CSEE W3827
Fundamentals of Computer Systems
Homework Assignment 1

Prof. Stephen A. Edwards
Columbia University
Due Tuesday, June 9th, 2015 at 5:30 PM

Print this out and turn it in. Enter answers on the computer or manually on the printout.

This homework requires you to use Logisim, which you can download from http://www.cburch.com/logisim/

Name: SOLUTION

Uni:
1. (5 pts.) What are the values, in decimal, of the following bytes if they are interpreted as 8-bit numbers in

<table>
<thead>
<tr>
<th></th>
<th>Binary</th>
<th>One’s Complement</th>
<th>Two’s Complement</th>
</tr>
</thead>
<tbody>
<tr>
<td>00010011 10011010</td>
<td>19</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>154</td>
<td>-101</td>
<td>-102</td>
</tr>
</tbody>
</table>
2. (5 pts.) Complete the truth table for the following Boolean functions:

\[ a = X \overline{Y} + \overline{X} Y Z + \overline{X} \overline{Z} \]
\[ b = (X + \overline{Y})(X + Z)(\overline{X} + Z) \]

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
3. (20 pts.) Consider the function $F$ whose truth table is shown below.

(a) Write the function $F$ in sum-of-minterms form. Two are given.

$$
!WXY + !WXY + !WXY + !WX + WX +WX + !WXY +
$$

(b) Fill in this Karnaugh map for $F$

(c) Use your Karnaugh map to write a minimal sum-of-products representation for $F$

$$WX!Y + !WXZ + !W!XY$$
In Logisim,

(d) Implement the circuit corresponding to your minimal sum-of-products representation. Verify your circuit using Logisim’s Combinational Analysis feature (Project→Analyze Circuit).

Print your solution and attach it.

(e) Use your Karnaugh map to write a minimal product-of-sums representation for F.

\[ (!W+!Y)(X+Y)(W+!X+Z) \]

(f) Implement the circuit corresponding to your minimal product-of-sums. Again, verify your circuit.

Print your solution and attach it.
Circuit for 3(d)
4. (20 pts.) Create a circuit for a 4-to-10 decoder using AND gates and inverters only. Arrange and name the inputs and outputs as shown below. Treat $W$ as the most significant bit. Only one of the outputs should ever be true.

\[
\begin{align*}
W & \rightarrow A0 \\
X & \rightarrow A1 \\
Y & \quad \quad \vdots \\
Z & \rightarrow A9
\end{align*}
\]

Implement your circuit in Logisim, verify it, and print and attach it.
5. (15 pts.) In Logisim, implement \( F = X \bar{Y} \bar{Z} + YZ + \bar{X}Y \) using just constants and and

(a) a 3-to-8 decoder (under “Plexers→Decoder.” Set “include enable” to “No” and note the input wires are a bundle at the bottom) and an OR gate;

(b) an 8 input mux; and

(c) a 4 input mux whose select inputs are \( X \) and \( Y \), and an inverter.

Implement each of these circuits in Logisim, verify them, and print and attach them.

\[
\begin{align*}
X & \rightarrow \\
Y & \rightarrow \\
Z & \rightarrow \\
\rightarrow F
\end{align*}
\]
Circuit for 5(a)
Circuit for 5(c)
6. (15 pts.) Implement an eight-input mux using two-input muxes only (constants are OK).

Arrange your inputs and outputs as shown below.

\[
\begin{align*}
A_0 & \rightarrow \\
A_1 & \rightarrow \\
\vdots & \\
A_7 & \rightarrow \quad \rightarrow F \\
X & \rightarrow \\
Y & \rightarrow \\
Z & \rightarrow 
\end{align*}
\]

Here, \(A_0\) through \(A_7\) are the eight inputs, and \(X\), \(Y\), and \(Z\) are the three selects. \(X\) is the most significant bit, selecting between, e.g., \(A_0\) and \(A_4\).

Implement your circuit in Logisim, verify it, and print and attach it.
7. (20 pts.) Implement the combinational portion of a three-bit binary counter with an enable input. Give it four inputs, X, Y, Z, and E, and three outputs A, B, and C.

When E is 0, A, B, and C should be X, Y, and Z respectively.
When E is 1, A, B, and C should be X, Y, and Z plus one, with A and X the MSBs.

Your counter should wrap around, i.e., 7 + 1 = 0.

Implement your circuit in Logisim, verify it, and print and attach it.