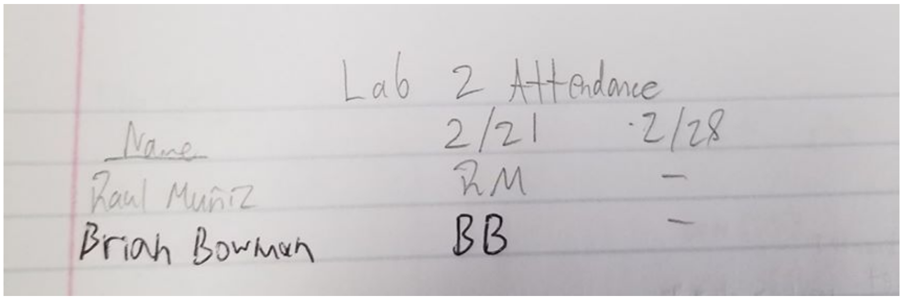
Lab2
Brian Bowman
Raul Muniz



Lab 2 Attendance

Name	2/21	·2/28
Raul Muñiz	RM	–
Brian Bowman	BB	–

Part 1

I. **Problem:** Display a 4-bit binary-coded-decimal (BCD) on a 7-segment display.

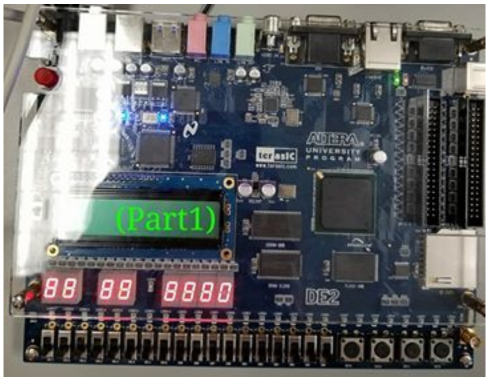
II. **Conceptual Design:** The rightmost four switches, SW[3:0] are used to input the BCD, and the rightmost 7-segment display, HEX0, will output the number. If a number greater than 9 is input, the output is blank.

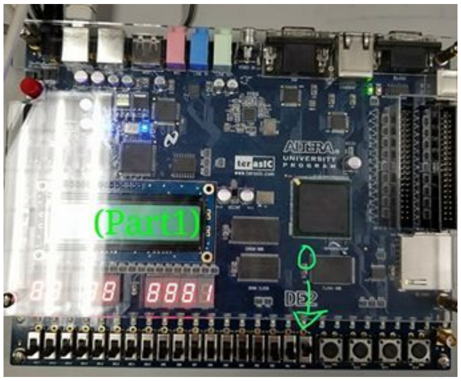
III. **Verilog Design:**

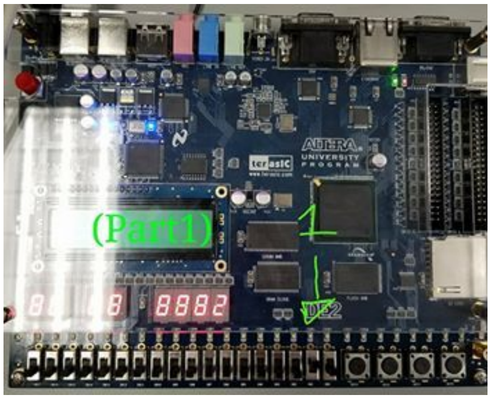
```
module bcd2leds (bcd, leds);
    input [3:0] bcd;
    output reg [6:0] leds;

    always @ (*) begin
        case (bcd)
            4'b0000 : leds = 7'b1000000; // 0
            4'b0001 : leds = 7'b1111001; // 1
            4'b0010 : leds = 7'b0100100; // 2
            4'b0011 : leds = 7'b0110000; // 3
            4'b0100 : leds = 7'b0011001; // 4
            4'b0101 : leds = 7'b0010010; // 5
            4'b0110 : leds = 7'b0000010; // 6
            4'b0111 : leds = 7'b1111000; // 7
            4'b1000 : leds = 7'b0000000; // 8
            4'b1001 : leds = 7'b0010000; // 9
            default : leds = 7'b1111111; // blank
        endcase
    end
endmodule

module lab2part1 (SW, HEX0);
    input [3:0] SW;
    output [6:0] HEX0;

    bcd2leds U0 (SW, HEX0);
endmodule
```

