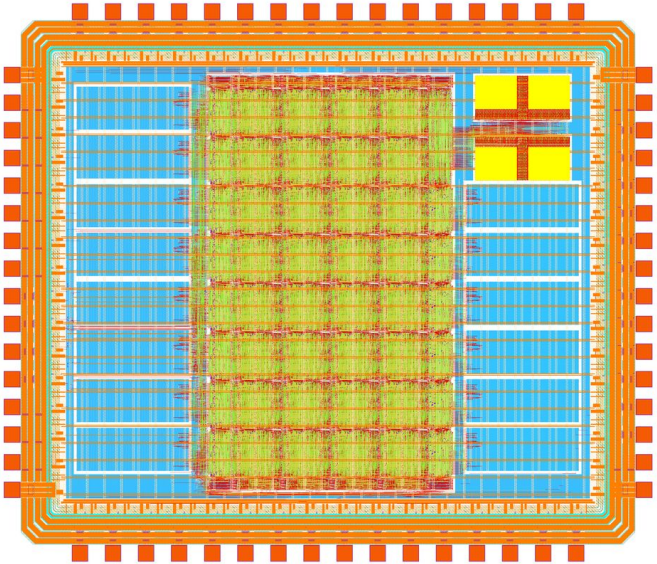

# HeiChips 2025 Hackathon

Introduction and Submission Process

Leo Moser

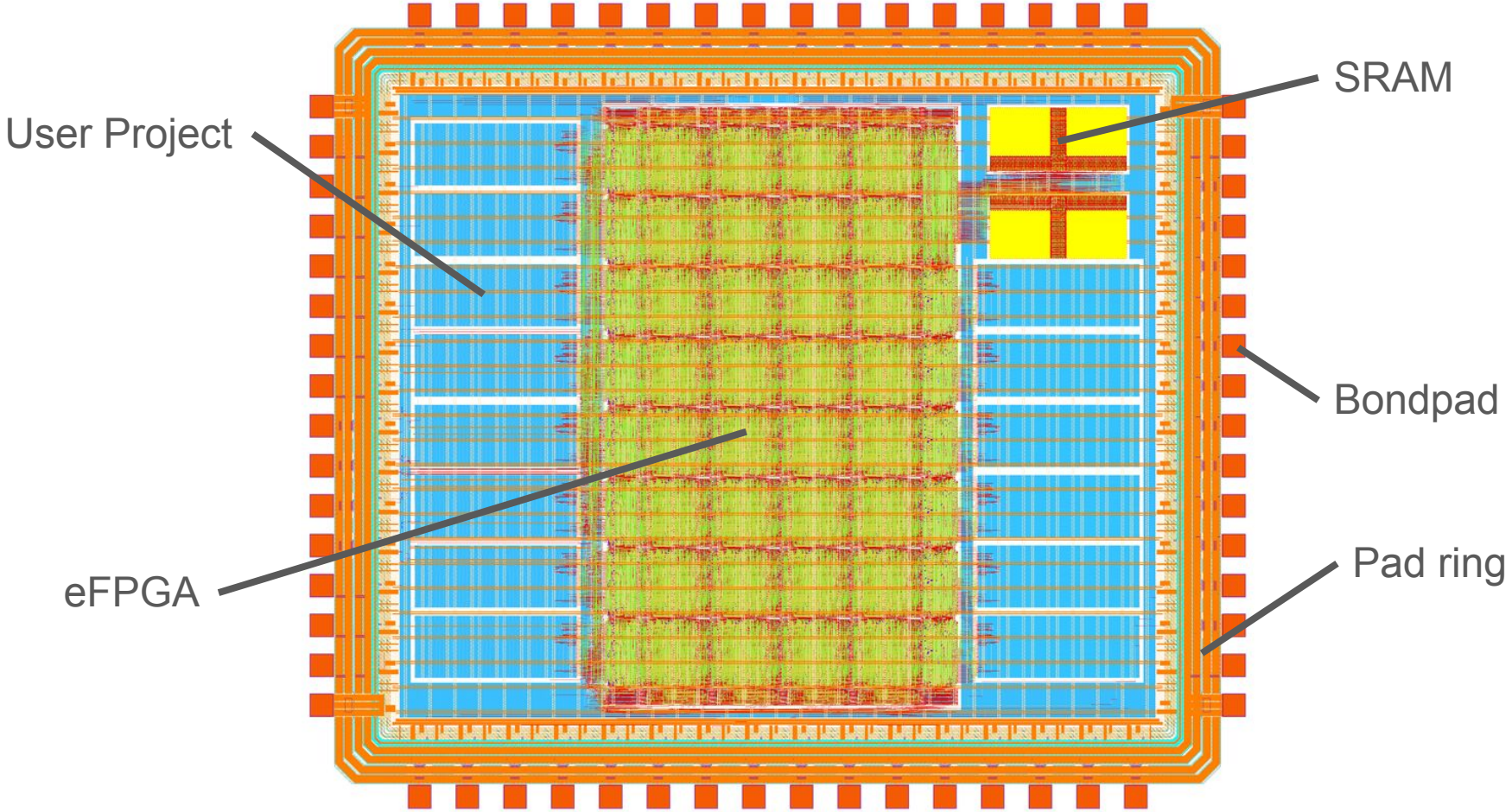
# HeiChips 2025 Tapeout!

