به نام خدا

UNIVERSITY OF TEHRAN
Electrical and Computer Engineering Department

Computer Assignment 5

Digital Systems I - ECE 894

Deadline : 23 Khordad 1400

Erfan Panahi

810198369

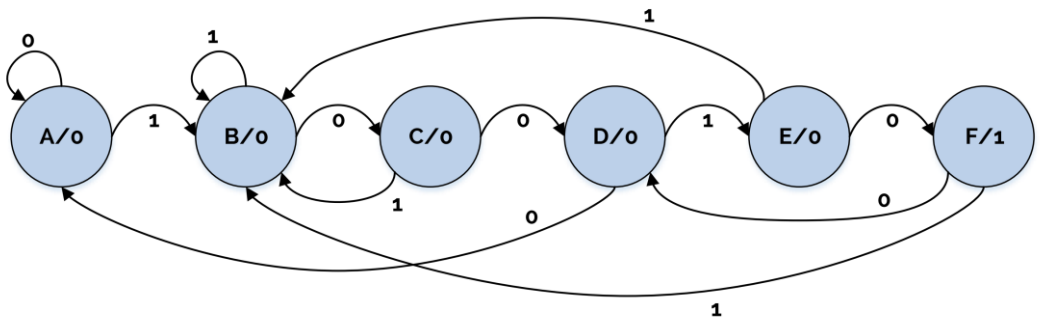
Spring 1400

List of Problems:.

Computer Assignment 5

Problem a. Moore 10010 detector

A. State diagram:



B. State table:

	state	$j=0$	$j=1$	w
000	A	A	B	0
001	B	C	B	0
010	C	D	B	0
011	D	A	E	0
100	E	F	B	0
101	F	B	D	1
		$state^+$		

C. SystemVerilog description:

File: Moore10010.v

```
`timescale 1ns/1ns
module moore10010 (input clk,rst,j, output w);
    reg [2:0] ns,ps;
    parameter [2:0] A=3'd0, B=3'd1, C=3'd2, D=3'd3, E=3'd4, F=3'd5;
    always @ (ps,j) begin
        ns = A;
        case(ps)
            A: ns = j ? B : A;
            B: ns = j ? B : C;
            C: ns = j ? B : D;
            D: ns = j ? E : A;
            E: ns = j ? B : F;
            F: ns = j ? D : B;
            default: ns = A;
        endcase
    end
    assign w = (ps==F) ? 1'b1 : 1'b0;
    always @ (posedge clk,posedge rst) begin
        if(rst) ps <= A;
        else ps <= ns;
    end
endmodule
```

