CS 223

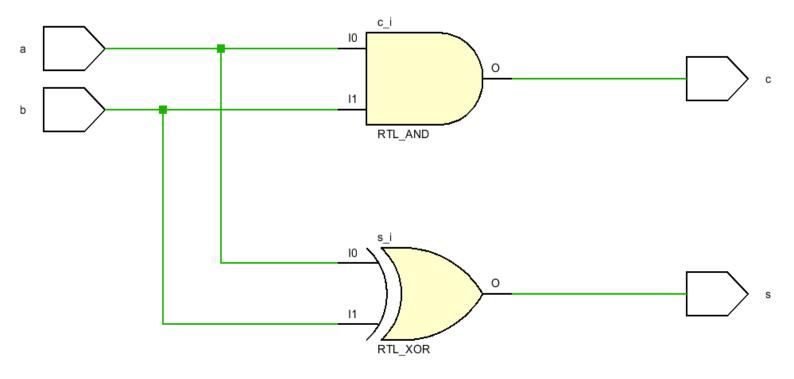
Section – 6

Lab - 02

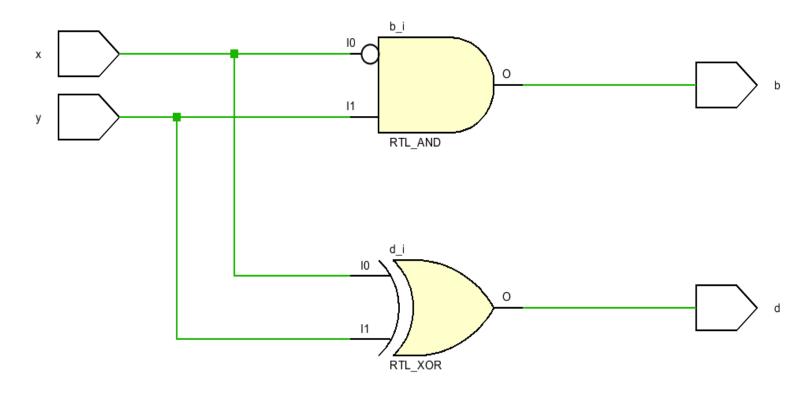
Mert Fidan 22101734

15.10.2022

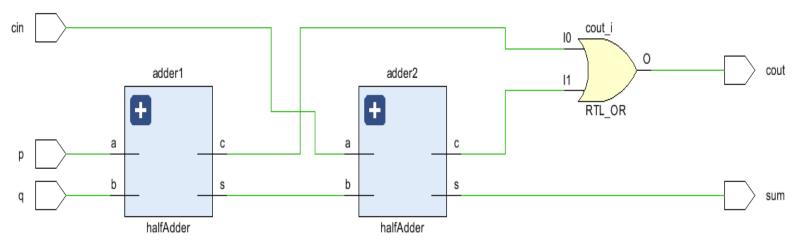
• Half Adder Circuit Schematic:



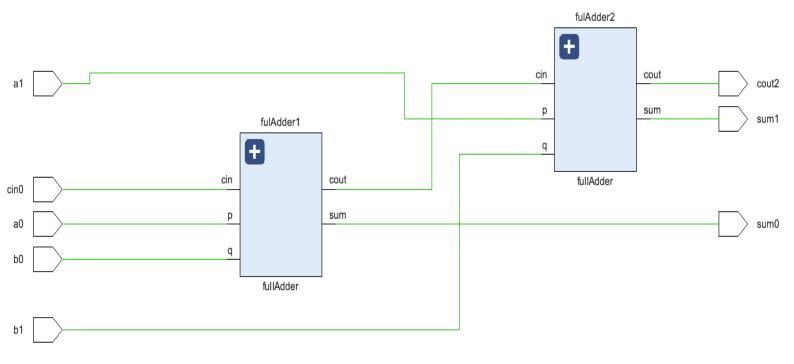
• Half Subtractor Circuit Schematic:



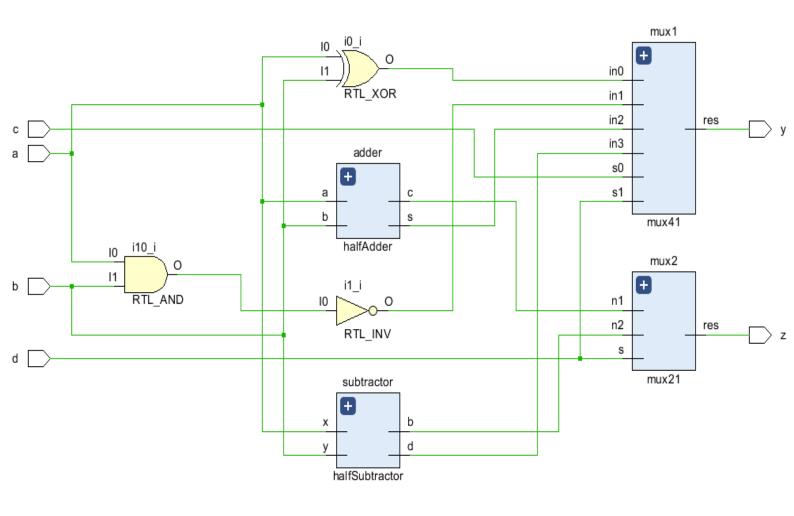
• Full Adder Circuit Schematic:



• 2-bit Adder Circuit Schematic:



• Lab Calculator Circuit Schematic:



• Behavioral 2:1 Multiplexer Module:

```
14 : // Dependencies:
15
    -//
    // Revision:
16
  // Revision 0.01 - File Created
17
    // Additional Comments:
18
    11
19
20
    21
22
23
    module mux21(input logic nl, n2, s, output logic res);
        assign res = (\sim s \& n1) + (s \& n2);
24
25
  endmodule
26
         • Behavioral 4:1 Multiplexer Module:
 'timescale lns / lps
 // Engineer:
 // Create Date: 16.10.2022 16:47:44
 // Design Name:
 // Module Name: mux41
```

```
3 // Company:
6
   // Project Name:
   // Target Devices:
   // Tool Versions:
11
   // Description:
12
13
   11
   // Dependencies:
14
15
   // Revision:
16
17
   // Revision 0.01 - File Created
18
   // Additional Comments:
19
21
23 — module mux41(input logic s0, s1, in0, in1, in2, in3, output logic res);
24
      assign res = (~s0 & ~s1 & in0) + (~s0 & s1 & in1) + (s0 & ~s1 & in2) + (s0 & s1 & in3);
25 endmodule
26
```

• Behavioral Half Adder Module:

```
1 'timescale lns / lps
3 // Company:
  // Engineer:
5 //
6 // Create Date: 15.10.2022 16:43:57
  // Design Name:
8 // Module Name: halfAdder
  // Project Name:
10 // Target Devices:
11
   // Tool Versions:
   // Description:
12
13
  // Dependencies:
14
15 //
16 | // Revision:
17 // Revision 0.01 - File Created
18 : // Additional Comments:
19 : //
21
22
23 - module halfAdder(input logic a, b, output logic s, c);
24 assign s = a ^ b;
25
      assign c = a & b;
26 endmodule
27
```

• Half Adder Testbench:

```
6 | // Create Date: 15.10.2022 17:02:10
   // Design Name:
7
8 | // Module Name: testbench1
9 : // Project Name:
   // Target Devices:
10
11 // Tool Versions:
12 // Description:
13
  // Dependencies:
14
15 ; //
16 // Revision:
   // Revision 0.01 - File Created
18 // Additional Comments:
21
2.2
23 - module testbenchl();
24
      logic a, b, s, c;
      halfAdder(a, b, s, c);
25
26 🖯
     initial begin
27
         a = 0; b = 0; #10;
28
          a = 0; b = 1; #10;
29
         a = 1; b = 0; #10;
          a = 1; b = 1; #10;
30
31
       end
32 endmodule
33
```

• Behavioral Half Subtractor Module:

```
1 'timescale lns / lps
3 // Company:
4 // Engineer:
  // Create Date: 15.10.2022 16:43:57
   // Design Name:
   // Module Name: halfSubtractor
   // Project Name:
10
   // Target Devices:
11 | // Tool Versions:
12 // Description:
13 //
  // Dependencies:
14
  11
15
16 / // Revision:
17 // Revision 0.01 - File Created
18 | // Additional Comments:
19
21
22
23 - module halfSubtractor(input logic x, y, output logic d, b);
      assign d = x ^ y;
25
      assign b = ~x & y;
26 endmodule
27
            • Half Subtractor Testbench:
    // Create Date: 15.10.2022 17:02:10
 7 // Design Name:
 8 / // Module Name: testbench2
    // Project Name:
 9
10 | // Target Devices:
11 // Tool Versions:
12 // Description:
13
14 // Dependencies:
15 ; //
   // Revision:
    // Revision 0.01 - File Created
17
18 // Additional Comments:
19 //
21
22
23 module testbench2();
        logic x, y, d, b;
        halfSubtractor(x, y, d, b);
26 🖯
        initial begin
27
           x = 0; y = 0; #10;
28
          x = 0; y = 1; #10;
29
          x = 1; y = 0; #10;
           x = 1; y = 1; #10;
30
31
       end
```

