Lab 4 – Combinational Circuits

CS1050 Computer Organization and Digital Design

Name : **Dissanayake D.M.A.K.**
Index No. : **2202135N**
Group : **44**

Lab Task: Design and Testing of Decoder(2-to-4 and 3-to-8) and 8-to-1 Multiplexer

Introduction

This lab sheet focuse on the design and testing of digital decoders and multiplexers, specifically the 2-to-4 decoder, 3-to-8 decoder, and 8-to-1 multiplexer. These components are useful in information processing and signal routing in electronic devices.

Decoders, like the 2-to-4 and 3-to-8 variants, serve to interpret coded inputs, translating them into distinct outputs. The 2-to-4 decoder, for instance, takes two inputs and generates four outputs, playing a fundamental role in binary decoding. Building on this, the 3-to-8 decoder expands the capacity to handle three inputs and produce eight unique output states, showcasing the scalability of these components in complex digital systems.

Complement of the decoders is the 8-to-1 multiplexer, a versatile component essential for signal routing. It merges eight input lines into a single output line, with the selection lines determining the output source.

The steps of this lab involve creating truth tables, deriving Boolean expressions, developing circuit diagrams, simulating the designed circuits and finally testing the multiplexer on the BASYS3 board.

CONTENT

1) 2-to-4 Decoder

Truth table/boolean expression with steps

Enable (EN)	Input (I1, I0)	Output (Y3, Y2, Y1, Y0)
0	x	0000
1	00	0001
1	01	0010
1	10	0100
1	11	1000

We can derive the boolean expressions for the outputs Y3, Y2, Y1, and Y0 based on the input variables I1, I0, and the enable pin EN.

The boolean expressions for the outputs are as follows:

$$Y3 = EN * I1' * I0'$$

$$Y2 = EN * I1' * I0$$

$$Y1 = EN * I1 * I0'$$

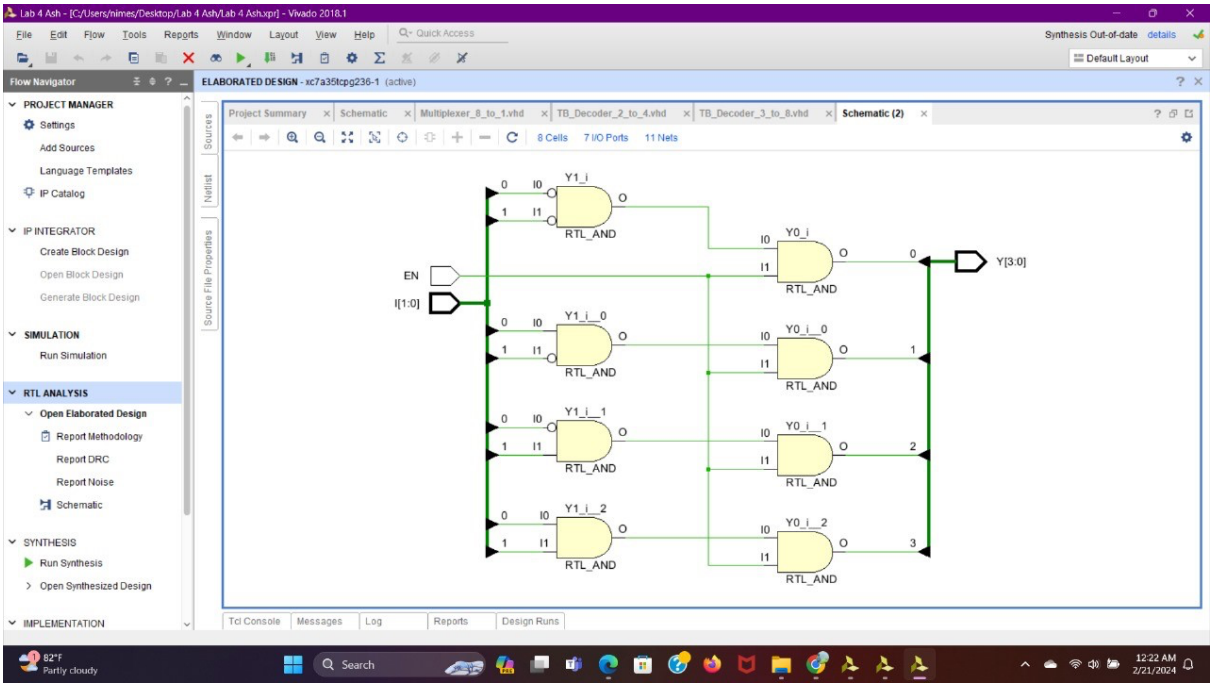
$$Y0 = EN * I1 * I0$$

Design source file

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

## - Create Date: 02/20/2024 01:32:24 PM
-- 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 instantiati
-- 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

