

# Digital VLSI & SoC Design planning using Open-Lane/sky130 Open-Source EDA Tools

Name: Swetha Manupati

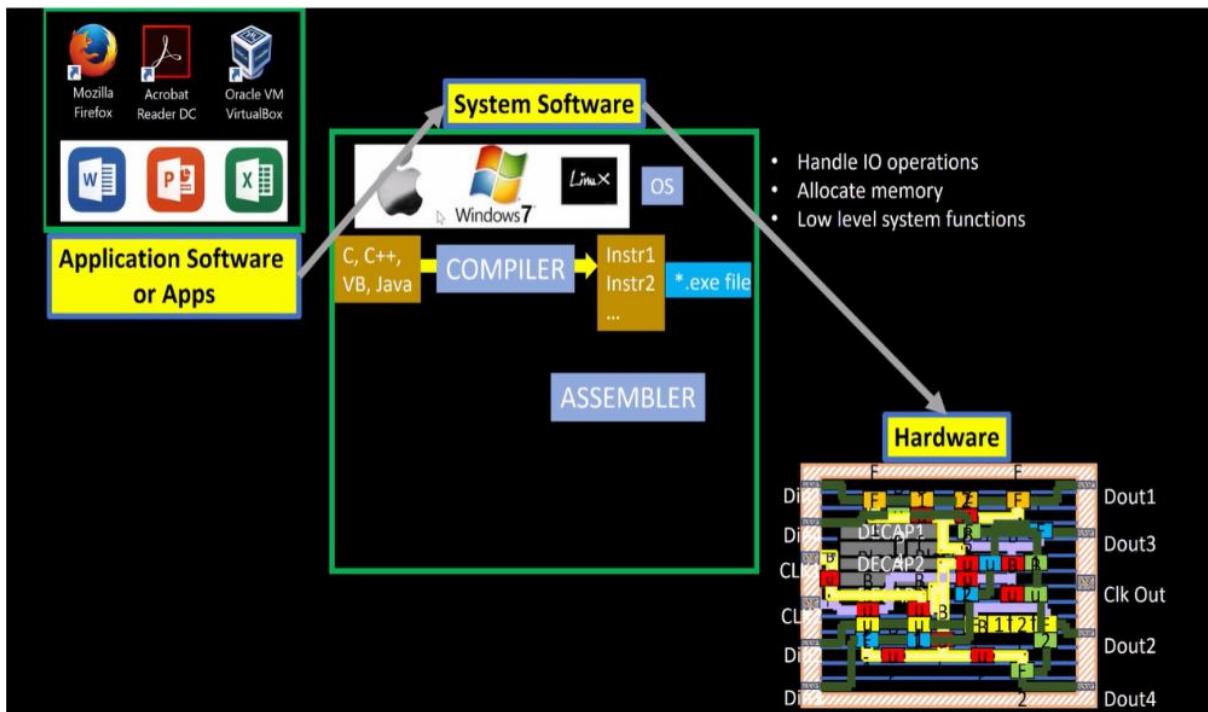
Cohort: VLSI Physical design Course\_Nov22\_Nov30(2025)

## Table of Contents

<b>Day 1- Inception of Open-Source EDA tools, openlane &amp; Sky130 PDK .....</b>	<b>1</b>
How to talk to computers.....	4
Soc design and openlane.....	5
Get familiar to open-Source EDA tools	
<b>Day 2- Good Floorplanning Considerations.....</b>	<b>9</b>
Chip Floorplanning and Consideration.....	12
Library building and placement .....	15
Cell design and characterization flows.....	18
Cell design parameters .....	20
<b>Day 3- Design of library cell using Magic Layout &amp; Ngspice characterization .....</b>	<b>22</b>
Labs for CMOS inverter ngpsice simulations .....	24
Inception of layout CMOS fabrication process .....	26
<b>Day 4- Timing Analysis .....</b>	<b>27</b>
Timing Analysis with ideal clock openSTA.....	28
<b>Day 5- Final Step for RTL to GDS using TrionRoute and OpenSTA.....</b>	<b>30</b>

## Day1: How to talk to computers?

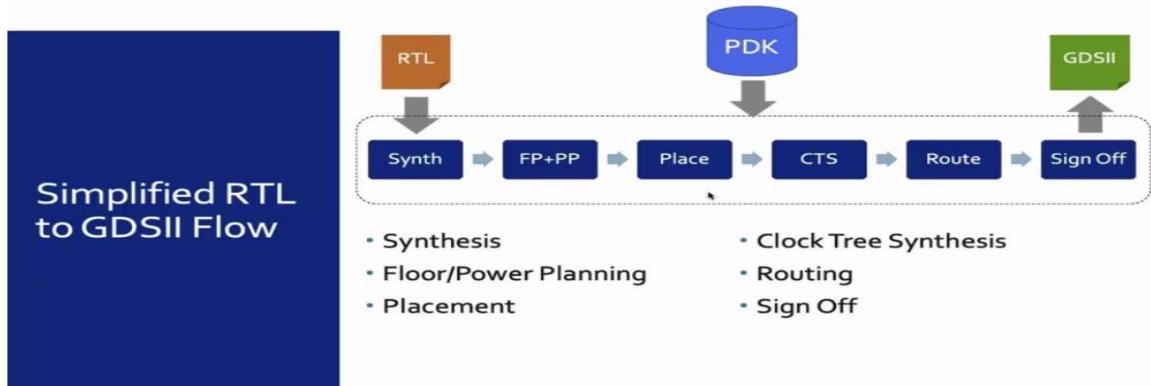
Talking to a computer means giving some set of instructions in a form that hardware understands and whose hardware is based on RISC-V core. If you want to run a software/ Programming code like C,C++ or java it can be easily understandable by humans, but the hardware cannot understand it. So the compiler converts high level code to RISC-V instructions which processor can decode.



## SoC Design and Openlane

A **System-on-Chip (SoC)** integrates multiple components of a computer or electronic system into a single silicon die. A typical SoC contains, Processor, Memory Subsystem, Peripherals, Bus architecture. To design the ASIC, it requires ASIC flow. The objective of this ASIC flow is to convert RTL to GDSII for the final layout. The whole ASIC flow is automated using EDA tools which is a software automatically does place & Route.

### RTL to GDSII flow

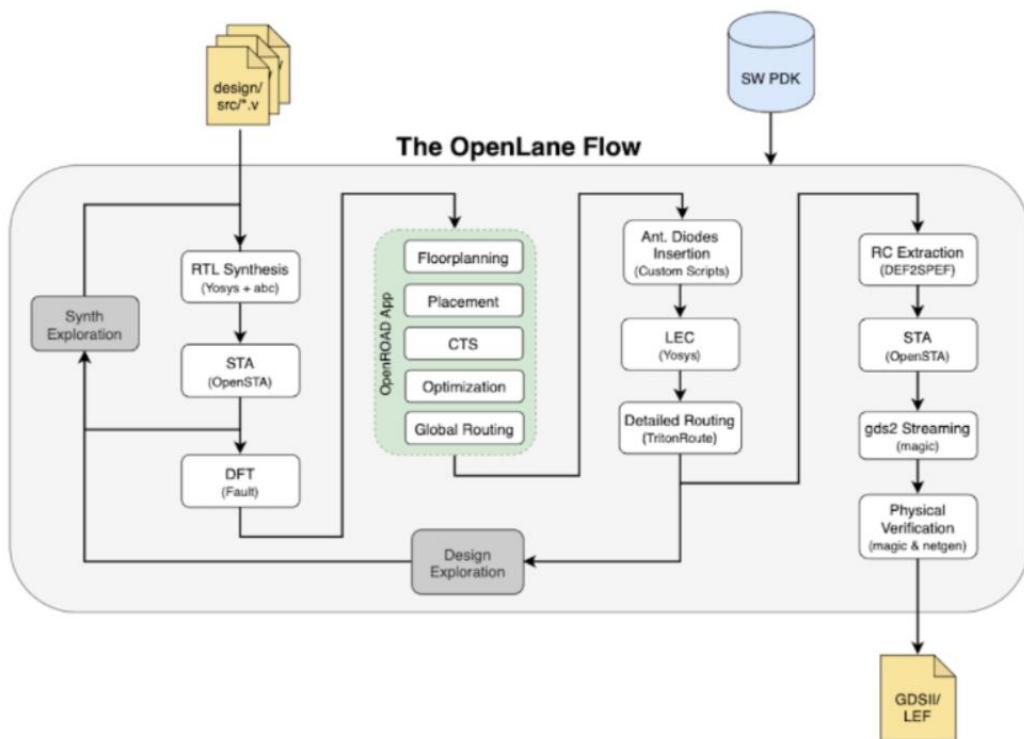


1. **Synthesis:** The code written by the RTL designers cannot be directly Physically implemented on the chip, so synthesis is a process where the RTL code is converted into gate-level netlist using standard cell libraries which can be easily understandable by EDA tools
2. **Floor Planning:** It's the process of creating the area of chip using aspect ratio and utilization factor, placement of macro cells, creating power grids and adding decap cells/ physical cells

3. Placement: Placement is process where the standard cells are placed in optimal positions basically in assigned floor plan and ensures to reduce congestion. Placement is done in two stages
  - Global Placement: Cells are placed at optimal positions, there might be overlaps
  - Detailed Placement: Cells are placed at legal positions with no overlaps
4. Routing: Routing is making connections between the standard cells. It done in two phases one is
  - Global Routing
  - Detailed Routing
5. Sign off: Sign off is the last stage of Physical design flow where it verifies layout satisfies the rules for manufacturing process like DRC, LVS and Timing verifications.

## OpenLane ASIC flow

The openLane ASIC flow is as follows:



Opensource EDA tools that supports each stage of ASIC flow is represented in the following table

Task	Tools
RTL Synthesis	Yosys, abc
Floor plan	Init_fp, ioplacer
Placement	Magic
CTS	Trion CTS

Routing	Trion Route and FastRoute
Static Timing Analysis	OpenSTA

## Get familiar to Open-source EDA

Before mastering opensource EDA tools, the first and foremost thing is to get familiar with basic linux commands to communicate with the open source software on Linux environment. Some of the basic commands are mentioned below

- cd file name: takes to the directory “filename”
- ls -ltr: list the files/ directories
- cd ..: exits from the directory
- pwd: prints the current directory
- cp: copy files to the active directory
- Command --help: shows the use of the command
- clear: clears the screen

In our design flow, we use the **Sky130\_fd\_sc\_hd** PDK variant. This naming convention describes important details about the process and the type of standard-cell library included.

- **sky130** → Refers to the SkyWater **130 nm technology node**.
- **fd** → Indicates the **foundry designation**, meaning this PDK is released by **SkyWater Foundry**.
- **sc** → Stands for **standard cell library**, which provides the logic cells required for digital design.
- **hd** → Represents the **high-density** variant of the standard-cell library. This library prioritizes smaller cell area, enabling higher logic density at the cost of drive strength.

The **Sky130\_fd\_sc\_hd** PDK contains a wide range of technology files used at different stages of the digital ASIC flow. Some important file types are Verilog files, lef lifes, tech files, lib files. The **Sky130\_fd\_sc\_hd** library provides all the necessary functional, timing, and physical data required for RTL-to-GDSII implementation.

## Design preparation steps to invoke openlane

To invoke open lane, first the terminal should be opened in openlane working directory. Once the path is created the openlane can be invoked by the following commands

- Docker
- ./flow.tcl -interactive (it's a script that specifies the details to interact with the openlane flow)

- Package require openlane 0.9

```

vsdworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal Mon 05:15
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/... vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/... vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/...
File Edit View Search Terminal Tabs Help
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/... vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/... vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/...
drwxr-xr-x 10 vsduser docker 4096 Jun 29 2021 openlane_old
drwxr-xr-x 11 vsduser docker 4096 Nov 27 18:51 openlane
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/... vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/... vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/...
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/... docker run -it
docke: Error response from daemon: failed to create task for container: failed to create shim task: OCI runtime create failed: runc create failed: unable to start container process: exec: "run": executable file not found in $PATH: unknown.
[0001] error waiting for container:
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane$ docker
bash-4.2$ ls -lrt
total 140
drwxr-xr-x 15 1000 997 4096 Jun 29 2021 scripts
-rw-r--r-- 1 1000 997 28787 Jun 29 2021 run_designs.py
-rw-r--r-- 1 1000 997 7898 Jun 29 2021 report_generation_wrapper.py
drwxr-xr-x 3 1000 997 4896 Jun 29 2021 regression_results
drwxr-xr-x 2 1000 997 2048 Jun 29 2021 tcl
drwxr-xr-x 5 1000 997 4996 Jun 29 2021 docs
drwxr-xr-x 5 1000 997 4996 Jun 29 2021 docker_build
drwxr-xr-x 44 1000 997 4896 Jun 29 2021 design
drwxr-xr-x 2 1000 997 2048 Jun 29 2021 configuration
drwxr-xr-x 1 1000 997 5514 Jun 29 2021 conf.py
-rwxr-xr-x 1 1000 997 960 Jun 29 2021 clean_runs.tcl
-rw-r--r-- 1 1000 997 2048 Jun 29 2021 config.tcl
-rw-r--r-- 1 1000 997 7274 Jun 29 2021 Makefile
-rw-r--r-- 1 1000 997 11350 Jun 29 2021 LICENSE
-rw-r--r-- 1 1000 997 1285 Jun 29 2021 CONTRIBUTING.md
-rw-r--r-- 1 1000 997 4996 Jun 29 2021 default.vcvc
drwxr-xr-x 6 1000 1000 4996 Nov 28 14:31 vdsdcelldesign
bash-4.2$ ./flow/tcl_interactive
bash: ./flow/tcl_interactive: No such file or directory
bash-4.2$ ./flow/tcl -interactive
bash: ./flow/tcl: No such file or directory
bash-4.2$ ./flow.tcl -interactive
[INFO]: Version: v0.21
[INFO]: Running interactively

```

The design folder of openlane consists of 30~40 designs. As this VSD Physical design course is based on Soc design of picorv32a, we will be doing the whole design flow of this CPU core that implements RISC-V instruction set. Once the openlane is invoked the design can be initialized into the tool by following command

- prep -design picorv32a

```

vsdworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal Mon 10:13
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/... vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/... vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/...
File Edit View Search Terminal Tabs Help
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/... vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/... vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/...
bash-4.2$ ./flow/tcl -interactive
bash: ./flow/tcl: No such file or directory
bash-4.2$ ./flow.tcl -interactive
[INFO]:
[INFO]: Version: v0.21
[INFO]: Running interactively
[INFO]: Package require openlane 0.9
B.9
% prep -design picorv32a
[INFO]: Using design configuration at /openlane_flow/designs/picorv32a/config.tcl
[INFO]: Copying configurations from /openlane_flow/designs/picorv32a/config.tcl
[INFO]: PDKs root directory: /home/vsduser/Desktop/work/tools/openlane_working_dir/pdk
[INFO]: PDK: sky130
[INFO]: Extracting metal layers to /home/vsduser/Desktop/work/tools/openlane_working_dir/pdk/sky130A
[INFO]: Standard Cell Library: sky130_fd_sc_hd
[INFO]: Sourcing Configurations from /openlane_flow/designs/picorv32a/config.tcl
[INFO]: Copying configurations from /openlane_flow/designs/picorv32a/runs/01_12_04-42
[INFO]: Preparing LEF files
[INFO]: Extracting the number of available metal layers is 6
[INFO]: The number of available metal layers is 6
[INFO]: The available metal layers are 1st mett1 met2 met3 met2 met3 met4 met5
[INFO]: Merging LEF files...
merglelef.py : Merging LEF files
sky130_ef_sc_hd.lef: MACROS matched found: 0
sky130_ef_sc_hd.lef: SITEs matched found: 437
sky130_ef_sc_hd.fill_12_left.SITEs matched found: 0
sky130_ef_sc_hd.fill_12_right.SITEs matched found: 0
sky130_ef_sc_hd.decap_12_left.SITEs matched found: 0
sky130_ef_sc_hd_decap_12_left.MACROS matched found: 1
sky130_ef_sc_hd_takedown_2_left.MACROS matched found: 0
sky130_ef_sc_hd_takedown_2_left.SITEs matched found: 1
merglelef.py : Merging LEF files complete
[INFO]: Trimming library...
[INFO]: Generating LEF files lists...
[INFO]: Storing config into config.tcl ...
[INFO]: Preparation complete
t_rdm_synthesis

```

Once the command is initialized it prepares all the files required for running the openlane flow.

