NCTU-EE IC Design LAB – Fall2023

Final Project: Innovus - From RTL to GDSII

Due to the main goal of this course is to teach students on the front-end design methodology, the APR of final project will be modified to pin design rather than pad design so it's easier.

1. Data Preparation

1. Complete rtl simulation, synthesis, and gate level simulation.
2. **cd ./05_APR/ write and check your CHIP.sdc file.**

   (Hint: you may modify the file generated during synthesize in ../02_SYN/Netlist folder, and carefully check the constraints following the check_list.pdf document.)
3. **cp CPU_SYN.v CHIP_SYN.v** and change the **module name** to **CHIP**.

   (Because the design does not require any pad connection, you won't need to write your own shell file, nor do you need to execute ./00_combine.)
4. You don't need to write the CHIP.io file when you run Innovus at the first time. (The import design step should use CHIP_SYN.v and leave the CHIP.io part empty.)

   Notice that TA has provided 05_APR/mmmc.view for you, please do not create mmmc.view by yourself, or you may fail in demo.

   Please modify the mmmc.view file to include memory lib file:

   You should include all your memory you used if you have multiple memory.

create_library_set -name lib_max -timing "XX ../04_MEM/SRAM_XXX_WC.lib" …
create_library_set-name lib_min -timing "XX ../04_MEM/SRAM_XXX_BC.lib" …

```
# Version:1.0 MMMC View Definition File
# Do Not Remove Above Line
# ============================================================
# Project:  iclab APR mmmc configure
# File:     mmmc.tcl
# Author:   Lai Lin-Hung @ Si2 Lab
# Date:     2023.07.25
# ============================================================

set ProcessRoot "/RAID2/COURSE/BackUp/2023_Spring/iclab/iclabta01/UMC018_CBDK/CIC/SOCE/"
set MemoryRoot "../Memory/ftclib_200901.2.1/EXE/"
set enc_source_continue_on_error true
create_rc_corner -name RC_Corner -cap_table $ProcessRoot/u18_Faraday.CapTbl -qx_tech_file $ProcessRoot/FireIce/icecaps.tch
create_library_set -name lib_max -timing "$ProcessRoot/lib/fsa0m_a_generic_core_ss1p62v125c.lib $ProcessRoot/lib/fsa0m_a_t33_generic_io_ss1p62v125c.lib ../04_MEM/SRAM_XXX_WC.lib" -si $ProcessRoot/celtic/u18_ss.cdb
create_library_set -name lib_min -timing "$ProcessRoot/lib/fsa0m_a_generic_core_ff1p98vm40c.lib $ProcessRoot/lib/fsa0m_a_t33_generic_io_ff1p98vm40c.lib ../04_MEM/SRAM_XXX_BC.lib" -si $ProcessRoot/celtic/u18_ff.cdb
create_constraint_mode -name func_mode -sdc_files CHIP.sdc
create_delay_corner -name Delay_Corner_max -library_set lib_max -rc_corner RC_Corner
create_delay_corner -name Delay_Corner_min -library_set lib_min -rc_corner RC_Corner
create_analysis_view -name av_func_mode_max -constraint_mode func_mode -delay_corner Delay_Corner_max
create_analysis_view -name av_func_mode_min -constraint_mode func_mode -delay_corner Delay_Corner_min
set_analysis_view -setup av_func_mode_max -hold av_func_mode_min
```

5. Open file: 05_APR/cmd/run_apr.cmd
   This file sets up the initial design environment. You can use these commands as an alternative to performing actions step-by-step via the GUI. However, the GUI option remains available if you find it more convenient.

```
# source cmd/floorPlan.cmd
# source cmd/powerRing.cmd
# source cmd/powerStripe.cmd
# source cmd/addIOFiller.cmd
# source cmd/place.cmd
# source cmd/ccopt.cmd
# source cmd/postCTSTiming.cmd
# source cmd/nanoRoute.cmd
# source cmd/postRouteTiming.cmd
# source cmd/postRouteVerify.cmd
# source cmd/addFiller.cmd
# source cmd/signOff.cmd
# source cmd/streamOut.cmd
```

   Notice that please add your memory LEF file PATH at the end of lefFile variable.

