Digital Design and Computer Organisation Laboratory
3rd Semester, Academic Year 2025

NAME	SRN
IBRAHIM KHALEEL	PES2UG25CS809
DEERAJ S	PES2UG25CS806
L A GOKUL	PES2UG24CS246

TITLE: **Stop Watch(Start, Reset, Stop)**

Deliverables

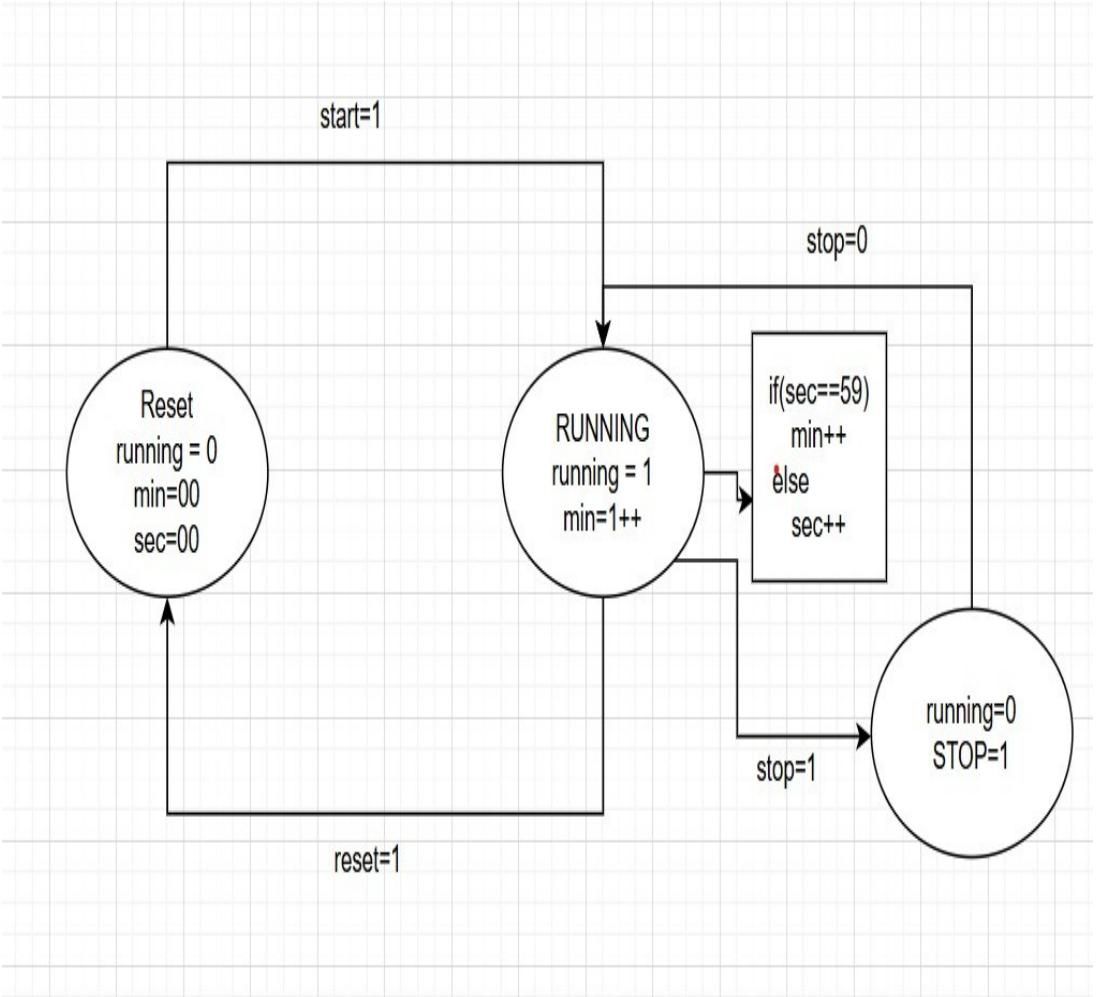
I. Verilog Code Screenshot

II. Verilog VVP Output Screen Shot

III. GTKWAVE Screenshot

IV. Output Table to be completed and included

State Diagram



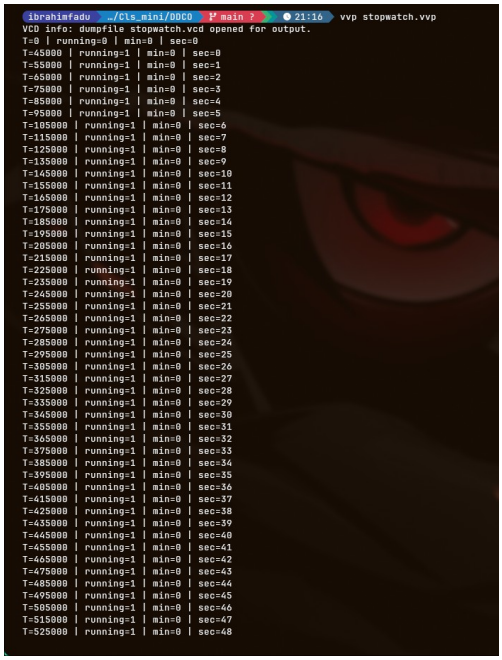
IVERLOG CODE

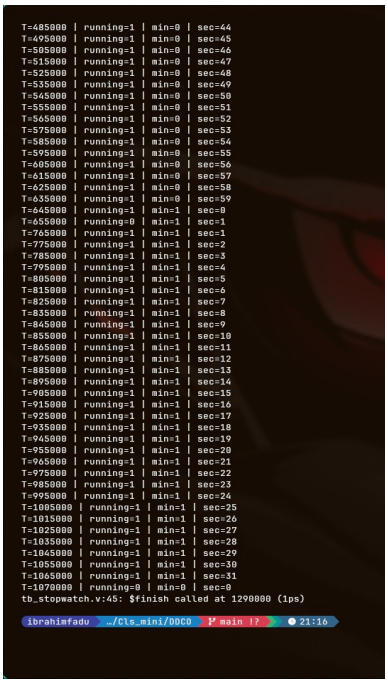
```
module stopwatch (
    input clk,          // clock input (1 Hz or divided clock)
    input reset,        // reset stopwatch
    input start,        // start counting
    input stop,         // stop counting
    output reg [5:0] sec,  // seconds (0-59)
    output reg [5:0] min   // minutes (0-59)
);

    reg running; // keeps track if stopwatch is running

    always @(posedge clk or posedge reset) begin
        if (reset) begin
            // Reset all values
            sec <= 0;
            min <= 0;
            running <= 0;
        end
        else begin
            // Control logic
            if (start)
                running <= 1;
            else if (stop)
                running <= 0;

            // Time counting
            if (running) begin
                if (sec == 59) begin
                    sec <= 0;
                    if (min == 59)
                        min <= 0;     // reset after 59:59
                    else
