Brutus Buckeye brutus.1 mux2\_1x8

**HDL CODE for mux2\_1x8**

LIBRARY ieee;

USE ieee.std\_logic\_1164.all;

ENTITY mux2\_1x8 IS

PORT (a,b : IN std\_logic\_vector(7 downto 0);

sel : IN std\_logic;

r : OUT std\_logic\_vector(7 downto 0));

END mux2\_1x8;

ARCHITECTURE one OF mux2\_1x8 IS

SIGNAL sel\_vec : std\_logic\_vector(7 downto 0);

BEGIN

sel\_vec <= sel & sel & sel & sel & sel & sel & sel & sel;

r <= (a AND NOT sel\_vec) OR (b AND sel\_vec);

END one;

**HDL CODE for testbench of mux2\_1x8**

LIBRARY ieee;

USE ieee.std\_logic\_1164.all;

ENTITY tb\_mux2\_1x8 IS

END tb\_mux2\_1x8;

ARCHITECTURE one OF tb\_mux2\_1x8 IS

--Declare and Configure DUT

COMPONENT mux2\_1x8 IS

PORT (a,b : IN std\_logic\_vector(7 downto 0);

sel : IN std\_logic;

r : OUT std\_logic\_vector(7 downto 0));

END COMPONENT;

FOR all : mux2\_1x8 USE ENTITY work.mux2\_1x8(one);

--Declare signals to hook up DUT

SIGNAL a,b,r : std\_logic\_vector(7 downto 0);

SIGNAL sel : std\_logic;

BEGIN

u0 : mux2\_1x8 PORT MAP (a,b,sel,r);

--apply stimulus

PROCESS

BEGIN

WAIT FOR 10 ns;

a <= "10100101"; b <= "00000000"; sel <= '0';

WAIT FOR 10 ns;

a <= "00000000"; b <= "00000000"; sel <= '0';

WAIT FOR 10 ns;

a <= "00000000"; b <= "10100101"; sel <= '0';

WAIT FOR 10 ns;

a <= "00000000"; b <= "00000000"; sel <= '0';

WAIT FOR 10 ns;

a <= "00000000"; b <= "10100101"; sel <= '0';

WAIT FOR 10 ns;

a <= "00000000"; b <= "00000000"; sel <= '0';

WAIT FOR 20 ns;

a <= "00000000"; b <= "10100101"; sel <= '1';

WAIT FOR 10 ns;

a <= "00000000"; b <= "00000000"; sel <= '1';

WAIT FOR 10 ns;

a <= "00000000"; b <= "10100101"; sel <= '1';

WAIT FOR 10 ns;

a <= "00000000"; b <= "00000000"; sel <= '1';

WAIT FOR 10 ns;

a <= "10100101"; b <= "00000000"; sel <= '1';

WAIT FOR 10 ns;

a <= "00000000"; b <= "00000000"; sel <= '1';

WAIT FOR 50 ns;

WAIT;

END PROCESS;

END one;

**SIMULATION RESULTS**



**QUARTIS SYNTHESIS**

The HDL code was entered into QUARTIS with the following results:

Basic Information

Combinational ALUTs 8

Registers

Pins 25

RTL VIEWER:

