Course Name: EMBEDDED SYSTEMS I / III

Course Number and Section: **14:332:493:03 / 16:332:579:05**

Year: Spring 2025

Lab Report #: 0

Lab Instructor: Milton Diaz

Student Name and RUID: Nikhil Chandra, 208005066

Date Submitted: 2/7/2025

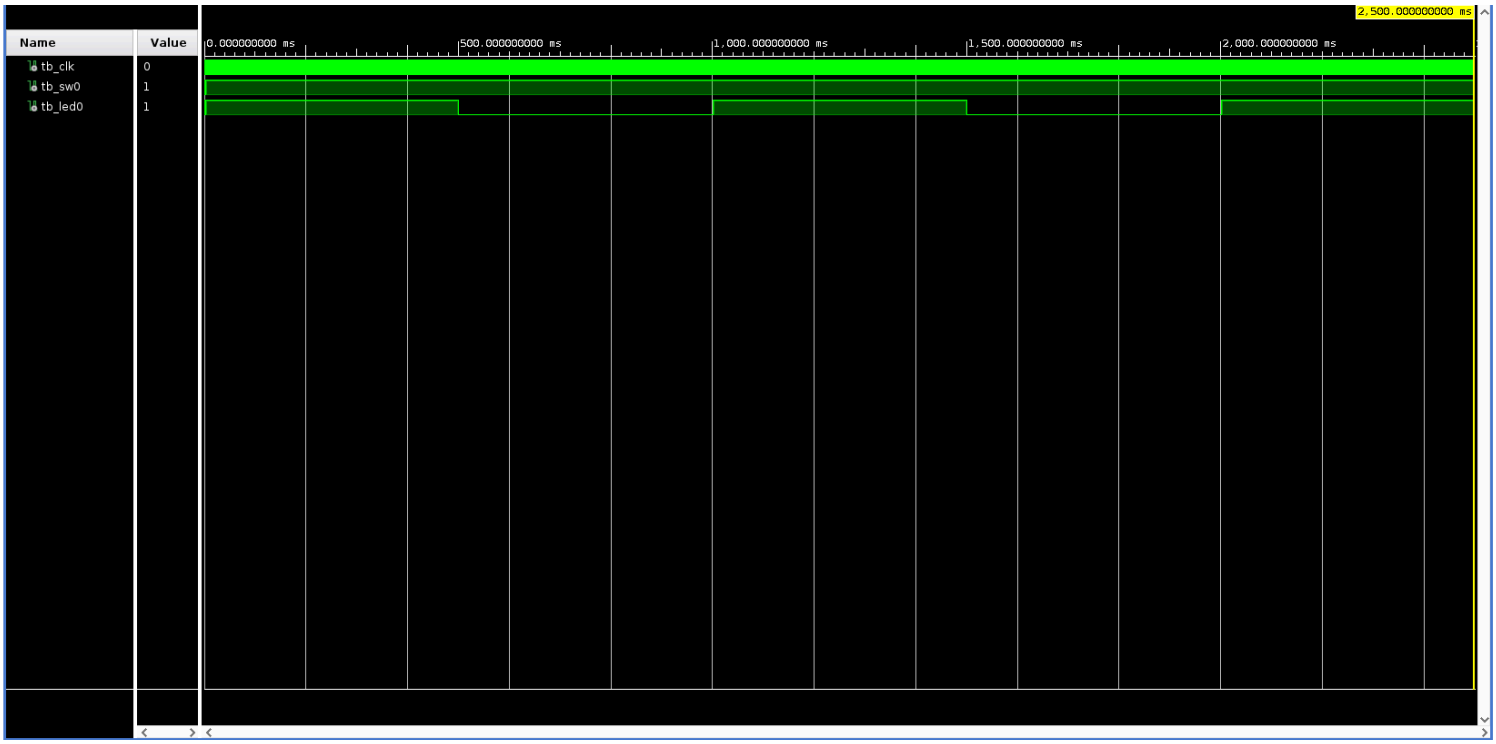
GitHub Link: [embedded-systems-1-spring-2025-labs/lab-0-nikhilChandra-coder: lab-0-nikhilChandra-coder created by GitHub Classroom]

Purpose/Objective: To understand how to use a counter to slow down the processes to our desired frequency

