

Lab 2
Four to Sixteen Decoder

Group G
Chang, Philbert (013179257)
Hua, Russell (013184015)
Thai, Paul (014760252)

ECE 3300L.01
Professor Aly

06/28/2021

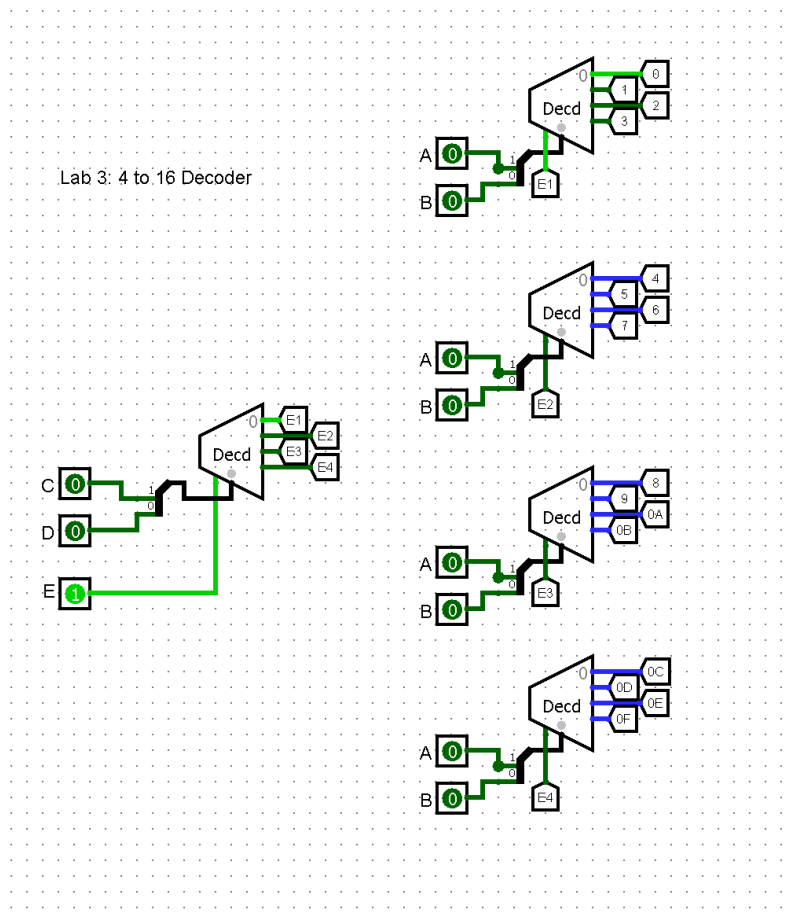
Problem

Create a 4 to 16 decoder using the gate level implementation and demonstrate on an FPGA board.

Code Detail

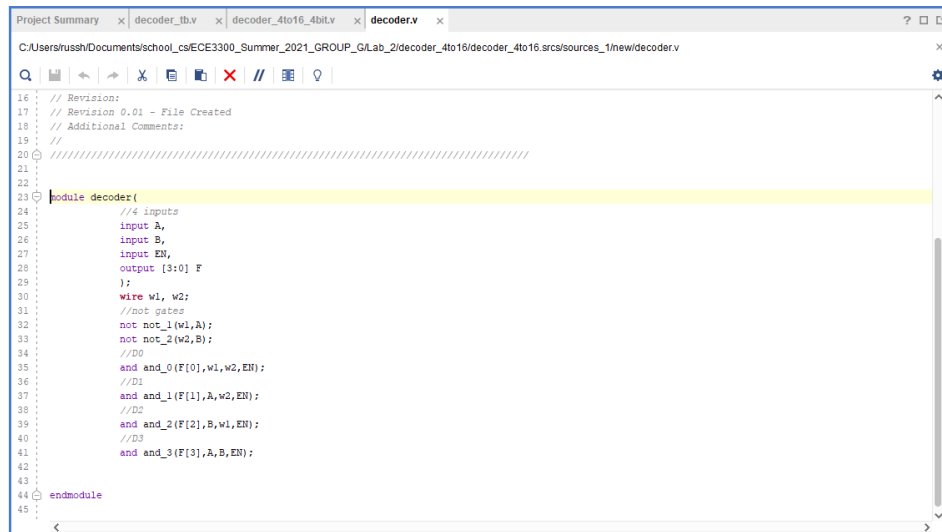
The Verilog code uses 5, 2 to 4 decoders built from logic gates to implement a 4 to 16 decoder that turns on a certain LED when triggered on the FPGA - binary. The two lower bits of the input are internally wired to 4 decoders, while the fifth decoder uses the higher two bits to control which 4 decoders are active based on the fifth decoder's output. The four decoders output is then represented as the output of the decoder via sixteen LEDs.

