# 电子科技大学电子科学与工程学院

# 实 验 报 告

## (2022 -2023)

**课程名称** IC 综合实验 2

**实验名称** IC 后端设计

**指导老师** 王忆文

**学生姓名** 曾立伟，高德睿，蒋熠凡，易健驰

**学生学号** 2020340105007，2020340102017，2020340102009，2020340107022

# 电 子 科 技 大 学

# 实 验 报 告

**实验地点：清水河校区芯火基地 B 区 519**

**实验时间：2022.11-2023.1**

## 报告目录

一、实验室名称：

二、实验项目名称：后端设计

三、实验学时：

四、实验原理：请附页

五、实验目的：请附页

六、实验内容：请附页

七、实验器材（设备、元器件）：请附页

八、实验步骤：请附页

九、实验数据及结果分析：请附页

十、实验结论：请附页

十一、总结及心得体会：请附页

十二、对本实验过程及方法、手段的改进建议：请附页

实验报告成绩：_____

## 附页：

## 四、实验原理：

### 1.ICC 软件介绍

IC Compiler，简称 ICC，是 Synopsys 新一代布局布线系统（Astro 是前一代布局布线系统），通过将物理综合扩展到整个布局和布线过程以及 Sign off 驱动的设计收敛，来保证卓越的质量并缩短设计时间。上一代解决方案由于布局、时钟树和布线独立运行，有其局限性。IC Compiler 的扩展物理综合(XPS)技术突破了这一局限，将物理综合扩展到了整个布局和布线过程。IC Compiler 采用基于 TCL 的统一架构，实现了创新并利用了 Synopsys 的若干最为优秀的核心技术。作为一套完整的布局布线设计系统，它包括了实现下一代设计所必需的一切功能，如物理综合、布局、布线、时序、信号完整性(Signal Integrity, SI)优化、低功耗、可测性设计(Design For Test, DFT)和良率优化。新版 ICC 运行时间更快、容量更大、多角/多模优化(MCMM)更加智能、而且具有改进的可预测性，可显著提高设计人员的生产效率。同时，新版本还推出了支持 45 nm、32 nm 技术的物理设计。IC Compiler 正成为越来越多市场领先的 IC 设计公司在各种应用和广泛硅技术中的理想选择。新版的重大技术创新将为加速其广泛应用起到重要作用。IC Compiler 引入了用于快速运行模式的新技术，在保证原有质量的情况下使运行时间缩短了 35%。新版增加了集成的、层次化的设计规划的早期介入，有助于用户高效处理一亿门级的设计。提高生产能效的另一个关键在于物理可行性流程，它能够使用户迅速生成和分析多次试验布局，以确定具体实现的最佳起始值。

### 2. 版图基本原理

使用 MOS 工艺制作集成电路时，由于集成电路的制作是平面加工工艺，而芯片是立体结构。平面工艺到立体结构的实现，需要多层掩模版。每一层掩模版需要用一层版图来表示，因此版图也是分层的，即不同层的版图代表不同的掩模版。版图是用二维图形表示电路的三维结构，版图设计的目的是完成集成电路加工所需的各个掩模版上的图形的设计。

在进行复杂集成电路加工时，单层金属连线往往不能够满足需求，因此会有

多层金属连线。如本次实验使用的是 6 层金属布线工艺。

**3. TCL 脚本**

本次实验通过编写运行 Tcl 脚本完成各步骤，减少在图形化界面的繁杂操作。Tcl 是一种很通用的脚本语言，它几乎在所有的平台上都可以解释运行，功能强大。是 tool command language 的缩写，发音为 "tickle”。 实际上包含了两个部分：一个语言和一个库。 首先，Tcl 是一种简单的脚本语言，主要使用于发布命令给一 些互交程序如文本编辑器、调试器和 shell。它有一个简单的语法 和很强可扩充性，Tcl 可以创建新的过程以增强其内建命令的能力。其次，Tcl 是一个库包，可以被嵌入应用程序，Tcl 的库包含了一个分析器、用于执行内建命令的例程和可以使你扩充（定义新的 过程）的库函数。应用程序可以产生 Tcl 命令并执行，命令可以由用户产生，也可以从用户接口的一个输入中读取（按钮或菜单等）。 但 Tcl 库收到命令后将它分解并执行内建的命令，经常会产生递归的调用。