RUNS folder is created in picorv32a folder, where all the runs for the design will be stored after each stage is executed

```

vsdworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal Mon 10:13 •
vsduser@vsdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane
File Edit View Search Terminal Tabs Help
vsduser@vsdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane vsduser@vsdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane vsduser@vsdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane
bash-4.2$ ./flow/tcl -interactive
bash: ./flow/tcl: No such file or directory
bash-4.2$ ./flow/tcl -interactive
[INFO]:
[INFO]: Version: v0.21
[INFO]: Running interactively
% package require openlane 0.9
0.9
# Prep -design picorv32a
? [INFO]: Using design configuration at /openLANE_flow/designs/picorv32a/config.tcl
[INFO]: Sourcing Configurations from /openLANE_flow/designs/picorv32a/config.tcl
[INFO]: PDK root directory: /home/vsduser/Desktop/work/tools/openlane_working_dir/pdks
[INFO]: PDK: sky130A
[INFO]: Standard Cell Library: sky130_fd_sc_hd
[INFO]: Sourcing Configurations from /openLANE_flow/designs/picorv32a/config.tcl
[INFO]: Current run directory is /openLANE_flow/designs/picorv32a/runs/01-12_04-42
[INFO]: Preparing LEF Files...
[INFO]: Extracting the number of available metal layers from /home/vsduser/Desktop/work/tools/openlane_working_dir/pdks/sky130A/lts.ref/sky130_fd_sc_hd/techlef/sky130_fd_sc_hd.lef
[INFO]: The available metal layers are l1 met1 met2 met3 met4 met5
[INFO]: Merging LEF Files...
mergelef.py : Merging LEFs
sky130_ef_sc_hd.lef: MACROS matched found: 0
sky130_ef_sc_hd_1ff_12.lef: SITEs matched found: 0
sky130_ef_sc_hd_1ff_12.lef: MACROS matched found: 0
sky130_ef_sc_hd_1decap_12.lef: SITEs matched found: 0
sky130_ef_sc_hd_1decap_12.lef: MACROS matched found: 0
sky130_ef_sc_hd_1decap_2_1.lef: SITEs matched found: 0
sky130_ef_sc_hd_1decap_2_1.lef: MACROS matched found: 0
mergelef.py : Merging LEFs complete
[INFO]: Trimming liberty...
[INFO]: Generating ABC file...
[INFO]: Writing config into config.tcl ...
[INFO]: Preparation complete
% run_synth

```

To run synthesis in the openlane interactive terminal, the command is

- `run_synth`

As seen earlier to run synthesis openlane uses yosys and ABC to convert RTL to gatelevel netlist

```

vsdworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal Mon 10:15 •
vsduser@vsdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane
File Edit View Search Terminal Tabs Help
vsduser@vsdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane vsduser@vsdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane vsduser@vsdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane
29. Executing Verilog Backend.
Dumping module `picorv32a'.
Warnings: 307 unique messages, 307 total
End of script. Logfile hash: d272dd05b3, CPU: User 37.69s system 1.46s, MEM: 96.46 MB peak
yosys -q -p "script.sh" -t 3.1 -f PIC_CG
Time spent: 53% 2x abc (42 sec) - 15% 3x opt_expr (9 sec), ...
[INFO]: Changing netlist from @ to /openLANE_flow/designs/picorv32a/runs/01-12_04-42/results/synthesis/picorv32a.synthesis.v
[INFO]: Running Static Timing Analysis...
[INFO]: current step Index: 2
OpenSTA 2.3.0-38ba0d89ab Copyright (c) 2019, Parallax Software, Inc.
License GPLv3+: GNU GPL version 3 <http://gnu.org/licenses/gpl.html>
This is free software, and you are free to change and redistribute it
under certain conditions; type 'show copying' for details.
This program comes with ABSOLUTELY NO WARRANTY.
Warning: /home/vsduser/Desktop/work/tools/openlane_working_dir/pdks/sky130A/lts.ref/sky130_fd_sc_hd/lib/sky130_fd_sc_hd_ff_n40C_iv95.lib line 31, default_operating_condition ff_n40C_iv95 not found.
Warning: /home/vsduser/Desktop/work/tools/openlane_working_dir/pdks/sky130A/lts.ref/sky130_fd_sc_hd/lib/sky130_fd_sc_hd_ss_100C_iv60.lib line 32, default_operating_condition ss_100C_iv60 not found.
create_clock [get_ports $::env(CLOCK_PORT)] -name $::env(CLOCK_PORT) -period $::env(CLOCK_PERIOD)
set_input_delay_value [expr $::env(CLOCK_PERIOD) * $::env(TD_PCT)]
set_output_delay_value [expr $::env(CLOCK_PERIOD) * $::env(TD_PCT)]
set_max_fanout $::env(MAX_FANOUT) [current_design]
set_max_fanout $::env(MAX_FANOUT) [get_clocks $::env(CLOCK_PORT)]
set_max_fanout $::env(MAX_FANOUT) [get_ports $::env(CLOCK_PORT)]
set_rst_idx [lsearch [all_inputs] {get_port resetn}]
set_all_inputs_wo_clk [lreplace [all_inputs] $clk_idx $clk_idx]
set_all_inputs_wo_clk_rst [lreplace $all_inputs_wo_clk $rst_idx $rst_idx]
set_all_inputs_wo_clk_rst [lreplace $all_inputs_wo_clk_rst $clk_idx $clk_idx]
set_all_inputs_wo_clk_rst [lreplace $all_inputs_wo_clk_rst $rst_idx $rst_idx]
set_input_delay $input_delay -clock [get_clocks $::env(CLOCK_PORT)] $all_inputs_wo_clk_rst
set_input_delay 0.0 -clock [get_clocks $::env(CLOCK_PORT)] [resetn]
set_output_delay $output_delay -clock [get_clocks $::env(CLOCK_PORT)] [all_outputs]
# TODO: this is as parameter
set_driving_cell $cell $::env(SYNTH_DRIVING_CELL) -pin $::env(SYNTH_DRIVING_CELL_PIN) [all_inputs]
set_ap_load [expr ($::env(SYNTH_CAP_LOAD) / 1000.0)]
puts "\[INFO\]: Setting load to: Scap_load"
[INFO]: Setting load to: 0.01765
set_load Scap_load [all_outputs]
tns -759.46
tns -24.00
[INFO]: Synthesis was successful

```

## Steps to characterize synthesis

To analyze the results for synthesis, go to runs folder results/synthesis you can find gate level netlist file generated with .v extension

```

28. Printing statistics.

==== plcorv32a ====
Number of wires: 14596
Number of wire bits: 14978
Number of public wires: 1565
Number of public wire bits: 1947
Number of memory bits: 0
Number of memory cells: 0
Number of cell bits: 0
Number of cells: 14876
sky130_fd_sc_hd_a2110_2 1613
sky130_fd_sc_hd_a2110_2 35
sky130_fd_sc_hd_a2110_2 60
sky130_fd_sc_hd_a2110_2 149
sky130_fd_sc_hd_a2110_2 9
sky130_fd_sc_hd_a2110_2 57
sky130_fd_sc_hd_a2110_2 244
sky130_fd_sc_hd_a2210_2 86
sky130_fd_sc_hd_a2210_2 103
sky130_fd_sc_hd_a22b20_2 1748
sky130_fd_sc_hd_a2bb20_2 81
sky130_fd_sc_hd_a3110_2 2
sky130_fd_sc_hd_a3110_2 49
sky130_fd_sc_hd_a3110_2 7
sky130_fd_sc_hd_a330_2 46
sky130_fd_sc_hd_a410_2 1
sky130_fd_sc_hd_and2_2 157
sky130_fd_sc_hd_and3_2 58
sky130_fd_sc_hd_and4_2 345
sky130_fd_sc_hd_buf_1 1
sky130_fd_sc_hd_buf_2 8
sky130_fd_sc_hd_cronh_1 1
sky130_fd_sc_hd_dfftp_2 1613
sky130_fd_sc_hd_latch_2 615
sky130_fd_sc_hd_mux2_1 1224
sky130_fd_sc_hd_mux2_2 2
sky130_fd_sc_hd_mux4_1 221
sky130_fd_sc_hd_mand2_2 78
sky130_fd_sc_hd_mand2_2 524
sky130_fd_sc_hd_nor2b_2 1
l-yosys 4.stat.rpt

```

To calculate the flip-flop ratio, it can be calculated by using below formula:

$$\text{flipflop ratio} = \frac{\text{Number of flipflops}}{\text{Total number of cells}}$$

From the above generated data, the cell count is 14876, and the flip flop is 1613. So the flop ratio is 10.8%.

Before the synthesis, we observed that results folder is empty, now after the synthesis process the ABC mapped the results to synthesis.

```

/* Generated by Yosys 0.9+3621 (git sha1 84e9fa7, gcc 8.3.1 -fPIC -Os) */

module plicorv32a(clk, resetn, trap, mem_valid, mem_instr, mem_ready, mem_addr, mem_wdata, mem_wstrb, mem_rdata, mem_la_read, mem_la_write, mem_la_addr, mem_la_wdata, mem_la_wstrb, pcpt_valid, pcpt_insn, pcpt_wstrb, pcpt_rd, pcpt_wt, pcpt_ready, irq, eot, trace_vald, trace_data);
    wire _0000;
    wire _0001;
    wire _0002;
    wire _0003;
    wire _0004;
    wire _0005;
    wire _0006;
    wire _0007;
    wire _0008;
    wire _0009;
    wire _0010;
    wire _0011;
    wire _0012;
    wire _0013;
    wire _0014;
    wire _0015;
    wire _0016;
    wire _0017;
    wire _0018;
    wire _0019;
    wire _0020;
    wire _0021;
    wire _0022;
    wire _0023;
    wire _0024;
    wire _0025;
    wire _0026;
    wire _0027;
    wire _0028;
    wire _0029;
    wire _0030;
    wire _0031;
    wire _0032;
    wire _0033;
    wire _0034;
    wire _0035;
    wire _0036;
    wire _0037;
    wire _0038;
    wire _0039;
endmodule

```

And from the above report we can see that the actual synthesis is done.

## Day 2 Floor Planning Considerations

Calculation of Area of core and Die:

The first thing to be considered is defining the aspect ratio and utilization factor for core and die. The generated netlist will have the standard cells and electrical connectivity between the gates. To ease the calculation, we consider each cell in square representation, and calculate the area of core and die.

$$\text{Utilization Factor} = \frac{\text{Area occupied by netlist}}{\text{Total area of core}}$$

$$\text{Area} = \frac{\text{Height}}{\text{Width}}$$

If the utilization factor is 1 it represents that 100% of the chip is used and there is no space to add extra cells such as buffers/inverters. And if the aspect ratio is ratio is 1 its signifies the chip is in square, other than 1 it considered to be a rectangular chip

Pre-placed cells:

Pre-placed cells, often referred to as **macros**, are large blocks whose internal structure cannot be modified.

Examples: SRAMs, ROMs, PLLs, analog IPs.

These macros are positioned before standard cell placement. They are placed in locations that optimize connectivity and reduce routing congestion, especially when they interact frequently with surrounding logic.

Decoupling capacitors:

Pre-placed cells should be surrounded by decoupling caps, as macros is huge and consists of millions of gate inside it, the supply voltage from VDD to macro might be insufficient due to the voltage drops because of long wire lengths from supply voltage to the pre-placed cells. This can lead to non-functionalities, and the output detected by logic might be incorrect because of default noise margins which can lead to metastability. So, to avoid these a huge decap is placed nearby the preplaced cells such that enough amount of supply voltage can be drawn by the cells.

Power planning:

Power planning involves creating a robust Power Distribution Network (PDN) to ensure that all standard cells and macros receive a stable and uniform supply of VDD and GND. During this stage, power rings are built around the core boundary, and horizontal and vertical power straps are inserted across the chip to distribute current evenly. These straps connect to the power rails inside each standard-cell row, enabling reliable current flow during switching activity.

Pin placement:

The space between the core and die is where usually the I/O pins are placed. We can also optimize the spacing between the pins whether they should be spaced equidistant or unequal by modifying the reports at floorplan stage.

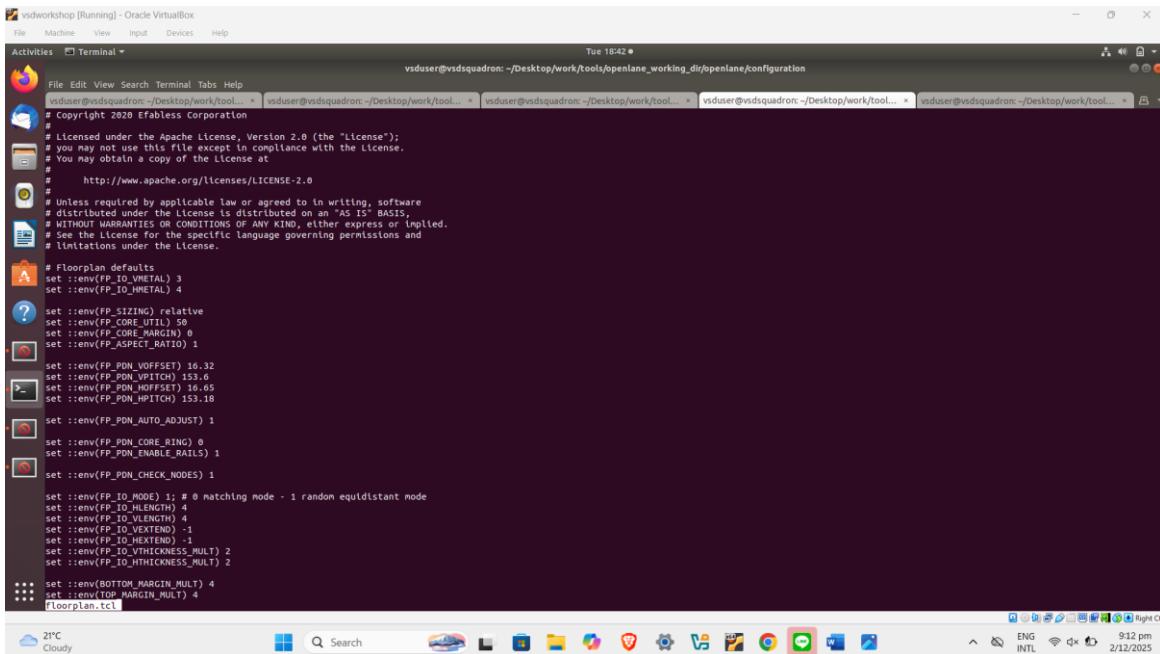
## Floorplan run on Openlane & View in Magic

The priority order of files in increasing order

1. Floorplan.tcl

2. config.tcl

3. sky130A\_sky130\_fd\_sc\_hd\_config.tcl



```

vsduser@vsdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane/configuration
Tue 1842

# Copyright 2020 Efabless Corporation
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#     http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

# Floorplan defaults
set :env(FP_IO_VMETAL) 3
set :env(FP_IO_HMETAL) 4

set :env(FP_SIZING) relative
set :env(FP_CORE_UTIL) 50
set :env(FP_CORE_MARGIN) 0
set :env(FP_ASPECT_RATIO) 1

set :env(FP_PDN_VOFFSET) 16.32
set :env(FP_PDN_HOFFSET) 16.65
set :env(FP_PDN_HPISTM) 153.18

set :env(FP_PDN_AUTO_ADJUST) 1

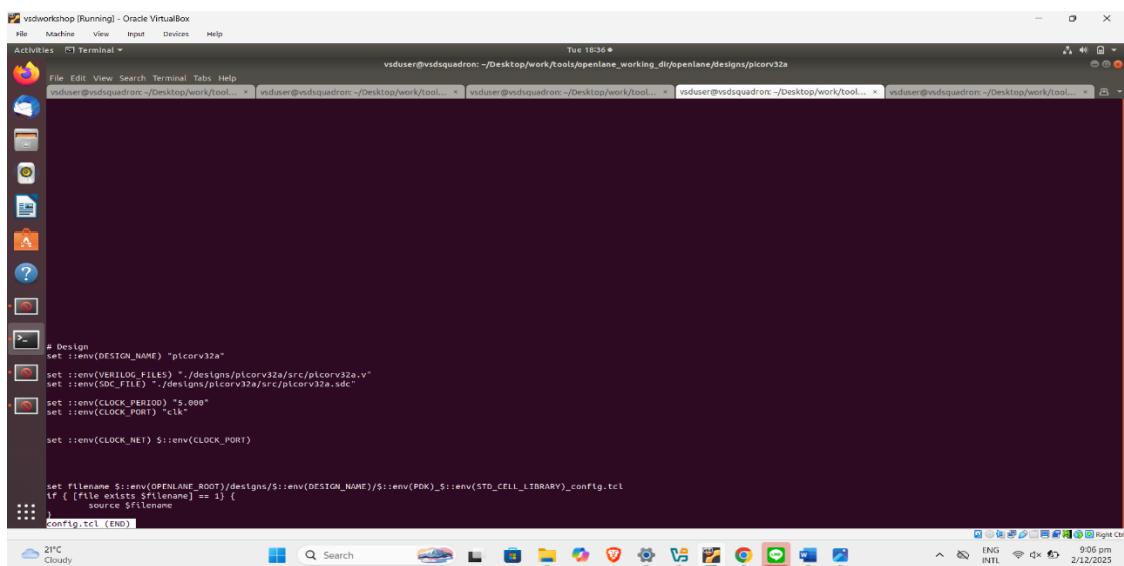
set :env(FP_PDN_CORE_RING) 0
set :env(FP_PDN_ENABLE_RAILS) 1
set :env(FP_PDN_CHECK_NODES) 1

set :env(FP_IO_MODE) 1; # @ matching mode - 1 random equidistant mode
set :env(FP_IO_LENGTH) 4
set :env(FP_IO_VLENGTH) 4
set :env(FP_IO_XTEND) 1
set :env(FP_IO_XEXTEND) -1
set :env(FP_IO_HXTEND) -1
set :env(FP_IO_VTHICKNESS_MULT) 2
set :env(FP_IO_HTHICKNESS_MULT) 2

set :env(BOTTOM_MARGIN_MULT) 4
set :env(TOP_MARGIN_MULT) 4
Floorplan.tcl

