SubWave: a Methodology for Modeling Digital Substrate Noise Injection in Mixed-Signal ICs

Paolo Miliozzi, Luca Carloni, Edoardo Charbon†, and Alberto Sangiovanni-Vincentelli

Department of Electrical Engineering and Computer Sciences,
University of California, Berkeley, CA

† Cadence Design Systems, Santa Clara, CA

Abstract

A methodology is presented for generating compact models of substrate noise injection in complex logic networks. For a given gate library, the injection patterns associated with a gate and an input transition scheme are accurately evaluated using device-level simulation. Assuming spatial independence of all noise generating devices, the cumulative switching noise resulting from all injection patterns is efficiently computed using a gate-level event-driven simulator. The resulting injected signal is then sampled and translated into an energy spectrum which accounts for fundamental frequencies as well as glitch energy. Preliminary results demonstrate the validity of the assumptions and the accuracy of the approach on a set of standard benchmark circuits.

1 Introduction

With the continuous increase of chip complexity, device density and circuit speed, noise immunity has become a major source of concern in the design of reliable VLSI ICs. Due to the decline of the distance between high-swing high-frequency noise sources and sensitive devices, the substrate has become a major carrier of spurious signals. In mixed-signal circuits in particular, coupling of analog, relatively slow signals with quickly switching digital ones is often disastrous. To alleviate the problem, heavily over-designed structures are generally adopted, thus seriously limiting the advantages of innovative technologies. For this reason substrate modeling has received attention from mixed-signal circuit designers, attempting to integrate RF analog and baseband digital circuitry on the same chip.

In digital circuits a large number of gates undergoes a transition periodically. When a transition occurs a spike of current is absorbed from a power bus and used to charge a load in the signal path. In general, a significant portion of this current is discharged to a ground bus through direct feedthrough, or it is injected directly into the substrate through various mechanisms [1]. The cumulative contribution of currents injected by switching gates is felt in sensitive circuits in a form of a spurious signal, referred to as *switching noise*.

The spatial location and rate of occurrence of transitions in digital circuits are generally time-variant and difficult to characterize exactly. In addition, due to the complexity of the circuits, an exact waveform characterization of switching noise is impractical. For this reason, switching noise is often modeled based on its macroscopic appearance. For example, if the number of switching gates is large enough and the global switching activity of the digital circuit is uniformly distributed over a large section of the spectrum, the switching noise can be modeled as a single Gaussian white or pink noise source. An alternative modeling technique, proposed in [2, 3], consists of representing switching noise as a capacitively and/or resistively coupled signal derived from the global clock of the circuit.

Switching noise models in this compact form are useful to efficiently estimate the impact of an architecture or a floorplan to system performance, and they can be used to drive a number of global optimization tools. However, the approximations for injected switching noise using such a method often capture only a relative small portion of the entire noise energy spectrum. Thus, potentially detrimental noise components may be underestimated or ignored. Moreover, with the emergence of submicron technologies, the problem of high-speed substrate noise interference may compromise performance at the nominal frequencies for which the circuits are designed.

In this paper a novel method is proposed for accurate and efficient characterization of switching noise generated by the logic components of mixed-signal ICs. The method is structured as follows. First, the noise injected in the substrate from a certain gate with a given input transition pattern is accurately simulated using detailed extraction. Call *noise signature* such a signal. Second, an event-driven simulator computes the rate of occurrence for each noise signature cumulatively over all the gates present in the circuit. Third, the noise signal associated with each signature is evaluated over the entire simulation time span. Fourth, the energy spectrum of the resulting signal is computed using the Fast Fourier Transform (FFT). Figure 1 shows all the phases of Sub-