## 五、实验目的

将 DC 综合后输出的 SDC 约束文件，门极网表，代工厂提供的元件库（.db），物理库（physical lib），工艺文件（.tf），RC 参数文件（tlu+）等输入 ICC 后，通过软件中完成布局布线操作，最终导出网表（.v），提取寄生参数、导出时序描述（.sdf），导出寄生参数文件（.spef），以及版图阶段用于加工的 GDSII 文件。

在 ICC 中再次导出门极网表和时序描述文件用于后仿真，其与 DC 导出文件的区别是：

(1) ICC 导出的门极网表文件中的模块与连线与版图是严格对应的

(2) ICC 导出的时序描述文件中的延时由 RC 参数计算而来

## 六、实验内容

（1）数据准备（Data Setup）：将门极网表、约束文件、元件库、工艺文件、物理库、寄生参数模型等输入 ICC；

（2）布局规划（Design Planning）：规定芯片尺寸、形状，确定 IO、电源、pad 位置等，放置宏单元，放置标准单元，铺设电源网络；

（3）布局（Placement）：将电路中各个基本单元在芯片中进行布局；

（4）时钟树综合（Clock Tree Synthesis）：将时钟信号连接至需要驱动的基本单元；

（5）布线（Routing）：将各个基本单元对应端口进行连接；

（6）优化（Chip Finishing）：连线拓展、加宽连线，冗余通孔插入，插入填充单元，填充金属等。

## 七、实验器材

**Centos7、ICC 软件**

## 八、实验步骤

**1. 数据准备**

（1）配置环境

在启动 icc 之前，我们有如下文件：

①ICC 启动的环境设置文件：.synopsys_dc.setup 文件，此文件需要修改一些变量名。

主要的设置内容有：

禁止显示一些警告，比如创建库、布局布线过程中的一些警告；

设置一些有用的功能操作（ICC 中没有的），需要相应的设置文件；

设置一些变量开关状态（开或者关）；

设置允许记录 shell 的命令信息；

设置一些关联命令；

逻辑库设置（设置 search_path ,target_libr,link_library,设置 min 库：包括标准单元库、IO 库、ram 库），这就需要库了；

定义一些变量，这些变量内容是不同路径下一些文件的名称；

②设计的 verilog 门级网表：.v 文件，此文件通过 dc 综合生成。

③设计的平面布局图：.def 文件，这个文件通过布局规划得到。

④时序约束文件：.sdc 文件，主要是时序的约束，来自 dc 综合。

⑤时序和优化控制文件：opt_ctrl.tcl 文件，对设计进行优化的设置。

⑥检查零互联时的时序约束文件：zic_timing.tcl 文件，主要是检查设计在 0 负载的时候，时序的情况。

⑦创建电源和地的逻辑连接文件：derive_pg.tcl 文件。顾名思义就是创建电源和地的逻辑连接。

（2）了解库文件

在 ref 参考库文件夹里，有如下文件：

①普通的库文件（db）：里面有标准单元库、IO 单元库、RAM 宏单元库等

②参考的 milkyway 物理库（mw_lib）：里面有各种单元库（IO 单元、标准单元等）文件夹

③工艺库文件夹（tech）:主要是.tf 工艺库文件

④寄生参数库（tlup）：主要是 RC 参数（.tluplus 库文件），用来提取寄生参数用。.map 文件主要是用来映射的文件。

（3）创建 milkyway 设计库

对于一个设计，需要根据参考可以创建相应的 milkyway 库。

create library:主要是填写 new library name(这个跟设计名称一样)、工艺库名字，不需要导入物理库；然后导入参考库 ref 中的库单元（ref 中的 sc、io、ram）；勾选打开库，然后确定之后就可以完成 mw_lib 的创建了。

