CONESTOGA
Connect Life and Learning

DIGITAL DESIGN PRINCIPLES

LAB 4 – OCTAL DECODER

Name & Surname : Sinan KARACA

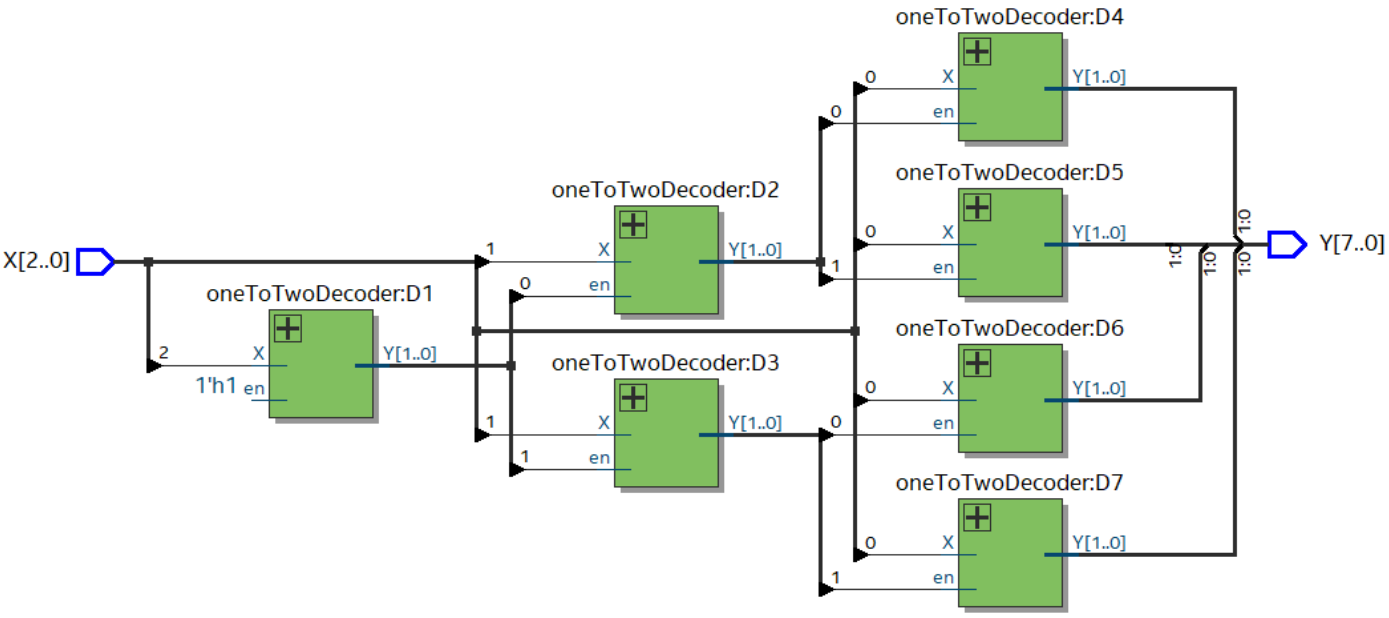
Date : 28 / 05 / 2021

Student ID : 8743013

INTRODUCTION

The purpose of the project is to design 3 to 8 Octal decoder, 7 amount of 1 to 2 Octal decoders are implemented. In the main file, 3 sized STD_LOGIC_VECTOR, 8 sized STD_LOGIC_VECTOR has been used. Inside the 1 to 2 decoder modules, 1 input, 1 enable and 2 sized output has been used.

RTL VIEWER



As it can be seen from the above diagram, first decoder has been used to select D2 or D3 and also D2 and D3 used to select D4,D5,D6 and D7. As the decoders output are 2 sized array, output signals has been added to create Y output signal.

VHDL CODE

```
----------------------------------------
----------------------------------------
-- Project Name  : 3to 8 Octal Decoder
-- File          : skaraca_lab4
-- Creator       : Sinan KARACA
-- Student number: 8743013
-- Date          : 17.06.2021
----------------------------------------
----------------------------------------

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity skaraca_lab4 is
    port(
        X : in STD_LOGIC_VECTOR(2 downto 0);
        Y : out STD_LOGIC_VECTOR(7 downto 0));
    end skaraca_lab4;


architecture thereToEightDecoder of skaraca_lab4 is


component oneToTwoDecoder is
    port(
        X  : in STD_LOGIC;
        en : in STD_LOGIC;
        Y  : out STD_LOGIC_VECTOR(1 downto 0));
    end component oneToTwoDecoder;

signal s1,s2,s3,so1,so2,so3,so4 : STD_LOGIC_VECTOR(1 downto 0);


begin

    D1: oneToTwoDecoder port map (x(2), '1', s1);
    D2: oneToTwoDecoder port map (x(1), s1(0),s2);
    D3: oneToTwoDecoder port map (x(1), s1(1),s3);
    D4: oneToTwoDecoder port map (x(0), s2(0),so1);
    D5: oneToTwoDecoder port map (x(0), s2(1),so2);
    D6: oneToTwoDecoder port map (x(0), s3(0),so3);
    D7: oneToTwoDecoder port map (x(0), s3(1),so4);

    Y <= so4 & so3 & so2 & so1;

end thereToEightDecoder;
```

