1 D4 - TEKKOM B

POST TEST

COMPARATOR 2 BIT

Nama	:	Septian Bagus Jumantoro
Kelas	:	1 – D4 Teknik Komputer B
NRP	:	3221600039
Dosen	:	Reni Soelistijorini B.Eng., MT.
Mata Kuliah	:	Praktikum Rangkaian Logika 2
Hari/Tgl. Praktikum	:	Jumat, 8 April 2022

POST TEST
COMPARATOR 2 BIT

I. TUJUAN

1. Mahasiswa dapat mendisain rangkaian dari Comparator 2bit.

II. TEORI

Fungsi dasar dari komparator adalah untuk membandingkan magnitudo relatif dua kuantitas. Komparator mempunyai tiga output :

- A=B (A sama dengan B)
- A>B (A lebih besar dari B)
- A<B (A kurang dari B)

Hanya satu dari tiga output dapat TINGGI. Output tinggi memberitahu bahwa apakah bit digital lebih kecil, lebih besar, atau sama dari input binary.

Tabel Kebenaran dari Comparator 1 digit.

A	B	A<B	A=B	A>B
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0

Tabel Kebenaran dari Comparator 2 digit.

INPUT				OUTPUT		
A1	A0	B1	B0	A<B	A=B	A>B
0	0	0	0	0	1	0
0	0	0	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	0	1	0

III. ALAT DAN BAHAN

1. Software ISE Design Suite Xilinx

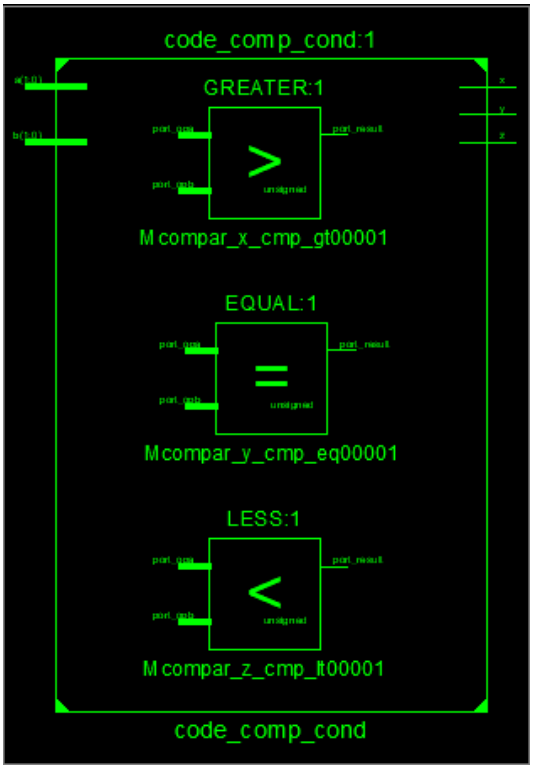
IV. LANGKAH PERCOBAAN

A. Comparator Statement pengkondisian

1. Code VHDL Statement pengkondisian

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

entity code_comp_cond is
port (
            a : in std_logic_vector (1 downto 0);
            b : in std_logic_vector (1 downto 0);
            x, y, z : out std_logic
            );
end code_comp_cond;

architecture Behavioral of code_comp_cond is

begin
x <= '1' when (a > b) else '0';
y <= '1' when (a = b) else '0';
z <= '1' when (a < b) else '0';

