# Novel Cross-Transition Elimination Technique Improving Delay and Power Consumption for On-Chip Buses

Antoine Courtay[1,2], Johann Laurent[1], Olivier Sentieys[2], Nathalie Julien[1]

[1] Université Européenne de Bretagne - UBS, Lab-STICC, rue de saint maudé,
56100 Lorient, France
firstname.lastname@univ-ubs.fr
[2] Université de Rennes, IRISA, rue de kerampont,
22300 Lannion, France
lastname@irisa.fr

**Abstract.** Interconnects are now considered as the bottleneck in the design of system-on-chip (SoC) since they introduce delay and power consumption. To deal with this issue, data coding for interconnect power and timing optimization has been introduced. In today's SoCs these techniques are not efficient anymore due to their codec complexity or to their unrealistic experimentations. Based on some realistic observations on interconnect delay and power estimation, the spatial switching technique [1] is proposed. It allows the reduction of delay and power consumption (including extra power consumption due to codecs) for on-chip buses. The concept of the technique is to detect all cross-transitions on adjacent wires and to decide if the adjacent wires are exchanged or not. Results show the spatial switching efficiency for different technologies and bus lengths. The power consumption reduction can reach up to 12% for a 5-mm bus and more if buses are longer.

## 1 Introduction

Today's System-on-Chip (SoC) are more and more complex and require many computational resources, implying a large volume of data to be stored or to be transmitted. To transfer this data from memory to processor or from one processor to another, on-chip interconnect buses or networks have to be used. In state-of-the-art SoC, interconnect can represent up to 50% of the total power consumption [2]. Moreover, the transistor and wire dimension scaling has an increasingly strong impact on the propagation time and the energy due to wires [3]. Therefore, estimation and optimization of power and delay due to interconnections became a major issue in SoC design. Many works have been proposed in the past around interconnect power optimization at different abstraction levels [4–14].

Unfortunately most of the proposed techniques are not efficient for reducing the power consumption due to interconnects. This is first due to the fact that interconnect length used in their experimentations are not corresponding to nowadays' SoCs ones. Therefore, the efficiency of most techniques designed for long interconnects is not valid anymore [15]. It is also due to the hardware complexity used by codecs of the published techniques which consumes too much power compared to the power reduction gained on buses. Finally, previous works showed that, in the case of crosstalk effects, the impact of reducing the delay is quite different of reducing the power.

In this paper we propose a novel cross-transition elimination technique (spatial switching) which improves delay and power consumption for on-chip buses. This paper is organized as follows. Section 2 presents the starting key issues that allowed us to develop the spatial switching technique. Spatial switching technique and its hardware implementation are described in Section 3. In Section 4, experimental results in terms of power consumption and frequency variation using spatial switching are discussed. The last section concludes this paper.

## 2 Previous Works

Crosstalk is the effect of the coupling capacitance between a victim wire and its neighboring wires which depends on their transitions and states. Results presented in [16] show that the transition classification (according to the crosstalk capacitance presented by the victim wire) differ if power consumption or delay is considered. Table 1, extracted from [16], shows that transition classification from a power consumption point of view starts with rising transitions followed by falling transitions. Therefore, a first key point for power optimization is to encode data such as falling transitions on the bus are achieved with the lowest

**Table 1.** Delay and power consumption results according to the transition patterns. The total capacitance of the wire depends on $C_s$ which is the wire-to-substrate capacitance and $C_c$ which is the crosstalk capacitance. Delays are measured by computing the time taken by the output to switch half the supply voltage value and is given in picoseconds. Power consumption is given as energy per transition expressed in femtojoules. Here, ↑ represents a rising transition, ↓ represents a falling one and − means that there is no transition on the wire