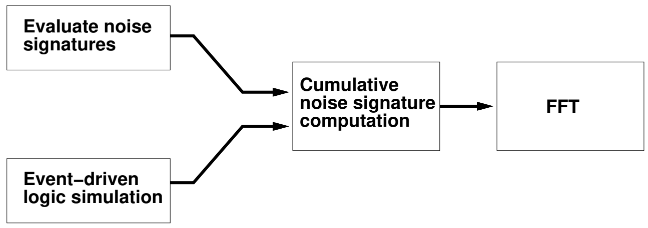

Figure 1: Flow diagram of SubWave

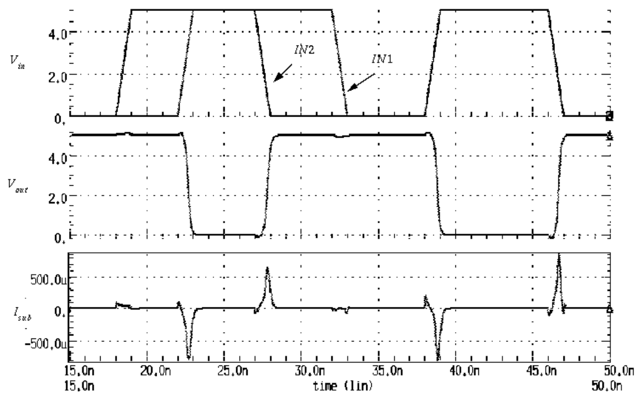

Figure 2: Substrate noise signature of the nand gate

Wave.

There are several advantages to this procedure. First, the substrate noise model is compact and, although accurate, it can be translated into a single noise signature which is unique for a given digital component and hence it can be re-used in future re-design loops. Second, the energy spectrum captures the energy associated with glitches and other high-speed phenomena. Finally, due to the efficiency of the event-driven evaluation techniques and of the FFT, a rapid characterization of injected noise can be obtained and possibly drive the synthesis of *silent* digital circuits.

The paper is organized as follows. Section 2 describes the techniques for the generation of the noise signatures associated with the gates and corresponding input transition schemes. In section 3 the event-driven simulator and the cumulative noise computation are described in detail. Finally, in section 4 a number of experiments on a number of benchmarks is presented and discussed.

2 Modeling Substrate Injection

Substrate noise is injected mainly through devices and capacitive coupling from interconnects.

Device noise injection is caused by impact ionization and drain/source-substrate junctions. The effects of inverse biased n/p junctions are effectively modeled through non-linear capacitances. Impact ionization on the contrary, is a highly localized phenomenon due to high electric fields experienced in the drain-channel-substrate interface. Impact ionization is particularly important in sub-micron technologies, where small channel size and a reduced oxide thickness cause electric fields to exceed the critical value of $4 \cdot 10^4 V/cm$, thus resulting in large electron-hole pair generation. Impact ionization currents can be expressed through the following equation [4]

$$I_{impact} = C_1(V_{ds} - V_{dsat}) I_d exp\left(-\frac{C_2 t_{ox}^{1/3} x_j^{1/2}}{V_{ds} - V_{dsat}}\right).$$

Device level simulations have been performed to extract parameters C_1, C_2 using the 2-D device level simulator Pisces [5]. These parameters have been then used in standard Spice models.

There exist two main classes of substrates: one referred to as *low-resistivity substrate* which consists of a thick, high-resistivity epitaxial layer (d $\simeq 10\mu m$, $\rho \simeq 10 \div 15\ \Omega cm$) and a low-resistivity bulk($\rho \simeq 1m\ \Omega cm$). A second class, referred to as *high-resistivity substrate*, is composed of a uniformly doped layer with a resistivity coefficient of $20 \div 50\ \Omega cm$. Recently, low-resistivity substrates have been widely adopted for desirable latch-up suppression properties [4]. In general, it has been found that at low and medium frequencies, typically less than $5GHz$, substrate shows a resistive behavior. At higher frequencies on the contrary, the transport patterns are too complex to be accurately modeled using RC meshes [1].

