

UNIC-CASS Design-to-Tapeout Meetup #2 (Aug. 28, 2024)

Digital and analog design flows

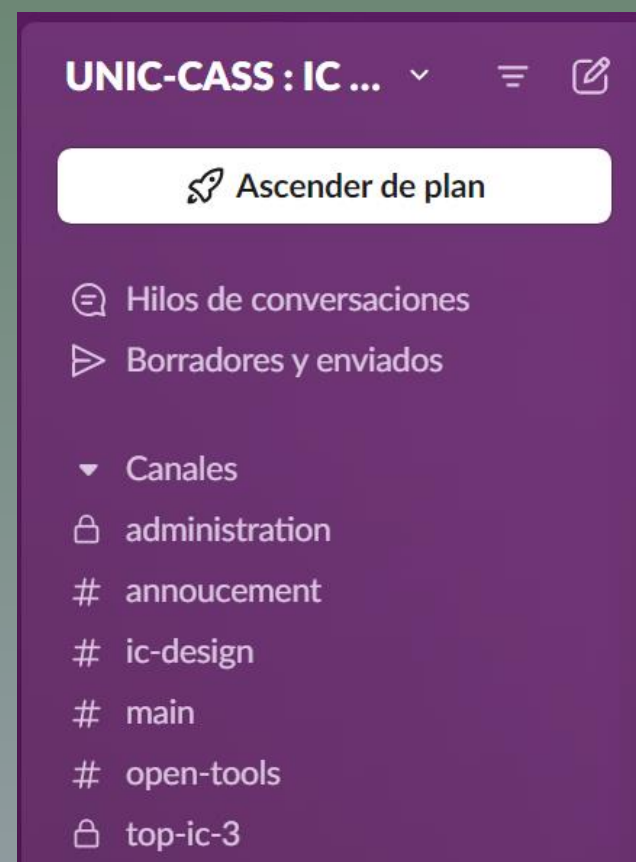
Presented by: D-to-T mentoring team



Design-to-Tapeout Mentoring

Communication and organization

- ▶ Bi-weekly mentoring sessions
- ▶ Slack Channel
- ▶ UNIC-CASS website



UNIC-CASS Slack channel:
<https://unic-cass.slack.com/>

A screenshot of the UNIC-CASS website. The header shows 'Universalization of IC Design From CASS (UNIC-CASS)'. The main heading is 'Universalization of IC Design from CASS (UNIC-CASS)'. Below the heading is a paragraph describing the program: 'The Universalization of IC Design from CASS (UNIC-CASS) program is a structured end-to-end Integrated Circuit (IC) design-to-test experiential learning. The program aims to improve the know-how and accessibility to IC Design technologies for enthusiasts and design communities worldwide in low-to-middle-income and/or low-opportunity countries. This program is of strategic cooperation with the Solid-State Circuits Society serving geographical-complementing locations.' Below the paragraph is a link: 'UNIC-CASS Mentoring Session Recordings & Presentations'. The main content area contains a paragraph: 'The IEEE Circuits and Systems Society (CASS) invites you to participate in the Universalization of IC Design from CASS (UNIC-CASS) program, a structured end-to-end Integrated Circuit (IC) design-to-test experiential learning. The program aims to improve the know-how and accessibility to IC Design technologies for enthusiasts and design communities worldwide in low-to-middle-income and/or low-opportunity countries. This program is of strategic cooperation with the Solid-State Circuits Society serving geographical-complementing locations.' Below this paragraph is another paragraph: 'There are no restrictions to designing your ideal chip on design complexity or type (digital, analog/RF, or mixed-signal). Based on the quality of their submitted chip design proposals, selected participants will get the opportunity to learn from carefully curated materials on the CASS MiLe platform, hands-on design mentoring by experts in the field, submit designs to CASS-sponsored fabrication runs via Efabless chipIgnite program and test your fabricated chip at selected testing facilities.' Below the paragraphs is a section titled 'UNIC-CASS Materials & Tools' with four buttons: 'GitHub', 'UNIC-CASS Slack Channel', 'CASS Microlearning', and 'UNIC-CASS Training Materials (GitHub)'. Below these buttons is a link: 'UNIC-CASS Mentoring Session Recordings & Presentations'.

Recordings and presentations: <https://ieeecas.org/unic-cass-mentoring-session-recordings-presentations>

UNIC-CASS educational material



Universalization of IC Design from CASS (UNIC-CASS)

9 followers <https://ieee-cas.org/universalization-i...>

Source: <https://github.com/unic-cass/unic-cass.github.io>

- New videos made by volunteers following the lecture notes
- Lecture notes in Markdown format
 - ⇒ You can contribute by updating them
- New document theme using Just-the-docs Jekyll theme
- Latest materials in Github
- Materials in CASS-MiLe will be updated into v2.0 soon
- Add instructions for Klayout in analog design flow

- Layout
- DRC
- LVS

The screenshot shows the website's navigation menu on the left with items like 'Home', 'Course introduction', '1.1 Introduction to the course', '1.2 Introduction to PDKs', '1.3 Introduction to the Digital Design Flow', '1.4 Introduction to Analog Design Flow with opensource tools', 'Environment Setup', 'Analog Design Flow', 'Digital Design Flow', 'Preparing the design for tapeout', and 'Design examples'. The main content area displays '1. Course Introduction' and a 'TABLE OF CONTENTS' with links to '1.1 Introduction to the course', '1.2 Introduction to PDKs', '1.3 Introduction to the Digital Design Flow', and '1.4 Introduction to Analog Design Flow with opensource tools'. A copyright notice for 2024 IEEE CASS and a link to 'Edit this page on GitHub' are also visible.

Website: <https://unic-cass.github.io>

The screenshot shows the CASS-MiLe website with a search bar and a grid of seven course cards. The cards are: UNICASS 1 - Course Introduction; UNICASS 2 - Design Tool Installation & Working Environment setup; UNICASS 3 - Analog design flow with opensource tools; UNICASS 4 - Digital design flows with Opensource tools; UNICASS 5 - Functional Verification (In Depth); UNICASS 6 - Preparing the design for tapeout (guided by Efabless); and UNICASS 7 - Design examples. Each card features a background image related to the course topic.

CASS-MiLe: <https://ml.ieee-cas.org>

Design-to-Tapeout Mentoring Team 2024



**Gabriel Maranhão,
UFSC (Brasil)**



**Deni Alves
UGA (France)/UFSC (Brasil)**



**Francisco Brito
UFERSA (Brasil)**



**Duy-Hieu Bui, VNU
Hanoi (Vietnam)**



**Rodrigo Wuerdig
KUL (Belgium)/UFRGS (Brasil)**



**Jorge Marin
UTFSM (Chile)**



**Sergio Bampi
UFRGS (Brasil)**

Main Design-to-tapeout milestones

- ▶ **First design review (mock tapeout)**
 - Purpose: Prepare the teams in advance for the final tape-out phase
 - Date: Last week of September
- ▶ **Second design review (first DRC/LVS clean design delivery)**
 - Purpose: Verify feasibility of designs and top integration
 - Date: Second week of October
- ▶ **UNIC-CASS tapeout**
 - Purpose: final delivery of top designs
 - Date: November 8th

Design-to-Tapeout Mentoring

Weekly mentoring sessions 1

| Activities | Date/ Deadline | Purpose/ comments | Your Deliverables (provide info/links on your GitHub) | We will provide/handle |
|--|------------------------------------|---|---|--|
| 1st Meet-up | Aug 12, 1pm-2pm UTC | Kick-off meeting | N/A | N/A |
| Weekly meet-ups | 21/8,28/8,4/9, 18/9 1pm-2pm UTC | Mentoring and Design groups to present progress. Short presentations on topics such as Caravel/Caravan, best practices, pitfalls to avoid. | N/A | Slides and videos will be posted on the UNI-CASS website |
| 1st Design submission [Mock tape-out] | Sep 25 | Prepare the teams in advance for the final tape-out phase | Passing pre-check report | 1. Detailed instructions for pre-check 2. Example files |
| 1st Design Review & Feedback | Oct 2 | Provide feedback and advice for the next stages | N/A | Feedback on the pre-check work |
| Weekly meet-ups | 16/10, 30/10, 5/11 1pm-2pm UTC | Focused mentoring + preparation for final tape-out + evaluate status of potential drop-outs | N/A | Slides and videos will be posted on the UNI-CASS website |

Design-to-Tapeout Mentoring

Weekly mentoring sessions 2

| Activities | Date/ Deadline | Purpose/ comments | Your Deliverables (provide info/links on your GitHub) | We will provide/handle |
|--|-------------------|--|--|---|
| 2nd Design submission | | Verify feasibility of designs and top integration Oct 16 | Detailed circuit description (pinout, area and functionality) + DRC/LVS reports on Efabless platform | Check reports to confirm all blocks are LVS and DRC clean. |
| 2nd Design Review & Feedback from mentors | | Provide feedback and advice for the final stages Oct 23 | N/A | Feedback on the individual designs + advice for top merge |
| Efabless pre-check + tapeout | | Complete tape-out work on Efabless platform Nov 8 | Deliver COMPLETE Caravel/Caravan drop-ins to Efabless final check | Provide assistance to tape-out leaders within the teams |
| Tape-out | | See: Nov 11 https://efabless.com/chipignite | N/A | List of winner teams included in this tapeout List of unfinished designs to candidate for 2025 UNIC-CASS program |
| Chip delivery date | | See: Apr 11 https://efabless.com/chipignite | N/A | Chip shipping and testing logistics |

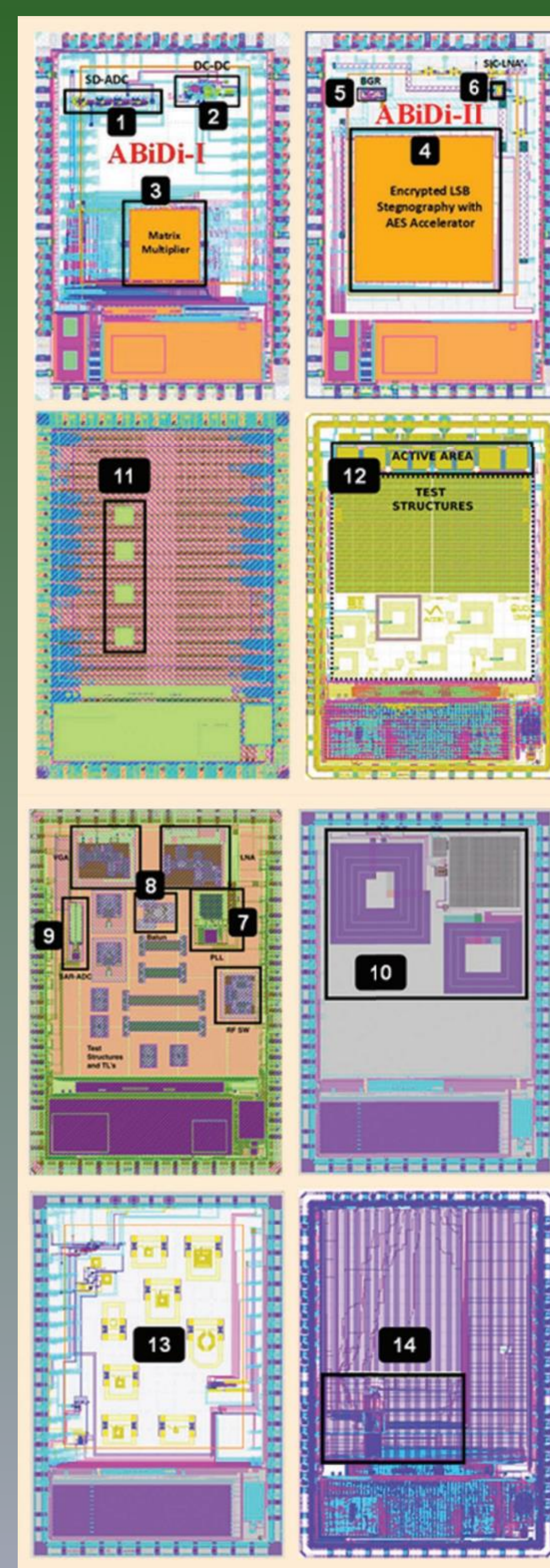
Additional remarks I

▶ Firm deadlines!

- November 11 is the hard tapeout date provided by Efabless
- Please plan according to the proposed activities: design submission and reviews (see schedule)
- Define a “minimum viable product” as soon as possible, e.g.:
 - A working design based on open source IP for analog
 - Working Verilog code with the main functionality for digital
 - For inspiration: Open source designs – public projects in the efabless platform: <https://platform.efabless.com/projects/public>
 - Also, join the OSS slack channel: <https://join.skywater.tools>

Additional remarks II

- ▶ **UNIC-CASS uses multi-block caravels**
 - Area and pin count are limited for each project
 - Exact numbers will be adjusted as we see progress
 - Minimize merge work as much as possible



Additional remarks III

- ▶ **Finalizing the design is not the finishing line!**
 - Efabless pre-checks
 - Tapeout checks
 - Based on previous experience, some issues can arise at these steps
 - Allocate extra time to understand and execute this
 - Mock tapeout!