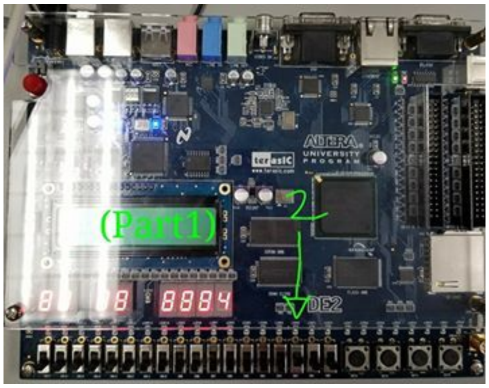
IV. Lab Procedure:

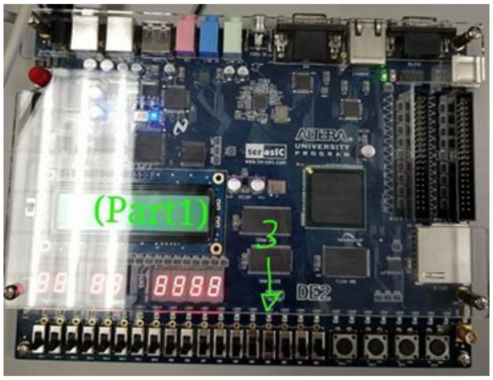
Pictures

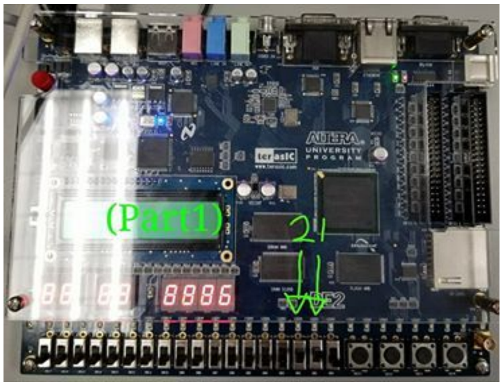












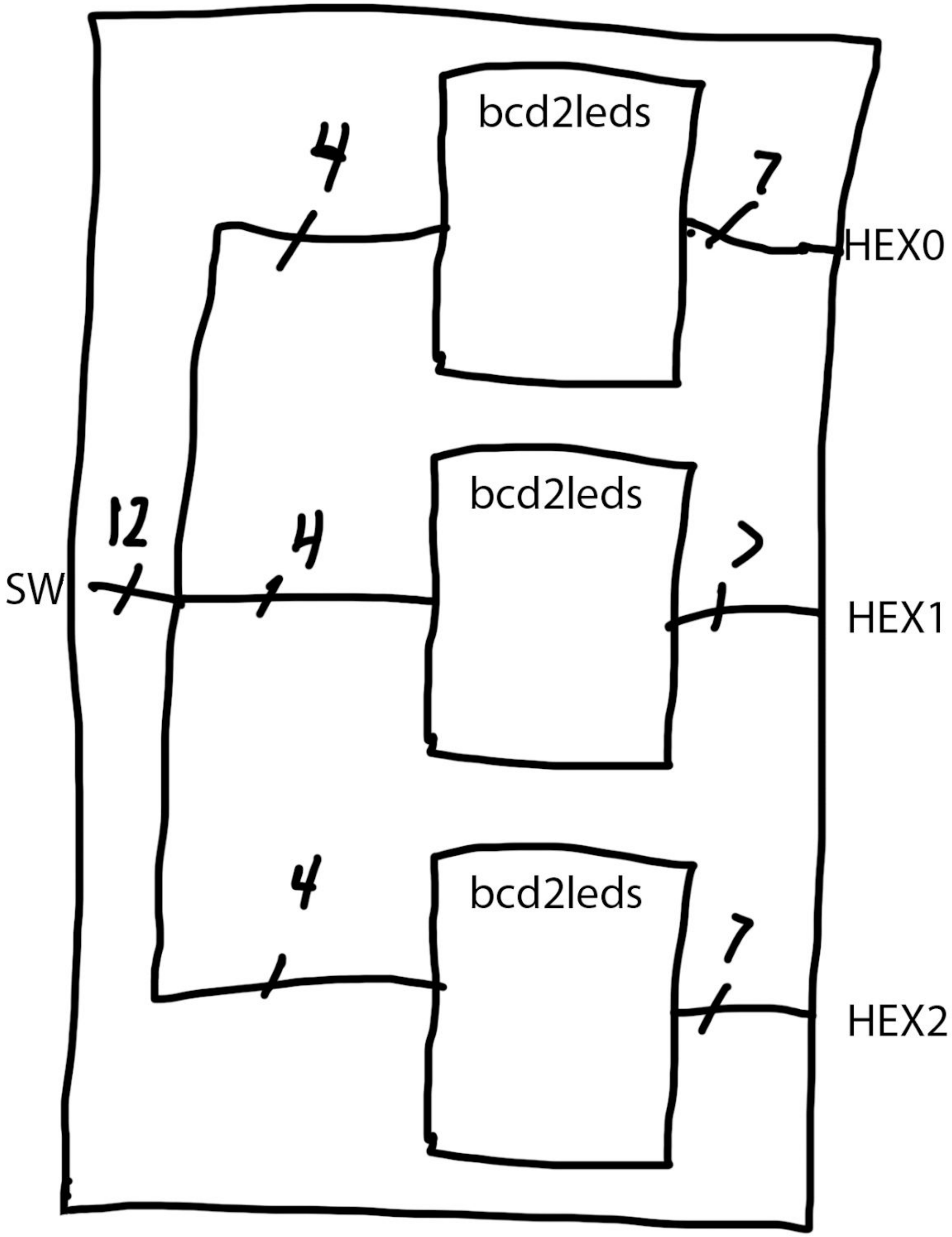
Troubleshooting: n/a

Conclusion: The lab was straightforward. The bcd2leds module is just like the module from lab 1 that displays the characters ‘h’, ‘e’, ‘l’, ‘o’, except this module displays numbers.