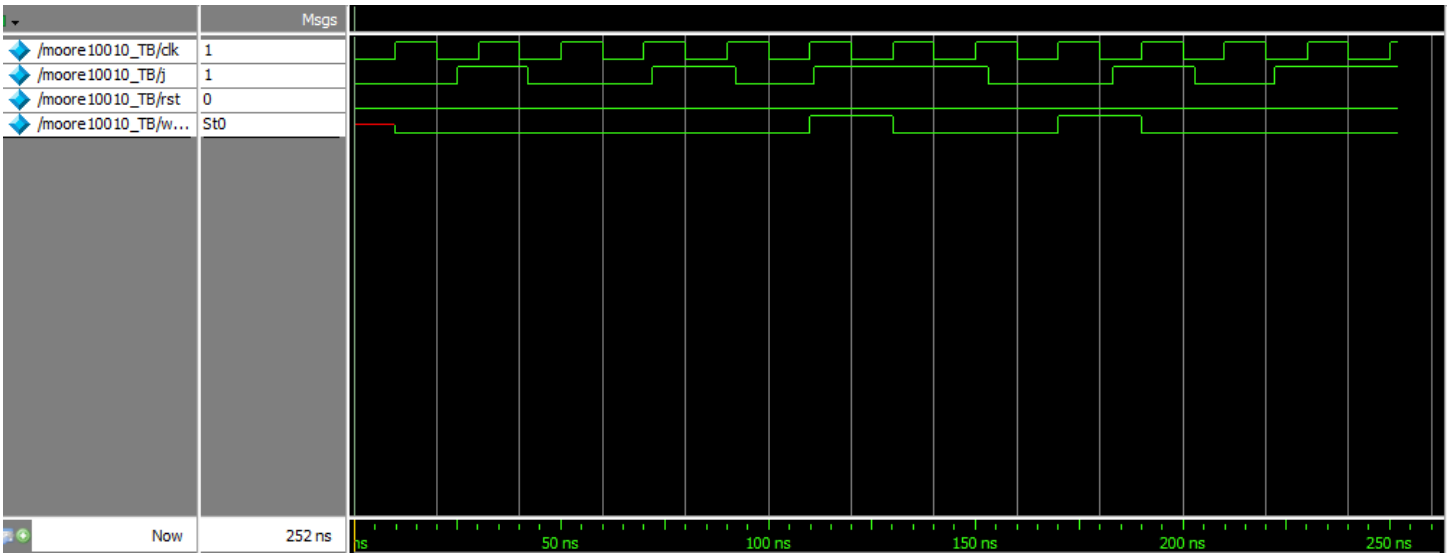
Part i. pre-synthesis description

A. TestBench:

File: Moore10010_TB.v

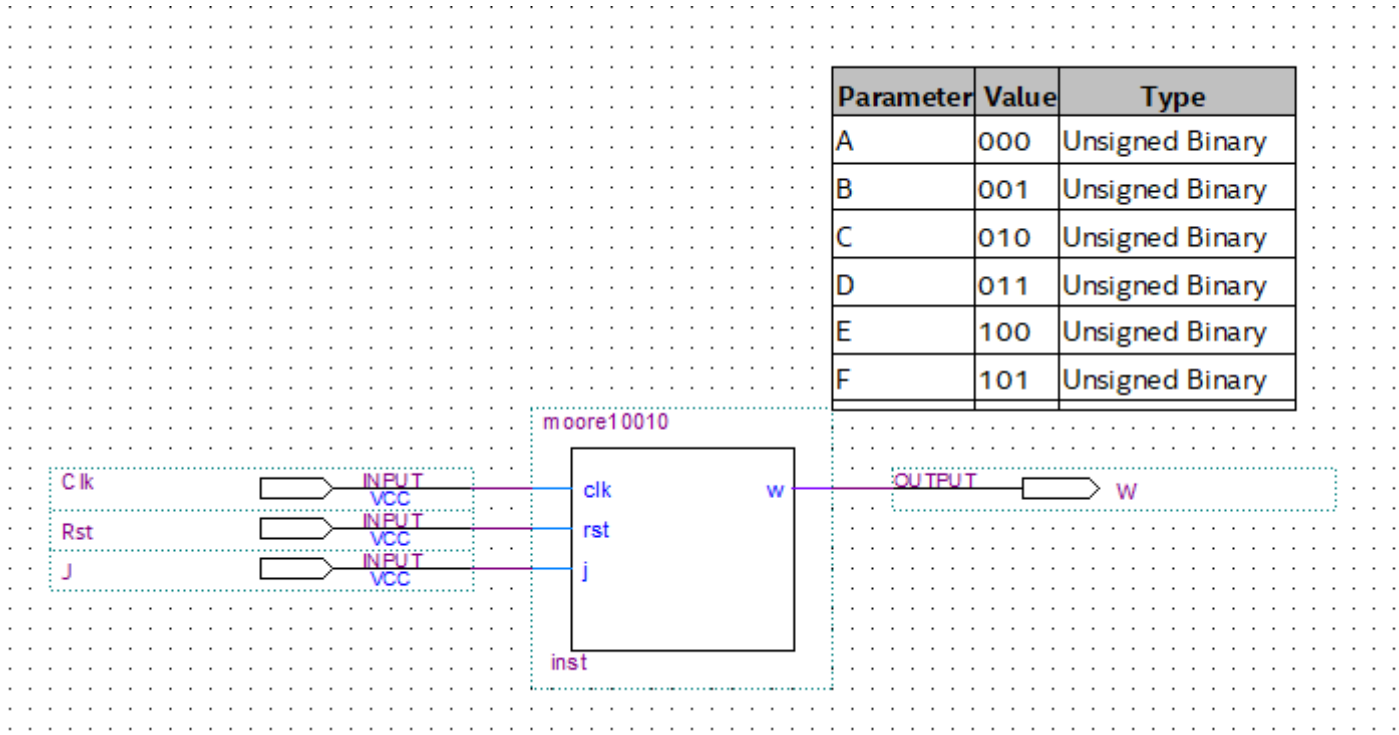
```
`timescale 1ns/1ns
module moore10010_TB ();
    reg clk = 0, j = 0, rst = 0;
    wire w_moore;
    moore10010 UUT1(clk,rst,j,w_moore);
    always #10 clk <= ~clk;
    initial begin
    #25 j = 1;
    #17 j = 0;
    #15 j = 0;
    #15 j = 1;
    #20 j = 0;
    #19 j = 1;
    #25 j = 1;
    #17 j = 0;
    #15 j = 0;
    #15 j = 1;
    #20 j = 0;
    #19 j = 1;
    #30 $stop;
    end
endmodule
```

B. Waveform:



Part ii. post-synthesis description

A. Symbol: File: moore10010Blk.bdf



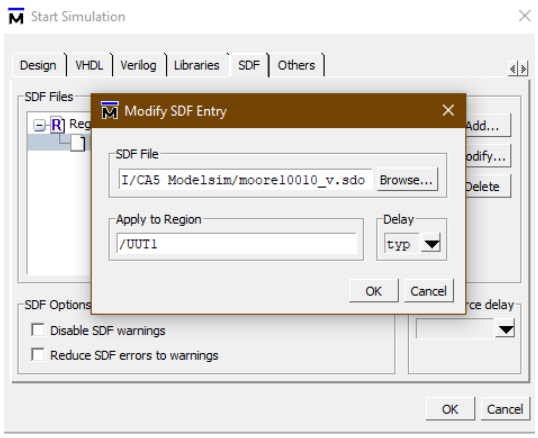
B. Timing:

*** We change the name of module .vo ('Moore10010') to 'Moore10010Q'.

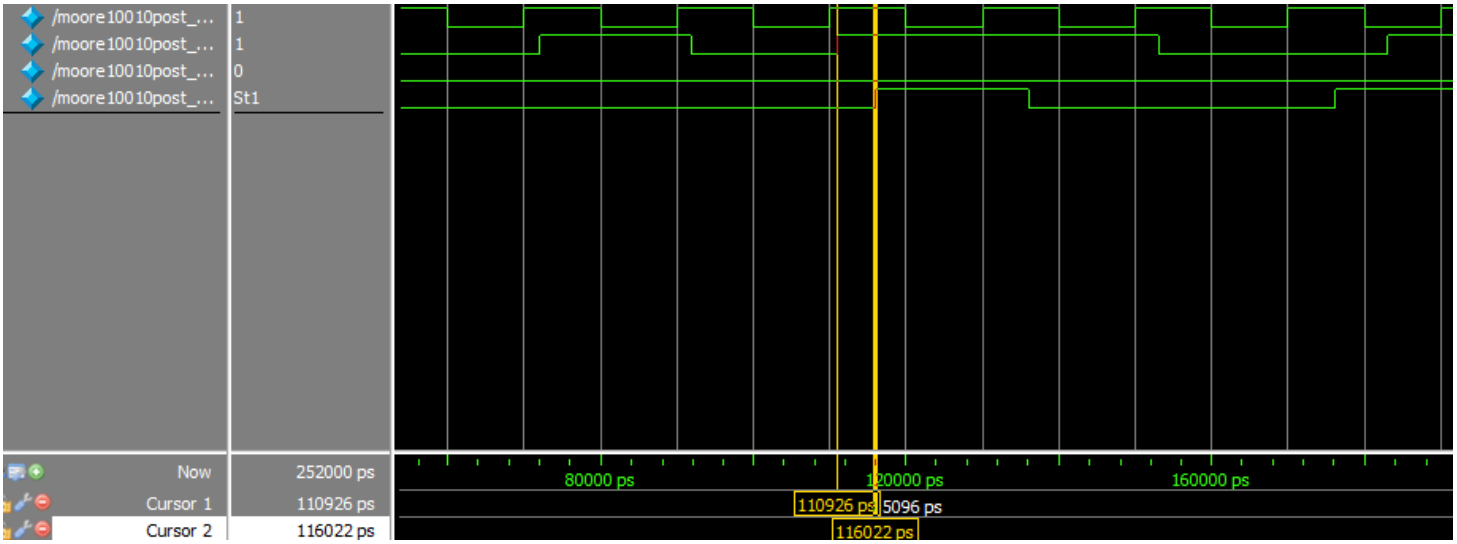
File: Moore10010post_TBii.v

```
`timescale 1ns/1ns
module moore10010post_TBii ();
    reg clk = 0, j = 0, rst = 0;
    wire w_moore;
    moore10010Q UUT1(clk,rst,j,w_moore);
    always #10 clk <= ~clk;
    initial begin
    #25 j = 1;
    #17 j = 0;
    #15 j = 0;
    #15 j = 1;
    #20 j = 0;
    #19 j = 1;
    #25 j = 1;
    #17 j = 0;
    #15 j = 0;
    #15 j = 1;
    #20 j = 0;
    #19 j = 1;
    #30 $stop;
    end
endmodule
```

