ECGR 4146 Introduction to VHDL
Lab 3
Jonathon Nguyen
ID: 801093003

Objective of the lab:

The objective of the lab is to complete the architecture for the 2 to 1 mux, design and write the testbench in order to simulate the mux. The mux will be simulated and the waveform will be recorded.

The entity code was provided.

VHDL code for the 2 to 1 Mux:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity mux_2x1 is
    Port ( in1 : in STD_LOGIC;
           in2 : in STD_LOGIC;
           sel : in STD_LOGIC;
           output : out STD_LOGIC);
end mux_2x1;

architecture Behavioral of mux_2x1 is
begin

    process (in1, in2, sel)
    begin
        if (sel = '0') then
            output <= in1;
            else
            output <= in2;
        end if;
    end process;

end Behavioral;
```

The architecture was coded by making a process to select the wires. The process is sesenitive to the signals in1, in2, and sel. When any of those signal changes, the process will run

Inside the process, there is a if statement that checks the bit on the select wire. If the sel is equal to 0, then the bit on the in1 will be passed to the output. If the sel is equal to 1, then the bit on the in2 will be passed to the output.

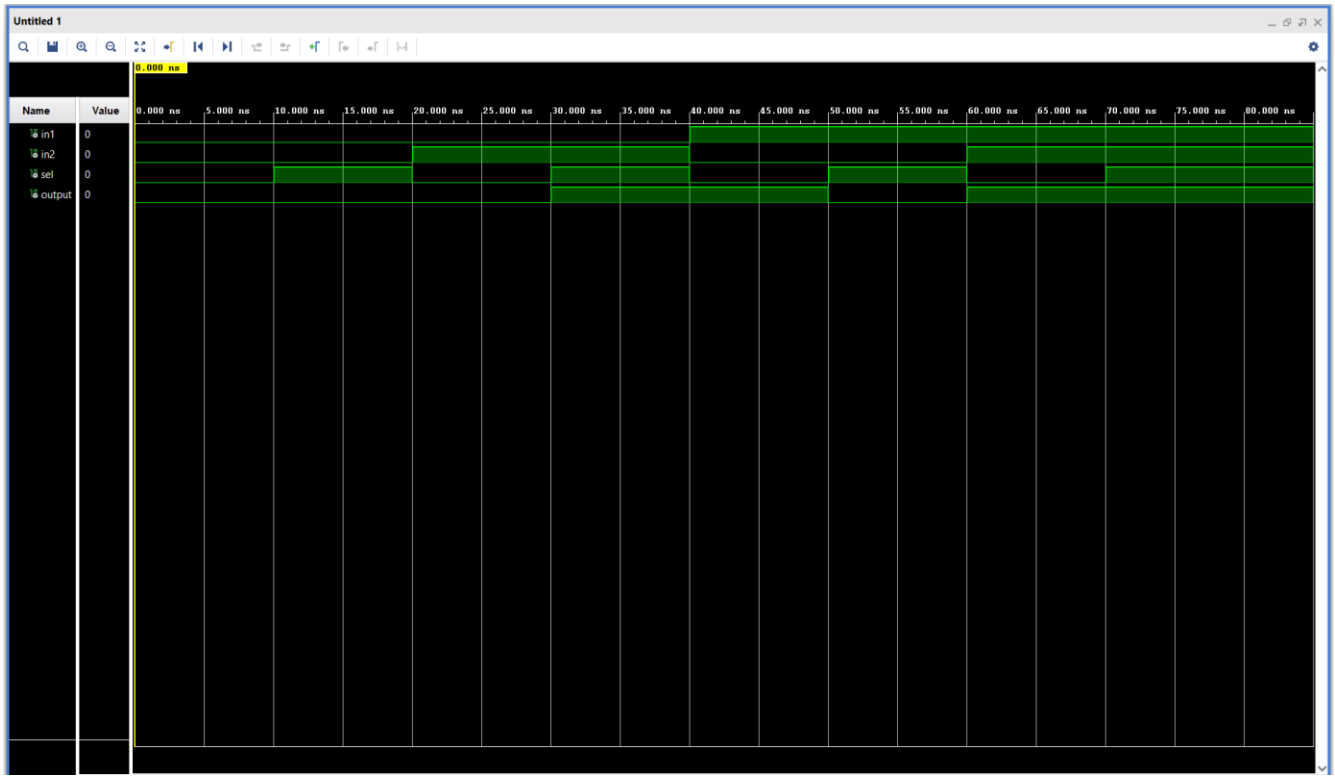
Testbench code for the 2 to 1 Mux:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;
library work;

entity mux2x1tb is
--  Port ( );
end mux2x1tb;

architecture Behavioral of mux2x1tb is
component mux_2x1 is
    port (
    in1 : in STD_LOGIC;
    in2 : in STD_LOGIC;
    sel : in STD_LOGIC;
    output : out STD_LOGIC
    );
end component;

-- Signal assignments-
signal in1 : std_logic;
signal in2 : std_logic;
signal sel : std_logic;
signal output : std_logic;

begin
UUT : entity work.mux_2x1
        port map (
        in1 => in1,
        in2 => in2,
        sel => sel,
        output => output);

    Process
    variable temp : std_logic_vector(2 downto 0);
    begin

        for i in 0 to 7 loop
            temp := std_logic_vector(to_unsigned(i, 3));
            in1 <= temp(2);
            in2 <= temp(1);
            sel <= temp(0);
            wait for 10 ns;
        end loop;

    wait;
    end process;


end Behavioral;
```

In the testbench code, the entity does not have anything in it. In the architecture, the mux was defined as a component with all the ports it needs. The signal was defined to connect to the mux.

The mux was instancated while also connecting the signals to it. A process was defined with no senetivity list. In the process, A for loop was used to loop through all the different input combination. The integer I is casted to a unsigned integer and then converted to a standard vector. Then the vector was split up for the different input.

Simulation Waveform for the Mux:



The simulation was ran on the testbench code and produce this waveform. Looking at the waveform, the architecture was designed correctly.