begin

Y(0) <= not I(0) and not I(1) and EN;
Y(1) <= I(0) and not I(1) and EN;
Y(2) <= not I(0) and I(1) and EN;
Y(3) <= I(0) and I(1) and EN;

end Behavioral;
```

Elaborated design schematic



Simulation source file

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

## - Create Date: 02/20/2024 01:37:57 PM
-- 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
instantiati-- 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 EN : std_logic; SIGNAL Y :
std_logic_vector(3 downto 0);

begin

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

process

begin
```

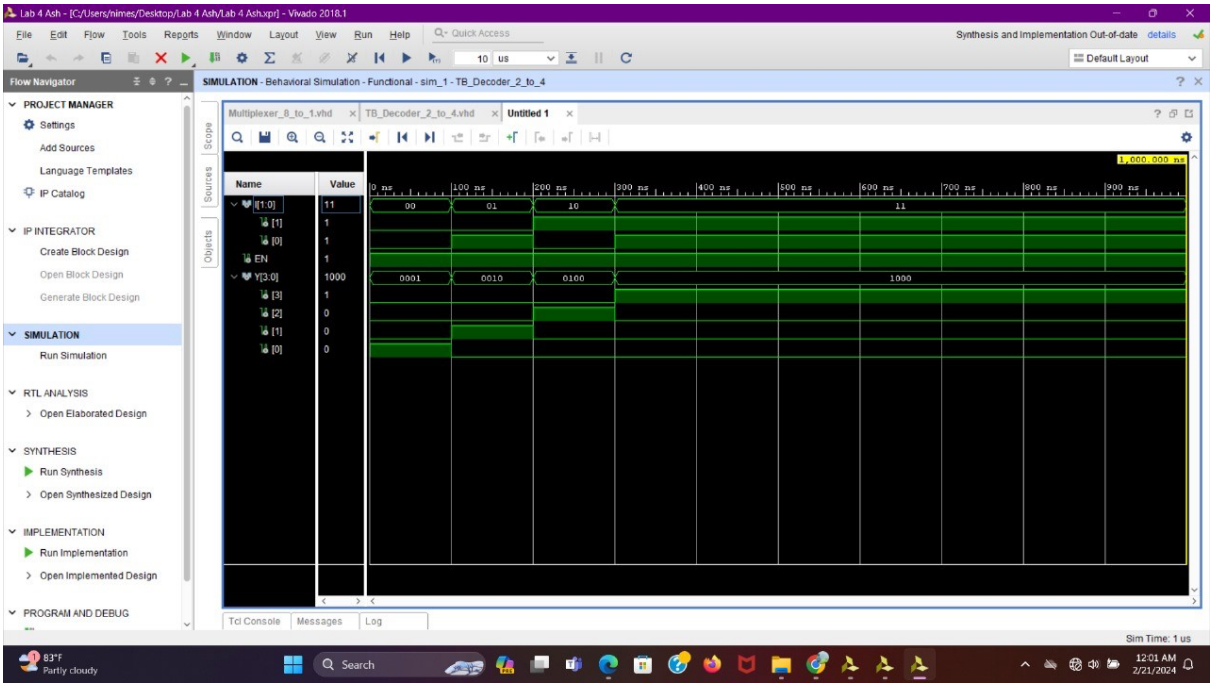
```
EN <= '1';
I <= "00";
WAIT FOR 100 ns;

```
I <= "01";
WAIT FOR 100 ns;

I <= "10";
WAIT FOR 100 ns;

I <= "11";
WAIT;

```

end process;

end Behavioral;
```

Timing diagram



2) 3-to-8 Decoder

Truth table/boolean expression with steps

Enable (EN)	Inputs (I2, I1, I0)	Outputs (Y7, Y6, Y5, Y4, Y3, Y2, Y1, Y0)
0	xxx	00000000
1	000	00000001
1	001	00000010
1	010	00000100
1	011	00001000
1	100	00010000
1	101	00100000
1	110	01000000
1	111	10000000

