# Delay and Reflection Noise Macromodeling for Signal Integrity Management of PCBs and MCMs

Slobodan Simovich, *Student Member, IEEE*, Sharad Mehrotra, *Student Member, IEEE*, Paul Franzon *Member, IEEE*, and Michael Steer *Senior Member, IEEE*

NCSU Technical Report NCSU-VLSI-93-09
Also to appear in Trans. CHMT, Feb 94.

**Abstract**

The current approaches to generating wiring rules for high speed PCBs and MCMs are unsatisfactory because they require intensive manual efforts or use over-simplifications. In this paper, an automated approach based on a-priori simulation-based characterization of the interconnect circuit configurations is presented. The improved flexibility and accuracy provided by this approach, when compared with traditional approaches, are demonstrated via an MCM interconnect example.

## I. Introduction

In high speed design it is necessary to constrain the lengths and geometry of the nets on a Printed Circuit Board (PCB) or Multi Chip Module (MCM) to ensure that timing and signal integrity requirements are met. These constraints are typically referred to as 'wiring rules' or 'design rules'. It is also often necessary to constrain driver and receiver circuit choices, termination choice, and the number of vias.

A well developed, but highly manual approach, to generating and managing wiring rules is presented by Davidson and Katopis [1], [2] and has been extended and used successfully elsewhere [3]. In their approach, the signal integrity expert conducts intensive simulation studies of different net configurations. From these studies, 'wiring rules' which guarantee first incident switching are established. (Note that this is a slightly different use of the term 'wiring rule' than the definition given in the first paragraph.) Delay and noise equations are then established within the bounds of the rules. A variation of their approach, with some limited tool support, has also been developed by Matsui et. al. [4], [5]. However, most companies do not have the signal integrity manpower for conducting such intensive simulation studies and use a somewhat different approach.

The usual industry computer-aided approach to managing timing and signal integrity requirements is illustrated in Figure 1. In this approach, a signal integrity expert specifies an initial rule

set based on a combination of simple analytic expressions (e.g. time-of-flight), rules of thumb (e.g. keep stub length less than $t_r v/10$, where $t_r$ is the rise time, and $v$ is the propagation speed), and previous experience, including simulation studies. The physical design tools then produce a placement and routing which is verified through simulation. If significant violations are detected through simulation, the designer's judgement is used to adjust the rules. The layout-simulate-adjust loop is iterated until there are no violations. There are a number of problems with this approach:

- Even the most complete theory-based equations for predicting delay and noise assume linear drivers and receivers and capacitive loads only. The delay and noise predictions made by these equations are inaccurate when compared with simulation results. Thus, in high speed systems, if the design is conducted according to these equations, the post-layout simulator will report many delay and signal integrity violations. The layout-simulate-adjust loop must then be iterated many times, increasing total design time.

- Many of the rules of thumb are also very conservative. In addition, in order to save time during rule development, the signal integrity experts prefer to specify very conservative initial rules, e.g. the stub length rule above. Such rules over-constrain the router, making its job more difficult. The effective result is increased average wire length, increased routing overflows (requiring hand routing) and, possibly, additional layers, and thus higher packaging cost.

- Specifying and adjusting the rules requires the close attention of a signal integrity expert as well as design and layout engineers. This is inefficient and many companies do not have requisite manpower. Furthermore, the signal integrity expert is often not called in until it is found that the prototype does not work.

One approach to solving the last problem above is to replace the expert with an expert system that judges the simulation results. Such an expert system has been built by Simoudis [6]. However, an expert system gives only general simple advice, e.g. 'if delay is too large then shorten etch length'. Detailed, quantitative advice is needed to do design.

To solve these problems, we have developed a wiring-rule management approach based on automatic *a priori* simulation-based characterization of the electrical properties over the physical design space. The *a priori* characterization replaces the theory-based equations often used currently. It also replaces the delay and noise equations used in the approach developed by Davidson and Katopis [1], [2]. The purpose of this paper is to describe this characterization procedure, from a signal integrity viewpoint, and to present the methodology and tool set for automatically generating wiring rules.

The advantages of the approach presented here are as follows:

- Simulation-based characterizations are more accurate and more complete than the theory based equations. Using this approach decreases design time and decreases the need for a signal integrity expert to closely monitor each design.

- The generated rules are not as conservative as those based on existing approaches and thus do not over-constrain the router. They produce a larger permitted physical design space (i.e. wiring rule extent) consistent with timing and noise requirements.

- Although generating the characterization requires several hundred simulation runs, one characterization is applied to all of the nets that have the same types and number of drivers and receivers. Additionally, the results are re-used across multiple designs and for multiple design purposes, including floorplanning, placement and routing, within a design. Additionally, the characterizations can be conducted well before physical design is commenced. Thus the total design time is significantly reduced.

A number of problems had to be solved to make this approach viable. First, the generation of simulation studies and the analysis of simulation results was automated. Second, the number of studies required to generate a sufficiently accurate characterization was controlled so that this process is completed in reasonable time. Third, the characterizations were used with the timing and noise constraints so as to generate wiring rules for a router. The solutions implemented are discussed in the next section. Two high speed MCMnet design examples are used to illustrate and verify the utility of the methodology in Sections III and IV. In those sections we also define and use two measures concerned with measuring the quality of the wiring rules produced, the flexibility coefficient and the safety coefficient.

