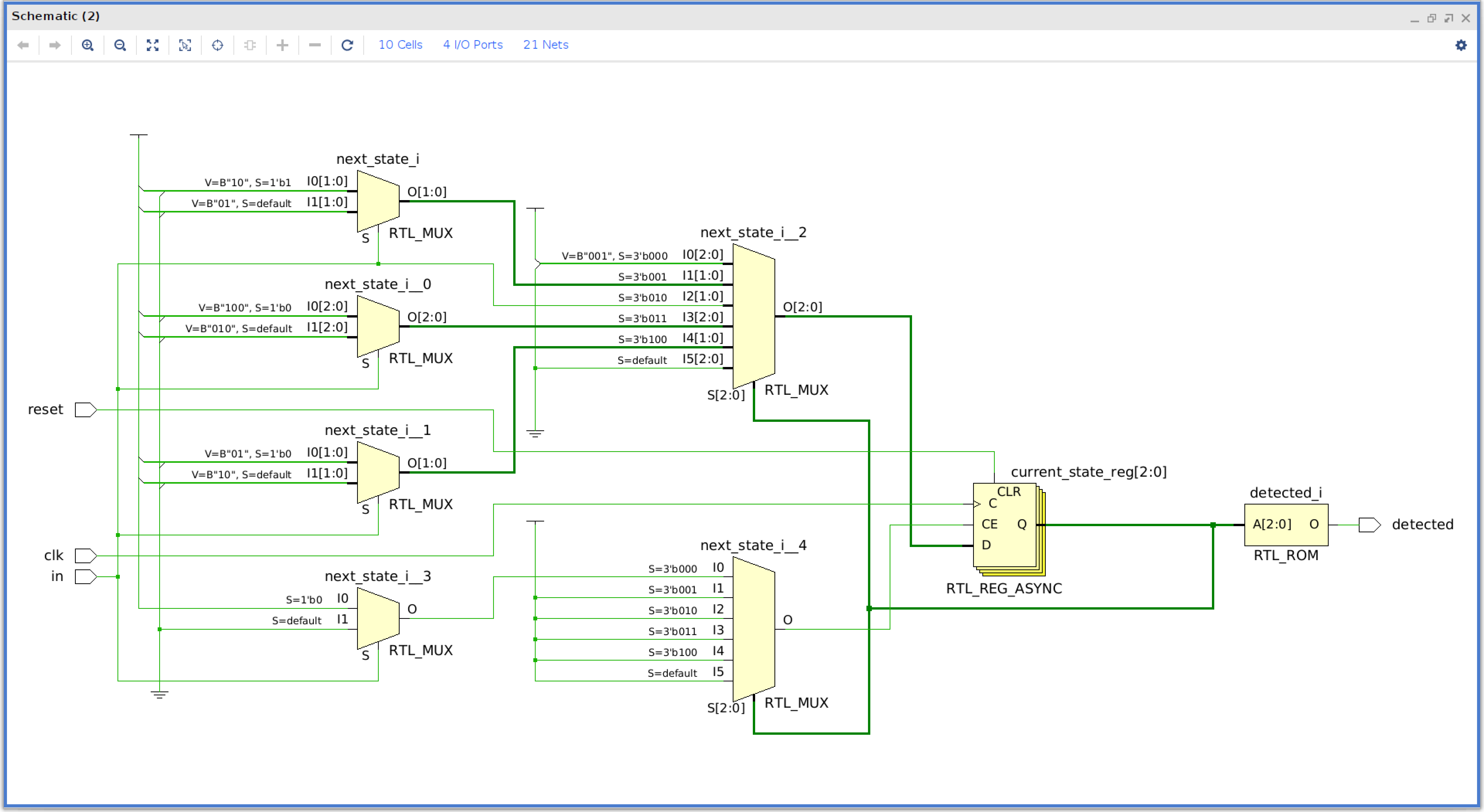
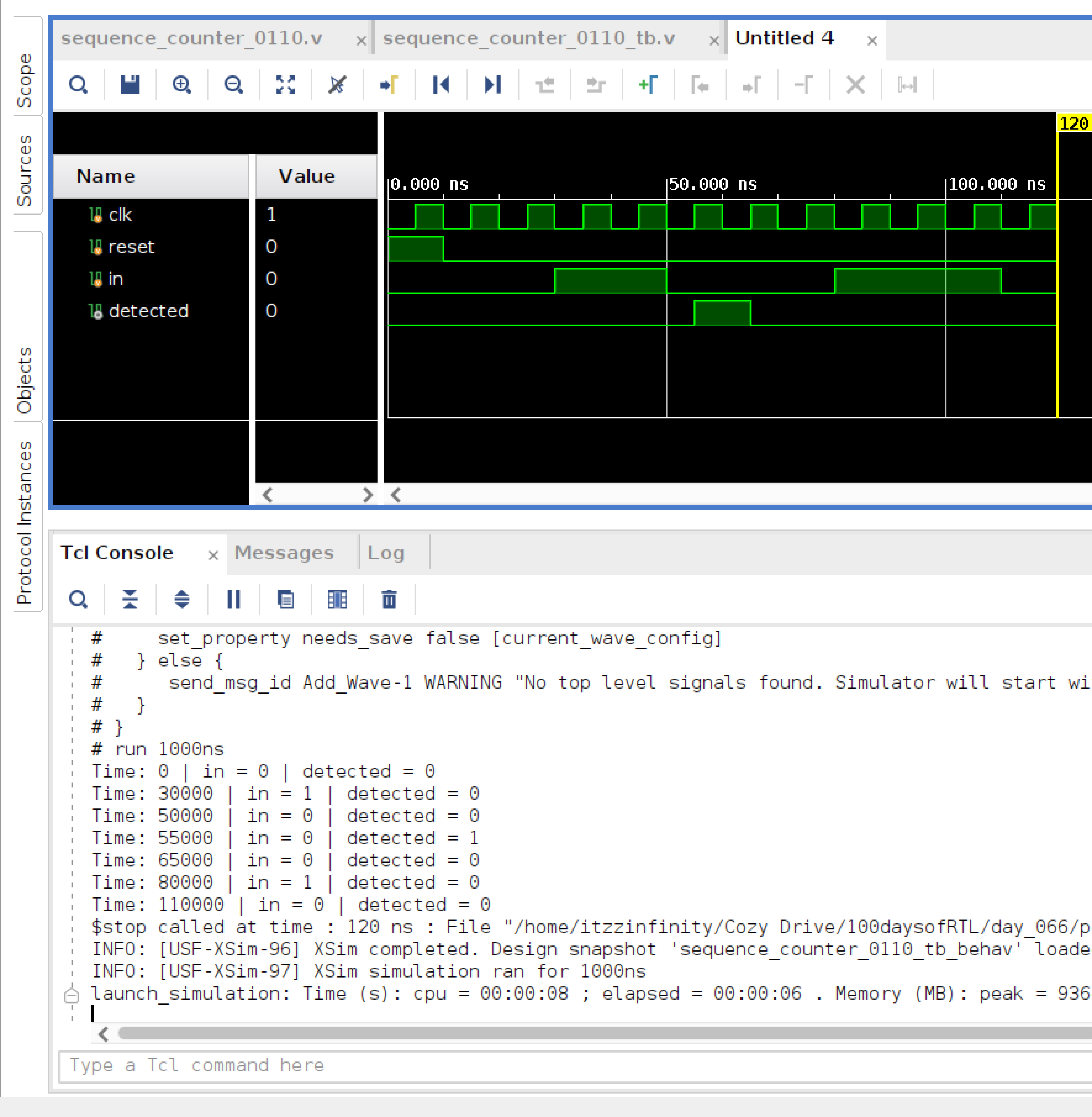
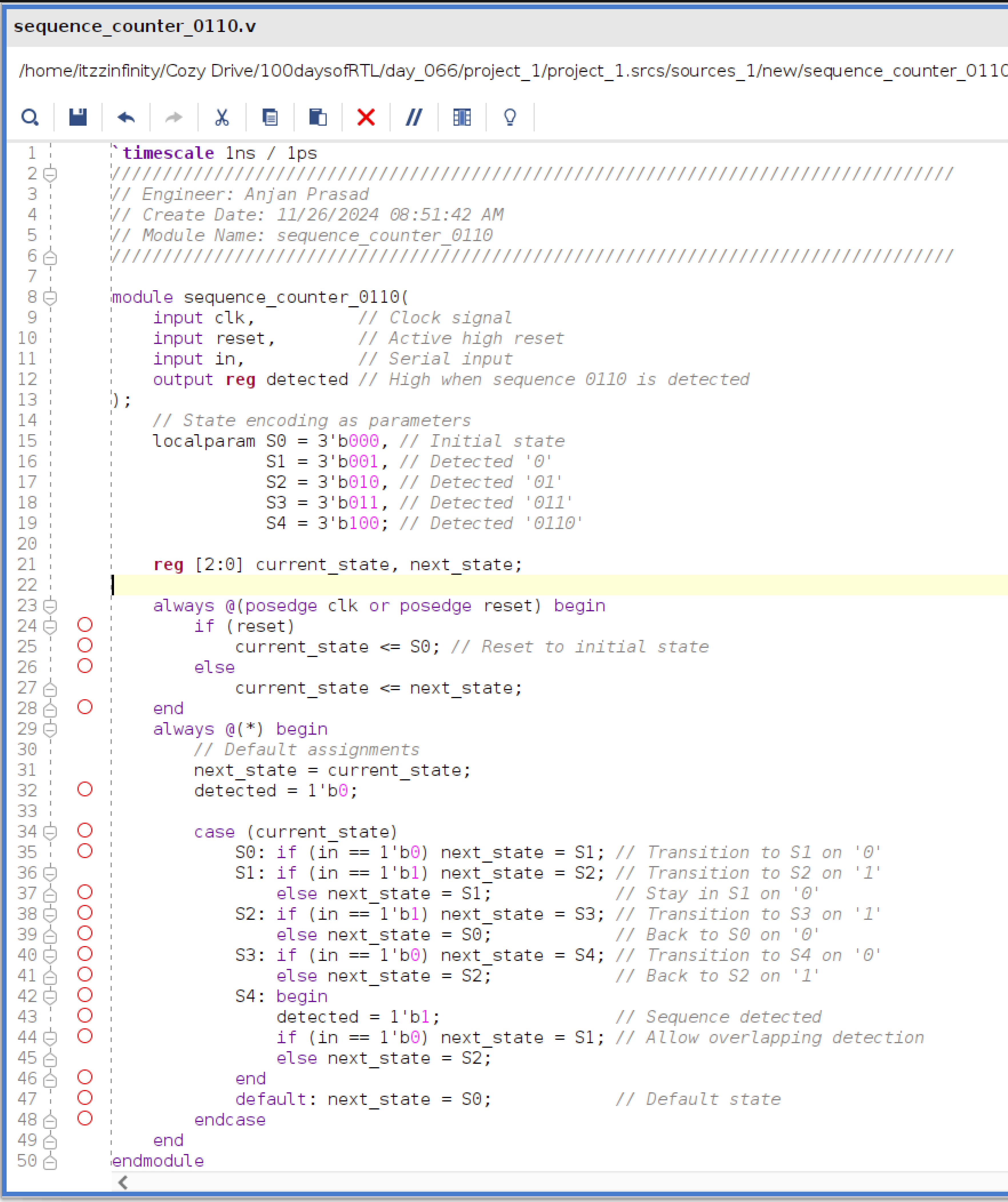

Schematic (2)
10 Cells
4 I/O Ports
21 Nets
next_state_i
V=B"10", S=1'b1
I0[1:0]
V=B"01", S=default
I1[1:0]
O[1:0]
S
RTL_MUX
next_state_i__0
V=B"100", S=1'b0
I0[2:0]
V=B"010", S=default
I1[2:0]
O[2:0]
S
RTL_MUX
next_state_i__2
V=B"001", S=3'b000
I0[2:0]
S=3'b001
I1[1:0]
S=3'b010
I2[1:0]
S=3'b011
I3[2:0]
S=3'b100
I4[1:0]
S=default
I5[2:0]
O[2:0]
S[2:0]
RTL_MUX
reset
next_state_i__1
V=B"01", S=1'b0
I0[1:0]
V=B"10", S=default
I1[1:0]
O[1:0]
S
RTL_MUX
current_state_reg[2:0]
CLR
C
CE
Q
D
RTL_REG_ASYNC
detected_i
A[2:0]
O
RTL_ROM
detected
clk
in
next_state_i__3
S=1'b0
I0
S=default
I1
O
S
RTL_MUX
next_state_i__4
S=3'b000
I0
S=3'b001
I1
S=3'b010
I2
S=3'b011
I3
S=3'b100
I4
S=default
I5
O
S[2:0]
RTL_MUX



```
#     set_property needs_save false [current_wave_config]
#   } else {
#      send_msg_id Add_Wave-1 WARNING "No top level signals found. Simulator will start wi
#   }
# }
# run 1000ns
Time: 0 | in = 0 | detected = 0
