Chips4Makers Toolchain

Is an ASIC made with fully open source tool chain possible ?

Is it affordable ?

Staf Verhaegen

Overview

- Chips want to be free
- I have a dream
- Pilot Project: Retro-uC
- ASIC toolchain
- PCB toolchain
- Summary

Chips want to be free

- Linux: hackers (makers) coding at night on their PC

- For custom ASICs high start-up cost but now we have:

Crowdfunding

KICKSTARTER

CROWD SUPPLY

Distributed Development

GitHub

GitLab

I have a dream

- Low-volume Open Source ASIC service
  - ASIC production targets high-volume: high startup costs, low per unit costs
    - Mask costs
    - EDA tools license
    - Engineering
    - IP blocks
    - The fine print is not maker friendly

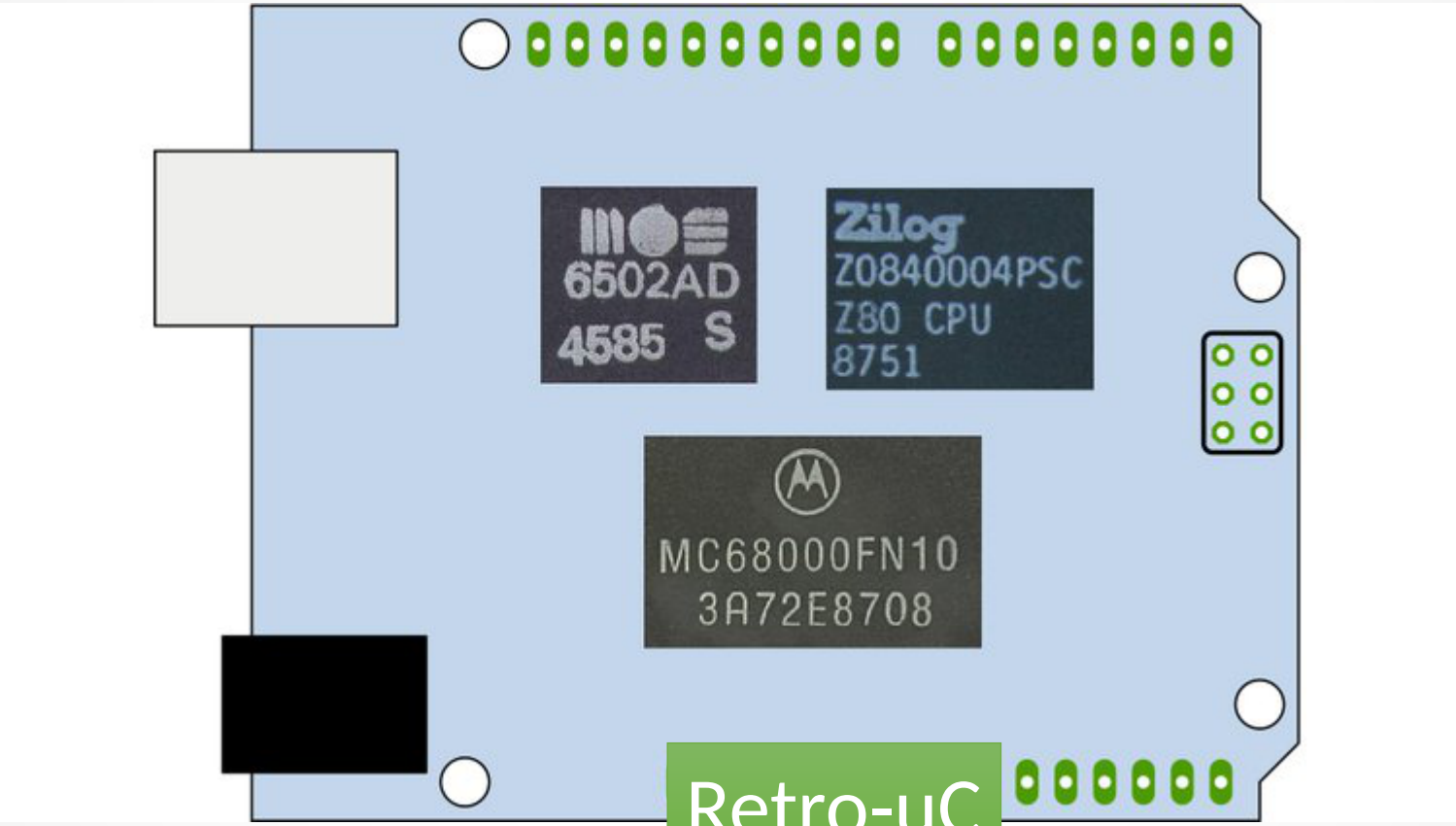
I have a dream

- Low-volume Open Source ASIC service
  - Cost < €100/piece for 100+ devices/boards; lower prices for higher volume
    - Open source RTL
      not reinvent/repay for the wheel
    - Pool different chips to share set-up cost;
      use multi-project wafer services
    - Push button open source EDA flow;
      not only cost but also flow innovation
    - Intermediate in the legal affairs