Current actions by design teams

▶ Get the OS design tools working

- Instructions available in UNIC-CASS educational material website: <https://unic-cass.github.io/training/02-env-setup.html>
- Example: docker version with all the tools you need → <https://github.com/iic-jku/IIC-OSIC-TOOLS>

▶ Start your design → define your “roadmap”

- Distribute tasks among members (e.g. analog/digital, schematic/layout, etc.)
- Define concrete building blocks and their specifications
- Create testbenches and get started with simulations

UNIC-CASS Design-to-Tapeout Meetup #2 (Aug. 28, 2024)

Introduction to the Opensource Analog Design Flow

Presented by: Jorge Marín, Research Associate, AC3E-USM



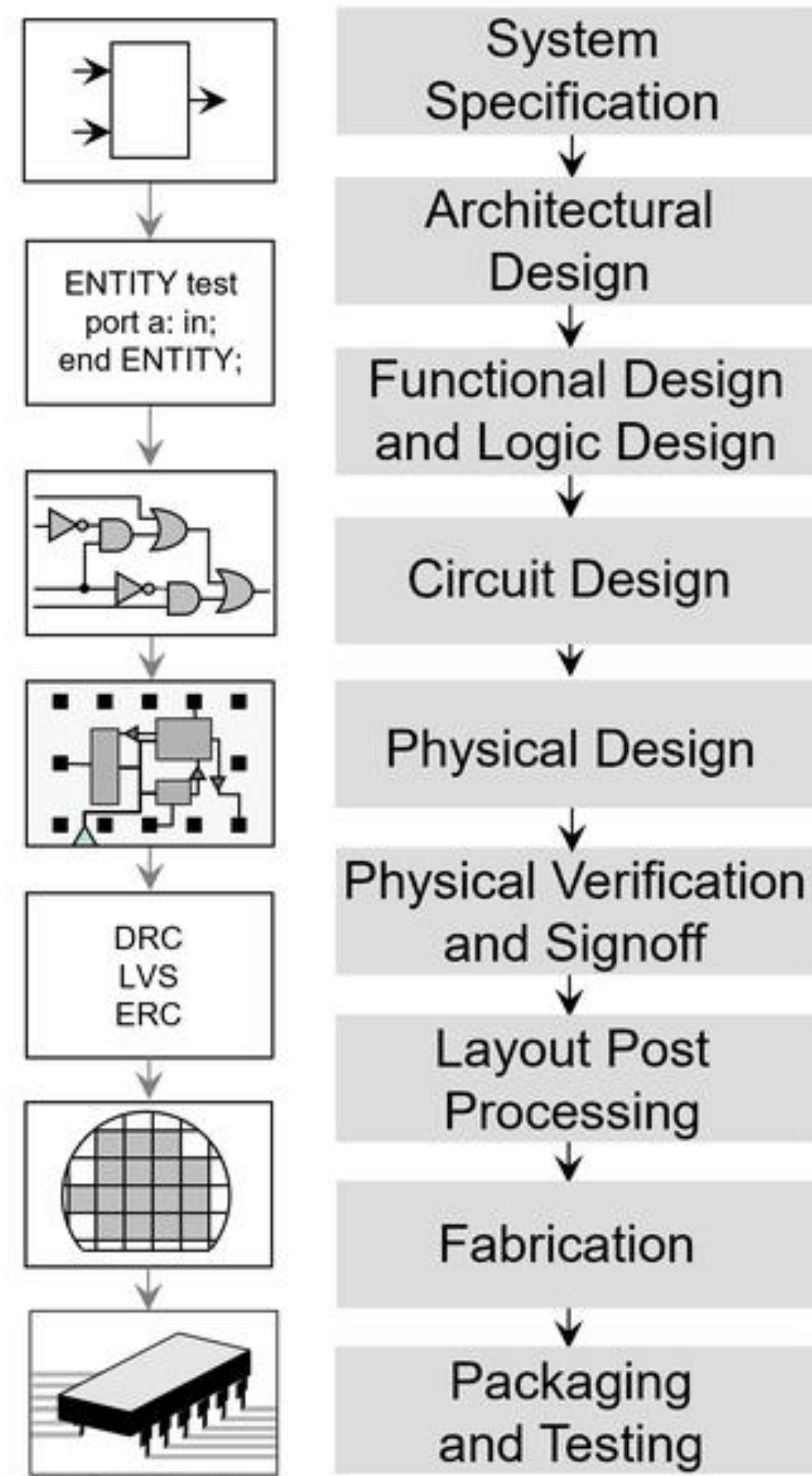


INTRODUCTION

Analog vs digital design flows

OPENSOURCE EDA TOOLS

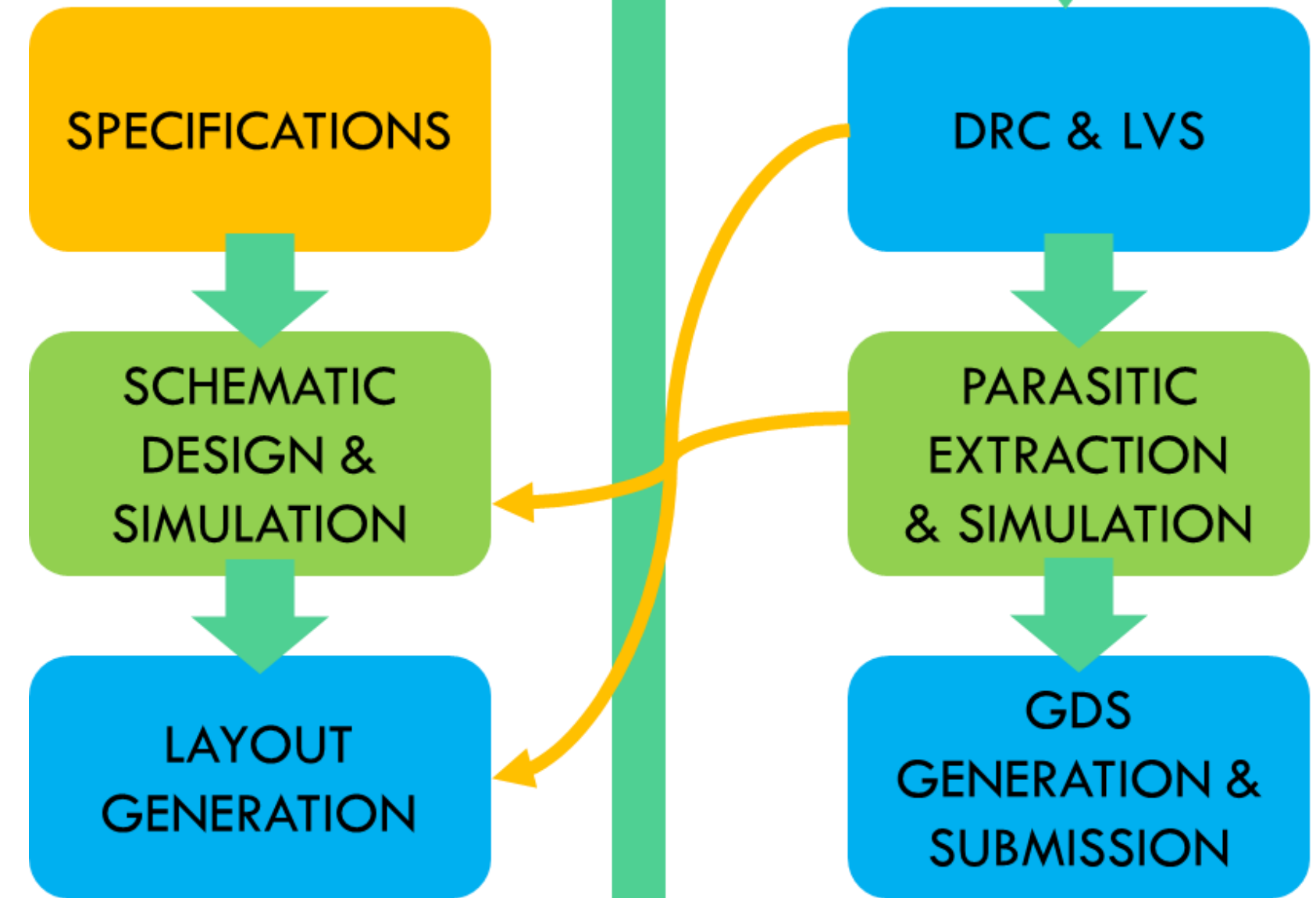
DIGITAL FLOW



OpenROAD

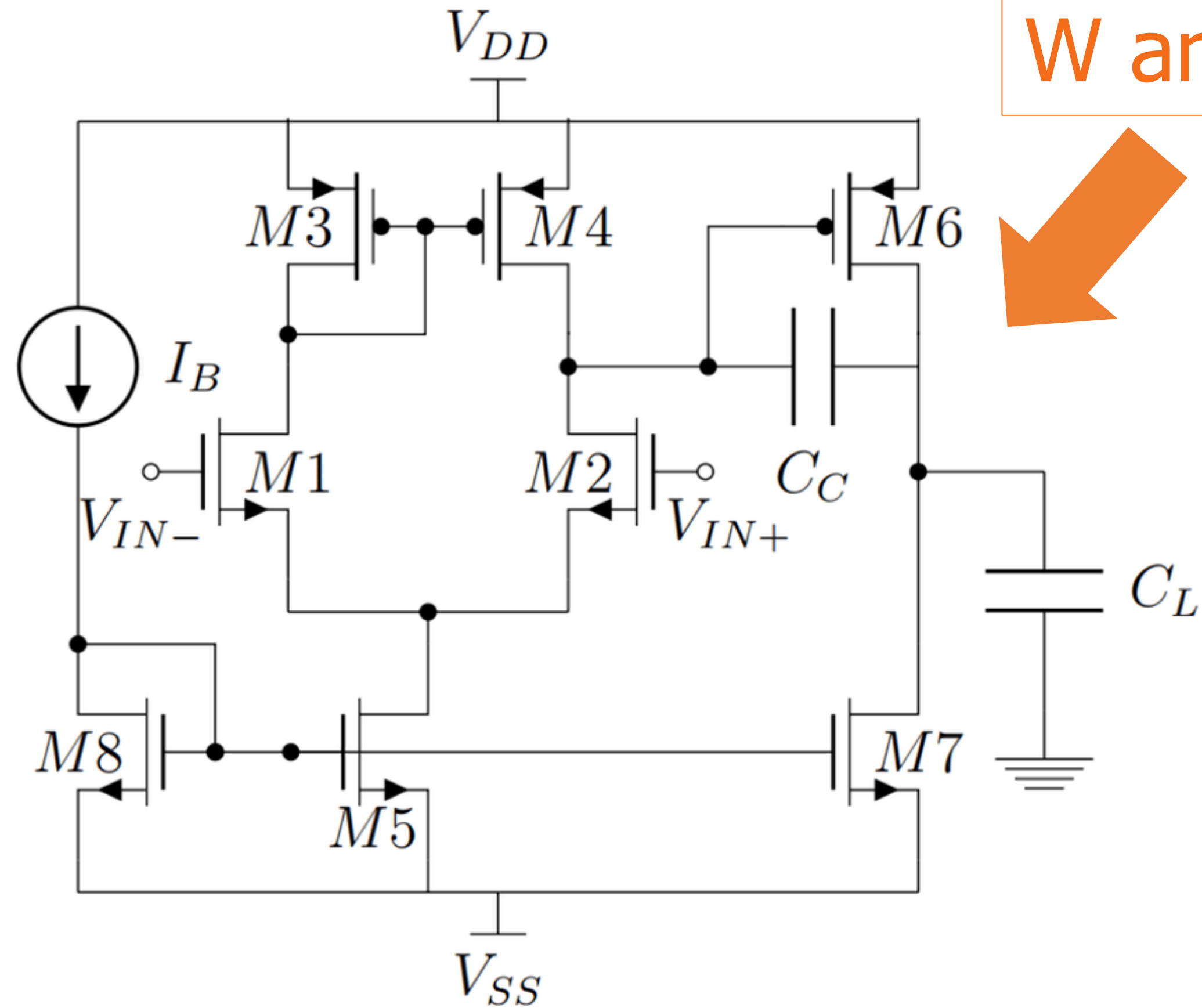


ANALOG FLOW

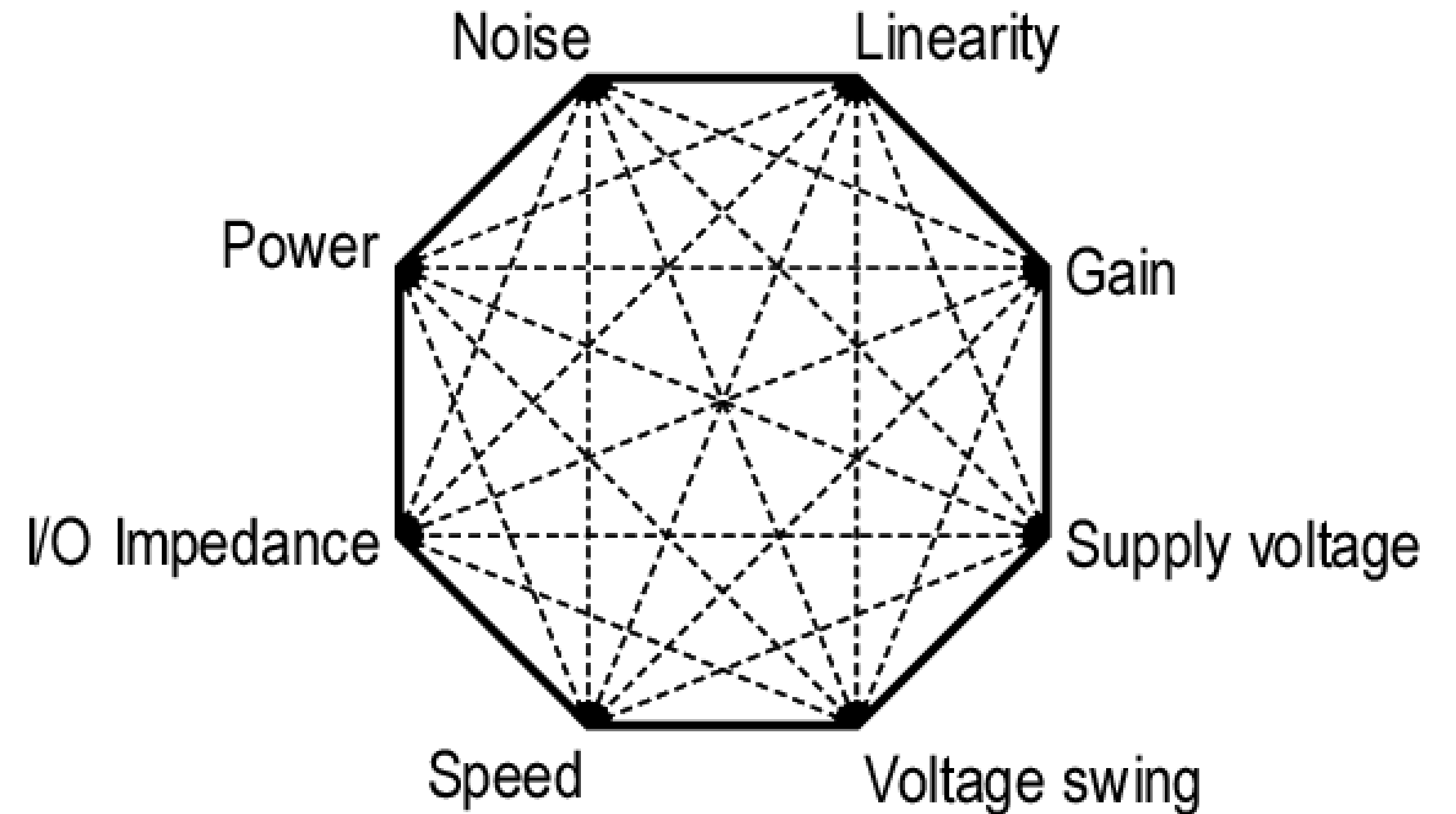


Tradeoffs in analog design

W and L for each transistor M1 to M8?



Analog block example: Miller OTA



[B. Razavi]

Performance is limited by the tradeoff in target specifications



**BASIC USE OF
ANALOG TOOLS**

Analog design flow environment setup

How to get started with design and simulation?

- Environment setup

- Option 1: Analog Mixed Signal Design using docker image & remote desktop

- e.g. IIC-OSIC-TOOLS docker

- Clone at: <https://github.com/iic-jku/iic-osic-tools>

- Follow detailed instructions

- Option 2: Analog Mixed Signal Design tools on Linux or WSL using Conda

See the environment setup material in the UNIC-CASS page:
<https://unic-cass.github.io/training/02-env-setup.html>

Schematic design and simulation

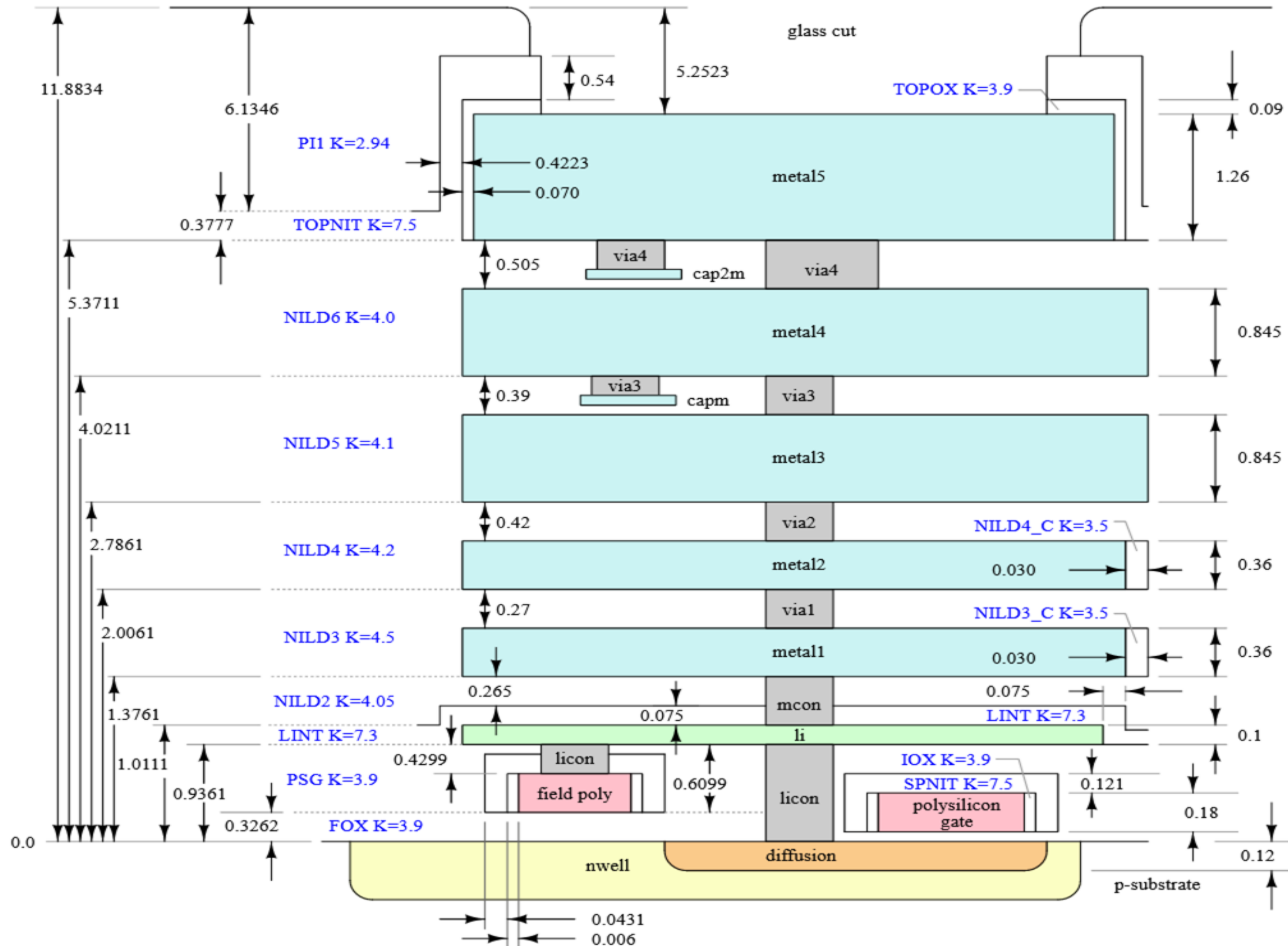
- Relevant tools
 - Xschem → schematic entry and netlist generation
 - Ngspice → simulation based on netlist generated by Xschem
- Visualization
 - Ngspice window → quick checks
 - GAW → integrated in Xschem
 - External viewer through raw data (e.g. Python script)

Simulation types

- Simulation types
 - DC → operating point
 - AC → frequency sweep
 - Transient → time-domain behavior
