

# Challenges of physical verification on heterogeneous platform

Alexey Kalinov

Cadence Design Systems,  
Moscow, Russia  
[akalinov@cadence.com](mailto:akalinov@cadence.com)





# Content

- Cadence
- Physical verification
- Parallel implementation challenges
- Possible benefits of heterogeneous platform



## Cadence

Designing and manufacturing of modern semiconductor devices would not be possible without electronic design automation (EDA).

Cadence Design Systems is the world's leading EDA company. Cadence provide software, hardware and service for all tasks necessary for design and manufacturing.

Moscow site is specialized in Physical Verification and Parasitic Extraction. About third of R&D have PhD or higher level.



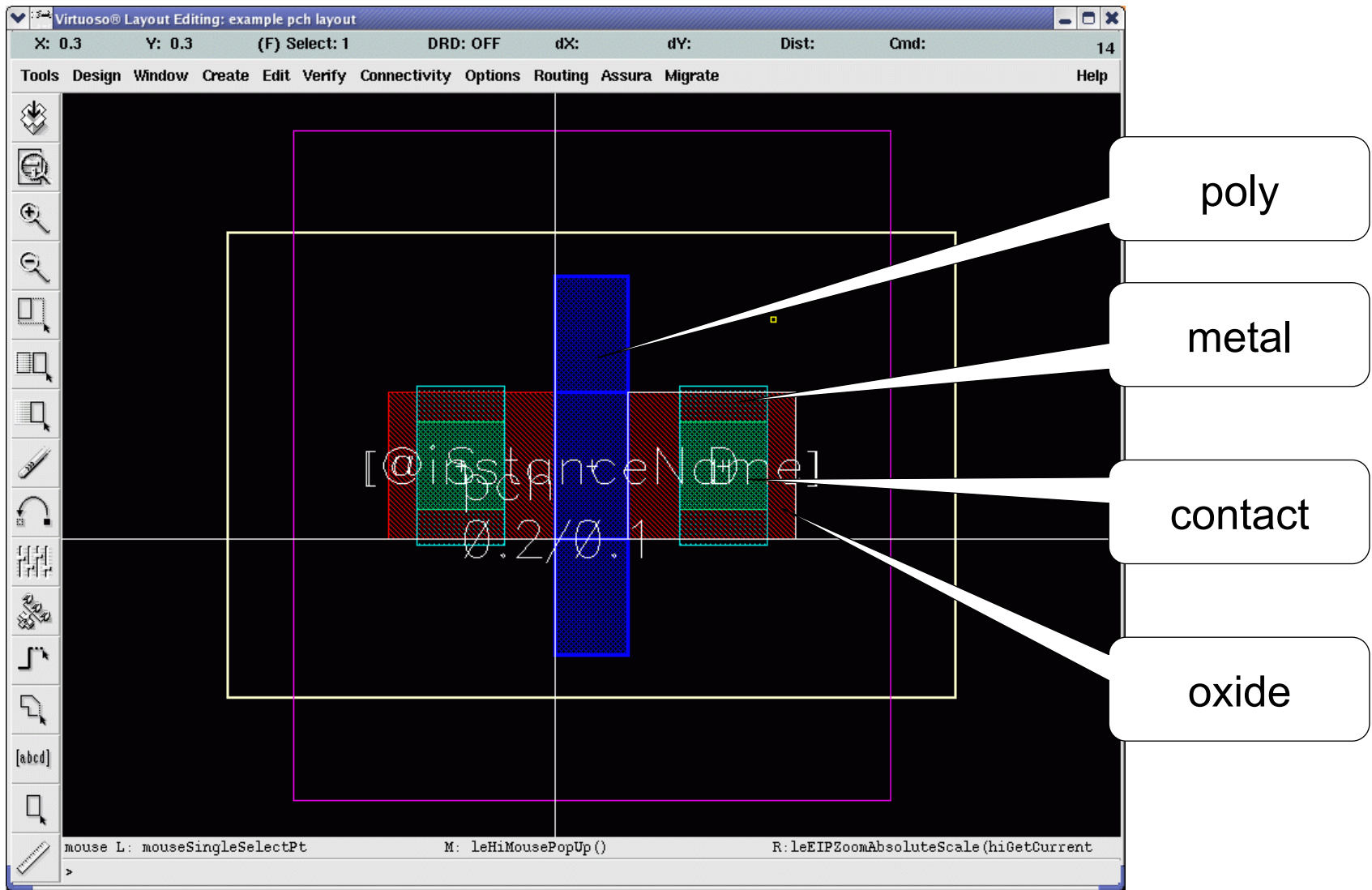
## Physical verification

The integrated circuit layout is the representation of an integrated circuit in terms of planar geometric shapes which correspond to the patterns of metal, oxide, or semiconductor layers that make up the components of the integrated circuit.

Physical Verification is a process whereby an **integrated circuit layout** is checked via Electronic Design Automation software tools to see if it meets certain criteria.

Very computationally intensive process.

# Layout example (p-channel transistor)

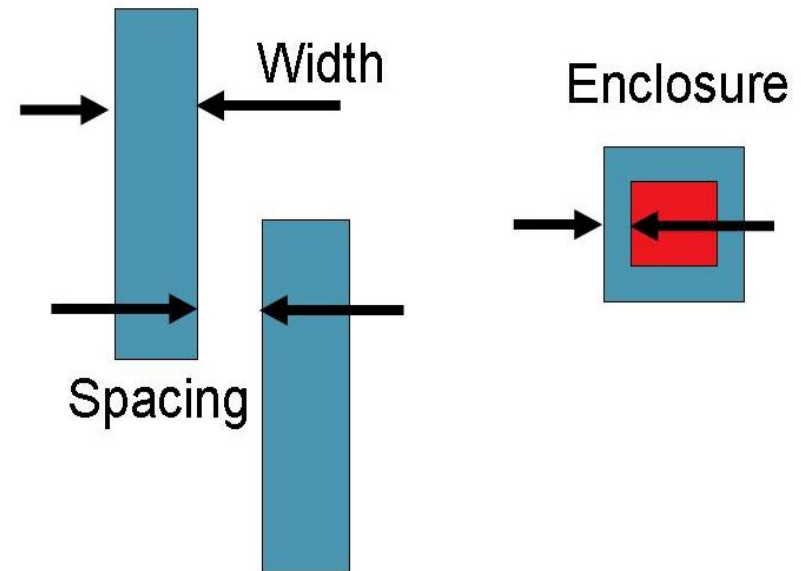



## Physical verification steps (DRC - Design Rule Check)

Design Rule Check determines whether a particular chip layout satisfies a series of recommended parameters called Design Rules. They are specific to a particular semiconductor manufacturing process.

A design rule set specifies certain geometric and connectivity restrictions to ensure sufficient margins to account for variability in semiconductor manufacturing processes, so as to ensure that most of the parts work correctly.

The three basic DRC checks





## Physical verification steps (LVS – Layout Versus Schematic)

A successful DRC ensures that the layout conforms to the rules designed/required for faultless fabrication. However, it does not guarantee if it really represents the circuit you desire to fabricate. This is where an LVS check is used.

The Layout Versus Schematic determines whether a particular integrated circuit layout corresponds to the original schematic or circuit diagram of the design.

LVS steps:

- devices and connectivity extraction;
- devices parameters extraction;
- creation of layout diagram (netlist);
- schematic and layout diagram comparison





## Designer <-> fab

Fab (semiconductor manufacturer) – formulate DRC and LVS rules according to its manufacturing process.

Layout designer check his/her design according to the rules.

Fab examine input layout for rules compliance.



## Layout representation

Hierarchical structure

Building blocks – cells contains shapes and instances of other cells.

A shape belongs to a layer( oxide, poly, contact).

Shape/intances introduce data parallelism.

Highly iregular structure of the data.



## Rule representation

DRC and LVS rules are like

gate = and( oxide poly )

Rule corresponds to a set of tasks. Input/output of the task is set of layers. Rules introduce a task parallelism.

It is very hard to predict time of the tasks execution.

Layers introduce additional irregularity of computations.



## Parallel implementation challenges

Huge potential for data parallelism – modern designs are about tens gigabytes. Irregular structure of data is the main obstacle in use of distributed data parallelism.

Computations can be represented with Directed Acyclic Graph. Tasks communicate with huge amount of data.



## Current parallel implementations

The main platform – systems in the box.

Use combination of task parallelism and data (multithreading) parallelism.

A few attempts to use clusters, NOW or grid.



## Heterogeneous platform challenges and promises

Traditional heterogeneous platforms bring additional problems in data and computations distribution because it adds additional irregularity. But those problems are not principal because in any case self adaptable algorithms are used.

Modern heterogeneous platforms promise new ideas.



## Possible heterogeneous platform benefits

### **GPGPU**

Majority of computations use integer arithmetic. Computations are very similar to that GPU are originally intended.

### **Multicore systems**

Modern systems in box are NUMA systems. It is very interesting to understand how to exploit full benefits of the architecture.



Thank you for attention

**cā d e n c e**<sup>TM</sup>