The described methodology and tool set considerably eases the burden on the Signal Integrity expert (the person responsible for generating wiring rules) by automating the process of characterization and rule generation. The expert still must choose appropriate simulation models for the different circuits and interconnect structures within the design. The rest of the rule generating process is automated, though the expert can monitor and change its progress, if desired. The methodology described here encapsulates some of the qualitative knowledge of the signal integrity expert while providing numerical electrical response 'macromodels' that offer a new, higher level of abstraction. The end result is a methodology that leads to accurate but flexible wiring rules and subsequently to less conservative designs, with fewer signal integrity violations, lower cost and reduced time-to-market.

# II. Steps in Wiring Rule Production

Our approach hinges on establishing simplified models of various interconnect responses, e.g. delay and noise, accurately and efficiently. The steps in the approach are illustrated in Figure 2. There are three phases, a characterization phase, a macromodeling phase and a wiring rule generation phase. The first two phases are conducted before detailed design of the layout takes place. They can be performed as soon as the circuit and interconnect technologies have been determined. Rule generation is performed at the time of actual physical design, when the timing and noise requirements for each net are precisely known.

The objective of the characterization phase is to obtain a *response surface* that captures the circuit responses of interest (e.g. delay and reflection noise) of a generalized interconnect circuit topology, over a range of certain *design variables*, such as length. Characterization is performed for each unique circuit topology, e.g. the two-receiver CMOS driver/receiver circuit topology as shown in Figure 3. In this example, the design variables are the branch lengths $l1$ and $l2$, and the stub length $l3$. Other design variables, such as via count, can also be specified, if the designer desires.

Figure 4 shows an example of part of a response surface for the circuit topology shown in Figure 3. (Plotting the response surface is not a step in our methodology. Part of the [4-dimensional] response surface is shown here solely for illustrative reasons.) The electrical response characterized is settling delay, which is defined as the delay from when the signal at the output of the driver passes through the 50% point until the signal at the input of the receiver is sufficiently settled for correct latching, as shown in Figure 5. It is a useful delay measure for latched data signal nets [7]. The delay is characterized over a *design space*, specified by the ranges allowed on the design variables, in this case $0 < l1 < 10$ cm, $0 < l2 < 10$ cm, and $0 < l3 < 10$ cm. The designer, or another tool, must specify reasonable minimum and maximum ranges for the design variables. Ranges consistent with the board/module size are usually adequate.

Characterization consists of repeatedly assigning values to the design variables from their given ranges, and simulating the interconnect structure with these values to obtain the responses. The choice of values to be assigned is determined by computer experimental design techniques [8], [9]. Basically, these techniques involve choosing a suitable data interpolating function, and several combinations of values assigned to variables (called "samples"), so that the interpolating function accurately predicts the response at untried points in the design space.

Sequential sampling is used to achieve this objective as illustrated in Figure 6. Initially, a random sample is drawn over the entire design space using Latin Hypercube Sampling [10]. To determine the accuracy of the response surface, the response value at each "sampled" point is computed using the interpolant, assuming that the response at that point is not known. The data interpolation is performed by Moving Least Square Interpolation [11]. This interpolated value is then compared against the simulation result. If the difference is unacceptably large, then new samples are taken from the vicinity of the sample point in order to improve the accuracy of the interpolant. The "new sampling space" specifies this vicinity. Physically speaking, the second sampling step takes more samples in the vicinity of the less smooth regions in the response surface, for example, in the vicinity of the 'cliff' visible in Figure 4. Further details can be found in [12].

The samples are automatically simulated and analyzed with a tool called MetaSim [13], [14]. MetaSim allows generalized circuit structures to be specified as a collection of parameterized physical objects so that simulation studies can be easily conducted.

Once a suitable sample is determined, it is necessary to capture the response surface as a set of fitted equation(s) so that wiring rules can be generated. Linear equation(s) are preferred so that the wiring rule can be obtained analytically. For example, if delay is expressed as a function of the lengths,

$$t(l) = a_0 + a_1 l_1 + a_2 l_2 + a_3 l_3, \qquad (1)$$

then the length ranges that meet a delay requirement of say $t(l) \leq 1.2$ ns can easily be found algebraically. However, it is easy to see that a single linear equation does not accurately capture the response shown in Figure 4. From visual inspection at least two equations would be preferable, one for the response to the 'left' of the 'cliff', and one for the 'right'. We have developed an approach whereby a piecewise linear set of equations are automatically produced to represent the response surface. This set of equations is referred to as a *macromodel*.

There are two requirements on this macromodel. First, it must be sufficiently conservative so that a safe design is produced. Second, it must be sufficiently accurate so that the final design

is not over-constrained. One approach that would satisfy the safety requirement would be to generate a linear macromodel over the entire design space that lies 'above' all of the sample points (assuming a "smaller is better" delay requirement in the design). Visually, this macromodel would be a plane that would lie above the response surface shown in Figure 4.

The techniques used to generate a piecewise linear macromodel is explained fully in References [14], [15]. An illustration of the output produced by the macromodeling tool is shown in Figure 7. (Physically, such a response shape might be found in a lossless example, such as a Printed Circuit Board interconnection. The 'dip' between sub-regions 2 and 3 might result from a resonance that decreases reflection noise for these lengths. A similar, but smaller, 'dip' can be seen in the response surface shown in Figure 4 for $l2 = l3$. The magnitude of the resonance is small due to resistive damping.) In this simplified example, the macromodel produces two surfaces in each region, one above all of the sampled points and one below. The macromodel thus produces conservative rules for both minimum and maximum delay design requirements. The macromodel is also accurate because the different sub-regions were selected so as to produce small maximum errors in the fit. This entire process is illustrated using two examples, one suitable for a data net, another for a clock net. In both cases the circuit structure given in Figure 3 is used.