| Delay classification | | | Energy classification | | |
|---|---|---|---|---|---|
| $(-, -, -)$ | 0 | 0 ps | $(-, -, -)$ | 0 | 0.21 fJ |
| $(\uparrow, \uparrow, \uparrow)$ | $C_s$ | 49 ps | $(\uparrow, \uparrow, \uparrow)$ | $C_s$ | 13.29 fJ |
| $(\downarrow, \uparrow, \uparrow)$ | $C_s$ | 49 ps | $(-, \uparrow, \uparrow)$ | $C_s + C_c$ | 13.43 fJ |
| $(-, \uparrow, \uparrow)$ | $C_s + C_c$ | 67 ps | $(-, \uparrow, -)$ | $C_s + 2C_c$ | 13.45 fJ |
| $(-, \downarrow, \downarrow)$ | $C_s + C_c$ | 67 ps | $(\uparrow, \uparrow, \downarrow)$ | $C_s + 2C_c$ | 13.89 fJ |
| $(\uparrow, \uparrow, \downarrow)$ | $C_s + 2C_c$ | 99 ps | $(-, \uparrow, \downarrow)$ | $C_s + 3C_c$ | 14.10 fJ |
| $(-, \uparrow, -)$ | $C_s + 2C_c$ | 99 ps | $(\downarrow, \uparrow, \downarrow)$ | $C_s + 4C_c$ | 14.86 fJ |
| $(-, \downarrow, -)$ | $C_s + 2C_c$ | 99 ps | $(\downarrow, \downarrow, \downarrow)$ | $C_s$ | 33.77 fJ |
| $(\downarrow, \downarrow, \uparrow)$ | $C_s + 2C_c$ | 99 ps | $(-, \downarrow, \downarrow)$ | $C_s + C_c$ | 92.00 fJ |
| $(-, \uparrow, \downarrow)$ | $C_s + 3C_c$ | 139 ps | $(-, \downarrow, -)$ | $C_s + 2C_c$ | 150.35 fJ |
| $(-, \downarrow, \uparrow)$ | $C_s + 3C_c$ | 139 ps | $(\downarrow, \downarrow, \uparrow)$ | $C_s + 2C_c$ | 150.73 fJ |
| $(\downarrow, \uparrow, \downarrow)$ | $C_s + 4C_c$ | 173 ps | $(-, \downarrow, \uparrow)$ | $C_s + 3C_c$ | 207.76 fJ |
| $(\uparrow, \downarrow, \uparrow)$ | $C_s + 4C_c$ | 173 ps | $(\uparrow, \downarrow, \uparrow)$ | $C_s + 4C_c$ | 265.07 fJ |

crosstalk capacitance and thus consume less energy as possible.

Secondly, when the activity profile of data stimuli files is analyzed, it can be noticed that applying performance optimization techniques on least significant bits (LSB) has a better impact in terms of power consumption reduction. This is due to the fact that the LSBs have the strongest activity as demonstrated in [17]. For instance, the Partial Bus Invert technique [11] (Partial Bus Invert is Bus Invert [10] applied to LSBs) has better results in terms of power consumption reduction than the classical Bus Invert.

Thirdly, if the transition class percentage of appearance is analyzed, it can be noticed that transitions which are eliminated by performance optimization techniques are not appearing very often. For example, eliminating the two last transitions (↓, ↑, ↓) and (↑, ↓, ↑), as it is often proposed for delay optimization, is not efficient for power reduction since these transitions have a very low arising probability.

Finally, most of the presented techniques do not always take into account the extra power consumption due to codecs, that is the reason why the results presented in [15] show that the extra power consumption due to codecs is often higher than the power consumption reduction on buses.

Based on the work presented in [16], some new key issues for timing and power consumption optimization can be put forward:

- Do not only focus on $C_s + 3C_c$ and $C_s + 4C_c$ transitions since they are not predominant in the total transition number.
- Focus on the lines with the largest data activity (i.e. LSB) because these are the more consuming lines.
- Avoid falling transitions as much as possible: a key point for power optimization can be to encode data such as falling transitions on the bus are achieved with the lowest crosstalk capacitance and thus consume less energy as possible.
- Design codec with power overhead as low as possible, and therefore focus on very simple techniques.

