Lab : **04 (Decoder and Multiplexer)**

Student Name : **M.K.S.L.Weerasiri**

Index Number : **220689N**

Group : **Group 32**

1.0 Introduction

Encoders, Decoders and Multiplexers are key components of a microprocessor. In order to build a microprocessor, we have to design, simulate and implement 2x4 Decoder, 3x8 Decoder and 8:1 MUX in this lab. Decoder helps to convert n coded signal in to 2^n signals. Multiplexer is acting as a "Data Selector". In this lab, we have to add extra signal called EN(Enable) which is acting as a global on/off switch for all the inputs and outputs. We are going to design them as below.

1.1. Lab Task

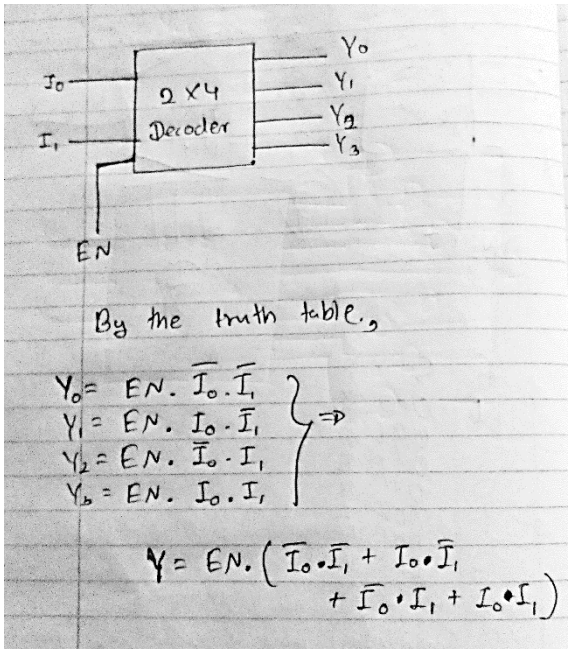
- Design and simulate 2x4 decoder (Inputs => I(0-1),EN | Outputs => Y(0-3))
- Design and simulate 3x8 decoder using 2x4 decoder (Inputs => I(0-2),EN | Outputs => Y(0-7))
- Design and simulate 8x1 MUX (Multiplexer) using 3x8 decoder (Inputs => D(0-7),S(0-2),EN | Outputs => Y)

2.0 2x4 Decoder

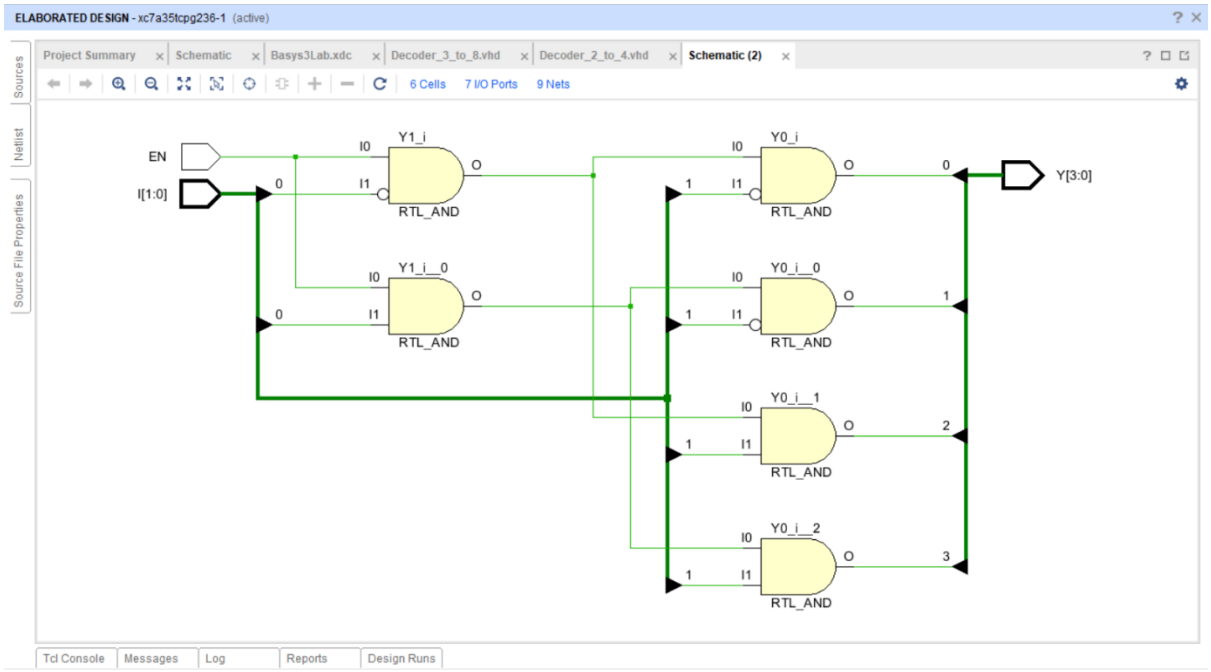
2.1. Truth table

EN	I		Y			
	I1	I0	Y3	Y2	Y1	Y0
0	x	x	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

2.2. Simplification Steps



2.3. Schematics



2.4. VHDL files

2.4.1. Decoder_2_to_4.vhd

```
----------------------------------------------------------------------------------
----
-- Company:
-- Engineer:
--
-- Create Date: 02/20/2024 01:50:22 PM
-- Design Name:
```

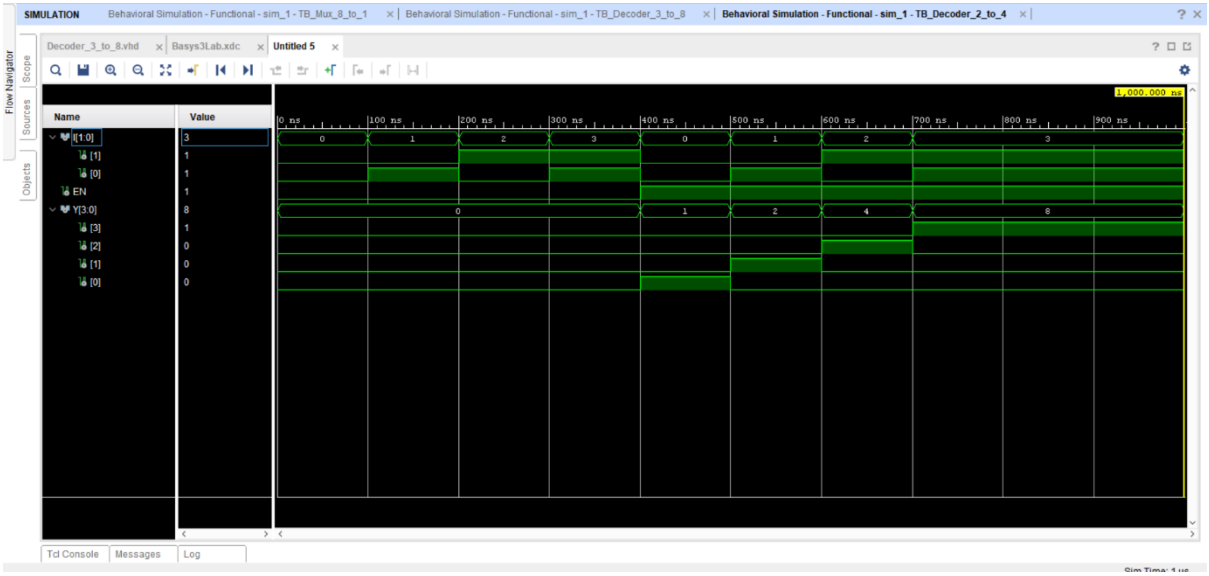
```
-- Module Name: Decoder_2_to_4 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------
----


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 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

begin
    Y(0) <= EN AND (NOT I(0)) AND (NOT I(1));
    Y(1) <= EN AND I(0) AND (NOT I(1));
    Y(2) <= EN AND (NOT I(0)) AND I(1);
    Y(3) <= EN AND I(0) AND I(1);



end Behavioral;
```

2.4.2. TB_Decoder_2_to_4.vhd

```
----------------------------------------------------------------------------------
----
-- Company:
-- Engineer:
--
-- Create Date: 02/20/2024 01:55:45 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 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 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<='0';I(1)<='0';I(0)<='0';WAIT FOR 100 NS;

            EN<='0';I(1)<='0';I(0)<='1';WAIT FOR 100 NS;

            EN<='0';I(1)<='1';I(0)<='0';WAIT FOR 100 NS;

            EN<='0';I(1)<='1';I(0)<='1';WAIT FOR 100 NS;

            EN<='1';I(1)<='0';I(0)<='0';WAIT FOR 100 NS;

            EN<='1';I(1)<='0';I(0)<='1';WAIT FOR 100 NS;

            EN<='1';I(1)<='1';I(0)<='0';WAIT FOR 100 NS;

            EN<='1';I(1)<='1';I(0)<='1';WAIT;

    END PROCESS;



end Behavioral;
```

2.5. Timing Diagram (for all combinations)

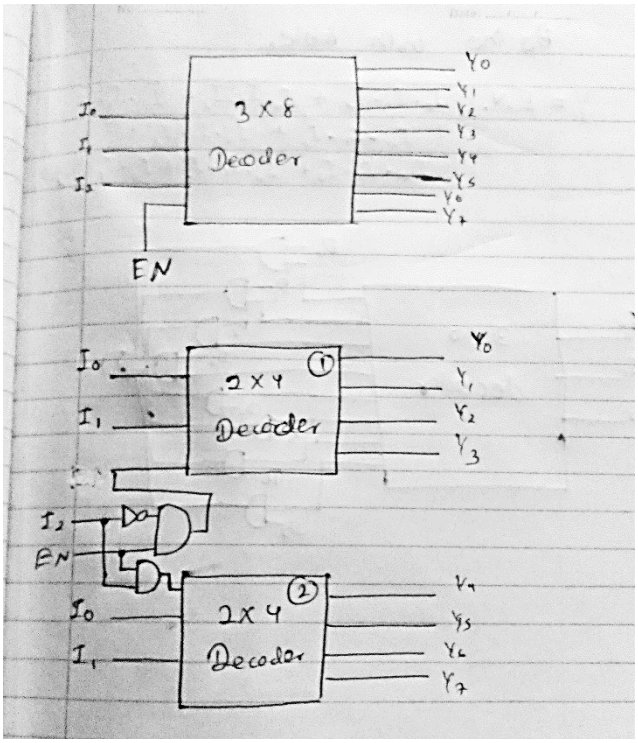


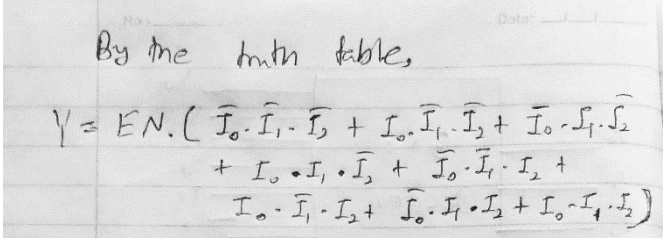
3.0 3x8 Decoder

3.1. Truth table

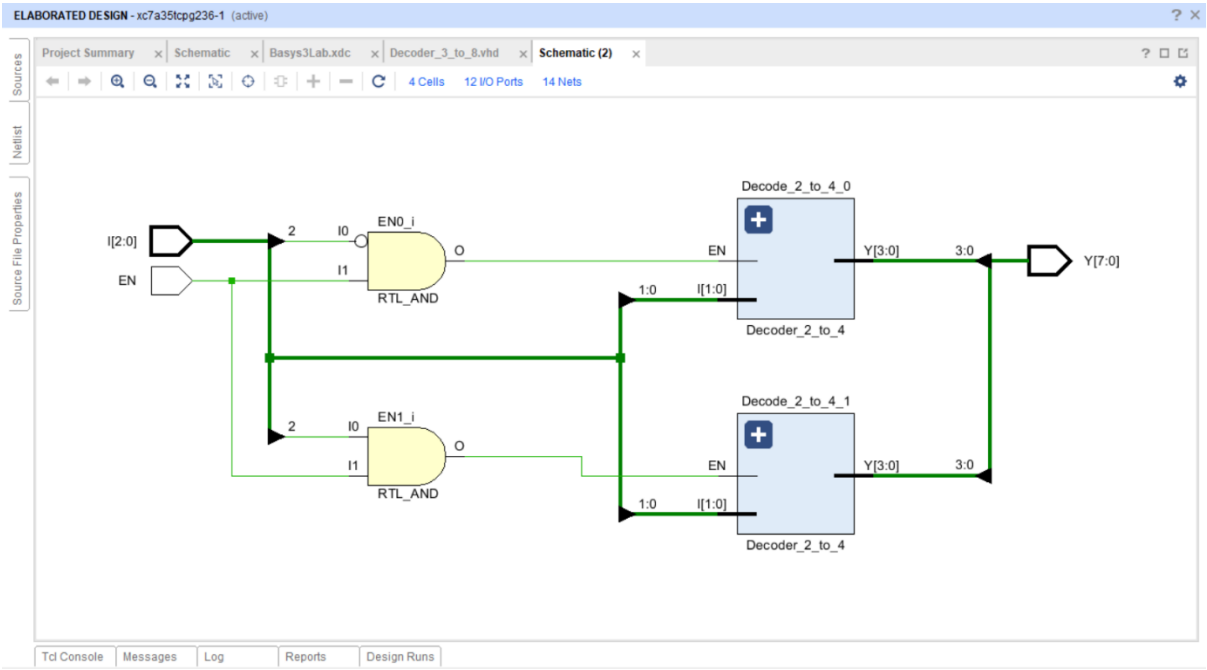
EN	I			Y							
	I2	I1	I0	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
0	x	x	x	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0	0	0	1	0
1	0	1	0	0	0	0	0	0	1	0	0
