Lab 9-10 Report
Nano Processor Design Competition
CS 1050 – Computer Organization and Digital Design

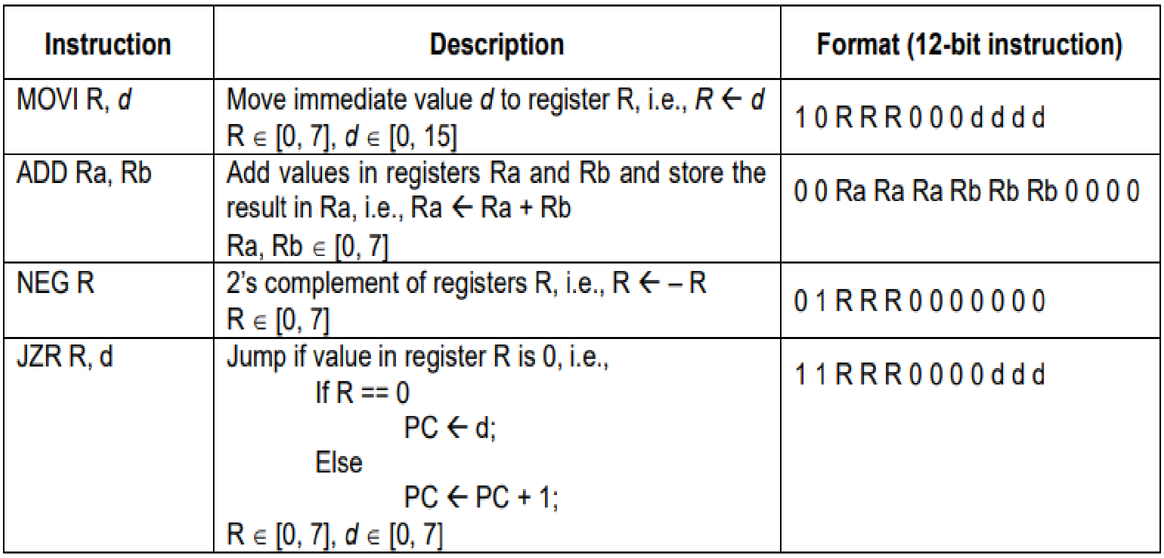
Group Members: Chinthana S.K.G. 200093M
Ekanayake E.M.C.L. 200157
Paramulla P.D.N.T. 200444R
Rupasinghe R.A.M.T. 200546G
Wickramarathna H.K.G.V.L. 200710F

Lab Task

- The main task in this lab is to create a 4-bit Nano-processor that can handle 8 instructions.
- After that write instructions to add/ subtract 4-bit integers.
- Then verify the developed processor using Vivado simulation.
- Finally, test the code on the basys3 board.

Assembly Program

Table 1 – Instruction Set.



Instruction	Description	Format (12-bit instruction)
MOVI R, *d*	Move immediate value *d* to register R, i.e., *R* ← *d* R ∊ [0, 7], *d* ∊ [0, 15]	1 0 R R R 0 0 0 d d d d
ADD Ra, Rb	Add values in registers Ra and Rb and store the result in Ra, i.e., Ra ← Ra + Rb Ra, Rb ∊ [0, 7]	0 0 Ra Ra Ra Rb Rb Rb 0 0 0 0
NEG R	2's complement of registers R, i.e., R ← – R R ∊ [0, 7]	0 1 R R R 0 0 0 0 0 0 0
JZR R, d	Jump if value in register R is 0, i.e., If R == 0 PC ← d; Else PC ← PC + 1; R ∊ [0, 7], *d* ∊ [0, 7]	1 1 R R R 0 0 0 0 d d d

Content

High-level diagram of the Nano Processor

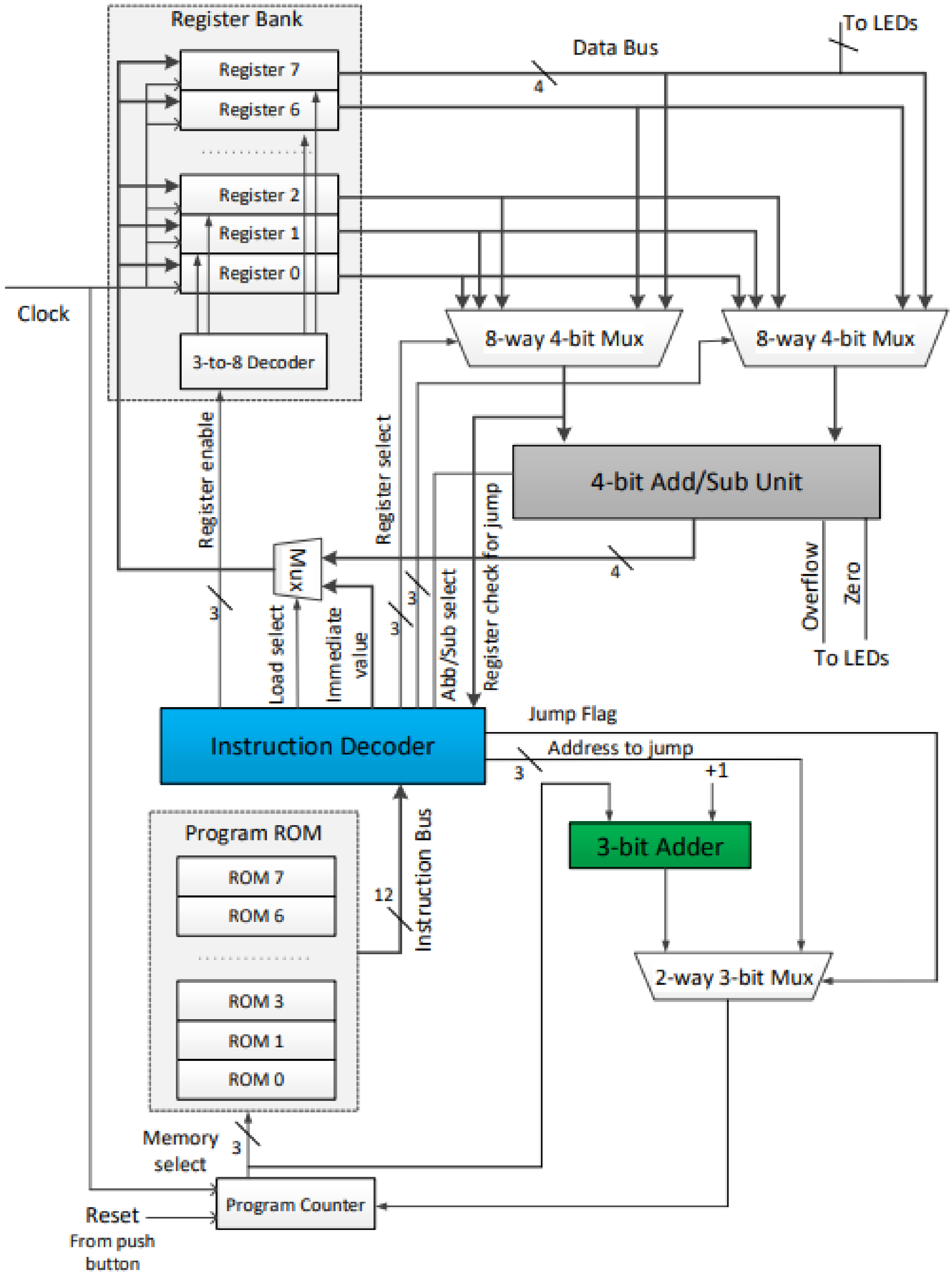


Figure 1 – High-level diagram of the nanoprocessor.

3-bit Program Counter (PC)

VHDL Code

```
----------------------------------------------------------------------------------
-- Company: 
-- Engineer: 
-- 
-- Create Date: 07/10/2022 11:20:45 PM
-- Design Name: 
-- Module Name: Add_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 Add_3 is
    Port ( A : in STD_LOGIC_VECTOR (2 downto 0);
           B : in STD_LOGIC_VECTOR (2 downto 0);
           C_in : in STD_LOGIC;
           S : out STD_LOGIC_VECTOR (2 downto 0);
           C_out : out STD_LOGIC);
end Add_3;
```

```
architecture Behavioral of Add_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;
signal FA0_C,FA1_C,FA2_C: STD_LOGIC;
signal B0,B1,B2: STD_LOGIC_VECTOR (2 downto 0);
begin
 FA_0 : FA
    port map (
        A => '1',
        B => B(0),
        C_in => '0',
        S => S(0),
        C_Out => FA0_C);
 FA_1 : FA
    port map (
         A => '0',
         B => B(1),
         C_in => FA0_C,
         S => S(1),
         C_Out => FA1_C);
 FA_2 : FA
    port map (
         A => '0',
         B => B(2),
         C_in => FA1_C,
         S => S(2),
         C_Out => C_Out);
end Behavioral;
```

Test Bench Code

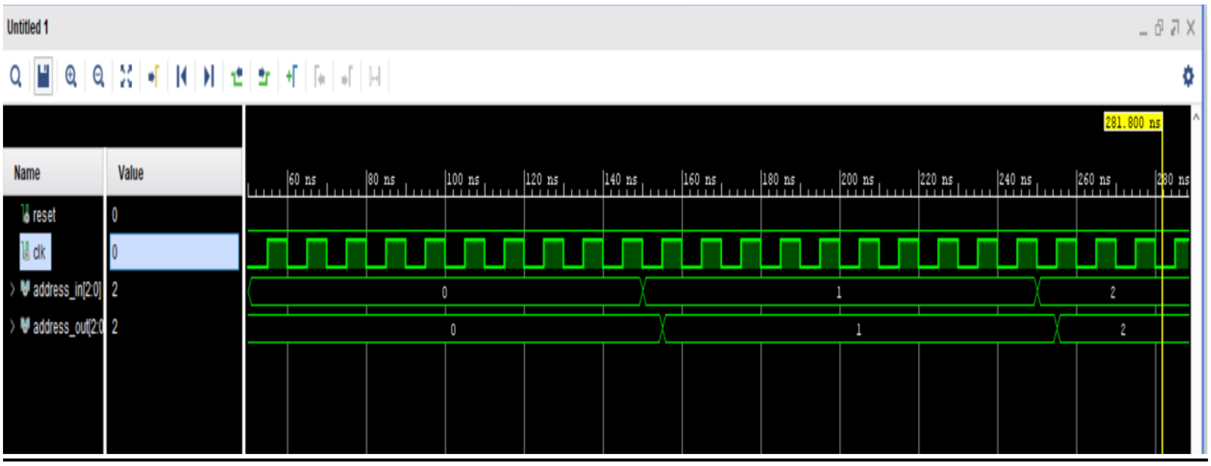
```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 07/27/2022 09:51:07 PM
-- Design Name:
-- Module Name: Program_Counter_0_TB - Behavioral
-- Project Name:
```

```
-- 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 Program_Counter_0_TB is
--  Port ( );
end Program_Counter_0_TB;

architecture Behavioral of Program_Counter_0_TB is

component Program_Counter
   Port ( Pro_Counter_Reset : in STD_LOGIC;
          Clk : in STD_LOGIC;
          Pro_Counter_Address_In : in STD_LOGIC_VECTOR (2 downto 0);
          Pro_Counter_Address_Out : out STD_LOGIC_VECTOR (2 downto 0) := "000");
end component;

signal reset : std_logic;
signal clk : std_logic := '0';
signal address_in : std_logic_vector(2 downto 0);
signal address_out : std_logic_vector(2 downto 0);

begin
```

```
UUT : Program_Counter
   port map (
      Pro_Counter_Reset => reset,
      Clk => clk,
      Pro_Counter_Address_In => address_in,
      Pro_Counter_Address_Out => address_out
   );

process
begin
   wait for 5ns ;
   clk <= NOT clk;
end process;

process
begin

   reset <= '1';
   wait for 50ns;

   reset <= '0';
   address_in <= "000";
   wait for 100ns;

   address_in <= "001";
   wait for 100ns;

   address_in <= "010";
   wait for 100ns;

   address_in <= "011";
   wait for 100ns;

   address_in <= "100";
   wait for 100ns;

   address_in <= "101";
   wait for 100ns;

   address_in <= "110";
   wait for 100ns;

   address_in <= "111";
   wait for 100ns;
```

```
    reset <= '1';
    wait;

end process;
end Behavioral;
```

