

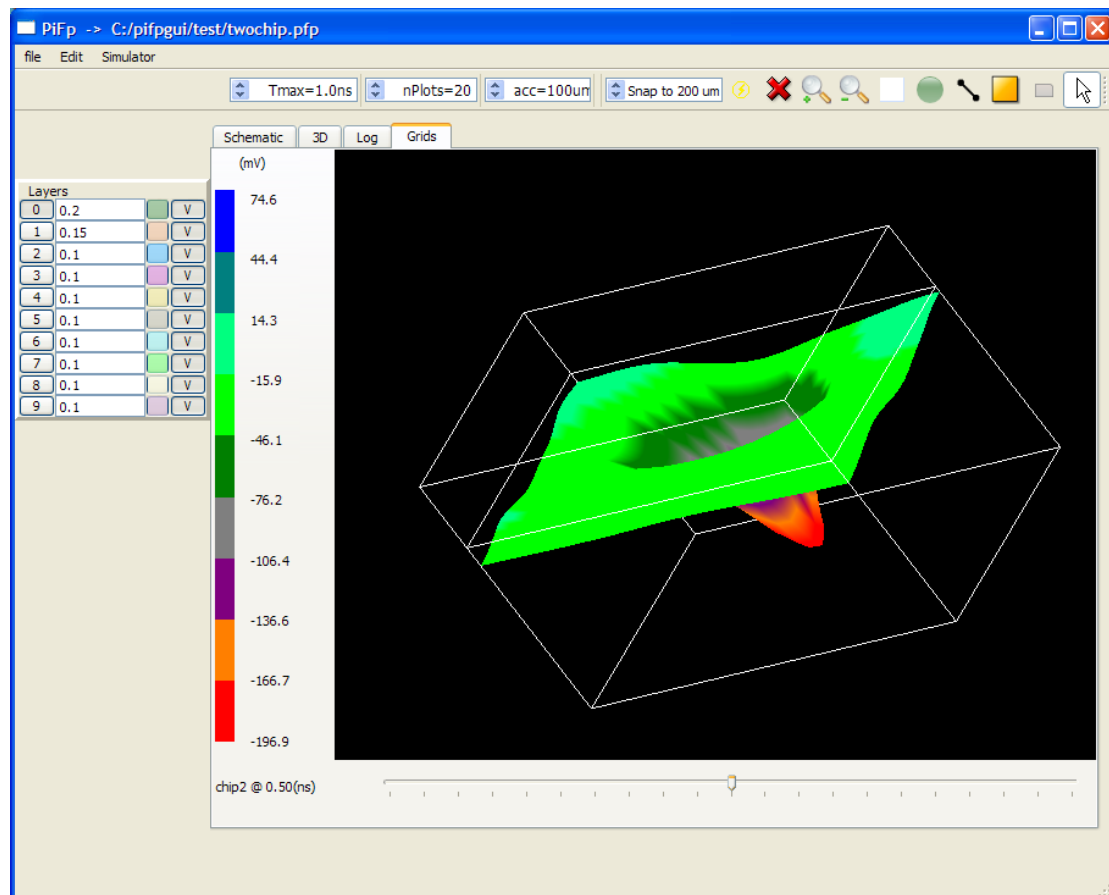
Π-FP User Manual

Version 1.0

Anasim Corporation

3838 E Encinas Ave
Gilbert, AZ 85234, USA

Tel: +1 (480) 626-7535 / email: enquiries@anasim.com



Licence

II-FP

Please note that this software product together with its accompanying documentation is the property of Anasim Corporation. The product is made available to you under the following license agreement as described on our web pages:



pi-fp by [Anasim Corp.](#) is licensed under a [Creative Commons Attribution-NonCommercial 3.0 Unported License](#).

Please contact us with any questions / clarifications needed. For contributions toward further development of pi-fp and other efforts at Anasim, please use our [PAYPAL](#) a/c.

Anasim Corporation
3838 E Encinas Ave
Gilbert, AZ 85234, USA
Tel: +1 (480) 626-7535

Web: <http://www.anasim.com/> email: enquiries@anasim.com

Copyright © 2003-2015 Anasim Corporation, π -fp software and documentation. All Rights Reserved.

Contents

Chapter 1	
Introduction to Π-FP	5
Chapter 2	
Tutorial	12
Chapter 3	
Netlist and parameters	17
# comment	18
Global on-chip grid definition	19
On-chip current sources	20
On-chip capacitance	21
Transmission lines	22
Connection node	23
Ideal cap	24
.TRAN	25
Accuracy parameter	26
.PLOT	23
.PRINTNODE	27
Chapter 4	
Graphical output using GNUplot	28
Chapter 5	
Invocation and Misc.	34

Chapter 1

Introduction to Π -FP

Increased functionality, power density and lower supply voltages place increasing demands on the power distribution networks of current and future systems containing large scale integrated circuits. Careful power planning is necessary to reduce power noise related problems such as timing errors, increased stress in thin gate oxides and circuit failure. Π -FP is a fast, system level power integrity aware floor planner. It calculates resistive and inductive voltage changes in on-chip and system level power grids as a function of grid parameters, time dependent current sources, on and off chip capacitance. By separating the power net into a global net (two layer symmetric on-chip and external t-lines) where most of the energy is stored in the magnetic field and dissipated as heat, and a local net (everything else) where almost all of the energy is stored in the electric field (de-cap), the problem can be simplified to the point where simulation run times can be reduced by many orders of magnitude.

Figure 1.1 shows the main components in a system level power network designed using Π -FP. A key component is the on-chip global power distribution grid. This is a two layer symmetric grid constructed using an array of closely spaced transmission line pairs. The wires in each power/ground pair are placed at or near the minimum design space to minimise inductance and maximise capacitance in the global grid. The designer connects all active and inactive (de-cap) circuits to this grid using an array of short vertical wires (vias). If enough of these connections are made then all resistive and inductive voltage drops in the vertical direction are small enough to ignore (the typical distance to all other wiring layers in the IC is no more than a few microns).

