



# Semiconductors

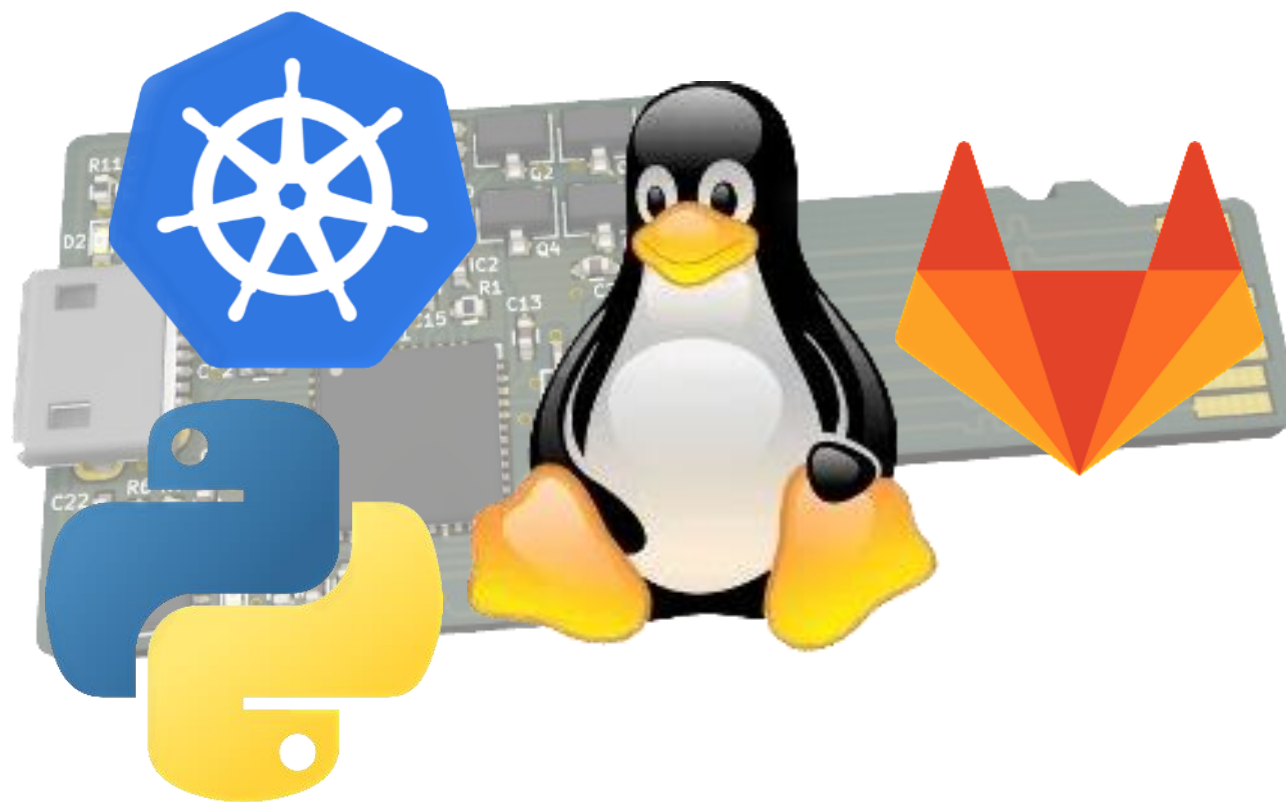
## The Final Frontier Of Open Source

by Daniel Bovensiepen

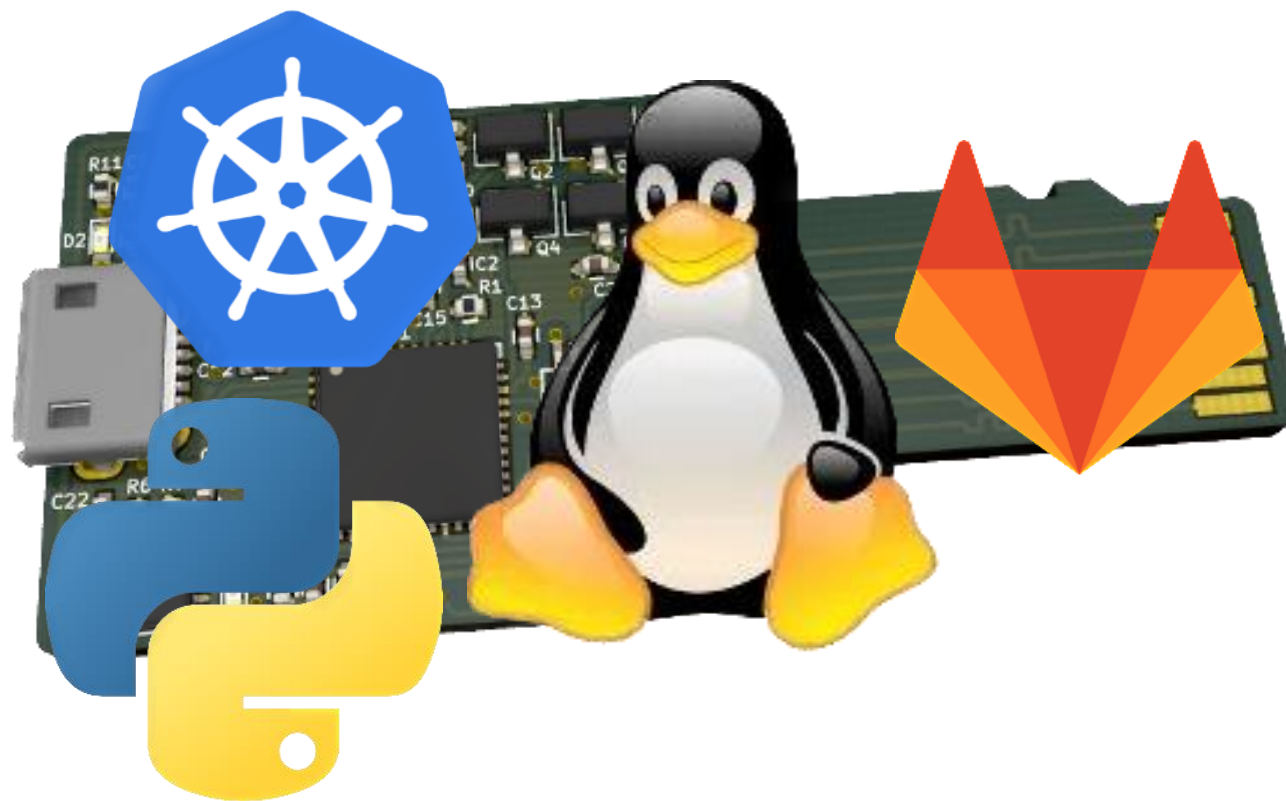
Open Source @ Siemens on 2023-05-24

# | prolog

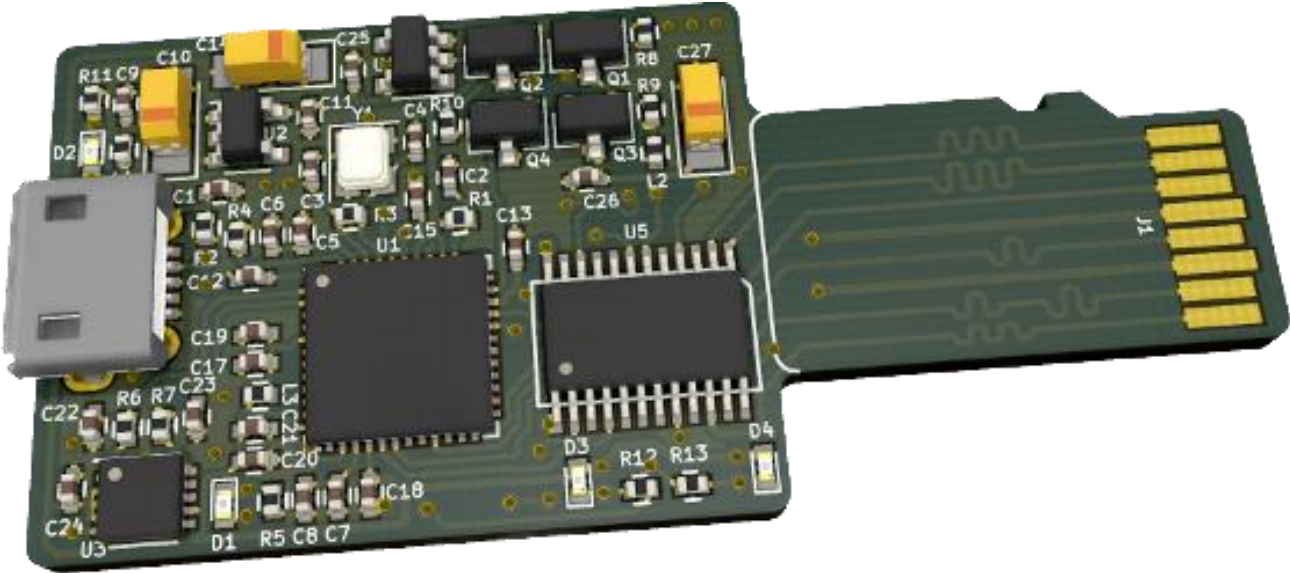
# Open Source Software



# Open Source Hardware



# Open Source Semiconductors





# Semiconductors

## The Final Frontier Of Open Source

by Daniel Bovensiepen

Open Source @ Siemens on 2023-05-24

## Who am I and why do I talk about this?

Hi 🙋

I'm Daniel Bovensiepen

👤 📁 research on industrial communication

👤 💻 open source stuff @ [github.com/bovi](https://github.com/bovi)

📍 living in 北京



## “thought-provoking” questions

- 1) Do you have basic knowledge how an ASIC is made?**
- 2) Do you think an Open Source ASIC is possible today?**
- 3) Do you dare to design your own ASIC?**



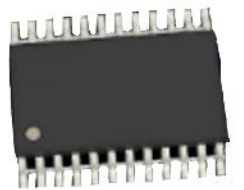
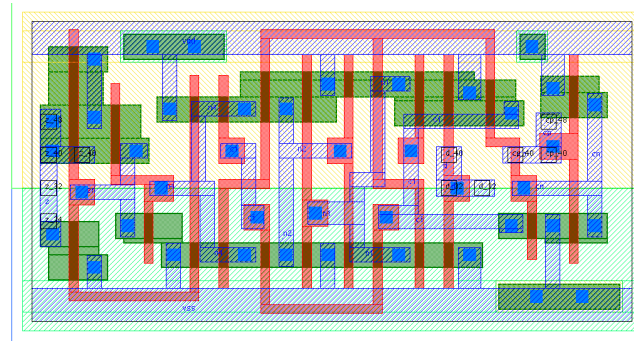
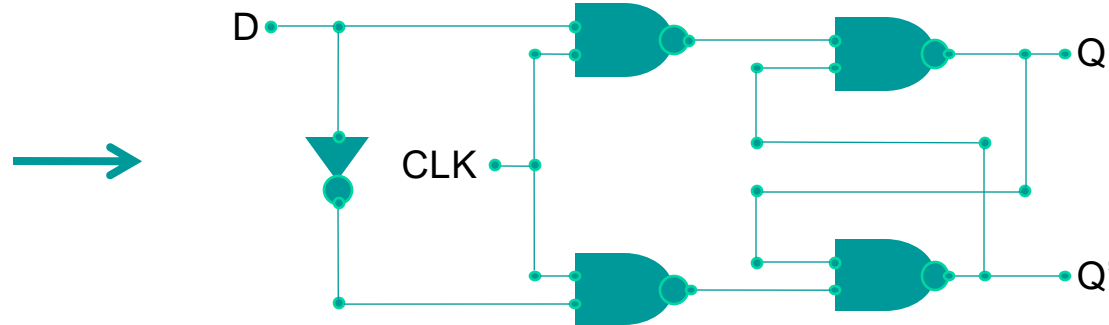
# | ASIC design flow

# ASIC design flow

```
module nand_gate(c,a,b);  
  input a,b;  
  output c;  
  assign c = ~(a&b);  
endmodule
```

```
module not_gate(f,e);  
  input e;  
  output f;  
  assign f = ~e;  
endmodule
```

```
module d_ff_struct(q,qbar,d,clk);  
  input d,clk;  
  output q, qbar;  
  not_gate not1(dbar,d);  
  nand_gate nand1(x,clk,d);  
  nand_gate nand2(y,clk,dbar);  
  nand_gate nand3(q,qbar,y);  
  nand_gate nand4(qbar,q,x);  
endmodule
```



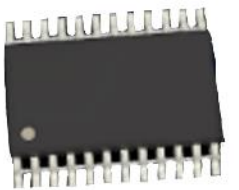
ASIC

## ASIC design flow

```
module nand_gate(c,a,b);  
  input a,b;  
  output c;  
  assign c = ~(a&b);  
endmodule
```

```
module not_gate(f,e);  
  input e;  
  output f;  
  assign f = ~e;  
endmodule
```

```
module d_ff_struct(q,qbar,d,clk);  
  input d,clk;  
  output q, qbar;  
  not_gate not1(dbar,d);  
  nand_gate nand1(x,clk,d);  
  nand_gate nand2(y,clk,dbar);  
  nand_gate nand3(q,qbar,y);  
  nand_gate nand4(qbar,q,x);  
endmodule
```