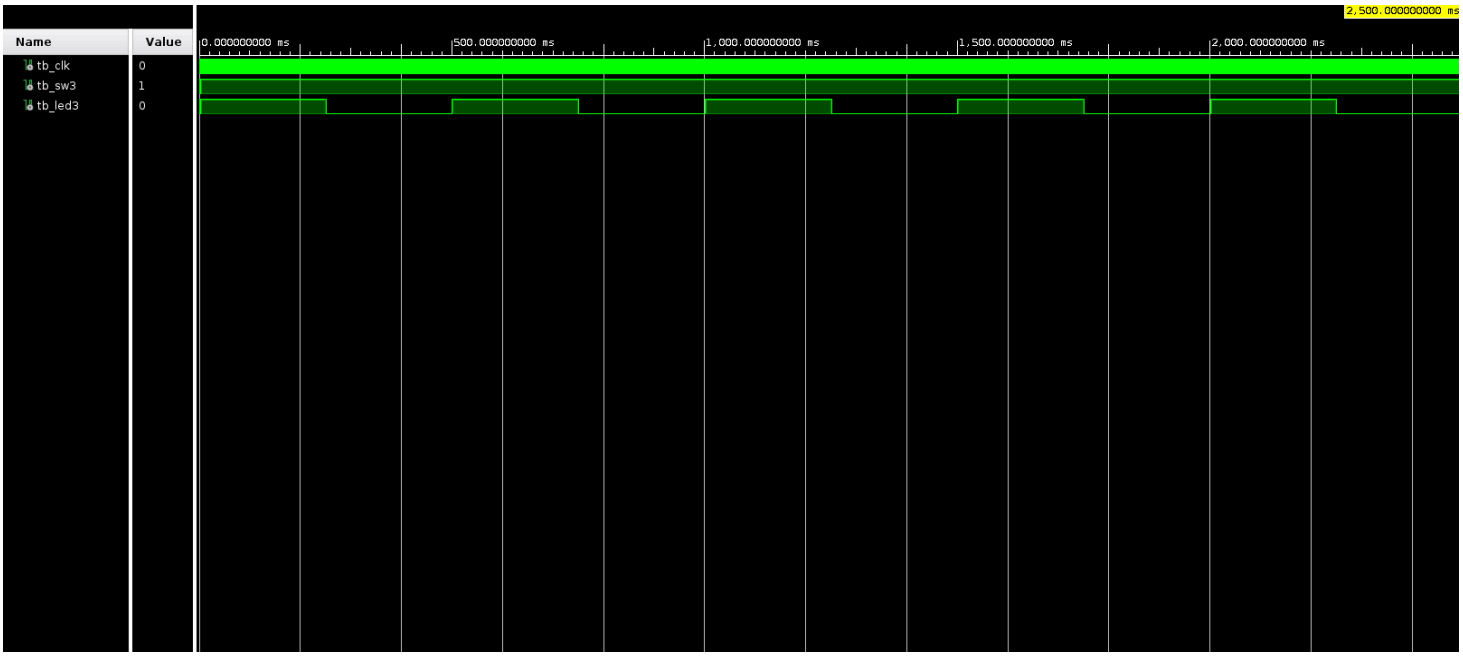
Theory of Operation:

Create new project → select target device (zybo) →import design and constraint files → run synthesis, simulation, generate bitstream →upload program/test on zybo → modify counter for new blinker frequency/adjust as needed

Simulation Waveforms:



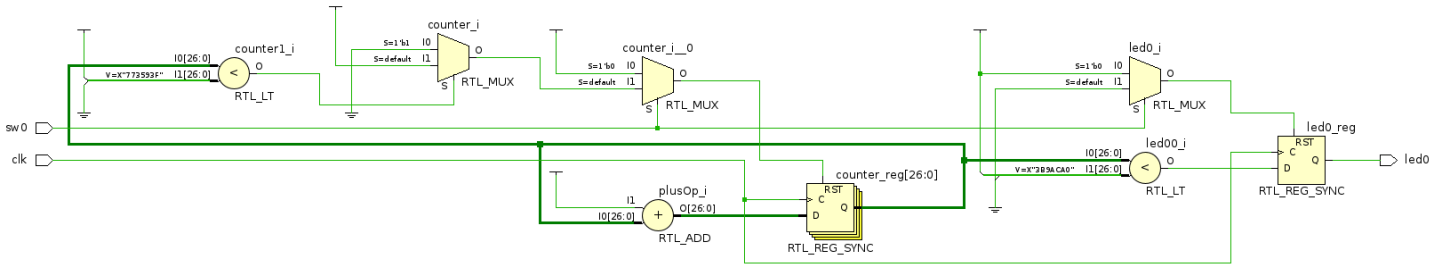
Blinker simulation - 1Hz



2Hz

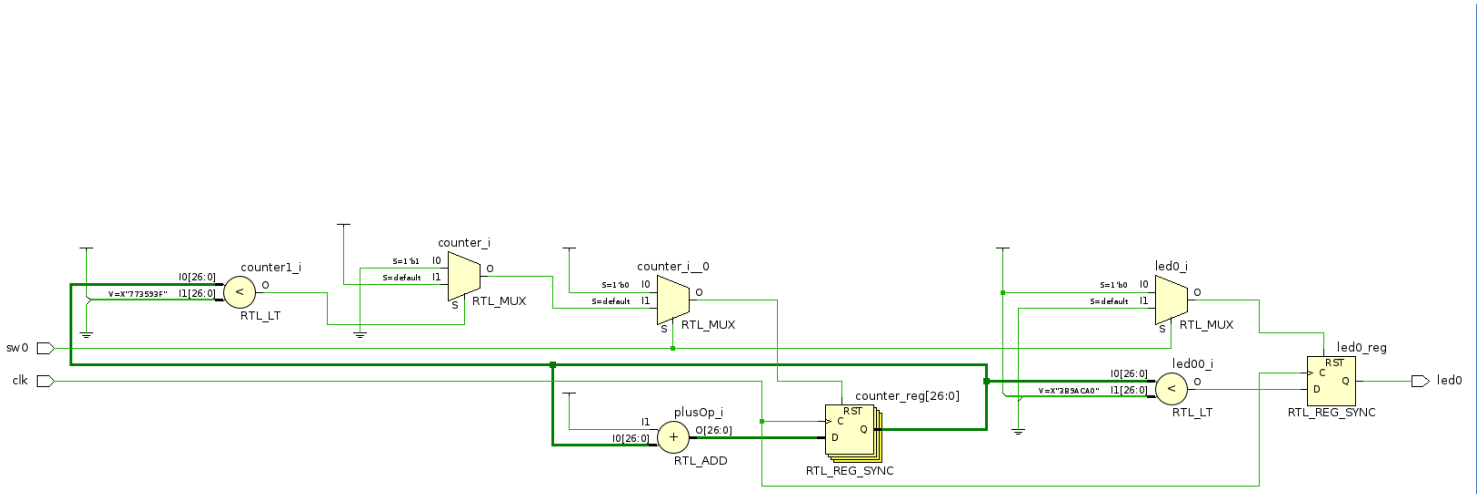
Vivado Schematics:

a) Vivado Elaboration Schematic



i)

