**DAY -04 Pre–Layout Timing Analysis and importance of good clock tree**

**Timing modeling Using Delay table**

**1.Lab Steps to Convert grid info to track info**

Next, we need to extract the .lef file from the .mag file and place it into the picorv32a flow.

So from PnR point of view we need to follow some rules –

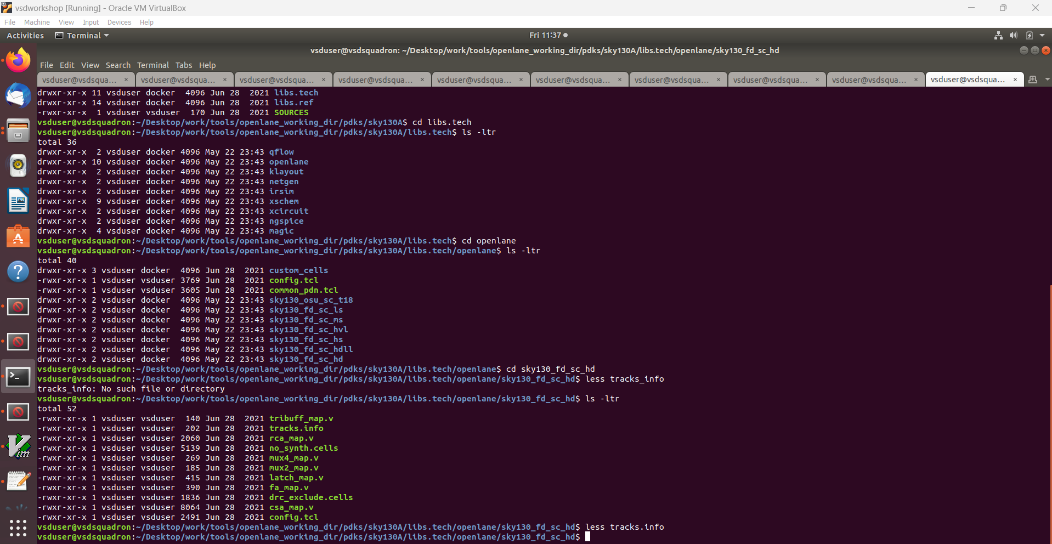
1. I/p and O/p port must lie on the intersection of horizontal and vertical track.
2. Width of standard cell = odd\*track pitch
3. Height of standard cell = odd \* track pitch

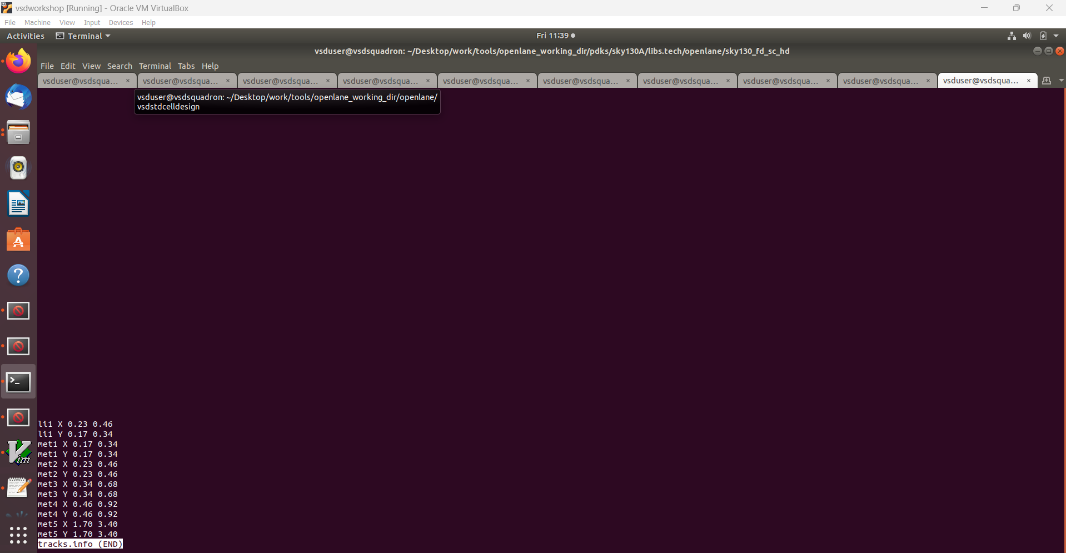
So we will see how accurate is the layout –

Now we will open the track file using below command –

**#track\_file**

pdk/sky130/libs.tech /openlane/sky130\_fd\_sc\_hd/track.info





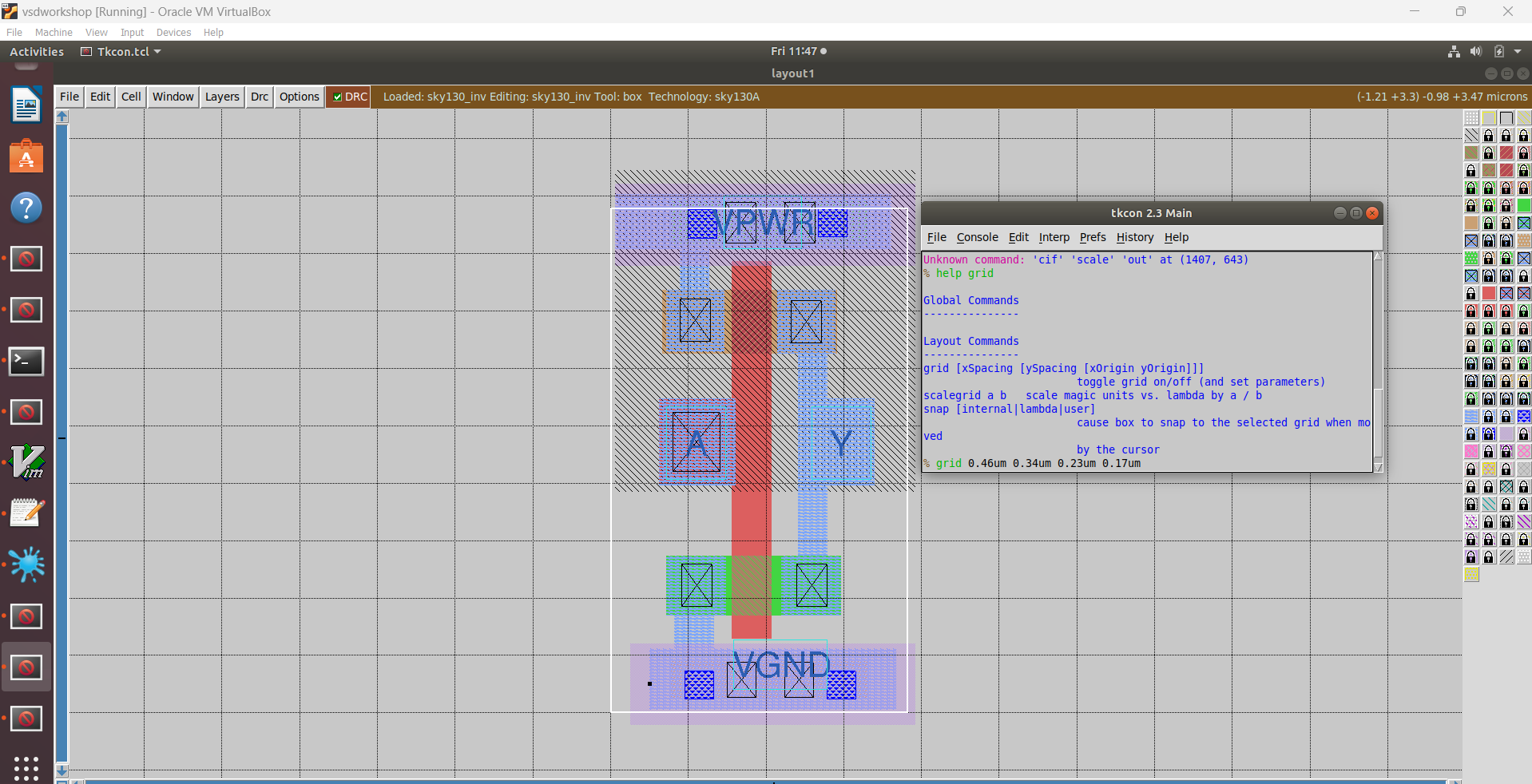
Tracks are used during the routing stage and serve as traces for metal layers such as Metal 1, Metal 2, etc.