end Behavioral;
```

2. RTL Schematic Statement Pengkondisian

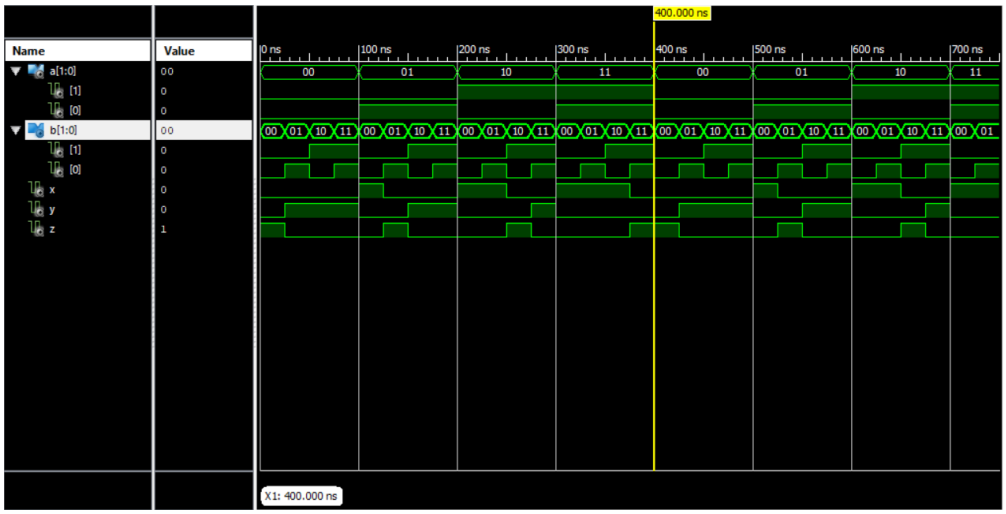


3. TestBench Statement pengkondisian

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY tb_comp_cond IS
END tb_comp_cond;

ARCHITECTURE behavior OF tb_comp_cond IS

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

    COMPONENT code_comp_cond
    PORT(
         a : IN  std_logic_vector(1 downto 0);
         b : IN  std_logic_vector(1 downto 0);
         x : OUT  std_logic;
         y : OUT  std_logic;
         z : OUT  std_logic
        );
    END COMPONENT;


   --Inputs
   signal a : std_logic_vector(1 downto 0) := (others => '0');
   signal b : std_logic_vector(1 downto 0) := (others => '0');

        --Outputs
   signal x : std_logic;
   signal y : std_logic;
   signal z : std_logic;
   -- No clocks detected in port list. Replace <clock> below with
   -- appropriate port name

BEGIN

        -- Instantiate the Unit Under Test (UUT)
   uut: code_comp_cond PORT MAP (
          a => a,
          b => b,
          x => x,
          y => y,
          z => z
        );

   stim_proc: process
   begin
        a <= "00";
        b <= "00";
        wait for 50 ns;
        b <= "01";
        wait for 50 ns;
        b <= "10";
        wait for 50 ns;
        b <= "11";
        wait for 50 ns;
        a <= "01";
        b <= "00";
```

```
        wait for 50 ns;
        b <= "01";
        wait for 50 ns;
        b <= "10";
        wait for 50 ns;
        b <= "11";
        wait for 50 ns;
        a <= "10";
        b <= "00";
        wait for 50 ns;
        b <= "01";
        wait for 50 ns;
        b <= "10";
        wait for 50 ns;
        b <= "11";
        wait for 50 ns;
        a <= "11";
        b <= "00";
        wait for 50 ns;
        b <= "01";
        wait for 50 ns;
        b <= "10";
        wait for 50 ns;
        b <= "11";
        wait for 50 ns;
        wait;
    end process;

END;
```

