

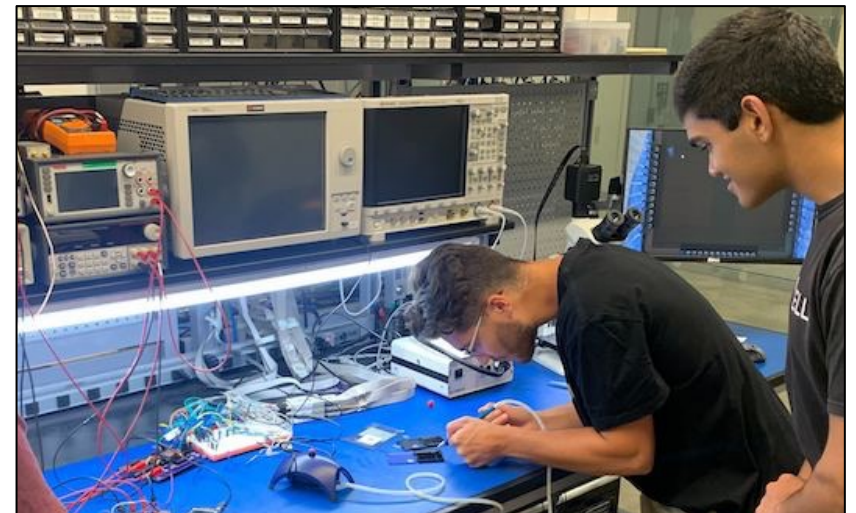
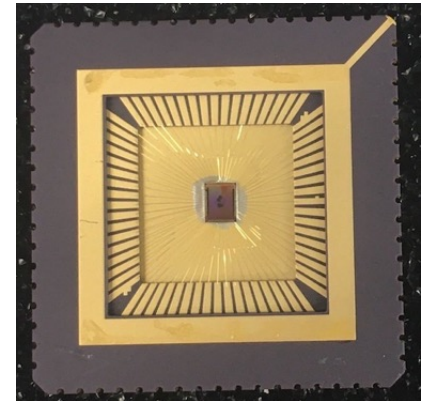
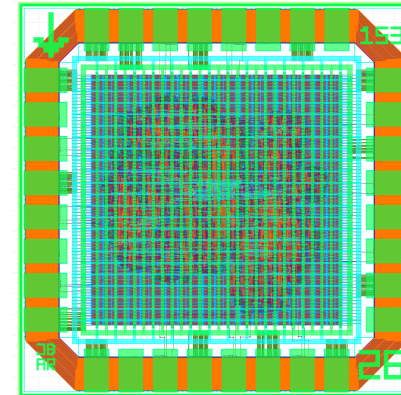
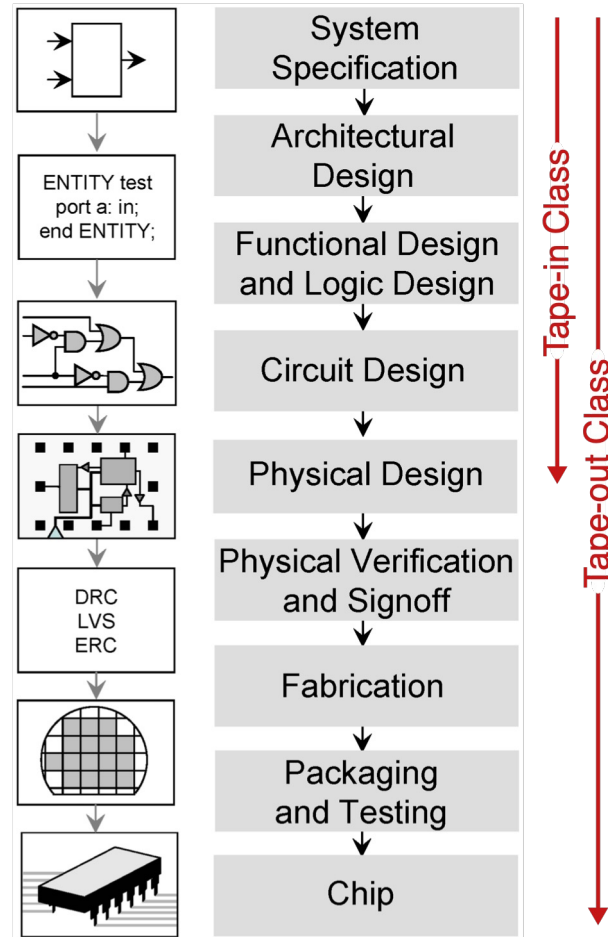
ECE 4745/5745

Digital ASIC Design

Christopher Batten
Cornell University
Spring 2026

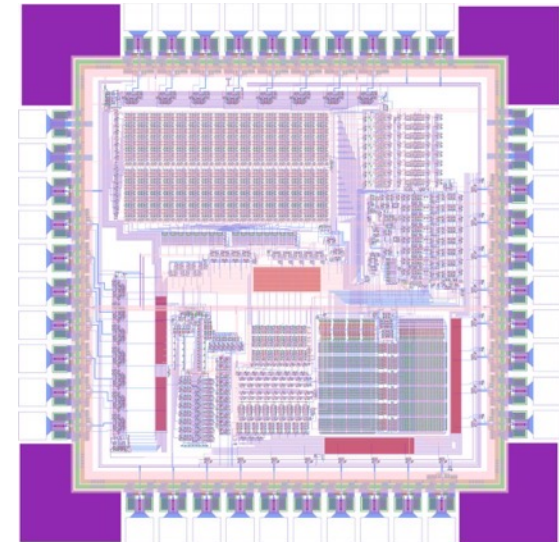
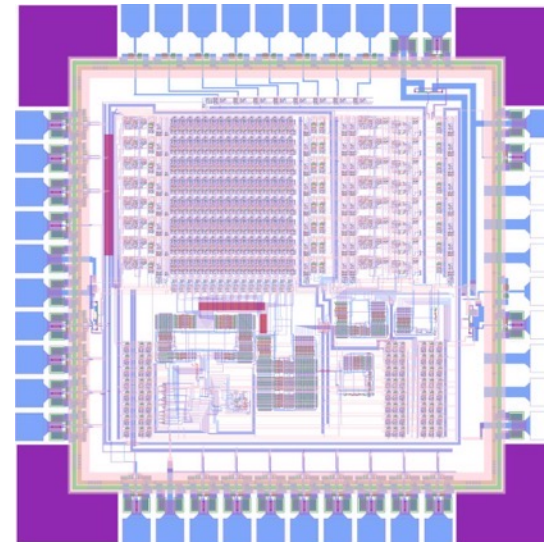
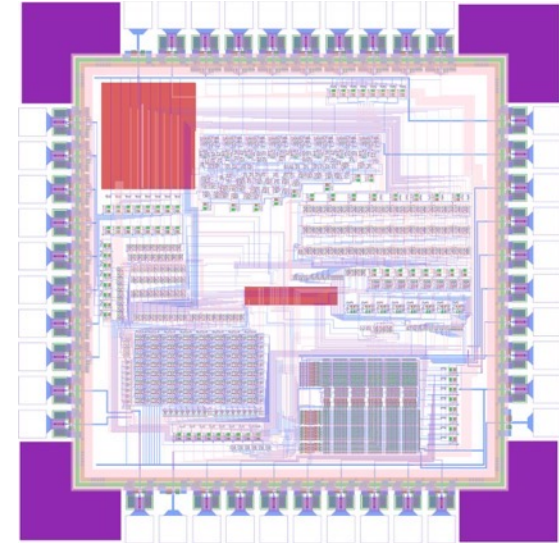
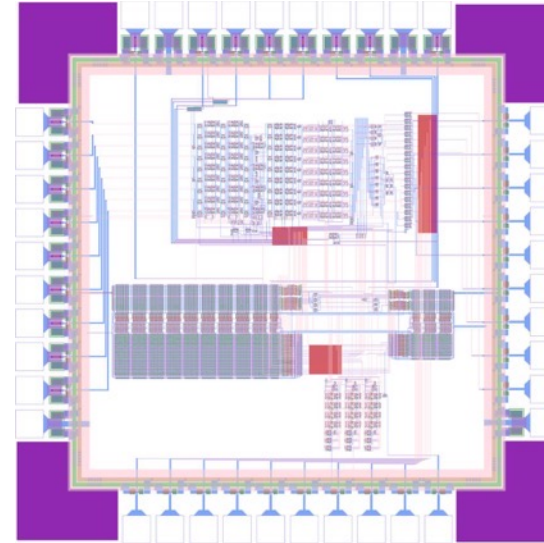
Teaching Assistants

- Elton Shih
- Parker Schless $\mapsto$ Apple
- Vayun Tiwari $\mapsto$ NVIDIA
- Irwin Wang



