EXTRACT

The **EXTRACT** statement extracts selected data from the solution over a specified cross-section of the device.

```
EXTRACT

 Extraction Using Names and Expressions
   {  ( EXPRESSI=<c> NAME=<c> [UNITS=<c>] [CONDITIO=<c>] [INITIAL=<n>]
        [ {AT.BIAS | NOW} ] [OUT.FILE=<c> TWB] [CLEAR] [PRINT]

 Optimization Using Targets and Expressions
        [ TARGET=<n> [WEIGHT=<n>] [MIN.REL=<n>] [MIN.ABS=<n>]
          [TARTOL=<n>] [TARREL=<n>]
        ]
      )

 Extract Physical Quantities from Solution
    | ( {  NET.CHAR | NET.CARR | ELECTRON | HOLE | RECOMBIN | IONIZATI
         | RESISTAN | N.RESIST | P.RESIST | ( METAL.CH CONTACT=<c> )
         | ( {N.CURREN | P.CURREN} {CONTACT=<c> | REGIONS=<c>} )
         | II.GENER | (SHEET.RE X.POINT=<n>)
        }
        [X.MIN=<n>] [X.MAX=<n>] [Y.MIN=<n>] [Y.MAX=<n>]
        [OUT.FILE=<c>]

 Device to Extract (Circuit Analysis AAM)
        [STRUCTUR=<c>]
      )

 Extract MOS Device Parameters
    | ( MOS.PARA [DRAIN=<c>] [GATE=<c>] [IN.FILE=<c>] [I.DRAIN=<n>]
        [OUT.FILE=<c> [TWB] ] [CONDITIO=<c>] [APPLIED]
      )
   }
```

Parameter	Type	Definition	Default	Units
Extraction Using Names and Expressions				
NAME	char	Name of the assigned variable where the numeric value calculated by **EXPRESSI** is to be stored.	none	
EXPRESSI	char	Specifies the numeric character expression which is to be evaluated. This expression may contain assigned variables or solution variables.	none	
UNITS	char	Units to be used for the extracted quantity. These are used during plotting.	none	
CONDITIO	char	Specifies a numeric character expression which must evaluate to "TRUE" before **EXPRESSI** is evaluated.	TRUE	

Parameter	Type	Definition	Default	Units
INITIAL	number	Specifies the initial value to use during evaluation of **EXPRESSI**. This parameter is useful mainly for accumulator type expression such as: **EXTRACT NAME=DOG EXPRESS=MAX(@DOG;@TL) INITIAL=0**. Here the variable DOG must be assigned some value at the start of evaluation, its value comes from **INITIAL=0**. **synonym: INITIAL.V**.	0	none
AT.BIAS	logical	Specifies that the expressions **CONDITIO** and **EXPRESSI** are evaluated at the conclusion of each bias / time point.	TRUE, unless **NOW**=TRUE	
NOW	logical	Specifies that the expressions **CONDITIO** and **EXPRESSI** are evaluated when the **EXTRACT** statement is encountered. If this parameter is not specified, the extraction is performed after each **SOLVE** step or when a **PLOT.1D** statement is encountered.	false	
OUT.FILE	char	Specifies a file to which **TARGET** information is written to in **TWB** format.	none	
TWB	logical	Specifies that the result calculated by **EXPRESSI** is to be stored in the file specified by **OUT.FILE** in the format used by **TWB**. **synonym: CAESAR**.	false	
CLEAR	logical	Specifies that all previous **EXTRACT** expressions are deleted.	false	
PRINT	logical	Specifies that the value of the extracted expression is displayed on the user's terminal and printed to the standard output file.	false	
Optimization Using Targets and Expressions				
TARGET	number	The desired value of the target being defined for an optimization loop. The optimization attempts to match the extracted value with the value of this parameter.	none	Defined by the extracted value.
WEIGHT	number	The weighting factor applied to the target being defined for optimization. The weights are used to control the relative importance of the individual targets in calculation of error during optimization.	1.0	none
MIN.REL	number	The minimum target ratio for which relative error is used to calculate error during optimization.	1e-2	none
MIN.ABS	number	The minimum target value for which relative error is used to calculate the error during optimization.	1e-10	none
TARTOL	number	The target tolerance used during optimization.	5	%
TARREL	number	The relative target tolerance used during optimization (see below).	0.1	%
Extract Physical Quantities from Solution				
NET.CHAR	logical	Specifies that the integrated net charge is extracted.	false	
NET.CARR	logical	Specifies that the integrated net carrier concentration is extracted.	false	
ELECTRON	logical	Specifies that the integrated electron concentration is extracted.	false	
HOLE	logical	Specifies that the integrated hole concentration is extracted.	false	
RECOMBIN	logical	Specifies that the integrated recombination rate is extracted.	false	
IONIZATI	logical	Specifies that the maximum electron and hole ionization integrals are extracted.	false	

