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

entity d_ff is
      Port(d_in: in std_logic; clk: in std_logic; rst: in std_logic;
d_out: out std_logic);
end d_ff;

architecture Behaviorial of d_ff is
begin
      p1: process(d_in,clk,rst)
      begin
            if (rst='1') then d_out <='0';
            elsif (rising_edge(clk)) then d_out <=d_in;
            end if;
      end process p1;
end Behaviorial;



library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
use IEEE.std_logic_unsigned.all;

entity ms_dff is
      Port(d_in:in std_logic; clk: in std_logic; rst: in std_logic; q:
std_logic);
end ms_dff;

architecture Structural of ms_dff is
Component d_ff is
      Port(d_in:in std_logic; clk: in std_logic; rst: in std_logic; d_out:
std_logic);
end Component;
Signal dff1_out: std_logic;
Signal not_clk: std_logic;
begin
      not_clk<= not clk;
      dff_1: d_ff port map(d_in,clk,rst,dff1_out);
      dff_2: d_ff port map(dff1_out,not_clk,rst,q);
end Structural;
```