Design Methodologies for Noise in Digital Integrated Circuits

Kenneth L. Shepard
Department of Electrical Engineering
Columbia University, New York, NY 10027

Abstract

In this paper, we describe the growing problems of noise in digital integrated circuits and the design tools and techniques used to ensure the noise immunity of digital designs.

1 Introduction

Noise has become a metric in the design of digital integrated circuits of comparable importance to area, timing, and power for four principle reasons: increasing interconnect densities, faster clock rates, more aggressive use of high-performance circuit families, and scaling threshold voltages. Increasing interconnect densities imply a significant increase in coupling capacitance as a fraction of self-capacitance. Faster clock rates imply faster on-chip slew times. These two effects combine to make capacitive coupling a growing source of noise on-chip. Many high-performance circuit styles try to speed up one transition (usually falling) at the expense of the other and assign logical evaluates to the faster edge. Any circuit that utilizes these techniques we refer to as a *skewed-evaluate* circuit (e. g. domino). Skewed-evaluate circuits have noise sensitivities directly related to the threshold voltages of the transistors responsible for the evaluate transitions (usually n-FETs). Threshold voltages are, however, scaling lower to maintain drive in the presence of scaling supply voltages. These effects combine to produce more sources of on-chip noise due to switching circuits as well as less immunity to this noise. More details of these technology trends can be found in Reference [1].

Noise has two deleterious effects on digital design. When noise acts against a normally static signal, it can transiently destroy the logical information carried by the static node in the circuit. If this ultimately results in incorrect machine state stored in a latch, functional failure will result. When noise acts simultaneously with a *switching* node, this is manifest as a change in the timing (delay and slew) of the transition (a noise-on-delay effect[2]). We are concerned with the former effect in this paper.

Successful design methodologies to ensure noise immunity incorporate a three-level strategy. The first line of defense is a set of noise avoidance rules to guide circuit and interconnect design. These rules should prevent most noise problems but not introduce too many constraints on area or timing. This is followed by detailed static noise analysis on the design to find all possible noise failures. Because of the potential of false paths in static noise analysis, some failures flagged by static noise analysis can be allowed after careful circuit simulation.

DAC 98, June 15-19, 1998, San Francisco, CA USA
ISBN 1-58113-049-x/98/06...$5.00

2 Noise fundamentals

In digital circuits, an *evaluation node* is any node of the circuit that carries logical information by means of an analog voltage level. Evaluation nodes are generally chosen as the inputs and outputs of the logic gates of the circuit, which are generally abstractions for sets of channel-connected components (CCCs); that is, transistors connected through their sources and drains. Noise in the context of CMOS digital integrated circuits refers to any deviation from the nominal supply or ground voltages at nodes which should otherwise represent stable logic 1 or 0. Although noise causes these analog voltages to vary, the system still functions as long as the voltages fall into a valid range. If this is not the case, then the circuits' correct functioning cannot be certain. The complexity of noise analysis comes from the fact that the voltage ranges that represent valid logic levels depend on the precise time-domain characteristics of the noise appearing on the evaluation nodes as well as the sensitivity of receiving circuits to this noise.

The *noise stability* metric is used to determine if the noise appearing on a given evaluation node is of sufficient magnitude to risk functional failure. Noise stability requires that each restoring logic gate, when acted upon by a noise stimulus, must have a dc-noise time-domain sensitivity that is always less than one[1, 3]. This is a generalization of dc noise margins[4] to take into account the fact that restoring logic gates are much more effect at rejecting ac noise than dc noise.

It is convenient to classify noise according to the voltages' relationship to the rails: V_H noise reduces an evaluation node voltage below the supply; V_H^* noise increases an evaluation node voltage above the supply level; V_L noise increases an evaluation node voltage above the ground level; and V_L^* noise decreases an evaluation node voltage below the ground level. These noise classifications are useful because circuits generally propagate noise types in well-defined way. For example, a CMOS inverter is sensitive to V_L and V_H noise on its input, propagating it as V_H and V_L noise, respectively, to its output. We refer to V_H^* and V_L^* noise as *bootstrap* noise.

