```
/*
   Caitlin DeShazo-Couchot and Ashley Guillard
   EE 271 Au23: Professor Hussein
   November 9, 2023
   Lab 4: D Flip Flops to Handle Key[0] and Key[3] User Input
*/

// D Flip Flops to ensure Metastability during Tug-Of-War
// Inputs: CLOCK, Reset, and the button being pressed
// Outputs: Out value of the flip flop
module metaStability (

   // INPUT LOGIC
   input logic CLOCK, Reset, KEY,

   // OUTPUT LOGIC
   output logic out
);
   logic tug_out;

   //  Next State and Output Logic
    always_ff @(posedge CLOCK or posedge Reset) begin
        if (Reset) begin
            tug_out <= 1'b0;
            out     <= 1'b0;
        end
        else begin
            tug_out <= KEY;
            out     <= tug_out;
        end
    end

endmodule

module metaStability_testbench ();

   logic CLOCK, Reset, KEY, out;
   logic tug_out;

   //Instantiate the metaStability module
   metaStability dut (.CLOCK, .Reset, .KEY, .out);

       //CLOCK setup
   parameter CLOCK_period = 10;

   initial begin
      CLOCK <= 0;
      forever #(CLOCK_period) CLOCK <= ~CLOCK;

   end //initial

   // Test cases for metaStability. Since this works as a basic ff,
   // the modelSim should work with the standard inputs for a ff truth table.
   initial begin
      KEY = 0;    Reset = 0;  @(posedge CLOCK);
      KEY = 0;    Reset = 0;  @(posedge CLOCK); //Letting it cycle on purpose twice.
      KEY = 0;    Reset = 1;  @(posedge CLOCK);
      KEY = 0;    Reset = 1;  @(posedge CLOCK);
      KEY = 1;    Reset = 0;  @(posedge CLOCK);
      KEY = 1;    Reset = 0;  @(posedge CLOCK);
      KEY = 1;    Reset = 1;  @(posedge CLOCK);
      KEY = 1;    Reset = 1;  @(posedge CLOCK);

      $stop;
   end //initial
endmodule
```