



HAL
open science

From Semiconductor to Transistor-Level: Modeling, Simulation, and Layout Rendering Tools

João R. Raposo de O. Martins, Francisco Alves, Pietro Maris Ferreira

► **To cite this version:**

João R. Raposo de O. Martins, Francisco Alves, Pietro Maris Ferreira. From Semiconductor to Transistor-Level: Modeling, Simulation, and Layout Rendering Tools. Colloque du GdR SOC2, Jun 2022, Strasbourg, France. hal-03690082

HAL Id: hal-03690082

<https://hal.science/hal-03690082v1>

Submitted on 7 Jun 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

From Semiconductor to Transistor-Level: Modeling, Simulation, and Layout Rendering Tools

João R. R. O. Martins, Francisco Alves, and Pietro M. Ferreira

Université Paris-Saclay, CentraleSupélec, CNRS, Lab. de Génie Électrique et Électronique de Paris
, 91192, Gif-sur-Yvette, France.

Sorbonne Université, CNRS, Lab. de Génie Électrique et Électronique de Paris, 75252, Paris, France.

Email: maris@ieee.org

Abstract—Research in semiconductor field has been overconfident in commercial tools, while open science urges for novel solutions based in open source tools. The MOSLab.jl is presented as a solution aiming to a fast, accurate and modular compact models to bridge the gap between semiconductor applied physics and microelectronics research in nanometer scaled technology. The icLayoutRender tool is demonstrated as a user-friendly solution to a GDSII transcription in a vectorial graph quality image saved as a pdf file.

Index Terms—Julia, Python, MOS Modeling, Layout, GDSII

I. INTRODUCTION

Research in semiconductor field has been overconfident in commercial tools, while open science urges for novel solutions based in open source tools. From scientific MOS modeling and simulations, one may decide to study the model atoms and ions interactions through Newtonian or quantum physics. However, they do not present any analytical closed-form solution for systems with more than a few dozen particles. Besides, publishing scientific communications require accurate and repeatable results, where microelectronics research field requires a high-quality layout illustration. First, MOSLab.jl is presented. It is a recent tool aiming to a fast, accurate and modular compact models to bridge the gap between semiconductor applied physics and microelectronics research in nanometer scaled technology. Later, icLayoutRender is demonstrated as a user-friendly tool to a GDSII transcription in a vectorial graph quality image. Numerical simulation results are presented to illustrates the flexibility of MOSLab.jl. The vectorial graphical rendering is obtained by icLayoutRender in which standard tools are unable to attain.

II. MOSLAB IN JULIA

In semiconductor applied physics research, material scientist relies on 3D finite-element technology modeling and simulations to understand and explain the physics of semiconductors (Sze, 1991). To understand the electric behavior of semiconductors, one may investigate those phenomena using Technology Computer-Aided Design (CAD) tools. Such technology CAD tools, as (Synopsis TCAD, 2021), are powerful computer simulation software to develop and optimize semiconductor process technologies to an electronic device. However, reasonable simulation time and accuracy limit TCAD simula-

tions to few devices. Despite semiconductor physics complexity, microelectronic scientists can model physical phenomena in more complex structures as Metal-Oxide-Semiconductor Field-Effect-Transistors (MOSFET). Transistor models must calculate all its physical properties millions of times per second for even simple simulations. To be usable in CAD simulation tools, transistors are modeled using compact models through SPICE. Those models tradeoff between numerical simulation accuracy and speed. When mature technology models are available, such tradeoff is manageable and simulation results are as close as possible to measurements. However, this is hardly the case in novel nanometer scaled technologies. Research in microelectronics is driven by either pushing circuitry design in nanometer scaled technology (More-Moore) or circumventing semiconductor physics limitations in low-cost technologies (More-than-Moore) (Xiu, 2019). In both cases, designers should rely on the transferal of compact models from technology to SPICE CAD tools. Besides, some compact models are prohibitive since they would lack of physical phenomena coherence or require extensive hand calculations. In this scenario, having an accurate, fast, and reproducible simulation result are a challenge. The MOSLab.jl tool aims a fast, accurate and modular compact models to bridge the gap between semiconductor applied physics and microelectronics research in nanometer scaled technology. Developed in Julia, MOSLab.jl requires some college background in semiconductor physics and in microelectronics to offer flexible MOS compact models from semiconductor to transistor-level. The MOSLab.jl tool includes:

- 1) Material properties analysis of electronic characteristics using Fermi-Dirac distribution, Boltzmann-Maxwell distribution, and Blakemore distribution (at this stage limited to Silicon)
- 2) Numerical Simulations of MOS structures and MOS transistors in 1D and 2D powered by VoronoiFVM.jl
- 3) Implementation of industry and research standard MOSFET models (Pah-Sah, Brews, BSIM 6, PSP 103, UICM and EKV), with different mobility models.
- 4) Circuit Level Simulation including DC Operating Point, Transient analysis (powered by ModelingToolkit.jl), and AC Analysis (powered by ControlSystems.jl).

