Engineering Educators Bringing the World Together
2025 ASEE Annual Conference & Exposition
Palais des congrès de Montréal, Montréal, QC • June 22–25, 2025
ASEE
Paper ID #47365

Design and Organization of a Bring-Up First Vertically-Integrated Chip Design and Fabrication Co-Curricular

Gregory Theodore Ling, Iowa State University of Science and Technology

Gregory Ling is a recent graduate from Iowa State University with a Master's degree in Computer Engineering with a focus in Computing and Networking Systems, now working at IBM. He was the primary student lead creating the ChipForge co-curricular on which this paper is based.

Jackson Hafele, Iowa State University of Science and Technology

Jackson Hafele is currently working in industry as a logic design engineer at ForwardEdge ASIC in St. Paul, Minnesota. He completed his B.S. and M.S. in electrical and computer engineering at Iowa State University, with a focus on Computing and Networking Systems in his graduate program.

Dr. Nicholas D. Fila, Iowa State University of Science and Technology

Nicholas D. Fila is an assistant teaching professor in the Department of Electrical and Computer Engineering at Iowa State University. He earned a B.S. in Electrical Engineering and a M.S. in Electrical and Computer Engineering from the University of Illinois-Urbana-Champaign and a Ph.D. in Engineering Education from Purdue University. His research interests include empathy, ethics, design thinking, and course design.

Dr. Henry Duwe, Iowa State University of Science and Technology

Henry Duwe received his B.S. degree in computer engineering and computer science from the University of Wisconsin–Madison and his M.S. and Ph.D. in electrical and computer engineering (ECE) from the University of Illinois at Urbana-Champaign (UIUC). He is an Associate Professor in the Electrical and Computer Engineering (ECpE) Department at Iowa State University. His research interests include computer architecture, compilers, design automation, and engineering education. He is focusing on the architecture and design of dependable and intelligent energy-harvesting computer systems. He instigated and advises the Chip Forge chip fabrication co-curricular.

Design and Organization of a Bring-up First
Vertically-integrated Chip Design and Fabrication Co-curricular

Abstract

The advent of lower-cost application-specific integrated circuit (ASIC) fabrication through multi-project wafer submissions and open-source synthesis tooling, such as those provided by Efabless, brings a novel opportunity for a vertically-integrated student-led co-curricular based on the design and fabrication of custom ASICs. Based on a bring-up first methodology that emphasizes the hands-on experience of testing and validating physical hardware early in the learning journey, ChipForge encourages students to extend the concepts learned in their existing coursework to practical applications in physical projects. It is the culmination of years of technical and organizational work through two master's students, many senior design teams, a group of freshman honors students, and the first members of the co-curricular. With this co-curricular as a foundation, students integrate information from separate engineering design courses–from digital logic to embedded firmware to microarchitecture development–in each cohesive output product.

This paper presents the technical and organizational infrastructure from which ChipForge was formed as well as results gathered from participating students. Results of this project are presented to evaluate the effectiveness of the co-curricular for improving not only the hard skills of ASIC development but also the soft skills of leadership and group collaboration. By analyzing available evidence, we explore the effects of the documentation, training approach, and co-curricular environment of a vertically-integrated, bring-up first methodology on students' career readiness.

Introduction

Students studying computer engineering at Iowa State University are taught a series of skill sets through multiple courses related to Application Specific Integrated Circuit (ASIC) design. However, hands-on ASIC design experience is generally unachievable in a single quarter or semester course due to the fabrication latency. This study centers around a co-curricular designed to bridge this gap in hands-on experience by encouraging all years of students, from freshman to graduate, to work together on ASIC design, implementation, and bring-up across their academic journey. By spanning multiple years, ChipForge can provide a full round-trip experience to students who join in their freshman or sophomore years. In addition, as students work across class boundaries, freshman students can draw from senior- or graduate-level expertise and resources while the more experienced students gain valuable opportunities to consolidate their knowledge and develop leadership skills.

Background

ASIC Design Flow

The ASIC design flow generally follows a linear progression of six broad steps:

- Design the overall ASIC, the high-level components, their functional requirements, and the required data flow
- Implement the components in a hardware design language such as Verilog (for digital projects) or an analog design program (for analog components)
- Create and run component-level tests to ensure correct functionality
- Integrate the components, then create and run system-level tests
- Verify constraints and high-level verification
- Submit the design for fabrication, also referred to as "tapeout"
- Bring up the design, including any associated physical hardware required to interface with the ASIC, and run validation tests on the fabricated ASIC

Of this flow, most steps are performed by the students in the co-curricular, with the exception of the fabrication itself, which Efabless performs.

Multi Project Wafer (MPW)

Since 1981, the concept of packing multiple independent projects on a single wafer has been used to reduce development costs of Very-large-scale Integration (VLSI) design [1]. The mask is by far the most costly part of the VLSI fabrication process. Creating an entire mask for a single project is cost-prohibitive for a low-quantity development run. By packing multiple low-quantity projects into a single mask, the mask cost can be shared across all the combined developers, so each developer gets their design fabricated at a fraction of the cost of fabricating the ASIC on their own. This is the model this co-curricular uses, as we only intend to use a small number of ASICs per fabrication run for bring-up testing and future development.

Efabless OpenMPW Shuttles

Efabless defines itself as "the first creator platform for chips" [2]. They are a company that performs MPW shuttle runs for submitted designs. Using funding from Google, Efabless provided free MPW shuttle runs for any design that was open source and approved by Efabless for fabrication from 2020 [3] to 2022 [4].

ChipForge was initially bootstrapped by a series of senior design teams that created designs for Efabless fabrication as part of these OpenMPW shuttle programs. Of the designs, one was successfully submitted in December 2022, fabricated by Efabless, and returned to us in March

2024, providing this co-curricular's first hardware ASIC. These open-source fabrication shuttles were the original motivation to begin designing this co-curricular as a student organization, providing years of experience to develop the specific tooling infrastructure setup and methodology described in this paper. As part of their fabrication runs, Efabless also provides a wrapper template that includes a RISC-V microcontroller with an exposed memory bus, general-purpose input/output (GPIO) pin interface, and a core management area to interact with the user's design. This template allowed the senior design projects to begin writing design code quickly without worrying about the specifics of the fabrication technology node or becoming overwhelmed by the complex tooling abstracted by the template project.

Efabless ChipIgnite Shuttles

As useful as the OpenMPW fabrication was to starting ChipForge, Efabless has not provided OpenMPW shuttle submissions since MPW-8 in 2022 [4]. However, Efabless still offers commercial options, which cost around $10,000 for a single design. These use the same infrastructure and template as the OpenMPW shuttles but do not require the design files to be open source. With recent NSF funding, this option has allowed this co-curricular to continue fabricating designs.

