

Analysis of the Impact of Bus Implemented EDCs on On-Chip SSN

Daniele Rossi

Carlo Steiner

Cecilia Metra

DEIS, University of Bologna, Viale Risorgimento 2, 40136, Bologna, Italy
{drossi, cmetra}@deis.unibo.it

Abstract

In this paper we analyze the impact of error detecting codes, implemented on an on-chip bus, on the on-chip simultaneous switching noise (SSN). First, we analyze in detail how SSN is impacted by different bus transitions, pointing out its dependency on the number and placement of switching wires. Afterwards, we present an analytical model that we have developed in order to estimate the SSN, and that we prove to be very accurate in SSN prediction. Finally, by employing the developed model, we estimate the SSN due to different EDCs implemented on an on-chip bus. In particular, we highlight how their differences in the number of switching wires, bus parallelism and codewords influence the on-chip SSN.

1 Introduction

Very deep submicron technology (VDSM) poses many challenges to the design and test community, mainly due to the increased integration density, reduced node capacitances, power supply and noise margins. Future ICs will consequently be more prone to both permanent and transient faults [1].

On-chip interconnects in VDSM technology are becoming more sensitive to errors caused by power supply noise (frequently referred to as simultaneously switching noise, *SSN*), crosstalks, delay variations and transient faults [4, 5]. Especially if the bus is communicating with flip-flops, incorrectly sampled data may propagate throughout the whole system.

To guarantee signal integrity of the on-chip communication, a fault tolerant bus can be adopted. This could be achieved by implementing techniques based on on-line testing, followed by proper fault recovery. For instance, an error due to noise affecting the bus wires can be concurrently revealed, for instance by using proper error detecting codes (EDCs) and a bus checker [7]. Then, in order to restore the system correct operation, a proper recovery phase has to be activated.

If retransmission can not be accomplished by the system,

error correcting codes could be implemented [13, 11]. This kind of codes, in fact, allow on-line correction. However, if the error probability is not very high, techniques based on error detection followed by retransmission have been proven to be more cost effective, in terms of both power and performance [8].

On the other hand, we can expect that the application of error detecting codes aggravates *SSN*, because of the increased number of switching signals. This may in turn drawback systems' reliability, and cause more electromagnetic interference. In fact, *SSN* has five primary undesirable effects on system reliability [6, 3]: (i) it increases the propagation delay through the switching drivers and the neighboring circuitry sharing the same V_{dd} and ground distribution networks, with possible consequent violations of circuit timing constraints; (ii) it induces resonance in the power distribution network, which further alters the values of V_{dd} and ground; (iii) it causes false switching of a circuit, affecting the logical one or logical zero voltage thresholds; (iv) it causes false switching of a circuit whose input nets are capacitively coupled to V_{dd} or ground rails; (v) it alters the logical value stored in a high impedance node of a dynamic circuit, making conducting a transistor that was expected to be temporarily off.

In this paper we analyze the impact on on-chip *SSN* of several EDCs implemented on an on-chip bus. First we evaluate the *SSN* induced by different kinds of transitions, pointing out the effect of both the number of switching wires and the mutual switching activity of adjacent wires.

Successively, we present an analytical model employed to estimate the *SSN* generated by the switching drivers of an on-chip bus. This model allows to predict accurately the value of the worst case *SSN*, given the bus parallelism and the number of switching wires.

Finally, by means of the developed analytical model, we compare the different EDCs from the point of view of the worst case *SSN*, by means of the developed analytical model. We highlight how their differences in the maximum number of simultaneous switching wires, bus parallelism and codewords influence the on-chip *SSN*.

This paper is organized as follows. In Sect. 2, we give some preliminaries on *SSN*. In Sect. 3, we present the elec-

trical model employed to perform our analysis. In Sect. 4, we discuss some theoretical aspects of the on-chip *SSN* which have led us to the definition of an analytical model able to estimate accurately the *SSN*, as presented in Sect. 5. In Sect. 6, we evaluate the impact of different EDCs on *SSN*. Some conclusive remarks are given in Sect. 7.

2 Where *SSN* Comes From

Let us consider a generic chip and let us indicate by: a) V_{dd}^{ext} and V_{ss}^{ext} , the external power supply and ground, respectively (i.e., the power supply and ground of a printed-circuit board); b) L and R , the parasitic inductance and resistance of the power supply and ground networks; c) V_{dd}^{int} and V_{ss}^{int} , the on-chip power supply and ground, respectively. Considering the case of the power supply, we can easily derive that, if $I(t)$ is the total current drawn by the chip, the value of the on-chip V_{dd} is given by:

$$V_{dd}^{int}(t) = V_{dd}^{ext} - L \frac{dI(t)}{dt} - RI(t) \quad (1)$$

The power supply noise is therefore composed by two factors: an inductive noise (which plays a dominant role) and a resistive noise [14].

The circuit devices which draw the larger part of current are the strong drivers of the bus wires which, in synchronous systems, switch simultaneously. Consequently, approximating the total current I with the sum of the current I_i that is drawn by each switching driver, and assuming for simplicity that the currents drawn by the switching drivers are all equal, when n drivers switch simultaneously, we can write:

$$V_{dd}^{int}(t) = V_{dd}^{ext} - n(L \frac{dI_i(t)}{dt} + RI_i(t)). \quad (2)$$

Similarly, we can derive:

$$V_{ss}^{int}(t) = V_{ss}^{ext} + n(L \frac{dI_i(t)}{dt} + RI_i(t)). \quad (3)$$

By subtracting Eq. 3 from Eq. 2, we obtain:

$$V_{dd}^{int}(t) - V_{ss}^{int}(t) = V_{dd}^{ext} - V_{ss}^{ext} - 2n(L \frac{dI_i(t)}{dt} + RI_i(t)). \quad (4)$$

The first member of the previous equation represents the on-chip effective power supply, which differs from the nominal value of power supply ($V_{dd}^{ext} - V_{ss}^{ext}$) by a term representing the total power supply noise. Therefore we can write:

$$SSN = 2n(L \frac{dI_i(t)}{dt} + RI_i(t)). \quad (5)$$

From the previous equation we can notice that, in a first order approximation, the simultaneous switching noise (*SSN*) is linearly dependent on the number of drivers that switch simultaneously [13, 11, 2]. We will show that, going deeper into the *SSN* analysis, this conclusion turns out to be inaccurate.

3 Electrical Model

To evaluate how different bus transitions can affect the on-chip power supply and ground voltages, we have considered the equivalent circuit schematically shown in Fig. 1. It

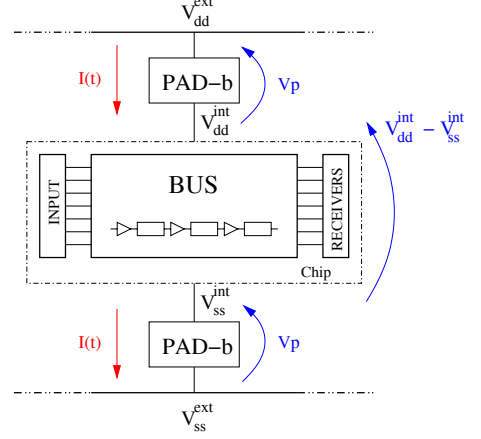


Figure 1. Schematic representation of the circuit employed for *SSN* analysis.

consists of a synchronous on-chip bus, whose internal supply and ground signals are connected to the external (nominal) values by two RLC circuits (Pad-b), as reported in [9]. They account for the parasitics due to both the bond wires and the package pins [9].

The bus lines, implemented by a standard $0.25\mu m$ CMOS technology, are buffered, and are $10mm$ long, with a buffer every $2mm$. Each $2mm$ bus segment has been modeled as shown in Fig. 2.

In particular, each $2mm$ line segment has been modeled by a π RC circuit (Fig. 2(a)), with the wire capacitances split-up into two: a half accounting for the coupling between the line and the on chip ground plane (V_{ss}^{int}), and a half accounting for the coupling between the line and the on chip power supply plane (V_{dd}^{int}). This allows to better model the current return paths during the bus switching [2]. Furthermore, the block referred to as *Crosstalk i* accounts for the crosstalk coupling capacitances between adjacent lines. It has been modeled as shown in Fig. 2(b).

4 *SSN* Analysis

As known, and recalled above, any bus switching may cause *SSN*. -In a previous work ([13]), it has been verified that, given the technology and number of bus wires, *SSN* increases rather linearly with the number of switching wires. Furthermore, it was shown that the most noisy transitions for a bus are those from *all 0s* to *all 1s*, or vice versa. However, that work did not investigate the effect on *SSN* of

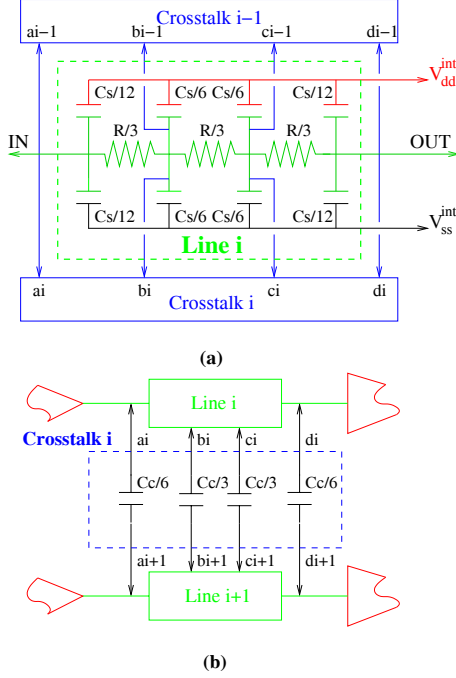


Figure 2. π equivalent RC circuit of a wire segment (a) and the relative crosstalk coupling capacitance (b).

the placement of the switching wires (when the number of switching wires is lower than the bus parallelism, hereafter referred to as $\#Bus$).

In this section, by performing an accurate electrical analysis, we highlight that transitions with the same value of switching wires (hereafter referred to as ΔBit_k), but with different placement cause a different value of SSN . As an example, let us consider the two transitions $00000000 \rightarrow 10101010$ and $10001010 \rightarrow 11111110$: both of them have $\Delta Bit = 4$, and all wires switch from 0 to 1, but the first induces a noise $SSN_1 = 368mV$, while for the second we have $SSN_2 = 242mV$, with a difference greater than the 5% of the power supply.

Furthermore, here we point out that the SSN does not depend linearly on the number of switching wires for all values of $\Delta Bit \in [1, \#Bus]$. Instead, as clarified later on, it presents a piece-wise linear dependency. To explain this "anomalous" behavior, in this section we analyze in detail how the switching of any single line contributes to the SSN . In particular, we show how this contribution depends critically on the behavior of the adjacent wires. To estimate the SSN , we suppose that the transitions are all simultaneous (worst case analysis).

First of all, let us clarify the definition of SSN adopted throughout the paper. Since V_{dd}^{int} and V_{ss}^{int} present the typical damped oscillations of an RLC circuit, we define the

noise affecting the on-chip power supply as:

$$SSN_{V_{dd}} = \max(|V_{dd}^{ext} - V_{dd}^{int}(t)|) \quad (6)$$

Similarly, the noise affecting the on-chip ground is:

$$SSN_{V_{ss}} = \max(|V_{ss}^{ext} - V_{ss}^{int}(t)|) \quad (7)$$

Therefore, the total amount of the noise can be written as: $SSN = SSN_{V_{dd}} + SSN_{V_{ss}}$.

Increasing the number of switching wires, the SSN increases as well, as shown in the previous section. Let us represent by $\Delta Bit_k = W[C(k) - C(k+1)]$ the number of switching wires involved in the transition $C(k) \rightarrow C(k+1)$. It is given by the weight $W(\cdot)$ of the pattern obtained by subtracting mod 2 bit by bit the final pattern from the starting one, and it coincides with the Hamming distance between $C(k)$ and $C(k+1)$. In Fig. 3 we represent the worst case value of SSN , for different values of bus parallelism ($\#Bus$), as a function of the number of switching wires.

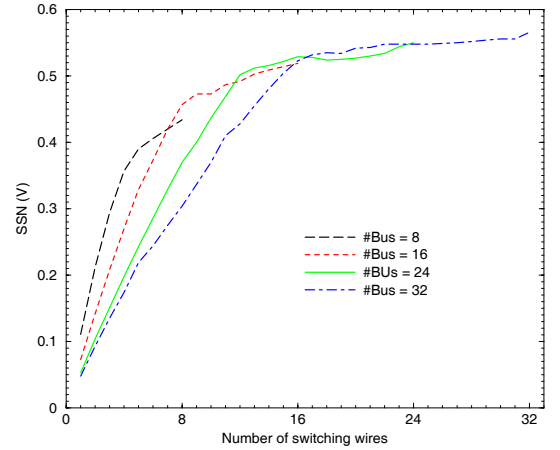


Figure 3. SSN for buses with 8, 16, 24 and 32 wires as a function of ΔBit .

Differently from what could be derived from equations in Sect. 2, the behavior of the on-chip power supply, and hence that of SSN , is not linear with the number of switching wires. In particular, it presents a piece-wise linear behavior, with two main segments with different slopes.

The graphs in Fig. 3 allow a qualitative analysis of the SSN . For all 4 cases, we have an analogous variation of SSN as a function of ΔBit , for the worst case placement. The SSN increases linearly for $1 \leq \Delta Bit \leq \frac{\#Bus}{2}$, then the curves change their slopes and tend to saturate.

Moreover, we can observe that the SSN decreases as $\#Bus$ increases, for a given value of ΔBit . In fact, as reported also in [11, 9], the parasitics (wire's resistances, capacitances and inductances) concur to the definition of the amount of SSN generated by the switching of the bus. In

particular, wires' parasitic capacitances can act as decoupling capacitors during a transition, thus reducing the maximum value of the SSN [11].

In order to explain this behavior, let us analyze the contribution to SSN of a single line switching. To accomplish this task, we need a deeper insight in the relation between the placement of the switching wires and the SSN . As an example, let us consider a wire switching from 0 to 1. Analyzing the contribution to SSN of this wire as a function of the behavior of the two adjacent wires, we can distinguish 4 cases: i) adjacent wires switching from 0 to 1; ii) adjacent wires switching from 1 to 0; iii) adjacent lines stable at 0; iv) adjacent lines stable at 1.

In the first case, the central wire and its two neighbors switch in the same direction. Since the voltage drop across the coupling capacitances is always zero, they do not need to be charged, and do not contribute to the current drawn from the power supply.

In the second case, the central and the two adjacent wires switch oppositely and, consequently, due to the *Miller Effect*, the crosstalk coupling capacitances assume an effective value which is the double of the physical one. We have verified that this case, although requiring a larger current drawn from the power supply than the previous case (since the bus drivers have to charge a larger capacitance), induces less SSN . This because, as reported in [13], the transition is slower compared to that of case i), and consequently the inductive noise is less impacting.

If the central line has two neighboring wires at a constant value (cases iii. and iv.), the transition of the central wire will certainly cause the charge or discharge of the crosstalk coupling capacitances. If the two stable wires are at "0", the crosstalk coupling capacitance C_c will be charged (Fig. 4(a)), while it will be discharged if the stable wires are at "1" (Fig. 4(b)). In these figures, we have represented only the crosstalk coupling capacitance between the switching wire and one neighbor. Analogous considerations could have been made if we had considered both the neighboring wires.

By means of electrical simulations, we have verified that the two transitions have (as expected) approximately the same speed. However, they differ when considering the charge redistribution phenomena. In fact, in the case shown in Fig. 4(a), the crosstalk coupling capacitances are initially charged. During the transition, they contribute to provide the charge required by the bottom capacitances by means of the charge redistribution effect. This charge flow is completely on-chip, allowing to reduce the current flown through the external pad P . As a consequence, the SSN is reduced compared to the case represented in Fig. 4(b). In this last case, in fact, the crosstalk coupling capacitances are initially discharged, thus they can not contribute to charge the bottom capacitance of the switching wire by means of the charge redistribution effect. Moreover, the power sup-

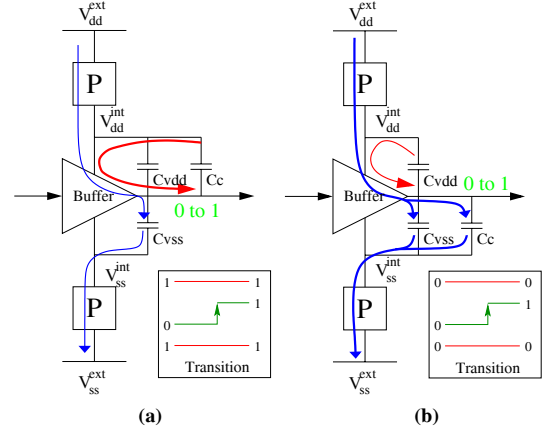


Figure 4. Equivalent circuit schematically representing a $0 \rightarrow 1$ wire transition, with the adjacent wires fixed to 1 (a) and 0 (b).

ply has to provide the current to charge also the capacitances C_c . Therefore, the current flow through the external pad P is considerably higher than in the previous case. Electrical simulations have shown that a switching wire generates the largest amount of SSN when both its neighboring wires are stable at their initial value. We will refer to this case as *worst case placement*.

Now, let us go back to Fig. 3. As long as $\Delta Bit \leq \#Bus/2$, it is always possible to have switching wires with the two neighboring wires stable at the starting constant value. Consequently, from the analysis performed above, it derives that the worst case SSN increases rather linearly for $1 \leq \Delta Bit \leq \#Bus/2$. When $\Delta Bit > \#Bus/2$, instead, there is no possibility to have all switching wires with stable neighboring wires. Increasing the number of switching wires, the number of switching adjacent wires increases as well. As shown above, this case does not represent the worst case placement regarding the contribution to the SSN of a single wire switching. Therefore, the slope of the curves in Fig. 3 decreases for $\Delta Bit > \#Bus/2$.

5 Developed Analytical Model

In order to evaluate the SSN , we have developed a semi-empirical analytical model which allows to estimate the SSN for a generic transition on a bus with a given $\#Bus$. We express the total SSN as the superposition of the noise produced by each line involved in the bus activity. The contribution of each line is influenced directly by the two adjacent lines (terms r in Fig. 5) and indirectly by all the others (Fig. 5). The effect of the distant lines is attenuated by the interposed lines (terms a in Fig. 5). Therefore, for the line i we have:

$$R_i = \sum_{j=0}^{i-2} r_j^i \prod_{k=j+1}^{i-1} a_k^i + r_{i-1}^i + r_{i+1}^i + \sum_{j=i+2}^{\#Bus-1} r_j^i \prod_{k=j+1}^{i+1} a_k^i \quad (8)$$

Consequently, the SSN can be estimated as: $SSN \approx \sum_{i=0}^{\#Bus-1} R_i$, where r_y^x and a_y^x are fitting parameters which depend on the technology and $\#Bus$.

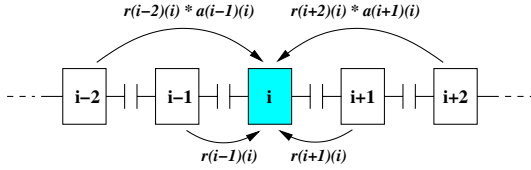


Figure 5. Schematic representation of the effect of all bus lines in determining the SSN due to the transition of the wire i .

In Fig. 6 we plot the simulated (by means of HSPICE) and estimated (by means of our model) worst case SSN , as a function of the number of switching wires, considering a 16 wires bus.

0 We can see that the model estimates with a very high

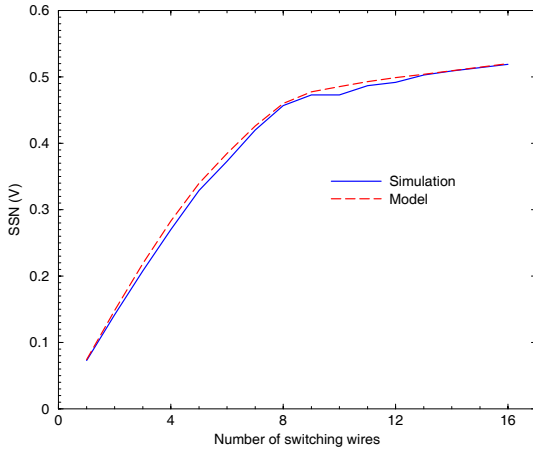


Figure 6. Simulated and estimated worst case SSN for a 16 wire bus as a function of the number of switching wires.

accuracy the results obtained by means of electrical simulations. In particular, the average accuracy is approximately the 98%. In the worst case (when all 16 wires switch), SSN overcomes the 20% of the nominal power supply.

6 Impact of EDCs on SSN

The developed model has been employed to evaluate the impact of several error detecting codes on SSN . We have considered the parity, Berger, cyclic and m-out-of-n codes,

which are most commonly used in practical applications [10, 12]. Such codes differ in their detection ability, whose evaluation, however, is out the scope of this work.

As an example, we have considered a bus with 16 information bits (plus the check bits required by the EDCs). However, analogous conclusions could be drawn for buses with different parallelisms.

We have supposed that the data bits can present all the possible (2^{16}) configurations, and that the switch of the bits is simultaneous. The last assumption has led us to pessimistic results. In fact, if we had taken into account the delay introduced by the encoder in the check bits switching, we would have obtained lower values of SSN . However, since the goal of this work is to estimate the SSN of different EDCs without going through time expensive electrical level simulations, we have made the above mentioned worst case assumption. In Fig. 7 we show the obtained results.

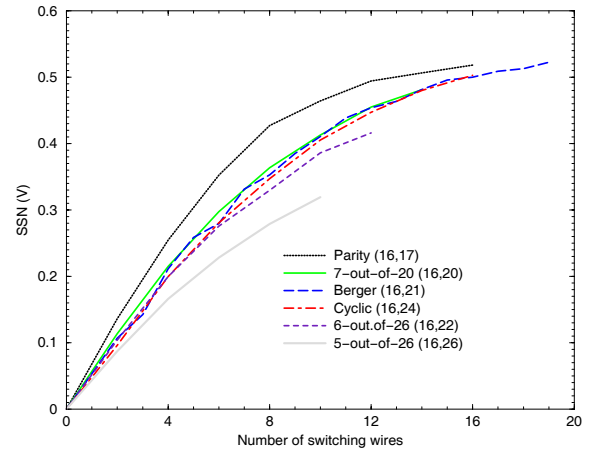


Figure 7. Worst case SSN for buses implementing different EDCs in case of 16 information bits, as a function of ΔBit .

All curves present the same behavior as that described in Sect. 4. They differ mainly in the maximum number of switching wires, due to the differences in the code characteristics. Let us describe more in detail the obtained results.

As for the parity code, it has $\Delta Bit_{max} = 16$. This because when all the 16 wires switch, the parity does not change. It implies the lowest redundancy, but for all values of $\Delta Bit_{max} \in [1, 16]$, it generates the highest SSN .

The Berger code is the one that, among those considered here, requires the highest number of simultaneous switching wires $\Delta Bit_{max} = 19$. In fact, the code is systematic, and for the made hypotheses, all the data bits can switch simultaneously. Besides this, we also have 3 of the 5 check bits that switch when all the data bits switch from 0 to 1. For the (16,24) cyclic code, similar considerations hold true.

The m-out-of-n codes present some different characteristics. Since they are not systematic, we can not distinguish

the data bits from the check bits within a codeword. Generally, for an m -out-of- n code, the following inequality has to be satisfied:

$$\binom{m}{n} \geq 2^k, \quad (9)$$

where k is the number of information bits to be encoded ($k = 16$ in this example).

In our case, it results: $\Delta Bit_{max} = 2m$ and $\#Bus = n$. As shown before, by reducing ΔBit_{max} we can reduce the SSN.

By implementing an m -out-of- n code, we can obtain a value of ΔBit_{max} lower than $k = 16$, which is the lowest number of switching wires considering a bus implementing a systematic code and supposing that the data bits can present all the possible (2^{16}) configurations. Therefore, we can select m -out-of- n codes with values of m satisfying the following inequality: $\Delta Bit_{max} = 2m < k = 16 \Rightarrow m < 8$. Choosing $m = 7$, we need a value of $n \geq 20$ to satisfy the inequality in Eq. 9. This leads to a 7-out-of-20 code, with $\Delta Bit_{max} = 14$. In order to further reduce ΔBit_{max} , thus reducing the SSN, we need to increase the redundancy. For instance, considering $m = 6$ (implying $\Delta Bit_{max} = 12$), we need a value of $n \geq 22$ (obtaining a 6-out-of-22 code), while, if $m = 5$ ($\Delta Bit_{max} = 10$), n should be equal to or greater than 26.

It is worth noticing that, since we need to represent always the same amount of information (2^{16} codewords), if we reduce the number of 1s per codeword (by reducing m), we need to increase the redundancy, that is $\#Bus = n$.

Tab. 1 resumes the obtained results.

Table 1. SSN noise for the considered EDCs, for a bus with 16 information bits.