```

Here it should be noted that the vertical and horizontal metal layer values will be precedent to the system defaults that specified in the files



```

vsduser@vsdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane/designs/picorv32a
Tue 1836

# Design
set :env(DESIGN_NAME) "picorv32a"

set ::env(VPRLOG_FILES) "./designs/picorv32a/src/picorv32a.v"
set ::env(SDC_FILE) "./designs/picorv32a/src/picorv32a.sdc"

set ::env(CLOCK_PERIOD) "5.000"
set ::env(CLOCK_PORT) "clk"

set ::env(CLOCK_NET) $::env(CLOCK_PORT)

set filename $::env(OPENLANE_ROOT)/designs/$::env(DESIGN_NAME)/$::env(PDK)_$::env(STD_CELL_LIBRARY)_config.tcl
if {[file exists $filename] == 1} {
    source $filename
}

config.tcl (END)

```

```

# SCL Configs
set ::env(GLB_RT_ADJUSTMENT) 0.1
set ::env(SYNTH_MAX_FANOUT) 6
set ::env(CLOCK_PERIOD) "24.73"
set ::env(FP_CORE_UTIL) 35
set ::env(PL_TARGET_DENSITY) [ expr $(::env(FP_CORE_UTIL)+5) / 100.0 ]
::: sky130A_sky130_fd_sc_hd_config.tcl (END)

```

## Some of the switches for floorplan

```

| 'CLK_BUFFER_FANOUT' | Fanout of clock tree buffers. <br> (Default: '16') |
| 'ROOT_CLK_BUFFER' | Root clock buffer of the clock tree. <br> (Default: 'sky130_fd_sc_hd_clkbuf_16') |
| 'CLK_BUFFER' | Clock buffer used for inner nodes of the clock tree. <br> (Default: 'sky130_fd_sc_hd_clkbuf_4') |
| 'CLK_BUFFER_INPUT' | Input pin of the clock tree buffer. <br> (Default: 'A') |
| 'CLK_BUFFER_OUTPUT' | Output pin of the clock tree buffer. <br> (Default: 'B') |
| 'DIE_AREA' | The die area to be used for Static Timing Analysis. <br> (Default: '$::env(OPENLANE_ROOT)/scripts/base.sdc') |
| 'VERILOG_INCLUDE_DIRS' | Specifies the include directories. <br> (Default: '') |
| 'SYNTH_FLAT_TOP' | Specifies whether or not the top level should be flattened during elaboration. 1 = True, 0 = False <br> Default: '0' |
| 'IO_PCT' | Specifies the percentage of the clock period used in the input/output delays. Ranges from 0 to 1.0. <br> (Default: '0.2') |

### Floorplanning
| Variable | Description |
|-----|-----|
| 'FP_CORE_UTIL' | The core utilization percentage. <br> (Default: '50 percent') |
| 'FP_ASPECT_RATIO' | The aspect ratio of the core area. <br> (Default: '1:1') |
| 'FP_SIZING' | Whether to use relative sizing by making use of 'FP_CORE_UTIL' or absolute one using 'DIE_AREA'. <br> (Default: 'relative' - accepts 'absolute' as well) |
| 'DIE_AREA' | Specific die area to be used in floorplanning. Specified as a 4-corner rectangle. Units in mm <br> (Default: unset) |
| 'FP_IO_HMETAL' | The metal layer on which to place the IO pins horizontally (top and bottom of the die). <br> (Default: '4') |
| 'FP_IO_VMETAL' | The metal layer on which to place the IO pins vertically (sides of the die) <br> (Default: '3') |
| 'FP_PLACEMENT_MODE' | Decides the placement mode for the power distribution network. <br> (Default: 'random') |
| 'FP_WELTAP_CELL' | The name of the welltap cell during welltap insertion. |
| 'FP_ENDCAP_CELL' | The name of the endcap cell during endcap insertion. |
| 'FP_PDN_VOFFSET' | The offset of the vertical power stripes on the metal layer 4 in the power distribution network <br> (Default: '16.32') |
| 'FP_PDN_VPITCH' | The pitch of the vertical power stripes on the metal layer 4 in the power distribution network <br> (Default: '153.6') |
| 'FP_PDN_HOFFSET' | The offset of the horizontal power stripes on the metal layer 4 in the power distribution network <br> (Default: '16.32') |
| 'FP_PDN_HPITCH' | The pitch of the horizontal power stripes on the metal layer 5 in the power distribution network <br> (Default: '153.18') |
| 'FP_PDN_AUTO_ADJUST' | Decides whether or not the flow should attempt to re-adjust the power grid, in order for it to fit inside the core area of the design, if needed. <br> 1=enabled, 0 =disabled (Default: '1') |
| 'FP_TAPCELL_DIST' | The horizontal distance between two tapcell columns <br> (Default: '14') |
| 'FP_TAPCELL_MARGIN' | The margin between the tapcell columns and the specified units <br> (Default: '-1' Disabled) |
| 'FP_IO_HEXTEND' | Extends the horizontal IO pins outside of the die by the specified units <br> (Default: '-1' Disabled) |
| 'FP_IO_VLENGTH' | The length of the vertical IOs in microns. <br> (Default: '4') |
| 'FP_IO_HLENGTH' | The length of the horizontal IOs in microns. <br> (Default: '4') |
| 'FP_IO_VTHICKNESS_MULT' | A multiplier for vertical pin thickness. Base thickness is the pins layer minwidth <br> (Default: '2') |
| 'FP_PDN_VTHICKNESS_MULT' | A multiplier for the vertical power distribution network thickness layer <br> (Default: '2') |
| 'BOTTOM_MARGIN_MULT' | The core margin, in multiples of site heights, from the bottom boundary <br> (Default: '2') |
| 'TOP_MARGIN_MULT' | The core margin, in multiples of site heights, from the top boundary <br> (Default: '4') |
| 'LEFT_MARGIN_MULT' | The core margin, in multiples of site widths, from the left boundary. <br> (Default: '12') |
| 'RIGHT_MARGIN_MULT' | The core margin, in multiples of site widths, from the right boundary. <br> (Default: '12') |
| 'FP_PDN_ENABLE_BALLS' | Enables adding the corner array balls design. more details in the control variables in the pdk configurations documentation. 0=Disable 1=Enable. <br> (Default: '0') |
| 'FP_PDN_CHECK_NODES' | Enables checking for unconnected nodes in the power grid. 0=Disable 1=Enable. <br> (Default: '1') |
| 'FP_HORIZONTAL_HALO' | Sets the horizontal halo around the tap and decap cells. The value provided is in microns. <br> Default: '10' |
| 'FP_VERTICAL_HALO' | Sets the vertical halo around the tap and decap cells. The value provided is in microns. <br> Default: set to the value of 'FP_HORIZONTAL_HALO' |

```

FP\_CORE\_UTIL – floorplan core utilization

FP\_ASPECT\_RATIO- floorplan aspect ratio

FP\_CORE\_MARGIN- core to die margin area

FP\_IO\_MODE- defines pin configurations

FP\_CORE\_VMETAL- vertical metal layer

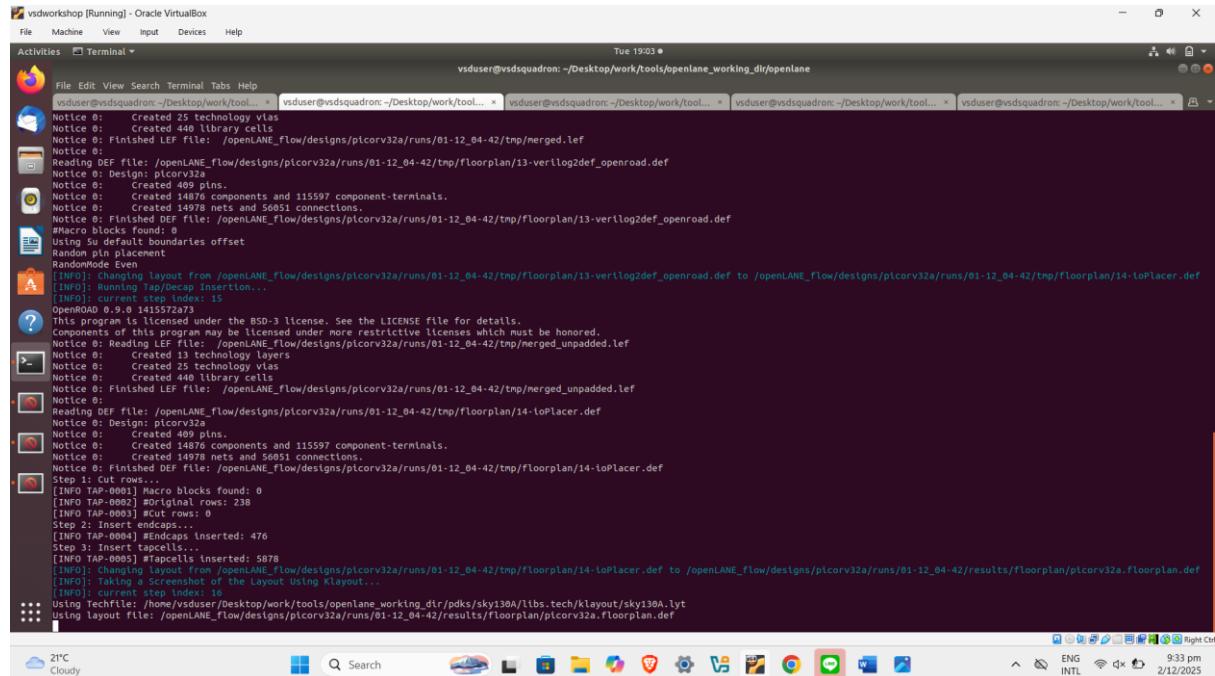
FP\_CORE\_HMETAL- Horizontal metal layer

# Steps to run floorplan

To run floorplan on open lane, input the following command

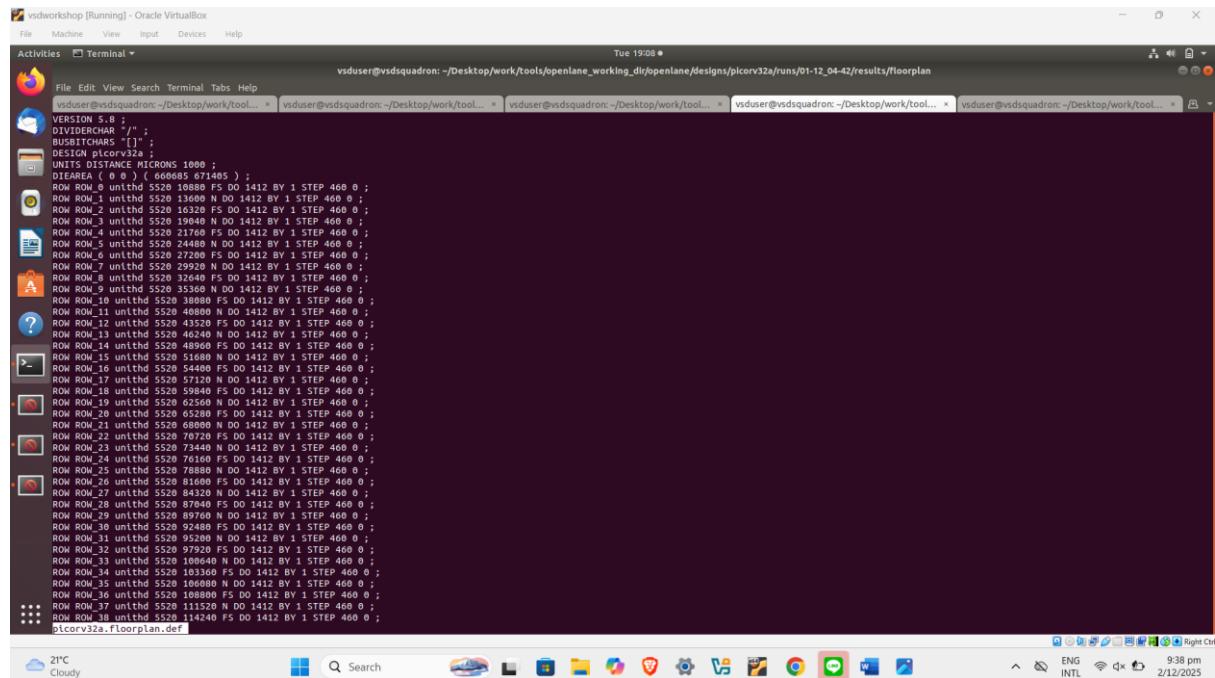
run\_floorplan

Post floorplan, .def file will be created within the directory results/floorplan. We can review the floorplan files by checking floorplan.tcl



```
vsdworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal Tue 19:03
vsduser@vdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane
File Edit View Search Terminal Tabs Help
vsduser@vdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane
Notice 0: Created 25 technology vias
Notice 0: Created 440 library cells
Notice 0: Finished LEF file: /openLANE_flow/designs/picorv32a/runs/01-12_04-42/tmp/merged.lef
Notice 0:
Reading DEF file: /openLANE_flow/designs/picorv32a/runs/01-12_04-42/tmp/floorplan/13-verilog2def_openroad.def
Notice 0: Design: picorv32a
Notice 0: Created 14976 components and 115597 component-terminals.
Notice 0: Created 14978 nets and 56851 connections.
Notice 0: Finished DEF file: /openLANE_flow/designs/picorv32a/runs/01-12_04-42/tmp/floorplan/13-verilog2def_openroad.def
#Macro blocks found: 0
Using Si default boundaries offset
Random Si placement
RandomMode Even
[INFO]: Changing layout from /openLANE_flow/designs/picorv32a/runs/01-12_04-42/tmp/floorplan/13-verilog2def_openroad.def to /openLANE_flow/designs/picorv32a/runs/01-12_04-42/tmp/floorplan/14-ioPlacer.def
[INFO]: Running Tap/Decap Insertion...
[DRC]: 0 DRC errors found; 15 DRC warnings
[DRC]: 0 DRC errors found; 15 DRC warnings
This program is licensed under the BSD-3 license. See the LICENSE file for details.
Components of this program may be licensed under more restrictive licenses which must be honored.
Notice 0: Reading LEF file: /openLANE_flow/designs/picorv32a/runs/01-12_04-42/tmp/merged_unpadded.lef
Notice 0: Created 11 technology layers
Notice 0: Created 25 technology vias
Notice 0: Created 440 library cells
Notice 0: Finished LEF file: /openLANE_flow/designs/picorv32a/runs/01-12_04-42/tmp/merged_unpadded.lef
Notice 0:
Reading DEF file: /openLANE_flow/designs/picorv32a/runs/01-12_04-42/tmp/floorplan/14-ioPlacer.def
Notice 0: Design: picorv32a
Notice 0: Created 489 pins.
Notice 0: Created 14976 components and 115597 component-terminals.
Notice 0: Created 14978 nets and 56851 connections.
Notice 0: Finished DEF file: /openLANE_flow/designs/picorv32a/runs/01-12_04-42/tmp/floorplan/14-ioPlacer.def
[INFO]: Cut rows: 0
[INFO TAP-0001] Macro blocks found: 0
[INFO TAP-0002] #Original rows: 238
[INFO TAP-0003] #Cut rows: 0
Step 2: Insert endcaps...
[DRC]: 0 DRC errors found; 15 DRC warnings
[DRC]: 0 DRC errors found; 15 DRC warnings
[INFO TAP-0005] #Tapecells inserted: 476
[INFO]: Taking a Screenshot of the Layout Using Klayout...
[INFO]: current step index: 16
Using Techfile: /home/vsduser/Desktop/work/tools/openlane_working_dir/pdk/sky130A/libs.tech/klayout/sky130A.lyt
Using Layout file: /openLANE_flow/designs/picorv32a/runs/01-12_04-42/results/floorplan/picorv32a.floorplan.def
Tue 19:03
File Machine View Input Devices Help
Activities Terminal 21°C Cloudy Search 9:33 pm ENG INTL 2/12/2025 Right Ctrl
```

