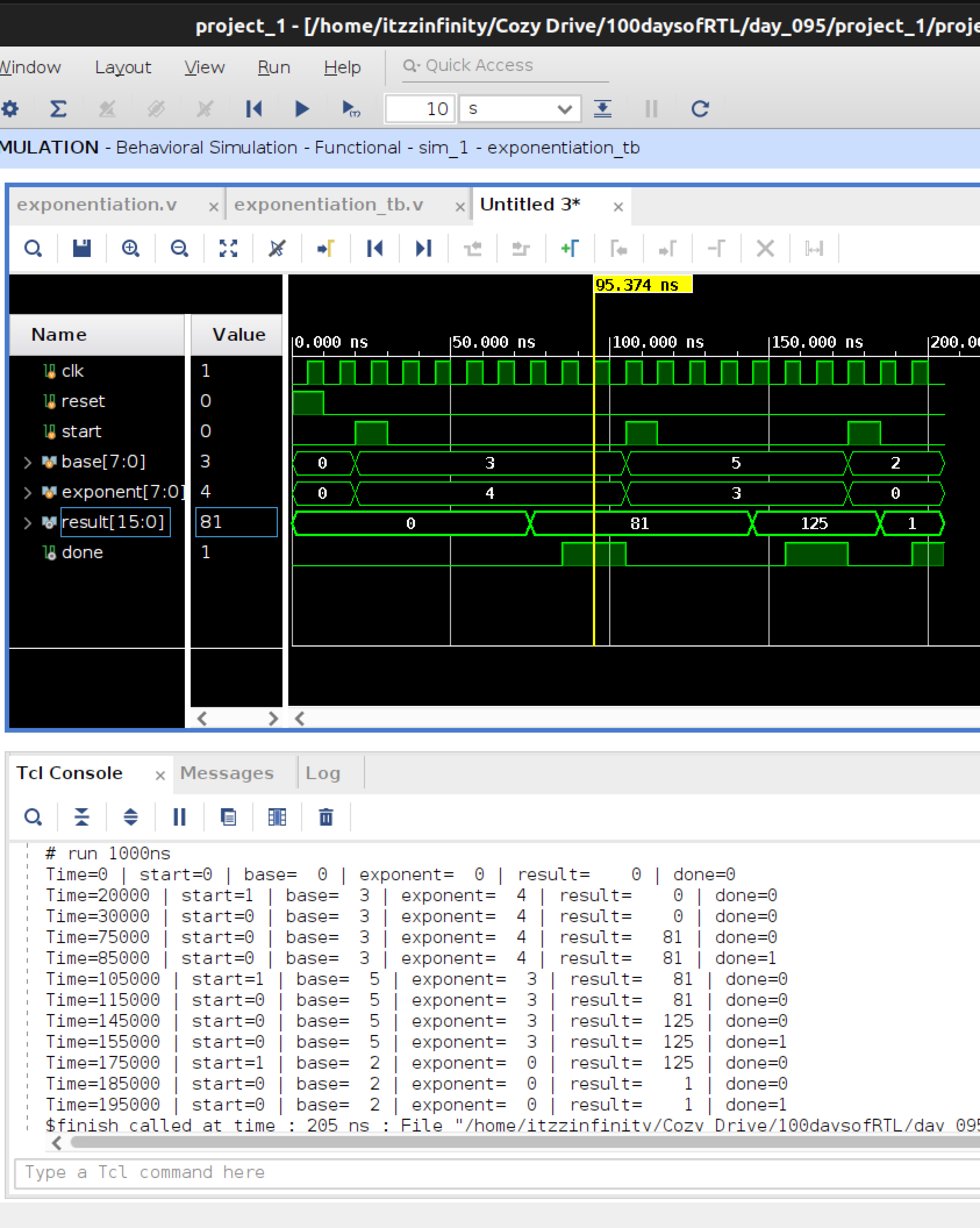
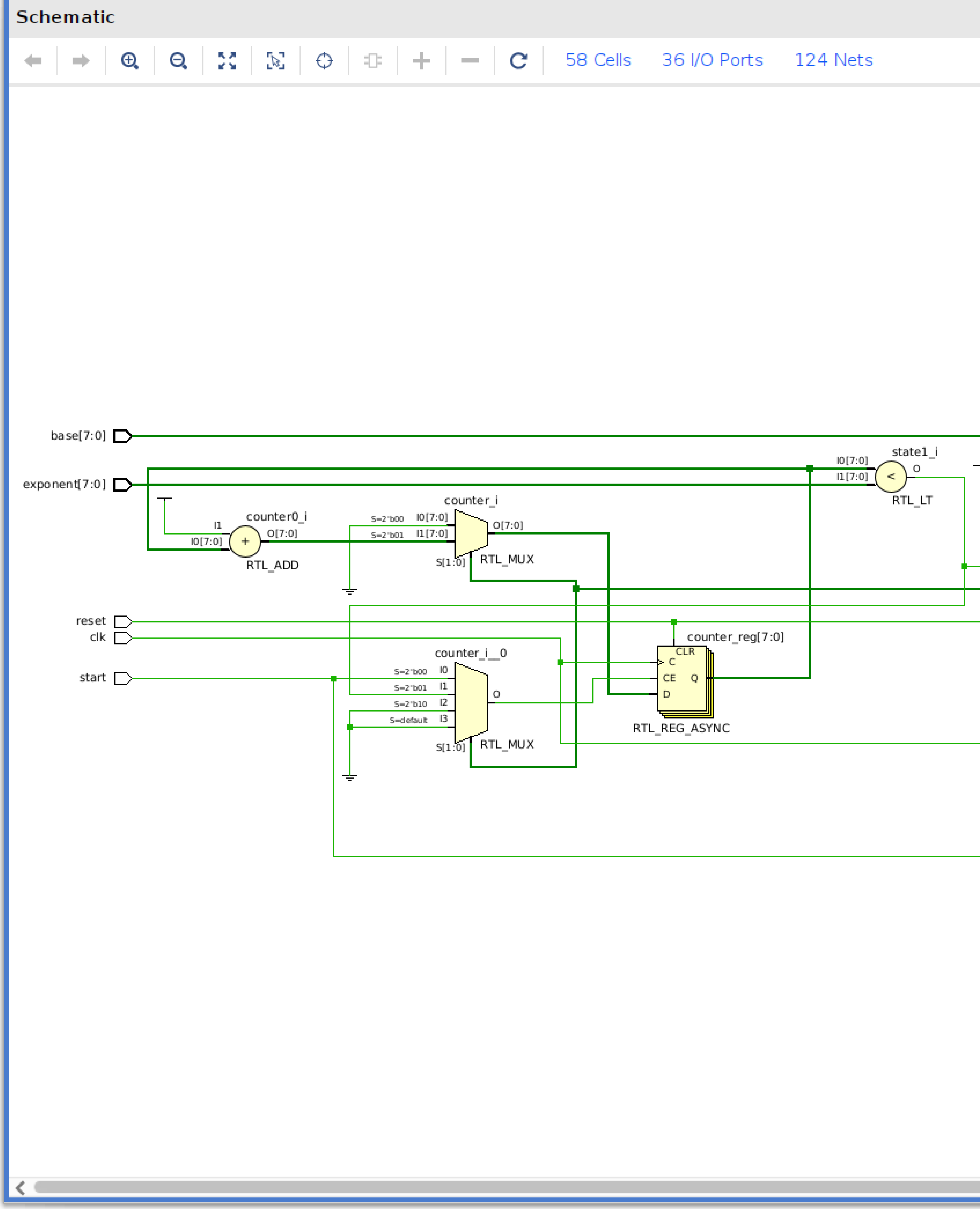
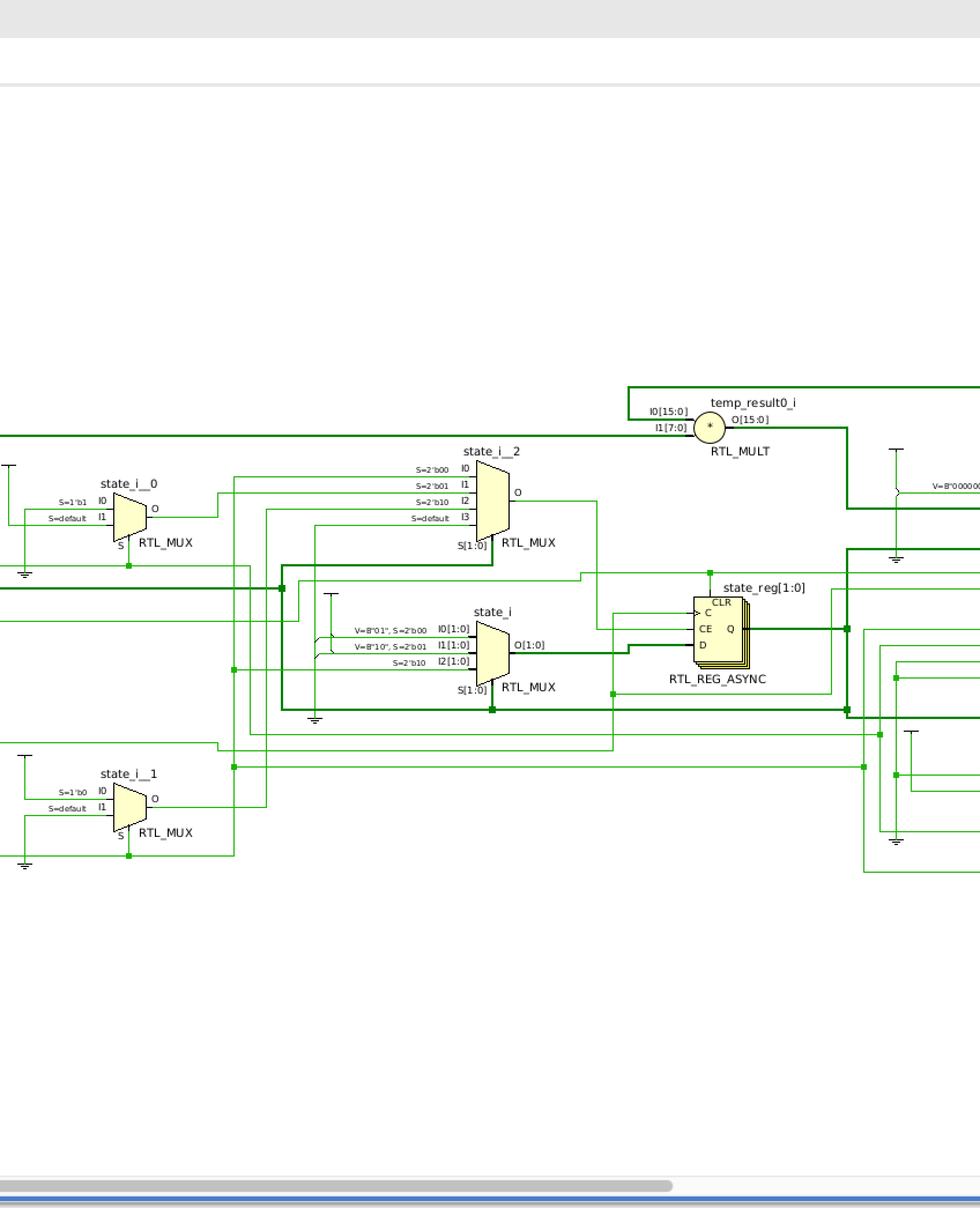
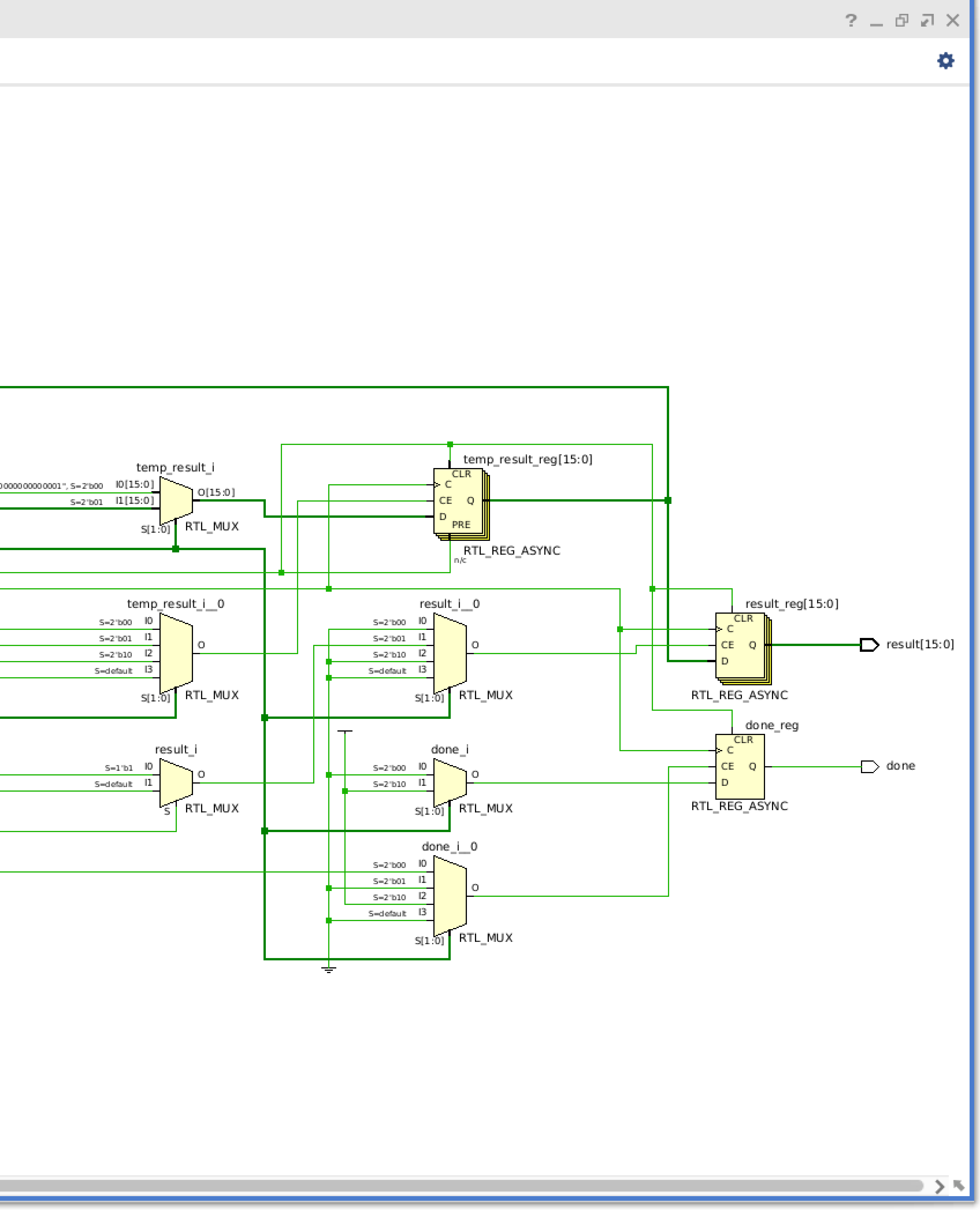
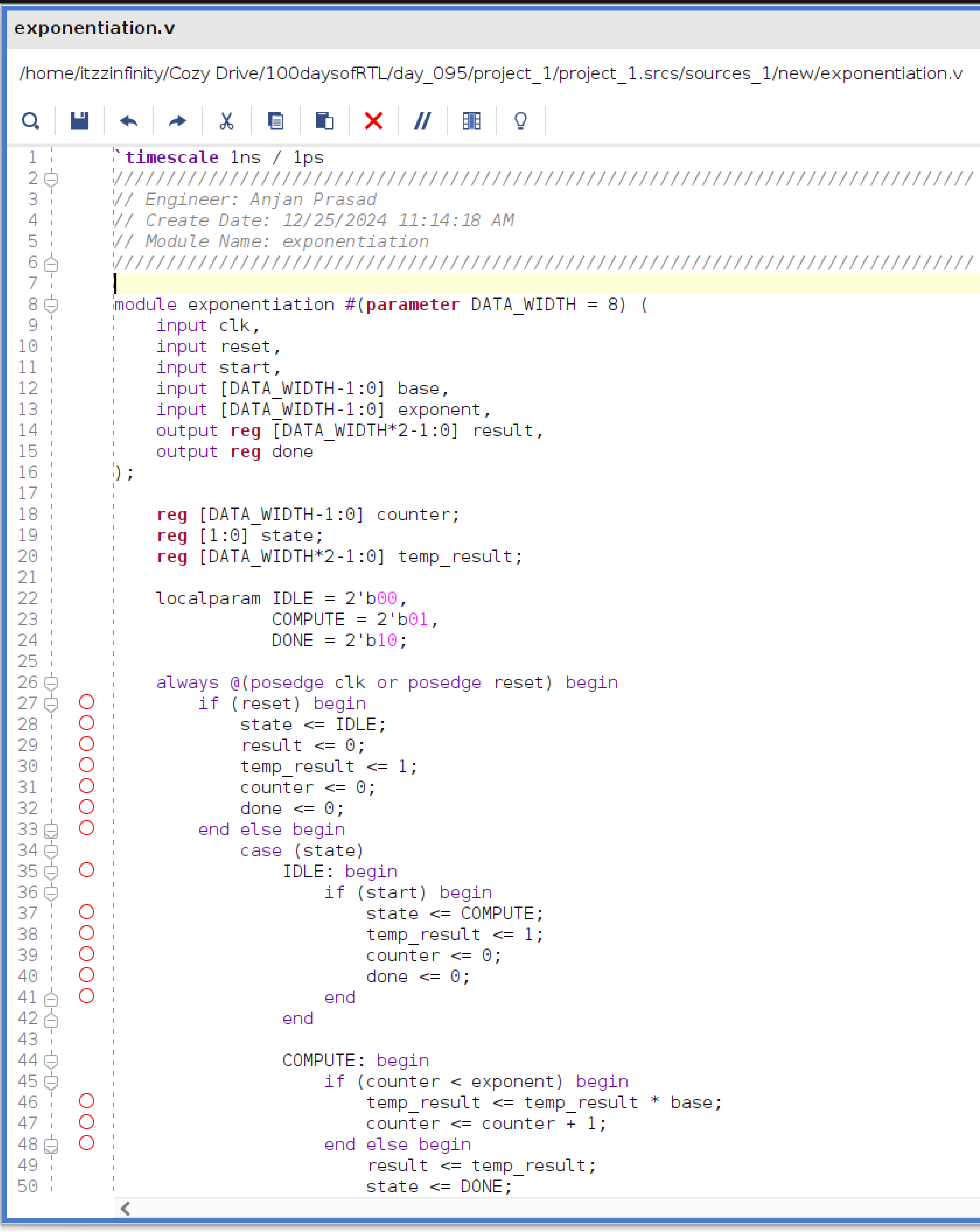


```
# run 1000ns
Time=0 | start=0 | base=  0 | exponent=  0 | result=    0 | done=0
