**EENG 2910 Project III – Digital System Design**

Design Project 4

Nathan A Ruprecht

Due 19 October 2016

# Introduction:

For our fourth project, we were required to design, implement, and test a 4-bit universal shift register.

A universal shift register is a sequential logic circuit that can be used as either serial-to-serial (SISO), serial-to-parallel (SIPO), parallel-to-parallel (PIPO), parallel-to-serial (PISO), left shifting, or right shifting register. We were shown the below picture as a visual aid.

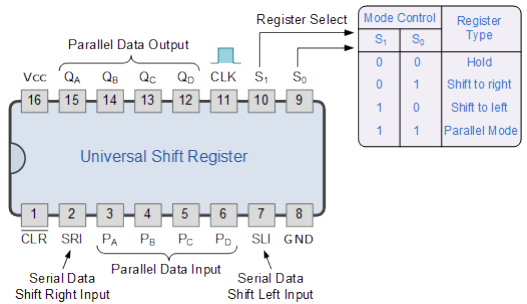


Figure : Shift Register Visual

# Theory of Operation and Explanation of the Design:

I originally started by following past code, specifically the ALU, where there were multiple functions (in this case the 6 types of shifting) and a case statement to select which one is shown through to the output. For each type of shifting, our lecture notes showed using D flip flops with the output usually feeding into the other. Since we were given the code for SIPO, I was able to take that and tailor the inputs and outputs to make SISO, PIPO, and PISO.

I ran into problems for PISO and left / right shifting. In the lecture notes, the figure shows a mux at the input to each flip flop for PISO. The idea is that the parallel input is put into the flip flops so each dff holds the value for one of the indexes of the input. Then the output of each dff is fed into the input of the next dff so it’s now a serial out. The way it was explained to me to actually implement is, is to send the parallel in, then shift the values so the original input is outputted one bit at a time.

# Experimental Results:

Since I recycled code structure from the ALU assignment, I took the long way to make the shift register. This mainly pertains to the left / right shift since I first tried to do them as separate components. Instead, the best way was to manipulate the variables in the main function.

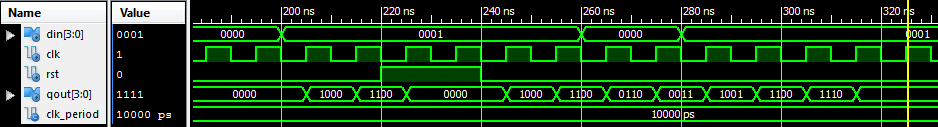


Figure : SIPO

The SIPO code was given to us. Figure 4 is a screenshot of the simulation of the code given to show that it works just as promised.

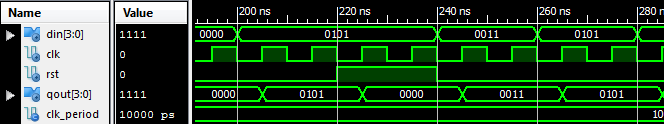


Figure : PIPO

PIPO and SISO were the easiest since the output is the input. For PIPO, the output of the dff isn’t fed into the next dff, but is fed into the actual output. Figure 3 shows that the output reflects the input on the next rising edge of the clock.

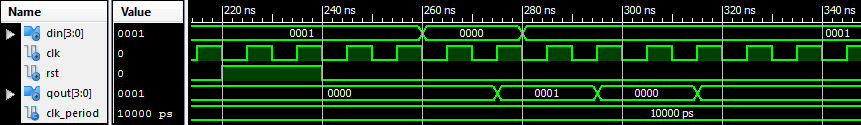


Figure : SISO

With SISO, there’s a delay for seeing the input show up on the output because it’s going through the 4 flip flops before assigned as the output.