We can derive the boolean expressions for the outputs Y7 to Y0 based on the input variables I2, I1, I0, and the enable pin EN. The boolean expressions for the outputs are as follows:

$$Y7 = EN * I2' * I1' * I0'$$

$$Y6 = EN * I2' * I1' * I0$$

$$Y5 = EN * I2' * I1 * I0'$$

$$Y4 = EN * I2' * I1 * I0$$

$$Y3 = EN * I2 * I1' * I0'$$

$$Y2 = EN * I2 * I1' * I0$$

$$Y1 = EN * I2 * I1 * I0'$$

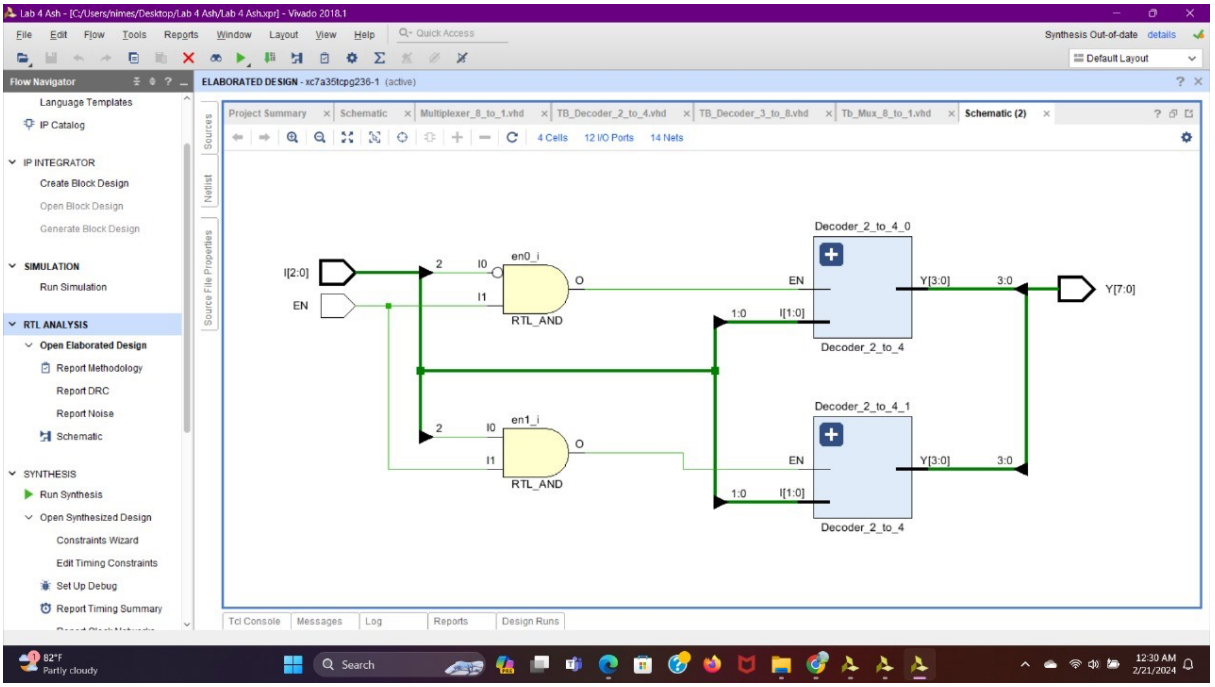
$$Y0 = EN * I2 * I1 * I0$$

Design source file

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

## - Create Date: 02/20/2024 02:00:34 PM
-- 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 instantiati
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity Decode_3_to_8 is
Port (  I : in STD_LOGIC_VECTOR (2 downto 0);
EN : in STD_LOGIC;
Y : out STD_LOGIC_VECTOR (7 downto 0));
end Decode_3_to_8;
```

