Εργαστηριακή Άσκηση 5

ΧΡΗΣΤΟΣ ΚΟΥΤΟΥΛΗΣ ΑΜ:5064
ΜΑΡΙΑ ΚΑΤΩΛΗ, ΑΜ:5083
ΟΜΑΔΑ 18

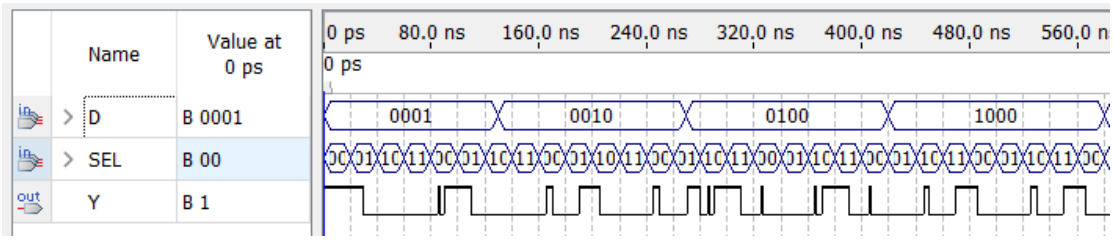
Μέρος 1ο: Συνδυαστικά κυκλώματα

Ερώτημα 1ο

Ο κώδικας για τον πολυπλέκτη 4 σε 1:

```
library IEEE;
use IEEE.std_logic_1164.all;
entity MUX4_1  is
port  (   D :  in std_logic_vector  (3 downto 0);
SEL:  in std_logic_vector  (1 downto 0);
Y: out   std_logic);
end   MUX4_1;
architecture   RTL4_1 of   MUX4_1   is
begin
Y  <= D(0)  when  SEL="00" else  D(1)  when  SEL="01" else  D(2)  when  SEL="10" else  D(3);
end RTL4_1;
```

Η εξομοίωση:



Ερώτημα 2ο:

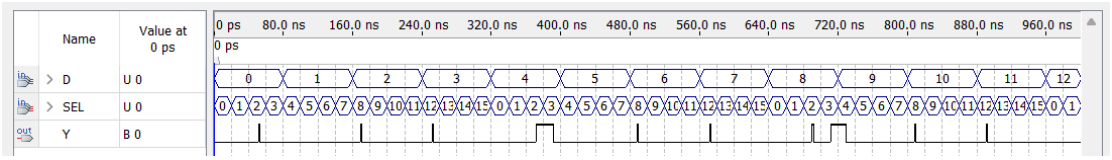
Ο κώδικας για τον πολυπλέκτη 16 σε 1:

```
library IEEE;
use IEEE.std_logic_1164.all;

entity MUX16_1 is
port  (   D :  in std_logic_vector  (15   downto 0);
SEL:  in std_logic_vector  (3 downto 0);
Y: out   std_logic );
end   MUX16_1;

architecture   RTL   of MUX16_1  is
component MUX4_1
port  (   D :  in std_logic_vector  (3 downto 0);
SEL:  in std_logic_vector  (1 downto 0);
Y: out   std_logic );
end   component;
signal   F :   std_logic_vector  (3 downto 0);
begin
u0:   MUX4_1   port  map   (D => D(3   downto   0),    SEL  => SEL(1 downto   0),   Y  => F(0));
u1:   MUX4_1   port  map   (D => D(7   downto   4),    SEL  => SEL(1 downto   0),   Y  => F(1));
u2:   MUX4_1   port  map   (D => D(11  downto   8),    SEL  => SEL(1 downto   0),   Y  => F(2));
u3:   MUX4_1   port  map   (D => D(15  downto   12), SEL => SEL(1 downto   0),   Y  => F(3));
u4:   MUX4_1   port  map   (D => F(3   downto   0),    SEL  => SEL(3 downto   2),   Y  => Y);
end RTL;
```

Η εξομοίωση:



Η μέγιστη καθυστέρηση είναι 12.247ns:

Propagation Delay

	Input Port	Output Port	RR	RF	FR	FF
1	D[0]	Y	11.859	11.859	11.859	11.859
2	D[1]	Y	11.451			11.451
3	D[2]	Y	11.673	11.673	11.673	11.673
4	D[3]	Y	11.170			11.170
5	D[4]	Y	11.852	11.852	11.852	11.852
6	D[5]	Y	12.155	12.155	12.155	12.155
7	D[6]	Y	8.937			8.937
8	D[7]	Y	11.743			11.743
9	D[8]	Y	11.621	11.621	11.621	11.621
10	D[9]	Y	11.452			11.452
11	D[10]	Y	11.756			11.756
12	D[11]	Y	11.090			11.090
13	D[12]	Y	11.636	11.636	11.636	11.636
14	D[13]	Y	11.813			11.813
15	D[14]	Y	11.957			11.957
16	D[15]	Y	11.809			11.809
17	SEL[0]	Y	12.247	12.247	12.247	12.247
18	SEL[1]	Y	9.476	9.476	9.476	9.476
19	SEL[2]	Y	11.246	11.246	11.246	11.246
20	SEL[3]	Y	11.221	11.707	11.707	11.221

<u>Ερώτημα 3ο</u>:

Ο κώδικας για τον πλήρη αθροιστή:

```
1   library IEEE;
2   use IEEE.std_logic_1164.all;
3   entity FULLADDER is
4   port (A,B,Cin : in std_logic;
5         Y, Cout: out std_logic);
6   end FULLADDER;
7
8   architecture behaviour of FULLADDER is
9   begin
10  Y<= (A XOR B) XOR Cin;
11  Cout<=((A XOR B) AND Cin) OR (A AND B);
12  end behaviour;
13
```

Ο κώδικας για τον αθροιστή 8 δυαδικών ψηφίων:

```
1  library IEEE;
2  use IEEE.std_logic_1164.all;
3  entity Adder8 is
4  port  (   A,   B :    in   std_logic_vector  (7 downto 0);
5     Cin   :   in   std_logic;
6     Y:  out std_logic_vector   (7 downto 0);
7     Cout:  out  std_logic );
8
9  end Adder8;
10 architecture RTL of Adder8 is
11 component FULLADDER
12 port (A,B,Cin : in std_logic;
13       Y, Cout: out std_logic);
14 end component;
15 signal F: std_logic_vector (6 downto 0);
16 begin
17    u0: FULLADDER port map(A(0),B(0),Cin,Y(0),F(0));
18    u1: FULLADDER port map(A(1),B(1),F(0),Y(1),F(1));
19    u2: FULLADDER port map(A(2),B(2),F(1),Y(2),F(2));
20    u3: FULLADDER port map(A(3),B(3),F(2),Y(3),F(3));
21    u4: FULLADDER port map(A(4),B(4),F(3),Y(4),F(4));
22    u5: FULLADDER port map(A(5),B(5),F(4),Y(5),F(5));
23    u6: FULLADDER port map(A(6),B(6),F(5),Y(6),F(6));
24    u7: FULLADDER port map(A(7),B(7),F(6),Y(7),Cout);
25 end RTL;
```

Μέρος 2ο:

Ερώτημα 1ο:

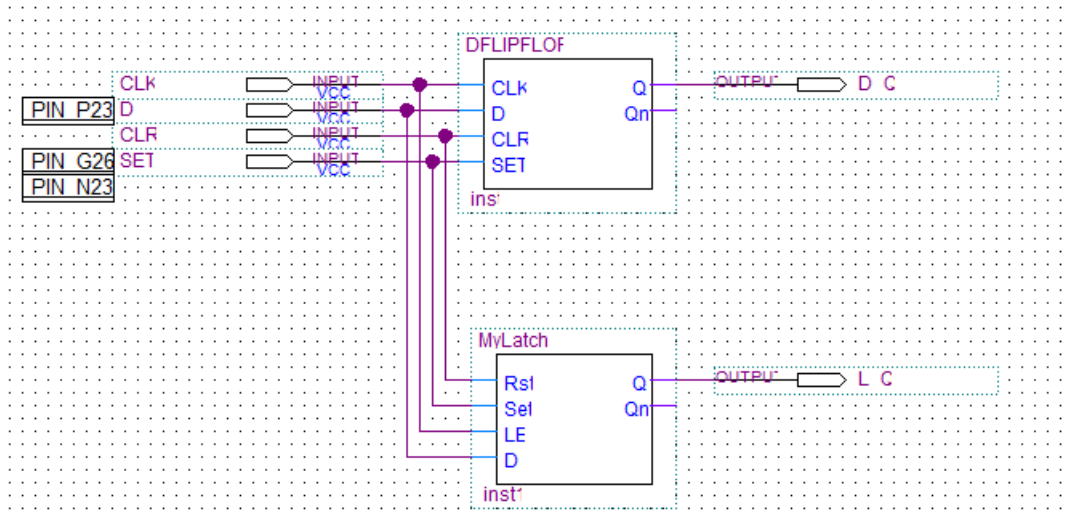
Ο κώδικας για το D Flip Flop:

```
1  library  IEEE;
2  use   IEEE.std_logic_1164.all;
3  entity DFLIPFLOP is
4     port (    CLK, D, CLR, SET, EN:  in    std_logic;
5               Q,   Qn    :  out   std_logic );
6  end DFLIPFLOP;
7
8  architecture   RTL   of DFLIPFLOP is
9         signal  DFF   :  std_logic;
10 begin
11        seq0 :  process (CLK, CLR, SET, EN )
12        begin
13                if (EN='0') then DFF<='0';
14                elsif (CLR='1')   then DFF<='0';
15                elsif (SET='1')   then DFF<='1';
16                elsif (CLK'event  and   CLK   =  '1')  then  DFF   <=D;
17             end   if;
18        end  process;
19        Q <= DFF;  Qn <= not   DFF;
20 end RTL;
```

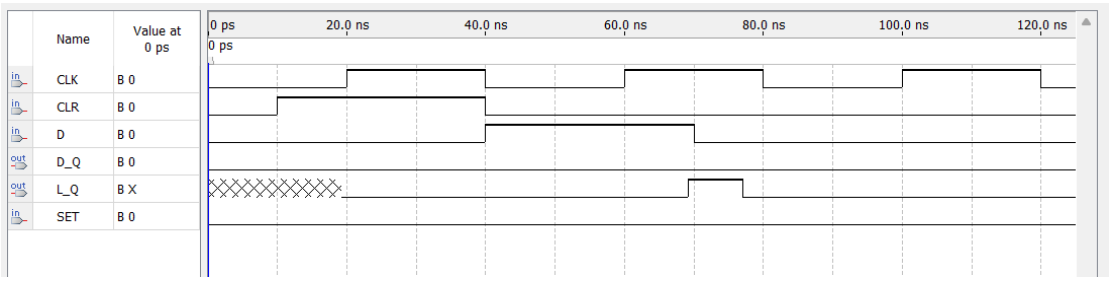
Ο κώδικας για το latch:

```
1  library IEEE;
2  use IEEE.std_logic_1164.all;
3  entity MyLatch is
4     port ( Rst, Set, LE, D : in std_logic;
5         Q, Qn: out std_logic );
6  end MyLatch;
7
8  architecture RTL of MyLatch is
9  signal FF: std_logic;
10 begin
11    seq0 : process (Rst, Set, D, LE)
12    begin
13       if Rst = '1' then FF <= '0';
14       elsif Set = '1' then FF <= '1';
15       elsif LE = '1' then FF <= D;
16       end if;
17    end process;
18    Q <= FF;
19    Qn <= not FF;
20 end RTL;
```