A Computer Engineering Curriculum

A generic computer engineering curriculum such as the one this co-curricular intends to work alongside contains several core courses summarized below.

- An introductory digital logic course taught in students' late freshman or early sophomore years that introduces students to digital circuits using basic logic gates such as AND, OR, and NOT to construct more complex primitives such as D-flip flops and more complex circuits such as shift registers and state machines.
- An introductory embedded firmware course taught in the sophomore year that teaches students how to use Memory-Mapped Input-Output (MMIO) peripherals using C firmware on an embedded microcontroller to complete various tasks and the importance of reading the datasheet for understanding the various functions of a device.
- A more advanced processor design course in students' late sophomore to early junior years that guides students in creating a small MIPS processor using a hardware design language and introduces students to more complex topics such as different memory architectures, caching, and pipelining.

These three are the core courses most relevant to digital ASIC design, taught early in a student's academic journey, which complement their first experiences interacting with and designing an ASIC in this co-curricular.

Motivation

Student Interest

Due to the significant infrastructure cost of ASIC development, students interested in the field are left unable to learn and grow in an important area of computer engineering at the university level. Courses individually teach some aspect of ASIC development, but students fail to see the overall picture of how the independent concepts taught in these courses work together in the process of designing, fabricating, testing, and finally making use of a custom ASIC. ChipForge, as a student organization outside the time and cost limitations of a normal course, provides a means for students to get that overall picture. Rather than the limited time frame of three months or less in which a course must focus on a single aspect of the development flow, a co-curricular can provide a longer-term experience in which students can engage directly with the entire ASIC development flow.

Course Cohesion

Unencumbered by course-related time constraints, ChipForge can improve cohesion between the existing courses. As students work through introductory classes on digital logic, they can work through designing small components of a larger processor (some examples being a Serial Peripheral Interface (SPI) bus, an Arithmetic Logical Unit (ALU), or an integer multiply unit). As they proceed through introductory classes on embedded systems firmware development, they can write embedded firmware tests of their standalone components via an existing processing core and memory bus. As they go through more advanced classes on processor design, they can work on more detailed work on full processor or more complex accelerator layouts, then work on designing the high-level architecture for the next ASIC to be developed. Rather than independent courses that allow students only a brief glimpse into one of these aspects of development in an independent project, this co-curricular enables students to work through the entire development cycle on a long-term project that enhances their understanding of their coursework by applying their learning on one cohesive ASIC.

Vertical Integration

This co-curricular is vertically-integrated by design. It enhances the cohesion between courses for an individual student, and the design encourages more experienced members to help newer members through their learning on the project. To design a single ASIC, the new members working on the small components have to work with the more experienced members designing the overall architecture of the project to make sure their contribution will work in the larger scheme of the design. In coursework, students become used to working only with people at their own skill level. They rarely have to interact with more- or less-experienced students to accomplish a shared objective, so often members become very used to working with others at their skill level. ChipForge provides an environment where the experienced members are encouraged to work with newer members, accurately defining the specification of their design's criteria in ways they can implement and teaching them when necessary. As all levels of this process are within the scope of this co-curricular, newer members gain access to the expertise gained by existing members, and existing members are pushed to learn leadership and teaching skills to interact with the newer members.

Internship Opportunities

Students are constantly searching for ways to stand out on internship applications, and ChipForge provides them such an opportunity. The courses taken are common to most applicants, but the practical application of those skills is not. This co-curricular provides one such practical application their learning on a project most students are unable to attempt on their own due to the high cost of fabrication. Experiencing the entire ASIC design flow empowers students to decide which aspects of the design process they most want to explore internships in, as well as making them more well-rounded candidates wherever they apply. As part of the vertically integrated structure, this co-curricular also creates more networking and mentorship opportunities between experienced members who have completed internships and new members who may be looking for their first one.

Workforce Development

As with seeking internships, few people first enter the workforce having experience developing an ASIC from start to finish. ChipForge provides the funding (through grants and sponsorships), the technical infrastructure, and the experience of existing members to students desiring a competitive advantage in the job market. New hires who have already worked on a full cycle of ASIC development require less training and can contribute meaningfully to a team more quickly, lowering the overhead cost of a new hire. This co-curricular allows students to have three to four years of practical ASIC design experience before graduation–time in which they are learning ASIC development while their coursework is fresh in their minds. Students in ChipForge also gain experience with common tooling such as Python, Git, Verilog, Make, C, Tcl, and KiCad, which can also streamline onboarding of new engineers. The soft skills of collaboration and leadership developed through this co-curricular's emphasis on community and mentorship further contribute to students' future careers.

Related Work

Co-Curricular Impacts

The effects of co-curriculars on leadership have been studied in several papers. An analysis of student skill development in extra-curriculars and co-curriculars showed project teams, such as this co-curricular, exhibit more distinct soft skills than any other category explored [5]. A second supporting paper found "evidence for the view widely held by engineering faculty and administrators that co-curricular activities provide a reasonable venue for leadership skill development (Cox et al., 2010; Knight & Novoselich, 2014) and are consistent with the broader literature on undergraduate leadership development (Dugan & Komives, 2007)" in spite of also claiming that "co-curriculum is an inefficient method of developing leadership" [6]. Two other papers also note the positive association between involvement in co-curricular activities and employability of graduates in engineering [7], [8].

Vertical Integration

This method of vertically integrating students has been implemented in several other scenarios including the Purdue Vertically Integrated Projects program [9], in which freshman through senior students work on group projects to encourage inter-year collaboration. The University of Evansville also implemented a similar program in their Vertically Integrated Design sequence [10].

Debugging Mindset

The debugging mindset encourages students to reconsider their mental models as their understanding of the problems in front of them evolve and to continuously experiment with the resources they have [11], [12], [13], [14]. To this end, ChipForge is designed to encourage discovery and interaction during the development process, and provide an environment for students to ask questions, either to their peers, or to the hardware itself via experimentation.

ASIC Design Flow

The ASIC design flow as a whole has been the topic of numerous research papers, including [15] and [16]. Research in [17] explored a fully-automated ASIC design flow akin to the design flow used by Efabless in the Caravel Harness, and [18] investigated improvements to the design flow for larger and more complex projects. The most relevant is OpenLANE [19], the open-source synthesis suite used by the Caravel Harness and by extension ChipForge, with a more detailed description of its development by [20]. The OpenLANE process has been used in several research endeavors as well, including [21], [22], and [23]. A comparison of varoius open-source ASIC development flows in [24] show that they "are capable to produce physical layouts for realistic examples."