 - Noise → simulation of device intrinsic noise
 - And others...

Ngspice manual will become your best friend!
<https://ngspice.sourceforge.io/docs/ngspice-manual.pdf>

Available devices in SKY130



Device Details

1.8V NMOS FET

Spice Model Information

- Cell Name: sky130_fd_pr__nfet_01v8
- Model Name: sky130_fd_pr__nfet_01v8

Operating Voltages where SPICE models are valid

- $V_{DS} = 0$ to 1.95V
- $V_{GS} = 0$ to 1.95V
- $V_{BS} = +0.3$ to -1.95V

SPICE models inside IIC-OSIC-TOOLS docker:
`.lib /foss/pdks/sky130A/libs.tech/ngspice/sky130.lib.spice tt`

Explore
www.skywater-pdk.readthedocs.io
for device info

Xschem GUI

www.xschem.sourceforge.io

The screenshot shows the Xschem GUI interface. At the top is a menu bar with options: File, Edit, Options, View, Properties, Layers (highlighted), Tools, Symbol, Highlight, Simulation, Netlist, Simulate, Waves, and Help. Below the menu bar is a toolbar with various icons for file operations and editing. The main workspace is dark and contains a list of keyboard shortcuts and their corresponding actions. At the bottom, there is a status bar with settings like SNAP: 10, GRID: 20, NETLIST MODE: spice, and mouse = 40 -630 - selected: 0 path: .X1.

| | |
|------------------|---------------------------|
| 'w' | draw wire |
| 'm' | move element |
| 'ALT + f' | flip |
| 'SHFT + r' | rotate |
| 'u' | undo |
| 'c' | copy |
| 'q' | open object query window |
| 'a' | auto symbol creation |
| 'insert' | open insert device window |
| 'e' / 'CTRL + E' | descend/ascend hierarchy |

Generate netlist

Execute simulation

Check waveforms (GAW)

Simulation scope

- Nominal
 - Ideal simulation without considering many fabrication effects
- P(VT) corners
 - Considers global process variation (P) and environment (Voltage, temperature, T)

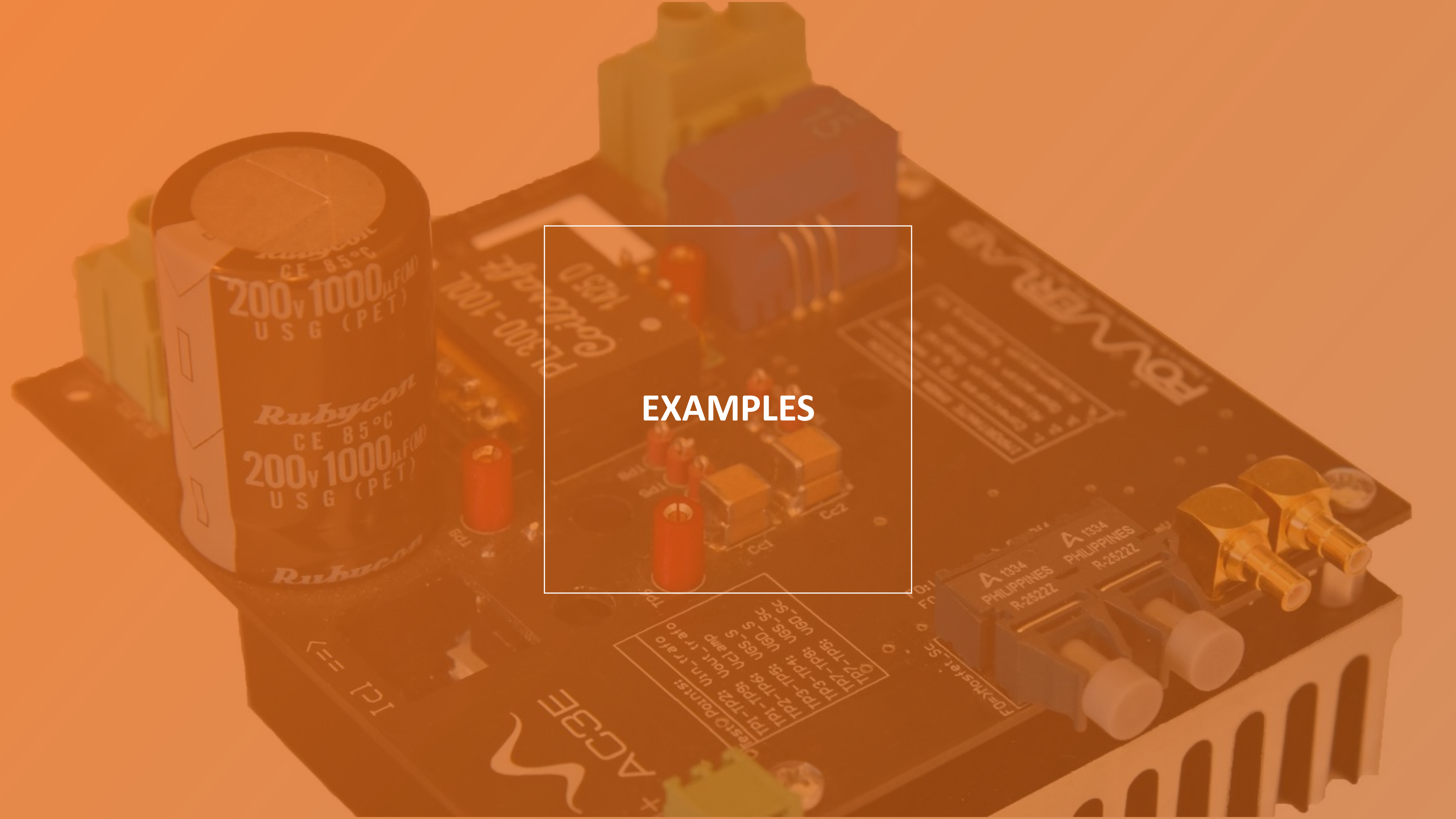
```
.lib /foss/pdks/sky130A/libs.tech/ngspice/sky130.lib.spice tt
```

- Mismatch
 - Considers local statistical variation among devices
 - See example: <https://unic-cass.github.io/training/7.7-design-examples-xschem-ngspice-mc-sim.html>
- Parasitic extraction/ post layout simulations
 - Components associated to extrinsic structures (metallization)

