CE 1911

Homework 4

1 – Write behavioral VHDL code for a D-FF with asynchronous resetb and synchronous set. Provide code, test-bench and a simulation 50pts

```
--
-- created 2/1/2018
-- tj
--
-- rev 0
----------------------------------------
--
-- special DFF with synchronous set
--
----------------------------------------
--
-- Inputs: clk, rstb, din, set
-- Outputs: q
--
----------------------------------------
library ieee;
use ieee.std_logic_1164.all;

entity dffwS is
    port (
      i_clk :     in std_logic;
      i_rstb :    in std_logic;
      i_din :     in std_logic;
      i_set :     in std_logic;

      o_q :       out std_logic
    );
end entity;

architecture behavioral of dffwS is
```

architecture
~14 lines

1 – Write behavioral VHDL code for a D-FF with asynchronous resetb and synchronous set. Provide code, test-bench and a simulation 50pts

testbench

```
--
-- created: 1/26/18
-- by: johnsontimoj
-- rev: 0
--
-- testbench for dff with set
-- of dffwS.vhdl
--
---------------------------------
library ieee;
use ieee.std_logic_1164.all;

entity dffwS_tb is
    -- no entry - testbench
end entity;

architecture testbench of dffwS_tb is
    signal   CLK:      std_logic;
    signal   RSTB:     std_logic;
    signal   DIN:      std_logic;
    signal   SET:      std_logic;

    signal   Q:        std_logic;

    constant PER:      time  := 20 ns;

    ---------------------------------
    -- Component prototype
    ---------------------------------
COMPONENT dffwS
    PORT
    (
       i_clk    :   IN STD_LOGIC;
       i_rstb   :   IN STD_LOGIC;
       i_din    :   IN STD_LOGIC;
       i_set    :   IN STD_LOGIC;
       o_q      :   OUT STD_LOGIC
    );
END COMPONENT;
    ---------------------------------

    begin
```

```
    ---------------------------------
    -- Device under test (DUT)
    ---------------------------------
    DUT: dffwS
       port map(
                i_clk    => CLK,
                i_rstb   => RSTB,
                i_din    => DIN,
                i_set    => SET,

                o_q      => Q
                );


    ---------------------------------
    -- Test processes
    ---------------------------------
```

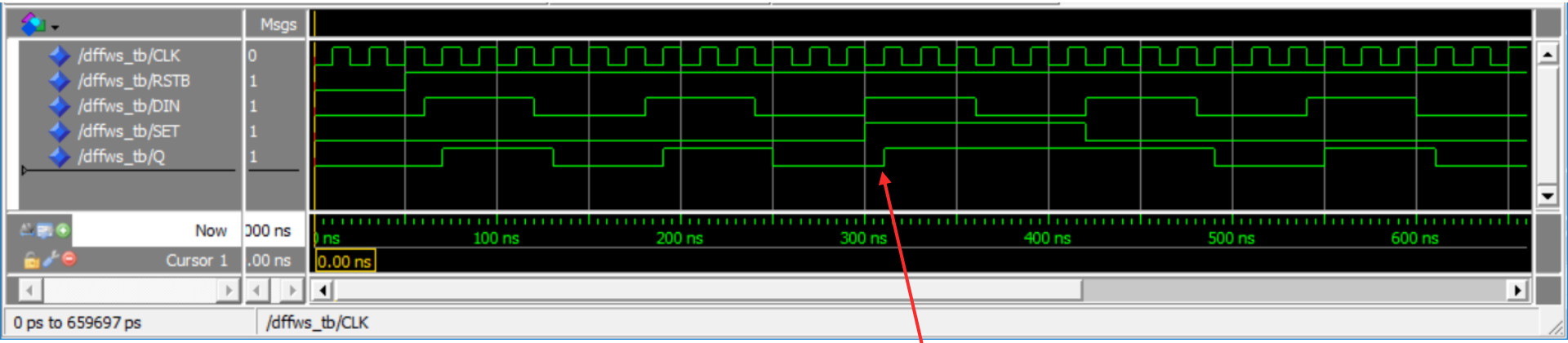
Clock process

Reset process

```
    -- Run process
    run: process      -- note - no sensitivity
    begin
       -------------------------
       -- Initialize  inputs
       DIN <= '0';
       SET <= '0';
       wait for 3*PER;    -- wait for reset

       -- verify normal operation
       DIN <= '1'; wait for 3*PER;
       DIN <= '0'; wait for 3*PER;
       DIN <= '1'; wait for 3*PER;
       DIN <= '0'; wait for 3*PER;

       -- verify set operation
       SET <= '1';
       DIN <= '1'; wait for 3*PER;
       DIN <= '0'; wait for 3*PER;

       -- verify normal operation
       SET <= '0';
       DIN <= '1'; wait for 3*PER;
       DIN <= '0'; wait for 3*PER;
       DIN <= '1'; wait for 3*PER;
       DIN <= '0'; wait for 3*PER;
    end process run;

    ---------------------------------
    -- End test processes
    ---------------------------------


end architecture;
```