```
architecture Behavioral of Decode_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)) AND EN;
en1 <= I(2) AND EN;
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;
```

Elaborated design schematic



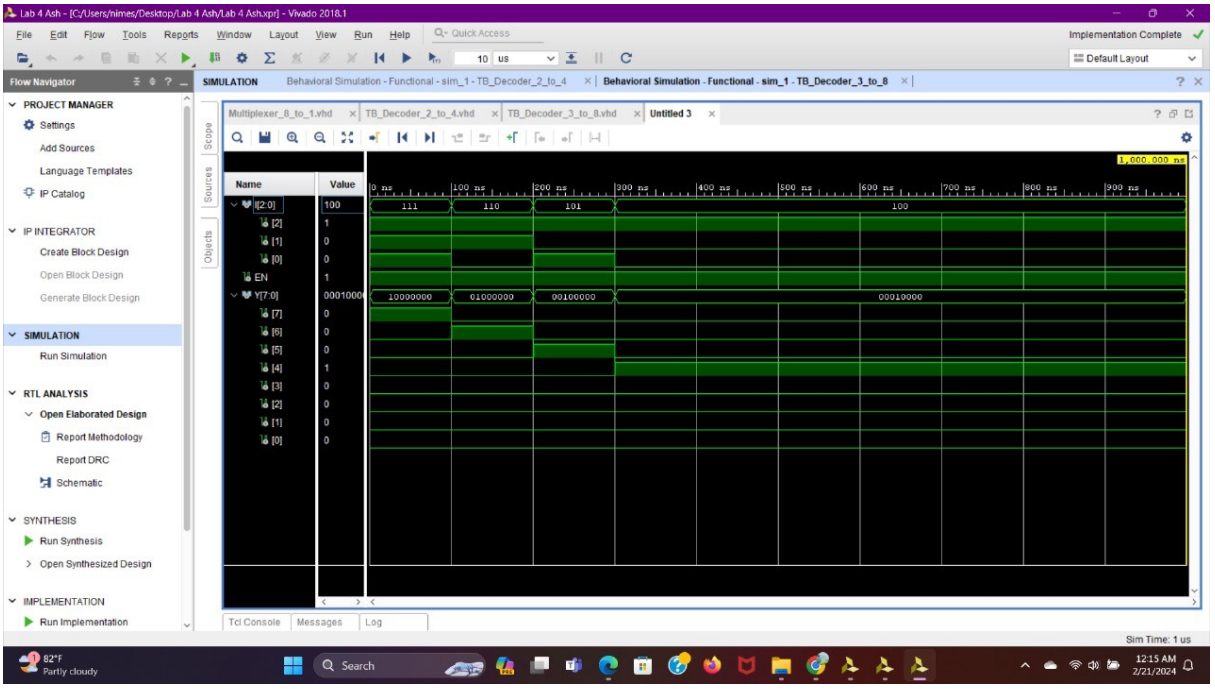
Simulation source file

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

## - Create Date: 02/20/2024 02:12:12 PM
-- 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 instantiati
-- 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 Decode_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;

SIGNAL I : std_logic_vector(2 downto 0);
SIGNAL EN : std_logic;
SIGNAL Y : std_logic_vector(7 downto 0);

begin

UUT: Decode_3_to_8 PORT MAP(
I => I,
EN => EN,
Y => Y );

process
begin
```

```
EN <= '1';

I <= "111";
WAIT FOR 100 ns;

I <= "110";
WAIT FOR 100 ns;

I <= "101";
WAIT FOR 100 ns;

I <= "100";
WAIT FOR 100 ns;
WAIT;

end process;

end Behavioral;
```

Timing diagram



3) 8-to-1 Multiplexer

Truth table/boolean expression with steps

Enable (EN)	Select Lines (S2, S1, S0)	Inputs (D7, D6, D5, D4, D3, D2, D1, D0)	Output (R)
0	xxx	00000000	0
1	000	I0	I0
1	001	I1	I1
1	010	I2	I2
1	011	I3	I3
1	100	I4	I4
1	101	I5	I5
1	110	I6	I6
1	111	I7	I7

Derive the Boolean Expression The boolean expression for the output Y can be derived based on the select lines (S2, S1, S0), the inputs (D7, D6, D5, D4, D3, D2, D1, D0) and the enable pin (EN).