CACE: Circuit Automatic Characterization Engine

- Allows to run Montecarlo, corner sweeps and post-layout simulations in a structured way
 - Usage and examples: <https://github.com/efabless/cace>
 - Documentation: <https://cace.readthedocs.io/en/latest/>
 - See also the “cace” channel in the OSS Slack
- "CACE Study: Open source analog and mixed-signal design flow" - Tim Edwards (Latch_2024):
 - <https://youtu.be/0UMb-vd4MtU?si=4eZUPfScApC2MfHf>

EXAMPLES

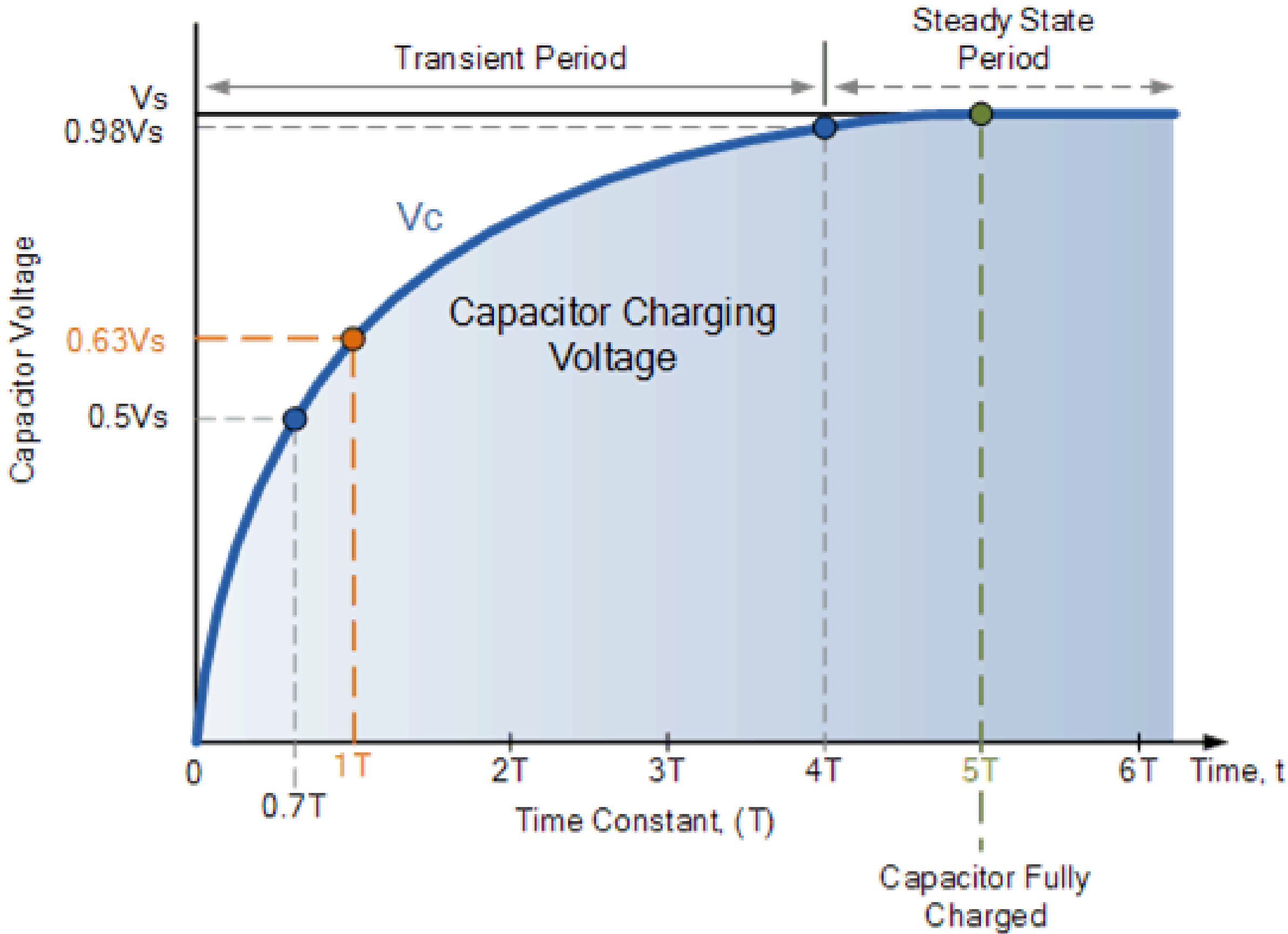
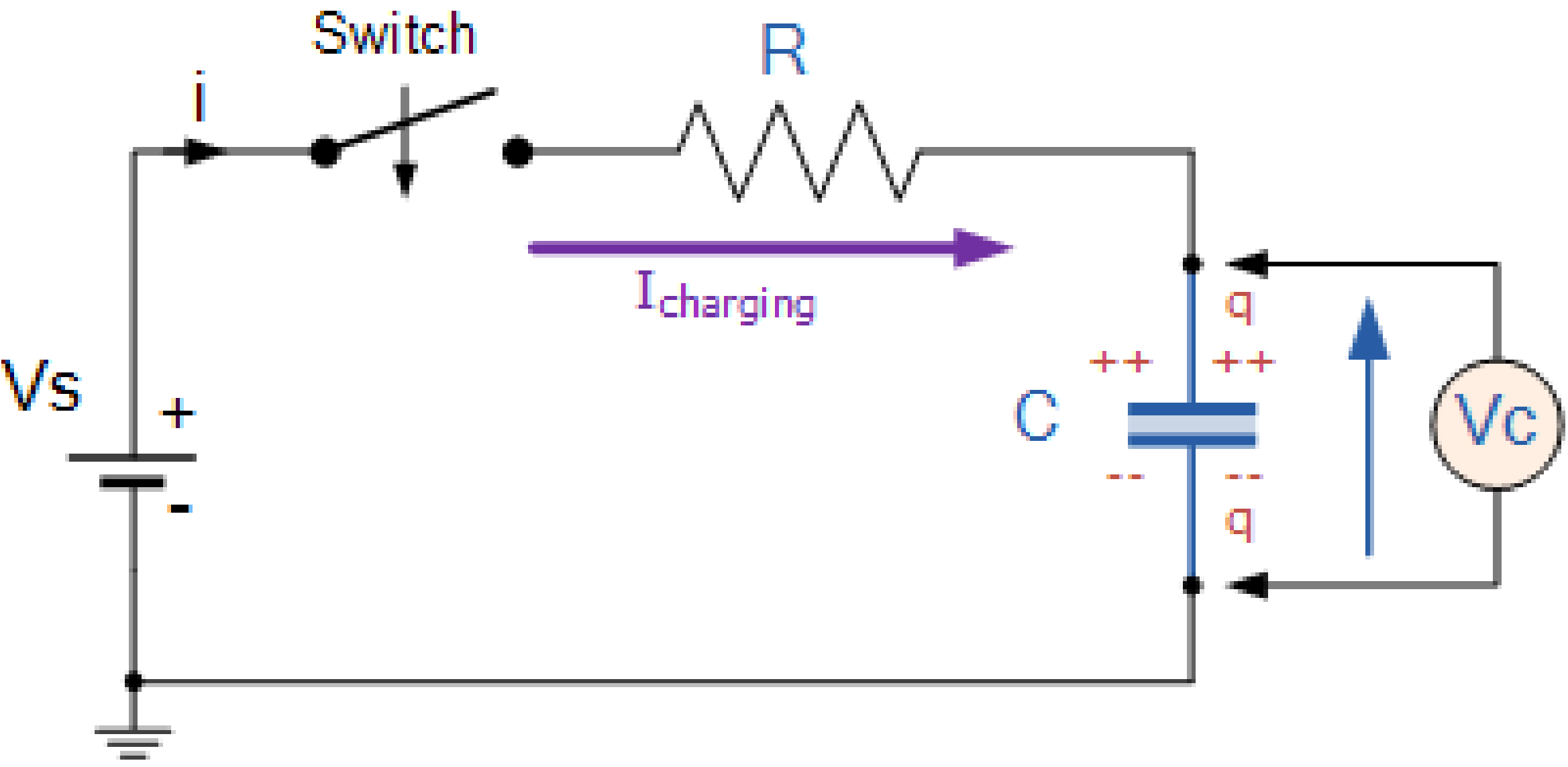


An inverter schematic in Xschem with Skywater 130nm

Check video tutorial:

<https://unic-cass.github.io/training/3.1.3-schematic-capture-inverter.html>

Example #2: RC constant calculation



Source: https://www.electronics-tutorials.ws/rc/rc_1.html

Example #2: RC constant calculation

- Github link:

→ https://github.com/JorgeMarinN/UNIC-CASS2024_AnalogOSIC/blob/main/cgate-ext_circuit_UNIC-CASS2024.sch

Schematic

```
s1
.tran 100p 20n
.save all
.ic v(vout) = 0
.control
run
meas tran teval WHEN v(vout) = 0.63
let res_val = 1000
let cap_val = teval/res_val
print cap_val
.endc
```

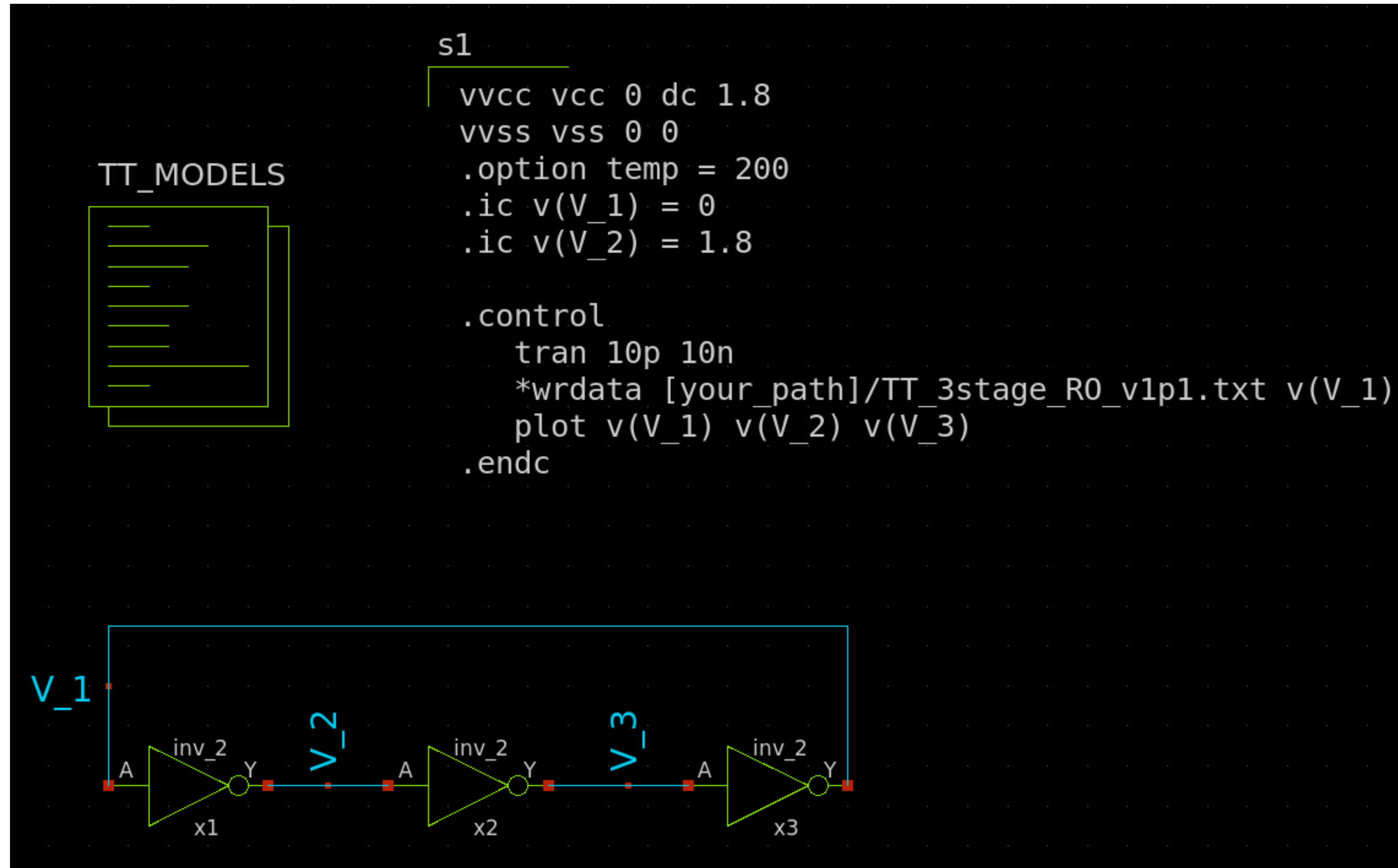
SPICE deck definition

```
TT_MODELS
[...]
```

