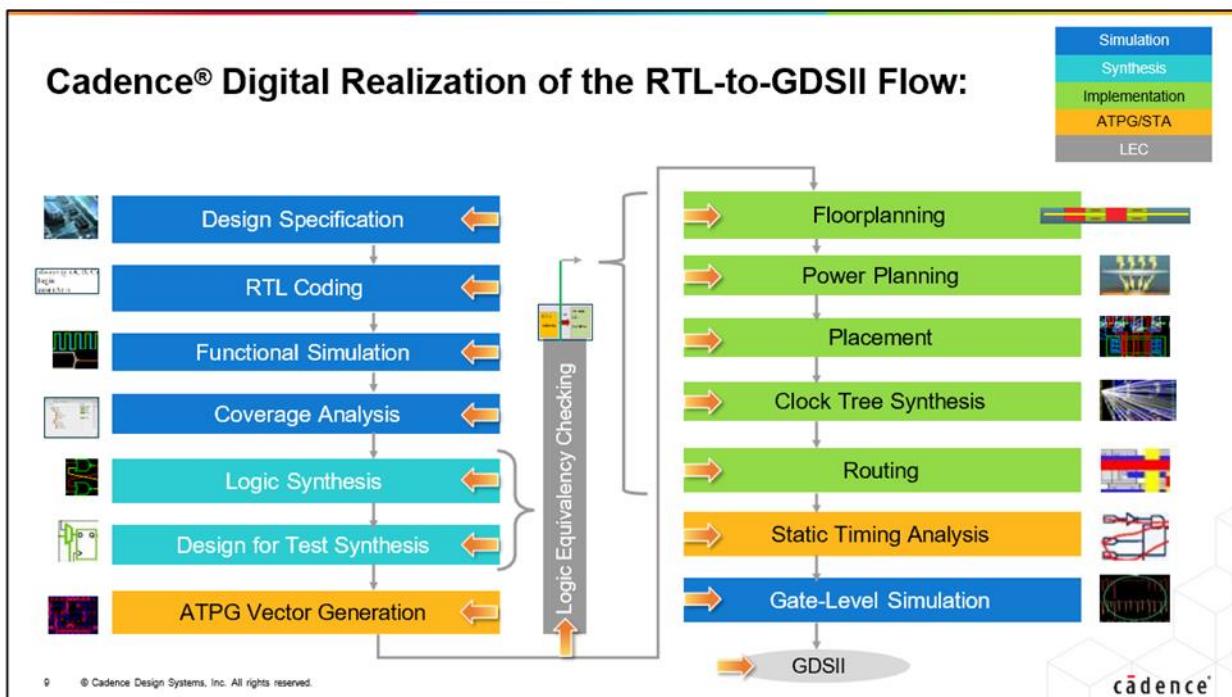


HƯỚNG DẪN SỬ DỤNG CADENCE TRONG THIẾT KẾ IC SỐ

Mục lục

I.	Tổng quan về thiết kế IC số:	3
II.	Viết Specification (sử dụng Word):	3
III.	Tạo lập các thư mục để chạy phần mềm	3
IV.	Code RTL (sử dụng Gedit)	5
1.	Thiết lập code Verilog cho project	5
2.	Chạy Simulation cho RTL (sử dụng Xcelium):	7
V.	Synthesis (sử dụng Genus)	10
VI.	Physical design (sử dụng Innovus)	22
1.	Quy trình sử dụng Innovus:	22
2.	Chạy Innovus	24
VII.	Signoff (sử dụng Tempus, Voltus, Pegasus)	41
Phụ lục		42
	Những lưu ý khi sử dụng database:	42

I. Tổng quan về thiết kế IC số:



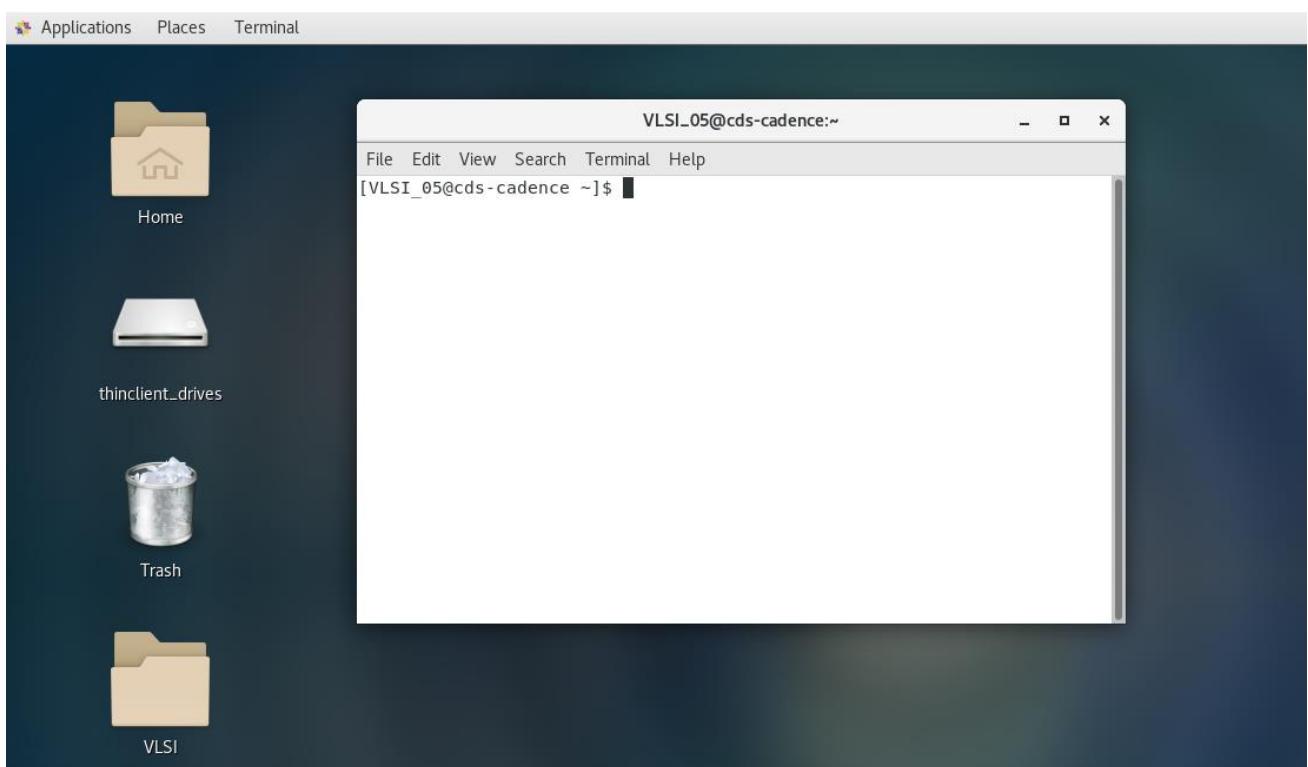
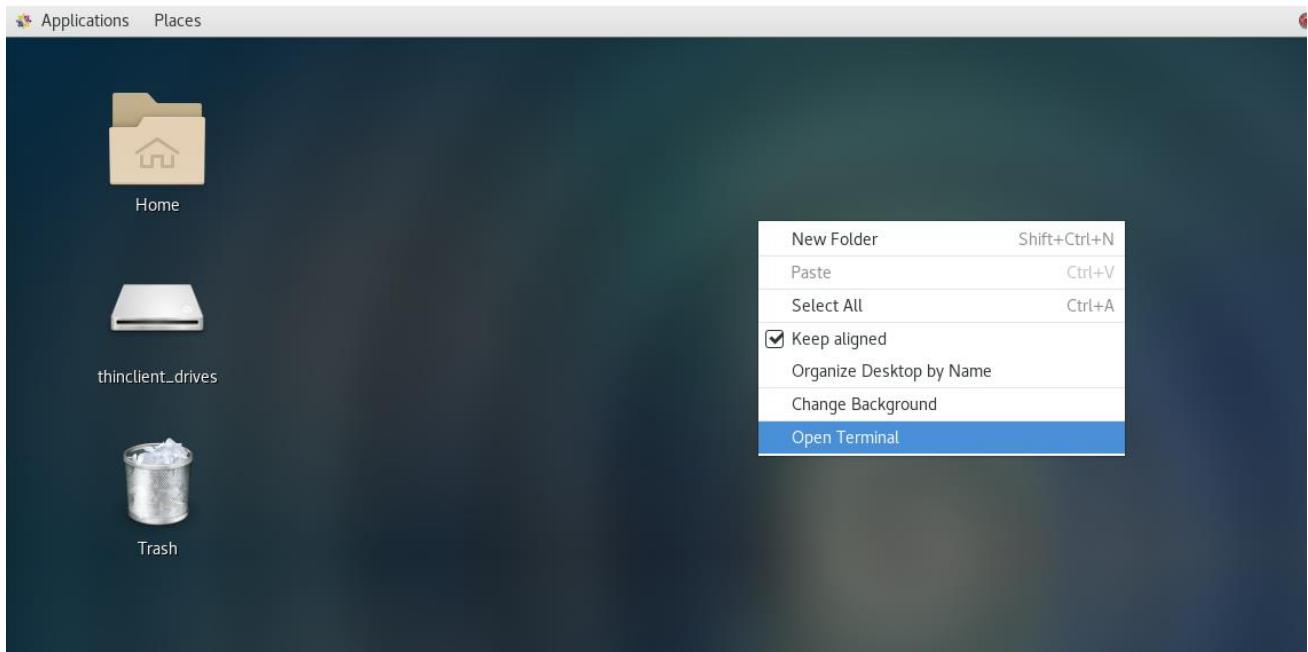
II. Viết Specification (sử dụng Word):

Thiết kế và miêu tả thiết kế

III. Tạo lập các thư mục để chạy phần mềm

Bước 1: Các lệnh ở trong bản hướng dẫn đều dùng trong Terminal vậy ta cần mở nó bằng cách:

Click chuột phải vào màn hình và chọn **Open Terminal**



Bước 2: Tạo 1 thư mục làm cho Project và đặt tên bằng cách nhập dòng lệnh sau:

```
$ mkdir <path/to/project>  
$ cd <path/to/project>
```

```
[VLSI_05@cds-cadence Documents]$ mkdir digital_clock_full  
[VLSI_05@cds-cadence Documents]$ cd digital_clock_full
```

Bước 3: Đối với 1 project digital thì ta cần có tổng cộng 8 folders rtl, tcl, libraries, work, QRC, constraints, sim, def.

Tạo 8 folders này bằng cách:

```
$ mkdir rtl tcl library work QRC constraints sim def
```

Xem nó được tạo ra chưa bằng cách:

```
$ ls
```

```
[VLSI_05@cds-cadence digital_clock_full]$ mkdir rtl tcl library work QRC constraints sim def
[VLSI_05@cds-cadence digital_clock_full]$ ls
constraints  def  library  QRC  rtl  sim  tcl  work
[VLSI_05@cds-cadence digital_clock_full]$
```

IV. Code RTL (sử dụng Gedit)

1. Thiết lập code Verilog cho project

Bước 1: Tạo 1 file .v để viết chương trình vào

```
$ gedit
```

Sẽ hiện lên cửa sổ:



Bước 2: Nhập code vào đó:

```
*Untitled Document 1
Save  x

module counter_led_7(enable,clk,rst,led_unit,led_ten);
input enable,clk,rst;
output [6:0] led_unit;
output [6:0] led_ten;

wire [3:0] cnt;
wire [3:0] t,u;
counter cnt1 (.clk(clk),
    .rst(rst),
    .counter(cnt));
extrac_bit ex1 (.number(cnt),
    .ten_o(t),
    .unit_o(u));

led_ten_7_seg l1 (.en(enable),
    .ten_i(t),
    .led_7_seg(led_ten));

led_unit_7_seg l2 (.en(enable),
    .unit_i(u),
    .led_7_seg(led_unit));

endmodule
```

Bước 3: Nhấn Save và lưu nó tại folder **rtl** với tên file <tên>.v thì ta sẽ được chuyển được dạng file sang Verilog:

```
counter_led_7.v
~/Documents/digital_clock_full/rtl

module counter_led_7(enable,clk,rst,led_unit,led_ten);
input enable,clk,rst;
output [6:0] led_unit;
output [6:0] led_ten;

wire [3:0] cnt;
wire [3:0] t,u;
counter cnt1 (.clk(clk),
    .rst(rst),
    .counter(cnt));
extrac_bit ex1 (.number(cnt),
    .ten_o(t),
    .unit_o(u));

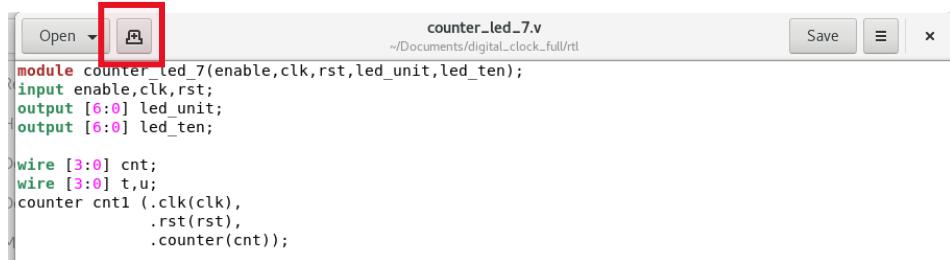
led_ten_7_seg l1 (.en(enable),
    .ten_i(t),
    .led_7_seg(led_ten));

led_unit_7_seg l2 (.en(enable),
    .unit_i(u),
    .led_7_seg(led_unit));

endmodule
```

Bước 4: RTL không chỉ có 1 file như vậy mà ít nhất là 2 file gồm 1 file function và 1 file testbench. Còn với các thiết kế phức tạp hơn thì sẽ cần nhiều hơn như vậy nên ta cần phải tạo thêm file .v

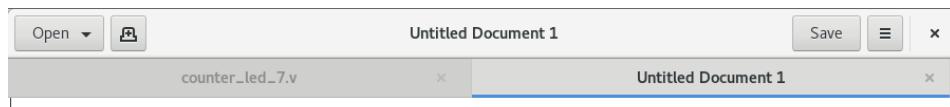
Để tạo thêm 1 cửa sổ gedit ta nhấp vào icon sau



```
counter_led_7.v
~/Documents/digital_clock_full/rtl
module counter_led_7(enable,clk,rst,led_unit,led_ten);
  input enable,clk,rst;
  output [6:0] led_unit;
  output [6:0] led_ten;

  wire [3:0] cnt;
  wire [3:0] t,u;
  counter cnt1 (.clk(clk),
                .rst(rst),
                .counter(cnt));
```

Ta được 1 tab khác nhau sau:



Sau đó ta sẽ tiếp tục làm như từ bước 2 trở đi.

2. Chạy Simulation cho RTL (sử dụng Xcelium):

Để biết được code Verilog của mình đã đúng với Spec chưa thì ta cần mô phỏng xem kết quả bằng Xcelium

Bước 1: Để mô phỏng thì ta cần vào tool Incisive bằng cách chạy lệnh:

```
$ xrun -64bit -access rwc -gui < path/to/file.v/của/module/nhỏ>
<path/to/file.v/của/module/lớn> <path/to/file_tb.v/của/module/lớn>
```

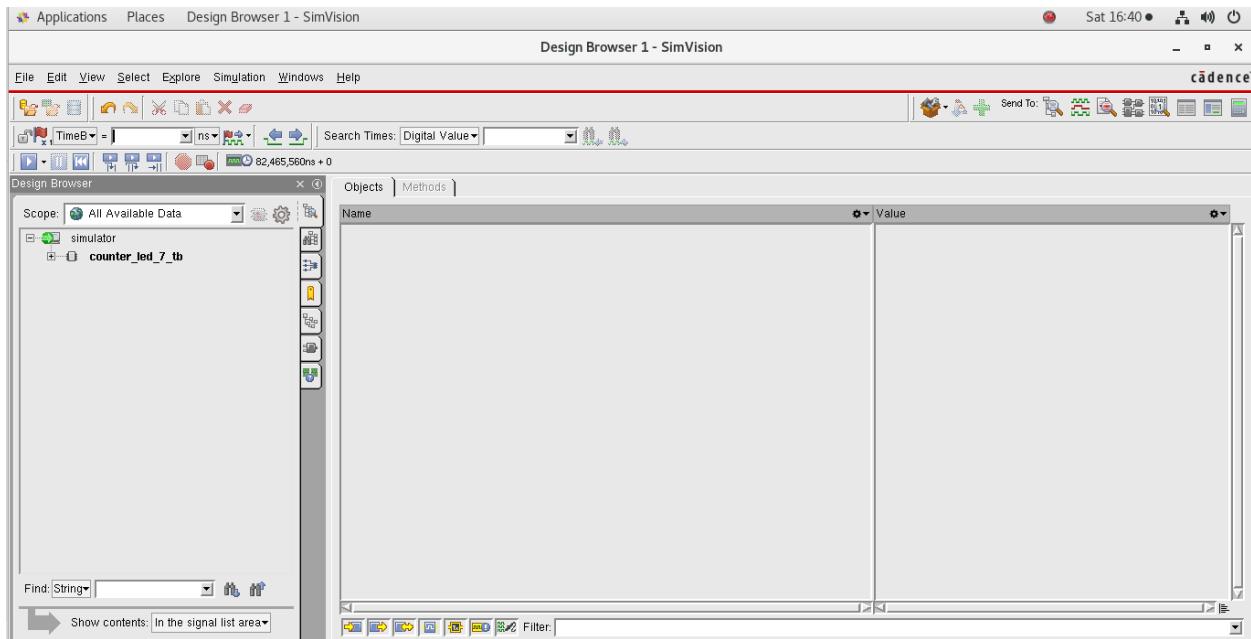
Lưu ý: phải đưa các đường dẫn của module con trước đường dẫn của module lớn và của testbench

VD:

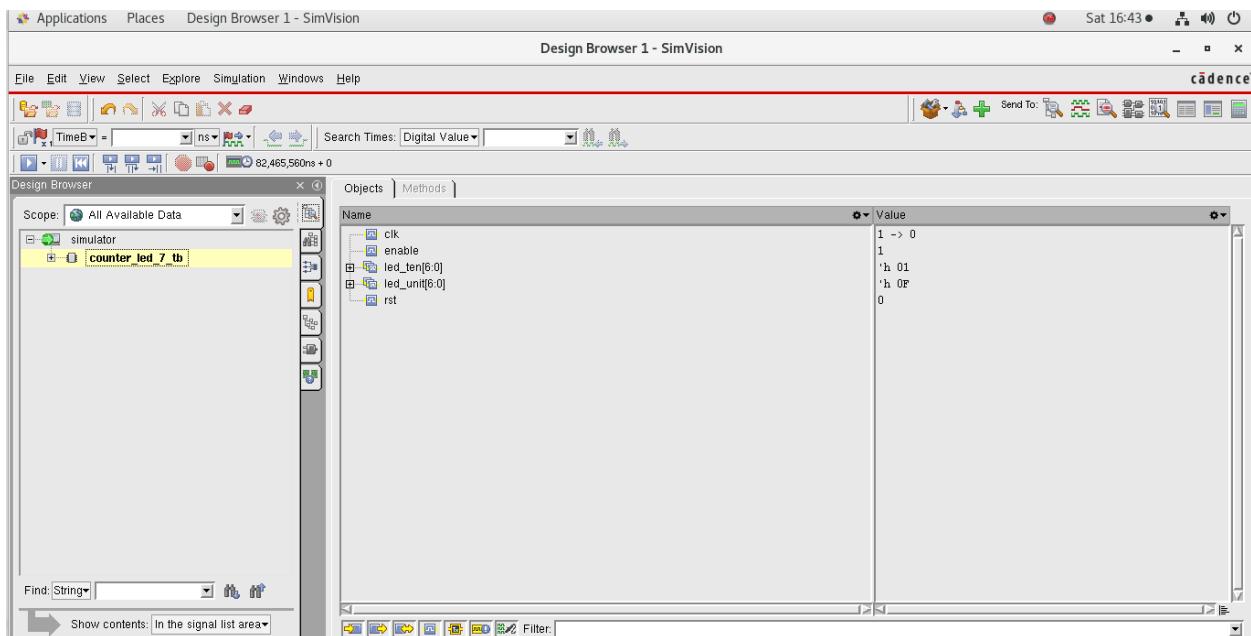
```
$ xrun -64bit -access rwc -gui
/home/VLSI_05/Documents/digital_clock_full/rtl/counter.v \
/home/VLSI_05/Documents/digital_clock_full/rtl/extrac_bit.v \
/home/VLSI_05/Documents/digital_clock_full/rtl/led_ten_7_seg.v \
/home/VLSI_05/Documents/digital_clock_full/rtl/led_unit_7_seg.v \
```

```
/home/VLSI_05/Documents/digital_clock_full/rtl/counter_led_7.v \
/home/VLSI_05/Documents/digital_clock_full/rtl/counter_led_7_tb.v
```

Ta sẽ được kết quả như sau nếu code của ta đúng:

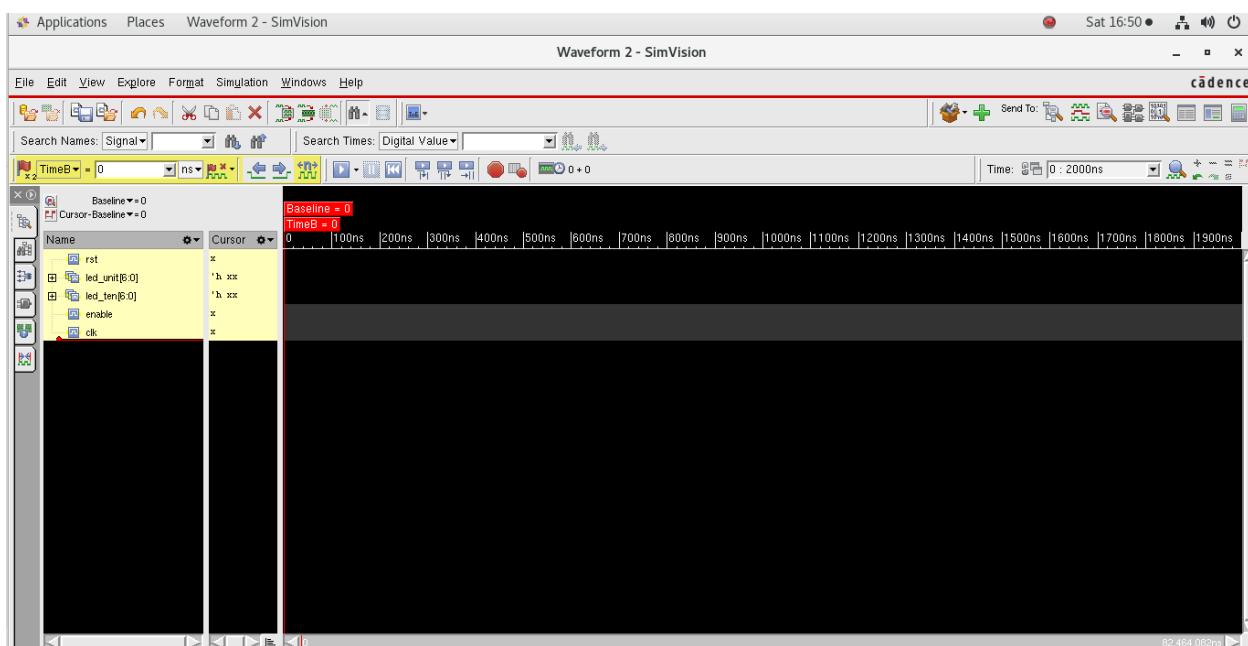
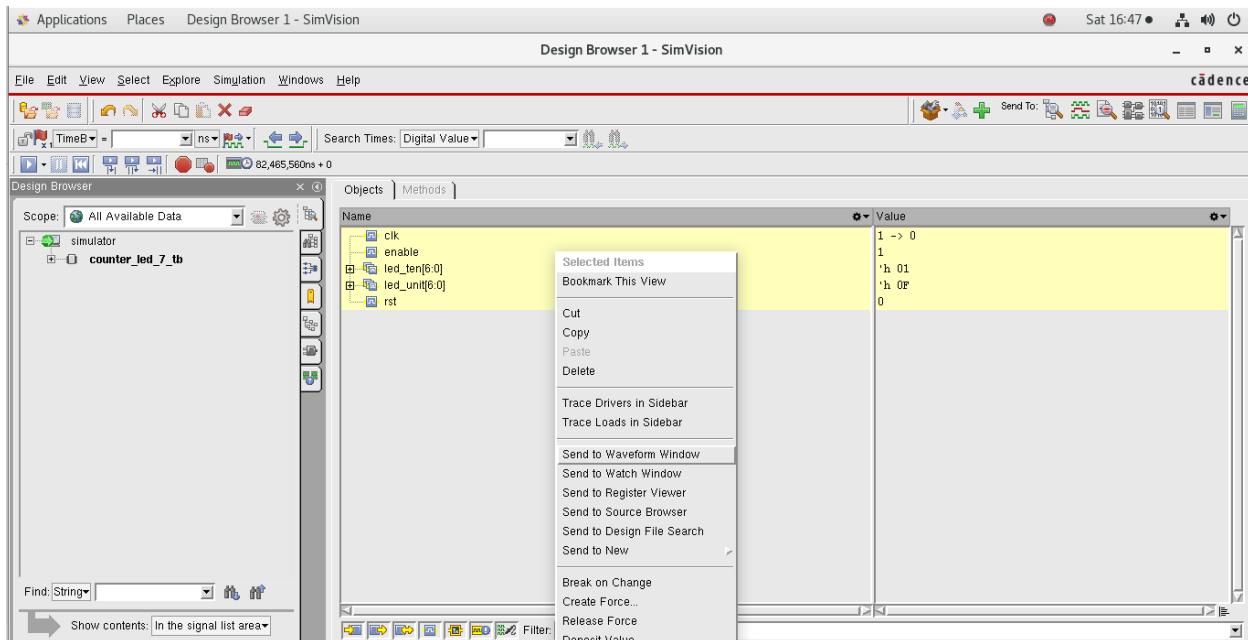


Bước 2: Để xuất hiện các biến của testbench ta click vào testbench:

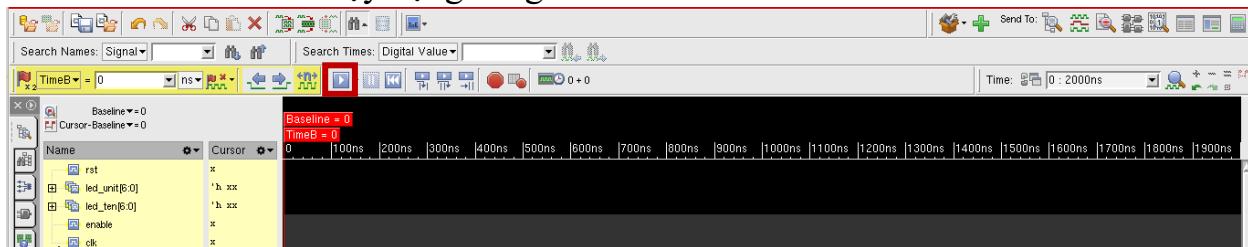


Bước 3: Để mở Cửa sổ dạng sóng ta cần chọn tất cả các biến bằng cách nhấn **Ctrl + chuột trái** rồi chọn từng biến, sau đó click chuột phải vào đó rồi chọn **Send to**

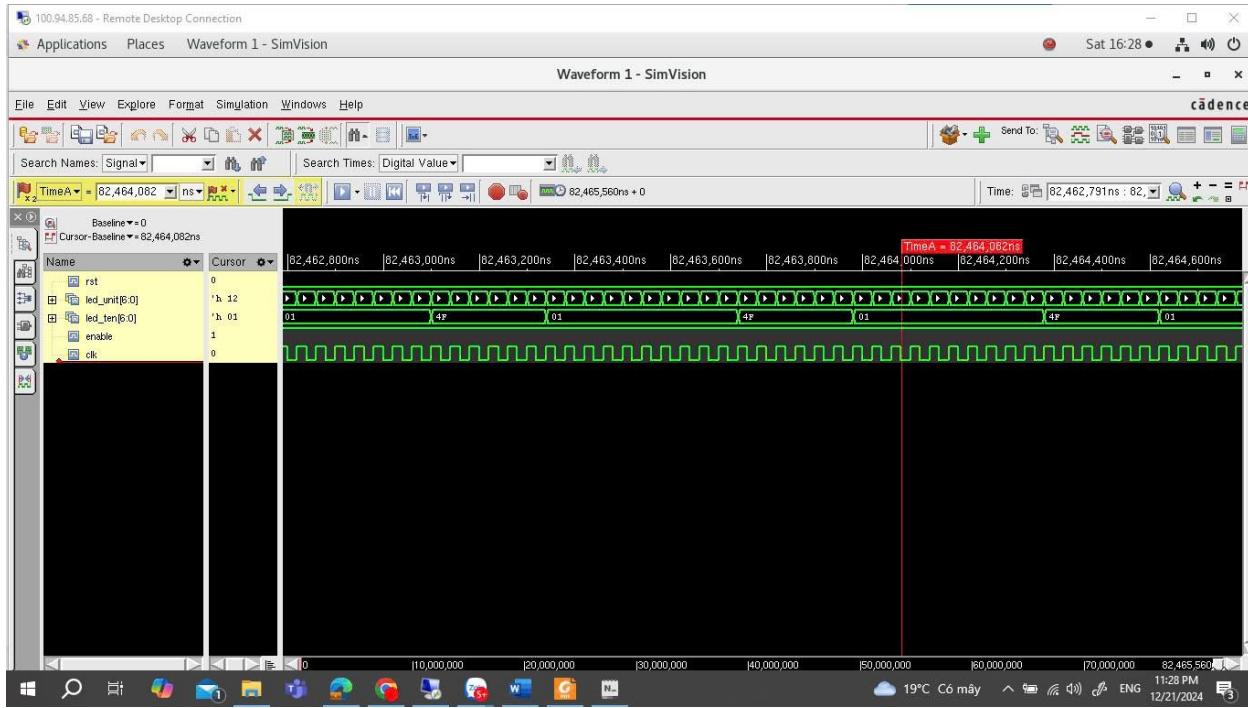
Waveform Window



Bước 5: Để bắt đầu chạy dạng sóng ta click icon trên thanh điều khiển



Và ta được kết quả sau:



Kết thúc RTL

V. Synthesis (sử dụng Genus)

Để chạy được Synthesis chúng ta cần các file sau :

- Synthesis library (thư viện để sysnthesis)
- Verilog netlist
- Constraints
- Library Exchange Foorpland

Bước 1: xác định các file .lib và .lef cần thiết, đường dẫn

Đường dẫn tới các file .lib:

/home/Cadence/PDK/gsclib045_all_v4.4/gsclib045/timing

Đường dẫn tới các file .lef:

/home/Cadence/PDK/gsclib045_all_v4.4/gsclib045/lef

Bước 2: tạo file constraints.sdc ở folder constraints:

```
$ cd path/to/constraints/folder  
$ gedit constraints.sdc
```

Rồi nhập nội dung.

VD: đối với counter_led_7_seg file constraints như sau:

```
*constraints.sdc  
~/Documents/digital_clock_full...  
# Tạo clock đầu vào với chu kỳ 10ns (tần số 100 MHz)  
create_clock -name clk -period 10.0 [get_ports clk]  
  
# Ràng buộc reset (asynchronous reset)  
set_input_delay -clock clk 0.5 [get_ports rst]  
  
# Ràng buộc input enable  
set_input_delay -clock clk 0.5 [get_ports enable]  
  
# Ràng buộc output delays cho các tín hiệu LED  
set_output_delay -clock clk 1.0 [get_ports led_unit]  
set_output_delay -clock clk 1.0 [get_ports led_ten]  
  
# Đặt false path cho các đường không quan trọng  
set_false_path -from [get_ports rst]  
set_false_path -from [get_ports enable]
```

Bước 3: Để có thể thiết lập môi trường khi chạy genus ta cần 1 file setup.tcl nên ta sẽ phải tạo file đó bằng cách:

```
$ cd .. /tcl  
$ gedit setup.tcl
```

Trong file setup của module cơ bản có các phần sau:

```
# Set tên DESIGN
  set DESIGN "<tên_module_chính>"

# Thiết lập các đường dẫn
  set RTL_PATH /path/to/rtl/folder"
  set LIB_PATH /path/to/lib/folder /path/to/lef/folder"
  set TCL_PATH "path/to/tcl/folder /path/to/constraints/folder"

# Đường dẫn SDC
  set SDC_FILE "/path/to/constraints.sdc"
  set MSGS_TO_BE_SUPRESSED {LBR-58 LBR-40 LBR-41 VLOGPT-35}

# Bỏ qua các thông báo không cần thiết
  suppress_messages {LBR-30 LBR-31 LBR-40 LBR-41 LBR-72 LBR-77 LBR-162}

# Danh sách file .lib
  set LIB_LIST {
    Các \
    Thư \
    Viện \
    Cân \
    Thiết \
  }

# Danh sách các file .lef
  set LEF_LIST {
    Các \
    File \
    LEF \
  }

# Danh sách các file RTL
  set RTL_LIST {
    Tất \
    Cả \
    File \
  }
```

```

Verilog \
}

# Đường dẫn tìm kiếm

set_db hdl_track_filename_row_col true
set_db lp_power_unit mW
set_db init_lib_search_path $LIB_PATH
set_db init_hdl_search_path $RTL_PATH
set_db script_search_path $TCL_PATH
set_db error_on_lib_lef_pin_inconsistency true

# Đọc các file LIB
read_libs $LIB_LIST

# Đọc file LEF
read_physical -lef $LEF_LIST

# Đọc file RTL
read_hdl $RTL_LIST

```

VD:

The image shows two windows of a Cadence tool interface. The left window is titled 'setup1.tcl' and contains the following code:

```

set DESIGN "counter_led_7"
# Đường dẫn RTL
set RTL_PATH "/home/VLSI_05/Documents/digital_clock_full/rtl"
set LIB_PATH "/home/Cadence/PDK/gsclib045_all_v4.4/gsclib045/timing /home/Cadence/PDK/gsclib045_all_v4.4/gsclib045/lef"
set TCL_PATH "/home/VLSI_05/Documents/digital_clock_full/tcl /home/VLSI_05/Documents/digital_clock_full/constraints"
# Đường dẫn SDC
set SDC_FILE "/home/VLSI_05/Documents/digital_clock_full/constraints/constraints.sdc"

set MSGS_TO_BE_SUPPRESSED {LBR-58 LBR-40 LBR-41 LBR-72 LBR-77 LBR-162}

# Danh sách thư viện timing
set LIB_LIST {
fast_vddiv0_basicCells.lib \
slow_vddiv0_basicCells.lib \
fast_vddiv2_basicCells.lib \
slow_vddiv2_basicCells.lib
}

# Đường dẫn các file LEF
set LEF_LIST {
gsclib045_macro.lef \
gsclib045_multibitsdff.lef \
gsclib045_tech.lef
}

```

The right window is also titled 'setup1.tcl' and contains the following code:

```

*gconstraints.sdc* \
gsclib045_multibitsdff.lef \
gsclib045_tech.lef \
}

# Danh sách các file RTL
set RTL_LIST {
counter_led_7.v \
counter.v \
extrac_bit.v \
led_unit_7_seg.v \
led_ten_7_seg.v \
}

suppress_messages {LBR-30 LBR-31 LBR-40 LBR-41 LBR-72 LBR-77 LBR-162}

# Đường dẫn tìm kiếm
set_db hdl_track_filename_row_col true
set_db lp_power_unit mW
set_db init_lib_search_path $LIB_PATH
set_db init_hdl_search_path $RTL_PATH
set_db script_search_path $TCL_PATH
set_db error_on_lib_lef_pin_inconsistency true

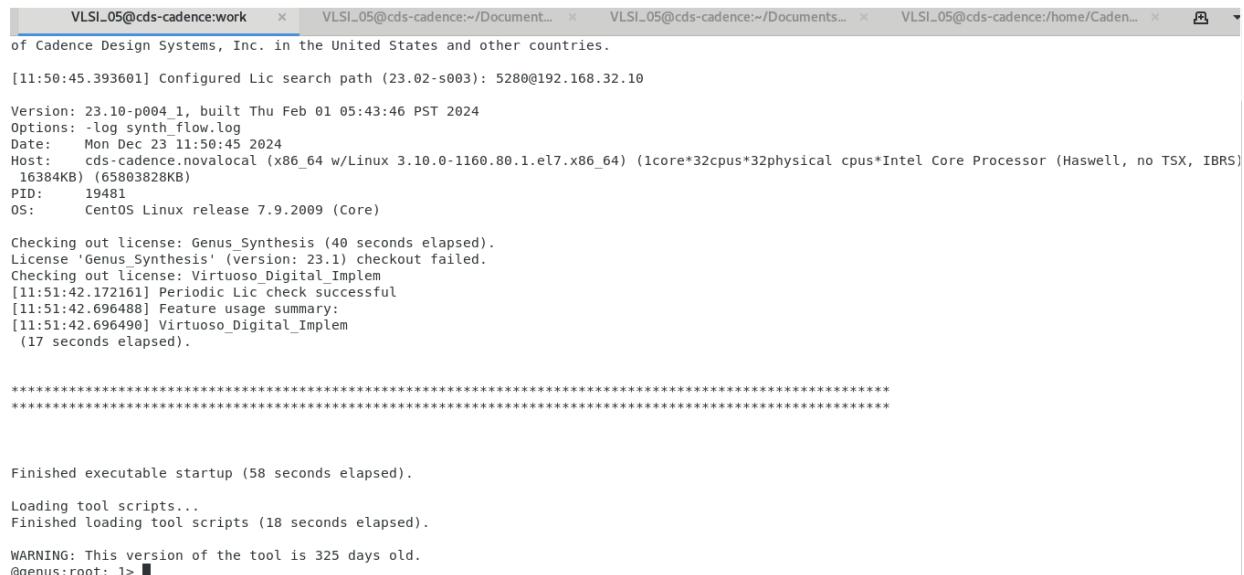
# Đọc thư viện chuẩn
puts "Reading standard cell libraries..."
read_libs $LIB_LIST
# Đọc file LEF
puts "Reading LEF files..."
read_physical -lef $LEF_LIST
# Đọc file RTL
puts "Reading RTL files..."
read_hdl $RTL_LIST

```

Bước 4: Sau khi đã chuẩn bị xong ta khởi động Genus, và để kết quả cần lưu vào folder work:

```
$ cd ../work  
$ genus -log synth_flow.log
```

Sau khi chạy lệnh đợi 1 chút ta sẽ thấy trên Terminal sẽ đổi thành giao diện khác:



```
VLSI_05@cds-cadence:work ✘ VLSI_05@cds-cadence:~/Documents... ✘ VLSI_05@cds-cadence:~/Documents... ✘ VLSI_05@cds-cadence:/home/Caden... ✘  
of Cadence Design Systems, Inc. in the United States and other countries.  
[11:50:45.393601] Configured Lic search path (23.02-s003): 5280@192.168.32.10  
Version: 23.10-p004_1, built Thu Feb 01 05:43:46 PST 2024  
Options: -log synth_flow.log  
Date: Mon Dec 23 11:50:45 2024  
Host: cds-cadence.novalocal (x86_64 w/Linux 3.10.0-1160.80.1.el7.x86_64) (1core*32cpus*32physical cpus*Intel Core Processor (Haswell, no TSX, IBRS)  
16384KB) (65803828KB)  
PID: 19481  
OS: CentOS Linux release 7.9.2009 (Core)  
  
Checking out license: Genus_Synthesis (40 seconds elapsed).  
License 'Genus_Synthesis' (version: 23.1) checkout failed.  
Checking out license: Virtuoso_Digital_Implem  
[11:51:42.172161] Periodic Lic_check successful  
[11:51:42.696488] Feature usage summary:  
[11:51:42.696490] Virtuoso_Digital_Implem  
(17 seconds elapsed).  
  
*****  
*****  
  
Finished executable startup (58 seconds elapsed).  
Loading tool scripts...  
Finished loading tool scripts (18 seconds elapsed).  
WARNING: This version of the tool is 325 days old.  
@genus:root: l>
```

Bước 5: Ta thiết lập các biến trong khi chạy Genus bằng cách chạy:

```
source ../tcl/setup.tcl
```

Sau khi chạy lệnh trên nó có thể có một vài Warning nhưng ta có thể bỏ qua:

```
VLSI_05@cds-cadence:work
File Edit View Search Terminal Tabs Help
VLSI_05@cds... × VLSI_05@cds... × VLSI_05@cds... × VLSI_05@cds... ×
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell FILL1 cannot be found in library.
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell FILL16 cannot be found in library.
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell FILL2 cannot be found in library.
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell FILL32 cannot be found in library.
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell FILL4 cannot be found in library.
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell FILL64 cannot be found in library.
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell FILL8 cannot be found in library.
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell FSWNX1 cannot be found in library.
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell FSWX1 cannot be found in library.
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell HSWDNX1 cannot be found in library.
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell HSWDX1 cannot be found in library.
Warning : Physical cell not defined in library. [PHYS-279]
          : The physical (LEF) cell HSWNX1 cannot be found in library.
Warning : Maximum message print count reached. [MESG-11]
          : Maximum print count of '20' reached for message 'PHYS-279'.
@file(setup1.tcl) 53: puts "Reading RTL files..."
Reading RTL files...
@file(setup1.tcl) 54: read_hdl $RTL_LIST
#@ End verbose source ../tcl/setup1.tcl
@genus:root: 2> █
```

Bước 6: Kiểm tra và chuẩn bị các file RTL cho quá trình synthesis hay simulation.

[elaborate](#)

Ta có thể bỏ qua 1 vài Warning xuất hiện nếu chương trình đã chạy xong như này:

```

VLSI_05@cds-cadence:work
File Edit View Search Terminal Tabs Help
VLSI_05@cds... VLSI_05@cds... VLSI_05@cds... VLSI_05@cds...
| ume_ssm | 0 | 0 | 0.00 |
| ume_cse | 0 | 0 | 0.00 |
| ume_shrink | 0 | 0 | 0.00 |
| ume_sweep | 0 | 0 | 0.00 |
Starting optimize datapath shifters [v1.0] (stage: post_elab, startdef: counter_led_7, recur: true)
Completed optimize datapath shifters (accepts: 0, rejects: 0, runtime: 0.000s)
Starting clip mux common data inputs [v1.0] (stage: post_elab, startdef: counter_led_7, recur: true)
Completed clip mux common data inputs (accepts: 0, rejects: 0, runtime: 0.000s)
Starting clip the non-user hierarchies [v2.0] (stage: post_elab, startdef: counter_led_7, recur: true)
Completed clip the non-user hierarchies (accepts: 0, rejects: 0, runtime: 0.000s)
Starting basic netlist cleanup [v1.0] (stage: post_elab, startdef: counter_led_7, recur: true)
Completed basic netlist cleanup (accepts: 0, rejects: 0, runtime: 0.000s)
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 |
SUM: flow.cputime flow.realtime timing.setup.tns timing.setup.wns snapshot elaborate
UM:*
design:counter_led_7
@genus:root: 3>

```

Bước 7: Để kết quả cho ra đúng với yêu cầu ràng buộc (constraints) thì ta cần cho Genus đọc:

`read_sdc $SDC_FILE`

Bước 8: Tiếp đến ta cần generate các file RTL ra thành file netlist trung gian chưa được ánh xạ (mapping) vào thư viện công nghệ

`syn_gen`

Bước 9: Có được netlist trung gian rồi ta tiếp tục phải ánh xạ netlist trung gian trên vào thư viện công nghệ mà chúng ta đã khai báo để tạo ra 1 file netlist cụ thể đã được ánh xạ (mapped netlist):

`syn_map`

Bước 10: sau khi đã có bản netlist ta cần xem các cổng logic đã được tạo như thế nào:

`report_gate`

Và ta sẽ được kết quả dạng như sau

```

VLSI_05@cds-cadence:work
File Edit View Search Terminal Tabs Help
VLSI_05@cds... x VLSI_05@cds... x VLSI_05@cds... x VLSI_05@cds... x
Gate Instances Area Library
-----
ADDHX1      3  2.821  fast_vddiv0
CLKXOR2X1   1  0.684  fast_vddiv0
DFFRHQX1    4  6.156  fast_vddiv0
INVX1       3  0.513  fast_vddiv0
INVXL       2  0.342  fast_vddiv0
NAND2BX1    2  0.684  fast_vddiv0
NAND2X1    1  0.257  fast_vddiv0
NAND3BXL    3  1.282  fast_vddiv0
NAND3X1    3  1.282  fast_vddiv0
NOR2BX1    2  0.684  fast_vddiv0
OAI211X1   1  0.427  fast_vddiv0
OAI21X1    1  0.427  fast_vddiv0
OAI221X1   1  0.599  fast_vddiv0
OAI2BB1X1  1  0.427  fast_vddiv0
OR2X1      2  0.684  fast_vddiv0
-----
total        30 17.271

Type Instances Area Area %
-----
sequential   4  6.156  35.6
inverter     5  0.855  5.0
logic        21 10.260  59.4
physical_cells 0  0.000  0.0
-----
total        30 17.271 100.0
@genus:root: 8>

```

Bước 11: Vì kết quả cho ra là chưa tối ưu nên ta cần phải tối ưu hóa nó bằng cách chạy

syn_opt

```

VLSI_05@cds-cadence:work
File Edit View Search Terminal Tabs Help
VLSI_05@cds... x VLSI_05@cds... x VLSI_05@cds... x VLSI_05@cds... x
plc_star      0 ( 0 / 0 ) 0.00
drc_bufs      0 ( 0 / 0 ) 0.00
drc_fopt      0 ( 0 / 0 ) 0.00
drc_bufb      0 ( 0 / 0 ) 0.00
dup           0 ( 0 / 0 ) 0.00
crit_dnsz     0 ( 0 / 0 ) 0.00
crit_upsz     0 ( 0 / 0 ) 0.00
=====
Stage : incr_opt
=====
Message Summary
=====
| Id | Sev |Count| Message Text |
|---|---|---|
|CFM-1 |Info | 1|Wrote dofile.
|CFM-5 |Info | 1|Wrote formal verification information.
|PA-7 |Info | 4|Resetting power analysis results.
| | | |All computed switching activities are removed.
|SYNTH-5|Info | 1|Done mapping.
|SYNTH-7|Info | 1|Incrementally optimizing.
|SYNTH-33|Warning| 1|The selected flow setting will be removed in a
| | | |future release.
| | | |Contact Cadence support to understand current flows.
|---|---|---|
Info : Done incrementally optimizing. [SYNTH-8]
      : Done incrementally optimizing 'counter_led_7'.
      flow.cputime flow.realtime timing.setup.tns timing.setup.wns snapshot
UM:*
@genus:root: 9>

```

Và ta xem lại các công logic sau khi được tối ưu hóa bằng:

`report_gate`

Bước 12: Trước khi xuất ra các file output ta cần kiểm tra xem còn lỗi nào chưa được giải quyết bằng lệnh

`check_design -unresolved`

Nếu không còn lỗi kết quả trả về sẽ như sau:

```
VLSI_05@cds-cadence:work
File Edit View Search Terminal Tabs Help
VLSI_05@cds... × VLSI_05@cds... × VLSI_05@cds... × VLSI_05@cds... ×
=====
|   Id   | Sev | Count| Message Text
|-----|
|CFM-1 |Info |    1|Wrote dofile.
|CFM-5 |Info |    1|Wrote formal verification information.
|PA-7  |Info |    4|Resetting power analysis results.
|      |     |    1|All computed switching activities are removed.
|SYNTH-5|Info |    1|Done mapping.
|SYNTH-7|Info |    1|Incrementally optimizing.
|SYNTH-33|Warning|    1|The selected flow setting will be removed in a
|      |     |    1| future release.
|      |     |    1|Contact Cadence support to understand current flows.
Info   : Done incrementally optimizing. [SYNTH-8]
        : Done incrementally optimizing 'counter_led_7'.
        flow.cptime flow.realtime timing.setup.tns timing.setup.wns snapshot
UM:*
@genus:root: 9> check_design -unresolved
syn_opt

Check Design Report (c)
-----
Unresolved References & Empty Modules
-----
No unresolved references in design 'counter_led_7'

No empty modules in design 'counter_led_7'

Done Checking the design.
@genus:root: 10> █
```

Bước 13: Để xem các báo cáo tương đối về timing, area, power ta chạy lần lượt các lệnh sau:

`report_timing`

```
VLSI_05@cds-cadence:work
File Edit View Search Terminal Tabs Help
VLSI_05@cds-cad... VLSI_05@cds-cad... VLSI_05@cds-cad... VLSI_05@cds-cad...
clock: (R) clk
    Capture      Launch
Clock Edge:+ 10000      0
Src Latency:+ 0          0
Net Latency:+ 0 (I)     0 (I)
Arrival:=   10000      0

Output Delay:- 1000
Required Time:= 9000
Launch Clock:- 0
Data Path:- 364
Slack:= 8636

Exceptions/Constraints:
output_delay      1000      constraints.sdc_line_11_3_1
#-----#
#  Timing Point      Flags Arc Edge Cell      Fanout Load Trans Delay Arrival Instance
#                                         (ff) (ps) (ps) (ps) Location
#-----#
cnt1_counter_reg[3]/CK -      - R (arrival) 4 - 0 0 0 (-,-)
cnt1_counter_reg[3]/Q -      CK->Q F DFFRHQX1 5 8.2 56 79 79 (-,-)
g869_5115/C0 -      A->CO F ADDHX1 1 2.5 20 35 114 (-,-)
g868_1881/Y -      B->Y R NOR2BX1 4 6.8 94 61 175 (-,-)
g865_7098/Y -      A->Y F NAND2X1 4 6.8 95 83 258 (-,-)
g855_4319/Y -      A1->Y R OAI221X1 2 3.5 81 73 331 (-,-)
g851_2398/Y -      B->Y R OR2X1 1 2.1 17 34 364 (-,-)
led_unit[3] <<< - R (port) - - - 0 364 (-,-)
#-----#
@genus:root: 13> □
```

report_area

```
VLSI_05@cds-cadence:work
File Edit View Search Terminal Tabs Help
VLSI_05@cds-cad... VLSI_05@cds-cad... VLSI_05@cds-cad... VLSI_05@cds-cad...
Exceptions/Constraints:
output_delay      1000      constraints.sdc_line_11_3_1
#-----#
#  Timing Point      Flags Arc Edge Cell      Fanout Load Trans Delay Arrival Instance
#                                         (ff) (ps) (ps) (ps) Location
#-----#
cnt1_counter_reg[3]/CK -      - R (arrival) 4 - 0 0 0 (-,-)
cnt1_counter_reg[3]/Q -      CK->Q F DFFRHQX1 5 8.2 56 79 79 (-,-)
g869_5115/C0 -      A->CO F ADDHX1 1 2.5 20 35 114 (-,-)
g868_1881/Y -      B->Y R NOR2BX1 4 6.8 94 61 175 (-,-)
g865_7098/Y -      A->Y F NAND2X1 4 6.8 95 83 258 (-,-)
g855_4319/Y -      A1->Y R OAI221X1 2 3.5 81 73 331 (-,-)
g851_2398/Y -      B->Y R OR2X1 1 2.1 17 34 364 (-,-)
led_unit[3] <<< - R (port) - - - 0 364 (-,-)
#-----#
@genus:root: 13> report_area
=====
Generated by: Genus(TM) Synthesis Solution 23.10-p004_1
Generated on: Dec 23 2024 12:42:49 pm
Module: counter_led_7
Operating conditions: PVT_1P1V_0C
Interconnect mode: global
Area mode: physical library
=====

Instance Module Cell-Count Cell-Area Net-Area Total-Area
=====
counter_led_7 NA        28      17.271   21.689   38.960
=====

@genus:root: 14>
```

report_power

```

VLSI_05@cds-cadence:work
File Edit View Search Terminal Tabs Help
VLSI_05@cds-cadence:work VLSI_05@cds-cadence:work VLSI_05@cds-cadence:work VLSI_05@cds-cadence:work
: -fromGenus : 1
Info : ACTP-0001 Timing initialization started
Info : ACTP-0001 Timing initialization ended
Info : PWRA-0002 [PwrInfo] Skipping activity propagation due to -skip_ap
: option...
Warning: PWRA-0302 [PwrWarn] Frequency scaling is not applicable for vectorless
: flow. Ignoring frequency scaling.
Warning: PWRA-0304 [PwrWarn] -stim option is not applicable with vectorless mode
: of power analysis, ignored this option.
Info : PWRA-0002 Started 'vectorless' power computation.
Info : PWRA-0002 Finished power computation.
Info : PWRA-0007 [PwrInfo] Completed successfully.
: Info=6, Warn=2, Error=0, Fatal=0
Instance: /counter_led_7
Power Unit: W
PDB Frames: /stim#0/frame#0
-----  

Category Leakage Internal Switching Total Row%
-----  

memory 0.00000e+00 0.00000e+00 0.00000e+00 0.00000e+00 0.00%  

register 1.59423e-09 1.76575e-06 3.49206e-07 2.11655e-06 50.78%  

latch 0.00000e+00 0.00000e+00 0.00000e+00 0.00000e+00 0.00%  

logic 2.78378e-09 5.97380e-07 1.30602e-06 1.90618e-06 45.73%  

bbox 0.00000e+00 0.00000e+00 0.00000e+00 0.00000e+00 0.00%  

clock 0.00000e+00 0.00000e+00 1.45200e-07 1.45200e-07 3.48%  

pad 0.00000e+00 0.00000e+00 0.00000e+00 0.00000e+00 0.00%  

pm 0.00000e+00 0.00000e+00 0.00000e+00 0.00000e+00 0.00%  

-----  

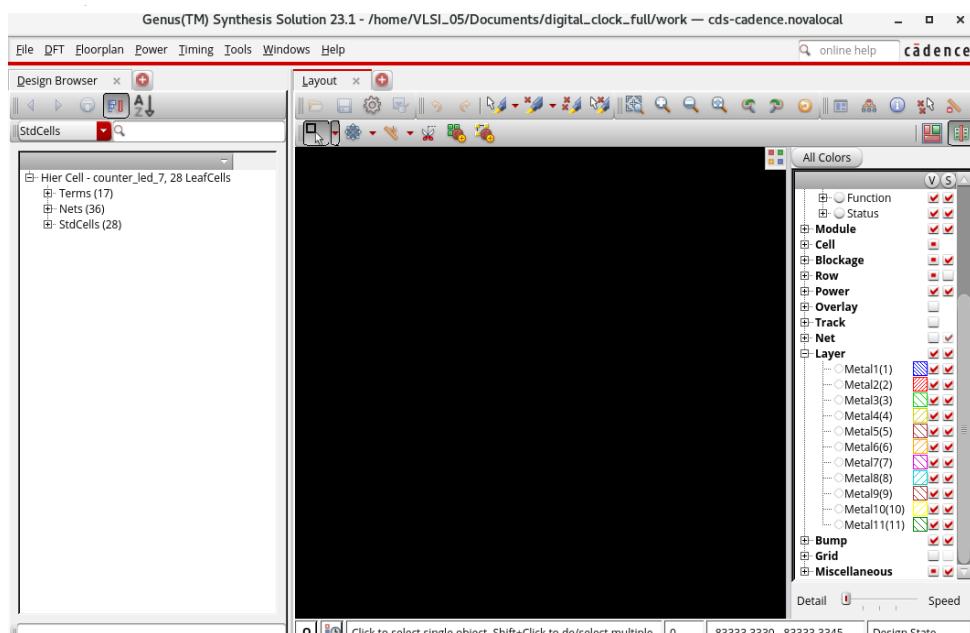
Subtotal 4.37801e-09 2.36313e-06 1.80043e-06 4.16793e-06 99.99%  

Percentage 0.11% 56.70% 43.20% 100.00% 100.00%
-----  

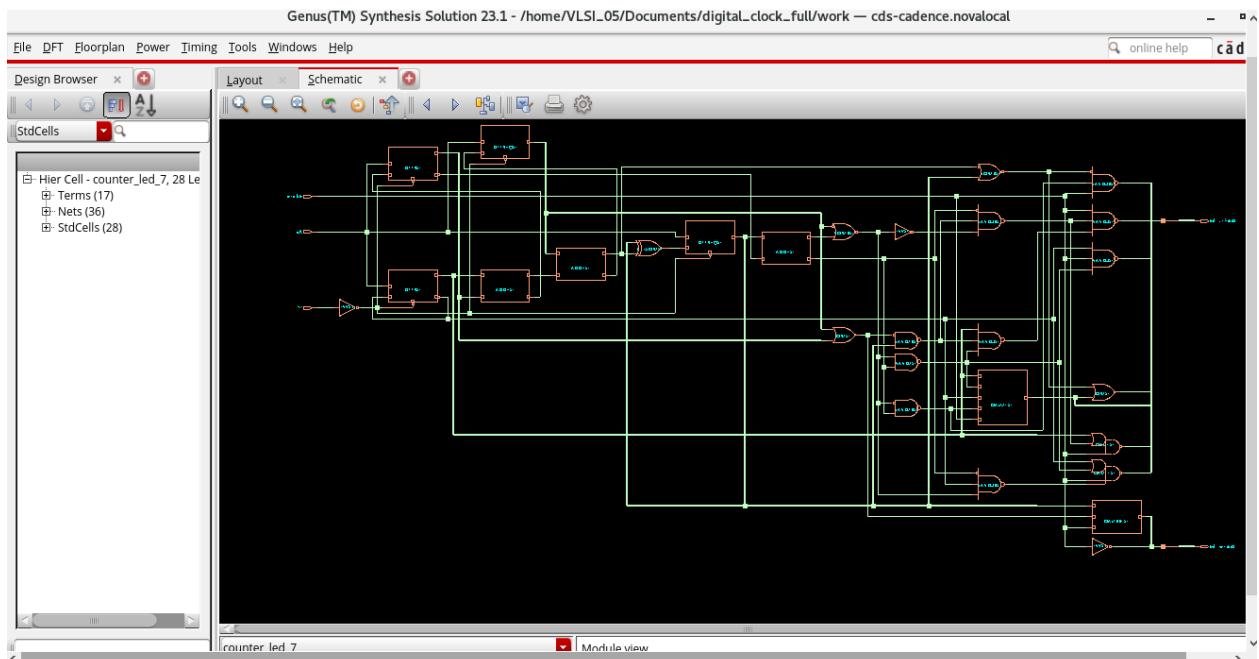
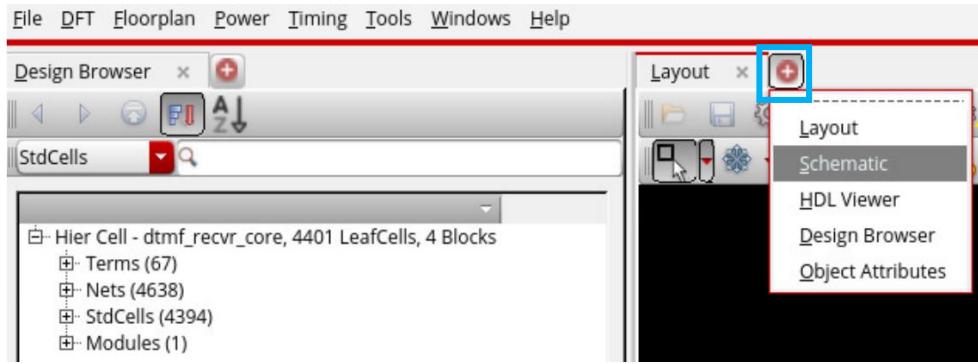
@genus:root: 15>

```

Bước 14: Sau khi xong ta muốn xem bản schematic thì ta sẽ sử dụng lệnh `gui_show`



Để hiện Schematic ta cần click vào icon + như sau để chọn Schematic



Bước 15: Khi đã check xong ta xuất các file trạng thái thiết kế ở dạng nhị phân (.db), netlist (.v), constraints (.sdc)(sau synthesis),

```
write_db $DESIGN -to_file ${DESIGN}.db
write_netlist > ${DESIGN}_syn.v
write_sdc > ${DESIGN}_syn.sdc
```

Các file trên sẽ được lưu trong folder work.

Kết thúc synthesis

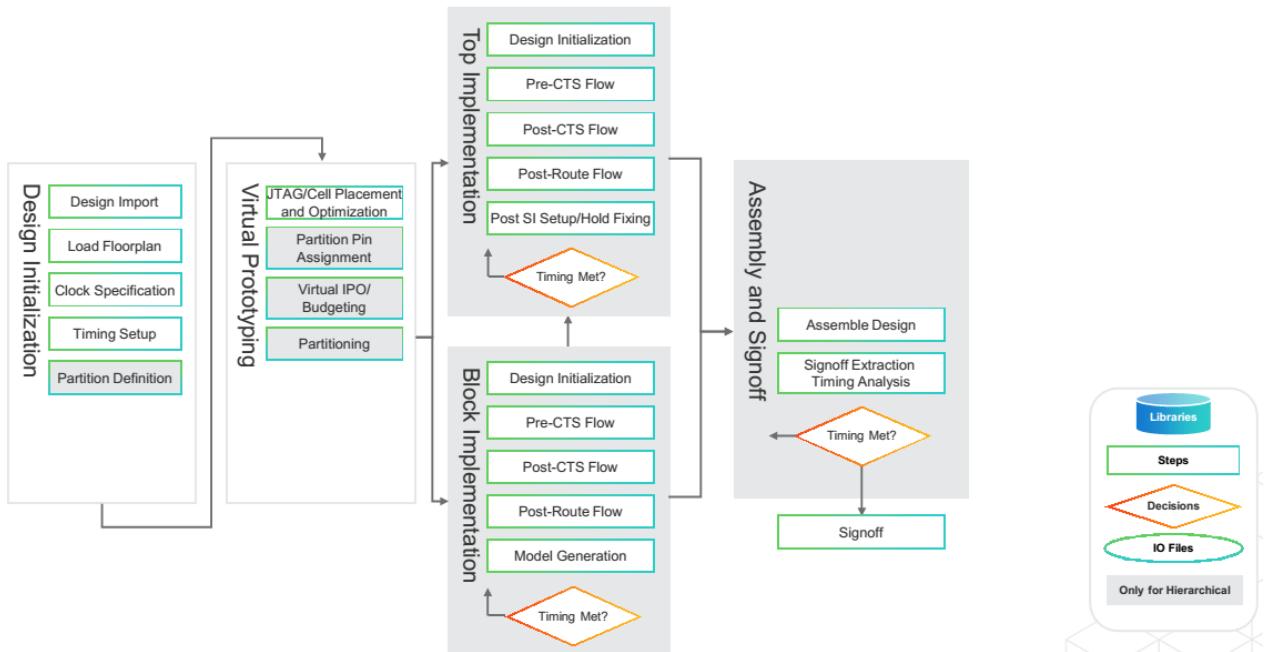
VI. Physical design (sử dụng Innovus)

Đối với Innovus ta có 2 cách chạy chương trình: chạy bằng batch và chạy trên interface.