Pilot Project: Retro-uC

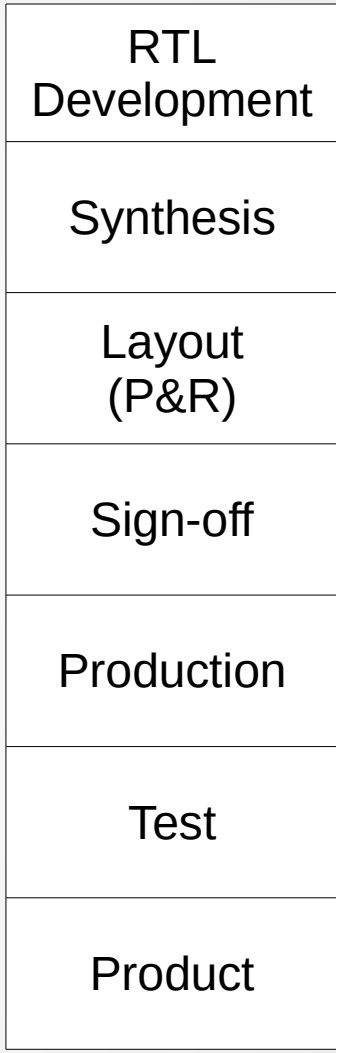
- Why ?
  - Find the shape of puzzle pieces
    - Start set of open source reusable RTL cores
    - Open source EDA flow
    - Investigate startup cost reduction and sharing potential
    - Dismantling legal minefield
      (took more time then projected and continuous effort)

Pilot Project: Retro-uC



micro-controller with Z80, MOS6502, Motorola 68000 cores
Tomorrow Talk @ Retrocomputing Devroom: Retro-uC

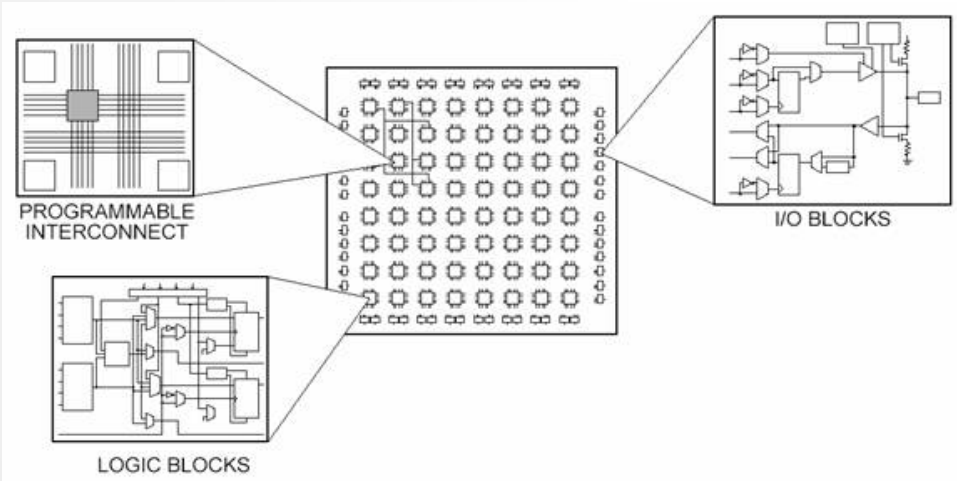
ASIC toolchain



ASIC toolchain

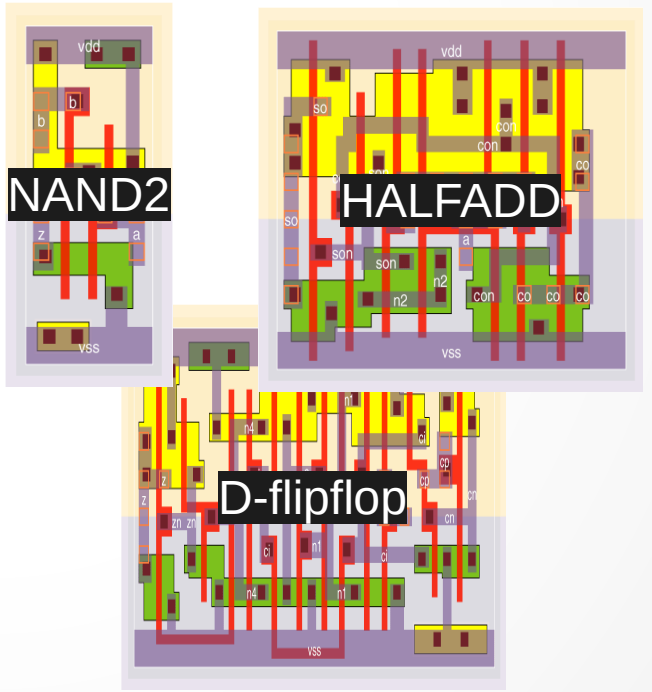
- FPGA vs ASIC

FPGA:
fixed programmable blocks



Courtesy National Instruments

Standard cells:
small blocks with logical function



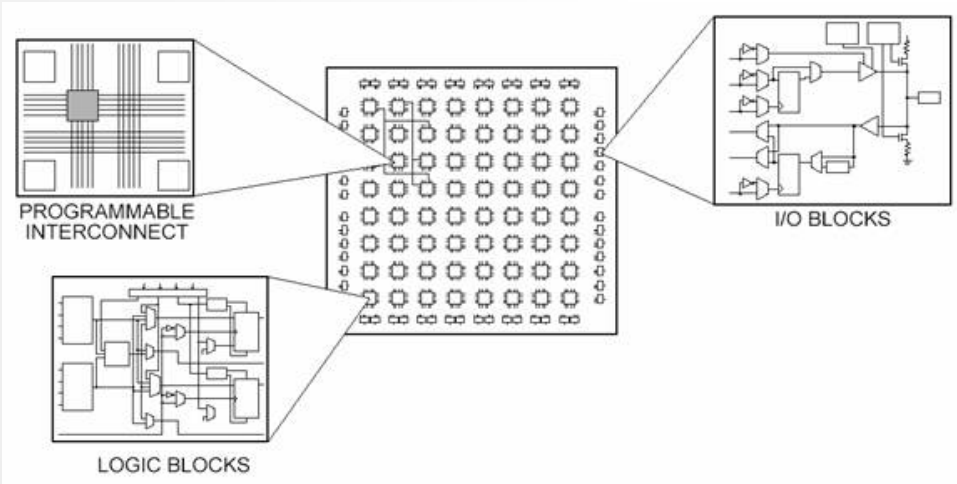
Courtesy vlsitechnology.org

RTL Development
Synthesis
Layout (P&R)
Sign-off
Production
Test
Product

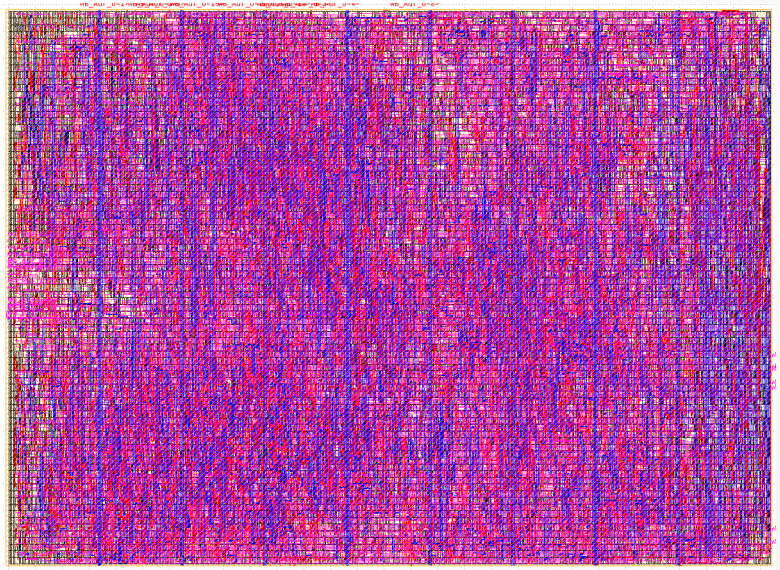
ASIC toolchain

- FPGA vs ASIC

FPGA:
fixed programmable blocks



Standard cells:
small blocks with logical function
Combined on chip



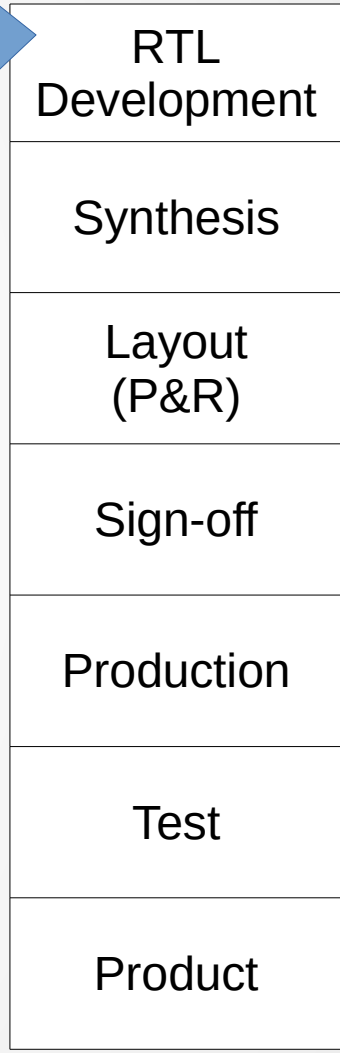
Courtesy National Instruments

RTL Development
Synthesis
Layout (P&R)
Sign-off
Production
Test
Product

ASIC toolchain

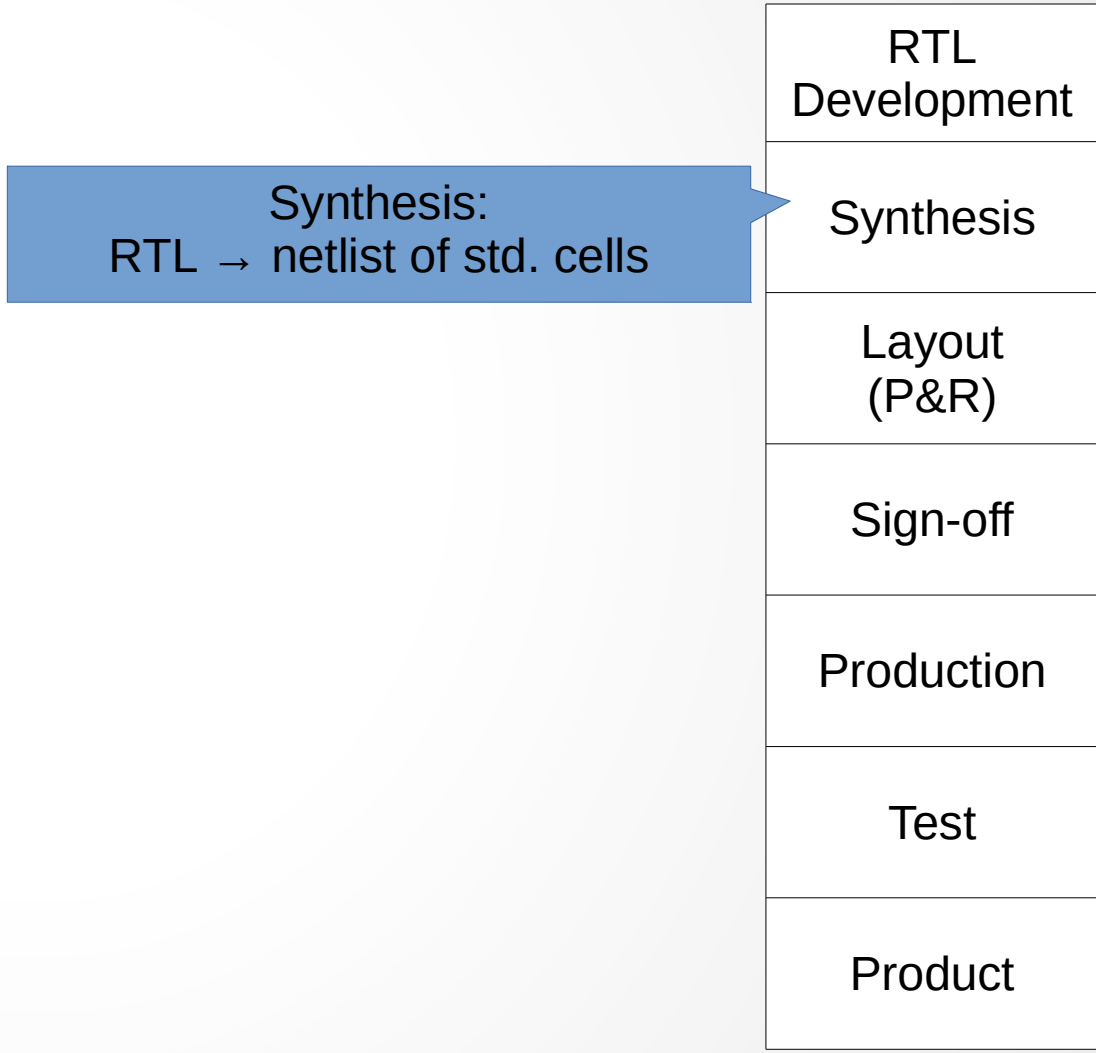
- Tasks

RTL (register transfer language): event driven description of circuit
- Entry
- Simulation/FPGA
- (Formal) Verification



