University of Tehran
College of Engineering
School of Electrical and Computer Engineering

Digital Systems 1

Dr.Navabi

Computer Assignment 5

Soroush Mesforush Mashhad

SN:810198472

Khordad 00

1 Project generated files

A full view of the generated and used SystemVerilog files in my project can be seen below:

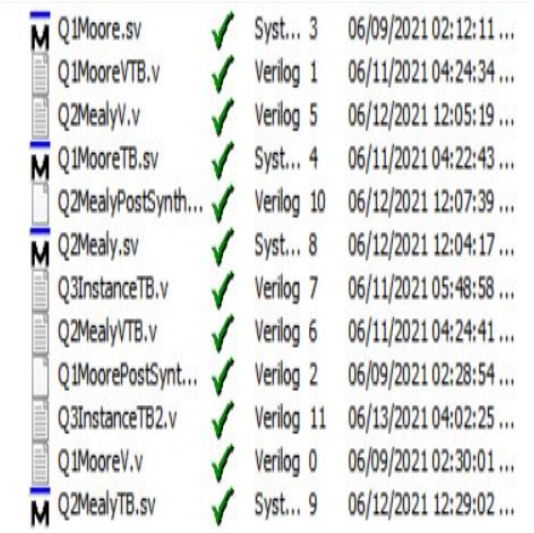


Figure 1: Generated and Compiled .sv files

2 Questions

2.1 A

2.1.1 i

The state diagram can be seen accordingly

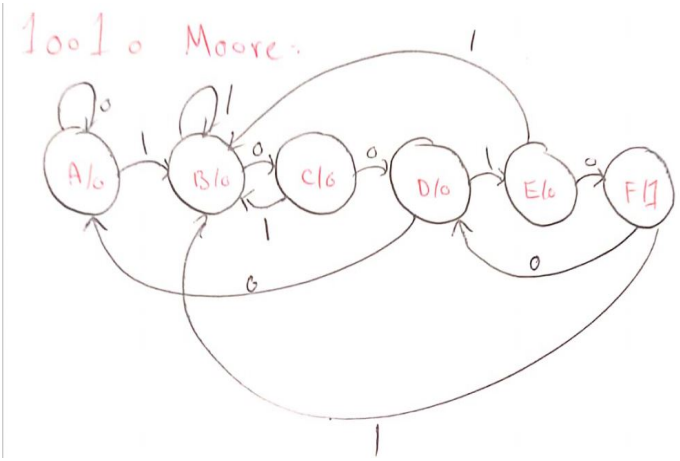


Figure 2: State Diagram

2.1.2 SystemVerilog Code & Testbench

```
`timescale 1ns/1ns
module Q1Moore(input clk,rst,j,output w);
  logic[2:0] ns,ps;
  parameter[2:0] A=3'b000,B=3'b001,C=3'b010,D=3'b011,E=3'b100,F=3'b101;

  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?B:D;
         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
```

```
`timescale 1ns/1ns
module Q1MooreTB();
  logic j,clk=0,rst=0;
  wire Out;
  Q1Moore UUT(clk,rst,j,Out);
  always #25 clk<=~clk;
  initial begin
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #100 $stop;
  end
endmodule
```

Figure 3: SystemVerilog Code & Testbench

2.1.3 Waveform

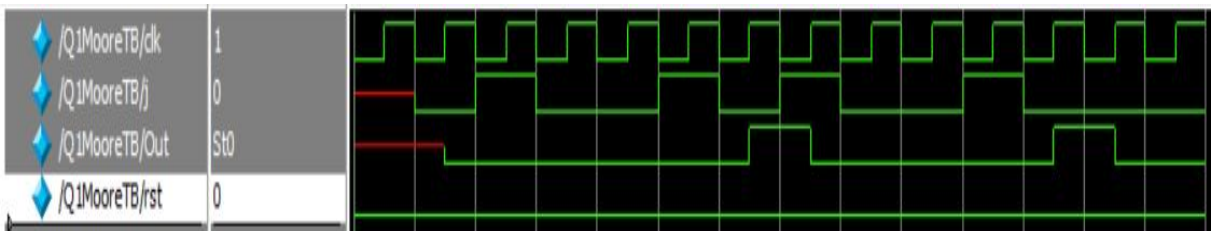


Figure 4: Moore Machine Waveform

2.2 ii

By using Quartus we synthesize the verilog code and obtain the following results.

