ECE 2031 FINAL PROJECT PROPOSAL

INTRODUCTIONS

Patrick Mock
2nd Year EE Major

Mitchell Turton
2nd Year CompE Major

Anthony Kennedy
2nd Year EE Major

Chinmay Bandapalli
1st Year EE Major

Landon Slater
1st Year CompE Major

DESIGN PROBLEM AND PROJECT SUMMARY

- For this project we were tasked with creating a SCOMP peripheral that:
  - Can interface with and control the LEDs on the DE10 board
  - Provides functionalities that would be useful to a SCOMP programmer
  - Is easy and intuitive for a SCOMP programmer to use

OUR SOLUTION TO THE TASK AT HAND

Create a peripheral for SCOMP with the following functionalities:

Set all LEDs using a bitmask

Sets which LEDs are turned on by sending a 10-bit value to serve as a bit mask

Turn on a specific LED

Targets a specific LED with an index from 0 to 9, and sets its value without affecting the other LEDs

Set brightness of a specific LED

Sets the brightness of a specified LED with an index from 0 to 9, with a brightness represented by 8 bits with values from 0 to 255

IMPLEMENTATION

To be able to perform more than one functionality we split the 16-bit IO_Data input from SCOMP into several sections:

- OP_Code: 2 bits that represent what function the peripheral will perform
- LED_Sel: 4 bits that determine which LED we are targeting
- Data: 10 bits that store the data itself we want to send to the peripheral
  - All 10 used for setting bitmask
  - Only lower 8 bits used when setting brightness
  - Only first bit used when setting a single LED state

```
[      x x      |     x x x x     | x x x x x x x x x x ]
[ OP_Code (2 bits) | LED_Sel (4 bits) |    Data (10 bits)   ]
```

OP_Code	Function
00	Set bitmask
01	Set Single Bit
10	Set Brightness
11	Open for a potential extra feature

IMPLEMENTATION

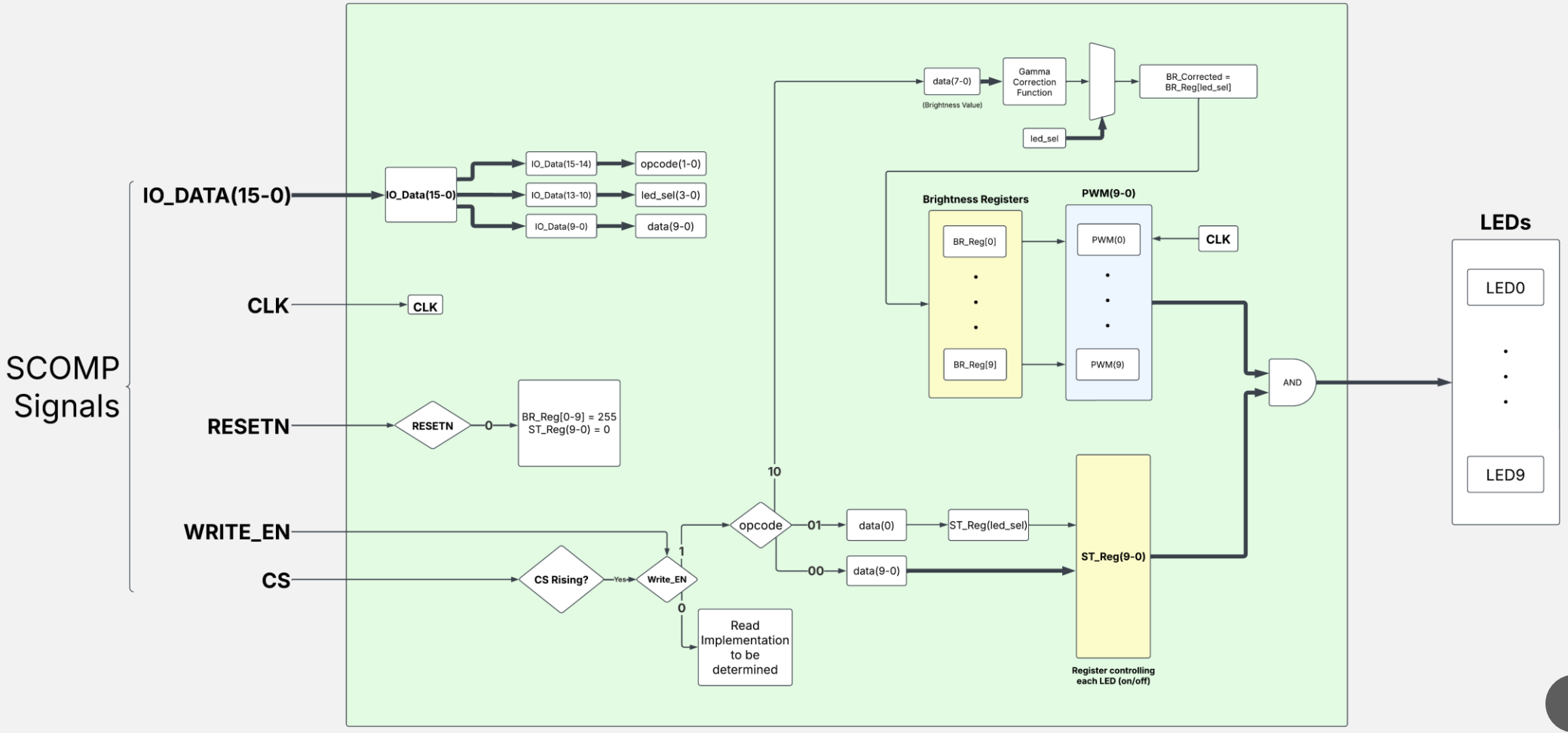
There are 11 Registers we use internally:

- One 10 bit "State Register" which determines if an LED is "on" (1) or "off" (0) with each bit corresponding to 1 LED
- There are 10 Brightness Registers that are 8 bits each that hold the brightness value of each LED
  - This value is fed into a PWM generator to create a square wave of the correct duty cycle to create the correct "brightness"

Registers	Bit-Width	Description	Default State
ST_REG	10-bit	Show active LEDs	00000000000
BR_REG[0 – 9]	8-bit	Display Brightness Level	11111111

IMPLEMENTATION

LED Controller

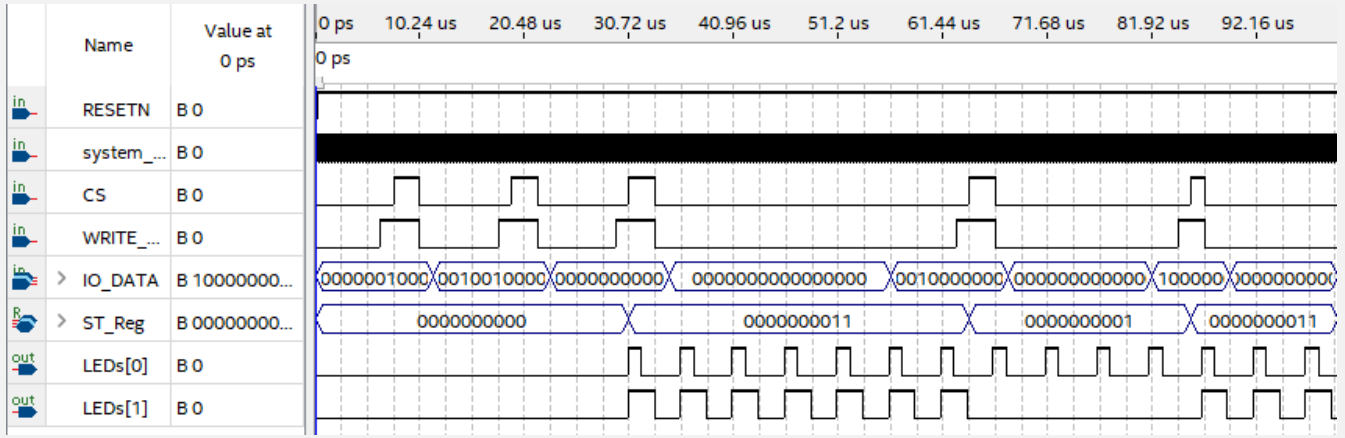


TESTING AND DEBUGGING - SIMULATION

We test our peripheral both in simulation and on the physical device

In our functional simulation we isolate just the peripheral and simulate the inputs SCOMP would send to validate:

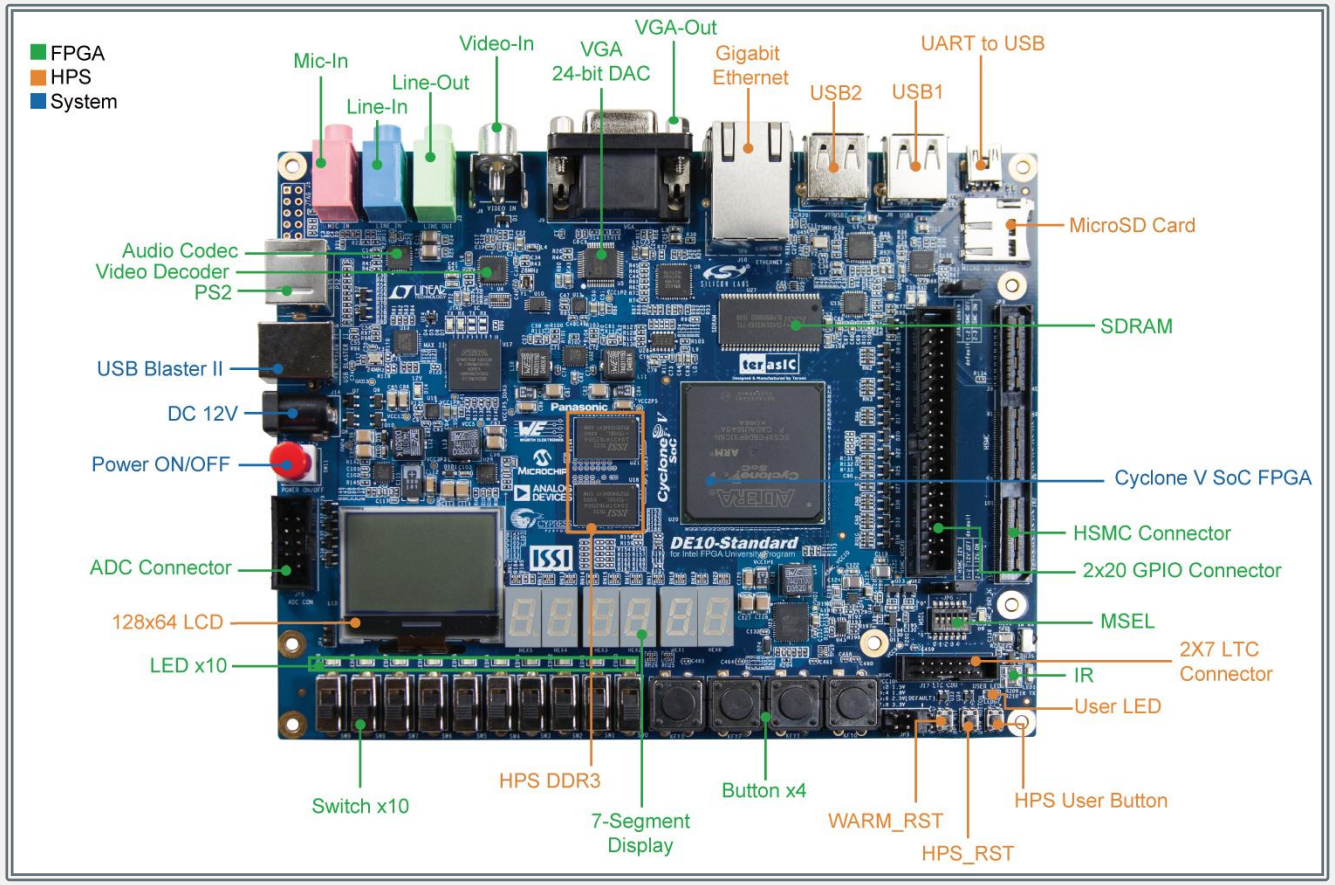
- Registers have correct behavior as we test a variety of different inputs
- PWM signal gives correct duty cycle
- State Register passes / blocks signals as intended
- Reset works as intended



PHYSICAL TESTING AND DEBUGGING

To physically test our peripheral, we:

- Write a simple SCASM test program
- Upload VHDL to the DE-10
- Fix issues not seen in Functional Simulation:
  - PWM clock being at correct rate to correctly see differences in LED brightness
  - Interface with SCOMP working as intended
- Will use digital logic probes and oscilloscope to debug if issues arise