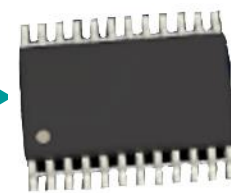
ASIC

# ASIC design flow

Design

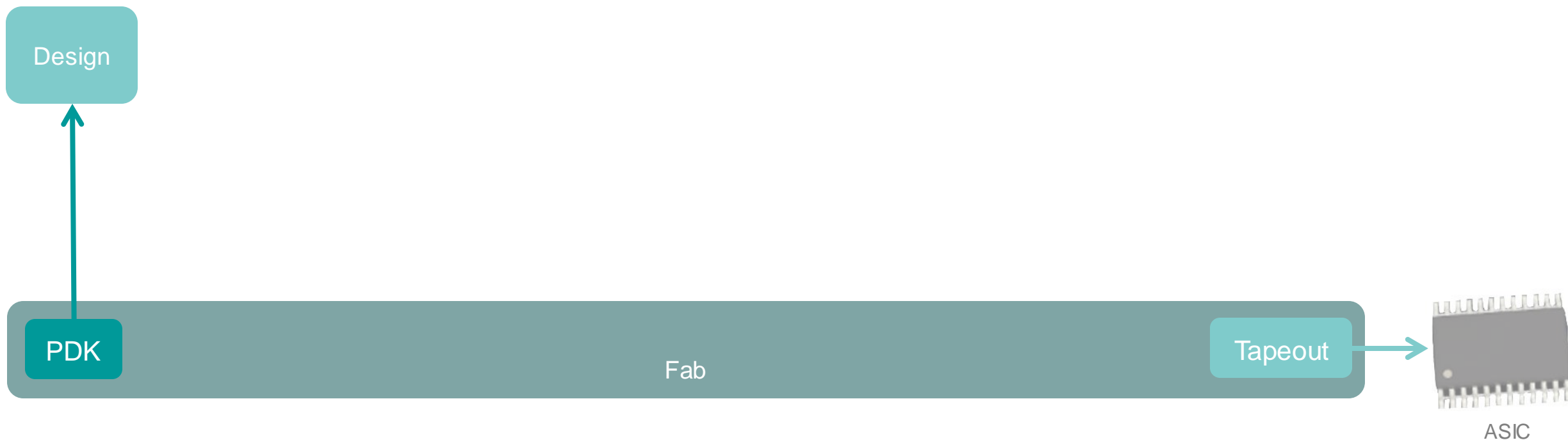
Fab

Tapeout

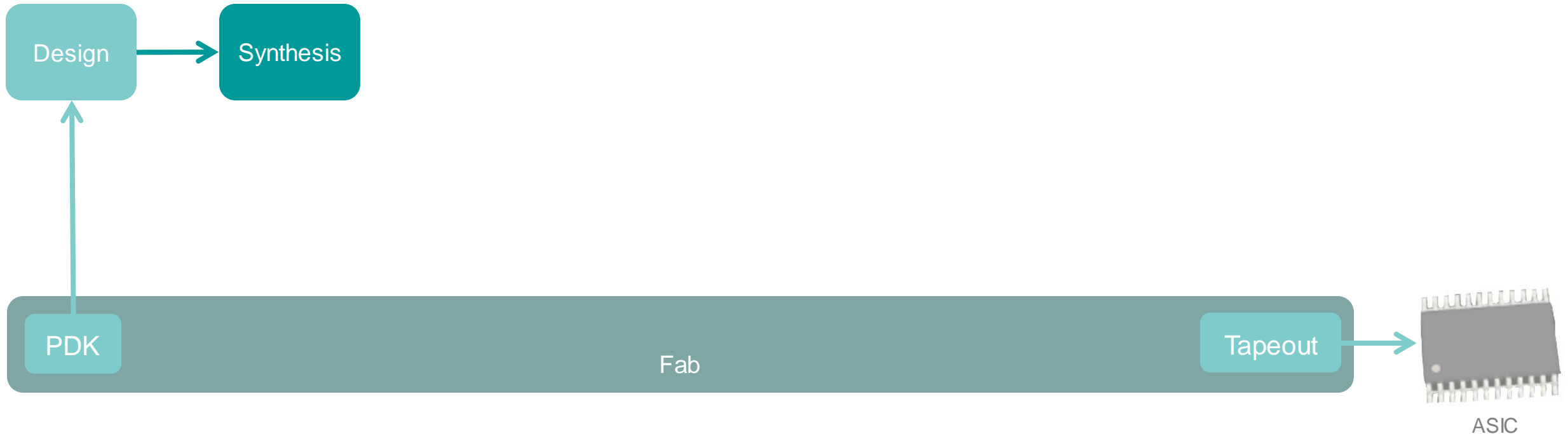


ASIC

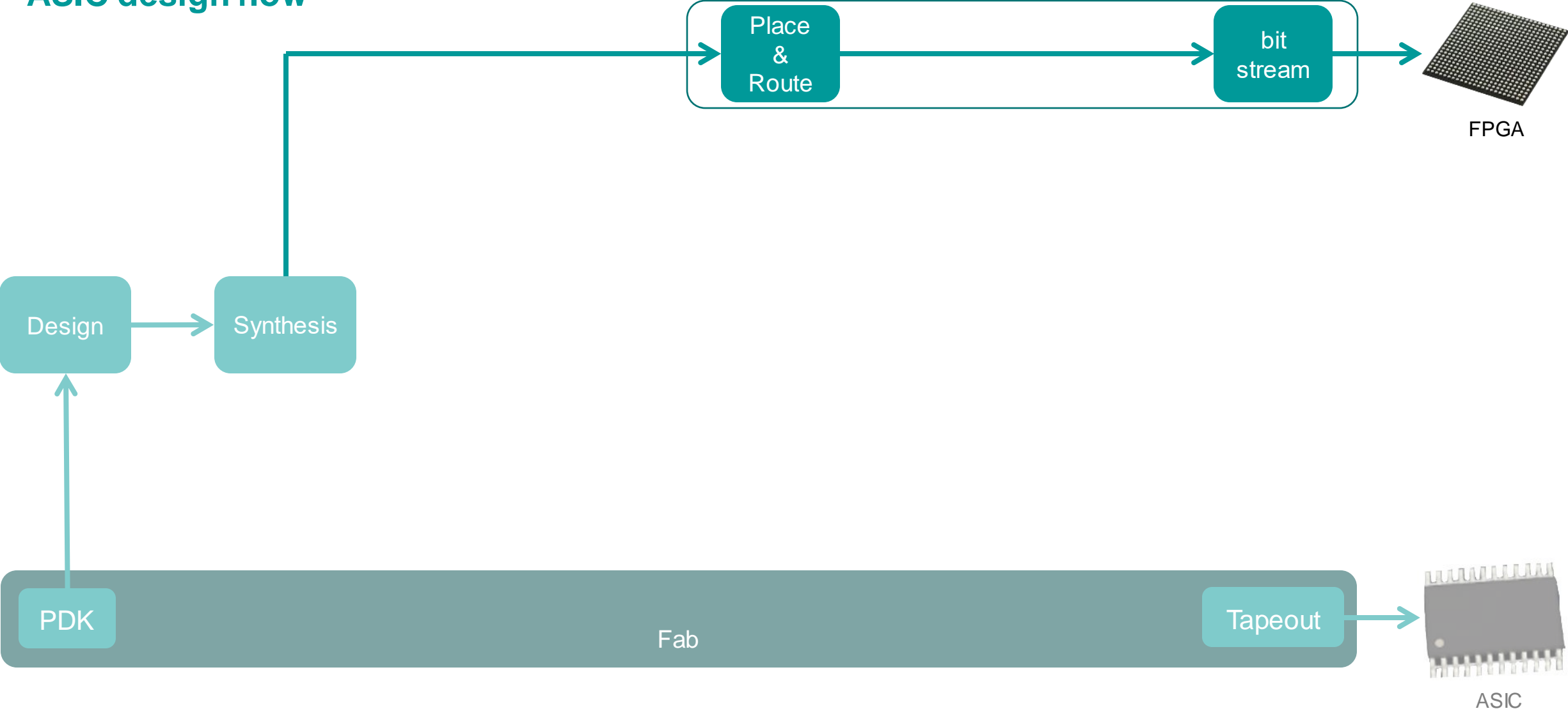
# ASIC design flow



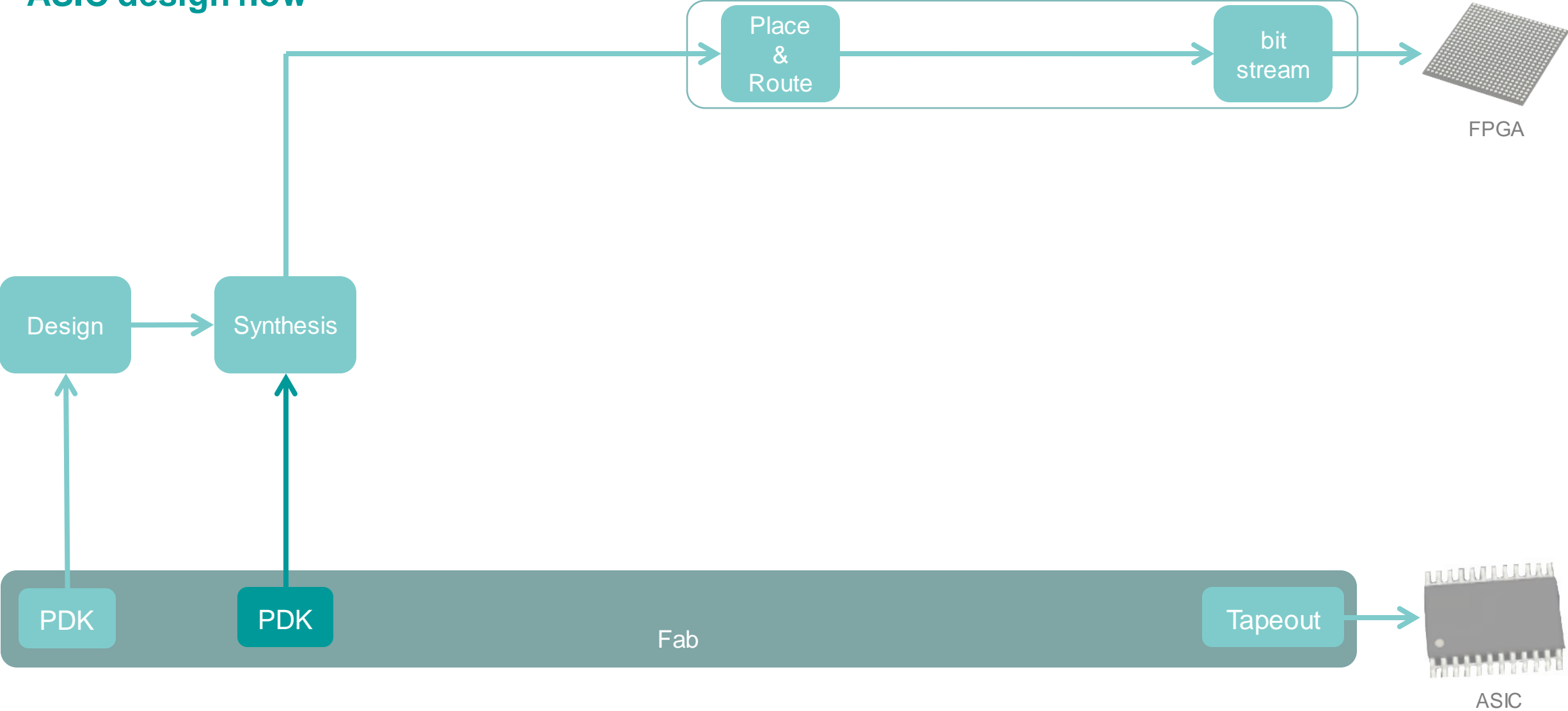
# ASIC design flow



# ASIC design flow

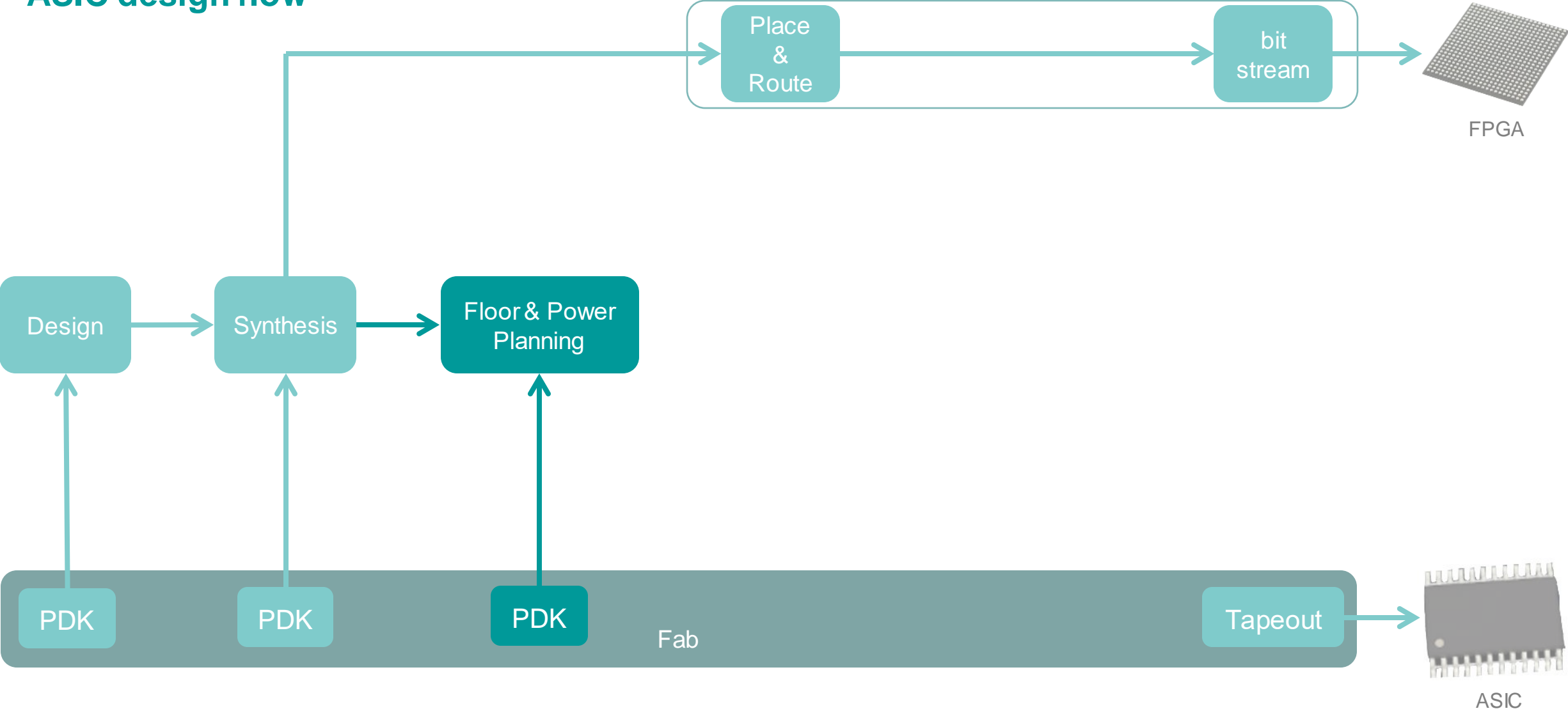