Summary

When you click Finish, the project will be created with the following settings:

Project directory:	F:/UT/UT Electrical Engineering/4th Semester/Digital Logic Design/Computer Assignments/CA5/Quartus Moore Synthesis
Project name:	Q1MoorePostSynth
Top-level design entity:	Q1MoorePostSynth
Number of files added:	1
Number of user libraries added:	0
Device assignments:	
Design template:	n/a
Family name:	Cyclone IV E
Device:	EP4CE6F17C8L
Board:	n/a
EDA tools:	
Design entry/synthesis:	<None> (<None>)
Simulation:	ModelSim-Altera (Verilog HDL)
Timing analysis:	()
Operating conditions:	
VCCINT voltage:	1.0V
Junction temperature range:	0-85 °C

Figure 5: Summary

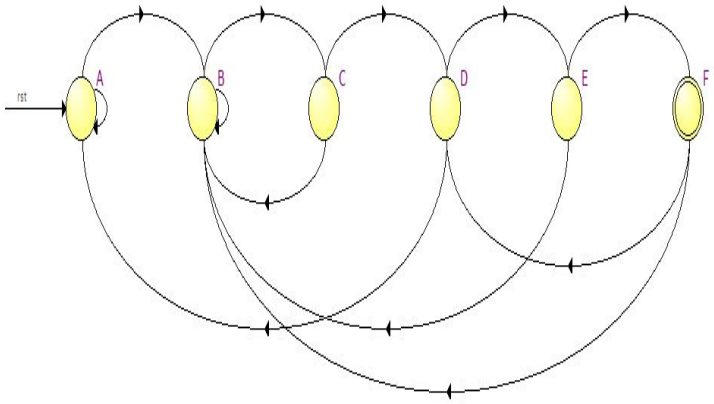


Figure 6: Quartus State Diagram

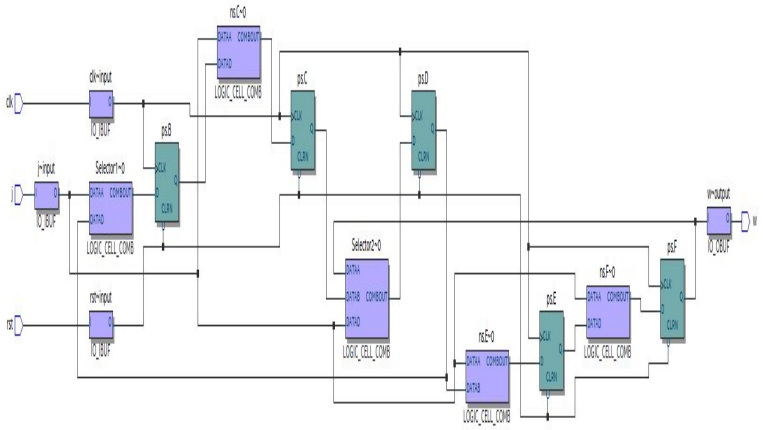


Figure 7: Post mapping technology map