1 – Write behavioral VHDL code for a D-FF with asynchronous resetb and synchronous set. Provide code, schematic, test-bench and a simulation 50pts

Simulation Results



data mode Set mode data mode

note – no set until clk - synchronous

2 – Write behavioral VHDL code for a special new FlipFlop you have conceived of – this FlipFlop has a second data input called Toggle(T). When T is low, the DFF operates normally, when T is high, the DFF ignores the D input and synchronously toggles the output. Provide code and a simulation 50pts

```
--
-- created 2/1/2018
-- tj
--
-- rev 0
----------------------------------------
--
-- special DFF with toggle mode
--
----------------------------------------
--
-- Inputs: clk, rstb, din, t
-- Outputs: q
--
----------------------------------------
library ieee;
use ieee.std_logic_1164.all;

entity dffwT is
    port (
      i_clk :  in std_logic;
      i_rstb : in std_logic;
      i_din :  in std_logic;
      i_t :    in std_logic;

      o_q :    out std_logic
    );
end entity;

architecture behavioral of dffwT is
```

architecture
~14 lines

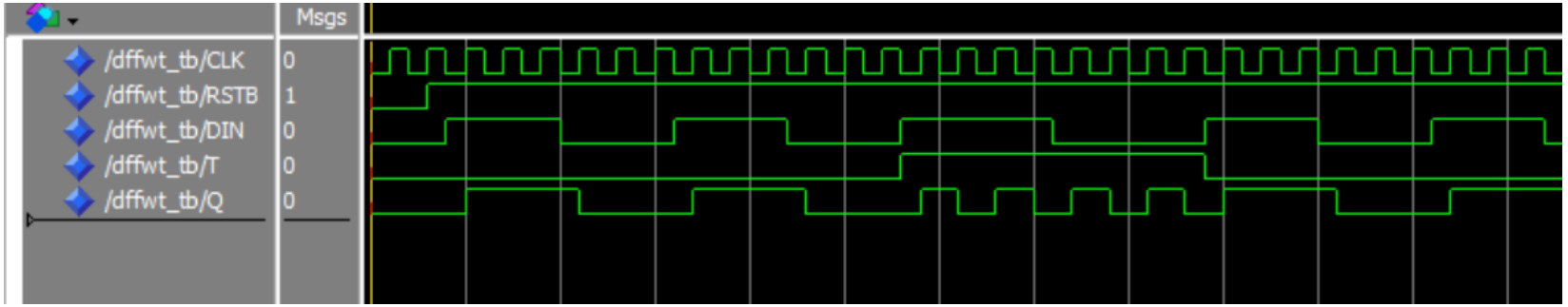
2 – Write behavioral VHDL code for a special new FlipFlop you have conceived of – this FlipFlop has a second data input called Toggle(T). When T is low, the DFF operates normally, when T is high, the DFF ignores the D input and synchronously toggles the output. Provide code and a simulation 50pts

testbench

```
   -- Run process
   run: process      -- note - no sensitivity list allowed
   begin
      ------------------------
      -- Initialize  inputs
      DIN <= '0';
      T <= '0';
      wait for 2*PER;   -- wait for reset

      -- verify normal operation
      DIN <= '1'; wait for 3*PER;
      DIN <= '0'; wait for 3*PER;
      DIN <= '1'; wait for 3*PER;
      DIN <= '0'; wait for 3*PER;

      -- verify toggle operation
      T <= '1';
      DIN <= '1'; wait for 4*PER;
      DIN <= '0'; wait for 4*PER;

      -- verify normal operation
      T <= '0';
      DIN <= '1'; wait for 3*PER;
      DIN <= '0'; wait for 3*PER;
      DIN <= '1'; wait for 3*PER;
      DIN <= '0'; wait for 3*PER;
   end process run;

   -----------------------------------
   -- End test processes
   -----------------------------------


end architecture;
```

2 – Write behavioral VHDL code for a special new FlipFlop you have conceived of – this FlipFlop has a second data input called Toggle(T). When T is low, the DFF operates normally, when T is high, the DFF ignores the D input and synchronously toggles the output. Provide code and a simulation 50pts

Simulation Results



data mode toggle mode data mode