                        min <= min + 1;
                end
                else begin
                    sec <= sec + 1;
                end
            end
        end
    end
endmodule

```

TEST BENCH

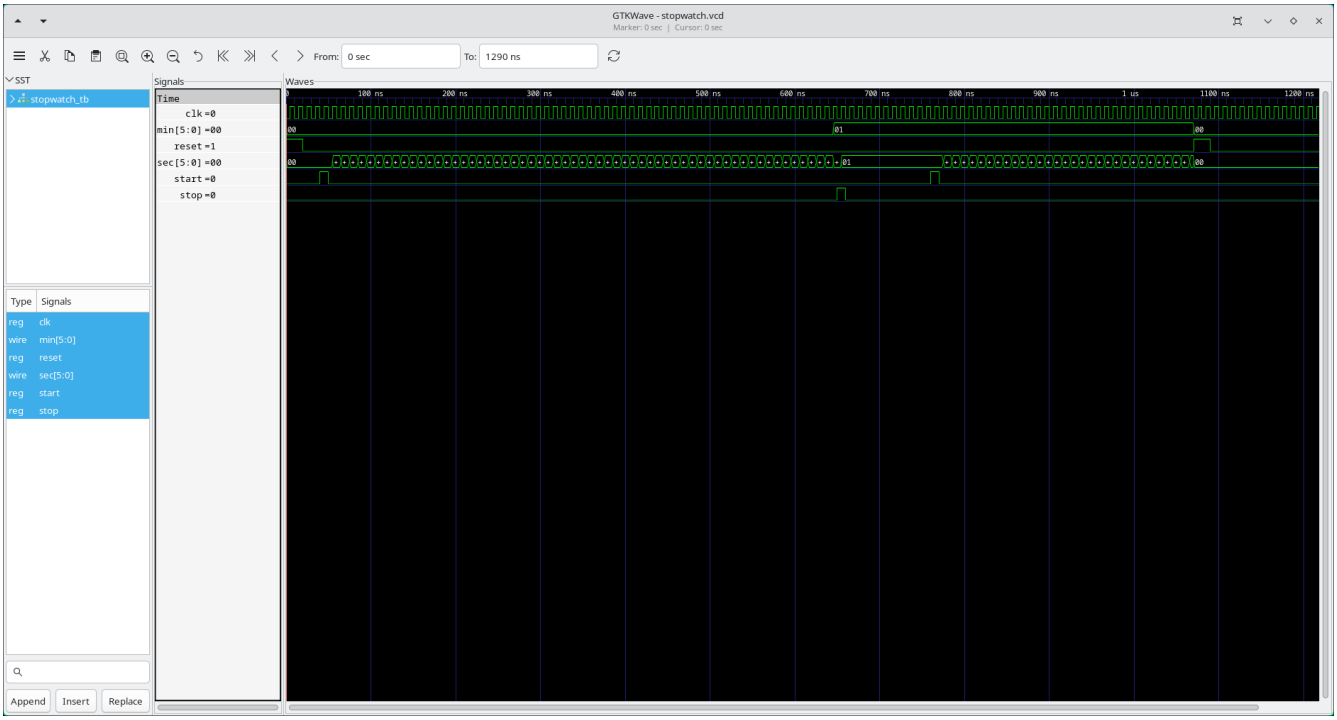
```
`timescale 1ns/1ps

module stopwatch_tb;
    reg clk, reset, start, stop;
    wire [5:0] sec, min;

    // Instantiate the stopwatch
    stopwatch uut (
        .clk(clk),
        .reset(reset),
        .start(start),
        .stop(stop),
        .sec(sec),
        .min(min)
    );

    // Clock generation: toggle every 5 ns -> 100 MHz
    initial begin
        clk = 0;
        forever #5 clk = ~clk;
    end

    initial begin
        // For GTKWave dump
        $dumpfile("stopwatch.vcd");
        $dumpvars(0, stopwatch_tb);

        // Initialize signals
        reset = 1; start = 0; stop = 0;
        #20 reset = 0;    // Release reset after 20 ns

        #20 start = 1;    // Start counting
        #10 start = 0;    // Pulse style start button

        // Let it run for a while
        #600 stop = 1;    // Stop after ~600 ns
        #10 stop = 0;

        #100 start = 1;   // Start again
        #10 start = 0;

        #300 reset = 1;   // Reset stopwatch
        #20 reset = 0;

        #200 $finish;     // End simulation
    end

    // Display output on console
    initial begin
        $monitor("T=%0t | running=%b | min=
 | sec=          $time, uut.running, min, sec);
",  end
endmodule