   For example:
   add **../04_MEM/XXXX.lef** beyond the $ProcessRoot/lef/BONDPAD.lef

```
# ======================================================
# Project:  iclab APR Flow
# File:     run_apr.cmd
# Author:   Lai, Lin-Hung @ Si2 Lab
# Date:     2023.07.25
# ======================================================
##########################################
############  Paremeter setting  ###########
##########################################
set ProcessRoot "/RAID2/COURSE/BackUp/2023_Spring/iclab/iclabta01/UMC018_CBDK/CIC/SOCE/"
set MemoryRoot "../Memory/ftclib_200901.2.1/EXE/"
set NUM_OF_CPU 48
set mmmcFile "mmmc.view"
set lefFile "
    $ProcessRoot/lef/header6_V55_20ka_cic.lef
    $ProcessRoot/lef/fsa0m_a_generic_core.lef
    $ProcessRoot/lef/FSA0M_A_GENERIC_CORE_ANT_V55.lef
    $ProcessRoot/lef/fsa0m_a_t33_generic_io.lef
    $ProcessRoot/lef/FSA0M_A_T33_GENERIC_IO_ANT_V55.lef
    $ProcessRoot/lef/BONDPAD.lef
"
set topDesign "CHIP"
set verilogFile "./CHIP_SYN.v"
set ioFile "./CHIP.io"
set pwrNet "VCC"
set gndNet "GND"
```

   Comment this line for the first time without CHIP.io file:
   set init_io_file $ioFile

```
##########################################
############  Initial Design     ###########
##########################################
set init_mmmc_file $mmmcFile
set init_lef_file $lefFile
set init_verilog $verilogFile
set init_top_cell $topDesign
#set init_io_file $ioFile
set init_pwr_net $pwrNet
set init_gnd_net $gndNet
init_design -setup {av_func_mode_max} -hold {av_func_mode_min}

save_global CHIP.globals
win
```

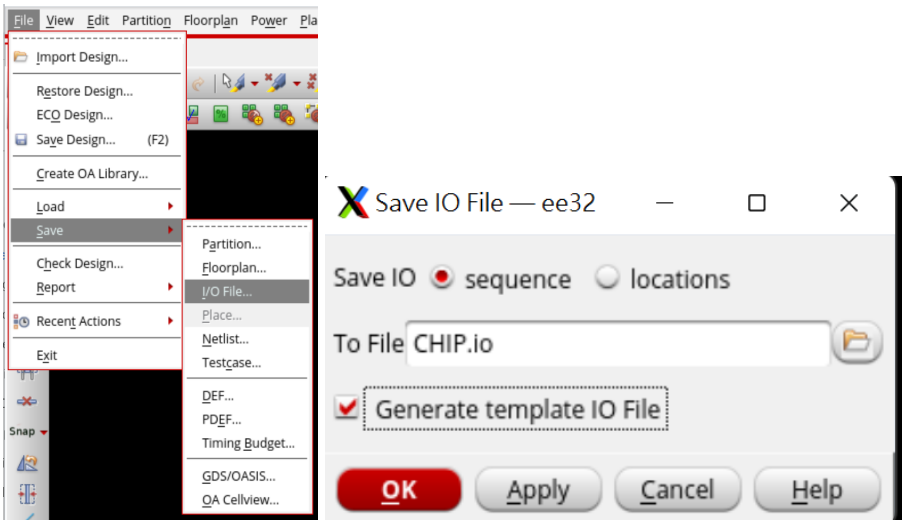
6. Launch innovus by the following command:

   innovus -cpus 8 -init ./cmd/run_apr.cmd -log ./log/run_apr.log -cmd ./log/run_apr.cmdlog

   or
   make apr_init

   you may find pin are gathering together due to no CHIP.io provided, thus you can do step 7 to auto generate this io file.

7. Upper left corner: FILE -> SAVE -> IO FILE ->
   sequence check & generate template IO file check
   change the file name to CHIP.io
   and click OK.

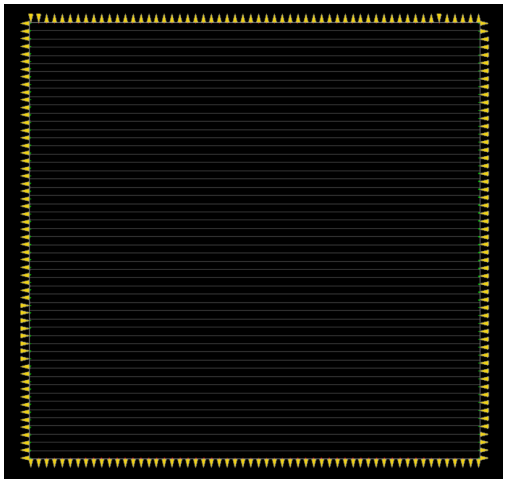


   Then exit
   Uncomment this line for the first time without CHIP.io file:

set init_io_file $ioFile

```
##############################################
############  Initial Design     ############
##############################################
set init_mmmc_file $mmmcFile
set init_lef_file $lefFile
set init_verilog $verilogFile
set init_top_cell $topDesign
set init_io_file $ioFile
set init_pwr_net $pwrNet
set init_gnd_net $gndNet
init_design -setup {av_func_mode_max} -hold {av_func_mode_min}
```

Restart innovus again. You may see the pin shape below.



