1. Each of the following state machines can be implemented with a single $D$ flip-flop and some additional gates. Determine the state transition function for these state machines, and draw a logic diagram. Implement both of these as Moore state machines, and be sure to include clock and reset lines in your diagram.
   
a. There is a single input $X$. When $X = 0$, the next state $N$ is equal to the current state $S$; when $X = 1$, the next state is equal to the complement of the current state.
   
b. There are two inputs $A$ and $B$. When $A = 0$ and $B = 0$, the next state is equal to the complement of the current state; when $A = 0$ and $B = 1$, the next state is 1; when $A = 1$ and $B = 0$, the next state is 0; and when $A = 1$ and $B = 1$, the next state is equal to the current state.

2. Consider the following state machine:

   a. Is this a Mealy state machine or a Moore state machine? Why?
   
b. Draw a state transition diagram for this state machine.
3. Create a state machine to recognize the sequence 1001. Your state machine should have a single bit of input $X$ and a single bit of output $Y$, which is 1 for the input sequence 1001, and 0 otherwise. Note that the sequence 1001001 should cause the output to be 1 twice: once for the second 1 in the input and once for the third 1 in the input.

   a. Draw a state diagram for a Moore state machine recognizing the sequence 1001.
   b. Draw a state diagram for a Mealy state machine recognizing the sequence 1001.
   c. How many bits of state are required for the Moore state machine? How many for the Mealy? Which design would use fewer flip-flops?

4. Create a Verilog implementation and test bench for problem 1b by instantiating a single $D$ flip-flop inside your own module

   module hw7_1b(clk, rst, A, B, S);

   You can use the starter code on the website to help you get started – it contains an example for problem 1a, along with an implementation of a $D$ flip-flop. 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 $A$ and $B$ inputs should be as follows:

   \[
   \begin{array}{c|cccccccc}
   t & 0 & 20 & 40 & 60 & 80 & 100 & 120 \\
   AB & 00 & 01 & 10 & 11 & 00 & 11 & 10 \\
   \end{array}
   \]

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

5. Create a Verilog implementation and test bench for problem 2 by instantiating two $D$ flip-flops along with two full adders (also contained in the starter code distribution) inside your own module

   module hw7_2(clk, rst, X, S);

   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:

   \[
   \begin{array}{c|cccccccc}
   t & 0 & 20 & 40 & 60 & 80 & 100 & 120 \\
   X & 01 & 01 & 11 & 10 & 10 & 00 & 11 \\
   \end{array}
   \]

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

2