```

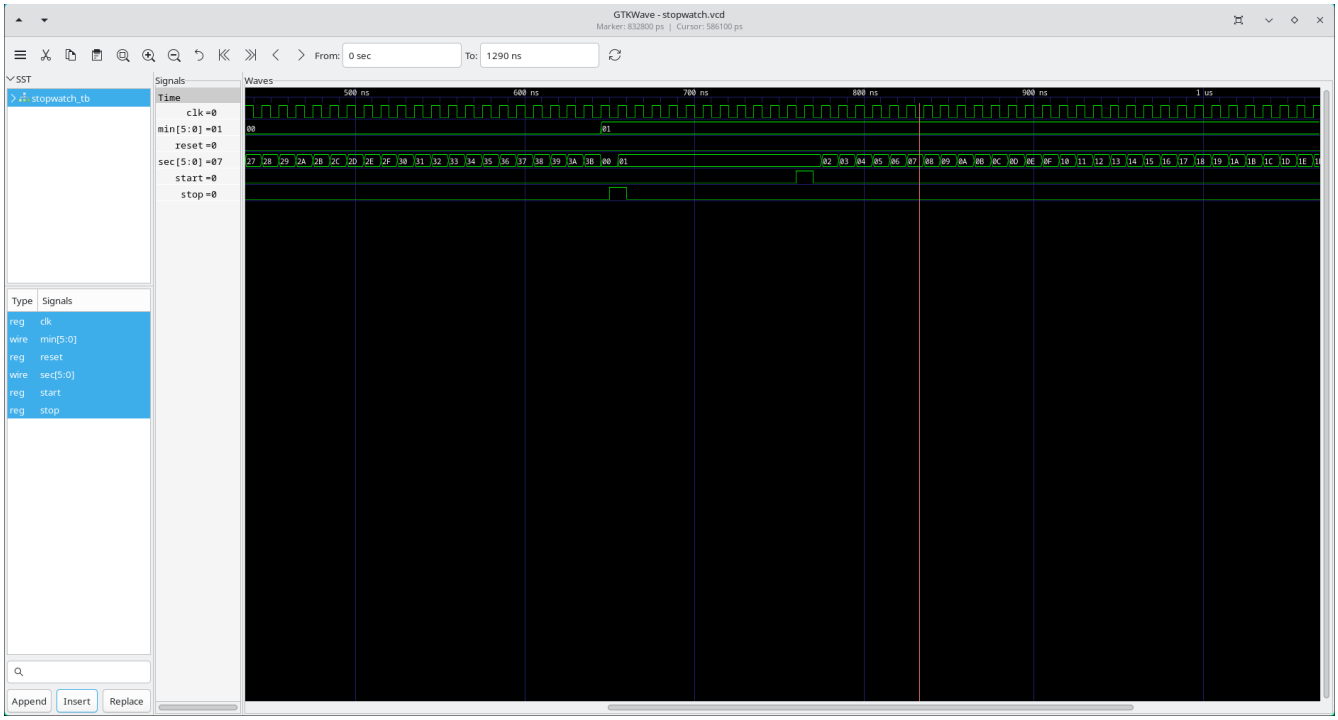
VVP OUTPUT





GTKWAVE




GTKWave - stopwatch.vcd
Marker: 832800 ps | Cursor: 586100 ps
From: 0 sec
To: 1290 ns
SST
stopwatch_tb
Type Signals
reg clk
wire min[5:0]
reg reset
wire sec[5:0]
reg start
reg stop
Append
Insert
Replace
Signals
Time
clk =0
min[5:0] =01
reset =0
sec[5:0] =07
start =0
stop =0
Waves
500 ns
600 ns
700 ns
800 ns
900 ns
1 us