

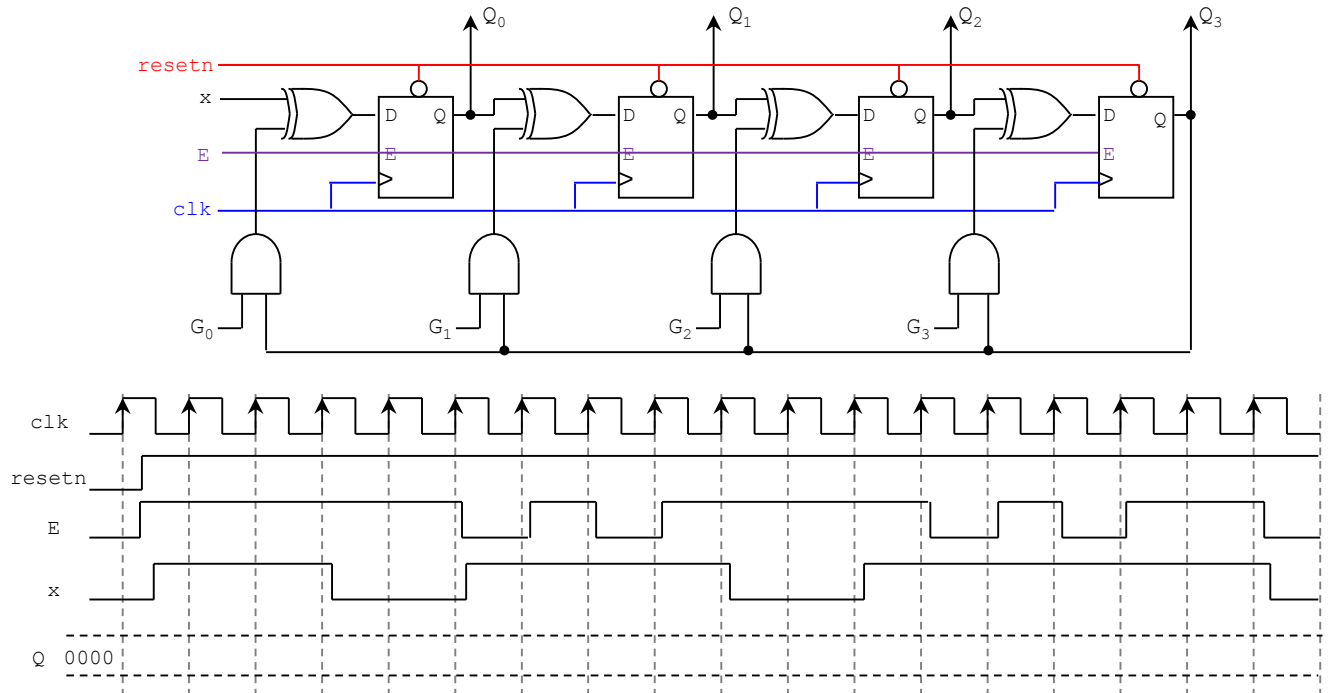
Final Exam

(April 19th @ 7:00 pm)

Presentation and clarity are very important! Show your procedure!