Το σχηματικό:



Η εξομοίωση:

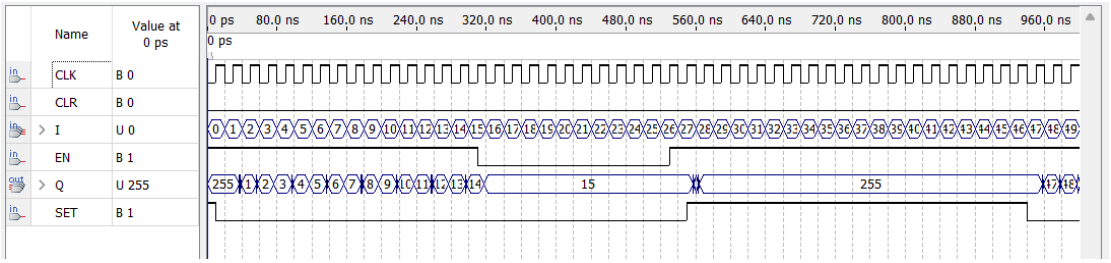


Ερώτημα 2ο:

Ο κώδικας:

```
1  library  IEEE;
2  use   IEEE.std_logic_1164.all;
3
4  entity Reg8 is
5        port (CLK , CLR, SET, EN:  in    std_logic;
6              I: in std_logic_vector(7 downto 0);
7              Q: out   std_logic_vector(7 downto 0));
8  end Reg8;
9
10
11 architecture RTL of Reg8 is
12       signal DFF: std_logic_vector(7 downto 0);
13   begin
14       seq0 : process (CLK, CLR, SET)
15          begin
16                if (CLR='1') then DFF<=(7 downto 0 => '0');
17                elsif (SET='1') then DFF<=(7 downto 0 => '1');
18                elsif (CLK'event  and   CLK   =  '1' and EN='1') then DFF<=I;
19                end   if;
20       end   process;
21       Q  <= DFF;
22 end RTL;
```

Η εξομοίωση:



Τα αποτελέσματα της στατικής χρονικής ανάλυσης:

	Input Port	Output Port	RR	RF	FR	FF
1	CLR	Q[0]		11.393	11.393	
2	CLR	Q[1]		11.887	11.887	
3	CLR	Q[2]		11.781	11.781	
4	CLR	Q[3]		11.579	11.579	
5	CLR	Q[4]		11.589	11.589	
6	CLR	Q[5]		11.605	11.605	
7	CLR	Q[6]		11.885	11.885	
8	CLR	Q[7]		11.698	11.698	
9	SET	Q[0]	12.005			12.005
10	SET	Q[1]	12.499			12.499
11	SET	Q[2]	12.393			12.393
12	SET	Q[3]	12.191			12.191
13	SET	Q[4]	12.201			12.201
14	SET	Q[5]	12.217			12.217
15	SET	Q[6]	12.497			12.497
16	SET	Q[7]	12.310			12.310

Ερώτημα 3ο:

Ο κώδικας:

```
1  library  IEEE;
2  use   IEEE.std_logic_1164.all;
3  use   IEEE.std_logic_unsigned.all;
4
5  entity Counter8 is
6        port (CLK , CLR, SET, EN:  in    std_logic;
7              Q: out   std_logic_vector(7 downto 0));
8  end Counter8;
9
10
11 architecture RTL of Counter8 is
12       signal sum: std_logic_vector(7 downto 0);
13    begin
14       seq0 : process (CLK, CLR, SET)
15          begin
16                if (CLR='1') then sum<=(7 downto 0 => '0');
17                elsif (SET='1') then sum<=(7 downto 0 => '1');
18                elsif (CLK'event and CLK='1' and EN='1')  then  sum<=(sum + '1');
19                end   if;
20       end   process;
21       Q<=sum;
22 end RTL;
```

Η εξομοίωση: