PISCES-II Input Specification

CHECK 4
COMMENT 5
CONTACT 6
CONTOUR 9
DOPING 13
ELECTRODE 19
ELIMINATE 20
END 21
EXTRACT 22
IMPACT 24
INCLUDE 25
INTERFACE 26
LOAD 27
LOG 29
MATERIAL 30
MESH 33
METHOD 36
MODELS 40
OPTIONS 43
PLOT.1D 45
PLOT.2D 49
PRINT 51
REGION 54
REGRID 56
SOLVE 59
SPREAD 65
SYMBOLIC 69
TITLE 71
VECTOR 72
X.MESH 74
Y.MESH 74

Overview

1. Card format

PISCES-II takes its input from a user specified disk file. The input is read by GENII, the same input processor that is used in SUPREM and other Stanford programs. Each line is a particular statement, identified by the first word on the card. The remaining parts of the line are the parameters of that statement. The words on a line are separated by blanks or tabs. If more than one line of input is necessary for a particular statement, it may be continued on subsequent lines by placing a plus sign (+) as the first non-blank character on the continuation lines. Parameter names do not need to be typed in full; only enough characters to ensure unique identification is necessary.

Parameters may be one of three types : numerical, logical or character. Numerical parameters are assigned values by following the name of the parameter by an equal sign (=) and the value. Character parameters are assigned values by following the name of the parameter by an equal sign (=) and a sequence of characters. The first blank or tab delimits the string. The presence of a logical value indicates TRUE while a logical value preceded by a caret (ˆ) indicates FALSE.

In the card descriptions which follow, the letters required to identify a parameter are printed in upper case, and the remainder of the word in lower case. A phrase enclosed in angle brackets <...> represents a parameter list to be explained in further detail. Bars in the right-hand margin denote changes in this version of the manual.

2. Card sequence

The order of occurrence of cards is significant in some cases. Be aware of the following dependencies:

→ The MESH card must precede all other cards, except TITLE, COMMENT and OPTIONS.

→ When defining a rectangular mesh, the order of specification is

MESH

X.MESH *(all)*

Y.MESH *(all)*

ELIMINATE

SPREAD

REGION

ELECTRODE

ELIMINATE and SPREAD cards are optional but if they occur they must be in that order.

→ DOPING cards must follow directly after the mesh definition

→ Before a solution, a symbolic factorization is necessary. Unless solving for the equilibrium condition, a previous solution must also be loaded to provide an initial guess.

→ Any CONTACT cards must precede the SYMBOLIC card.

→ Physical parameters **may not** be changed using the MATERIAL, CONTACT or MODEL cards after the first SOLVE or LOAD card is encountered. The MATERIAL and CONTACT cards precede the MODEL card.

→ A PLOT.2D must precede a contour plot, to establish the plot bounds.

→ PLOT.2D, PLOT.1D, REGRID or EXTRACT cards which access solution quantities (ψ, n, p, ϕ_n, ϕ_p, currents, recombination) must be preceded somewhere in the input deck by a LOAD or SOLVE card to provide those quantities.

3. References (used below)

[1] Mark R. Pinto, Conor S. Rafferty and Robert W. Dutton, ‘‘PISCES-II - Poisson and Continuity Equation Solver,’’ Stanford Electronics Laboratory Technical Report, Stanford University, September 1984.

[2] J. T. Watt, "Improved Surface Mobility Models in PISCES," presented at Computer-Aided Design of IC Fabrication Processes, Stanford University, August 6, 1987.

[3] J. T. Watt and J. D. Plummer, "Universal Mobility-Field Curves for Electrons and Holes in MOS Inversion Layers," presented at 1987 Symposium on VLSI Technology, Karuizawa, Japan.

[4] W. N. Grant, "Electron and Hole Ionization Rates in Epitaxial Silicon at High Electric Fields," *Solid State Electron.*, vol. 16, pp. 1189-1203, 1973.

[5] S. E. Laux, "A General Control-volume Formulation for Modeling Impact Ionization in Semiconductor Transport," *IEEE Trans. Computer-Aided Design*, vol. CAD-4, pp. 520-526, Oct., 1985.

[6] G. A. Baraff, "Distribution Functions and Ionization Rates for Hot Electrons in Semiconductors," *Physical Review*, vol. 128, pp. 2507-2517, 1962.

[7] C. R. Crowell, and S. M. Sze, "Temperature Dependence of Avalanche Multiplication in Semiconductors," *Applied Physics Letters*, 9, pp. 242-244, 1966.

[8] N. Arora, J. Hauser and D. Roulston, "Electron and Hole Mobilities in Silicon as a Function of Concentration and Temperature," *IEEE Trans. Electron Devices*, vol. ED-29, pp. 292-295, 1982.

[9] J. M. Dorkel and P. Leturcq, "Carrier Mobilities in Silicon Semi-Empirically Related to Temperature, Doping and Injection Level," *Solid State Electron.*, vol. 24, pp. 821-825, 1981.

[10] H. Shin, A. Tasch, C. Maziar and S. Banerjee, "A New Approach to Verify and Derive a Transverse-Field-Dependent Mobility Model for Electrons in MOS Inversion Layers," *IEEE Trans. Electron Devices*, vol. ED-36, pp. 1117-1123, 1989.

[11] V.M. Agostinelli, H. Shin, A. Tasch, "A Comprehensive Model of Inversion Layer Hole Mobility for Simulation of Submicron Mosfets," NUPAD III Technical Digest, pp. 39-40, 1990.

[12] S. Schwarz and S. Russek, "Semi-Empirical Equations for Electron Velocity in Silicon: Part II--MOS Inversion Layer," *IEEE Trans. Electron Devices*, vol. ED-30, pp. 1634-1639, 1983.

† Note: All equations referenced in this appendix can be found in the PISCES-IIA manual [1] unless otherwise stated (eg. an equation suffixed with ‘‘IIB-sm’’ can be found in the PISCES-IIB supplementary manual).

The CHECK card

1. Syntax

```
CHeck <file specification>
```

2. Description

The CHECK card compares a specified solution against the current solution, returning the maximum and average difference in electrostatic and quasi-fermi potentials. The check card is particularly useful for comparing solutions that have been obtained on different generations of regrids.

3. Parameters

<file specification>

```
Infile      \=    <filename>
Mesh        \=    <filename>
Samemesh    \=    <logical>      (default is false)
```

INFILE specifies the name of the solution file to compare. MESH is the name of the file containing the mesh for that solution. SAMEMESH indicates the the solution in INFILE used the same mesh as the current solution.

4. Examples

Compare solution ‘‘sol1’’ obtained using mesh ‘‘mesh1’’ against the current solution.

```
CHECK    INFILE=sol1  MESH=mesh1
```

The COMMENT card

1. Syntax

```
COMment <character string>

                    or

$ <character string>
```

2. Description

The COMMENT card allows comments to be placed in the PISCES-II input file. PISCES-II ignores the information on the COMMENT card.

3. Examples

```
$   **** This is a comment - wow! ****
```

The CONTACT card

1. Syntax

CONTAct <number> <workfunction> <special conditions>

2. Description

The CONTACT card defines the physical parameters of an electrode. If no contact card is supplied for an electrode, it is assumed to be charge-neutral (ohmic). Lumped elements are also specified here.

3. Parameters

<number>

NUmber \= <integer>
or
ALL \= <logical>

The number must be that of a previously defined electrode. Using ALL instead of <number> defines the same properties for all electrodes.

<workfunction>

One of :

Material	*Work function used*
NEutral	(calculated from doping)
ALUminum	4.17
P.polysilicon	$4.17 + Egap$
N.polysilicon	4.17
MOLybdenum	4.53
TUNgsten	4.63
MO.disilicide	4.80
TU.disilicide	4.80

or

Workfunction \= <real>

The work function is set to the above values for the standard materials, or to the given value. The value is interpreted in volts. NEUTRAL (the default) stands for charge-neutral (ohmic).

<special conditions>

Surf.rec	\=	<logical>	(default false)
VSURFN	\=	<real>	Surface recombination rate for electrons
VSURFP	\=	<real>	Surface recombination rate for holes
BArrierl	\=	<logical>	(default false)
ALPha	\=	<real>	(default is 0)
	or		
CUrrent	\=	<real>	(default is false)
	or		
Resistance	\=	<real>	(default is 0)
	or		
CApacitance	\=	<real>	(default is 0)
	or		
COn.resist	\=	<real>	(default is 0)

If none of the above are specified, this contact is a normal ohmic (Dirichlet) boundary condition. If SURF.REC is specified, finite surface recombination velocities are used at the respective contact. VSURFN and VSURFP have the units cm/sec; their defaults are calculated using equations 2.45 and 2.46 [1]. BARRIERL is a flag to turn-on the barrier lowering mechanism, while ALPHA is the linear, dipole barrier lowering coefficient (see chapter 2 of this report) and has the units of cm. CURRENT denotes a current boundary condition. RESISTANCE specifies a lumped resistance value ($\Omega - \mu$m) and CAPACITANCE a lumped capacitance value (Ω / μm) to be attached to the contact. CON.RESIST is a distributed contact resistance in $\Omega - \mathrm{cm}^2$. See chapter 2 of this report for details on current, lumped element and distributed-resistance boundary conditions.

4. Examples

Define all electrodes except number 2 to be neutral, and number 2 is aluminum. Besides a workfunction, electrode number 2 also includes finite surface recombination velocities and barrier lowering. Note that the definition on the second card overrides that of the first.

```
CONTACT ALL NEUTRAL
CONTACT NUM=2 ALUM SURF BARR
```

Attach a lumped resistor to contact number 2 with a value of $10^5 \Omega - \mu$m. Include distibuted contact resistance ($10^{-6} \Omega - \mathrm{cm}^2 \approx$ Aluminum) on contact 4.

```
CONTACT NUM=2 RESIS=1E5
CONTACT NUM=4 CON.RES=1E-6
```

The CONTOUR card

1. Syntax

```
CONTOur <plotted quantity> <range definition> <control>
```

2. Description

The CONTOUR card plots contours on a plotted two-dimensional area of the device, as specified by the most recent PLOT.2D card.

3. Parameters

<plotted quantity>

One of:

POtential	\=	<logical>	Mid-gap potential
QFN	\=	<logical>	Electron quasi-fermi level
QFP	\=	<logical>	Hole quasi-fermi level
Valenc.b	\=	<logical>	Valence band potential
CONduc.b	\=	<logical>	Conduction band potential
DOping	\=	<logical>	Doping
ELectrons	\=	<logical>	Electron concentration
Holes	\=	<logical>	Hole concentration
NET.CHarge	\=	<logical>	Net charge concentration
NET.CArrier	\=	<logical>	Net carrier concentration
J.Conduc	\=	<logical>	Conduction current
J.Electr	\=	<logical>	Electron current
J.Hole	\=	<logical>	Hole current
J.Displa	\=	<logical>	Displacement current
J.Total	\=	<logical>	Total current
E.field	\=	<logical>	Electric field
Recomb	\=	<logical>	Net recombination
Flowlines	\=	<logical>	Current flow lines
ALPHAN	\=	<logical>	Ionization rate for electrons (/cm)
ALPHAP	\=	<logical>	Ionization rate for holes (/cm)
IMPACT	\=	<logical>	Generated carrier density due to impact ionization (/cm2)

The above parameters specify the quantity to be plotted. For vector quantities the magnitude is plotted. Model dependent parameters (current and recombination) are calculated with the models currently defined, *not* with the models that were defined when the solution was computed. This allows the display of, for instance, Auger and Shockley-Read-Hall components of recombination separately. For consistent values of current, the models used in the solution should be specified. The quantity to be plotted has no default.

<range definition>

```
MIn.value     \=    <real>
MAx.value     \=    <real>
DEl.value     \=    <real>
NContours     \=    <integer>
```

MIN.VALUE and MAX.VALUE specify the minimum and maximum contours to be plotted. MIN.VALUE and MAX.VALUE default to the minimum and maximum values of the quantity to be plotted over the device (these are printed during execution). DEL.VALUE specifies the interval between contours. Alternatively, NCONTOURS specifies the number of contours to be plotted. If the plot is logarithmic, the minimum and maximum should be given as the logarithmic bounds.

<control>

LIne.type	\=	<integer>	(default is 1)
ABsolute	\=	<logical>	(default is false)
LOgarithm	\=	<logical>	(default is false)
NEGative	\=	<logical>	(default is false)
X.compon	\=	<logical>	(default is false)
Y.compon	\=	<logical>	(default is false)
PAuse	\=	<logical>	(default is false)
COLor	\=	<logical>	(default is false)
C1.color	\=	<integer>	(default is 6)
C2.color	\=	<integer>	(default is 7)
C3.color	\=	<integer>	(default is 8)
C4.color	\=	<integer>	(default is 9)
C5.color	\=	<integer>	(default is 10)
C6.color	\=	<integer>	(default is 11)
C7.color	\=	<integer>	(default is 12)
C8.color	\=	<integer>	(default is 13)
C9.color	\=	<integer>	(default is 14)
C0.color	\=	<integer>	(default is 15)

LINE.TYPE defines the plot line type. ABSOLUTE specifies that the absolute value of the variable be taken. For rapidly varying quantities, the LOGARITHM is often more revealing. Since many of the quantities may become negative, PISCES-II actually uses

$\log(x) = \mathrm{sign}(x) \times \log_{10}(1 + |x|)$ to avoid overflow. To get the true logarithm of a quantity, specify ABSOLUTE and LOGARITHM - the absolute is taken first and there is no danger of negative arguments. NEGATIVE negates the values of thevariables plotted. X.COMPON and Y.COMPON take the x and y components of a vector quantity, respectively. The PAUSE option causes PISCES-II to stop at the end of the plot so that a hardcopy may be made before continuing. Execution can be resumed by hitting a carriage return. COLOR specifies that color fill, as opposed to simple lines, should be used to delineate contours. C1.COLOR, C2.COLOR, etc. specify the color types for the contours.

4. Examples

The following plots the contours of potential from -1 volts to 3 volts in steps of .25 volts:

```
CONTOUR  POTEN MIN=-1 MAX=3 DEL=.25
```

In the next example, the log of the doping concentration is plotted from 1.0e10 to 1.0e20 in steps of 10. By specifying ABSOLUTE, both the n-type and p-type contours are shown.

```
CONTOUR  DOPING MIN=10 MAX=20 DEL=1 LOG ABS
```

In the following, current flow lines are plotted. The number of flow lines is 11 so that 10% of the current flows between adjacent lines.

```
CONTOUR  FLOW NCONT=11
```

The DOPING card

1. Syntax

DOping <profile type> <location> <region> <profile specification> <save>

2. Description

The DOPING card dopes selected regions of the device.

3. Parameters

<profile type>

One of the following types must be selected.

Erfc
Gaussian
Uniform
SUprem3
OLD.Suprem3
AScii
S4geom
SImpldop

ERFC, GAUSSIAN and UNIFORM are used to analytically describe profile shapes. SUPREM3 reads data files produced by the more recent release of SUPREM-III, which supports the "*export*" output file format. The default is to read binary formatted export files. If the ASCII parameter is also specified, then ASCII (text) formatted SUPREM-III export files will be expected. OLD.SUPREM3 is used to get profile information from the original version of the SUPREM-III process modeling program. This reads the binary structure file. ASCII without the SUPREM3 parameter allows for input of simple ascii data files containing concentration versus depth information. The format of the ASCII input file is a depth in μm followed by a concentration in cm^{-3} − one pair per line. By convention, positive concentrations refer to donors (n-type), while negative concentration values refer to acceptors (p-type). S4GEOM allows doping from a SUPREM-IV file (using the struct pisces=foo command in SUPREM-IV) to be interpolated onto an existing PISCES-II mesh. SIMPLDOP takes doping from a SIMPL-2 file (rectangular grid of doping values) to be interpolated onto an existing PISCES-II mesh.

<location>

Parameter	DEFAULT		
	(Uniform)	(Erfc/Gaussian/Suprem/Ascii)	
		(x-direction)	(y-direction)
X.Left \= <real>	−∞	*SP*	−∞
X.Right \= <real>	∞	X.LEFT	∞
Y.Top \= <real>	−∞	−∞	Y.BOT
Y.Bottom \= <real>	∞	∞	*SP*

The box given by the X and Y bounds locates the profile within a device, and defines an area (or line) where the profile is constant (mask edges). Outside this area, it falls off along the principal axis according to the profile type, and along the lateral axis according to the lateral specifications. The default bounds of the box are chosen depending on the type and principal direction of the profile. In the Erfc/Gaussian/Suprem case, the bounds default to a line perpendicular to the principal axis and located at the peak/start of the profile, respectively. This is denoted by the entry SP in the above table.

<region>

REgion \= <integer>

Region number where doping is to be added (optional). Multiple regions may be included by concatenating their region numbers into a single integer. If no region is specified, the entire semiconductor portion of the device is used.

If <profile type>=S4GEOM the following profile specification is relevant:

<Input filename>

Infile \= <filename>

The doping is obtained from Infile and interpolated onto an existing PISCES-IIB mesh.

If <profile type>=SIMPLDOP the following profile specification is relevant:

<Input filename>

Infile \= <filename>

The doping is obtained from Infile and interpolated onto an existing PISCES-IIB mesh.

If <profile type>=SUPREM3 or OLD.SUPREM3, the following profile specifications are relevant:

<Input filename>

Infile	\=	<filename>

<dopant>: One of

Boron	\=	<logical>
PHosphorus	\=	<logical>
ARsenic	\=	<logical>
ANtimony	\=	<logical>

The selected dopant profile will be extracted from the SUPREM-III save file.

<Two-dimensional spread>

Parameter			Default
DIrection	\=	x or y	y
STart	\=	<real>	0
RAtio.Lateral	\=	<real>	0.8

This group of parameters specifies where to locate the one-dimensional profile in the 2-dimensional device and how to extend it to the second dimension. DIRECTION is the axis along which the profile is directed. START is the depth in the specified direction where the profile should start, and should normally be at the surface. The lateral profile is assumed to have the same form as the principal profile, but shrunk/expanded by the factor RATIO.LATERAL. The defaults for the location box are set up as a line, parallel to the surface, and located at START.

If <profile type>=ASCII the following profile specifications are relevant:

<Input filename>

Infile	\=	<filename>

The ascii concentrations in INFILE are read and added to the impurity profiles.

<Two-dimensional spread>

As above - see description for SUPREM3 and OLD.SUPREM3.

If <profile type>= GAUSSIAN, the following profile specifications are relevant:

<profile>

COncentration	\=	<real>	(no default)	(cm^{-3})
JUnction	\=	<real>	(no default)	(μm)
SLice.lat	\=	<real>	(see below)	(μm)

or

DOSe	\=	<real>	(no default)	(cm^{-2})
CHaracteristic	\=	<real>	(no default)	(μm)

or

COncentration	\=	<real>	(no default)	(cm^{-3})
CHaracteristic	\=	<real>	(no default)	(μm)

and one of :

N.type/DONor	\=	<logical>	(no default)
P.type/ACceptor	\=	<logical>	(no default)

and any combination of :

RAtio.Lateral	\=	<real>	(default: 0.8)	
Erfc.Lateral	\=	<logical>	(default: false)	
Lat.char	\=	<real>	(see below)	
PEak	\=	<real>	(default: 0)	(μm)
DIrection	\=	<character>	(default: y)	

If <profile type>= ERFC, the following profile specifications are relevant:

<profile>

COncentration	\=	<real>	(no default)	(cm^{-3})
JUnction	\=	<real>	(no default)	(μm)
J.Conc	\=	<real>	(CO/100)	(cm^{-3})
SLice.lat	\=	<real>	(see below)	(μm)

or

COncentration	\=	<real>	(no default)	(cm^{-3})
CHaracteristic	\=	<real>	(no default)	(μm)

and one of :

N.type/DONor	\=	<logical>	(no default)

P.type/ACceptor	\=	<logical>	(no default)	

and any combination of :

RAtio.Lateral	\=	<real>	(default: 0.8)	
Erfc.Lateral	\=	<logical>	(default: false)	
Lat.char	\=	<real>	(see below)	
PEak	\=	<real>	(default: 0)	(μm)
DIrection	\=	<character>	(default: y)	

These parameters govern the profile outside the constant box. DIRECTION defines the principal axis of the profile. CONCENTRATION is the peak concentration, DOSE the total dose. J.CONC is the concentration at the junction, JUNCTION is the location of the junction and must be in silicon, outside the constant box; CHARACTERISTIC is the principal characteristic length. The peak concentration and principal characteristic length are computed from the given combination of the first four parameters. When JUNCTION is used, PISCES-II computes the characteristic length by examining the doping at a point half way between the ends of the constant box and at the given depth; if some other lateral position is desired for the computation, use the parameter SLICE.LATERAL=<real>. The lateral impurity profile may be an error function instead of gaussian, and its characteristic length is either the product of RATIO.LATERAL and the principal characteristic length (default) or can be specified using LAT.CHAR. PEAK specifies the position of the peak. The defaults for the constant box are set up as a line, parallel to the surface and located at PEAK.

If <type>=UNIFORM the following parameters are relevant:

<concentration>

COncentration	\=	<real>
N.type/DONor	\=	<logical>
P.type/ACceptor	\=	<logical>

Concentration is the value of the uniform doping level. It should be given in units of atoms/cm^3 and be positive. The polarity is given by the logical parameters. Doping is introduced in the intersection of the box and the region selected. The default box is set up to include the entire region.

<save>

Outfile	\=	<filename>

The save option allows the user to save a machine-readable copy of all the DOPING cards in a file. The first DOPING card should have the OUTFILE parameter, so that the doping information on it and all subsequent DOPING cards are saved in that file. The file can be reread after regrid to calculate the doping on the new grid.

4. Examples

A one-dimensional diode with substrate doping 10^{16} cm^{-3} and Gaussian profile.

```
DOP   UNIF CONC=1E16 P.TYPE
DOP   GAUSS CONC=1E20 JUNC=0.85 N.TYPE PEAK=0
```

An n-channel MOSFET with Gaussian source and drain. Because the default X.RIGHT is $+\infty$, for the source we must limit the constant part to X.RIGHT=4, and conversely for the drain. Thus the profile has a constant part along the surface, falls off as an error function towards the gate, and as a gaussian in the direction of the bulk. In both cases, the vertical junction is at $1.3\mu m$.

```
DOP  UNIF CONC=1E16 P.TYPE
DOP  GAUSS CONC=9E19 N.TYPE
+    X.RIGHT=4 JUNC=1.3 R.LAT=0.6 ERFC.LAT
DOP  GAUSS CONC=9E19 N.TYPE
+    X.LEFT=12 JUNC=1.3 R.LAT=0.6 ERFC.LAT
```

Read a Suprem bipolar profile and add it to a uniform substrate concentration. Add doping only to those points lying in region 1.

```
COM  *** SUBSTRATE ***
DOP  REGION=1 UNIF CONC=1E16 N.TYPE
COM  *** BASE ***
DOP  REGION=1 ASCII SUPREM   BORON R.LAT=0.7 INF=plt3.out1
+    START=0
COM  *** EMITTER ***
DOP  REGION=1 ASCII SUPREM   PHOS  R.LAT=0.8 INF=plt3.out1
+    X.LEFT=12.0 X.RIGHT=13.0 START=0
```

Simulate a triple diffused bipolar by using a mixture of analytic and SUPREM-III profiles. Use an erfc for the emitter, a SUPREM-III profile for the base, a gaussian for the collector, and add it to a uniform substrate concentration. Add doping only to those points lying in region 1.

```
COM *** SUBSTRATE ***
DOP REGION=1 UNIFORM CONC=9.999463e+14 p.type
COM *** EMITTER ***
DOP REGION=1 ERFC N.TYPE CON=1e20 CHAR=0.1
+  X.LEF=-1 X.RIG=0 R.LAT=0.8
COM *** BASE ***
DOP REGION=1 SUPREM3 INFILE=base.exp BORON
+      X.LEF=-4 X.RIG=0 R.LAT=0.8
COM *** COLLECTOR ***
DOP REGION=1 GAUSS PHOS CON=1e17 CHAR=0.8
+      X.LEF=-7 X.RIG=0 R.LAT=0.8
```

The ELECTRODE card

1. Syntax

```
ELEctrode <number> <position>
```

2. Description

The ELECTRODE card specifies the location of electrodes in a rectangular mesh.

3. Parameters

<number>

Number	\=	<integer>

There may be up to ten electrodes, numbered 1,2,3,...,9,0. They may be assigned in any order, but if there are N electrodes, none can have an electrode number above N.

<location>

IX.Low	\=	<integer>
IX.High	\=	<integer>
IY.Low	\=	<integer>
IY.High	\=	<integer>

Nodes having x and y indices between IX.LOW and IX.HIGH and between IY.LOW and IY.HIGH respectively are designated electrode nodes.

4. Examples

Define a typical back-side contact.

```
ELEC   N=1  IX.LOW=1  IX.HIGH=40  IY.LOW=17  IY.HIGH=17
```

The ELIMINATE card

1. Syntax

```
ELIMINATE <range> <direction>
```

2. Description

The ELIMINATE card terminates mesh points along lines in a rectangular grid.

3. Parameters

<direction>

X.direction	\=	<logical>
Y.direction	\=	<logical>

These parameters determine whether to eliminate points along vertical or horizontal lines. One must be chosen.

<range>

IX.Low	\=	<integer>
IX.High	\=	<integer>
IY.Low	\=	<integer>
IY.High	\=	<integer>

Points along every second line within the chosen range is removed. Successive eliminations of the same range remove points along every fourth, eighth line, and so on. For horizontal elimination, the vertical bounds should be decreased by one at each re-elimination of the same region, and conversely for vertical eliminations.

4. Examples

Points along vertical lines between 10 and 20 are removed.

```
ELIM    Y.DIR IY.LO=10 IY.HI=20 IX.LO=1 IX.HI=8
ELIM    Y.DIR IY.LO=10 IY.HI=20 IX.LO=1 IX.HI=7
```

The END card

1. Syntax

ENd

2. Description

The END card specifies the end of a set of PISCES-II input cards. The END card may be placed anywhere in the input deck; all input lines below the occurrence of the END card will be ignored. If an END card is not included, all cards in the input file are processed.

The EXTRACT card

1. Syntax

EXtract <variable> <bounds> <file i/o>

2. Description

The EXTRACT card extracts selected electrical data from the solution.

3. Parameters

<variable>

NET.CHar	\=	<logical>	Integrated net charge
NET.CArr	\=	<logical>	Integrated carrier concentration
Electron	\=	<logical>	Integrated electron concentration
Hole	\=	<logical>	Integrated hole concentration
Metal.Ch	\=	<logical>	Integrated charge on a contact
N.Resist	\=	<logical>	n-Resistance of a cross section
P.Resist	\=	<logical>	p-Resistance of a cross section
N.Current	\=	<logical>	n-current through an electrode
P.Current	\=	<logical>	p-current through an electrode

The net carrier, charge, electron or hole concentrations can be integrated over a section of a device. The charge on a part of an electrode can be calculated, as can the current through that part. This is useful for capacitance studies, in conjunction with the difference mode of the load card. The resistance of a cross sectional structure, for instance a diffused line, can be calculated.

<bounds>

X.MIn	\=	<real>
X.MAx	\=	<real>
Y.MIn	\=	<real>
Y.MAx	\=	<real>
Contact	\=	<integer>
Regions	\=	<integer>

Only nodes falling within the rectangle X.MIN-Y.MAX are included in the integrations. The default bounds include the entire device. For electrode quantities (current and metal charge) a CONTACT must be chosen; only nodes falling within the bounds and belonging to the contact are included in the integration. REGIONS can be optionally specified, forcing integration only on nodes within the specified bounds that are also part of a particular set of regions.

<file i/o>

Outfile \= <filename>

An optional ascii OUTPUT file can be specified to which the result and bias infor will be written.

4. Examples

The following extracts the resistance of a p-type line diffused into a lightly doped n substrate. Since the p-conductivity of the substrate is negligible, the bounds of the integration can include the whole device.

```
EXTRACT    P.RESIST
```

In the next example, the charge on the lower surface of a gate electrode is integrated. There is 0.05 μm of gate oxide on the surface, which is at y=0.

```
EXTRACT    METAL.CH  CONT=1  X.MIN=-2.0  X.MAX=2.0
+               Y.MAX=-0.0499  Y.MIN=-0.0501
```

The IMPACT card

1. Syntax

```
IMpact <parameters>
```

2. Description

The IMPACT card specifies the use of the impact ionization model. For many devices, the impact ionization model for continuity equations allows the accurate prediction of a valanche breakdown. Baraff's model ([6]) has been approximated with compact formulae by Crowell and Sze ([7]). The current models are for Si only. See also, the IMPACT parameter to the MODEL card and the impact parameters to the CONTOUR card.

3. Parameters

CRowell	\=	<logical>	(default is false)
MOnte	\=	<logical>	(default is false)
LAMDAE	\=	<real>	(default is 6.2e-7)
LAMDAH	\=	<real>	(default is 3.8e-7)

CROWELL specifies the use of Crowell and Sze formulae. if MONTE is given then the alpha values are extracted by Monte Carlo simulation. If CROWELL is not specified, Grant's model [4] is used. The basic implementation idea follows Laux [5], but using Scharfetter-Gummel current discretization formula without a weighting scheme. LAMBDAE and LAMDAH specify the mean free path in *cm* for electrons and holes respectively.

Also the Newton method with 2-carrier must be specified on the METHOD card since impact ionization is a 2-carrier process.

4. Examples

Use the Crowell and Sze formulae with the default mean free paths.

```
IMPACT   CROWELL  LAMDAE=6.2e-7   LAMDAH=3.8e-7
```

The INCLUDE card

1. Syntax

```
INClude <filename>

                    or

SOURCE <filename>
```

2. Description

The INCLUDE statement provides a shorthand way to include information from other files in the PISCES-II input file. The statements in the INCLUDEd file will be inserted into the PISCES-II input file in place of the INCLUDE statement when the input file is processed. The statements in the INCLUDEd file must use correct PISCES-II input syntax, and they must be in correct order with respect to the other statements in the PISCES-II input file when the INCLUDEd file is expanded by the input parser. This is most useful for libraries of material and model parameters.

3. Examples

```
INCLUDE MAT.init
```

The INTERFACE card

1. Syntax

Interface <parameters> <location>

2. Description

The INTERFACE card allows the specification of interface parameters (recombination velocities and fixed charges) at semiconductor-insulator boundaries.

3. Parameters

<parameters>

S.N	\=	<real>	Electron surface recombination velocity (cm/sec)
S.P	\=	<real>	Hole surface recombination velocity (cm/sec)
Qf	\=	<real>	Fixed charge density (cm^{-2}).

See chapter 2 of this report for a description of interface surface recombination velocities.

<location>

X.Min	\=	<real>
X.Max	\=	<real>
Y.Min	\=	<real>
Y.Max	\=	<real>

X.MIN, X.MAX, Y.MIN and Y.MAX define a bounding box, measured in μm. Any oxide/semiconductor interfaces found within this region are charged. A non-planar surface may be defined by using a box which contains the whole device, provided there is only one interface in the device.

4. Examples

Define an interface with both fixed charge and recombination velocities.

```
INTERFACE  X.MIN=-4 X.MAX=4 Y.MIN=-0.5 Y.MAX=4
+          QF=1E10 S.N=1E4 S.P=1E4
```

The LOAD card

1. Syntax

Load <solution files>

2. Description

The LOAD card loads previous solutions from files for plotting or as initial guesses to other bias points.

3. Parameters

<solution files>

INFile (or IN1file)	\=	<filename>	
IN2file	\=	<filename>	
Outdiff	\=	<filename>	
Differ	\=	<logical>	(default is false)
Ascii	\=	<logical>	(default is false)
No.check	\=	<logical>	(default is false)

The INFILE (or IN1FILE) and IN2FILE parameters specify input files names for solution data and may be up to 20 characters in length. INFILE (or IN1FILE) and IN2FILE represent a present and previous solution respectively. If only one solution is to be loaded (for plotting or as a single initial guess using the PREVIOUS option on the SOLVE card) then INFILE should be used. If two input files are needed to perform an extrapolation for an initial guess (i.e., the PROJECT option on the SOLVE card), IN1FILE and IN2FILE should be used. The solution in IN2FILE is the first to be lost when new solutions are obtained. The difference between two solutions (IN1FILE-IN2FILE) can be analyzed by reading in both with the mode DIFFER set. The difference is stored; this solution may not be used as an initial guess, or for any purpose other than plotting or extracting data. The difference solution may also be stored in another file using the parameter OUTDIFF. ASCII specifies that any files read or written by this card should be ascii rather than binary. NO.CHECK prevents PISCES-II from checking material parameter differences between the loaded files and the values set in the current PISCES-II input file. Checking is never done for ascii solution files.

4. Examples

The following specifies that a single solution file called SOL.IN should be loaded.

```
LOAD    INF=SOL.IN
```

In the next example, two solutions are loaded. The present solution is to SOL1.IN and the previous solution is SOL2.IN. We intend to use SOL1.IN and SOL2.IN to project an initial guess for a third bias point.

```
LOAD  IN1F=SOL1.IN  IN2F=SOL2.IN
```

Finally, two solutions are loaded, and the difference calculated and stored in a third file.

```
LOAD  IN1F=SOL1.IN  IN2F=SOL2.IN  DIFF  OUTD=SOL1-2
```

The LOG card

1. Syntax

```
LOG <file specification>
```

2. Description

The LOG card allows the I-V and/or AC characteristics of a run to be logged to disk. Any I-V or AC data subsequent to the card is saved. If a log file is already open, it is closed and a new file opened.

3. Parameters

<file specification>

```
Outfile or Ivfile   \=   <filename>
Acfile              \=   <filename>
```

OUTFILE or IVFILE specify the log file for I-V information. ACFILE specifies the file for AC data.

4. Examples

Save the I-V data in a file called IV1 and AC data in AC1.

```
LOG OUTF=IV1 ACFILE=AC1
```

The MATERIAL card

1. Syntax

MAterial <region> <material definitions>

2. Description

The material card associates physical parameters with the materials in the mesh. Many of the parameters are default for standard materials. Any equation numbers referred to below correspond to [1].

3. Parameters

<region>

NUmber	\=	<integer>
	or	
Region	\=	<integer>

NUMBER (or REGION) specifies the region number to which these parameters apply. Only one set of semiconductor parameters is allowed. Therefore, if the region specified is a semiconductor region, all other semiconductor regions (if there are any) will be changed as well.

<material definitions>

EG300	\=	<real>	: Energy gap at 300K (eq. 2.16)	(eV)
EGAlpha	\=	<real>	: Alpha (eq. 2.16)	
EGBeta	\=	<real>	: Beta (eq. 2.16)	
AFfinity	\=	<real>	: Electron affinity	(eV)
Permittivity	\=	<real>	: Dielectric permittivity	(F/cm)
Vsaturation	\=	<real>	: Saturation velocity (eq. 2.34, 2.35)	(cm/s)
MUN	\=	<real>	: Low-field electron mobility	(cm^2/s)
MUP	\=	<real>	: Low-field hole mobility	(cm^2/s)
G.surface	\=	<real>	: surface mobility reduction (eq. 2.33)	
TAUP0	\=	<real>	: SRH Electron lifetime (eq. 2.6, 2.8)	(s)
TAUN0	\=	<real>	: SRH Hole lifetime (eq. 2.6, 2.9)	(s)

...continued...