VHDL COMPONENT CODE

```
----------------------------------------
----------------------------------------
-- Project Name   : 3to 8 Octal Decoder
-- File           : oneToTwodecoder
-- Creator        : Sinan KARACA
-- Student number: 8743013
-- Date           |: 17.06.2021
----------------------------------------
----------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity oneToTwoDecoder is
    port(
        X  : in STD_LOGIC;
        en : in STD_LOGIC;
        Y  : out STD_LOGIC_VECTOR(1 downto 0));
    end oneToTwoDecoder;


architecture Behavioral of oneToTwoDecoder is

    begin
        process(X)
            begin

            if (en = '1') then
                case X is

                    when '0' => Y <= "01";
                    when '1' => Y <= "10";

                end case;
            else

                Y <= "00";

            end if;

        end process;
end Behavioral;
```

VERILOG COMPONENT CODE

```
//////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////
// Project Name   : 3to 8 Octal Decoder
// File           : oneToTwoDecoder
// Creator        : Sinan KARACA
// Student number: 8743013
// Date           : 17.06.2021
//////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////

module oneToTwoDecoder(X, En, Y);
    input [1: 0]X, En;
    output [1:0] Y;

    always @(X) begin
        case(X&En)

            3'b000: begin Y = 2'b00; end
            3'b010: begin Y = 2'b00; end
            3'b001: begin Y = 2'b01; end
            3'b011: begin Y = 2'b10; end

        endcase
    end
endmodule
```

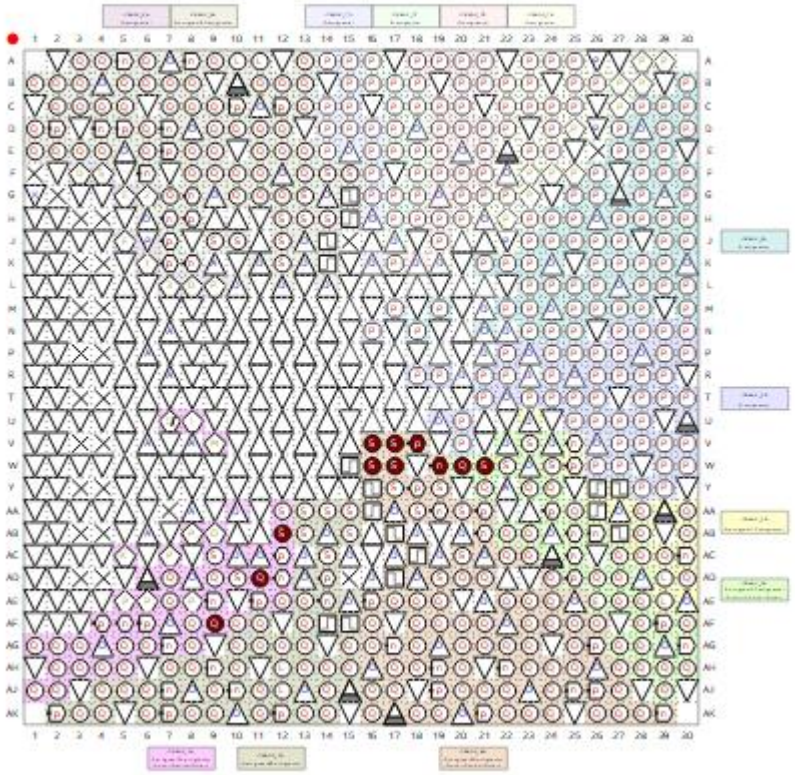
VERILOG CODE

```
//////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////
// Project Name  : 3to 8 Octal Decoder
// File          : skaraca_lab4
// Creator       : Sinan KARACA
// Student number: 8743013
// Date          : 17.06.2021
//////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////

module skaraca_lab4(X, Y);
    input [2:0]X;
    output [7:0] Y;

    wire [1:0]s1;
    wire [1:0]s2;
    wire [1:0]s3;
    wire [1:0]so1;
    wire [1:0]so2;
    wire [1:0]so3;
    wire [1:0]so4;

    oneToTwoDecoder(X[2], 1'b1 ,  s1);
    oneToTwoDecoder(X[1], s1[0],  s2);
    oneToTwoDecoder(X[1], s1[1],  s3);
    oneToTwoDecoder(X[0], s2[0], so1);
    oneToTwoDecoder(X[0], s2[1], so2);
    oneToTwoDecoder(X[0], s3[0], so3);
    oneToTwoDecoder(X[0], s3[1], so4);

    assign Y = so4 & so3 & so2 & so1;


endmodule
```

