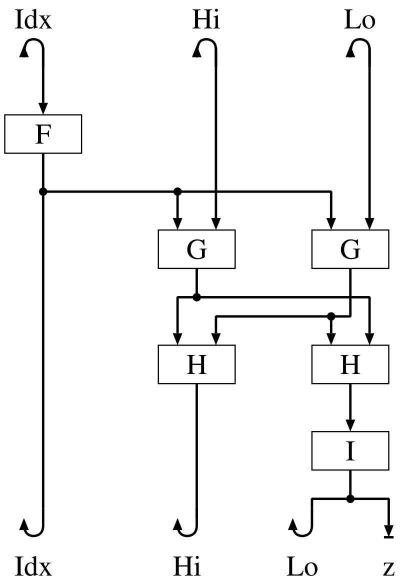
```
clk2 <= clk1;
b    <= a;

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;
```

	value at 15ns
c	
d	
e	
f	

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,

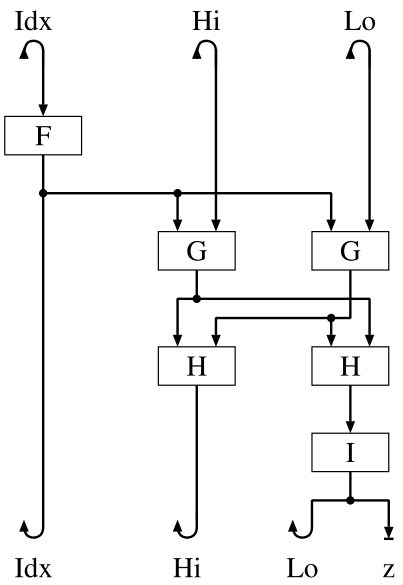
Use this space for your final, neat copy of the DFD

Analysis:

Latency:	
Clock period:	
Number of F:	
Number of G:	
Number of H:	
Number of I:	
Number of regs:	
Number of 2:1 muxes:	
Number of chip-enables:	

The next page is for scratch work

Scratch work



Q3 (25 Marks) Control Circuitry

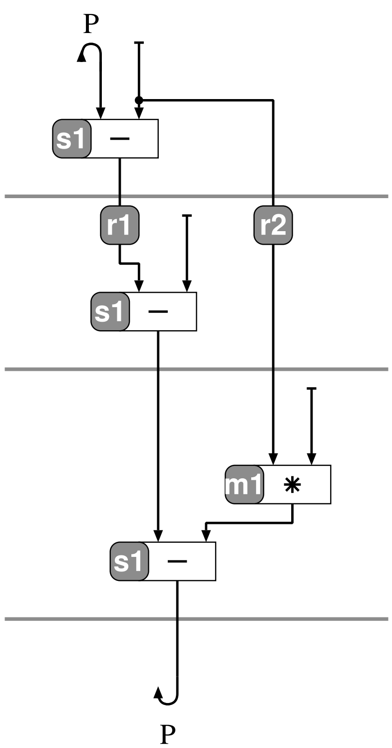
(*estimated time: 15 minutes*)

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.

