

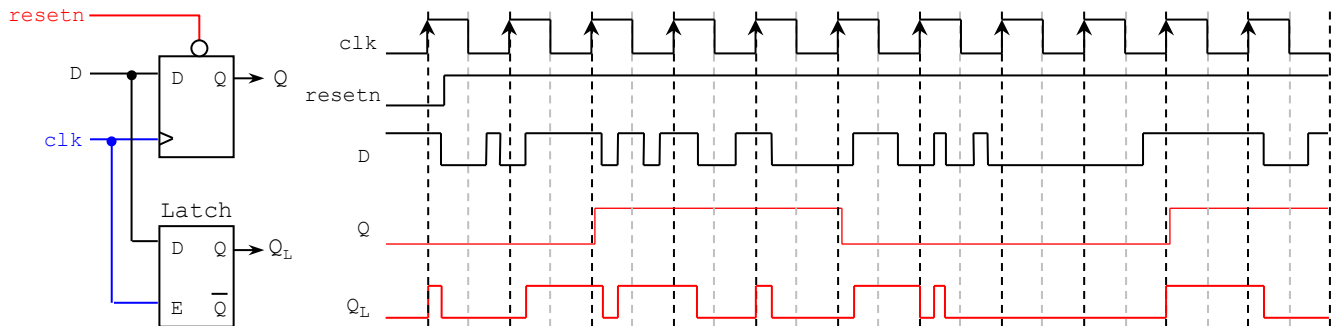
Solutions - Homework 3

(Due date: November 2nd @ 11:59 pm)

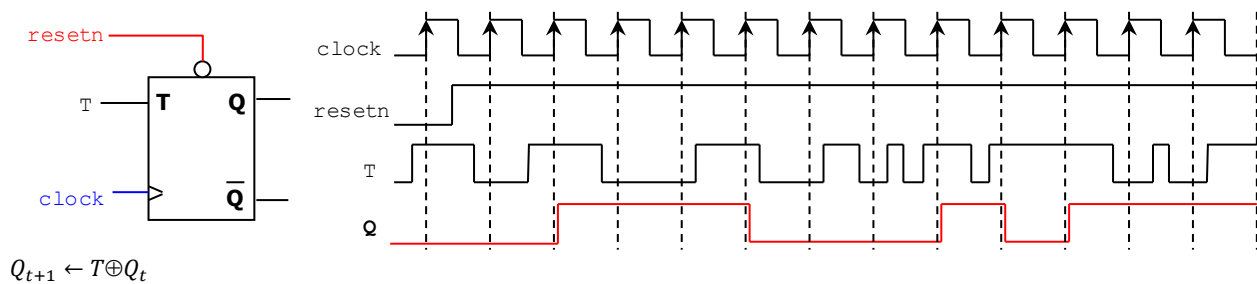
Presentation and clarity are very important! Show your procedure!

PROBLEM 1 (12 PTS)

- Complete the timing diagram of the circuit shown below. (7 pts)



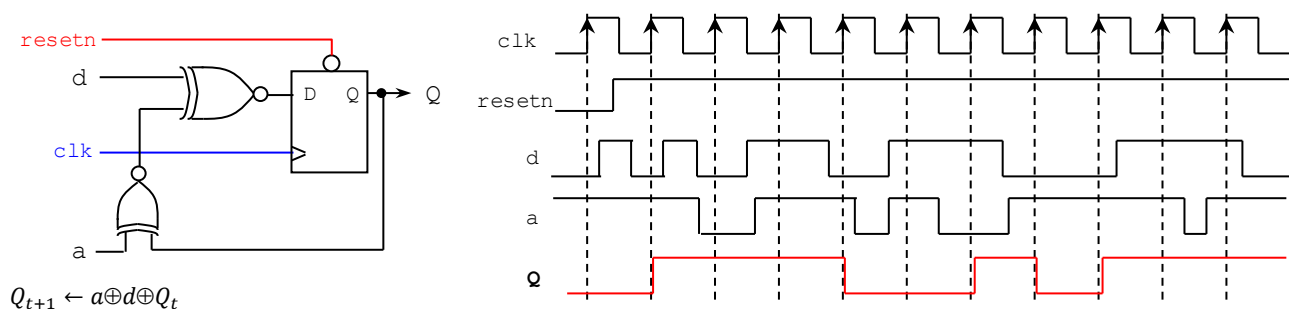
- Complete the timing diagram of the circuit shown below: (5 pts)