4. Timing Diagram Statement pengkondisian

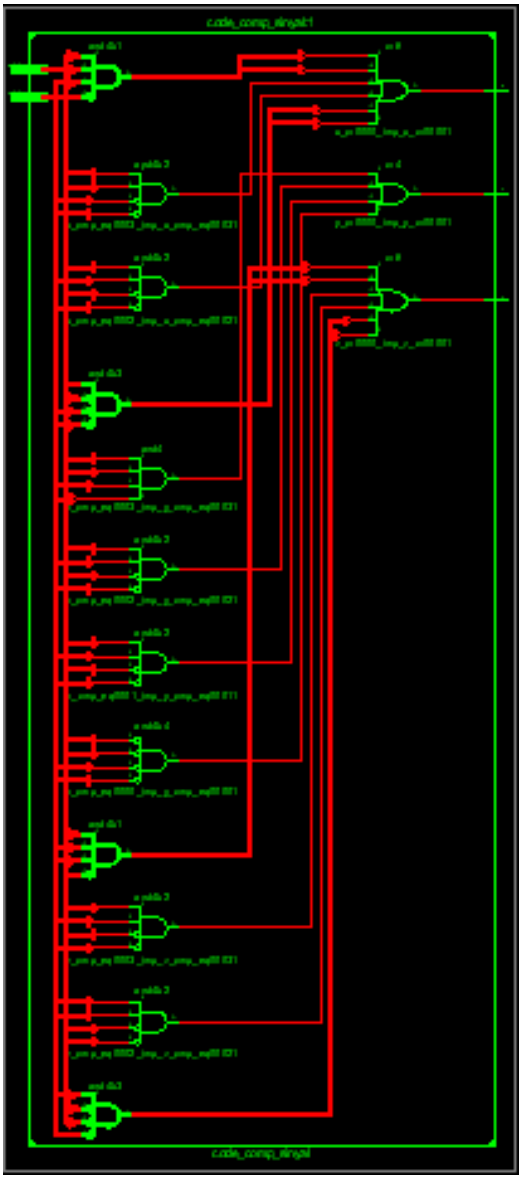


B. Comparator Seleksi Sinyal

1. Code VHDL Seleksi Sinyal

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

entity code_comp_sinyal is
port (
              a, b : in std_logic_vector (1 downto 0);
              x, y, z : out std_logic
              );
end code_comp_sinyal;

architecture Behavioral of code_comp_sinyal is
signal s : std_logic_vector (3 downto 0);

begin
s <= a & b;
with s select
x <= '1' when "0100" | "1000" | "1001" | "1100" | "1101" | "1110",
         '0' when others;
with s select
y <= '1' when "0000" | "0101" | "1010" | "1111",
         '0' when others;
with s select
z <= '1' when "0001" | "0010" | "0011" | "0110" | "0111" | "1011",
         '0' when others;

end Behavioral;
```

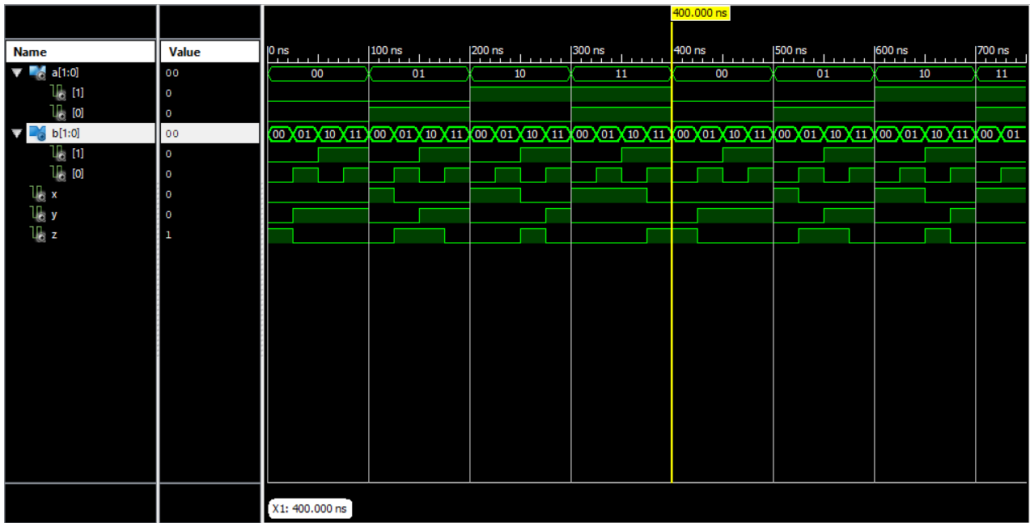
2. RTL Seleksi Sinyal



3. TestBench Seleksi Sinyal

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY tb_comp_sinyal IS
END tb_comp_sinyal;

ARCHITECTURE behavior OF tb_comp_sinyal IS

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

    COMPONENT code_comp_sinyal
    PORT(
         a : IN  std_logic_vector(1 downto 0);
         b : IN  std_logic_vector(1 downto 0);
         x : OUT  std_logic;
         y : OUT  std_logic;
         z : OUT  std_logic
        );
    END COMPONENT;


   --Inputs
   signal a : std_logic_vector(1 downto 0) := (others => '0');
   signal b : std_logic_vector(1 downto 0) := (others => '0');

          --Outputs
   signal x : std_logic;
   signal y : std_logic;
   signal z : std_logic;
   -- No clocks detected in port list. Replace <clock> below with
   -- appropriate port name

BEGIN

          -- Instantiate the Unit Under Test (UUT)
   uut: code_comp_sinyal PORT MAP (
          a => a,
          b => b,
          x => x,
          y => y,
          z => z
        );


   -- Stimulus process
   stim_proc: process
   begin
                    a <= "00";
                    b <= "00";
      wait for 50 ns;
                    b <= "01";
                    wait for 50 ns;
                    b <= "10";
                    wait for 50 ns;
                    b <= "11";
                    wait for 50 ns;
                    a <= "01";
                    b <= "00";
      wait for 50 ns;
                    b <= "01";
                    wait for 50 ns;
                    b <= "10";
                    wait for 50 ns;
                    b <= "11";
                    wait for 50 ns;
                    a <= "10";
                    b <= "00";
      wait for 50 ns;
                    b <= "01";
                    wait for 50 ns;
                    b <= "10";
                    wait for 50 ns;
                    b <= "11";
                    wait for 50 ns;
                    a <= "11";
                    b <= "00";
      wait for 50 ns;
                    b <= "01";
                    wait for 50 ns;
                    b <= "10";
                    wait for 50 ns;
                    b <= "11";
                    wait for 50 ns;
      wait;
   end process;

END;
```

