Hacettepe University

Computer Engineering Department

BM233 Logic Design Lab - 2020 Fall

Experiment 4 - Combinational Circuits in Verilog

December 5, 2020

Student name
İlayda Atmaca

Student Number:
b21827101

1 Include the problem statement with figures of combinational circuits that are being implemented.

In this experiment, we aim to Designing and simulating combinational circuits in Verilog HDL. In the problem, A decoder is a combinational circuit that converts binary information from n binary inputs to a maximum of (2^n) unique output lines.

Decoders with enable inputs can be connected together to form a larger decoder circuit. We have obtained a 4x16 decoder.

2 Include four Verilog code solutions for all specified modules

```
module decoder_3x8(a,b,c,en,d0,d1,d2,d3,d4,d5,d6,d7);
    // Declare input and output ports
    input a,b,c;
    output d0,d1,d2,d3,d4,d5,d6,d7;
    input en;
 assign d0=(en& ~a&~b&~c);
 assign d1=(en& ~a&~b&c);
 assign d2=(en& ~a&b&~c);
 assign d3=(en& ~a&b&c);
 assign d4=(en& a&~b&~c);
 assign d5=(en& a&~b&c);
 assign d6=(en& a&b&~c);
 assign d7=(en& a&b&c);
endmodule
```

```

module decoder_3x8_tb;

reg a,b,c; //inputs
reg en;    //enable input
wire [7:0]d;  //outputs
integer i;

decoder_3x8 Decoder(.a(a),.b(b),.c(c),.en(en),.d0(d[0]),.d1(d[1]),.d2(d[2])
,.d3(d[3]),.d4(d[4]),.d5(d[5]),.d6(d[6]),.d7(d[7]));

initial begin
a=1'b0;b=1'b0;c=1'b0;en=1'b0;
#100 a=1'b1;b=1'b0;c=1'b0;en=1'b0;
for(i=0;i<8;i=i+1)begin
#100
{a,b,c} = i; en=1'b1;
end
#100 $finish;
  end
```

```
endmodule
```

```
`include "decoder_3x8.v"
module decoder_4x16(
    input x,y,z,w,
    output [15:0]d
    );
    decoder_3x8 dec1(.a(x),.b(y),.c(z),.en(~w),.d0(d[0]),.d1(d[1]),.d2(d[2]),.d3(d[3]
    ,.d4(d[4]),.d5(d[5]),.d6(d[6]),.d7(d[7]));

    decoder_3x8 dec2(.a(x),.b(y),.c(z),.en(w),.d0(d[8]),.d1(d[9]),.d2(d[10])
    ,.d3(d[11]),.d4(d[12]),.d5(d[13]),.d6(d[14]),.d7(d[15]));

endmodule
```

```

module decoder_4x16_tb;
    reg x,y,z,w;  //inputs
    wire [15:0]d; //outputs
    integer i;

    decoder_4x16 de1(.x(x),.y(y),.z(z),.w(w),.d(d));

    initial begin
    x=1'b0;y=1'b0;z=1'b0;w=1'b0;
    for(i=1;i<16;i=i+1)begin
    #100{w,x,y,z} = i;
    end
    #100 $finish;

  end
endmodule
```

