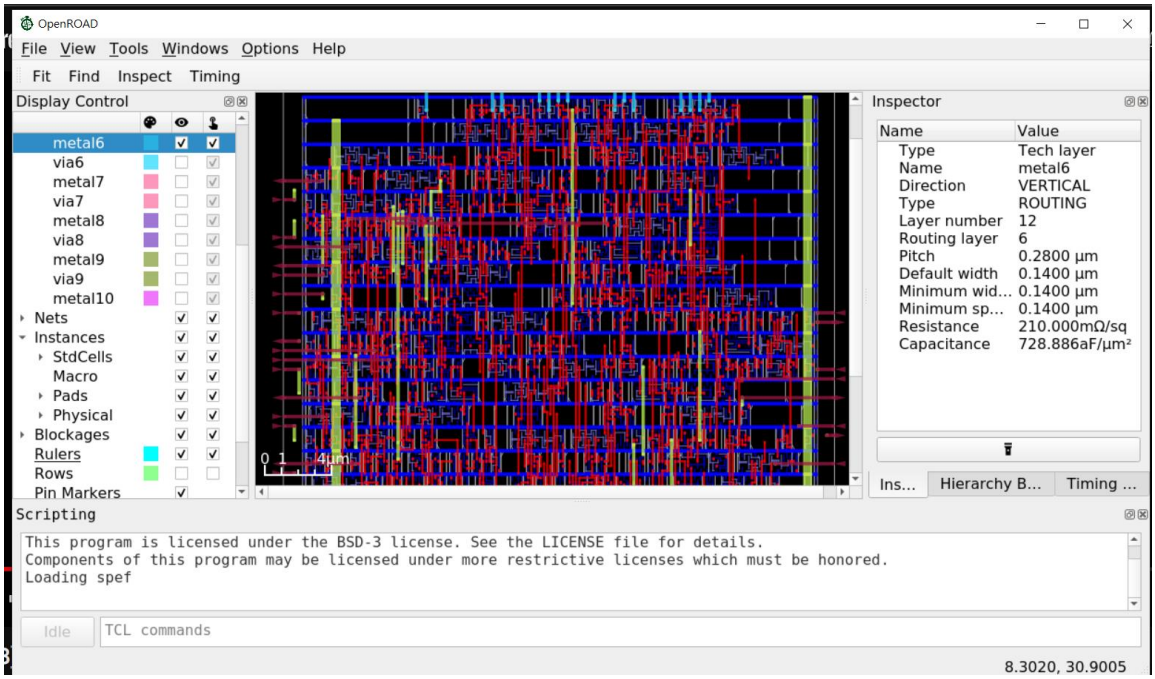
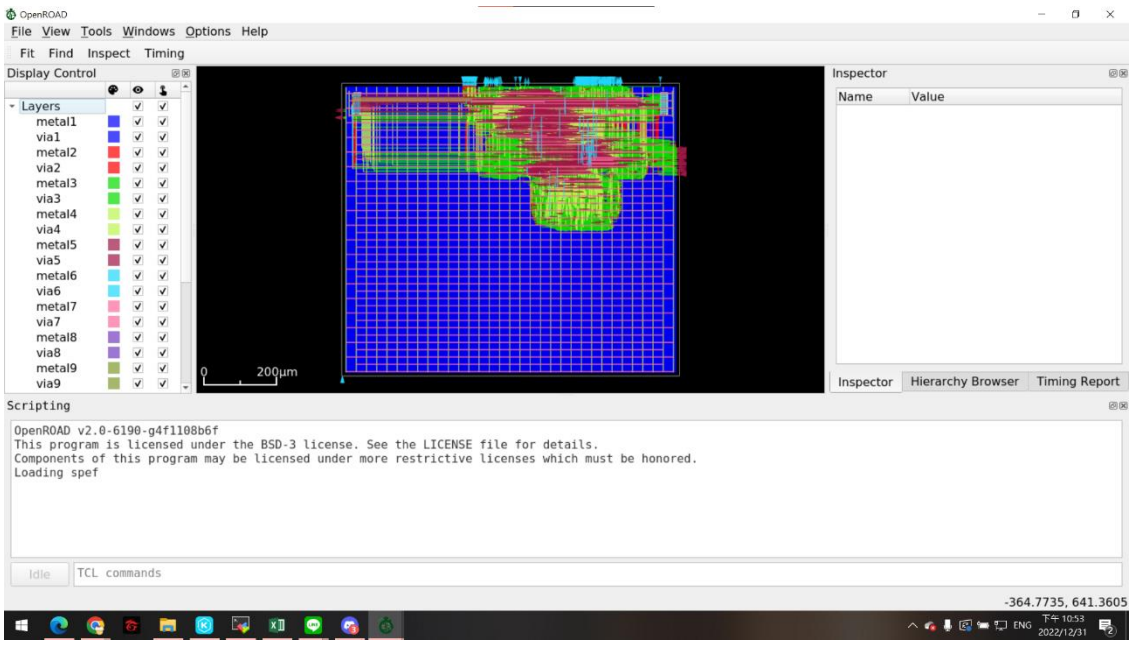
OpenROAD

