Lab Report

Nano Processor

Fernando T.H.L - 210167E

Lab Tasks

In this lab, students are tasked with designing and implementing a 4-bit nano processor capable of executing 4 instructions. The lab focuses on developing components such as an add/subtract unit, an adder, a program counter, and multiplexers. Students will also create an assembly program to find the summation of values between 1 and 3 using the nano processor. The lab involves teamwork, simulation testing, and verifying the functionality of the design on the BASYS 3 development board. A lab report documenting the project is required, and extra credit can be earned for innovative designs and features.

Assembly program and its machine code representation

Assembly Programme

```
MOVI        R1, 3
MOVI        R2, 1
NEG         R2
ADD         R7, R1
ADD         R1, R2
JZR         R1, 7
JZR         R0, 3
```

Machine Code representation

100010000011	Move immediate value 3 to the R1 register
100100000001	Move immediate value 1 to the R2 register
010100000000	Take the 2's complement of the value in the R2 register
001110010000	Add R1 register to R7 register
000010100000	Add R2 register to the R1
110010000111	If the R1 register value is 0 jump to the instruction whose address is 7
110000000011	If the R0 register value is 0 jump to the instruction whose address is 3
110000000111	If the R0 register value is 0 jump to the instruction whose address is 7

VHDL codes

Design Codes

1. D Flip-flop

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 04/08/2023 07:00:42 AM
-- Design Name:
-- Module Name: D_FF - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity D_FF is
    Port ( D : in STD_LOGIC;
           Clk : in STD_LOGIC;
           Clr : in STD_LOGIC;
           En : in STD_LOGIC;
           Q : out STD_LOGIC);

end D_FF;

architecture Behavioral of D_FF is
```

```
begin

process (Clk) begin
        if (Clr='1') then
            Q <='0';
        else
            if (rising_edge(Clk)) then  --Check for rising-edge of
clock pulz
                if(En='1') then
                    Q  <= D;
                end if;
            end if;
       end if;
end process;

end Behavioral;
```

2. 3-to-8 Decoder

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 03/23/2023 07:19:32 AM
-- Design Name:
-- Module Name: Decoder_3_to_8 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Decoder_3_to_8 is
    Port ( I : in STD_LOGIC_VECTOR (2 downto 0);
           Y : out STD_LOGIC_VECTOR (7 downto 0));
end Decoder_3_to_8;

architecture Behavioral of Decoder_3_to_8 is
    component Decoder_2_to_4
        port(
        I: in STD_LOGIC_VECTOR;
        EN: in STD_LOGIC;
        Y: out STD_LOGIC_VECTOR );
    end component;

signal I0,I1 : STD_LOGIC_VECTOR (1 downto 0);
signal Y0,Y1 : STD_LOGIC_VECTOR (3 downto 0);
signal en0,en1, I2 : STD_LOGIC;

begin
    Decoder_2_to_4_0 : Decoder_2_to_4
        port map(
```

```
        I => I0,
        EN => en0,
        Y => Y0 );
    Decoder_2_to_4_1 : Decoder_2_to_4
        port map(
        I => I1,
        EN => en1,
        Y => Y1 );

en0 <= NOT(I(2)) ;
en1 <= I(2);

I0 <= I(1 downto 0);
I1 <= I(1 downto 0);
I2 <= I(2);

Y(3 downto 0) <= Y0;
Y(7 downto 4) <= Y1;

end Behavioral;
```

3. 2-to-4 Decoder

```
          ----------------------------------------------------------
----------------------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 03/23/2023 06:26:05 AM
-- Design Name:
-- Module Name: Decoder_2_to_4 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Decoder_2_to_4 is
    Port ( I : in STD_LOGIC_VECTOR (1 downto 0);
           EN : in STD_LOGIC;
           Y : out STD_LOGIC_VECTOR (3 downto 0));
end Decoder_2_to_4;

architecture Behavioral of Decoder_2_to_4 is

signal NI0 :  STD_LOGIC;
signal NI1 :  STD_LOGIC;

begin

    NI0 <= NOT I(0);
    NI1 <= NOT I(1);

    Y(0) <= (NI0) AND (NI1) AND EN;
    Y(1) <= I(0) AND (NI1) AND EN;
    Y(2) <= (NI0) AND I(1) AND EN;
    Y(3) <= I(0) AND I(1) AND EN;
```

```
end Behavioral;
```

4. Full Adder

```
----------------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 10:31:30 AM
-- Design Name:
-- Module Name: FA - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity FA is
    Port ( A : in STD_LOGIC;
           B : in STD_LOGIC;
           C_in : in STD_LOGIC;
           S : out STD_LOGIC;
           C_out : out STD_LOGIC);
end FA;

architecture Behavioral of FA is

    component HA
        port (
        A: in std_logic;
        B: in std_logic;
        S: out std_logic;
        C: out std_logic);
    end component;

    SIGNAL HA0_S, HA0_C, HA1_S, HA1_C : std_logic;
```

```
begin

HA_0 : HA
    port map (
    A => A,
    B => B,
    S => HA0_S,
    C => HA0_C);
HA_1 : HA
    port map (
    A => HA0_S,
    B => C_in,
    S => HA1_S,
    C => HA1_C);

S <= HA1_S;
C_out <= HA0_C OR HA1_C;

end Behavioral;
```

5. Half Adder

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 10:12:29 AM
-- Design Name:
-- Module Name: HA - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity HA is
    Port ( A : in STD_LOGIC;
           B : in STD_LOGIC;
           S : out STD_LOGIC;
           C : out STD_LOGIC);
end HA;

architecture Behavioral of HA is


begin
    S <= A XOR B;
    C <= A AND B;

end Behavioral;
```

6. Instruction Decoder

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Gamage M.S & Prabashwara D G H
--
-- Create Date: 05/29/2023 08:42:19 PM
-- Design Name:
-- Module Name: Instruction_Decoder - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Instruction_Decoder is
    Port ( Instruct_Bus : in STD_LOGIC_VECTOR (11 downto 0);
           Jump_Check_Zero : in STD_LOGIC ;
           Reg_EN : out STD_LOGIC_VECTOR (2 downto 0);
           Reg_Sel_1 : out STD_LOGIC_VECTOR (2 downto 0);
           Reg_Sel_2 : out STD_LOGIC_VECTOR (2 downto 0);
           Load_Sel : out STD_LOGIC;
           Add_Sub_Sel : out STD_LOGIC;
           Immediate_Val : out STD_LOGIC_VECTOR (3 downto 0);
           Jump_Flag : out STD_LOGIC;
           Jump_Addr : out STD_LOGIC_VECTOR (2 downto 0));
end Instruction_Decoder;

architecture Behavioral of Instruction_Decoder is

begin

Reg_EN <= Instruct_Bus(9 downto 7);
Reg_Sel_1 <= Instruct_Bus(9 downto 7);
Reg_Sel_2 <= Instruct_Bus(6 downto 4);
```

```
Jump_Addr <= Instruct_Bus(2 downto 0);
Immediate_Val <= Instruct_Bus(3 downto 0);

Load_Sel <= Instruct_Bus(11) and (not Instruct_Bus(10));
Add_Sub_Sel <= Instruct_Bus(10) and (not Instruct_Bus(11));
Jump_Flag <= Instruct_Bus(10) and Instruct_Bus(11) and Jump_Check_Zero;

end Behavioral;
```

7. Look-Up-Table for Seven Segment Display

```
----------------------------------------------------------------------------------
-- Company: UOM
-- Engineer: Gamage M.S
--
-- Create Date: 05/02/2023 03:04:33 PM
-- Design Name:
-- Module Name: LUT_16_7 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity LUT_16_7 is
    Port ( address : in STD_LOGIC_VECTOR (3 downto 0);
           data : out STD_LOGIC_VECTOR (6 downto 0));
end LUT_16_7;

architecture Behavioral of LUT_16_7 is

type rom_type is array (0 to 15) of std_logic_vector(6 downto 0);
    signal sevenSegment_ROM : rom_type :=("0000001", -- 0
                                          "1001111", --1
                                          "0010010", --2
                                          "0000110", --3
```

```
                                        "1001100", --4
                                        "0100100", --5
                                        "0100000", --6
                                        "0001111", --7
                                        "0000000", --8
                                        "0000100", --9
                                        "0000010", -- a
                                        "1100000", -- b
                                        "0110001", -- c
                                        "1000010", -- d
                                        "0110000", -- e
                                        "0111000"  -- f
                                          );

begin

data <= sevenSegment_ROM(to_integer(unsigned(address)));

end Behavioral;
```

8. 2-to-1 Multiplexer

```
----------------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 07:19:20 PM
-- Design Name:
-- Module Name: MUX_2_to_1 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity MUX_2_to_1 is
    Port ( I : in STD_LOGIC_VECTOR (1 downto 0);
           S : in STD_LOGIC;
           Y : out STD_LOGIC);
end MUX_2_to_1;

architecture Behavioral of MUX_2_to_1 is
```

```
begin

Y <= (I(0) and ( not S )) or (I(1) and S);

end Behavioral;
```

9. 3-bit 2-to-1 Multiplexer

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 09:06:53 PM
-- Design Name:
-- Module Name: MUX_2_to_1_3bit - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity MUX_2_to_1_3bit is
    Port ( I1 : in STD_LOGIC_VECTOR (2 downto 0);
           I2 : in STD_LOGIC_VECTOR (2 downto 0);
           O : out STD_LOGIC_VECTOR (2 downto 0);
           S : in STD_LOGIC);
end MUX_2_to_1_3bit;

architecture Behavioral of MUX_2_to_1_3bit is

component MUX_2_to_1
    port(
        I : in STD_LOGIC_VECTOR (1 downto 0);
        S : in STD_LOGIC;
        Y : out STD_LOGIC
    );

end component;

begin
```

```
MUX_2_to_1_0 : MUX_2_to_1
port map(
    I(0) => I1(0),
    I(1) => I2(0),
    S => S,
    Y => O(0)
);


MUX_2_to_1_1 : MUX_2_to_1
port map(
    I(0) => I1(1),
    I(1) => I2(1),
    S => S,
    Y => O(1)
);


MUX_2_to_1_2 : MUX_2_to_1
port map(
    I(0) => I1(2),
    I(1) => I2(2),
    S => S,
    Y => O(2)
);

end Behavioral;
```

10. 4-bit 2-to-1 Multiplexer

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 09:16:12 PM
-- Design Name:
-- Module Name: MUX_2_to_1_4bit - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity MUX_2_to_1_4bit is
    Port ( I0 : in STD_LOGIC_VECTOR (3 downto 0);
           I1 : in STD_LOGIC_VECTOR (3 downto 0);
           O : out STD_LOGIC_VECTOR (3 downto 0);
           S : in STD_LOGIC);
end MUX_2_to_1_4bit;

architecture Behavioral of MUX_2_to_1_4bit is

component MUX_2_to_1
    port(
        I : in STD_LOGIC_VECTOR (1 downto 0);
        S : in STD_LOGIC;
        Y : out STD_LOGIC
    );

end component;

begin
```

```
MUX_2_to_1_0 : MUX_2_to_1
port map(
    I(0) => I0(0),
    I(1) => I1(0),
    S => S,
    Y => O(0)
);


MUX_2_to_1_1 : MUX_2_to_1
port map(
    I(0) => I0(1),
    I(1) => I1(1),
    S => S,
    Y => O(1)
);


MUX_2_to_1_2 : MUX_2_to_1
port map(
    I(0) => I0(2),
    I(1) => I1(2),
    S => S,
    Y => O(2)
);

MUX_2_to_1_3 : MUX_2_to_1
port map(
    I(0) => I0(3),
    I(1) => I1(3),
    S => S,
    Y => O(3)
);

end Behavioral;
```

11.8 to 1 Multiplexer

```
----------------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 09:53:38 PM
-- Design Name:
-- Module Name: MUX_8_to_1 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Mux_8_to_1 is
    Port ( S : in STD_LOGIC_VECTOR (2 downto 0);
           I : in STD_LOGIC_VECTOR (7 downto 0);
           Y : out STD_LOGIC);
end Mux_8_to_1;

architecture Behavioral of Mux_8_to_1 is
     component Decoder_3_to_8
       port(
       I: in STD_LOGIC_VECTOR;
       Y: out STD_LOGIC_VECTOR);
     end component;

signal D1 : STD_LOGIC_VECTOR(7 downto 0);

begin
    Decoder_3_to_8_0 : Decoder_3_to_8
    port map(
            I => S,
            Y => D1);
```

```
Y <=  (I(0) AND D1(0)) OR (I(1) AND D1(1)) OR (I(2) AND D1(2)) OR (I(3)
AND D1(3)) OR (I(4) AND D1(4)) OR (I(5) AND D1(5)) OR (I(6) AND D1(6))
OR (I(7) AND D1(7));

end Behavioral;
```

12. 4-bit 8-to-1 Multiplexer

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 10:08:00 PM
-- Design Name:
-- Module Name: MUX_8_to_1_4bit - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity MUX_8_to_1_4bit is
    Port ( I1 : in STD_LOGIC_VECTOR (3 downto 0);
           I2 : in STD_LOGIC_VECTOR (3 downto 0);
           I3 : in STD_LOGIC_VECTOR (3 downto 0);
           I4 : in STD_LOGIC_VECTOR (3 downto 0);
           I5 : in STD_LOGIC_VECTOR (3 downto 0);
           I6 : in STD_LOGIC_VECTOR (3 downto 0);
           I7 : in STD_LOGIC_VECTOR (3 downto 0);
           I8 : in STD_LOGIC_VECTOR (3 downto 0);
           S : in STD_LOGIC_VECTOR (2 downto 0);
           Y : out STD_LOGIC_VECTOR (3 downto 0));
end MUX_8_to_1_4bit;

architecture Behavioral of MUX_8_to_1_4bit is

component MUX_8_to_1
    port(
        S : in STD_LOGIC_VECTOR (2 downto 0);
        I : in STD_LOGIC_VECTOR (7 downto 0);
        Y : out STD_LOGIC
```

```
    );
end component;

begin

MUX_8_to_1_0 : MUX_8_to_1
port map(
    I(0) => I1(0),
    I(1) => I2(0),
    I(2) => I3(0),
    I(3) => I4(0),
    I(4) => I5(0),
    I(5) => I6(0),
    I(6) => I7(0),
    I(7) => I8(0),
    S => S,
    Y => Y(0)
);

MUX_8_to_1_1 : MUX_8_to_1
port map(
    I(0) => I1(1),
    I(1) => I2(1),
    I(2) => I3(1),
    I(3) => I4(1),
    I(4) => I5(1),
    I(5) => I6(1),
    I(6) => I7(1),
    I(7) => I8(1),
    S => S,
    Y => Y(1)
);


MUX_8_to_1_2 : MUX_8_to_1
port map(
    I(0) => I1(2),
    I(1) => I2(2),
    I(2) => I3(2),
    I(3) => I4(2),
    I(4) => I5(2),
    I(5) => I6(2),
    I(6) => I7(2),
    I(7) => I8(2),
    S => S,
    Y => Y(2)
);

MUX_8_to_1_3 : MUX_8_to_1
port map(
    I(0) => I1(3),
    I(1) => I2(3),
    I(2) => I3(3),
    I(3) => I4(3),
    I(4) => I5(3),
    I(5) => I6(3),
    I(6) => I7(3),
```

```
    I(7) => I8(3),
    S => S,
    Y => Y(3)
);


end Behavioral;
```

13. Nano Processor

```
----------------------------------------------------------------------------
-----------
-- Company:
-- Engineer:
--
-- Create Date: 05/31/2023 01:58:15 PM
-- Design Name:
-- Module Name: Nano_Processor_4bit - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Nano_Processor_4bit is
    Port ( RES : in STD_LOGIC;
           CLK : in STD_LOGIC;
           O : out STD_LOGIC_VECTOR (3 downto 0);
           OVF : out STD_LOGIC;
           OSD : out STD_LOGIC_VECTOR (6 downto 0);
           Z : out STD_LOGIC);
end Nano_Processor_4bit;

architecture Behavioral of Nano_Processor_4bit is

component Instruction_Decoder is
    Port ( Instruct_Bus : in STD_LOGIC_VECTOR (11 downto 0);
           Jump_Check_Zero : in STD_LOGIC ;
           Reg_EN : out STD_LOGIC_VECTOR (2 downto 0);
           Reg_Sel_1 : out STD_LOGIC_VECTOR (2 downto 0);
           Reg_Sel_2 : out STD_LOGIC_VECTOR (2 downto 0);
           Load_Sel : out STD_LOGIC;
           Add_Sub_Sel : out STD_LOGIC;
           Immediate_Val : out STD_LOGIC_VECTOR (3 downto 0);
```

