Developing and Documenting the Chip Design, Fabrication, and Bring-up Process for a Vertically-integrated Co-curricular

Jackson Hafele

Masters of Science Creative Component

11/21/2024

IOWA STATE UNIVERSITY

Introduction

What is Chip Forge?

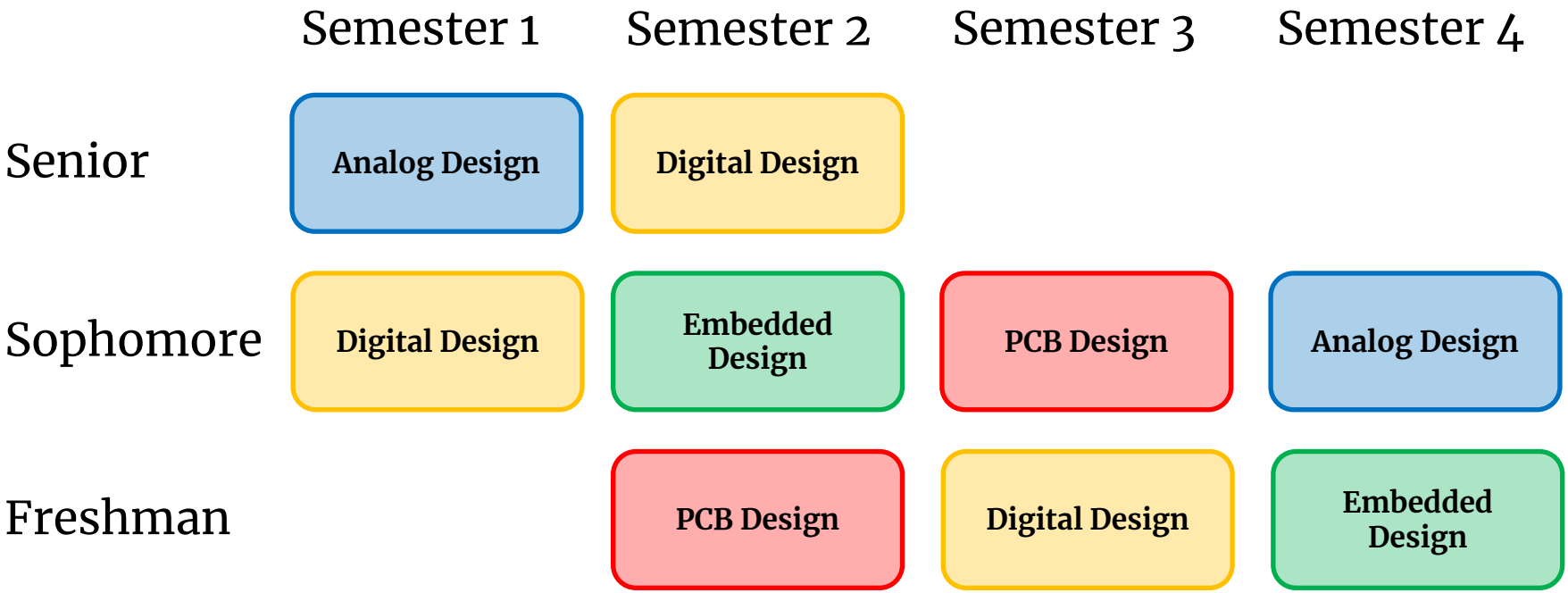
- Iowa State's first ASIC design co-curricular
- Students placed in a vertically integrated environment to learn multiple aspects of ASIC design

Chip Forge Goals

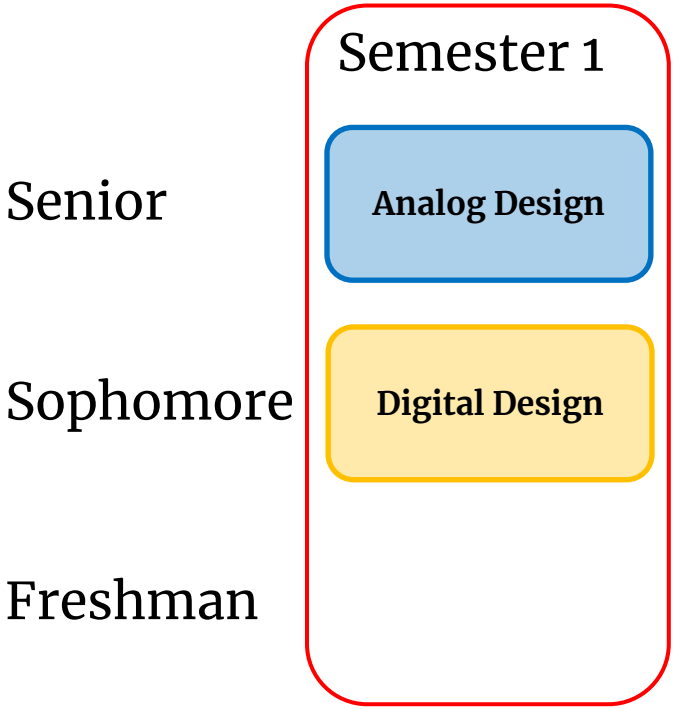
- Design and bring-up a digital/analog ASIC each semester
- Support early and sustained student involvement in ASIC design

Figure 1: Chip Forge Logo

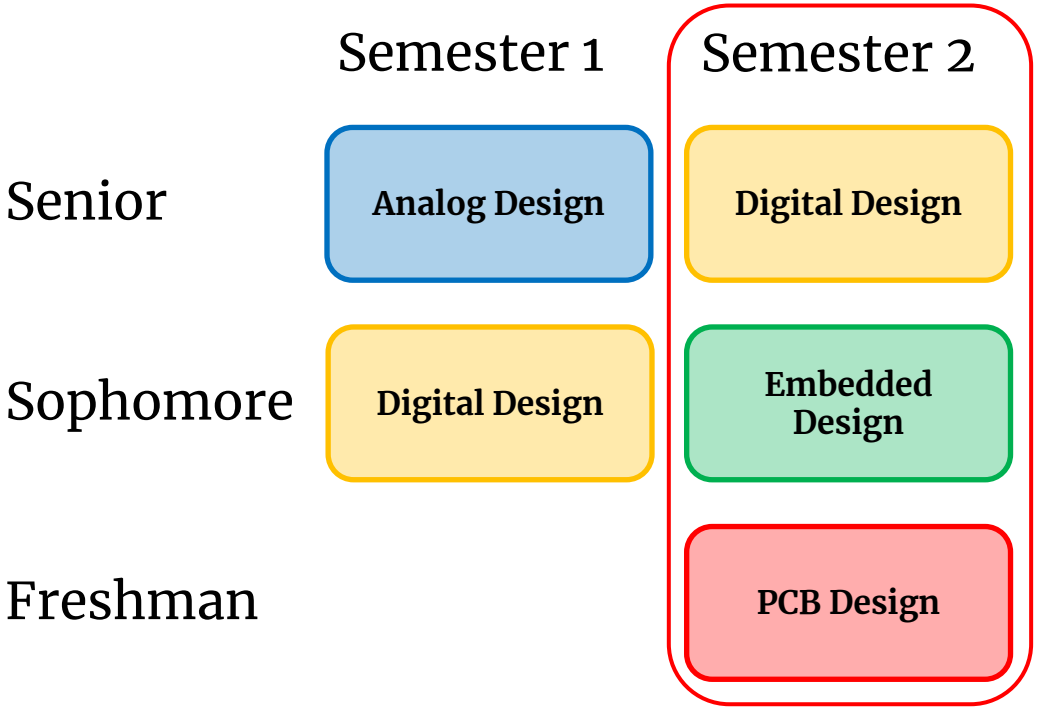
Introduction



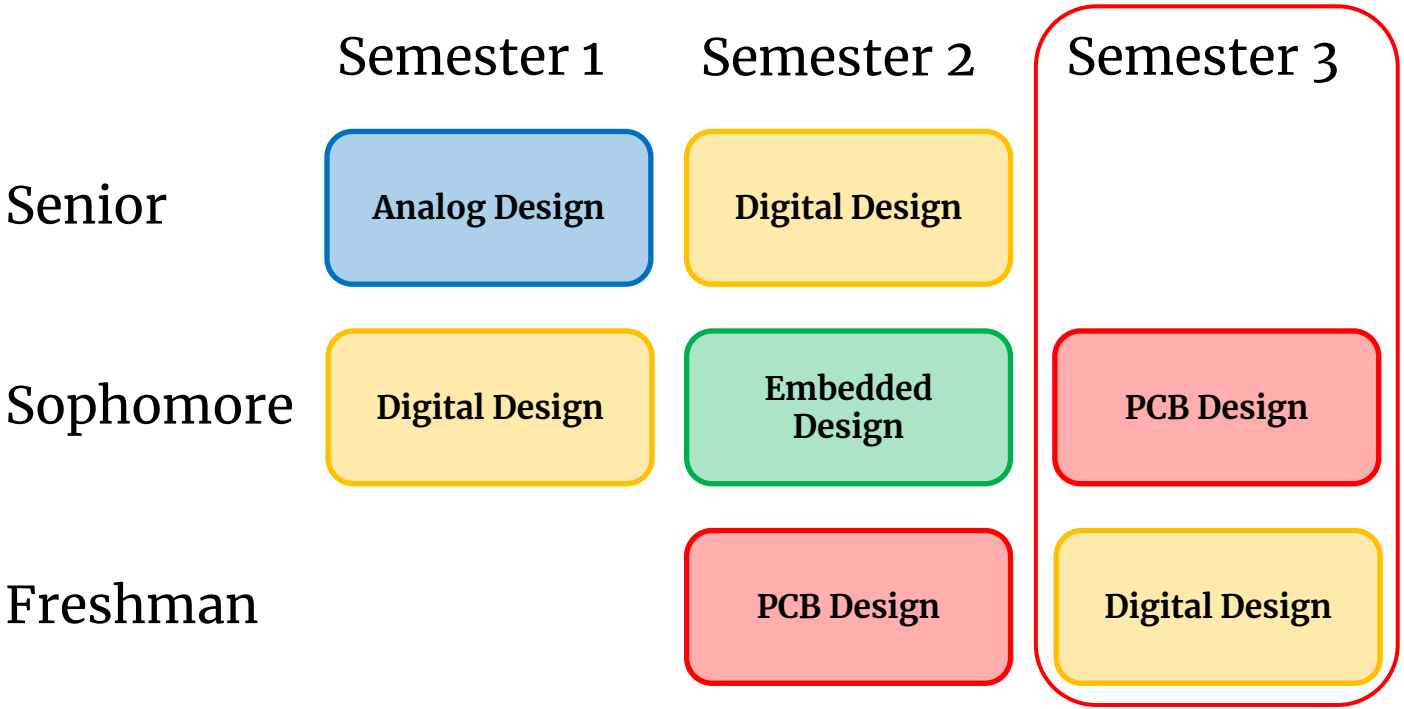
Introduction



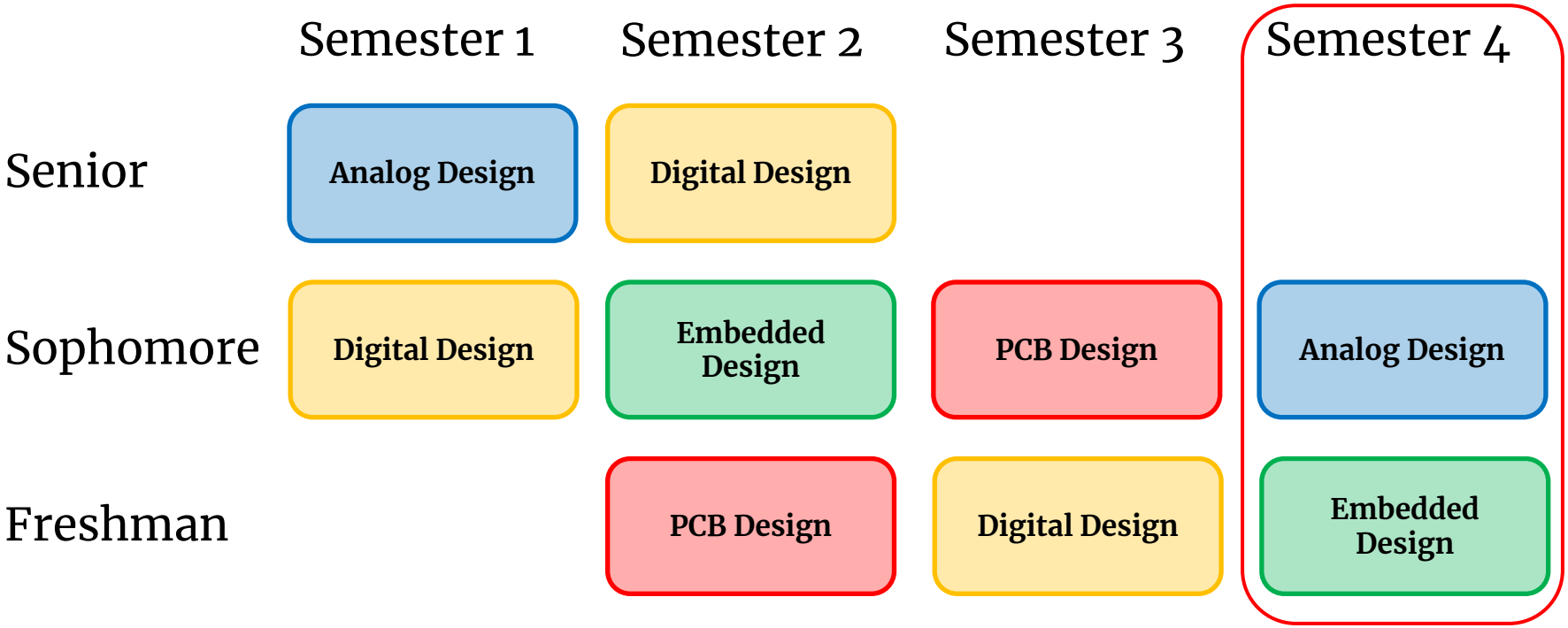
Introduction



Introduction



Introduction



Introduction

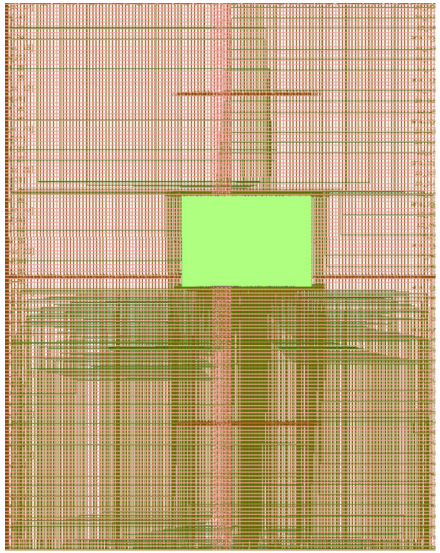


Figure 2: SHA-1 Hasher

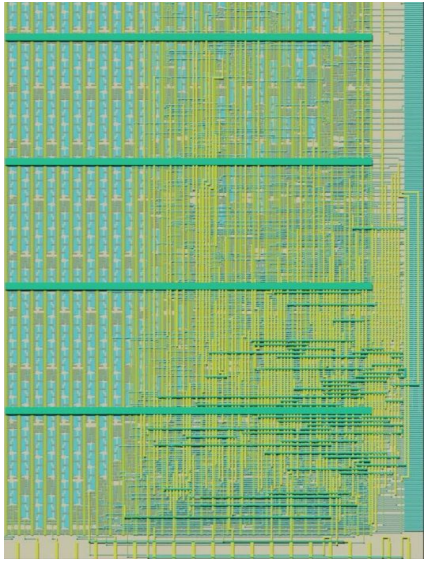


Figure 3: Spiking Neural Network Controller

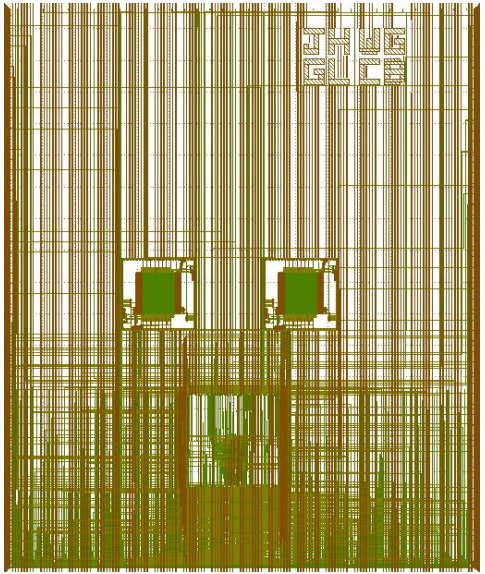


Figure 4: Multi Cell Design

Problem

- Limited resources at ISU to connect end to end ASIC development from initial design to post silicon bring-up

- Tools utilized in Chip Forge are not natively installed or documented at ISU

- The required open source tools used by Chip Forge are challenging to use

Solution