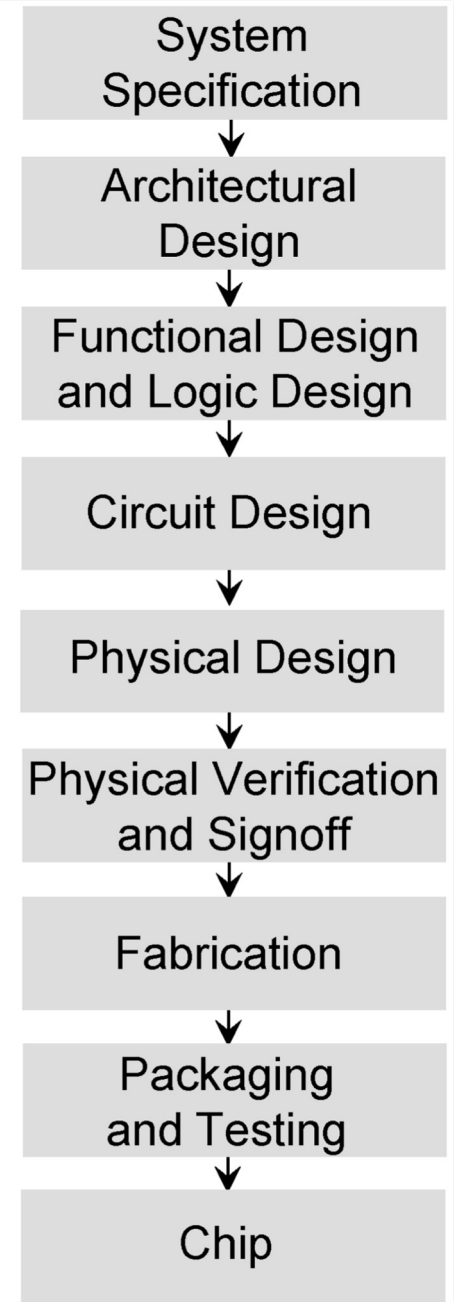
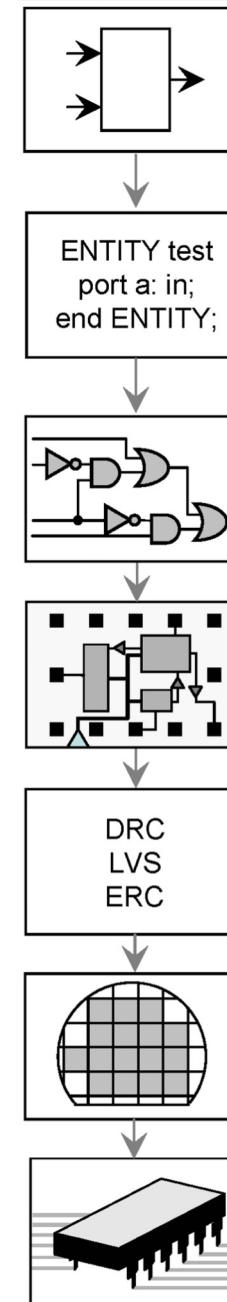
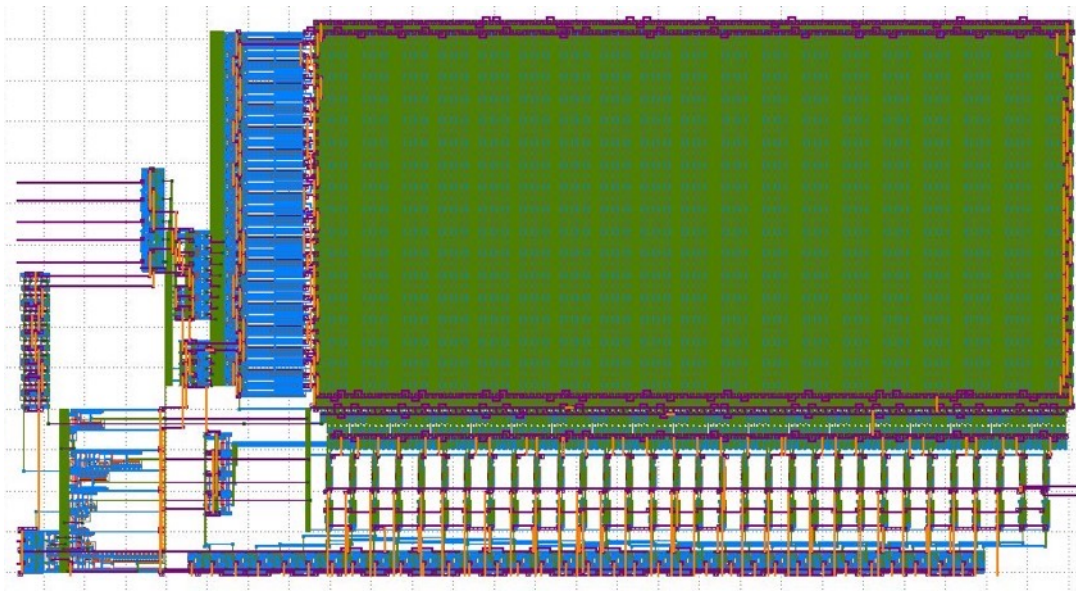
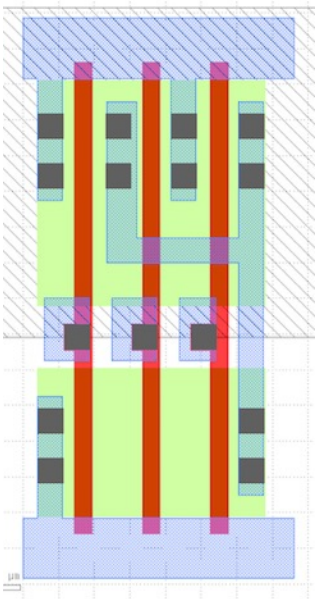
ECE 4750 Digital VLSI Tape-Out Option

- ECE 4740 included a tape-out option in early 2000s when taught by Rajit Manohar (now at Yale)
- 1.5x1.5mm in AMI 0.5um
- Full-custom digital design using Magic and open-source tools
- Fully funded through MOSIS instructional program
- 2003 was the final year of the tape-out option due to funding and few instructors who could teach the course



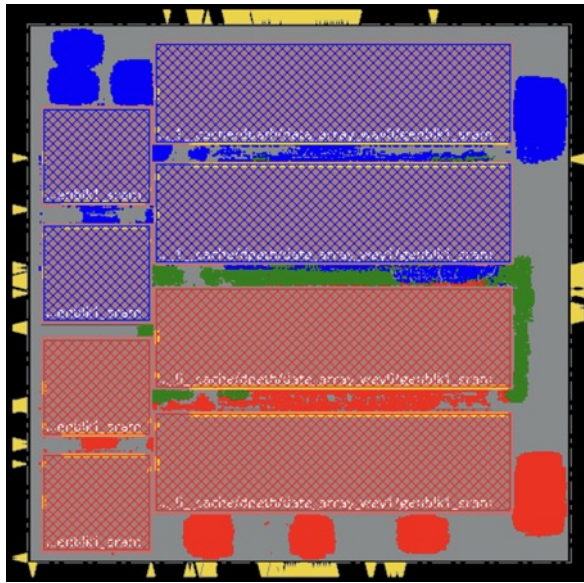
ECE 6745 Digital ASIC Design

- ECE 6745 teaches digital ASIC design using automated standard-cell-based methodologies
- Students "tape-in" design projects using commercial tools from Synopsys and Cadence with FreePDK45, Nangate standard cells, and OpenRAM SRAM generator
- Only prerequisite is **senior-level computer architecture course** (no advanced digital VLSI is required)

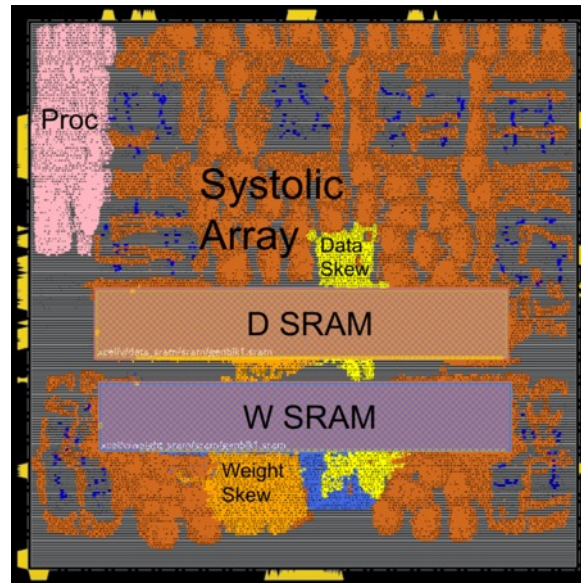


Tape-in Class
Tape-out Class

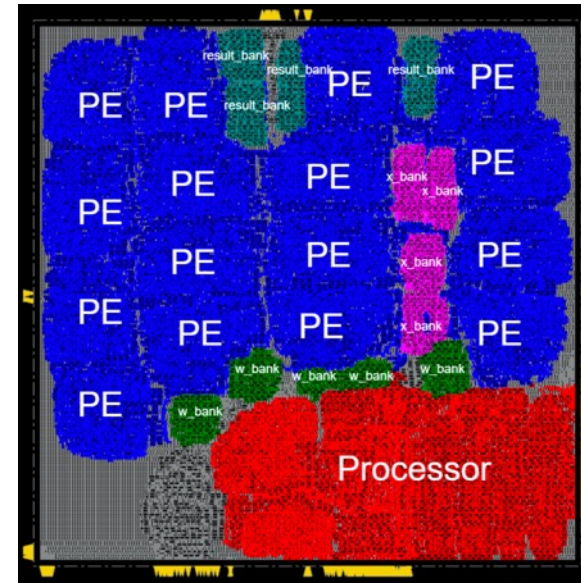
ECE 6745 Digital ASIC Design Tape-In Projects