- Our own chip using the IHP Open PDK
- Embedded FPGA in the center
- 8 slots on both sides (left/right)
- User Projects
  - Small: 1 slot
  - Large: 2 slots
- 4 KiB SRAM (2 slots)
- Repository:
  - https://github.com/FPGA-Research/heichips25-tapeout
- Bitstream generation:
  - main/ip/fabric/user_designs

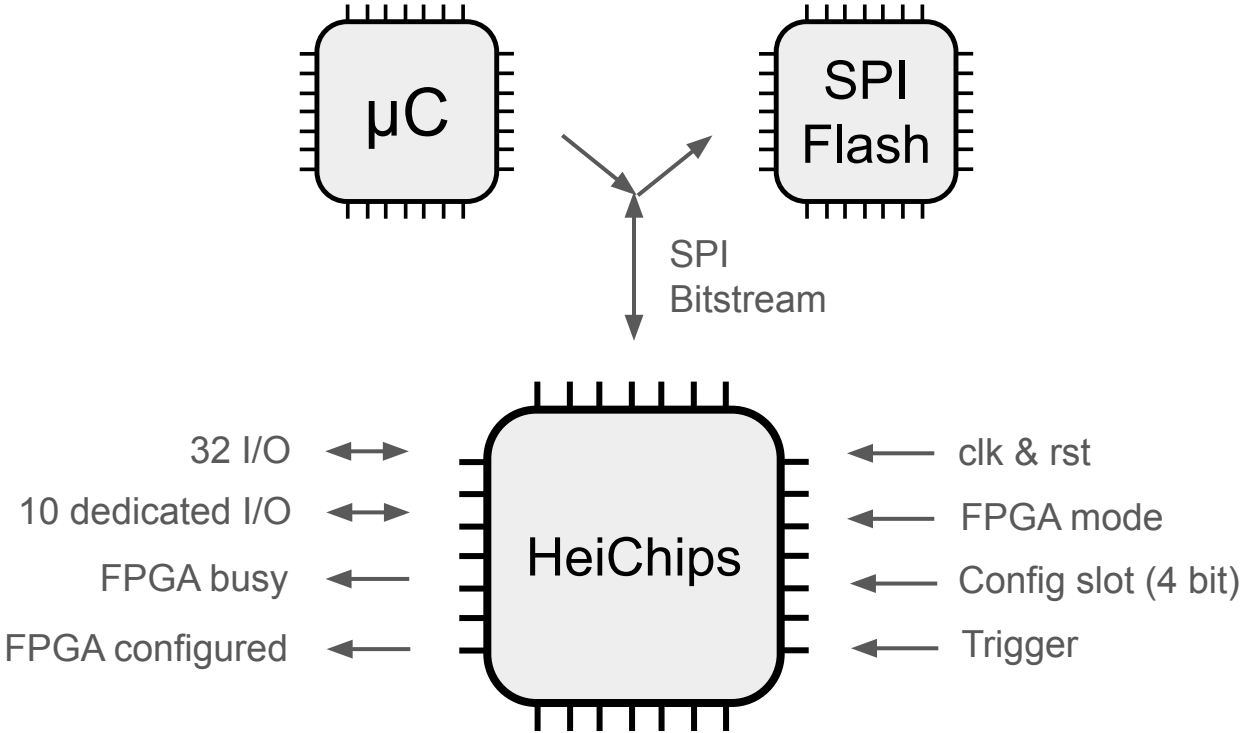


HeiChips 2025 Tapeout

256 LCs, 16 slots, ~8mm²

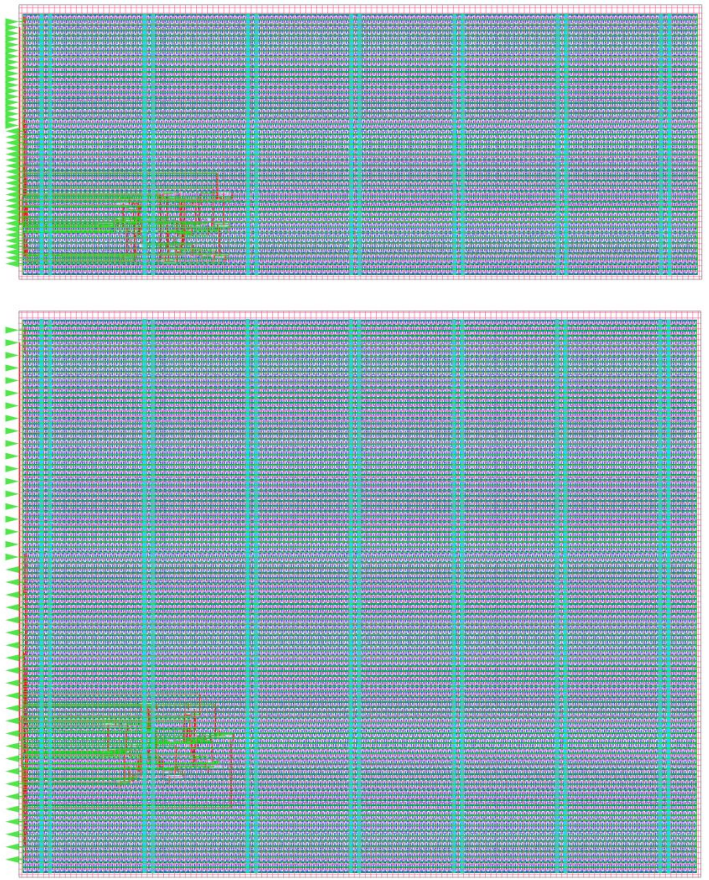

User Project
SRAM
Bondpad
eFPGA
Pad ring