- Create centralized source of documentation outlining both internal and external tools
- Build **Digital** and **Analog** tutorials as examples of functioning ASIC design projects
- Cater to students in context of ISU curriculum and internal Chip Forge information

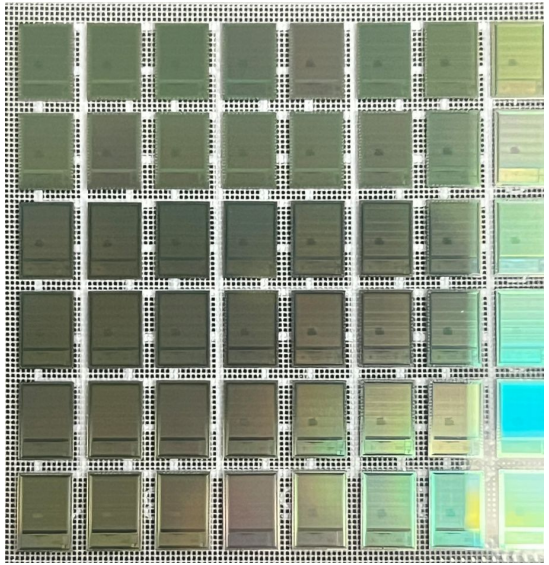


Figure 5: SHA-1 Hasher Die

Related Work - Student Collaboration

Co-curricular impacts

- Student surveys show positive signs of relevant experiences, career opportunities, and leadership for members [1]
- Likely for students to connect experiences in clubs to coursework and industry [2]

Vertically Integrated Curriculum

- Other universities have successfully implemented vertically integrated projects and classes into their programs [3-4]

Motivation - Documentation

What documentation is required?