To develop a comprehensive strategy for noise analysis, we must consider all the possible sources of noise on chip. For this purpose, noise can be classified as power-supply noise, circuit noise, or interconnect noise. Power-supply noise refers to noise appearing on the on-chip power and ground distribution network produced by the current demands of the switching circuits. Circuit noise includes noise propagated onto an evaluation node from the driving gate or charge-sharing effects onto the output of the driving gate[1]. For silicon-on-insulator (SOI) circuits, circuit noise might also include the effects of the parasitic bipolar in floating-body MOS devices[5], but we will not consider SOI devices

further here. Interconnect noise refers to noise appearing as a result of capacitive or inductive coupling in the interconnect or as a result of poor impedance matching between the devices and interconnect.

3 Noise-reducing design techniques

Designers can use a number of techniques and practices to control noise in digital circuits. Controlling power supply noise involves careful design of the power and ground distribution network and adequate decoupling capacitance. To achieve effective supply decoupling, the ratio of on-chip decoupling capacitance to total effective switching capacitance is generally greater than 10:1[6]. To control circuit and interconnect noise, circuit design rules are commonly employed. Some examples: disallow single n-FET or p-FET pass gates because of the V_T voltage drop they cause; disallow pass gates at the ends of long wires; disallow long wire runs feeding domino gate inputs; and disallow high beta static circuits feeding low beta static circuits, or vice versa.

In addition, design rules can require "baby-sitting" devices added to internal nodes in n-FET pull-down stacks of domino gates to mitigate charge-sharing noise, and half-latches added to dynamic nodes. Rules of this sort are sometimes accompanied by a circuit-checking tool. In addition to tuning gate widths, one can tune gate lengths to balance performance against leakage noise. Weaker devices at the top of the n-FET stack can also reduce charge-sharing noise in the presence of a half-latch device.

In addition to circuit techniques and optimization, there are also many ways to reduce interconnect noise in the design of the on-chip wiring network. For example, one can increase spacing between wires or route signal lines alternately with power or ground. In the Digital 21264 design[7], two metal layers are devoted exclusively for the distribution of power and ground. These ground planes more tightly control on-chip inductance and reduce coupling interactions.

4 Static noise analysis

Design rules can be used to prevent most noise problems in the design phase but are very dependent on design style and circuit topology. Making them rigid enough to preclude noise failures would be exordinarily difficult and result in a significant penalty to area and performance. Instead, design rules are augmented with a *static noise analysis* of the design. In this type of analysis, the noise stability of a ditial circuit is determined by performing small simulations on an individual CCC basis and combining these simulation results statically.

Several fundamental assumptions guide the technique:

1. Gate inputs can be replaced by grounded capacitors. This creates a clean partitioning between one CCC and the next and is a technique commonly employed in fast circuit simulation engines[8].

2. Worst-case *sensitization* conditions drive the CCC simulations used for calculating circuit noise and interconnect noise. By this, we mean how the transistor gates are biased and internal node voltages are preset before the noise stimulus or switching waveform is applied. We will describe the conditions that guide this sensitization in more detail below.

3. We assume that the superposition principle applies in adding (in the time domain) circuit noise and interconnect noise sources. For noise sources small enough to satisfy the noise stability requirement, active FET channels (that is, those attempting to hold nodes to their static level) are biased in the triode regions of their current-voltage characteristics, justifying this "linear" assumption. In particular, charge-sharing noise and propagated noise can be calculated on a "single-input" changing basis and superposed with the coupled noise calculations to find the total noise. The sensitization producing the producing the largest amplitude output noise is used. Noise sources can only be combined when the sensitization conditions are mutually satisfiable.

4. Worst-case temporal relationships are defined by superposing the peak responses of the circuit and interconnect noise for each allowable noise (V_L, V_H, V_L^*, V_H^*) type. When timing information is known, it can be used to reduce pessimism in coupled noise analysis by disallowing the simultaneous switching of signals. Hazard-free logic constraints can also be used to disallow simultaneous switching.

5. A noise stability check is performed across every restoring logic gate in the design. Noise stability violations are assumed to be a sufficient condition for finding the circuit to be nonfunctional.

6. Power-supply integrity analysis is performed independently and is generally characterized by dc bounds on the local power supply variation. In calculating propagated noise, collapsed rails are used, characterized by dc values V_{DD}^{min} and Gnd^{max}. For switching waveforms producing charge-sharing or coupled noise, expanded rails are used, characterized by dc values V_{DD}^{max} and Gnd^{min}. In doing a noise stability check, expanded rails are also used.

7. Drivers on switching perpetrator nets (which we refer to as *secondary nets*) are modelled as ideal voltage sources. This presumes that the noise-on-delay effect has been handled elsewhere.

8. In the case that the circuit contains feedback (as in a latch circuit), the feedback loop is broken at a restoring logic gate. The worst possible dc noise that can be propagated without producing a stability violation in the "broken" gate is assumed. To determine the magnitude of this dc *noise-limited* propagated noise, the subunity gain criterion is applied to a dc voltage transfer characteristic with supplies defined by V_{DD}^{min} and Gnd^{max}. The use of noise-limited propagation in one restoring gate of a logic loop breaks the cyclic dependence in determining the noise potentially circulating the loop.

9. Noise-limited propagation is also used for restoring logic gates which have a stability violation. This allows the noise analysis to continue, despite the violation, and places the burden on fixing the noise problem on the circuits driving the violating gate.

The key abstraction in static noise analysis is the noise graph, a directed graph containing all of the circuit's evaluation nodes connected by segments that move and transform noise. In many ways, this graph is analogous to the timing

graph used in static timing analysis. There are three types of segments in a noise graph: restoring segments, propagate segments, and node injection segments. *Restoring segments* cross gates that at some dc bias point have a small-signal gain greater than one. Noise is propagated across restoring segments; in addition, a noise stability check must also be performed. *Propagate segments* connect nodes, between which there is subunity gain at all dc bias points. Noise-stability checking is not required across propagate segments. Each restoring and propagate segment in the noise graph is labelled by the type of noise propagated by the segment. For example $L \rightarrow H$ indicates that the segment propagates V_L noise and transforms it into V_H noise. The *node injection segment* can introduce noise directly onto an evaluation node, superposing with the propagated noise. Coupled interconnect noise and charge-sharing noise are both modelled as node-injection segments. Once the noise graph is constructed, the loops of the graph are broken and the graph is topologically sorted for traversal. Assumption 8 is used to propagate noise across each of the "loop snips." The graph is then searched in a breadth-first fashion to propagate noise through the network, and in the case of restoring segments, to perform the sensitivity tests required to ensure noise stability.

In general, transistor path-based functional extraction[9] guides three main types of sensitization required for Assumption 2: sensitization for coupled noise calculation on the output node of a CCC, sensitization for noise stability and propagated noise calculation from a given input, and sensitization for charge-sharing noise calculation from a switching waveform on a given input. In all three cases, allowable sensitizations are determined by the Boolean satisfiability of constraint relations determined by this functional extraction. Logic conditions between the input variables (denoted as f_{input}), when they exist, must be included in these constraint relations. The constraint equations differ slightly for the case of precharge logic to account for the state established by the reset condition.

