ES204 Digital Systems
LAB Assignment - 7

Indian Institute of Technology, Gandhinagar
February 28, 2024

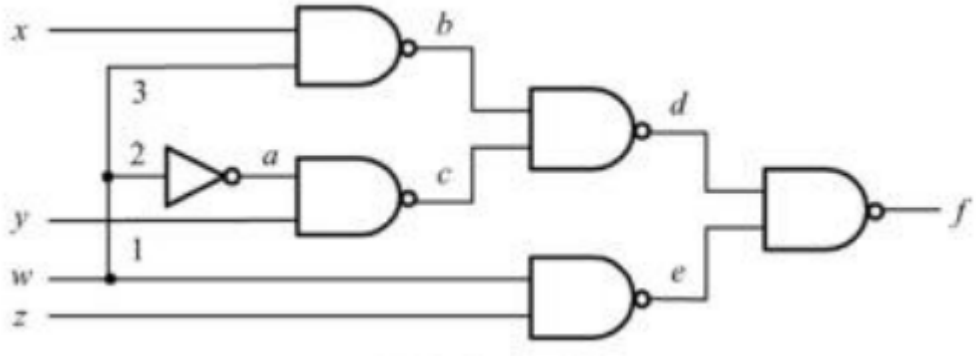
Marks : 30

Submission instructions:

Only one student from the team will submit with the word doc name Rollno1_Rollno2.pdf. The PDF will contain the code, testbench and simulation results.

1. Implement the following circuit in Verilog in such a way that all the hazards can be viewed. Write the testbench such that all the static and dynamic hazards in the following circuit can be shown on the simulator. You can use appropriate delay values.

   Implement the circuit as is, then look for static hazards in the circuit. If there are static hazards, then only you can also have Dynamic hazards. For our case, only ONE input change should give rise to static and dynamic hazard.



Lab 7 ES 204

Birudugadda Srivibhav (22110050)
Reddybathuni Venkat (22110220)

#Static Hazard

1. Code

```
`timescale 1ns / 1ps
module static(input x, y, w, z, output a, b, c, d ,e, f);

    not #(1) n1(a, w);
    nand #(1) g1(b, x, w);
    nand #(1) g2(c, a, y);
    nand #(1) g3(d, b, c);
    nand #(1) g4(e, w, z);
    nand #(1) g5(f, d, e);

endmodule
```

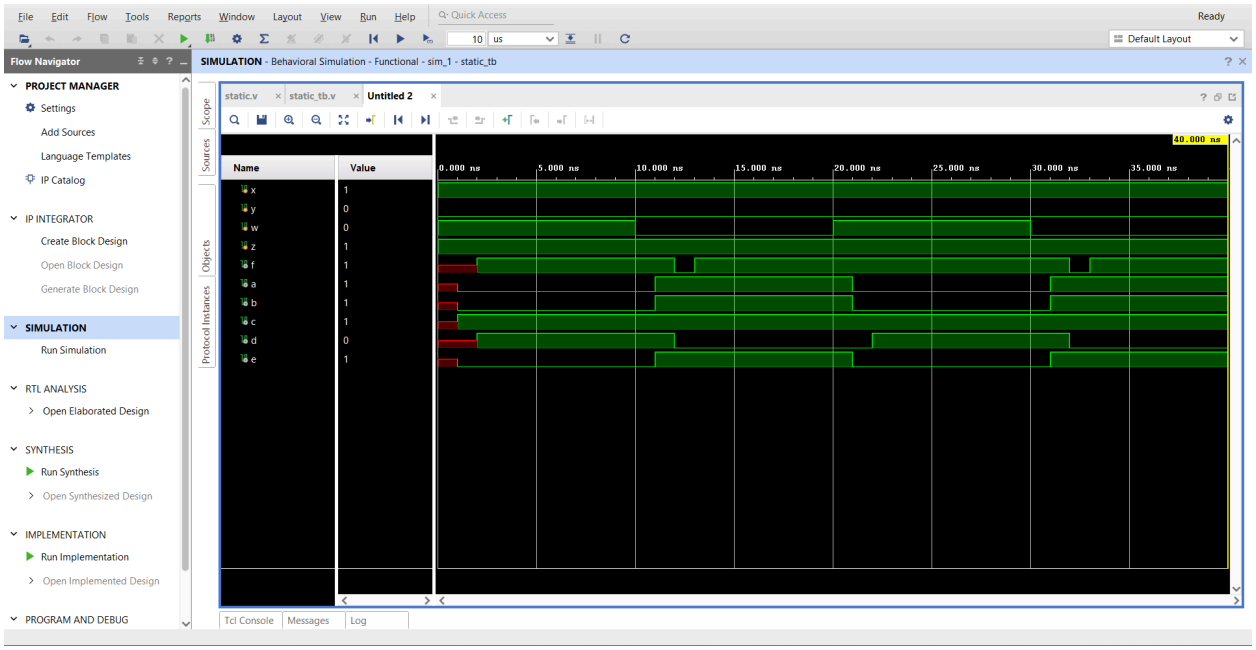
2. Test bench

```
`timescale 1ns / 1ps
module static_tb();
reg x, y, w, z;
wire f;

static uut(.x(x), .y(y), .w(w), .z(z), .a(a), .b(b), .c(c), .d(d), .e(e), .f(f));
    initial
    begin
//    x = 1; y = 0; w = 1; z = 1;
//    #10;
//    x = 1; y = 0; w = 0; z = 1;
//    #10;
//    x = 1; y = 0; w = 1; z = 1;
//    #10;
//    x = 1; y = 0; w = 0; z = 1;
```

