Advanced VLSI Design (Module 24151)

Course and Contest – Phase 5

Prof. Dirk Timmermann, Christoph Niemann,
Jakob Heller, Franz Plocksties, Hannes Raddatz

Outline

1. Presentations of synthesized designs and results for ST65
2. Introduction of design flow
   ↳ Final chip layout
3. Task and further information for phase 5

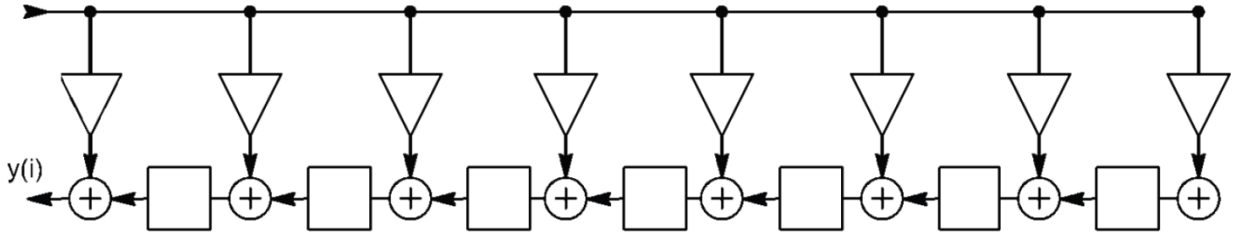
Presentation of synthesized designs

		Metric	Ranking
1.	Kapikad, Rakshan Premsagar	$4.85*10^9$	2.
2.	Björner, Mathis	$8.79*10^9$	1.
3.	Koch, Christian	----------	--

Consider: 5 minutes per presentation

Introduction to Cadence Innovus Digital Implementation

Design–Flow

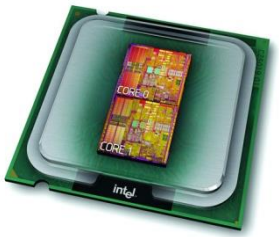


Hardware Language ✓ *Textual description of the circuit*

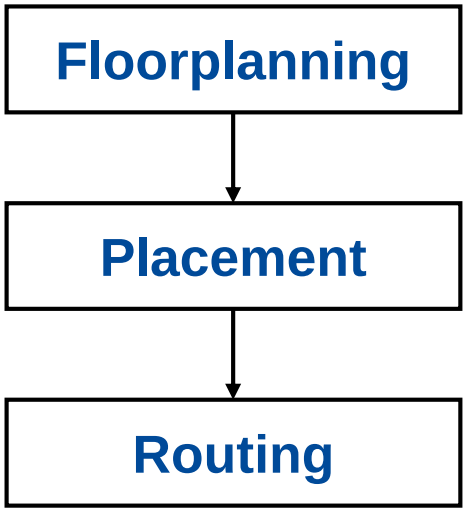
Synthesis ✓ *Transfer to logic gates*

Layout

Production



Layout–Flow I

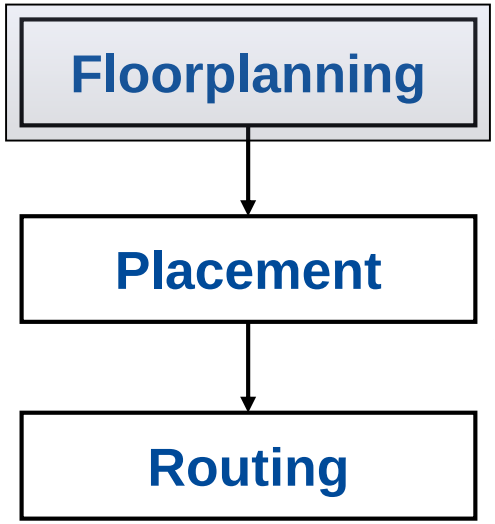


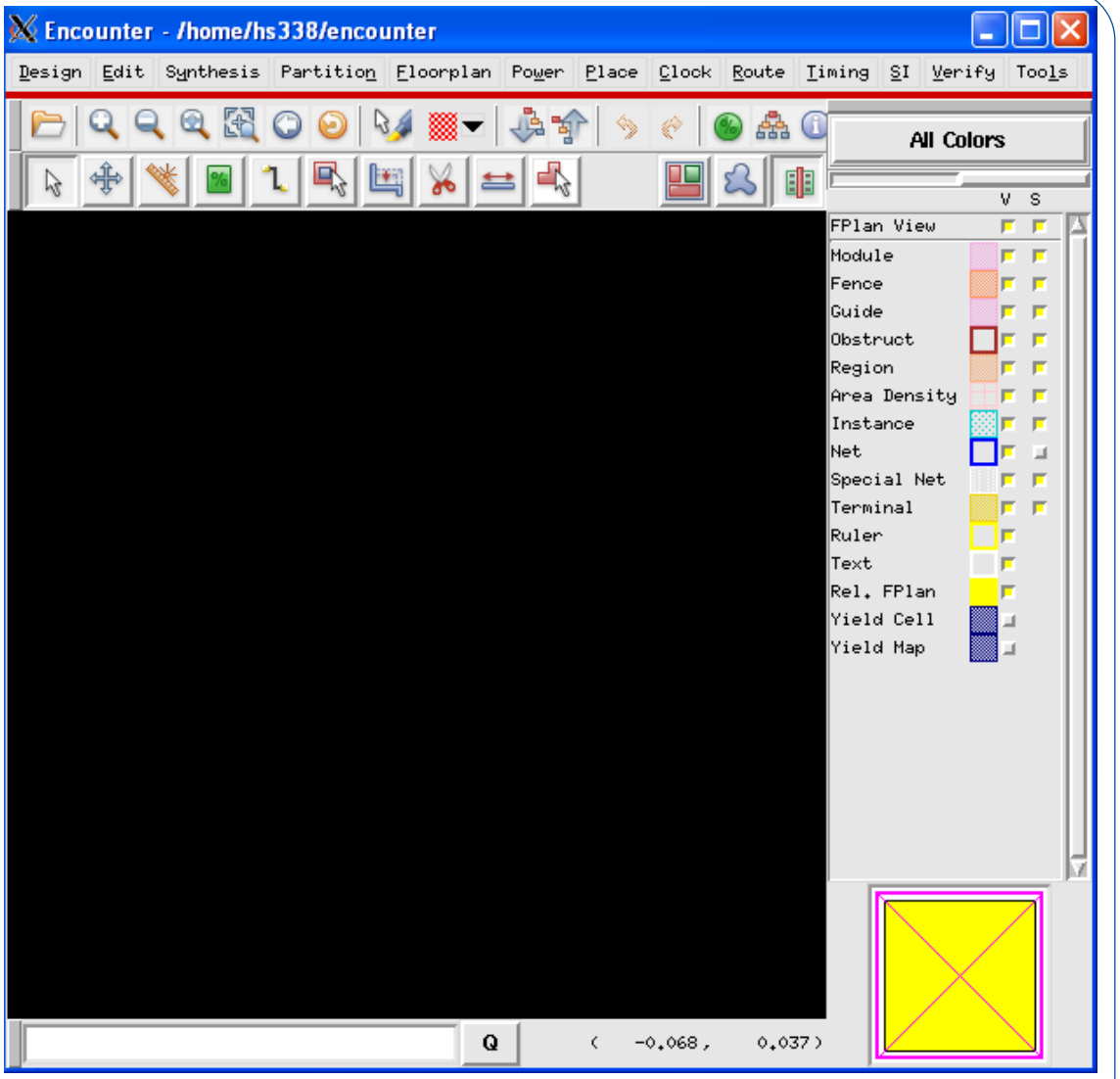
Basic structure of the chip (size, I/O, power supply …)

Placing the gates on the chip (core area)

Wiring of the gates, pads and power nets

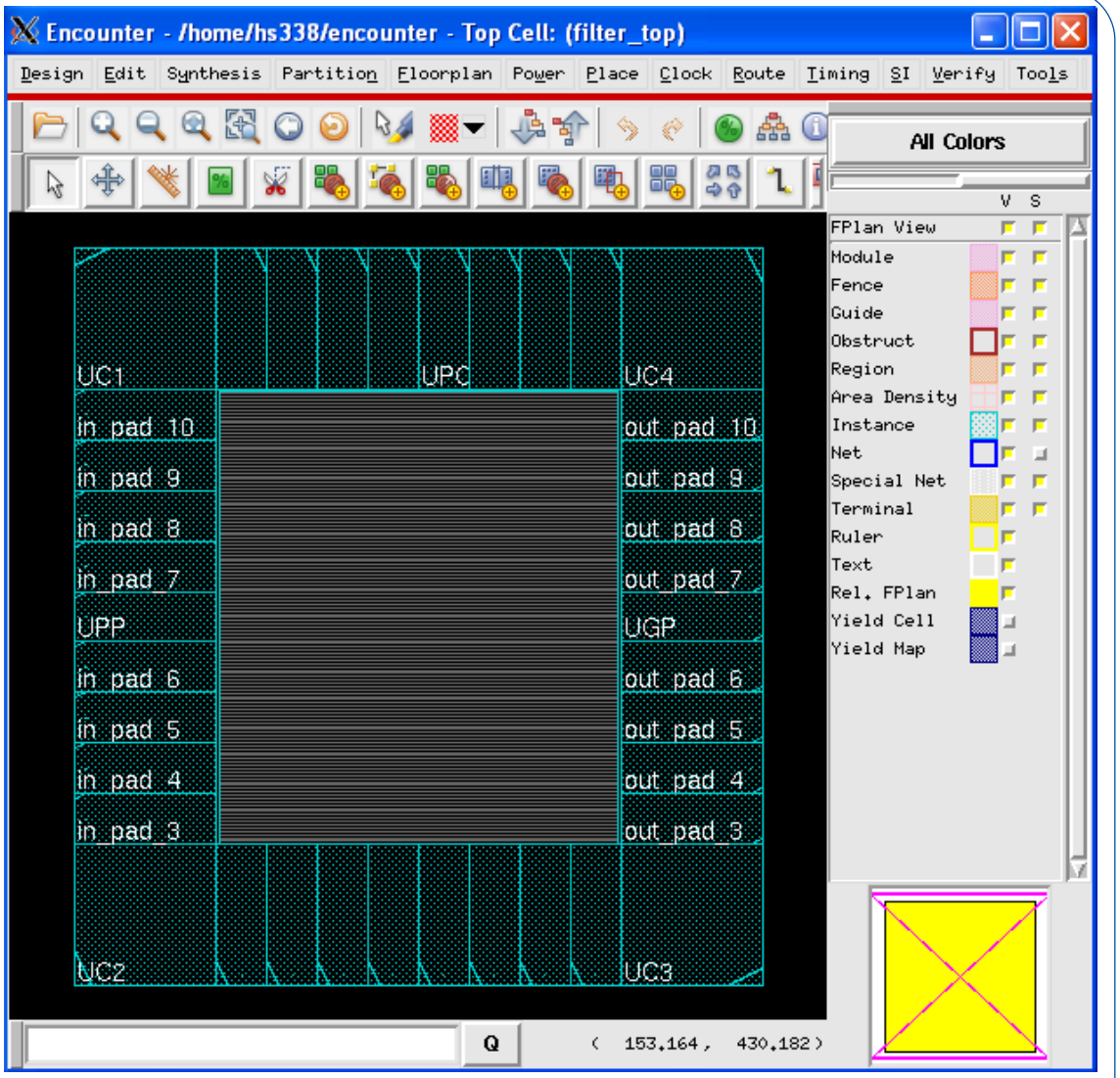
Layout–Flow II



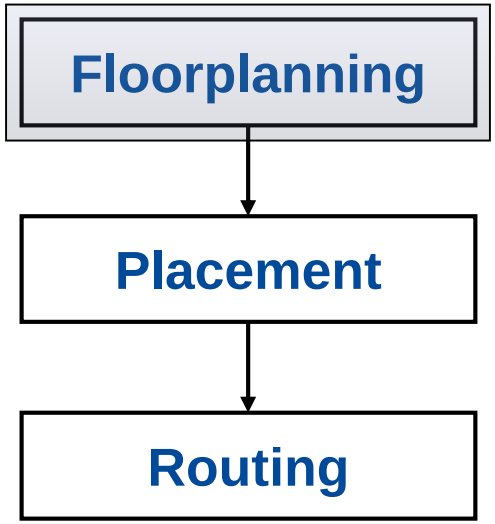


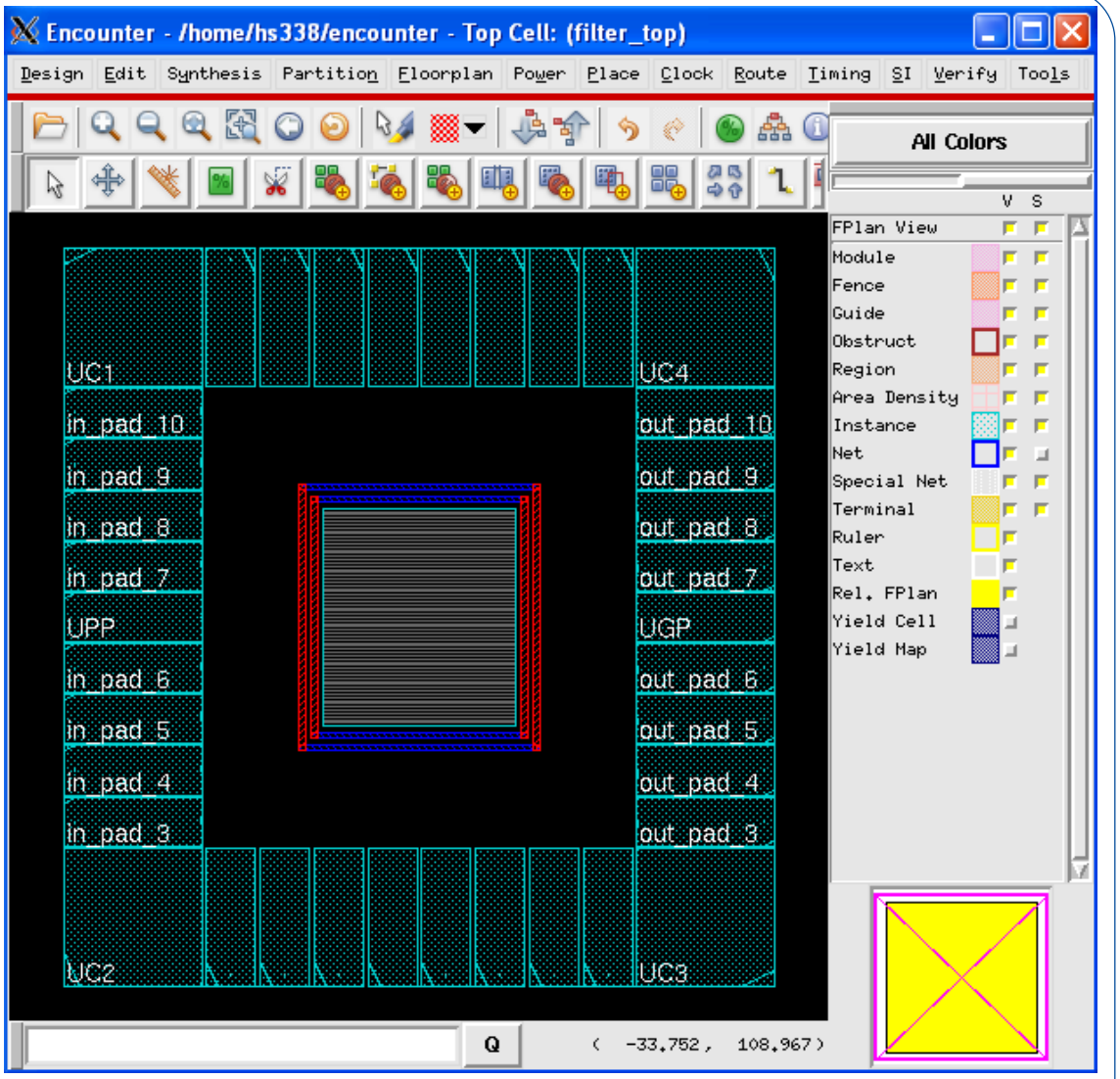
Layout–Flow III

- Floorplanning
- Placement
- Routing



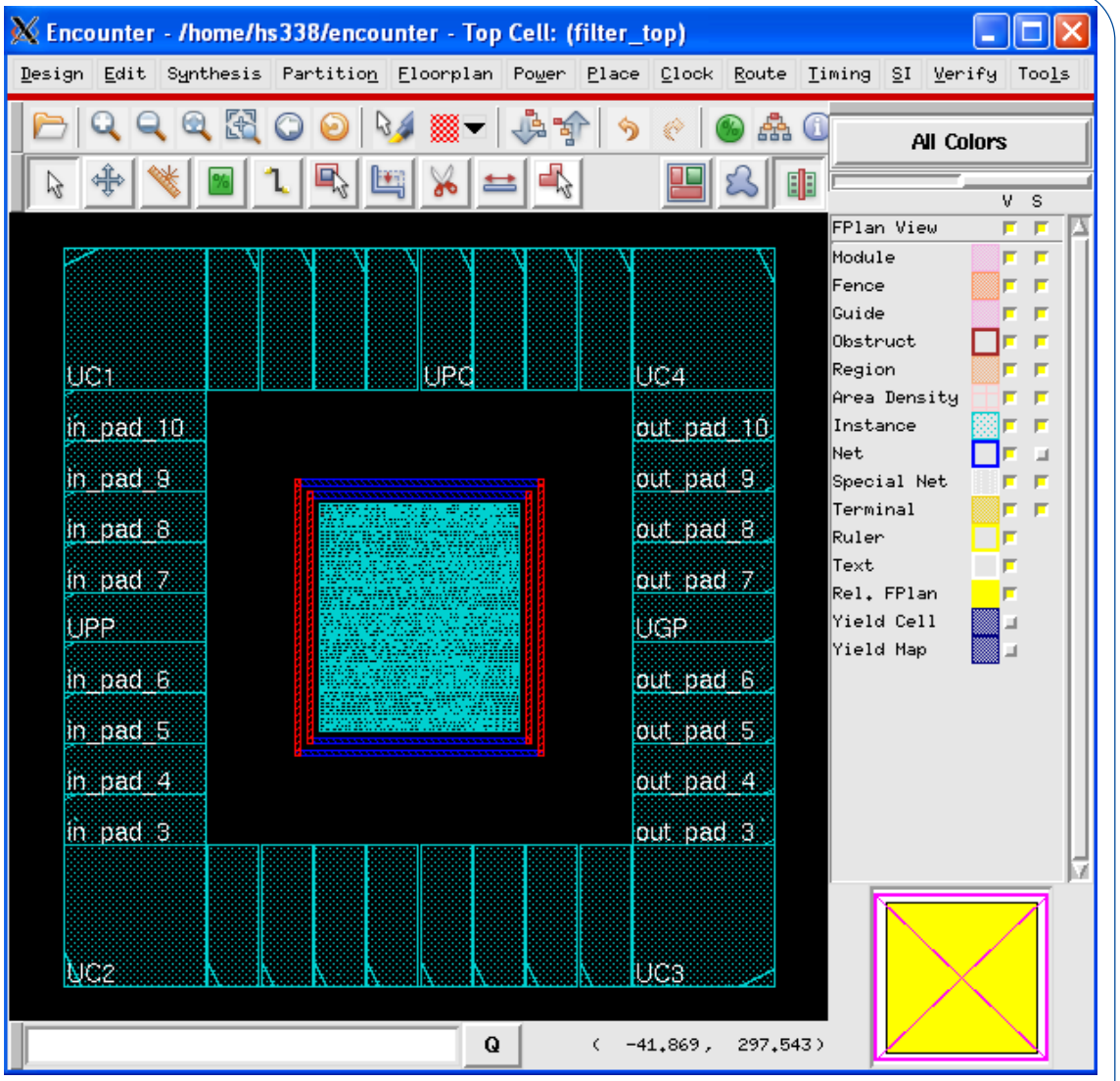
Layout–Flow IV





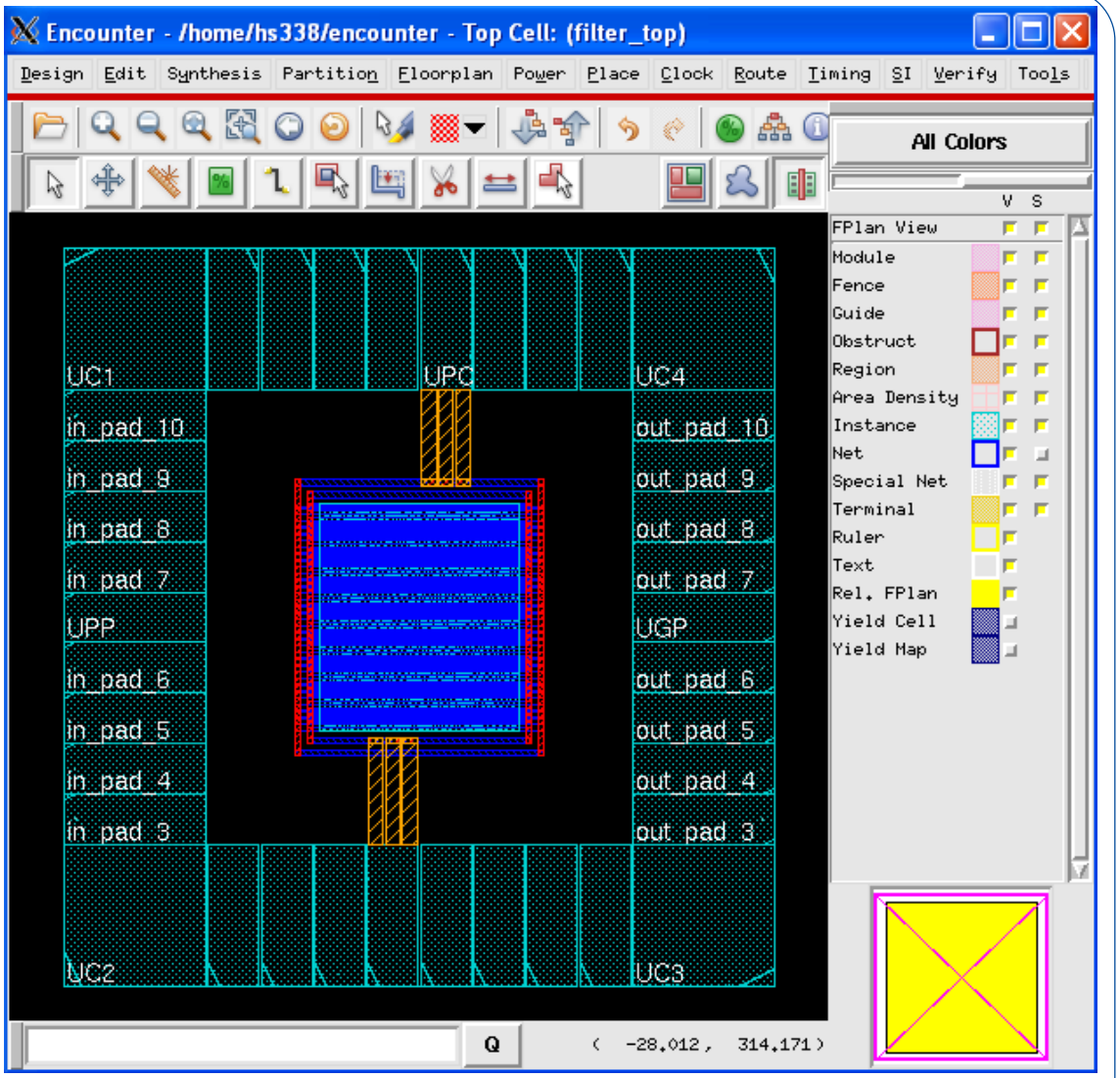
Layout–Flow V

- Floorplanning
- Placement
- Routing



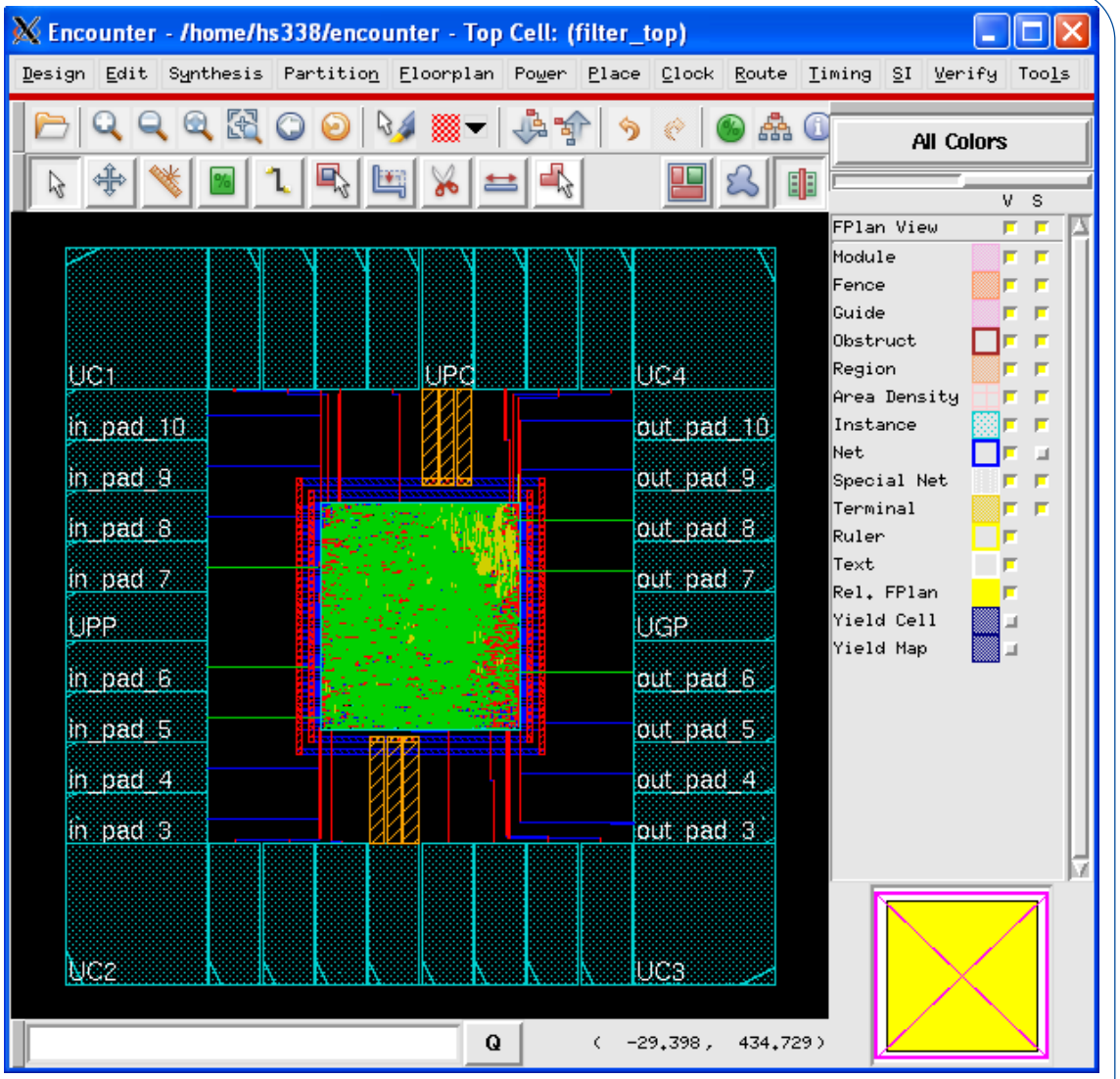
Layout–Flow VI

- Floorplanning
- Placement
- Routing



Layout–Flow VII

- Floorplanning
- Placement
- Routing



Preparation for Cadence Innovus

1. Copy scripts into your ‘vlsi_contest‘ directory
   (‘synopsys‘ folder **MUST** be located in the same folder)

```
<home>/vlsi_contest$ cp
/home/hr243/share/AdvVLSI_WS19_20/cadence_WS19_20.tar.gz .
```

2. Unpack the file and change to the cadence directory

```
tar -xvzf cadence_WS19_20.tar.gz
cd cadence
```

3. Copy your final netlist into the netlist folder

```
cp ../synopsys/results/your_filter.v ./import/netlist/
```

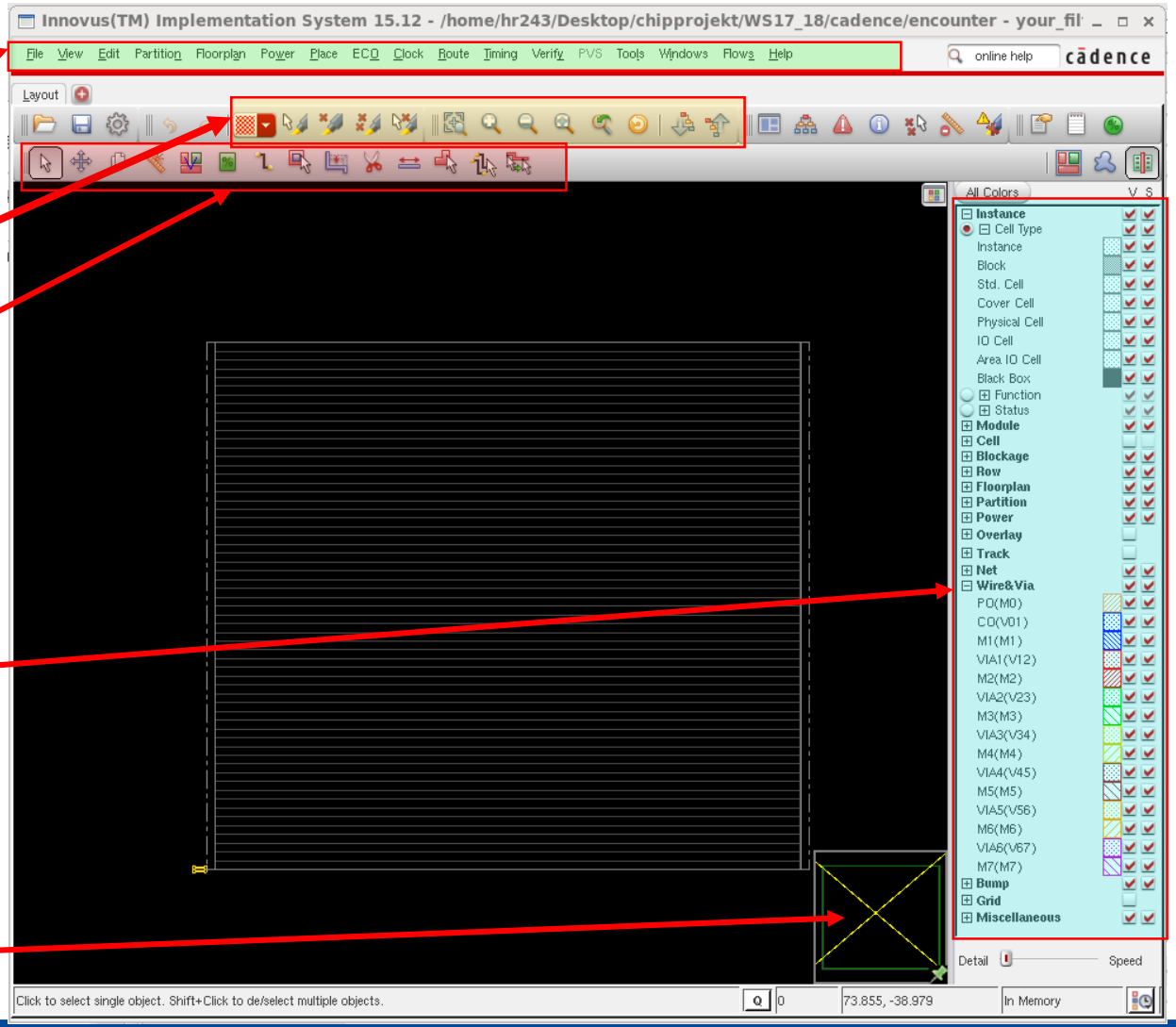
4. Execute start_innovus.csh in the innvous folder to start Innovus

```
csh start_innovus.csh
```

Remark: The names of the nets in your final netlist **must not include** slashes, backslashes or squared brackets

Overview

- Menu
- View options
- Tools
- Visibility / Selectability
- Global View



Very first step in project setup

Set timing constraint to target frequency (1GHz):

1. Change content of file ‘timing.constr’ in working directory, e.g.:

```
create_clock -name "clk" -period 1.0 -waveform {0.0  0.5} {clk}
```

Remarks:

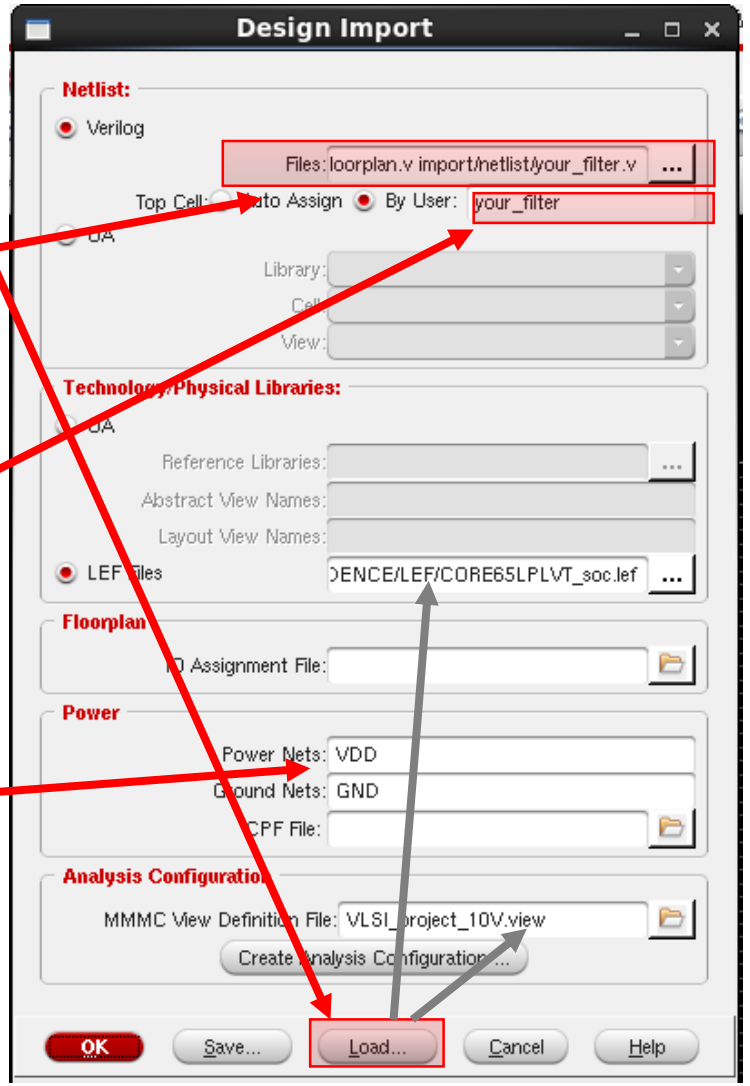
- Keep second line, which is an exception for the reset
- Changing the timing constraint during design flow is not possible!
- If a change of the timing constraint is required, there are two options:
  - Start a new chip design from the beginning
  - Lookup the path of the currently used constraint file in the ‘MMMC Browser’ and edit this file, however this can only be done **before** placement and routing of cells *(saving a design, creates a copy of the ‘timing.constr’ file in the related save-folder)*

Design-Import I

1. File → Import Design
2. Loading a global configuration file (*.globals)
3. Include the Verilog files:

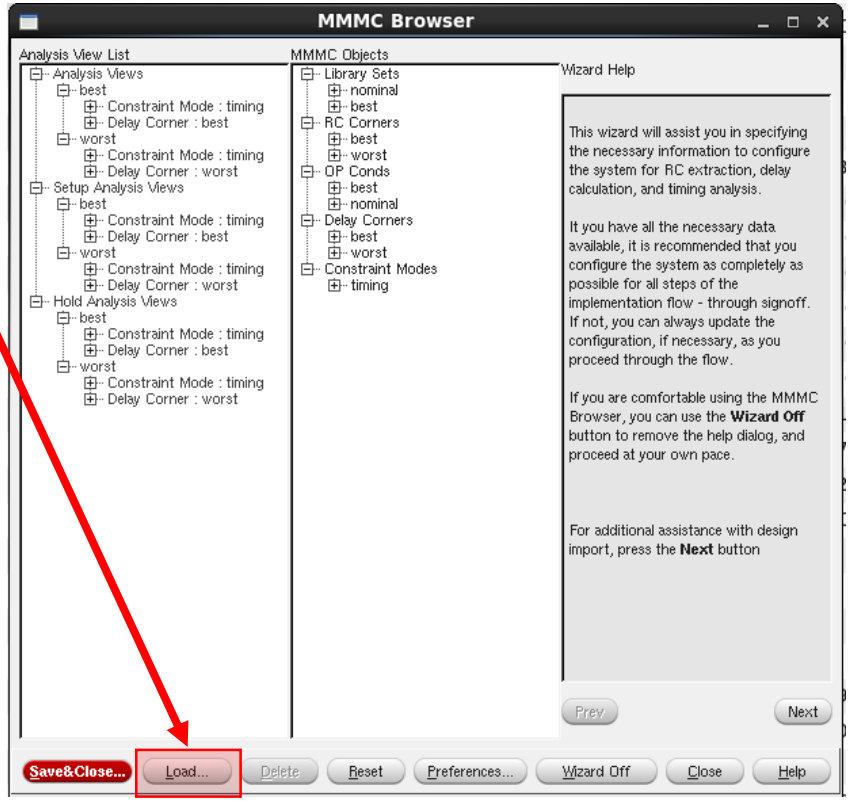
   import/floorplan/CORE65LPSVT_floorplan.v
   import/floorplan/CORE65LPHVT_floorplan.v
   import/floorplan/CORE65LPLVT_floorplan.v
   import/netlist/your_filter.v

4. Top Cell: your_filter
5. Sign global nets for power:
   - Power Nets: VDD
   - Ground Nets: GND
6. Click „Create Analysis View Configuration …“



Design-Import II

7. In the ‘Multi-Mode Multi-Corner Configuration‘ (MMMC) browser:
   - Load a specific configuration file (*.view) according to your chosen supply voltage:
     VLSI_project_10V.view *or*
     VLSI_project_11V.view *or*
     VLSI_project_12V.view *or*
     VLSI_project_13V.view



→ Loads related timing libraries, contraints, capactiance files and creates modes (best, worse)

→ **Have a look and get comfortable with all settings and parameters!**

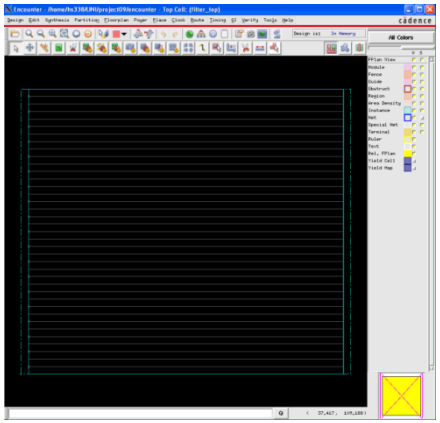
Design-Import III

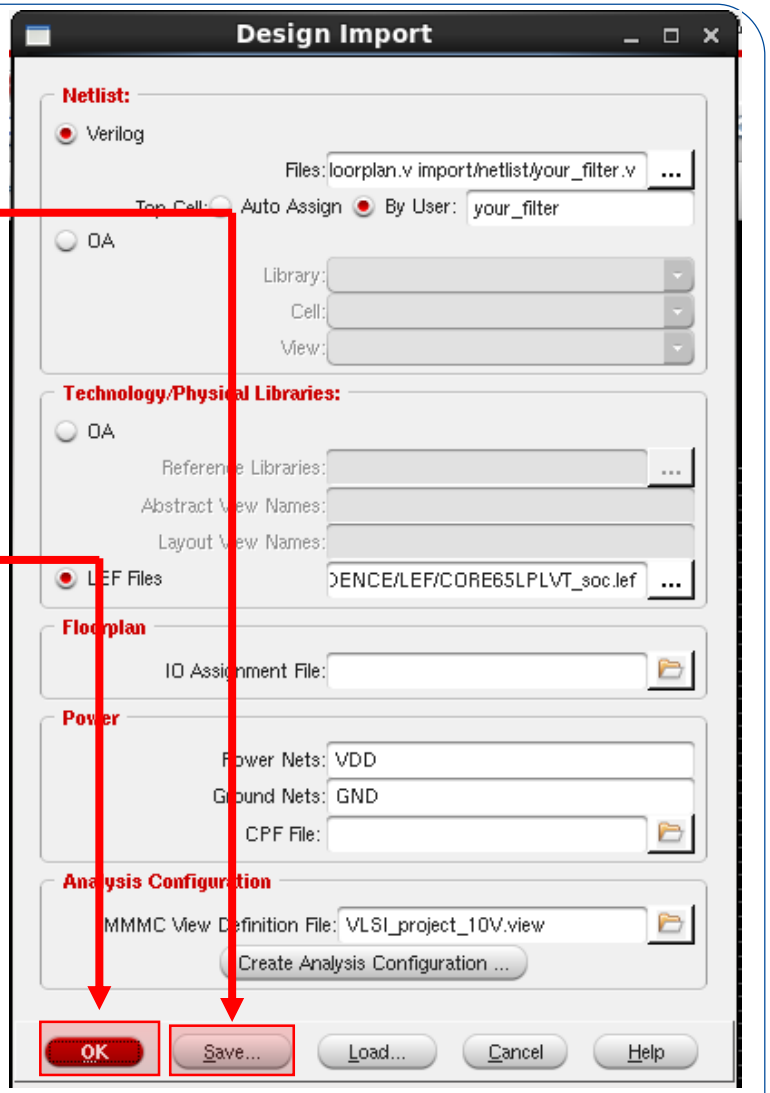
8. Save the import settings in your encounter directory:

   ‘*<home>/cadence/VLSI_project.globals*‘

   (Allows to reload default settings)

9. Click OK to complete import





Floorplanning

← *Most critical step in place & route flow*

General steps for floorplanning:

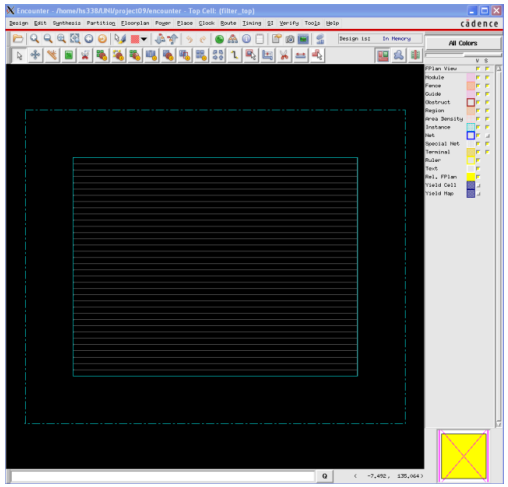
- Determination of chip and core size
- Placement of pads
- Creation of the rows for gate placement
- Placement of macro blocks (memory, PLL, DAC, ...)
- Applying power supply

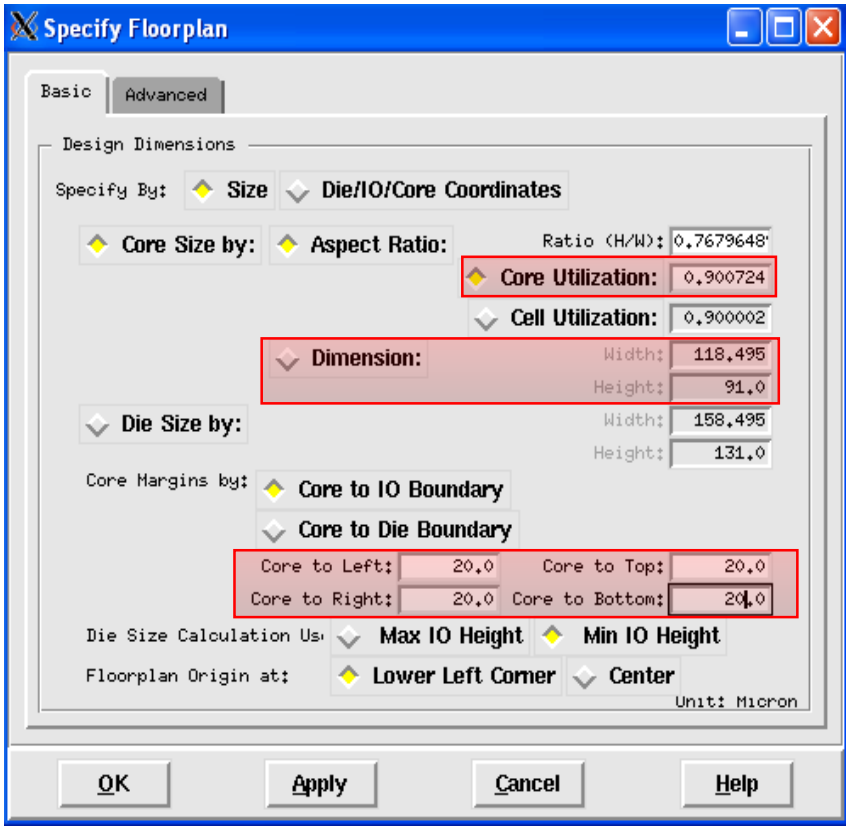
Hint: It is advisable to save the complete design after each layout step

- *File → Save Design → select type 'Innovus' → save as '<name>.invs'*
- Innvous will create a .invs file and a.dat directory
- Create a new subfolder in folder '…/cadence/saved_designs/' for each saved design

Floorplanning I

1. Floorplan → Specify Floorplan
   - Modify ‘Core Utilization‘ and ‘Core Margins‘ to suit your needs
2. Click OK → layout appears





Hint: Initial Core Utilization of 60-75% and 20µm Core Margins are a good starting point

Floorplanning II

3. Power → Connect Global Nets…

   (Connect all power pins to the global power nets)

