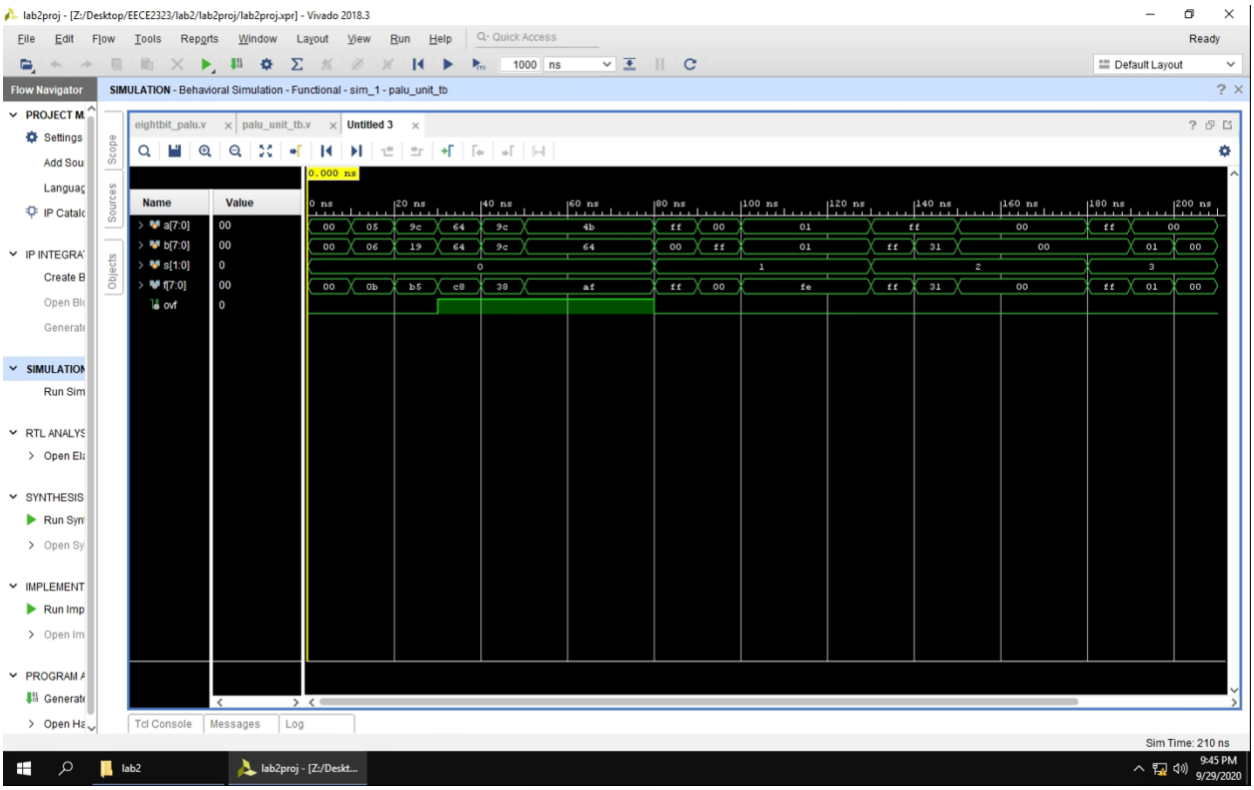
Alex Oswald
EECE 2323
Due: 29 Sep 2020

Pre-Lab Assignment 2

Simulation Screenshot:



Verilog Code for Parial ALU:

```
1.  module eightbit_palu(
2.
3.      input signed[7:0] a,
4.      input signed[7:0] b,
5.      input [1:0] s,
6.      output reg [7:0] f,
7.      output reg ovf);
8.
9.      wire [7:0] op1; // a + b
10.     wire [7:0] op2; // bitwise inversion of b
11.     wire [7:0] op3; // bitwise AND of a and b
12.     wire [7:0] op4; // bitwise OR of a and b
13.
14.     assign op1 = a + b;
15.     assign op2 = ~b;
16.     assign op3 = a & b;
17.     assign op4 = a | b;
18.
19.     always @(s or op1 or op2 or op3 or op4)
```

```
        case(s)
            2'b00: begin
                f = op1;
                ovf = (a[7] && b[7] && ~f[7]) || (~a[7] && ~b[7] && f[7]);
            end
            2'b01: begin
                f = op2;
                ovf = 0;
            end
            2'b10: begin
                f = op3;
                ovf = 0;
            end
            2'b11: begin
                f = op4;
                ovf = 0;
            end
            default:  begin
                f = 0;
                ovf = 0;
            end
        endcase

endmodule
```

Verilog Code for Testbench:

```
module palu_unit_tb ();
    // Inputs
    reg signed [7:0] a;
    reg signed [7:0] b;
    reg [1:0] s;
    // Outputs
    wire signed [7:0] f;
    wire ovf;

    // Initiate the Unit Under Test (UUT)
    eightbit_palu uut (
        .a(a),
        .b(b),
        .s(s),
        .f(f),
        .ovf(ovf));

    // Initialize Inputs (stimulus)
    initial
    begin
        #0
        a = 8'd0;  b = 8'd0;  s = 0;  #10;
        a = 8'd5;  b = 8'd6;  s = 0;  #10;
        a = -8'd100;  b = 8'd25;  s = 0;  #10;
        a = 8'd100;  b = 8'd100;  s = 0;  #10;
        a = -8'd100;  b = -8'd100;  s = 0;  #10;
        a = 8'd75;  b = 8'd100;  s = 0;  #10;

        #20
        a = -8'd1;  b = 8'd0;  s = 1;  #10;
        a = 8'd0;  b = -8'd1;  s = 1;  #10;
        a = 8'd1;  b = 8'd1;  s = 1;  #10;

```

```
        #20
        a = -8'd1;  b = -8'd1;  s = 2;  #10;
        a = -8'd1;  b = 8'd49;  s = 2;  #10;
        a = 8'd0;  b = 8'd0;  s = 2;  #10;

        #20
        a = -8'd1;  b = 8'd0;  s = 3;  #10;
        a = 8'd0;  b = 8'd1;  s = 3;  #10;
        a = 8'd0;  b = 8'd0;  s = 3;  #10;
    end

endmodule
```