Schematic



(Input D is MSB, A is LSB)

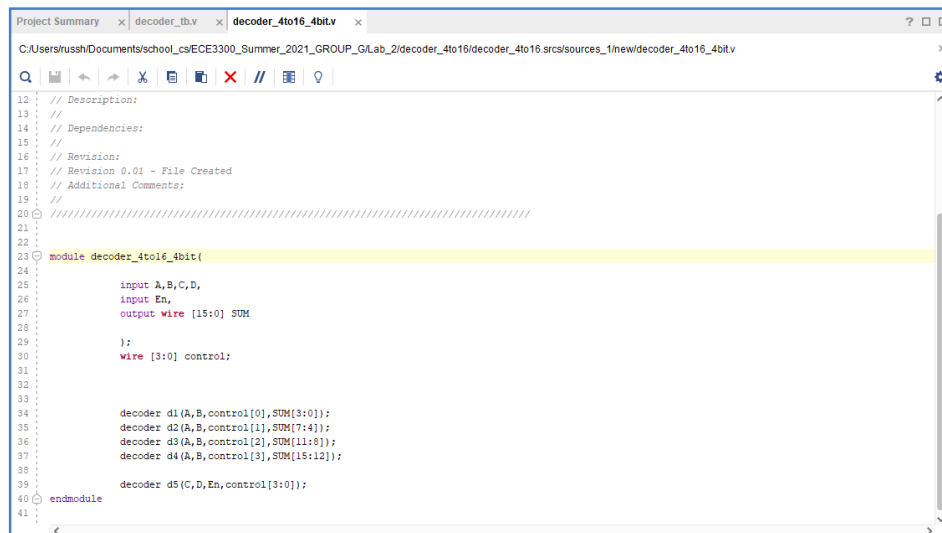
Results



```
Project Summary | decoder_tb.v | decoder_4to16_4bit.v | decoder.v |
C:/Users/rush/Documents/school_cs/ECE3300_Summer_2021_GROUP_G/Lab_2/decoder_4to16/srcs/sources_1/new/decoder.v

16 // Revision:
17 // Revision 0.01 - File Created
18 // Additional Comments:
19 //
20 ///////////////////////////////////////////////////////////////////
21
22
23 module decoder(
24     //4 inputs
25     input A,
26     input B,
27     input EN,
28     output [3:0] F
29 );
30     wire w1, w2;
31     //not gates
32     not not_1(w1,A);
33     not not_2(w2,B);
34     //D0
35     and and_0(F[0],w1,w2,EN);
36     //D1
37     and and_1(F[1],A,w2,EN);
38     //D2
39     and and_2(F[2],B,w1,EN);
40     //D3
41     and and_3(F[3],A,B,EN);
42
43
44 endmodule
45
```

Figure 1: Verilog Code for 2 to 4 Decoder



```
Project Summary | decoder_tb.v | decoder_4to16_4bit.v |
C:/Users/rush/Documents/school_cs/ECE3300_Summer_2021_GROUP_G/Lab_2/decoder_4to16/srcs/sources_1/new/decoder_4to16_4bit.v

12 // Description:
13 //
14 // Dependencies:
15 //
16 // Revision:
17 // Revision 0.01 - File Created
18 // Additional Comments:
19 //
20 ///////////////////////////////////////////////////////////////////
21
22
23 module decoder_4to16_4bit(
24     input A,B,C,D,
25     input En,
26     output wire [15:0] SUM
27 );
28
29     wire [3:0] control;
30
31
32
33     decoder d1(A,B,control[0],SUM[3:0]);
34     decoder d2(A,B,control[1],SUM[7:4]);
35     decoder d3(A,B,control[2],SUM[11:8]);
36     decoder d4(A,B,control[3],SUM[15:12]);
37
38     decoder d5(C,D,En,control[3:0]);
39
40 endmodule
41
```