Research Objective

Based on the prior motivation and related work, this paper explores the efficacy of a co-curricular designed to encourage ASIC development in a bring-up first, vertically integrated environment with an emphasis on developing transferable skills such as leadership and collaboration. The following describe the specifically defined approaches identified as primary foundations, the implementation of these approaches, and an evaluation of the co-curricular's current impact on students.

Approach

Bring-up First

This co-curricular is bring-up first in that new members' first exposure to the co-curricular will be through bring-up of existing ASIC designs. For a new member with minimal ASIC development experience, bring-up gives a highly interactive platform to learn about how an ASIC functions. To this end, most of the co-curricular's infrastructure improvements have been designed to enhance the bring-up experience for new members. This stage is critical, as new member retention is a common issue among student organizations. ChipForge captures students' interest by immediately engaging them in an interactive environment where their changes have readily apparent, real-world effects. This also provides a minimal barrier to entry for new members, as designing complex ASICs is a complex task requiring high-level knowledge of the entire system and every component involved, from firmware to high-level architecture. Bring-up relies on embedded systems experience, which is taught as early as students' second year. Even without that prior knowledge, students can learn as they go via interactive tutorials and physical experimentation with real hardware that gives real-time feedback.

Vertical Integration

Vertical integration is another core component of how this co-curricular is designed, based on [10]. The different levels of the design hierarchy all work together to design and test a single ASIC. The high-level architecture, logic design, design verification, and bring-up are not restricted to their own silos but instead are worked on collaboratively. Having a single co-curricular that contains all four major components of ASIC development together allows students to ask questions and learn in a much more collaborative environment. The architects learn to take into account the bring-up difficulty of their designs, and the bring-up team members can directly question the more experienced members, who probably contributed to bring-up at some point in the past. Through vertical integration, the co-curricular retains cohesion among all aspects of the design process and should have a greater chance of persisting through leadership changes as students graduate and new students join.

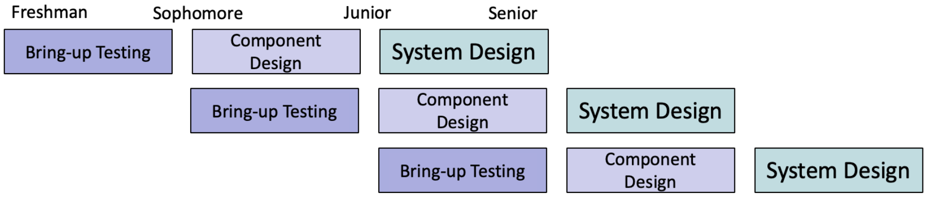

Figure 1: Design Schedule

As bring-up is the most discovery-oriented segment of the project, students joining in their freshman or sophomore years focus on the bring-up of the prior students' design as shown in Figure 1. At a single point in time, a new design may be in progress, being designed by late junior through senior students and primarily implemented by late sophomore through junior students, while the prior design recently returned from fabrication is being validated and analyzed by the incoming freshman through sophomore students. This provides the incoming students time to learn the basics of ASIC design through experimentation on a prior ASIC before entering the high-pressure and time-critical portion of meeting a tapeout deadline for fabrication. This also addresses the natural turnover of new members in student organizations; new members who may just be exploring their opportunities are not given time-critical tasks, but the pressure of those items are instead reserved for the more experienced members.

Coursework Consistency

Coursework consistency is another major factor influencing the design of this co-curricular. ChipForge is not meant to replace the coursework systems already in place, but instead work with and beyond them to bring greater cohesion to the learning that already occurs. To this end, existing coursework is referenced when possible throughout the documentation and tutorials, and leaders in the organization keep the members' coursework experience in mind when explaining concepts. As this organization is primarily composed of computer engineering students, leaders in the organization have already taken the common introductory classes previously mentioned, which allows new members to ask questions about course experiences, improving engagement in those classes as they see what can be done with the knowledge they impart.

Debugging Mindset

The debugging mindset is one of the major underlying principles taught by the computer engineering coursework as a whole. This mindset is characterized by a willingness to delve deeper into a subject. When a project fails to work as expected, students are taught to seek to understand why it failed and to learn from the mistake, turning the failure into a learning opportunity. With a debugging mindset, students try different solutions to discover the answer to the problem on their own, but also ask for help from peers, TAs, or professors. ChipForge encourages students to seek out the documentation, their mentor, or other club leadership when there is a problem they cannot address on their own. It also encourages students to be flexible with their mental models, willing to adapt to new information as they discover it through hands-on work. This relates to the bring-up first model, which allows students to discover what an ASIC is through testing and exploration before attempting to design one themselves.

Interactivity

This debugging mindset cannot occur without interactive feedback from both human and nonhuman sources. Documentation is a major source of students' learning, but learning from purely written documentation alone becomes stale and does not allow the student to ask questions or discover for themselves what happens if they try something. To support the "what happens if" mentality, the tutorials and documentation are designed for interactivity, encouraging students to apply the principles described and see what happens. The tooling supports feedback mechanisms allowing students to see the results of their tests. There are several forms: physical interactivity such as a light-emitting diode (LED) turning on or being able to press a button, interpreted physical results such as the Universal Asynchronous Receiver-Transmitter (UART) terminal sending and receiving characters, lower-level analysis such as using Saleae [25] logic analyzers used to debug GPIO functionality, single-step debugging via firmware debug probes, and simulation analysis (either textual or graphical in a waveform viewer such as GTKWave [26]). Through the use of a field-programmable gate array (FPGA), all of these forms of interactivity are available at all levels of the design flow, allowing members to interact in the way best suited to their task.

Furthermore, interactivity is key in any student organization's longevity. Students' time is a highly-valued resource, and with 781 student organizations to choose from at Iowa State

University, organizations compete for students' attention. For this co-curricular to persist, it must sustain student engagement to have a body of leadership that can answer questions and keep the co-curricular functioning at a high level without burning out.

Agile Workflow

As a small student organization, ChipForge has the ability to be extremely flexible in how progress is made. Using an agile workflow where students can claim their own tasks to complete and work at a more flexible pace motivates students to maintain interest in the project as their course workload fluctuates.

The centralized project tracking process on our GitLab instance allows students to contribute their own ideas to the pool of future projects. Weekly workdays encourage regular progress updates as students have a designated time to gather and work together as a community, keeping each other accountable to their own deadlines and staying flexible around coursework commitments.

