CS223-003 (DIGITAL DESIGN)

LAB 2

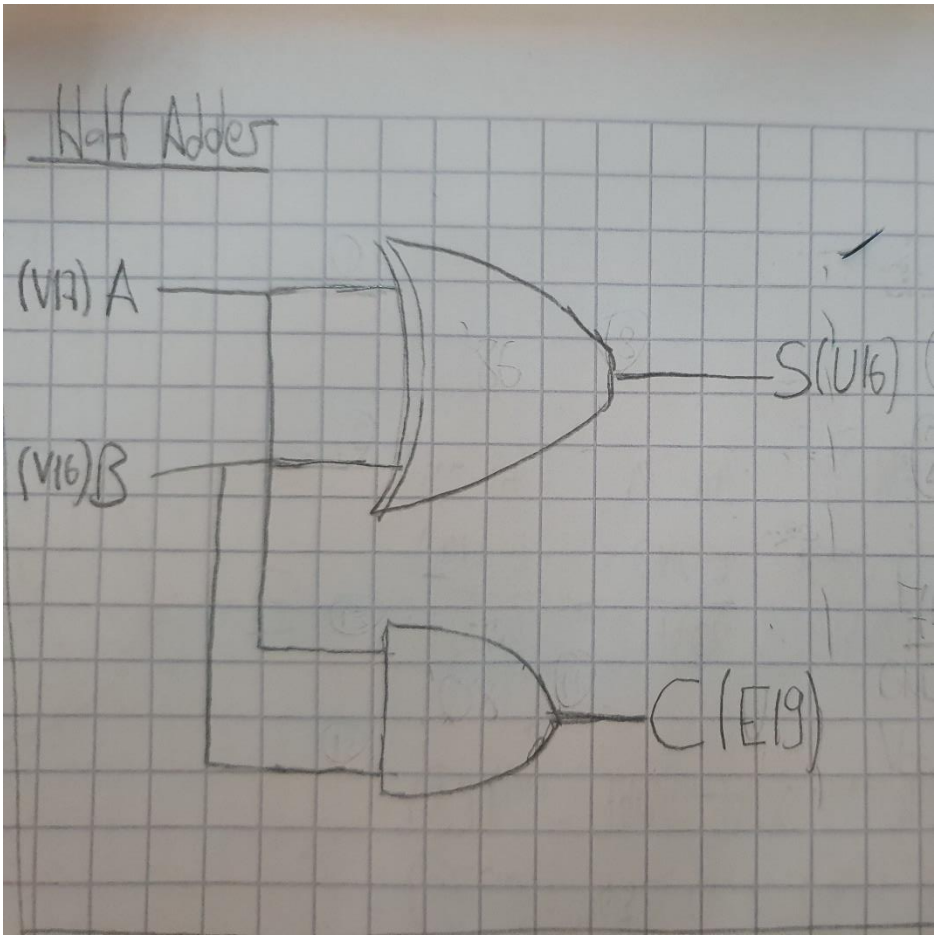
NAME: SERTAÇ DERYA

ID: 22003208

DATE: 17.10.2022

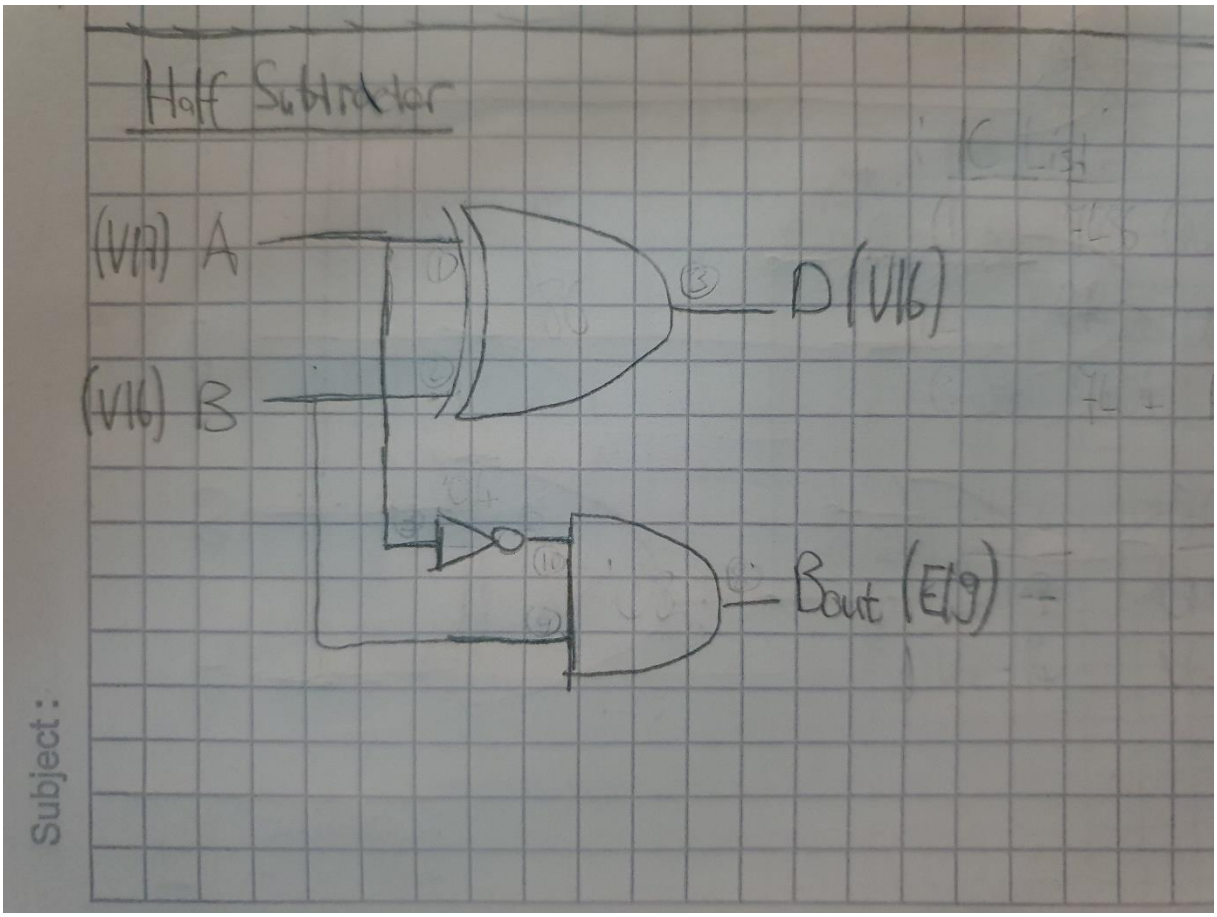
Half Adder Schematic & Module & TestBench

```
module half_adder_source(
    input a,
    input b,
    output cout,
    output s
    );

    assign s = a ^ b;
    assign cout = a & b;

endmodule

module half_adder_sim( );
    logic a, b, s, cout;
    half_adder_source dut(.a(a),.b(b),.s(s),.cout(cout));
    initial begin
        a = 0; b = 0; #10;
        b = 1; #10;