Time Diagram



3-bit Program Counter Adder

VHDL Code

```
----------------------------------------------------------------------------------
-- Company: 
-- Engineer: 
-- 
-- Create Date: 07/10/2022 11:20:45 PM
-- Design Name: 
-- Module Name: Add_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 Add_3 is
    Port ( Add3_A : in STD_LOGIC_VECTOR (2 downto 0);
           Add3_B : in STD_LOGIC_VECTOR (2 downto 0);
           Add3_C_in : in STD_LOGIC;
           Add3_S : out STD_LOGIC_VECTOR (2 downto 0);
           Add3_C_out : out STD_LOGIC);
end Add_3;
```

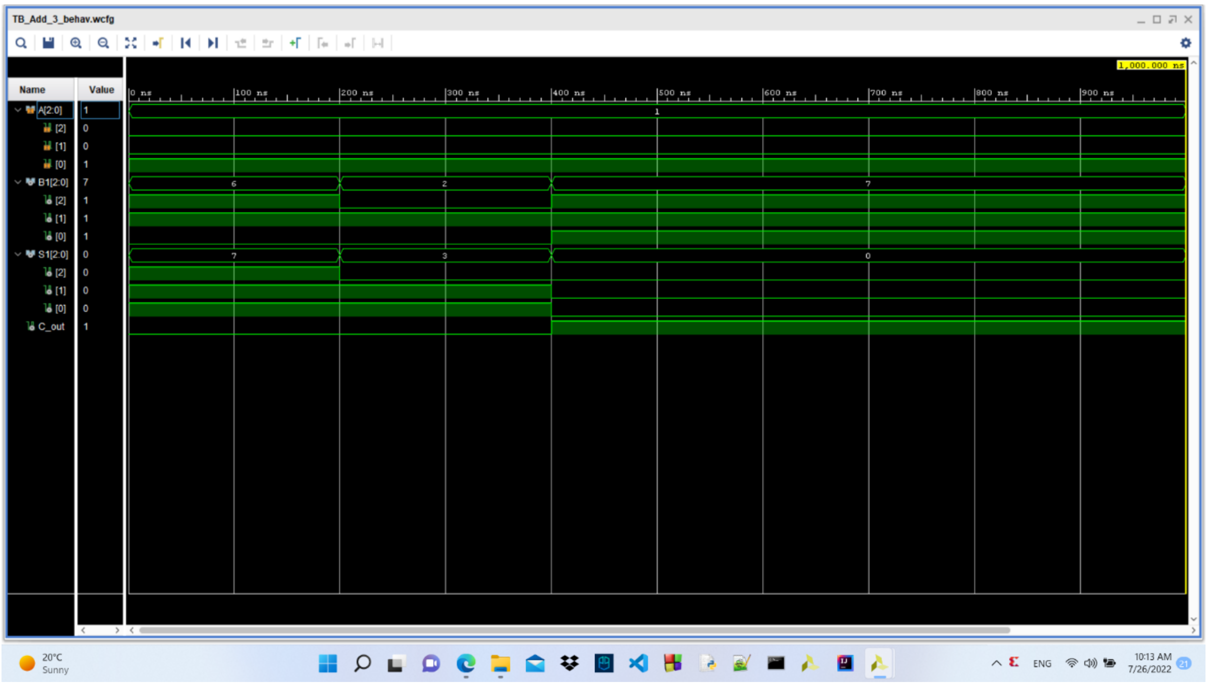
```
architecture Behavioral of Add_3 is
component FA
    port (
        A: in std_logic;
        B: in std_logic;
        C_in: in std_logic;
        S: out std_logic;
        C_out: out std_logic);
end component;
signal FA0_C,FA1_C,FA2_C: STD_LOGIC;
signal B0,B1,B2: STD_LOGIC_VECTOR (2 downto 0);
begin
 FA_0 : FA
   port map (
       A => Add3_A(0),
       B => Add3_B(0),
       C_in => '0',
       S => Add3_S(0),
       C_Out => FA0_C);
 FA_1 : FA
   port map (
        A => Add3_A(1),
        B => Add3_B(1),
        C_in => FA0_C,
        S => Add3_S(1),
        C_Out => FA1_C);
 FA_2 : FA
   port map (
        A => Add3_A(2),
        B => Add3_B(2),
        C_in => FA1_C,
        S => Add3_S(2),
        C_Out => Add3_C_Out);
end Behavioral;
```

Test Bench Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 07/10/2022 11:44:04 PM
-- Design Name:
-- Module Name: TB_Add_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_Add_3 is
--  Port ( );
end TB_Add_3;

architecture Behavioral of TB_Add_3 is
component Add_3
    Port ( Add3_A : in STD_LOGIC_VECTOR (2 downto 0);
           Add3_B : in STD_LOGIC_VECTOR (2 downto 0);
           Add3_C_in : in STD_LOGIC;
           Add3_S : out STD_LOGIC_VECTOR (2 downto 0);
           Add3_C_out : out STD_LOGIC);
end component;
signal B1: STD_LOGIC_VECTOR (2 downto 0);
signal S1 : STD_LOGIC_VECTOR (2 downto 0);
signal C_out :STD_LOGIC;
begin
    UUT: Add_3
        PORT MAP(
            Add3_A => "001",
            Add3_B => B1,
            Add3_C_in => '0',
            Add3_S => S1,
            Add3_C_out => C_out);
          process
            begin
              B1 <= "110";
              wait for 200ns;

              B1 <= "010";
              wait for 200ns;

              B1 <= "111";
              wait;

          end process;

end Behavioral;
```

Time Diagram



TB_Add_3_behav.wcfg

Name	Value
A[2:0]	1
[2]	0
[1]	0
[0]	1
B1[2:0]	7
[2]	1
[1]	1
[0]	1
S1[2:0]	0
[2]	0
[1]	0
[0]	0
C_out	1

Program ROM

VHDL Code

```
----------------------------------------------------------------------------------
-- Company: 
-- Engineer: 
-- 
-- Create Date: 07/10/2022 01:26:48 PM
-- Design Name: 
-- Module Name: ROM - 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 is
    Port ( Mem_Select : in STD_LOGIC_VECTOR (2 downto 0);
           instruction : out STD_LOGIC_VECTOR (11 downto 0));
end ROM;

architecture Behavioral of ROM is

type rom_type is array(0 to 5) of STD_LOGIC_VECTOR(11 downto 0);
```

```
    signal Program_ROM : rom_type := (
        "100010001010", --MOVI R1, 10 ; R1 ? 10
        "100100000001", --MOVI R2, 1 ; R2 ? 1
        "010100000000", --NEG R2 ; R2 ? -R2
        "000010100000", --ADD R1, R2 ; R1 ? R1 + R2
        "110010000111", --JZR R1, 7 ; If R1 = 0 jump to line 7
        "110000000011" --JZR R0, 3 ; If R0 = 0 jump to line 3
        ---need seven
);
begin

instruction <= Program_ROM(to_integer(unsigned(Mem_Select)));
end Behavioral;
```

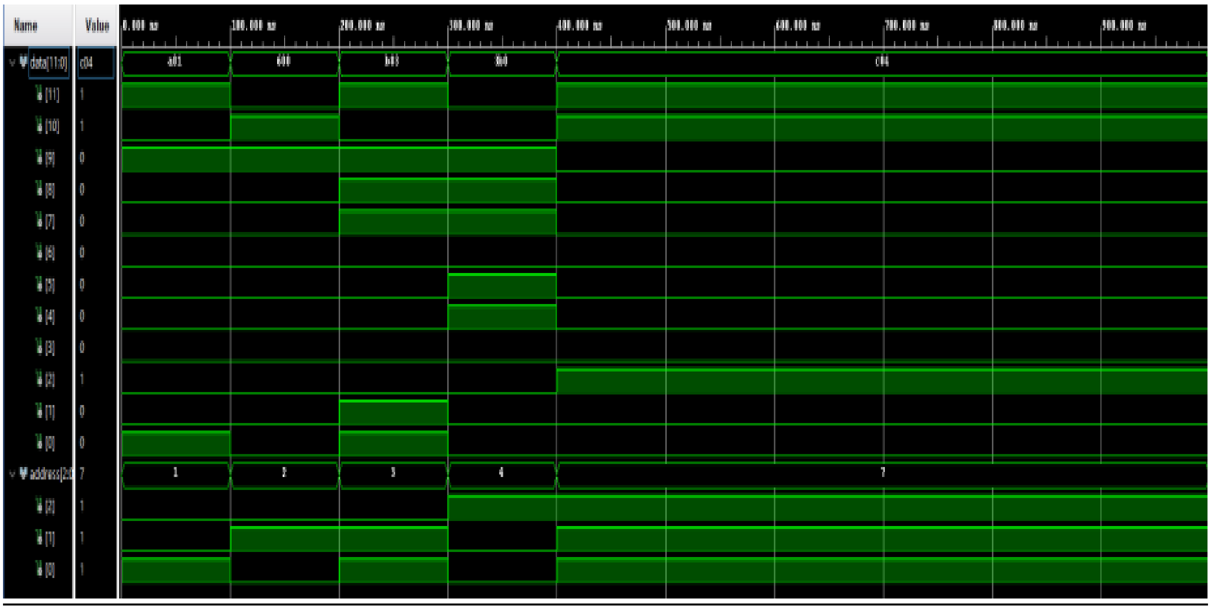
Test Bench Code