- Onboarding information for new members
- Usage of required tools with chosen ASIC design solution
- Usage of internally developed tools by Chip Forge

What documentation is helpful?

- Tutorials to demonstrate different aspects of ASIC design
- Reference material for digital/analog/embedded topics
- Coding standards and maintenance guidelines

Motivation - Design Solution

Market Survey

- eFabless Open MPW and chipIgnite programs are most cost effective ($/mm^2)
- Tiny Tapeout utilizes same toolflow, less area for cost
- Other options do not provide tools, would be restricted to ISU ECpE capabilities

Table 1: Multi Project Wafer Cost Comparison [5–7]

Shuttle	Price ($/$mm^2)	Min Area (mm^2)	Quantity	Node (nm)	Foundry
OpenMPW	0	10.8	20	130	SkyWater
ChipIgnite	926	10.8	100	130	SkyWater
180 MS RF G	1,250	5	40	130	TSMC
Tiny Tapeout (Large)	4,167	1.2	5	130	SkyWater
Tiny Tapeout (Medium)	5,000	0.4	3	130	SkyWater
65 MS RF GP	5,800	1	100	65	TSMC
Tiny Tapeout (Small)	6,250	0.08	1	130	SkyWater

Related Work - Design Solution

Open Source ASIC Design

- Multiple different open source tools have been combined to create a full stack ASIC design flow, including OpenLane [8]

Projects utilizing eFabless

- Digital IP for air coupled ultrasonic wave reconstruction [9]
- 4x8 Coarse Grained Reconfigurable Array [10]
- On chip 1 KB SRAM (OpenRAM test chip) [11]

Motivation In Context - eFabless

eFabless Documentation

- Documents digital and analog design repositories with example code [12]
- References for RISC-V management core and design wrapper
- **Downsides**: Missing critical information, dependent on eFabless updates

eFabless tutorials/webinars

- Short videos on how to run through single design step or process [13]
- Longer seminars with guest speakers for in depth topics
- **Downsides**: Videos become stale, challenging to find information needed

Motivation In Context - Coursework

Coursework

- Multiple courses relevant to ASIC design exist, with limited connection
- When to introduce certain topics, in what order, how in depth to go

What existing classes can be referenced?

Digital	Bring-up	Analog	Hardware
CPRE 281	CPRE 288	EE 330	EE 201
CPRE 381	CPRE 488	EE 435	EE 230
CPRE 480		EE 501	EE 333
CPRE 487			
EE 465			

Motivation In Context - Past Work

Past Senior Design Teams

- 4 Digital teams, 1 Analog team
- Each group generated an independent design using eFabless and related tools
- Received fabricated/packaged chips from first digital senior design group

Documentation

- Targeted **Digital** and **Analog** sections
- Most beneficial to show how to run required tools

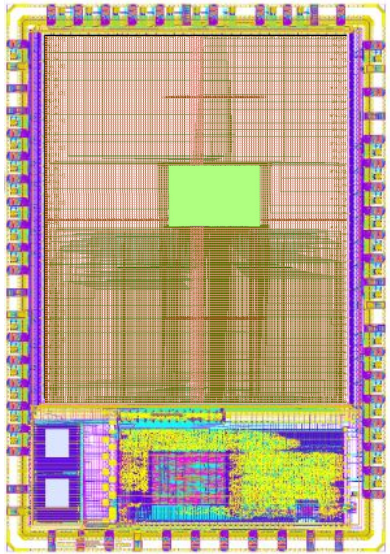


Figure 6: Fall 2022 SHA-1 Hasher GDSII [12]