# ASIC design flow

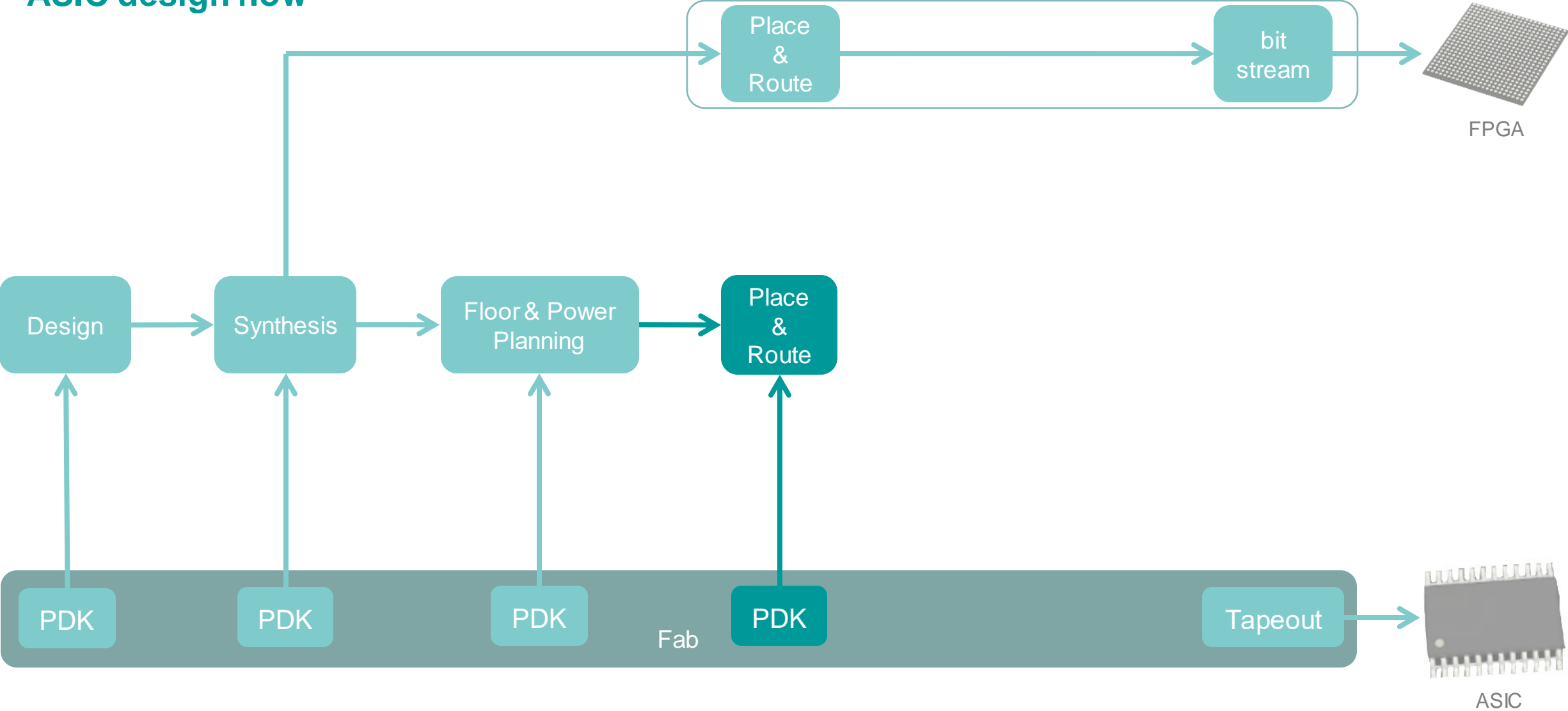




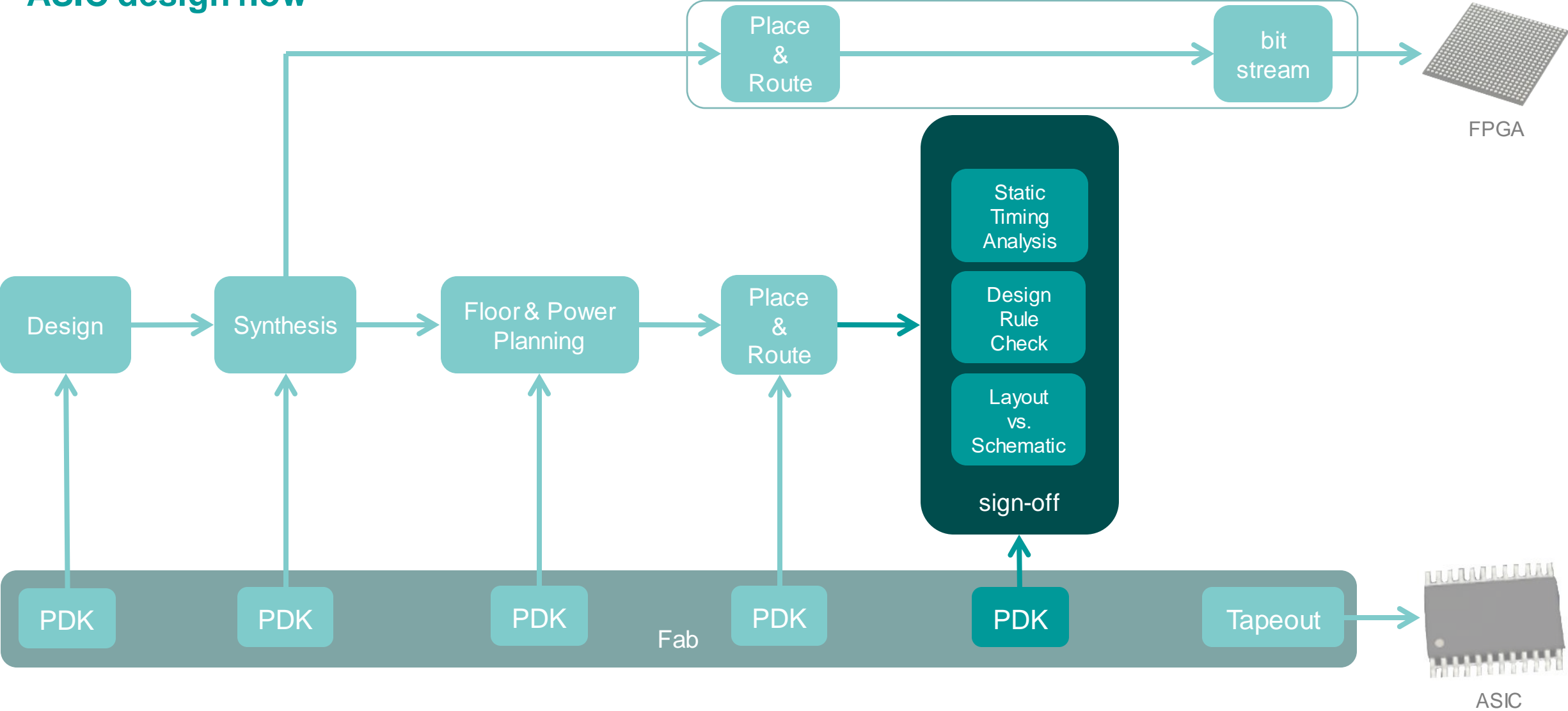
# ASIC design flow



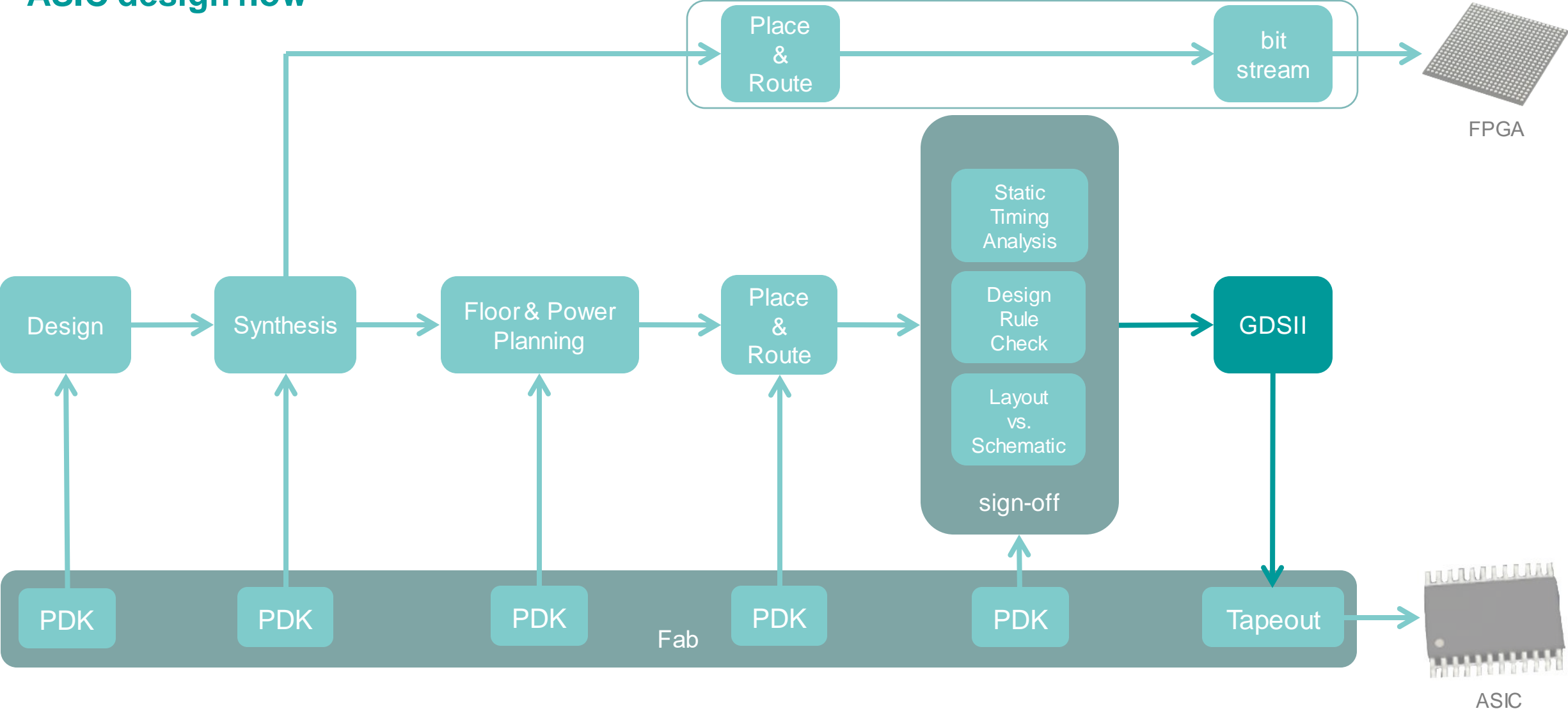
# ASIC design flow



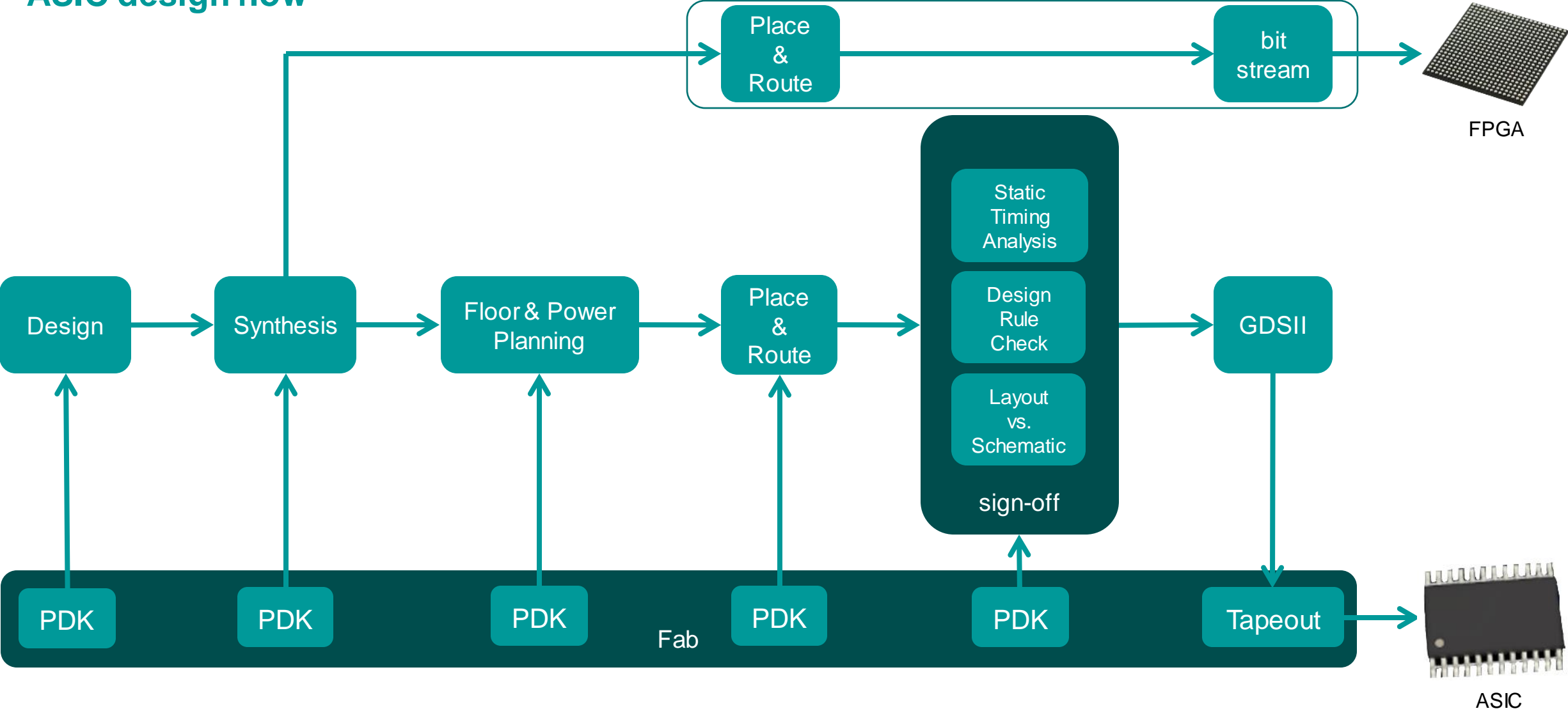
# ASIC design flow



# ASIC design flow

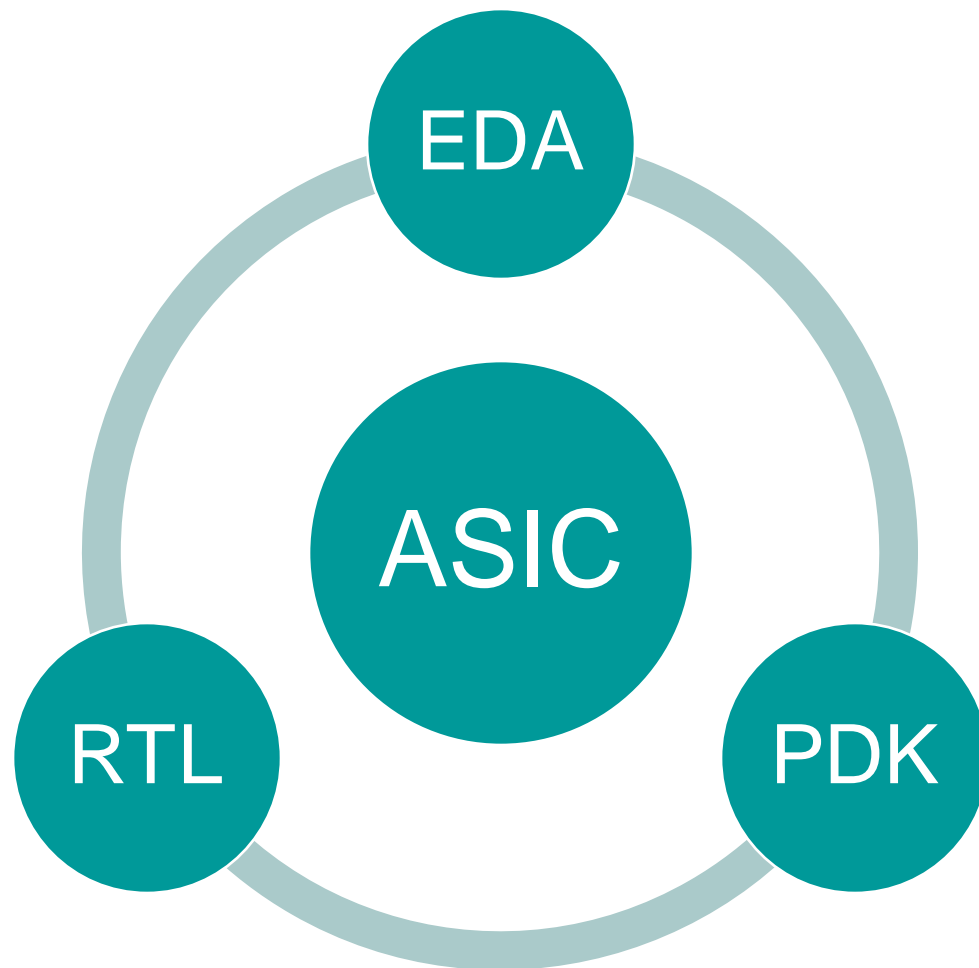