Time: 30000 | in = 1 | detected = 0
Time: 50000 | in = 0 | detected = 0
Time: 55000 | in = 0 | detected = 1
Time: 65000 | in = 0 | detected = 0
Time: 80000 | in = 1 | detected = 0
Time: 110000 | in = 0 | detected = 0
$stop called at time : 120 ns : File "/home/itzzinfinity/Cozy Drive/100daysofRTL/day_066/p
INFO: [USF-XSim-96] XSim completed. Design snapshot 'sequence_counter_0110_tb_behav' loade
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:08 ; elapsed = 00:00:06 . Memory (MB): peak = 936
```



sequence_counter_0110.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_066/project_1/project_1.srcs/sources_1/new/sequence_counter_0110

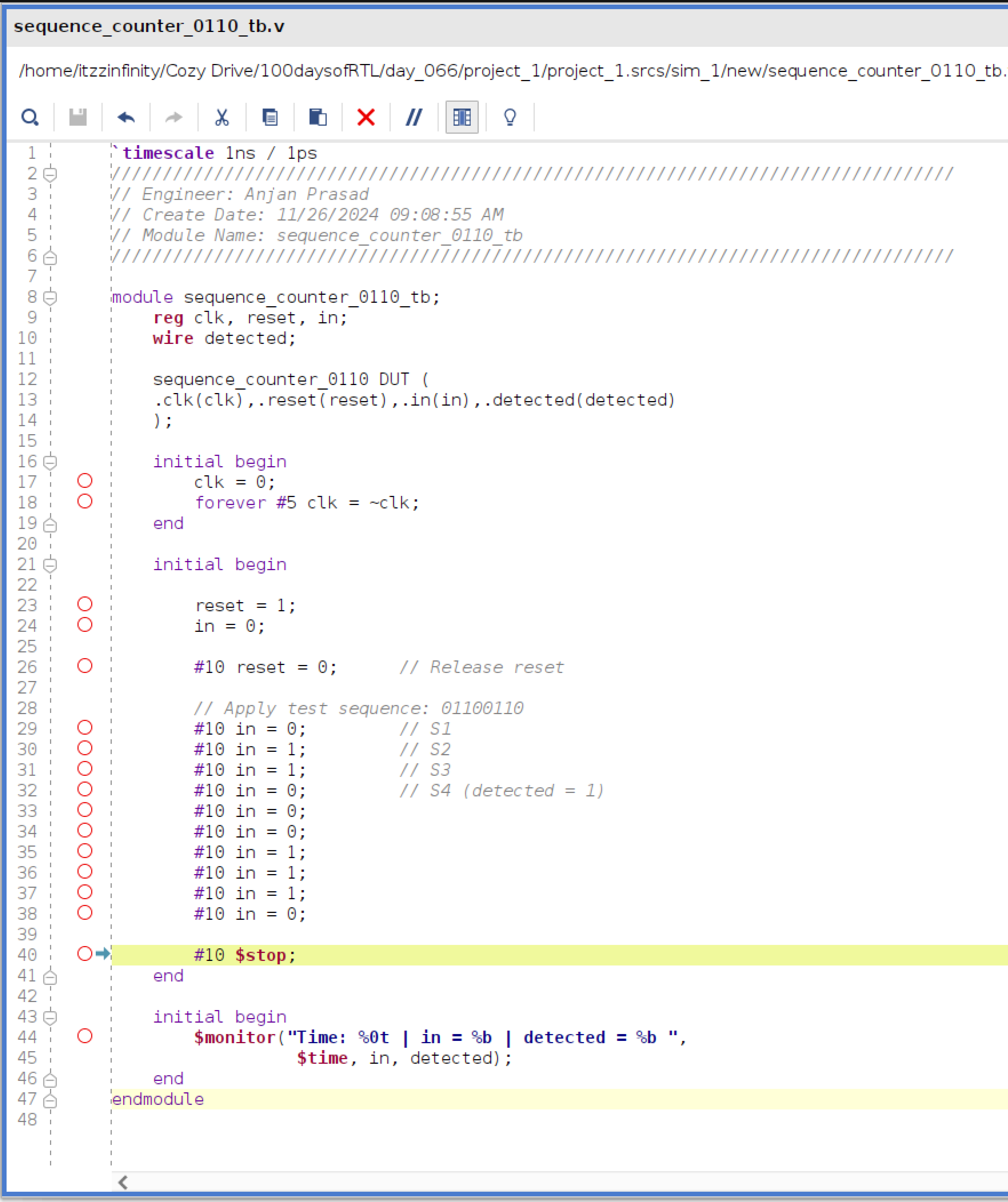
```verilog
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Engineer: Anjan Prasad
// Create Date: 11/26/2024 08:51:42 AM
// Module Name: sequence_counter_0110
//////////////////////////////////////////////////////////////////////////////////