NSRHN \= <real> : SRH conc. parameter - electrons (eq. 2.8) (cm^{-3})
NSRHP \= <real> : SRH conc. parameter - holes (eq. 2.9) (cm^{-3})
ETrap \= <real> : Trap level = $E_t - E_i$ (eq. 2.6)
AUGN \= <real> : Auger coefficient (c_n) (eq. 2.7) (cm^6/s)
AUGP \= <real> : Auger coefficient (c_p) (eq. 2.7) (cm^6/s)
NC300 \= <real> : Conduction band density at 300K (eq. 2.17) (cm^{-3})
NV300 \= <real> : Valence band density at 300K (eq. 2.18) (cm^{-3})
ARICHN \= <real> : Richardson constant for electrons (eq. 2.45)
ARICHP \= <real> : Richardson constant for holes (eq. 2.46)
GCb \= <real> : Conduction-band degeneracy factor (eq. 2.33a,IIB-sm)
GVb \= <real> : Valence-band degeneracy factor (eq. 2.33b,IIB-sm)
EDb \= <real> : Donor energy level (eq. 2.34a,IIB-sm)
EAb \= <real> : Acceptor energy level (eq. 2.34b,IIB-sm)

Defaults:

Semiconductors

Constant	Silicon	Gallium Arsenide	Arbitrary
Energy gap (300K)	1.08	1.43	0.0
Alpha	4.73×10^{-4}	5.405×10^{-4}	0.0
Beta	636.	204	0.0
Electron affinity	4.17	4.07	0.0
Permittivity	11.8	10.9	0.0
Saturation velocity	(eq. 2.36)	(eq. 2.37)	0.0
Electron mobility	1000	5000	0.0
Hole mobility	500	400	0.0
Surface mobility reduction	1.0	1.0	0.0
SRH Electron lifetime	1.0×10^{-7}	1.0×10^{-7}	0.0
SRH Hole lifetime	1.0×10^{-7}	1.0×10^{-7}	0.0
Auger coefficient (n)	2.8×10^{-31}	2.8×10^{-31}	0.0
Auger coefficient (p)	9.9×10^{-32}	9.9×10^{-32}	0.0
Cond band density (300K)	2.8×10^{19}	4.7×10^{17}	0.0
Val band density (300K)	1.04×10^{19}	7.0×10^{18}	0.0
Eff Richardson const (n)	110	6.2857	0.0
Eff Richardson const (p)	30	105	0.0
SRH conc. parameter (n)	5.0×10^{16}	5.0×10^{16}	0.0
SRH conc. parameter (p)	5.0×10^{16}	5.0×10^{16}	0.0
Trap level	0.0	0.0	0.0
Cond band degen factor	2.0	2.0	0.0
Val band degen factor	4.0	2.0	0.0
Donor energy level (eV)	0.044	0.005	0.0
Acceptor energy level (eV)	0.045	0.005	0.0

Insulators

Constant	Silicon dioxide	Silicon nitride	Sapphire	Arbitrary
Permittivity	3.9	7.5	12.0	0.0

4. Examples

The following defines SRH lifetimes and concentration-independent low-field mobilities for all the semiconductor regions within the device (all the other parameters are assumed to be their default, consistent with the semiconductor type chosen):

```
MATERIAL    TAUN0=5.0e-6  TAUP0=5.0e-6  MUN=3000  MUP=500
```

The MESH card

1. Syntax

```
MESh <type> <cylindrical> <output files> <smoothing key>
```

2. Description

The mesh card either initiates the mesh generation phase or reads a previously generated mesh.

3. Parameters

<type>: One of

<Previous>

Previous	\=	<logical>	
Infile	\=	<filename>	
ASCII.In	\=	<logical>	(default is false)

Reads a previously generated mesh from a save file. ASCII.IN is a flag to indicate if the save file is ascii as opposed to binary.

<Rectangular>

Rectangular	\=	<logical>	
NX	\=	<integer>	
NY	\=	<integer>	
Diag.fli	\=	<logical>	(default is false)

These parameters initiate the generation of a rectangular mesh. NX is the number of nodes in the x-direction, NY the number in the y-direction. DIAG.FLIP which if set, flips the diagonals in a square mesh about the center of the grid. If DIAG.FLIP is false, all the diagonals will be in the same direction.

<Geometry>

```
Geometry   \=   <logical>
Infile     \=   <filename>
Flip.y     \=   <logical>    (default is false)
SCale      \=   <integer>    (default is 1)
```

This reads a mesh (ascii) generated by an external grid editor. FLIP.Y is a flag which will reverse the sign of the y-coordinate. SCALE is a factor by which all the coordinates read are multiplied by.

<Output files>

<PISCES-II format>

```
OUTFile     \=   <filename>
ASCII.Out   \=   <logical>    (default is false)
```

OUTFILE is the PISCES-II format output file to be read by a later run. If ASCII.OUT is set, OUTFILE will be written in ascii, otherwise it will be binary.

<Grid editor>

```
OUT.asc   \=   <filename>
Flip.y    \=   <logical>    (default is false)
SCale     \=   <integer>    (default is 1)
```

OUT.ASC is an ascii output file intended to be read by an external grid editor. See appendix B of [1] for details of the format. FLIP.Y and SCALE are as above.

<smoothing key>

```
SMooth.key   \=   <integer>
```

This causes mesh smoothing as described in section 4.6 of [1]. The digits of the integer are read in reverse order and decoded as follows:

1 Triangle smoothing, maintaining all region boundaries fixed.
2 Triangle smoothing, maintaining only material boundaries.
3 Node averaging.

Options 1 and 3 are the most common; 2 is used only if a device has several regions of the same material and the border between the different regions is unimportant.

<Cylindrical coordinates>

CYLindrical \= <logical>

Using the CYL parameter specifies that the mesh, whether generated in the current input deck or read from a file, is to be rotated about the y-axis to permit the simulation of cylindrically symmetrical devices. This information is NOT written to the mesh file; it must be specified in the input deck which calculates a solution.

4. Examples

Initiate a rectangular mesh and request it to be stored in mesh1.pg :

```
MESH   RECTANGULAR  NX=40  NY=17  OUTF=mesh1.pg
```

Read a previously generated mesh and generate an ascii file for a grid editor (the y axis is inverted because the grid editor obeys the convention that positive y is upward, while PISCES-II follows the semiconductor convention of positive y being into the bulk) :

```
MESH   INF=mesh1.pg  OUT.ASC=mesh1.pa  FLIP  ASCII.OUT
```

Read a geometry file, smooth the mesh, and store the file for a later run (ascii format):

```
MESH   GEOM  INF=geom1  SMOOTH.K=13131  OUTF=mesh1.pg
```

The smoothing does several averaging and flipping steps. The digits are read in reverse order, so that the flipping comes first, followed by node averaging, and so on.

The METHOD card

1. Syntax

METhod <general parameters> <method-dependent parameters>

2. Description

The METHOD card sets parameters associated with the particular solution algorithm chosen on the SYMBOLIC card. There can be more than one METHOD card in a single simulation, so that parameters can be altered. The default values of the parameters are used on the first occurrence of the METHOD card; subsequent METHOD cards only alter those coefficients specified.

3. Parameters

<general parameters>

Parameter			*Default*
ITlimit	\=	<integer>	15
Xnorm	\=	<logical>	true
Rhsnorm	\=	<logical>	false
P.toler	\=	<real>	Depends on choice of norm
C.toler	\=	<real>	Depends on choice of norm
LImit	\=	<logical>	false
PRint	\=	<logical>	false
Fix.qf	\=	<logical>	false
TRap	\=	<logical>	false
ATrap	\=	<real>	0.5

The above parameters are used to determine the convergence of the solution methods. ITLIMIT is the maximum number of allowed outer loops (i.e., Newton loops or Gummel continuity iterations). P.TOL and C.TOL are the termination criteria for the Poisson and continuity equations, respectively. If XNORM is chosen as the error norm, the Poisson updates are measured in units of kT/q, and carrier updates are measured relative to the local carrier concentration. In this case the default value for both P.TOL and C.TOL is 1×10^{-5}. If RHSNORM is selected, the Poisson error is measured in C/μm and the continuity error in A/μm. P.TOL then defaults to $1 \times 10^{-26} C/\mu m$, and C.TOL to $5 \times 10^{-18} A/\mu m$.

LIMIT indicates that the convergence criterion should be ignored, and iterations are to proceed until ITLIMIT is reached. PRINT prints the terminal fluxes/currents after each continuity iteration; if this parameter is not set, the terminal fluxes/currents are only printed after the solution converges. FIX.QF fixes the quasi-Fermi potential of each non-solved for carrier to a single value, instead of picking a value based on local bias (see section 5 of chapter 2 [1] and the ‘‘p.bias’’ and ‘‘n.bias’’ parameters on the SOLVE card). TRAP specifies that if a solution process starts to diverge, the electrode bias steps taken from the initial guess are reduced by the multiplicative factor ATRAP.

<method-dependent parameters>

For the Gummel method:

The following parameters are for damping the Poisson updates.

DVlimit	\=	<real>	(default is 0.1)
	or		
DAMPEd	\=	<logical>	(default is false)
DElta	\=	<real>	(default is 0.5)
DAMPLoop	\=	<integer>	(default is 10)
DFactor	\=	<real>	(default is 10.0)

The DVLIMIT parameter limits the maximum ψ update for a single loop. The DAMPED parameter indicates the use of a more sophisticated damping scheme proposed by Bank and Rose (this is the recommended option, particularly for large bias steps). The remaining damping parameters are only interpreted if the DAMPED parameter is specified. DELTA is the threshold for determining the damping factor for $\Delta\psi$ and must be between 0 and 1. DAMPLOOP is the maximum number of damping loops allowed to find a suitable damping coefficient. DFACTOR is a factor which serves to increase the initial damping coefficient for the next Newton loop.

The following parameters select acceleration methods for the Gummel iteration.

SInglepoisson	\=	<logical>	(default is false)
ICcg	\=	<logical>	(default is false)
LU1cri	\=	<real>	(default is $3\times10^{-3)}$
LU2cri	\=	<real>	(default is $3\times10^{-2)}$
Maxinner	\=	<integer>	(default is 25)
ACCEleration	\=	<logical>	(default is false)
ACCSTArt	\=	<real>	(default is 0.3)
ACCSTOp	\=	<real>	(default is 0.6)
ACCSTEp	\=	<real>	(default is 0.04)

The first two parameters specify how the Poisson equation iterations are to be performed. The SINGLEPOISSON option indicates that only a single Poisson iteration is to be performed per Gummel loop as opposed to the default where the continuity equation is only treated after Poisson has fully converged. The ICCG parameter chooses whether or not to use iteration to solve the multi-Poisson loops. It should be set whenever doing multi-Poisson. The next two parameters specify how much work is done per Poisson loop (cf. Section 3.3 [1]). The inner norm is required to decrease by at least LU1CRI before returning, or to reach a factor of LU2CRI below the projected Newton error, whichever is the smaller. (If the inner norm is allowed to exceed the projected Newton error, quadratic convergence is lost). MAXINNER sets the maximum number of ICCG iterations. The remaining parameters deal with an acceleration method for attaining faster overall convergence in the single-Poisson mode. The ACCELERATION option specifies that acceleration is to be used. ACCSTART is the starting value of the acceleration parameter, ACCSTOP is the final (limiting) value of the acceleration parameter and ACCSTEP is the step to be added to the value of the acceleration parameter after each iteration [1].

For the direct Newton method:

AUtonr	\=	<logical>	(default is false)
NRcriterion	\=	<real>	(default is 0.1)
2ndorder	\=	<logical>	(default is true)
TAuto	\=	<logical>	
TOl.time	\=	<real>	(default is 5×10^{-3})
L2norm	\=	<logical>	(default is true)
Dtmin	\=	<real>	(default is 1×10^{-25})
Extrapolate	\=	<logical>	(default is false)

The first two of the above parameters are for implementing an automated Newton-Richardson procedure which attempts to reduce the number of LU decompositions per bias point. The AUTONR option indicates that this algorithm is to be used. NRCRITERION is the ratio by which the norm from the previous Newton loop must go down in order to be able to use the same Jacobian (i.e., LU decomposition) for the current Newton loop. This is strongly recommended for full Newton iteration.

The remaining parameters are for transient simulations. 2NDORDER specifies that the second-order discretization of Bank, et. al (see chapter 2 of this report) be used as opposed to the first-order backward difference of PISCES-IIA. TAUTO forces PISCES-IIB to select time-steps automatically from the local truncation error estimates. Note that automatic time-stepping is the default for the second-order discretization but is not allowed for the first-order scheme. TOL.TIME is the maximum allowed local truncation error. L2NORM specifies that the error norms be L2 as opposed to infinity norms for calculating the time-steps. DT.MIN is the minimum time-step allowed in seconds, and EXTRAPOLATE uses a second-order extrapolation to compute initial guesses for successive time-steps.