Part 2

I. **Problem:** Use three instances of the module from part 1 to display a 3-digit BCD number on the three 7-segment displays HEX2-HEX0, with each digit being controlled by four switches.

II. Conceptual Design:

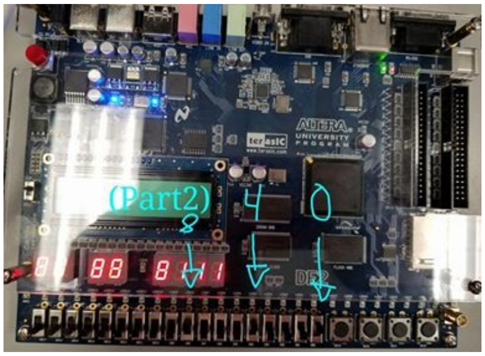


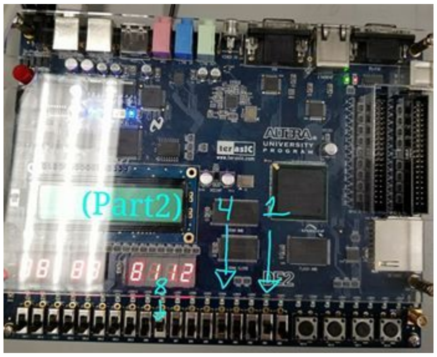
III. Verilog Design:

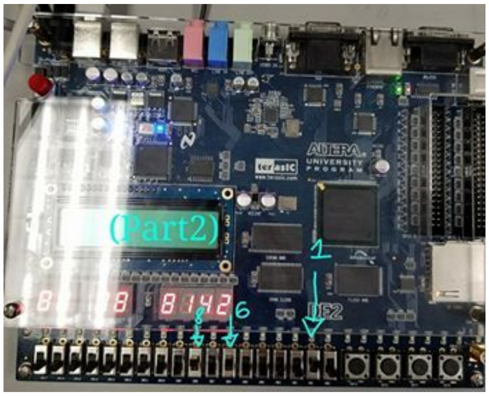
```
module lab2part2 (SW, HEX0, HEX1, HEX2);
     input [11:0] SW;
     output [6:0] HEX0, HEX1, HEX2;

     bcd2leds U0 (SW[3:0], HEX0);
     bcd2leds U1 (SW[7:4], HEX1);
     bcd2leds U2 (SW[11:8], HEX2);
endmodule
```

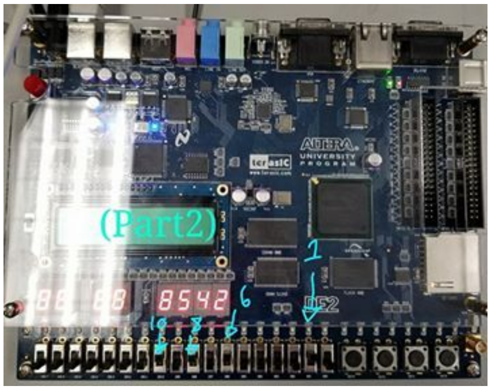
IV. Lab Procedure:

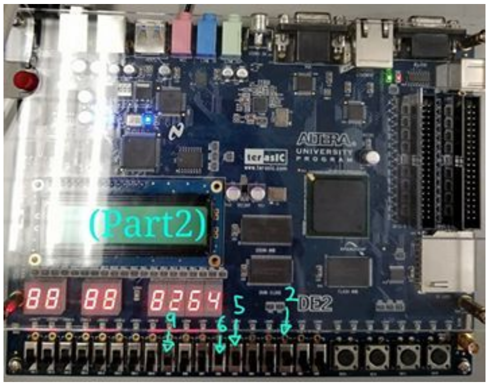
Pictures











Troubleshooting: n/a

Conclusion: This part was straightforward, just consisting of three instances of part 1.

Part 3

I. **Problem:** Compute the sum of two single-digit numbers, and display the input and output digits over four 7-segment displays.

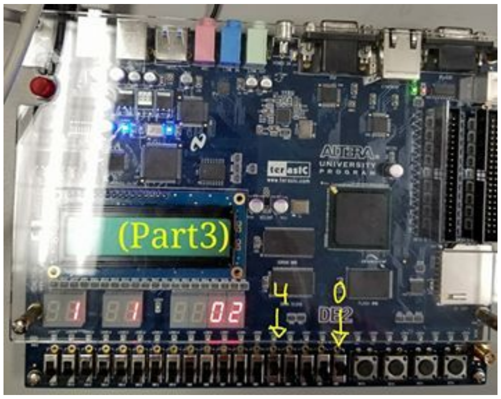
II. **Conceptual Design:** The eight rightmost switches are used to input the two digits to be summed. These inputs are fed to the bcdadd1 module which computes the 2-digit sum, and are also displayed on HEX4 and HEX6 using the bcd2leds module. The sum is displayed on the two rightmost 7-segment displays using the bcd2leds module.