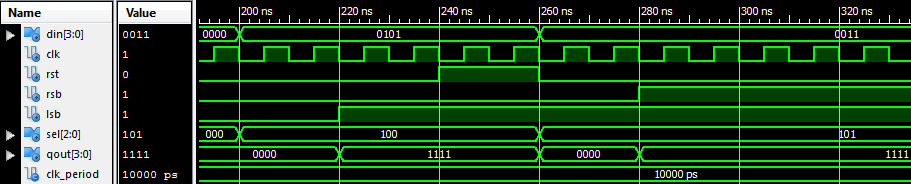


Figure : Right / Left Shift

For the left and right shift, the temporary output signal is shifted with the left shift bit (LSB) or right shift bit (RSB) before assigned to the actual output.

# Conclusion:

The goal of the lab was to design, implement, and test a 4-bit universal shift register. In the process, I was able to recycle code structure I used from the ALU, and take the given code to tailor it to the other functions. I used component style coding for SIPO, PIPO, and SISO, used signals to show the left shift, right shift, and a combination for PISO (PIPO to save data, then right shift).

# References:

Figure 1 from class lecture or found at:

"Shift Register - Parallel and Serial Shift Register." *Basic Electronics Tutorials The Shift Register*

*Comments*. N.p., 09 Sept. 2016. Web. 19 Oct. 2016.

Figure 2-5 are screenshots from iSim running the VHDL tb in source code

# Source Code:

----------------------------------------------------------------------------------

-- Company:

-- Engineer: Nathan A Ruprecht

--

-- Create Date: 15:35:08 10/05/2016

-- Design Name:

-- Module Name: ShiftRegister\_4bit - Behavioral

-- Project Name:

-- Target Devices:

-- Tool versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--use IEEE.NUMERIC\_STD.ALL;

-- Uncomment the following library declaration if instantiating

-- any Xilinx primitives in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity ShiftRegister\_4bit is

port (

Din : in std\_logic\_vector(3 downto 0);

CLK : in std\_logic;

RST : in std\_logic;

RSB : in std\_logic;

LSB : in std\_logic;

Sel : in std\_logic\_vector(2 downto 0);

Qout : out std\_logic\_vector(3 downto 0)

);

end ShiftRegister\_4bit;

architecture Behavioral of ShiftRegister\_4bit is

----------------------------------------------------------------------------------

component SIPO is

port (

Din : in std\_logic\_vector(3 downto 0);

CLK : in std\_logic;

RST : in std\_logic;

Qout : out std\_logic\_vector(3 downto 0)

);

end component;

----------------------------------------------------------------------------------

component PIPO is

port (

Din : in std\_logic\_vector(3 downto 0);

CLK : in std\_logic;

RST : in std\_logic;

Qout : out std\_logic\_vector(3 downto 0)

);

end component;

----------------------------------------------------------------------------------

component SISO is

port (

Din : in std\_logic\_vector(3 downto 0);

CLK : in std\_logic;

RST : in std\_logic;

Qout : out std\_logic\_vector(3 downto 0)

);

end component;

----------------------------------------------------------------------------------

component PISO is

port (

Din : in std\_logic\_vector(3 downto 0);

CLK : in std\_logic;

RST : in std\_logic;

Qout : out std\_logic\_vector(3 downto 0)

);

end component;

----------------------------------------------------------------------------------

signal q0, q1, q2, q3 : std\_logic;

signal s\_Y0, s\_Y1, s\_Y2, q\_YQ : std\_logic\_vector(3 downto 0);

signal s\_sig : std\_logic\_vector(2 downto 0);

begin

s\_sig <= Sel;

shift\_1: SIPO --code given and works with tb

port map (

Din => Din,

CLK => CLK,

RST => RST,

Qout => s\_Y0

);

shift\_2: PIPO --works with tb

port map (

Din => Din,

CLK => CLK,

RST => RST,

Qout => s\_Y1

);

shift\_3: SISO -- works with tb

port map (

Din => Din,

CLK => CLK,

RST => RST,

Qout => s\_Y2

);