# Chip I/Os


µC
SPI Flash
SPI
Bitstream
32 I/O
10 dedicated I/O
FPGA busy
FPGA configured
HeiChips
clk & rst
FPGA mode
Config slot (4 bit)
Trigger

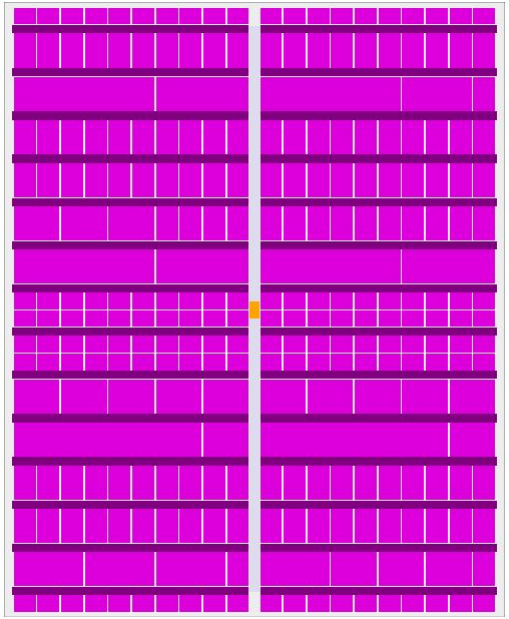
# HeiChips User Projects

- Two choices: Small & Large → ask us!
- Same interface as Tiny Tapeout
  - 10 inputs, 8 outputs, 8 bidirectional I/Os
- Connects to the I/O pads through the eFPGA fabric
- Need dedicated pins?
  - High-speed digital
  - Analog-mixed signal
  - 10 pins are available→ ask us!



