No : Answers **Name : Solutions**

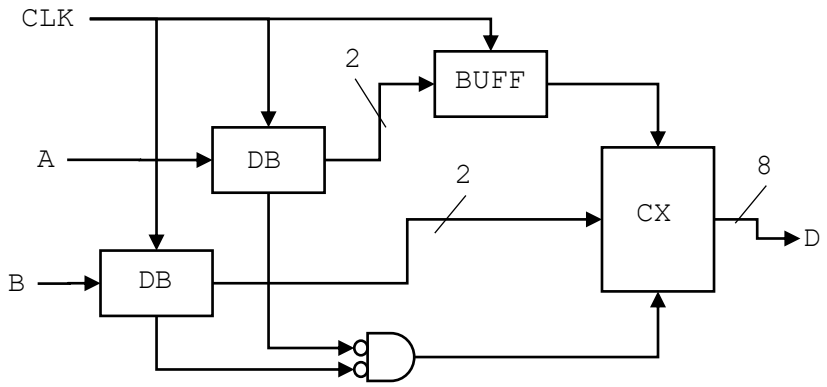
Eskişehir Osmangazi University Faculty of Engineering and Architecture

Department of Electrical Engineering & Electronics 18.11.2021

"*Introduction To FPGA-VHDL*" Midterm Exam.

Note : Books, notes, computers are allowed, communication of all kind is prohibited. 90 (ninety) minutes.

1. Complete VHDL top-level entity for the following block diagram and components.

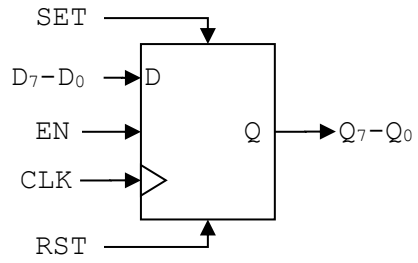



```
entity MCIR is Port (
  A, B : in  STD_LOGIC;
     D : out STD_LOGIC_VECTOR(7 downto 0);
   CLK : in  STD_LOGIC);
end MCIR;

architecture MCIR of MCIR is
  component DB is port (
         X : out STD_LOGIC_VECTOR(1 downto 0);
    I, CLK : in  STD_LOGIC;
         S : out STD_LOGIC);
  end component;
  component BUFF is port (
    CLK : in  STD_LOGIC;
      X : in  STD_LOGIC_VECTOR(1 downto 0);
      Y : out STD_LOGIC_VECTOR(1 downto 0));
  end component;
  component CX is port (
     XY : in  STD_LOGIC;
    X,Y : in  STD_LOGIC_VECTOR(1 downto 0);
      Z : out STD_LOGIC_VECTOR(7 downto 0));
  end component;

  signal X1, X2, X3 : STD_LOGIC_VECTOR(1 downto 0);
  signal S1, S2, XY : STD_LOGIC;
begin

  XY <= (not S1)and(not S2);
  DB1: DB port map (CLK=>CLK, I=>A, S=>S1, X=>X1);
  DB2: DB port map (CLK=>CLK, I=>B, S=>S2, X=>X2);
  BUFF1: BUFF port map(CLK=>CLK, X=>X1, Y=>X3);
  CX1: CX port map(X=>X2, Y=>X3, XY=>XY, Z=>D);

end MCIR;
```

2. When the input `btn1` goes high, output `A` goes high and stays high for 2 seconds. If `btn2` goes high during the first 1 second output `B` is set to high too. However, if `btn2` goes high within the second half of the 2 seconds, output `C` is set to high. Inputs are disabled after their first rises until the 2 seconds cycle completes. After 2 seconds, all outputs are set to low and inputs are enabled. `bnt2` has no effect unless the cycle is started by `btn1`. System operates on a 1 MHz `clk` input. Design the circuit using VHDL.

```
entity BSEQ is Port (
  btn1, btn2  : in  STD_LOGIC;
      A, B, C : out STD_LOGIC;
         CLK  : in  STD_LOGIC); -- 1 MHz
end BSEQ;

architecture BSEQ of BSEQ is
  signal cntr: integer range 0 to 2000000;
  signal Bx, Cx : STD_LOGIC;
begin
  B <= Bx; C <= Cx;
  process(CLK, btn1, btn2, cntr, Bx, Cx) is begin
    if(rising_edge(CLK)) then
      if((cntr=0)and(btn1='1')) then
        cntr <= 1; A <= '1';
      else
        if(cntr=2000000) then
          cntr <= 0; Bx <= '0'; Cx <= '0';
        else
          cntr <= cntr+1;
        end if;
      end if;
      if((Bx&Cx="00")and(cntr>0)and(btn2='1')) then
        if(cntr<1000000) then
          Bx <= '1';
        else
          Cx <= '1';
        end if;
      end if;
    end if;
  end process;
end BSEQ;
```

3. Design the following 8-bits up-counter using VHDL.



CLK	EN	RST	SET	Q →
x	0	x	x	Q
↑	1	1	x	0...0
↑	1	0	1	D_7...D_0
↑	1	0	0	Q+1

```
entity UPCNT is Port (
            CLK : in  STD_LOGIC;
              D : in  STD_LOGIC_VECTOR(7 downto 0);
  EN, SET, RST : in  STD_LOGIC;
              Q : out STD_LOGIC_VECTOR(7 downto 0));
end UPCNT;

architecture UPCNT of UPCNT is
  signal Qx : STD_LOGIC_VECTOR(7 downto 0);
begin
  Q <= Qx;
  process(CLK,RST,SET,EN,D,Qx) is begin
    if(rising_edge(CLK)) then
      if(EN='1') then
        if(RST='1') then
          Qx <= (others=>'0');
        elsif(SET='1') then
          Qx <= D;
        else
          Qx <= Qx +1;
        end if;
      end if;
    end if;
  end process;
end UPCNT;
```