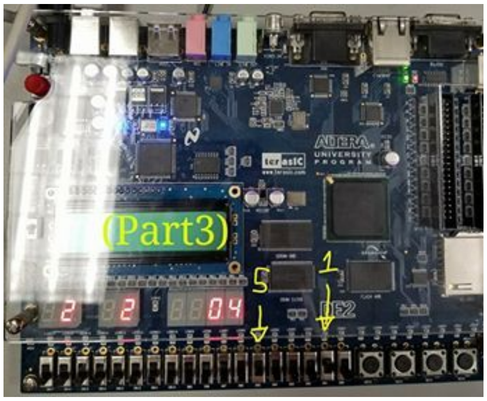
III. **Verilog Design**

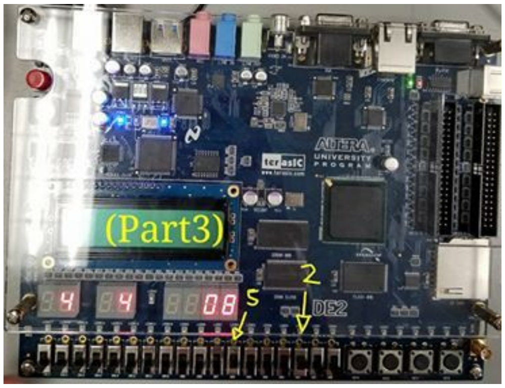
```
module bcdadd1 (a0, b0, s0, s1);
     input [3:0] a0, b0;
     output [3:0] s0;
     output s1;

     assign s0 = (a0 + b0) % 10;
     assign s1 = (a0 + b0) / 10;
endmodule

module lab2part3 (SW, HEX0, HEX1, HEX2, HEX3, HEX4, HEX5, HEX6, HEX7);
     input [7:0] SW;
     output [6:0] HEX0, HEX1, HEX2, HEX3, HEX4, HEX5, HEX6, HEX7;
     wire s1; // tens sum digit
     wire [3:0] s0; // ones sum digit

     // blank unused 7-segment displays
     assign HEX2 = 7'b1111111;
     assign HEX3 = 7'b1111111;
     assign HEX5 = 7'b1111111;
     assign HEX7 = 7'b1111111;

     // display input
     bcd2leds U0 (SW[3:0], HEX4);
     bcd2leds U1 (SW[7:4], HEX6);

     // display sum
     bcd2leds U2 (s0, HEX0);
     bcd2leds U3 (s1, HEX1);

     // calculate sum
     bcdadd1 U4 (SW[3:0], SW[7:4], s0, s1);
endmodule
```

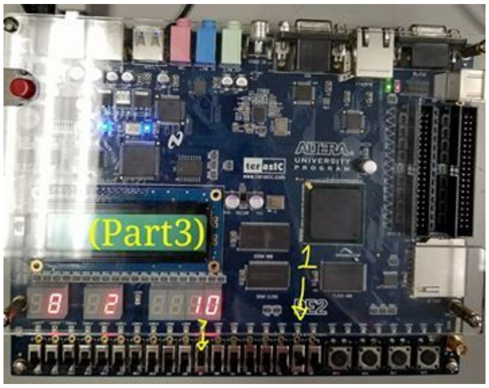
IV. Lab Procedure:

Pictures









Troubleshooting: n/a

Conclusion: We managed to successfully execute the correct output by assigning accordingly the correct number of inputs and outputs which were 3 as in the parts before. Then s0 will be set to the number of the input of LEDS mod 10. S1 will be a direct integer division of the input of LEDS by 10. Next the module "lab2part3" assigned each digit accordingly with blanks and corresponding switches with LEDS.

Part 4

I. **Problem:** Compute the sum of two two-digit numbers, and display the input and output digits over seven 7-segment displays.

II. **Conceptual Design:** The two numbers to be added are input using SW[15:0]. The inputs are visualized on HEX4-7. The result of the addition is displayed on HEX0-2. HEX3, which isn't used, is turned off to make reading the inputs and outputs easier.