4. Examples

The following specifies that for a simulation using the Gummel method (as previously specified by an appropriate symbolic card), that damping is to be employed and the Poisson error tolerance should be 1×10^{-30} coul/μm. Note that because XNORM defaults to true, XNORM must be turned off to use the rhs norm as a convergence criterion. If XNORM=FALSE had not been specified, the rhs norm and the update norm would have both been printed, but only the update norm would have been used to determine convergence.

```
METHOD   DAMPED  P.TOL=1.e-30  RHSNORM  XNORM=FALSE
```

The next example illustrates the trap feature, which can be quite useful for capturing knees of IV curves for devices such as SCRs. The first SOLVE card solves for the initial, zero bias case. On the second SOLVE card, we attempt to solve for V2=3 volts V3=5 volts. If such a large bias change caused the solution algorithms to diverge for this bias point, the bias steps would be multiplied by ATRAP(0.5); i.e., an intermediate point (V2=1.5 volts, V3=2.5 volts) would be attempted before trying to obtain V2=3 volts and V3=5 volts again. If the intermediate point can not be solved for either, PISCES-II will continue to reduce the bias step (the next would be V2=0.75 volts and V3=1.25 volts) up to 4 times. Note also that the intermediate solutions will be saved in output files in a manner similar to voltage stepping using the VSTEP parameter on the SOLVE card; i.e., if two intermediate steps to V2/V3=3/5 volts were required, they would be stored in ‘‘outa’’ and ‘‘outb’’ while V2/V3=3/5 volts would be stored in ‘‘outc’’.

```
METHOD   TRAP  ATRAP=0.5
SOLVE    INIT
SOLVE    V2=3  V3=5  OUTFILE=outa
```

Finally, an example of transient simulation. By default, the second-order discretization is used, but the required LTE, 1×10^{-3}, is smaller than the default. Newton-Richardson is also used. Note that because the Jacobian is exact for the second part (BDF-2) of the composite time-step, there should be very few factorizations for the BDF-2 interval when AUTONR is specified (see chapter 2 of this report).

```
METHOD   TOL.TIME=1E-3  AUTONR
```

The MODELS card

1. Syntax

MOdels <model flags> <numerical parameters>

2. Description

The model card sets the temperature for the simulation and specifies model flags to indicate the inclusion of various physical mechanisms and models.

3. Parameters

<model flags>

Srh	\=	<logical>	(default is false)
CONSrh	\=	<logical>	(default is false)
AUger	\=	<logical>	(default is false)
BGn	\=	<logical>	(default is false)
CONMob	\=	<logical>	(default is false)
ANalytic	\=	<logical>	(default is false)
ARORA	\=	<logical>	(default is false)
FLdmob	\=	<logical>	(default is false)
SURFmob	\=	<logical>	(default is false)
TFLDmob	\=	<logical>	(default is false)
OLDtfld	\=	<logical>	(default is false)
CCSmob	\=	<logical>	(default is false)
USER1	\=	<logical>	(default is false)
IMPAct	\=	<logical>	(default is false)
BOltzmann	\=	<logical>	(default is true)
FErmidirac	\=	<logical>	(default is false)
Incomplete	\=	<logical>	(default is false)
Photogen	\=	<logical>	(default is false)
Print	\=	<logical>	(default is false)

CONSRH/SRH and AUGER specify Shockley-Read-Hall (eq. 2.6 [1]) and Auger recombination (eq. 2.7 [1]) respectively. SRH uses fixed lifetimes and CONSRH uses concentration-dependent lifetimes. (eq. 2.8, 2.9 [1]). BGN is band-gap narrowing (eq. 2.20 [1]). CONMOB is concentration dependent mobility from tables for 300K (currently only silicon and gallium arsenide have been implemented). ANALYTIC is an analytical concentration dependent mobility model for silicon only (see chapter 1, section 3 of [1]) which will include temperature dependence.
ARORA specifies an alternative concentration dependent mobility model for silicon [8]. FLDMOB specifies a lateral field-dependent model (eq. 2.34, 2.35 [1]). SURFMOB invokes the effective field based surface mobility model. ([2] [3]) The current implementation accounts only for phonon scattering at room temperature,

restricting it to effective fields of less than 0.5 MV/cm. CCSMOB invokes the mobility model of Dorkel and Leturcq[9], which includes carrier-carrier scattering effects, temperature dependence and concentration dependence. TFLDMOB invokes the transverse-field mobility model whose derivation is described in [10]. OLDTFLD specifies the use of the Schwarz-Russek formulation [12], while TFLDMOB invokes an extended version of the Schwarz-Russek formulation for both holes and electrons [10][11]. USER1 specifies the user-customizable concentration-dependent mobilty model. Only one of ANALYTIC, ARORA, CCSMOB or USER1 may be specified. IMPACT invokes the empirical impact ionization model. ([4] [5]) A more rigorous impact ionization model can be specified with IMPACT command. BOLTZMANN and FERMIDIRAC indicate the carrier statistics to be used (eq. 2.13, 2.14 and 2.10, 2.11 [1]), while INCOMPLETE indicates that incomplete-ionization of impurities should be accounted for (eq. 2.31, 2.32 [1]). PHOTOGEN specifies that photogeneration is to be used; FLUX and ABS.COEF must also be specified to use this model. PRINT prints the status of all models and a variety of coefficients and constants.

<Numerical parameters>

Temperature	\=	<real>	(default is 300K)
B.Electrons	\=	<real>	(default is 2)
B.Holes	\=	<real>	(default is 1)
E0	\=	<real>	(default is 4×10^3V/cm)
Flux	\=	<real>	(default is 0.0 cm^{-2})
Abs.coef	\=	<real>	(default is 0.0 cm^{-1})
Acc.sf	\=	<real>	(default is 0.87)
Inv.sf	\=	<real>	(default is 0.75)
Ox.left	\=	<real>	
Ox.right	\=	<real>	
Ox.bottom	\=	<real>	

TEMPERATURE should be specified in Kelvin units. B.ELECTRONS and B.HOLES are parameters used in the field-dependent mobility expression for silicon (eq. 2.34 [1]), while E0 is a parameter used in the field-dependent mobility model for gallium arsenide (eq. 2.35 [1]). FLUX is the incident photon flux at the y=0 surface in $photons/cm^2$, and ABS.COEF is the optical absorption coefficient in cm^{-1}. ACC.SF is the low-field surface reduction factor for accumulation layers, used in conjunction with the transverse-field mobility model TFLDMOB. INV.SF is the inversion layer low-field surface reduction factor for the transverse-field mobility model. OX.LEFT, OX.RIGHT, and OX.BOTTOM define the location of the gate region for the transverse-field mobility model.

4. Examples

The following example selects concentration dependent mobility and SRH recombination. Fermi-dirac statistics are used, and the simulation is specified to be performed at 290K.

```
MODELS   CONMOB  SRH  FERMI  TEMP=290
```

The OPTIONS card

1. Syntax

Options <run control> <plot control>

2. Description

The OPTIONS card sets options for an entire run.

3. Parameters

<run control>

G.debug or Debug	\=	<logical>	(default is false)
N.debug	\=	<logical>	(default is false)
CPUStat	\=	<logical>	(default is false)
CPUFile	\=	<character>	(default is pisces.cpu)

G.DEBUG (or DEBUG) and N.DEBUG print debugging information to the PISCES-II standard output. G.DEBUG (or DEBUG) prints general information, N.DEBUG more specifically numerical parameters. CPUSTAT is a flag to indicate that a cpu profile of the solution process is to be printed to the file specified by CPUFILE.

<Plot control>

PLOTDevice	\=	<character>	(default login terminal)
PLOTFile	\=	<character>	(default device specific)
X.Screen	\=	<real>	(default devices size)
Y.Screen	\=	<real>	(default devices size)
X.Offset	\=	<real>	(default 0 inches)
Y.Offset	\=	<real>	(default 0 inches)

The first parameter specifies the output plot device. If no device is given, a default (usually the user's graphics terminal) will be used. Most versions of PISCES-II use the PLOTCAP graphics package from Stanford. Please refer to the PLO TCAP documents for further details. The full set of supported de vices is contained in the PLOTCAP data base. Possibilities include:

hp2648	hp2623	tek4107	vt240
laserwriter	ditroff	latex	xwindows
hp9873	printronix	sunview	save

Plots will be scaled to the size of the specific device. Also note that on color graphics terminals, the different line types are implemented as different colors; on the black and white monitors they are implemented as dot and line patterns. X.SCREEN is the physical width of the screen and Y.SCREEN is the height. They are set automatically depending on the plot device, but can be altered for special effects (split screen plots, for instance). The offset from the bottom left corner of the screen may be set using X.OFFSET and Y.OFFSET.

The output file is generally defined by the plot device. For example, a graphics terminal will use the terminal as the output *file*. Printers may have the output file be a spooler. The graphics output file can be explicitly set by the the PLOTFILE command. All graphics output will then be routed to the given file. Note that the contents of the file will be in a format specific to the given device.

4. Examples

The following sets up a plot for a Tektronix terminal, using a small centered window. Cpu information is also logged to the default file.

```
OPTIONS    PLOTDEV=tek4107  X.S=6  Y.S=5  X.Off=1  Y.OFF=0.5
+              CPUSTAT
```

Here we set the plot device to the LaserWriter and save the output in a file called **plot.ps**.

```
OPTIONS    PLOTDEV=lw   PLOTFILE=plot.ps
```

The PLOT.1D card

1. Syntax

PLOT.1d <line segment definition> <plotted quantity> <control>

2. Description

The PLOT.1D card plots a specific quantity along a line segment through the device (mode A), or plots an I-V curve of data (mode B).

3. Parameters

<line segment definition>

X.Start or A.X	\=	<real>
Y.Start or A.Y	\=	<real>
X.End or B.X	\=	<real>
Y.End or B.Y	\=	<real>

The above parameters define the Cartesian coordinates of the start (A.X,A.Y) and the end (B.X,B.Y) of the line segment along which the specified quantity is to be plotted. The data is plotted as a function of distance from the start (A). The line segment may not be defaulted. It is required in mode A.

<plotted quantity>

One of:

POTential	\=	<logical>	Mid-gap potential
QFN	\=	<logical>	Electron quasi-fermi level
QFP	\=	<logical>	Hole quasi-fermi level
Doping	\=	<logical>	Doping
ELectrons	\=	<logical>	Electron concentration
Holes	\=	<logical>	Hole concentration
NET.CHarge	\=	<logical>	Net charge concentration
NET.CArrier	\=	<logical>	Net carrier concentration
J.Conduc	\=	<logical>	Conduction current
J.Electr	\=	<logical>	Electron current
J.Hole	\=	<logical>	Hole current
J.Displa	\=	<logical>	Displacement current
J.Total	\=	<logical>	Total current
E.Field	\=	<logical>	Electric field
Recomb	\=	<logical>	Net recombination

BAND.Val	\=	<logical>	Valence band potential
BAND.Con	\=	<logical>	Conduction band potential

or :

X.Axis	\=	<character>
Y.Axis	\=	<character>
INFile	\=	<character>

The above parameters specify the quantity to be plotted. There is no default. In mode A, one of the solution variables is plotted versus distance into the device. For vector quantities, the magnitude is plotted. In mode B, terminal characteristics can be plotted against each other by choosing the value to be plotted on each axis (XAXIS=,YAXIS=). Quantities available for plotting include applied biases (XAXIS/YAXIS=VA1, VA2, ..., VA9, VA0), actual contact bias which may differ from applied bias in the case of lumped element boundary conditions (V1, V2, etc.), terminal current (I1, I2, etc.), AC capacitances (C11, C12, C21, etc.), AC conductance (G11, G12, G21, etc.) and AC admittance (Y11, Y12, Y21, etc.). Additionally, any of the voltages or currents can be plotted versus time for transient simulations, and any AC quantity can be plotted versus frequency. The values plotted are the I-V or AC data of the present run, provided a log is being kept (see the LOG card). Alternatively, a different log file can be loaded with INFILE.

