# Getting started with Silvaco ATLAS

شبیه ساز **Silvaco TCAD** از چندین شبیه ساز دیگر تشکیل شده است که مهمترین آنها **ATLAS** و **ATHENA** می باشند. **ATLAS** برای شبیه سازی ادوات نیمه هادی و بدست آوردن مشخصه های الکتریکی دیوایس ها بکار می رود و **ATHENA** مشخصه های فیزیکی دیوایس را در طول فرآیند ساخت پیش بینی میکند.

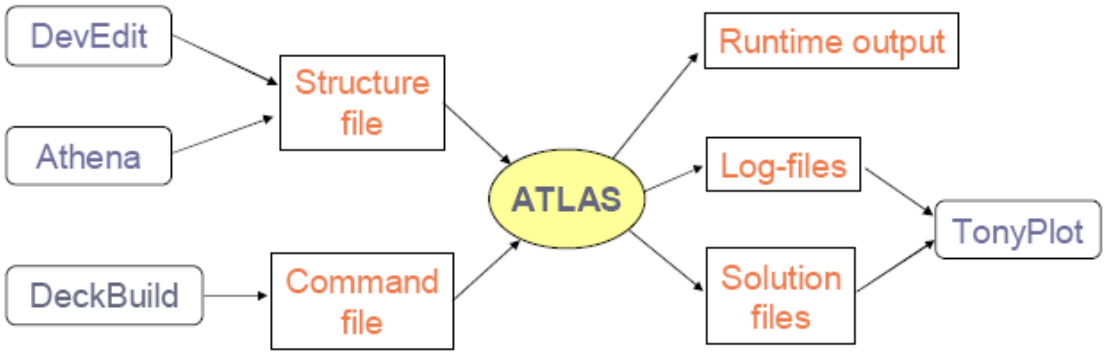
اکثر شبیه سازی های **ATLAS** دو نوع ورودی :

*1) Text files*
*2) Structure files*

و سه نوع خروجی :

*1) Runtime output*
*2) Log files*
*3) Solution files*

دارند که بطور خلاصه در شکل زیر نشان داده شده است.




ما با برنامه **Deckbuild** کار میکنیم که اطلاعات ورودی را در قالب یک فایل متنی با پسوند **in.** در اختیار شبیه ساز **ATLAS** قرار میدهد. خروجی های **Runtime** خروجی هایی هستند که در حین شبیه سازی نمایش داده می شوند. مانند شماره تکرار فعلی ، روش حل عددی معادلات و پارامترهای دیگر. **LOG files** حاوی مشخصه های دیوایس می باشند. **Solution files** حاوی اطلاعاتی است که برای رسم نمودار در برنامه **TONY PLOT** نیاز است. پس در ادامه به مراحل آماده کردن فایل ورودی در محیط **Deckbuild** می پردازیم. برای هر شبیه سازی باید مراحل زیر به ترتیب با دستورات لازم انجام شود:

***A) Structure specification by commands*: MESH, REGION, ELECTRODE, DOPING**
***B) Material models specification by commands:* MATERIAL, MODELS, CONTACT, INTERFACE**
***C) Numerical method selection by commands:* METHOD**
***D) Solution specification by commands:* LOG, SOLVE, LOAD, SAVE**
***E) Results analysis by commands:* EXTRACT, TONYPLOT**

# تعریف مش

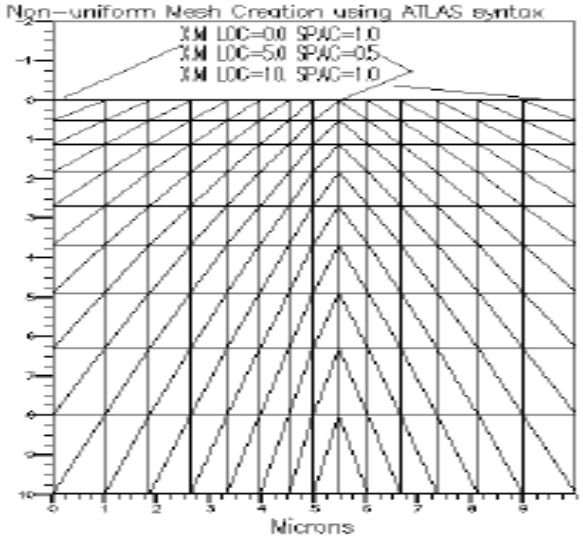
**تعریف مش شامل یک سرس خطهای عمودی و افقی با فواصل تعیین شده است. هر چه فواصل بیشتر باشد شبیه سازی سریعتر و دقت کمتر است. تخصیص مش بصورت زیر است:**

```
x.mesh Loc=<value> spac=<value>
y.mesh Loc=<value> spac=<value>
```

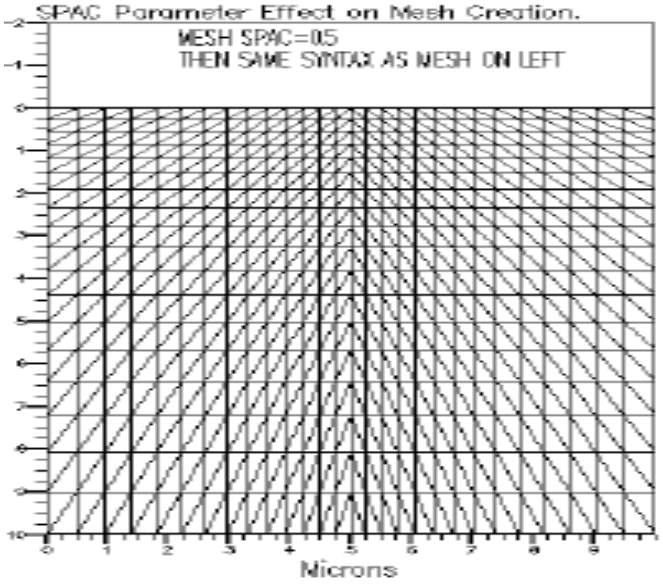
**مثلا:**

```
x.mesh loc=0.0 spac=1
x.mesh loc=5 spac=0.5
x.mesh loc=10 spac=1
```

**باعث می شود فاصله خط های عمودی از ۰ میکرومتر تا ۵ میکرومتر از ۱ میکرومتر تا ۰,۵ میکرومتر تغییر کند و از ۵ میکرومتر تا ۱۰ میکرومتر از ۰,۵ میکرومتر تا ۱ میکرومتر تغییر کند. برای Y نیز به همین ترتیب عمل میکنیم. شکل زیر بیانگر این حالت است.**




**با دستور Mesh.spac=0.5 میتوان تعداد مش ها را دو برابر کرد بصورت زیر:**



## تعریف نواحی

**کار تعریف مش به پایان رسید حال نوبت به تعریف نواحی مختلف ساختار می شود یعنی نواحی مختلف ساختار مورد نظر را با یک شماره و ماده مورد نظر آن ناحیه و مختصاتش مشخص کنیم:**

```
REGION number=<integer> <material_type> <position
parameters>
```

مثال:

```
region     num=1   material=GaAs y.min=0.055
region     num=2  material=AlGaAs y.max=0.055
x.composition=0.3
```

**توجه کنید که در دستور اول Y.max مشخص نشده است پس طبق مش بندی y.max مقدار موجود در Y.mesh را به خود گرفته و X.min و X.max هم به همین صورت در نظر گرفته می شود. در دستور دوم Y.min مشخص نشده که قاعدتا صفر می باشد یعنی از ۰,۰۵۵ میکرومتر تا انتهای دیوایس ناحیه شماره یک و از جنس گالیم آرسناید و از ۰ میکرومتر تا ۰,۰۵۵ ناحیه شماره دو و از جنس AlGaAs می باشد. X.compositionهم مقدار ترکیبی آلومنیوم می باشد ($Al_xGa_{1-x}As$.) حال نوبت به تعریف الکترود ها می رسد.**

