TCAD simulation I

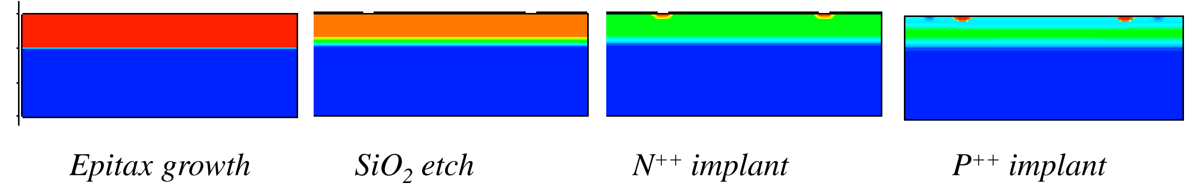
E. Giulio Villani

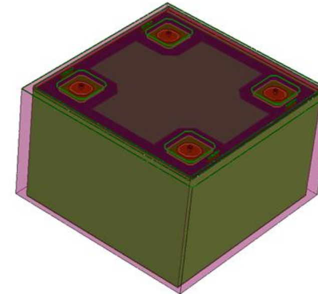
Overview

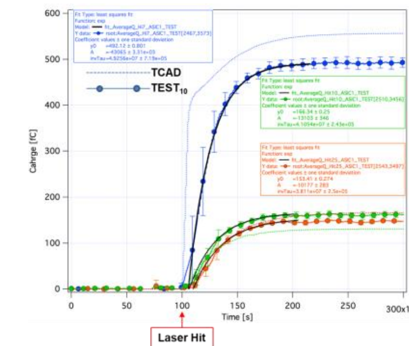
- **Introduction**
  - needs for TCAD simulations
  - Transport regimes and related equations
  - Discretization techniques: meshing
  - Discretization of semiconductor equations: Scharfetter-Gummel technique
- **Synopsys TCAD intro**
- **Synopsys TCAD simulation: Structure Editor (SDE)**
  - 2D simulation of PN junction

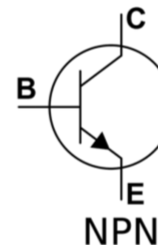
Introduction

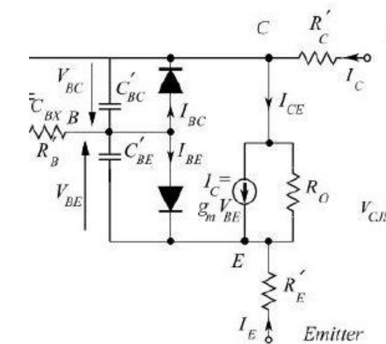
- TCAD (Technology Computer Aided Design) divides into three groups:
  - **Process simulation**, i.e. simulation of fabrication process steps (oxidation, implantation, diffusion…)
  - **Device simulation**, i.e. simulation of the thermal/electrical/optical behavior of electronic devices,(IV,CV, frequency response…)
  - **Device modeling**, i.e. creating compact behavioral models for devices for circuit simulation (SPICE, Cadence…)







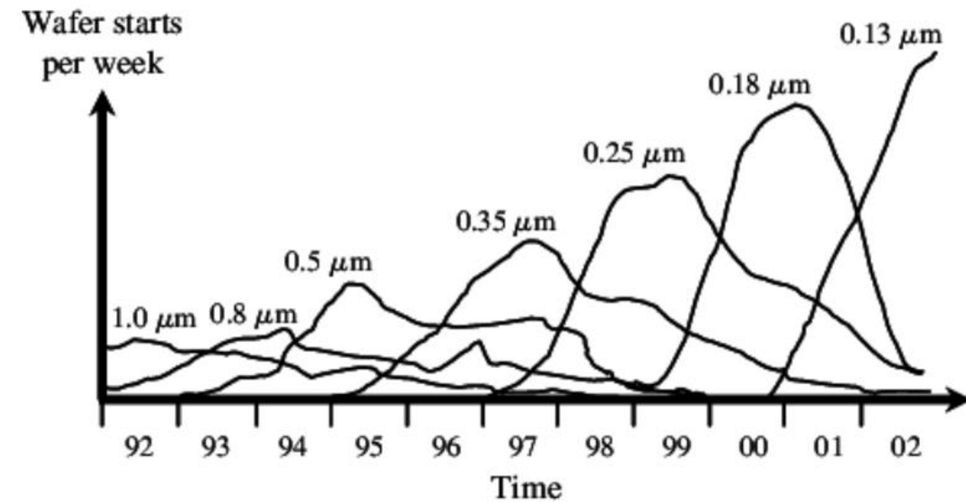




UNIVERSITY OF OXFORD

Introduction

- Reasons why TCAD simulations are needed:
  - Market demands cycle of design to production of 18 months or less. Typically 2-3 months required for wafer tape out implies short time for development
  - Reduce cost to run experiments on new devices and circuits



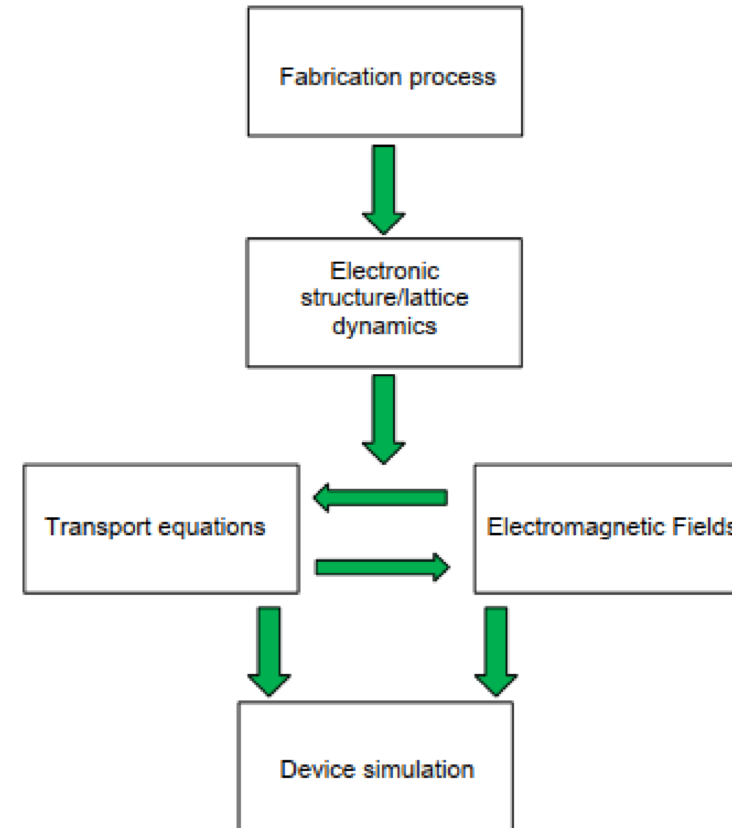
Shrinking product life cycles in semiconductor industry over time

Hochbaum, Dorit S. et al., Rating Customers According to Their Promptness to Adopt New Products., Operations Research, vol. 59, no. 5, p. 1171–83, 2011.

Introduction

- Main components of semiconductor device simulation include fabrication process, description of electronic structure, driving forces and transport phenomena

- The two kernels of semiconductor transport equations and fields that drive charge flow are coupled to each other and needs solving self-consistently



Transport regimes

- Usually only the quasi-static electric fields from the solution of Poisson's equation are necessary

- Transport regime in semiconductors depends on length scale

Modern Silicon technology already requires tools to describe transport in quantum regime [D. K. Ferry and S. M. Goodnick, Transport in Nanostructures, 1997]

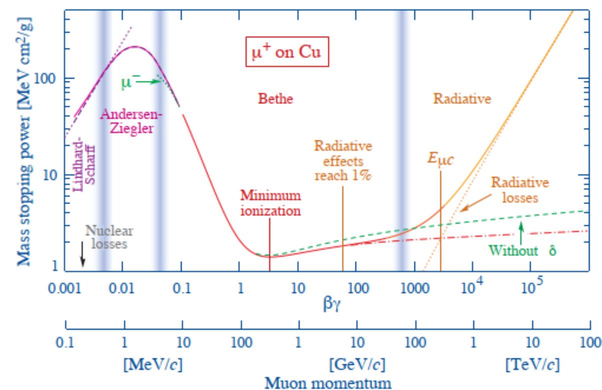
	$L << l_{e-ph}$			$L \sim l_{e-ph}$	$L >> l_{e-ph}$
	$L < \lambda$	$L < l_{e-e}$	$L >> l_{e-e}$		
Transport regime	Quantum	Ballistic	Fluid	Fluid	Diffusive
Scattering	Rare	Rare	e–e (Many), e–ph (Few)		Many
Model:					
Drift-diffusion					
Hydrodynamic	Quantum hydrodynamic				
Monte Carlo					
Schrödinger equation					
Green's function					
Applications	Nanowires, superlattices	Ballistic transistor	Present time ICs	Present time ICs	Older ICs