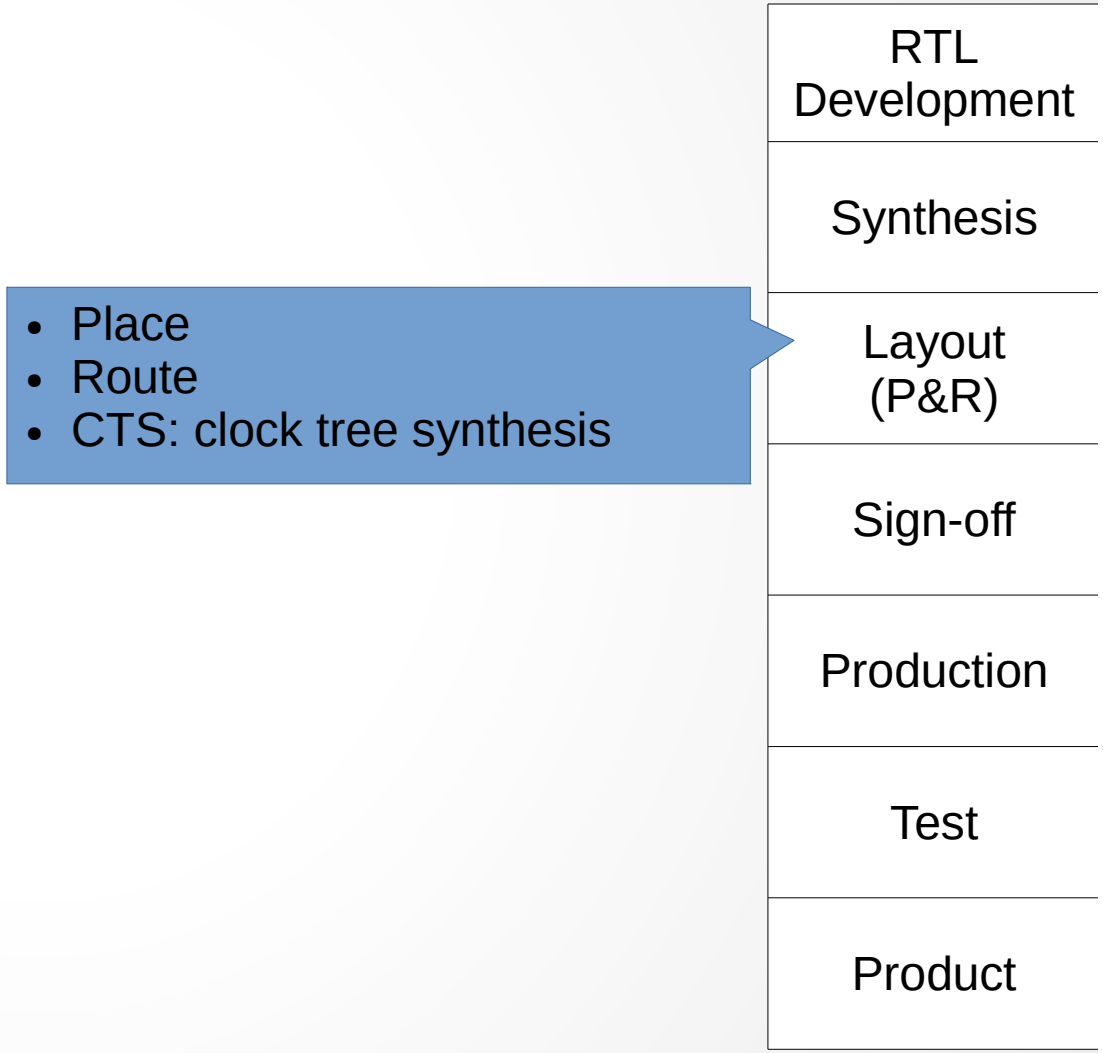
ASIC toolchain

- Tasks



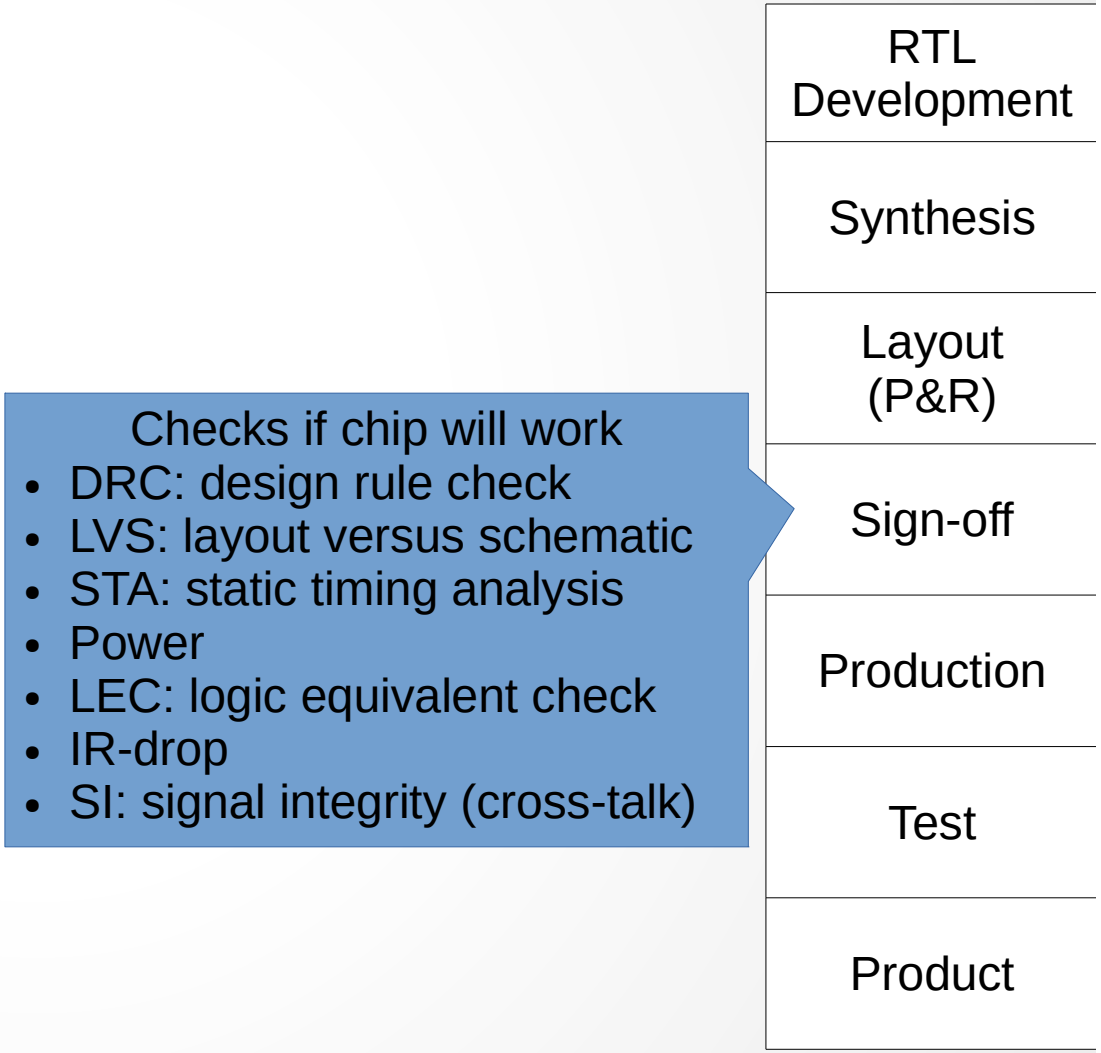
ASIC toolchain

- Tasks



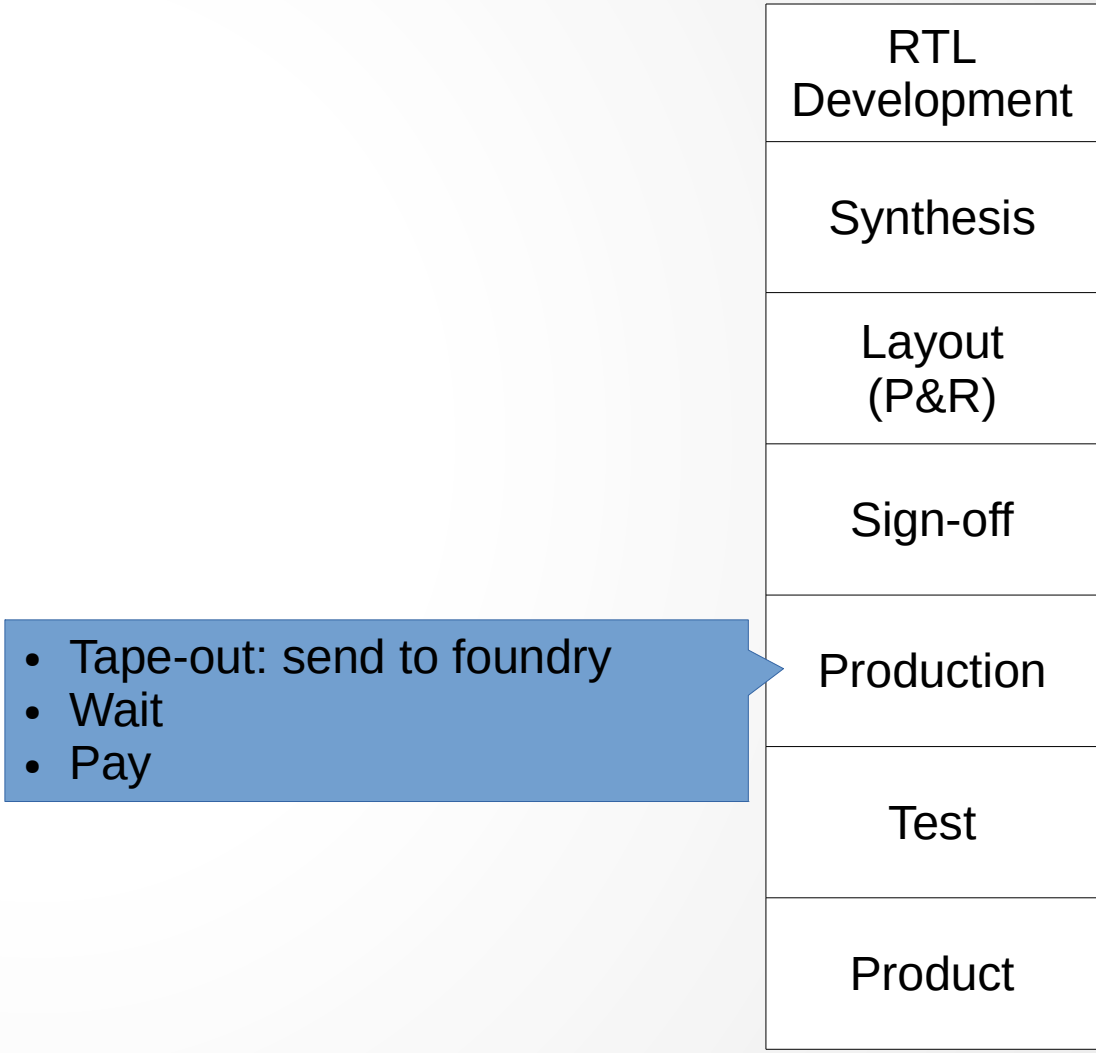
ASIC toolchain

- Tasks



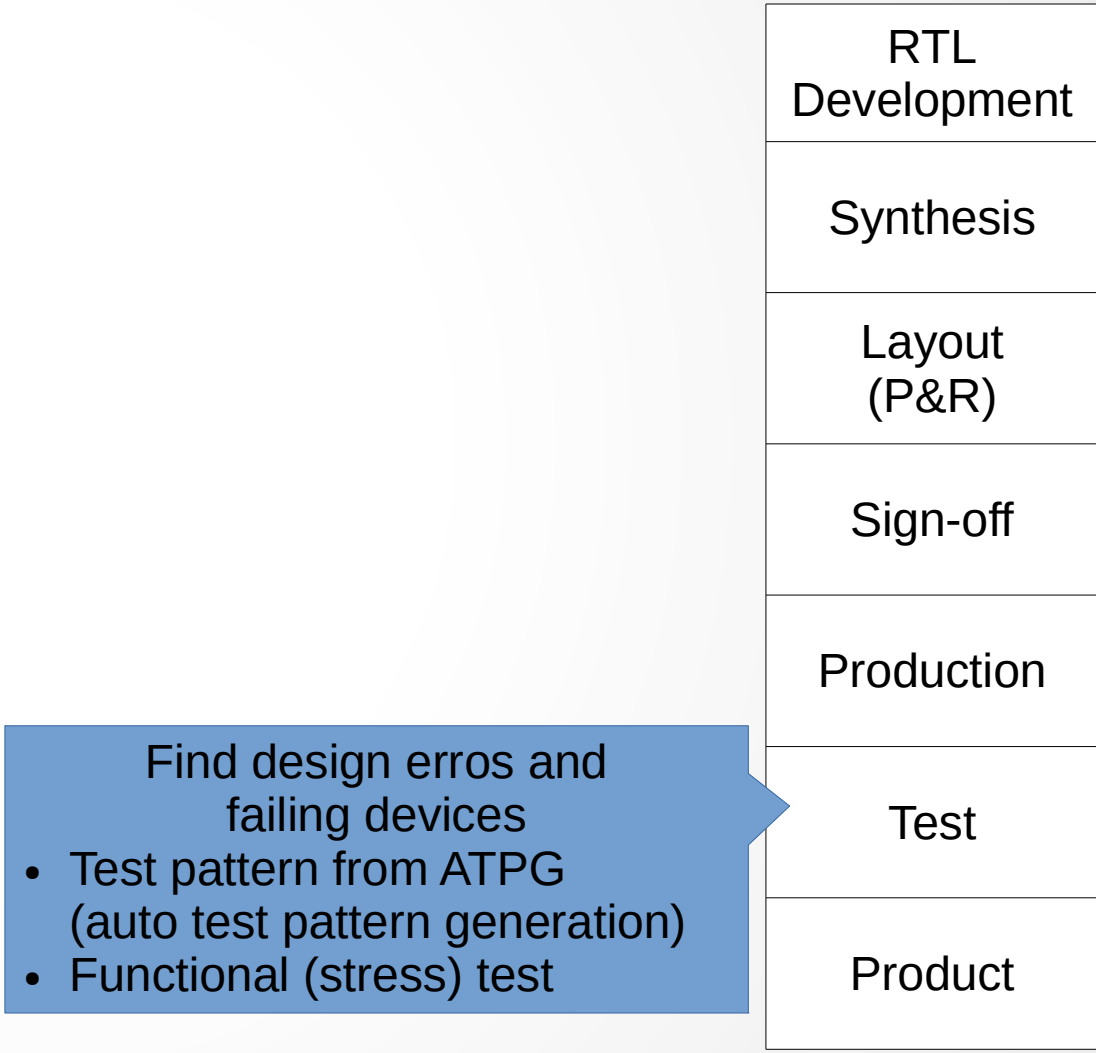
ASIC toolchain

- Tasks



ASIC toolchain

- Tasks

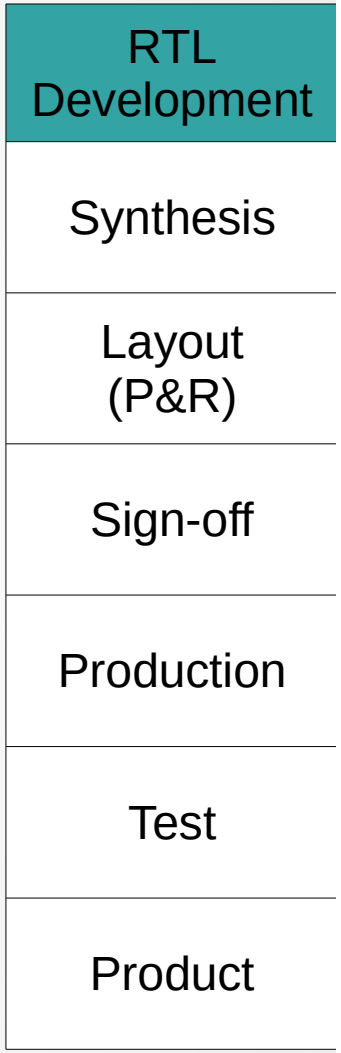


ASIC toolchain

- IP
  - T80: VHDL
  - T65: VHDL
  - ao68000: Verilog