Accessibility

Accessibility for new students is another goal of this co-curricular. Students generally have a limited budgets, so this co-curricular does not currently charge dues andwill attempt to avoid onerous dues in the future. However, it is not a company and, therefore, has no profits of its own to finance its operations. ChipForge currently has an NSF grant to fund six paid fabrications, but it will have to sustain itself by more grants, donations, and/or corporate sponsorships in the future. To minimize expenses, the tooling infrastructure is designed to be usable at no cost to a student, using university-provided resources or free public software for ASIC development. The Efabless infrastructure works well in this scenario because the entire toolflow is open-source and free to use.

The other major component of accessibility is ease of access. While some forms of interactivity can only easily occur in the physical lab space (e.g. pressing a button or using a logic analyzer to debug a design), most components of interactivity are designed to be accessible remotely as well. This is achieved through a combination of the Virtual Desktop Infrastructure (VDI) already provided by the university, supplying a consistent remote computing platform to all computer engineering students, a webcam setup over the development boards to provide remote access to view the physical boards, secure shell (SSH) access to the lab computer connected to the development boards, and more. This allows students to be flexible when working on projects, as they are not limited by the constraints of a physical lab.

Cohesion

Cohesion has been mentioned several times already in the context of coursework, but the goal of cohesion in this co-curricular goes beyond coursework and extends throughout the design philosophy. Research in [27] explores the necessity of designing cohesion into a course during its conception and reinforcing it as the course progresses. ChipForge models its emphasis of cohesion after this idea, striving for cohesion both internally (documentation, tutorials, and designs are mutually consistent) and externally (references to coursework stay current as courses evolve). The documentation for all levels of the design flow is in a single documentation website [28], the tooling runs on top of a consistent RedHat Enterprise Linux 8 (RHEL8) environment provided by the university (accessible in person or remotely via VDI), and the same project repository structure is used for the entire design flow. This supports tighter vertical integration between all layers in the design flow; while different teams may need different software systems to perform their task, the backing infrastructure is the same.

Ease of Maintenance

Ease of maintenance is another driver of infrastructure development. For instance, this co-curricular uses Continuous Integration and Continuous Delivery (CI/CD) in GitLab and simple scripts to perform updates to infrastructure tools. This shifts a significant portion of the burden of infrastructure maintenance from the students leading the club to scripts in GitLab. Students have limited time to spend on student organizations, so reducing the maintenance burden on the students enables them to focus on the primary goal of producing future designs for fabrication.

This also extends to the documentation itself; maintaining one single source for all documentation in a single website hosted GitLab Pages minimizes obstacles to updating that documentation. As all infrastructure systems are documented in the same place, there is less burden on the incoming leadership to remember how to perform infrastructure maintenance tasks or locate disorganized documentation.

Institutional Adoption and Faculty Involvement

One of the key advantages of our approach is that it primarily leverages existing institutional resources, from computing infrastructure to technical services facilities and faculty. For example, the open-source build system can run on any departmental lab computer or remotely accessible virtual machines. With this setup, only a single workstation is needed to host remote access to the shared physical development boards and FPGAs. Custom PCBs designed by co-curricular participants designed to interface with fabricated ASIC designs have been submitted and assembled in our department's technical services shop facility. Direct course involvement has mainly focused on awareness of the co-curricular, accomplished by both short student presentations in lectures and faculty advisor guest lectures in early core courses.

Although the co-curricular is student-led by design, faculty support has been essential in initiating, mentoring, and sustaining the co-curricular. Given the scope of expertise needed to design full ASICs successfully, the students have reached out to many faculty and senior graduate students to get advice, direction, and resources (e.g. analog VLSI faculty to aid with PLL and ADC designs).

The initial vision and basic structure were developed by the faculty advisor and pitched as a series of senior design projects that recruited key student leaders, built institutional knowledge and infrastructure, and resulted in the first fabricated ASIC necessary for starting the co-curricular. The faculty advisor has also played an important role in sustaining the co-curricular by securing multi-year funding for the fabrication of ASICs and PCB boards for bring-up, ensuring institutional knowledge is retained in written documentation, and facilitating interactions between senior design teams and industry mentors.

Bootstrapping

This co-curricular started as a series of senior design projects. Senior design is a year-long required course for seniors working on assorted group projects led by a faculty advisor for a student, faculty, or commercial client. Beginning in 2022, several senior design projects were led by a faculty advisor to design ASICs for fabrication while researching Efabless' resources and its submission process.

This first team used an open-source Secure Hashing Algorithm 1 (SHA-1) hashing algorithm [29] and implemented a small wrapper around it to work with the Caravel Harness[30]. They submitted it for fabrication in December 2022, and Efabless returned the fabricated ASIC to us in March 2024. This is so far the only ASIC returned from fabrication and was used as the first bring-up ASIC to start the co-curricular.

Subsequent senior design teams built off prior teams' success to create

- a digital neuron for a spiking neural network which encountered size limitations and issues with implementing additional Random Access Memory (RAM) [31],
- a modular design testing multiple aspects of the synthesis flow, including adding additional RAM, creating a custom SPI controller, using the provided Wishbone Bus, and implementing a custom digital cell layout in Magic to instantiate in Verilog[32],
- an analog design experimenting with creating a small Resistive RAM (ReRAM) implementation, including a transimpedance amplifier, a digital-analog converter, and an analog-digital converter[33],
- and a framework for incorporating many small projects into a single larger project to enable fabrication of multiple small designs into a single MPW submission, as it was noticed that none of the prior designs efficiently maximized the area available in a single MPW submission[34].

Together, these provided a foundation of project design knowledge that can pass the design and submission checks and several example projects to draw from. As the first ASIC returned in March 2024, the second phase of bootstrapping this co-curricular began. A group of high-achieving freshman students joined to learn about ASIC bring-up. As Efabless provided a development board, these students primarily focused on learning the basics of digital logic and embedded firmware development to interact with the built-in RISC-V microcontroller. They all had minimal digital design and C firmware expertise prior to starting with the project. Led by two graduate students, these freshman students were able to successfully acquire a basic understanding, both of understanding a digital logic design in Verilog, and interacting with that design using C firmware on the RISC-V embedded microcontroller. They then ran several verification tests against the SHA-1 ASIC, which passed. This provided some preliminary evidence that new members could work on bring-up related projects on fabricated ASIC designs with guidance from more experienced members.

