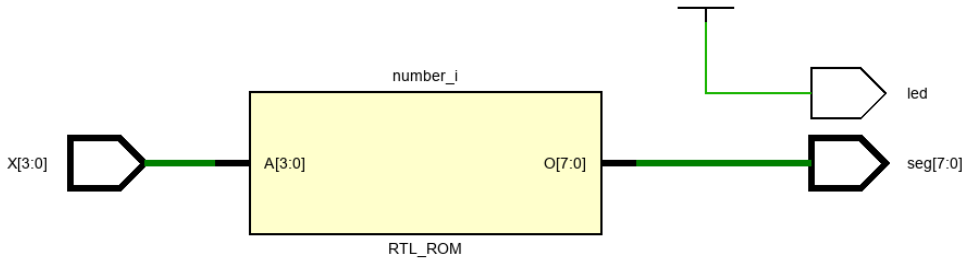
Lab 4

Student ID: F10915108 Name: Gerardo Sigfredo Fisch Paredes

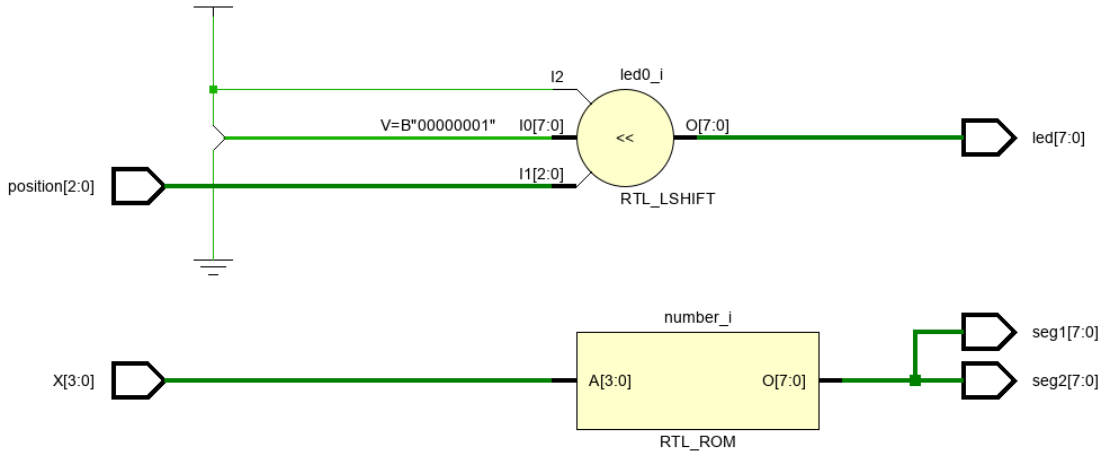
1. Topic: Seven-Segment Controller

2. Design Structure:

a) Question 1



b) Question 2



3. Code:

a) Question 1

i) Code:

```
module four_digit_seven_segment(X,seg,led);

    input [3:0] X;              // used to choose which number to display
    output reg [7:0]seg;        // stores the pattern of the current number to be displayed
    output reg led;             // used to turn on the last digit of the board

    wire [7:0] number [15:0];   // number[i] = byte that displays i

    assign number[0] = 8'b10111111;
    assign number[1] = 8'b10000110;
    assign number[2] = 8'b11011011;
    assign number[3] = 8'b11001111;
    assign number[4] = 8'b11100110;
    assign number[5] = 8'b11101101;
    assign number[6] = 8'b11111101;
    assign number[7] = 8'b10100111;
    assign number[8] = 8'b11111111;
    assign number[9] = 8'b11101111;
    assign number[10] = 8'b11110111;
    assign number[11] = 8'b11111100;
    assign number[12] = 8'b11011000;
    assign number[13] = 8'b11011110;
    assign number[14] = 8'b11111001;
    assign number[15] = 8'b11110001;

    always @(*)
    begin
        seg = number[X];
        led = 1;
    end
endmodule
```

ii) Test Bench:

```
module tb_4_digit_7_segment;
    // Input
    reg [3:0] X;
    // Outputs
    wire [7:0]seg;
    wire led;

    integer i;
    // Initiate the Unit Under Test (UUT)
    four_digit_seven_segment uut(X,seg,led);

    initial begin
        for (i=0;i<16;i=i+1) // X goes from 0 to 15
        begin
            X = i;
            // Wait 100ns for global reset to finish
            #100;
        end
        $finish;
    end
endmodule
```

b) Question 2

i) Code:

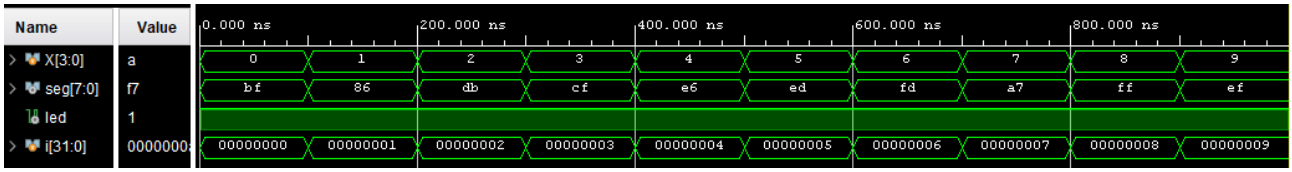
```
module four_digit_seven_segment(X, position, seg1, seg2, led);
    input [3:0] X;              // used to choose which number to display
    input [2:0] position;       // used to choose which 7 segment to turn on
    output reg [7:0] seg1, seg2; // stores the pattern of the current number to be displayed
    output reg [7:0] led;       // selects which 7 segment to turn on

    wire [7:0] number [15:0];   // number[i] = byte that displays i

    assign number[0]  = 8'b10111111;
    assign number[1]  = 8'b10000110;
    assign number[2]  = 8'b11011011;
    assign number[3]  = 8'b11001111;
    assign number[4]  = 8'b11100110;
    assign number[5]  = 8'b11101101;
    assign number[6]  = 8'b11111101;
    assign number[7]  = 8'b10100111;
    assign number[8]  = 8'b11111111;
    assign number[9]  = 8'b11101111;
    assign number[10] = 8'b11110111;
    assign number[11] = 8'b11111100;
    assign number[12] = 8'b11011000;
    assign number[13] = 8'b11011110;
    assign number[14] = 8'b11111001;
    assign number[15] = 8'b11110001;

    always @(*)
    begin
        seg1 = number[X];
        seg2 = number[X];

        led = 1 << position;
    end
endmodule
```

ii) Test Bench:

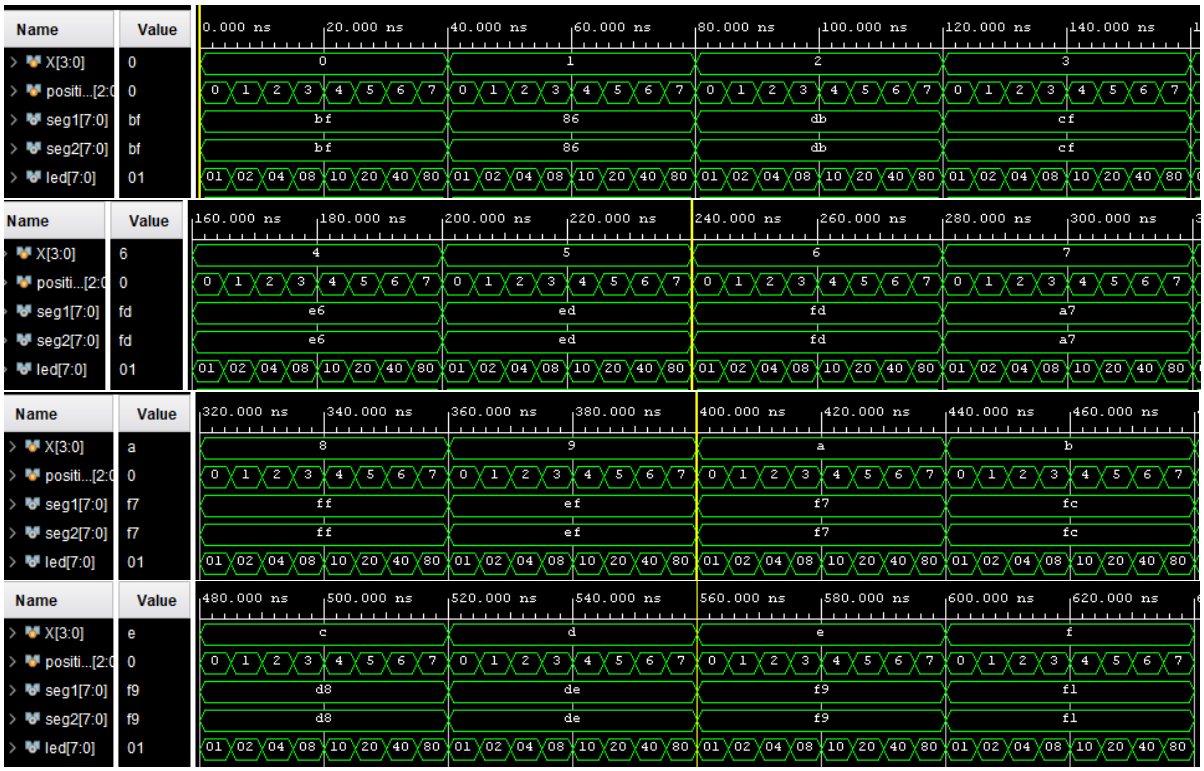
```
module tb_4_digit_7_segment();
    // Inputs
    reg [3:0] X;
    reg [2:0] position;
    // Outputs
    wire [7:0] seg1, seg2, led;

    integer i, j;
    // Initiate the Unit Under Test (UUT)
    four_digit_seven_segment uut(X, position, seg1, seg2, led);

    initial begin
        for (i=0;i<16;i=i+1) // X goes from 0 to 15
        begin
            X = i;
            for (j=0;j<8;j=j+1) // position goes from 0 to 7
            begin
                position = j;
                // Wait 5ns for global reset to finish
                #5;
            end
        end
        $finish;
    end
endmodule
```

