POSTMINI User's Manual
Version 9.2-000

John Faricelli

Alpha Semiconductor Technology

Compaq Computer Corporation

Contents

1 Introduction

POSTMINI is an interactive graphical postprocessor for device and process simulators. The postprocessor reads the save files from a number of simulation programs (e.g. MINIMOS, PISCES, SUPREM4 etc.) and allows the user to examine or plot quantities stored in the file. In addition, POSTMINI can import data from ASCII files, and can be used as a general plotting program. Graphical output under Tru64 Unix or OpenVMS is available on workstations running X/Motif, PostScript printers (monochrome, color and encapsulated forms), HP7550 (HP-GL) plotters, HP PCL, and all VT series graphics terminals (e.g. VT340). Under Win32, hardcopy output is limited to screen dumps of windows. The POSTMINI command language can be used to re-create any plot from commands in a file.

2 Functionality

The available functions in POSTMINI are:

- 1D - Plot an X-Y graph of 1D data, or plot a 1D cross-section of 2D data along any vertical or horizontal line.
- 2D - Plot 2D contours of a quantity.
- 3D - Plot a quantity as a surface in 3D. These kind of plots are also known as “bird’s eye” plots.
- Compare - Plot several curves on the same graph (can be from the same or different data files). Can also plot bar charts.
- Find - Find where an internal quantity reaches a specified value.
- Integrate - Integrate an internal quantity in a region or along a line.
- Line - Print a cross-section of a 2D data along any vertical or horizontal cut line into a file.
- Minmax - Determine the minimum/maximum of an internal quantity.
- Print - Print 2D data in a formatted report into a file.
- Read - Read in another data file.
- Show - List information about the simulation run (e.g. terminal voltages and currents). Currently only for MINIMOS.

- Default - Change default plot attributes of POSTMINI
- Restore - Restore a plot from a Postmini command file
- Shell - Execute operating commands without leaving POSTMINI.
- Save - Save the current plot in a Postmini command file
- Window - Manage multiple plot windows on workstations.
- Exit, Quit - Terminate POSTMINI.

The quantities that can be printed/plotted depend on the simulation program. For MINIMOS, they are:

- Potential
- Doping concentration
- Electron/hole concentration
- Electric field (x,y components, magnitude)
- Electron/hole current density (x,y components, magnitude)
- Generation rate
- Electron/hole quasi-Fermi levels
- Electron/hole mobility
- Electron/hole carrier temperature
- Interface charge (fixed and chargeable)

3 How to run POSTMINI on VMS

Place the Postmini program in any directory on your machine, for example `DISK:[TCAD.BIN]`. To run POSTMINI, type:

```
$ POSTMINI :== $DISK:[TCAD.BIN]POSTMINI.EXE
$ POSTMINI [ options ] [ datafile ] [ filetype ]
```

where `datafile` is a datafile (e.g. MINIMOS binary file) and `filetype` is the type of datafile being read (e.g. MINIMOS, PROMIS, etc.). If the datafile has an extension that POSTMINI recognizes (e.g. `.bin` for MINIMOS, `.pmi` for POSTMINI command file), then the filetype is optional. For a complete list of datafiles and command line options, see Appendix F. Note: since POSTMINI uses the Digital GKS graphics package, you must have Digital GKS installed on your system. It is suggested that you have Digital GKS V6.0 or higher installed.

By default, POSTMINI assumes that you will be displaying your graphics on a workstation running Motif. You can specify a different plot output device by using the `/device` option on the POSTMINI command line. The following devices are supported:

Option name	Output device
`MOTIF`	Workstation running Motif
`VT240`	Digital color VT240
`VT240BW`	Digital B&W VT240
`VT330`	Digital VT330
`VT340`	Digital VT340
`XTERM`	Xterm using Tek4014 emulation
`integer`	GKS workstation type (e.g. 231)

Example:

```
$ postmini /device=vt340
```

If you are running Motif, you must also have the logical name `DECW$DISPLAY` defined to point to your X display device. This is normally done automatically for you by the operating system.

Finally, when you first make a plot with POSTMINI, you may get an error message similar to:

```
WARNING - 'TITLE SIZE' attribute incorrect: use
"DEFINE GKS$DECW_TITLE_SIZE 20" and restart application.
```

The message may come up quickly and get overwritten by later output by POSTMINI. POSTMINI should open up a window about 1/4 of the screen, in the upper right hand corner. If you do not get this behavior, it is likely that the above logical name has to be set. Unfortunately, the value you need to use depends on the particulars of your workstation (e.g. screen resolution). Also, if you are running Motif V1.2, and you continue to see these messages, even though you have set the logicals, see the comments in the "Things to look out for" section.

NOTE! POSTMINI V7.3 and higher no longer examines the logical name `GKS$WS_TYPE` to determine which output device to use. POSTMINI also tries to work correctly if the logical name `GKS$CONID` is not defined.

4 Operation under Tru64 Unix (Digital Unix)

POSTMINI runs under the Tru64 Unix operating systems. POSTMINI operation is essentially the same as under VMS. You must have Digital GKS V5.2 or higher installed on your system. Binary data files must have been created under Tru64 Unix, as the format of binary files differs between VMS and (Tru64 Unix). If you need to read VMS binary files on (Tru64 Unix) contact the support person, as this is possible to do with a special version of Postmini.

The following implementation differences are noted:

- Place the postmini executable into a directory contained in your `PATH` shell variable. Alternately, C-shell and Korn-shell users can define an alias for postmini as follows:

```
alias postmini /directory/where/postmini/is/postmini
```

- The shell command creates a process running the C-shell.
- Since (Tru64 Unix) do not support file versions, the plot file naming scheme is slightly different. If a new plot file would overwrite an existing file, the new plot file name is modified to be of the form "`file.n.extension`" where "`n`" is an integer. For example: `nmos.2.ps`.
- Environment variables are used instead of logical names. The following environment variables are used by Postmini; remember (Tru64 Unix) are case sensitive.

Environment variable	Function
`DISPLAY`	Default X window display

C-shell example:

```
% setenv DISPLAY  asda02:0
```

5 Operation under Win32 (Windows 95/98/NT)

POSTMINI runs under the Windows 95/98/NT operating systems. POSTMINI operation is essentially the same as under OpenVMS or Tru64 Unix.

The following implementation differences are noted:

- Hardcopy is limited to screen dumps of the plot window, due to the limitations of the graphics library used.
- The shell environment symbol `Windows-NT` is used to determine whether Postmini is running on Windows-NT. It used is mainly to determine which executable to use to start up a new shell.
- Since win32 does not support file versions, the plot file naming scheme is slightly different. If a new plot file would overwrite an existing file, the new plot file name is modified to be of the form "`file.n.extension`" where "`n`" is an integer. For example: `nmos.2.ps`.

6 Supported datafile formats

POSTMINI can read the output file formats for a number of simulators: DEC SPICE output files (via the ASCII file reader), HSPICE output files (via the ASCII file reader), SPICE3 "raw" files, MINIMOS (binary internals `.BIN`, binary doping `.DOP, .2DOP`, binary dump format `.USEOUT`), SUPREM3 structure files, SUPREM4 ASCII structure files, PISCES-IIB mesh and solution file (Stanford version 9009), TMA TIF (TMA Interchange Format) files, Vector Technologies FAIM and MCP2D files, and PROMIS (binary `.SAVE`) files.

POSTMINI can read binary files from either 2D or 3D MINIMOS runs. If you are plotting from a 3D binary file, you will be requested for a "cut plane" which will "slice" the device along the length, width or wafer (top view) direction. POSTMINI supports 3D MINIMOS doping files in a similar manner.

In the case of DSUPREM3, only 1D and comparison plots are possible, since DSUPREM3 is only a 1D simulator. POSTMINI also shifts the DSUPREM3 coordinate system such that the zero coordinate occurs at the silicon surface. This is to be consistent with other simulators. This also allows quantites such as junction depths to be directly read off plots.

You can also import data from an external ASCII file. The data should be arranged in columns. Any line which begins with a "C", "c", "!" or "#"is taken as a comment line. You can also use `.OFF` and `.ON` to exclude portions of a file from being read. If you do not have enough data in a line to satisfy a read request,

or have non-numeric data on a line, that line is silently ignored. You can specify which columns are to be read, as well as data scaling factors `mx, my` and `ax, ay`. These scaling factors transform the data as follows:

$$\text{scaled data} = \text{mx} \cdot (\text{input data}) + \text{ax} \tag{1}$$

You can also specify how many lines to skip, and how many lines to read in a file. If you have a data file which contains several data sets appended together, you can read all the data in one step by setting the option `Multiple curves in datafile` to `yes`. This will suppress the retrace when the X axis value goes "backwards." This is useful for plotting current-voltage or capacitance-voltage data files. You can also set the "sampling" frequency, which will cause Postmini to use every second, third, fourth, etc. point in a file. This is useful for reducing the size of large, closely spaced, data sets, especially if symbols are also plotted. Postmini also recognizes SPICE output files and has special support to skip automatically to the simulation results output via `.PRINT` statements.

Postmini can handle ASCII files with up to 50000 data points each (version 9.0 and higher). If you wish to plot different columns of data as one curve, you can specify more than Y column (version 8.2 and higher) or `ALL` to select all remaining columns.

In version 8.1 and higher, Postmini can import ASCII files which have X-Y-Z data. The X-Y-Z format is simple way to import 2D data from an arbitrary simulator. Postmini normally expects the the data to lie on a rectangular grid. The file format is similar to the X-Y ASCII format. You may use the same commenting conventions, along with scaling factors, including scaling the Z axis. The file format is:

```
x1  y1  z
x2  y1  z
x3  y1  z
.
.
xn  y1  z
x1  y2  z
x2  y2  z
x3  y2  z
.
.
xn  y2  z
```

.
.
.

where the x's are the X coordinate, the y's are the Y coordinate, and the z's are the function value f(x,y). The X and Y coordinates must be unique and must be specified in increasing order. In version 8.2 and higher, the ASCII file reader has been extended to read data that is not on a rectangular grid. A triangular grid is imposed on the data.

In version 8.3 and higher, Postmini can apply general expressions to ASCII data. For example, given a file with quantities `C` and `E`, one could plot `C**2` vs. `1/E`.

In version 8.3 and higher, Postmini can also process ASCII files that have character string labels in the X column. Character strings are delimited by blanks, tabs, equal signs or commas. Mathmode strings may be used. To embed a blank, use the mathmode escape
`;`. This type of data is used to place labels under each bar on the X axis on bar charts. Up to 200 labels may be input. The user must tell Postmini to expect X column character data via the datafile type option in the ASCII file reader menu.

7 Using POSTMINI for visualization

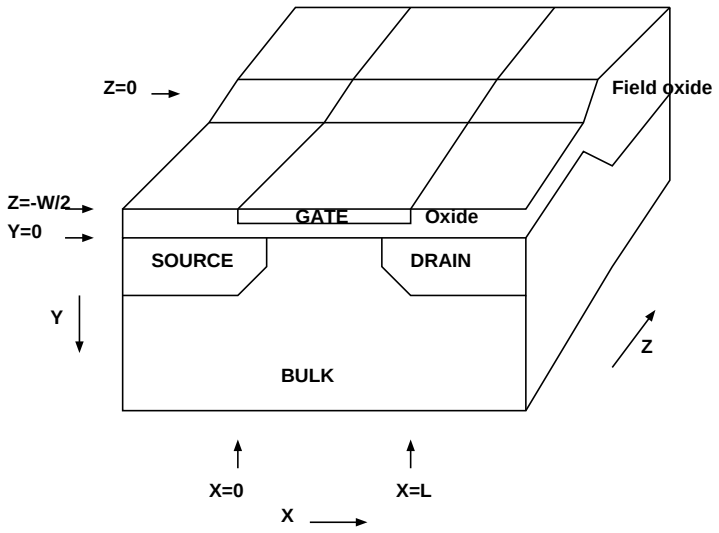
After you have read in a file, you can examine the data with a number of visualization techniques. POSTMINI allows the user to visualize his/her simulation results with contour maps (both filled color and tradition level curves), quasi-3D plots, plots at various cross-sections in the structure, etc. After selecting a menu item, the program will display the data that can be examined (e.g. potential, doping). Select the quantity to be plotted, or enter `999`, `Q` or `<return>` to exit this menu.

You will then be placed into a full menu screen that will allow you to make a plot on your graphics device, annotate the plot, make a hardcopy, or alter the plot characteristics, such as the horizontal axis limits. To alter a quantity, enter the number next to the quantity to be modified. You will then be prompted for the quantity. If, at this point, you decide that you don't want to change the quantity, enter an end-of-file (control-Z on VMS, control-D on Tru64 Unix). If you enter an invalid response, an error message will appear under the "Messages:" line. If the screen becomes corrupted by a mail message notification or other system message, enter `R` to repaint the menu on the screen.

7.1 MINIMOS coordinate system and terminology

The following remarks are specific to using POSTMINI when examining MINIMOS output files. The x coordinate goes along the length direction of the FET, from source to drain. X=0 occurs at the source end of the gate edge. The y coordinate goes along the depth direction of the FET, with y increasing with increasing depth. Y=0 is at the oxide/silicon interface. Thus, negative y coordinates are in the oxide; positive y coordinates are in the silicon. For binary files from 3D MINIMOS runs, you have a third (width) dimension. In the width (z) dimension, the device extends from the middle of the channel width ($-W/2$) towards the field oxide (positive). Z=0 occurs at the thin oxide mask edge in the width direction.

The following diagram illustrates the MINIMOS coordinate system.



7.2 1D plots

POSTMINI provides a quick way to do an X-Y plot, or plot a quantity along either a vertical or horizontal line. POSTMINI provides the following default plot scales, which the user can override:

- The default limits for the abscissa (horizontal) axis is the entire device width/length/depth.
- The default limits for the ordinate (vertical) axis is the entire data range, rounded to "nice" numbers. Data outside the range 0.001 to 1000.0 will be scaled by a power of ten.

You have the option of changing the default limits to plot a portion of the data or range.

On 1D and comparison plots, you can modify the "scale factor" or remote exponent for the plot (Note, as of Postmini V7.2, you can only modify the scale factor for the ordinate). The scale factor is an integer power of ten that will be used as the remote exponent in the graph. The plot data will then be scaled according to the following relation

$$\text{scaled\_data} = \text{actual\_data}/10^{\text{scale\_factor}}$$

A suggested scale factor will be displayed, this usually results in the best plotting. A zero scale factor means no scaling will take place before plotting. Example: to plot data with a range of 2.2×10^3 to 3.5×10^3, you might select a data range of 2.0 to 4.0, with a scale factor of 3 (this would have been the default!).

Axis tic marks are also provided. You can specify the major tic mark increment (real) and the frequency of minor ticmarks (integer). A major ticmark is a long tic mark which has a number next to it. Example: a major tic mark increment of 0.1 with a minor tic frequency of 5 would plot major tics every 0.1 units, and plot a minor tic every 0.02 units (thus dividing the major tic interval in five).

The annotation option can be used to interactively add text, lines, arrows, boxes, symbols and elliptical arcs to the plot. Text is processed by the "math-mode" utility, which allows you to enter sub and/or superscripts, Greek letters, and certain math symbols using a subset of the TEX math syntax. Read section 13 for details on how to enter mathmode format strings.

7.3 2D contour plots

POSTMINI can plot a contour map (level curves) of simulation quantities over all or a portion of the device. The user is asked to specify the portion of the device to plot, and the contour values to plot. POSTMINI can plot up to 9 contours on one graph. There are several different ways to specify contour values:

- You can specify the contours by individual value, e.g.

  ```
  1.0 2.0 3.0 4.0 5.0
  ```

- To specify a number of contours equally spaced between a min and max range, use the `e` notation:

  ```
  e number-of-contours  minimum  maximum
  ```

For example, to specify 5 contours between 1 and 5:

```
e 5 1 5
```

This would result in contours at 1, 2, 3, 4 and 5.

- To specify contours with a given step value, use the `s` notation:

  ```
  s step-value  minimum  maximum
  ```

  For example, to specify contours between 1 and 5 with a step of 1:

  ```
  s 1 1 5
  ```

  This would result in contours at 1, 2, 3, 4 and 5.

- To specify a number of contours with logarithmic spacing, use the `l` notation:

  ```
  l number-of-contours  minimum  maximum
  ```

  For example, to specify 5 contours between 10^{14} and 10^{18} in logarithmic steps:

  ```
  l 5 1E14 1E18
  ```

  This would result in contours at 10^{14}, 10^{15}, 10^{16}, 10^{17}, and 10^{18}.

- To specify contours with a logarithmic step, use the `d` notation:

  ```
  d step minimum  maximum
  ```

  For example, to specify contours between 10^{4} and 10^{6} with a log step of 10:

  ```
  d 10 1E4 1E6
  ```

  This would result in contours at 10^{4}, 10^{5}, 10^{6}. When using log steps, the minimum and maximum have the same sign.

You can combine any or all of these notations when specifying contour values, but you must not exceed the total number of contours.

On devices supporting many colors such as workstations, the default is to plot contours with color fill between the contours. A different color is used to denote different data values. This makes the contour map easier to interpret. A legend on the side of the plot tells what values the colors represent. The first color represents all data below the first contour value (note the "<" before the printed value), the second color represents all values between the first and second contour value, etc. The last color represents all values above the maximum of the data (note the "<").

You may also specify tranditional contour plots using lines. Different contours may be distinguished by different colors or by different line styles (solid, dashed, dotted, etc.), depending on the output device. A key to the contour values is plotted on the right side of the plot.

You can optionally plot the location of all the junctions in the structure. If you select this option, a dashed line will be plotted at the approximate junction location. If you request, the plot software will place small labels on each level curve so they can be distinguished.

On workstations, you can use the mouse to intteractively zoom in on a portion of the plot. Enter the `Z` (zoom) command at the menu prompt. A crosshair cursor will appear in the plot window. Click and release the first mouse button to enter the new lower left corner of the plot. The cursor will change to a "stretchy" box. Click the first mouse button again to enter the new upper right corner of the plot. If you click outside the plot area, the coordinate of the closest corner is used. Press the second mouse button to cancel input. To return to the full coordinates, enter the `U` (unzoom) command.

On workstations, you can use the mouse to interactively take a "sample" of the contour data, and display the data value and coordinates. Enter the `S` (sample) command a the menu prompt. A crosshair cursor will appear in the plot window. Click and release the first mouse button to take a sample. The data value at that point, plus its coordinates, will be displayed on the menu. You can continue to click the mouse button to sample other areas of the plot. Press mouse button two to exit sample mode.

If your data is better represented by discrete values, rather than a continuous function, you can switch Postmini into a mode where it plots a box around each data point in a different color, rather than interpolating a surface. Go to the top level "Defaults" menu, and change the "Default hidden line method" to histogram. This will also change 3D surface plots, so that Postmini plots a 3D histogram around each data point, rather than interpolating a surface.

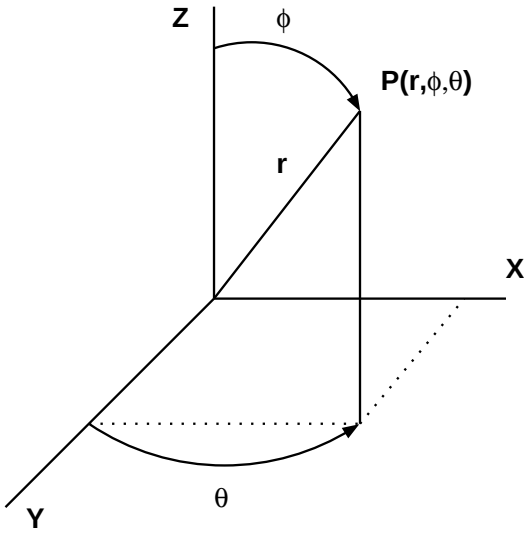
The annotation option can be used to interactively add text, lines, arrows, boxes, symbols and elliptical arcs to the plot. Text is processed by the "math-mode" utility, which allows you to enter sub and/or superscripts, Greek letters, and certain math symbols using a subset of the TEX math syntax. Read section 13 for details on how to enter mathmode format strings.

7.4 3D surface plots

POSTMINI can plot an internal quantity as a surface in 3D perspective. The user is asked to specify the portion of the device to plot; this can be used to "zoom" in on a region of detail in the device.

You can also move the viewing position of your "eye" relative to the 3D plot. POSTMINI uses a polar coordinate system to specify the eye position. Position is specified with three numbers: a radius r in microns, polar angle ϕ, in degrees, and azimuth angle θ, in degrees. Increasing θ rotates the eye counter-clockwise around the "equator." Values for θ range from 0^o to 360^o. You may also specify a negative angle; it is converted to the corresponding positive angle. Increasing ϕ moves the eye down from the "north" pole to the "south" pole. Values for ϕ range from 0^o to 180^o. The default eye position is $r = 8.0$, $\phi = 45^o$, and $\theta = -45^o$.

The following diagram illustrates the polar coordinate system. You may notice that this is a left-handed coordinate system. A left-handed system is used to be consistent with the way 3D plots are displayed by other workers. In the default view, the source is on the left, and the drain on the right. Note that this is *opposite* from the way the SURF program displays MINIMOS data.



On devices that support color, the plot will be done using color area fill technique. This method plots the surface in different colors, depending on the "height" of the data. This makes is very simple to determine the actual value of data being plotted. A legend is printed on the side of the plot, similar to the 2D color contour map. On all other devices, the surface will be plotted using a hidden line removal technique. On color devices, the "underside" of the plot will be done in a different color.

On workstations, you can use the mouse to intteractively rotate the 3D plot. Enter the 0 (orbit) command at the menu prompt. A box around the 3D plot will appear in the plot window. Click and release the first mouse button at the right edge of the window to rotate the plot clockwise. Click and release the first mouse button at the left edge of the window to rotate the plot counter-clockwise. Click and release the first mouse button at the top edge of the window to move your eye position towards the "north pole". Click and release the first mouse button at the bottom edge of the window to move your eye position towards the "south pole".

If your data is better represented by discrete values, rather than a continuous function, you can switch Postmini into a mode where it plots a 3D histogram, rather than a surface. Go to the top level "Defaults" menu, and change the "Default hidden line method" to histogram. This will also change contour plots, so that Postmini plots boxes centered on each data point, rather than interpolating a surface.

The annotation option can be used to interactively add text, lines, arrows, boxes, symbols and elliptical arcs to the plot. Text is processed by the "math-

mode" utility, which allows you to enter sub and/or superscripts, Greek letters, and certain math symbols using a subset of the TEX math syntax. Read section 13 for details on how to enter mathmode format strings.

7.5 Comparison plots

POSTMINI can plot several 1D cross-sections on the same graph. The 1D slices can be of the same internal quantity, different quantites, or from different simulation runs. You can also import data files from different simulators, including 1D ASCII files, for plotting.

To use this feature, select COMPARE at the main menu. You will be put into a subsystem which has these features:

- Add - Add a curve to the plot list. You will be prompted for the quantity to plot, the cut line and coordinate (for 2D data sets). You can also specify plot attributes such as line type (e.g. solid, dashed), color, and plotting of symbols. A total of forty separate curves can be plotted at once.

  Postmini version 8.1 and higher supports multiple X and Y axes (earlier versions only supported separate axes from Postmini command files). You can associate a data curve with either the top or bottom X axis, and the left or right Y axis. Use the `xaxis_tag` and `yaxis_tag` menu items.

- Barchart - Plot all the data as a bar chart.

- Clear - Delete all curves from the plot list. Useful for "starting over".

- Delete - Delete a curve from the plot list.

- Integrate - Integrate a quantity vertically or horizontally, and add to plot list. This option applies to 2D data only.

- List - List all curves to be plotted, with their labels.

- Modify - Modify various aspects of a curve, such as the plot label, curve color, linetype, or symbol. You can also modify the scale factors (AX, AY, MX, MY) that were applied the data, or apply an expression to the data. You can also select whether a curve will be included in the plot. This is useful if you wish to store a number of data items at one time, but selectively plot the entries. You can also change the axes the curve will be plotted on. Postmini supports a top and bottom X axis, and a left and right Y axis.

- Plot - Plot all the data as an X–Y graph.

- Read - Read a new data file.
- Show - List information about a curve
- Exit, quit - Return to main menu.

On workstations, you can use the mouse to interactively take a "sample" of the plot, and display the X-Y coordinates. Enter the S (sample) command at the menu prompt. A crosshair cursor will appear in the plot window. Click and release the first mouse button to take a sample. The data value coordinates will be displayed on the menu. You can continue to click the mouse button to sample other areas of the plot — the change from the last sample point is also printed. Press mouse button two to exit sample mode.

On workstations, you can use the mouse to interactively "zoom" into a region of the plot. Enter the Z (zoom) command at the menu prompt. A crosshair cursor will appear in the plot window. Click and release the first mouse button to define the lower left of the zoom area. The cursor will now change to a "strechy" box, which defines the area to be zoomed. Click and release the first mouse button to perform the zoom. Press mouse button two to cancel the zoom. The unzoom function U will rescan all the curves and pick bounds which will enclose all the data.

The annotation option can be used to interactively add text, lines, arrows, boxes, symbols and elliptical arcs to the plot. Text is processed by the "math-mode" utility, which allows you to enter sub and/or superscripts, Greek letters, and certain math symbols using a subset of the TEX math syntax. Read section 13 for details on how to enter mathmode format strings.

7.6 Overlay plots

An overlay plot consists of multiple 2D contour plots in the same window. Plots can overlay each other, such as line contours of electron concentration over color contours of doping concentration, or they can be spaced apart by scaling the x and y coordinates.

To use this feature, select OVERLAY at the main menu. You will be put into a subsystem which has these features:

- Add - Add a contour plot to the plot list. You will be prompted for the quantity to plot, if the dataset contains multiple quantities. You can also specify plot attributes such as contour values, grid, junctions, etc. You can also apply scaling or expressions to the x and y coordinates and the data itself. A total of forty separate contour plots can be plotted at once.

- Clear - Delete all data from the plot list. Useful for “starting over”.
- Delete - Delete a dataset from the plot list.
- List - List all data to be plotted.
- Modify - Modify various aspects of a dataset, such as contour values, grid, junctions, etc. You can also apply scaling or expressions to the x and y coordinates and the data itself.

  You can also select whether a dataset will be included in the plot. This is useful if you wish to store a number of data items at one time, but selectively plot the entries.
- Plot - Plot all the data as an contour graph. The plot menu allows the user to set certain global plot attributes, such as applying a global set of contours to be used with all datasets, and setting the dataset that will be used for the sample function and also for plotting the contour map legend at the side of the plot. The other functions available for contour plots (e.g. zoom, unzoom, sample, save) are available for overlay plots.
- Read - Read a new data file.
- Show - List information about a dataset
- Exit, quit - Return to main menu.

7.7 The FIND function

The FIND function will search along a 1D cut line in the device and print out where a simulation quantity reaches a certain value. For example, you can use FIND to determine where the lateral electric field is zero along the surface of the MOSFET (Y=0).

7.8 The INTEGRATE function

The INTEGRATE function will integrate a quantity in an arbitrary rectangular region in the device, or along a 1D cut line in the X, Y or Z direction. INTEGRATE prompts for the region to integrate, and will check to make sure the region is inside the device. INTEGRATE prints out certain landmarks in the device, such as the source and drain contact positions, and the gate position, for the user’s convenience in specifing the region. Note that the integration area is rounded to the

nearest simulation mesh line, so integration over exact regions is not possible in general. In most cases, there are sufficient mesh lines to resolve the use specified integration region. As a check, POSTMINI will print out the actual integration region for the user to check. Also note that the resulting integrated value is given per centimeter (cm) device width (or length/depth, if 3D data is being integrated).

Integration over a line can be thought of as integrating over a plane that is inserted perpendicularly to the simulated device (in the z, or width direction, for example). When performing a line integration of a vector quantity, such as current density or electric field, the vector component perpendicular to the cut line is usually chosen. For example, if the cut line is in the vertical (Y) direction, the integral of the x component of the current density gives the amount of current flowing though across that line. One should be cautious when integrating the y component of the electric field at the oxide interface since field is discontinuous at that point. POSTMINI computes the electric field at the semiconductor side of the interface and uses the result at that point.

POSTMINI also allows you to perform a 1D integration of any 1D data that has been loaded, e.g. from an ASCII file, SUPREM3, etc.

Note well: integrating over an area with the same X or Y coordinate is *not* the same as integrating over a line. Quantities integrated over a region are weighted by area (cm^2), while those integrated over a line are weighted by length (cm). POSTMINI prints the units of the integrated quantity to remind the user.

Note well: the integration method does not take nonplanar boundaries correctly into account (uses entire area weight, rather than area in just silicon). Use caution when integrating near nonplanar features.

Example 1: Integrate the avalanche generation over the entire device. Multiply this number by the electronic charge q to get the bulk current per cm of device width. Note: this number can be different from what MINIMOS prints, especially when the bulk current is very small compared to the drain current.

Example 2: Integrate the X component of the minority current in a vertical direction in the middle of the channel. This gives the drain current per cm width of the device. Note, if MODEL=AVAL or HOT, there may be a slight difference between this current and the drain current printed by MINIMOS, due to additional current generated by impact ionization. By varing the length of the integration line, one can determine the amount of current that passes a certain depth.

7.9 The LINE function

Using the LINE function, you can print simulation internal quantities along any vertical or horizontal line. The first lines of the file are a header which lists the

quantity and cut line selected. This header is commented out using the CURV program convention, so these files can be directly read by CURV. Following the header, the data is printed out in two columns: the coordinate (in microns) and the quantity at that coordinate.

When you print data into a file, POSTMINI creates a file in your current directory, using the following algorithm to generate the file name:

1. The stem of the binary file name is the first part of the print file. Example: binary file is GEN505000.BIN; stem is GEN505000.

2. The coordinate of the cut is appended to the file name in the format _nnnnC, where nnnn is the absolute value of the cut coordinate value in 1000'ths of microns, and C is the cut direction (X or Y). Examples: cut at X = 1.25 microns $\rightarrow$ _1250X; cut at Y = -0.050 microns $\rightarrow$ _0050Y.

3. The file type is a mnemonic code which depends on the quantity selected. See section 11dix B for the list of extensions and their meaning.