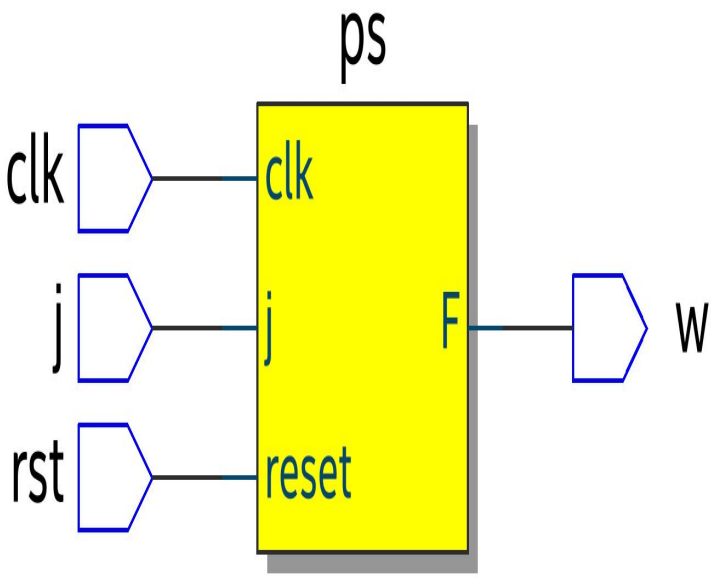


Figure 8: RTL

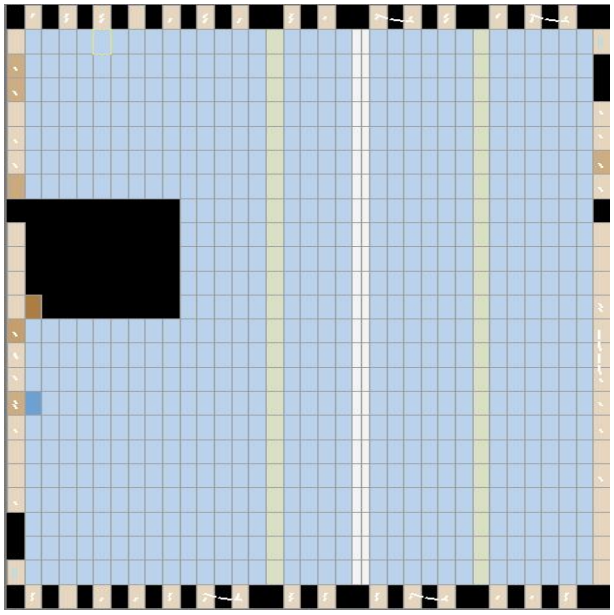


Figure 9: Chip planner

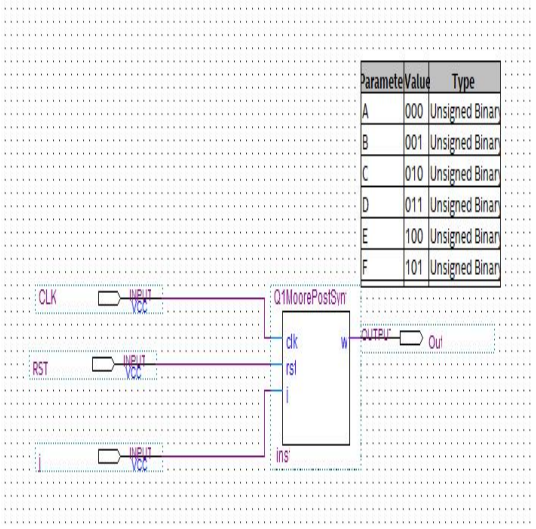


Figure 10: Symbol

We have generated the symbol as same as in the tutorial video uploaded, also all of the above results have been generated utilizing Quartus.

2.3 iii

With the help of the .vo and .sdo files generated by Quartus, we instantiate the original design and the synthesized one in modelsim and obtain the following results.

2.3.1 Verilog Testbench

```
`timescale 1ns/1ns
module Q1MooreVTB();
  reg j,clk=0,rst=0;
  wire Out1,Out2;
  Q1MoorePreSynth UUT1(clk,rst,j,Out1);
  Q1MoorePostSynth UUT2 (clk,rst,j,Out2);
  always #25 clk<=~clk;
  initial begin
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #100 $stop;
  end
endmodule
```

