CSE140 – Spring 2013

CSE140 Homework #8

You must *SHOW ALL STEPS* for obtaining the solution. Reporting the correct answer, without showing the work performed at each step will result in getting 0 points for that problem.

1. RTL Design:
   Exercise 5.10 from Vahid's book
   (a) Use the RTL design method of Table 5.1 to convert the high-level state machine in Figure 5.94 to a controller and a datapath. Design the datapath to structure, but design the controller to an FSM only.

   ![Controller FSM](image1)

   ![Datapath](image2)

2. RTL Design:
   Exercise 5.11 from Vahid's book

   ![State Machine](image3)
3. RTL Design:
Exercise 5.18 from Vahid's book
Using the RTL design method shown in Table 5.1, create an RTL design for a digital filter that outputs the average of the current 32-bit input and the previous 32-bit sample. *Hint:* You can use a right shift to implement the divide within your datapath.
4. RTL Design:
Convert the following C code, which calculates the number of values that are not equal to \( b \) are within an array \( A \) consisting of 128 8-bit values, into a high-level state machine.

Inputs: byte \( a[128] \), byte \( b \), bit go
Outputs: byte \( freq \), bit \( done \)

FREQUENCY:
while(1) {
    while(!go);
    done = 0;
    i = 0;
    freq = 0;
    while (I < 128) {
        if (a[i] != b) {
            freq = freq + 1;
        }
        i = i + 1;
    }
    done = 1;
}

Inputs: go (bit), a (128-byte memory), b (8 bits)
Outputs: done (bit), \( freq \) (8 bits)

5. ALU Design:
Design a 4-bit ALU with the following functional table:

<table>
<thead>
<tr>
<th>M1</th>
<th>M0</th>
<th>Function Name</th>
<th>( F(A,B) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>( A+B ) multiplied by 2</td>
<td>( 2*(A+B) )</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Increment A</td>
<td>( A+1 )</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Subtract B from A</td>
<td>( A-B )</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>If (A == B) output A, otherwise output B</td>
<td>If (A==B) then A, else B</td>
</tr>
</tbody>
</table>

A and B are two 4-bit binary numbers.
M1, M0 are the control inputs for the Arithmetic Unit.
Use a minimum number of Full Adders, Comparators, and Multiplexers.