module sequence_counter_0110(
    input clk,          // Clock signal
    input reset,        // Active high reset
    input in,           // Serial input
    output reg detected // High when sequence 0110 is detected
);
    // State encoding as parameters
    localparam S0 = 3'b000, // Initial state
               S1 = 3'b001, // Detected '0'
               S2 = 3'b010, // Detected '01'
               S3 = 3'b011, // Detected '011'
               S4 = 3'b100; // Detected '0110'

    reg [2:0] current_state, next_state;

    always @(posedge clk or posedge reset) begin
        if (reset)
            current_state <= S0; // Reset to initial state
        else
            current_state <= next_state;
    end
    always @(*) begin
        // Default assignments
        next_state = current_state;
        detected = 1'b0;

        case (current_state)
            S0: if (in == 1'b0) next_state = S1; // Transition to S1 on '0'
            S1: if (in == 1'b1) next_state = S2; // Transition to S2 on '1'
                else next_state = S1;            // Stay in S1 on '0'
            S2: if (in == 1'b1) next_state = S3; // Transition to S3 on '1'
                else next_state = S0;            // Back to S0 on '0'
            S3: if (in == 1'b0) next_state = S4; // Transition to S4 on '0'
                else next_state = S2;            // Back to S2 on '1'
            S4: begin
                detected = 1'b1;                 // Sequence detected
                if (in == 1'b0) next_state = S1; // Allow overlapping detection
                else next_state = S2;
            end
            default: next_state = S0;            // Default state
        endcase
    end
endmodule
```



sequence_counter_0110_tb.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_066/project_1/project_1.srcs/sim_1/new/sequence_counter_0110_tb.

```verilog
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Engineer: Anjan Prasad
// Create Date: 11/26/2024 09:08:55 AM
// Module Name: sequence_counter_0110_tb
//////////////////////////////////////////////////////////////////////////////////

module sequence_counter_0110_tb;
    reg clk, reset, in;
    wire detected;

    sequence_counter_0110 DUT (
    .clk(clk),.reset(reset),.in(in),.detected(detected)
    );

    initial begin
        clk = 0;
        forever #5 clk = ~clk;
    end

    initial begin

        reset = 1;
        in = 0;

        #10 reset = 0;      // Release reset

        // Apply test sequence: 01100110
        #10 in = 0;         // S1
        #10 in = 1;         // S2
        #10 in = 1;         // S3
        #10 in = 0;         // S4 (detected = 1)
        #10 in = 0;
        #10 in = 0;
        #10 in = 1;
        #10 in = 1;
        #10 in = 1;
        #10 in = 0;

        #10 $stop;
    end

    initial begin
        $monitor("Time: %0t | in = %b | detected = %b ",
                  $time, in, detected);
    end
endmodule

```