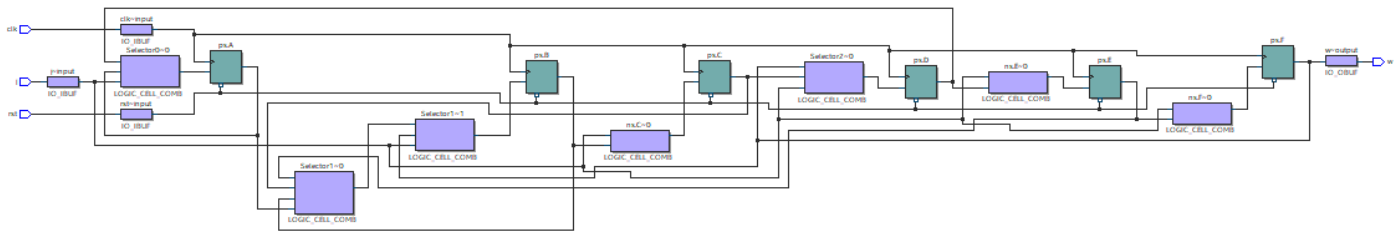
When we simulate the testbench we add .sdo file to SDF:



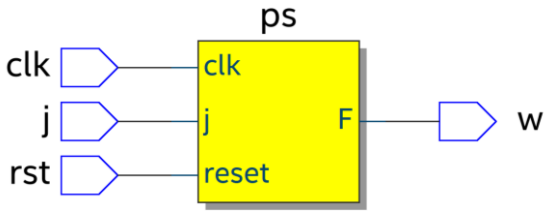
And finally the output waveform is shown in below figure. As we see there is a delay when we change the input and output changes.

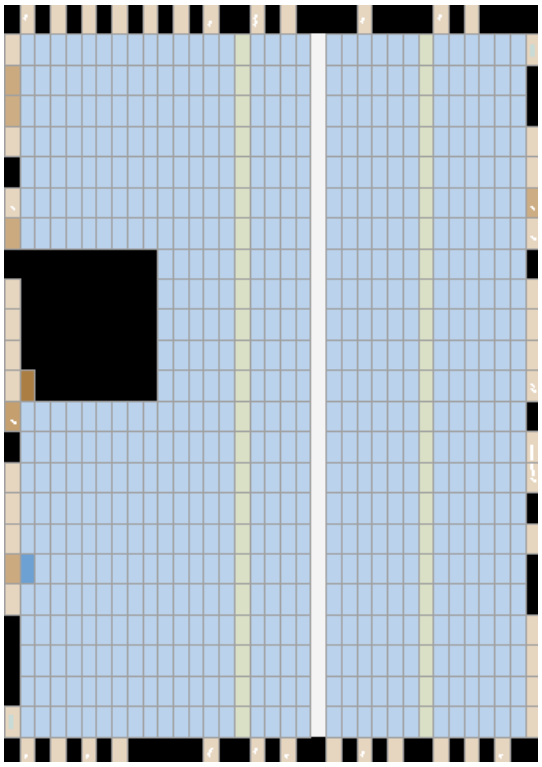


C. Cell used:



D. Floor Plan:





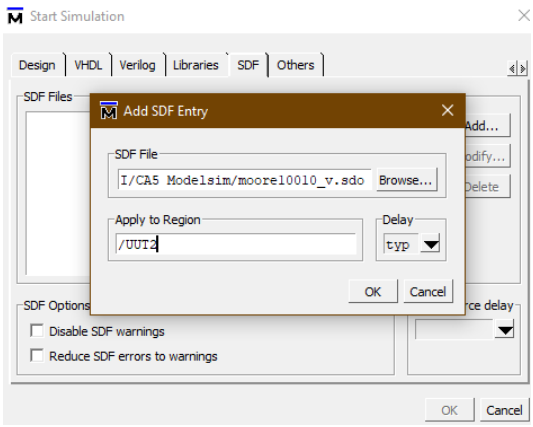
Part iii.