8. You may change the orders in CHIP.io file if you wish to modify your pin orders.

```
#############################################################
#  Generated by:      Cadence Innovus 20.15-s105_1
#  OS:                Linux x86_64(Host ID ee31)
#  Generated on:      Mon Dec 25 21:08:16 2023
#  Design:            CHIP
#  Command:           saveIoFile -byOrder -temp CHIP.save.io
#############################################################

(globals
    version = 3
    io_order = default
)
(iopin
    (top
    (pin name="clk"  layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="rst_n"    layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="IO_stall"     layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awid_m_inf[3]"    layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awid_m_inf[2]"    layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awid_m_inf[1]"    layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awid_m_inf[0]"    layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awaddr_m_inf[31]"     layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awaddr_m_inf[30]"     layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awaddr_m_inf[29]"     layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awaddr_m_inf[28]"     layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awaddr_m_inf[27]"     layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awaddr_m_inf[26]"     layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awaddr_m_inf[25]"     layer=2 width=0.2800 depth=0.7000 place_status=placed  )
    (pin name="awaddr_m_inf[24]"     layer=2 width=0.2800 depth=0.7000 place_status=placed  )
```

9. After that, most of the steps are the same as Lab11 and Lab12. The only

difference is that there are no pads, so you should skip the step **(Add PAD Filler)**.

2. SDC FILE

SDC file CHIP.sdc

This file is generated from 02_SYN.
You may change this to your target cycle time by yourself.
Normally, this may be equal or larger than 02_SYN due to wiring delay.

```
create_clock [get_ports clk]  -period 20  -waveform {0 10}
set_clock_uncertainty 0.1  [get_clocks clk]
set_input_delay -clock clk  0  [get_ports clk]
```

Before CTS stage, you may no need to comment out the set_clock_uncertainty & set_clock_transition instruction.

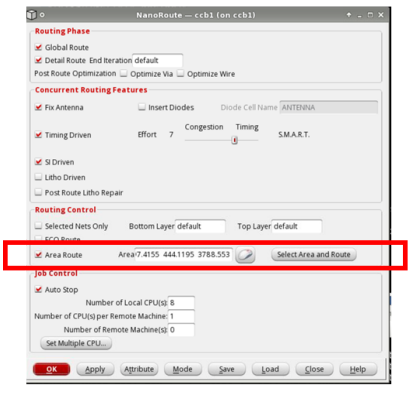
SDC file CHIP_cts.sdc

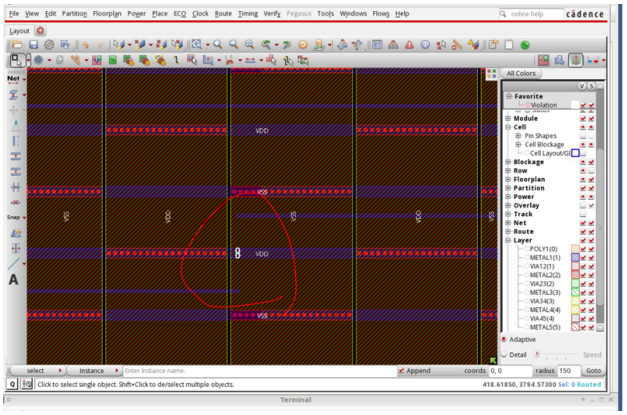
The only difference of CHIP_cts.sdc is that you need to comment out the set_clock_uncertainty & set_clock_transition instruction.

Then do the CTS stage.

3. Short issue

Delete the short metal line, then use area nano route (Area Route--> Select Area and Route) to re-route that path. (sometimes may need to delete a longer metal line so the new routed result won't have short again)





4. Check Area

type > **summaryReport -noHtml -outfile summaryReport.rpt**
then you will see the CORE area.