## تعریف الکترودها

**حد اقل یک الکترود در شبیه سازی باید وجود داشته باشد. با دستور زیر نام و محل الکترود تعیین میشود.**

```
ELECTRODE NAME=<electrode name>
<position_parameters>
```

**در قسمت electrode name نام الکترود آورده می شود مثلا درین یا گیت یا سورس. در قسمت position مختصات قرارگرفتن الکترودها بوسیلهx.min,x.max,y.min,y.max تعیین میشود که پیش فرض همه صفر میباشد. اگر دو الکترود را با نامهای یکسان تعریف کنیم از لحاظ الکتریکی متصل به هم در نظر گرفته می شوند.از عبارات و length, top,left,right,bottom هم میتوان برای تعریف مختصات الکترودها استفاده کرد.مثلا:**

```
Electrode Name=gate LEFT Length=0.5
```

**الکترودی به نام سورس در بالا سمت چپ ساختار ایجاد می کند که باندازه ۰,۵ میکرومتر به سمت راست ادامه دارد. الکترود تعریف شده که در تماس با نیمه هادی است بصورت پیش فرض از نوع اهمی میباشد.اگر بخواهیم اتصال از نوع شاتکی باشد باید تابع کار تعریف شود بصورت زیر:**

```
Contact Name=<electrode name> workfunction
```

قسمت **Name** مربوط به نام الکترودی است که قبلا تعریف کرده ایم و اکنون میخواهیم برای آن تابع کار در نظر بگیریم. بجای نوشتن یک عدد بجای قسمت **work function** میتوان از نام مواد استفاده کرد مانند **N. Polysilicon** و**P.polysilicon** و **tungsten** یا **Tu.disilicide** . مثال:

```
CONTACT NAME=gate N.POLYSILICON
```

**وقتی از سدهای شاتکی استفاده میکنیم پیشنهاد می شود درزیر اتصال شاتکی و درون نیمه هادی مشهای **y** را **refine** کنیم تا شبیه سازی ناحیه تخلیه شاتکی با دقت بیشتری انجام شود. همچنین میتوانیم مانند زیر برای اتصال مقامت یا ظرفیت خازنی یا اندوکتانس تعریف کنیم:

```
CONTACT NAME=drain RESISTANCE=50.0 CAPACITANCE=20e-12 INDUCTANCE=1e-6
```

دستور بالا یک مقاومت و یک خازن موازی سری با یک سلف را برای الکترود درین در نظر میگیرد. البته واحد مثلا مقاومت **50 Ω-µm** است چون شبیه سازی دوبعدی است و عرض دیوایس **1um** در نظر گرفته می شود. وقتی مقاومت برای اتصال تعریف میکنیم روش حل باید **block** یا **Newton** باشد.

## ناخالصی ها

مرحله بعدی تعریف ناخالصی هاست. بوسیله عبارت زیر:

```
DOPING <distribution_type> <dopant_type> <position_parameters>
```

توزیع ناخالصی(**distribution_type**) در هر ناحیه دو نوع میتواند باشد یکنواخت و گاوسی. به مثالهای زیر توجه کنید:

```
DOPING UNIFORM CONCENTRATION=1E16 N.TYPE REGION=1

DOPING GAUSSIAN CONCENTRATION=1E18 CHARACTERISTIC=0.05 P.TYPE
X.LEFT=0.0 X.RIGHT=1.0 PEAK=0.1
```

در تعریف اول ناخالصی نوع **n** با مقدار **1E6** بطور یکنواخت در ناحیه یک که قبلا تعریف شده بود توزیع می شود.بازهم بجای نام ناحیه میتوانیم از**x.min,x.max,y.min,y.max** استفاده کنیم. در تعریف دوم ناخالصی از نوع **p** با مقدار ماکزیمم **1E18** در ناحیه بین **x=0** تا **x=1um** توزیع شده و از **1um** به بعد ناخالصی با یک توزیع گاوسی با انحراف معیار ۰,۵ کاهش می یابد.