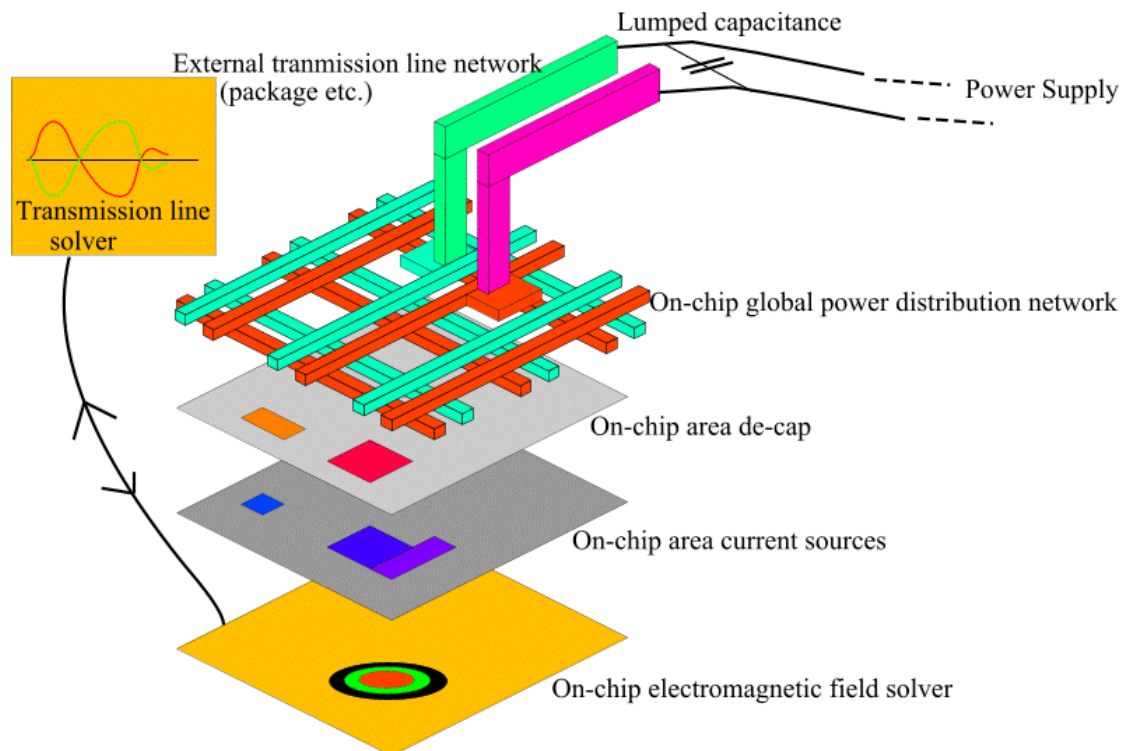


Figure 1.1. Components in a system level power network

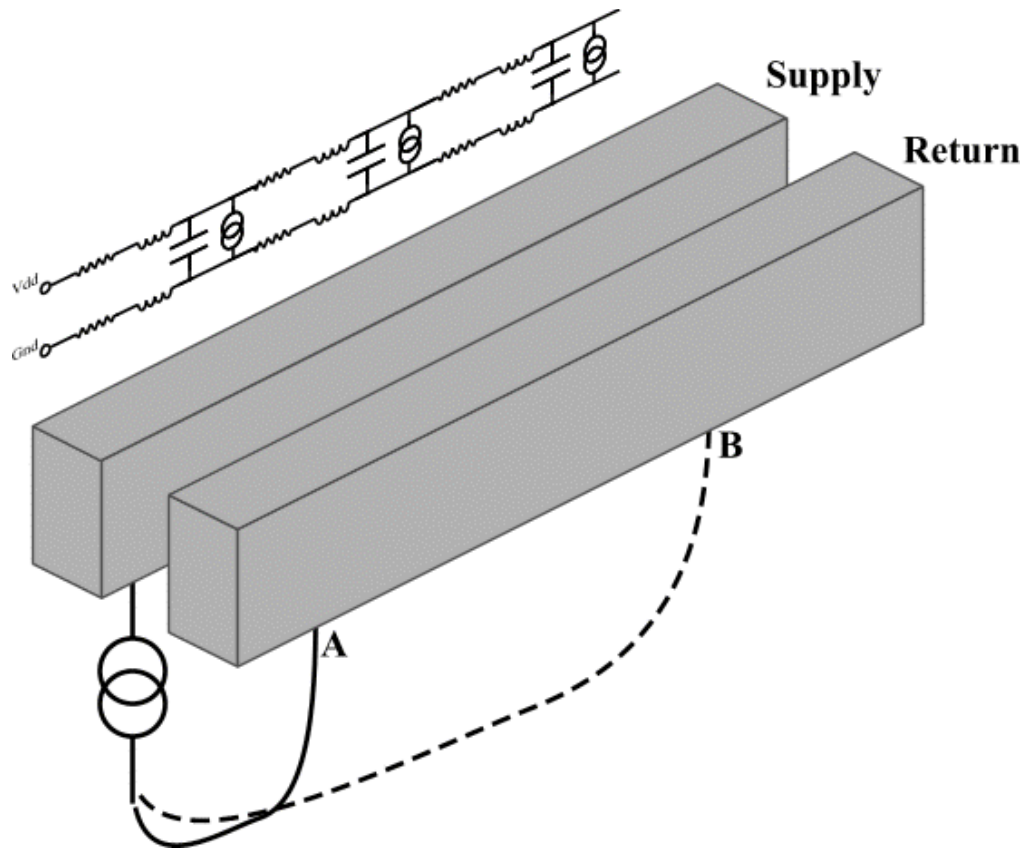


Figure 1.2 Symmetric connections to the on-chip power grid.

This also ensures that all voltage drops in horizontal wires in all other layers are small enough to ignore. Under these conditions a significant portion of on-chip voltage drops and increases are caused by horizontal current flow in the global two layer grid.

Π -FP is a multi-chip simulation tool. Any number of connected IC's, transmission lines and floor plan grids can be simulated. In high speed systems the interactions between multiple IC's, package and board level components often dominate power noise characteristics. Large capacity, system level, dynamic power noise simulations are now possible using Π -FP and symmetric transmission line power grid designs.

Figure 1.2 illustrates the importance of symmetry in high-speed power network designs. Connecting the current source at point A rather than point B ensures that voltage and current waves are equal in magnitude and have opposite signs in supply and return nets. Equal and opposite currents in closely spaced transmission line pairs have the advantage of minimising unwanted magnetic coupling between the power supply and other networks. This symmetry also means that it is not necessary to calculate power noise on both rails. We can simulate one rail only using a single uncoupled partial differential equation as shown in figure 1.3.

In Π -FP current supplied to active devices in the chip is modelled using a distributed current source. The user supplies a single current profile for each circuit block (e.g. obtained from a transistor or gate level simulation). The "block" could represent the current sourced by a single transistor or gate but this level of detail is usually not necessary for power grid analysis. Worst case power noise occurs when

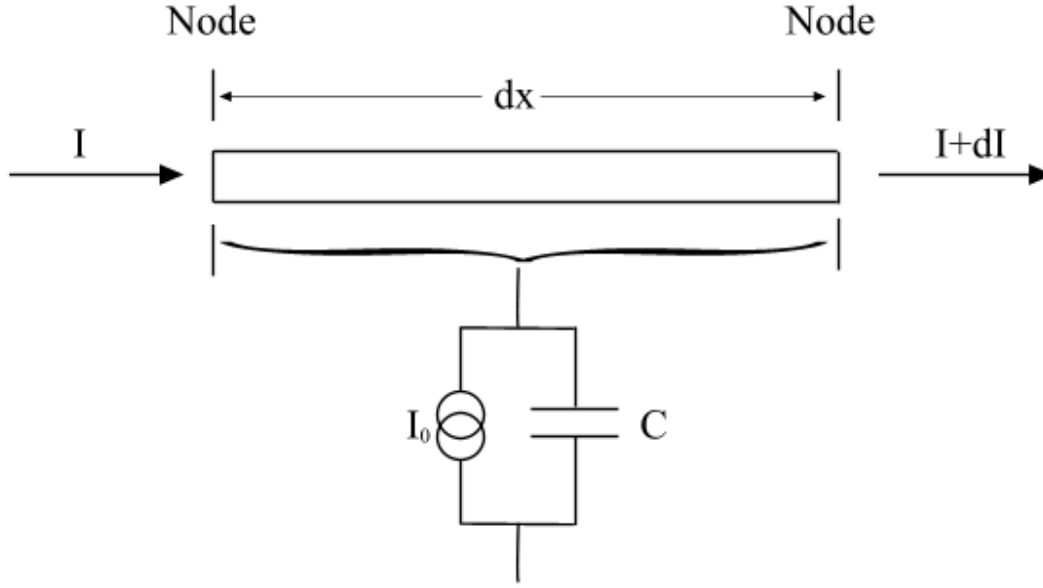


Figure 1.3. A short section of power grid wire showing the distributed capacitance and current source

large numbers of closely spaced gates switch simultaneously or circuit blocks are switched on and off (e.g. for power saving).

The current source is distributed uniformly across the block area. This approximation is valid for a block which is well connected, i.e. it has many (symmetric) tap points uniformly distributed across the block. The user also supplies a value of decoupling capacitance for each block. This capacitance is also distributed uniformly across the block area.

Dividing each transmission line pair into short elements the voltage change across each element is given by:

$$-dV = \left(\frac{dx}{w} \right) R_s I + L dx \frac{dI}{dt} \quad (1.1)$$

$$-dI = I_0 dx + C dx \frac{dV}{dt} \quad (1.2)$$

where dx is the length of the element, w is the wire width, R_s is the sheet resistance, L is the inductance per unit length of the wire, I_0 is current per unit length and C is capacitance per unit length. Efficient algorithms are used to solve these equations for all on and off chip transmission line pairs.