STUDENT ID:B11015022

NAME:程帝鈞



圖表 一 GCD design



圖表 二 TinyRocket Design.png

```
b11015022@CDJ-IdeaPad:~/OpenROAD-flow-scripts/flow/designs/src/tinyRocket$ ls -all
total 10112
drwxr-xr-x  2 b11015022 b11015022     4096 Dec 31 14:53 .
drwxr-xr-x 27 b11015022 b11015022     4096 Dec 31 14:53 ..
-rw-r--r--  1 b11015022 b11015022     1722 Dec 31 14:53 AsyncResetReg.v
-rw-r--r--  1 b11015022 b11015022      654 Dec 31 14:53 ClockDivider2.v
-rw-r--r--  1 b11015022 b11015022      838 Dec 31 14:53 ClockDivider3.v
-rw-r--r--  1 b11015022 b11015022     1403 Dec 31 14:53 LICENSE.Berkeley
-rw-r--r--  1 b11015022 b11015022    11348 Dec 31 14:53 LICENSE.SiFive
-rw-r--r--  1 b11015022 b11015022     1483 Dec 31 14:53 LICENSE.jtag
-rw-r--r--  1 b11015022 b11015022      598 Dec 31 14:53 README.md
-rw-r--r--  1 b11015022 b11015022 10304063 Dec 31 14:53 freechips.rocketchip.system.TinyConfig.v
-rw-r--r--  1 b11015022 b11015022      553 Dec 31 14:53 plusarg_reader.v
```

圖表 三 TinyRocket list

```
b11015022@CDJ-IdeaPad:~/OpenROAD-flow-scripts/flow/designs/src/tinyRocket$ cat README.md
# Summary

RISC-V Rocket Chip design (TinyConfig).
Currently has about 30,000 cells.

# Source

Contributed by Ben Keller on Jun 25 2019.
Original source [available here](https://github.com/chipsalliance/rocket-chip).

# Modifications

- Use the default TinyConfig.
- Turn the IMem and DMem sizes down to minimum.
- Mapped some memory macros instead.
- Target RocketTile for synthesis, which is the actual processor (this avoids all the system bus overhead includ
ed in the top level).
- Added sdc timing constraints.
- Added LICENSE files from [here](https://github.com/chipsalliance/rocket-chip).
```