## تعیین خصوصیات مواد:

تمام مواد کتابخانه **silvaco** دارای خصوصیات معینی هستند مثلا **bandgap,density of states,...** که هرکدام از این پارامترها مقدار مشخصی دارند.(**appendixB:material system**) . با دستور **material** می توانیم مقادیر را به دلخواه تغییر دهیم. مقادیر میتوانند برای یک ماده در کل دیوایس یا در ناحیه خاصی تغییر داده شوند. مثلا:

```
MATERIAL MATERIAL=Silicon EG300=1.12 MUN=1100
MATERIAL REGION=2 TAUN0=2e-7 TAUP0=1e-5
```

**عبارت اول باندگپ سیلیکون را و موبیلیتی الکترون را در تمام نواحی سیلیکنی دیوایس تغییر میدهد. در عبرت دوم طول عمر SRH را برای الکترون و حفره ها در ناحیه ۲ تغییر می دهد. بصورت زیر نیز میتوان نوشت یعنی با نام ناحیه تعریف شده در قسمت REGION .**

```
MATERIAL NAME=base NC300=3e19
```

**عبارت بالا چگالی حالتها را در لایه هدایت برای ناحیه ای که نامش بیس است تغییر میدهد.در SEC18.24 لیست تمامی پارامترهای موجود برای مواد آورده شده اند.**

## تعیین خواص سطوح فصل مشترک

**به کمک دستور INTERFACE برای تعیین چگالی بار در فصل مشترک بین نیمه هایدی و عایق استفاده می شود. بعلاوه میتوان سرعت بازترکیب سطحی را تغییر داد.**

```
INTERFACE     QF=3E10
```

**دستور بالا به تمامی سطوح مشرک عایق و نیمه هادی موجود چگالی بار تعیین شده را اعمال می کند. بعلاوه میتوانیم این چگالی بار را در ناحیه مشخصی از سطح مشترک عایق و نیمه هادی اعمال کرد. این کار بصورت زیر اعمال میشود:**

```
INTERFACE QF=3e10 X.MIN=1.0 X.MAX=2 Y.MIN=0.0
Y.MAX=0.5 S.N=1E4  S.P=1E4
```

**می توانیم بجای QF از پارامتر CHARGE استفاده کنیم می توانیم بار سطحی را بین هر دو فصل مشترکی مثل نیمه هادی-نیمه هادی(S.S) یا نیمه هادی – عایق(S.I) یا در لبه های نیمه هادی(S.X )تعریف کنیم. بصورت زیر:**

```
INTERFACE CHARGE=3e10 S.S X.MIN=1.0 X.MAX=2
Y.MIN=0.0 Y.MAX=0.5
```

****علامت بار مثبت یا منفی میتواند باشد.**

## انتخاب روش عددی حل

**چندین روش حل عددی معادلات نیمه هادی وجود دارد. سه روش مرسوم حل عبارتند از:**

```
GUMEL, NEWTON, BLOCK
```

روش گامل وقتی استفاده میشود که دستگاه معادلات ارتباط کمی با هم داشته باشند. همگرایی در ایـن روش خطی است. روش نیوتن تمام معادلات را با هم حل میکنـد و دارای همگرایـی درجـه دو اسـت.**BLOCK** بعضی از تعدادی از معادلات را با هم حل کرده و بقیه را مستقل فرض می کنـد. روش **NEWTON** زمـان زیادی را برای تعیین متغیرهایی که اساسا ثابت یا مستقل مـی باشـند صـرف میکنـد ولـی از دقـت بـالاتری برخوردار است. همچنین روش **NEWTON** احتیاج به حدسهای اولیه بهتری نسبت بـه بقیـه روشـها بـرای همگرا شدن دارد. از طرفی روش گامل می تواند حدسهای اولیه مناسبی را فراهم کند. یک روش مفید اینست که ابتدا حل با روش گامل شروع می شود و پس از چند تکرار روش حل به نیوتن سوئیچ شود. که با دسـتور زیر انجام میشود:

```
METHOD GUMEL NEWTON
```

