A METHODOLOGY FOR THE COMPUTATION OF AN UPPER BOUND ON NOISE CURRENT SPECTRUM OF CMOS SWITCHING ACTIVITY

by

Alessandra Nardi, Luca Daniel
and Alberto L. Sangiovanni-Vincentelli

Memorandum No. UCB/ERL M02/20

20 June 2002

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

A Methodology for the Computation of an Upper Bound on Noise Current Spectrum of CMOS Switching Activity

Alessandra Nardi, Luca Daniel, Alberto L. Sangiovanni-Vincentelli

Abstract

Currents injected by CMOS digital circuit blocks into the Gnd/Vdd system and into the substrate of a system-on-a-chip may affect reliability and performance of other sensitive circuit blocks. To verify the correct operation of the system, an *upper bound* for the spectrum of the noise current has to be provided with respect to all possible transitions of the circuit inputs. The number of input transitions is exponential in the number of circuit inputs. In this paper, we present a novel approach for the computation of the upper bound that avoids the untractable exhaustive exploration of the entire space. Its computational complexity is indeed linear in the number of gates. Our approach requires CMOS standard cell libraries to be characterized for injected noise current. In this paper, we also present an approach for this characterization of CMOS standard cells.

1 Introduction

The complexity of systems-on-a-chip design requires an aggressive re-use of IP (intellectual properties) circuit blocks. However IP blocks can be safely re-used only if they do not affect other sensitive components. The switching activity of CMOS digital circuit blocks typically injects high frequency current noise both into the Gnd/Vdd system, and into the substrate of integrated circuits. Such currents can potentially affect circuit reliability and performance of other sensitive components [9]. For instance the Gnd/Vdd currents may produce electromigration, IR voltage drops, voltage oscillations due to resonances, and, possibly, electromagnetic interference. The substrate currents may couple noise to sensitive analog circuitry through body effect or direct capacitive coupling. Current injection analysis is needed to properly account for all such effects during the design phase. Different effects require different types of current injection models. For instance, power consumption analysis requires time-domain average current estimation over several clock periods. IR drop, electromigration, and timing performance analysis require a time-domain noise current upper-bound with respect to all possible combinations of the inputs. Signal integrity, Gnd/Vdd grid resonances, electromagnetic interference, and substrate coupling on mixed-signal ICs require instead

an upper-bound on the spectrum of the current injected into the Gnd/Vdd system or into the substrate respectively over all the possible input transition vectors.

The methodology in [6] can be used to accurately estimate both time-domain and frequency domain injected noise for a given set of input vectors. However exhaustive circuit simulation for all the possible input transition vectors would be required for upper-bound estimation. The stochastic approach in [4] can estimate frequency domain *average* current injection, but not an upper bound on all possible input transition vectors.

The approaches in [3, 7, 1, 8] can estimate such an upper-bound in the *time domain*, but not in the frequency domain. In fact, these methodologies are suited to derive the maximum current envelope in the time domain, which in general does not correspond to an upper bound in the frequency domain. All these approaches divide the time domain into time intervals and search for an upper bound to the current in each interval, by identifying all the gates that could potentially switching in that interval. Therefore, the logic correlation inside the circuit is neglected or is at most considered only between each pair of gates. Devadas et al. [5] account for this logic correlation, but, since the original target was power consumption estimation, the approach relies on the assumption that the maximum (weighted) switching activity corresponds to the maximum current. The problem is translated into a weighted max-satisfiability problem and, therefore, it can be solved only for relatively small circuits. Furthermore, the use of this methodology is restricted to the time domain.

Signal integrity, Gnd/Vdd grid resonances, electromagnetic interference, and substrate coupling on mixed-signal ICs are not addressed by any of the existing current injection analysis algorithms. For these problems, we have developed a general methodology that estimates the '*Noise Current Spectrum Upper Bound*' (*NISUB*) of a digital block by combining the noise current injected by each gate and accounting for the circuit logic functionality. Our approach includes glitches and accounts for the logic correlation at the entire circuit level. Even though the entire logic space is explored by the algorithm, the complexity is simply linear in the number of gates. We describe a heuristic algorithm for a tight *NISUB* estimation and also show how to derive an exact although somewhat looser *NISUB*.

In our algorithm, we need a standard cell library characterization for the spectrum of the current noise. CMOS standard cell libraries nowadays are commonly characterized for timing performance analysis purposes, measuring and tabulating only output transition times and propagation delays. To the best of our knowledge no procedure for noise current analysis library characterization is yet available. In this paper we also present a methodology to characterize CMOS standard cell libraries for injected current noise. This part has highlighted some interesting issues about multiple input switching events that were mostly neglected in the past.

The paper is organized as follows: Section 2 gives a formal description of the problem we want to solve, Section 3 reports observations and results concerning the library characterization, and Section 4 describes the proposed solutions.

2 Problem Definition

2.1 Assumptions

The noise current spectrum calculated in this work is intended to be used as input for signal integrity or electromagnetic interference tools analyzing the power distribution grid of a large integrate circuit or multi-chip module (MCM). The same methodology can be used to calculate the noise current spectrum intended to be used as input for tools analyzing the effect of noise coupling through the substrate from a digital circuit block to some analog circuit blocks in a mixed signal integrated circuit.

Such types of analyses are typically performed using electromagnetic field solvers that consider interconnect (or substrate) layout geometries and can account for all sorts of capacitive, inductive, skin, proximity, diffusion, and even fullwave effects. Such field solvers typically identify on the powergrid (or substrate) some input ports and some output ports and calculate frequency domain noise transfer functions from unit current excitations located on the input ports to the output ports.

A key assumption in this approach is that a large integrated circuit (or MCM) can be subdivided into smaller circuit blocks, such that within each of these blocks all effects accounted for by the field solvers are negligible. Hence, we can assume that:

- for each small circuit block all noise injectors can be collected into one single injection port in the global powergrid of the entire integrated circuit (or MCM).
- within a circuit block the power supply voltage is uniformly constant and the current drawn by each gate does not affect significantly such voltage. In other words we assume each gate can be modeled as an independent current source.

Within this framework, the work presented in this paper is intended to provide an estimation for the maximum amplitude of the input excitation current of a single circuit block to be applied at its injection port.

2.2 Maximum Current Spectrum Envelope

The current spectrum due to the switching activity of a CMOS digital circuit block is typically discrete. Significant non-zero components are present at the clock frequency f_0 and at its first P harmonics: $f_k = k \cdot f_0,\ k = 0,\ldots,P$. In practical circuits P is typically not larger than 10 to 15 harmonics. The goal of this work is to find an upper bound of such noise current spectrum. One practical way to estimate such an upper bound is to consider separately each harmonic f_k in the spectrum, and to independently estimate an upper bound $I_{max}(f_k)$ for the current drawn by the circuit at that particular harmonic. The final result of this procedure is a *"Maximum Current Spectrum Envelope"*, obtained by collecting the individual bounds

$$\{I_{max}(f_0), I_{max}(f_1), \ldots, I_{max}(f_P)\}$$

at the $P+1$ harmonics in the spectrum.

2.3 Noise Current Model

Throughout the paper we will use the following definitions:

Definition 1 *We will indicate as $I_G(f_k)$ (or often simply as I_G), and refer to it as* 'the current of gate *G'*, *the noise current injected by the gate G alone at frequency f_k, assuming a constant supply V_{DD}.*

Definition 2 *We will indicate as $I_z(f_k)$, (or often simply as I_z), and refer to it as* 'the current of a node *z'*, *the sum of the noise currents injected at frequency f_k by all the gates in the transitive fanin network*[1] *of node z.*

Definition 3 *The* transition time of a node *z is the interval of time between the* 10% *and* 90% *points of the waveform* [2] *at node z. The* transition time of a gate *G is the transition time of its output node.*

Definition 4 *The* arrival time of a node *z is the instant of time corresponding to the* 50% *point of the node waveform, with respect to the beginning of a clock cycle.*

Definition 5 *The* propagation delay of a gate *G is the interval of time between the* 50% *values of the input and output waveforms.*

Let G be a gate with n inputs and a single output z. For sake of simplicity and without loss of generality, we can consider gates with only one output. The noise current spectrum value of G at a given frequency f_k is given by:

$$I_G = f(v, T_T, T_A, C_L); \quad I_G \in \mathbb{C}$$

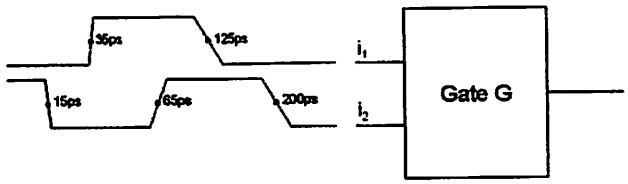
where:

- $v = \{v_1, v_2, \ldots, v_n\}$ is the input transition vector.
  $v_i \in B^{q_i}$ where $B = \{00, 01, 10, 11\}$ and q_i is the number of transitions on the i-th input.
  Therefore, $v \in V = B^{q_1} \times B^{q_2} \times \ldots \times B^{q_n}$

- $T_T = \{T_{T1}, T_{T2}, \ldots, T_{Tn}\}$ is the input transition time vector.
  $T_{Ti} \in S_i^{q_i}$ where $S_i = [T_{Tmi}, T_{TMi}]$ and q_i is the number of transitions on the i-th input.
  Therefore, $T_T \in S = S_1^{q_1} \times S_2^{q_2} \times \ldots \times S_n^{q_n}$.
  $[T_{Tmi}, T_{TMi}]$ represents the range of possible values for the transition time of input i. This range is specified in the standard cell library characterization.

- $T_A = \{T_{A1}, T_{A2}, \ldots, T_{An}\}$ is the input input arrival vector. $T_{Ai} \in [0, T_c)^{q_i}$ where q_i is the number of transitions on the i-th input, and T_c is the clock period. Finally, $T_A \in A = [0, T_c)^{q_1} \times [0, T_c)^{q_2} \times \ldots \times [0, T_c)^{q_n}$.

- $C_L \in \mathbb{R}^+$ is the output capacitive load.



$$
\begin{aligned}
v &= \{\ \{01\ ,10\ \},\{10\ ,01\ ,10\ \}\ \} = \{v_1\ ,v_2\ \} \\
T_T &= \{\ \{10ps\ ,50ps\ \},\{10ps\ ,30ps\ ,50ps\ \}\ \} = \{T_{T1}\ ,T_{T2}\} \\
T_A &= \{\ \{35ps\ ,125ps\},\{15ps\ ,65ps\ ,200ps\}\ \} = \{T_{A1}\ ,T_{A2}\}
\end{aligned}
$$

Figure 1: Example to illustrate the notation in the case $n = 2$; $q_1 = 2$ and $q_2 = 3$.

To understand this formulation more intuitively, we report an example in Figure1.

Similarly, the output transition time and the propagation delay of the gate are given by:

$$T_{TG} = g(v, T_T, T_A, C_L) \in [0, T_c)$$

$$T_{PG} = h(v, T_T, T_A, C_L) \in [0, T_c)$$

Given a certain circuit topology, C_L is fixed and, therefore, it can be eliminated from the search space.

If the same notation introduced above for a gate is now used for a circuit block with p primary inputs, then the *primary input transition vector space* $\mathcal{L}_p$ is described in the previous model by the case $q_i = 1, \forall i = 1, \ldots, n$.
Therefore, $\mathcal{L}_p = B^p$ and has cardinality $card(\mathcal{L}_p) = 2^{2p}$.

2.4 Problem Statement

For a circuit C with p primary inputs, input transition vector $v = \{v_1, v_2, \ldots, v_p\}$, input transition times T_T and arrival times T_A, for each frequency f_k in the spectrum we want to calculate:

$$I_{max}(f_k) = \max_{v \in \mathcal{L}_p, T_T \in S, T_A \in A} |I_C|$$

where I_C is the total noise current of the circuit block:

$$I_C = \sum_{i=1}^{N} I_{G_i},$$

[1]The transitive fanin network of z, is the cone at node z including z and *all* its predecessors. A cone at node z, denoted as C_z , is a subgraph consisting of z and some of its predecessors such that any path connecting a node in C_z and z lies entirely in C_z.

[2]Since our analysis is performed only on digital circuits, the input waveform space is restricted to linear ramps.

and N is the number of gates in the circuit block. We will further restrict the exploration space assuming our circuit block in exam is a combinatorial block standing between edge-triggered flip-flop's: therefore, primary inputs transition time and arrival times are assumed to be given. In particular, we will assume that all primary inputs will only switch at time $t = 0$. The problem is then reformulated as finding for each frequency f_k in the spectrum

$$I_{max}(f_k) = \max_{v \in \mathcal{L}_p} |I_C| .$$

3 Library Characterization

The algorithm for the upper bound of the noise current spectrum presented in this paper requires that each gate in the library be characterized both for timing and for noise injection analysis purposes. This means deriving the current spectrum, the output transition time and propagation delay of a gate for all possible input vectors. This section

- gives an overview of the library characterization issues,
- highlights some cases that require a special attention and that are typically not considered
- describes the criteria we derived to face these special cases.

Note that, referring to the formalism presented in the previous section, the characterization process for a gate G with n inputs assumes $q_i = 1, \forall i = 1, \ldots, n$.

Non-Switching-Output Events A Non-Switching-Output (NSO) event occurs when, in relation to some input transitions, the output of a gate does not switch. For example, $(a : 0 \rightarrow 0, b : 0 \rightarrow 1, z : 0 \rightarrow 0)$ and $(a : 0 \rightarrow 1, b : 0 \rightarrow 0, z : 0 \rightarrow 0)$ are NSO events for a 2-input AND gate with inputs a and b and output z. Noise current may be injected as a consequence of input transitions even if the output does not switch. Therefore, for a n-input gate, all the 2^{2n} possible input transitions should be modeled. For general transitions, the injected noise current spectral contents are a function of both the input transition time and the output capacitive load. However, the mechanism for noise current injection in NSO cases is different from Switching-Output (SO) cases. For example, the noise current injected during NSO events depends only on input transition time and not on the capacitive load.

Multiple-Input-Switching Transitions We define a Multiple-Input-Switching (MIS) transition, a transition where more than one gate input switches at the same time. For example, $(a : 0 \rightarrow 1, b : 0 \rightarrow 1)$, $(a : 0 \rightarrow 1, b : 1 \rightarrow 0)$, $(a : 1 \rightarrow 0, b : 0 \rightarrow 1)$, and $(a : 1 \rightarrow 0, b : 1 \rightarrow 0)$ are all the MIS transitions for a 2-input gate.

A first attempt at characterizing MIS transitions has been presented in [2]. However in that work only timing performance is considered.

Intuitively, if the current waveforms due to two consecutive input events do not overlap, then the MIS transition current is simply the superposition of the corresponding two SIS transitions currents. For sake of simplicity and without loss of generality, we will refer to a 2-input gate. Let $T_{Ai,k}$ and $T_{Az,k}$ be the mid-point of the input and output voltage waveforms respectively of a SIS transition k; and $T_{Ti,k}$ and $T_{Tz,k}$ the input and output transition times respectively for transition k. We call Δ_k the *Temporal Distance* among two consecutive input events in transition k:

1. if $(MIS,NSO)_k = (SIS,NSO)_{k1} + (SIS,NSO)_{k2}$, then:
   $\Delta_k = T_{Ai,k2} - T_{Ai,k1}$
2. if $(MIS,NSO)_k = (SIS,SO)_{k1} + (SIS,SO)_{k2}$, then:
   $\Delta_k = 1/2 \cdot [T_{Ai,k2} - 5/8 \cdot T_{Ti,k2} + T_{Az,k2} + 5/8 \cdot T_{Tz,k2}] +$
   $- 1/2 \cdot [T_{Ai,k1} - 5/8 \cdot T_{Ti,k1} + T_{Az,k1} + 5/8 \cdot T_{Tz,k1}]$
3. if $(MIS,SO)_k = (SIS,NSO)_{k1} + (SIS,SO)_{k2}$, then:
   $\Delta_k = 1/2 \cdot [T_{Ai,k2} - 5/8 \cdot T_{Ti,k2} + T_{Az,k2} + 5/8 \cdot T_{Tz,k2}] - T_{Ai,k1}$
4. if $(MIS,SO)_k = (SIS,SO)_{k1} + (SIS,NSO)_{k2}$, then:
   $\Delta_k = T_{Ai,k2} - 1/2 \cdot [T_{Ai,k1} - 5/8 \cdot T_{Ti,k1} + T_{Az,k1} + 5/8 \cdot T_{Tz,k1}]$

We define *Current Width* W_k of a transition k, the interval of time during which the current waveform related to transition k is not zero. We call *Disjunction Threshold* $\Delta_{TH,k}$, the value of the *Temporal Distance* Δ_k beyond which the MIS transition k becomes the superposition of its corresponding SIS transitions k_1 and k_2. $\Delta_{TH,k}$ is comparable to the semi-sum of the two SIS transition current widths: $\Delta_{TH,k} = 1/2 \cdot (W_{k1} + W_{k2})$. We finally propose to use an on-off type of model for the characterization of MIS transitions:

- If $\Delta_k \geq \Delta_{TH,k}$, we consider the MIS transition k simply as the superposition of the SIS transitions $k1$ and $k2$, each with its own input slope
- If $\Delta_k < \Delta_{TH,k}$, then we assume the inputs as simultaneous ($\Delta_k = 0$).

If this simple model is used, the library needs to be characterized only for $\Delta_k = 0$. An intuitive motivation for the previous model can be given observing that for NSO transitions, the current injection begins when the first input moves, and ends when the last input settles. For SO transitions, the current injection begins when the first input moves, and ends when the output settles. For both type of transitions the current waveform is approximately a peak centered around its mid-point.

The Base Table In summary, the model we propose for a gate G is a *Base Table*(BT_G), an example of which is shown in Table 1. Such table is derived for each gate G in the library for each frequency $f = k \cdot f_0$, as defined in Subsection 2.2.

As it can be seen from Table 1, BT_G of a gate G with n inputs, has 2^{2n} rows: each row corresponds to an input transition vector. For each transition j, the following data are calculated during the characterization:

- The current spectrum value of the gate $I_G^j = f_j(v, T_T, C_L)$

Table 1: The *Base Table* for a 2-input AND gate G.

$j: aba'b'$	zz'	T_{TG}	T_{PG}	I_G
0: 0000	00	0	0	0
1: 0001	00	T_{TG}^1	T_{PG}^1	I^1
2: 0010	00	T_{TG}^2	T_{PG}^2	I^2
3: 0011	01	T_{TG}^3	T_{PG}^3	I^3
4: 0100	00	T_{TG}^4	T_{PG}^4	I^4
5: 0101	00	0	0	0
6: 0110	00	T_{TG}^6	T_{PG}^6	I^6
7: 0111	01	T_{TG}^7	T_{PG}^7	I^7
8: 1000	00	T_{TG}^8	T_{PG}^8	I^8
9: 1001	00	T_{TG}^9	T_{PG}^9	I^9
10: 1010	00	0	0	0
11: 1011	01	T_{TG}^{11}	T_{PG}^{11}	I^{11}
12: 1100	10	T_{TG}^{12}	T_{PG}^{12}	I^{12}
13: 1101	10	T_{TG}^{13}	T_{PG}^{13}	I^{13}
14: 1110	10	T_{TG}^{14}	T_{PG}^{14}	I^{14}
15: 1111	11	0	0	0

- The propagation delay $T_{PG}^j = g_j(v, T_T, C_L)$
- The transition time $T_{TG}^j = h_j(v, T_T, C_L)$

where T_T and C_L are the gate input transition time and output capacitive load respectively.

The characterization assumes all gate inputs have the same arrival time: $T_{Ai} = 5/8 \cdot T_{Ti}$ for each input i.

Note that the *Base Table* is characteristic of a gate of the library: nevertheless, since I_G^j, T_{PG}^j and T_{TG}^j depend on the gate input transition time and output capacitive load, the *Base Table* has to be "instantiated" in the real circuit to obtain the value regarding the gate in the circuit.

An *Instantiated Base Table* of gate G (IBT_G), is obtained from BT_G by calculating I_G^j, T_{PG}^j, T_{TG}^j for each transition j by using the value of the gate input transition time imposed from the circuit environment. Notice that the current injected (at frequency

f_0) by a gate G whose input arrival time is T_s can be obtained from I_G in BT_G shifted by $\phi = -2\pi f_0(T_s - 5/8 \cdot T_{Ti})$ in the frequency domain.

Equivalence Classes According with the BT concept, a gate input vector transition space V is partitioned into four equivalence classes with an equivalence-relation defined as "generating the same output transition". An equivalence-class $E_G^b, b \in B$ for gate G is the set of all the rows of BT_G such that the output transition is equal to b.

The equivalence-classes defined by this relation for a two-input AND gate are:

$E_G^{00} = \{0000, 0001, 0010, 0100, 0101, 0110, 1000, 1001, 1010\}$
$E_G^{01} = \{0011, 0111, 1011\}$
$E_G^{10} = \{1100, 1101, 1110\}$
$E_G^{11} = \{1111\}$

Since there is a bijective relation among the row number and the input transition vector, we can equivalently write:

$E_G^{00} = \{0, 1, 2, 4, 5, 6, 8, 9, 10\}$
$E_G^{01} = \{3, 7, 11\}$
$E_G^{10} = \{12, 13, 14\}$
$E_G^{11} = \{15\}$

3.1 Experimental Results

In this Section we present results obtained by using our procedure to characterize the STMicroelectronics $0.18\mu m$ library optimized for high speed performance. As a test case we considered a small circuit containing 6 gates: two AND's, three OR's with different driving capabilities, and one EXOR. We analyzed only one primary input transition, but the input transition times and arrival times of the primary inputs are chosen in order to generate on the internal nodes all the particularly critical cases mentioned in the previous library characterization section. We compare in Fig. 2 two waveforms representing the current injected into Gnd/Vdd: the solid curve is obtained by circuit level simulation, while the dashed one is derived by using exclusively library characterization information. In particular, we used gate delays from the library in order to determine the switching instants of each gate. The corresponding current injection waveforms from the library are then positioned accordingly and added together to obtain the total current injection. The case shown in Fig. 2.a includes: 2 (MIS,SO) transitions with $\Delta < \Delta_{TH}$, 2 (MIS,SO) with $\Delta \geq \Delta_{TH}$, 4 (MIS,NSO) with $\Delta \geq \Delta_{TH}$. Glitches are also present. Fig. 2.b compares the spectrum of the two curves obtained by using a Fast Fourier Transform. As mentioned in the previous Section, simultaneous MIS transitions correspond to the case $\Delta < \Delta_{TH}$, and we propose to model such cases assuming for simplicity $\Delta = 0$. In Fig. 2.c and Fig. 2.d good results are obtained even when our algorithm uses $\Delta = 0$ to model one of the simultaneous MIS transitions with $\Delta \approx \Delta_{TH}$. The complete set of transitions in Fig. 2.c and Fig. 2.d includes: 2 (MIS,SO) transitions with $\Delta < \Delta_{TH}$, 1 (MIS,NSO) with $\Delta < \Delta_{TH}$, 1 (SIS,NSO), and 1 (SIS,SO).

As an additional remark, while performing the tests in this Section we observed that propagation delays of corresponding MIS and SIS transitions are different as claimed in [2]. We observed that such difference can be particularly critical when modeling noise injected currents. Hence when using propagation delays in a current injection

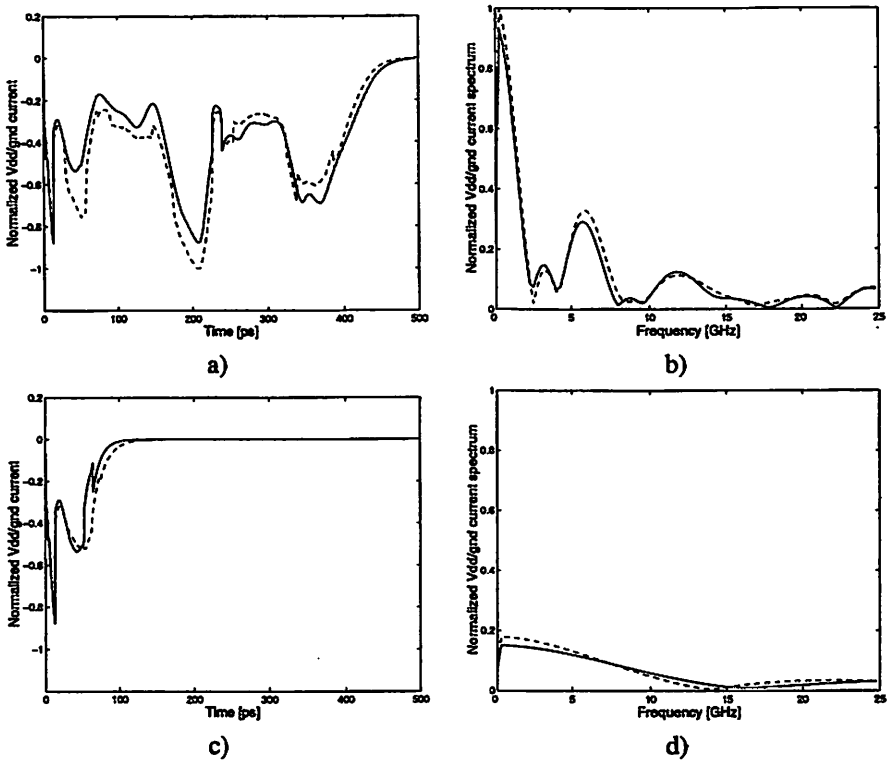


Figure 2: a) and c) compares current injection into Vdd/Gnd according to a circuit simulation and according a reconstruction from the characterized library. b) and d) are the spectrum of the two curves in a) and c) respectively.

estimation algorithm, a library characterized for timing in the classical may not be appropriate, but rather a timing model should be used which distinguishes between MIS and SIS transitions.

4 Upper Bound Estimation

In this section we describe our algorithm for the estimation of a Noise Current Spectrum Upper Bound (*NISUB*). First, to simplify our presentation we give a description of the algorithm neglecting glitches. Subsection 4.2 explains how to modify the algorithm and introduce glitches obtaing a heuristic estimation of the upper bound. In the last subsection, we derive an exact although somewhat looser upper bound.

Legenda
- PI = Primary Input of the circuit
- PO = Primary Output of the circuit
- CT_z^j = row j of the Composite Table at node z
- fanin_set(z) = set of all the fanin nodes of node z
- z.status: if set to VISITED, the CT_z has been already calculated

SEARCH_UB(PO)

SEARCH_UB(z) {
if ((z = PI) OR (z.status = VISITED))
  return;
else
  foreach FI ∈ fanin_set(z)
    SEARCH_UB(FI);
  end foreach
  CALC_CT(z);
  return;
end if

CALC_CT(z) {
foreach row j in CT_z
  foreach input k
    $E_{jk} \leftarrow$ EXTRACT_CLASS(j,k);
  end foreach
  $E_j \leftarrow E_{j1} \times \cdots \times E_{jn}$;
  $e^j_{MAX} = \underset{e \in E_j}{argmax}$ { CALC_ROW_CURRENT(e, j) };
  $CT_z^j \leftarrow$ CALC_ROW(e^j_{MAX}, j);
end foreach
}

EXTRACT_CLASS(j,k) {
extracts from CT^j the equivalence-class of the input k.}.

CALC_ROW_CURRENT(e, j) {
Calculates the current of a prospective row j for
the CT of node z from the rows specified by e.}

CALC_ROW(e, j) {
Calculates a prospective row j for the CT of node z
from the rows specified by e.}

Figure 3: Algorithm pseudo-code.

4.1 Computing *NISUB* without Glitches

The Composite Table Before describing the algorithm, we need to introduce the notion of *Composite Table* of a node z (CT_z). The *Composite Table* has the same structure of the *Instantiated Base Table*, i.e. each row j corresponds to an input transition vector. For each transition vector the following data are included in CT_z:

- The maximum current at frequency f_k of node z : I_z^j
- The arrival time: T_{Az}^j
- The transition time: T_{Tz}^j

Considering a gate G with output z, it is important to keep in mind that, although the IBT_G and the CT_z have basically the same structure, there is a crucial difference between them:
the values reported in IBT_G are related to the single gate G, while those in the CT_z are related to the entire transitive fanin network of node z.

The same equivalence-classes defined in Section 3 for IBT_G of a gate G can be used for the CT_z of a node z. We recall here that an equivalence-class E_G^b, $b \in B$ for node z is the set of all rows of CT_z such that the output transition is equal to b.

Observe that the case of no glitches corresponds to $q_k = 1$ for each input k of each gate, in the formal model given in Section 2.3.

The Algorithm The recursive algorithm for the estimation the *NISUB* of a combinatorial circuit has the following key properties:

- The recursion step processes one and only one gate
- Each gate is processed just once
- A gate is processed only when all its inputs have already been processed
- To process a gate means to calculate its *Composite Table*
- It is applied for each frequency $f = k \cdot f_0$, as defined in Subsection 2.2. The composition of all the resulting values gives a *Maximum Current Spectrum Envelope* (*MCSE*) for the circuit.

The key idea is that: *each row of CT_z is associated with a different input transition vector and includes for that input transition vector the upper bound on the current injected by the transitive fanin up to node z.* Hence, when the algorithm has finished processing all gates and termines, the upper bound of the entire circuit can be obtained by simply inspecting and picking from the rows of the composite table of the primary output the one with the largest current.

The pseudo-code for the algorithm is reported in Figure 3. Some comments and explanations on the algorithm:

1. The *Composite Table* of a gate is generated from the *Composite Tables* of its inputs and its *Instantiated Base Table*

2. For the primary inputs the composite table is given and represents the constraints we mentioned in Subsection 2.4. Formally, for uniformity of notation, each primary input may be considered as output of a dummy buffer.

3. Each recursion step builds the *Composite Table* of a node z. In particular, the *Composite Table* is built row by row. For each row, there is a local search for the maximum: among all the possible combinations of the cartesian product of the equivalence-classes of the inputs, only the one giving the maximum current is chosen. It is worth noticing that the reduction of complexity comes exactly from this step.

4. For the cases $v_i = 00$ or $v_i = 11$, we assign a symbolic value VOID to T_{Tz} and T_{Az}.

Example of application As an example, we show some steps of the algorithm applied to the simple circuit shown in Figure 4.
The frequency under analysis is $f_0 = 1GHz$.

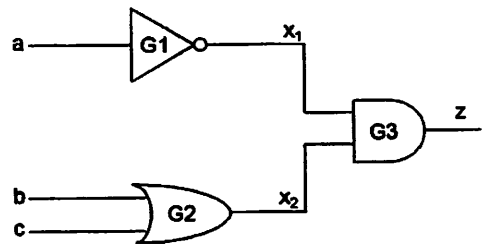


Figure 4: A simple circuit to illustrate the application of the algorithm.

1. Nodes are processed in this order: x_1, x_2 and z.

2. The *Composite Tables* of the primary inputs are given. We only report CT_a in Figure 5.a: the word dummy in the heading row recall that the primary inputs are assumed to be the output of a dummy buffer.

3. The first step is to derive CT_{x_1}.
   CT_a is given, IBT_{x_1} (shown in Figure 5.b) is calculated by using the T_{Ta} and T_{Aa} values specified in CT_a. The following formulas are applied for each row j:
   $T^j_{Ax_1} = T^j_{Aa} + T^j_{Px_1}(T^j_{Aa}, C_{Lx_1})$
   $T^j_{Tx_1} = T^j_{Tx_1}(T^j_{Aa}, C_{Lx_1})$
   $I^j_{x_1} = I^j_a + I^j_{G_1} \cdot e^{-iw_0\Delta}$ with $\Delta = T_{Aa} - 5/8 \cdot T_{Ta}$.
   In particular, in this case, $\Delta = 0$ for both the rows. Note that, since all equivalence-classes are composed by only one element, there is no actual search for the maximum here: the choice for each row of is CT_{x_1} unique.The solution is reported in Figure 5.c. It is also important to observe that, in case of more than one input,

the current of all the input nodes is summed to the current of the gate (properly shifted in the frequency domain) to give the current of the output node.

4. CT_{x_2} is calculated in the same way.

5. The last step is to calculate (row by row) CT_z from CT_{x_1}, CT_{x_2} and IBT_Z.
   We show the calculation only for row $j = 9$, also indicated as $j = 1001$.
   The EXTRACT_CLASS function returns:
   $E_{9x_1} = E^{10}_{x_1} = \{01\}$ (denoted as case x_{11} in Figure 6.a) and
   $E_{9x_2} = E^{01}_{x_2} = \{0001, 0010, 0011\}$ (denoted respectively as cases x_{21}, x_{22} and x_{23} in Figure 6.b).
   Therefore: $E_9 = E^{10}_{x_1} \times E^{01}_{x_2} = \{x_{11}x_{21}, x_{11}x_{22}, x_{11}x_{23}\}$.
   From direct inspection of the prospective CT^9_z reported in Figure 6.d:
   $I^j_{MAX} = 29.13\mu A$ and $e^j_{MAX} = \{x_{11}x_{23}\}$.
   This means that x_{11} and x_{23} are chosen as representative of $E^{10}_{x_1}$ and $E^{01}_{x_2}$ respectively to compute CT^9_z and the other cases are thrown away.

Just as a side observation, to propagate the transitions we are using the criteria based on the *DisjunctionThreshold* given in Section 3.

Reconvergent Fanout The algorithm sketched in Figure 3 can introduce a large error if the circuit presents reconvergent fanout. In fact, the current of a node z with fanout fo_z is counted fo_z times in the total circuit current.

It is actually correct to include these multiple counts because otherwise the choice of the maximum current on the following nodes would be falsed, but the final total current has to be adjusted properly to remove the multiple contributions.

To way to do so is to store the CT_z of a node z if $fo_z > 1$ (otherwise it is trashed after use). From these we can derive the values to be subtracted from the circuit current I_C to obtain the corrected value $I_{C,corr}$:

$$I_{C,corr} = I_C - \sum_{k=1}^{N} I_k \cdot (fo_k - 1)$$

where N is the number of gates in the circuit.

Algorithm Complexity Since each gate in the circuit is processed just once, the complexity of graph traversal is $O(N)$, where N is the number of gates in the circuit. The cost of processing each node, i.e. building its *Composite Table*, is basically $C_{node} = c \cdot C_p$ where:

- C_p is the cost of the CALC_ROW function
- c represents the number of times the CALC_ROW function is applied to process a node

Therefore, the cost of our algorithm is: $C_{imp} = N \cdot C_{node} = N \cdot C_p \cdot c$. The value of c is bounded by $2^{2n^2_{max}}$, where n_{max} is the maximum number of inputs of a gate (a

$dummy$	aa'	$T_{Ta}[ps]$	$T_{Aa}[ps]$	$I_a[\mu A]$
00	00	VOID	VOID	0
01	01	60	37.5	0
10	10	40	25	0
11	11	VOID	VOID	0

a)

aa'	x_1x_1'	$T_{TG_1}[ps]$	$T_{PG_1}[ps]$	$I_{G_1}[\mu A]$
00	00	VOID	VOID	0
01	01	100	70	$12e^{j2\pi 0.1}$
10	10	70	80	$15e^{j2\pi 0.8}$
11	11	VOID	VOID	0

b)

aa'	x_1x_1'	$T_{Tx_1}[ps]$	$T_{Ax_1}[ps]$	$I_{x_1}[\mu A]$
00	00	VOID	VOID	0
01	01	100	107.5	$12e^{j2\pi 0.1}$
10	10	70	105	$15e^{j2\pi 0.8}$
11	11	VOID	VOID	0

c)

Figure 5: a) *Composite Table* for node a. b) *Instantiated Base Table* for gate G_1. c) *Composite Table* for node x_1.

commonly used value is $n_{max} = 5$). As explained in Appendix A, this worst case is extremely unrealistic and a conservative average case can be estimated to be $c = 2^{10}$. Please, refer to Appendix A for more comments on this topic.

For a circuit with p primary inputs, an explicit exhaustive search on $\mathcal{L}_p$ (using for example the methodology in [6]) would have cost $C_{exp} = N \cdot C_p \cdot 2^{2p}$.

Independently from the assumptions dictated by common sense, the comparison shows that $C_{imp} < C_{exp}$ for any circuit with $p > 25$. The comparison becomes even sharper when using a more realistic value for c.

It is worth noticing that it is the use of the equivalence-classes of the *Composite Table* that allows to explore the space $\mathcal{L}_p$ *implicitely* and, therefore, without processing all the 2^{2p} input transitions.

Case	aa'	x_1x_1'	$T_{Tx_1}[ps]$	$T_{Ax_1}[ps]$	$I_{x_1}[\mu A]$
x_{11}	01	10	70	105	$15\mu Ae^{j2\pi 0.8}$

a)

Case	$bcb'c'$	x_2x_2'	$T_{Tx_2}[ps]$	$T_{Px_2}[ps]$	$I_{x_2}[\mu A]$
x_{21}	0001	01	60	100	$20\mu Ae^{j2\pi 0.3}$
x_{22}	0010	01	70	110	$27\mu Ae^{j2\pi 0.1}$
x_{23}	0011	01	55	98	$23\mu Ae^{j2\pi}$

b)

Case	$x_1x_2x_1'x_2'$	zz'	$T_{Tz}^9[ps]$	$T_{Az}^9[ps]$	$I_G^9[\mu A]$
$x_{11}x_{21}$	1001	00	80	150	$12e^{j2\pi 0.2}$
$x_{11}x_{22}$	1001	00	70	130	$16e^{j2\pi 0.6}$
$x_{11}x_{23}$	1001	00	60	120	$14e^{j2\pi 0.3}$

c)

Case	$x_1x_2x_1'x_2'$	zz'	$T_{Tz}^9[ps]$	$T_{Az}^9[ps]$	$I_z^9[\mu A]$
$x_{11}x_{21}$	1001	00	80	255	$20.94e^{j2\pi 0.166}$
$x_{11}x_{22}$	1001	00	70	235	$10.95e^{-j2\pi 0.0.24}$
$x_{11}x_{23}$	1001	00	60	225	$29.13e^{j0}$

d)

Figure 6: a) Rows of CT_{x_1} belonging to the Equivalence Class $E_{x_1}^{10}$ of node x_1. b) Rows of CT_{x_2} belonging to the Equivalence Class $E_{x_2}^{01}$ of node x_2. c) Row $j = 9$ of the *Instantiated Base Table* of node z for the three cases and d) all the corresponding prospective row $j = 9$ of CT_z.

Remarks The logic space exploration is complete according to the definition of equivalence-class we have given, but the choice of the representative element of a class is performed by using as cost function only the maximum current for the node under analysis. This choice does not necessarily imply maximum current for the following nodes, because the current of the gate in the next step also depends on the combination with the arrival times of its other inputs. Nevertheless, the effect of this error should not be significant given that our approach explores the entire primary input transition vector space. Most of the previous approaches [3, 7, 1, 8] use a much simpler model for the gate current and rely on stronger assumptions: e.g. the current injection of a gate is

considered only if the output node switches, the dependency on input transition time of the current, the propagation delay and the output transition time is neglected, etc. Furthermore, the logic correlation inside the circuit is neglected or is at most considered only between each pair of gates.

4.2 Computing *NISUB* with Glitches

The generic formulation in Section 2.3 accounts for glitches using the variables q_k for each gate input k. Exploiting such formulation we can easily extend the approach with no glitches presented in Subsection 4.1 to include glitches by simply re-defining the equivalence-relation that partition the input transition vector space. Specifically, a gate input space V can be partitioned into four equivalence classes by the equivalence-relation defined as follows: *'Two input transition vectors are in the same equivalence class if their correspondent output transition has the same initial and final values.'* The same algorithm in 4.1 can now be used to operate on the newly defined classes. Only minor details in the CALC_ROW function are needed to handle the new classes. Notice also that, having assumed that the p primary inputs can switch only once at the beginning of the clock cycle, the cardinality of the input transition vector space is still 2^{2p}.

4.3 Exact Upper Bound

The algorithm presented above does not compute an exact Upper Bound but rather an *estimation*. The computed bound is indeed a true upper bound with respect to the logic input space. However, some, albeit small, approximations are introduced when capturing the dependency on transition and arrival times of the current injected by each gate. This dependency has to be overestimated for the upper bound to become exact.

- The dependency on the input transition time can be eliminated at the library characterization phase by measuring and tabulating the maximum current injected as a function of all possible input transition times.
- At each node the algorithm in the Section above selects the representatives of each equivalence class that are associated with the largest transitive fanin current. Such choice was then used to identify also the output transition vector for each class. One way to overestimate the dependency of the current arrival times is to choose at each node and for each equivalence class the output transition vector that produces the most transitions, even if it is not associated with the largest transitive fanin current up to that node. When calculating the largest transitive fanin current, or the output transition vector with most transitions, we freely move the gate input transitions in order to maximize each cost function independently. Unfortunately in this way, the information about the phases is lost and we can only sum up the modules of the currents. This can lead to a quite loose upper bound (large overestimation) especially at high frequencies.