# ASIC design flow

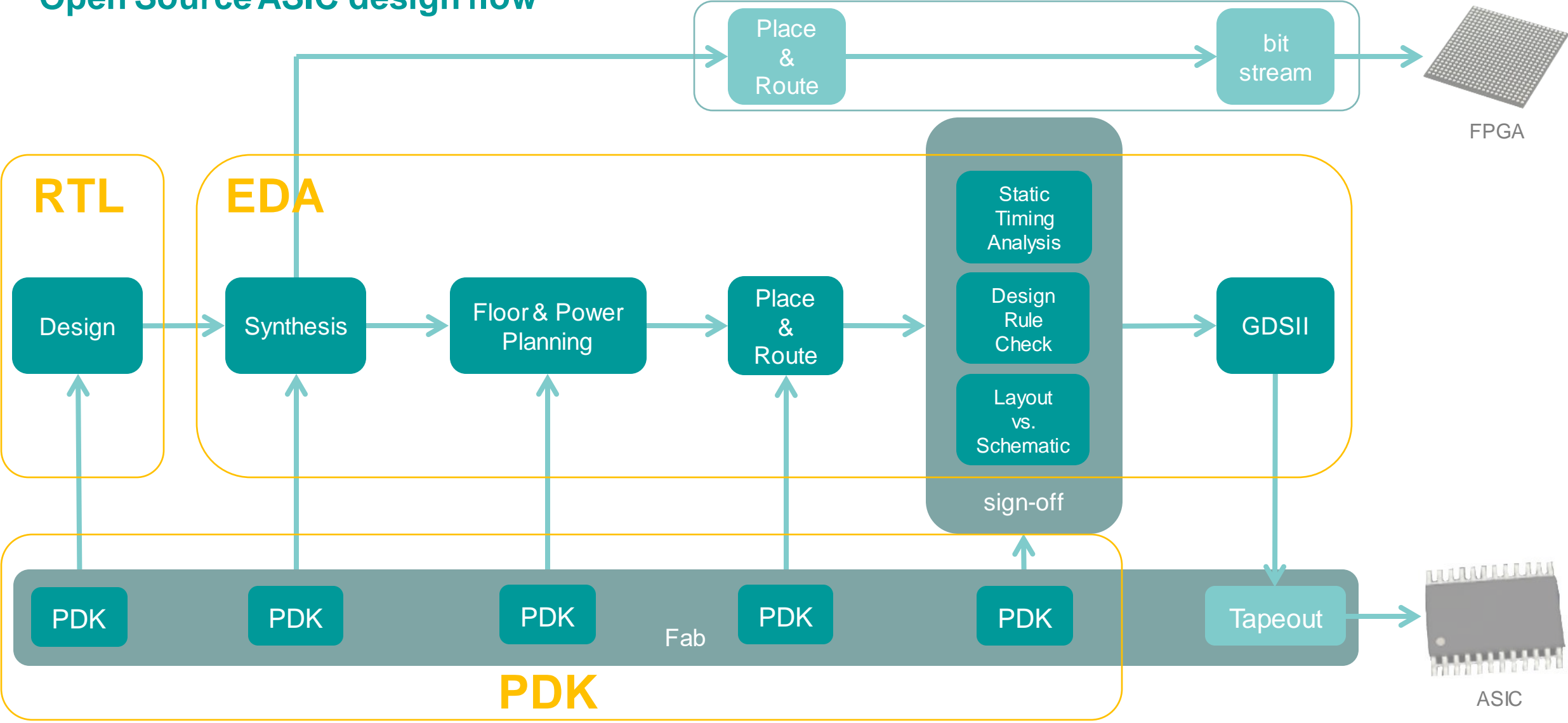


# Open Source ASIC design flow

## The three pillars of ASIC design

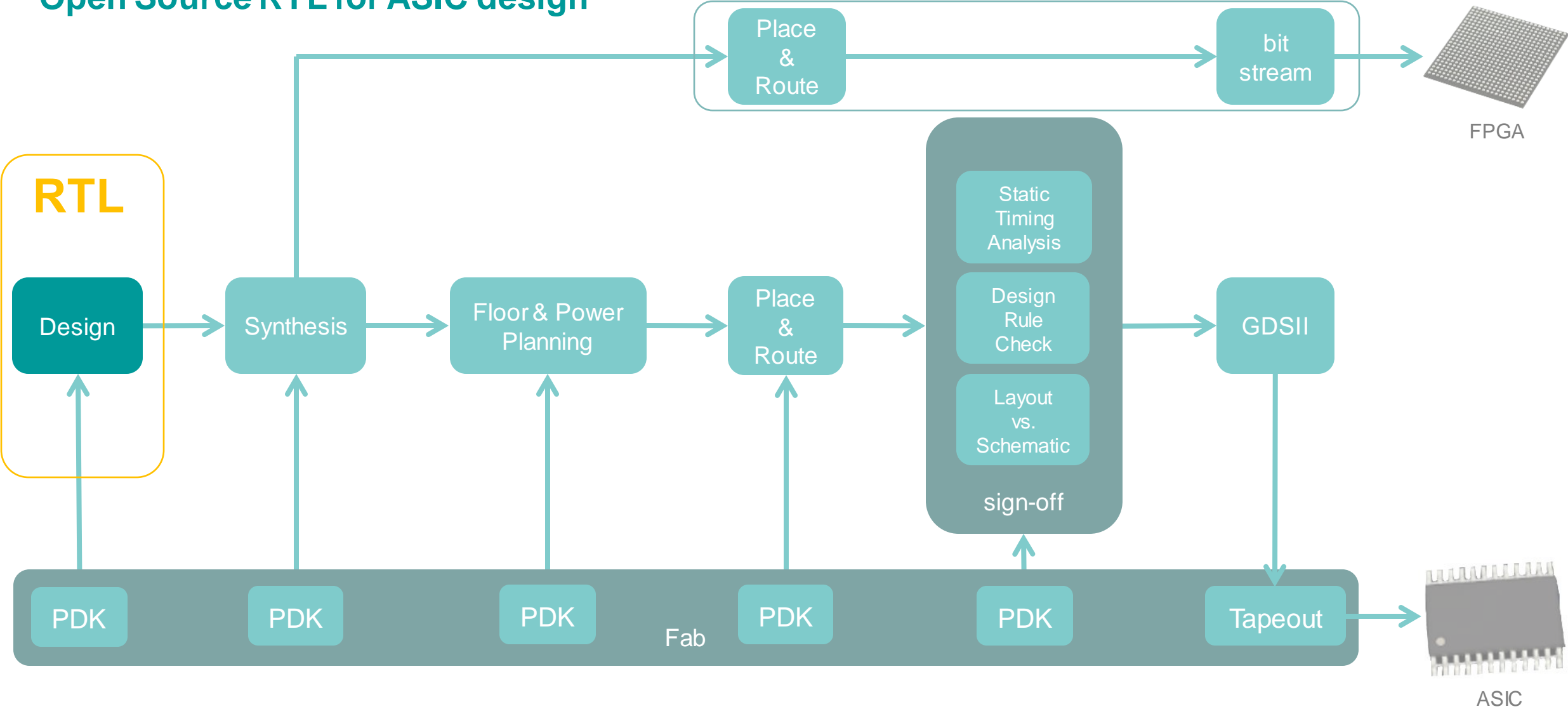


# Open Source ASIC design flow





# Open Source RTL for ASIC design



# Open Source RTL for ASIC design

Hardware Description Languages (HDL)

