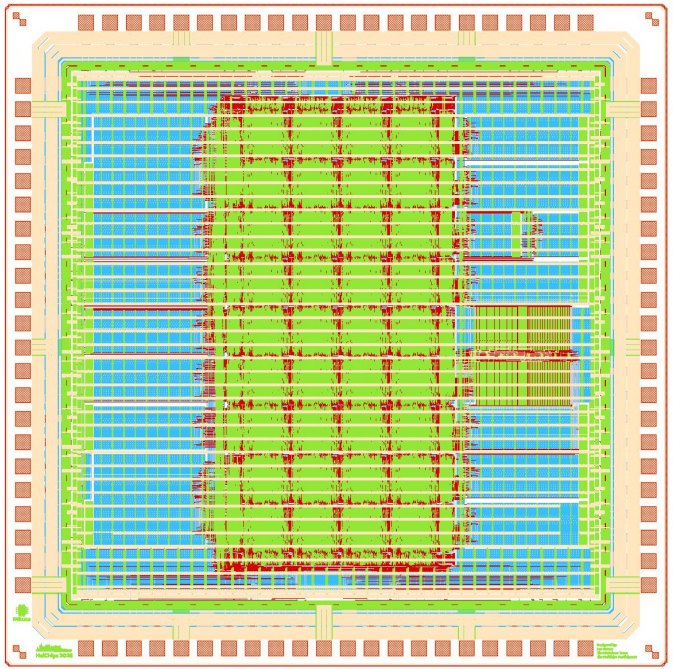
Hackathon Kickoff

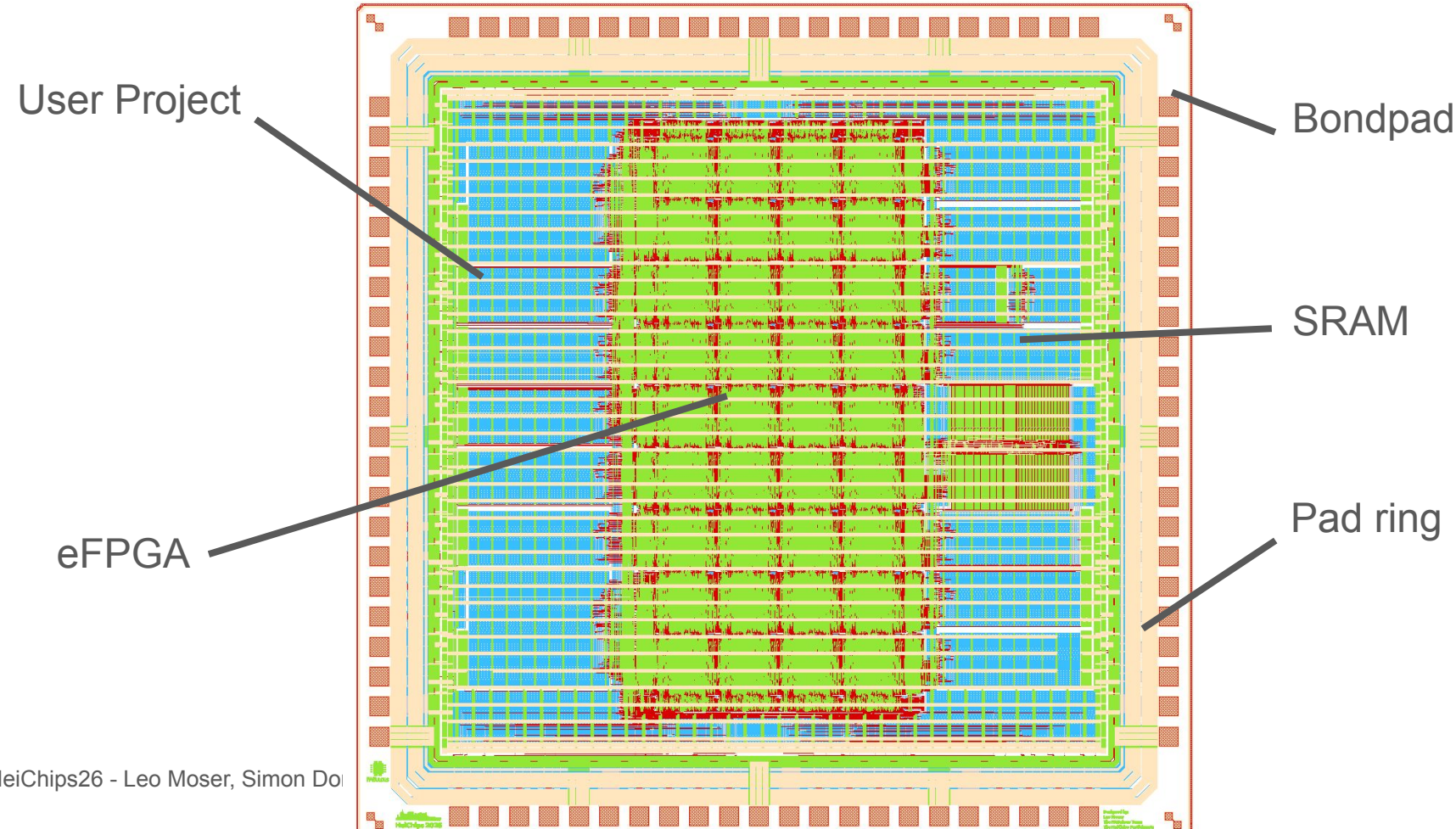
Leo Moser, Simon Dorrer

HeiChips 2026 Tapeout!

