Problem 1 Part A (10 points) Convert the following notations:

<table>
<thead>
<tr>
<th>Binary representation</th>
<th>Hexadecimal representation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1010100010101</td>
<td>AC95</td>
</tr>
<tr>
<td>Signed Binary two's complement represented</td>
<td>Decimal</td>
</tr>
<tr>
<td>11100.01</td>
<td>-3.75</td>
</tr>
<tr>
<td>Decimal</td>
<td>Octal</td>
</tr>
<tr>
<td>673</td>
<td>1241</td>
</tr>
<tr>
<td>Decimal Notation</td>
<td>Binary notation</td>
</tr>
<tr>
<td>7.75</td>
<td>111.11</td>
</tr>
</tbody>
</table>

\[
111.00.01 \rightarrow 0001.101 + 1 \rightarrow 0000.11 \rightarrow 3 + \frac{1}{2} + \frac{1}{2^2} = 3.75
\]
Problem 1 Part B (15 points) For the 24 (and 20) bit representations below, determine the most negative value, most positive value, and step size (difference between sequential values). All answers must be expressed in decimal notation. Fractions (for example 3/16ths) may be used. All signed representations are two's complement signed numbers.

<table>
<thead>
<tr>
<th>representation</th>
<th>most negative value</th>
<th>most positive value</th>
<th>step size</th>
</tr>
</thead>
<tbody>
<tr>
<td>unsigned integer</td>
<td>0</td>
<td>16M - 1</td>
<td>1</td>
</tr>
<tr>
<td>(24 bits). (0 bits)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Signed integer</td>
<td>-8M</td>
<td>8M - 1</td>
<td>1</td>
</tr>
<tr>
<td>(24 bits). (0 bits)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>unsigned fixed-point</td>
<td>0</td>
<td>128K - 1/128</td>
<td>1/128</td>
</tr>
<tr>
<td>(17 bits). (7 bits)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>signed fixed-point</td>
<td>-16K</td>
<td>16K - 1/32</td>
<td>1/32</td>
</tr>
<tr>
<td>(15 bits). (5 bits)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**UN SIGNED:**

\[
\frac{2^{24} - 1}{2^8} = 2^{24} - 1 = 2^{16}.2^8 - 1 = 16M - 1
\]

**SIGNED:**

\[
-2^{16} \cdot \frac{2^{17} - 1}{2^7} = -2^{16} \cdot \frac{2^{17} - 1}{2^7} = -2^{10}.2^8 + 2^8 + 2^2 - 1
\]

**UNSIGNED FIXED POINT**

\[
\frac{2^{17} - 1}{2^7} = 2^{17} - 1/2^7 = 2^{10}.2^7 - 1/2^7 = 128K - 1/128
\]

**SIGNED**

\[
-2^{14} + 2^{14} = -2^{10}.2^4 + 2^4 + 2^2 - 1/4
\]

\[
= -16K + 16K - 1/32
\]
Problem 2 Part A Arithmetic (5 points) For each problem below, compute the operations using the rules of arithmetic, and indicate whether an overflow occurs assuming all numbers are expressed using a six bit **unsigned** representation

\[
\begin{array}{c}
110110 \\
+100110 \\
\hline
011001 \\
+010110 \\
\hline
111111
\end{array}
\]

result: 111111

unsigned error? Y N

Problem 2 Part B Arithmetic (10 points) For each problem below, compute the operations using the rules of arithmetic, and indicate whether an overflow occurs assuming all numbers are expressed using a six bit **signed** two's complement representations.

\[
\begin{array}{c}
011111 \\
+000100 \\
\hline
111111
\end{array}
\quad
\begin{array}{c}
111111 \\
+000010 \\
\hline
111011
\end{array}
\quad
\begin{array}{c}
011000 \\
+001111 \\
\hline
111011
\end{array}
\quad
\begin{array}{c}
001010 \\
+010101 \\
\hline
111010
\end{array}
\]

result: 111011

signed error? Y N N N
3) Given the circuit below:

a) (5 points) The designer wants to implement an exclusive-or function where out = Y \( \text{exclusive-or} \) X. Fill in the values for F_3 F_2 F_1 F_0 to implement the desired exclusive-or function.

b) (5 points) The designer wants to implement the NAND function where out = Y \( \text{NAND} \) X. Fill in the values for F_3 F_2 F_1 F_0 to implement the desired NAND function.

C) (5 points) The designer wants to implement the NOR function where out = Y \( \text{NOR} \) X. Fill in the values for F_3 F_2 F_1 F_0 to implement the desired NOR function.

<table>
<thead>
<tr>
<th>function desired</th>
<th>F_3</th>
<th>F_2</th>
<th>F_1</th>
<th>F_0</th>
</tr>
</thead>
<tbody>
<tr>
<td>a) exclusive-or</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>b) NAND</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>c) NOR</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
4) Counters (15 points) Connect the needed toggle cells below to build a multiple digit counter (like in an alarm clock type application) that counts in the following strange 2 digit sequence. Include any circuitry needed to allow this to work. The toggle cells in the right column are for the least significant digit (right value), the toggle cells in the left column are for the most significant digit (left value). Include an active high count enable (CE) and an active high reset (RESET) inputs.

desired count sequence: 00, 01, 02, 03, 04, 10, 11, 12, 13, 14, 20, 21, 22, 23, 24, 30, 31, 32, 33, 34, 40, 41, 42, 43, 44, 00, 01, 02, 03, 04, 10, ........ THIS IS A DIVIDE BY 5 FOR BOTH DIGITS SO MAX COUNT AT VALUE OF 4

NOTE: ALL CLK 1'S CONNECTED
ALL CLK 2'S CONNECTED
5) Priority Encoders (15 points) Given the truth table for the following priority encoder:

<table>
<thead>
<tr>
<th>IN3</th>
<th>IN2</th>
<th>IN1</th>
<th>IN0</th>
<th>OUT1</th>
<th>OUT0</th>
<th>VALID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>X</td>
<td>X</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>X</td>
<td>1</td>
<td>0</td>
<td>X</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>1</td>
<td>X</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>X</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

a) List the priority order of the inputs IN3, IN2, IN1, IN0:

\[
\text{highest} \quad \frac{\text{IN1}}{} > \frac{\text{IN2}}{} > \frac{\text{IN0}}{} > \frac{\text{IN3}}{} \quad \text{lowest}
\]

b) Using basic gates (AND, OR, NAND, NOR, NOT) show the gate level implementation for this priority encoder.

![Gate level implementation diagrams]
6) Registers (15 points) Consider the register implementation below.

Assume the following signals are applied to your register. Draw the signal at point A (output of the first latch), the signal at point OUT (output of second latch). Assume A and OUT start at unknown values.
K MAPS:

Diagram of K maps with variables A, B, C, and D.