EE735
MSL

Assignment 5: Device Simulation in TCAD

06 Sept.2023

Faculty In charge
Prof. Saurabh Lodha

Course TA
Hemant Kumar
hemantkumar.svits@gmail.com

Outline

❑Assignment Questions

❑Introduction to CAD

❑ Why TCAD

❑ Mobaxterm

❑ Gateway of TCAD

❑ Know your SWB

❑ Working with SDE

❑ Working with SDEVICE

❑How to Read code in TCAD

Assignment-5

Deadline: 27 Sept 2023 @11:59Pm

Q1 Consider a Si bar of length 0.1µm and thickness 0.1µm Assume it is doped N-type with a density of 1016 cm-3. Simulate I-V (voltage: 0 to 3V) for the following mobility models:

a. Constant Mobility
b. Doping Dependence
c. High-field Saturation

Now do the following for each of the above three cases :

i) Plot TCAD simulated device Structure showing Mesh points. [2 marks]
ii) Simulated Band diagram (TCAD as well as Manual drawing) in equilibrium [3 marks]
iii) Plot I-V characteristics of the device and make an observation with appropriate comment if ant to these various Models plot [5]

Assignment-5

Q2. Let us start with simple design of a uniform resistor in TCAD and see its characteristics

Consider a Si bar of length 10 μm and the width being 2μm. Assume it is doped P-type with a density of 10e+18 cm$^{-3.}$ Apply bias to left contact and simulate I-V (voltage: 0 to 3 V) for the following conditions:
a. both contacts are ohmic
b. left contact is ohmic and right contact is Schottky (φB = 0.3 eV)

Now do the following for each of the above three cases :

i) Plot TCAD simulated device Structure showing Mesh points. [5 marks]

ii) Band diagram (TCAD as well as Manual drawing) in equilibrium [5 marks]

iii) I-V characteristic of device and put a qualitative analysis for the observed plot. [5 marks]

*(**Note:** φB is the barrier height (the difference between the contact work-function and the electron affinity of the semiconductor in n-type semiconductors, or the difference between the band gap and the barrier as defined for n-type semiconductors in the case of p-type semiconductors))*

Assignment 5

Q3 Consider a Si based two terminal NPN device as shown in following figure with indicated dimension and doping concentration

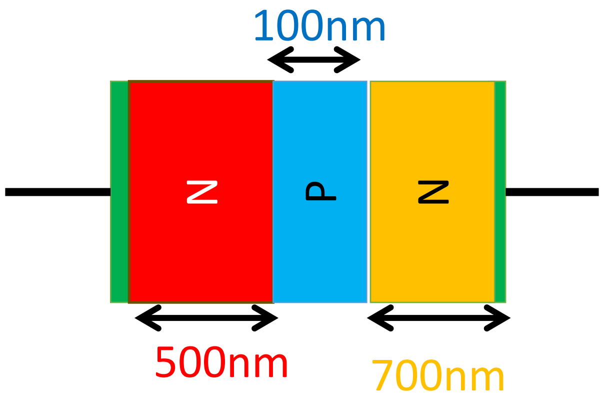
Simulate the above structure in TCAD and do the following:

i) Plot Device Structure showing mesh points.

ii) Plot the Doping concentration, Donar Concentration, and acceptor concentration

iii) Plot the space charge density, Electric field, Electric potential and Energy Band diagram at equilibrium

iv) Now apply -0.5 volt at left terminal and plot the Electric field, Electric potential and Energy Band diagram



Regions	Constant Doping density (cc)
Red colored N region	10E+18(Arsenic doped)
P-Region	1E+14 (Boron doped)
Orange colored N-Regio	3E+16 (Arsenic doped)

CAD: Computer Aided Design

❑ The use of computers to aid in the develop, analysis, modification, or optimization of a design/Product before making it in real life

❑ Used in several profession

Have you ever used CAD?

Yes!

Variant	Courses	Software
Mechanical Design Automation	Engineering Drawing	AutoCAD (2D, 3D), SolidWorks (3D)
Electronic Design Automation	Digital/Analog Circuits	SPICE (NG/LT), Keil (Verilog, VHDL), Cadence Virtuoso
Technology CAD	**Nanoelectronics, Physics of Transistors**	**Sentaurus TCAD, Silvaco TCAD**

When Purpose of CAD is to develop and optimize semiconductor process technologies and devices It is

TCAD

What is TCAD

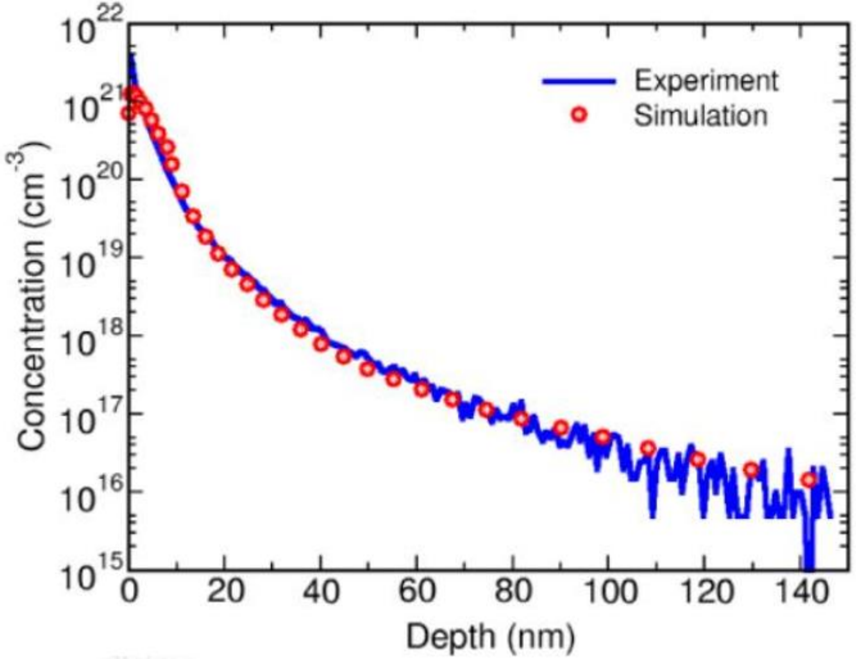
- ❑ TCAD is a branch of Electronic Design Automation (EDA) that models semiconductor device fabrication and semiconductor device operation.
- ❑ Essentially, TCAD is like a virtual lab that helps create and improve the technology for electronic devices.
- ❑ Process TCAD: Modelling of semiconductor-chip process-manufacturing steps like lithography, deposition, etching, ion implantation, diffusion, oxidation, silicidation, mechanical stress, etc..
  - ➢ Requires detailed modeling of the physical principles of manufacturing, as well as the modeling of the specific equipment's used.
- ❑ Device TCAD: Modeling of electrical, thermal, optical and mechanical behavior of semiconductor devices (e.g., diode, BJT, MOSFET, solar cell,…)
  - ➢ It focuses on the physical principles at the basis of carrier transport and of optical generation in semiconductor devices.

- Two main tools are available: **Silvaco** and **Sentarus** TCAD tools.

What is TCAD-Examples

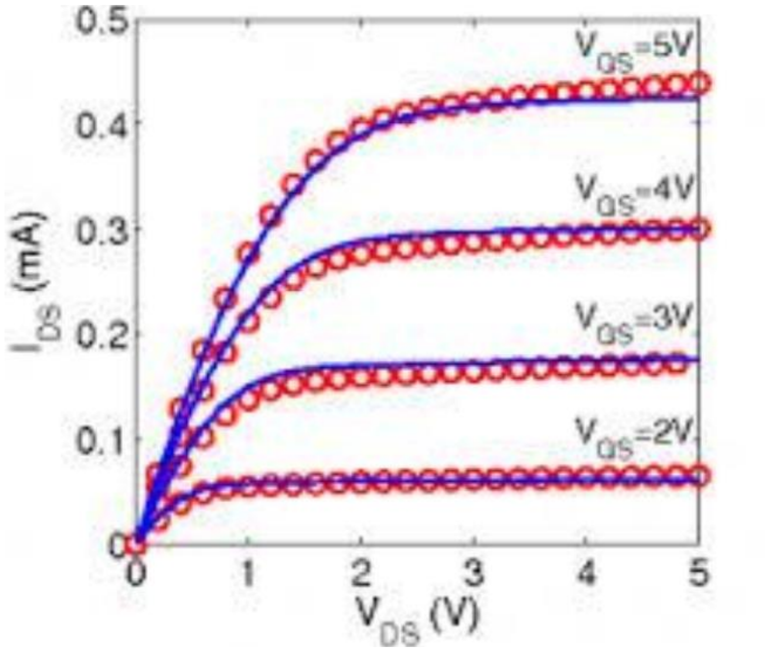
Process Simulation

- Simulate doping profiles obtained by specific processing techniques, calibrate the model with experimental data and then optimize the process to obtain the desired profile.



Device Simulation

- Simulate the output characteristics of a MOSFET device and calibrate the device architecture to fine-tune the device performance.

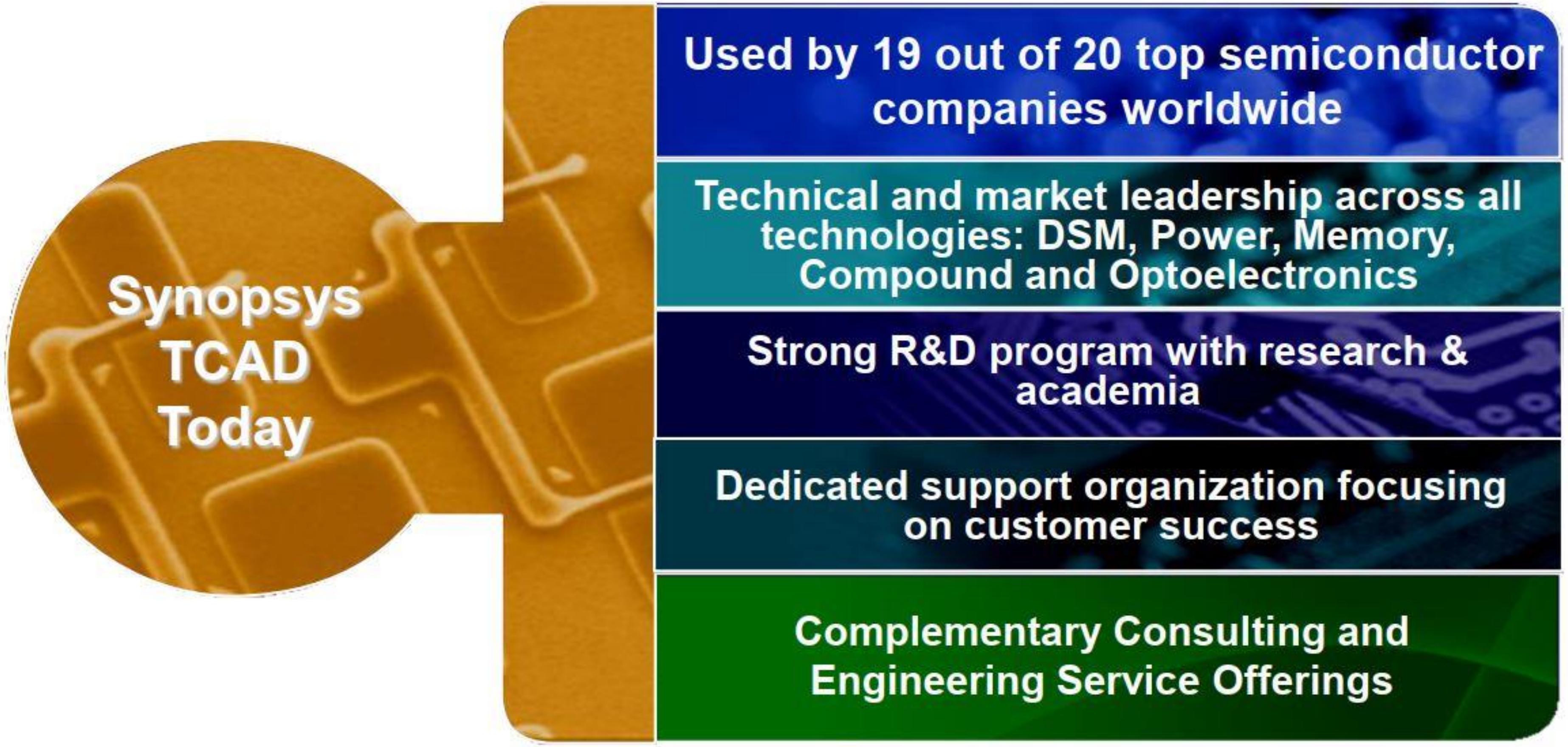


Why TCAD?