32 endmodule

33

• Structural Full Adder Module:

```
'timescale lns / lps
// Company:
    // Engineer:
5
   ! // Create Date: 15.10.2022 16:43:57
    // Design Name:
8
   // Module Name: fullAdder
   // Project Name:
10
    // Target Devices:
   // Tool Versions:
11
12 // Description:
13
   // Dependencies:
14
15
   1 //
   // Revision:
16
   // Revision 0.01 - File Created
17
18 | // Additional Comments:
19
22
23 — module fullAdder(input logic p, q, cin, output logic sum, cout);
       logic suml, coutl, cout2;
25
       halfAdder adderl(p, q, suml, coutl);
26
       halfAdder adder2(cin, suml, sum, cout2);
       assign cout = cout1 | cout2;
28 endmodule
29
             • Full Adder Testbench:
9 // Project Name:
10 | // Target Devices:
11 // Tool Versions:
    // Description:
12
13
14
    // Dependencies:
   11
15
16 // Revision:
17 | // Revision 0.01 - File Created
18 // Additional Comments:
19
22
23 module testbench3();
     logic p, q, cin, sum, cout;
25
      fullAdder(p , q, cin, sum, cout);
26 🖯
      initial begin
27
         p = 0; q = 0; cin = 0; #10;
         p = 0; q = 0; cin = 1; #10;
28
29
         p = 0; q = 1; cin = 0; #10;
30
         p = 0; q = 1; cin = 1; #10;
31
         p = 1; q = 0; cin = 0; #10;
32
         p = 1; q = 0; cin = 1; #10;
         p = 1; q = 1; cin = 0; #10;
33
34
         p = 1; q = 1; cin = 1; #10;
35 🖹
36 endmodule
37
```

• Structural 2-bit Adder Module:

```
'timescale lns / lps
// Company:
   // Engineer:
   // Create Date: 15.10.2022 16:43:57
    // Design Name:
    // Module Name: twoBitAdder
    // Project Name:
10
    // Target Devices:
    // Tool Versions:
11
   // Description:
12
13
  // Dependencies:
15
   11
    // Revision:
16
17
    // Revision 0.01 - File Created
18
    // Additional Comments:
19
21
23 — module twoBitAdder(input logic a0, b0, a1, b1, cin0, output logic sum0, sum1, cout2);
      logic coutl:
25
       fullAdder fulAdder1(a0, b0, cin0, sum0, cout1);
26
       fullAdder fulAdder2(al, bl, coutl, suml, cout2);
27 endmodule
```

