**No : Answers** **Name : Solutions**

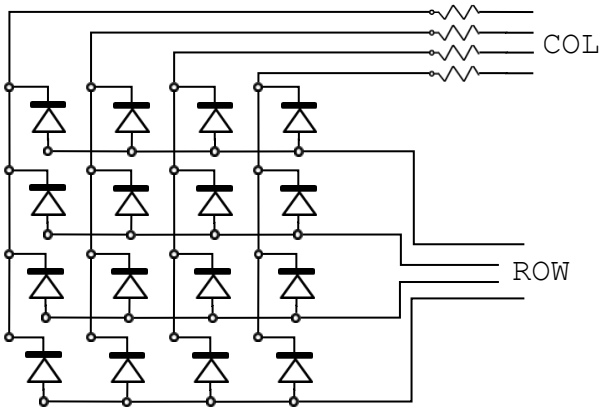
**Eskişehir Osmangazi University Faculty of Engineering and Architecture**

**Department of Computer Engineering** 13.11.2018

"*Introduction To VHDL-FPGA*" Midterm

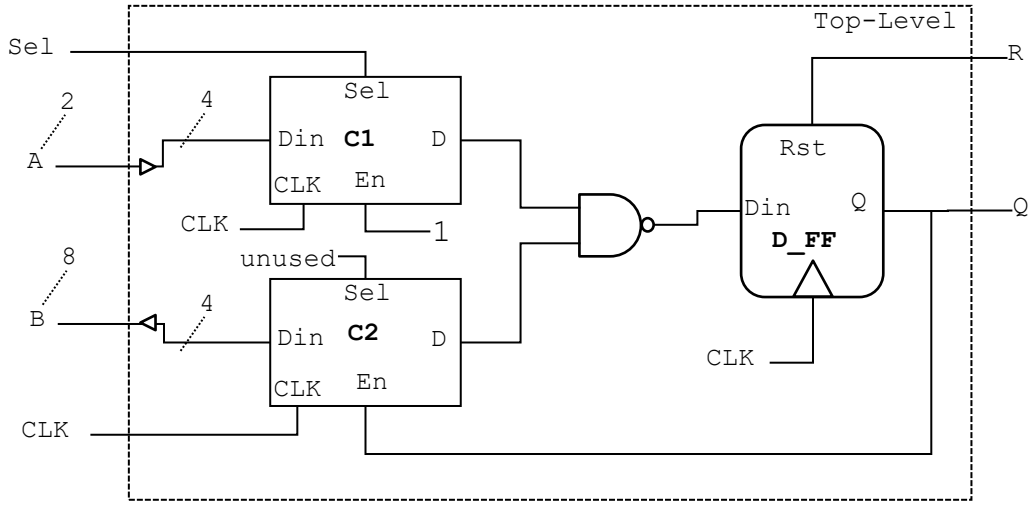
**Note : Books, notes, computers are allowed, communication of all kind is prohibited. 75 minutes.**

1. A LED matrix is given as shown. You are asked to drive these LEDS to display any pattern given in a 16-bits signal. Since the row and column lines are shared/common, you need to light-up a row of LEDs for 10-50 ms and move on to next row for doing the same thing. This repeated scan of rows will trick human eye/brain into seeing the pattern.



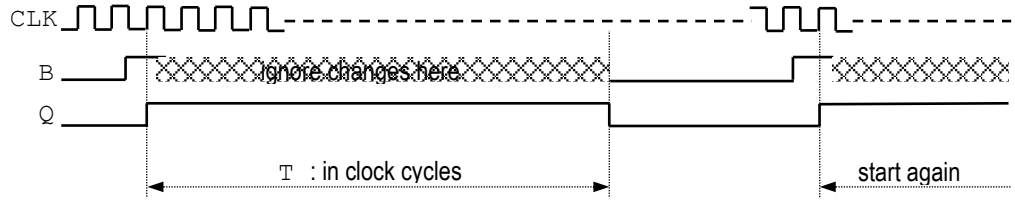

```
entity SCAN is Port (
   CLK  : in  STD_LOGIC; -- 1 MHz clock signal
   PTR  : in  STD_LOGIC_VECTOR(15 downto 0); -- 4x4 pattern
   ROW  : out STD_LOGIC_VECTOR(3 downto 0);  -- one hot row selection
   COL  : out STD_LOGIC_VECTOR(3 downto 0)); -- inverted col pattern
end SCAN;

architecture SCAN of SCAN is
  signal cntr : STD_LOGIC_VECTOR(16 downto 0);
begin
  process(CLK, cntr) is begin
    if(RISING_EDGE(CLK)) then cntr <= cntr + 1; end if;
  end process;
  with cntr(16 downto 15) select
    ROW <= "0001" when "00",
           "0010" when "01",
           "0100" when "10",
           "1000" when others;
  with cntr(16 downto 14) select
    COL <= not PTR(3  downto 0)  when "00",
           not PTR(7  downto 4)  when "01",
           not PTR(11 downto 8)  when "10",
           not PTR(15 downto 12) when others;

end SCAN;
```

2. The following partial code and a block diagram are given. Complete the top-level circuit using VHDL. MSBs of A and B inputs will be used to feed Din. Remaining LSB bits of Din (if any) should be connected to 0.




```
entity TOP is Port (
  CLK, R : in  STD_LOGIC;
  A      : in  STD_LOGIC_VECTOR(1 downto 0);
  B      : in  STD_LOGIC_VECTOR(7 downto 0);
  Sel, Q : out STD_LOGIC);
end TOP;
architecture TOP of TOP is
  component SUB is Port(
    CLK, En : in STD_LOGIC;
    D, Sel  : out STD_LOGIC;
    Din     : in STD_LOGIC_VECTOR(3 downto 0))
  end component;
  Qx,D1,D2 : STD_LOGIC;
begin
  Q <= Qx;
  C1: SUB port map(
    CLK => CLK,
    En  => '1',
    D   => D1,
    Sel => Sel,
    Din => (A & "00")
  );
  C2: SUB port map(
    CLK => CLK,
    En  => Qx,
    D   => D2,
    Sel => open,
    Din => B(7 downto 4)
  );
process(CLK,R,D1,D2) is begin
  if(RISING_EDGE(CLK)) then
    if(R='1') then Qx <= '0';
    else Qx <= not(D1 and D2);
  end if;
end process;
end TOP;
```

3. Design a fully synchronous timer circuit that activates the output Q for T clock cycles when the input B is '1'. Timer should ignore changes on input until the end of the period. If, at the end of the period, B is '1', timer should start another cycle without lowering Q. That is, when Q goes high, it stays high for nT clocks where n is an integer, checking the input B at the end of every cycle.




```
entity TMR is Port (
  CLK, B : in  STD_LOGIC;
  Q      : out STD_LOGIC);
end TMR;
architecture TMR of TMR is
  constant T : integer := 100; -- for example
  signal cntr : integer := 0;
  signal Qx : STD_LOGIC;
begin
  Q <= Qx;
  process(CLK,B,Qx,cntr) is begin
    if(RISING_EDGE(CLK)) then
      if((cntr=T)or(Qx='0')) then
        cntr <= 0;
        Qx <= B;
      else
        cntr <= cntr + 1;
      end if;
    end if;
  end process;
end TMR;
```