# Tiny Tapeout: A Shared Silicon Tapeout Platform

- Many small designs in one chip
- One design can be active at once (Multiplexer)
- Interface:
  - 10 inputs, 8 outputs, 8 bidirectional I/Os
- Designs are power-gated
- You pay for a number of tiles
  - Tiles can be merged together
- You pay for the PCB + Chip
- https://tinytapeout.com



# HeiChips User Projects **vs.** Tiny Tapeout Tile

1x Tiny Tapeout Tile: 202.08um x 154.98um

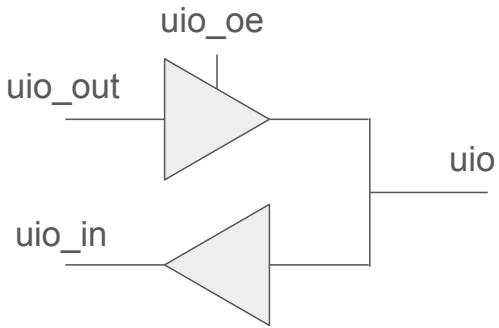
HeiChips User Project Small: 500um x 200um → **~ 3.2 TT Tiles**

HeiChips User Project Large: 500um x 415um → **~ 6.6 TT Tiles**

# Tiny Tapeout Pinout

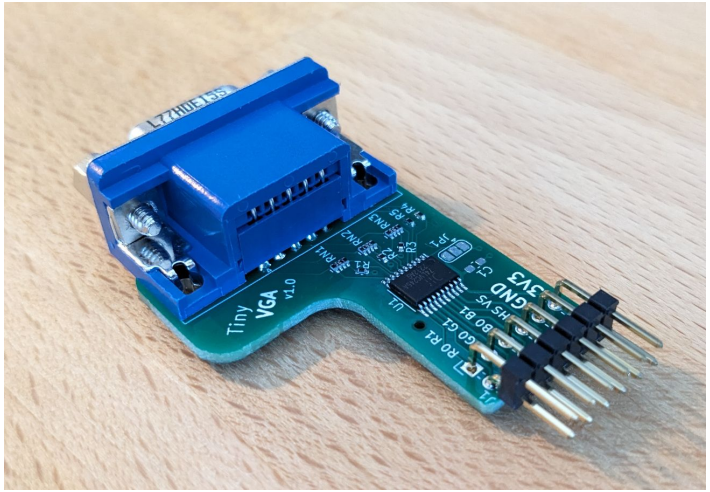
- Pinout
  - 10 input pins — clk, rst, 8x ui_in
  - 8 output pins — 8x uo_out
  - 8 bidirectional pins — 8x uio_in, 8x uio_out, 8x uio_oe
  - always 1 — ena (just ignore it)
- Common pin assignment
  - Makes it easier to test other designs
  - https://tinytapeout.com/specs/pinouts
- Standardized Pmods
  - VGA, QSPI, Audio etc.
  - You can buy them: https://store.tinytapeout.com

```
module heichips25_template (
    input  wire [7:0] ui_in,
    output wire [7:0] uo_out,
    input  wire [7:0] uio_in,
    output wire [7:0] uio_out,
    output wire [7:0] uio_oe,
    input  wire       ena,
    input  wire       clk,
    input  wire       rst_n
);
```



# Tiny VGA

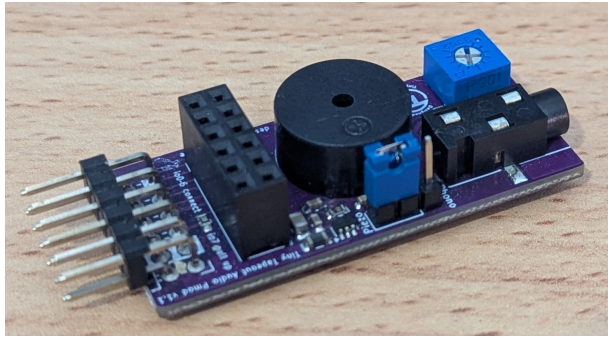
- Pmod (Standardized interface)
- Repo: https://github.com/mole99/tiny-vga
- 2 bit per color (RGB), hsync & vsync
- More colors?
  - dithering: temporal & spatial
- TT demoscene
  - Incredible entries!
  - Nyan Cat, Warp, Drop, and more!
  - Demoscene TT08
- I have some Tiny VGAs with me!