（4）加载/读入.v 网表文件、tlu+(RC 参数)文件、约束以及控制文件

①确保设计库打开的情况下，导入.v 文件（FILE-import design），导入过程中，需要注意设置顶层设计。

②加载 tlup+寄生文件：file -- set tlu+(寄生文件在 ref 的寄生文件夹下面)，注意需要对于的文件（max、min 还是.map）

③检查物理库和逻辑库的一致性;如果有缺失或者不匹配的标准单元库（standard cells）、IP cells、IOpads(注意不是 pin)，那么需要修改库，否则可以忽视警告消息。

④检查 tlu+文件是否添加;有三个 passed 才算通过。

⑤验证指定的已经加载的库（list_lib）:查看.synopsys_dc.setup 文件中定义的库是否正确。

⑥定义电源、地和网络之间的连接，同时检查是否有未连接的电源和地引脚

⑦读入时序约束文件，进行时序约束

**2.布局规划**

（1）添加 IO Corner、为 Core 供电的 IO 和为 pad 供电的 IO；然后读入管脚约束文件，它指定了每个 IO 在整个芯片中的位置和排列顺序

（2）在设计中添加 physical only cells，读入 IO 约束文件（.tdf 文件）

（3） 创建 Floorplan。点击菜单栏“Floorplan-Create Floorplan”，在弹出的窗口中设置 core 和 pads 之间的距离（13），同时合理设置 core 的面积，点击“Apply”“OK”

（4）加入 Pad filler，通过脚本添加

（5）添加电源：点击“Preroute-Power Network Constraints”对电源环（ring）和一些较宽的电源和地的金属（strap）进行设置，点击“Preroute-Synthesize Power Network”自动产生电源网络，查看 IR drop 图看是否符合要求

（6）自动做 Floorplan 的 Placement，对 Floorplan 的结果进行评估，使用 repot 命令对时序（timing）和堵塞（congestion）进行检查

**3.布局**

在布局之前 ,需要进行布局准备的检查,一次性的设计和流程的设置,拥塞,时

序,功耗,还有面积的 QoR 的设置;最后才是 placement & optimization.

在布局规划阶段完成了芯片的整体规划，而布局阶段主要是软件自动的标准单元的摆放；用命令 place_opt -area_recovery -power 完成布局，此后使用 repot 命令对时序（timing）和堵塞（congestion）进行检查。本阶段脚本如下图所示：

```
open_mw_lib ./my_design.mw
open_mw_cel divider_chip


set_separate_process_options -placement false


place_opt -area_recovery -power

#incremental optimization
#excute this steps if your design have some congestion problems.
psynopt -area_recovery -power

report_timing > ./report/3_place_timing.rpt
report_congestion > ./report/3_place_congestion.rpt

save_mw_cel -as placed
save_mw_cel  -design "divider_chip"
exit
```

**4. 时钟树综合**

（1）将所有时钟的时钟目标偏移设置为 0.5；调整 CTS 后时序分析和优化的时钟不确定性，set_clock_uncertainty 0.5

（2）指定在 CTS 的“DRC 缓冲”阶段使用的缓冲区：set_clock_tree_references -references

（3）配置 CTS 使用非默认布线：

（4）开始时钟树综合，合成所有时钟树，无需任何时序优化或布线：clock_opt -only_cts -no_clock_route

（5）生成时序报告：report_timing；生成所有的违规报告，并查看：v report_constraint -all；保存设计：save_mw_cel -as clock_opt_cts

（6）启用保持时间修复：set_fix_hold [all_clocks]