Figure 2: Verilog Code for 4 to 16 Decoder

```

23 module decoder4to16bit_tb();
24
25     reg [3:0] a_tb;
26     reg en_tb;
27
28     wire [15:0] out;
29
30     decoder4to16bit X (.A(a_tb[0]), .B(a_tb[1]), .C(a_tb[2]), .D(a_tb[3]), .En(en_tb), .SUN(out));
31
32     initial
33     begin
34         a_tb = 4'd0;
35         en_tb = 1;
36         #10
37         a_tb = 4'd1;
38         en_tb = 1;
39         #10
40         a_tb = 4'd2;
41         en_tb = 1;
42         #10
43         a_tb = 4'd3;
44         en_tb = 1;
45         #10
46         a_tb = 4'd4;
47         en_tb = 1;
48         #10
49         a_tb = 4'd5;
50         en_tb = 1;
51         #10
52         a_tb = 4'd6;
53         en_tb = 1;
54     end
55 endmodule

```

Figure 3: Testbench for the First Portion of Code

```

103     a_tb = 4'd7;
104     en_tb = 0;
105     #10
106     a_tb = 4'd8;
107     en_tb = 0;
108     #10
109     a_tb = 4'd9;
110     en_tb = 0;
111     #10
112     a_tb = 4'd10;
113     en_tb = 0;
114     #10
115     a_tb = 4'd11;
116     en_tb = 0;
117     #10
118     a_tb = 4'd12;
119     en_tb = 0;
120     #10
121     a_tb = 4'd13;
122     en_tb = 0;
123     #10
124     a_tb = 4'd14;
125     en_tb = 0;
126     #10
127     a_tb = 4'd15;
128     en_tb = 0;
129     #100
130     $finish;
131 end
132 endmodule