Parameter	Type	Definition	Default	Units
RESISTAN	logical	Specifies that the resistance (including both electrons and holes) of a cross-section is extracted. The current flow is assumed to be perpendicular to the xy plane. The result is given in Ohms per micron of distance perpendicular to the xy plane.	false	
N.RESIST	logical	Specifies that the n-resistance of a cross-section is extracted. The current flow is assumed to be perpendicular to the xy plane. The result is given in Ohms per micron of distance perpendicular to the xy plane.	false	
P.RESIST	logical	Specifies that the p-resistance of a cross-section is extracted. The current flow is assumed to be perpendicular to the xy plane. The result is given in Ohms per micron of distance perpendicular to the xy plane.	false	
METAL.CH	logical	Specifies that the integrated charge on an electrode is extracted.	false	
CONTACT	char	The electrode name over which the integration takes place when **METAL.CH**, **N.CURREN**, or **P.CURREN** are specified. Only nodes falling within the specified bounds and belonging to the electrode are included in the integration.	none	
N.CURREN	logical	Specifies that the electron current through an electrode or the electron current across the boundary between two regions is extracted.	false	
P.CURREN	logical	Specifies that the hole current through an electrode or the hole current across the boundary between two regions is extracted.	false	
REGIONS	char	The names of two adjacent regions. Electron or hole current flow across the boundary between these regions is extracted if **N.CURREN** or **P.CURREN**, respectively, is specified. The two region names should be enclosed in parentheses and separated by a comma.	none	
II.GENER	logical	Specifies that the number of electron-hole pairs generated by impact ionization in the entire device or, if specified, in the area identified by **X.MIN**, **X.MAX**, **Y.MIN**, and **Y.MAX**, is extracted.	false	
SHEET.RE	logical	Extracts the sheet resistance of the layers intersected by the vertical line at x=**X.POINT** producing a table giving the resistance of each doped layer. Current flow is assumed to be perpendicular to xy plane.	false	
X.POINT	number	The x location of the line along which to extract sheet resistance.	none	microns
X.MIN	number	The minimum x coordinate of the region over which the specified quantity is extracted.	The minimum x location in the device structure.	microns
X.MAX	number	The maximum x coordinate of the region over which the specified quantity is extracted.	The maximum x location in the device structure.	microns
Y.MIN	number	The minimum y coordinate of the region over which the specified quantity is extracted.	The minimum y location in the device structure.	microns
Y.MAX	number	The maximum y coordinate of the region over which the specified quantity is extracted.	The maximum y location in the device structure.	microns
OUT.FILE	char	The identifier of a formatted output file for storing the extracted information. **synonym: OUTFILE**	none	

Parameter	Type	Definition	Default	Units
Device to Extract (Circuit Analysis AAM)				
STRUCTUR	char	Specifies the device in which the extraction is performed. This parameter is only used with the Circuit Analysis AAM.	first element	
Extract MOS Parameters				
MOS.PARA	logical	Specifies that various parameters associated with MOS devices are extracted. If an I-V log file is available, or if one is specified with **IN.FILE**, the program attempts to extract information such as threshold voltage and subthreshold slope. If a mesh file is available, the program attempts to extract the channel length for the structure.	false	
DRAIN	char	The electrode name associated with the drain in an open log file, or one that is read in using **IN.FILE**.	none	
GATE	char	The electrode name associated with the gate in an open log file, or one that is read in using **IN.FILE**.	none	
IN.FILE	char	The identifier of an I-V log file to use for extracting MOS parameters.	Currently open I-V log file.	
I.DRAIN	number	The drain current for which the corresponding voltage is extracted. The extraction is performed using both linear and logarithmic interpolation.	none	A/micron
CONDITIO	char	Specifies a numeric expression that must evaluate to "TRUE" before data will be used for the MOS parameter extraction.	none	
APPLIED	logical	Specifies that applied biases are used for the MOS parameter extraction. Specifying **^APPLIED** will cause the contact voltages to be used.	true	

Description

The **EXTRACT** statement extracts selected data from the solution over a specified cross-section of the device.

See Also…

To further illustrate the **EXTRACT** statement, refer to input files:

- *mdex1m* in N-Channel MOSFET Examples, Chapter 4, "Analysis Including Band-to-Band Tunneling" on page 4-25
- *mdex2fp* in NPN Bipolar Transistor Examples, Chapter 5, "Simulation of Forward Characteristics" on page 5-5
- *mdex2m* in NPN Bipolar Transistor Examples, Chapter 5, "Simulation of a One-Dimensional Bipolar Transistor" on page 5-18

Extraction with Expressions and Names

This capability allows access to internal (predefined) variables through an algebraic expression. The expression is evaluated (at each device node if required) and stored in the variable given by **NAME**. The **NAME** variable may then be used for plotting, to control program execution, or in other expressions.

If two extract statements use the same **NAME**, the second extract statement replaces the first, thus the following two statements:

```
EXTRACT   NAME=DOG  EXP="@X*@X"
EXTRACT   NAME=DOG  EXP="@X*@X*@X"
```

would assign x^3 to the variable dog. All extract definitions may be deleted by use of the **CLEAR** parameter.

Four examples of **EXTRACT** capabilities are presented below.

Extraction of Maximum Electric Field

Calculate the maximum value of the electric field within the portion of the device x>5 and y<3. Next on the **EXTRACT** statement set **EMAX=MAX(@EMAX;E(i))** where **E(i)** represents the magnitude of the electric field at each grid point. The **CONDITIO** parameter is used to limit the points to be checked to those within the portion of the device where x>5 and y<3. The parameter **INITIAL** is used to set the target EMAX to zero at the start of the calculation.

```
EXTRACT  EXPRESS="MAX(@EMAX;@EM)"  COND="@X>5&@Y<3"  NAME=EMAX
+        INITIAL=0
```

$\vec{J} \cdot \vec{E}$ Calculation

Plot the heating term $\vec{J} \cdot \vec{E}$. The predefined array **ARRAY1** is used to hold the result of the calculation. The program multiplies J_n and E at each grid point and stores the result in the array **ARRAY1**. **ARRAY1** may then be plotted in the normal way.

```
EXTRACT  EXPRESS="(@JNX+@JPX)*@EX+(@JNY+@JPY)*@EY"  NAME=ARRAY1
PLOT.2D  BOUND
CONTOUR  FILL  ARRAY1
```

Defining Mobility and Lifetime Values

Note that lifetime and mobility are predefined quantities that are "writable". This means they can be used to define the mobility and lifetime to be used during the solution process. For example, the following statement can be used to define the low-field electron mobility in the structure based on total donor concentration and lattice temperature:

```
EXTRACT  NAME=mobn NOW ^AT.BIAS
+        EXPRESS=1000*(1e18/@Nd)*(300/@tl)**2
```

CAUTION

When defining mobility or lifetime using an EXTRACT expression, models should not be specified that will cause the defined values to be overwritten. For example, if "MODELS CONMOB FLDMOB" is specified, then during the iterative solution process, low-field mobility will be calculated using CONMOB rather than the defined expression on the EXTRACT statement.

