

Chip-Level Verification for Parasitic Coupling Effects in Deep-Submicron Digital Designs

Lun Ye¹ Foong-Charn Chang¹ Peter Feldmann¹ Nagaraj NS² Rakesh Chadha¹ Frank Cano³

¹Bell Laboratories, Murray Hill, New Jersey 07974

²Texas Instruments Inc., 8505 Forest Ln, M/S 8635, Dallas, Texas 75251

³Texas Instruments Inc., P.O. Box 1443, M/S 714, Stafford, Texas 77251

Abstract

Interconnect parasitics are playing a dominant role in determining chip performance and functionality in deep-submicron designs. This problem is compounded by increasing chip frequencies and design complexity. As parasitic coupling capacitances are a significant portion of total capacitance in deep-submicron designs, verification of both performance and functionality assumes greater importance. This paper describes techniques for the modeling and analysis of parasitic coupling effects for large VLSI designs. Analysis results from a controlled experimental setup are presented to show the need for accurate cell models. Results from application of these techniques on a leading edge Digital Signal Processor (DSP) design are presented. Accuracy comparison with detailed SPICE-level analysis is included.

1. Introduction

Accelerated trends in technology scaling, increasing chip frequencies and growing design complexities are highlighted in the SIA roadmap [12]. Design productivity crisis and some efforts to address this are discussed in [13] and [14]. Increasing dominance of interconnect parasitics is a major concern for present and next generation VLSI designs. Due to increasing layers of interconnect and reduced metal pitch, coupling capacitance could contribute in excess of 70% of total parasitic capacitance. This coupling can critically affect a signal by modifying its timing in either direction, or by degrading the slew and other signal characteristics. Crosstalk induced glitches in adjacent circuits may produce logic errors and voltage levels that are unacceptable for electromigration safety. These effects are commonly referred to as signal integrity effects and could lead to functional failures as well as performance degradation.

The cross-talk induced glitches are increasingly difficult to detect and verify at 0.25 μ feature size and below. False switching due to glitches can cause a change in the intended functionality of the design. It is critical to verify against crosstalk glitches in the design verification phase. While the crosstalk analysis is intended as an audit to produce conservative results and not to miss any real problems, worst case design assumptions are often an overkill and may not enable meeting design specifications. This paper describes techniques and methodology useful for predicting the glitches at the full-chip level. Results for several test cases and a real design are included.

This paper is organized as follows. Section 2 reviews the parasitic coupling effects and previous work. Section 3 discusses the MPVL technique used for the interconnect analysis. Cell modeling techniques and results are presented in Section 4. Results from application of the cell modeling technique and MPVL analysis on a leading edge DSP design are presented in Section 5. Conclusions with remarks on future work are presented in Section 6.

2. Impact of cross-talk on glitches and timing

A signal integrity violation, which includes glitch and timing deterioration, occurs when a signal on a net (called victim) is adversely affected by electrical activity on other nets (called aggressors) through parasitic coupling capacitors. Unintended glitches can be introduced on a victim net while aggressor nets are switching. Moreover, the interconnect delay from input to output and the output slope of the victim net will change when the aggressor nets are switching [11]. The impact of interconnect on glitches and timing in analyzing the signal integrity problems is described as follows.

The coupling capacitances of the interconnect network are the primary consideration in crosstalk analysis. A larger coupling capacitance will result in a larger crosstalk voltage and will have a larger effect on timing. In this

paper, we assume a cell based methodology where the inputs to the cells are buffered and the input impedance is mainly capacitive. In addition to the interconnect, the relative strengths of the cells driving the aggressor and victim nets play an important role in determining the impact of the crosstalk. Switching on an aggressor net driven by a strong cell is likely to cause a larger glitch on the victim net. Similarly, a victim net driven by a weak cell is likely to have a larger glitch due to switching of the aggressor nets.

To obtain more accurate results from the analysis, several aspects of circuit functionality should be considered. One major source of discrepancy comes from the bus design style in which many tri-state outputs are driving a bus net. Since only one tri-state output is normally activated in real operation, such cases are handled by assuming strongest of all bus drivers is switching, thus ensuring that the worst case is analyzed and the results are conservative. In addition, the logic and timing correlation information is used to improve the accuracy of the analysis significantly. As an example, the fact that the logic values of flip-flop outputs are normally complementary can be used as logic correlation information. The timing information could be used to set up proper stimulus for the drivers.

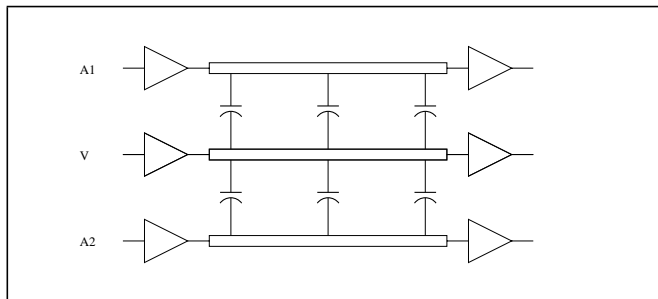


Figure 1: Coupling between parallel wires

To illustrate the impact of crosstalk, interconnect circuits corresponding to different lengths of coupled wires implemented in a 0.25 μm technology are analyzed to predict the peak glitch. Figure 1 shows the simple test case. Table 1 shows the length of coupled wires used in the tested interconnect circuits and the corresponding peak glitch value. The peak glitch increases as the coupled length becomes larger.

Similarly, we compared the interconnect delays calculated with and without the effect of coupling capacitances. The coupling capacitances are considered as grounded for the decoupled case. The delays with coupling are computed at the worst case condition where the aggressors are switching in the opposite direction to the victim net. As shown in Table 2, the deterioration of the delays is significant. Similarly, optimistic delay values can be obtained if the aggressor nets are switching in the same direction as the victim nets.

	ckt1	ckt2	ckt3	ckt4
length	100 μm	1000 μm	2000 μm	4000 μm
glitch	0.530 v	1.112 v	1.231 v	1.351 v

Table 1: Coupled wire length and glitch

ckt	Rise delay without coupling	Rise delay with coupling	Fall delay without coupling	Fall delay with coupling
ckt1	0.001 ns	0.001 ns	0.001 ns	0.001 ns
ckt2	0.034 ns	0.056 ns	0.032 ns	0.060 ns
ckt3	0.120 ns	0.231 ns	0.123 ns	0.234 ns
ckt4	0.483 ns	0.907 ns	0.496 ns	0.928 ns

Table 2: Interconnect delays

An overview of the signal integrity problems and some solutions is presented in [15] and [16]. A technique to account for logic correlations between coupled signals is discussed in [17]. Analytical models for crosstalk are discussed in [2],[18].

3. MPVL for coupled interconnect

The parasitic data from extraction is usually in RC equivalent circuit form, with millions of resistors and (grounded or coupling) capacitors. A pruning technique can be used to filter out coupling effects that are small and to decouple weak couplings. Pruning can be based on techniques such as capacitance ratio, and can be further enhanced by taking into consideration cell and context information [6], [7], [8]. Pruning identifies potentially problematic nets and reduces the size of potentially problematic clusters by decoupling weak crosstalk. After pruning, only a relatively small number of clusters have to be further analyzed, and the clusters are of much smaller size. In an example 0.25 μm design, each cluster contained on average of 105 coupling nets before pruning. After pruning, the number of nets in the clusters is reduced to 2 to 5 coupling nets.

The pruning yields the final circuit analysis problem. The circuit cluster then consists of a number of nets, their driver and load cells, together with all the coupling elements that connect them, as shown in Figure 2. The nets themselves and the couplings are modeled by the extracted resistors and capacitors. A driver cell is modeled as a general source with nonlinear impedance and the load cells are treated as capacitive terminations. A straightforward analysis of such a problem would employ a general, nonlinear, time-domain simulator such as SPICE. Unfortunately, the extracted nets

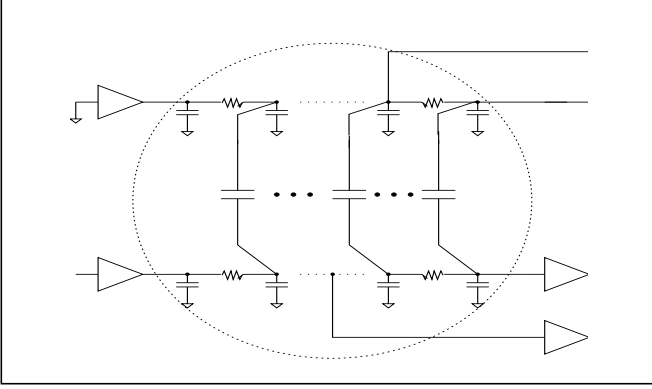


Figure 2: An example of a circuit cluster

and their couplings (even after the pruning stage) can be large, as is the number of cases that need to be analyzed. Therefore, the use of a SPICE-type simulator would require an impractically long computation time.

In our methodology, we develop a significantly more efficient circuit analysis procedure by exploiting the fact that most of the circuit is linear. More specifically, we pre-analyze the linear subcircuit and construct for it a reduced-order model using SyMPVL [1], [2], the symmetric version of the MPVL [3] algorithm. The reduced-order model is then analyzed together with the driver and load models under the various excitation and bias conditions, as necessary for crosstalk analysis.

The SyMPVL reduced-order modeling procedure is summarized below: The original MNA equations just for the interconnect RC circuit are

$$Gv + C \frac{dv}{dt} = Bi_x \quad (1)$$

where G and C are symmetric positive definite matrices representing the contribution of the resistors and the capacitors, respectively, and B is a rectangular matrix that specifies the I/O ports of the linear subcircuit. The unknowns are the nodal voltages v , and i_x represents the vector of currents flowing into the I/O ports. The first step of the SyMPVL is to produce an equivalent system of equations, where the two matrices G and C are collapsed into one. This is obtained by performing a Cholesky factorization of $G = F^T F$, and performing a change of variables $x = Fv$. We obtain

$$x + A \frac{dx}{dt} = Li_x, \quad (2)$$

where $A = F^{-T} C F^{-1}$, and $L = F^{-T} B$.

Model reduction is obtained by projecting the equation and the unknown vectors into the Krylov subspace: L, AL, A^2L, \dots , the basis of which is computed via a block-Lanczos algorithm. The corresponding reduced time-domain equations are:

$$v + T \frac{dv}{dt} = \rho \cdot i_x, \quad (3)$$

where T , and ρ represent projections of A and L , respectively. In fact the block-Lanczos algorithm will compute the projections T , and ρ directly without having to store all the Krylov subspace basis vectors.

The projected system of equations represents a good approximation of the I/O behavior of the linear subcircuit. In fact, it is shown [3] that the transfer-function matrix of the reduced system represents the matrix-Padé approximation of the original subcircuit matrix-transfer-function. Moreover, it is further proven that the reduced system remains stable and passive [4].

When performing crosstalk analyses, the most typical case is to use a more accurate model, $i_x(v_x)$, for the active driver (see next section), and assume linear terminations for the rest of the ports. The corresponding reduced time-domain equations are:

$$v + T \frac{dv}{dt} = \rho \cdot i_x(V_x - \rho^T v). \quad (4)$$

Observe that, except for the driver contribution, the system is linear. Therefore we employ an integration algorithm that takes advantage of this near linearity.

We first diagonalize the linear part of the system by factoring $T = Q^T D Q$ where Q is orthogonal, and D diagonal, and substituting $x = Qv$, $\eta = Q\rho$:

$$D^{-1}x + \frac{dx}{dt} = D^{-1}\eta \cdot i_x(V_x - \eta^T x). \quad (5)$$

This equation is integrated using a linear multi-step method: at time t_k the derivative is $\frac{dx_k}{dt} \cong \alpha x_k + \beta$ with α, β being time-step and integration-method dependent. The following system of equations needs to be solved by a Newton method at each time point:

$$D^{-1}x_k + \alpha_k x_k + \beta_k - D^{-1}\eta \cdot i_x(V_x - \eta^T x_k) = 0. \quad (6)$$

Each Newton iteration must solve a linear system with the Jacobian matrix of (6)

$$(D^{-1} + \alpha_k I + g_k \cdot D^{-1}\eta\eta^T)\Delta x = r_{kn}. \quad (7)$$

This step dominates the cost of the integration and is implemented efficiently exploiting the fact that the Jacobian matrix is a rank-1 modification of a diagonal matrix.

Multiple nonlinear terminations can also be handled efficiently by a suitable extension of this algorithm.

4. Modeling of digital cells for signal integrity verification

For DSM VLSI crosstalk analysis, the computational requirement is prohibitive. Even when a reduced order model is used for interconnect circuit, a SPICE-like driver cell model will still make the computation too expensive. Simplified cell model is mandatory for full-chip signal integrity verification. In this section, two approaches for generating simplified cell model are discussed.

4.1 Timing library based model

A first attempt for cell model is to use a linear resistor for the driving cells of both victim net and aggressor nets. In [5], it is shown that in order to obtain meaningful results, the interconnect wire resistance has to be much larger than the linear resistance used for driving cells, hence the network is dominated by the interconnect circuit parameters. However, this scenario corresponds only to very long interconnect. In [9], a Thevenin equivalent circuit driver model is proposed and is used in [10].

One way to obtain the resistance for driving cells is to utilize the characterization information contained in the cell timing library. Since the cell timing library contains characterization data for different loading conditions, the resistance used for cell driving model can be deduced accordingly. Table 3 summarizes the driving cell model results and the comparison with SPICE analysis for rising glitch analysis.

In the analysis, more than 60 different interconnect length values are used, ranging from 10 μ m to 5000 μ m, and 50 different types of 0.25 μ m cells are included. From the results obtained by using linear resistor cell model, it is clear that for high-confidence analysis, more accurate driving cell model is needed.

4.2 Non-linear cell model

To achieve better accuracy for glitch analysis, timing recalculation and electromigration analysis, the driving cell model has to be accurate enough to capture not only the average and RMS current and/or voltage at the cell driving point, but also the transient waveform at the driving point, taking into consideration the resistive effect of the interconnect. In order to capture the cell output transient waveform, non-linear yet simple cell models can be used. The process of pre-characterizing cells to generate the non-linear cell models (a one-time task) is simple in setup, and fast in

glitch(v)	avg err	std err	min err	max err
0.3 - 0.6	21.31	27.80	-43.57	70.98
0.6 - 0.9	27.08	25.62	-35.66	71.81
0.9 - 1.2	28.99	22.19	-34.37	57.18
1.2 - 1.5	16.94	14.77	-33.05	48.67
1.5 - 3.0	-4.24	12.67	-33.85	16.36

Table 3: Timing library based model (Vdd = 3.0)

execution. In Table 4, the analysis results from using non-linear cell model for computing rising glitches are presented. Comparing to that in Table 3, significant improvement in accuracy is achieved. Out of about 400 test cases with 53 different 0.25 μ m cells, only two cases have error greater than 50% (over estimate). Over 85% test cases have error less than 10% of full SPICE runs.

Peak glitch(v)	avg err	std err	min err	max err
0.3 - 0.6	7.89	12.73	-9.83	61.60
0.6 - 0.9	8.37	8.83	-4.21	32.32
0.9 - 1.2	3.43	6.52	-6.78	25.90
1.2 - 1.5	-2.43	4.28	-7.58	16.15
1.5 - 3.0	-0.32	3.01	-5.86	5.11

Table 4: Non-linear cell model (Vdd = 3.0)

Although non-linear model is used for glitch analysis, the non-linearity of the model has an minimal impact on computation speed, due to the simplicity of the non-linear model and the use of MPVL analysis engine, enhanced by applying the techniques discussed in previous section.

5. Results

In this section, results from application of the crosstalk verification tool on a real-life leading edge Digital Signal Processor(DSP) are presented. Separate experiments were conducted to quantify the sources of error due to MPVL reduced order modeling and non-linear cell modeling. A total of 113 coupled networks with number of aggressors ranging from 2-12 were simulated using SPICE and MPVL assuming a linear drive resistance of 1k-ohm. Figure 3 shows the distribution of percentage error between SPICE and MPVL on the crosstalk peaks from these cases. A negative error means that MPVL is overestimating crosstalk peak w.r.t SPICE. From these test cases, it can be seen that average

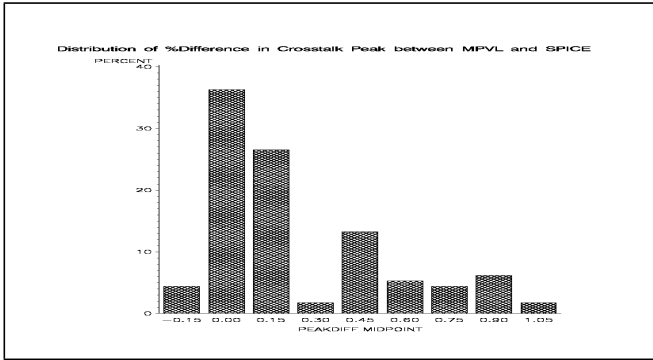


Figure 3: Accuracy comparison between MPVL and SPICE

percentage error is 0.24% and maximum percentage error is 1.05%. Complete cross-talk waveform for the case that yielded maximum percentage error is shown in Figure 4. A magnified view of the crosstalk waveform in Figure 5 shows that only the peaks differ by a small and practically negligible value. Average speed-up of 15X was observed between SPICE and MPVL. Given that the crosstalk peak is a highly non-linear function, this trade-off between accuracy and speed w.r.t SPICE is practically significant.

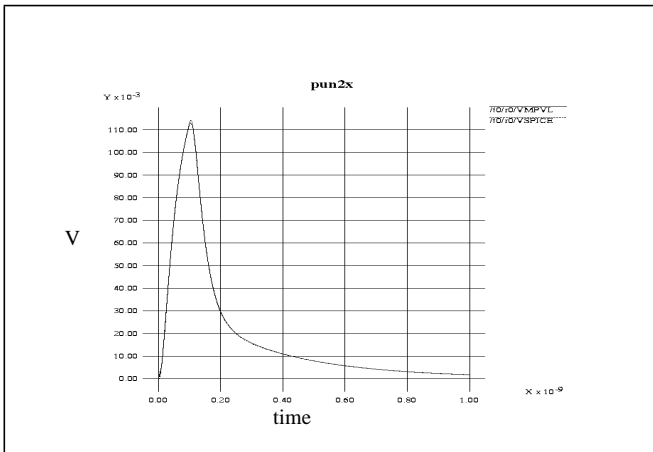


Figure 4: Comparison of crosstalk signal from MPVL and SPICE

In order to validate the accuracy of the cross-talk computation in presence of actual drivers for aggressors and victims, 101 potential victims were chosen among the inputs to latches from the same DSP design. Figure 6 and Figure 7 show the distribution of percentage errors between the SPICE with actual transistor level subcircuit and MPVL with the non-linear cell model, for crosstalk peak greater than 10% of supply voltage. A negative error indicates that SPICE results are more pessimistic. Large percentage errors are important to verify for large glitches and are not important for small glitch values. For cases where crosstalk voltage was greater than

20% of supply voltage, errors ranged between -6.9% to -0.94% for rising crosstalk and -6.1% to 10.5% for falling crosstalk. This is a desired behavior as tighter bounds are expected for larger values of cross-talk peaks. CPU time improvements averaged around 25X over SPICE. From these results, it is obvious that improvement in non-linear cell models would contribute to larger improvements in overall accuracy. More experiments are being planned to tighten error bounds on crosstalk peaks.

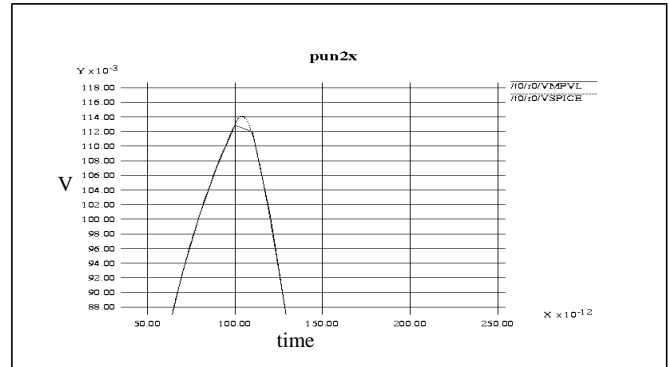


Figure 5: Crosstalk peak comparison in MPVL and SPICE

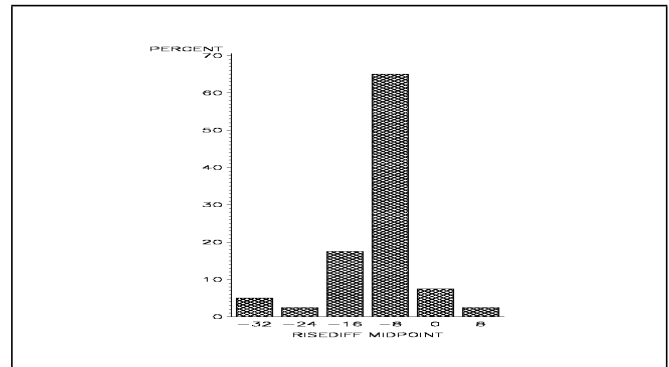


Figure 6: Rising Crosstalk peak comparison with Non-linear cell model and SPICE at transistor level

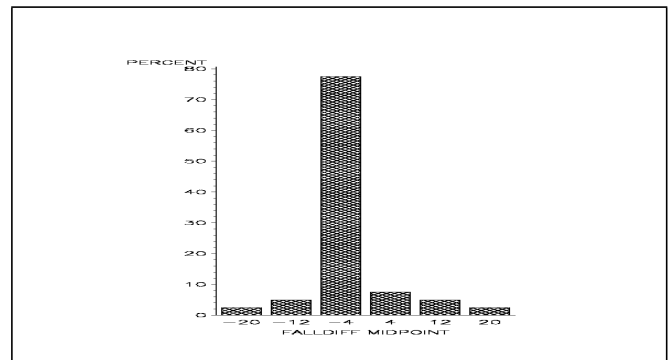


Figure 7: Falling Crosstalk peak comparison with Non-linear cell model and SPICE at transistor level

6. Conclusion

In this paper, techniques for analyzing full-chip parasitic coupling effects are discussed. A novel circuit analysis method combining order-reduction and non-linear termination makes chip-level crosstalk analysis practical. Methods for modeling driving cells are compared. To obtain analysis results that are not overly pessimistic, timing window and logic/timing correlation information is utilized in pruning and in analysis. Crosstalk analysis results on a state of the art deep-submicron Digital Signal Processor are included. Future work involves extending it to transistor-level crosstalk analysis for higher accuracy.

References

- [1] R. W. Freund and P. Feldmann, *Reduced-order modeling of large linear passive multi-terminal circuits using matrix-Pade approximation*, Proceedings, DATE-98.
- [2] Peter Feldmann and Roland W. Freund, *Reduced-Order Modeling of Large Linear Subcircuits via a Block Lanczos Algorithm*, Proceedings DAC'95.
- [3] Peter Feldmann and Roland W. Freund, *The SyMPVL algorithm and its applications to interconnect simulation*, International Conference on Simulation of Semiconductor Processes and Devices, p.113-16 Sept. 1997.
- [4] Z. Bai, P. Feldmann, and R. W. Freund, *How to Make Theoretically Passive Reduced-Order Models Passive in Practice*, CICC'98
- [5] Hiroshi Kawaguchi, Takayasu Sakurai, *Noise Expressions for Capacitance-Coupled Distributed RC Lines*, t97, December 4-5, 1997. Austin, Texas.
- [6] Ashok Vittal and Malgorzata Marek-Sadowska, *Crosstalk Reduction for VLSI*, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 16, No. 3, March 1997.
- [7] Anirudh Devgan, *Efficient Coupled Noise Estimation for On-Chip Interconnects*, Proceedings of ICCAD'97.
- [8] Antonio Rubio *et al*, *An Approach to the Analysis and Detection of Crosstalk Faults in Digital VLSI Circuits*, *IEEE Trans. on CAD*, Vol. 13, No. 3, March 1994.
- [9] N. Menezes *et al*, *Performance Computation for Precharacterized CMOS Gates with RC-Loads*, *IEEE Trans. on CAD*, Vol. 15, No. 5, May 1996.
- [10] Davide Pandini *et al*, *Network Reduction for Crosstalk Analysis in Deep Submicron Technologies*, t97, December 4-5, 1997. Austin, Texas.
- [11] Gin Yee *et al*, *Wire Delay in the Presence of Crosstalk*, t97, December 4-5, 1997. Austin, Texas.
- [12] *The National Technology Roadmap for Semiconductors, 1997*, <http://notes.sematech.org/97melec.htm>
- [13] R Bushroe *et. al.*, *Chip Hierarchical Design System (CHDS): A Foundation For Timing-Driven Physical Design into the 21st Century*, 1997 International Symposium on Physical Design (ISPD97) on April 14-16, 1997, at Napa, CA
- [14] *Design Needs for the 21st Century, A White Paper*, SRC, 1994
- [15] K.L. Shepard *et al*, *Global Harmony: Coupled Noise Analysis for Full-chip RC Interconnect Networks*, in Proc. of ICCAD '97.
- [16] K.L. Shepard, V. Narayanan, *Noise in deep submicron digital design*, in Proc. of ICCAD'96
- [17] DA Kirkpatrick and Alberto Vincentelli, *Digital Sensitivity Predicting Signal Interaction Using Functional Analysis*, in Proc. of ICCAD'96
- [18] Sakurai T., *Closed-Form Expressions for Interconnect Delay, Coupling, Crosstalk in VLSI's*, *IEEE Trans. on Electron Devices*, vol. ED-40, pp. 118-124, 1993.