```vhdl
-------------------------------------------------------------------------------
--
--  Purpose:
--
--    D FlipFlop
--
--  Discussion:
--
--
--  Licensing:
--
--    This code is distributed under the GNU LGPL license.
--
--  Modified:
--
--    2012.04.02
--
--  Author:
--
--    Young W. Lim
--
--  Parameters:
--
--    Input:
--
--    Output:
--------------------------------------------------------------------------------

library STD;
use STD.textio.all;

library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.numeric_std.all;





entity dff is
  generic (
    WD     : in natural := 32);

  port (
    clk    : in   std_logic := '0';
    rst    : in   std_logic := '0';
    di     : in   std_logic_vector (WD-1 downto 0) := (others=>'0');
    dq     : out  std_logic_vector (WD-1 downto 0) := (others=>'0') );

end dff;



architecture rtl of dff is
begin

  Reg: process (clk, rst)
  begin  -- process Reg
    if rst = '0' then                   -- asynchronous reset (active low)
      dq <= (others=>'0');
    elsif clk'event and clk = '1' then  -- rising clock edge
      dq <= di;
    end if;
  end process Reg;

end rtl;



-- entity dff1 is
--   generic (
--     WD     : in natural := 32);
--
--   port (
--     clk    : in   std_logic := '0';
--     rst    : in   std_logic := '0';
--     di     : in   std_logic;
--     dq     : out  std_logic);

-- end dff1;



-- architecture rtl of dff1 is
-- begin

--   Reg: process (clk, rst)
--   begin  -- process Reg
--     if rst = '0' then                   -- asynchronous reset (active low)
--       dq <= (others=>'0');
--     elsif clk'event and clk = '1' then  -- rising clock edge
--       dq <= di;
--     end if;
--   end process Reg;

-- end rtl;
```