دانشکده کامپیوتر

آزمایشگاه معماری کامپیوتر
آزمایش دوم

دکتر محبتی

عرشیا آرین نژاد، حوریه سبزواری، الناز رضایی

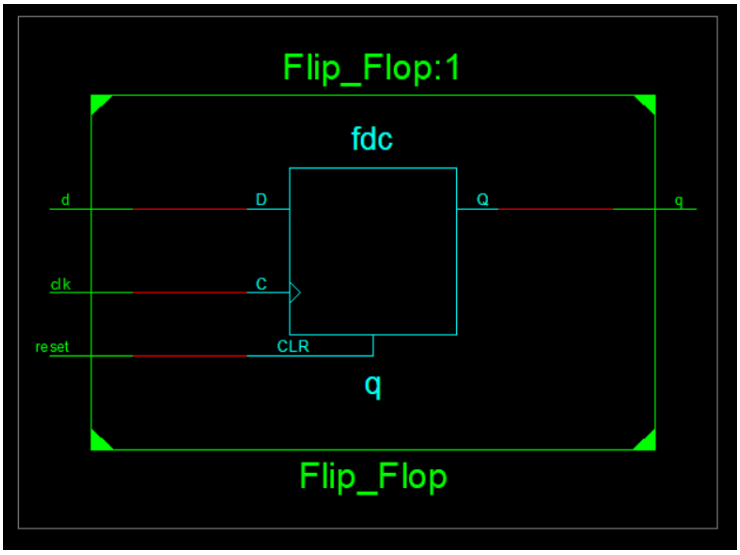
فروردین ۱۴۰۱

هدف آزمایش:

در این آزمایش با المان های نگهدارنده حساس به سطح و لبه مانند T و JK و D فلیپ فلاپ آشنا شدیم. همچنین نحوه نوشتن TESTBENCH و نحوهٔ صحت سنجی کد خود را یاد گرفتیم.

D Flip Flop:

طراحی شماتیک طرح به صورت فیزیکی:



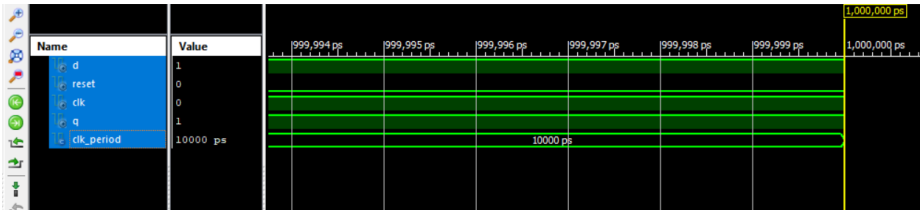
پیاده‌سازی مدارها با استفاده از زبان‌های توصیف:

```
32  entity Flip_Flop is
33      Port ( d : in  STD_LOGIC;
34             reset : in  STD_LOGIC;
35             clk : in  STD_LOGIC;
36             q : out  STD_LOGIC);
37  end Flip_Flop;
38
39  architecture Behavioral of Flip_Flop is
40
41  begin
42     q<='0' when reset ='1' else
43     d when clk'event and clk = '1';
44
45  end Behavioral;
```

Testbench:

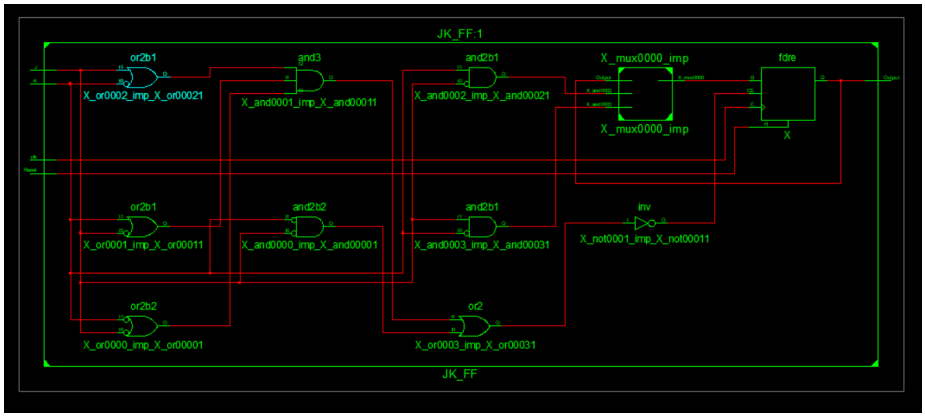
```
35  ENTITY Simulate IS
36  END Simulate;
37
38  ARCHITECTURE behavior OF Simulate IS
39
40      -- Component Declaration for the Unit Under Test (UUT)
41
42      COMPONENT Flip_Flop
43      PORT(
44           d : IN  std_logic;
45           reset : IN  std_logic;
46           clk : IN  std_logic;
47           q : OUT  std_logic
48          );
49      END COMPONENT;
50
51
52     --Inputs
53     signal d : std_logic := '0';
54     signal reset : std_logic := '0';
55     signal clk : std_logic := '0';
56
57     --Outputs
58     signal q : std_logic;
59
60     -- Clock period definitions
61     constant clk_period : time := 10 ns;
62
63  BEGIN
64
65     -- Instantiate the Unit Under Test (UUT)
66     uut: Flip_Flop PORT MAP (
67            d => d,
68            reset => reset,
69            clk => clk,
70            q => q
71          );
```

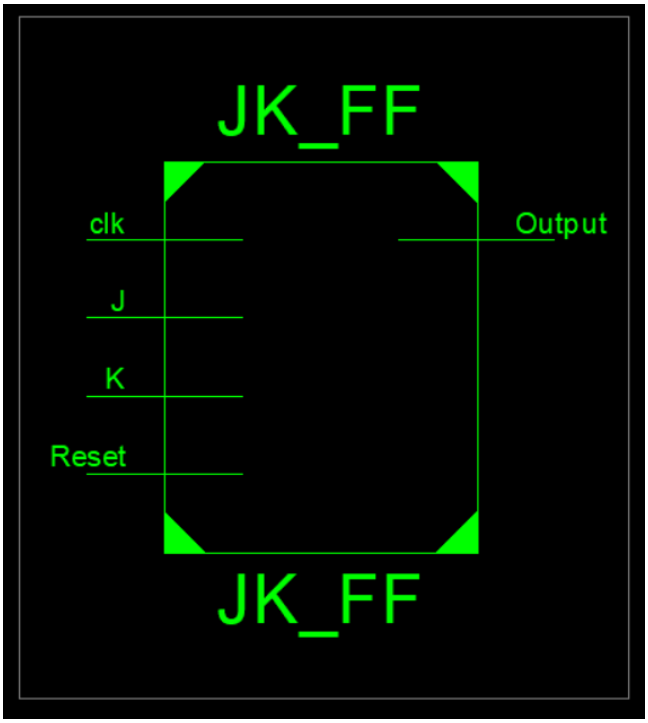
```
72
73    -- Clock process definitions
74    clk_process :process
75    begin
76       clk <= '0';
77       wait for clk_period/2;
78       clk <= '1';
79       wait for clk_period/2;
80    end process;
81
82
83    -- Stimulus process
84    stim_proc: process
85    begin
86       -- hold reset state for 100 ns.
87       wait for 100 ns;
88       d <= '1', 'X' after 22 ns, '1' after 24 ns, 'U' after 35 ns , '1' after 40 ns;
89       reset <= '1' after 50 ns , '0' after 60 ns;
90       wait for clk_period*10;
91
92       -- insert stimulus here
93
94       wait;
95    end process;
96
97 END;
98
```



JK Flip Flop:

طراحی شماتیک طرح به صورت فیزیکی:





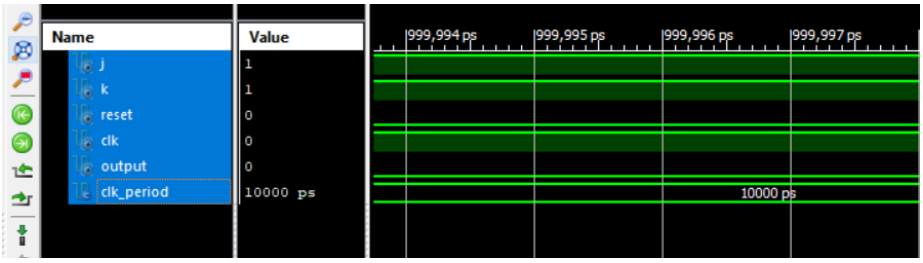
پیاده‌سازی مدارها با استفاده از زبان‌های توصیف:

```
32  entity JK_FF is
33      port( J,K: in  std_logic;
34            Reset: in std_logic;
35            clk: in std_logic;
36            Output: out std_logic);
37  end JK_FF;
38
39  architecture Behavioral of JK_FF is
40     signal X: std_logic;
41  begin
42  process (clk)
43     begin
44        if rising_edge(clk) then
45           if Reset='1' then
46              X <= '0';
47           else
48              if (J='0' and K='0') then
49                 X <= X;
50              elsif (J='0' and K='1') then
51                 X <= '0';
52              elsif (J='1' and K='0') then
53                 X <= '1';
54              elsif (J='1' and K='1') then
55                 X <= not (X);
56              end if;
57           end if;
58        end if;
59     end process;
60     Output <= X;
61
62  end Behavioral;
```

Testbench:

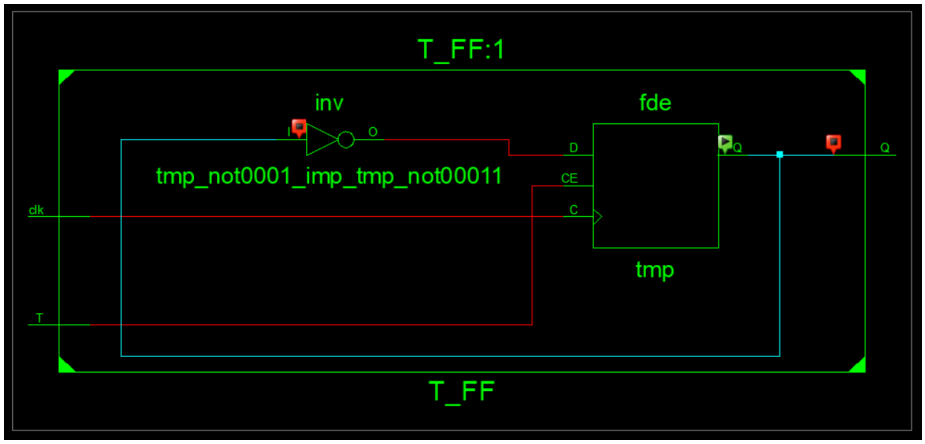
```
35  ENTITY Simulate IS
36  END Simulate;
37
38  ARCHITECTURE behavior OF Simulate IS
39
40      -- Component Declaration for the Unit Under Test (UUT)
41
42      COMPONENT JK_FF
43      PORT(
44           J : IN  std_logic;
45           K : IN  std_logic;
46           Reset : IN  std_logic;
47           clk : IN  std_logic;
48           Output : OUT  std_logic
49          );
50      END COMPONENT;
51
52
53     --Inputs
54     signal J : std_logic := '0';
55     signal K : std_logic := '0';
56     signal Reset : std_logic := '0';
57     signal clk : std_logic := '0';
58
59     --Outputs
60     signal Output : std_logic;
61
62     -- clk period definitions
63     constant clk_period : time := 10 ns;
64
65  BEGIN
66
67     -- Instantiate the Unit Under Test (UUT)
68     uut: JK_FF PORT MAP (
69            J => J,
70            K => K,
71            Reset => Reset,
72            clk => clk,
73            Output => Output
74          );
```