```
           Jump_Flag : out STD_LOGIC;
           Jump_Addr : out STD_LOGIC_VECTOR (2 downto 0));
end component;


component ROM_8_12 is
    Port ( M : in STD_LOGIC_VECTOR (2 downto 0);
           I : out STD_LOGIC_VECTOR (11 downto 0));
end component;


component RCAS_4 is
    Port ( A             : in STD_LOGIC_VECTOR (3 DOWNTO 0);
           B             : in STD_LOGIC_VECTOR (3 DOWNTO 0);
           ADD_SUB_SELECT : in STD_LOGIC; --ADD :0 , SUBTRACT :1
           S             : out STD_LOGIC_VECTOR (3 DOWNTO 0);
           OVERFLOW      : out STD_LOGIC;
           ZERO          : out STD_LOGIC);
end component;


component Reg_Bank is
    Port ( D : in STD_LOGIC_VECTOR (3 downto 0);
           EN : in STD_LOGIC_VECTOR (2 downto 0);
           CLK : in STD_LOGIC;
           CLR : in STD_LOGIC;
           O0 : out STD_LOGIC_VECTOR (3 downto 0);
           O1 : out STD_LOGIC_VECTOR (3 downto 0);
           O2 : out STD_LOGIC_VECTOR (3 downto 0);
           O3 : out STD_LOGIC_VECTOR (3 downto 0);
           O4 : out STD_LOGIC_VECTOR (3 downto 0);
           O5 : out STD_LOGIC_VECTOR (3 downto 0);
           O6 : out STD_LOGIC_VECTOR (3 downto 0);
           O7 : out STD_LOGIC_VECTOR (3 downto 0));
end component;


component MUX_2_to_1_4bit is
    Port ( I0 : in STD_LOGIC_VECTOR (3 downto 0);
           I1 : in STD_LOGIC_VECTOR (3 downto 0);
           O : out STD_LOGIC_VECTOR (3 downto 0);
           S : in STD_LOGIC);
end component;


component MUX_8_to_1_4bit is
    Port ( I1 : in STD_LOGIC_VECTOR (3 downto 0);
           I2 : in STD_LOGIC_VECTOR (3 downto 0);
           I3 : in STD_LOGIC_VECTOR (3 downto 0);
           I4 : in STD_LOGIC_VECTOR (3 downto 0);
           I5 : in STD_LOGIC_VECTOR (3 downto 0);
           I6 : in STD_LOGIC_VECTOR (3 downto 0);
           I7 : in STD_LOGIC_VECTOR (3 downto 0);
           I8 : in STD_LOGIC_VECTOR (3 downto 0);
           S : in STD_LOGIC_VECTOR (2 downto 0);
           Y : out STD_LOGIC_VECTOR (3 downto 0));
end component;
```

```
component  MUX_2_to_1_3bit is
    Port ( I1 : in STD_LOGIC_VECTOR (2 downto 0);
           I2 : in STD_LOGIC_VECTOR (2 downto 0);
           O : out STD_LOGIC_VECTOR (2 downto 0);
           S : in STD_LOGIC);
end component;


component PC_3 is
    Port ( D : in STD_LOGIC_VECTOR (2 downto 0);
           CLR : in STD_LOGIC;
           CLK : in STD_LOGIC;
           Q : out STD_LOGIC_VECTOR (2 downto 0));
end component;


component RCA_3 is
    Port ( A : in STD_LOGIC_VECTOR (2 downto 0);
        B : in STD_LOGIC_VECTOR (2 downto 0);
        S : out STD_LOGIC_VECTOR (2 downto 0);
        C_out : out STD_LOGIC);
end component;


component Slow_CLK is
    Port ( Clk_in : in STD_LOGIC;
           Clk_out : out STD_LOGIC);
end component;

component LUT_16_7 is
    Port ( address : in STD_LOGIC_VECTOR (3 downto 0);
           data : out STD_LOGIC_VECTOR (6 downto 0));
end component;


signal PC_Current_Address, PC_Next_Address, Jump_Address, Adder_Address
: std_logic_vector(2 downto 0);

signal Instrction : std_logic_vector(11 downto 0);
signal Immediate_Val, Result_Val ,Data_Val : std_logic_vector(3 downto
0);
signal Reg_En, Reg_Sel_1, Reg_Sel_2 :  std_logic_vector(2 downto 0);
signal Load_Sel,  Add_Sub_Sel : std_logic;
signal Zero : std_logic;
signal REG_0,REG_1,REG_2,REG_3,REG_4,REG_5,REG_6,REG_7 :
std_logic_vector(3 downto 0);
signal MUX_REG_0, MUX_REG_1 :  std_logic_vector(3 downto 0);

signal clock : std_logic;
signal Jump_Flag : std_logic;
signal PC_OVF : std_logic;

begin

Slow_CLK_0 : Slow_CLK
```

```
    Port map (
        Clk_in => CLK,
        Clk_out => clock
   );

PC_3_0 : PC_3
port map(
    D => PC_Next_Address,
    CLR => RES,
    CLK => clock,
    Q => PC_Current_Address
);


 MUX_2_to_1_3bit_0 : MUX_2_to_1_3bit
    Port map( I1 => Adder_Address,
           I2 => Jump_Address,
           O => PC_Next_Address,
           S =>  Jump_Flag
    );



RCA_3_0 : RCA_3
    Port map ( A => PC_Current_Address,
        B => "001",
        S => Adder_Address,
        C_out => PC_OVF
    );


ROM_8_12_0 : ROM_8_12
    Port map( M => PC_Current_Address,
           I => Instrction
    );


Instruction_Decoder_0 : Instruction_Decoder
    Port map( Instruct_Bus =>  Instrction,
           Jump_Check_Zero => Zero,
           Reg_EN =>  Reg_En,
           Reg_Sel_1 => Reg_Sel_1,
           Reg_Sel_2 => Reg_Sel_2 ,
           Load_Sel => Load_Sel,
           Add_Sub_Sel =>  Add_Sub_Sel,
           Immediate_Val =>Immediate_Val ,
           Jump_Flag => Jump_Flag,
           Jump_Addr => Jump_Address);


MUX_2_to_1_4bit_0 : MUX_2_to_1_4bit
    Port map( I0 =>  Result_Val,
           I1 => Immediate_Val,
           O => Data_Val,
           S =>  Load_Sel);
```

```
Reg_Bank_0 : Reg_Bank
    Port map( D => Data_Val,
           EN => Reg_En,
           CLK => clock,
           CLR => RES,
           O0 => REG_0,
           O1 => REG_1,
           O2 => REG_2,
           O3 => REG_3,
           O4 => REG_4,
           O5 => REG_5,
           O6 => REG_6,
           O7 => REG_7
           );


MUX_8_to_1_4bit_0 : MUX_8_to_1_4bit
    Port map ( I1 => REG_0,
           I2 => REG_1,
           I3 => REG_2,
           I4 => REG_3,
           I5 => REG_4,
           I6 => REG_5,
           I7 => REG_6,
           I8 => REG_7,
           S => Reg_Sel_1,
           Y => MUX_REG_0 );


MUX_8_to_1_4bit_1 : MUX_8_to_1_4bit
    Port map ( I1 => REG_0,
           I2 => REG_1,
           I3 => REG_2,
           I4 => REG_3,
           I5 => REG_4,
           I6 => REG_5,
           I7 => REG_6,
           I8 => REG_7,
           S => Reg_Sel_2,
           Y => MUX_REG_1 );


RCAS_4_0 : RCAS_4
   Port map( A => MUX_REG_1,
          B => MUX_REG_0,
          ADD_SUB_SELECT => Add_Sub_Sel,
          S => Result_Val,
          OVERFLOW => OVF,
          ZERO => Zero);

LUT_16_7_0 :  LUT_16_7
    Port map(address => Reg_7,
           data => OSD);

O<=reg_7;
Z<=Zero;
```

```
end Behavioral;
```

14. 3-bit Program Counter

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 12:14:12 PM
-- Design Name:
-- Module Name: PC_3 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

library UNISIM;
use UNISIM.Vcomponents.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity PC_3 is
    Port ( D : in STD_LOGIC_VECTOR (2 downto 0);
           CLR : in STD_LOGIC;
           CLK : in STD_LOGIC;
           Q : out STD_LOGIC_VECTOR (2 downto 0));
end PC_3;

architecture Behavioral of PC_3 is

 component D_FF
    port (
        D : in STD_LOGIC;
        Clk : in STD_LOGIC;
        Clr : in STD_LOGIC;
        En : in STD_LOGIC;
        Q : out STD_LOGIC);
    end component;
```

```
signal D_IN : std_logic_vector(2 downto 0);

begin


    D_FF0 : D_FF
    port map (
        D => D(0),
        En => '1',
        Clk => CLK,
        Clr => CLR,
        Q => Q(0));

    D_FF1 : D_FF
    port map (
        D => D(1),
        En => '1',
        Clk => CLK,
        Clr => CLR,
        Q => Q(1));

    D_FF2 : D_FF
    port map (
        D => D(2),
        En => '1',
        Clk => CLK,
        Clr => CLR,
        Q => Q(2));


end Behavioral;
```

15. 3-bit Ripple Carry Adder

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 11:52:57 AM
-- Design Name:
-- Module Name: RCA_3 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity RCA_3 is
    Port ( A : in STD_LOGIC_VECTOR (2 downto 0);
           B : in STD_LOGIC_VECTOR (2 downto 0);
           S : out STD_LOGIC_VECTOR (2 downto 0);
           C_out : out STD_LOGIC);
end RCA_3;

architecture Behavioral of RCA_3 is
    component FA
    port (
        A: in std_logic;
        B: in std_logic;
        C_in: in std_logic;
        S: out std_logic;
        C_out: out std_logic);
    end component;

    component HA
    port(
        A : in STD_LOGIC;
```

```
        B : in STD_LOGIC;
        S : out STD_LOGIC;
        C : out STD_LOGIC);
    end component;

    SIGNAL  HA0_C, FA1_C : std_logic;

begin
    HA_0 : HA
        port map (
            A => A(0),
            B => B(0),
            S => S(0),
            C => HA0_C);
    FA_1 : FA
        port map (
            A => A(1),
            B => B(1),
            C_in => HA0_C,
            S => S(1),
            C_Out => FA1_C);
    FA_2 : FA
        port map (
            A => A(2),
            B => B(2),
            C_in => FA1_C,
            S => S(2),
            C_Out => C_out);

end Behavioral;
```

16. 4-bit Adder Subtractor

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Gamage M.S
--
-- Create Date: 03/14/2023 03:52:53 PM
-- Design Name:
-- Module Name: RCA_4 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity RCAS_4 is
    Port ( A              : in STD_LOGIC_VECTOR (3 DOWNTO 0);
           B              : in STD_LOGIC_VECTOR (3 DOWNTO 0);
           ADD_SUB_SELECT : in STD_LOGIC; --ADD :0 , SUBTRACT :1
           S              : out STD_LOGIC_VECTOR (3 DOWNTO 0);
           OVERFLOW       : out STD_LOGIC;
           ZERO           : out STD_LOGIC);
end RCAS_4;

architecture Behavioral of RCAS_4 is

    component FA
     port (
         A     : in std_logic;
         B     : in std_logic;
         C_IN  : in std_logic;
         S     : out std_logic;
         C_OUT : out std_logic);
     end component;
```

```
    SIGNAL  FA0_C, FA1_C, FA2_C ,FA3_C : std_logic;
    SIGNAL B_1,RESULT : STD_LOGIC_VECTOR (3 DOWNTO 0);

begin
    FA_0 : FA
        port map (
            A => A(0),
            B => B_1(0),
            C_IN => ADD_SUB_SELECT,
            S => RESULT(0),
            C_OUT => FA0_C);
    FA_1 : FA
        port map (
            A => A(1),
            B => B_1(1),
            C_IN => FA0_C,
            S => RESULT(1),
            C_OUT => FA1_C);
    FA_2 : FA
        port map (
            A => A(2),
            B => B_1(2),
            C_IN => FA1_C,
            S => RESULT(2),
            C_OUT => FA2_C);
    FA_3 : FA
        port map (
            A => A(3),
            B => B_1(3),
            C_IN => FA2_C,
            S => RESULT(3),
            C_OUT => FA3_C);

    B_1(0) <= ADD_SUB_SELECT XOR B(0);
    B_1(1) <= ADD_SUB_SELECT XOR B(1);
    B_1(2) <= ADD_SUB_SELECT XOR B(2);
    B_1(3) <= ADD_SUB_SELECT XOR B(3);

    S <= RESULT;
    ZERO <= NOT ( RESULT(0) OR RESULT(1) OR RESULT(2) OR RESULT(3) );
    OVERFLOW <= FA2_C XOR FA3_C;



end Behavioral;
```

17.4-bit Register

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 04/21/2023 06:47:19 PM
-- Design Name:
-- Module Name: Reg_4 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

library UNISIM;
use UNISIM.Vcomponents.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Reg_4 is
    Port ( D : in STD_LOGIC_VECTOR (3 downto 0);
           EN : in STD_LOGIC;
           CLR : in STD_LOGIC;
           CLK : in STD_LOGIC;
           Q : out STD_LOGIC_VECTOR (3 downto 0));
end Reg_4;

architecture Behavioral of Reg_4 is

 component D_FF
    port (
        D : in STD_LOGIC;
        Clk : in STD_LOGIC;
        Clr : in STD_LOGIC;
        En : in STD_LOGIC;
        Q : out STD_LOGIC);
```

```
    end component;

signal D_IN : std_logic_vector(3 downto 0);
signal EN_CLOCK : std_logic;

begin


    D_FF0 : D_FF
    port map (
        D => D(0),
        En => EN,
        Clr => CLR,
        Clk => CLK,
        Q => Q(0));

    D_FF1 : D_FF
    port map (
        D => D(1),
        En => EN,
        Clr => CLR,
        Clk => CLK,
        Q => Q(1));

    D_FF2 : D_FF
    port map (
        D => D(2),
        En => EN,
        Clr => CLR,
        Clk => CLK,
        Q => Q(2));

    D_FF3 : D_FF
    port map (
        D => D(3),
        En => EN,
        Clr => CLR,
        Clk => CLK,
        Q => Q(3));


end Behavioral;
```

18. Register Bank

```
----------------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/23/2023 11:33:03 AM
-- Design Name:
-- Module Name: Reg_Bank - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Reg_Bank is
    Port ( D : in STD_LOGIC_VECTOR (3 downto 0);
           EN : in STD_LOGIC_VECTOR (2 downto 0);
           CLK : in STD_LOGIC;
           CLR : in STD_LOGIC;
           O0 : out STD_LOGIC_VECTOR (3 downto 0);
           O1 : out STD_LOGIC_VECTOR (3 downto 0);
```

```
            O2 : out STD_LOGIC_VECTOR (3 downto 0);
            O3 : out STD_LOGIC_VECTOR (3 downto 0);
            O4 : out STD_LOGIC_VECTOR (3 downto 0);
            O5 : out STD_LOGIC_VECTOR (3 downto 0);
            O6 : out STD_LOGIC_VECTOR (3 downto 0);
            O7 : out STD_LOGIC_VECTOR (3 downto 0));
end Reg_Bank;


architecture Behavioral of Reg_Bank is

    component Reg_4
    port (
        D : in STD_LOGIC_VECTOR (3 downto 0);
        EN : in STD_LOGIC;
        CLR : in STD_LOGIC;
        CLK : in STD_LOGIC;
        Q : out STD_LOGIC_VECTOR (3 downto 0));
    end component;

    component Decoder_3_to_8
    port (
        I : in STD_LOGIC_VECTOR (2 downto 0);
        Y : out STD_LOGIC_VECTOR (7 downto 0));
    end component;

signal DecodedEN : std_logic_vector (7 downto 0);


begin

    Decoder_3_to_8_1 : Decoder_3_to_8
    port map(
        I => EN,
        Y => DecodedEN
    );