```
----------------------------------------------------------------------------------
-- Company: 
-- Engineer: 
-- 
-- Create Date: 07/27/2022 09:05:33 PM
-- Design Name: 
-- Module Name: TB_ROM - 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 is
--  Port ( );
end TB_ROM;

architecture Behavioral of TB_ROM is

component ROM
   Port ( Mem_Select : in STD_LOGIC_VECTOR (2 downto 0);
      instruction : out STD_LOGIC_VECTOR (11 downto 0));
end component;

signal Mem_Select : STD_LOGIC_VECTOR (2 downto 0);
signal instruction : STD_LOGIC_VECTOR (11 downto 0);

begin
UUT: ROM
   PORT MAP(
   Mem_Select => Mem_Select,
   instruction => instruction);

process
begin

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

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

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

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

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

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

```
    Mem_Select <= "110";
    wait for 125ns;

    Mem_Select <= "111";
    wait;

end process;
end Behavioral;
```

Time Diagram



Instruction Decoder

VHDL Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 08/01/2022 11:41:48 PM
-- Design Name:
-- Module Name: inst_decoder_unit - 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 inst_Decoder_unit is
  Port ( inst_bus : in STD_LOGIC_VECTOR (11 downto 0);
        check_jump : in STD_LOGIC_VECTOR (3 downto 0);
        register_en : out STD_LOGIC_VECTOR (2 downto 0);
        register_sel_A : out STD_LOGIC_VECTOR (2 downto 0);
        register_sel_B : out STD_LOGIC_VECTOR (2 downto 0);
        immediate_value : out STD_LOGIC_VECTOR (3 downto 0);
        load_sel : out STD_LOGIC;
        jump_flag : out STD_LOGIC;
        add_sub_sel : out STD_LOGIC;
        Addr_jump : out STD_LOGIC_VECTOR (2 downto 0)
        );
end inst_Decoder_unit;

architecture Behavioral of inst_Decoder_unit is
signal pc : STD_LOGIC ;
begin
   register_en <= inst_bus(9 downto 7);
   register_sel_A <= inst_bus(9 downto 7);
   register_sel_B <= inst_bus(6 downto 4);
   immediate_value <= inst_bus(3 downto 0);
   Addr_jump <= inst_bus(2 downto 0);


   process(check_jump)
   begin
   if(check_jump = "0000") then
      pc <= '1' ;
   else
      pc <= '0' ;
   end if;
   end process;

    load_sel<=(NOT(inst_bus(11))) OR ((inst_bus(11)) AND (inst_bus(10)));
    jump_flag <= pc AND (inst_bus(11)) AND (inst_bus(10));
    add_sub_sel <= (NOT(inst_bus(11))) AND (inst_bus(10));

end Behavioral;
```

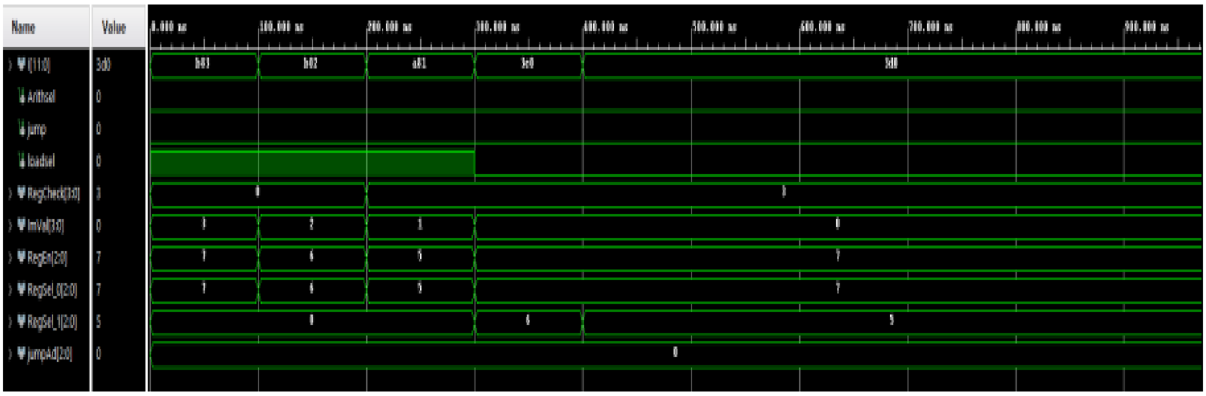
Test Bench Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 08/02/2022 12:39:10 AM
-- Design Name:
-- Module Name: inst_decoder_unit_TB - 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 inst_decoder_unit_TB is
--  Port ( );
end inst_decoder_unit_TB;
architecture Behavioral of inst_decoder_unit_TB is

component inst_decoder_unit
   Port ( inst_bus : in STD_LOGIC_VECTOR (11 downto 0);
          check_jump : in STD_LOGIC_VECTOR (3 downto 0);
          register_en : out STD_LOGIC_VECTOR (2 downto 0);
          register_sel_A : out STD_LOGIC_VECTOR (2 downto 0);
          register_sel_B : out STD_LOGIC_VECTOR (2 downto 0);
          immediate_value : out STD_LOGIC_VECTOR (3 downto 0);
          load_sel : out STD_LOGIC;
          jump_flag : out STD_LOGIC;
          add_sub_sel : out STD_LOGIC;
          Addr_jump : out STD_LOGIC_VECTOR (2 downto 0)
          );
end component;

signal inst_bus : STD_LOGIC_VECTOR (11 downto 0);
signal register_en,register_sel_A,register_sel_B,Addr_jump  :STD_LOGIC_VECTOR (2 downto 0);
signal immediate_value,check_jump  : STD_LOGIC_VECTOR (3 downto 0);
signal load_sel,jump_flag,add_sub_sel : STD_LOGIC;

begin
```

```
UUT:inst_decoder_unit
   PORT MAP(
        inst_bus => inst_bus,
        check_jump => check_jump,
        register_en => register_en,
        register_sel_A => register_sel_A,
        register_sel_B => register_sel_B,
        immediate_value => immediate_value,
        load_sel => load_sel,
        jump_flag => jump_flag,
        add_sub_sel => add_sub_sel,
        Addr_jump => Addr_jump
        );
process
begin
   inst_bus <= "100110000101";
   wait for 200ns;

   inst_bus <= "000110010000";
    wait for 200ns;

    inst_bus <= "011010000000";
     wait for 200ns;

     inst_bus <= "111110000010";
     wait;
end process;

end Behavioral;
```

Time Diagram



Register Bank

VHDL Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 07/10/2022 03:24:01 PM
-- 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);
    CLK : in STD_LOGIC;
    I : in STD_LOGIC_VECTOR (2 downto 0);
    CLR : in STD_LOGIC;
    R0 : out STD_LOGIC_VECTOR (3 downto 0);
    R1 : out STD_LOGIC_VECTOR (3 downto 0);
```

```
      R2 : out STD_LOGIC_VECTOR (3 downto 0);
      R3 : out STD_LOGIC_VECTOR (3 downto 0);
      R4 : out STD_LOGIC_VECTOR (3 downto 0);
      R5 : out STD_LOGIC_VECTOR (3 downto 0);
      R6 : out STD_LOGIC_VECTOR (3 downto 0);
      R7 : out STD_LOGIC_VECTOR (3 downto 0)
    );
end Reg_Bank;

architecture Behavioral of Reg_Bank is

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

component Reg
   Port ( D : in STD_LOGIC_VECTOR (3 downto 0);
        En : in STD_LOGIC;
        Reset : in STD_LOGIC;
        Clk : in STD_LOGIC;
        Q : out STD_LOGIC_VECTOR (3 downto 0));
end component;

signal Reg_enable : STD_LOGIC_VECTOR(7 downto 0);

begin
Decoder_3_to_8_0: Decoder_3_to_8
   PORT MAP(
      I => I,
      EN => '1',
      Y => Reg_enable);

--Hardcoding R0
Reg_0: Reg
   PORT MAP(
      D => "0000",
      En => Reg_enable(0),
      Reset => CLR,
      CLK => CLK,
      Q => R0);

Reg_1: Reg
   PORT MAP(
      D => D,
      En => Reg_enable(1),
      Reset => CLR,
      CLK => CLK,
      Q => R1);
Reg_2: Reg
    PORT MAP(
      D => D,
      En => Reg_enable(2),
      Reset => CLR,
      CLK => CLK,
      Q => R2);

Reg_3: Reg
    PORT MAP(
      D => D,
      En => Reg_enable(3),
      Reset => CLR,
      CLK => CLK,
      Q => R3);

Reg_4: Reg
      PORT MAP(
      D => D,
      En => Reg_enable(4),
      Reset => CLR,
      CLK => CLK,
      Q => R4);

Reg_5: Reg
      PORT MAP(
      D => D,
      En => Reg_enable(5),
      Reset => CLR,
      CLK => CLK,
      Q => R5);

Reg_6: Reg
      PORT MAP(
      D => D,
      En => Reg_enable(6),
      Reset => CLR,
      CLK => CLK,
      Q => R6);
```