Since Place and Route (PNR) is automated, we need to specify the desired routing paths using tracks. For the layers li1, Metal 1, and Metal 2, tracks should be placed at intervals of 0.23µm and 0.46µm horizontally, and 0.17µm and 0.34µm vertically.

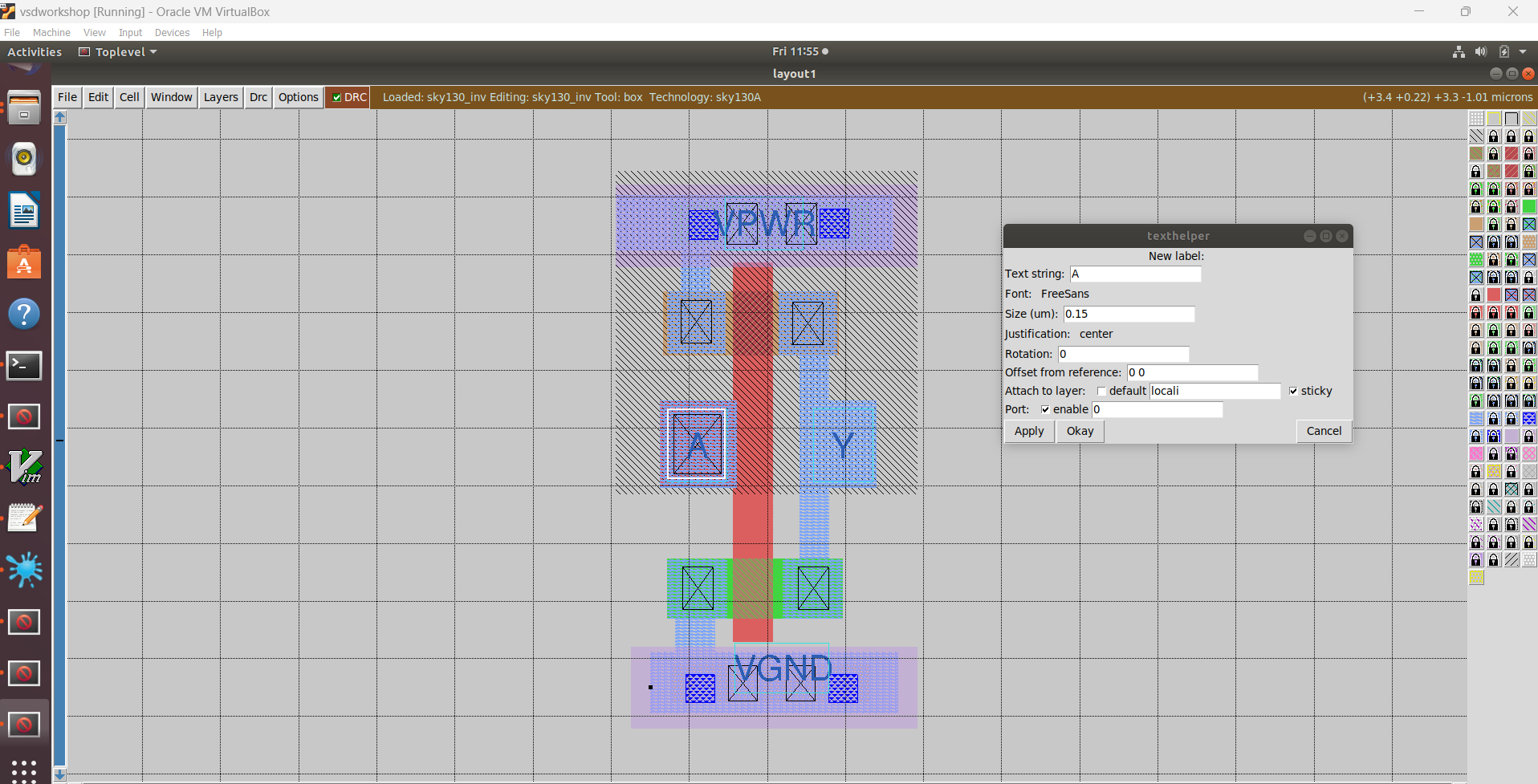
In the layout, the ports are located on the li1 layer. To ensure the ports align with the intersections of the tracks, we need to convert the grid into tracks.

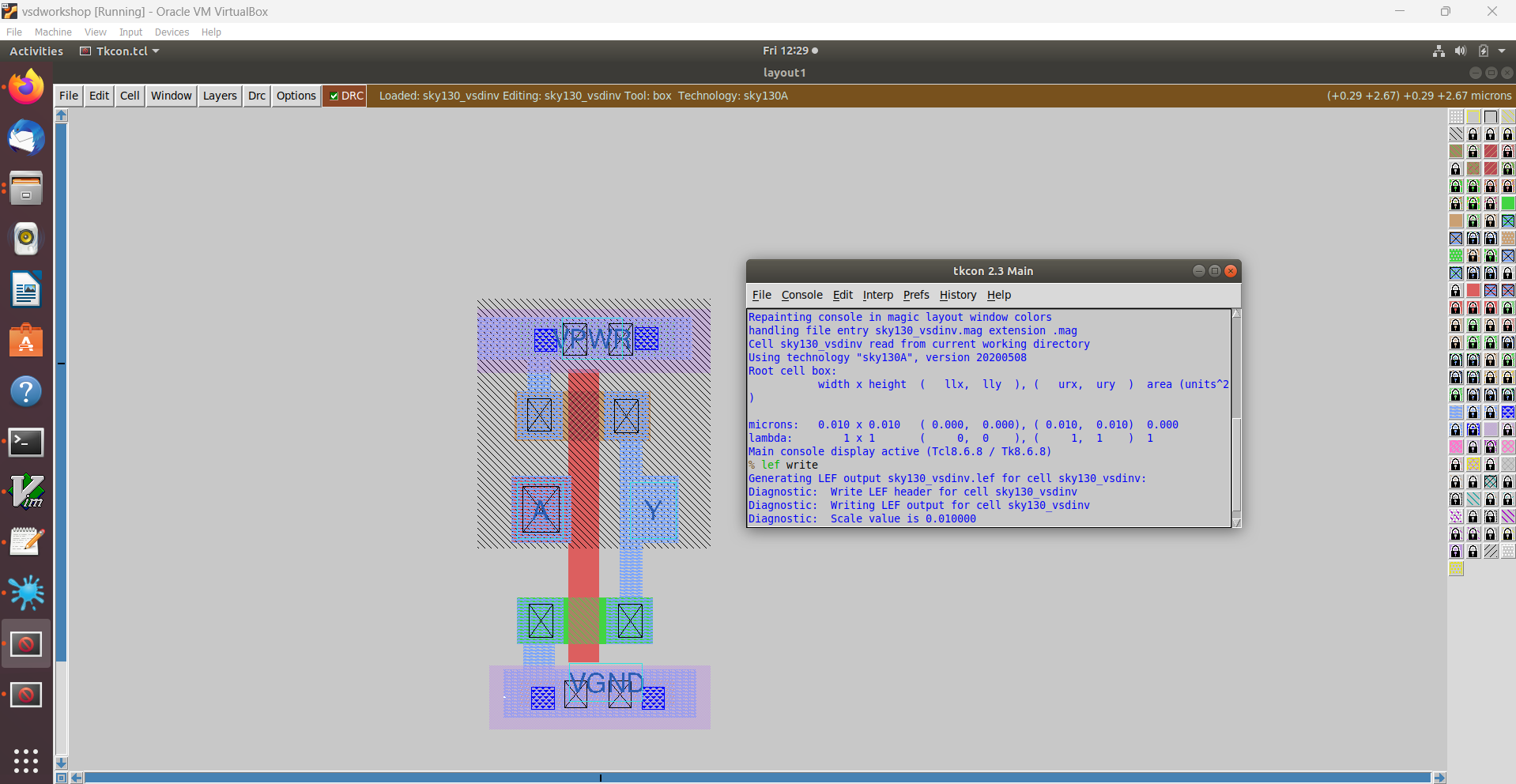
**# for grid generation in tkcon window we will use this command**