The boolean expression for the output R is as follows:

$$R = (EN * S2' * S1' * S0' * D0) + (EN * S2' * S1' * S0 * D1) + (EN * S2' * S1 * S0' * D2) + (EN * S2' * S1 * S0 * D3) + (EN * S2 * S1' * S0' * D4) + (EN * S2 * S1' * S0 * D5) + (EN * S2 * S1 * S0' * D6) + (EN * S2 * S1 * S0 * D7)$$

$$R = EN((S2' * S1' * S0' * D0) + (S2' * S1' * S0 * D1) + (S2' * S1 * S0' * D2) + (S2' * S1 * S0 * D3) + (S2 * S1' * S0' * D4) + (S2 * S1' * S0 * D5) + (S2 * S1 * S0' * D6) + (S2 * S1 * S0 * D7))$$

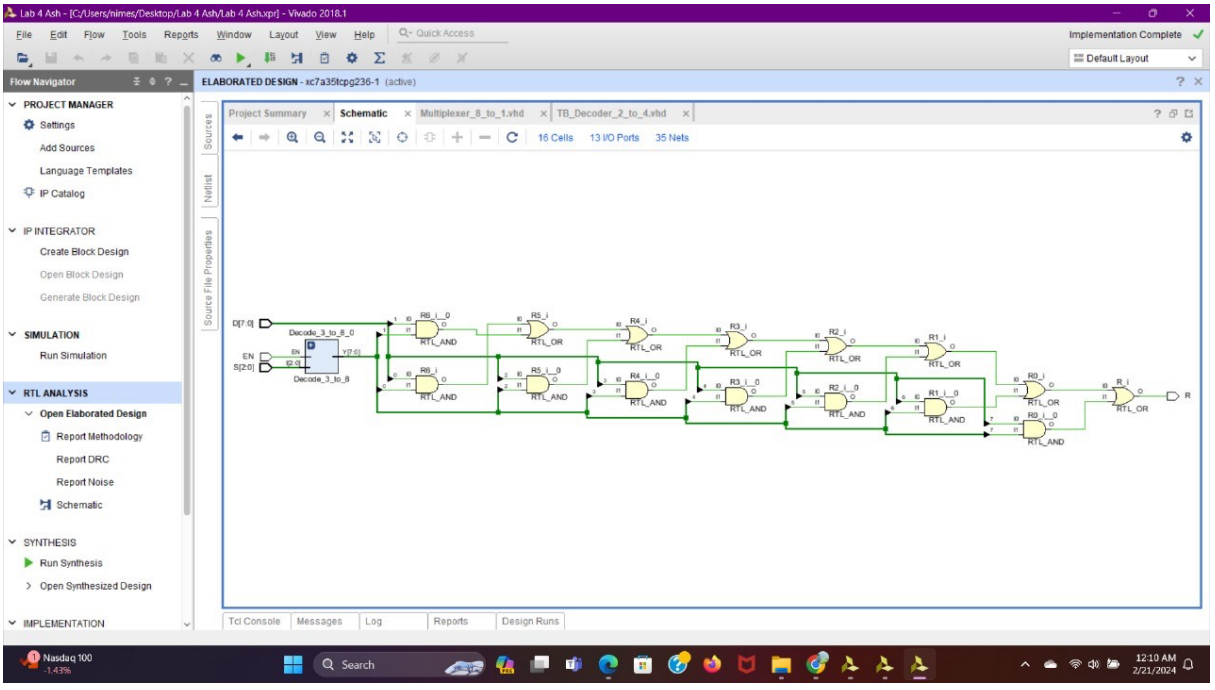
Design source file

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

## - Create Date: 02/20/2024 03:07:18 PM
-- Design Name:
-- Module Name: Muxr_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
instantiati-- any Xilinx leaf cells in this code.
--library UNISIM; --use
UNISIM.VComponents.all;

entity Muxr_8_to_1 is Port ( D : in
STD_LOGIC_VECTOR (7 downto 0);
S : in STD_LOGIC_VECTOR (2 downto 0);
EN : in STD_LOGIC;
R : out
STD_LOGIC); end
Muxr_8_to_1;

architecture Behavioral of Muxr_8_to_1 is

component Decode_3_to_8
port( I: in
STD_LOGIC_VECTOR;
EN: in STD_LOGIC;
Y: out STD_LOGIC_VECTOR );
```

```
end component;

