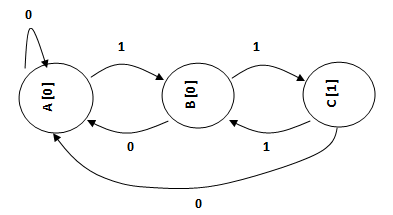
****

**Two consecutive 1s (11) sequence or pattern detector using Moore FSM with three states. Here only non-overlapping patterns are detected.**

**module fsm\_two1s\_moore2 (IN, OUT, R, CK);**

**input IN, R, CK;**

**output OUT;**

**wire OUT;**

**parameter A = 0, B = 1, C = 2;**

**reg [1:0] PS, NS;**

**always @(posedge CK, R)**

**if (R)**

**PS <= A;**

**else**

**PS <= NS;**

**always @(PS or IN)**

**case (PS)**

**A: NS = IN?B:A;**

**B: NS = IN?C:A;**

**C: NS = IN?B:A;**

**endcase**

**assign OUT = (PS==C);**

**endmodule**

**module tst\_fsm\_moore;**

**reg x, rst, clk;**

**wire z;**

**fsm\_two1s\_moore2 fsm (.IN(x), .R(rst), .CK(clk), .OUT(z));**

**initial begin**

**clk = 0;**

**x = 0;**

**#10 x = 1;**

**#20 x = 0;**

**#20 x = 1;**

**#20 x = 1;**

**#20 x = 1;**

**#20 x = 0;**

**end**

**initial begin**

**rst = 1;**

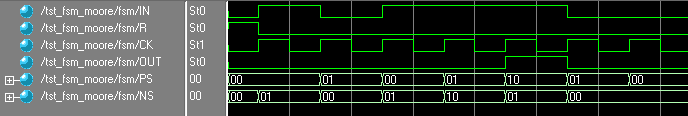
**#10 rst = 0;**

**end**

**always**

**#10 clk = ~clk;**

**endmodule**

****