**Verilog**      **Bluespec**      **Amaranth HDL**  
**VHDL**      **SpinalHDL**      **CHISEL**

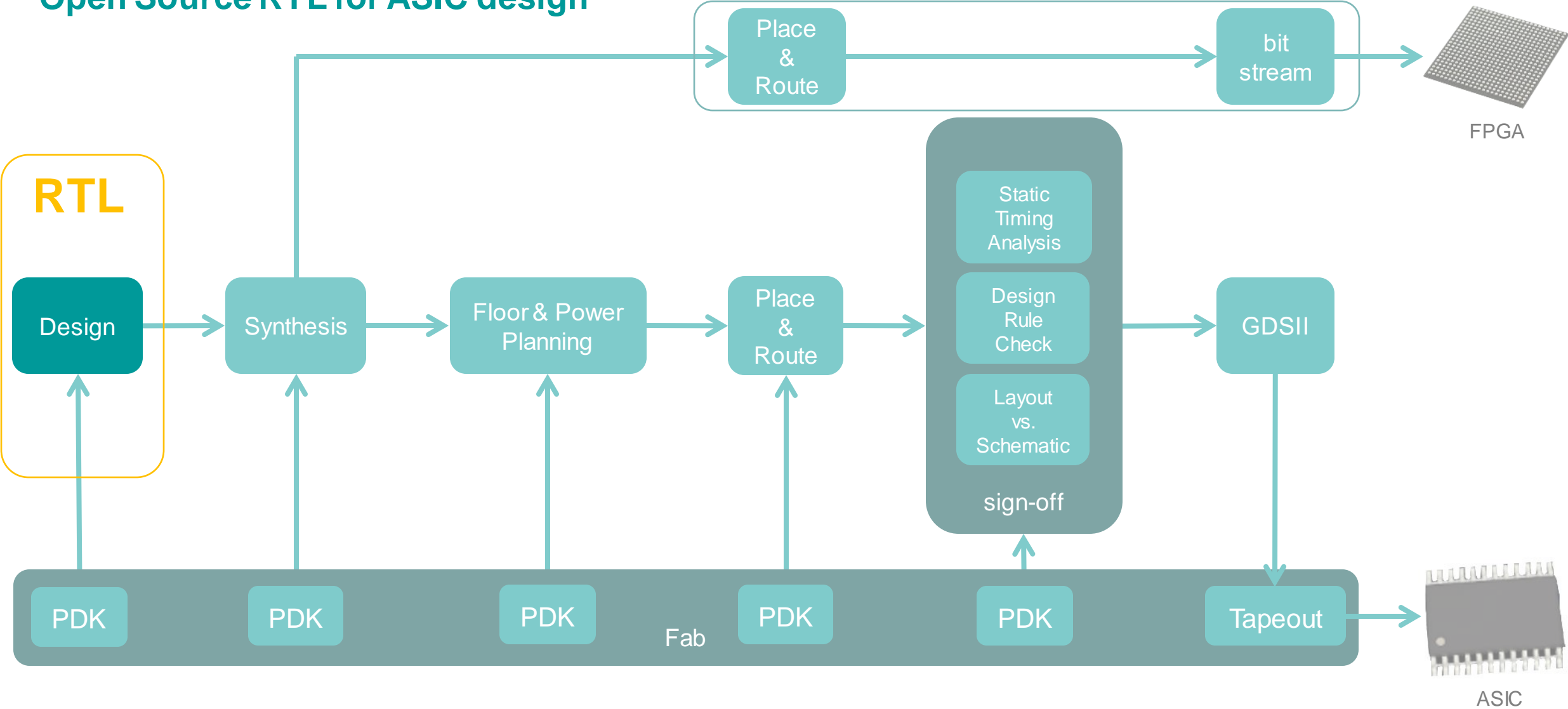
Intellectual Property Cores (IP Cores)

**picorv32**      **OpenRAM**  
**FASoC**

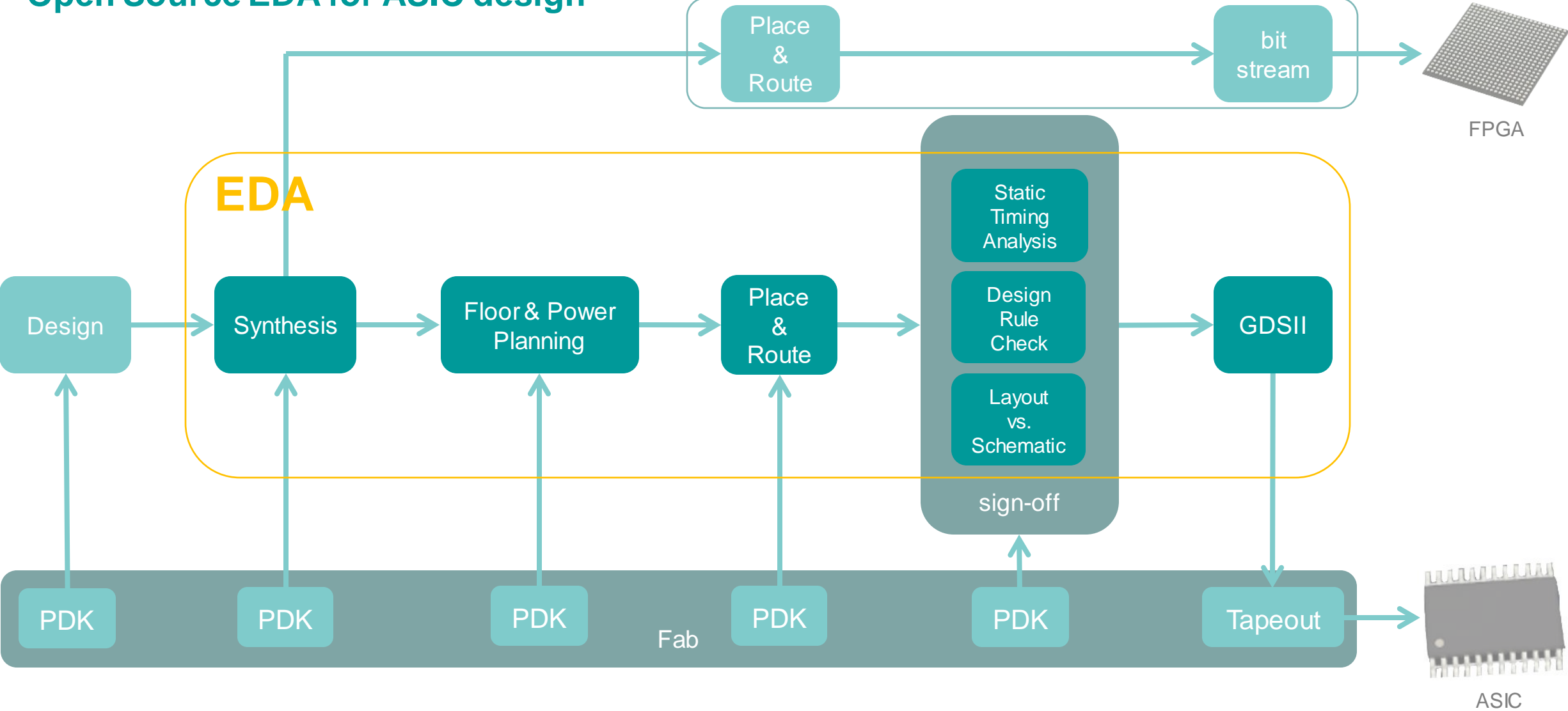
Design Communities

[opencores.org](https://opencores.org)  
[librecores.org](https://librecores.org)  
[github.com](https://github.com)

