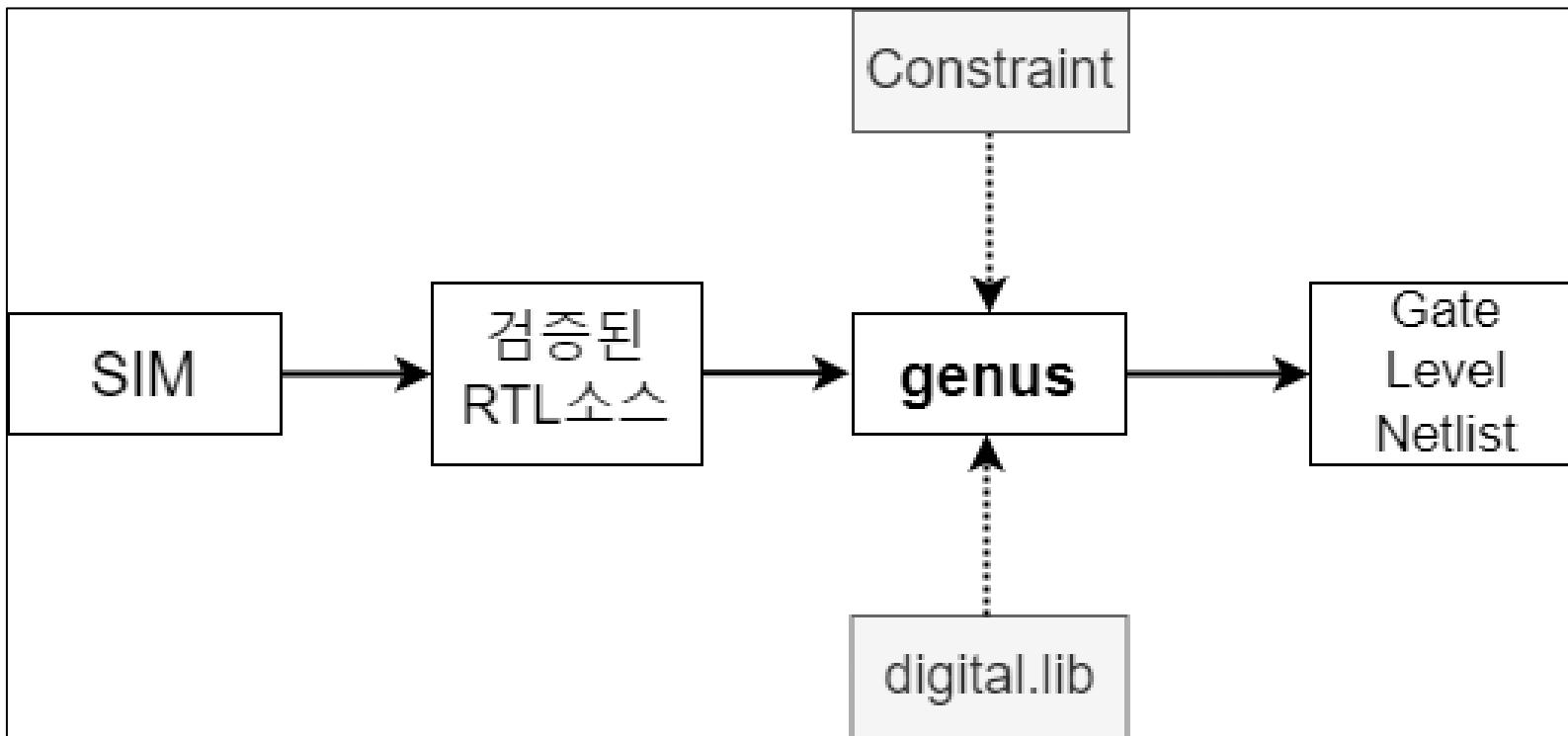


genus

# Synthesis

## genus란?

- 시뮬레이션을 통해 검증된 RTL소스를 합성하여 G.L.N(Gate Level Netlist)를 만들어내는 툴
- 합성 과정에서 Constraint 목표를 만족해야 하며 공정사에서 제공한 라이브러리(.lib)를 사용함



# Synthesis

## genus

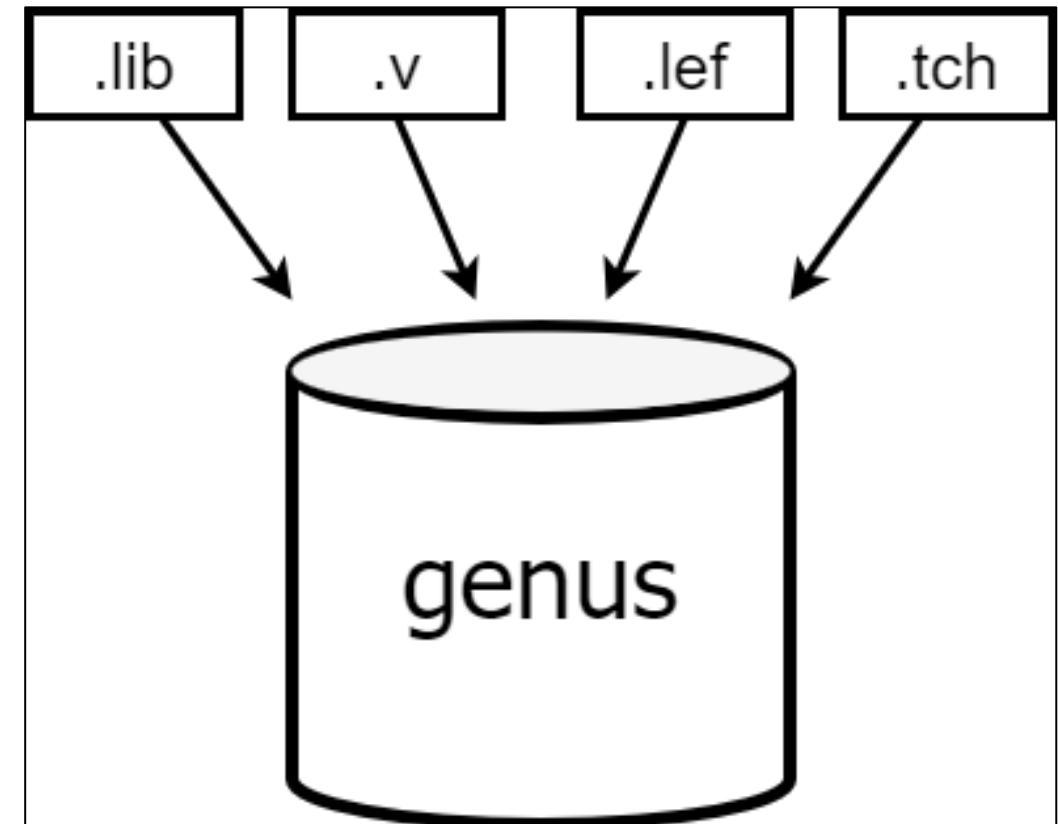
- 합성에 필요한 확장자 파일

**.lib:** STD셀과 IO셀 등이 있는 공정사의 라이브러리 파일

**.v:** 디자인의 Verilog 소스코드가 적혀 있는 파일

**.lef:** STD셀의 프레임 셀

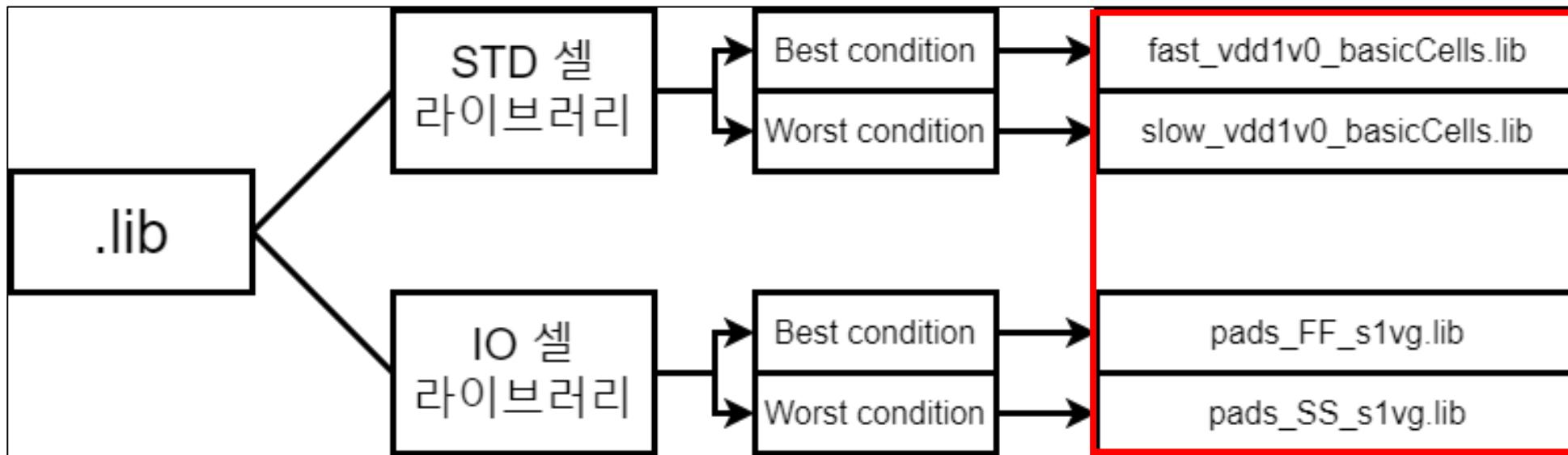
**.tch:** 기생 성분을 담고 있는 파일



# Synthesis

genus

Genus 툴에 사용할 4개의 .lib  
파일을 결정

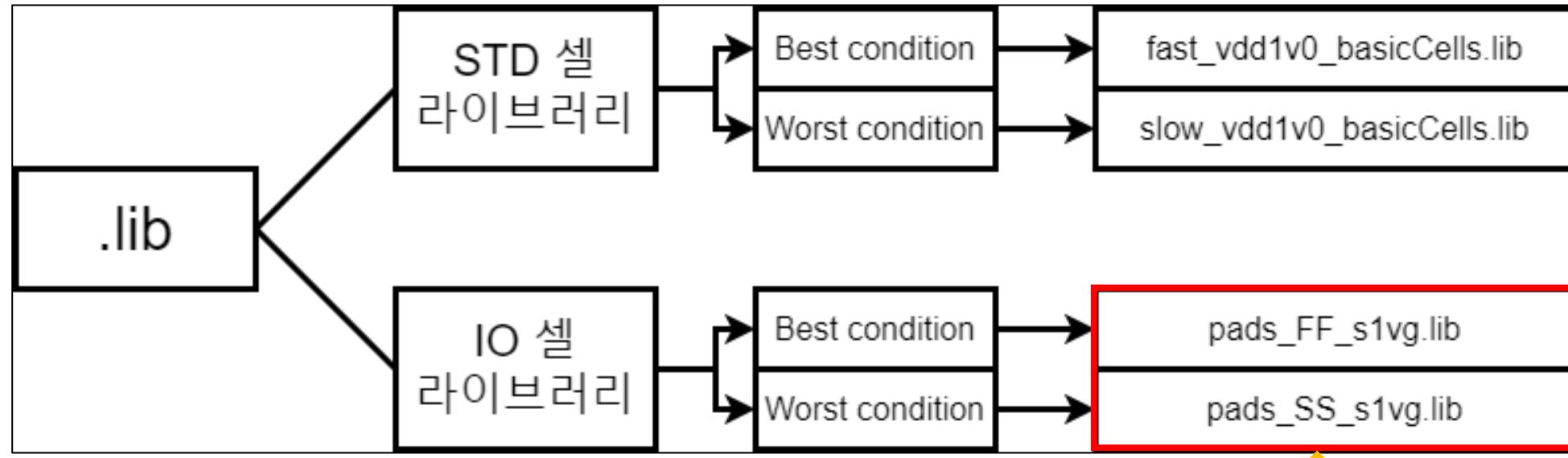


set\_db 명령어를 통해 라이브러리의 검색 경로를 정의함

`set_db init_lib_search_path [concat $STD_LIB_PATH $IO_LIB_PATH $LEF_LIB_PATH $QRC_FILE_PATH]`

# Synthesis

genus



**충돌 이슈 발생:** IO셀 .lib 파일의 라이브러리 내용을 보면 이름이 겹쳐 충돌하는 문제가 있음

**해결 제안 :** IO셀 라이브러리 파일은 하나만 기입

# Synthesis

genus

- IO셀 충돌 이슈 확인 방법

```
/home/ex_poly1/SoC2/GPDK045/digital/giolib045_v3.5/timing
```

pads\_FF\_s1vg.lib    pads\_SS\_s1vg.lib    pads\_TT\_s1vg.lib

vi 진행 방법

```
$> vi pads_FF_s1vg.lib
```

```
: vs
```

```
: e .
```

```
pads_SS_s1vg.lib -> Enter
```

```
2 delay model : typ  
3 check model :  
4 power model :  
5 capacitance model :  
6 other model : typ  
7 */  
8 library(giolib045) {  
9
```

라이브러리 이름이  
같으므로 충돌 발생

```
2 delay model : typ  
3 check model : typ  
4 power model : typ  
5 capacitance model : typ  
6 other model : typ  
7 */  
8 library(giolib045) {  
9
```

해결 제안 : IO셀 라이브러리 파일은 하나만 기입

```
29 set LIB_LIST {fast_vdd1v0_basicCells.lib slow_vdd1v0_basicCells.lib  
pads_SS_s1vg.lib}
```

# Synthesis

genus

\$> genus

genus> gui\_show

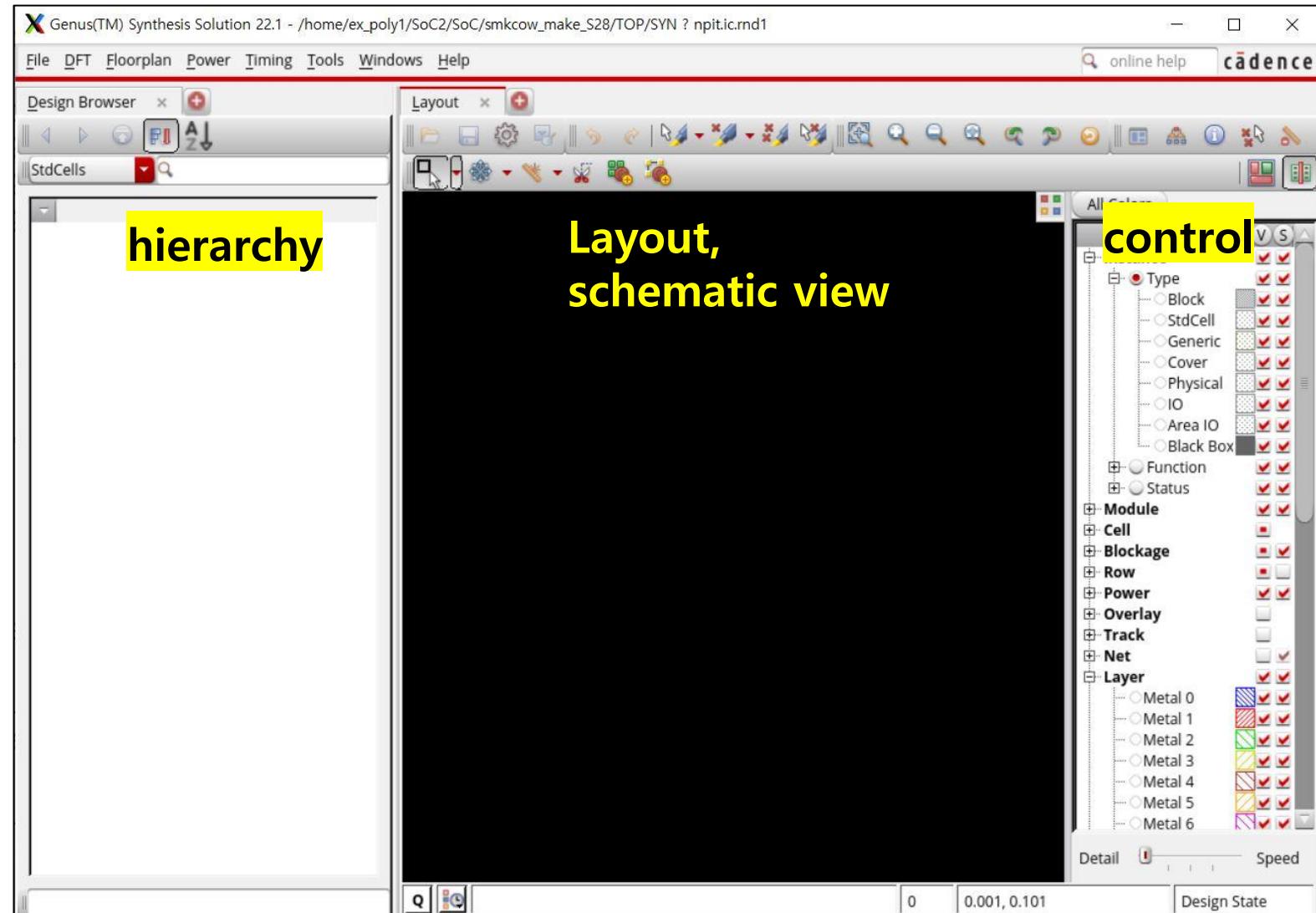
[ex\_poly1@npit SYN]\$ genus

@genus:root: 1> gui\_show

# Synthesis

genus

- Gui 확인



# Synthesis

genus

\$> vi run\_synthesis

/home/ex\_poly1/SoC2/SoC/smkcw\_make\_S28/TOP/SYN

clean.tcl cons fv log mapped report **run\_synthesis** script unmapped

자동 배치모드로 합성을  
진행하기 위한  
실행 파일

# Synthesis

## genus

- 2행 cortexm0\_45nm.tcl 실행할 것임

-f : 자동으로  
script 실행

-log : 툴 실행부터  
종료까지의 전체  
log를 저장

```
1 #!/bin/csh -T
2 genus -f ./script/cortexm0_45nm.tcl -log ./log/all.log
3
```

LAB

# Synthesis

## genus를 사용 준비

- 두개의 터미널 사용 (아래의 경로에서 진행)
- 하나는 genus를 실행, 나머지는 vi run\_synthesis

/home/ex\_poly1/SoC2/SoC/smkcowl\_make\_S28/TOP/SYN

The image shows two terminal windows side-by-side. The left terminal window displays the output of the 'genus' command, which includes the Cadence Genus Synthesis Solution copyright notice, system configuration details (version 22.16-s078\_1, built on Sun Jun 09 22:32:57 PDT 2024), and license information. It also shows the execution of the 'run\_synthesis' script, which is a template for RTL-to-Gate-Level Flow. The right terminal window shows the continuation of the 'run\_synthesis' script, detailing various environment variable assignments such as TOP DESIGN, LOG\_DIR, RPT\_DIR, UNMAPPED\_DIR, MAPPED\_DIR, and LIB\_LIST, along with specific paths for GPKD045 libraries and RTL sources.

```
[ex_poly1@npit:SYN]$ genus
TMPDIR is being set to /tmp/genus_temp_22450_npit.ic.rnd1_ex_poly1_Fn4CGL
Cadence Genus(TM) Synthesis Solution.
Copyright 2024 Cadence Design Systems, Inc. All rights reserved worldwide.
Cadence and the Cadence logo are registered trademarks and Genus is a trademark
of Cadence Design Systems, Inc. in the United States and other countries.

[13:11:50.440153] Configured Lic search path (22.01-s003): 35266@npit-service1.localhost.org

Version: 22.16-s078_1, built Sun Jun 09 22:32:57 PDT 2024
Options:
Date: Thu Feb 06 13:11:50 2025
Host: npit.ic.rnd1 (x86_64 w/Linux 3.10.0-1160.119.1.el7.x86_64) (20cores*80cpus*
2physical cpus*Intel(R) Xeon(R) Silver 4316 CPU @ 2.30GHz 30720KB) (395262912KB)
PID: 22450
OS: Red Hat Enterprise Linux Server release 7.9 (Maipo)

Checking out license: Genus_Synthesis
[13:11:50.411130] Periodic Lic check successful
[13:11:50.411145] Feature usage summary:
[13:11:50.411145] Genus_Synthesis

*****
Finished executable startup (1 second elapsed).

Loading tool scripts...
Finished loading tool scripts (16 seconds elapsed).

WARNING: This version of the tool is 240 days old.
@genus:root: 1>
```

```
## Template Script for RTL->Gate-Level Flow (generated from GENUS 17.10-p007_1)
if {[file exists /proc/cpuinfo]} {
    sh grep "model name" /proc/cpuinfo
    sh grep "cpu MHz"      /proc/cpuinfo
}

puts "Hostname : [info hostname]"

#####
# Preset global variables and attributes
#####

set TOPDESIGN cmsdk_mcu

set LOG_DIR      "./log"
set RPT_DIR      "./report"
set UNMAPPED_DIR "./unmapped"
set MAPPED_DIR   "./mapped"

#GPKD 045 SVT
set STD_LIB_PATH "../../../../GPKD045/digital/gsclib045_all_v4.4/gsclib045/timing"
set IO_LIB_PATH  "../../../../GPKD045/digital/giolib045_v3.5/timing"
set LEF_LIB_PATH "../../../../GPKD045/digital/gsclib045_all_v4.4/gsclib045/lef ../../../../../../giolib045_v3.5/lef"
set RTL_PATH     "../RTL/"
set QRC_FILE_PATH "../../../../GPKD045/digital/gsclib045_all_v4.4/gsclib045_tech/qrc/qx"

#GPKD 045 SVT
set LIB_LIST {fast_vdd1v0_basicCells.lib slow_vdd1v0_basicCells.lib pads_SS_s1vg.lib}
set LEF_LIST {gsclib045_tech.lef gsclib045_macro.lef giolib045.lef}
set QRC_LIST {gpkd045.tch}

set RTL_SOURCES "../RTL/cmsdk_mcu.v \
                  ../RTL/verilog_smkcowl/cmsdk_mcu_clkctrl.v \
                  ../RTL/verilog_smkcowl/cmsdk_mcu_pin_mux.v \
                  ../../../../../../cortexm0_designstart/implementation/cortex_m0_mcu_system_cadence_k_mcu_system_syn_final.v \
                  "
```

# Synthesis

genus

```
[ex_poly1@npit SYN]$ genus
```

```
@genus:root: 1> gui_show
```

## ◆ 터미널 1

\$> genus

genus> gui\_show

```
[ex_poly1@npit SYN]$ genus
TMPDIR is being set to /tmp/genus_temp_78333_npit.ic.rnd1_ex_poly1_iGZGh9
Cadence Genus(TM) Synthesis Solution.
Copyright 2024 Cadence Design Systems, Inc. All rights reserved worldwide.
Cadence and the Cadence logo are registered trademarks and Genus is a trademark
of Cadence Design Systems, Inc. in the United States and other countries.

[16:27:16.440249] Configured Lic search path (22.01-s003): 35266@npit-service1.iptime.org

Version: 22.16-s078_1, built Sun Jun 09 22:32:57 PDT 2024
Options:
Date:    Thu Jan 16 16:27:16 2025
Host:    npit.ic.rnd1 (x86_64 w/Linux 3.10.0-1160.119.1.el7.x86_64) (20cores*80cpus*2physic
PID:    78333
OS:     Red Hat Enterprise Linux Server release 7.9 (Maipo)

Checking out license: Genus_Synthesis
[16:27:16.310939] Periodic Lic check successful
[16:27:16.310947] Feature usage summary:
[16:27:16.310947] Genus_Synthesis

*****
*****
```

Finished executable startup (1 second elapsed).

Loading tool scripts...

Finished loading tool scripts (16 seconds elapsed).

WARNING: This version of the tool is 220 days old.

```
@genus:root: 1> █
```

# Synthesis

genus

## ◆ 터미널 2

```
$> vi run_synthesis
```

```
clean.tcl  cons  fv  log  mapped  report  run_synthesis  script  unmapped
```

# Synthesis

genus

## ◆ 터미널 2

- 2행의 스크립트 파일 진행예정

커서를 위치시킨 뒤 g+f

```
1#!/bin/csh -f
2genus -f ./script/cortexm0_45nm.tcl -log ./log/all.log
3
```

# Synthesis

genus

## ◆ script/cortexm0\_45nm.tcl

```
### Template Script for RTL->Gate-Level Flow (generated from GENUS 17.10-p007_1)

if {[file exists /proc/cpuinfo]} {
    sh grep "model name" /proc/cpuinfo
    sh grep "cpu MHz"      /proc/cpuinfo
}

puts "Hostname : [info hostname]"

#####
# Preset global variables and attributes
#####

set TOP DESIGN cmsdk_mcu

set LOG_DIR      "./log"
set RPT_DIR      "./report"
set UNMAPPED_DIR "./unmapped"
set MAPPED_DIR   "./mapped"
```



# Synthesis

## Genus 툴 실행 결과

### genus

- 합성 과정에서 생성되는 폴더의 경로를 설정하는 명령임

```
10 #####
11 # Preset global variables and attributes
12 #####
13
14 set TOP_DESIGN cmsdk_mcu
15
16 set LOG_DIR      "./log"
17 set RPT_DIR      "./report"
18 set UNMAPPED_DIR "./unmapped"
19 set MAPPED_DIR   "./mapped"
20
21 #GPKD 045 SVT
22 set STD_LIB_PATH "../../../../../GPKD045/digital/gsclib045_all_v4.4/gsc
lib045/timing"
23 set IO_LIB_PATH  "../../../../../GPKD045/digital/giolib045_v3.5/timing"
24 set LEF_LIB_PATH "../../../../../GPKD045/digital/gsclib045_all_v4.4/gsc
lib045/lef ../../../../../GPKD045/digital/giolib045_v3.5/lef"
25 set RTL_PATH     "../RTL/"
26 set QRC_FILE_PATH "../../../../../GPKD045/digital/gsclib045_all_v4.4/gsc
lib045_tech/qrc/qx"
```

합성 전의 결과는  
unmapped에 입력되고  
합성 후의 결과는  
mapped에 입력됨

```
@genus:root: 8> set TOP_DESIGN cmsdk_mcu
cmsdk_mcu
@genus:root: 9>
@genus:root: 9> set LOG_DIR      "./log"
./log
@genus:root: 10> set RPT_DIR      "./report"
./report
@genus:root: 11> set UNMAPPED_DIR "./unmapped"
./unmapped
@genus:root: 12> set MAPPED_DIR   "./mapped"
./mapped
```

주의: 행 번호가 보이지 않는 상태에서  
진행하도록 함

# Synthesis

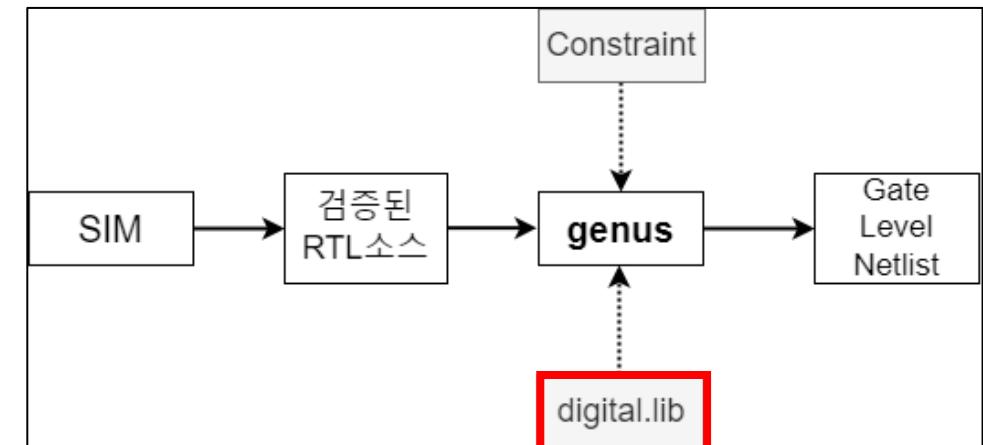
## genus

- genus를 사용하기 위한 환경 setup을 함
- digital.lib을 불러오는 과정임

## Genus 툴 실행 결과

```
@genus:root: 14> set STD_LIB_PATH "../../../GPDK045/digital/gsclib045_all_v4.4/gsclib045/timing"
../../../GPDK045/digital/gsclib045_all_v4.4/gsclib045/timing
@genus:root: 15> set IO_LIB_PATH "../../../GPDK045/digital/giolib045_v3.5/timing"
../../../GPDK045/digital/giolib045_v3.5/timing
@genus:root: 16> set LEF_LIB_PATH "../../../GPDK045/digital/gsclib045_all_v4.4/gsclib045/lef"
../../../GPDK045/digital/giolib045_v3.5/lef
../../../GPDK045/digital/gsclib045_all_v4.4/gsclib045/lef ../../../GPDK045/digital/giolib045_v3.5/lef
@genus:root: 17> set RTL_PATH      "../RTL/"
../RTL/
@genus:root: 18> set QRC_FILE_PATH "../../../GPDK045/digital/gsclib045_all_v4.4/gsclib045_tech/qrc/qx"
../../../GPDK045/digital/gsclib045_all_v4.4/gsclib045_tech/qrc/qx
```

```
10 #####
11 # Preset global variables and attributes
12 #####
13
14 set TOP_DESIGN cmsdk_mcu
15
16 set LOG_DIR      "./log"
17 set RPT_DIR      "./report"
18 set UNMAPPED_DIR "./unmapped"
19 set MAPPED_DIR   "./mapped"
20
21 #GPDK 045 SVT
22 set STD_LIB_PATH "../../../GPDK045/digital/gsclib045_all_v4.4/gsclib045/timing"
23 set IO_LIB_PATH  "../../../GPDK045/digital/giolib045_v3.5/timing"
24 set LEF_LIB_PATH "../../../GPDK045/digital/gsclib045_all_v4.4/gsclib045/lef"
../../../GPDK045/digital/giolib045_v3.5/lef
25 set RTL_PATH     "../RTL/"
26 set QRC_FILE_PATH "../../../GPDK045/digital/gsclib045_all_v4.4/gsclib045_tech/qrc/qx"
```



주의: 행 번호가 보이지 않는 상태에서  
진행하도록 함

# Synthesis

genus

경로확인\_0

22행 STD\_LIB\_PATH

- 스탠다드 셀에 대한 .lib 파일 확인

22행 빨간색 상대경로에  
커서를 위치시킨 뒤 g+f

```
22 set STD_LIB_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gsc  
lib045/timing"  
23 set IO_LIB_PATH "../../../../../GPK045/digital/giolib045_v3.5/timing"  
24 set LEF_LIB_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gsc  
lib045/lef ../../../../../GPK045/digital/giolib045_v3.5/lef"  
25 set RTL_PATH "../RTL/"  
26 set QRC_FILE_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gsc  
lib045 tech/qrc/qx"
```

이 현황이 확인된다면 경로에  
오타, 오류가 없는 것임  
돌아오기 위해서는 Ctrl+o

10 README.txt  
11 fast\_vdd1v0\_basicCells.lib  
12 fast\_vdd1v0\_extvdd1v0.lib  
13 fast\_vdd1v0\_extvdd1v2.lib  
14 fast\_vdd1v0\_multibitsDFF.lib  
15 fast\_vdd1v2\_basicCells.lib  
16 fast\_vdd1v2\_extvdd1v0.lib  
17 fast\_vdd1v2\_extvdd1v2.lib  
18 fast\_vdd1v2\_multibitsDFF.lib  
19 report  
20 slow\_vdd1v0\_basicCells.lib  
21 slow\_vdd1v0\_extvdd1v0.lib  
22 slow\_vdd1v0\_extvdd1v2.lib  
23 slow\_vdd1v0\_multibitsDFF.lib  
24 slow\_vdd1v2\_basicCells.lib  
25 slow\_vdd1v2\_extvdd1v0.lib  
26 slow\_vdd1v2\_extvdd1v2.lib  
27 slow\_vdd1v2\_multibitsDFF.lib

# Synthesis

genus

파란 상자와 같이 현황이 확인된다  
면 경로에 오타, 오류가 없는 것임  
돌아오기 위해서는 Ctrl+o

경로확인\_1

23행 IO\_LIB\_PATH

- Io셀에 대한 .lib 파일 확인

10 pads\_FF\_s1vg.lib  
11 pads\_SS\_s1vg.lib  
12 pads\_TT\_s1vg.lib

23행 빨간색 상대경로에  
커서를 위치시킨 뒤 g+f

경로확인\_2

24행 LEF\_LIB\_PATH

- lef셀에 대한 .lef 파일 확인

.lef 파일이란?  
STD셀의 프레임 셀

```
22 set STD_LIB_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gsc  
lib045/timing"  
23 set IO_LIB_PATH "../../../../../GPK045/digital/giolib045_v3.5/timing"  
24 set LEF_LIB_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gsc  
lib045/lef ../../../../../GPK045/digital/giolib045_v3.5/lef"  
25 set RTL_PATH "../RTL/"  
26 set QRC_FILE_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gs  
clib045 tech/qrc/qx"
```

24행 빨간색 상대경로에  
커서를 위치시킨 뒤 g+f

10 gsclib045\_macro.lef  
11 gsclib045\_multibitsDFF.lef  
12 gsclib045\_tech.lef

# Synthesis

genus

경로확인\_3

25행 RTL\_PATH

- 작성한 RTL소스 코드에 대한 경로

25행 빨간색 상대경로에  
커서를 위치시킨 뒤 g+f

```
22 set STD_LIB_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gsc  
lib045/timing"  
23 set IO_LIB_PATH "../../../../../GPK045/digital/giolib045_v3.5/timing"  
24 set LEF_LIB_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gsc  
lib045/lef ../../../../../GPK045/digital/giolib045_v3.5/lef"  
25 set RTL_PATH      "../RTL/"  
26 set QRC_FILE_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gs  
clib045_tech/qrc/qx"
```

이 현황이 확인된다면 경로에  
오타, 오류가 없는 것임  
돌아오기 위해서는 Ctrl+o

```
10 memories_smkcw/  
11 memories_smkcw_1/  
12 verilog_smkcw/  
13 add_pad*  
14 cmsdk_ahb_memory_models_defs.v  
15 cmsdk_mcu.v  
16 cmsdk_mcu_defs.v
```

# Synthesis

genus

경로확인\_4

26행 QRC\_FILE\_PATH

- QRC 파일에 대한 경로

26행 빨간색 상대경로에  
커서를 위치시킨 뒤 g+f

```
22 set STD_LIB_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gsc  
lib045/timing"  
23 set IO_LIB_PATH "../../../../../GPK045/digital/giolib045_v3.5/timing"  
24 set LEF_LIB_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gsc  
lib045/lef ../../../../../GPK045/digital/giolib045_v3.5/lef"  
25 set RTL_PATH      "../RTL/"  
26 set QRC_FILE_PATH "../../../../../GPK045/digital/gsclib045_all_v4.4/gs  
clib045_tech/qrc/qx"
```

이 현황이 확인된다면 경로에  
오타, 오류가 없는 것임  
돌아오기 위해서는 Ctrl+o

8 ../  
9 ./  
10 README  
11 gpk045.tch  
12 hosts

# Synthesis

## genus

- 29행 ~ 31행은 사용할 셀과 파일을 지정
- 33행은 사용할 verilog 소스 코드를 지정

## Genus 툴 실행 결과

```
@genus:root: 24> set LIB_LIST {fast_vdd1v0_basicCells.lib slow_vdd1v0_basicCells.lib  
pads_SS_s1vg.lib}  
fast_vdd1v0_basicCells.lib slow_vdd1v0_basicCells.lib pads_SS_s1vg.lib  
@genus:root: 25> set LEF_LIST {gsclib045_tech.lef gsclib045_macro.lef giolib045.lef}  
gsclib045_tech.lef gsclib045_macro.lef giolib045.lef  
@genus:root: 26> set QRC_LIST {gpdk045.tch}  
gpdk045.tch  
@genus:root: 27>  
@genus:root: 27> set RTL_SOURCES "../RTL/cmsdk_mcu.v \  
==>           ../RTL/verilog_smkcow/cmsdk_mcu_clkctrl.v \  
==>           ../RTL/verilog_smkcow/cmsdk_mcu_pin_mux.v \  
==>           ../../../../../../cortexm0_designstart/implementation/cortex_m0_mcu_syste  
m_cadence_7ns/mapped/cmsdk_mcu_system_syn_final.v \  
==> "  
..../RTL/cmsdk_mcu.v ..../RTL/verilog_smkcow/cmsdk_mcu_clkctrl.v ..../RTL/verilog_smkcow/  
/cmsdk_mcu_pin_mux.v ..../../../../cortexm0_designstart/implementation/cortex_m0_mcu_sys  
tem_cadence_7ns/mapped/cmsdk_mcu_system_syn_final.v
```

```
28 #GPDK 045 SVT  
29 set LIB_LIST {fast_vdd1v0_basicCells.lib slow_vdd1v0_basicCells.lib  
pads_SS_s1vg.lib}  
30 set LEF_LIST {gsclib045_tech.lef gsclib045_macro.lef giolib045.lef}  
31 set QRC_LIST {gpdk045.tch}  
32  
33 set RTL_SOURCES "../RTL/cmsdk_mcu.v \  
34           ../RTL/verilog_smkcow/cmsdk_mcu_clkctrl.v \  
35           ../RTL/verilog_smkcow/cmsdk_mcu_pin_mux.v \  
36           ../../../../../../cortexm0_designstart/implementation/cortex_  
m0_mcu_system_cadence_7ns/mapped/cmsdk_mcu_system_syn_final.v \  
37 "  
38 set_db auto_ungroup none  
39 set_db max_cpus_per_server 2
```

# Synthesis

genus

```
/home/ex_poly1/SoC2/GPDK045/digital/gsclib045_all_v4.4/gsclib045/timing
```

```
$> vi README.txt
```

- 29행은 사용할 STD셀과 io셀 라이브러리를 지정
- 29행 LIB\_LIST와 해당 경로의 README.txt 파일에 동일한 파일 이름이 있는지 확인

```
7 Operating Conditions
8 =====
9 PROJECT SELECTION
10 ##fast_vdd1v0_basicCells.lib : PVT_1P1V_0C
11 ##slow_vdd1v0_basicCells.lib : PVT_0P9V_125C
12 =====
13 SUMMARY OF basicCells
14 ##fast_vdd1v0_basicCells.lib : PVT_1P1V_0C
15 ##fast_vdd1v2_basicCells.lib : PVT_1P32V_0C
16 ##slow_vdd1v0_basicCells.lib : PVT_0P9V_125C
17 ##slow_vdd1v2_basicCells.lib : PVT_1P08V_125C
18 =====
```

```
29 set LIB_LIST [fast_vdd1v0_basicCells.lib slow_vdd1v0_basicCells.lib
pads SS_s1vg.lib}
```

# Synthesis

genus

```
/home/ex_poly1/SoC2/GPDK045/digital/giolib045_v3.5/timing
```

pads\_FF\_s1vg.lib    pads\_SS\_s1vg.lib    pads\_TT\_s1vg.lib

vi 진행 방법

```
$> vi pads_FF_s1vg.lib
```

```
: vs
```

```
: e .
```

pads\_SS\_s1vg.lib -> Enter

```
2 delay model : typ  
3 check model :  
4 power model :  
5 capacitance model : 라이브러리 이름이  
6 other model : typ 같으므로 충돌 발생  
7 */  
8 library(giolib045) {  
9 }
```

```
2 delay model : typ  
3 check model : typ  
4 power model : typ  
5 capacitance model : typ  
6 other model : typ  
7 */  
8 library(giolib045) {  
9 }
```

Io셀 라이브러리 파일은  
하나만 기입

```
2 vdd1v0_basicCells.lib slow_vdd1v0_basicCells.lib  
pads_SS_s1vg.lib}
```

# Synthesis

## genus

### 30행 LEF\_LIST

- 사용할 LEF파일을 지정

```
30 set LEF_LIST {gsclib045_tech.lef gsclib045_macro.lef giolib045.lef}
```

### 31행 QRC\_LIST

- Parasitic을 뽑거나 Back-End 작업을 할 때 주로 사용되는 파일임

```
31 set QRC_LIST {gpdk045.tch}
```

# Synthesis

genus

## 33행 RTL\_LIST

- 사용할 verilog 파일을 지정
- 33행: IO셀을 포함한 TOP모듈
- 34행: clkctrl에 대한 Verilog 코드
- 35행: pin\_mux에 대한 Verilog 코드
- 36행: clkctrl과 pin\_mux를 제외한 나머지 모듈들에 대한 verilog 코드 (미리 완성한 Gate Level Netlist )

```
33 set RTL_SOURCES "../RTL/cmsdk_mcu.v \
34                 ../RTL/verilog_smkcowl/cmsdk_mcu_clkctrl.v \
35                 ../RTL/verilog_smkcowl/cmsdk_mcu_pin_mux.v \
36                 ../../../../cortexm0_designstart/implementation/cortex_ \
                         m0_mcu_system_cadence_7ns/mapped/cmsdk_mcu_system_syn_final.v \
```

# Synthesis

Genus 툴 실행 결과

genus

```
@genus:root: 28> set_db auto_ungroup none
    Setting attribute of root '/': 'auto_ungroup' = none
1 none
@genus:root: 29> set_db max_cpus_per_server 2
    Setting attribute of root '/': 'max_cpus_per_server' = 2
1 2
```

- set\_db는 genus에서 환경을 설정하는 명령임

```
28 #GDK 045 SVT
29 set LIB_LIST {fast_vdd1v0_basicCells.lib slow_vdd1v0_basicCells.lib
  pads_SS_s1vg.lib}
30 set LEF_LIST {gsclib045_tech.lef gsclib045_macro.lef giolib045.lef}
31 set QRC_LIST {gdk045.tch}
32
33 set RTL_SOURCES "../RTL/cmsdk_mcu.v \
34           ../RTL/verilog_smkcow/cmsdk_mcu_clkctrl.v \
35           ../RTL/verilog_smkcow/cmsdk_mcu_pin_mux.v \
36           ../../cortexm0_designstart/implementation/cortex_
  m0_mcu_system_cadence_7ns/mapped/cmsdk_mcu_system_syn_final.v \
37 "
38 set_db auto_ungroup none
39 set db max cpus per server 2
```

# Synthesis

genus

Effort는 high에 가까울수록 좋은 결과를 기대할 수 있음  
(Default 값은 medium)

Genus 툴 실행 결과

```
@genus:root: 30> set_db syn_generic_effort high; # Default medium
Setting attribute of root '/': 'syn_generic_effort' = high
1 high
@genus:root: 31> set_db syn_map_effort high;      # Default medium
Setting attribute of root '/': 'syn_map_effort' = high
1 high
@genus:root: 32> set_db syn_opt_effort high;      # Default medium
Setting attribute of root '/': 'syn_opt_effort' = high
1 high
```

```
41 ##Set synthesizing effort for each synthesis stage
42 set_db syn_generic_effort high; # Default medium
43 set_db syn_map_effort high;      # Default medium
44 set_db syn_opt_effort high;      # Default medium
45
46 set_db init_lib_search_path [concat $STD_LIB_PATH $]
47 set_db init_hdl_search_path [concat $RTL_PATH]
48
49 read_libs ${LIB_LIST}
50 read_physical -lef ${LEF_LIST}
51 read_qrc ${QRC_LIST}
52
53 ##generates <signal>_reg[<bit_width>] format
54 set_db hdl_array_naming_style %s_%d
55 set_db / .uniquify_naming_style ${TOP DESIGN}_%s_%d
```

- set\_db는 genus에서 환경을 설정하는 명령임

# Synthesis

## Genus 툴 실행 결과

### genus

- 최초 라이브러리의 검색 경로를 연결시켜 줌
- hdl소스를 어디서 읽을 것인지 경로를 연결시켜 줌

```
@genus:root: 33> set_db init_lib_search_path [concat $STD_LIB_PATH $IO_LIB_PATH $LEF_LIB_PATH $QRC_FILE_PATH]
  Setting attribute of root '/': 'init_lib_search_path' = ../../../../../../GPKD045/digital/gsclib045_all_v4.4/gsclib045/timing ../../../../../../GPKD045/digital/giolib045_v3.5/timing ../../../../../../GPKD045/digital/gsclib045_all_v4.4/gsclib045/lef ../../../../../../GPKD045/digital/giolib045_v3.5/lef ../../../../../../GPKD045/digital/gsclib045_all_v4.4/gsclib045_tech/qrc/qx
1 {../../../../GPKD045/digital/gsclib045_all_v4.4/gsclib045/timing ../../../../../../GPKD045/digital/giolib045_v3.5/timing ../../../../../../GPKD045/digital/gsclib045_all_v4.4/gsclib045/lef ../../../../../../GPKD045/digital/giolib045_v3.5/lef ../../../../../../GPKD045/digital/gsclib045_all_v4.4/gsclib045_tech/qrc/qx}
@genus:root: 34> set_db init_hdl_search_path [concat $RTL_PATH]
  Setting attribute of root '/': 'init_hdl_search_path' = ./RTL/
1 ./RTL/
```

```
41 ##Set synthesizing effort for each synthesis stage
42 set_db syn_generic_effort high; # Default medium
43 set_db syn_map_effort high;      # Default medium
44 set_db syn_opt_effort high;     # Default medium
45
46 set_db init_lib_search_path [concat $STD_LIB_PATH $IO_LIB_PATH $LEF_LIB_PATH $QRC_FILE_PATH]
47 set_db init_hdl_search_path [concat $RTL_PATH]
48
49 read_libs ${LIB_LIST}
50 read_physical -lef ${LEF_LIST} 
51 read_qrc ${QRC_LIST}
52
53 ##generates <signal>_re mat
54 set_db hdl_array_naming_style %s_%d
55 set_db / .uniquify_naming_style ${TOP DESIGN}_%s_%d
```

연결한다는 뜻을  
가지고 있음

# Synthesis

## Genus 툴 실행 결과

genus

- 29행의 LIB\_LIST 파일을 읽음

```
@genus:root: 35> read_libs ${LIB_LIST}

Message Summary for Library all 3 libraries:
*****
Missing a function attribute in the output pin definition. [LBR-518]: 2
Missing library level attribute. [LBR-516]: 1
An unsupported construct was detected in this library. [LBR-40]: 16
*****


Warning : Libraries have inconsistent nominal operating conditions. In the Liberty l
ibrary, there are attributes called nom_voltage, nom_process and nom_temperature. Ge
nus reports the message, if the respective values of the 2 given .libs differ. [LBR-
38]
      : The libraries are 'fast_vdd1v0' and 'slow_vdd1v0'.
      : This is a common source of delay calculation confusion and should be avoid
ed.
Warning : Libraries have inconsistent nominal operating conditions. In the Liberty l
```

```
41 ##Set synthesizing effort for each synthesis stage
42 set_db syn_generic_effort high; # Default medium
43 set_db syn_map_effort high;      # Default medium
44 set_db syn_opt_effort high;     # Default medium
45
46 set_db init_lib_search_path [concat $STD_LIB_PATH $IO_LIB_PATH $LEF_LIB_PATH $QRC_FILE_PATH]
47 set_db init_hdl_search_path [concat $RTL_PATH]
48
49 read_libs ${LIB_LIST}
50 read_physical -lef ${LEF_LIST}
51 read_qrc ${QRC_LIST}
52
53 ##generates <signal>_reg[<bit_width>] format
54 set_db hdl_array_naming_style %s_%d
55 set_db / .uniquify_naming_style ${TOP DESIGN}_%s_%d
```

# Synthesis

## Genus 툴 실행 결과

genus

- 30행의 LEF\_LIST 파일을 읽음

```
@genus:root: 36> read_physical -lef ${LEF_LIST}
Info      : Via with no resistance will have a value of '0.0' assigned for resistance
value. [PHYS-129]
          : Via 'M2_M1_HV' has no resistance value.
          : If this is the expected behavior, this message can be ignored.
Info      : Via with no resistance will have a value of '0.0' assigned for resistance
value. [PHYS-129]
          : Via 'M2_M1_VV' has no resistance value.
Info      : Via with no resistance will have a value of '0.0' assigned for resistance
value. [PHYS-129]
          : Via 'M2_M1_VH' has no resistance value.
Info      : Via with no resistance will have a value of '0.0' assigned for resistance
value. [PHYS-129]
```

```
41 ##Set synthesizing effort for each synthesis stage
42 set_db syn_generic_effort high; # Default medium
43 set_db syn_map_effort high;      # Default medium
44 set_db syn_opt_effort high;     # Default medium
45
46 set_db init_lib_search_path [concat $STD_LIB_PATH $IO_LIB_PATH $LEF_LIB_PATH $QRC_FILE_PATH]
47 set_db init_hdl_search_path [concat $RTL_PATH]
48
49 read_libs ${LTB_LTST}
50 read_physical -lef ${LEF_LIST} // This line is highlighted with a red box.
51 read_qrc ${QRC_LIST}
52
53 ##generates <signal>_reg[<bit_width>] format
54 set_db hdl_array_naming_style %s_%d
55 set_db / .uniquify_naming_style ${TOP DESIGN}_%s_%d
```

# Synthesis

## Genus 툴 실행 결과

genus

```
@genus:root: 37> read_qrc ${QRC_LIST}
```

```
According to qrc_tech_file, there are total 11 routing layers [ V(5) / H(6) ]
```

```
Done reading qrc_tech_file
```

- 31행의 QRC\_LIST 파일을 읽음

```
41 ##Set synthesizing effort for each synthesis stage
42 set_db syn_generic_effort high; # Default medium
43 set_db syn_map_effort high;      # Default medium
44 set_db syn_opt_effort high;     # Default medium
45
46 set_db init_lib_search_path [concat $STD_LIB_PATH $IO_LIB_PATH $LEF_LIB_PATH $QRC_FILE_PATH]
47 set_db init_hdl_search_path [concat $RTL_PATH]
48
49 read_libs ${LIB_LIST}
50 read_physical -lef ${LEF_LIST}
51 read_qrc ${QRC_LIST} // Line 51 is highlighted with a red box
52
53 ##generates <signal>_reg[<bit_width>] format
54 set_db hdl_array_naming_style %s_%d
55 set_db / .uniquify_naming_style ${TOP DESIGN}_%s_%d
```

# Synthesis

## Genus 툴 실행 결과

genus

```
@genus:root: 38> ##generates <signal>_reg[<bit_width>] format
@genus:root: 39> set_db hdl_array_naming_style %s_%d
    Setting attribute of root '/': 'hdl_array_naming_style' = %s_%d
1 %s_%d
@genus:root: 40> set_db / .uniquify_naming_style ${TOP DESIGN}_%s_%d
    Setting attribute of root '/': 'uniquify_naming_style' = cmsdk_mcu_%s_%d
1 cmsdk_mcu_%s_%d
```

- 디자인 내에서 instance 이름 충돌을 방지하기 위해 이름 생성 과정에서 규칙을 설정함

```
41 ##Set synthesizing effort for each synthesis stage
42 set_db syn_generic_effort high; # Default medium
43 set_db syn_map_effort high;      # Default medium
44 set_db syn_opt_effort high;     # Default medium
45
46 set_db init_lib_search_path [concat $STD_LIB_PATH $IO_LIB_PATH $LEF_LIB_PATH $QRC_FILE_PATH]
47 set_db init_hdl_search_path [concat $RTL_PATH]
48
49 read_libs ${LIB_LIST}
50 read_physical -lef ${LEF_LIST}
51 read_qrc ${QRC_LIST}
52
53 ##generates <signal>_req[<bit width>] format
54 set_db hdl_array_naming_style %s_%d
55 set_db / .uniquify_naming_style ${TOP DESIGN}_%s_%d
```

# Synthesis

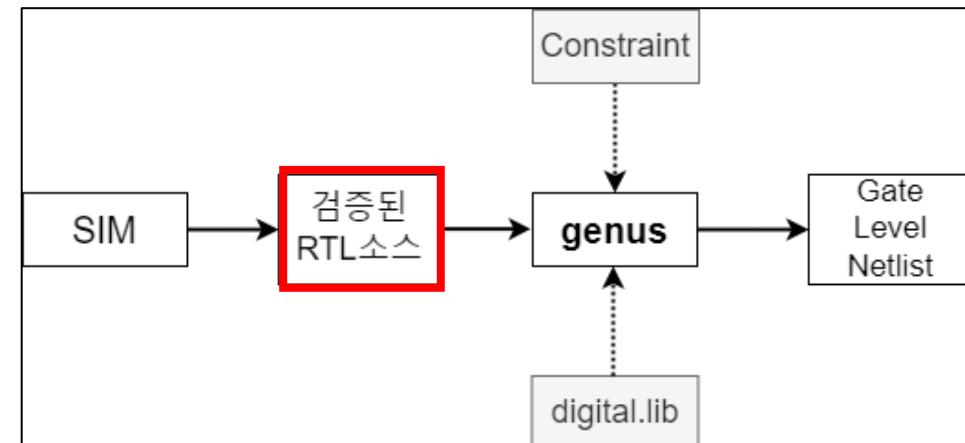
## Genus 툴 실행 결과

genus

```
@genus:root: 41> read_hdl $RTL_SOURCES  
@genus:root: 42>
```

```
57 #####  
58 ## Load Design  
59 #####  
60  
61 #read_hdl -f $RTL_SOURCES -define Func_mode -define cadence_045nm  
62 # -f option is for the case that all files are defined in a specific  
file  
63 read_hdl $RTL_SOURCES  
64  
65 elaborate ${TOP DESIGN}  
66 puts "Runtime & Memory after 'read_hdl'"  
67 time_info Elaboration  
68  
69 set_db hinst:u_cmsdk_mcu_system .preserve true  
70 set_db optimize_merge_flops false  
71 set_db optimize_merge_latches false  
72  
73 uniquify -verbose ${TOP DESIGN}  
74  
75 check_design > ${RPT_DIR}/00_check_design.rpt
```