```
Reg_7: Reg
    PORT MAP(
      D => D,
      En => Reg_enable(7),
      Reset => CLR,
      CLK => CLK,
      Q => R7);
end Behavioral;
```

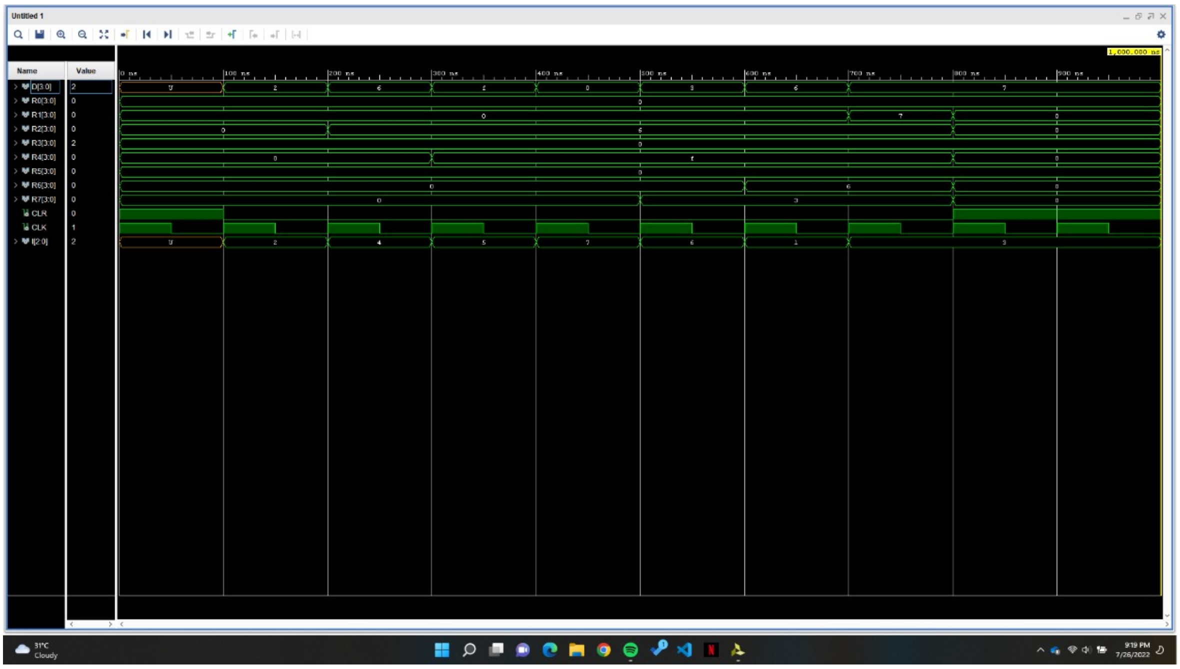
Test Bench Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 07/12/2022 11:30:53 AM
-- 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);
      CLK : in STD_LOGIC;
      I : in STD_LOGIC_VECTOR (2 downto 0);
      CLR : in STD_LOGIC;
      R0 : out STD_LOGIC_VECTOR (3 downto 0);
      R1 : out STD_LOGIC_VECTOR (3 downto 0);
      R2 : out STD_LOGIC_VECTOR (3 downto 0);
      R3 : out STD_LOGIC_VECTOR (3 downto 0);
      R4 : out STD_LOGIC_VECTOR (3 downto 0);
      R5 : out STD_LOGIC_VECTOR (3 downto 0);
      R6 : out STD_LOGIC_VECTOR (3 downto 0);
      R7 : out STD_LOGIC_VECTOR (3 downto 0)
   );
end component;

signal D,R0,R1,R2,R3,R4,R5,R6,R7: STD_LOGIC_VECTOR (3 downto 0);
signal CLR : STD_LOGIC := '0';
signal CLK : STD_LOGIC := '0';
signal I : STD_LOGIC_VECTOR (2 downto 0);

begin
UUT: Reg_Bank
   PORT MAP(
      D => D,
      I => I,
      CLK => CLK,
      CLR => CLR,
      R0 => R0,
      R1 => R1,
      R2 => R2,
      R3 => R3,
      R4 => R4,
      R5 => R5,
      R6 => R6,
      R7 => R7);

 process
   begin
```

```
        CLK <= not CLK;
        wait for 50ns;
 end process;

process
   begin
        --200546G
        --0011 0000 1111 0110 0010
        --000 110 000 111 101 100 010
        CLR <= '1';
        wait for 100ns;

        CLR <= '0';
        D <= "0010";
        I <= "010";
        wait for 100ns;

        CLR <= '0';
        D <= "0110";
        I <= "100";
        wait for 100ns;

        CLR <= '0';
        D <= "1111";
        I <= "101";
        wait for 100ns;

        CLR <= '0';
        D <= "0000";
        I <= "111";
        wait for 100ns;

        CLR <= '0';
        D <= "0011";
        I <= "110";
        wait for 100ns;

        CLR <= '0';
        D <= "0110";
        I <= "001";
        wait for 100ns;

        CLR <= '0';
        D <= "0111";
        I <= "011";
        wait for 100ns;

        CLR <= '1';
        wait for 100ns;
end process;
end Behavioral;
```

Time Diagram



k-way b–bit Multiplexers

2-way 3-bit Multiplexer

VHDL Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 07/11/2022 12:30:10 PM
-- Design Name:
-- Module Name: mux_2_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 mux_2_3_x is
   Port ( mux_2_3_A : in STD_LOGIC_VECTOR (2 downto 0);
          mux_2_3_B : in STD_LOGIC_VECTOR (2 downto 0);
          mux_2_3_C0 : in STD_LOGIC;
          mux_2_3_Q : out STD_LOGIC_VECTOR (2 downto 0));
end mux_2_3_x;
```

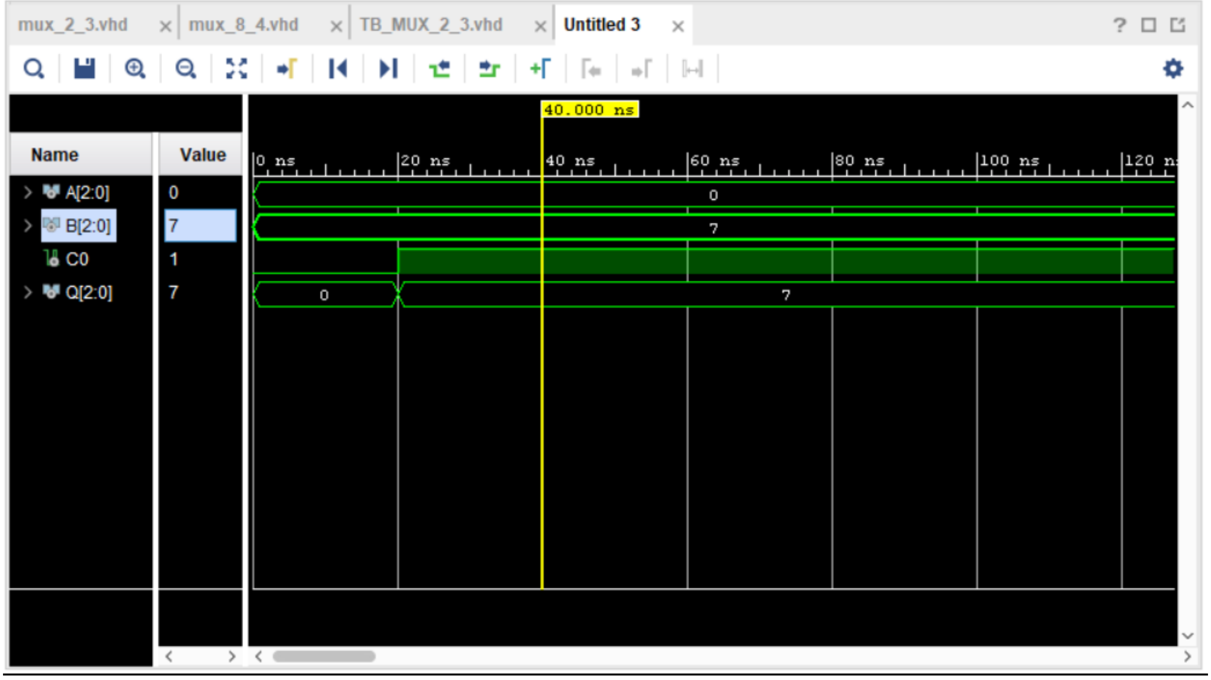
```
architecture Behavioral of mux_2_3_x is

component mux_2_1_x
   port(A0 : in STD_LOGIC;
        A1 : in STD_LOGIC;
        S : in STD_LOGIC;
        Y : out STD_LOGIC);
end component;

begin
   mux_2_1_0: mux_2_1_x
   port map(
   A0 => mux_2_3_A(0),
   A1 => mux_2_3_B(0),
   S => mux_2_3_C0,
   Y =>mux_2_3_Q(0));

   mux_2_1_1: mux_2_1_x
   port map(
   A0 => mux_2_3_A(1),
   A1 => mux_2_3_B(1),
   S => mux_2_3_C0,
   Y => mux_2_3_Q(1));

   mux_2_1_2: mux_2_1_x
   port map(
   A0 => mux_2_3_A(2),
   A1 => mux_2_3_B(2),
   S => mux_2_3_C0,
   Y => mux_2_3_Q(2));

end Behavioral;
```

Test Bench Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 07/26/2022 09:52:22 AM
-- Design Name:
-- Module Name: TB_MUX_2_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_MUX_2_3 is
--  Port ( );
end TB_MUX_2_3;

architecture Behavioral of TB_MUX_2_3 is
component mux_2_3
port(mux_2_3_A : in STD_LOGIC_VECTOR (2 downto 0);
    mux_2_3_B : in STD_LOGIC_VECTOR (2 downto 0);
    mux_2_3_C0 : in STD_LOGIC;
    mux_2_3_Q0 : out STD_LOGIC_VECTOR (2 downto 0));
end component;

signal A,B : STD_LOGIC_VECTOR (2 downto 0);
signal C0 : STD_LOGIC;
signal Q0 : STD_LOGIC_VECTOR (2 downto 0);

begin
UUT: mux_2_3
PORT MAP(
mux_2_3_A => A,
mux_2_3_B => B,
mux_2_3_C0 => C0,
mux_2_3_Q0 => Q0);

process
   begin
   A <= "000";
   B <= "111";
   C0 <= '0';
   wait for 20ns;
   C0 <= '1';
   wait;
end process;
end Behavioral;
```

Time Diagram



Name	Value
A[2:0]	0
B[2:0]	7
C0	1
Q[2:0]	7

2-way 4-bit Multiplexer

VHDL Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 07/11/2022 12:41:19 PM
-- Design Name:
-- Module Name: mux_2_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 mux_2_4 is
    Port ( mux_2_4_A : in STD_LOGIC_VECTOR (3 downto 0);
           mux_2_4_B : in STD_LOGIC_VECTOR (3 downto 0);
           mux_2_4_C0 : in STD_LOGIC;
           mux_2_4_Q : out STD_LOGIC_VECTOR (3 downto 0));
end mux_2_4;

architecture Behavioral of mux_2_4 is
```

```
component mux_2_1_x
port(A0 : in STD_LOGIC;
     A1 : in STD_LOGIC;
     S : in STD_LOGIC;
     Y : out STD_LOGIC);
end component;

begin

mux_2_1_0: mux_2_1_x
   port map(
   A0 => mux_2_4_A(0),
   A1 => mux_2_4_B(0),
   S => mux_2_4_C0,
   Y => mux_2_4_Q(0));

mux_2_1_1: mux_2_1_x
   port map(
   A0 => mux_2_4_A(1),
   A1 => mux_2_4_B(1),
   S => mux_2_4_C0,
   Y => mux_2_4_Q(1));

mux_2_1_2: mux_2_1_x
   port map(
   A0 => mux_2_4_A(2),
   A1 => mux_2_4_B(2),
   S => mux_2_4_C0,
   Y => mux_2_4_Q(2));

mux_2_1_3: mux_2_1_x
   port map(
   A0 => mux_2_4_A(3),
   A1 => mux_2_4_B(3),
   S => mux_2_4_C0,
   Y => mux_2_4_Q(3));