L: device length
l_{e-e}: electron-electron scattering length
l_{e-ph}: electron-phonon scattering length
λ: electron wavelength

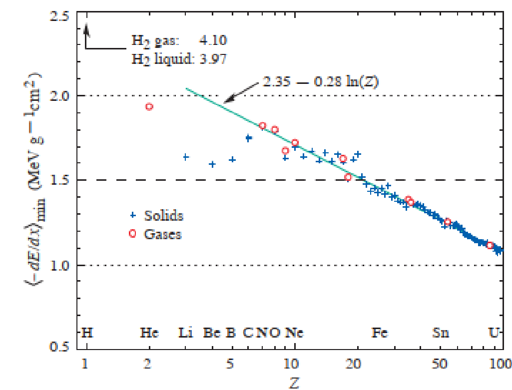
D. Vasileska, et al., Computational Electronics: Semiclassical and Quantum Device Modelling and Simulation, RC press, 2010,ISBN-13: 978-1-4200-6484-1.

Transport regimes

- Charge carrier dynamics in Si detectors usually does not require QM
- Semiclassical laws of motions apply
- Drift-diffusion (DD) equations are often valid, i.e. provided the electron gas is in thermal equilibrium with lattice temperature ($T_n = T_L$)



Mass stopping power (= −dE/dx) for positive muons in copper as a function of βγ = p/Mc



Mass stopping power at minimum ionization for the chemical elements.

$$R \sim h\frac{\gamma v}{< I >} \sim 10's\ [nm]\ Ionization\ radius\ for\ MIP\ in\ Si\ *$$

$$n = \frac{dE}{dx}\rho\frac{L}{E_{th}}\frac{1}{\pi R^2 L} \sim up\ to\ 10^{18}\ MIP\ charge\ density\ within\ Ionization\ radius$$

$$n^{-1/3} \sim 10[nm]$$

$$\lambda \leq \frac{h}{\sqrt{2meV_b}} \sim \begin{cases} 0.38\ [nm]@10\ V \\ 0.12\ [nm]@100\ V \end{cases} De\ Broglie\ wavelength\ of\ carriers$$

$$@\ full\ depletion$$

* **Nuclei And Particles, Emilio Segre** Published by W.A. Benjamin, Inc, New York, 1965

Drift diffusion model

- The semiconductor equations derived from 1st moment of Boltzmann Transport Equation (BTE) are referred to as **D**rift **D**iffusion (DD) model (see the notes)
- The model consists of Poisson's equation and PDE: continuity and current density equations for electrons and holes
- They express charge and momentum conservation
- Their self-consistent solutions are obtained via discretization, using finite element methods FEM

$$\partial_t f + \mathbf{u}\cdot\nabla_{\mathbf{r}} f + \frac{\mathbf{F}}{\hbar}\cdot\nabla_{\mathbf{k}} f = C[f] \qquad BTE$$

$$\mathrm{div}\,(\varepsilon\cdot\mathrm{grad}\,\psi) = q\cdot(n - p + N_A - N_D)$$

$$\mathrm{div}\,\mathbf{J}_n - q\cdot\frac{\partial n}{\mathrm{d}t} = q\cdot R$$

$$\mathrm{div}\,\mathbf{J}_p + q\cdot\frac{\partial p}{\mathrm{d}t} = -q\cdot R$$

$$\mathbf{J}_n = q\cdot n\cdot\mu_n\cdot\mathbf{E} + q\cdot D_n\cdot\mathrm{grad}\,n$$

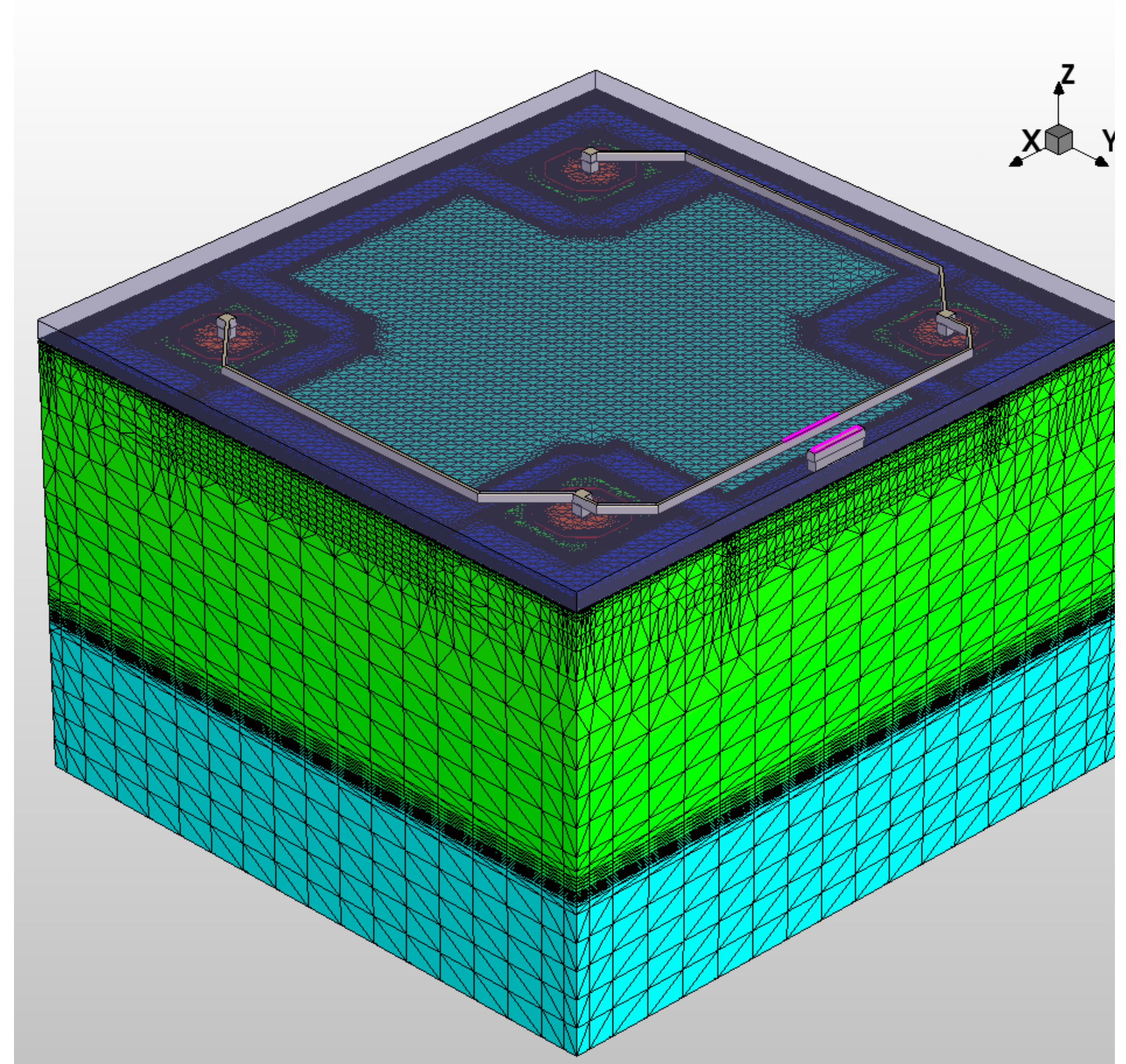
$$\mathbf{J}_p = q\cdot p\cdot\mu_p\cdot\mathbf{E} - q\cdot D_p\cdot\mathrm{grad}\,p$$

Discretization and meshing

- The device simulations process consists of two steps:

1. The test volume is obtained through grid generation ('mesh generation')

2. Solving the discretized differential equations using Finite-Boxes method (box integration method) . This method integrates PDEs over a test volume



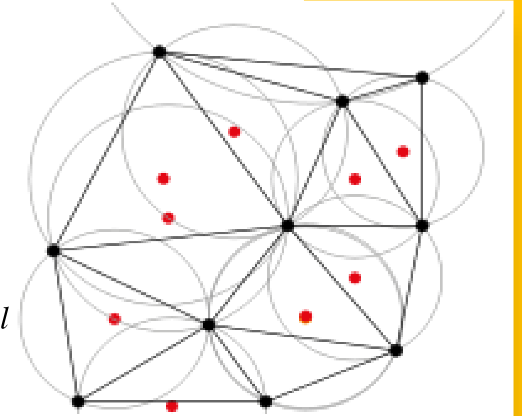
Discretization and meshing

- The meshing used in most finite elements methods (FEM) relies on Delaunay triangulations:

the interior of the circumsphere of each element contains no mesh vertices.