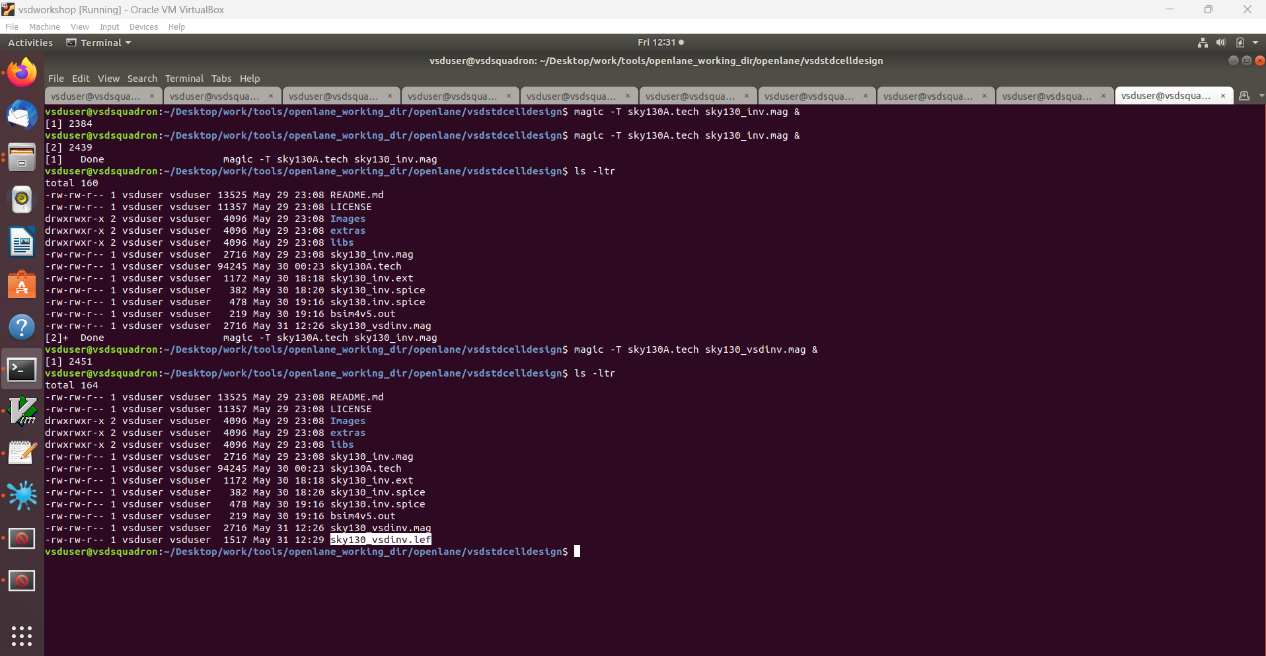
help grid

****

**Lab steps to convert magic layout to std cell LEF**

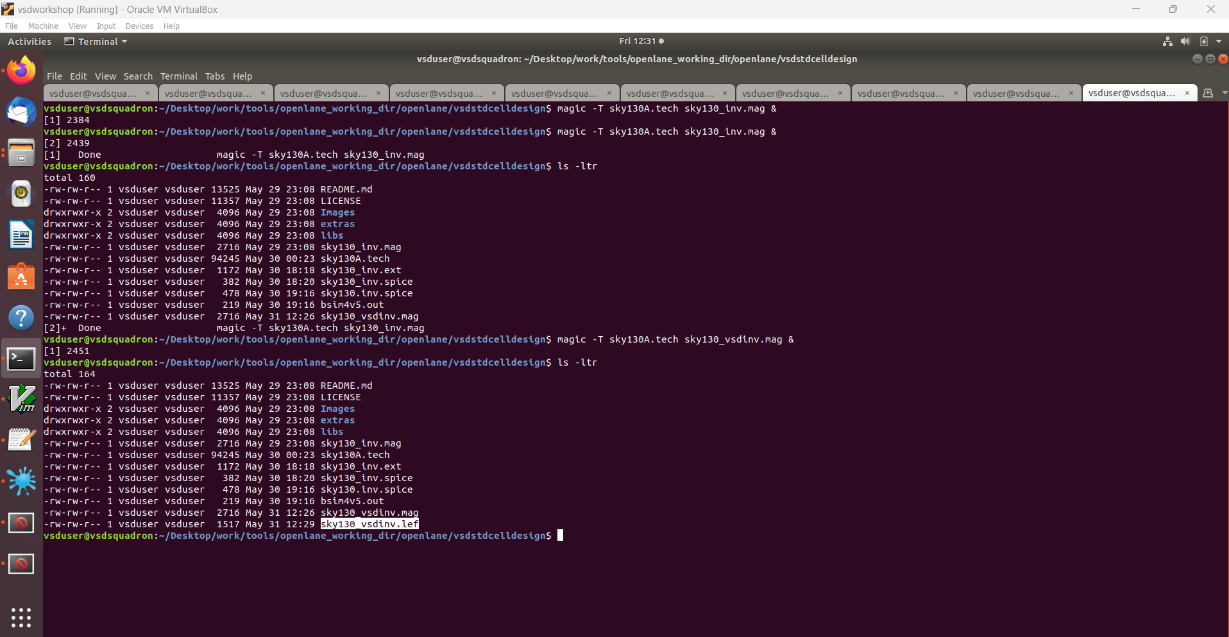
****Next, we need to determine the names and values for the ports. We can assign values to different ports, and for the power and ground ports, we must change the 'attach to layer' setting to Metal1.



Once these parameters are configured, we will be ready to extract the LEF file from the .mag file.

Now we will open this file in magic with the help of below command –

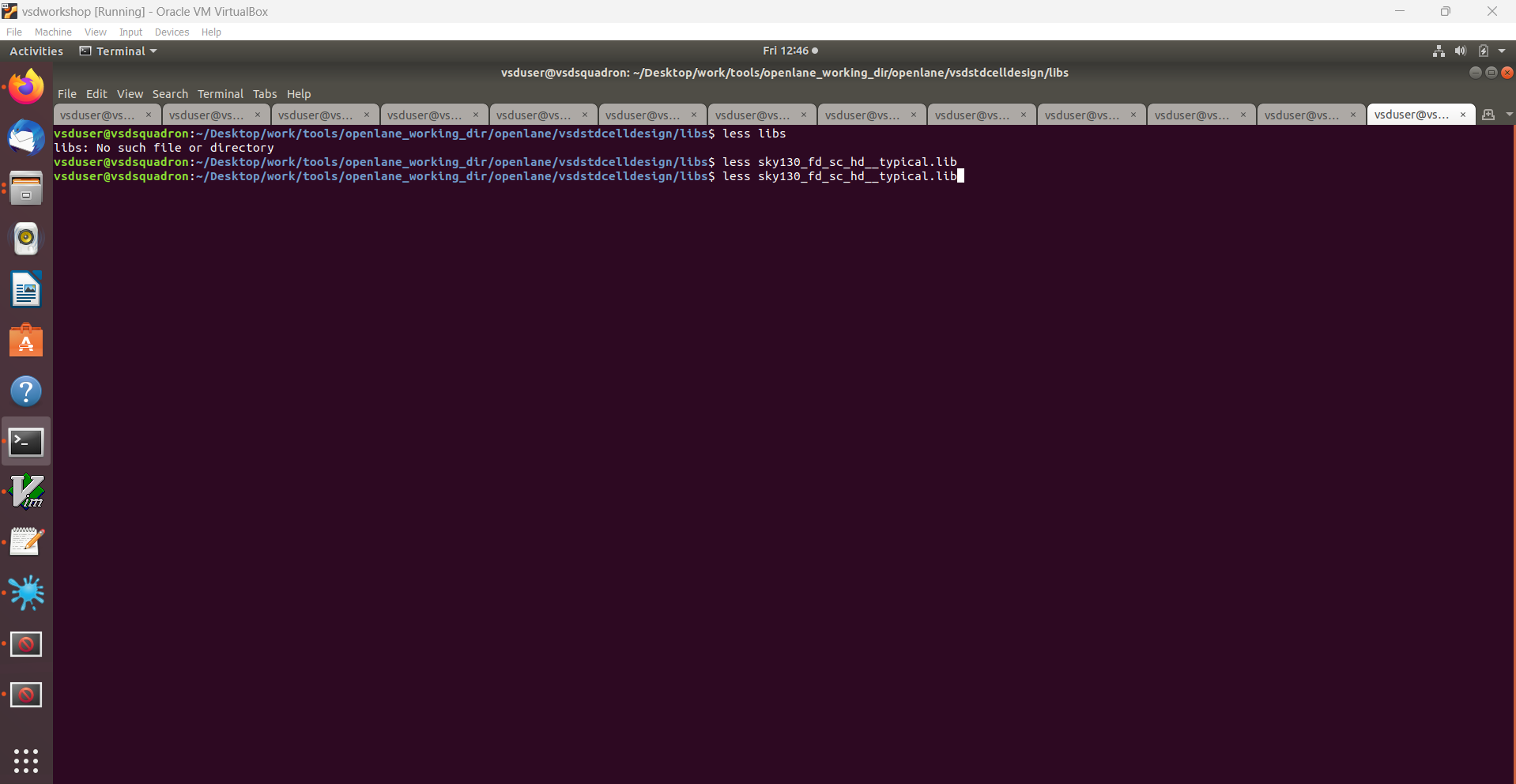
**#open lef file in magic using below command**