A. TestBench:

File: Moore10010_TBiii.sv

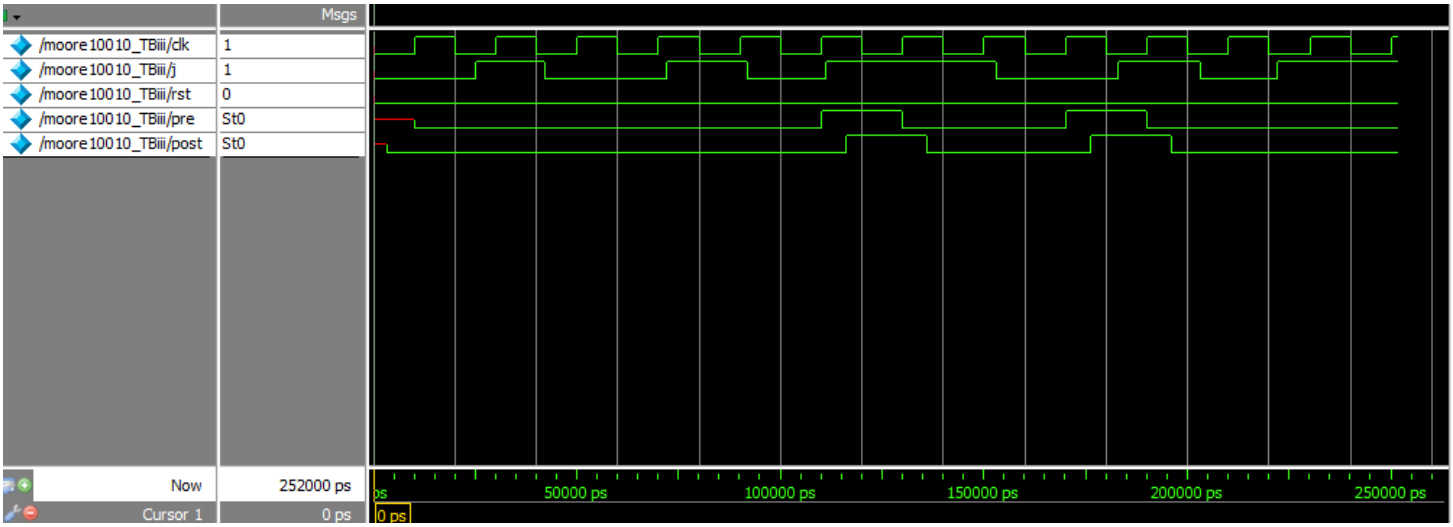
```
`timescale 1ns/1ns
module moore10010_TBiii ();
    reg clk = 0, j = 0, rst = 0;
    wire pre,post;
    moore10010 UUT1(clk,rst,j,pre);
    moore10010Q UUT2(clk,rst,j,post);
    always #10 clk <= ~clk;
    initial begin
    #25 j = 1;
    #17 j = 0;
    #15 j = 0;
    #15 j = 1;
    #20 j = 0;
    #19 j = 1;
    #25 j = 1;
    #17 j = 0;
    #15 j = 0;
    #15 j = 1;
    #20 j = 0;
    #19 j = 1;
    #30 $stop;
    end
endmodule
```

When we simulate the testbench we add .sdo file to SDF:



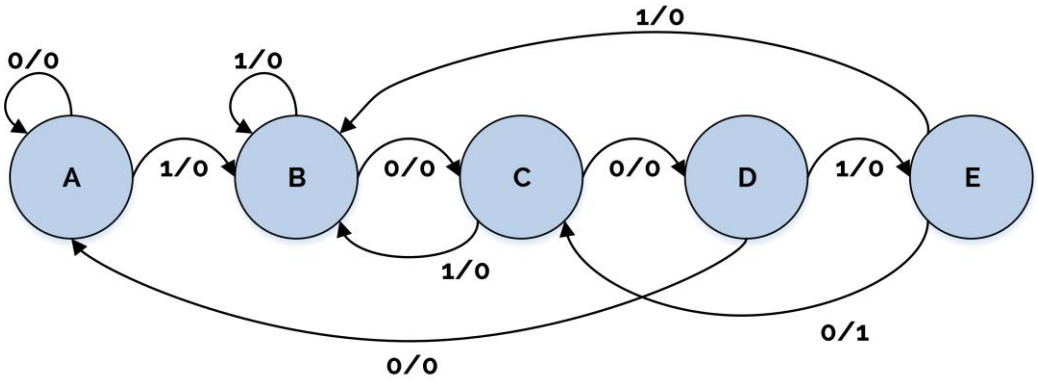
B. Waveform:

As we see waveform in below figure, the post-synthesis has a delay to change its output. Because of the gate used in this structure. But we do not see this case in pre-synthesis.



Problem b. Mealy 10010 detector

A. State diagram:



B. State table:

	state	$j=0$	$j=1$	w
000	A	A	B	0
001	B	C	B	0
010	C	D	B	0
011	D	A	E	0
100	E	C	B	1
		$state^+$		

C. SystemVerilog description:

File: Mealy10010.v

```
`timescale 1ns/1ns
module mealy10010 (input clk,rst,j, output w);
    reg [2:0] ns,ps;
    parameter [2:0] A=3'd0, B=3'd1, C=3'd2, D=3'd3, E=3'd4;
    always @ (ps,j) begin
        ns = A;
        case(ps)
            A: ns = j ? B : A;
            B: ns = j ? B : C;
            C: ns = j ? B : D;
            D: ns = j ? E : A;
            E: ns = j ? B : C;
            default: ns = A;
        endcase
    end
    assign w = (ps==E) ? ~j : 1'b0;
    always @ (posedge clk,posedge rst) begin
        if(rst) ps <= A;
        else ps <= ns;
    end
endmodule
```