We define a path function $f_{P_{i,j}}$ as the logical condition for the path from i to j to conduct. For example, to sensitize for noise appearing on the output, O, of a CCC due to capacitive coupling to a given node, D, in the CCC, there are different logic constraint functions depending on whether we are interested in V_H or V_L noise at the output. In particular, let us consider the sensitizations that allow V_L noise to appear on O due to capacitive coupling to D. In this case, the V_L noise at O is produced by a net switching from ground to V_{DD}, capacitively coupled to node D. The constraint relation in this case is: $f_{P_{O,Gnd}} \cdot f_{P_{O,D}} \cdot f_{input}$. All input sensitizations must satisfy this constaint to inject V_L noise onto node O due to coupling onto node D from a net switching from ground to V_{DD}. Similar sensitization relations exist for charge-sharing noise and propagated noise.

In general, to find the noise appearing at the output of a given CCC, according to Assumption 3, we must find the sensitization producing the largest amplitude output noise for each noise type (V_L or V_H). Stability violations that occur due to noise appearing on any input are reported independently of the sensitization conditions required for the maximum output noise.

We seek to justify and clarify the concepts of static noise analysis by means of a simple, though comprehensive, example. Consider the circuit of Figure 1, in which a domino gate drives a latch. The dynamic node (E) of the domino gate is capacitively coupled to another switching net. In Figure 2, we show circuit simulation results of a functional fail that results from noise. As shown in Figure 2(a), A switches high, while B is still low, resulting in charge-sharing noise on node E. Node I switches, adding coupled noise onto node E. At the same time, noise appears on nodes C and D as shown in Figure 2(b). Power-supply noise is causing the rails to expand during the evaluate phase of the domino gate as can be expected to typically happen since the voltage rails often collapse transiently during precharge. These noise sources *together* are enough switch the output of the dynamic gate, F, and change the state of the latch (node G), as shown in Figure 2(c). Since the latch should have a logic 1 as its output, but instead has a logic 0, functional failure of this hardware will result.

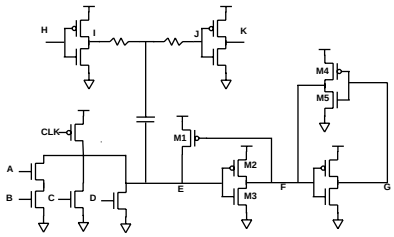

Figure 1: Example circuit for noise analysis consisting by a domino gate driving a latch.

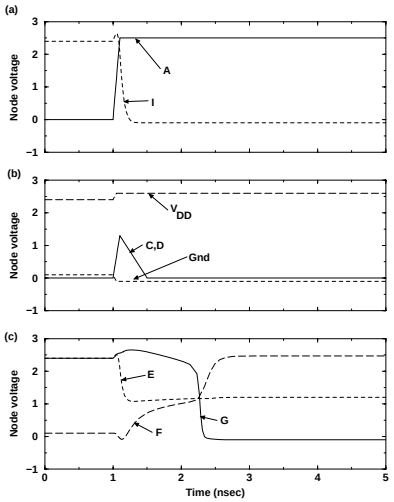

Figure 2: Circuit simulation of a functional failure due to noise. (a) Driver output I is switching as is node A. (b) Coupling noise appears on inputs C and D along with power-supply noise on the voltage rails. (c) The dynamic node (E) falls, switching the output inverter of the domino gate (F) and the latch output (G).

The noise failure demonstrated in Figure 2 is critically dependent on the contributions of all of the noise sources at work: power-supply noise, charge-sharing noise, coupling noise, and propagated noise. Figure 3 shows how this noise fail would not occur in the absence of any of these noise sources. Figure 3(a) shows the node voltages E, F, and G in the absence of input noise on either node C or node D. In Figure 3(b), we show the voltages in the case that there is no coupling noise; that is, node I does not switch. In Figure

3(c), we show the voltages in the case that node A does not switch. We show the voltages in the case that there is no power supply noise in Figure 3(d).