4. Timing Diagram Seleksi Sinyal

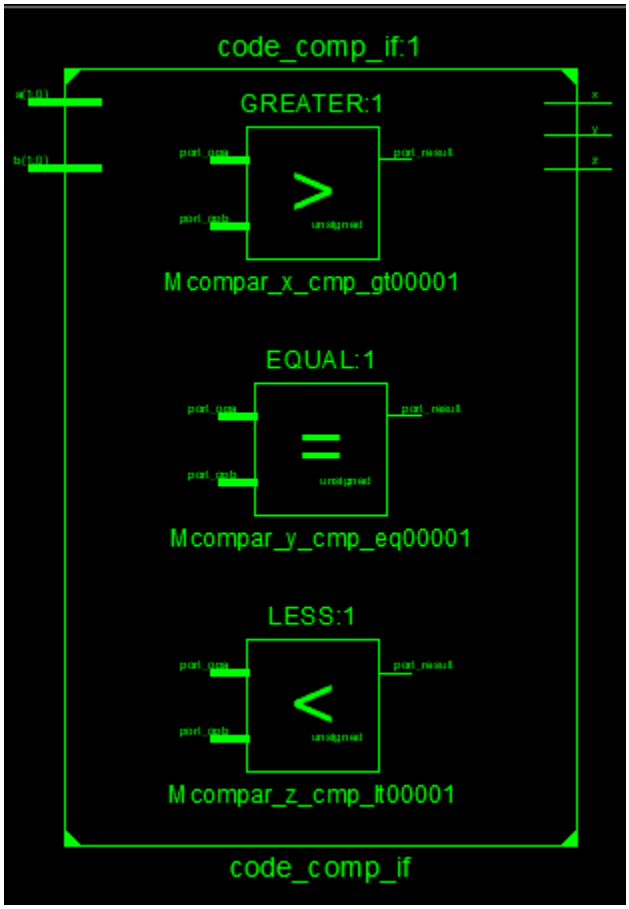


C. Comparator IF statement

1. Code VHDL IF statement

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

entity comp_if is
port (
              a, b :in std_logic_vector (1 downto 0);
              x, y, z : out std_logic
              );
end comp_if;