process(s\_sig, Din, CLK, RST, LSB,RSB,q\_YQ)

begin

case s\_sig is

when "000" => q\_YQ <= s\_Y0; --SIPO

when "001" => q\_YQ <= s\_Y1; --PIPO

when "010" => q\_YQ <= s\_Y2; --SISO

when "011" => q\_YQ <= s\_Y1; --PISO

when "100" => q\_YQ <= q\_YQ(2 downto 0) & LSB; --LeftShift

when "101" => q\_YQ <= RSB & q\_YQ(3 downto 1); --RightShift

when others => q\_YQ <= (others => '0');

end case;

end process;

Qout<=q\_YQ;

end Behavioral;

--------------------------------------------------------------------------------

-- Company:

-- Engineer: Nathan A Ruprecht

--

-- Create Date: 11:55:53 10/19/2016

-- Design Name:

-- Module Name: E:/2910/DP4/Code/Shift\_Register/tb\_shiftregister.vhd

-- Project Name: Shift\_Register

-- Target Device:

-- Tool versions:

-- Description:

--

-- VHDL Test Bench Created by ISE for module: ShiftRegister\_4bit

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

-- Notes:

-- This testbench has been automatically generated using types std\_logic and

-- std\_logic\_vector for the ports of the unit under test. Xilinx recommends

-- that these types always be used for the top-level I/O of a design in order

-- to guarantee that the testbench will bind correctly to the post-implementation

-- simulation model.

--------------------------------------------------------------------------------

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--USE ieee.numeric\_std.ALL;

ENTITY tb\_shiftregister IS

END tb\_shiftregister;

ARCHITECTURE behavior OF tb\_shiftregister IS

-- Component Declaration for the Unit Under Test (UUT)

COMPONENT ShiftRegister\_4bit

PORT(

Din : IN std\_logic\_vector(3 downto 0);

CLK : IN std\_logic;

RST : IN std\_logic;

RSB : IN std\_logic;

LSB : IN std\_logic;

Sel : IN std\_logic\_vector(2 downto 0);

Qout : OUT std\_logic\_vector(3 downto 0)

);

END COMPONENT;

--Inputs

signal Din : std\_logic\_vector(3 downto 0) := (others => '0');

signal CLK : std\_logic := '0';

signal RST : std\_logic := '0';

signal RSB : std\_logic := '0';

signal LSB : std\_logic := '0';

signal Sel : std\_logic\_vector(2 downto 0) := (others => '0');

--Outputs

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

-- Clock period definitions

constant CLK\_period : time := 10 ns;

BEGIN

-- Instantiate the Unit Under Test (UUT)

uut: ShiftRegister\_4bit PORT MAP (

Din => Din,

CLK => CLK,

RST => RST,

RSB => RSB,

LSB => LSB,

Sel => Sel,

Qout => Qout

);

-- Clock process definitions

CLK\_process :process

begin

CLK <= '0';

wait for CLK\_period/2;

CLK <= '1';

wait for CLK\_period/2;

end process;

-- Stimulus process

stim\_proc: process

begin

-- hold reset state for 100 ns.

wait for 100 ns;

wait for CLK\_period\*10;

-- insert stimulus here

Din <= "0101";

Sel <= "100";

LSB <= '0';

wait for 2 \* clk\_period;

LSB <= '1';

wait for 2 \* clk\_period;

RST <= '1';

wait for 2 \* clk\_period;

RST <= '0';

Din <= "0011";

Sel <= "101";

RSB <= '0';

wait for 2 \* clk\_period;

RSB <= '1';

wait for 2 \* clk\_period;

RST <= '0';

Din <= "0011";

Sel <= "000";

wait for 2 \* clk\_period;

Din <= "0011";

Sel <= "001";

wait for 2 \* clk\_period;

Din <= "0011";

Sel <= "010";

wait for 2 \* clk\_period;

Din <= "0011";