Motivation In Context - Past Work

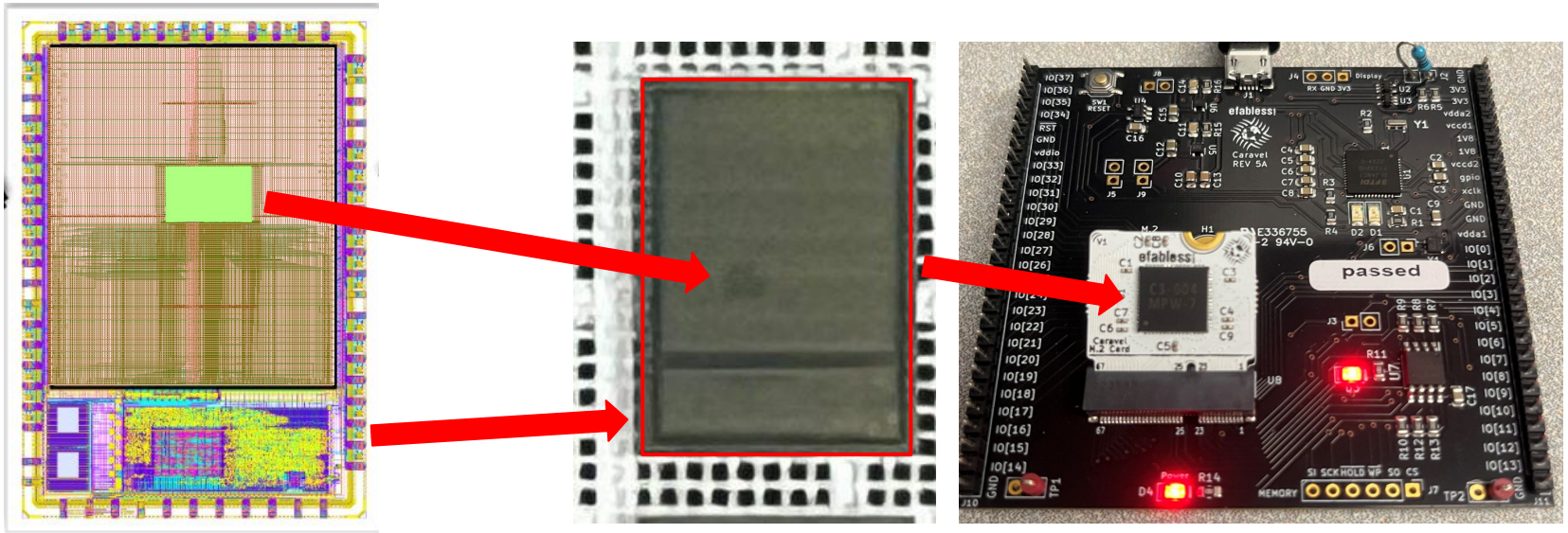
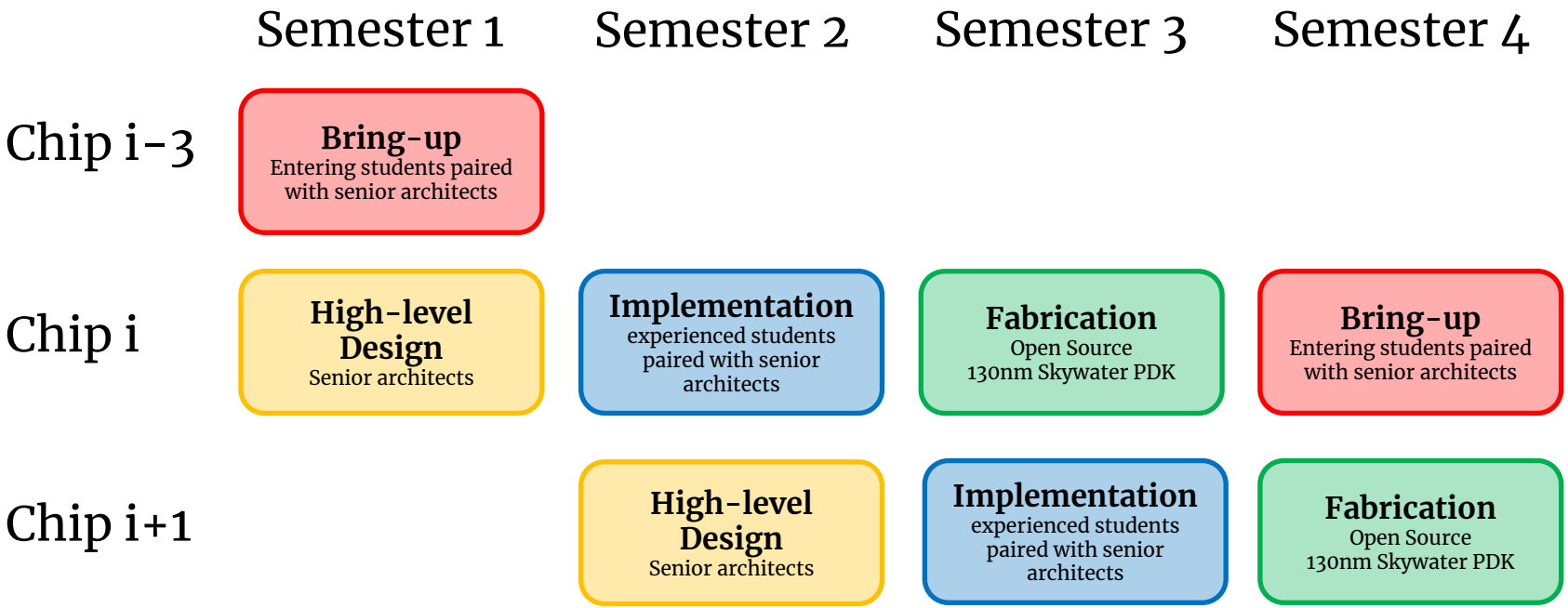


Figure 7: SHA-1 Fabricated Die and Package [12]

Motivation In Context - Chip Forge



Motivation In Context - Chip Forge

Current Senior Design Teams

- 2 Digital Teams, 1 Analog team
- Provided documentation/tutorials for design support
- Received valuable feedback to improve work

First-Year Honors Mentor Program

- Initial bring-up of first fabricated ASIC with 5 freshman ECpE students
- Targeted for **Bring-up** documentation section

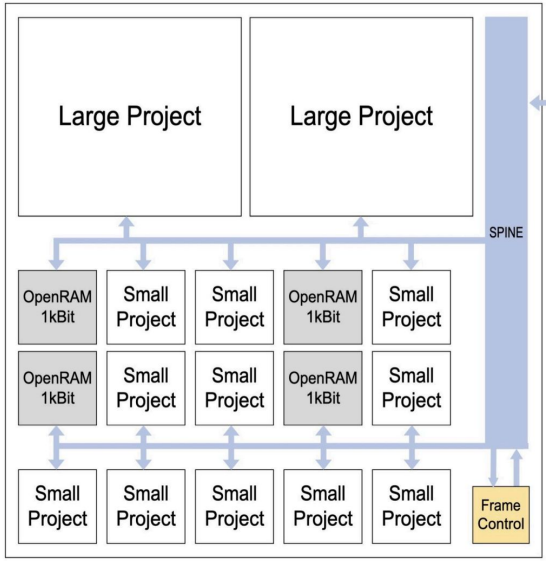


Figure 8: Fall 2024 Tapeout Architecture

Motivation In Context - Chip Forge

Internal Chip Forge Tools

- Derived as part of Gregory's Thesis [14]
- Centralized dependency package in Toolchain
- Remote access with comparch lab machine
- Development Board with M.2 Connector
- Seven Segment Display and Memory PMODs

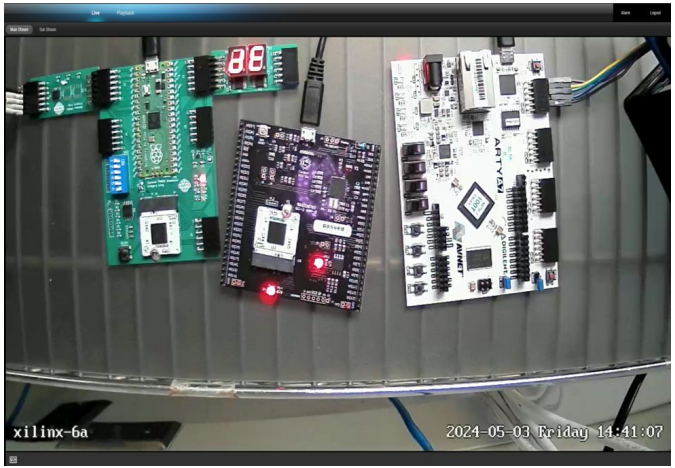


Figure 9: Chip Forge Camera Interface

Related Work - Documentation

Technical Documentation

- Developer surveys show most effective attributes of documentation include up-to-date information, examples, and organized sections [15]
- Developers prioritize content in documentation over how it is written [16]
- Documenting using tools was easier to create on average, but challenging to maintain. Tutorials are an effective training mechanism [17]

