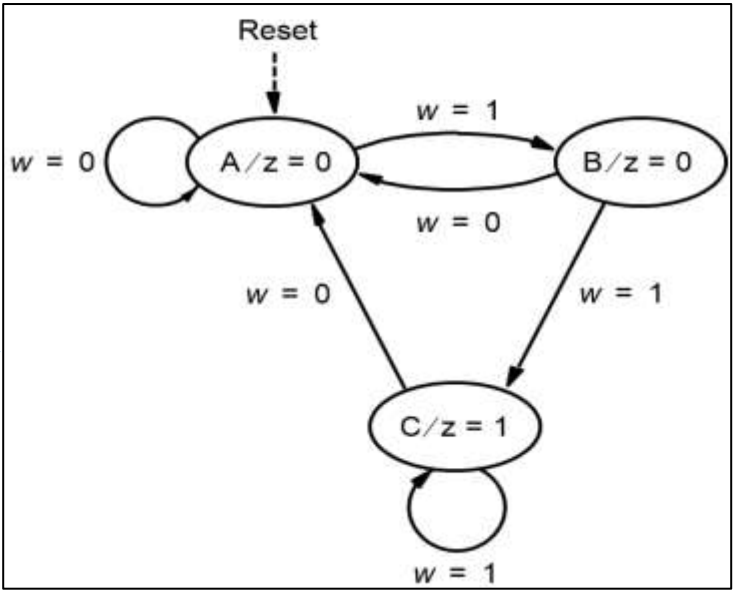
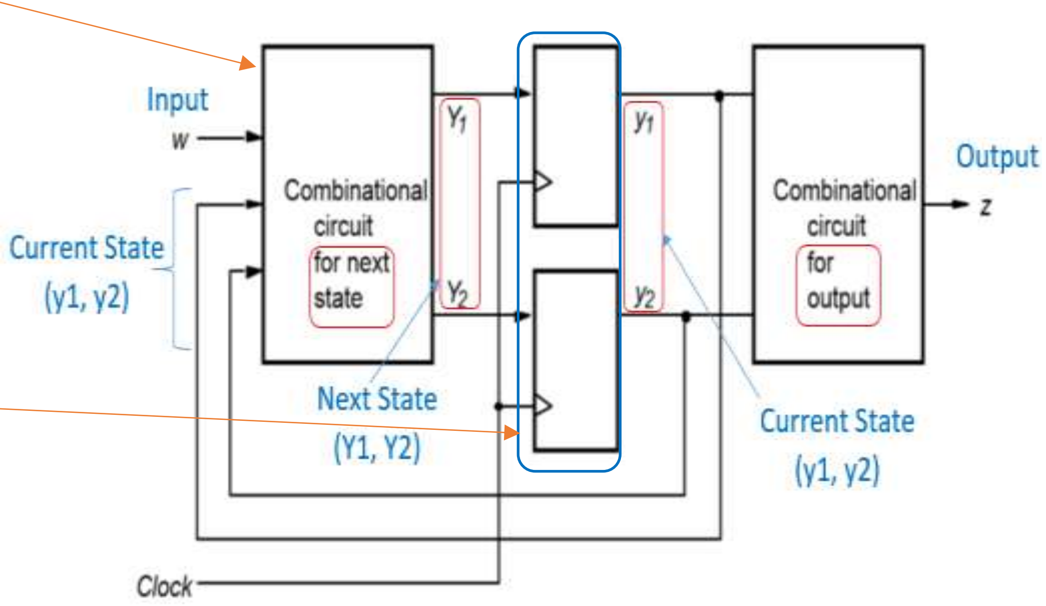
Verilog for Moore type “11” sequence detector (Notes 3-11, 12)

```
module simple (Clock, Resetn, w, z);
      input Clock, Resetn, w;
      output z;
      reg [2: l] y, Y;
      parameter [2:1] A=2'b00, B=2'b01, C=2'b10;
      // Define the next state combinational circuit
      always @(w,y)
            case (y)
            A:    if (w) Y=B;
                  else Y=A;
            B:    if (w) Y=C;
                  else Y=A;
            C:    if (w) Y=C;
                  else Y=A;
            default: Y = 2'bxx;
            endcase
      // Define the sequential block
      always @(negedge Resetn, posedge Clock)
            if (Resetn==0) y<= A;
            else y <= Y;
      // Define output
      assign z = (y == C);
endmodule
```