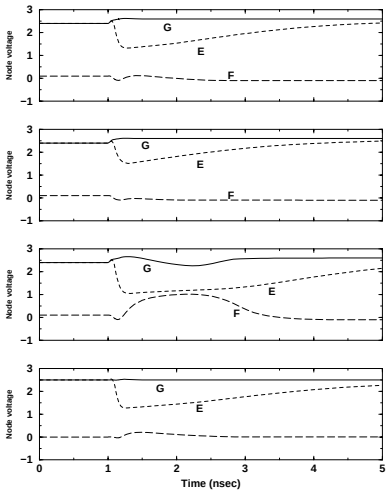

Figure 3: Noise failure will not occur in the absence of (a) injected noise on either input C or input D, (b) coupling noise, (c) charge-sharing noise, or (d) power-supply noise.

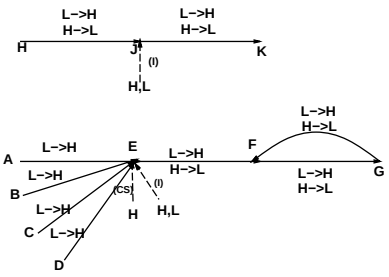

Figure 4: Noise graph for the example circuit.

We now show how a static noise analysis of this network precisely predicts this fail using the assumptions outlined above. The noise graph for this circuit is shown in Figure 4. Restoring segments are denoted by solid lines, while dashed lines denote node-injection segments. We must first calculate the worst-possible noise which can appear on node E according to Assumption 4. This will be, according to Assumption 3, the superposition of the charge-sharing noise injected by the switching of node A, the noise propagated from node C, the noise propagated from node D, and the interconnect coupling noise injected by the switching of node I.

To calculate each of these noise components, we establish the network shown in Figure 5. By Assumption 1, gate inputs are treated as linear capacitors tied to ground. The driver at I is replaced by a independent voltage source by Assumption 7. We now calculate each noise source independently. The sensitization producing the worst total V_H noise response at node E has A, C, D, and F all set to 0 and B set to 1. CLK is 1 for the evaluate phase of this dynamic gate. In Figure 6(a), we show the coupling noise appearing on node E. In this case, node I is switched from

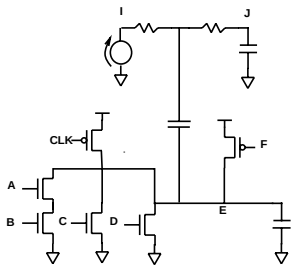

Figure 5: Network for the simulations to compute the noise at node E.

V_{DD}^{max} to Gnd^{min}. V_{DD}^{min} is used for supply and Gnd^{max} is used for ground for the CCC itself, following Assumption 6. In Figure 6(b), we show the charge-sharing noise calculated on node E due to the switching of node A. Figure 6(c) shows the noise propagated to node E due to V_L noise injected onto node C. The exact same curve results for V_L noise propagated to node E from node D under comparable sensitization conditions. In Figure 6(d), we show the noise that results by superposing in the time-domain the results of graphs (a)-(c) with (c) contributing twice. Following Assumption 3, all of the peak noises are aligned. We compare this result with a full simulation, shown as the dashed curve in Figure 6(d). They are almost the same, demonstrating the accuracy of the superposition aspect of Assumption 3.

Having calculated the total noise appear on node E, following the noise graph, we must now propagate this noise across the inverter and perform a noise stability check, following Assumption 5. Figure 7 shows the network for this analysis. Node G is set to 1. The capacitor C_1 represents the gate capacitance of transistor M1, while C_2 represents the gate capacitance of transistors M2 and M3. In Figure 8, the noise propagated to node F is shown as well as the time-domain sensitivity of this noise to dc variations on node E. The sensitivity at the peak position is almost exactly -1, indicating that the inverter is biased to the verge of a noise instability. Static noise analysis would then flag this as a possible noise failure.