• 2-bit Adder Testbench:

```
23 module testbench4();
         logic a0, b0, a1, b1, cin0, sum0, sum1, cout2;
         twoBitAdder(a0, b0, a1, b1, cin0, sum0, sum1, cout2);
26 🖯
         initial begin
           a0 = 0; b0 = 0; a1 = 0; b1 = 0; cin0 = 0; #10;
28
            a0 = 0; b0 = 0; a1 = 0; b1 = 0; cin0 = 1; #10;
29
            a0 = 0; b0 = 0; a1 = 0; b1 = 1; cin0 = 0; #10;
            a0 = 0; b0 = 0; a1 = 1; b1 = 0; cin0 = 0; #10;
31
             a0 = 0; b0 = 1; a1 = 0; b1 = 0; cin0 = 0; #10;
32
             a0 = 0; b0 = 0; a1 = 0; b1 = 1; cin0 = 1; #10;
            a0 = 0; b0 = 0; a1 = 1; b1 = 0; cin0 = 1; #10;
33
34
             a0 = 0; b0 = 1; a1 = 0; b1 = 0; cin0 = 1; #10;
35
             a0 = 0; b0 = 0; a1 = 1; b1 = 1; cin0 = 1; #10;
            a0 = 0; b0 = 1; a1 = 1; b1 = 0; cin0 = 1; #10;
36
37
             a0 = 0; b0 = 1; a1 = 0; b1 = 1; cin0 = 1; #10;
             a0 = 0; b0 = 1; a1 = 1; b1 = 1; cin0 = 0; #10;
39
            a0 = 0; b0 = 1; a1 = 1; b1 = 1; cin0 = 1; #10;
             a0 = 0; b0 = 0; a1 = 1; b1 = 1; cin0 = 0; #10;
41
             a0 = 0; b0 = 1; a1 = 0; b1 = 1; cin0 = 0; #10;
            a0 = 0; b0 = 1; a1 = 1; b1 = 0; cin0 = 0; #10;
42
            a0 = 1; b0 = 0; a1 = 0; b1 = 0; cin0 = 0; #10;
44
             a0 = 1; b0 = 0; a1 = 0; b1 = 0; cin0 = 1; #10;
45
             a0 = 1; b0 = 0; a1 = 0; b1 = 1; cin0 = 0; #10;
            a0 = 1; b0 = 0; a1 = 1; b1 = 0; cin0 = 0; #10;
47
             a0 = 1; b0 = 1; a1 = 0; b1 = 0; cin0 = 0; #10;
48
             a0 = 1; b0 = 0; a1 = 0; b1 = 1; cin0 = 1; #10;
            a0 = 1; b0 = 0; a1 = 1; b1 = 0; cin0 = 1; #10;
50
             a0 = 1; b0 = 1; a1 = 0; b1 = 0; cin0 = 1; #10;
51
             a0 = 1; b0 = 0; a1 = 1; b1 = 1; cin0 = 0; #10;
            a0 = 1; b0 = 1; a1 = 0; b1 = 1; cin0 = 0; #10;
53
             a0 = 1; b0 = 1; a1 = 1; b1 = 0; cin0 = 0; #10;
             a0 = 1; b0 = 1; a1 = 1; b1 = 1; cin0 = 0; #10;
55
            a0 = 1; b0 = 1; a1 = 1; b1 = 0; cin0 = 1; #10;
             a0 = 1; b0 = 1; a1 = 0; b1 = 1; cin0 = 1; #10;
             a0 = 1; b0 = 0; a1 = 1; b1 = 1; cin0 = 1; #10;
             a0 = 1; b0 = 1; a1 = 1; b1 = 1; cin0 = 1; #10;
58
59 📄
60 endmodule
```

• Structural Lab Calculator Module:

```
3 // Company:
   // Engineer:
5
6 | // Create Date: 16.10.2022 16:13:23
7 : // Design Name:
    // Module Name: labCalculator
8
9
   // Project Name:
   // Target Devices:
10
11
  // Tool Versions:
12 // Description:
13
14
   // Dependencies:
15
16
  // Revision:
17
  ! // Revision 0.01 - File Created
18 / // Additional Comments:
19 //
22
23 module labCalculator(input logic a, b, c, d, output logic y, z);
24
      logic i0, i1, i2, i3;
25
      assign i0 = a ^ b;
26
       assign il = \sim (a & b);
27
       halfAdder adder(a, b, i2, z);
28
       halfSubtractor subtractor(a, b, i3, z);
29
       mux41 mux(c, d, i0, i1, i2, i3, y);
30 endmodule
                   Lab Calculator Testbench:
     // Revision 0.01 - File Created
18
    // Additional Comments:
19
22
23   module testbench5();
24
        logic a, b, c, d, y, z;
25
        labCalculator labCalc(a, b, c, d, y, z);
        initial begin
26 🖯
27
            c = 0; d = 0; a = 0; b = 0; \#10;
28
            c = 0; d = 0; a = 0; b = 1; #10;
29
            c = 0; d = 0; a = 1; b = 0; $10;
            c = 0; d = 0; a = 1; b = 1; \#10;
30
31
            c = 0; d = 1; a = 0; b = 0; $10;
32
            c = 0; d = 1; a = 0; b = 1; #10;
33
            c = 0; d = 1; a = 1; b = 0; $10;
34
            c = 0; d = 1; a = 1; b = 1; #10;
35
            c = 1; d = 0; a = 0; b = 0; $10;
36
            c = 1; d = 0; a = 0; b = 1; #10;
            c = 1; d = 0; a = 1; b = 0; \#10;
37
38
            c = 1; d = 0; a = 1; b = 1; #10;
39
            c = 1; d = 1; a = 0; b = 0; $10;
40
            c = 1; d = 1; a = 0; b = 1; \#10;
41
            c = 1; d = 1; a = 1; b = 0; #10;
            c = 1; d = 1; a = 1; b = 1; #10;
42
43
        end
44 endmodule
```