Mặc dù có 2 cách sử dụng như vậy nhưng ta có thể dùng 2 cách song song nhau

1. Quy trình sử dụng Innovus:

Flowchart: Hierarchical Implementation



Để thực hiện layout bằng Innovus ta cần 3 file cơ bản:

- File netlist đã được tạo từ bước Synthesis trước (.v)
- Thư viện LEF (.lef)
- File mmmc (.tcl)

Riêng 2 đầu tiên ta đã có ở các bước trước đó nhưng còn file .mmmc ta cần phải tự tạo trong folder work

Bước 1: cd đến folder tcl

```
$ cd path/to/tcl
```

Bước 2: tạo file mmmc

```
$ gedit <tên_project>.tcl
```

Lưu ý về cấu trúc file mmmc:

- Tạo thư viện cho 2 trường hợp best và worst (không cần chỉnh đường dẫn)

```
create_library_set -name libset_worst \
    -timing \
    [list
/home/Cadence/PDK/gsclib045_all_v4.4/gsclib045/timing/slow_vdd1v0_basicCells
.lib] \
    -si \
    [list /home/Cadence/PDK/gsclib045_all_v4.4/gsclib045/celtic/slow.cdb]
```

```
create_library_set -name libset_best \
    -timing \
    [list
/home/Cadence/PDK/gsclib045_alzl_v4.4/gsclib045/timing/fast_vdd1v0_basicCell
s.lib] \
    -si \
    [list /home/Cadence/PDK/gsclib045_all_v4.4/gsclib045/celtic/fast.cdb]
```

- Tạo Time Condition cho 2 trường hợp worst và best case

```
create_timing_condition -name timing_condition_worst \
    -library_sets [list libset_worst]
```

```
create_timing_condition -name timing_condition_best \
    -library_sets [list libset_best]
```

- Tạo các thông số cho RC corner:

VD:

```
create_rc_corner -name rc_corner_default \
    -pre_route_res 1.0 \
    -post_route_res 1.2 \
    -pre_route_cap 1.0 \
    -post_route_cap 1.3 \
    -post_route_cross_cap 0.8 \
    -pre_route_clock_res 0.5 \
    -pre_route_clock_cap 0.6 \
    -qrc_tech
/home/Cadence/PDK/gsclib045_all_v4.4/gsclib045/qrc/qx/gpdk045.tch
```

- Tạo delay corner cho 2 trường hợp worst và best case:

VD:

```
create_delay_corner -name delay_corner_worst \
    -timing_condition {timing_condition_worst} \
    -rc_corner rc_corner_default

create_delay_corner -name delay_corner_best \
    -timing_condition {timing_condition_best} \
    -rc_corner rc_corner_default
```

- Tạo các ràng buộc (sử dụng file constraints được tạo sau khi synthesis)

```
create_constraint_mode -name constraint_mode_default \
-sdc_files [list /path/to/constraints/file]
```

- Tạo các Analysis view từ các constraint_mode và các delay_corner cho 2 trường hợp worst và best case

VD:

```
create_analysis_view -name analysis_view_setup \
-constraint_mode constraint_mode_default \
-delay_corner delay_corner_worst
```

```
create_analysis_view -name analysis_view_hold \
-constraint_mode constraint_mode_default \
-delay_corner delay_corner_best
```

- Thiết lập cho Analysis Views:

```
set_analysis_view -setup [list analysis_view_setup] \
-hold [list analysis_view_hold]
```

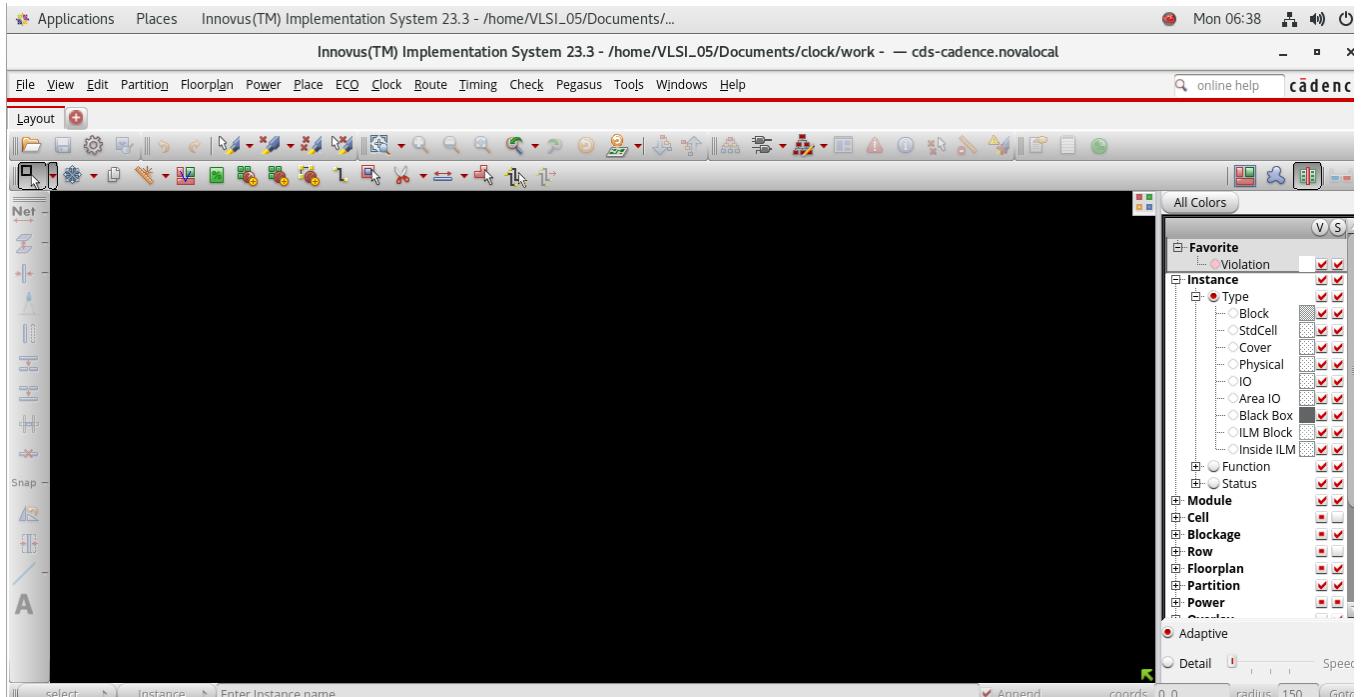
Sau khi hoàn thành thì lưu file và chuyển đến các bước tiếp theo.

2. Chạy Innovus

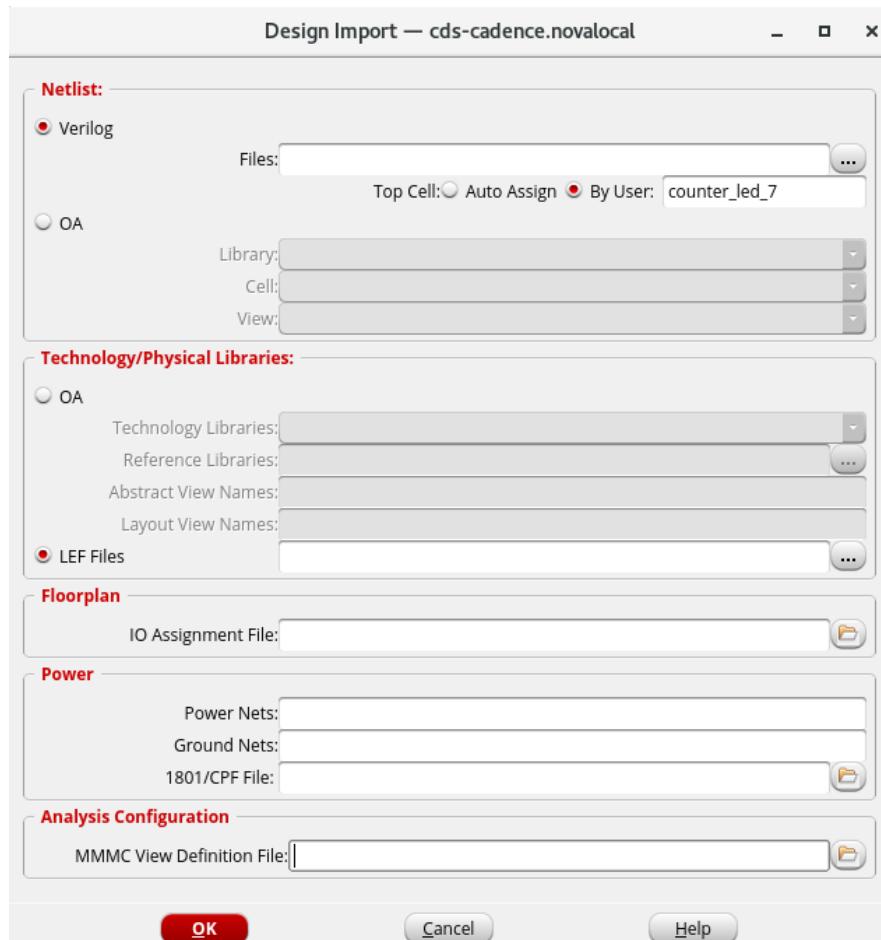
Bước 1: Khởi động innovus:

```
$ innovus -stylus
```

Sau khi chạy lệnh, cửa sổ innovus sẽ mở ra:

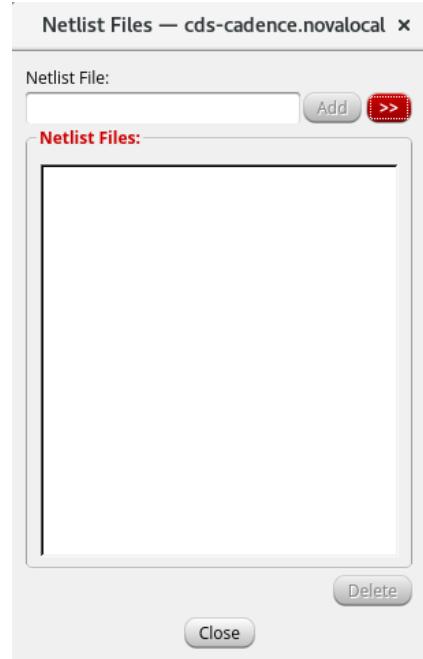


Bước 2: Để bắt đầu công việc layout ta cần import design của mình vào bằng cách vào **File > Import Design** và ta sẽ được cửa sổ Import Design:

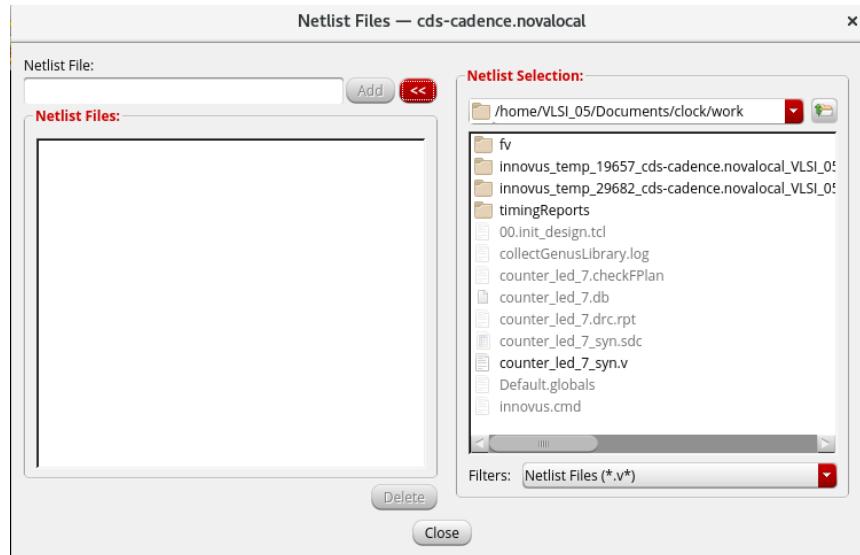


Bước 3: Nhập các file vào form:

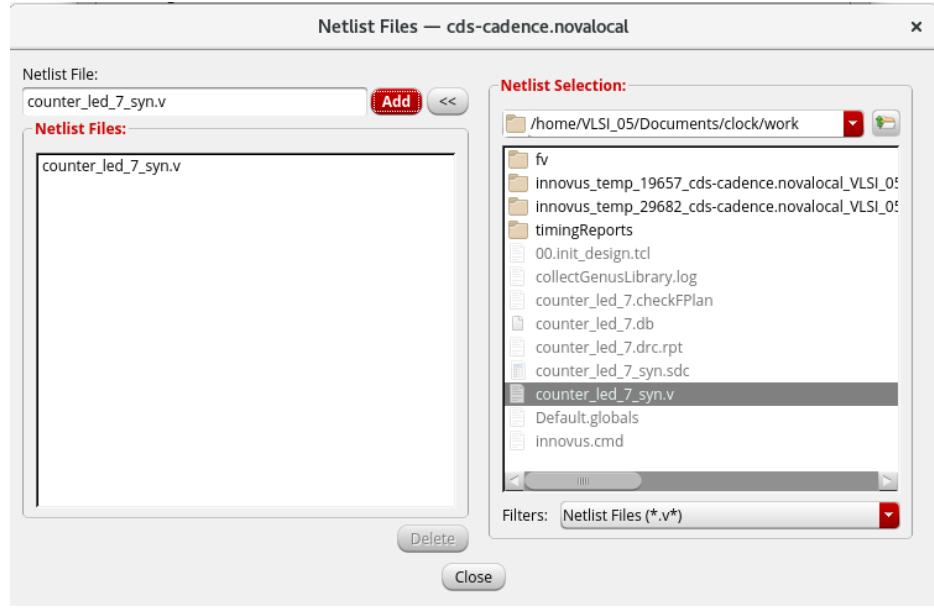
1. Netlist: click vào icon ... để chọn file, theo các bước trước ta sẽ tìm nó ở ngay trong folder work với tên là file <tên_project>_syn.v



Click vào icon để hiện ra cửa sổ Add các file



Chọn 1 file và click Add để thêm vào mục Netlist Files:

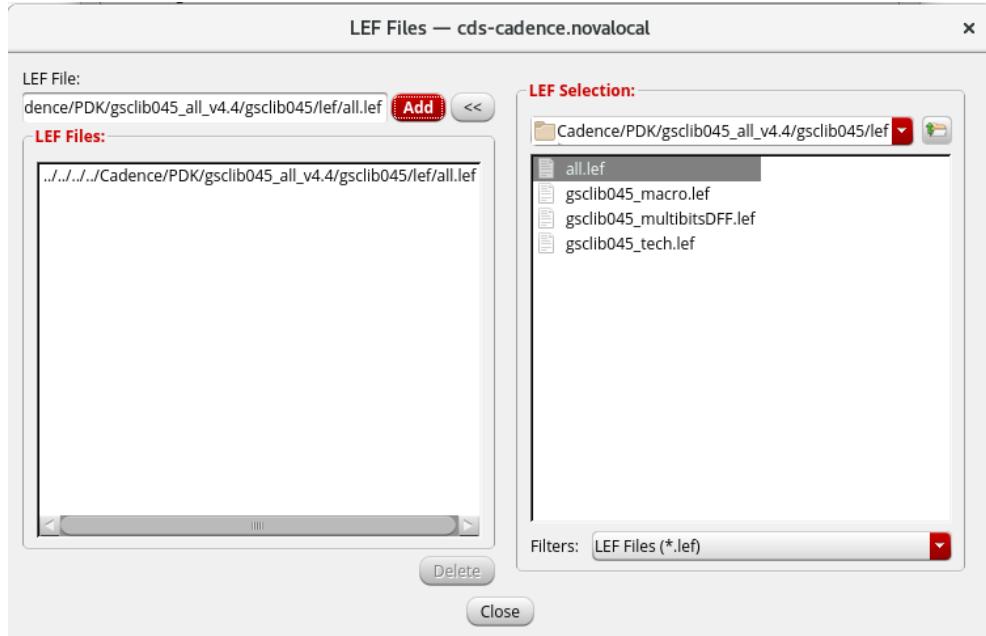


Cuối cùng là tắt của sổ Netlist File là xong bước này.

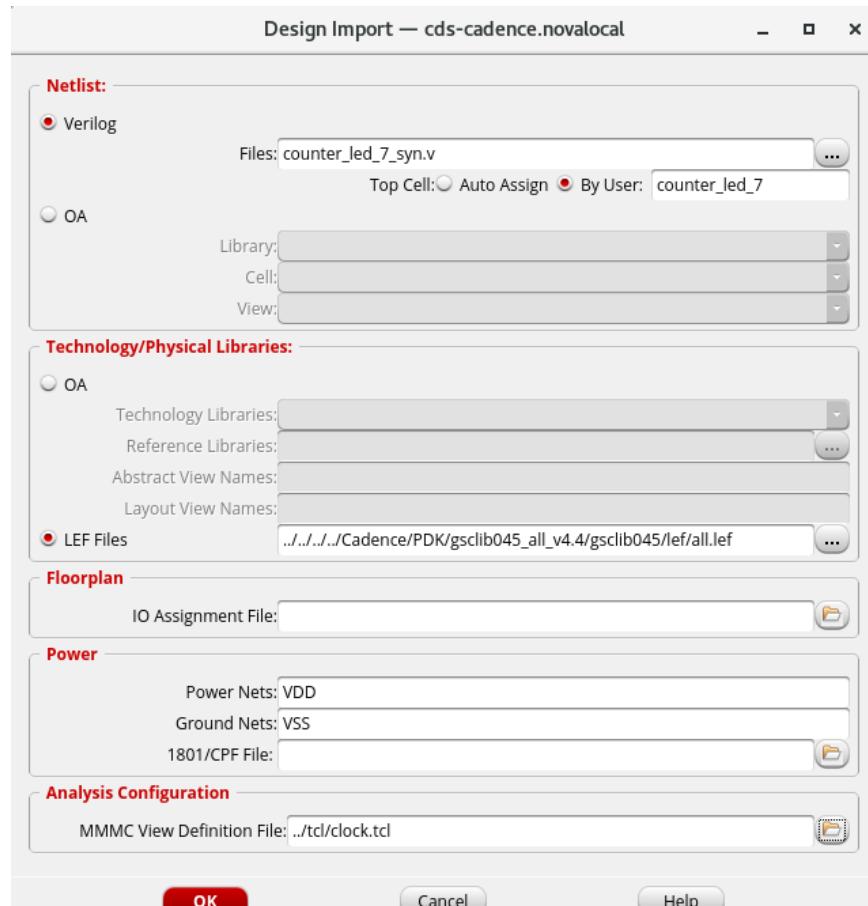
2. Technology/Physical Libraries

Đầu tiên ta chọn mục LEF và thư viện sẽ nằm ở đường path này
`/home/Cadence/PDK/gsclib045_all_v4.4/gsclib045/lef/all.lef`