Given that a noise instability occurs in propagating to node F, a noise-limited value of this propagated noise should be used at node F, following Assumptions 8 and 9. The dc voltage transfer characteristic of the inverter between nodes E and F is used for the calculation of the noise-limited propagated noise at node F, in this case, 0.23 V. In handling the latch, the restoring segment from G to F is "snipped". Noise limited propagation is assumed out of the inverter defined by transistors M4 and M5. This noise is superposed with the noise propagated from E to F, and the sum propagated across the restoring segment from F to G with the associated noise stability check. To complete the analysis, a noise stability check is also performed at the M4-M5 inverter with the noise propagated to node G.

There are three main sources of pessimism in this analysis. The first is the conservatism of the metric; that is, a circuit can still function even in the presence of a noise instability. For example, in the circuit of Figure 1, the noise instability detected across the M3-M4 inverter would not result in an erroneous value stored in the latch in the absence of the half-latch M1 device. Despite this fact, the noise instability violation is indicative of a serious design weakness which should be corrected. The second source of pessimism comes from the worst-case temporal correlation of Assump-

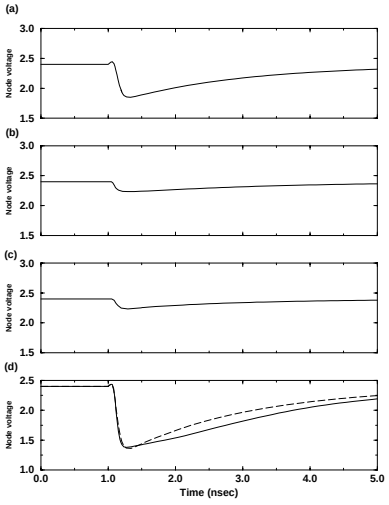

Figure 6: Noise calculation at node E: (a) coupled noise due to switching driver at I; (b) noise due to charge-sharing from switching input A; (c) noise propagated from node C or D; and (d) superposed noise. The solid curve in (d) comes from a strict time-domain sum. The dashed curve shows the exact result from circuit simulation.

tion 3. Not only are we assuming that all possible sources of noise are acting, but that they are acting together in time in the worst possible way. False paths can exist in static noise analysis which can be disallowed by both timing constraints and logic constraints as described in more detail in Section 7. The last major source of pessimism comes from Assumption 8. Using noise-limited propagation at a loop snip can introduce more noise into the circuit than would be introduced if the actual noise appearing on the input of the loop segment were propagated to the output. Because of these sources of pessimism, some fails identified by static noise analysis may be allowed after careful circuit simulation.

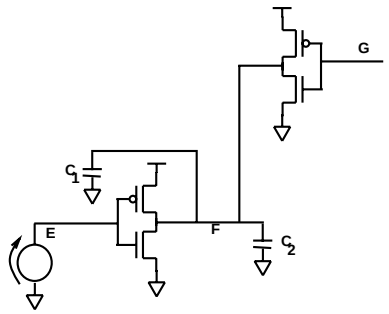

Figure 7: Network to perform the stability check and propagate the noise from node E to node F.

5 Power supply integrity analysis

Power and ground integrity analysis is also an important part of noise verification. There are two components to power-supply noise. The first is the variations in the dc power supply and ground levels due to the resistance of the power and ground distribution network. To calculate these IR drops, we can apply a separate analysis of the chip's spatial current demands against a supply- and ground-rail resistance extraction[10]. A second type of power supply noise is delta-I noise, produced by the simultaneous switching of off-chip drivers and internal circuits, usually synchronized with clock activitiy. Detailed transient analysis of the power grid involves applying models of the circuits' current demands to a detailed RLC extraction of the power grid combined with the package model[11, 10]. The current models usually take the form of Norton equivalent circuits at designated points in the power or ground distribution hierarchy, usually on a designated via layer.

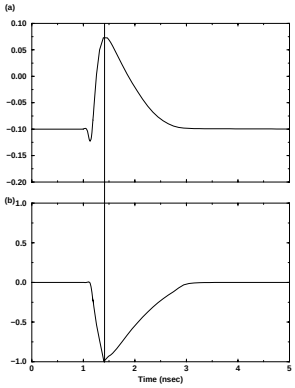

Figure 8: Noise calculation at node F: (a) propagated noise at node F, and (b) time-domain sensitivity of this noise to dc noise at the input.