Figure 11: Testbench

2.3.2 Waveform

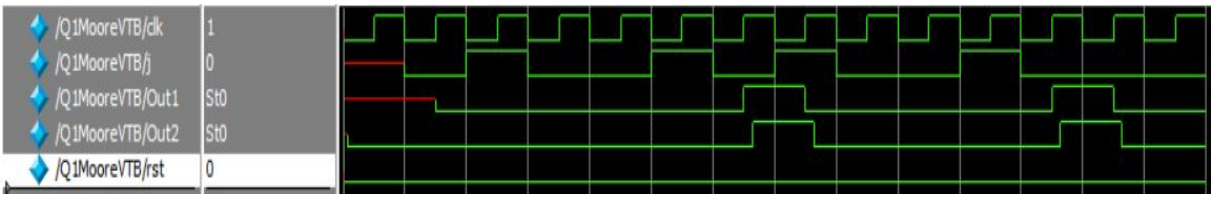


Figure 12: Waveform

2.4 B

2.4.1 i

The state diagram can be seen accordingly

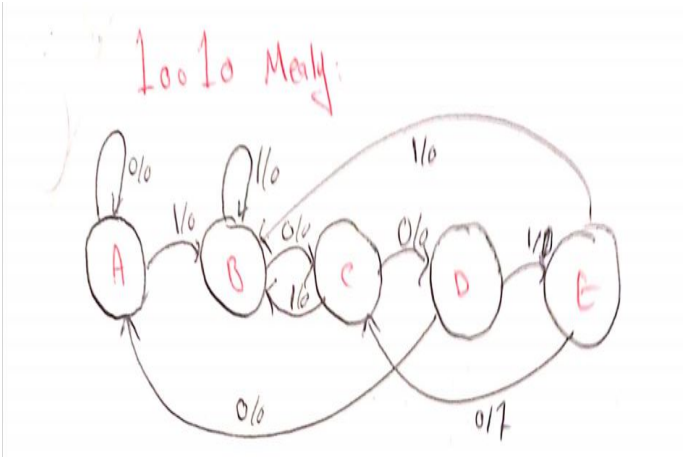


Figure 13: State Diagram

2.4.2 SystemVerilog Code & Testbench

```
`timescale 1ns/1ns
module Q2Mealy(input clk,rst,j,output w);
  logic[2:0] ns,ps;
  parameter[2:0] A=3'b000,B=3'b001,C=3'b010,D=3'b011,E=3'b100;

  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
```

```
`timescale 1ns/1ns
module Q2MealyTB();
  logic j,clk=0,rst=0;
  wire Out;
  Q2Mealy UUT(clk,rst,j,Out);
  always #25 clk<=~clk;
  initial begin
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #100 $stop;
  end
endmodule
```

Figure 14: SystemVerilog Code & Testbench

2.4.3 Waveform

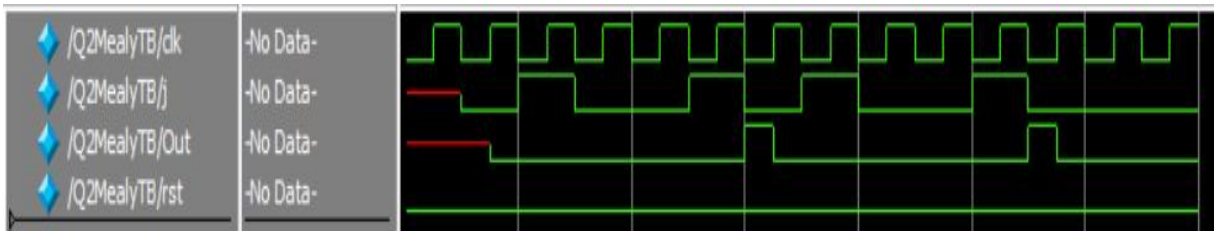
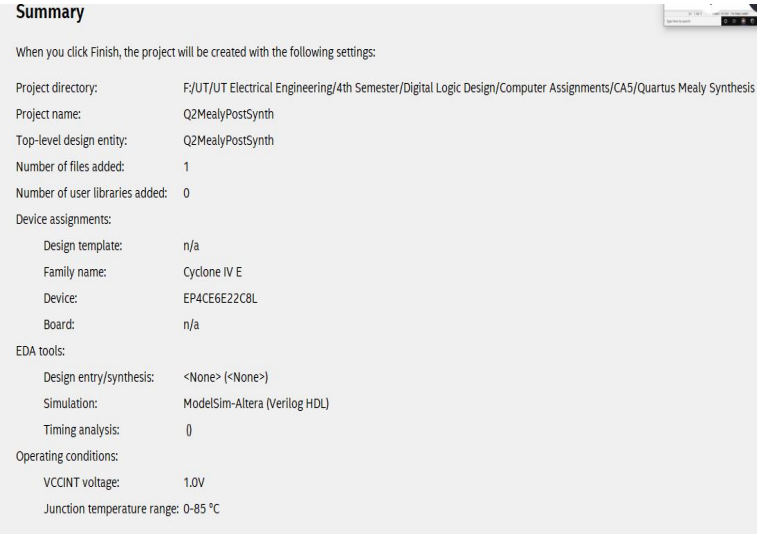