- To optimize the device features when hands-on calculations are too complicated or impose unacceptable assumptions.
- To make predictions (scaling, new device concepts) when hands-on calculations are not viable (e.g., complex devices, modeling of distributed statistical effects or process yield)
- To get insights. No real experiment will probably being able of measuring some of the physical quantities calculated by TCAD tools (e.g., local distribution of carriers, local electric field, etc.)
- **Features of the TCAD tool:**
  1. Fabrication, Operation, Reliability Simulations
  2. Reduce time and cost
  3. Physical insights
  4. Advanced numerical solvers and meshing techniques

TCAD Tools

- Now a Days for TCAD, the two major players are
- Synopsys Sentaurus, by far the most used   this course !
  - Silvaco ATLAS



Used by 19 out of 20 top semiconductor companies worldwide

Technical and market leadership across all technologies: DSM, Power, Memory, Compound and Optoelectronics

Strong R&D program with research & academia

Dedicated support organization focusing on customer success

Complementary Consulting and Engineering Service Offerings

Mobaxterm

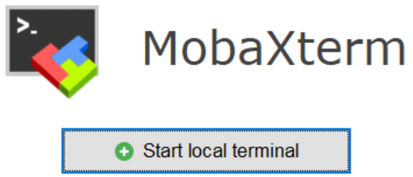
- Can we install TCAD in our lappy and start working like MATLAB…

BIG NO!!

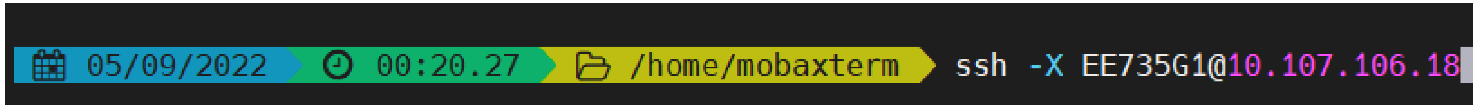
- Solution: Mobaxterm in PC + MCL Lab Server + TCAD License

- MobaXterm: A **toolbox for remote computing**.
- Download Link: https://mobaxterm.mobatek.net/download.html
- Install: Home (Installer)edition
  - Pin on task bar at least this semester!!

1.

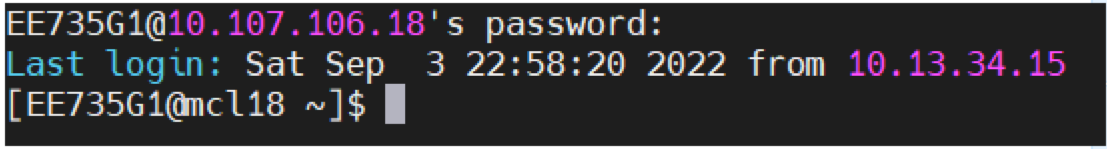


2.



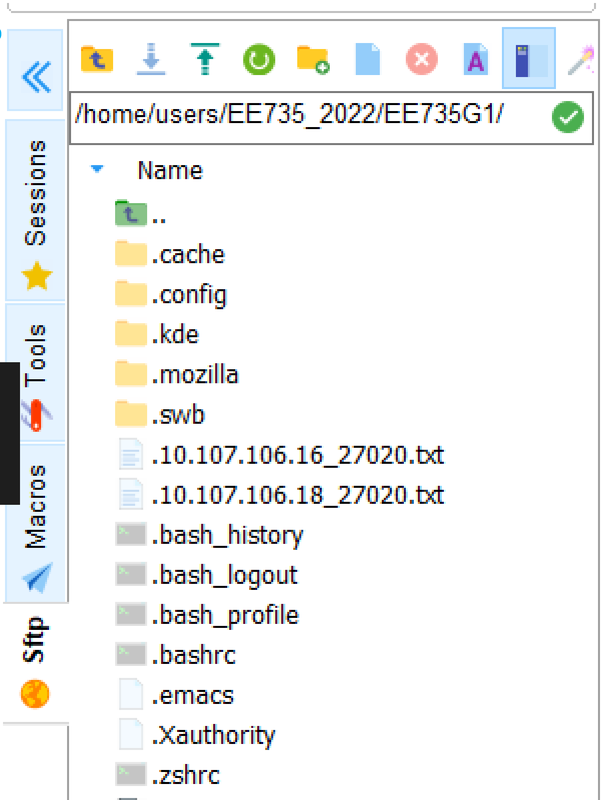
```
05/09/2022  00:20.27  /home/mobaxterm  ssh -X EE735G1@10.107.106.18
```

3.



```
EE735G1@10.107.106.18's password:
Last login: Sat Sep  3 22:58:20 2022 from 10.13.34.15
[EE735G1@mcl18 ~]$ 
```

4.



```
/home/users/EE735_2022/EE735G1/
..
.cache
.config
.kde
.mozilla
.swb
.10.107.106.16_27020.txt
.10.107.106.18_27020.txt
.bash_history
.bash_logout
.bash_profile
.bashrc
.emacs
.Xauthority
.zshrc
```

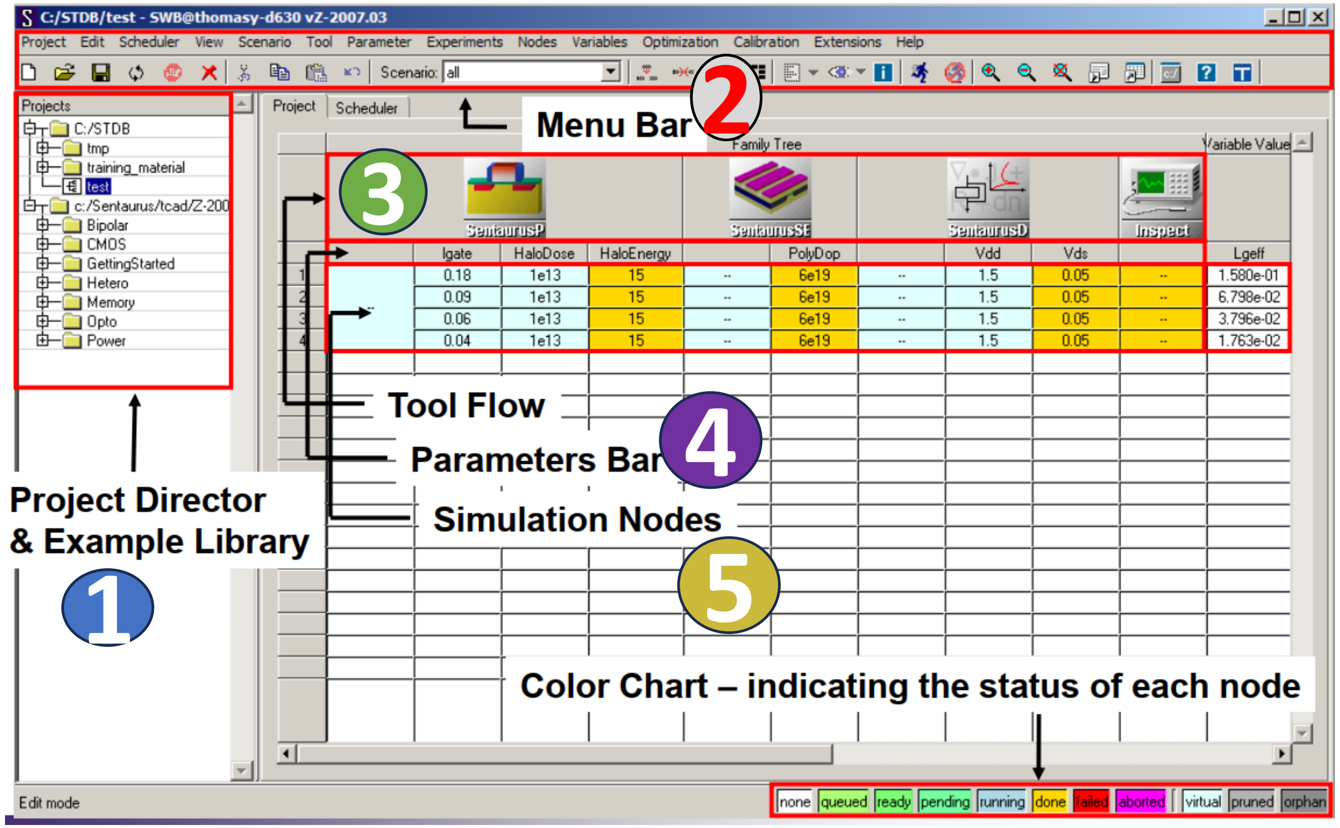
Gateway of TCAD : Sentaurus Workbench(SWB)

- What is Sentaurus Work Bench: A graphical front end that integrates TCAD Sentaurus simulation tools into one environment
- It allows
  - to define parameters and variables
  - to run comprehensive parametric analyses
  - To make several experiment with different parameter value on existing code

Sentaurus Smiulation tools:**Tools Suite**

1. SWB (workbench)
2. SDE (structure editor)
3. SDEVICE (device simulation)
4. SVISUAL (visualise)
5. SMESH (meshing)
6. SBAND (band structure)

and many more… All have GUI and/or code-based interaction



		Igate	HaloDose	HaloEnergy		PolyDop		Vdd	Vds		Lgeff
1		0.18	1e13	15	--	6e19	--	1.5	0.05	--	1.580e-01
2	--	0.09	1e13	15	--	6e19	--	1.5	0.05	--	6.798e-02
3		0.06	1e13	15	--	6e19	--	1.5	0.05	--	3.796e-02
4		0.04	1e13	15	--	6e19	--	1.5	0.05	--	1.763e-02