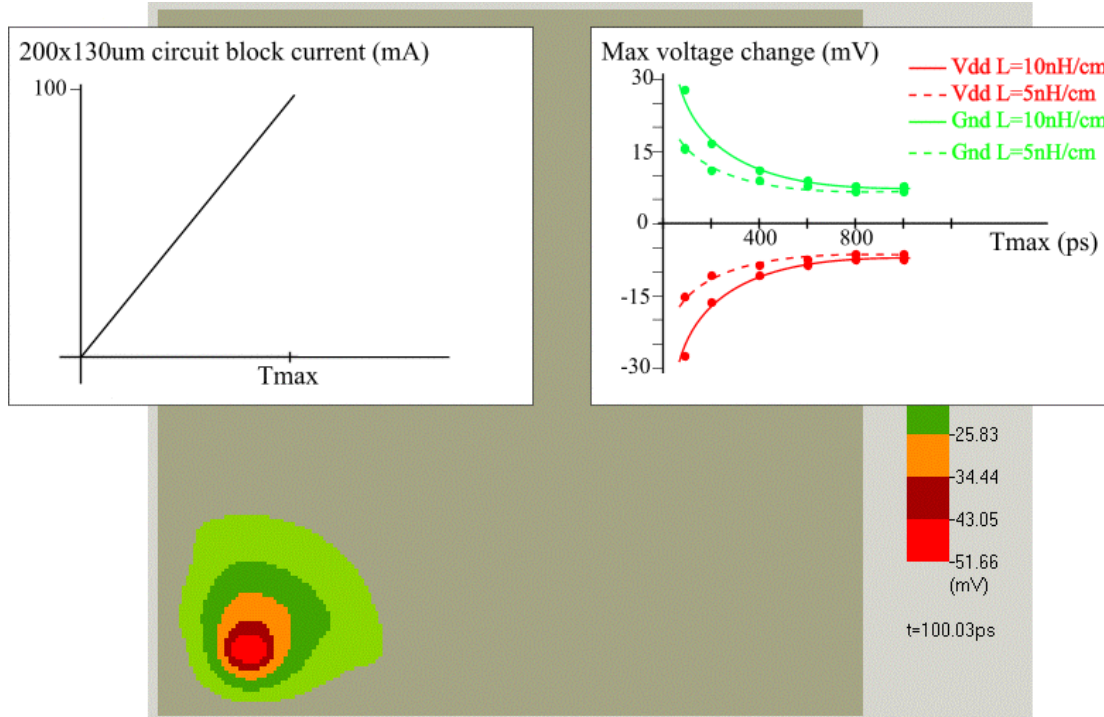


Figure 1.4 Simple II-FP result for a single IC and ideal voltage sources.

Figure 1.4 shows II-FP results for a single IC with ideal voltage sources at the chip boundary. This type of simulation can be used to get a quick estimate for on-chip grid dimensions and decoupling capacitance before running a complete system level simulation. The IC is 4x4mm in size. Each transmission line pair in the two layer global on-chip grid has 10um wide, 20mOhm/sq wires and each pair is spaced periodically at 80um. The inductance per unit length of each transmission line pair is 10nH/cm and the IC has a default de-cap per unit area of 5nF/cm². This default cap can be overridden at any location on the chip if required. Both inductance and resistance are frequency dependent between 1 and 10GHz so a number of simulation runs with different R_s and L may be needed to find the worst case power noise.

On and off chip decoupling capacitance are important parameters in any high speed power noise simulations. This can be obtained using an extraction tool or estimated from the number of transistors/gates and the wire density within a specified region. In a more realistic example a number blocks of varying size/cap would be defined in different regions of the IC to study the effect of on-chip de-cap on noise level and distribution. Note that this capacitance is the total capacitance between power and ground nets within the defined region. It includes all parasitics and any de-cap added intentionally by the designer. A simple linear ramp is defined for a single active current source in a small region of the IC. All other circuits are assumed to switched off. Note that inductance dominates voltage drop and ground bounce at high frequency (dI/dt). The voltage wave front propagates out from the current source at a velocity given by:

$$v = \frac{1}{\sqrt{sLC_A}}$$

where s is the periodic distance between transmission line pairs in the on-chip grid, L is inductance per unit length and C_A is the capacitance per unit area in the region

surrounding the source. These results also show clearly the effect of minimising inductance by reducing the space between power and ground wires in each transmission line pair. This is not necessary at low frequency where, as the graph shows, we get a frequency independent IR drop and ground bounce.

Figure 1.5 shows a result using a more realistic gate switching profile. This plot is output by the on-chip field solver and shows the maximum and minimum supply voltage change ($\Delta[V_{dd}-Gnd]$) across the chip surface. As the current ramps down the energy stored in the magnetic field surrounding on-chip power grid wires is returned to the grid causing an increase in supply voltage. This energy is, of course, available to power active circuits during subsequent clock cycles but it is not always available at the place and time you need it! Supply bounce (V_{dd} up, Gnd down) can cause damage to thin dielectrics, particularly if it is already close to its specified maximum value (e.g. for high speed operation). It also contributes to timing variations etc. Supply bounce can also be caused by resonance in the on-chip grid, package and elsewhere. All of these effects can be studied using II-FP .

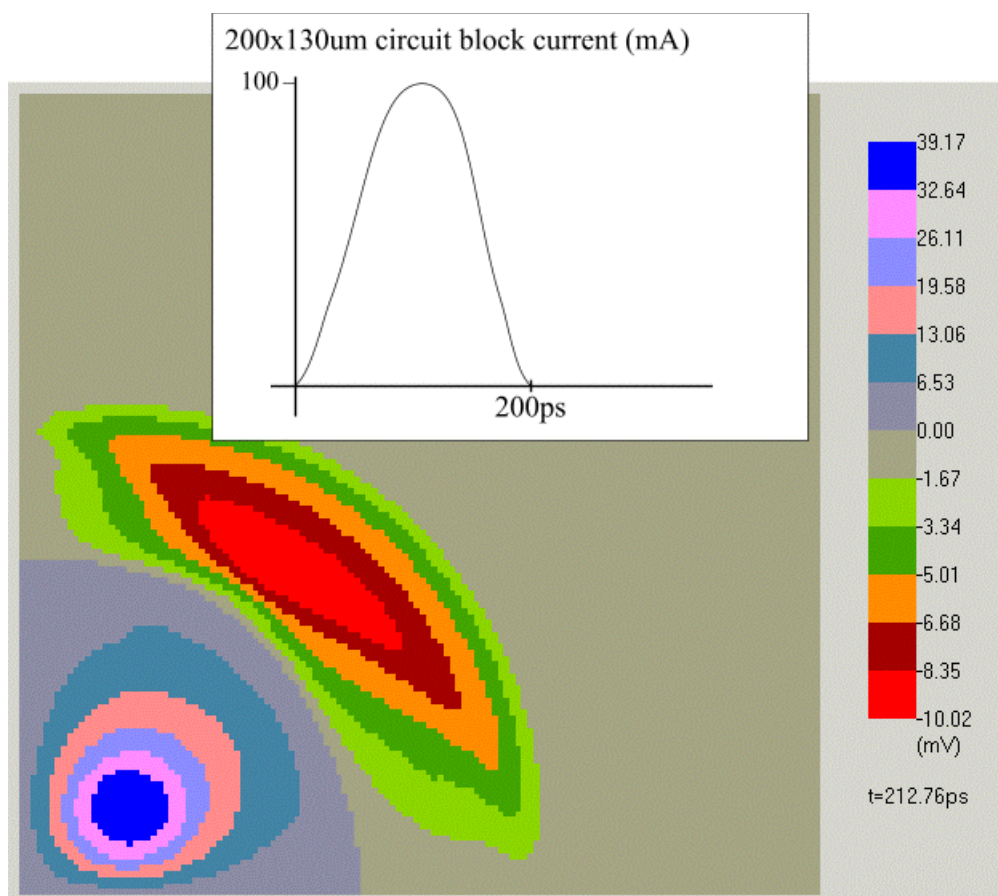


Figure 1.5. Power noise at $t=212ps$ showing voltage wave propagation across a $4 \times 4mm$ IC. The supply voltage dropped by a maximum of 62mV and increased by a maximum of 58mV during this simulation.