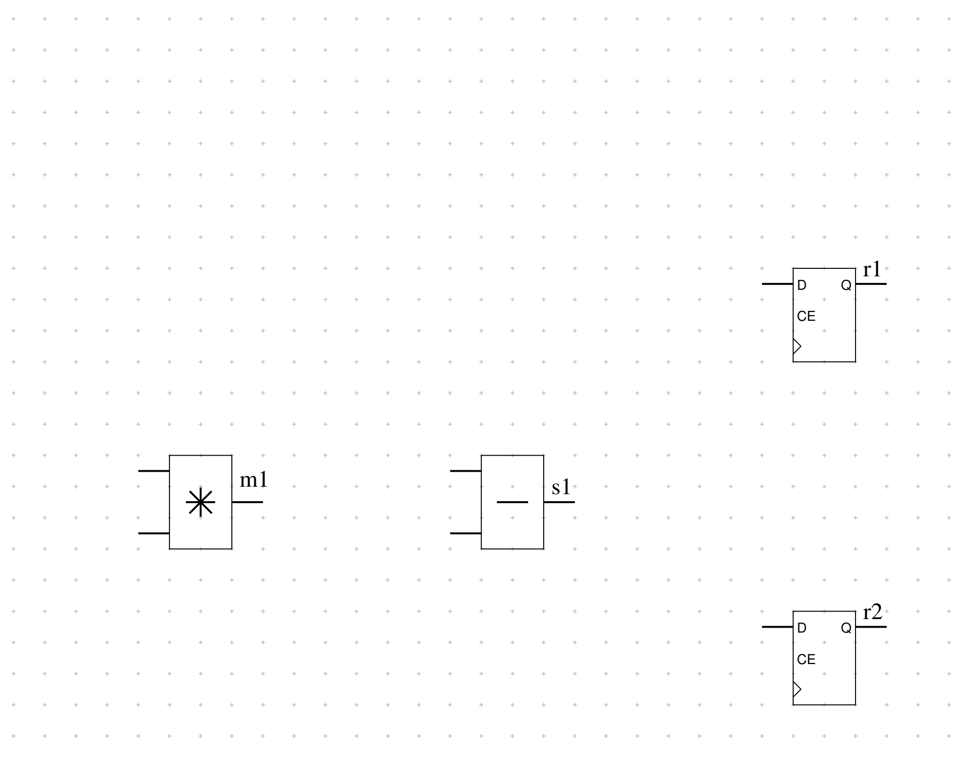
NOTES:

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

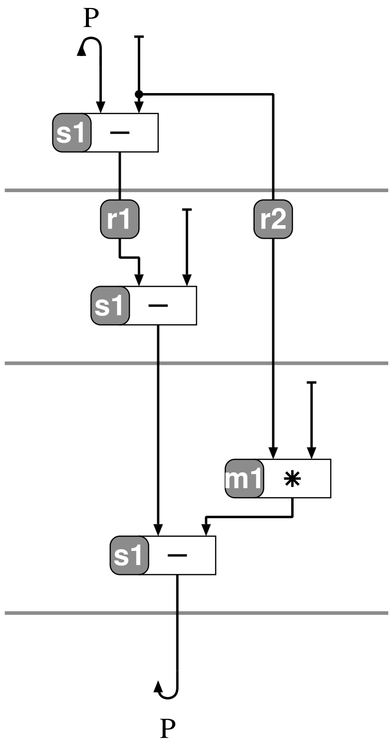


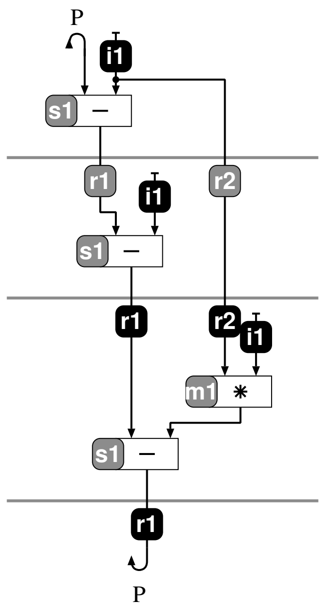
Draw the schematic

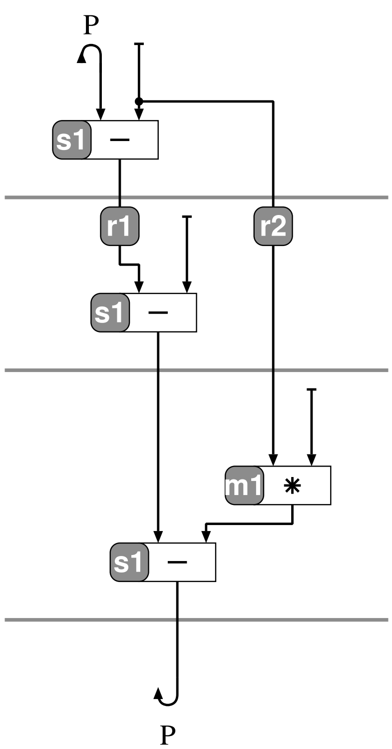


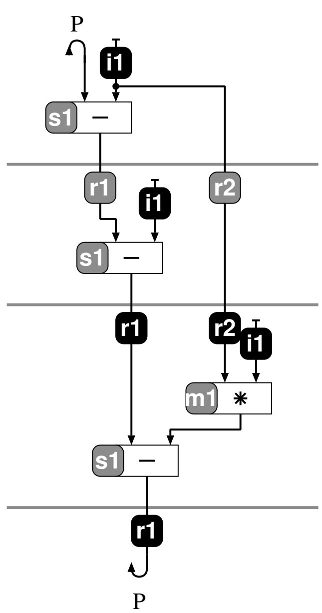
The next page is available for scratch work