    Reg_4_0 : Reg_4
    port map(
        D => D,
        EN => DecodedEN(0),
        CLR => CLR,
        CLK => CLK
        --Q => O0 -- Set R0 to zero
```

```
);

Reg_4_1 : Reg_4
    port map(
        D => D,
        EN => DecodedEN(1),
        CLR => CLR,
        CLK => CLK,
        Q => O1
    );

Reg_4_2 : Reg_4
    port map(
        D => D,
        EN => DecodedEN(2),
        CLR => CLR,
        CLK => CLK,
        Q => O2
    );

Reg_4_3 : Reg_4
port map(
    D => D,
    EN => DecodedEN(3),
    CLR => CLR,
    CLK => CLK,
    Q => O3
);

Reg_4_4 : Reg_4
port map(
    D => D,
    EN => DecodedEN(4),
    CLR => CLR,
    CLK => CLK,
    Q => O4
);

Reg_4_5 : Reg_4
port map(
    D => D,
    EN => DecodedEN(5),
    CLR => CLR,
    CLK => CLK,
```

```
        Q => O5
    );

    Reg_4_6 : Reg_4
    port map(
        D => D,
        EN => DecodedEN(6),
        CLR => CLR,
        CLK => CLK,
        Q => O6
    );

    Reg_4_7 : Reg_4
    port map(
        D => D,
        EN => DecodedEN(7),
        CLR => CLR,
        CLK => CLK,
        Q => O7
    );

 O0<="0000";

end Behavioral;
```

19.8-to-12 ROM

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Gamage M.S
--
-- Create Date: 05/19/2023 06:42:40 PM
-- Design Name:
-- Module Name: ROM_8_12 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity ROM_8_12 is
    Port ( M : in STD_LOGIC_VECTOR (2 downto 0);
           I : out STD_LOGIC_VECTOR (11 downto 0));
end ROM_8_12;

architecture Behavioral of ROM_8_12 is

type rom_type is array (0 to 7) of std_logic_vector(11 downto 0);
    signal ROM_12 : rom_type :=("100010000011",--0rom address 000
                                "100100000001",--1rom address 001
                                "010100000000",--2rom address 010
                                "001110010000",--3rom address 011
                                "000010100000",--4rom address 100
                                "110010000111",--5rom address 101
                                "110000000011",--6rom address 110
                                "110000000111" --7rom address 111
                                );

begin
I <= ROM_12(to_integer(unsigned(M)));
```

```
end Behavioral;
```

20. Slow Clock

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Gamage M.S
--
-- Create Date: 05/31/2023 02:22:37 PM
-- Design Name:
-- Module Name: Slow_CLK - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Slow_CLK is
    Port ( Clk_in : in STD_LOGIC;
           Clk_out : out STD_LOGIC);
end Slow_CLK;

architecture Behavioral of Slow_CLK is

signal count : integer := 1;
signal clk_status:  std_logic :='0';


begin

--For 100 MHz input clock this g nerates 1 Hz clock
    process (Clk_in) begin
        if (rising_edge(Clk_in)) then
            count <= count + 1;               --Increment: counter
            if(count = 50000000) then        --Count 50M pluses (1/2 of
period)
                clk_status <= not clk_status; -- Inve-rt clock status
```

```
                Clk_out <= clk_status;
                count<= 1;                              --RE.se-c councer
end if;
            end if;
        end if;
    end process;

end Behavioral;
```

Test Bench Codes

1. D Flip-flop

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/23/2023 06:31:00 AM
-- Design Name:
-- Module Name: TB_D_FF - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

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

architecture Behavioral of TB_D_FF is

component D_FF
    Port ( D : in STD_LOGIC;
           Clk : in STD_LOGIC;
           Clr : in STD_LOGIC;
           En : in STD_LOGIC;
           Q : out STD_LOGIC);
end component;

signal D,Clr,En: STD_LOGIC;
signal Q: STD_LOGIC;
signal Clk:STD_LOGIC:='0';
```

```
begin
UUT: D_FF PORT MAP(
   D => D,
   Clr => Clr,
   Clk => Clk,
   En => En,
   Q => Q
);

process
begin
wait for 50 ns;
Clk <= not Clk;

end process;

process
begin
    En <= '1';
    D <= '0';
    Clr <= '0';
    wait for 50ns;

    D <= '0';
    Clr <= '0';
    wait for 50ns;


    D <= '0';
    Clr <= '1';
    wait for 50ns;


    D <= '0';
    Clr <= '1';
    wait for 50ns;


    D <= '1';
    Clr <= '0';
    wait for 50ns;


    D <= '1';
    Clr <= '0';
    wait for 50ns;


    D <= '1';
    Clr <= '1';
    wait for 50ns;


    D <= '1';
    Clr <= '1';
    wait for 50ns;
```

```
    D <= '0';
    Clr <= '0';


    wait;

end process;

end Behavioral;
```

2. 3 to 8 Decoder

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 03/23/2023 07:24:47 AM
-- Design Name:
-- Module Name: TB_Decoder_3_to_8 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_Decoder_3_to_8 is
--  Port ( );
end TB_Decoder_3_to_8;

architecture Behavioral of TB_Decoder_3_to_8 is

COMPONENT Decoder_3_to_8
    PORT(I: IN STD_LOGIC_VECTOR(2 downto 0);
     Y: OUT STD_LOGIC_VECTOR(7 downto 0));
END COMPONENT;

signal I: STD_LOGIC_VECTOR(2 downto 0);
signal Y: STD_LOGIC_VECTOR(7 downto 0);

begin
UUT: Decoder_3_to_8 PORT MAP(
    I => I,
    Y => Y
);
```

```
process
begin

    I(0) <= '0';
    I(1) <= '0';
    I(2) <= '0';
    wait for 60 ns;

    I(0) <= '1';
    I(1) <= '0';
    I(2) <= '0';
    wait for 60 ns;

    I(0) <= '0';
    I(1) <= '1';
    I(2) <= '0';
    wait for 60 ns;

    I(0) <= '1';
    I(1) <= '1';
    I(2) <= '0';
    wait for 60 ns;

    I(0) <= '0';
    I(1) <= '0';
    I(2) <= '1';
    wait for 60 ns;

    I(0) <= '1';
    I(1) <= '0';
    I(2) <= '1';
    wait for 60 ns;

    I(0) <= '0';
    I(1) <= '1';
    I(2) <= '1';
    wait for 60 ns;

    I(0) <= '1';
    I(1) <= '1';
    I(2) <= '1';
    wait for 60 ns;


    wait;

end process;
end Behavioral;
```

3. 2 to 4 Decoder

```
----------------------------------------------------------------------------------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 03/23/2023 06:29:30 AM
-- Design Name:
-- Module Name: TB_Decoder_2_to_4 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_Decoder_2_to_4 is
--  Port ( );
end TB_Decoder_2_to_4;

architecture Behavioral of TB_Decoder_2_to_4 is

COMPONENT Decoder_2_to_4
    PORT(I: IN STD_LOGIC_VECTOR (1 downto 0);
     EN: IN STD_LOGIC;
     Y: OUT STD_LOGIC_VECTOR (3 downto 0)
     );
END COMPONENT;

signal I : STD_LOGIC_VECTOR (1 downto 0);
signal Y : STD_LOGIC_VECTOR (3 downto 0);
signal EN : STD_LOGIC;


begin
```

```
UUT: Decoder_2_to_4  PORT MAP(
    I => I,
    EN => EN,
    Y => Y
);

process
begin
    EN <= '1';

    I(0) <= '0';
    I(1) <= '0';
    wait for 100 ns;

    I(0) <= '1';
    I(1) <= '0';
    wait for 100 ns;

    I(0) <= '0';
    I(1) <= '1';
    wait for 100 ns;

    I(0) <= '1';
    I(1) <= '1';
    wait for 100 ns;

    EN <= '0';

    I(0) <= '0';
    I(1) <= '0';
    wait for 100 ns;

    I(0) <= '1';
    I(1) <= '0';
    wait for 100 ns;

    I(0) <= '0';
    I(1) <= '1';
    wait for 100 ns;

    I(0) <= '1';
    I(1) <= '1';
    wait for 100 ns;


    I(0) <= '0';
    I(1) <= '0';
    wait;

end process;

end Behavioral;
```

4. Full Adder

```
----------------------------------------------------------------------------
-----------
-- Company:
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 10:47:33 AM
-- Design Name:
-- Module Name: TB_FA - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

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

architecture Behavioral of TB_FA is

COMPONENT FA
    PORT(A, B, C_in: IN STD_LOGIC;
     S, C_out: OUT STD_LOGIC);
END COMPONENT;

SIGNAL A, B, C_in: std_logic;
SIGNAL S, C_out: std_logic;

begin
UUT: FA PORT MAP(
    A => A,
    B => B,
    C_in => C_in,
    S => S,
    C_out => C_out
```

```
);


process
begin
    A <= '0';
    B <= '0';
    C_in <= '0';
    wait for 125 ns;

    A <= '0';
    B <= '0';
    C_in <= '1';
    wait for 125 ns;

    A <= '0';
    B <= '1';
    C_in <= '0';
    wait for 125 ns;

    A <= '0';
    B <= '1';
    C_in <= '1';
    wait for 125 ns;

    A <= '1';
    B <= '0';
    C_in <= '0';
    wait for 125 ns;

    A <= '1';
    B <= '0';
    C_in <= '1';
    wait for 125 ns;

    A <= '1';
    B <= '1';
    C_in <= '0';
    wait for 125 ns;

    A <= '1';
    B <= '1';
    C_in <= '1';

    wait;

end process;
end Behavioral;
```

5. Half Adder

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 10:41:36 AM
-- Design Name:
-- Module Name: TB_HA - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

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

architecture Behavioral of TB_HA is

COMPONENT HA
    PORT(A, B: IN STD_LOGIC;
     S, C: OUT STD_LOGIC);
END COMPONENT;

SIGNAL A, B: std_logic;
SIGNAL S, C: std_logic;


begin
UUT: HA PORT MAP(
    A => A,
    B => B,
    S => S,
```

```
    C => C
);

process
begin
    A <= '0';
    B <= '0';
    wait for 250 ns;

    A <= '0';
    B <= '1';
    wait for 250 ns;

    A <= '1';
    B <= '0';
    wait for 250 ns;

    A <= '1';
    B <= '1';
    wait;
    end process;

end Behavioral;
```

6. Instruction Decoder

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Gamage M S
--
-- Create Date: 05/29/2023 09:43:01 PM
-- Design Name:
-- Module Name: TB_Instruction_Decoder - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

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

architecture Behavioral of TB_Instruction_Decoder is

component Instruction_Decoder
    Port ( Instruct_Bus : in STD_LOGIC_VECTOR (11 downto 0);
           Jump_Check_Zero : in STD_LOGIC;
           Reg_EN : out STD_LOGIC_VECTOR (2 downto 0);
           Reg_Sel_1 : out STD_LOGIC_VECTOR (2 downto 0);
           Reg_Sel_2 : out STD_LOGIC_VECTOR (2 downto 0);
           Load_Sel : out STD_LOGIC;
           Add_Sub_Sel : out STD_LOGIC;
           Immediate_Val : out STD_LOGIC_VECTOR (3 downto 0);
           Jump_Flag : out STD_LOGIC;
           Jump_Addr : out STD_LOGIC_VECTOR (2 downto 0));
end component;
```

```
signal Instruct_Bus :  STD_LOGIC_VECTOR (11 downto 0);
signal Jump_Check_Zero :  STD_LOGIC;
signal Reg_EN :  STD_LOGIC_VECTOR (2 downto 0);
signal Reg_Sel_1 :  STD_LOGIC_VECTOR (2 downto 0);
signal Reg_Sel_2 :  STD_LOGIC_VECTOR (2 downto 0);
signal Load_Sel :  STD_LOGIC;
signal Add_Sub_Sel :  STD_LOGIC;
signal Immediate_Val :  STD_LOGIC_VECTOR (3 downto 0);
signal Jump_Flag :  STD_LOGIC;
signal Jump_Addr :  STD_LOGIC_VECTOR (2 downto 0);

constant delay : time    := 150 ns;

begin
UUT : Instruction_Decoder
port map(
    Instruct_Bus => Instruct_Bus ,
    Jump_Check_Zero => Jump_Check_Zero,
    Reg_EN =>  Reg_EN,
    Reg_Sel_1 => Reg_Sel_1,
    Reg_Sel_2 => Reg_Sel_2,
    Load_Sel => Load_Sel,
    Add_Sub_Sel => Add_Sub_Sel,
    Immediate_Val => Immediate_Val,
    Jump_Flag => Jump_Flag,
    Jump_Addr => Jump_Addr
);

process begin
Jump_Check_Zero <= '0';
Instruct_Bus <= "100010000010";
wait for delay;

Instruct_Bus <= "100100000001";
wait for delay;

Instruct_Bus <= "010100000000";
wait for delay;

Instruct_Bus <= "000010100000";
wait for delay;


Instruct_Bus <= "110010000111";
wait for delay;

Jump_Check_Zero <= '1';
Instruct_Bus <= "110000000011";
wait for delay;

wait;

end process;

end Behavioral;
```

7. Look-Up-Table for Seven Segment Display

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Gamasge M S
--
-- Create Date: 05/02/2023 03:24:48 PM
-- Design Name:
-- Module Name: LUT_Sim - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_LUT_16_7 is
--  Port ( );
end TB_LUT_16_7;

architecture Behavioral of TB_LUT_16_7 is
component LUT_16_7
    port(
    address : in STD_LOGIC_VECTOR (3 downto 0);
    data : out STD_LOGIC_VECTOR (6 downto 0));
end component;
signal address: STD_LOGIC_VECTOR (3 downto 0);
signal data : STD_LOGIC_VECTOR (6 downto 0);

begin
UUT: LUT_16_7
    Port Map ( address => address,
           data => data);

process
    begin
```

```
        address <= "0010";--2
        wait for 100ns;
        address <="0001";--1
        wait for 100ns;
        address <="0000";--0
        wait for 100ns;
        address <="0100";--4
        wait for 100ns;
        address <="1000";--8
        wait for 100ns;
        address <="0011";--3
        wait for 100ns;
        address <="0111";--7
        wait for 100ns;
        address <="0110";--6
        wait for 100ns;
        address <="1111";--F
        wait;
    end process;

end Behavioral;
```

8. 2-to-1 Multiplexer

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 08:52:22 PM
-- Design Name:
-- Module Name: TB_MUX_2_to_1 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_MUX_2_to_1 is
--  Port ( );
end TB_MUX_2_to_1;

architecture Behavioral of TB_MUX_2_to_1 is

component MUX_2_to_1
    port(
        I : in STD_LOGIC_VECTOR (1 downto 0);
        S : in STD_LOGIC;
        Y : out STD_LOGIC
    );
end component;

signal I :  STD_LOGIC_VECTOR (1 downto 0);
signal S,Y :  STD_LOGIC;


begin
UUT : MUX_2_to_1
```

```
port map(
    I => I,
    S => S,
    Y => Y
);

process begin

    I <= "00";
    S <= '0';
    wait for 125ns;
    S <= '1';
    wait for 125ns;

    I <= "01";
    S <= '0';
    wait for 125ns;
    S <= '1';
    wait for 125ns;

    I <= "10";
    S <= '0';
    wait for 125ns;
    S <= '1';
    wait for 125ns;