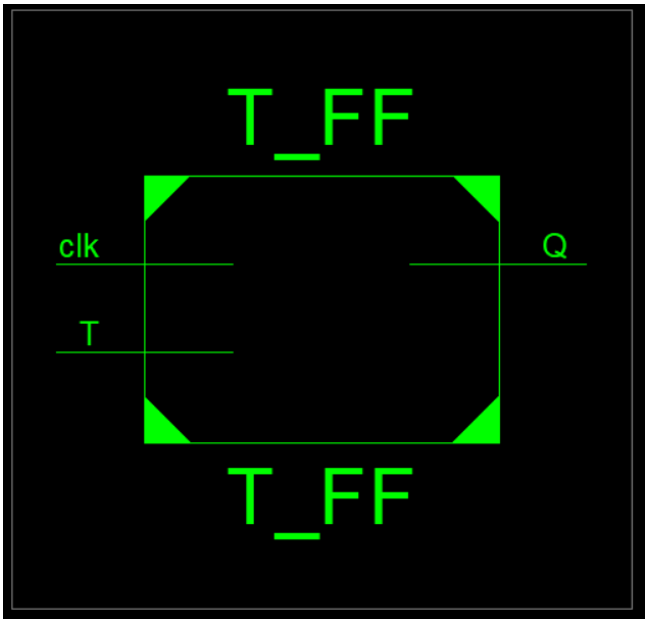
```
75
76      -- clk process definitions
77      clk_process :process
78      begin
79          clk <= '0';
80          wait for clk_period/2;
81          clk <= '1';
82          wait for clk_period/2;
83      end process;
84
85
86      -- Stimulus process
87      stim_proc: process
88      begin
89          -- hold reset state for 100 ns.
90          J <= '0';
91          k <= '0';
92          wait for 100 ns;
93          J <= '1';
94          k <= '0';
95          wait for 100 ns;
96          J <= '0';
97          k <= '1';
98          wait for 100 ns;
99          J <= '1';
100         k <= '1';
101         wait for 100 ns;
102         wait for clk_period*10;
103
104         -- insert stimulus here
105
106         wait;
107     end process;
108
109 END;
```



T Flip Flop:

طراحی شماتیک طرح به صورت فیزیکی:





پیاده‌سازی مدارها با استفاده از زبان‌های توصیف:

```
32  entity T_FF is
33      port( T: in std_logic;
34            clk: in std_logic;
35            Q: out std_logic);
36  end T_FF;
37
38  architecture Behavioral of T_FF is
39     signal tmp: std_logic := '0';
40  begin
41     process (clk)
42     begin
43        if clk'event and clk='1' then
44
45           if T='0' then
46              tmp <= tmp;
47           elsif T='1' then
48              tmp <= not (tmp);
49           end if;
50        end if;
51     end process;
52  Q <= tmp;
53
54  end Behavioral;
```

Testbench:

```
35  ENTITY Simulate IS
36  END Simulate;
37
38  ARCHITECTURE behavior OF Simulate IS
39
40      -- Component Declaration for the Unit Under Test (UUT)
41
42      COMPONENT T_FF
43      PORT(
44           T : IN  std_logic;
45           clk : IN  std_logic;
46           Q : OUT  std_logic
47          );
48      END COMPONENT;
49
50
51     --Inputs
52     signal T : std_logic := '0';
53     signal clk : std_logic := '0';
54
55     --Outputs
56     signal Q : std_logic;
57
58     -- Clock period definitions
59     constant clk_period : time := 10 ns;
60
61  BEGIN
62
63     -- Instantiate the Unit Under Test (UUT)
64     uut: T_FF PORT MAP (
65            T => T,
66            clk => clk,
67            Q => Q
68          );
```

```
69
70      -- Clock process definitions
71      clk_process :process
72      begin
73          clk <= '0';
74          wait for clk_period/2;
75          clk <= '1';
76          wait for clk_period/2;
77      end process;
78
79
80      -- Stimulus process
81      stim_proc: process
82      begin
83          -- hold reset state for 100 ns.
84          T <='0';
85          wait for 100 ns;
86          T <='1';
87          wait for 100 ns;
88          T <='0';
89          wait for 100 ns;
90          T <='1';
91          wait for 100 ns;
92
93          wait for clk_period*10;
94
95          -- insert stimulus here
96
97          wait;
98      end process;
99
100 END;
101
```