=> mixed language support needed

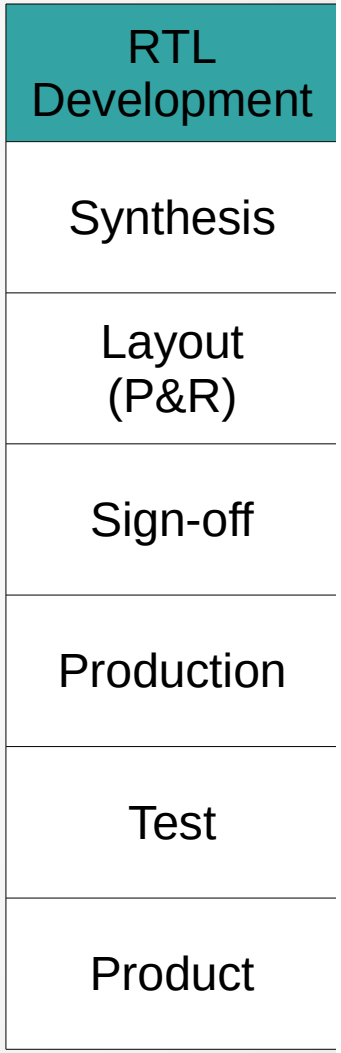
I think this is a general requirement if we want a vibrant open source IP portfolio



ASIC toolchain

- Tools
  - Editing
    - Emacs
    - *Quartus (propr.)*
    - *ISE (propr.)*
  - Simulation
    - cocotb+iverilog: Verilog
    - cocotb+GHDL: VHDL
    - *Modelsim (propr.): mixed language*

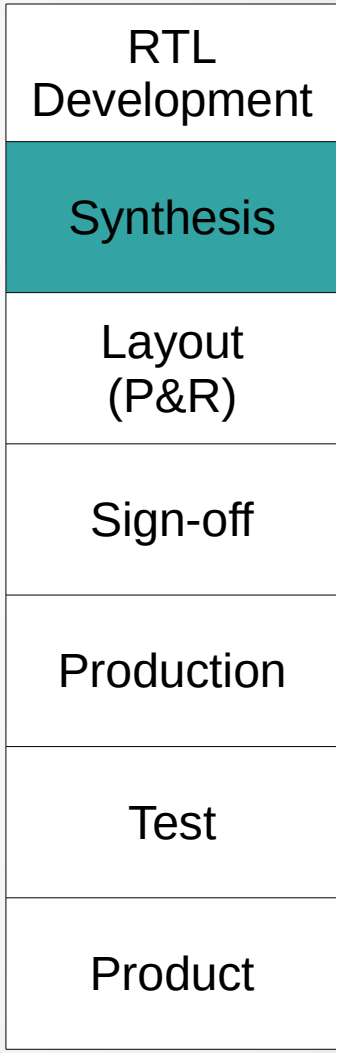