    I <= "11";
    S <= '0';
    wait for 125ns;
    S <= '1';
    wait for 125ns;

    wait;

end process;

end Behavioral;
```

9. 3-bit 2-to-1 Multiplexer

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 09:43:17 PM
-- Design Name:
-- Module Name: TB_MUX_2_to_1_3bit - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_MUX_2_to_1_3bit is
--  Port ( );
end TB_MUX_2_to_1_3bit;

architecture Behavioral of TB_MUX_2_to_1_3bit is

component MUX_2_to_1_3bit
    port(I1 : in STD_LOGIC_VECTOR (2 downto 0);
           I2 : in STD_LOGIC_VECTOR (2 downto 0);
           O : out STD_LOGIC_VECTOR (2 downto 0);
           S : in STD_LOGIC);
end component;

signal I1, I2, O : STD_LOGIC_VECTOR (2 downto 0);
signal S : STD_LOGIC;


begin
UUT : MUX_2_to_1_3bit
port map(
```

```
    I1 => I1,
    I2 => I2,
    O => O,
    S => S
);

process begin

     I1 <= "010";
    I2 <= "111";

    S <= '0';
    wait for 500ns;

    S <= '1';
    wait for 500ns;

    wait;

end process;

end Behavioral;
```

10. 4-bit 2-to-1 Multiplexer

```
----------------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 09:25:58 PM
-- Design Name:
-- Module Name: TB_MUX_2_to_1_4bit - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_MUX_2_to_1_4bit is
--  Port ( );
end TB_MUX_2_to_1_4bit;

architecture Behavioral of TB_MUX_2_to_1_4bit is

component MUX_2_to_1_4bit
    port (
        I0 : in STD_LOGIC_VECTOR (3 downto 0);
        I1 : in STD_LOGIC_VECTOR (3 downto 0);
        O : out STD_LOGIC_VECTOR (3 downto 0);
        S : in STD_LOGIC
    );
end component;

SIGNAL I0 :  STD_LOGIC_VECTOR (3 downto 0);
SIGNAL I1 :  STD_LOGIC_VECTOR (3 downto 0);
SIGNAL O : STD_LOGIC_VECTOR (3 downto 0);
SIGNAL S : STD_LOGIC;
```

```
begin
UUT : MUX_2_to_1_4bit
port map(
    I0 => I0,
    I1 => I1,
    O => O,
    S => S
);

process begin

    I0 <= "1010";
    I1 <= "1111";

    S <= '0';
    wait for 500ns;

    S <= '1';
    wait for 500ns;

    wait;

end process;

end Behavioral;
```

11.8 to 1 Multiplexer

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 09:55:19 PM
-- Design Name:
-- Module Name: TB_MUX_8_to_1 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_Mux_8_to_1 is
--  Port ( );
end TB_Mux_8_to_1;

architecture Behavioral of TB_Mux_8_to_1 is

COMPONENT Mux_8_to_1
    PORT(S: IN STD_LOGIC_VECTOR (2 downto 0);
     I: IN STD_LOGIC_VECTOR (7 downto 0);
     Y: OUT STD_LOGIC);
END COMPONENT;

signal S: STD_LOGIC_VECTOR(2 downto 0);
signal I : STD_LOGIC_VECTOR(7 downto 0);

signal Y: STD_LOGIC;

signal delay : time := 50ns;

begin
```

```
UUT: Mux_8_to_1 PORT MAP(
    S => S,
    I => I,
    Y => Y
);


process
begin

    I <= "00000001";
    S <= "000";
    wait for delay;

    I <= "11111110";
    S <= "000";
    wait for delay;


    I <= "00000010";
    S <= "001";
    wait for delay;

    I <= "11111101";
    S <= "001";
    wait for delay;


    I <= "00000100";
    S <= "010";
    wait for delay;

    I <= "11111011";
    S <= "010";
    wait for delay;


    I <= "00001000";
    S <= "011";
    wait for delay;

    I <= "11110111";
    S <= "011";
    wait for delay;


    I <= "00010000";
    S <= "100";
    wait for delay;

    I <= "11101111";
    S <= "100";
    wait for delay;


    I <= "00100000";
```

```
        S <= "101";
        wait for delay;

        I <= "11011111";
        S <= "101";
        wait for delay;


        I <= "01000000";
        S <= "110";
        wait for delay;

        I <= "10111111";
        S <= "110";
        wait for delay;


        I <= "10000000";
        S <= "111";
        wait for delay;

        I <= "01111111";
        S <= "111";
        wait for delay;


        wait;

    end process;
    end Behavioral;
```

12.4-bit 8-to-1 Multiplexer

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 10:27:50 PM
-- Design Name:
-- Module Name: TB_MUX_8_to_1_4bit - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_MUX_8_to_1_4bit is
--  Port ( );
end TB_MUX_8_to_1_4bit;

architecture Behavioral of TB_MUX_8_to_1_4bit is

component MUX_8_to_1_4bit
    port(
        I1 : in STD_LOGIC_VECTOR (3 downto 0);
        I2 : in STD_LOGIC_VECTOR (3 downto 0);
        I3 : in STD_LOGIC_VECTOR (3 downto 0);
        I4 : in STD_LOGIC_VECTOR (3 downto 0);
        I5 : in STD_LOGIC_VECTOR (3 downto 0);
        I6 : in STD_LOGIC_VECTOR (3 downto 0);
        I7 : in STD_LOGIC_VECTOR (3 downto 0);
        I8 : in STD_LOGIC_VECTOR (3 downto 0);
        S : in STD_LOGIC_VECTOR (2 downto 0);
        Y : out STD_LOGIC_VECTOR (3 downto 0)
    );
end component;
```

```
signal I1 :  STD_LOGIC_VECTOR (3 downto 0);
signal I2 :  STD_LOGIC_VECTOR (3 downto 0);
signal I3 :  STD_LOGIC_VECTOR (3 downto 0);
signal I4 :  STD_LOGIC_VECTOR (3 downto 0);
signal I5 :  STD_LOGIC_VECTOR (3 downto 0);
signal I6 :  STD_LOGIC_VECTOR (3 downto 0);
signal I7 :  STD_LOGIC_VECTOR (3 downto 0);
signal I8 :  STD_LOGIC_VECTOR (3 downto 0);
signal S :  STD_LOGIC_VECTOR (2 downto 0);
signal Y :  STD_LOGIC_VECTOR (3 downto 0);


begin
UUT : MUX_8_to_1_4bit
port map(
    I1 => I1,
    I2 => I2,
    I3 => I3,
    I4 => I4,
    I5 => I5,
    I6 => I6,
    I7 => I7,
    I8 => I8,
    S => S,
    Y => Y
);


process begin
    I1 <= "0001";
    I2 <= "0010";
    I3 <= "0011";
    I4 <= "0100";
    I5 <= "0101";
    I6 <= "0110";
    I7 <= "0111";
    I8 <= "1000";



    S <= "000";
    wait for 125 ns;

    S <= "001";
    wait for 125 ns;

    S <= "010";
    wait for 125 ns;

    S <= "011";
    wait for 125 ns;

    S <= "100";
    wait for 125 ns;
```

```
        S <= "101";
        wait for 125 ns;

        S <= "110";
        wait for 125 ns;

        S <= "111";
        wait for 125 ns;

        wait;

    end process;

    end Behavioral;
```

13. Nano Processor

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Gamage M S
--
-- Create Date: 05/31/2023 07:42:11 PM
-- Design Name:
-- Module Name: TB_Nano_Processor_4 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_Nano_Processor_4 is
--  Port ( );
end TB_Nano_Processor_4;

architecture Behavioral of TB_Nano_Processor_4 is
 component Nano_Processor_4bit is
    Port ( RES : in STD_LOGIC;
           CLK : in STD_LOGIC;
           O : out STD_LOGIC_VECTOR (3 downto 0);
           OVF : out STD_LOGIC;
           OSD : out STD_LOGIC_VECTOR (6 downto 0);
           Z : out STD_LOGIC);
end component;


signal OVF, Z  : STD_LOGIC;
signal O : STD_LOGIC_VECTOR (3 downto 0);
signal OSD : STD_LOGIC_VECTOR (6 downto 0);
signal CLK:  std_logic :='0';
signal RES:  std_logic :='1';
```

```
begin
UUT : Nano_Processor_4bit
    Port map( RES => RES,
           CLK => CLK,
           O => O,
           OVF => OVF,
           OSD =>OSD,
           Z => Z);


process
begin
    RES <='0';
    CLK <= not clk;
    wait for 2ns;
end process;

process
begin
    wait for 2ns;
    RES <='0';
end process;

end Behavioral;
```

14. 3-bit Program Counter

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 05:05:44 PM
-- Design Name:
-- Module Name: TB_PC_3 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_PC_3 is
--  Port ( );
end TB_PC_3;

architecture Behavioral of TB_PC_3 is

component PC_3
    Port ( D : in STD_LOGIC_VECTOR (2 downto 0);
           CLR: in STD_LOGIC;
           CLK : in STD_LOGIC;
           Q : out STD_LOGIC_VECTOR (2 downto 0));
end component;

signal D : STD_LOGIC_VECTOR (2 downto 0);
signal CLR : STD_LOGIC;
signal CLK : STD_LOGIC:='0';
signal Q : STD_LOGIC_VECTOR (2 downto 0);

signal clock_delay : time := 50ns;
```

```
begin
UUT: PC_3
port map(
    D => D,
    CLR => CLR,
    CLK => CLK,
    Q => Q
);

process
    begin
    wait for 25 ns;
    CLK <= not CLK;

    end process;

    process begin
         -- our indexes are 210176F and 210483T
    -- In this test bench we are storing each digits of 21, 176, and
483


       CLR <= '0';
       D <= "010";


       wait for clock_delay;
       D <= "001";

       wait for clock_delay;
       D <= "111";


       wait for clock_delay;
       D <= "110";

       wait for clock_delay;
       D <= "100";


       wait for clock_delay;
       D <= "111";
       wait for clock_delay;
       CLR<='1';
       wait for clock_delay;




       wait;

end process;


end Behavioral;
```

15. 3-bit Ripple Carry Adder

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 05/28/2023 10:53:06 AM
-- Design Name:
-- Module Name: TB_RAC_3 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_RCA_3 is
--  Port ( );
end TB_RCA_3;

architecture Behavioral of TB_RCA_3 is

COMPONENT RCA_3
    PORT(A,B : IN STD_LOGIC_VECTOR (2 downto 0);
     C_out : OUT STD_LOGIC;
     S : OUT STD_LOGIC_VECTOR (2 downto 0));
END COMPONENT;

signal A, B, S : std_logic_vector (2 downto 0);
signal C_out: std_logic;

begin
UUT: RCA_3 PORT MAP(
    A => A,
    B => B,
    S => S,
    C_out => C_out
```

```
);


process
begin
    A <= "000";
    B <= "001";
    wait for 125 ns;

    A <= "001";
    wait for 125 ns;

    A <= "010";
    wait for 125 ns;

    A <= "011";
    wait for 125 ns;

    A <= "100";
    wait for 125 ns;

    A <= "101";
    wait for 125 ns;

    A <= "110";
    wait for 125 ns;

    A <= "111";

    wait;

end process;

end Behavioral;
```

16. 4-bit Adder Subtractor

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: MALINDU GAMAGE
--
-- Create Date: 05/28/2023 08:18:31 PM
-- Design Name:
-- Module Name: TB_RCAS_4 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_RCAS_4 is
  --Port ( );
end TB_RCAS_4;

architecture Behavioral of TB_RCAS_4 is
    COMPONENT RCAS_4
        Port ( A              : in STD_LOGIC_VECTOR (3 DOWNTO 0);
               B              : in STD_LOGIC_VECTOR (3 DOWNTO 0);
               ADD_SUB_SELECT : in STD_LOGIC;
               S              : out STD_LOGIC_VECTOR (3 DOWNTO 0);
               OVERFLOW       : out STD_LOGIC;
               ZERO           : out STD_LOGIC);
    END COMPONENT;

    SIGNAL A,B,S : STD_LOGIC_VECTOR (3 DOWNTO 0);
    SIGNAL ADD_SUB_SELECT, OVERFLOW, ZERO : STD_LOGIC;

begin

UUT: RCAS_4
```

```
Port Map ( A              => A,
           B              => B,
           ADD_SUB_SELECT => ADD_SUB_SELECT,
           S              => S,
           OVERFLOW       => OVERFLOW,
           ZERO           => ZERO);
PROCESS
    BEGIN
        ADD_SUB_SELECT <='0';
        A <="0000";
        B <="0000";
        WAIT FOR 100NS;


        A <="0001";
        B <="0000";
        WAIT FOR 100NS;


        A <="0010";
        B <="0010";
        WAIT FOR 100NS;


        A <="0111";
        B <="1111";
        WAIT FOR 100NS;


        A <="0011";
        B <="0001";
        WAIT FOR 100NS;

        ADD_SUB_SELECT <='1';
        A <="1111";
        B <="0000";
        WAIT FOR 100NS;


        A <="1000";
        B <="0001";
        WAIT FOR 100NS;


        A <="1111";
        B <="1111";
        WAIT FOR 100NS;


        A <="0001";
        B <="0100";
        WAIT FOR 100NS;


        A <="0000";
        B <="0000";
        WAIT FOR 100NS;
```

```
    END PROCESS;
end Behavioral;
```

17.4-bit Register

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Prabashwara D G H
--
-- Create Date: 04/21/2023 09:13:04 PM
-- Design Name:
-- Module Name: TB_Reg - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

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

architecture Behavioral of TB_Reg is

component Reg_4
    Port ( D : in STD_LOGIC_VECTOR (3 downto 0);
           EN : in STD_LOGIC;
           CLR: in STD_LOGIC;
           CLK : in STD_LOGIC;
           Q : out STD_LOGIC_VECTOR (3 downto 0));
end component;

signal D : STD_LOGIC_VECTOR (3 downto 0);
signal EN : STD_LOGIC;
signal CLR : STD_LOGIC;
signal CLK : STD_LOGIC;
signal Q : STD_LOGIC_VECTOR (3 downto 0);
```

```
signal clock_delay : time := 7ns;

begin
UUT: Reg_4
port map(
    D => D,
    EN => EN,
    CLR => CLR,
    CLK => CLK,
    Q => Q
);

process begin

    EN <= '0';
    CLK <= '0';
    CLR <= '0';


    D <= "0000";

    CLK <= '1';
    EN <= '0';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;

    CLK <= '1';
    EN <= '1';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;


    D <= "0001";

    CLK <= '1';
    EN <= '0';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;

    CLK <= '1';
    EN <= '1';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;


    D <= "0010";

    CLK <= '1';
    EN <= '0';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;
```

```
CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "0011";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "0100";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "0101";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "0110";

CLK <= '1';
EN <= '0';
wait for clock_delay;
```

```
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "0111";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "1000";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "1001";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "1010";
```

```
CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "1011";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "1100";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "1101";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;
```

```
  D <= "1110";

  CLK <= '1';
  EN <= '0';
  wait for clock_delay;
  CLK <= '0';
  wait for clock_delay;

  CLK <= '1';
  EN <= '1';
  wait for clock_delay;
  CLK <= '0';
  wait for clock_delay;


  D <= "1111";

  CLK <= '1';
  EN <= '0';
  wait for clock_delay;
  CLK <= '0';
  wait for clock_delay;

  CLK <= '1';
  EN <= '1';
  wait for clock_delay;
  CLK <= '0';
  wait for clock_delay;
------------------------

    EN <= '0';
    CLK <= '0';
    CLR <= '1';


    D <= "0000";

    CLK <= '1';
    EN <= '0';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;

    CLK <= '1';
    EN <= '1';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;


    D <= "0001";

    CLK <= '1';
    EN <= '0';
    wait for clock_delay;
    CLK <= '0';
```

```
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "0010";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "0011";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "0100";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "0101";

CLK <= '1';
```

```
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "0110";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "0111";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "1000";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;
```

```
D <= "1001";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "1010";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "1011";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;


D <= "1100";

CLK <= '1';
EN <= '0';
wait for clock_delay;
CLK <= '0';
wait for clock_delay;

CLK <= '1';
EN <= '1';
wait for clock_delay;
CLK <= '0';
```

```
    wait for clock_delay;


    D <= "1101";

    CLK <= '1';
    EN <= '0';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;

    CLK <= '1';
    EN <= '1';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;


    D <= "1110";

    CLK <= '1';
    EN <= '0';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;

    CLK <= '1';
    EN <= '1';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;


    D <= "1111";

    CLK <= '1';
    EN <= '0';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;

    CLK <= '1';
    EN <= '1';
    wait for clock_delay;
    CLK <= '0';
    wait for clock_delay;



------------------------

  D <= "0000";

  CLK <= '1';
  EN <= '1';
  wait for clock_delay;
  CLK <= '0';
```

```
    wait for clock_delay;

    wait;
end process;

end Behavioral;
```

18. Register Bank

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineers: Prabashwara D G H & Gamage M S
--
-- Create Date: 05/23/2023 12:10:17 PM
-- Design Name:
-- Module Name: TB_Reg_Bank - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

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

architecture Behavioral of TB_Reg_Bank is

    component Reg_Bank
    Port (
        D : in STD_LOGIC_VECTOR (3 downto 0);
        EN : in STD_LOGIC_VECTOR (2 downto 0);
        CLK : in STD_LOGIC:='1';
        CLR : in STD_LOGIC;
        O0 : out STD_LOGIC_VECTOR (3 downto 0);
        O1 : out STD_LOGIC_VECTOR (3 downto 0);
        O2 : out STD_LOGIC_VECTOR (3 downto 0);
        O3 : out STD_LOGIC_VECTOR (3 downto 0);
        O4 : out STD_LOGIC_VECTOR (3 downto 0);
        O5 : out STD_LOGIC_VECTOR (3 downto 0);
        O6 : out STD_LOGIC_VECTOR (3 downto 0);
        O7 : out STD_LOGIC_VECTOR (3 downto 0)
```

```
    );
    end component;

    signal D :  STD_LOGIC_VECTOR (3 downto 0);
    signal EN :  STD_LOGIC_VECTOR (2 downto 0);
    signal CLK :  STD_LOGIC:='1';
    signal CLR :  STD_LOGIC;
    signal O0 :  STD_LOGIC_VECTOR (3 downto 0);
    signal O1 :  STD_LOGIC_VECTOR (3 downto 0);
    signal O2 :  STD_LOGIC_VECTOR (3 downto 0);
    signal O3 :  STD_LOGIC_VECTOR (3 downto 0);
    signal O4 :  STD_LOGIC_VECTOR (3 downto 0);
    signal O5 :  STD_LOGIC_VECTOR (3 downto 0);
    signal O6 :  STD_LOGIC_VECTOR (3 downto 0);
    signal O7 :  STD_LOGIC_VECTOR (3 downto 0);

    signal clock_delay : time := 25ns;

begin

    UUT: Reg_Bank
    port map(
        D => D,
        EN => EN,
        CLK => CLK,
        CLR => CLR,
        O0 => O0,
        O1 => O1,
        O2 => O2,
        O3 => O3,
        O4 => O4,
        O5 => O5,
        O6 => O6,
        O7 => O7
    );
process
    begin
    wait for 25 ns;
    CLK <= not CLK;

    end process;

    process begin
         -- our indexes are 210176F and 210483T
    -- In this test bench we are storing each digits of 21, 176, and
483
       CLR<='1';
       wait for clock_delay;

       wait for clock_delay;
       CLR <= '0';


       wait for clock_delay;
       EN <= "001";
       wait for clock_delay;
       D <= "0010";
```

```
        wait for clock_delay;
        EN <= "010";
         wait for clock_delay;
         D <= "0001";



         wait for clock_delay;
          EN <= "011";
         wait for clock_delay;
         D <= "0111";



         wait for clock_delay;
          EN <= "100";
         wait for clock_delay;
         D <= "0110";



         wait for clock_delay;
          EN <= "101";
         wait for clock_delay;
         D <= "0100";



         wait for clock_delay;
         EN <= "110";
         wait for clock_delay;
         D <= "1000";



         wait for clock_delay;
         EN <= "111";
         wait for clock_delay;
         D <= "1111";



         wait;
      end process;

end Behavioral;
```

19.8-to-12 ROM

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Gamage M S
--
-- Create Date: 05/19/2023 06:50:43 PM
-- Design Name:
-- Module Name: TB_ROM_8_12 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TB_ROM_8_12 is
--  Port ( );
end TB_ROM_8_12;

architecture Behavioral of TB_ROM_8_12 is

    component ROM_8_12
        Port ( M : in STD_LOGIC_VECTOR (2 downto 0);
               I : out STD_LOGIC_VECTOR (11 downto 0));
    end component;

    signal M: std_logic_vector (2 downto 0);
    signal I   : std_logic_vector (11 downto 0);

begin

    uut: ROM_8_12
        port map ( M => M,
                   I=> I);
    process
```

```
        begin
            M<="000";
            wait for 100ns;
            M<="001";
            wait for 100ns;
            M<="010";
            wait for 100ns;
            M<="011";
            wait for 100ns;
            M<="100";
            wait for 100ns;
            M<="101";
            wait for 100ns;
            M<="110";
            wait for 100ns;
            M<="111";
            wait;
    end process;


end Behavioral;
```

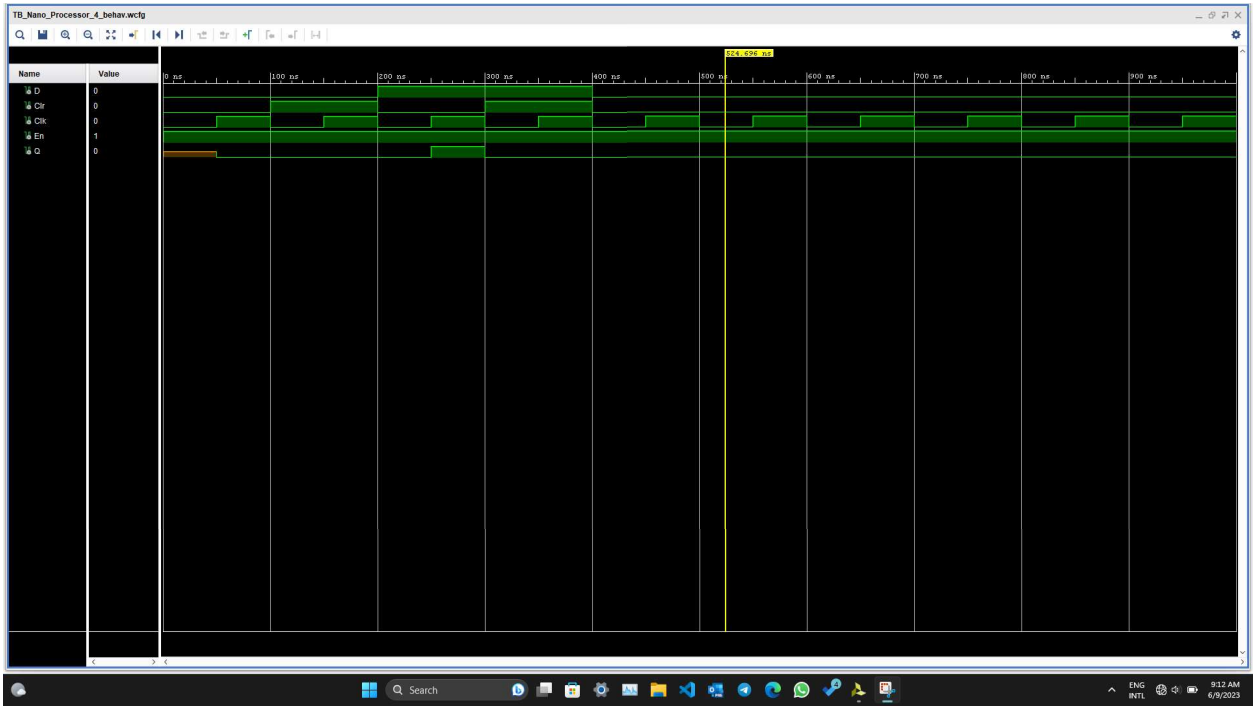
20. Slow Clock

```
----------------------------------------------------------------------------
-----------
-- Company: UOM
-- Engineer: Gamage M S
--
-- Create Date: 05/31/2023 02:23:54 PM
-- Design Name:
-- Module Name: TB_Slow_CLK - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
-----------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

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

architecture Behavioral of TB_Slow_CLK is


component Slow_CLK is
    Port ( Clk_in : in STD_LOGIC;
           Clk_out : out STD_LOGIC);
end component;

signal Clk_in, Clk_out :  STD_LOGIC;
signal clk_status:  std_logic :='0';
signal delay : time := 10 ns;

begin
UUT: Slow_Clk PORT MAP(
   Clk_in => Clk_in,
   Clk_out => Clk_out
```

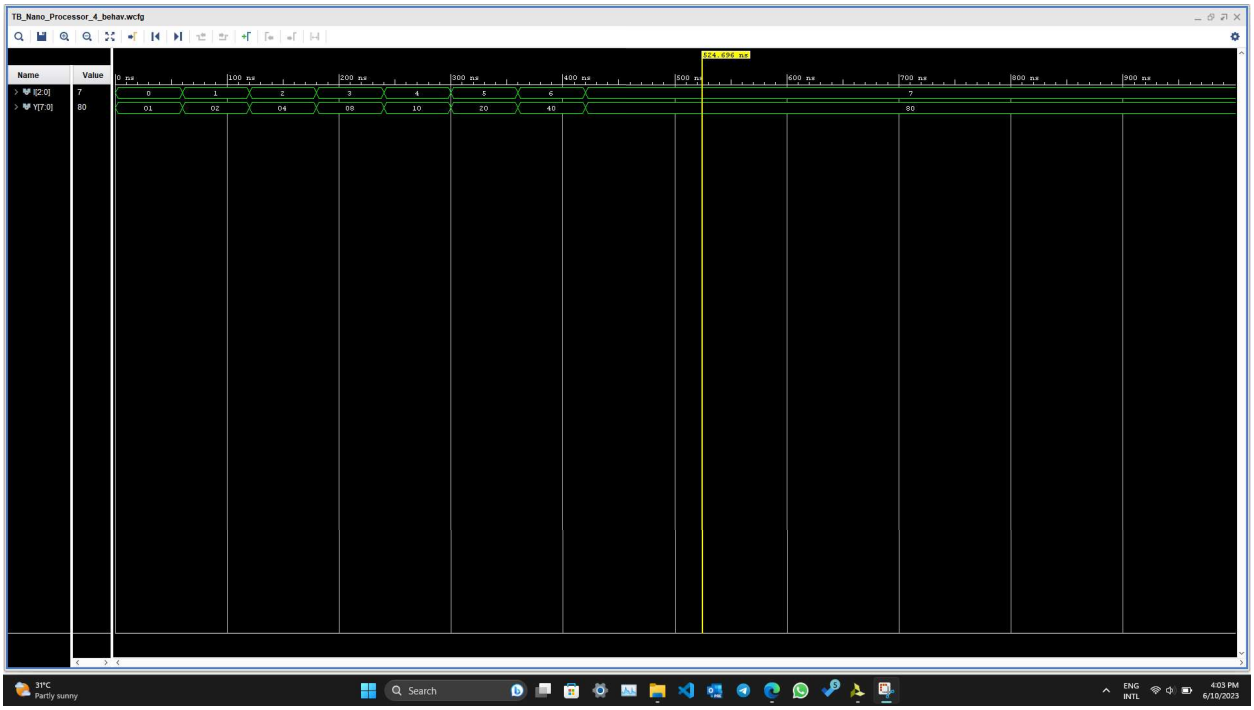
```
);


process
begin

    for i in 1 to 100 loop
        Clk_in <= clk_status;
        clk_status <= not clk_status;
        wait for delay;
    end loop;

    wait;
end process;