7.10 The MINMAX function

The MINMAX function allows you to locate the minimum and maximum of a quantity in the simulation. Note that it is possible that there will be several places in the device which have the same minimum or maximum value. POSTMINI will report the first one it finds as it scans the device. You can limit the seach to a portion of the device; this is useful if you wish to limit seaching to the semiconductor region (e.g. $y \geq 0$ in a planar MOSFET).

7.11 The PRINT function

Using the PRINT function, you can print simulation internal quantities over the entire device to a file. Output consists of a neatly printed array of data values, with corresponding x and y coordinates.

When you print data into a file, POSTMINI creates a file in your current directory, using the following algorithm to generate the file name:

1. The stem of the binary file name is the first part of the print file. Example: binary file is GEN505000.BIN; stem is GEN505000.

2. The file type is a mnemonic code which depends on the quantity selected. See section 11 for the list of extensions and their meaning.

7.12 Changing POSTMINI defaults

POSTMINI allows you to change a number of default plot attributes. Use the `default` function to modify them. The following attributes can be modified from within Postmini:

Attribute	Possible values
Force solid lines on contour plots	Yes/No
Linewidth scale factor	0.0 – 5.0
Hidden line algorithm	Device dependent Horizon function Painter's algorithm
Default hardcopy device	(See list of supported devices)
Workstation window size factor	0.1 >= factor >= 1.0 the factor is a percent of the full workstation screen
Hardcopy reduction factor	0.1 > factor > 1.0
Hardcopy orientation	Landscape/Portrait
Text scale factor	> 1.0 enlarges all text < 1.0 shrinks
Force autoscale of coordinate axes	Yes/No
Length unit name	Arbitrary string

7.13 The shell function

You can execute operating system commands without leaving POSTMINI by using the shell function. Under VMS, your DCL prompt will be set to `Postmini>`, to remind you that you must return to POSTMINI (type LOGOUT to do this). Under Tru64 Unix, a C-shell process is created. Type "exit" or "control-D" to return to POSTMINI.

7.14 Managing windows on workstations

When POSTMINI is run on a workstation, it can plot data in multiple windows. Select the "windows" option from the main menu. From there, you can opt to delete any or all plot windows on the screen, or select the "style" of window management. The three available styles are:

- Single window - All plots go to a single window.

- Multiple - Each plot type (e.g. 1D, contour, 3D, comparison) uses a separate window.
- Reactivate - Reactivate plot windows that have been “saved” on the screen.
- Retained - All plots go to a single window; however, you can elect to save the current window on screen. POSTMINI will prompt you when the plot is completed if you want to save the window. POSTMINI will open a new window for the next plot. Retained windows remain on the screen until you delete them. You may keep up to 10 windows on the screen at once.

The default style is “retained”.

If you use several windows, you will probably want to rearrange or shrink/expand the windows to your liking. Use the normal Motif window manager functions to manipulate the GKS plot window. Under Motif, when a window is re-sized, the underlying graphics are redrawn to fit in the window, while maintaining the original aspect ratio of the plot.

Under Motif, you may notice that the window “input focus” is transferred to the GKS window whenever there in any input or output to that window. This is inconvenient when using Postmini, since you need to click on your terminal window to restore input focus so you can type at the terminal window. This also has the side effect of popping the terminal window up so that it covers the graphics output. This is especially annoying when performing graph annotations. One way to avoid this problem is to change the Motif window manager input focus policy so that input focus is assigned to the window where the mouse is pointing. To do this, click and hold down mouse button 1 on any part of the background window. A menu should come up. Point to “Options $\rightarrow$” and select the “Workspace” submenu. Under “To make a window active”, select the “Move the pointer into the window” option. Also, turn off the option “Raise it to the top of the screen” under “When a window becomes active.” Then save the settings and restart the Motif window manager.

If you do not like the “focus follows mouse” approach, you can just turn off the option “Raise it to the top of the screen” and restart the Motif window manager. When the Postmini graphic window comes up, focus will be on the graphics window, but you can move it back to the terminal string by clicking anywhere on the terminal window. Click on the window border to raise the window to the foreground.

7.15 The save/restore functions

Postmini supports the option to save a description of the current plot in a Postmini command file. The command file is ASCII text, and can be edited with any text editor. It contains all the information necessary to re-plot your graphics. The command file consists of several sections:

- PLOT – The PLOT command describes the type of plot – e.g. 1D, 2D, 3D or COMPARE.
- GLOBAL – The GLOBAL command and its subsequent lines set overall factors, such as defining colors
- AXIS – The AXIS command defines the limits on the various axis parameters, such as minimum, maximum, tic marks, etc.
- CURVE – The CURVE command defines the how to restore an individual curve of a 1d or comparison plot. It contains datafile information (filename, data type, cut line coordinate, etc.) as well as plotting information (color/marker information).
- CONTOUR – The CONTOUR command defines how to restore a contour plot. It contains datafile information (filename, data type, etc) as well as contour values used, and other plot options.
- SURFACE – The SURFACE command defines how to restore a surface plot. It contains datafile information (filename, data type, etc) as well as other options, such as eye position.
- ANNOTATE – The ANNOTATE command defines the various annotations of the plot – boxes, lines/arrows, text, markers and ellipses.

In general, you would generate a POSTMINI command file using the save function. It is entirely possible to generate one yourself, although it would probably be better to use a POSTMINI generated file as a template. The restore function attempts to catch invalid input as best as possible, and reports syntax errors to the terminal by line number of the POSTMINI command file. Section sec:pmifile lists the current set of command file keywords.

Once you restore a plot, you will be placed into an appropriate full screen, e.g. if you restored a 2D contour plot, you will be placed in the 2D contour plot menu. You can then plot, annotate, make a hardcopy, etc.

Note that POSTMINI when restores a plot, it re-reads the original datafiles. If the datafiles have been deleted, modified or are in another directory, then POSTMINI may not be able to restore the plot, or a unexpected graph may be displayed.

POSTMINI automatically saves your last plot in a file called `POSTMINI.PMI` when you exit the program. Under VMS, a new version of the file is created, so you should occasionally purge the `POSTMINI.PMI` files.

8 POSTMINI printer support

If you want to make a hardcopy plot file, select the `Hardcopy` menu option. Another full screen menu will come up. To generate a hardcopy, enter `P`. To return to the previous menu without making a hardcopy, enter `Q` or `E`. Note: POSTMINI will only create the plot file; it is up to you to print the file. Note: unlike previous versions of POSTMINI, you may make a hardcopy without previewing the graph on your screen.

POSTMINI supports a number of output devices. The supported devices are: color and monochrome PostScript, encapsulated PostScript (for inclusion in other documents, like Microsoft Word or LaTeX), HP-7550 (HPGL), Laserjet-II (HP-PCL), and the LJ250 (color sixels). The plot file will have the name corresponding to the data file you read in, with an extension of `.ps`, `.eps`, `.hpgl` and `.pcl` for the PostScript, Encapsulated PostScript, HP7550 and PCL devices, respectively. You may specify a different output file name from the hardcopy menu.

One might ask why there are two PostScript file types: color and monochrome. Postmini will alter the graphics appearance for the monochrome device for maximum legibility. For example, if one has an X-Y plot with curves of different colors, the curves might be indistinguishable when printed to a non-color printer. For example, both red and blue will appear as black. Postmini will change the different line colors to different line types, in order to clearly show them. The expert user will, of course, just modify graph to use line types and not use color if s/he knows that the graphic will be printed on a non-color printer.

When you generate encapsulated PostScript for inclusion in other documents, you should generate the file in portrait mode, so that the graph does not appear rotated by 90 degrees.

9 Plot annotations

POSTMINI allows the user to interatively annotate any graph. This feature is available only on workstations. The user can add the following items to a graph:

- Boxes, both filled or outline
- Elliptical arcs (useful in grouping multiple curves together)
- Horizontal, vertical or angled lines
- Horizontal, vertical or angled arrows
- Markers
- Mathmode text

Each item is positioned on the graph using the workstation mouse. The user can also delete any item, or reposition items. The final result is a professional looking graph, especially when plotted to the PostScript output device.

Here are a few hints on using the plot annotation feature:

- When you ask to draw a vertical or horizontal line, POSTMINI will "snap" the line to the vertical or horizontal, even if you try to draw a line that is at an angle.
- Filled boxes drawn using a color of "0" (background color) can be used to block out portions of the plot (like electronic "white–out"). This is especially useful in placing text in colored or busy areas of a plot. First draw a filled box the size of the text, then place the text over the box.
- Elliptical arcs are drawn in two steps (v8.2 and higher). First, the user clicks on the lower left and upper right corners of a rectangle that is the bounding box for the arc. Postmini then displays an elliptical arc in "real time." The end points of the arc change as the user moves the mouse. Clicking on mouse button one finishes the process.
- Under some conditions, it is possible for an elliptic arc to "cover" another item, so that the underlying item cannot be selected for deletion or re-positioning. To uncover the item, move the elliptic arc off the item you wish to select.
- POSTMINI uses the built-in PostScript fonts (default: Helvetica Bold) when plotting to the PostScript device. These fonts are professional quality, and are highly legible, even after photo-reduction. The down side of using these fonts is that they are not exactly the same as the fonts used on the workstation. Although the fonts are all scaled to the same size, it is possible that text may come out slightly different in size on these two devices. The difference is usually negligible. The other hardcopy devices do not have this problem.

- You can use horizontal or vertical line annotations to “measure” distances. For example, to measure the horizontal distance from the edge of a MOSFET gate to the junction edge, enter a horizontal line with one end on the gate and the other on the junction edge (remember, horizontal lines “snap” to horizontal, so you enter an angled line if that’s more convenient). Then move the horizontal line down to the horizontal plot axis to determine its length.

10 Expression Evaluator

Postmini V8.3 and higher introduces an expression evaluator code. This allows the user to apply arbitrary expressions to their X or Y axis data, either when read in from an ASCII file, or in a comparison plot. Expressions use a syntax similar to Fortran. The common operators +, -, *, / and ** (exponentiation) are allowed, parenthesis to group operations, as well as a number of functions, listed below. A single variable is allowed, either `x` or `y`, which refers to the X or Y data which will be transformed. If an expression is used, any linear scaling factors (`AX,AY,MX,MY`) are not used.

function name	Operation
sqrt	sqrt
log	log base e
exp	exponential function
log10	log base 10
abs	absolute value
sin	sine function (radian argument)
cos	cosine function (radian argument)
tan	tangent function (radian argument)
asin	arcsin function (returns radians)
acos	arccos function (returns radians)
atan	arctan function (returns radians)
atan2	arctan function with two arguments (returns radians)
max(...)	maximum function (arb. number of arguments)
min(...)	minimum function (arb. number of arguments)
sind	sine function (degree argument)
cosd	cosine function (degree argument)
tand	tangent function (degree argument)
ran	random function [0,1.0], takes one (dummy) argument
The following functions are valid for 1-D data	
integrate(y,x)	Integrate y(x) from initial x to current x value
deriv(y,x)	Differentiate y with respect to x

Note that only one differentiate function is allowed per expression. Versions 9.1-002 and higher of Postmini implement a two-pass function evaluator, which allows implementation of a central difference differentiator, which is more numerically accurate than in previous versions.

11 POSTMINI startup file

When POSTMINI starts up, it reads a file named `POSTMINI_DEFAULTS.DAT` from your current directory. If no such file exists, it tried to read it from your home directory. The startup file can be used to modify many features of Postmini. The file consists of lines as follows:

```
keyword =  [value1 [[value2] ...]
```

Keywords are case independent. Blanks or commas may be used to separate values. Comment lines begin with an exclamation point "!" in column 1. You may place several items on one line by separating the items with a semicolon. The following is a sample startup file:

```
!
!  A sample Postmini default file
!
!  Make hardcopy in portrait orientation
hc_orientation = portrait
!  Make all hardcopy at 60% reduction
hc_scale_factor = 0.6
!  Define default 3D view at radius=8, phi=45 degrees,
!  theta=45 degrees
3d_view = 8 45 45
! Use Courier font for Postscript text
ps_font_id = -109
! Redefine color #2 as light gray
color_2 = 0.7 0.7 0.7
```

The following table lists all valid keywords for the Postmini startup file POSTMINI_DEFAULTS.DAT. In the following table, <string>means any string, <int>means any integer, and <real>means any real number. String values are taken from the first non-blank character after the equal sign to the last non-blank character of the line. The following keywords are recognized:

Postmini default file keywords		
Key	Values	Description
3D_VIEW	<real><real><real>	Default 3D viewpoint (radius, phi, theta)
AUTOSCALE	YES | NO	If "YES", always generate new default plot axis limits using an autoscaling algorithm If "NO", remember the last coordinate axis limits the user specified and use for the next plot (default)
COLOR_nn	<real><real><real>	Defines red/green/blue components of color nn nn ranges from 0 to 15. Postmini normally uses colors 0–11. Color 0 specifies the background color
– or –	<string>	Name of color nn
FONT_ID	<int>	GKS stroke font id for plots to the screen (see Digital GKS manual)
HARDCOPY	postscript | ln03plus | hp7550 | lj250	Default hardcopy device
HC_ORIENTATION	portrait | landscape	Hardcopy orientation
HC_SCALE_FACTOR	<real>(0.1 to 1.0)	Hardcopy reduction factor

Postmini default file keywords (cont)		
Key	Values	Description
HIDDEN_LINE_METHOD	painter — horizon	3D hidden line method
LENGTH_UNIT_NAME_ AND_SCALE	<string><real>	Name of length unit and scale to cm
OXIDE_FILL	YES | NO	Fill regions of oxide with background color if oxide does not have any data associated with it For debugging only, default: YES
PLOT_SCALE_FACTOR	<real>(0.1 to 10.0)	Plot reduction/enlargement factor
PS_FONT_ID	<string>	Selects text font to use for Postscript plots (see section 12)
LINEWIDTH_SCALE	<real>	Linewidth scale factor
SOLID_CONTOUR_LINES	No value	Makes all 2D contour lines solid
TEXT_SCALE	<real>(0.1 to 10.0)	Text reduction/enlargement factor
WRITE_PMIFILE	YES | NO	If YES, writes .PMI file at end of Postmini run (default: YES)
Z_COLORMAP_MINIMUM	<real>	Value of colormap minimum for 3D plots
Z_COLORMAP_MAXIMUM	<real>	Value of colormap maximum for 3D plots

12 POSTMINI command file syntax

The Postmini command file describes all the data needed to recreate a Postmini plot from the original data files. The command file breaks a Postmini plot into several parts, such as type of plot, axis description, data source, annotations, etc. In order to use the same code as for reading the Postmini defaults file, the Postmini command file uses a similar general syntax: a keyword, an (optional) equal sign, and an (optional) list of arguments. Certain special keywords called *main* keywords introduce each part of the plot, and tell Postmini what to expect next. Each main keyword is followed by subkeys which give further information.

The following table lists all valid main keywords for the Postmini command file. Note that some keys go together; for example, a file with a PLOT 2D key must have a CONTOUR key. Keywords may be in any case. They must be spelled out completely. Comments are started by an exclamation point ! in column 1. A choice is denoted by the syntax word1 | word2 | .... Note that new main keywords may be added in later versions of Postmini.