Figure 15: Mealy Machine Waveform

2.5 ii

By using Quartus we synthesize the verilog code and obtain the following results.



Summary

When you click Finish, the project will be created with the following settings:

Project directory:	F:/UT/UT Electrical Engineering/4th Semester/Digital Logic Design/Computer Assignments/CA5/Quartus Mealy Synthesis
Project name:	Q2MealyPostSynth
Top-level design entity:	Q2MealyPostSynth
Number of files added:	1
Number of user libraries added:	0
Device assignments:	
Design template:	n/a
Family name:	Cyclone IV E
Device:	EP4CE6E22C8L
Board:	n/a
EDA tools:	
Design entry/synthesis:	<None> (<None>)
Simulation:	ModelSim-Altera (Verilog HDL)
Timing analysis:	()
Operating conditions:	
VCCINT voltage:	1.0V
Junction temperature range:	0-85 °C

Figure 16: Summary

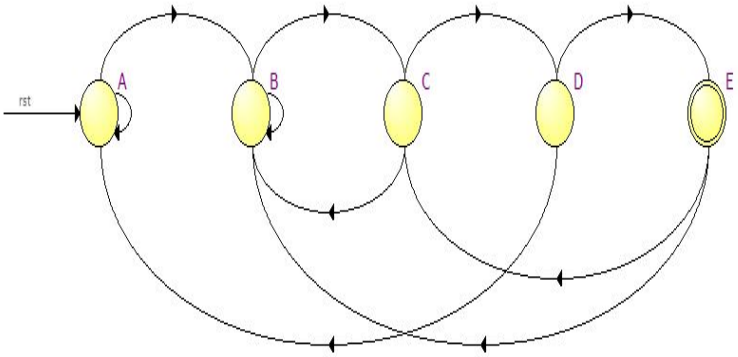


Figure 17: Quartus State Diagram