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

begin
Decode_3_to_8_0 : Decode_3_to_8
port map(
I => S,
EN => EN,
Y => Y );
R <= (D(0) and Y(0)) or (D(1) and Y(1)) or (D(2) and Y(2)) or
     (D(3) and Y(3)) or (D(4) and Y(4)) or (D(5) and Y(5))or
     (D(6) and Y(6)) or (D(7) and Y(7));

end Behavioral;
```

Elaborated design schematic



Simulation source file

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

## - Create Date: 02/20/2024 03:18:57 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 instantiati
-- 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 Muxr_8_to_1
```

```
PORT( D : in STD_LOGIC_VECTOR (7 downto 0);
S : in STD_LOGIC_VECTOR (2 downto 0);
EN : in STD_LOGIC;
R : out STD_LOGIC);
END COMPONENT;

SIGNAL D :  std_logic_vector (7 downto 0);
SIGNAL S :  std_logic_vector (2 downto 0);
SIGNAL EN : std_logic;
SIGNAL R : std_logic;

begin

UUT: Muxr_8_to_1 PORT MAP(
D => D,
S => S,
EN => EN,
R => R);

process
begin

```
EN <= '1';
D <= "10101010";
S <= "111";

WAIT for 100 ns;

S <= "110";
WAIT for 100 ns;

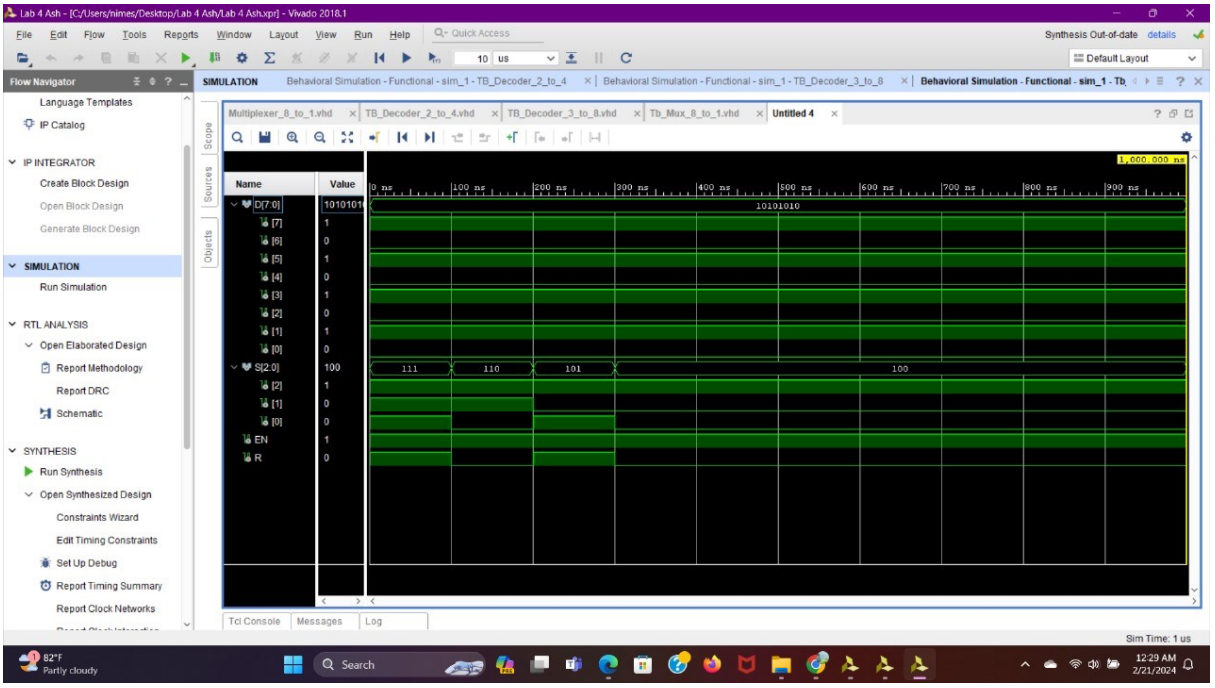
S <= "101";

WAIT for 100 ns;