- 검증된 RTL소스를 불러옴



# Synthesis

## Genus 툴 실행 결과

genus

- 읽어 들인 RTL소스 파일을 elaborate

```
57 #####
58 ## Load Design
59 #####
60
61 #read_hdl -f $RTL_SOURCES -define Func_mode -define cadenc
62 # -f option is for the case that all files are defined in
   file
63 read_hdl $RTL_SOURCES
64
65 elaborate ${TOP_DESIGN}
66 puts "Runtime & Memory after 'read_hdl'"
67 time_info Elaboration
68
69 set_db hinst:u_cmsdk_mcu_system .preserve true
70 set_db optimize_merge_flops false
71 set_db optimize_merge_latches false
72
73 uniquify -verbose ${TOP_DESIGN}
74
75 check_design > ${RPT_DIR}/00_check_design.rpt
```

```
@genus:root: 42> elaborate ${TOP_DESIGN}
Libraries have 324 usable logic and 128 usable sequential lib-cells.
Info      : Elaborating Design. [ELAB-1]
            : Elaborating top-level block 'cmsdk_mcu' from file '../RTL/../RTL/cmsdk_mcu.v'.
Warning   : Using default parameter value for module elaboration. [CDFG-818]
            : Elaborating block 'cmsdk_mcu' with default parameters value.
Warning   : Removing unused register. [CDFG-508]
            : Removing unused flip-flop register 'dbgrst_reg' in module 'cmsdk_mcu_clkctrl_CLKGATE_PRESENT0' in file '../RTL/../RTL/verilog_smkcw/cmsdk_mcu_clkctrl.v' on line 115.
            : Genus removes the flip-flop or latch inferred for an unused signal or variable. To preserve the flip-flop or latch, set the hdl_preserve_unused_registers attribute to true or use a pragma in the RTL.
```

Stage: post\_elab

Trick	Accepts	Rejects	Runtime (s)
ume_constant_bmux	0	0	0.00
ume_merge	0	0	0.00
ume_ssm	0	0	0.00
ume_cse	0	0	0.01
ume_shrink	0	0	0.00
ume_sweep	0	0	0.00

Stage: post\_elab

Transform	Accepts	Rejects	Runtime (s)
hlo_optimize_datapath_shifters	0	0	0.00
hlo_clip_mux_input	0	0	0.00
hlo_clip	0	0	0.00
hlo_cleanup	0	0	0.00

```
UM:    flow.cputime    flow.realtime    timing.setup.tns    timing.setup.wns    snapshot
UM:*    elaborate
```

# Synthesis

## Genus 툴 실행 결과

genus

```
@genus:root: 43> puts "Runtime & Memory after 'read_hdl'"  
Runtime & Memory after 'read_hdl'
```

```
57 #####  
58 ## Load Design  
59 #####  
60  
61 #read_hdl -f $RTL_SOURCES -define Func_mode -define cadence_045nm  
62 # -f option is for the case that all files are defined in a specific  
file  
63 read_hdl $RTL_SOURCES  
64  
65 elaborate ${TOP DESIGN}  
66 puts "Runtime & Memory after 'read_hdl'"  
67 time_info Elaboration  
68  
69 set_db hinst:u_cmsdk_mcu_system .preserve true  
70 set_db optimize_merge_flops false  
71 set_db optimize_merge_latches false  
72  
73 uniquify -verbose ${TOP DESIGN}  
74  
75 check_design > ${RPT_DIR}/00_check_design.rpt
```

- puts는 메시지를 출력하는 명령임

# Synthesis

## Genus 툴 실행 결과

genus

```
@genus:root: 44> time_info Elaboration
stamp 'Elaboration' being created for table 'default'

Total Time (Wall) | Stage Time (Wall) | % (Wall) | Date - Time | Memory | Stage
-----+-----+-----+-----+-----+-----+
00:00:08(00:00:00) | 00:00:00(00:00:00) | 0.0( 0.0) | 20:44:02 (Jan26) | 241.0 MB | init
-----+-----+-----+-----+-----+
00:00:46(01:00:25) | 00:00:38(01:00:25) | 100.0(100.0) | 21:44:27 (Jan26) | 531.7 MB | Elaboration
-----+-----+-----+-----+-----+
Number of threads: 2 * 1 (id: default, time_info v1.57)
Info: (*N*) indicates data that was populated from previously saved time_info database
Info: CPU time includes time of parent + longest thread
```

```
57 #####
58 ## Load Design
59 #####
60
61 #read_hdl -f $RTL_SOURCES -define Func_mode -define cadence_045nm
62 # -f option is for the case that all files are defined in a specific
   file
63 read_hdl $RTL_SOURCES
64
65 elaborate ${TOP DESIGN}
66 puts "Runtime & Memory after 'read_hdl'"
67 time_info Elaboration
68
69 set_db hinst:u_cmsdk_mcu_system .preserve true
70 set_db optimize_merge_flops false
71 set_db optimize_merge_latches false
72
73 uniquify -verbose ${TOP DESIGN}
74
75 check_design > ${RPT_DIR}/00_check_design.rpt
```

- Elaborate에서 소요된 시간과 메모리를 확인하는 명령임

# Synthesis

genus

## Genus 툴 실행 결과

```
@genus:root: 45> set_db hinst:u_cmsdk_mcu_system .preserve true
    Setting attribute of hinst 'u_cmsdk_mcu_system': 'preserve' = true
1 true
@genus:root: 46> set_db optimize_merge_flops false
    Setting attribute of root '/': 'optimize_merge_flops' = false
1 false
@genus:root: 47> set_db optimize_merge_latches false
    Setting attribute of root '/': 'optimize_merge_latches' = false
1 false
```

```
57 #####
58 ## Load Design
59 #####
60
61 #read_hdl -f $RTL_SOURCES -define Func_mode -define cadence_045nm
62 # -f option is for the case that all files are defined in a specific
   file
63 read_hdl $RTL_SOURCES
64
65 elaborate ${TOP DESIGN}
66 puts "Runtime & Memory after 'read_hdl'"
67 time_info Elaboration
68
69 set_db hinst:u_cmsdk_mcu_system .preserve true
70 set_db optimize_merge_flops false
71 set_db optimize_merge_latches false
72
73 uniquify -verbose ${TOP DESIGN}
74
75 check_design > ${RPT_DIR}/00_check_design.rpt
```

- 기존의 u\_cmsdk\_mcu\_system 모듈은 건드리지 않고 보존할 것임
- Set\_don't\_touch와 비슷한 개념임

# Synthesis

## Genus 툴 실행 결과

genus

```
@genus:root: 48> uniquify -verbose ${TOP DESIGN}
Uniquified instance: u_pin_mux/mux_119_30, and its subdesign name is: cmsdk_mcu_cmsdk_mcu_bmux_1_10
Uniquified instance: u_pin_mux/mux_117_30, and its subdesign name is: cmsdk_mcu_cmsdk_mcu_bmux_1_9
Uniquified instance: u_pin_mux/mux_115_30, and its subdesign name is: cmsdk_mcu_cmsdk_mcu_bmux_1_8
Uniquified instance: u_pin_mux/mux_130_33, and its subdesign name is: cmsdk_mcu_cmsdk_mcu_bmux_1_7
Uniquified instance: u_pin_mux/mux_128_33, and its subdesign name is: cmsdk_mcu_cmsdk_mcu_bmux_1_6
Uniquified instance: u_pin_mux/mux_126_33, and its subdesign name is: cmsdk_mcu_cmsdk_mcu_bmux_1_5
Uniquified instance: u_cmsdk_mcu_clkctrl/mux_165_22, and its subdesign name is: cmsdk_mcu_cmsdk_mcu
bmux_1_4
Uniquified instance: u_cmsdk_mcu_clkctrl/mux_166_22, and its subdesign name is: cmsdk_mcu_cmsdk_mcu
bmux_1_3
```

```
57 #####
58 ## Load Design
59 #####
60
61 #read_hdl -f $RTL_SOURCES -define Func_mode -define cadence_045nm
62 # -f option is for the case that all files are defined in a specific
   file
63 read_hdl $RTL_SOURCES
64
65 elaborate ${TOP DESIGN}
66 puts "Runtime & Memory after 'read_hdl'"
67 time_info Elaboration
68
69 set_db hinst:u_cmsdk_mcu_system .preserve true
70 set_db optimize_merge_flops false
71 set_db optimize_merge_latches false
72
73 uniquify -verbose ${TOP DESIGN}
74
75 check_design > ${RPT_DIR}/00_check_design.rpt
```

- 디자인 내에서 instance 이름을 겹치지 않도록 정리해주는 명령임

# Synthesis

genus

- RTL소스의 무결성을 확인하는 명령임

```
57 #####
58 ## Load Design
59 #####
60
61 #read_hdl -f $RTL_SOURCES -define Func_mode -define cadence_045nm
62 # -f option is for the case that all files are defined in a specific
   file
63 read_hdl $RTL_SOURCES
64
65 elaborate ${TOP_DESIGN}
66 puts "Runtime & Memory after 'read_hdl'"
67 time_info Elaboration
68
69 set_db hinst:u_cmsdk_mcu_system .preserve true
70 set_db optimize_merge_flops false
71 set_db optimize_merge_latches false
72
73 uniquify -verbose ${TOP_DESIGN}
74
75 check_design > ${RPT_DIR}/00_check_design.rpt
```

## Genus 툴 실행 결과

큰 문제가 없음을  
확인함

Check Design Report (c)

Summary

Name	Total
Unresolved References	0
Empty Modules	0
Unloaded Port(s)	0
Unloaded Sequential Pin(s)	0
Unloaded Combinational Pin(s)	11
Assigns	137
Undriven Port(s)	0
Undriven Leaf Pin(s)	0
Undriven hierarchical pin(s)	1241
Multidriven Port(s)	0
Multidriven Leaf Pin(s)	0
Multidriven hierarchical Pin(s)	0
Multidriven unloaded net(s)	0
Constant Port(s)	0
Constant Leaf Pin(s)	880
Constant hierarchical Pin(s)	56
Preserved leaf instance(s)	13566
Preserved hierarchical instance(s)	30
Feedthrough Modules(s)	0
Libcells with no LEF cell	4
Physical (LEF) cells with no libcell	107
Subdesigns with long module name	0
Physical only instance(s)	0
Logical only instance(s)	0

Done Checking the design.

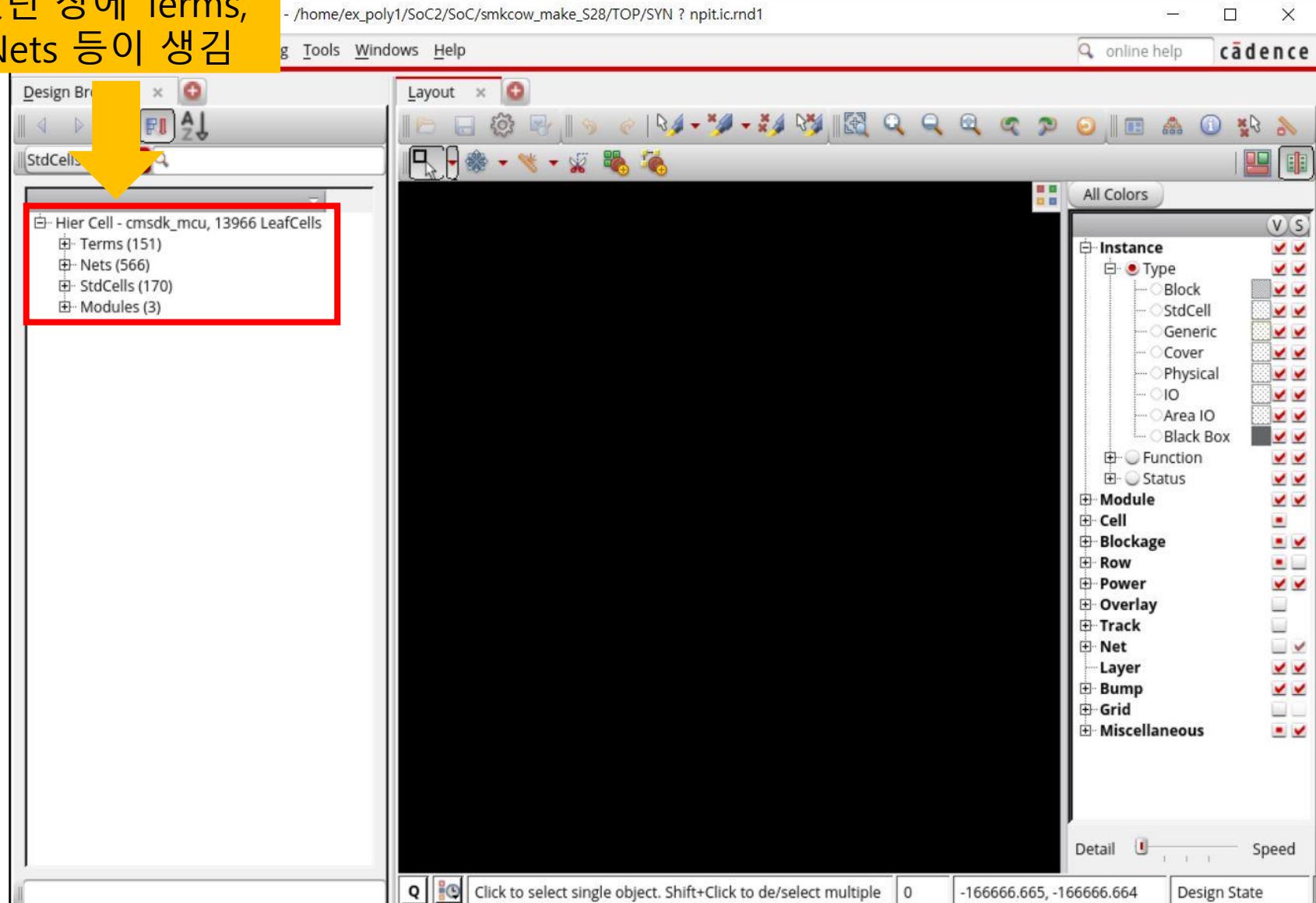
# Synthesis

genus

\$> gui\_show

- Check Design 후 gui 확인

처음엔 아무것도  
없던 창에 Terms,  
Nets 등이 생김

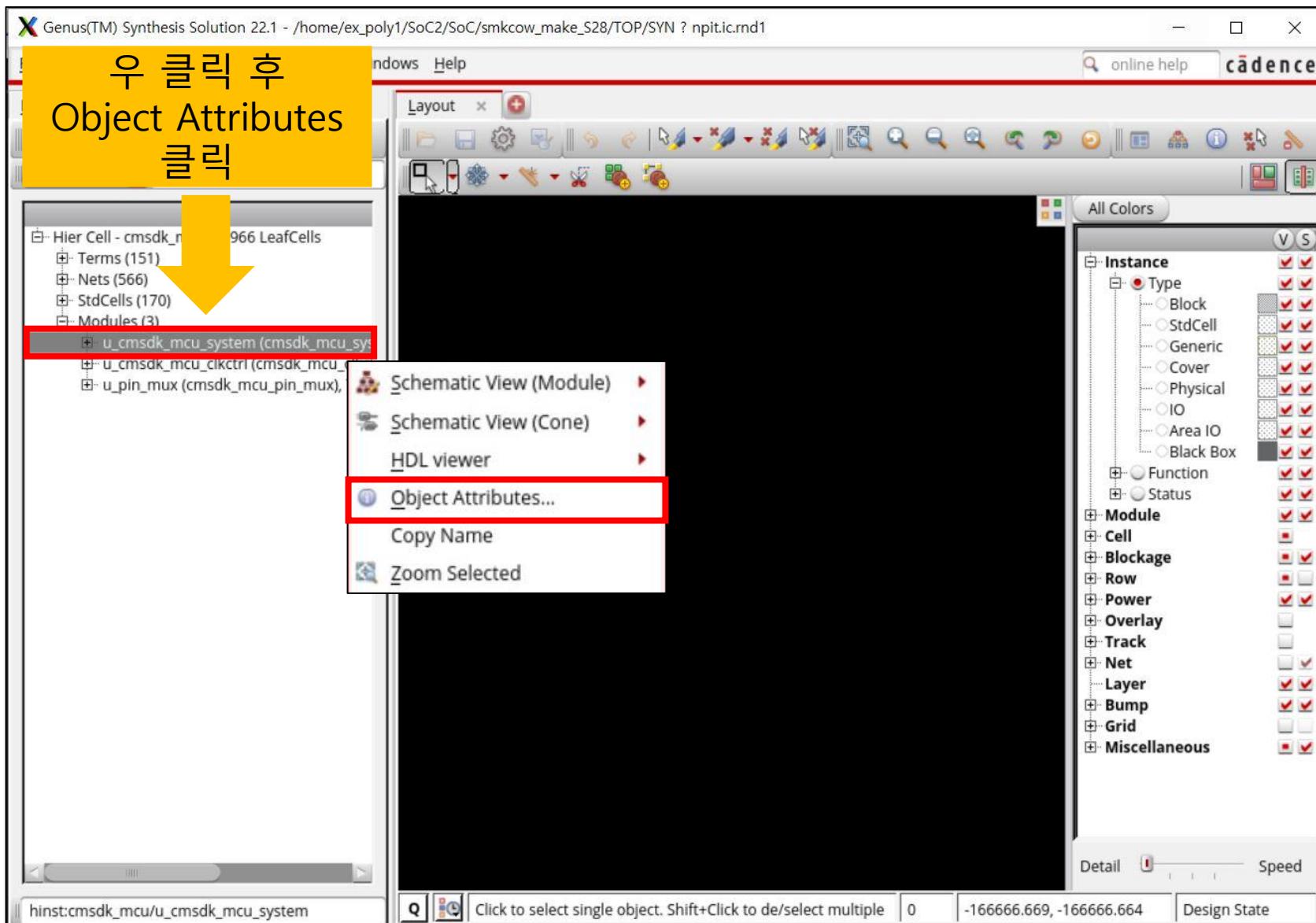


# Synthesis

genus

\$> **gui\_show**

- **gui 확인**

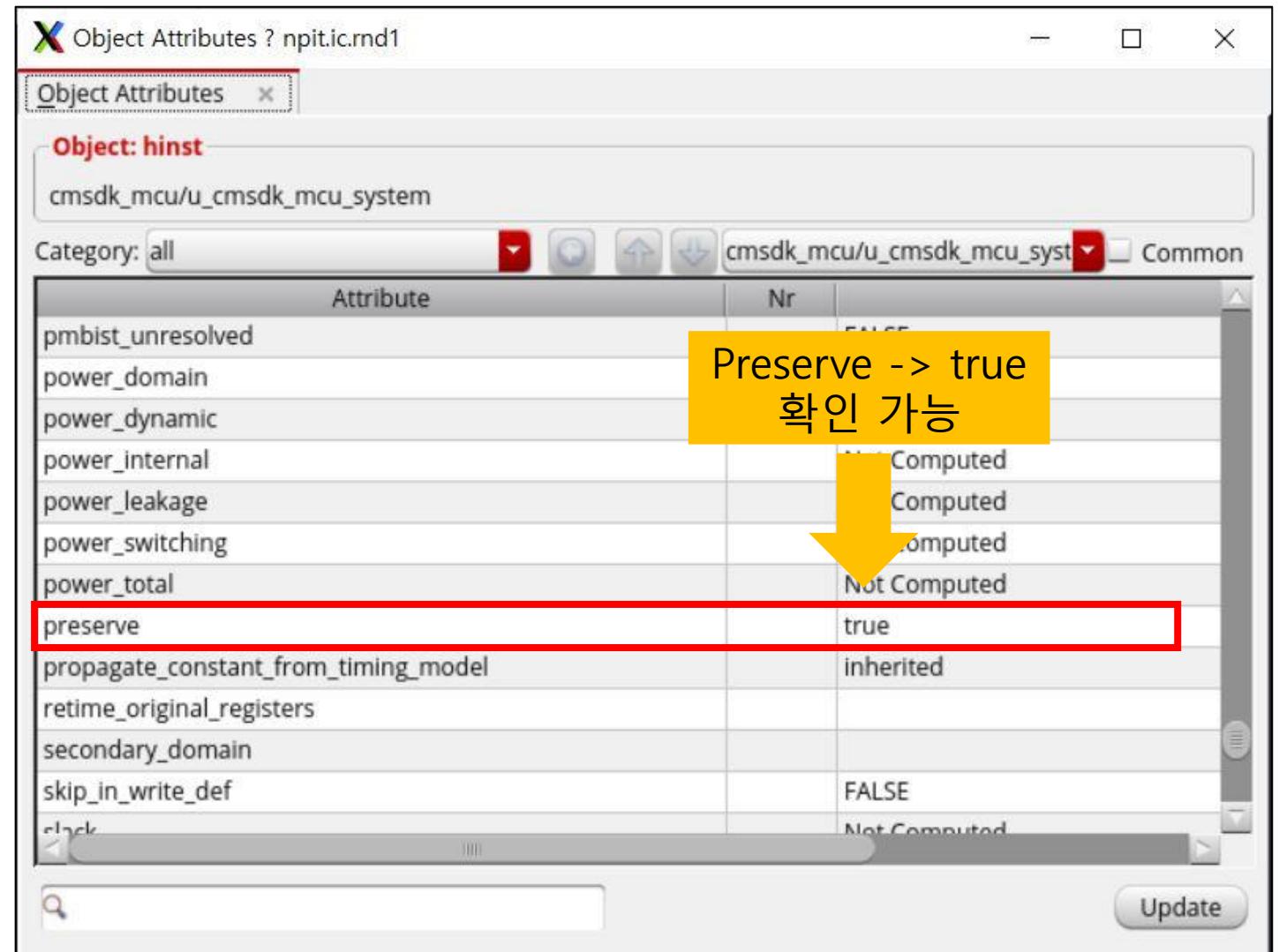


# Synthesis

genus

\$> gui\_show

- gui 확인

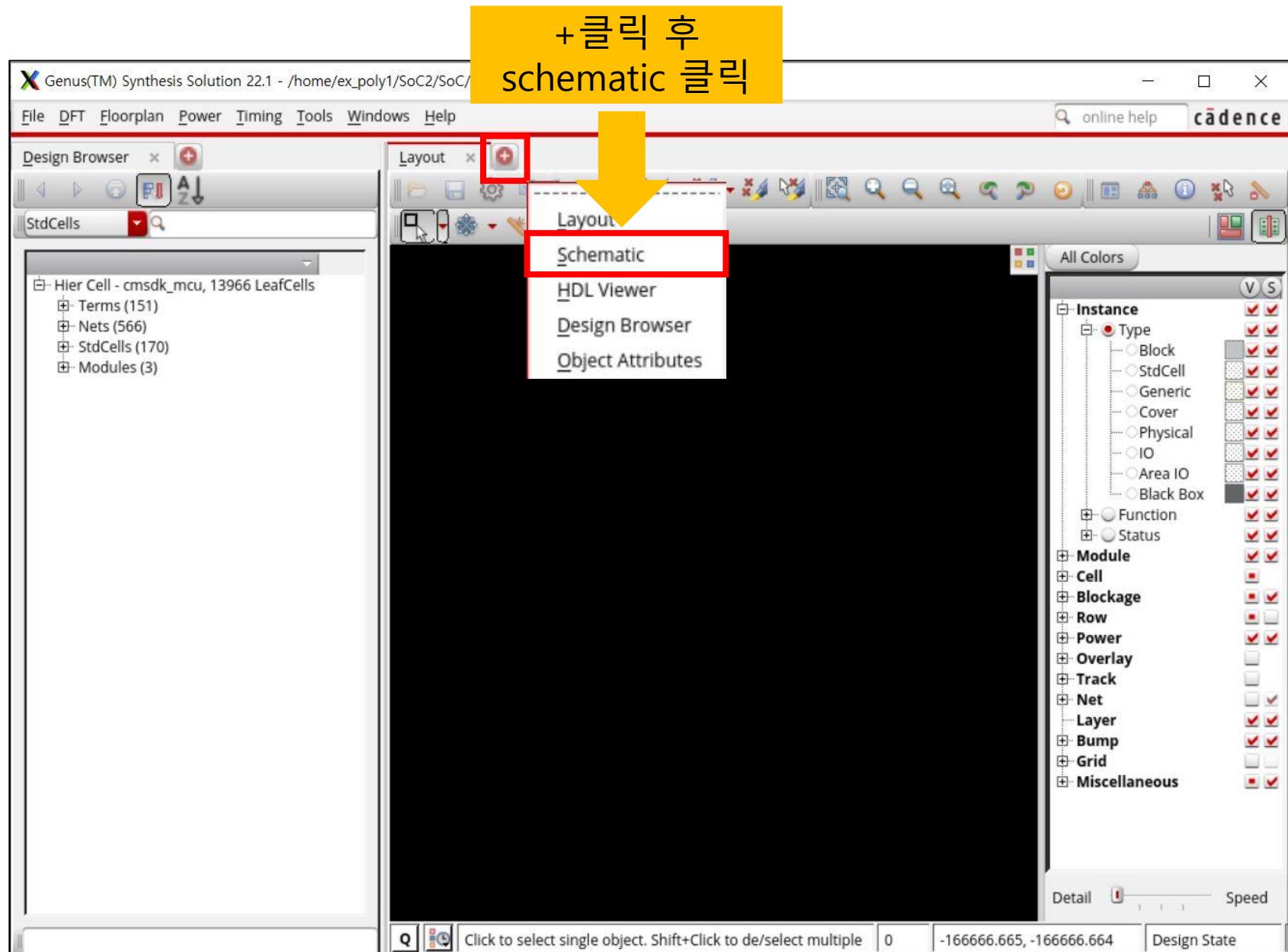


# Synthesis

genus

\$> **gui\_show**

- **gui 확인**

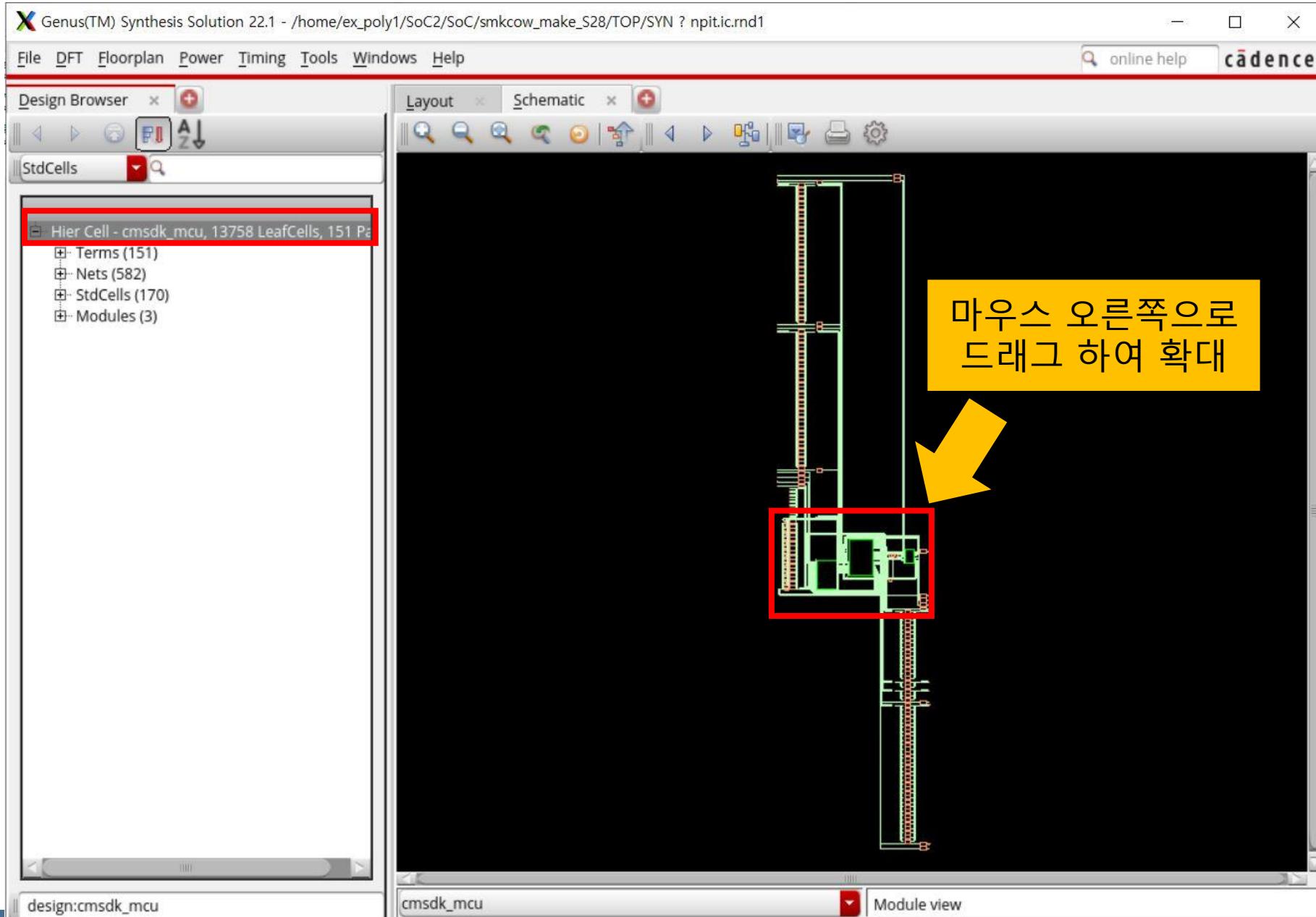


# Synthesis

genus

\$> **gui\_show**

- Gui로 TOP모듈 확인



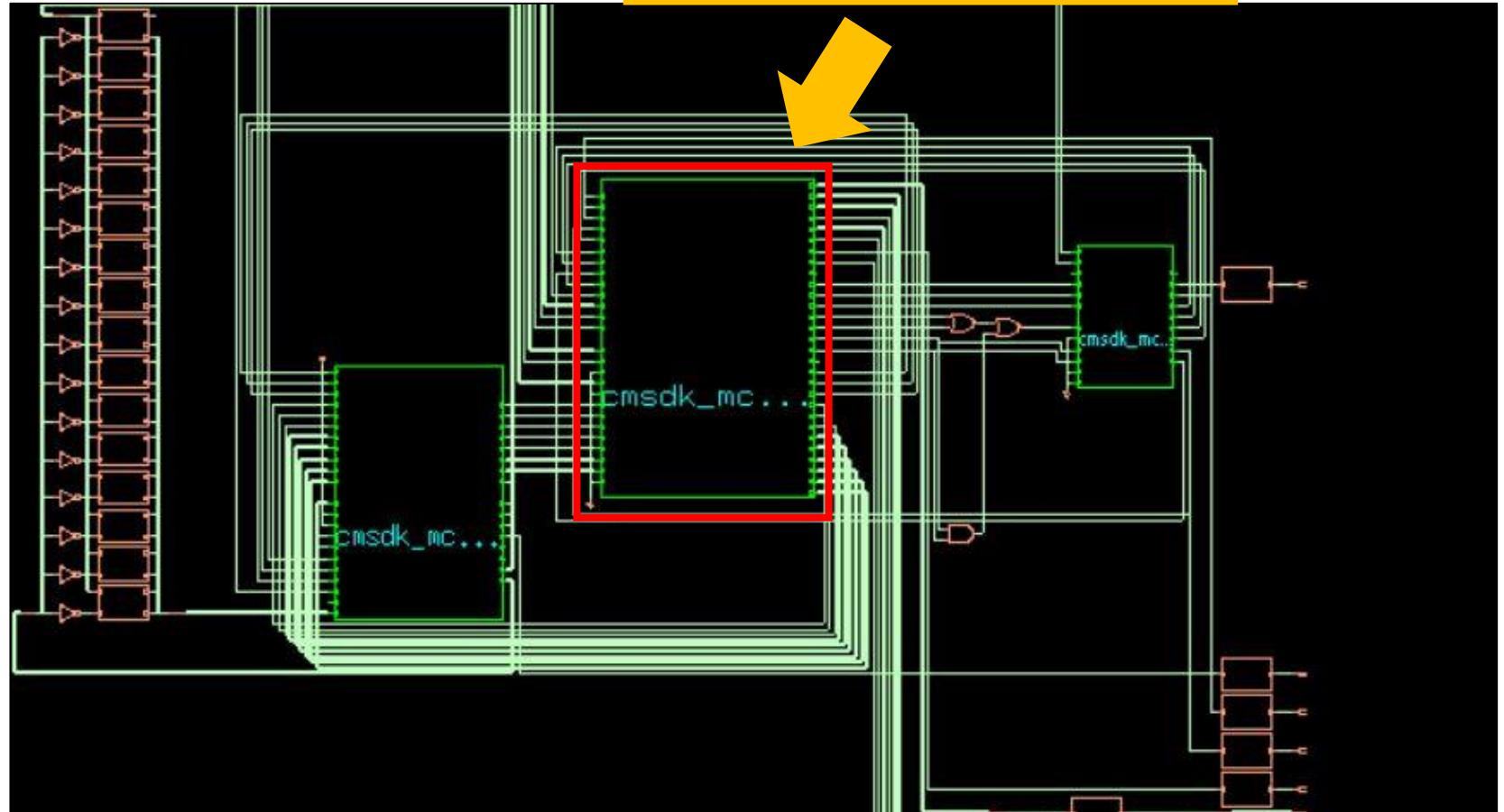
# Synthesis

genus

\$> **gui\_show**

- gui 확인

미리 합성했던 모듈도 잘  
load 되어있음을 확인  
해당 모듈은 Gate Mapping이  
끝나 있음

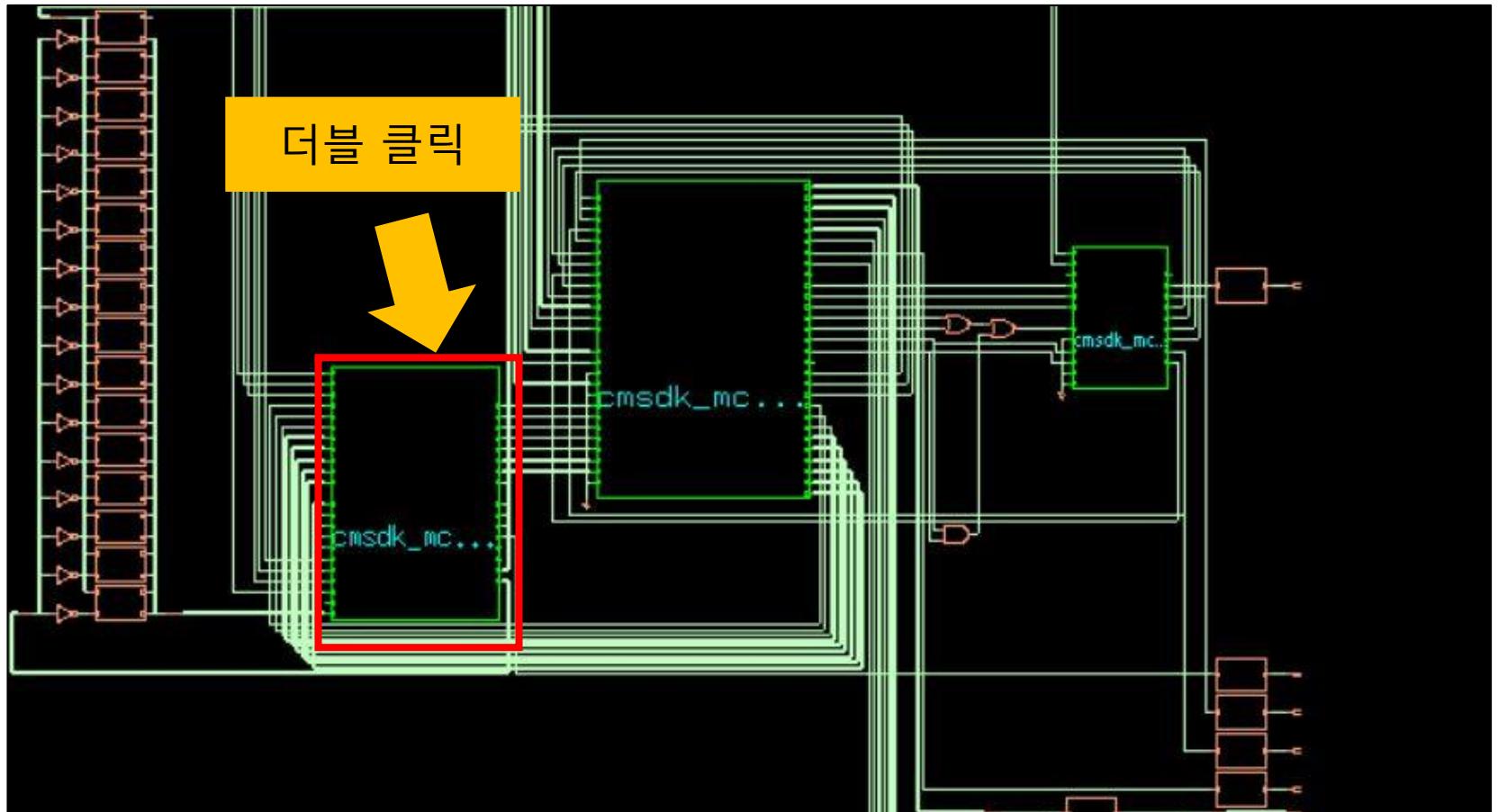


# Synthesis

genus

\$> **gui\_show**

- 합성 전 STD셀 이름 확인



# Synthesis

genus

\$> **gui\_show**

- 합성 전 STD셀 이름 확인

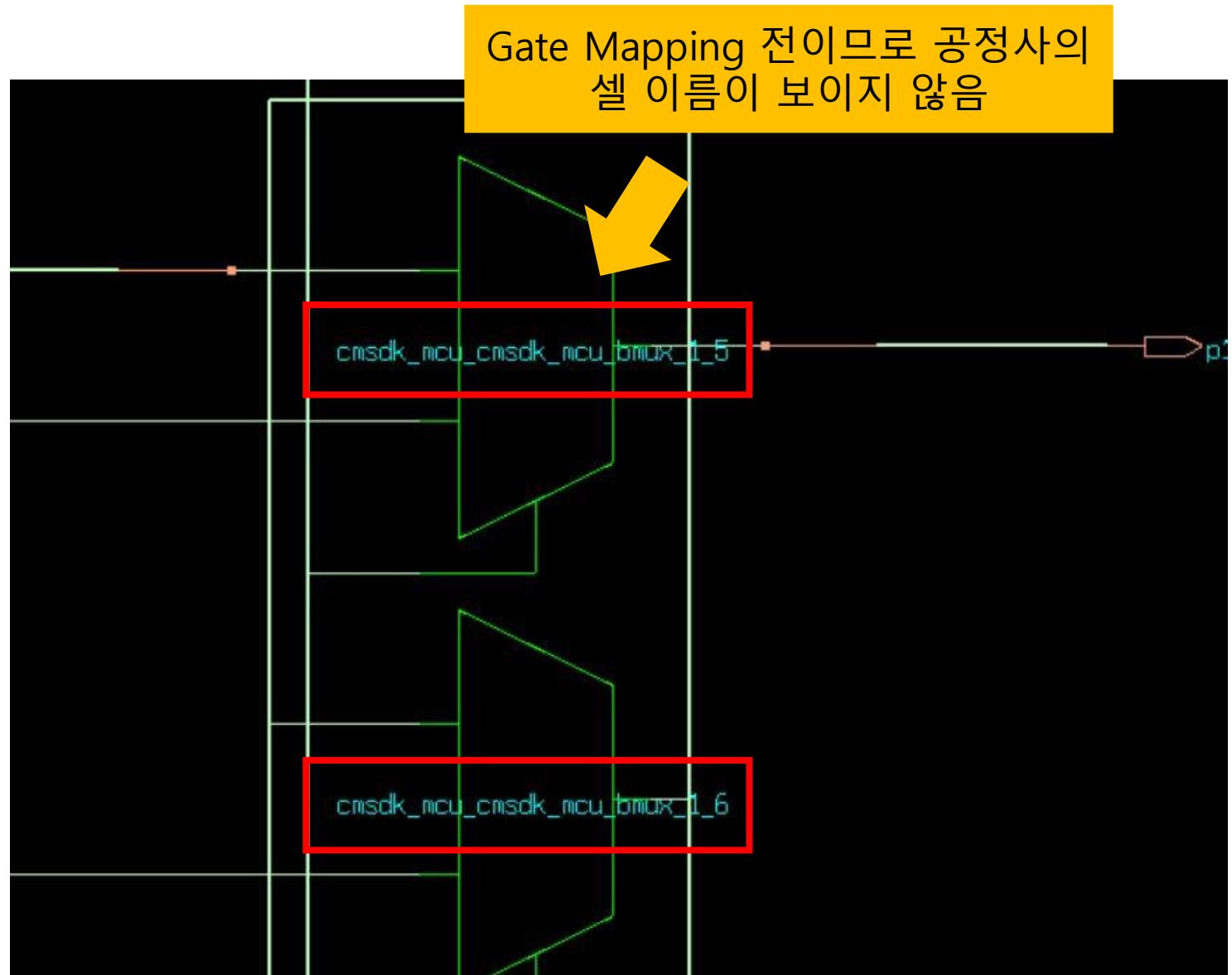


# Synthesis

genus

\$> **gui\_show**

- 합성 전 STD셀 이름 확인



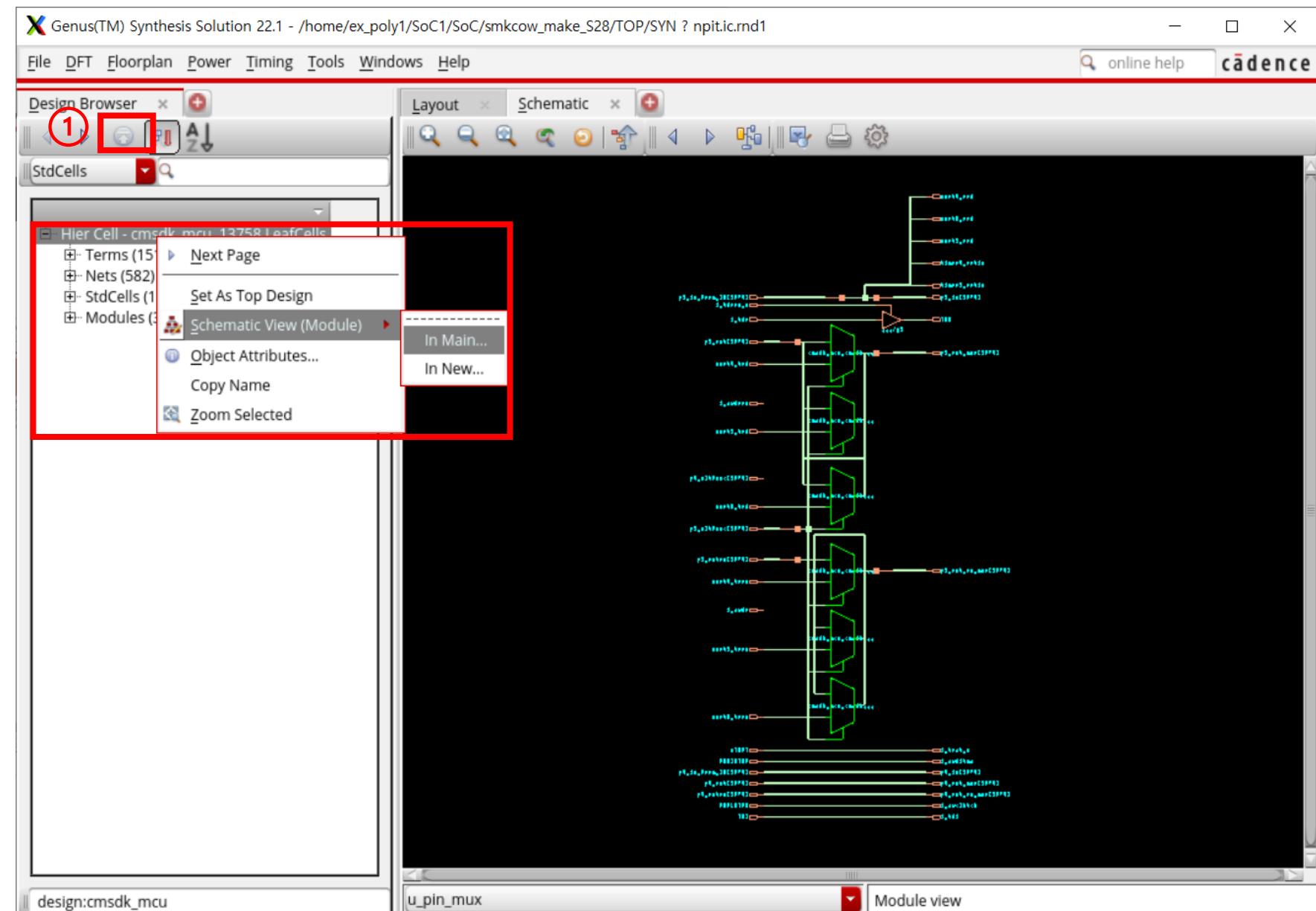
# Synthesis

genus

- TOP 디자인으로 이동

1번 클릭

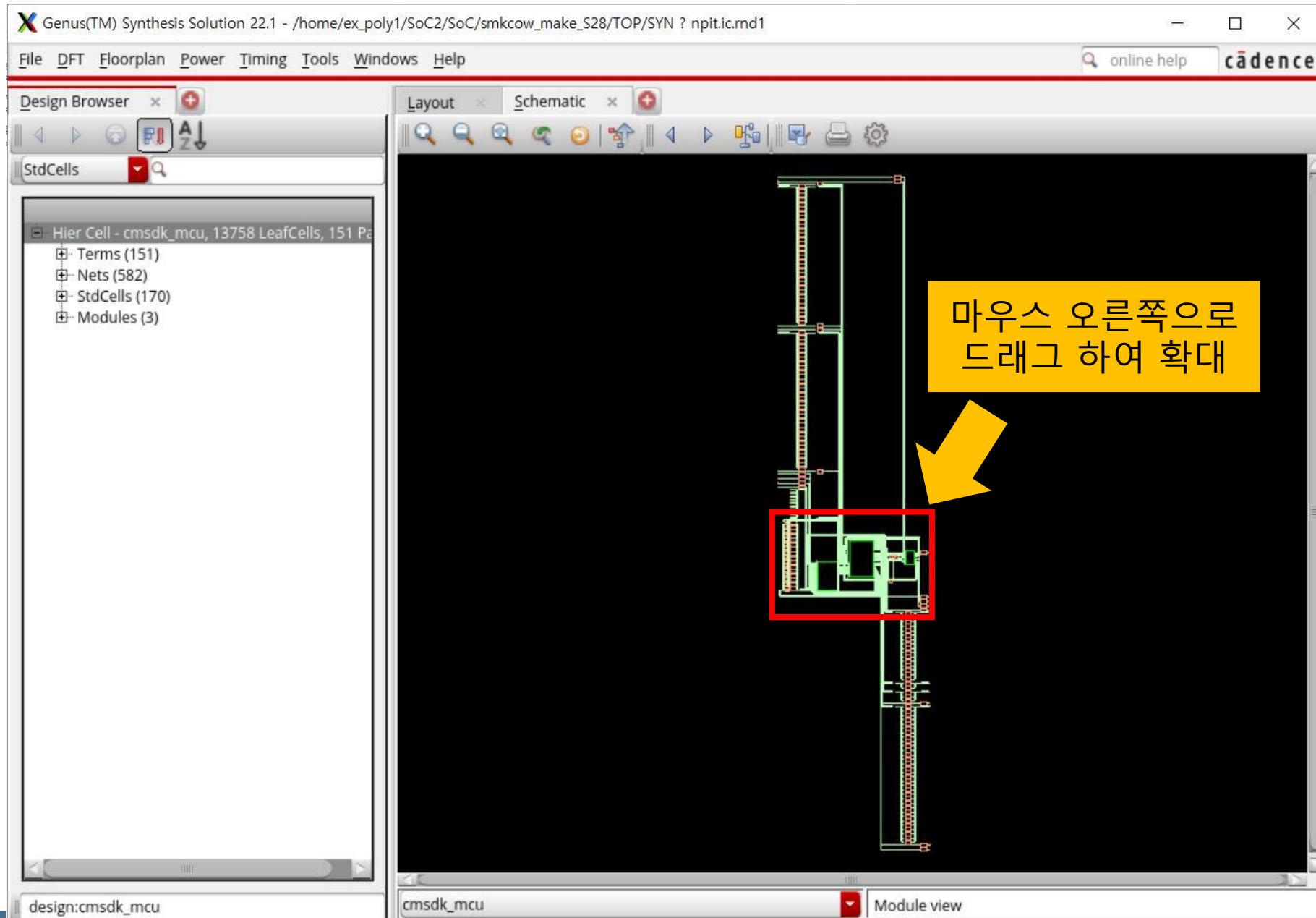
- Hier Cell 우 클릭
- schematic view 클릭
- In Main 클릭



# Synthesis

genus

\$> **gui\_show**

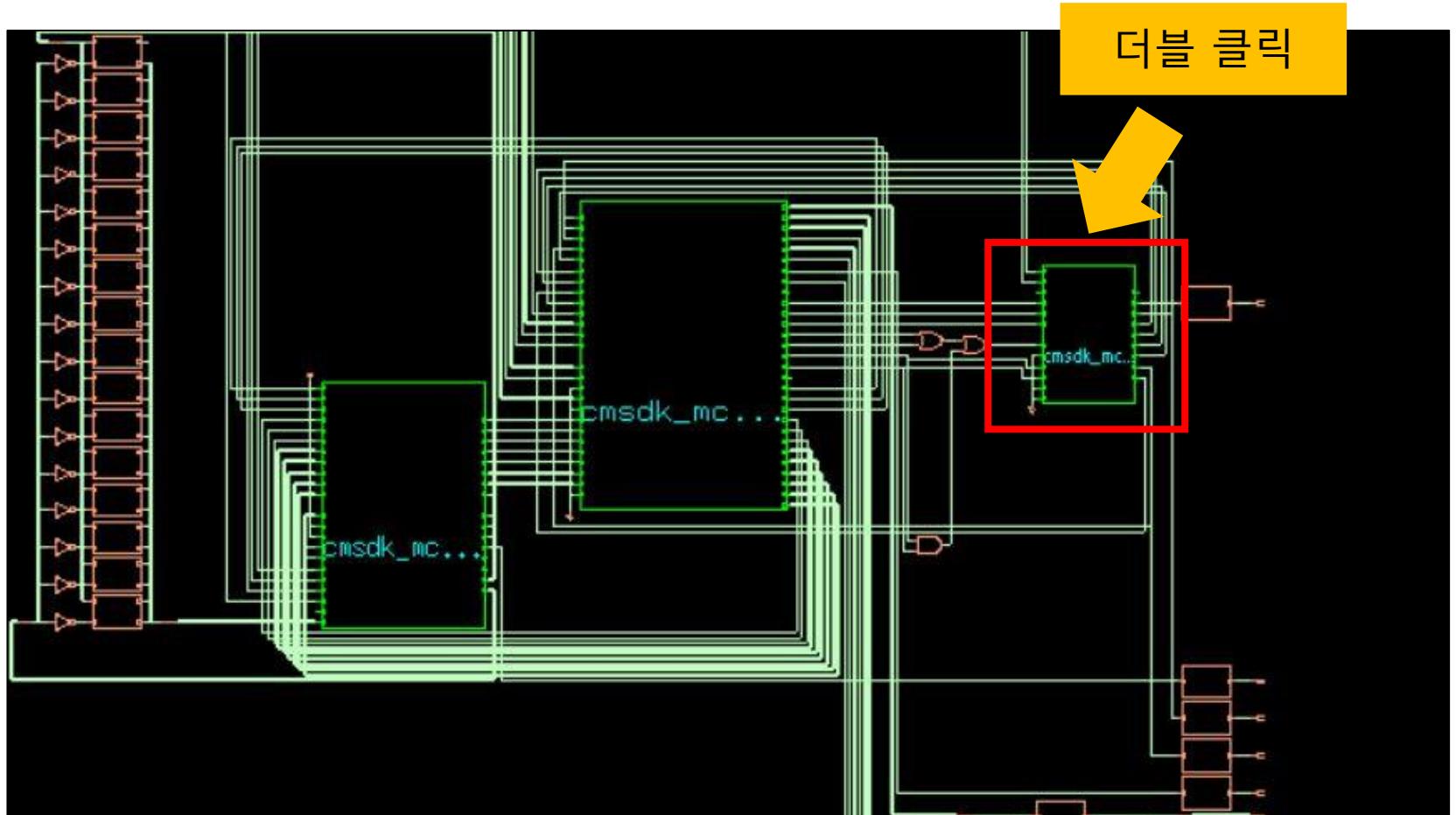


# Synthesis

genus

\$> gui\_show

- 합성 전 schematic 확인

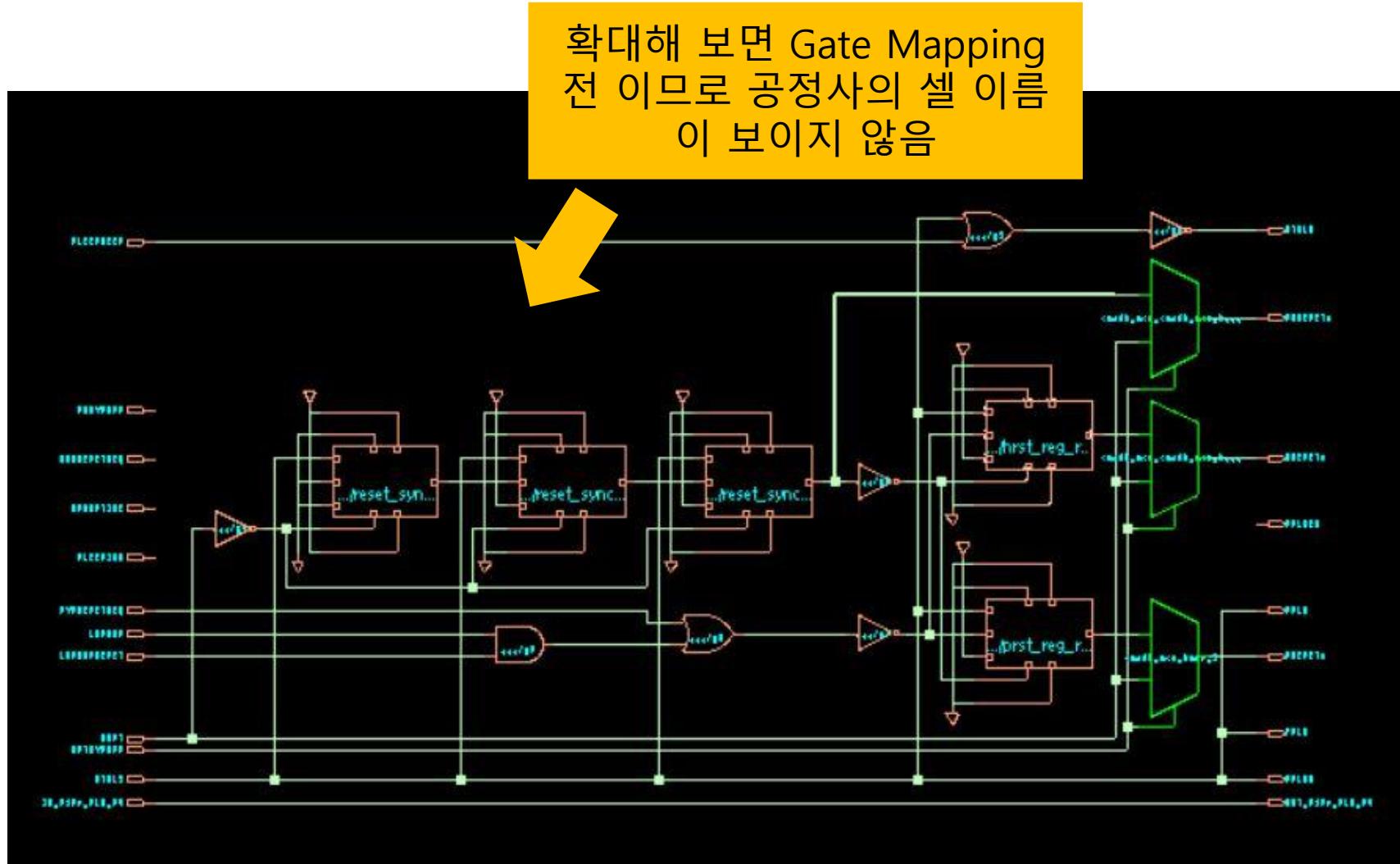


# Synthesis

genus

\$> **gui\_show**

- 합성 전 schematic 확인



# Synthesis

genus

## ◆ script/cortexm0\_45nm.tcl

Setup이 완료되었으므로 83행 부터  
차례차례 진행할 예정

```
78 #####
79 ## Constraints Setup
80 #####
81
82 #read_sdc ./cons/cmsdk_mcu.sdc
83 source -verbose ./cons/cmsdk_mcu.sdc ; #This command can't write report by using the mark '>'
84 source -verbose ./cons/dont_use_45nm.tcl ; #This command can't write report by using the mark '>'
85 puts "The number of exceptions is [llength [vfind "design:$TOP_DESIGN" -exception *]]"
86 #set_db "design:$DESIGN" .force_wireload <wireload name>
87
88 check_timing_intent -verbose > ${RPT_DIR}/01_check_timing.rpt
89
90 report_port -delay -driver [all_inputs] > ${RPT_DIR}/02_report_port.rpt
91 report_port -delay -load [all_outputs] >> ${RPT_DIR}/02_report_port.rpt
92
93 echo -n "\n TIMEX > compile is ready : ";
94 sh date
```