        b = 0; a = 1; #10;
        b = 1;
    end
endmodule
```



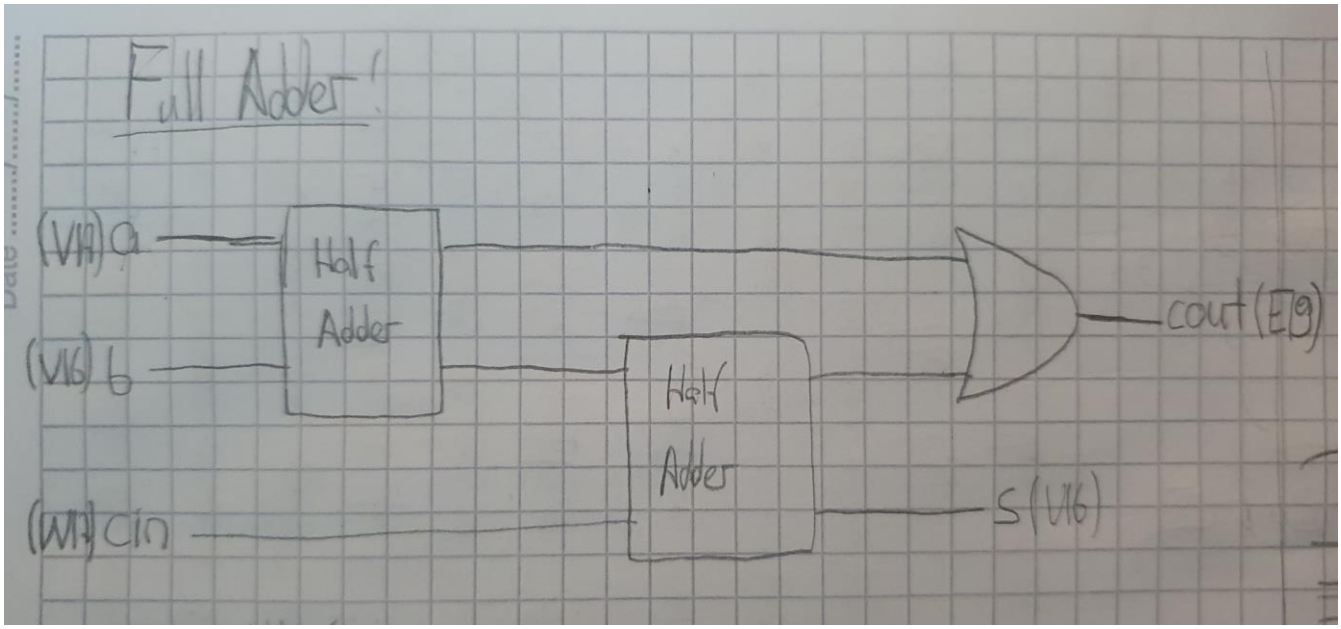
Half Subtractor Schematic & Module & TestBench

```
module half_subtractor_source(
    input a,
    input b,
    output d,
    output bout
    );

    assign d = a ^ b;
    assign bout = ~a & b;

endmodule

module half_substractor_sim( );
    logic a, b, bout, d;
    half_subtractor_source dut(.a(a), .b(b), .bout(bout), .d(d));
    initial begin
        #10;
        b = 1; #10;
        b = 0; a = 1; #10;
        b = 1;
    end
endmodule
```