Time=20000 | start=1 | base=  3 | exponent=  4 | result=    0 | done=0
Time=30000 | start=0 | base=  3 | exponent=  4 | result=    0 | done=0
Time=75000 | start=0 | base=  3 | exponent=  4 | result=   81 | done=0
Time=85000 | start=0 | base=  3 | exponent=  4 | result=   81 | done=1
Time=105000 | start=1 | base=  5 | exponent=  3 | result=   81 | done=0
Time=115000 | start=0 | base=  5 | exponent=  3 | result=   81 | done=0
Time=145000 | start=0 | base=  5 | exponent=  3 | result=  125 | done=0
Time=155000 | start=0 | base=  5 | exponent=  3 | result=  125 | done=1
Time=175000 | start=1 | base=  2 | exponent=  0 | result=  125 | done=0
Time=185000 | start=0 | base=  2 | exponent=  0 | result=    1 | done=0
Time=195000 | start=0 | base=  2 | exponent=  0 | result=    1 | done=1
$finish called at time : 205 ns : File "/home/itzzinfinity/Cozy Drive/100daysofRTL/day_095
```


Schematic
58 Cells
36 I/O Ports
124 Nets
base[7:0]
exponent[7:0]
state1_i
I0[7:0]
I1[7:0]
O
<
RTL_LT
counter_i