Binary Search

Use a binary search to find the gate voltage needed to make the drain current equal to 1mA. Search the interval Vg=1V to Vg=5V. The **EXTRACT** statement is used to check whether or not the drain current is greater than 1e-3A. If it is, the assigned variable **TEST** is set to 1. Next, check **TEST**; if **TEST**=1, decrease the bias.

This example illustrates a powerful method of extraction or optimization. This case varied a bias voltage, but a doping profile or model parameter could have been varied instead.

```
MESH .....
SYMBOLIC .....
ASSIGN  NAME=LOW N.VAL=1
ASSIGN  NAME=HIGH N.VAL=5
ASSIGN  NAME=STEP  N.VAL=(@HIGH-@LOW)/2
ASSIGN  NAME=VA  N.VAL=@LOW+@STEP
EXTRACT EXPRESS="@I(Drain)>1E-3"  NAME=TEST
LOOP STEPS=10
   SOLVE V(Gate)=@VA
   ASSIGN NAME=STEP  N.VAL=@STEP/2
   IF COND=@TEST
      ASSIGN NAME=VA N.VAL=@VA-@STEP
   ELSE
      ASSIGN NAME=VA N.VAL=@VA+@STEP
   IF.END
L.END
ECHO "THE REQUIRED VOLTAGE IS: "@VA" +/-"@STEP
```

Creating a Ring of Constant Doping

The following example creates a ring of dopant with a constant density of le17 cm^{-3}, with an inner radius of 5 microns and an outer radius of 8 microns.

```
EXTRACT  EXPRESS="@NET+1E17"  NAME=NET
         COND="(@X*@X+@Y*@Y)>25&(@X*@X+@Y*@Y)<64"
```

Optimization

The optimization functions built into the input parser can be used for optimizing a wide variety of parameters such as:

- Bias voltages
- Doping profiles
- Model parameters

The following example optimizes two doping profiles in a simple 1D diode to give specific IV results:

```
LOOP  OPTIMIZE
      ASSIGN NAME=DOP1 UP=1E20 LOW=2E17 N.VAL=3E18 OPTIMIZE
      ASSIGN NAME=DOP2 UP=1E17 LOW=1E14 N.VAL=1E16 OPTIMIZE
      MESH
      X.MESH   WIDTH=1  N.SPACES=1
      Y.MESH   WIDTH=1  N.SPACES=50
      ELECT    NUM=ANODE  TOP
      ELECT    NUM=CATHODE  BOTTOM
      PROFILE  N-TYPE  N.PEAK=@DOP2  UNIF
      PROFILE  P-TYPE  N.PEAK=@DOP1  Y.JUNC=0.2
      SYMB     NEWT  CARR=2
      EXTRACT  NAME=P1  EXP="@I(ANODE)"  TARG=2.2E-8
+              COND="@V(ANODE)=.5"
      EXTRACT  NAME=P2  EXP=@I(ANODE)  TARG=4.6E-7
+              COND="@V(ANODE)=.6"
      SOLVE    V(ANODE)=0.5
      SOLVE    V(ANODE)=0.6
L.END
```

where:

- The **LOOP** statement defines the start of the optimization loop.
- The statements between the **LOOP** and the **L.END** are repeated until the optimization is complete.
- The **OPTIMIZE** parameter on the **LOOP** statement tells the program to perform the optimization.
- The two **ASSIGN** statements set up the following:
  - The initial values for the variables DOP1 and DOP2
  - The **LOWER** and **UPPER** bounds to constrain the optimization
  - Tells the program to actually optimize these variables via the **OPTIMIZE** parameter
- The two **PROFILE** statements make use of the optimized doping values via the DOP1 and DOP2 assigned used as the values for **CONC**.
- The two **EXTRACT** statements are used to extract the anode currents into variables P1 and P2.

  At the same time the **EXTRACT** statement defines the **TARGET** values for the optimization.
- The program then tries to adjust DOP1 and DOP2 so that the extracted anode currents match the targets (2.2e-8 and 4.6e-7).

