به نام خدا

دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده مهندسی برق و کامپیوتر

سیستم‌های دیجیتال 1

--- ECE 894 ---

نیمسال دوم (00-99)

استاد: پروفسور نوابی

COMPUTER ASSIGNMENT 5

State Machine Coding,
Pre- and Post-Synthesis

محمدمهدی عبدالحسینی

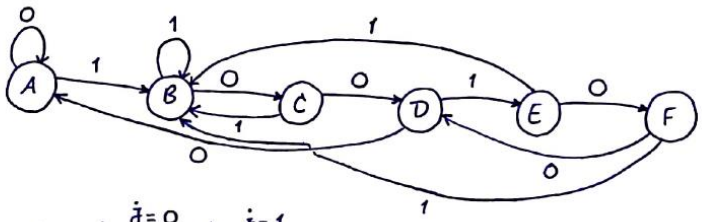
810 198 434

DIGITAL SYSTEMS

فهرست مطالب

بخش اول : moore

Moore 10010 :



		$\dot{j}=0$	$\dot{j}=1$
000	A	A: 000	B: 001
001	B	C: 010	B: 001
010	C	D: 011	B: 001
011	D	A: 000	E: 100
100	E	F: 101	B: 001
101	F	D: 011	B: 001

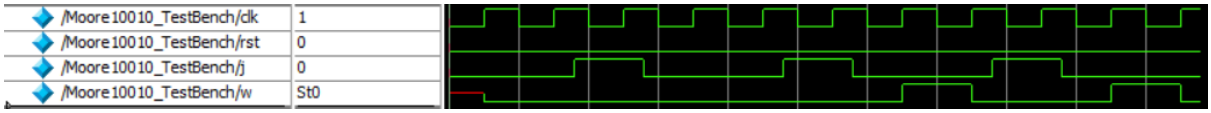
Moore :

```
module Moore10010 (input clk, rst, j, output w);

        reg [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
```

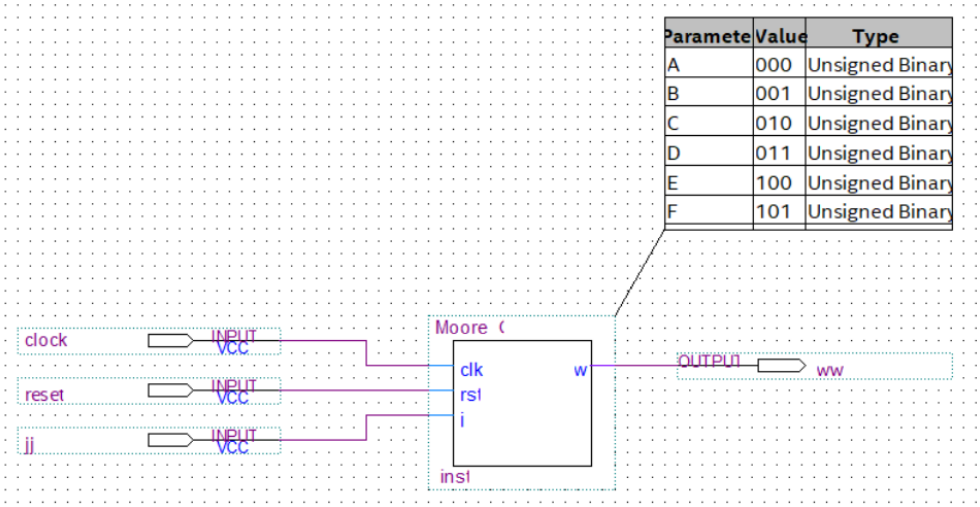
Moore TB :

```
2   module Moore10010_TestBench ();
3
4           reg clk = 0;
5           reg rst = 0;
6           reg j = 0;
7           wire w;
8
9           Moore10010 M1 (clk, rst, j, w);
10
11          always #50 clk = ~clk;
12          initial begin
13          #180 j = 1;
14          #100 j = 0;
15          #100 j = 0;
16          #100 j = 1;
17          #100 j = 0;
18          #100 j = 0;
19          #100 j = 1;
20          #100 j = 0;
21          #200 $stop;
22          end
23
24  endmodule
```

TB :