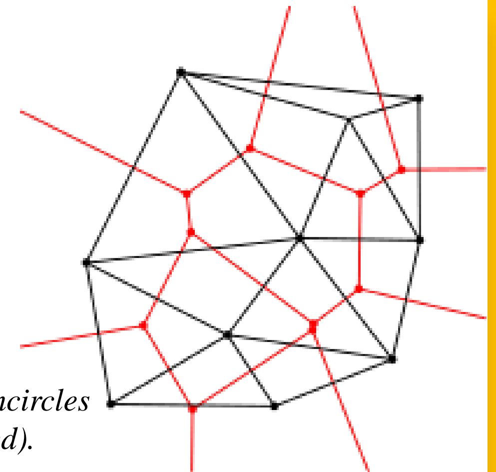
- The Delaunay triangulation of a discrete point set **P** in general corresponds to the dual graph of the Voronoi diagram for **P**

the set of all locations x closest to P_i than to any other point of the grid



The Delaunay triangulation with all the circumcircles and their centres

$$\mathcal{T}_{\mathrm{D},k}(\mathcal{P}_{\mathrm{D},k}) \Longleftrightarrow \exists\, \mathbf{x} \quad \mathbf{x} \in \Omega \;\; \wedge \;\; |\mathbf{x}-\mathbf{x}_i| = |\mathbf{x}-\mathbf{x}_j| \;\; \forall\; P_i, P_j \in \mathcal{P}_{\mathrm{D},k} \;\; \wedge \;\; |\mathbf{x}-\mathbf{x}_i| < |\mathbf{x}-\mathbf{x}_k| \;\; \forall\; P_i \in \mathcal{P}_{\mathrm{D},k}, P_j \notin \mathcal{P}_{\mathrm{D},k}$$



Connecting the centres of the circumcircles produces the Voronoi diagram (in red).

$$\Omega_i = \{\mathbf{x} \;|\; |\mathbf{x}-\mathbf{x}_i| \leq |\mathbf{x}-\mathbf{x}_j| \;\; \forall\; i \neq j,\; \mathbf{x} \in \Omega\}$$

Discretization and meshing

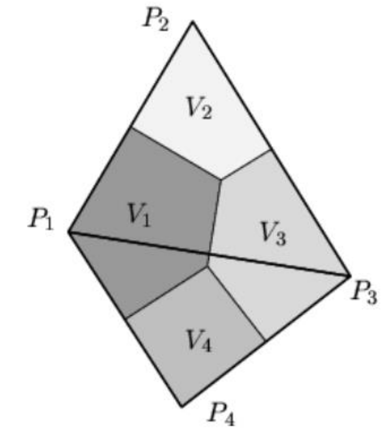
- Correct Delaunay triangulation* guarantees element-volume conservation, important in many problems (diffusion, charge generation, et cetera)

- Delaunay triangulation maximizes the minimum angle of the triangle

*A. Okabe, B. Boots, and K. Sugihara, Spatial Tessellations, John Wiley and Sons Ltd, 1992.

Voronoi boxes do not overlap (each circumcircle does not include a point of another triangle). Each can be uniquely assigned to its corresponding grid points.

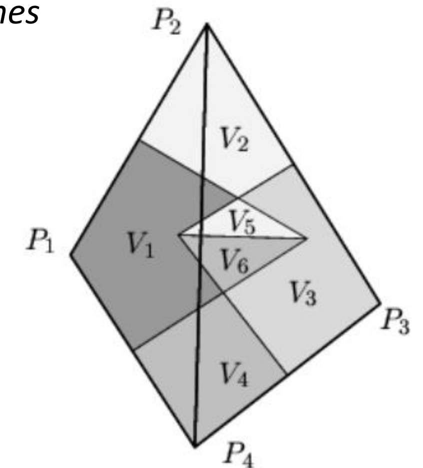
$$P_i = V_i$$



Voronoi boxes do overlap (each circumcircle does include a point of another triangle). Each cannot be uniquely assigned to its corresponding grid points. Wrong volumes calculated

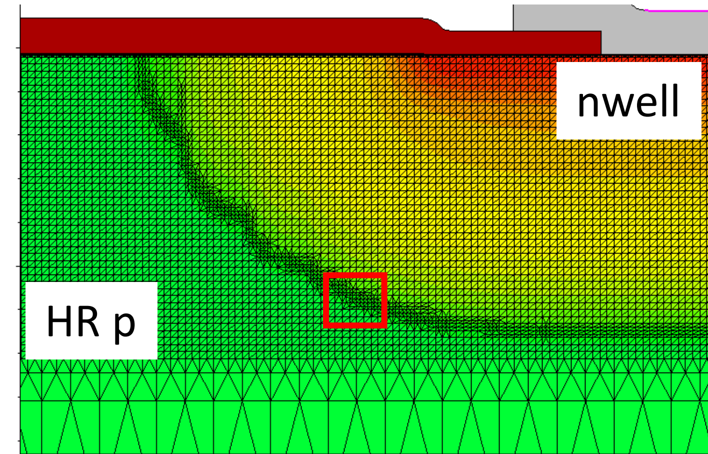
$$P_1 = V_1 + V_5 + V_6$$

$$P_3 = V_3 + V_5 + V_6$$



Discretization and meshing

- Discretization of equations imposes some constraints on spatial and temporal mesh size
- Mesh size should be smaller than **Debye length** (i.e. the characteristic length for screening of field by charges) where charge variations in space have to be resolved



$$L_D = \sqrt{\frac{\varepsilon_s k_B T}{e^2 N}} \; Debye\ length$$

$$N = 10^{13}[cm^{-3}]: L_D \approx 1.3\ [um]@T = 300\ [K]$$
$$N = 10^{17}[cm^{-3}]: L_D \approx 13\ [nm]@T = 300\ [K]$$
$$N = 10^{19}[cm^{-3}]: L_D \approx 1.3\ [nm]@T = 300\ [K]$$

Discretization and meshing

- Also temporal 'mesh' size should be smaller than the **dielectric relaxation time** t_{dr} (i.e. time it takes to charge fluctuations to decay under the field they produce)
- Time interval smaller than t_{dr} might give unrealistic transient results ('oscillations' in estimated transient currents)

$$\tau_{dr} \sim \frac{\varepsilon_s}{eN\mu} \textit{Dielectric relaxation time}$$

$$N = 10^{13}[cm^{-3}], \mu_n \approx 1400 \left[[cm^{-3}V^{-1}s^{-1}\right]@T = 300\ [K]: \tau_{dr} \approx 400\ [ps]$$

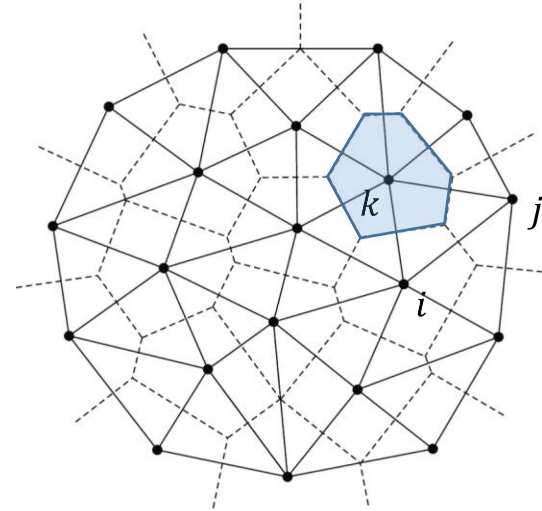
$$N = 10^{15}[cm^{-3}], \mu_n \approx 1350 \left[[cm^{-3}V^{-1}s^{-1}\right]@T = 300\ [K]: \tau_{dr} \approx 4.8\ [ps]$$

$$\frac{\partial \Delta n}{\partial t} = -\frac{\Delta n(t=0)}{t_{dr}}$$

$$\Delta n\ (\Delta t) = \Delta n\ (0) - \Delta t \frac{\Delta n(0)}{t_{dr}}$$

Box integration method

- Discretization of Poisson's and continuity equations is done via Box Integration method
- The LHS of equations is transformed via Gauss' theorem and integrated over a Voronoi box Ω_k of point P_k



$$\nabla \cdot \varepsilon_S \nabla \varphi = e(n - p - N_D + N_A) \quad \equiv \int D \cdot dS = \int \rho dV$$

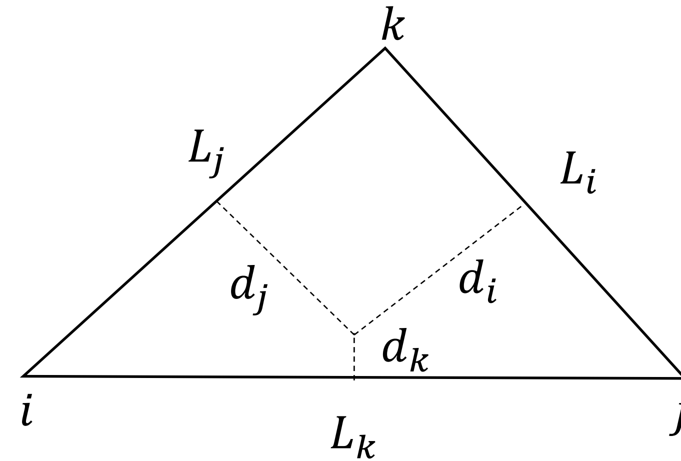
$$\nabla \cdot J_n - q\frac{\partial n}{\partial t} = qR$$

$$\nabla \cdot J_p + q\frac{\partial p}{\partial t} = -qR$$

Box integration method

- Example of **Poisson's equation discretization**

- Assume that the electric potential is linearly varying over each elementary domain



$$\nabla \cdot \varepsilon_S \nabla \varphi = e(n - p - N_D + N_A) \equiv \int D \cdot dS = \int \rho dV$$

i, j, k: *nodal indices*

$\mathbf{L_i}, \mathbf{L_j}, \mathbf{L_k}$: *side vectors*

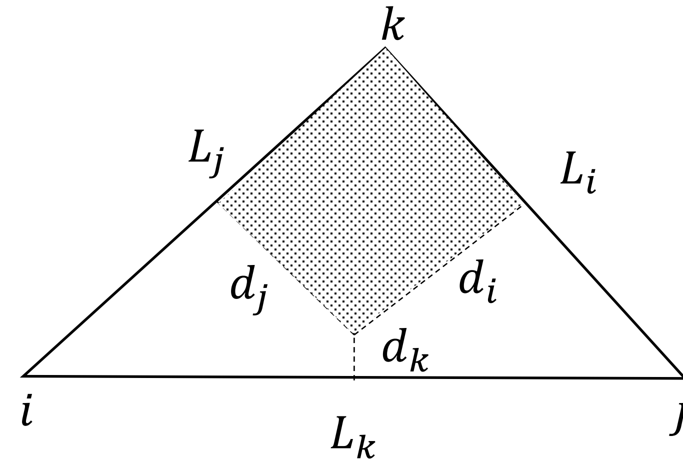
L_i, L_j, L_k: *magnitude side vectors*

$u := \frac{e}{k_B T} \varphi$: *normalized potential*

UNIVERSITY OF OXFORD

Box integration method

- Components of D vector along sides $L_{i,j,k}$
- Flux of D vector associated to node k:
- Discretization of RHS is obtained by multiplying the node value of charge by the area of the portion of the Voronoi box



$$\frac{k_B T}{e}\varepsilon_s\frac{1}{L_i}(u_j - u_k)$$
$$\frac{k_B T}{e}\varepsilon_s\frac{1}{L_j}(u_k - u_i)$$
$$\frac{k_B T}{e}\varepsilon_s\frac{1}{L_k}(u_i - u_j)$$

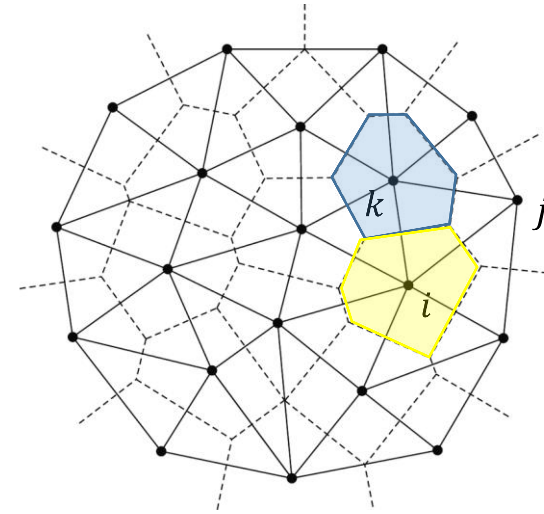
$$\frac{k_B T}{e}\varepsilon_s[\frac{d_i}{L_i}(u_j - u_k)+\frac{d_j}{L_j}(u_i - u_k)] \quad Flux\ of\ D$$

UNIVERSITY OF OXFORD

Box integration method

- Summing over all points P_k of Voronoi boxes

- Same approach to discretize the **continuity equations** for electrons and holes



$$\sum_k D_{ik} A_{ik} = \rho_k V_k$$

A_{ik}: *area of K − Vbox*
D_{ik}: *projection along grid line*

$$\sum_k J_{n,ik} A_{ik} = q(R_k + \frac{d}{dt} n_k) V_k$$

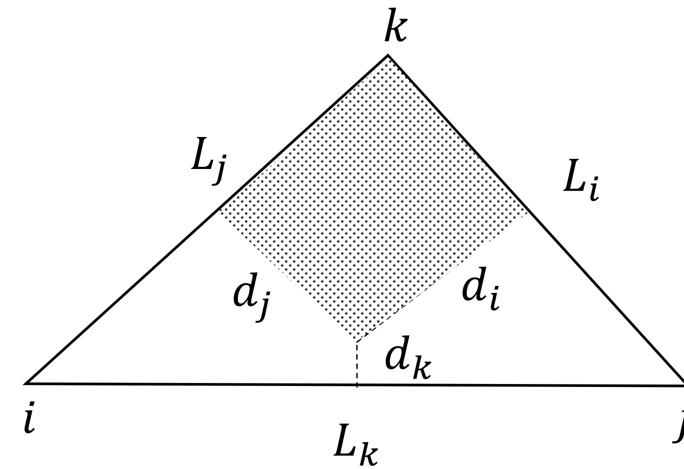
$$\sum_k J_{p,ik} A_{ik} = -q(R_k + \frac{d}{dt} p_k) V_k$$

UNIVERSITY OF OXFORD

Scharfetter-Gummel discretization

- In case of no strong generation-recombination the current density varies little within each domain
- Even so, this implies an exponential dependence of electron / hole density n/p with position along grid's edge
- Using previous discretization method would require very dense mesh: the Scharfetter-Gummel technique* includes such dependence, requiring less grid points

*D. L. Scharfetter and H. K. Gummel, "Large-signal analysis of a silicon read diode oscillator," IEEE Trans. Electron Devices, vol. ED-16, pp. 64–77, Jan. 1969



$$\mathbf{J}_n = q \cdot n \cdot \mu_n \cdot \mathbf{E} + q \cdot D_n \cdot \mathrm{grad}\, n$$

$$\mathbf{J}_p = q \cdot p \cdot \mu_p \cdot \mathbf{E} - q \cdot D_p \cdot \mathrm{grad}\, p$$

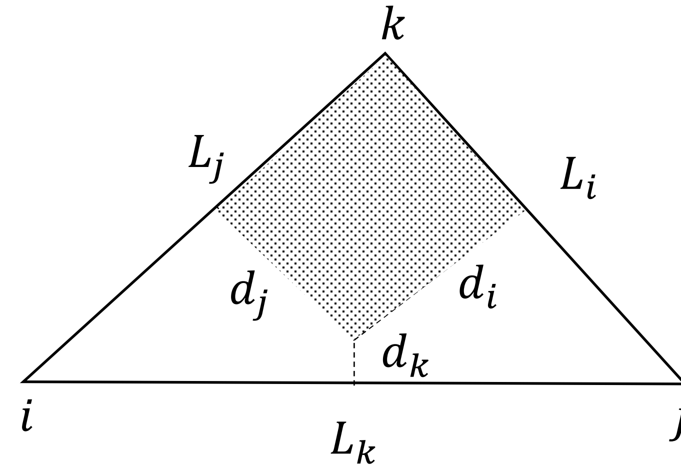
$$u := \frac{e}{k_B T}\varphi \qquad D_n := \frac{k_B T}{e}\mu_n$$

$$J_n := qD_n[\nabla n - n\nabla u] \qquad \textit{from } J_n$$

$$J_{nk} := qD_n[\frac{dn}{dl_k} - n\frac{du}{dl_k}] \qquad \textit{projection along } L_k \sim \textit{constant}$$

