Lab 8: Design a Door Code Detector

EECE 2060-002

Pre-lab Report

David Luria and Ojas Anand and Maulik Patel

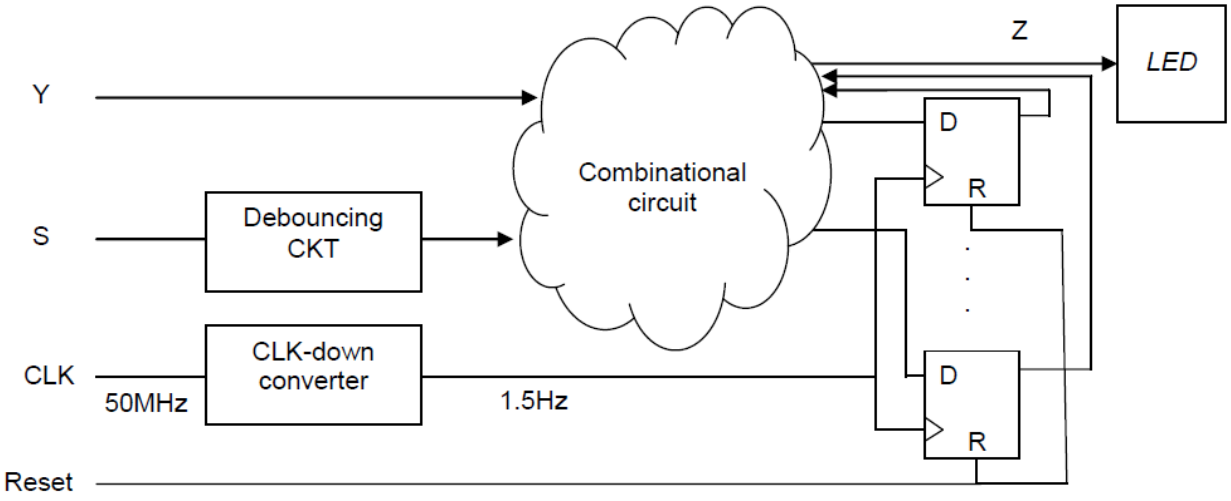
Introduction:

The purpose of this laboratory is to fully understand the design process of a sequential circuit, from specification to the final circuit design, using Verilog and FPGA.

Design Specification:

- Design a door code Detector has two inputs S and Y where S is the Start signal and Y gives the code that is sequences logic 0's and 1's.
- To open the door (i.e., Z=1), you must press the S button once and then input Y with sequence 100*10 where 0* means zero or more logic 0.
- Design a door code Detector has one Output Z which shows that whether the door is unlocked, by a LED.

Block Diagram:



Verilog code for the Code_detector:

```
module code_detector(clk, Y, S, reset, out);

        input clk, Y, reset, S;
```

```
output out;
reg [1:0] CurState;
reg [1:0] out;
initial begin
	CurState = 0;
	out = 0;
end

always @ (CurState) begin
	case(CurState)
		0:
			out = 0;
		1:
			out = 0;
		2:
			out = 0;
		3:
			out = 0;
		4:
			out = 0;
		5:
			out = 1;
	endcase
end

always @ (posedge clk) begin
	if(reset) begin
		CurState = 0;
	end else if(S) begin
```

```
case (CurState)
        0:
                if ( Y == 1 ) begin
                        CurState = 1;
                end else begin
                        CurState = 0;
                end
        1:
                if ( Y == 0 ) begin
                        CurState = 2;
                end else begin
                        CurState = 0;
                end
        2:
                if ( Y == 0 ) begin
                        CurState = 3;
                end else begin
                        CurState = 0;
                end
        3:
                if ( Y == 0)
                CurState = 3;
                end else begin
                        CurState = 4;
        4:
                if (Y == 0)
                        CurState =0;
                end else begin
                        CurState = 5;
```

```
                endcase
            end
        end

endmodule
```

Verilog Code of the Top module:

```
input S, I, clock, reset, clear;
        output z, clkOut;
        wire SlowClock, FastClock, dbS, dbI;
        reg z, clkOut;

        Clock_down_converter inst_Clock(Clock, clear, SlowClock, FastClock);
        Debounce inst_Debounce_S(FastClock, clear, S, dbS);
        Debounce inst_Debounce_I(FastClock, clear, I, dbI);
        Detector_101 inst_Detector( SlowClock, dbI, dbS, reset, z);
        always @ (posedge SlowClock or negedge SlowClock) begin
            clkOut = SlowClock;
        end
```