   (1) Instance Basename: *

   (2) Pin Name(s): vdd

   (3) To Global Net: VDD

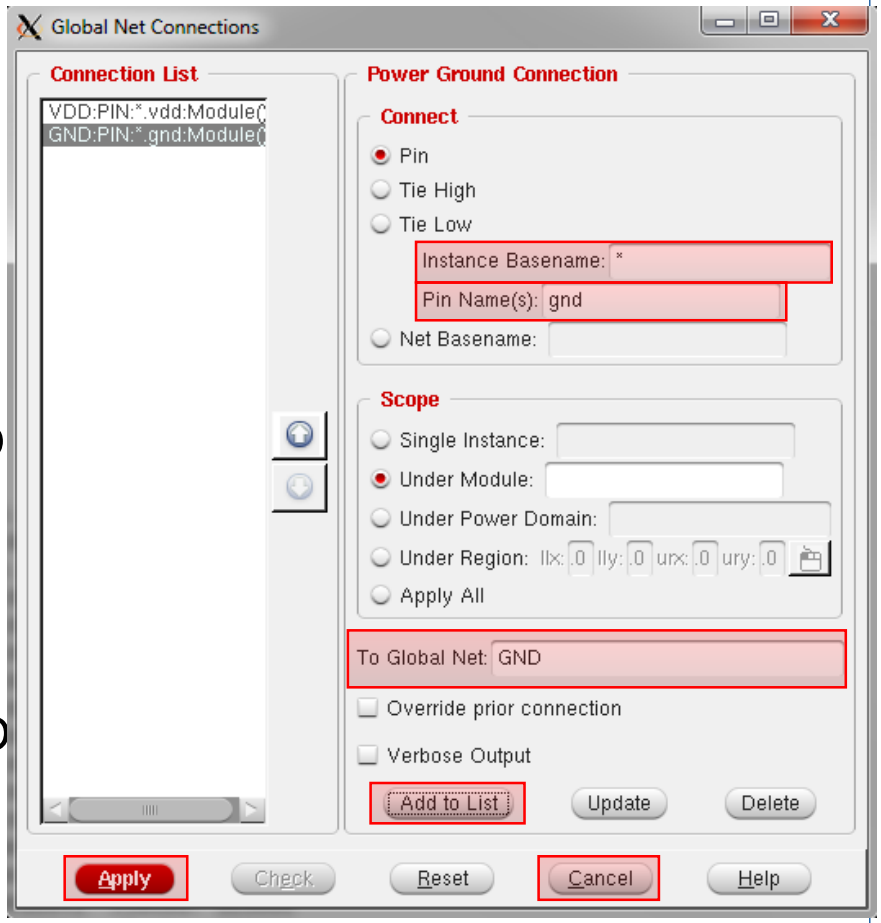
   (4) Add to List

   (5) Instance Basename: *

   (6) Pin Name(s): gnd

   (7) To Global Net: GND

   (8) Add to List

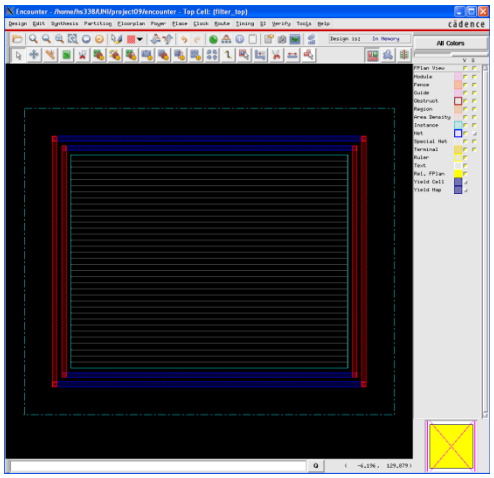


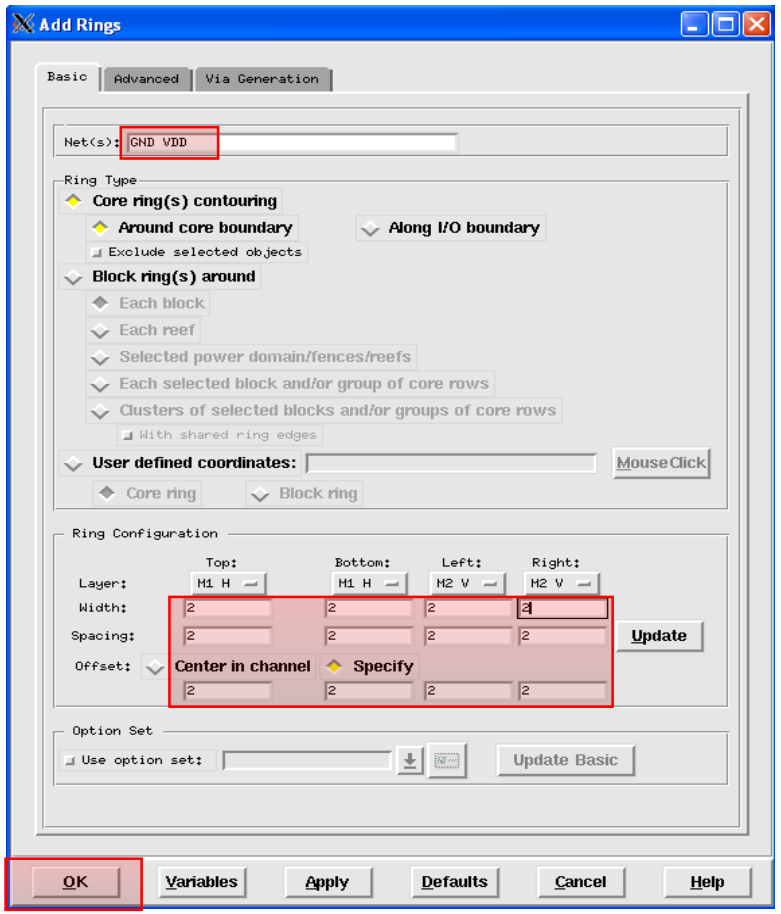
4. Apply and Cancel to close

Floorplanning III

5. Power → Power Planning → Add Ring…
   (1) Net(s): GND VDD
   (2) Ring configuration: 2 µm (can be adapted to your needs)
6. Click OK

   Modified layout appears





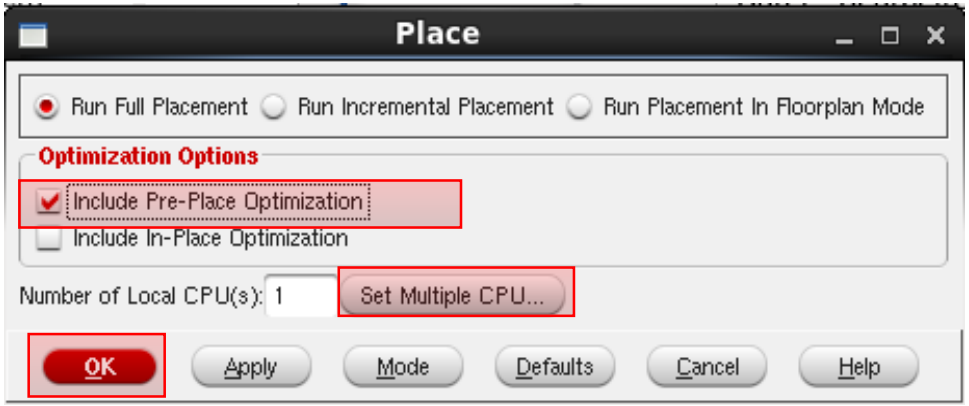
Hint: Also multiple rings can be used (‘Net: GND VSS GND VSS …‘)

Placement

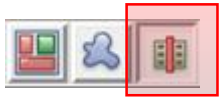
Placement of standard cells. Goals:

- Connections as short as possible
- Wireable design

1. Place → Place Standard Cell…
   - Deselect ‘Include Pre-Place Optimization‘
   - Set number of CPUs
   - Click OK



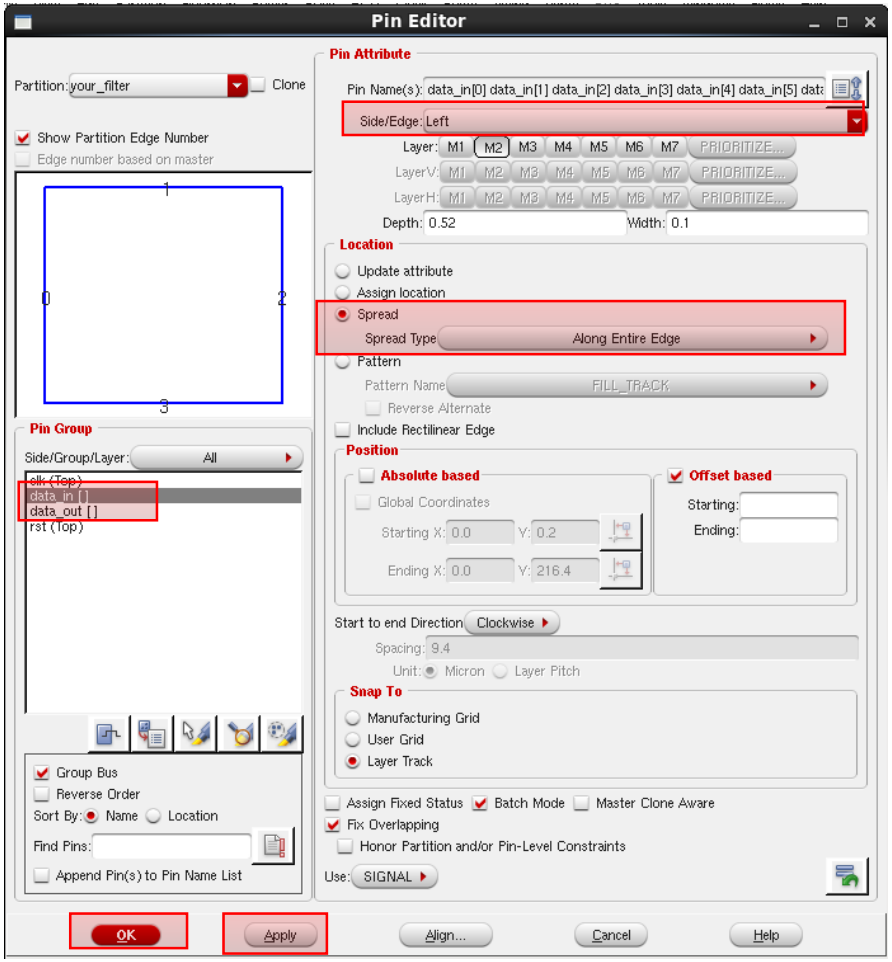
2. Modified layout appears after change to physical view:
   (placed standard cells and **trial** routing will be shown,
   the **trail** routing will disappear during the next steps → routing will be done later)



Routing / Floorplanning

1. Edit → Pin Editor…
   (1) Pin Group: data_in[ ]
   (2) Spread: Along Entire Edge
   (3) Side: Left
   (4) Click Apply
   (5) Pin Group: data_out[ ]
   (6) Spread: Along Entire Edge
   (7) Side: Right
   (8) Click Apply

2. Click OK



Routing

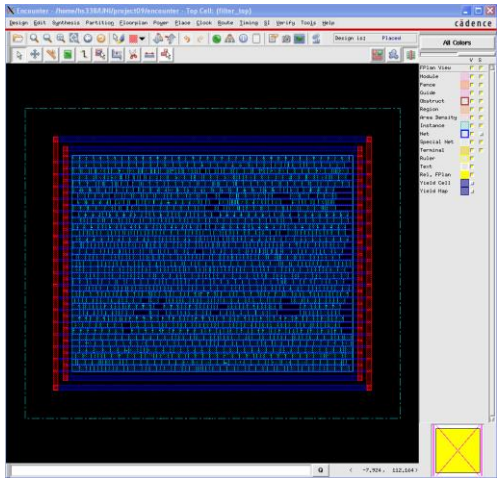
General remarks on routing:

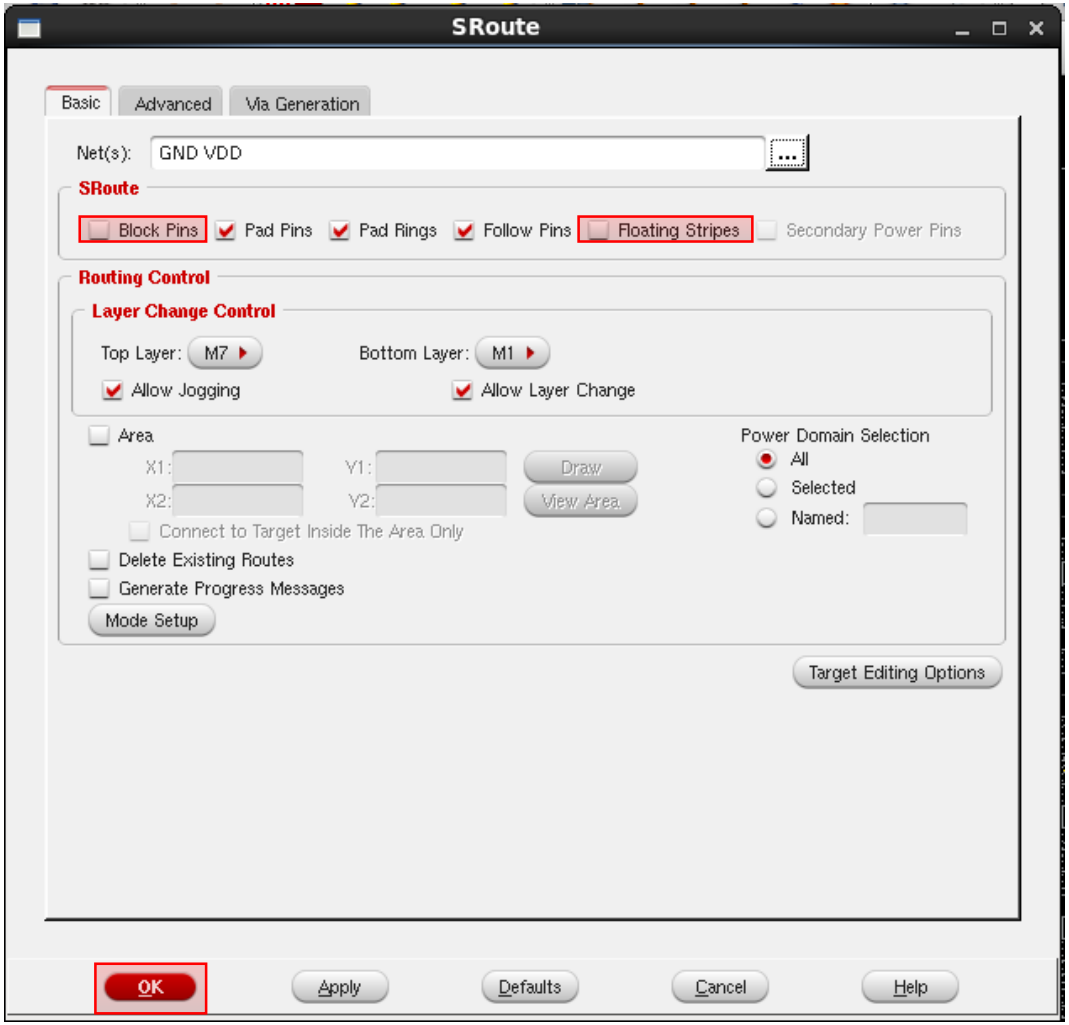
- Wiring of power nets
- Wiring of signal wires between the standard cells and pads
- Up to 7 metal layers
- Nano Route:
  - Wiring of power nets for complete power supply
  - Global Routing
  - Detailed Routing
  - Timing driven possible

Routing I

Routing of power nets

1. Route → Special Route
   Deselect “Block Pins“, “Stripes“
   select nets: GND VDD
2. Click OK
   Modified layout appears



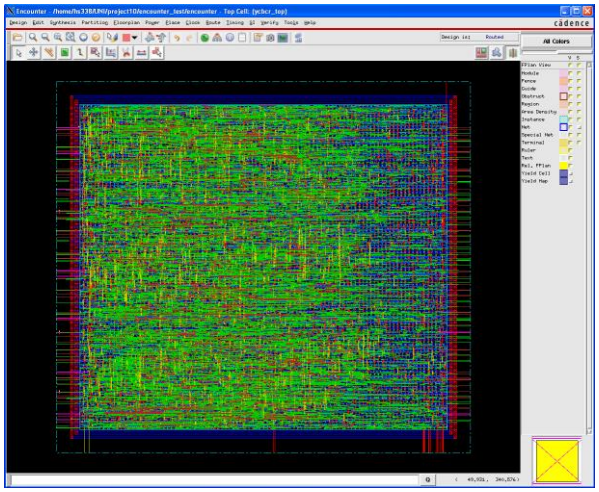


Routing II

Routing of cells

3. Route → NanoRoute → Route…
4. Click OK

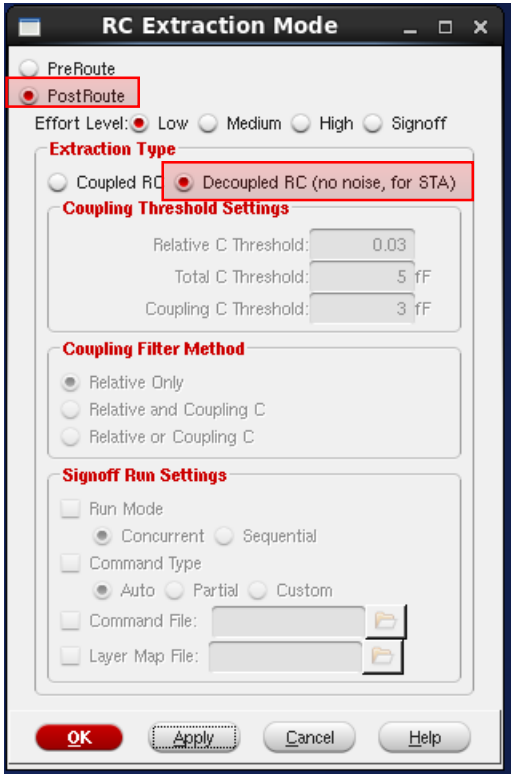
   Modified layout appears



Hint: Several optimizations possible due to combination of FixAnt, Timing-, SI-Driven,…

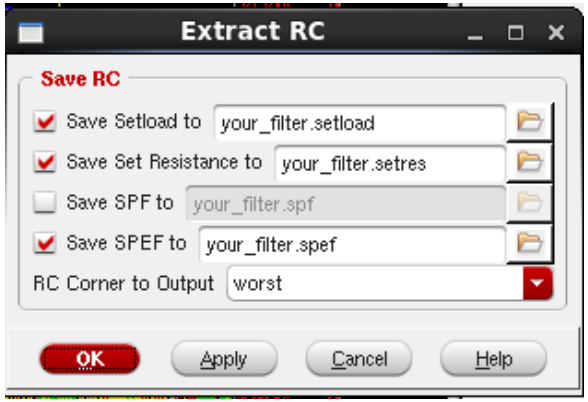
Timing Analysis I

11. Tools → Set Mode → ‘Specify RC Extraction Mode…‘
12. Select „Post-Route“ and „Decoupled“
13. Click OK

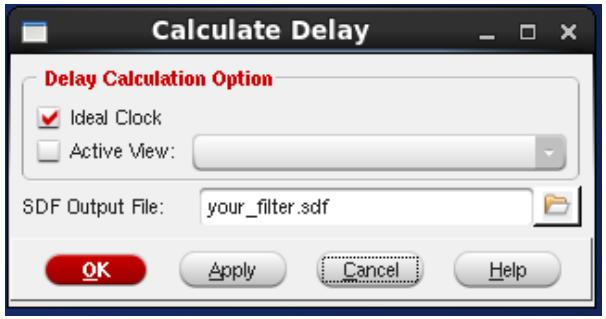


Timing Analysis II

7. Timing → Extract RC
   (1) Select „Save Setload to“
   (2) Select „Save Set Resistance to“
   (3) Select „Save SPEF to“
8. Click OK



9. Timing → Write SDF
10. Click OK



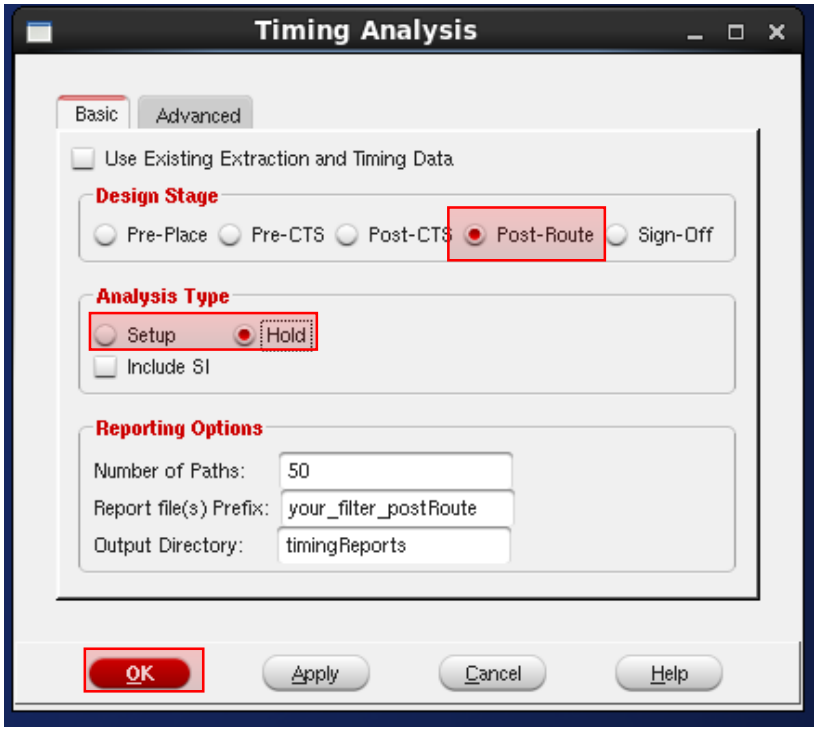
11. Execute to following command in the Innovus terminal:

```
innovus X> setAnalysisMode –analysisType onChipVariation –cppr both
```

Timing Analysis III

11. Timing → Report Timing …
    (1) Select „Post-Route“
    (2) Select „Hold“ time analysis
    (3) Click OK
    (4) Check, if a hold time violation occurs in Console
    (see next slide)
12. Timing → Report Timing …
    (1) Select „Post-Route“
    (2) Select „Setup“ time analysis
    (3) Click OK

→Calculate your clock period(next slide)



Information for the slack is given at the terminal (see next slide):

- Negative slack: Violation
- Positive slack: No violation

Timing Analysis IV

Based on setup analysis: T_{min} = clk – WNS

Max. frequency of your design = $\frac{1}{T_{min}}$

Ex.: create_clock -name "clk" -period 1.5 -waveform {0.0 0.75} {clk}
clk = 1.5 ns

a) Timing violation
T_{min} = 1.5 – (– 0.034)
= 1.534 ns
f_{max} = 651.9 MHz

b) No violation
T_{min} = 1.5 – 0.166
= 1.334 ns
f_{max} = 749.6 MHz

timeDesign Summary

Setup mode	all	reg2reg
WNS (ns):	-0.034	-0.034
TNS (ns):	-0.644	-0.644
Violating Paths:	19	19
All Paths:	2548	1200

Setup mode	all	reg2reg
WNS (ns):	0.166	0.166
TNS (ns):	0.000	0.000
Violating Paths:	0	0
All Paths:	2548	1200