Sel <= "011";

wait for 2 \* clk\_period;

wait;

end process;

END;

----------------------------------------------------------------------------------

-- Company:

-- Engineer: Nathan A Ruprecht

--

-- Create Date: 19:30:23 10/05/2016

-- Design Name:

-- Module Name: SIPO - Behavioral

-- Project Name:

-- Target Devices:

-- Tool versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--use IEEE.NUMERIC\_STD.ALL;

-- Uncomment the following library declaration if instantiating

-- any Xilinx primitives in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity SIPO is

port (

Din : in std\_logic\_vector(3 downto 0);

CLK : in std\_logic;

RST : in std\_logic;

Qout : out std\_logic\_vector(3 downto 0)

);

end SIPO;

architecture Behavioral of SIPO is

----------------------------------------------------------------------------------

component dff is

port(

D : in std\_logic;

CLK : in std\_logic;

RST : in std\_logic;

Q : out std\_logic

);

end component;

----------------------------------------------------------------------------------

signal q0, q1, q2, q3 : std\_logic;

begin

ff1: dff

port map (

D => Din(0),

CLK => CLK,

RST => RST,

Q => q0

);

ff2: dff

port map (

D => q0,

CLK => CLK,

RST => RST,

Q => q1

);

ff3: dff

port map (

D => q1,

CLK => CLK,

RST => RST,

Q => q2

);

ff4: dff

port map (

D => q2,

CLK => CLK,

RST => RST,

Q => q3

);

Qout <= q0&q1&q2&q3;

end Behavioral;

--------------------------------------------------------------------------------

-- Company:

-- Engineer: Nathan A Ruprecht

--

-- Create Date: 15:22:04 10/14/2016

-- Design Name:

-- Module Name: E:/2910/DP4/Code/Shift\_Register/tb\_SIPO.vhd

-- Project Name: Shift\_Register

-- Target Device:

-- Tool versions:

-- Description:

--

-- VHDL Test Bench Created by ISE for module: SIPO

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

-- Notes:

-- This testbench has been automatically generated using types std\_logic and

-- std\_logic\_vector for the ports of the unit under test. Xilinx recommends

-- that these types always be used for the top-level I/O of a design in order

-- to guarantee that the testbench will bind correctly to the post-implementation

-- simulation model.

--------------------------------------------------------------------------------

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--USE ieee.numeric\_std.ALL;

ENTITY tb\_SIPO IS

END tb\_SIPO;

ARCHITECTURE behavior OF tb\_SIPO IS

-- Component Declaration for the Unit Under Test (UUT)

COMPONENT SIPO

PORT(

Din : IN std\_logic\_vector(3 downto 0);

CLK : IN std\_logic;

RST : IN std\_logic;

Qout : OUT std\_logic\_vector(3 downto 0)

);

END COMPONENT;

--Inputs

signal Din : std\_logic\_vector(3 downto 0) := (others => '0');

signal CLK : std\_logic := '0';

signal RST : std\_logic := '0';

--Outputs

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

-- Clock period definitions

constant CLK\_period : time := 10 ns;

BEGIN

-- Instantiate the Unit Under Test (UUT)

uut: SIPO PORT MAP (

Din => Din,

CLK => CLK,

RST => RST,

Qout => Qout

);

-- Clock process definitions

CLK\_process :process

begin

CLK <= '0';

wait for CLK\_period/2;

CLK <= '1';

wait for CLK\_period/2;

end process;

-- Stimulus process

stim\_proc: process

begin

-- hold reset state for 100 ns.

wait for 100 ns;

wait for CLK\_period\*10;

-- insert stimulus here

Din <= "0001";

wait for 2 \* clk\_period;

RST <= '1';

wait for 2 \* clk\_period;

RST <= '0';

Din <= "0001";

wait for 2 \* clk\_period;

Din <= "0000";

wait for 2 \* clk\_period;