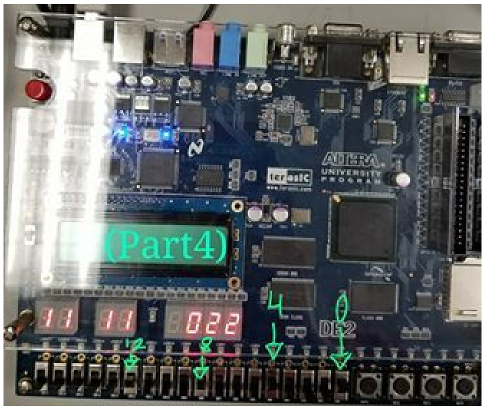
III. **Verilog Design:**

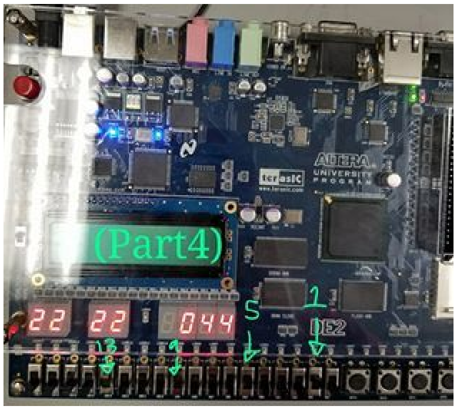
```
module bcdadd1fa (a0, b0, c_in, s0, c_out);
     input [3:0] a0, b0;
     input c_in;
     output [3:0] s0;
     output c_out;

     assign s0 = (a0 + b0 + c_in) % 10;
     assign c_out = (a0 + b0 + c_in) / 10;
endmodule

module lab2part4 (SW, HEX0, HEX1, HEX2, HEX3, HEX4, HEX5, HEX6, HEX7);
     input [15:0] SW;
     output [6:0] HEX0, HEX1, HEX2, HEX3, HEX4, HEX5, HEX6, HEX7;
     wire c0, c1; // carries
     wire [3:0] s0, s1; // ones and tens digits

     // blank unused 7-segment display
     assign HEX3 = 7'b1111111;

     // display input
     bcd2leds U0 (SW[3:0], HEX4);
     bcd2leds U1 (SW[7:4], HEX5);
     bcd2leds U2 (SW[11:8], HEX6);
     bcd2leds U3 (SW[15:12], HEX7);

     // display sum
     bcd2leds U4 (s0, HEX0);
     bcd2leds U5 (s1, HEX1);
     bcd2leds U6 (c1, HEX2);

     // calculate sum
     bcdadd1fa U7 (SW[3:0], SW[11:8], 0, s0, c0);
     bcdadd1fa U8 (SW[7:4], SW[15:12], c0, s1, c1);
```

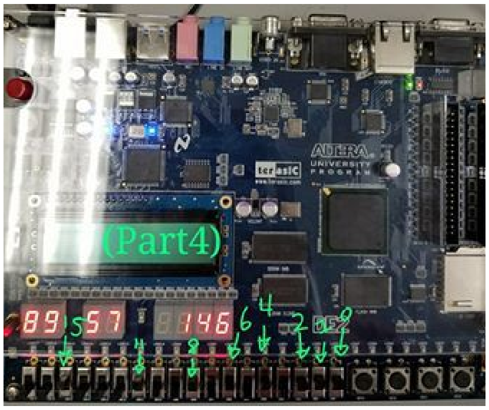
```
endmodule
```

IV. Lab Procedure:

Pictures:







Troubleshooting: n/a

Conclusion: In conclusion, we managed to successfully execute the correct output by assigning accordingly the correct number of inputs and outputs which were 3 as in the parts before. Then s0 will be set to the number of the input of LEDS mod 10. S1 will be a direct integer division of the input of LEDS by 10. Next the module “lab2part4” assigned each digit accordingly with blanks and corresponding switches with LEDS.