Chapter 2 - OpenROAD workflow

Course authors (Git file)

Section 1

History of OpenROAD

Foundations and Realization of Open, Accessible Design (OpenROAD)

At the top of the documentation:

https://openroad-flow-scripts.readthedocs.io

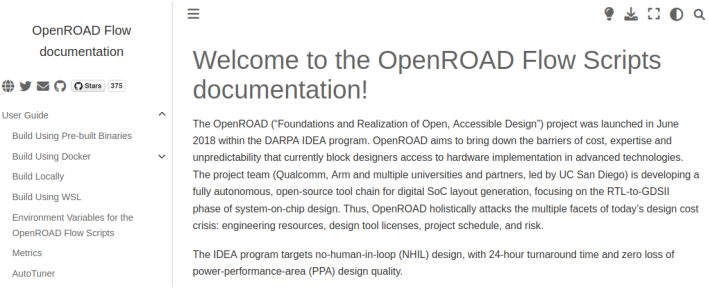


Figure 1: Darpa IDEA [1]

[1] Source: Screenshot of the webpage.

Darpa IDEA

https://www.darpa.mil/research/programs/intelligent-design-of-electronic-assets

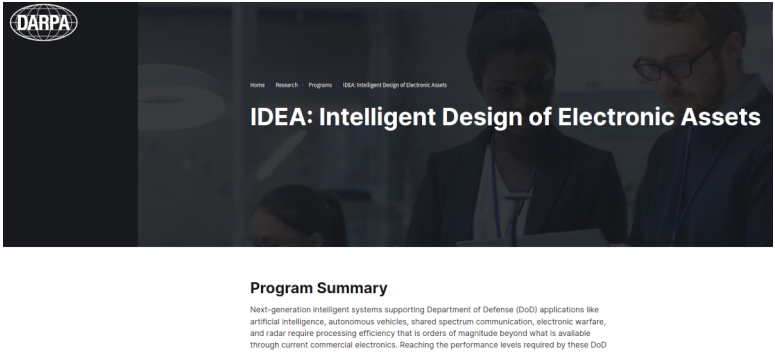


Figure 2: Darpa IDEA [2]

[2] Source: Screenshot of the webpage.

Darpa ERI

2018/2019 Darpa ERI, cadence and the people

https://community.cadence.com/tags/openroad

Start reading from the bottom!

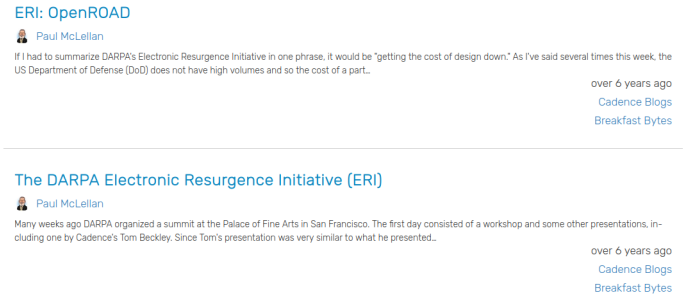
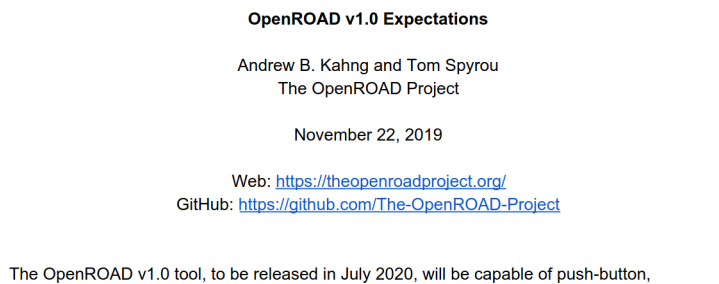


Figure 3: Darpa ERI [3]

OpenROAD V1.0

Document with OpenROAD V1.0 Expectations:

https://vlsicad.ucsd.edu/NEWS19/OpenROAD%20RTL-to-GDS%20v1.0%20Expectations.pdf



OpenROAD v1.0 Expectations

Andrew B. Kahng and Tom Spyrou
The OpenROAD Project

November 22, 2019

Web: https://theopenroadproject.org/
GitHub: https://github.com/The-OpenROAD-Project

