-------------------------------------- Code --------------------------------------------

-- Name:

-- Experiment No. 1: To Study VLSI Design flow using an EDA Tool.

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

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity be\_a4\_and\_vhdl is

Port ( A : in STD\_LOGIC;

B : in STD\_LOGIC;

Y : out STD\_LOGIC);

end be\_a4\_and\_vhdl;

architecture Behavioral of be\_a4\_and\_vhdl is

begin

Y<= A AND B;

end Behavioral;

-------------------------------------- Test Bench --------------------------------------------

-- Name:

-- Experiment No. 1: To Study VLSI Design flow using an EDA Tool.

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

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

ENTITY BE\_A4\_AND\_A4 IS

END BE\_A4\_AND\_A4;

ARCHITECTURE behavior OF BE\_A4\_AND\_A4 IS

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

COMPONENT be\_a4\_and\_vhdl

PORT( A : IN std\_logic;

B : IN std\_logic;

Y : OUT std\_logic);

END COMPONENT;

--Inputs

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

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

--Outputs

signal Y : std\_logic;

BEGIN

-- Instantiate the Unit Under Test (UUT)

uut: be\_a4\_and\_vhdl PORT MAP (

A => A,

B => B,

Y => Y

);

-- Stimulus process

stim\_proc: process

begin

A<='0';

B<='0';

-- hold reset state for 100 ns.

wait for 100 ns;

A<='0';

B<='1';

wait for 100 ns;

A<='1';

B<='0';

wait for 100 ns;

A<='1';

B<='1';

wait;

end process;

END;

-------------------------------------- Code --------------------------------------------

-- Name:

-- Roll No:

-- Experiment No. 2: To write a VHDL code for ALU.

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

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity Be\_A4\_ALU is

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

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

S : in STD\_LOGIC\_VECTOR (2 downto 0);

Y : out STD\_LOGIC\_VECTOR (4 downto 0));

end Be\_A4\_ALU;

architecture Behavioral of Be\_A4\_ALU is

begin

Y <= ('0' & A) + ('0' & B) when S = "000" else

('0' & A) - ('0' & B) when S = "001" else

'-' & (A AND B) when S = "010" else

'-' & (A NAND B) when S = "011" else

'-' & (A XOR B) when S = "100" else

'-' & (A XNOR B) when S = "101" else

'-' & (A OR B) when S = "110" else

('-' & A);

end Behavioral;

-------------------------------------- Test Bench --------------------------------------------

-- Name:

-- Experiment No. 2: To write a VHDL code for ALU.

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

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

ENTITY BE\_A4\_ALU\_Test\_bench IS

END BE\_A4\_ALU\_Test\_bench;

ARCHITECTURE behavior OF BE\_A4\_ALU\_Test\_bench IS

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

COMPONENT Be\_A4\_ALU

PORT(

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

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

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

Y : OUT std\_logic\_vector(4 downto 0)

);

END COMPONENT;

--Inputs

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

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

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

--Outputs

signal Y : std\_logic\_vector(4 downto 0);

-- No clocks detected in port list. Replace <clock> below with

-- appropriate port name

BEGIN

-- Instantiate the Unit Under Test (UUT)

uut: Be\_A4\_ALU PORT MAP (

A => A,

B => B,

S => S,

Y => Y

);

-- Stimulus process

stim\_proc: process

begin

-- hold reset state for 100 ns.

A <= "1011";

B <= "0111";

S <= "000"; --Addition

wait for 100 ns;

S <= "001"; --Subtraction

wait for 100 ns;

S <= "010"; --AND

wait for 100 ns;

S <= "011"; --NAND

wait for 100 ns;

S <= "100"; --XOR

wait for 100 ns;

S <= "101"; --XNOR

wait for 100 ns;

S <= "110"; --OR

wait for 100 ns;

S <= "111"; --ALU Pass

wait;

end process;

END;

-------------------------------------- Code --------------------------------------------

-- Name:

-

-- Experiment No. 3: VHDL code for Universal Shift Register (USR).

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

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity usr is

Port( clk,rst : in std\_logic;

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

m : in std\_logic\_vector(1 downto 0);

y : out std\_logic\_vector(3 downto 0) );

end usr;

architecture Behavioral of usr is

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

begin

process(clk,rst)

begin

if rst = '1' then

y <= "0000";

elsif clk'event and clk = '1' then

case(m) is

when "00" => -- SISO

t(3) <= s(0); -- LSB as input S(0)

t(2) <= t(3);

t(1) <= t(2);

t(0) <= t(1);

y(0) <= t(0); -- LSB as output Y(0)

when "01" => -- SIPO

t(3) <= s(0); -- LSB as input S(0)

t(2) <= t(3);

t(1) <= t(2);

t(0) <= t(1);

y <= t ; -- 4 bit output Y

output Y(0) <= S(0) -- LSB

when "10" => -- PIPO

t <= s; -- 4 bit input S

y <= t; -- 4 bit output Y

when "11" => -- PISO

t <= s; -- 4 bit input S & t <= s

t(2) <= t(3); -- MSB input as t(3) <= S(3)

t(1) <= t(2);

t(0) <= t(1);

y(3) <= t(0); -- MSB as output Y(3) <= S(3)

when others => NULL ;

end case;

end if;

end process;

end Behavioral;

-------------------------------------- Test Bench --------------------------------------------

-- Name:

-- Experiment No. 3: VHDL code for Universal Shift Register (USR).

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

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

ENTITY usr2\_tb IS

END usr2\_tb;

ARCHITECTURE behavior OF usr2\_tb IS

COMPONENT usr

PORT(

clk : IN std\_logic;

rst : IN std\_logic;

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

m : IN std\_logic\_vector(1 downto 0);

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

);

END COMPONENT;

--Inputs

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

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

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

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

--Outputs

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

-- Clock period definitions

constant clk\_period : time := 10 ns;

BEGIN

-- Instantiate the Unit Under Test (UUT)

uut: usr PORT MAP (

clk => clk,

rst => rst,

s => s,

m => m,

y => y

);

-- 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

wait for 10 ns;

rst <= '1';

wait for 10 ns;

rst <= '0';

wait for 20 ns;

m <= "00" ; -- SISO

s <= "UUU1";

wait for 80 ns;

s <= "111U";

wait for 80 ns;

s <= "---0";

wait for 80 ns;

m <= "01" ; -- SIPO

s <= "0U0-";

wait for 80 ns;

s <= "U001";

wait for 80 ns;

s <= "-110";

wait for 80 ns;

m <= "10" ; -- PIPO

s <= "1010";

wait for 70 ns;

s <= "0U01";

wait for 70 ns;

s <= "1-00";

wait for 70 ns;

m <= "11" ; -- PISO

s <= "U000";

wait for 80 ns;

s <= "1010";

wait for 80 ns;

s <= "-111";

wait for 80 ns;

wait;

end process;

END;