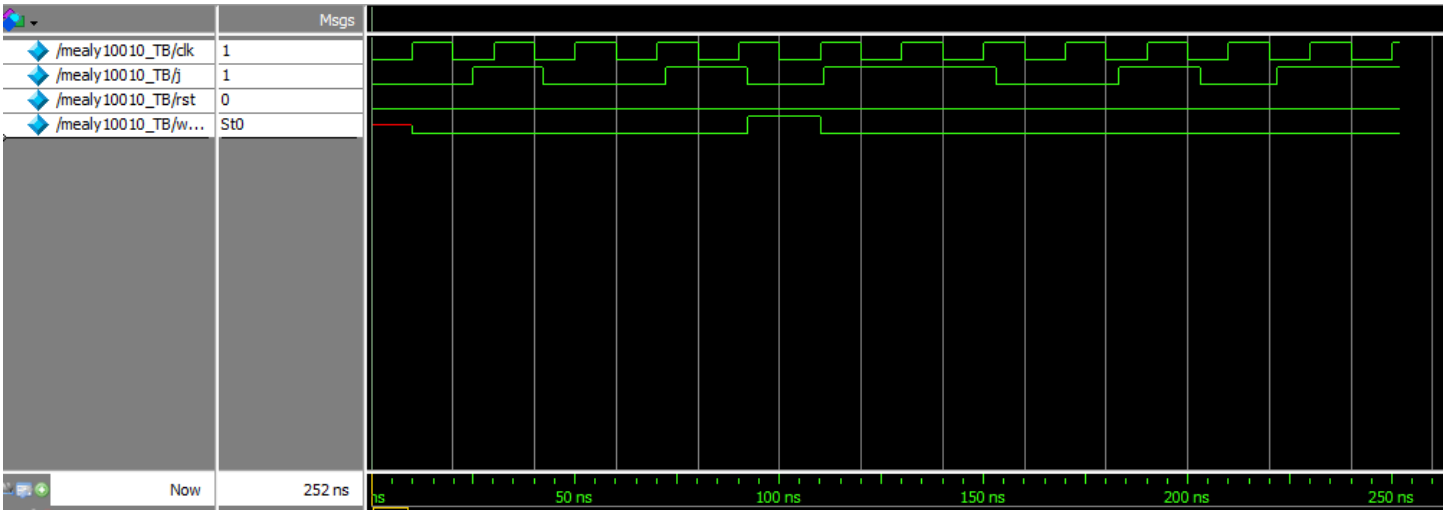
Part i. pre-synthesis description

A. TestBench:

File: Mealy10010_TB.v

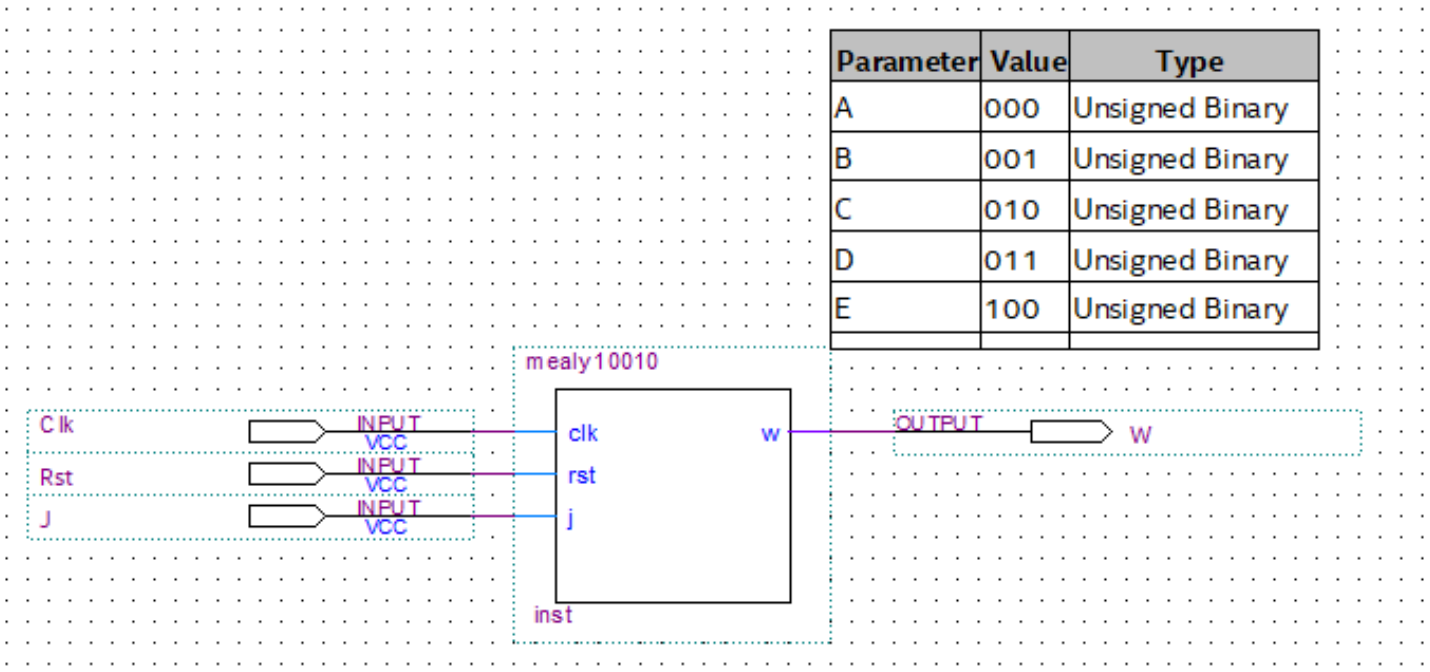
```
`timescale 1ns/1ns
module mealy10010_TB ();
    reg clk = 0, j = 0, rst = 0;
    wire w_mealy;
    mealy10010 UUT1(clk,rst,j,w_mealy);
    always #10 clk <= ~clk;
    initial begin
    #25 j = 1;
    #17 j = 0;
    #15 j = 0;
    #15 j = 1;
    #20 j = 0;
    #19 j = 1;
    #25 j = 1;
    #17 j = 0;
    #15 j = 0;
    #15 j = 1;
    #20 j = 0;
    #19 j = 1;
    #30 $stop;
    end
endmodule
```

B. Waveform:



Part ii. post-synthesis description

A. Symbol: File: mealy10010Blk.bdf



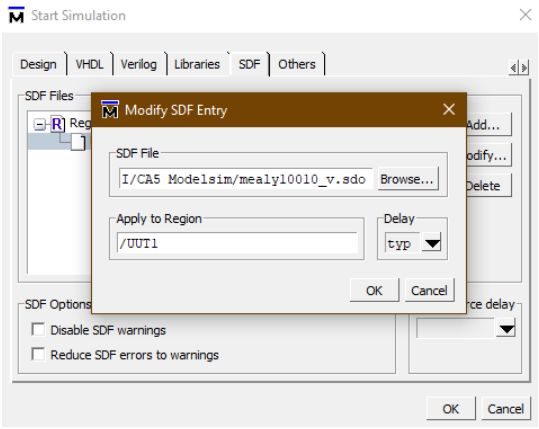
B. Timing:

*** We change the name of module .vo ('Mealy10010') to 'Mealy10010Q'.

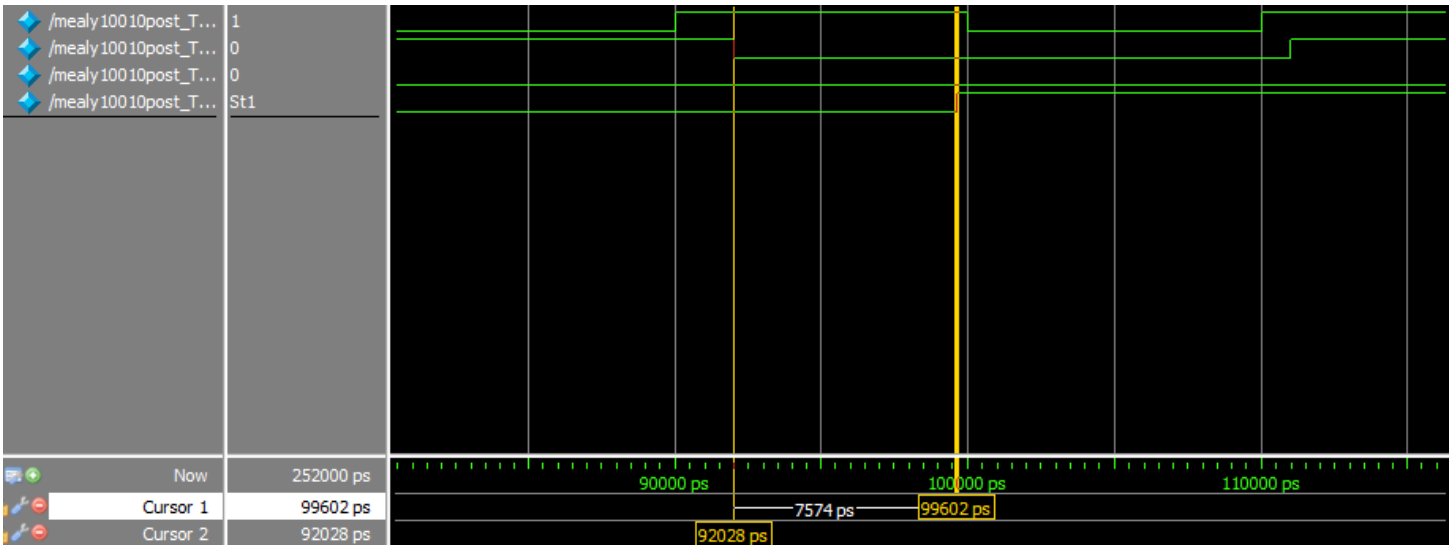
File: Mealy10010post_TBii.v

```
`timescale 1ns/1ns
module mealy10010post_TBii ();
    reg clk = 0, j = 0, rst = 0;
    wire w_mealy;
    mealy10010Q UUT1(clk,rst,j,w_mealy);
    always #10 clk <= ~clk;
    initial begin
    #25 j = 1;
    #17 j = 0;
    #15 j = 0;
    #15 j = 1;
    #20 j = 0;
    #19 j = 1;
    #25 j = 1;
    #17 j = 0;
    #15 j = 0;
    #15 j = 1;
    #20 j = 0;
    #19 j = 1;
    #30 $stop;
    end
endmodule
```

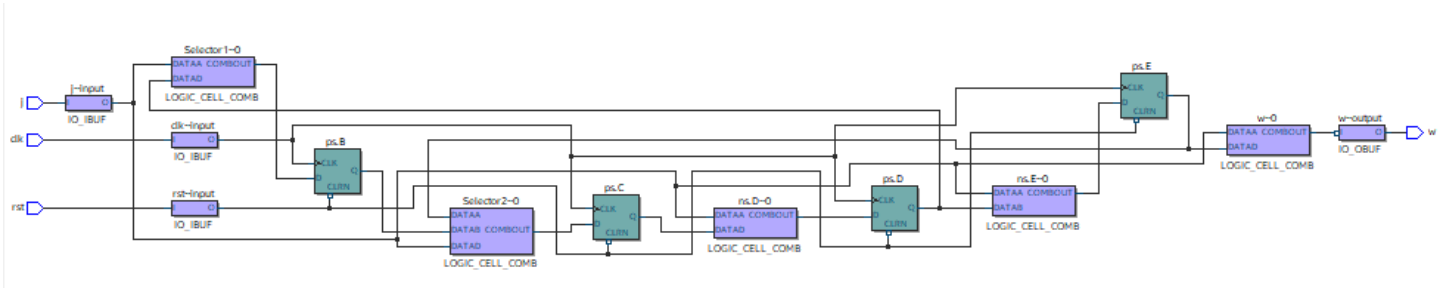
When we simulate the testbench we add .sdo file to SDF:



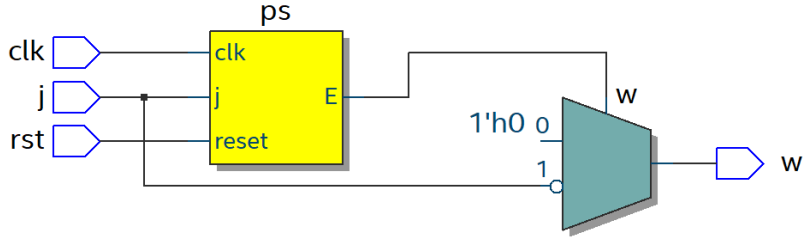
And finally the output waveform is shown in below figure. As we see there is a delay when we change the input and output changes.