end Behavioral;
```

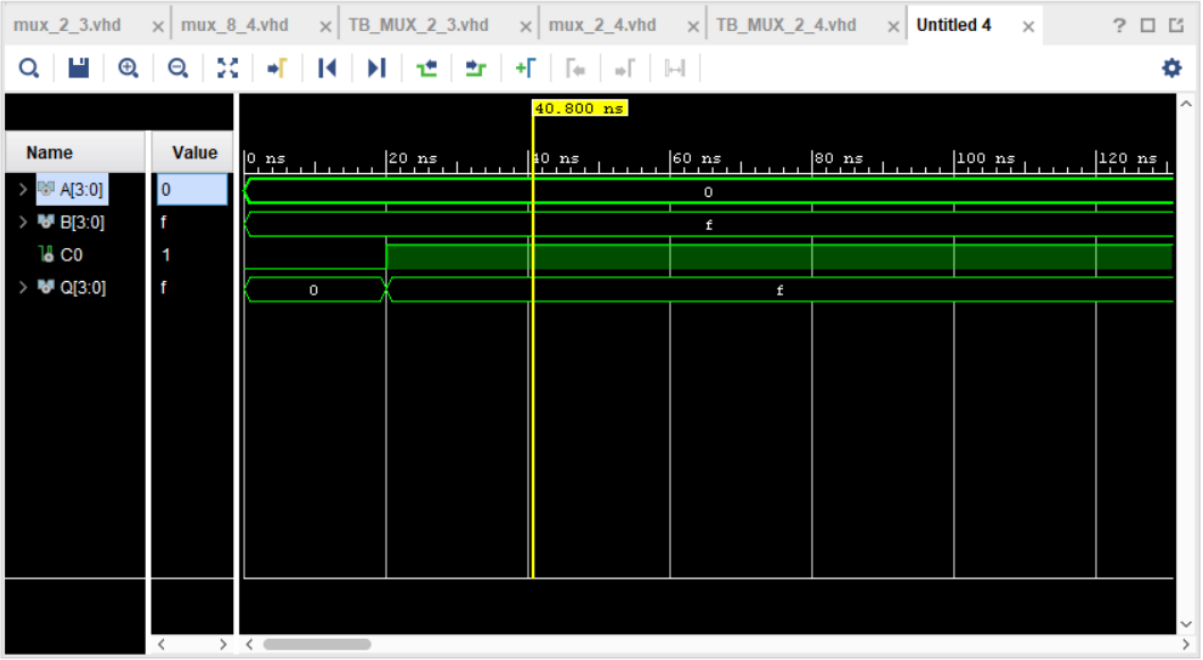
Test Bench Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
```

```
-- Create Date: 07/26/2022 10:03:30 AM
-- Design Name:
-- Module Name: TB_MUX_2_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_MUX_2_4 is
--  Port ( );
end TB_MUX_2_4;

architecture Behavioral of TB_MUX_2_4 is
component mux_2_4
port(mux_2_4_A : in STD_LOGIC_VECTOR (3 downto 0);
    mux_2_4_B : in STD_LOGIC_VECTOR (3 downto 0);
    mux_2_4_C0 : in STD_LOGIC;
    mux_2_4_Q : out STD_LOGIC_VECTOR (3 downto 0));
end component;

signal A,B : STD_LOGIC_VECTOR (3 downto 0);
signal C0: STD_LOGIC;
signal Q : STD_LOGIC_VECTOR (3 downto 0);
```

```
begin
UUT : mux_2_4
   port map(
   mux_2_4_A =>A,
   mux_2_4_B=> B,
   mux_2_4_C0 =>C0,
   mux_2_4_Q => Q);

process
begin
A <= "0000";
B <= "1111";
C0 <= '0';
wait for 20ns;
C0 <= '1';
wait;
end process;
end Behavioral;
```

Time Diagram



8-way 4-bit Multiplexer

VHDL Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 07/11/2022 03:45:53 PM
-- Design Name:
-- Module Name: mux_8_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 mux_8_4_x is
Port ( mux_8_4_R0 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R1 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R2 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R3 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R4 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R5 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R6 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R7 : in STD_LOGIC_VECTOR (3 downto 0);
```

```
      mux_8_4_S : in STD_LOGIC_VECTOR (2 downto 0);
      --mux_8_4_EN : in STD_LOGIC;
      mux_8_4_Q : out STD_LOGIC_VECTOR (3 downto 0));
end mux_8_4_x;

architecture Behavioral of mux_8_4_x is

component Mux_x
   port(D : in STD_LOGIC_VECTOR (7 downto 0);
        S : in STD_LOGIC_VECTOR (2 downto 0);
        EN : in STD_LOGIC;
        Y : out STD_LOGIC);
end component;

begin
   mux_8_4_0:Mux_x
   port map(
   D(0) => mux_8_4_R0(0),
   D(1) => mux_8_4_R1(0),
   D(2) => mux_8_4_R2(0),
   D(3) => mux_8_4_R3(0),
   D(4) => mux_8_4_R4(0),
   D(5) => mux_8_4_R5(0),
   D(6) => mux_8_4_R6(0),
   D(7) => mux_8_4_R7(0),

   S =>mux_8_4_S,
   EN => '1',
   Y => mux_8_4_Q(0));

   mux_8_4_1:Mux_x
   port map(
   D(0) => mux_8_4_R0(1),
   D(1) => mux_8_4_R1(1),
   D(2) => mux_8_4_R2(1),
   D(3) => mux_8_4_R3(1),
   D(4) => mux_8_4_R4(1),
   D(5) => mux_8_4_R5(1),
   D(6) => mux_8_4_R6(1),
   D(7) => mux_8_4_R7(1),

   S =>mux_8_4_S,
   EN => '1',
   Y =>mux_8_4_Q(1));
```

```
    mux_8_4_2:Mux_x
    port map(
    D(0) => mux_8_4_R0(2),
    D(1) => mux_8_4_R1(2),
    D(2) => mux_8_4_R2(2),
    D(3) => mux_8_4_R3(2),
    D(4) => mux_8_4_R4(2),
    D(5) => mux_8_4_R5(2),
    D(6) => mux_8_4_R6(2),
    D(7) => mux_8_4_R7(2),

    S =>mux_8_4_S,
    EN => '1',
    Y => mux_8_4_Q(2));

    mux_8_4_3:Mux_x
    port map(
    D(0) => mux_8_4_R0(3),
    D(1) => mux_8_4_R1(3),
    D(2) => mux_8_4_R2(3),
    D(3) => mux_8_4_R3(3),
    D(4) => mux_8_4_R4(3),
    D(5) => mux_8_4_R5(3),
    D(6) => mux_8_4_R6(3),
    D(7) => mux_8_4_R7(3),

    S =>mux_8_4_S,
    EN =>'1',
    Y => mux_8_4_Q(3));

end Behavioral;
```

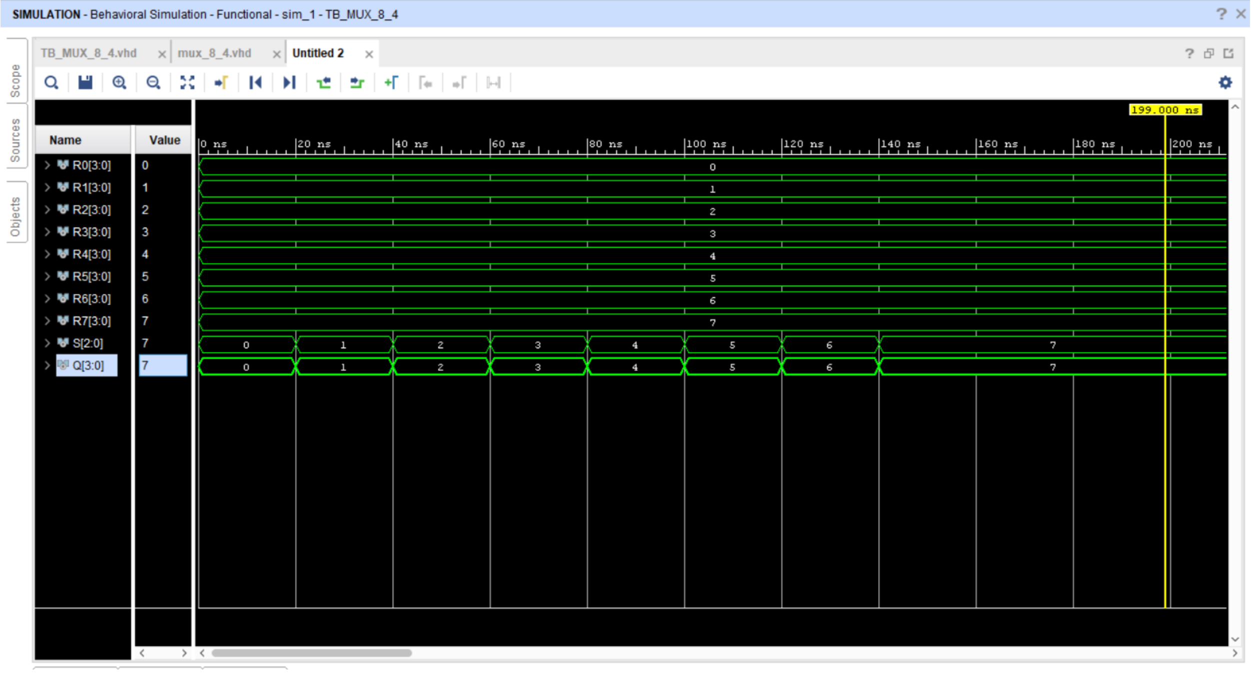
Test Bench Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 07/26/2022 08:44:18 AM
-- Design Name:
-- Module Name: TB_MUX_8_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_MUX_8_4 is
--  Port ( );
end TB_MUX_8_4;

architecture Behavioral of TB_MUX_8_4 is
component mux_8_4
port(mux_8_4_R0 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R1 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R2 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R3 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R4 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R5 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R6 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_R7 : in STD_LOGIC_VECTOR (3 downto 0);
    mux_8_4_S : in STD_LOGIC_VECTOR (2 downto 0);
    mux_8_4_EN : in STD_LOGIC;
    mux_8_4_Q : out STD_LOGIC_VECTOR (3 downto 0));
end component;

signal R0,R1,R2,R3,R4,R5,R6,R7 : STD_LOGIC_VECTOR (3 downto 0);
signal  S :  STD_LOGIC_VECTOR (2 downto 0);
signal  EN : STD_LOGIC;
signal  Q :  STD_LOGIC_VECTOR (3 downto 0);
```

```
begin
UUT: mux_8_4
PORT MAP(
mux_8_4_R0 => R0,
mux_8_4_R1 => R1,
mux_8_4_R2 => R2,
mux_8_4_R3 => R3,
mux_8_4_R4 => R4,
mux_8_4_R5 => R5,
mux_8_4_R6 => R6,
mux_8_4_R7 => R7,
mux_8_4_S => S,
mux_8_4_EN => EN,
mux_8_4_Q => Q);

process
   begin
   EN <= '1';
   R0 <="0000";
   R1 <="0001";
   R2 <="0010";
   R3 <="0011";
   R4 <="0100";
   R5 <="0101";
   R6 <="0110";
   R7 <="0111";
   S <= "000";
   wait for 20ns;
   S <= "001";
   wait for 20ns;
   S <= "010";
   wait for 20ns;
   S <= "011";
   wait for 20ns;
   S <= "100";
   wait for 20ns;
   S <= "101";
   wait for 20ns;
   S <= "110";
   wait for 20ns;
   S <= "111";
   wait;
end process;
end Behavioral;
```