The below is how the floorplan.def looks like

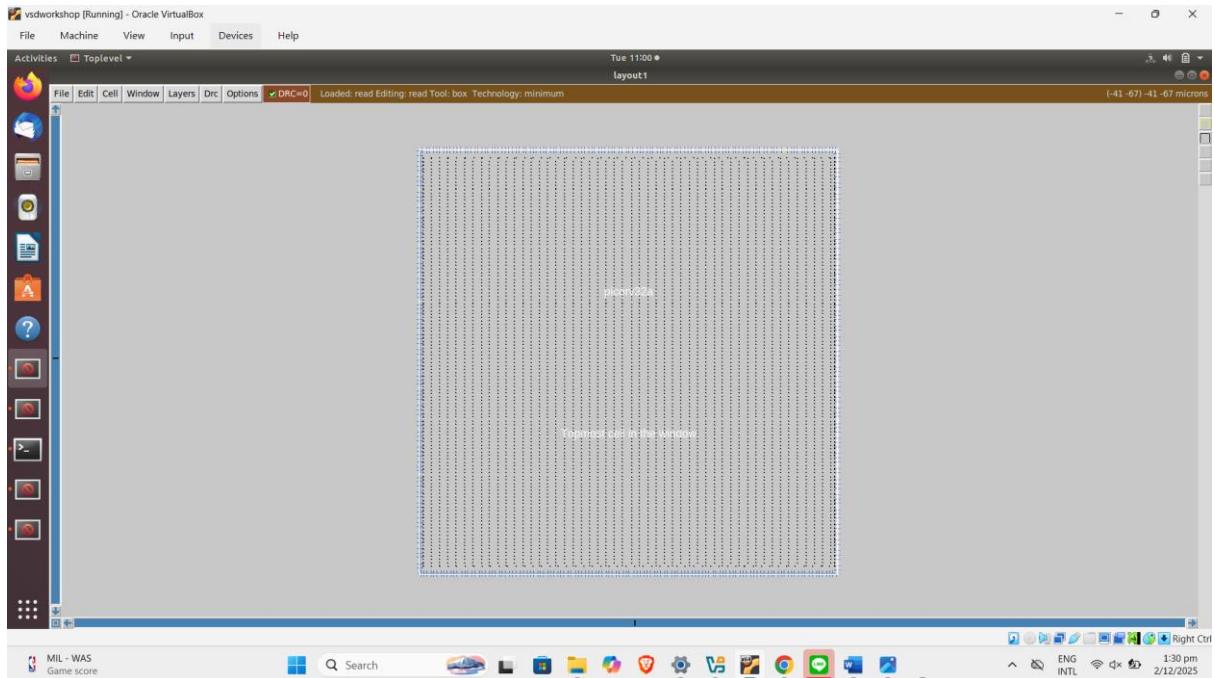


```
vsdworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal Tue 19:08
vsduser@vdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane/designs/picorv32a/runs/01-12_04-42/results/floorplan
VERSION 5.8 ; DIVIDERCHAR '/';
DESIGN picorv32a ;
UNITS DISTANCE MICRONS 1000 ;
DIEAREA ( 0 0 ) ( 660685 671485 ) ;
ROW ROW_0 unithd 5520 10880 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_1 unithd 5520 13600 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_2 unithd 5520 16320 S DO 1412 BY 1 STEP 460 0 ;
ROW ROW_3 unithd 5520 19040 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_4 unithd 5520 21760 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_5 unithd 5520 24480 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_6 unithd 5520 27200 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_7 unithd 5520 30000 S DO 1412 BY 1 STEP 460 0 ;
ROW ROW_8 unithd 5520 32640 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_9 unithd 5520 35360 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_10 unithd 5520 38080 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_11 unithd 5520 40800 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_12 unithd 5520 43520 S DO 1412 BY 1 STEP 460 0 ;
ROW ROW_13 unithd 5520 46240 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_14 unithd 5520 48960 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_15 unithd 5520 51680 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_16 unithd 5520 54400 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_17 unithd 5520 57120 S DO 1412 BY 1 STEP 460 0 ;
ROW ROW_18 unithd 5520 59840 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_19 unithd 5520 62560 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_20 unithd 5520 65280 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_21 unithd 5520 68000 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_22 unithd 5520 70720 S DO 1412 BY 1 STEP 460 0 ;
ROW ROW_23 unithd 5520 73440 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_24 unithd 5520 76160 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_25 unithd 5520 78880 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_26 unithd 5520 81600 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_27 unithd 5520 84320 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_28 unithd 5520 87040 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_29 unithd 5520 89760 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_30 unithd 5520 92480 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_31 unithd 5520 95200 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_32 unithd 5520 97920 S DO 1412 BY 1 STEP 460 0 ;
ROW ROW_33 unithd 5520 100640 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_34 unithd 5520 103360 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_35 unithd 5520 106080 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_36 unithd 5520 108800 FS DO 1412 BY 1 STEP 460 0 ;
ROW ROW_37 unithd 5520 111520 N DO 1412 BY 1 STEP 460 0 ;
ROW ROW_38 unithd 5520 114240 FS DO 1412 BY 1 STEP 460 0 ;
picorv32a.floorplan.def
Tue 19:08
File Machine View Input Devices Help
Activities Terminal 21°C Cloudy Search 9:38 pm ENG INTL 2/12/2025 Right Ctrl
```

If we see the file consists of die area information. And it's mentioned the units in microns which means 1 micron is equal to 1000 data base units. So, if we divide the die area by 1000

we can get the dimensions of the chip. So, the width of the chip is 660.68 and height is 671.45

To see the layout of the floorplan, you should invoke magic tool by adding command magic - T to the results/floorplan directory. And then after pressing enter Magic file will open.

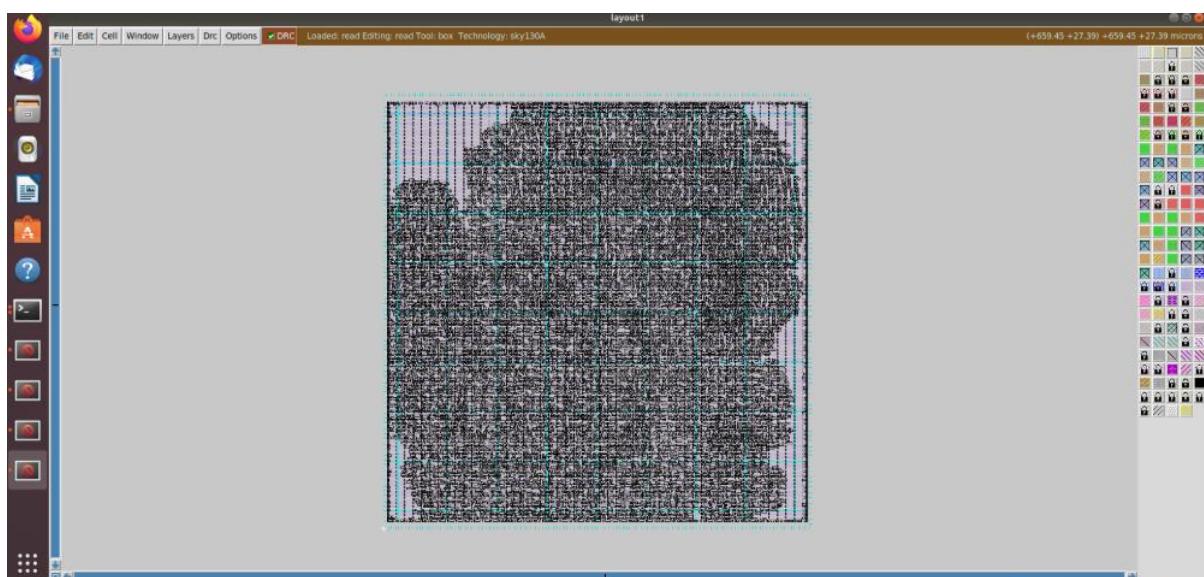


## Placement run on openlane and view in Magic

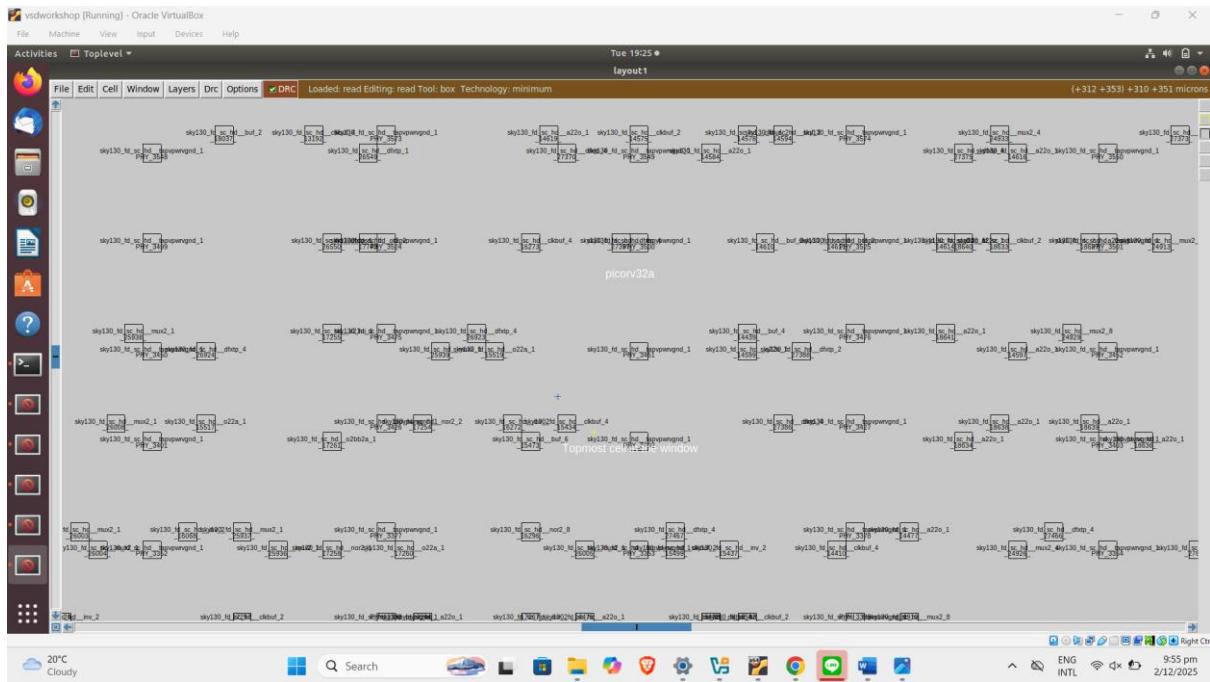
To run placement, using replace the following command is used in the flow

run\_placement

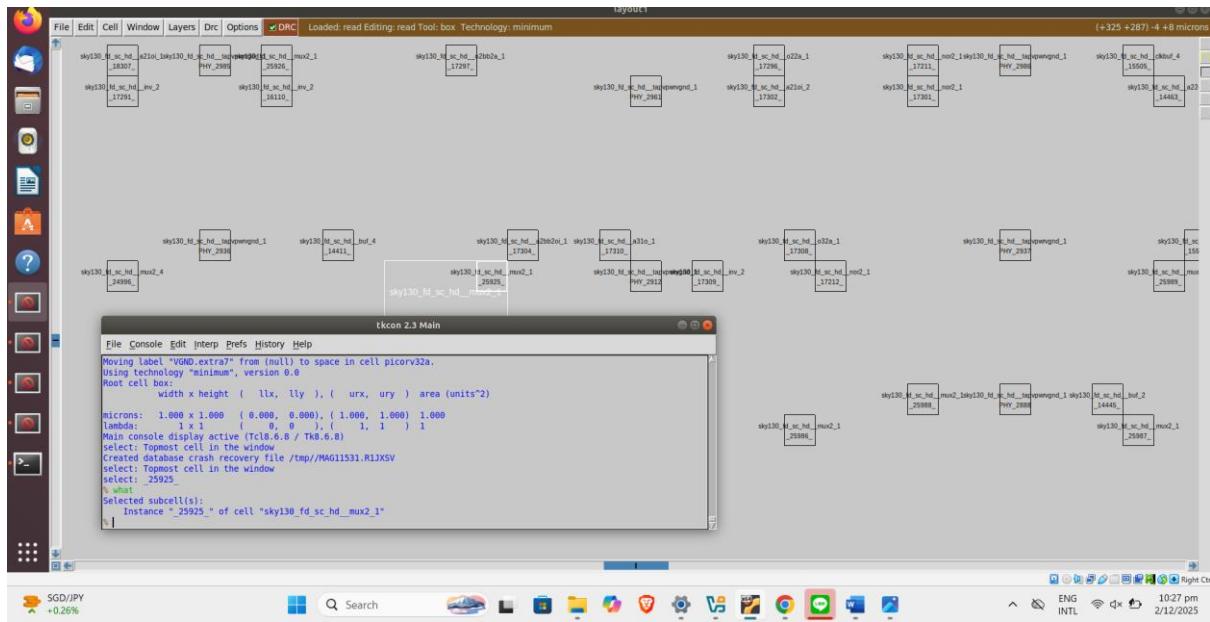
Post placement, the results will be stored in results/placement directory and the layout can be viewed by magic.



Zoomed in gives the standard placement view of the design



If you click on the cell instance and type What on tkcon window, it gives the detailed information of the cell.



## Standard cell design flow

A standard cell design flow includes the following steps

- Read the models and tech files

- Analyse the behaviour of the cell
- Read the extracted spice netlist
- Read the sub circuits
- Attach power sources
- Provide necessary output capacitance

## Timing Parameter definitions for standard cell Analysis

**Rise\_time:** Rise time is the time required for a signal to transition from a low voltage level to a high voltage level. It is typically measured between **10% and 90% of the supply voltage (VDD)**. Rise time indicates how quickly a signal can move from logic ‘0’ to logic ‘1’.

**Fall\_time:** Fall time is the time required for a signal to transition from a high voltage level to a low voltage level. It is typically measured between **90% and 10% of VDD**. Fall time indicates how quickly a signal can move from logic ‘1’ to logic ‘0’

## Day 3: Design cell library

### Spice deck creation & Simulation'

A spice deck includes the following information

- Netlist description
- Model description
- Component connectivity
- Assign component values
- Define nodes
- Name nodes

### CMOS inverter Switching threshold (Vm)

It can be defined as the point on transfer characteristics plot where Vin equals vout. At this point bot PMOS and NMOS will be in ON state leading to rise in leakage current

## CMOS fabrication process

### 1. Substrate selection

Selecting the substrate material.

### 2. Well Formation

Create N-well and P-well regions where PMOS and NMOS transistors will be placed.

### 3. Isolation

Form isolation regions (oxide trenches) to separate transistors on the wafer.

### 4. Gate Formation

Grow a thin gate oxide and deposit/pattern polysilicon to create the transistor gate.

## 5. Source/Drain Formation

Dope the regions on both sides of the gate to form source and drain terminals.

## 6. Contact Creation

Open small holes (contacts) in the insulating layer so metal can touch the transistor terminals.

## 7. Metallization

Deposit and pattern metal layers to connect different devices and form interconnects.

## 8. Passivation

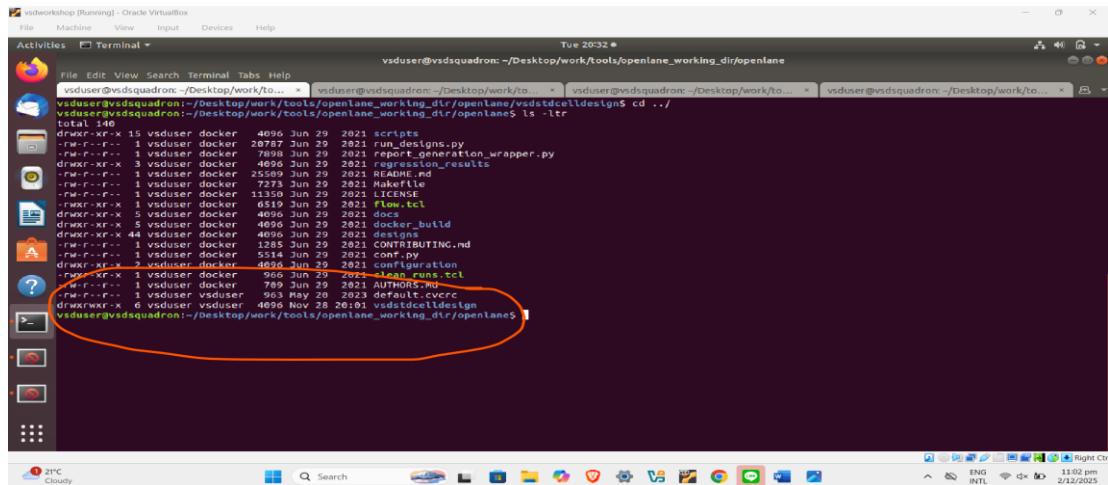
Add a protective top layer and open windows for bonding pads used in packaging.