architecture Behavioral of comp_if is

begin
process (a, b)
begin
       if (a > b) then x <= '1';
       else x <= '0';
       end if;
       if (a = b) then y <= '1';
       else y <= '0';
       end if;
       if (a < b) then z <= '1';
       else z <= '0';
       end if;
end process;

end Behavioral;
```

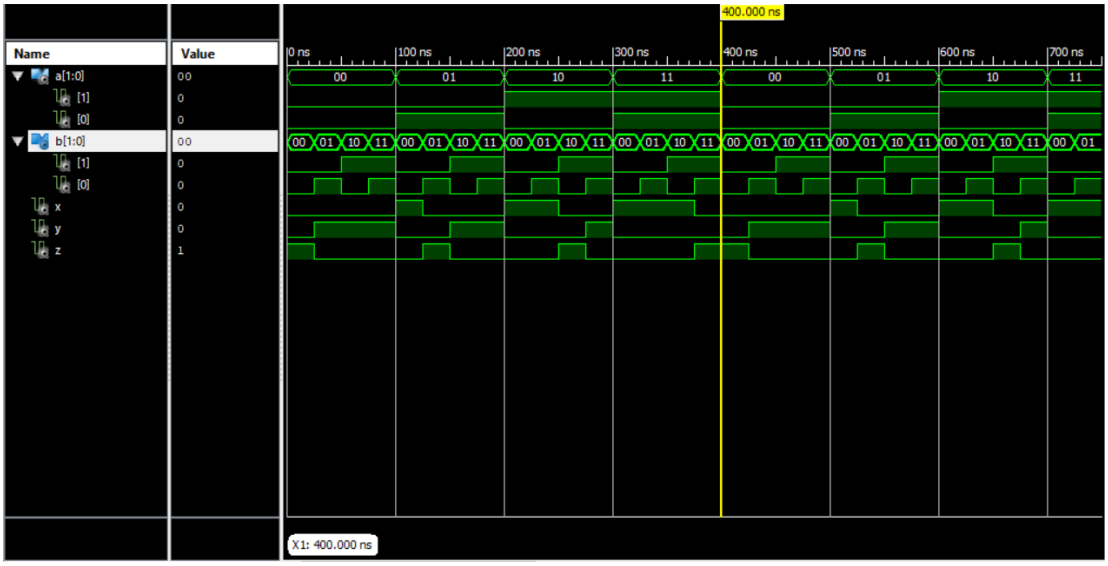
2. RTL IF statement



3. TestBench IF statement

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY tb_comp_if IS
END tb_comp_if;

ARCHITECTURE behavior OF tb_comp_if IS

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

    COMPONENT code_comp_if
    PORT(
         a : IN  std_logic_vector(1 downto 0);
         b : IN  std_logic_vector(1 downto 0);
         x : OUT  std_logic;
         y : OUT  std_logic;
         z : OUT  std_logic
        );
    END COMPONENT;


   --Inputs
   signal a : std_logic_vector(1 downto 0) := (others => '0');
   signal b : std_logic_vector(1 downto 0) := (others => '0');

          --Outputs
   signal x : std_logic;
   signal y : std_logic;
   signal z : std_logic;
   -- No clocks detected in port list. Replace <clock> below with
   -- appropriate port name

BEGIN
   A1_proc: process
   begin
          wait for 200 ns;
          A(1) <= not A(1);
   end process;
   A0_proc: process
   begin
          wait for 100 ns;
          A(0) <= not A(0);
   end process;
   B1_proc: process
   begin
          wait for 50 ns;
          B(1) <= not B(1);
   end process;
   B0_proc: process
   begin
          wait for 25 ns;
          B(0) <= not B(0);
   end process;
END;
```