$$Q_{t+1} \leftarrow T \oplus Q_t$$

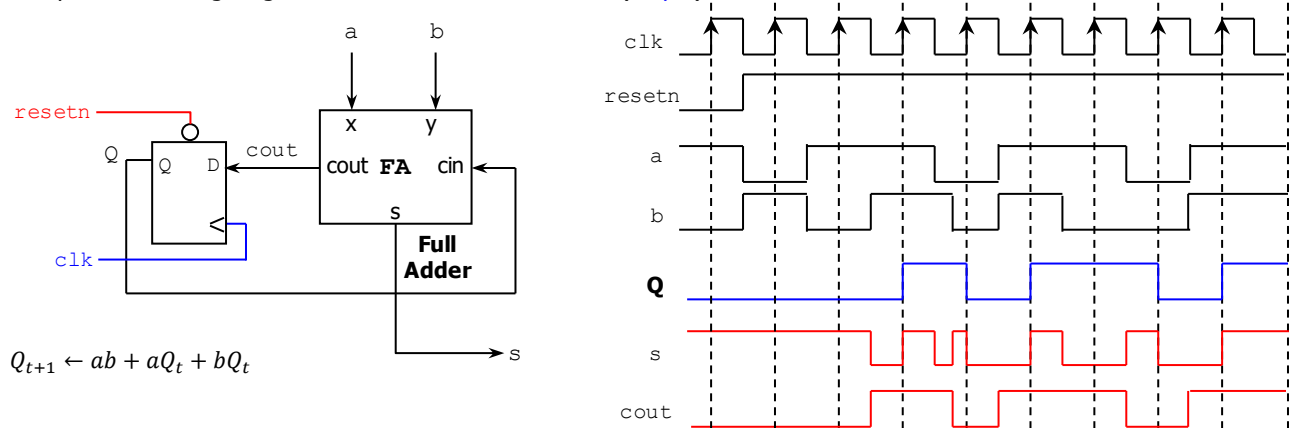
PROBLEM 2 (16 PTS)

- Complete the timing diagram of the circuit shown below. Get the excitation equation for Q. (6 pts)



$$Q_{t+1} \leftarrow a \oplus d \oplus Q_t$$

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



$$Q_{t+1} \leftarrow ab + aQ_t + bQ_t$$

✓ **VHDL Code: Top File**

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity lfsr_mic_rev is
    generic (N: INTEGER:= 4);
    port ( resetn, clock: in std_logic;
          E: in std_logic;
          P: in std_logic_vector (N-1 downto 0);
          Q: out std_logic_vector (N-1 downto 0));
end lfsr_mic_rev;

architecture structural of lfsr_mic_rev is

    component dffe
    port ( d : in STD_LOGIC;
          clrn: in std_logic:= '1';
          prn: in std_logic:= '1';
          clk : in STD_LOGIC;
          ena: in std_logic;
          q : out STD_LOGIC);
    end component;

    signal D, Qt: std_logic_vector (N-1 downto 0);

begin

    D(0) <= Qt(N-1) xor Qt(0) xor P(0);

    g0: for i in N-1 downto 1 generate
        D(i) <= Qt(i-1) xor P(i);
    end generate;

    g1: for i in 0 to N-1 generate
        di: dffe port map (d => D(i), clrn => resetn, prn => '1', clk => clock, ena => E, q => Qt(i));
    end generate;

    Q <= Qt;

end structural;
```

✓ **VHDL Code: D-Type flip flop**

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity dffe is
    port ( d : in STD_LOGIC;
          clrn, prn, clk, ena: in std_logic;
          q : out STD_LOGIC);
end dffe;

architecture behaviour of dffe is

begin
    process (clk, ena, prn, clrn)
    begin
        if clrn = '0' then q <= '0';
        elsif prn = '0' then q <= '1';
        elsif (clk'event and clk='1') then
            if ena = '1' then q <= d; end if;
        end if;
    end process;
end behaviour;
```