Vì vậy, ta cần mở cửa sổ như ở bước trên nhưng do file này không ở trong cùng folder work mà ta đang làm nên ta phải click vào icon để di chuyển tới folder mà ta cần truy cập và nó sẽ hiện ra như sau:

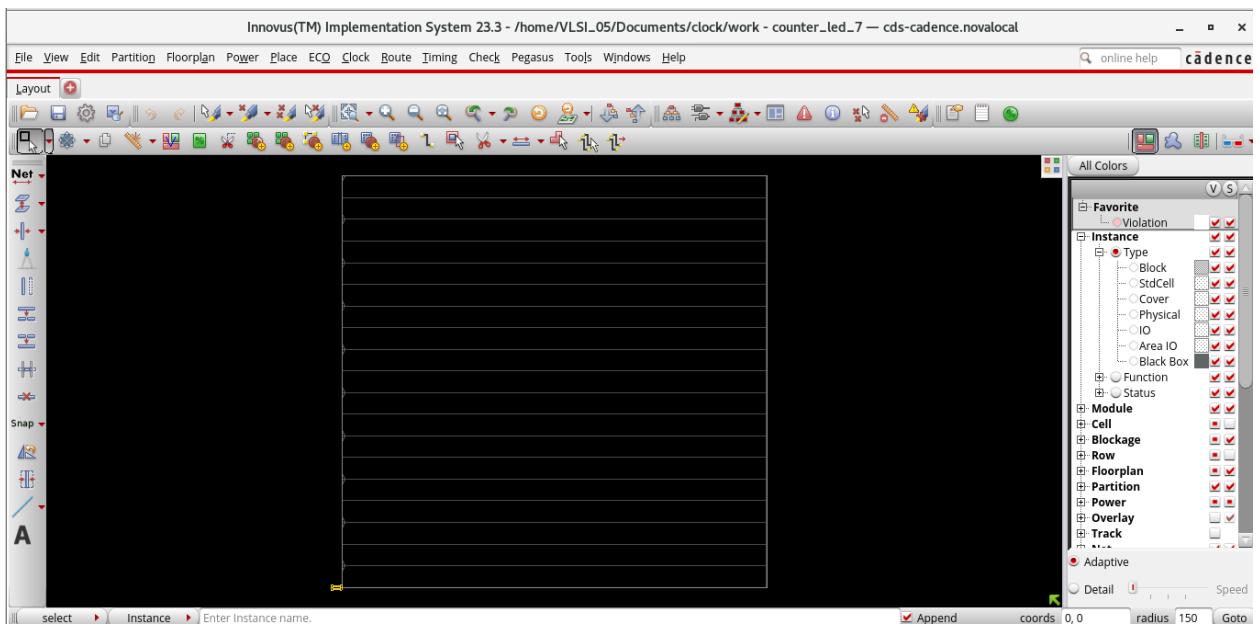


3. Analysis Configuration: Ta thực hiện các bước tương tự để nhập file mmmc ta tạo ở phần trước và ta sẽ được Form hoàn thiện sau:

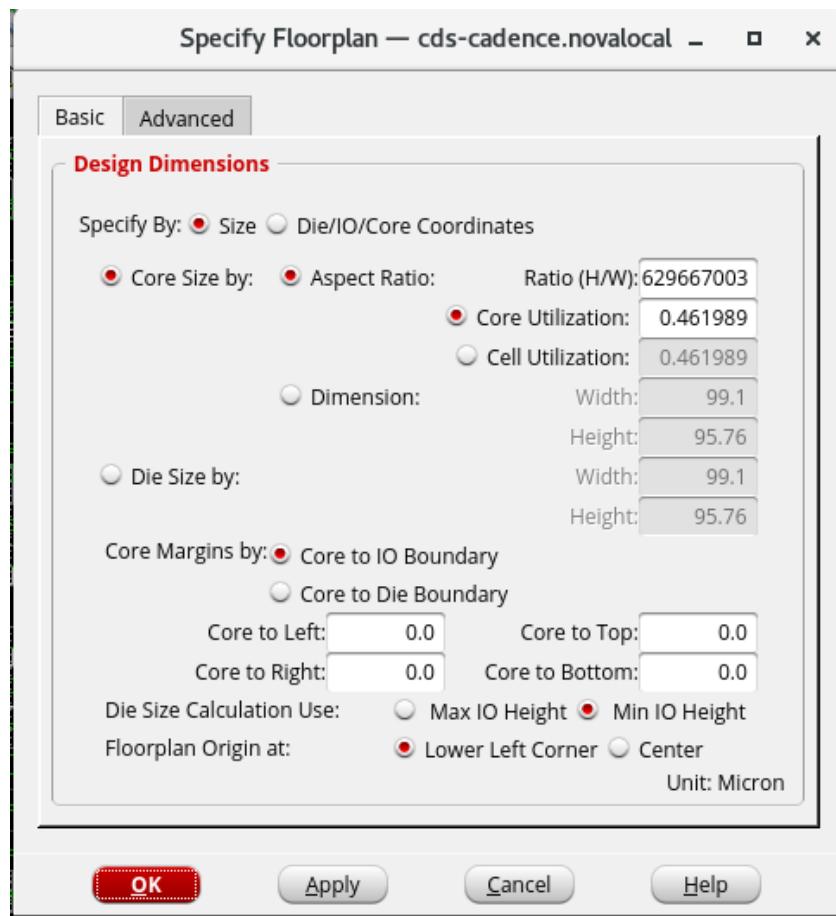


Cuối cùng là click **OK**.

Sau khi tạo xong thì ta được:



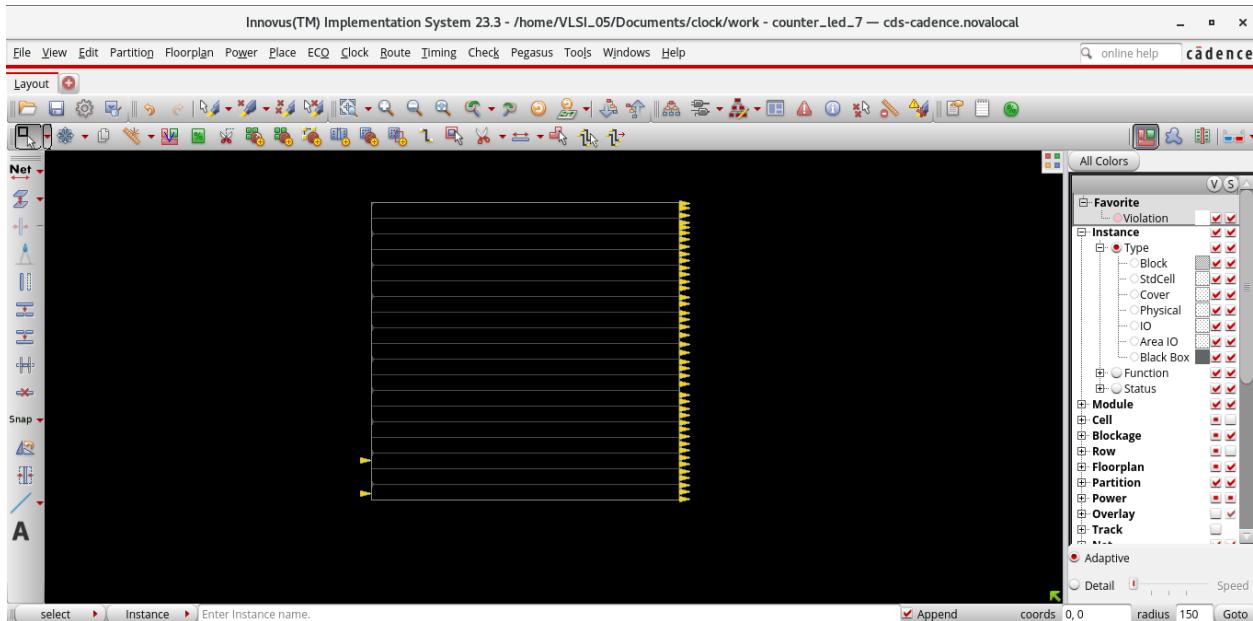
Bước 4: Floorplan, ta cần điều chỉnh thông số về Size, Die/IO/Core Coordinate cho khối ta đang làm việc. Chọn **Floorplan > Specify Floorplan ...** và ta được form sau:



Bước 5: Tiếp theo là ta cần placing pin bằng cách sử dụng icon  trên thanh công cụ và di chuyển các pin đến vị trí mong muốn



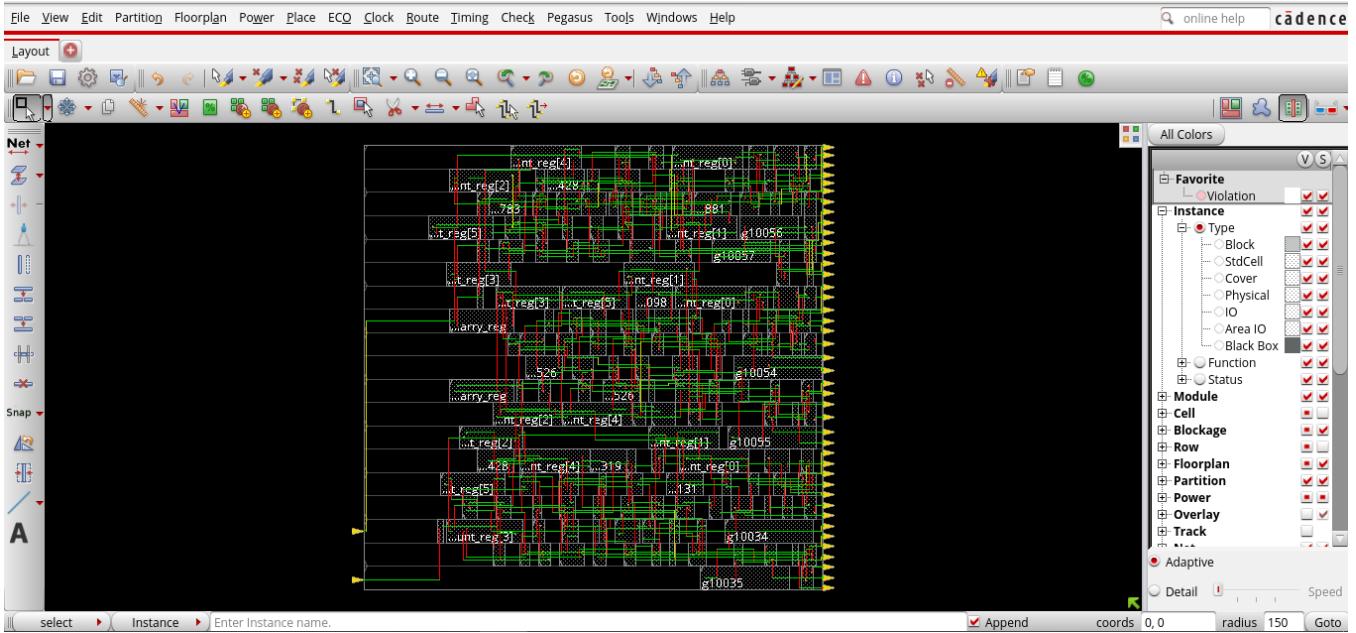
Sau khi placing xong ta được và ta chọn tất cả các pin nó sẽ hiện lên như sau:



Bước 6: Place and Route ta placing các cell vào trong khối chính bằng cách sử dụng lệnh trên terminal:

[place_design](#)

Ta được:



Bước 7: Để tối ưu hóa lại thiết kế ta sử dụng lệnh sau:

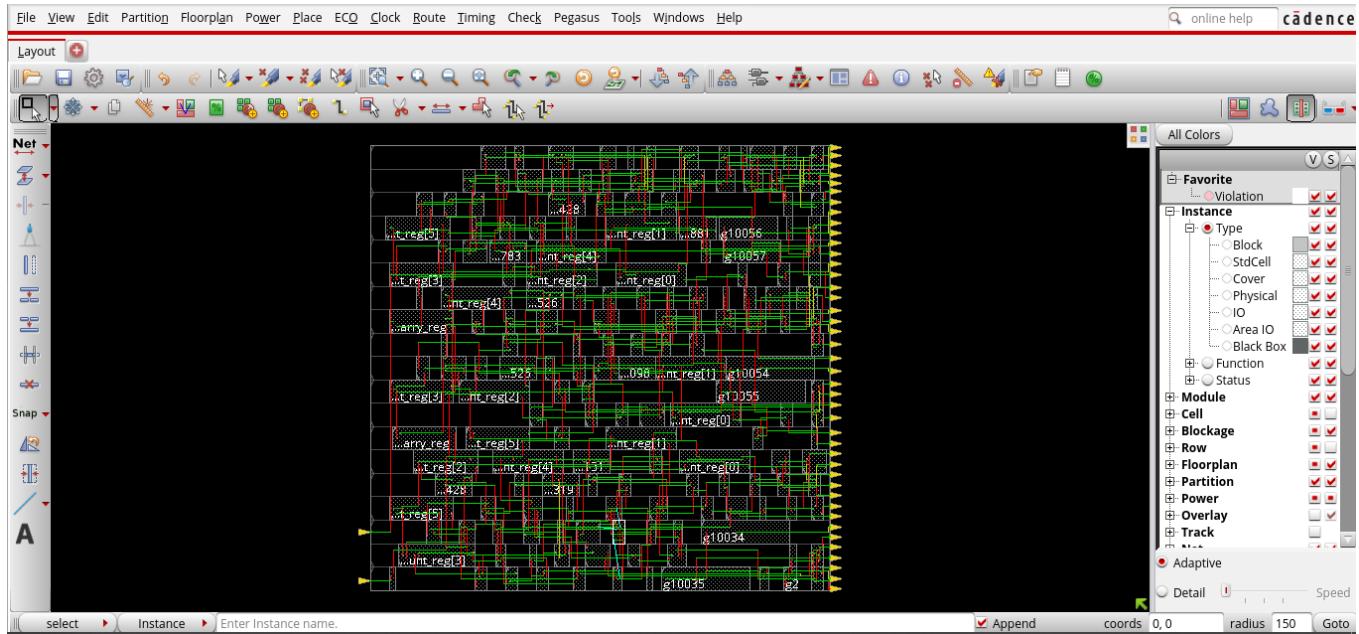
```
set_db place_design_floorplan_mode false
place_opt_design
```

```
VLSI_05@cds-cadence:~/Documents/clock/work
```

File Edit View Search Terminal Tabs Help

```
VLSI_05@cds-caden... VLSI_05@cds-caden... VLSI_05@cds-caden...
[2] led_unit_hour[1] led_unit_hour[0] led_ten_hour[6] led_ten_hour[5] led_ten_hour[4] led_ten_hour[3] led_ten_hour[2] led_ten_hour[1] led_ten_hour[0] led_unit_min[0] led_unit_min[1] led_unit_min[2] led_unit_min[3] led_unit_min[4] led_unit_min[5] led_unit_min[6] led_ten_sec[0] led_ten_sec[1] led_unit_sec[1] led_ten_sec[2] led_ten_sec[3] led_ten_sec[4] led_ten_sec[5] led_ten_sec[6] led_ten_sec[3] led_ten_sec[2] led_ten_sec[0] led_unit_sec[5] led_unit_sec[4] led_unit_sec[3] led_unit_sec[6] led_ten_hour[0] led_ten_hour[1] led_ten_hour[2] led_ten_hour[3] led_ten_hour[4] led_ten_hour[5] led_ten_hour[6] led_unit_hour[0] led_unit_hour[1] led_unit_hour[2] led_unit_hour[3] led_unit_hour[4] led_unit_hour[5] led_unit_hour[6] led_ten_min[1] led_ten_min[2] led_ten_min[3] led_ten_min[4] led_ten_min[5] led_ten_min[6] rst clk
@innovus 191> set_db edit_wire_shape iowire
@innovus 192> set_db edit_wire_layer_horizontal M3 ; set_db edit_wire_width_horizontal 0.280
@innovus 193> set_db edit_wire_layer_vertical M3 ; set_db edit_wire_width_vertical 0.280
@innovus 194> gui_select -point {167.62000 5.03600}
@innovus 195>
@innovus 195> set_db place_design_floorplan_mode false
1 false
@innovus 196> place_opt_design
```

Đợi 1 lúc và ta được:

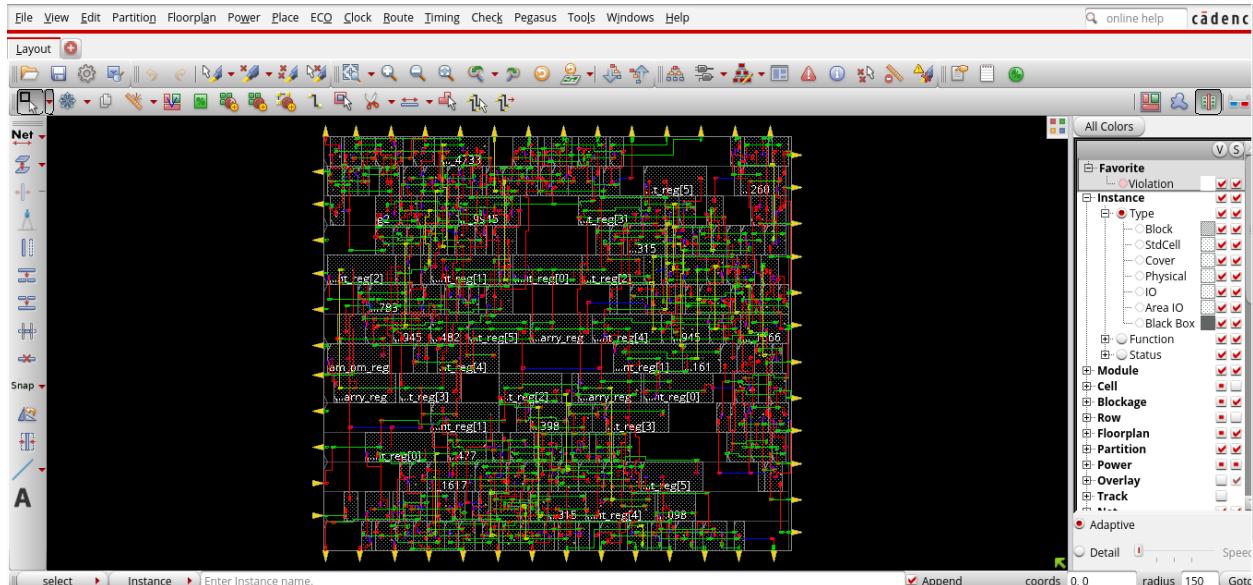


Bước 8: Vì việc đi dây sau bước place_design là nó đi dây global để ước tính các thông số. Nên ở bước này ta cần đi dây bằng lệnh:

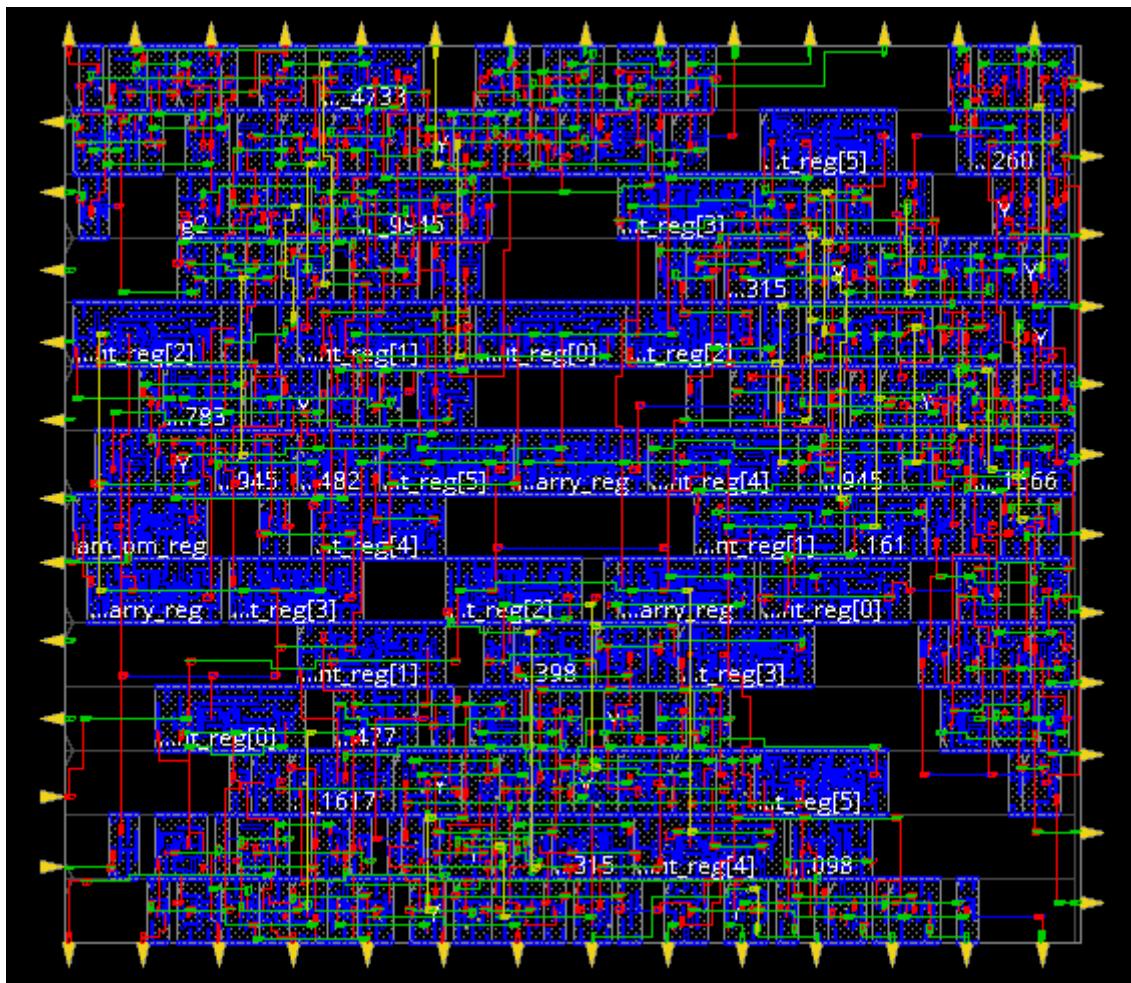
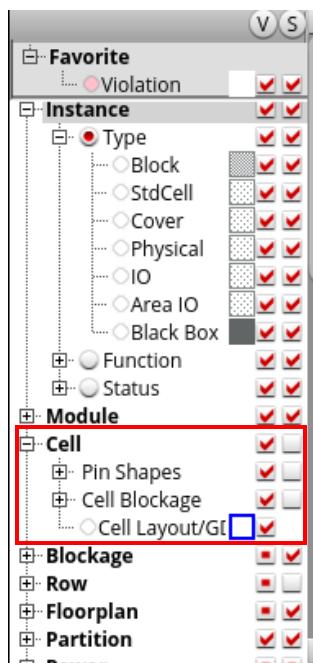
[route_design](#)

Và tiếp theo cần tối ưu hóa nó bằng

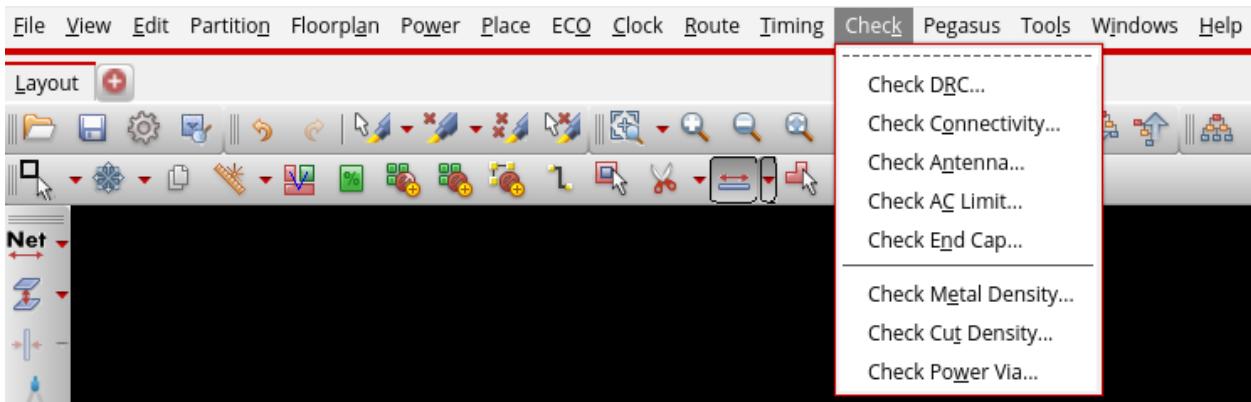
[route_opt_design](#)



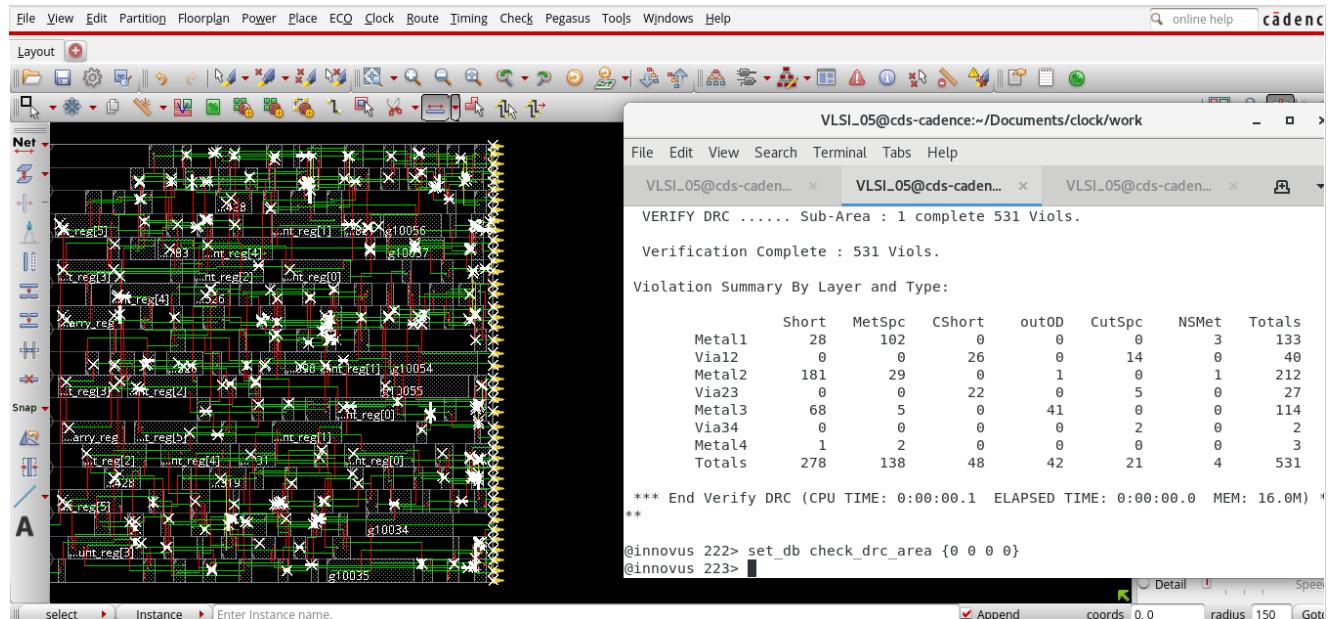
Và nếu cần hiển thị các đường đi dây từ Cell standard chọn checkbox V của mục cell



Bước 9: Để xem bản layout của mình đã đúng so với Design Rules chưa thì ta chạy DRC. Chọn **Check > Check DRC**

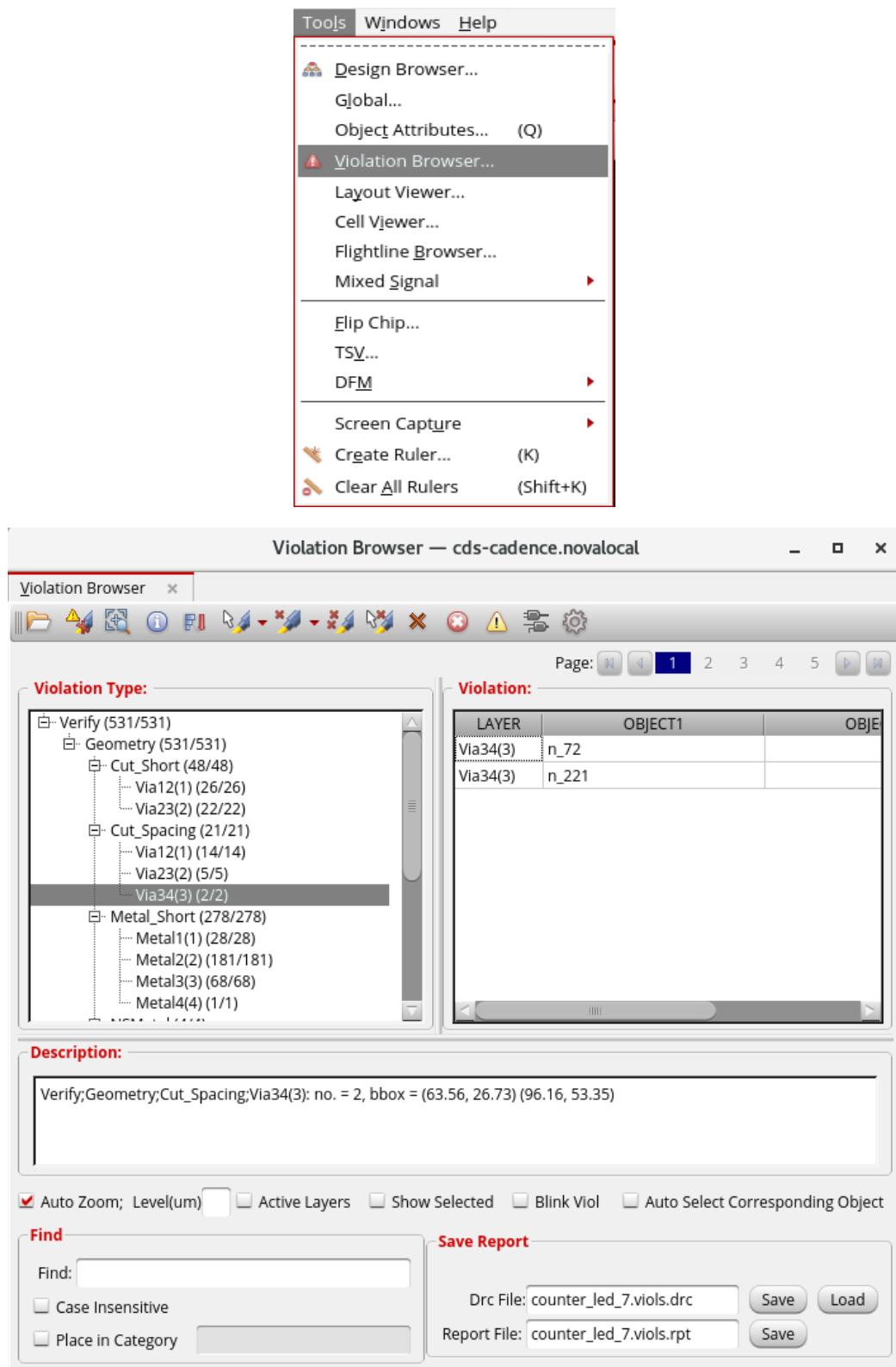


Đây là kết quả của DRC sau khi xem kiểm tra lỗi:



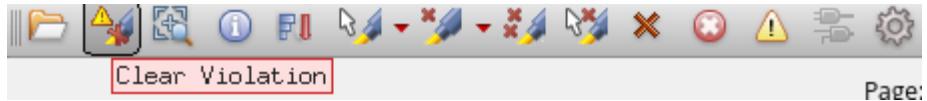
Nếu không lỗi thì nó sẽ không có dấu x nào

Bước 10: Xác định các lỗi. chọn **Tool > Violation Browser**.



Nếu không có lỗi nào thì nó sẽ không hiển thị gì

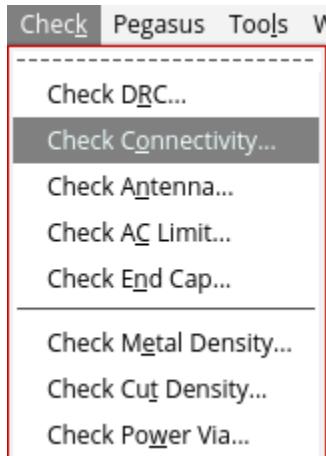
Xử lý từng violation, khi xử lý xong ta có thể xóa bằng cách chọn icon **Clear Violation**



Page:

Xử lý xong tất cả các lỗi ta đến bước tiếp theo.