In these simple examples any energy which is not dissipated as heat in the on-chip grid is reflected from the ideal voltage sources at the chip boundary. When the IC is connected to the package and board level power network the results can be very different. These long transmission lines have much greater inductance and can give rise to powerful system level transients when blocks are switched in and out during normal IC operation. The system illustrated in figure 1.6. includes a package grid and 5 long board level connections to the power supply. The average current consumption of all active current sources in this simulation is 1.43A. A simple static analysis using this current distributed uniformly across the chip surface gives 29.5mV maximum IR drop. Figure 1.7. shows what happens when all the current sources shown are clocked at 2.5GHz at time $t=0$. It takes approximately 300 clock cycles for this system level transient to die out.

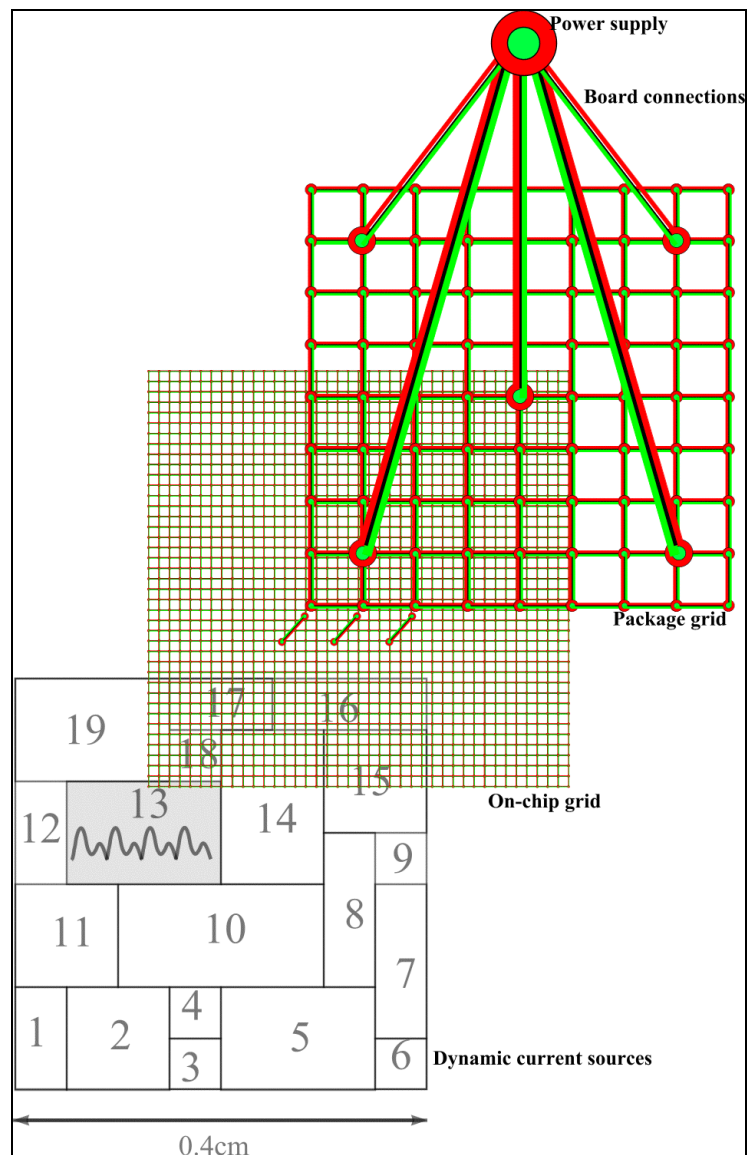


Figure 1.6. Example using a package grid and board level connections.

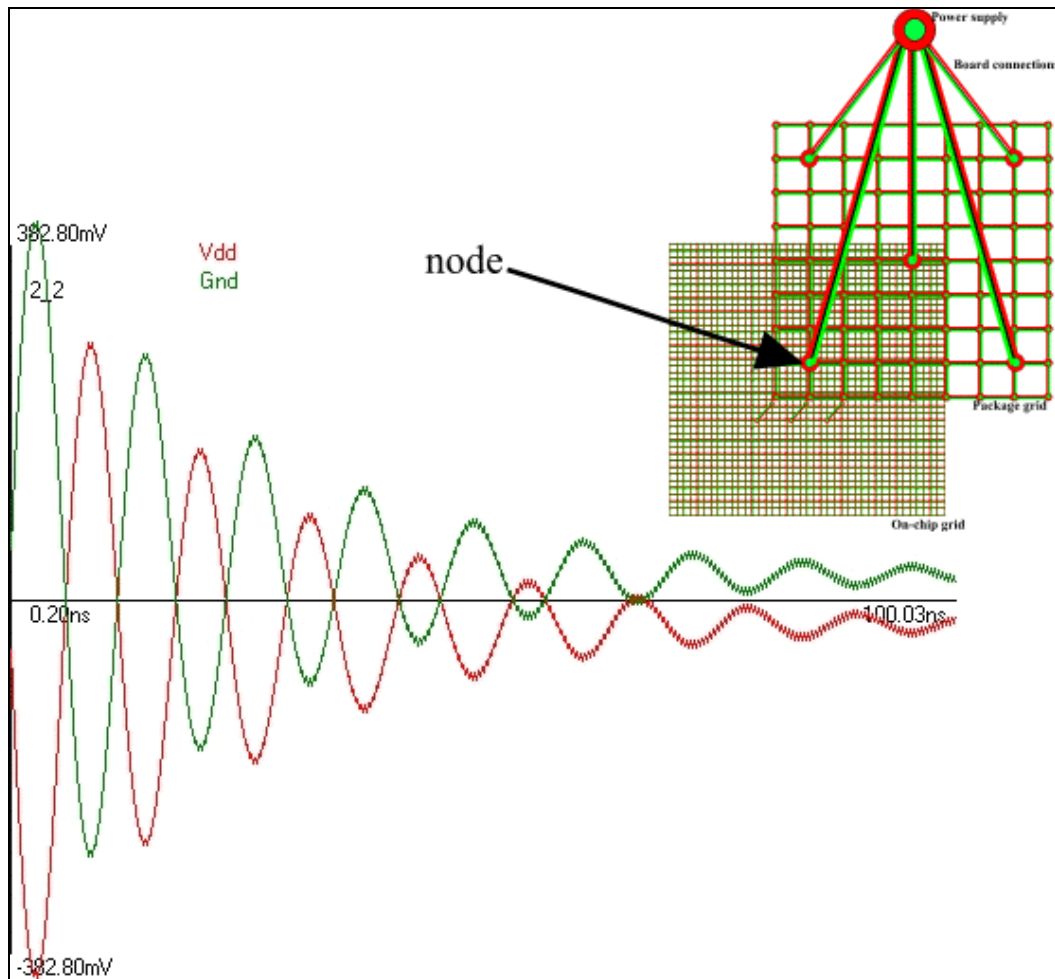


Figure 1.7. System level transient at the node indicated caused by the simultaneous switch on of many on-chip current sources.

Clearly measures have to taken to suppress these transients or prevent them from disrupting on-chip power supplies. Ideal or non-ideal decoupling capacitors can be added to the off-chip transmission line network in Π -FP, if required.