Current State

Next State



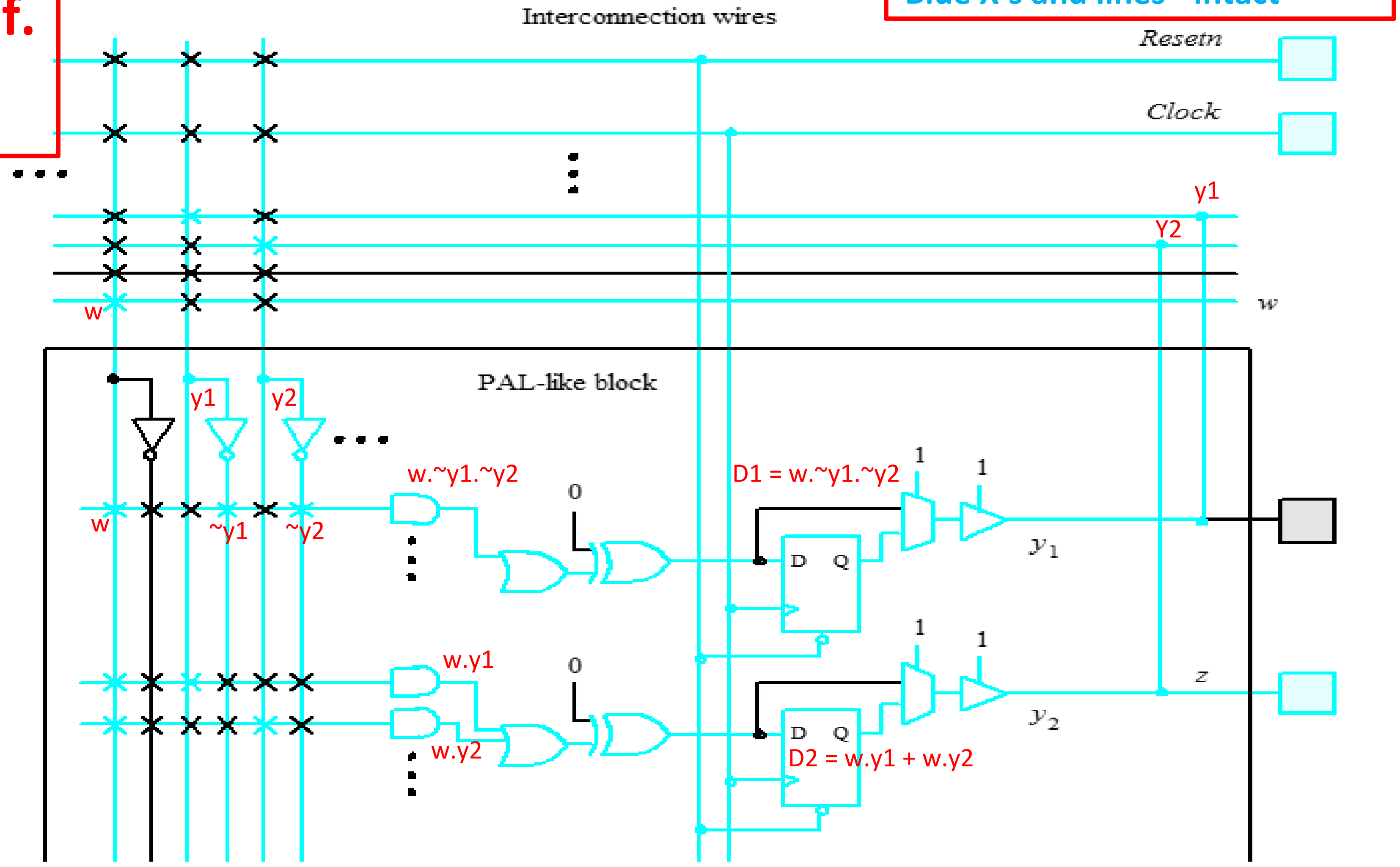


Implementation on a CPLD (Notes 3-13)