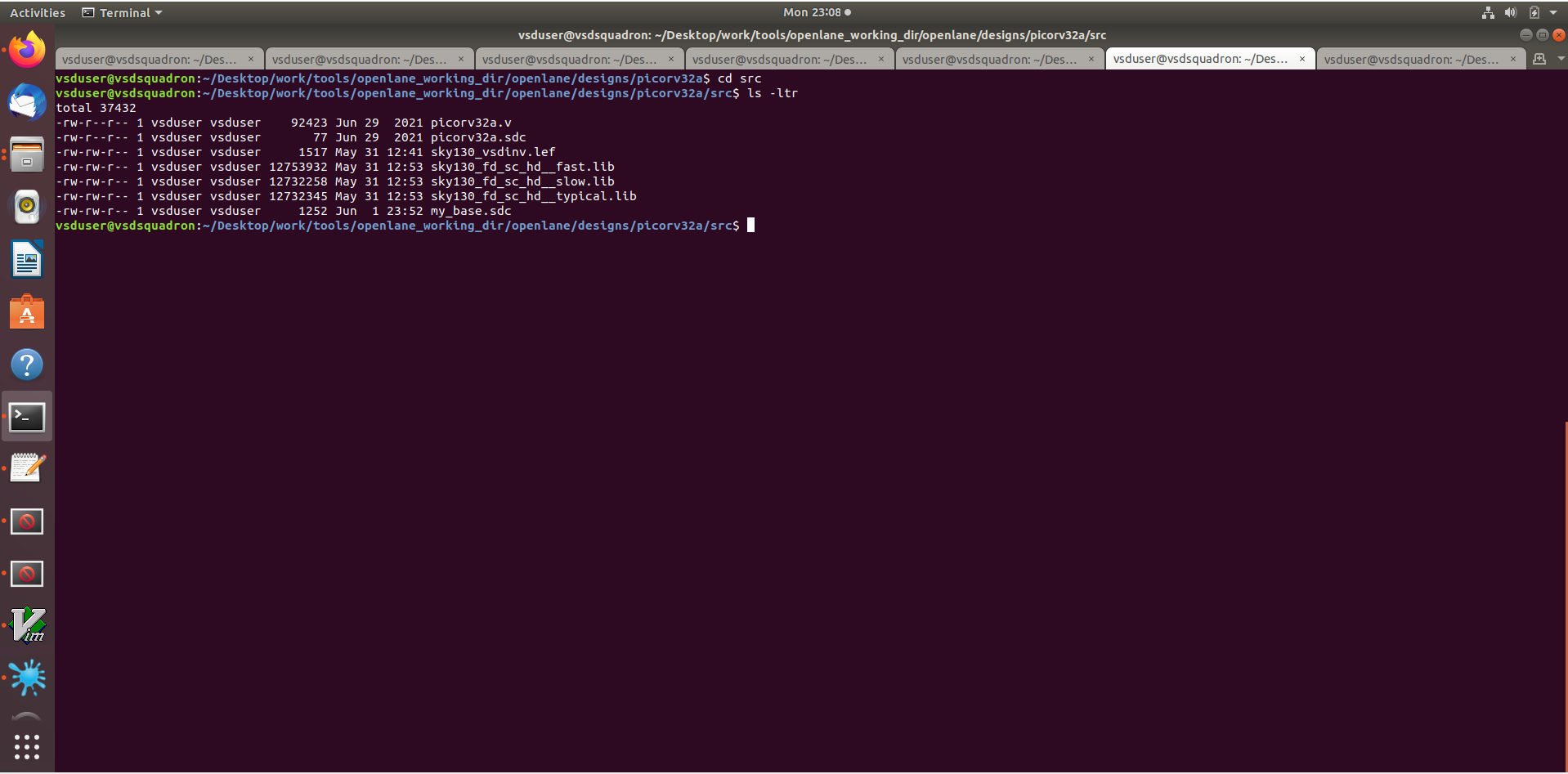
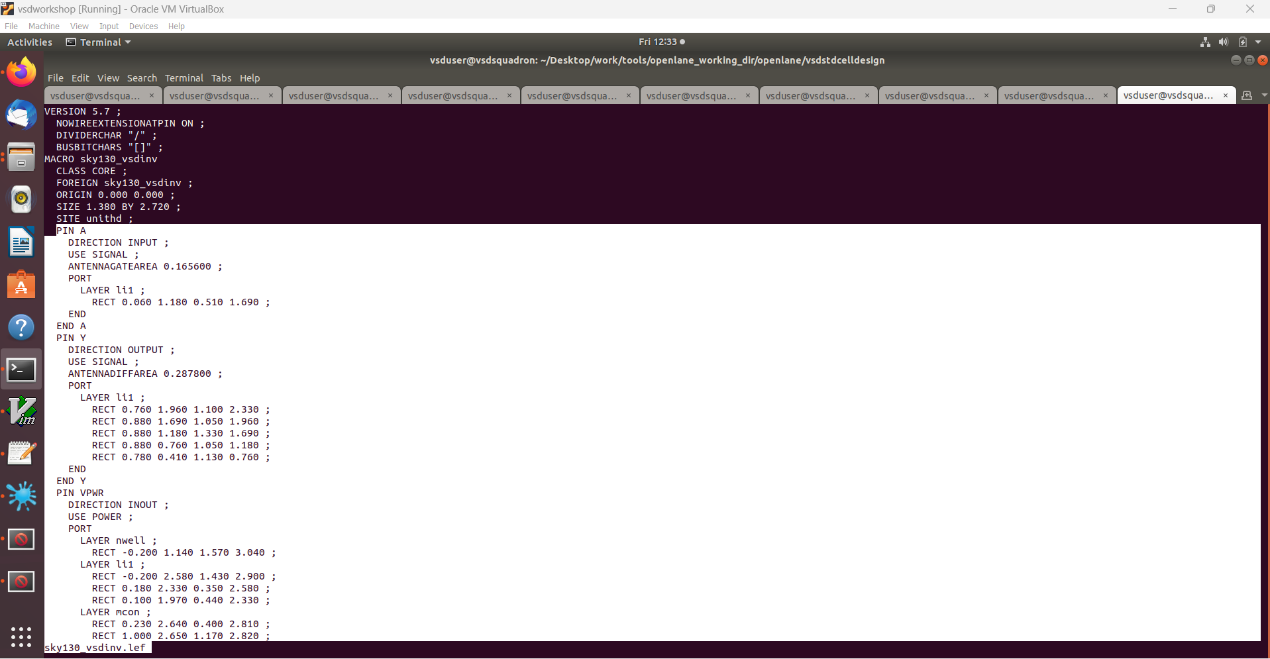
magic -T sky130A.tech sky130\_vsdinv.mag &