PROJECT STATUS UPDATE

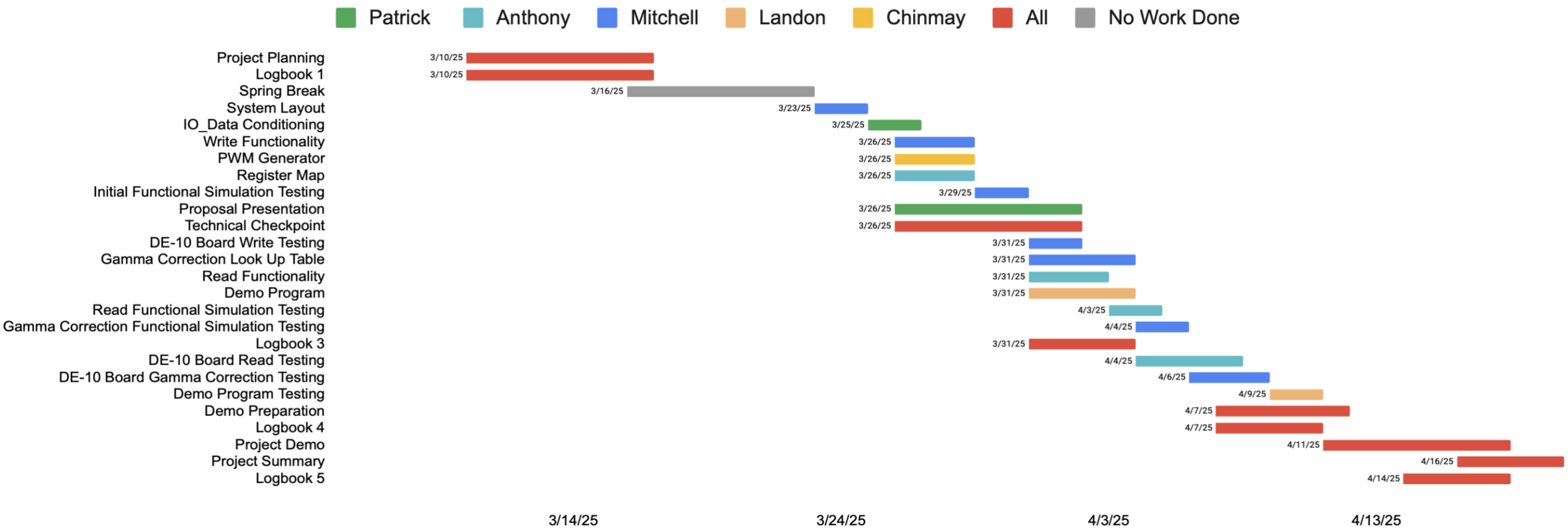
- Completed the VHDL for a basic LED control peripheral that has 3 functionalities:
  - Setting all LEDs using a bitmask - Tested: Works ✔
  - Setting the brightness of a specific LED - Tested: Works ✔
    - PWM – Tested: Works ✔
  - Turning on a specific LED without affecting others - Tested: Works ✔
  - At this point we have tested using functional simulation and know our device:
    - Correctly decodes a simulated SCOMP inputs
    - Outputs the correct duty cycle to the LEDs
- Testing on physical DE-10 Board required to confirm:
  - LED brightness changes as we expect
  - Interface with SCOMP works properly

- Need to implement:
  - Gamma Correction
  - Read Functionality

DEMO PLAN

- Current Demonstration Plan:
  - Loading Bar Looping Animation (SCOMP)
  - Shows LED select (setting all LEDs at once to reset)
  - Shows Specific LED toggles
  - Shows brightness selection

MANAGEMENT PLAN / TIMELINE



FINAL REMARKS

- For this project, we were tasked with creating an intuitive and useful LED interface peripheral for SCOMP.
- We have proposed and are currently working on a prototype that satisfies most of the needs of our customer.
- As we stand, we have a peripheral that has working core functionality and still working to implement even more features (read, gamma correction, etc.)
- Although we haven't fulfilled all the deliverables for the project yet, we are confident that we are still on schedule to deliver a fully functional peripheral that fulfills all the requirements given.