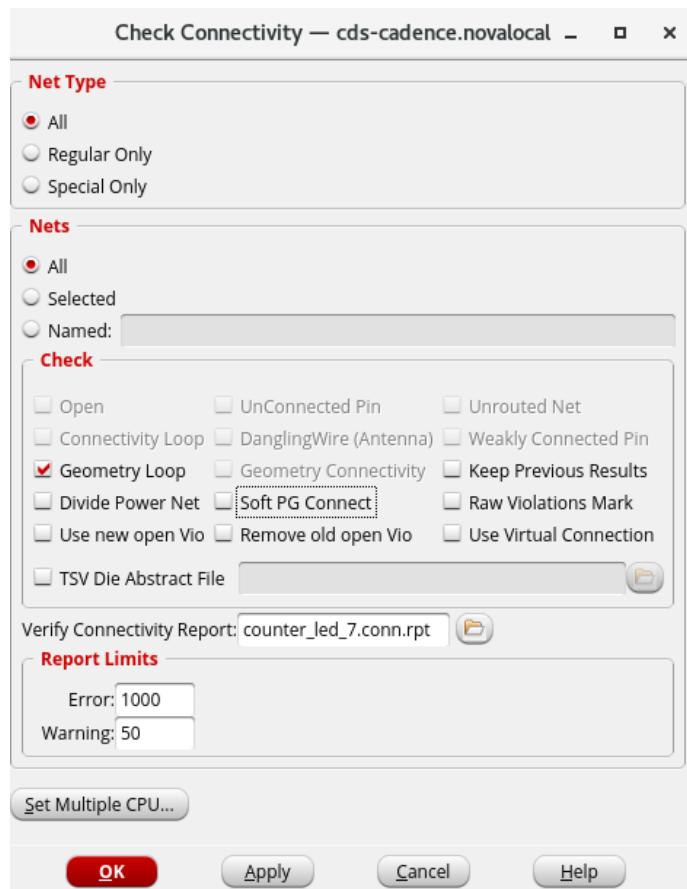
Bước 11: Check connectivity, **Check > Check Connectivity**.



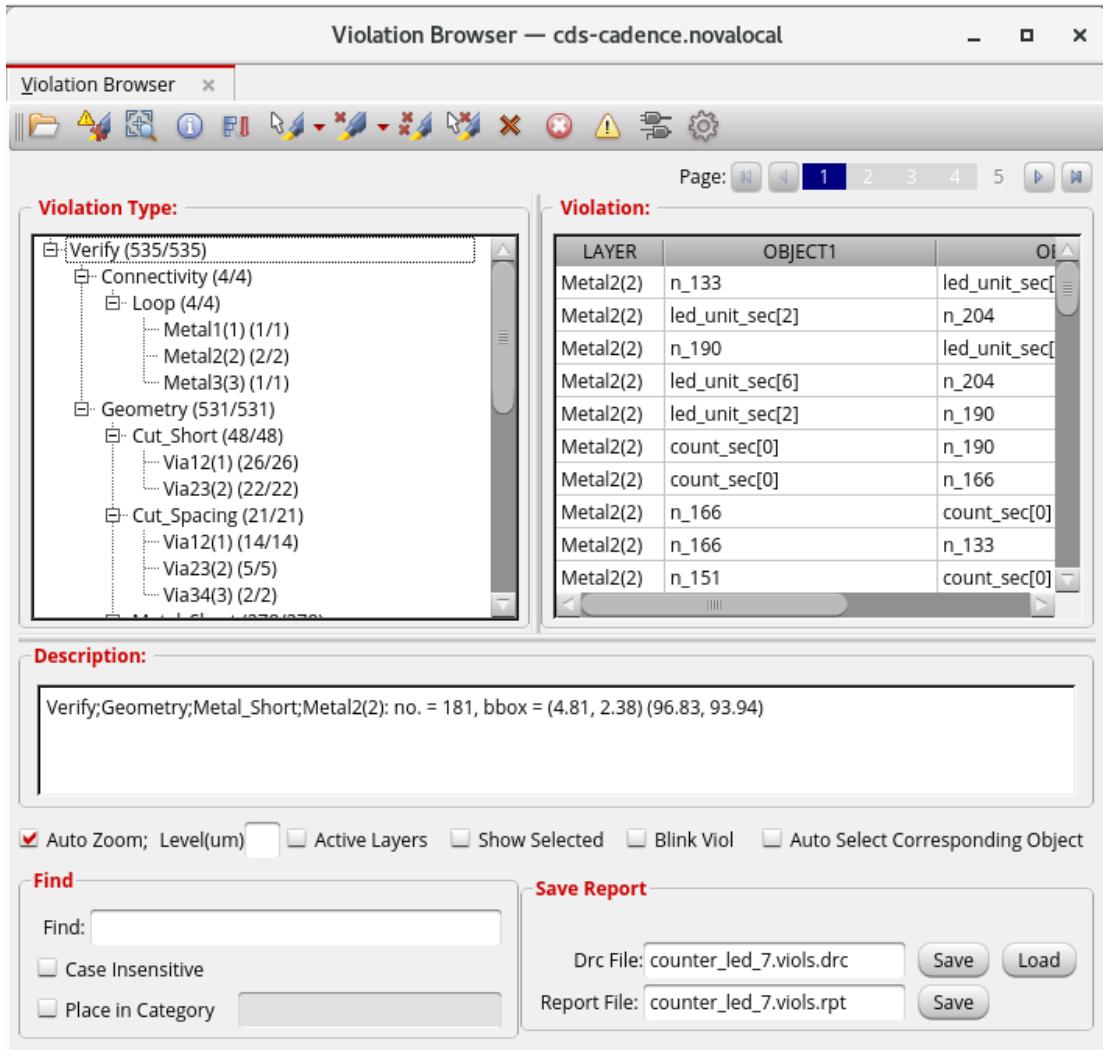
Bước 12: Hãy đảm bảo rằng bạn ở Physical View.



Sau đó ở form **Check Connectivity**, bỏ chọn tất cả các checkbox khác ngoại trừ **Geometry Loop**, và cuối cùng click **OK**.



Bước 13: Mở Violation Browser.



Xử lý hết các lỗi rồi chuyển đến bước tiếp theo.

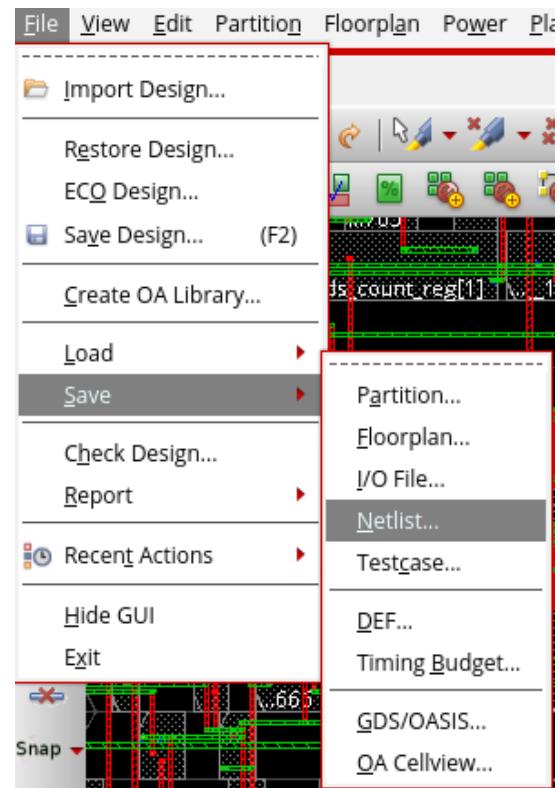
Bước 14: Để biết được thiết kế đã đúng với spec chưa ta cần kiểm tra 3 thông số về delay, area và power chạy lần lượt 3 lệnh:

`report_timing`

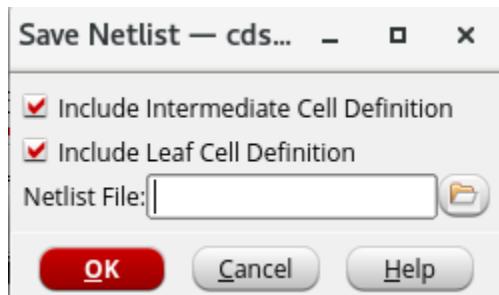
`report_area`

`report_power`

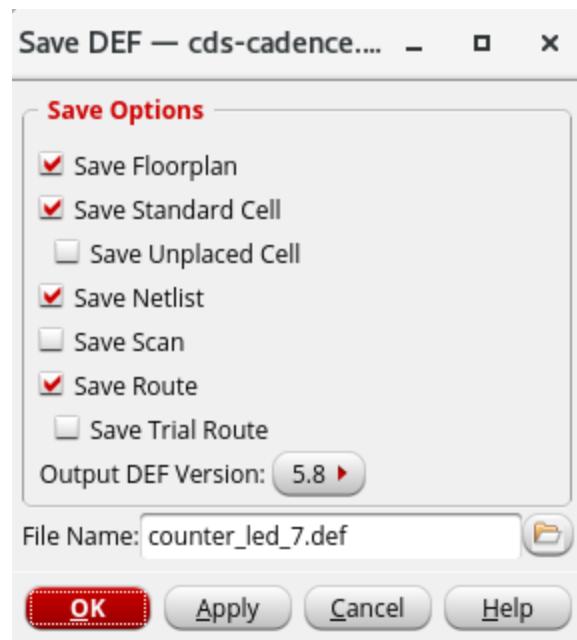
Bước 15: Xuất file layout ra dạng netlist. Chọn **File > Save > Netlist**.



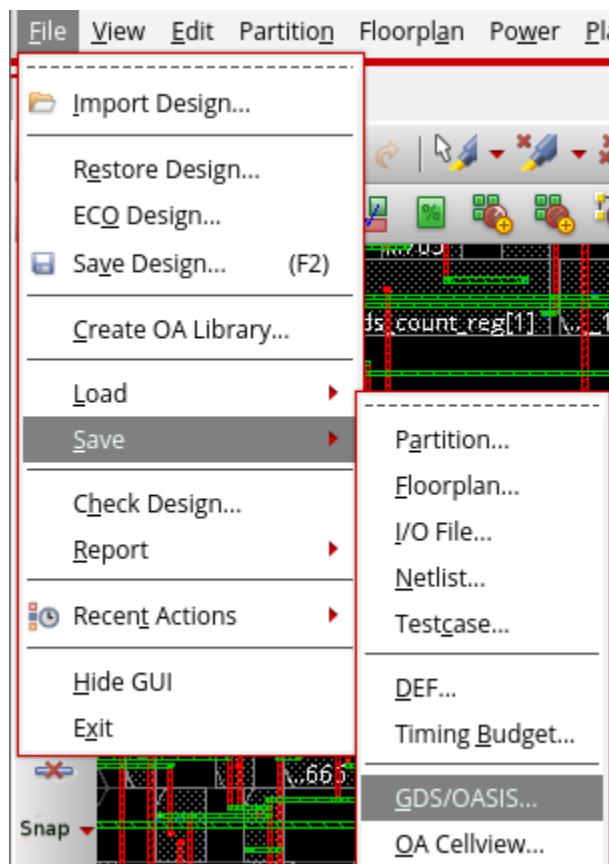
Sau đó, điền tên của project.



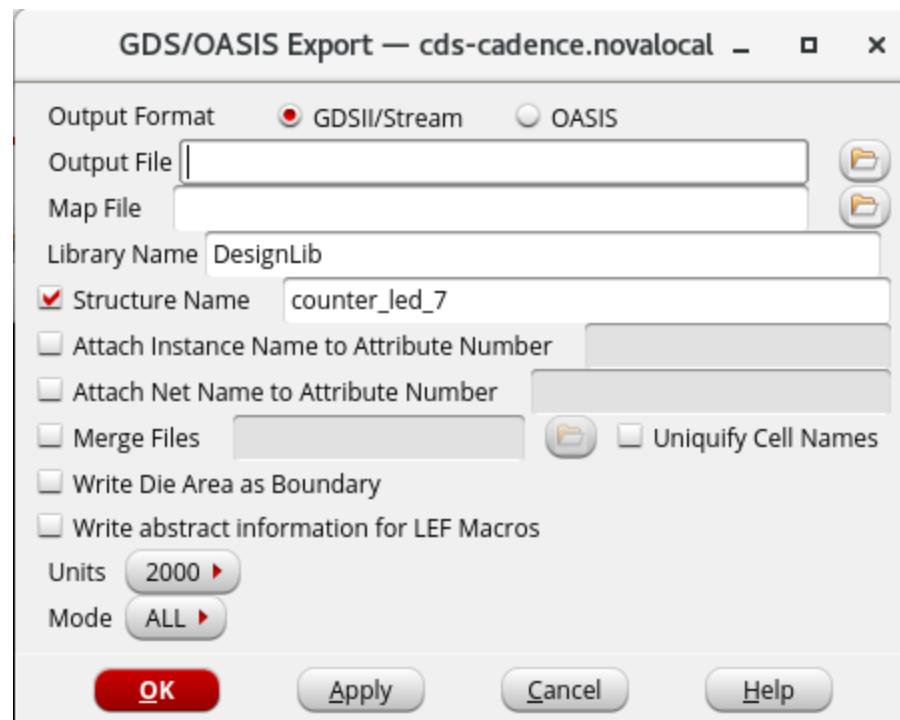
Bước 16: Lưu file DEF. Chọn File > Save > DEF



Bước 17: Lưu thành file GDSII. Chọn File > Save > GDS/OASIS



Điền tên vào Output File bằng cách click vào icon để chọn folder lưu và tên cho file GDS cần kết thúc bằng .gds hoặc.gds.gz để nén và chuyển tiếp cho khâu sau.



Kết thúc innovus

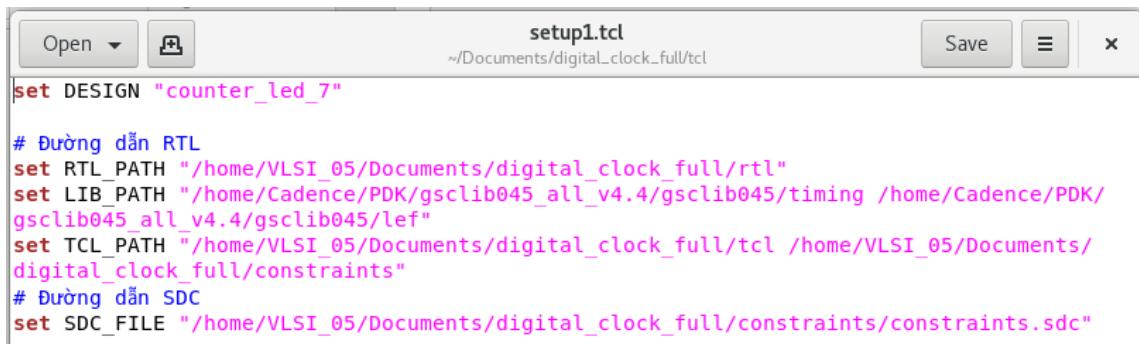
VII. Signoff (sử dụng Tempus, Voltus, Pegasus)

Phụ lục

Những lưu ý khi sử dụng database:

1. File setup.tcl

Các biến đường dẫn (_PATH) cần phải đưa về đúng địa chỉ lưu thư mục cần chỉ định



```
setup1.tcl
~/Documents/digital_clock_full/tcl

set DESIGN "counter_led_7"

# Đường dẫn RTL
set RTL_PATH "/home/VLSI_05/Documents/digital_clock_full/rtl"
set LIB_PATH "/home/Cadence/PDK/gsclib045_all_v4.4/gsclib045/timing /home/Cadence/PDK/
gsclib045_all_v4.4/gsclib045/lef"
set TCL_PATH "/home/VLSI_05/Documents/digital_clock_full/tcl /home/VLSI_05/Documents/
digital_clock_full/constraints"
# Đường dẫn SDC
set SDC_FILE "/home/VLSI_05/Documents/digital_clock_full/constraints/constraints.sdc"
```

2. Đọc và chỉnh file .sdc

Các file .sdc để mở ra đọc thì cần cd đến path của thư mục chứa file .sdc đó
Rồi \$ gedit <tên_file>.sdc

Thì ta mới đọc và chỉnh sửa được file .sdc

3. Cấu trúc thư mục