Creating the co-curricular

In August of 2024, ChipForge became a registered student organization and began advertising for new members. The two leaders were graduate students who completed the modular senior design project described above, but one only participated virtually so the leadership team was originally very short-staffed in-person. With the help of several interested undergraduate students, the co-curricular began designing its first ASIC. The co-curricular met twice a week for roughly two-hour workdays. Members were asked to attend one workday a week, with the alternate available for scheduling conflicts. During this workday, students worked on their tasks in close proximity: either running through a set of bring-up tutorials to introduce new members to ASIC testing, designing a new component for the upcoming fabrication, working to integrate components together, or providing assistance to various students as necessary.

One important organizational decision was to assign mentors. New members were assigned an older student as a mentor who could be their point person for questions and feedback, to ensure everyone has someone they know they can ask. Upon arrival, new members were introduced to their mentor and began working through the first bring-up tutorial on making an LED blink on the SHA-1 ASICs. Most new members were able to complete that first bring-up interaction within the first workday. From there, subsequent tutorials taught more information about interacting with various components of the Caravel Harness and user designs. Once a student finished the tutorials, they were given the opportunity to pick a first project from a list of projects or choose their own if they had an idea of something they wanted to try.

Infrastructure

Several infrastructure developments were critical in starting the co-curricular, documented in [35]. The primary infrastructure resource was the toolchain, coupling all the required open-source tools including GCC, OpenLANE, Podman, Magic, XSchem, and many more into a single downloadable tarball that students could extract and use on any of the university-provided RedHat Enterprise Linux 8 instances, either physical or remote. This utilized existing machine infrastructure and added an easy-to-use collection of utilities required for Efabless ASIC design on top without requiring administrator access for each user or individual machine. The toolchain also provided a consistent set of tools to minimize problems caused by tool variation.

Another critical infrastructure development was the ability to use an FPGA. Students wanted to work with real hardware and see physical interactions, and the Efabless Caravel Harness was not originally designed to run on an FPGA. To introduce more interactivity in debugging physical devices, a set of adaptation scripts was developed to enable designs to be tested on an FPGA with minimal and consistent modification. In parallel, a seven-segment display module and external Static RAM (SRAM) module were created to provide a target for new students to test writing simple Verilog designs. Together, the FPGA and interactive hardware modules provided incoming students a hands-on experience to test their first, simple Verilog implementations on physical hardware with a short design iteration time.

As the quantity of hardware resources was limited, a webcam was installed, pointed at the various hardware systems including two FPGAs and two SHA-1 ASIC development boards, connected to a computer accessible via SSH by members. This enabled students to work on the hardware remotely and time share usage of the limited hardware efficiently.

Evaluation

Freshman Students

The first results produced were the final projects from the five freshman students mentioned previously, working with ChipForge as a part of an introduction to research program. These freshman students were tasked with completing several final projects by the end of the semester, primarily validation of the SHA-1 hashing ASIC from sddec22-17 (see Table 1 below), in addition to interacting with the seven-segment display and external SRAM modules.

The two students working on the SHA-1 hashing ASIC were able to research how SHA-1 works, explore the example design-verification test code written by the first senior design team, adapt it to running on physical hardware, and then run the test program. They concluded that the ASIC worked as designed in the test cases they ran.

Another pair of students worked on the seven-segment display module, interacting with the physical shift registers driving the seven-segment displays via bit-banged SPI. They used the management core of the SHA-1 hashing ASICs to interface with the seven-segment displays from firmware. They successfully displayed hex digits as a precursor to writing a full Verilog driver for the seven-segment displays.

The remaining student began interfacing with the external SPI SRAM module. Unfortunately, due to the lack of external SPI peripherals in the SHA-1 ASIC, that too had to be bit-banged via the management core RISC-V processor using direct manipulation of the GPIO pins. This student was able to read an identifier register from the SRAM to ensure communication succeeded.

Together, these three final projects demonstrate that bring-up can be in the realm of a second-semester computer engineering freshman’s ability with guidance from more experienced members.

Senior Design Progress

The senior design teams have public progress reports, weekly during the first semester working on the project and biweekly during the second semester on the project. Significant milestones are tabulated from among the most recent seven digital design teams’ progress reports in Table 1, ordered chronologically. Milestones are denoted by semester and week from beginning the senior design project, and teams are named based on the semester they complete (so sddec22-17 started in the spring of 2022 and graduated in December of 2022). Dashes denote incomplete milestones, and question marks denote milestones not yet achieved or recorded for teams still in progress.

The double horizontal line marks when the co-curricular infrastructure became ready for senior design teams to use. Prior to team sddec24-12, the FPGA infrastructure did not exist, the prebuilt toolchain wasn’t available, and the documented tutorials specifically created for new members were also not created yet. Also, teams sdmay23-28, sddec23-06, and sdmay24-21 were not

Table 1: Senior Design Progress Report Data

Team #	RTL Sim Example	Hardened Example	RTL Sim Custom	Hardened Custom	Tested on FPGA	GL Sim Custom	Passed Precheck	Passed Tapeout
sddec22-17 [36]	–	–	S1 W6	S2 W8	–	S2 W12	–	–
sdmay23-28 [37]	S1 W3	S1 W5	S1 W3	–	–	–	–	–
sddec23-06 [38]	S1 W2	S1 W2	S1 W4	S2 W1	–	–	S2 W6	–
sdmay24-21 [39]	–	–	–	–	–	–	–	–
sddec24-12 [40]	S1 W4	S1 W2	S1 W3	S1 W4	S2 W11	S1 W4	S2 W11	S2 W11
sdmay25-27 [41]	S1 W2	?	?	?	S1 W2	?	?	S1 W11
sdmay25-28 [42]	–	S1 W2	S1 W4	S1 W3	?	?	S1 W7	S1 W11

expected to tapeout their chips during senior design, but instead provided a tapeout-ready design that could be submitted for fabrication at a later time as there was no nearby submission they could target.

Team sddec22-17 did not record running an example Register-Transfer Level (RTL) simulation or hardening on an example project; the first recorded successful hardening was in the eighth week of their second semester. They also did not record a successful hardening and precheck of their design until after the design was submitted to Efabless for fabrication.

Team sdmay23-28 did not record when they finished hardening their neuron, but their presentation shows an image of the final hardened design, so it was hardened at some point. Their final biweekly progress report states that they were working on hardening it.

Team sddec23-06 was the first team to successfully pass precheck with a final design, with one minor automated Design Rule Check (DRC) failure due to the use of OpenRAM that was acknowledged by eFabless as expected when using the provided OpenRAM components in their tooling the team had minimal control over.

Team sdmay24-21 did not record successful completion of any of these tasks.

Team sddec24-12 was the first team to successfully complete the entire digital flow from RTL simulation to tapeout, including bring-up tests on an FPGA and submitting their design. There was one DRC error during precheck, which Efabless described as expected and requested that they address it after submission.

