```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 01/24/2018 04:29:10 PM
// Design Name:
// Module Name: MultSim
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module MultSim();

reg [7:0] sw;
reg clkin, btnR;
wire [6:0] seg;
wire dp;
wire [3:0] an;

Top UUT (sw, clkin, btnR, seg, dp, an);

parameter PERIOD = 10;
    parameter real DUTY_CYCLE = 0.5;
    parameter OFFSET = 2;

    initial    // Clock process for clkin
    begin
        #OFFSET
                  clkin = 1'b1;
        forever
        begin
            #(PERIOD-(PERIOD*DUTY_CYCLE)) clkin = ~clkin;
        end
        end
```

```
initial
begin
      // add your stimuli here
      // to set signal foo to value 0 use
      // foo = 1'b0;
      // to set signal foo to value 1 use
      // foo = 1'b1;
      //always advance time my multiples of 100ns
      // to advance time by 100ns use the following line
      sw = 8'b00000000;
      #1000;
      sw = 8'b00010001;
      #100;
      sw = 8'b00010010;
      #100;
      sw = 8'b00010011;
      #100;
      sw = 8'b00010100;
      #100;
      sw = 8'b00010101;
      #100;
      sw = 8'b00010111;
      #100;
      sw = 8'b00011000;
      #100;
      sw = 8'b00011001;
      #100;
      sw = 8'b00011010;
      #100;
      sw = 8'b00011011;
      #100;
      sw = 8'b00011100;
      #100;
      sw = 8'b00011101;
      #100;
      sw = 8'b00011110;
      #100;
      sw = 8'b00011111;
      #100;
      sw = 8'b00010001;
       #100;
       sw = 8'b11110010;
       #100;
       sw = 8'b11110011;
       #100;
       sw = 8'b11110100;
```

```
        #100;
        sw = 8'b11110101;
        #100;
        sw = 8'b11110111;
        #100;
        sw = 8'b11111000;
        #100;
        sw = 8'b11111001;
        #100;
        sw = 8'b11111010;
        #100;
        sw = 8'b11111011;
        #100;
        sw = 8'b11111100;
        #100;
        sw = 8'b11111101;
        #100;
        sw = 8'b11111110;
        #100;
        sw = 8'b11111111;
       #100;
      end
endmodule
```