Key	Values	Description
PLOT	1D | 2D | 3D | COMPARE | BARCHART OVERLAY	Type of plot
GLOBAL	none	Global plot parameters
TITLE	<string>	Titling to appear at top of plot
AXIS	X | Y | Z (optional) BOTTOM | TOP | LEFT | RIGHT	Axis parameters
CURVE	none	Source of data for a 1D or COMPARE plot. There can be multiple CURVE commands in a COMPARE plot
CONTOUR	none	Source of data for a 2D contour plot There can be multiple CONTOUR commands in an OVERLAY plot
SURFACE	none	Source of data for a 3D surface plot
ANNOTATE	LINE | BOX | TEXT | MARKER | ELLIPSE	Data for an annotation

The subkeywords for each main keyword are listed in the following tables. Note that new subkeywords may be added in later versions of Postmini. In the following table, <string>means any string, <int>means any integer, and <real>means any real number. String values are taken from the first non-blank character after the equal sign to the last non-blank character of the line. Quoting is not needed to maintain the case of strings.

Postmini 7.4-000 and higher supports symbolic names for colors, line types, marker types and PostScript fonts, as well as integers. Postmini recognizes the following:

Item	Values
Color	Dependent on plot type
	See tables below
	Also: `invisible, off, none`
Line types	`solid, dashed, dotted, dash_dot`
	`dash_2_dot dash_3_dot long_dash`
	`long_short_dash spaced_dash spaced_dot`
	`double_dot triple_dot`
	`none off omit`
Marker types	`circle, square, triangle_up, triangle_down`
	`solid_circle, solid_square`
	`solid_tri_up, solid_tri_down`
	`dot, plus, asterisk, cross`
	`bowtie, hourglass, diamond`
	`solid_bowtie, solid_hourglass, solid_diamond`
	`none off omit`
PostScript	`times, times_italic`
fonts	`times_bold, times_bold_italic`
	`helvetica, helvetica_oblique`
	`helvetica_bold, helvetica_bold_oblique`
	`courier, courier_olblique`
	`courier_bold, courier_bold_oblique`

You can still use integers to specify these items, if you wish. *Note well!* If you have used the `COLOR_NN` command to change Postmini's default colors, the color name associated with index `NN` will no longer be recognized by Postmini. In its place, the name `userdefinednn` will be recognized. The color names are taken from the list of X11 colors (on Unix, see: `/usr/lib/X11/rgb.txt`).

The default colors for 1D and comparison plots are:

Color Index	Name
0	white
1	black
2	red
3	green
4	blue
5	cyan
6	magenta
7	yellow
8	lightgreen
9	gold
10	lightblue
11	orange
12	purple
13	brown
14	gray
15	violet

The default colors for 2D and 3D plots are:

Color Index	Name
0	white
1	black
2	purple
3	blue
4	deepskyblue
5	cyan
6	green3
7	green
8	yellow
9	gold
10	orange
11	red
12	undefined1
13	undefined2
14	undefined3
15	undefined4

12.1 Keyword PLOT

Keyword PLOT		
Subkey	Values	Description

Does not have any subkeys at this time.

12.2 Keyword GLOBAL

Keyword GLOBAL		
Subkey	Values	Description
COLOR_nn	<real><real><real> or <string>	Red,green,blue componens of color color name
FONT_ID	<int>	GKS stroke font id
HARDCOPY_DEVICE	<string>	Default hardcopy device (see section 11)
HC_ORIENTATION	LANDSCAPE | PORTRAIT	Hardcopy orientation
HC_SCALE_FACTOR	<real>	Hardcopy reduction factor, < 1.0
HIDDEN_LINE_METHOD	device | painter | horizon	Type of hidden line algorithm
LENGTH_UNIT_NAME_AND_SCALE	<string><real>	
LINEWIDTH_SCALE	<real>	Scales all lines
PLOT_SCALE_FACTOR	<real>	Scales plot window, < 1.0 makes window smaller, > 1.0 makes window larger
PS_FONT_ID	<string>	PostScript font name (see list of names above)
TEXT_SCALE	<real>	Scales all text
HC_SCALE_FACTOR	<real>	Scales the hardcopy plot (0.1 < 1.0)
HC_ASPECT_RATIO	<real>	Modifies the aspect ratio of the hardcopy plot (0.1 < 2.0, default=0.77)
HC_XOFFSET	<real>	Offsets the hardcopy plot in the X direction (< 1.0)
HC_YOFFSET	<real>	Offsets the hardcopy plot in the Y direction (< 1.0)

12.3 Keyword TITLE

Keyword TITLE	<string>	
Subkey	Values	Description

Does not have any subkeys at this time.

The TITLE command is used to put text labels at the top of plots. For 1D, 2D, 3D plots, only one title line can be specified. For comparison plots, up to five title lines can be specified. For overlay plots, up to three title lines can be specified. If you use more than two titles, you should also specify either SUPPRESS_LABELS or CURV_MODE on the GLOBAL command to make more room on the plot for titling.

12.4 Keyword AXIS

Keyword AXIS		
Subkey	Values	Description
ABS	No value	Take absolute value of data
EXPONENT	<int>	Power of 10 used to scale (divide) data
LABEL	<string>	Axis label
LOG	No value	Take $\log_{10}$ of data
LOG_TIC_FREQ	<int>	Number of decades between major tic marks on log scale (Z axis only)
MAJOR_TIC	<real>	Major ticmarks placed at this interval
MAX	<real>	Axis maximum
MIN	<real>	Axis minimum
MINOR_TIC_FREQ	<real>	Number of minor ticmarks per major tic
UNITS	<string>	Data units
The following are supported for comparison plots only		
AUTOSCALE	No value	Choose min/max, depending on data MIN,MAX,MAJOR_TIC,MINOR_TIC_FREQ are ignored.
COLOR	<string>	Axis color
FORMAT	<string>	Any value "C" language format for a floating point number
MAJOR_GRID	No value	Plot a grid line at each major tic mark
MAJOR_GRID_LINETYPE	<int>	Linetype for major grid lines
MAJOR_TIC_FACTOR	<real>	Major tic scale factor (ratio of length of major tic to minor tic)
MINOR_GRID	No value	Plot a grid line at each minor tic mark
MINOR_GRID_LINETYPE	<int>	Linetype for minor grid lines
MINOR_TIC_SIZE	<real>	Minor tic scale factor (in percent of entire graph)
NONUMBERING	No value	Omits numbering at major tic marks
OMIT	No value	Do not draw in the axis at all
THICKNESS	<real>	Axis thickness scale factor
The following are supported for contour and overlay plots only		
INVERT	No value	Draw the Y axis with increasing values going down

12.5 Keyword CURVE

Keyword CURVE		
Subkey	Values	Description
ALONG_OXIDE	No value	Make cutline along oxide interface
AX	<real>	Added to X axis data
AY	<real>	Added to Y axis data
COLOR	<string>	Color of curve
CUT_LINE_AXIS	X | Y | Z	Make cutline along this axis
CUT_LINE_COORD	<real>	Make cutline at this coordinate
CUT_PLANE_AXIS	X | Y | Z	Make cutline on this plane (MINIMOS 3D)
CUT_PLANE_COORD	<real>	Make cutline at this plane coordinate (MINIMOS 3D)
DATAFILE	<string>	Data file name
DATAFILE_DIMENSION	1 | 2 | 3	Spatial dimension (3 for MINIMOS 3D)
DATAFILE_QUANTITY	<string>	Name of data quantity (see later tables)
DATAFILE_TYPE	MINIMOS | PROMIS | SUPREM3S | SUPREM3P | 2DOP | USEOUT | SUPREM4 | PISCES | ASCII | TIF | SPICE3 | FAIM | MCP2D | BAMBI | VLSICAP ANALYTICAL	Data file type
DATAFILE_RECORD	<int>	Record number of data set inside a file (PROMIS, USEOUT, PISCES)
DOPFILE_GATE_LENGTH	<real>	Scaling length for MINIMOS doping file
DOPFILE_INTERFACE_FILE	<string>	Interface file associated with a MINIMOS doping file
INTEGRATED		Quantity is integrated
INTEGRATION_DIRECTION	X | Y | Z	Direction to integrate

Keyword CURVE (cont)		
Subkey	Values	Description
LABEL	<string>	Text string
LINETYPE	<string>	Line types (see list above)
LINEWIDTH	<real>	Linewidth (1.0 == default)
MARKER_COLOR	<string>	Marker color
MARKER_SIZE	<real>	Marker size (1.0 == default)
MARKER_TYPE	<string>	Postmini marker type (see list above)
MX	<real>	Multiplies X axis data
MY	<real>	Multiplies Y axis data
SORT	No value	Sorts data on X coordinate before plotting
XABS	No value	Takes absolute value of X coordinate
XAXIS_TAG	TOP | BOTTOM	Associates this curve with the top or bottom axis (comparison plots only)
X_EXPRESSION	<string>	Expression to apply to X axis data
YABS	No value	Takes absolute value of Y coordinate
YAXIS_TAG	LEFT | RIGHT	Associates this curve with the left or right axis (comparison plots only)
Y_EXPRESSION	<string>	Expression to apply to Y axis data
VLSICAP_GATE_EDGE	<real>	Position of gate edge
VLSICAP_SI_SIO2_INTERFACE	<real>	Position of Si/SiO2 interface

Keyword CURVE (cont)		
The following parameters are valid only for the ASCII datatype		
Subkey	Values	Description
AZ	<real>	Adds to Z coordinate
CHARACTER_DATA_IN_X_COL	No value	Labels for barcharts
COL_X	<int>	Column which has X axis data
COL_Y	<int>	Column(s) which have Y axis data
– or –	ALL	All remaining columns
COL_Z	<int>	Column which has Z axis data
ERRORBARS		Plot error bars using Z column data
MZ	<real>	Multiplies Z coordinate
READ_LINES	<int>	Read only n lines in file
SAMPLE_FREQ	<int>	Read every <int>lines
SKIP_LINES	<int>	Skip n lines in file before reading
SWEEP	No value	Suppress retrace if X axis data is non-monotonic
ZABS	No value	Takes absolute value of Z coordinate
ZSORT	No value	Sorts data on Z coordinate before plotting
The following parameters are valid only for the SPICE3 datatype		
Subkey	Values	Description
SPICE3_XAXIS_QUANTITY		Quantity to plot along X axis

Keyword CURVE (cont)		
The following parameters are valid only for the ANALYTICAL datatype		
Subkey	Values	Description
ANALYTICAL_EXPRESSION	<string>	Analytical expression
ANALYTICAL_FUNCTION_TYPE	F(X) | F(X,P) | F(X,Y)	Analytical function type
ANALYTICAL_XSTART	<real>	Starting x value
ANALYTICAL_XEND	<real>	Ending x value
ANALYTICAL_XSTEP	<real>	X step
ANALYTICAL_PSTART	<real>	Starting p value
ANALYTICAL_PEND	<real>	Ending p value
ANALYTICAL_PSTEP	<real>	P step
ANALYTICAL_YSTART	<real>	Starting y value
ANALYTICAL_YEND	<real>	Ending y value
ANALYTICAL_YSTEP	<real>	Y step

12.6 Keyword CONTOUR

Keyword CONTOUR		
Subkey	Values	Description
COLOR	No value	Enables color fill contour
CUT_PLANE_AXIS	X | Y | Z	Make plot on this plane (MINIMOS 3D)
CUT_PLANE_COORD	<real>	Make plot at this plane coordinate (MINIMOS 3D)
DATAFILE	<string>	Data file name
DATAFILE_DIMENSION	2 | 3	Spatial dimension (3 for MINIMOS 3D)
DATAFILE_QUANTITY	<string>	Name of data quantity (see later tables)
DATAFILE_TYPE	MINIMOS | PROMIS SUPREM3S | SUPREM3P | 2DOP | USEOUT | SUPREM4 | PISCES | ASCII | TIF | FAIM | MCP2D | BAMBI | VLSICAP ANALYTICAL	Data file type
DATAFILE_RECORD	<int>	Record number of data set inside a file (PROMIS, USEOUT, PISCES)
DOPFILE_GATE_LENGTH	<real>	Scaling length for MINIMOS doping file
DOPFILE_INTERFACE_FILE	<string>	Interface file associated with a MINIMOS doping file
JUNCTIONS	No value	Plots junctions
GRID	No value	Plots grid
NEGATE	No value	Negates data before plotting
VLSICAP_GATE_EDGE	<real>	Position of gate edge
VLSICAP_SI_SIO2_INTERFACE	<real>	Position of Si/SiO2 interface
VALUES	<real><real>... (up to 9 values)	Contour values

Keyword CONTOUR (cont)		
The following parameters are valid only for the ASCII datatype		
Subkey	Values	Description
AX	<real>	Added to X axis data
AY	<real>	Added to Y axis data
AZ	<real>	Added to Z axis data
COL_X	<int>	Column which has X axis data
COL_Y	<int>	Column which has Y axis data
COL_Z	<int>	Column which has Z axis data
MX	<real>	Multiplies X axis data
MY	<real>	Multiplies Y axis data
MZ	<real>	Multiplies Z axis data
SKIP_LINES	<int>	Skip n lines in file before reading
The following parameters are valid only for the ANALYTICAL datatype		
Subkey	Values	Description
ANALYTICAL_EXPRESSION	<string>	Analytical expression
ANALYTICAL_FUNCTION_TYPE	F(X) | F(X,P) | F(X,Y)	Analytical function type
ANALYTICAL_XSTART	<real>	Starting x value
ANALYTICAL_XEND	<real>	Ending x value
ANALYTICAL_XSTEP	<real>	X step
ANALYTICAL_PSTART	<real>	Starting p value
ANALYTICAL_PEND	<real>	Ending p value
ANALYTICAL_PSTEP	<real>	P step
ANALYTICAL_YSTART	<real>	Starting y value
ANALYTICAL_YEND	<real>	Ending y value
ANALYTICAL_YSTEP	<real>	Y step

Keyword CONTOUR (cont)		
The following parameters are may be used for any datatype in OVERLAY plots		
Subkey	Values	Description
AX	<real>	Added to X axis data
AY	<real>	Added to Y axis data
AZ	<real>	Added to Z data
MX	<real>	Multiplies X axis data
MY	<real>	Multiplies Y axis data
MZ	<real>	Multiplies Z data
X_EXPRESSION	<string>	Expression to apply to X axis data
Y_EXPRESSION	<string>	Expression to apply to Y axis data
Z_EXPRESSION	<string>	Expression to apply to Z data

12.7 Keyword SURFACE

Keyword SURFACE		
Subkey	Values	Description
CUT_PLANE_AXIS	X | Y | Z	Make plot on this plane (MINIMOS 3D)
CUT_PLANE_COORD	<real>	Make plot at this plane coordinate (MINIMOS 3D)
DATAFILE	<string>	Data file name
DATAFILE_DIMENSION	2 | 3	Spatial dimension (3 for MINIMOS 3D)
DATAFILE_QUANTITY	<string>	Name of data quantity (see later tables)
DATAFILE_TYPE	MINIMOS | PROMIS | SUPREM3S | SUPREM3P | 2DOP | USEOUT | ASCII | BAMBI | VLSICAP | ANALYTICAL	Data file type
DATAFILE_RECORD	<int>	Record number of data set inside a file (PROMIS, USEOUT)
DOPFILE_GATE_LENGTH	<real>	Scaling length for MINIMOS doping file
DOPFILE_INTERFACE_FILE	<string>	Interface file associated with a MINIMOS doping file
EYE_POSITION	<real><real><real>	Eye position (radius, phi, theta) phi and theta are in degrees phi rotates eye around graph (-180 to 180) theta moves eye over or under graph (0 to 180) Default: 8 -45 45
VLSICAP_GATE_EDGE	<real>	Position of gate edge
VLSICAP_SI_SIO2_INTERFACE	<real>	Position of Si/SiO2 interface