  Looking for open source mixed language simulator vhdlpp for iverilog not complete, other ?



ASIC toolchain

- Tools
  - Yosys (in qflow) for Verilog
  - *Verific parser (propr.)* in Yosys for VHDL

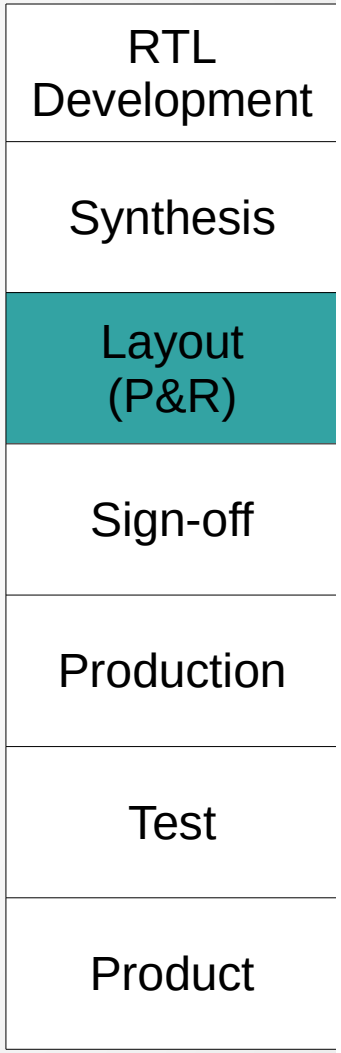
    PS: I'm not aware of modern open source VHDL synthesis tools.
    Alliance nor vhdl2vl could parse the cores I used.
    ghdlsynth-beta is mainly proof-of-concept.



