

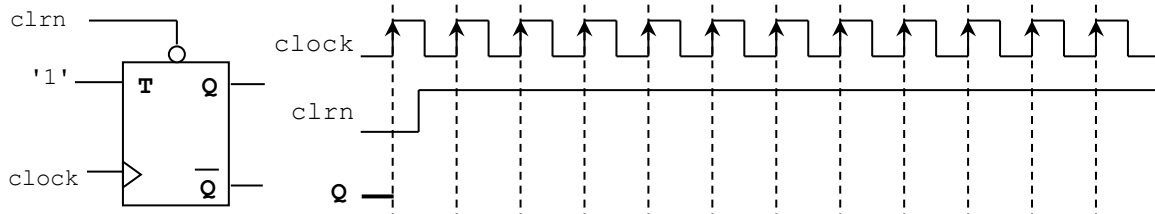
Homework 3

(Due date: October 31st @ 9:30 am)

Presentation and clarity are very important!

PROBLEM 1 (20 PTS)

- Complete the timing diagram of the circuit shown below. If the frequency of the signal clock is 25 MHz, what is the frequency (in MHz) of the signal Q? (5 pts)



- Complete the timing diagram of the circuit whose VHDL description is shown below: (5 pts)

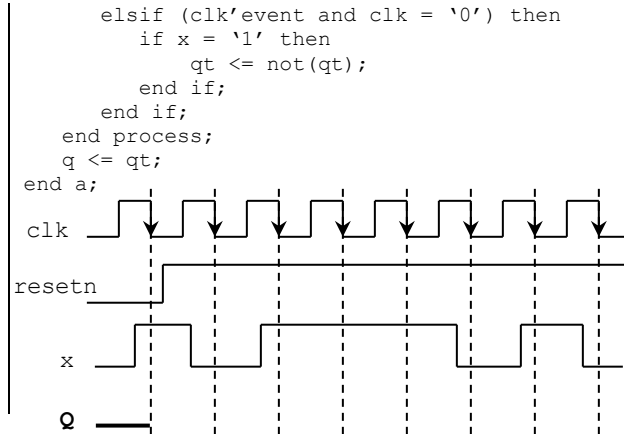
```
library ieee;
use ieee.std_logic_1164.all;

entity circ is
port ( resetn, x, clk: in std_logic;
      q: out std_logic);
end circ;

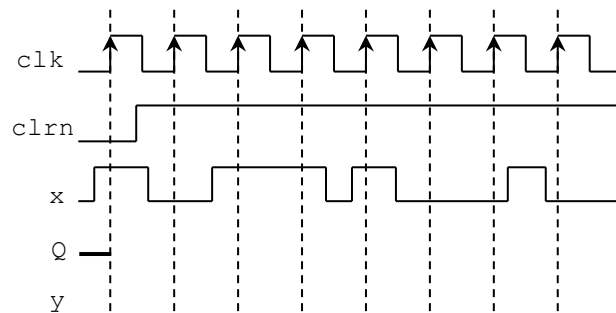
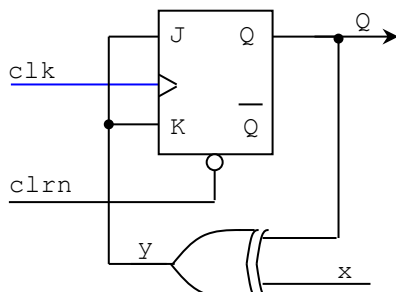
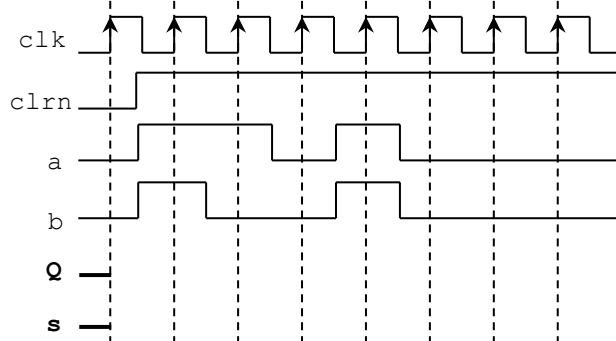
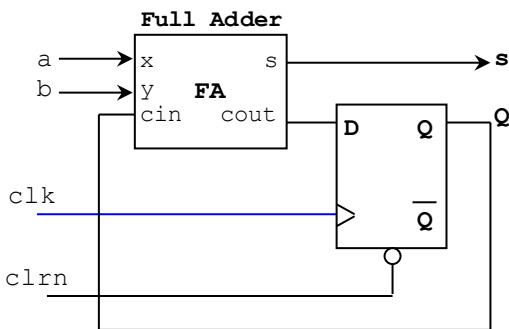
architecture a of circ is
signal qt: std_logic;

begin
process (resetn, clk, x)
begin
if resetn = '0' then
qt <= '0';