# III. Example 1 – Data Net

In this section, we obtain a wiring rule satisfying a settling delay requirement for the net topology shown in Figure 3. The noise budget for reflection noise was 0.1 V and the lossy lines were unterminated. The cross-section of the copper lines was 8 $\mu$m wide by 4 $\mu$m thick. A buried microstrip configuration was used with 8 $\mu$m of polyimide dielectric. Full circuit models for 24 mA CMOS drivers and receivers were used to generate the results. As the parasitic capacitances were very low, the rise time was fast, about 700 ps. Thus, without terminations, there is considerable ringing noise in the 5 cm+ long interconnection whose settling delay response is illustrated in Figure 4. However, at a certain point when the nets get longer, the line losses damp the ringing signal. Thus settling delay is larger for the shorter lengths shown in Figure 4 than for the longer lengths.

The 'above' half of the macromodel was generated for settling delay. (The response surface was obtained with 240 simulations, performed over two sampling stages.) Eleven disjoint sub-regions were formed as a result of the macromodeling process, the equations for two of which are given in Table 1. Also given in Table 1 is the equation for the one-piece macromodel obtained if no subdivision is done.

A wiring rule was generated by solving the macromodel equations for the electrical requirement that delay must be less than 3.5 ns. The resulting wiring rule is a region enclosed by a set of planar surfaces in the three-dimensional design space.

The rule was tested by comparing it with simulation results of 2,000 randomly generated designs, all with lengths within the design space specified ($0 < l1, l2, l3 < 10$ cm). A Monte-Carlo approach was used to ensure that the designs were evenly distributed throughout this design space.

The results are listed in Table 2. The *flexibility coefficient* measures the ability of the wiring rule to capture all feasible designs. A feasible design is one that meets all of the electrical

requirements upon simulation. The coefficient is defined as

$$\text{flexibility coefficient} = \frac{N_{rule}}{N_{sim}}, \tag{2}$$

where $N_{sim}$ is the number of randomly generated designs that are feasible, by simulation, and $N_{rule}$ is the number of designs that were also predicted to be feasible by the wiring rule. A higher flexibility coefficient is better, 100% being ideal. A low flexibility coefficient means that the wiring rule is much more conservative than it needs to be. The lower the flexibility coefficient, the harder is the task faced by the autorouter. The net result is either more time is spent hand-routing unroutable nets (or modifying the rules by hand) or additional routing layers have to be added. Conventional wiring rule generation approaches have often been criticized by designers as making their job too difficult by artificially over-constraining the nets.

The *safeness coefficient* measures the safety of the wiring rule. It is defined as

$$\text{safeness coefficient} = \frac{N'_{rule-sim}}{N'_{sim}}, \tag{3}$$

where $N'_{sim}$ is the number of randomly generated designs predicted to be infeasible only by simulation. $N'_{rule-sim}$ is the subset of the $N'_{sim}$ designs that are also predicted to be infeasible by the wiring rule. At design time, a safeness violation is detected by post-layout simulation. A safeness violation results in some post-layout manual re-routing to fix the signal integrity violation. The relative importance of flexibility and safeness depends on the relative designer time needed to re-route a net to overcome a congestion problem versus the time required to re-route a net to fix a signal integrity violation.

# IV. Example 2 – Clock Net

In this example, we show how the tools develop a rule satisfying a first incident switching criterion, i.e. undershoot and overshoot are less than the noise budget specifications (0.8 V and 1.0 V, respectively) and that there are no inflections or porches ('plateaus' or flat periods) in the rising and falling edges of the signal. Such a high signal quality is essential for clock distribution. The same circuit elements and topology is used as the one described in Section II. The results are compared with the set of 2,000 random designs and with the simplified ('standard') rule generation approach. A rule is generated for a net that must meet the above general requirements as well as the specific requirement that the 50% delay from driver output to receiver input be less than 1 ns. A rule is generated using a 1-piece macromodel, as obtained before subdivision, as well as a 10-piece macromodel.

A wiring rule was also generated using the standard approach in which 50% delay is estimated via the equation [16],

$$t_{50\%\text{-}delay} = \frac{l}{v_{prop}}\sqrt{1 + C_L/C_0}, \tag{4}$$

where $l$ is the main line length, $v_{prop}$ is the signal propagation speed, $C_L$ is the load capacitance per unit length and $C_0$ is the unloaded line capacitance per unit length. (Admittedly, Equation 4

assumes lossless lines. However, as no comparably simple equation exists for estimating delay of lossy lines, we are forced to use this equation as the 'standard approach' reference.) Overshoot and undershoot were controlled by using the rule of thumb for stub length, $l_{stub}$, [17]

$$l_{stub} < v_{prop} t_r / 10, \tag{5}$$

where $t_r$ is the rise time.

The rules generated by each of these approaches were tested by again comparing them with the simulation results of 2,000 randomly generated designs. The results are reported in Table 3. The results show that the approach described in this paper is much safer than the standard approach to generating rules and produces a significantly higher flexibility coefficient, particularly if the 10-piece macromodel is used. The safeness coefficient is improved over that for the settling delay example in the previous section because the response surface for 50% delay is smoother than that for settling delay and thus easier to capture through sampling. The flexibility coefficient is worse because of the multiple electrical constraints on the design specified by the first incident switching requirement.

# V. Discussion

In this section we discuss the time costs associated with this approach, discuss some design issues, and describe how statistical variations are accounted for.

Most of the time costs involved with this approach relate to performing the simulations and the fitting of the macromodels, the overhead associated with specifying and testing the sampling steps being negligible.

In the case studies we have conducted, the number of simulation samples needed to obtain a sufficiently accurate characterization is related mainly to the nature of the response and less to the complexity of the circuit (e.g. the number of receivers). If the response surface has many discontinuities, or is otherwise highly non-regular, then a large number of samples are needed. For the MCM layout example used here, the number of studies grew approximately linearly with the number of parameters being varied. For example, a four terminal net was adequately characterized with less than 600 samples. For a lossless PCB line characterization, a little insight can be used to improve the regularity of the surface. It was found that limiting the stub length to less than half the maximum branch length, resulted in a much smoother surface than if the stub was allowed to be as long as the branch. Such simple rules can be easily automated into the approach. (Davidson and Katopis [1], [2] were the first to realize that stub lengths must be limited to generate good rules. Our approach differs from theirs in that they require the stub length be short enough to guarantee first incident switching. We just require the stub lengths be short enough to keep the response surfaces from being too 'rough'.)

The time cost of the macromodeling step increases mainly with the dimensionality. With the current macromodeling approach, when there is likely to be more than 8-10 dimensions, steps need to be taken to reduce the dimensionality in order to control run-time. For example in a 10 receiver net that is actually a bus, the regularity present in the layout could be used to reduce a 20-dimensional problem to a two-dimensional problem. This is achieved by treating all of the stub lengths as the same, and similarly for the branch lengths. The macromodeling step can take

5 minutes to many hours for each net topology on a typical workstation. Faster macromodeling algorithms are being investigated. These time costs are small, however, especially considering that the characterization and macromodeling steps can be done early in the design and that these tasks can be easily parallelized across a network.

Our results indicate the value of using a settling delay measure for latched (i.e. non-clock) data nets and the value of the multi-piece macromodel. Many designers use a first incidence switching criteria for non-clock nets. In comparing the results in Tables 2 and 3, it is clear that though first incident switching is a slightly safer measure than settling delay, it is significantly less flexible (i.e. more conservative). In this example, if the designer used a first incident switching criterion for signal nets, the result could well be a higher degree of routing congestion, requiring more hand-routing, depending on the number of wires to be routed. First incident switching is slightly safer because the corresponding response surfaces are smoother for first-incident delay, overshoot, and undershoot, than they are for settling delay. The sampling strategy is more effective at capturing smooth response surfaces. In Table 3, the value of the multi-piece macromodel over a single equation model is well validated. The flexibility improves considerably with only negligible degradation of safety.

Process variations can be easily accounted for by specifying 3-$\sigma$ or 6-$\sigma$ statistical corner models using techniques established by Chang [18]. Characterizations are performed separately for each model and then the macromodel is fitted around all responses together.

# VI. Conclusions

We have established an automated process in which signal delay and reflection noise are managed without resorting to overly conservative and ultimately costly design practices. Central to this process is the production of macromodels that accurately capture the electrical responses of interconnect circuits over any required range of physical design variables (including wiring lengths, number of vias, layer assignment, number of loads, driver/receiver circuit types etc). The macromodels are obtained by fitting piecewise-linear equations to a set of simulation results. The macromodel must be obtained from simulation results because the available analytical expressions for circuit electrical responses are inaccurate, especially for driver circuits with non-linear output impedances, for multi-receiver nets containing stubs, and for highly resistive thin film interconnections. We have also shown some measurement results that demonstrate the improved safety (fewer signal integrity violations) and flexibility (greater ease of routing) provided by our approach.

Our solution greatly reduces the burden on the signal integrity and layout engineers. The automatic production of macromodels releases the signal integrity engineer from the need to manually conduct a large number of simulation runs. Our solution also prevents the signal integrity engineer from being forced to specify overly conservative rules just to reduce the effort required.

The automatic application of the macromodels to the generation of a package layout from the timing design releases the signal integrity engineer from the need to become involved in the detailed layout of a large number of designs. This automation also helps guarantee first pass success for high speed digital designs.

## Acknowledgements

The authors would like to thank Zaki Rakib, Garrett Simpson, and C. Kumar of Cadence, George Katopis and Jay Diepenbrock of IBM, John Grebenkemper of Tandem, Steve Lipa and J. C. Lu of North Carolina State University and Wayne Dai of the University of California (Santa Cruz) for discussions during our work. The CMOS driver models were provided by Wayne Detloff of MCNC. The authors would also like to thank the reviewers for their careful reading and valuable comments.

# References

[1] E. E. Davidson. Electrical design of a high speed computer package. *IBM J. Res. Dev.*, 26(3): pp. 349–361, May 1982.

[2] E. E. Davidson and G. A. Katopis. Package electrical design. In R. R. Tummala and E. J. Rymaszewski, editors, *Microelectronics Packaging Handbook*, chapter 3. Van Nostrand Reinhold, 1989.

[3] L. Heid, M. Degerstrom, C. Nelson, E. Priest, and G. Katopis. Designing for signal integrity in high speed systems. In *Proc. 43rd Electronic Components and Technology Conference*, pp. 536–537, 1993.

[4] T. Mikazuki and N. Matsui. Statistical design techniques for high-speed circuit boards. In *IEEE/CHMT'90 IEMT Symposium*, pp. 185–191, 1990.

[5] N. Matsui and T. Mikazuki. Electrical design techniques for high-speed circuit boards. In *IEEE/CHMT'90 IEMT Symposium*, pp. 234–243, 1990.

[6] E. Simoudis. A knowledge-based system for the evaluation and redesign of digital circuit networks. *IEEE Trans on CAD*, 8(3): pp. 302–315, March 1989.

[7] P. D. Franzon. Electrical Design. In D. Doane and P. Franzon, editors, *Multichip Module Technologies and Alternatives: The Basics*, chapter 11. Van Nostrand Reinhold (New York), 1993.

[8] J. Sacks, S. B. Schiller, and W. J. Welch. Designs for computer experiments. *Technometrics*, 31: pp. 41–47, 1989.

[9] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn. Design and analysis of computer experiments. *Statistical Science*, 4(4): pp. 409–435, 1989.

[10] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2): pp. 239–245, May 1979.

[11] Peter Lancaster and Kestutis Salkauskas. *Curve and surface fitting : An introduction*. Academic Press, 1986.

[12] S. Mehrotra, S. Simovich, P.D. Franzon, and M.B. Steer. Automatic circuit characterization through computer experiments. *NCSU Technical Report*, 1993.

[13] P.D. Franzon, S. Simovich, M. Steer, M. Basel, S. Mehrotra, and T. Mills. Tools to aid in wiring rule generation for high speed interconnects. In *Proc. Design Automation Conference*, pp. 466–471, 1992.

[14] S. Simovich. *Methodology for Automated Electrical Design of High Speed Systems*. PhD thesis, North Carolina State University, 1993.

[15] P.D. Franzon, S. Simovich, S. Mehrotra, and M.B. Steer. Macromodels for generating signal integrity and timing management advice for package design. In *Proc. IEEE 1993 ECTC Conference*, pp. 523–529, 1993.

[16] W.R. Hood Jr. *MECL System Design Handbook*. Motorola Inc., 1983.

[17] J.A. West. *Spice simulation design guide*. Signetics, Philips, 1989.

[18] F.Y. Chang. Generation of 3-sigma circuit models and their application to statistical worst-case analysis of integrated circuit designs. In *11th Asilomar Conference*, pp. 29–34, October 1978.

## Footnotes

Manuscript received , revised . This work was supported by the National Science Foundation under grants MIP-901704 and DDM9215755 and by Cadence Design Systems.

S. Mehrotra, P. Franzon and M. Steer are with the PicoLab, Digital Picosecond Laboratory, Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695-7911.

S. Simovich was with North Carolina State University. He is now with Sun Microsystems, Mountain View, CA 94043.

IEEE Log Number .

## Biographies

**Slobodan Simovich** received the Dipl. Ing. degree in Electrical Engineering from the University of Belgrade, Belgrade, Serbia, Yugoslavia, in 1987. In May of 1987, he joined the Computer Systems Group at Mihajlo Pupin Institute of Belgrade, where he worked as a computer hardware designer int he field of high-performance computer graphics and real-time systems. In 1989, he joined the Department of Electrical and Computer Engineering at North Carolina State University. There we worked on low-power systems design and CAD tools for signal integrity, earning the PhD degree in 1993. His interests include VLSI design, CAD tools for VLSI design, signal integrity in high-speed digital systems, and computer architecture. Dr. Simovich is now with Sun Microsystems.

**Sharad Mehrotra** (S'92) received the Bachelor of Technology degree in Electrical Engineering from Indian Institute of Technology, Kanpur, India, in 1989, and the M.S. Degree in Electrical Engineering from Vanderbilt University, Nashville, TN, in 1991. He is currently working towards the Ph. D degree in the Department of Electrical and Computer Engineering at North Carolina State University.

His research interests include CAD for performance driven layout synthesis, timing verification and design of high speed CMOS circuits by wave-pipelining.

**Paul Franzon** received his PhD in Electrical Engineering from the University of Adelaide, Adelaide, Australia in 1989. Before completing his PhD he had worked at AT&T Bell Laboratories in Holmdel, NJ in 1986 and 1987 and with the Australian Defence Science and Technology Organization. In these positions he worked on problems in wafer scale integration, IC yield modeling, and VLSI design. In 1989, Paul Franzon joined North Carolina State University as an Assistant Professor.

His current research interests include the design sciences for high speed packaging and interconnect, particularly signal integrity management and optimized MCM system design. Other interests include low-power electronics and applications of MEMS. His teaching interests focus on microeletronic system building including design for signal integrity, packaging system optimization, circuit design, and processor design. He has written and edited a book on Multichip Modules.

Dr. Franzon is a member of the IEEE and ISHM. He is on the steering committees for the IEEE MCM Conference and IEEE Topical Meeting on Electrical Performance of Electronic Packaging. In 1993, he received a National Science Foundation Young Investigators Award.

**Michael Steer** received his B.E. and Ph.D. in Electrical Engineering from the University of Queensland, Brisbane, Australia, in 1978 and 1983 respectively. Currently he is Associate Professor of Electrical and Computer Engineering at North Carolina State University, Raleigh, North Carolina, U.S.A.

His research involves the simulation, and computer aided design of high speed digital sys-

tems including interconnect simulation, the behavioral model development of digital drivers and receivers incorporating simultaneous switching noise, the experimental characterization of multichip modules and printed circuit boards, the computer aided analysis and design of nonlinear microwave circuits and systems, parameter extraction using simulated annealing techniques, microwave measurements, and the design of millimeter-wave quasi-optical power combining systems. He has authored or co-authored more than 80 papers on these topics.

Dr. Steer is a senior member of the Institute of Electrical and Electronic Engineers and is active in the Microwave Theory and Techniques (MTT) Society and the Components Hybrids and Manufacturing Technology (CHMT) Society. In the MTT society he serves on the technical committees on Field Theory and on Computer Aided Design. In the CHMT society he serves on the technical program committee on simulation and modeling of packaging for the Electronics Components and Technology Conference. In 1987 he was named a Presidential Young Investigator.

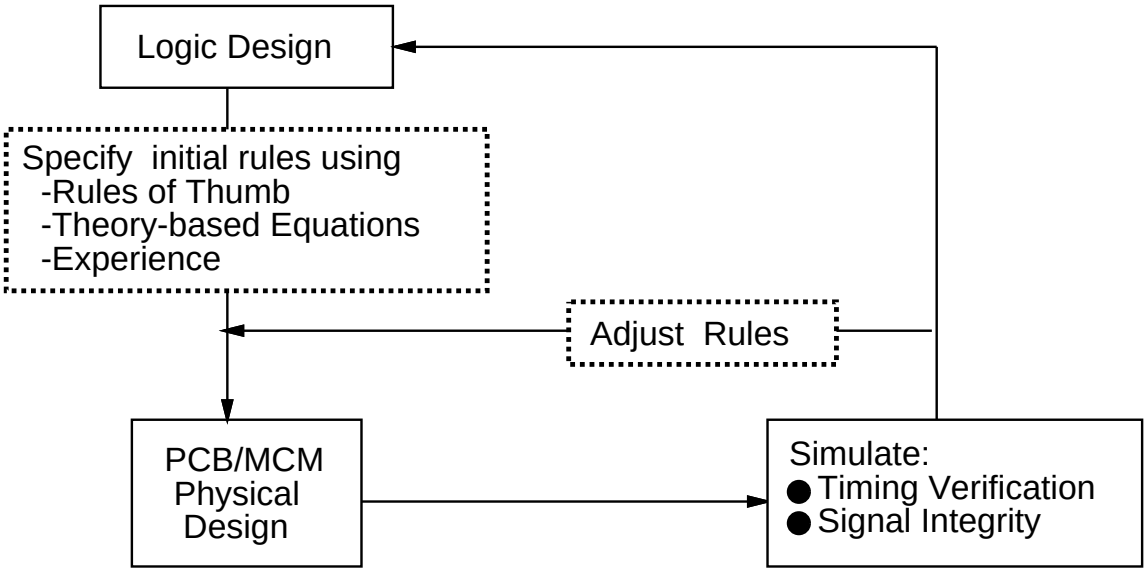


Figure 1: Current Approach to Managing Wiring Rules.

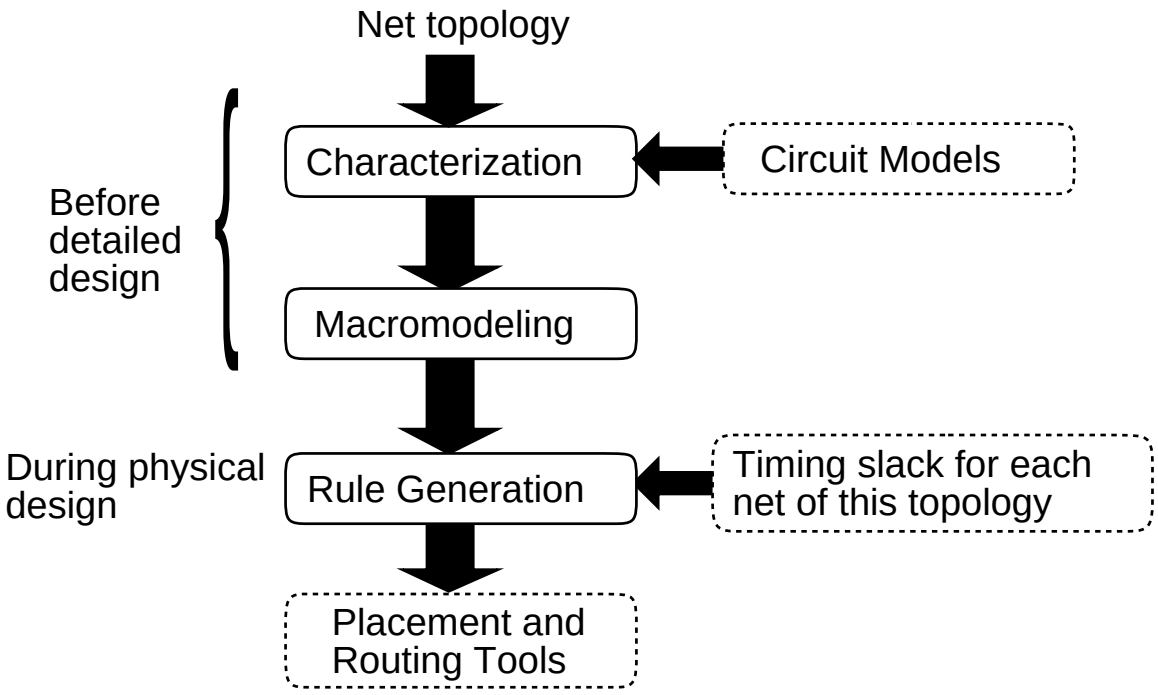


Figure 2: CAD Process Flow

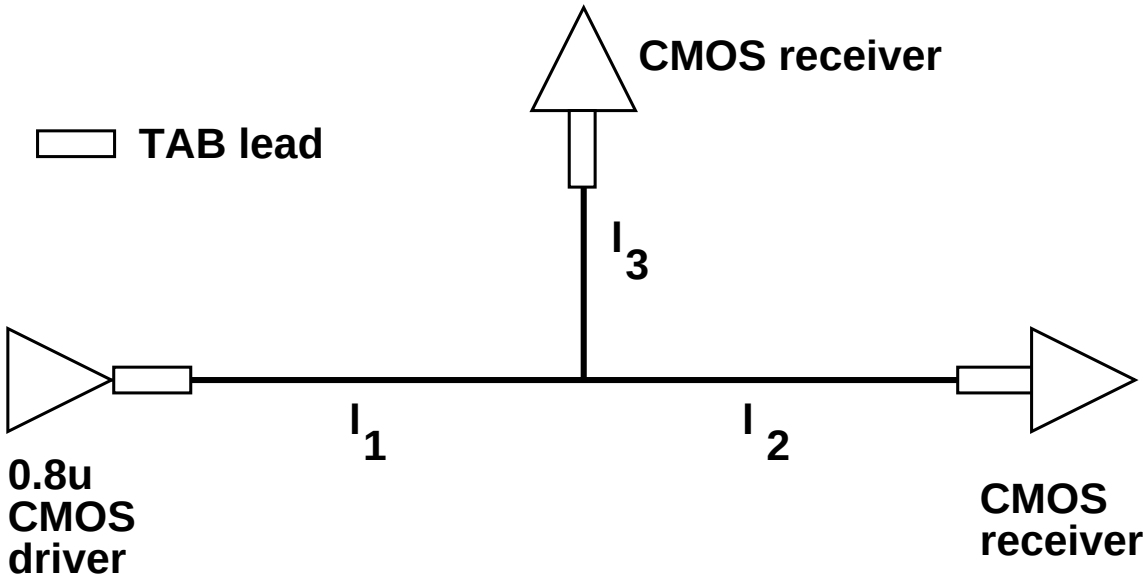


Figure 3: Example of a circuit topology, in this case a two receiver net. This case study is the one used throughout this paper.

Figure 4: Settling delay versus lengths for the net class given in Figure 3. This response surface was obtained from a 1331-point full parametric study. $l_1$ is fixed at 5 cm.

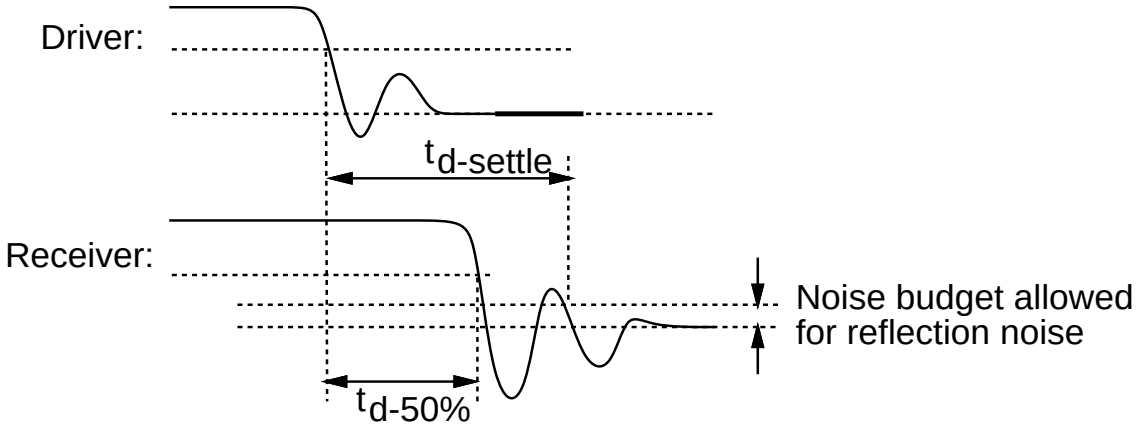


Figure 5: Definitions of settling delay $t_{d\text{-}settle}$ and 50% delay $t_{d\text{-}50\%}$.

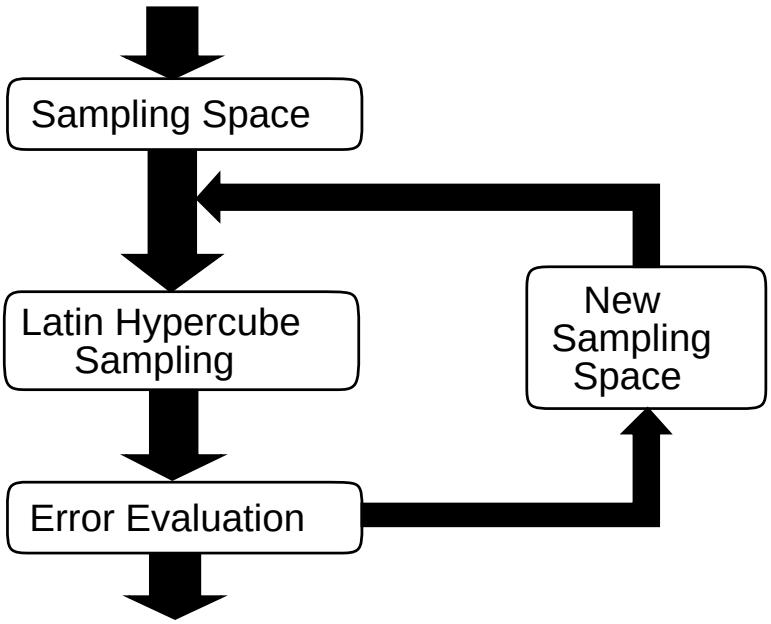


Figure 6: Steps in sequential sampling.

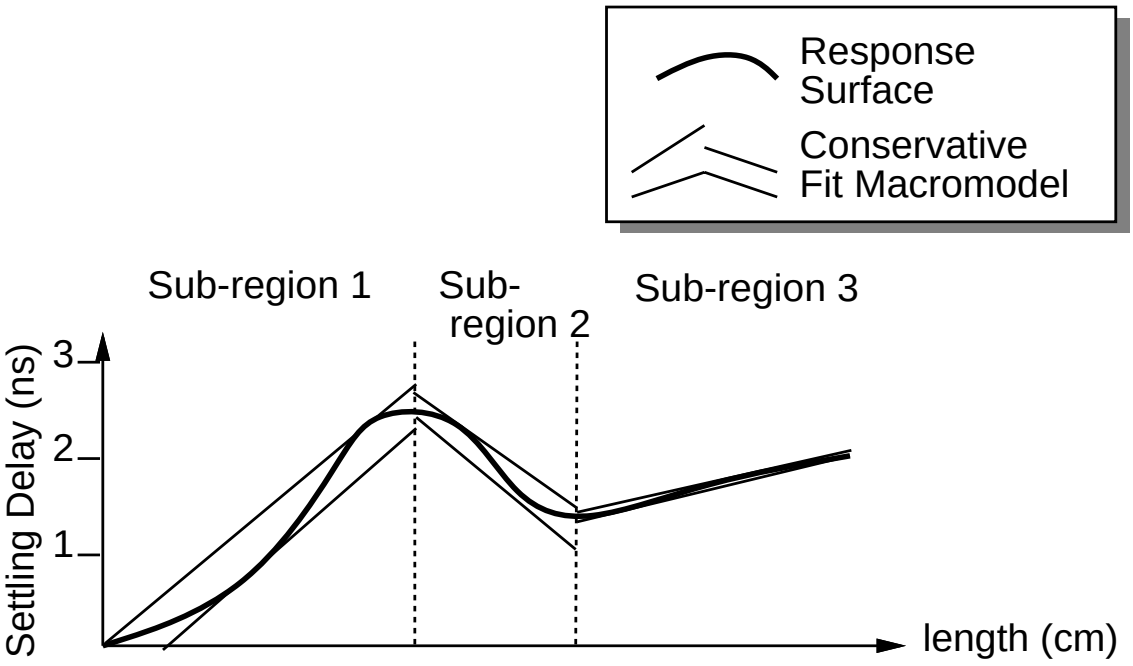


Figure 7: Simple example of a generated macromodel.

Table 1: Equations for 'above' macromodel for the entire initial region and two of the sub-regions.

| Region | Equation: $t_{settle}$(ns) = | Domain (cm) |
|---|---|---|
| One piece | $0.693 + 0.412 l_1 + 0.242 l_2 + 0.188 l_3$ | $0 \leq l_1, l_2, l_3 \leq 10$ |
| subreg 1 | $0.824 + 0.407 l_1 + 0.332 l_2 + 0.093 l_3$ | $0 \leq l_1, l_2, l_3 \leq 10$<br>$3.19 - 0.141 l_1 + 0.545 l_2 - l_3 \leq 0$<br>$2.00 + 1.13 l_1 - 0.511 l_2 - l_3 \leq 0$<br>$3.49 - 0.158 l_1 + 0.791 l_2 - l_3 \leq 0$<br>$-0.903 - 0.668 l_1 + 1.99 l_2 - l_3 \leq 0$ |
| subreg 2 | $2.67 + 0.586 l_1 - 0.188 l_2 + 0.195 l_3$ | $0 \leq l_1, l_2, l_3 \leq 10$<br>$3.19 - 0.141 l_1 + 0.545 l_2 - l_3 \leq 0$<br>$2.00 + 1.13 l_1 - 0.511 l_2 - l_3 \leq 0$<br>$3.49 - 0.158 l_1 + 0.791 l_2 - l_3 \leq 0$<br>$-0.903 - 0.668 l_1 + 1.99 l_2 - l_3 > 0$ |

Table 2: Flexibility and safeness coefficients for the rule produced by the 11-piece macromodel for settling delay.

| Approach | Flexibility Coefficient | Safeness Coefficient |
|---|---|---|
| 11-piece Macromodel | 53% | 91.2% |

Table 3: Comparison of the Standard approach with a 1-piece macromodel and a 10-piece macromodel.

| Approach | Flexibility Coefficient | Safeness Coefficient |
|---|---|---|
| Standard | 12.9% | 90% |
| 1-piece Macromodel | 14.6% | 99.7% |
| 10-piece Macromodel | 39% | 99.1% |