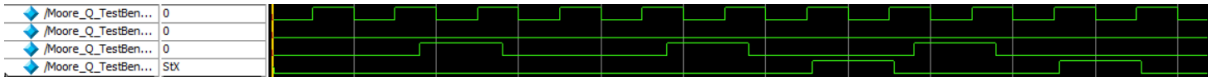
Quartus :

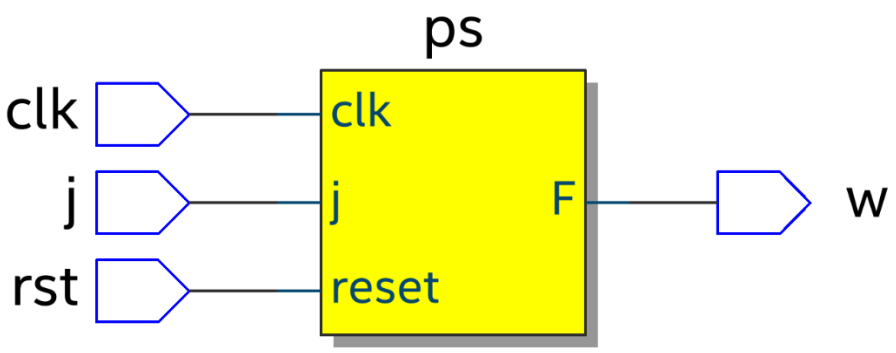


Moore Q TB :

```
2   `timescale 1ns/1ns
3
4   module Moore_Q_TestBench ();
5
6           reg clk = 0;
7           reg rst = 0;
8           reg j = 0;
9           wire w;
10
11          Moore_Q MR1 (clk, rst, j, w);
12
13          always #50 clk = ~clk;
14          initial begin
15          #180 j = 1;
16          #100 j = 0;
17          #100 j = 0;
18          #100 j = 1;
19          #100 j = 0;
20          #100 j = 0;
21          #100 j = 1;
22          #100 j = 0;
23          #200 $stop;
24          end
25
26  endmodule
```

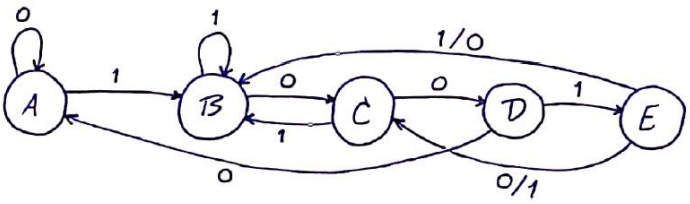
TB :





بخش دوم : mealy

Mealy 10010 :



		$j=0$	$j=1$	j = 0	j = 1
000	A	A	B	0	0
001	B	C	B	0	0
010	C	D	B	0	0
011	D	A	E	0	0
100	E	C	B	1	0
				W	

Mealy :

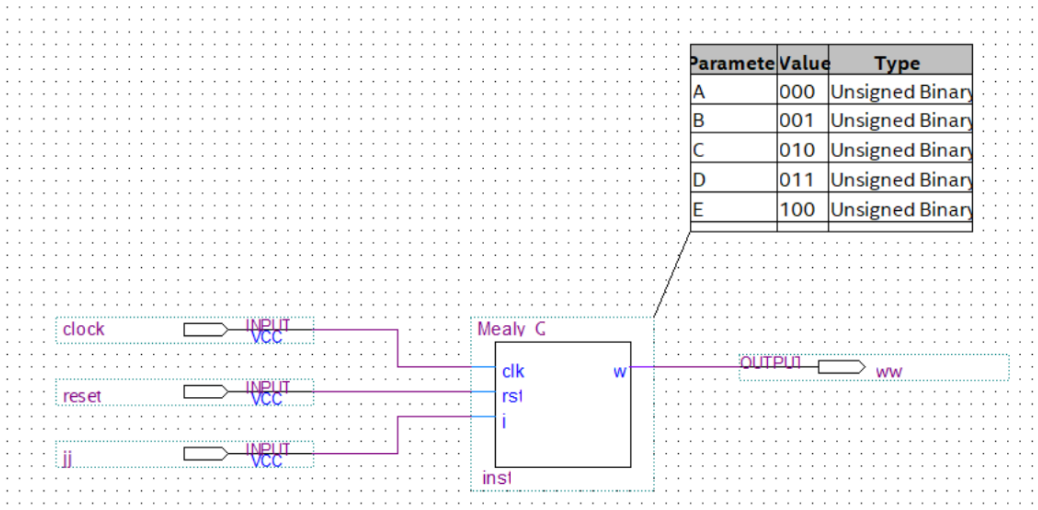
```
2   module Mealy10010 (input clk, rst, j, output w);
3
4           reg [2:0] ns, ps;
5           parameter [2:0] A=3'b000, B=3'b001, C=3'b010, D=3'b011, E=3'b100;
6
7           always @(ps,j) begin
8                   ns = A;
9                   case (ps)
10                          A: ns=j ? B : A;
11                          B: ns=j ? B : C;
12                          C: ns=j ? B : D;
13                          D: ns=j ? E : A;
14                          E: ns=j ? B : C;
15                          default : ns = A;
16                  endcase
17          end
18
19          assign w = (ps==E) ? ~j : 1'b0;
20
21          always @(posedge clk, posedge rst) begin
22                  if (rst)
23                          ps <= A;
24                  else
25                          ps <= ns;
26          end
27
28  endmodule
```