The next section introduces the concept of our technique for the optimization of delay and power consumption in the case of on-chip buses.

## 3 Proposed Optimization Technique

The spatial switching technique uses the classical architectural coding which consists in an encoding block at the bus input and a decoding block at the bus output as illustrated in Fig. 1. The encoder aim is to code the $n$ input bits into $m$ bits for the coded bus (with $m \geq n$), and the decoder realizes the inverse coding to recover the original values.

When using spatial switching, the $n$-bit bus is divided into sets of two wires. Each pair of wires is coded with the technique described below and an extra control wire is added at the output of the block as illustrated in Fig. 2. A total of $n/2$ extra wires is therefore needed. The principle of the coding is to replace

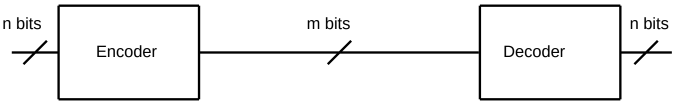

**Fig. 1.** Principle of the bus coding technique

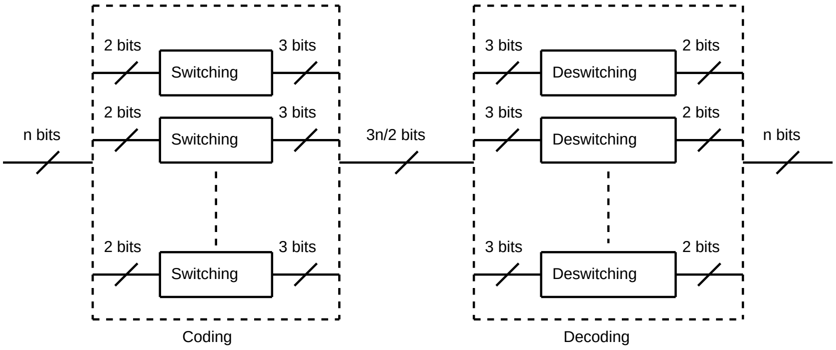

**Fig. 2.** Macro-blocks of spatial switching coding and decoding blocks

consuming transitions by less consuming ones such as transitions on the bus that are achieved with the lowest crosstalk capacitance. The key point of the technique is to detect all cross-transitions on adjacent wires, if a cross-transition is detected, data on the two wires are crossed. Then, a signal is sent to the decoder with the extra control wire to enable the correct redirection of data at the bus output.

Let $D_{W_j}(t_i)$ be the data on wire $j$ at clock cycle $t_i$, $D_{W_j}(t_{i-1})$ be the data on

**Table 2.** Algorithm and logical functions used in the spatial switching technique

| IF | THEN | ELSE |
|---|---|---|
| $\left[D_{W_j}(t_i) \neq D_{W_{j+1}}(t_i)\right]$ AND<br>$\left[D_{W_j}(t_i) \neq D_{W_j}(t_{i-1})\right]$ AND<br>$\left[D_{W_{j+1}}(t_i) \neq D_{W_{j+1}}(t_{i-1})\right]$ | $D_{W_j}(t_i) = D_{W_{j+1}}(t_i)$<br>$D_{W_{j+1}(t_i)} = D_{W_j(t_i)}$<br>$Switch = 0$ | $D_{W_j}(t_i) = D_{W_j}(t_i)$<br>$D_{W_{j+1}(t_i)} = D_{W_{j+1}(t_i)}$<br>$Switch = 1$ |
| IF | THEN | ELSE |
| $\left[D_{W_j}(t_i) \oplus D_{W_{j+1}}(t_i)\right]$ AND<br>$\left[D_{W_j}(t_i) \oplus D_{W_j}(t_{i-1})\right]$ AND<br>$\left[D_{W_{j+1}}(t_i) \oplus D_{W_{j+1}}(t_{i-1})\right]$<br>$= 0$ | $D_{W_j}(t_i) = D_{W_{j+1}}(t_i)$<br>$D_{W_{j+1}(t_i)} = D_{W_j(t_i)}$<br>$Switch = 0$ | $D_{W_j}(t_i) = D_{W_j}(t_i)$<br>$D_{W_{j+1}(t_i)} = D_{W_{j+1}(t_i)}$<br>$Switch = 1$ |