5 Conclusions

A methodology has been presented to characterize the noise current spectrum injected by CMOS switching gates into the Gnd/Vdd system or into the substrate of integrated circuits. Specifically we have described a procedure to estimate an upper bound for such noise current spectrum with respect to all possible transition vectors at the circuit primary inputs. Our algorithm has linear complexity in the number of gates and has been shown to provide significant computational advantage with respect to an exhaustive exploration of the input space. In this paper, a procedure has also been presented for CMOS standard cell libraries characterization of the switching current injection, which we use in the upper bound estimation algorithm. Our model captures special important cases such as Non-Switching-Output, and Multiple-Input-Switching events which are typically neglected in classical library characterization procedures. Example results have been given for a simple 6-gate circuit which show very good agreement with full circuit level simulation.

Acknowledgments

The authors would like to thank Claudio Pinello for useful discussions and suggestions on the formalization of the problem.

References

[1] S. Bobba and I. N. Hajj. Estimation of maximum current envelope for power bus analysis and design. *Proceedings of International Symposium on Physical Design*, pages 141–146, April 1998.

[2] V. Chandramouli and K. A. Sakallah. Modeling the effects of temporal proximity of input transitions on gate propagation delay and transition time. In *Design Automation Conference*, pages 617–22, June 1996.

[3] S. Chowdhury and J. S. Barkatullah. Estimation of maximum currents in mos ic logic circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 9(6):642–654, June 1990.

[4] A. Demir and P.Feldmann. Modeling and simulation of the interference due to digital switching in mixed-signal ics. In *Proceedings of International Conference on Computer-Aided Design*, pages 70–74, November 1999.

[5] S. Devadas, K. Keutzer, and J. White. Estimation of power dissipation in cmos combinational circuits using boolean function manipulation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 11(3):373–383, March 1992.

[6] C. E., R. Garpurey, P. Miliozzi, R. G. Mayer, and A. Sangiovanni-Vincentelli. *Substrate noise : analysis and optimization for IC design.* Kluwer Academic Publishers, 2001.

[7] H. Kriplani, F. N. Najm, and I. N. Hajj. Pattern independent maximum current estimation in power and ground buses of cmos vlsi circuits: Algorithms, signal correlations, and their resolution. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 14(8):998–1012, August 1995.

[8] A. Krstic and K. Cheng. Vector generation for maximum instantaneous current through supply lines for cmos circuits. In *Proceedings of Design Automation Conference*, pages 383–388, June 1997.

[9] S. Lin and N. Chang. Challenges in power-ground integrity. In *IEEE/ACM Internat. Conf. on Computer-Aided Design*, Nov. 2001.

A Algorithm Complexity

To derive the cost of building a *Composite Table*, we refer to the generic situation depicted in Figure 7, where each box is a gate and the number inside the box represents the number of inputs. Following the definition given in Section 3, each gate G_{1i} has four equivalence-classes $C_{1i}^{00}, C_{1i}^{01}, C_{1i}^{10}, C_{1i}^{11}$. C_{1i}^{u} indicates a generic equivalence-class of gate G_{1i} and its cardinality is denoted by $card(C_{1i}^{u})$.
Indicated as n_{max} the maximum number of inpus for the gates of the library, the worst case is given by:
- $n_2 = n_{max}$
- $n_{1i} = n_{max}, \forall i = 1,\ldots,n_2$
- $card(C_{1i}^{u}) = 2^{2n_{1i}}, \forall i = 1,\ldots,n_2$ (which means that all the inputs of gate G_2 have a single equivalence-class)

and the cost is:

$$c = \prod_{i=1}^{n_2} 2^{2n_{1i}} = 2^{2n_{max}^2}$$

Since $n_{max} = 5$, this value is obviously unacceptable, but if we consider more realistic cases this cost drops dramatically. In the following we consider some cases to give a more reasonable estimation of the cost c.

In particular, we first consider the cases in which, for each input gate G_{1i}, all the four equivalence-classes have the same cardinality, i.e. G_{1i}, $card(C_{1i}^{u}) = 2^{n_{1i}/2}, \forall i = 1,\ldots,n_2$:

1. $n_{11} = n_{12} = 2, n_2 = 2 \Rightarrow c = 2^6$
2. $n_{11} = n_{12} = 3, n_2 = 2 \Rightarrow c = 2^7$
3. $n_{11} = n_{12} = n_{13} = 2, n_2 = 3 \Rightarrow c = 2^9$
4. $n_{11} = n_{12} = n_{13} = 3, n_2 = 3 \Rightarrow c = 2^{10.5}$
5. $n_{11} = n_{12} = n_{13} = 4, n_2 = 3 \Rightarrow c = 2^{12}$

Just as a reference, we also report some cases in which all the inputs of gate G_2 have a single equivalence-class:

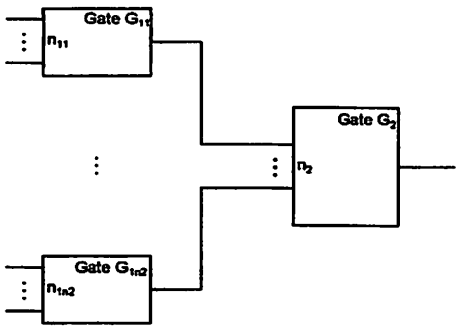


Figure 7: Generic situation for a recursion step.

1. $n_{11} = n_{12} = 2, n_2 = 2 \Rightarrow c = 2^8$
2. $n_{11} = n_{12} = 3, n_2 = 2 \Rightarrow c = 2^{12}$

Since a good CMOS design rule is to minimize the transistor stack size, the worst case illustrated at the beginning is extremely unlikely. The other cases we reported are much more realistic and, therefore, we believe that $c = 2^{10}$ is a reasonable value for the cost.