Menu Bar 2

3

Tool Flow

Parameters Bar 4

Simulation Nodes

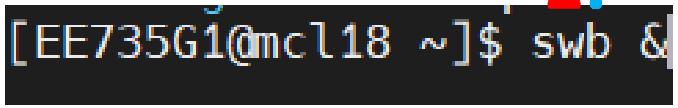
Project Director & Example Library 1

5

Color Chart – indicating the status of each node

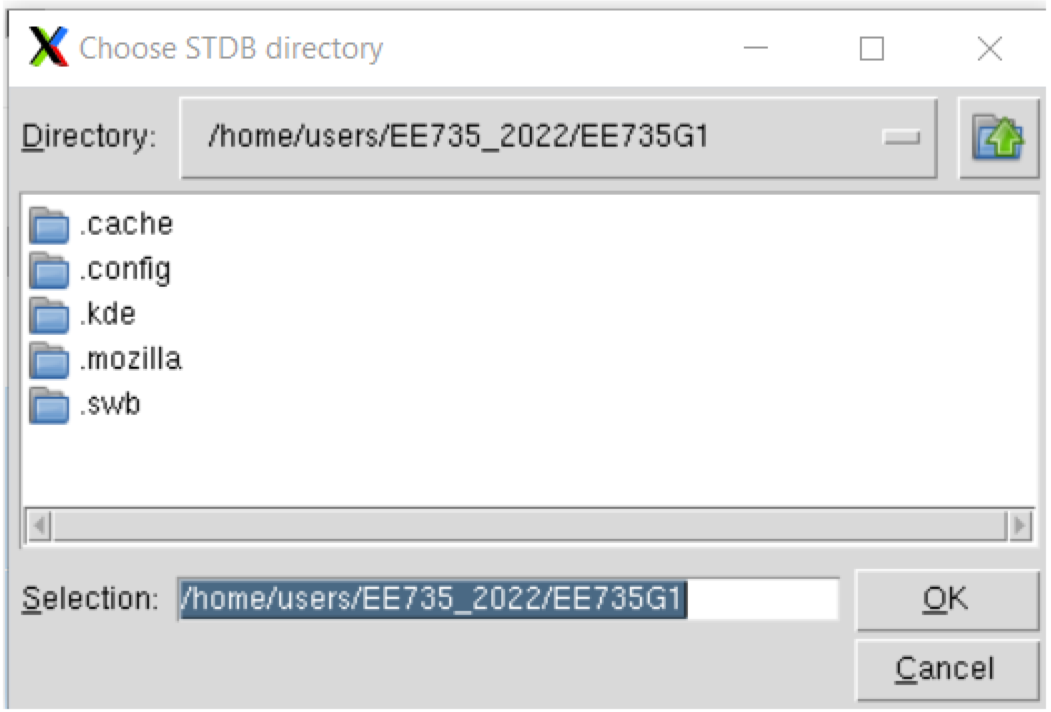
Launching SWB

1.



```
[EE735G1@mcl18 ~]$ swb &
```

2.



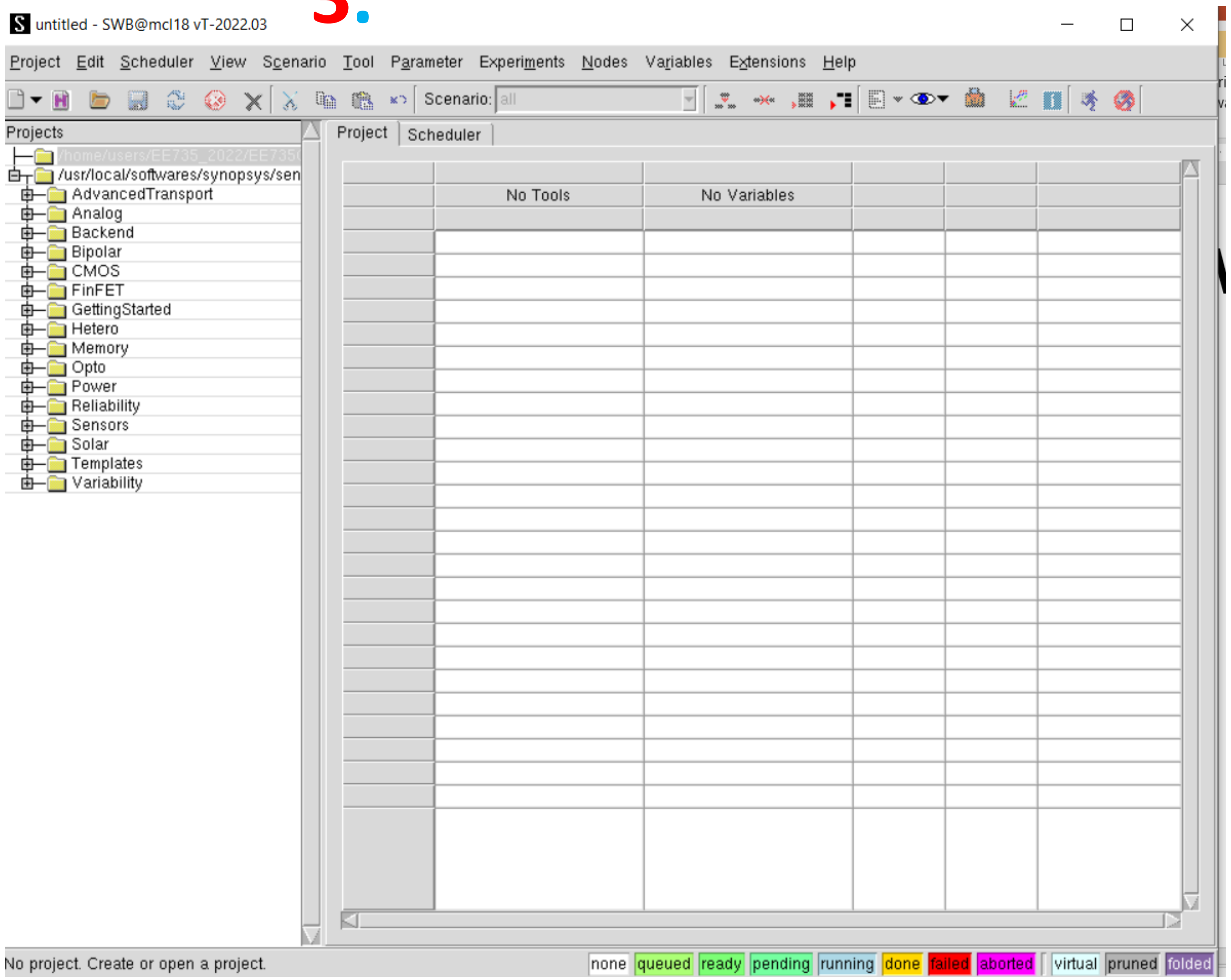
Choose STDB directory

Directory: /home/users/EE735_2022/EE735G1

.cache
.config
.kde
.mozilla
.swb

Selection: /home/users/EE735_2022/EE735G1

3.



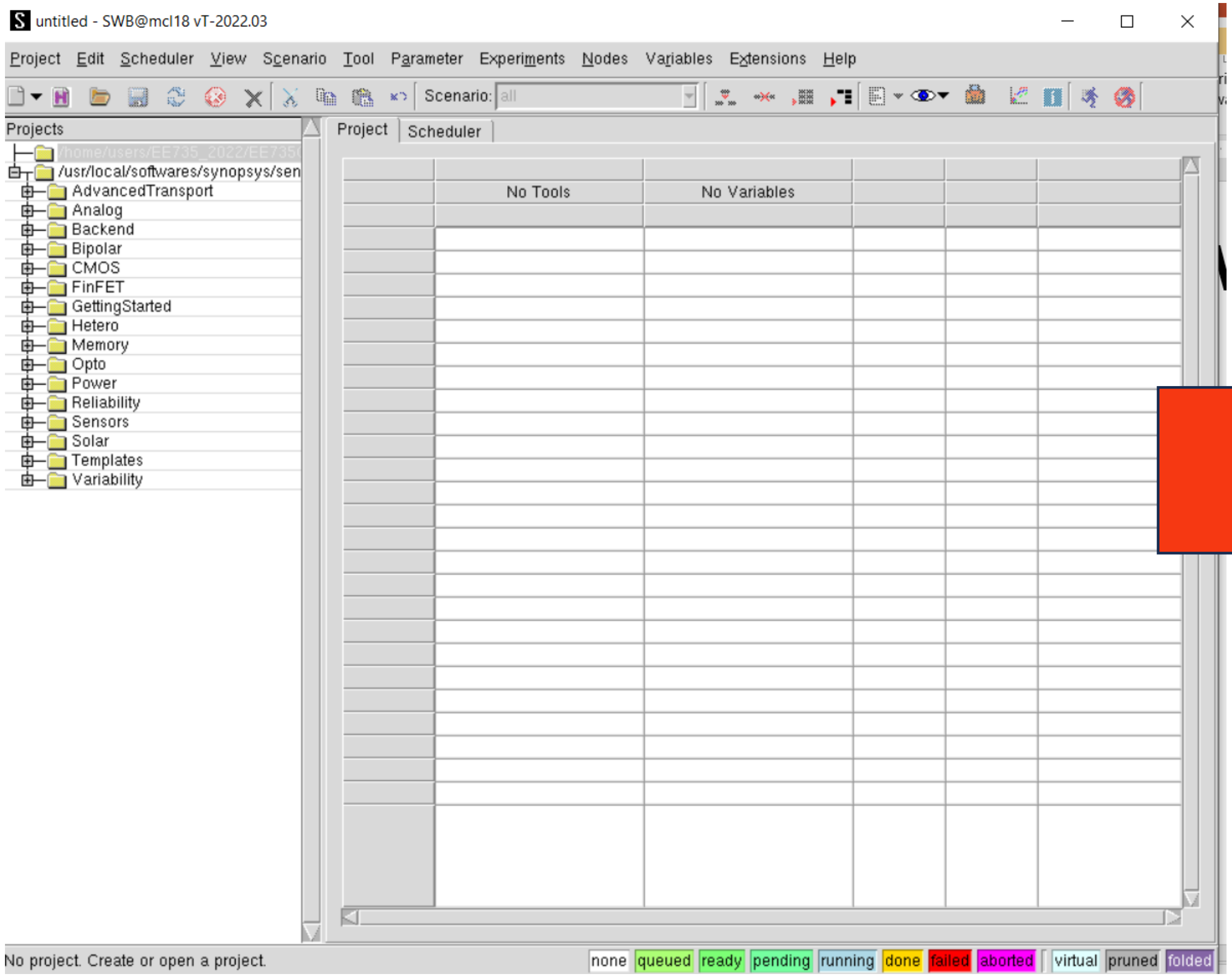
untitled - SWB@mcl18 vT-2022.03

Projects
/home/users/EE735_2022/EE735G
/usr/local/softwares/synopsys/sen
AdvancedTransport
Analog
Backend
Bipolar
CMOS
FinFET
GettingStarted
Hetero
Memory
Opto
Power
Reliability
Sensors
Solar
Templates
Variability

No project. Create or open a project.