```



- Complete the timing diagram of the circuits shown below: (10 pts)



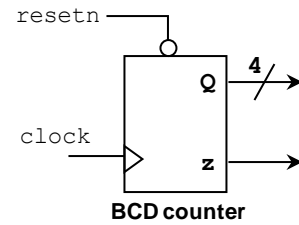
PROBLEM 2 (15 PTS)

Design a BCD counter via a Finite State Machine (FSM):

BCD counter features:

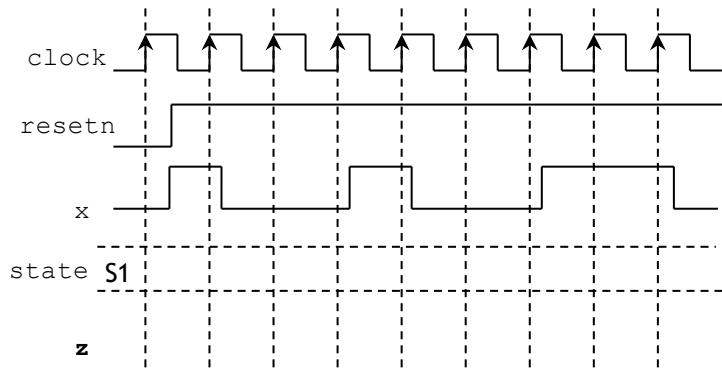
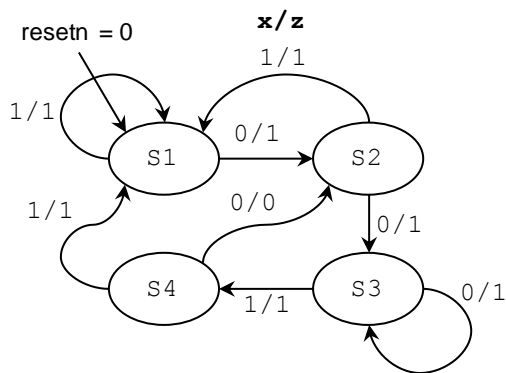
- count: 0000, 0001, 0010, 0011, ..., 1000, 1001, 0000, ...
- resetn: Asynchronous input signal. It initializes the count to "0000"
- output 'z': It becomes '1' when the count is 1001.

- ✓ Provide the State Diagram and the Excitation table. Is this a Moore or Mealy machine?
- ✓ Sketch the circuit (simplify your circuit using K-maps).