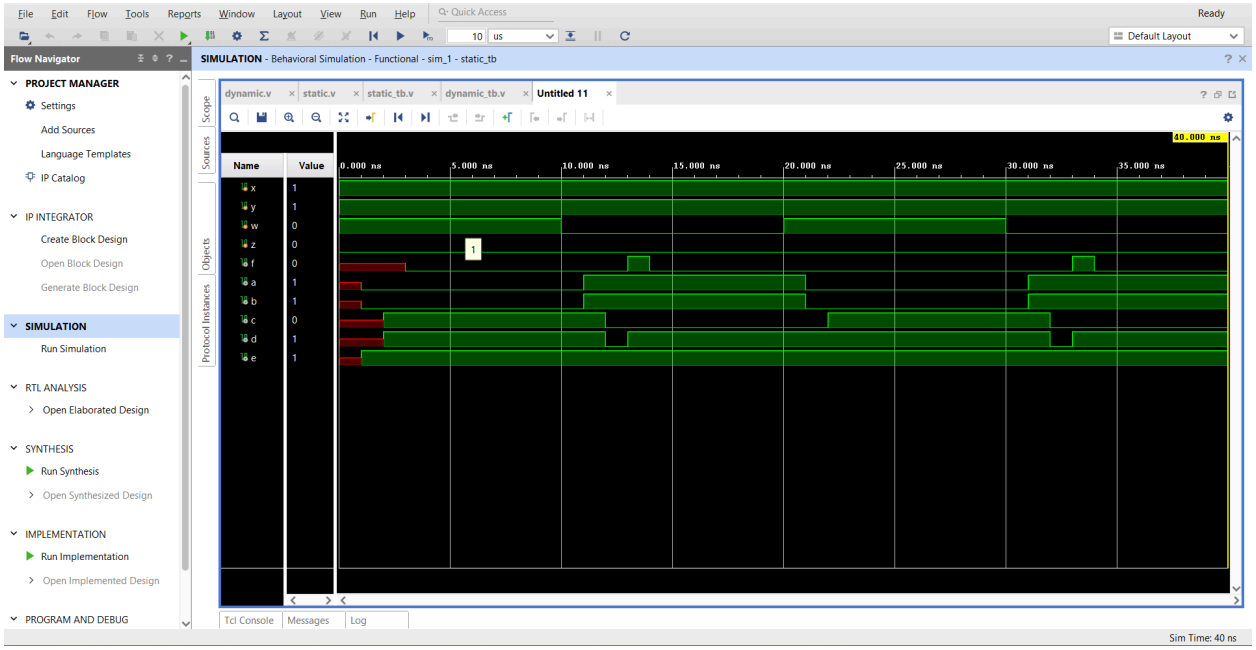
```
//    #10;


    x = 1; y = 1; w = 1; z = 0;
    #10;
    x = 1; y = 1; w = 0; z = 0;
    #10;
    x = 1; y = 1; w = 1; z = 0;
    #10;
    x = 1; y = 1; w = 0; z = 0;
    #10;

    $finish();
    end
endmodule
```

3. Simulation

Static 1 hazard for
x = 1; y = 0; z = 1;
w: 1 to 0;



Static 0 hazard for
x = 1; y = 1; z = 0;
w: 1 to 0;