wire $j$ at clock cycle $t_{i-1}$, $D_{W_{j+1}}(t_i)$ be the data on wire $j+1$ at clock cycle $t_i$ and $D_{W_{j+1}}(t_{i-1})$ be the data on wire $j+1$ at clock cycle $t_{i-1}$; the algorithm used by spatial switching can be expressed as in Table 2. A cross-transition occurs when data on the two neighbor wires are different at clock cycle $t_i$ and at clock cycle $t_{i-1}$, and also when data on the same wire are different at clock cycles $t_i$ and $t_{i-1}$.

To achieve this test, XOR gates are used. If all XOR gate outputs are equal to 1 (i.e. there is a cross-transition), the extra switching control signal is set to 0. The generation of the switching signal is illustrated in Fig.3 in the cross-transition detection block. Fig. 3 also presents the switching block (i.e. routing of data on neighbor wires ), this block consists of two (2:1) multiplexors with their outputs depending on the switching control signal. At the bus end, a reverse switching block is used (also driven by the switching control signal) to recover the original data on the two neighbor wires.

Fig. 4 illustrates the spatial switching on a simple data flow example. The original data flow contains two cross-transitions. When applying spatial switching, the coded data flow does not contain cross-transition anymore. Next section presents the power consumption improvements of our technique for different technological parameters.

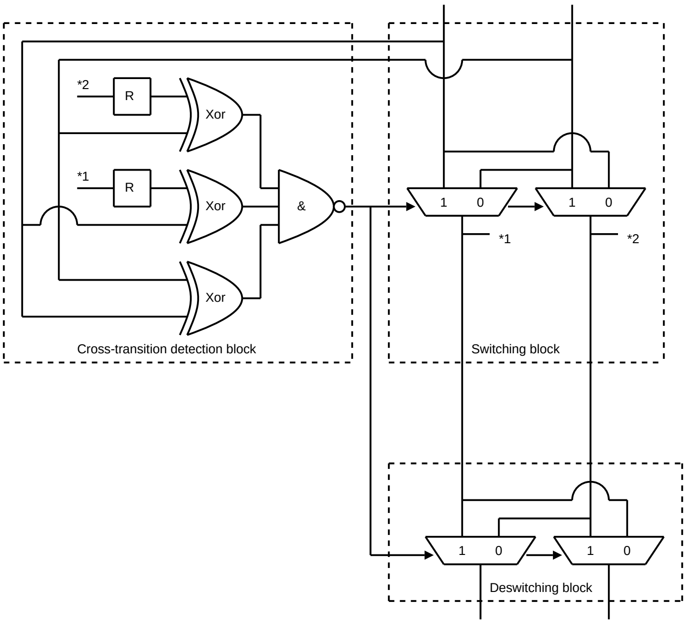

**Fig. 3.** Detailed cells used in spatial switching coding and decoding blocks

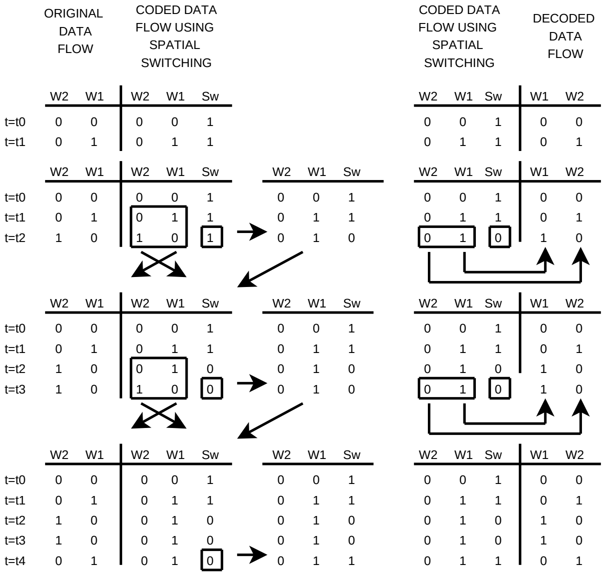
ORIGINAL DATA FLOW
CODED DATA FLOW USING SPATIAL SWITCHING
CODED DATA FLOW USING SPATIAL SWITCHING
DECODED DATA FLOW
W2
W1
Sw
t=t0
t=t1
t=t2
t=t3
t=t4

**Fig. 4.** Spatial switching applied to a simple data flow example where $W_i$ represents the $i^{th}$ wire and $Sw$ the switching signal

## 4 Experimental Results

For our simulations, the buses have been modeled as a distributed $RC\Pi3$ model considering crosstalk capacitances as defined in [16]. Experimental results have been obtained using a SPICE simulator (ELDO v5.7) for different technologies (130nm, 90nm and 65nm) and for an image stimuli file (results follow the same behavior for other data files such as music or speech).

Each technology has a specific number of metal layers: 5 for 130nm, 6 for 90nm and 7 for 65nm. SPICE simulations have been achieved on all metal layers from the lowest ones (mostly reserved for short wires) to the highest ones (mostly reserved for buses which is our topic of interest).

As propagation time becomes critical on interconnects, some techniques are proposed in [18], [19], [20] to accelerate the data propagation by inserting some buffers on wires. Thus, in our experimentations, buffered and non-buffered interconnects have been simulated.

As both power consumption and propagation time are key issues in SoC and depends on the transistor size, the transistor width $W$ of the codecs has been considered as a parameters in our simulations. $W$ varies from twice the technology length $2\lambda$ (which is the minimum design rule fixed by technological design kits) to 6 times the technology length. Power consumption results have been obtained considering the extra power consumption due to codecs. First, it can be noticed that the spatial switching efficiency increases with technology shrinking,

**Table 3.** Power consumption variation on buses including extra codec power consumption expressed in % for different technologies, metal layers, bus lengths, transistor lengths $\lambda$ and bit number on which spatial switching is applied (2LSB to 8LSB). When power consumption is saved values are positive, when power is increased (i.e. power consumption due to codecs is higher than the power consumption saved on the encoded bus), values are negative

| | 90nm / Layer 3 / 3mm | | | | 90nm / Layer 6 / 3mm | | | | 90nm / Layer 6 / 5mm | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2LSB | 4LSB | 6LSB | 8LSB | 2LSB | 4LSB | 6LSB | 8LSB | 2LSB | 4LSB | 6LSB | 8LSB |
| $4\lambda$ | 4.42 | 7.52 | 10.44 | 10.00 | 4.42 | 7.74 | 10.36 | 9.95 | 4.95 | 8.35 | 11.75 | 12.05 |
| $6\lambda$ | 3.11 | 4.88 | 6.49 | 4.73 | 3.11 | 5.05 | 6.73 | 5.10 | 4.31 | 7.06 | 9.82 | 9.47 |
| $8\lambda$ | 2.09 | 2.86 | 3.45 | 0.68 | 2.09 | 3.11 | 3.82 | 1.22 | 3.78 | 6.01 | 8.25 | 7.38 |
| $10\lambda$ | 1.85 | 2.38 | 2.73 | -0.28 | 1.85 | 1.30 | 1.10 | -2.40 | 3.22 | 4.89 | 6.56 | 5.13 |
| $12\lambda$ | -3.75 | -0.52 | -1.61 | -6.07 | 0.41 | -0.24 | -1.20 | -5.47 | 2.60 | 3.66 | 4.71 | 2.66 |

