1. Give an efficient mixed-logic synthesis of

\[ Y = (A \cdot B + C) \cdot (D \oplus E + \overline{B}) \]

using two-input nor gates (LS02), two-input exclusive-or (⊕) gates (LS86) and inverters (LS04), and signals \( A \cdot l, B \cdot h, C \cdot l, D \cdot h, E \cdot l \) and \( Y \cdot l \). Try to minimize the number of chips.
2. Consider the logic schematic

(a) Express this logic as a system of boolean equations.

(b) Implement this specification with LS00, LS02 and LS04 devices. The I/O constraints are $A.H$, $B.H$, $E.I$ and $W.L$. $C$ and $D$ may be either polarity.
3. (a) Using the K-map below, derive a reduced (near-minimal) sum-of-products expression for the function whose truth table is shown to the right. Unspecified or conflicting cases, if any, may be regarded as don’t-cares.

(b) Using boolean algebra, further optimize to an equivalent multi-level term with fewer operations than the SoP form.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>X</td>
<td>0</td>
<td>X</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>X</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>X</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>X</td>
<td>X</td>
<td>e</td>
<td>π</td>
</tr>
<tr>
<td>1</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>1</td>
<td>0</td>
<td>X</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>X</td>
<td>X</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>X</td>
<td>1</td>
<td>X</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>X</td>
<td>0</td>
</tr>
</tbody>
</table>

\[
\begin{array}{cc|cc|c}
A & B & C & D & E & Z \\
\hline
00 & 01 & 11 & 10 &
\end{array}
\]
4.

(a) Give a mixed-logic synthesis of the logic expression \( A (B \overline{C} + \overline{B} D) + B \overline{C} \overline{D} \) using only 74LS06 open-collector inverters and pull-up resistors (which you may assume are appropriately sized). Assume all inputs and outputs must have positive \((T = H)\) polarity.

(b) A SoP form of the original expression, \( A B C + A \overline{B} D + B \overline{C} \overline{D} \). Can also be implemented with '06s, using a more regular logic array layout structure (possibly at the cost of more gates). Draw such an implementation.
5. Design a one-bit *full adder-subtractor* taking operand inputs \( A \) and \( B \); carry/borrow input \( C/B_{in} \); and add/subtract mode input \( M \); and generating outputs \( S \) for the sum/difference and \( C/B_{out} \) for the propagated carry/borrow bit.

When \( M = 1 \) the device should behave as a full adder as described in class; and when \( M = 0 \) it should behave as a full subtractor. The functions should compose in a *ripple-carry* way to get \( n \)-bit adder/subtractor combinations.
6. In certain technologies (to be described later in this course) the only gate provided is a $2^n$-input multiplexer. Suppose you have only 2-input multiplexers for implementing the expression

$$C \cdot D + A \cdot C + B \cdot \overline{C} + A \cdot C \cdot \overline{D}$$

That is, you must convert this expression to one composed of nested combinations of:

$$\text{mux}(s, x, y) \overset{\text{def}}{=} (s \cdot x) + (\overline{s} \cdot y)$$

(a) Design a mux-based implementation of $E$. If you employed a specific technique (e.g. boolean algebra, etc.) to accomplish this task, explain it for extra credit. For even more extra credit, implement your strategy in software.

(b) Synthesize a solution using only 74LS125 tri-state buffers and 74LS04 inverters.