Review Problem 38

- Create a **truth table** for a circuit which tells if a 3-bit number is evenly divisible by 3 (num/3 leaves no remainder). Have a separate output for the unsigned, 2’s comp, and sign-magnitude versions.

<table>
<thead>
<tr>
<th>B2 B1 B0</th>
<th>Uns</th>
<th>S-M</th>
<th>2’s</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>+0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0 0 1</td>
<td>+1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0 1 0</td>
<td>+2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0 1 1</td>
<td>+3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1 0 0</td>
<td>40</td>
<td>-01</td>
<td>-40</td>
</tr>
<tr>
<td>1 0 1</td>
<td>50</td>
<td>-10</td>
<td>-31</td>
</tr>
<tr>
<td>1 1 0</td>
<td>61</td>
<td>-20</td>
<td>-20</td>
</tr>
<tr>
<td>1 1 1</td>
<td>70</td>
<td>-31</td>
<td>-10</td>
</tr>
</tbody>
</table>
Converting Decimal to Two’s Complement

- Convert absolute value to unsigned binary, then fixed width, then negate if necessary

- Convert $(-9)_{10}$ to 6-bit Two’s Complement

  \[ -(9) = -(1001) = -1011001 = 110110 + 1 = 110111 \]

- Convert $(9)_{10}$ to 6-bit Two’s Complement

  \[ +(9) = +1001 = +001001 = 001001 \]
Converting Two's Complement to Decimal

- If Positive, convert as normal;
  If Negative, negate then convert.

- Convert \((11010)_2\) to Decimal

  \[
  \begin{align*}
  &= -(-11010) \\
  &= -(00101) \\
  &= -(00110) = -(6) = -6_{10}
  \end{align*}
  
- Convert \((01101)_2\) to Decimal

  \[
  +13_{10}
  \]
Sign Extension

- To convert from N-bit to M-bit Two’s Complement (N<M), simply duplicate sign bit:

- Convert \((0010)_2\) to 8-bit Two’s Complement

\[
00000010
\]

-5

- Convert \((1011)_2\) to 8-bit Two’s Complement

\[
11111011
\]

Check that it is -5

\[
= -(-11111011)
= -(100000001)
= -(5)
= -5_{10}
\]
Solving Complex Problems

- Many problems too complex to build as one system
  - Replace with communicating sub-circuits

- Design process:
  - Understand the problem
  - Break problem into subsystems, identifying connections
  - Design individual subsystems.
Complex Problem Example

- Design a digital clock, which can
  - Display the seconds, minutes and hours
  - Have three inputs
    - Increment hour
    - Increment minute
    - Reset seconds
Complex Problem Example (cont.)

![Diagram showing complex problem example with labeled components: '0-59', '0-5', '0-9', 'inc_in', 'inc_out', and 'Reset'.]
Complex Problem Example (cont.)
Complex Problem Example (cont.)