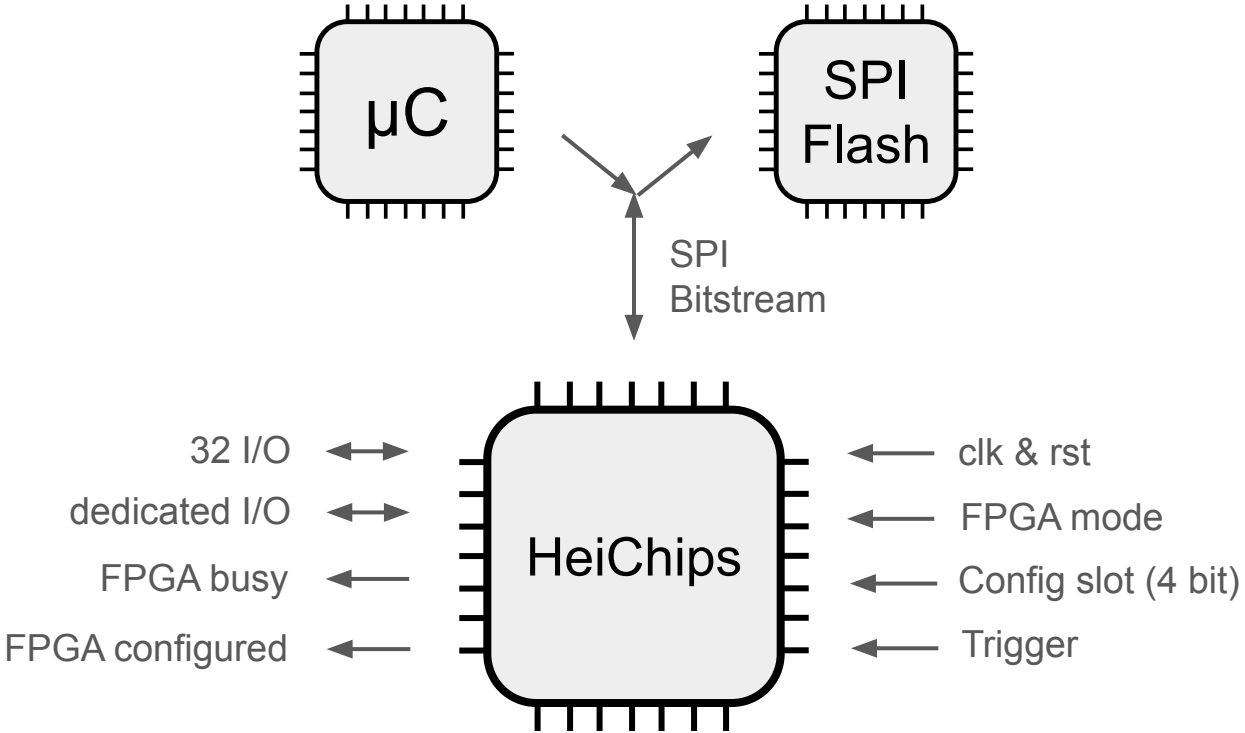
- IHP Open PDK - SG13CMOS5L
- Embedded FPGA in the center
- 32 slots available
- User Projects
  - Tiny: 1 slot
  - Small: 2 slots
  - Large: 4 slots
- 4 KiB SRAM
- Repository:
  - https://github.com/HeiChips/heichips26-tapeout



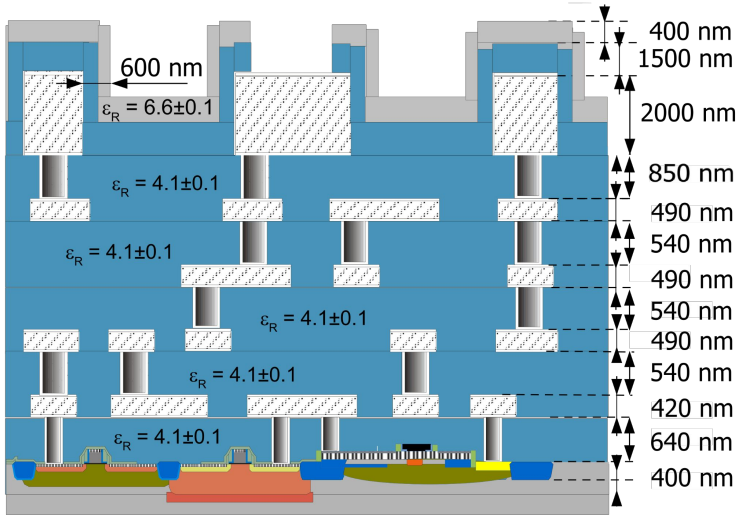
HeiChips 2026 Tapeout


User Project
Bondpad
SRAM
eFPGA
Pad ring

Chip I/Os

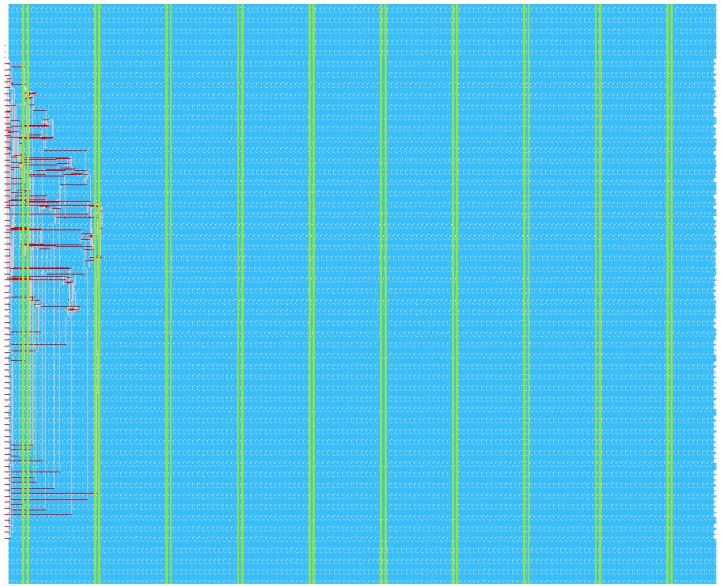

µC
SPI Flash
SPI Bitstream
HeiChips
32 I/O
dedicated I/O
FPGA busy
FPGA configured
clk & rst
FPGA mode
Config slot (4 bit)
Trigger