For ref. only

Black X's and lines – Removed

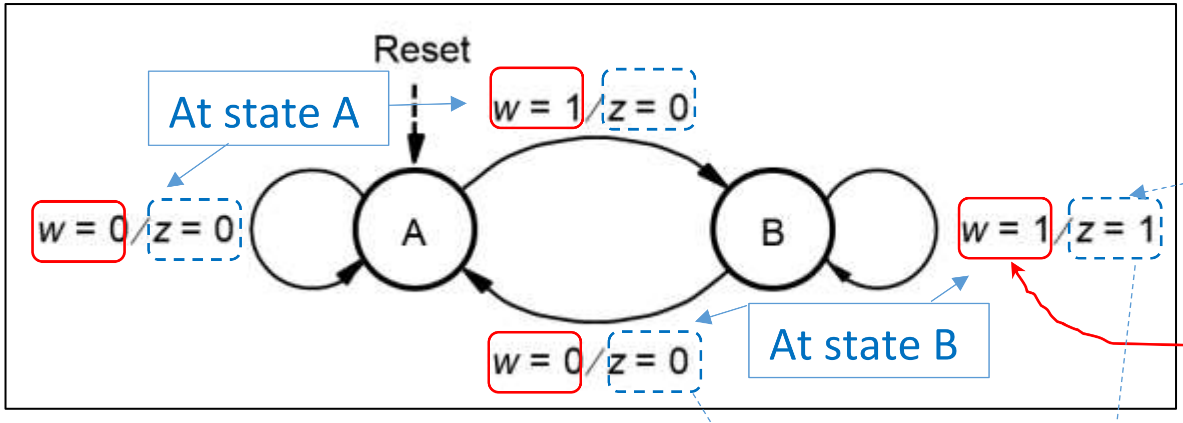
Blue X's and lines - Intact



$Z = y_2$ $\qquad Y_1 = D_1 = w\bar{y}_1\bar{y}_2$ $\qquad Y_2 = D_2 = wy_1 + wy_2$

$= w(y_1 + y_2)$

State diagram & table of Mealy's FSM (Notes 3-13, 14)



Mealy type FSM:

outputs depend on current state as well as inputs.

Each state has an output value for each input combination.

Usually Mealy FSM has fewer states than Moore FSM for the same logic function.

Mealy's State Table:

Present state	Next state		Output z	
	$w = 0$	$w = 1$	$w = 0$	$w = 1$
A	A	B	0	0
B	A	B	0	1

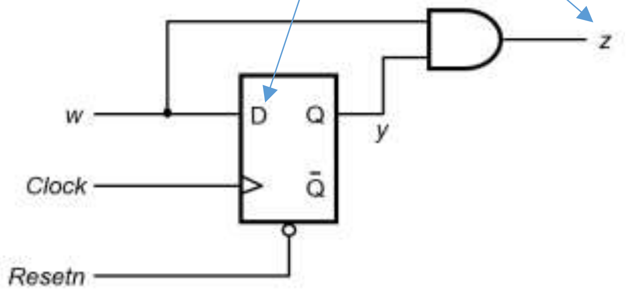
Mealy's State-assigned Table:

	Present state	Next state		Output	
		$w = 0$	$w = 1$	$w = 0$	$w = 1$
	y(Q)	Y(D)	Y(D)	z	z
A	0	A 0	B 1	0	0
B	1	A 0	B 1	0	1