✓ VHDL Tesbench:

```

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY tb_lfsr_mic_rev IS
    generic (N: integer:= 4);
END tb_lfsr_mic_rev;

ARCHITECTURE behavior OF tb_lfsr_mic_rev IS
    component lfsr_mic_rev
        port ( resetn, clock: in std_logic;
              E: in std_logic;
              P: in std_logic_vector (N-1 downto 0);
              Q: out std_logic_vector (N-1 downto 0));
    end component;

-- Inputs
signal E : std_logic := '0';
signal resetn : std_logic := '0';
signal clock : std_logic := '0';
signal P: std_logic_vector (N-1 downto 0):= (others => '0');

-- Outputs
signal Q : std_logic_vector(N-1 downto 0);

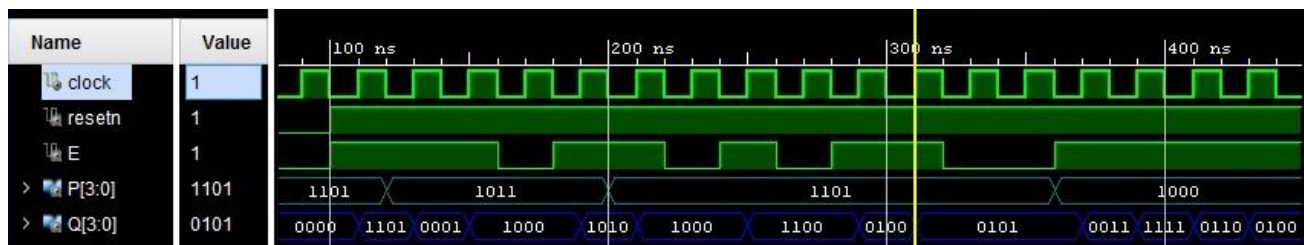
-- Clock period definitions
constant T : time := 20 ns;

BEGIN
    -- Instantiate the Unit Under Test (UUT)
    uut: lfsr_mic_rev PORT MAP (resetn => resetn, clock => clock, E => E, P => P, Q => Q);

    -- Clock process definitions
    clock_process :process
    begin
        clock <= '0'; wait for T/2;
        clock <= '1'; wait for T/2;
    end process;

    -- Stimulus process
    stim_proc: process
    begin
        resetn <= '0'; P <= "1101"; wait for 100 ns;
        resetn <= '1';
        P <= "1101"; E <= '1'; wait for T;
        P <= "1011"; E <= '1'; wait for T;
        P <= "1011"; E <= '1'; wait for T;
        P <= "1011"; E <= '0'; wait for T;
        P <= "1011"; E <= '1'; wait for T;
        P <= "1101"; E <= '1'; wait for T;
        P <= "1101"; E <= '0'; wait for T;
        P <= "1101"; E <= '1'; wait for T;
        P <= "1101"; E <= '0'; wait for T;
        P <= "1101"; E <= '1'; wait for 2*T;
        P <= "1101"; E <= '0'; wait for 2*T;
        P <= "1000"; E <= '1'; wait for 4*T;
        wait;
    end process;

END;
```



PROBLEM 7 (8 PTS)

- Attach your Project Status Report (no more than 1 page, single-spaced, 2 columns, only one submission per group). This report should contain the initial status of your project. For formatting, use the provided template (Final Project - Report Template.docx). The sections included in the template are the ones required in your Final Report. At this stage, you are only required to:
 - ✓ Include a (draft) project description and title.
 - ✓ Include a draft Block Diagram of your hardware architecture.
- As a guideline, the figure shows a simple Block Diagram. There are input and output signals, as well as internal components along with their interconnection.
 - ✓ At this stage, only a rough draft is required. There is no need to go into details: it is enough to show the tentative top-level components that would constitute your system as well as the tentative inputs and outputs.
- Only student is needed to attach the report (make sure to indicate all the team members).