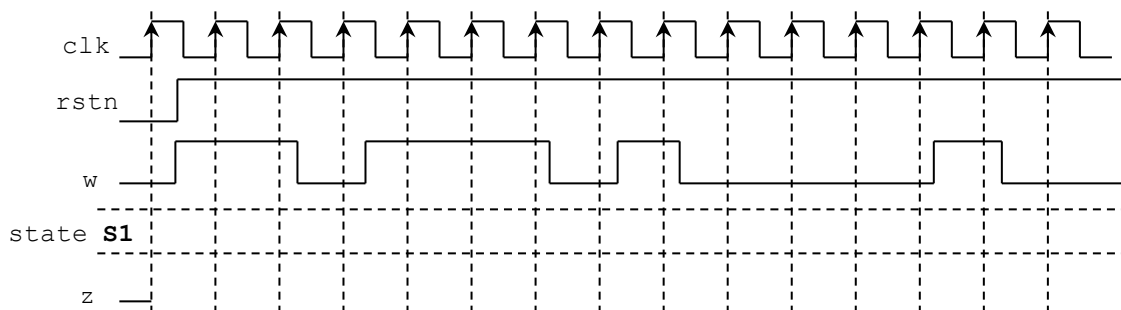
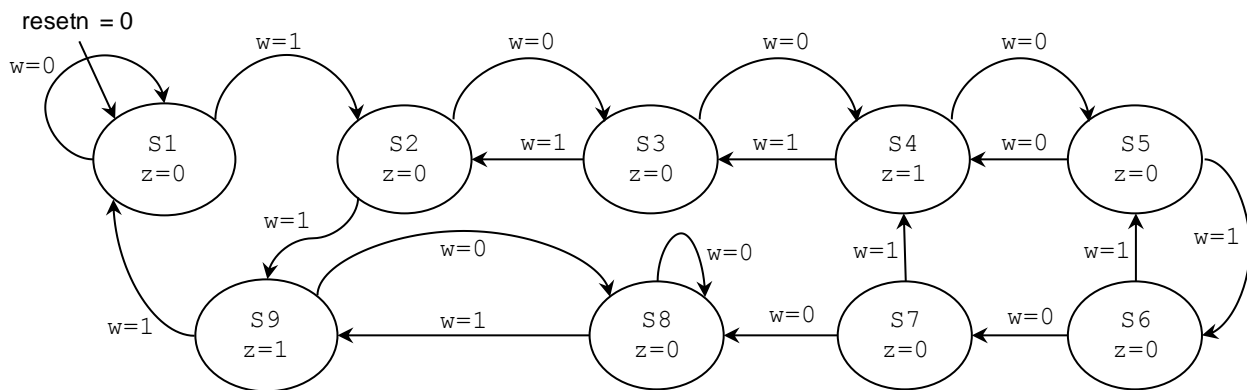


PROBLEM 3 (15 PTS)

- Sequence detector (with overlap): Draw the state diagram of a circuit that detects the following sequence: 1011010. The detector must assert an output 'z=1' when the sequence is detected.
- Complete the timing diagram of the following state machine and **provide the VHDL code**:

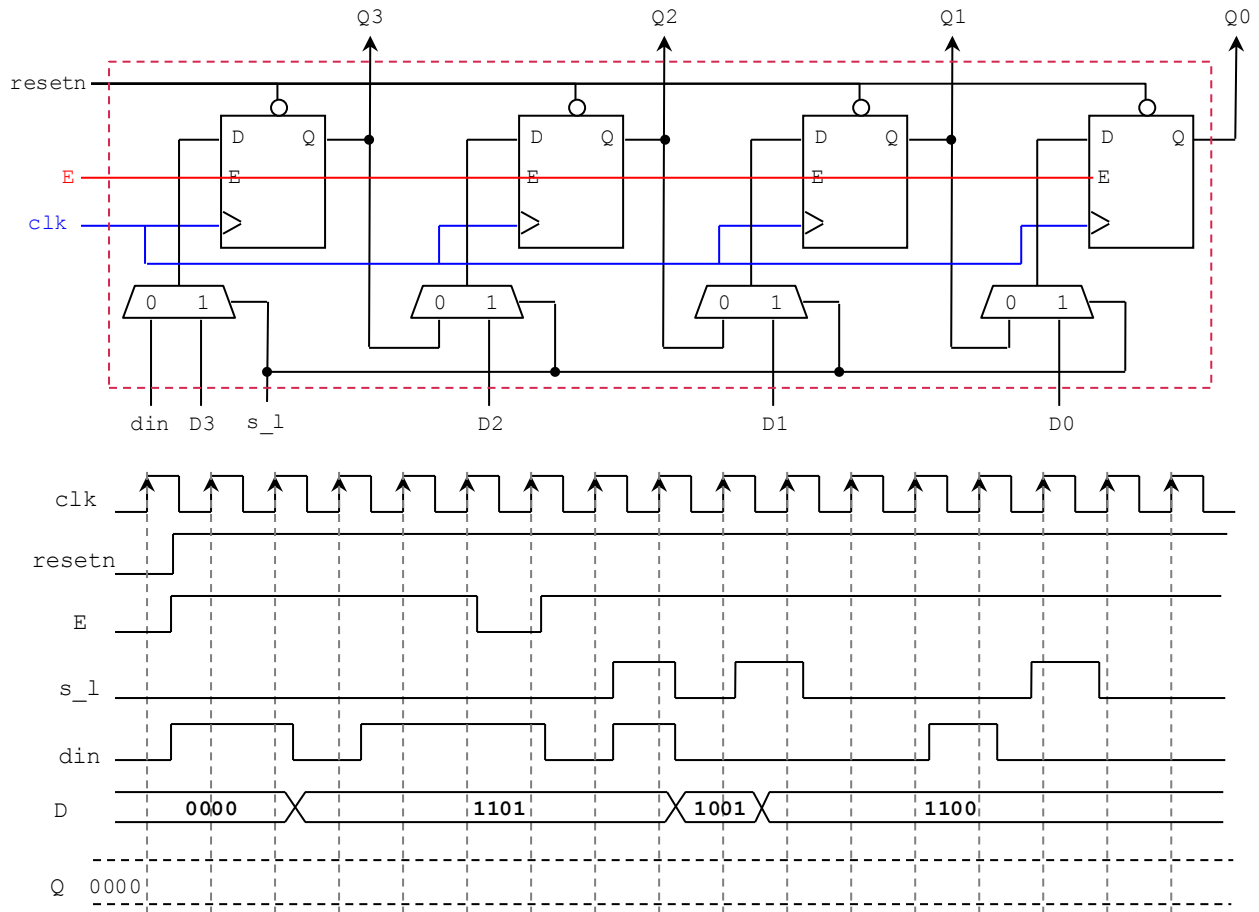


- Complete the timing diagram of the following Moore-type FSM and **provide the VHDL code**:



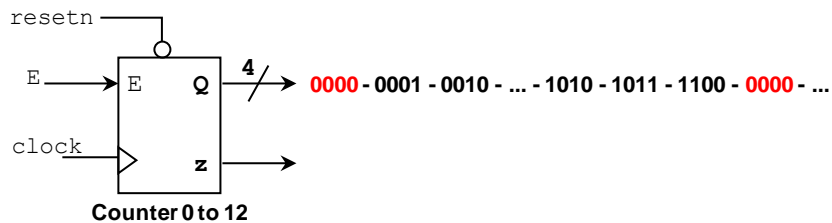
PROBLEM 4 (20 PTS)

- Parallel/serial load shift register with enable input. Shifting operation: $s_1=0$. Parallel load: $s_1=1$.
- ✓ Provide the VHDL code of the circuit shown below.
- ✓ Create a VHDL testbench according to the timing diagram shown below. Complete the timing diagram by simulating your circuit. The clock frequency must be 50 MHz.

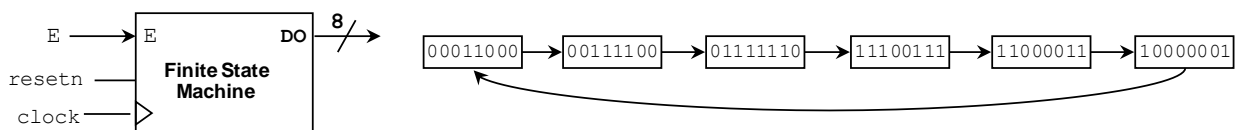


PROBLEM 5 (20 PTS)

- Provide the VHDL code of a counter (from 0 to 12) with enable (E). If $E=0$, the count stops. The output 'z' is asserted ($z=1$) when $Q=1100$. (5 pts)



