ECAD

A Verilog programmer has written the code below in an attempt to produce a periodic square wave with a programmable high:low ratio. This module forms part of an interface between a microprocessor software algorithm and a variable speed electric motor.

The correct behaviour should be (1) when go is low the output is zero, (2) when go is high the output oscillates, (3) tcycle and ton are read at the start of each oscillation cycle.

module pwmcontroller(out, tcycle, ton, go, clk);
  output out; // controlled output, 1=on, 0=off
  input [15:0] tcycle; // cycle (on+off) time in clock cycles
  input [15:0] ton; // on time in clock cycles
  input go; // enable the output
  input clk; // system clock
  parameter sReload=0, sOn=1, sOff=2;
  reg [1:0] state;
  reg oncount, offcount;
  always @(posedge clk) begin
    if (!go) state <= sReload;
    case (state)
    sReload: begin
      oncount <= ton;
      offcount <= tcycle-ton-1;
      state <= sOn;
    end
    sOn: if (oncount==0) state <= sOff; else oncount <= oncount - 1;
    sOff: if (offcount==0) state <= sOn; else offcount <= offcount - 1;
    endcase
    end
  assign out <= state==sOn;
endmodule

(a) The above code contains errors. Write a corrected version marking any changes. [6 marks]

(b) Draw a labelled state diagram of the corrected circuit’s operation. [3 marks]

(c) Explain what clock gating is and why it is used. How might it be applied to the above circuit? [6 marks]

(d) What functionality could easily be moved to the software algorithm to reduce the size of the circuit? How would this change the module interface? [2 marks]

(e) Sketch an example standard cell ASIC design flow and state the importance of production testing. [3 marks]