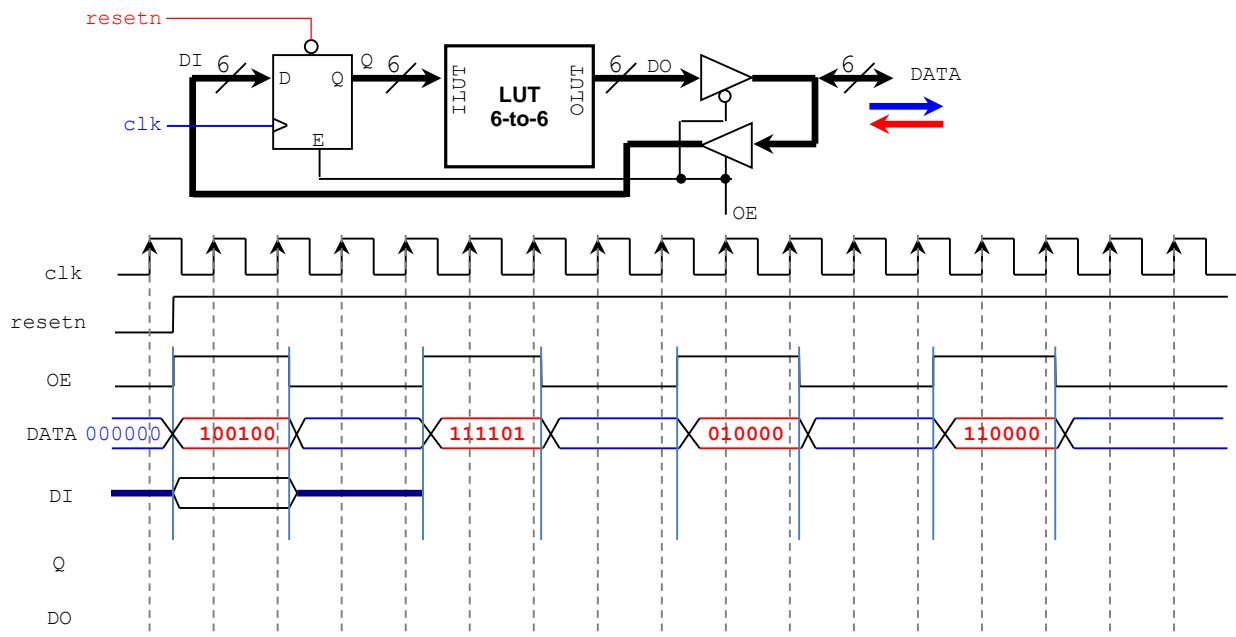
PROBLEM 1 (12 PTS)

- Complete the timing diagram of the following circuit. $G = G_3G_2G_1G_0 = 1001$, $Q = Q_3Q_2Q_1Q_0$



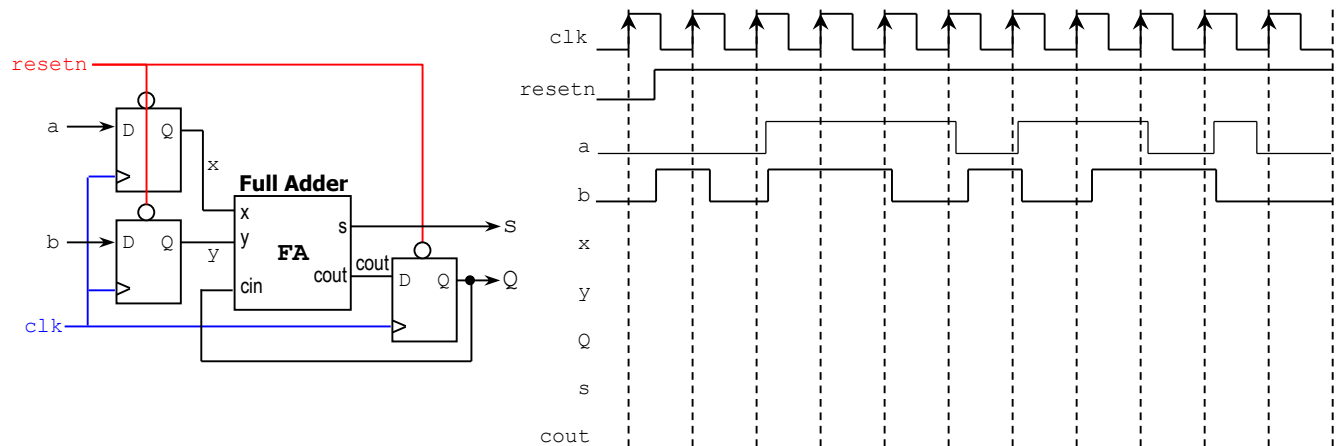
PROBLEM 2 (12 PTS)