Team sdmay25-27 has not published all their reports as of writing (denoted with question marks), but they also contributed a final design that was hardened, passed precheck, and passed tapeout checks to the first co-curricular tapeout submission described below.

Team sdmay25-28 did not record running an RTL sim on an example project, but have run RTL simulations and hardening on custom projects. They also passed precheck on their own, and contributed a design to the first co-curricular tapeout submission.

Of important note is that neither team sdmay25-27 nor sdmay25-28 had this tapeout as an express goal of their senior design project, but submitted designs to help them gain familiarity with the tools. Furthermore, both teams submitted a custom design within the first semester of their project. In contrast, the senior design teams prior to the introduction of the co-curricular infrastructure struggled to harden a custom design at all in the first semester. This shows a level of correlation between ChipForge infrastructure and improved productivity in the senior design teams.

First Co-curricular Tapeout

On November 11th, 2024, the first fabrication submission as a registered student organization was sent to Efabless, to be returned in early 2025. This first co-curricular tapeout contained many small, independent projects, combined together by the frame created by a senior design team to multiplex all inputs and outputs between the small projects. These projects are tabulated below.

Table 2: First Co-curricular Tapeout Project Descriptions

Years of Students Involved	**Description**
Freshman	A pulse-width modulation (PWM) controller designed by a freshman student as their first project; they wrote the Verilog and bring-up firmware in parallel using the FPGA, driving an external piezo buzzer. It supported sending low-resolution sine waves or simpler square waves.
Sophomore & Junior	A seven-segment display controller designed by a sophomore and a junior working together, interfacing with a pair of external seven-segment displays connected via a shift register, controllable via the RISC-V microcontroller.
Junior	A SPI SRAM controller designed by a junior to interface with an external SRAM Integrated Circuit (IC) as SRAM consumes too much space to instantiate in high quantity in the ASIC itself.
Senior	A second seven-segment display controller designed independently by one of the current senior design teams to practice designing a simple interface.
Senior	A small portion of a final project taken from a MIPS processor design course by another current senior design team as a first test of submitting a design.
Senior & Graduate	A variable precision multiply-accumulate unit designed by a senior and a graduate student as a test of a more complex and resource-intensive design in the framework.
Graduate	A simple addition module designed by a graduate student that received two 32-bit inputs via either the Wishbone memory bus or a set of internal IO pins to act as a simple validation of the frame and ASIC as a whole.

This design was a collaborative effort of about 15 people working on the various projects, across all years of learning as a clear example of this co-curricular at work.

Of note is the freshman's PWM controller, as he designed that module with only loose guidance from more experienced students. The original intent was not to introduce Verilog design in a member's first semester, but it appeared that the Verilog and firmware co-design approach he and others used worked well, writing the C firmware to interact with the design and the design itself in parallel, using fast iterations on FPGA or in RTL simulation.

Workday Attendance

One more source of available results is workday attendance. Workdays are structured twice a week for scheduling purposes, with students expected to attend one of the two workdays. Figure 2 shows the attendance on these workdays.

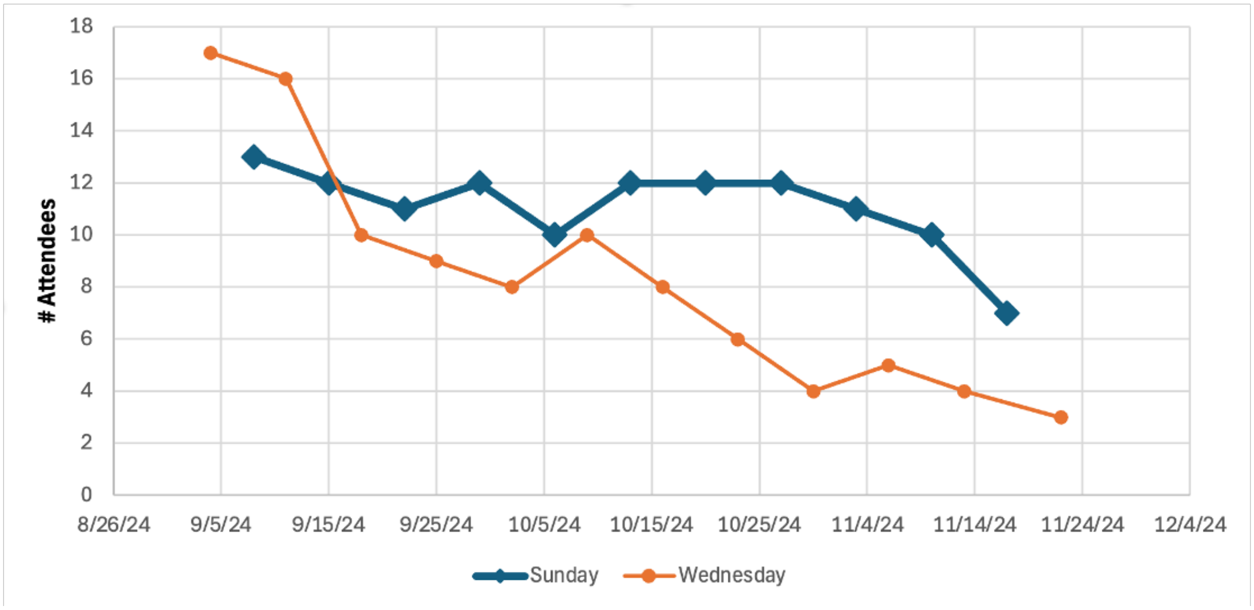

Figure 2: Workday Attendance

The Wednesday workdays appear to have steadily declined from the first workday. The cause is unclear, but there was limited leadership bandwidth this semester as there was only one in-person student leader who had experience with the ASIC flow at the beginning of the semester. This caused the newly-formed leadership team to struggle to keep track of student progress as well as they would have liked, necessary to ensure students are provided engaging projects in a timely manner.

In contrast, Sunday workdays remained relatively constant at around 11 people each workday. As November 11th was the tapeout deadline for the last ASIC, several students took a break on the last Sunday workday following that tapeout, leading to lower attendance on that day than the average.

The greater decline in Wednesday attendance could be attributed to some students transferring from Wednesday to Sunday workdays, or that weekday meetings were harder for students to attend as the semester's workload increased. Another possible factor is that Wednesday workdays were advertised to the public when the club was first being advertised at the beginning of the semester, but the Sunday workdays were started to accommodate scheduling conflicts for interested members, possibly leading to a natural separation where Sunday workdays attracted more students with greater dedication to this co-curricular.

Discussion

The following maps the four primary categories of intended outcomes to the sources of evidence (mostly anecdotal) as seen in the co-curricular.