## تعیین نوع حل(DC,AC,TRANSIENT)

در حل **dc** برای بدست آوردن منحنی های مشخصه **DC** معمولا یه یک الکترود ولتاژ اعمال کرده و جریان را در الکترود دیگر اندازه گیری میکنیم. با دستور **SOLVE** معادلات به ازاء ولتاژ مشخصی حل می شوند مثلا:

```
SOLVE VGATE=1.0
SOLVE VGATE=2.0
```

توجه داشته باشید که اگر برالکترودی در هر جای برنامه ولتاژی در نظر گرفته نشود حل با آخرین ولتاژاعمال شده به الکترودها انجام می شود. برای **sweep** کردن ولتاژ یکی از الکترودها بصورت زیر عمل میکنیم:

```
SOLVE VCOLLECTOR=2.0
SOLVE VBASE=0.0 VSTEP=0.05 VFINAL=1.0 NAME=base
```

اگر بخواهیم دسته ای از منحنی ها را بدست آوریم بصورت زیر عمل میکنیم:

```
SOLVE VGATE=1.0 OUTF=solve_vgate1
SOLVE VGATE=2.0 OUTF=solve_vgate2
SOLVE VGATE=3.0 OUTF=solve_vgate3

LOAD INFILE=solve_vgate1
LOG OUTFILE=mos_drain_sweep1
SOLVE NAME=drain VDRAIN=0 VFINAL=3.3 VSTEP=0.3

LOAD INFILE=solve_vgate2
LOG OUTFILE=mos_drain_sweep2
SOLVE NAME=drain VDRAIN=0 VFINAL=3.3 VSTEP=0.3

LOAD INFILE=solve_vgate3
LOG OUTFILE=mos_drain_sweep3
SOLVE NAME=drain VDRAIN=0 VFINAL=3.3 VSTEP=0.3
```

هر با ر که از عبارت **log** استفاده می شود در واقع اطلاعات**ID-vds** تا قبل از تعریف یک **LOG** دیگـر در فایل مربوطه ذخیره می شوند. از عبارت **solve init** قبل از همه **solve** ها استفاده میکنیم تا حـدس اولیـه های مناسبی برای پتانسیل و غلظت حامل ها بدست آید. البته اطلس یک سری الگوریتمها در مواجهه شدن بـا واگرایی دارد. در عمل دیده می شود که اطلس برای حل اولین و دومین نقطه بایاس همگرایی خـوبی نـدارد. یک روش این است که از اولین تا دومین نقطه بایاس ولتاژ را با گامهای کمتری اف زایش دهیم بـه دو مثـال زیر توجه کنید:

```
1. SOLVE INIT
SOLVE VDRAIN=0.1
SOLVE VDRAIN=0.2
SOLVE VDRAIN=2.0

2. SOLVE INIT
SOLVE VDRAIN=2.0
```

در مورد اول حل انجام شده به احتمال زاد همگراست اما دومی ممکن است نباشد. برای تحلیل های **AC,transient** به **sec2.9** مراجعه کنید.

## خروجی های اطلس:

خروجی **atlas** بر سه نوعند **Runtime Output, Log Files, Structure Files**. خروجی های **run time** در همان زمان شبیه سازی در پنجره پایین پنجره اصلی نمایش داده می شوند.در بخش ۲,۱۰ خروجی های بلادرنگ نمایش داده شده اند.

لوگ فایلها اطلاعات ترمینالها را در خود ذخیره میکنند. مثلا در حالت **DC** ولتاژو جریان الکترودها ذخیره میشوند. در حالت **transient** زمان و در حالت **AC** فرکانس رسانایی و ظرفیت خازنی ذخیره می شود.
عبارت:

```
LOG OUTF=<FILENAME>
```

یک فایل با نام مشخص را باز کرده و نتایج تمام عبارات **solve** در آن ذخی ره مـی شـود. ولتـاژ و جریـان ترمینالها تا زمانیکه فایل جدیدی باز نشود در همین لوگ فایل زخیره می شود. واحد ذخیره سازی جریـان در لوگ ها آمپر بر میکرون می باشد. از لوگ فایلها میتوان به استخراج پارامتر پرداخت:

```
EXTRACT INIT INF="<filename>"
```