The OpenROAD v1.0 tool, to be released in July 2020, will be capable of push-button, DRC-clean RTL-to-GDS layout generation in a commercial FinFET process node. The tool is currently visible here. In its v1.0 form, it will be integrated on an incremental substrate provided by the OpenDB database and the OpenSTA static timing engine. It will also offer users and

Figure 4: OpenROAD v1.0 [4]

V1.0 Roadmap

How to deal with these expectations:

https://eri-summit.darpa.mil/docs/ERIsummit2019/posh/08IDEA%20UCSD%20Website.pdf

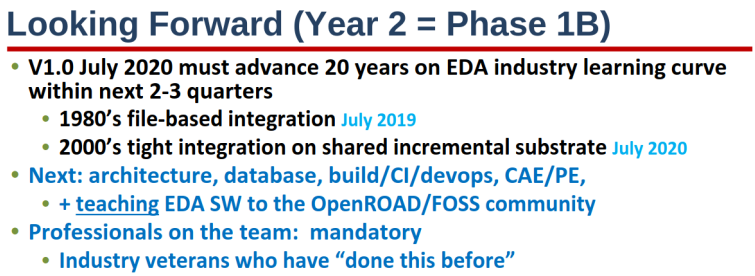


Figure 5: Looking forward [5]

[5] Source: Screenshot of the webpage.

Courses for OpenROAD

A single excellent project (some of us might know):

https://theopenroadproject.org/Courses/

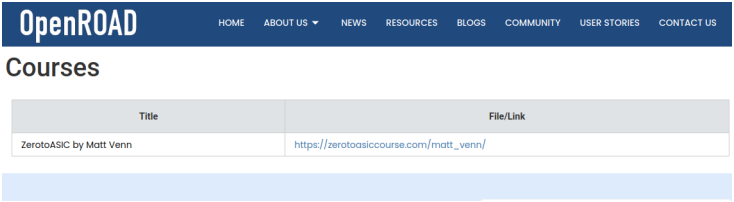


Figure 6: Courses [6]

[6] Source: Screenshot of the webpage.

Help is on the way

Kudos to this course:

https://theopenroadproject.org/news/6455/

Figure 7: This course [7]

[7] Source: Screenshot of the webpage.

Section 2

OpenROAD Flow Scripts

Flow steps: Overview

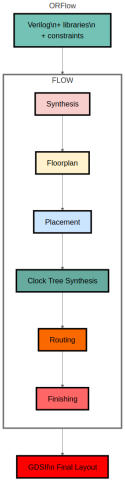


Figure 8: OpenROAD flow steps [8]

Flow step: Synthesis

Wikipedia:

https://en.wikipedia.org/wiki/Logic_synthesis

Zero to ASIC Terminilogy:

https://www.zerotoasiccourse.com/terminology/synthesis/

One sentence:

Synthesis takes the RTL (HDL, Verilog) and transforms it to a netlist (wires and components, standard cells).

Flow step: Floorplan

Wikpedia:

https://en.wikipedia.org/wiki/Floorplan_(microelectronics)#Floorplanning

Zero to ASIC Terminilogy:

https://www.zerotoasiccourse.com/terminology/floorplan/

One sentence:

Floorplaning creates a rough plan of the component areas (netlist, padframe,) and macros onto the space of the microchip (die area).

Flow step: Placement

Wikpedia:

https://en.wikipedia.org/wiki/Placement_(electronic_design_automation)

Zero to ASIC Terminilogy:

https://www.zerotoasiccourse.com/terminology/place_and_route/

One sentence:

Placement algorithmically determines fixed places for every component of the design, according to the floorplan.

Flow step: Clock Tree Synthesis

Wikpedia:

https://en.wikipedia.org/wiki/Physical_design_(electronics)#Clock_tree_synthesis

Zero to ASIC Terminilogy:

https://www.zerotoasiccourse.com/terminology/place_and_route/ Scroll down the page!

One sentence:

Clock tree synthesis builds a tree out of signal paths (with the root as the start point) for a given clock signal under the premise of given timing constraints.

