CS 151
Midterm

Name : _________________, _________________
      (Last Name)            (First Name)
Student ID : _______________
Signature : _______________

Instructions:

1. Please verify that your paper contains **10 pages** including this cover.
2. Write down your Student-Id on the top of each page of this quiz.
3. This exam is **closed book**. No notes or other materials are permitted.
4. Total credits of this midterm are **55 points**.
5. To receive credit you must show your work clearly.
6. **For possible re-grade request make sure that your write clearly.**
7. Calculators are **NOT** allowed.
Q1: [ALU] [15 points]

We are going to design a 4-bit Arithmetic Unit (AU) 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>Add A and B</td>
<td>A+B</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Subtract B from A</td>
<td>A-B</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Decrement A</td>
<td>A-1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Add A and 2*B and 2</td>
<td>A+2*B+2</td>
</tr>
</tbody>
</table>

A and B are two 4-bit binary numbers \(a_3a_2a_1a_0\) and \(b_3b_2b_1b_0\). M1, M0 are the control inputs to this AU.

For doing this, the blocks labeled “Black Box” and CE (Carry Extender) in the following block diagram should be designed:
a. Fill the following table for \( y_3, y_2, y_1, y_0 \) and \( c_0 \) based on the inputs of the AU which are \( a_3, a_2, a_1, a_0, b_3, b_2, b_1, b_0, M_1 \) and \( M_0 \). [10 points]

<table>
<thead>
<tr>
<th>( M_1 )</th>
<th>( M_0 )</th>
<th>( y_3 )</th>
<th>( y_2 )</th>
<th>( y_1 )</th>
<th>( y_0 )</th>
<th>( c_0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>( b_3 )</td>
<td>( b_2 )</td>
<td>( b_1 )</td>
<td>( b_0 )</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>( b_3' )</td>
<td>( b_2' )</td>
<td>( b_1' )</td>
<td>( b_0' )</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>( b_2 )</td>
<td>( b_1 )</td>
<td>( b_0 )</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
b. Using the table that you reached in part (a), derive the logic equations for $y_1$, $y_0$ and $c_0$. [5 points]

$$y_1 = m_1'm_0'b_1 + m_1'm_0b_1' + m_1m_0' + m_1m_0b_0$$

$$y_0 = m_1'm_0'b_0 + m_1'm_0b_0' + m_1m_0' + m_1m_0$$

$$c_0 = m_1'm_0 + m_1m_0$$
Q2: [FSM Design] [15 points]

Design an FSM for a circuit which has a 2-bit input X and an output Y. The output Y becomes 1 if the cumulative sum of the numbers in sequence X multiple of 3. Otherwise Y is 0.

For example:

\[
\begin{align*}
X: & \ 00 \rightarrow 11 \rightarrow 01 \rightarrow 10 \rightarrow 00 \rightarrow 01 \ldots \\
Y: & \ 1 \rightarrow 1 \rightarrow 0 \rightarrow 1 \rightarrow 1 \rightarrow 0 \ldots
\end{align*}
\]
Q3: [Component based design] [15 points]

Using only one 3-bit binary counter and components as listed below, design a counter that generates the following sequence repeatedly:

(Note: The 3-bit counter starts counting from 0 to 7.)

25 → 22 → 19 → 16 → 13 → 10 → 7 → 4 → 25 ..... 

a) Shifter  
b) Adder  
c) Subtractor  
d) Comparator

The output has the form of (25 – 3x)
Q4: [Custom Design] [10 points]

Design a circuit which executes the following code:

If \((A+B > 14)\)

Then \(S = C + D + 1\)

Else
\[ S = C + D \]

A, B, C and D are 4-bit binary numbers!

You can use the following components:
- Adder
- Comparator
- Subtractor
- Multiplexer
- Counter

HINT: There is a tricky solution to this problem which uses just adders. If you achieve designing this circuit using only 4-bit adders you can earn an extra credit of 5 points.

The tricky solution:
The obvious solution:

```
4 bit Adder Cin Cout
X0  X1  X2  X3  Y0  Y1  Y2  Y3
a 0 a 1 a 2 a 3 b 0 b 1 b 2 b 3
c 0 c 1 c 2 c 3 d 0 d 1 d 2 d 3
```

Comparator

```
A+B<14
A+B=14
A+B>14
```

```