SG13CMOS5L Process Cross Section

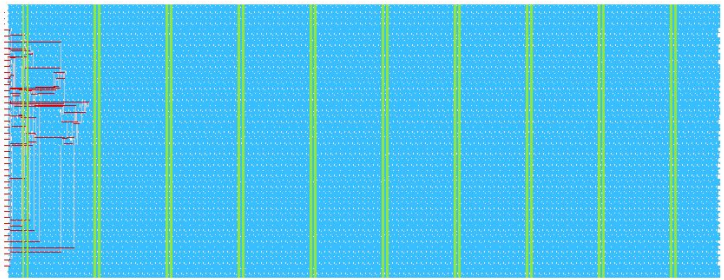


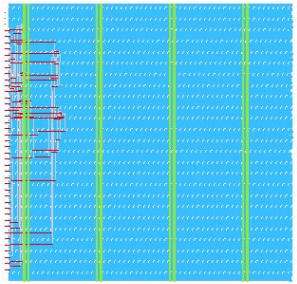
- 5 Metal layers
  - 1 TopMetal
  - 4 Metal
- No deep n-well
- No MIM caps
- No HBT BJTs

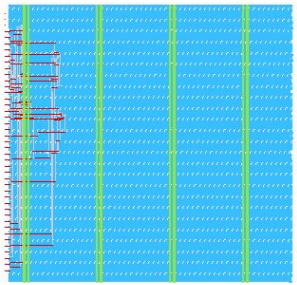
Large



Small







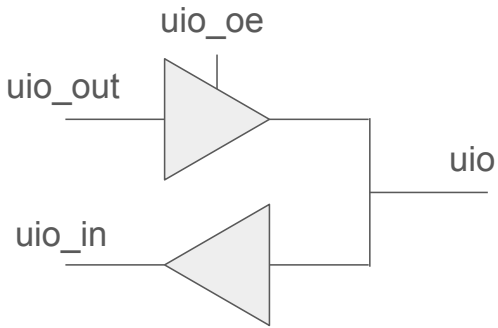
Tiny

Tiny

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 heichips26_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
);
```



Project Templates

- **Tiny** project template: **200** µm × **200** µm
- **Small** project template: **500** µm × **200** µm
- **Large** project template: **500** µm × **415** µm

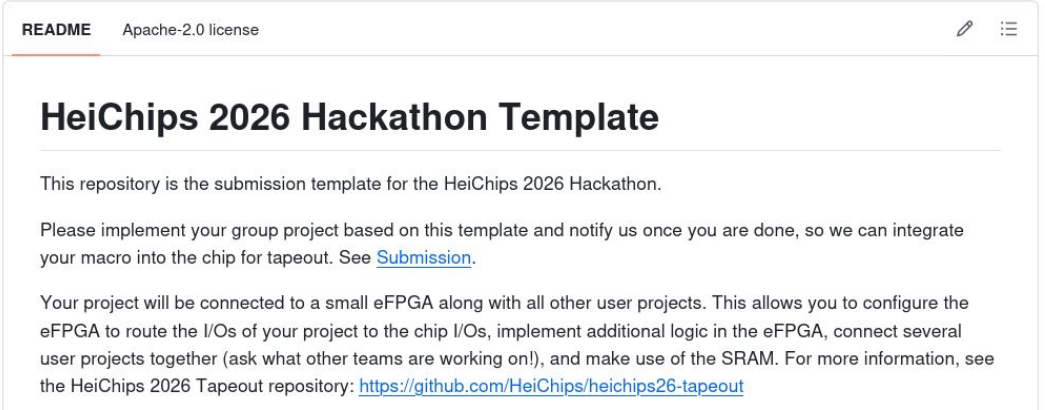
- Chip provides 32 slots
  - Large: 4 slots
  - Small: 2 slots
  - Tiny: 1 slot

- Chip has 18 unused I/Os
  - direct connection to pads
  - up to 3 analog pins per project