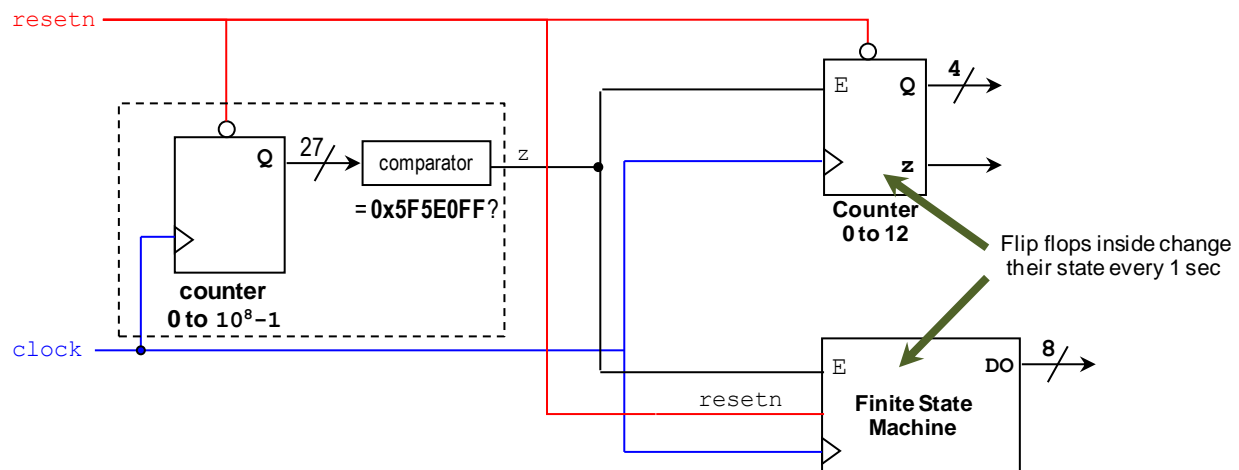
- Provide the state diagram and the VHDL code of a Moore-type Finite State Machine that generates the following sequence: (15 pts)



- **Important:** Include an enable input in your state machine. When $E=1$, state transitions do occur. When $E=0$ there are no state transitions.
- **Tip:** the output bits don't have to be the outputs of the flip flops. You can create this state machine with only 6 states, thereby requiring only 3 flip flops.

PROBLEM 6 (10 PTS)

- We want to connect the output bits of the circuits in Problem 5 to LEDs in the NEXYS3 Board. And we want to see the output transitions before our eyes. In the NEXYS3 Board, the input clock frequency is 100 MHz (Period: 10 ns), making it impossible for our eyes to perceive the transitions.
- If we want the output bits to change every **1 second** (for example) a straightforward solution is to modify the clock frequency to 1 Hz. But this can be a hard problem if a precise input clock is required.
- **Alternative solution:** We create a circuit that generates a one-period (10 ns) pulse every 1 second. This output is then connected to the enable input of every flip flop, counter, and register whose rate of operation we would like to modify. This way, we get the same effect as modifying the clock frequency to 1 Hz. And we get to use the 100 MHz clock for all the flip flops. The figure below depicts this circuit (black dotted box).
 - We need to count 10^8 clock cycles to get to 1 second (100 MHz amounts to a period of 10 ns). When the count becomes $10^8 - 1$, the circuit generates a pulse ($z=1$). The counter requires $\lceil \log_2 10^8 \rceil = 27$ bits, and the comparison has to be with $10^8 - 1 = 0x5F5E0FF$.
 - In VHDL, the 'z' output signal can be generated internally in the counter description, or we can attach an external comparator to the output of the counter.



- ✓ Provide the VHDL code of the circuit that generates a 10 ns pulse every **500 milliseconds** (this circuit allows the circuits of Problem 5 to operate every 500 milliseconds).

EXTRA CREDIT (+15 PTS)

- Implement the circuits of Problem 5 in VHDL, with the output bits changing every 500 milliseconds.
- Demonstrate the circuits working on the NEXYS3 Board.