Keyword SURFACE (cont)		
The following parameters are valid only for the ASCII datatype		
Subkey	Values	Description
AX	<real>	Added to X axis data
AY	<real>	Added to Y axis data
AZ	<real>	Added to Z axis data
COL_X	<int>	Column which has X axis data
COL_Y	<int>	Column which has Y axis data
COL_Z	<int>	Column which has Z axis data
MX	<real>	Multiplies X axis data
MY	<real>	Multiplies Y axis data
MZ	<real>	Multiplies Z axis data
SKIP_LINES	<int>	Skip n lines in file before reading
Subkey	Values	Description
ANALYTICAL_EXPRESSION	<string>	Analytical expression
ANALYTICAL_FUNCTION_TYPE	F(X) | F(X,P) | F(X,Y)	Analytical function type
ANALYTICAL_XSTART	<real>	Starting x value
ANALYTICAL_XEND	<real>	Ending x value
ANALYTICAL_XSTEP	<real>	X step
ANALYTICAL_PSTART	<real>	Starting p value
ANALYTICAL_PEND	<real>	Ending p value
ANALYTICAL_PSTEP	<real>	P step
ANALYTICAL_YSTART	<real>	Starting y value
ANALYTICAL_YEND	<real>	Ending y value
ANALYTICAL_YSTEP	<real>	Y step

12.8 Keyword ANNOTATE BOX

Keyword ANNOTATE BOX		
Coordinate range from X=0 to 1.0 and Y=0 to 1.0 (unless using graph coordinates)		
Subkey	Values	Description
COLOR	<string>	Color of box outline
FILLED	No value	Fill interior of box with color
INTERIOR_COLOR	<string>	Color of interior of box
X_LOC	<real>	X coordinate of lower left corner
X_LOC2	<real>	X coordinate of upper right corner
Y_LOC	<real>	Y coordinate of lower left corner
Y_LOC2	<real>	Y coordinate of upper right corner
USE_GRAPH_COORDS		Use graph coordinates to position annotation

12.9 Keyword ANNOTATE LINE

Keyword ANNOTATE LINE		
Coordinate range from X=0 to 1.0 and Y=0 to 1.0 (unless using graph coordinates)		
Subkey	Values	Description
ARROW	No value	Put arrow head on line
DOUBLE_ARROW	No value	Put arrow head on both ends of line
COLOR	<string>	Color of line
LINETYPE	<string>	Linetype
X_LOC	<real>	X coordinate of lower left corner
X_LOC2	<real>	X coordinate of upper right corner
Y_LOC	<real>	Y coordinate of lower left corner
Y_LOC2	<real>	Y coordinate of upper right corner
USE_GRAPH_COORDS		Use graph coordinates to position annotation

12.10 Keyword ANNOTATE TEXT

Keyword ANNOTATE TEXT		
Coordinate range from X=0 to 1.0 and Y=0 to 1.0 (unless using graph coordinates)		
Subkey	Values	Description
COLOR	<string>	Text color
JUSTIFICATION	LEFT | RIGHT	
	CENTERED	Text justification
LABEL	<string>	Text string
ORIENTATION	HORIZONTAL	Text orientation
	VERTICAL	
SCALE	<real>	Scale factor for text (default: 1.0)
X_LOC	<real>	X coordinate of lower left corner
Y_LOC	<real>	Y coordinate of lower left corner
USE_GRAPH_COORDS		Use graph coordinates to position annotation

12.11 Keyword ANNOTATE MARKER

Keyword ANNOTATE MARKER		
Coordinate range from X=0 to 1.0 and Y=0 to 1.0 (unless using graph coordinates)		
Subkey	Values	Description
MARKER_COLOR	(See list of colors)	Color of marker
MARKER_TYPE	(See list of markers)	Marker type
X_LOC	<real>	X coordinate of lower left center of marker
Y_LOC	<real>	Y coordinate of lower left center of marker
USE_GRAPH_COORDS		Use graph coordinates to position annotation

12.12 Keyword ANNOTATE ELLIPSE

Keyword ANNOTATE ELLIPSE		
Coordinate range from X=0 to 1.0 and Y=0 to 1.0 (unless using graph coordinates)		
Subkey	Values	Description
COLOR	<int>	Color of ellipse
X_LOC	<real>	X coordinate of lower left corner of bounding box
X_LOC2	<real>	X coordinate of upper right corner of bounding box
X_LOC3	<real>	X coordinate of cut vector from center of bounding box
Y_LOC	<real>	Y coordinate of lower left corner of bounding box
Y_LOC2	<real>	Y coordinate of upper right corner of bounding box
Y_LOC3	<real>	Y coordinate of cut vector from center of bounding box
USE_GRAPH_COORDS		Use graph coordinates to position annotation

13 Mathmode

The mathmode utility provides a way to enter strings with sub and/or superscripts, Greek letters, and a limited number of mathematical symbols using a subset of the TEX math syntax.

13.1 Features

- Subscripts — Underscore (_) followed by subscript in curly braces.
  Example: `V_{gs}` produces V_{gs}

- Superscripts — Caret (^) followed by superscript in curly braces.
  Example: `n^{+}` produces n^{+}

- Greek letters — Backslash (\) followed by the name of the Greek letter. Upper case letters are specified by capitalizing the *first* letter only.
  Example: `\delta` produces δ
  Example: `\Delta` produces Δ

- Text font styles: Mathmode supports three text styles: `\rm` (Roman, or normal) `\bf` (bold) and `\it` (italic). You can change the font of a portion of text by enclosing the text in curly braces.

- Text fonts (PostScript output only): Mathmode supports three text fonts: `\Times`, `\Courier` and `\Helvetica`. Note that these keys must be entered with an initial upper case in order to be recognized. You can change the font of a portion of text by enclosing these in curly braces. On devices other than PostScript, these keys have no meaning.

- Mathematical symbols — Mathmode can produce the following TEX math symbols:

`\cdot`	$\cdot$	`\oint`	$\oint$	`\_`	_
`\geq`	$\geq$	`\partial`	∂	`\^`	^
`\infty`	∞	`\pm`	$\pm$	`\{`	{
`\int`	$\int$	`\prime`	$\prime$	`\}`	}
`\leq`	$\leq$	`\sum`	$\sum$	`\\`	\
`\nabla`	∇	`\times`	$\times$	`\;`	(space)
`\AA`	Å	`\bot`	$\bot$	`\surd`	$\surd$

- Mathmode also defines the following control sequences:

`\time`	Current time
`\date`	Current date
`\user`	User name

13.2 Implementation details

- Input is always considered to be in-line mathematics mode; that is, there is no need to include dollar signs ($) to start math mode.
- Unlike TEX, you must always enclose any subscripts or superscripts in braces, even if the superscript or subscript is only one character. This exception was allowed so other underscores in the text will not inadventantly cause subscripts.

Right	Wrong
`10^{2}`	`10^2`
`\mu_{0}`	`\mu_0`

- Unlike TEX, spaces are not ignored, with the following exceptions:
  - initial spaces before the first non-blank character
  - trailing spaces after the last non-blank character
  - the first space after a TEX command

  NOTE: In mathmode context, \\ means output a \ not "end of line."
- Only one level of super/sub scripting is supported.
- The hardware fonts on VTxxx terminals are used to increase plotting speed and legibility. As a result, there is limited support for mathmode on these terminals. All commands are output verbatim to the terminal, minus the intial \. Example: `\mu A` becomes `muA`. Sub and superscripts are still plotted correctly.

14 Things to look out for (bugs)

- POSTMINI prints out all quantities at the coordinates of simulation mesh points. MINIMOS prints out electric field and current density at the midpoints between mesh points. The user must be careful in comparing MINIMOS printouts of these quantities with POSTMINI printouts.

- The horizon hidden line removal algorithm for hardcopy devices sometimes makes mistakes when the plotted data has spikes in it. It also will fail if you select a theta angle within a few degrees of zero. To work around the problem, switch to the painters method, which is more accurate, but takes longer to plot.

- Users who use the Motif window manager that comes with Motif V1.2 (e.g. those who use VMS V6.1 and higher), may get a series of messages from GKS to adjust the logicals `GKS$DECW_BORDER_SIZE` and `GKS$DECW_TITLE_SIZE`. Normally, exiting the program and setting the logicals (environment variables on Tru64 Unix) will stop the messages. A change in Motif V1.2 causes them to reappear again and again with different values. To workaround this, add the entry:

  ```
  Mwm*clientAutoPlace:    False
  ```

  in your Motif Window manager startup file (DECW$MWM.DAT on VMS; Mwm on Tru64 Unix). Then re-start the Motif window manager (click mouse button 1 outside any window or icon to get the Motif window manager pop-up menu, then release over the restart menu item).

- GKS V6.5 under Tru64 Unix V4.0F reports a memory deallocation error when it closes any plot window or hardcopy file, and the process is terminated. The workaround is to link in the malloc library from libc from V4.0E. This is probably a bug in the GKS package.

- The QuickWin library supplied with certain versions of Compaq Visual Fortran does not use fixed width fonts to display text from Fortran IO statements under Windows98. The solution is to use the updated library in Compaq Visual Fortran V6.1. This bug does not appear under Windows95 or NT 4.0.

- There is a GKS bug in displaying the text path on certain labels in surface plots on the PostScript output devices. Some labels may print at incorrect angles, depending on the eye position used to view the plot. See the Postmini contact for a workaround. This bug should be fixed in GKS V6.0.

- There is a GKS bug in creating color encapsulated PostScript files. A second file with just the GKS PostScript header information is created. On Unix, the file is of the form `filename.eps_2`. You may delete this file. This bug does

not appear in when creating regular PostScript files. It appears that this bug was introduced sometime in the GKS V6.0 timeframe. This bug does not seem to occur on VMS under GKS V6.4, so it may have been fixed.

15 List of improvements and bug fixes

15.1 Improvements in version V9.2-000

- Analytical functions can now be specified and plotted. Two 1D datasets functions are supported: y as a function of x (y=f(x)) and y as a function of x with an additional parameter p (y=f(x,p)). In addition, a 2D dataset function z as a function of x and y (z=f(x,y)) may be created. The range of independent variables (e.g. x) is specified by a start/stop/step triad. Analytical functions are specified by using the Read function, and specifying the analytical datatype.

- Annotations can now be "attached" to either the background (how things used to work before) or to the graph coordinates When annotations are attached to the background, their position remains fixed as the graph's minimum or maximum X or Y plot bounds are changed. In effect, the annotations are overlaid on the graphics window. When annotations are attached to the graph, their position tracks changes in the graph's minimum or maximum X or Y plot bounds. When annotations are attached to graph coordinates, their angle, aspect ratio, etc. will change according to the aspect ratio of the graph. Text font size, however, remains fixed. Please note: attachment to graph coordinates is not available for 3D plots, since it is impossible to map a 2D position on the plot window uniquely onto a 3D space.

- The function evaluator has been enhanced to allow implementation of central difference differentiation, which is more numerically accurate that the backward difference previously used. As a side-effect, functions that have use the deriv() expression are now evaluated twice.

- The function evaluator now supports more than one integrate function in an expression.

- The print function will now print data from triangular datasets. Data is output in x,y,value triplets. Triangle information is not output.

- The size of two dimensional data sets (both rectangular and triangular grid based) from process and device simulation tools has been raised to from 14400 to 22500 points.
- Arrays used in 3D surface plots are now allocated dynamically. This reduces the memory usage of Postmini when these features are not used.
- Line annotations may now have arrowheads on both ends.
- Mathmode now has expressions for `\date`, `\time` and `\user`.
- Rotated text in annotations used to appear distorted. This has been corrected.

15.2 Improvements in version V9.1-000

- Multiple 2D plots can be performed using the 'overlay' set of functions. Overlay works much like 'compare' plot does, but for 2D contour plots. Multiple datasets can be read in, and plot attributes for each dataset (e.g. contour levels) can be set. The user can apply linear scaling or arbitrary expressions to the X, Y or Z data. Plots can overlap each other, or can be spaced apart by scaling the x or y coordinates. NOTE: the contour plot routines for rectangular grid datasets assume that the X and Y coordinates are monotonically increasing. If you change this by using scaling or applying an expression, the plot may be invalid or crash Postmini.
- 3D surface plots are now supported for triangular based meshes. A crude clipping algorithm has been implemented: any triangle that crosses the current X or Y axis limits is not plotted. 3D histogram plots are not yet implemented; it is not obvious how to do determine how large to make each histogram box.
- An option to set the axis parameters so that the plot appears in the correct aspect ratio has been added to comparison plots. The aspect ratio flag is not obeyed if any axis has log scaling.
- A new command line option "papersize" has been added to select the papersize (e.g. A4, legal) for hardcopy plots.

15.3 Improvements in version V9.0-000

- The limit on the number of data points in X-Y plots (1D or comparison) is now 50000. Some arrays which are used to store data are now dynami-

cally allocated, in order to reduce memory consumption. Standard Fortran90 features are used to implement dynamic arrays.

- The ASCII file reader has been enhanced to support plotting expressions which involve multiple columns of data. For example, if one had three columns of data, one could load column 1 for the X axis, and columns 2 and 3 for the Y axis. One could then apply a y expresion involving column 2 and 3. When more than one Y column is loaded, the Y expression uses variable names of the form "ynnn", where "nnn" is the column number. To continue the previous example, to plot the average of columns 2 and 3, use the y expression: "(y2+y3)/2".

- The zoom/unzoom feature of contour and comparison plots has been enhanced. If the user types `Z2`, Postmini will zoom in by a factor of 2X in each direction. Similarly, if the user types `U2`, Postmini will zoom out by a factor of 2X in each direction.

- A bug parsing the `.PRINT` header of DECSpice ASCII output files has been fixed (commas were treated as field delimiters).

- A SPICE3 file reader has been added. The SPICE3 reader can input so-called SPICE3 "raw" files, both ASCII files generated by SPICE3, and ASCII and binary files generated by PowerSpice. Byte swapping of big-endian binary data files is provided as an option.

- The ASCII file reader has been enhanced to better deal with output files generated by HSPICE. In order for Postmini to read HSPICE files, the HSPICE input file should specify ".options ingold = 2" in his/her input file so that numbers are formatted in exponential notation, rather than use letter suffixes (e.g. `1.0n` for `1.0e-9`).

15.4 Improvements in version V8.3-001

- The number of curves in a comparison plot has been increased to 40.

- For 1D or comparison plots, the maximum number of X-Y points has been reduced to 10000 in order to conserve memory.

- The ASCII file reader for X-Y data has been enhanced to allow non-numeric data in any column that is not used. For example, if the X column is 1, and the Y column is 4, the following lines represent valid input:

```
1.0   2.0  3.0    4.0
1.0   ABC  3.0    4.0
1.0   2.0  DEFG   4.0
1.0  A:B/C Y2K    4.0
```

Note: the field separators are any number of blanks, tabs, commas or equal signs.