The gate count of realistic digital circuits is typically in the millions. A complete extraction and detailed simulation of each individual substrate noise injector is impractical. If the substrate underlying the circuit can be approximated to be equipotential, then simultaneous substrate currents injected in different locations contribute in a cumulative fashion to the spurious potential sensed remotely. Each transition in the input pattern of each gate causes it to inject a unique noise signature. Hence, a set of S_C *signature classes* can be identified for a circuit, with $S_C \leq \sum_{c=1}^{N_L} s_c$, where N_L is the size of the technology library and s_c is the number of unique signatures per gate. Assuming that the activity of each signature class is known, then the resulting injection patterns due to all gates can be cumulatively evaluated and propagated to the sensing point.

As shown in Figure 2, when a circuit is realized with basic logic gates (nand, nor, not,..), two classes are necessary to capture the behavior of each gate with respect to substrate injection. In fact, it can be seen that the injection depends only on the output transition (rise and fall).

For a given technology a table (*Wave_Table*) is generated which stores, for each class, a waveform obtained from a Spice simulation.

```
SIMULATE(Mapped_Network, Input_Vectors)
  foreach vector k ∈ Input_Vectors
    foreach node p ∈ PI
      SetNode(p, t_0^k, input_value(p))
      foreach node n ∈ FO(p)
        Mark(n, (t_0^k + wire_delay(p, n)))
    foreach instant t ∈ T^k
      foreach node n ∈ (DFO − PI)
        if IsMarked(n, t))
          SimulateNode(n, t)
          if (Event(n, t) = Rise or Event(n, t) = Fall)
            g_n ← gate_of(n)
            UpdateGateTable(g_n, Event(n, t), t)
            foreach fanout n' ∈ FO(n)
              Mark(n', t + wire_delay(n, n'))
          else
            SetNode(n, t, value_at_time(t − 1))
        else
          SetNode(n, t, value_at_time(t − 1))
  return Gate_Table
```

Figure 3: Event-driven logic simulator

3 Event-Driven Logic Simulation and Cumulative Noise Signal Computation

The simulator is described in Figure 3. The technology mapping of the Boolean network associated with the circuit (*Mapped_Network*) and a sequence of inputs (*Input_Vectors*) are passed to the simulator. The simulator produces the switching activity of each signature class as an output *Gate_Table*. The notation used in Figure 3 is the same as the one used in [6].

Let $\mathcal{N}$ be the set of nodes of the Boolean network, f_n the logic function associated with node n, and y_n the logic variable associated with n. Let us define $\mathcal{PI}$ as the primary input set, $\mathcal{FO}(n)$ as the set of fan-out of a node n, and $\mathcal{TFI}(n)$ as the set of transitive fan-in of a node n. Moreover, it is useful to introduce the ordered set $\mathcal{DFO}$, which contains all nodes in *Depth-First Order* from the outputs.

Let assume that every node n has been bound to a basic logic gate g_n by the technology mapping procedure. Hence, every arc in the graph is individually associated with a wire connecting two different gates of the circuit. Furthermore, for each gate g_n two signature classes, one for the rise and one for the fall transition are necessary.

SIMULATE performs internally an event-driven gate level simulation based on a *pure bounded wire delay model* [7]: in event-driven simulation each change in value of a wire in the circuit is regarded as an event, while a pure bounded wire delay model implies that there is exactly one delay element per gate input. Therefore, for a given sequence of input vectors, SIMULATE determines the output vector for each input vector, by evaluating for each node n the Boolean equation f_n, after considering the delay elements associated with g_n. This model permits to consider implicitly and accurately every possible spurious transition (*glitch*) in the circuit.

SIMULATE introduces a discretization of time based on the delay informations available from the gate library and the network topology. Let $t_0 = 0$ be the instant when the simulation begins, t_{max} the delay of the longest path in the network, and K is the number of input vectors, then the simulation interval is $\mathcal{T} = [t_0, t_{end}]$, with $t_{end} = K \times t_{max}$. Moreover, while processing the k^{th} vector, just the sub-interval $\mathcal{T}^k = [t_0^k, t_{end}^k]$, where $t_0^k = (k-1) \times t_{max}$ and $t_{end}^k = k \times t_{max}$, has to be considered.