## Lab steps to git clone vsdstd cell design

To clone the standard cell, copy the file path from github repository and paste it in the openlane directory after initiating the command git clone. The file path to clone the std cell is as follows

```
git clone https://github.com/nickson-jose/vsdstdcelldesign
```

This creates vsdstd cell design in openlane directory



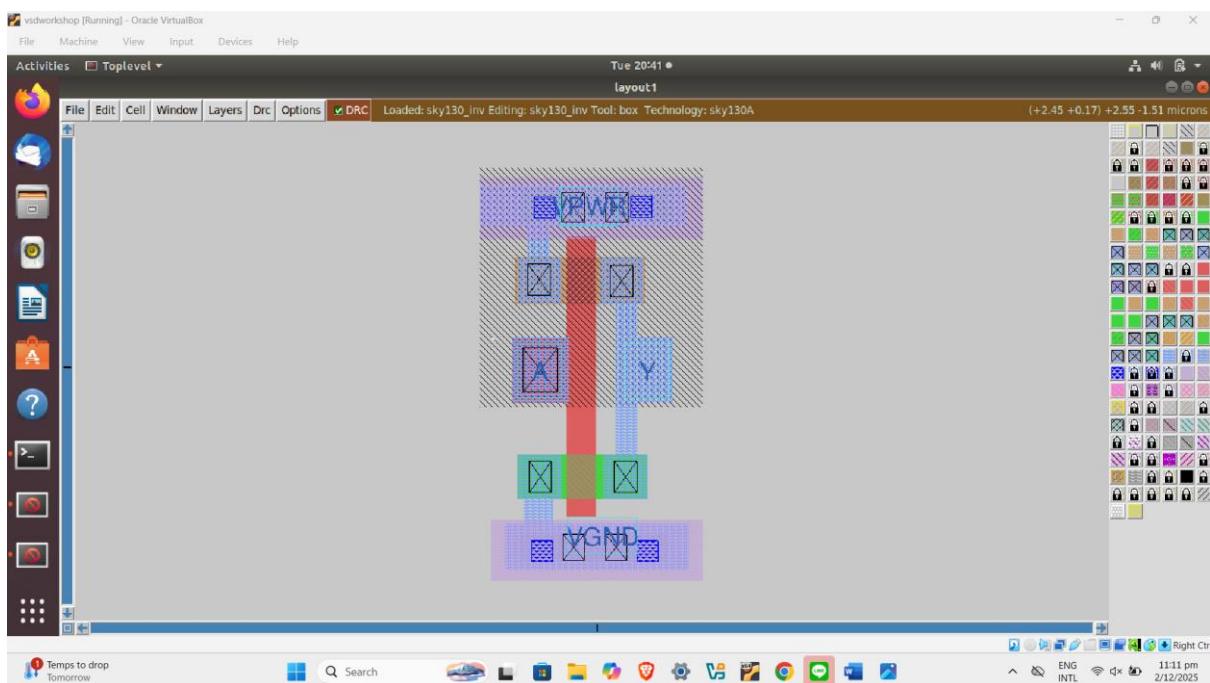
Now, if we go into the vsdstd cell directory, you can see .mag file, libs file and so on. Now let's view .mag file see what layers are used to build the inverter. So, to view the file we need technology file. So, we will copy the file from given address. Now the file is copied to vsdstd cell design folder

```

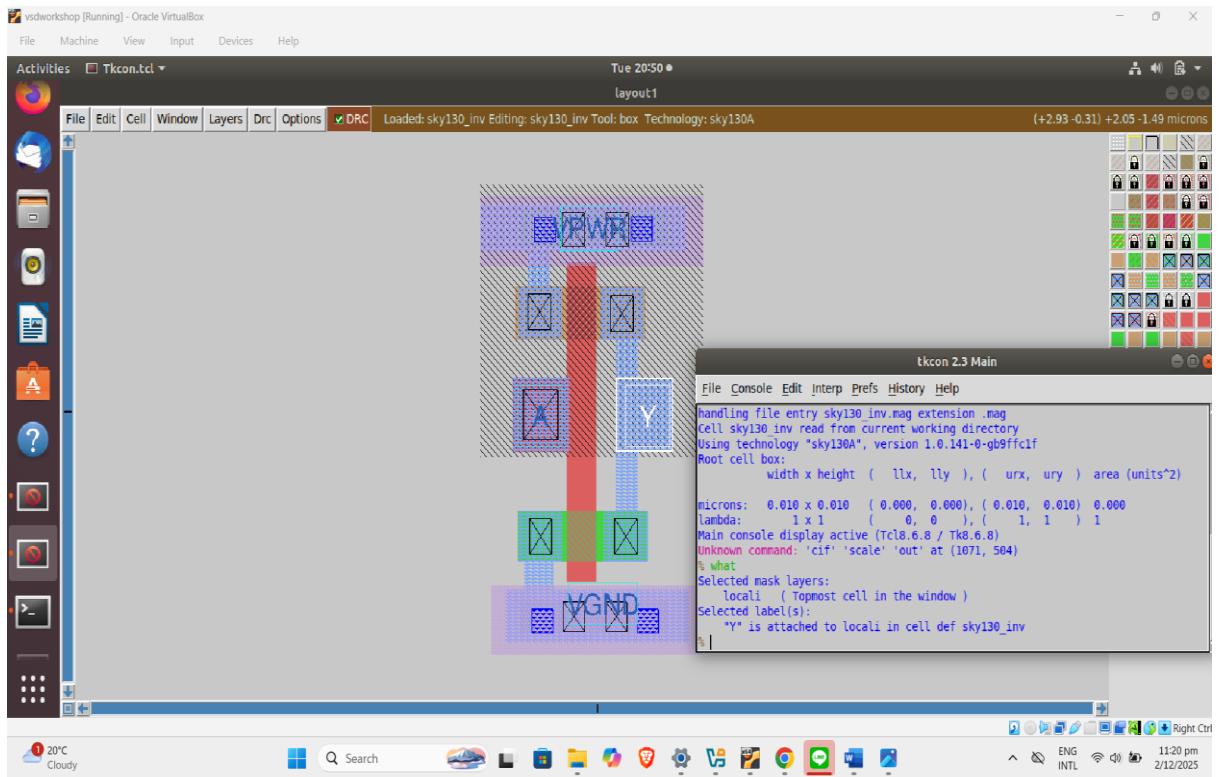
vsdworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal Tue 20:38
vsduser@vdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane$ ls -ltr
total 140
drwxr-xr-x 1 vsduser docker 4096 Jun 29 2021 scripts
-rw-r--r-- 1 vsduser docker 20787 Jun 29 2021 run_designs.py
-rw-r--r-- 1 vsduser docker 7898 Jun 29 2021 report_generation_wrapper.py
drwxr-xr-x 3 vsduser docker 4096 Jun 29 2021 regression_results
-rw-r--r-- 1 vsduser docker 25589 Jun 29 2021 README.md
-rw-r--r-- 1 vsduser docker 7273 Jun 29 2021 Makefile
-rw-r--r-- 1 vsduser docker 11350 Jun 29 2021 LICENSE
-rwrxr-x 1 vsduser docker 6519 Jun 29 2021 flow.tcl
drwxr-xr-x 5 vsduser docker 4096 Jun 29 2021 docs
drwxr-xr-x 5 vsduser docker 4096 Jun 29 2021 docker_build
drwxr-xr-x 44 vsduser docker 4096 Jun 29 2021 designs
-rw-r--r-- 1 vsduser docker 1285 Jun 29 2021 CONTRIBUTING.md
-rw-r--r-- 1 vsduser docker 5514 Jun 29 2021 conf.py
drwxr-xr-x 2 vsduser docker 4096 Jun 29 2021 configuration
-rwrxr-x 1 vsduser docker 966 Jun 29 2021 clean_runs.tcl
-rw-r--r-- 1 vsduser docker 789 Jun 29 2021 AUTHORS.md
-rw-r--r-- 1 vsduser vdsuser 963 May 28 2023 default.cvcrc
drwxrwxr-x 6 vsduser vdsuser 4096 Nov 28 20:01 vsdstdceldesign
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane$ cd vsdstdceldesign
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ ls -ltr
total 196
-rw-rw-r-- 1 vsduser vdsuser 13525 Nov 27 18:52 README.md
-rw-rw-r-- 1 vsduser vdsuser 11357 Nov 27 18:52 LICENSE
drwxrwxr-x 2 vsduser vdsuser 4096 Nov 27 18:52 Images
drwxrwxr-x 2 vsduser vdsuser 4096 Nov 27 18:52 extras
drwxrwxr-x 2 vsduser vdsuser 4096 Nov 27 18:52 libs
-rw-rw-r-- 1 vsduser vdsuser 2716 Nov 27 18:52 sky130_inv.mag
-rwrxr-x 1 vsduser vdsuser 136710 Nov 27 19:01 sky130A.tech
-rw-rw-r-- 1 vsduser vdsuser 1365 Nov 27 19:41 sky130_inv.ext
-rw-rw-r-- 1 vsduser vdsuser 562 Nov 28 18:29 sky130_inv.spice
-rw-rw-r-- 1 vsduser vdsuser 2779 Nov 28 19:58 sky130_vsdlnv.mag
-rw-rw-r-- 1 vsduser vdsuser 1395 Nov 28 20:01 sky130_vsdlnv.lef
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ 

```

Now, to see the layout we invoke magic tool and the layout of the inverter looks something like this

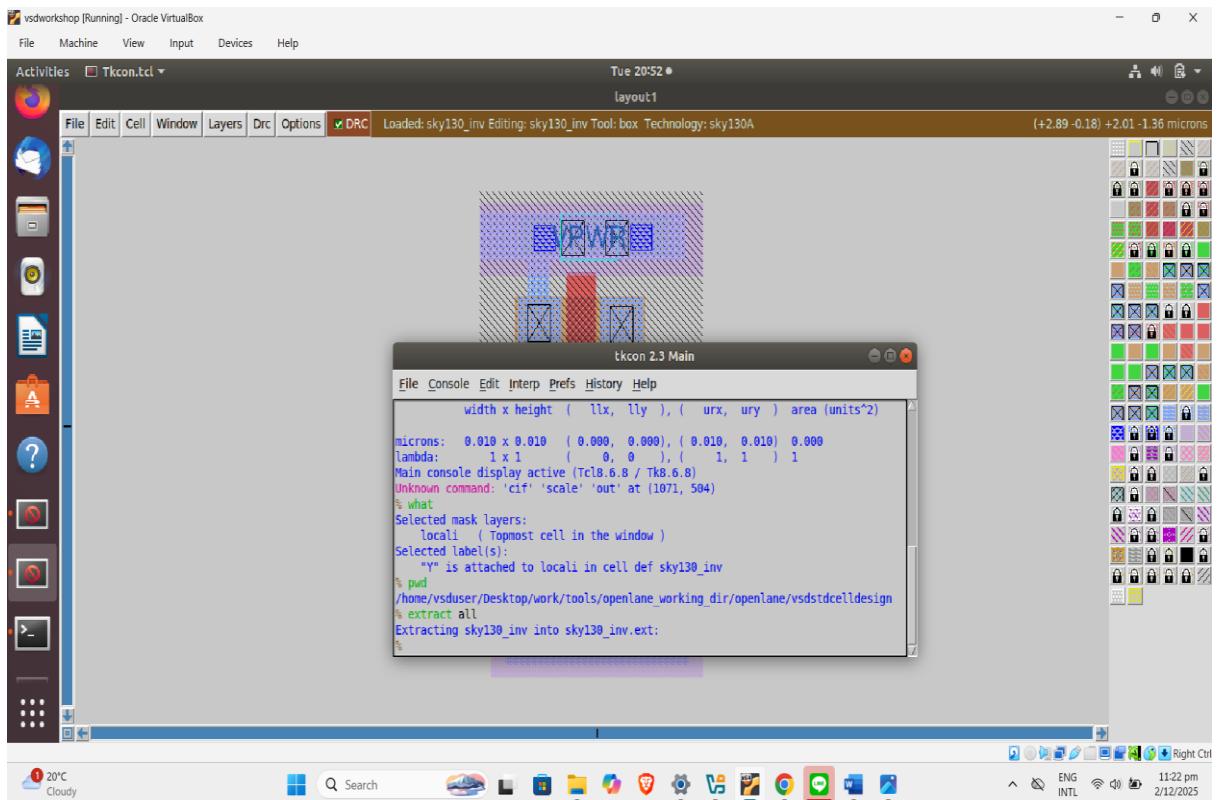


If we want to know the information about the layers, click on the area and press and type what in tckon window, similarly we can check for output terminal also, but double pressing "S"

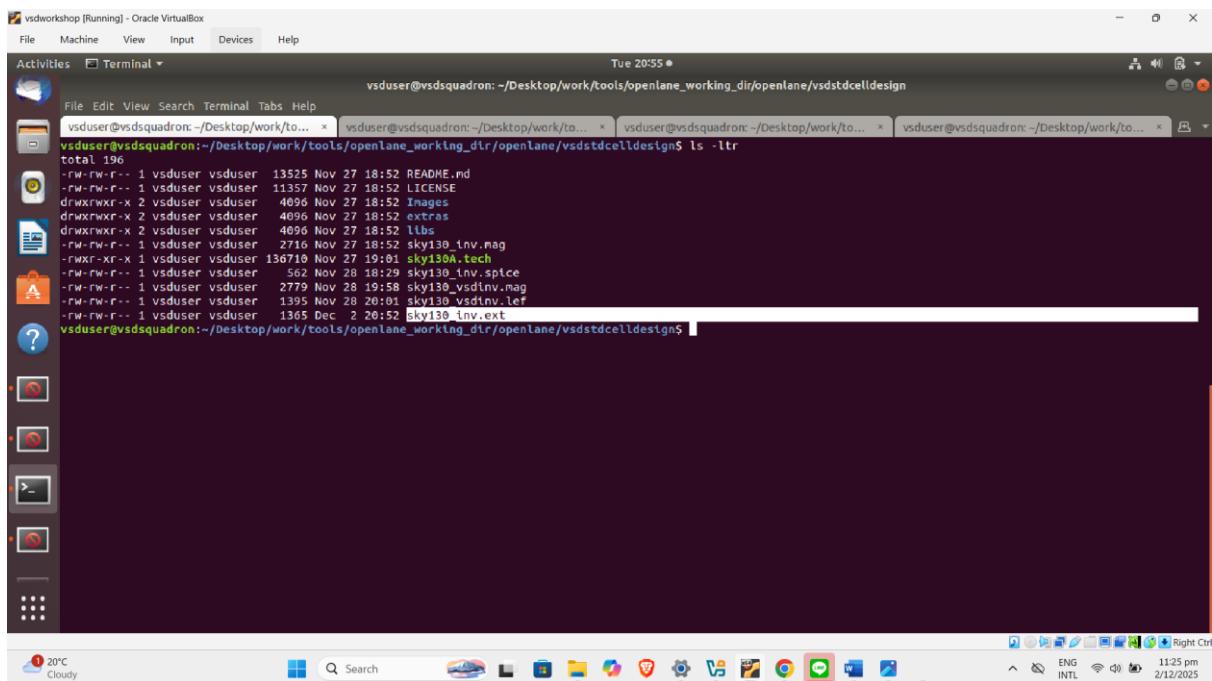


## Lab steps to create std cell layout and extract spice netlist

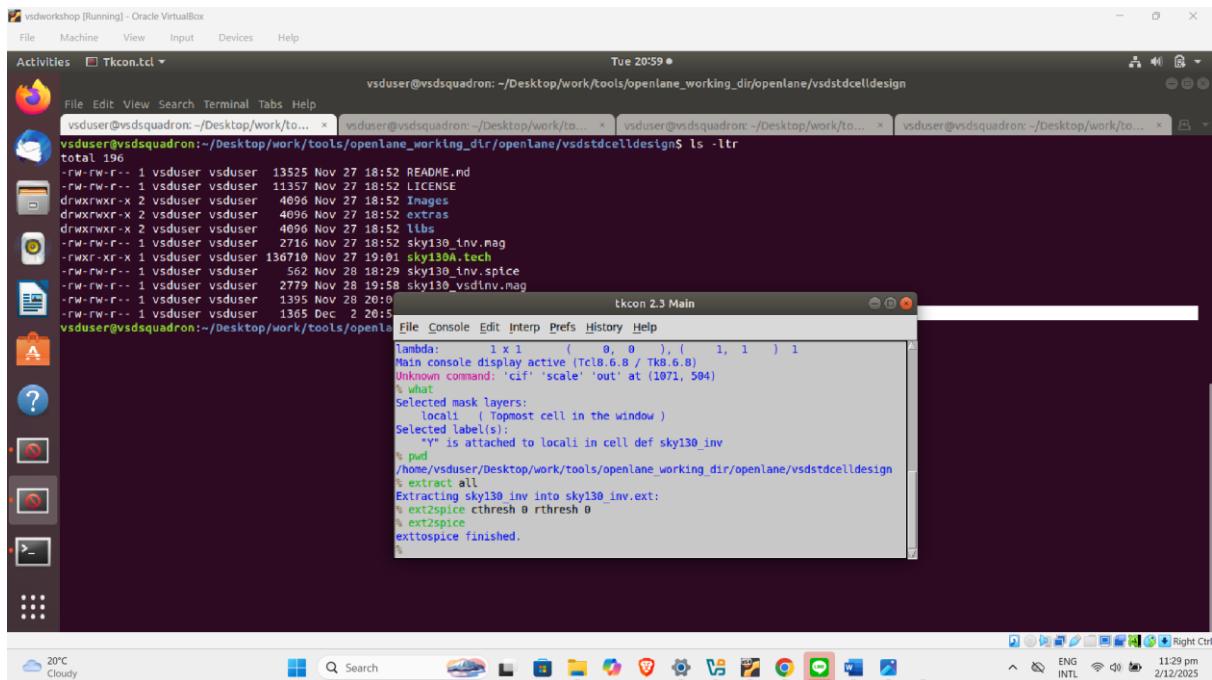
To extract spice netlist, input command extract all to tkcon window. It extracts the spice netlist