Din <= "0001";

wait for 2 \* clk\_period;

wait;

end process;

END;

----------------------------------------------------------------------------------

-- Company:

-- Engineer: Nathan A Ruprecht

--

-- Create Date: 20:05:30 10/05/2016

-- Design Name:

-- Module Name: SISO - Behavioral

-- Project Name:

-- Target Devices:

-- Tool versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--use IEEE.NUMERIC\_STD.ALL;

-- Uncomment the following library declaration if instantiating

-- any Xilinx primitives in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity SISO is

port (

Din : in std\_logic\_vector(3 downto 0);

CLK : in std\_logic;

RST : in std\_logic;

Qout : out std\_logic\_vector(3 downto 0)

);

end SISO;

architecture Behavioral of SISO is

----------------------------------------------------------------------------------

component dff is

port(

D : in std\_logic;

CLK : in std\_logic;

RST : in std\_logic;

Q : out std\_logic

);

end component;

----------------------------------------------------------------------------------

signal q0, q1, q2, q3 : std\_logic;

begin

ff1: dff

port map (

D => Din(0),

CLK => CLK,

RST => RST,

Q => q0

);

ff2: dff

port map (

D => q0,

CLK => CLK,

RST => RST,

Q => q1

);

ff3: dff

port map (

D => q1,

CLK => CLK,

RST => RST,

Q => q2

);

ff4: dff

port map (

D => q2,

CLK => CLK,

RST => RST,

Q => q3

);

Qout <= "000" & q3;

end Behavioral;

--------------------------------------------------------------------------------

-- Company:

-- Engineer: Nathan A Ruprecht

--

-- Create Date: 09:22:56 10/18/2016

-- Design Name:

-- Module Name: E:/2910/DP4/Code/Shift\_Register/tb\_SISO.vhd

-- Project Name: Shift\_Register

-- Target Device:

-- Tool versions:

-- Description:

--

-- VHDL Test Bench Created by ISE for module: SISO

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

-- Notes:

-- This testbench has been automatically generated using types std\_logic and

-- std\_logic\_vector for the ports of the unit under test. Xilinx recommends

-- that these types always be used for the top-level I/O of a design in order

-- to guarantee that the testbench will bind correctly to the post-implementation

-- simulation model.

--------------------------------------------------------------------------------

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--USE ieee.numeric\_std.ALL;

ENTITY tb\_SISO IS

END tb\_SISO;

ARCHITECTURE behavior OF tb\_SISO IS

-- Component Declaration for the Unit Under Test (UUT)

COMPONENT SISO

PORT(

Din : IN std\_logic\_vector(3 downto 0);

CLK : IN std\_logic;

RST : IN std\_logic;

Qout : OUT std\_logic\_vector(3 downto 0)

);

END COMPONENT;

--Inputs

signal Din : std\_logic\_vector(3 downto 0) := (others => '0');

signal CLK : std\_logic := '0';

signal RST : std\_logic := '0';

--Outputs

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

-- Clock period definitions

constant CLK\_period : time := 10 ns;

BEGIN

-- Instantiate the Unit Under Test (UUT)

uut: SISO PORT MAP (

Din => Din,

CLK => CLK,

RST => RST,

Qout => Qout

);

-- Clock process definitions

CLK\_process :process

begin

CLK <= '0';

wait for CLK\_period/2;

CLK <= '1';

wait for CLK\_period/2;

end process;

-- Stimulus process

stim\_proc: process

begin

-- hold reset state for 100 ns.

wait for 100 ns;

wait for CLK\_period\*10;

-- insert stimulus here

Din <= "0001";

wait for 2 \* clk\_period;

RST <= '1';

wait for 2 \* clk\_period;

RST <= '0';

Din <= "0001";

wait for 2 \* clk\_period;

Din <= "0000";

wait for 2 \* clk\_period;

Din <= "0001";

wait for 2 \* clk\_period;

wait;

end process;

END;