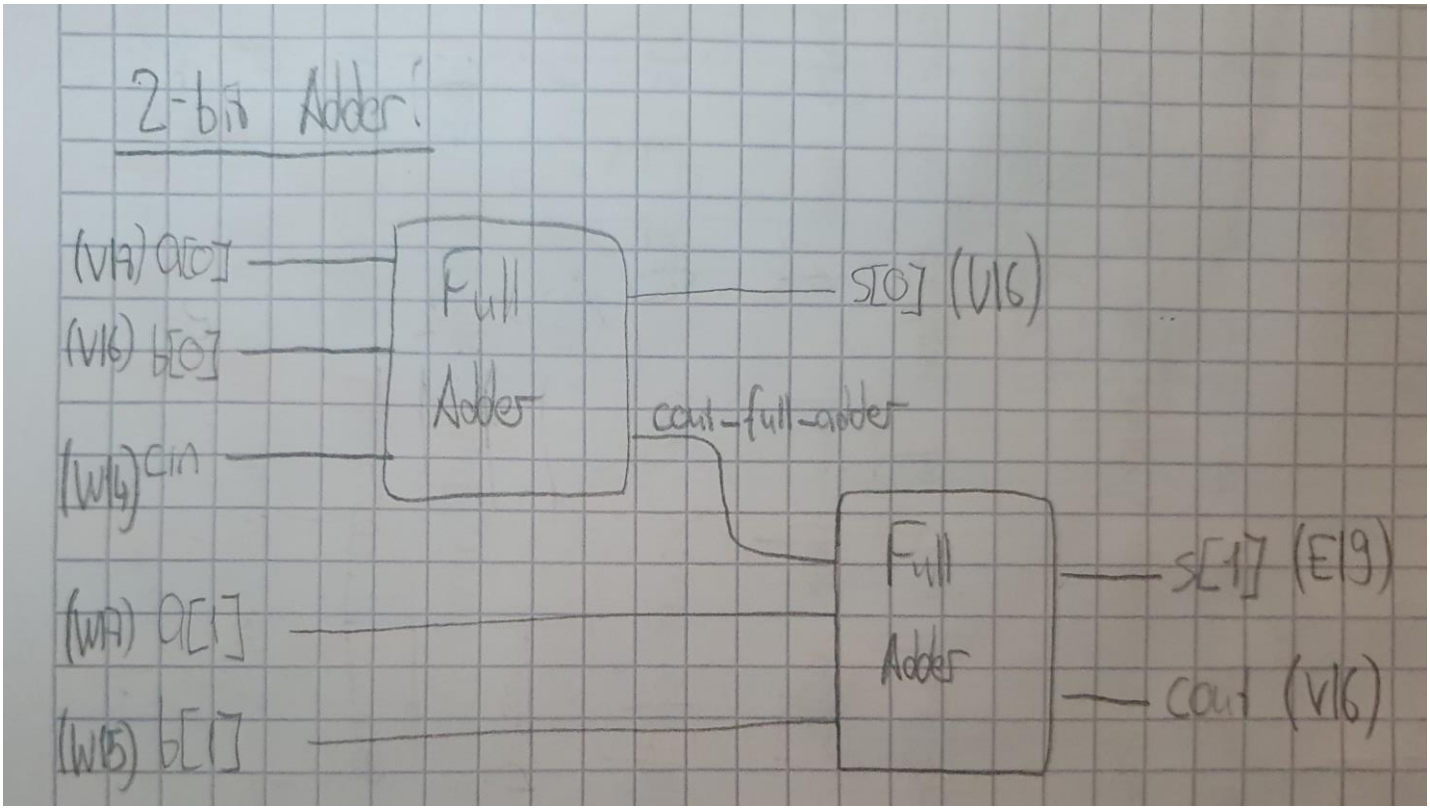
Full Adder Schematic & Module & TestBench

```
module full_adder_source(
    input a,
    input b,
    input cin,
    output s,
    output cout
    );
    logic result_adder1_s;
    logic result_adder1_cout;
    logic result_adder2_cout;

    half_adder_source adder1(.a(a), .b(b),
.cout(result_adder1_cout), .s(result_adder1_s));
    half_adder_source adder2(.a(result_adder1_s), .b(cin),
.cout(result_adder2_cout), .s(s));

    assign cout = result_adder1_cout | result_adder2_cout;
endmodule

module lab_calc_sim( );
    logic c, d, a, b, y, z;
    lab_calc_source(.c(c),.d(d),.a(a),.b(b),.y(y),.z(z));
    initial begin
        #10;
        b = 1; #10;
        b = 0; a = 1; #10;
        b = 1; #10;
        b = 0; a = 0; d = 1; #10;
        b = 1; #10;
        b = 0; a = 1; #10;
        b = 1; #10;
        d = 0; a = 0; b = 0; c = 1; #10;
        b = 1; #10;
        b = 0; a = 1; #10;
        b = 1; #10;
        a = 0; b = 0; d = 1; #10;
        b = 1; #10;
        b = 0; a = 1; #10;
        b = 1; #10;
    end
endmodule
```