Tutorials

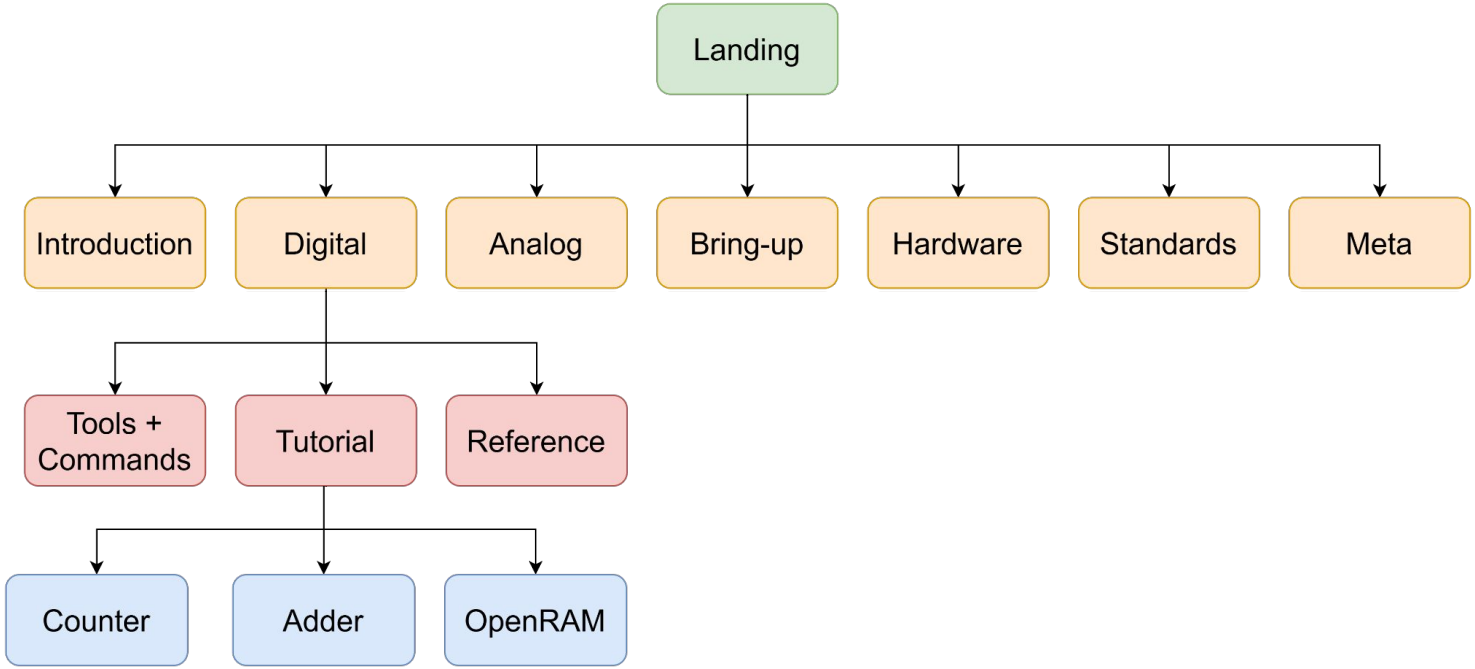
- Important aspects of programming tutorials include reference code and testing tutorials after writing [18]

Main Idea - Principles

Principles derived from Related Work

- Keep it Simple
- Direct Access
- Ease of Maintenance
- Contextual Information

Main Idea - Structure



Main Idea - Development

How is documentation constructed?

Main Idea - Development

```
\begin{circuitikz}
    \draw (0, 2) node[buffer port, scale=-1](A){};
    \draw (0, 0) node[buffer port](B){};
    \draw (A.out) -- ++(-0.5, 0) -- ++(0, -2) -- (B.in 1);
    \draw (A.south){} -- ++(0, -0.3) -- ++(0, 1.3) -- ++(2, 0)
          node[right]{Dir};
    \draw (A.in) -- ++(1, 0) node[right]{Out};
    \draw (B.out) -- ++(1, 0) node[right]{In};
    \draw (A.out) -- ++(-2, 0) node[left,
anchor=south]{In/Out};
\end{circuitikz}
```

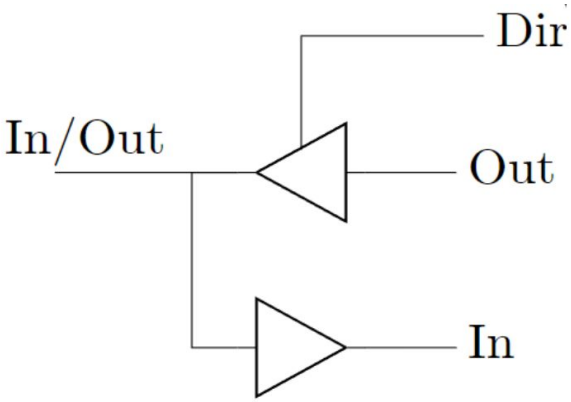


Figure 10: CircuiTikz Tri-State Buffer

Documentation Content - Introduction

Landing Page

- Provides a direct pointer to each subpage of the wiki
- Guides readers to digital, analog, bring-up sub sections
- Includes index bar for clear navigation

Introduction

- Intended for new member onboarding
- Includes instructions for internal tools and accesses
- References for git and linux for inexperienced members

Figure 11: Landing Page Link

https://git-pages.ece.iastate.edu/isu-chip-fab/documentation/

Documentation Content - Digital

Tools + Commands

- iVerilog (Register Transfer Level and Gate Level Simulations)
- Yosys (Synthesis)
- OpenLane (Place and Route, Static Timing Analysis)
- GTKWave (Waveform Viewer)
- KLayout (GDS Viewer)

