**Sipo Main Code**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity sipo is

Port ( clk : in STD\_LOGIC;

slow\_clk: inOUT STD\_LOGIC;

in2 : in STD\_LOGIC;

out2 : inout STD\_LOGIC\_VECTOR (3 downto 0));

end sipo;

architecture Behavioral of sipo is

signal counter: std\_logic\_vector(1 downto 0):=(others =>'0');

begin

process(slow\_clk,in2,out2)

begin

if(slow\_clk'EVENT and slow\_clk='1') THEN

out2(3 DOWNTO 1) <= out2(2 DOWNTO 0);

out2(0) <= in2;

end if;

end process;

process(clk)

begin

if (clk'event and clk='1')then

counter <=counter+'1';

end if;

end process;

slow\_clk <= counter(1);

end Behavioral;

**Sipo Tb**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity sipo\_tb is

-- Port ( );

end sipo\_tb;

architecture Behavioral of sipo\_tb is

component sipo is

Port ( clk : in STD\_LOGIC;

slow\_clk: inout STD\_LOGIC;

in2 : in STD\_LOGIC;

out2 : inout STD\_LOGIC\_VECTOR (3 downto 0));

end component;

signal clk, in2,slow\_clk: std\_logic;

signal out2 : std\_logic\_vector( 3 downto 0);

begin

u1: sipo port map ( clk,slow\_clk,in2,out2);

process

begin

clk<= '0';

wait for 10 ns;

clk<= '1';

wait for 10 ns ;

end process;

in2<= '1', '0' after 100 ns;

end Behavioral;

**OUTPUT:**