ASIC toolchain

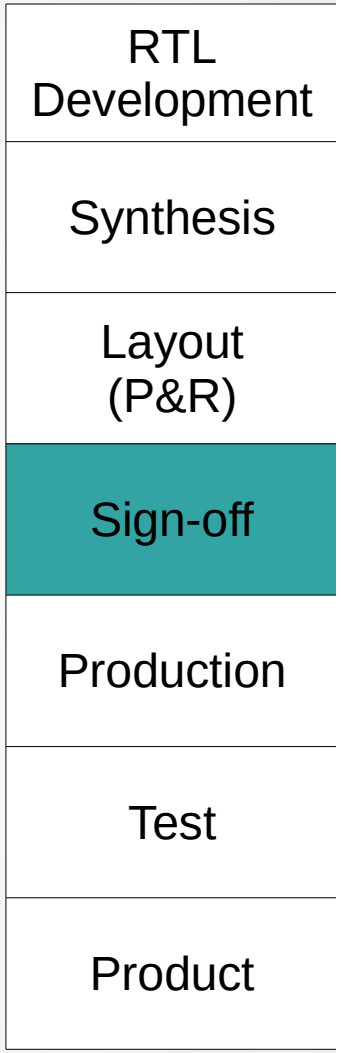
- Tools
  - Qflow
    - Place: graywolf
    - Route: qrouter
  - Coriolis
    - Place: Etasian
    - Global router: Knik
    - Detail router: Kite
- Status
  - Qflow: first tests show room for improvement
  - Coriolis: support for standard file formats under development

  In contact with both authors (Tim Edwards and Jean-Paul Chaput) and I am convinced these problems will be solved.



ASIC toolchain

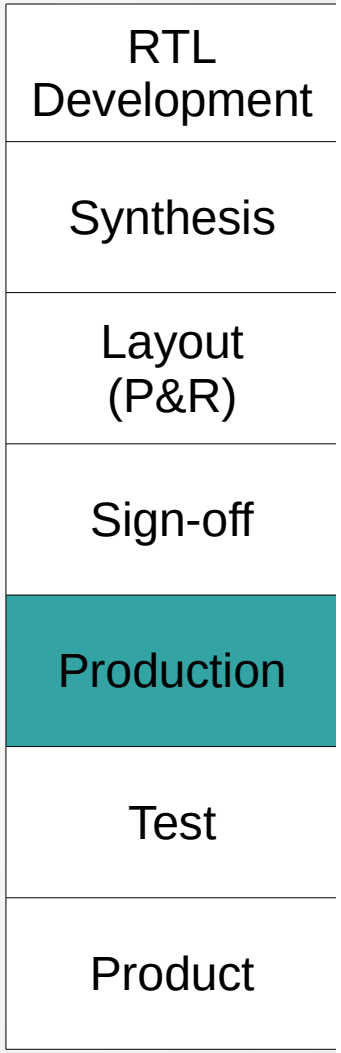
- Did not do thorough search but feeling is that existance of open source tools is limited
  - Magic: DRC/LVS support but lambda rules
  - Electric: same if I remember correctly



ASIC toolchain

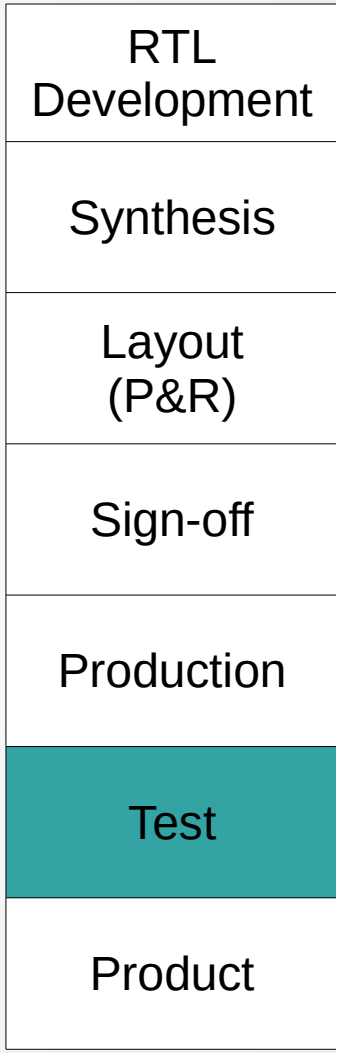
- Mature technology node
  - TSMC/OnSemi/…, 0.35um or 0.25um technology
  - silicon cost starting from below $10000
  - 5V and for future high voltage,
    should still be perfect for analog designs.
- Other big costs
  - Subdicing of MPW dies
  - Packaging setup costs

    => references welcome



ASIC toolchain

- Implemented JTAG interface with boundary scan
- I’m not aware of open source ATPG software (but did not look hard yet)



PCB toolchain

- Design: KiCad

  Has lot's of features but sometimes it still feels I'm fighting with the tool

- Production: eurocircuits.com
  - PCB production
  - Assembly under beta
- Test
  - Eating own food: plan is to use FPGA board with Retro-uC core for test.

Summary

Is an ASIC made with fully open source tool chain possible ?

Yes but renewed effort in mixed language support needed.

For smaller nodes new tools needed for sign-off and test needed.

Is it affordable ?

Should be affordable for crowdfunded low-volume projects.

Would like to reduce setup costs for packaging more.