

# **COL215: Digital Logic and System Design**

## **Lab 3 Report**

### **7-Segment Display on Basys3**

Parth Arvind Rathi 2024CS50875

Harsh Verma 2024CS10499

---

## **1. Introduction**

The goal of this lab assignment is to design and simulate a Verilog module that can capture and display a 4-digit decimal number using the 7-segment display units on the Basys3 FPGA board. The digits are entered using switches (SW0–SW9) and stored sequentially into four internal registers, triggered by switches SW10–SW13.

## **2. Design Overview**

### **2.1 Digit Capture Logic**

The 10-bit input from SW0–SW9 is treated as a one-hot representation of digits 0–9. Depending on which of SW10–SW13 is high, the value is stored in one of four registers: digit0 to digit3. This is done on every rising edge of the clock. Only one digit is captured at a time.

### **2.2 Display Multiplexing**

Since there is only one set of cathode signals for all four displays, a 2-bit counter is used to cycle through the four digits. A 17-bit clock divider creates a 1ms refresh cycle per digit (for 100MHz clock). The corresponding anode is enabled (active-low), and its digit is decoded into a 7-bit pattern for the cathode.

## **3. Constraints**

- Clock pin: W5 mapped to clk
- Switches SW0–SW13: Mapped using constraints file
- 7-segment segment pins: [6:0] seg and dp
- Anode control: [3:0] an
- All pins were mapped according to the provided basys3.xdc file, ensuring compatibility with the Basys3 board.

## 4. Simulation Results

### 4.1 Functional Verification

The simulation was run in Vivado using the testbench `display_tb.v`. Switches were toggled sequentially to enter different digits. Simulation confirms correct digit capture and segment output patterns.

