EE 628

Analysis and Design of Integrated Circuits

- Introduction -

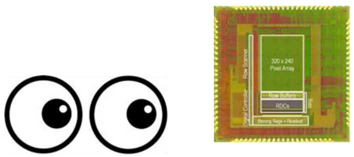
Boris Murmann

bmurmann@hawaii.edu

About Your Instructor

- Born and raised in Germany
- First-generation college graduate
- Spent four years in industry (1990s)
- PhD at UC Berkeley (2003)
- Professor at Stanford (2004-2023)
- Joined UH in Fall 2023
- Research on mixed-signal circuits
  - Data converters
  - Sensor interfaces
  - Embedded machine learning
- Hobbies
  - Soccer, scuba diving
  - Blockchain, digital assets

[Young, ISSCC 2019]

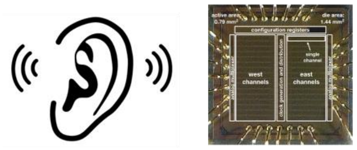


Video Wake-up

[Bankman, ISSCC 2018]

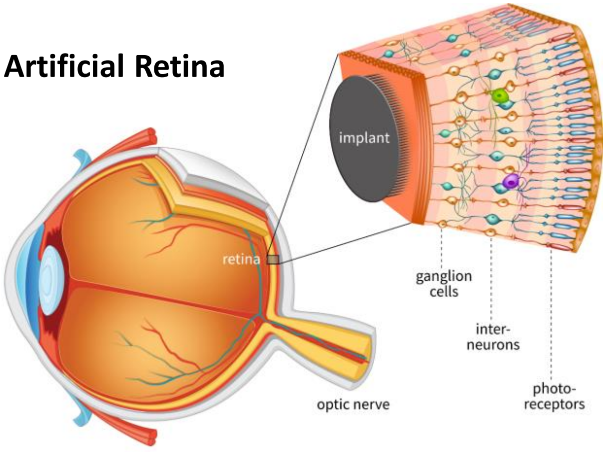
Binary Neural Network Accelerator

[Villamizar, TCAS-I 2021]



Audio Wake-up

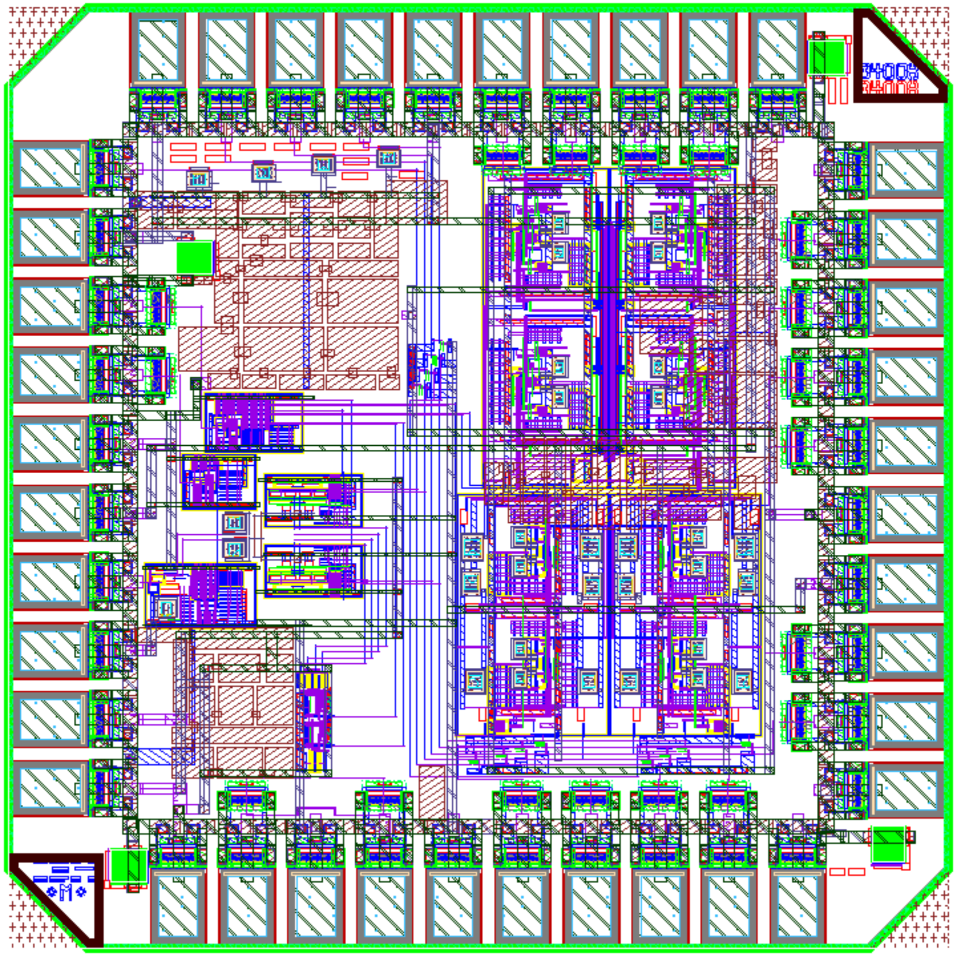
Artificial Retina



What is EE 628?

- Open-ended project course for self-driven students interested in exploring mixed-signal chip design using open-source software tools
- Students work in project teams to generate a database that can (potentially) be sent out for fabrication to a CMOS foundry
- Advanced students may pursue their own design, while less experienced students can follow a template project
- Learning goal is to become familiar with chip design flow
  - Concept development
  - Circuit analysis, design and simulation
  - Layout, and verification (DRC, LVS and post-layout checks)

Example



RF Front-End Receiver for ISM-900M
https://www.ee.columbia.edu/~kinget/EE6350_S16/01_DCRRX_Hao_Tuo/Layout.html

Similar Course at Columbia University

Could be done
through EE 699 units

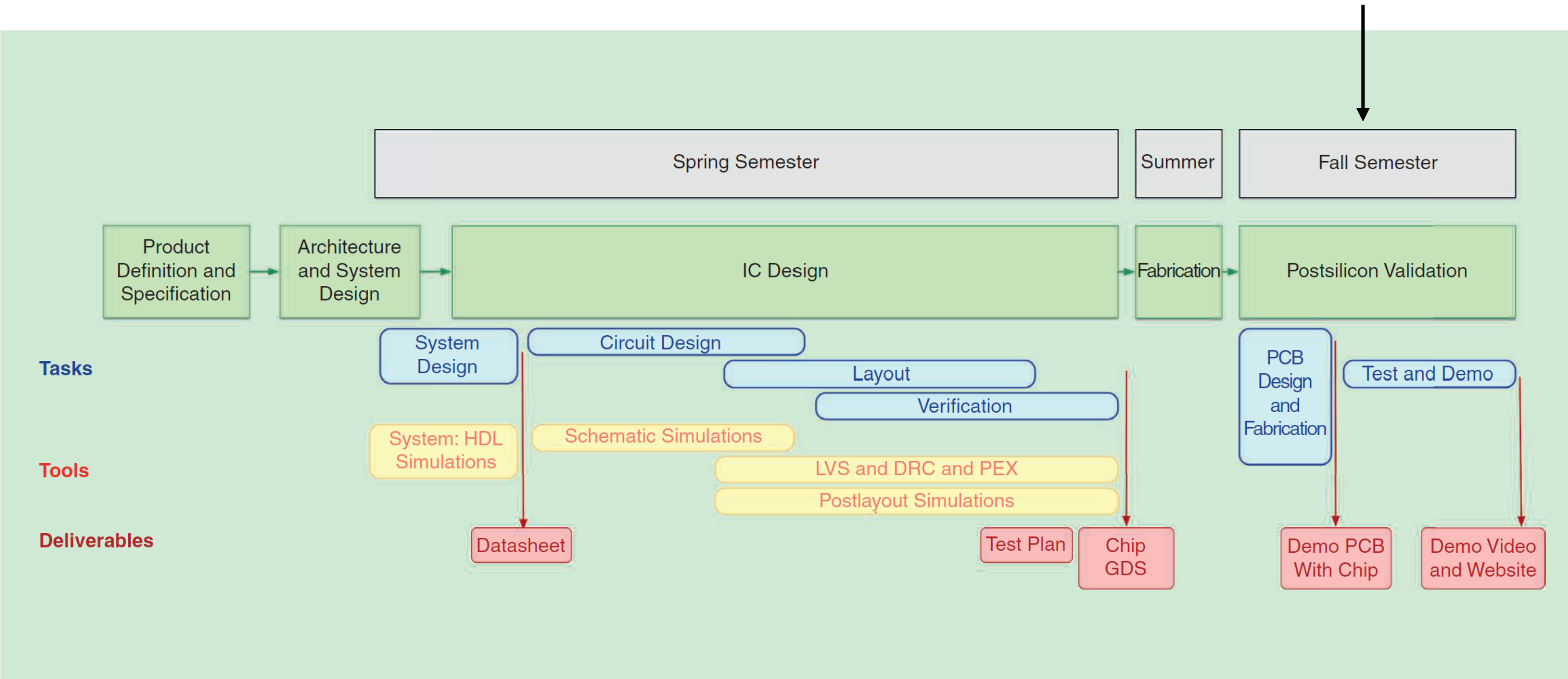
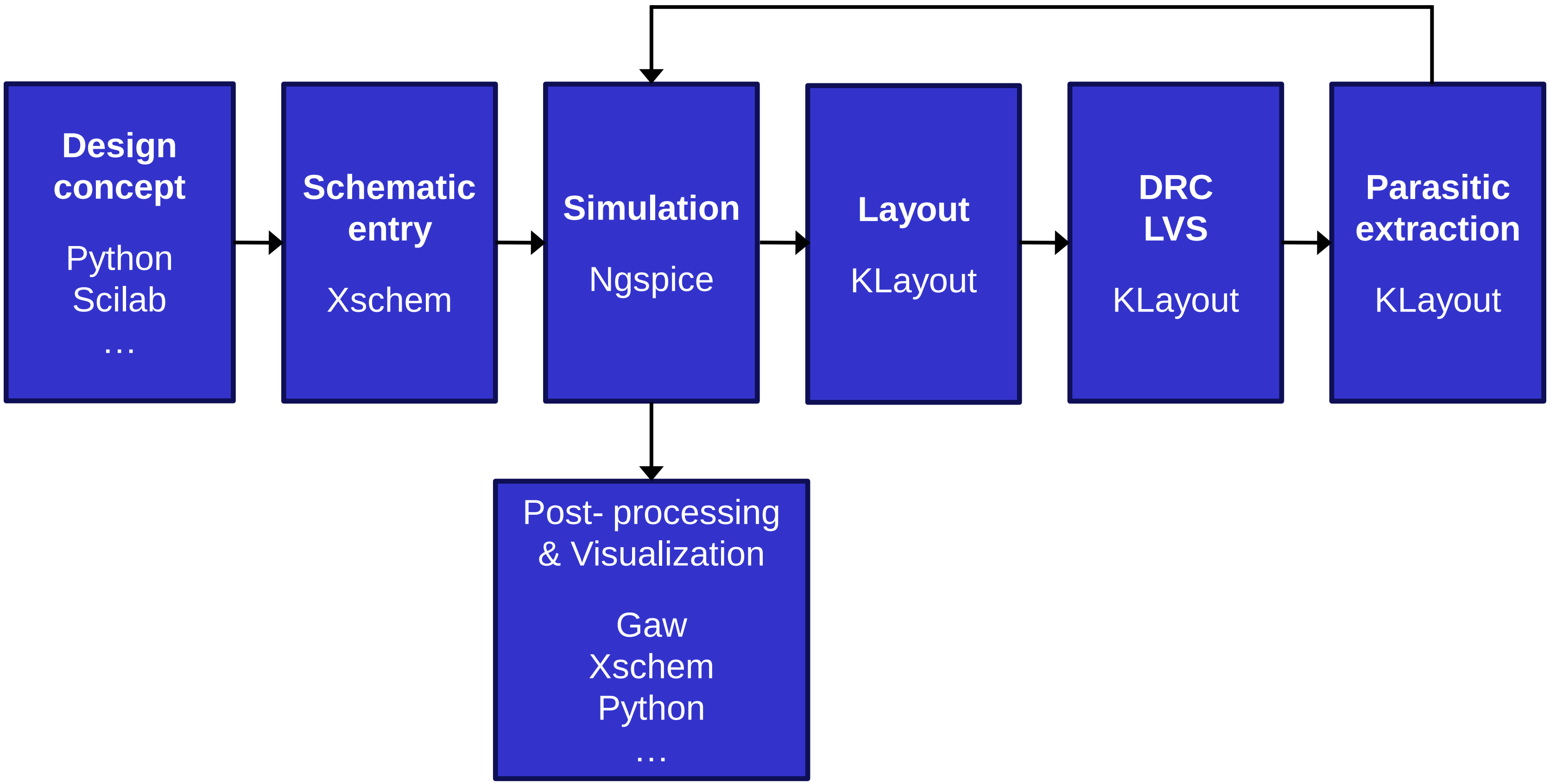


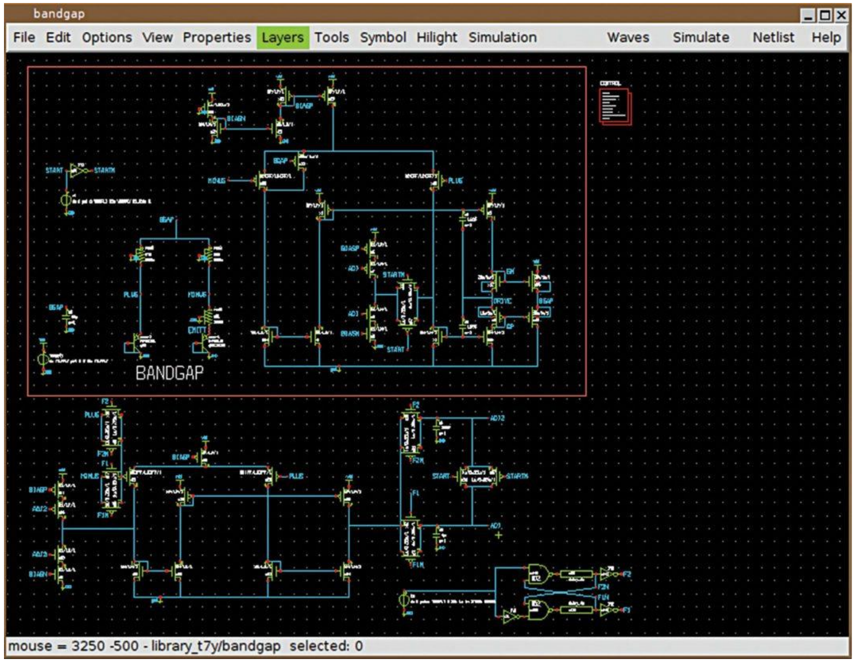
FIGURE 1: The timeline of the VLSI Design Lab including a breakdown of the tasks, tools used, and deliverables produced.

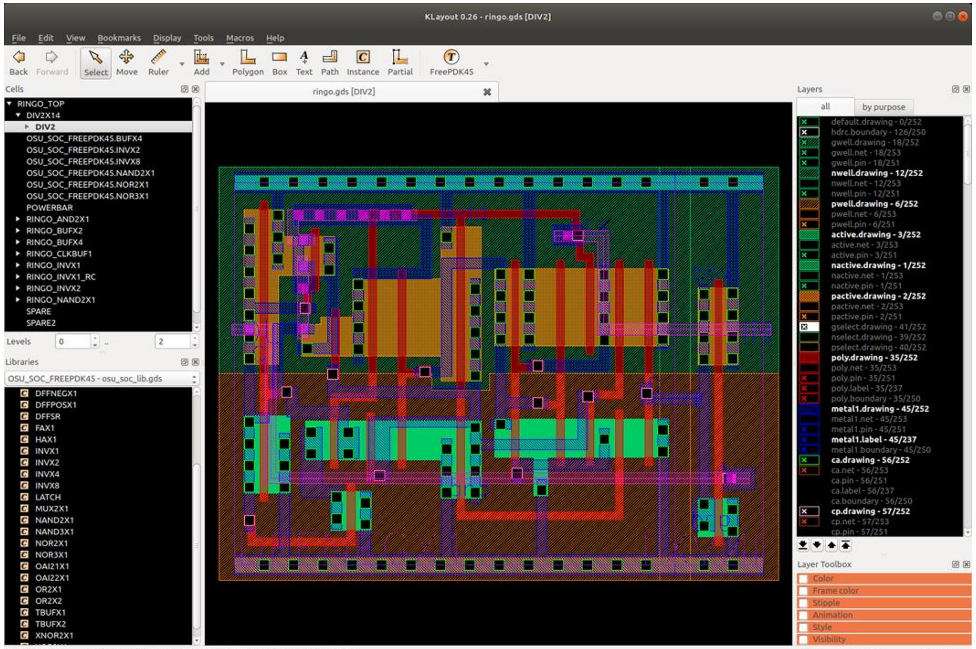
EE 628 Open-Source (Analog) Design Flow



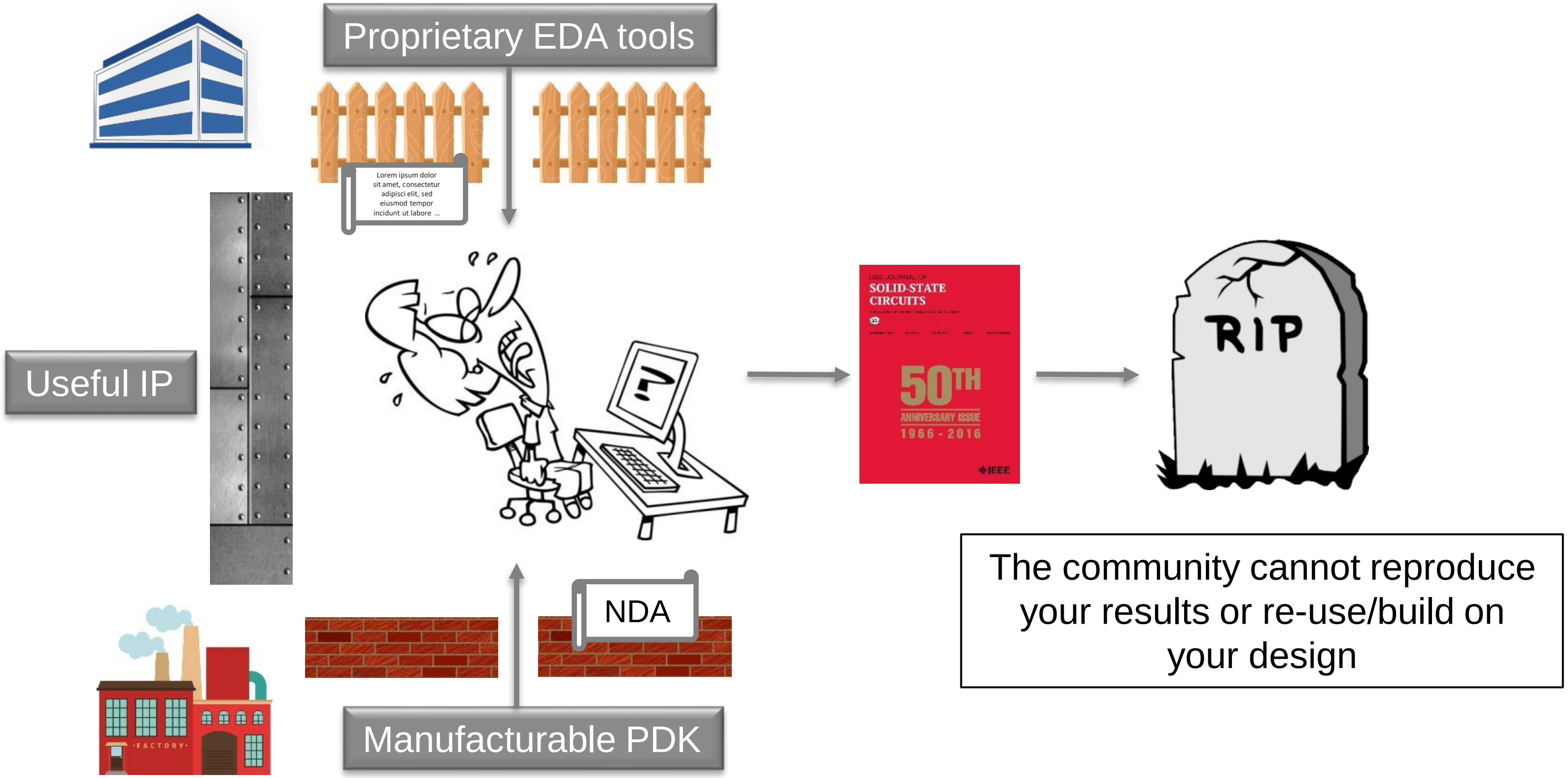
Interested students may explore the use of OpenRoad for digital design

Xschem & KLayout





Why Open Source?



EDA = Electronic Design Automation
PDK = Process Design Kit
NDA = Nondisclosure agreement
IP = Intellectual Property

Big-Bang Events: Open-Source PDKs

- First open-source PDK (November 2020)
  - SkyWater 130nm CMOS
  - https://github.com/google/skywater-pdk

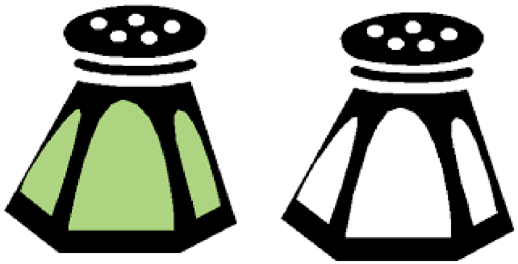
- Second open-source PDK (October 2022)
  - GlobalFoundries 180nm MCU
  - https://github.com/google/gf180mcu-pdk

- Third open-source PDK (March 2023)
  - IHP 130nm BiCMOS
  - https://github.com/IHP-GmbH/IHP-Open-PDK
  - **We will use this technology in EE 628**

Open Source in a Nutshell

- Core principles
  - Open exchange, collaboration, transparency, meritocracy
- Typical benefits
  - Improves productivity, managing complexity
  - Enables community review and steady improvements, re-use
  - Promotes education and tinkering
- Open source does not imply "free"
  - Can make money with open-source products
    - Red Hat, Ruby on Rails, …
  - Proper terminology
    - Proprietary vs. open source (NOT: commercial vs. open source)

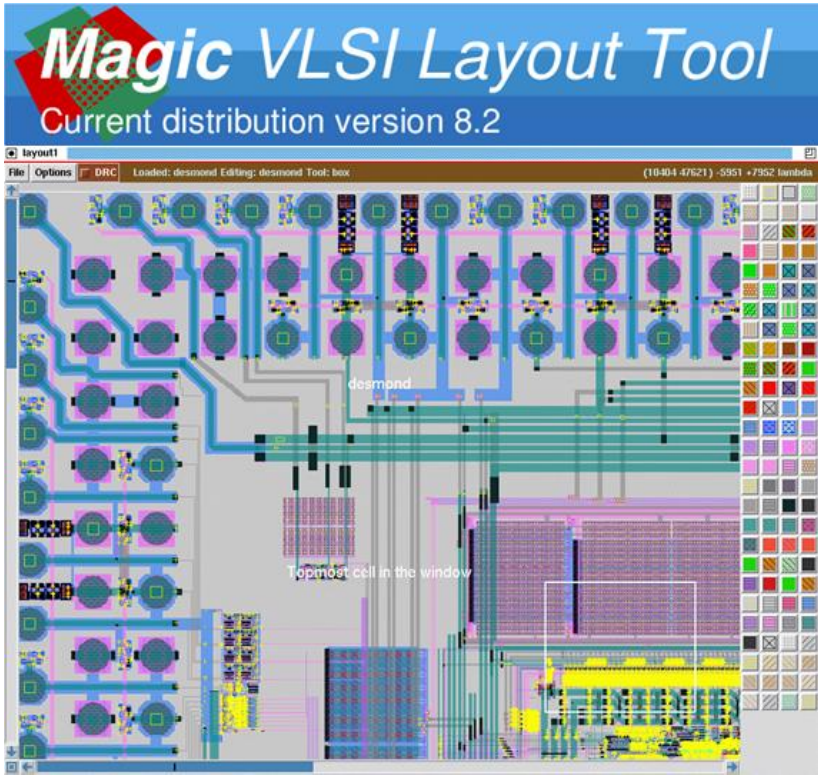
Open Source is in Our DNA!



SPICE (Simulation Program with Integrated Circuit Emphasis)

Laurence W. Nagel and D.O. Pederson

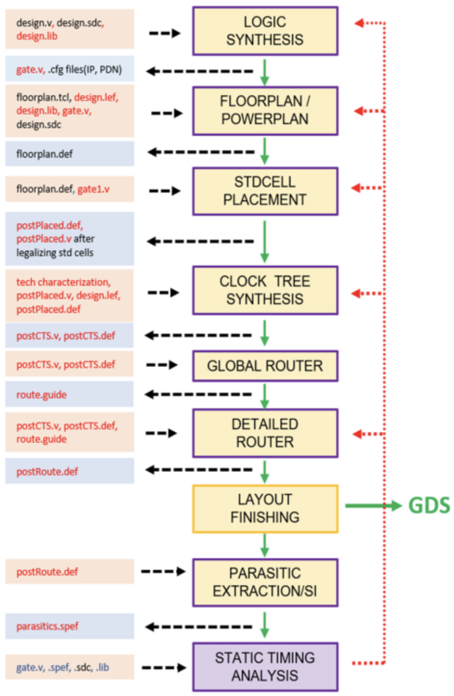
EECS Department
University of California, Berkeley
Technical Report No. UCB/ERL M382
April 1973

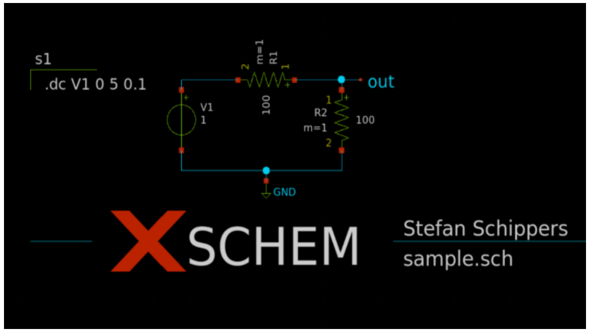


Sources: http://www.omega-enterprises.net, http://opencircuitdesign.com/magic

Examples of Today's Open-Source EDA Tools

OpenROAD





NGSPICE

Magic VLSI Layout Tool

Current distribution version 8.2

Sandia National Laboratories

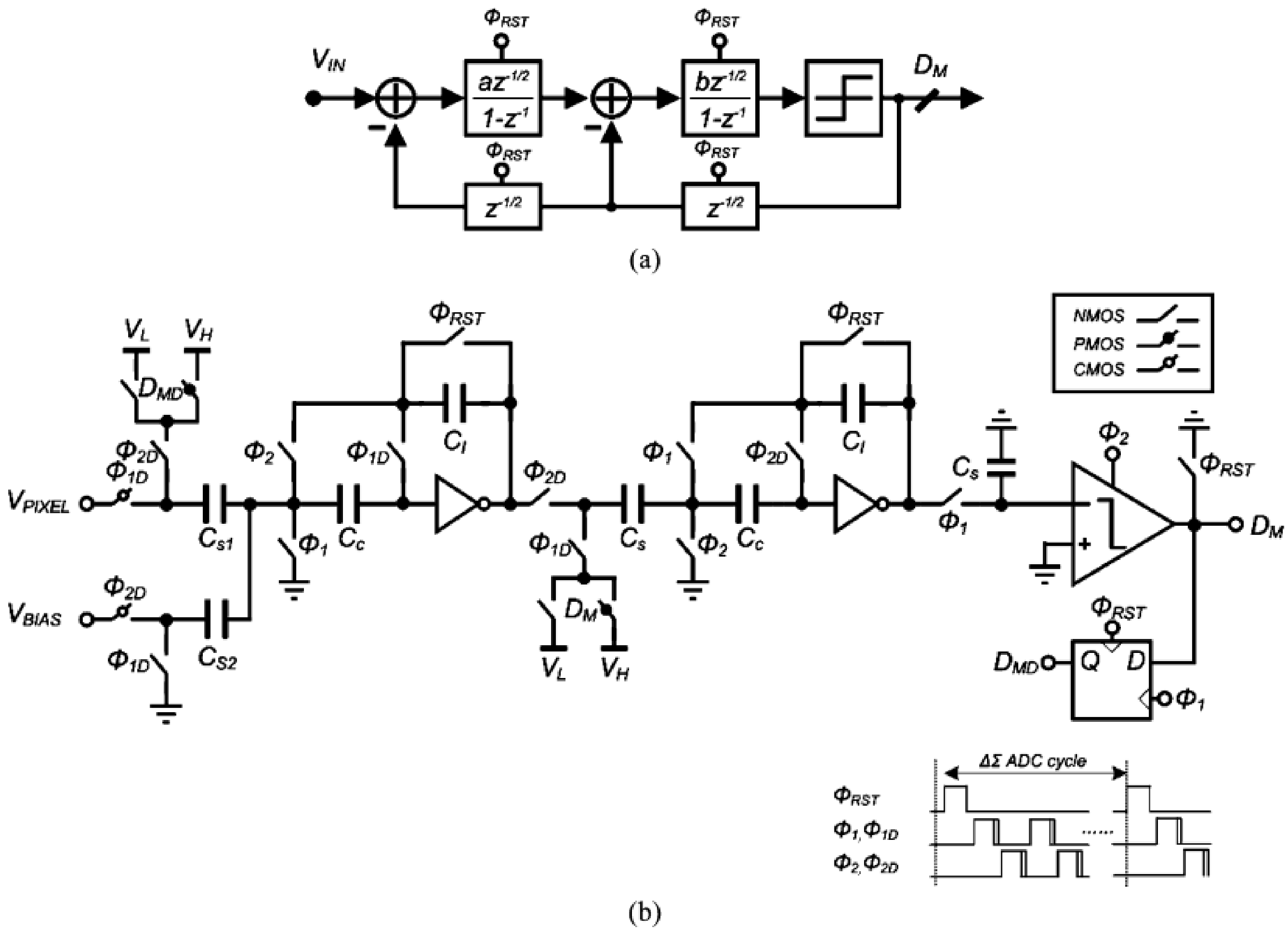
Xyce

Parallel electronic simulation

KLayout

GDS_FACTORY

Template Project: Incremental Delta-Sigma A/D Converter



Y. Chae et al., "A 2.1 M Pixels, 120 Frame/s CMOS Image Sensor With Column-Parallel ADC Architecture," in IEEE Journal of Solid-State Circuits, Jan. 2011. https://ieeexplore.ieee.org/document/5641589

First Implementation (1977)

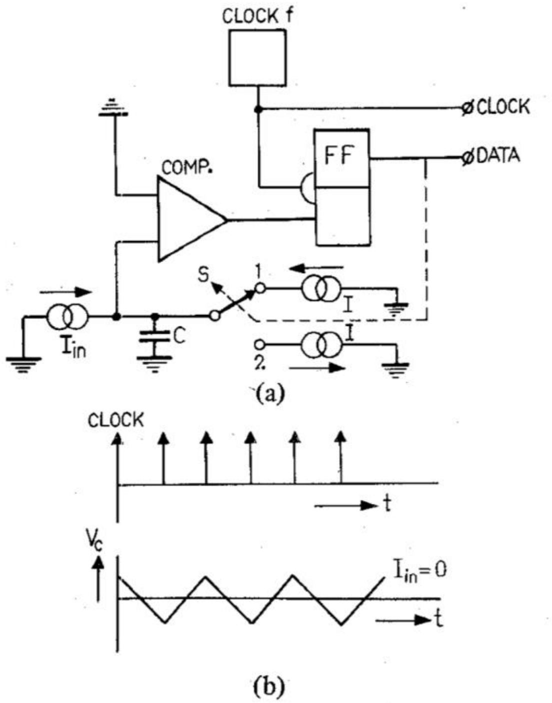


Fig. 1. (a) Basic sigma-delta modulator. (b) Pulse patterns as a function of time.

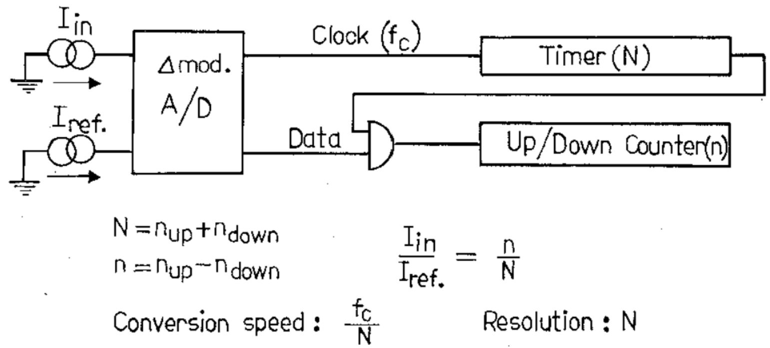
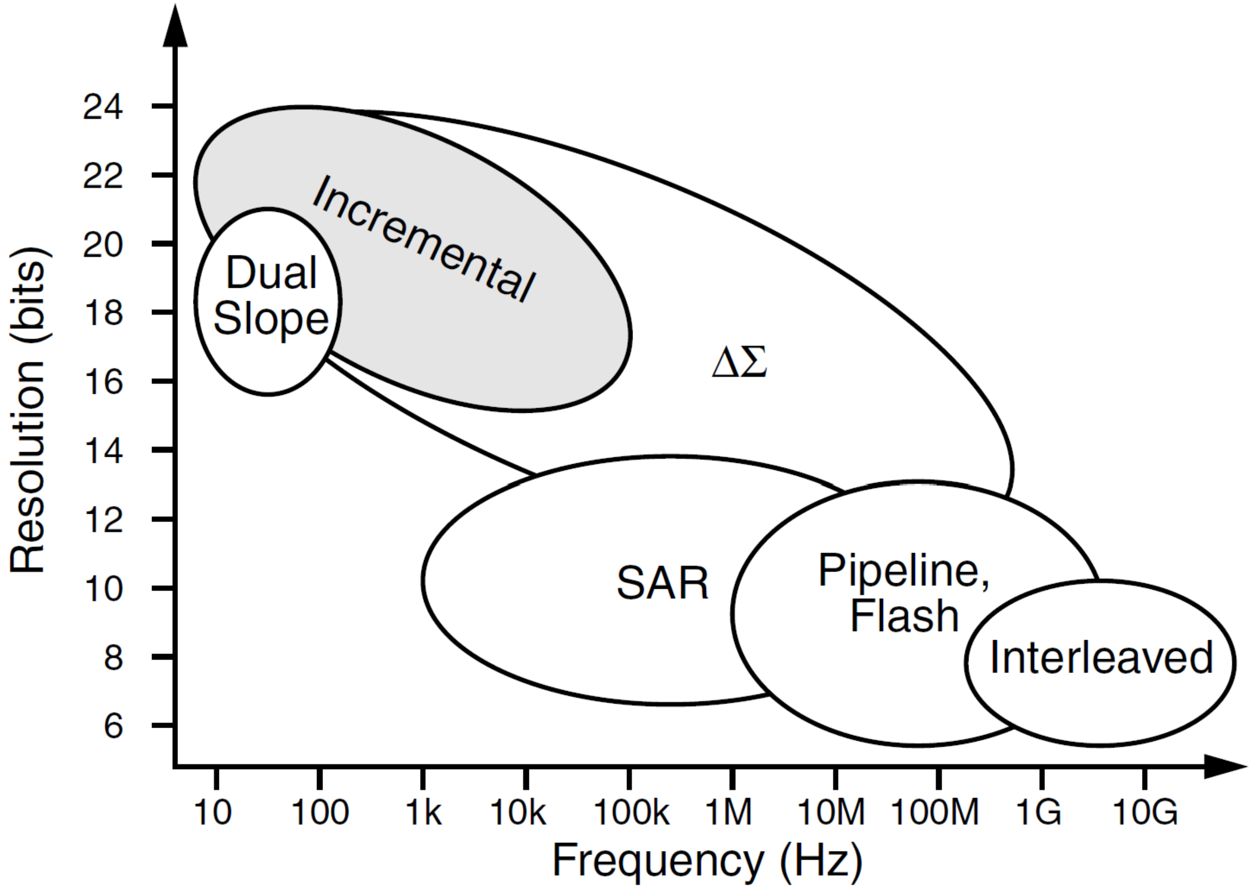


Fig. 3. Digital controller circuit.

R. van de Plassche and R. E. J. van Der Grift, "A five-digit analog-digital converter," in IEEE Journal of Solid-State Circuits, Dec. 1977. https://ieeexplore.ieee.org/document/1050975

A/D Converter Architectures



Shanthi Pavan; Richard Schreier; Gabor C. Temes, "Incremental Analog-to-Digital Converters," in Understanding Delta-Sigma Data Converters, pp.407-423 (Chapter 12). https://ieeexplore.ieee.org/document/7906298.

Lots of Interesting Things to Learn

How does the ideal model work?

Correlated double sampling

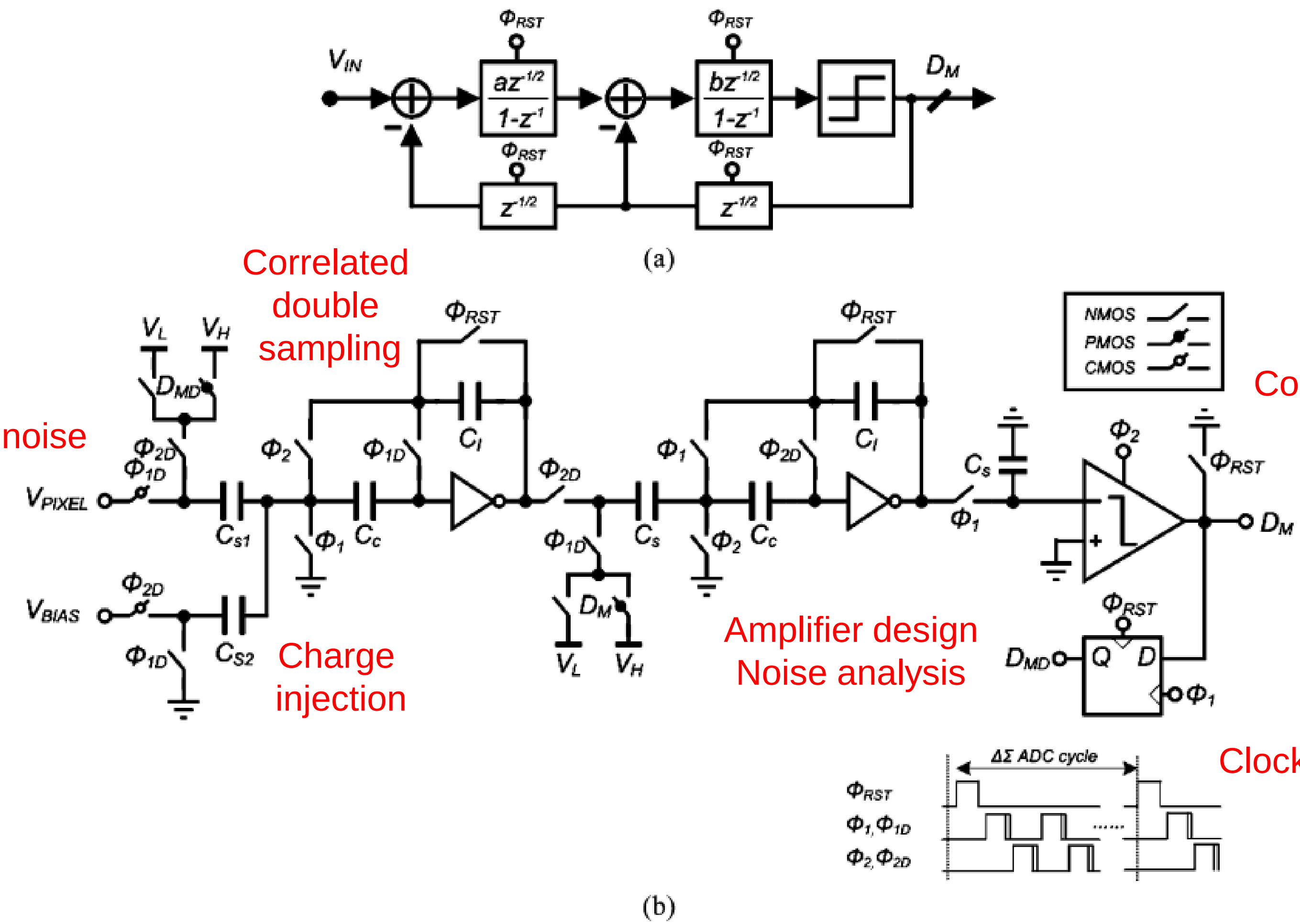
kT/C noise

Comparator design

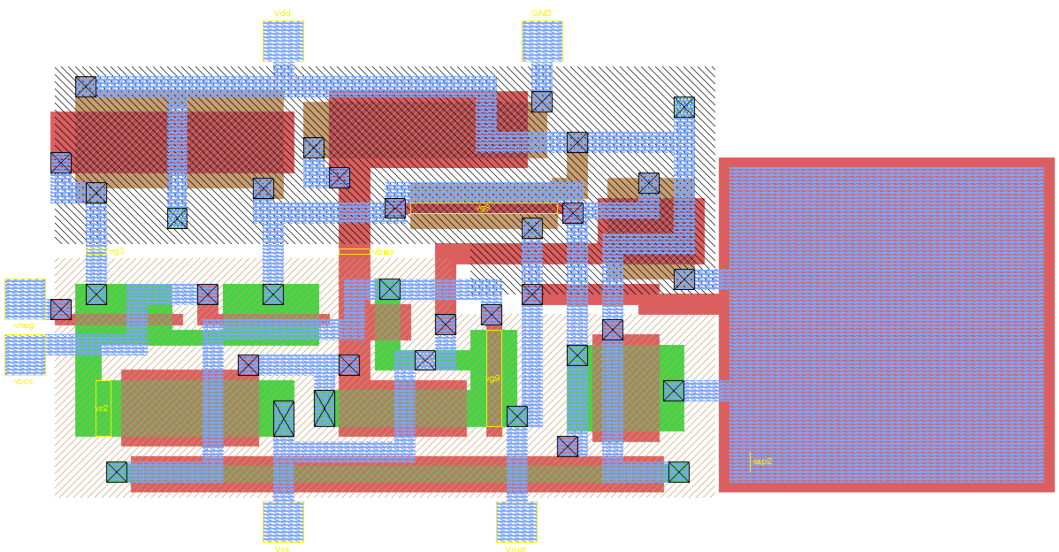
Amplifier design
Noise analysis

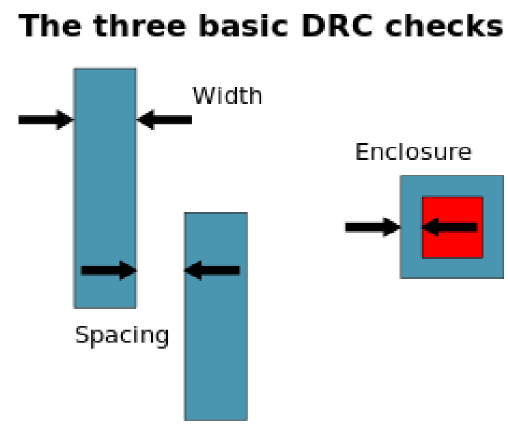
Charge injection

Clock generation



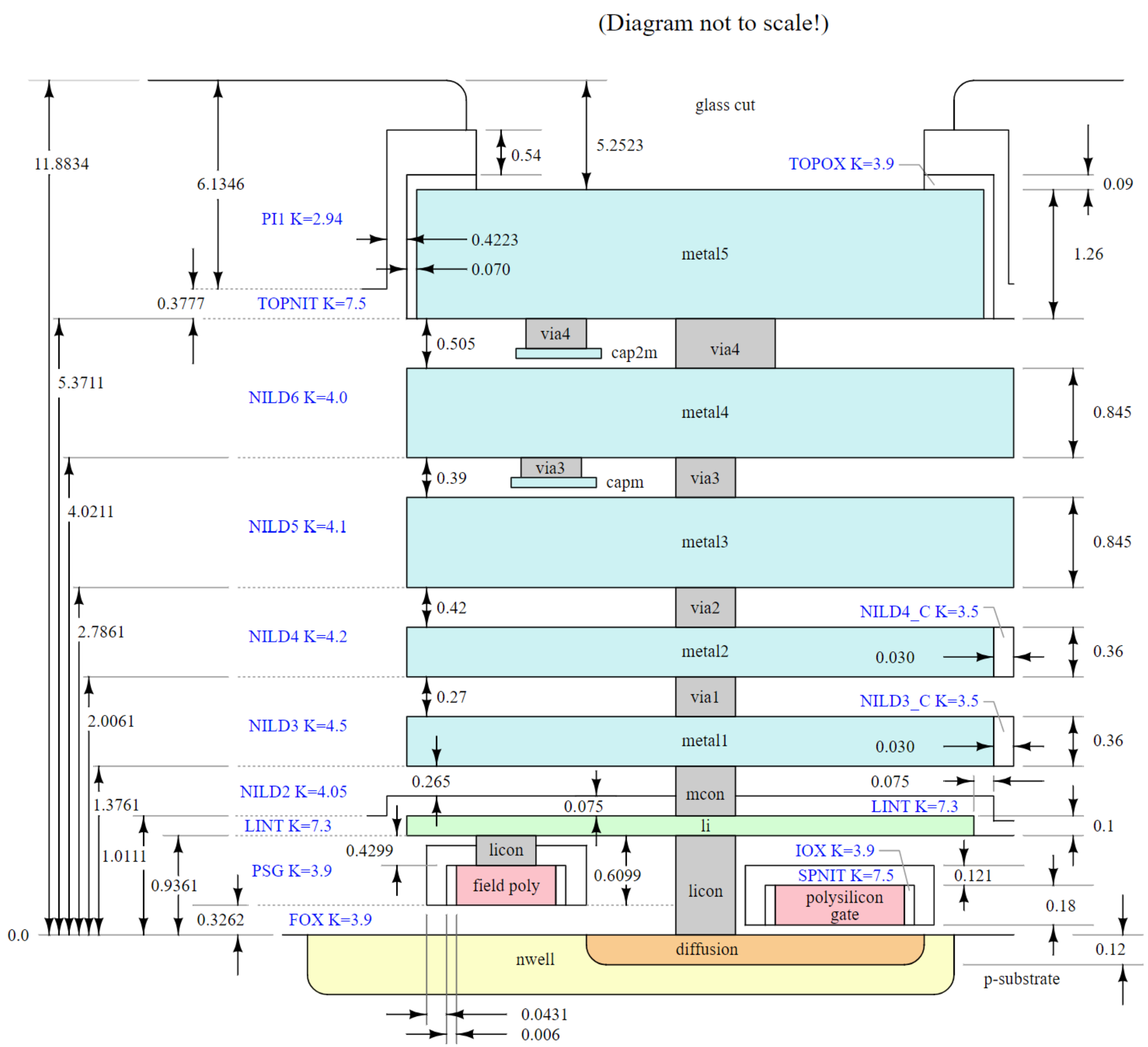
Also Need to Learn Integrated Circuit Layout (Using KLayout)





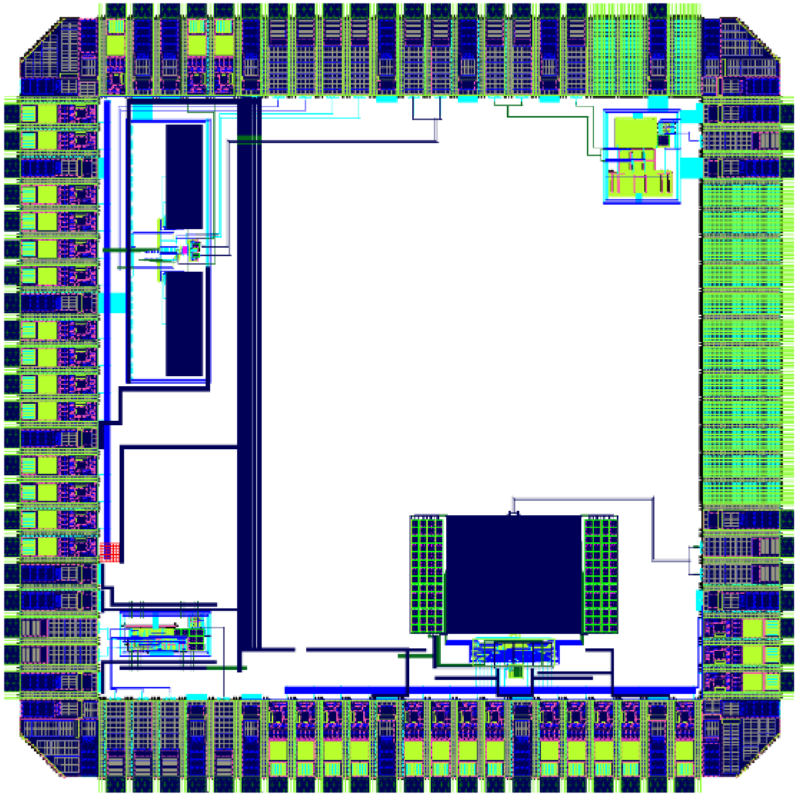
https://en.wikipedia.org/wiki/Integrated_circuit_layout
https://en.wikipedia.org/wiki/Design_rule_checking

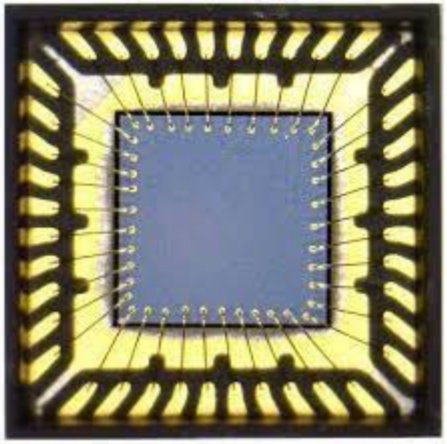
Cross Section (SKY130 Process)



Designer can only manipulate x & y dimensions, z dimensions are fixed

Padring & Wirebond Package Example





Prerequisites

- EE 323 (ideally EE 326)
- Prior exposure to analog circuit design and transistor modeling
- Basic familiarity with analog circuit simulation (LTspice, PSpice, etc.)
- Working knowledge of Laplace and z-transforms
- Basic coding (ideally Python) and Linux commands

- Time and interest in the subject!

(Rough) Course Outline

- High-level analysis and simulation of the template ADC
  - Using Scilab, Simulink, etc.
- Build and simulate the idealized spice-level circuit
  - Using ideal switches and controlled sources (no transistors)
- Build, analyze and simulate the components (with transistors)
  - Switches, integrator, comparator, clock generator
- Midsemester design review
  - Team presentation (3-4 students)
- Assemble the complete circuit
  - Insert components one by one and verify operation
- Layout, DRC, LVS and parasitic extraction
  - First using a trivial example, then for the designed blocks & chip level
- Final design review
  - Team presentation; high-quality designs will be considered for tapeout

Lecture Structure

<u>Classical lecture material</u>
Circuit design
Circuit simulation
Analysis of nonidealities
Technology aspects
Layout basics
…

<u>Demo & Discussion Time</u>
Logistics
Tool demos
Troubleshooting
Student presentations
…

Assignments

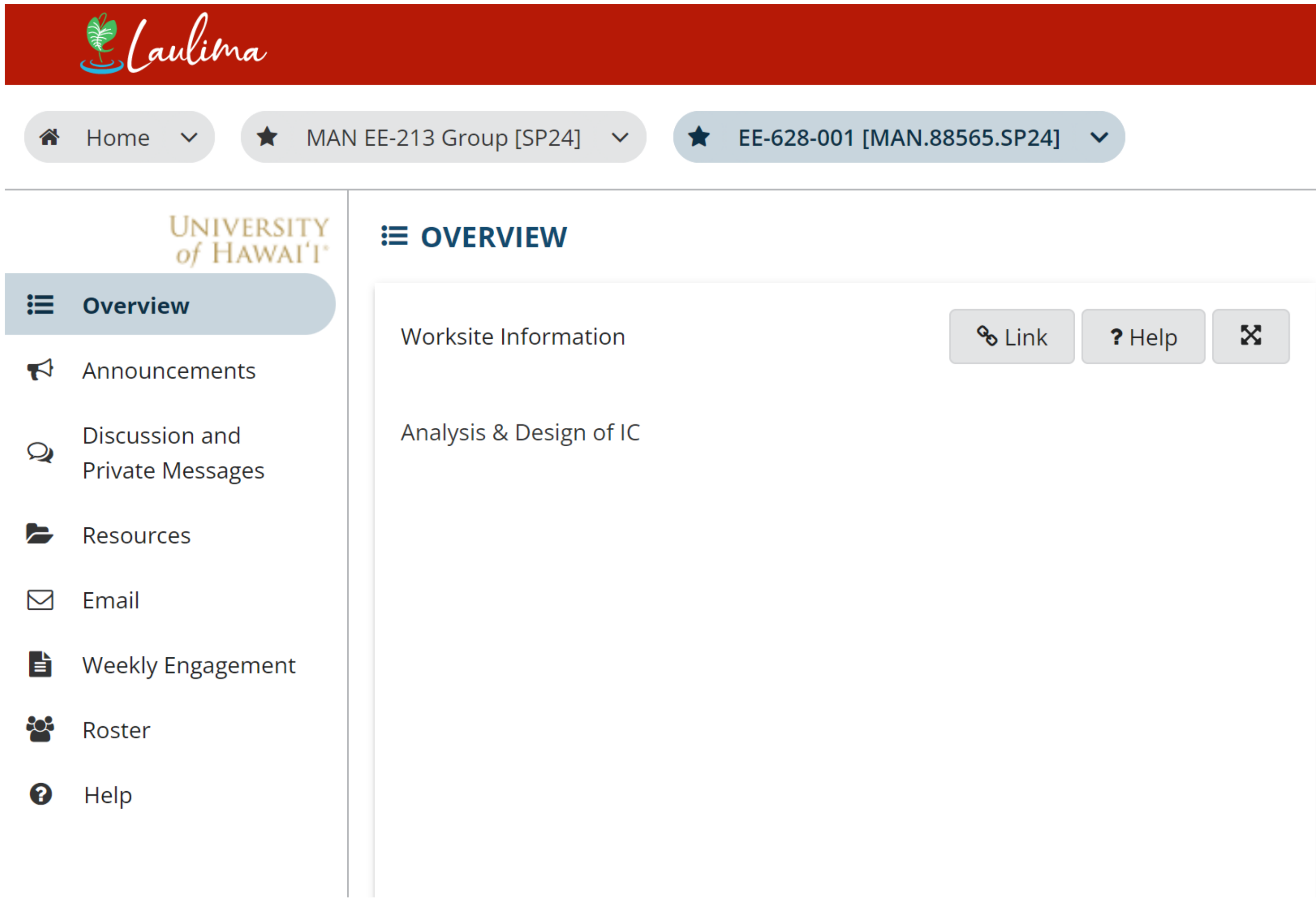
- Individual effort
  - Weekly reports (20%)
    - Submit notes that you take while working on weekly assignments (posted on GitHub site)
    - You don’t need to complete the weekly assignments completely to get full credit; just do what you can and document in your report

- Team effort
  - Midsemester design review (20%)
  - Final design review (20%)
  - Final report (40%)

Laulima Site: EE-628-001 [MAN.88565.SP24]

Syllabus

Submit weekly reports here



GitHub Site: https://github.com/bmurmann/EE628/

bmurmann / EE628 Public

Notifications Fork 0

Code Issues Pull requests Actions Projects Security Insights

Files

main

Go to file

- 1_Schedule
- 2_Assignments
  - README.md
- 3_Tools
- 4_Technology
- 5_References
- LICENSE
- README.md

EE628 / 2_Assignments /

bmurmann Update README.md d9e(

Name	Last commit message
..	
README.md	Update README.md

README.md

Week 1

- Take notes as you work through the items below and submit them as your weekly report. The report does not need to be pretty, but it should document what you have done, what worked/didn't, things you don't understand etc. You can also suggest topics that should be covered in future lectures.
- Read the following two papers:
  https://ieeexplore.ieee.org/document/10224621
  https://ieeexplore.ieee.org/document/9805608
- Skim through the following paper, which describes the A/D converter that we'll use as our template project. You won't understand everything; we'll work things out in the coming weeks!
  https://ieeexplore.ieee.org/document/5641589
- Have a look at Raymond Yang and Yaqing Xia's Stanford EE 372 project at the link below. This will give you a feel for what your project deliverables will look like (GitHub repo, presentations, reports, etc.). Note that their circuit is also similar to our template project (albeit a little more complex).
  https://priyanka-raina.github.io/ee372-spring2022/
- Think about your project topic. Do you have a specific idea or do you want to follow the template project?
- Network with your peers and join a team of 3-4 students. Of course, each team must work on the same project topic. The idea is to split the work and help each other toward delivering one common chip design per team.

We Need a Discussion Forum

- Laulima?
- Slack?
- Discord?
- …