4. Timing Diagram IF statement

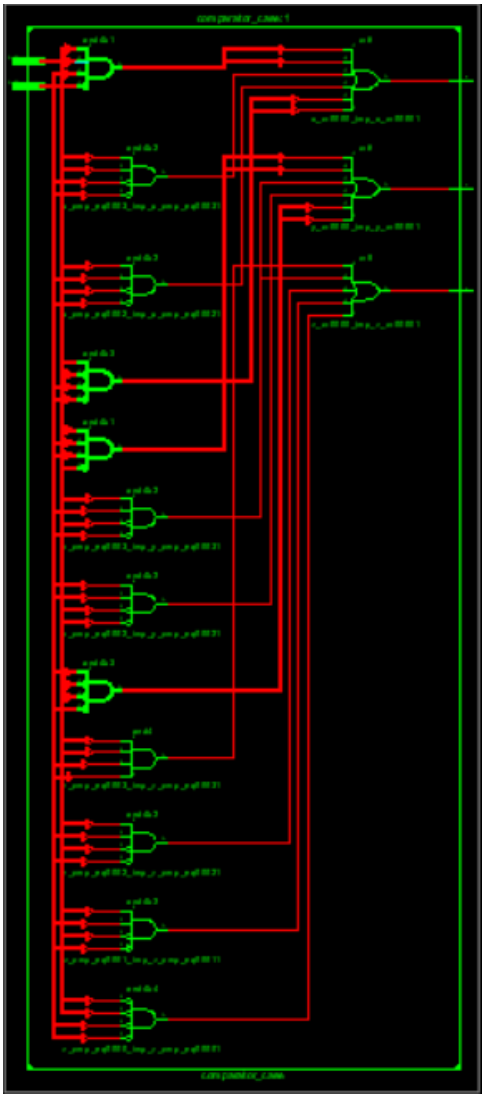


D. Comparator CASE statement

1. Code VHDL CASE statement

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity code_comp_case is
port (
              a, b : in std_logic_vector (1 downto 0);
              x, y, z : out std_logic
              );
end code_comp_case;

architecture Behavioral of code_comp_case is
signal s : std_logic_vector ( 3 downto 0);

