**Report for the mux2-to-1 by 1-bit**

**Brutus Buckeye**

**1.0 The VHDL code and simulation**

The VHDL code for the 2-to-1 mux is as follows:

LIBRARY IEEE;

USE IEEE.std\_logic\_1164.all;

ENTITY mux2\_1 IS

PORT (l,r : IN std\_logic;

sel : IN std\_logic;

muxout : OUT std\_logic);

END mux2\_1;

ARCHITECTURE one OF mux2\_1 IS

BEGIN

muxout <= (sel AND l) OR (NOT sel AND r);

END one;

This code was tested using the following testbench which exhaustively tested it.

ENTITY tm IS

END tm;

LIBRARY IEEE;

USE IEEE.std\_logic\_1164.all;

ARCHITECTURE one OF tm IS

COMPONENT mux2\_1 IS

PORT (l,r : IN std\_logic;

sel : IN std\_logic;

muxout : OUT std\_logic);

END COMPONENT;

FOR all : mux2\_1 USE ENTITY work.mux2\_1(one);

-- signals

SIGNAL l,r,sel,muxout : std\_logic;

BEGIN

-- hook up DUT

dut : mux2\_1 PORT MAP (l,r,sel,muxout);

-- create stimulut

PROCESS

BEGIN

l <= '0'; r<= '0'; sel<= '0';

WAIT FOR 10 ns;

l <= '1'; r <= '0'; sel <= '1';

WAIT FOR 10 ns;

l <= '0'; r<= '1'; sel<= '0';

WAIT FOR 10 ns;

l <= '0'; r<= '1'; sel<= '1';

WAIT FOR 10 ns;

l <= '0'; r<= '0'; sel<= '0';

WAIT FOR 10 ns;

l <= '1'; r<= '1'; sel<= '0';

WAIT FOR 10 ns;

l <= '0'; r<= '0'; sel<= '1';

WAIT FOR 10 ns;

l <= '1'; r<= '1'; sel<= '1';

WAIT FOR 10 ns;

l <= '1'; r<= '0'; sel<= '0';

WAIT FOR 10 ns;

l <= '0'; r<= '1'; sel<= '0';

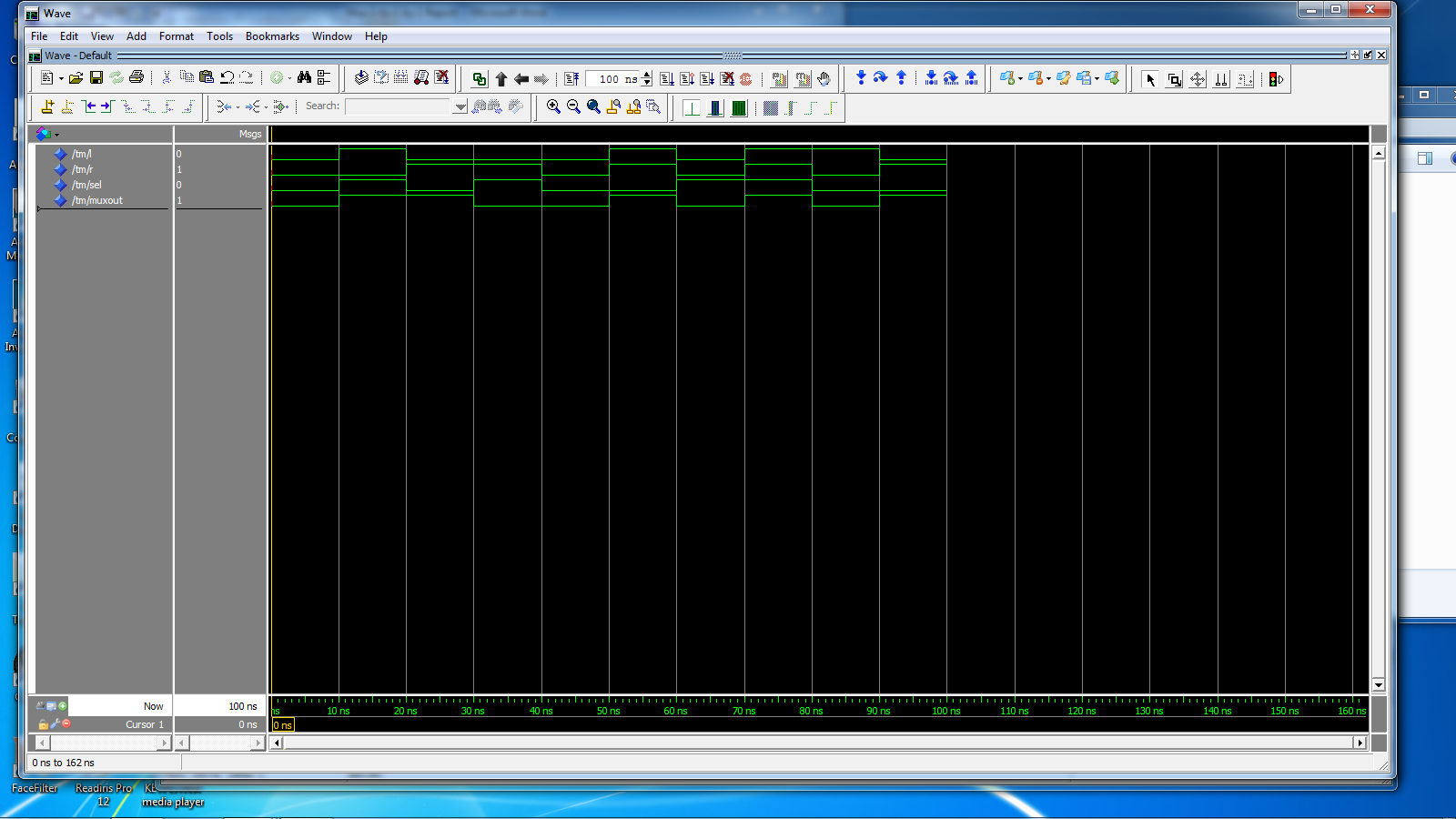
WAIT FOR 10 ns;

WAIT;

END PROCESS;

END one;

Simulation of this testbench and model produced the following simulation results:



**2.0 Quartis Synthesis of the 2-to-1 mux produced the following results**

Synthesis was successful and the summary report reported the following:

Combinational ALUTs : 1

Logic Registers : 0

Total Pins : 4

The RTL viewer generated the following view of the logic:



This is a very simple unit and the is the complete logic and is exactly as expected.