1	0	1	1	0	0	0	0	1	0	0	0
1	1	0	0	0	0	0	1	0	0	0	0
1	1	0	1	0	0	1	0	0	0	0	0
1	1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0

3.2. Simplification Steps





3.3. Schematics



3.4. VHDL files

3.4.1. Decoder_3_to_8.vhd

```
----------------------------------------------------------------------------------
----
-- Company:
-- Engineer:
--
-- Create Date: 02/20/2024 02:09:13 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 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);
           EN : in STD_LOGIC;
           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 EN0,EN1,I2 : STD_LOGIC;
SIGNAL Y0,Y1 : STD_LOGIC_VECTOR (3 DOWNTO 0);
begin
    Decode_2_to_4_0 : Decoder_2_to_4
        PORT MAP(
            I => I0,
            EN => EN0,
            Y => Y0
        );
    Decode_2_to_4_1 : Decoder_2_to_4
        PORT MAP(
            I => I1,
            EN => EN1,
            Y => Y1
        );
    EN0 <= (NOT I(2)) AND EN;--first one should only work 3rd wire is removed
    EN1 <= I(2) AND EN;--second one should work only 3rd wire is available

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

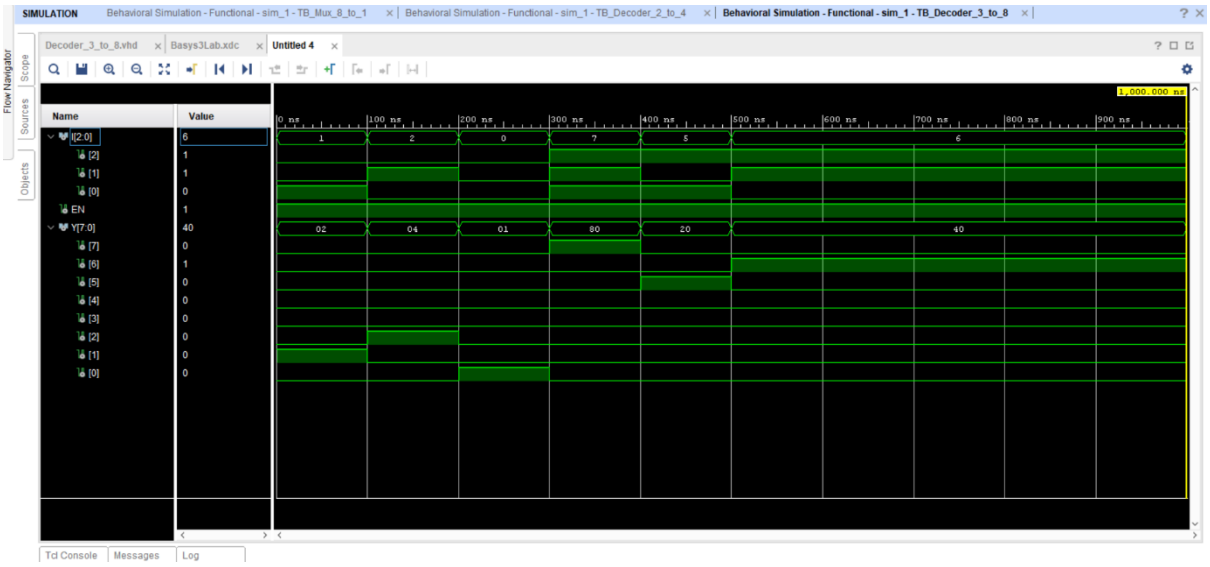
```
    Y(3 DOWNTO 0) <= Y0;
    Y(7 DOWNTO 4) <= Y1;