HeiChips 2026 Hackathon Template

Everything is Open Source!

- Digital Flow
  - Simulation and Verification
  - Emulation on FPGA
  - Physical Implementation using LibreLane
- Analog Flow
  - Schematic entry
  - Simulation
  - Layout entry
  - DRC and LVS
  - Parasitic extraction


README Apache-2.0 license

HeiChips 2026 Hackathon Template

This repository is the submission template for the HeiChips 2026 Hackathon.

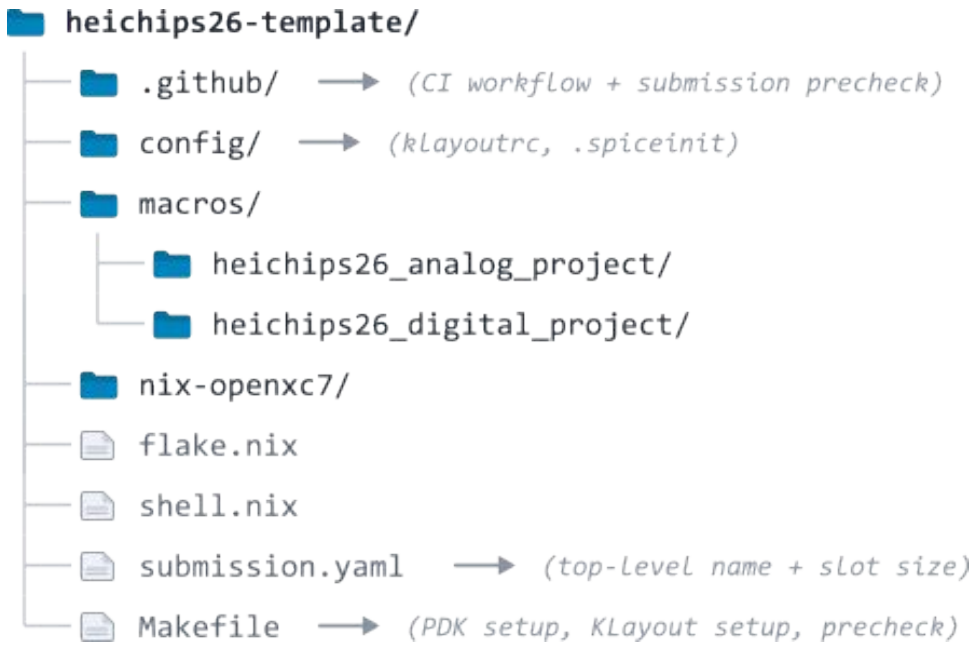
Please implement your group project based on this template and notify us once you are done, so we can integrate your macro into the chip for tapeout. See Submission.

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 2026 Tapeout repository: https://github.com/HeiChips/heichips26-tapeout

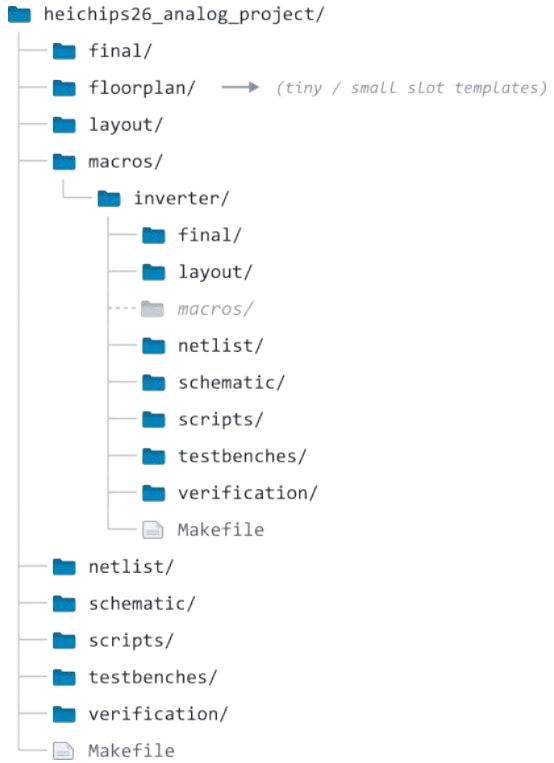
https://github.com/HeiChips/heichips26-template

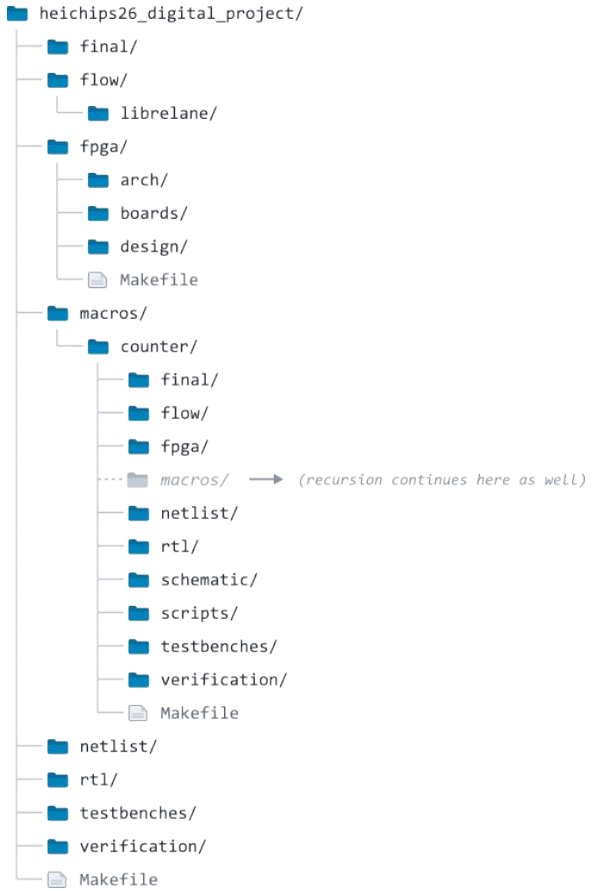