Full Adder
(V17) a
(V16) b
(W17) cin
Half Adder
Half Adder
cout (E19)
S (U16)

2-Bit Adder Schematic & Module & TestBench

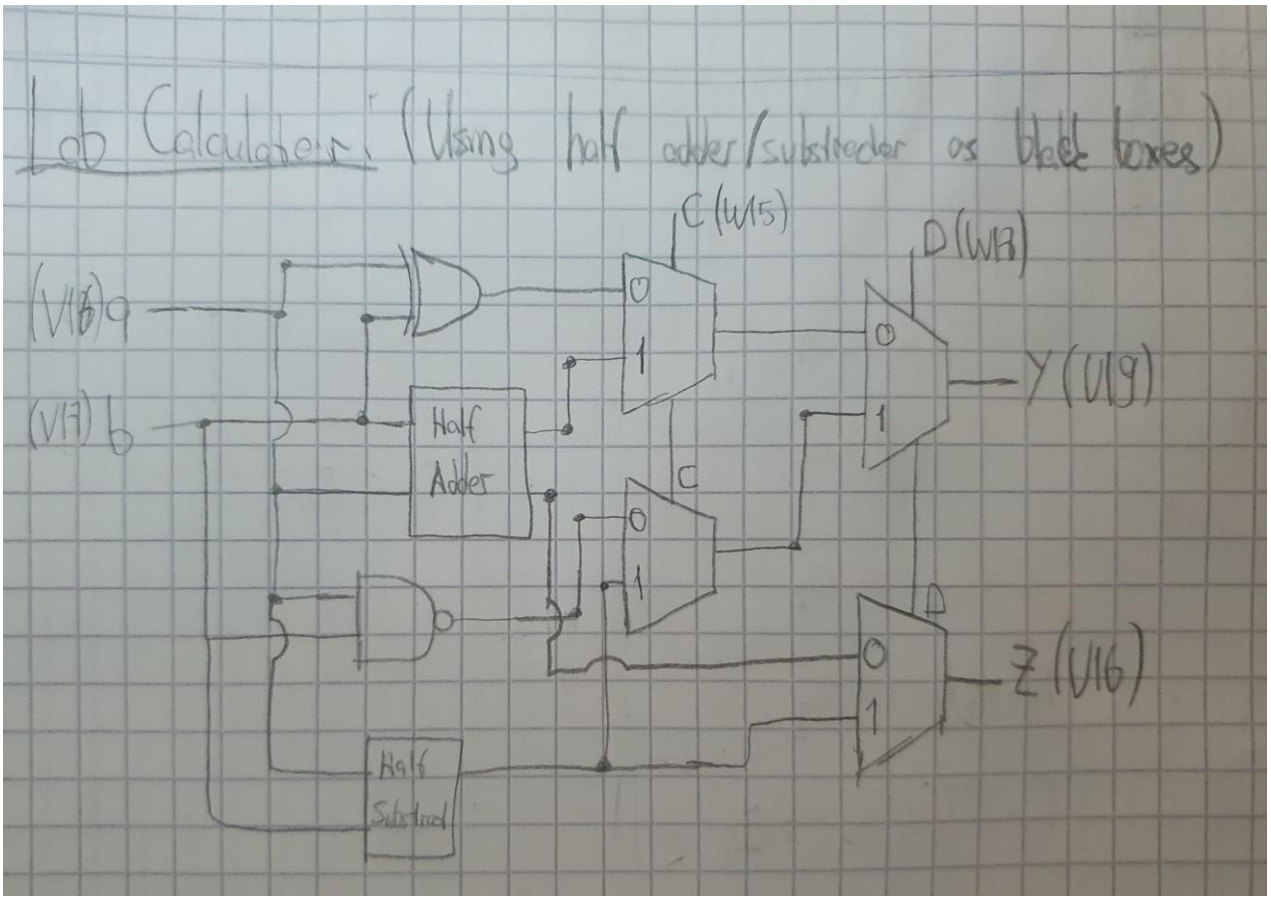
```
module two_bit_adder_source(
    input a[1:0],
    input b[1:0],
    input cin,
    output s[1:0],
    output cout
    );
    logic cout_full_adder1;

    full_adder_source full_adder1(.a(a[0]), .b(b[0]), .cin(cin),
.s(s[0]), .cout(cout_full_adder1));
    full_adder_source full_adder2(.a(a[1]), .b(b[1]),
.cin(cout_full_adder1), .s(s[1]), .cout(cout));
endmodule
```



```
module two_bit_adder_sim( );
    logic clk, reset;
    logic a[1:0], b[1:0], cin, s_expected[1:0], cout_expected;
    logic s[1:0], cout;
    logic [31:0] vectornum, errors;
    logic [7:0] testvectors[10000:0];

    two_bit_adder_source dut(.a(a), .b(b), .cin(cin), .cout(cout),
.s(s));

    always
        begin
            $readmemb("testvectors.txt", testvectors);
            vectornum = 0; errors = 0;
            reset = 1;; #27; reset = 0;
        end

    always @(posedge clk)
        begin
            #1; {a[1],a[0], b[1], b[0], cin, s_expected[1],
s_expected[0], cout_expected} = testvectors[vectornum];
        end

    always @(negedge clk)
        if(~reset) begin
            if(s[1:0] !== s_expected[1:0] | cout !== cout_expected)
begin
                $display("Error");
                errors = errors + 1;
            end
