VLSI Lab Report

Submitted by:

Name: Sourav Dutta

Roll: 001610501076

Group: A3

Department of Computer Science and Engineering (BCSE-4)
Jadavpur University, Kolkata

CONTENTS

ASSIGNMENT – 1

Problem

Design a 4x2 Encoder using

(a) Structural modelling
(b) Behavioural modelling

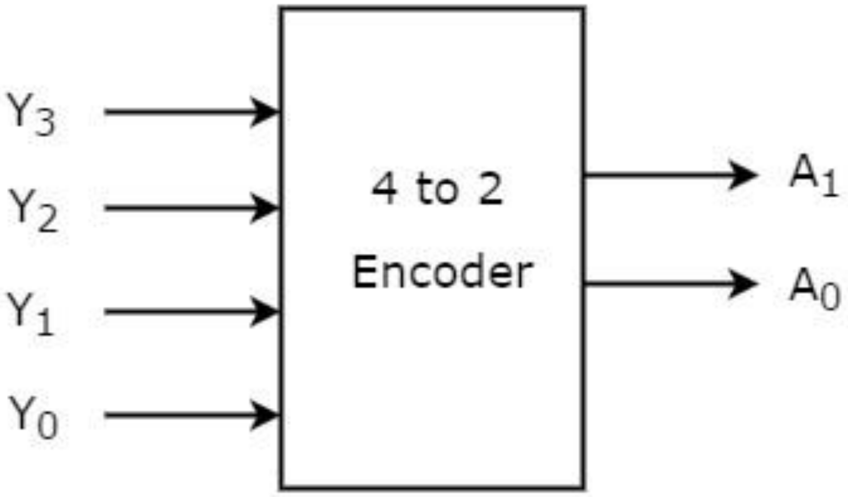
Description

An **Encoder** is a combinational circuit that performs the reverse operation of Decoder. It has maximum of 2^n input lines and ‘n’ output lines. It will produce a binary code equivalent to the input, which is active High. Therefore, the encoder encodes 2^n input lines with ‘n’ bits. It is optional to represent the enable signal in encoders. Here, the value of n is 2, as there are 4 input lines and 2 output lines.

Truth Table

Input				**Output**	
i_3	**i_2**	**i_1**	**i_0**	**o_1**	**o_0**
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

Block Diagram



VHDL Code

i) Using structural modelling

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

entity fourtotwoencoder_structural is
    Port ( i : in  STD_LOGIC_VECTOR (3 downto 0);
           o : out  STD_LOGIC_VECTOR (1 downto 0));
end fourtotwoencoder_structural;

architecture Structural of fourtotwoencoder_structural is
begin

    o(0)<=i(1) or i(3);
    o(1)<=i(2) or i(3);

end Structural;
```

ii) Using behavioral modelling (using concurrent statements)

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

entity fourtotwoencoder_behavioral_c is
    Port ( i : in  STD_LOGIC_VECTOR (3 downto 0);
           o : out  STD_LOGIC_VECTOR (1 downto 0));
end fourtotwoencoder_behavioral_c;

architecture Behavioral of fourtotwoencoder_behavioral_c is
begin

with i select o<=
    "00" when "0001",
    "01" when "0010",
    "10" when "0100",
    "11" when "1000",
    "ZZ" when others;

end Behavioral;
```

iii) Using behavioural modelling (using sequential statements)

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

entity fourtotwoencoder_behavioral_s is
    Port ( i : in  STD_LOGIC_VECTOR (3 downto 0);
           o : out STD_LOGIC_VECTOR (1 downto 0));
end fourtotwoencoder_behavioral_s;
```

```vhdl
architecture Behavioral of fourtotwoencoder_behavioral_s is
begin

Process(i)
begin
    case i is
        when "0001" => o<="00";
        when "0010" => o<="01";
        when "0100" => o<="10";
        when "1000" => o<="11";
        when others => o<="ZZ";
    end case;

end Process;