S=2'b00
S=2'b01
I0[7:0]
I1[7:0]
O[7:0]
S[1:0]
RTL_MUX
counter0_i
I1
I0[7:0]
O[7:0]
+
RTL_ADD
reset
clk
start
counter_reg[7:0]
CLR
C
CE
Q
D
RTL_REG_ASYNC
counter_i__0
S=2'b00
S=2'b01
S=2'b10
S=default
I0
I1
I2
I3
O
S[1:0]
RTL_MUX


temp_result0_i
I0[15:0]
I1[7:0]
*
O[15:0]
RTL_MULT
state_i__2
S=2'b00
I0
S=2'b01
I1
S=2'b10
I2
S=default
I3
O
S[1:0]
RTL_MUX
state_i__0
S=1'b1
I0
S=default
I1
O
S
RTL_MUX
V=8"000000
state_reg[1:0]
CLR
C
CE
Q
D
RTL_REG_ASYNC
state_i
V=8"01", S=2'b00
I0[1:0]
V=8"10", S=2'b01
I1[1:0]
S=2'b10
I2[1:0]
O[1:0]
S[1:0]
RTL_MUX
state_i__1
S=1'b0
I0
S=default
I1
O
S
RTL_MUX

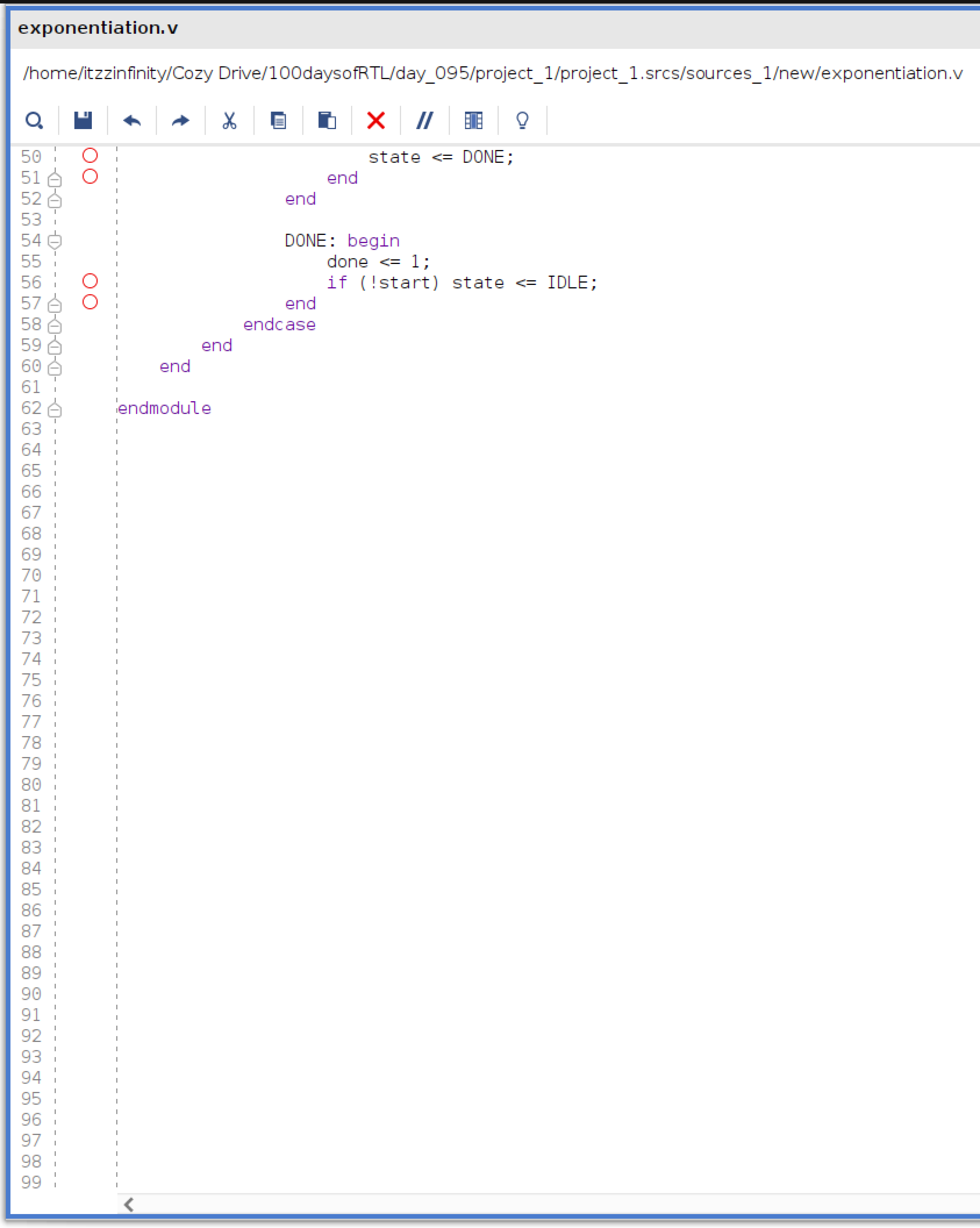

?