end Behavioral;
```

Timing Diagrams

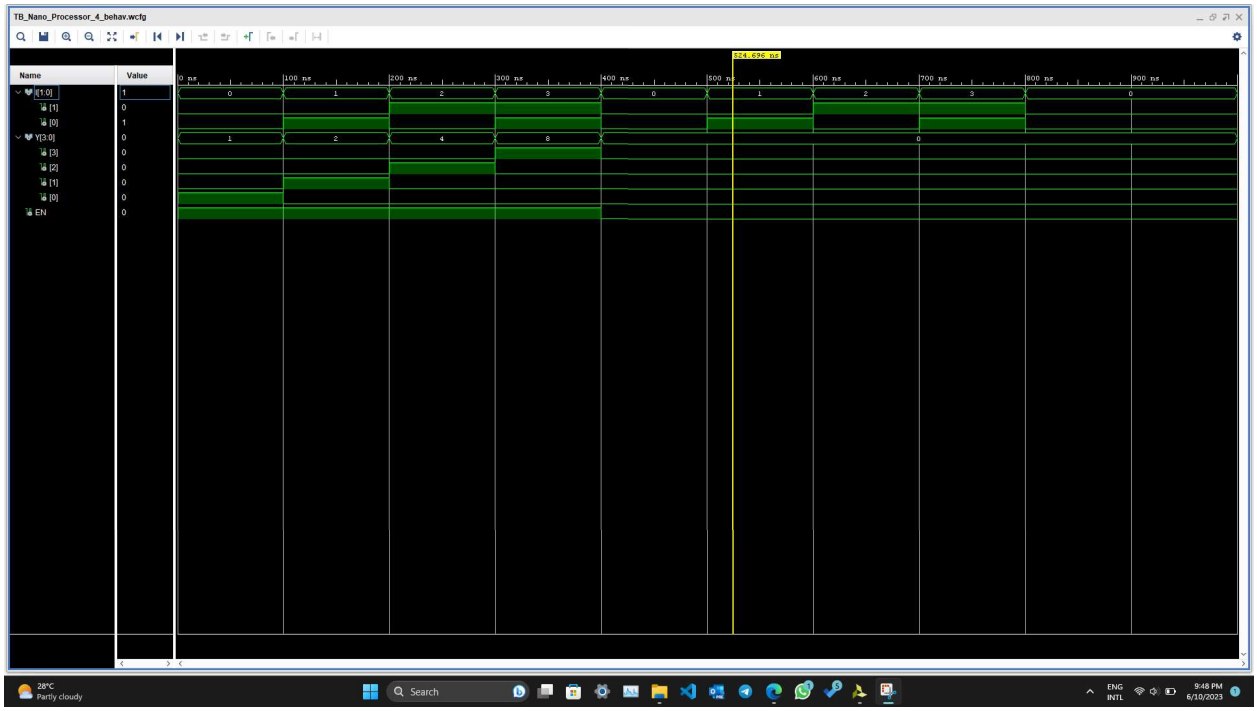
1. D FlipFlop



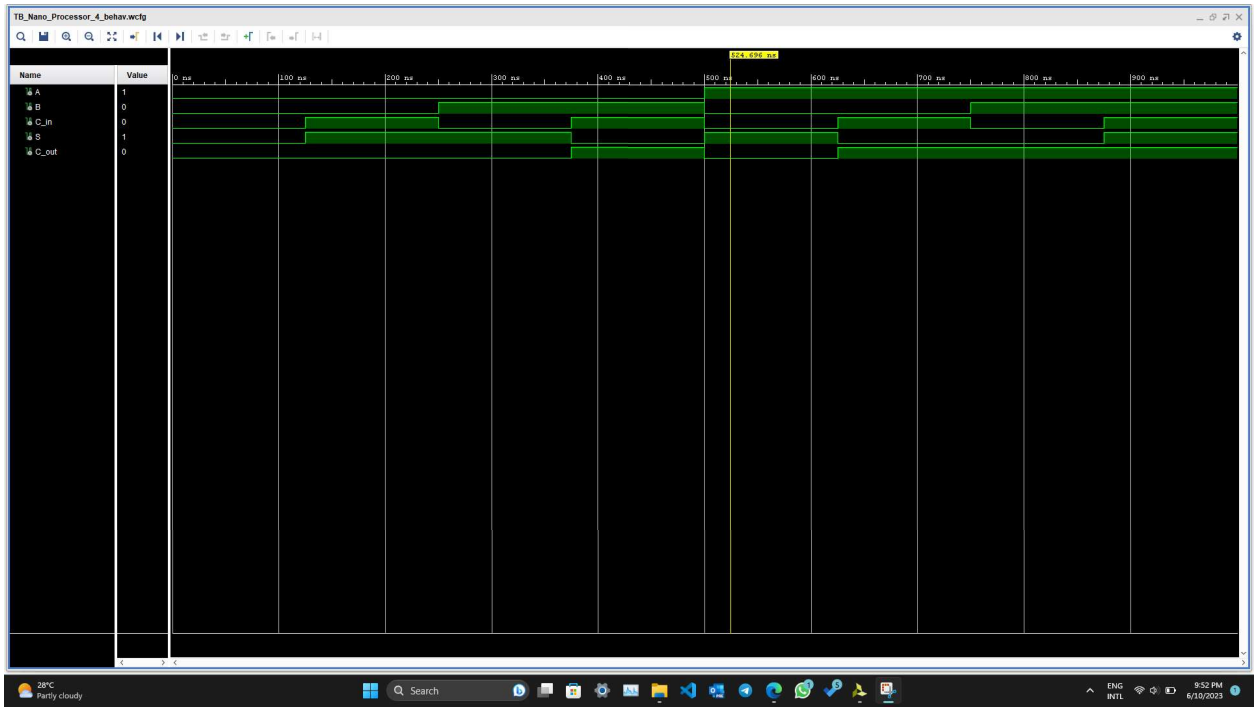
2. 3 to 8 Decoder



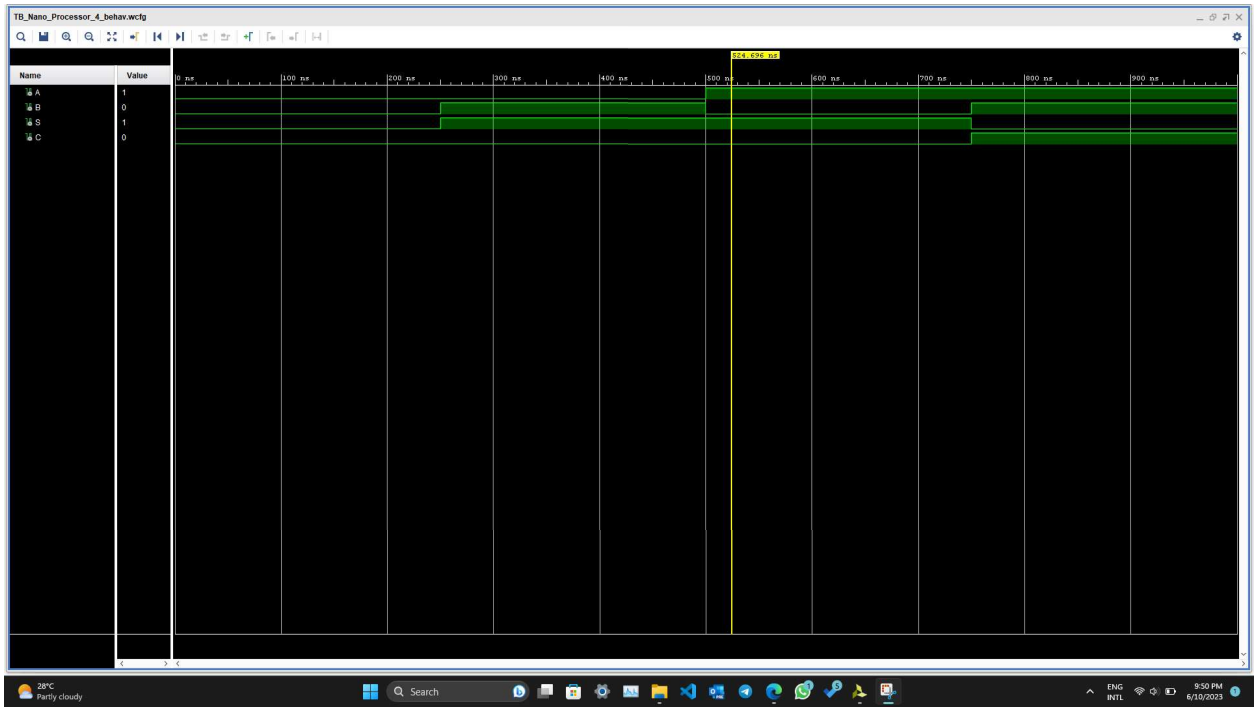
3. 2 to 4 Decoder



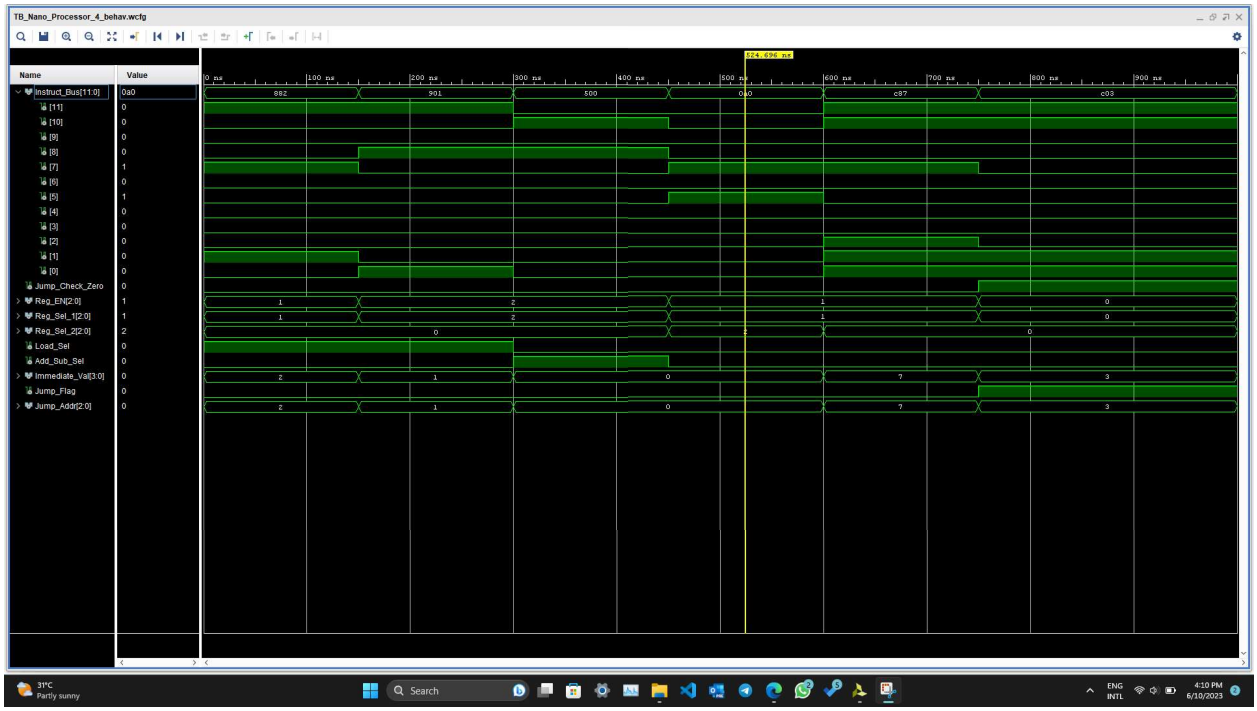
4. Full Adder



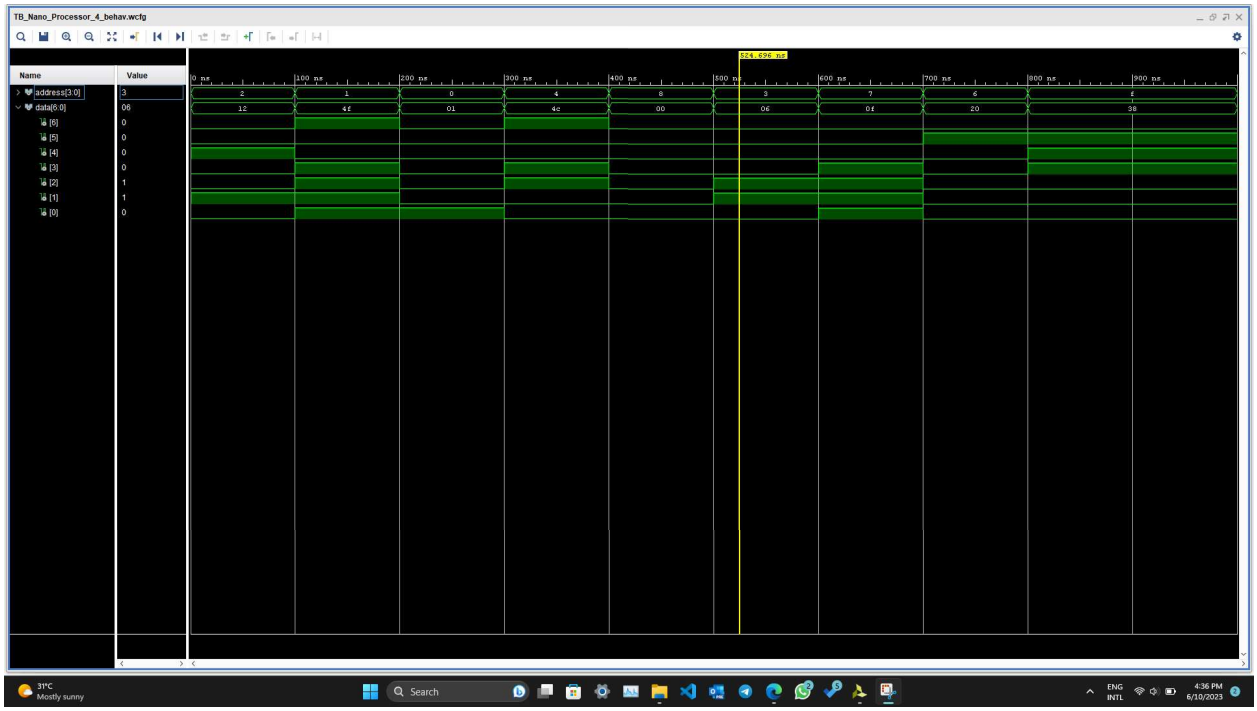
5. Half Adder



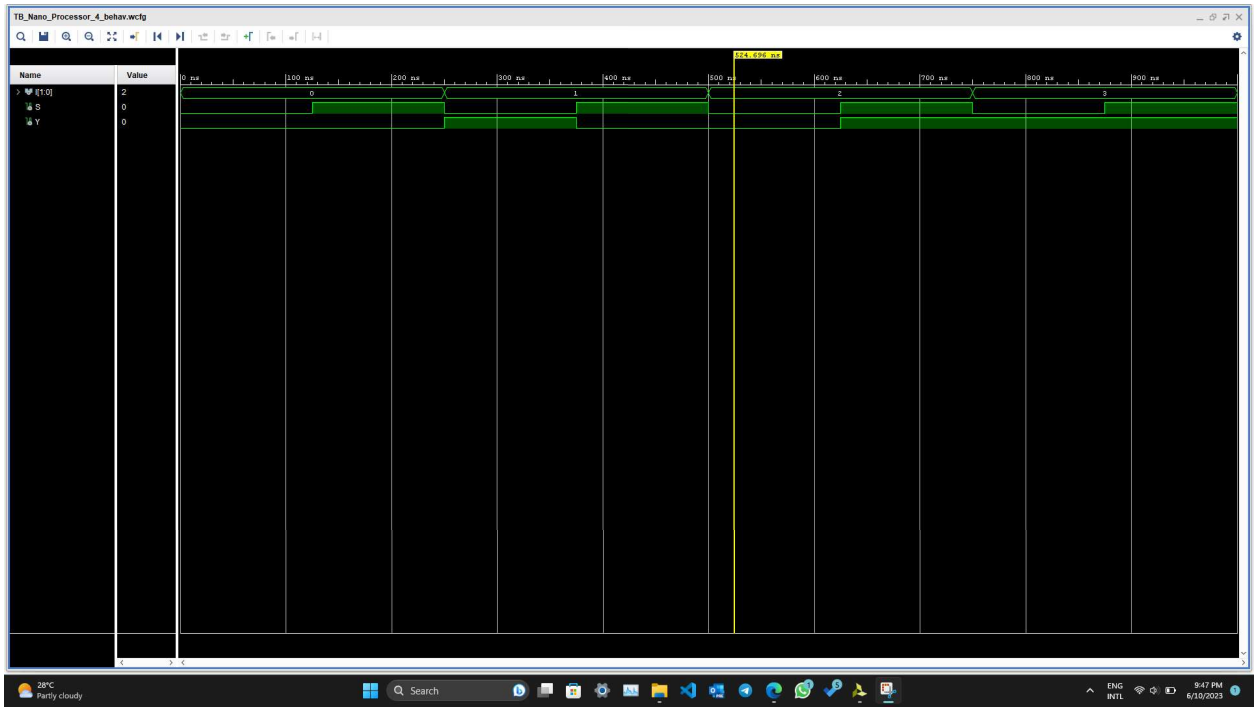
6. Instruction Decoder



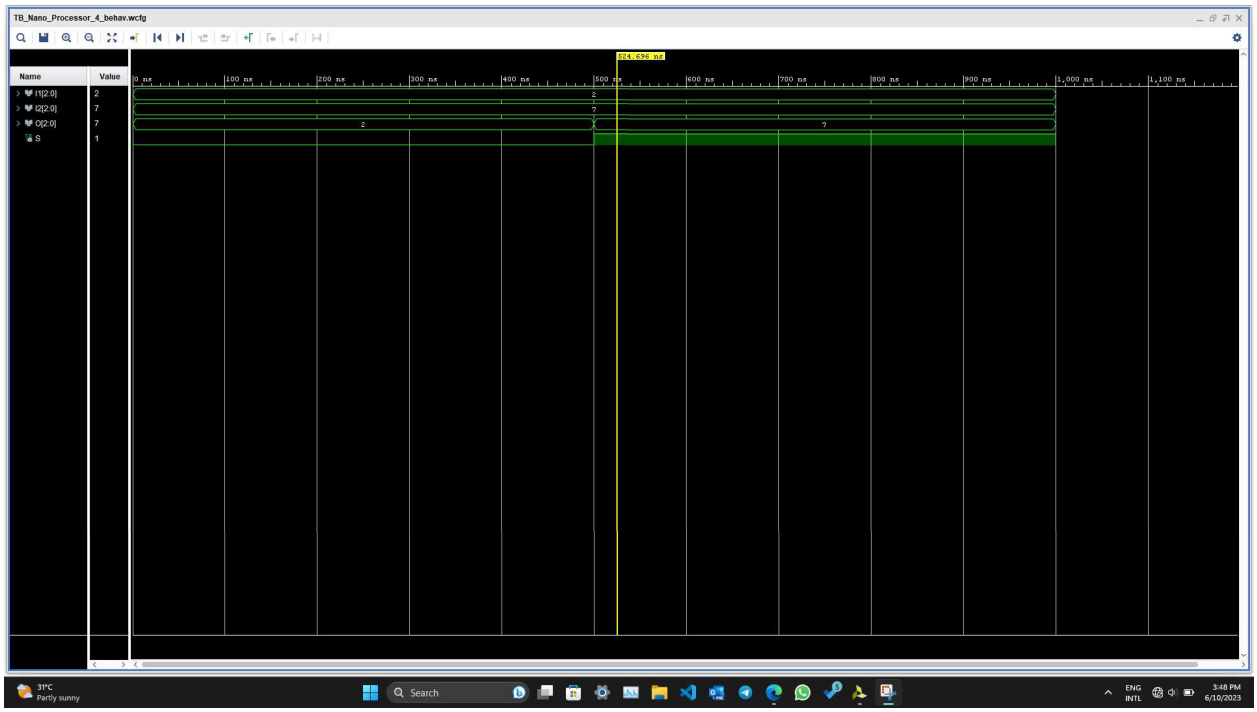
7. Look-Up-Table for Seven Segment Display



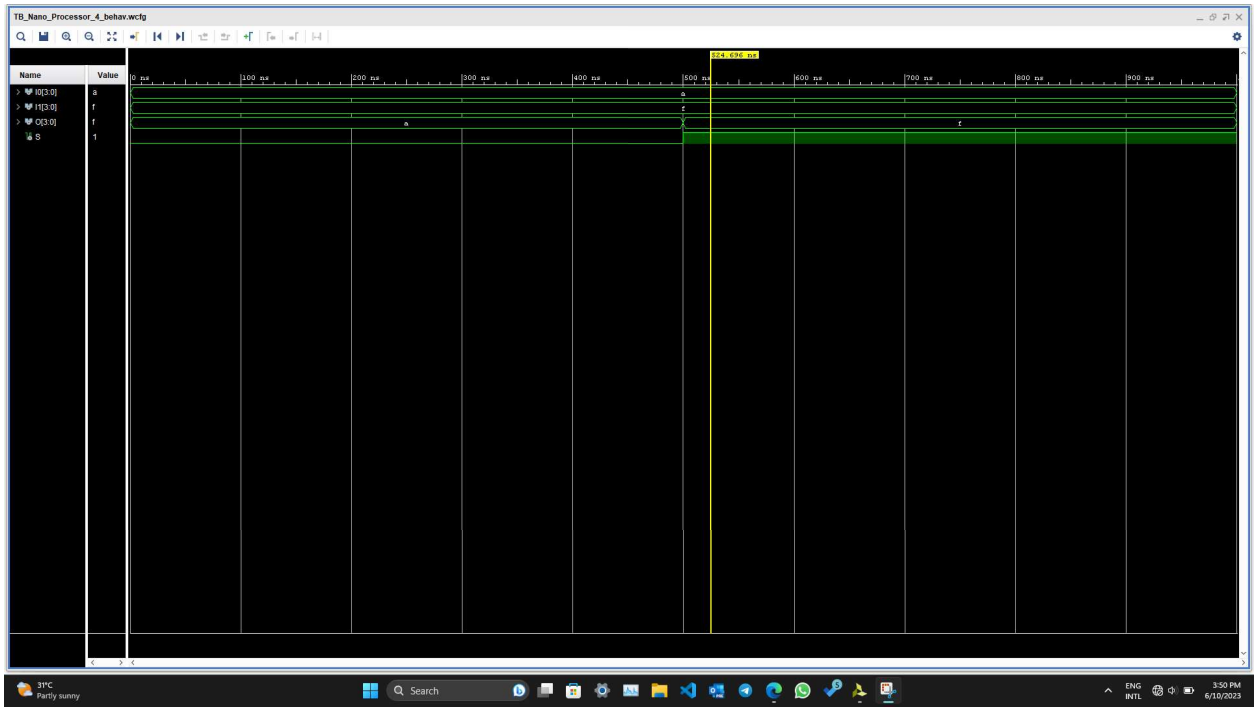
