

```
module count (input clk, input en, input up, input reset, output [15:0] out_data);
    reg [15:0] out;
    initial out = 16'b0;
    assign out_data = out;
    always @(posedge clk) begin
        if (reset) begin
            out <= 16'b0;
        end else begin
            if (en) begin
                if (up) begin
                    out <= out + 16'b1;
                end else begin
                    out <= out - 16'b1;
                end
            end
        end
    end
endmodule
```