            vectornum = vectornum + 1;
            if(testvectors[vectornum] === 4'bx) begin
                $display("%d test completed with %d errors",
vectornum, errors);
                $finish;
            end
        end
   endmodule
```

Lab Calculator Schematic & Module & TestBench

```
module lab_calc_source(
    input c,
    input d,
    input a,
    input b,
    output y,
    output z
    );
    logic difference, bout;
    logic sum, cout;

    half_subtractor_source
subtractor(.a(a),.b(b),.bout(bout),.d(difference));
    half_adder_source adder(.a(a),.b(b),.cout(cout),.s(sum));

    assign y = c ? (d ? difference : sum) : (d ? ~(a & b) : a ^ b);
    assign z = d ? bout : cout;
endmodule

module lab_calc_sim( );
    logic c, d, a, b, y, z;
    lab_calc_source(.c(c),.d(d),.a(a),.b(b),.y(y),.z(z));
    initial begin
        #10;
        b = 1; #10;
        b = 0; a = 1; #10;
        b = 1; #10;
        b = 0; a = 0; d = 1; #10;
        b = 1; #10;
        b = 0; a = 1; #10;
        b = 1; #10;
        d = 0; a = 0; b = 0; c = 1; #10;
        b = 1; #10;
        b = 0; a = 1; #10;
        b = 1; #10;
        a = 0; b = 0; d = 1; #10;
        b = 1; #10;
        b = 0; a = 1; #10;
        b = 1; #10;
    end
endmodule
```


Lab Calculator: (Using half adder/subtracter as black boxes)
C (U15)
D (U18)
(V16) a
(V17) b
Half Adder
Y (U19)
Z (V16)
Half Subtract