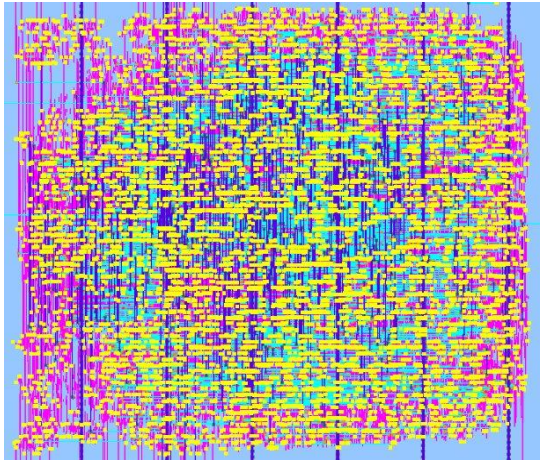


Figure 12: 381 Processor GDSII

Documentation Content - Digital

Tutorials

- Multiple tutorials with sample code
  - Counter/adder on wishbone bus
  - Open source RAM module (OpenRAM)
- Tutorials provide background on functionality of design, alongside how to run each step of tool

Reference

- Point to helpful links/deliverables/terminology

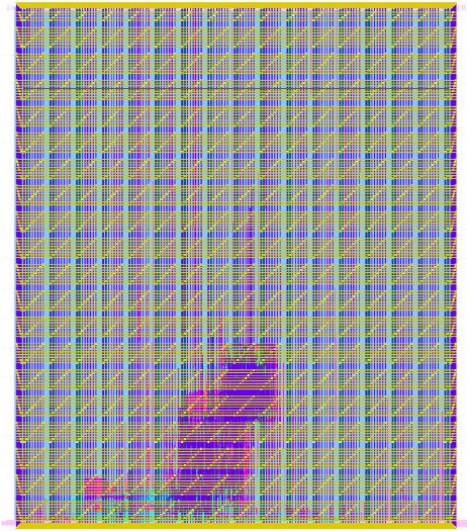


Figure 13: Fall 2024 Chip Forge Tapeout

Documentation Content - Analog

Tools + Commands

- Commands for analog design, verification, layout, and submission
  - XSchem (Schematic design and sim)
  - Magic VLSI (Layout)
  - netgen (LVS)

Tutorials

- End to End Inverter design, utilized past deliverable
- Less developed than Digital pages due to less analog teams through process

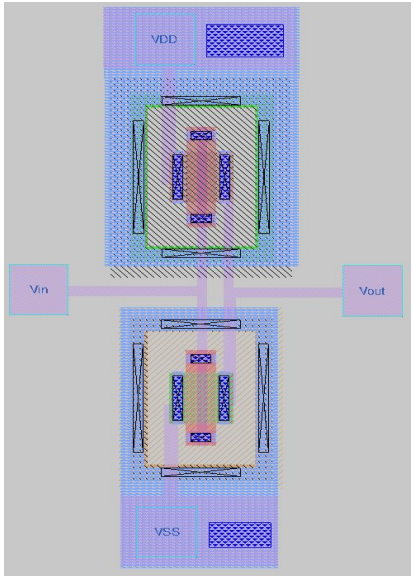


Figure 14: Inverter layout in Magic VLSI

Documentation Content - Bring-up

- Driven by Gregory as part of thesis
- Technical introduction for new members

- Includes tutorials for flashing C code to fabricated ASICS and synthesized FPGAs

- CircuiTikz schematics included as examples for new members guidance

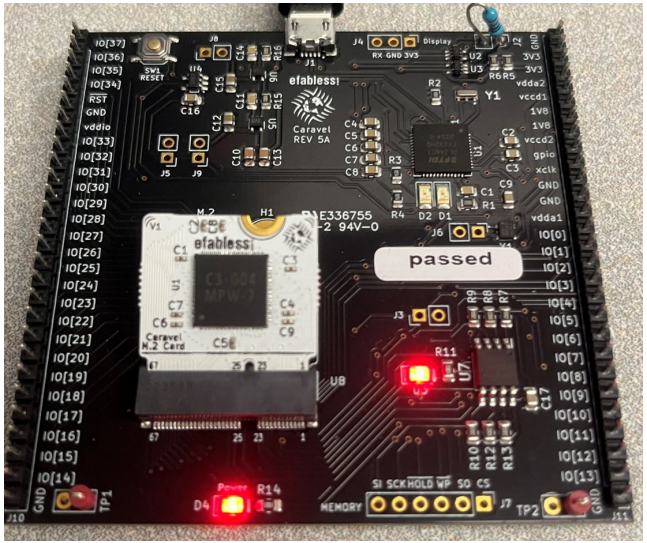


Figure 15: eFabless Development Board (Black) and Breakout Board (White)

Documentation Content - Hardware

- Points to hardware repo and website developed by Gregory
- Explains each file type on hardware html website
- Important to reference when interfacing off-chip hardware
- Could be improved with more PCB design in future

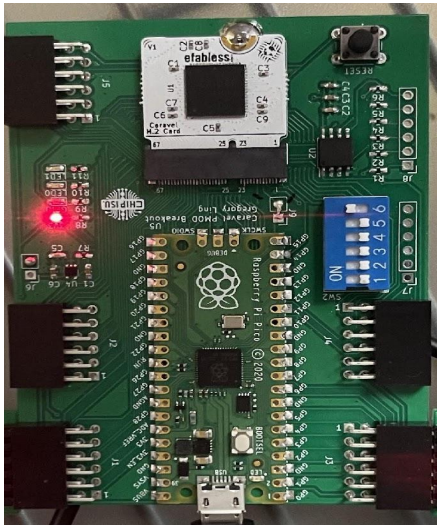


Figure 16: Chip Forge Development Board (Green) and Breakout Board (White)

Documentation Content - Standards/Meta

Standards

- Includes sample set of Verilog coding standards for digital designs
- Stubbed out sections for C code, File naming, and Git

Meta

- Where to add new members/leadership when positions change
- How to maintain docsify wiki
- Where to update new information on wiki

```
module DESIGN(input D0,
              input D1,
              input S,
              inout D,
              output X);
```

```
module DESIGN(input D0_i,
              input D1_i,
              input S_i,
              inout D_io,
              output X_o);
```

Figure 17: Verilog Coding Example

Evaluation

Documentation Usage

- 4 Senior Design Teams (Estimated 15 students)
- 5 Freshman Honors Students
- Chip Forge (Estimated 20 students)

Informal Feedback

- Difficult to navigate landing page
- Pages with tutorials/tool commands were too long
- Not enough background information on digital/embedded design

Evaluation

Future Survey Questions

1. What aspects of documentation were most beneficial/challenging?
2. What was used most often from the provided documentation?
3. How did the provided wiki impact your onboarding experience?
4. When did you need to reference external documentation not on the wiki?
5. Are there any suggestions for improvements?