Optimization Guidelines

Device simulation and optimization can be a lengthy process. Since both device simulation and optimization are highly non-linear processes there is no guarantee that the optimization will succeed. To get the most from the optimization, *Avant!* TCAD recommends the following guidelines.

Parameter Selection

- Optimize as few parameters as possible.
- While the **EXTRACT** statement can optimize up to 20 parameters at once, a practical upper limit is about three or four.
- If possible, break a complex optimization up into several smaller optimizations.
- For example, for a MOSFET, you may want to first optimize mobility parameters which effect the forward region and then separately optimize band-to-band tunneling parameters in the reverse region.

Initial Guess and Limits

- Carefully choose the initial guess and limits.
- Use the **UPPER** and **LOWER** parameters to constrain the solution to values which should be close to the true solution.

The optimizer can become easily confused if the initial guess is far from the true solution. The reason is that the optimizer only finds local minima. If a local minimum lies between the initial guess point and the true solution the optimizer finds the local minimum, not the true solution. In a solution space of several variables, there may be a great many local minima, but only one true solution.

In general, picking many parameters, using randomly chosen initial values and letting the simulator "sort them out" does not work.

Examining the Results

- Carefully examine the results of the optimization.

The minimum found is not guaranteed to give a good fit and even if it does, the parameter set that was found may not be unique or the best set. The parameters which give the best fit may not be the best choices from the standpoint of producing predictive simulations. This is particularly true if a large number of parameters are optimized.

Terminating the Process

There are two parameters which determine when the optimization process terminates.

- **TARTOL** is the absolute tolerance and the optimization stops as soon as the error becomes less than **TARTOL** percent.
- **TARREL** stops the optimization when the error between two consecutive optimization steps becomes less than **TARREL**.

  The second limit is needed since in general the bottom of a minima is not at zero. At the bottom of a minima, however, the error stops changing and **TARREL** terminates the process.

Simple Extraction

The **EXTRACT** statement can be used for the following "simple" processes:

- To integrate concentrations over a specified cross-section of a device for net carrier, net charge, electron, and hole.
- To extract the charge on part of an electrode, as well as the current through that part.

  This is useful for capacitance studies, in conjunction with the difference mode on the **LOAD** statement.
- The current flow across the boundary between two adjacent regions can be extracted by using the **REGIONS** parameter.

  The resistance of a cross-sectional structure, for instance a diffused line, can also be extracted.
- Ionization integrals for electrons and holes are computed by using the **IONIZATI** parameter.

  Ionization integrals are computed along potential gradient paths initiated at nodes lying in the rectangular region defined by the parameters **X.MIN**, **X.MAX**, **Y.MIN**, and **Y.MAX**.

  The output consists of the maximum ionization integral for both electron and holes, the peak field along the path that produced the maximum ionization integral, and the location of the peak field along this path.

Extraction of MOS Device Parameters

Various parameters associated with MOS devices can be extracted by selecting **MOS.PARA**. If an I-V log file is available, or if one is specified with **IN.FILE**, the program attempts to extract information such as threshold voltage and sub-threshold slope. The program will look for electrodes that have names "gate" and "drain." If these names are not available for the device structure, then you must specify the electrode names that correspond to the gate and drain using the **GATE** and **DRAIN** parameters, respectively.

If a valid Medici device structure is available, the program attempts to extract the MOS channel length from it. For this purpose, the channel length is defined as the distance between the source-channel and drain-channel metallurgical junctions at the semiconductor-insulator interface.