https://github.com/mole99/tiny-vga

# Audio Pmod

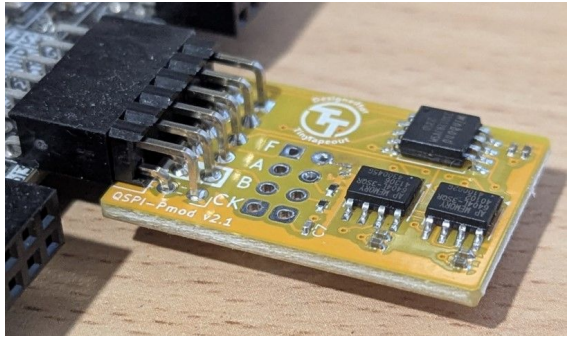
- Piezo speaker
- Headphone jack



https://github.com/MichaelBell/tt-audio-pmod

# QSPI Pmod

- 1x 128M-bit QSPI Flash
- 2x 64M-bit QSPI PSRAM

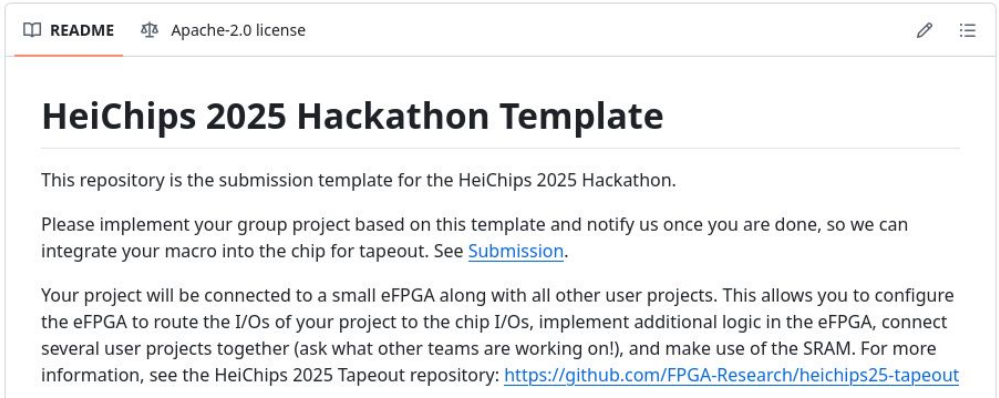



https://github.com/mole99/qspi-pmod

# HeiChips 2025 Hackathon Template

## Everything is Open Source!

- Simulation and Verification
- Emulation on FPGA
- Physical Implementation using LibreLane


README Apache-2.0 license

HeiChips 2025 Hackathon Template

This repository is the submission template for the HeiChips 2025 Hackathon.

Please implement your group project based on this template and notify us once you are done, so we can integrate your macro into the chip for tapeout. See Submission.

Your project will be connected to a small eFPGA along with all other user projects. This allows you to configure the eFPGA to route the I/Os of your project to the chip I/Os, implement additional logic in the eFPGA, connect several user projects together (ask what other teams are working on!), and make use of the SRAM. For more information, see the HeiChips 2025 Tapeout repository: https://github.com/FPGA-Research/heichips25-tapeout

https://github.com/FPGA-Research/heichips25-template

# Installation of the Tools

- Using Nix
  - Native Installation: Linux (x86_64, aarch64) and macOS (x86_64, arm64)
  - Windows: Emulation (e.g. WSL2)
  - Based on: [fossi-foundation/nix-eda]
- Follow the installation instructions
- To enable the tools:
  - Run: `nix-shell`
- For openXC7 Toolchain
  - `cd nix-openxc7 && nix-shell`

# Simulation & Verification

## Simulators

- Icarus Verilog
- Verilator

## Testbench environment

- cocotb
  - Python-based
  - simulator-agnostic
  - Easy to use :)



