Tutorial Number 1

Getting start with OpenLane

LEARNING OUTCOMES:

At the end of this lab, participants will be able to:

1. Prepare a design to go through the openlane flow
2. Distinguish between interactive and non-interactive flow
3. Get an understanding of all the steps involved in auto place and route (APR)
4. Traverse between all the directories created from a run

PREREQUISITE:

Before this lab, participants should:

1) have a basic understanding of linux command like:

   a-cd -Change Directory

   cd /home/merl-tools/openlane_v0.9/openlane [to go to the working directory]
   cd .. [to go back one directory]

   b-ls -List

   c-vim -Text Editor

   :q! -to exit

   i - to enter insert mode

   escape - to get out of insert mode
   :x! or :wq -to save and exit

   d-dp -To copy

   cp <old_name> <new_name>
2) know how to run TCL scripts. A good TCL tutorial can be found here.
3) know digital logic design and basic electronics

The editor we have used is a terminal based editor called Vim and you can use an editor of your own choice.

MAJOR COMPONENTS OF PHYSICAL DESIGN (OPEN-SOURCE)

Google and Skywater foundry in collaboration with efabless announced an open-source 130nm process design kit (PDK). This PDK is itself built from two components which give us the final **sky130A** directory:

- Skywater-pdk - Contains all the foundry provided PDK related files.
- Open_pdk - Contains scripts that are used to bridge the gap between closed-source and open-source PDK to EDA tool compatibility

```
~/Documents/pdks
❯ ls
drwxrwxr-x - rameen 12 Jul 19:49 open_pdks
drwxrwxr-x - rameen 12 Jul 19:53 sky130A
drwxrwxr-x - rameen 12 Jul 19:47 skywater-pdk
```

The EDA tool mentioned above is openlane.It ships with a pre-installed

PDK structure

Sky130A

- libs.tech
  - openlane
  - klayout
  - ngspice
  - magic
  - qflow
- libs.ref
  - sky130_fd_sc_hd
  - sky130_fd_sc_hdll
  - sky130_fd_sc_hs
- SOURCE

RUNNING THE FLOW

1. make test

```
OpenLane on  HEAD (7496f21) via  v3.8.10
❯ make test
cd /home/rameen/Documents/OpenLane && \
	docker run -it --rm -v /home/rameen/Documents/OpenLane:/openLANE_flow -v /home/rameen/Documents/pdks:/home/rameen/Documents/pdks -e PDK_ROOT=/
home/rameen/Documents/pdks -u 1000:1000 efabless/openlane:v0.9 sh -c "./flow.tcl -design spm -tag openlane_test -disable_output -overwrite"
[INFO]:

       __   ___   __ ___  _      ___  ___    __
      /  \ |   \ /  _]|   \ | |    /   ||   \  /  _]
      |    ||  o  )  [_ |  _  || |   |  o  ||  _  | /  [_
      |  O ||   _/    _]|  |  || |__ |     ||  |  ||    _]
      |    ||  | |   [_ |  |  ||    ||  _  ||  |  ||   [_
      \__/ |__| |____||__|__||____||__|__||__|__||____|
```

2. Non-Interactive

```
OpenLane on  HEAD (7496f21) [!] via  v3.8.10
❯ make mount
cd /home/rameen/Documents/OpenLane && \
        docker run -it --rm -v /home/rameen/Documents/OpenLane:/openLANE_flow -v /home/rameen/Documents/pdks:/home/rameen/Documents/pdks -e PDK_ROOT=/
home/rameen/Documents/pdks -u 1000:1000 efabless/openlane:v0.9
bash-4.2$ ./flow.tcl -design spm
[INFO]:
       ___   ____   ___   ____   _       ____  ____     ___
       /   \ |    \ /   \|    \ | |     /    ||    \   /  _]
       |     ||  o  )     |  _  | | |    |  o  ||  _  | /  [_
       |  O  ||   _/|  O  |  |  | | |___ |     ||  |  ||    _]
       |     ||  |  |     |  |  | |     ||  _  ||  |  ||   [_
       \___/ |__|  |_____|__|__| |_____||__|__||__|__||_____|


[INFO]: Version: v0.9
[INFO]: Running non-interactively
[INFO]: Using design configuration at /openLANE_flow/designs/spm/config.tcl
[INFO]: Sourcing Configurations from /openLANE_flow/designs/spm/config.tcl
```

