**Encoders:**

library IEEE;

use IEEE.STD\_LOGIC\_1164.all;

entity encoder2 is

port(

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

b : out STD\_LOGIC\_VECTOR(1 downto 0)

);

end encoder2;

architecture bhv of encoder2 is

begin

b(0) <= a(1) or a(2);

b(1) <= a(1) or a(3);

end bhv;

**ENCODER\_TB code:**

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

ENTITY tb\_encoder IS

END tb\_encoder;

ARCHITECTURE behavior OF tb\_encoder IS

component encoder2 PORT(

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

b : OUT std\_logic\_vector(1 downto 0)

);

END COMPONENT;

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

signal b : std\_logic\_vector(1 downto 0);

BEGIN

uut: encoder2 PORT MAP (a => a,b => b);

stim\_proc: process

begin

wait for 100 ns;

a <= "0000";

wait for 100 ns;

a <= "0001";

wait for 100 ns;

a <= "0010";

wait for 100 ns;

a <= "0100";

wait for 100 ns;

a <= "1000";

wait;

end process;

END;