مثلا دستور:

```
EXTRACT NAME="nvt" XINTERCEPT(MAXSLOPE(CURVE
(V."GATE", (I."DRAIN")))
-(AVE(V."DRAIN"))/2.0)
```

ولتاژ آستانه را استخراج می کند. نتایج استخراج در خروجی **runtime** نشان داده میشود همچنین در یک فای بنام **results.final** ذخیره میشوند. میتوانیم نتایج را در هر فایل دلخواهی ذخیره کنیم:

```
EXTRACT....DATAFILE="<filename>"
```

خروجی های نوع سوم **Solution files or structure files** هستند که یک تصویر از دیوایس را در یک بایاس معین (در شبیه سازی **dc**) ذخیره میکند. از **structure files** میتوان به ارزیابی هر متغیری درون ساختار مثل میدان و چگالی الکترون و ... پرداخت. این فایلها درون نرم افزار **tony plot** رسم می شوند. به دو صورت زیر میتوان چنین فایلهایی را تولید کرد:

```
(1) SAVE OUTFILE=<filename>
(2) SOLVE .... OUTFILE=<filename>.sta MASTER
[ONEFILEONLY]
```

در حالت دوم در هر بایاسی یک فایل ساختاری ذخیره میشود که حرف آخر پسوند **sta** یعنی **a** به **b,c,d,...** با افزایش بایاس ، تغییر میکند. اگر فقط حل را در نقطه بایاس پایانی می خواهیم **ONEFILEONLY** را اضافه میکنیم. متغیرهایی مانند **doping, electron concentration, potential** در فایلهای ساختاری ذخیره می شوند. میتوانیم متغیرهای دیگری را نیز با دستور**output** اضافه کنیم بطور مثال:

```
OUTPUT CON.BAND VAL.BAND
```

پتانسیل باند هدایت و ظرفیت را نیز د رخروجی ذخیره می کند.

برای رسم خروجی ها از لوگ فایلها در **tonyplot** بصورت زیر عمل می کنیم:

```
tonyplot -overlay file1.log file2.log
```

وبرای رسم از فایلهای ساختاری(**structure files**):

```
tonyplot file.str -set mx.set iv.data
```

**set file** ها مربوط به جزییات شکل رسم شده در **tonyplot** می باشند. وقتی مثلا ساختار یک دیوایس در **tonyplot** رسم میشود میتوانیم به شکل جزییات اضافه کرده و جزییات را از منوی **file** و گزینه **save set** **file** ذخیره کنیم و در دستور **tonyplot** استفاده کنیم. پس میتوانیم وقتی تعریف ساختار در برنامه به اتمام رسید یک بار دستور `tonyplot file.str` را اجرا کرده و شکل ساختار زسم شود سپس جزییات را اضافه کرده و ذخیره کنیم. البته این گزینه اختیاری است و نبودش مشکلی ایجاد نمی کند. حال دستورات مورد نیاز برای شبیه سازی یک دیود را مینویسیم مراحل را از نوشته های بالا دنبال کنید:

```
go atlas
```

```
# MESH SPECIFICATION PART
mesh space.mult=1.0
```

```
#
x.mesh loc=0.00 spac=0.5
x.mesh loc=3.00 spac=0.2
x.mesh loc=5.00 spac=0.25
x.mesh loc=7.00 spac=0.25
x.mesh loc=9.00 spac=0.2
x.mesh loc=12.00 spac=0.5
#
y.mesh loc=0.00 spac=0.1
y.mesh loc=1.00 spac=0.1
y.mesh loc=2.00 spac=0.2
y.mesh loc=5.00 spac=0.4
```

## # REGIONS AND ELECTRODES SPECIFICATION

```
region num=1 silicon
electr name=anode x.min=5 length=2
electr name=cathode bot
```

## # DOPING SPECIFICATION

```
#.... N-epi doping
doping n.type conc=5.e16 uniform
```

### #.... Guardring doping

```
doping p.type conc=1e19 x.min=0 x.max=3 junc=1
rat=0.6 gauss
doping p.type conc=1e19 x.min=9 x.max=12 junc=1
rat=0.6 gauss
```