Template Structure

Directory Structure

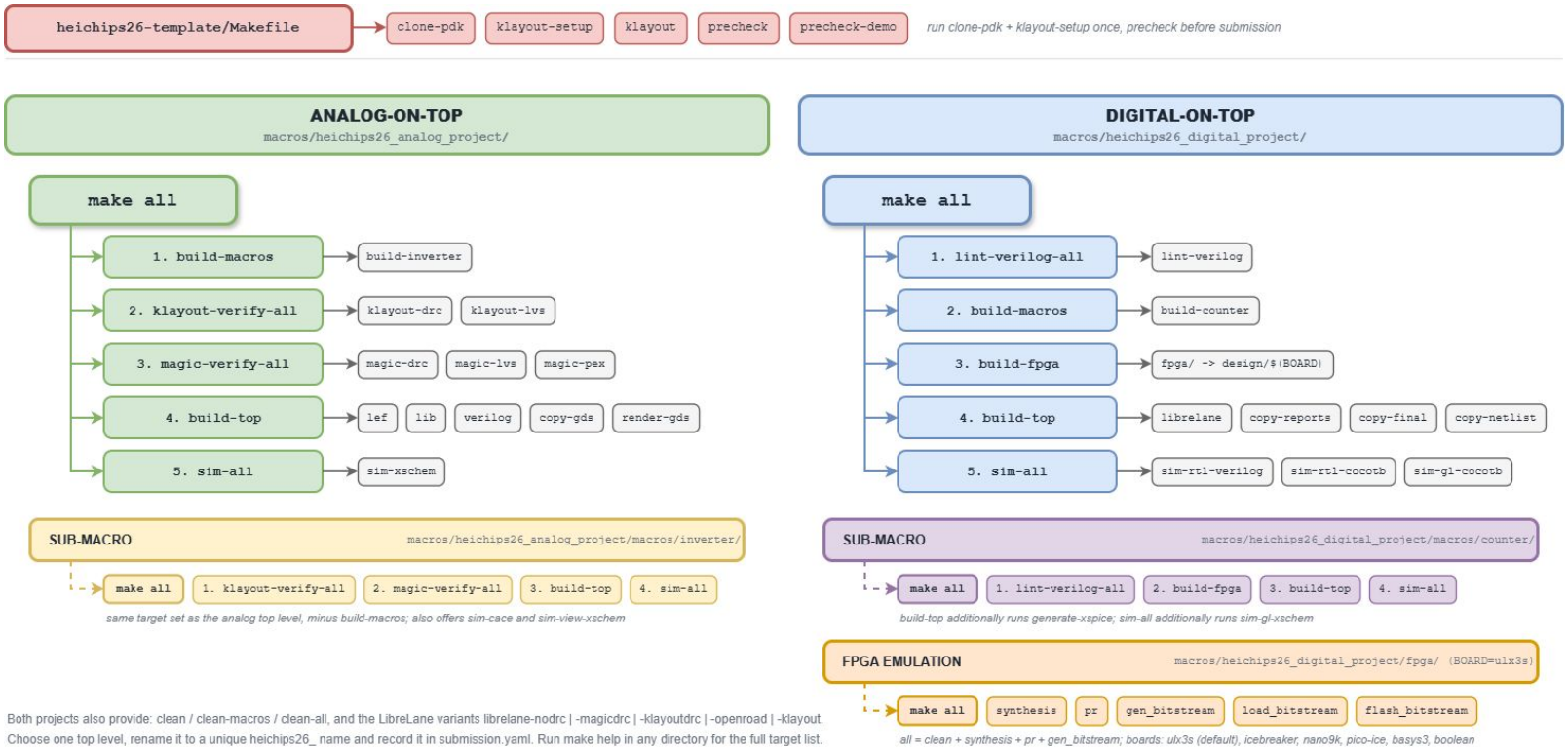


Directory Structure





Makefile Structure



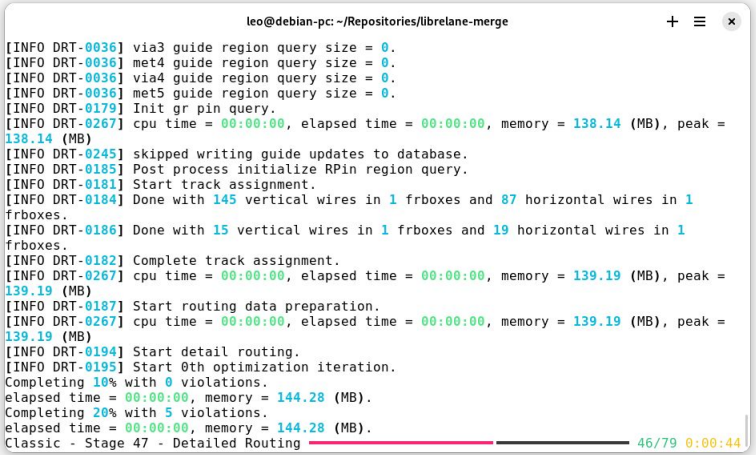
Both projects also provide: clean / clean-macros / clean-all, and the LibreLane variants librelane-nodrc | -magicdrc | -klayoutdrc | -openroad | -klayout.
Choose one top level, rename it to a unique heichips26_ name and record it in submission.yaml. Run make help in any directory for the full target list.

Implementation

Digital: Physical Implementation using LibreLane

LibreLane

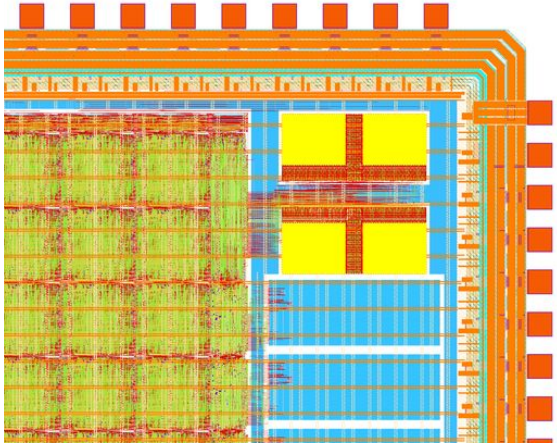
- Choose one of the user project DEF templates
- Write your design
- Implement the macro
- Make sure DRC and LVS is clean!



Ready for submission!

Digital: 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
- Prefered:
  - Optimize your design (small memories, latches instead of FFs)
  - Use the SRAM from the eFPGA (1024x32 bit, single-ported)



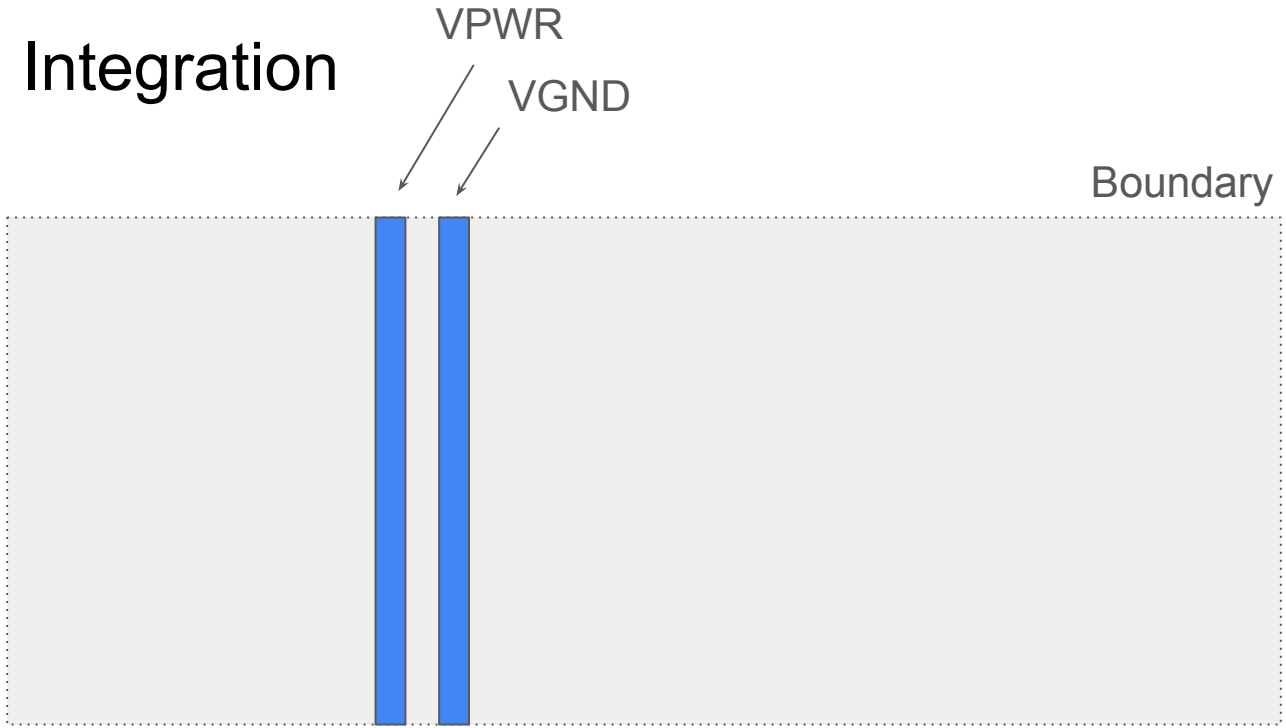
Analog: Physical Implementation

Xschem, Ngspice, KLayout, Magic

- Create the schematic
- Run the simulations
- Draw the layout
  - manually or by scripting
- Run DRC and LVS
- Parasitic Extraction

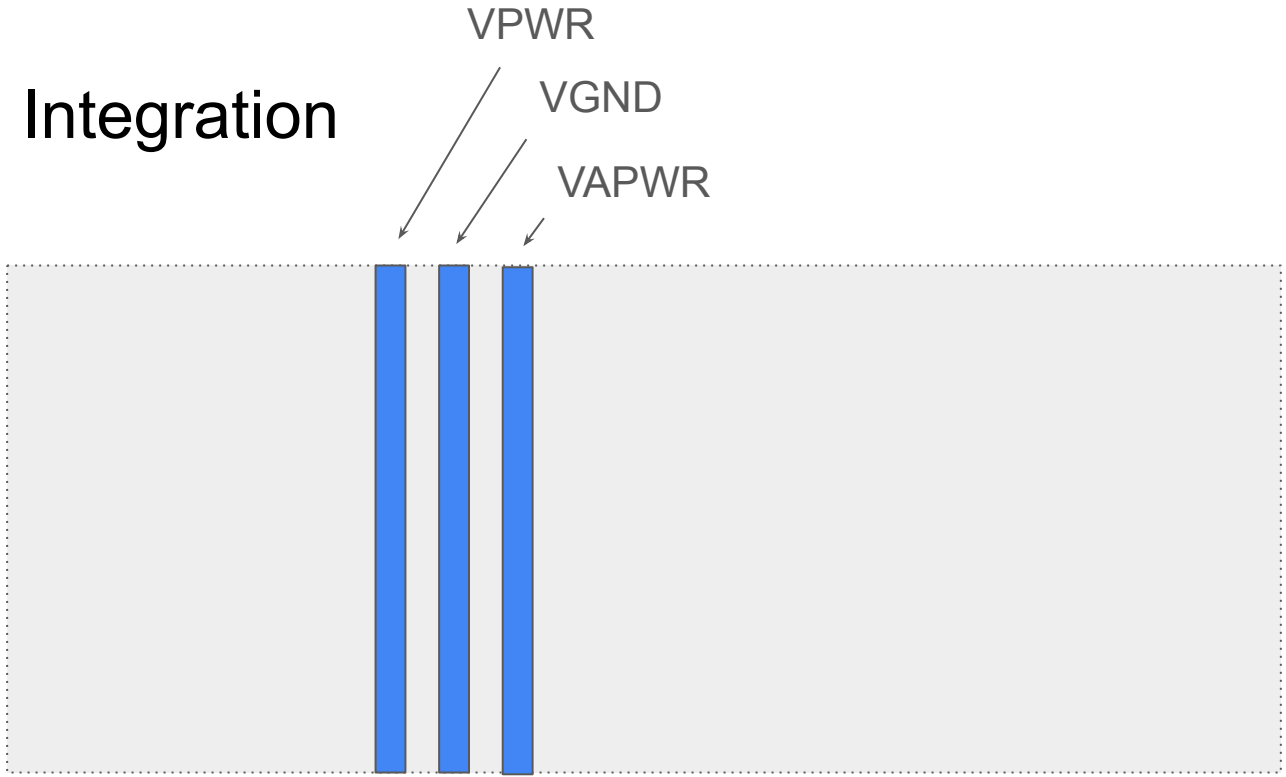
Ready for submission!

Top-level Integration

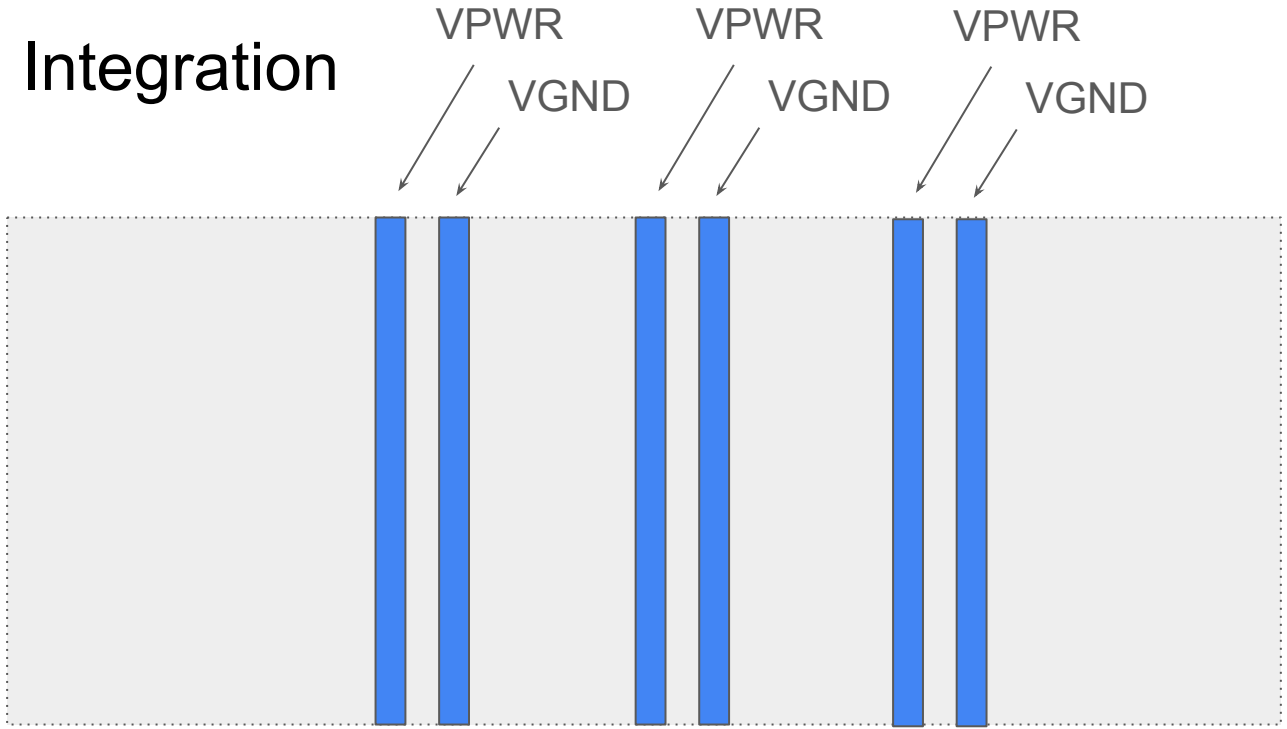


vertical power/ground straps on Metal4

Top-level Integration

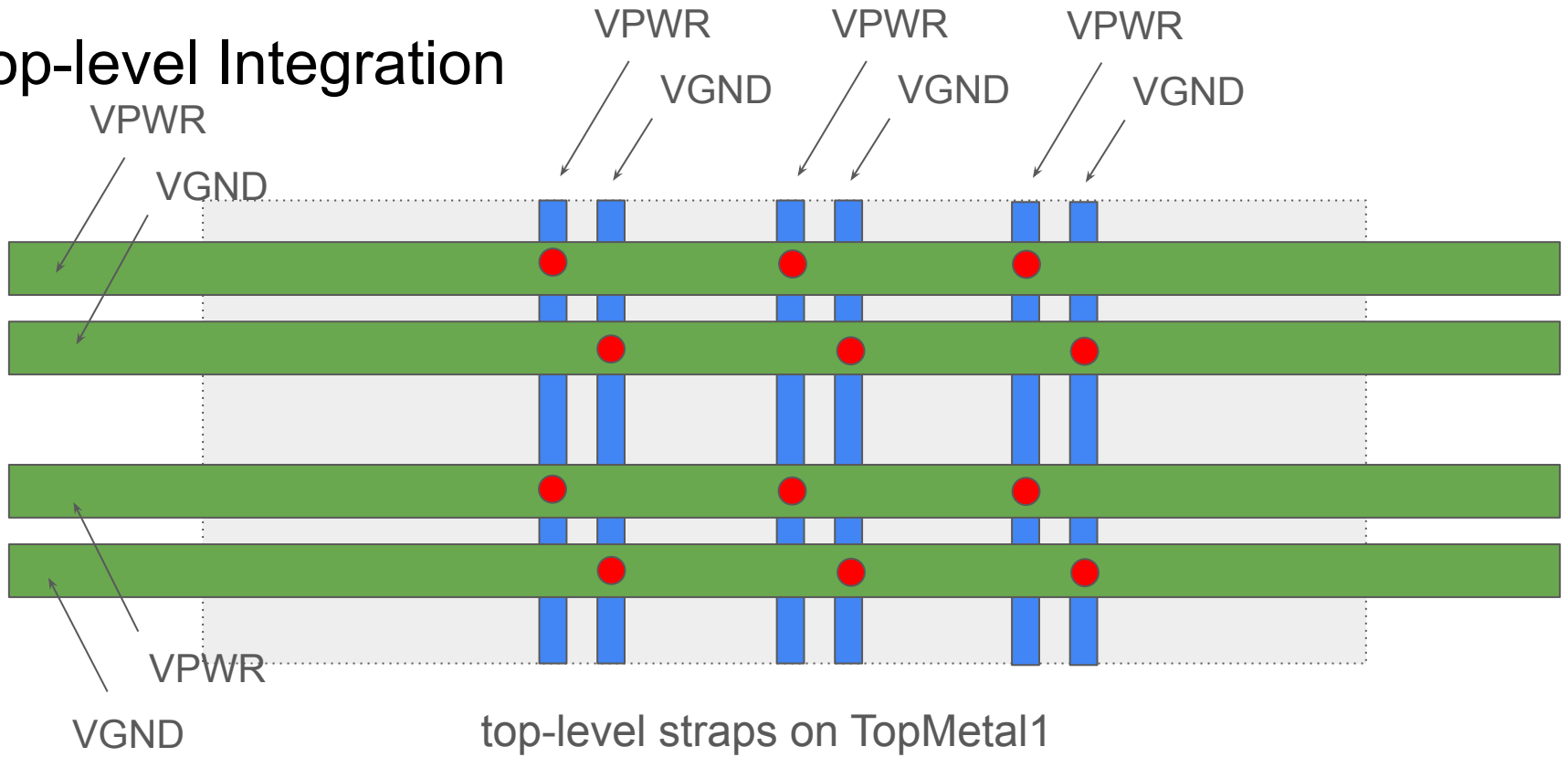


Top-level Integration



use multiple strap pairs

Top-level Integration



Submission Process

Files to Submit

1. **GDS** (.gds)
   - The actual layout (polygons).
   - Must be DRC clean.
2. **LEF** (.lef)
   - Abstract view of your design.
   - Includes information about the pins, required for top-level integration.