6 Interconnect analysis

Central to noise analysis is the analysis of large coupled RLC networks. Two key ideas are central to this analysis: net complexes and hybrid admittance-transfer multiport macromodels. The net complex as shown in Figure 9 allows one to create a local environment to analysis the coupling for a given net in the design[12]. The primary net of the complex is the one for which we are trying to calculate the noise. The complex also includes secondary nets with significant coupling to the primary net. Coupling between the significant secondary nets and nets other than those already in the net complex are grounded and disconnected so that no current flows through them. This means that mutual inductances to these nets are discarded and coupling capacitors are treated as capacitors tied to ground. When ports of the resulting net complex terminate on MOS gates, we replace these loads with linear capacitors, which allows us to reduce the number of driving ports. At these "tap points", we instead calculate a transfer frunction from each of the ports. Krylov-subspace reduced-order modelling techniques, such as Arnoldi[13] and Lanczos[14], can be used to reduce these net complexes with ports and taps into hybrid admittance-transfer multiport macromodels.

Because of the performance limitations of interconnect RC delays, much practical effort has been focussed on reducing them through technology and design. Next generation technologies can be expected to make increasing use of low-resistivity metals (copper) and low-dielectric-constant insulators. Designers increasingly use wide, thick upper level metal for power and ground distribution and for long signal

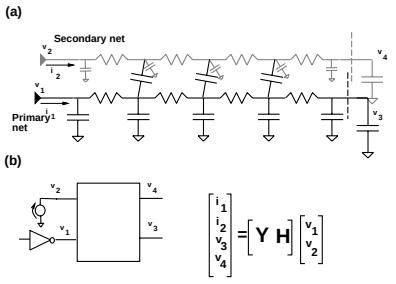

Figure 9: Modelling of interconnect: (a) a typical net complex consisting of a primary net coupled (in this case) to a single secondary net, (b) the hybrid admittance-transfer multiport model for this net complex

runs. These efforts, however, have resulted in the growing importance of inductance and inductive coupling in the timing and noise and noise analysis of signal lines for three important reasons. First, inductance must be incluced to accurately predict rise and fall times and delays in timing analysis. Secondly, if an inductive net is overdriven, an underdamped ringing response can be observed. This ringing is noise to subsequent receiving circuits and can potentially result in functional failure of the design. Lastly, inductive coupling along with capacitive coupling can be a significant source of noise on quiet nets due to the switching of nearby perpetrators. In the case of inductive interconnect, when a ringing response is possible, the temporal relationship that guides superposition in static noise analysis is an open question.

7 False paths in static noise analysis

False paths in static noise analysis correspond to cases in which signals cannot switch simultaneously because of non-overlapping arrival time windows or because of hazard-free logic constraints. Within the context of static noise analysis, these cases are easy to detect within the analysis associated with a single CCC. More complex situations, however, span multiple CCCs. Consider again the example of Figure 1 but include the circuits driving nodes C and D as shown in Figure 10. Now assume that the source of the noise appearing at C and D is charge-sharing noise due to the switching of M and K from 0 to 1 with L and N set to 0. Timing orthogonality[12] or hazard-free logic constraint information for switching signals H, A, M, and K could now be used to limit the pessimism associated with the noise calculation at node E. These false paths are usually eliminated with designer involvement.

8 Conclusions

Noise immunity has become of comparable importance to area, timing, and power in the design of digital integrated circuits. The noise stability criterion can be used to recognize possible failures and static noise analysis provides a technique for checking this criterion on a chip-wide basis. A successful design methodology for noise combines static noise analysis with design rules and power-supply integrity analysis. Hybrid multiport admittance-transfer macromodels are used to model the coupled RLC on-chip interconnect networks.