A. MOSLab Modeling and Simulation

Several examples using MOSLab.jl are freely available on the official website. Compact model implementation is explained in the user manual, as also installation procedure and the tool usage. From semiconductor to transistor-level modeling MOSLab.jl has been used in recent scientific communication (Martins PhD, 2021) and (Martins, 2022). For example, the following Julia code numerical simulate the band diagram of a MOS capacitor as depicted in Fig. 1(a), and then it calculates the surface potential using Boltzmann and Fermi distributions, as depicted in Fig. 1(b).

```
using MOSLab
using Plots
T = 300.0 # Temperature Kelvin
N_a = 5e18 # Bulk Doping in cm^-3
E_a = 0.044 # The ground state energy of the
            # acceptor in eV
t_ox = 4e-7 # oxide thickness in cm
t_si = 2.5e-5 # Silicon thickness
Vgv = 1.0 # Gate Voltage
PB = SemiconductorData(T,BoltzmanDist(),
    PSilicon(N_a,E_a))
MOS = MOSStructure(Alluminium(), SiO2(),
    PB,t_ox,t_si)
sol = PoissonVoronoi(Vgv,MOS,201)
BandDiagram(sol)
plot!(legend=:topright)
savefig("BandDiagram.png")
PBFermi = SemiconductorData(T,FermiDist(),
    PSilicon(N_a))
MOSFermi = MOSStructure(Alluminium(), SiO2(),
    PBFermi,t_ox,t_si)
plot(-2.3:0.05:2.3,x-> s(x,MOS),
    label="Boltzmann Distribution")
plot!(-2.3:0.05:2.3,x-> s(x,MOSFermi),
    label="Fermi Distribution",
    legend=:topleft)
xlabel!("Vg [V]")
ppsi = ylabel!("psi_s \ [V]")
savefig("psi_comp.png")
```

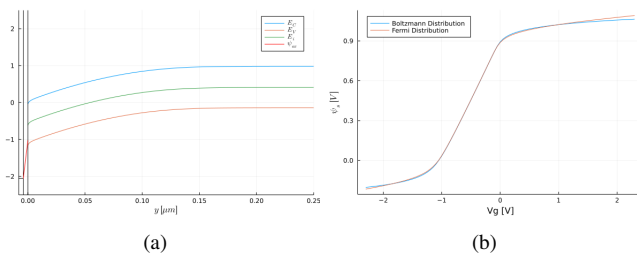


Fig. 1. Numerical simulations of MOS Capacitor, being (a) the band diagram, and (b) the surface potential (ψ_s) using Boltzmann and Fermi distributions.

III. IC-LAYOUT RENDER IN PYTHON

Graphic Design System (GDSII) from (Calma Company, 1987) is a common database file format to stream out integrated circuit (IC) masks, also called layout, before fabrication. Being an essential part of circuit development, designers are often familiarized in how to generate this binary file on computer-aided design (CAD) tools. This output file contains planar geometric shapes which represent physical 3D layers from a specific process design kit (PDK). Commercial PDK usually presents more than 50 layers, which are 2D visualized

in CAD tools. Reading a layout in those CAD tools is not a simple task, however high-quality graphical image is available in such tool. Thus, an IC designer is capable to understand the data representation and point out improvements for the IC. Nevertheless, IC documentation is often delivered in a lower quality 2D image depicted in a PDF file. Once the PDF is generated; the circuit layout becomes hardly readable even for experienced IC designers. Scientific communications require accurate and repeatable results to be considered prior publication. In microelectronics research field, a layout picture is mandatory. While illustrations should be a vectorial graph like, IC layout is often depicted from a low-quality bitmap obtained from a screenshot or an image saving through the CAD tool. In this scenario, having an accurate, readable, and reproducible layout result are a challenge. Most publications illustrate IC layouts in lower standards than the other illustration results, which hinders the required physical solution to address state-of-the-art IC performance. The tool icLayoutRender aims to a user-friendly transcription of a GDSII file in a pdf file. Developed in Python 3, the image rendering requires an input GDSII file **cellName.gds** and the PDK layer color file **LayerColor_PDK.map** to produce an output PDF file **cellName.pdf**. The GDSII layers will be rendered using the available PDK layer colors. Missing layers will be neglected. The tool icLayoutRender does

- 1) confirm the entered values;
- 2) convert the *.gds* to a *.tex* file;
- 3) call LuaTeX or pdflatex to generate a *.aux*, and *.pdf*

The tool icLayoutRender requires

- 1) Python3 installed with devel options;
- 2) gdspy, pandas, math, and GDSLAtexConverter (Vollmer, 2020) Python libraries installed;
- 3) TexLive for a Linux installation, or MikTEX for a Windows installation;
- 4) tikz LaTeX package to compile the output **.pdf* file.

A. IC LayoutRender Design Illustration

Several examples using icLayoutRender are freely available on the official website. Layer color file generation is also explained in the user manual, as installation procedure and the tool operation in Linux and Windows. The icLayoutRender image rendering tool has been used in recent scientific communication (Martins et al., 2021). The illustration readability is remarkable, and the authors invite the readers to compare Fig. 2(a) with 2(b), and Fig. 2(c) with 2(d) (Fonseca et al., 2017).