Flow step: Routing

Wikpedia:

https://en.wikipedia.org/wiki/Routing_(electronic_design_automation)

Zero to ASIC Terminilogy:

https://www.zerotoasiccourse.com/terminology/place_and_route/ Scroll down the page!

One sentence:

Routing algorithmically creates all the wires between the components of the design obeying all given design rules and timing contraints.

Flow step: Finishing

Wikpedia:

https://en.wikipedia.org/wiki/Physical_design_(electronics)#Layout_post_processing

One sentence:

The step of finishing the design can include creating metal fills, finalising the logs and reports, merging multiple GDS to a final one and do the final LVS/DRC checks.

Flow components

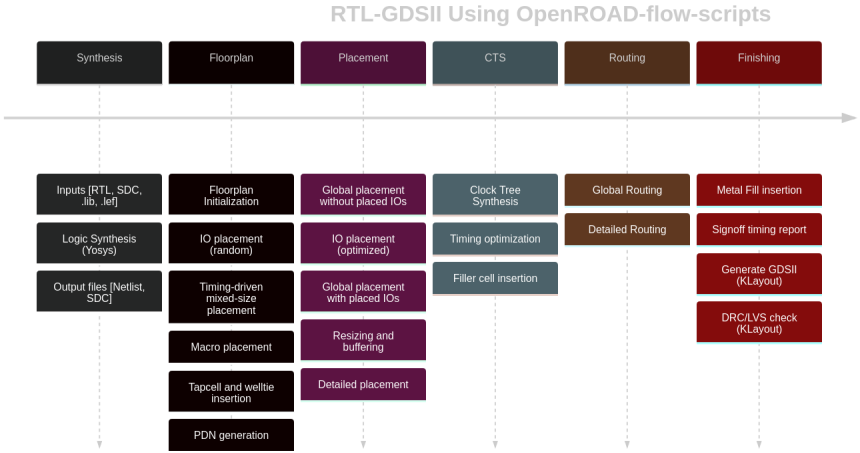


Figure 9: OpenROAD flow components [9]

Section 3

Fill insertion

Fill insertion

There are two flow components that take care of filling area gaps in the design:

- Filler cell insertion
- Metal fill insertion

Both will be explained in the following section.

Filler cells: Reasons

Why filling the design area with more cells?

- After floorplaning and placement there will always be gaps between the standard cells
- These gaps must be filled to keep the continuity of
  - n-well and p-well spacing
  - railing
  - implants
- The filling with cells helps against the WPE (well proximity effect)

Filler cells: Properties

Filler cells

- have no logic function
- have no inputs and no outputs
- are not in the netlist of the design
- are available in different widths

Filler cells: GDS

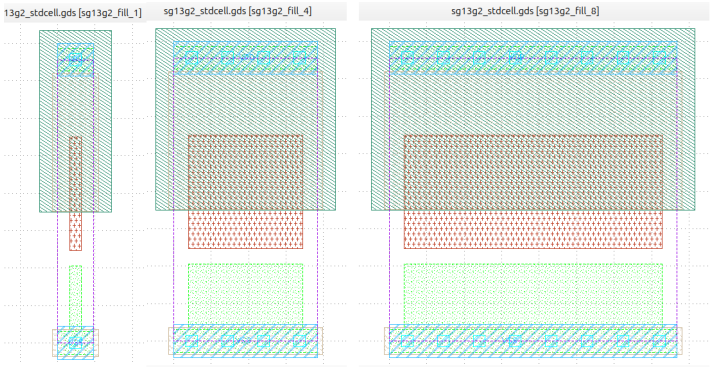


Figure 10: Filler cells in different widths (1, 4, 8)

Metal fill

What is metal filling?

- Metal filling adds polygons and shapes to the metal layer(s).
- These fillings have no logic functionality.
- The purpose is creating a more even or uniform distribution of the metal area.
- Physical reasons are planarity and reducing thinkness variations (CMP).
- Metal filling can affect timing and signal integrity.
- Metal filling is done algorithmical (mostly with scripts).