8. 2-to-1 Multiplexer



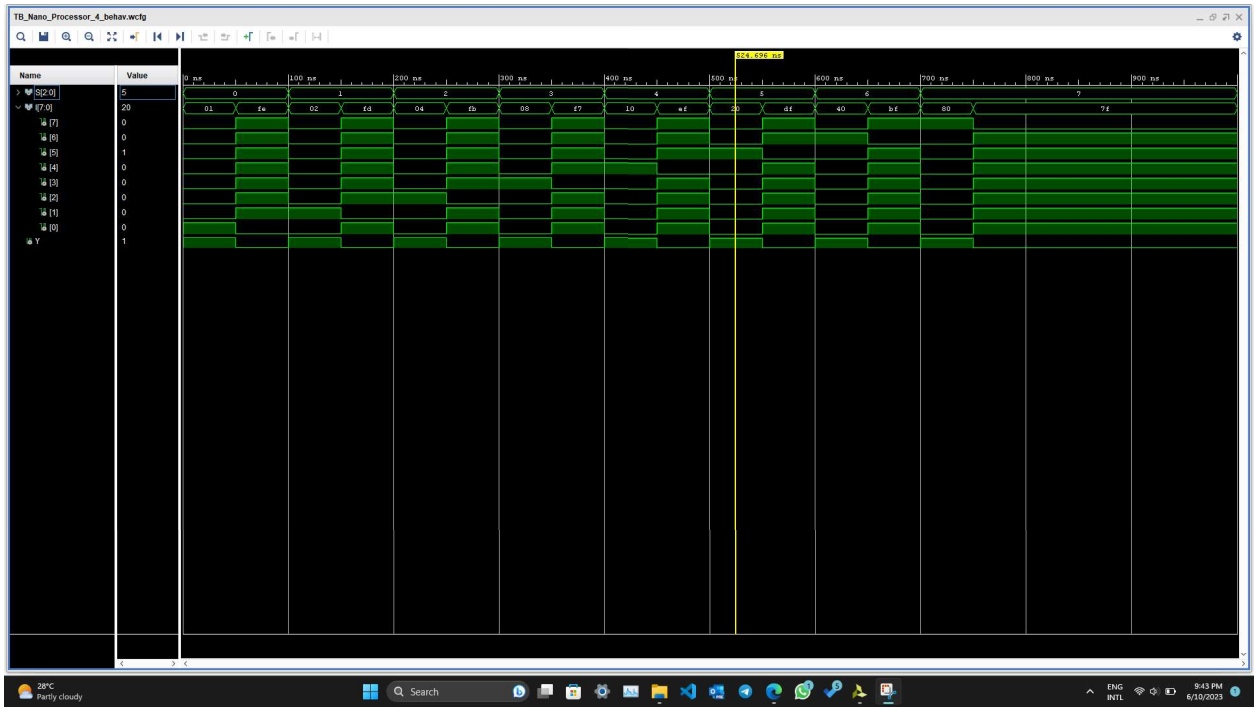
9. 3-bit 2-to-1 Multiplexer



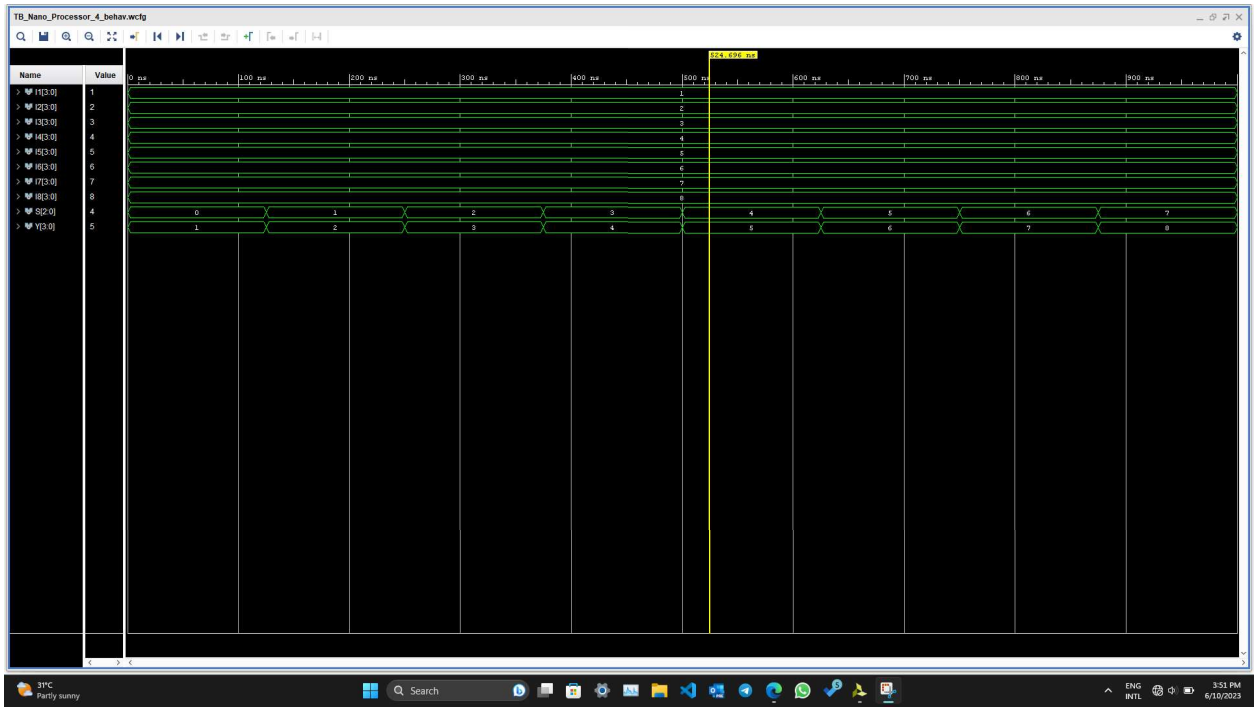
10. 4-bit 2-to-1 Multiplexer



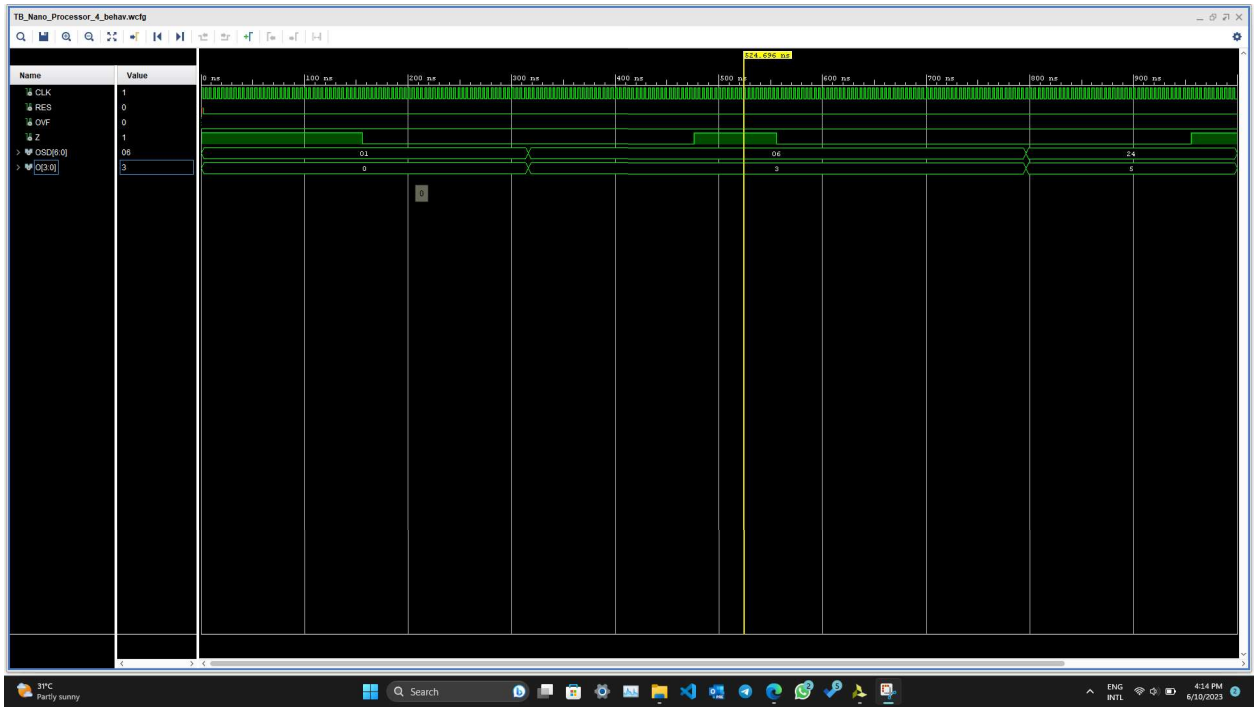
11. 8 to 1 Multiplexer



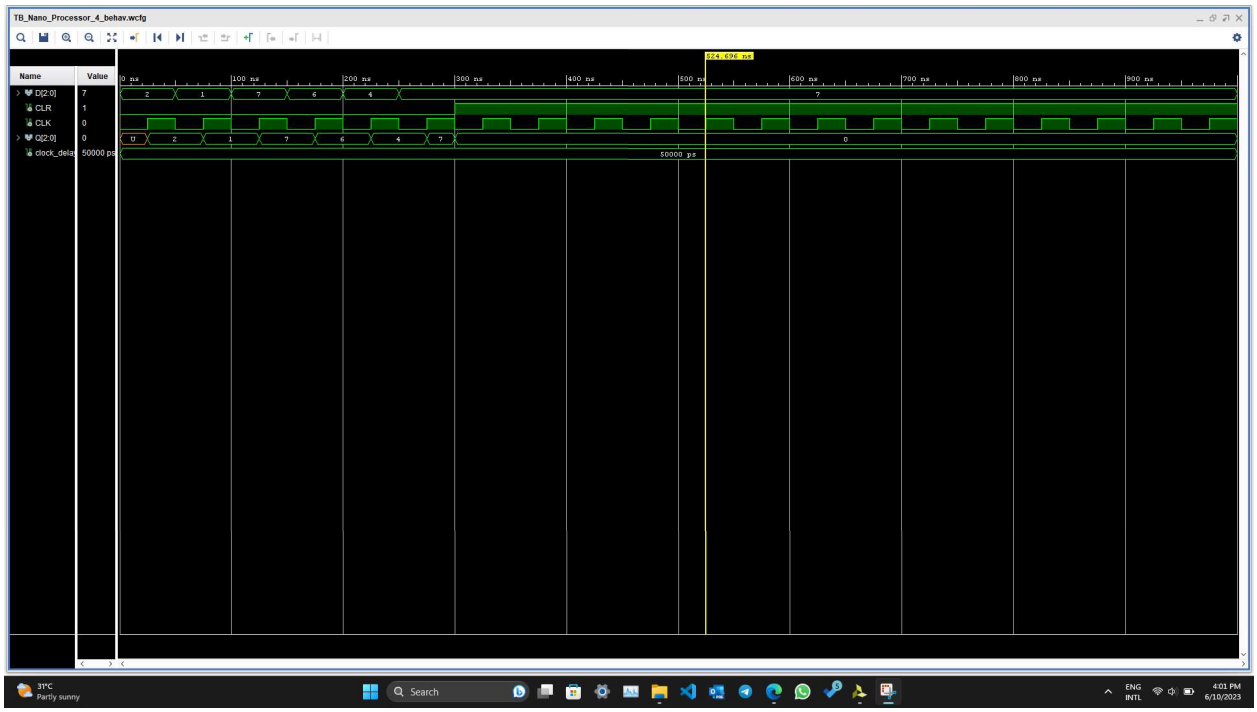
12. 4-bit 8-to-1 Multiplexer



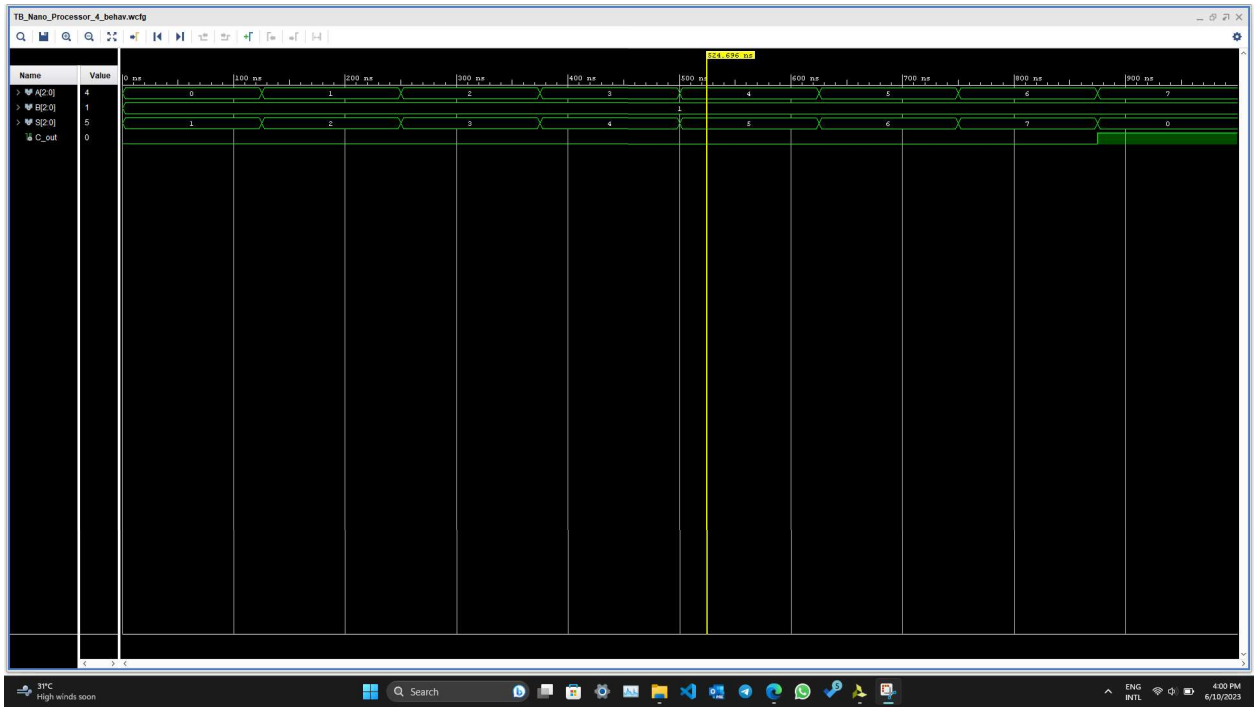
13. Nano Processor



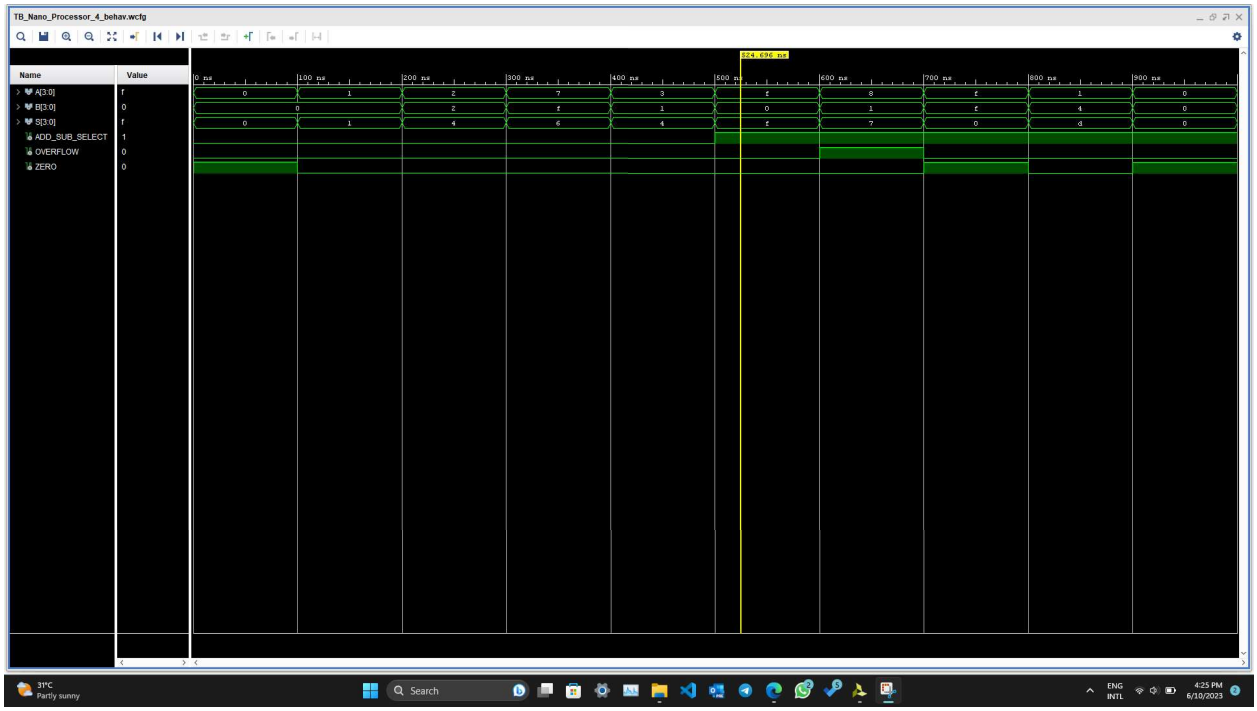
14. 3-bit Program Counter



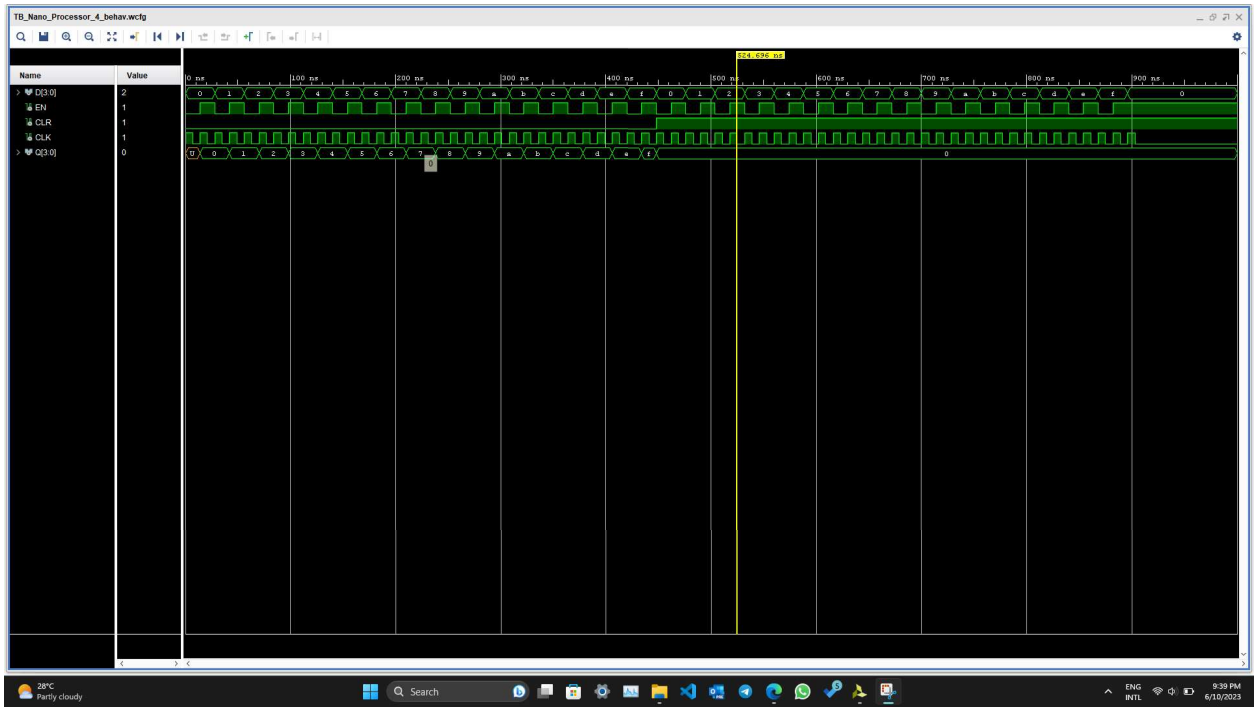
15. 3-bit Ripple Carry Adder



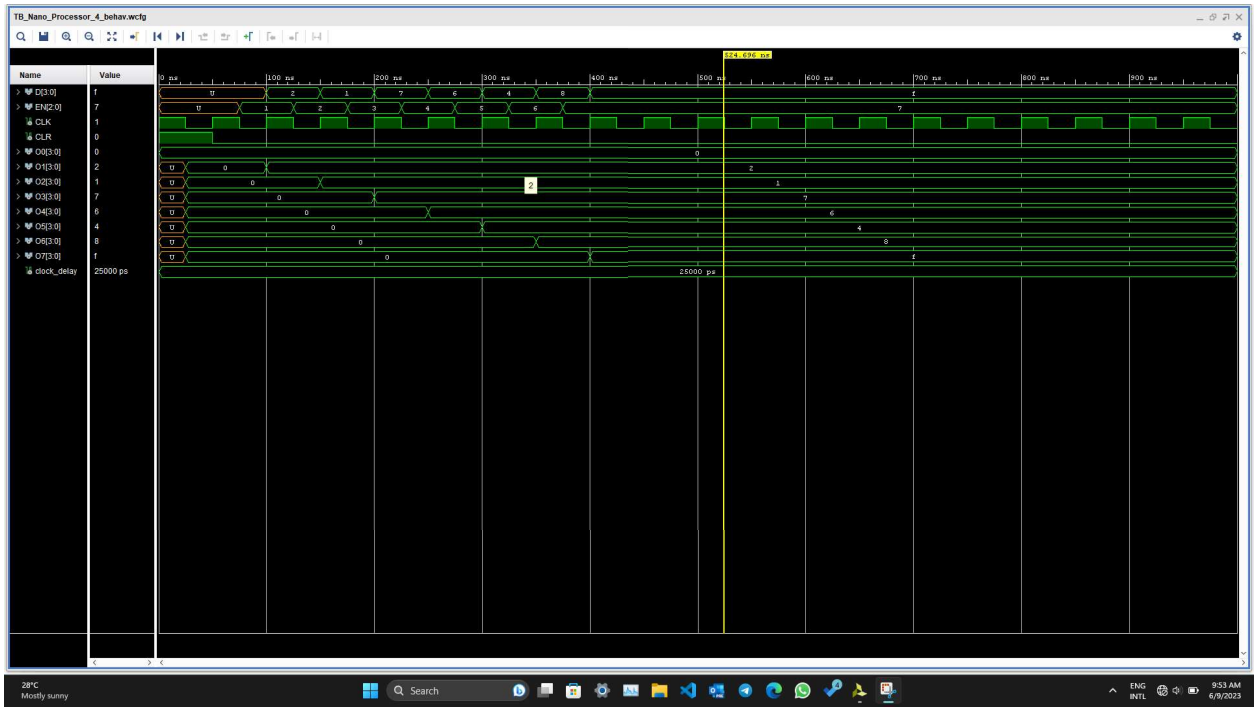