Chapter 2

Tutorial

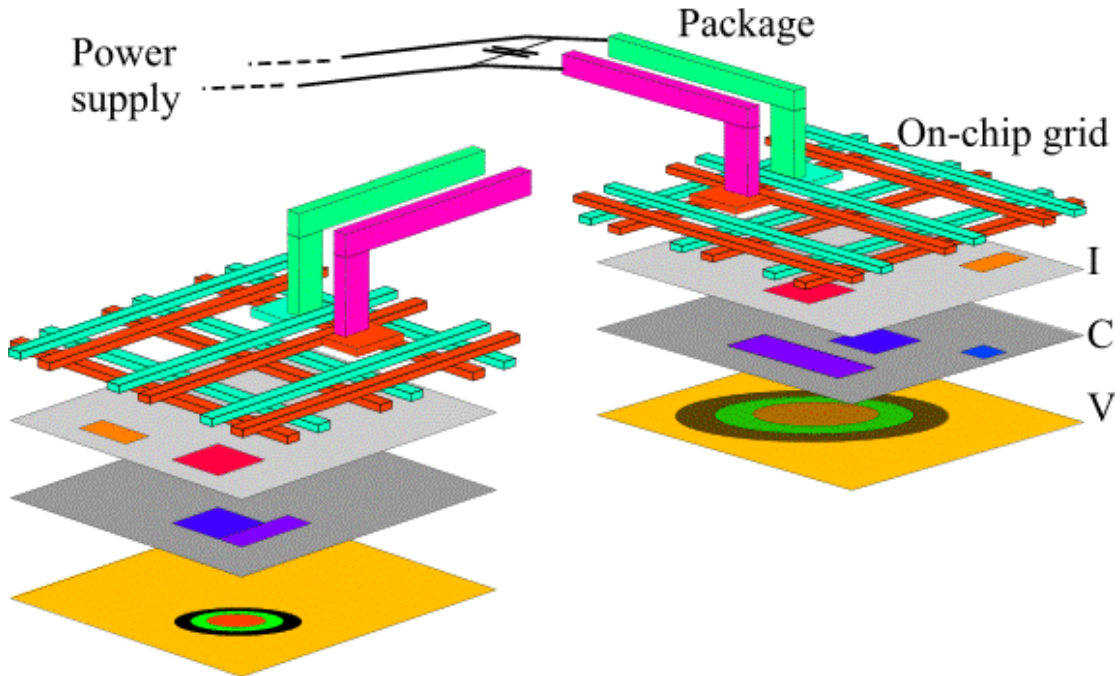


Figure 2.1. Multi-chip simulation

Π-FP is a multi-chip floor planning simulation tool. Any number of connected IC's, transmission lines and power supplies can be simulated. The starting point for any power noise analysis is the dynamic current sources connected to each on-chip grid. Figure 2.2 shows an example of the time varying current profiles used to model gate switching currents within a defined area of an IC. In Π-FP this model is implemented using the following statement in the netlist:

```
Ichip1 0.04 0.03 0.02 0.013 profile1.txt 40
```

In this statement the name “chip1” is the name of the IC containing this current source. The next four parameters are the location and dimensions of the source and “profile1.txt” is a file containing a piecewise linear representation of the profile. The final parameter is the number of times this profile is repeated during the simulation. For example, a simple linear ramp could be implemented using the following two lines in a separate text file:

```
0.0 0.0  
100e-12 0.0100
```

The profile can be used for multiple sources if required. In this example the current ramps to 100mA in 100ps. The block dimensions could be as small as a single gate but this level of detail is usually not necessary since worst case power noise occurs when large numbers of gates switch simultaneously within localised areas of the chip.

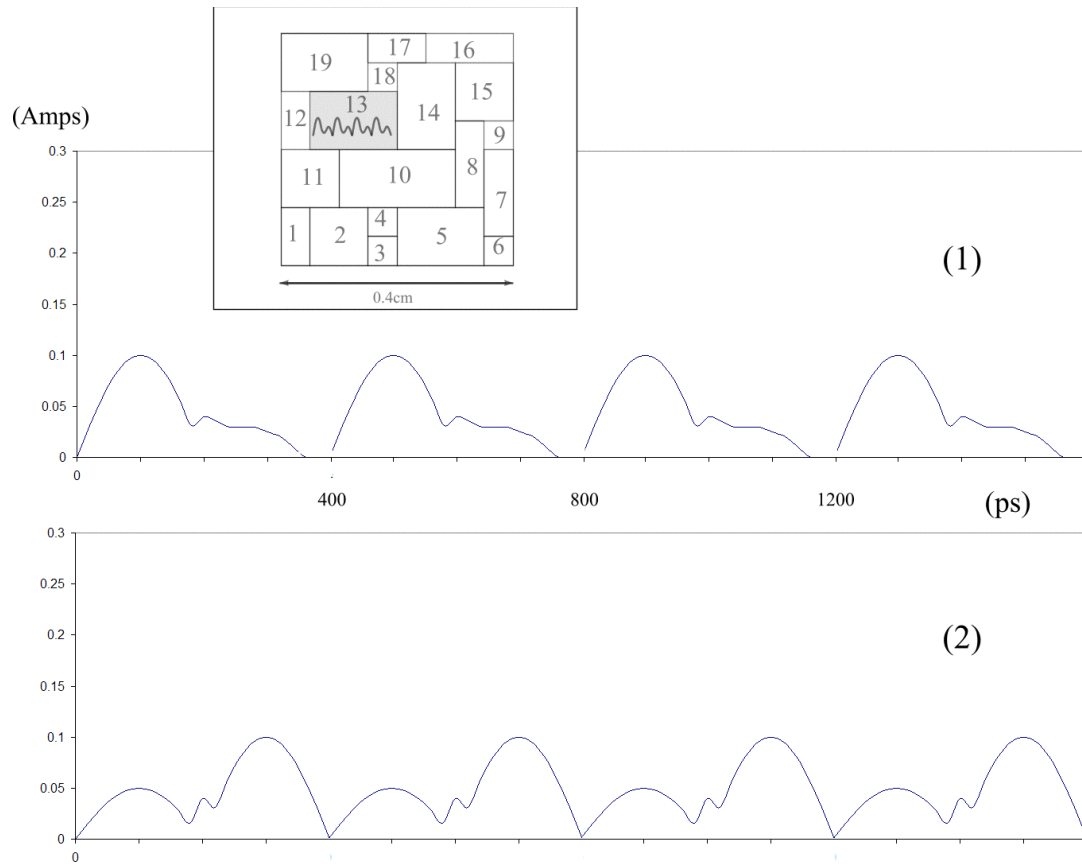


Figure 2.2. Dynamic current sources

Each current source is attached to an on-chip global power distribution grid. The grid is implemented using the following statement in the netlist:

```
Gchip1 0.4 0.4 0.0010 0.0080 0.020 10e-9 20e-9
```

The IC is 4x4mm in size. Each transmission line pair in the global two layer symmetric grid has 10um wide, 20mOhm/sq wires and each pair is spaced periodically at 80um. The inductance per unit length of each transmission line pair is 10nH/cm and the default capacitance per unit area is 20nH/cm². This default de-cap can be overridden at any location on the chip surface using the following statement:

```
Cchip1 0 0 0.2 0.2 2e-9
```

In this simple example 2nF is distributed uniformly across a 2x2mm area of the chip surface. In a more realistic example a number blocks of varying size/cap would be defined in different regions of the IC to study the effect of on-chip de-cap on noise level and distribution. Note that this capacitance is the total capacitance between power and ground nets within the defined region. It includes parasitics from all layers and any de-cap added intentionally by the designer.

So far no connections to the package, board or power supply have been added but we can run the simulation with just these three statements. In this case the tool adds ideal voltage sources to the on-chip grid boundary. Results of simulations using a linear

ramp for the current profile are shown in fig.1.4, chapter. 1 and a result using a 200ps current pulse is shown in fig.1.5, chapter 1. The wave velocity across a low resistance grid can be obtained from the equation for a lossless transmission line.

$$\frac{d^2V}{dt^2} = \frac{1}{LC} \frac{d^2V}{dx^2} \quad (2.1)$$

where L is inductance per unit length and C is capacitance per unit length. Substituting the wave

$$V = e^{j(\omega t + kx)} \quad (2.2)$$

into (2.2) gives the wave velocity

$$v = \frac{\omega}{k} = \frac{1}{\sqrt{LC}} \quad (2.3)$$

The capacitance per unit length associated with each power grid wire is (figure 2.3)

$$C = sC_A \quad (2.4)$$

where C_A is the capacitance per unit area in the region surrounding the wire and s is the wire spacing.