temp_result_i
000000000001", S=2'b00
I0[15:0]
S=2'b01
I1[15:0]
O[15:0]
S[1:0]
RTL_MUX
temp_result_reg[15:0]
CLR
C
CE
Q
D
PRE
RTL_REG_ASYNC
n/c
temp_result_i__0
S=2'b00
I0
S=2'b01
I1
S=2'b10
I2
S=default
I3
O
S[1:0]
RTL_MUX
result_i__0
S=2'b00
I0
S=2'b01
I1
S=2'b10
I2
S=default
I3
O
S[1:0]
RTL_MUX
result_reg[15:0]
CLR
C
CE
Q
D
RTL_REG_ASYNC
result[15:0]
result_i
S=1'b1
I0
S=default
I1
O
S
RTL_MUX
done_i
S=2'b00
I0
S=2'b10
I1
O
S[1:0]
RTL_MUX
done_reg
CLR
C
CE
Q
D
RTL_REG_ASYNC
done
done_i__0
S=2'b00
I0
S=2'b01
I1
S=2'b10
I2
S=default
I3
O
S[1:0]
RTL_MUX



exponentiation.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_095/project_1/project_1.srcs/sources_1/new/exponentiation.v

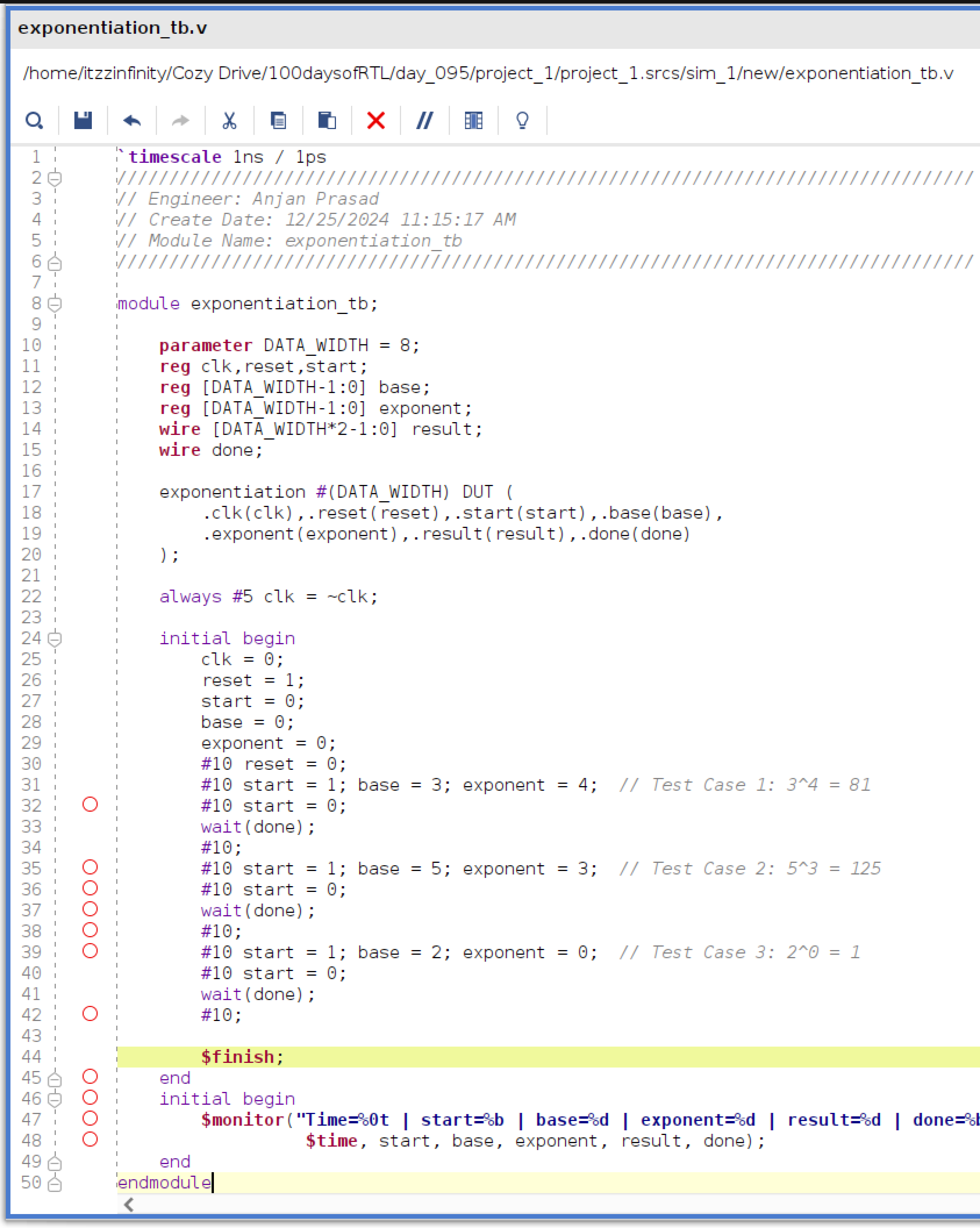
```verilog
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Engineer: Anjan Prasad
// Create Date: 12/25/2024 11:14:18 AM
// Module Name: exponentiation
//////////////////////////////////////////////////////////////////////////////////

module exponentiation #(parameter DATA_WIDTH = 8) (
    input clk,
    input reset,
    input start,
    input [DATA_WIDTH-1:0] base,
    input [DATA_WIDTH-1:0] exponent,
    output reg [DATA_WIDTH*2-1:0] result,
    output reg done
);

    reg [DATA_WIDTH-1:0] counter;
    reg [1:0] state;
    reg [DATA_WIDTH*2-1:0] temp_result;

    localparam IDLE = 2'b00,
               COMPUTE = 2'b01,
               DONE = 2'b10;

    always @(posedge clk or posedge reset) begin
        if (reset) begin
            state <= IDLE;
            result <= 0;
            temp_result <= 1;
            counter <= 0;
            done <= 0;
        end else begin
            case (state)
                IDLE: begin
                    if (start) begin
                        state <= COMPUTE;
                        temp_result <= 1;
                        counter <= 0;
                        done <= 0;
                    end
                end

                COMPUTE: begin
                    if (counter < exponent) begin
                        temp_result <= temp_result * base;