<control>

LOgarithm or Y.Log	\=	<logical>	(default is false)
X.Log	\=	<logical>	(default is false)
ABsolute	\=	<logical>	(default is false)
NO.Clear	\=	<logical>	(default is false)
NO.Axis	\=	<logical>	(default is false)
Unchanged	\=	<logical>	(default is false)
INTegral	\=	<logical>	(default is false)
NEGative	\=	<logical>	(default is false)
NO.Order	\=	<logical>	(default is false)
POInts	\=	<logical>	(default is false)
PAuse	\=	<logical>	(default is false)
LIne.type	\=	<integer>	(default is 1)
MIn.value	\=	<real>	
MAx.value	\=	<real>	
X.Max	\=	<real>	
X.Component	\=	<logical>	(default is false)
Y.Component	\=	<logical>	(default is false)
Spline	\=	<logical>	(default is false)
NSpline	\=	<logical>	(default is 100)
OUTFile	\=	<character>	(default from OPTION card)
AScii	\=	<logical>	(default is false)

ABSOLUTE specifies that the absolute value of the variable be taken. For rapidly varying quantities, the LOGARITHM (Y.LOG, X.LOG) is often more revealing. Since many of the quantities may become negative, PISCES-II actually uses

$\log(x) = \mathrm{sign}(x) \times \log_{10}(1 + |x|)$ to avoid overflow. To get the true logarithm of a quantity, specify ABSOLUTE and LOGARITHM - the absolute is taken first and there is no danger of negative arguments. NO.CLEAR indicates that the screen is not to be cleared before the current plot so that several curves can be plotted on the same axis. NO.AXIS indicates that the axes for the graph are not to be plotted. UNCHANGED is a synonym for NO.AXIS and NO.CLEAR, but additionally it forces the use of the previous axis bounds so that a number of curves can easily be put on the same axis. INTEGRAL plots the integral of the specified ordinate. NEGATIVE negates the ordinate values. PISCES-II by default will order the plot coordinates by abscissa value; this ordering will result in unusual plots for IV curves with negative resistance, for example. The NO.ORDER parameter forces PISCES-II to plot the data points as they naturally occur. POINTS marks the data points on the plotted curve. The PAUSE option causes PISCES-II to stop at the end of the plot so that a hardcopy may be made before continuing. Execution can be resumed by hitting a carriage return. LINE.TYPE specifies the line type for the plotted curve. MIN.VALUE and MAX.VALUE specify minimum and maximum values for the ordinate of the graph; their defaults are found automatically from the data to be plotted. X.MAX allows a maximum value for the abscissa to be specified (default is just the maximum abscissa value in the data to be plotted). X.COMPONENT and Y.COMPONENT force the x and y components respectively of any vector quantities to be plotted as opposed to the default total magnitude. The SPLINE option indicates that spline-smoothing should be performed on the data using NSPLINE interpolated points (maximum is 500). The default plot device is generally the user’s terminal but may be reset with the OPTION card. If OUTFILE is specified, the graphics output will be directed to that file. For further discussion, see the OPTION card.

4. Examples

The following plots a graph of potential along a straight line from (0.0,0.0) to (5.0,0.0):

```
PLOT.1D   POTEN  A.X=0  A.Y=0  B.X=5  B.Y=0
```

In the next example, the log of the electron concentration is plotted from (1.0,-0.5) to (1.0,8.0) with bounds on the plotted electron concentration of 1.0e10 and 1.0e20. A spline interpolation is performed with 300 interpolated points. The non-spline-interpolated points are marked.

```
PLOT.1D   ELECT  LOG  A.X=1  A.Y=-.5  B.X=1  B.Y=8
+         MIN=10  MAX=20  SPLINE  NSPL=300  POINTS
```

In the following example, the current in contact 1 is plotted as a function of contact 2 voltage, then the curve is compared with a previous run.

```
PLOT.1D   X.AXIS=V2  Y.AXIS=I1
PLOT.1D   X.AXIS=V2  Y.AXIS=I1  INF=logf0  UNCH
```

The following plots the actual contact voltage on a contact versus the applied voltage.

```
PLOT.1D   X.AXIS=V3  Y.AXIS=VA3   OUTFILE=save.plot
```

Finally, the following shows a plot of two capacitance components versus the log of frequency. A different line type is chosen for the second component.

```
PLOT.1D   X.AXIS=FREQ  Y.AXIS=C21  X.LOG
PLOT.1D   X.AXIS=FREQ  Y.AXIS=C31  X.LOG  UNCH  LINE=4
```

The PLOT.2D card

1. Syntax

PLOT.2d <area definition> <plotted quantity> <control>

2. Description

The PLOT.2D card plots quantities in a specified two-dimensional area of the device. A PLOT.2D card is required before performing a contour plot (see CONTOUR card) in order to obtain the plot boundaries.

3. Parameters

<area definition>

X.MIn	\=	<real>
X.MAx	\=	<real>
Y.MIn	\=	<real>
Y.MAx	\=	<real>

The above parameters define the rectangular area of the device to be plotted. The default area is a rectangle around the entire device.

<plotted quantity>

Grid or Mesh	\=	<logical>	(default is false)
Crosses	\=	<logical>	(default is false)
Boundary	\=	<logical>	(default is false)
Depl.edg	\=	<logical>	(default is false)
Junction	\=	<logical>	(default is false)

The GRID option plots the grid, including lines delineating elements. CROSSES plots crosses at the locations of grid points. BOUNDARY indicates that the boundaries around the device and between regions are to be plotted. DEPL.EDG indicates that depletion edges are to be plotted (note: depletion edges can only be plotted after a solution is present). The JUNCTION option specifies that the junctions from the doping profiles are to be plotted.

<control>

NO.TIc	\=	<logical>	(default is false)
NO.TOp	\=	<logical>	(default is false)
NO.Fill	\=	<logical>	(default is false)
NO.Clear	\=	<logical>	(default is false)
LAbels	\=	<logical>	(default is false)
Flip.x	\=	<logical>	(default is false)
Pause	\=	<logical>	(default is false)
L.Elect	\=	<integer>	
L.Deple	\=	<integer>	
L.Junct	\=	<integer>	
L.Bound	\=	<integer>	
L.Grid	\=	<integer>	
Outfile	\=	<character>	(default from OPTION card)

NO.TIC indicates that tic marks are not to be included around the plotted area. NO.TOP indicates that tic marks are not to be put on the top of the plotted region. The NO.FILL option will force PISCES-II to draw the device area plotted to scale; if this option is not specified, the plot will fill the screen, and the triangles will appear distorted. NO.CLEAR specifies that the screen is not to be cleared before plotting. LABELS makes room for color contour labels on the right side of the plot device. FLIP.X flips the plot about the y-axis; i.e., it negates all x coordinates so that the plot is mirrored. The PAUSE option causes PISCES-II to stop at the end of the plot so that a hardcopy may be made before continuing. Execution can be resumed by hitting a carriage return. L.ELECT, L.DEPLE, L.JUNCT, L.BOUND and L.GRID set line types for electrodes, depletion edges, junctions, region boundaries and grid, respectively. The default plot device is generally the user's terminal but may be reset with the OPTION card. If OUTFILE is specified, the graphics output will be directed to that file. For further discussion, see the OPTION card.

4. Examples

The following plots the entire grid to scale with tic marks:

```
PLOT.2D   GRID  NO.FILL
```

In the next example, the device and region boundaries, junctions and depletion edges are plotted in the rectangular area bounded by $0 < x < 5\mu$ and $0 < y < 10\mu$. The plot is allowed to fill the screen and tic marks are not included along the top of the plot.

```
PLOT.2D   X.MIN=0  X.MAX=5  Y.MIN=0  Y.MAX=10
+         JUNCT  BOUND  DEPL  NO.TOP
```

The PRINT card

1. Syntax

```
PRint <location> <quantity> <flags>
```

2. Description

The PRINT card prints specific quantities at points within a defined area of the device.

3. Parameters

<location>

```
X.MIn    \=   <real>
X.MAx    \=   <real>
Y.MIn    \=   <real>
Y.MAx    \=   <real>
```

or

```
IX.Low    \=   <integer>
IX.High   \=   <integer>
IY.Low    \=   <integer>
IY.High   \=   <integer>
```

The above parameters define area in which the points of interest lie. X.MIN, X.MAX, Y.MIN and Y.MAX specify an area in physical coordinates (in μm). IX.LOW, IX.HIGH, IY.LOW and IY.HIGH specify the area by the bounding indices (valid only for rectangular meshes). The default area is the entire device.

<quantity>

POints	\=	<logical>
Elements	\=	<logical>
Geometry	\=	<logical>
Solution	\=	<logical>
P.SOL1	\=	<logical>
P.SOL2	\=	<logical>
Current	\=	<logical>
P.CURR1	\=	<logical>
P.CURR2	\=	<logical>
Que	\=	<logical>
P.QUE1	\=	<logical>
P.QUE2	\=	<logical>
Material	\=	<logical>

The above parameters specify the quantities to be plotted. Any or all may be specified, and each defaults to false. POINTS prints node information (coordinates, doping, etc.). ELEMENTS prints information on the triangular elements (number, nodes, material). GEOMETRY prints geometrical information on the triangles. SOLUTION prints the *present* solution (ψ, n, p and quasi-fermi potentials), while P.SOL1 and P.SOL2 print the *previous* two solutions. CURRENT prints currents (electron, hole, conduction, displacement and total) at each node for the present solution; P.CURR1 and P.CURR2 print currents for previous solutions. QUE prints space charge, recombination and electric field for the present solution; P.QUE1 and P.QUE print the same quantities for the previous two solutions. MATERIAL prints material information (permittivity, band-gap, etc.), including the value of the concentration dependent mobility and lifetime (if specified) at each point.

<flags>

X.Component	\=	<logical>	(default is false)
Y.Component	\=	<logical>	(default is false)

X.COMPONENT and Y.COMPONENT specify how any of the various vector quantities (currents, fields) should be printed. The default is the magnitude of the vector as a whole. X.COMPONENT specifies that the magnitude of the x-component of all vectors be printed, while Y.COMPONENT specifies the y-component. Only one of these (or neither) can be specified on a single card.

4. Examples

The following prints the physical coordinates, doping and region/electrode information for points along the 10th x grid line, from the 1st to the 20th y grid lines.

```
PRINT   POINTS  IX.LO=10  IX.HI=10  IY.LO=1  IY.HI=20
```

In the next example, solution information is printed for $0 < x < 1\mu$m and $0 < y < 2\mu$m.

```
PRINT   SOLUTION  X.MIN=0  X.MAX=1  Y.MIN=0  Y.MAX=2
```

The REGION card

1. Syntax

REGIon <number> <position> <material>

2. Description

The region card defines the location of materials in a rectangular mesh. Every triangle must be defined to be some material.

3. Parameters

<number>

NUmber	\=	<integer>

This parameter selects the region in question. There is a maximum of eight regions in a PISCES-II structure.

<position>

IX.Low	\=	<integer>
IX.High	\=	<integer>
IY.Low	\=	<integer>
IY.High	\=	<integer>

These parameters are the indices of a box in the rectangular mesh.

<material>

One of:

SILicon

Gaas

SEmiconductor

Oxide or SIO2

NItride or SI3n4

SApphire

INsulator

4. Examples

The following defines a silicon region extending from nodes 1 to 25 in the x direction and nodes 4 to 20 in the y direction :

```
          REGION  NUM=1  IX.LO=1  IX.HI=25  IY.LO=4  IY.HI=20  SIL
```

Note that region cards are cumulative in effect :

```
          REGION  NUM=1  IX.LO=4   IX.HI=5   IY.LO=1  IY.HI=20  OXIDE
          REGION  NUM=1  IX.LO=36  IX.HI=37  IY.LO=1  IY.HI=40  OXIDE
```

defines one region comprised of two separate strips.

The REGRID card

1. Syntax

REGRid <location> <variable> <control> <files>

2. Description

The REGRID card allows refinement of a crude mesh. Any triangle across which the chosen variable changes by more than a specified tolerance, or in which the chosen variable exceeds a given value, is refined.

3. Parameters

<location>

One of :

X.MIn	\=	<real>	
X.MAx	\=	<real>	
Y.MIn	\=	<real>	
Y.MAx	\=	<real>	
REGion	\=	<integer>	default:all
IGnore	\=	<integer>	default:none

The bounds X.MIN-Y.MAX are used to limit the refinement; only triangles which have nodes which fall inside the box are considered for refinement. The REGION parameter has a similar use; only regions specified are refined according to the user criterion. (Others may be refined as a side effect, to maintain well-shaped triangles). The default is to refine all regions for potential and electric field regrids, and all semiconductor regions for regrids which depend on the other variables. The parameter IGNORE is similar to REGION, but opposite in effect. Ignored regions are not regridded either according to the user criterion or according to the ‘‘obtuse criterion’’ (see below); nor are they smoothed after regrid. The default is not to ignore any region.

<variable>

One of :

Potential	\=	<logical>	Mid-gap potential	(Volts)
EL.field	\=	<logical>	Electric-field	(Volts/cm)
QFN	\=	<logical>	Electron quasi-fermi level	(Volts)
QFP	\=	<logical>	Hole quasi-fermi level	(Volts)
DOPIng	\=	<logical>	Net doping	(cm^{-3})
ELEctron	\=	<logical>	Electron concentration	(cm^{-3})

Hole	\=	<logical>	Hole concentration	(cm^{-3})
NET.CHrg	\=	<logical>	Net charge	(cm^{-3})
NET.CArr	\=	<logical>	Net carrier concentration	(cm^{-3})
MIn.carr	\=	<logical>	Minority carrier conc.	(cm^{-3})

This parameter selects the discriminatory variable.