end Behavioral;
```

ASSIGNMENT – 2

Problem

Design a 8x3 Encoder using

(a) Select statement

(b) Case statement

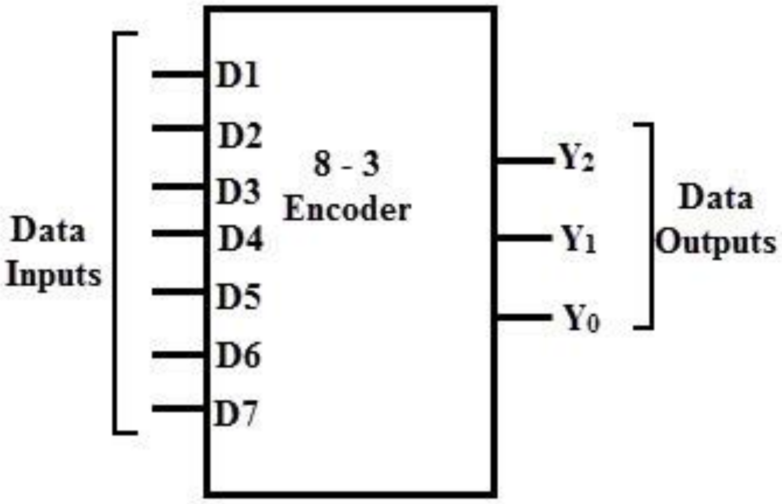
Description

An **Encoder** is a combinational circuit that performs the reverse operation of Decoder. It has maximum of 2^n input lines and 'n' output lines. It will produce a binary code equivalent to the input, which is active High. Therefore, the encoder encodes 2^n input lines with 'n' bits. It is optional to represent the enable signal in encoders. Here, the value of n is 3, as there are 8 input lines and 3 output lines.

Truth Table

Input								**Output**		
i_7	i_6	i_5	i_4	i_3	i_2	i_1	i_0	o_2	o_1	o_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

Block Diagram



VHDL Code

i) Using select statements (behavioural modelling)

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

entity eighttothree_select is
    Port ( i : in  STD_LOGIC_VECTOR (7 downto 0);
           o : out  STD_LOGIC_VECTOR (2 downto 0));
end eighttothree_select;

architecture Behavioral of eighttothree_select is
begin

with i select o<=
    "000" when "00000001",
    "001" when "00000010",
    "010" when "00000100",
    "011" when "00001000",
    "100" when "00010000",
    "101" when "00100000",
    "110" when "01000000",
    "111" when "10000000",
    "ZZZ" when others;

end Behavioral;
```

ii) Using case statements (behavioural modelling)

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

entity eighttothreee_case is
    Port ( i : in  STD_LOGIC_VECTOR (7 downto 0);
           o : out  STD_LOGIC_VECTOR (2 downto 0));
end eighttothreee_case;

architecture Behavioral of eighttothreee_case is

begin

Process(i)
begin
      case i is
            when "00000001" => o<="000";
            when "00000010" => o<="001";
            when "00000100" => o<="010";
            when "00001000" => o<="011";
            when "00010000" => o<="100";
            when "00100000" => o<="101";
            when "01000000" => o<="110";
            when "10000000" => o<="111";
            when others => o<="ZZZ";
      end case;
end Process;

end Behavioral;
```

ASSIGNMENT – 3

Problem

Design a Decimal-to-BCD Encoder.

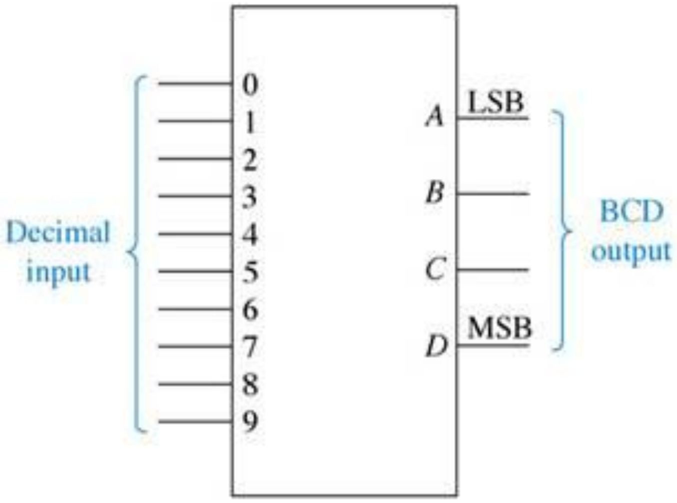
Description

An **Encoder** is a combinational circuit that performs the reverse operation of Decoder. It has maximum of 2^n input lines and 'n' output lines. It will produce a binary code equivalent to the input, which is active High. Therefore, the encoder encodes 2^n input lines with 'n' bits. It is optional to represent the enable signal in encoders. Here, the Decimal-to-BCD encoder has 10 input lines and 4 output lines.

Truth Table

Input										**Output**			
i_9	i_8	i_7	i_6	i_5	i_4	i_3	i_2	i_1	i_0	o_3	o_2	o_1	o_0
0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	0	0	0	0	1	0	0	0	0	1	0
0	0	0	0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	0	0	1	0	1
0	0	0	1	0	0	0	0	0	0	0	1	1	0
0	0	1	0	0	0	0	0	0	0	0	1	1	1
0	1	0	0	0	0	0	0	0	0	1	0	0	0
1	0	0	0	0	0	0	0	0	0	1	0	0	1

Block Diagram



VHDL Code

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

 entity decimaltoBCD_structural is
     Port ( i : in  STD_LOGIC_VECTOR (9 downto 0);
            o : out  STD_LOGIC_VECTOR (3 downto 0));
 end decimaltoBCD_structural;

architecture Structural of decimaltoBCD_structural is
 begin
     o(3) <= i(9) or i(8);
     o(2) <= i(7) or i(6) or i(5) or i(4);
     o(1) <= i(7) or i(6) or i(3) or i(2);
     o(0) <= i(9) or i(7) or i(5) or i(3) or i(1);

end Structural;
```