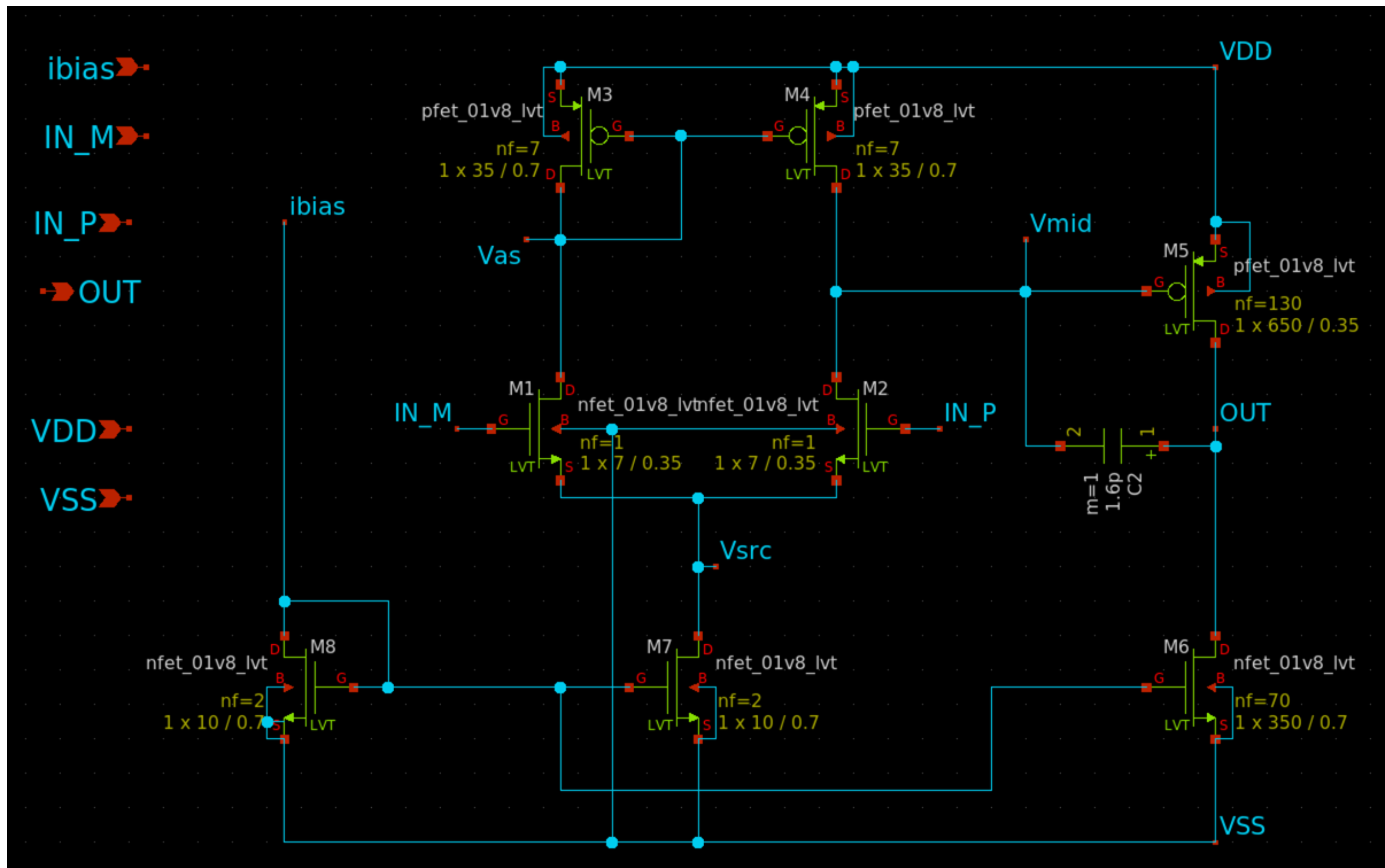
Example #3: ring oscillator

- Github link:

→ https://github.com/JorgeMarinN/UNIC-CASS2024_AnalogOSIC/blob/main/tb_3stage_RO_UNIC-CASS2024.sch



Example #4: Miller OTA simulation



$$GBW = \frac{gm_{1,2}}{2\pi C_c}$$

$$f_{nd} = \frac{gm_5}{2\pi C_L} \cdot \frac{1}{1 + \frac{C_{n1}}{C_c}} > 3GBW$$

Two equations for

Three variables g_{m1} , g_{m6} and C_c ?!?

Solution : choose g_{m1} or g_{m6} or C_c !!!

[W. Sansen, Analog Design Essentials]

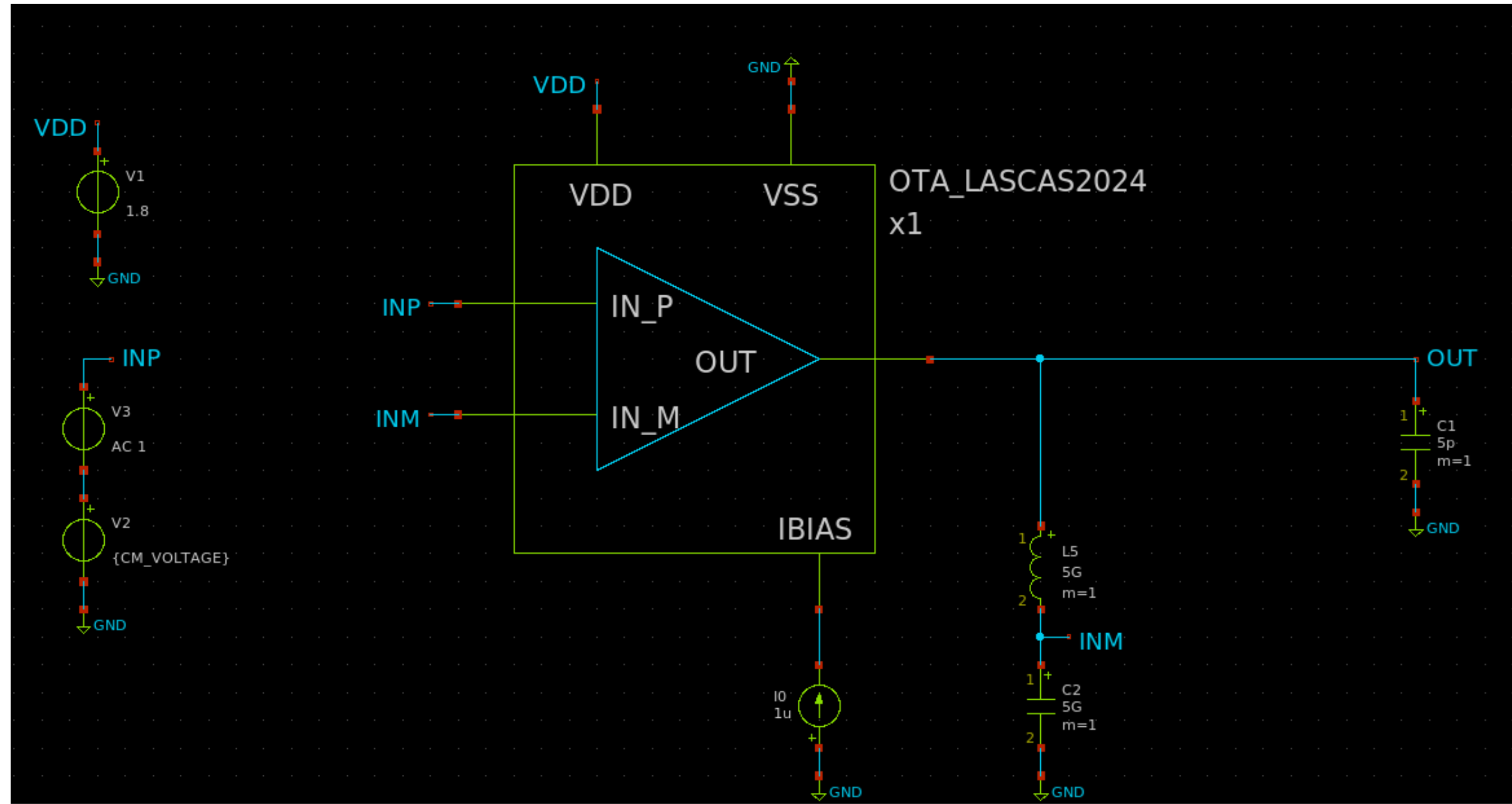
Example #4: Miller OTA simulation

- Github link:

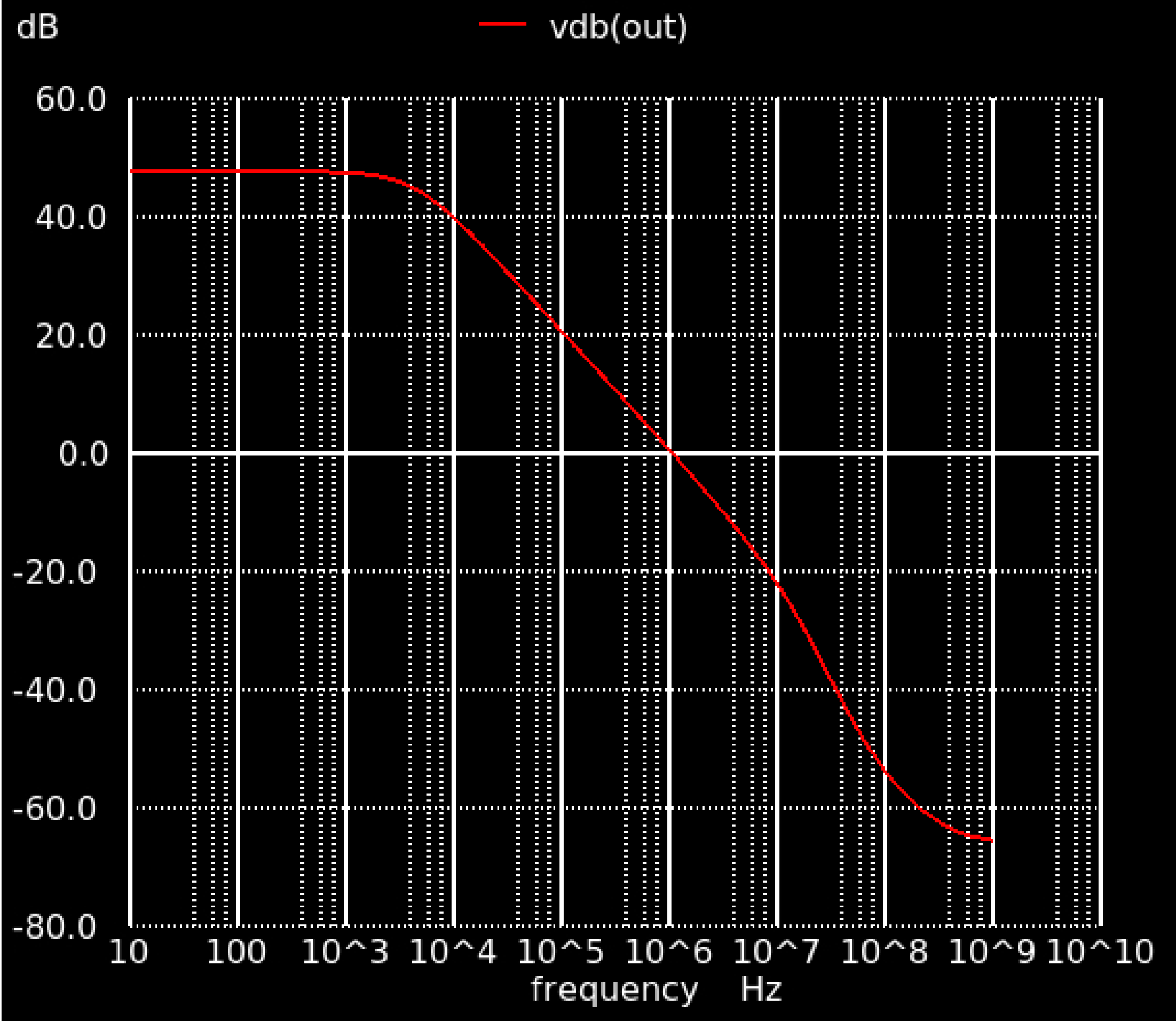
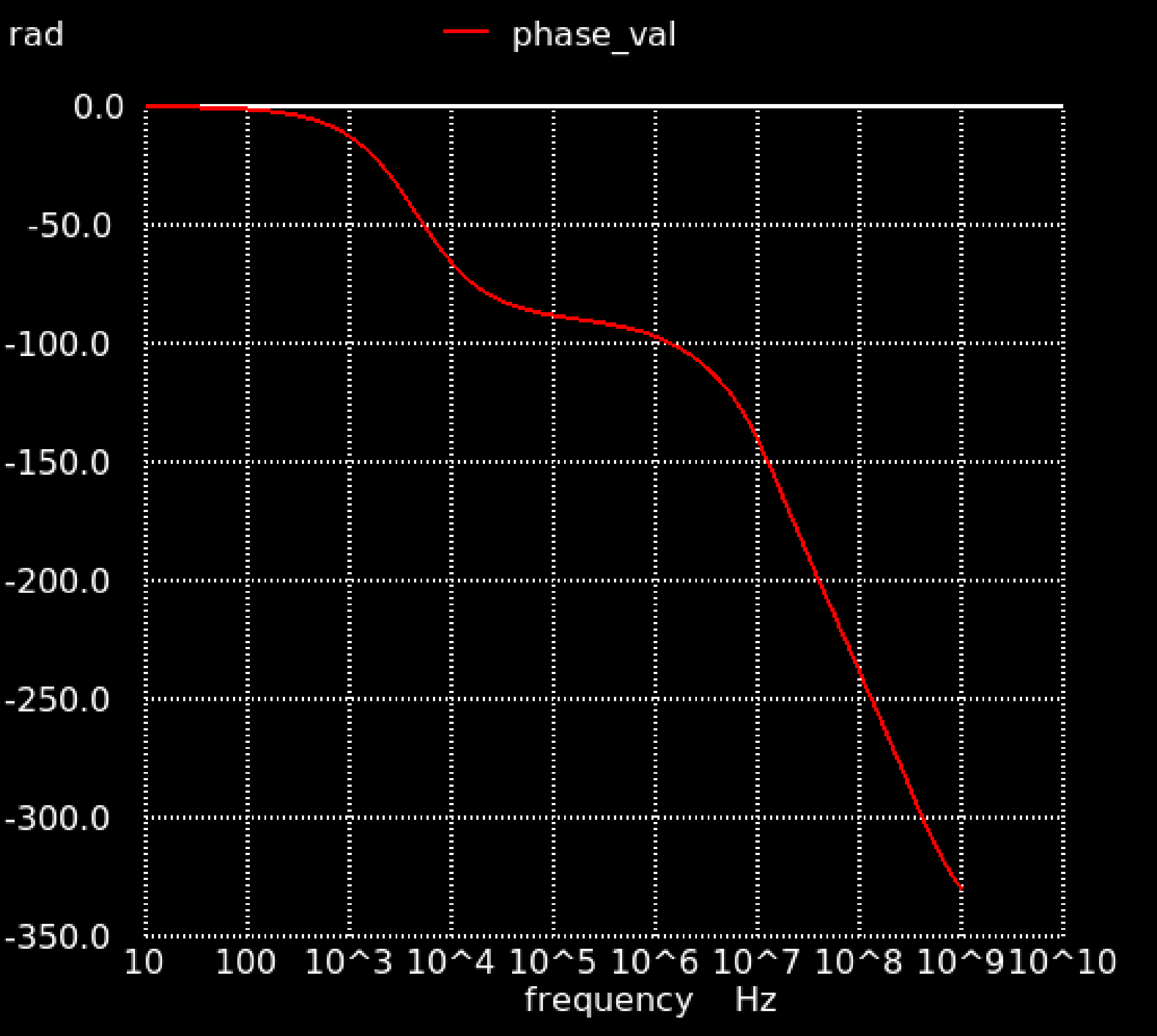
→ https://github.com/JorgeMarinN/UNIC-CASS2024_AnalogOSIC/blob/main/tb_OTA_stability_UNIC-CASS2024.sch