3 Include two waveforms obtained from both parts: test results for 3x8 decoder with enable, and 4x16 decoder.

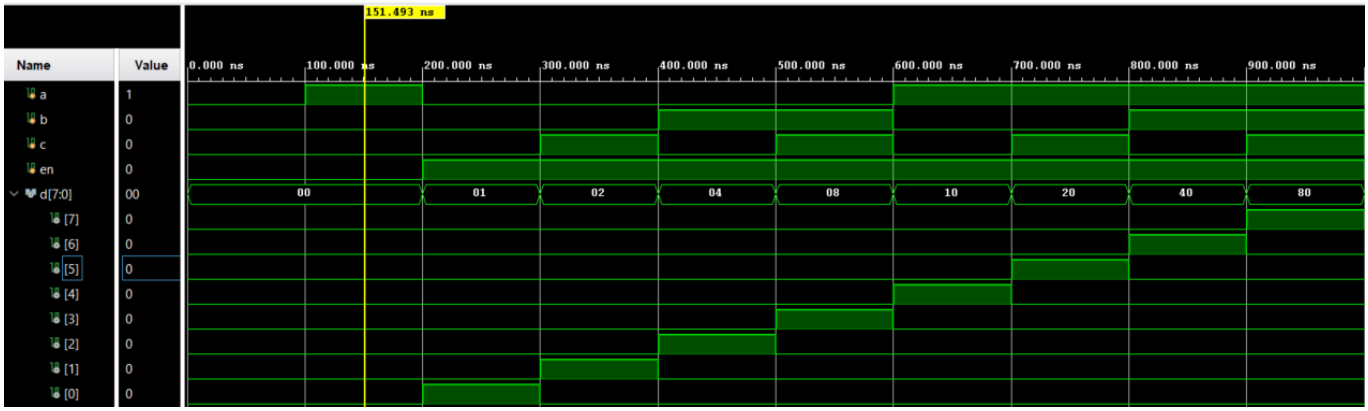

Figure 1: 3x8 decoder

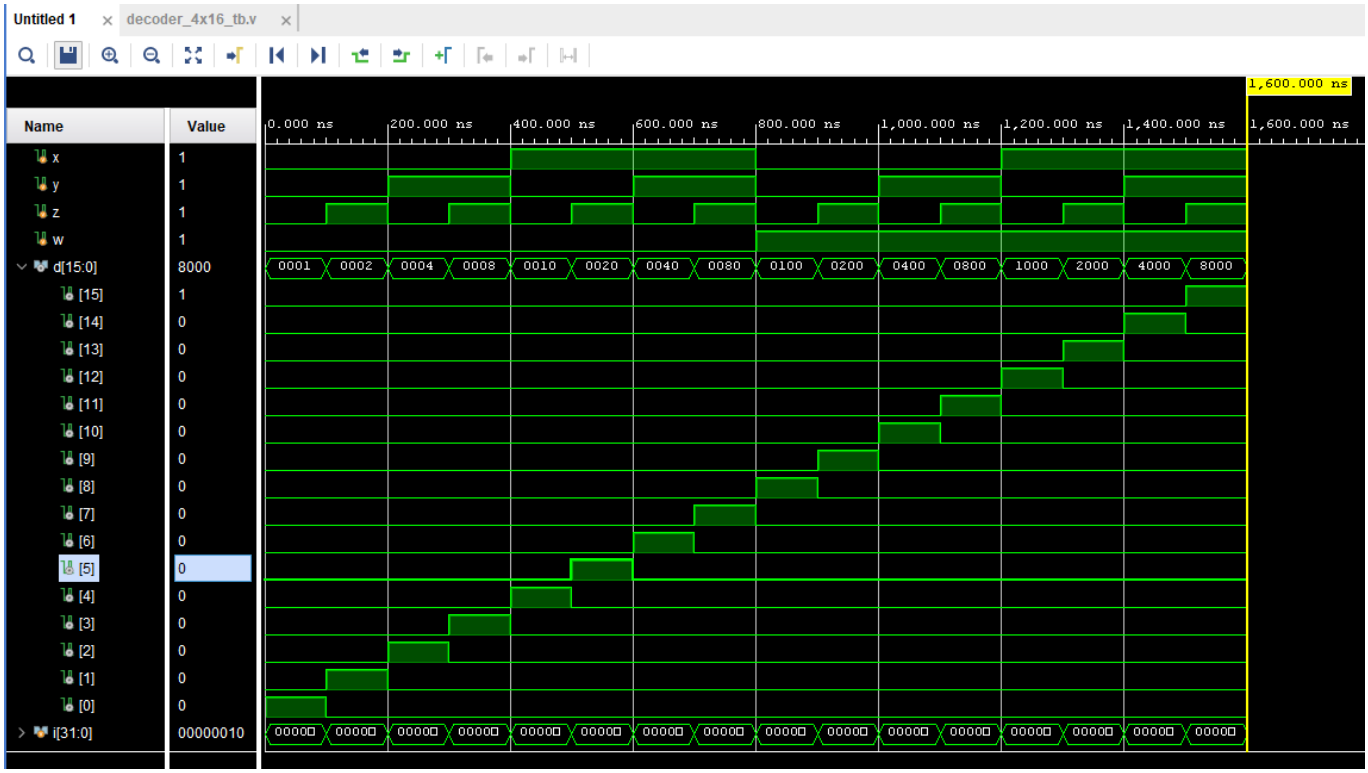

Figure 2: 4x16 decoder

4 Explain the obtained results and how they show that your designs are working correctly. You may indicate the results directly on the waveforms.

First, while trying to design the 3x8 decoder with enable input and implement it in Verilog HDL, I obtained the boolean equation of the decoder from the truth table. and I also got information on how enable input affects the equation.

As a result, I also saw in my waveform(Figure 1) that all outputs are ineffective when enable is 0. Also, when I looked at the value of my outputs in my waveform, I realized that my inputs had the correct values.

In my 4x16 decoder, I tried to get it using 3x8 decoders and variable enable value and as a result I saw in the Waveform (as shown in the figure 2) that When enable is 0, the top decoder (D0 to D7) is enabled and the other (D8 to D15) is disabled. When enable is 1, the enable conditions are reversed. At the same time, when I looked at the value of my outputs in the waveform with the boolean equation I obtained, I saw that my inputs had the correct values.

References

- Reference 1: `https://www.tutorialspoint.com/digital_circuits/digital_circuits_decoders.htm`