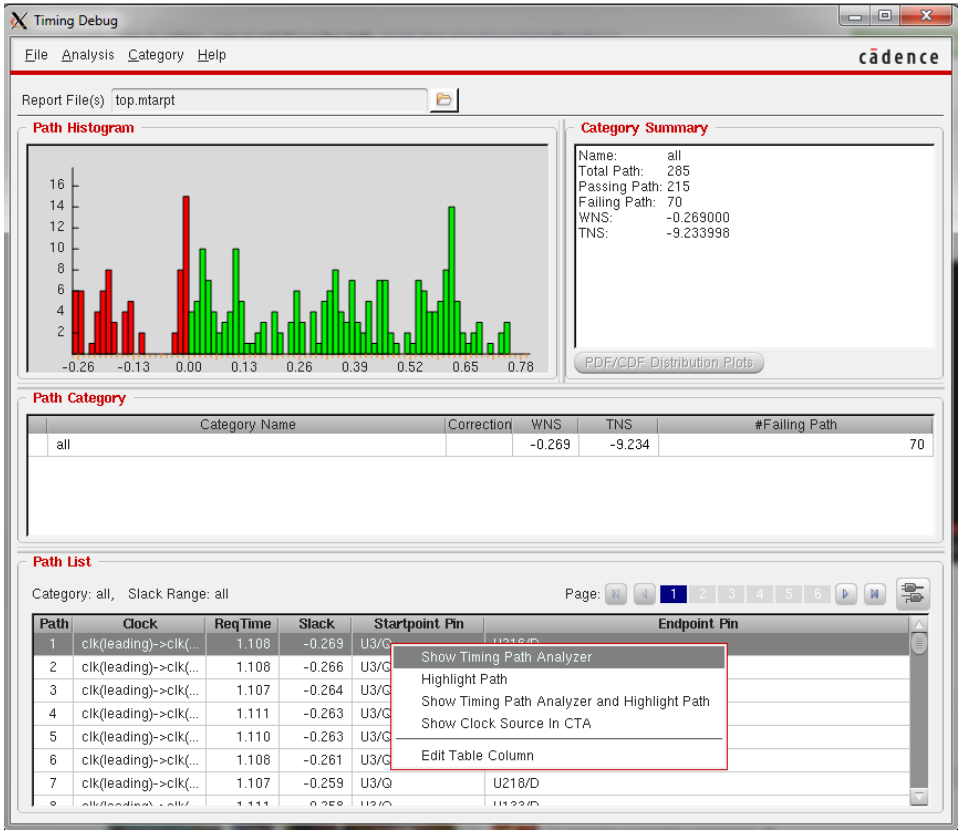
- Change clock in file ‘timing.constr‘ (-> 1ns) (in work dir)
- Keep second line, which is an exception for the reset
- Constraint is also used during placment and routing (optional)

Timing Analysis V

Further specific timing information is available:

- Clock declaration
- Exception for the reset
- Path slacks
- Number of violated paths
- Path highlighting
- ...

15. Timing → Debug Timing
16. Click OK



Power Analysis I

1. Change to the ‘synopsys’ folder

```
<home>/vlsi_contest/cadence$ cd ../synopsys/
```

2. Move the following files from ‘cadence’ folder to ‘synopsis’ folder:
   - `filter_power_cadence.tcl`
   - `start_power_analysis.sh`
   - `setload_fix.pl`
   - `your_filter.setload`

3. Default path and file name of netlist used in this step:
   ‘./results/your_filter.v’

   (If you changed the netlist with Innovus you need to update this file! E.g. using ECO->optimize_design)

Power Analysis II

4. Adapt the following parameters in *'filter_power_cadence.tcl'* script:
   - Frequency (change also in test bench 'synopsys/sim/files/fir_filter_tb.vhd')
   - Voltages
   - Libraries

5. Start the power script, this will convert the parasitics file from Cadence Innvous and start Synopsys Design Vision

```
<home>/vlsi_contest/synopsys$ ./start_power_analysis.sh
```

4. In Design Vision

   File → Execute Script → 'filter_power_cadence.tcl'

5. Power values in the final power report:

   './results/design_report.txt'

Attention: If you copy new files into the 'synopsys' folder, restart design vision!

If you use a netlist modified by Innovus, ensure that you load all voltage threshold libraries of your selected VDD (LVT,SVT,HVT) in script 'filter_power_cadence'

Tasks and further information for phase 5

Tasks for phase 5

- Target: Presentation of results for a working and optimized design layout (Observe/discuss differences to synthesis results)
- **Minimum frequency of your design: 1GHz (period 1ns)**
- Target benchmark: $$\boldsymbol{Metric} = \frac{\boldsymbol{Freq}\,[\boldsymbol{MHz}]^2}{\boldsymbol{Att} * (\boldsymbol{P_{leak}} + \mathbf{0.01} * \boldsymbol{P_{dyn}})}$$ 1 % Duty Cycle
- P_{leak} and P_{dyn} are available from: ‘./results/design_report.txt’
- **Final Phase for Contest (ASIC Metric)**
- Consider impact for your design:
  - Differences to synthesis results
  - Layout targets / constraints

Handover infos for phase 5

- Final firm design deadline: **January 27th 2018, 23:59 CET**
- Upload your presentation: **January 28th 2018, 23:59 CET**

- Design handover:

1. Copy your latest verilog netlist (of your final/best chip layout):

```
cp /home/<your_login>/vlsi_contest/cadence/import/netlist/your_filter.v
/home/hr243/share/AdvVLSI_WS19_20/<your_login>/
```

2. Save your final chip layout in Innvous in your directory:
   - *File → Save Design → as type 'Innovus'*
   - File name: *'your_filter_<your_login>.invs'* (e.g. *your_filter_at101.invs*)

Handover infos for phase 5

3. Please verify that your saved design and your parameters are correct:
   (1) Restart Cadence Innovus
   (2) **File → Restore Design** (choose Design from step 2: your_filter_<your_login>.invs)
   (3) Rerun the timing AND power analysis, record the final achieved values

4. Copy your final layout and its directory:

```
cp /home/<your_login>/vlsi_contest/cadence/your_filter_<your_login>.invs
/home/hr243/share/AdvVLSI_WS19_20/<your_login>/
```

(e.g. cp /home/at101/vlsi_contest/cadence/your_filter_at101.invs /home/hr243/share/AdvVLSI19_20/at101/)

```
cp –r
/home/<your_login>/vlsi_contest/cadence/your_filter_<your_login>.invs.dat
/home/hr243/share/AdvVLSI_WS19_20/<your_login>/
```

(e.g. cp –r /home/at101/vlsi_contest/cadence/your_filter_at101.invs.dat /home/hr243/share/AdvVLSI19_20/at101/)

5. Copy your whole ‘vlsi_contest’-folder (incl. synopsys and cadence folders)

```
cp -R /home/<your_login>/vlsi_contest
/home/hr243/share/AdvVLSI_WS19_20/<your_login>/
```

Handover infos for phase 5

5. Check the transfer directory, should be similar like this:

```
ls -lasi /home/hr243/share/AdvVLSI_WS19_20/<your_login>/
```

corresponding_data.txt	(file)
your_filter.v	(file)
your_filter_<your_login>.invs	(file)
your_filter_<your_login>.invs.dat	(dir)
vlsi_contest	(dir)

6. Content of file ‘corresponding_data.txt’ :

(1) Power (dynamic and leakage)

(2) Number of pipeline stages

(3) Frequency of your design (calc. frequency at 0 slack from Innovus Timing Report)

(4) Benchmark metric for the ASIC

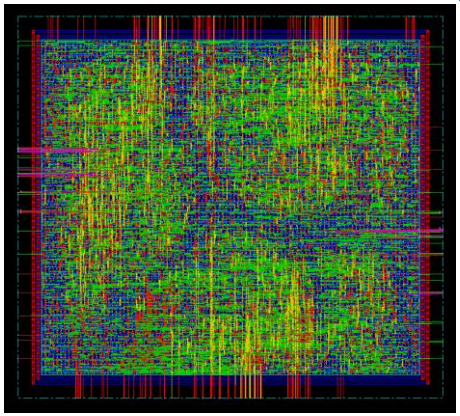
(5) Core size (width, height), core utilization from Innovus: Floorplan->Specify Floorplan

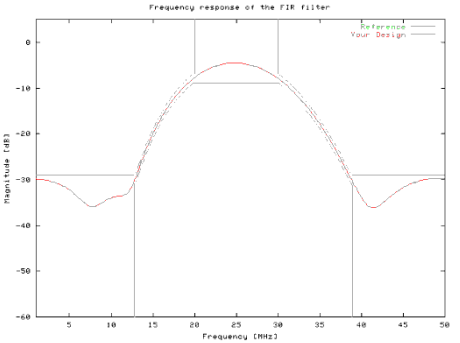
Mandatory infos for results display

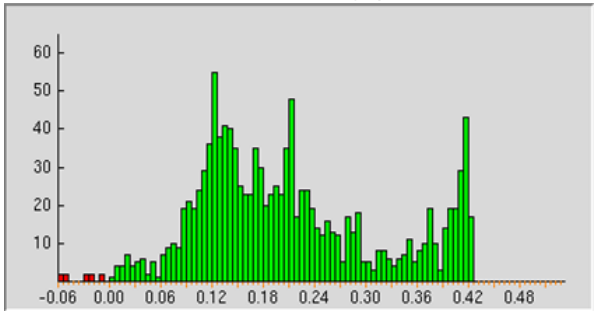
Include these elements in your presentation:

- Picture of chip layout
- Frequency response
- Path histogram
- Design parameters (with correct units)

Timing (T_{min} / f_{max})	743 ps / 1345.9 MHz
Power (P_{dyn} / P_{leak})	106.83 mW / 2 µW
Pipeline Stages	2
Benchmark	[x]
Core size	90777 µm²
Core utilization	90.09 %







Ideas for chip design and presentation

Possible fields for investigation (all have to include layout phase):

- Core row utilization
- Core and chip size
- Core position
- Aspect ratio of core area
- Pin positions
- Placing options (Global, incremental, timing driven …)
- Routing options (NanoRoute, WRoute, Timing driven, DFM …)
- Size and spacing of power rings
- Different power nets (rings, grids, stripes …)
- Limited metal usage
- Different gate libraries
- Application conditions
- Clock tree synthesis
- Manual placement and routing
- Architectural impact on ASIC and FPGA
- DFM: Design For Manufacturing (Via doubling, Via minimization, Metal fill, Wire spreading …)
- Electromigration
- Antenna fixing
- ...

Use build-in help

- search field and help buttons in GUI
- Example using command line:
  - `help *Design*`
  - `help checkDesign`
  - `man checkDesign`

Remarks on Power Stripes and Blockage Fields

Stripes

- If design is easy to route, use thick frequent stripes on the top 2 metal layers
- Otherwise start stripes on Metal4 and have stripes on every layer up
- Layer Wire Directions
  - Top/Bottom: Metal 7,5,3,1
  - Left/Right: Metal 8,6,4,2

Blockage

- Block standard cell placement under power and ground stripes
- Place -> Specify -> Placement Blockage…
- Select a metal layer and click OK

Questions?

Design Deadline (files in centOS folder): January, 27th, 23:59
Presentation Deadline: January, 28th, 23:59

Next meeting: January, 29th

Target: Presentation of complete chip layout and its results.
(Observe differences of the design phases)