<control>

STep or RAtio	\=	<real>	no default
CHange	\=	<logical>	default : true
ABsolute	\	<logical	default: false
LOGarithm	\=	<logical>	default: false
MAx.level	\=	<integer>	default: dynamic
SMooth.k	\=	<integer>	default: 0
LOCaldop	\=	<logical>	default: false
COs.ang	\=	<real>	default:2.0

CHANGE determines whether to use the magnitude or the difference of a variable in a triangle as the criterion of refinement. It is normally set to true (difference). STEP is the numerical criterion for refining a triangle. RATIO is a synonym. If the variable ranges across many orders of magnitude, it is advisable to examine its logarithmic variation, using the LOG flag. In this case STEP will be interpreted as the step in the logarithm. ABSOLUTE specifies that the absolute value of the quantity is to be used. Since many of the quantities may become negative, PISCES-II actually uses

$\log(x) = \text{sign}(x) \times \log_{10}(1 + |x|)$ to avoid overflow. To get the true logarithm of a quantity, specify ABSOLUTE and LOGARITHM - the absolute is taken first and there is no danger of negative arguments. LOCALDOP is used with minority carrier regrids, and specifies that if the minority carrier concentration exceeds the local doping, the grid is to be refined. MAX.LEVEL is the maximum level of any triangle relative to the original mesh. It defaults to one more than the maximum level of the grid, but can be set to a smaller value to limit refinement. Values less than or equal to zero are interpreted relative to the current maximum level. SMOOTH.K has the same meaning as on the mesh card. COS.ANGLE defines the ‘‘obtuse criterion’’ to limit the creation of obtuse angles in the mesh. If regrid would create a triangle with an angle whose cos is less than -COS.ANGLE, nodes are added so that this does not occur. The test can be turned off locally by using the ignore parameter; it can be turned off everywhere by using a value of COS.ANG greater than 1. The default is to turn it off everywhere.

<files>

```
OUTFile      \=    <filename>
OUT.green    \=    <filename>
IN.green     \=    <filename>
DOPFile      \=    <filename>
AScii        \=    <logical>      (default is false)
```

OUTFILE is the binary output mesh file, and is necessary if the mesh is to be used for subsequent runs. A history of the triangle tree is always generated to assist further regriding steps. Its name can be specified by OUT.GREEN, and its default is generated from OUTFILE by concatenating the letters “tt” to the end. Additionally, a triangle tree for the previous mesh (if a tree exists) is used for this regrid. By default, PISCES-II will look for a file with the same name as the current mesh plus “tt” at the end as above. Alternatively, IN.GREEN can be used to implement a different file name. DOPFILE is the name of a file (up to 20 characters) which contains the doping for the device (see DOPING card). Specifying DOPFILE avoids interpolating doping values at any newly created grid points (the default), by using the initial doping specification to redope the structure. ASCII specifies that all mesh files and triangle trees (not DOPFILE) for this card should be done in ascii rather than the default - binary.

4. Examples

Starting with an initial grid, we refine twice, requesting that all triangles with large doping steps be refined:

```
REGRID   LOG DOPING STEP=6 OUTF=grid1 DOPF=dopxx1
REGRID   LOG DOPING STEP=6 OUTF=grid2 DOPF=dopxx1
```

A similar effect could be obtained with just one regrid statement:

```
REGRID   LOG DOPING STEP=6 OUTF=grid2 DOPF=dopxx1
+        MAX.LEVEL=2
```

In both cases two levels of refinement are done. The first choice is preferable however, because new doping information is introduced at each level of refinement. This gives a better criterion for refinement, and fewer triangles.
Now we perform an initial solution and refine triangles which exhibit large potential steps:

```
SOLVE   INIT OUT=grid2.si
REGRID POTENTIAL STEP=0.2 OUTF=grid3
```

The SOLVE card

1. Syntax

SOlve <estimate> <dc bias> <transient> <ac> <files>

2. Description

The SOLVE card instructs PISCES-II to perform a solution for one or more specified bias points.

3. Parameters

<estimate>

INitial	\=	<logical>
PREvious	\=	<logical>
PROject	\=	<logical>
LOcal	\=	<logical>

The above parameters are used to specify how the initial guess for the solution is to be obtained. The first bias point for a given structure must have the INITIAL parameter specified. From then PISCES-II will either use the previous solution (PREVIOUS), or if there are two previous solutions present and equivalent bias steps are taken on any electrodes that are changed (see chapter 3 in this report and the example below), an extrapolation (PROJECT) from the preceding two solutions will be used to get an improved initial guess. After the initial bias point, PISCES-II will automatically use extrapolation wherever possible if no estimate parameter is supplied. A different type of previous guess is available by using LOCAL values of the quasi-fermi levels.

<bias conditions>

V1	\=	<real>	
I1	\=	<real>	
V2	\=	<real>	
I2	\=	<real>	
	.		
	.		
	.		
V9	\=	<real>	
I9	\=	<real>	
V0	\=	<real>	
I0	\=	<real>	
VStep	\=	<real>	(default is 0.0)
IStep	\=	<real>	(default is 0.0)
NSteps	\=	<integer>	(default is 0)
Electrode	\=	<integer>	
N.bias	\=	<real>	
P.bias	\=	<real>	

The parameters V1, V2,...,V9, V0 represent the bias voltages (for non-current boundaries only) and I1, I2,...,I9, I0 represent the terminal currents in units of Amps/μm (for current boundary contacts only - see CONTACT card) applied at contacts 1, 2,...,9, 0. The defaults for these parameters are normally the potentials (currents) from the previous bias point. However, if the INITIAL flags is specified on the solve card, then all contact potentials that are not specified on the curent contact card will be reset to 0.0v. This reset holds even if a contact potential has been set on a previous solve card. VSTEP (ISTEP) is a voltage (current) increment to be added to one or more electrodes, as specified by the integer assigned to ELECTRODE. If more than one electrode is to be stepped, ELECTRODE should then be an n-digit integer, where each of the n-digits is a separate electrode number (and if there are 10 electrodes, don’t put electrode 0 first in the sequence!). NSTEPS is the number of bias increments (steps) to be taken; i.e., if VSTEP (ISTEP) is specified, the specified electrode is incremented NSTEPS times. Finally, N.BIAS and P.BIAS specify fixed quasi-Fermi potentials for carriers (electrons and holes, respectively) that are not being solved for. If N.BIAS or P.BIAS are not specified, then PISCES-II either will choose local quasi-Fermi potentials based on bias and doping (see chapter 2 of [1]) or if ‘‘fix.qf’’ is set on the METHOD card, will set the quasi-Fermi levels where applicable to values which produce the least amount of free carriers (maximum bias for electrons and minimum bias for holes).

<transient>

Dt or TSTEp	\=	<real>	(default is 0)
TSTOp or TFinal	\=	<real>	
Ramptime	\=	<real>	(default is 0)
ENdramp	\=	<real>	(default is 0)

DT (or TSTEP) is the time-step to be taken. For automatic time-step runs (see the METHOD card), DT (TSTEP) is used to select the first time step only. TSTOP (or TFINAL) specifies the end of the time interval to be simulated so that if the simulation begins at $t = t_0$, it will end at $t = \mathrm{TSTOP}$ (TFINAL). Alternatively, NSTEPS can be used to signal the end of the run; i.e., the final time would be $t = t_0 + \mathrm{NSTEPS} * \mathrm{TSTEP}$. RAMPTIME and ENDRAMP apply any bias changes as linear ramps. RAMPTIME specifies a ramp interval in seconds; i.e., the ramp will begin at $t = t_0$ and end at $t = t_0 + \mathrm{RAMPTIME}$. Alternatively, ENDRAMP specifies the exact end of the ramp in running time; i.e. the ramp will start at $t = t_0$ and end at $t = \mathrm{ENDRAMP}$.

<ac>

AC.analysis	\=	<logical>	(default is false)
FRequency	\=	<real>	
FStep	\=	<real>	(default is 0)
MUlt.freq	\=	<logical>	(default is false)
NFsteps	\=	<integer>	(default is 0)
VSs	\=	<real>	(default is 0.1*kT/q)
TErminal	\=	<integer>	(default is all)
S.omega	\=	<real>	(default is 1.0)
MAx.inner	\=	<integer>	(default is 25)
TOlerance	\=	<real>	(default is $1x10^{-5}$)

AC.ANALYSIS is a flag to indicate that AC sinusoidal small-signal analysis (see chapter 2 of this report) be performed after the DC condition is solved for. Note that the full Newton method (2 carriers) must be used for this analysis. FREQUENCY is the frequency (in Hz) at which this analysis is to be performed. The analysis can be repeated at a number of different frequencies (without resolving the DC condition) by selecting an FSTEP. FSTEP is a frequency increment which is added to the previous frequency by default, or it may be multiplied by setting MULT.FREQ. The number of increments is given by NFSTEPS. VSS is the magnitude of the applied small-signal bias ($\tilde{V}_i$ in equation (2.19)). TERMINAL is the contact to which this AC bias will be applied. More than one contact number may be specified (via concatenation), but each will be solved separately. Each contact that is specified yields a column of the admittance matrix as defined by equation (2.17). The analysis makes use SOR to solve a linear system. S.OMEGA is the relaxation parameter for SOR (it is NOT a frequency), MAX.INNER is the maximum number of SOR iterations and TOLERANCE is the SOR convergence criterion. Specification of the ‘‘g.debug’’ flag on the OPTIONS card will provide some detailed information on the AC solution procedure.

<files> (optional)

Outfile	\=	<filename>	
Currents	\=	<logical>	(default is false)
AScii	\=	<logical>	(default is false)

The OUTFILE parameter specifies the name of the binary output file for the solution of this bias point. The file names may contain up to 20 characters. If an electrode is stepped so that more than one solution is generated by this card, the last non-blank character of the supplied file name will have its ascii code incremented by one for each bias point in succession, resulting in a unique file per bias point. If CURRENTS is specified, the electron, hole, and displacement currents, and the electric field, will be computed and stored with the solution. IF ASCII is specified, OUTFILE will be ascii as opposed to binary.

4. Examples

The following performs an initial bias point, saving the solution to the data file OUT0:

```
SOLVE   INIT  OUTF=OUT0
```

In the next example, bias stepping is illustrated. The two solve cards produce the following bias conditions:

Bias point #	V1	V2	V3
1	0.0	0.5	-0.5
2	1.0	0.5	0.0
3	2.0	0.5	0.0
4	3.0	0.5	0.0
5	4.0	0.5	0.0
6	5.0	0.5	0.0

The solutions for these bias points will be saved to the files OUT1, OUTA, OUTB, OUTC, OUTD and OUTE. Note that the initial guess for the first bias point is obtained directly from the preceding solution because the PREVIOUS option was specified. The initial guesses for bias points 2 and 3 will also be obtained as if PREVIOUS had been specified since two electrodes (numbers 1 and 3) had their biases changed on bias point 2. However, for bias points 4, 5 and 6, PISCES-II will use a projection to obtain an initial guess since starting with bias point 4, both of its preceding solutions (bias points 2 and 3) only had the same electrode bias (number 1) altered.

```
SOLVE   PREV  V1=0  V2=.5  V3=-.5  OUTF=OUT1
SOLVE   V1=1  V2=.5  V3=0  VSTEP=1  NSTEPS=4
+         ELECT=1  OUTF=OUTA
```

Here is a case where two electrodes are stepped (2 and 3). The bias points solved for will be (0,0,1), (0,.5,1.5), (0,1,2) and (0,2,3). PISCES-II will use the PROJECT option to predict an initial guess for the third and fourth bias points since the bias voltages on both electrodes 2 and 3 have been altered by the same amount between each point.

```
SOLVE      V1=0  V2=0  V3=1  VSTEP=.5  NSTEPS=2
+          ELECT=23
SOLVE     V2=2  V3=3
```

If no new voltages are specified and a VSTEP is included, the first bias point solved for is the preceding one incremented appropriately by VSTEP. This is illustrated by repeating the above example as a three card sequence:

```
SOLVE   V1=0  V2=0  V3=1
SOLVE   VSTEP=.5  NSTEPS=2  ELECT=23
SOLVE   VSTEP=1   NSTEPS=1  ELECT=23
```