（7）进行面积范围设置：set physopt_area_critical_range 0.4

（8）RC 参数的提取：extract_rc

（9）执行 CTS 后无时钟布线的时序、面积和扫描链优化：clock_opt -only_psyn

-area_recovery -optimize_dft -no_clock_route

（10）进行时钟树布线，：route_zrt_group -all_clock_nets -reuse_existing_global_route true

（11）最后查看报告和违规并保存到 report 文件夹

## 5. 布线

时钟树综合结束之后，接下来的工作是布线(routing)。routing 阶段完成标准单元的信号线的连接，布线工具会自动进行布线拥塞消除、优化时序、减小耦合效应、消除窜扰、降低功耗、保证信号完整性等问题。在本次综合实验中，我们需要分析设计的时序、逻辑和物理的 DRC、LVS 违规，并且修复 LVS 错误。本阶段工作的脚本如下图所示：

```
open_mw_lib ./my_design.mw
open_mw_cel divider_chip

set_separate_process_options -routing false -placement false -extraction false

remove_ideal_network -all
set_propagated_clock [all_clocks]
set_fix_hold [all_clocks]

report_constraint -all > ./report/5_timing_before_route.rpt


clock_opt -only_hold_time

verify_pg_nets

preroute_standard_cells -remove_floating_pieces

verify_pg_nets

set_route_zrt_common_options -post_detail_route_redundant_via_insertion medium

set_route_zrt_detail_options -optimize_wire_via_effort_level medium

route_opt -initial_route_only

route_opt -skip_initial_route -power

### DRC
verify_zrt_route > ./report/5_route_DRC.rpt


### LVS
verify_lvs > ./report/5_route_LVS.rpt

### route incrementlly if have DRC and LVS issues.
route_opt -incremental

save_mw_cel -as route
save_mw_cel  -design "divider_chip"
exit
```

## 6. chip finishing

本阶段的主要内容是检查设计，以及为提高良率（yield）和解决物理规则违规做的工作。主要包括：进行连线扩展和加宽连线来减少关键面积；DRC 和 LVS 检查；进行冗余通孔插入（冗余通孔的插入是可选的）；使用填充单元填充核心中未填满的位置；填充金属以满足金属密度的要求；生成 GDSII 数据。

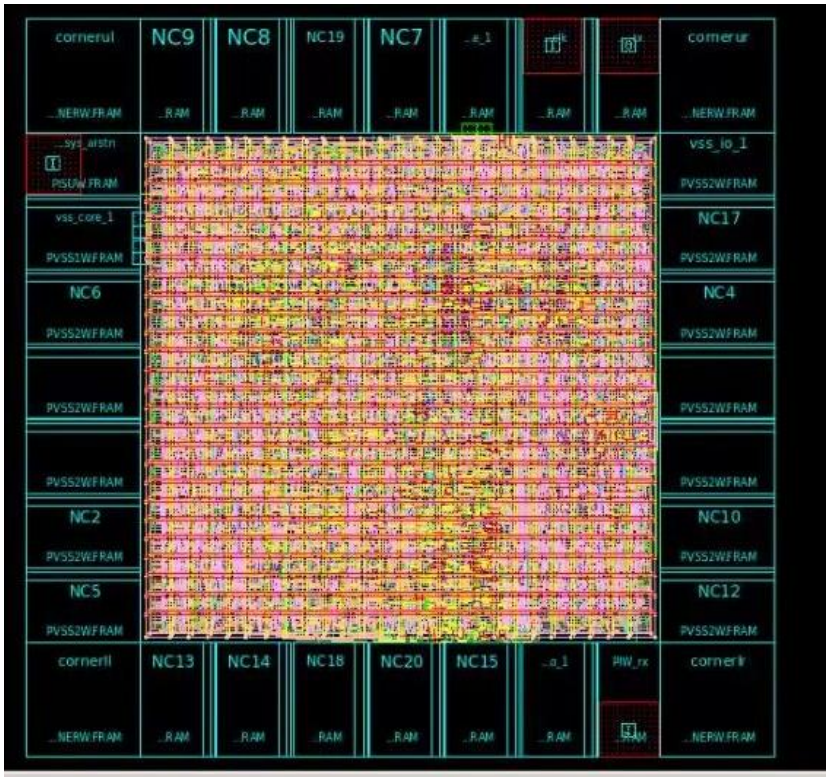
可以在 Layout window 调取 IR-drop、cell density，点击菜单栏的 verification 可以进行 DRC 检查和 LVS 验证，多数操作由脚本来完成

