1. (15 points) Write a function which rotates an 8-bit character left 1-bit. For example, roll(0x81) returns 0x00; roll(0xa0) returns 0xa1.

```c
char roll(char a) {
    int shiftout;
    shiftout = (a & 0x80) << 1;  // 0
    a = ((a << 1) & 0xff) | shiftout;  /* also can use 0xff* /
    return shiftout;
}
```

**Best solution**

```c
char roll(char a) {
    return ((a << 1) | ((a & 0x80) << 1)) & 0xff;
}
```

2. (15 points) Draw the Dataflow diagram for the following code. The char size is 8-bits.

```c
char a, b, c, d;
if (a < 0) { c = c | 0x80; }
else { c = c & 0x7f; }
```

![Dataflow Diagram](image-url)
3. (10 points) Draw the state transition diagram for a 1-bit input for the regular expression "0+1+0+." Use the following state symbols: S for start, F for final, E for Error and W, X, Y, Z, for all others.

4. (15 points) Given the state transition diagram for a 1-bit input (a) draw the error state, E, for the missing inputs and (b) give the transition table.

5. (15 points) Given the transition table, give the state encodings using S=11, W=10, E=00 and F=01.
6. (15 points) (a) Give the three excitation optimal k-map of the transition encoded table and clearly show circles. Treat each k-map independently (i.e., do not do multi-output k-map optimisation). (b) Give the minimal SOP expression for each k-map. Let \( q_1 = a, q_2 = b \) and \( i = c \).

<table>
<thead>
<tr>
<th>( t )</th>
<th>( a )</th>
<th>( b )</th>
<th>( c )</th>
<th>( d )</th>
</tr>
</thead>
<tbody>
<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>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

MSOP of \( q_1 \) is \( a \) \( \lor \) \( b \)

MSOP of \( q_2 \) is \( a \) \( \lor \) \( \overline{b} \)

7. (15 points) Given the following gate logic diagram draw the timing diagram including the triggering lines. Each logic gate has 1 nanosecond delay.
x1. (extra credit, 10 points) Write a multi-precision function which rotates left by 1 bit and returns the rotated out bit. char r Code (char *r, char *a, int a, int a) {

    int i, shiftout, shiftin;
    shiftin = (a & 0xf0) >> 5;
    for (i = an-1; i >= 0; i--) {
        shiftout = (a[i] & 0x80) >> 5;
        f[i] = ((a[i] & 0xf0) | shiftin) << (i + b);
        shiftin = shiftout;
    }
    return shiftout;
}

x2. (extra credit, 10 points) Draw the Dataflow diagram of problem 2 using only wires and no logic gates, arithmetic logic, or transistors.

\[ \begin{align*}
    a & \rightarrow S \\
    c & \rightarrow d \\
    a & \rightarrow d
\end{align*} \]

x3. (extra credit, 10 points) Draw the state transition diagram for a 1-bit input for the regular expression "00|11|00|". Use the following state symbols: S for start, F for final, E for Error and W, X, Y, Z, for all others.

\[ \begin{align*}
    & \text{S} \quad \text{E} \quad \text{F} \\
    0 & \quad \text{E} \quad \text{F} \\
    1 & \quad \text{E} \quad \text{F} \\
    0 & \quad \text{E} \quad \text{F} \\
    1 & \quad \text{E} \quad \text{F} \\
    0 & \quad \text{E} \quad \text{F} \\
    1 & \quad \text{E} \quad \text{F} \\
    0 & \quad \text{E} \quad \text{F} \\
    1 & \quad \text{E} \quad \text{F} \\
\end{align*} \]

x4. (extra credit, 10 points) Given the three input bit string of problem 4. (a) Show the truth table if the final state, F, is true and the error state, E, is false. (b) What is the minimal boolean expression for this function?

\[ \begin{array}{ccc|c}
    a & b & c & \text{Output} \\
    0 & 0 & 0 & 0 \\
    0 & 0 & 1 & 1 \\
    0 & 1 & 0 & 1 \\
    0 & 1 & 1 & 1 \\
    1 & 0 & 0 & 1 \\
    1 & 0 & 1 & 0 \\
    1 & 1 & 0 & 0 \\
    1 & 1 & 1 & 1 \\
\end{array} \]

Output expression:

\[ \begin{align*}
    (a \oplus b) \oplus c
\end{align*} \]