Suppose node m changes its output at t, each fanout n of m is marked by $\mathbf{Mark}(n, t')$, with $t' = wire\_delay(m, n)$), so that when SIMULATE processes the instant t', $\mathbf{IsMarked}(n, t')$ returns a true value and function f_n is evaluated by $\mathbf{SimulateNode}(n, t')$. Hence, $\mathbf{SimulateNode}(n, t')$ computes the output value for node n at instant t', detecting if an event, i.e. a logic transition, has occurred, while $\mathbf{Event}(n, t')$ returns its type. When an event is detected for a node n at the instant t', SIMULATE calls **UpdateGateTable** to update the data associated to g_n within the table *Gate_Table*.

Gate_Table contains a complete trace of the switching activity of each signature class c within the digital circuit, for the overall simulation time interval $\mathcal{T}$. Let us call $tr_c(t)$ such a trace. Each signature class is also associated with a unique substrate injection waveform $w_c(t)$ stored in *Wave_Table*. The noise cumulative injection signal $i_c(t)$ for signature c is computed as the convolution

$$i_c(t) = \sum_{t'=t_0}^{t_{end}} tr_c(t - t') w_c(t') \ .$$

The global injection noise $i(t)$ is obtained by accumulating the noise signals associated with all S_C signature classes as

$$i(t) = \sum_{c=1}^{S_C} i_c(t)$$

As a final step, the FFT $i(t)$ yields the power spectrum density $I(f)$.

4 Results

The methodology proposed in this paper is supported by several tools which implement the various functions described in Figure 1. The tools are implemented in C/C++ running under the UNIX operating system. Figure 4 shows the flow adopted for the testing of the methodology. A Boolean network is mapped onto a given technology using the synthesis tool SIS [8] and compiled in standard cell style within the OCTTOOLS environment. Event-driven simulation is performed on the mapped network and the resulting signal is convolved and transformed in the frequency domain. As a verification step, the layout and underlying substrate is fully extracted

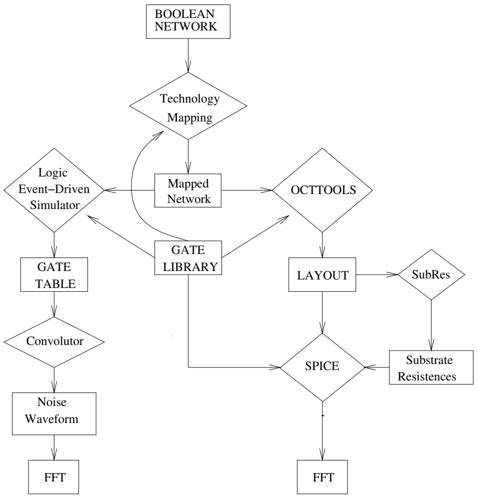

Figure 4: Testing SubWave methodology

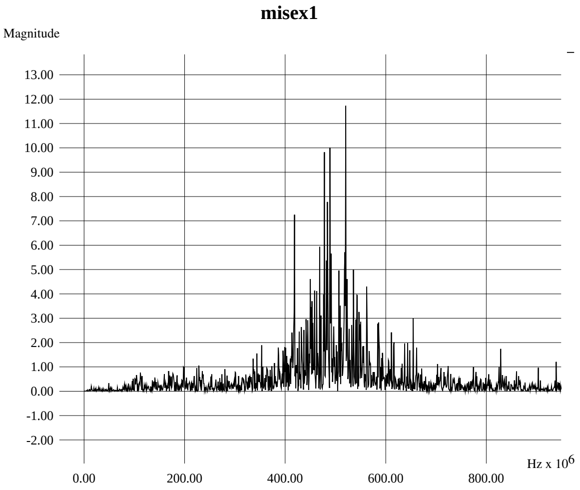

Figure 5: Substrate noise spectrum of **misex1**

with SubRes [1] and the injected currents are evaluated at a common sensing point using Spice.