```

Figure 4: Testbench End-portion of Code

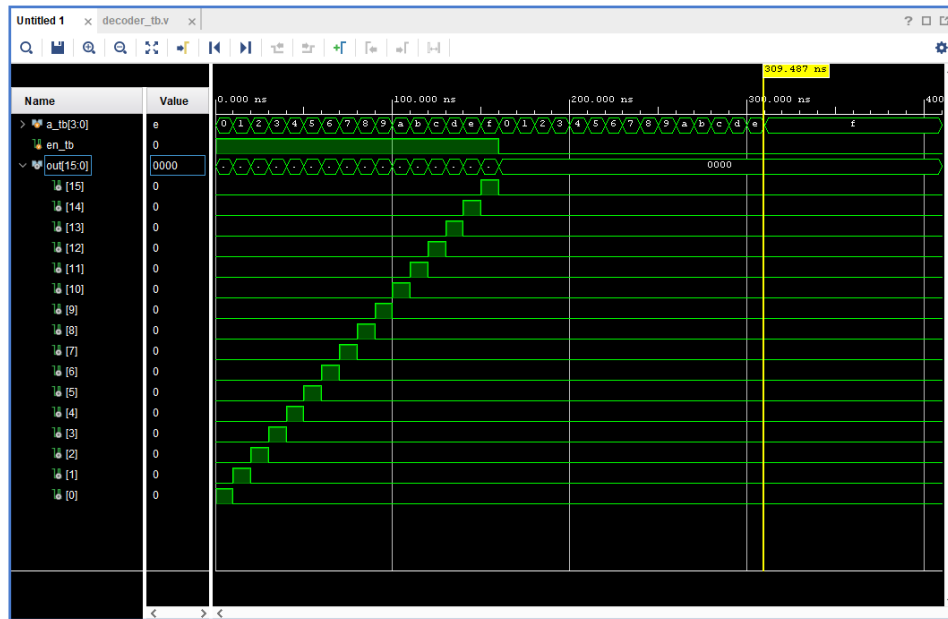


Figure 5: Behavioral simulation

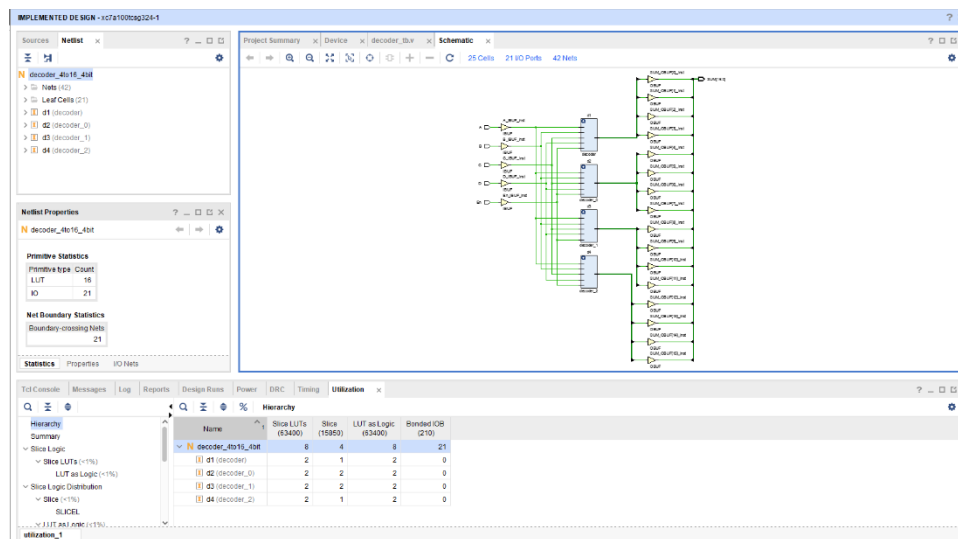


Figure 6: Utilization of Resources

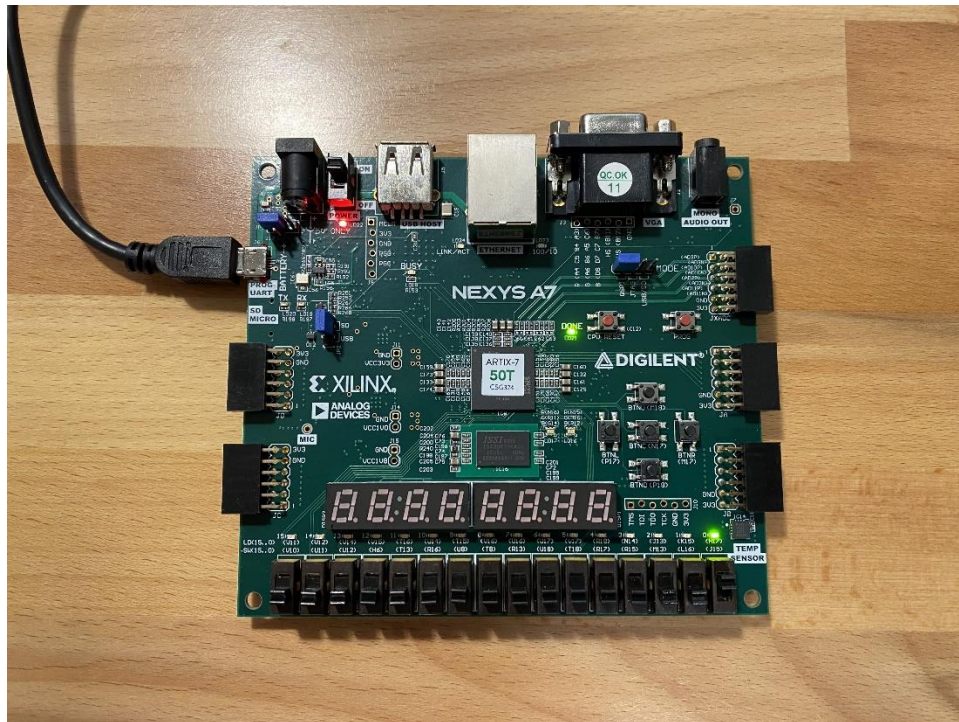


Figure 7: Enable Pin is High and the LED at 0 is ON

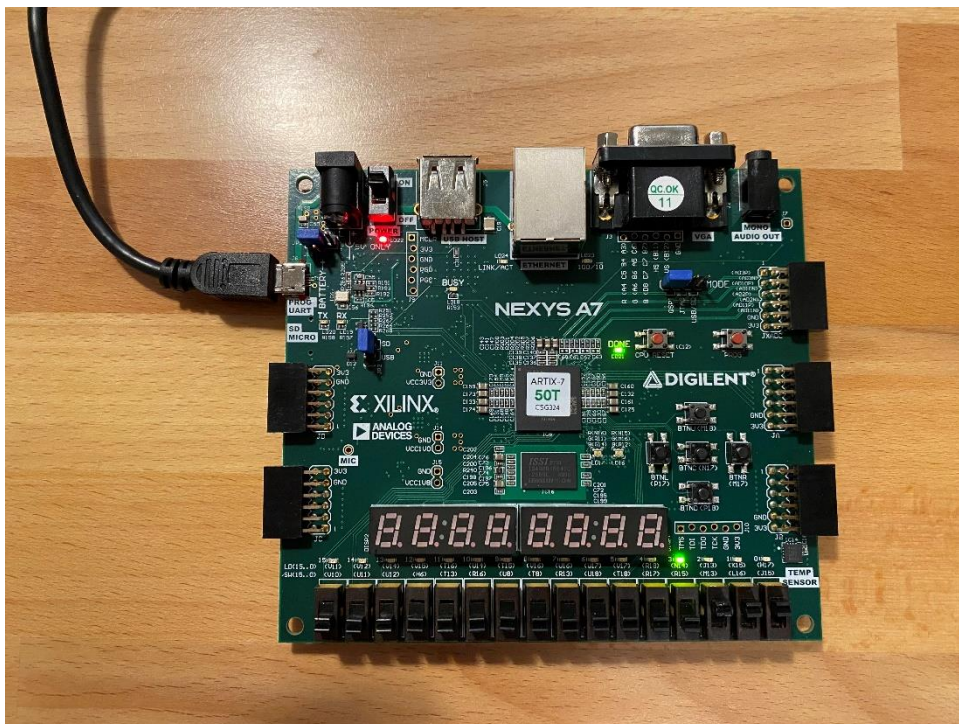


Figure 8: Enable Pin is High and the LED at 3 is ON

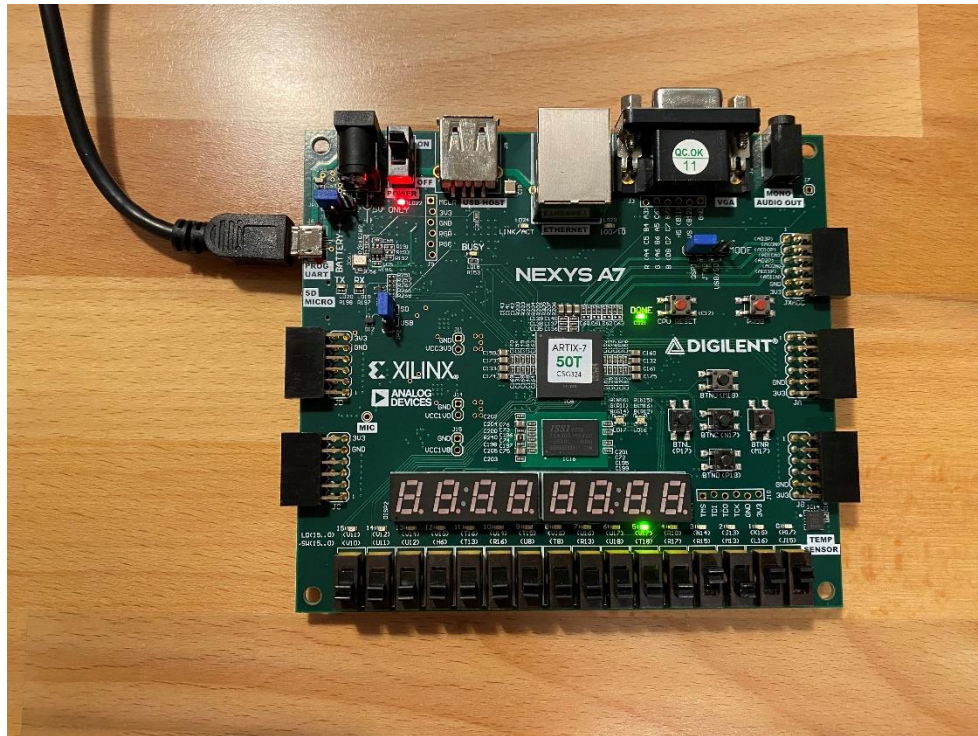