Time Diagram



SIMULATION - Behavioral Simulation - Functional - sim_1 - TB_MUX_8_4

Name	Value
R0[3:0]	0
R1[3:0]	1
R2[3:0]	2
R3[3:0]	3
R4[3:0]	4
R5[3:0]	5
R6[3:0]	6
R7[3:0]	7
S[2:0]	7
Q[3:0]	7

4-bit Add / Subtract Unit

VHDL Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 07/22/2022 02:03:25 AM
-- Design Name:
-- Module Name: Add_Sub - 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 Add_Sub is
    Port ( add_sub_A : in STD_LOGIC_VECTOR (3 downto 0);
           add_sub_B : in STD_LOGIC_VECTOR (3 downto 0);
           add_sub_out : out STD_LOGIC_VECTOR (3 downto 0);
           add_sub_sel : in STD_LOGIC;
           Overflow : out STD_LOGIC;
           Zero : out STD_LOGIC);
end Add_Sub;
```

```
architecture Behavioral of Add_Sub is

component mux_2_4_add_sub
   Port( Mux_A : in STD_LOGIC_VECTOR (3 downto 0);
        Mux_B : in STD_LOGIC_VECTOR (3 downto 0);
        NegMux : in STD_LOGIC;
        Mux_out : out STD_LOGIC_VECTOR (3 downto 0));
end component;

component RCA_4_add_sub
   Port (  Ax : in STD_LOGIC_VECTOR(3 downto 0);
          Bx : in STD_LOGIC_VECTOR(3 downto 0);
          C_in : in STD_LOGIC;
          result : out STD_LOGIC_VECTOR(3 downto 0);
          C_out : out STD_LOGIC;
          C_out_FA3: out STD_LOGIC);
end component;

signal  not_add_sub_B, Mux_out,Result: STD_LOGIC_VECTOR (3 downto 0);
signal C_out,Neg,AS,C_out_FA3:std_logic;

begin
mux_2_4_add_sub_0:mux_2_4_add_sub
   PORT MAP(Mux_A => add_sub_B,
          Mux_B => not_add_sub_B,
          NegMux => Neg,
          Mux_out => Mux_out);

RCA_0:RCA_4_add_sub
   PORT MAP(Ax => add_sub_A,
          Bx => Mux_out,
          C_in =>AS,
          result => result,
          C_out => C_out,
          C_out_FA3 => C_out_FA3
          );


process(add_sub_sel)
   begin
   if (add_sub_sel='0') then
      AS<='0';
      Neg<='0'; --not sure
    else
```

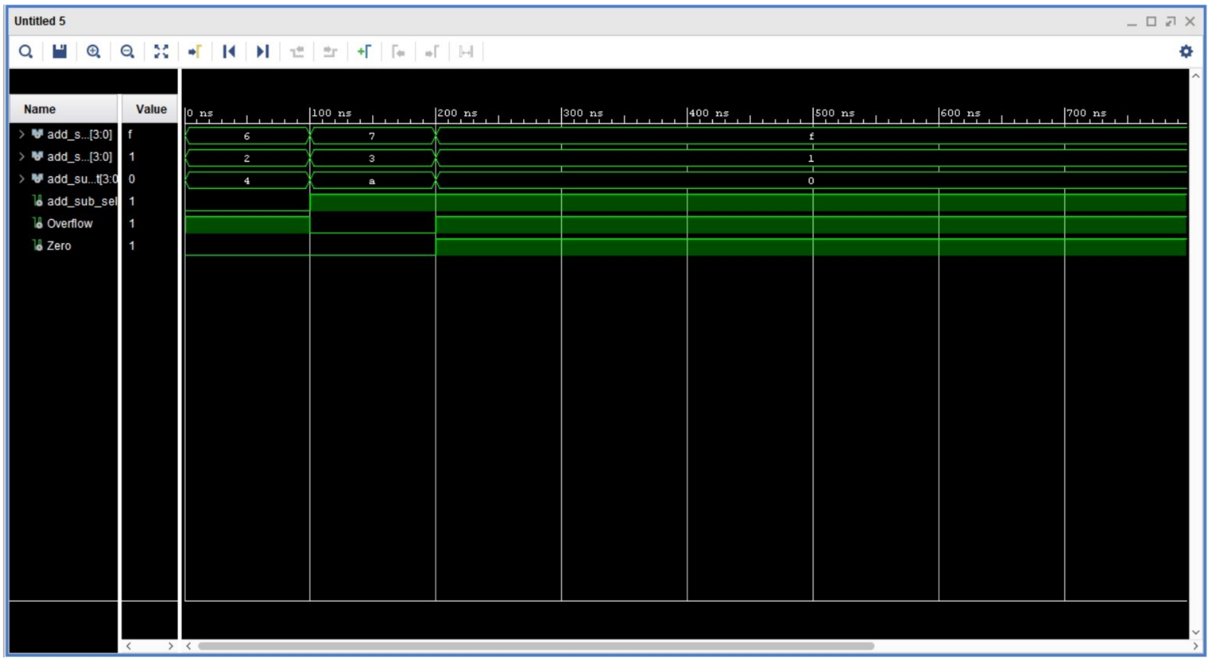
```
        AS<='1';
        Neg<='1'; --not sure
     end if;
end process;

not_add_sub_B <= NOT(add_sub_B);
add_sub_out <= result;
Zero <= NOT(result(0) OR result(1) OR result(2) OR result(3));
--Overflow <= C_out_FA3 XOR C_out;
Overflow <= C_out;
end Behavioral;
```

Test Bench Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer: P.D.N.T.Paramulla
--
-- Create Date: 07/22/2022 04:06:11 AM
-- Design Name: 4 bit add sub unit
-- Module Name: Add_Sub_TB - 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 Add_Sub_TB is
--  Port ( );
end Add_Sub_TB;

architecture Behavioral of Add_Sub_TB is
component Add_Sub
   Port ( add_sub_A : in STD_LOGIC_VECTOR (3 downto 0);
          add_sub_B : in STD_LOGIC_VECTOR (3 downto 0);
          add_sub_out : out STD_LOGIC_VECTOR (3 downto 0);
          add_sub_sel : in STD_LOGIC;
          Overflow : out STD_LOGIC;
          Zero : out STD_LOGIC);

end component;
  signal add_sub_A, add_sub_B, add_sub_out : STD_LOGIC_VECTOR(3 downto 0);
  signal add_sub_sel,Overflow,Zero : STD_LOGIC;

begin
   UUT: add_sub
   Port map(add_sub_A => add_sub_A,
            add_sub_B => add_sub_B,
            add_sub_out =>add_sub_out,
            add_sub_sel => add_sub_sel,
            Overflow => Overflow,
            Zero=>Zero);
Process

begin
   add_sub_sel <= '0';--for sub
   add_sub_A <= "0110";
   add_sub_B <= "0010";
   wait for 100ns;

   add_sub_sel <= '1'; --for add
   add_sub_A <= "1111";
   add_sub_B <= "0001";
   wait;
end process;
```

Time Diagram



Slow Clock

VHDL Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 06/14/2022 07:57:47 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
  process (Clk_in) begin
    if (rising_edge(Clk_in)) then
      count <=count+1;
      if(count = 4) then
        Clk_status <= not Clk_status;
        Clk_out <= Clk_status;
        count <=1;
       end if;
     end if;
  end process;
```

Test Bench Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
-- Create Date: 06/18/2022 11:01:27 PM
-- Design Name:
-- Module Name: Slow_clk_TB - 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_TB is
--  Port ( );
end Slow_clk_TB;

architecture Behavioral of Slow_clk_TB is
component Slow_clk
Port (Clk_in : in STD_LOGIC;
      Clk_out : out STD_LOGIC);
end component;
signal Clk_in : STD_LOGIC :='0';
signal Clk_out: STD_LOGIC;
begin
  UUT: Slow_Clk
  PORT MAP(
      Clk_in =>Clk_in,
      Clk_out =>Clk_out);
 process
 begin
    wait for 20ns;
    Clk_in <= NOT(Clk_in);
    end process;
end Behavioral;
```

7 Segment Display

VHDL Code (LUT)

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 06/28/2022 01:27:20 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;
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 := (
                    "1000000", --0
                    "1111001", --1
                    "0100100", --2
                    "0110000", --3
                    "0011001", --4
                    "0010010", --5
                    "0000010", --6
                    "1111000", --7
                    "0000000", --8
                    "0010000", --9
                    "0001000", --a
                    "0000011", --b
                    "1000110", --c
                    "0100001", --d
                    "0000110", --e
                    "0001110"  --f
                    );

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

end Behavioral;
```

Test Bench Code

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 06/23/2022 06:21:25 PM
-- Design Name:
-- Module Name: LUT_16_7_TB - 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_TB is
--  Port ( );
end LUT_16_7_TB;

architecture Behavioral of LUT_16_7_TB 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 <= "1100";
    wait for 50ns;

    address <= "1111";
    wait for 50ns;

    address <= "1110";
    wait for 50ns;

    address <= "0011";
    wait for 50ns;
   end process;

end Behavioral;
```