The following sequence is an example of a time-dependent solution. The METHOD card specifies the second-order discretization and automatic time-step selection option, along with Newton-Richardson. The first SOLVE card then computes the solution for a device with 1 volt on V1 and 0 on V2 in steady-state. The second SOLVE card specifies that V1 is to be ramped to 2 volts over a period of 10ns and is left on until 25 ns. Each solution is written to a file; the name of the file is incremented in a manner similar to that described above for a dc simulation (UP1, UP2, etc.). Note that an initial time step had to be specified on this card. The third SOLVE card ramps V1 down from 2 volts to -1 volts in 20 ns (end of ramp is at t = 45ns). The device is then solved at this bias for another 55 ns (out to 100 ns). Note that again each solution is saved in a separate file (DOWN1, DOWN2, etc.) and that no initial time-step was required since one had been estimated from the last transient solution for the previous SOLVE card. Finally, the fourth SOLVE card performs the steady-state solution at V1=-1 and V2=0.

```
METHOD  2ND  TAUTO  AUTONR
SOLVE   V1=1  V2=0
SOLVE   V1=2  TSTART=1E-12  TSTOP=25E-9  RAMPTIME=10E-9
+       OUTF=UP1
SOLVE   V1=-1  TSTOP=100E-9  RAMPTIME=20E-9  OUTF=DOWN1
SOLVE   V1=-1  V2=0
```

Finally, an AC example is presented. Assume the device to be simulated has 3 electrodes. Starting from solved DC conditions at V1 = 0, 0.5, 1.0, 1.5 and 2.0 volts, 10 mV AC signals of frequency 1 MHz, 10 MHz, 100 MHz, 1 GHz, 10 GHz and 100GHz are applied to each electrode in the device. Note that the number of AC solutions to be performed is 5*6*3=90.

```
SOLVE   V1=0  V2=0  V3=0  VSTEP=0.5  NSTEPS=4  ELECT=1
+       AC  FREQ=1E6  FSTEP=10  MULT.F  NFSTEP=5  VSS=0.01
```

The SPREAD card

1. Syntax

SPread <direction> <region> <specifics>

2. Description

The SPREAD card provides a way to distort rectangular grids in the vertical direction to follow surface and junction contours. SPREAD is very useful in reducing the amount of grid for some specific problems, most notably MOSFET's. The SPREAD card is somewhat complicated; it is suggested to follow the supplied examples very carefully (particularly the MOSFET example in Chapter 5 in [1]).

3. Parameters

<direction>

One of:

LEft	\=	<logical>	(default is false)
Right	\=	<logical>	(default is false)

LEFT and RIGHT specify that the left and right-hand sides of the grid respectively be distorted.

<region>

Width	\=	<real>
Upper	\=	<integer>
LOwer	\=	<integer>

WIDTH specifies the width from the left or right edge (depending on the LEFT and RIGHT parameters) of the distorted area. The actual x-coordinate specified by WIDTH (min[x] + WIDTH for LEFT, max[x] - WIDTH for RIGHT) will lie in the middle of the transition region between the distorted and undistorted grid regions. UPPER and LOWER specify the upper and lower y-grid lines between which the distortion will take place.

<specifics>

One of:

Y.Lower	\=	<real>
Thickness	\=	<real>

And:

Vol.ratio	\=	<real>	(default is 0.44)
Encroach	\=	<real>	(default is 1.0)
GRAding	\=	<real>	(default is 1.0)
GR1	\=	<real>	(default is 1.0)
GR2	\=	<real>	(default is 1.0)
Middle	\=	<real>	
Y.Middle	\=	<real>	

The Y.LOWER and THICKNESS parameters define the distorted grid region; only one should be supplied. Y.LOWER is the physical location in the distorted region at which the line specified by LOWER will be moved. The line specified by UPPER is not moved. THICKNESS is the thickness of the distorted region; THICKNESS will usually move the positions of both the UPPER and LOWER grid lines (unless VOL.RATIO is set to 0 or 1). VOL.RATIO specifies the ratio of the downward displacement of the lower grid line to the net increase in thickness. The default is 0.44 so that oxide-silicon interfaces are correct. VOL.RATIO is ignored if Y.LOWER is specified. ENCROACH is a factor which defines the abruptness of the transition between distorted and non-distorted grid. The transition region becomes more abrupt with smaller ENROACH factors (the minimum is 0.1). An important note: depending on the characteristics of the undistorted grid, very bad triangles (long, thin and obtuse) may result if ENCROACH is set too low. GRADING specifies a grid ratio (identical to the RATIO parameter on the X.MESH and Y.MESH cards) to produce a non-uniform grid in the distorted region. As alternative to a single grading parameter, GR1 and GR2 can be specified along with the y grid line MIDDLE and location Y.MIDDLE so that GR1 is used as the grading in the spread region from UPPER to MIDDLE and GR2 is the grading from MIDDLE to LOWER.

4. Examples

The following spreads what was previously a uniform 400 Angstroms of oxide to 1000 Angstroms on the left side of the device. This will result in a net increase in thickness of 600 Angstroms of oxide. Because the default VOL.RATIO is used, 0.44*(600) = 264 Angstroms of the net increase will lie below the original 400 Angstroms and 0.56*(600) = 336 Angstroms of the net increase will lie above the original 400 Angstroms. The width of the spread region is 0.5 μm and the oxide taper is quite gradual because of the high encroachment factor. The grid is left uniform in the spread region.

```
$ *** Mesh definition ***
MESH     NX=30 NY=20 RECT
X.M      N=1  L=0
X.M      N=30 L=2
Y.M      N=1  L=-.04
Y.M      N=5  L=0
Y.M      N=20 L=1 R=1.4

$ *** Thin oxide ***
REGION   X.L=1 X.H=30 Y.L=1 Y.H=5

$ *** Silicon substrate ***
REGION   X.L=1 X.H=30 Y.L=5 Y.H=20

$ *** Spread ***
SPREAD   LEFT WIDTH=0.5 UP=1 LO=5 THICK=0.1 ENC=1.3
```

In the next example, the right side of the grid is distorted in order to follow a junction contour. Assume that the initial grid is defined as above. Y.LOWER is used so that there is no increase in the size of the device, just grid redistribution. With Y.LOWER set to the junction, the ENCROACH parameter should be chosen such that the lower grid line (LOWER=10) follows the junction as closely as possible. Note that the grid is graded so that the grid lines are spaced closer together as they approach the junction. Because the point specified by WIDTH on the spread card lies in the middle of the transition region, it should be chosen to be slightly larger than the width of the doping ‘‘box’’ (WIDTH < X.LEFT - X.RIGHT = 0.5 μm).

```
$ *** Doping ***
DOPING   UNIFORM  N.TYPE  CONC=1E15
DOPING   GAUSS  P.TYPE  X.LEFT=1.5  X.RIGHT=2
+          PEAK=0  CONC=1e19  RATIO=.75  JUNC=0.3

$ *** Spread ***
SPREAD   RIGHT  WIDTH=0.7  UP=5  LO=10  Y.LO=0.3
+          ENC=1.2  GRAD=0.7
```

The SYMBOLIC card

1. Syntax

SYmbolic <solution method> <carriers> <options>

2. Description

The symbolic card performs a symbolic factorization in preparation for the LU decompositions in the solution phase of PISCES-II. Because each of the available numerical solution techniques used by PISCES-II may result in entirely different linear systems, the method used and the number of carriers to be simulated must be specified at this time. The symbolic factorization may be optionally read from or written to a file; if an input file is specified, the symbolic factorization information in that file must be consistent with the method specified on the present card.

3. Parameters

<solution method>

One of:

Newton	\=	<logical>
Gummel	\=	<logical>

The solution method must always be specified, except for the Poisson only case (carriers = 0). The different methods and their applications are discussed in Chapter 3 of [1] and this report.

<carriers>

Carriers	\=	<integer>	(default is 1)
Electrons	\=	<logical>	(default is true)
Holes	\=	<logical>	(default is false)

The CARRIERS parameter specifies the number of carriers to be simulated. If only one carrier is to be simulated, the specific carrier can be specified by including either HOLES or ELECTRONS.

<options>

Min.degree	\=	<logical>	(default is true)
Strip	\=	<logical>	(default is true)
Infile	\=	<character string>	
Outfile	\=	<character string>	
Print	\=	<logical>	(default is false)

MIN.DEGREE uses a minimum degree ordering of the pivots for decomposition in order to reduce the size of the generated L and U and hence, to reduce the amount of cpu spent in solving linear systems. This parameter is definitely recommended. STRIP specifies that redundancy (zero couplings) be removed from the symbolic map, and is naturally on. The INFILE and OUTFILE parameters specify input/output files names for the symbolic factorization. The file names may contain up to 20 characters. Note that these binary files can be quite large, so it may be advisable not to use this feature. (In some computing environments it is also faster to compute the symbolic information than to read it from disk). The PRINT parameter indicates that information about the memory allocated for the run should be printed to the PISCES-II standard output file.

4. Examples

The following specifies a symbolic factorization for a simulation with only holes and using the Gummel method (the symbolic factorization is saved in a file called SYMB.OUT):

```
SYMBOLIC   GUMMEL  CARR=1  HOLES  OUTF=SYMB.OUT
```

In the next example, a previously generated symbolic factorization is read in from a file called SYMB.IN. The method used is the full Newton method, and both carriers are included in the simulation. Additionally, the sizes and dimensions of all arrays associated with the sparse matrix solution are printed:

```
SYMBOLIC   NEWTON  CARR=2  INF=SYMB.IN  PRINT
```

The TITLE card

1. Syntax

```
Title <character string>
```

2. Description

The TITLE card specifies a title (up to 60 characters) to be used in PISCES-II standard output.

3. Examples

```
TITLE   *** CMOS p-channel device ***
```

The VECTOR card

1. Syntax

Vector <plotted quantity> <control>

2. Description

The VECTOR card plots vector quantities over an area of the device defined by the previous PLOT.2D card.

3. Parameters

<plotted quantity>

One of:

J.Conduc	\=	<logical>	Conduction current
J.Electr	\=	<logical>	Electron current
J.Hole	\=	<logical>	Hole current
J.Displa	\=	<logical>	Displacement current
J.Total	\=	<logical>	Total current
E.field	\=	<logical>	Electric field

The above parameters specify the quantity to be plotted. There is no default.

<control>

LOgarithm	\=	<logical>	(default is false)
MInimum	\=	<real>	(default is 0)
MAximum	\=	<real>	(default is 0)
Scale	\=	<real>	(default is 1)
Clipfact	\=	<real>	(default is 0.1)
LIne.type	\=	<integer>	(default is 1)

LOGARITHM specifies logarithmically-scaled magnitudes. By default, all vectors in a linear plot are scaled by the maximum magnitude of the quantity of interest over the grid, while if LOGARITHM is set, the default scaling uses the minimum (non-zero) magnitude. The minimum and maximum values, both of which are printed during execution of a plot, can be arbitrarily set using the MINIMUM and MAXIMUM parameters, so that it is possible to plot two bias conditions or devices with the same scaling. SCALE a constant scale factor to be multiply all magnitudes by. CLIPFACT is a threshold below which vectors are not plotted. LINE.TYPE specifies the vector line type for plotting.

4. Examples

Plot electron and hole currents over a device :

```
PLOT.2D BOUN NO.FILL
VECTOR J.ELEC LINE=2
VECTOR J.HOLE LINE=3
```

The X.MESH card

The Y.MESH card

1. Syntax

```
X.Mesh   <node>   <location>   <ratio>
Y.Mesh   <node>   <location>   <ratio>
```

2. Description

The X.MESH and Y.MESH cards specify the location of lines of nodes in a rectangular mesh.

3. Parameters

<node>

Node \= <integer>

This is the number of the line in the mesh. There can be at most 120 lines in either direction. Lines are assigned consecutively, beginning with the first and ending with the last.

<location>

Location \= <real>

This is where to locate the line. The location is interpreted in microns.

<ratio>

Ratio \= <real>

This gives the ratio to use when interpolating lines between the given lines. The spacing grows/shrinks by ratio in each subinterval, and the ratio should usually lie between 0.667 and 1.5.

4. Examples

Space grid lines closely around a junction in a 1-d diode with the junction at 0.85 microns:

```
Y.MESH  N=1  LOC=0.0
Y.MESH  N=20  LOC=0.85  RATIO=0.75
Y.MESH  N=40  LOC=2  RATIO=1.333
```