```
--------------------------------------------------------------------------------
--
--  Purpose:
--
--    Monitors signals and writes their values
--
--  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;

use WORK.cordic_pkg.all;



entity disp is
  generic (
    WD     : in natural := 32;
    SH     : in natural :=  5;
    PWR    : in natural := 64);

  port (
    load  : in   std_logic := '0';
    ready : in   std_logic := '0';
    cnt   : in   std_logic_vector (SH-1 downto 0) := (others=>'0');
    xn    : in   std_logic_vector (WD-1 downto 0) := (others=>'0');
    yn    : in   std_logic_vector (WD-1 downto 0) := (others=>'0');
    zn    : in   std_logic_vector (WD-1 downto 0) := (others=>'0');
    angle : in   std_logic_vector (WD-1 downto 0) := (others=>'0') );

end disp;

architecture beh of disp is

  signal dinInc : std_logic_vector (SH-1 downto 0);

begin

  monitor: process
  begin  -- process Reg
    wait for 1 ns;

    if (load='1') then
      DispReg(xn, yn, zn, 0);
      DispAng(angle);
    elsif (ready='1') then
```

```
      DispReg(xn, yn, zn, 2);
      DispAng(angle);
    else
      DispReg(xn, yn, zn, 1);
      DispAng(angle);
    end if;

    wait on cnt;
  end process monitor;

end beh;
```