3. Interactive

```
OpenLane on  HEAD (7496f21) [?] via  v3.8.10 took 5m12s
❯ make mount
cd /home/rameen/Documents/OpenLane && \
        docker run -it --rm -v /home/rameen/Documents/OpenLane:/openLANE_flow -v /home/rameen/Documents/pdks:/home/rameen/Documents/pdks -e PDK_ROOT=/
home/rameen/Documents/pdks -u 1000:1000 efabless/openlane:v0.9
bash-4.2$ ./flow.tcl -interactive
[INFO]:
       ___   ____   ___   ____   _       ____  ____     ___
       /   \ |    \ /   \|    \ | |     /    ||    \   /  _]
       |     ||  o  )     |  _  | | |    |  o  ||  _  | /  [_
       |  O  ||   _/|  O  |  |  | | |___ |     ||  |  ||    _]
       |     ||  |  |     |  |  | |     ||  _  ||  |  ||   [_
       \___/ |__|  |_____|__|__| |_____||__|__||__|__||_____|


[INFO]: Version: v0.9
[INFO]: Running interactively
% prep -design spm
[INFO]: Using design configuration at /openLANE_flow/designs/spm/config.tcl
[INFO]: Sourcing Configurations from /openLANE_flow/designs/spm/config.tcl
[INFO]: PDKs root directory: /home/rameen/Documents/pdks
[INFO]: PDK: sky130A
[INFO]: Setting PDKPATH to /home/rameen/Documents/pdks/sky130A
[INFO]: Standard Cell Library: sky130_fd_sc_hd
[INFO]: Sourcing Configurations from /openLANE_flow/designs/spm/config.tcl
[INFO]: Current run directory is /openLANE_flow/designs/spm/runs/13-07_18-52
[INFO]: Preparing LEF Files
[INFO]: Extracting the number of available metal layers from /home/rameen/Documents/pdks/sky130A/libs.ref/sky130_fd_sc_hd/techlef/sky130_fd_sc_hd__tlef
```

1. You can check there are two folders in design/gcd

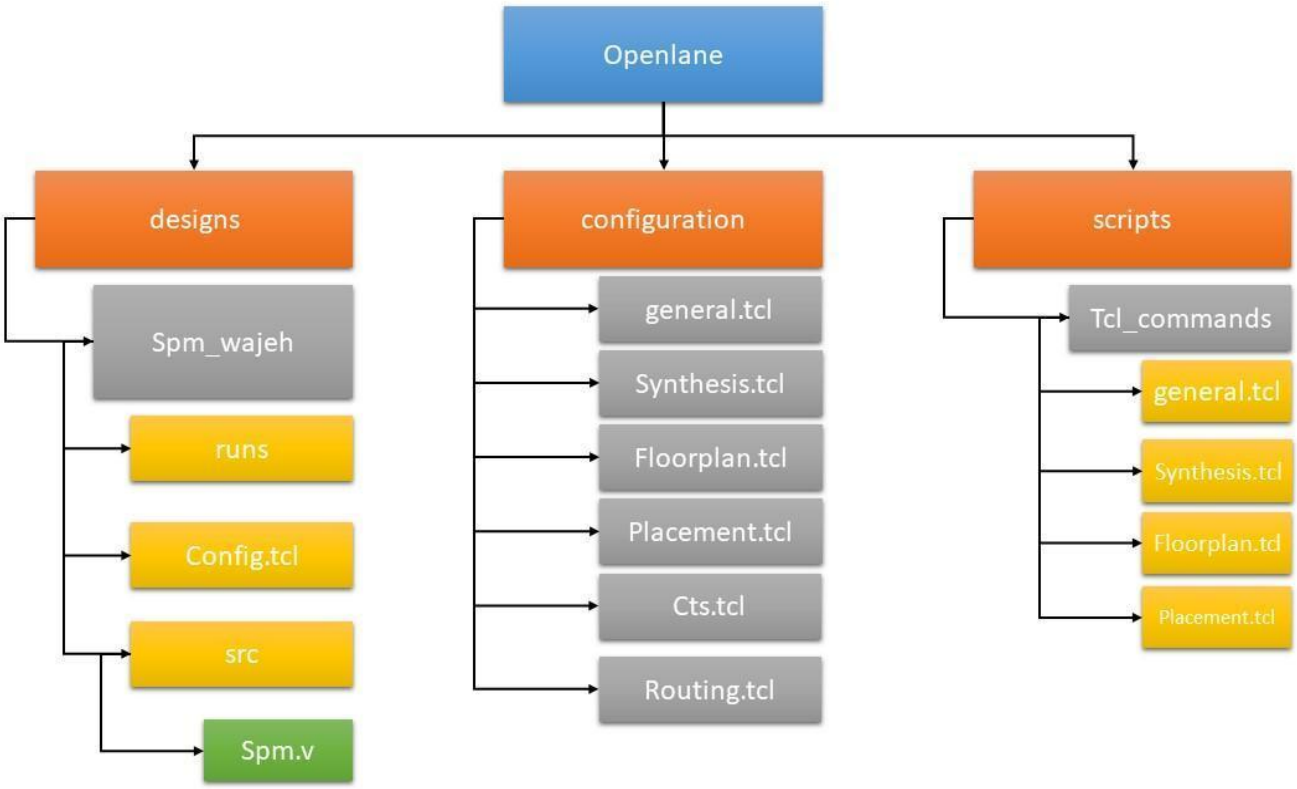
```
designs/gcd on  HEAD (7496f21) [!?]

x    - rameen 13 Jul 22:22 src
-  329 rameen 13 Jul 22:22 config.tcl

designs/gcd on  HEAD (7496f21) [!?]
```

