endmodule

## Digital Design using FPGAs EE 417

## Example: for loop and disable Verilog command

Write a Verilog module that detects whether an input word has its majority of bits equal to 1s or not. The module should be parametrized, and should use the 'disable' command.

```
module majority_bits_are_1s #(
                   parameter word_size = 8,
                   parameter majority = 5
                               ) (
                   input
                             [word_size -1 : 0]
                                                  data_in,
                   output
                                                  majority_ones_out
                              );
integer
                                index;
parameter
                                count_size = 3,
                                                  // Should accommodate a count of 101 for majority
            [count_size - 1: 0]
                                count;
reg
          majority ones out = (count>=majority);
assign
always @ (data_in)
                     begin: count_ones
     count = 0;
     for ( index = 0;  index < word_size; index = index+1)</pre>
         begin
             count = count + data_in[index];
             if (count >= majority)
               disable count_ones;
         end
     end
```

```
module majority_bits_are_1s
                                    #(
                                                          word_size = 8,
                                          parameter
                                                          majority = 5,
                                          parameter
                                                          count_size = 3
                                         parameter
                                    )(
                                                                                        data_in,
                                          input
                                                          [word_size -1: 0]
                                                                                        majority_ones_out,
                                         output
                                         output reg
                                                          [count_size -1: 0]
                                                                                        count
                                      );
//parameter
                                         count_size = 3;
//reg
             [count_size-1:0]
                                         count:
integer
                                       index;
assign
                majority_ones_out = (count>=majority);
always @ (data_in)
                          begin: count_ones
    count = 0;
    for (index = 0;
                          index < word_size; index = index+1)</pre>
          begin
                count = count + data_in[index];
                if (count >= majority)
                      disable count_ones;
        end
end
endmodule
module majority_1s_tb ();
                                 word_size = 8;
 parameter
 parameter
                                  count_size = 3;
reg
                                 data_in;
             [word_size -1:0]
wire
                                 majority_ones_out;
             [count_size-1:0]
wire
                                 count;
majority_bits_are_1s
                                     (data_in, majority_ones_out,count);
                            UUT
 initial
 begin
 #0
          data_in = 8'b0000_0000;
          data_in = 8'b1111_0001;
 #10
                                        // Expected high out
          data_in = 8'b0000_11111;
 #10
          data_in = 8'b1111_1111;
data_in = 8'b1010_1010;
 #10
                                        // Expected high out
 #10
          data_in = 8'b0111_1111;
 #10
                                        // Expected high out
          data_in = 8'b1100_0001;
data_in = 8'b1111_0011;
 #10
 #10
                                        // Expected high out
          data_in = 8'b0000_0011;
 #10
          data_in = 8'b1010_1111;
 #10
                                        // Expected high out
 end
 // Using the disable command, the count should not exceed 5.
// majority_ones_out should only be high when the majority of bits are ones.
 endmodule
```

Wave - Default ::::::::::::::::::::::::::::::::::::											+ 3
<b>\$</b> 1 →	Msgs										
+- /majority_1s_tb/count		(000	101	100	101	100	101	011	101	010	
		3									
+> /majority_1s_tb/dat		0	241	15	255	170	127	193	243	3	
🥠 /majority_1s_tb/ma	St1										П
🥠 /majority_1s_tb/wo	8	8									口