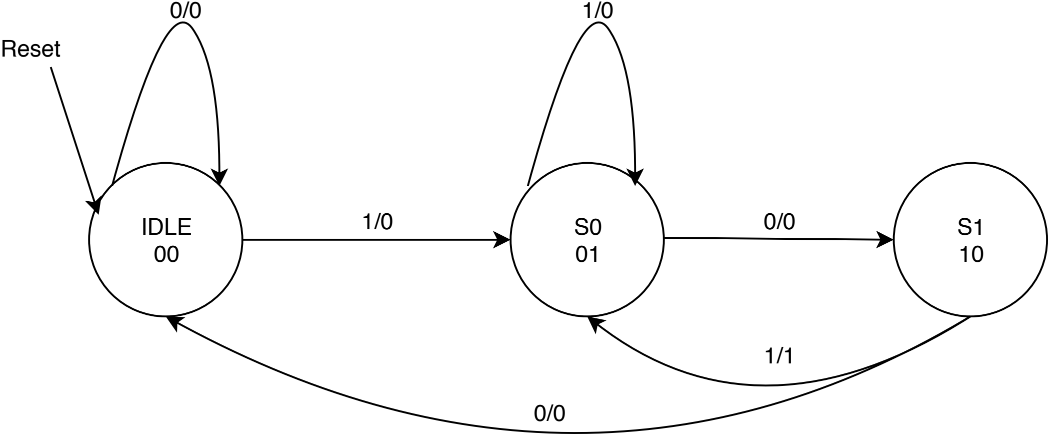
SACRAMENTO
STATE

Name: Karthik Shekarappa
Email: karthikms@csus.edu
Instructor: Dr. Behnam Arad
Course: EEE 273-Hierarchical Digital Design Methodology
Assignment Number: 2
Date Submitted: 8th Feb 2019
Due Date: 11th Feb 2019 by 11:59pm


0/0
1/0
Reset
IDLE
00
1/0
S0
01
0/0
S1
10
1/1
0/0

```
module seq_detect#(parameter[1:0] idle = 2'b00, s0 = 2'b01, s1 =
2'b10)(input x,clk,rst, output reg Z);
reg[1:0] current_state, next_state;
always @(posedge clk, negedge rst)
begin
      if(!rst)   current_state <= idle;
      else       current_state <= next_state;
end
always @(*)
begin
       next_state = idle ; Z = 1'b0;
case(current_state)
      idle: Z = idle_fn(x);

      s0:   Z = S0_fn(x);

      s1:   Z = S1_fn(x);
      default:   begin  next_state = idle; Z = 1'b0;     end

endcase
end
function idle_fn(input x);
      begin
            if(x) begin next_state = s0; idle_fn = 1'b0;   end
            else begin  next_state = idle; idle_fn = 1'b0;end
      end
endfunction
function S0_fn(input x);
      begin
            if(!x) begin     next_state = s1; S0_fn = 1'b0;    end
            else begin  next_state = s0; S0_fn = 1'b0;    end
      end
endfunction
function S1_fn(input x);
      begin
            if(x) begin next_state = s0; S1_fn = 1'b1;    end
            else begin  next_state = idle; S1_fn = 1'b0;  end
      end
endfunction
endmodule
```

```
`include "seqdetect.v"
module seq_detect_fixture;
reg x,clk,rst;
wire Z;
seq_detect u1(.x(x), .clk(clk), .rst(rst), .Z(Z));
initial
      $monitor($time,"   X = %b     Z = %b     reset = %b\n",x,Z,rst);
initial
      $vcdpluson;
initial
           begin x = 1'b0;rst = 1'b0; end
initial
begin
      reset(0);
      seq_in(0,0,0);
      reset(1);
      seq_in(0,0,0);
      seq_in(0,0,1);
      seq_in(0,1,0);
      seq_in(1,0,1);
      seq_in(1,1,0);
      reset(0);
      reset(1);
      seq_in(1,1,1);
      reset(0);
      seq_in(1,0,1);
      reset(1);
      seq_in(0,0,1);
      seq_in(0,1,0);
      reset(0);
      seq_in(1,0,1);
      reset(1);
      seq_in(1,0,1);
      seq_in(0,0,1);
      seq_in(0,1,0);
      $finish;
end
task seq_in(input in1,in2,in3);
begin
      @(posedge clk)x = in1;
      @(posedge clk)x = in2;
      @(posedge clk)x = in3;
end
endtask
task reset(input r);
begin
      if (r)  #10 rst = 1;
      else  #10 rst = 0;
end
endtask
initial  begin clk = 1'b0;
forever #10 clk=~clk;
end
endmodule
```

```
Chronologic VCS simulator copyright 1991-2014
Contains Synopsys proprietary information.
Compiler version I-2014.03-2; Runtime version I-2014.03-2;  Mar  5 23:46
2019
VCD+ Writer I-2014.03-2 Copyright (c) 1991-2014 by Synopsys Inc.
                   0   X = 0     Z = 0   reset = 0

                  60   X = 0     Z = 0   reset = 1

                 170   X = 1     Z = 0   reset = 1

                 190   X = 0     Z = 0   reset = 1

                 210   X = 1     Z = 1   reset = 1

                 230   X = 0     Z = 0   reset = 1

                 250   X = 1     Z = 1   reset = 1

                 270   X = 0     Z = 0   reset = 1

                 290   X = 1     Z = 1   reset = 1

                 310   X = 1     Z = 0   reset = 1

                 350   X = 0     Z = 0   reset = 1

                 360   X = 0     Z = 0   reset = 0

                 370   X = 0     Z = 0   reset = 1

                 390   X = 1     Z = 0   reset = 1

                 440   X = 1     Z = 0   reset = 0

                 470   X = 0     Z = 0   reset = 0

                 490   X = 1     Z = 0   reset = 0

                 500   X = 1     Z = 0   reset = 1

                 510   X = 0     Z = 0   reset = 1

                 550   X = 1     Z = 0   reset = 1

                 570   X = 0     Z = 0   reset = 1

                 590   X = 1     Z = 1   reset = 1

                 610   X = 0     Z = 0   reset = 1

                 620   X = 0     Z = 0   reset = 0

                 630   X = 1     Z = 0   reset = 0
                 650   X = 0     Z = 0   reset = 0

                 670   X = 1     Z = 0   reset = 0

                 680   X = 1     Z = 0   reset = 1

                 710   X = 0     Z = 0   reset = 1

                 730   X = 1     Z = 1   reset = 1

                 750   X = 0     Z = 0   reset = 1

                 790   X = 1     Z = 0   reset = 1

                 810   X = 0     Z = 0   reset = 1

                 830   X = 1     Z = 1   reset = 1

$finish called from file "seqdetect_fixture.v", line 36.
$finish at simulation time                  850
           V C S   S i m u l a t i o n   R e p o r t
Time: 850
CPU Time:      0.350 seconds;       Data structure size:   0.0Mb
Tue Mar  5 23:46:11 2019
```