**ELEC 204 Digital Design Lab Report**

Lab 02

Name: Umur Demircioğlu

Date: 27/10/2019

1. **Introduction and objectives**

We designed a 4 to 2 encoder, 2 to 4 decoder, 4 to 1 multiplexer and 1 to 4 de-multiplexer. We designed them all in separate codes with the help of the truth tables and by using simple if else statements.

1. **Methods**

We designed encoder, decoder, multiplexer and de-multiplexer all in different project folders. First

for the encoder with the help of the truth table we wrote the if statements. In encoder (a) was our 4 bit

input which were assigned to 4 pins in our FPGA board and (b) was our 2 bit output which was assigned

to 2 LEDS. When we entered the inputs that are in our truth table corresponding LEDS lighted up. The

same strategy worked for the decoder just the reverse our input was 2 bit and output was 4 bit and again

inputs were pins and outputs were LEDS. For the multiplexer we had 1 output that lighted up according

to the 2 select inputs and the 1 of the 4 other inputs. For the de-multiplexer it is the reverse of

multiplexer we had 4 outputs A, B, C , D which were the inputs in multiplexer and 2 select inputs and

F input. A, B, C, D lighted up according to 2 selects inputs and F. We controlled every stage in this lab

by simply writing if else statements.

1. **Problems encountered, errors and warnings resolved**

In this lab I had no problems with the code because the logic behind it was simple however I had problems while uploading the bit file to the board.

1. **Conclusion**

In this experiment we got used to using Xilinx ISE software, Prometheus and the FPGA board. We learned more about encoders, decoders. multiplexer and de-multiplexer and how to implement them to the board.

4 to 2 Encoder

Inputs:

|  |
| --- |
| a : in STD\_LOGIC\_VECTOR(3 downto 0); |
| Outputs:   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | ../../../../Downloads/WhatsApp%20Image%202019-10-27%20at%2013.38.18.jpeg b : out STD\_LOGIC\_VECTOR(1 downto 0);  Code:   |  | | --- | | process(a) |  |  | | --- | | begin |  |  | | --- | | if (a="0001") then |  |  | | --- | | b <= "00"; |  |  | | --- | | elsif (a="0010") then |  |  | | --- | | b <= "01"; |  |  | | --- | | elsif (a="0100") then |  |  | | --- | | b <= "10"; |  |  | | --- | | elsif (a="1000") then |  |  | | --- | | b <= "11"; |  |  | | --- | | end if; |  |  | | --- | | end process; |   2 to 4 Decoder  ../../../../Downloads/WhatsApp%20Image%202019-10-27%20at%2013.37.00.jpeg Inputs:   |  | | --- | | a : in STD\_LOGIC\_VECTOR(1 downto 0); | | Outputs:   |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | b : out STD\_LOGIC\_VECTOR(3 downto 0);  Code:   |  | | --- | | process(a) |  |  | | --- | | begin |  |  | | --- | | if (a="00") then |  |  | | --- | | b <= "0001"; |  |  | | --- | | elsif (a="01") then |  |  | | --- | | b <= "0010"; |  |  | | --- | | elsif (a="10") then |  |  | | --- | | b <= "0100"; |  |  | | --- | | else |  |  | | --- | | b <= "1000"; |  |  | | --- | | end if; |  |  | | --- | | end process; | | |   4 to 1 Multiplexer  ../../../../Downloads/WhatsApp%20Image%202019-10-27%20at%2013.37.52.jpeg Inputs:   |  |  |  | | --- | --- | --- | | |  | | --- | | A,B,C,D : in STD\_LOGIC; |  |  | | --- | | S0,S1: in STD\_LOGIC; | | | Outputs:   |  | | --- | | Z: out STD\_LOGIC; | |  | | Code:   |  | | --- | | process (A,B,C,D,S0,S1) is |  |  | | --- | | begin |  |  | | --- | | if (S0 ='0' and S1 = '0') then |  |  | | --- | | Z <= A; |  |  | | --- | | elsif (S0='1' and S1= '0') then |  |  | | --- | | Z <= B; |  |  | | --- | | elsif (S0='0' and S1= '1') then |  |  | | --- | | Z <= C; |  |  | | --- | | else |  |  | | --- | | Z <= D; |  |  | | --- | | end if; | | |   ../../../../Downloads/WhatsApp%20Image%202019-10-27%20at%2013.37.29.jpeg  1 to 4 Demultiplexer  Inputs:   |  |  |  |  | | --- | --- | --- | --- | | |  | | --- | | F : in STD\_LOGIC; |  |  | | --- | | S0,S1: in STD\_LOGIC; | | | |  | | Outputs:  A,B,C,D: out STD\_LOGIC   |  | | --- | | Code: | | |  | | --- | | begin |  |  | | --- | | process (F,S0,S1) is |  |  | | --- | | begin |  |  | | --- | | if (S0 ='0' and S1 = '0') then |  |  | | --- | | A <= F; |  |  | | --- | | elsif (S0 ='1' and S1 = '0') then |  |  | | --- | | B <= F; |  |  | | --- | | elsif (S0 ='0' and S1 = '1') then |  |  | | --- | | C <= F; |  |  | | --- | | else |  |  | | --- | | D <= F; |  |  | | --- | | end if; |  |  | | --- | |  |  |  | | --- | | end process; | | | | |  | | | |  | |