Combining (2.3) and (2.4) gives

$$v = \lambda f = \frac{1}{\sqrt{sLC_A}} \quad (2.5)$$

where λ and f are the wavelength and frequency of the power grid noise.

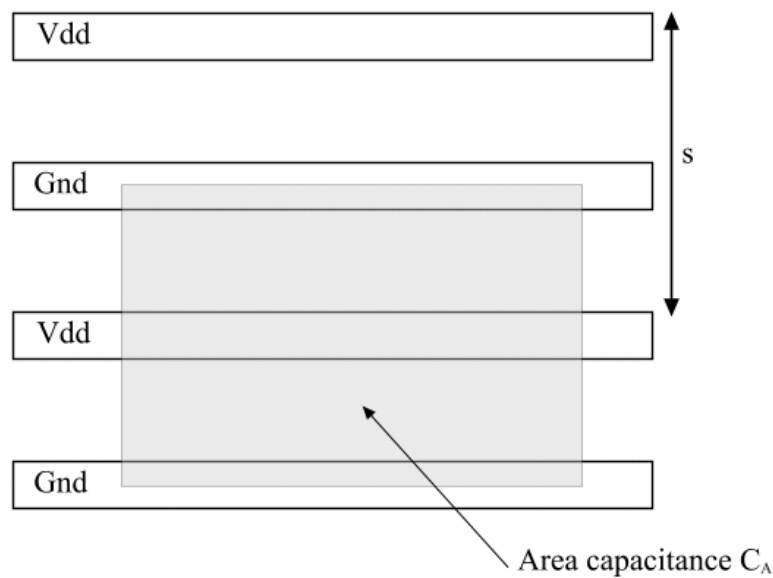


Figure 2.3. Relationship between C and C_A in a power grid.

In Π -FP, transmission lines and lumped decoupling capacitors model the power network in each IC package, all board level connections and the power supply. Figure 2.4 shows a single transmission line pair. Power and ground wires have the same resistance per unit length thus the magnitude of any IR drop is equal to ground bounce

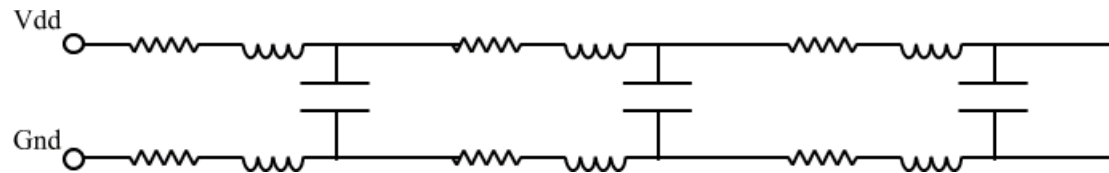


Figure 2.4. A transmission line

at each point on the line. The inductance of the pair includes magnetic flux from both power and ground wires. This flux should be minimised by routing the wires at minimum space wherever possible. Designs using singly routed power or ground wires are not allowed. Unpaired power grid wires couple strongly to other nets in high speed systems so these are best avoided at the design stage.

A transmission line is included in the netlist using:

```
Tname node1 node2 R L C length
```

Where R,L and C are per unit length. Only two node names are required in the transmission line definition. Power and ground wires must always be routed together so there is no need for separate node names for supply and return nets.

The electromagnetic field solver calculates potential changes everywhere on the chip, including the spaces containing dielectric between each wire. This means that there is no on-chip network and set of nodes at which to connect the external transmission lines. We need to create a node for each transmission line to connect to an on-chip grid. This done using:

```
Nchip1 node1 0.04 0.05
```

This node is located at x=0.4mm, y=0.5mm on chip1. The node name should correspond to a node in the package/board transmission line network. When a transmission line is added to the netlist all ideal voltage sources are removed from on-chip grids and the tool operates in full system mode. Figures 2.5 and 2.6 show results with one transmission line connected to this node. As expected the on-chip supply voltage continues to drop with each clock cycle. A single transmission line is not enough to power this IC even if only one small area of the circuit is active. A large lumped de-cap is used to model an ideal voltage source. It is added to the right hand side of the transmission line.

```
L1 node2 1e-3
```

Since there is no current flow to ideal voltage sources at the chip boundary all voltage contours are now normal to the chip edge. Initial and subsequent voltage drops are significantly higher than they were with ideal sources in place.

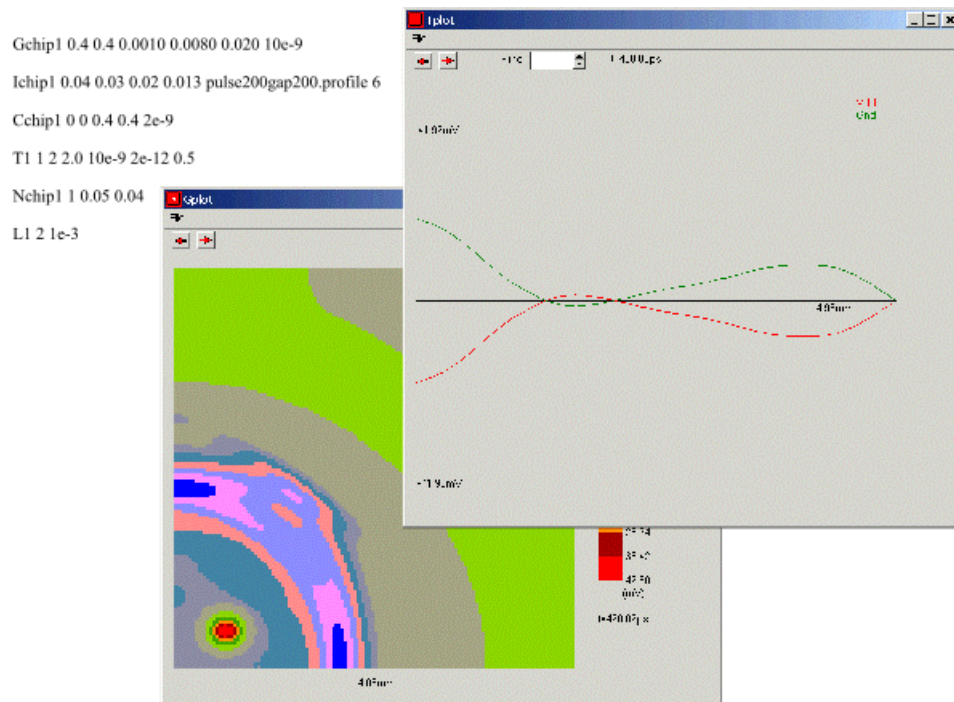


Figure 2.5. On-chip and external transmission line results at $t=420\text{ps}$.

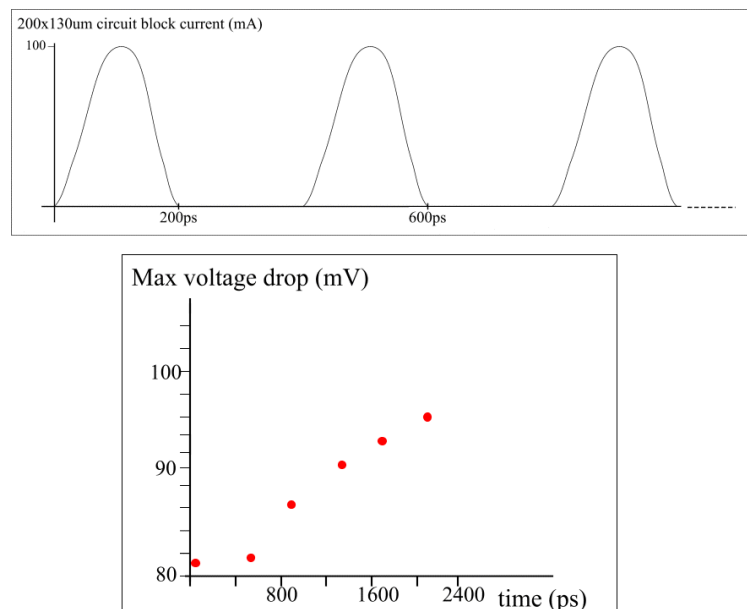


Figure 2.6. A single transmission line is not enough to re-charge the on-chip grid quickly enough.

