

2009 Paper 3 Question 4

ECAD

Below is a Verilog module that is supposed to produce a debounced version of a physical button input (**keyin**) on the output (**keyout**). Any bouncing behaviour of the input will have settled down within 100ms. Input **rst** is the global reset signal and **clk** is the global clock running at 50MHz. The module is syntactically correct, but is functionally slightly erroneous.

```
module debounce(input keyin, input clk, input rst, output reg keyout);
    reg prevkeyin;
    reg [7:0] ctr;

    always @(posedge clk or posedge rst)
        if(rst) begin
            prevkeyin <= 0;
            ctr <= 0;
            keyout <= 0;
        end else begin
            prevkeyin <= keyin;
            if(keyin!=prevkeyin) ctr <= -1; // set ctr to maximum value
            if(ctr>0) ctr <= ctr-1;
            if(ctr==0) keyout <= keyin;
        end
endmodule
```

- (a) What is the difference between *synchronising* and *debouncing* an input? [3 marks]
- (b) What is the high-level circuit diagram corresponding to the debounce module? You may assume that multi data-bit input blocks like adders, multiplexers and comparators can be used, that blocks of D-flip-flops (with clear, preset and enable inputs) are provided as well as Boolean gates. [5 marks]
- (c) What are the functional errors in the debounce module and how can they be corrected? [8 marks]
- (d) How does functional testing differ from production testing? [2 marks]
- (e) How would scannable flip-flops speed up production test of the debounce module? [2 marks]