**Introduction to timing libs and steps to include new cell in synthesis**

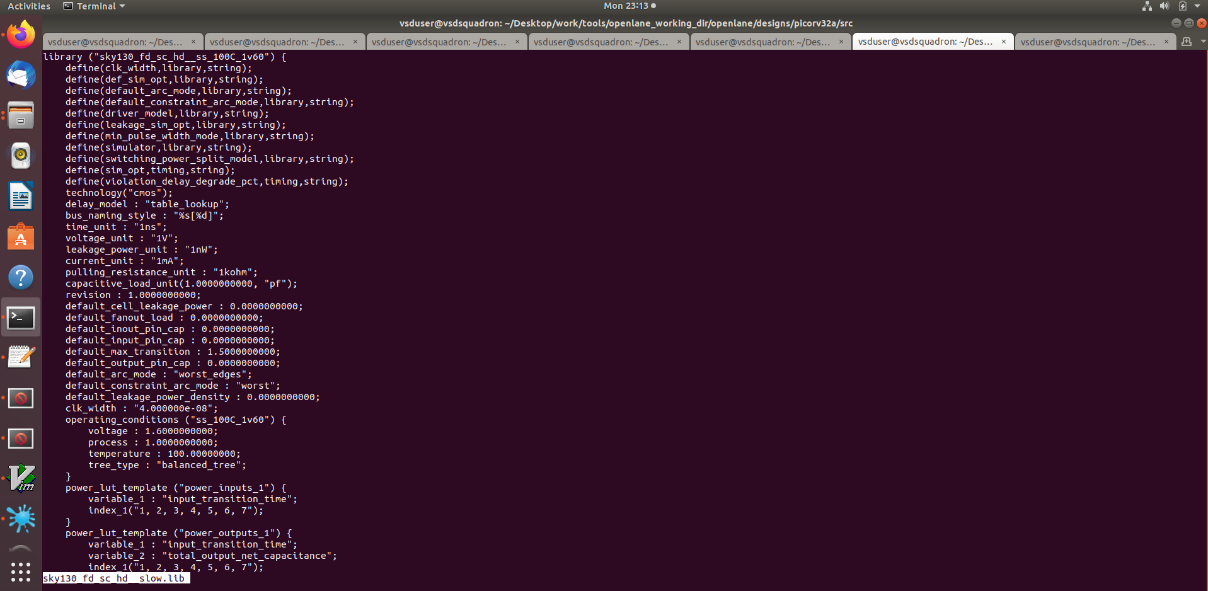
With the .lef file created, the next step is to integrate it into picorv32a. Before doing this, we need to move the files to the src folder, where all the design files are located.

To do this we can copy the file using command **cp**

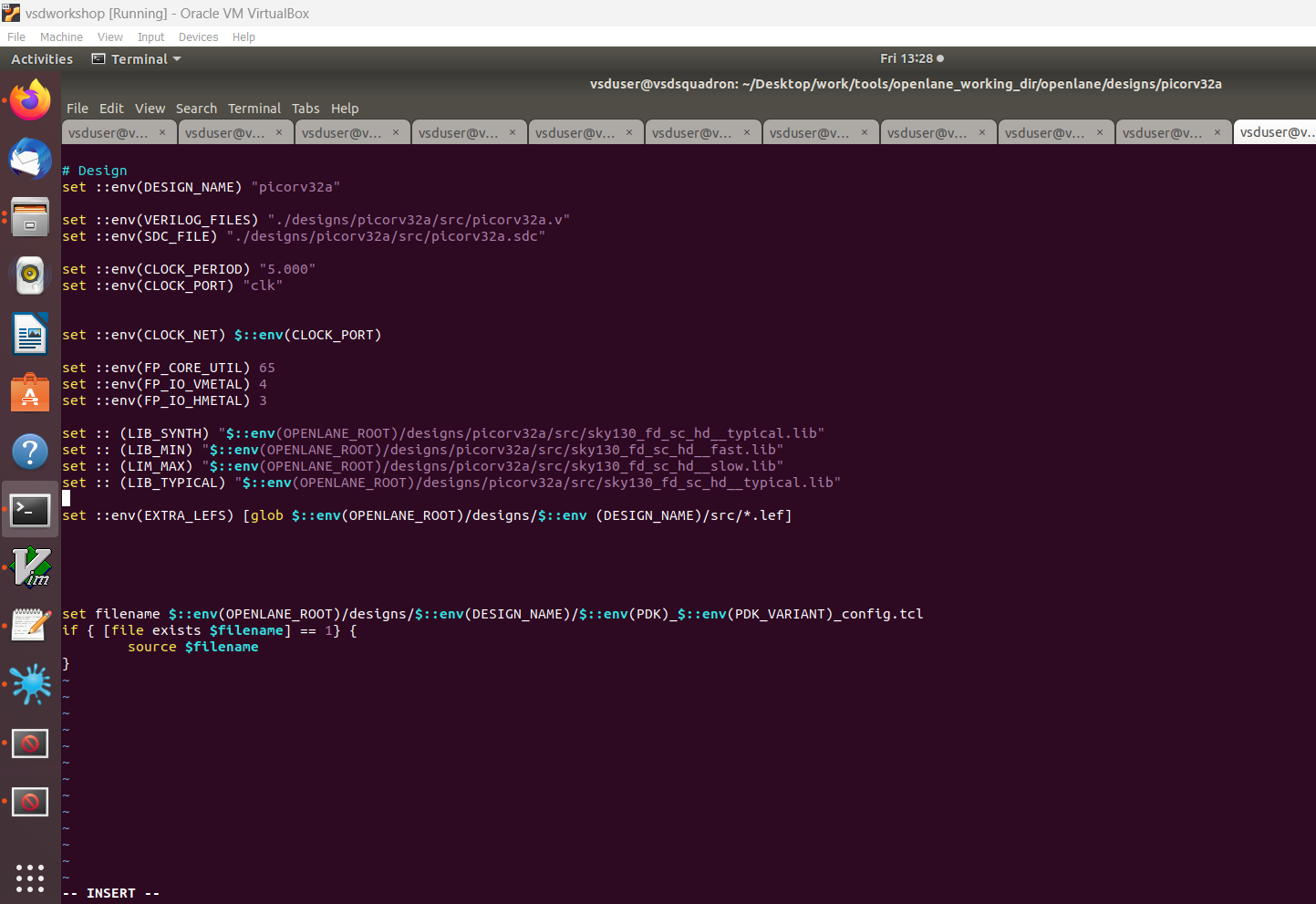
****



Now we have different library file names fast,slow and typical that we can see by using the below command –



To proceed, we'll need to edit the config.tcl file within the picorv32a directory.

Open the config.tcl file and insert the commands depicted in the image below.