#Dynamic Hazard

4. Code

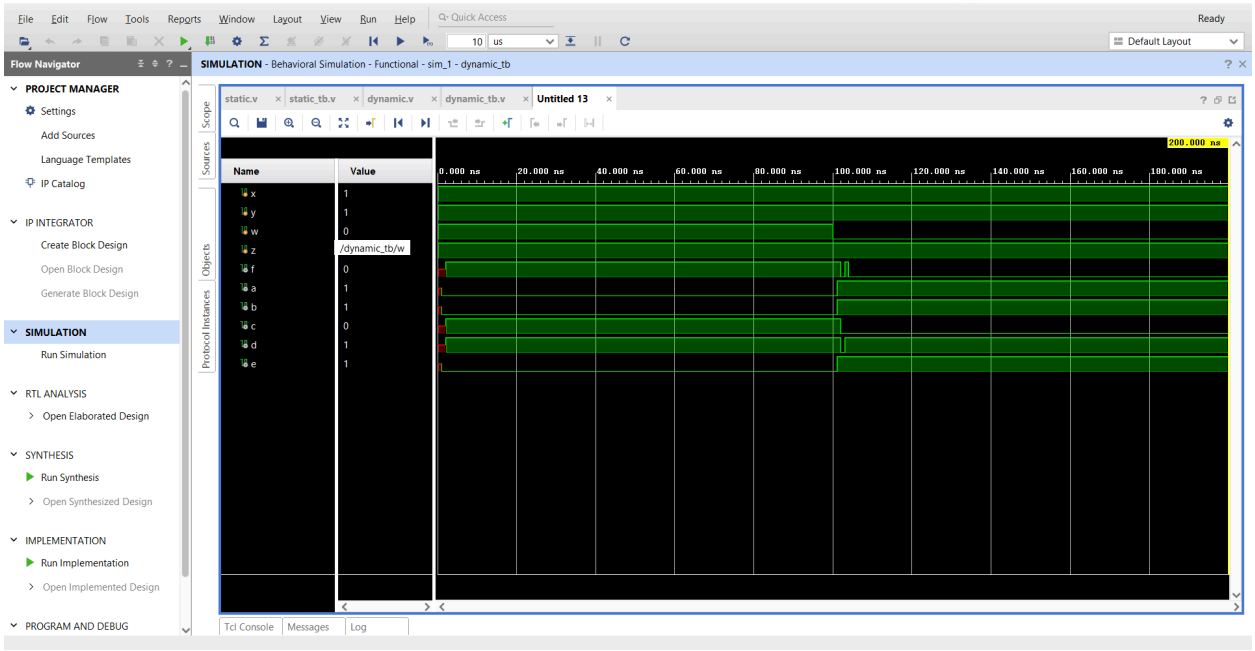
```
`timescale 1ns / 1ps
module dynamic(input x, y, w, z, output a, b, c, d ,e, f);

    not #(1) n1(a, w);
    nand #(1) g1(b, x, w);
    nand #(1) g2(c, a, y);
    nand #(1) g3(d, b, c);
    nand #(1) g4(e, w, z);
    nand #(1) g5(f, d, e);

endmodule
```

5. Test Bench

```
`timescale 1ns / 1ps
module dynamic_tb();
reg x, y, w, z;
wire f;

dynamic uut(.x(x), .y(y), .w(w), .z(z), .a(a), .b(b), .c(c), .d(d), .e(e), .f(f));
    initial
    begin
    x = 1; y = 1; w = 1; z = 1;
    #100;
    x = 1; y = 1; w = 0; z = 1;
    #100;
    $finish();
    end
endmodule
```

6. Simulation

Dynamic hazard for
x = 1; y = 1; z = 1;
w: 1 to 0;



#Without Delays

7. Code

```
`timescale 1ns / 1ps
module static(input x, y, w, z, output a, b, c, d ,e, f);

    not n1(a, w);
    nand g1(b, x, w);
    nand g2(c, a, y);
    nand g3(d, b, c);
    nand g4(e, w, z);
    nand g5(f, d, e);

endmodule
```

8. Test Bench

```
`timescale 1ns / 1ps
module static_tb();
reg x, y, w, z;
wire f;

static uut(.x(x), .y(y), .w(w), .z(z), .a(a), .b(b), .c(c), .d(d), .e(e), .f(f));
    initial
    begin
//    x = 1; y = 0; w = 1; z = 1;
//    #10;
//    x = 1; y = 0; w = 0; z = 1;
//    #10;
//    x = 1; y = 0; w = 1; z = 1;
//    #10;
//    x = 1; y = 0; w = 0; z = 1;
//    #10;


    x = 1; y = 1; w = 1; z = 0;
    #10;
```

```
    x = 1; y = 1; w = 0; z = 0;
    #10;
    x = 1; y = 1; w = 1; z = 0;
    #10;
    x = 1; y = 1; w = 0; z = 0;
    #10;

    $finish();
    end
endmodule
```

9. Simulation