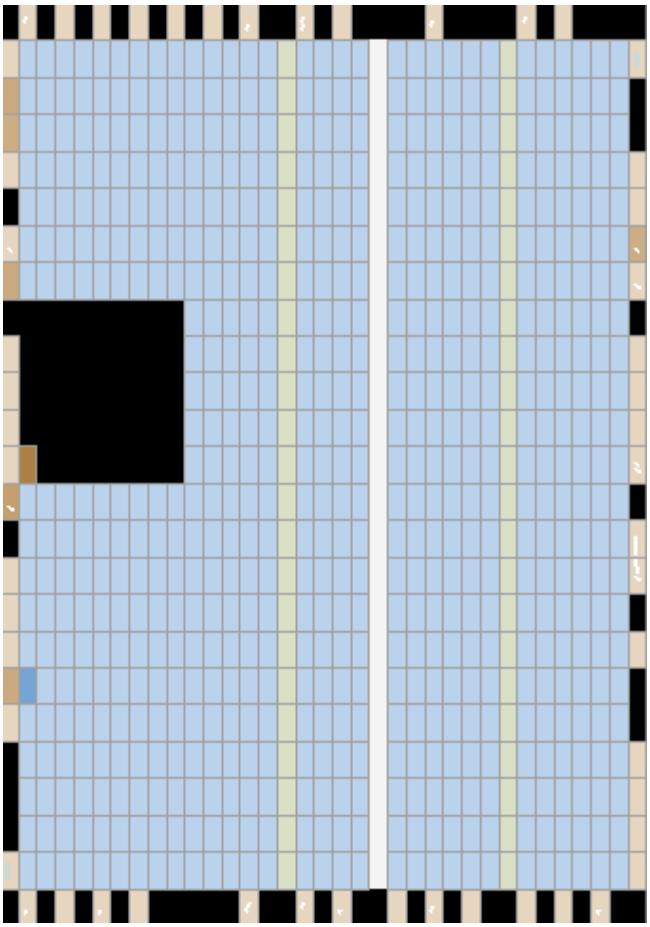


C. Cell used:



D. Floor plan:



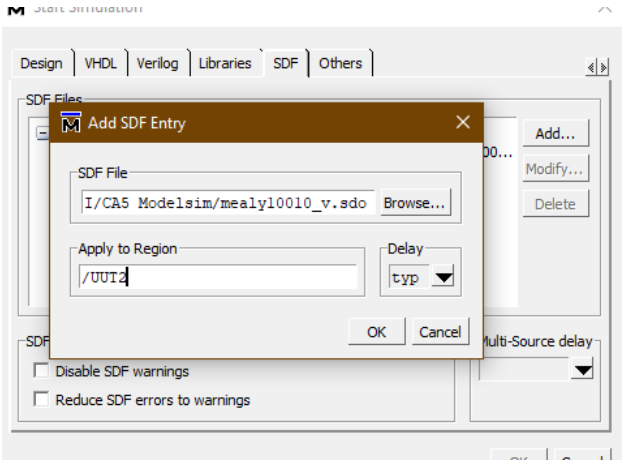


Part iii.

A. TestBench:

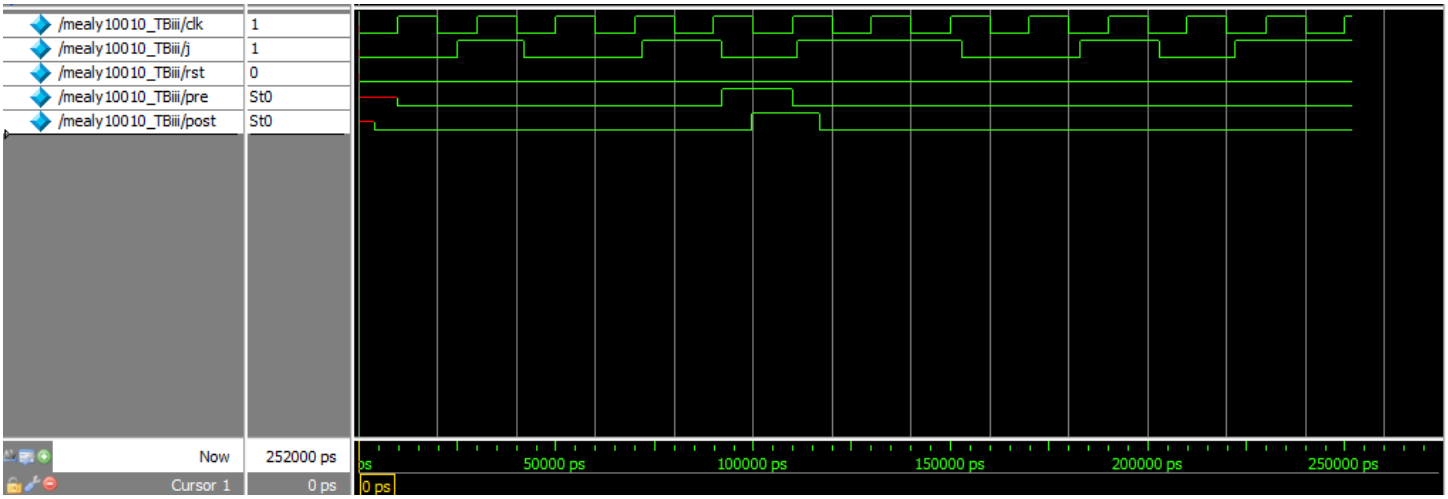
File: Mealy10010_TBiii.v

When we simulate the testbench we add .sdo file to SDF:



B. Waveform:

As we see waveform in below figure, the post-synthesis has a delay to change its output. Because of the gate used in this structure. But we do not see this case in pre-synthesis.



Problem c. Mealy 10010 detector

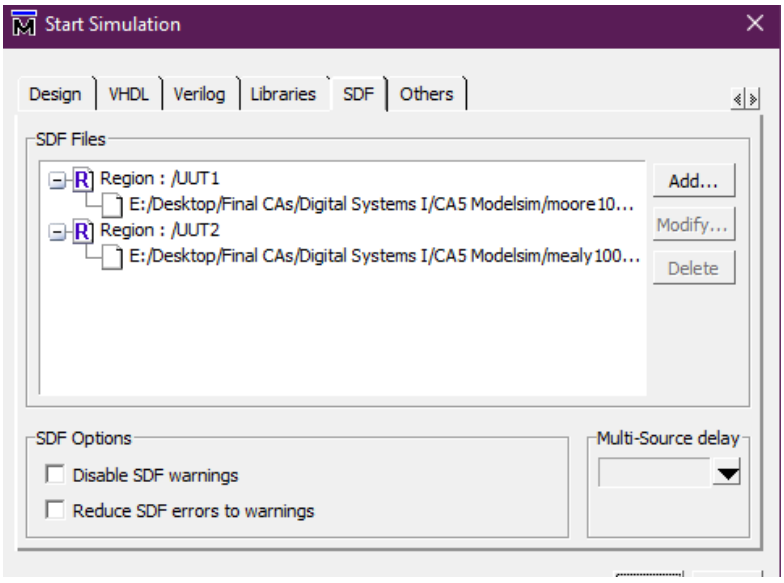
Part i.