Example:

```
EXTRACT  MOS.PARA  IN.FILE=GATE.IVL
```

An example of the output produced by this statement follows:

```
 MOS Parameter Extraction
------------------------
Lchan:          1.50238      microns
  x(start) =   0.748808       y(start) =   6.914763E-04
  x(end)   =    2.25119       y(end)   =   6.914803E-04

file: GATE.IVL
number of data values:   11
first point: Vg=      0.    , Vd=  0.1000    , Id=  8.8634E-12
last point:  Vg=   2.000    , Vd=  0.1000    , Id=  8.2236E-06
 S_lin (linear slope):    5.5002E-06 (A/um-V)   at Vg=  0.8000     (V)
      Vth (intercept):    0.3621     (V)
 S_sub (subthreshold slope):   81.76    (mV/dec) at Vg=  0.2000     (V)
```

Predefined Quantities in Expressions

Expressions follow the general rules set forth in "Numerical Expressions" on page 3-5. The following predefined quantities are available for use in calculations. These quantities allow access to the simulation structure and the solution process.

The special variables are accessed in parameter expressions by prefixing the variable name with an at sign "@" in the same way as normal assigned variables. Some of the quantities are writable meaning that the value currently in use by the simulator can be altered. For example, you can create new initial guesses by altering *n*, *p*, or *pot*. Footnotes are at the end of the table.

Note:

In addition to the predefined quantities listed below, quantities that were originally defined on the* `PROFILE` *statement using the* `IMPURITY` *or* `OTHER` *parameters can also be used in expressions by prefixing their names with "@". The* `IMPURITY` *and* `OTHER` *quantities are also writable.

Table 3-1 Predefined Quantities Used in Calculations

Name	Quantity	Units	Writable?
x	Node x coordinate	Microns	No
y	Node y coordinate	Microns	No
net	Net node doping	#/cm^3	Yes
total	Total node doping	#/cm^3	Yes
Na	Total acceptor concentration	#/cm^3	No
Nd	Total donor concentration	#/cm^3	No
region	Node region #	-	No
node.num	Node number	-	No
interfac	At interface?	-	No
electrod	Node electrode #	-	No
taun	Electron lifetime	sec^{-1}	Yes
taup	Hole lifetime	sec^{-1}	Yes

Table 3-1 Predefined Quantities Used in Calculations

Name	Quantity	Units	Writable?
mobn	Low-field electron mobility	cm^2/V-sec	Yes
mobp	Low-field hole mobility	cm^2/V-sec	Yes
photogen	Photogeneration rate (1)	Ehp/cm^3/sec	Yes
time	Time	Seconds	No
delt	Current time step	Seconds	No
iterat	Number of Newton iterations	-	No
nx.int	At node next to interface	-	No
sem.area	Semiconductor area of a node	cm^2	No
ins.area	Insulator area of a node	cm^2	No
prp.dist	Perpendicular interface dist (2)	cm	No
pot	Node potential	Volts	Yes
n	Node electron conc	#/cm^3	Yes
p	Node hole conc	#/cm^3	Yes
tn	Node electron temp.	Kelvin	Yes
tp	Node hole temp	Kelvin	Yes
tl	Node Lattice temp	Kelvin	Yes
jnm	Electron current density	Amps/cm^2	No
jpm	Hole current density	Amps/cm^2	No
jdm	Displacment current density	Amps/cm^2	No
jtm	Total current density	Amps/cm^2	No
ii.gener	Impact Ionization rate	#/cm^3/sec	No
recomb	Recombination rate (for unequal electron and hole recombination, **recombin** is the same as **n.recomb**)	#/cm^3/sec	No
n.recomb	Electron recombination rate	#/cm^3/sec	No
p.recomb	Hole recombination rate	#/cm^3/sec	No
em	Electric field	Volts/cm	No
bb.gener	Band-to-band generation rate	#/cm^3/sec	No
trap.occ	Trap occupation	#/cm^3	No
qfn	Electron quasi Femi level	eV	No
qfp	Hole quasi Fermi level	eV	No
conduc.b	Conduction band energy	eV	No
valenc.b	Valence band energy	eV	No
vacuum	Vacuum level	eV	No