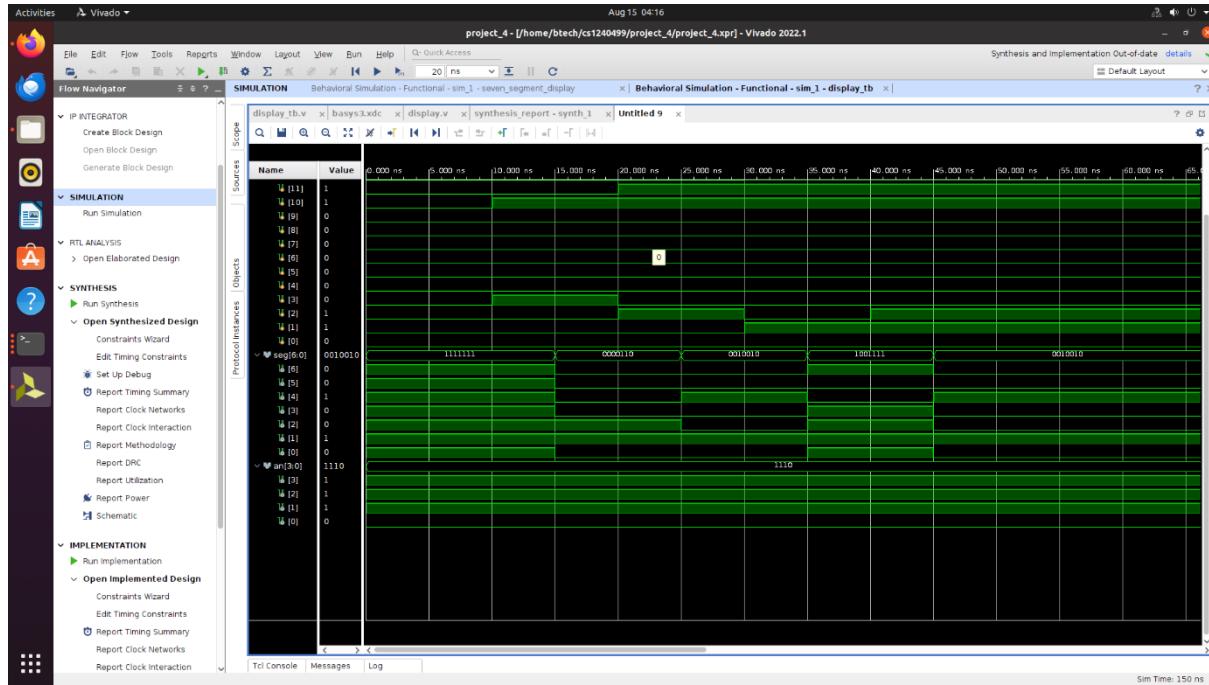


Figure 1: Simulation waveform showing digit loading and segment control

## 5. Synthesis and Resource Utilization

The design was synthesized successfully with no critical warnings. Utilization summary:

LUT	FF	BRAM	URAM	DSP
32	59	0	0	0
32	59	0	0	0

LUT: 32 & Flipflop:59

## 6. Generated Schematic

The schematic below shows the internal connectivity of the synthesized design. The anode decoder, digit storage, and segment mapping logic are visible.

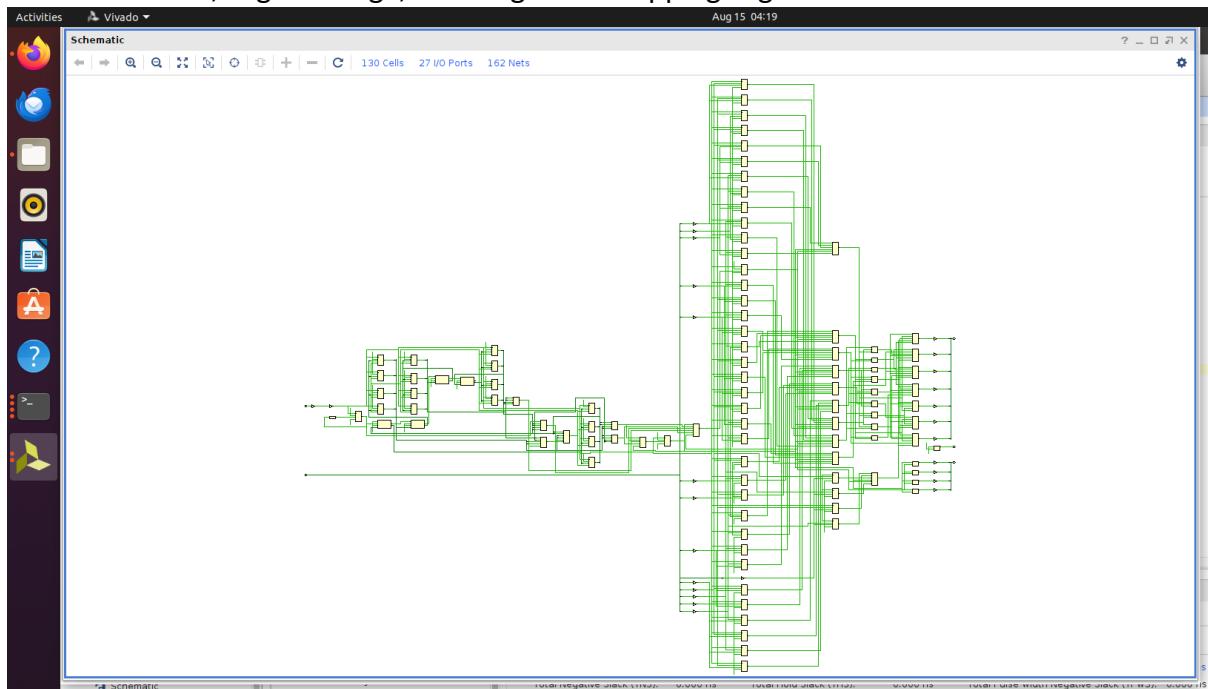


Figure 3: Generated Schematics

## 7. Design Decisions

- Used one-hot encoding for digit input to avoid extra decoding logic.
- Created a clock divider to generate 4ms refresh period (digit period = 1ms) for display cycling.
- Reused digit display logic from Assignment 2 for simplicity.
- Used active-low logic for anodes and cathodes as per Basys3 manual.