## 九、实验数据及结果分析：

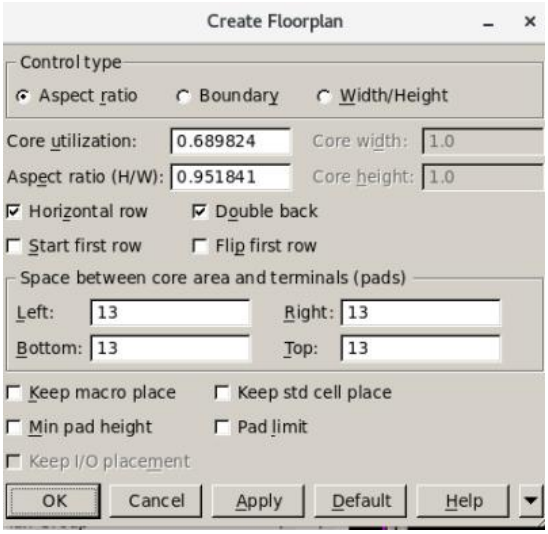
**1. 布局布线完成后的版图参数和芯片版图。**

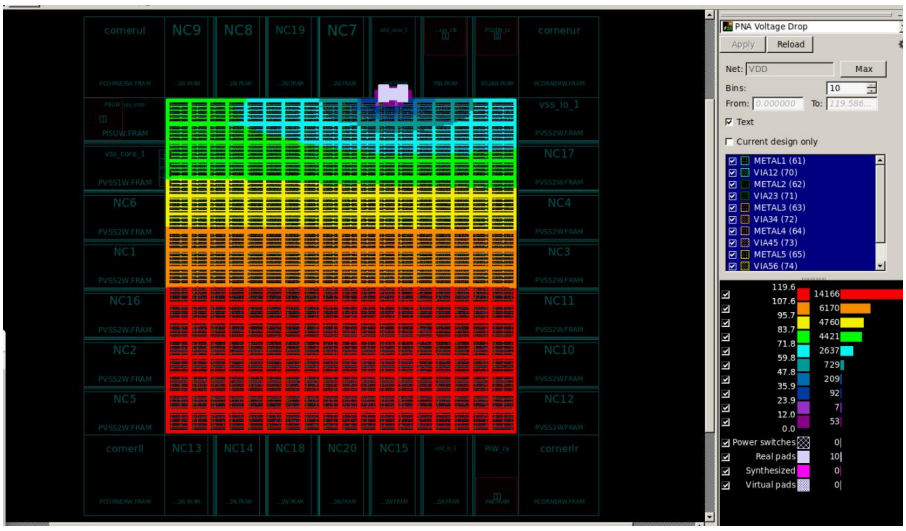
| 面积 | 长 | 宽 | 利用率 |
|---|---|---|---|
| 83.4wum2 | 914um（153+13+748） | 913um | 86.66% |

| 时钟频率 | 最坏时钟偏差 | 最长时钟延迟 | 关键路径到达时间 | 时钟余量 |
|---|---|---|---|---|
| 25MHz | 0.647 | 1.987 | 35.96 | 3.34 |

