

```

-----
--
-- Purpose:
--
--   utility package of cordic
--
-- Discussion:
--
--
-- Licensing:
--
--   This code is distributed under the GNU LGPL license.
--
-- Modified:
--
--   2012.03.16
--
-- Author:
--
--   Young W. Lim
--
-- Functions:
-- Conv2fixedPt (x : real; n : integer) return std_logic_vector;
-- Conv2real (s : std_logic_vector (31 downto 0) ) return real;
--
-----

```

```

library STD;
use STD.textio.all;

```

```

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

```

```

package cordic_pkg is

```

```

  function Conv2fixedPt (x : real; n : integer) return std_logic_vector;
  function Conv2real (s : std_logic_vector (31 downto 0) ) return real;

```

```

  procedure DispReg (x, y, z : in std_logic_vector (31 downto 0);
                    flag : in integer );

```

```

  constant clk_period : time := 20 ns;
  constant half_period : time := clk_period / 2.0;

```

```

  constant pi : real := 3.141592653589793;
  constant K : real := 1.646760258121;

```

```

end cordic_pkg;

```

```

package body cordic_pkg is

```

```

-----
function Conv2fixedPt (x : real; n : integer) return std_logic_vector is
-----

```

```

  constant shft : std_logic_vector (n-1 downto 0) := X"2000_0000";
  variable s : std_logic_vector (n-1 downto 0) ;
  variable z : real := 0.0;
-----

```

```

begin

```

```

  -- shft = 2^29 = 536870912
  -- bit 31 : msb - sign bit
  -- bit 30,29 : integer part
  -- bit 28 ~ 0 : fractional part
  -- for the value of 0.5
  -- first 4 msb bits [0, 0, 0, 1] --> X"1000_0000"

```

```

--
-- To obtain binary number representation of x,
-- where the implicit decimal point between bit 29 and bit 28,
-- multiply "integer converted shft"
--
z := x * real(to_integer(unsigned(shft)));

s := std_logic_vector(to_signed(integer(z), n));

return s;

end Conv2fixedPt;
-----

function Conv2real (s : std_logic_vector (31 downto 0) ) return real is
-----
constant shft : std_logic_vector (31 downto 0) := X"2000_0000";
variable z : real := 0.0;
-----
begin
z := real(to_integer(signed(s))) / real(to_integer(unsigned(shft)));
return z;
end Conv2real;
-----

procedure DispReg (x, y, z : in std_logic_vector (31 downto 0);
flag : in integer ) is
-----
variable l : line;
begin
if (flag = 0) then
write(l, String'("----- "));
writeline(output, l);
end if;
write(l, String'(" x = "));
write(l, real(to_integer(signed(x))));
write(l, String'(" y = "));
write(l, real(to_integer(signed(y))));
write(l, String'(" z = "));
write(l, real(to_integer(signed(z))));
writeline(output, l);
end DispReg;
-----

end cordic_pkg;

```