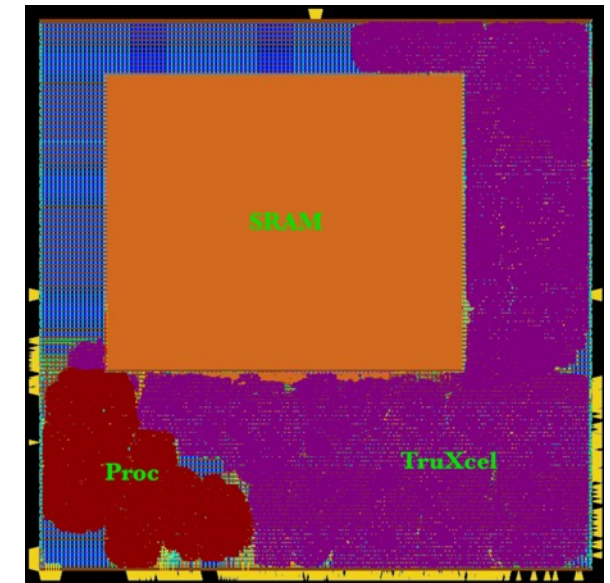
Snoopy Cache Coherent
Memory System



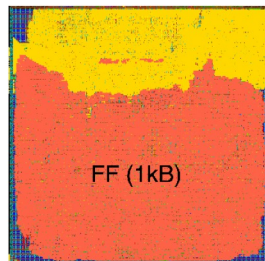
MLP Xcel



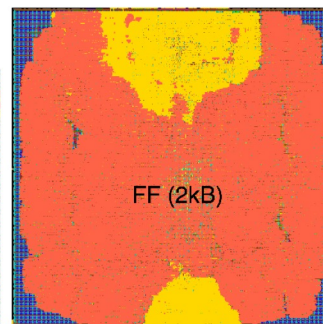
MLP Xcel



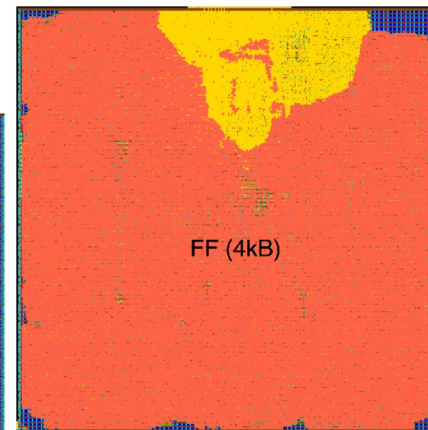
Graphics Rendering Xcel



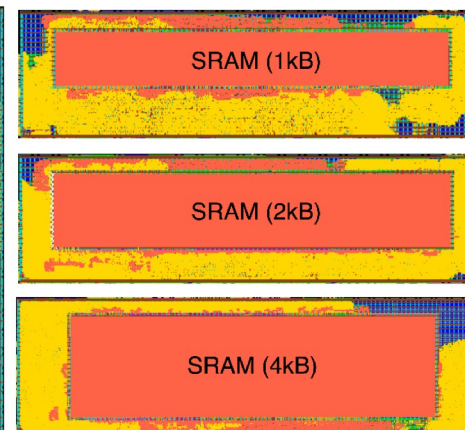
FF (1kB)



FF (2kB)



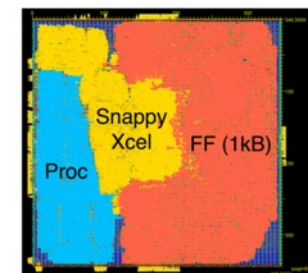
FF (4kB)



SRAM (1kB)

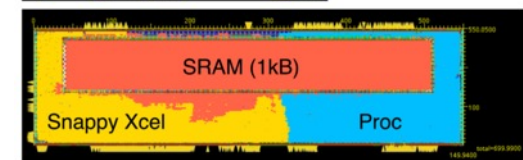
SRAM (2kB)

SRAM (4kB)



Snappy
Xcel

FF (1kB)



Snappy Xcel

SRAM (1kB)

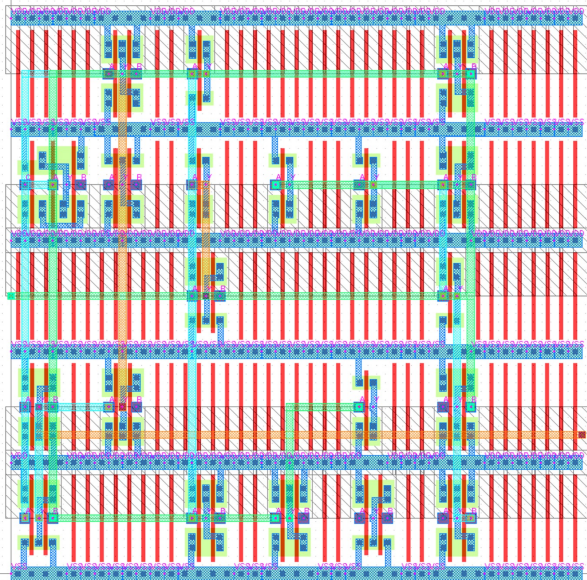
Snappy
Decompression
Xcel

Challenges in Tape-in vs. Tape-out Courses

	Advantages	Disadvantages
Tape-In Course	Enables large complex design projects or projects in other areas (e.g., semi-custom, EDA)	No experiential learning on an actual tape-out and chip testing
Tape-Out Course w/ small groups taping out their own chip	Each group has complete ownership over their own tape-out	NDA delays start; Relatively small projects with significant area dedicated to I/O in older technology node
Tape-Out Course w/ large group taping out large shared chip	Tape-out can be a large and sophisticated SoC on a more advanced technology node	NDA delays start; Students have less ownership over the actual tape-out (e.g., final physical design)
Tape-Out Course w/ small groups taping out blocks on shared chip	Blocks can share I/O and potentially a more advanced technology node	NDA delays start; Students have even less ownership over the actual tape-out (e.g., integration, I/O)
		Digital ASIC flow is often treated like an automated black box

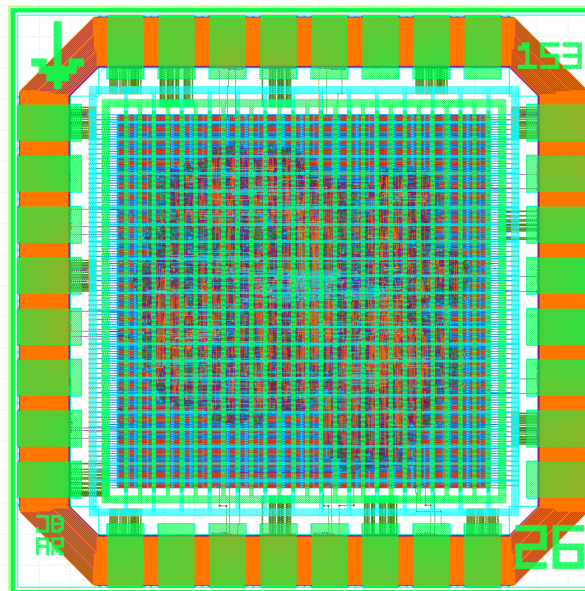
ECE 4745/5745 Digital ASIC Design

Project 1: Tiny Flow Tape-Out (4-5 weeks, 2-3 students/group)



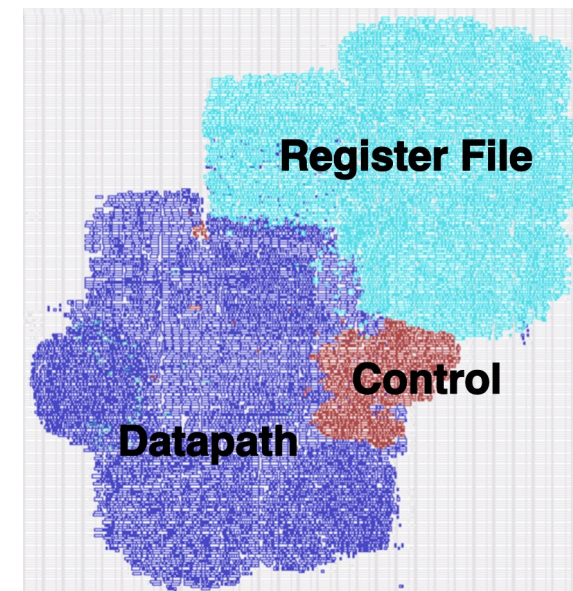
Implement standard cells using lambda rules then very simple Python-based combinational RTL-to-layout flow from scratch; blocks integrated onto shared 1x1mm TSMC 180nm chip

Project 2: Accelerator Tape-Out (4-5 weeks, 2-3 students/group)



Implement accelerator using commercial standard cells and EDA tools; evaluate with RISC-V processor+caches but tapeout just accelerator on 1x1mm TSMC 180nm chip

Project 3: SoC Tape-In (4-5 weeks, 2-3 students/group)