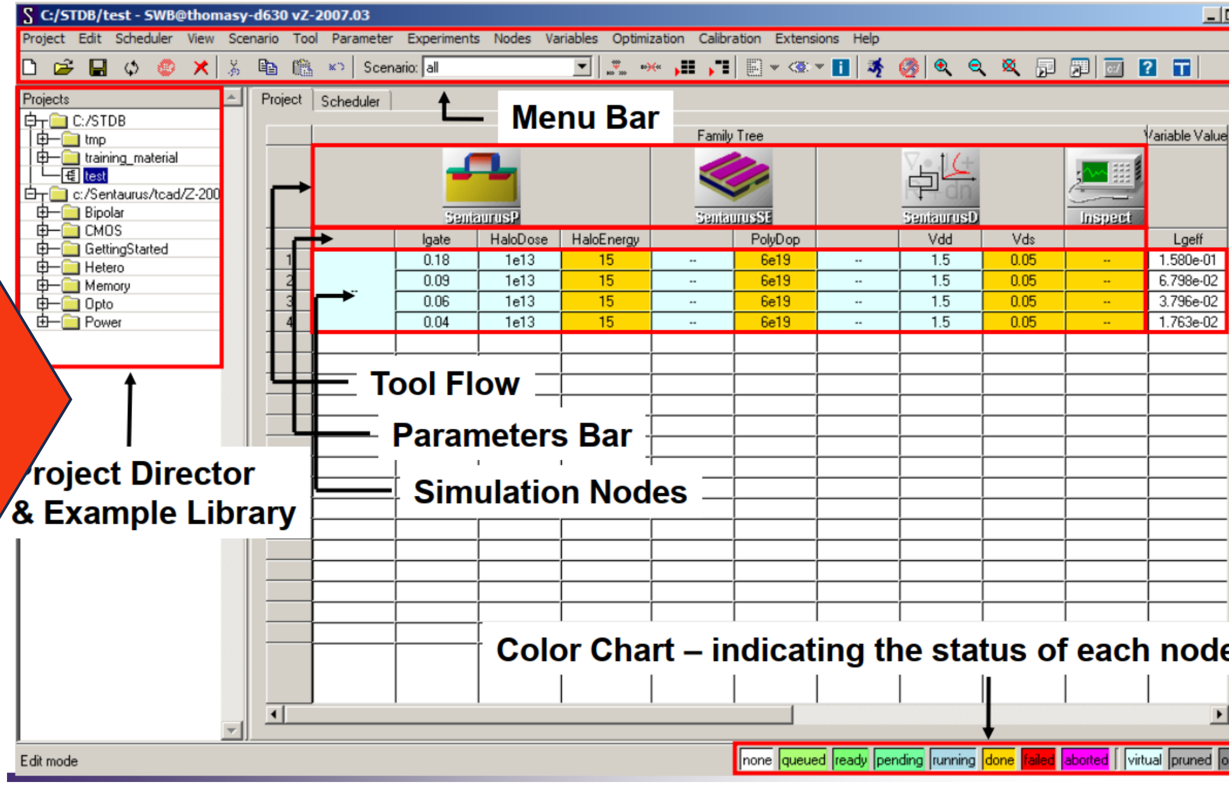
Figure 1. Main window of Sentaurus Workbench.

How to add tools and parameters?



No project. Create or open a project.

Figure 1. Main window of Sentaurus Workbench.



	Igate	HaloDose	HaloEnergy		PolyDop		Vdd	Vds		Lgeff
1	0.18	1e13	15	--	6e19	--	1.5	0.05	--	1.580e-01
2	0.09	1e13	15	--	6e19	--	1.5	0.05	--	6.798e-02
3	0.06	1e13	15	--	6e19	--	1.5	0.05	--	3.796e-02
4	0.04	1e13	15	--	6e19	--	1.5	0.05	--	1.763e-02

Edit mode

Menu Bar

Tool Flow

Parameters Bar

Simulation Nodes

roject Director
& Example Library

Color Chart – indicating the status of each node

Figure 2

Starting with Sentaurus Workbench

Task #1

- ❑ Create Your first folder
- ❑ Name your Project & Save in appropriate folder otherwise TCAD will save your project in default folder named “tmp”
- ❑ Add tools in sentaurus Workbech SDE and SDEVICE
- ❑ Add Code to your respective tools (**At this stage do not bother about how to write code**)

Sentaurus Smiulation tools:**Tools Suite**

1. SWB (workbench)
2. SDE (structure editor)
3. SDEVICE (device simulation)
4. SVISUAL (visualise)
5. SMESH (meshing)
6. SBAND (band structure)

and many more… All have GUI and/or code-based interaction

Working with Sentaurus Workbench

Task #2

- ❑ Run your Tool in order one by one
- ❑ Observe the Running status with color code
- ❑ How & Where to look simulation output : **Role of Svisual tool**
- ❑ Know your different types of output files

Concept of node:

It can be thought as a container at which values of various physical properties are stored and updated during the simulation. In workbench, it is the rightmost column of each tool.

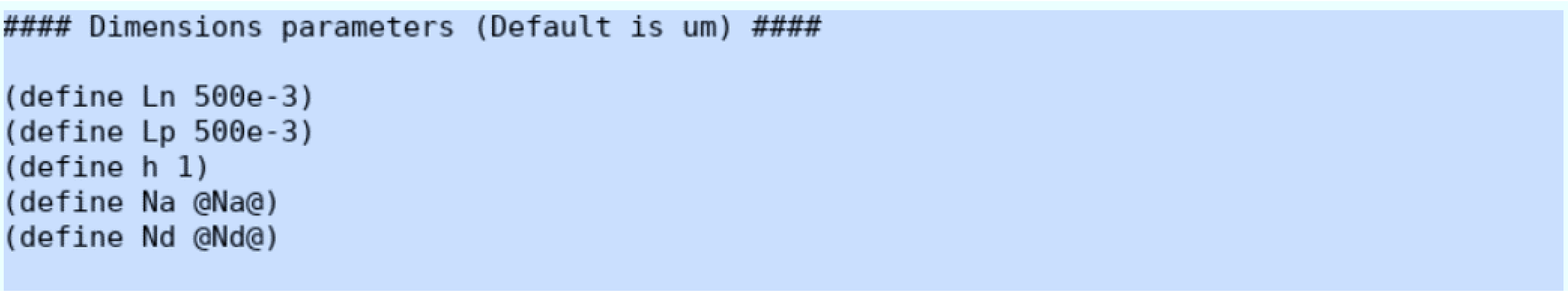
File name extension	Description	Remarks
.cmd	Preprocessed simulation input command file.	Sentaurus Workbench variables are replaced by their actual values.
.err	File containing simulator error messages.	Error messages generated by Tcl procedures or licensing can be found here as well.
sge.err	Error messages from gjob.	Contains error messages from gsub if the job failed.
.out, .log	Simulation log file.	Information about the progress of the simulation.
.plt	Device simulation current file.	Contains solution variables at device terminals (such as voltages and currents).
.sta	Status of the simulation.	The status can be one of the following: none | queued | running | failed | aborted | done. The status is indicated by the color of the simulation node, which also contains information about the execution host.
.tdr	The structure with simulation results.	Contains field data such as doping distribution and electrostatic potential.
.job	Information from gjob.	Contains information from the preprocessor and the simulation job submission. Watch here for error messages if a simulation node is not executed.
.par	Preprocessed parameter file.	Used in Sentaurus Device, EMW, and Ligament Layout Editor.

How to read TCAD code

- Programming language used in TCAD: **Scheme**
  - A Scheme command is enclosed in parentheses (white spaces are allowed)
  - A Scheme command can extend over several lines, or several Scheme commands can be placed on a single line. A given Scheme command is identified by the outermost matching parentheses pair: `(beginning of Scheme command`

    ```
    continuation of the same Scheme command
    end of the Scheme command)
    ```
  - `(define i 3)`
  - `(define j (+ i 5));` Means j = I + 5
- TCAD command format:
  - (Command "Argument1" "Argument2" "Argument3" "Argument4" etc.....)
- Some arguments are:
  - Compulsory
  - Optional
- Searching in Manual
  - Pick the command & copy from any code->Go to pdf manual ->Press CTRL + F ->Paste the copied item & Enter

SDE tool code: Part by part



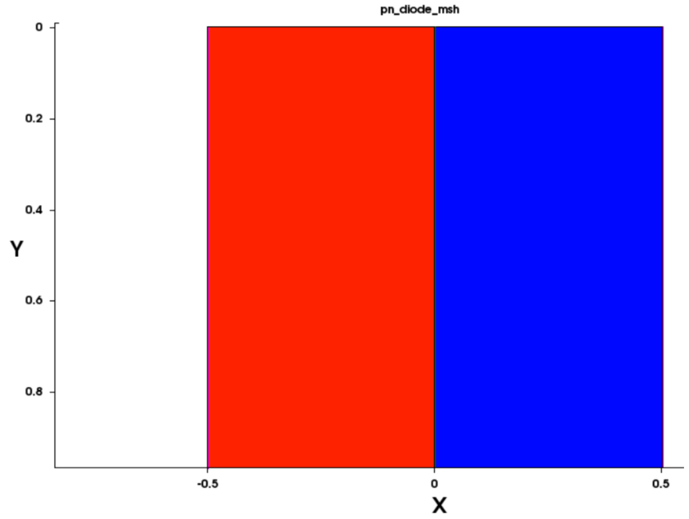
```
#### Dimensions parameters (Default is um) ####

(define Ln 500e-3)
(define Lp 500e-3)
(define h 1)
(define Na @Na@)
(define Nd @Nd@)
```

Any dimension in TCAD is by default in um

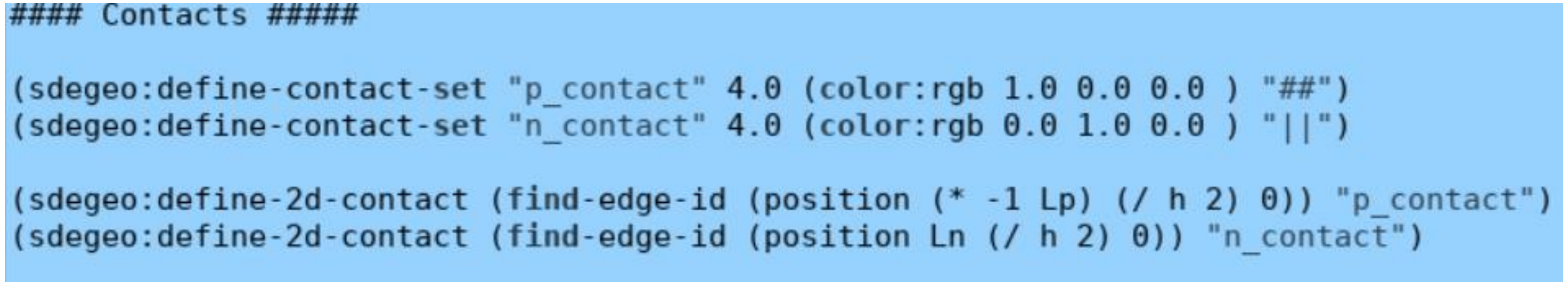
Ln = 500e-3 *1e-6

Or **Ln = 0.5 um**

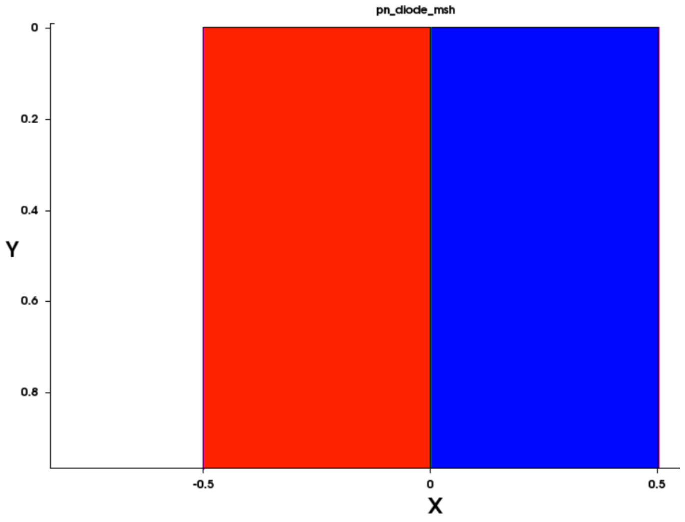
h = 1 um



SDE tool code: Part by part



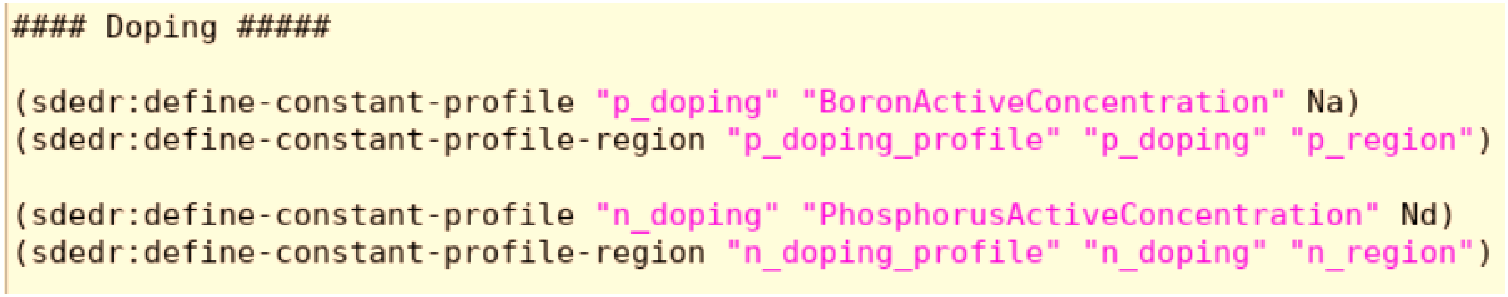
```
#### Contacts #####

(sdegeo:define-contact-set "p_contact" 4.0 (color:rgb 1.0 0.0 0.0 ) "##")
(sdegeo:define-contact-set "n_contact" 4.0 (color:rgb 0.0 1.0 0.0 ) "||")

(sdegeo:define-2d-contact (find-edge-id (position (* -1 Lp) (/ h 2) 0)) "p_contact")
(sdegeo:define-2d-contact (find-edge-id (position Ln (/ h 2) 0)) "n_contact")
```