Nano Processor

VHDL Code

```
----------------------------------------------------------------------------------
-- Company: 
-- Engineer: 
-- 
-- Create Date: 08/03/2022 09:56:03 PM
-- Design Name: 
-- Module Name: Nano_processor - 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 is
  Port ( Clk : in STD_LOGIC;
         Reset0 : in STD_LOGIC;
         Sev_seg : out STD_LOGIC_VECTOR (3 downto 0);
         Zero0 : out STD_LOGIC;
         Overflow0 : out STD_LOGIC;
         To_LED : out STD_LOGIC_VECTOR (3 downto 0));
end Nano_processor;
```

```
architecture Behavioral of Nano_processor is

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

component Program_Counter
   Port ( Pro_Counter_Reset : in STD_LOGIC;
         Clk : in STD_LOGIC;
         Pro_Counter_Address_In : in STD_LOGIC_VECTOR (2 downto 0);
         Pro_Counter_Address_Out : out STD_LOGIC_VECTOR (2 downto 0));
end component;

component ROM
   Port ( Mem_Select : in STD_LOGIC_VECTOR (2 downto 0);
         instruction : out STD_LOGIC_VECTOR (11 downto 0));
end component;

component Add_3
   Port ( Add3_A : in STD_LOGIC_VECTOR (2 downto 0);
         Add3_B : in STD_LOGIC_VECTOR (2 downto 0);
         Add3_C_in : in STD_LOGIC;
         Add3_S : out STD_LOGIC_VECTOR (2 downto 0);
         Add3_C_out : out STD_LOGIC);
end component;

component mux_2_3_x
    Port ( mux_2_3_A : in STD_LOGIC_VECTOR (2 downto 0);
        mux_2_3_B : in STD_LOGIC_VECTOR (2 downto 0);
        mux_2_3_C0 : in STD_LOGIC;
        mux_2_3_Q : out STD_LOGIC_VECTOR (2 downto 0));
end component;

component inst_Decoder_unit
   Port ( inst_bus : in STD_LOGIC_VECTOR (11 downto 0);
         check_jump : in STD_LOGIC_VECTOR (3 downto 0);
         register_en : out STD_LOGIC_VECTOR (2 downto 0);
         register_sel_A : out STD_LOGIC_VECTOR (2 downto 0);
         register_sel_B : out STD_LOGIC_VECTOR (2 downto 0);
         immediate_value : out STD_LOGIC_VECTOR (3 downto 0);
         load_sel : out STD_LOGIC;
         jump_flag : out STD_LOGIC;
         add_sub_sel : out STD_LOGIC;
```

```
          Addr_jump : out STD_LOGIC_VECTOR (2 downto 0)
          );
end component;

component mux_2_4
    Port ( mux_2_4_A : in STD_LOGIC_VECTOR (3 downto 0);
         mux_2_4_B : in STD_LOGIC_VECTOR (3 downto 0);
         mux_2_4_C0 : in STD_LOGIC;
         mux_2_4_Q : out STD_LOGIC_VECTOR (3 downto 0));
end component;

component Register_Bank
   Port ( Data : in STD_LOGIC_VECTOR (3 downto 0);
          Clk : in STD_LOGIC;
          Reset : in STD_LOGIC;
          Enable : in STD_LOGIC_VECTOR (2 downto 0);
          R0 : out STD_LOGIC_VECTOR (3 downto 0);
          R1 : out STD_LOGIC_VECTOR (3 downto 0);
          R2 : out STD_LOGIC_VECTOR (3 downto 0);
          R3 : out STD_LOGIC_VECTOR (3 downto 0);
          R4 : out STD_LOGIC_VECTOR (3 downto 0);
          R5 : out STD_LOGIC_VECTOR (3 downto 0);
          R6 : out STD_LOGIC_VECTOR (3 downto 0);
          R7 : out STD_LOGIC_VECTOR (3 downto 0));
end component;

component mux_8_4_x
   Port ( mux_8_4_R0 : in STD_LOGIC_VECTOR (3 downto 0);
      mux_8_4_R1 : in STD_LOGIC_VECTOR (3 downto 0);
      mux_8_4_R2 : in STD_LOGIC_VECTOR (3 downto 0);
      mux_8_4_R3 : in STD_LOGIC_VECTOR (3 downto 0);
      mux_8_4_R4 : in STD_LOGIC_VECTOR (3 downto 0);
      mux_8_4_R5 : in STD_LOGIC_VECTOR (3 downto 0);
      mux_8_4_R6 : in STD_LOGIC_VECTOR (3 downto 0);
      mux_8_4_R7 : in STD_LOGIC_VECTOR (3 downto 0);
      mux_8_4_S : in STD_LOGIC_VECTOR (2 downto 0);
      mux_8_4_Q : out STD_LOGIC_VECTOR (3 downto 0));
end component;

component Add_Sub
   Port ( add_sub_A : in STD_LOGIC_VECTOR (3 downto 0);
          add_sub_B : in STD_LOGIC_VECTOR (3 downto 0);
          add_sub_out : out STD_LOGIC_VECTOR (3 downto 0);
          add_sub_sel : in STD_LOGIC;
          Overflow : out STD_LOGIC;
      Zero : out STD_LOGIC);
end component;

signal Slow_Clk_out,Add3_C_out,jump_flag,load_sel,add_sub_sel,Overflow,Zero:STD_LOGIC;
signal
mux_8_4_A_out,immediate_value,add_sub_out,mux_2_4_out,mux_8_4_B_out:STD_LOGIC_VECTO
R (3 downto 0);
signal instruction:STD_LOGIC_VECTOR (11 downto 0);
signal
Pro_Counter_Address_Out,Add3_out,register_en,register_sel_A,register_sel_B,mux_2_3_out,Addr_
jump:STD_LOGIC_VECTOR (2 downto 0);
signal data_bus:STD_LOGIC_VECTOR (31 downto 0);
begin
Slow_Clk_0:Slow_Clk
  Port map ( Clk_in => Clk,
        Clk_out => Slow_Clk_out);

Program_Counter_0:Program_Counter
  Port map(Pro_Counter_Reset => Reset0,
       Clk => Slow_Clk_out,
       Pro_Counter_Address_In => mux_2_3_out,
       Pro_Counter_Address_Out => Pro_Counter_Address_Out);

ROM_0:ROM
  Port map(
       Mem_Select => Pro_Counter_Address_Out,
       instruction => instruction);

Add_3_0:Add_3
  Port map(Add3_A => "001",
       Add3_B => Pro_Counter_Address_Out,
       Add3_C_in =>'0',
       Add3_S =>Add3_out,
       Add3_C_out =>Add3_C_out);

mux_2_3_x_0:mux_2_3_x
  Port map(mux_2_3_A =>Add3_out,
       mux_2_3_B =>Addr_jump,
       mux_2_3_C0 =>jump_flag,
       mux_2_3_Q => mux_2_3_out
      );

inst_Decoder_unit_0:inst_Decoder_unit
  Port map(inst_bus => instruction,
      check_jump => mux_8_4_A_out,
      register_en => register_en,
      register_sel_A => register_sel_A,
      register_sel_B => register_sel_B,
      immediate_value => immediate_value,
      load_sel => load_sel,
      jump_flag => jump_flag,
      add_sub_sel => add_sub_sel,
      Addr_jump => Addr_jump
      );

mux_2_4_0 :mux_2_4
  Port map(mux_2_4_A => immediate_value,
       mux_2_4_B => add_sub_out,
       mux_2_4_C0 => load_sel,
       mux_2_4_Q => mux_2_4_out
        );

Register_Bank_0:Register_Bank
  Port map(Data=>mux_2_4_out,
      Clk => Slow_Clk_out,
      Reset => Reset0,
      Enable => register_en,
      R0 => data_bus(3 downto 0),
      R1 => data_bus(7 downto 4),
      R2 => data_bus(11 downto 8),
      R3 => data_bus(15 downto 12),
      R4 => data_bus(19 downto 16),
      R5 => data_bus(23 downto 20),
      R6 => data_bus(27 downto 24),
      R7 => data_bus(31 downto 28)
      );

mux_8_4_x_0:mux_8_4_x
  Port map(mux_8_4_R0 => data_bus(3 downto 0),
          mux_8_4_R1 => data_bus(7 downto 4),
          mux_8_4_R2 => data_bus(11 downto 8),
          mux_8_4_R3 => data_bus(15 downto 12),
          mux_8_4_R4 => data_bus(19 downto 16),
          mux_8_4_R5 => data_bus(23 downto 20),
          mux_8_4_R6 => data_bus(27 downto 24),
          mux_8_4_R7 => data_bus(31 downto 28),
          mux_8_4_S => register_sel_A,
          mux_8_4_Q => mux_8_4_A_out
     );
```

```
mux_8_4_x_1:mux_8_4_x
     Port map(mux_8_4_R0 => data_bus(3 downto 0),
             mux_8_4_R1 => data_bus(7 downto 4),
             mux_8_4_R2 => data_bus(11 downto 8),
             mux_8_4_R3 => data_bus(15 downto 12),
             mux_8_4_R4 => data_bus(19 downto 16),
             mux_8_4_R5 => data_bus(23 downto 20),
             mux_8_4_R6 => data_bus(27 downto 24),
             mux_8_4_R7 => data_bus(31 downto 28),
             mux_8_4_S => register_sel_B,
             mux_8_4_Q => mux_8_4_B_out
            );

Add_Sub_0:Add_Sub
    Port map(add_sub_A => mux_8_4_A_out,
            add_sub_B => mux_8_4_B_out,
           add_sub_out =>add_sub_out,
           add_sub_sel => add_sub_sel,
          Overflow => Overflow,
          Zero => Zero
          );

Sev_seg <=data_bus(31 downto 28);
Zero0 <= Zero;
Overflow0 <= Overflow;
To_LED <= data_bus(31 downto 28);

end Behavioral;
```

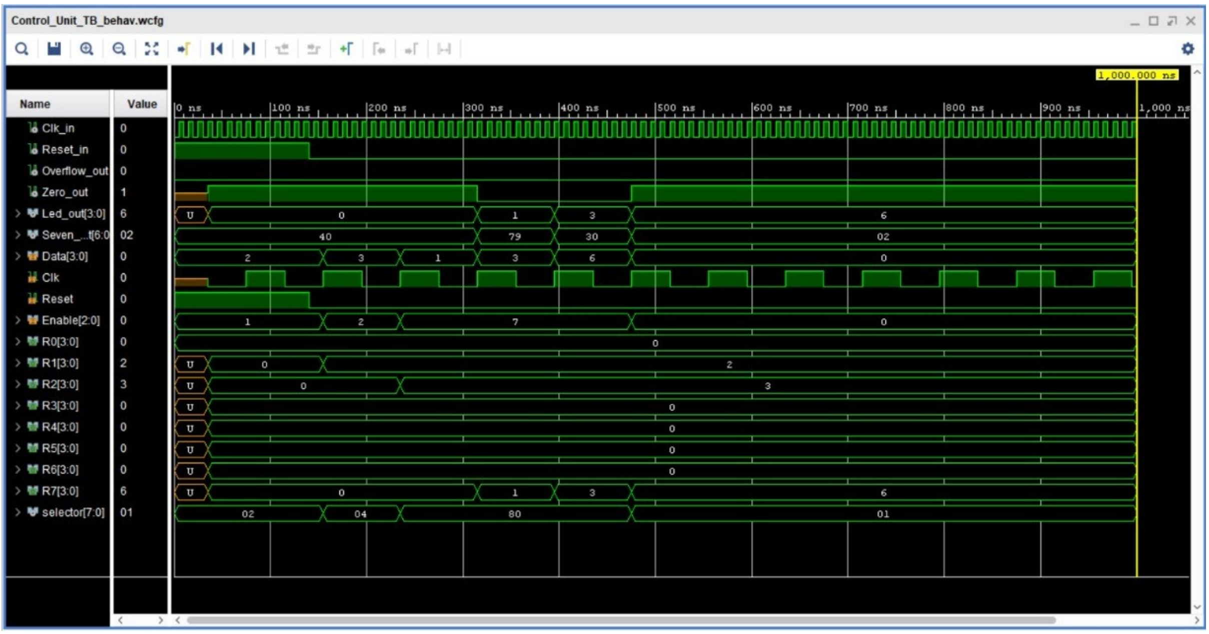
Test Bench Code

```
-- Company:
-- Engineer:
--
-- Create Date: 08/04/2022 12:40:19 AM
-- Design Name: Control unit
-- Module Name: Control_Unit_TB - 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 Control_Unit_TB is
--  Port ( );
end Control_Unit_TB;

architecture Behavioral of Control_Unit_TB is
component Control_Unit
  Port ( Clk_in : in STD_LOGIC;
         Reset_in : in STD_LOGIC;
         Seven_seg_out : out STD_LOGIC_VECTOR (6 downto 0);
         Led_out : out STD_LOGIC_VECTOR (3 downto 0);
         Overflow_out : out STD_LOGIC;
         Zero_out : out STD_LOGIC);
end component;

signal Clk_in: STD_LOGIC:='0';
signal Reset_in,Overflow_out,Zero_out : STD_LOGIC;
signal Seven_seg_out : STD_LOGIC_VECTOR (6 downto 0);
signal Led_out : STD_LOGIC_VECTOR (3 downto 0);

begin
UUT: Control_Unit
  Port map(Clk_in => Clk_in,
          Reset_in => Reset_in,
          Seven_seg_out => Seven_seg_out,
          Led_out => Led_out,
          Overflow_out => Overflow_out,
          Zero_out => Zero_out );
```