Scratch work







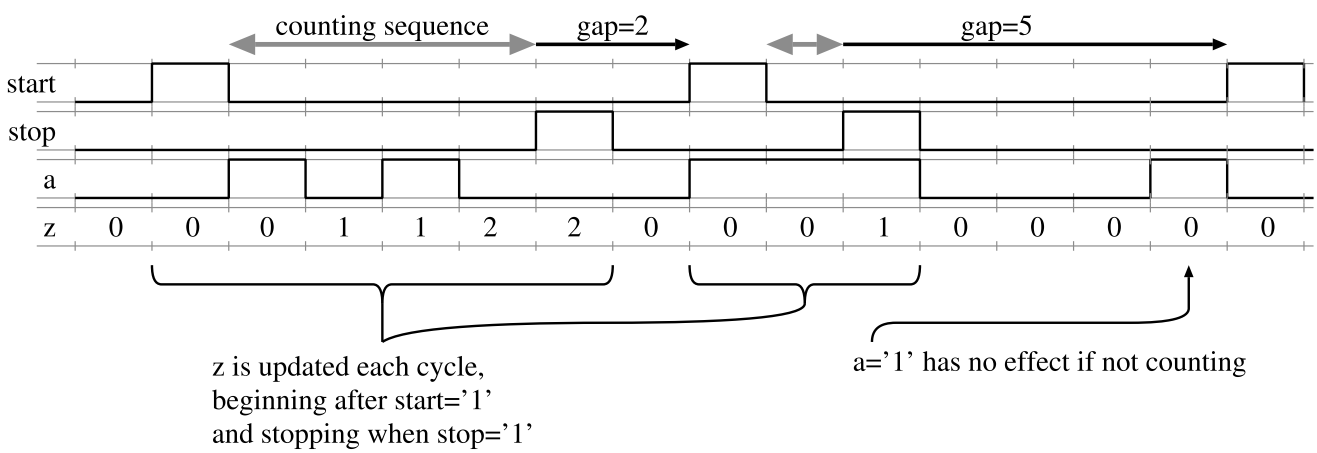


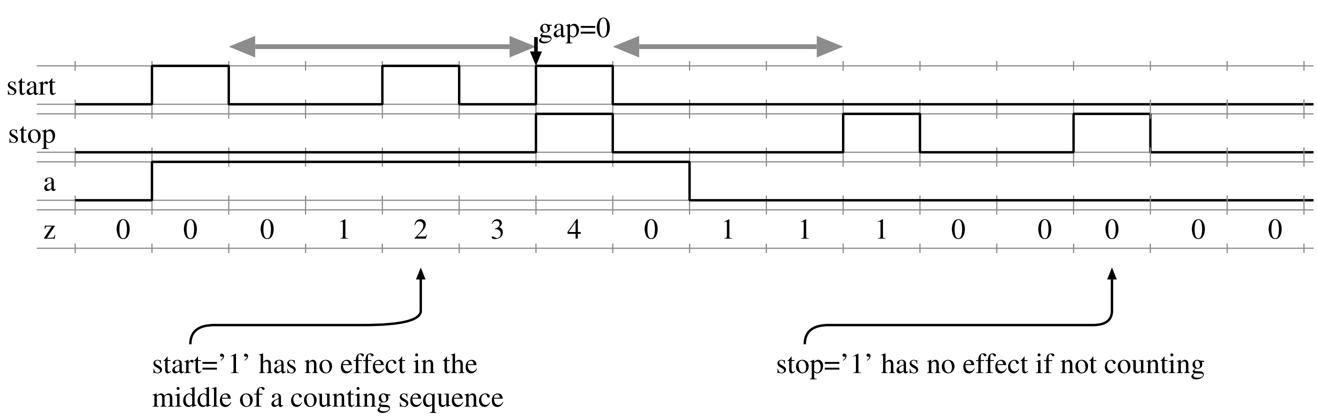
Q4 (25 Marks) FSM

(*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.

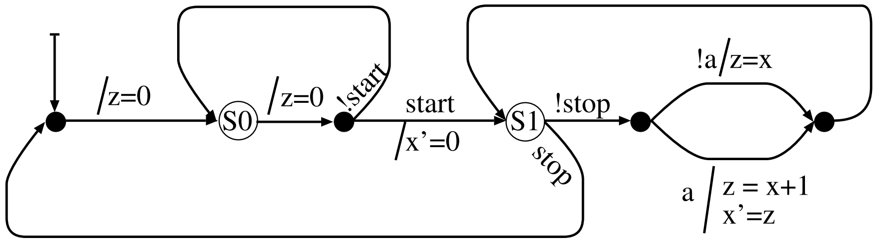




NOTES:

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



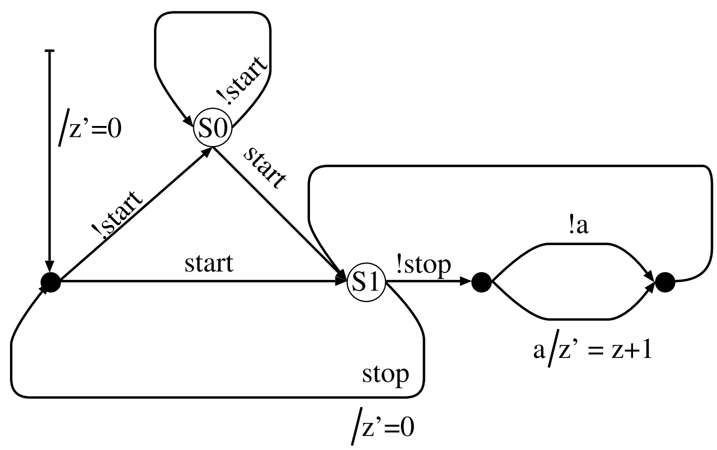
Correct Yes ☐ No ☐

If correct:

minimum gap = ☐

If incorrect, explanation: ____________________

Q4b



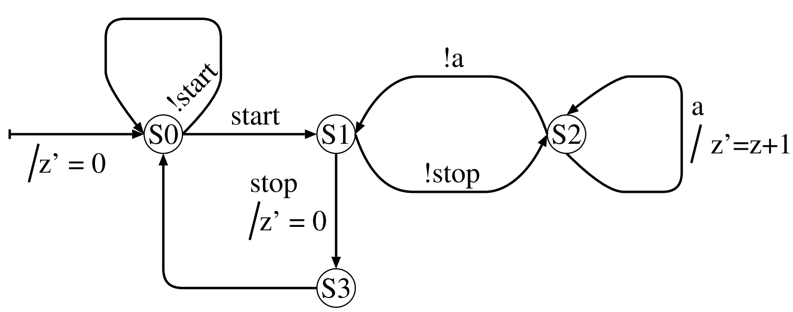
Correct Yes ☐ No ☐

If correct:

minimum gap = ☐

If incorrect, explanation: ____________________

Q4c



Correct Yes ☐ No ☐

If correct:

minimum gap = ☐

If incorrect, explanation: ____________________