Illustrating code

1. **sdegeo:** This is the name of the TCAD module that defines geometric objects for the SDE. It allows users to create and modify the geometry of a device, such as a semiconductor device.
2. **define-contact-set**: This is a function within the sdegeo module that defines a new contact set in the SDE. A contact set is a collection of points or surfaces in a device where electrical contacts can be applied.
3. **"p_contact " 4.0:** This is the name of the contact set being defined by users. It could be any string of characters. 4 represents the depth of the contact set, measured in microns.
4. **(color:rgb 1.0 0.0 0.0):** This specifies the color of the contact set in the SDE. It is a shade of red with full intensity in the red channel, and no intensity in the green or blue channels.
5. **The double hash or percent or || sign** is often used to represent an electrical contact in TCAD simulations.
6. **define-2d-contact":** This is a function provided by the SDEGEO module that defines a 2D contact in the simulation.
7. **"find-edge-id":** This is a function that is used to find the ID of an edge in the device structure. The edge being referred to in this code lies at a specific position in the structure.
8. **"n_contact":** This is a string that represents the name of the 2D contact being defined in the simulation. This name can be used to refer to the contact in other parts of the simulation.

SDE tool code: Part by part

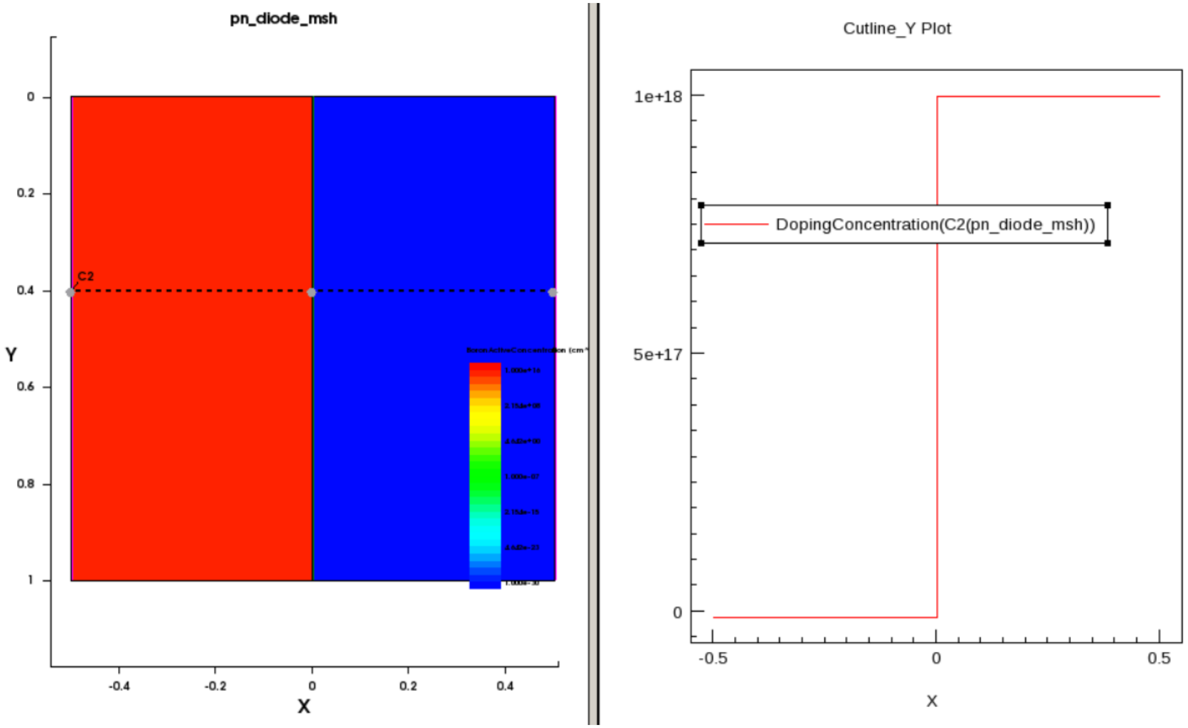


```
#### Doping #####

(sdedr:define-constant-profile "p_doping" "BoronActiveConcentration" Na)
(sdedr:define-constant-profile-region "p_doping_profile" "p_doping" "p_region")

(sdedr:define-constant-profile "n_doping" "PhosphorusActiveConcentration" Nd)
(sdedr:define-constant-profile-region "n_doping_profile" "n_doping" "n_region")
```

Understanding Y-Cut and X-cut

Y-Cut: **Fix** y-coordinate, vary-x

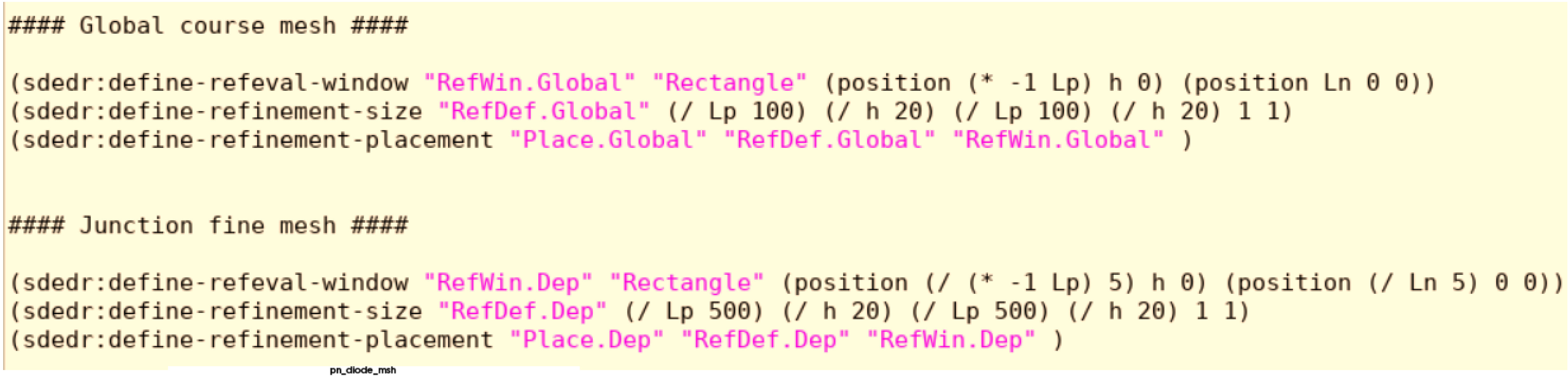
X-cut: Fix x -coordinate vary- y



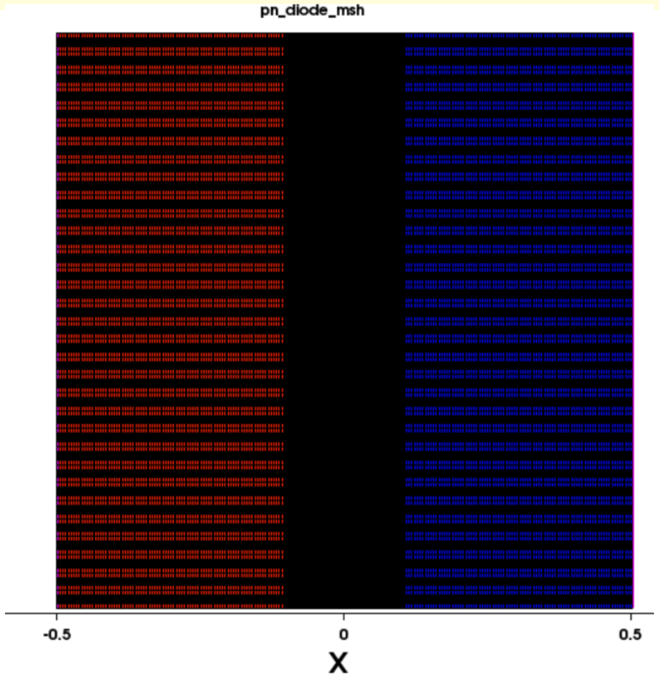
Illustrating code

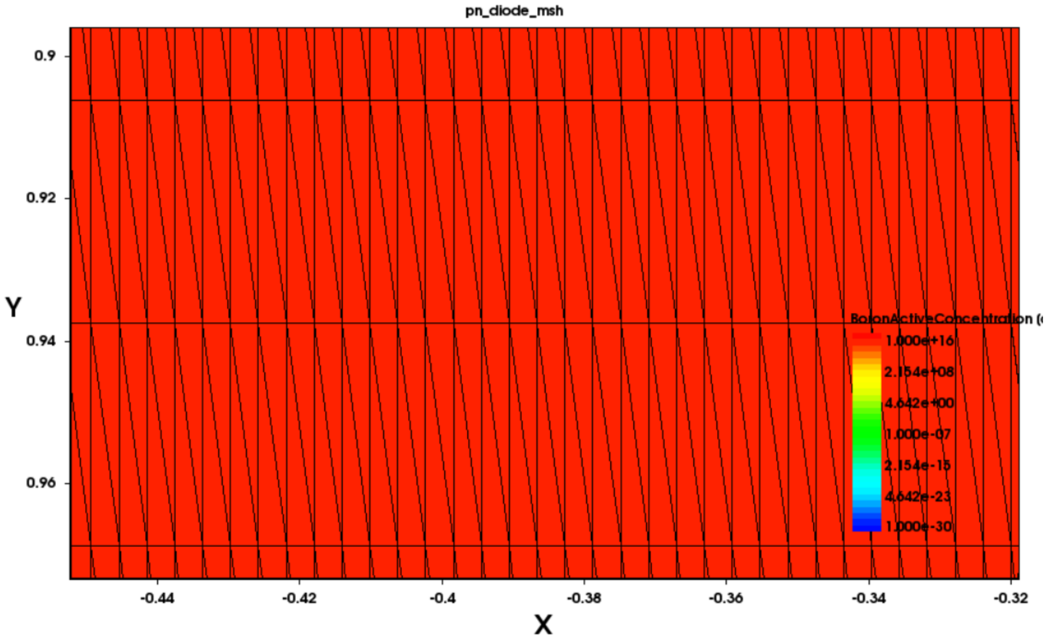
1. **sdedr: You tell**
2. **define-constant-profile**: This is a function within the sdedr module that commands for defining a constant profile with the user defined name”p_doping” and the value as”BoronActiveConcentration= Na”
3. **define-constant-profile-region: A** function within the sdedr module that defines a constant profile region with name “p_doping_profile” **and value= p_doping in region “p_region”**