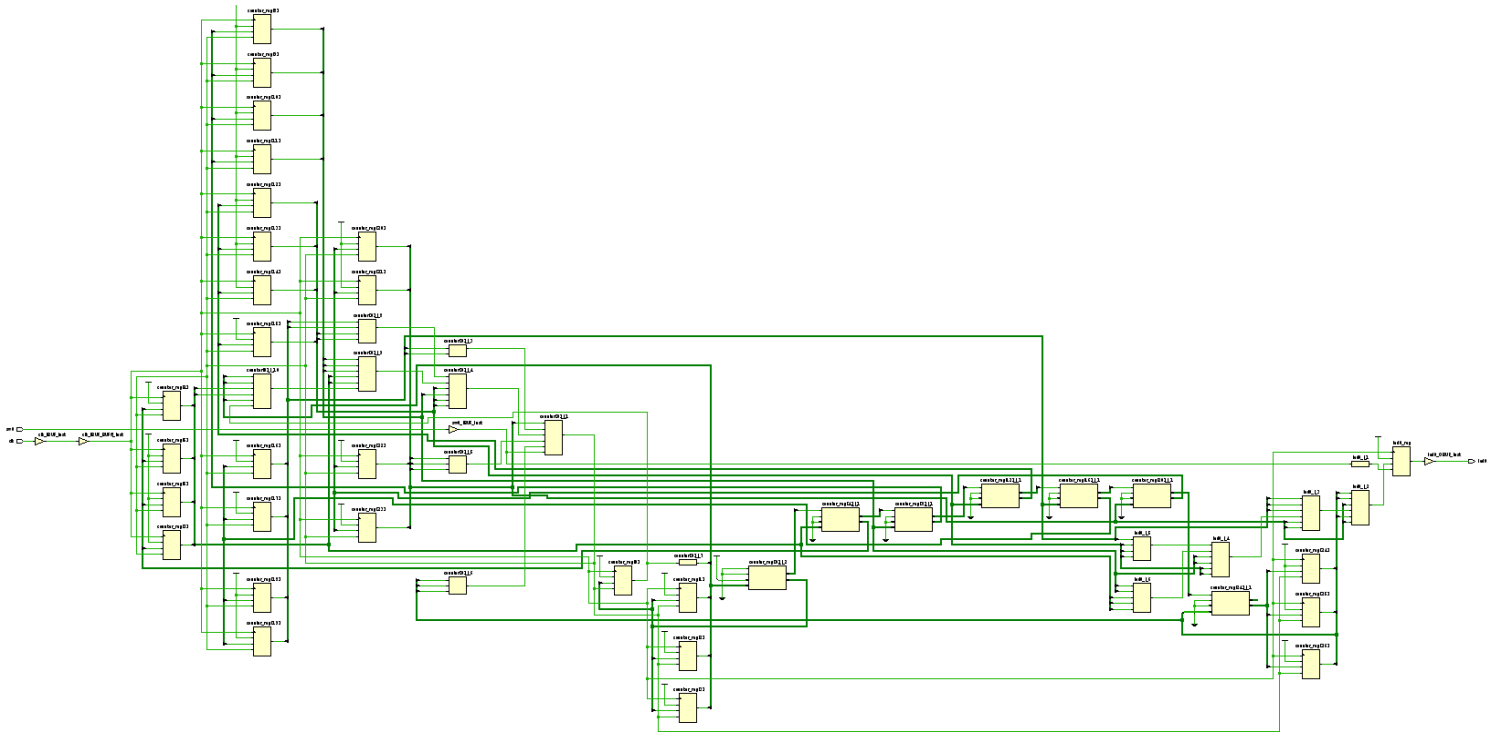
Blinker 1Hz elaboration schematic



ii)

2Hz

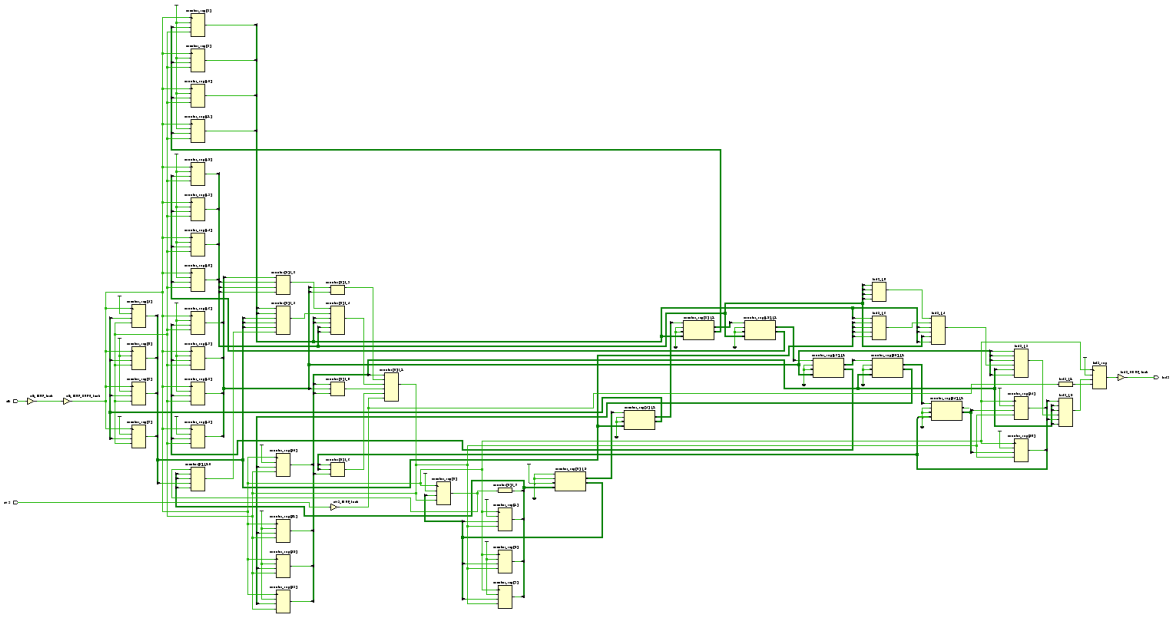
b) Vivado Synthesis Schematic



i)

Blinker 1Hz synthesis schematic

ii)



2Hz synthesis schematic

c) Post- Synthesis Utilization Table

i)

Resource	Estimation	Available	Utilization %
LUT	14	17600	0.08
FF	28	35200	0.08
IO	3	100	3.00
BUFG	1	32	3.13