| | 65nm / Layer 3 / 3mm | | | | 65nm / Layer 7 / 3mm | | | | 65nm / Layer 7 / 5mm | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2LSB | 4LSB | 6LSB | 8LSB | 2LSB | 4LSB | 6LSB | 8LSB | 2LSB | 4LSB | 6LSB | 8LSB |
| $4\lambda$ | 4.06 | 7.11 | 10.19 | 10.20 | 4.15 | 7.26 | 10.46 | 10.50 | 4.11 | 7.38 | 10.64 | 10.55 |
| $6\lambda$ | 3.63 | 6.25 | 8.90 | 8.48 | 3.74 | 6.44 | 9.22 | 8.86 | 3.87 | 6.90 | 9.92 | 9.59 |
| $8\lambda$ | 3.18 | 5.35 | 7.54 | 6.67 | 3.31 | 5.57 | 7.92 | 7.12 | 3.62 | 6.40 | 9.18 | 8.60 |
| $10\lambda$ | 2.71 | 4.41 | 6.13 | 4.79 | 2.84 | 4.64 | 6.53 | 5.26 | 3.36 | 5.88 | 8.40 | 7.57 |
| $12\lambda$ | 2.20 | 3.38 | 4.60 | 2.74 | 2.37 | 3.70 | 5.12 | 3.38 | 3.10 | 5.38 | 7.61 | 6.51 |

which is a major issue for power consumption reduction in current and future technologies.

Secondly, the spatial switching is already efficient on the lowest metal layers, but it is more efficient on the highest metal layers reserved in particular for buses which is our topic of interest.

Thirdly, Table 3 shows that the longer the bus is, the higher the power consumption reduction. The power consumption reduction can rise up to 12% for a 5-mm bus and more for longer buses. In nowadays SOC, buses can vary on average from 1 mm to 7 mm for very long buses. Therefore, spatial switching can bring a significant power consumption reduction for this bus length variation range. In state-of-the-art optimization techniques, the interconnect length used for experimental results are not often realistic. For instance, the technique used in [12] claims power consumption reduction for a 7.5 cm bus length. In addition, results presented in [15] show that many encoding techniques start to be efficient for extremely long buses because of the extra power due to codecs (e.g. 2 cm for Bus Invert). Moreover, many coding techniques ([13], [14] for instance) do not always take the extra power consumption due to codecs into account, when presenting power consumption results for buses. Table 4 underlines this last remark, where the comparison between power consumption reduction on busses without considering extra power consumption due to codecs (Case 1) and power consumption reduction on busses considering extra power consumption due to codecs is done between spatial switching (SS) and partial bus invert (PBI). It can

**Table 4.** This Table presents the comparison between the Spatial Switching (SS) and the Partial Bus Invert (PBI) technique. Case 1 is the power consumption reduction (in %) on the bus without the extra power consumption due to codecs. Case 2 is the power consumption reduction (in %) on the bus considering the extra power consumption due to codecs. The transistor, register and extra wire number is also compared in this table.

| | Case 1 | | Case 2 | | Transistors | | Registers | | Extra wire(s) | |
|---|---|---|---|---|---|---|---|---|---|---|
| | SS | PBI | SS | PBI | SS | PBI | SS | PBI | SS | PBI |
| 2LSB | 6.2 | 3.9 | 4.4 | <0 | 48 | 48 | 2 | 2 | 1 | 1 |
| 4LSB | 11.1 | 10.0 | 7.5 | <0 | 96 | 144 | 4 | 4 | 2 | 1 |
| 6LSB | 15.8 | 14.3 | 10.4 | <0 | 144 | 240 | 6 | 6 | 3 | 1 |
| 8LSB | 17.1 | 6.8 | 10.0 | <0 | 192 | 336 | 8 | 8 | 4 | 1 |

be noticed that the complexity in terms of transistor number impact strongly the power consumption reduction (compared to the addition of an extra wire), that is the reason why partial bus invert which is more complex than spatial switching has a negative impact on power consumption reduction when considering power consumption due to codecs.