SDE tool code: Part by part

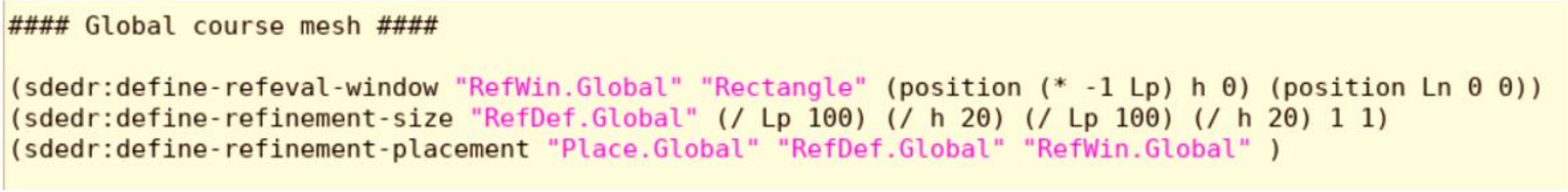


```
#### Global course mesh ####

(sdedr:define-refeval-window "RefWin.Global" "Rectangle" (position (* -1 Lp) h 0) (position Ln 0 0))
(sdedr:define-refinement-size "RefDef.Global" (/ Lp 100) (/ h 20) (/ Lp 100) (/ h 20) 1 1)
(sdedr:define-refinement-placement "Place.Global" "RefDef.Global" "RefWin.Global" )


#### Junction fine mesh ####

(sdedr:define-refeval-window "RefWin.Dep" "Rectangle" (position (/ (* -1 Lp) 5) h 0) (position (/ Ln 5) 0 0))
(sdedr:define-refinement-size "RefDef.Dep" (/ Lp 500) (/ h 20) (/ Lp 500) (/ h 20) 1 1)
(sdedr:define-refinement-placement "Place.Dep" "RefDef.Dep" "RefWin.Dep" )
```





SDE tool code: Part by part



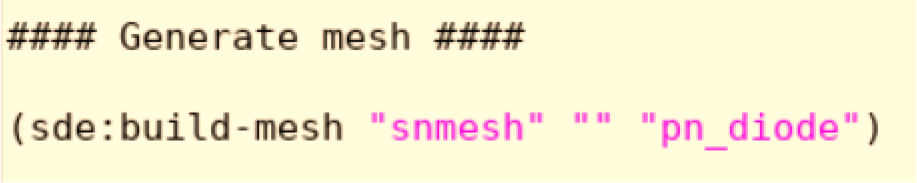
```
#### Global course mesh ####

(sdedr:define-refeval-window "RefWin.Global" "Rectangle" (position (* -1 Lp) h 0) (position Ln 0 0))
(sdedr:define-refinement-size "RefDef.Global" (/ Lp 100) (/ h 20) (/ Lp 100) (/ h 20) 1 1)
(sdedr:define-refinement-placement "Place.Global" "RefDef.Global" "RefWin.Global" )
```

Code Breakdown:

1. **sdedr**: Please Recall !!
2. **define-refeval-window:** A command in TCAD used to define a refinement window. The name of this window is user defined here it is “RefWin.Global” .
3. "**Rectangle**": This is the shape of the refinement window. In this case, it is a rectangle.
4. **(position (* -1 Lp) h 0) (position Ln 0 0)** : This specifies the coordinates of the lower left and upper right corner of the rectangle.
5. **define-refinement-size:** This command allows the user to specify the desired mesh size or resolution in a particular region or layer of the device being simulated. “RefDef.Global” is identifier or name
6. (/ Lp 100) (/ h 20) (/ Lp 100) (/ h 20) 1 1): Defines mesh size for x-direction and y-direction

Mesh Size Plays important Role in device Simulation. We spent considerable time in our Real device for optimizing appropriate mesh size

SDE tool code: Part by part



```
#### Generate mesh ####

(sde:build-mesh "snmesh" "" "pn_diode")
```

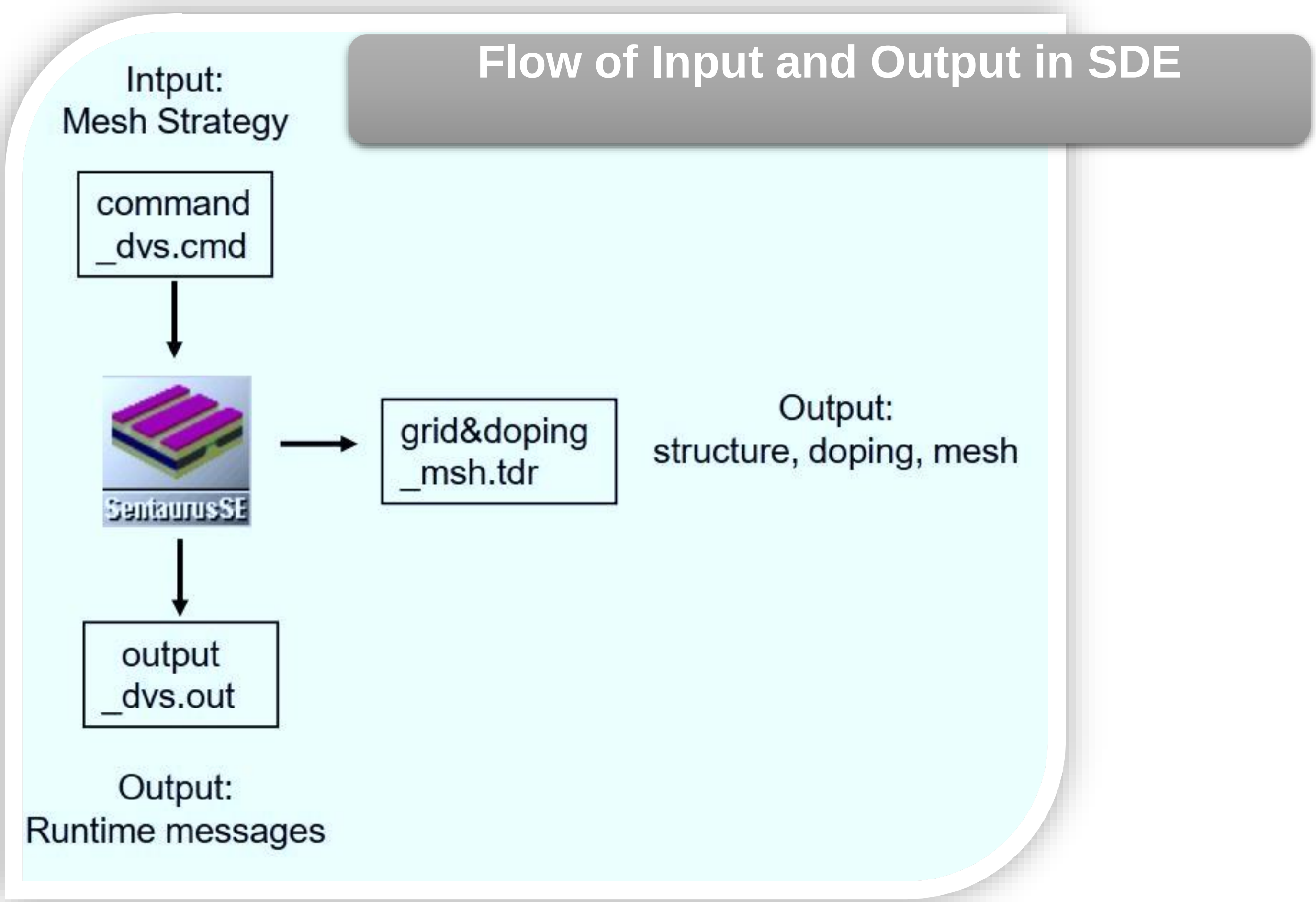
Code Breakdown:

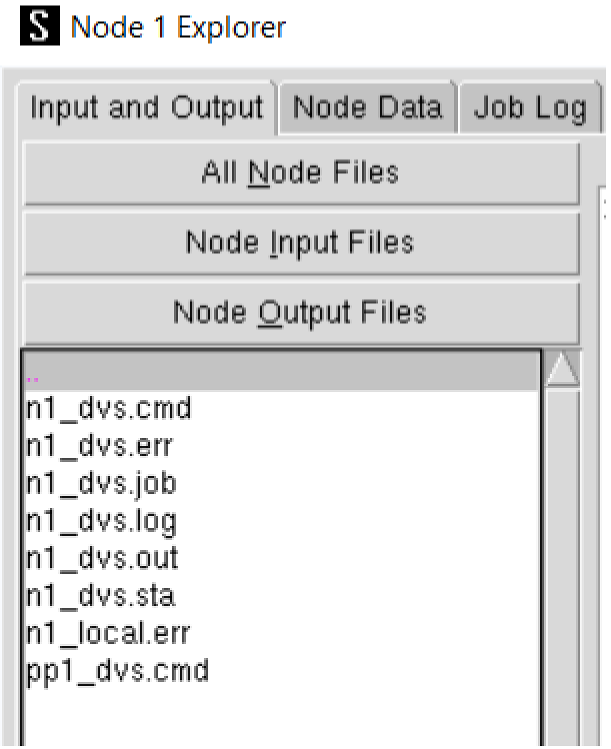
1. **sde**: Please Recall !!
2. **build-mesh:** A command/function in TCAD used to create mesh. As we have discussed that mesh is grid like structure where all the physical quantity of interest is computed internally.

3. **“snmesh"**: This refers the name/identifier of mesh given by user to refer of TCAD.
4. "“ : This empty string likely represents some optional parameters or settings for the mesh building process. Here no additional parameters are being specified.
5. **"pn_diode“ :** This specifies the name of structure or device (given by user) you want to create a mesh

SDE tool code: Part by part

```
#### Generate mesh ####

(sde:build-mesh "snmesh" "" "pn_diode")
```



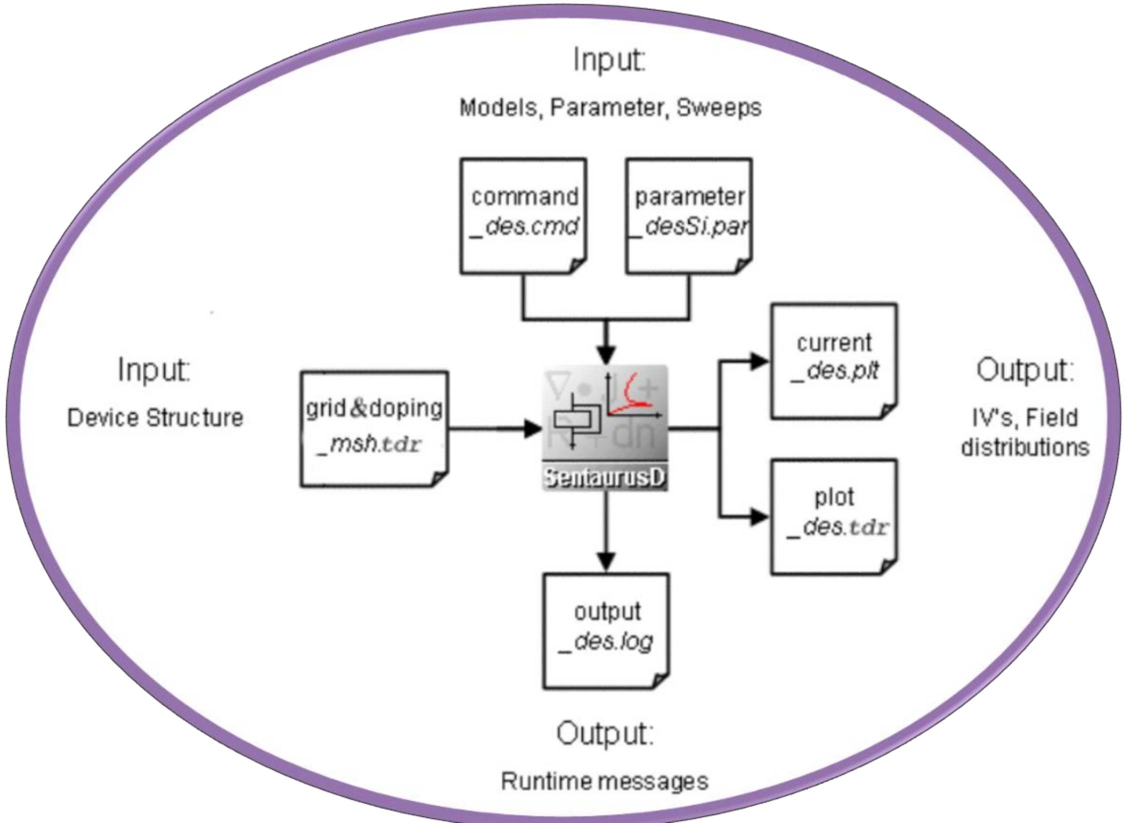


```
..
n1_dvs.cmd
n1_dvs.err
n1_dvs.job
n1_dvs.log
n1_dvs.out
n1_dvs.sta
n1_local.err
pp1_dvs.cmd
```