Future Work

Additional Documentation

- Analog tutorials and tooling
  - Has been improved by current senior design team some already
- Update documentation based on survey and informal feedback

Maintenance

- Ensure documentation matches current eFabless design flows
- Update as internal Chip Forge tools developed

Future Work

Integration into existing curriculum

- Chip Forge members can utilize experience and documentation in curriculum

Example

- Multiply and Accumulate module from CPRE 587
- Provided tapeout ready design in **1 day** and verified with RTL/GL/FPGA in **2 weeks**

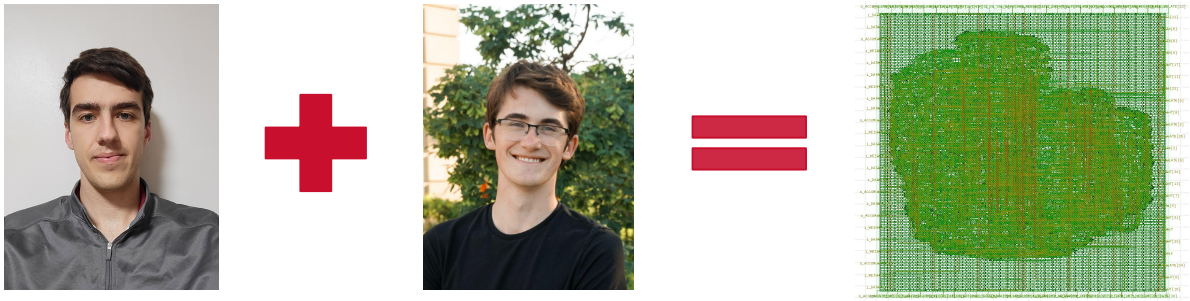


Figure 18: CPRE 587 MAC Unit GDSII

Conclusion

Summary

- Students are provided required information for onboarding, tooling, and references for each ASIC design flow within Chip Forge
- Set baseline for future documentation with chosen tools and strategies

Results

- Documentation has enabled groups to reach tapeout ready designs faster
- Very first tapeout of Chip Forge has been submitted

Future

- Determine how to improve and expand documentation with survey results
- Integrate Chip Forge into coursework and industry

References

[1] Powell, H., Williams, R., Weikle, R., and Brandt-Pearce, M. (2015). Towards a t shaped electrical and computer engineering curriculum: A vertical and horizontally integrated laboratory/lecture approach. 122.

[2] Olewnik, A., Chang, Y., and Su, M. (2015). Co-curricular engagement among engineering undergrads: do they have the time and motivation? 10.

[3] Spickard-Prettyman, S., Qammar, H., Broadway, F., Cheung, H., and Evans, E. (2004). The impact of vertical integration of design teams on the chemical engineering program. In 34th Annual Frontiers in Education, 2004. FIE 2004., pages T2G/15–T2G/19 Vol. 1.

[4] Ramirez, N. and Zoltowski, C. B. (2022). First-year experiences – how the vertically integrated projects (vip) model addresses grand challenges and abet outcomes. In 2022 IEEE Frontiers in Education Conference (FIE), pages 1–4.

[5] eFabless. efabless. https://efabless.com/. Accessed: 2024-27-7.

[6] Tiny Tapeout. Tiny tapeout. https://tinytapeout.com/. Accessed: 2024-4-8.

[7] Muse Semiconductor. Muse semiconductor. https://www.musesemi.com/. Accessed: 2024-4-8.

References

[8] Edwards, R. T., Shalan, M., and Kassem, M. (2021). Real silicon using open-source eda. IEEE Design Test, 38(2):38–44.

[9] Herman, K., Montanares, M., and Marin, J. (2023). Design and implementation of integrated circuits using open source tools and sky130 free pdk. In 2023 30th International Conference on Mixed Design of Integrated Circuits and System (MIXDES), pages 105–110.

[10] Chen, P.-H., Tsao, C., and Raina, P. (2023). An open-source 4 × 8 coarse-grained reconfigurable array using skywater 130 nm technology and agile hardware design flow. In 2023 IEEE International Symposium on Circuits and Systems (ISCAS), pages 1–5.

[11] Cirimelli-Low, J., Khan, M. H., Crow, S., Lonkar, A., Onal, B., Zonenberg, A. D., and Guthaus, M. R. (2023). Sram design with openram in skywater 130nm. In 2023 IEEE International Symposium on Circuits and Systems (ISCAS), pages 1–5.

[12] eFabless Caravel. efabless caravel. https://github.com/efabless/caravel. Accessed: 2024-28-7.

[13] Efabless YouTube Channel. Efabless youtube channel. https://www.youtube.com/@efabless_channel. Accessed: 2024-3-8.

[14] Ling, G. (2024). Design and organization of a bring-up first vertically-integrated chip design and fabrication co-curricular.

References

[15] Forward, A. and Lethbridge, T. C. (2002). The relevance of software documentation, tools and technologies: a survey. In Proceedings of the 2002 ACM Symposium on Document Engineering, DocEng '02, page 26–33, New York, NY, USA. Association for Computing Machinery.

[16] Visconti, M. and Cook, C. (2002). An overview of industrial software documentation practice. In 12th International Conference of the Chilean Computer Science Society, 2002. Proceedings., pages 179–186.

[17] Dagenais, B. and Robillard, M. P. (2010). Creating and evolving developer documentation: understanding the decisions of open source contributors. In Proceedings of the Eighteenth ACM SIGSOFT International Symposium on Foundations of Software Engineering, FSE '10, page 127–136, New York, NY, USA. Association for Computing Machinery.

[18] Head, A., Jiang, J., Smith, J., Hearst, M. A., and Hartmann, B. (2020). Composing flexibly-organized step-by-step tutorials from linked source code, snippets, and outputs. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, CHI '20, page 1–12, New York, NY, USA. Association for Computing Machinery.

Questions

```
{{FINISH}} Executing Finished, the full log 'precheck.log'
{{SUCCESS}} All Checks Passed !!!
```