2. Copy your design into src folder

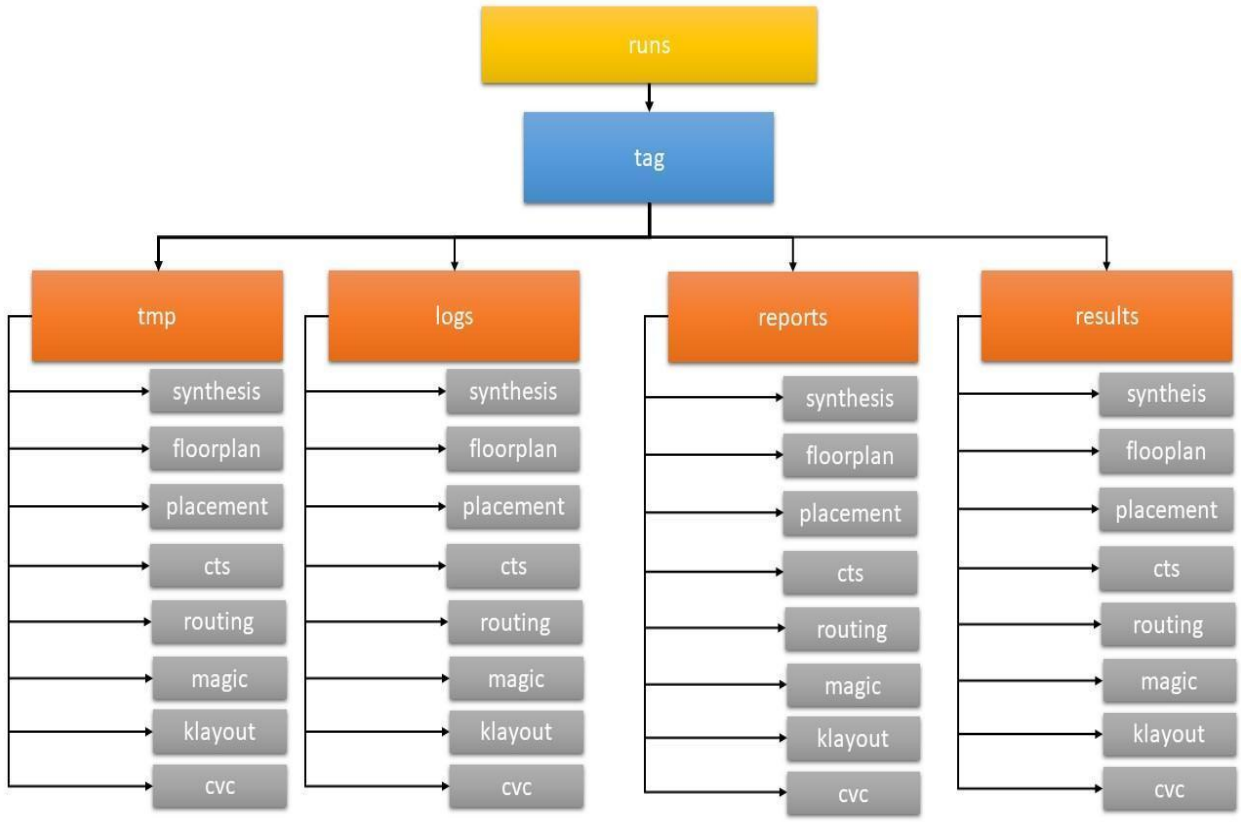
```
~/Downloads
❯ mv gcd/src /home/rameen/Documents/OpenLane/designs/gcd
```

3. Verify the Design name and Clock port name into src/<top_module_file> and config.tcl file
4. Now you can prep the design

OpenLane structure



Runs structure



Tasks

1. Explore different flags with **./flow.tcl**, for example:
   a) ./flow.tcl –design <design_name> -config_file
2. Locate the GDS file of inverter provided by the foundry
   a) HINT: The name of the file is **sky130_fd_sc_hd__inv_2.gds**
3. Explore which configuration file has the highest priority
4. Explore if an environment variable can be changed after **prepping** the design, if yes then how?
5. Explore if we can use different library other than sky130_fd_sc_hd