end Behavioral;
```

3.4.2. TB_ Decoder_3_to_8.vhd

```
----------------------------------------------------------------------------------
----
-- Company:
-- Engineer:
--
-- Create Date: 02/20/2024 02:21:19 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 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);
        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 : Decoder_3_to_8
        PORT MAP(
            I => I,
            EN => EN,
            Y => Y
        );
    PROCESS
        BEGIN
            --bin(220689)
            --'0b 110 101 111 000 010 001'
            --001
            EN<='1';I(2)<='0';I(1)<='0';I(0)<='1';WAIT FOR 100 NS;
            --010
            EN<='1';I(2)<='0';I(1)<='1';I(0)<='0';WAIT FOR 100 NS;
            --000
            EN<='1';I(2)<='0';I(1)<='0';I(0)<='0';WAIT FOR 100 NS;
            --111
            EN<='1';I(2)<='1';I(1)<='1';I(0)<='1';WAIT FOR 100 NS;
            --101
            EN<='1';I(2)<='1';I(1)<='0';I(0)<='1';WAIT FOR 100 NS;
            --110
            EN<='1';I(2)<='1';I(1)<='1';I(0)<='0';WAIT;
    END PROCESS;

end Behavioral;
```

3.5. Timing Diagram (for all combinations)



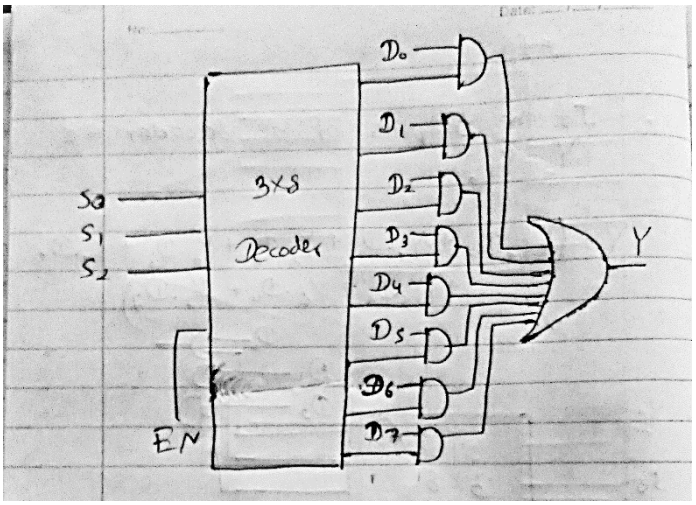
4.0 8x1 MUX

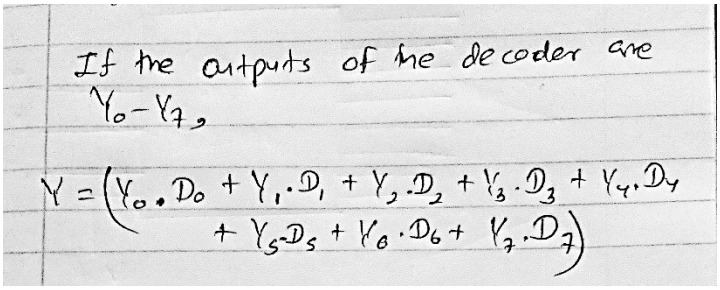
4.1. Truth table (Only for test cases)

(Registration No in binary 11 0101 1110 0001 0001_2 **)**

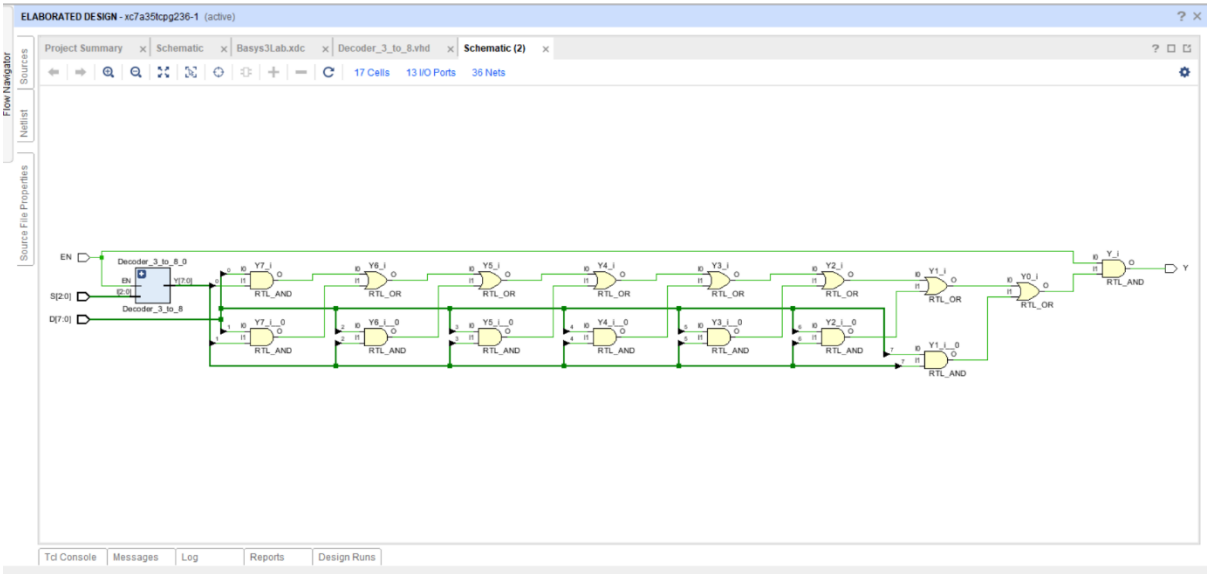
EN	S			D								Y0								Y
	S2	S1	S0	D7	D6	D5	D4	D3	D2	D1	D0	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	
0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0
1	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0
1	0	1	0									0	0	0	0	0	1	0	0	0
1	0	0	0									0	0	0	0	0	0	0	1	1
1	1	1	1									1	0	0	0	0	0	0	0	0
1	1	0	1									0	0	1	0	0	0	0	0	0
1	1	1	0									0	1	0	0	0	0	0	0	0
1	0	0	1	0	1	0	1	1	1	1	0	0	0	0	0	0	0	1	0	1
1	0	1	0									0	0	0	0	0	1	0	0	1
1	0	0	0									0	0	0	0	0	0	0	1	0
1	1	1	1									1	0	0	0	0	0	0	0	0
1	1	0	1									0	0	1	0	0	0	0	0	0
1	1	1	0									0	1	0	0	0	0	0	0	1

4.2. Simplification Steps





4.3. Schematics



4.4. VHDL files

4.4.1. Mux_8_to_1.vhd