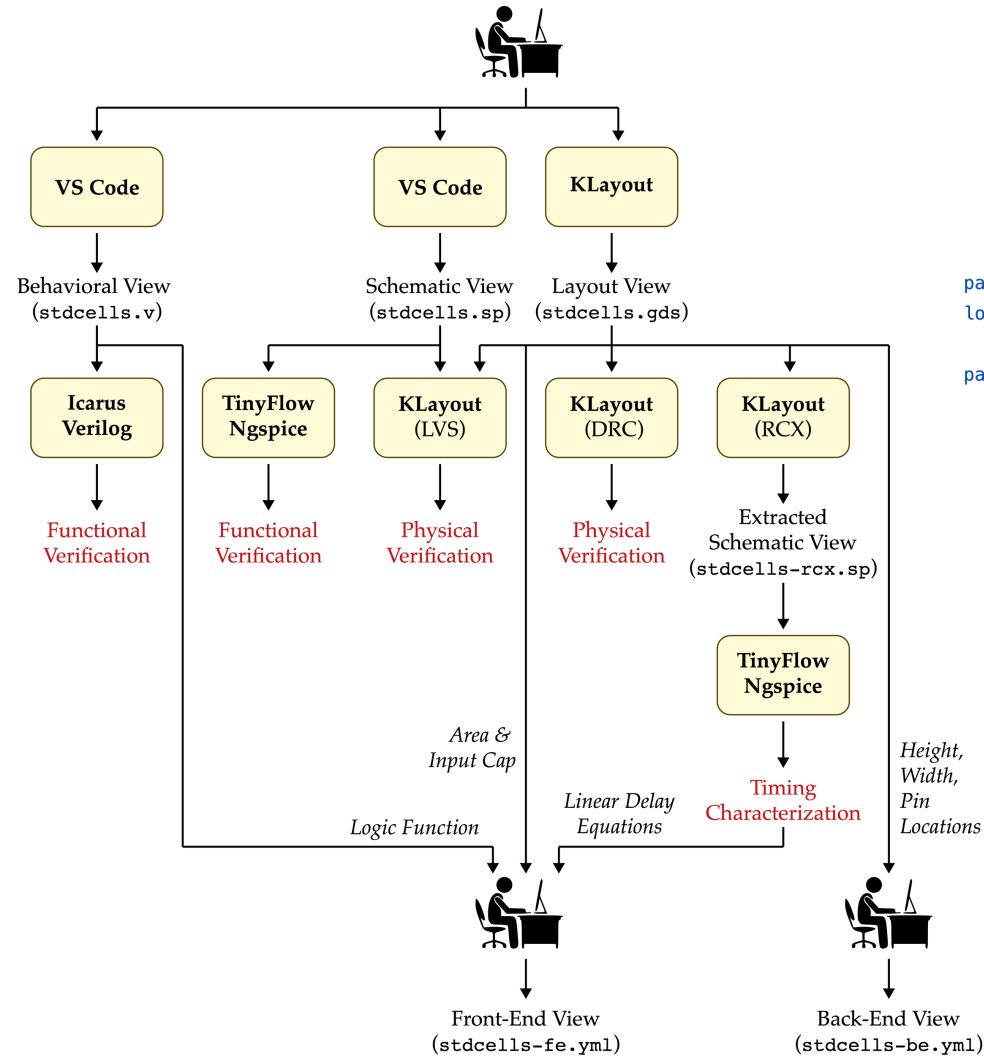
Students free to explore advanced processors, memories, networks and/or accelerators with a strong emphasis on a rigorous comparative analysis of performance, area, and energy

Challenges in Tape-in vs. Tape-out Courses

	Advantages	Disadvantages	ECE 4745/5745
Tape-In Course	Enables large complex design projects or projects in other areas (e.g., semi-custom, EDA)	No experiential learning on an actual tape-out and chip testing	Projects 1+2 enable tape-out
Tape-Out Course w/ small groups taping out their own chip	Each group has complete ownership over their own tape-out	NDA delays start; relatively small projects with significant area dedicated to I/O in older technology node	Project 1 first; Project 3 enables large project
Tape-Out Course w/ large group taping out large shared chip	Tape-out can be a large and sophisticated SoC on a more advanced technology node	NDA delays start; students have less ownership over the actual tape-out (e.g., final physical design)	Project 1 first; Project 2 enables ownership
Tape-Out Course w/ small groups taping out blocks on shared chip	Blocks can share I/O and potentially a more advanced technology node	NDA delays start; students have even less ownership over the actual tape-out (e.g., integration, I/O)	Project 1 first; Project 2 enables ownership
		Digital ASIC flow is often treated like an automated black box	Project 1 demystifies flow; all projects use pyhflow
Christopher Batten	Cornell ECE 4745/5745 Digital ASIC Design		7

Project 1: Tiny Flow Tape-Out

- Students develop four standard cells using KLayout and 180nm lambda rules including
 - Behavioral view
 - Schematic view
 - Layout view
 - Extracted schem view
 - Front-end view
 - Back-end view



stdcells-fe.yml

```
- name: INVX1
  area_cost: 1536 # lambda^2
  pins:

  - name: A
    type: input
    cgate: 3.6 # fF

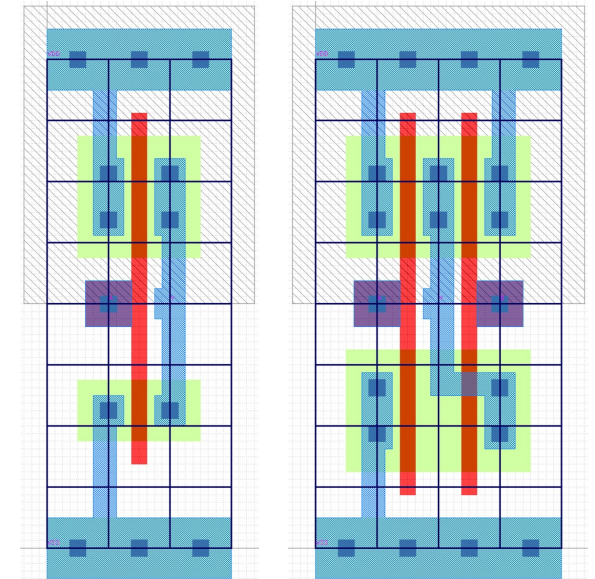
  - name: Y
    type: output
    function: INV(A)

  parasitic_delay: 11.454 # ps
  load_delay_factor: 3.306 # ps/ff

  patterns:
    - INV(A)
    - NAND2(A, A)
    - INV(INV(INV(A)))
```

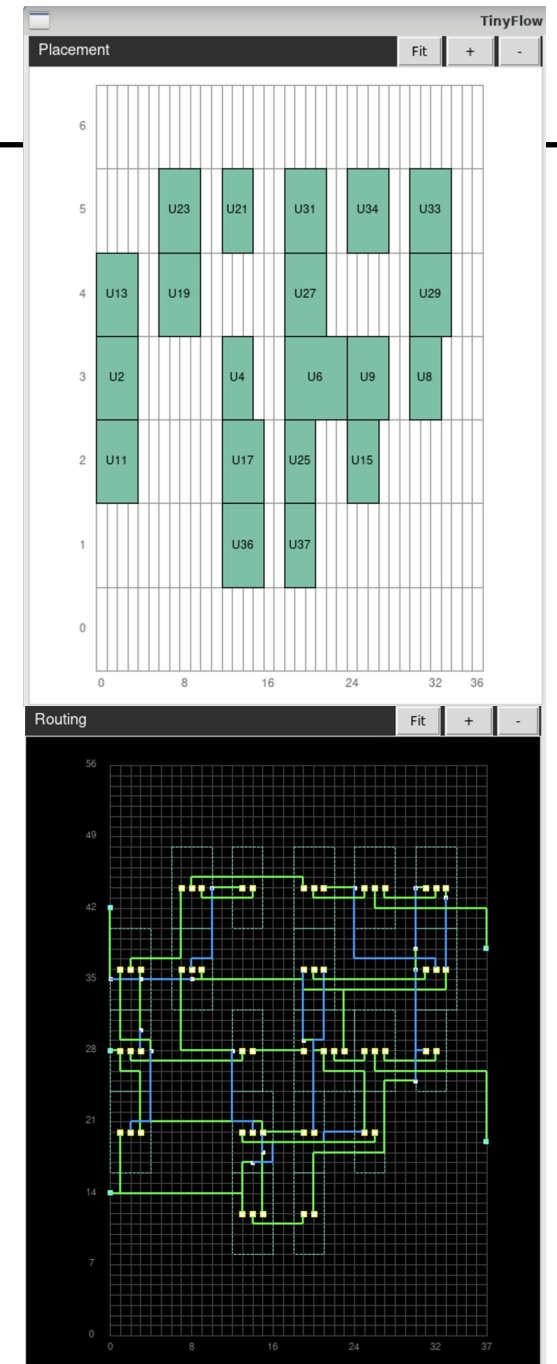
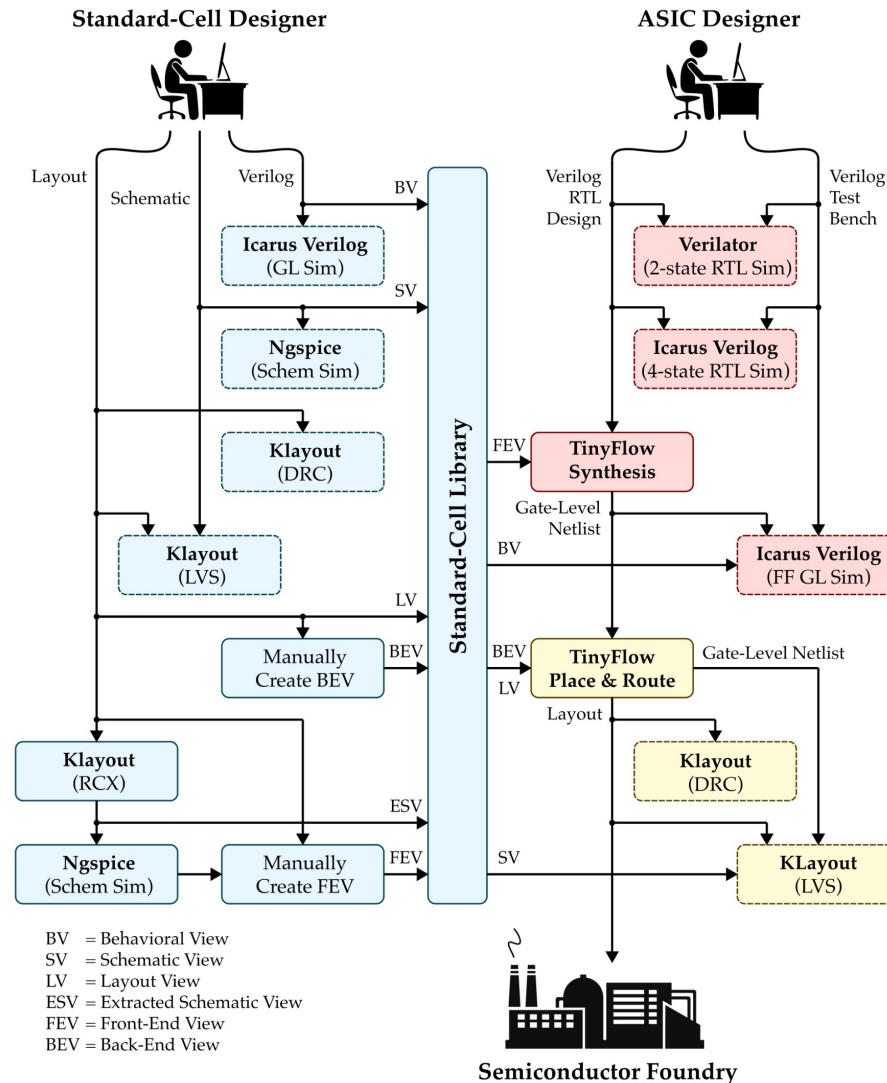
stdcells-be.yml

```
- name: INVX1
  size:
    width: 24
    height: 64
  pins:
    - name: A
      loc: (8,32)
    - name: Y
      loc: (16,32)
```