Several circuits from the MCNC91 benchmark suite have been tested using one thousand randomly generated input vectors. In Table 1 the CPU times for the generation of the models and for the verification step are reported for a DEC AlphaServer 2100 5/250. The model for injected switching noise obtained through full simulation was in general two order of magnitudes slower than that obtained with SubWave. A direct comparison of the waveforms obtained with the two approaches shows a contained error estimated to be less than 10% for all the benchmarks. The largest circuits, such as the 16 bit multiplier, could not be simulated using Spice due to their extreme complexity. Figure 5 shows the spectrum generated for the `misex1` circuit. It results that the most of the spectrum is centered around the 500 MHz frequency. The same occurred for all the other benchmark mapped on the same library based on gates with a common 1 ns delay. This suggests that the substrate noise spectrum is mainly dependent on the technology rather than other

Mapped_Network				SubWave	SubRes	Spice
Circuit	function	I/O	Gates	*sec*	*sec*	
misex1	pla	8/7	70	75	257	395
z4ml	2b add	5/16	75	54	69	509
mux	mux	21/1	115	75	129	9139
my_adder	16b add	33/17	242	134	3383	-
alu2_cl	alu	10/6	506	430	10105	-
C1355	ecc	41/32	1215	202	-	-
C6288	16b mlt	32/32	2731	1426	-	-

Table 1: Results on MCNC91 Benchmarks

design parameters (clock, topology). Hence, noise models based on clock rate may account in general for less than 10-20% of the energy injected in the substrate.

5 Conclusions

A novel methodology has been proposed for the synthesis of compact injection noise models for large logic circuits. The injection signatures associated with given gates and all possible input transitions are evaluated and stored. All noise generating devices are cumulatively accounted for and the resulting switching noise is efficiently derived using a gate-level event-driven simulator. Finally, the FFT of the switching noise is performed. A set of standard benchmarks has been successfully used to test the accuracy of the models by comparing them against full extraction of the circuit and simulation. A strong dependency between the noise spectrum and the gate delay has been pointed out.

Acknowledgments The authors would like to thank Ranjit Gharpurey and Alex Saldanha for their support and useful discussions. A special thank goes to Alberto Ferrari for his invaluable technical support.

References

[1] R. Gharpurey and R. G. Meyer, "Modeling and Analysis of Substrate Coupling in ICs", in *Proc. IEEE CICC*, pp. 125–128, May 1995.

[2] B. R. Stanisic, N. K. Verghese, D. J. Allstot, R. A. Rutenbar and L. R. Carley, "Addressing Substrate Coupling in Mixed-Mode ICs: Simulation and Power Distribution Synthesis", *IEEE Journal of Solid State Circuits*, vol. SC-29, n. 3, pp. 226–237, March 1994.

[3] S. Mitra, R. A. Rutenbar, L. R. Carley and D. J. Allstot, "Substrate-Aware Mixed-Signal Macro-Cell Placement in WRIGHT", in *Proc. IEEE CICC*, pp. 529–532, May 1994.

[4] C. Hu, *VLSI Electronics: Microstructure Science*, volume 18, Academic Press, New York, 1981.

[5] M. R. Pinto, C. S. Rafferty, H. R. Yeager and R. W. Dutton, *PISCES-IIB*, Stanford University, Stanford, CA, 1985.

[6] R. K. Brayton, G. D. Hachtel and A. L. Sangiovanni-Vincentelli, "Multilevel Logic Synthesis", *Proceedings of the IEEE*, vol. 78, n. 2, pp. 264–300, February 1990.

[7] L. Lavagno and A. Sangiovanni-Vincentelli, *Algorithms for synthesis and testing of asynchronous circuits*, Kluwer Academic Publishers, 1993.

[8] E. M. Sentovich, K. J. Singh, C. Moon, H. Savoj, R. K. Brayton and A. L. Sangiovanni-Vincentelli, "Sequential Circuit Design Using Synthesis and Optimization", in *Proc. Intl. Conf. on Computer Design*, pp. 328–333, October 1992.