Sdevice Tool

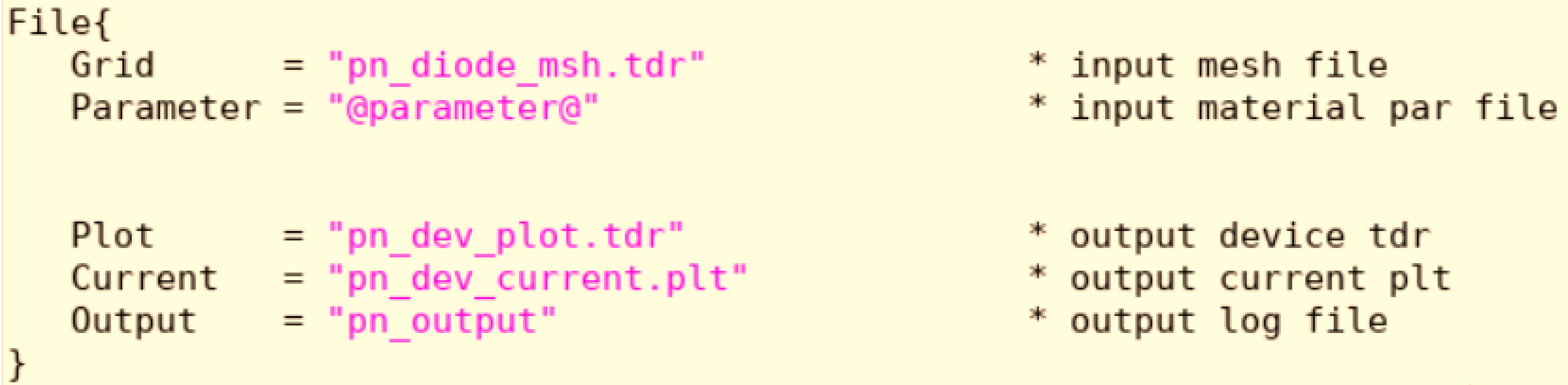
- Sections of Sdevice Command file
  - **File** {“define the input and output files of the simulation }
  - **Electrode** { “define electrical (or thermal) contacts, initial bias condition, special boundary condition” }
  - **Physics**{Declare Physics Models}
  - **Plot** {Mention the solution variables that need to be plotted for our device}
  - **Math**{Numerical solver method is mentioned}
  - **Solve**{Set bias sweeps}

Flow of Input and Output in Sdevice



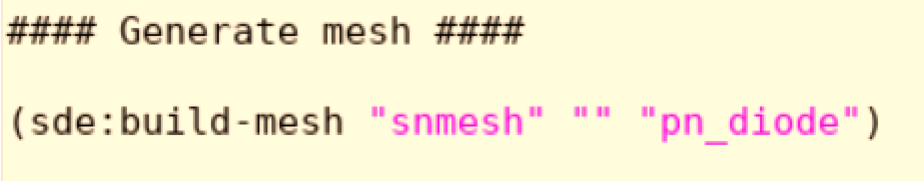
Reading SDEVICE Code

File section



```
File{
   Grid      = "pn_diode_msh.tdr"                * input mesh file
   Parameter = "@parameter@"                     * input material par file


   Plot      = "pn_dev_plot.tdr"                 * output device tdr
   Current   = "pn_dev_current.plt"              * output current plt
   Output    = "pn_output"                       * output log file
}
```

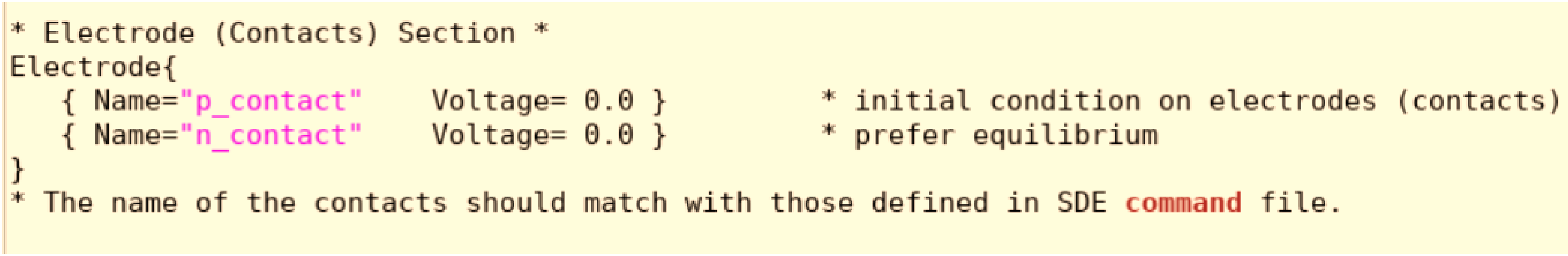
Grid input file should be same as that you have named mesh file in SDE



```
#### Generate mesh ####

(sde:build-mesh "snmesh" "" "pn_diode")
```

Reading SDEVICE Code

Electrode Section



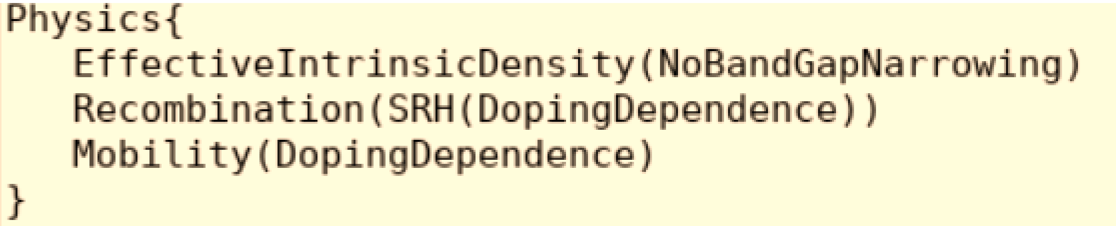
```
* Electrode (Contacts) Section *
Electrode{
   { Name="p_contact"    Voltage= 0.0 }         * initial condition on electrodes (contacts)
   { Name="n_contact"    Voltage= 0.0 }         * prefer equilibrium
}
* The name of the contacts should match with those defined in SDE command file.
```

A contact may be made Schottky or Ohmic in this section of code

Reading SDEVICE Code

Physics section - What physical models to activate in our semiconductor device

Physics Section

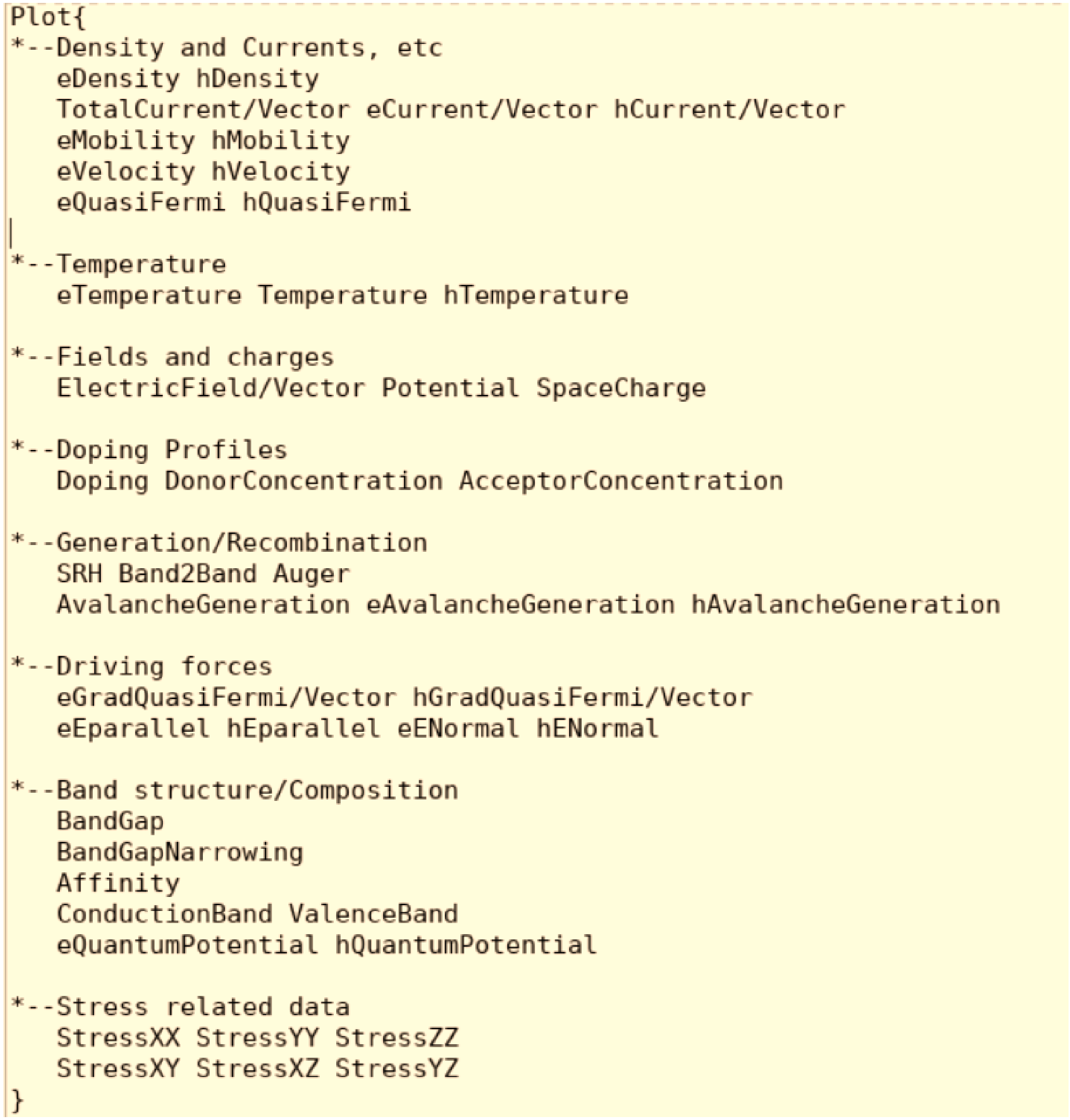


```
Physics{
   EffectiveIntrinsicDensity(NoBandGapNarrowing)
   Recombination(SRH(DopingDependence))
   Mobility(DopingDependence)
}
```