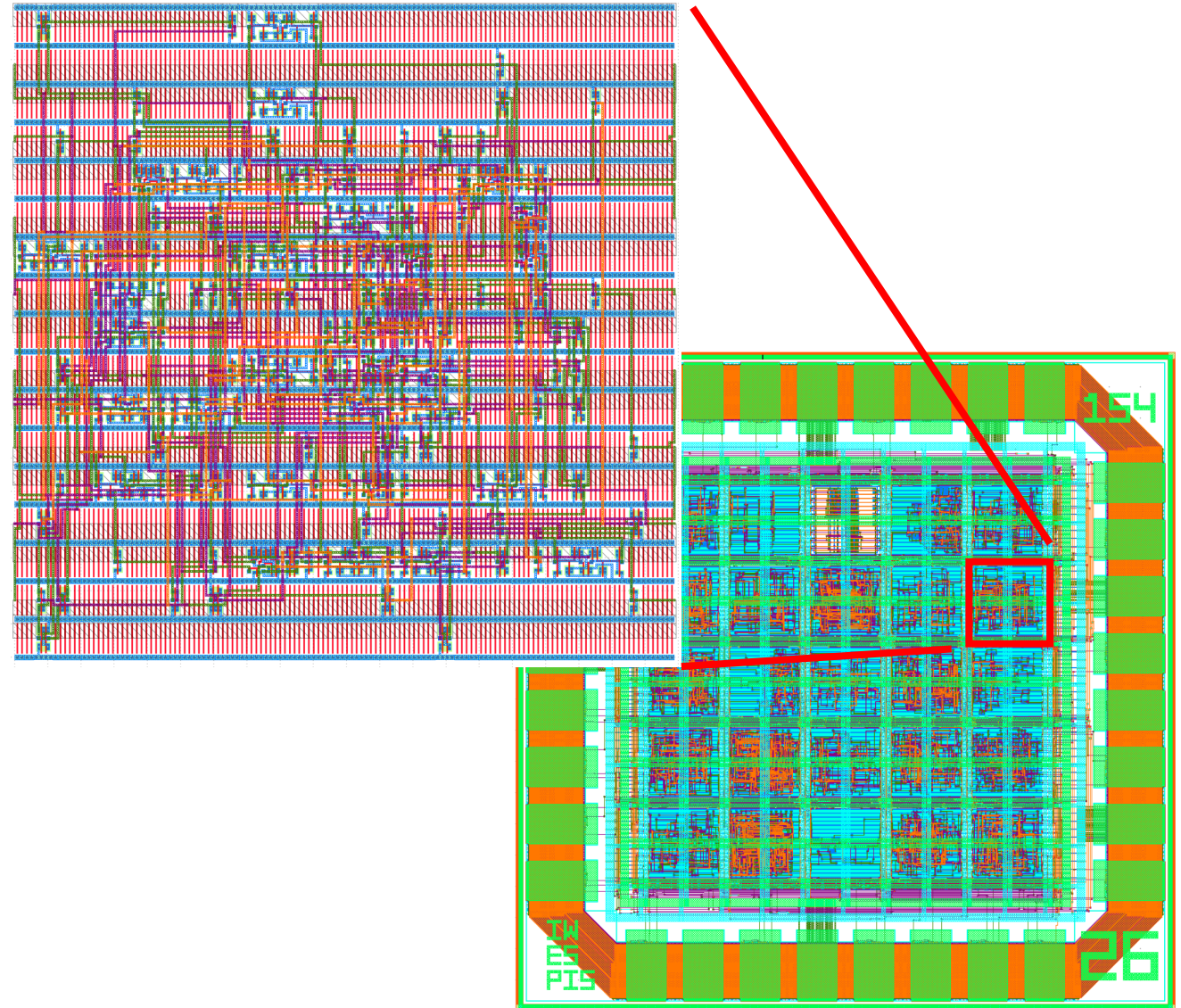
Project 1: Tiny Flow Tape-Out

- Students implemented simple algorithms for synthesis and place-and-route
 - Technology mapping** using tree covering and dynamic programming
 - Static timing analysis** to find the critical path using multiple tree traversals
 - Standard-cell placement** using simulated annealing and HPWL cost function
 - Interconnect routing** using 3D maze routing across metal layers 2–4
- Use pyhflow to lay the foundation for understanding commercial flow



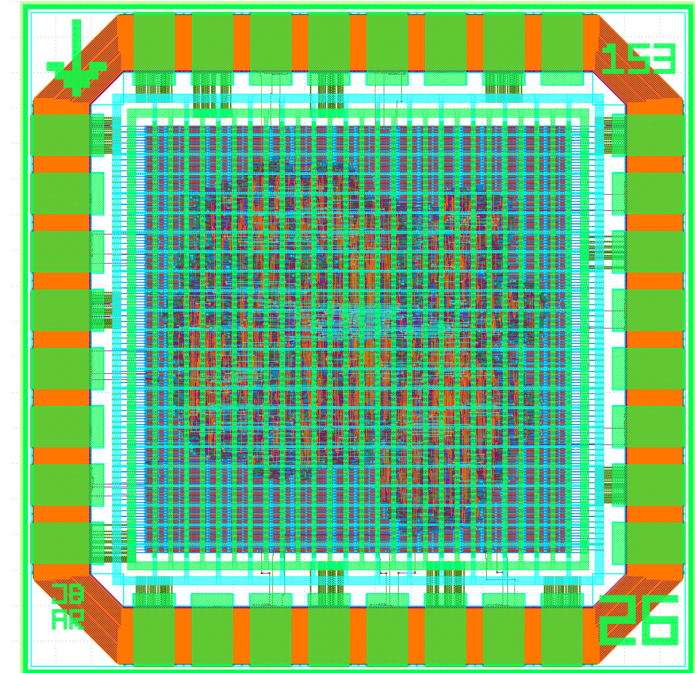
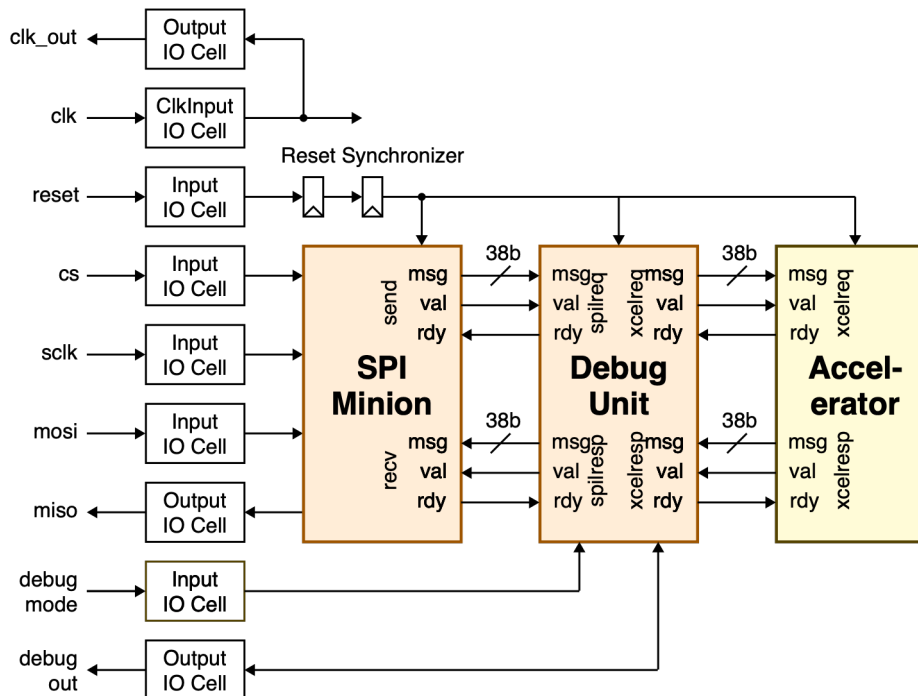
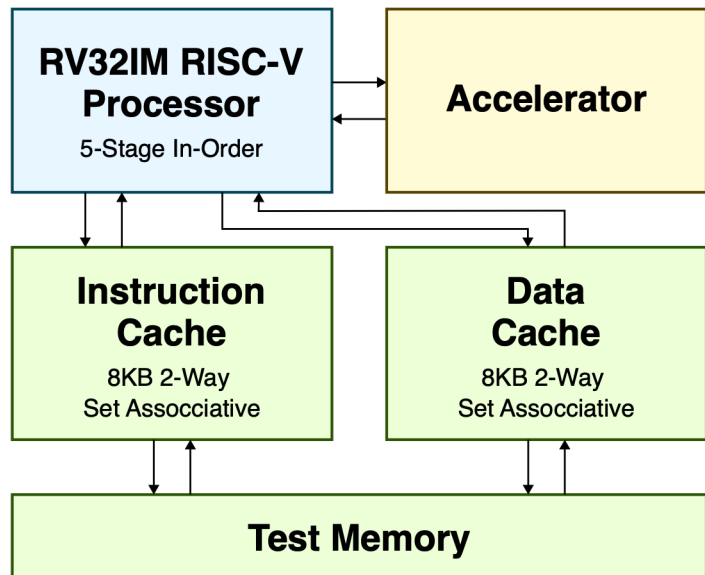
Project 1: Tiny Flow Tape-Out

- Two students implemented a four-bit multiplier as a 100x100um macro using their Tiny Flow with 544 cell instances including optional XOR2X1 standard cell; input/output pins fixed on left/right
- Lambda rules carefully designed to be nearly TSMC 180nm compliant
 - Developed Python script to transform layer numbering and VIA sizing
- TAs integrated 19 student tiny flow blocks + 6 extra tiny flow blocks onto a 1x1mm TSMC 180nm tape-out
 - Commercial standard cells for muxing logic
 - Commercial I/O cells
 - Commercial EDA tools for top-level synthesis and place-and-route
 - Careful top-level power routing!



Project 2: Accelerator Tape-Out

- Students leverage what they learned in the first project to transition to using a commercial standard-cell library and commercial electronic design automation tools
- Students develop a co-processor accelerator in Verilog RTL and evaluate the performance, area, and energy of using this accelerator compared to their own baseline software implementation on a RISC-V processor



Project 2: Accelerator Tape-Out

- pyhflow is ~100-line Python script that takes a YAML configuration file and uses the jinja2 templating library to fill in step templates
- Both templates and generated flow scripts are simple and easy to read

```
% git clone ... repo ...  
% cd repo/asic  
% mkdir build-proj2  
% cd build-proj2  
% pyhflow ../designs/proj2.yml  
% ./00-padring-gen/run  
% ./01-pymtl-rtlsim/run  
% ./02-synopsys-vcs-rtlsim/run
```

- **Block Flow**

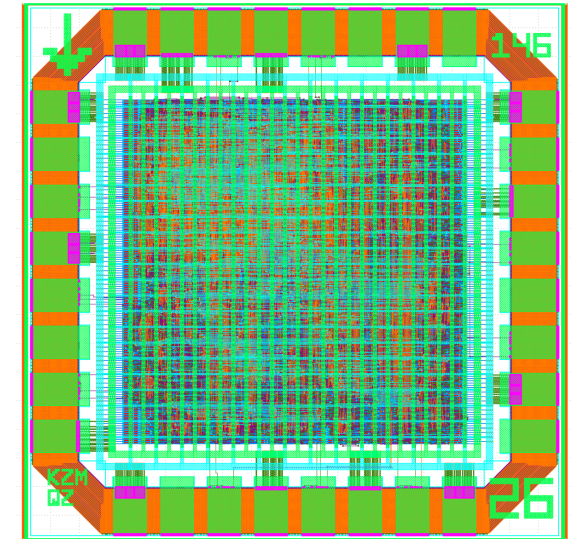
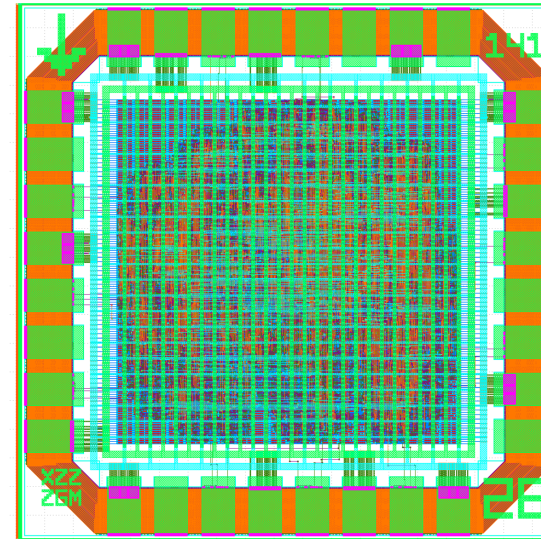
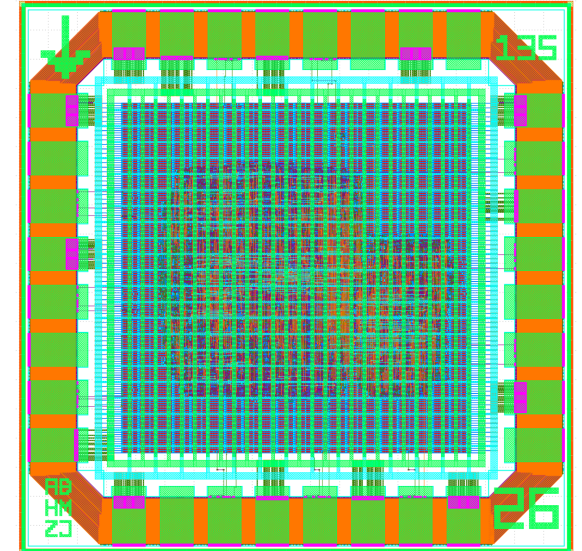
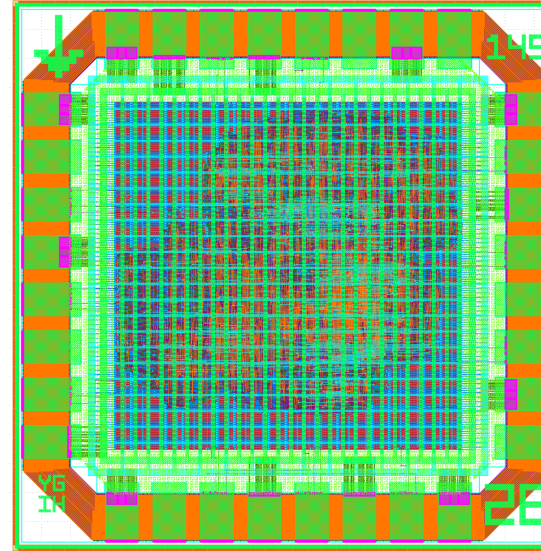
- 01-pymtl-rtlsim
- 02-synopsys-vcs-rtlsim
- 03-synopsys-dc-synth
- 04-synopsys-vcs-ffglsim
- 05-cadence-innovus-pnr
- 06-synopsys-pt-sta
- 07-synopsys-vcs-baglsim
- 08-synopsys-pt-pwr
- 09-mentor-calibre-drc
- 10-mentor-calibre-lvs
- 11-summarize-results

- **Chip Flow**

- 00-padring-gen
- 01-pymtl-rtlsim
- 02-synopsys-vcs-rtlsim
- 03-synopsys-dc-synth
- 04-synopsys-vcs-ffglsim
- 05-cadence-innovus-pnr
- 06-synopsys-pt-sta
- 07-synopsys-vcs-baglsim
- 08-synopsys-pt-pwr
- 09-mentor-calibre-seal
- 10-mentor-calibre-fill
- 11-mentor-calibre-label
- 12-mentor-calibre-drc
- 13-mentor-calibre-lvs
- 14-summarize-results

Project 2: Accelerator Tape-Out

- Binary Neural Network Layer Xcel
- DES Encryption Xcel
- FAST Image Xcel
- Simple Vector Engine (2x)
- Batched Nonlinear Unit
- Approximate Multiplier
- Dense MatMul Xcel (4x)
- Smith-Waterman Xcel (2x)
- Coarse-Grain Reconfig Array
- CORDIC Xcel (2x)
- Digital LIF Neuron Xcel

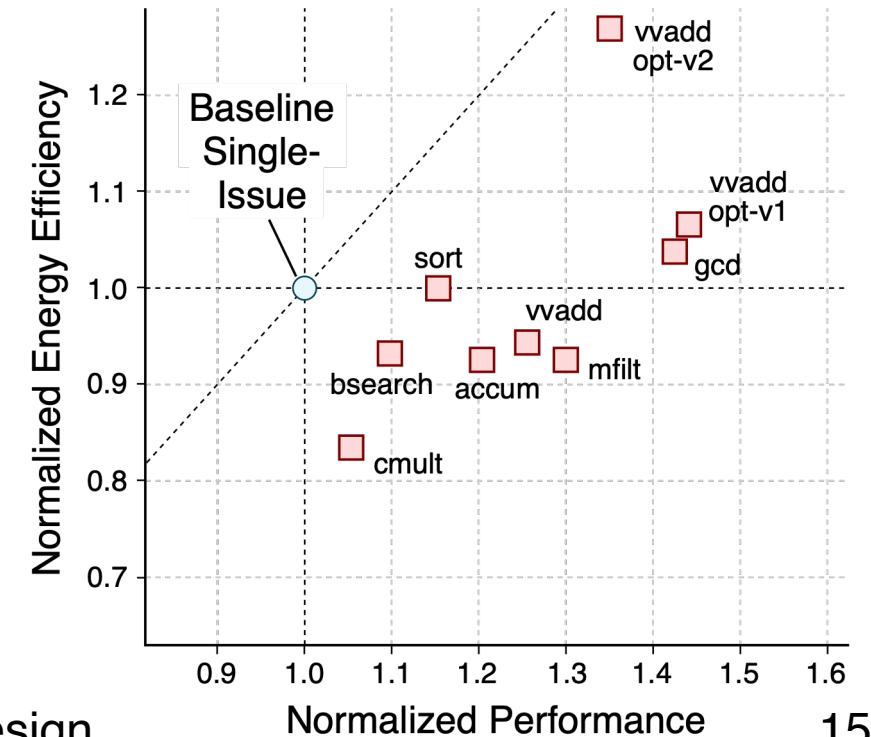
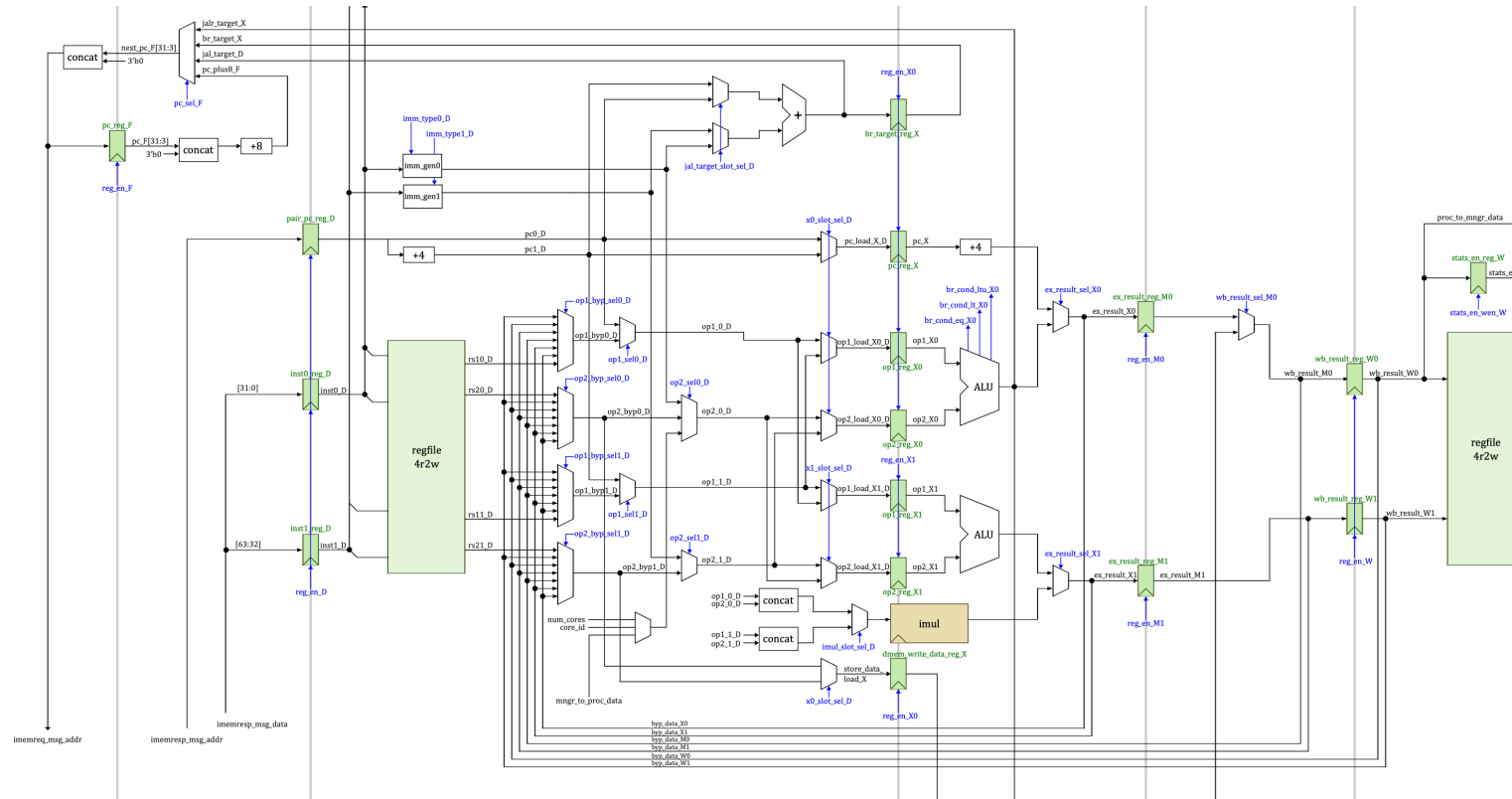
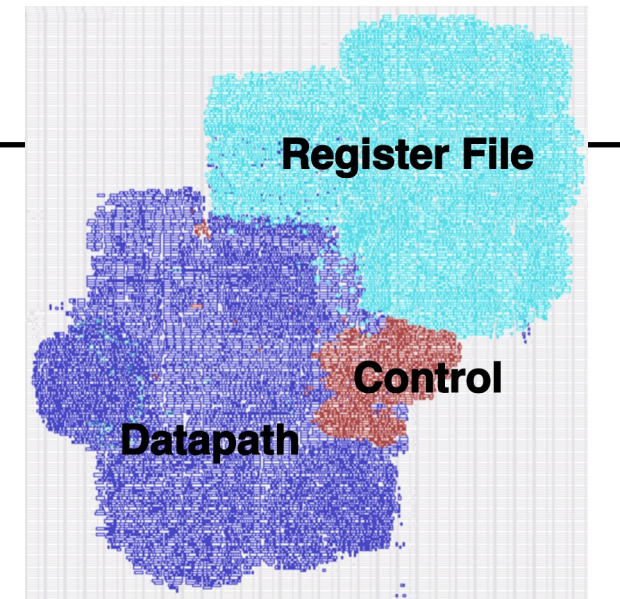


Project 3: System-on-Chip Tape-In

- Binary Neural Network Xcel
- DES Encryption/Decryption Xcel
- Optical Flow Xcel
- Coherent Memory System
- Batched Nonlinear Xcel
- Dual-Issue Superscalar Proc
- Single-Issue Out-of-Order Proc
- Non-Blocking Cache
- Iris MLP Xcel with Fixed Weights
- Mixed-Signal DVFS Exploration
- Coarse-Grain Reconfig Array
- MNIST MLP Xcel (2x)
- Lightweight Crypto Xcel
- Graphics Rendering Xcel
- Smith-Waterman Xcel
- Priority Queue Xcel
- Floating Point Unit
- Digital LIF Neuron Xcel

Project 3: System-on-Chip Tape-In

Three students implemented a **dual-issue in-order superscalar processor tape-in** which would have been too large to fit on a 1×1 mm tape-out; students completed a detailed comparison of energy & performance across various ubmarks



Spring 2026 Schedule, Statistics, Collaborations

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
19	20 Overview	21	22 T01 Basic Devices	23 Lab 1 Basic Layout	24	25
26	27 T02 Basic Std Cells	28	29 T02 Basic Std Cells	30 Lab 2 Basic Std Cells	31	Feb 1
2 <i>Add Deadline</i>	3 T03 Basic Front End	4 → Proj 1A Std Cells	5 T03 Basic Front End	6 Lab 3 Basic Front End	7	8
9	10 T04 Basic Back End	11 → Proj 1B Front End	12 T04 Basic Back End	13 Lab 4 Basic Back End	14	15
16 <i>Feb Break</i>	17 <i>Feb Break</i>	18	19 Quiz	20 Lab 5 TinyRV2 Xcels	21	22
23 → Proj 2 Proposal	24 T05 Advanced Devices	25 → Proj 1C Back End	26 T06 Advanced Std Cells	27 Lab 6 Advanced Front End	28	Mar 1
2	3 T06 Advanced Std Cells	4 → Proj 2A Software/Test	5 T06 Advanced Std Cells	6 Lab 7 Advanced Back End	7	8
9	10 T07 SRAMs	11 → Proj 2B Xcel Milestone	12 T08 Advanced Front End	13 Lab 8 Automated Flow & SRAM	14	15
16	17 T08 Advanced Front End	18 → Proj 2C Accelerator	19 T09 Advanced Back End	20 Lab 9 Tapeout Push	21	22

23 → Proj 3 Proposal	24 Apple Guest Lecture	25 → Proj 2D Evaluation	26 T10 Chip Flow	27 <i>No Lab</i>	28	29
30 <i>Spring Break</i>	31 <i>Spring Break</i>	Apr 1 <i>Spring Break</i>	2 <i>Spring Break</i>	3 <i>Spring Break</i>	4	5
6	7 Proj 2+3 Meetings	8	9 Proj 2+3 Meetings	10 Proj 2E Tapeout + Report	11	12
13	14 Proj 3 Meetings	15	16 Proj 3 Meetings	17 Proj 3A Baseline Testing	18	19
20	21 Proj 3 Meetings	22	23 Proj 3 Meetings	24 Proj 3B Alternative	25	26
27	28 Proj 3 Meetings	29	30 Proj 3 Meetings	May 1 Proj 3C Evaluation	2	3
4	5 Proj 3 Meetings	6	7 Proj 3 Meetings	8 Project Demos	9	10
11	12	13 Final Report @ noon	14	15	16	17

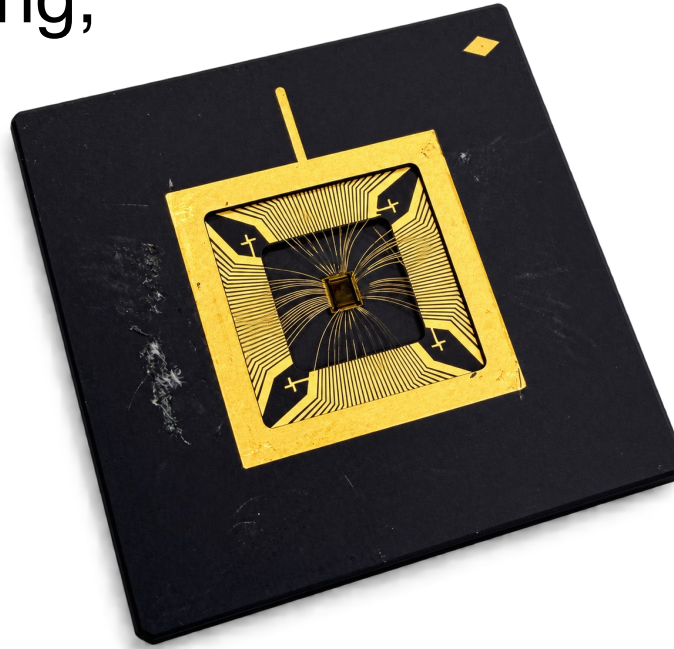
- 47 students
 - 19 seniors
 - 25 Mengs
 - 3 PhDs
- 19 groups leading to one shared Project 1 tapeout and 19 Project 2 tapeouts
- Industry interaction with **Apple** including in-person design reviews, demos, and guest lecture
- Additional funding from **Duffield College of Engineering** for packaging and chip testing equipment

Student Course Evaluations

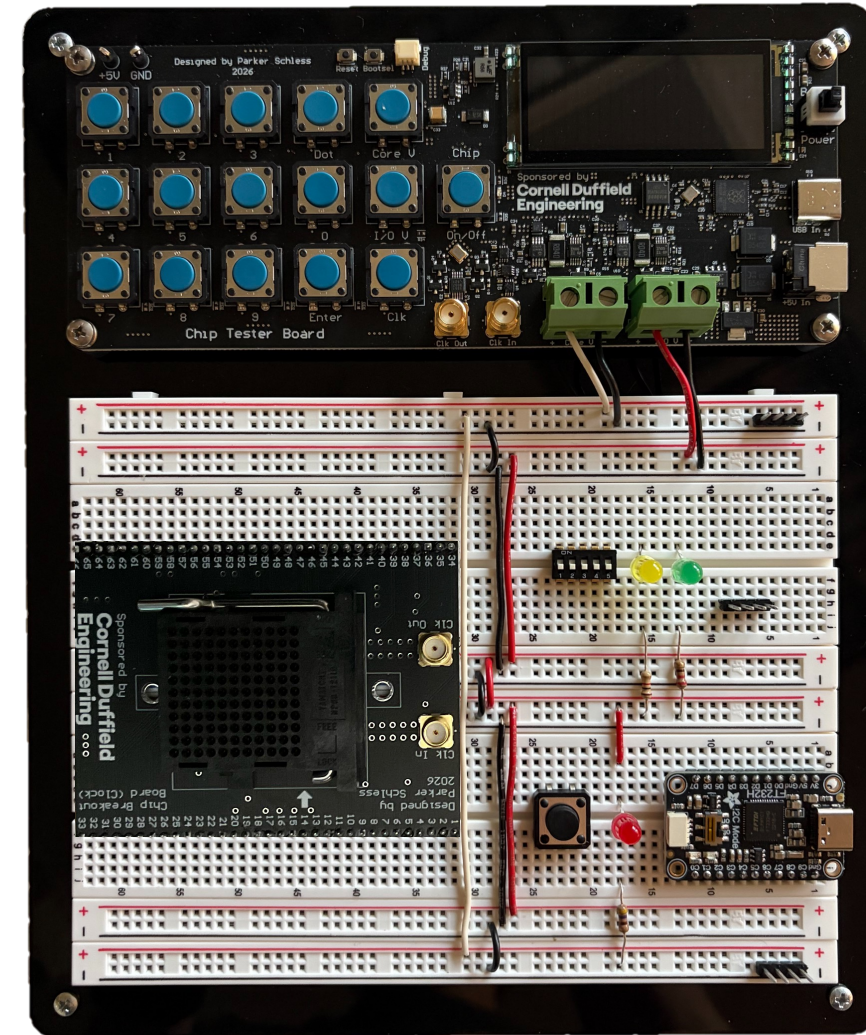
- "This is **one of the courses that made me feel like a degree at Cornell was actually worth the tuition ... I talked about the tapeout project in my interview at Google, and I think it definitely impressed them and helped me land the full time job!**"
- "It absolutely **solidified my drive to pursue a career as a Digital IC Design Engineer focusing on ASIC front-end design and SoC architecture. Having hands-on experience with a complete RTL-to-GDSII flow, especially navigating the "tiny flow" tapeout, is a massive differentiator.** It provides substantial, practical talking points for technical interviews, which is highly attractive to employers and significantly helps in securing full-time hardware engineering roles"
- "I came in with experience working in ASIC, and **I think the course strongly reinforce my interest in the industry. I talked about TinyFlow during my interviews and my interviewers were very interested.** I think it definitely helped me with securing my full-time role at a coherent DSP ASIC startup."
- "**I secured an ASIC internship at one of my dream companies due to this class** and was able to showcase the projects during the interviews. **Prior to this course I didn't even know what an ASIC was** and now I want to pursue designing them because of this course."

ECE 4747/5757 Digital ASIC Chip Testing

- Lectures on I/O, packaging, verification strategies
- Series of two-hour labs
 - Lab 1: FPGA Emulation
 - Lab 2: Power Wiring
 - Lab 3: Clock Testing
 - Lab 4: SPI Testing
 - Lab 5: Accelerator Testing
 - Lab 6: Accelerator Evaluation
 - Lab 7: Tiny Flow Block Testing



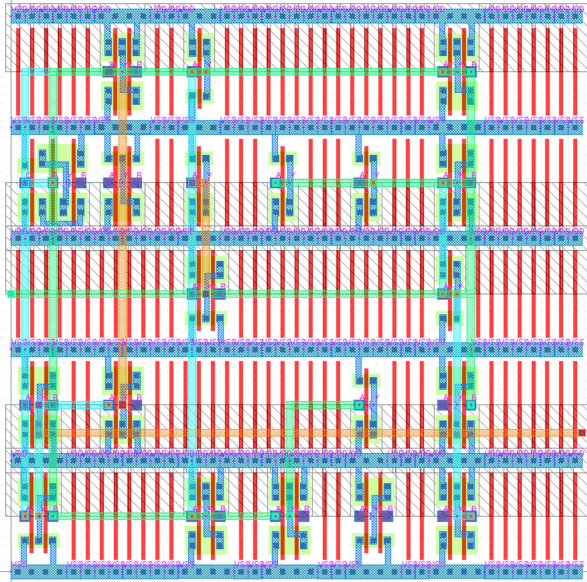
CPGA package with glass lids to enable students to use in-lab microscopes for viewing their chip and taking high-quality photos



Custom chip tester board designed by Parker Schless

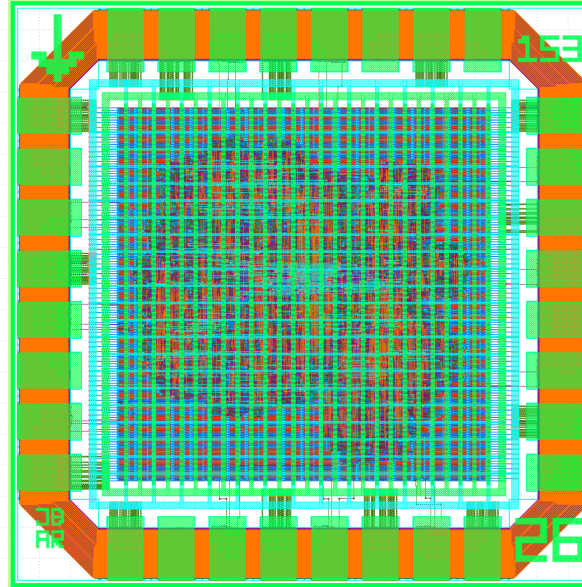
ECE 4745/5745 Digital ASIC Design

Project 1: Tiny Flow Tape-Out (4-5 weeks, 2-3 students/group)



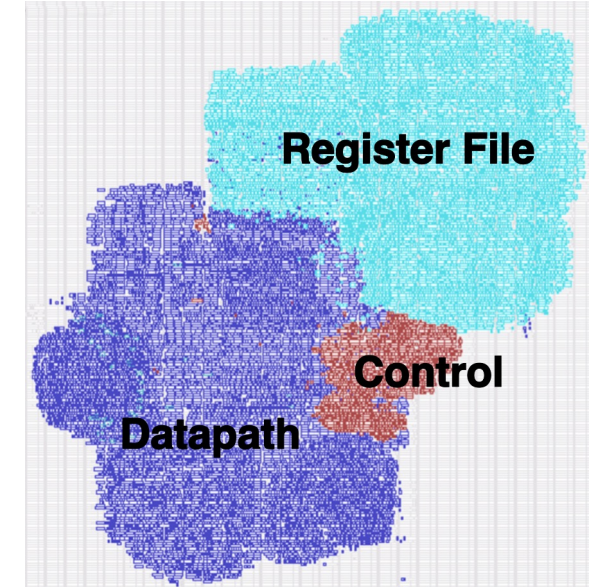
Implement standard cells using lambda rules then very simple Python-based combinational RTL-to-layout flow from scratch; blocks integrated onto shared 1x1mm TSMC 180nm chip

Project 2: Accelerator Tape-Out (4-5 weeks, 2-3 students/group)



Implement accelerator using commercial standard cells and EDA tools; evaluate with RISC-V processor+caches but tapeout just accelerator on 1x1mm TSMC 180nm chip

Project 3: SoC Tape-In (4-5 weeks, 2-3 students/group)



Students free to explore advanced processors, memories, networks and/or accelerators with a strong emphasis on a rigorous comparative analysis of performance, area, and energy