1. **Transversal Skills: Leadership**

   (a) **New-Member Onboarding Tutorials**
   The onboarding structure, with written tutorials guided by more experienced members, requires a greater amount of leadership by more experienced members. Onboarding a new member through these tutorials is a hands-on process and requires attention to maintain student interest and engagement.

   (b) **Documentation**
   Documentation is rarely a sought-after task in any coursework or co-curricular; it takes dedication and a concern for the future to encourage students to keep documentation updated. Several students exhibited this leadership by submitting updates to the documentation.

   (c) **Common Resources**
   The focused effort on keeping resources in common locations has allowed at least one student to explore the backing infrastructure on their own to make improvements to the Python Unified CLI. They were able to find the source code to submit an update, exhibiting a significant sense of shared ownership of the co-curricular and leadership initiative in maintaining infrastructure without instruction.

2. **Transversal Skills: Debugging Mindset**

   (a) **New-Member Onboarding Tutorials**
   Several students enjoyed the extra challenges listed at the bottom of each tutorial. The challenges, such as using the LED to display a message in morse code or using the UART to create a program that responds to a specific password, encouraged students to spend effort understanding what the tutorial was teaching and to investigate its application in a novel context. One such novel context is the idea of using morse code to display an error message when only an LED is available.

   (b) **Custom Hardware Devices with FPGA**
   Students used the custom hardware devices a significant amount, particularly the seven-segment displays as evidenced by the two seven-segment controllers independently written by two separate groups of students. The student who wrote the PWM controller enjoyed seeing the successes (and failures) of their work as it

progressed through development on the FPGA using a simple piezo buzzer attached to the FPGA. The existence of the FPGA helped give students the ability to perform “trial and error,” experimenting with their designs where they could see results.

    (c) **Logic Analyzer**
    The logic analyzer was also well-used, especially for the SPI SRAM controller, PWM controller, and seven-segment display controllers. It allowed an iterative debug process that let growing students unfamiliar with ASIC design understand how their signals and logic affected the physical hardware.

3. **ASIC Bring-up Experience**

    (a) **Freshman Final Projects**
    The greatest source of evidence for members gaining ASIC bring-up experience is the freshmen students who were able to accomplish the final projects listed above from little to no background knowledge.

    (b) **Digital & Bring-Up Co-Design**
    This was an unintentional yet interesting result that came from the freshman who wrote the PWM controller. They went from the bring-up tutorials to co-designing the PWM controller hardware design with firmware side-by-side in a very iterative process. Instead of considering bring-up and digital design as separate components, they designed both in tandem, iteratively testing the combination on an FPGA.

    (c) **First Co-curricular Tapeout**
    The first co-curricular tapeout also provides some evidence of increased bring-up experience as most of the components on the tapeout were tested prior to fabrication on the FPGA with validation tests (some automated unit-tests that can be run unattended, others physical verification via logic analyzer). Only two of the sub-projects had merely simulated verification results prior to fabrication.

4. **ASIC Design Experience**

    (a) **Senior Design Progress Reports**
    The most significant evidence for this is the improvement in time to first custom hardening in the senior design progress reports, along with the fact that all three senior design teams since this co-curricular began have submitted ASIC designs, long before prior teams even hardened their first custom design.

    (b) **First Co-curricular Tapeout**
    The first co-curricular tapeout reveals significant ASIC design experience in the group, none of whom had ever submitted an ASIC for fabrication before, specifically in the quantity of separate projects and in the quantity of contributing students from all classifications.

    (c) **Future Complexity**
    Having seen the success of this first submission, multiple students are interested in tackling a much more complex project, designing a very simple GPU, for the next tapeout as part of this co-curricular. This is a very ambitious project, revealing a

significant confidence in their abilities in the ASIC development flow.

Conclusion

Even in these very early stages of the co-curricular, there is evidence that the vertically-integrated, bring-up first co-curricular supports the transversal skills of leadership and the debugging mindset. The successful first co-curricular tapeout and the considerable increase in productivity in the senior design teams also reveal a significant effect on students' technical knowledge of designing ASICs through practical experience. Long-term effects will continue to be examined with interest.

Acknowledgments

This work was supported in part by the U.S. National Science Foundation under Grant 2144757. The authors want to thank the members of senior design teams who developed the initial institutional knowledge, processes, and test chips that have made ChipForge possible.

References

[1] T. M. Service, "The mosis service," https://themosisservice.com/about, accessed: 2024-12-02.

[2] eFabless, "Company," https://efabless.com/company, accessed: 2024-12-02.

[3] ——, "Home," http://web.archive.org/web/20201111215953/https://efabless.com/, accessed: 2024-12-02.

[4] ——, "Shuttle mpw-8," https://platform.efabless.com/shuttles/MPW-8, accessed: 2024-12-02.

[5] D. R. Fisher, "Educating engineers for the 21st century : a framework for skill development through co-curricular and extracurricular involvement," 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:112008165

[6] D. B. Knight and B. J. Novoselich, "Curricular and co-curricular influences on undergraduate engineering student leadership," *Journal of Engineering Education*, vol. 106, no. 1, pp. 44–70, 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/jee.20153

[7] S. Kovalchuk, M. Ghali, M. Klassen, D. Reeve, and R. Sacks, "Transitioning from university to employment in engineering: The role of curricular and co-curricular activities." American Society for Engineering Education, 2017.

[8] E. E. Stiwne and T. Jungert, "Engineering students' experiences of transition from study to work," *Journal of Education and Work*, vol. 23, pp. 417 – 437, 2010. [Online]. Available: https://api.semanticscholar.org/CorpusID:146467802

[9] C. B. Zoltowski and E. J. Delp, "Vertically integrated projects (vip) program at purdue university: A research experience for undergraduate students," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*, 2021, pp. 1141–1145.

[10] D. W. Stamps, "A vertically integrated design sequence," *IEEE Transactions on Education*, vol. 29, no. 6, pp. 1580–1590, 2013.

[11] D. H. O'Dell, "The debugging mindset: Understanding the psychology of learning strategies leads to effective problem-solving skills." *Queue*, vol. 15, no. 1, p. 71–90, feb 2017. [Online]. Available: https://doi.org/10.1145/3055301.3068754

[12] H. Duwe, D. T. Rover, P. H. Jones, N. D. Fila, and M. Mina, "Defining and supporting a debugging mindset in computer engineering courses," in *2022 IEEE Frontiers in Education Conference (FIE)*, 2022, pp. 1–9.