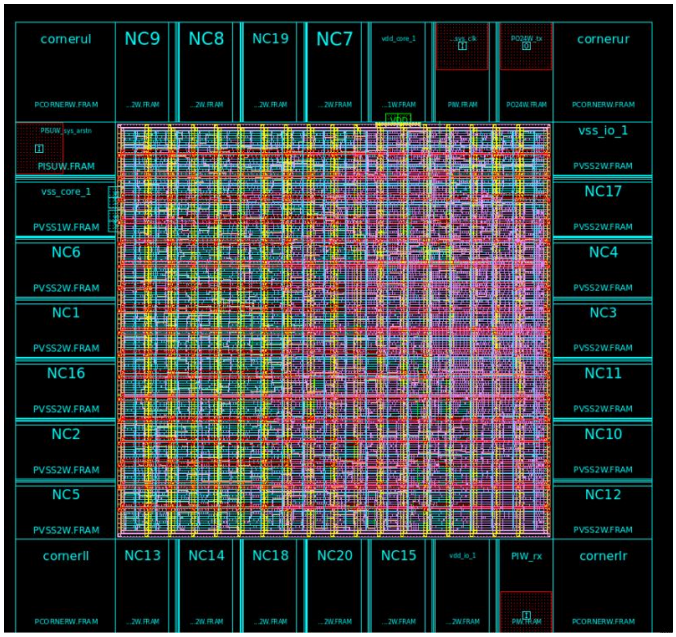



**2. floorplan 阶段的参数设置与 powerstrap 视图。**

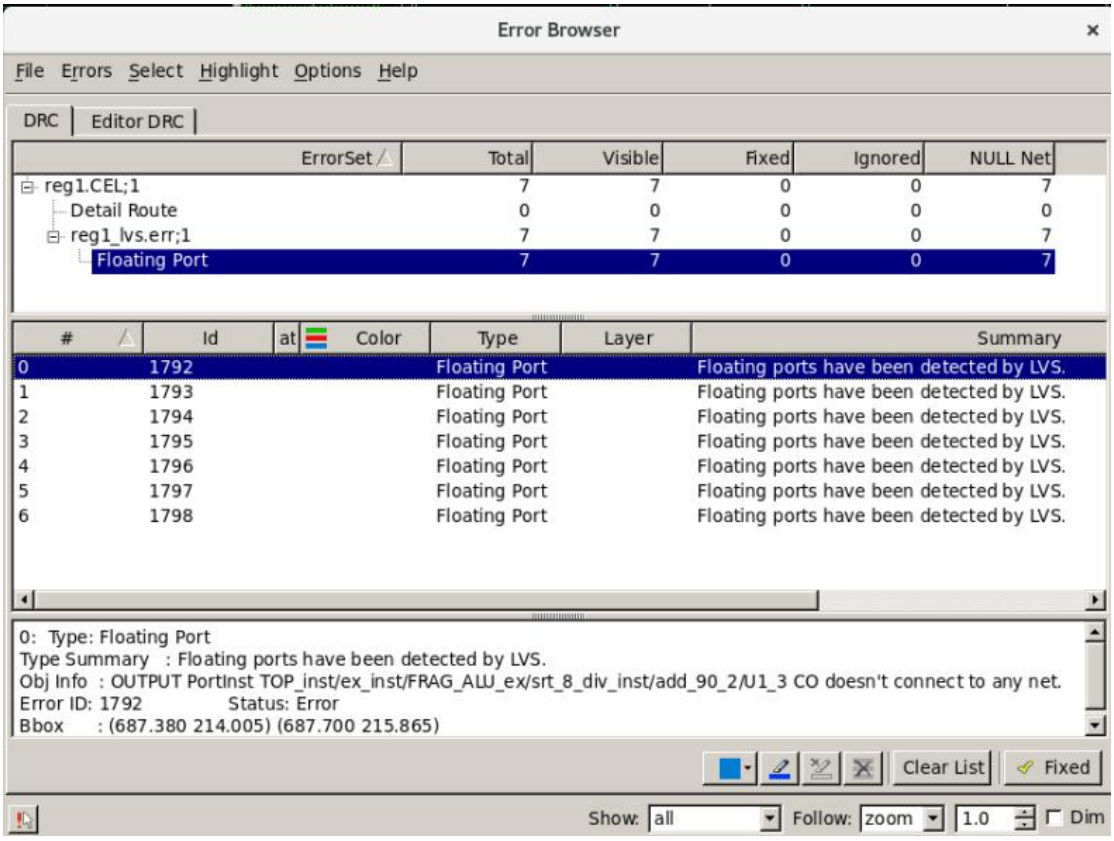



### 3. 布局布线完成后的版图



**4.布线完成后在 error browser 里查看违例**




**5.最终的时序检查报告**

```
                                                          0.00       35.56 r
TOP_inst/id_ex_inst/pipline_reg2_r_data/U9/Z (AO222UHDV1)
                                                          0.40 *     35.96 r
TOP_inst/id_ex_inst/pipline_reg2_r_data/data_temp_reg_1_/D (DSRNQUHDV1)
                                                          0.00 *     35.96 r
data arrival time                                                    35.96

clock clk (rise edge)                                    40.00       40.00
clock network delay (ideal)                               0.00       40.00
clock uncertainty                                        -0.50       39.50
TOP_inst/id_ex_inst/pipline_reg2_r_data/data_temp_reg_1_/CK (DSRNQUHDV1)
                                                          0.00       39.50 r
library setup time                                       -0.21       39.29
data required time                                                   39.29
-----------------------------------------------------------------------------
data required time                                                   39.29
data arrival time                                                   -35.96
-----------------------------------------------------------------------------
slack (MET)                                                          3.34
```