begin
s <= a & b;
process (s)
begin
       case s is
              when "0100" | "1000" | "1001" | "1100" | "1101" | "1110" =>
              x <= '1';
              when others => x <= '0';
       end case;
       case s is
              when "0000" | "0101" | "1001" | "1010" | "1111" =>
              y <= '1';
              when others => y <= '0';
       end case;
       case s is
              when "0001" | "0010" | "0011" | "0110" | "0111" | "1011" =>
              z <= '1';
              when others => z <= '0';
       end case;
end process;
end Behavioral;
```

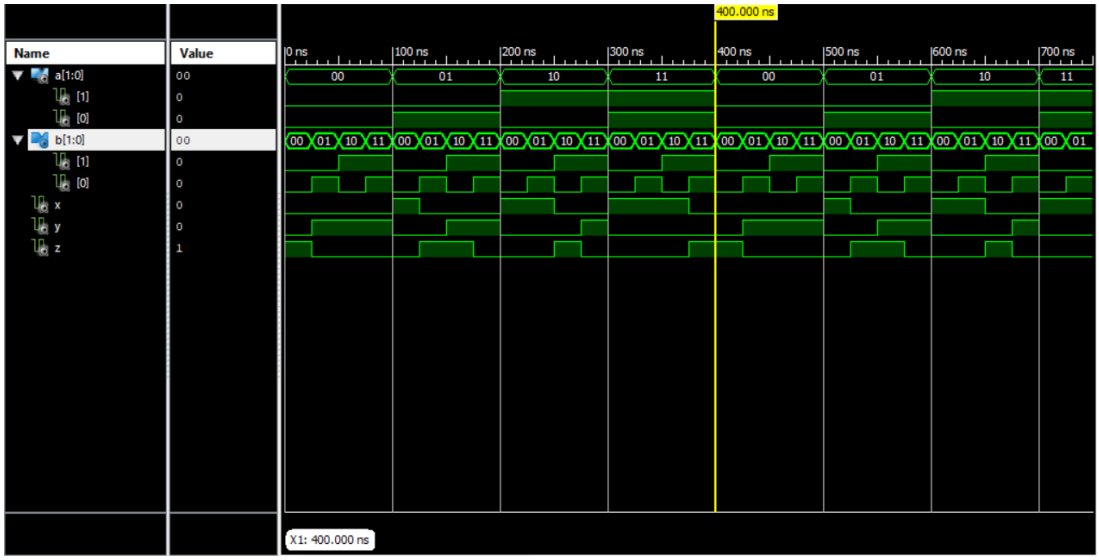
2. RTL CASE statement



3. TestBench CASE statement

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY tb_comp_case IS
END tb_comp_case;

ARCHITECTURE behavior OF tb_comp_case IS

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

    COMPONENT code_comp_case
    PORT(
         a : IN  std_logic_vector(1 downto 0);
         b : IN  std_logic_vector(1 downto 0);
         x : OUT  std_logic;
         y : OUT  std_logic;
         z : OUT  std_logic
        );
    END COMPONENT;


   --Inputs
   signal a : std_logic_vector(1 downto 0) := (others => '0');
   signal b : std_logic_vector(1 downto 0) := (others => '0');

          --Outputs
   signal x : std_logic;
   signal y : std_logic;
   signal z : std_logic;
   -- No clocks detected in port list. Replace <clock> below with
   -- appropriate port name

BEGIN

          -- Instantiate the Unit Under Test (UUT)
   uut: code_comp_case PORT MAP (
          a => a,
          b => b,
          x => x,
          y => y,
          z => z
        );

   -- Stimulus process
   stim_proc: process
   begin
                    a <= "00";
                    b <= "00";
      wait for 50 ns;
                    b <= "01";
                    wait for 50 ns;
                    b <= "10";
                    wait for 50 ns;
                    b <= "11";
                    wait for 50 ns;
                    a <= "01";
                    b <= "00";
      wait for 50 ns;
                    b <= "01";
                    wait for 50 ns;
                    b <= "10";
                    wait for 50 ns;
                    b <= "11";
                    wait for 50 ns;
                    a <= "10";
                    b <= "00";
      wait for 50 ns;
                    b <= "01";
                    wait for 50 ns;
                    b <= "10";
                    wait for 50 ns;
                    b <= "11";
                    wait for 50 ns;
                    a <= "11";
                    b <= "00";
      wait for 50 ns;
                    b <= "01";
                    wait for 50 ns;
                    b <= "10";
                    wait for 50 ns;
                    b <= "11";
                    wait for 50 ns;
                    wait;
   end process;

END;
```

4. Timing Diagram CASE statement



V. ANALISA

No.

Date:

Pada post test tersebut merupakan program VHDL Comparator 2 bit dengan 4 macam cara pembuatannya, yaitu menggunakan pengkondisian, penyeleksian sinyal, if dan case. Pada satement pengkondisian, ketika sinyal yang pertama true / sesuai, maka ekspresi diterapkan pada target. Jika tidak ada, maka else yang akan diterapkan. Pada statement penyeleksian sinyal menggunakan kondisi tunggal untuk memilih diantara beberapa opsi. Pada statement if melode kerjanya hampir sama dengan statement pengkondisian. Pada statement case metode kerjanya hampir sama dengan statement penyeleksian sinyal.

Pada rangkaian Comparator 2 bit tersebut memiliki 3 output yaitu $y_0 (A<B)$, $y_1 (A>B)$, $y_2 (A=B)$. Dimana salah satu dari ke-3 outputnya bernilai 1 (high). Untuk persamaannya sebagai berikut :

$y_0 (A<B) = (\overline{A_2} B_2) + (\overline{A_1} B_2 B_0) + (\overline{A_2} \overline{A_1} B_1)$

$y_1 (A>B) = (A_2 \overline{B_2}) + (A_1 \overline{B_2} \overline{B_1}) + (A_2 A_1 \overline{B_1})$

$y_2 (A=B) = \overline{(A_1 \oplus B_1)} \; \overline{(A_2 \oplus B_2)}$