When implemented with D flip-flop:

z=1 when y=1 and w=1 → **z = w.y**

D=Y=1 when w=1 → **D=Y=w**



J-K Excitation table & implementation of Mealy's FSM (Notes 3-15)

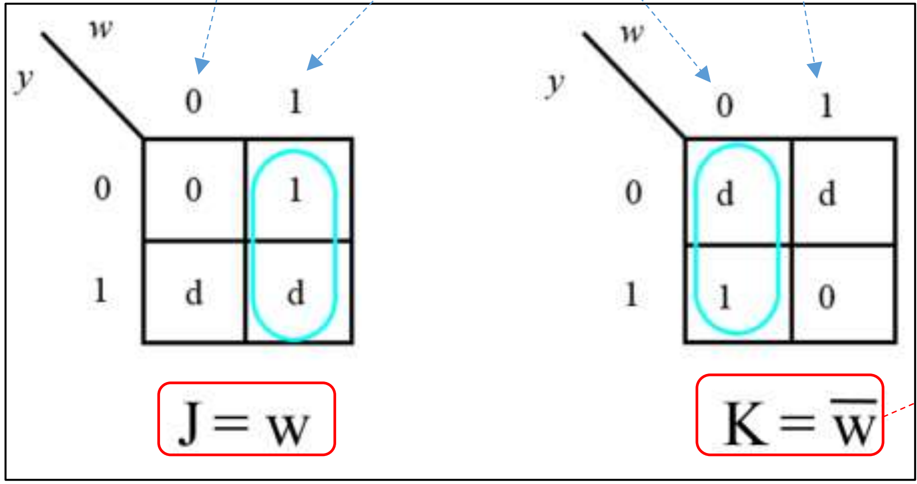
For ref. only

Present state	w=0			w=1			w=0	w=1
	Next state	flip-flop input		Next state	flip-flop input		Output	
y	Y	J	K	Y	J	K	z	
0	0	0	d	1	1	d	0	0
1	0	d	1	1	d	0	0	1

Recall from 3-7:

$Q_{(t)}$	$Q_{(t+1)}$	J	K
0	0	0	d
0	1	1	d
1	0	d	1
1	1	d	0

z = w.y



y \ w	0	1
0	0	1
1	d	d

$J = w$

y \ w	0	1
0	d	d
1	1	0

$K = \overline{w}$

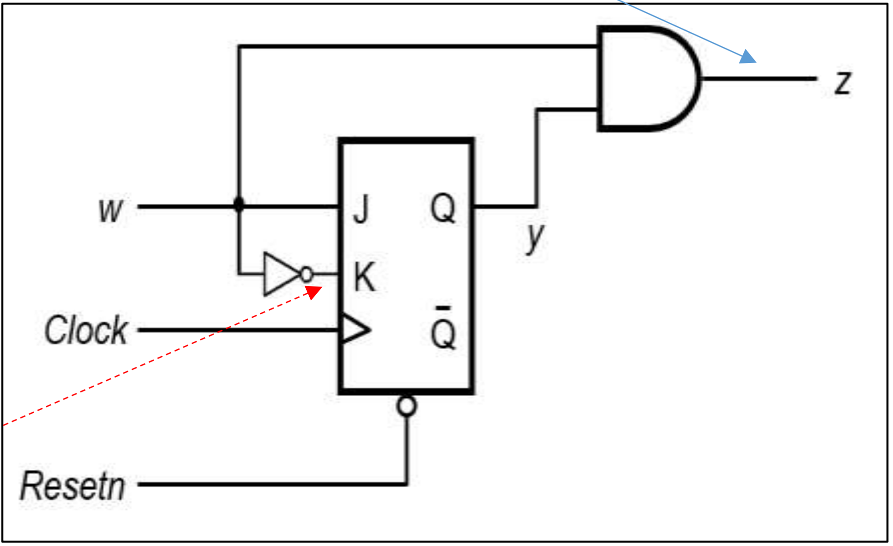
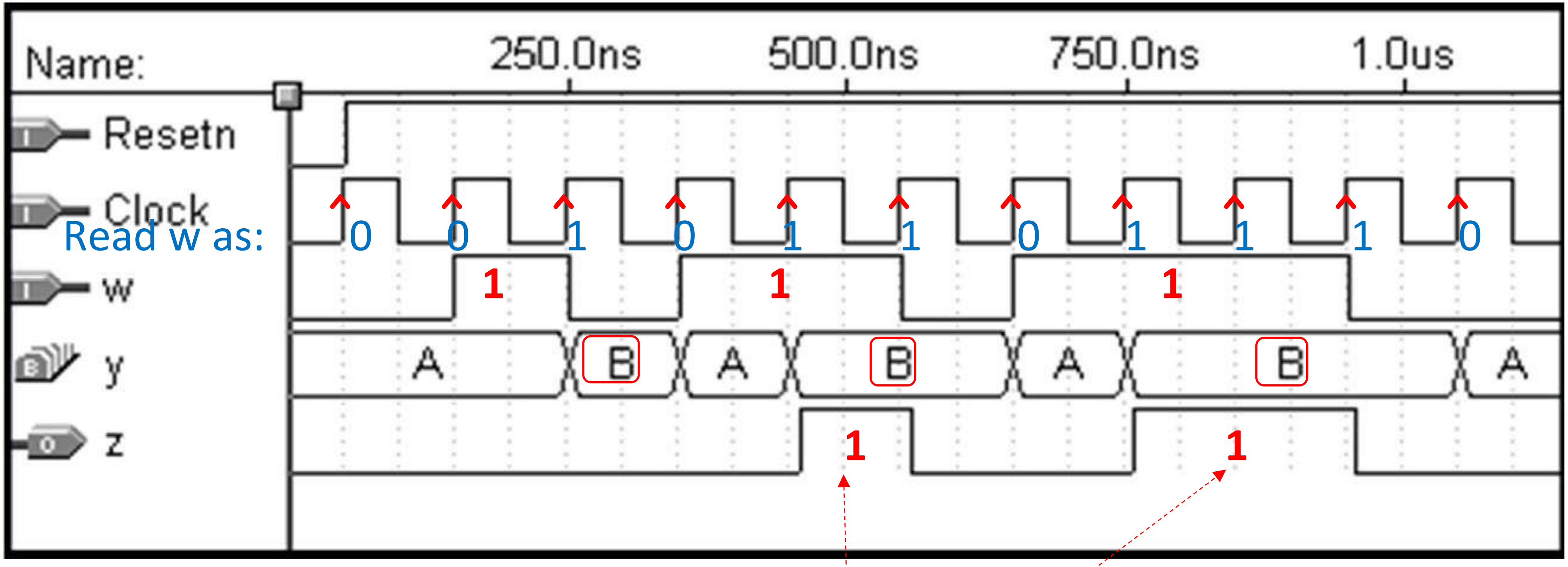
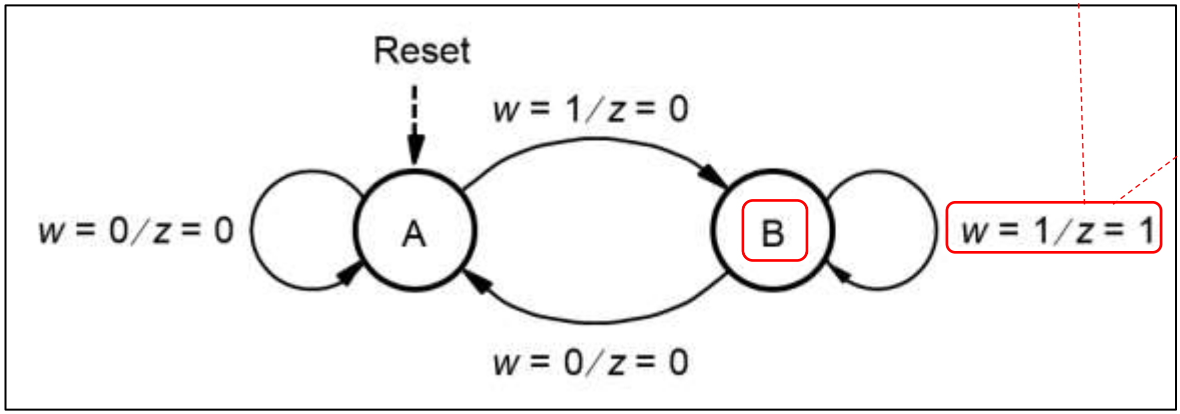