Mealy TB :

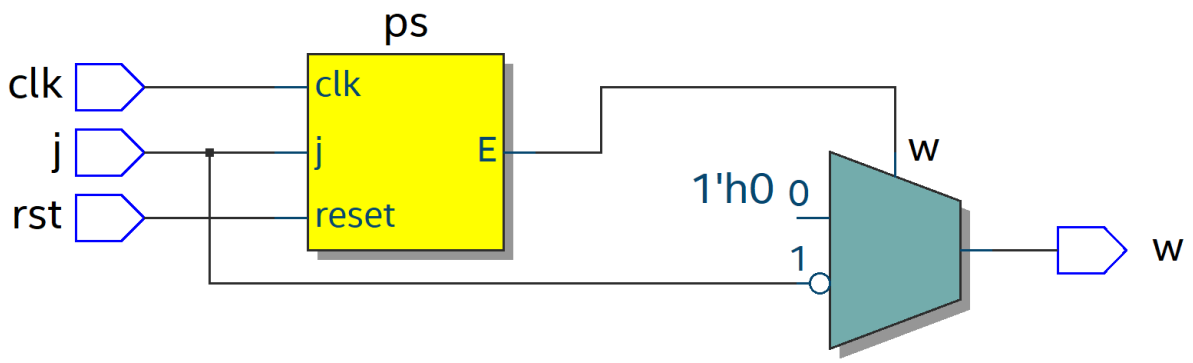
```
2   module Mealy10010_TestBench ();
3
4           reg clk = 0;
5           reg rst = 0;
6           reg j = 0;
7           wire w;
8
9           Mealy10010 M2 (clk, rst, j, w);
10
11          always #50 clk = ~clk;
12          initial begin
13          #180 j = 1;
14          #100 j = 0;
15          #100 j = 0;
16          #100 j = 1;
17          #100 j = 0;
18          #100 j = 0;
19          #100 j = 1;
20          #100 j = 0;
21          #200 $stop;
22          end
23
24  endmodule
```

TB :

Quartus :



Mealy Q TB :



بخش سوم : مقایسه

```
2   `timescale 1ns/1ns
3
4   module Moore_Mealy_TestBench ();
5
6           reg clk = 0;
7           reg rst = 0;
8           reg j = 0;
9           reg XOR = 0;
10          wire w_moore, wQ_moore, w_mealy, wQ_mealy;
11
12          Moore10010 mrVS2 (clk, rst, j, w_moore);
13          Moore_Q mrVS1 (clk, rst, j, wQ_moore);
14          Mealy10010 mlVS2 (clk, rst, j, w_mealy);
15          Mealy_Q mlVS1 (clk, rst, j, wQ_mealy);
16
17          assign XOR = wQ_moore ^ wQ_mealy;
18
19          always #50 clk = ~clk;
20          initial begin
21          #180 j = 1;
22          #100 j = 0;
23          #100 j = 0;
24          #100 j = 1;
25          #100 j = 0;
26          #100 j = 0;
27          #100 j = 1;
28          #100 j = 0;
29          #100 j = 0;
30          #100 j = 0;
31          #100 j = 1;
32          #100 j = 0;
33          #200 $stop;
34          end
35
36  endmodule
```