`timescale 1ns / 1ps

//////////////////////////////////////////////////////////////////////////////////

// Company: Ratner Surf Designs

// Engineer: James Ratner

//

// Create Date: 07/07/2018 08:05:03 AM

// Design Name:

// Module Name: fsm\_template

// Project Name:

// Target Devices:

// Tool Versions:

// Description: Generic FSM model with both Mealy & Moore outputs.

// Note: data widths of state variables are not specified

//

// Dependencies:

//

// Revision:

// Revision 1.00 - File Created (07-07-2018)

// Additional Comments:

//

//////////////////////////////////////////////////////////////////////////////////

module fsm\_template(reset\_n, x\_in, clk, mealy, moore);

input reset\_n, x\_in, clk;

output reg mealy, moore;

//- next state & present state variables

reg [1:0] NS, PS;

//- bit-level state representations

parameter [1:0] st\_A=2'b00, st\_B=2'b01, st\_C=2'b11;

//- model the state registers

always @ (negedge reset\_n, posedge clk)

if (reset\_n == 0)

PS <= st\_A;

else

PS <= NS;

//- model the next-state and output decoders

always @ (x\_in,PS)

begin

mealy = 0; moore = 0; // assign all outputs

case(PS)

st\_A:

begin

moore = 1;

if (x\_in == 1)

begin

mealy = 0;

NS = st\_A;

end

else

begin

mealy = 1;

NS = st\_B;

end

end

st\_B:

begin

moore = 0;

mealy = 1;

NS = st\_C;

end

st\_C:

begin

moore = 1;

if (x\_in == 1)

begin

mealy = 1;

NS = st\_B;

end

else

begin

mealy = 0;

NS = st\_A;

end

end

default: NS = st\_A;

endcase

end

endmodule