- Given the following circuit, complete the timing diagram.
The LUT 6-to-6 implements the following function: $OLUT = \lceil \sqrt{ILUT} \rceil$, where $ILUT$ is a 6-bit unsigned number.
For example $ILUT = 41 (101001_2) \rightarrow OLUT = \lceil \sqrt{41} \rceil = 7 (000111_2)$

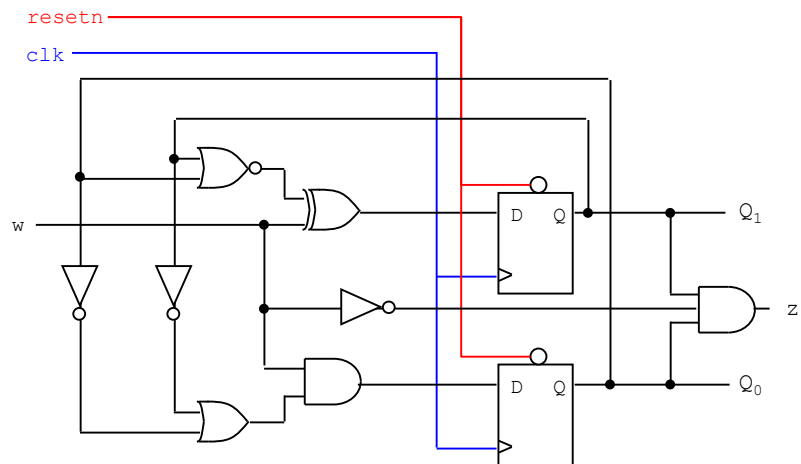


PROBLEM 3 (20 PTS)

- Complete the timing diagram of the circuit shown below: (8 pts)

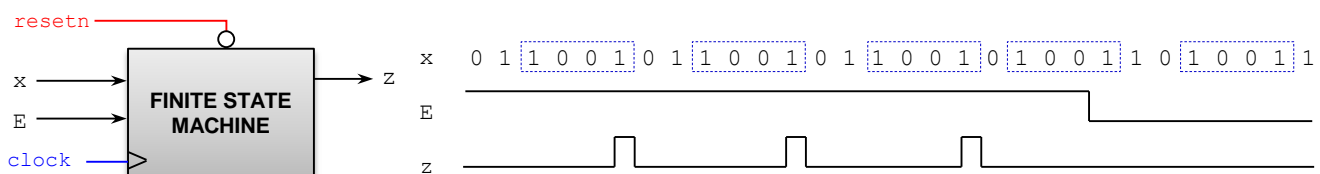


- Provide the State Diagram (any representation), the Excitation Table, and the Excitation equations of the following Finite State Machine: (12 pts)



PROBLEM 4 (22 PTS)

- Sequence detector: The machine has to generate $z = 1$ when it detects the sequence 1001. Once the sequence is detected, the circuit looks for a new sequence.
- The signal E is an input enable: It validates the input x , i.e., if $E = 1$, x is valid, otherwise x is not valid.



- Draw the State Diagram (any representation), State Table, and the Excitation Table of this circuit with inputs E and x and output z . Is this a Mealy or a Moore machine? Why? (15 pts)
- Provide the excitation equations (simplify your circuit using K-maps or the Quine-McCluskey algorithm) (4 pts)
- Sketch the circuit. (3 pts)

PROBLEM 5 (16 PTS)

- Draw the State Diagram (in ASM form) of the FSM whose VHDL description is shown below. Is it a Mealy or a Moore FSM?
- Complete the Timing Diagram.

