

```

// File: minor_tb.v
// John Hubbard, 19 Oct 2014
// hw5a assignment for Verilog 0764 class: Problem 4.4.1: Minority Circuit
//
// Steps to run in ModelSim (on Linux this time):
// cd $HOME/git_wa/classes/ucsc/verilog_0764/hw5/modelsim
// vlib work
// vlog -sv ../major.v ../major_tb.v
// vsim work.major_tb
// run 400 ns
//

`timescale 1ns/1ns
module minor_tb;
    reg [4:0] atb;
    wire ftb, reftb;
    integer count;

    minor DUT(atb, ftb);
    minor_reference DUT_REFERENCE(atb, reftb);

    // This block prints whenever the inputs or outputs change, for either
    // the DUT or the DUT_REFERENCE instantiations:
    initial
    begin
        $monitor("Time %t, a: %b%b%b%b%b, f: %b, reftb: %b",
            $time, atb[4], atb[3], atb[2], atb[1], atb[0], ftb, reftb);
    end

    // This always block monitors for errors, by continuously comparing the
    // output of the device under test (DUT), against a reference device that
    // was built with a (slightly) simpler dataflow technique:
    always @(*)
    begin
        if (ftb ^ reftb)
            $display("FAILURE:                      a: %b%b%b%b%b, f: %b, reftb: %b",
                atb[4], atb[3], atb[2], atb[1], atb[0], ftb, reftb);
    end

    // This is the test pattern, which is exhaustive:
    initial
    begin
        for (count = 0; count < 32; count++)
            begin
                {atb} = count;
                #10;
            end
    end
endmodule

/* Sample test run:
vlog -sv ../*.v
# Model Technology ModelSim ALTERA vlog 10.1e Compiler 2013.06 Jun 12 2013
# -- Compiling module minor

```

```

# -- Compiling module major
# -- Compiling module minor_reference
# -- Compiling module minor_tb
#
# Top level modules:
#   minor_tb
vsim work.minor_tb
# vsim work.minor_tb
# Loading sv_std.std
# Loading work.minor_tb
# Loading work.minor
# Loading work.major
# Loading work.minor_reference
run 400 ns
# Time          0, a: 00000, f: 1, reftb: 1
# Time         10, a: 00001, f: 1, reftb: 1
# Time         20, a: 00010, f: 1, reftb: 1
# Time         30, a: 00011, f: 1, reftb: 1
# Time         40, a: 00100, f: 1, reftb: 1
# Time         50, a: 00101, f: 1, reftb: 1
# Time         60, a: 00110, f: 1, reftb: 1
# Time         70, a: 00111, f: 0, reftb: 0
# Time         80, a: 01000, f: 1, reftb: 1
# Time         90, a: 01001, f: 1, reftb: 1
# Time        100, a: 01010, f: 1, reftb: 1
# Time        110, a: 01011, f: 0, reftb: 0
# Time        120, a: 01100, f: 1, reftb: 1
# Time        130, a: 01101, f: 0, reftb: 0
# Time        140, a: 01110, f: 0, reftb: 0
# Time        150, a: 01111, f: 0, reftb: 0
# Time        160, a: 10000, f: 1, reftb: 1
# Time        170, a: 10001, f: 1, reftb: 1
# Time        180, a: 10010, f: 1, reftb: 1
# Time        190, a: 10011, f: 0, reftb: 0
# Time        200, a: 10100, f: 1, reftb: 1
# Time        210, a: 10101, f: 0, reftb: 0
# Time        220, a: 10110, f: 0, reftb: 0
# Time        230, a: 10111, f: 0, reftb: 0
# Time        240, a: 11000, f: 1, reftb: 1
# Time        250, a: 11001, f: 0, reftb: 0
# Time        260, a: 11010, f: 0, reftb: 0
# Time        270, a: 11011, f: 0, reftb: 0
# Time        280, a: 11100, f: 0, reftb: 0
# Time        290, a: 11101, f: 0, reftb: 0
# Time        300, a: 11110, f: 0, reftb: 0
# Time        310, a: 11111, f: 0, reftb: 0

```

*/