4. Simulation Waveforms:

a) Question 1



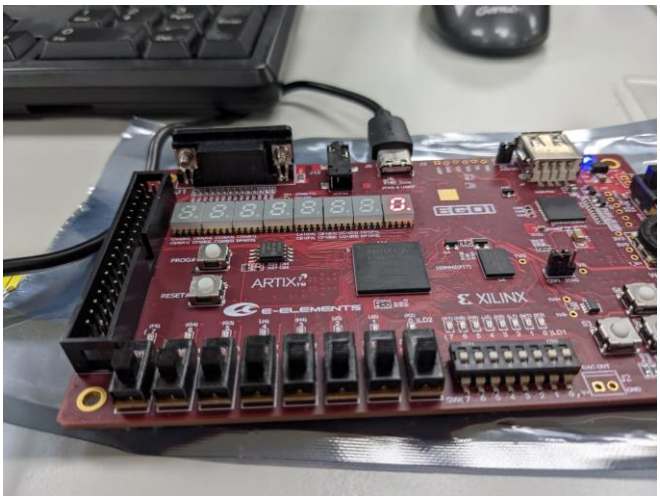
b) Question 2



5. FPGA Results:

a) Question 1

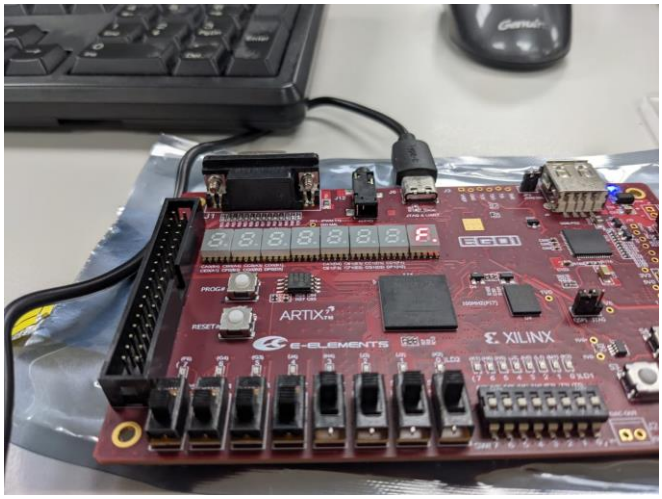
- SW3 – SW0 represents the digit



[SW3...SW0] = $0000_2 = 0_{16}$

[SW3...SW0] = $1010_2 = A_{16}$

[SW3...SW0] = $1110_2 = E_{16}$

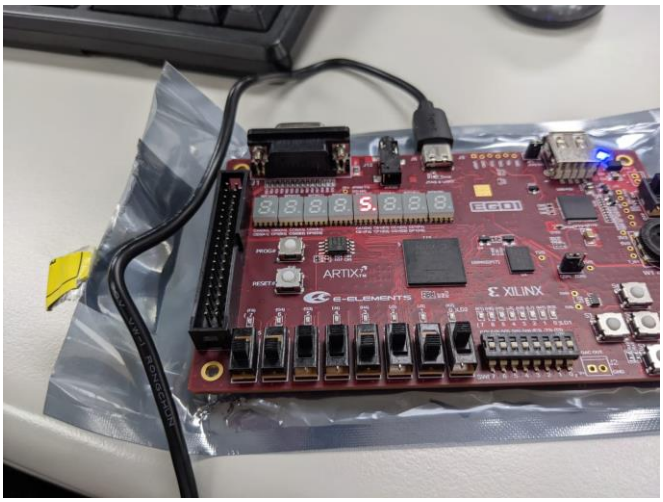


[SW3...SW0] = $1111_2 = F_{16}$

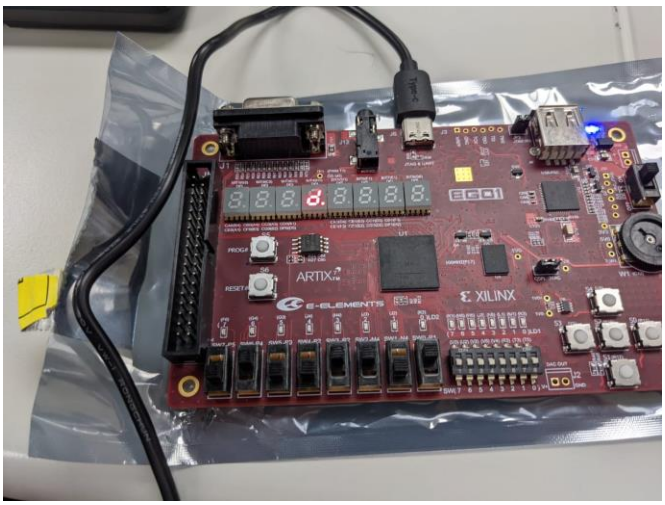
b) Question 2

- SW3 – SW0 represents the digit
- SW6 – SW4 represents the position

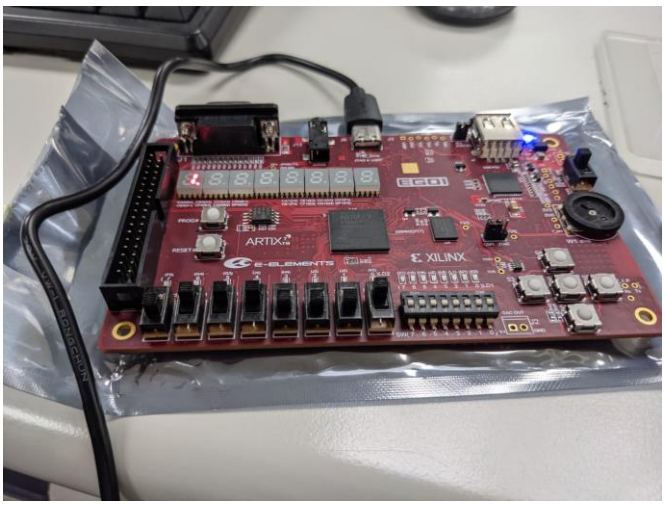
[SW3...SW0] = $1111_2 = F_{16}$
[SW6...SW4] = $001_2 = 1_{10}$



[SW3...SW0] = $0101_2 = 5_{16}$
[SW6...SW4] = $011_2 = 3_{10}$



[SW3...SW0] = $1101_2 = D_{16}$
[SW6...SW4] = $100_2 = 4_{10}$



[SW3...SW0] = $0001_2 = 1_{16}$
[SW6...SW4] = $111_2 = 7_{10}$

6. Reflection:

The code from this lab were quite simple. At first, it can appear that some variables are not needed, such as seg, seg1 and seg2. However, these variables were used to set up ports of the “xc7a35tcsg324-1” to make the program work as intended on the board.

We can see from our results that both experiments went as expected. We learned how to use "always" on the Vivado software. This lab also helped us to remember how to use seven-segment displays.