1. Draw a state diagram for this state machine:

![State Diagram](image)

2. Create a Moore state machine (i.e. one whose outputs depend on the current state and not directly on the inputs) to recognize the input sequence 010. Your state machine should have a single bit of input $X$ and a single bit of output $Y$, which is 1 if the last three inputs were 010, and zero otherwise. Note that the input sequence 01010 should cause the output to be 1 twice: once for the second 0 in the input and once for the third 0 in the input.

   a. Draw a state diagram for this state machine. Your implementation should have two bits of state for a total of four possible states.

   b. Derive the state transition function and the output function from your state diagram.

   c. Draw a logic diagram implementing this circuit using two $D$ flip-flops and any additional gates necessary (you may also use MUXes if you like).
3. Create a Verilog implementation and test bench for problem 1 by instantiating two 
$D$ flip-flops along with two full adders (both contained in the starter code distribution) 
inside your own module

\[
\text{module hw7}_1(\text{clk, rst, X, S});
\]

You may wish to look at the \texttt{jk\_flipflop} example to see how to use the $D$ flip-flop 
module. For the test bench, keep a regular clock with a 20-tick period, and the first 
positive edge at $t = 10$. The timing for the $X$ input should be as follows:

<table>
<thead>
<tr>
<th>$t$</th>
<th>0</th>
<th>20</th>
<th>40</th>
<th>60</th>
<th>80</th>
<th>100</th>
<th>120</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X$</td>
<td>01</td>
<td>11</td>
<td>10</td>
<td>10</td>
<td>00</td>
<td>11</td>
<td></td>
</tr>
</tbody>
</table>

Your Verilog simulation should terminate at $t = 140$.

4. Create a Verilog implementation and test bench for problem 2 using behavioral Verilog 
(i.e. non-blocking assignments inside of an \texttt{always} block, similar to the example 
shown in class) to create a module

\[
\text{module hw7}_2(\text{clk, rst, X, Y});
\]

For the test bench, keep a regular clock with a 20-tick period, and the first positive edge 
at $t = 10$. The timing for the $X$ input should be as follows:

<table>
<thead>
<tr>
<th>$t$</th>
<th>0</th>
<th>20</th>
<th>40</th>
<th>60</th>
<th>80</th>
<th>100</th>
<th>120</th>
<th>140</th>
<th>160</th>
<th>180</th>
<th>200</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Your Verilog simulation should terminate at $t = 220$. 