EE/CprE/SE 4920 WEEKLY REPORT 2

# Digital ASIC Fabrication

9/6/24 – 9/19/24

**Group Number:** SDDec24-12

**Faculty Advisor & Client:** Henry Duwe

**Team Members:** Mitchell Driscoll, Evan Dunn, Baoshan Liang, Katie Wolf

## Summary

**Week 1**
During week one of the Fall semester, we met with Duwe on Wednesday to discuss our progress from the previous semester of senior design. This included where each member left off and the future plans for the project. We began integrating each module/component we designed in the previous semester into the project wrapper and continued with the hardening of the User Project Wrapper. We also have been working to harden the OpenRAM wishbone interface individually, and now we are working to integrate that into the User project wrapper. Finally, we set a time to meet every Thursday to work as a group and ensure we are not overlapping work.

**Week 2**

- https://git-pages.ece.iastate.edu/isu-chip-fab/hardware/master/7617015b2c405c48ff6d0367fd9bdd85e15f191e/CaravelPmod/CaravelPmod-sch.pdf
- https://github.com/efabless/caravel/blob/main/verilog/dv/caravel/defs.h

## Past Week Accomplishments

- **Mitchell:** I began the integration of submodules into the user project wrapper and began making connections within the wrapper. Wrote the verilog for 32to1mux. I began going through tutorials for implementing C code for rtl simulations in our design. We discussed the IO configurations for our main project wrapper.
- **Evan:** Established a working openRAM base from the tutorial, however, I've run into issues porting this to another top-level user_project_wrapper. Perhaps I will start over, building an openRAM project from the ground up and eliminating this issue. Duplicating effort inadvertently: All-but-hardened the user_project_wrapper with OpenRAM, ran into the congestion issues, and stopped. Fixed certain issues in the component modules RE: Syntax, and finally at the last minute, was working to understand and hopefully implement C code to get our modules running on the hardware.
- **Baoshan:** In order to integrate the Verilog codes of each part, I studied the read and write code of the Wishbone bus using C to control the input and output of the GPIO pins. I also conducted a simple test to verify its input and output and learned the address information of the GPIO pins.
- **Katie:** After successfully hardening OpenRAM, I attempted to run precheck on a wrapper with OpenRAM but faced many errors. Per Jake and Gregory's suggestion, I

paused working on OpenRAM to focus on the framework. I implemented the major connections in the user_project_wrapper, minus OpenRAM, and included two adder projects to stand in for user projects.

## Pending Issues

- **Mitchell:** Continue working to complete rtl for testing of modules and components.
- **Evan:**
- **Baoshan:** After RTL checking, more components are integrated.
- **Katie:** OpenRAM is on hold until talking with Efabless.

## Individual Contributions

| Name | Contributions | Weekly Hrs | Total Hrs |
|---|---|---|---|
| Mitchell | Began integration of sub-modules into user project wrapper, wrote verilog code for 32to1mux | 6 | 15 |
| Evan | Similar contributions to Katie, Mitchell. Updated the master branch with a OpenRAM-compatible setup. | | |
| Baoshan | Instantiate the Wishbone and GPIO modules. Use the Wishbone Master to control the input and output registers of GPIO。 | 5 | |
| Katie | Implement framework in user_project_wrapper | 5 | |

## Plans for the Upcoming Week

| Action Item | Person in Charge | Expected Date |
|---|---|---|
| Write C code and testbenches to test framework | Katie and Mitchell (and others, add your name here) | 9/25 |
| | | |
| | | |
| | | |
| | | |

## Advisor Meeting Summary

**Week 1**

- The deposit has been paid for the November 11th deadline; this gives us access to tech support from ebfabless/caravel.
- Jake - recommended pushing it further, pushing the task of two simple adder projects through pre-check;
- Don't want to spend too much time on OpenRAM and fail to finish… two options for the future of OpenRAM are to set a deadline and, if not met by then, go to tech support, rapidly go through the rest of the design, and get through pre-check
- Can I write to individual components and continue to build off that… Dont need to be thorough but should be able to show that the modules are working.
- Take tests and be more aggressive, testing edge cases, weird combinations, → trying to break components to ensure we get full coverage.
- Pre-harden example adder from the tutorial and drop that into user project example
- Component testing should be done in C

**Week 2**

- Prioritize getting OpenRAM functionality working over the number of modules we can implement. The focus is on solidifying the ‘process’.

- Incorporate at least *two* previous projects into the design, ideally *four*. Tapeout is priority.

- //toggle GPIO 7 – reg_mprj_io_7 = GPIO_MGMT_OUT;
- Toggle= reg_mprj_datal &= ~(1 << 7)