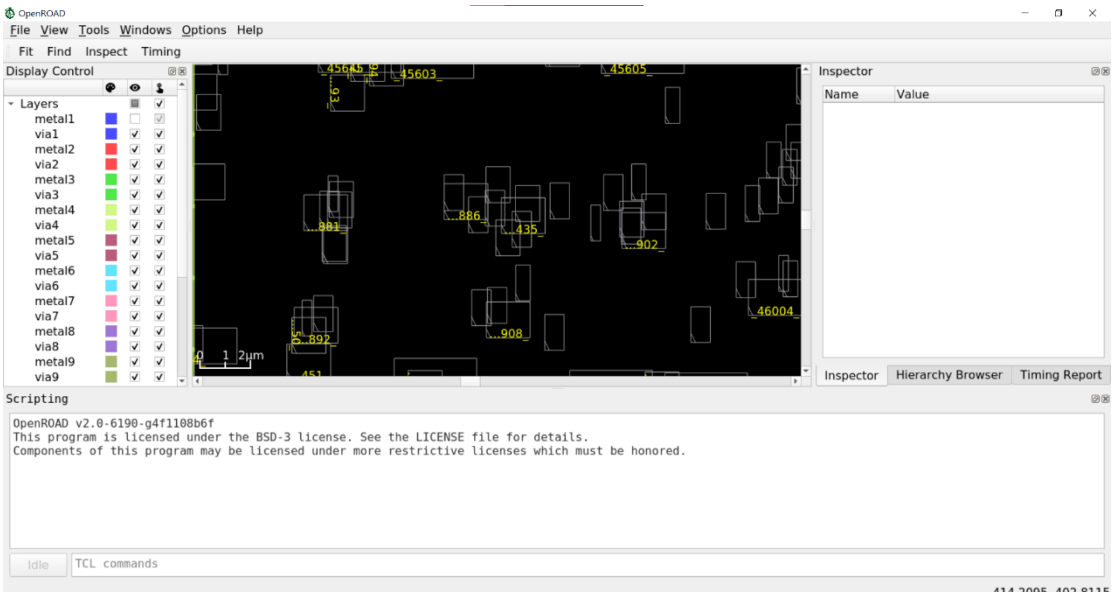
圖表 四 TinyRocket readme

```
  assign _T_983 = _T_978 < plusarg_reader_out; // @[Monitor.scala 479:60:freechips.rocketchip.system.TinyConfig.
fir@1497.4]
  assign _T_984 = _T_982 | _T_983; // @[Monitor.scala 479:48:freechips.rocketchip.system.TinyConfig.fir@1498.4]
  assign _T_986 = _T_984 | reset; // @[Monitor.scala 479:12:freechips.rocketchip.system.TinyConfig.fir@1500.4]
  assign _T_987 = _T_986 == 1'h0; // @[Monitor.scala 479:12:freechips.rocketchip.system.TinyConfig.fir@1501.4]
  assign _T_989 = _T_978 + 32'h1; // @[Monitor.scala 481:26:freechips.rocketchip.system.TinyConfig.fir@1507.4]
  assign _T_992 = _T_801 | _T_850; // @[Monitor.scala 482:27:freechips.rocketchip.system.TinyConfig.fir@1511.4]
  assign _GEN_19 = io_in_a_valid & _T_68; // @[Monitor.scala 49:14:freechips.rocketchip.system.TinyConfig.fir@16
6.10]
  assign _GEN_35 = io_in_a_valid & _T_142; // @[Monitor.scala 60:14:freechips.rocketchip.system.TinyConfig.fir@2
74.10]
  assign _GEN_53 = io_in_a_valid & _T_220; // @[Monitor.scala 72:14:freechips.rocketchip.system.TinyConfig.fir@3
98.10]
  assign _GEN_65 = io_in_a_valid & _T_294; // @[Monitor.scala 81:14:freechips.rocketchip.system.TinyConfig.fir@5
05.10]
  assign _GEN_75 = io_in_a_valid & _T_371; // @[Monitor.scala 89:14:freechips.rocketchip.system.TinyConfig.fir@6
04.10]
  assign _GEN_85 = io_in_a_valid & _T_450; // @[Monitor.scala 97:14:freechips.rocketchip.system.TinyConfig.fir@6
95.10]
  assign _GEN_95 = io_in_a_valid & _T_517; // @[Monitor.scala 105:14:freechips.rocketchip.system.TinyConfig.fir@
784.10]
  assign _GEN_105 = io_in_a_valid & _T_584; // @[Monitor.scala 113:14:freechips.rocketchip.system.TinyConfig.fir
@873.10]
  assign _GEN_115 = io_in_d_valid & _T_660; // @[Monitor.scala 276:14:freechips.rocketchip.system.TinyConfig.fir
@931.10]
  assign _GEN_125 = io_in_d_valid & _T_680; // @[Monitor.scala 284:14:freechips.rocketchip.system.TinyConfig.fir
@973.10]
  assign _GEN_137 = io_in_d_valid & _T_708; // @[Monitor.scala 294:14:freechips.rocketchip.system.TinyConfig.fir
@1031.10]
  assign _GEN_149 = io_in_d_valid & _T_737; // @[Monitor.scala 304:14:freechips.rocketchip.system.TinyConfig.fir
@1090.10]
  assign _GEN_155 = io_in_d_valid & _T_754; // @[Monitor.scala 312:14:freechips.rocketchip.system.TinyConfig.fir
@1125.10]
  assign _GEN_161 = io_in_d_valid & _T_772; // @[Monitor.scala 320:14:freechips.rocketchip.system.TinyConfig.fir
@1161.10]
`ifdef RANDOMIZE_GARBAGE_ASSIGN
`define RANDOMIZE
`endif
`ifdef RANDOMIZE_INVALID_ASSIGN
`define RANDOMIZE
                                                                625,1         0%
```