Scharfetter-Gummel discretization

- Assume u varies linearly along the edge and current density $J_n \simeq$ constant over the domain
- Define the reduced current j_{nk} and assume an average diffusion D_{nk} along the edge
- Obtain first order equation in n along the edge



$$u = \frac{u_j - u_i}{L_k} l_k + u_i = a_k l_k + u_i$$

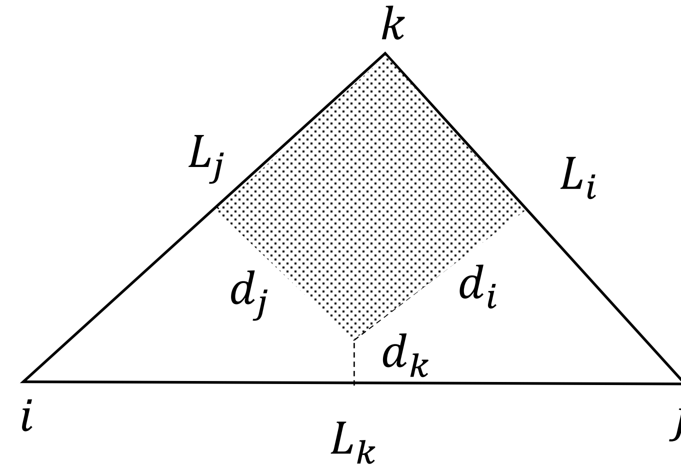
$$j_{nk} := \frac{J_{nk}}{qD_{nk}}, \qquad D_{nk} := < D_{ni}, D_{nj} >$$

$$j_{nk} = \frac{dn}{dl_k} - n a_k$$

Scharfetter-Gummel discretization

- Integrate from node i to node j, i.e. for l_k=[0, L_k]

- Obtain expression relating potential and carriers concentration



$$\int_0^{L_k} \exp(-a_k l_k)\, j_{nk} = \int_0^{L_k} \exp(-a_k l_k)\left(\frac{dn}{dl_k} - n a_k\right) d\, l_k$$

$$= \int_0^{L_k} \frac{d}{dl_k}(\exp(-a_k l_k) n) dl_k$$

$$j_{nk} \frac{1}{a_k}(1 - \exp(\, -u_{ji}) = \exp(-u_{ji}) n_j - n_i$$

$$j_{nk} = a_k(\frac{n_j}{\exp(u_{ji})-1} + \frac{n_i}{\exp(-u_{ji})-1})$$

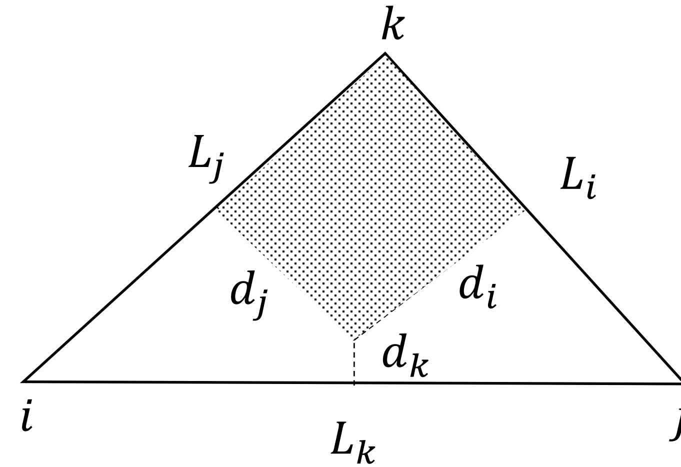
UNIVERSITY OF OXFORD

Scharfetter-Gummel discretization

- Obtain the flux of current density relative to node k

- Scharfetter –Gummel discretization requires less fine mesh as the exponential dependence of carriers concentration is included in the discretization scheme

- It also depends on boundary values, i.e. 2D and 3D cases can be reduced to local 1D cases



$$j_{nk} = \frac{1}{L_k}\left(\frac{u_{ji}n_j}{\exp(u_{ji})-1} - \frac{u_{ij}n_i}{\exp(u_{ij})-1}\right)$$

$$j_{nk} = \frac{1}{L_k}\left(B(u_{ji})n_j - B(u_{ij})n_i\right) \quad \textit{Bernoulli function } B(x) = \frac{x}{\exp(x)-1}$$

calculate the flux of j_{nk} *to get the charge at k*

$$qD_{ni}\frac{d_i}{L_i}\left(B(u_{jk})n_j - B(u_{kj})n_k\right) + qD_{nj}\frac{d_j}{L_j}\left(B(u_{ik})n_i - B(u_{ki})n_k\right) = q_k$$

Synopsys TCAD

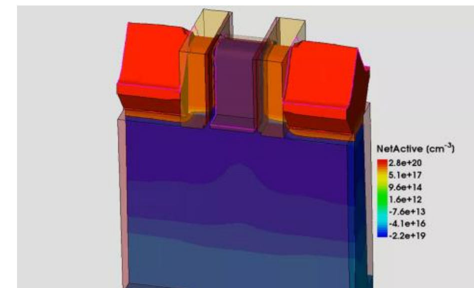
E. Giulio Villani

TCAD Synopsys Simulation

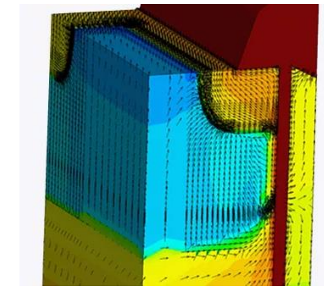
- Synopsys Sentaurus TCAD [*] allows the complete simulation of semiconductor devices

- Fabrication process, including epitaxial growth, oxidation, implantation, etching, diffusion, metallisation can be simulated

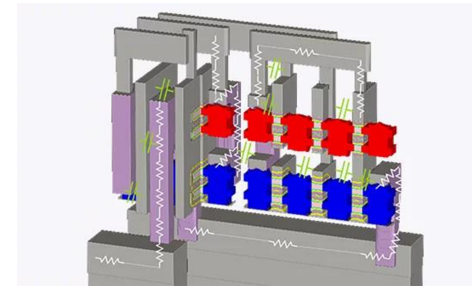
- Full electrical, thermal and optical simulations are possible, including IV/CV/noise, Charge transport and response to radiation



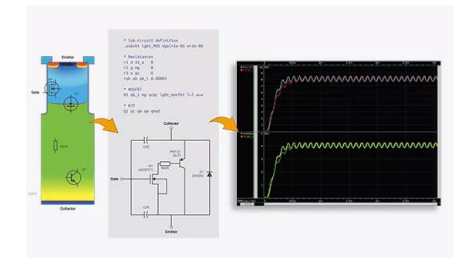
Process simulation



Device simulation



Device interconnect simulation



SPICE parameters extraction

[*] https://www.synopsys.com/manufacturing/tcad.html

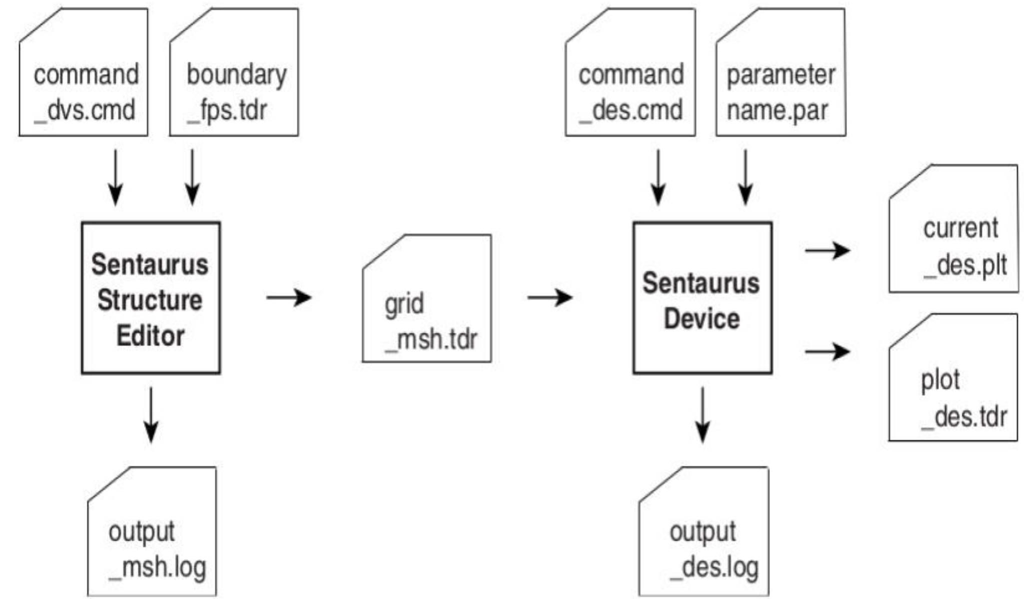
TCAD Synopsys Simulation

- The description of the devices can be done either **graphically** or via **scripts**, using code compatible with Tool Command Language [TCL]

