library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity uni\_shift1 is

port(reset, si, clk, load : in STD\_LOGIC;

pi :in STD\_LOGIC\_VECTOR ( 3 downto 0);

po: out STD\_LOGIC\_VECTOR ( 3 downto 0);

mode : in STD\_LOGIC\_VECTOR ( 1 downto 0);

so : out STD\_LOGIC

);

end uni\_shift1;

architecture Behavioral of uni\_shift1 is

signal temp: std\_logic\_vector(3 downto 0):="0000";

signal clk\_s: std\_logic\_vector(23 downto 0):=(others=>'0');

begin

--Count&lt;= **clk\_s**(22);

--Process(clk)

--Begin

--If rising\_edge (clk) then

--Clk\_s<=clk\_s+ '1';

--End if;[[1]](#footnote-1)

--End process;--

Process(**clk**, mode, si, load, reset)

Begin

If reset='1' then temp<="0000"; po<="0000"; so<='0';

Elsif rising\_edge(clk) then

Case mode is

--SISO

When "00"=> temp(3 downto 1) <= temp(2 downto 0);

Temp(0)<=si;

So<=temp(3);

--SIPO

When "01"=> temp(0)<=si;

temp(3 downto 1) <= temp(2 downto 0);

po<=temp;

--PIPO

When "10"=> if load='1' then

temp<=pi;

else

po<= temp;

end if;

--PISO

When "11"=> if(load='1') then

Temp(3 downto 0)<=pi(3 downto 0);

Else

So<=temp(3);

temp(3 downto 1)<= temp(2 downto 0);

temp(0)<='0';

end if;

when others=> null;

end case;

end if;

end process;

end behavioral;

1. Please see the coloured code part carefully.it is about clock clock reduction [↑](#footnote-ref-1)