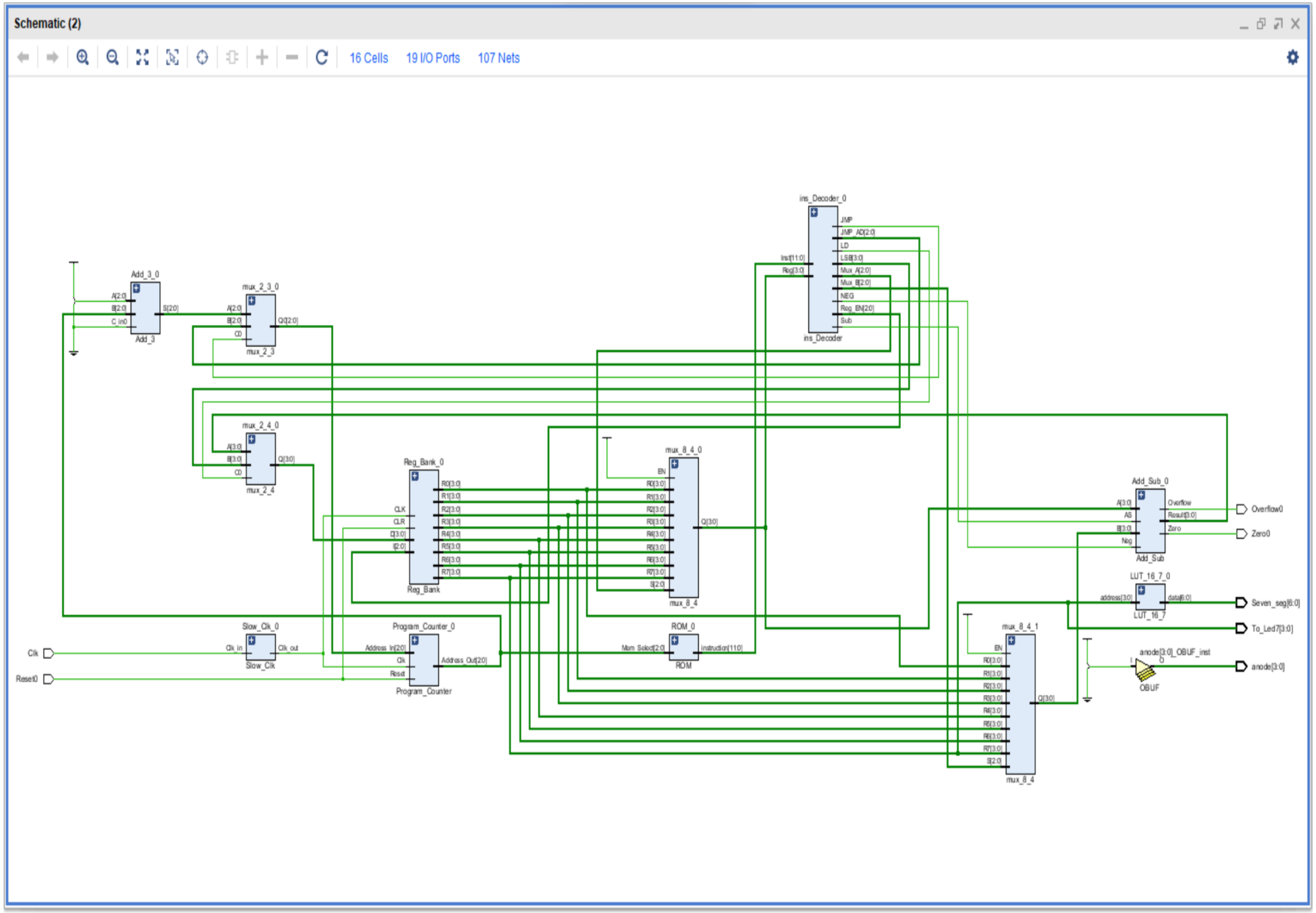
```
process
   begin
      wait for 5ns;
      Clk_in <=NOT(Clk_in);
end process;

process
begin
   Reset_in <='1';
   wait for 140ns;

   Reset_in <='0';
   wait;
end process;

end Behavioral;
```

Time Diagram

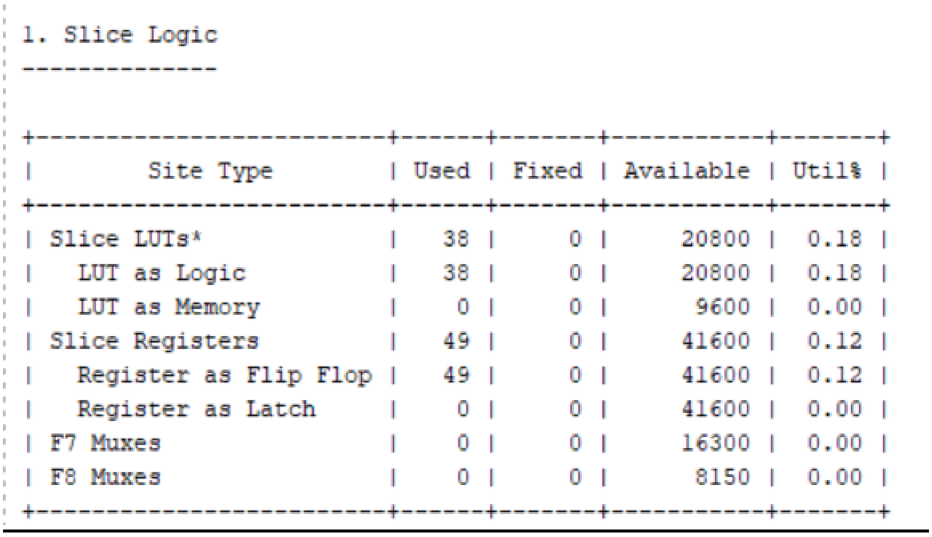


Name	Value
Clk_in	0
Reset_in	0
Overflow_out	0
Zero_out	1
Led_out[3:0]	6
Seven_...t[6:0	02
Data[3:0]	0
Clk	0
Reset	0
Enable[2:0]	0
R0[3:0]	0
R1[3:0]	2
R2[3:0]	3
R3[3:0]	0
R4[3:0]	0
R5[3:0]	0
R6[3:0]	0
R7[3:0]	6
selector[7:0]	01

Schematic Diagram



Resource consumption optimizations to the basic program designs, and final LUT/FF counts



```
1. Slice Logic
--------------

+-------------------------+------+-------+-----------+-------+
|        Site Type        | Used | Fixed | Available | Util% |
+-------------------------+------+-------+-----------+-------+
| Slice LUTs*             |   38 |     0 |     20800 |  0.18 |
|   LUT as Logic          |   38 |     0 |     20800 |  0.18 |
|   LUT as Memory         |    0 |     0 |      9600 |  0.00 |
| Slice Registers         |   49 |     0 |     41600 |  0.12 |
|   Register as Flip Flop |   49 |     0 |     41600 |  0.12 |
|   Register as Latch     |    0 |     0 |     41600 |  0.00 |
| F7 Muxes                |    0 |     0 |     16300 |  0.00 |
| F8 Muxes                |    0 |     0 |      8150 |  0.00 |
+-------------------------+------+-------+-----------+-------+
```

Additional instructions/features and the Optimization

As for optimizations, our main concern was towards the instruction decoder. We identified the repeated use of 'If' loops along with the inefficient extraction of data from the instructions read as potential issues that could be further perfected.

Hence the repeated ' If ' loops were replaced by logic gates while the instruction data that was read at the initialization of the instruction decoder was sliced and assigned then and there for later usage.

Even more, we made a slight improvement within the adder-subtractor unit by replacing the Two mux enablers which were responsible for invertion of a signal and adding a logic 1 to the end bit respectively, with a signal enabler which returned the same functionality hence reducing the number of signals that had to be passed back and forth.

Our main concern with the last implementation of the Nano processor was the presence of 11 latches. After the modifications mentioned above, we brought the latches count to 0. Even more, the usage of LUTs dropped from 41 to 38 due to the replacement of multiple repeated 'If' loops.

Conclusion

- The main objective of this lab is to develop a larger component using simple components.

- The development of the instruction decoder, register bank, and program ROM led to the creation of the nano processor.

- Also, we create an 8-way 4-bit multiplexer using an 8-way 1-bit multiplexer. Similarly, we use 2 way 1-bit multiplexer to create the 2-way 3-bit multiplexer and 2-way 4-bit multiplexer.

- We evaluated the accuracy of the internal components by simulating several inputs.

- In addition, the component can be made more efficient by using tri-state buffers to create multiplexers.

- This design can be made more complex and larger by expanding the storage of the register bank and program ROM.

- Finally, we can improve the component by reducing the fundamental logic gates and adding more features.

Contribution of team members

Task	Group Member	Hours Spent
Design Code for ROM	Rupasinghe R.A.M.T.	2 Hours
Test Bench for ROM	Rupasinghe R.A.M.T.	1 Hour
Design Code for Registers	Rupasinghe R.A.M.T.	2 Hours
Test Bench for Registers	Rupasinghe R.A.M.T.	1 Hour
Design Code for all the Multiplexers	Wickramarathna H.K.G.V.L.	2 Hours
Test Bench for Multiplexers	Wickramarathna H.K.G.V.L.	1 Hour
Design Code for Instruction Decoder	Paramulla P.D.N.T.	3 Hours
Test Bench for Instruction Decoder	Paramulla P.D.N.T.	1 Hour
Design Code for Program Counter	Ekanayake E.M.C.L.	2 Hours
Test Bench for Program Counter	Ekanayake E.M.C.L.	1 Hour
Design Code for Add/Sub Unit	Chinthana S.K.G.	3 Hours
Test Bench for Add/Sub Unit	Chinthana S.K.G.	1 Hour
Final Processor Design Code	Everyone	6 Hours
Test Bench for final Processor	Everyone	3 Hours
Final Project report	Everyone	1 Hour
Total Time Spent		30 Hours