PIN ASSIGNMENT

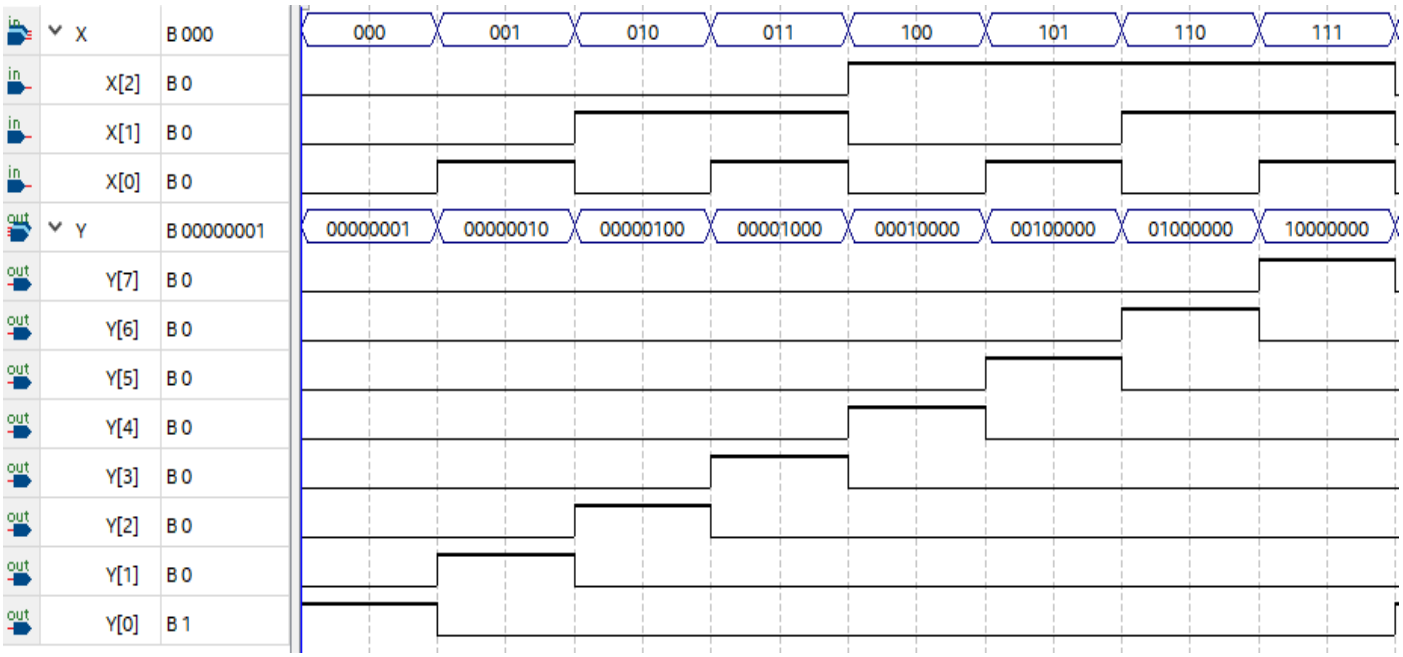
Top View - Wire Bond

Cyclone V - 5CSEMA5F31C6



X[2]	Input	PIN_AB12	3A	B3A_N0	PIN_AB12	2.5 V		12mA (default)	
X[1]	Input	PIN_AF9	3A	B3A_N0	PIN_AF9	2.5 V		12mA (default)	
X[0]	Input	PIN_AD11	3A	B3A_N0	PIN_AD11	2.5 V		12mA (default)	
Y[7]	Output	PIN_V16	4A	B4A_N0	PIN_V16	2.5 V		12mA (default)	1 (default)
Y[6]	Output	PIN_W16	4A	B4A_N0	PIN_W16	2.5 V		12mA (default)	1 (default)
Y[5]	Output	PIN_V17	4A	B4A_N0	PIN_V17	2.5 V		12mA (default)	1 (default)
Y[4]	Output	PIN_V18	4A	B4A_N0	PIN_V18	2.5 V		12mA (default)	1 (default)
Y[3]	Output	PIN_W17	4A	B4A_N0	PIN_W17	2.5 V		12mA (default)	1 (default)
Y[2]	Output	PIN_W19	4A	B4A_N0	PIN_W19	2.5 V		12mA (default)	1 (default)
Y[1]	Output	PIN_W20	5A	B5A_N0	PIN_W20	2.5 V		12mA (default)	1 (default)
Y[0]	Output	PIN_W21	5A	B5A_N0	PIN_W21	2.5 V		12mA (default)	1 (default)

SIMALATION WAVEFORM



CONCLUSION

3 to 8 decoder has been designed in both VHDL and Verilog languages. 1 to 2 decoder components used as subcomponents. In this way, splitting different components makes the implementation easier for further use. Usage of subcomponents are more usable than basic implementation. To design more inputs and outputs decoders, the subsystems that we created in this experiment can be used.