Figure 21 (Notes 3-15) - Timing diagram of "11" sequence detector for Mealy's circuit with input w changes at **active** clock edges (PGT)

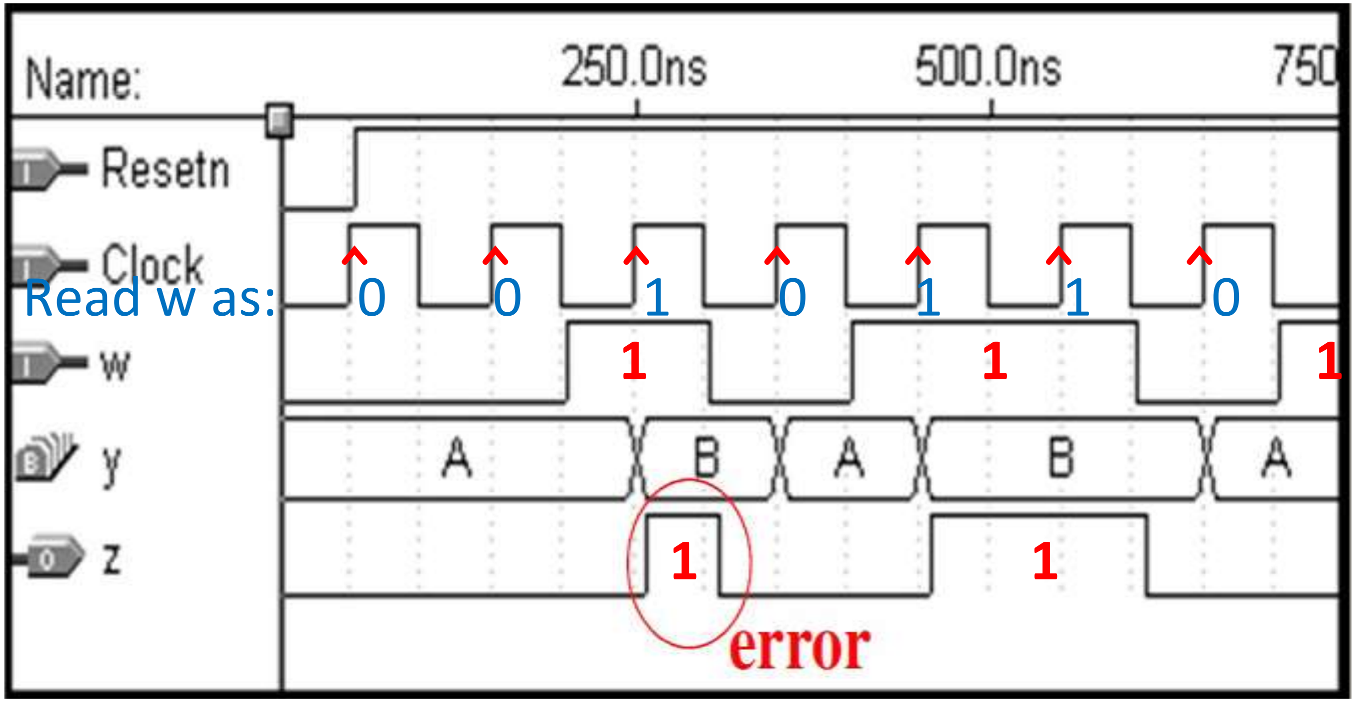
DE2: If hold-time, t_H = 0, as for all modern ICs, Q takes the value just before the active clock edge.