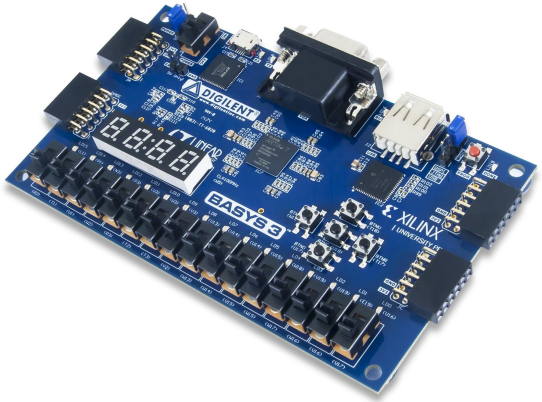
# FPGA Prototyping

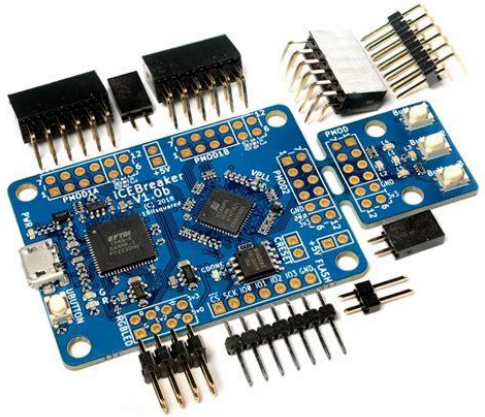
## Open Source Toolchain

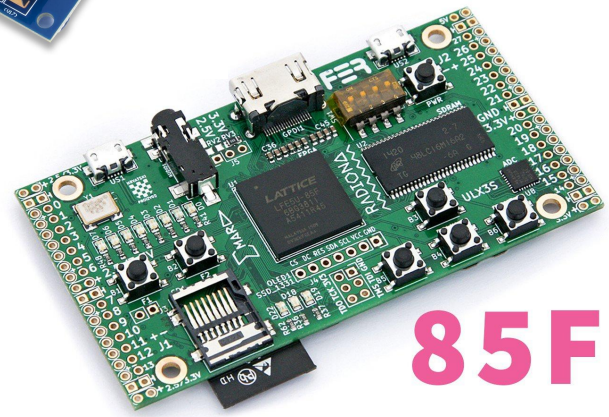
- Yosys - synthesis tool
- nextpnr - place and route
- openFPGALoader

## Supported Boards

- iCEBreaker
- ULX3S
- iCE40HX8K-EVB
- Basys 3
- …



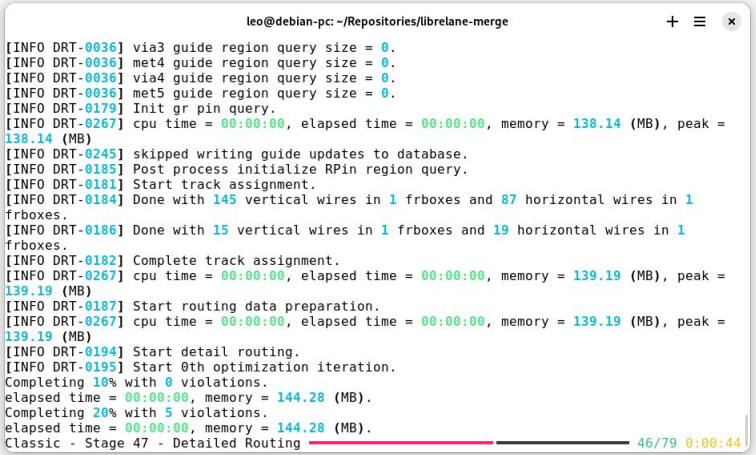



# Physical Implementation using LibreLane

## LibreLane

- Choose one of the user project DEF templates
- Write your design
- Implement the macro
- Make sure DRC and LVS is clean!