**6.芯片面积利用率报告**

```
*********************************
      Sub-Region Utilization
*********************************
Number of regions with placement utilization 0 - 0.125 is 0 (0.00%)
Number of regions with placement utilization 0.125 - 0.25 is 0 (0.00%)
Number of regions with placement utilization 0.25 - 0.375 is 0 (0.00%)
Number of regions with placement utilization 0.375 - 0.5 is 1 (0.08%)
Number of regions with placement utilization 0.5 - 0.625 is 2 (0.15%)
Number of regions with placement utilization 0.625 - 0.75 is 34 (2.62%)
Number of regions with placement utilization 0.75 - 0.875 is 347 (26.77%)
Number of regions with placement utilization 0.875 - 1 is 912 (70.37%)
```

```
****************************************
  Report : Chip Summary
  Design : CHIP
  Version: O-2018.06-SP1
  Date   : Thu Dec 29 15:44:08 2022
****************************************
Std cell utilization: 86.66%  (163198/(188328-0))
(Non-fixed + Fixed)
Std cell utilization: 90.20%  (162865/(188328-7768))
(Non-fixed only)
Chip area:             188328   sites, bbox (153.00 153.00 748.84 747.72) um
Std cell area:         163198   sites, (non-fixed:162865 fixed:333)
                       15172    cells, (non-fixed:15146  fixed:26)
Macro cell area:       0        sites
                       0        cells
Placement blockages:   0        sites, (excluding fixed std cells)
                       7768     sites, (include fixed std cells & chimney area)
                       0        sites, (complete p/g net blockages)
Routing blockages:     0        sites, (partial p/g net blockages)
                       0        sites, (routing blockages and signal pre-route)
Lib cell count:        152
Avg. std cell width:   4.95 um
Site array:            unit     (width: 0.56 um, height: 3.36 um, rows: 177)
Physical DB scale:     1000 db_unit = 1 um
```

## 7.时钟偏移报告

```
Information: Float pin scale factor for the 'max' operating condition of scenario 'default' is set to 1.000 (CTS-375)

============ Global Skew Report ================

Clock Tree Name               : "clk"
Clock Period                  : 40.00000
Clock Tree root pin           : "chip_sys_clk"
Number of Levels              : 5
Number of Sinks               : 2135
Number of CT Buffers          : 26
Number of CTS added gates     : 0
Number of Preexisting Gates   : 1
Number of Preexisting Buf/Inv : 0
Total Number of Clock Cells   : 27
Total Area of CT Buffers      : 626.57269
Total Area of CT cells        : 11266.56738
Max Global Skew               : 0.64683
Number of MaxTran Violators   : 0
Number of MaxCap Violators    : 2
Number of MaxFanout Violators : 0


Operating Condition             worst
Clock global Skew               0.647
Longest path delay              1.987
Shortest path delay             1.340

The longest path delay end pin: TOP_inst/if_inst/FRAG_inst_mem_if/mem_reg_10__5_/CK
The shortest path delay end pin: TOP_inst/if_inst/inst_addr_temp_reg_21_/CK
```