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

entity myfsm is
    port ( clk, resetn: in std_logic;
          r, p, q: in std_logic;
          x, w, z: out std_logic);
end myfsm;
```

```
architecture behavioral of myfsm is
    type state is (S1, S2, S3);
    signal y: state;
begin
    Transitions: process (resetn, clk, r, p, q)
    begin
        if resetn = '0' then y <= S1;
        elsif (clk'event and clk = '1') then
            case y is
                when S1 =>
                    if r = '1' then
                        y <= S2;
                    else
                        if p = '1' then y <= S3; else y <= S1; end if;
                    end if;

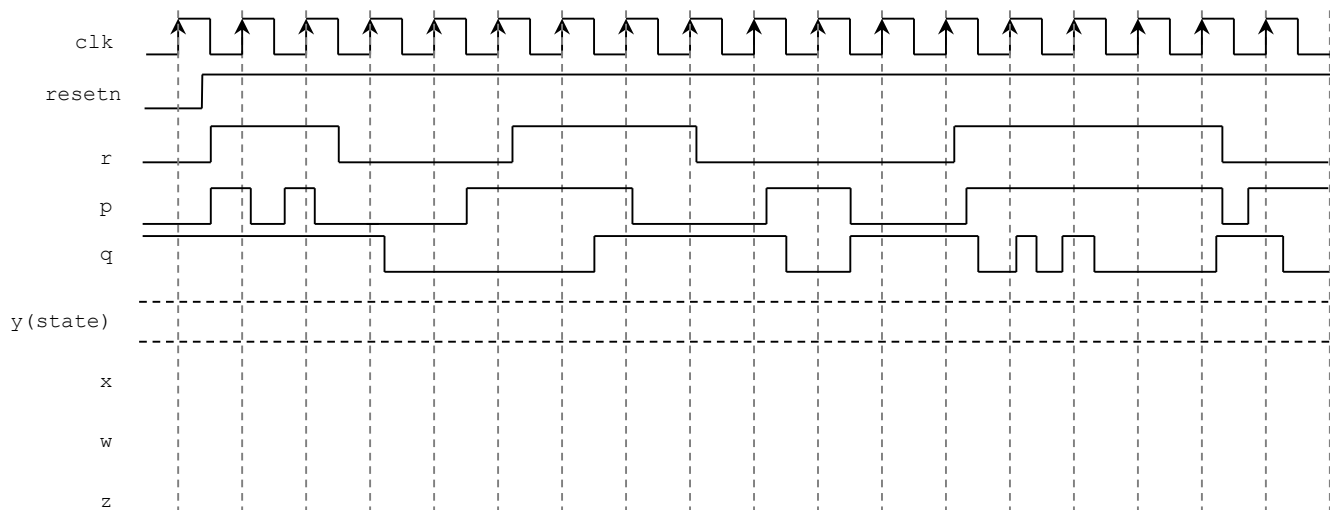
                when S2 =>
                    if p = '1' then y <= S1; else y <= S3; end if;

                when S3 =>
                    if q = '1' then y <= S3; else y <= S2; end if;
            end case;
        end if;
    end process;

    Outputs: process (y, r, p, q)
    begin
        x <= '0'; w <= '0'; z <= '0';
        case y is
            when S1 => if r = '0' then
                            if p = '0' then
                                z <= '1'; x <= '1';
                            end if;
                        end if;

            when S2 => if q = '0' then x <= '1'; end if;
                       if p = '0' then w <= '1'; end if;

            when S3 => if q = '0' then x <= '1'; end if;
        end case;
    end process;
end behavioral;
```



PROBLEM 6 (18 PTS)

- “Counting 1’s” Circuit: It counts the number of bits in register A that has the value of ‘1’. The digital system is depicted below: FSM + Datapath. Example: For $n = 8$: if $A = 00110110$, then $C = 0100$.
 - ✓ m-bit counter: If $E = sclr = 1$, the count is initialized to zero. If $E = 1, sclr = 0$, the count is increased by 1.
 - ✓ Parallel access shift register: If $E = 1: s_l = 1 \rightarrow \text{Load}, s_l = 0 \rightarrow \text{Shift}$.
- Complete the timing diagram (next page) where $n = 8, m = 4$.

```

C ← 0
while A ≠ 0
    if a0 = 1 then
        C ← C + 1
    end if
    right shift A
end while
    
```