### #.... N+ doping

```
doping n.type conc=1e20 x.min=0 x.max=12 y.top=2
y.bottom=5 uniform
```

## # SAVING THE MESH

```
save outf=diodeex01_0.str
tonyplot diodeex01_0.str -set diodeex01_0.set
```

## # MODELS SPECIFICATION

```
model conmob fldmob srh auger bgn
contact name=anode workf=4.97
```

## # SOLUTION PART

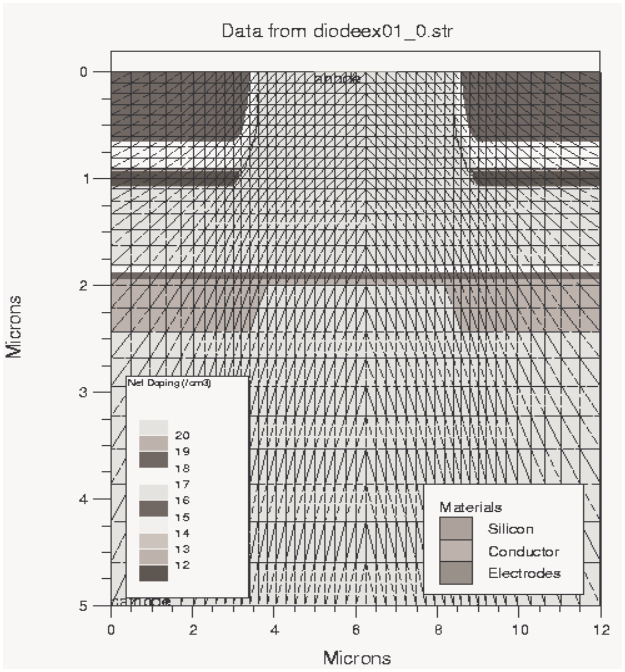
### #.... Initial solution part

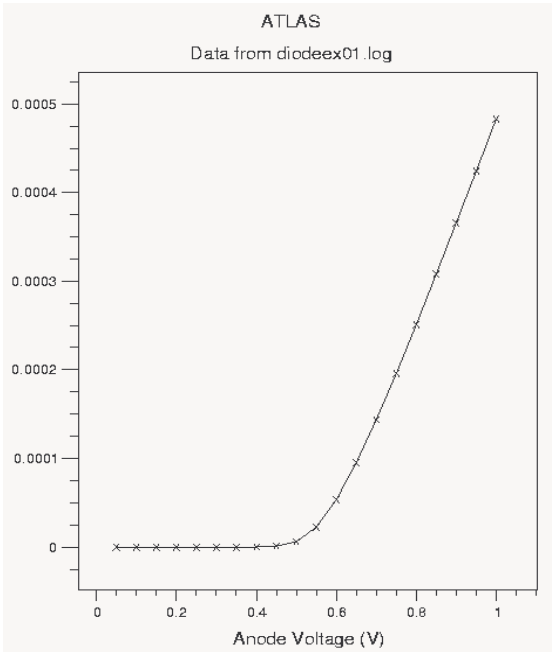
```
solve init
method newton
```

## #…. Stepping the anode voltage and saving the data

```
log outfile=diodeex01.log
Solve vanode=0.05 vstep=0.05 vfinal=1 name=anode
tonyplot diodeex01.log -set diodeex01_log.set
quit
```

**نتایج در پایین آورده شده اند:**



این مقدماتی برای انجام شبیه سازی با Atlas بود که در محیط deckbuild انجام میشود.هنگام شبیه سازی ممن است پس از گذشت مدتی از باز بودن محیط deckbuild با پیغام خطا ... SFLM SERVER مواجه شویم در این حالت شبیه سازی را با دکمه stop متوقف کرده و پیغامهای خطا را ok میکنیم سپس server status را آنقدر اجرا میکنیم تا licence را بشناسد.کمی حوصله کنید. یا وقتی نمیخواهید با برنامه کار کنید آن را ببندید.

تهیه کننده:سید احسان ابطحی abtahi.ehsan@gmail.com