Chapter 3

Netlist and parameters

This chapter describes each netlist statement and associated parameters.

Netlist # (comment)

Lines beginning with # are assumed to be comments and are ignored.

Netlist: Global on-chip grid definition

Gchipname <chip_width> <chip_height> <tline_wire_width> <tline_periodic_space>
<sheet_resistance> <inductance> <Default_cap>

Parameters

Chipname	A unique name for the IC.
Chip_width	Width of the IC (cm).
Chip_height	Height of the IC (cm).
Tline_wire_width	Width of supply and return wires in the two layer global power distribution grid (cm).
Tline_periodic_space	Distance between the center lines of supply/return transmission line pairs in the on-chip global grid (cm).
Sheet_resistance	Sheet resistance of wires in the two layer on-chip grid (Ohms/sq).
Inductance	Inductance per unit length of on-chip transmission line pairs (H/cm).
Default_cap	Default capacitance per unit area in regions where no capacitance blocks have been defined (F/cm ²).

Example

Gcpu1 0.35 0.38 0.0020 0.0090 0.025 8e-9 12.5e-9

Netlist: On-chip current sources

Ichipname <x_location> <y_location> <width> <height> <filename> <repeats>

Chipname	The name of the global on-chip grid defined elsewhere in the netlist.
X_location	Horizontal distance between the left chip edge and the left edge of the current source (cm).
Y_location	Vertical distance between the bottom chip edge and the bottom edge of the current source (cm).
Width	Width of the current source (cm).
Height	Height of the current source (cm).
Filename	File containing PWL current source profile
Repeats	The number of times this profile is to be repeated.

Example

Ichip1 0.04 0.04 0.02 0.02 /profiles/profile1.txt 20

Example PWL profile

```
0.0      0.0
20e-12   0.010
30e-12   0.015
...      ....
<time(s)> <current(A)>
etc.....
```

Netlist: On-chip capacitance

Cchipname <x_location> <y_location> <width> <height> <cap>

Chipname	The name of the global on-chip grid defined elsewhere in the netlist.
X_location	Horizontal distance between the left chip edge and the left edge of the capacitance block (cm).
Y_location	Vertical distance between the bottom chip edge and the bottom edge of the capacitance block (cm).
Width	Width of the de-cap block (cm).
Height	Height of the de-cap block (cm).
cap	Total capacitance within the block area (F)

Example

Cchip1 0.0210 0.0324 0.0133 0.0232 6.2e-12

Netlist: Transmission lines

Tname <nodename1> <nodename2> <R> <L> <C> <length>

Name	The name of the transmission line pair.
Nodename1	First node name.
Nodename2	Second node name.
R	Resistance per unit length (Ohms/cm).
L	Inductance per unit length (H/cm).
C	Capacitance per unit length (F/cm).
Length	The length of the transmission line pair (cm).

Example

T1 1 2 5 10e-9 10e-12 0.0100

Netlist: Connection node

Nchipname <nodename> <X> <Y>

chipname	The name of the on-chip grid to which this node will be attached
Nodename	A node in the transmission line network.
X	X coordinate of the node (cm).
Y	Y coordinate of the node (cm).

Example:

Nchip1 1 0.05 0.05

Netlist: Ideal cap

Lcapname <nodename> <cap (F)>

nodename	A node in the transmission line network.
cap	Lumped capacitance added to node (F).

Example

Lcap1 2 1e-9

Netlist: **.TRAN <simtime>**

Simulation run time (t=0 to t=simtime).

Example

To simulate the system from t=0 to t=120ps use:

```
.TRAN 120e-12
```

Netlist: Accuracy parameter.

Each transmission line pair is divided into short sections. The suggested length of this section is set using this parameter (default 0.0080).

Example

Divide each tline pair into 80um lengths.

.ACC 0.0080

Netlist: **.PLOT <nplots>**

The number of prints/plots during the simulation (equally spaced time intervals).
Default 10.

Example.

.PLOT 50

Netlist: .PRINTNODE <nodename|ALL>

Sets the node currents/potentials to print to standard output.

Examples.

To print nodes n1 and a2 use:

```
.PRINTNODE n1  
.PRINTNODE a2
```

To print all nodes use:

```
.PRINTNODE ALL
```

Chapter 4

Graphical output using GNUplot.

The simulator outputs voltage and current data for nodes, transmission lines and on-chip grids in text files with a .dat extension. A set of default GNUplot scripts is also created during the simulation. GNUplot is a general purpose plotting program available in various cross platform versions. It can be obtained from www.gnuplot.info.

The following netlist contains a single on-chip grid and a single transmission line pair. Node 1 of the t-line is connected to the on-chip grid at coordinates $x=1.1\text{mm}$, $y=1.1\text{mm}$. The piecewise linear current profile in the file pulse.txt is also shown.

```
.TRAN 200e-12
.PLOT 20
.ACC 0.0060
.PRINTNODE ALL
Ggrid1 0.2 0.2 0.0005 0.0080 0.030 10e-9 10e-9
Igrid1 0.1 0.1 0.02 0.02 pulse.txt 1
Ttline1 1 2 0.01 10e-9 100e-12 0.3
Ngrid1 1 0.11 0.11
```

pulse.txt

```
0 0
22E-12 0.030901699
40E-12 0.058778525
60E-12 0.080901699
80E-12 0.095105652
100E-12 0.1
120E-12 0.095105652
140E-12 0.080901699
160E-12 0.058778525
180E-12 0.030901699
200E-12 0
```

The following file is output during each simulation run:

node.gnuplot

In this simulation the file contains the following GNUplot script:

```

set size 1,0.5
set style data linespoints
set xlabel 'time(ns)'
unset mouse
# To plot a single node use one of the following in Gnuplot
#-----
set title 'node 1 delta(Vs-Vg)'
set multiplot
set origin 0,0.5
plot 'nodev.dat' using 1:2 title '(mV)'
set title 'net current out of node 1'
set origin 0,0
plot 'nodei.dat' using 1:2 title '(mA)'
unset multiplot
pause 1
#-----
set title 'node 2 delta(Vs-Vg)'
set multiplot
set origin 0,0.5
plot 'nodev.dat' using 1:3 title '(mV)'
set title 'net current out of node 2'
set origin 0,0
plot 'nodei.dat' using 1:3 title '(mA)'
unset multiplot
pause 1

```

This script can be used to plot all node potentials and currents at one second intervals. At a command prompt type:

(w)gnuplot node.gnuplot

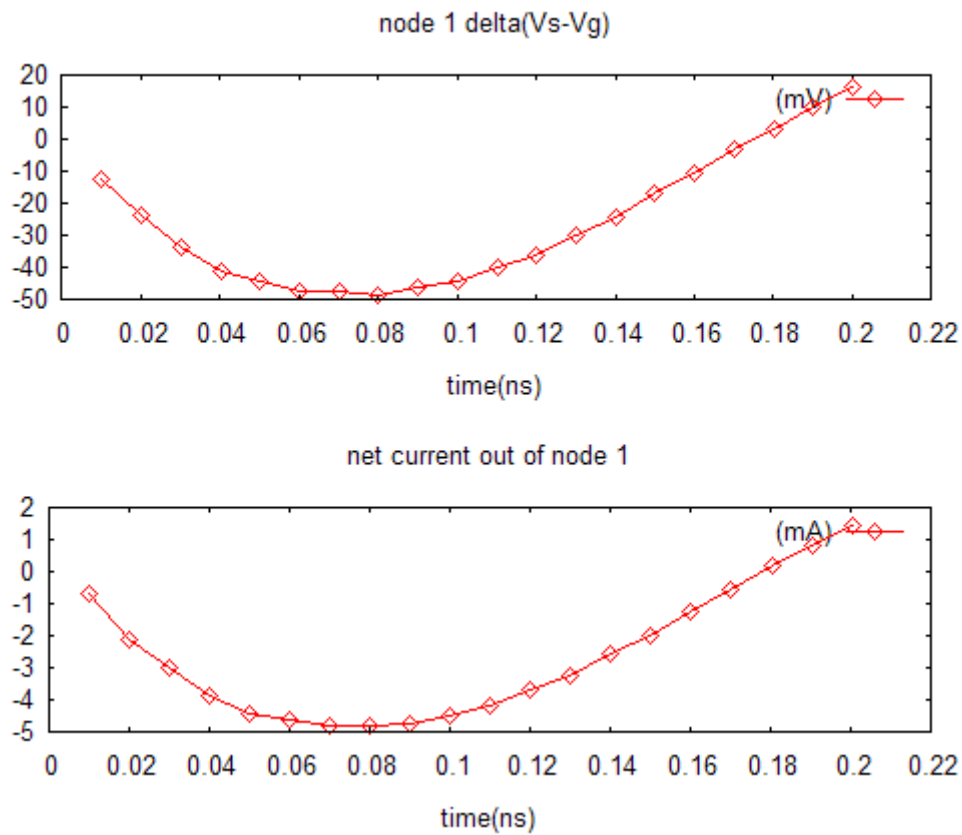
To plot a single node, remove all the other nodes from this script. For example to plot node 1 only use:

```

set size 1,0.5
set style data linespoints
set xlabel 'time(ns)'
unset mouse
#-----
set title 'node 1 delta(Vs-Vg)'
set multiplot
set origin 0,0.5
plot 'nodev.dat' using 1:2 title '(mV)'
set title 'net current out of node 1'
set origin 0,0
plot 'nodei.dat' using 1:2 title '(mA)'
unset multiplot
pause -1

```

GNUplot will create the following plot showing the supply voltage variation and the net current flowing out of this node.



The style of the plot can be changed by modifying the set statements at the start of the script. For example to use a mouse to zoom in and out of the plot remove the 'unset mouse' statement. See the GNUplot manual for more details.

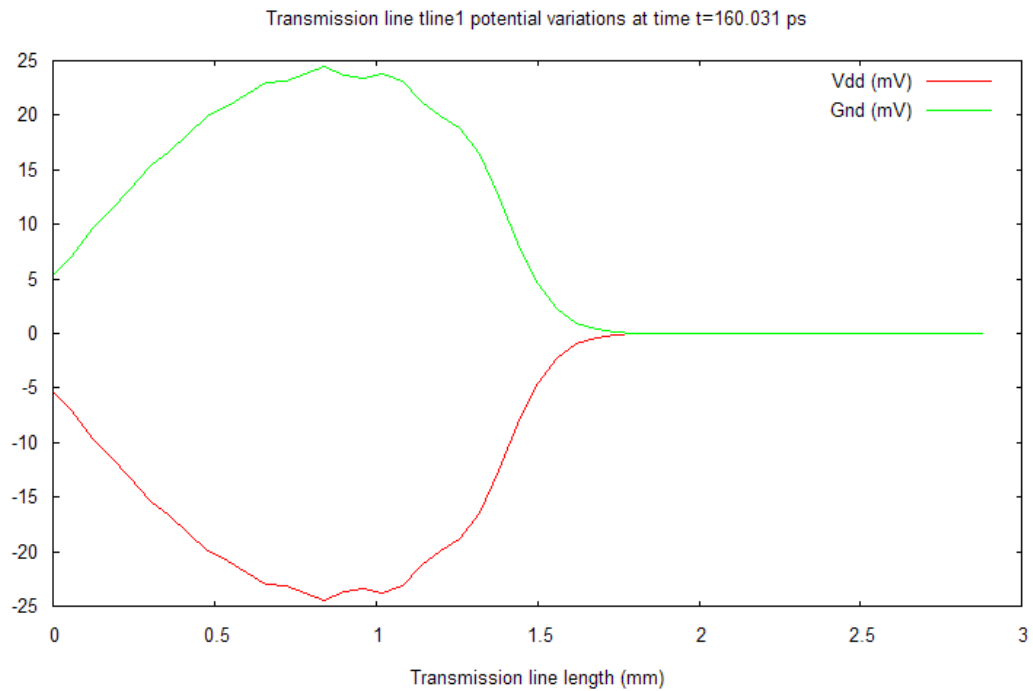
A GNUplot script is created for each transmission line. These files have the form:

<name>_tline.gnuplot

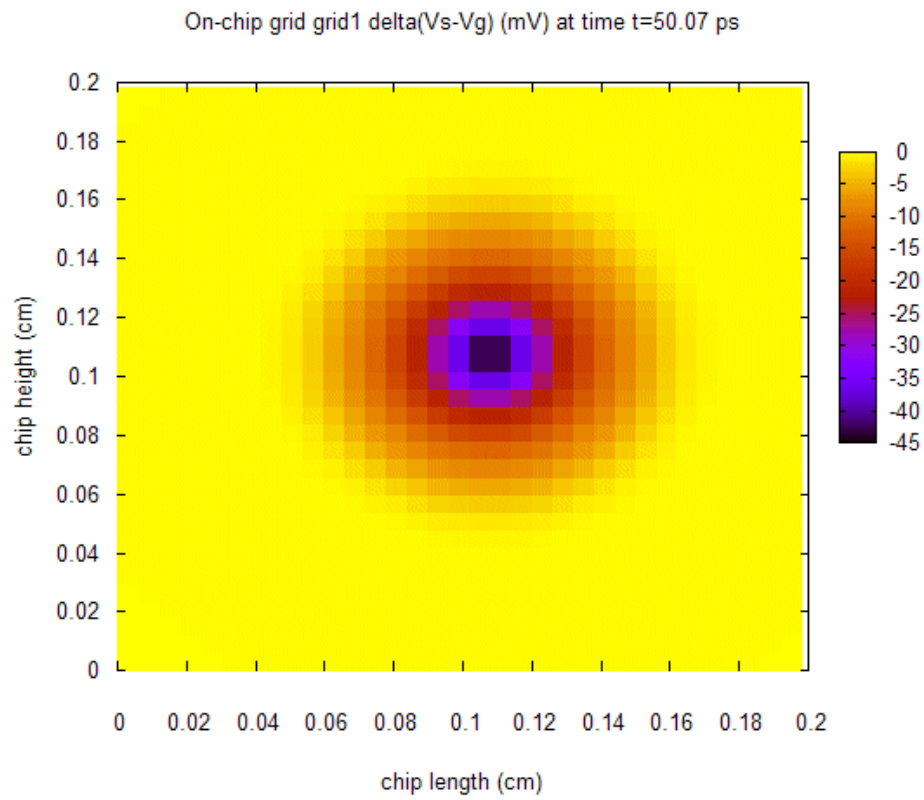
For example to plot the single transmission line in this system use:

(w)gnuplot tline1_tline.gnuplot

This will plot the transmission line potential variation at 20ps intervals. This script can also be edited to plot the power noise at a fixed time. For example:



On chip grid potentials can be plotted using the script <name>_grid.gnuplot. For example:



Chapter 5

Invocation of command-line version

The simulator (if compiled independently of the GUI) may be invoked by the command line:

```
<executable_name> <netlist_file_name>
```

For example, in Linux systems, the command line may look like:

```
pifp mychip.pfn OR </path/>pifp mychip1.pfn
```

where pifp is the executable name, and mychip1.pfn is the netlist output in simple text formatting from the tool GUI or compiled manually.