## 8.时钟延迟报告：

```
The longest path delay end pin: TOP_inst/if_inst/FRAG_inst_mem_if/mem_reg_10__5_/CK
The shortest path delay end pin: TOP_inst/if_inst/inst_addr_temp_reg_21_/CK

The longest Path:
Pin                                        Cap       Fanout    Trans     Incr      Arri
-------------------------------------------------------------------------------------------
chip_sys_clk                               0.000          1  0.000     0.000     0.000     r
chip_sys_clk                               4.541          1  2.000     0.000     0.000     r
PIW_sys_clk/PAD                            4.541          1  2.000     0.064     0.064     r
PIW_sys_clk/C                              0.813          4  0.402     0.757     0.821     r
TOP_inst/CLKINUHDV8_G2B2I1/I               0.813          1  0.733     0.368     1.188     r
TOP_inst/CLKINUHDV8_G2B2I1/ZN              0.147          4  0.343     0.309     1.497     f
TOP_inst/if_inst/FRAG_inst_mem_if/CLKINUHDV16_G2B1I2/I
                                           0.147          1  0.344     0.003     1.500     f
TOP_inst/if_inst/FRAG_inst_mem_if/CLKINUHDV16_G2B1I2/ZN
                                           0.528         97  0.654     0.475     1.975     r
TOP_inst/if_inst/FRAG_inst_mem_if/mem_reg_10__5_/CK
                                           0.528          0  0.655     0.013     1.987     r
[clock delay]                                                                     1.987
-------------------------------------------------------------------------------------------

The Shortest Path:
Pin                                        Cap       Fanout    Trans     Incr      Arri
-------------------------------------------------------------------------------------------
chip_sys_clk                               0.000          1  0.000     0.000     0.000     r
chip_sys_clk                               4.541          1  2.000     0.000     0.000     r
PIW_sys_clk/PAD                            4.541          1  2.000     0.064     0.064     r
PIW_sys_clk/C                              0.813          4  0.402     0.757     0.821     r
TOP_inst/CLKINUHDV16_G2B2I3/I              0.813          1  0.178     0.009     0.829     r
TOP_inst/CLKINUHDV16_G2B2I3/ZN             0.205          5  0.155     0.138     0.968     f
TOP_inst/CLKINUHDV16_G2B1I8/I              0.205          1  0.155     0.003     0.971     f
TOP_inst/CLKINUHDV16_G2B1I8/ZN             0.456         94  0.560     0.367     1.338     r
TOP_inst/if_inst/inst_addr_temp_reg_21_/CK 0.456          0  0.560     0.002     1.340     r
[clock delay]                                                                     1.340
-------------------------------------------------------------------------------------------


***** End Of Report *****
```

**9.寄存器间数据延迟余量报告**

```
                                                          0.00       35.56 r
TOP_inst/id_ex_inst/pipline_reg2_r_data/U9/Z (AO222UHDV1)
                                                          0.40 *     35.96 r
TOP_inst/id_ex_inst/pipline_reg2_r_data/data_temp_reg_1_/D (DSRNQUHDV1)
                                                          0.00 *     35.96 r
data arrival time                                                    35.96

clock clk (rise edge)                                    40.00       40.00
clock network delay (ideal)                               0.00       40.00
clock uncertainty                                        -0.50       39.50
TOP_inst/id_ex_inst/pipline_reg2_r_data/data_temp_reg_1_/CK (DSRNQUHDV1)
                                                          0.00       39.50 r
library setup time                                       -0.21       39.29
data required time                                                   39.29
-------------------------------------------------------------------------------
data required time                                                   39.29
data arrival time                                                   -35.96
-------------------------------------------------------------------------------
slack (MET)                                                          3.34
```

## 十、实验结论：

由于我们仅仅有 4 个信号引脚和 4 个供电引脚，其余都是空引脚，相比于引脚占用多的设计来说，经过自动优化以后，我们对于 pad 布局的可能性更加有限，因此可以通过简单的分类穷举，把所有合理的 pad 布局穷举出来查看利用率，最终我们的总面积利用率达到了 86.66%，为尽可能减小芯片总面积，通过反复实

验，电源环宽度采取确保 PVSS1W 正常布线的最小值 13um，最终面积为 914*913um，在 25MHz 下时钟余量为 3.34ns，完成了后端设计，成功导出了相关版图文档与后仿文档。

## 十一、总结及心得体会：

初学者可使用 ICC 的图形界面操作，有助于增进对使用 ICC 软件进行自动布局布线的理解；使用脚本能大大提高工作的效率；dc 综合后的面积报告会初步确定布局时的标准单元面积，但是同时面积会因连线的复杂性有些许增加；布局规划时可以查看利用率，后续可以查看 cell density，密度太高时自动布局布线会出现问题（如 lvs 的 short），利用率应保持在 70 左右。同时，编写 padcons 脚本时，对于电源位置的摆放、电源环的一些设置非常关键，是大部分 error 的源头。

## 十二、对本实验过程及方法、手段的改进建议：

（1）实验操作是在 Linux 系统下完成的，因此对 Linux 系统的不熟悉可能会导致操作效率低下，建议实验前能增加对 Linux 系统命令的更详细讲解和练习。

（2）建议在实验前增加对 icc 软件运行原理方面的讲解，脚本中一些关键操作的语句可以讲讲原理。