NGSPICE

```
.param CM_VOLTAGE = 0.9
.control
save all
ac dec 200 10 1000Meg
settype decibel out
plot vdb(out)
let phase_val = 180/PI*cph(out)
let PM_val = 180 + 180/PI*cph(out)
settype phase phase_val
plot phase_val
meas ac PM_FIND PM_val WHEN vdb(out)=0
meas ac GBW WHEN vdb(out)=0
op
```



Example #4: Miller OTA simulation



| | | |
|-----|---|--------------|
| pm | = | 8.248775e+01 |
| gbw | = | 1.058205e+06 |

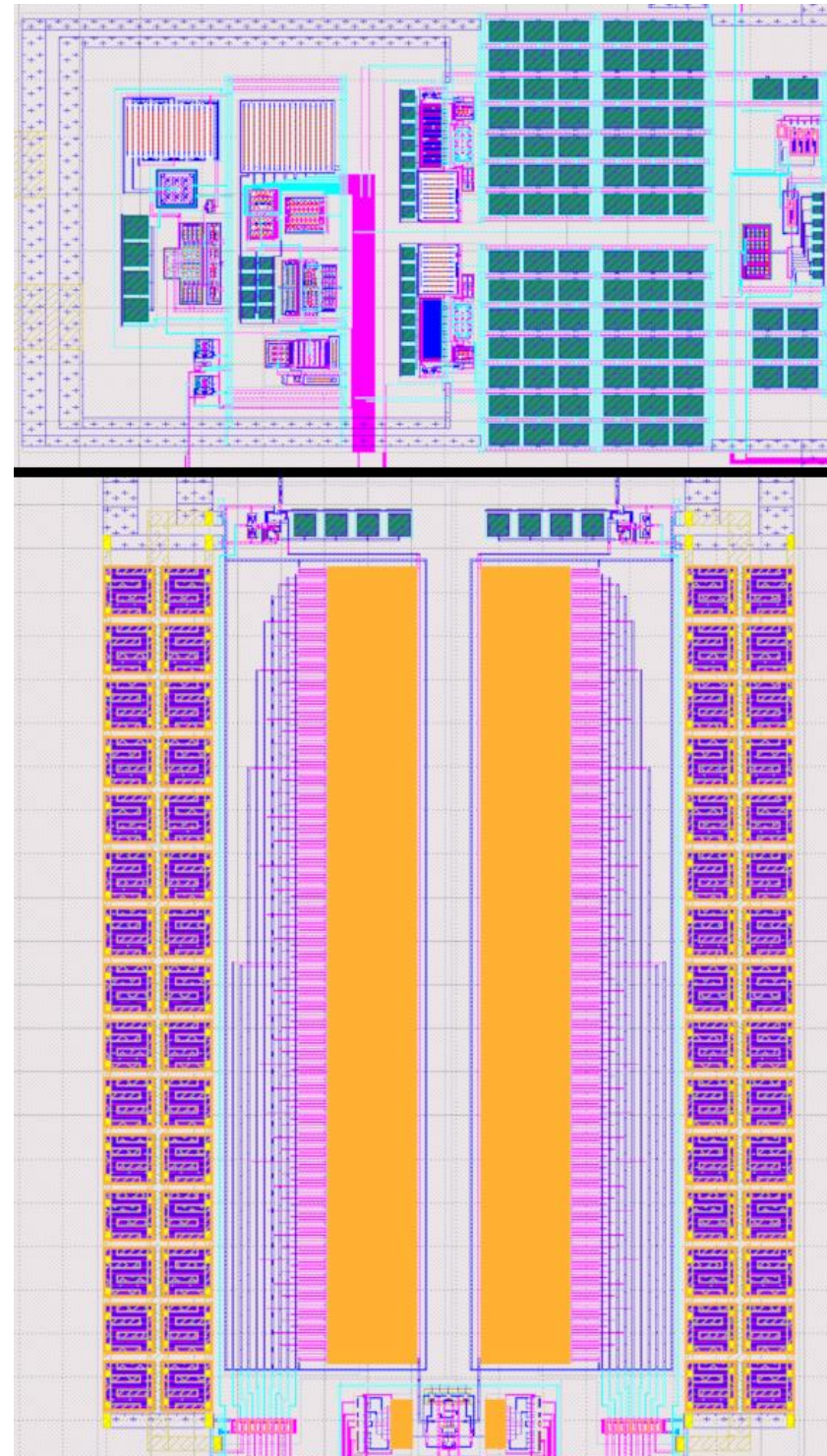


Other resources

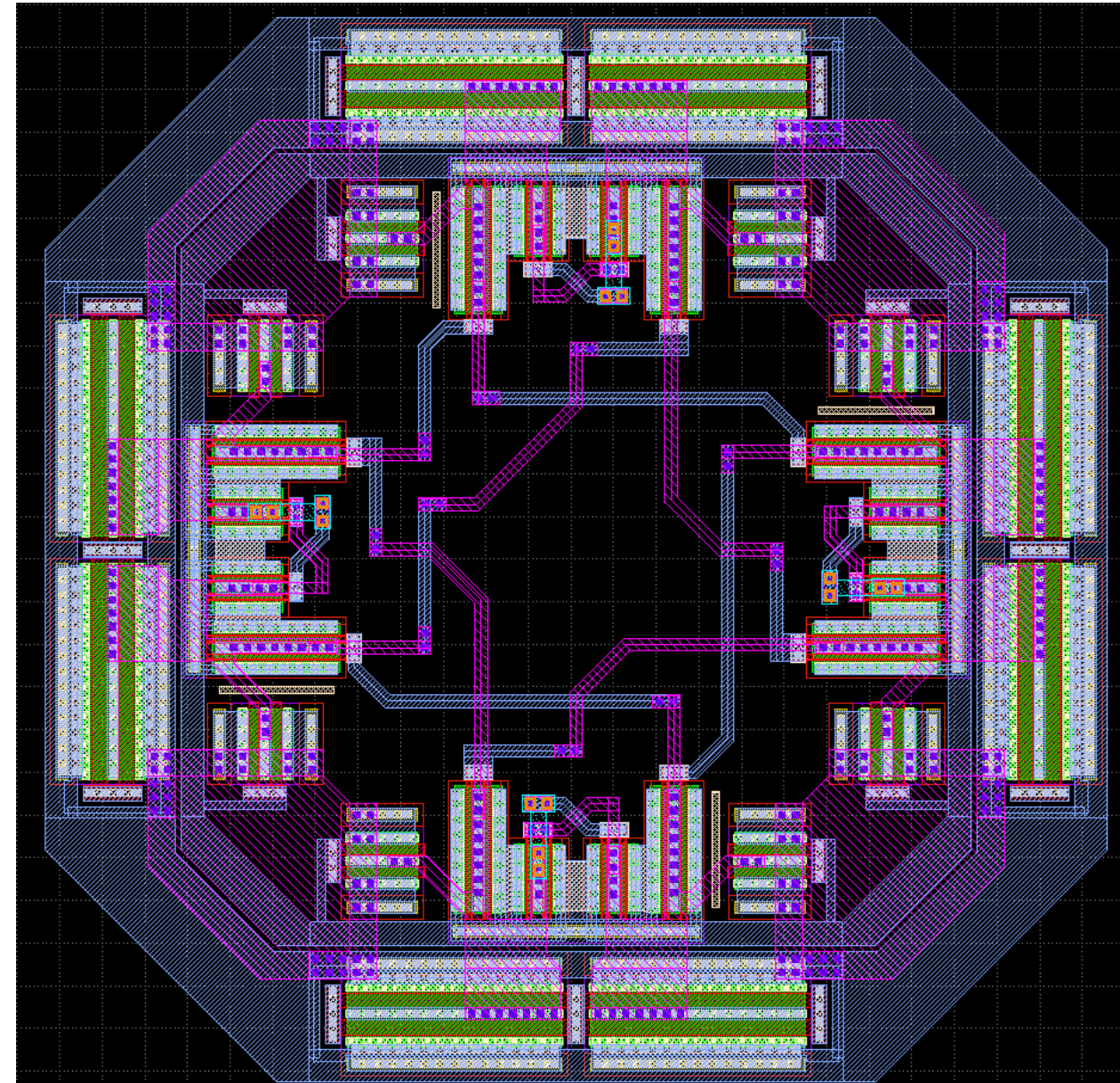
Opensource designs → re-use and get inspired