Code	$\#Bus$	Redun.	ΔBit_{max}	SSN (mV)
Parity	17	6.25%	16	518
Berger	21	31.25%	19	523
Cyclic	24	50%	16	503
7-out-of-20	20	25%	14	481
6-out-of-22	22	31.6%	12	416
5-out-of-26	26	62.5%	10	320

As can be seen, the parity code is the most noisy for all values of ΔBit , while the less noisy among the considered ones is the 5-out-of-26 code. However, this latter implies the highest redundancy (62.5%), against the 6.25% required by the parity code.

7 Conclusions

In this paper we have performed a worst case analysis to evaluate how different on-chip bus transitions impact the SSN. We have highlighted that, because of the effect of the coupling capacitances between adjacent wires, the SSN presents a piece-wise linear dependency on the number of

switching wires. This analysis has led us to the definition of a semi-empirical analytical model able to predict very accurately the worst case SSN, given the technology and the bus parallelism. This model has been employed to estimate the impact of several EDCs on SSN. We have verified that the m -out-of- n code, allows to reduce drastically the number of simultaneous switching wires, thus the SSN. The drawback is its high redundancy, besides the complexity of the encoding/decoding circuitry.

References

- [1] Semiconductor Industry Assoc., San Jose, Calif. *The 2003 National Technology Roadmap for Semiconductors*, 2003.
- [2] X. Aragones, J. L. Gonzales, F. Moll, A. Rubio. Noise Generation and Coupling Mechanisms in Deep-Submicron ICs. *IEEE Design & Test of Computers*, pages 27 – 35, Sept.-Oct. 2002.
- [3] G. Bai and I. N. Hajj. Simultaneous Switching Noise and Resonance Analysis of On-Chip Power Distribution Network. In *Proc. of IEEE Int. Symp. on Quality Electronic Design*, pages 163 – 168, 2002.
- [4] F. Caignet, S. Delmas-Bendhia, and E. Sicard. The Challenge of Signal Integrity in Deep-Submicrometer CMOS Technology. *Proc. of IEEE*, pages 556 – 573, April 2001.
- [5] Y. Chang, S. K. Gupta, and C. Koh. Analysis of Ground Bounce in Deep Sub-Micron Circuits. In *Proc. of IEEE VLSI Test Symposium*, pages 110 – 116, 1997.
- [6] C. L. Chen and B. W. Curran. Switching Codes for Delta-I Noise Reduction. *IEEE Trans. on Computers*, pages 1017 – 1021, September 1996.
- [7] M. Favalli, C. Metra. Bus Crosstalk Fault-Detection Capabilities of Error Detecting Codes for On-Line Testing. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, pages 392 – 396, Sep. 1999.
- [8] R. Hegde, N. R. Shanbhag. Toward achieving Energy Efficiency in Presence of Deep Submicron Noise. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, pages 379 – 391, Aug. 2000.
- [9] P. Heydari and M. Pedram. Ground Bounce in Digital VLSI Circuits. *IEEE Trans. on VLSI Systems*, pages 180 – 193, April 2003.
- [10] P. K. Lala. *Self-Checking and Fault-Tolerant Digital Design*. Morgan Kaufmann, 2001.
- [11] A. K. Nieuwland, A. Katoch, D. Rossi, C. Metra. Coding Techniques for Low Switching Noise in Fault Tolerant Busses. In *Proc. of IEEE Int. On-Line Testing Symp.*, pages 183 – 189, 2005.
- [12] D. K. Pradhan. *Fault Tolerant Computing: Theory and Techniques*. Prentice Hall, Englewood Cliffs, New Jersey, 1986.
- [13] D. Rossi, A. Muccio, A. K. Nieuwland, A. Katoch, C. Metra. Impact of ECCs on Simultaneously Switching Output Noise for On-Chip Busses of High Reliability Systems. In *Proc. of IEEE Int. On-Line Testing Symp.*, pages 135 – 140, 2004.
- [14] J. Segura and C. F. Hawkins. *CMOS Electronics: How It Works, How It Fails*. IEEE Press, 2004.
- [15] K. T. Tang and E. G. Friedman. Simultaneous Switching Noise in On-Chip CMOS Power Distribution Network. *IEEE Trans. on VLSI Systems*, pages 487 – 493, Aug. 2002.