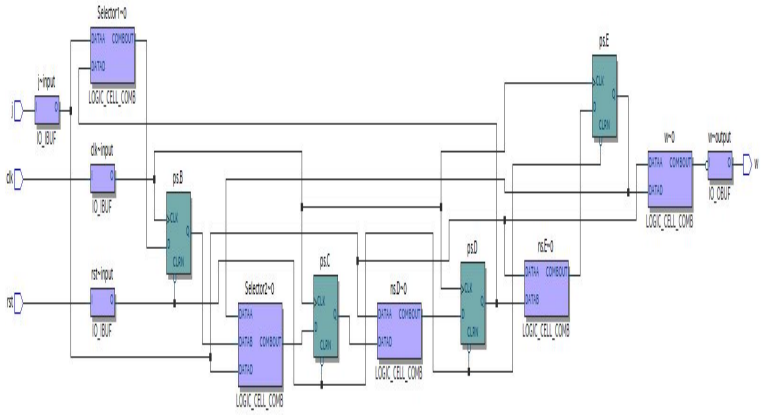


Figure 18: Post mapping technology map

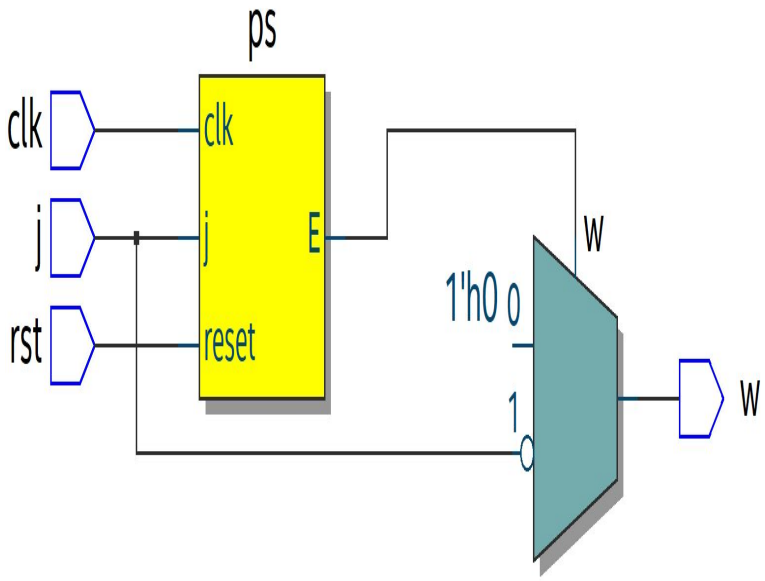


Figure 19: RTL

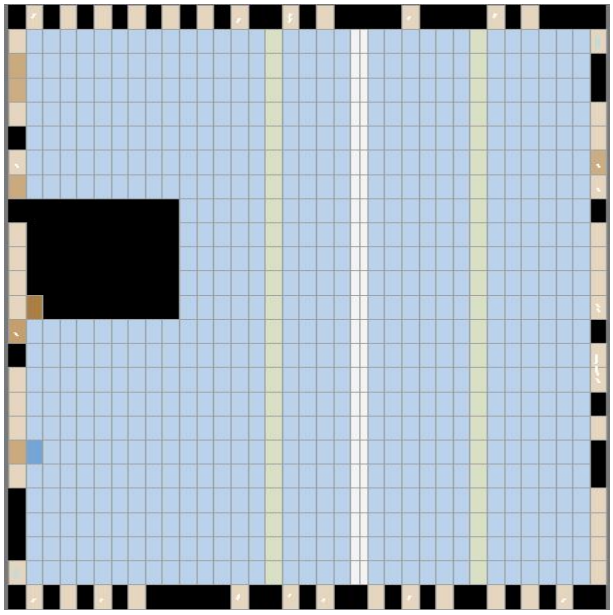


Figure 20: Chip planner view

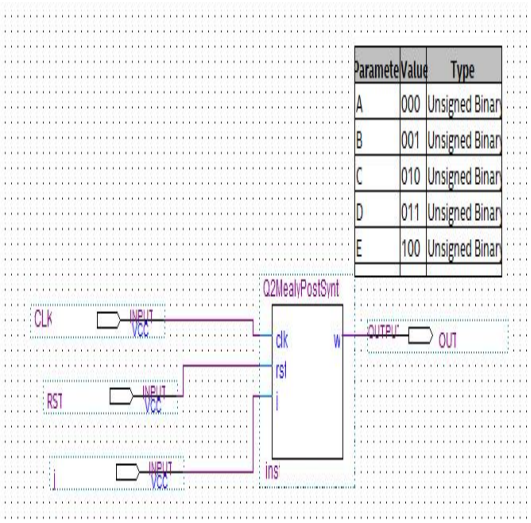


Figure 21: Symbol

We have generated the symbol as same as in the tutorial video uploaded, also all of the above results have been generated utilizing Quartus.

2.6 iii

With the help of the .vo and .sdo files generated by Quartus, we instantiate the original design and the synthesized one in modelsim and obtain the following results.

2.6.1 Verilog Testbench

```
`timescale 1ns/1ns
module Q2MealyVTB();
  reg j,clk=0,rst=0;
  wire Out1,Out2;
  Q2MealyPreSynth UUT1(clk,rst,j,Out1);
  Q2MealyPostSynth UUT2 (clk,rst,j,Out2);
  always #25 clk<=~clk;
  initial begin
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #100 $stop;
  end
endmodule
```