- The various individual simulation tools are arranged on a Workbench (SWB) in a pipeline fashion

- Default models and parameters for fabrication and physics of most semiconductor devices are available and can be fully customised by the user

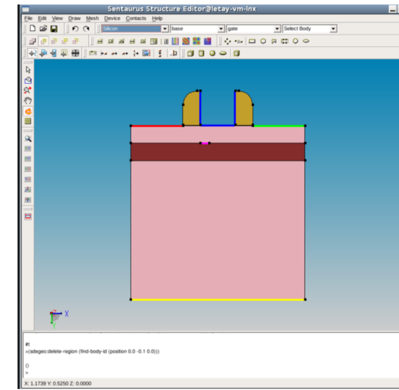
GL_Implants_Energy_KeV	BackCont		HIT		Temp	AvalMod	BETAE	BETAH	RGuard	OxchargeConc	OxIntchargeConc	RadiationFluence	RadiationModel	
								1	1e15	1e11	1.20e11	0	24GeVp+	--
							1	1.25	1e15	1e11	1.20e11	0	24GeVp+	--
								1.50	1e15	1e11	1.20e11	0	24GeVp+	--
								1	1e15	1e11	1.20e11	0	24GeVp+	--
180	0	--	0	--	300	Okuto	1.25	1.25	1e15	1e11	1.20e11	0	24GeVp+	--
								1.50	1e15	1e11	1.20e11	0	24GeVp+	--
								1	1e15	1e11	1.20e11	0	24GeVp+	--
							1.50	1.25	1e15	1e11	1.20e11	0	24GeVp+	--
								1.50	1e15	1e11	1.20e11	0	24GeVp+	--



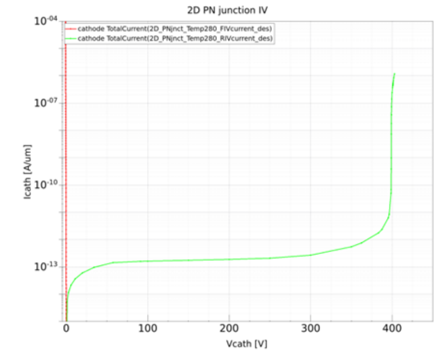
Cross section of simulated LGAD (top) and PIN diode (bottom)

TCAD Synopsys Simulation

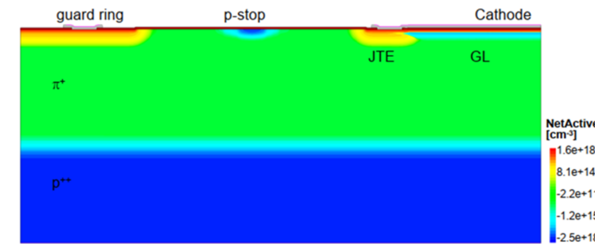
- The simulation tools we'll use are:
  - Sentaurus Device Editor (SDE)
    - structure definition
  - Sentaurus Device (SDEVICE)
    - device electrical simulation
  - Sentaurus Visual (SVISUAL)
    - visualisation tool
  - Sentaurus Process (SPROCESS)
    - fabrication process simulation



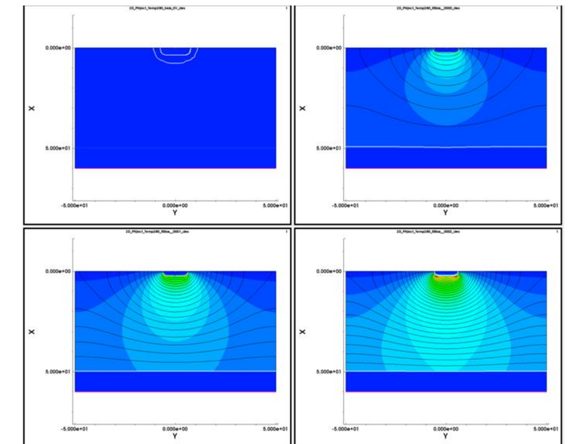
SDE GUI



SDEVICE results (.plt) plotted with SVISUAL



SPROCESS device

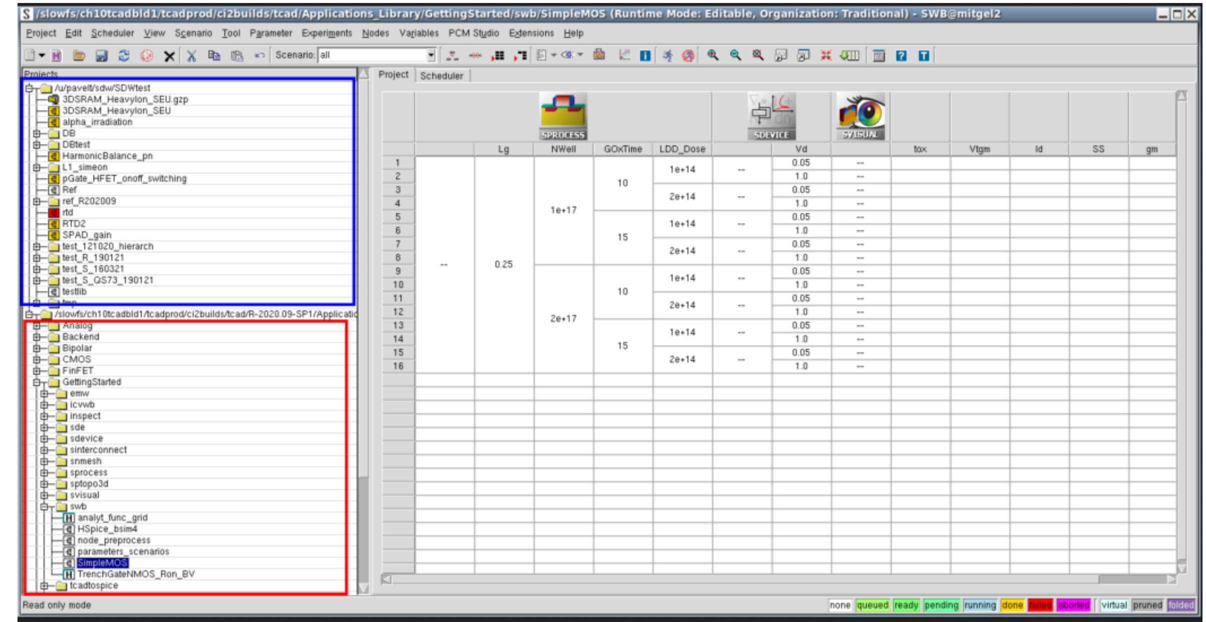


SDEVICE results (.tdr) plotted with SVISUAL

TCAD Synopsys Simulation

- Sentaurus Workbench (**SWB**) is the graphical front end that integrates TCAD Sentaurus simulation tools into one environment.
- Simulations are organized comprehensively into projects. **SWB** automatically manages the information flow, which includes preprocessing user input files, parameterizing projects, setting up and executing tool instances, and visualizing results.

- **SWB** allows you to define parameters and variables to run comprehensive parametric analyses. The resulting data can be used with statistical and spreadsheet applications.