# Synthesis

## Genus 툴 실행 결과

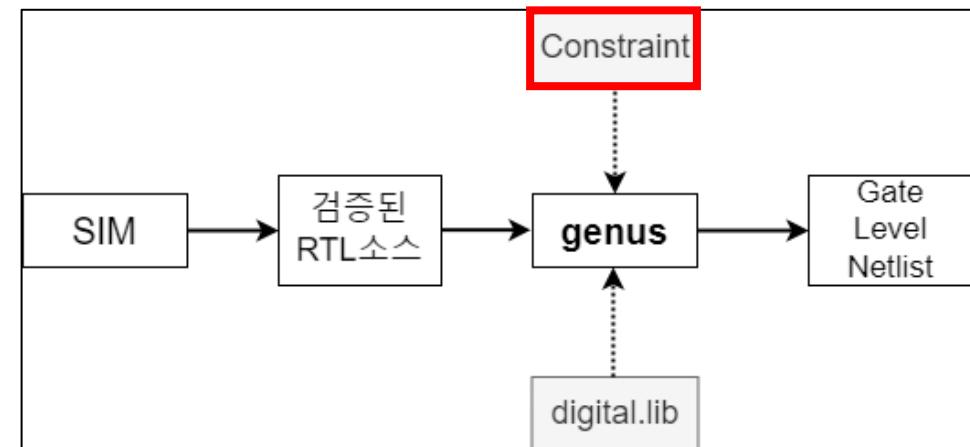
### genus

- cons의 .sdc 파일을 읽어 들임
- 검사 결과 Warning 발생

```
78 #####
79 ## Constraints Setup
80 #####
81
82 #read sdc ./cons/cmsdk_mcu.sdc
83 source -verbose ./cons/cmsdk_mcu.sdc ; #This command can
   e report by using the mark '>'
84 source -verbose ./cons/dont_use_45nm.tcl ; #This command can
   e report by using the mark '>'

85 puts "The number of exceptions is [llength [vfind "design:$DESIGN"
      N" -exception *]]"
86 #set_db "design:$DESIGN" .force_wireload <wireload name>
87
88 check_timing_intent -verbose > ${RPT_DIR}/01_check_timing.rpt
89
90 report_port -delay -driver [all_inputs] > ${RPT_DIR}/02_report_port.
rpt
91 report_port -delay -load [all_outputs] >> ${RPT_DIR}/02_report_port.
rpt
92
93 echo -n "\n TIMEX > compile is ready : "
94 sh date
```

```
@genus:root: 52> source -verbose ./cons/cmsdk_mcu.sdc ; #This command can't write report by using the mark '>'  
Sourcing './cons/cmsdk_mcu.sdc' (Sun Jan 26 21:55:43 KST 2025)...  
#@ Begin verbose source ./cons/cmsdk_mcu.sdc  
@file/cmsdk_mcu.sdc 13: set STD_LIB slow_vdd1v0_basicCells.lib:slow_vdd1v0  
@file/cmsdk_mcu.sdc 15: set_units -capacitance 1000.0fF  
@file/cmsdk_mcu.sdc 16: set_units -time 1000.0ps  
@file/cmsdk_mcu.sdc 19: reset_design  
@file/cmsdk_mcu.sdc 21: create_clock -period 7 -name MAIN_CLOCK [get_ports XTAL1]  
@file/cmsdk_mcu.sdc 24: set_clock_latency -source -max 0.2 [get_clocks MAIN_CLOCK]  
@file/cmsdk_mcu.sdc 25: set_clock_latency -max 0.2 [get_clocks MAIN_CLOCK]  
@file/cmsdk_mcu.sdc 31: set_clock_uncertainty -setup 0.1 [get_clocks MAIN_CLOCK]  
@file/cmsdk_mcu.sdc 34: set_clock_transition 0.1 [get_clocks MAIN_CLOCK]  
@file/cmsdk_mcu.sdc 50: set_input_delay -max 0.1 -clock MAIN_CLOCK [all_inputs]  
Warning : Unsupported SDC command option. [SDC-201] [set_input_delay]  
          : The set_input_delay command is not supported on ports which have a clock already defined 'port:cmsdk_mcu/XTAL1'.  
Warning : Unsupported SDC command option. [SDC-201] [set_input_delay]  
          : The set_input_delay command is not supported on ports which have a clock already defined 'port:cmsdk_mcu/XTAL1'.  
          : The current version does not support this SDC command option. However, future versions may be enhanced to support this option.  
@file/cmsdk_mcu.sdc 51: set_input_delay -min 0.01 -clock MAIN_CLOCK [all_inputs]  
Warning : Unsupported SDC command option. [SDC-201] [set_input_delay]  
          : The set_input_delay command is not supported on ports which have a clock already defined 'port:cmsdk_mcu/XTAL1'.
```



# Synthesis

genus

Clock을  
제외시킴



- set\_input\_delay는 Clock에 적용할 수 없다는 Warning이 발생 (50, 51행)
- 이후에는 set\_input\_delay에 Clock을 제외시키는 명령이 들어가므로 이 Warning 무시

```
47 #####  
48 ##### input, output delay ####  
49 #####  
50 set_input_delay -max 0.1 -clock MAIN_CLOCK [all_inputs]  
51 set_input_delay -min 0.01 -clock MAIN_CLOCK [all_inputs]  
52  
53 remove_input_delay [get_ports "XTAL1"]  
54  
55 set_output_delay -max 0.1 -clock MAIN_CLOCK [all_outputs]  
56 set_output_delay -min -0.01 -clock MAIN_CLOCK [all_outputs]  
57  
58 #####  
59 # DESIGN AREA #  
60 #####  
61 # Below is no use in Genus tool  
62 #set_max_area 0
```

```
@file(cmsdk_mcu.sdc) 50: set_input_delay -max 0.1 -clock MAIN_CLOCK [all_inputs]  
Warning : Unsupported SDC command option. [SDC-201] [set_input_delay]  
          : The set_input_delay command is not supported on ports which have a clock already defined 'p  
ort:cmsdk_mcu/XTAL1'.  
          : The current version does not support this SDC command option. However, future versions may  
be enhanced to support this option.  
@file(cmsdk_mcu.sdc) 51: set_input_delay -min 0.01 -clock MAIN_CLOCK [all_inputs]  
Warning : Unsupported SDC command option. [SDC-201] [set_input_delay]  
          : The set_input_delay command is not supported on ports which have a clock already defined 'p  
ort:cmsdk_mcu/XTAL1'.
```

50, 51행에 대한 Warning 내용

# Synthesis

genus

```
78 #####
79 ## Constraints Setup
80 #####
81
82 #read_sdc ./cons/cmsdk_mcu.sdc
83 source -verbose ./cons/cmsdk_mcu.sdc ; #This command can't write report by using the mark '>'
84 source -verbose ./cons/dont_use_45nm.tcl ; #This command can't write report by using the mark '>'
85 puts "The number of exceptions is [llength [vfind "design:$TOPDESIGN -exception *]]"
86 #set_db "design:$DESIGN" .force_wireload <wireload name>
87
88 check_timing_intent -verbose > ${RPT_DIR}/01_check_timing.rpt
89
90 report_port -delay -driver [all_inputs] > ${RPT_DIR}/02_report_port.rpt
91 report_port -delay -load [all_outputs] >> ${RPT_DIR}/02_report_port.rpt
92
93 echo -n "\n TIMEX > compile is ready : ";
94 sh date
```

## Genus 툴 실행 결과

```
@genus:root: 54> source -verbose ./cons/dont_use_45nm.tcl ; #This command can't write report by using the mark '>'  
Sourcing './cons/dont_use_45nm.tcl' (Sun Jan 26 22:03:58 KST 2025)...  
#@ Begin verbose source ./cons/dont_use_45nm.tcl  
@file(dont_use_45nm.tcl) 1: set_dont_use [get_lib_cells */ANTENNA*]  
@file(dont_use_45nm.tcl) 2: set_dont_use [get_lib_cells */DLY*]  
@file(dont_use_45nm.tcl) 4: set_dont_use [get_lib_cells */HOLD*]  
@file(dont_use_45nm.tcl) 5: set_dont_use [get_lib_cells */TIE*]  
@file(dont_use_45nm.tcl) 6: set_dont_use [get_lib_cells */TLA*]  
@file(dont_use_45nm.tcl) 7: set_dont_use [get_lib_cells */DFFS*]  
@file(dont_use_45nm.tcl) 8: set_dont_use [get_lib_cells */SDFF*]  
#@ End verbose source ./cons/dont_use_45nm.tcl
```

- cons의 .tcl 파일을 읽어 들임
- 사용하지 않는 셀들을 설정함

# Synthesis

## Genus 툴 실행 결과

genus

```
@genus:root: 55> puts "The number of exceptions is [llength [vfind "design:$TOP DESIGN" -exception *]]"  
The number of exceptions is 1
```

```
78 #####  
79 ## Constraints Setup  
80 #####  
81 #read_sdc ./cons/cmsdk_mcu.sdc  
82 source -verbose ./cons/cmsdk_mcu.sdc ; #This command can't write report by using the mark '>'  
83 source -verbose ./cons/dont_use_45nm.tcl ; #This command can't write report by using the mark '>'  
84 puts "The number of exceptions is [llength [vfind "design:$TOP DESIGN" -exception *]]"  
85 #set_db "design:$DESIGN" .force_wireload <wireload name>  
86 check_timing_intent -verbose > ${RPT_DIR}/01_check_timing.rpt  
87 report_port -delay -driver [all_inputs] > ${RPT_DIR}/02_report_port.rpt  
88 report_port -delay -load [all_outputs] >> ${RPT_DIR}/02_report_port.rpt  
89  
90 echo -n "\n TIMEX > compile is ready : ";  
91 sh date
```

- puts는 메시지를 출력하는 명령임

# Synthesis

genus

## Genus 툴 실행 결과

Lint summary

Unconnected/logic driven clocks	0
Sequential data pins driven by a clock signal	0
Sequential clock pins without clock waveform	0
Sequential clock pins with multiple clock waveforms	0
Generated clocks without clock waveform	0
Generated clocks with incompatible options	0
Generated clocks with multi-master clock	0
Generated clocks with master clock not reaching the generated clock target	0
Paths constrained with different clocks	0
Loop-breaking cells for combinational feedback	0
Nets with multiple drivers	0
Timing exceptions with no effect	0
Suspicious multi_cycle exceptions	0
Pins/ports with conflicting case constants	0
Inputs without clocked external delays	0
Outputs without clocked external delays	0
Inputs without external driver/transition	0
Outputs without external load	0
Exceptions with invalid timing start-/endpoints	0

Total: 0

```
78 #####
79 ## Constraints Setup
80 #####
81
82 #read_sdc ./cons/cmsdk_mcu.sdc
83 source -verbose ./cons/cmsdk_mcu.sdc      ;
e report by using the mark '>'
84 source -verbose ./cons/dont_use_45nm.tcl ;
e report by using the mark '>'
85 puts "The number of exceptions is [llength [vfind "design:$TOP_DESIGN
N" -exception *]]"
86 #set_db "design:$DESIGN" .force_wireload <wireload name>
87
88 check_timing_intent -verbose > ${RPT_DIR}/01_check_timing.rpt
89
90 report_port -delay -driver [all_inputs] > ${RPT_DIR}/02_report_port.
rpt
91 report_port -delay -load [all_outputs] >> ${RPT_DIR}/02_report_port.
rpt
92
93 echo -n "\n TIMEX > compile is ready :           ";
94 sh date
```

- 내가 입력한 constraint에 문제가 없음을 확인

# Synthesis

genus

## Genus 툴 실행 결과

```
@genus:root: 58> report_port -delay -driver [all_inputs] > ${RPT_DIR}/02_report_port.rpt
@genus:root: 59> report_port -delay -load [all_outputs] >> ${RPT_DIR}/02_report_port.rpt
@genus:root: 60> echo -n "\n TIMEX > compile is ready : ";
TIMEX > compile is ready :
Sun Jan 26 22:08:49 KST 2025
@genus:root: 61> sh date
```

```
78 #####
79 ## Constraints Setup
80 #####
81
82 #read_sdc ./cons/cmsdk_mcu.sdc
83 source -verbose ./cons/cmsdk_mcu.sdc      ; #This command can't write report by using the mark '>'
84 source -verbose ./cons/dont_use_45nm.tcl ; #This command can't write report by using the
85 puts "The number of exception is $TOP_DESIGN_N" -exception *]]"
86 #set_db "design:$DESIGN_NAME"
87
88 check_timing_intent -verbose > ${RPT_DIR}/01_check_timing.rpt
89
90 report_port -delay -driver [all_inputs] > ${RPT_DIR}/02_report_port.rpt
91 report_port -delay -load [all_outputs] >> ${RPT_DIR}/02_report_port.rpt
92
93 echo -n "\n TIMEX > compile is ready : ";
94 sh date
```

>: 새로운 레포트를 생성함  
>>: 기존 레포트에 이어서 씀



- 입력 포트와 출력 포트의 지연 값을 02\_report\_port에 입력함

# Synthesis

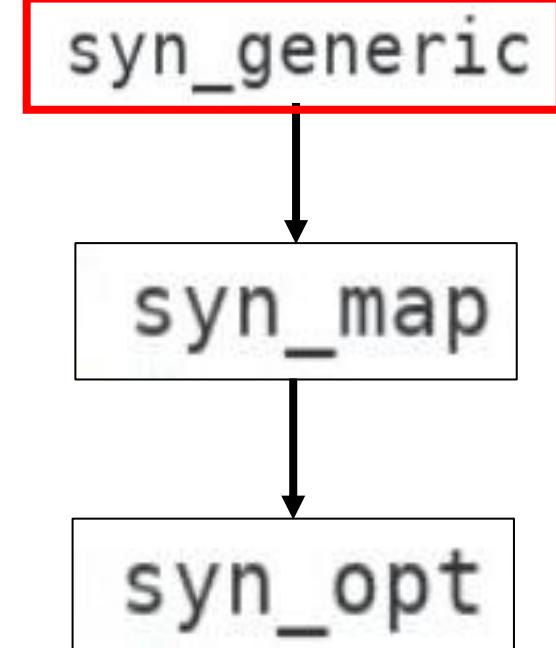
## Genus 툴 실행 결과

genus

```
@genus:root: 62> write_db -to_file ${UNMAPPED_DIR}/syn_generic.db  
Finished exporting design database to file './unmapped/syn_generic.db' for 'cmsdk_mc  
u' (command execution time mm:ss cpu = 00:01, real = 00:02).
```

- 합성의 세가지 단계 중 첫번째 단계임
- unmapped에 report 생성

```
96 #####  
97 ## Synthesizing to generic  
98 #####  
99  
100 syn_generic ; #This command can't write report by using the mark '>'  
101 #syn_generic -physical  
102 puts "Runtime & Memory after 'syn_generic'"  
103 time_info GENERIC  
104 report_datapath > ${RPT_DIR}/03_report_datapath_generic.rpt  
105  
106 write_snapshot -outdir ${RPT_DIR} -tag syn_generic  
107 # Below 2 lines are the same with the above command :: write_snapshot  
108 #report_summary -directory ${REPORTS_PATH}  
109 #summary_table -directory ${RPT_DIR}  
110  
111 write_db -to_file ${UNMAPPED_DIR}/syn_generic.db
```