Visit: <https://platform.efabless.com/projects/public>

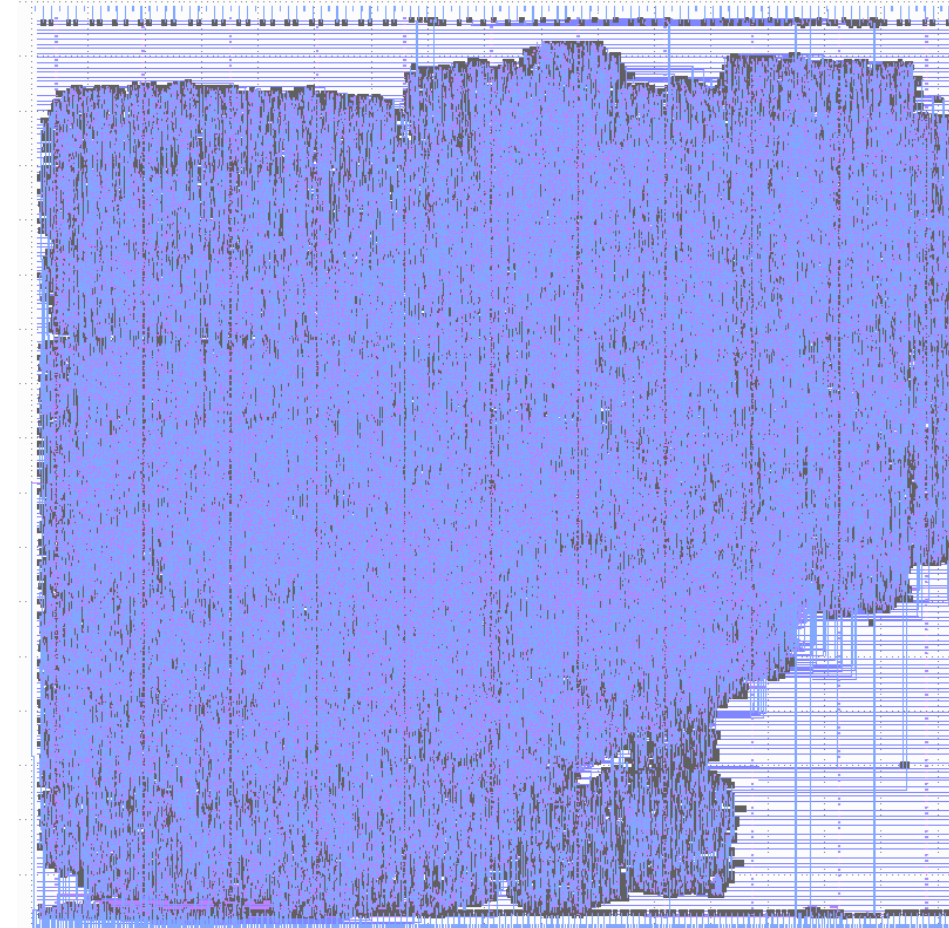
SAR ADC



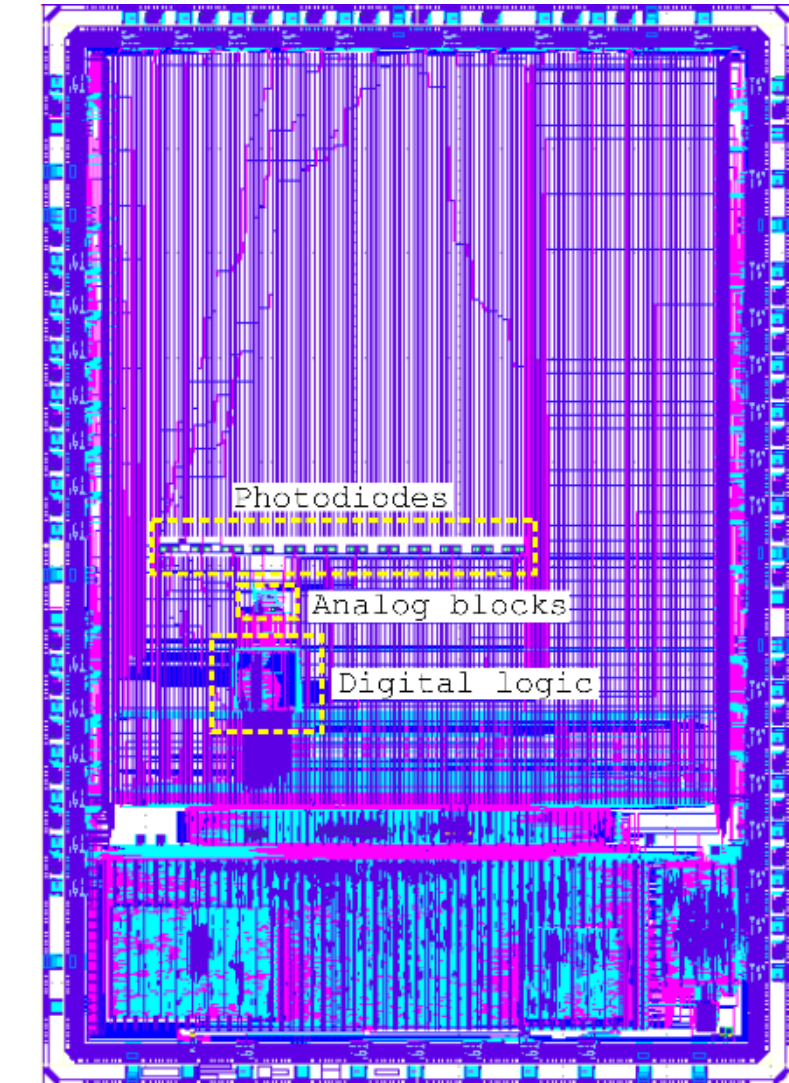
Satellite transceiver



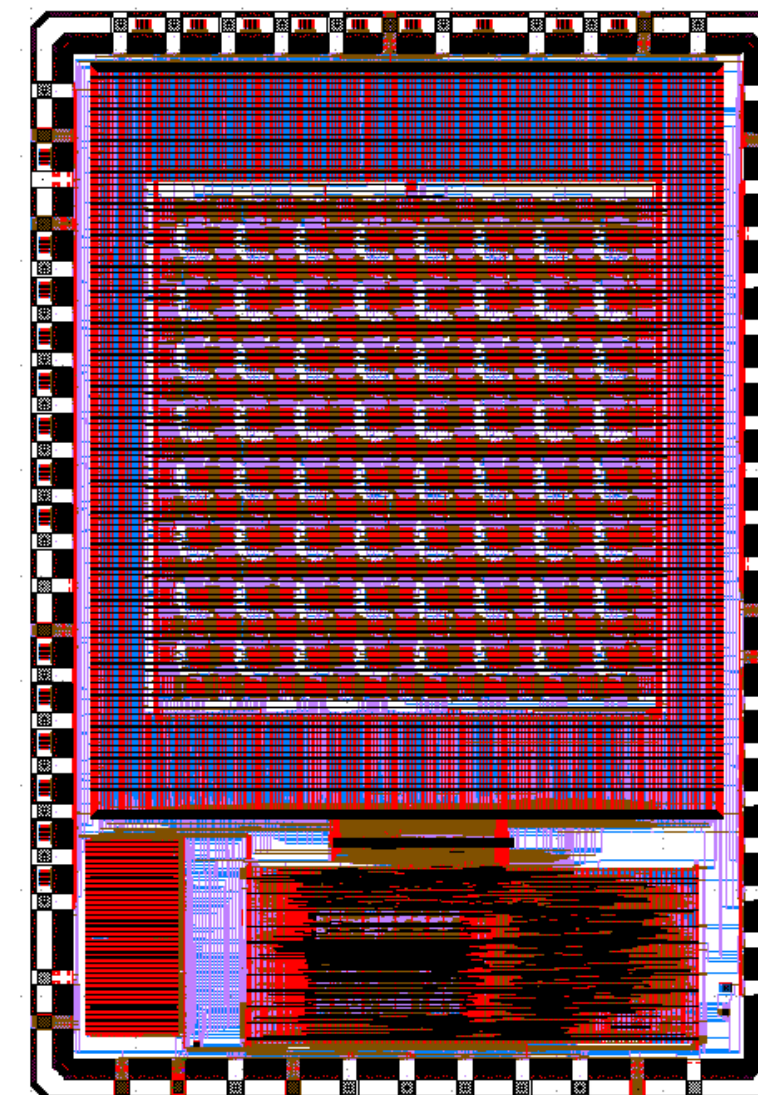
Sudoku Accelerator



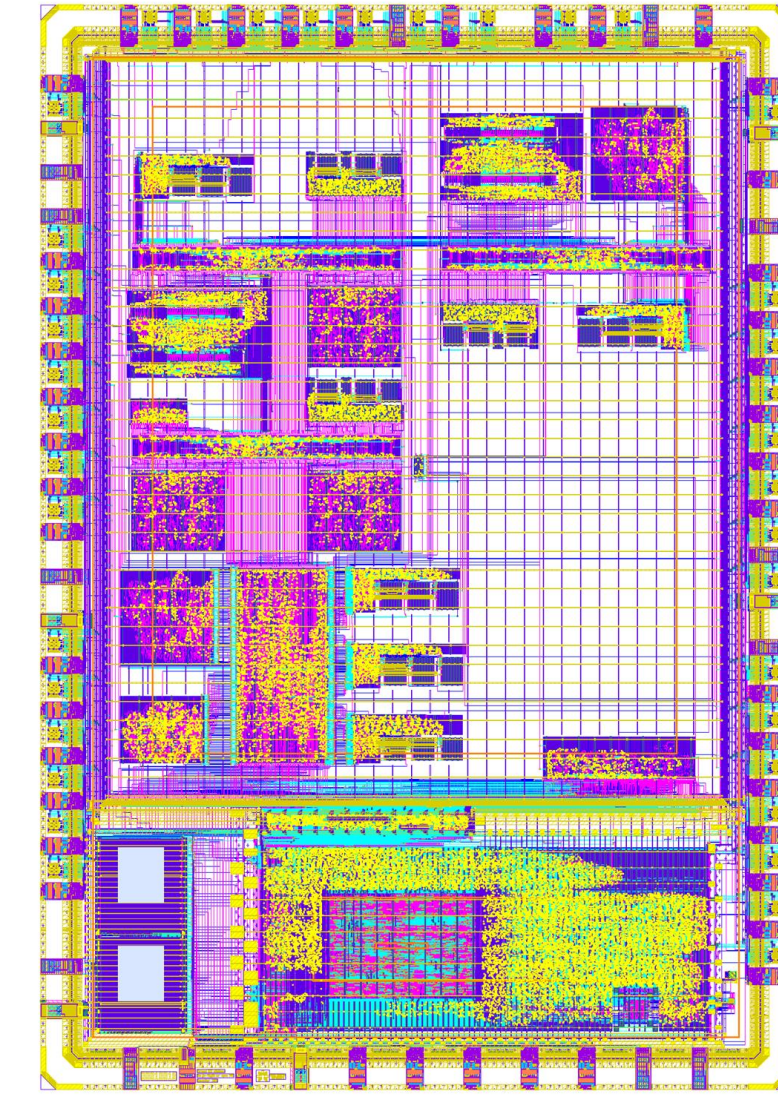
MixPix



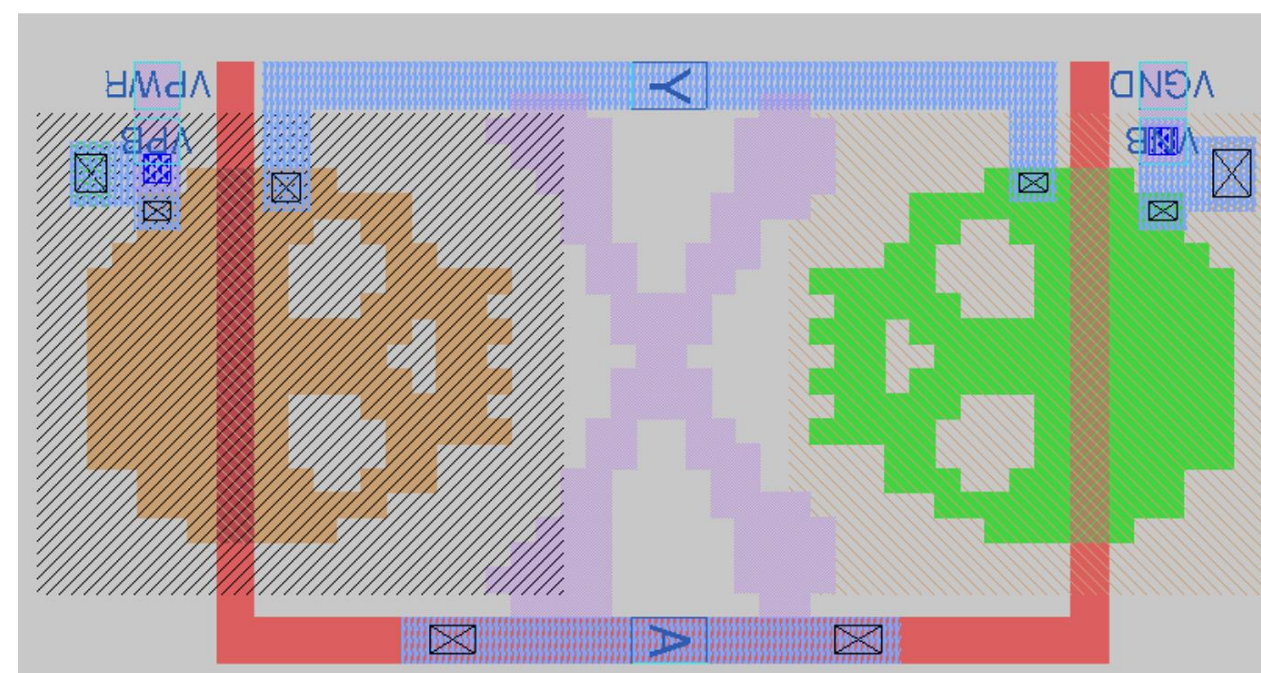
OpenFPGA



Time to Data Converter



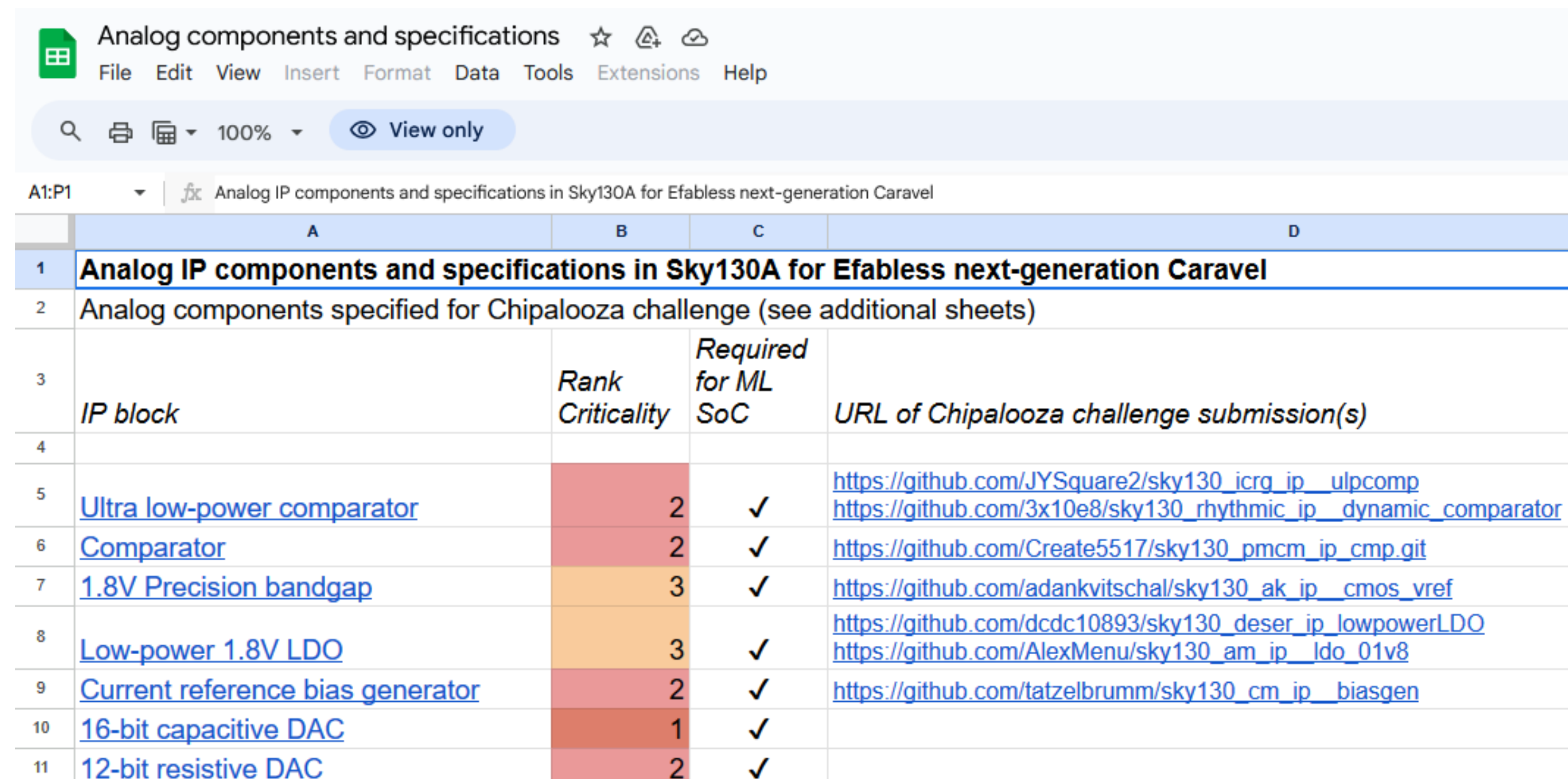
Logic inverter



Opensource designs → re-use and get inspired

- Analog IP components and specifications in Sky130A for Efabless next-generation Caravel (Chipalooza):

