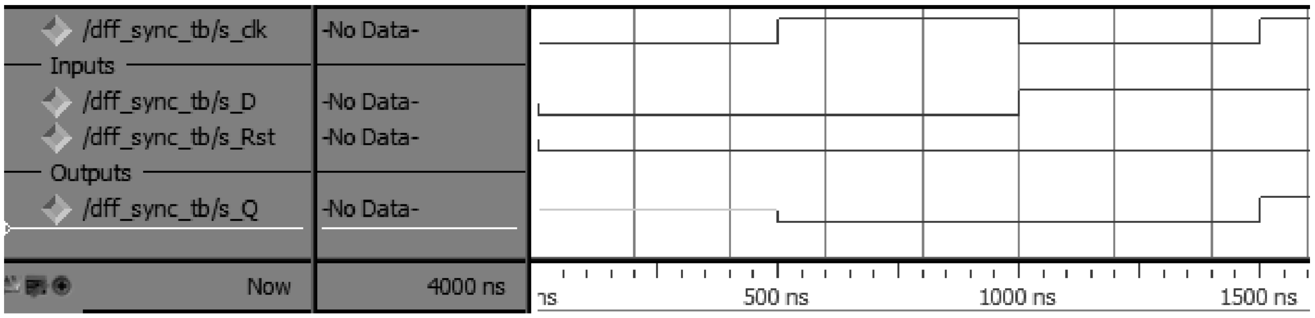
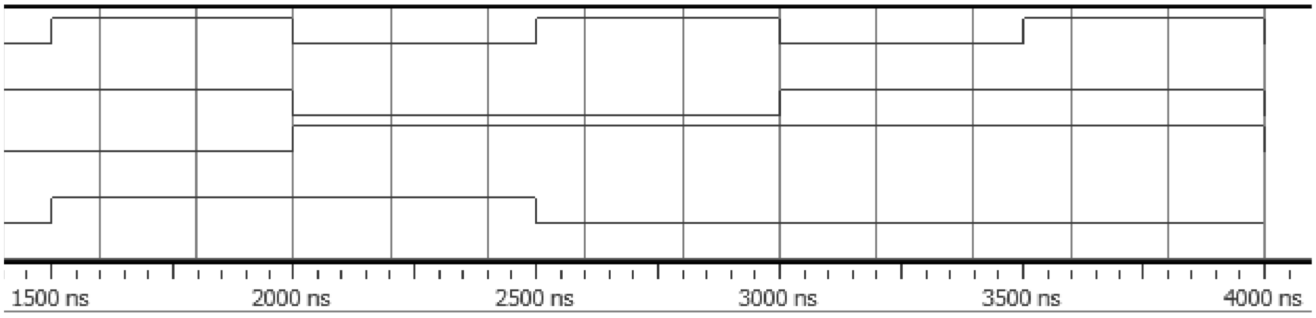
```
entity DFF_sync is
   port(i_D        : in std_logic;
            i_Clk : in std_logic;
            i_Rst : in std_logic;

            o_Q         : out std_logic);
end DFF_sync;

architecture behavior of DFF_sync is
begin

      process(i_Clk)
      begin

            if(rising_edge(i_Clk)) then
                  if(i_Rst='1') then
                        o_Q <= '0';
                  else
                        o_Q <= i_D;
                  end if;
            end if;

      end process;
end behavior;
```

1 MHz clock → 500 ns per ½ period





```vhdl
entity DFF_sync_tb is
end DFF_sync_tb;

architecture behavior of DFF_sync_tb is
      signal s_clk      : std_logic := '0';

      signal s_D        : std_logic;
      signal s_Rst      : std_logic;
      signal s_Q        : std_logic;
begin

DUT : entity work.DFF_sync
  port map (i_D   => s_D,
                  i_Clk => s_clk,
                  i_Rst => s_Rst,

                  o_Q     => s_Q);

      s_clk <= not s_clk after 500 ns;



process is
begin

      s_D   <= '0';
      s_Rst <= '0';
    wait for 500 ns;

      s_D   <= '0';
      s_Rst <= '0';
    wait for 500 ns;

      s_D   <= '1';
      s_Rst <= '0';
    wait for 500 ns;

      s_D   <= '1';
      s_Rst <= '0';
    wait for 500 ns;
```

```
        s_D   <= '0';
        s_Rst <= '1';
      wait for 500 ns;

        s_D   <= '0';
        s_Rst <= '1';
      wait for 500 ns;

        s_D   <= '1';
        s_Rst <= '1';
      wait for 500 ns;

        s_D   <= '1';
        s_Rst <= '1';
      wait for 500 ns;

  end process;
  end behavior;
```