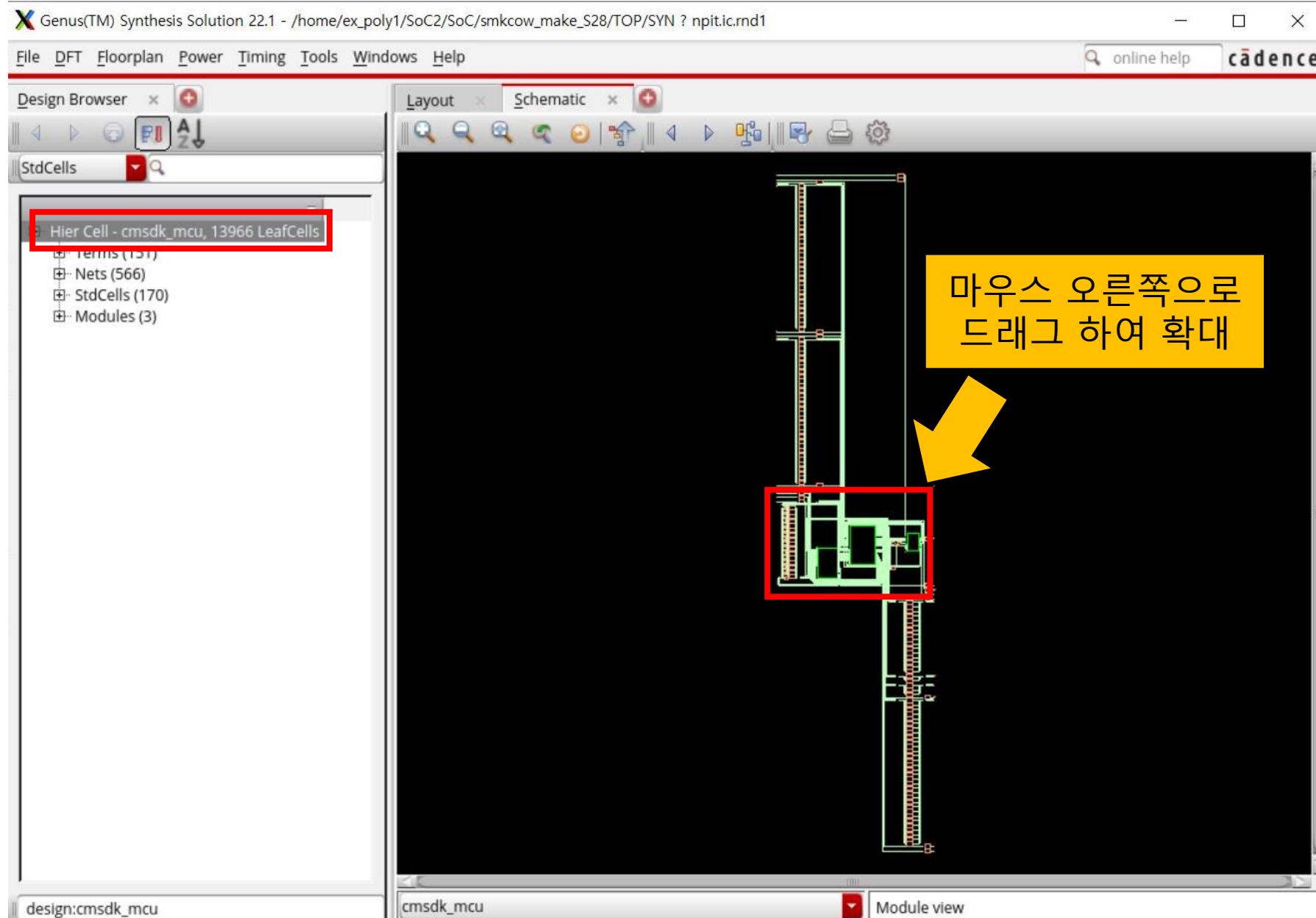


# Synthesis

genus

\$> gui\_show

- syn\_generic 실행 후  
TOP모듈 확인

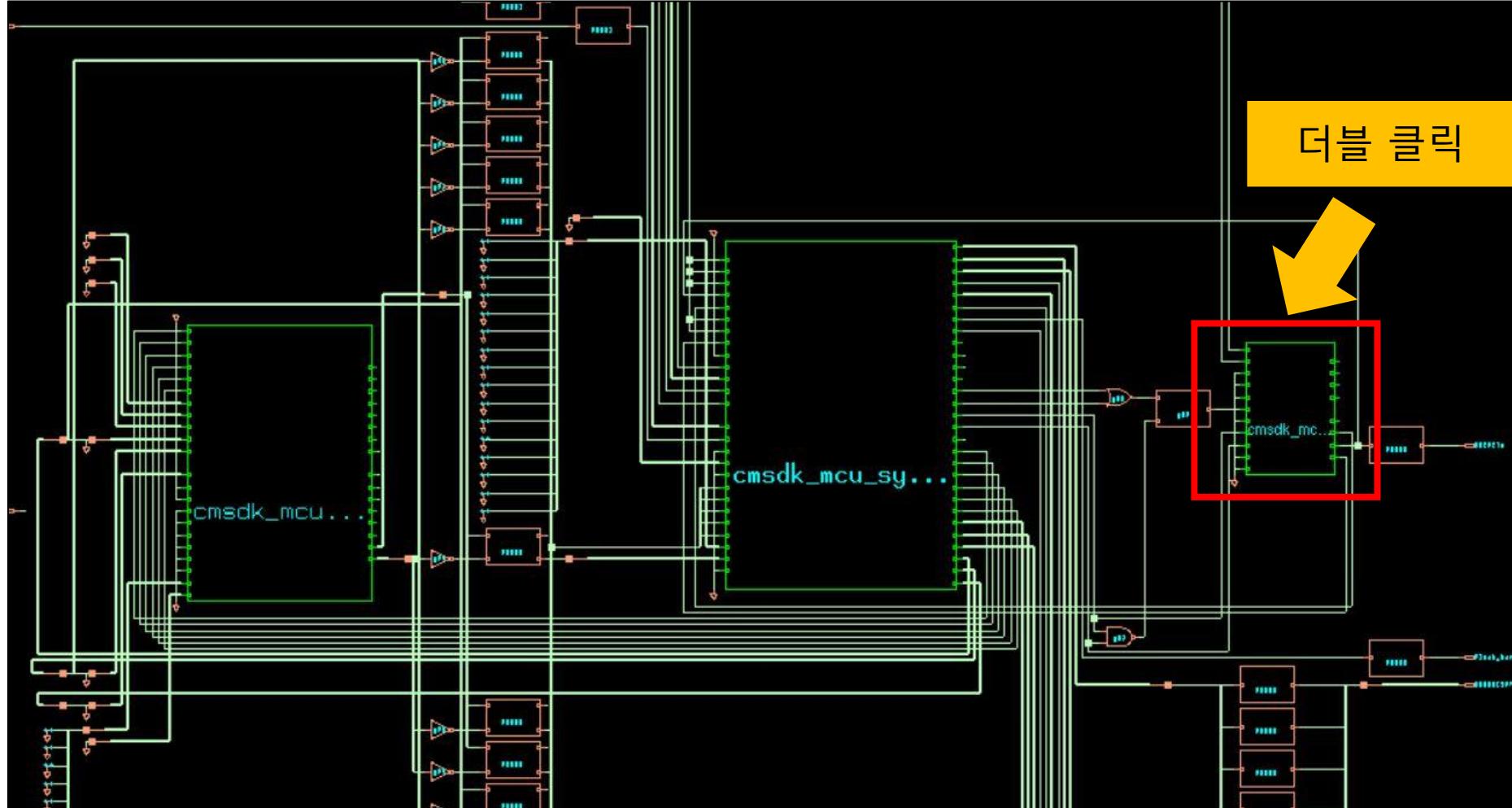


# Synthesis

genus

\$> gui\_show

- syn\_generic 실행 후 schematic 확인

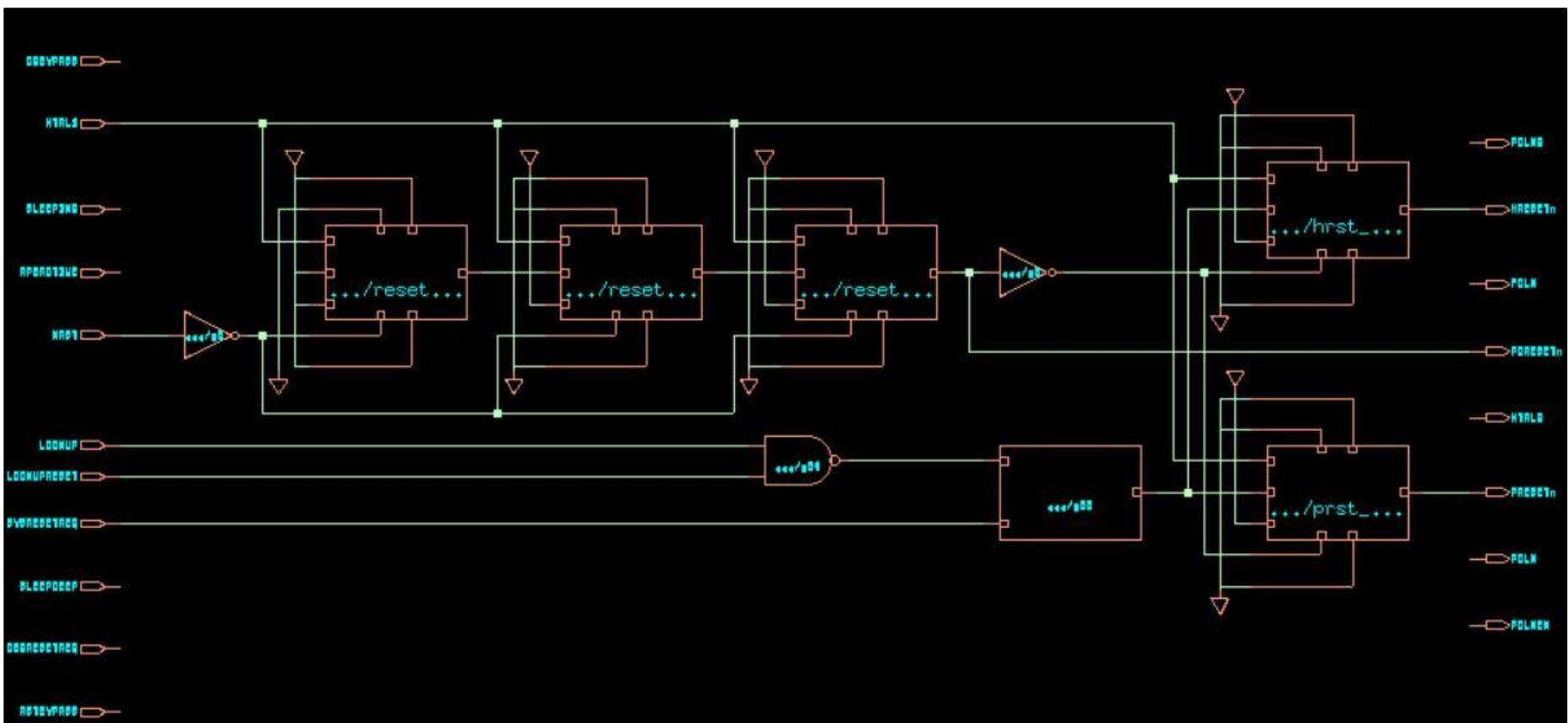


# Synthesis

genus

\$> gui\_show

- syn\_generic 실행 후 schematic 확인



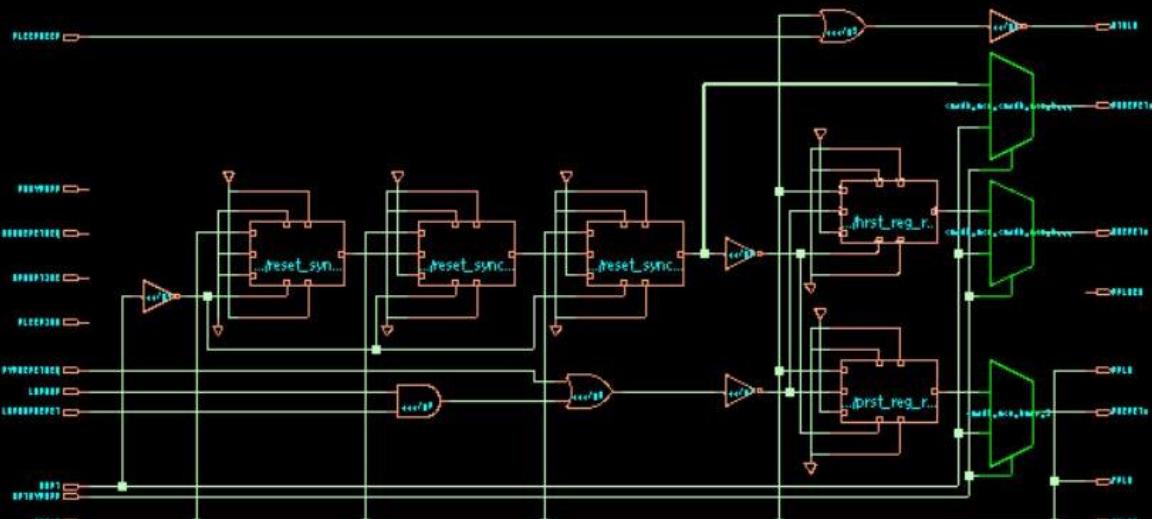
# Synthesis

genus

- 합성 전 후 비교

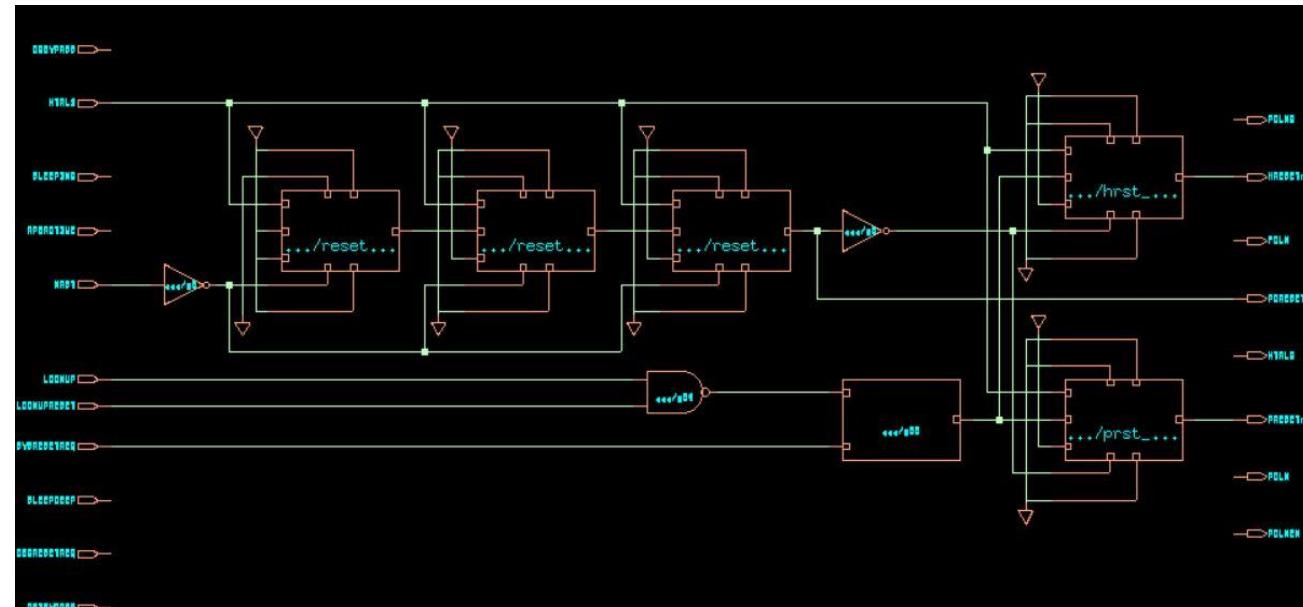
합성 전

(syn\_generic 실행 전)



합성 후

(syn\_generic 실행 후)

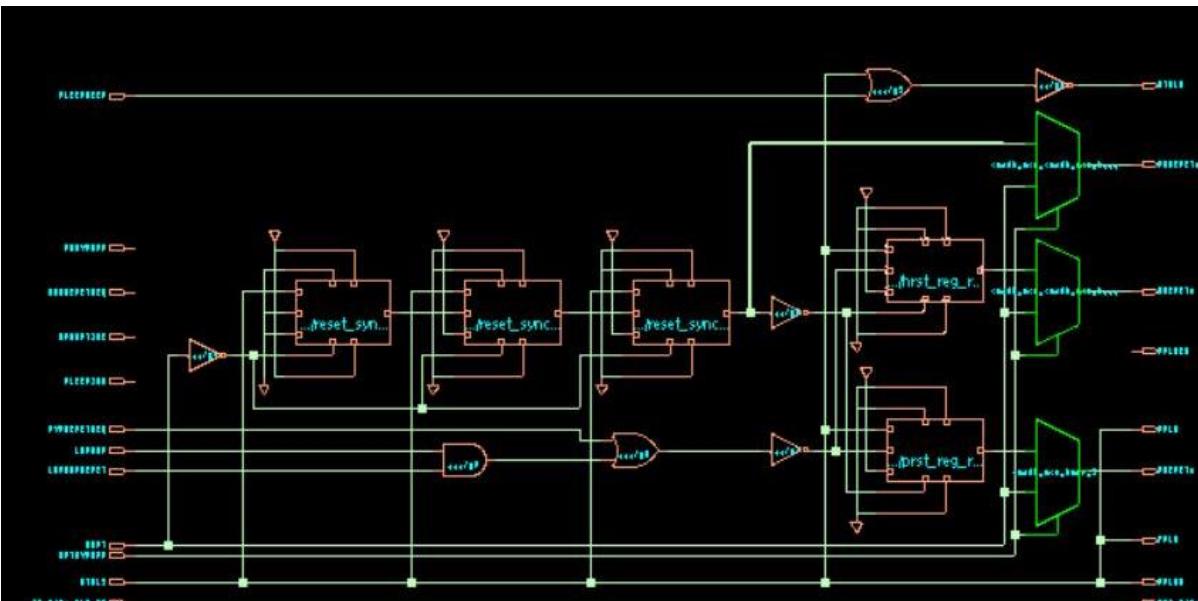


# Synthesis

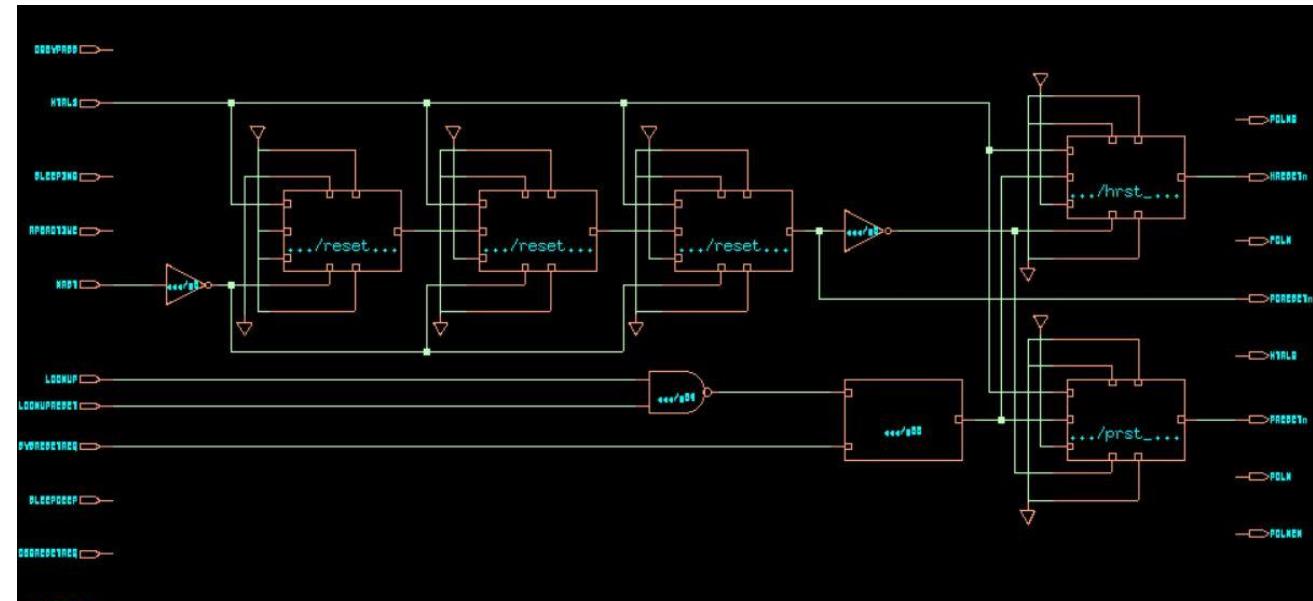
## genus

- 합성 전 후 비교
- 셀 이름은 동일함

합성 전  
(syn\_generic 실행 전)



합성 후  
(syn\_generic 실행 후)



# Synthesis

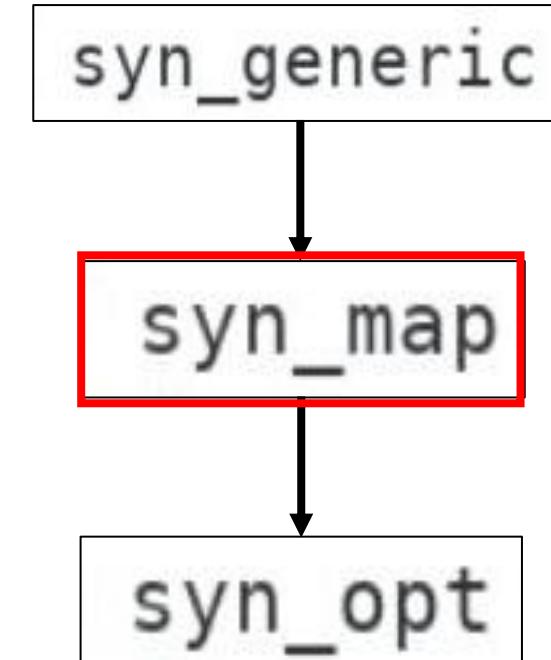
genus

Genus 툴 실행 결과

```
@genus:root: 73> write_db -to_file ${MAPPED_DIR}/syn_map.db
Finished exporting design database to file './mapped/syn_map.db' for 'cmsdk_mcu' (co
mmand execution time mm:ss cpu = 00:01, real = 00:01).
```

- 합성의 세가지 단계 중 두번째 단계임
- 공정사의 STD셀과 연결하는 Mapping 단계임

```
113 #####
114 ## Synthesizing to gates
115 #####
116
117 syn_map ; #This command can't write report by using the mark '>'
118 #syn_map -physical
119 puts "Runtime & Memory after 'syn_map'"
120 time_info MAPPED
121
122 report_datapath > ${RPT_DIR}/04_report_datapath_map.rpt
123
124 foreach cg [vfind / -cost_group *] {
125   report timing -cost_group [list $cg] > ${RPT_DIR}/05_[basename $cg
126 ]_post_syn_map.rpt
127 }
128
129 write_snapshot -outdir ${RPT_DIR} -tag syn_map
130 # Below line is the same with the above command :: write_snapshot
131 #summary_table -directory ${RPT_DIR}
132 write_db -to_file ${MAPPED_DIR}/syn_map.db
```