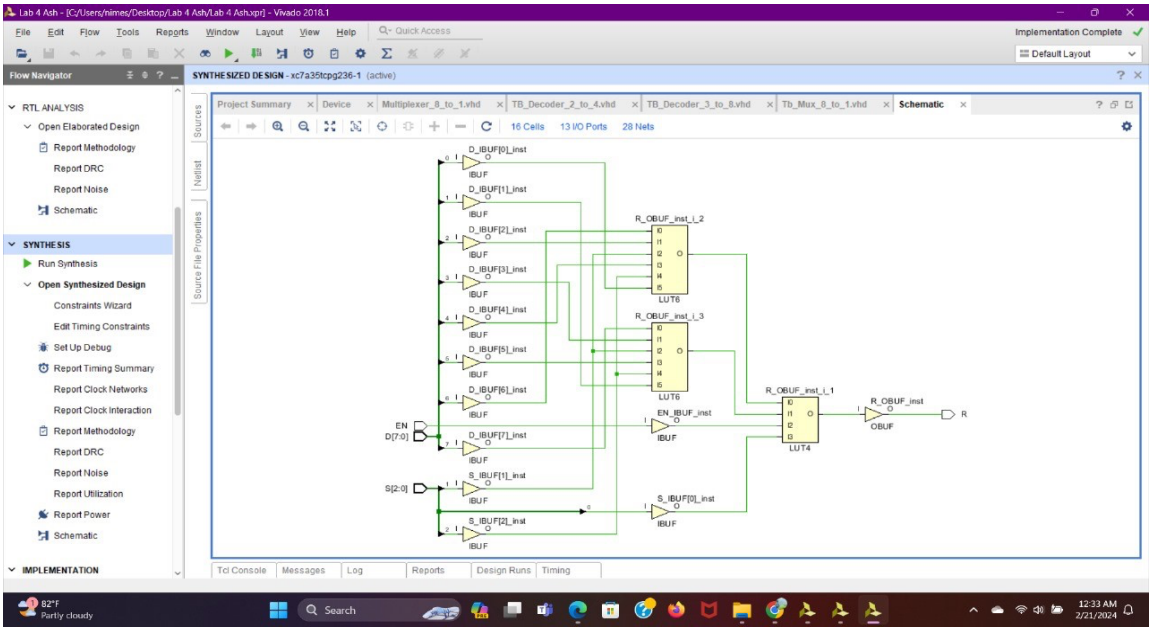
S <= "100";
WAIT;
```
```

```
end process;

end Behavioral;
```

Timing diagram



Implemented design schematic



Constraints files

```
## Switches

set_property PACKAGE_PIN V17 [get_ports {D[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {D[0]}]
set_property PACKAGE_PIN V16 [get_ports {D[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {D[1]}]
set_property PACKAGE_PIN W16 [get_ports {D[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {D[2]}]
set_property PACKAGE_PIN W17 [get_ports {D[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {D[3]}]
set_property PACKAGE_PIN W15 [get_ports {D[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {D[4]}]
set_property PACKAGE_PIN V15 [get_ports {D[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {D[5]}]
set_property PACKAGE_PIN W14 [get_ports {D[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {D[6]}]
set_property PACKAGE_PIN W13 [get_ports {D[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {D[7]}]
set_property PACKAGE_PIN U1 [get_ports {S[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {S[0]}]
set_property PACKAGE_PIN T1 [get_ports {S[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {S[1]}]
set_property PACKAGE_PIN R2 [get_ports {S[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {S[2]}]

## LEDs

set_property PACKAGE_PIN U16 [get_ports {R}]
set_property IOSTANDARD LVCMOS33 [get_ports {R}]

##Buttons
set_property PACKAGE_PIN U18 [get_ports EN]
set_property IOSTANDARD LVCMOS33 [get_ports EN]
```

4) Conclusion:

- At the end of this lab project, we can conclude that the process of designing a 2-to-4 decoder and utilizing it to build a 3-to-8 decoder, followed by the construction of an 8-to-1 multiplexer using the 3-to-8 decoder, has provided a practical understanding of digital circuit design.

- In simple terms, we can say that by learning how to design and implement these digital circuits, we have gained valuable insights into the fundamental building blocks of digital systems. This experience has allowed us to comprehend how smaller components, such as the 2-to-4 decoder, can be combined to create more complex circuits like the 3-to-8 decoder and the 8-to-1 multiplexer. Overall, this project has provided a hands-on opportunity to grasp the concepts of digital logic and circuit design.

-End-