**Overview**

During the Spring 2023 Semester, I began my senior design project at Iowa State University. Senior Design is a two-semester course, with a focus on research and planning in the first semester, and then design and implementation in the second semester. The project that I chose to join with three other group members was Digital ASIC Fabrication, which involves designing and fabricating a digital ASIC. Our client and advisor are Dr. Duwe, a professor at Iowa State who researches and teaches about computer architecture and cyber security topics. Dr. Duwe has had two previous senior design groups follow the same process and design cycle as we are, each staggered one semester apart. The first senior design team was able to successfully get their chip approved and fabricated, and we are currently waiting to receive it. For our project, our requirements are to design a digital ASIC chip using Verilog and open-source tools, alongside developing firmware and a bring-up plan to assist a future team in verifying the functional requirements of our design.

We are able to create our digital ASIC by working through the company eFabless, which is sponsored by Google. The goal of eFabless is to use open-source tools that are freely available to silicon prove both digital and analog ASICs for hobbyists, researchers, and students, who do not have a large pool of resources to take advantage of. Being a senior design team, this was an excellent opportunity to learn more about writing Verilog, using new open-source tools, and designing a chip for fabrication that would come back on an SOC development board. All of the submitted designs are required to be open-source, with the benefit of receiving 5 ASICs that are placed on a development board. eFabless produces their silicon wafers through the U.S. based foundry SkyWater, which takes advantage of 130 nm technology and a standard cell library to fabricate our project. Due to this, we are limited to this fabrication process and the standard cell library that is provided by SkyWater in their 130 nm product development kit. Each project submission is produced on a wafer that houses open-source designs from multiple different projects, enabling eFabless to produce small amounts of ASICs for each team at a reduced cost. While working with open-source tools and a limited production cycle has been difficult through eFabless, they are in a very unique and rare position by offering free digital ASIC fabrication, that is returned on an integrated development SOC board for us to test.

As of now, the first semester of senior design has been completed. During this time, we were able to determine our design plan with doctor Duwe, research more about the tools we will be using, and begin testing submodules in our design. Our team decided on a modular design with multiple different functions, including a clock gating module, digital signals processing filter, an external SPI interface, and more. The intent of choosing a modular design was to push the capabilities of the fabrication process at the SkyWater foundry, and see how many working pieces we could receive on our digital ASIC through working with eFabless.

Useful Links:

* Senior Design Website: <http://sddec23-06.sd.ece.iastate.edu/>
* Design Document: <http://sddec2306.sd.ece.iastate.edu/reports/Design_Doc_Rev1_1.pdf>
* Design Presentation: <http://sddec2306.sd.ece.iastate.edu/reports/Final_Presentation.pdf>
* eFabless: <https://efabless.com/>
* Other projects on eFabless: <https://platform.efabless.com/projects/project_definition/1>
* Toolflow Documentation: <https://caravel-user-project.readthedocs.io/en/latest/#quickstart>

**Semester One**

During the first semester of senior design, our team focused on ironing out our digital ASIC design, learning how to use all of the required open-source tools, and beginning implementation and testing on submodules of our design. Throughout this process, we were able to learn more about eFabless and what requirements and constraints we had to follow for our design, which helped motivate some of our design decisions.

We decided to implement a modular design which included the following modules all in the user area of our digital ASIC:

* Clock Gating
* Standard Cell Test
* Custom Cell Test
* Wishbone Test
* DSP Road Noise Audio Filter
* Backdoor SPI Interface

The clock gating, standard cell, custom cell, and wishbone tests were included in our design so that we could verify the fabrication of our chip with more concrete and basic tests. For example, if the rest of the fabrication failed, we should ideally be able to still probe the Wishbone communication bus with the Wishbone Test module to ensure that something is working in our chip. We also can use the Standard Cell Test to verify that one basic SkyWater 130 nm standard cell from the provided product development kit library is fabricated correctly. This cell will be a basic combinational circuit that will be easy to verify functionality of. Other tests, such as the Backdoor SPI Interface, are used in case the Wishbone bus fails. In case it does, we have included a separate SPI interface so we can use our own communication bus that we designed to program the configuration of each module, alongside reading data that is output from each addressable module. Of these modules, I am responsible for leading the Backdoor SPI module, alongside assisting with documentation and system level integration throughout the project. I was very excited to work on the Backdoor SPI module, since it meant that I would have to work with interfacing between each module to configure and receive data from them with a standardized serial communication protocol.

Before we could begin on designing the Verilog and testbenches for each module, we had to first install and configure our open-source tool flow. eFabless provides a GitHub repository that can be cloned, which includes make files and the 130 nm SkyWater standard cell library. Alongside this, the repository also includes an MPW shuttle precheck pass, which will be used next semester for our final acceptance testing, so that we will have a verified submission for the project’s fabrication. The main open-source tools that we are using in our project include OpenROAD0, KLayout, GTKWave, and Magic. OpenROAD is used for our RTL synthesis, and takes our Verilog designs and the SkyWater Product Development Kit to develop GDSII layout files of our design. With this layout, we are able to view our project with KLayout, verify our hardened gate level designs, and submit our project to eFabless by instantiating our project in a user wrapper module in the provided repository. GTKWave is used to verify our waveforms from Verilog testbenches and C firmware simulations at both the register transfer level and gate level. We can instantiate our modules either on their own or through the caravel wrapper, depending on the stage of verification and how robust we need at the time. The open-source tool Magic will be used for our Custom Cell module, as we can use it to layout the traces for the combinational cell we will design. After this cell is laid out in Magic, we can instantiate it within the user wrapper module. These tools have been difficult to setup and use initially, but it has been a great learning experience to try a new tool flow and new constraints, all for an open-source project.

After getting situated with our tools, I began to design the Backdoor SPI module with the help of my teammates. We decided to use shift registers to serially shift in data from an external microcontroller and shift out data from our user area modules. Alongside this, since we were using a clock provided from an external microcontroller, and the clock in the user area were different, we used 2 D Flip Flops to act as a buffer to ensure that there would be no metastability errors in our data flow. During this semester, I began by designing the shift registers in Verilog, and derived testbenches for them similar to my previous FPGA projects and MIPS processor. We used error flags and an overall test status bit to indicate when errors would occur, to help automate testing and assist us in our future bring-up plan. During this time, I also worked on developing a 10-page documentation sheet, including block diagrams on clock synchronization, module implementation, submodule design, and waveform results from my shift register tests. This documentation will act as a baseline for the Verilog design I will work on over the Summer for the top-level Backdoor SPI module.

Near the end of the semester, we finished the first iteration of our design document, which encompasses our design plan, testing plan, and much more context to our project. During the week before finals, we presented our project to a faculty panel of three professors at Iowa State University. This was an excellent opportunity to summarize our project in a concise 15-minute presentation, and both of these documents can be accessed on the Overview tab of my website, or our senior design website, which is also linked.