Output z=1 only when input w=1 during state B.

Figure 22 - Timing diagram of “11” sequence detector for Mealy’s circuit with input w **not** changing at **active** clock edges:



In order to make input w changes at active clock edges, a pre-stage flip-flop is required:

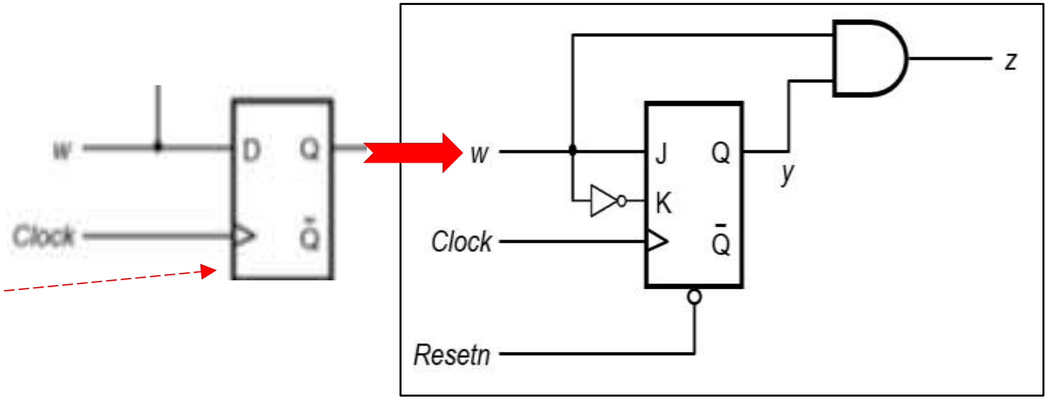
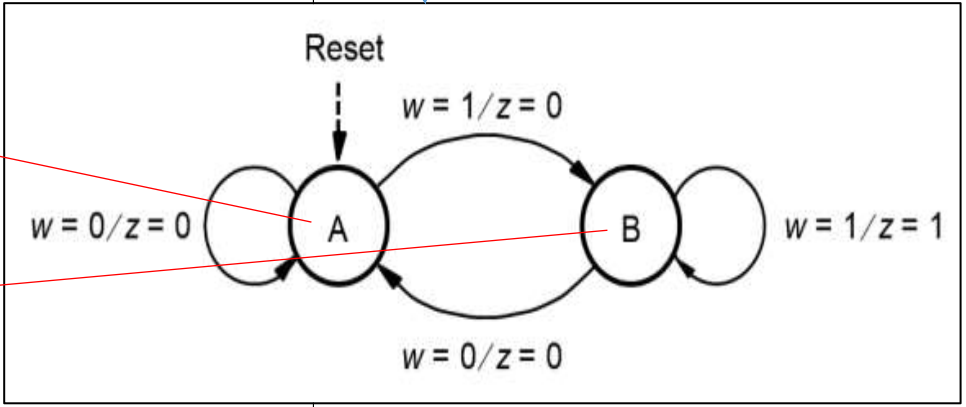


Figure 23 (Notes 3-19) - Verilog code for Figure 15

```
module mealy_Fig13 (Clock, Resetn, w, z);
    input Clock, Resetn, w; output reg z;

    reg y, Y;
    parameter A=1'b0, B=1'b1;

    // Define the next state and output combinational circuit:
    always @(w,y)
        case (y)
        A:   if (w==1) begin z=0; Y=B; end
             else      begin z=0; Y=A; end

        B:   if (w==1) begin z=1; Y=B; end
             else      begin z=0; Y=A; end
        endcase

    // Define the sequential block:
    always @(negedge Resetn, posedge Clock)
        if (Resetn==0)    y<=A;
        else              y <= Y;
endmodule
```

y: Current State, Y: Next State


Reset
w = 1/z = 0
w = 0/z = 0
A
B
w = 1/z = 1
w = 0/z = 0