Testbench for the Code_detector Module:

```
module Test_bench;
        reg Y;
        reg S;
        reg reset;
        // Outputs
        wire Z;

        // Instantiate the Unit Under Test (UUT)
```

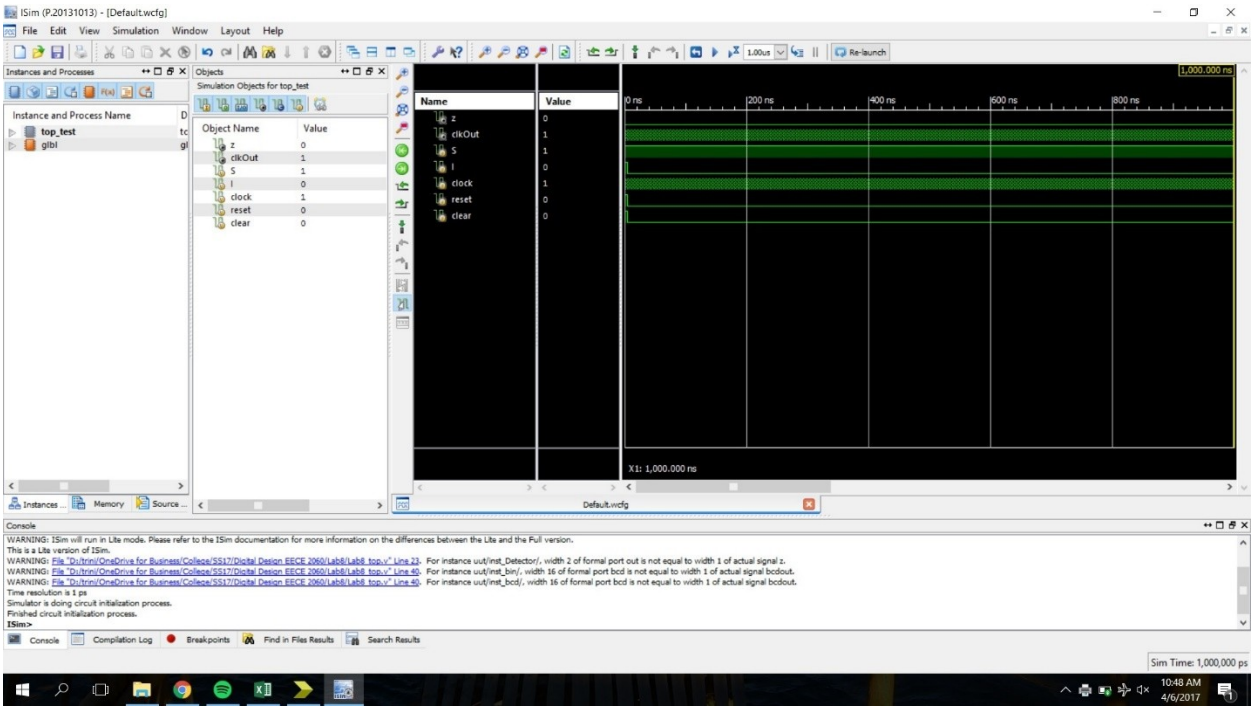
```
	Code_Detector uut (
		.Y(Y),
		.S(S),
		.reset(reset)
	);

	initial begin
		// Initialize Inputs

		// Wait 100 ns for global reset to finish
		#50
Y = 1;
		  S = 1;
		  reset = 0;

		// Add stimulus here
		#50
Y = 0;
		  S = 1;
		  reset = 0;

		 #50
Y = 0;
		  S = 1;
		  reset = 0;

		#50
Y = 0;
		  S = 1;
```

```
                reset = 0;

                #50
Y = 1;
                S = 1;
                reset = 0;


                #50
Y = 0;
                S = 1;
                reset = 0;

                #50
Y = 1;
                S = 1;
                reset = 0;

                #50
Y = 1;
                S = 1;
                reset = 0;

                #50
Y = 0;
                S = 1;
                reset = 0;

                #50
```

```
Y = 0;
                S = 1;
                reset = 0;

               #50
Y = 0;
                S = 1;
                reset = 0;

               #50
Y = 1;
                S = 1;
                reset = 0;

               #50
Y = 0;
                S = 1;
                reset = 0;

               #50
Y = 0;
                S = 1;
                                reset = 0;
               #50
                Y = 1;
                S = 1;
                reset = 0;

    end
```

endmodule

Waveform by Testbench:



Design Time:

2.5 hours