Blinker 1Hz post-synthesis utilization table

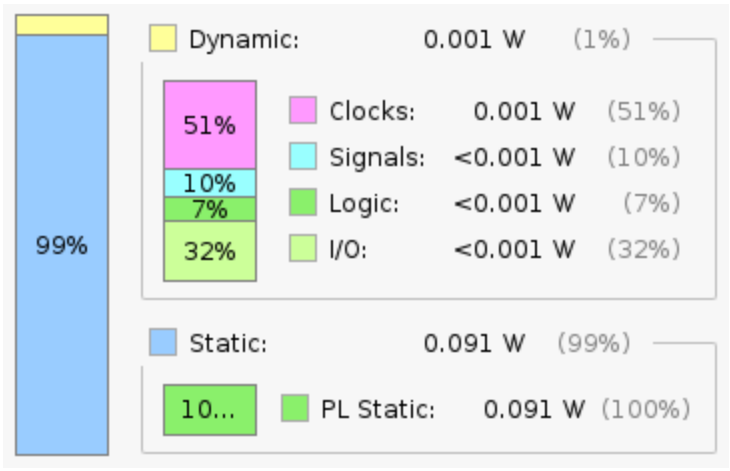
ii)

Resource	Estimation	Available	Utilization %
LUT	14	17600	0.08
FF	27	35200	0.08
IO	3	100	3.00
BUFG	1	32	3.13

2 Hz post-synthesis utilization table

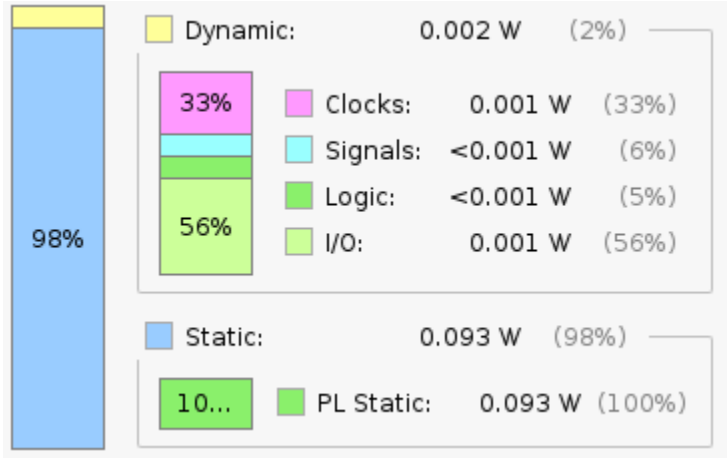
d) On-Chip Power Graphs

i)



Blinker 1Hz on-chip power graphs

ii)



2 Hz on-chip power graphs

For part 2, the pins for led0 and sw0 were on, and for part 3, the pins for led0 and sw0 were turned off and the pins for led3 and sw3 were turned on

Answers to Additional Questions and Extra Credit:

Part 2 – Setup and Analysis

Please answer the following questions:

1. Why did we simulate the design for such a long period of time?

   We set the simulation time to 2.5s as the led blinked at 1Hz and to verify that it blinked at the desired frequency and check its behavior over at least 2 cycles

2. What was the counter doing before the switch went high? What was it doing after the switch went high?

   The counter was set to 0 before the switch went high. After the switch went high, the counter begins counting up and resets when it hits 124999999

3. Why do you think the count value was set to 124999999?

   The counter variable increments from zero to 124999999 to divide the Zybo board's 125 MHz clock signal to a 1 Hz clock signal.

4. Comparing the imported vhdl code and simulation results, do you think the simulation behaves as expected? Why or why not?

   It does behave as expected because in the blinker_tb.vhd (testbench), the switch is turned on after 1ms and left on. This is supposed to trigger the blinker entity in blinker.vhd, which alternates between on and off every 1 second

Part 3 – Modification

1. What is the new counter value? Why?

   The new counter value is 62499999. This represents the amount of time after which the counter resets. It is half the original value (124999999) because we want the new frequency to be 2Hz, and so require the counter to reset twice as early as before.

2. Besides the count value, what else (if anything) did you have to change in the design to successfully complete the design modifications?

   1 also needed to change when the led turns on. This is done by changing the if statement to unsigned(counter) < 31250000, which ensures that the led is turned on. The pins for LED 3 and switch 3 were enabled, while the previously used pins were disabled.

3. In what ways (if any) did the RTL and Synthesis schematics change?

   The schematic does not change as the logic is essentially the same as before.

Conclusion: This lab builds on Vivado fundamentals from the introductory lab, adding hands-on experience in generating and running a bitstream on a Zybo board via a virtual lab. During the demo, I extended a 1 Hz signal by a factor of four by increasing the counter register from 27 to 29 bits. The counter functions as a frequency divider, creating a clock signal for the LED based on the board's clock frequency.

Follow Up: I understand the concept of the counter to slow down the onboard clock and the testbench to simulate the code.