LOG

The **LOG** statement allows the I-V and/or AC terminal data for a simulation to be written to a file. This statement also provides for the conversion of Medici I-V log files to data files that can be used by common parameter extraction programs.

```
LOG

{  ( [OUT.FILE=<c> [TIF] ] [CLOSE] )
 | ( [ {  ( AURORA [LENGTH=<n>] [WIDTH=<n>] [DEVID=<n>] [REGION=<n>]
                    [P.CHANNE]
           )
         | ( ICCAP [OUT1=<c>] [OUT2=<c>] [OUT3=<c>]
                    [INP1=<c>] [INP2=<c>] [INP3=<c>] [INP4=<c>]
           )
         | STANFORD
       }
       IN.FILE=<c> OUT.FILE=<c>
       [GATE=<c>] [SOURCE=<c>] [DRAIN=<c>] [SUBSTRAT=<c>]
       [BASE=<c>] [EMITTER=<c>] [COLLECT=<c>] [EXTRA=<c>]
     ]
   )
}
```

Parameter	Type	Definition	Default	Units
OUT.FILE	char	The identifier for the file to store data obtained at each bias or time point. In the case where Medici data is being converted to a new format, this parameter specifies the name of the new file. **synonym: IVFILE**	none	
TIF	logical	Specifies that the log file is to be written in the TIF format.	true	
CLOSE	logical	Specifies that the presently open log file is to be closed so that no additional data is written to the file.	false	
AURORA	logical	Specifies that a data file for *Avant!* TCAD's Aurora program is created. **synonym: TOPEX**	none	
LENGTH	number	The device channel length to be written to output files created for Aurora.	none	microns
WIDTH	number	The device channel width to be written to output files created for Aurora. This value also multiplies the values of current stored in the input I-V file before they are written to the output file.	none	microns
DEVID	number	An optional device identification parameter written to output files created for Aurora.	none	none
REGION	number	An optional device behavior region parameter written to output files created for Aurora.	none	none
P.CHANNE	logical	Specifies that the sign of all voltages and currents are changed before writing the data to the Aurora output file.	false	

Parameter	Type	Definition	Default	Units
ICCAP	logical	Specifies that a data file for Hewlett-Packard's IC-CAP program is created.	none	
OUT1	char	The first output variable written to the data file created for a parameter extraction routine. Valid choices include applied bias *VA(name)*, contact bias *V(name)*, and terminal current *I(name)*.	none	
OUT2	char	The second output variable written to the data file created for a parameter extraction routine. Valid choices include applied bias *VA(name)*, contact bias *V(name)*, and terminal current *I(name)*.	none	
OUT3	char	The third output variable written to the data file created for a parameter extraction routine. Valid choices include applied bias *VA(name)*, contact bias *V(name)*, and terminal current *I(name)*.	none	
INP1	char	The first input variable written to the data file created for a parameter extraction routine. Valid choices include applied bias *VA(name)*, contact bias *V(name)*, and terminal current *I(name)*.	none	
INP2	char	The second input variable written to the data file created for a parameter extraction routine. Valid choices include applied bias *VA(name)*, contact bias *V(name)*, and terminal current *I(name)*.	none	
INP3	char	The third input variable written to the data file created for a parameter extraction routine. Valid choices include applied bias *VA(name)*, contact bias *V(name)*, and terminal current *I(name)*.	none	
INP4	char	The fourth input variable written to the data file created for a parameter extraction routine. Valid choices include applied bias *VA(name)*, contact bias *V(name)*, and terminal current *I(name)*.	none	
STANFORD	logical	Specifies that the Medici log file identified with the **IN.FILE** parameter is converted to a Stanford PISCES-IIB log file format and stored in the file identified with **OUT.FILE**.	none	
IN.FILE	char	The identifier for a Medici I-V log file that is used to create a data file for a parameter extraction program. **synonym: INFILE**	none	
GATE	char	The electrode name associated with the gate for an MOS device.	none	
SOURCE	char	The electrode name associated with the source for an MOS device.	none	
DRAIN	char	The electrode name associated with the drain for an MOS device.	none	
SUBSTRAT	char	The electrode name associated with the substrate for either an MOS or bipolar device.	none	
BASE	char	The electrode name associated with the base for a bipolar device.	none	
EMITTER	char	The electrode name associated with the emitter for a bipolar device.	none	

Parameter	Type	Definition	Default	Units
COLLECT	char	The electrode name associated with the collector for a bipolar device.	none	
EXTRA	char	The electrode name associated with the an arbitrary contact.	none	

Description

The **LOG** statement allows the I-V and/or AC terminal data for a simulation to be written to a file. This statement also provides for the conversion of Medici I-V log files to data files that can be used by common parameter extraction programs. This section describes the **LOG** statement as it functions with the following:

- Medici log files
- Aurora data files
- IC-CAP data files

See Also…   To further illustrate the **LOG** statement, refer to:

- Input file *mdex1g* in N-Channel MOSFET Examples, Chapter 4, "Simulation of Gate Characteristics" on page 4-11
- Input file *mdex1d* in N-Channel MOSFET Examples, Chapter 4, "Simulation of Drain Characteristics" on page 4-12
- Most other examples when I-V characteristics are saved for later plotting.