----------------------------------------------------------------------------------

-- Company:

-- Engineer: Nathan A Ruprecht

--

-- Create Date: 20:05:54 10/05/2016

-- Design Name:

-- Module Name: PIPO - Behavioral

-- Project Name:

-- Target Devices:

-- Tool versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--use IEEE.NUMERIC\_STD.ALL;

-- Uncomment the following library declaration if instantiating

-- any Xilinx primitives in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity PIPO is

port (

Din : in std\_logic\_vector(3 downto 0);

CLK : in std\_logic;

RST : in std\_logic;

Qout : out std\_logic\_vector(3 downto 0)

);

end PIPO;

architecture Behavioral of PIPO is

----------------------------------------------------------------------------------

component dff is

port(

D : in std\_logic;

CLK : in std\_logic;

RST : in std\_logic;

Q : out std\_logic

);

end component;

----------------------------------------------------------------------------------

signal q0, q1, q2, q3 : std\_logic;

begin

ff1: dff

port map (

D => Din(0),

CLK => CLK,

RST => RST,

Q => q0

);

ff2: dff

port map (

D => Din(1),

CLK => CLK,

RST => RST,

Q => q1

);

ff3: dff

port map (

D => Din(2),

CLK => CLK,

RST => RST,

Q => q2

);

ff4: dff

port map (

D => Din(3),

CLK => CLK,

RST => RST,

Q => q3

);

Qout <= q3&q2&q1&q0;

end Behavioral;

--------------------------------------------------------------------------------

-- Company:

-- Engineer: Nathan A Ruprecht

--

-- Create Date: 15:25:52 10/14/2016

-- Design Name:

-- Module Name: E:/2910/DP4/Code/Shift\_Register/tb\_PIPO.vhd

-- Project Name: Shift\_Register

-- Target Device:

-- Tool versions:

-- Description:

--

-- VHDL Test Bench Created by ISE for module: PIPO

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

-- Notes:

-- This testbench has been automatically generated using types std\_logic and

-- std\_logic\_vector for the ports of the unit under test. Xilinx recommends

-- that these types always be used for the top-level I/O of a design in order

-- to guarantee that the testbench will bind correctly to the post-implementation

-- simulation model.

--------------------------------------------------------------------------------

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--USE ieee.numeric\_std.ALL;

ENTITY tb\_PIPO IS

END tb\_PIPO;

ARCHITECTURE behavior OF tb\_PIPO IS

-- Component Declaration for the Unit Under Test (UUT)

COMPONENT PIPO

PORT(

Din : IN std\_logic\_vector(3 downto 0);

CLK : IN std\_logic;

RST : IN std\_logic;

Qout : OUT std\_logic\_vector(3 downto 0)

);

END COMPONENT;

--Inputs

signal Din : std\_logic\_vector(3 downto 0) := (others => '0');

signal CLK : std\_logic := '0';

signal RST : std\_logic := '0';

--Outputs

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

-- Clock period definitions

constant CLK\_period : time := 10 ns;

BEGIN

-- Instantiate the Unit Under Test (UUT)

uut: PIPO PORT MAP (

Din => Din,

CLK => CLK,

RST => RST,

Qout => Qout

);

-- Clock process definitions

CLK\_process :process

begin

CLK <= '0';

wait for CLK\_period/2;

CLK <= '1';

wait for CLK\_period/2;

end process;

-- Stimulus process

stim\_proc: process

begin

-- hold reset state for 100 ns.

wait for 100 ns;

wait for CLK\_period\*10;

-- insert stimulus here

Din <= "0101";

wait for 2 \* clk\_period;

RST <= '1';

wait for 2 \* clk\_period;

RST <= '0';

Din <= "0011";

wait for 2 \* clk\_period;

Din <= "0101";

wait for 2 \* clk\_period;

Din <= "1111";

wait for 2 \* clk\_period;

wait;

end process;

END;