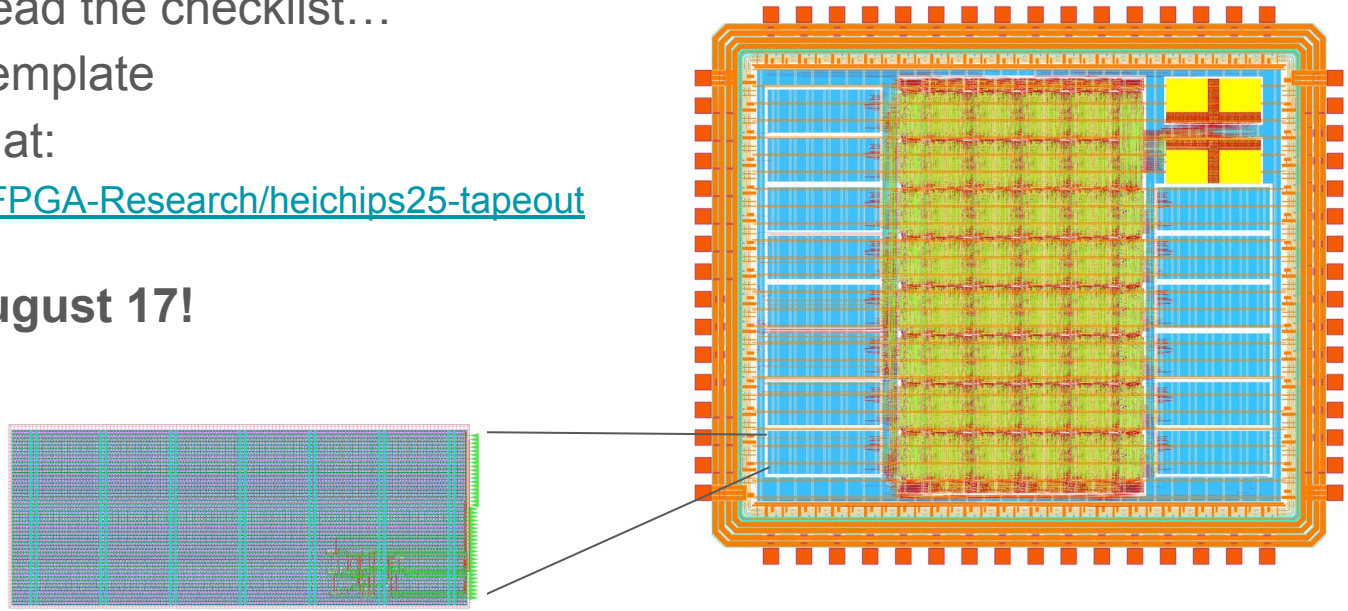



**Ready for submission!**

# Submission of Your User Project

- Make sure to read the checklist…
- Details in the template
- Open an issue at:
- https://github.com/FPGA-Research/heichips25-tapeout

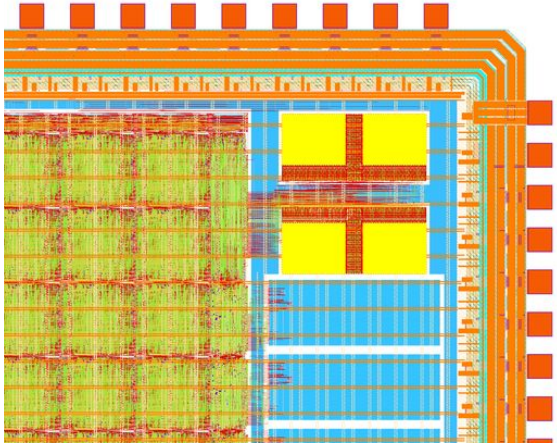
**Deadline is August 17!**



This could be your user project!

# Optimizations for ASICs

- Different than on FPGAs
- Rule of thumb:
  - Combinatorial logic is cheap
  - And flip-flops are expensive
- Dedicated SRAM macros in the PDK
  - Different widths / depths
  - Single-ported, (dual-ported)
  - For large memories
- Better:
  - Optimize your design (small memories, latches instead of FFs)
  - Use the SRAM from the eFPGA (512x32 bit, single-ported)



# Judgment Criteria

- *Your design*
- Efficient use of area
- Maximum clock frequency
- Verification!
- FPGA emulation!
- Combination with other designs
- Mixed signal design (power domain, analog pins?)
- … and more?

# Your Task: Start Hacking!

- In groups of 3 to 6 people
- Gather project ideas
- Choose a user project size→ tell us!
- Work on your design
- Optimize it
- Optimize it
- Optimize it
- Don’t forget to test!