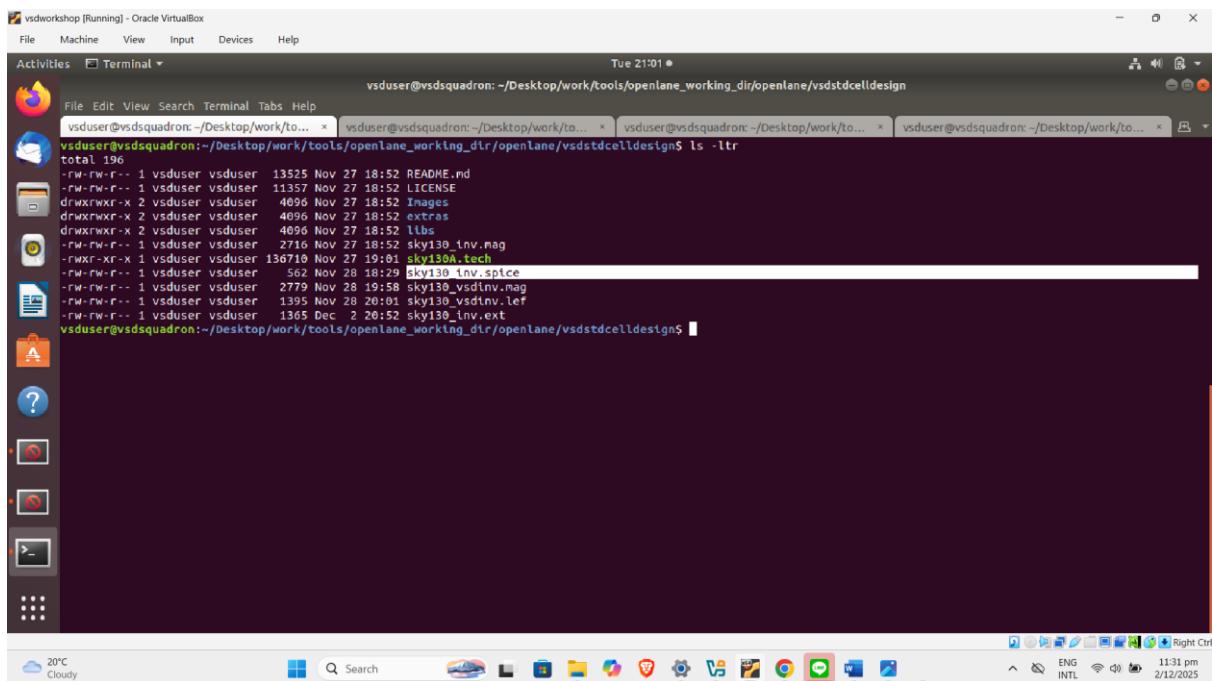
Now lets go to the vsdstd cell directory and check whether the spice netlist is extracted or not.



Now we will use this .ext file to extract the spice file to use with ngspice tool to verify the transfer characteristics. For extract the ngspice input the commands ext2spice cthresh 0 r thresh 0 followed by ext2spice



So, we are checking in the location whether the spice file has been created or not in the vsdstd cell directory.



Let's see what's inside the spice file

```

vsdworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal *
vsduser@vdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign
Tue 21:01 ●
vsduser@vdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ ls -ltr
total 196
-rw-rw-r-- 1 vsduser vsduser 13525 Nov 27 18:52 README.md
-rw-rw-r-- 1 vsduser vsduser 11357 Nov 27 18:52 LICENSE
drwxrwxr-x 2 vsduser vsduser 4696 Nov 27 18:52 Images
drwxrwxr-x 2 vsduser vsduser 4696 Nov 27 18:52 extras
drwxrwxr-x 2 vsduser vsduser 4696 Nov 27 18:52 libs
-rw-rw-r-- 1 vsduser vsduser 2716 Nov 27 18:52 sky130_inv.mag
-rw-rw-r-- 1 vsduser vsduser 136719 Nov 27 19:01 sky130A.tech
-rw-rw-r-- 1 vsduser vsduser 562 Nov 28 18:29 sky130_inv.spice
-rw-rw-r-- 1 vsduser vsduser 2779 Nov 28 19:58 sky130_vsdinv.mag
-rw-rw-r-- 1 vsduser vsduser 1395 Nov 28 20:01 sky130_vsdinv.lef
-rw-rw-r-- 1 vsduser vsduser 1365 Dec 2 20:52 sky130_inv.ext
vsduser@vdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ █

```

In the generated spicefile, we can see all the details about the NMOS and PMOS connectivity and information about the power also.

Now, we need to modify the ngspice file to simulate and view the results. So we include this file in terminal by .include.libs/pshort.lib and .include.libs/shsort.lib command

Set power supply VDD to 3.3V by assigning VDD VPWR 0 3.3V, and also add command to do transient analysis over certain time like. tran 1n 20n.

```

vsdworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal Wed 18:25
vsduser@vdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign
File Edit View Search Terminal Tab Help
vsduser@vdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ vns sky130_inv.spice
vsduser@vdsquadron: ~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ ngspice sky130_inv.spice
*****  

** ngspice-27 : Circuit level simulation program  

** The U. C. Berkeley CAD Group  

** Copyright (c) 1991-2017 by the University of California.  

** Please get your ngspice manual from http://ngspice.sourceforge.net/docs.html  

** Please file your bug-reports at http://ngspice.sourceforge.net/bugrep.html  

** Creation Date: Tue Dec 26 17:10:20 UTC 2017  

*****  

Circuit: * spice3 file created from sky130_inv.ext - technology: sky130a  

Scale set  

Doing analysis at TEMP = 27.000000 and TNOM = 27.000000  

Warning: va: no DC value, transient time 0 value used  

Initial Transient Solution  

Node Voltage  

----  

y 3.3  

gnd 0  

vdd 3.3  

vddbranch 0  

vssbranch 3.32336e-12  

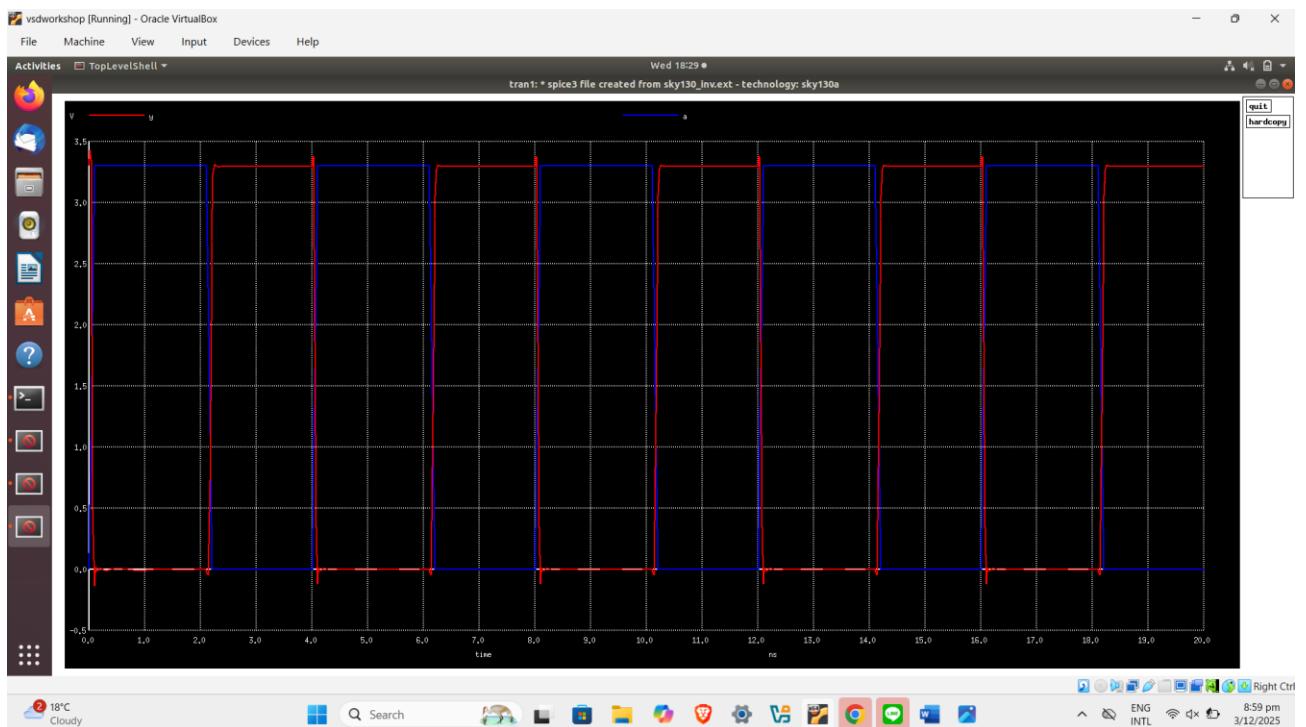
vddbranch -3.32337e-12  

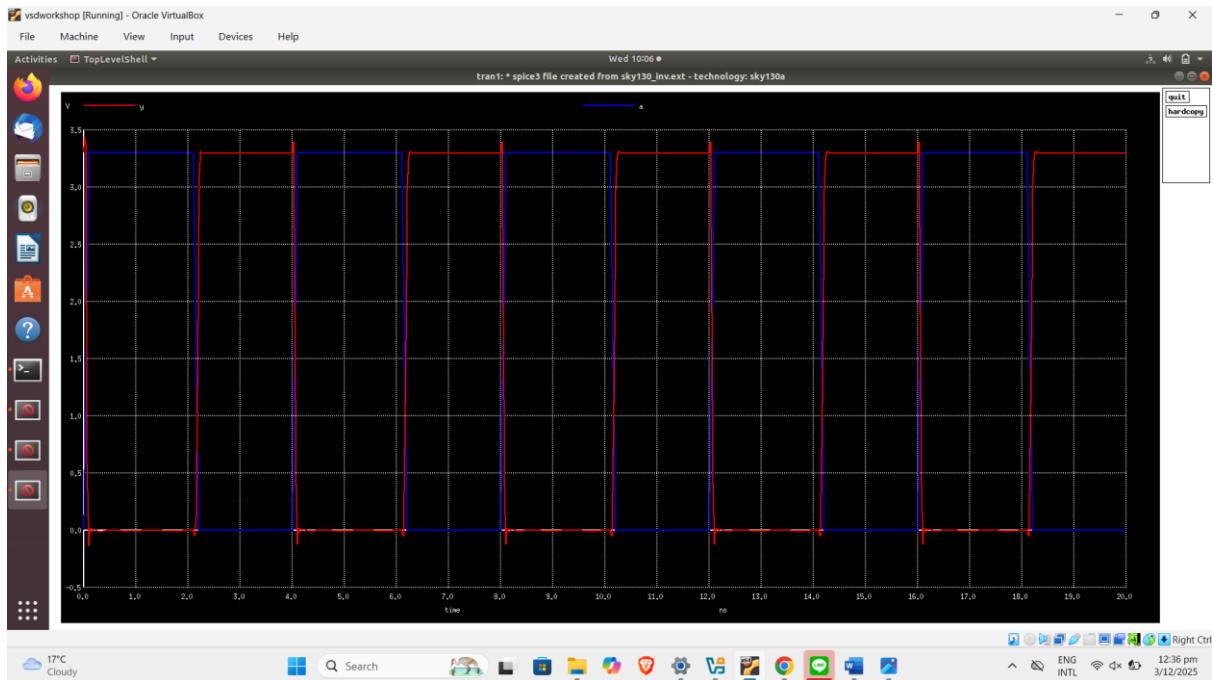
No. of Data Rows : 160
ngspice 1 -> plot y vs time a
ngspice 1 -> plot y vs time a
ngspice 1 -> plot y vs time a
ngspice 1 ->
x0 = 3.19277e-09, y0 = 2.14198 x1 = 6.49398e-09, y1 = 0.759259
dx = 3.3012e-09, dy = -1.38272
dy/dx = -4.18852e-08 dx/dy = -2.38748e-09
x0 = 2.71084e-09, y0 = 2.00025 x1 = 5.96386e-09, y1 = 1.16049

```

Now plotting the graph here by command, plot y vs time a



Now if we increase the C3 value from 0.00174ff to 2ff, the graph will look like this, it observed that the graph becomes smoother.



## Lab steps to characterize inverter using sky130 model files

Here we calculate the values of parameters

- Rise time

Rise time is defined as the time taken to reach 20% value to 80% value.

```
x0 = 6.15663e-09, y0 = 1.46154
x0 = 8.0241e-09, y0 = 1.98352
x0 = 5.72289e-09, y0 = 1.73077
```

So, the rise time =  $(5.72289 - 6.15663) \text{ e-}09 = 77.54 \text{ psec}$

- Fall time

It is time taken by output for transition from 80% to 20%

```
x0 = 2.09639e-09, y0 = 3.02747
x0 = 4.08434e-09, y0 = 2.9011
```

So, the fall time =  $(4.08434 - 2.09639) \text{ e-}09 = 19.9 \text{ psec}$

## Lab introduction to magic tool options and DRC rules

The technology file is set if rules that declares layers, types and electrical connectivity, DRC, device extraction rules, and rules to read lef and def files. It can be sourced from opencircuitdesign.com using the below path ‘

```
wget http://opencircuitdesign.com/open_pdks/archive/drc_tests.tgz
```

```
tar xfz drc_tests.tgz
```

```

vdsworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal
File Edit View Search Terminal Tabs Help
vsduser@vdsquadron: ~/Desktop/work/tool... vsduser@vdsquadron: ~/Desktop/work/tool... vsduser@vdsquadron: ~/Desktop/work/tool... vsduser@vdsquadron: ~/drc_tests vsduser@vdsquadron: ~/Desktop/work/tool...
Wed 18:56 •
vsduser@vdsquadron: ~/drc_tests
File Edit View Search Terminal Tabs Help
drc_tests.tgz
drwxr-xr-x 5 vsduser docker 4096 Jun 28 2021 pdks
drwxr-xr-x 10 vsduser docker 4096 Jun 29 2021 openlane_old
drwxr-xr-x 11 vsduser docker 4096 Nov 27 10:51 openlane
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir$ cd ../../..
vsduser@vdsquadron:~$ wget http://opencircuitdesign.com/open_pdks/archive/drc_tests.tgz
--2025-12-03 18:55:22-- http://opencircuitdesign.com/open_pdks/archive/drc_tests.tgz
Resolving opencircuitdesign.com... opencircuitdesign.com... 69.231.37.208
Connecting to opencircuitdesign.com (opencircuitdesign.com)|69.231.37.208|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 41651 (41K) [application/x-gzip]
Saving to: 'drc_tests.tgz'

drc_tests.tgz          100%[=====] 40.67K 127KB/s   0.3s

2025-12-03 18:55:21 (127 KB/s) - 'drc_tests.tgz' saved [41651/41651]

vsduser@vdsquadron:~$ tar xfz drc_tests.tgz
vsduser@vdsquadron:~$ ls -ltr
total 104
drwxr-xr-x 2 vsduser vsduser 4096 Sep 16 2020 drc_tests
-rw-rw-r-- 1 vsduser vsduser 41651 Sep 16 2020 drc_tests.tgz
drwxr-xr-x 2 vsduser vsduser 4096 May 19 2021 Videos
drwxr-xr-x 2 vsduser vsduser 4096 May 19 2023 Templates
drwxr-xr-x 2 vsduser vsduser 4096 May 19 2023 Public
drwxr-xr-x 2 vsduser vsduser 4096 May 19 2023 Music
drwxr-xr-x 2 vsduser vsduser 4096 May 19 2023 Documents
drwxr-xr-x 2 vsduser vsduser 4096 May 20 2023 Downloads
drwxr-xr-x 3 vsduser vsduser 4096 May 20 2023 Desktop
drwxr-xr-x 18 vsduser vsduser 4096 May 22 2023 sysos
drwxr-xr-x 47 vsduser vsduser 4096 May 22 2023 magic
drwxr-xr-x 24 vsduser vsduser 4096 May 22 2023 met3
drwxr-xr-x 2 vsduser vsduser 4096 Nov 25 17:49 Pictures
vsduser@vdsquadron:~/drc_tests
vsduser@vdsquadron:~/drc_tests$ ls -ltr
total 264
-rw-rw-r-- 1 vsduser vsduser 955 Sep 15 2020 via.mag
-rw-rw-r-- 1 vsduser vsduser 966 Sep 15 2020 via4.mag
-rw-rw-r-- 1 vsduser vsduser 1267 Sep 15 2020 via3.mag
-rw-rw-r-- 1 vsduser vsduser 1271 Sep 15 2020 via2.mag
-rw-rw-r-- 1 vsduser vsduser 3025 Sep 15 2020 rpm.mag
-rw-rw-r-- 1 vsduser vsduser 2591 Sep 15 2020 psd.mag
-rw-rw-r-- 1 vsduser vsduser 3164 Sep 15 2020 nsp.mag
-rw-rw-r-- 1 vsduser vsduser 2497 Sep 15 2020 nsd.mag
-rw-rw-r-- 1 vsduser vsduser 1948 Sep 15 2020 npe.mag

```

The desktop environment includes a taskbar with icons for various applications like a browser, file manager, and system tools. The system tray shows the date (3/12/2025), time (9:26 pm), battery level (ENG INTL), and network status.