Fourthly, it can be noticed that there is an optimal value of the bit number on which spatial switching can be applied. Spatial switching aims at removing cross-transitions, this kind of transitions has a higher probability of appearing where activity on the bus is strong. It is well known that for data flows the activity is the strongest on the least significant bits as demonstrated in [17]. When looking at the results of Table 3, it can be observed that the power consumption reduction is increased when applying spatial switching on the 2, 4 and 6 LSB, whereas on 8 LSB it starts to decrease. This is due to the fact that the MSBs are more correlated than the LSBs, and there are thus less cross-transitions. This decrease of the power consumption gain is due to the fact that for the 2 MSB wires, spatial switching codecs start to consume more power than the power consumption reduction on these two encoded wires.

Finally, Table 5 shows that, the larger the transistors are, the higher the codec

**Table 5.** Running frequency in codecs expressed in GHz for different technologies, metal layers, bus lengths and transistor lengths

| | 130nm / Layer 5 | | | | 90nm / Layer 6 | | | | 65nm / Layer 7 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1mm | 3mm | 5mm | 7mm | 1mm | 3mm | 5mm | 7mm | 1mm | 3mm | 5mm | 7mm |
| $4\lambda$ | 3.48 | 2.65 | 2.16 | 1.83 | 3.68 | 2.21 | 1.58 | 1.23 | 7.93 | 4.06 | 2.75 | 1.52 |
| $6\lambda$ | 3.94 | 3.26 | 2.75 | 2.39 | 3.81 | 2.73 | 2.21 | 1.75 | 9.36 | 5.43 | 3.85 | 2.22 |
| $8\lambda$ | 4.25 | 3.68 | 3.12 | 2.83 | 3.85 | 3.02 | 2.46 | 2.08 | 10.31 | 6.52 | 4.74 | 2.84 |
| $10\lambda$ | 4.46 | 3.97 | 3.46 | 3.14 | 4.00 | 3.22 | 2.75 | 2.34 | 10.86 | 7.50 | 5.52 | 3.41 |
| $12\lambda$ | 4.62 | 4.17 | 3.75 | 3.38 | 4.12 | 3.37 | 2.94 | 2.56 | 11.25 | 8.14 | 6.20 | 3.94 |

frequency can be. If the frequency is not a restrictive parameter, minimum sized transistors are the best choice for power consumption reduction on buses. If frequency is significant, then a trade-off has to be obtained between the bus frequency and the best power consumption reduction.

## 5 Conclusion

Based on some new key issues for interconnect delay and power optimization, a patented technique called *spatial switching* is presented. The technique aims at removing from an on-chip bus the most consuming transitions that are defined to be the falling cross-transitions.
After the presentation of the algorithm used in spatial switching, its implementation has been proposed. Then, experimental results in terms of power consumption reduction using three different technologies and their associated metal layers for different technological parameter variations are presented.
Results show the spatial switching efficiency for technologies and bus length used in today's SoCs. Results in terms of power consumption reduction can reach up to 12% for a 5-mm bus and more if buses are longer.
Future works on the spatial switching technique will focus on an hardware FPGA implementation and a DSP software version of the codecs.