More to read:

https://semiengineering.com/knowledge_centers/materials/fill/

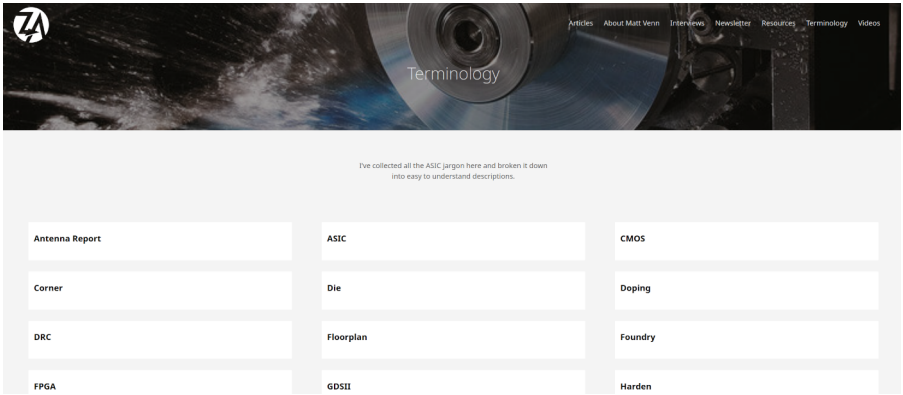
Section 4

Resources

Help with the terminology

Searching the terms with a standard search engine might not bring usefull results every time. Matt Venn created a page for EDA terminology:

https://www.zerotoasiccourse.com/terminology/



The list of open-source tools

The list of flow steps and flow components in ORFS contains the information about the original open-source tools.

We will dive a little deeper into this list with the course training:

https://openroad-flow-scripts.readthedocs.io/en/latest/tutorials/FlowTutorial.html#running-the-automated-rtl-to-gds-flow

Additional links

Awesome open-source asic resources:

https://github.com/mattvenn/awesome-opensource-asic-resources

Zero-to-ASIC resources list:

https://www.zerotoasiccourse.com/resources/

AIC 2025 lectures by Carsten Wullf:

https://github.com/wulffern/aic2025

https://analogicus.com/aic2025/

Meet the developers

There is a slack community with most of the open-source silicon projects and people in it:

https://open-source-silicon.dev

This is the perfect place, if you

- search for people you heard of in open-source EDA.
- want to reach out to tool developers.
- have all sorts of questions in open-source EDA.
- look for a specific problem and want to see if others already are in a discussion about it.
- want to share your experience and help others.

Most of the tools in open-source EDA have their own channel in this slack. Just try search and add for the channel names.

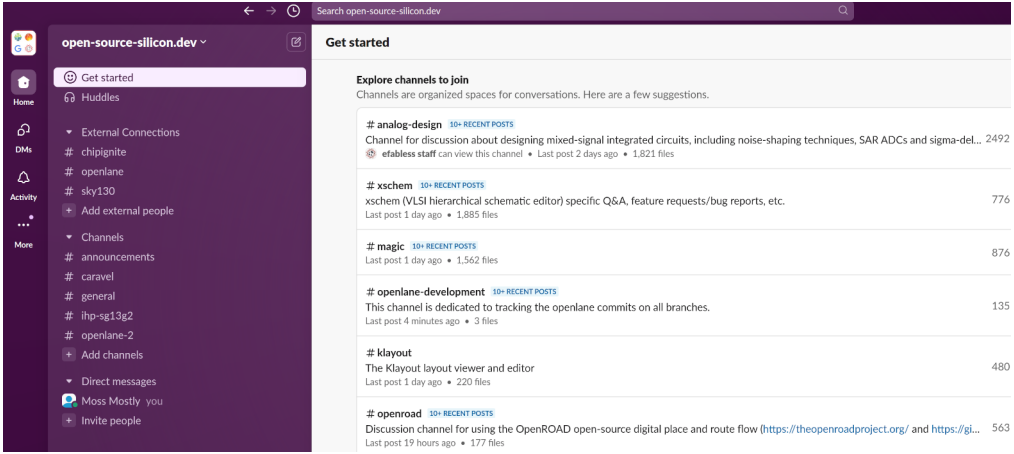


Figure 12: Channels in the open-source-silicon.dev slack