圖表 五 TinyRocket RTL code.png

```
b11015022@CDJ-IdeaPad:~/OpenROAD-flow-scripts/flow$ make
Display all 136 possibilities? (y or n)
2_1_floorplan.odb.def            deleteRoutingObstructions_issue  klayout_6_1_merged.gds
2_2_floorplan_io.odb.def         density_fill_issue               klayout_6_final.def
2_3_floorplan_tdms.odb.def       detail_place_issue               klayout_6_final.gds
2_4_floorplan_macro.odb.def      detail_route_issue               klayout_guides
2_5_floorplan_tapcell.odb.def    drc                              klayout_issue
2_6_floorplan_pdn.odb.def        fillcell_issue                   klayout_tr_rpt
2_floorplan.odb.def              final_report_issue               load_issue
3_1_place_gp_skip_io.odb.def     finish                           logs/
3_2_place_iop.odb.def            floorplan                        lvs
3_3_place_gp.odb.def             floorplan_issue                  macro_place_issue
3_4_place_resized.odb.def        gallery                          metadata
3_5_place_dp.odb.def             generate_abstract                nuke
3_place.odb.def                  generate_abstract_issue          objects/
4_1_cts.odb.def                  global_place_issue               pdn_issue
4_2_cts_fillcell.odb.def         global_place_skip_io_issue       place
4_cts.odb.def                    global_route_issue               placement_blockages_issue
5_1_grt.odb.def                  gui_2_1_floorplan.odb            read_liberty_issue
5_2_route.odb.def                gui_2_2_floorplan_io.odb         read_macro_placement_issue
5_route.odb.def                  gui_2_3_floorplan_tdms.odb       report_metrics_issue
6_1_fill.odb.def                 gui_2_4_floorplan_macro.odb      reports/
6_final.odb.def                  gui_2_5_floorplan_tapcell.odb    resize_issue
add_routing_blk_issue            gui_2_6_floorplan_pdn.odb        results/
all                              gui_2_floorplan.odb              route
all_defs                         gui_3_1_place_gp_skip_io.odb     run_all_issue
build_macros                     gui_3_2_place_iop.odb            run_test
cdl_issue                        gui_3_3_place_gp.odb             save_images_issue
clean                            gui_3_4_place_resized.odb        synth
clean_all                        gui_3_5_place_dp.odb             synth_hier_report_issue
clean_cts                        gui_3_place.odb                  synth_issue
clean_finish                     gui_4_1_cts.odb                  synth_preamble_issue
clean_floorplan                  gui_4_2_cts_fillcell.odb         tapcell_issue
clean_issues                     gui_4_cts.odb                    tdms_place_issue
clean_metadata                   gui_5_1_grt.odb                  update_metadata
clean_place                      gui_5_2_route.odb                update_ok
clean_resize                     gui_5_route.odb                  update_rules
clean_route                      gui_6_1_fill.odb                 update_rules_force
clean_synth                      gui_6_final.def                  update_sdc_clocks
clean_test                       gui_6_final.odb                  versions.txt
command                          gui_cts                          view_cells
convert_rve                      gui_final                        view_cells_issue
```

圖表 六執行清單.png



通過這次專案我對於設計的順序更有概念，跟著老師跑一套完整的流程，讓我對每個階段的工作有更深入的理解，雖然現在用演算法所產出作品的 PPA 不是特別亮眼，但在深度學習的成熟的時代，我覺得這些不足以成為未來發展的瓶頸，如這陣子熱門的 chatGPT，充分的表現出現階段演算法能達到的高度已經超過大家的預期，或許在不遠的將來，我們能看到由電腦設計出更完美、更符合大眾需求的數位電路。這次學習，讓我對迷茫的未來逐漸有些構圖。

這學期課程非常扎實，前三個作業讓我個別製作小專案能力提升不少。我覺得老師在未來可以將課程內容密度降低，然後將作業數量增加，期限縮短，讓學生練習次數增加，透過不斷實際演練，讓對數位系統設計學習有更有透徹的理解。