## 6 Acknowledgments

This work has been supported by the European Union and the Brittany Region in the context of *Programme Objectif 2 Bretagne 2000-2006.*

## References

1. Courtay, A., Laurent, J., Sentieys, O., Julien, N.: Patent reference BFF 08P0103/HC
2. Magen, N., Kolodny, A., Weiser, U., Shamir, N.: Interconnect-power dissipation in a microprocessor. Proceedings of the International Workshop on System Level Interconnect prediction (2004), pp.7–13
3. Ho, R., Mai, K., Horowitz, M.: The future of wires, Proceedings of the IEEE (2001), Vol. 89, No. 4, pp.490–504
4. Hirose, K., Yasuura, H.: A bus delay reduction technique considering crosstalk. Proceedings of the Conference on Design, Automation and Test in Europe (2000), pp.441–445
5. Shang, L., Peh, L., Jha, N.K.: Dynamic voltage scaling with links for power optimization of interconnection networks. Proceedings of the 9th International Symposium on High-Performance Computer Architecture (2003), pp.91–102
6. Macchiarulo, L., Macci, E., Poncio, M.: Wire placement for crosstalk energy minimization in address buses. Proceedings of the Conference on Design, Automation and Test in Europe (2002), pp.158–162.

7. Khatri, S.P., Brayton, R.K., Sangiovanni-Vincentelli, A.L.: Crosstalk Noise Immune VLSI Design Regular Layout Fabrics. Kluwer Academic Publishers, Hingham, MA, USA (2001)
8. Taylor, C.N., Dey, S., Zhao, Y.: Modeling and minimization of interconnect energy dissipation in nanometer technologies. Proceedings of the 38th Conference on Design Automation (2001), pp.754–757
9. Su, C.L., Tsu, C.Y., Despain, A.M.: Saving power in the control path of embedded processors. IEEE Design & Test of Computers (1994), Vol. 11, No. 4, pp.24–31.
10. Stan, M.R., Burleson, W.P.: Bus-invert coding for low-power I/O. IEEE Trans. on Very Large Scale Integration Systems (1995), Vol. 3, No. 1, pp.49–58
11. Shin, Y., Chae, S.-IK., Choi, K.: Partial bus-invert coding for power optimization of system level bus. Proceedings of the International Symposium on Low Power Electronics and Design (1998), pp.127–129
12. Komatsu, S., Ikeda, M., Asada, K.: Low power chip interface based on bus data encoding with adaptive code-book method. Proceedings of the 9th IEEE Great Lakes Symposium on VLSI (1999), pp.368–371
13. Benini, L., Micheli, E., Macii, E., Sciuto, D., Silvano, C.: Asymptotic zero-transition activity encoding for address busses in low-power microprocessor based systems. Proceedings of the 7th IEEE Great Lakes Symposium on VLSI (1997), pp.77–82
14. Philippe, J.M., Pillement, S., Sentieys, O.: Area efficient temporal coding schemes reducing crosstalk effects. Proceedings of the International Symposium on Quality Electronic Design (2006), pp.334–339
15. Kretzschmar, C., Nieuwland, A.K., Muller, D.: Why transition coding for power minimization of on-chip buses does not work. Proceedings of the Conference on Design, Automation and Test in Europe (2004), pp.10512–10517
16. Courtay, A., Sentieys, O., Laurent, J., Julien, N.: High-Level Interconnect Delay and Power Estimation. J. Low Power Electronics (2008), Vol. 4, No. 1, pp.21–33
17. Landman, P.E., Rabaey, J.M.: Architectural power analysis: the dual bit type method. IEEE Trans. on Very Large Scale Integration Systems (1995), Vol. 3, No. 2, pp.173–187.
18. Bakoglu, H.B., Meindl, J.D.: Optimal interconnection circuits for VLSI. IEEE Trans. on Electron. Devices (1985), Vol. 32, No. 5, pp.903–909.
19. Namalpu, A., Burleson, W.P.: Optimal wire sizing and buffer insertion for low power and a generalized delay model. Proceedings of the IEEE International Conference on ASIC/SOC (2001), Vol. 31, No. 3, pp.437–447
20. Chen, G., Friedman, E.G.: Low-power repeaters driving RC and RLC interconnects with delay and bandwidth constraints. IEEE Trans. on VLSI (2006), Vol. 14, No. 2, pp.161–172