**OPENLANE: Now we will go to the open lane directory and we will execute the docker command –**

./flow.tcl -interactive

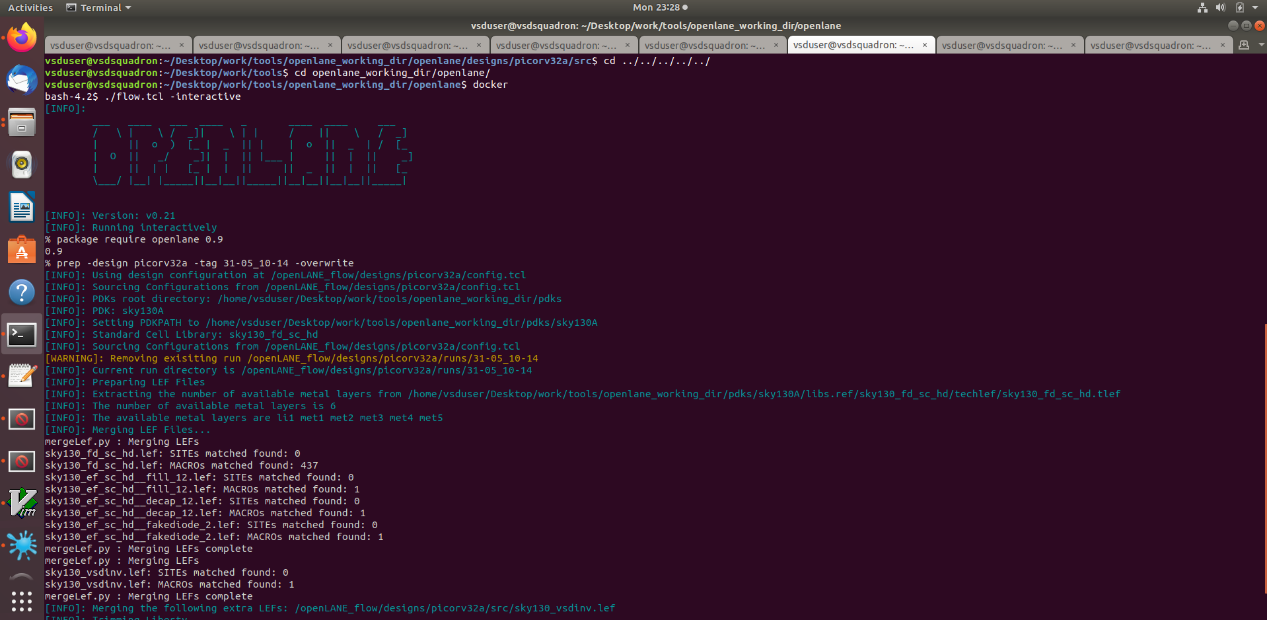
package require openlane 0.9

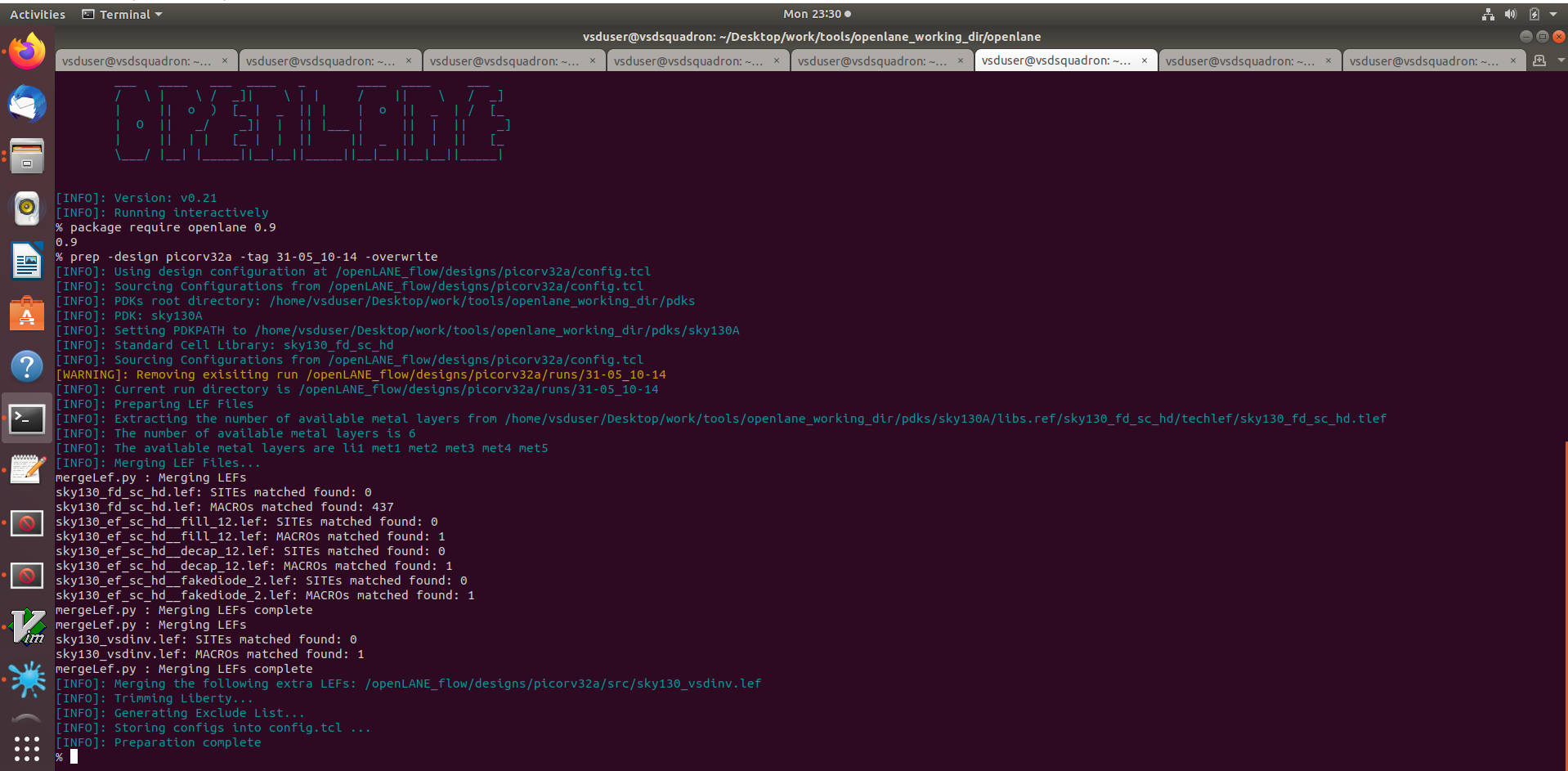
prep -design picorv32a -tag 31-05\_10-14 -overwrite

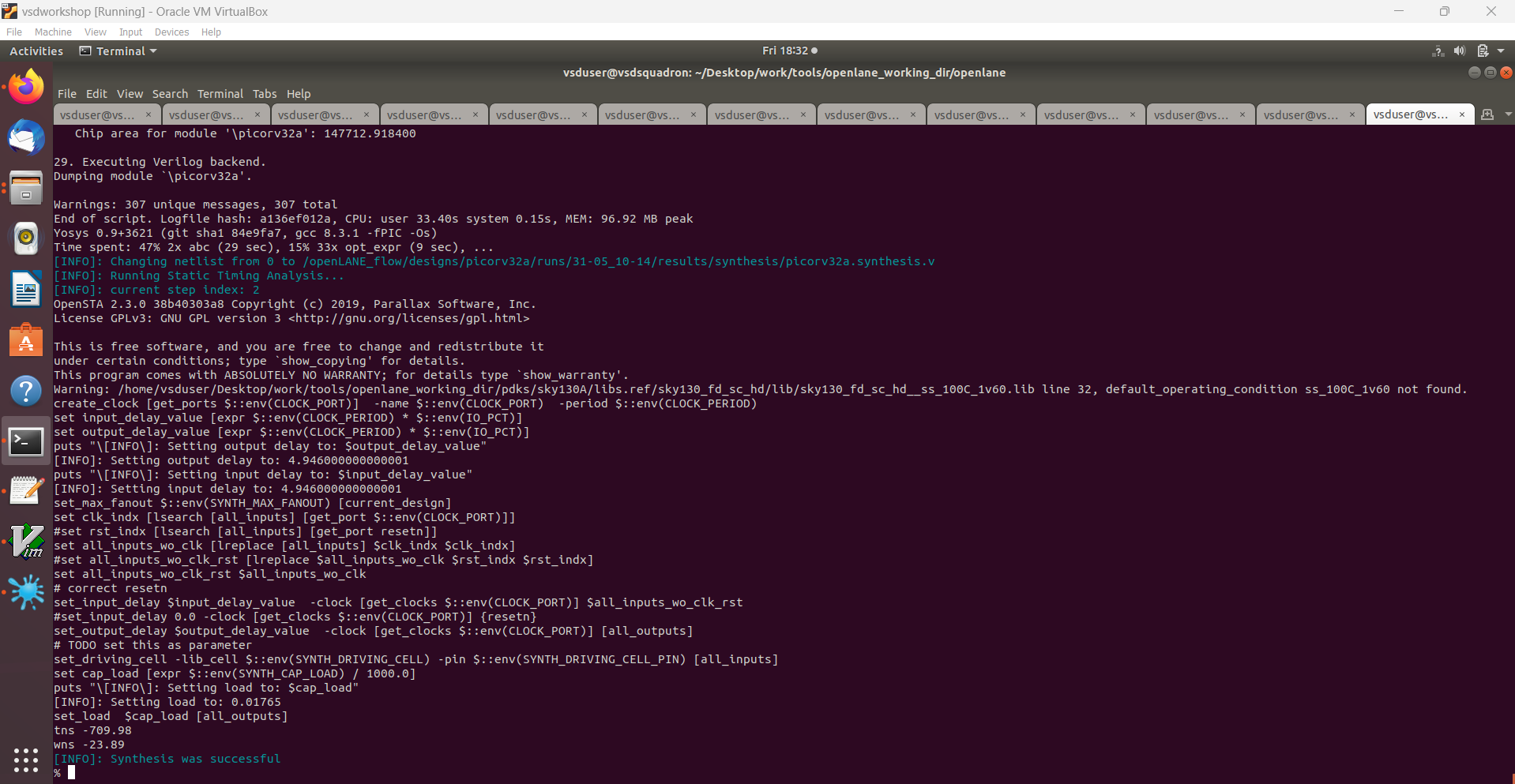
set lefs [glob $::env(DESIGN\_DIR)/src/\*.lef]