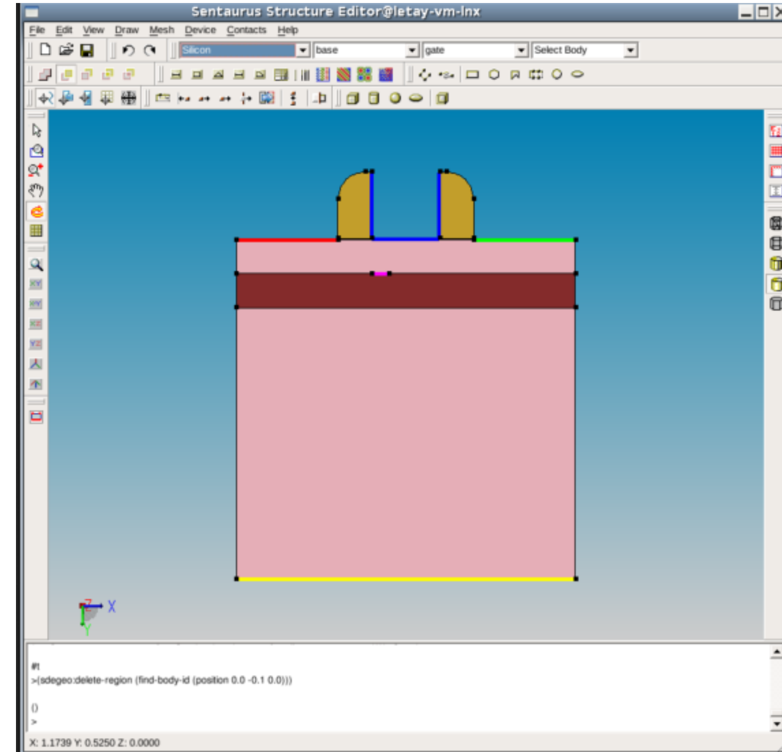


To start SWB:

swb &

TCAD Synopsys Simulation

- Sentaurus Structure Editor (**SDE**) is a structure editor for 2D and 3D device structures. Its operational modes are 2D structure editing and 3D structure editing.
- From the user interface, 2D and 3D device models are created geometrically, using 2D or 3D primitives such as rectangles, polygons, cuboids, cylinders, and spheres. Rounded edges are generated by filleting, 3D edge blending, and chamfering. Complex shapes are generated by intersecting primitive elements.
- The user interface of **SDE** features a command-line window, in which **SDE** prints script commands corresponding to the user interface operations.



To start SDE:

sde &

TCAD Synopsys Simulation

- **Sentaurus Device** is a numeric semiconductor device simulator, capable of simulating the electrical, thermal, and optical characteristics of various semiconductor devices.
- It simulates 1D, 2D, and 3D device behaviour over a wide range of operating conditions, including mixed-mode circuit simulation, combining numerically simulated devices with their compact modeling, which is performed on a SPICE-based circuit simulation level.

1 **File Section**: input/output files

2 **Electrode Section**: electrode definition, matching those in the input grid

3 **Physics Section**: physics models to use in the simulation

4 **Plot Section**: variables to plot

5 **Math Section**: solvers

6 **Solve Section**: what to solve (IV,CV, Charge injection)

A typical command file of Sentaurus Device consists of several sections (or statement blocks), with each section executing a relatively independent function. The default extension of the command file is _des.cmd, for example, pp1_des.cmd. To start: **sdevice**

UNIVERSITY OF OXFORD

TCAD Synopsys Simulation

- Sentaurus Process is a highly flexible multidimensional process modeling environment. It constitutes a solid base for process simulation.
- Calibrated to a wide range of the latest experimental data using proven calibration methodology, Sentaurus Process offers predictive capabilities for modern silicon and nonsilicon technologies.

Command file (*.cmd)
This is the main input file for Sentaurus Process. It contains all the process steps and can be edited.

Log file (*.log and *.out)
Sentaurus Process generates a log file when it is run. Both files contain information about each processing step.

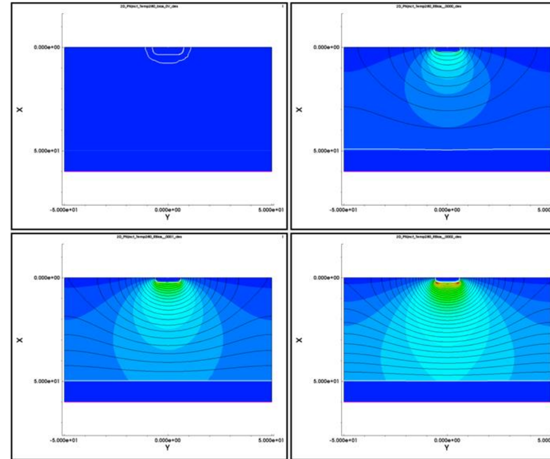
TDR boundary file (*_bnd.tdr)
This file stores the boundaries of the structure without the bulk mesh or fields.
TDR grid and doping file (*_fps.tdr)
This file stores the boundaries of the structure with the bulk mesh or fields
XGRAPH file (*.plx)
This file saves the 1D distributions of the doping concentration or other fields in a specified cross section.
To start: **sprocess**

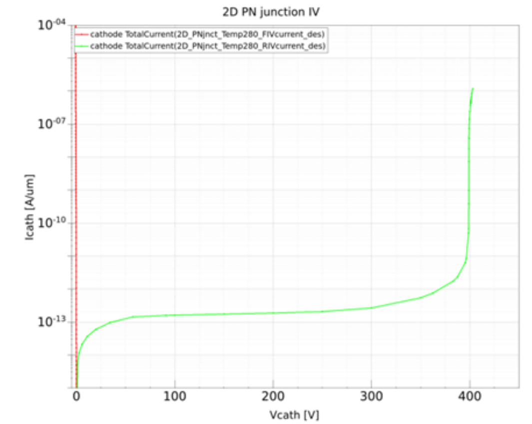
TCAD Synopsys Simulation

- Sentaurus Visual is advanced visualization software developed by Synopsys based on the Visualization Toolkit.

- Sentaurus Visual is used to visualize results created by TCAD simulation tools in one, two, and three dimensions.

- Sentaurus Visual provides tools to probe data fields, to make cutlines and cutplanes, to perform analysis on data fields, and to export data in ASCII format for further analysis by third-party software.





Two file types can be loaded into Sentaurus Visual:

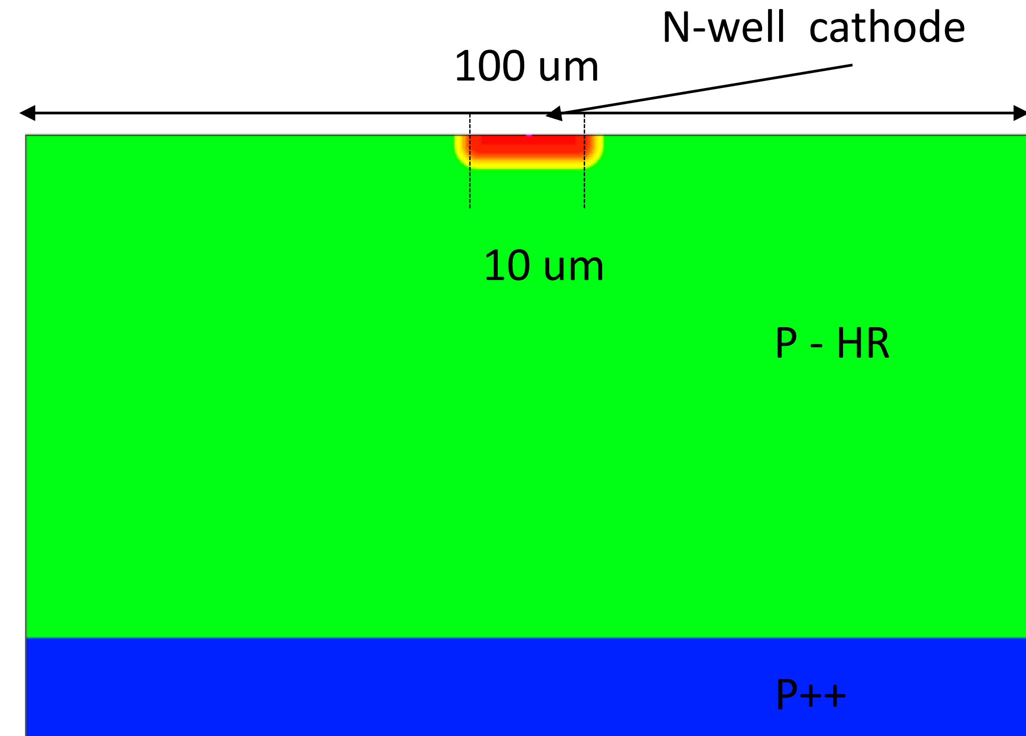
- Files with the *.tdr extension describe a device structure, its mesh, and data fields stored on the mesh.
- Files with the *.plt extension store terminal characteristics produced by the device simulator and are used to create I–V curve plots. To start: **svisual**

