**Gate-Level to GDSII with OpenLane**

The usual flow is RTL to GDSII, but here we have no RTL file, instead we want to convert a Gate-Level-Design to GDSII. We must skip RTL synthesis manually and prevent the tool from altering our cells. This is a possible workflow:

flow-tcl -init-design-config does not work, instead copy one of the design templates

* from /foss/tools/openlane/2022.07/designs/<designname>
* to /designs/<project-name>/openlane/<yourdesignname>

Write your verilog-file in the format as shown below. You can find names for the high-density standard cells in [https://antmicro-skywater-pdk-docs.readthedocs.io](https://antmicro-skywater-pdk-docs.readthedocs.io/en/test-submodules-in-rtd/contents/libraries/sky130_fd_sc_hd/cells/clkdlybuf4s50/README.html) and the local Verilog library in /foss/pdk/sky130A/libs.ref/sky130\_fd\_sc\_hd/verilog/sky130\_fd\_sc\_hd.v  
  
Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Change the config /designs/<project-name>/openlane/<yourdesignname>/config.tcl

* + Links to your Verilog-files
  + Name of your project
  + Prevent the router and placer from altering your cells by specifying
    1. set ::env(PL\_RESIZER\_TIMING\_OPTIMIZATIONS) {0}
    2. set ::env(PL\_RESIZER\_DESIGN\_OPTIMIZATIONS) {0}
    3. set ::env(GLB\_RESIZER\_TIMING\_OPTIMIZATIONS) {0}
  + Also set your pin order and fanout orientations with
    1. set ::env(FP\_PIN\_ORDER\_CFG) $::env(DESIGN\_DIR)/pin\_order.cfg

start a terminal at /designs/<project-name>/openlane/

run flow-tcl -interactive -design <yourdesignname> -tag <run-name>

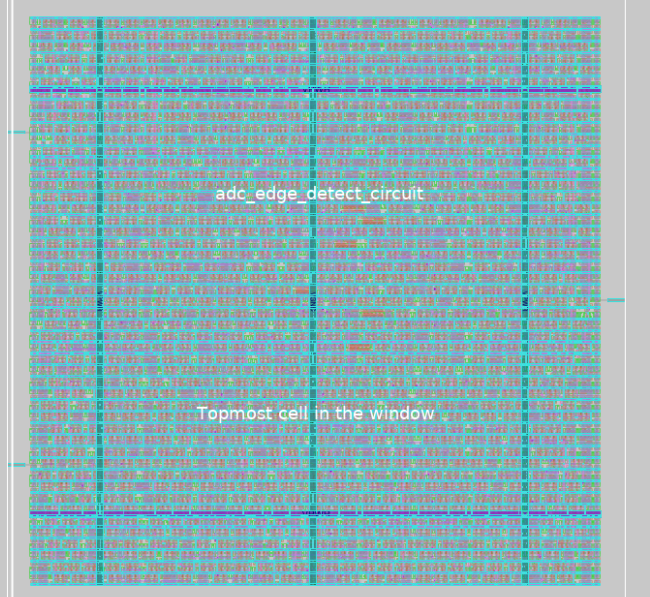
type package require openlane, console should return 0.9

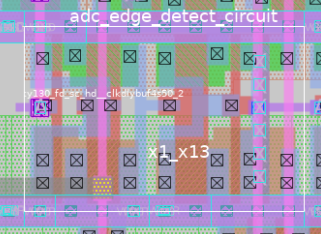
Before you start synthesis, place your already synthesized file in   
..openlane/<yourdesignname>/runs/<run-name>/results/synthesis

1. Type run\_synthesis in the console, synthesizer should recognize there is already a file (which is exactly what we want)  
   Ein Bild, das Text enthält.

   Automatisch generierte Beschreibung
2. Type run\_floorplan
3. Type run\_placement
4. Type run\_routing
5. Type run\_magic

Result:





**Appendix A: Complete list of interactive openlane commands**

<https://openlane-docs.readthedocs.io/en/rtd-develop/doc/advanced_readme.html>

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

**Appendix B: Configuration file of a CLKDLYBUF\_4S50\_2 combinatorial Delay-cell**

**Folder structure:**

/foss/designs/SKY130\_SAR-ADC

-/verilog

-/adc\_edge\_detect\_circuit/ adc\_edge\_detect\_circuit.v

-/openlane

-/adc\_edge\_detect\_circuit/

-/runs/

-/config.tcl

-/pin\_order.cfg

**Contents of config.tcl:**

set ::env(DESIGN\_NAME) "adc\_edge\_detect\_circuit"

set filename $::env(DESIGN\_DIR)/$::env(PDK)\_$::env(STD\_CELL\_LIBRARY)\_config.tcl

if { [file exists $filename] == 1} {

source $filename

}

set ::env(VERILOG\_FILES) "/foss/designs/SKY130\_SAR-ADC/verilog/adc\_edge\_detect\_circuit/adc\_edge\_detect\_circuit.v"

# set ::env(CLOCK\_PERIOD) "10.000"

set ::env(CLOCK\_PORT) "clk"

set ::env(CLOCK\_NET) $::env(CLOCK\_PORT)

# Floorplanning

set ::env(FP\_SIZING) "relative"

set ::env(FP\_ASPECT\_RATIO) {1}

set ::env(FP\_CORE\_UTIL) {75}

set ::env(FP\_PDN\_VPITCH) {100}

set ::env(FP\_PDN\_HPITCH) {100}

set ::env(FP\_PIN\_ORDER\_CFG) $::env(DESIGN\_DIR)/pin\_order.cfg

# Placement

set ::env(PL\_TARGET\_DENSITY) {0.80}

# set ::env(PL\_TIME\_DRIVEN) {0}

set ::env(PL\_RESIZER\_TIMING\_OPTIMIZATIONS) {0}

set ::env(PL\_RESIZER\_DESIGN\_OPTIMIZATIONS) {0}

#Clocktreesynthesis - unused

# set ::env(CLOCK\_TREE\_SYNTH) {0}

# set ::env(FILL\_INSERTION) {0}

# Router

set ::env(GLB\_RESIZER\_TIMING\_OPTIMIZATIONS) {0}

set ::env(CELL\_PAD) {0}