A. TestBench:

File: PartC_TBi.v

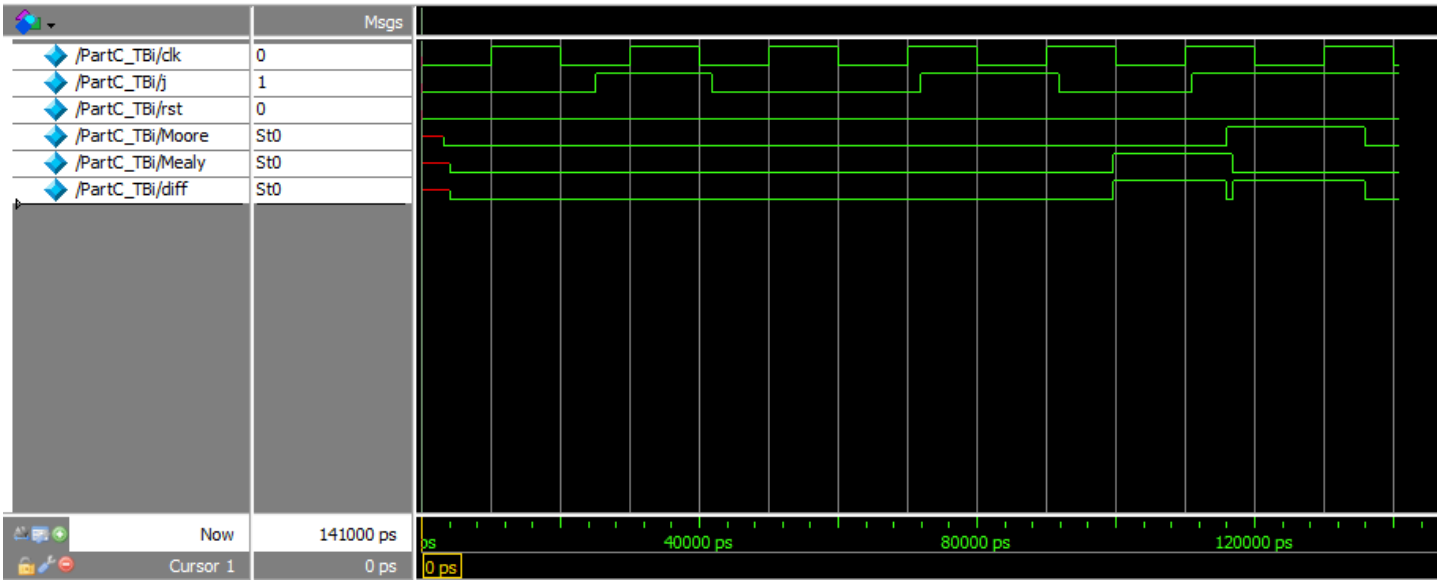
```
`timescale 1ns/1ns
module PartC_TBi ();
    reg clk = 0, j = 0, rst = 0;
    wire Moore,Mealy,diff;
    moore10010Q UUT1(clk,rst,j,Moore);
    mealy10010Q UUT2(clk,rst,j,Mealy);
    assign diff = Moore ^ Mealy;
    always #10 clk <= ~clk;
        initial begin
        #25 j = 1;
        #17 j = 0;
        #15 j = 0;
        #15 j = 1;
        #20 j = 0;
        #19 j = 1;
        #30 $stop;
        end
endmodule
```

When we simulate the testbench we add .sdo files to SDF:



B. Waveform:

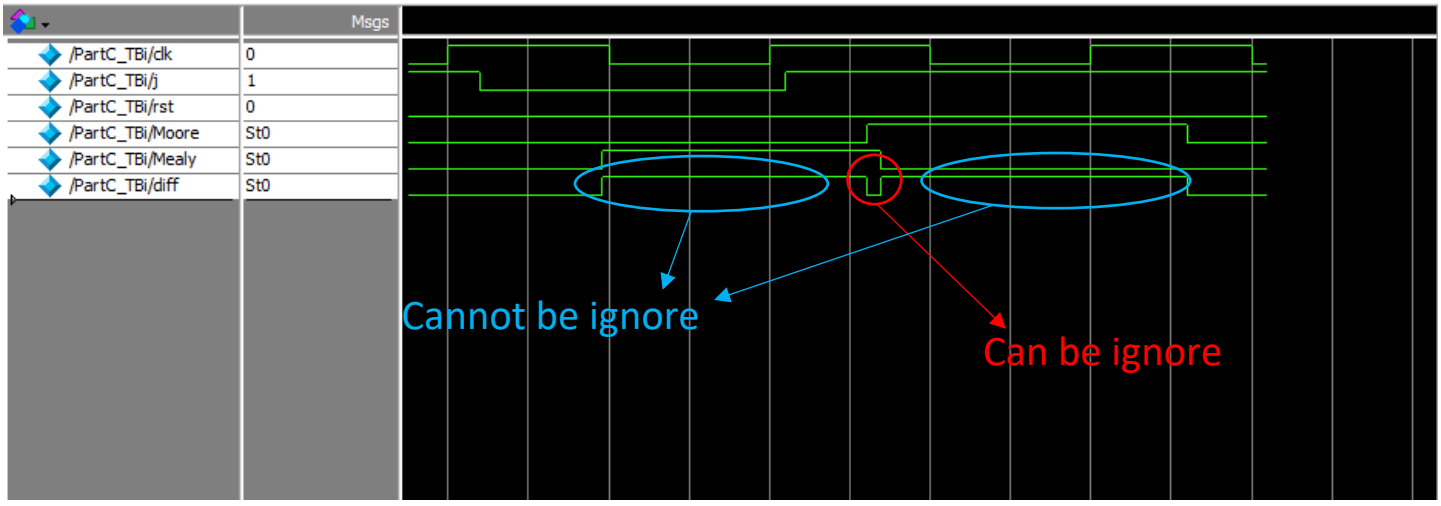
As we see waveform in below figure, the post-synthesis Mealy output changes earlier post-synthesis Moore. Because of the state diagram of this two machines.



Part ii.

Where differences can be ignores that is like a glitch and it depends on the gate delay differences.

Where differences cannot be ignores that is depends on being Mealy or Moore machines. In fact this differences is due to changes on the inputs.



Part iii.

We want to have an output that give us 1 when one of the Moore or Mealy output gets 1 value. So we can use OR gate like below figure:

A. TestBench:

File: PartC_TBiii.v

```
`timescale 1ns/1ns
module PartC_TBiii ();
    reg clk = 0, j = 0, rst = 0;
    wire Moore,Mealy,diff,diff2;
    moore10010Q UUT1(clk,rst,j,Moore);
    mealy10010Q UUT2(clk,rst,j,Mealy);
    assign diff = Moore ^ Mealy;
    assign diff2 = diff | Moore | Mealy;
    always #10 clk <= ~clk;
        initial begin
        #25 j = 1;
        #17 j = 0;
        #15 j = 0;
        #15 j = 1;
        #20 j = 0;
        #19 j = 1;
        #30 $stop;
        end
endmodule
```

B. Waveform: