

# CHALMERS



## **Power Integrity Analysis of AVR Processor**

*Master of Science (MSc) Thesis in Integrated Electronic System Design*

**BADRI NARAYANAN RAVI**

Chalmers University of Technology

Department of Computer Science and Engineering

Gothenburg, Sweden, November 2010

The Author grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet. The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let Chalmers University of Technology a store the Work electronically and make it accessible on the Internet.

Power Integrity analysis of AVR micro-processor

Badri Narayanan Ravi

© Badri Narayanan Ravi, November 2010.

Examiner: Per Larsson Edefors

Department of Computer Science and Engineering  
Chalmers University of Technology  
SE-41296 Gothenburg  
Sweden  
Telephone + 46 (0)31- 7721000

Department of Computer Science and Engineering  
Gothenburg, Sweden, November 2010

# **Power Integrity Analysis of AVR processor**

Badri Narayanan Ravi

## **Abstract**

*In the deep submicron era, the power supply voltage to the logic devices should be well within the bound limits. The degradation of power affects the timing closure of the chip, thus affecting the performance. In this thesis, the analysis flow used in previous study is modified to make it generic to all series of AVR processors. By extracting a power grid model and subjecting it to extensive simulations, the issue causing the power degradation is analysed. Analysis results show that the cause for the first voltage drop is due to two factors, that is, current loads and resistive property of the metal wires. The first overshoot after voltage drop increases for wider metal wires compared to lesser width of metal wires. Nodes of the power grid having large voltage drop show that on-chip self inductance can no longer be ignored in designing power distribution networks for high frequency circuits.*

## Acknowledgements

I would like to thank my parents for their support and encouragement throughout my career. I am blessed to have a wonderful relationship with my brother who has been my inspiration right from my childhood.

I am grateful to my guru's for their support and certainly it wouldn't be easy to carry the thesis work without them.

- Prof Per Larsson Edefors
- Prof Lars Svensson
- Dr Daniel Andersson
- Dr Johnny Pihl

## **Dedicated to**

My Aunt – Jayashree  
and  
My GrandFather - Srinivasan

## CONTENTS

<b>ABSTRACT</b>		<b>i</b>
<b>ACKNOWLEDGEMENTS</b>		<b>ii</b>
<b>A</b>	<b>THESIS</b>	<b>1</b>
<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Power Integrity – Background	3
1.2	Background – Power Integrity Analysis	3
1.3	Motivation	5
<b>2</b>	<b>The Power Grid of AVR micro-Processor</b>	<b>7</b>
2.1	Geometrical Model	8
2.1.1	Nets	9
2.1.2	Metal layers and Types	9
2.2	Electrical Model	10
2.2.1	Metal Wires	10
2.2.2	Vias	11
2.2.3	Grid Capacitors	11
2.2.4	Bonding Wires	12
2.2.5	Logic Gates	13
<b>3</b>	<b>The Design Set-up: Power Integrity Analysis of AVR Processor</b>	<b>14</b>
3.1	Grid Extraction Phase	16
3.1.1	Matlab	17
3.2	Grid Component Extraction Phase	18
3.2.1	Raphael Field Solvers	19
3.3	Grid Simulation Phase	20
3.3.1	Spectre	21
<b>4</b>	<b>Results of Power Integrity Analysis</b>	<b>22</b>
4.1	Resulting Power Grid Model	23
4.2	Results from Simulation	25
4.2.1	Inductance Dominance in Package	27
4.4.2	Corner Cell - Typical	29

4.4.3	Corner Cell - Minimal	31
4.4.4	Corner Cell - Maximal	32
<b>5</b>	<b>Discussion</b>	<b>35</b>
<b>6</b>	<b>Conclusion &amp; Future Work</b>	<b>36</b>
6.1	Conclusion	36
6.2	Future Work	37
<b>7</b>	<b>References</b>	<b>38</b>
<b>B</b>	<b>APPENDICES</b>	<b>41</b>
A.1	readCompleteDefnets.m	42
A.2	sortlayers.m	42
A.3	nodesIntialStripes.m	43
A.4	findIntersection.m	43
A.5	addNodes.m, addVias.m	44
A.6	powerGridGeneratorFull.m	45
A.7	Raphael Field Solvers	45
A.8	subCircuitGenerator.m and wrapNetlist.m	46
A.9	generateSwitchingPatterns.m	47
A.10	generateSourcenetlist.m	47
A.11	wrapSpiceFiles.m	48

## LIST OF FIGURES

1	Schematic View of Power Grid	6
2	Power Grid – Geometrical Layout (FLORIDA Design)	8
3	Electrical Model of a section in Power Grid	10
4	Array of Vias – Intersection between metal layers	11
5	R-C-L Link – Bonding Wire	12
6	Logic Gate as PWL Current sources	13
7	Power Grid Analysis Flow	15
8	Analysis Flow – Grid Extraction Phase	16
9	Analysis Flow – Grid Component Extraction Phase	19
10	Analysis Flow – Grid Simulation Phase	20
11	Geometrical Model – DELTA chip	23
12	Geometrical Model – FLORIDA chip	23
13	Interlayer <i>Vias</i>	24
14	Voltage Waveform of two Nodes	25
15	Power Grid Model with Multiple Current Loads	26
16	Intermediate Node Voltage of Bonding Wire	27
17	Output Voltage Waveform of Bonding Wire	28
18	Output Voltage Waveform	29
19	Volt Drop Vs No. of Nodes	30
20	Output Voltage Waveform	31
21	Volt Drop Vs No. of Nodes	32
24	Output Voltage Waveform	33
25	Volt Drop Vs No. of Nodes	33
A.1	Spectre Simulation	48



**LIST OF TABLES**

1	Contents in pg.def	8
2	Metal Layers and their Types	9
3	Parts in Electrical Model	10
4	Different Timing Cases	27
5	Parameters of Different Cornel Cells	34

**A**

**Thesis**

# 1

## Introduction

In the Deep Sub-micron era, as technology scales down, issues involving interconnects becomes a more dominant factor than that of devices. However, there even exist examples where the clock interconnects alone consume  $\sim 40\%$  of total power consumption [1], and thus efficient modelling of interconnects (signal and clock) with regards to its behaviour and usage is required.

With the greater functional integration resulting in increased device-count, smaller process geometries and decreased operating voltage of devices, the importance and need of power grid distribution networks in order to supply a stable voltage become larger and larger.

The main aim of this thesis is to modify an analysis flow used in [6] and implementing it on a new AVR micro-processor. This includes the grid extraction (translation of chip geometry to electrical model) and simulations which determine the parameters governing the response of the power grid.

## 1.1 Introduction to Power Integrity

One of the biggest challenges the VLSI industry is facing in today's technology is the issue of power integrity.

**Power Integrity Definition:** Assume, input ' x ' is fed at the source and output ' y ' is tapped at the destination. The input, which is the supply voltage, is being propagated through a grid but still via bond wires and output will be directed to the rails of Std-cells or logic devices. The source can be bond pad and the destination could be the nodes of Vdd grid to which the devices are connected. The value at the output (y) should stay within the reasonable boundary levels. In general, the output equation is given as:

$$y = x - (z\% \text{ of } x)$$

Where, the value of ' z ' determines the grid response. It is very important to keep value of ' z ' within the tolerance limit. What constitute the ' z ' will be answered in this thesis. The value of z is determined by two major properties, one being the power grid elements which constitute the power grid and other being the switching activity of devices (current demands).

In this thesis, the power integrity analysis with respect to the IC core level is demonstrated by performing circuit simulations on a real-time power grid model from one of Atmel's AVR processors.

## 1.2 Background - Power Integrity analysis

As the transistor size and supply voltage scales down, the power consumption of individual device reduces but the total power consumption, due to high speed switching, increases. At 90 nm and below, leakage power is catching up with the dynamic power and is becoming significant in newer technology nodes [2].

The following paragraph describes the challenges in power grid analysis and the work done in aim to mitigate the issues. In order to analyse the power integrity, predicting power dissipation accurately is a big challenge.

The initial method used was based on SPICE circuit simulator [3, 4, 5]. The technique used is based on pattern-dependent circuit simulators which are very slow to be used on modern day designs and are computationally insufficient. In order to improve the computational efficiency, simulation based techniques using timing, switch level and logic simulations in [7, 8, 9] respectively were proposed. The techniques are still slow for today's technology, where the simulation of entire chip is not possible but they are indeed faster than traditional circuit simulators.

In [10], non-linear devices (transistor network) are modelled as a time varying current sources. For the inverter case shown in [10], when PMOS is on, it acts decoupling capacitance to other switching circuits. The charge sharing effect is not present in the model used in [10] which results in over-estimation of the amount of de-cap. The model used in [20] relies on the model used in [10], but the difference is that transistor networks are replaced by a switch in series with an RC circuit as shown in Figure 2 in [20] where the effect of charge sharing is captured. The analysis result shows that size of de-cap in [20] is 2 – 8 times lesser than the size of de-cap in [10].

Due to increase in switching speed in VLSI circuits, the number of cells switching simultaneously in a short span of time increases causing huge amount of noise in power distribution network. The power supply noise is the drop in supply voltage that is due to resistive elements in the grid called - IR drop (resistive noise). Apart from resistance, on-chip capacitance also plays a dominant role in power supply noise.

A study on Pentium processor [11] shows that power supply noise can reduce clock frequency by 6.5% and 8% on 130 nm and 90 nm node respectively compared to a case where V<sub>dd</sub> is constant and nominal. When circuits operate at high frequency, the voltage drop is also caused by series inductance because of switching activity of the gates causing change in current which generates EMF equal to  $L \cdot di/dt$ . This is called simultaneous switching noise [12]. The inductance causes ringing effect or oscillatory behaviour.

Determining the worst case test vectors required in simulating an entire chip network to analyse power –ground (PG) noise is not straightforward. As the entire approach needs lot of data and memory it is not possible in today's SPICE simulators to handle the complexity in terms of runtime and storage capacity.

Today's industrial tools such as Magma (Blast rail) and Synopsys (Prime rail) etc. set their pre-defined power/ground noise margins to large value. They take only the IR drop for power noise analysis which gives pessimistic analysis results which leads to designs with less performance. It is not certain that worst case IR drop ensures worst case drop in supply voltage [6].

Also, the increase in current demands and the reduction in supply voltage have been major challenges in designing high performance distribution networks [13, 14, 15]. The demand for high performance and low power necessitates that modern day designs are characterized by the combination of increased functional integration, decreased process geometries and high clock speeds [15].

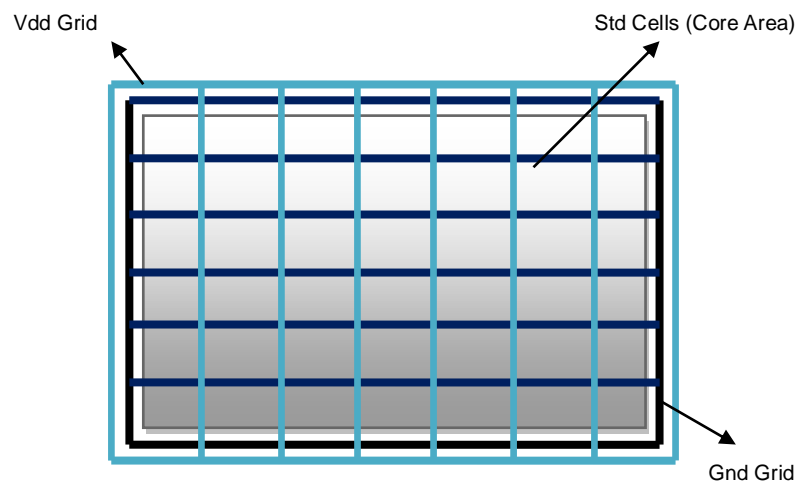
Traditionally, power grid distribution networks in high performance circuits are implemented as a uniform structure [16]. The effective resistance in each arm of uniform grid structure is same. In [17], the effective resistance between any two nodes is formulated in an infinite grid and the mathematical expression obtained in [17] is used in [16]. A few researches have been done in analysing the power with uniform grid structure. In [16], analysis on uniform power grid structure is performed with a single power supply and one current load connected to a node. The connection is done arbitrarily. The result shows that the maximum error obtained compared with SPICE simulations is about 1.44 mV which is less than 0.2% of supply voltage. The type of structure used in [16] is a mesh.

In this thesis, the grid extracted is non-uniform in structure and simulations are done on it with multiple current loads.

### **1.3 Motivation**

In modern day design, separate layers are devoted to Vdd and Gnd. The Vdd and Gnd plane are together known as power planes. The devices and wires of the standard cells are all below both the Vdd and Gnd grids as shown in Figure 1. The structure of Gnd plane is the mirror image of Vdd plane and vice versa. But, in an IC, power distribution is generally done with stacked metal layers at the top, which is connected to the package and active devices at the bottom.

- As the technology scales down, the voltage for the devices to operate also scale down. The power (Vdd and Gnd), acts as the reference voltage for the logic levels on which the devices operate. If Vdd layer fails to be steady within the tolerance limit, delay in gate switching increases, thus affecting the functional modules to deviate from their desired performance [12].
- Due to the voltage de-gradation, the delay in gate switching increases and transition activities across the device plane vary. This causes varying current injections to the ground thus resulting in ground spiking (voltage gradient) across the ground. Now, ground will not be a source of absolute potential.



**Figure 1:** Schematic View of Power Grid

## Thesis Outline

In order to have desired performance, the power integrity issue should be controlled. By performing circuit simulations on an AVR power grid, it would be interesting to see how different parameters have their impact on voltage drop.

The task in this thesis is to modify an analysis flow in [6] so that manual fixing of wires used in [6] in order to make the grid complete is avoided. Further, implementation of the flow is carried out on a new AVR processor, thus making the flow generic to any series of Atmel's AVR processors when compared to the flow used in [6] being specific to a particular type of IC chip called DELTA chip. In this thesis, to make the simulations feasible, the ground is set to be ideal zero.

# 2

## The Power Grid of AVR micro-Processor

The Power grid is a metal plane like structure where Vdd – no longer a plane with an absolute voltage value and Gnd - no longer a source of absolute zero potential. The power grid is a complex system with different parameters and it would be interesting for the chip designers to see how different design parameters impact and limit the performance of the design. Consequently, power grid research activities are required, that is, design, modelling the power grid, analysis and verification [20].

In this thesis, for analysis purpose, the power grid used is a real time application from Atmel AVR processors called FLORIDA chip. This chapter describes how complex the on-chip power grid is and how the power grid components and the power grid as a whole are modelled.

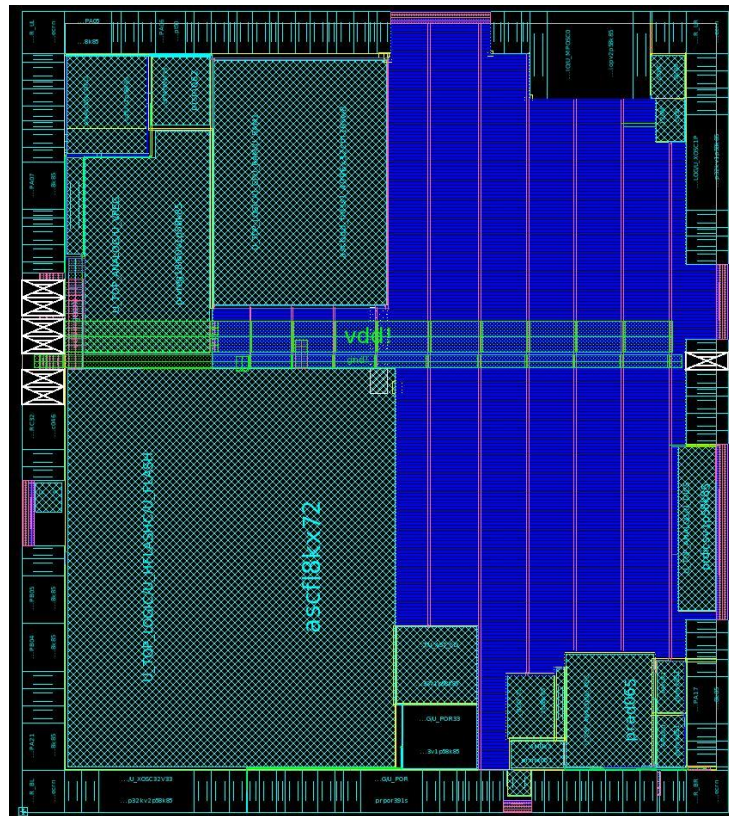


## 2.1 Geometrical Model

Knowing the target level of power density and noise threshold, it is straightforward to generate a basic grid image. As the design is refined many times the final result, the geometrical layout, of the power grid rarely looks like the initial image of the grid. The file pg.def, power grid geometrical layout definition file, basically defines the resources to be used for the power grid model extraction. The contents in Table 1 represent list of geometrical data available in pg.def. Figure 2 represents the power grid layout in geometrical form of FLORIDA design.

**Table 1:** Contents in pg.def

Nets	Chip Dimensions	Vias Co-ordinates
Metal layers	Process technology	Via Resistance
Types of metal layers	Wire Segments Co-ordinates	Decoupling Capacitors
Width of metal layers	Thickness of metal layers	Geometrical Coordinates Units



**Figure 2:** Power Grid – Geometrical Layout (FLORIDA design)

### 2.1.1 Nets

Types of net or grid available in pg.def are Vdd, Gnd and Vdd\_incore. The geometrical layout of Vdd and Gnd grids are identical as they are mirror images of each other. In this thesis, the net used is Vdd while Gnd net is assumed to be ideal at 0V. The idea behind considering the Vdd grid alone is to keep down the complexity of power grid model for the circuit simulators.

### 2.1.2 Metal Layers and Types

In this thesis, metal layers or wires used in the AVR power grid are shown in Table 2. Each metal layer of the power grid is divided into different types, as shown in Table 2. Apart from the types shown in Table 2, in the file pg.def, all the metal wires are further divided into *ring* and *blockring*.

**Table 2:** Metal layers and their types

Metal Layers	Types
M1	Stripe, Follow-pin
M3	Stripe, Follow-pin
M4	Stripe, Follow-Pin
M5	Stripe, Follow-Pin

In the previous study performed by Daniel Andersson [21], the only type used in extracting the power grid of DELTA design was *stripe* so the resultant power grid was an incomplete model. To make the power grid model more complete, the grid was further refined by Bjorn Nilsson in [6], by adding additional types such as *ring* and *blockring* for the metal layers of M1, M2 and M5.

## 2.2 Electrical Model

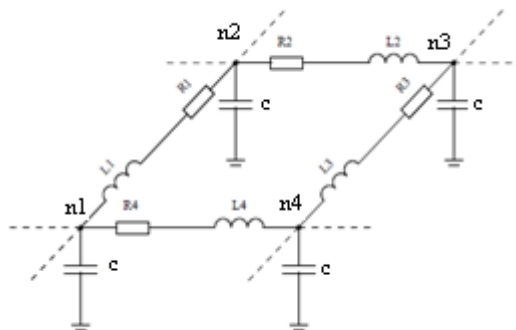
The electrical description of the geometrical model, the power grid, is used to carry out the simulations with other electrical components, shown in Table 3. This will be discussed in this section.

**Table 3:** Parts in Electrical Model

Metal Layers	Bonding Wires (R-C-L Link))
Vias	Logic Gates (modelled as current sources)
Grid Capacitors	Voltage Supply

### 2.2.1 Metal Wires

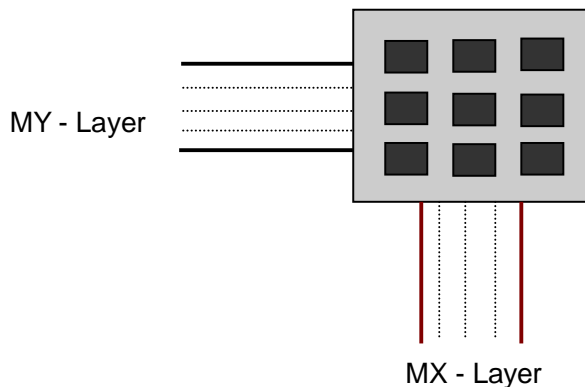
Figure 3 presents electrical model of a section of the power grid. A wire segment is made up of R and L components, which are placed between two nodes, for example, n1-n2. A power grid is designed with one or more metal layers. Each metal layer may have one or more wire segments, and connection of two or more wire segments forms a network. Each wire segment has electrical properties such as resistance, inductance and capacitance. At high frequency the additional component the on-chip inductance is important ignored for the analysis due to decrease in rise time of transient current and the emergence of low-inductance packaging technologies [22].



**Figure 3:** Electrical Model of a section in Power Grid – Schematic View

## 2.2.2 Vias

The *vias* are used to connect one metal layer to another. *Vias* are modelled as resistors and the value of *via* resistance is given in the process data file *pg.def*. They are basically used as arrays of *vias* in parallel fashion as shown in Figure 4. The value of each *via* in the array is calculated by dividing *via* resistance, given in *pg.def*, with number of *vias* in the array.



**Figure 4:** Array of Vias – Intersection between metal layers

## 2.2.3 Grid Capacitors

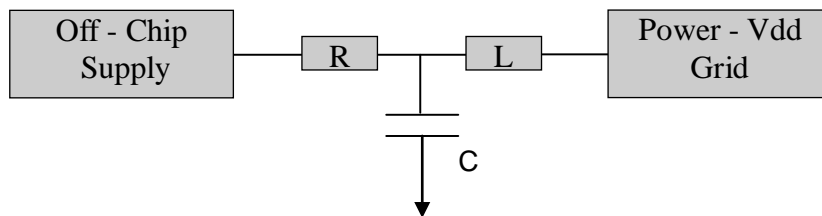
The grid capacitance at each node, modelled using the estimated amount of decap per unit area, is shown in Figure 6. Basically, decoupling capacitors are used to break a part of circuit,  $n_1n_2$  in Figure 3, from another,  $n_2n_3$ . This is done to suppress the noise effect caused by a circuit on other circuits. When logic switching activity of the gates takes place, the instantaneous current drawn by the gate is large and produces the drop in voltage due to the L and R of wires. The local charge stored in the decaps is used to suppress noise, by avoiding having to fetch charges via the L and R of wires.

The idea behind to determine the amount of decoupling capacitor is: First, the grid is divided into  $x * x$  squares and an estimated amount of decap per unit area, given in *pg.def*, is distributed equally among all the squares (*Decap in a Square = Total decap / No of Squares*). Second, the amount of capacitance in a square is divided by the nodes of the square which has the switching logic results in decap for a node. Thus, a

square containing switching logic has decap and a square without any switching logic does not have decap at all. This suggests that the more the nodes in square having switching logic, the less is the decap value for a node in that square.

## 2.2.4 Bonding Wires

The bonding wire, a type of packaging technology, is a medium between the off-chip power supply and the supply voltage pin on the power grid. A bonding wire is modelled as an R-C-L link as shown in Figure 5.



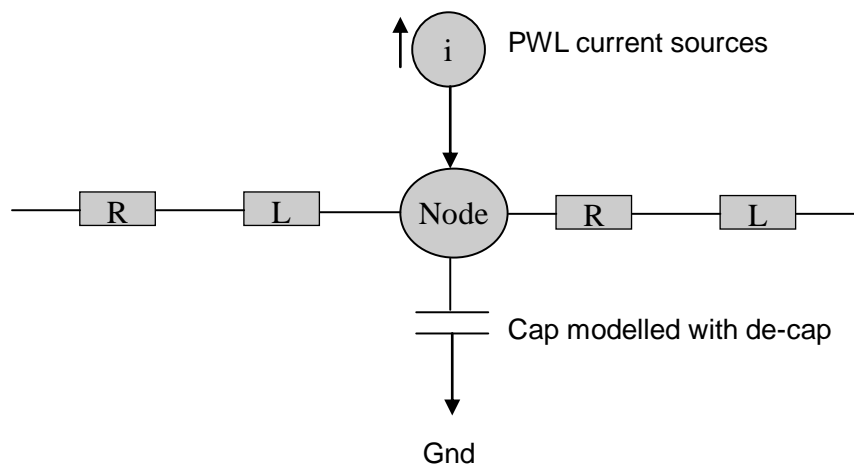
**Figure 5:** R-C-L Link – Bonding Wire

All packages have an effect on the performance of the IC because the electrical properties of the package are in the form of parasitic elements such as capacitive coupling between leads, and resistance and inductance in the leads [22]. In this thesis, the number of supply pins is two and the number of bonding wires connected to the power grid is two.

With the increased functionality of the chip and the demand for low power and high performance designs, the package required becomes complex in terms on pins, signal integrity, reliability etc [22].

## 2.2.5 Logic Gates

The drop in supply voltage at the grid nodes is due to electrical components of the metal layers and the load circuitry. The load circuit (logic gates) are modelled as time varying independent current sources as shown in Figure 6. Each current source is nothing but the superposition of current profiles of all the gates connected to a node. The simulation on power grid model is done with multiple current loads as shown in Figure 15. The gate, to be modelled as current loads, used in this thesis is *NAND* gate. To model the gate a lot of data is required such as geometrical position of the gate that needs to be coinciding with grid model geometrical position, switching activities, current waveforms, load capacitance etc. The information on how these data are obtained is explained in Chapter 2 of [6].



**Figure 6:** Logic Gate as PWL Current sources

# 3

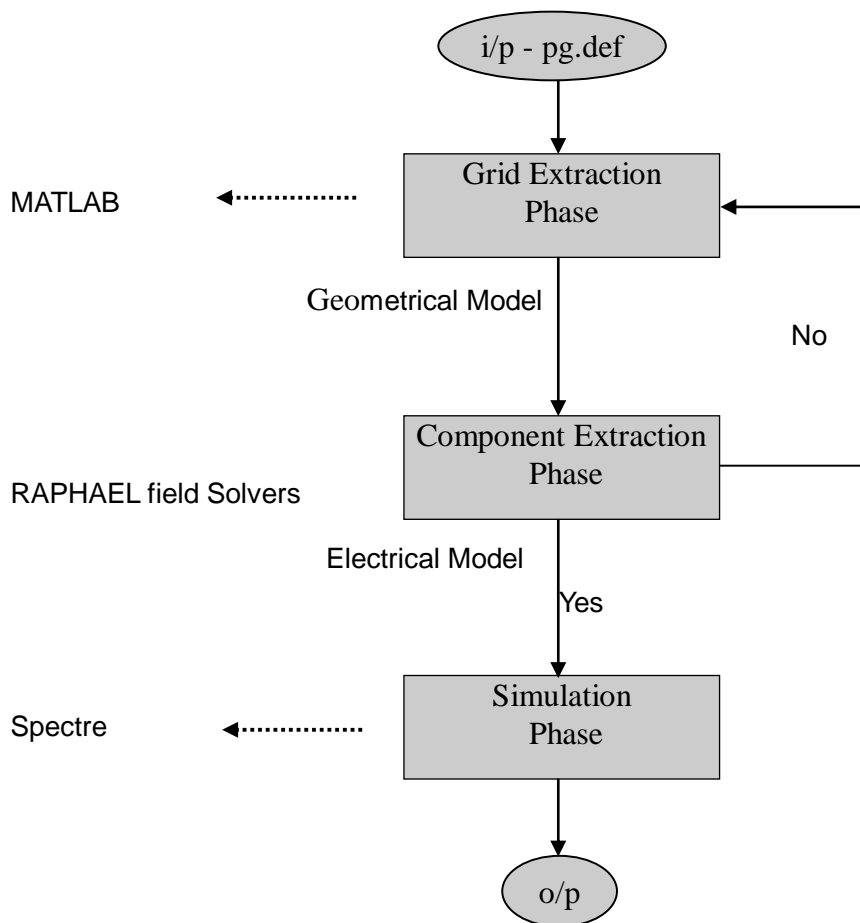
## The Design Set-up: Power integrity analysis Flow of AVR micro-Processor

The analysis flow used in [21] is made specific to a particular type of AVR processor called DELTA chip design. The power grid model is extracted with a specific type of metal wire called *stripe* and the grid is incomplete. But to make the grid complete a number of wires are added manually.

The flow used in [21] is refined by Bjorn Nilsson in his thesis work [6] and the addition of two more types of metal wires, such as *ring* and *blockring*, is used in generating the complete grid model. This eliminates the need for the work of adding wires manually. Also, the flow used in [6] is specific to DELTA chip only.

The task of this thesis work is to modify the analysis flow used in [21, 6] to make it generic to all types of AVR series, specifically it is implemented on another AVR processor called FLORIDA design. In the course of extracting the grid model, a few simplifications are done, by neglecting some sections of the power grid involving few types of metal layers, in order to keep down the grid model complexity for the circuit simulators to keep the simulation times within reasonable limits. The simulation time involved in extraction of the FLORIDA model is higher as compared to the DELTA grid model.

For the purpose of power grid analysis, the task of this thesis is divided into three phases such as *grid extraction*, *component extraction* and *simulation phase*. Figure 7 represents the overall flow of this thesis. The output obtained in each phase is made suitable for the next phase as input. The task and software tools used in each phase are discussed in this chapter. The geometrical model of the power grid is written in netlist form which is called the physical description of the model.



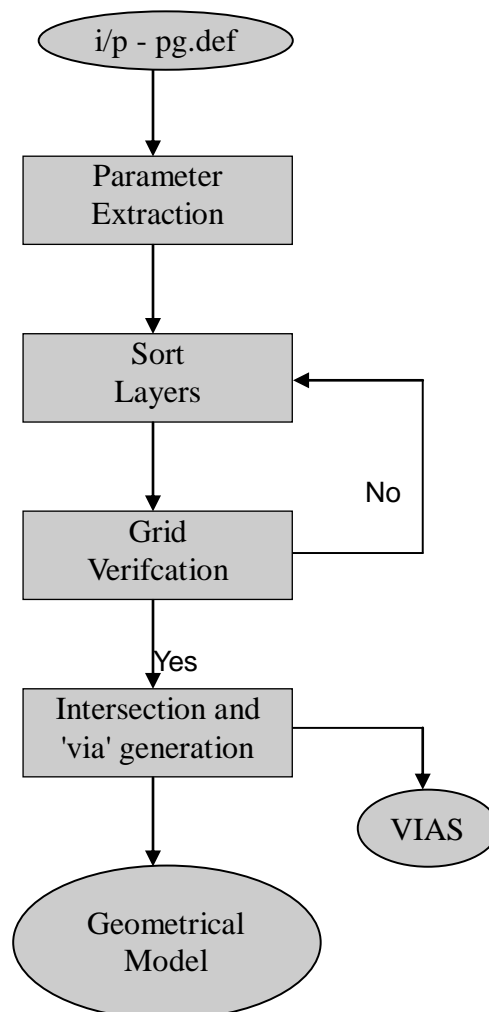
**Figure 7:** Power Grid Analysis Flow

This part of the thesis work is done in close collaboration with ATMEL Research team, Dr. Daniel Anderson and Dr. Johnny Pihl



### 3.1 Grid Extraction Phase

The input file `pg.def` is a geometrical data file, which basically contains the resources for the power grid such as *process technology*, *width*, and *metal wires etc*, given by the ATMEL team along with geometrical layout of FLORIDA design shown in Figure 2.



**Figure 8:** Analysis Flow – Grid Extraction Phase

The first step in this phase is to extract the necessary parameters for the power grid model (Vdd or Gnd) to be used for analysis purpose. This is done with the script `readCompleteDefnets.m`. It is important to browse through the `pg.def` file before moving into the first step as it would give a rough picture of how the power grid layout definition file looks like, what it contains and what are the parameters required for the

extraction of grid model which would eventually be the initial step of this thesis work. Each parameter extracted from file *pg.def* contains more than one data type and the next step would be to sort the data types corresponding to its parameters. This is done using the script *sortlayers.m*. The next step is using the script *plotGrid.m*, where a picture of the grid model is viewed and compared with geometrical layout (a pictorial verification) which confirms that the right combination of metal wire types is chosen. Different combinations of metal wire types result in different grid structures and hence proper selection of metal wire types is needed. The next step is to divide the metal wires into different wire segments which results in generation of interlayer *vias*. This is done using the script *findIntersection.m*. These *vias* are used in the next phase of the analysis flow. The last step in this phase is to generate the geometrical model of the power grid which is done by taking the internal representation of *findIntersection.m* and it is written in the form of netlist which is fed as input to the next phase. This is done with the script *powergridGenerator.m*.

For the task to be completed, MATLAB is used in this phase of grid extraction

### 3.1.1 MATLAB

MATLAB being an interpretation language [19] is used in the front end (grid extraction phase) and back-end (after the simulation phase) of the power grid analysis flow. The readily available plot functions and programming styles in MATLAB makes it suitable for this project. In this thesis, MATLAB is used to read the input definition file, generate power grid model and write the model in the form of net-list. But, the time involved in generating a complete grid model is very high. To reduce the grid extraction time, some sections of the power grid involving some types of metal wires such as *ring and blockring* are neglected. But care is taken to make sure that grid extracted with the types used is a complete connected grid.

## 3.2 Grid Component Extraction Phase

In this phase of the analysis flow, the geometrical model, being the input to this phase, is converted to electrical model in SPICE format. The width, thickness of metal layers and their resistive property shown in Figure 9 are not explicitly fed as input; instead they are taken into account at the last step of the previous phase. A general representation of geometrical model for a metal wire written in the netlist form is shown below.

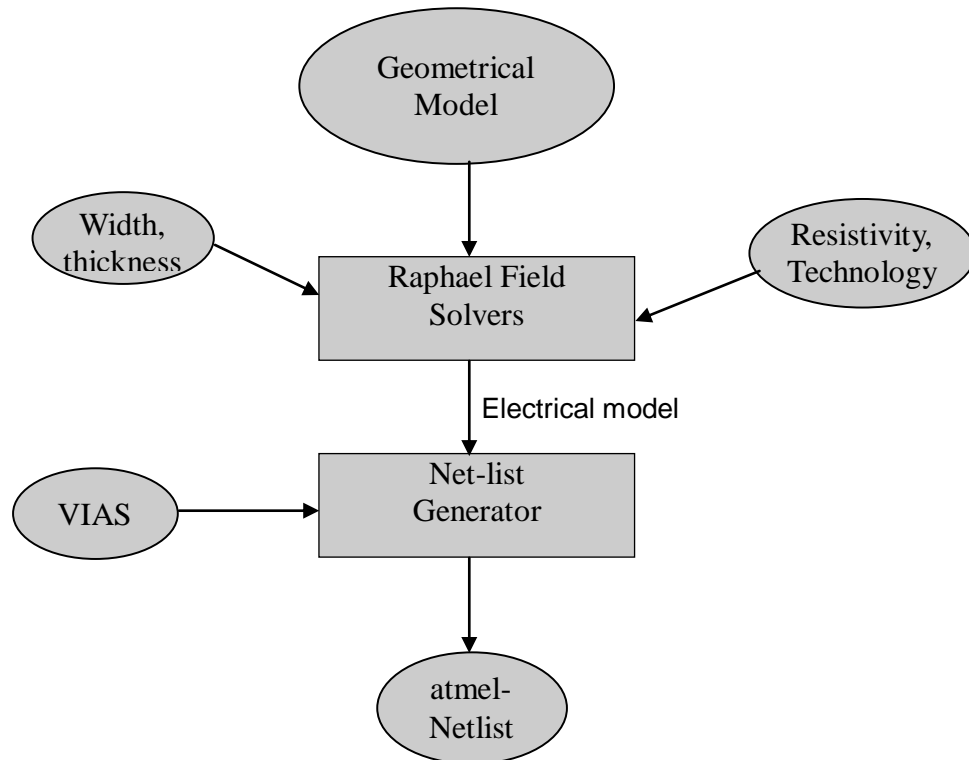
```
SINGLE_BAR NAME=FOLLOWPIN_1 NODE1=x772800y1969440z650 NODE2=x780000y1969440z650
W=1240 H=315 RHO=1.72e-9;
```

```
NODE NAME=x772800y1950240z650 POSITION=772800,1950240,6.5e-07;
NODE NAME=x780000y1969440z650 POSITION=780000,1969440,6.5e-07;
```

The script *powergridGeneratorFull.m* takes the internal representation of *powergridGenerator.m* and turns this into a file suitable for Raphael field solver [19]. Given the process technology and the netlist, resistance and inductance values of metal wires are extracted with the help of Raphael. This results in electrical model. A general description of an electrical model for a metal wire is shown below.

```
R_FOLLOWPIN_1 x772800y1969440z650 5_RI3 3.170507e-01
L_5 5_RI3 x780000y1969440z650 4.008262e-12
```

The electrical model contains all the metal wires needed for building a power network. To build a complete Vdd network, inter layer *vias* generated in the previous phase are used in this stage. Each interlayer *via is* modelled as resistors which act as a connecting link between metal layers. The *via* resistance value is given in grid definition file *pg.def*. The electrical model obtained is not on SPICE format and in order to make it compatible to SPICE or Spectre for simulations the script *subcircuitGenerator.m* is used and the resulting output is *atmel-Netlist.sp*



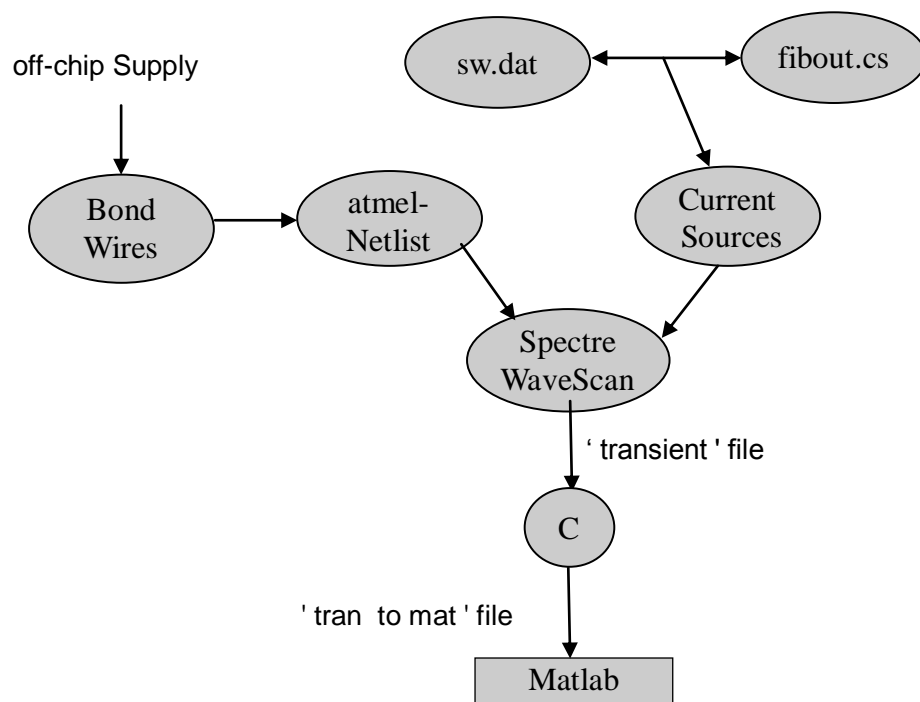
**Figure 9:** Analysis Flow – Grid Component Extraction Phase

### 3.2.1 Raphael Field Solvers

Raphael is an Electronic Design Automation tool from Synopsys [21]. By giving metal wires, their properties, position and co-ordinates, and process technology as input, the tool extracts the R and L components of the circuit. Raphael can extract C components of the grid as well. But the capacitance extraction from Raphael is ignored in this work because the capacitance of one particular grid (Vdd) does not have much coupling effect since Gnd is ideal. It would be interesting to see what happens when both Vdd and Gnd grid are taken into consideration, as most of the capacitive coupling will be between these two grids.

### 3.3 Grid Simulation Phase

For the power grid analysis, the entire model of the power grid is required. The off-chip supply shown in Figure 10 is modelled as voltage source to supply a stable voltage of  $1.8V$  and this voltage acts as reference for the logic levels. In order to make the connections between IC core and off-chip supply, the bonding wire is modelled using an R-C-L link. The netlist of bonding wires and the *atmel-Netlist.sp* are written in a file *atmelGridnetlist.sp* using the script *wrapNetlist.m*.



**Figure 10:** Analysis flow – Grid Simulation Phase

The remaining part left in the entire power grid model is the logic gates. These are modelled as piece-wise-linear current sources using the two scripts, that is, *generateSwitchingPatterns.m* and *generateSourceNetlist.m*. The first script reads current and switching profiles and the second script writes the current sources in a netlist file *atmelSourceNetlist.sp*.

The *sw.dat* (current profiles) file gives information regarding the time interval and its corresponding current signature values. The *fibout.cs* (switching profiles) file gives information regarding the nodes that have switching logic, logic capacitance and switching activities for the particular logic.

The real time application - FLORIDA design, used in this thesis, is an AVR 32 bit processor from ATMEL designed in a 180 nm process. The nominal supply voltage is 1.8V. Some simplifications have to be made while aiming to model the logic gates as current sources. All the gates are modelled with the same current waveform of NAND gate used in the process. This current waveform is applied to each gate in which its magnitude is scaled according to the load capacitance and the time is scaled relative to the rise time of the gate. The switching activities influence the switching of the gates at a given point of time interval. Thus, current waveforms of the nodes are modelled as time varying current sources which are connected between the power grid node and the ground.

The two SPICE files *atmelGridnetlist.sp*, *atmelSourceNetlist.sp* and voltage sources are wrapped in the file *wrapSpicefiles.sp* using the script *wrapSpiceFiles.m*, which forms an entire power grid model in which circuit simulations can be performed.

### 3.3.1 Spectre

Spectre is a simulator platform which comes under EDA Cadence tool, which has an in-built feature where SPICE files can be run without the need of special conversion or modification. Another reason for choosing Spectre is to avoid convergence problem that SPICE encounters where the simulator could not able to get the right DC operating point. Also, a problem of writing waveform data within the time limit occurs in Spectre when trying to simulate the grid with mutual inductances. For viewing the output waveform, Wave-scan can be used which; a tool which is supported by Spectre.

# 4

## Results of Power Integrity Analysis

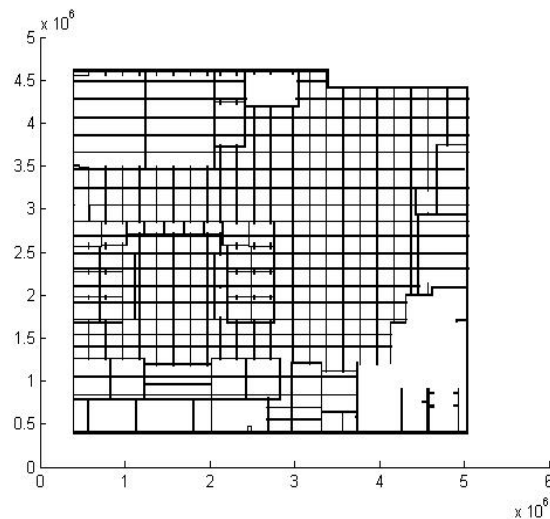
At low frequency impedance due to resistance dominates where as at high frequency the reactance dominates impedance. The FLORIDA design, used in this thesis, operates at 20 MHz which is not high speed but low speed. Thus on-chip inductance is considered in our power grid analysis.

The on-chip inductance is smaller than the inductance in bond wires. Due to the RL nature of the power grid and the current demands of load circuits, the drop in supply voltage delivered to logic cells limits the performance of the design.

The circuit simulations on the power grid model of FLORIDA design is done for three different cases, that is, best case, worst case and typical case timing. The voltage source connection, bond wires, the number of nodes and estimated amount of decap per unit area of the chip remains the same for all the three cases. But the difference in each case is the switching and current profiles.

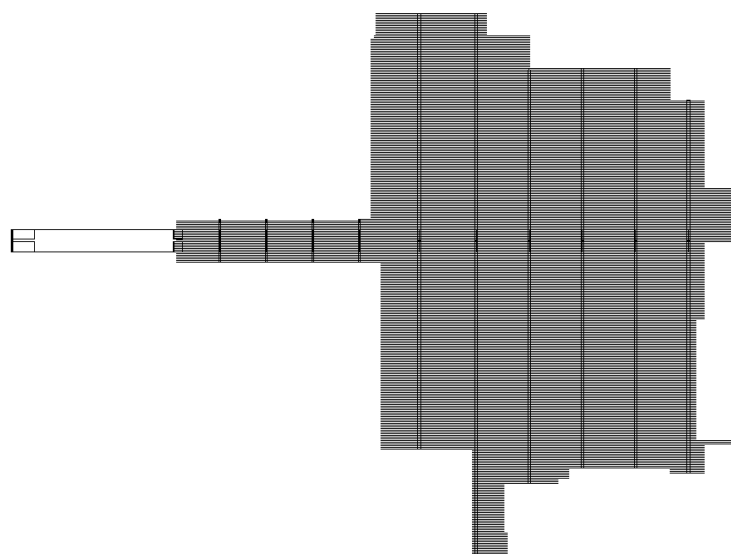
## 4.1 Resulting Power Grid Model

The modified flow used in this analysis is first implemented on the DELTA chip used in [6]. The outputs (physical description and geometrical representation) from the flow used in [6] and from the flow used in this thesis are compared. This result in the same output, thus suggesting that the flow used in this thesis can be implemented on any series of Atmel processors.



**Figure 11:** Geometrical Model: DELTA chip

Figure 11 presents the geometrical model of DELTA chip. The power grid is designed with three metal layers, that is, M1, M2 and M5, and the types considered for the extraction are *ring*, *blockring* and *stripes*.

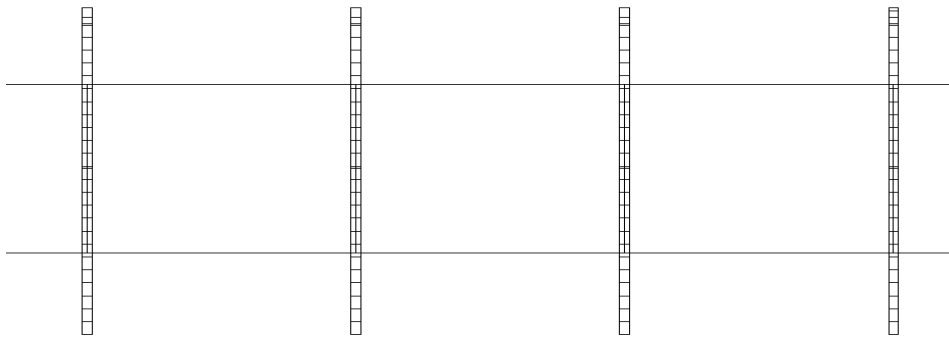


**Figure 12:** Geometrical Model – FLORIDA chip



The analysis flow used in this thesis is implemented on another processor from Atmel series, the FLORIDA chip. The power grid of this chip is designed with four metal layers, that is, M1, M3, M4 and M5 and the types taken into account for the extraction of the power grid model are *stripes* and *followpin*. Figure 12 presents the fish bone structure type of the FLORIDA chip power grid.

In the grid extraction phase, an important parameter of the power grid model to be generated is the interlayer *vias*. The idea behind the generation of *via* is: First, a metal wire of a type is divided into many wire segments and similarly it is done to all metal wires of its types. Second, a wire segment in a metal layer is checked against all the wire segments of different metal wires to find its connectivity and similarly it is done to all wire segments of all metal layers. Figure 13 shows the section of the grid model involving M4 horizontal *stripes* and M5 horizontal *followpin* and intersection between M4 and M5 is *via* represented by + symbol.

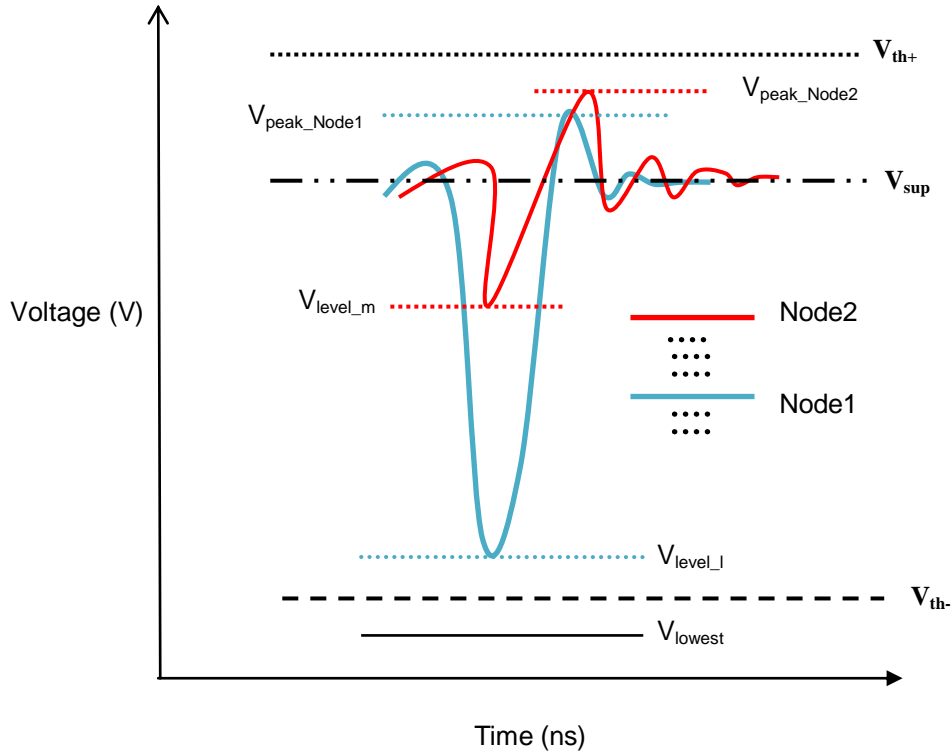


**Figure 13:** Interlayer - *Vias*

The simulation time involved in the generation of *vias* for FLORIDA design is roughly 8x times the time involved for the DELTA design. Maybe this can be due to the increase in the number of wire segments in FLORIDA as compared to the DELTA model.

## 4.2 Results from Simulation

Figure 14 presents the voltage waveform of two nodes, that is, Node1 and Node2 with time-the x-axis and voltage-the y-axis. The following terms will be useful in analysing voltage at the grid nodes.



**Figure 14:** Voltage Waveform of two nodes

**$V_{sup}$ :** This refers to the nominal voltage level. All the nodes are expected to be at this voltage level, so that the desired functionality of logic devices is certain.

**$V_{th+}$ :** This refers to the maximum voltage level; upper threshold limit. In this thesis,  $V_{th+}$  is assumed to be  $V_{sup} + 5\% * V_{sup}$

**$V_{peak\_X}$ :** This refers to the voltage level rise (voltage peak) that occurs directly on the node **X** after the first voltage drop. For example:  $V_{peak\_Node1}$  and  $V_{peak\_Node2}$  shown in Figure 14 presents the voltage level rise on the nodes Node1 and Node2 respectively. The value of voltage peak on any node should not rise above  $V_{th+}$  and the same will determine the characteristic of first overshoot on metal wires. Highest voltage peak value of all the nodes of the power grid for best, worst and typical case is referred to as  $V_{peak\_min}$ ,  $V_{peak\_max}$  and  $V_{peak\_typ}$  respectively.

**$V_{th}$ :** This refers to the minimum voltage level; lower threshold limit. In this thesis,  $V_{th}$  is assumed to be  $V_{sup} - 5\% * V_{sup}$ . Nodes having voltage level beyond  $V_{th}$ , it is no guarantee that gates connected to these nodes operate properly.

**$V_{level\_1}$ :** This refers to the node, for example: Node1 in Figure 14, with the lowest voltage level at the the node within the *lower threshold limit*.

**$V_{level\_m}$ :** This refers to the node, for example: Node2 in Figure 14, with the highest voltage level of all the nodes of the power grid.

**$V_{Node}$ :** This refers to the absolute voltage level at the nodes of the power grid except the node with  $V_{level\_1}$  and  $V_{level\_m}$ .

**$V_{max}$ :** This refers to the *maximum voltage drop* - the difference between nominal voltage level ( $V_{sup}$ ) and the lowest voltage level ( $V_{level\_1}$ ).

**$V_{min}$ :** This refers to the *minimum voltage drop* - the difference between nominal value ( $V_{sup}$ ) and the highest voltage level ( $V_{level\_m}$ ).

**$V_{drop}$ :** This refers to the voltage drop - the difference between nominal value ( $V_{sup}$ ) and the node voltage ( $V_{node}$ ), exclusive  $V_{level\_1}$  and  $V_{level\_m}$ .

**$V_{lowest}$ :** This refers to the lowest voltage level at a node of all the nodes of the power grid. The difference between nominal value and  $V_{lowest}$  is the largest voltage drop ( $V_{largest}$ ).

The reduction in supply voltage at the nodes is due to grid parameters and the current demands of load circuitry. In Figure 14, the initial voltage drop is because of the current loads and the resistive property of metal wire. The amount of current that the gates draw at the node where they get connected will be proportional to the drop. This is called IR drop.

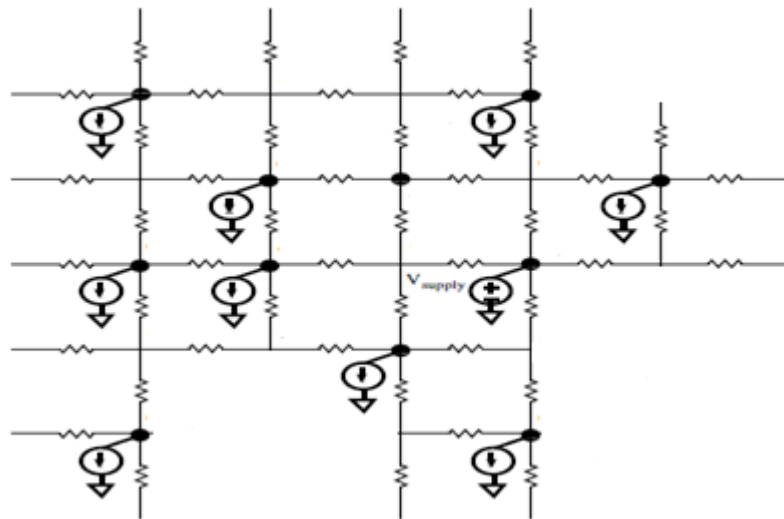
The voltage drop is also due to another electrical property, the *inductance*, which is in series with the resistance. This is because, due to switching activity of the logic gates,

there will be rapid change in current,  $i(t)$ , which develops a backward force equivalent to  $L(di/dt)$ . This shows the importance of on-chip inductance at high frequency.

Simulations are performed for three different timing cases as shown in Table 4, which differ from each other with respect to the timing of a chip shown in Table 4.

**Table 4:** Different timing cases

Minimal: Best case delay or timing	Typical: at normal case	Maximal : Worst case delay or timing
---------------------------------------	----------------------------	-----------------------------------------

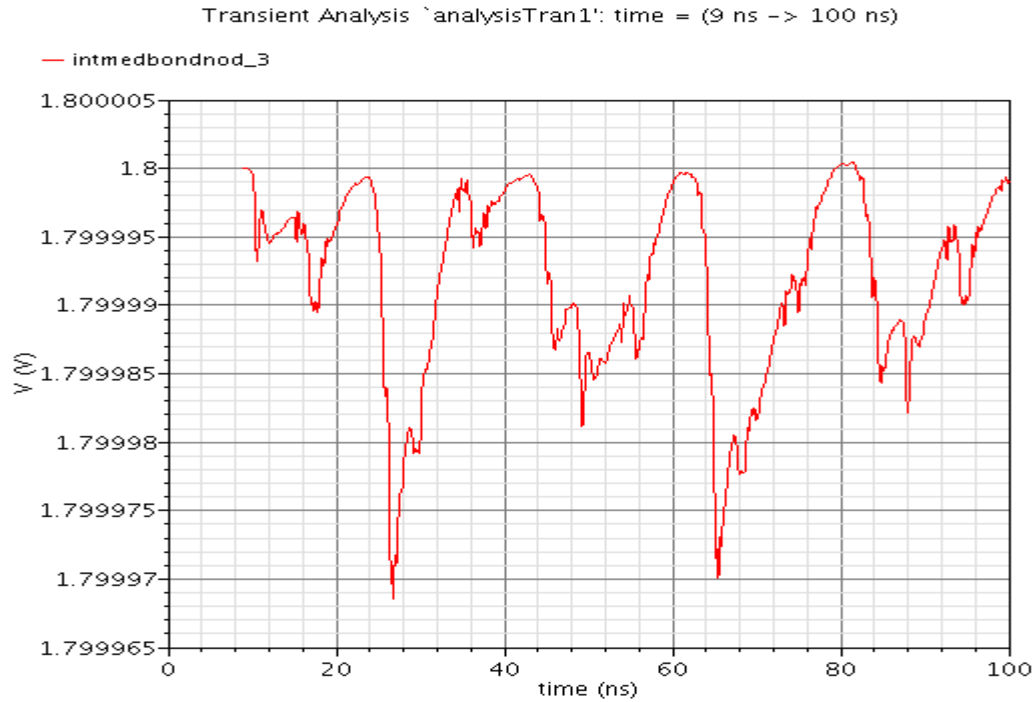


**Figure 15:** Power Grid model with multiple current loads

The characteristic of the grid and current loads determine the response of the grid. Simulations are performed on power grid network with multiple PWL current loads connected at various nodes of the grid as shown in Figure 15. The value of impedance in each arm of power grid is different.

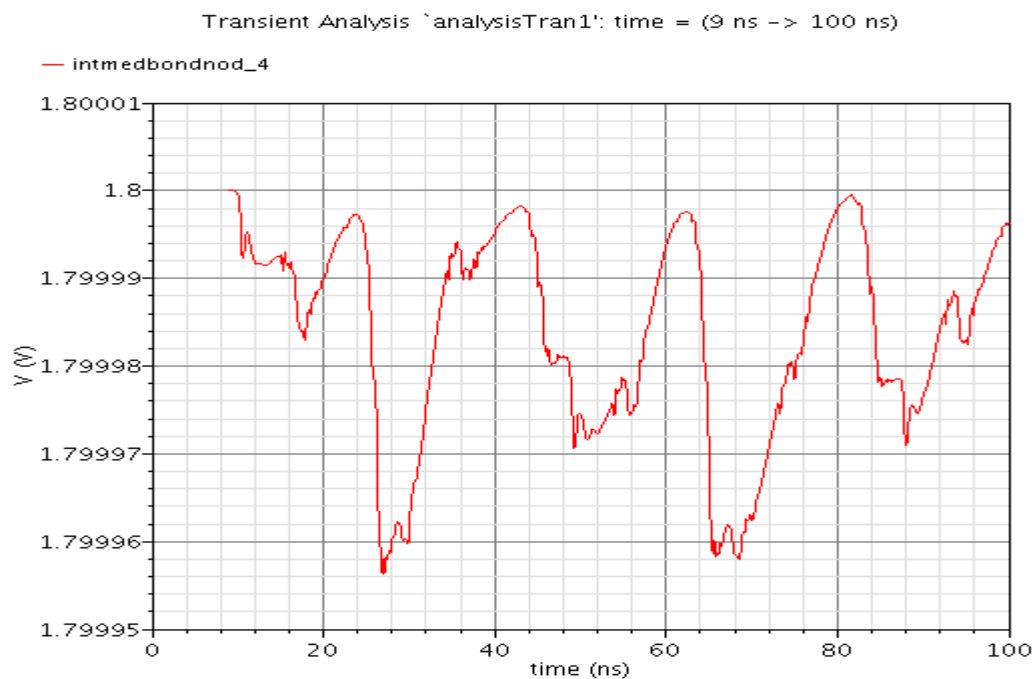
#### 4.2.1 Inductance Dominance in Package

The connection between off-chip supply and die is done using flip-chip package, wire bond etc. The type of packaging technology used in FLORIDA design is wire bond. Bonding wires are modelled as R-C-L link as shown in Figure 5. Figure 16 present the voltage waveform at the intermediate node of bonding wire which has a voltage drop of 0.001 % of  $V_{sup}$ .



**Figure 16:** Intermediate Node Voltage of Bonding Wire

Due to the interaction between the bonding wire and the die, the inductance effect in bond pad contributes a significant voltage drop noise, shown in Figure 16, to the power grid net which could limit the performance of the devices.

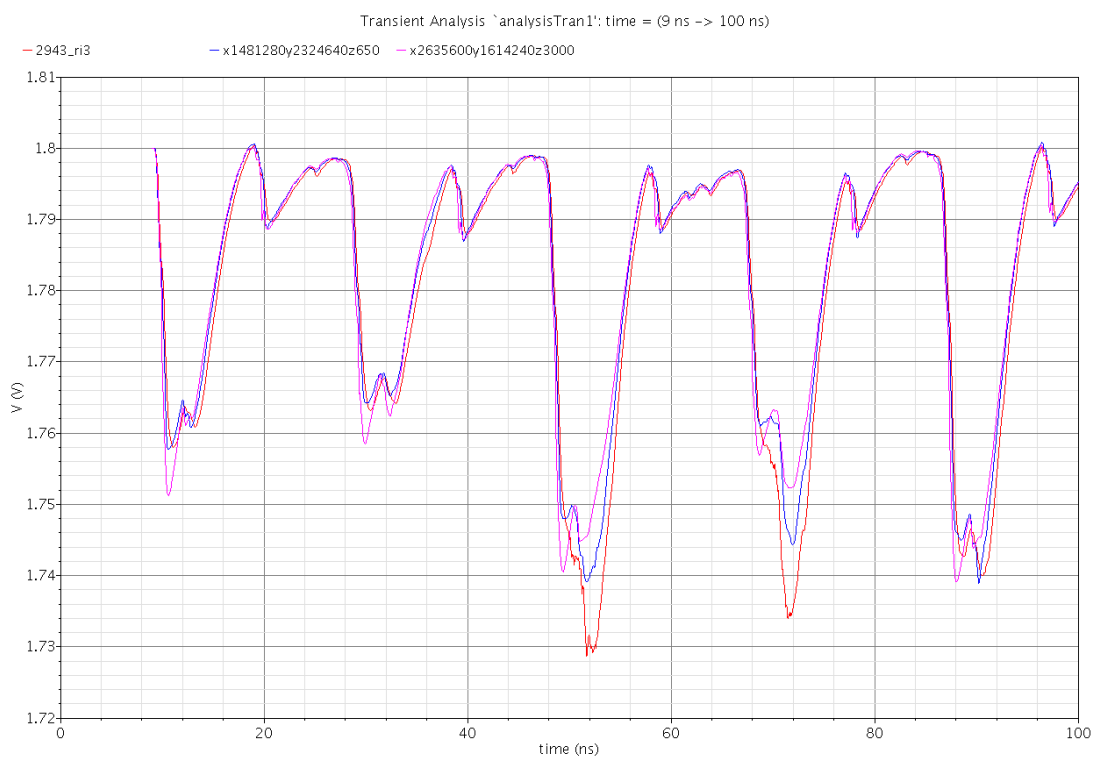


**Figure 17:** Output Voltage Waveform of Bonding Wire

To analyse the effects of package on the die precisely, a complete model of the logic gates, package and the power grid are necessary. Package-die co-design and methodologies are becoming a new area of research and tools must be available to support the features of co-design [22].

#### 4.4.2 Corner Cell - Typical

Figure 18 presents the output voltage waveform at three different nodes for the case - typical. The x-axis shows the time interval in ns and voltage in volts at y-axis.



**Figure 18:** Output Voltage Waveform

**Simulation Time:** 26 minutes and 33 seconds.

**Vdd:** 1.8 V

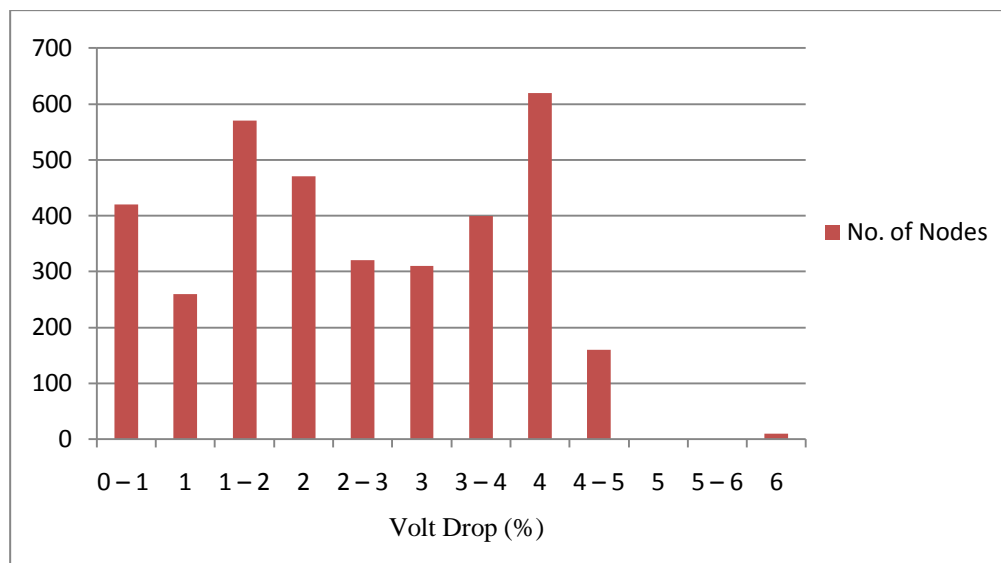
**Transient Time (T.R):** 50 ps

**C<sub>L</sub>:** 2 fF

**Timing Effort** - Typical

Parameter	Volt (V)
V <sub>level_l</sub>	1.711
V <sub>level_m</sub>	1.795
V <sub>peak_typ</sub>	1.804

In Figure 18 the voltage drop ( $V_{\text{drop}}$ ) of 0.071 V occurs at a node  $x2635600y1614240y3000$  for a particular time of logic switching (49 to 55 ns). For the same time interval,  $V_{\text{level}_m}$  and  $V_{\text{level}_l}$ , obtained using the script *backend\_process.m*, represent the node with the highest and the lowest voltage level within the threshold level, assumed  $\pm 5\%$ . The difference of  $V_{\text{max}}$  and  $V_{\text{min}}$  is found to be 84 mV. This signifies that different nodes have varying voltage drops within the range of 84 mV which are due to multiple current loads and electrical properties of the metal wires.



**Figure 19:** Volt drop Vs No.of Nodes

Figure 19 present a histogram in which most of the nodes in the power grid have their voltage drop below 4.9% of supply voltage with high percentage of nodes between 1-2% and 4%. This includes  $V_{\text{max}}$  and  $V_{\text{min}}$ . Figure 19 present the drop in voltage in the bonding wire as well as the on-chip power grid. The value of voltage peak at the nodes of the power grid refer to first overshoot immediately after voltage drop is due to inductance effects which are brought on by the wider metal wires. The highest voltage peak value of all the nodes of the power grid for typical case is 1.804 V

### 4.4.3 Corner Cell - Minimal

Figure 20 presents the output voltage waveform at three different nodes for corner cell - Minimal. From Figure 21 the voltage drop of 81 mV (49 to 55 ns) occurs at a node `x2635600y1614240z3000`.

Compared to the timing effort - typical, the amount of voltage drop at the node `x2635600y1614240z3000` in this case is increased by 10 mV.

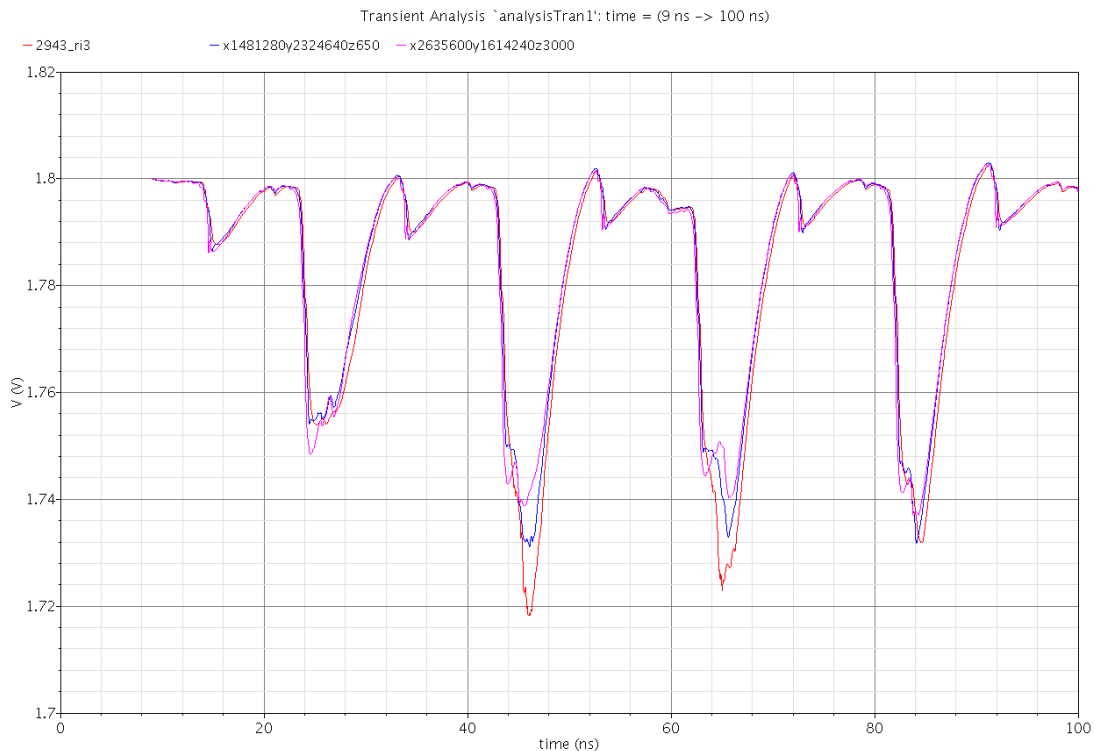
**Simulation Time:** 25 minutes and 53 seconds.

**Vdd:** 1.8 V

**Transient Time (T.R):** 50 ps

**C<sub>L</sub>:** 2 fF

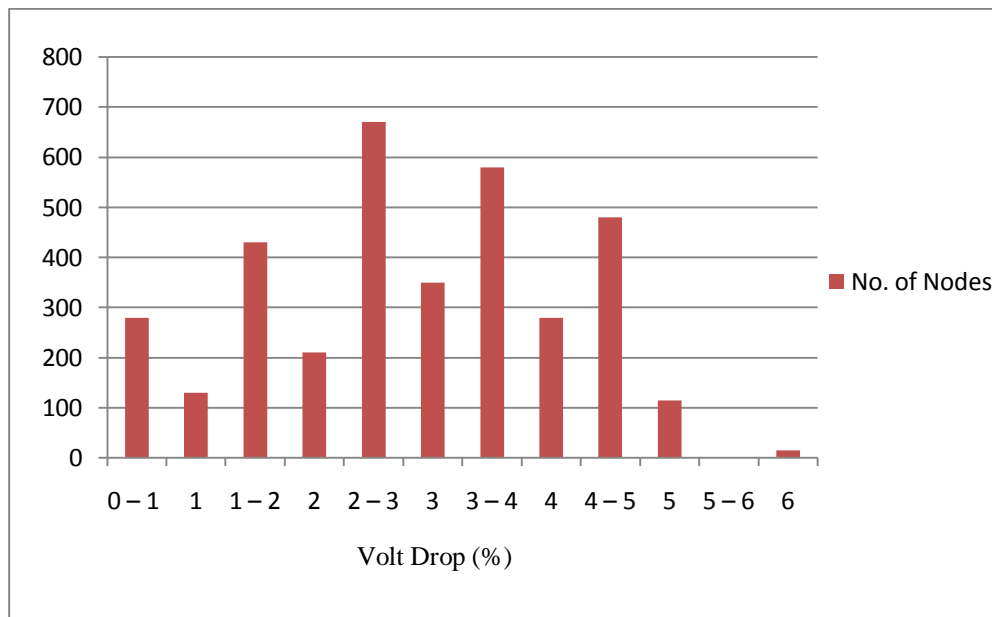
Parameter	Volt (V)
V <sub>level_1</sub>	1.706
V <sub>level_m</sub>	1.793
V <sub>peak_min</sub>	1.806



**Figure 20:** Output Voltage Waveform

Figure 21 presents a histogram with percentage drop in voltage along x-axis and number of nodes in y-axis. Most of the nodes have their voltage drop below 5.2% of supply voltage which apparently suggest the model with on-chip inductance well within reasonable limits. High percentage of nodes have voltage drop between 2-3%. The lowest voltage level at the node of the power grid is increased by a factor of 5 mV to the case typical.





**Figure 21:** Volt drop Vs No. of Nodes

#### 4.4.4 Corner Cell - Maximal

Figure 22 represents the output voltage waveform three different nodes for corner cell - Maximal.

**Simulation Time:** 27 minutes and 03 seconds.

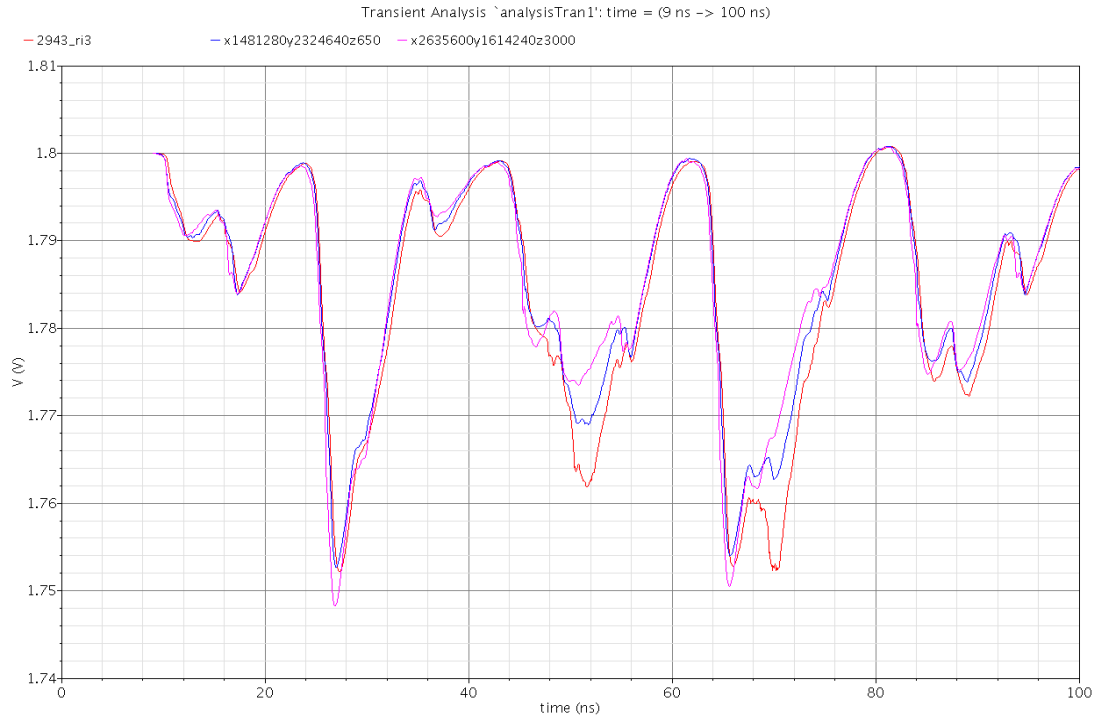
**Vdd:** 1.8 V

**Transient Time (T.R):** 50 ps

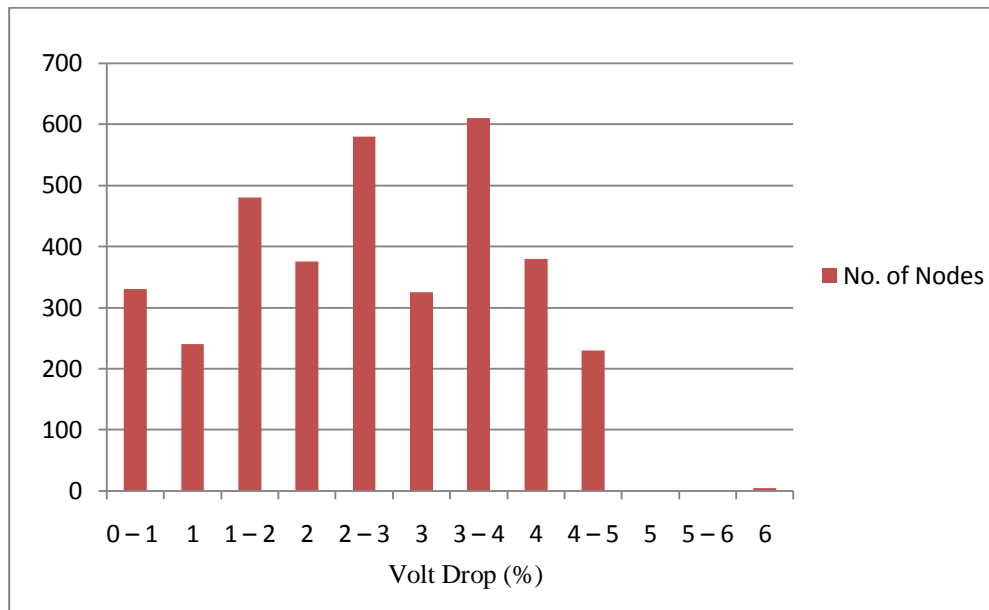
**C<sub>L</sub>:** 2 fF

Parameter	Volt (V)
V <sub>level_l</sub>	1.719
V <sub>level_m</sub>	1.794
V <sub>peak_max</sub>	1.803

From Figure 23 the voltage drop of 38 mV at a particular time interval (49 to 55 ns) occurs at the node *x2635600y1614240y3000*. In Figure 23 most of the power grid nodes have their voltage drop range between 3-4% of supply voltage. In all the three cases some nodes of the power grid have largest voltage drop of 6% which suggest that despite the metal wires being wider, on-chip inductance plays an important role in high voltage drops. This suggests that wire resistance does not scale linearly with the voltage drops. Table 6 shows some of the parameters with respect to three different cases.



**Figure 22:** Output Voltage Waveform



**Figure 23:** Volt drop Vs No. of Nodes

Some of the parameters listed in the Table 5 are obtained using the script *backend\_process.m*. The input to this script is a mat file, which has nodes and corresponding to it is the time intervals and voltage value at each time interval.

**Table 5:** Parameters of Different timing Cases

<b>Parameters</b>	<b>Minimal Cell</b>	<b>Typical Cell</b>	<b>Maximal Cell</b>
Timing Effort	Best Case delay	Nominal Case delay	Worst Case Delay
Simulation Time	25 min 53 sec	26 min 33 sec	27 min 03sec
Highest Voltage level	1.793 V	1.795 V	1.794 V
Lowest Voltage level	1.706 V	1.711 V	1.719 V
Highest Voltage Peak	1.804 V	1.806 V	1.803 V
$V_{\max}$	94 mV	89 mV	81 mV
$V_{\min}$	7 mV	5 mV	6 mV
$V_{\text{drop}}$ at node <i>x2635600y1614240y3000.</i>	71 mV	81 mV	38 mV
$V_{\text{largest}}$	6 % of $V_{\text{sup}}$	6 % of $V_{\text{sup}}$	6 % of $V_{\text{sup}}$
No of Nodes	23	32	14

# 5

## Discussion

The power grid analysis flow used in this thesis is implemented on a new processor called FLORIDA. The simulation results show that most of the grid nodes have voltage drops within reasonable bounds,  $\pm 5\%$  is assumed. The simulations are done for three corner cases, that is, minimal, typical and maximal which corresponds to the best, nominal and worst case timing effort respectively. Two important parameters determine the grid response, that is, as current loads and grid properties. The first voltage drop is mainly due to current profile and resistive property of metal wire. Some of the nodes have relatively low voltage drop (less than 1% of  $V_{\text{sup}}$ ) for all the three cases. This is due to wider wires which give low wire resistance as compared to narrower wires which give high wire resistance. At some nodes of the grid, the first overshoot after voltage drop increase beyond  $V_{\text{sup}}$  due to wider metal wires. Also, some nodes of the power grid have maximum voltage drop of around 6% which suggests that despite being wider metal layers, on-chip inductance plays an important role in high voltage drop.

# 6

## Conclusion & Future Work

### 6.1 Conclusion

The complexity of power grid model is kept low for the circuit simulators to handle and also to keep down the simulation time. This is done by neglecting some section of the power grid involving metal type *blockring* and *ring*.

The simulation result let me understand the importance of on-chip inductance in the process of analysing the power grid. Also, two important factors, that is, current loads and wire properties are responsible for the power grid response. Moreover, simulation results would have been optimistic if inductance was not considered.

## 6.2 Future Work

- In aiming to achieve a more precise solution, the Ground net should be considered along with Vdd net. Now, the coupling capacitive effects take place. In this case, the Raphael field solver would be used to extract the grid capacitance.
- The effect of mutual inductance should also be considered in order to improve the accuracy of the simulation results. However, this will increase the complexity of the grid model further, resulting in long simulation times. At the same time, to keep down the simulation time, innovative tools or circuit simulators will be important to run circuit simulations on complex power grid models.
- With the increase in current demands and reduce voltage supply, designing high performance power grid distribution network is needed. Here accurate and computationally efficient analysis will be a challenge for the future. The package-chip co design and analysis, and tools to support the co-design features will be a new area of research.

# 7

## References

- [1] Nir Magen, Avinoam Kolodny, Uri Weiser and Nachum Shamir, *Interconnect-Power dissipation in a Microprocessor*, in Proc. Int. workshop on System-level interconnects prediction, 2004.
- [2] Ndubuisi Ekekwe, *Power Dissipation and interconnect noise challenges in nanometer era*, IEEE POTENTIALS, 2010.
- [3] The SPICE Home Page, <http://bwrc.eecs.berkeley.edu/Classes/IcBook/SPICE/>.
- [4] Rabe, D. Jochens, G. Kruse and L. Nebel, *Power-simulation of cell based ASIC's - accuracy and performance trade-offs*, in Proc. Design automation and test in Europe, Feb 1998.
- [5] C. Y. Tsui, M. Pedram and A. Despain, *Efficient estimation of dynamic power dissipation under a real delay model*, in Proc. IEEE Int. Conf. Computer-Aided Design, 1993, pp. 224–228.
- [6] Bjorn Nilsson, *Integrated Circuit Supply Noise Study Based on an Extensive Power Grid Model*, Department of Computer Science and Engineering, Chalmers University of Technology. Göteborg, 2009.
- [7] B. J. George et al., *Power analysis and characterization for semi custom design*, in Proc. Int. Workshop Low Power Design, 1994, pp.215–218.
- [8] J. Y. Lin et al., *A cell-based power estimation in CMOS combinational circuits*, in Proc. IEEE Int. Conf. Computer-Aided Design, 1994, pp. 304–309.
- [9] H. Sarin and A. McNelly, *A power modeling and characterization method for logic simulation*, in Proc. IEEE Custom Integrated Circuits Conf., 1995, pp. 363–366.

- [10] H. Howard Chen and J. Scott Neely. “Interconnect and circuit modeling techniques for full-chip power supply noise analysis. *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, August 1998.
- [11] Martin Saint-Laurent and Madhavan Swaminathan, *Impact of Power Supply Noise on Timing In High Frequency Microprocessors*, IEEE Trans on Advanced Packaging, Feb 2004, pp. 135-144.
- [12] H. B. Bakoglu, *Circuits, Interconnections and Packaging for VLSI*, Addison Wesley Publishing Company, 1990.
- [13] M. Popovich, A. V. Mezhiba and E. G. Friedman, *Power Distribution Networks with On-Chip Decoupling Capacitors*, SpringerLink.
- [14] S. Pant, D. Blaauw and E. Chiprout, *Power Grid Physics and Implications for CAD*, IEEE Design and Test of Computers, May 2007, Vol. 24, No. 3, pp.246–254.
- [15] J. N. Kozhaya, S. R. Nassif and F. N. Najm, *A Multigrid-Like Technique for Power Grid Analysis*, IEEE Transactions on Computer- Aided Design of Integrated Circuits and Systems, October 2002, Vol. 21, No. 10, pp. 1148–1160.
- [16] Selcuk Köse and Eby G. Friedman, *Fast Algorithms for Power Grid Analysis Based on Effective Resistance*, in Proc. IEEE Int Symp. on Circuits and Systems, 2010.
- [17] G. Venezian, *On the Resistance between Two Points on a Grid*, American Journal of Physics, Nov 1994, Vol. 62, No. 11, pp. 1000–1004.
- [18] <http://www.mathworks.com/products/matlab/>
- [19] <http://www.synopsys.com/tools/tcad/interconnectsimulation/pages/raphael>
- [20] Anand Ramalingam, Giri Devarayanadrug and David Z. Pan, *Accurate Power Grid Analysis with Behavioral Transistor Network Modeling*, Proc. Int. Symp. Physical Design, 2007.



[21] Andersson, Daniel A, *Interconnect Delay and Integrity Issues in On-Chip Circuits*, Division of Computer Engineering, Department of Computer Science and Engineering, Chalmers University of Technology. Göteborg, 2008.

[22] Navin Srivasta, Xiaoning Qi and Kaustav Banarjee, *Impact of On-chip Inductance on Power Distribution Network Design for Nanometer Scale Integrated Circuits*, Proc. IEEE. Int. Symp. Quality Electronic Design, 2005.

[23] Fairchild Semiconductors, *IC Package Design's Effects on Signal Integrity*, High Performance Design System, Design Conf. 2003.

# B

## Appendices

In this section, the scripts used to extract the entire power grid are discussed in detail. All the scripts are interdependent to each other. Further the in-flow and the out-flow results from each script is shown with its relations to other scripts. The scripts are written in such a way that it can be used for any series of AVR processors thus making it generic.

For the post processing simulations MATLAB is used. The input to MATLAB is a converted file (tran to mat) which is done using an inbuilt C function given by Atmel team. The mat file contains signal nodes and voltage values at different time intervals in a tabular form. With these values,  $V_{\max}$ ,  $V_{\text{level}_1}$ ,  $V_{\text{level}_m}$  etc. is calculated using the script *backend\_process.m*.

## A.1 readCompleteDefnets.m

Input	Input Relations
Pg.def	-

This script reads the input, *pg.def* (*geometrical description of the power grid*) and extracts the lines which match the patterns specified in *buildNet.m*. The patterns specified in the script *buildNet.m*, used within the *readCompleteDefNets.m*, are related to wire parts (refer the source code for the functionality) such as wire types, wire width, metal layer and wire geometrical positions which are used as objects of the

Output	Output Relations
defNets.mat	Sortlayers.m

classes *blockComponents*. The script *readCompleteDefNets.m* is written in a way where the user can toggle between the nets such as Vdd or Gnd as both their grid structure are the mirror images of each other with the difference being in their nodes geographical positions. The output of this script contains the information about the metal layers properties stored in corresponding objects of classes *blockComponents*.

## A.2 sortlayers.m

Input	Input Relations
defNets.mat	readCompleteDefnets.m

The script *sortlayers.m* redefines the internal representation of *blockComponents* by sorting it into different metal layers and which type they they are. So, each metal layers may have one or more wire types and each wire type has start and end point (node).

Ouput	Output Relations
[lay <xxx> Vec.....]	nodesInitialStripes.m

The metal layers used in this design are M1, M3, M4 and M5 and where,  $\text{lay}\langle\text{xxx}\rangle\text{Vec} = [\text{lay1Vec } \text{lay3Vec } \text{lay4Vec } \text{lay5Vec}]$ . By interpreting in this way, the number of wire segments present in each layer can be extracted and also helps in selection of type's combination, as the types of metal wires determine the grid structure.

### A.3 nodesInitialStripes.m

Input	Output Relations
[lay <xxx> Vec], blockcomponents	sortlayers.m

The script *nodesInInitialStripes.m* creates only the nodes from the initial *stripe* i.e. the beginning and the end node. The nodes are saved in nodes.mat.

Output	Output Relations
nodes.mat	nodesInitialStripes.m

### A.4 findIntersection.m

Input	Output Relations
nodes.mat	findIntersection.m

Ouput	Output Relations
interResult.mat (which contains already set of nodes a and vias is null)	nodes.m, vias.m, generateSourcenetlist.m generateSwitchingPatterns.m powerGridGeneratorFull.m

The script *findIntersection.m* divides the each metal layers with its type separately into smaller wire segments creating *terminal nodes* which forms a way to intersect with the terminal nodes created from dividing other metal layer with its type into smaller wire segments. The intersection of terminal nodes of one metal layer to other metal layer results in addition of *extra nodes* which is nothing but the *vias*.

## A.5 addNodes.m and addVias.m

Input	Input Relations
nodes.mat, vias.mat, blockcomponents, lay <xxxx> Vec	nodesInInitialStrioes.m, findIntersection.m

The script *addNodes* only add the created *terminal nodes* to the already set of nodes.mat, it does not change anything regarding the physical description of the grid. The co-ordinates of vias will not be included.

The script *addVias.m* add the *extra nodes* to *vias.mat*. Note the *vias* does not include at the nodes. The *vias* will be used later to form complete grid. All the layers which contain the wire segments, its nodes and its properties are stored in *wire\_segments.mat*.

Output	Output Relations
wire_segmentss.mat	generateSourcenetlist.m, powerGridGeneratorFull.m, generateSwitchingPatterns.m

Where, *wire\_segments.mat* = [lay1Vec, lay2Vec lay3Vec, lay4Vec, lay5Vec]. This script *plotGrid.m* is used to visualize the grid model at every stage of the design.

Input	Input Relations
wire_segmentss.mat, nodes.mat, vias.mat	sortlayers.m, findIntersectin.m, addNodes.m, addVias.m

## A.6 powerGridGeneratorFull.m

Input	Input Relations
wire_segmentss.mat, nodes.mat, powerGridGenerators.m	addNodes.m, findIntersection.m

The script *powerGridGeneratorFull.m* creates a file *atmelGridnetlist*, which represents the physical description of the power grid, suitable for the RAPHAEL to extract the R and L grid parameter values.

The script *powerGridGenerator.m*, used within the *powerGridGeneratorFull.m*, is responsible to create the physical model of the power grid which has information about wire co-ordinates, wire thickness and resistivity.

Output	Output Relations
atmelGridnetlist	Raphael field Solvers

## A.7 Raphael Field Solvers

Raphael is used to extract the R and L values. The capacitance extraction using Raphael is neglected because the grid capacitance has no dominant effect over the logic capacitance. The sample script that used to extract the R and L values is

```
raphael ri3 -p < input file > -o < output file >
```

```
raphael ri3 -p atmelGridnetlist -o atmelGridnetlist.ri3
```

Also, the capacitance extraction would be important when the Gnd net is taken into account along with Vdd which would give precise solution in voltage drop. Also, when extracting the inductance for the grid, Raphael by default, extracts the self and mutual inductances. In order to perform simulations within reasonable bounds and also to keep down the complexity of the grid for circuit simulators, mutual inductance is neglected.

<b>Input</b>	<b>Input Relations</b>
atmelGridnetlist, process node, units, chip dimensions	powerGridGeneratorFull.m

<b>Output</b>	<b>Output Relations</b>
atmelGridnetlist.ri3	Spectre

## A.8 subCircuitGenerator.m and wrapNetlist.m

<b>Input</b>	<b>Input Relations</b>
atmelGridnetlist.ri3, vias.mat, bonding wires	Raphael field Solvers, addvias.m, findIntersection.m

The script *subCircuitGenerator.m* is used to convert the input file *atmelGridnetlist.ri3* to spice compatible format. The file *atmelnetlist.ri3* contains both self and mutual inductances. The script is written in such a way that mutual inductances are neglected and the output from this script will be a file with R and self-inductance (L) only.

Also, the generated file from Raphael is the net-list with no connection between different metal layers. The inclusion of *vias* explicitly, bonding wires and *atmelnetlist.ri3* are written in the file *atmelGridnetlist.sp* using the script *wrapNetlist.m* which results in the complete connected net-list (electrical model). The bonding wire network

<b>Output</b>	<b>Output Relations</b>
atmelGridnetlist.sp, atmelNetlist.ri3	Spectre

## A.9 generateSwitchingPatterns.m

Input	Input Relations
nodes.mat, sw.dat's wire_segments.mat, fibout.cs's	powerGridGeneratorFull.m

*fibout.cs's* (Max, Min, Typ cases) - Contains information of individual gate nodes, load capacitance, transient time and switching pattern and its corresponding time intervals.

*sw.dat's* (Max, Min, Typ cases) - Contains current signatures for a gate at a state transition for a number of load capacitances and transient times.

The script *genSwitchingPattern.m* reads the input file (*fibout.cs* and *sw.dat*) and extracts the parts specified in the same and are stored *sourceData.mat*. Each node in *fibout.cs* files have their switching properties which would be used to determine their corresponding node on the grid to have logic switching (modelled as current sources).

Ouput	Output Relations
sourceData.mat	generateSourcenetlist.m

## A.10 generateSourcenetlist.m

Input	Input Relations
sourceData.mat, wire_segments.mat, nodes.mat	generateSwitchingPatterns.m, powerGridGeneratorFull.m

The *sourceData.mat* contains the nodes and its corresponding logic switching activities, current profiles and time intervals. The script *generateSourcenetlist.m* reads the information available in *sourceData.mat* which are used to model logic devices as piece-wise linear current sources. This is done by by running the nodes present in *sourceData.mat* against the *nodes.mat*. The nodes which are common to both determine the location at which the current sources should be placed. Also, the position and

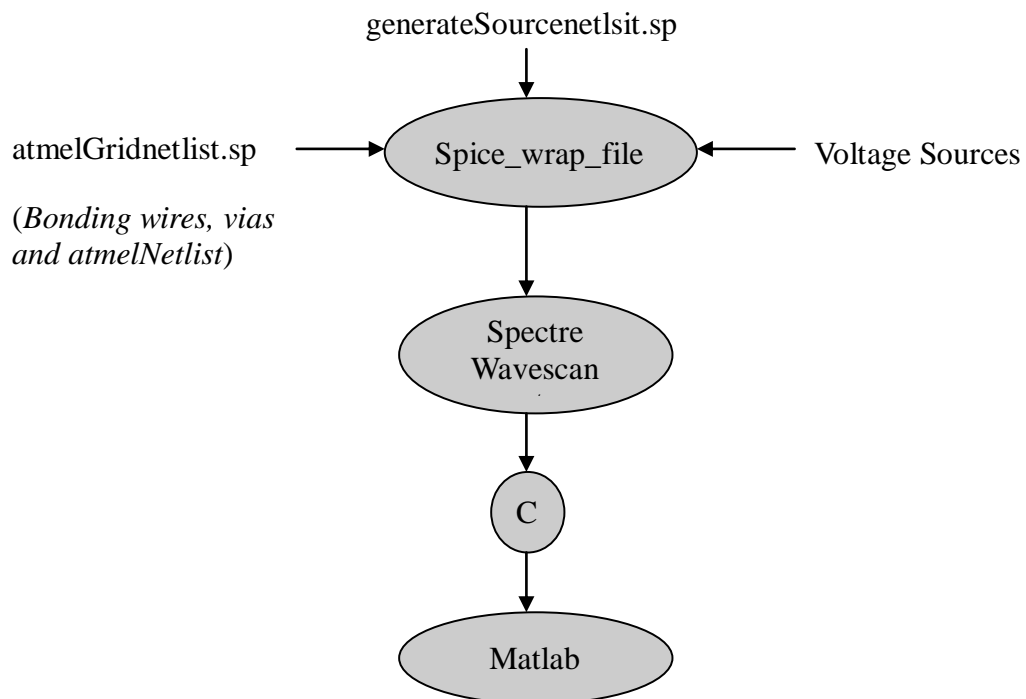


amount of decap for each node is modelled in this script. The output file contains the variables such as gate information, nodes, connecting nodes time, current and de-cap.

Output	Output Relations
generateSourcenetlist.sp	Spectre, Wavescan,

### A.11 wrapSpiceFiles.m

The script *wrapSpiceFiles.m* is used to wrap the two spice files (*atmelGridNetlist.sp*, *generatesSourcenetlist.sp*), in the file *wrapSpiceFiles.sp* which is fed as input to *spectre* for the purpose of simulations. Figure A.1 below represents the files which are needed to run the simulations on power grid in spectre. The output could be viewed through Wave-scan viewer supported by Spectre.



**Figure A.1:** Spectre Simulation