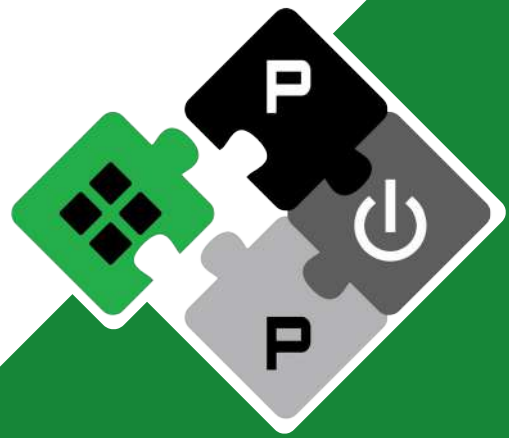
ETH zürich

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Five cases to show why open-source-hardware is Essential for modern SoC design in Europe

HiPEAC 2026, Kraków, Poland

Frank K. Gürkaynak kgf@iis.ee.ethz.ch



pulp-platform.org

@pulp_platform

company/pulp-platform

youtube.com/pulp_platform

PULP Platform

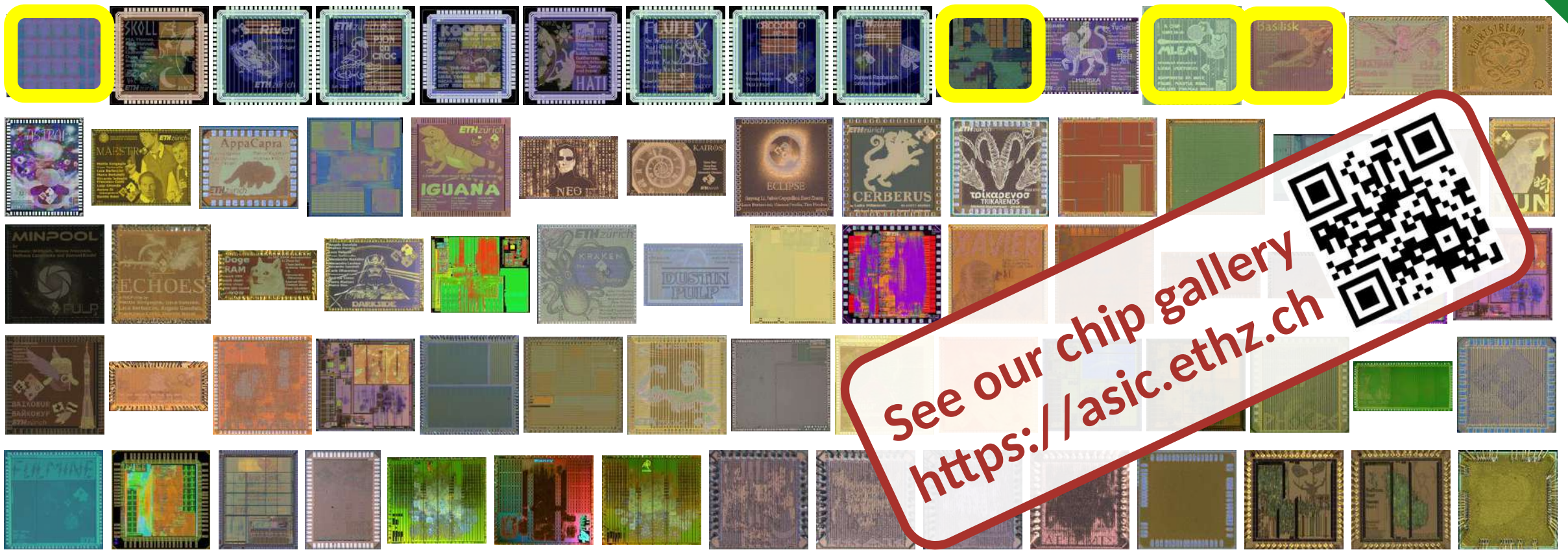
Open Source Hardware, the way it should be!

PULP platform has a long history in open source HW

- **PULP (Parallel Ultra Low Power) Platform started in 2013**
  - With the appointment of Luca Benini as professor to ETH Zurich while keeping his professorship at University of Bologna
- **From the beginning the group was active in open source hardware**
  - First open source project was PULPino, released in 2016
- **Starting in 2015 group started developing RISC-V based systems**
  - CVA6, CVE32E40P, Ibex/CVE2 all started their life as PULP platform projects

The PULP team is about 100 people Both in CH and IT

As part of our adventure we have designed 75 ASICs



Based on this experience I want to talk about why we see that

Open-source hardware is essential for Europe

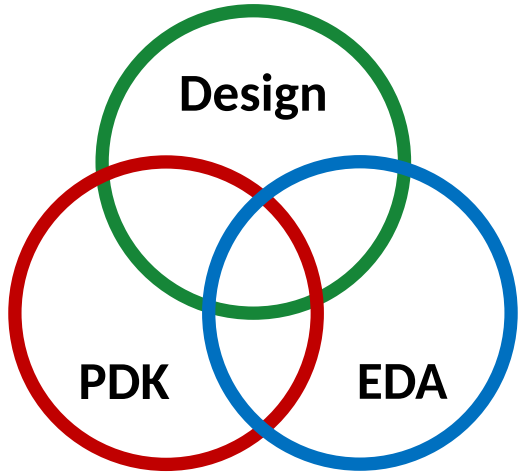
We will look at 3 connected phases of the IC design flow

Design

- RTL / HDL descriptions (quite common)
- Schematics / Physical Design (may have dependencies to technology information)

Tools (EDA)

- Front-end tools (Synthesis)
- Back-end tools (Placement and Routing)
- Verification tools (Simulation)

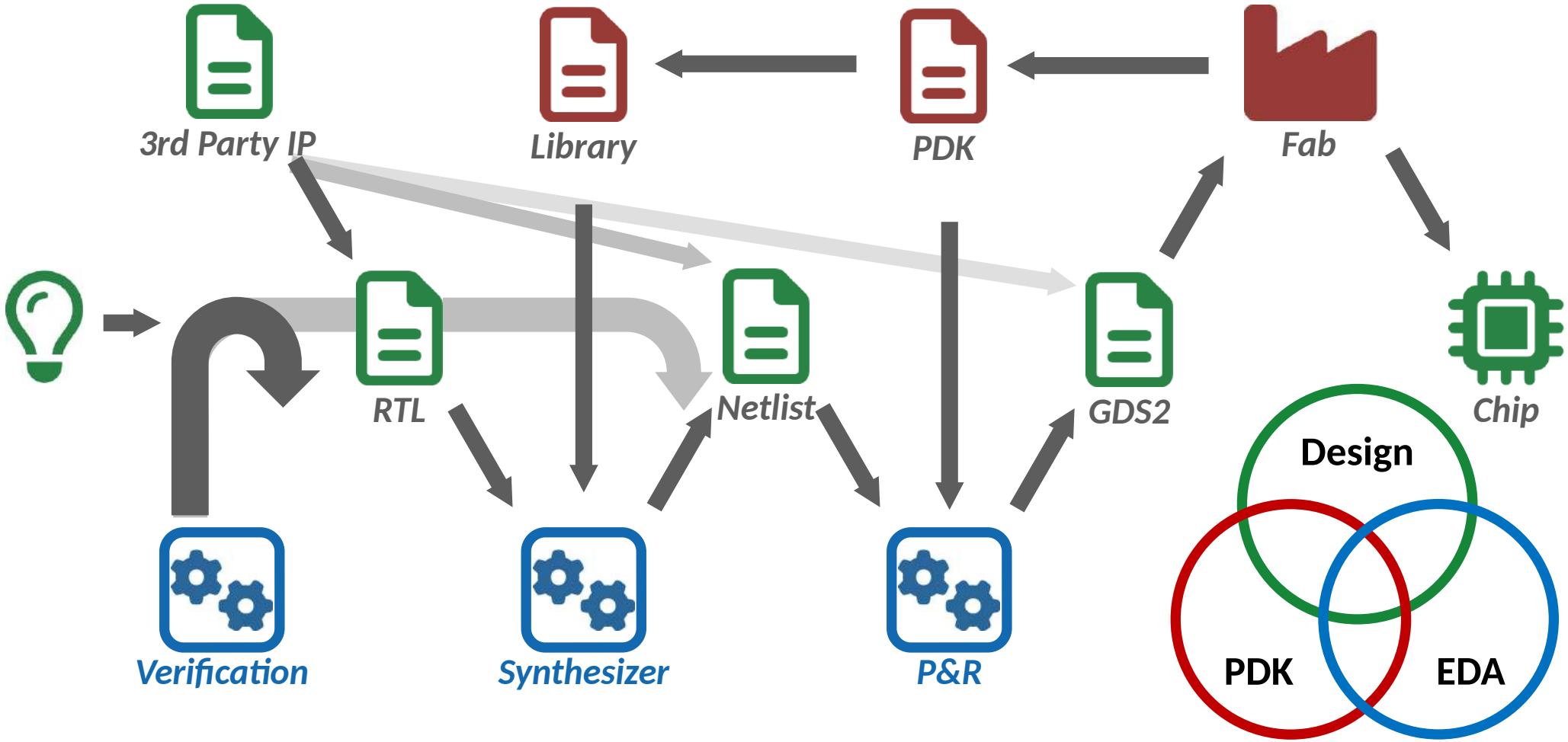


Manufacturing (PDK)

- Design rules for manufacturing (separation, minimum width of metals)
- Layer stack information for parasitics (thickness, dielectric constants..)
- Device models (SPICE parameters) for simulation

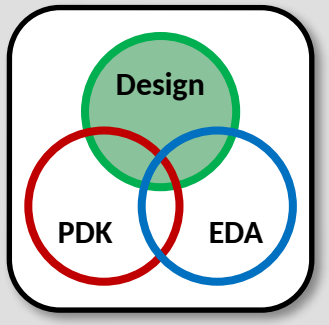
And it is important to have open solutions for all three phases as we will see

A simplified view of the IC design flow

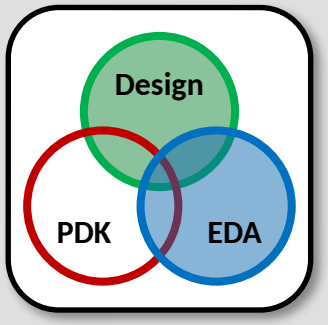


Notice that open parts in IC flow are independent

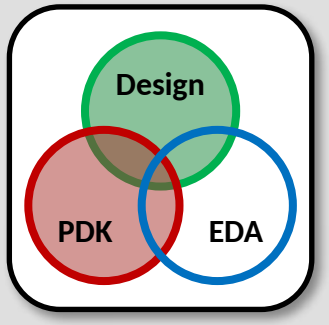
It is possible to use **open design** with closed EDA and closed PDK
(Picobello, Flamingo)



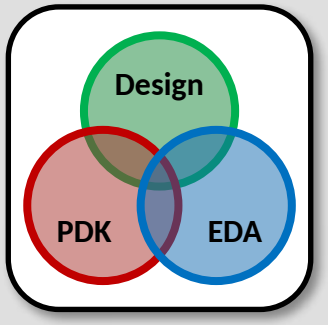
Or have **open designs** using **open EDA** on a closed PDK
(SeyrITA)



As well as having **open designs** using commercial EDA with **open PDKs**
(EZ Library)

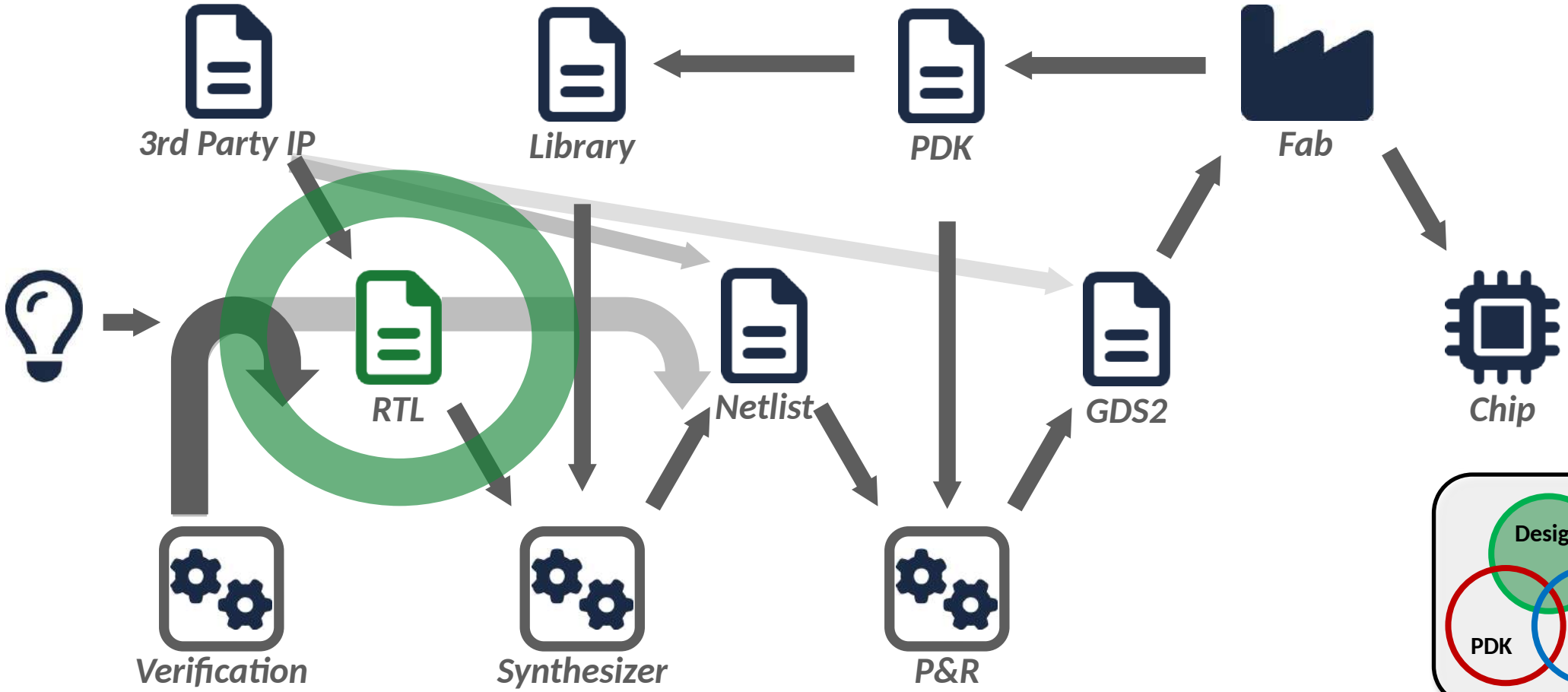


But end goal is to allow **open designs** using **open EDA** and **open PDKs**
(Basilisk, Mlem, Koopa)



On our journey at PULP we experienced all combinations

Most of open source hardware today is at **design** level



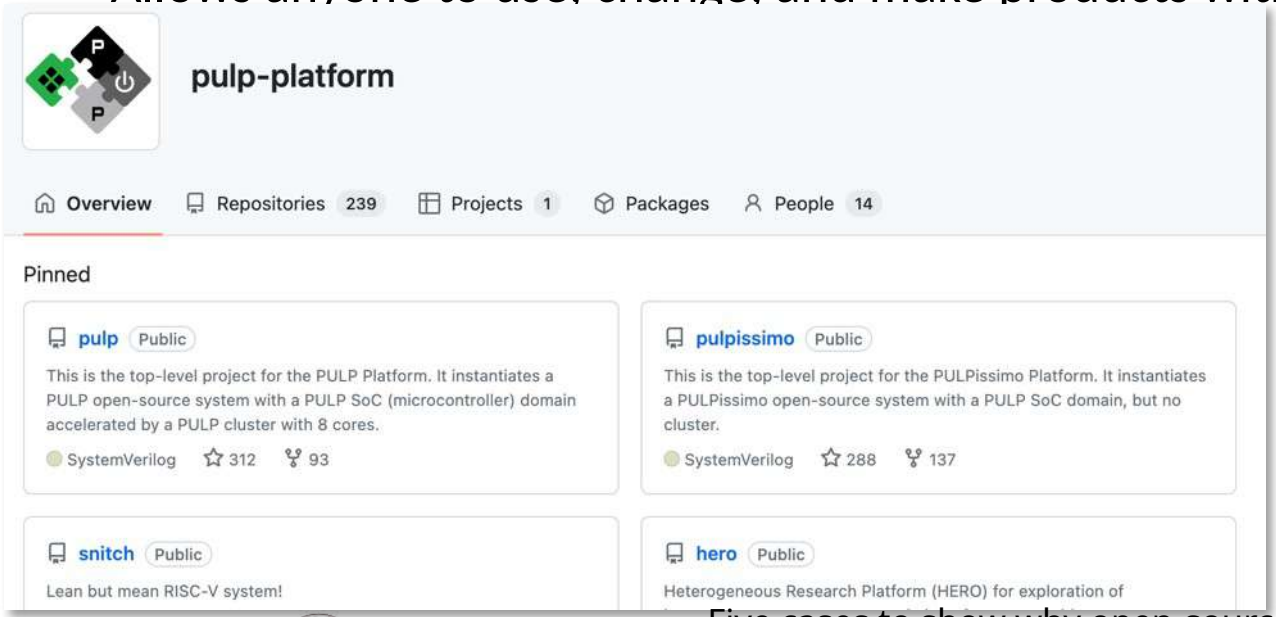
Icons taken from free icons from fontawesome.com

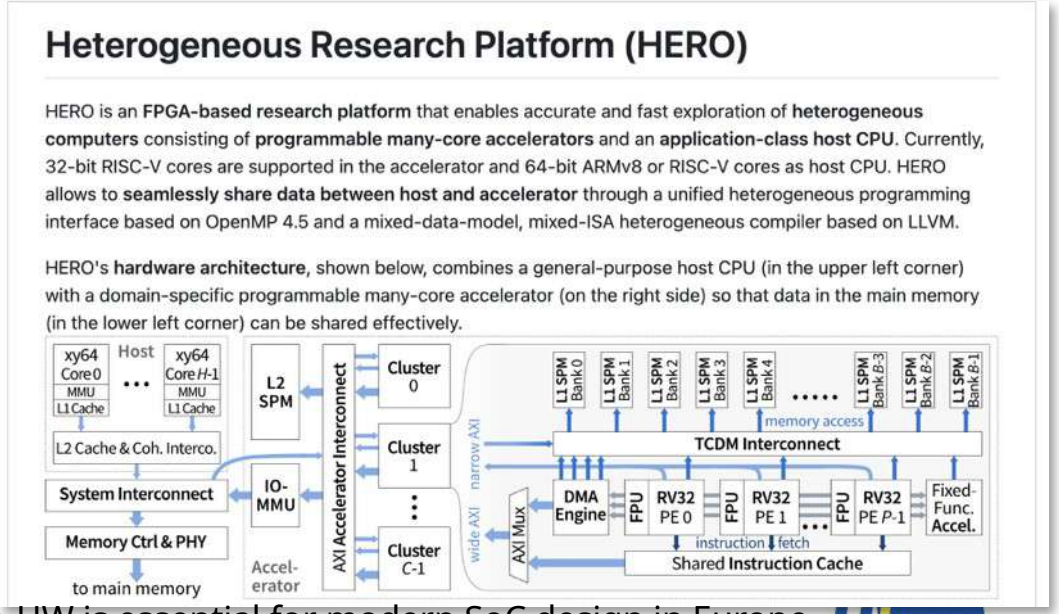
All of our designs are open-source hardware

- **All our development is on GitHub using a permissive license**
  - HDL source code, testbenches, software development kit, virtual platform

https://github.com/pulp-platform

- Allows anyone to use, change, and make products without restrictions.





What PULP provides is a box of building blocks

RISC-V Cores and Vector Units

RI5CY *CV32E*	Zero R *Ibex*	Snitch	Spatz	Ariane *CVA6*	ARA
RV32	RV32	RV32	RVV	RV64	RVV

Peripherals

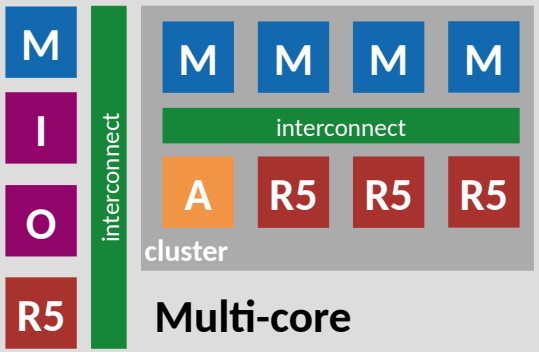
JTAG, SPI, UART, I2S, DMA, GPIO

Interconnects

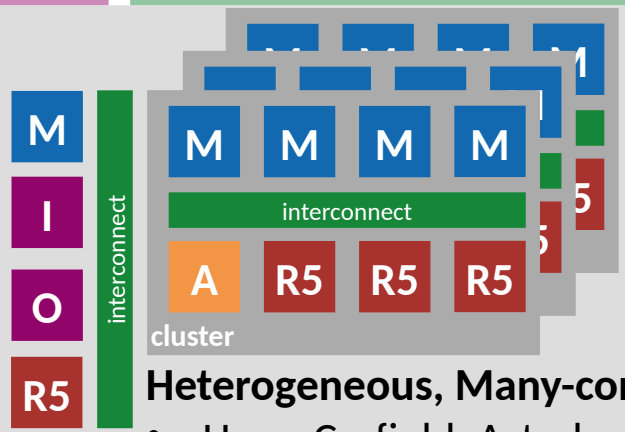
LIC, HCI, APB, FlooNoC, AXI4

Platforms

Single core
- Croc, PULPissimo
- Cheshire



Multi-core
- OpenPULP
- ControlPULP



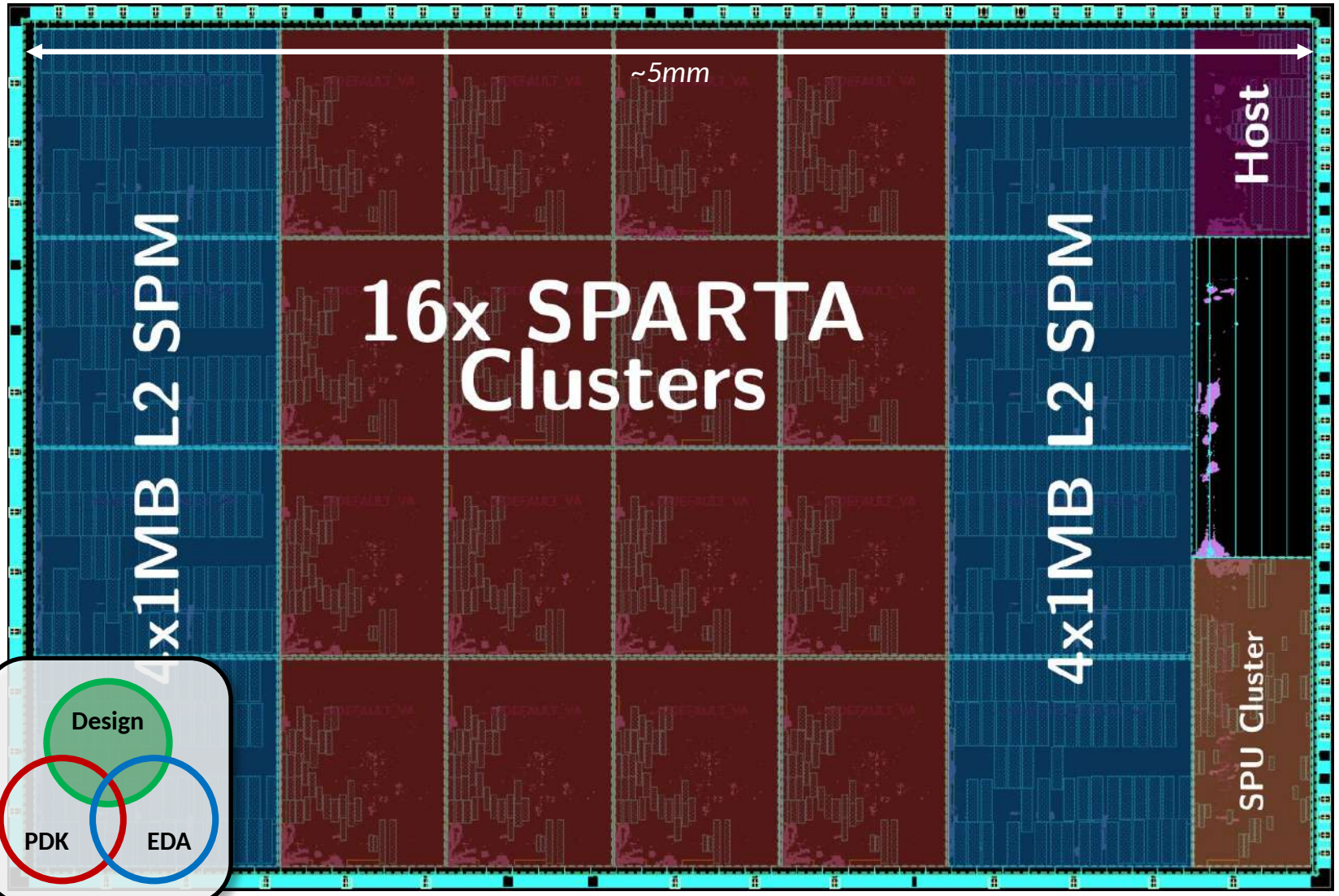
Heterogeneous, Many-core
- Hero, Carfield, Astral
- Occamy, Mempool

IOT → **HPC**

Accelerators and ISA extensions

XpulpNN, XpulpTNN	ITA (Transformers)	RBE, NEUREKA (QNNs)	FFT (DSP)	REDMULE (FP-Tensor)

Here is **Picobello**: our latest design in TSMC 7nm



- 16 SPARTA clusters totaling **144x** RISC-V cores with FP8-FP64-bit support
- **10MB** of on-chip SRAM
- Linux capable CVA6 Host
- Peripherals (JTAG, SPI, I2C)
- Running at 1+ GHz (WC),
  > 256 GFLOP/s (FP64)
  > 2 TFLOP/s (FP8)
- Currently being packaged
- Part of the EU Pilot project

THE EUPILOT

You need help to make large modern SoCs

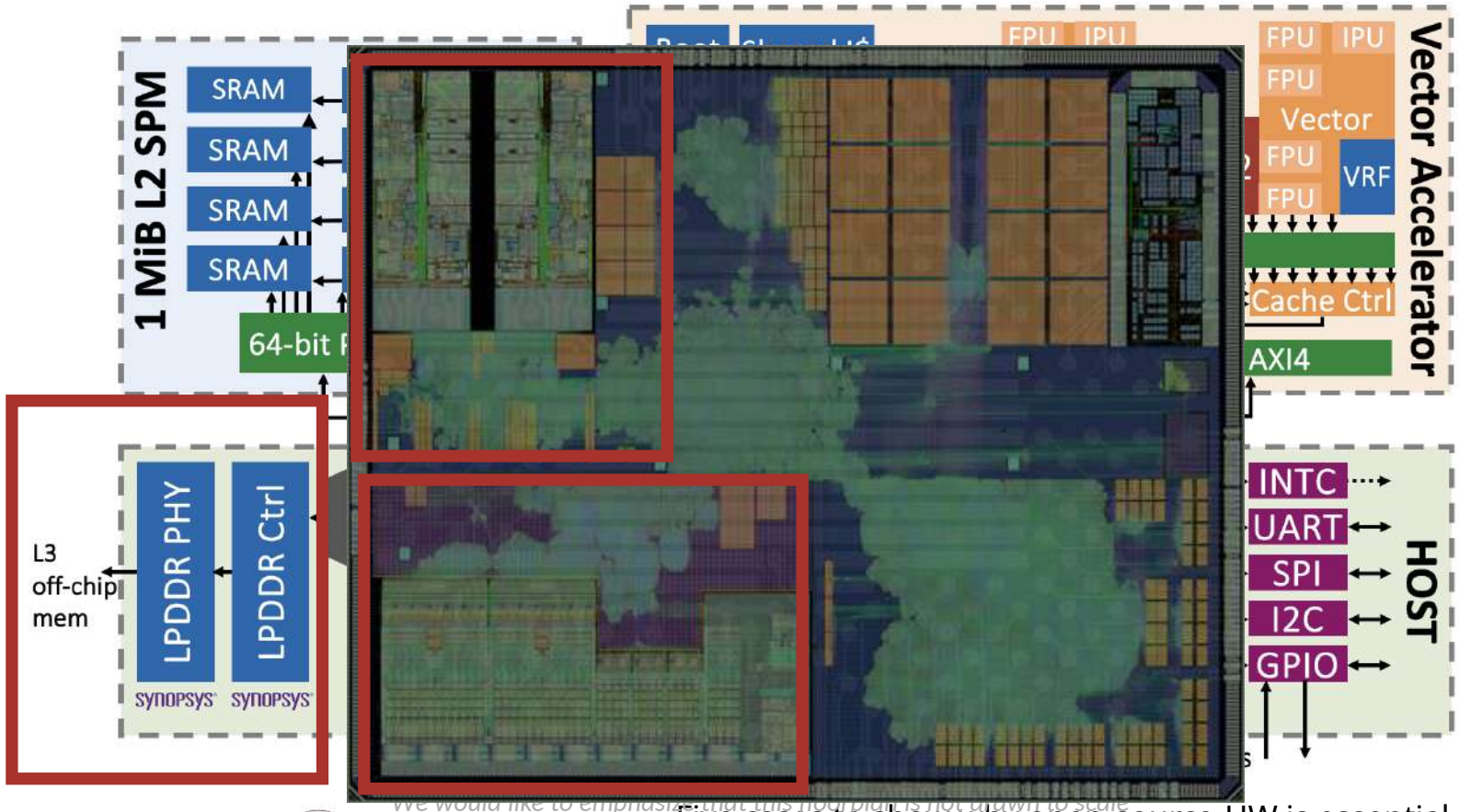
- **There are many innovative parts in Picobello**
  - Hopefully you will read about it in publications starting in 2026
- **But you can not afford to design all parts of an SoC from scratch by yourself**
  - **It builds on successful designs from the past**
    - CVA6 core : `https://github.com/openhwgroup/cva6`
    - Cheshire platform & peripherals : `https://github.com/pulp-platform/cheshire`
    - Snitch clusters : `https://github.com/pulp-platform/snitch_cluster`
    - FlooNoc : `https://github.com/pulp-platform/floonoc`
    - AXI : `https://github.com/pulp-platform/axi`
  - **Needs collaborations with experienced teams**
    - University of Bologna, TU-Munich

Supported by open-source designs in HDL

What about technology specific IPs? DDR, PCIe?

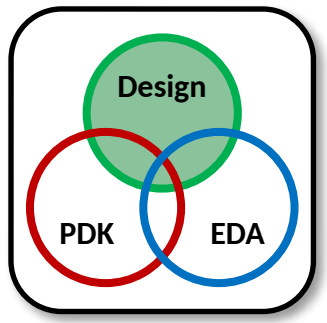
Finding such IP is not easy, you need great partners

- **Meet Flamingo**
  - SoC with Vector based accelerator for AI applications in GF22



GlobalFoundries™

- **EDA support by SYNOPSYS®**
- **Interface IP DDR / PCIe by SYNOPSYS®**



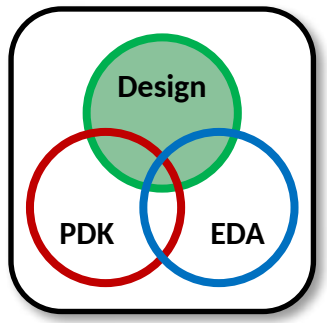
Finding such IP is not easy, you need great partners

- Meet Flamingo
  - SoC with Vector based accelerator for AI applications in GF22

What about technology specific IP we have developed?

GlobalFoundries™

- EDA support by SYNOPSYS®
- Interface IP DDR / PCIe by SYNOPSYS®



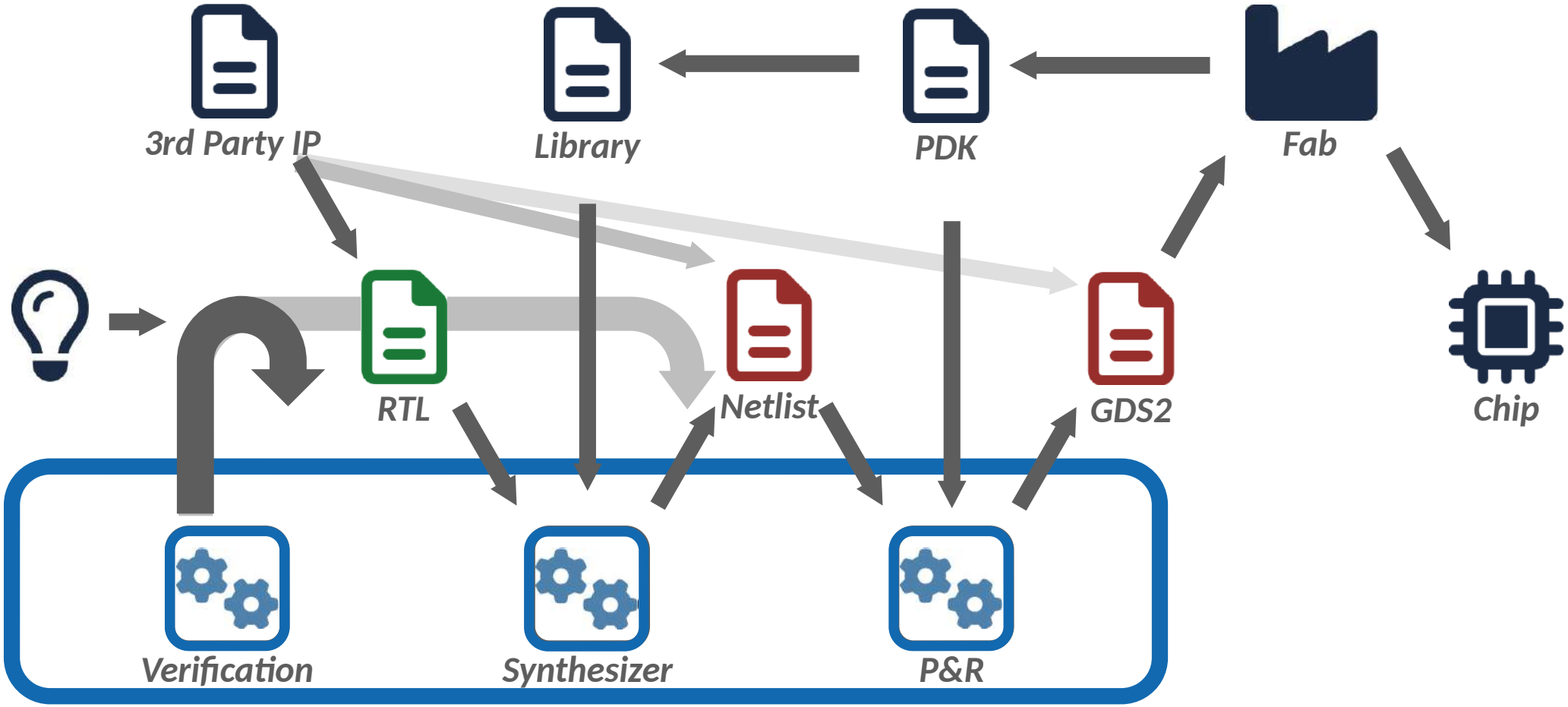
Sharing our **FLL** with others that need it

- **When you start designing in more modern technologies clock rates increase**
  - For 28nm and less clock rates of 500MHz – 2GHz are easily possible
  - It is difficult to bring such clocks externally, an internal clock generator could help
  - Good luck getting access to a low-cost clocking IP ☺
- **We designed an FLL (Frequency Locked Loop) back in 2016**
  - D. E. Bellasi and L. Benini, "Smart Energy-Efficient Clock Synthesizer for Duty-Cycled Sensor SoCs in 65 nm/28nm CMOS," in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 9, pp. 2322-2333, Sept. 2017, doi: 10.1109/TCSI.2017.2694322.
  - Ported and taped-out in many technologies: UMC65, TSCM65, GF22, GF12, TSMC7…

People have asked us repeatedly if we could share our FLL

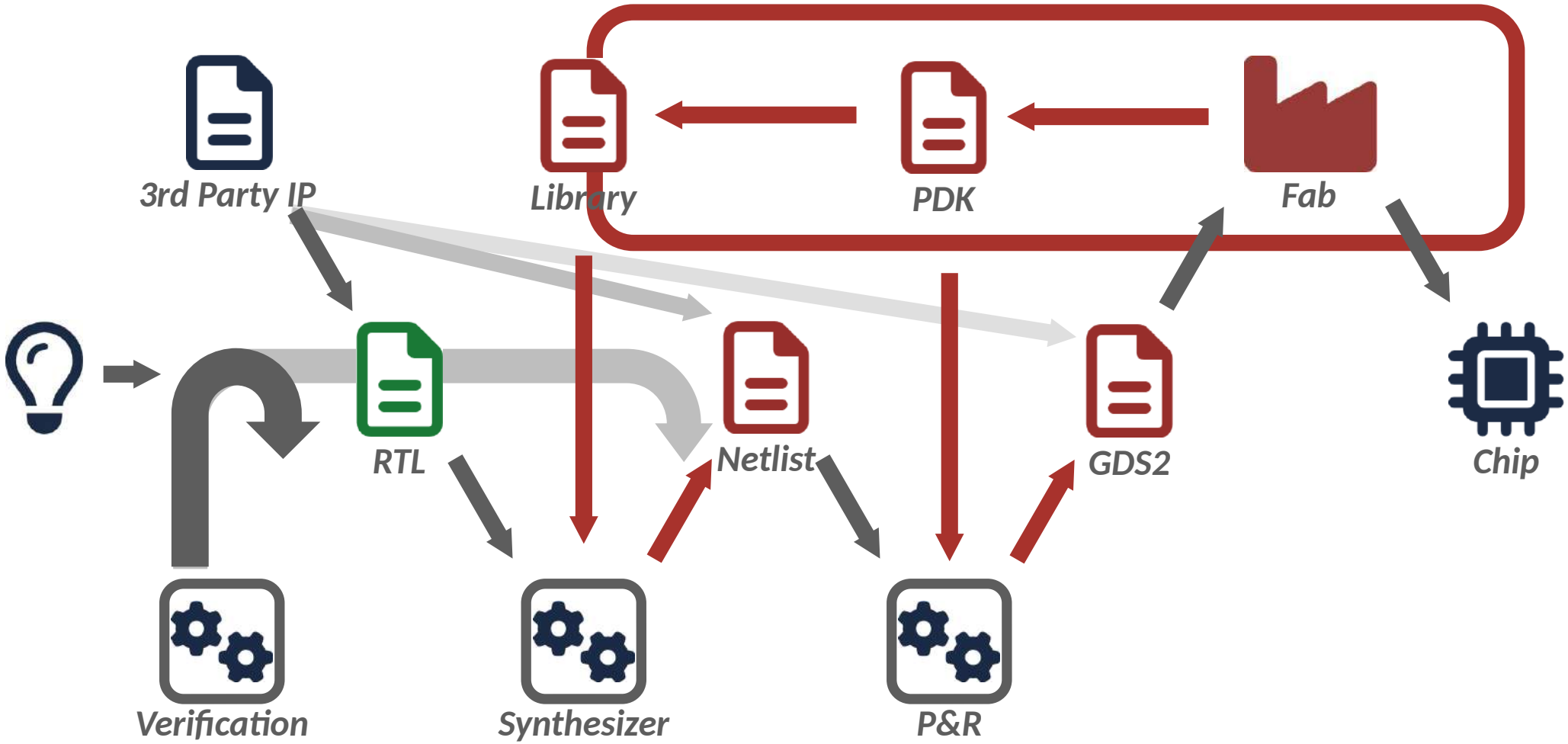
.. and we really want to share our FLL, we know how much it helped us

The output (and even scripts) of EDA vendors are closed



EDA vendors limit the output of their tools

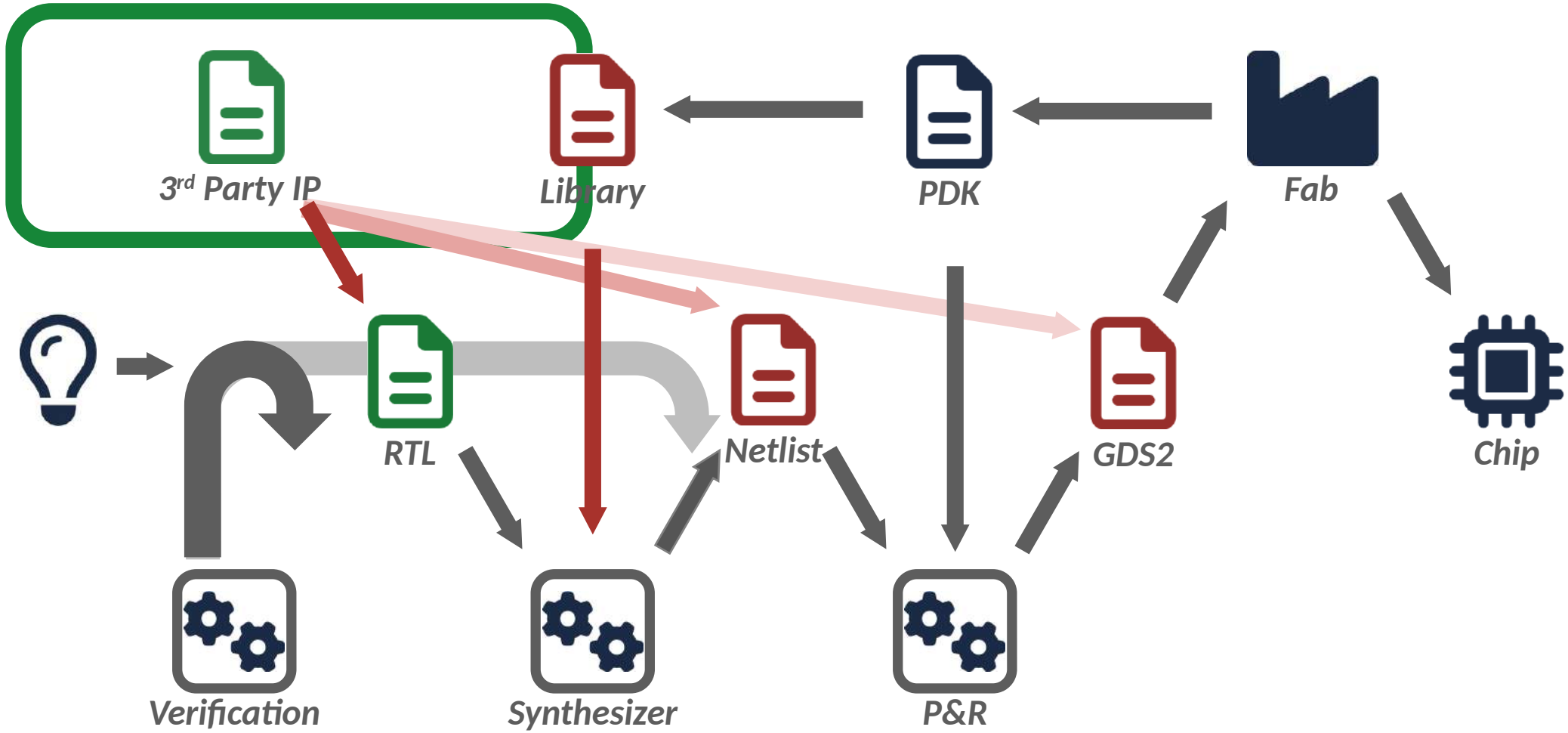
The chip will contain information from the PDK of the Fab



Fabs do not make PDK information accessible

essential for modern SoC design in Europe
- F. Gürkaynak

Most designs will include some 3rd party IP



3rd party IP when included can limit what can be open sourced

Issues of sharing technology specific IP (like our FLL)

- **Assuming you have no objections from your own institution to share your IP**

The design requires technology data that is under NDA

- You can not share anything without getting permission from the technology provider
- Usually this requires a multi-party NDA

Your design may contain standard cells that come from a different provider

- You will have to contact the IP provider to ask for permission
- Our FLL for GF22 uses INVECAS standard cell libraries now owned by Synopsys
- In theory you can send a design WITHOUT the cells only references, and add the cells locally

You have used EDA tools with academic licenses for your design

- Most academic license agreements would not allow you to transfer the output to others
- You need to contact the EDA vendor and ask for their permission to share

Issues of sharing technology specific IP (like our FLL)

- Assuming you have no objections from your own institution to share your IP

The design requires technology data that is under NDA

- You can not share anything without getting permission from the technology provider
- Usually this requires a multi-party NDA

Your design may contain standard cells that come from a different provider

- You will have to contact the provider to get permission
- Our FLL for GF22 uses INVECAS standard cell libraries now owned by Synopsys
- In theory you can send a design WITHOUT the cells and reference, and add the cells locally

You have used EDA tools with academic licenses for your design

- Most academic license agreements would not allow you to transfer the output to others
- You need to contact the EDA vendor to get permission to do so

This is a long and tedious process

On average it takes 6 months

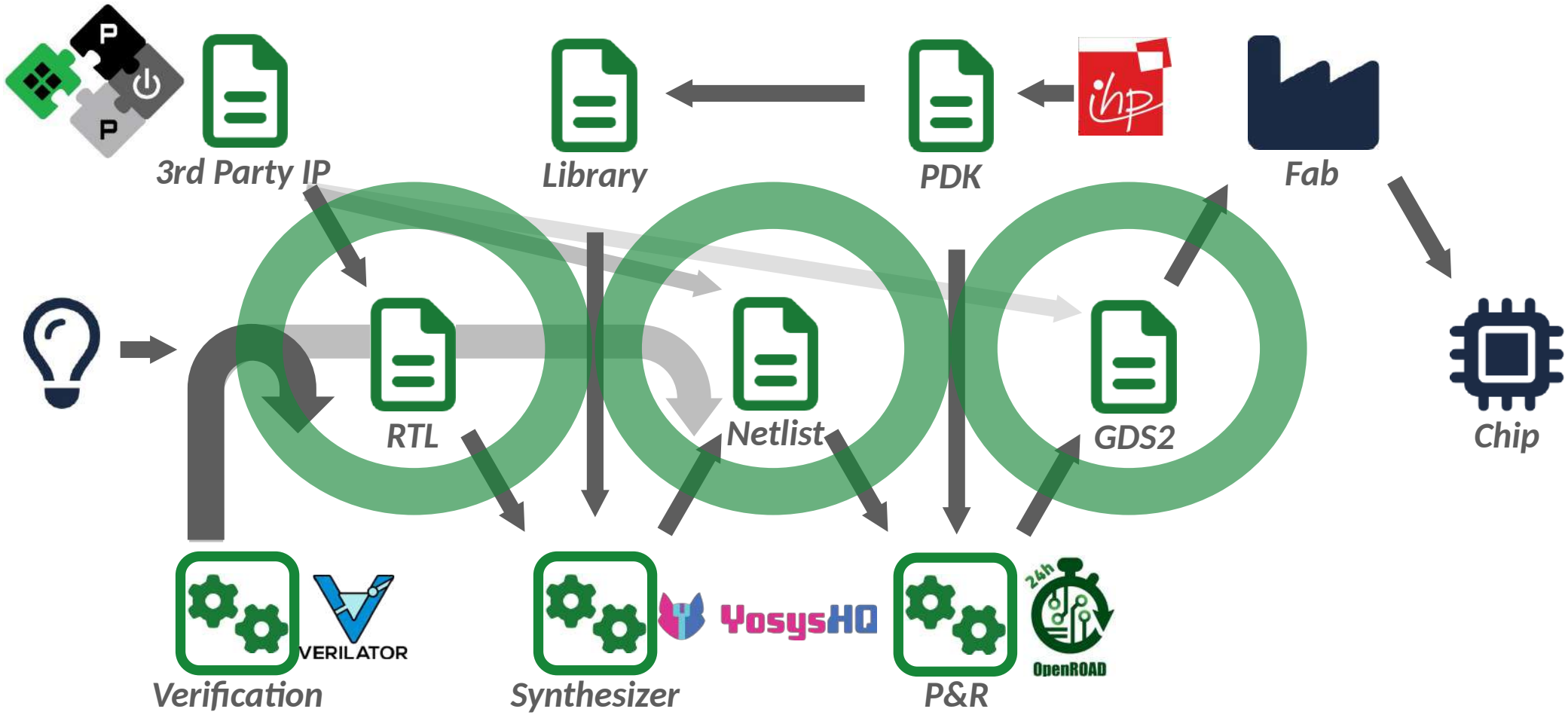
even with great support form everyone involved

Even worse this effort is

needed by the donor every single time

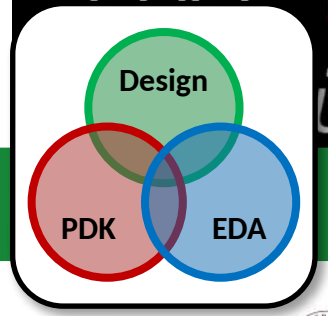
someone wants to use your IP

End-to-end Open-Source flow would help share IP



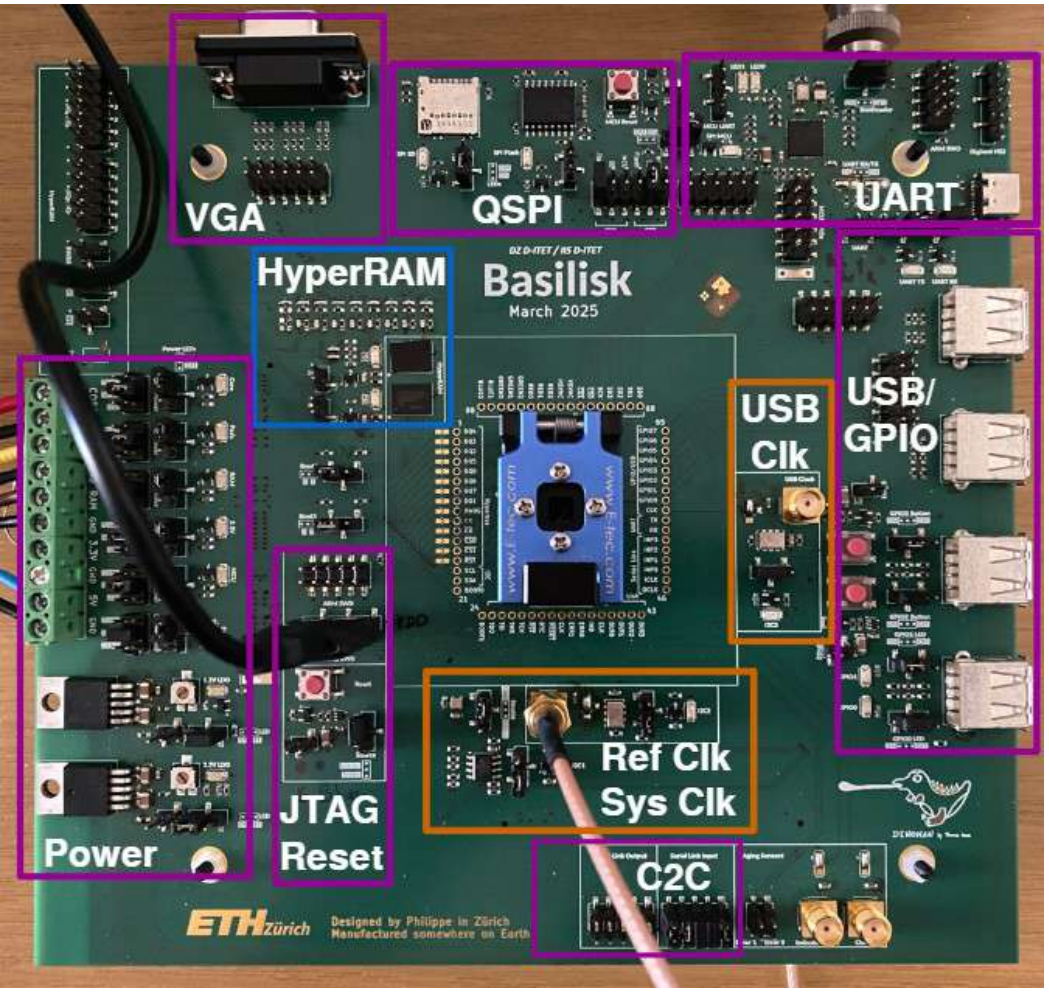
Meet Basilisk: End-to-end open SoC booting Linux

- **Open Design**
  - Cheshire with CVA6
- **Open EDA**
  - Yosys (with –slang)
  - OpenROAD
  - KiCAD (for board)
- **Open PDK**
  - IHP 130
  - Open standard cells
  - Memory macros



github.com/pulp-platform/cheshire-

Basilisk is proof that we can already make large(r) designs



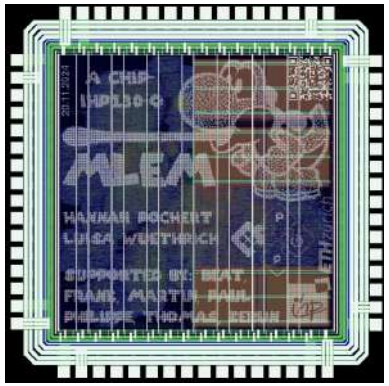
- **Designed in IHP 130nm OpenPDK**
  - **34mm²** (6.25mm x 5.50mm)
  - **~5× larger** than previous end-to-end OS designs
  - 2.7 MGE total, 1.14MGE logic
  - 24 SRAM macros (114 KiB)
  - **62MHz** at nominal voltage (1.2V)
- **RV64GC design runs Linux**
- **Active collaboration with**

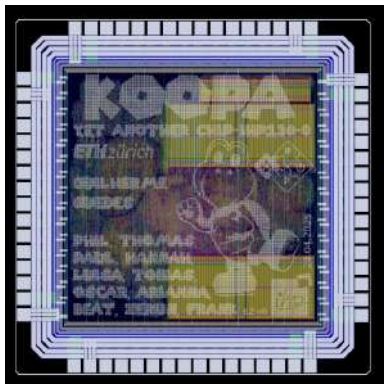
arxiv.org/pdf/2505.10060

At ETH Zürich, IC Design teaching uses open source HW

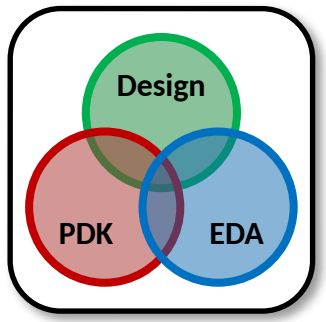
- **Starting 2025, our IC Design course switched to (mostly) open source**
  - Using IHP 130, Yosys and OpenROAD
    - Parts for backannotated simulation, test pattern generation, DRC/LVS, still use proprietary tools
- **Exercises based on a 32bit RISC-V microcontroller we call Croc**

https://github.com/pulp-platform/croc





- **Project based grading**
  - Students (in groups of two) will have to modify the exercise design
  - Best five designs will be taped-out
- **Huge advantage:**
  - We can share lecture and exercise content with everyone



https://vlsi.ethz.ch

Reliable collection of open source tools is essential!

- **Most open source developers have their own idea about the environment**
  - Getting several independent tools to work at the same time is very demanding work
  - Preparing/sharing lectures require you to be able to point to a reliable image

`https://github.com/iic-jku/IIC-OSIC-TOOLS` to the

README Apache-2.0 license

IIC-OSIC-TOOLS

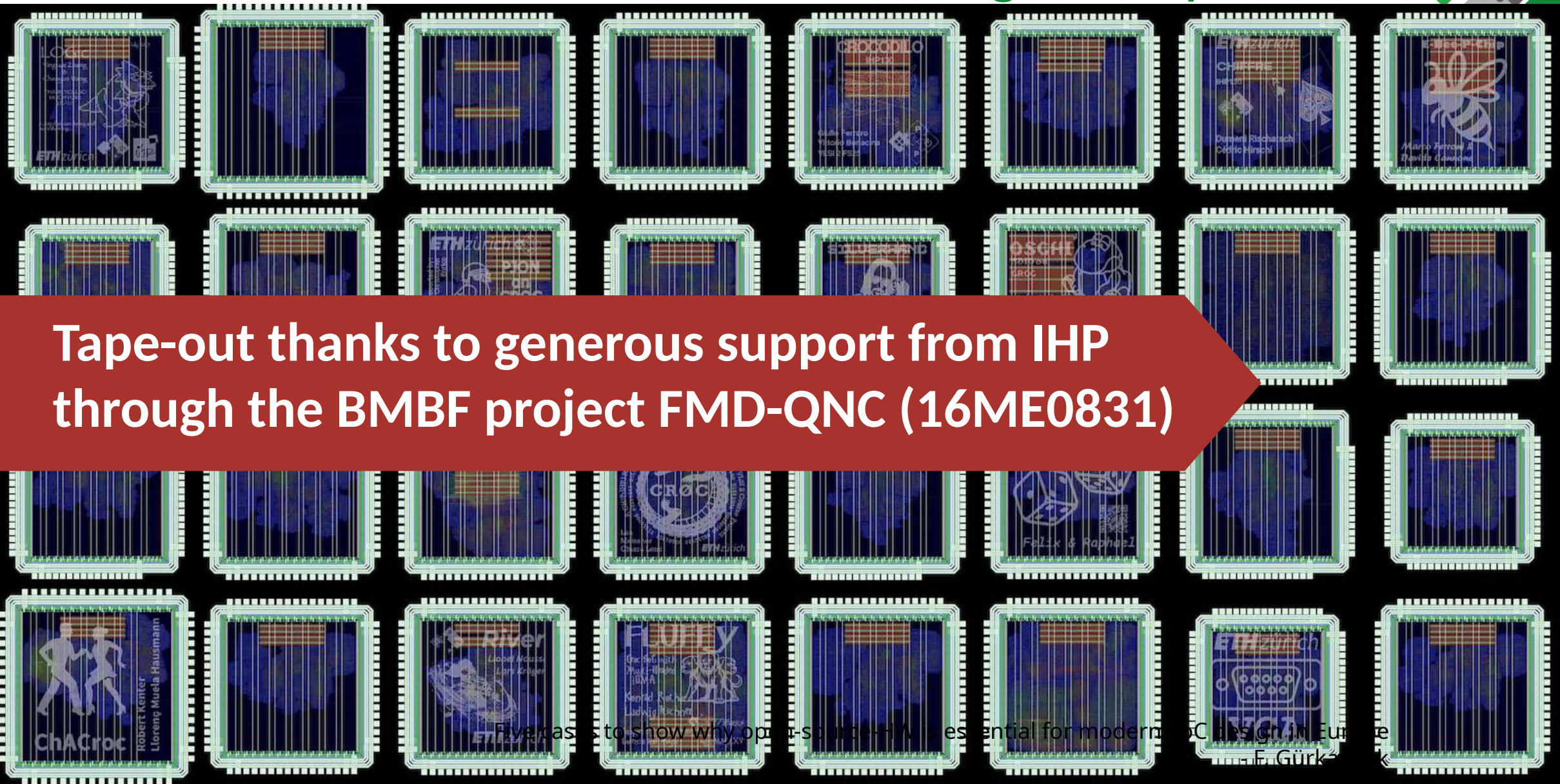
DOI 10.5281/zenodo.15044726

This environment is based on the efabless.com FOSS-ASIC-TOOLS.

IIC-OSIC-TOOLS (Integrated Infrastructure for Collaborative Open Source IC Tools) is an all-in-one Docker container for open-source-based integrated circuit designs for analog and digital circuit flows. The CPU architectures `x86_64/amd64` and `aarch64/arm64` are natively supported based on Ubuntu 24.04 LTS (since release `2025.01`). This collection of tools is curated by the Department for Integrated Circuits (DIC), Johannes Kepler University (JKU).

Thanks Harald ☺

And the students delivered: 33 valid designs, 5 taped-out



Tape-out thanks to generous support from IHP through the BMBF project FMD-QNC (16ME0831)

End-to-end open-source flows **REQUIRE** open PDKs

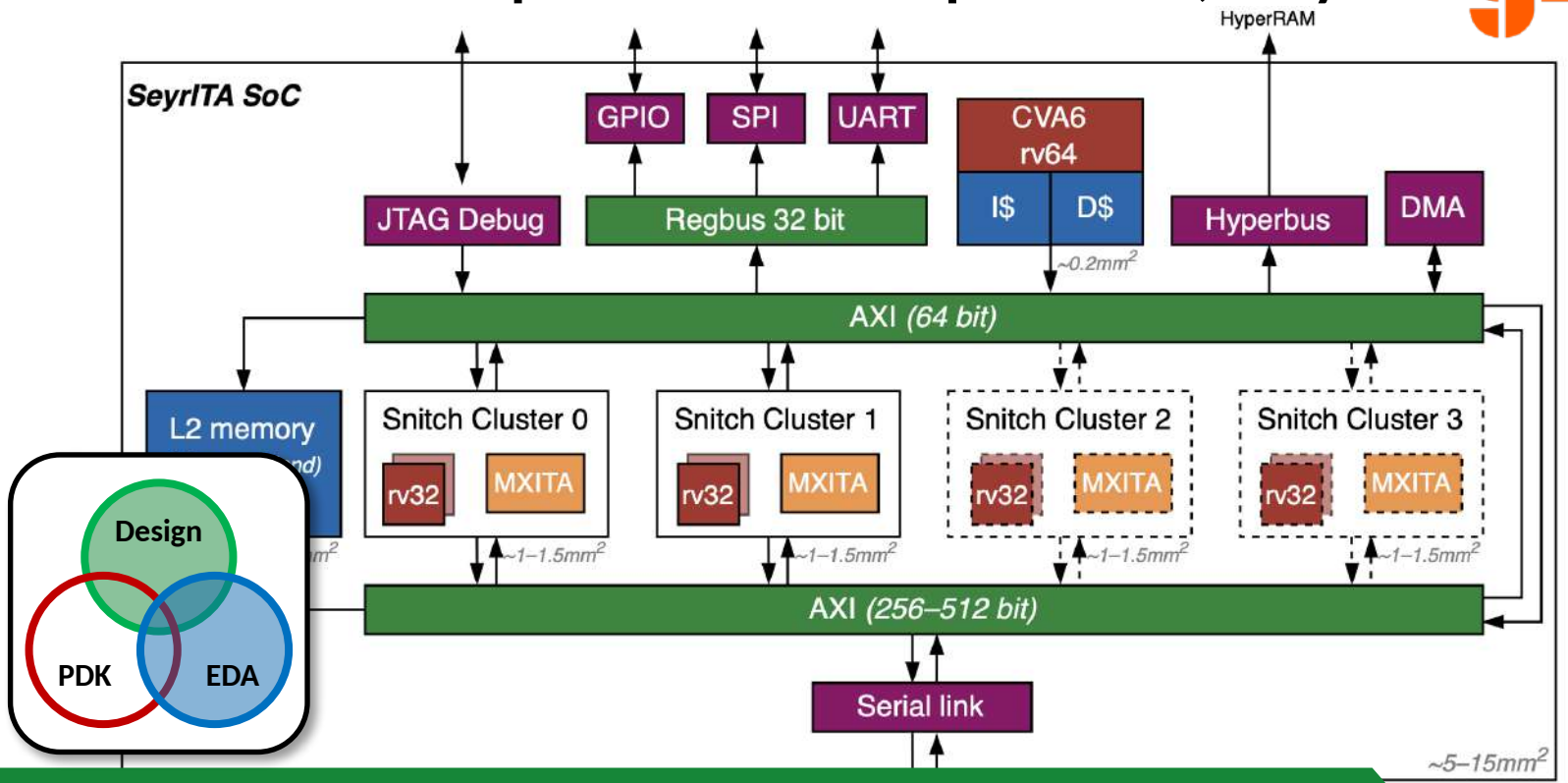
- **At the moment, the choice is limited**
  - Skywater 130nm
  - IHP 130nm
  - Globalfoundries 180MCU (aHigh Voltage technology that uses 500nm transistors)
- **Some are on the way**
  - IC Sprout 55nm (announced, but practical designs still a bit far away)
- **We need more open PDKs to design more relevant designs**
  - Something in the 65nm – 28nm range would be a **game changer**
  - Drafted an open letter to raise awareness with 300+ signatures

https://open-source-

What do we do until we have open PDKs in scaled technologies?

SeyrITA: our most ambitious project for embodied AI

- **Designing a research relevant SoC using open source EDA in GF22**
  - State of the art accelerator for embodied AI, using Integer Transformer Accelerator (ITA)
- **Collaborative open work with OpenROAD, Yosys**



- **Challenging work**
  - Large design, modern technology
  - We encounter problems daily
  - We try to solve them one problem at a time
  - Confident we will get it done
- **Target tape-out**
  - Mid 2026

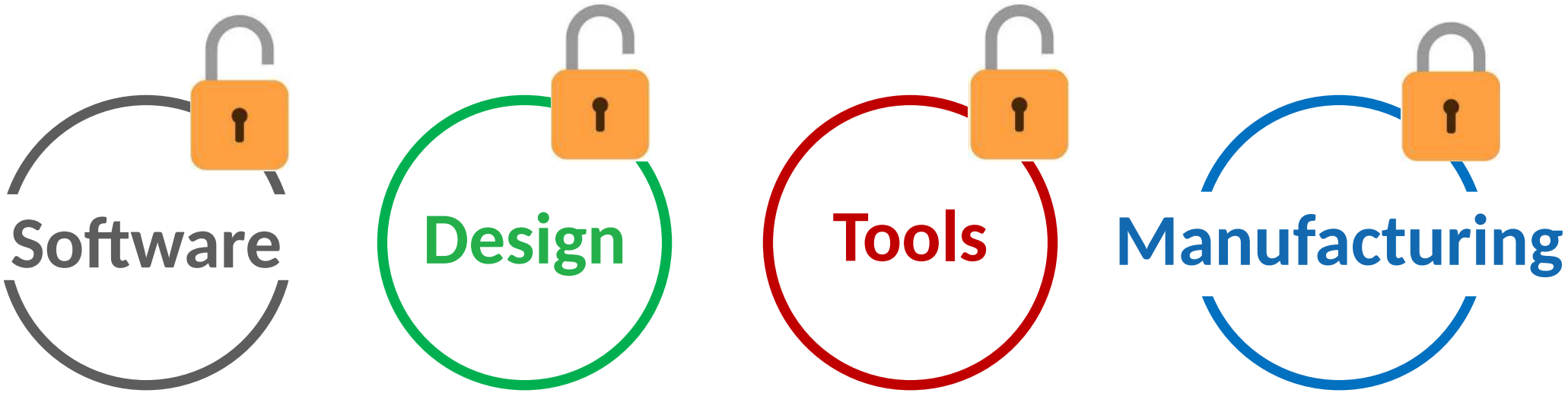
https://github.com/pulp-platform/

is essential for modern SoC design in Europe
- F. Gürkaynak

Open EDAs for more advanced technologies?

Low-Power, IoT, Automotive, ECUs, Wearables = Edge AI

Performance + Energy Efficiency is required!



HW-SW Co-Design

HW-SW & Tools Co-Optimization

Get the best from advanced nodes!

Open source EDA is not the enemy of commercial EDA

- **Open-source EDA allows developing skills needed for EDA companies**
- **Allows EDA companies to concentrate their efforts on differentiating features**
  - i.e. no need to develop waveform viewers
- **Lowering barriers: OS EDA allows you to experiment before investing**
  - You can see if your company can venture into IC Design by experimenting at your own pace
  - Try out how an IC design process fits your company, identify the gaps, judge the benefits
  - The more SMEs that venture into IC design the more EDA licenses will eventually be sold.
- **Not being limited by license costs but by available CPUs gives opportunities**
  - Makes public (or cross partner) CI flows much easier
  - Being able to run many iterations (with small variations) opens new possibilities
  - Early evaluations of technical choices do not require signoff accuracy

Open Source EDA for Europe: **ODE4EC**

- **HORIZON-JU-CHIPS-2025-IA-EDA-two-stage proposal**
  - 20MEUR funding from EU, total project **50MEUR**
  - Organized in three sub projects: Digital, Analog, Productivity
  - Currently in Grant Preparation Phase.
  - Start in April/May 2026, more details to follow
- **Large consortium**
  - 24 partners (DIG), 27 partners (AMS), 24 partners (PIV)
  - From 14 Countries (AT, DK, FI, FR, DE, GR, HU, IT, LT, PT, SI, ES, SE, SE, CH, UK)
  - Includes broad participation from most open source contributors in EU
- **Great opportunity to make a difference**

The five lessons from our experience

- **Open source is needed to share designs and ideas efficiently**
  - And we can not build everything ourselves, modern SoCs are way to complex
- **Sharing beyond RTL level requires both open EDA, and open PDKs**
  - There are several roadblocks that prevent sharing in IC Design
  - These can only be addressed once there are PDKs that allow sharing
- **End-to-end open-source flows can already deliver complex working chips**
  - There is a gap to commercial EDA, but it is less than what people think
- **End-to-end open-source flows are great for training and teaching**
  - Being able to share all aspects of the flow allows us to make materials available to everyone
- **There are many interesting opportunities for open EDA**
  - There is more to do than just replicate what commercial EDA has been doing
  - Opportunities for disruptive developments that will push performance beyond current levels

http://pulp-platform.org @pulp_platform

There is work to do, let’s go !

End-to-end Open-Source IC Design is already working!

Easier collaboration / sharing

- Need to stand on the shoulder of giants
- Share common parts that all need
- Concentrate work/time where it matters

Open reproducible results

- Everyone can verify performance claims
- Allows us to generate example datasets that can be used to train/improve tools

Reduce entry barriers for all

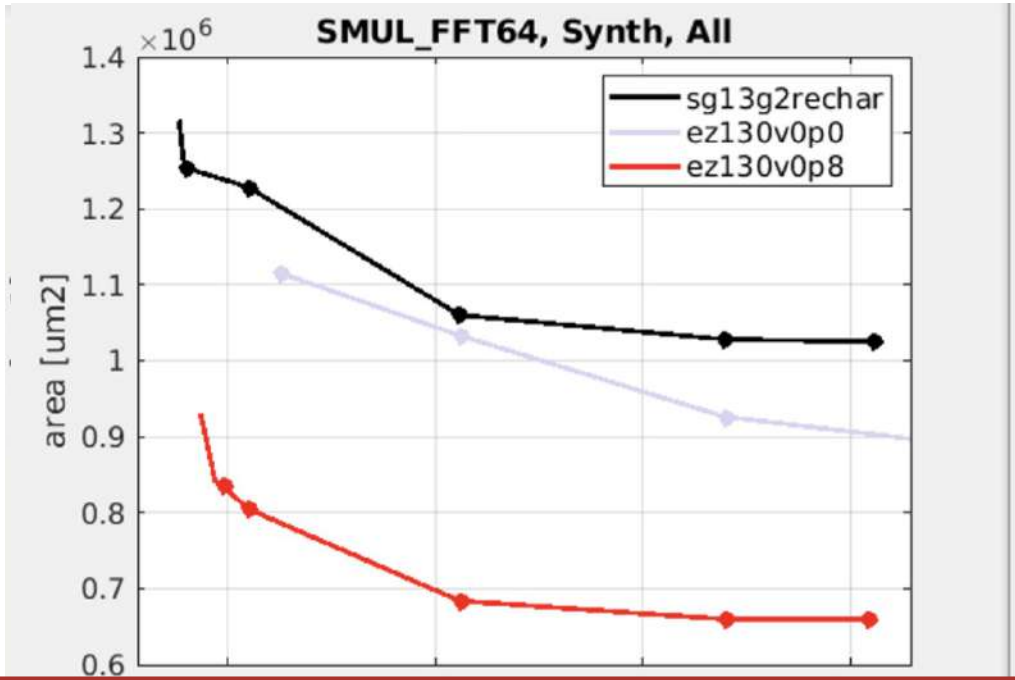
- You can easily get started with IC Design
- No agreements needed to get started
- Can then decide to stay open or not

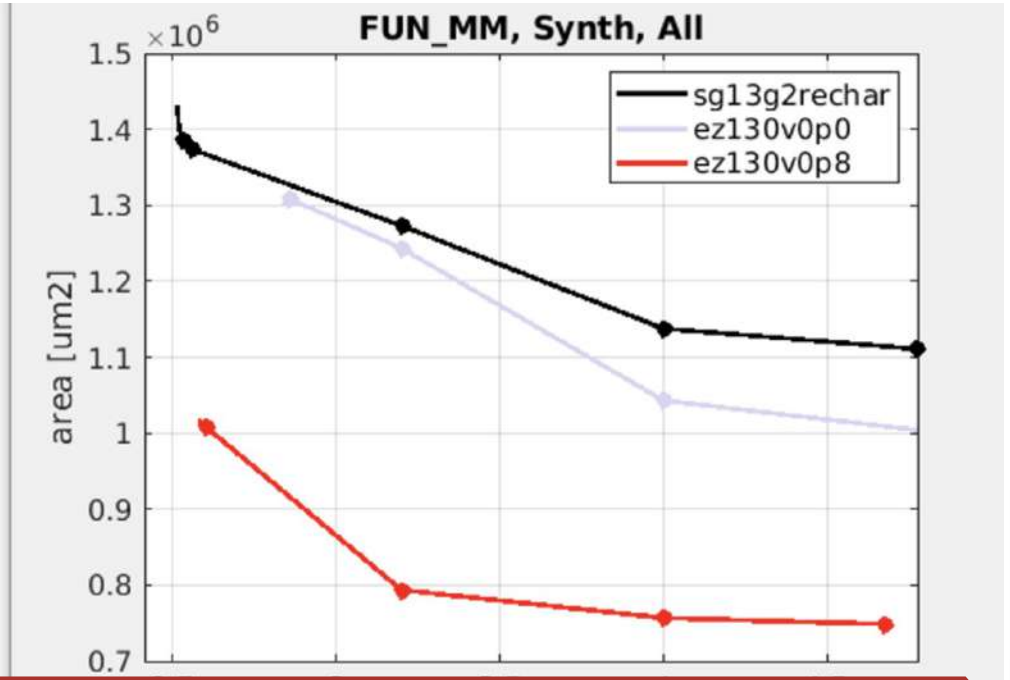
Accessible teaching for all

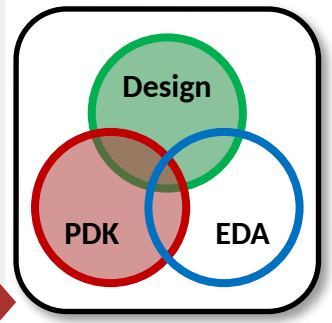
- Share courses, designs, examples
- Create tutorials, knowledge bases
- Training for industry

EZ Library: new standard cells for IHP130

- **VLSI 5 lecture by the Integrated Information Processing group at ETH Zürich**
  - By Oscar Castañeda Fernández, Christoph Studer and a bit of support from me
- **In one semester, 18 students re-designed a standard cell library for IHP130**







Needed additional negotiation with EDA tool vendor for release (granted)