Table 3-1 Predefined Quantities Used in Calculations

Name	Quantity	Units	Writable?
i(i)	Terminal current (3)	Amps/micron	No
va(i)	Applied voltage at the terminal	Volts	No
v(i)	Terminal Voltage	Volts	No
he(i)	Hot electron gate current at term.	Amps/micron	No
fe(i)	Fowler Nordheim current at term.	Amps/micron	No
q(i)	Terminal charge	Coul./micron	No
freq	AC analysis frequency	Hz	No
g(i,j)	AC small signal conductance	Mhos/micron	No
y(i,j)	Y parameter at terminal	Mhos/micron	No
c(i,j)	AC small signal capacitance	Farads/micron	No
sr(i,j)	Real S parameter (4)	none	No
si(i,j)	Imaginary S parameter (4)	none	No
vc(i)	Circuit node voltage	Volts	No
ic(i)	Circuit inductor/voltage src. current	Amps	No
vcr(i)	AC node voltage, real part	Volts	No
icr(i)	AC ind/voltage src. current (real)	Amps	No
vci(i)	AC node voltage imaginary part	Volts	No
ici(i)	AC ind/voltage src. current (Imag)	Amps	No
array1	Scratch array 1	-	Yes
array2	Scratch array 2	-	Yes
array3	Scratch array 3	-	Yes
ii.n.int	Electron ionization integral (**synonym: ii.integ**)	-	No
ii.p.int	Hole ionization integral	-	No
jnx	x-component elect. curr density	A/cm^2	No
jny	y-component elect. curr density	A/cm^2	No
jpx	x-component hole curr density	A/cm^2	No
jpy	y-component hole curr density	A/cm^2	No
ex	x-component Electric field	V/cm	No
ey	y-component Electric field	V/cm	No
nie	Intrinsic Carrier concentration	$\#/cm^3$	No
net.carr	net carriers	$\#/cm^3$	No
net.char	net charge	$\#/cm^3$	No
gin	Hot electron injection current	A/micron	No

Table 3-1 Predefined Quantities Used in Calculations

Name	Quantity	Units	Writable?
gip	Hot hole injection current	A/micron	No
x.mole	mole Fraction	none	No
Lchan	Channel length	microns	No
Vth	Threshold voltage(5)	Volts	No
S_lin	Linear region slope(5)	A/micron-V	No
S_sub	Subthreshold region slope(5)	mV/decade	No
Vg_lin	V_g where I_d = user specified value (using linear interpolation)(6)	Volts	No
Vg_log	V_g where I_d = user specified value (using logarithmic interpolation)(6)	Volts	No
Vd_lin	V_d where I_d = user specified value (using linear interpolation)(7)	Volts	No
Vd_log	V_d where I_d = user specified value (using logarithmic interpolation)(7)	Volts	No
Vth_sat	Saturation region intercept.(8)	Volts	No

Table Notes:

(1) This is the case for steady state simulations. In the case of time-dependent simulations, the rate is integrated over the duration of the simulation and a rate in #/cm^3 is used.

(2) This quantity gives the perpendicular distance to the interface associated with each interface node. If the node is not an interface node, 0 is returned. Refer to the section on the **TRAPS** statement for use.

(3) The "i" and "j" arguments used with the names of terminal characteristics represent valid electrode names. If "drain" and "gate" are valid electrode names in a simulation, then "c(gate,drain)" could be used in an expression.

(4) Only the real and imaginary parts of S(1,1), S(1,2), S(2,1), S(2,2) may be used in expressions.

(5) An appropriate set of I_d-V_g data must be available to extract this quantity. Vth is defined as the intercept with the V_g axis from the point of maximum slope.

(6) This quantity is only available if an appropriate set of I_d-V_g data is available and if a previous **EXTRACT** statement specified "**MOS.PARA I.DRAIN**=<n>"

(7) This quantity is only available if an appropriate set of I_d-V_d data is available and if a previous **EXTRACT** statement specified "**MOS.PARA I.DRAIN**=<n>"

(8) This quantity is only available if I_d-V_g data is available for which V_d=V_g. Vth_sat is defined as the intercept with the V_g axis from the point of maximum slope of the $\sqrt{I_d}$ - V_g curve.