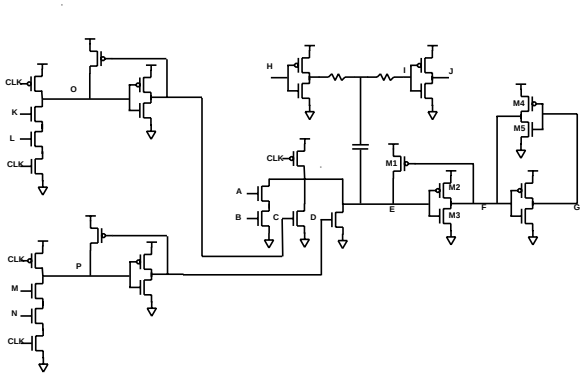

Figure 10: Figure 1 with the driving circuits included.

References

[1] K. L. Shepard and V. Narayanan. Conquering noise in deep submicron digital design. *IEEE Design and Test of Computers*, pages 51 – 62, January/March 1998.

[2] F. Dartu and L. T. Pileggi. Calculating worst-case gate delays due to dominant capacitance coupling. In *Proceedings of the IEEE/ACM Design Automation Conference*, pages 46 – 51, June 1997.

[3] K. L. Shepard and V. Narayanan. Noise in deep submicron digital design. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pages 524–531, San Jose, CA, November 1996.

[4] L. A. Glasser and D. W. Dobberpuhl. *The Design and Analysis of VLSI Circuits*. Addison-Wesley, 1985.

[5] Pong-Fei Lu et al. Floating-body effects in partially depleted SOI CMOS cicuits. *IEEE Journal of Solid-State Circuits*, 32(8):1241–1253, August 1997.

[6] D. Dobberpuhl et al. A 200 MHz 64b Dual-Issue CMOS Microprocessor. *IEEE Journal of Solid-State Circuits*, 27(11):1555 – 1567, 1992.

[7] Bruce A. Gieseke et al. A 600 MHz superscalar RISC microprocessor with out-of-order execution. In *Proceeding of the International Conference on Solid-State Circuits*, pages 176 – 177, 1997.

[8] Anirudh Devgan and Ronald A. Rohrer. Adaptively controlled explicit simulation. *IEEE Trans. CAD*, 13(6):746 – 761, 1994.

[9] A. Kuehlmann, A. Srinivasan, and D. P. Lapotin. Verity – a formal verification program for custom CMOS circuits. *IBM Journal of Research and Development*, 39(1/2):149 – 165, 1995.

[10] K. L. Shepard, S. Carey, E. Cho, B. Curran, R. Hatch, D. Hoffman, S. McCabe, G. Northrop, and R. Seigler. Design Methodology for the G4 S/390 Microprocessors. *IBM Journal of Research and Development*, 21(4/5):515 – 548, 1997.

[11] H. H. Chen. Minimizing chip-level simultaneous switching noise for high-performance microprocessor design. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, volume 4, pages 544 – 547, 1996.

[12] K. L. Shepard, V. Narayanan, P. C. Elmendorf, and Gutuan Zheng. Global Harmony: Coupled Noise Analysis for Full-Chip RC Interconnect Networks. In *Proceedings of the IEEE International Conference on Computer-Aided Design*, 1997.

[13] L. Miguel Silveira, Mattan Kamon, Ibrahim Elfadel, and Jacob White. A Coordinate-Transformed Arnoldi Algorithm for Generating Guaranteed Stable Reduced-Order Models of RLC Circuits. In *IEEE/ACM International Conference on Computer-Aided Design*, pages 288 – 294, San Jose, CA, November 1996.

[14] Peter Feldmann and Roland W. Freund. Reduced-order modeling of large linear subcircuits via a block lanczos algorithm. In 32^{nd} *ACM/IEEE Design Automation Conference*, pages 474–479, San Francisco, California, June 1995.