```
----------------------------------------------------------------------------------
----
-- Company:
-- Engineer:
--
-- Create Date: 02/20/2024 03:23:31 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;
```

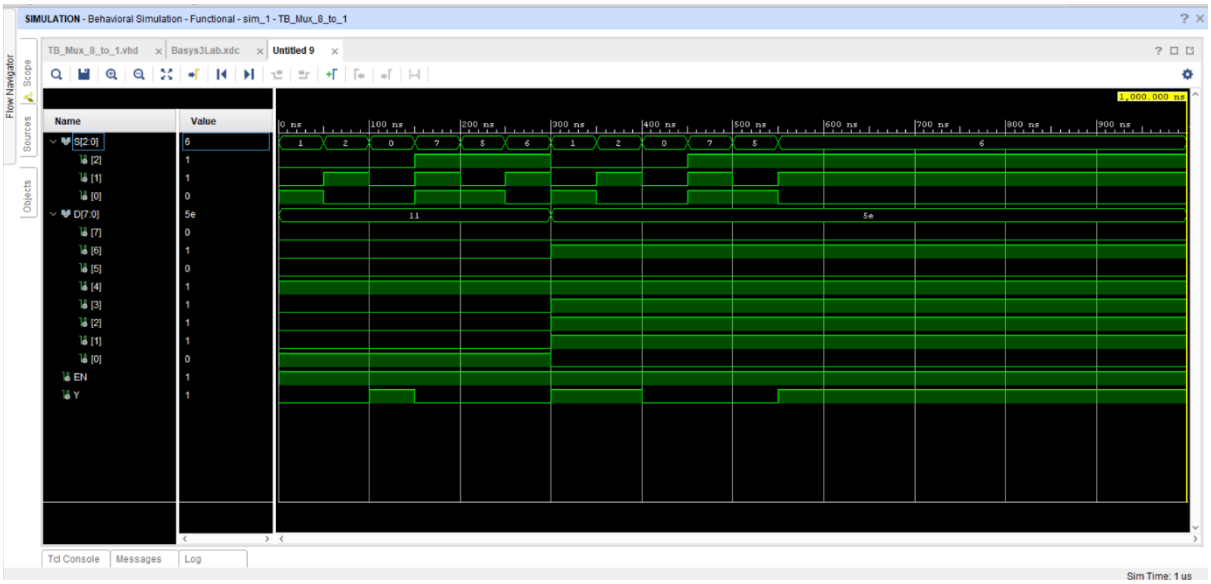
```
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- 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 ( 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 Mux_8_to_1;

architecture Behavioral of Mux_8_to_1 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;
SIGNAL Y0 : STD_LOGIC_VECTOR (7 DOWNTO 0);
begin
    Decoder_3_to_8_0 : Decoder_3_to_8
        PORT MAP(
            I => S,
            EN => EN,
            Y => Y0
        );
    Y <= EN AND ((D(0) AND Y0(0))
        OR (D(1) AND Y0(1))
        OR (D(2) AND Y0(2))
        OR (D(3) AND Y0(3))
        OR (D(4) AND Y0(4))
        OR (D(5) AND Y0(5))
        OR (D(6) AND Y0(6))
        OR (D(7) AND Y0(7))) ;