add\_lefs -src $lefs

run\_synthesis

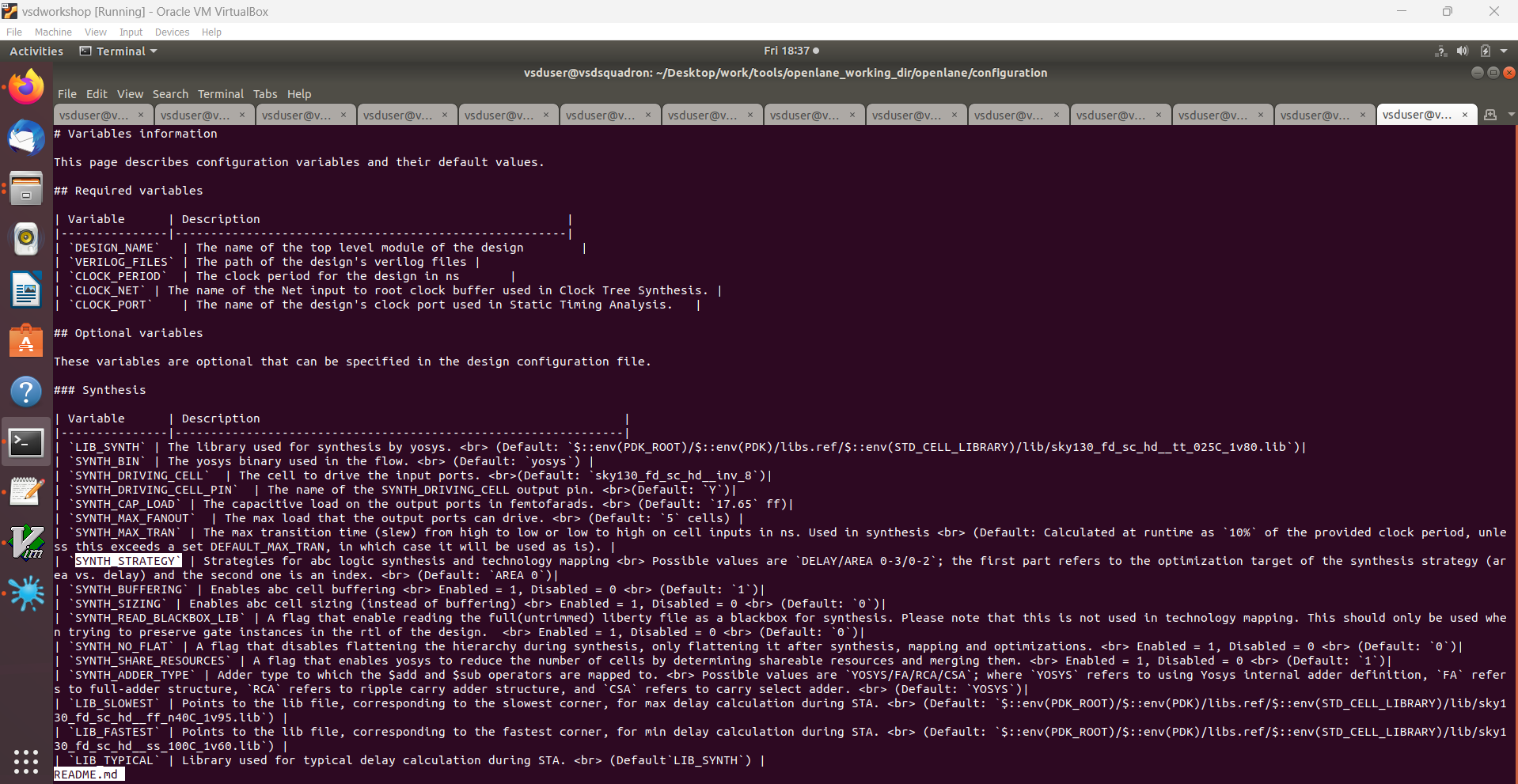
****





**Lab Steps to configure synthesis settings to fix slack and include vsdinv**

We'll attempt to adjust the parameters of our cell by consulting the README.md file located in the configuration folder within the OpenLANE directory. This README.md file provides details about the cell parameters.



Now we will give the certain commands to openlane directory after modification in README.md file –

prep -design picorv32a -tag 01-04\_12-54 -overwrite

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

add\_lefs -src $lefs

echo $::env(SYNTH\_STRATEGY)

set ::env(SYNTH\_STRATEGY) "DELAY 3"

echo $::env(SYNTH\_BUFFERING)

echo $::env(SYNTH\_SIZING)

set ::env(SYNTH\_SIZING) 1

echo $::env(SYNTH\_DRIVING\_CELL)

run\_synthesis

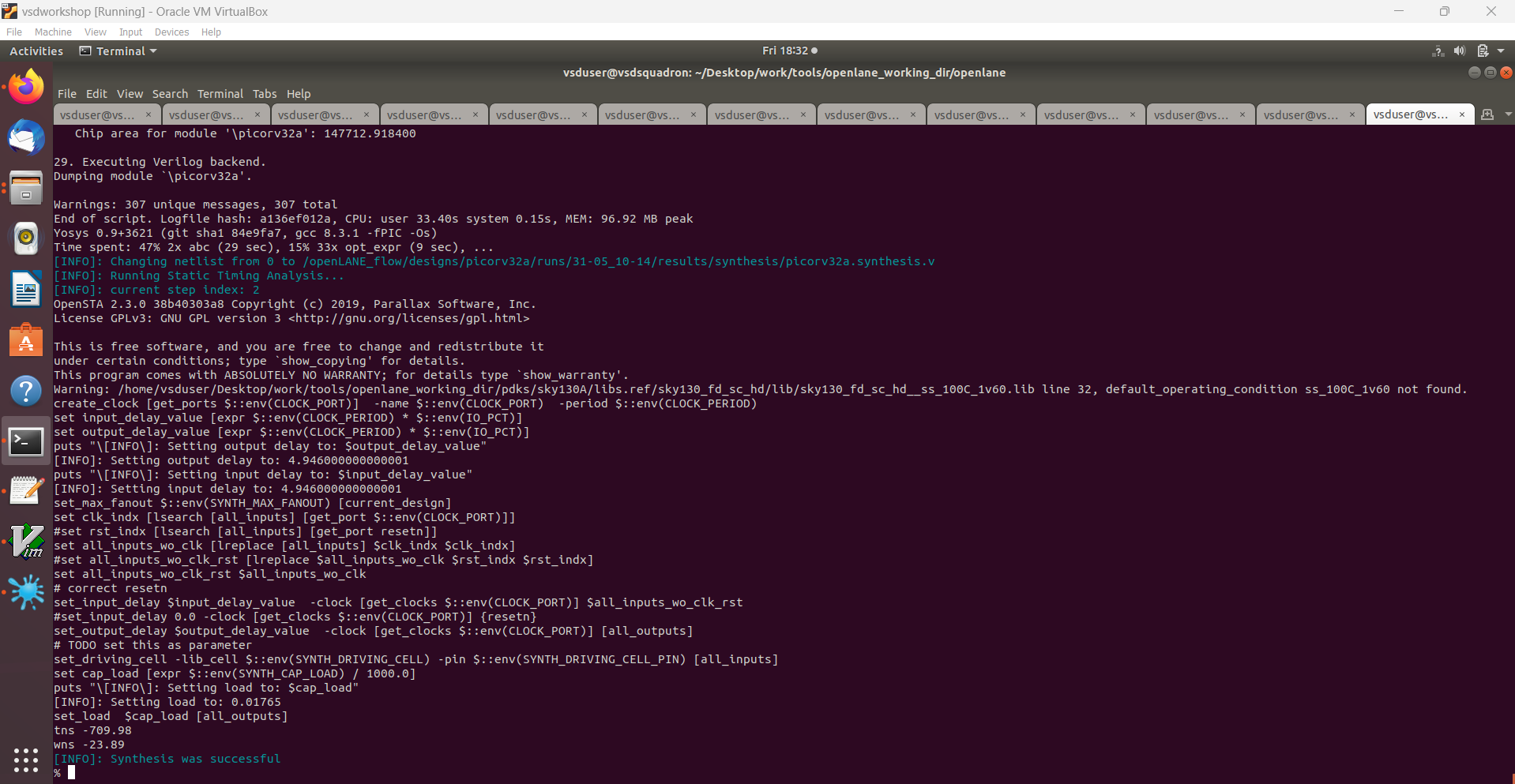
prep -design picorv32a -tag 01-04\_12-54 -overwrite