You may change the model and observe their impacts on device characteristics
The format is "Keyword(Options)"

Reading SDEVICE Code

Physics section



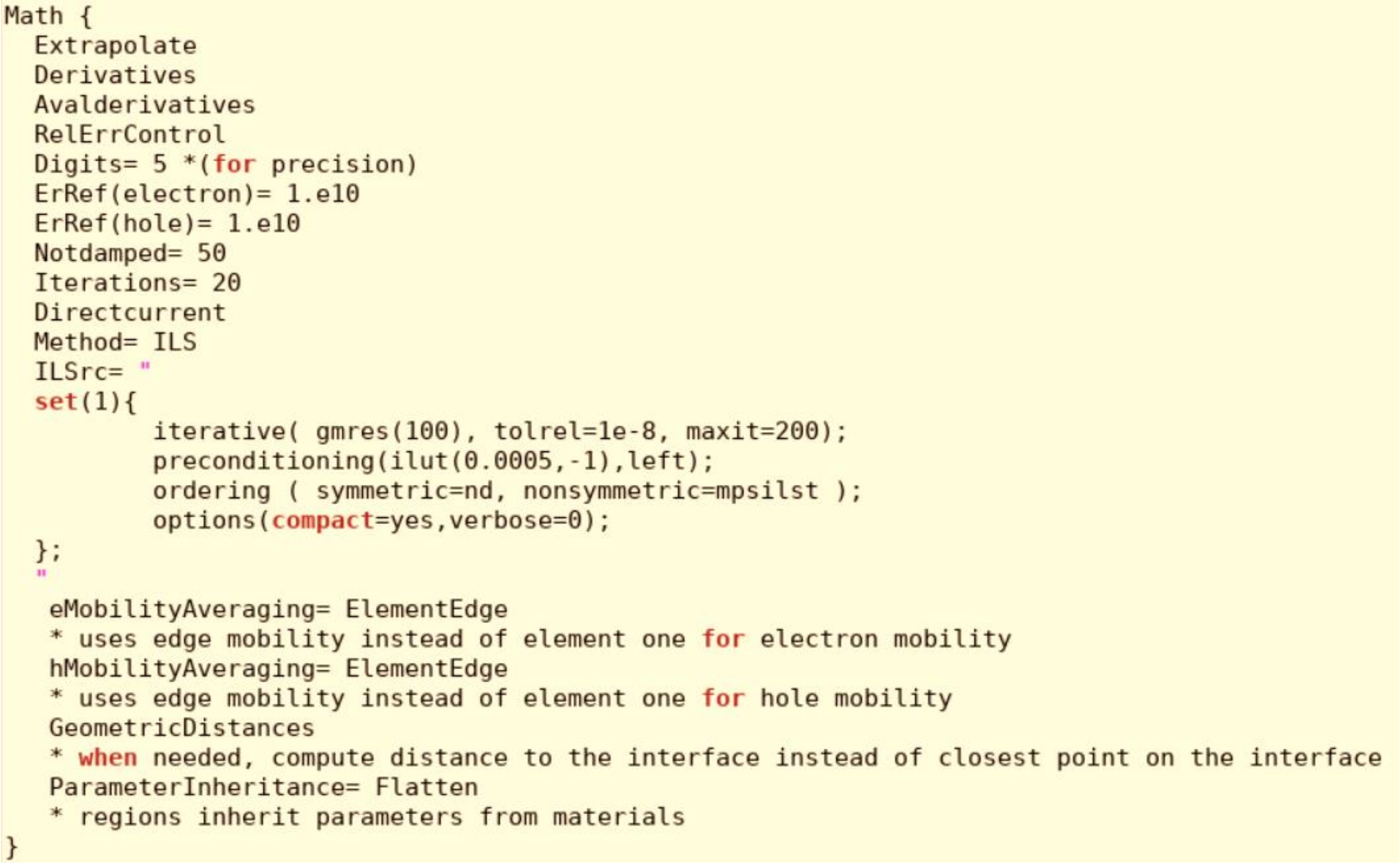
```
Plot{
*--Density and Currents, etc
   eDensity hDensity
   TotalCurrent/Vector eCurrent/Vector hCurrent/Vector
   eMobility hMobility
   eVelocity hVelocity
   eQuasiFermi hQuasiFermi

*--Temperature
   eTemperature Temperature hTemperature

*--Fields and charges
   ElectricField/Vector Potential SpaceCharge

*--Doping Profiles
   Doping DonorConcentration AcceptorConcentration

*--Generation/Recombination
   SRH Band2Band Auger
   AvalancheGeneration eAvalancheGeneration hAvalancheGeneration

*--Driving forces
   eGradQuasiFermi/Vector hGradQuasiFermi/Vector
   eEparallel hEparallel eENormal hENormal

*--Band structure/Composition
   BandGap
   BandGapNarrowing
   Affinity
   ConductionBand ValenceBand
   eQuantumPotential hQuantumPotential

*--Stress related data
   StressXX StressYY StressZZ
   StressXY StressXZ StressYZ
}
```

- ❖ In TCAD, the plot keyword is used to define a plot of various quantities that are computed during a simulation

- ❖ The plot keyword is typically followed by a list of one or more keywords that specify the quantities to be plotted. These keywords can include quantities such as the **electron and hole density**, e**lectrostatic potential, electric field, doping profile, mobility, and recombination rate**, among others.

Reading SDEVICE Code

Math section



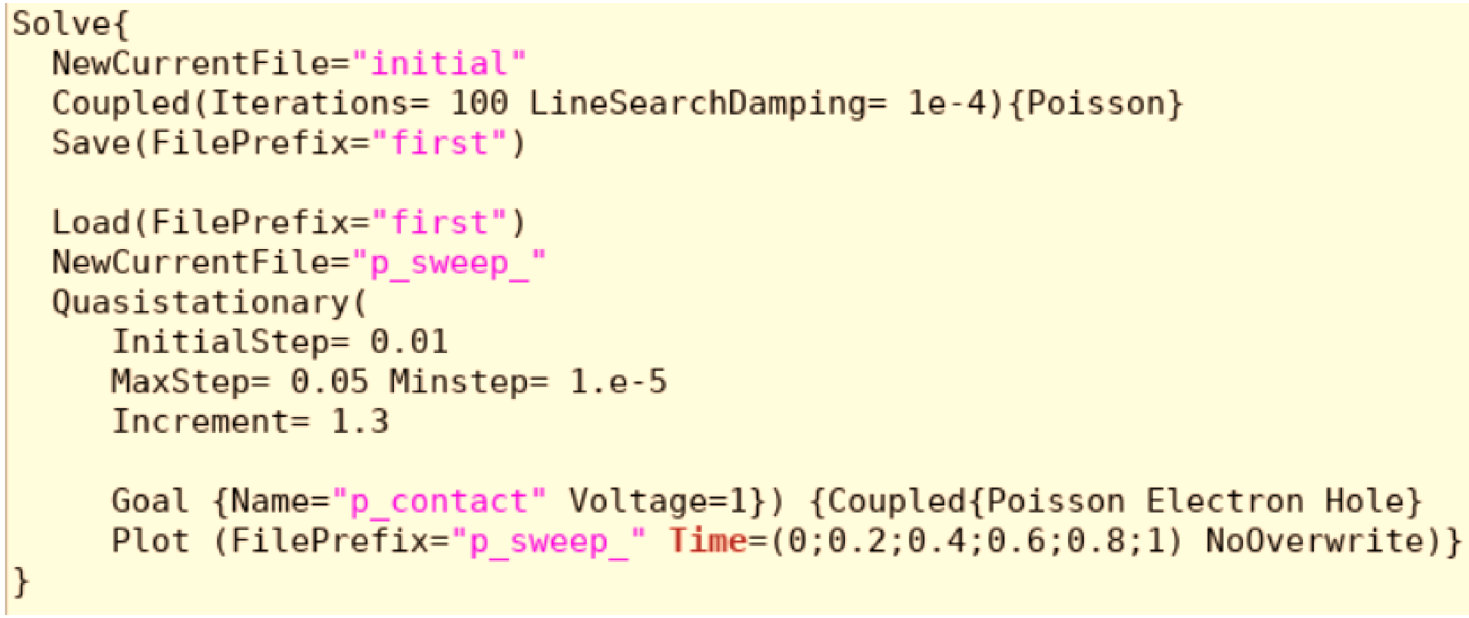
```
Math {
  Extrapolate
  Derivatives
  Avalderivatives
  RelErrControl
  Digits= 5 *(for precision)
  ErRef(electron)= 1.e10
  ErRef(hole)= 1.e10
  Notdamped= 50
  Iterations= 20
  Directcurrent
  Method= ILS
  ILSrc= "
  set(1){
          iterative( gmres(100), tolrel=1e-8, maxit=200);
          preconditioning(ilut(0.0005,-1),left);
          ordering ( symmetric=nd, nonsymmetric=mpsilst );
          options(compact=yes,verbose=0);
  };
  "
   eMobilityAveraging= ElementEdge
   * uses edge mobility instead of element one for electron mobility
   hMobilityAveraging= ElementEdge
   * uses edge mobility instead of element one for hole mobility
   GeometricDistances
   * when needed, compute distance to the interface instead of closest point on the interface
   ParameterInheritance= Flatten
   * regions inherit parameters from materials
}
```

Few **elaborations** from math section:

1. **Extrapolate:** This option enables extrapolation of solutions from previous time steps to improve the convergence rate of the solver.
2. ***Derivatives:*** This option enables automatic calculation of derivatives using finite differences, which can be useful for sensitivity analysis and optimization.
3. **Notdamped=50:** This specifies the number of iterations before the solver switches to a damping strategy to improve the convergence rate.
4. **Iterations=30:** This specifies the maximum number of iterations allowed for the solver.
5. **Now your turn** ………First & long time users should go in detail and take notes at least common keywords from SDEVICE manual

Reading SDEVICE Code

Solve section



```
Solve{
  NewCurrentFile="initial"
  Coupled(Iterations= 100 LineSearchDamping= 1e-4){Poisson}
  Save(FilePrefix="first")

  Load(FilePrefix="first")
  NewCurrentFile="p_sweep_"
  Quasistationary(
     InitialStep= 0.01
     MaxStep= 0.05 Minstep= 1.e-5
     Increment= 1.3

     Goal {Name="p_contact" Voltage=1}) {Coupled{Poisson Electron Hole}
     Plot (FilePrefix="p_sweep_" Time=(0;0.2;0.4;0.6;0.8;1) NoOverwrite)}
}
```

Few **elaborations** from Solve section:

1. **NewCurrentFile="initial":** This command sets the name for the current simulation run to "initial.
2. **Coupled(Iterations= 100 LineSearchDamping= 1e-4) {Poisson}:** This command specifies a coupled simulation involving the Poisson equation. It will run for 100 iterations with a line search damping factor of 1e-4.
3. **Save(FilePrefix="first"):** This command saves the results of the coupled simulation with the file prefix "first."
4. **Load(FilePrefix="first"):** This loads the results of the previous simulation run with the file prefix "first." It sets for another simulation based on the results of the initial run.
5. **Goal {Name="p_contact" Voltage=1}:** This command defines a goal for the simulation. **It means setting a voltage goal at a specific contact named "p_contact" to a value of 1**. This goal likely represents a voltage boundary condition for the simulation.

6. Now your turn….. Look for **Quasistaionary**

Thank You !

Happy Learning