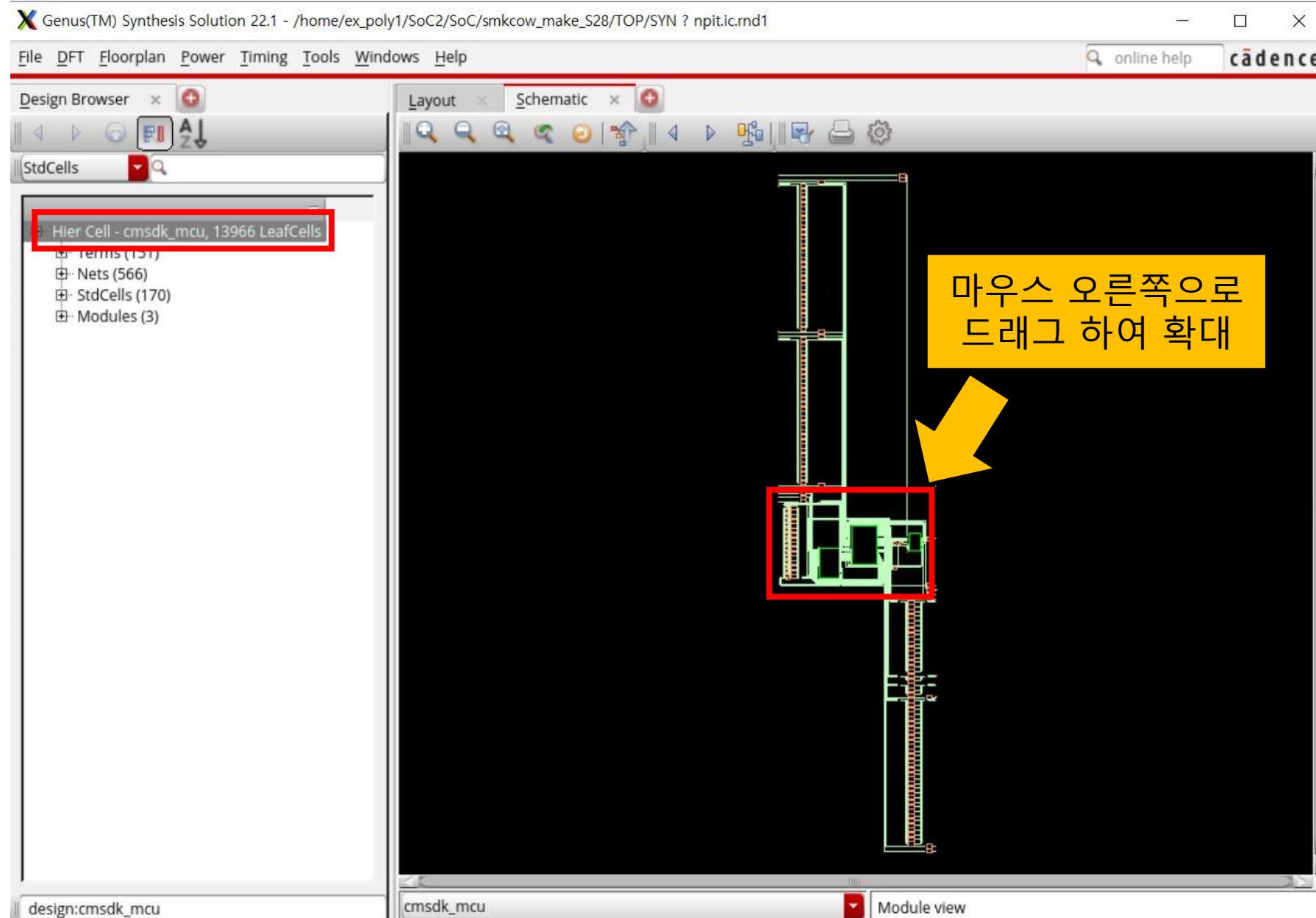


# Synthesis

genus

\$> gui\_show

- syn\_map 실행 후 TOP 모듈 확인

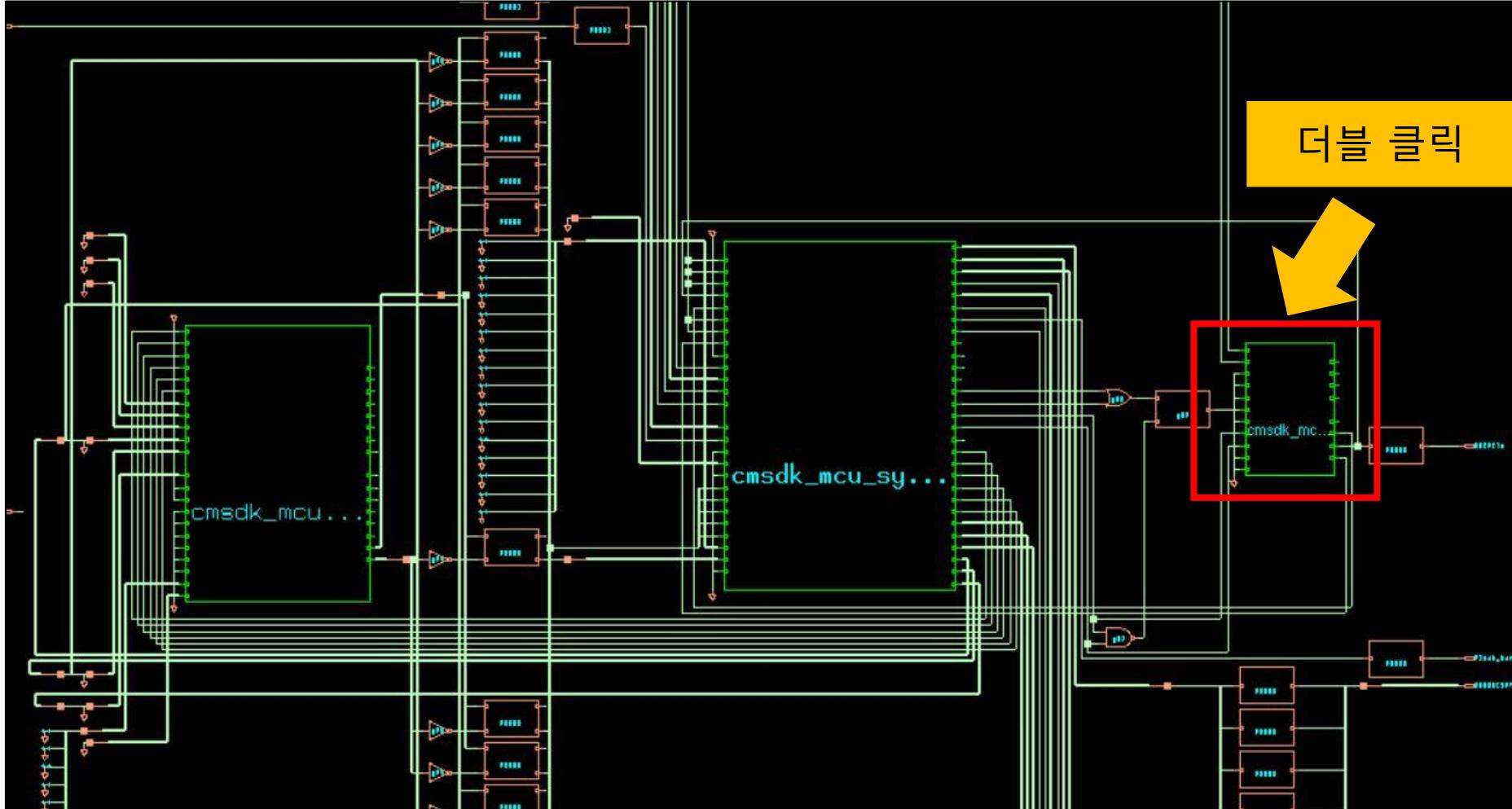


# Synthesis

genus

\$> gui\_show

- syn\_map 실행 후 schematic 확인

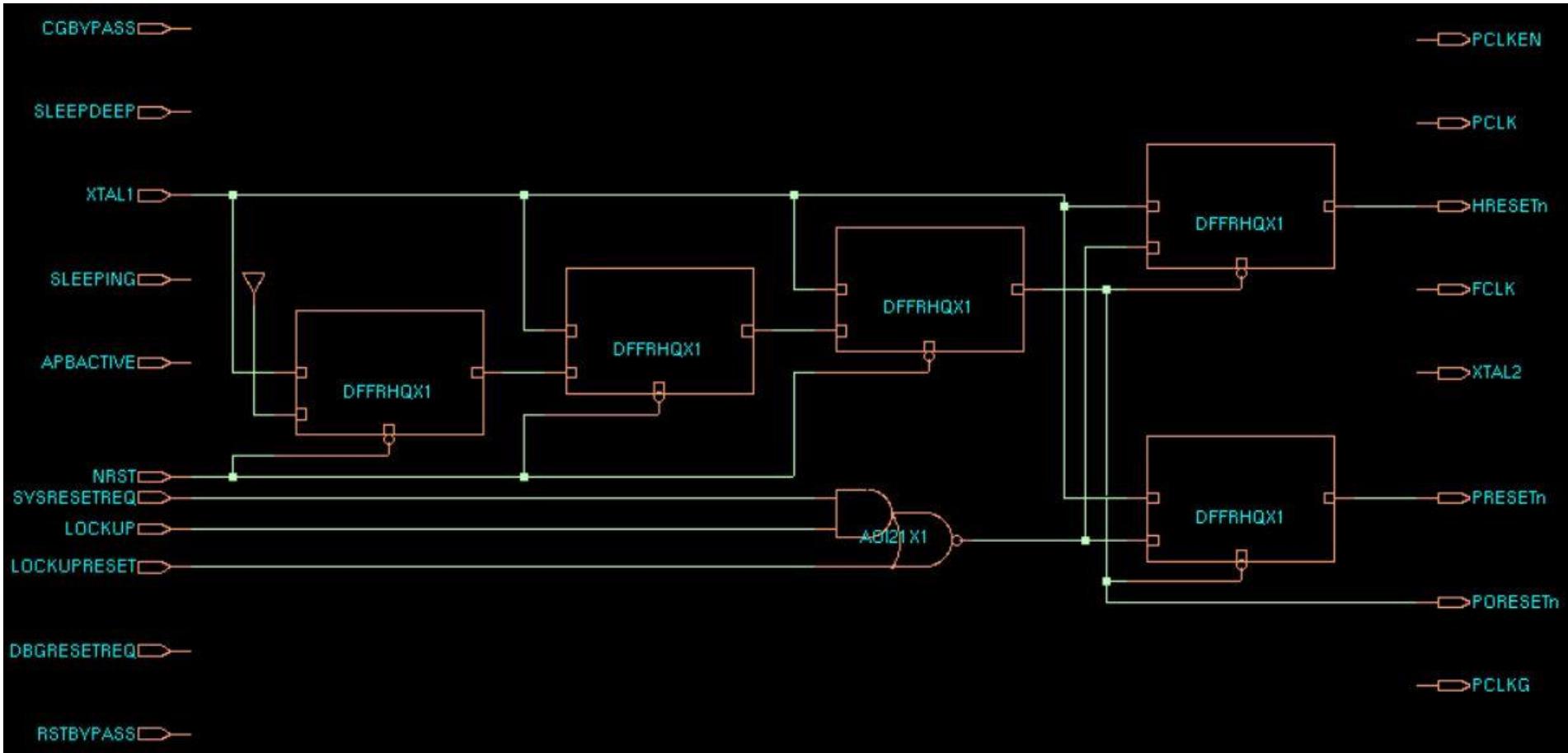


# Synthesis

genus

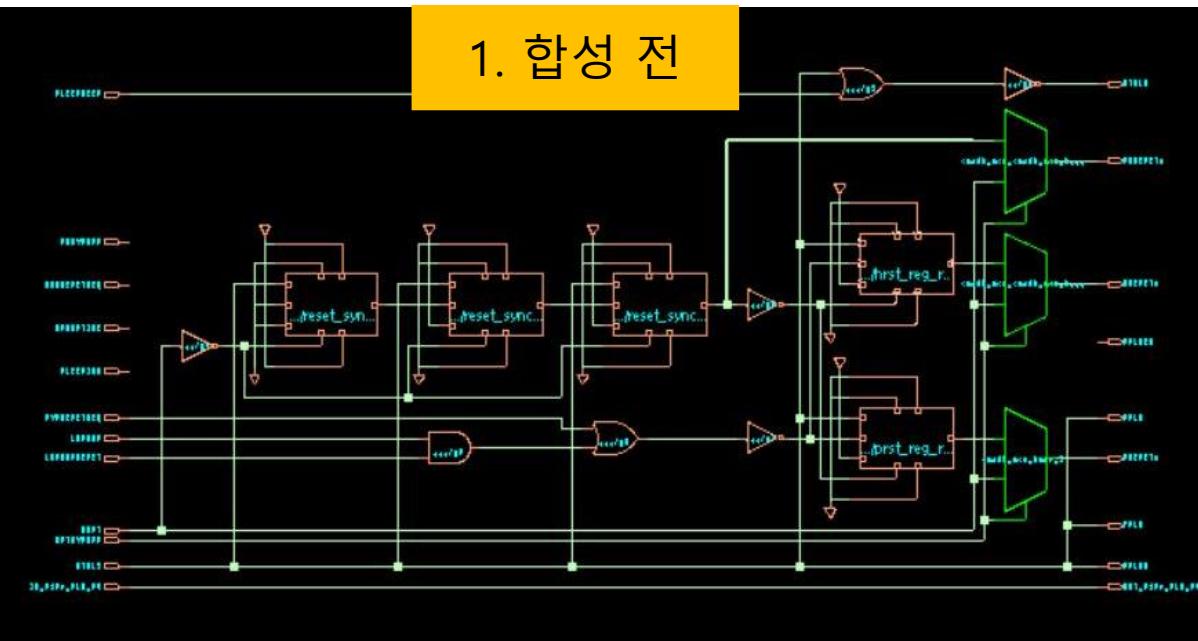
\$> gui\_show

- syn\_map 실행 후 schematic 확인
- 셀의 이름이 바뀌었고 잘 정리되어 있는 것을 볼 수 있음

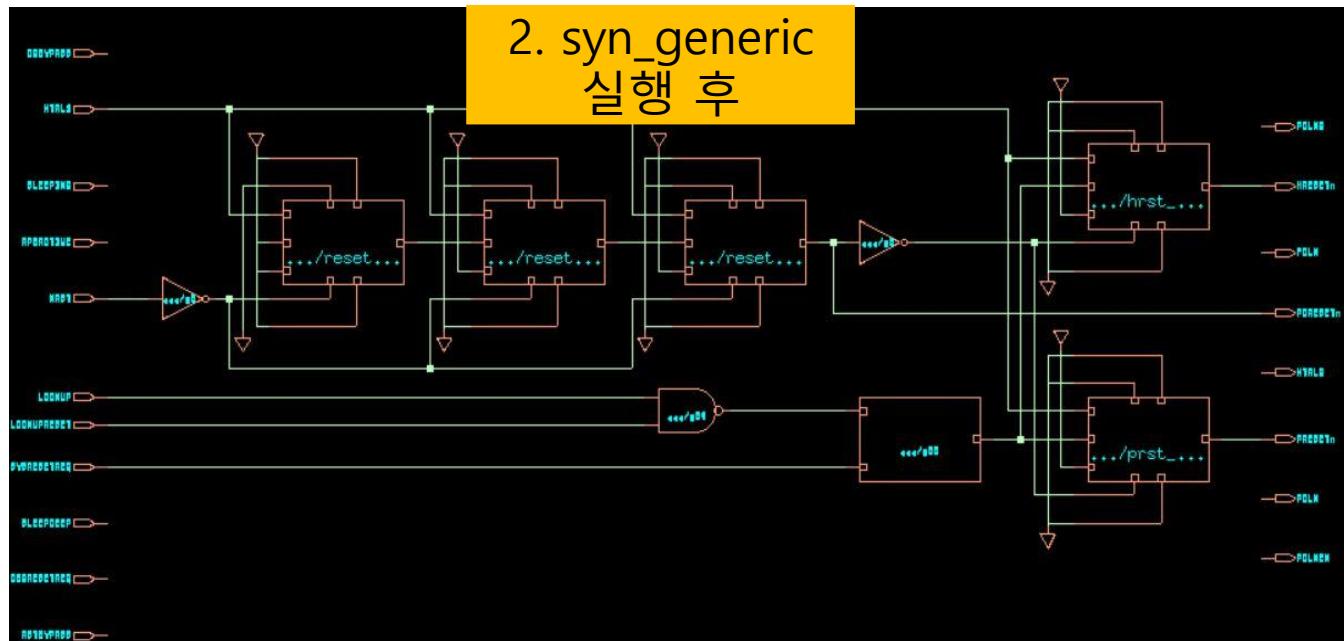


# Synthesis

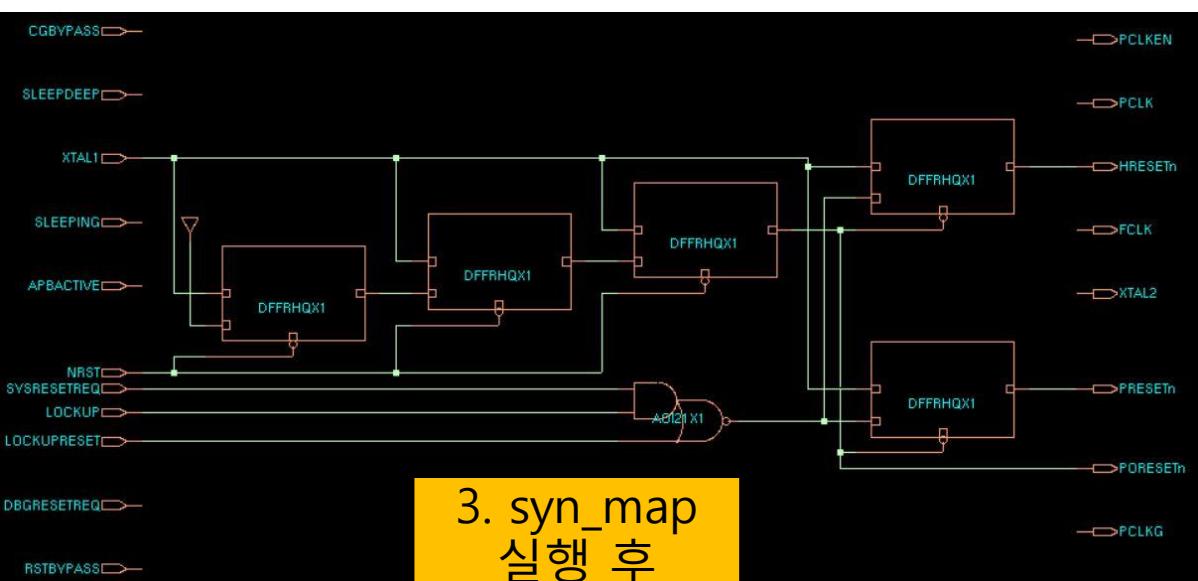
1. 합성 전



2. syn\_generic  
실행 후



3. syn\_map  
실행 후



# Synthesis

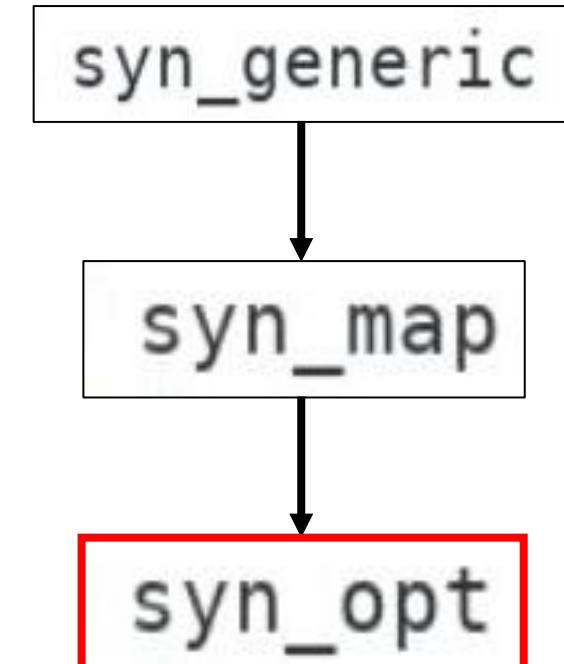
## Genus 툴 실행 결과

genus

```
@genus:root: 85> echo -n "\n TIMEX > Compile is completed : ";
TIMEX > Compile is completed :
Sun Jan 26 23:06:15 KST 2025
@genus:root: 86> sh date
```

- 합성의 세가지 단계 중 마지막 단계임
- Design을 최적화 하는 optimize 단계임

```
134 #####
135 ## Optimize Netlist + Incremental
136 #####
137
138 syn_opt ; #This command can't write report by using the mark '>'
139 #syn_opt -incremental
140 #syn_opt -physical
141
142 report_constraint -all_violators > ${RPT_DIR}/06_report_constraint.rpt
143
144 write_snapshot -outdir ${RPT_DIR} -tag syn_opt
145 # Below line is the same with the above command :: write_snapshot
146 #summary_table -directory ${RPT_DIR}
147
148 puts "Runtime & Memory after syn_opt"
149 time_info OPT
150
151 foreach cg [vfind / -cost_group *] {
152   report timing -cost_group [list $cg] > ${RPT_DIR}/07_[basename $cg]_post_syn_opt.rpt
153 }
154
155 write_db -to_file ${MAPPED_DIR}/syn_opt.db
156
157
158 echo -n "\n TIMEX > Compile is completed : ";
159 sh date
```



# Synthesis

## genus

- Constraint에 만족하는 디자인이 나왔는지 검증

### Genus 툴 실행 결과

```
134 ##### Incremental #####
13 report_constraint      Incremental
13                                     #####
13                                     실행
138 syn_opt ; "This command can't write report by using the mark
139 #syn_opt incremental
140 #syn_opt physical
141
142 report_constraint -all_violators > ${RPT_DIR}/06_report_cons
143
144 write_snapshot -outdir ${RPT_DIR} -tag syn_opt
145 # Below line is the same with the above command :: write_sna
146 #summary_table -directory ${RPT_DIR}
147
148 puts "Runtime & Memory after syn_opt"
149 time_info OPT
150
151 foreach cg [vfind / -cost_group *] {
152   report timing -cost_group [list $cg] > ${RPT_DIR}/07_[base
153 }
154
155 write_db -to_file ${MAPPED_DIR}/syn_opt.db
156
157
158 echo -n "\n TIMEX > Compile is completed :           ";
159 sh date
```

```
@genus:root: 8> report_constraint -all_violators
=====
Generated by:          Genus(TM) Synthesis Solution 22.16-s078_1
Generated on:          Feb 06 2025 11:18:16 pm
Module:                cmsdk_mcu
Operating conditions: PVT_0P9V_125C
Interconnect mode:    global
Area mode:             Setup Timing, clock_period,
                        pulse_width에 문제 없음
=====
=====
Checking for violation type : Setup Timing Slack
-----
No timing slack violation found.
-----
Checking for violation type : clock_period
-----
No paths found.
-----
Checking for violation type : pulse_width
-----
No paths found.
```

# Synthesis

genus

## Genus 툴 실행 결과

Checking for violation type : max\_capacitance

Pin	Required Load (ff)	Actual Load (ff)	Slack Load (ff)	Mode
port:cmsdk_mcu/P1[0]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[1]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[2]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[3]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[4]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[5]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[6]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[7]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[8]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[9]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[10]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[11]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[12]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[13]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[14]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/P1[15]	888.30	2388.40	-1500.10	default
port:cmsdk_mcu/flash_hrd[14]	888.30	2385.40	-1497.10	default
port:cmsdk_mcu/sram_hrd[13]	888.30	2385.40	-1497.10	default
port:cmsdk_mcu/sram_hrd[12]	888.30	2385.40	-1497.10	default
port:cmsdk_mcu/sram_hrd[11]	888.30	2385.40	-1497.10	default
port:cmsdk_mcu/sram_hrd[10]	888.30	2385.40	-1497.10	default
port:cmsdk_mcu/sram_hrd[9]	888.30	2385.40	-1497.10	default
port:cmsdk_mcu/sram_hrd[8]	888.30	2385.40	-1497.10	default
port:cmsdk_mcu/sram_hrd[7]	888.30	2385.40	-1497.10	default

- max\_capacitance에 Required Load보다 Actual Load가 크게 걸림
- Io셀을 이미 넣은 상황이기 때문에 **수정 불가**
- BE과정에서 개선의 여지가 있기 때문에 일단 유지

# Synthesis

## genus

### Genus 툴 실행 결과

Checking for violation type : max\_transition

Pin	Required Slew (ps)	Actual Slew (ps)	Slack Slew (ps)	Mode
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4850/B1	280	915	-635	default
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4851/B1	280	915	-635	default
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4852/B1	280	915	-635	default
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4853/B1	280	915	-635	default
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4854/B1	280	915	-635	default
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4855/B1	280	915	-635	default
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4856/B1	280	915	-635	default
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4857/B1	280	915	-635	default
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4858/B1	280	915	-635	default
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4859/B1	280	915	-635	default
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4860/B1	280	915	-635	default
pin:cmsdk_mcu/u_cmsdk_mcu_system/u_apb_subsystem/gen_apb_watchdog.u_apb_watchdog/u_apb_watchdog_frc/g 4861/B1	280	915	-635	default

No max\_fanout rule violations.

fanout 문제 없음

# Synthesis

## genus

- report\_timing 명령으로 합성 결과 중 WNS(제일 좋지 않은 Path) 결과 확인 가능
- Slack에 음수가 아닌 양수 값이 나옴으로 문제가 없음을 확인할 수 있음

Genus 툴 실행 결과

```
134 #####  
135 ## Optimize Netlist + Incremental  
136 #####  
137  
138 syn_opt ; #This command can't write repo  
139 #syn_opt -incremental  
140 #syn_opt -physical  
141  
142 report_constraint -all_violators > ${RPT_DIR}/constraint.rpt  
143  
144 write_snapshot -outdir ${RPT_DIR} -tag snapshot  
145 # same with the above  
146 report_timing  
147 실행  
148 puts "Report time & Memory after syn_opt"  
149 time_in OPT  
150  
151 foreach cg [vfind / -cost_group *] {  
152   report timing -cost_group [list $cg]  
153 }  
154  
155 write_db -to_file ${MAPPED_DIR}/syn_opt.db  
156  
157  
158 echo -n "\n TIMEX > Compile is completed  
159 sh date
```

Path 1: MET (1645 ps) Setup Check with Pin u\_cmsdk\_mcu\_system/u\_cortexm0integration/u\_cortexm0/u\_logic/0o3l85\_reg/CKN->D  
Group: MAIN\_CLOCK  
Startpoint: (R) u\_cmsdk\_mcu\_system/u\_cortexm0integration/u\_cortexm0/u\_logic/Xc8l85\_reg/CK  
Clock: (R) MAIN\_CLOCK  
Endpoint: (F) u\_cmsdk\_mcu\_system/u\_cortexm0integration/u\_cortexm0/u\_logic/0o3l85\_reg/D  
Clock: (R) MAIN\_CLOCK

Capture	Launch
Clock Edge:+ 7000	0
Src Latency:+ 200	200
Net Latency:+ 200 (I)	200 (I)
Arrival:= 7400	400

Setup:- 31  
Uncertainty:- 100  
Required Time:= 7269  
Launch Clock:- 400  
Data Path:- 5225  
Slack:= 1645

Timing Point

Arrival Instance	Flags	Arc	Edge	Cell	Fanout	Load	Trans	Delay
(ps) Location	(ff)	(ps)	(ps)					

u\_cmsdk\_mcu\_system/u\_cortexm0integration/u\_cortexm0/u\_logic/Xc8l85\_reg/CK  
400 (-,-)  
u\_cmsdk\_mcu\_system/u\_cortexm0integration/u\_cortexm0/u\_logic/Xc8l85\_reg/Q  
642 (-,-)  
u\_cmsdk\_mcu\_system/u\_cortexm0integration/u\_cortexm0/u\_logic/g215877/Y  
794 (-,-)

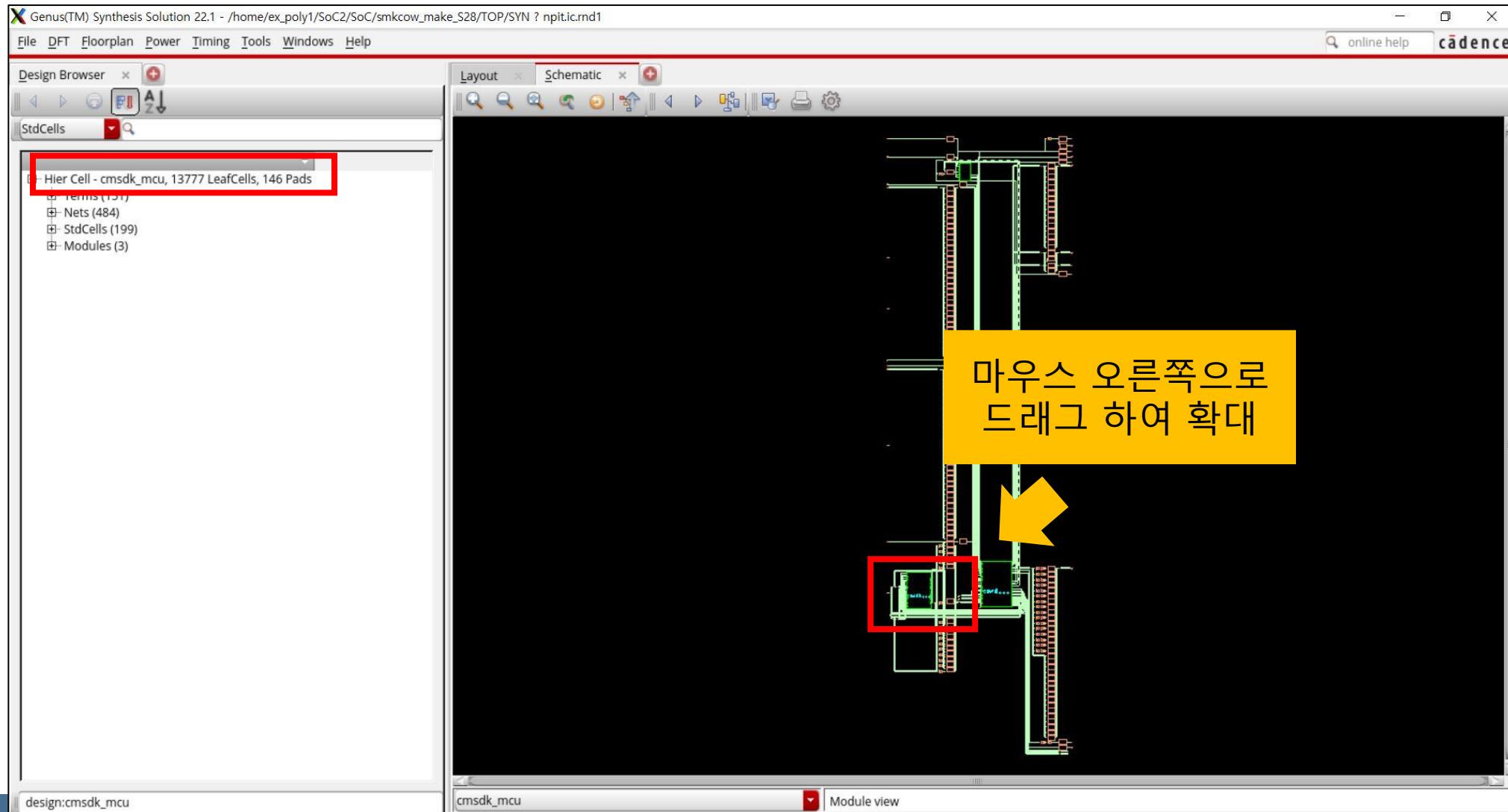
문제 없음을 확인

# Synthesis

\$> gui\_show

genus

- syn\_opt 실행 후 TOP모듈 확인

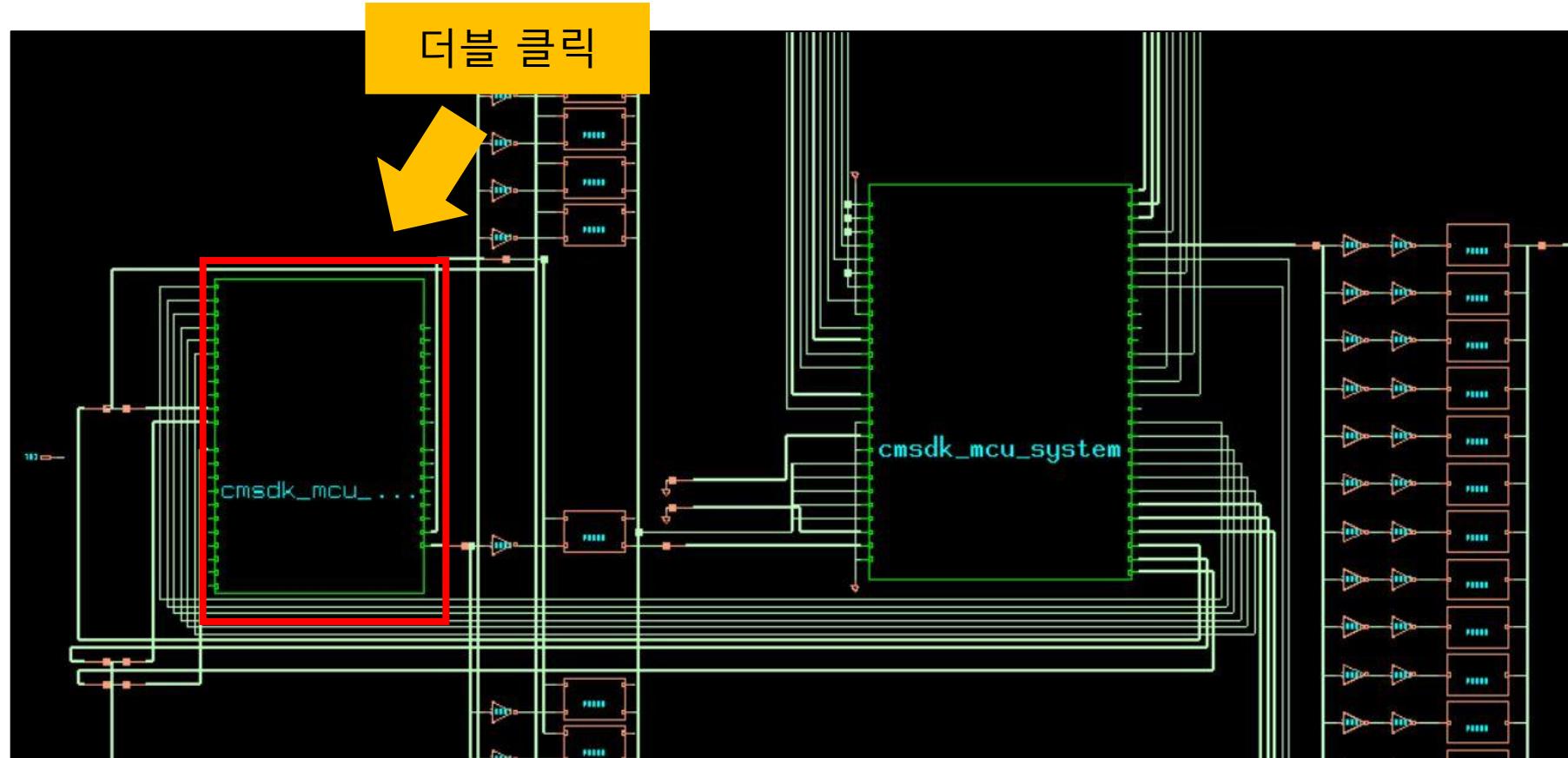


# Synthesis

genus

\$> gui\_show

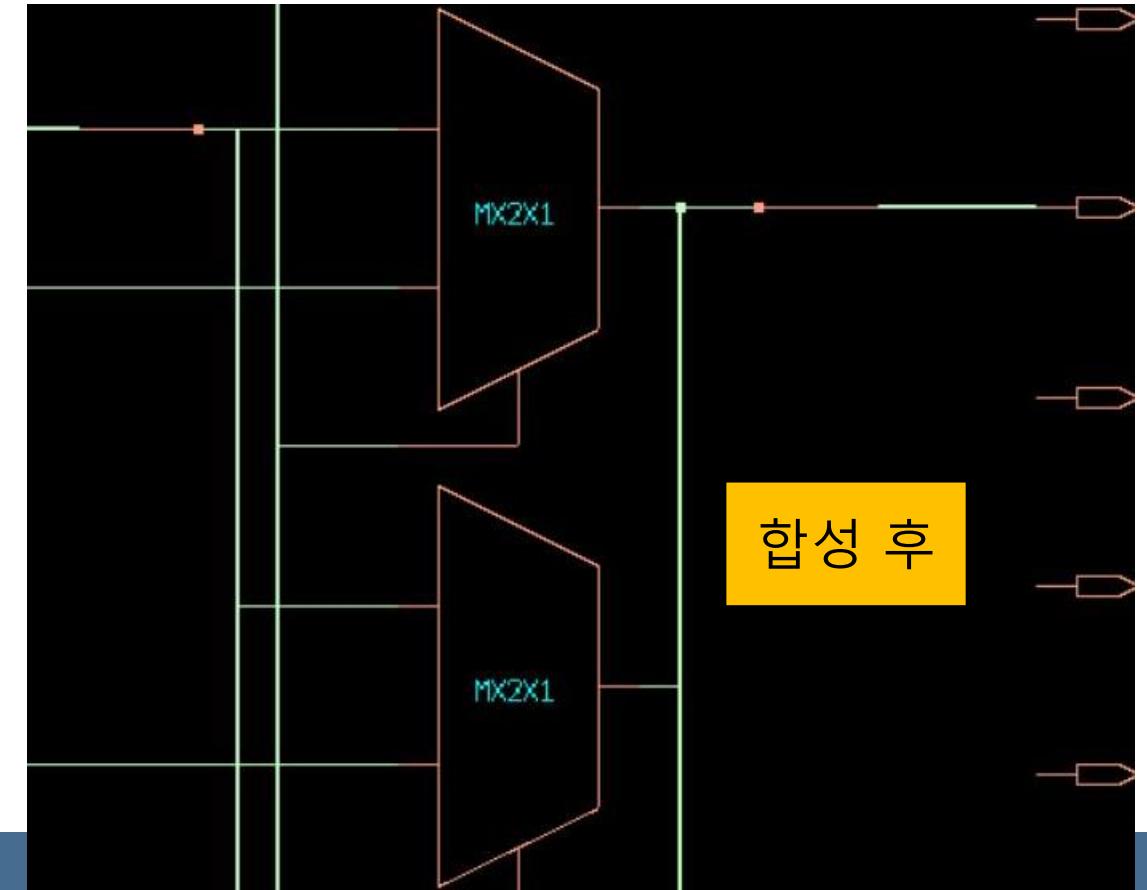
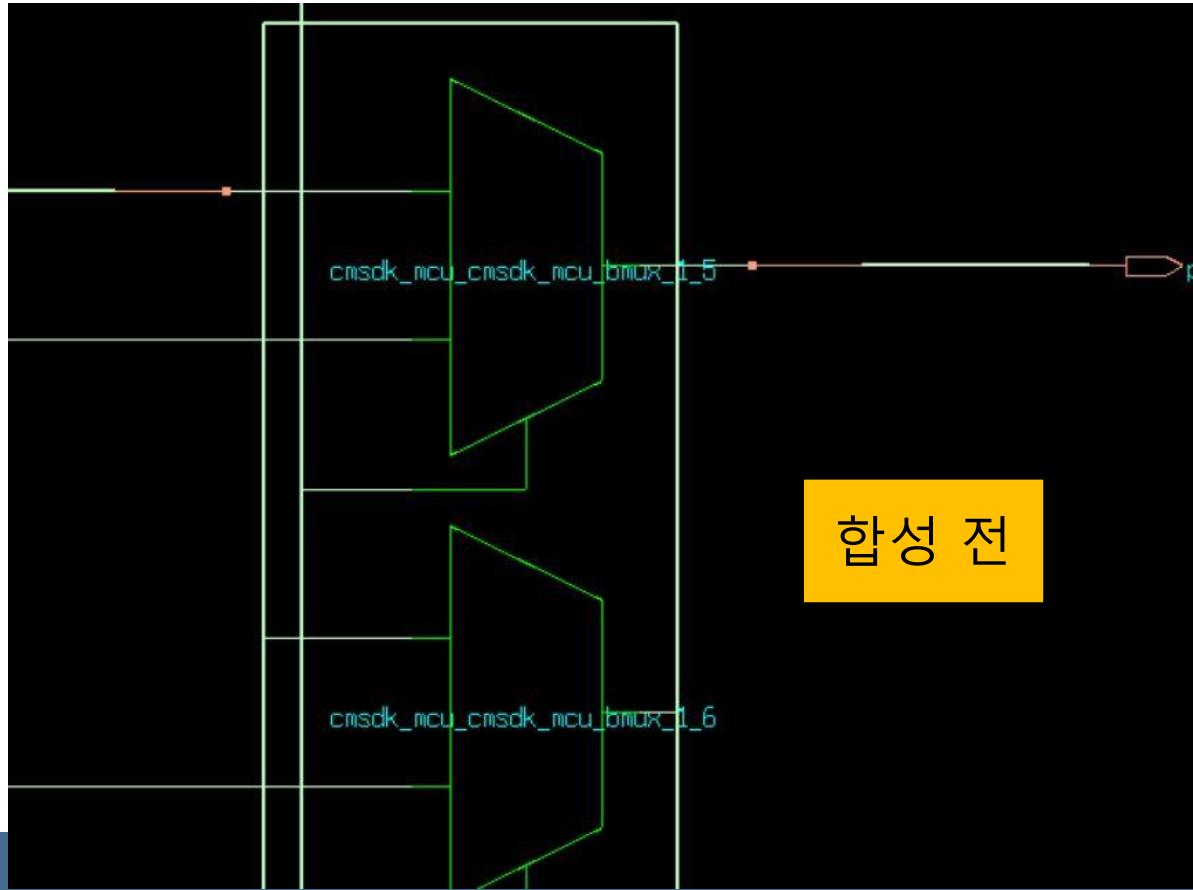
- syn\_opt 실행 후 schematic 확인