16. 4-bit Adder Subtractor



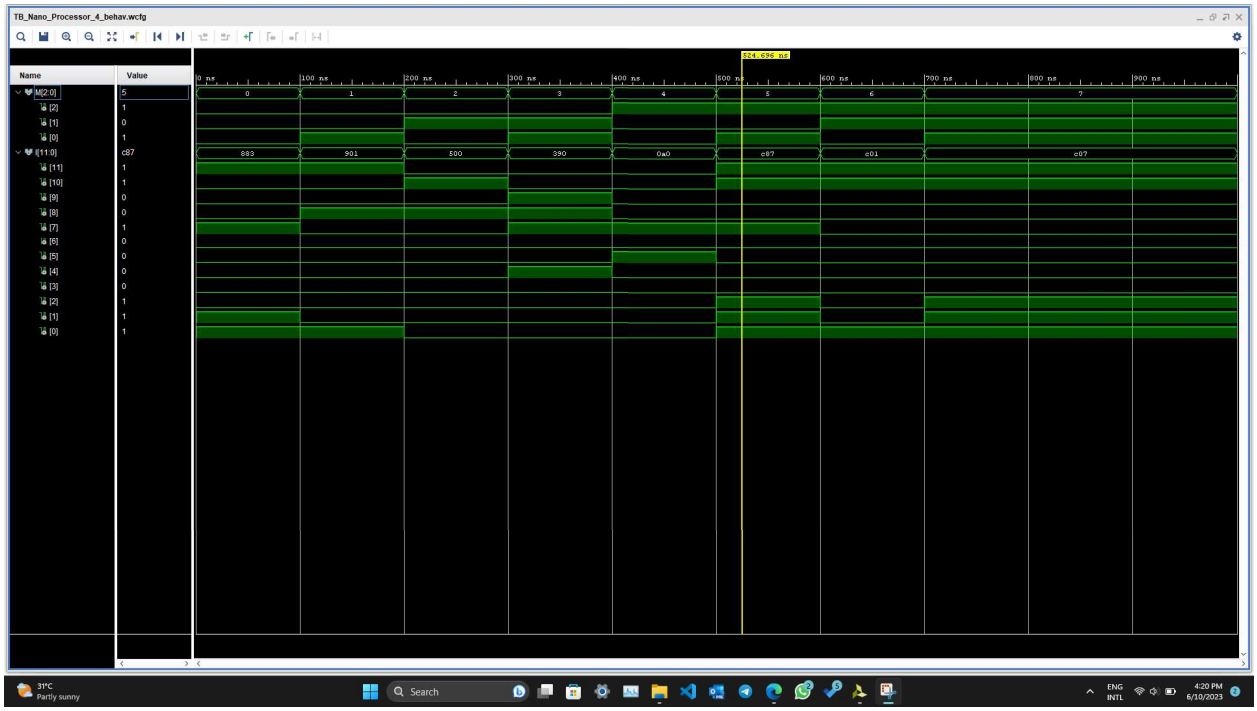
17. 4-bit Register



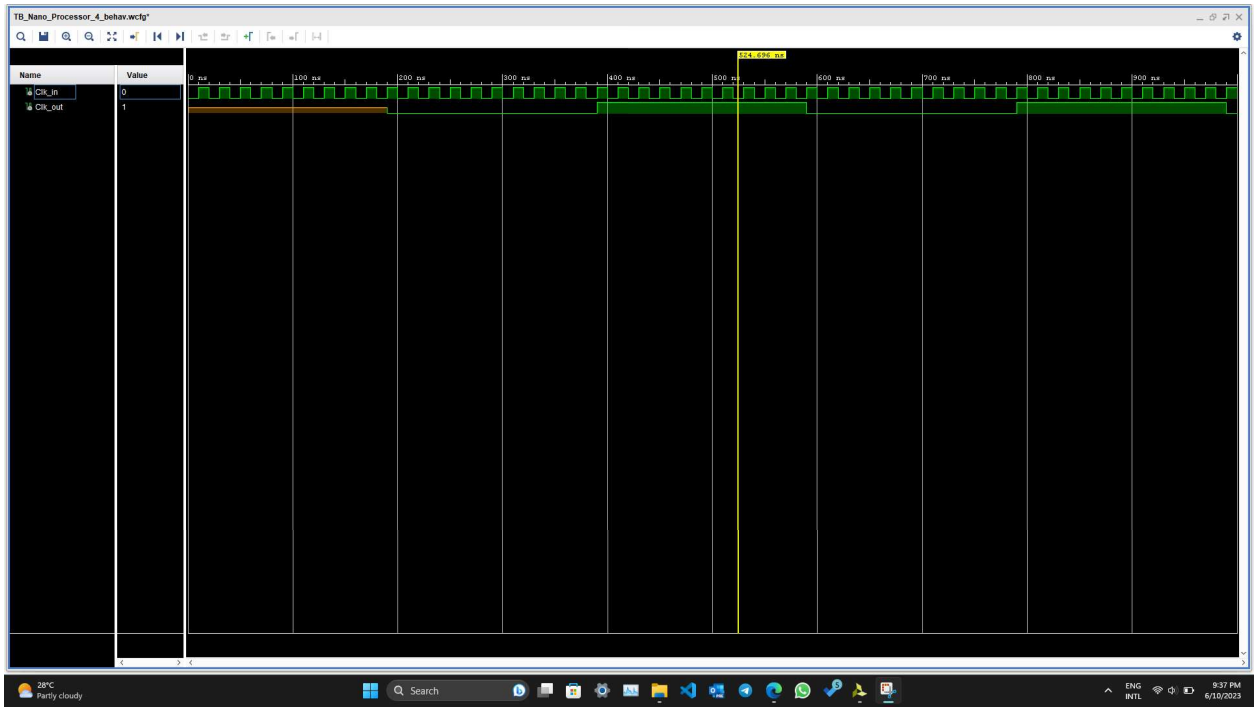
18. Register Bank



19. 8-to-12 ROM



20. Slow Clock



Conclusions from the lab

During the lab, the simulation process indicated that all the designed components performed as expected. However, when implementing the design on the Basys3 board, the results were not as anticipated. This disparity can be attributed to the Real-World Timing Constraints and hardware constraints.