The content of magicrc file is showcased below

```

vdsworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal
File Edit View Search Terminal Tabs Help
vsduser@vdsquadron: ~/Desktop/work/tool... vsduser@vdsquadron: ~/Desktop/work/tool... vsduser@vdsquadron: ~/Desktop/work/tool... vsduser@vdsquadron: ~/drc_tests vsduser@vdsquadron: ~/Desktop/work/tool...
Wed 18:59 •
vsduser@vdsquadron: ~/drc_tests
File Edit View Search Terminal Tabs Help
# Put grid on 0.005 pitch. This is important, as some commands don't
# rescale the grid automatically (such as lef read?).
# drc off
drc euclidean on
# Allow override of PDK path from environment variable PDKPATH
if {[catch {set PDKPATH $env(PDKPATH)}]} {
    set PDKPATH "~/cad/pdk/sky130A"
}
# loading technology
tech load $PDKPATH/libraries/tech/magic/sky130A.tech
tech load sky130A.techn
# Load device generator
# source $PDKPATH/libraries/tech/magic/sky130A-BlndKeys
# set units to lambda grid
snap lambda
# set sky130 standard power, ground, and substrate names
set VDD VDD
set VSS VSS
set SUB VSUBS
# Allow override of type of magic library views used, "mag" or "maglef",
# from environment variable MAGTYPE
# if {[catch {set MAGTYPE $env(MAGTYPE)}]} {
#     set MAGTYPE maglef
# }
# add path to reference cells
".maglef" 74L_256SC

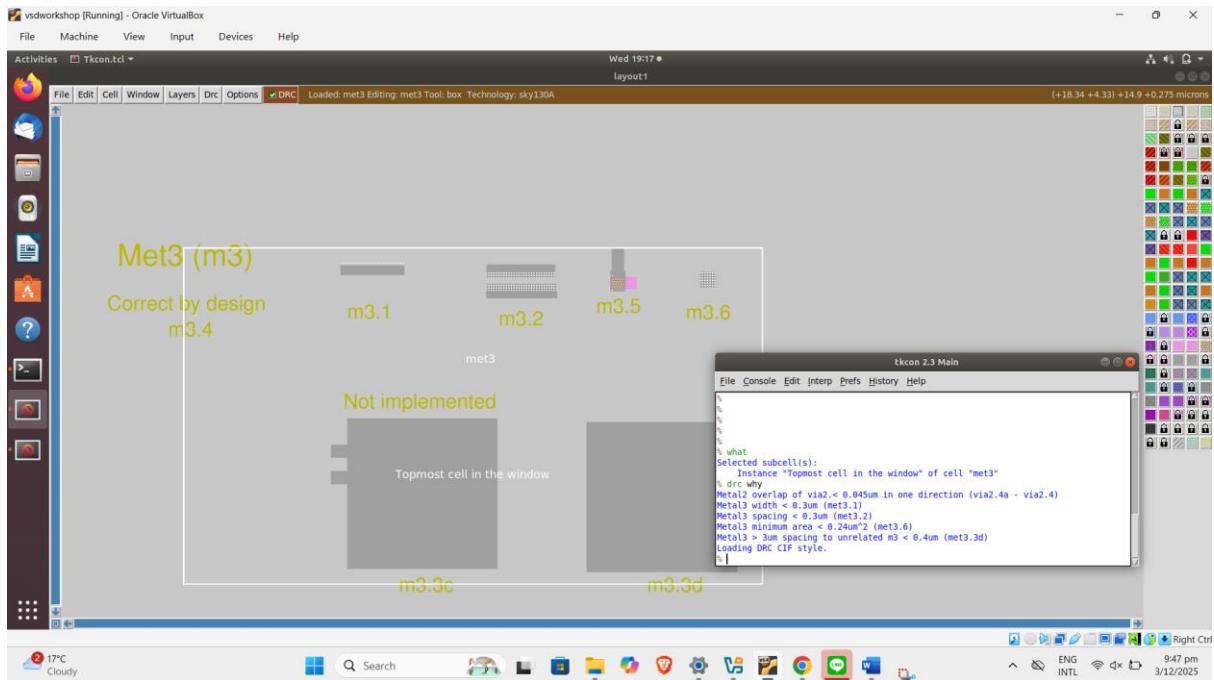
```

The desktop environment includes a taskbar with icons for various applications like a browser, file manager, and system tools. The system tray shows the date (3/12/2025), time (9:26 pm), battery level (ENG INTL), and network status.

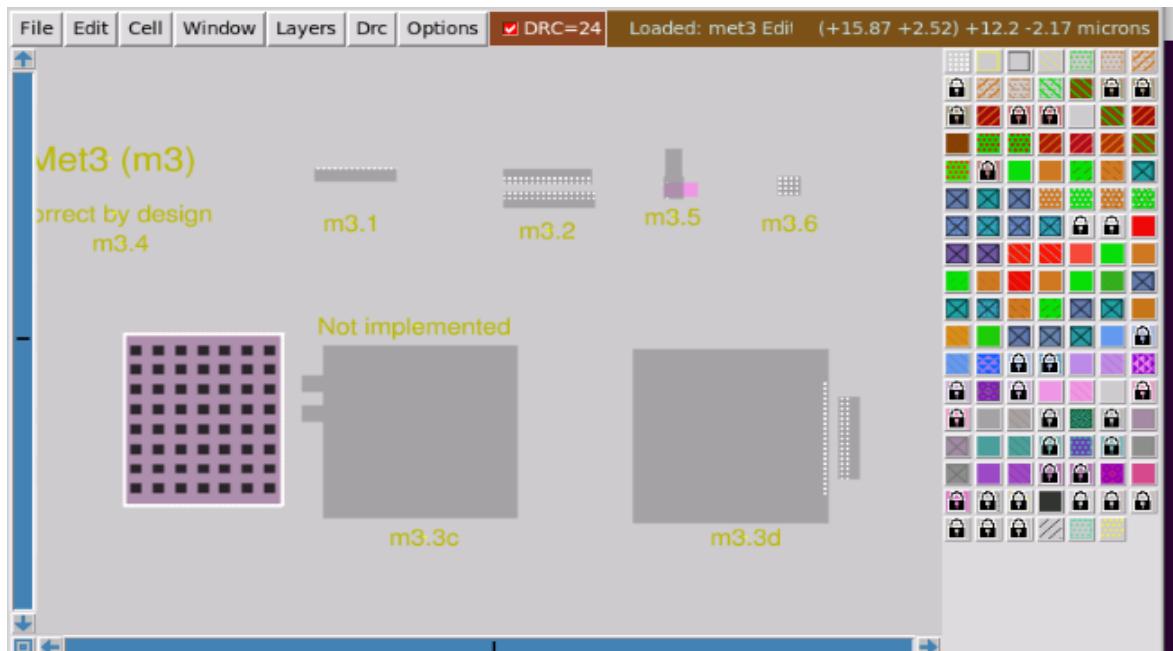
## Lab introduction to Magic steps and to load sky 130 tech rules

Use the command `magic -d XR`

To open the magic tool. Open `met3.mag` file from the file menu. We can see different layouts with different DRC values.



To check for vias in the metal3 layer, make a rectangular selection in an empty space and paint it with the m3contact color from the color palette by clicking middle mouse button. The vias can be viewed by: cif see VIA2



## Day 4: Timing Analysis and CTS

A requirement for ports is specified in tracks.info, because certain guidelines must be followed like, The input and output ports must lie on the intersection of the vertical and horizontal tracks and the width of the standard cell should be an odd multiple of the track pitch, and the height should be an odd multiple of the track vertical pitch.



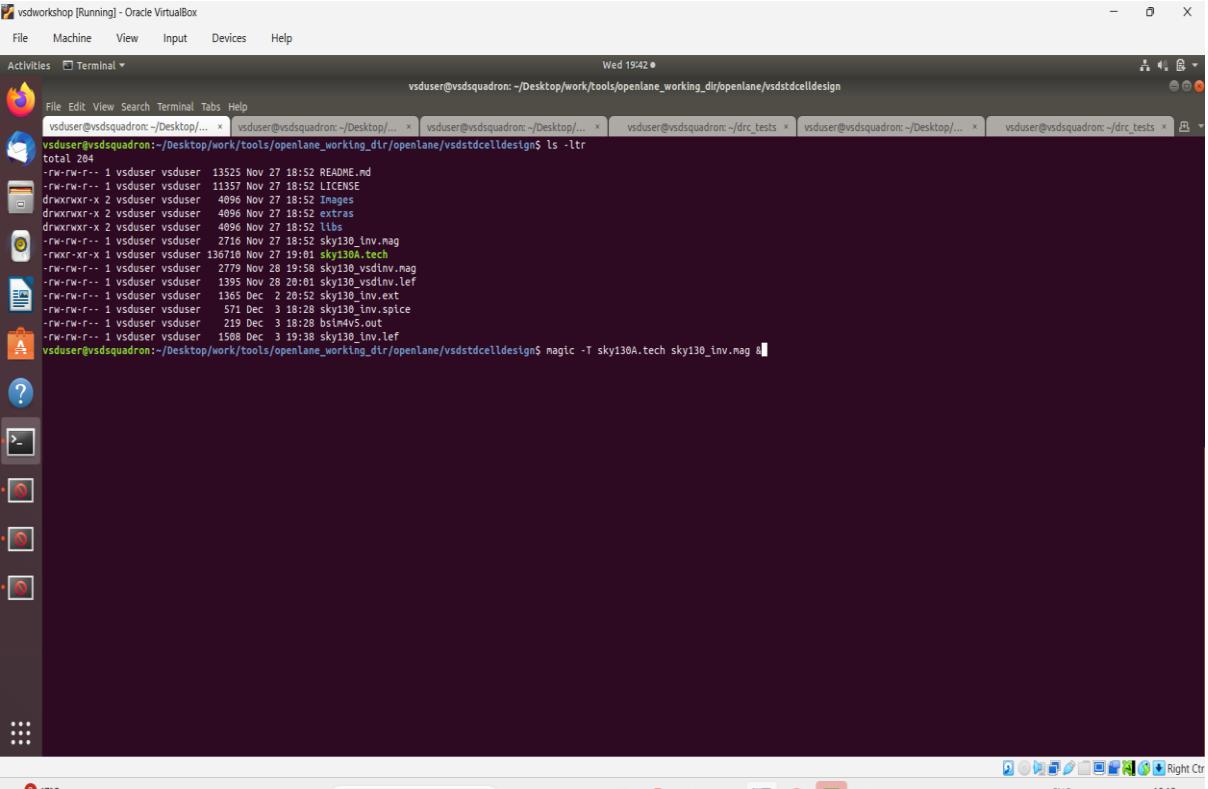
```

l1i X 0.23 0.46
l1i Y 0.17 0.34
met1 X 0.17 0.34
met1 Y 0.17 0.34
Met2 X 0.23 0.46
Met2 Y 0.23 0.46
met3 X 0.34 0.68
met3 Y 0.34 0.68
met4 X 0.46 0.92
met4 Y 0.46 0.92
met5 X 1.78 3.40
met5 Y 1.78 3.40
tracks.info (END)

```

## Standard cell lef generation

To generate .lef file, first invoke magic tool to view the vsdstd cell design. Now type lef write on tckon window, the lef file will be generated in the folder as shown in the below steps:



```

vsdworkshop [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal ▾ Wed 19:42 ●
vsduser@vsdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign
File Edit View Search Terminal Tabs Help
vsduser@vsdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ ls -ltr
total 264
-rw-rw-r-- 1 vsduser vsduser 13525 Nov 27 18:52 README.md
-rw-rw-r-- 1 vsduser vsduser 11357 Nov 27 18:52 LICENSE
drwxrwxr-x 2 vsduser vsduser 4096 Nov 27 18:52 Images
drwxrwxr-x 2 vsduser vsduser 4096 Nov 27 18:52 extras
drwxrwxr-x 2 vsduser vsduser 4096 Nov 27 18:52 lib
-rw-rw-r-- 1 vsduser vsduser 2716 Nov 27 18:52 sky130_inv.mag
-rw-rw-r-- 1 vsduser vsduser 136718 Nov 27 19:01 sky130A.tech
-rw-rw-r-- 1 vsduser vsduser 2779 Nov 28 19:58 sky130_vsdinv.mag
-rw-rw-r-- 1 vsduser vsduser 1395 Nov 28 20:01 sky130_vsdinv.lef
-rw-rw-r-- 1 vsduser vsduser 1365 Dec 2 20:52 sky130_inv.ext
-rw-rw-r-- 1 vsduser vsduser 571 Dec 3 18:28 sky130_inv.spice
-rw-rw-r-- 1 vsduser vsduser 219 Dec 3 18:28 bstM4V5.out
-rw-rw-r-- 1 vsduser vsduser 1588 Dec 3 19:38 sky130_inv.lef
vsduser@vsdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ magic -T sky130A.tech sky130_inv.mag

```

Now the lef file is generated on vsd design folder

```

vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ ls -ltr
total 264
-rw-rw-r-- 1 vsduser vsduser 13525 Nov 27 18:52 README.md
drwxrwxr-x 2 vsduser vsduser 4096 Nov 27 18:52 Images
drwxrwxr-x 2 vsduser vsduser 4096 Nov 27 18:52 extras
drwxrwxr-x 2 vsduser vsduser 4096 Nov 27 18:52 tcl
-rw-rw-r-- 1 vsduser vsduser 2716 Nov 27 18:52 sky130_inv.mag
-rw-rw-r-- 1 vsduser vsduser 136718 Nov 27 19:01 sky130A.tech
-rw-rw-r-- 1 vsduser vsduser 2779 Nov 28 19:58 sky130_vsdinv.mag
-rw-rw-r-- 1 vsduser vsduser 1395 Nov 28 20:01 sky130_vsdinv.lef
drwxrwxr-x 2 vsduser vsduser 4096 Nov 28 20:01 sky130_inv.tcl
-rw-rw-r-- 1 vsduser vsduser 1365 Nov 28 20:01 sky130_inv.lef
-rw-rw-r-- 1 vsduser vsduser 571 Dec 3 18:28 sky130_inv_spice
-rw-rw-r-- 1 vsduser vsduser 219 Dec 3 18:28 bsin4v5.out
-rw-rw-r-- 1 vsduser vsduser 1508 Dec 3 19:38 sky130_inv.lef
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ magic -T sky130A.tech sky130_inv.mag &
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ ls -lyr
ls: invalid option -- 'y'
Try 'ls --help' for more information.
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$ ls -ltr
total 264
-rw-rw-r-- 1 vsduser vsduser 13525 Nov 27 18:52 README.md
drwxrwxr-x 2 vsduser vsduser 4096 Nov 27 18:52 LICENSE
drwxrwxr-x 2 vsduser vsduser 4096 Nov 27 18:52 extras
drwxrwxr-x 2 vsduser vsduser 4096 Nov 27 18:52 tcl
-rw-rw-r-- 1 vsduser vsduser 2716 Nov 27 18:52 sky130_inv.mag
-rw-rw-r-- 1 vsduser vsduser 136718 Nov 27 19:01 sky130A.tech
-rw-rw-r-- 1 vsduser vsduser 2779 Nov 28 19:58 sky130_vsdinv.mag
-rw-rw-r-- 1 vsduser vsduser 1395 Nov 28 20:01 sky130_vsdinv.lef
drwxrwxr-x 2 vsduser vsduser 4096 Nov 28 20:01 sky130_inv.tcl
-rw-rw-r-- 1 vsduser vsduser 1365 Nov 28 20:01 sky130_inv.lef
-rw-rw-r-- 1 vsduser vsduser 571 Dec 3 18:28 sky130_inv_spice
-rw-rw-r-- 1 vsduser vsduser 219 Dec 3 18:28 bsin4v5.out
-rw-rw-r-- 1 vsduser vsduser 1508 Dec 3 19:38 sky130_inv.lef
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/vsdstdcelldesign$
```