- The PISCES file reader has been converted to read the 'classic' Stanford PISCES format (version 9009), rather than the one specific to Digital's internal PISCES.

15.5 Improvements in version V8.3-000

- It is now possible to apply a general expression to X or Y data as read in from an ASCII file, or as applied to a comparsion (multiple X-Y curve) plot.

  The expression can use the operators +, -, /, *, ** (exponentiation) parenthesis and the functions:

```
sqrt == sqrt
log  == log base e
exp  == exponential function
log10 == log base 10
abs  == absolute value
sin  == sine function (radian argument)
cos  == cosine function (radian argument)
tan  == tangent function (radian argument)
asin == arcsin function (returns radians)
acos == arccos function (returns radians)
atan == arctan function (returns radians)
atan2== arctan function with two arguments (returns radians)
max(...) == maximum function (arb. number of arguments)
min(...) == minimum function (arb. number of arguments)
sind == sine function (degree argument)
cosd == cosine function (degree argument)
tand == tangent function (degree argument)
```

Example: to plot I_{ds} vs. $1/L_{eff}$, one would set then the X expression to $1/x$.

Example: to plot failure rate vs. time using the Weibull probability distribution function, one would set the Y expression to $log(log(1/(1-y)))$.

- On contour plots, an option has been added to color regions of different materials with different colors. This is useful to examine the output of process simulation. If you wish to color materials, you must switch off color filled contours and plot contour lines.

- Hardcopy output options have changed. The options X and Y reduction factor, isotropic scaling and expand to full page have been removed. Hardcopy output now always expands to the full extent of the page, consistent with the portrait/landscape orientation of the plot. The user can manipulate the shape of the plot by changing the aspect ratio (height/width) and an overall reduction factor. The default aspect ratio is 0.77. Text scaling for the PostScript output device has been enhanced to try to maintain undistorted text, even when the plot aspect ratio changes drastically.

- Bar charts are now available as an option from the compare plot submenu. Plots with multiple datasets are supported; each dataset appears as a separately colored bar. On monochome output devices, colors are replaced by different types of hatch marks.

  In addition, the ASCII file reader has been extended to support character labels which are displayed under the X axis to label each bar. This new mode can be selected interactively from the `Datafile type` menu item, or the `CHARACTER_DATA_IN_X_COL` key in a Postmini command file. Up to 200 strings can be read in. Character strings are delimited in the file by spaces, tabs, commas or equal signs. Mathmode strings may be used. If you wish to embed a blank in a string, use the Mathmode escape code (
  `;`) for a space.

  Labels are normally displayed horizontally. Postmini estimates the overall label length and switched to vertical display when there are too many labels to plot legibly. The current heuristic is to switch when the product of the maximim string label length times the number of bars is greater than 80.

- Mathmode has been extended to support three different fonts for text (using codes: `\Times` `\Courier` `\Helvetica`) and three different font styles (using codes: `\roman` `\italic` `\bold`). Only the PostScript output device fully supports this new feature. On X displays, only the font style changes.

In order to support this change, Postmini now uses three GKS stroke fonts that have been modified to include more international characters.

On VMS, add these logicals to your login.com file (change the file disk and directory to where you copied the .FNT files)

```
$   DEFINE/NOLOG GKS$STROKE_FONT_NEG9
    yourdisk:[your.directory]VAXGFX$FONT_NEG09_MOD.FNT
$   DEFINE/NOLOG GKS$STROKE_FONT_NEG11 -
    yourdisk:[your.directory]VAXGFX$FONT_NEG11_MOD.FNT
$   DEFINE/NOLOG GKS$STROKE_FONT_NEG15 -
    yourdisk:[your.directory]VAXGFX$FONT_NEG15_MOD.FNT
```

On Unix, add these to your .login file (change the file disk and directory to where you copied the .fnt files)

```
setenv GKSstroke_font_neg9  ~/font_neg09_mod.fnt
setenv GKSstroke_font_neg11 ~/font_neg11_mod.fnt
setenv GKSstroke_font_neg15 ~/font_neg15_mod.fnt
```

- The XY ASCII column data reader now supports up to 100 columns. Some data files created by Warren Anderson use a ”*” to indicate a missing value from a column. Postmini interprets the ”*” as a zero.

- The TIF X-Y data reader now supports up to 250 columns.

15.6 Improvements in version V8.2-001

- Plot windows that have been “kept” on the screen can now be reactivated. Go to the top level menu and enter the “window” menu. The “reactivate” function can be used to reactivate any plot window on the screen. The current window is automatically “kept” and can be restored later. Postmini implements this function by saving a Postmini command file for each kept window to a temporary file in your home directory. The files are called `PM_TEMPWIN.XXXX`, where `XXXX` is a unique string. Restoring a window may require that a datafile be re-read; this may cause some delay.

- One dimensional data of up to 20000 points can now be handled. Support for larger data sizes have been frequently requested by users. This has significantly increased the amount of memory used by Postmini.

- Two dimensional data from ASCII files on a rectangular grid can support up to 50000 points. Data that is not on a rectangular grid has a limit of 14400 points.

- One can now load multiple Y columns when reading X-Y data from ASCII files. One can also specify `ALL` to load all the columns in the file (except for the X column). Postmini automatically assumes `SWEEP` mode, so that any retraces are suppressed.

- The ASCII file reader can now accept X-Y-Z data that is not on a rectangular grid, via a user controlled switch. A Delaunay triangulation code from Steve Fortune of AT&T Bell Laboratories (unencumbered) is used to impose a grid on the data. 3D surface plots are still not supported on triangular data.

- The ASCII file reader has been enchanced: the reader "understands" output files from both DEC SPICE and SOI-SPICE (and probably other SPICE derivatives). The `READ_LINES` option now works for all ASCII file types.

- The method for adding elliptical arc annotations has been improved. The user first draws a rubber-band box which is the bounding box of the ellipse. The user then moves the mouse around the ellipse; Postmini will draw the elliptical arc using the mouse position to cut the ellipse in "real time." Click mouse button 1 to add; mouse button 2 to reject.

- Mathmode has been extended so that it can be used to label the X and Y axes on surface plots (these appear at non-right angles).

- Postmini now intercepts interrupts (control-C on Unix, control-C and control-Y on VMS) and will make an orderly exit, including deleting any temporary files.

15.7 Improvements in version V8.1-003

- Most process and device simulators assume that the Y (depth) coordinate increases in the downward direction. In order for surface plots to appear correct, Postmini has always flipped the X and Y axes. Surface plots now have the option of using a "normal" coordinate system, which is useful for visualizing data from other simulators. When data is imported from ASCII files, the axes are set to use a right-handed coordinate system.

- The way to suppress lines or symbols in comparison plots has been made more intuitive. Previously, one had to specify that the color of the curve

or symbol was ”invisible” to suppress it. User feedback suggested that it was clearer to use the line type or symbol type field to suppress the line or symbol. Hence, users can now specify a line type or symbol type of `none` to suppress the line or symbol. Users can still use the old method of specifying an invisible color if they wish.

15.8 Improvements in version V8.1-001

- The ASCII file reader has been enhanced to use the Z column data to draw error bars on a curve. The error bar is drawn from Y-Z to Y+Z, in other words, the the Z column data is used as the ± “delta.” The error bar takes the color of the curve. If only symbols are plotted, then the error bar takes the color of the symbol.

  Error bars are only supported for comparison plots.

- The MINIMOS binary file reader has been enhanced to handle files created by recent versions of DEC MINIMOS (dated 6/96 or later).

- In the contour plot menu, both the min/max of the entire dataset and the min/max within the current X-Y plot limits is now shown.

- The number of symbols has been increased by six: bowtie, hourglass, diamond, and filled versions of the above.

- Linetypes for line annotations could be set to values 1-12, but only line types 1-4 would appear on the graph.

- Fix bug in which annotations did not show up in hardcopy plot when using the restore function and a plot was not first made to the screen.

15.9 Improvements in version V8.1

- It is now possible to specify two different X or Y axes interactively in comparison plots. Before, one had to use the Postmini command file to access this feature.

- The ASCII file reader has been enhanced to support X-Y-Z data and also X-Y data that is sorted on the Z column. The latter feature is only supported for comparison plots.

- The ASCII file reader has been enhanced to recognize DEC SPICE output files and extract column data from SPICE `.PRINT` statements. No special user intervention is necessary: Postmini examines the first line of the file to determine if the file is SPICE output.

- Handling of VT graphics terminals (ReGIS) is improved. The color sidebar on 3D plots is now readable. A better attempt is made to reset the colormap after each plot.

- Axis labels with values between 0.01 and 0.001 were not formatted with enough decimal places. This could cause erroneous roundup/down. Real numbers printed in the full screen menus had a similar problem. These numbers are now formatted with an additional decimal place.

15.10 Improvements in version V8.0

- Two new datatypes, `FAIM` and `MCP2D`, are now supported. These output file formats are generated by Vector Technologies tools. Due to the nature of the data from these tools, surface plots of FAIM output can be performed, but not for MCP2D. X–Y or contour plots can be done on either format file.

- Postmini versions up to 8.0 could only handle 1D data of up to 500 points, and 2D rectangular grid data of no more than 120 by 120 points. Postmini internals have been re-written to relax these limits without significantly increasing the amount of memory required by the program. One dimensional data of up to 2000 points can now be handled. 2D rectangular grid data (MINIMOS, 2DOP, PROMIS, VLSICAP, BAMBI) can have up 2000 points in either x or y direction, as long as the total number of points does not exceed 14,400. In addition, the FAIM and USEOUT data formats can support up to 50,000 data points. Datasets based on triangles (e.g. TIF, SUPREM4) are unaffected, and have total data point limit of 14,400. These new limits are also available to programs which use the callable (subroutine) interface to Postmini.

- In contour plots, a new mode is available which alters the coordinate plot bounds so that the plot appears in the correct aspect ratio. This is useful for examining physical features in process simulation. The mode may be set or unset from the contour plot menu. Note that when the aspect ratio flag is set, the zoom feature will not follow the exact bounding box that the user inputs with the mouse, but rather a box that maintains the correct aspect ratio.

- A new 3D surface plot type called HISTOGRAM is now available. In HISTOGRAM mode, 3D surface plots of data consist of rectangular box at each data point, with the box height and color corresponding to the size of the data. This mode can be entered through the DEFAULTS menu, or from the statement HIDDEN_LINE_METHOD = HISTOGRAM under the GLOBAL section of the .PMI file.

- The number of regions for TIF, SUPREM4, and PISCES plots has been increased to 20.

- Eight additional linetypes are available for comparison plots: dash-two-dots, dash-three-dots, long-dash, long-short-dash, spaced-dash, spaced-dot, double-dot, triple-dot.

- The 1D integration module now provides some program defined defaults for the integration limits and cut lines.

- Postmini now creates its GKS error log file in the user's home directory, rather than in the current directory. This allows Postmini to run in a directory where it does not have write permission. The form of the error log filename is now GKSERRORS_DATE_PID_COUNT. This change allows more than one Postmini to run in the same directory without each of them stomping on the GKS error file.

- On Ultrix and Digital Unix, Postmini now closes and then re-opens standard output to enable FORTRAN carriage control. This is mainly needed for callable Postmini.

15.11 Improvements in version V7.4

- In the POSTMINI command language, you can now use symbolic names for colors, markers and linetypes, e.g. COLOR = RED. Check section 12 for more information.

- When Postmini computes the quasi-Fermi potentials from data in a 2D MINIMOS binary file, it now sets the majority carrier potential to the bulk voltage (or source voltage for SOI) when only minority carriers are solved for (e.g. MODEL=2-D). This is consistent with what MINIMOS uses internally.

- POSTMINI can now read the "new" TIF and SUPREM4 file formats created by TMA tools such as TSUPREM-4 6.1. POSTMINI automatically detects the old or new file format.

- A number of bugs were fixed in TIF file support: certain quantities which should have appeared in non-silicon regions were not being plotted in contour plots; nodes were being assigned an incorrect material type, sometimes leading to oddities in contour plots; POSTMINI always computed the net doping, even if it appeared in the TIF file; TIF files that had oxide velocity in them caused POSTMINI to mislabel the quantities that appeared in the file.

- POSTMINI now maintains the entire MINIMOS 3D binary file in memory; this makes taking slices of the data along different planes much faster. The arrays to hold the 3D data are allocated only when needed; this minimizes memory requirements for users who do not process 3D files. This change avoids the need for the POSTMINI "incore" version.

- POSTMINI can now interpret CURV command files. POSTMINI uses the actual CURV language parser to read the file, in order to maintain maximum compatibility. You can invoke the CURV parser on a file by specifying the `/curv` option on the command line, or by simply using a file with an extension of `.crv`. POSTMINI's X-Y comparison plots were signficantly enhanced to support most of CURV's features (see below).

- You can now specify separate X and Y scaling factors, as well as X and Y offsets, for any graph when output to a hardcopy device.

- You can now specify an "isotropic" scaling for any graph when output to a hardcopy device. This will scale the plot to the largest square that will fit on the output device.

- A semicolon in column 1 or 2 may be used as a comment character for ASCII files.

The following new features apply only to X-Y comparison plots:

- If you do not specify the `MIN`, `MAX` or `MAJOR_TIC` parameters on a `AXIS` command, Postmini will use default values which depend on the data.

- You can now overlay an X or Y grid on the plot by using the `MAJOR_GRID`, `MINOR_GRID`, `MAJOR_GRID_LINETYPE`, `MINOR_GRID_LINETYPE` parameters.

- You can alter the axis apperance using the `COLOR`, `OMIT`, `NONUMBERING`, `FORMAT`, `THICKNESS`, `MINOR_TIC_SCALE` and `MAJOR_TIC_FACTOR` parameters.

- Specifying the XABS and YABS parameters on a CURVE command will make Postmini take the absolute value of the X or Y coordinate before plotting that data.

- Specifying the SORT parameter on a CURVE command will make Postmini sort the data by X coordinate.

- Specifying the ZSORT parameter on a CURVE command will make Postmini sort the data by a third column (Z coordinate) in an ASCII data file. Associated parameters are COL_Z, MZ, AZ and ZABS. This is equivalent to the CURV CONTOUR feature.

- The TITLE command can be used to specify multiple titles for a plot. If the main title is set from the interactive menu, that title will now appear the any saved .PMI file.

- Specifying the parameter CURV_MODE in the GLOBAL command will alter the the position and size of the plot so that it is compatible with the CURV program. Titles are centered over the graph. PostScript output files use the Times Roman font. The PS_FONT_ID key has no effect in CURV_MODE.

15.12 Improvements in version V7.3

- POSTMINI no longer uses the GKS logical GKS$WS_TYPE to determine which output device to use. The /device option can be used to specify this from the command line. The default type is Motif.

- Support has been added for PROMIS 1.7, which can use a curvilinear coordinate system mesh (this is approximated in Postmini by converting to a triangular mesh).

- Support has been added for TMA "TIF" (Technology Independent Format) files. Note well: Postmini can read TIF files generated by the current field test version of TMA tools on AXP OSF/1. The data formatting will change slightly in the released version of the TMA tools; this will require a small change to the TIF read routines in Postmini.

- Linear data scaling for ASCII files can now be done when the file is read in.

