CSE 140 Midterm 3 version B
Tajana Simunic Rosing
Fall 2015

Name of the person on your left: ________________________________
Name of the person on your right: ________________________________

- Do not start the exam until you are told.
- Write your name and PID at the top of every page. Write the names of people on your left and right on the first page.
- Turn off and put away all your electronics. This is a closed-book, closed-notes. You may only refer to one 8 ½ x 11” page of your handwritten notes.
- By turning in this exam for grading you are stating that you have followed the UCSD’s academic honesty policies. Do not look at anyone else’s exam or talk to anyone but an exam proctor.
- If you have a question, raise your hand and an exam proctor will come to you.
- You have 80 minutes to finish the exam. When the time is finished, you must stop writing.

Write your answers in the space provided. To get the most partial credit, clearly show all the steps of your work. Full credit may not be given for correct answers with no work shown.
Problem 1 (15pts)
Design a Mealy detector that generates a 1 on its $W$ output when the sequence of $0101$, $0100$, or an overlap of the two has been detected on its $A$ input. Draw the state diagram (FSM) using the minimum number of states.
For example:
A 0101000011
W 0001010000
Problem 2 (15pts)
Fill out the missing components on the circuit and in the description table so that the ALU works correctly.

<table>
<thead>
<tr>
<th>S1S0</th>
<th>Y</th>
<th>label1=</th>
<th>label2=</th>
<th>label3=</th>
<th>label4=</th>
<th>label5=</th>
<th>operation1=</th>
<th>operation2=</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>A-B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>A+B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>operation1=</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>operation2=</td>
<td></td>
</tr>
</tbody>
</table>

\[ A \rightarrow S1 \]
Problem 3 (15pts)
Consider the circuit given below. Assume that all components have near zero delay.

Draw the waveforms for W, X, Y, Z and OUT, given waveforms for A and B.
Problem 4 (15pts) You are given the RTL circuit below with Full-Adder (FA), register built from D flip-flops, mux and a decoder. The propagation delays of individual components are:
- NOT gate = 1ns
- All other logic gates = 2ns
- Multiplexer delay = 2ns
- Decoder delay = 3ns
- Clock->Q (D-FF propagation) delay = 2ns.
- The setup time for a flip-flop = 25ns. Assume zero clock skew.

a. Label the critical path.
b. What is the fastest clock frequency that this design can work at?
Problem 5 (20pts)
The state diagram of a sequential circuit is given below:

a. Fill out the following truth table and derive the Boolean equations for next state and output. Write the output for current state.
   Note: Qa/Qa+ is MSB and Qb/Qb+ is LSB of the state.

<table>
<thead>
<tr>
<th>Qa</th>
<th>Qb</th>
<th>Cnt</th>
<th>Qa+</th>
<th>Qb+</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

b. Write boolean equations for the next state and output:
   \[ Qa^+ = \]
   \[ Qb^+ = \]
   \[ Y = \]
c. Complete the following implementation by drawing the circuit to be placed in empty boxes Z1 and Z2.

![Circuit Diagram]

- Cnt
- Z1
- Z2
- Clk
- FFA
- FFB
- Y
Problem 6 (20pts)
Given the following C code that outputs the max value of an array,
Inputs: byte a[256], bit go
Outputs: byte max;
main:
  while(1) {
    while(!go);
    done = 0;
    i = 0;
    max= 0;
    while( i < 256 ) {
      if( a[i] > max ) {
        max = a[i];
      }
      i = i + 1;
    }
    done = 1;
  }

a. Convert it to an HLSM
b. Design the data path
This page was intentionally left blank.