end Behavioral;
```

4.4.2. TB_Mux_8_to_1.vhd

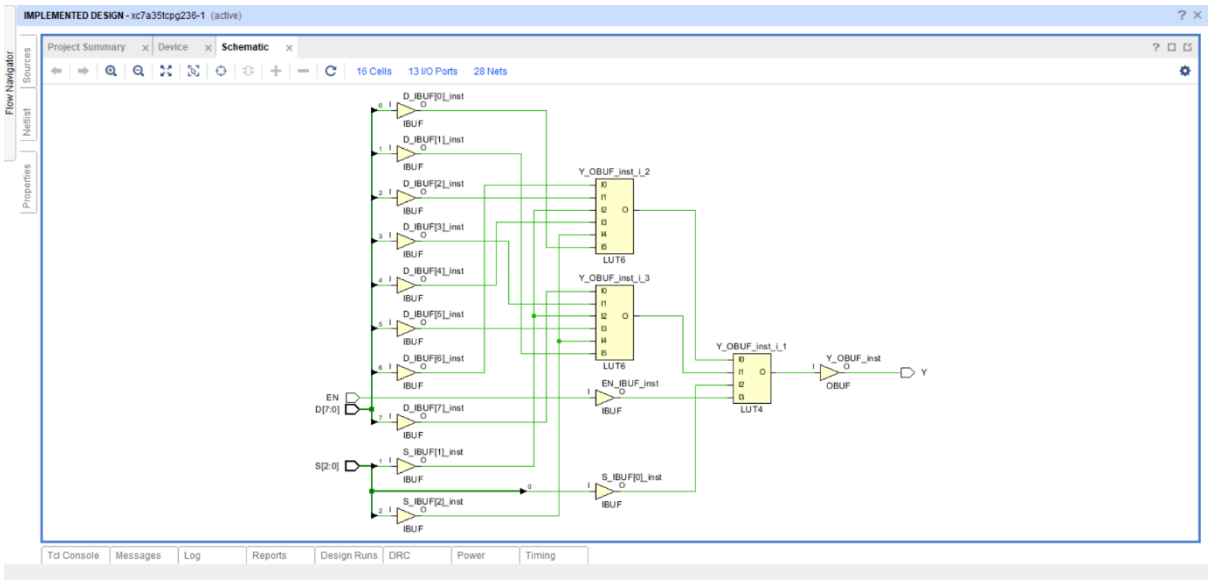
```
----------------------------------------------------------------------------------
----
-- Company:
-- Engineer:
--
-- Create Date: 02/20/2024 03:35:39 PM
-- Design Name:
-- Module Name: TB_Mux_8_to_1 - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- 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 ( 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;
```

```
SIGNAL S : STD_LOGIC_VECTOR (2 DOWNTO 0);
SIGNAL D : STD_LOGIC_VECTOR (7 DOWNTO 0);
SIGNAL EN,Y : STD_LOGIC;
begin
    UUT : Mux_8_to_1
        PORT MAP(
            D => D,
            S => S,
            EN => EN,
            Y => Y
        );
    PROCESS
        BEGIN
        EN <='1';
        --bin(220689)
        --'0b 110 101 111 000 010 001'
        D <= "00010001";
        --001
        S <= "001";WAIT FOR 100 NS;
        --010
        S <= "010";WAIT FOR 100 NS;
        --000
        S <= "000";WAIT FOR 100 NS;
        --111
        S <= "111";WAIT FOR 100 NS;
        --101
        S <= "101";WAIT FOR 100 NS;
        --110
        S <= "110";WAIT FOR 100 NS;

        D <= "01011110";
        --001
        S <= "001";WAIT FOR 100 NS;
        --010
        S <= "010";WAIT FOR 100 NS;
        --000
        S <= "000";WAIT FOR 100 NS;
        --111
        S <= "111";WAIT FOR 100 NS;
        --101
        S <= "101";WAIT FOR 100 NS;
        --110
        S <= "110";WAIT;

    END PROCESS;

end Behavioral;
```

4.5. Timing Diagram (for some combinations)



4.6. Implemented Design Schematics



4.7. Constraints file

```
## 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 {Y}]
    set_property IOSTANDARD LVCMOS33 [get_ports {Y}]


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

5.0 Conclusion

- First we designed 2x4 Decoder to decode 2 signals into 4 signals.
- Then we designed 3x8 Decoder by using the 2x4 decoder which we have created early steps.
- After that we designed 8:1 Multiplexer using 3x8 decoder which we have created early steps.
- For all the designs, we have included EN extra signal to ON/OFF the inputs and the outputs.
- And we have simulated all the designs and implemented in Basys3 boards.