REFERENCES

- [1] Calma Company. (1987). *GDSII Stream Format Manual* (February). http://bitsavers.informatik.uni-stuttgart.de/pdf/calma/GDS_II_Stream_Format_Manual_6.0_Feb87.pdf
- [2] Ferreira, P. M., Martins, J. R. R. O., Mostafa, A., & Juillard, J. (2019). Process-Voltage-Temperature Analysis of a CMOS-MEMS Readout Architecture. *Proc. IEEE Design, Test, Integration & Packaging of MEMS/MOEMS*, 1–4. <https://doi.org/10.1109/DTIP.2019.8752699>
- [3] Fonseca, A. V., Khattabi, R. E., Afshari, W. A., Barúqui, F. A. P., Soares, C. F. T., & Ferreira, P. M. (2017). A Temperature-Aware Analysis of Latched Comparators for Smart Vehicle Applications. *Proc ACM IEEE Symp. Integr. Circuits Syst. Design*, 1–6. <https://doi.org/10.29292/jics.v13i1.8>
- [4] Martins, J. R. R. O., Alves, F., Ferreira, P. M., (2022). A General g_m/I_D Temperature-Aware Design Methodology Using 180 nm CMOS up to 250 °C. *J. Integr. Circuits and Syst.*, 17(1), 19. <https://doi.org/10.29292/jics.v17i1.552>
- [5] Martins, João R. R. O., Alves, F., Ferreira, P. M., (2021). IC-Layout Render: Image rendering tool for integrated circuit layout in Python. *Open Source Software*. <https://doi.org/10.5281/zenodo.5618268>
- [6] Martins, João R. R. O., Alves, F., Ferreira, P. M., (2021). MOSLab: From Semiconductor to Transistor-Level Modeling in Julia. *Open Source Software*. <https://doi.org/10.5281/zenodo.5589821>

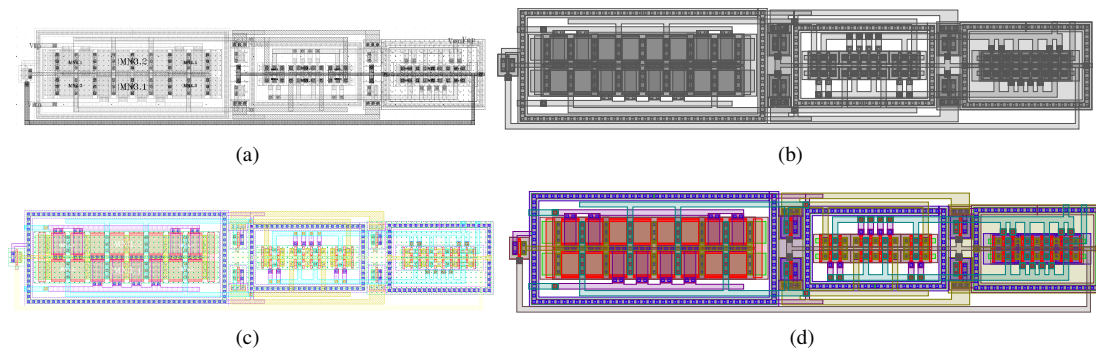


Fig. 2. Layout illustration of StrongArm latched comparator proposed in (Fonseca et al., 2017)(a) as its original illustration and (b) using IC layout render tool with gray scale color map, (c) in full-color version, and (d) using IC layout render tool with a full-color map.

- [7] Martins, J. R. R. O. (2021). A Temperature-Aware Framework for Analog Design Using a g_m/I_D Approach, *PhD. Thesis*, Université Paris-Saclay.
- [8] Martins, J. R. O. R., Alves, F., & Ferreira, P. M. (2021). A 237 ppm / °C L-Band Active Inductance Based Voltage Controlled Oscillator in SOI 0.18 μm . *Proc ACM IEEE Symp. Integr. Circuits Syst. Design*, 1–6. <https://doi.org/10.1109/SBCCI53441.2021.9529990>
- [9] Synopsys Technology Computer-Aided Design (TCAD). (2021). <https://www.synopsys.com/silicon/tcad.html>
- [10] Sze, S. M. (1981). *Physics of Semiconductor Devices* (2nd ed.). John Wiley Sons.
- [11] Vollmer, R. (2020). *GDSLatexConverter*. <https://github.com/Aypac/GDSLatexConverter>
- [12] XFAB Mixed-Signal Foundry Experts. (2019). *XH018 - 0.18 Micron Modular Analog Mixed HV Technology* (pp. 1–22). <https://www.xfab.com/technology/cmos/018-um-xh018/>
- [13] Xiu, L. (2019). Time Moore: Exploiting Moores Law from the Perspective of Time. *IEEE Solid-State Circuits Magazine*, 11(1), 3955. <https://doi.org/10.1109/MSSC.2018.2882285>