Figure 22: Testbench

2.6.2 Waveform

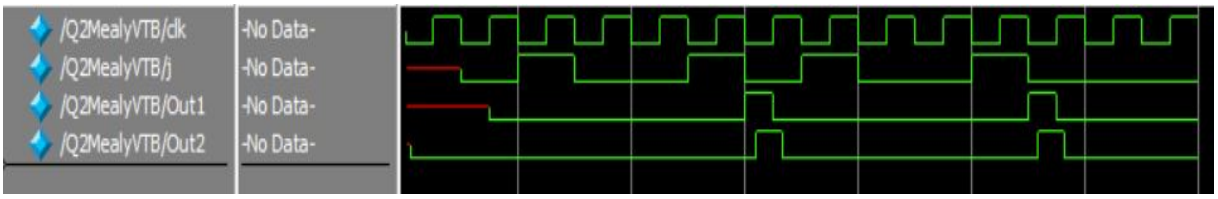


Figure 23: Waveform

2.7 C

2.7.1 i

By Xoring the outputs in the instantiations we get the following result.

2.7.2 Verilog Testbench

```
`timescale 1ns/1ns
module Q3InstanceTB();
  reg j,clk=0,rst=0;
  wire Out1,Out2,XOROut;
  Q2MealyPostSynth UUT2(clk,rst,j,Out1);
  Q1MoorePostSynth UUT1 (clk,rst,j,Out2);
  assign XOROut=Out1^Out2;
  always #25 clk<=~clk;
  initial begin
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #100 $stop;
  end
endmodule
```

Figure 24: Testbench

2.7.3 Waveform

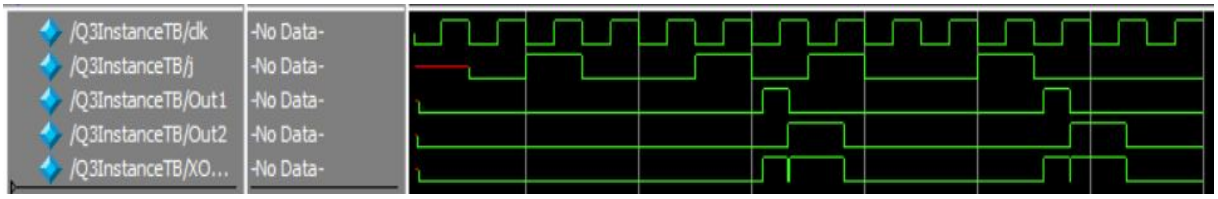


Figure 25: Waveform

2.7.4 ii

In a very small time period both the outputs are 1 which causes a glitch in the waveform which we get rid of in the next part. When the outputs are not the same the output of the XOR is 1, when both are 0 the output of the XOR is one.

2.7.5 iii

By making the following change in the testbench we can remove the glitch as demonstrated below.

2.7.6 Verilog Testbench

```
`timescale 1ns/1ns
module Q3InstanceTB2();
  reg j,clk=0,rst=0;
  wire Out1,Out2,XOROut;
  Q2MealyPostSynth UUT2(clk,rst,j,Out1);
  Q1MoorePostSynth UUT1 (clk,rst,j,Out2);
  assign XOROut=Out1^Out2;
  assign FOut=XOROut|(Out1&Out2);
  always #25 clk<=~clk;
  initial begin
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #50 j=1;
  #50 j=0;
  #50 j=0;
  #100 $stop;
  end
endmodule
```

Figure 26: Testbench

2.7.7 Waveform

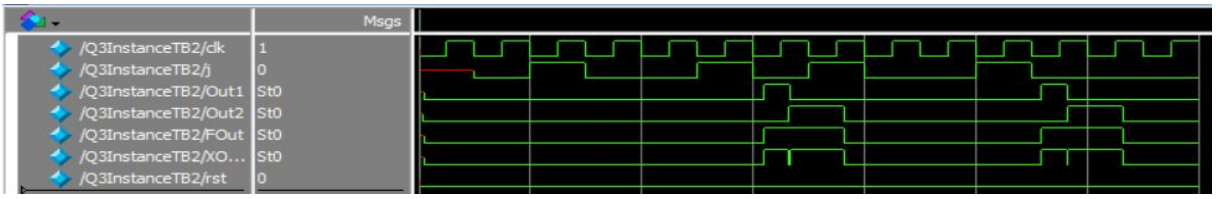


Figure 27: Waveform