3. **Verilog header** (.vh)
   - Simply lists the pins.

Submission Checklist

- The project top-level has a unique name starting with `heichips26_`.
- One of the available project templates is used (tiny, small or large).
- `TopMetal1` in the macro is empty. This is required for the integration.
- The design has been verified in simulation.
- The macro is DRC clean.
- The macro should be LVS clean.
- The macro uses the default power pins: VPWR, VGND, VAPWR (optional).
- The project is licensed under a compatible open-source license, for example Apache 2.0.

Submission of Your User Project

- Make sure to read the submission checklist…
- Details in the template:
  - https://github.com/HeiChips/heichips26-template
- Open an issue for submission at:
  - https://github.com/HeiChips/heichips26-tapeout

This could be your user project!

Judgment Criteria

- Creativity & Usefulness
- Efficient use of area
- Maximum clock frequency
- Level of maturity: Verification!
- FPGA emulation! (digital)
- Combination with other designs
- Mixed signal design (multiple power domains)
- Coolness level
- … and more?

Your Task: Start Working!

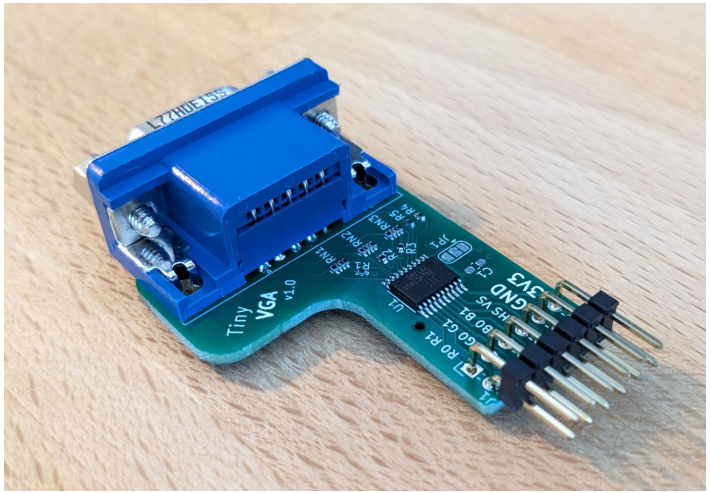
Happy Hacking!

- Choose a place to settle down.
- Split the work into smaller packages and distribute across team members.
- Work on your design.
- Optimize it.
- Optimize it.
- Optimize it.
- Don’t forget to test!
- Make sure to submit your design until the end of the hackathon.
- Start with a simplified design, **pass the precheck**, iterate!
- Prepare a short presentation (just a few slides).

If you have any questions:
please just let us know.

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