- Integration of 2D triangular data has been implemented.

- Multiple X and/or Y axes are now supported in COMPARE plots. This functionality is accessible at this time only through the .PMI file. The user may now specify a different X axis on the top or bottom of the graph, or a different Y axis on the left or right side of the graph. To acess this feature, specify `LEFT` or `RIGHT` on the `AXIS Y` command, or `BOTTOM` or `TOP` on the `AXIS X` command. To associate a `CURVE` with a particular axis, use `XAXIS_TAG = BOTTOM or TOP` and `YAXIS_TAG = LEFT or RIGHT`.

```
AXIS Y LEFT
MIN = ...; MAX = ...; ...
AXIS Y RIGHT
MIN = ...; MAX = ...;
CURVE
DATAFILE = ...;  ...; YAXIS_TAG = LEFT
CURVE
DATAFILE = ...;  ...; YAXIS_TAG = RIGHT
```

- Sample mode is now available in COMPARE plots. Sample mode now prints the change in coordinates since the last sample point, so it can be used as a measuring device.

- Lines in an ASCII file which contain non-numerical data are quietly skipped.

- Contour plots can use log or linear interpolation to compute contours. The interpolation type is selected depending on the specific data, but the use can override it.

- The Y axis on contour plots can be flipped, so that contour data from external sources where positive Y is “up” can be plotted correctly.

- Numeric values on the color bar in contour and surface plots are now done in scientific notation using Mathmode, rather than computer “E” notation, so plots conform to IEEE publication standards.

- You can suppress the label legends at the bottom of comparison plots via a switch from the comparison plot menu. This is useful when preparing plots for inclusion in another document, or for publication. It also allows more space for annotation text at the top of the graph.

15.13 Improvements in version V7.2.5

- Support for the SUPREM4 and PISCES simulators has been implemented. Postmini was extended to support the triangular meshes used by these simulators. 3D surface plots are not yet supported. In the case of SUPREM4, Postmini reads the ASCII structure file output by SUPREM4. In the case of PISCES, a new output file format was created which has mesh and simulation results all in one file. To create this file, specify `post=t` and `outfile=filename` on the PISCES `solve` statement. A second type of PISCES output file can be created where multiple bias points (or time steps) can be include in one file. Only doping, potential and carrier concentrations are available in this format, in order to conserve disk space. To create this file, specify `movie=t` and `outfile=filename` on the PISCES `solve` statement.

- Manipulation of ASCII files containing X-Y data in columns has been rationalized. To read an ASCII datafile into Postmini, the `READ` command is now used. This change allows Postmini to treat ASCII data the same as any other source of 1D data. Thus, one can now perform a 1D X-Y plot or integration of ASCII data.

- 1D X-Y plots can now have a logarithmic X axis.

- The user can now specify the colormap type (color, gray scale, inverse gray scale) from the hardcopy menu.

- The GKS V5.0A bug which causes POSTMINI to crash when linear scale X-Y plots have a large range has been worked around.

- In comparison plots, the user can cause the plot labels at the bottom of the graph to be suppressed via an option from the plot menu. The graph is then moved down on the page, leaving room for text annotations at the top of the graph.

- Initial plot attributes for a curve in a comparison plot are now set from a full screen menu. Information required for importing an ASCII file is now gotten from a full screen menu.

15.14 Improvements in version V7.2.4

- The `modify` submenu of comparison plots has been converted to a full screen menu, rather than prompting the user with several questions.

- The user can now change the AX, MX, AY, MY scaling parameters used in comparison plots "on the fly" without re-reading the datafile. The scaling parameters can now be applied to data from any source, not just ASCII data.
- Several small code changes were made to accomodate GKS V5.0A changes in default parameters.
- The input parser for PMI and default setting files has been altered to allow multiple parameters on one line. Use a semicolon to separate items. Example:

  ```
  AXIS X; MIN=0.0; MAX=10.0; LABEL= X axis;
  ```

  If you need to have a semicolon included in an item (a text label, or file name, for instance), use single quotes to quote the string. Example:

  ```
  LABEL='This is a string with a ; in it';
  ```

  Note that you only need to quote strings if you are placing them together with other items on a single line and only if they contain a semicolon. To enter a single quote in a quoted string, use two quotes.
- Many internal routines were re-named to make external name conflicts less likely when linking callable postmini with another application. Such applications should avoid external names beginning with pm, anno_, surf_ and mathmode_. Not all external Postmini names have been converted to have unique prefixes at this time.

15.15 Improvements in version V7.2.3

- 2D contour plots now have a sample mode, where the user can use the mouse to point to a position on the graph and find out the corresponding data value and coordinate information.
- 3D surface plots have an orbit feature, which allows the user to manipulate the bounding box of the plot with the mouse, then replot.
- If the axis min/max is changed such that the current axis tic mark parameters are invalid, they are automatically re-computed.
- A SWEEP mode for interpreting ASCII data has been implemented. This allows for plotting information such as multiple I–V curves in a single file by suppressing the 'retrace' at the start of the next curve.

15.16 Improvements in version V7.2.1

- The specification of contours has been enhanced. The user may specify contours by min/max/number of contours, or min/max/step value. Either a linear or log step may be used.
- 2D contour plots now have a zoom mode, where the user can use the mouse to zoom in on a region of interest.
- A logarithmic X (abscissa) axis may now be specified on comparison (multiple curve X–Y) plots.
- Save/restore of integrated quantities is now supported.
- Read of ASCII files with up to 25 columns of data and up to 250 characters per line are now supported. The user can specify a number of lines to be skipped before reading data.

15.17 Improvements in version V7.2

- POSTMINI has changed its plot user interface to a full screen menu mode. Users can change one or more plot attributes and re-plot without having to answer questions on all the plot attributes. More attributes, like plot labeling, can be now modified by the user.
- Hardcopy is more flexible. Users may request a hardcopy without making a screen preview first. User may change the hardcopy output file name, orientation, and use of color.
- The monochrome hidden line drawing algorithm has been optimized to be more efficient. Drawing speed has improved 20–50%, and PostScript hardcopy file size has decreased by 20–50%.
- A subroutine interface to Postmini is now available.
- POSTMINI now takes options from the command line.

15.18 Improvements in version V7.1

- POSTMINI now has the ability to save all the information necessary to recreate any plot in a file and restore it.

- The format of the POSTMINI.NOTE file has changed to an new format compatible with the POSTMINI command file.
- POSTMINI can now display arbitrary slices through a 3D doping file. It automatically reads in the interface file (.INT) file associated with a MINIMOS doping file (if it exists) and displays it.
- 2D PROMIS plots now include the oxide shape information.

15.19 Improvements in version V7.0

- POSTMINI can now read a number of device and process simulation data files, including PROMIS, SUPREM3, BAMBI, VLISCAP, 2DOP and USE-OUT files.
- Plot file names are no longer always `POSTMINI.XXX`, but take the name of the data file from which they were read.
- Curves in contour plots can now be automatically labeled, if desired.
- Plot annotations are automatically saved and can be restored. Line style can be specified on annotations.
- In contour plots of MINIMOS data, the oxide is painted white in order to be better distinguished. This is not done if the plotted quantity has a value in the oxide.
- In contour plots of VLSICAP data, the boundaries and original triangular mesh can be plotted. POSTMINI gets this information from the VLSICAP `.INP` and `.LGF` files. POSTMINI looks in the "standard" VLSICAP directories for these files (e.g. `.INP` files are in [.input] and `.LGF` files are in [.lgf_vlsi]). If these files do not exist or are inconsistent with the `.L3D` data file, then these annotations may not be plotted or may be incorrect.
- A `POSTMINI_DEFAULTS.DAT` file is read at startup to define Postmini defaults.

15.20 Improvements in version V6.1

- POSMTINI now supports color PostScript. Color PostScript can be printed on an LJ250 printer using the DEC PSPRINT utility (DECprint Utility for PostScript to Sixel Printing). This provides outstanding color printing.

- POSTMINI can now plot data from plain ASCII files, as long as the data is in columns. It supports the CURV `.ON` and `.OFF` commands in data files, comment lines using C or !, as well as scaling of the input data.
- POSTMINI now gives the user complete control of line type, line color, symbol type and symbol color on COMPARE plots.
- In comparison plots, you can now select whether a curve will be included in a plot. With this feature, you can load several different data curves at once, and plot only the ones you want.
- The user is now supplied with default values for most prompts.

15.21 Improvements in version V6.0

- POSTMINI now supports both 2D and 3D MINIMOS binary files. In the 3D case, users can plot data along the length, width or top (wafer) planes.
- A new 3D plotting algorithm has been developed which provides hidden line removal on hardcopy devices.
- Color fill contour and 3D plots are now available.
- Contour plots now have a position scale plotted next to the plot window.
- Plotting of the axis tic marks and scales now uses a "sliding-scale" technique, which is better for plot bounds that are not even numbers.
- 3D plots now zoom to the exact device coordinates specified by the user, interpolating when necessary.
- POSTMINI now supports DECWindows and the LJ250 printer.

16 For further information and support

The support contact (and author) of POSTMINI is John Faricelli, 508-841-3237, RICKS::FARICELLI. Your comments, questions, bug reports and suggestions for improvements to POSTMINI are highly valued. If you would like to see a new feature in POSTMINI, please contact me.

17 Acknowledgments

Nadim Khalil provided the 3D plot routines from his SURF program for use in POSTMINI and for the CURV parser code. Marden Seavey provided the initial code used to read VLSICAP data files, and Christian Schiebl provided assistance in interpreting the input and mesh files. Gerd Nanz provided the code to read BAMBI files. Nadim Khalil, Bill McGee and Christian Schiebl provided invaluable input on the new full screen menu user interface in POSTMINI V7.2.

The Delaunay triangulation code used in Postmini was written by Steve Fortune of AT&T Bell Laboratories. He requested that the following notice appear in any documentation.

A File quantity names

A.1 MINIMOS quantity names

MINIMOS quantity names	
Quantity	Description
POTENTIAL	Electrostatic potential
NET_DOPING	Net doping concentration
ELECTRONS	Electron concentration
HOLE	Hole concentration
EX	X component of electric field
EY	Y component of electric field
EJ	Electric field in direction of current density
EMAG	Magnitude of electric field
JX_N	X component of electron current density
JY_N	Y component of electron current density
JMAG_N	Magnitude of electron current density
JX_P	X component of hole current density
JY_P	Y component of hole current density
JMAG_P	Magnitude of hole current density
AVL	Avalanche generation rate
QFN	Electron quasi-Fermi potential
QFP	Hole quasi-Fermi potential
MU_N	Electron mobility
MU_P	Hole mobility
EXCESS_N	$N_d^+ - n$
EXCESS_P	$p - N_a^-$
NET_CHARGE	$N_d^+ - n + p - N_a^-$
NSS	Fixed interface charge
DIT	Fast interface charge
DIT_CHARGED	Charged fast interface charge
JZ_N	Z component of electron current density
JZ_P	Z component of hole current density
EZ_P	Z component of electric field
VX_N	X component of electron velocity
VY_N	Y component of electron velocity
VMAG_N	Magnitude of electron velocity
VX_P	X component of hole velocity
VY_P	Y component of hole velocity
VMAG_P	Magnitude of hole velocity

A.2 SUPREM3 quantity names

SUPREM3 quantity names
Starred quantities only available from SUPREM structure files

Quantity	Description
B	Active boron concentration
P	Active phosphorous concentration
AS	Active arsenic concentration
SB	Active antimony concentration
NET_DOPING	Net active concentration
B_CHEM	Chemical boron concentration*
P_CHEM	Chemical phosphorous concentration*
AS_CHEM	Chemical arsenic concentration*
SB_CHEM	Chemical antimony concentration*
B_INACTIVE	Inactive boron concentration*
P_INACTIVE	Inactive phosphorous concentration*
AS_INACTIVE	Inactive arsenic concentration*
SB_INACTIVE	Inactive antimony concentration*

A.3 PROMIS quantity names

PROMIS quantity names	
Quantity	Description
B	Boron concentration
P	Phosphorous concentration
AS	Arsentic concentration
SB	Antimony concentration
AS_INACTIVE	Inactive arsenic concentration
INTERSTITIALS	Silicon interstitial concentration
VACANCIES	Silicon vacancy concentration
ELECTRONCS	Electron concentration
HOLES	Hole concentration
ATOMIC_NO_NN	Element with atomic number NN
POTENTIAL	Electrostatic potential
SI_SIO2_INTERFACE	Silicon/Oxide interface
SIO2_THICKNESS	Oxide thickness
NET_DOPING	Net doping concentration

A.4 USEOUT quantity names

USEOUT quantity names	
Quantity	Description
USEOUT	Useout quantity (may represent anything)

A.5 VLSICAP quantity names

VLSICAP quantity names	
Quantity	Description
POTENTIAL	Potential
EX	X component of electric field
EY	Y component of electric field

A.6 BAMBI quantity names

BAMBI quantity names	
Quantity	Description
POTENTIAL	Electrostatic Potential
ELECTRONS	Electron concentration
HOLES	Hole concentration
EX	X component of electric field
EY	Y component of electric field
EMAG	Magnitude of electric field
MU_N	Electron mobility
MU_P	Hole mobility
JX_N	X component of electron current density
JY_N	Y component of electron current density
JMAG_N	Magnitude of electron current density
JX_P	X component of hole current density
JY_P	Y component of hole current density
JMAG_P	Magnitude of hole current density
JX_TOTAL	X component of total current density
JY_TOTAL	Y component of total current density
JMAG_TOTAL	Magnitude of total current density
NET_DOPING	Net doping concentration
NET_CHARGE	$N_d^+ - n + p - N_a^-$
NET_RECOMBINATION	Net recombination rate
X_ERROR	X error estimate
Y_ERROR	Y error estimate
TOTAL_ERROR	Total error estimate
N_INT	Integrated electron concentration
P_INT	Integrated hole concentration
N_P_INT	Integrated $n - p$
RECOMB_INT	Integrated recombination
QFN	Electron quasi-Fermi potential
QFP	Hole quasi-Fermi potential

A.7 2DOP quantity names

2DOP quantity names	
Quantity	Description
DONORS	Donor concentration
ACCEPTORS	Acceptor concentration
NET_DOPING	Net doping
UPPER_OXIDE	Upper oxide shape
LOWER_OXIDE	Lower oxide shape

A.8 PISCES quantity names

PISCES quantity names	
Quantity	Description
POTENTIAL	Electrostatic potential
NET_DOPING	Net doping concentration
ELECTRONS	Electron concentration
HOLES	Hole concentration
QFN	Electron quasi-Fermi potential
QFP	Hole quasi-Fermi potential
EX	X component of electric field
EY	Y component of electric field
JX_T	X component of total current density
JY_T	Y component of total current density
JX_N	X component of electron current density
JY_N	Y component of electron current density
JX_P	X component of hole current density
JY_P	Y component of hole current density
CUR_POT	Current flowlines
NET_RECOMB	Net recombination

A.9 SUPREM4 quantity names

SUPREM4 quantity names	
Quantity	Description
POTENTIAL	Electrostatic potential
NET_DOPING	Net doping concentration
VACANCIES	Vacancy concentration
INTERSTITIALS	Interstitial concentration
ELECTRONS	Electron concentration
HOLES	Hole concentration
AS_CHEM	Chemical arsenic concentration
P_CHEM	Chemical phosphorous concentration
B_CHEM	Chemical boron concentration
SB_CHEM	Chemical antimony concentration
XVELOCITY	X component of interface velocity
YVELOCITY	Y component of interface velocity
DRYO2	Dry O2 concentration
WETO2	Wet O2 concentration
ITRAPS	Interstitial trap concentration
DELA	Delta Interface Area
AS	Active arsenic concentration
P	Active phosphorous concentration
B	Active boron concentration
SB	Active antimony concentration
SUPREM_ID_NN	Impurity NN

B Notes for operation on Alpha based systems (OpenVMS and Digital Unix)

POSTMINI runs on Alpha based computers (OpenVMS and Digital Unix). Digital GKS V5.3 and higher supports all graphical output devices. For compatibility with VAX based systems, Postmini on OpenVMS Alpha is compiled with `/D_FLOAT`. This means that all binary files read by POSTMINI must be in `D_FLOAT` format. If you need to read data in `G_FLOAT` (default on OpenVMS Alpha systems), or `IEEE` float, you can do this by defining a logical name which will cause the floating point data to be converted on the fly. For OpenVMS:

```
$ define FOR$CONVERT009 vaxg
 or
$ define FOR$CONVERT009 ieee
```

On Digital Unix, Postmini only supports IEEE floating point. FORTRAN binary datafiles are incompatible between OpenVMS and DEC Digital Unix, due to different on-disk data and floating point formats. The Postmini code can be easily customized to read OpenVMS FORTRAN binary data files by making a one line change to the FORTRAN `OPEN` statement. See the contact person for details.