ASSIGNMENT – 4

Problem

Design a 1x2 Decoder using

(a) Structural design
(b) Sequential statement
(c) Concurrent statement

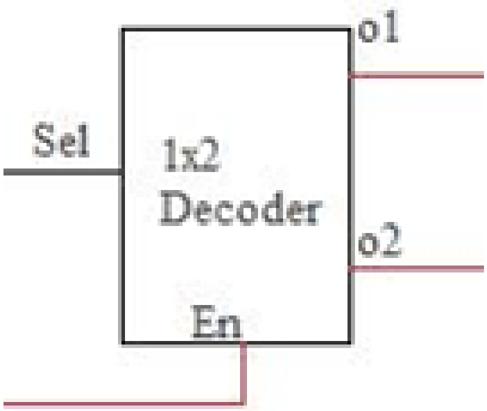
Description

Decoder is a combinational circuit that has ‘n’ input lines and maximum of 2^n output lines. One of these outputs will be active High based on the combination of inputs present, when the decoder is enabled. That means decoder detects a particular code. The outputs of the decoder are nothing but the **min terms** of ‘n’ input variables lines, when it is enabled. Here, the value of n is 1, as there are 1 input lines and 2 output lines.

Truth Table

Input		**Output**	
e	**i_0**	**o_1**	**o_0**
0	x	0	0
1	0	0	1
1	1	1	0

Block Diagram



VHDL Code

i) Using structural design

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

entity onetotwodecoder_structural is
    Port ( i : in  STD_LOGIC_VECTOR (3 downto 0);
           o : out  STD_LOGIC_VECTOR (1 downto 0));
end onetotwodecoder_structural;

architecture Structural of fourtotwoencoder_structural is
begin

    o(0)<= e and not(i);
    o(1)<= e and i;

end Structural;
```

ii) Using sequential statements (behavioural modelling)

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

entity onetotwodecoder_seq is
    Port ( e : in  STD_LOGIC;
           i : in  STD_LOGIC;
           o : out  STD_LOGIC_VECTOR (1 downto 0));
end onetotwodecoder_seq;

architecture Behavioral of onetotwodecoder_seq is
begin

process(e,i)
begin

       if(e='0') then
             o<="00";
      elsif(i='0') then
            o<="01";
      elsif(i='1') then
            o<="10";
      end if;

end process;
end Behavioral;
```

iii) Using concurrent statements (behavioural modelling)

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

```
 entity onetotwodecoder_con is
     Port ( e : in  BIT;
            i : in  BIT;
            o : out  BIT_VECTOR (1 downto 0));
 end onetotwodecoder2;

architecture Behavioral of onetotwodecoder2 is

 begin
      with (e & i) select o<=
       "01" when "10",
       "10" when "11",
       "00" when others;
 end Behavioral;
```