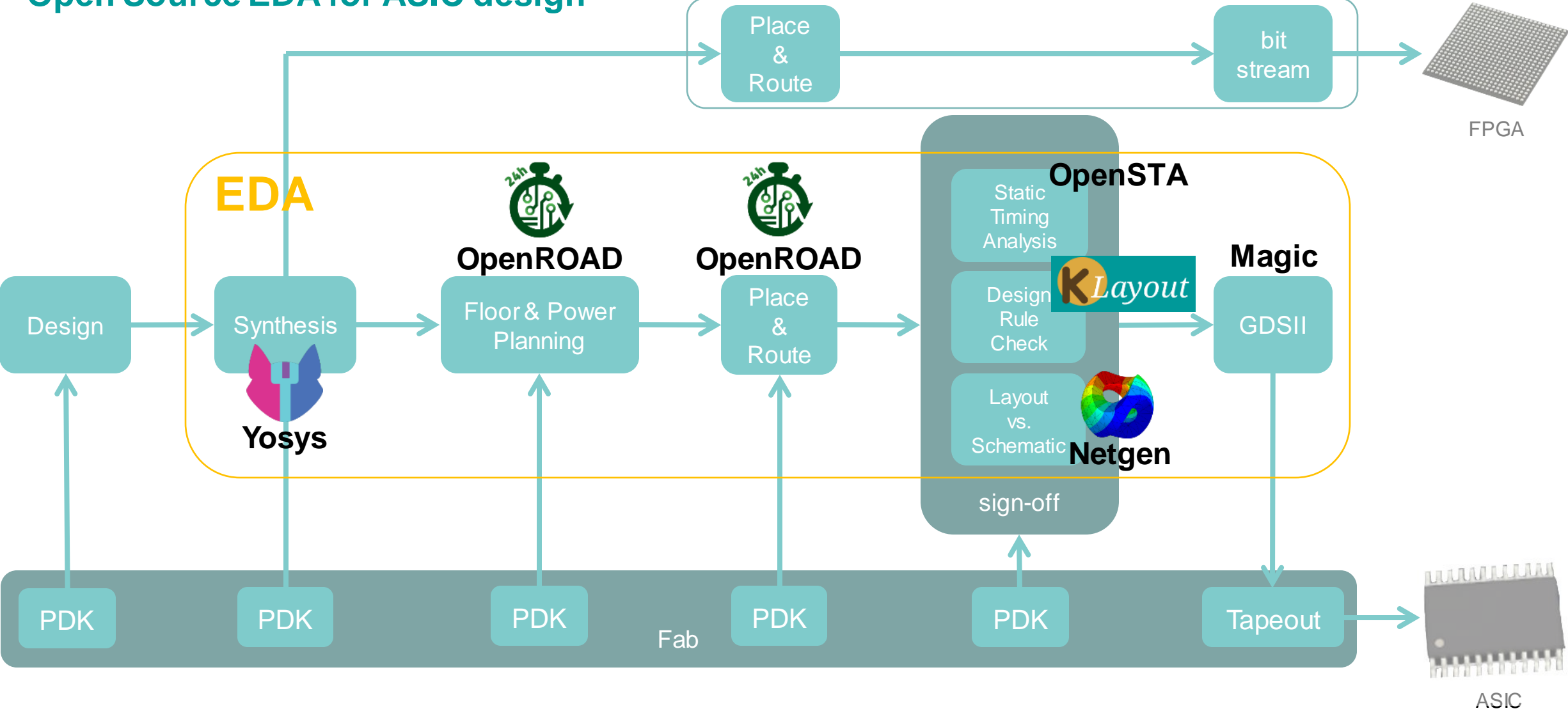
# Open Source RTL for ASIC design



# Open Source EDA for ASIC design



# Open Source EDA for ASIC design

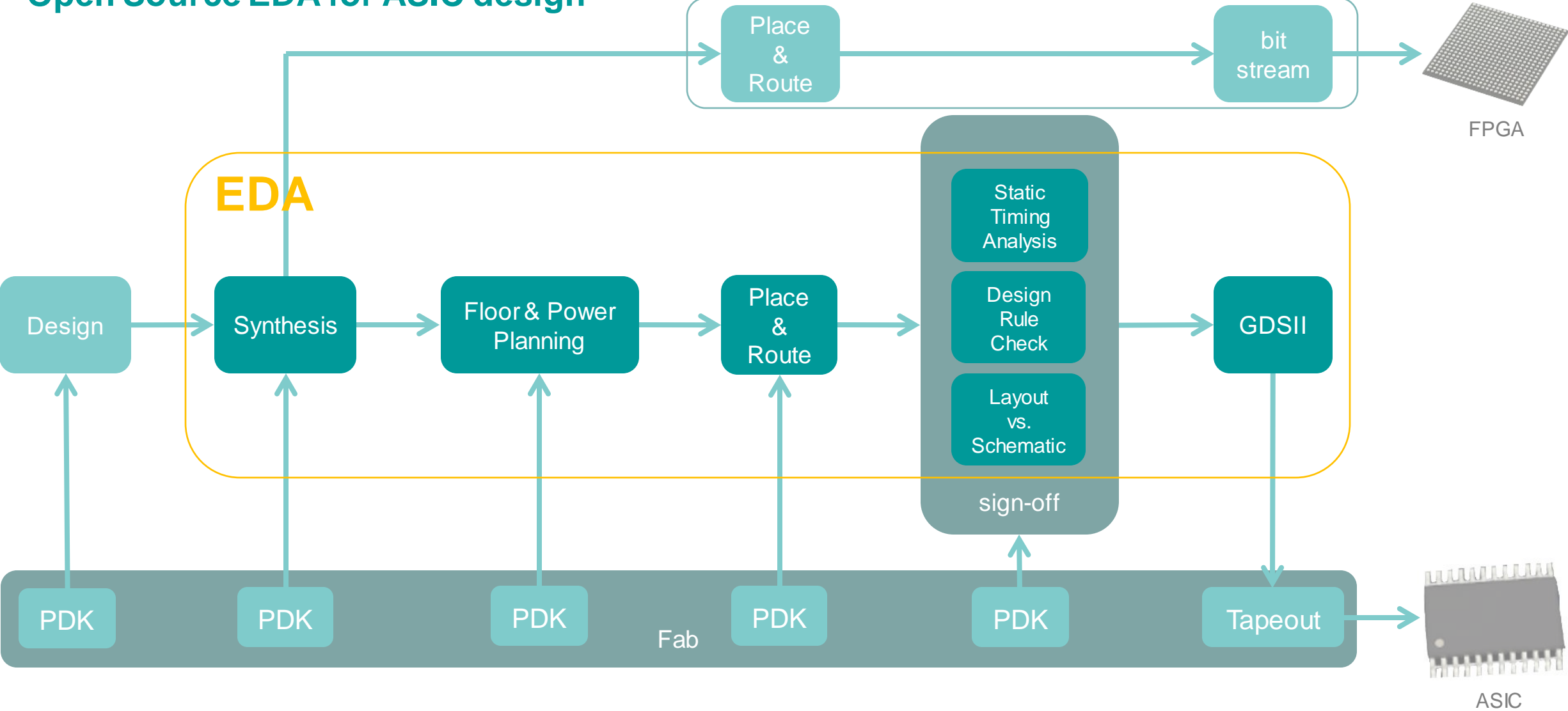


# Open Source EDA for ASIC design

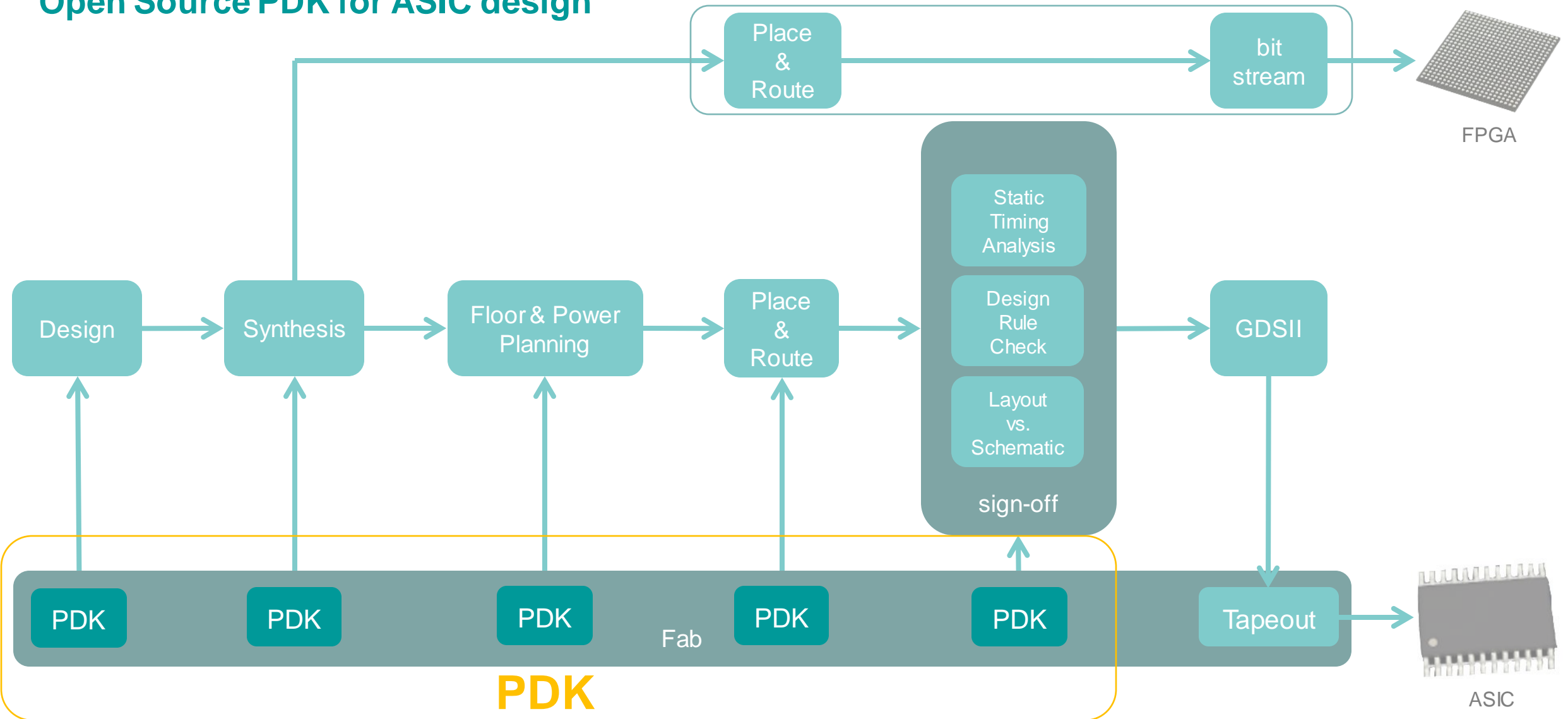
**OpenLane** is an automated RTL to GDSII flow based on several components including **OpenROAD**, **Yosys**, **Magic**, **Netgen** and custom methodology scripts for design exploration and optimization.

The screenshot shows the GitHub repository for 'The-OpenROAD-Project / OpenLane'. The repository is public and has 843 stars, 55 watchers, and 293 forks. The main branch is 'master' with 3 branches and 275 tags. The repository contains a file structure with folders like .github, configuration, dependencies, designs, docker, docs, regression\_results, scripts, tests, and files like .flake8, .gitattributes, .gitignore, .readthedocs.yml, AUTHORS.md, CONTRIBUTING.md, and Jenkinsfile. The 'About' section describes OpenLane as an automated RTL to GDSII flow based on several components including OpenROAD, Yosys, Magic, Netgen and custom methodology scripts for design exploration and optimization. The 'Releases' section shows 275 tags and the 'Packages' section is also visible.