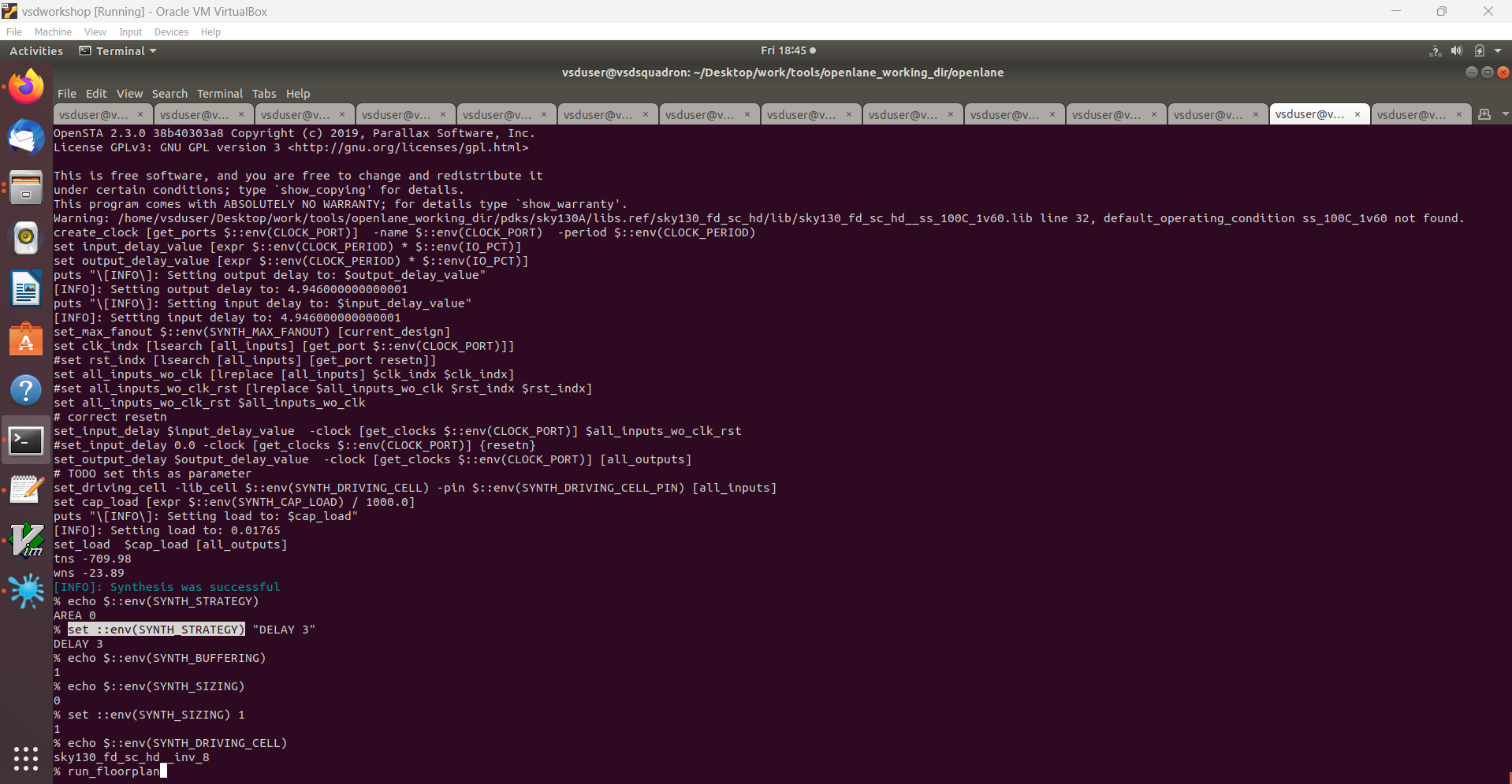
This is utilized to replace the current files with previous simulation values

After synthesis we got negative slack values for both wsn(worst negative slack) and tns(total negative slack). These values should be positive for proper functioning of our design.

wns(worst negative slack)= -23.89 ns

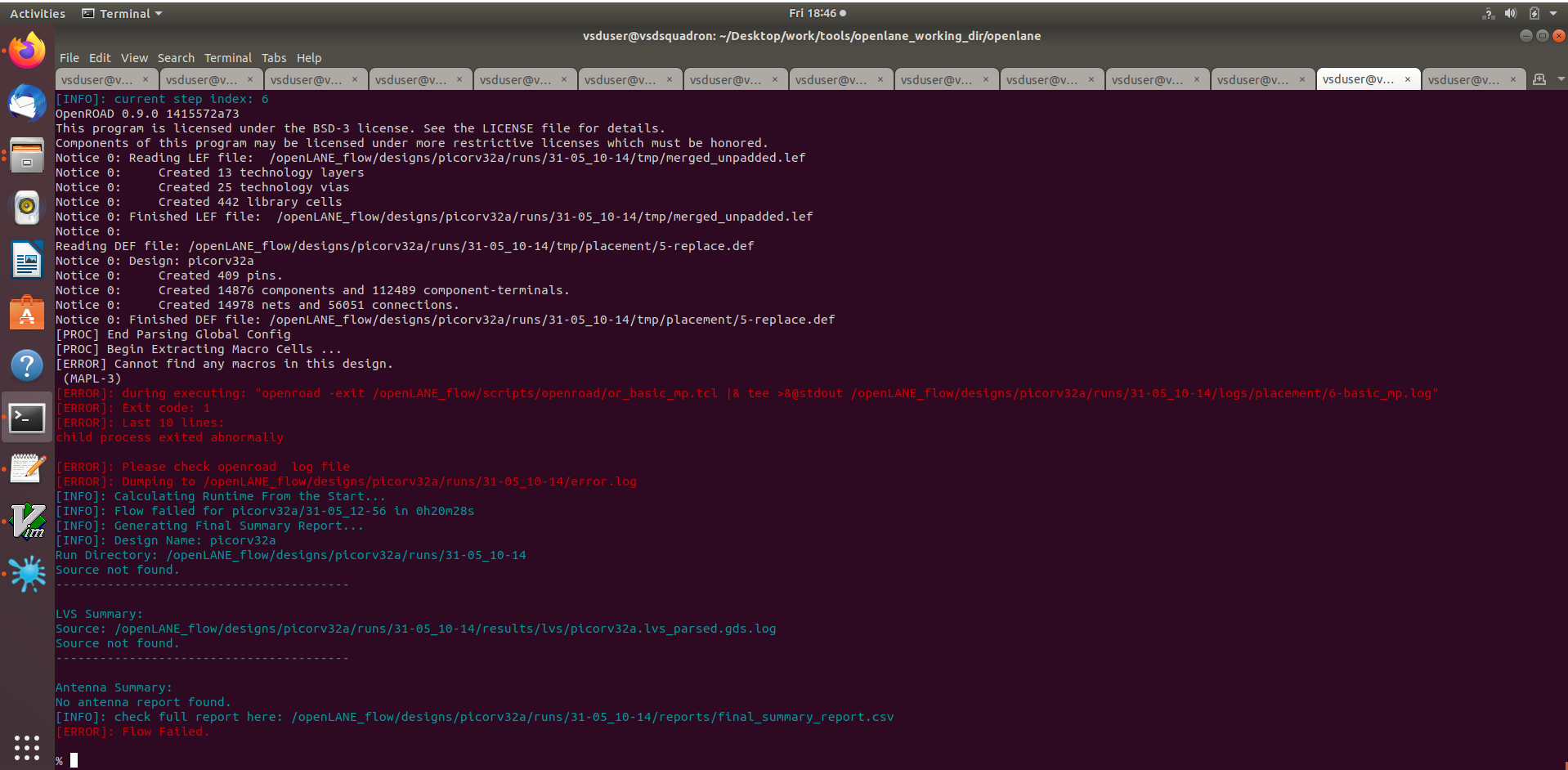
tns(total negative slack)= -709.98 ns





Upon executing run\_synthesis, we observe an increase in chip area and a reduction in the value of slack.

Given the successful completion of the synthesis for the picorv32a, we'll proceed to initiate the floorplan by running the command run\_floorplan.



Here we are getting errors so again we will open the docker and run synthesis using the command given above and after that we will follow the these commands and analyse the results-

init\_floorplan

place\_io

tap\_decap\_or