Medici Log Files

A **LOG** statement causes I-V or AC data obtained as a result of all subsequent **SOLVE** statements to be saved. When a **LOG** statement is encountered, any open log file is closed and a new file, as specified by **OUT.FILE**, is opened.

Data stored in log files is used by **PLOT.1D** statements to generate one-dimensional parametric plots. Data is generated for log files whenever a **SOLVE** statement is used to create steady state, transient, or AC solutions. The data stored in a log file can consist of the following:

- Frequency
- Terminal currents
- AC capacitances
- Optical data
- Hot electron injection and tunneling currents associated with each electrode

- User-defined parameters
- Transient simulation times
- Applied voltages
- AC conductances

- Contact voltages
- Electrode charge
- Impact ionization current
- AC admittances

Aurora Data Files

The **LOG** statement can also be used to create a data file that can be read directly by *Avant!* TCAD's parameter extraction program Aurora. To accomplish this:

- The **AURORA** parameter should be specified
- A previously created Medici log file containing the data of interest should be specified with the **IN.FILE** parameter
- The **OUT.FILE** parameter is used to specify the identifier for the output data file for Aurora.
- The parameters **GATE**, **DRAIN**, etc., identify the electrodes for the I-V data in **IN.FILE** that correspond to the gate contact, drain contact, etc., of the simulated structure.

For each such parameter that is specified, the program writes both the applied voltage and terminal current corresponding to this contact to the data file created for Aurora.

Note:

At the present time, the parameters **OUT1**, **OUT2**, **OUT3**, **INP1**, **INP2**, **INP3**, *and* **INP4** *are not applicable to data files written for* Aurora.

IC-CAP Data Files

The **LOG** statement can also be used to create a data file that can be read directly by Hewlett-Packard's IC-CAP program. To accomplish this:

- The **ICCAP** parameter should be specified.
- A previously created Medici I-V log file containing the data of interest should be specified with the **IN.FILE** parameter.
- The **OUT.FILE** parameter specifies the identifier for the output data file for IC-CAP.

Variables Up to three output variables and four input variables can be specified using the parameters **OUT1**, **OUT2**, **OUT3**, **INP1**, **INP2**, **INP3**, and **INP4**. The quantities that can be specified for these parameters are any of the available voltages and currents contained in the Medici I-V log file.

For the input variables, **INP1** should be used to specify the most rapidly varying quantity, **INP2** should be used to specify the next most rapidly varying quantity, and so on. For input variables that are constant, the order is unimportant as long as they are specified after quantities that vary.

Parameters The parameters **GATE**, **DRAIN**, etc., identify the electrode names for the I-V data in **IN.FILE** that correspond to the gate contact, drain contact, etc., of the simulated structure. If these parameters are specified, then appropriate mnemonics are

substituted for the electrode numbers when the input and output variables are written to the IC-CAP data file.

IC-CAP Example

Consider a case where Medici is used to simulate the gate characteristics for an MOS device (*ID* vs. *VG*) for several substrate biases (*VB*). Assume that in Medici, electrode names Drain, Gate, Src and Subst correspond to the drain, gate, source, and substrate contacts and that the I-V data is stored in the file *medici.log*.

The following statement can be used to create a data file for IC-CAP:

```
LOG     ICCAP   IN.FILE=medici.log   OUT.FILE=iccap.set
+       DRAIN=Drain   GATE=Gate    SOURCE=Src    SUBSTRAT=Subst
+       OUT1=I(Drain)   INP1=V(Gate)   INP2=V(Subst)
+       INP3=V(Drain)   INP4=V(Src)
```

where:

- Drain current is identified as the output variable with the parameter **OUT1**.
- Gate voltage is identified as the most rapidly varying input variable with the parameter **INP1**.
- The parameter **INP2** identifies the substrate voltage as the next most rapidly varying input variable.
- Both the drain voltage and the source voltage are taken as constant and are identified with the input parameters **INP3** and **INP4**.
- The parameters **DRAIN**, **GATE**, **SOURCE**, and **SUBSTRAT** cause the name *ID* to be used for the output variable and the names *VG*, *VB*, *VD*, and *VS* to be used for the four input variables in the IC-CAP data file.

Syntax and Format

To be compatible with IC-CAP nomenclature, the data file for IC-CAP should end with the suffix *.set* (for example, *iccap.set* in the above example).