ASSIGNMENT – 5

Problem

Design a 2x4 Decoder using

(a) Structural design
(b) Sequential statement
(c) Concurrent statement

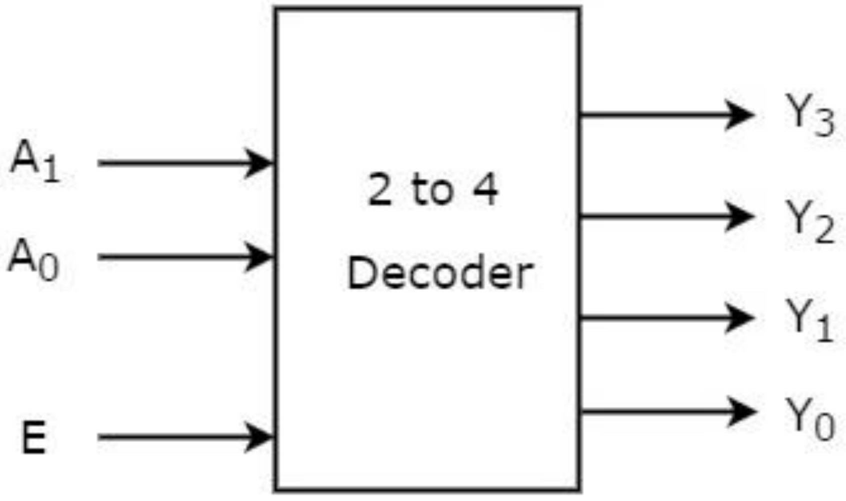
Description

Decoder is a combinational circuit that has ‘n’ input lines and maximum of 2^n output lines. One of these outputs will be active High based on the combination of inputs present, when the decoder is enabled. That means decoder detects a particular code. The outputs of the decoder are nothing but the **min terms** of ‘n’ input variables lines, when it is enabled. Here, the value of n is 2, as there are 2 input lines and 4 output lines.

Truth Table

Input			Output			
e	**i_1**	**i_0**	**o_3**	**o_2**	**o_1**	**o_0**
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

Block Diagram



VHDL Code

i) Using structural design

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

entity twotofourdecoder is
    Port ( e : in  BIT;
           i : in  BIT_VECTOR (1 downto 0);
           o : out  BIT_VECTOR (3 downto 0));
end twotofourdecoder;

architecture Structural of twotofourdecoder is

begin

      o(0)<= e and not(i(1)) and not (i(0));
      o(1)<= e and not(i(1)) and (i(0));
      o(2)<= e and(i(1)) and not (i(0));
      o(3)<= e and (i(1)) and (i(0));


end Structural;
```

ii) Using sequential statements (behavioural modelling)

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

entity twotofourdecoder_seq is
    Port ( e : in  STD_LOGIC;
           i : in  STD_LOGIC_VECTOR (1 downto 0);
           o : out  STD_LOGIC_VECTOR (3 downto 0));
end twotofourdecoder_seq;

architecture Behavioral of twotofourdecoder_seq is
begin

process(e,i)
begin

      if(e='0') then
            o<="0000";
      elsif(i="00") then
            o<="0001";
      elsif(i="01") then
            o<="0010";
      elsif(i="10") then
            o<="0100";
      elsif(i="11") then
            o<="1000";
      end if;
end process;
end Behavioral;
```

iii) Using concurrent statements (behavioural modelling)

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

entity twotofourdecoder_co is
    Port ( e : in  STD_LOGIC;
           i : in  STD_LOGIC_VECTOR (1 downto 0);
           o : out  STD_LOGIC_VECTOR (3 downto 0));
end twotofourdecoder_co;

architecture Behavioral of twotofourdecoder_co is
begin

with (e & i) select o<=
      "0001" when "100",
      "0010" when "101",
      "0100" when "110",
      "1000" when "111",
      "0000" when others;