[13] L. Morales-Navarro, D. A. Fields, and Y. B. Kafai, "Growing mindsets: Debugging by design to promote students' growth mindset practices in computer science class." *Proceedings of the 15th International Conference of the Learning Sciences - ICLS 2021.* [Online]. Available: https://par.nsf.gov/biblio/10309425

[14] H. Mehraban and J. Hu, "How to teach debugging? the next million-dollar question in microelectronics education." [Online]. Available: https://par.nsf.gov/biblio/10523315

[15] A. Shetty, "Asic design flow and methodology–an overview," *International Journal of Electrical and Electronics Engineering*, vol. 6, no. 1, pp. 1–5, 2019.

[16] M. Bombana, P. Cavalloro, S. Conigliaro, R. B. Hughes, G. Musgrave, and G. Zaza, "Design-flow and synthesis for asics: a case study," in *Proceedings of the 32nd annual ACM/IEEE Design Automation Conference*, 1995, pp. 292–297.

[17] T. Brandon, B. Cockburn, and D. Elliott, "Hdl2gds: a fully automated asic digital design flow," in *Canadian Conference on Electrical and Computer Engineering, 2005.*, 2005, pp. 1535–1538.

[18] V. Sahula, C. Ravikumar, and D. Nagchoudhuri, "Improvement of asic design processes," in *Proceedings of ASP-DAC/VLSI Design 2002. 7th Asia and South Pacific Design Automation Conference and 15h International Conference on VLSI Design*, 2002, pp. 105–110.

[19] A. Ghazy and M. Shalan, "Openlane: The open-source digital asic implementation flow," in *Proc. Workshop on Open-Source EDA Technol.(WOSET)*, 2020.

[20] M. Shalan and T. Edwards, "Building openlane: A 130nm openroad-based tapeout-proven flow," in *Proceedings of the 39th International Conference on Computer-Aided Design*, 2020, pp. 1–6.

[21] L. D. Franck, G. A. Ginja, J. P. Carmo, J. A. Afonso, and M. Luppe, "Custom asic design for sha-256 using open-source tools," *Computers*, vol. 13, no. 1, 2024. [Online]. Available: https://www.mdpi.com/2073-431X/13/1/9

[22] R. T. Edwards, M. Shalan, and M. Kassem, "Real silicon using open-source eda," *IEEE Design & Test*, vol. 38, no. 2, pp. 38–44, 2021.

[23] I. Rodriguez-Ferrandez, L. Kosmidis, M. Tali, and D. Steenari, "Space shuttle: A test vehicle for the reliability of the skywater 130nm pdk for future space processors," in *2023 IEEE 29th International Symposium on On-Line Testing and Robust System Design (IOLTS).* IEEE, 2023, pp. 1–3.

[24] M. Chupilko, A. Kamkin, and S. Smolov, "Survey of open-source flows for digital hardware design," in *2021 Ivannikov Memorial Workshop (IVMEM)*, 2021, pp. 11–16.

[25] Saleae, "Saleae logic analyzers," https://www.saleae.com, accessed: 2024-12-02.

[26] GTKWave, "Welcome to gtkwave," https://gtkwave.sourceforge.net, accessed: 2024-12-02.

[27] N. Bahr and M. Lloyd, "Course cohesion: An elusive goal for tertiary education." *Journal of Learning Design*, vol. 4, no. 4, pp. 21–30, 2011.

[28] J. Hafele, "Developing and documenting the chip design, fabrication, and bring-up process for a vertically-integrated co-curricular," 2024.

[29] J. Strömbergson, "sha1/src/rtl/sha1.v at d6f6cf95b503c171ca363f709aea8e3ce73591fa," https://github.com/secworks/sha1/blob/d6f6cf95b503c171ca363f709aea8e3ce73591fa/src/rtl/sha1.v, accessed: 2024-12-02.

[30] D. Ghauri, C. Mantas, S. Szabo, and C. Violett, "Digital asic fabrication," https://sddec22-17.sd.ece.iastate.edu/docs/Design_Document_492.pdf, accessed: 2024-12-02.

[31] T. Green, K. Gisi, A. Sledge, W. Zogg, and F. Zhu, “Open si asic chip design,” https://sdmay23-28.sd.ece.iastate.edu/docs/Final Design Document 2022.pdf, accessed: 2024-12-02.

[32] J. Hafele, G. Ling, C. Breeding, and W. Galles, “Digital asic fabrication,” https://sddec23-06.sd.ece.iastate.edu/reports/492/Design_Doc_Rev2_1.pdf, accessed: 2024-12-02.

[33] J. Thater, A. Petersen, M. Otters, and R. Dukele, “Reram compute asic fabrication,” https://sddec23-08.sd.ece.iastate.edu/reports/Design_Document_Final.pdf, accessed: 2024-12-02.

[34] M. Driscoll, E. Dunn, B. Liang, and K. Wolf, “Digital asic fabrication,” https://sddec24-12.sd.ece.iastate.edu/docs/Design_Document_V1_2.pdf, accessed: 2024-12-02.

[35] J. Hafele, G. Ling, and H. Duwe, “Chipforge documentation,” Dec. 2024. [Online]. Available: https://doi.org/10.5281/zenodo.14279431

[36] D. Ghauri, C. Mantas, S. Szabo, and C. Violett, “Digital asic fabrication,” https://sddec22-17.sd.ece.iastate.edu, accessed: 2024-12-02.

[37] T. Green, K. Gisi, A. Sledge, W. Zogg, and F. Zhu, “Open si asic chip design,” https://sdmay23-28.sd.ece.iastate.edu, accessed: 2024-12-02.

[38] J. Hafele, G. Ling, C. Breeding, and W. Galles, “Digital asic fabrication,” https://sddec23-06.sd.ece.iastate.edu, accessed: 2024-12-02.

[39] Y. W. Tan, J. Hess, S. Heikens, and H. Khan, “Digital asic fabrication,” https://sdmay24-21.sd.ece.iastate.edu, accessed: 2024-12-02.

[40] M. Driscoll, E. Dunn, B. Liang, and K. Wolf, “Digital asic fabrication,” https://sddec24-12.sd.ece.iastate.edu, accessed: 2024-12-02.

[41] N. Stark, N. Eastburn, N. Thompson, I. Shenouda, E. Kono, and W. Custis, “Open-sourced radio microcontroller for fabrication,” https://sdmay25-27.sd.ece.iastate.edu, accessed: 2024-12-02.

[42] C. Fergen, J. Huaracha, N. Lynch, C. Smith, and L. Wenck, “Digital asic fabrication,” https://sdmay25-28.sd.ece.iastate.edu, accessed: 2024-12-02.