## 8. Hardware Simulation

The figure shows the successful hardware implementation of 7-segment display

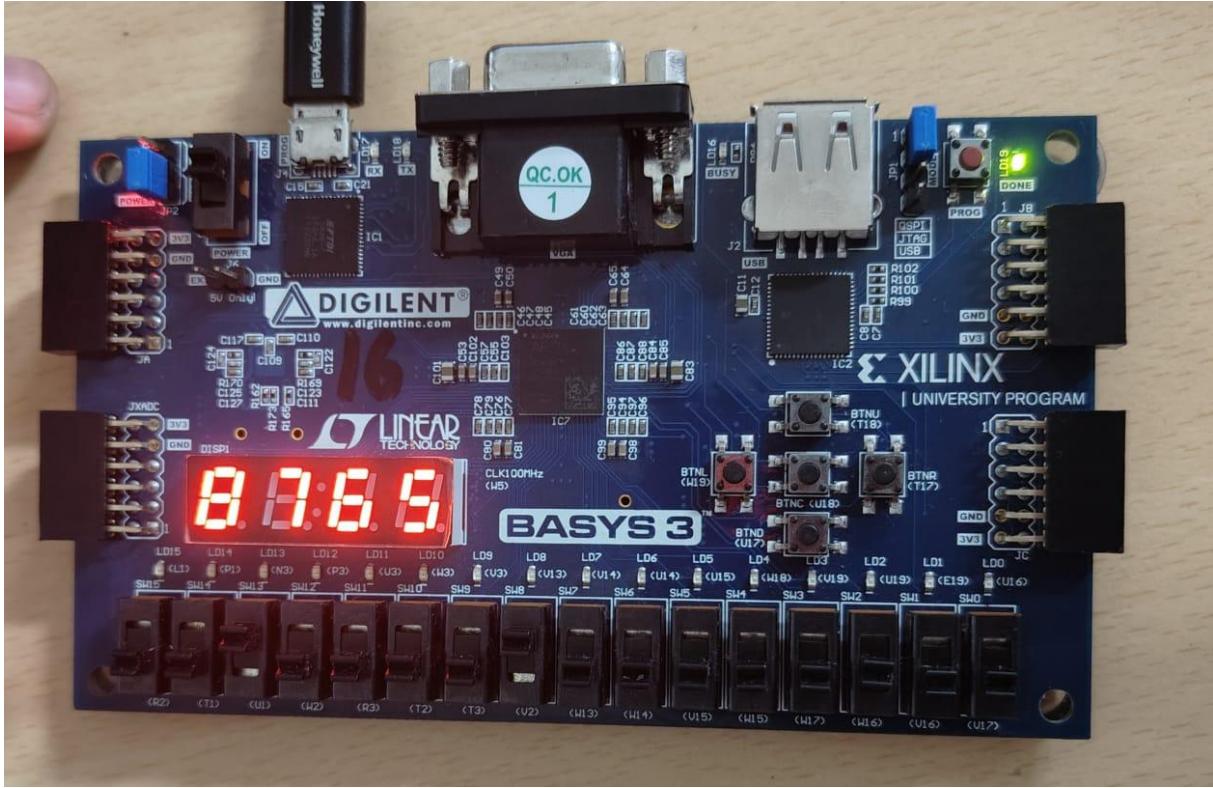


Figure 4: Hardware Demo

## 9. Conclusion

This lab demonstrated how to use Verilog to interact with real FPGA hardware and how to manage display multiplexing using digital logic. The system correctly captures and displays four user-entered digits on the 7-segment display using timing-based multiplexing.