end Behavioral;
```

ASSIGNMENT – 6

Problem

Design a 3x8 Decoder

(a) By component instantiate

(b) By using procedural statement

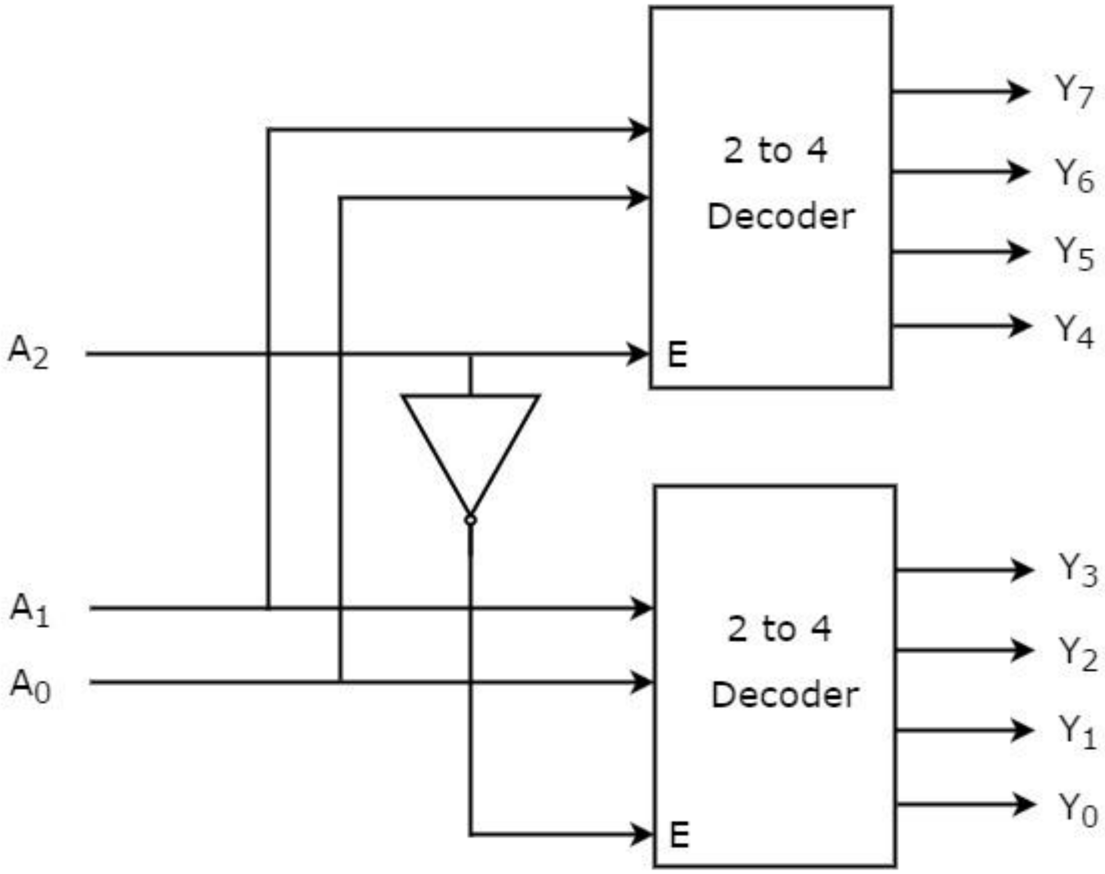
Description

Decoder is a combinational circuit that has ‘n’ input lines and maximum of 2^n output lines. One of these outputs will be active High based on the combination of inputs present, when the decoder is enabled. That means decoder detects a particular code. The outputs of the decoder are nothing but the **min terms** of ‘n’ input variables lines, when it is enabled. Here, the value of n is 3, as there are 3 input lines and 8 output lines.

Truth Table

Input				**Output**							
e	**i_2**	**i_1**	**i_0**	**o_7**	**o_6**	**o_5**	**o_4**	**o_3**	**o_2**	**o_1**	**o_0**
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

Block Diagram



VHDL Code

i) By component instantiate

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

entity threeto8dec_comp is
    Port ( inp : in  STD_LOGIC_VECTOR (2 downto 0);
                    en : in  STD_LOGIC;
           op : out  STD_LOGIC_VECTOR (7 downto 0));
end threeto8dec_comp;

architecture Behavioral of threeto8dec_comp is

component twotofourdecoder is
    Port ( e : in  STD_LOGIC;
           i : in  STD_LOGIC_VECTOR (1 downto 0);
           o : out  STD_LOGIC_VECTOR (3 downto 0));
end component;
signal notinp:STD_LOGIC;
begin
    notinp <= not inp(2);
   dec1:twotofourdecoder port map(inp(2),inp(1 downto 0),op(7 downto 4));
   dec2:twotofourdecoder port map(notinp,inp(1 downto 0),op(3 downto 0));
end Behavioral;
```

ii) By using procedural statement

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

entity threeto8dec_proc is
    Port ( inp : in  STD_LOGIC_VECTOR (2 downto 0);
           op : out  STD_LOGIC_VECTOR (7 downto 0));
end threeto8dec_proc;
architecture Behavioral of threeto8dec_proc is

procedure twotofourdecoder ( e : in   STD_LOGIC;
           i : in  STD_LOGIC_VECTOR (1 downto 0);
           o : out  STD_LOGIC_VECTOR (3 downto 0)) is

begin

      with (e & i) select o:=
            "0001" when "100",
            "0010" when "101",
            "0100" when "110",
            "1000" when "111",
            "0000" when others;

end procedure;

begin

      process(inp)
      variable varop:STD_LOGIC_VECTOR (7 downto 0);
      begin
      dec1:twotofourdecoder(inp(2),inp(1 downto 0),varop(7 downto 4));
      dec2:twotofourdecoder(not inp(2),inp(1 downto 0),varop(3 downto
0));
      op<=varop;
      end process;

end Behavioral;
```