# Open Source EDA for ASIC design

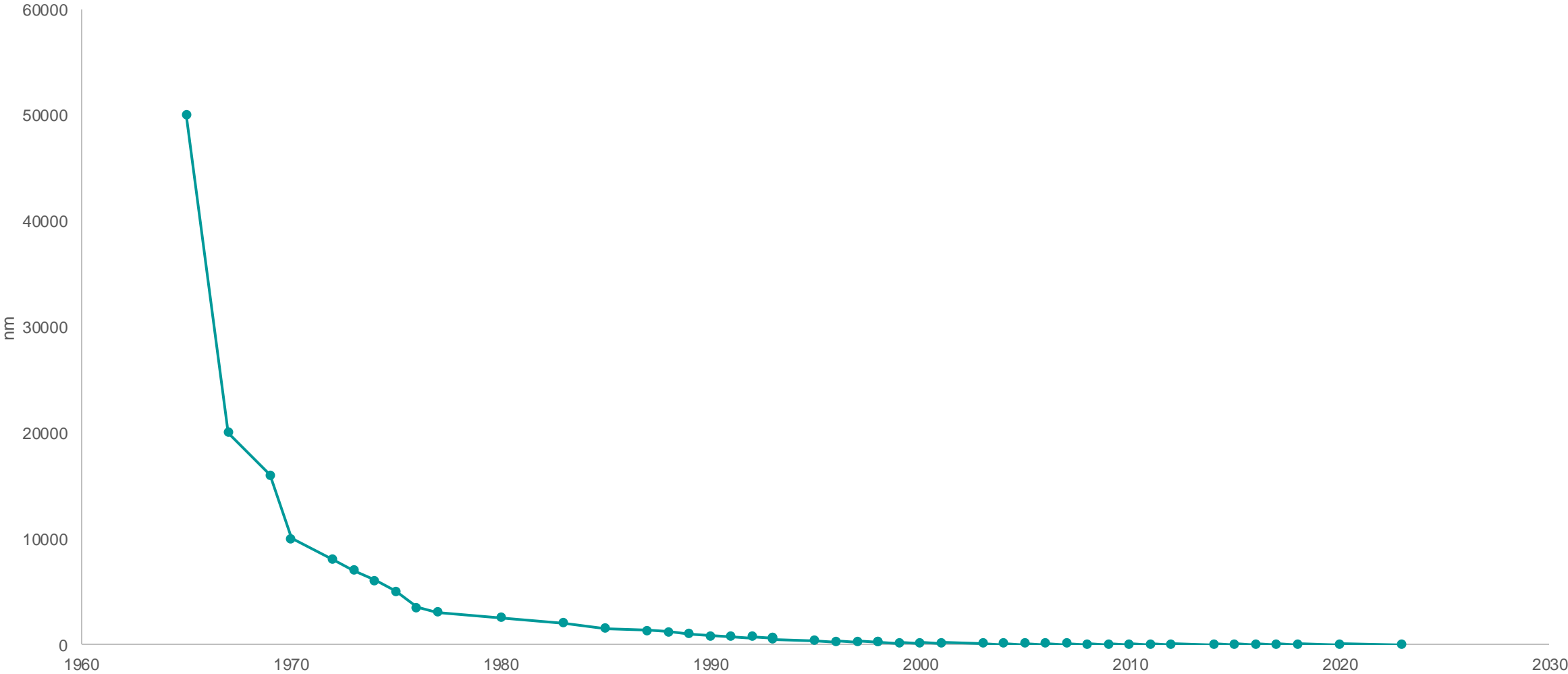


# Open Source PDK for ASIC design

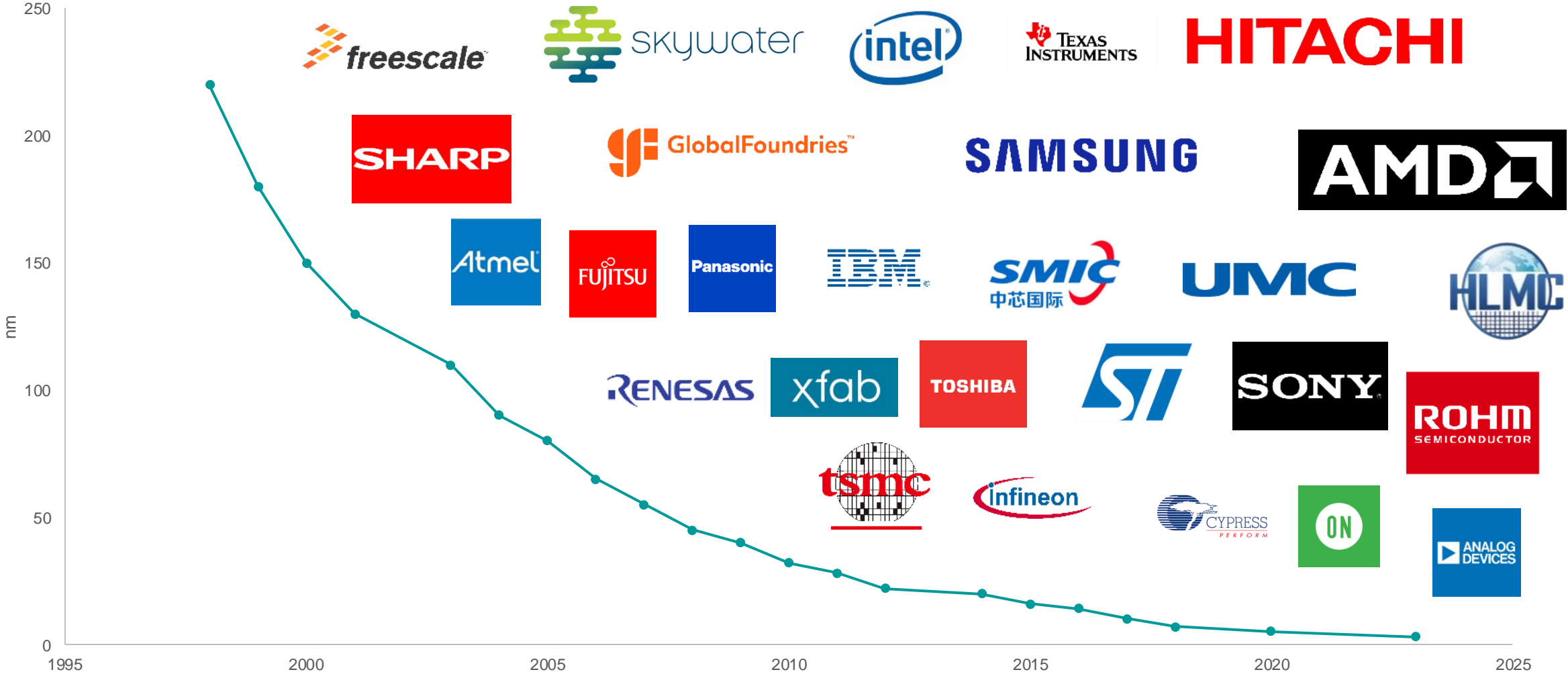




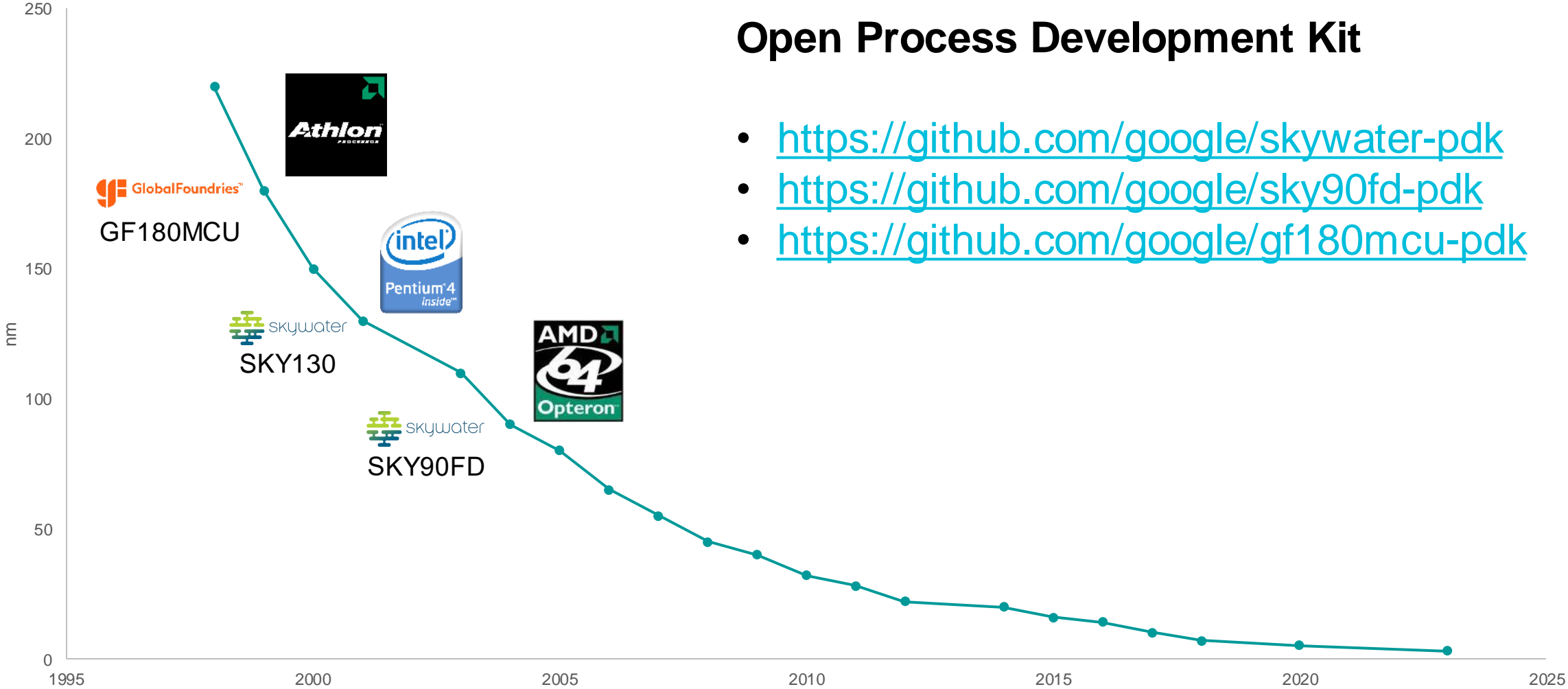
# Process nodes from ~1960 to ~2020



# Process nodes from ~2000 to ~2020



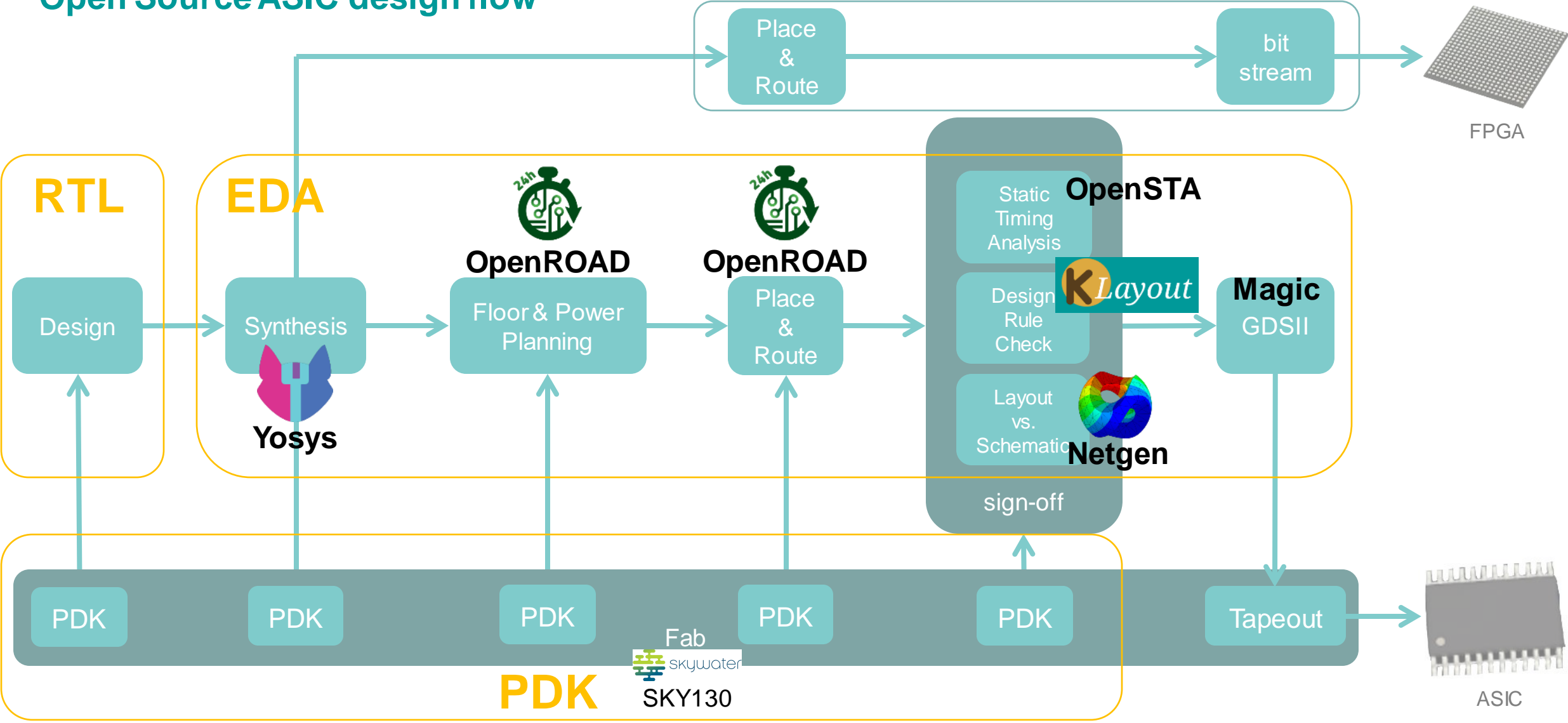
# Process nodes from ~2000 to ~2020



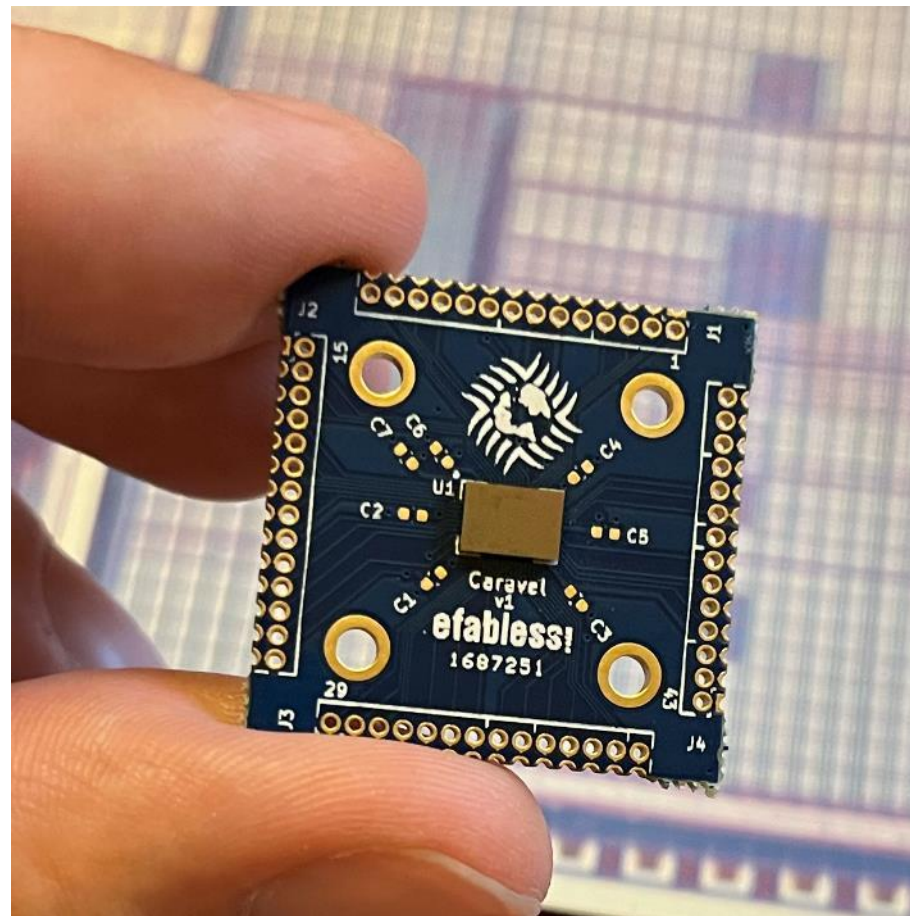
## Open Process Development Kit

- <https://github.com/google/skywater-pdk>
- <https://github.com/google/sky90fd-pdk>
- <https://github.com/google/gf180mcu-pdk>

# Open Source ASIC design flow



## Can we build an Open Source ASIC?



yes we can

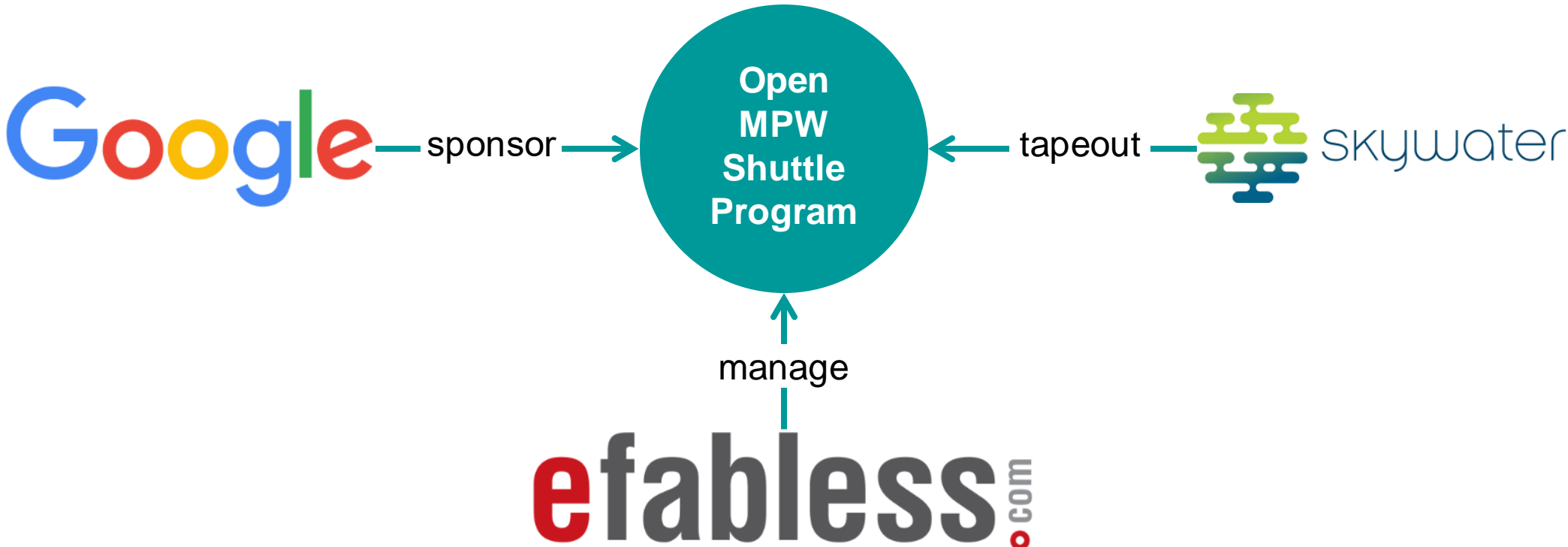


# How to build your own ASIC

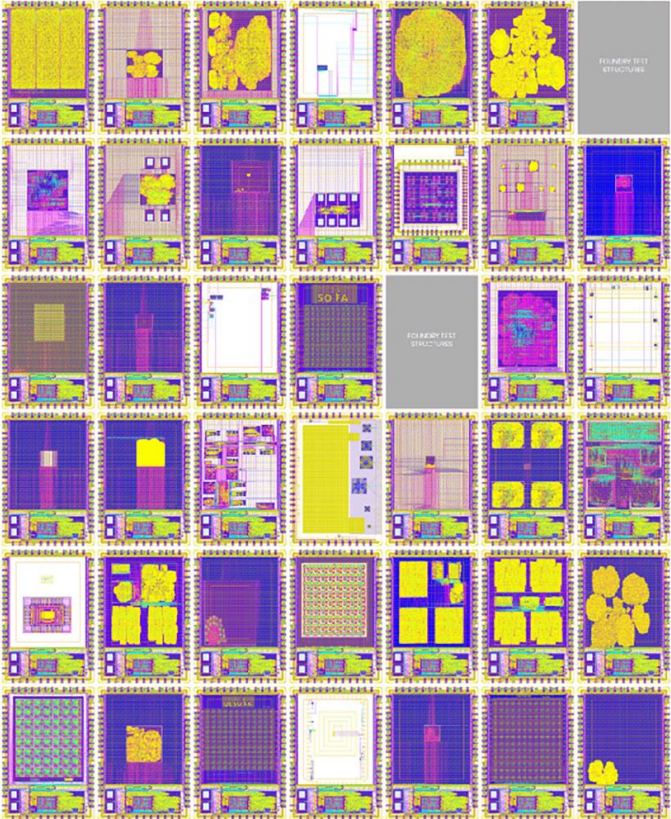
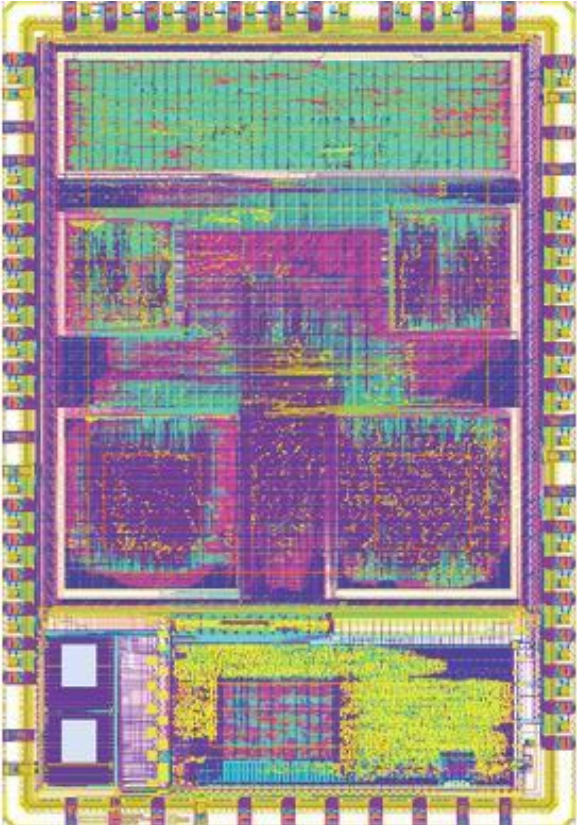
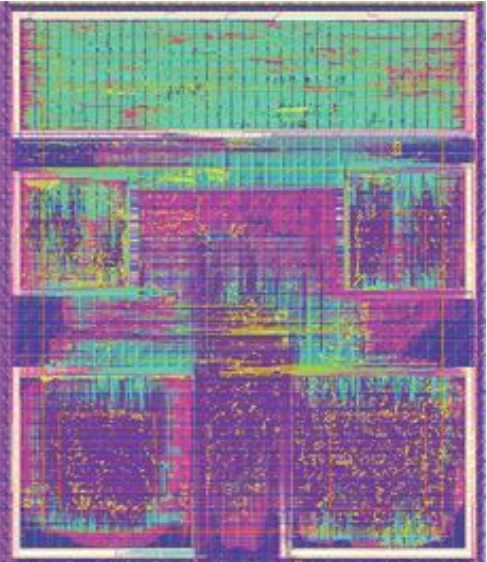
# Open Multi Project Wafer Shuttle Program