Figure 9: Enable Pin is High and the LED at 5 is ON

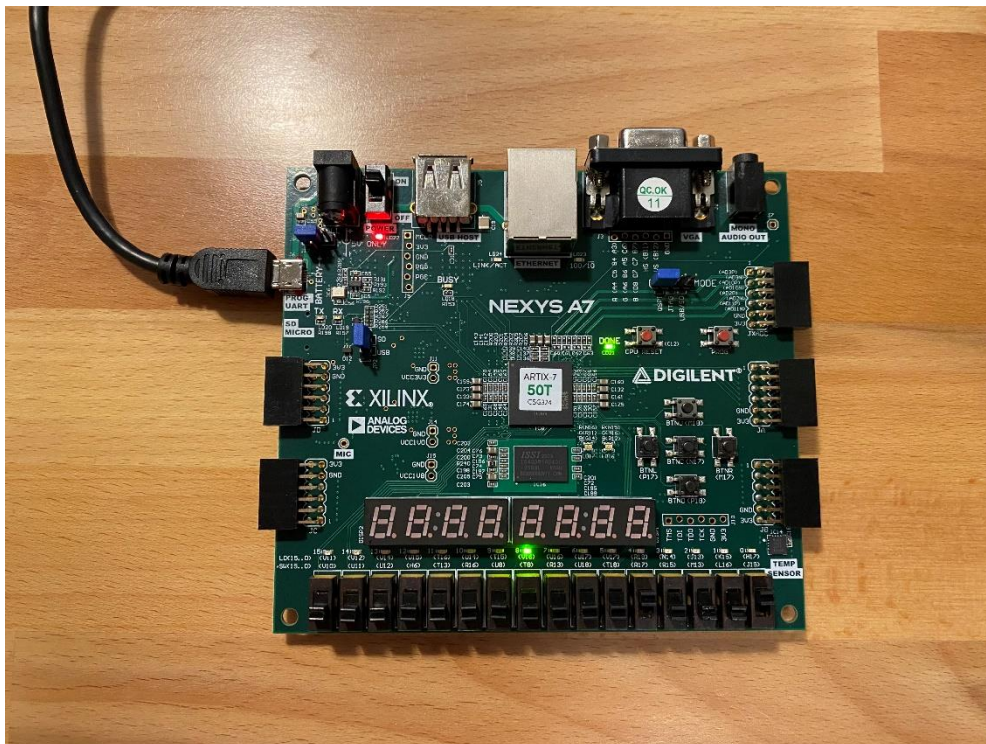


Figure 10: Enable Pin is High and the LED at 8 is ON

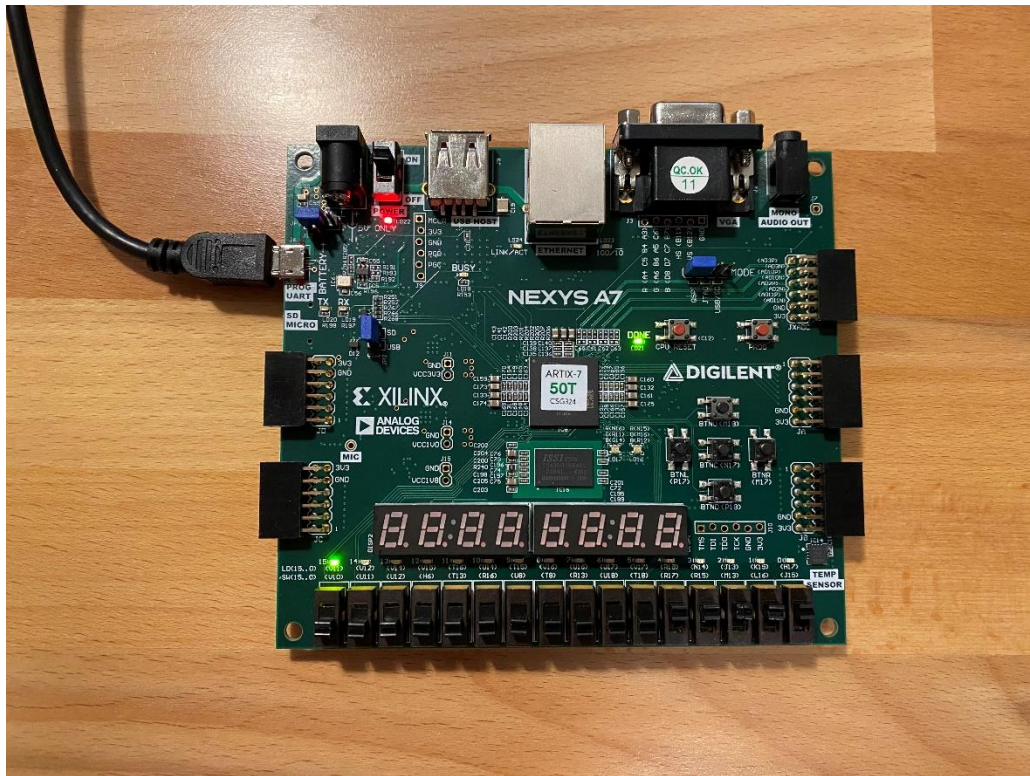


Figure 11: Enable Pin is High and the LED at 15 is ON

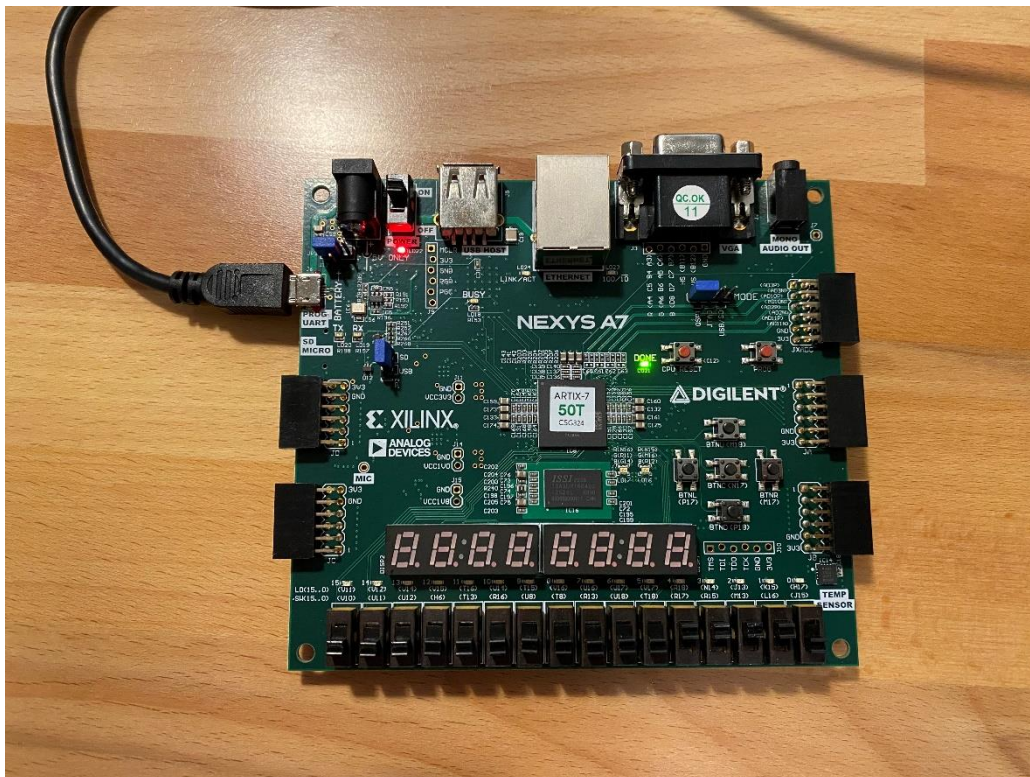


Figure 12: Enable Pin is Low and all LEDs are OFF

Discussion

The simulation displays the expected output for the decoder. When the enable is high, the correct decoder output is displayed based on the input. The implementation on the FPGA board also behaves as expected. The switches cause the correct LEDs to light without any issue. Each possible combination of switches is accounted for in the simulation and there aren't any edge cases we can find.

Conclusion

The implementation that we used works well, without any bugs that we could find. The implementation may be improved by using other types of modeling, such as dataflow modeling or behavioral modeling. The complexity of the current implementation may also be reduced by using other methods of modeling the desired behavior.

Work Distribution

Russell: Lab Report, Video and Photos regarding the FPGA, Source Code and Testbench.

Philbert: Lab Report, Video and Photos regarding the FPGA, Source Code and Testbench.

Paul: Lab Report, Video and Photos regarding the FPGA, Source Code and Testbench.