                        counter <= counter + 1;
                    end else begin
                        result <= temp_result;
                        state <= DONE;
```



exponentiation.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_095/project_1/project_1.srcs/sources_1/new/exponentiation.v

```
                        state <= DONE;
                    end
                end

                DONE: begin
                    done <= 1;
                    if (!start) state <= IDLE;
                end
            endcase
        end
    end

endmodule
```



exponentiation_tb.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_095/project_1/project_1.srcs/sim_1/new/exponentiation_tb.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Engineer: Anjan Prasad
// Create Date: 12/25/2024 11:15:17 AM
// Module Name: exponentiation_tb
//////////////////////////////////////////////////////////////////////////////////

module exponentiation_tb;

    parameter DATA_WIDTH = 8;
    reg clk,reset,start;
    reg [DATA_WIDTH-1:0] base;
    reg [DATA_WIDTH-1:0] exponent;
    wire [DATA_WIDTH*2-1:0] result;
    wire done;

    exponentiation #(DATA_WIDTH) DUT (
        .clk(clk),.reset(reset),.start(start),.base(base),
        .exponent(exponent),.result(result),.done(done)
    );

    always #5 clk = ~clk;

    initial begin
        clk = 0;
        reset = 1;
        start = 0;
        base = 0;
        exponent = 0;
        #10 reset = 0;
        #10 start = 1; base = 3; exponent = 4;  // Test Case 1: 3^4 = 81
        #10 start = 0;
        wait(done);
        #10;
        #10 start = 1; base = 5; exponent = 3;  // Test Case 2: 5^3 = 125
        #10 start = 0;
        wait(done);
        #10;
        #10 start = 1; base = 2; exponent = 0;  // Test Case 3: 2^0 = 1
        #10 start = 0;
        wait(done);
        #10;

        $finish;
    end
    initial begin
        $monitor("Time=%0t | start=%b | base=%d | exponent=%d | result=%d | done=%b
                  $time, start, base, exponent, result, done);
    end
endmodule
```