TCAD simulation

Structure Editor (SDE)

TCAD Synopsys Simulation

- First example: 2D pn junction using **SDE: 2D_PNjnct_SDE**
  - **INTERACTIVE using the GUI**
  - **BATCH using the script mode:** the scripting language of SDE is based on Scheme, a LISP-like programming language that differs significantly from most widely used programming languages. A working knowledge of Scheme is needed to create SDE scripts for parameterized devices.
  - https://docs.scheme.org/



TCAD Synopsys Simulation

- First example: 2D pn junction using **SDE: 2D_PNjnct_SDE**

```
(sde:clear)

;DEFINES THE GEOMETRY OF THE DEVICE
(sdegeo:create-rectangle (position 0 -50 0)  (position 60 50 0) "Silicon" "bulk")
(sdedr:define-refeval-window "substrate" "Rectangle"  (position 50 -50 0)
(position 60 50 0))

(sdedr:define-refeval-window "nwell_central" "Rectangle"  (position 0 -5 0)
(position 1 5 0))

;DEFINES THE DOPING PROFILES
(sdedr:define-constant-profile "ConstantProfileDefinition_subs"
"BoronActiveConcentration" 5e18)
(sdedr:define-constant-profile-placement "ConstantProfilePlacement_subs"
"ConstantProfileDefinition_subs" "substrate")
(sdedr:define-constant-profile "ConstantProfileDefinition_bulk"
"BoronActiveConcentration" 1e+13)
(sdedr:define-constant-profile-region "ConstantProfilePlacement_bulk"
"ConstantProfileDefinition_bulk" "bulk")


(sdedr:define-analytical-profile-placement "AnalyticalProfilePlacement_Cath"
"AnalyticalProfileDefinition_nwell" "nwell_central" "Positive" "NoReplace"
"Eval")
(sdedr:define-gaussian-profile "AnalyticalProfileDefinition_nwell"
"PhosphorusActiveConcentration" "PeakPos" 0  "PeakVal" 1e17 "Length" 1
"Gauss"  "Factor" 0.8)
```

TCAD Synopsys Simulation

- First example: 2D pn junction using **SDE: 2D_PNjnct_SDE**

```
;DEFINES THE REFINEMENTS FOR EACH REGION
(sdedr:define-refinement-size "RefinementDefinition_1" 1 1 1 0.1 0.1 0.1 )
(sdedr:define-refinement-placement "RefinementPlacement_bulk"
"RefinementDefinition_1" (list "region" "bulk" ) )
(sdedr:define-refinement-function "RefinementDefinition_1" "DopingConcentration"
"MaxTransDiff" 1)

(sdedr:define-refinement-size "RefinementDefinition_1" 1 1 1 0.1 0.1 0.1 )

(sdedr:define-refinement-placement "RefinementPlacement_bulk"
"RefinementDefinition_1" (list "region" "bulk" ) )
(sdedr:define-refinement-function "RefinementDefinition_1" "DopingConcentration"
"MaxTransDiff" 1)


;DEFINES THE CONTACTS
(sdegeo:define-contact-set "cathode" 4  (color:rgb 1 0 0 ) "##")

(sdegeo:define-contact-set "substrate" 4  (color:rgb 1 0 0 ) "##")

(sdegeo:set-contact (list (car (find-edge-id (position 60 -50 0)))) "substrate")

(sdegeo:insert-vertex (position 0 -0.25  0))

(sdegeo:insert-vertex (position 0 0.25  0))

(sdegeo:set-contact (list (car (find-edge-id (position 0.0 0 0)))) "cathode")
```

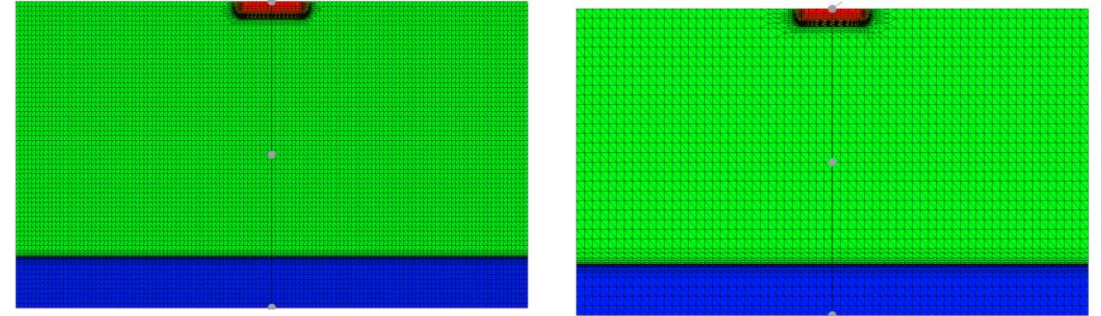
TCAD Synopsys Simulation

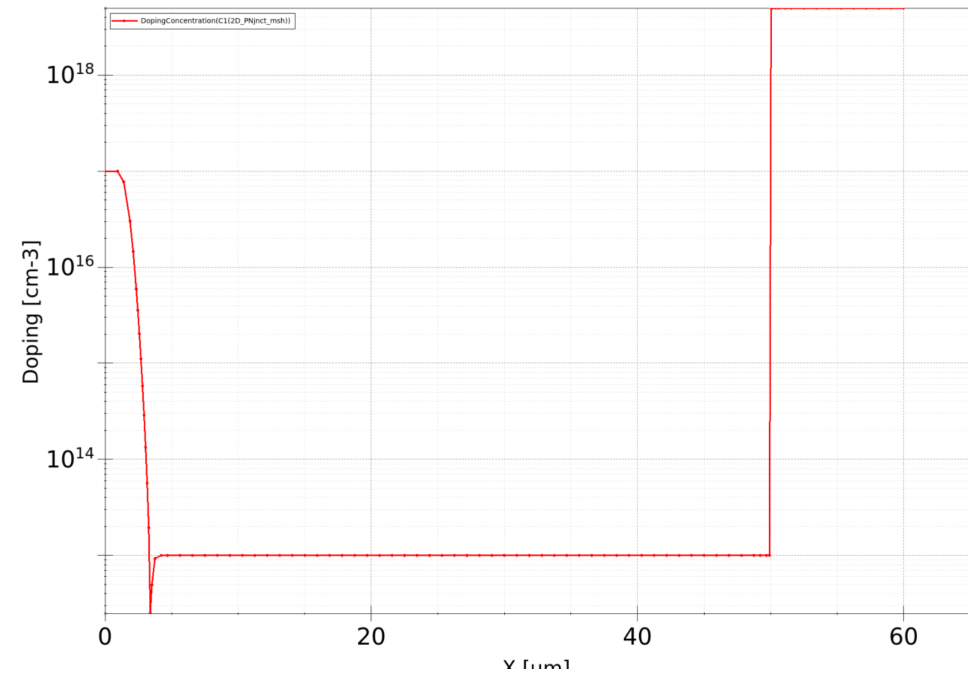
- First example: 2D pn junction using **SDE: 2D_PNjnct_SDE**

```
;SAVES THE MODEL
(sde:save-model
"/opt/ppd/fullcad/egv73/heplnm093/STDB/Lectures/2D_PNjnct_SDE/2D_PNjnct"
)
(sde:set-project-name
"/opt/ppd/fullcad/egv73/heplnm093/STDB/Lectures/2D_PNjnct_SDE/2D_PNjnct"
)

(sdesnmesh:iocontrols "outputFile"
"/opt/ppd/fullcad/egv73/heplnm093/STDB/Lectures/2D_PNjnct_SDE/2D_PNjnct"
)


; MAKES THE MESH
(sde:build-mesh ""
"/opt/ppd/fullcad/egv73/heplnm093/STDB/Lectures/2D_PNjnct_SDE/2D_PNjnct"
)
```

TCAD Synopsys Simulation

- Visualize the resulting mesh using **SVISUAL**
  - **Re-mesh the structure changing the p – doping**
  - **Visualise the meshing**





TCAD and simulation I

Thank you

giulio.villani@stfc.ac.uk

- Introduction to simulation
- Needs and transport regimes
- Meshing and discretization. Intro to DD model discretization. SG method
- example of TCAD simulations: SDE in interactive and batch mode, 2D PN junction