C Interoperability with PC X displays

Postmini was built using the Digital GKS layered product, an ANSI/ISO standard for two dimension graphic programs. Digital's implementation for X displays assumes that you are running a Motif window manager, along with the normal X server. Unfortunately, Digital's GKS does not interoperate well with many PC X servers, including Digital's eXcursion X server software, since the Windows 3.1 version does not come with the Motif window manager software. The symptom is that the Postmini icon appears on the PC screen, but the graphics window never de-iconizes. GKS verion 6.5 has some changes to improve interoperability with PC X servers. It is not perfect, but it usually works.

- Make sure eXcursion is running on your PC.
- If you do not already have your X display set on the host (VMS or Unix) machine, set it as follows:

  ```
  VMS:
  set display/create/node=xxxx/transport=tcpip

  Unix:
  setenv DISPLAY xxxx:0
  ```

  Note: `xxxx` should be replaced by the actual tcpip node name of your PC. You can find this from the eXcursion control panel. Right click on the eXcursion icon and select "Control panel".
- Run postmini as usual. Contrary to previous instructions, do not start up a Motif window manager process.
- When you make a plot, the window comes up the wrong size and position. Move the window slightly and then press the "Reset" button in the plot window. It should resize to the correct size and position.

As an alternative, you may wish to use the ReGIS graphics capabilities of the KEATerm terminal emulator that is commonly distributed on PC's in Digital. The display speed is quite good, although advanced capabilities, such as annotation, zoom, etc. are not supported. It is also the recommended solution for displaying Postmini graphics over a phone line to a home PC. Use the `/device=vt340` command line option (VMS) or `-device vt340` (Unix) to tell Postmini to output ReGIS graphics.

D Displaying Postmini Graphics to a Home PC

If you are logged into a VMS or Unix host from your home PC running Windows software via a dial-up phone line, the best performance can be achieved by using the ReGIS graphics mode of the KEAterm 340 terminal emulator. Although it is possible to run eXcursion software on the PC using the phone line as a network transport, the performance is very poor. You may wish to try the free PC X server software (Web URL: `http://www.microimages.com/freestuf`) as this is reported to be faster (I have not tried it with Postmini).

If you are running Linux on your PC, you may wish to log into your VMS or Unix host and use the `xterm` display device (e.g. on VMS: `/device=xterm`, or on Unix: `-device xterm`). This will cause Postmini to pop up the xterm's Tektronix 4014 emulation window and draw its graphics there. The text window retains the Postmini text menu, much like under Motif. Unfortunately, I can't get interactive features like annotation or zoom to work correctly, so I have disabled them for this device.

E ASCII output file extensions

The following extensions are used for data files created by the LINE and PRINT functions for MINIMOS. Similar extensions are used for other simulators.

`.PSI`	potential
`.DCONC`	doping concentration
`.ECONC`	electron carrier density
`.HCONC`	hole carrier density
`.EL`	lateral electric field
`.ET`	transverse electric field
`.EJ`	field in direction of current
`.EMAG`	magnitude of electric field
`.JL_E`	lateral electron current density
`.JT_E`	transverse electron current density
`.JMAG_E`	magnitude of electron current density
`.JL_H`	lateral hole current density
`.JT_H`	transverse hole current density
`.JMAG_H`	magnitude of hole current density
`.AVL`	avalanche generation rate
`.QF_E`	electron quasi-Fermi level
`.QF_H`	hole quasi-Fermi level
`.ML_E`	electron lateral mobility
`.MT_E`	electron transverse mobility
`.ML_H`	hole lateral mobility
`.MT_H`	hole transverse mobility
`.CH_E`	$(N_d - n)$
`.CH_H`	$(p - N_a)$
`.CH_NET`	net charge
`.ICH`	Fixed interface charge

F POSTMINI command line arguments and options

POSTMINI takes two optional arguments: the name of a data file to be read, and a datafile name type. If the data file name does not have an extension that POSTMINI recognizes, you must also specify the type. You may abbreviate the data file type, as long as it is unambiguous. Example:

```
$  postmini nchan_test.bin m
```

reads a `MINIMOS` binary file named `nchan_test.bin` into POSTMINI. You may also specify a `PMI` or `CURV` file on the command line.

The following table lists the supported data file types:

Datafile type	Description
`2dop`	2D or 3D MINIMOS doping file
`Ascii`	ASCII columns data file
`Bambi`	BAMBI data file
`Curv`	Curv command file
`Minimos`	MINIMOS 2D or 3D binary file
`PIsces`	PISCES binary save file (from DEC PISCES)
`PMi`	POSTMINI command file
`PRomis`	PROMIS binary save file
`SPICE3`	SPICE3 (PowerSpice) save file
`SUPREM3P`	SUPREM3 `PRINT MINIMOS` file
`SUPREM3S`	SUPREM3 binary structure file
`SUPREM4`	SUPREM4 ASCII structure file
`Useout`	USEOUT binary data file
`Vlsicap`	VLSICAP data file
`TIF`	TMA interchange format

The following table lists the datafile extensions that POSTMINI recognizes:

Datafile extension	Description
`.2dop or .dop`	2D or 3D MINIMOS doping file
`.bin or .bin3d`	MINIMOS 2D or 3D binary file
`.crv`	CURV command file
`.crv2d or .crv3d`	MINIMOS I-V data file (ASCII)
`.pmi`	POSTMINI command file
`.raw`	SPICE3 (or PowerSpice) save file
`.save`	PROMIS binary save file
`.spo`	DEC SPICE output file
`.tif`	TMA TIF output file
`.useout`	USEOUT binary data file

POSTMINI takes several command line options:

VMS options Option	Action
/batch	Runs Postmini in non-interactive mode
/output=file	Specifies output file for hardcopy
/curv	Invokes the CURV parser on the input file
/device=type	Specifies default plot or hardcopy device Plot devices: motif, vt240, vt240bw, vt330, vt340, xterm Hardcopy devices (normally used with /batch): postscript color_postscript hpgl ddif epsf color_epsf ln03 lj250
/inverse_gray	Use inverted gray scale on color output devices
/portrait	Use portrait mode on hardcopy output devices
/scale=fac	Hardcopy reduction scale factor 0.1 <= fac <= 1.0
/gray	Uses a gray scale on color output devices
/convert=type	Apply byte swapping to binary file (SPICE3 only) Allowed types: big_endian
/papersize=type	Set papersize for hardcopy plots Allowed types: A,B,C,D,E,legal (US) A0,A1,A2,A3,A4,B4,B5 (metric)

Ultrix, Digital Unix options	
Option	Action
`-batch`	Runs Postmini in non-interactive mode
`-o file`	Specifies output file for hardcopy
`-curv`	Invokes the CURV parser on the input file

`-device type`	Specifies default plot or hardcopy device
	Plot devices:
	`motif, vt240, vt240bw, vt330, vt340, xterm`
	Hardcopy devices (normally used with `-batch`):
	`postscript color_postscript hpgl ddif epsf color_epsf ln03 lj250`

`-inverse_gray`	Use inverted gray scale on color output devices
`-portrait`	Use portrait mode on hardcopy output devices
`-scale fac`	Hardcopy reduction scale factor 0.1 <= fac <= 1.0
`-gray`	Use gray scale on color output devices
`-convert type`	Apply byte swapping to binary file (SPICE3 only) Allowed types: big_endian
`-papersize type`	Set papersize for hardcopy plots Allowed types: A,B,C,D,E,legal (US) A0,A1,A2,A3,A4,B4,B5 (metric)

G Papersizes

US sizes (inches)	
Name	Size
A	8.5” x 11”
LEGAL	8.5” x 14”
B	11” x 17”
C	17” x 22”
D	22” x 34”
E	34” x 44”
COMP	11” x 14”
Metric sizes (cm)	
Name	Size
A0	84.1 x 118.9
A1	59.4 x 84.1
A2	42.0 x 59.4
A3	29.7 x 42.0
A4	21.0 x 29.7
A5	14.8 x 21.0
B4	25.7 x 36.4
B5	18.2 x 25.7

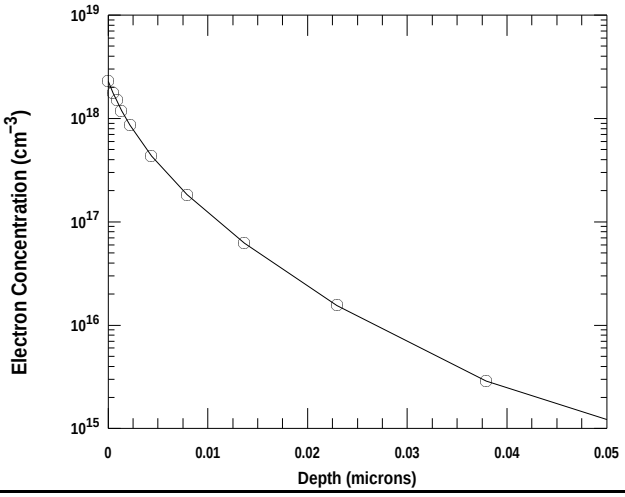
H Plot examples

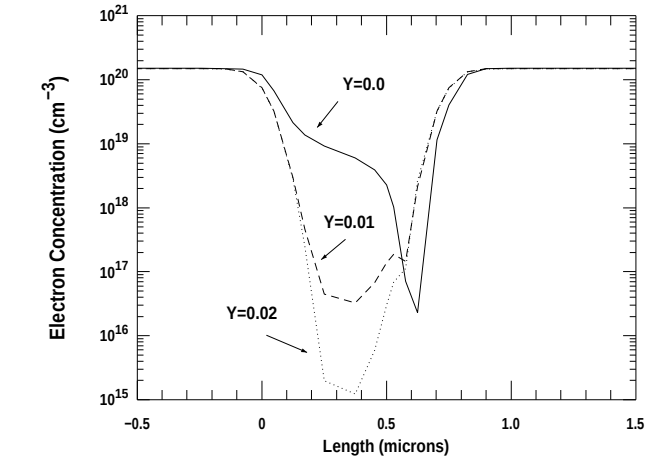
This section contains four plots from the results of a simulation of an n-channel device simulated with MINIMOS. Plot annotations were added to POSTMINI graphs interactively using a workstation and mouse. All plots were printed on a PostScript printer.

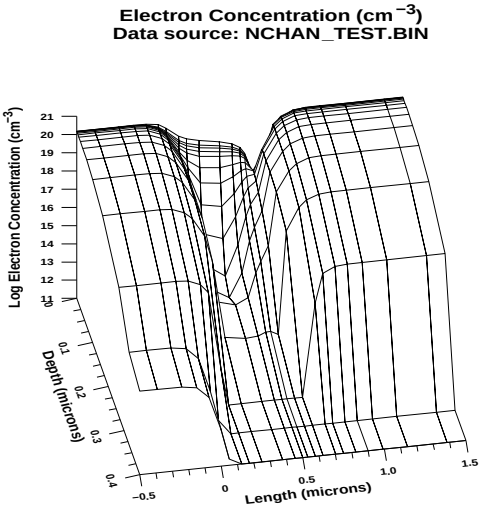
- A 1D plot of the electron current density versus depth at X = 0.5 μm.
- A comparison plot showing the electron concentration from source to drain at depths of Y = 0.0 μm, 0.01 μm and 0.02 μm.
- A 3D surface plot of the doping (log scale) over the region from X = -0.5 μm to 1.5 μm, and Y = 0.0 μm to 0.4 μm.
- A contour map of the doping in the same region as the 3D plot.

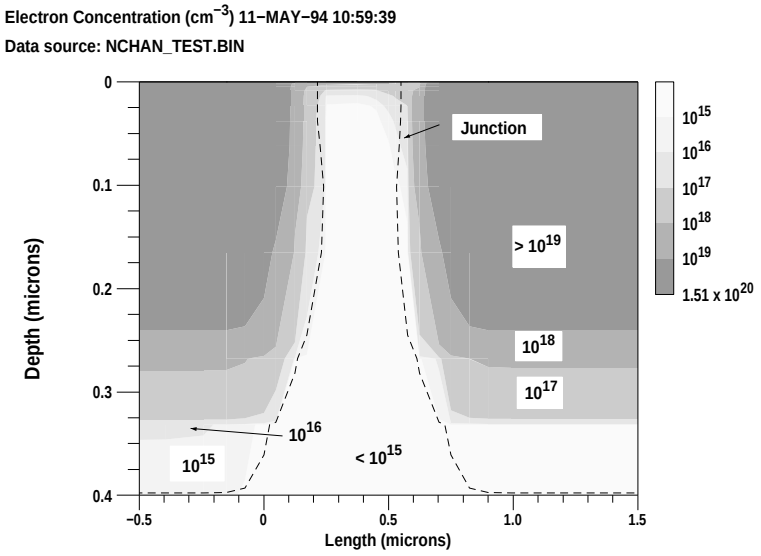
Electron Concentration (abs) at X = 0.50000 ; 11−MAY−94 10:59:25

Data source: NCHAN_TEST.BIN






Electron Concentration (cm^{-3})
Data source: NCHAN_TEST.BIN
Log Electron Concentration (cm^{-3})
21
20
19
18
17
16
15
14
13
12
11
Depth (microns)
0
0.1
0.2
0.3
0.4
−0.5
0
0.5
1.0
1.5
Length (microns)


Electron Concentration (cm^{-3}) 11−MAY−94 10:59:39
Data source: NCHAN_TEST.BIN
Depth (microns)
0
0.1
0.2
0.3
0.4
Junction
> 10^{19}
10^{18}
10^{17}
10^{16}
10^{15}
< 10^{15}
10^{15}
10^{16}
10^{17}
10^{18}
10^{19}
1.51 x 10^{20}
−0.5
0
0.5
1.0
1.5
Length (microns)