## Introduction to timing libraries and steps to include new cell in synthesis

Now the lef file has been created, now we need to move this file to picorv32a. Before that we need to move this file to src folder so that all the designs will be available at on location.

```

vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/designs/picorv32a/src$ ls -ltr
total 32
-rw-rw-r-- 1 vsduser docker 209 Jun 29 2021 sky130A.sky130_fd_sc_ms_config.tcl
-rw-rw-r-- 1 vsduser docker 209 Jun 29 2021 sky130A.sky130_fd_sc_ls_config.tcl
-rw-rw-r-- 1 vsduser docker 209 Jun 29 2021 sky130A.sky130_fd_sc_hs_config.tcl
-rw-rw-r-- 1 vsduser docker 209 Jun 29 2021 sky130A.sky130_fd_sc_hd_config.tcl
-rw-rw-r-- 1 vsduser docker 444 Jun 29 2021 config.tcl
drwxr-xr-x 2 vsduser docker 4096 Nov 28 20:25 src
drwxr-xr-x 4 vsduser vsduser 4096 Dec 1 10:12 run
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/designs/picorv32a/src$ cd src
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/designs/picorv32a/src$ ls -ltr
total 37426
-rw-rw-r-- 1 vsduser docker 92423 Jun 29 2021 picorv32a.v
-rw-rw-r-- 1 vsduser docker 77 Jun 29 2021 picorv32a.sdc
-rw-rw-r-- 1 vsduser docker 13826 Jun 29 2021 sky130_fd_sc_hd.lef
-rw-rw-r-- 1 vsduser vsduser 12753932 Nov 28 20:25 sky130_fd_sc_hd_fast.lib
-rw-rw-r-- 1 vsduser vsduser 12732258 Nov 28 20:25 sky130_fd_sc_hd_low.lib
-rw-rw-r-- 1 vsduser vsduser 12732345 Nov 28 20:25 sky130_fd_sc_hd_typical.lib
vsduser@vdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane/designs/picorv32a/src$
```

Here we need to modify the config.tcl of picorv32a directory and add commands shown as below image

```

# Design
set :env(DESIGN_NAME) "picorv32a"

set :env(VERILOG_FILES) ".:/designs/picorv32a/src/picorv32a.v"
set :env(SDC_FILE) ".:/designs/picorv32a/src/picorv32a.sdc"
set ::env(CLOCK_PERIOD) "5.000"
set ::env(CLOCK_PORT) "clk"

set ::env(CLOCK_NET) $::env(CLOCK_PORT)

set ::env(FP_CORE_UTIL) 05
set ::env(FP_IO_VHDL) 4
set ::env(FP_IO_METAL) 3

set ::env(LIB_SYNTH) "$::env(OPENLANE_ROOT)/designs/picorv32a/src/sky130_fd_sc_hd_typical.lib"
set ::env(LIB_FASTEST) "$::env(OPENLANE_ROOT)/designs/picorv32a/src/sky130_fd_sc_hd_fast.lib"
set ::env(LIB_SLOWEST) "$::env(OPENLANE_ROOT)/designs/picorv32a/src/sky130_fd_sc_hd_slow.lib"
set ::env(LIB_TYPICAL) "$::env(OPENLANE_ROOT)/designs/picorv32a/src/sky130_fd_sc_hd_typical.lib"

set ::env(EXTRA_LEFS) [glob $::env(OPENLANE_ROOT)/designs/$::env(DESIGN_NAME)/src/*.lef]

set filename $::env(OPENLANE_ROOT)/designs/$::env(DESIGN_NAME)/$::env(PDK)_$::env(ST0_CELL_LIBRARY)_config.tcl
if { [file exists $filename] == 1 } {
    source $filename
}

```

In order to integrate standard cell into the openlane input these command in the openlane flow

prep -design picorv32a -tag 02-07\_07-56 -overwrite

set lefs [glob \$::env(DESIGN\_DIR)/src/\*.lef]

add\_lefs -src \$lefs

run\_synthesis

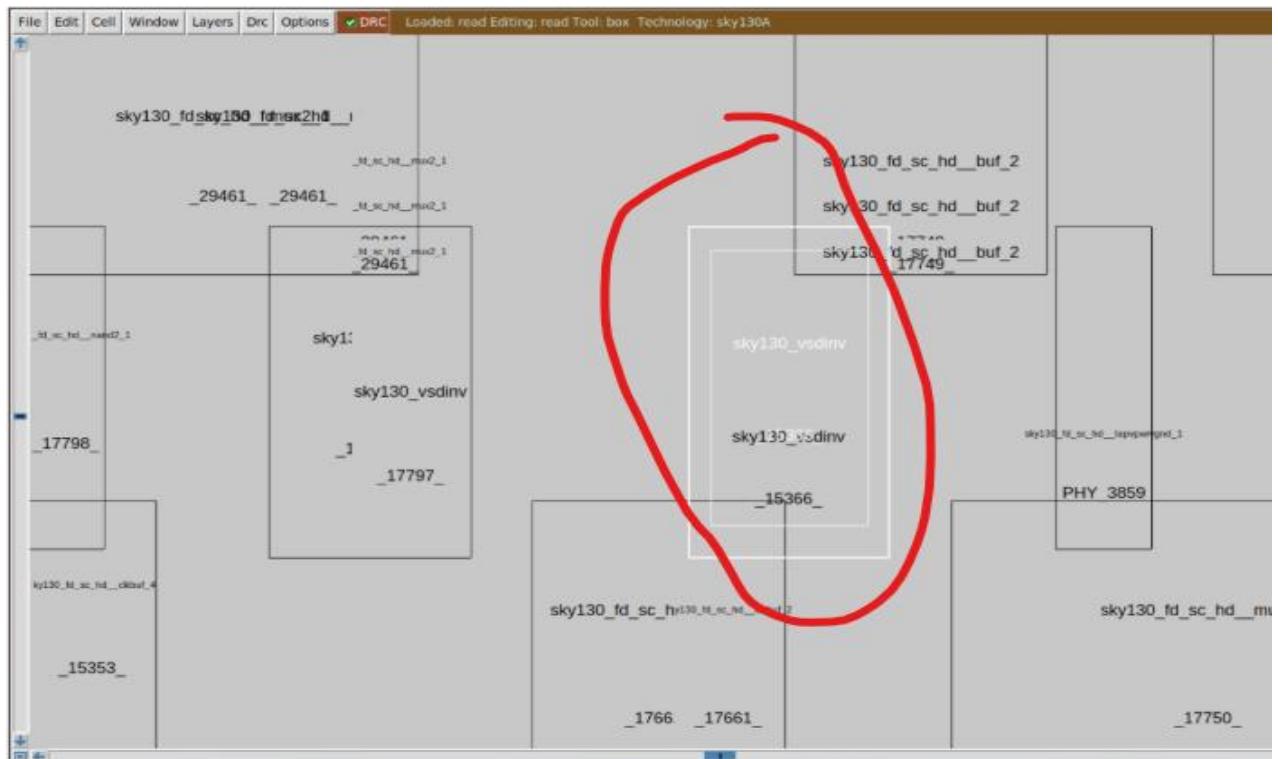
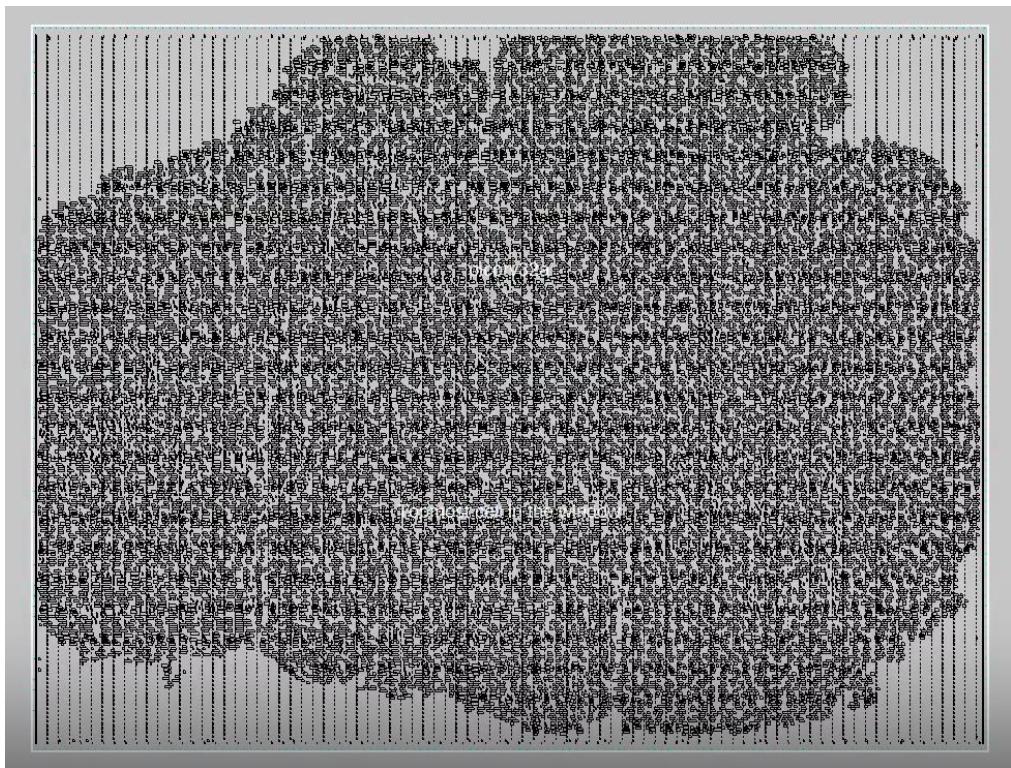
```

vsduser@vsdsquadron:~/Desktop/work/tools/openlane_working_dir/openlane
File Edit View Insert Help
chip area for module '\picorv32a': 17712.918400
29. Executing vsynth...
dumping module '\picorv32a'.
Warnings: 307 unique messages, 307 total
End of script. Logfile hash: db33866223, CPU: user 25.35s system 1.08s, MEM: 96.55 MB peak
Yosys 0.9+3621 (git sha1 84e9fa7, gcc 8.3.1 -fPIC -O5)
Time spent: 53K zx abc (29 sec), 12K 33x opt_expr (6 sec), ...
[INFO]: Changing working directory to /openlane_flow/designs/picorv32a/runs/30_03_20_42/results/synthesis/picorv32a.synthesis.v
[INFO]: Starting Static Timing Analysis...
[INFO]: current step index: 2
OpenSTA 2.3.0 3bb40303a Copyright (c) 2019, Parallax Software, Inc.
License GPLv3: GNU General Public Version 3 <http://gnu.org/licenses/gpl.html>
This is free software, and you are free to change and redistribute it
under certain conditions; type 'show warranty' for details.
This program comes with ABSOLUTELY NO WARRANTY; for details type 'show warranty'.
create_clock [get_ports $::env(CLOCK_PORT)] -name $::env(CLOCK_PORT) -period $::env(CLOCK_PERIOD)
set_input_delay_value [expr $::env(CLOCK_PERIOD) * $::env(CLOCK_PCT)]
set_output_delay_value [expr $::env(CLOCK_PERIOD) * $::env(IO_PCT)]
puts '\[INFO\]: Setting output delay to: $output_delay_value'
[INFO]: Setting output delay to: 4.946000000000000
puts '\[INFO\]: Setting input delay to: $input_delay_value'
[INFO]: Setting input delay to: 4.946000000000000
set_rst_inidx [search [all_inputs] [get_port $::env(CLOCK_PORT)]]
set_clk_inidx [search [all_inputs] [get_port resetn]]
set_all_inputs_wo_clk [replace [all_inputs] $clk_inidx $clk_inidx]
set_all_inputs_wo_clk_rst [replace $all_inputs_wo_clk $rst_inidx $rst_inidx]
# correct resetn
set_input_delay $input_delay_value -clock [get_clocks $::env(CLOCK_PORT)] $all_inputs_wo_clk_rst
#set_input_delay 0.0 -clock [get_clocks $::env(CLOCK_PORT)] (resetn)
set_output_delay $output_delay_value -clock [get_clocks $::env(CLOCK_PORT)] [all_outputs]
#set_output_delay 0.0 -clock [get_clocks $::env(CLOCK_PORT)] (resetn)
set_driving_cell $cell $::env(SYNTH_DRIVING_CELL) -pin $::env(SYNTH_DRIVING_CELL_PIN) [all_inputs]
set_cap_load [expr $::env(SYNTH_CAP_LOAD) / 1000.0]
puts '\[INFO\]: Setting load to: $cap_load'
[INFO]: Setting load to: 0.01765
set_low_cap_load [all_outputs]
time 731.59
ns 23.89
[INFO]: Synthesis was successful
%
```

Synthesis was successful

Post synthesis and floorplan placement is done.

To view the layout, magic tool is invoked in results/placement directory



Sky130\_vsdinv is added to the picorv32a design.

## Post synthesis timing analysis

Timing Analysis is carried out using openSTA outside the openlane flow. To invoke openSTA follow the command given below

Sta pre\_sta.conf

```
          0.12  0.00  46.54 v _13750/_B (sky130_fd_sc_hd__or2_2)
          0.14  0.00  47.22 v _13750/_X (sky130_fd_sc_hd__or2_2)
          0.14  0.00  10970_ (net)
          0.12  0.00  47.22 v _13751/_B (sky130_fd_sc_hd__or2_2)
          0.12  0.00  47.88 v _13751/_X (sky130_fd_sc_hd__or2_2)
          0.08  0.00  10971_ (net)
          0.12  0.00  47.88 v _13754/_B (sky130_fd_sc_hd__o221a_2)
          0.07  0.44  48.32 v _13754/_X (sky130_fd_sc_hd__o221a_2)
          0.08  0.00  0.3928_ (net)
          0.07  0.00  48.32 v _27762/_D (sky130_fd_sc_hd_dfxtpl_2)
          0.07  0.00  48.32 v _27762/_D (sky130_fd_sc_hd_dfxtpl_2)
          0.00  24.73  24.73  clock clk (rise edge)
          0.00  24.73  24.73  clock network delay (ideal)
          0.00  24.73  24.73  clock reconvergence pessimism
          24.73 ^ _27762/_CLK (sky130_fd_sc_hd_dfxtpl_2)
          -0.29  24.44  library setup time
          24.44  data arrival time
          24.44  data required time
          -48.32  data arrival time
          -23.89  slack (VIOLATED)

ms -711.59
ms -23.89
```

Since **Clock Tree Synthesis (CTS)** is not yet completed, the timing analysis is performed using an **ideal clock**. At this stage, only **setup time slack** is evaluated. Setup slack is defined as the difference between the **required arrival time** and the **actual data arrival time**. For timing to be met, the **worst-case slack must be zero or positive**. If a negative slack is observed during pre-CTS analysis, the following corrective actions can be taken:

- Adjust the **synthesis strategy**, including buffering and cell sizing options.
- Check for cells that violate maximum fanout limits and replace or restructure high-fanout cells to reduce delay.

## Day 5: Final steps to RTL to GDSII

In comparison with ASIC flow, the power distribution in openlane is out of floorplan. PDN must be generated post CTS and timing analysis

To generate PDN network give command mentioned below

gen\_pdn

To ensure that PDN network is generated successfully you can give variable

```
echo ${::env(CURRENT_DEF)}
```

Routing:

Openlane uses two stages of routing. One is

- Fast routing: Routing region is divided into rectangular grids
- Detailed routing: routing region is divided into finer grid to implement physical wiring using trionroute tool.

To run routing input command

```
run_routing
```

## References

- <https://github.com/google/skywater-pd>
- <https://github.com/nickson-jose/vsdstdcelldesign>
- <https://sourceforge.net/projects/ngspice/>