# Synthesis

## genus

- 합성 전 후 비교
- 공정사의 STD셀로 바뀐 것을 확인할 수 있음



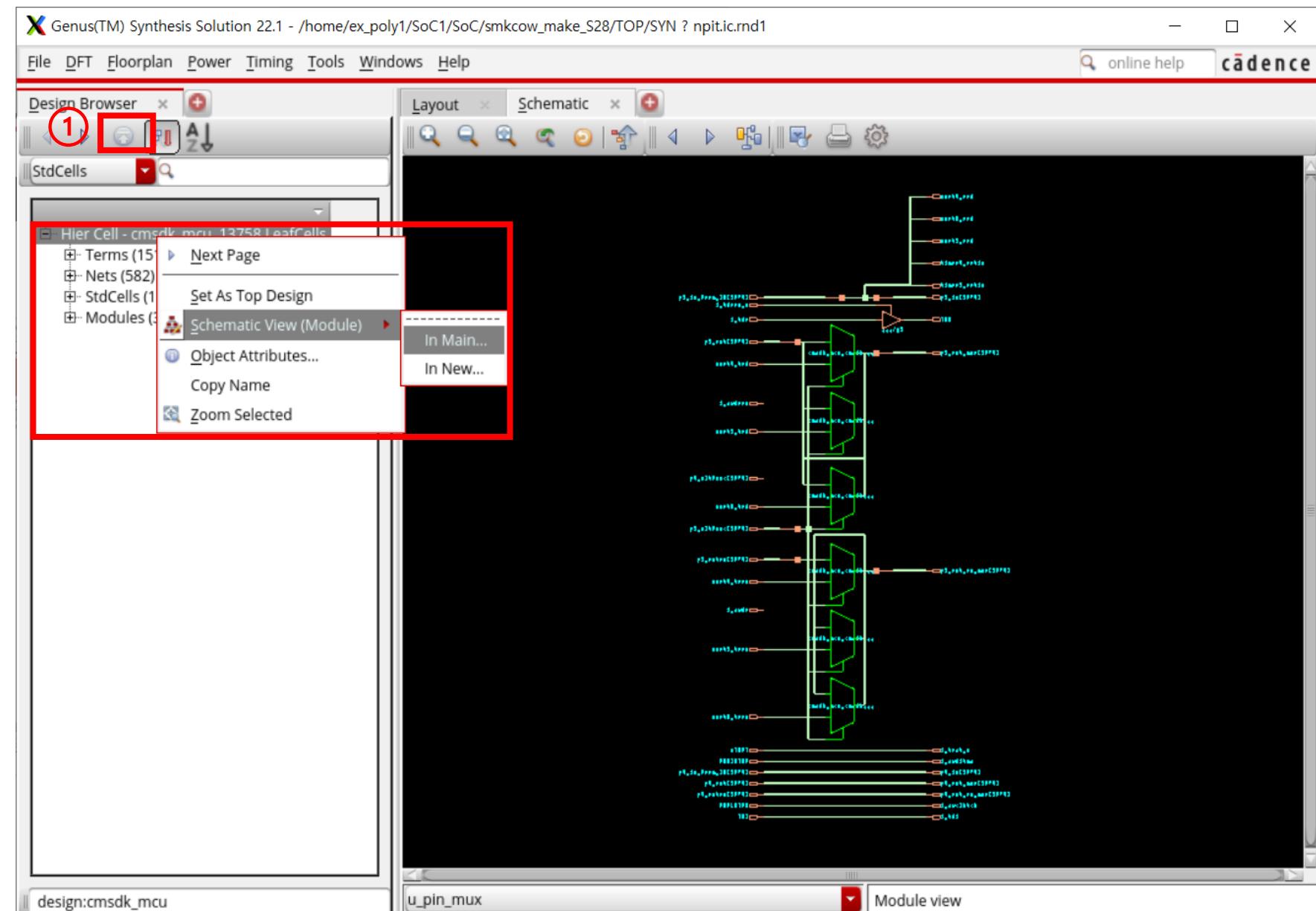
# Synthesis

genus

- TOP 디자인으로 이동

1번 클릭

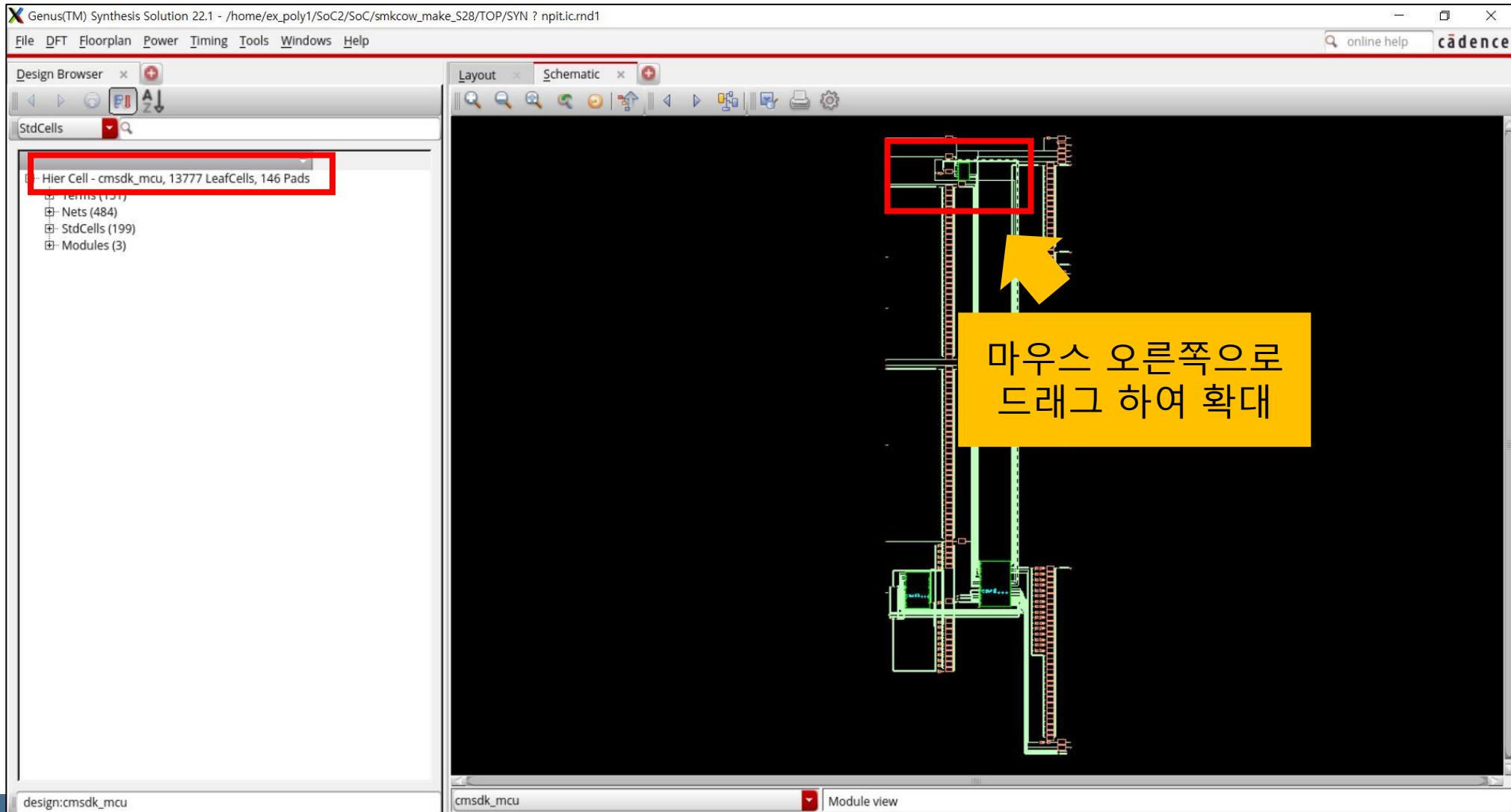
- Hier Cell 우 클릭
- schematic view 클릭
- In Main 클릭



# Synthesis

genus

\$> gui\_show

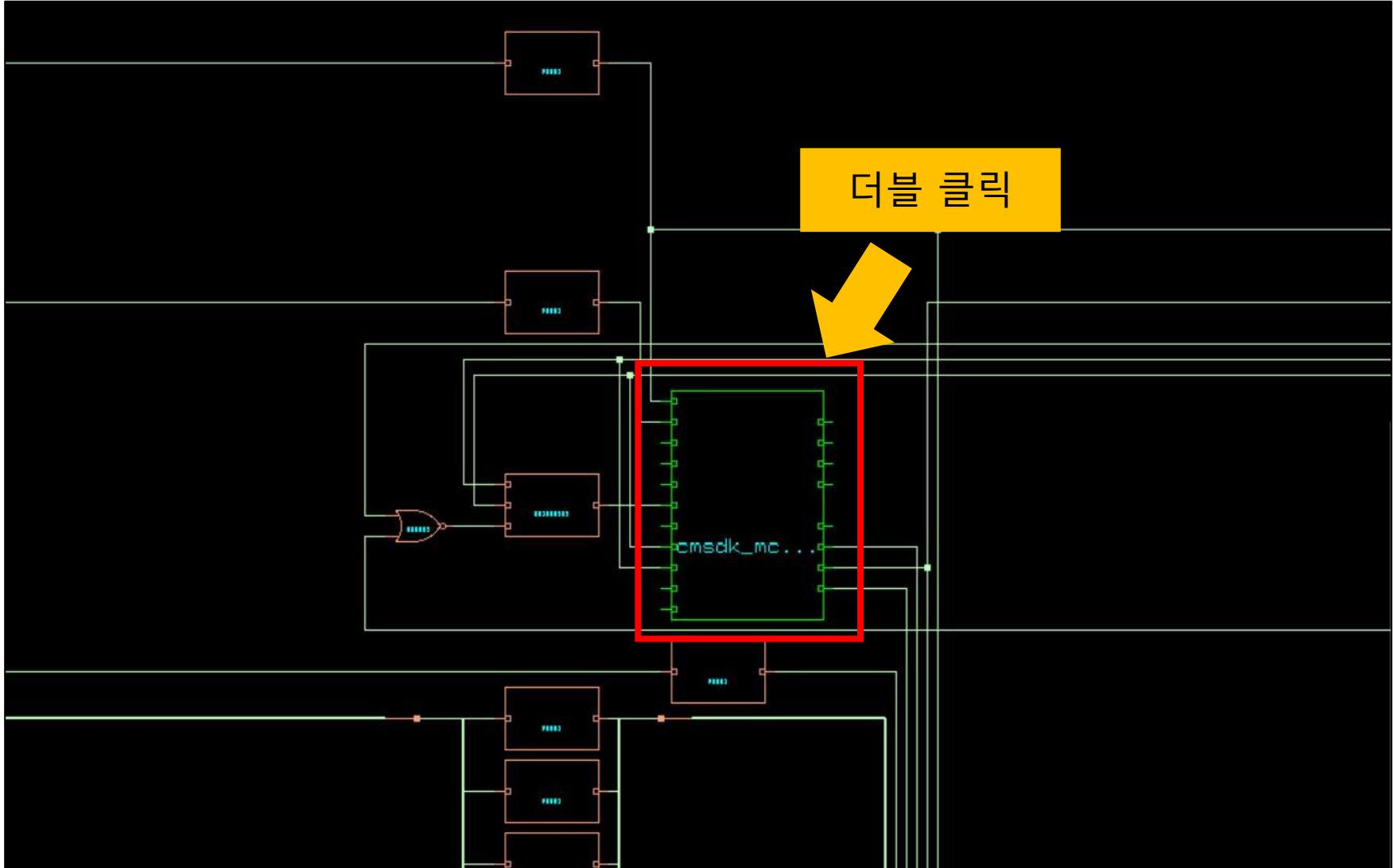


# Synthesis

genus

\$> gui\_show

- syn\_opt 실행 후 schematic 확인

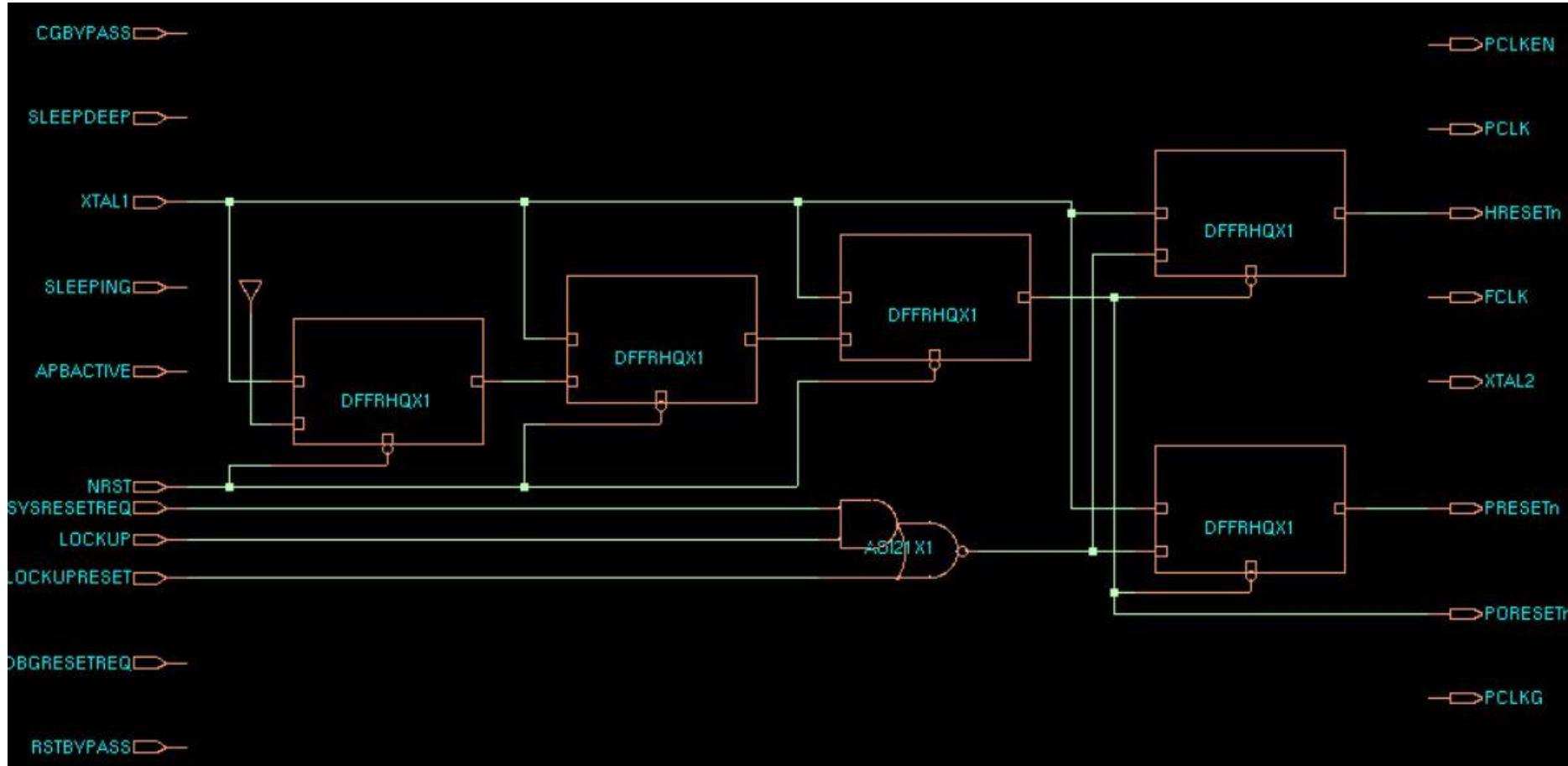


# Synthesis

genus

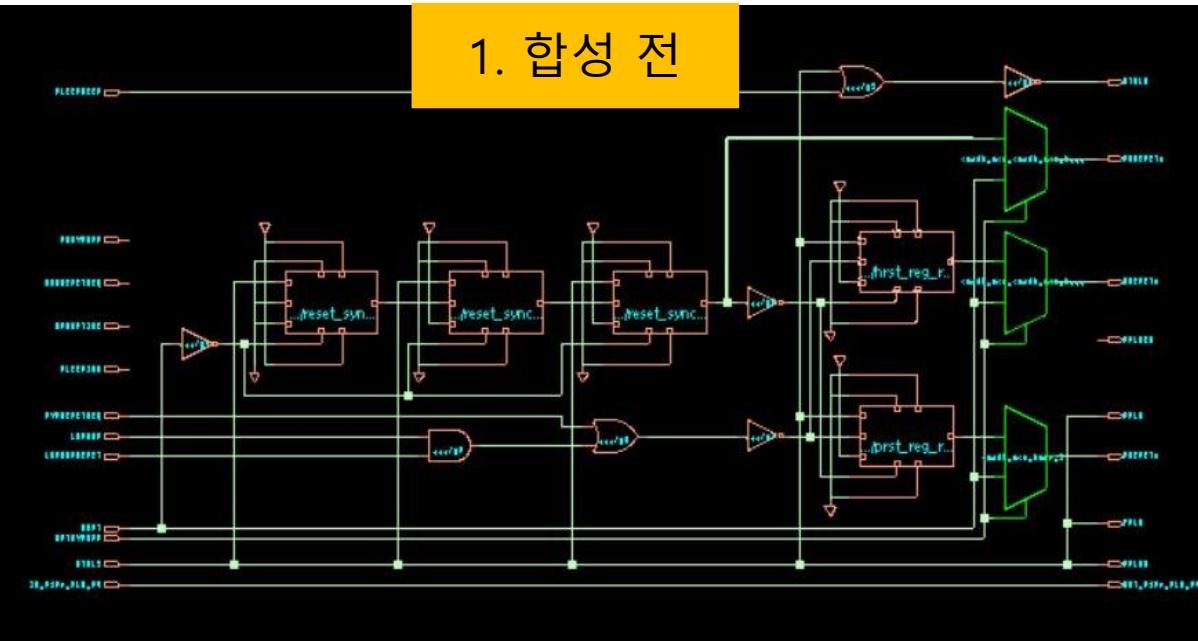
\$> gui\_show

- syn\_opt 실행 후 schematic 확인
- syn\_map과 차이가 없음  
→ 이 모듈에는 최적화할 곳이 없기 때문임

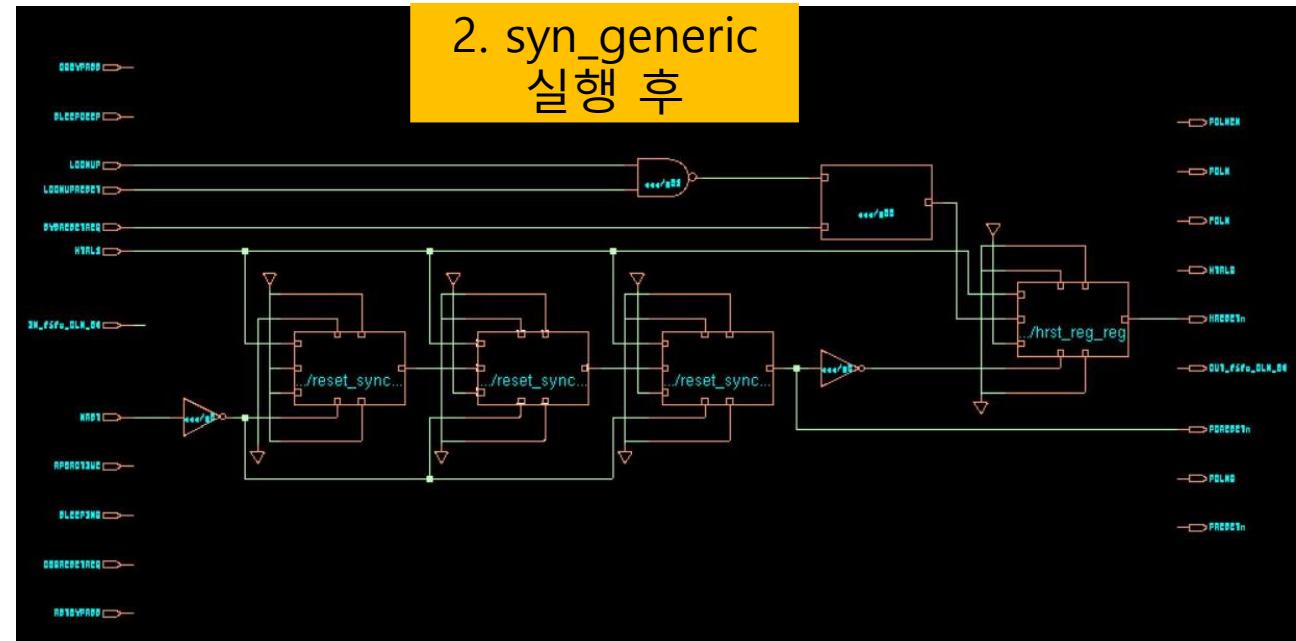


# Synthesis

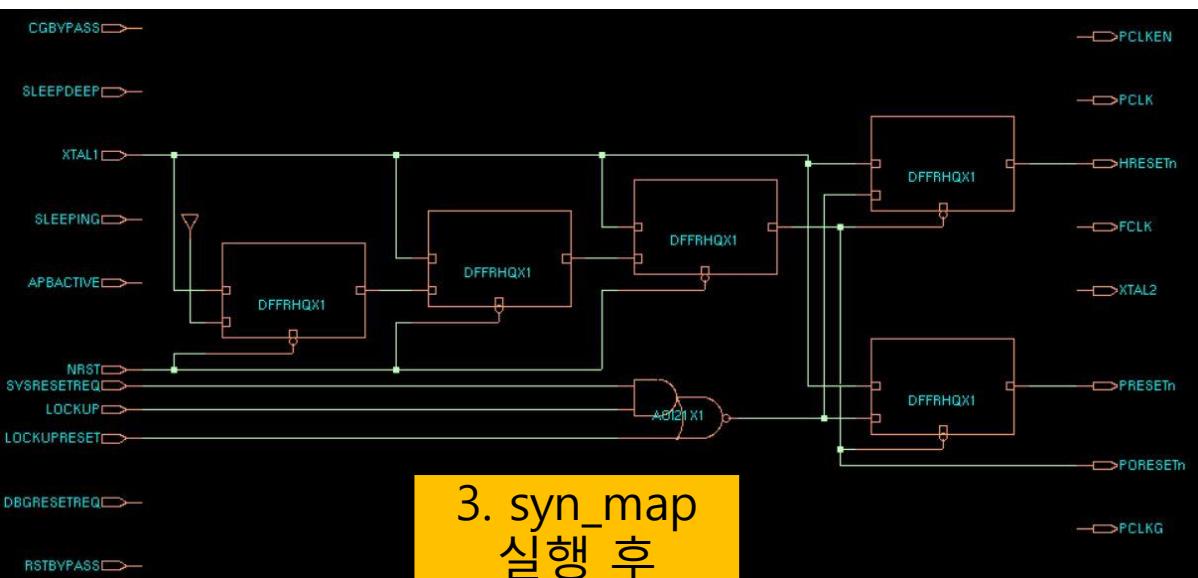
1. 합성 전



2. syn\_generic  
실행 후



3. syn\_map  
실행 후



4. syn\_opt  
실행 후