<https://docs.google.com/spreadsheets/d/132YkMiYaM0iHML5feT1yWLQIvP-pCM0lCArM9f1LxhM/edit?gid=0#gid=0>



The screenshot shows a Google Sheet titled "Analog components and specifications" for the "Analog IP components and specifications in Sky130A for Efabless next-generation Caravel" challenge. The sheet contains a table with the following data:

| | A | B | C | D |
|----|--|-----------------------------|------------------------------------|--|
| 1 | Analog IP components and specifications in Sky130A for Efabless next-generation Caravel | | | |
| 2 | Analog components specified for Chipalooza challenge (see additional sheets) | | | |
| 3 | <i>IP block</i> | <i>Rank Criticality</i> | <i>Required for ML SoC</i> | <i>URL of Chipalooza challenge submission(s)</i> |
| 4 | | | | |
| 5 | Ultra low-power comparator | 2 | ✓ | https://github.com/JYSquare2/sky130_icrg_ip_ulpcomp https://github.com/3x10e8/sky130_rhythmic_ip_dynamic_comparator |
| 6 | Comparator | 2 | ✓ | https://github.com/Create5517/sky130_pmcm_ip_cmp.git |
| 7 | 1.8V Precision bandgap | 3 | ✓ | https://github.com/adankvitschal/sky130_ak_ip_cmos_vref |
| 8 | Low-power 1.8V LDO | 3 | ✓ | https://github.com/dcdc10893/sky130_deser_ip_lowpowerLDO https://github.com/AlexMenu/sky130_am_ip_ldo_01v8 |
| 9 | Current reference bias generator | 2 | ✓ | https://github.com/tatzelbrumm/sky130_cm_ip_biasgen |
| 10 | 16-bit capacitive DAC | 1 | ✓ | |
| 11 | 12-bit resistive DAC | 2 | ✓ | |

Code-a-chip notebooks

Example: https://github.com/sscs-ose/sscs-ose-code-a-chip.github.io/tree/main/VLSI23/accepted_notebooks/3LFCC

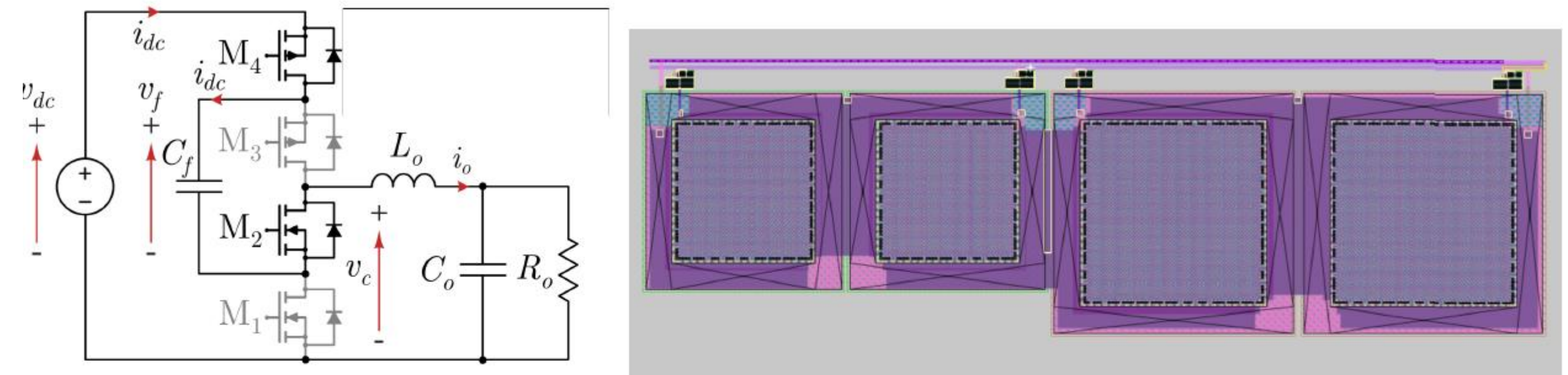
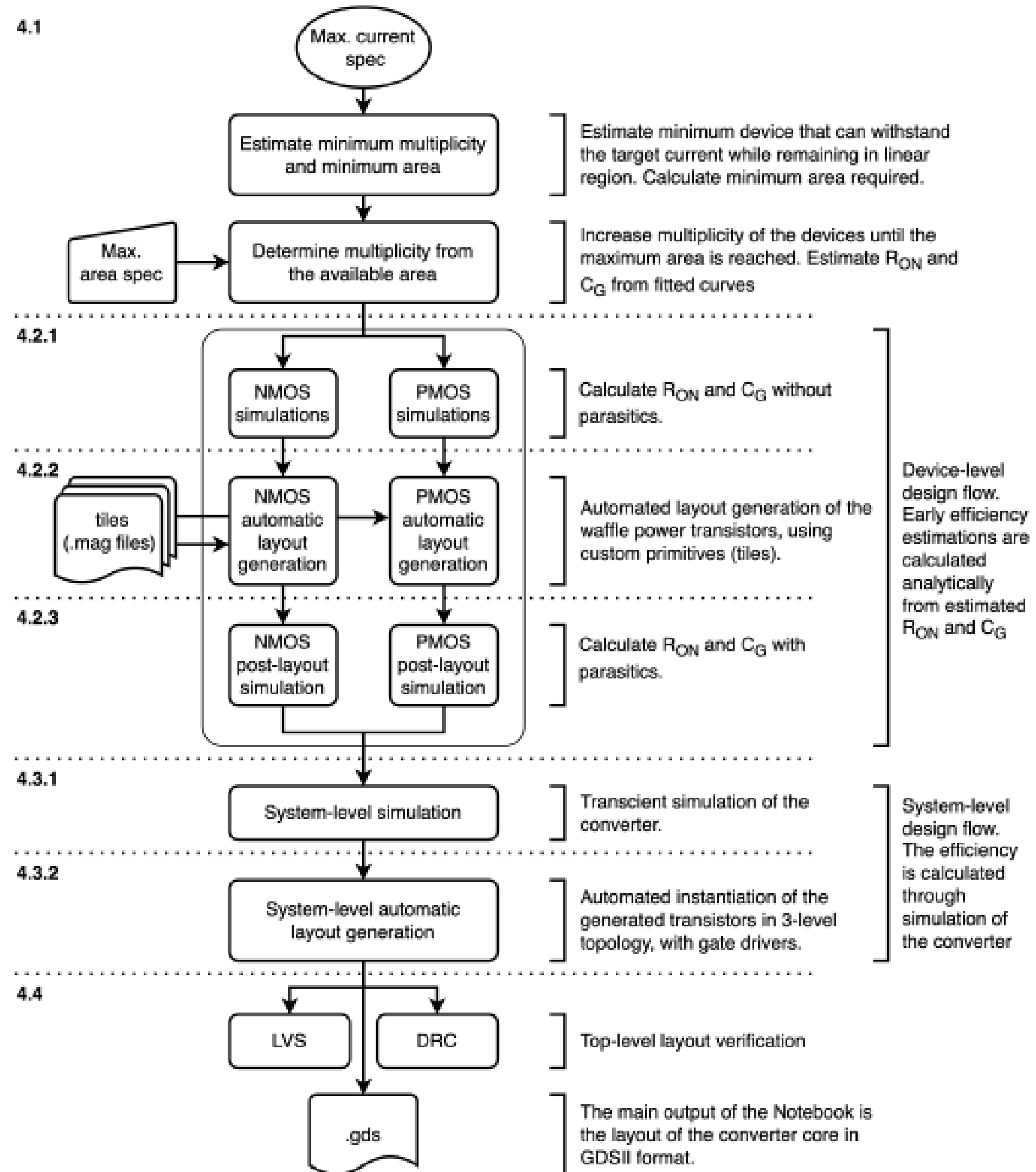
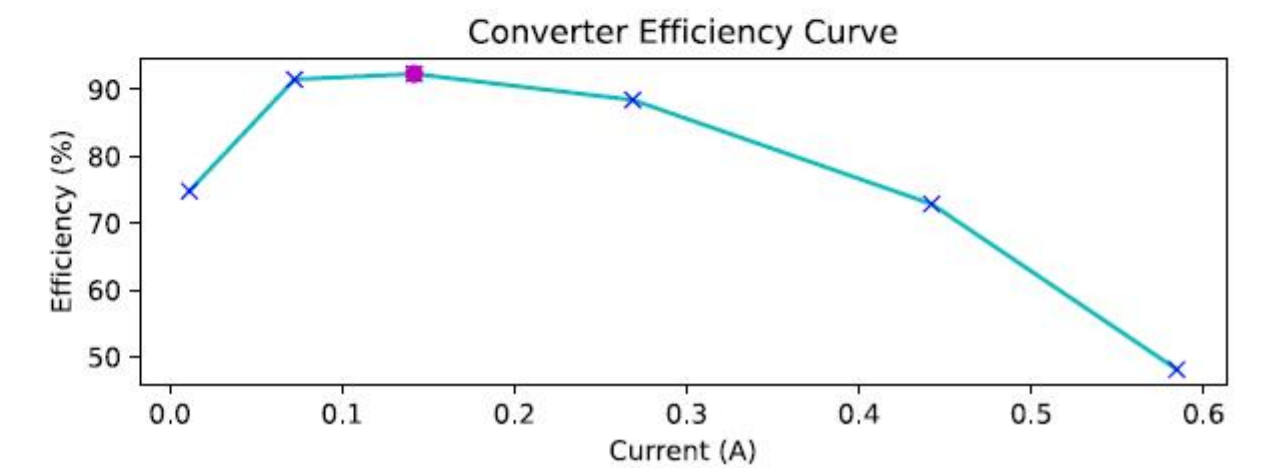
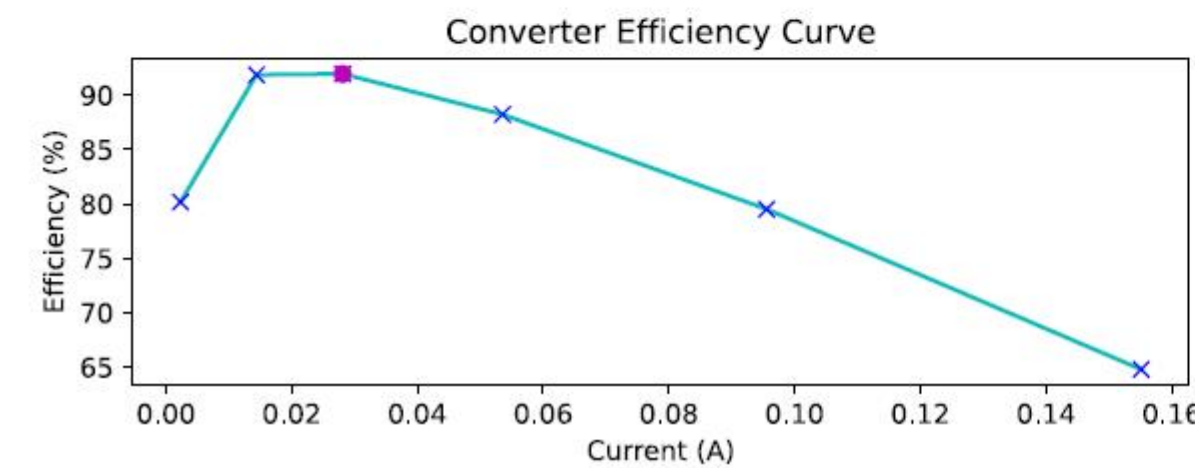


Figure 3: 3LFCC Converter schematic (left) and layout (right)



- Full Python-based automated flow using opensource toolkit
- From current, voltage, area and frequency specs to manufacturable layout (GDS file)



CAS

IEEE CIRCUITS AND SYSTEMS SOCIETY

For more information and to join CASS, visit:

IEEE-CAS.ORG