```vhdl
-------------------------------------------------------------------------------
--
--  Purpose:
--
--    Synchronous Counter
--
--  Discussion:
--
--
--  Licensing:
--
--    This code is distributed under the GNU LGPL license.
--
--  Modified:
--
--    2012.03.30
--
--  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 counter is
  generic (
    WD     : in natural := 32;
    SH     : in natural :=  5 );

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

end counter;


architecture rtl of counter is

  component adder
    generic (
      WD     : in natural := 32 );

    port (
      an    : in   std_logic_vector (SH-1 downto 0);
      bn    : in   std_logic_vector (SH-1 downto 0);
      ci    : in   std_logic;
      cn    : out  std_logic_vector (SH-1 downto 0);
      co    : out  std_logic );
  end component;

  signal cnt    : std_logic_vector (SH-1 downto 0);
  signal cntInc : std_logic_vector (SH-1 downto 0);
  signal tbd : std_logic;

begin
```

```
  inc : adder port map (an=>cnt, bn=>(others=>'0'), ci=>'1', cn=>cntInc, co=>tbd);

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

  dq <= cnt;

end rtl;
```