## Program details

- 8 shuttles
- 40 designs per shuttle
- 10mm<sup>2</sup> for each design
- everything is/must be Open Source



# Open Multi Project Wafer Shuttle Program



**Project Design**

**Caravel Harness  
with Project Design**

**Multi Project  
Integration**



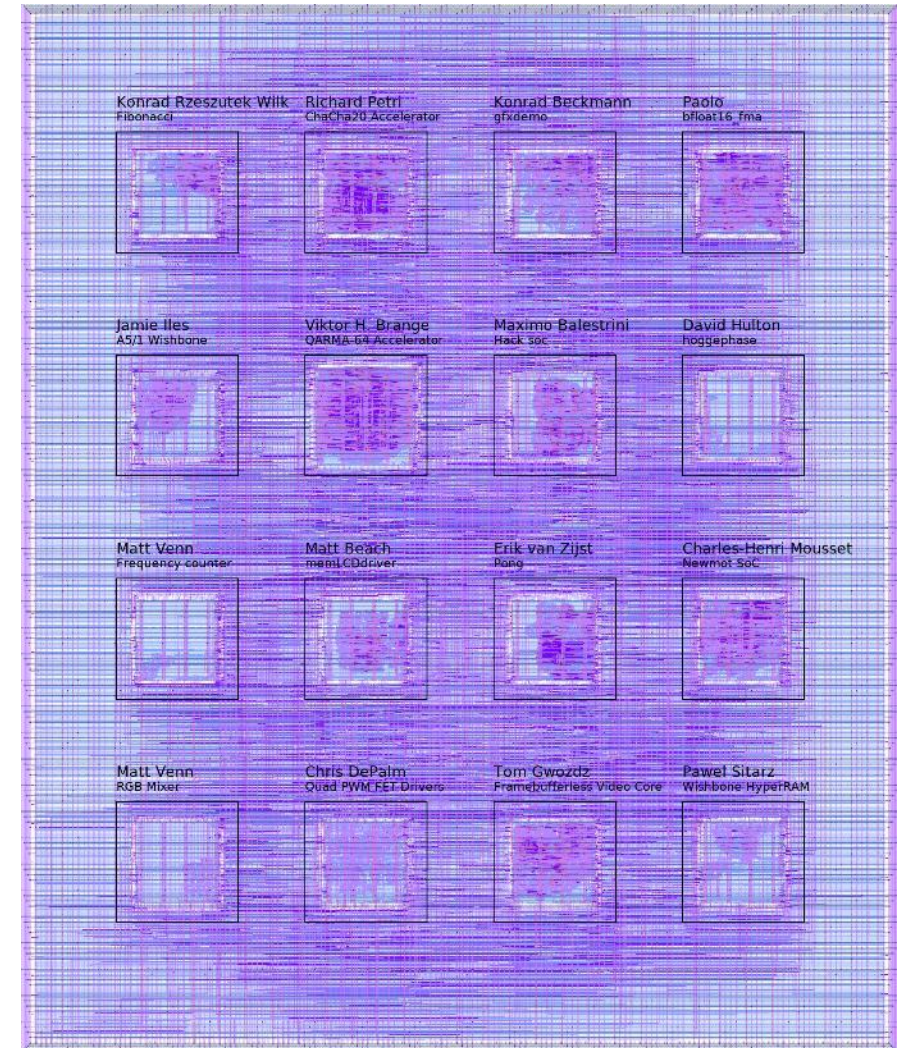
# Zero 2 ASIC

## Program details

- based on the Open MPW shuttle program
- 16 designs per Caravel user-space
- 300µm<sup>2</sup> for each design
- multi-project Caravel: <https://github.com/mattvenn/caravel-mph>



Organized by  
@matthewvenn



16 projects in one Caravel user space  
for MPW-2

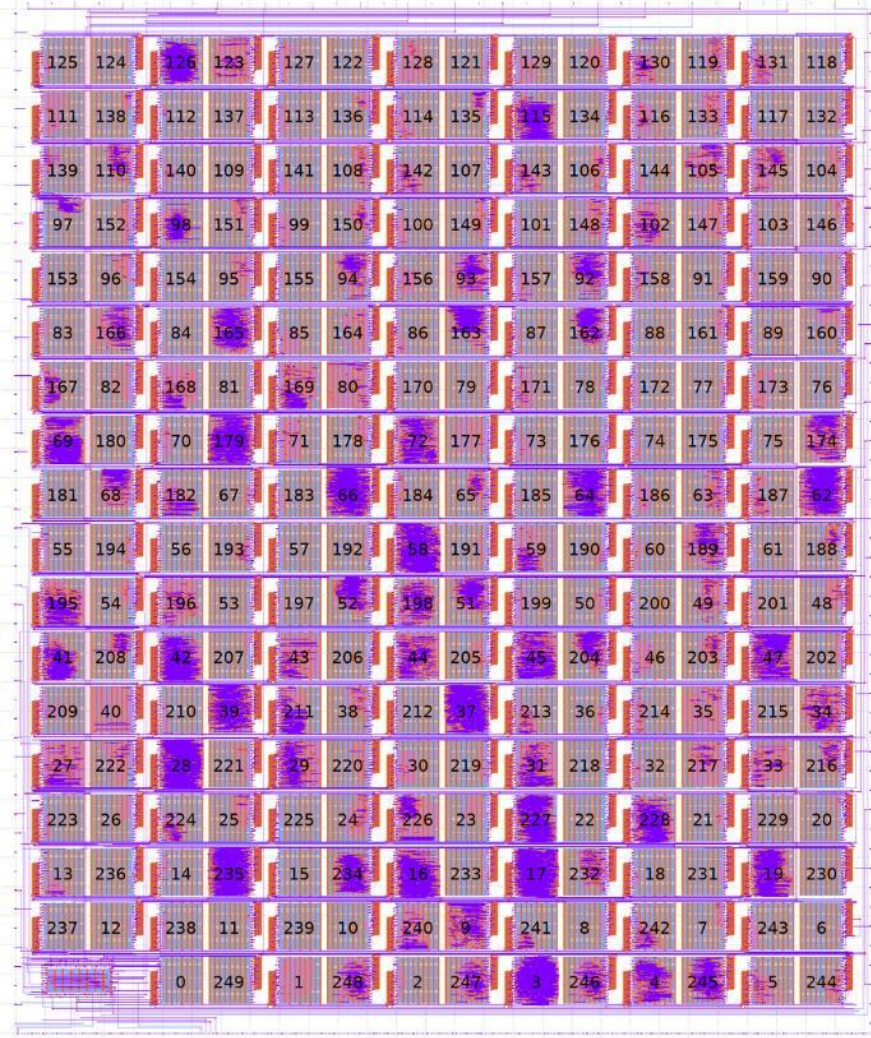
# Tiny Tapeout

## Program details

- tapeout guarantee with payment (25\$ for design / 100\$ for IC&PCB)
- 249 designs per Caravel user-space
- 150µm \* 170µm for each design



Organized by  
[@matthewvonn](#)



249 projects in one Caravel user space

# Estimated cost per design if Google doesn't pay anymore

	Multi Project Wafer	Zero2Asic	Tiny Tapeout
parts per design	300	~18	~1
design quantity	1	16	249
design size	10mm <sup>2</sup>	300μm <sup>2</sup>	150μm x 170μm
design cost	\$9,750.00	\$609.38*	\$39.16*
design examples	<ul style="list-style-type: none"> <li>• RISC-V based Arduino</li> <li>• Bitcoin mining accelerator</li> <li>• FPGA</li> <li>• I2C controller</li> </ul>	<ul style="list-style-type: none"> <li>• ChaCha20 accelerator</li> <li>• RGB mixer</li> <li>• A5/1 crypto block</li> <li>• Pong</li> </ul>	<ul style="list-style-type: none"> <li>• temp sensor</li> <li>• CRC calculator</li> <li>• guitar tuner</li> <li>• 12bit PDP8 CPU</li> </ul>

\* naïve estimation

# Understanding semiconductors on a substrate level



Layers

- active
  - p substrate
  - n well
  - n diffusion
  - p diffusion
  - p tap
  - n tap
- passive
  - polysilicon
  - polyres
  - metal1
  - mim capacitor
  - metal2
- via
  - metal1 via
  - metal2 via

CROSS SECTION & DRC ✓ SIMULATION

Plot signals: in out +

Input voltage:  
Min: 0V  
Max: 5V  
Pulse delay: 0µs  
Rise time: 50µs  
Time scale: 60µs

Show SPICE (advanced)

wokwi / siliwiz Public

Code Issues 14 Pull requests 2 Actions Projects Security Insights

main

Go to file Add file Code

About Silicon Layout Wizard  
app.siliwiz.com  
Readme View license 85 stars 5 watching 7 forks

Releases No releases published

Packages No packages published

Contributors 3  
urish Uri Shaked  
mattvenn matt venn  
bovi Daniel Bovensiepen Li

File	Commit Message	Time
.github/workflows	ci: fix branch name in workflow config	last month
.husky	chore: set up precommit hooks (prettier & lint)	last month
.vscode	chore: add recommended vscode extensions	last month
functions	chore: apply Apache 2.0 License #16	last month
patches	feat: drawing new rects, undo+redo	5 months ago
presets	feat: new app layout (#31)	3 weeks ago
public	fix: update favicon	3 weeks ago
src	fix: updated lesson link to tinytapeout resource	last week
.editorconfig	chore: add prettier config	5 months ago
.eslintrc.cjs	chore: set up eslint	last month
.gitignore	feat: app skeleton	5 months ago
.gitpod.yml	chore: change gitpod default to open preview	last month
LICENSE	chore: apply Apache 2.0 License #16	last month
README.md	docs: add badges to README	last month
package-lock.json	chore: npm audit fix	3 weeks ago
package.json	feat: new app layout (#31)	3 weeks ago

<https://app.siliwiz.com>

# Design electrical circuits in the browser



The screenshot shows the Wokwi web interface. At the top, there are buttons for 'SAVE', 'SHARE', and 'SIGN IN'. The main area is split into two panes. The left pane shows a 'README.md' file with the following content:

```
1 # Tiny Tapeout 3 Template Project
2
3 Use this template to start your Tiny Tapeout 3 project.
4
5 To learn more, check out our Digital Design Guide at https://tinytap
6
7
```

The right pane shows a circuit diagram with a simulation control bar containing a play button, a plus sign, and a menu icon. The circuit diagram includes a microcontroller, a resistor, a capacitor, and various logic gates.

The screenshot shows the Wokwi GitHub profile page. The profile name is 'Wokwi' with 227 followers and the website 'https://wokwi.com'. The navigation bar includes 'Overview', 'Repositories 106', 'Projects', 'Packages', and 'People'. The 'Pinned' section lists the following repositories:

- avr8js** (Public): Arduino (8-bit AVR) simulator, written in JavaScript and runs in the browser / Node.js. 381 stars, 61 forks.
- rp2040js** (Public): A Raspberry Pi Pico Emulator in JavaScript. 255 stars, 24 forks.
- wokwi-features** (Public): Wokwi Feature requests & Bug Reports. 49 stars, 9 forks.
- wokwi-elements** (Public): Web Components for Electronics and IoT Parts. 131 stars, 36 forks.
- wokwi-docs** (Public): Documentation for the Wokwi Simulator. 78 stars, 168 forks.
- splendida** (Public): 256 WS2812B LEDs arranged in Fermat's Spiral Shape. 39 stars, 5 forks.

<https://wokwi.com>



## “thought-provoking” answers

- 1) Do you have basic knowledge how an ASIC is made?  
**I guess somehow...**
- 2) Do you think an Open Source ASIC is possible today?  
**Looks like it**
- 3) Do you dare to design your own ASIC?  
**Sure**

# | epilog

## Q&A

Daniel Bovensiepen

[daniel.bovensiepen@siemens.com](mailto:daniel.bovensiepen@siemens.com)

Siemens

Beijing, China

Inverter by @marunmagesh

