

# Why Transition Coding for Power Minimization of on-Chip Buses does not work\*

Claudia Kretzschmar<sup>1</sup>, André K. Nieuwland<sup>2</sup>, Dietmar Müller<sup>3</sup>

<sup>1,3</sup>Dpt. of Systems and Circuit Design, Chemnitz University of Technology, Germany

<sup>2</sup>Philips Research Eindhoven, Netherlands

email: clkre@infotech.tu-chemnitz.de, andre.nieuwland@philips.com

## Abstract

*Encoding techniques which minimize the self- or coupling activity of buses are often proposed to reduce power dissipation on system buses. In this paper, we investigate the efficiency of several coding schemes for on-chip buses with respect to overall power dissipation. The power of the codec systems was estimated by power simulations with the lay-outs and related to the savings on the bus. We derived an expression for the energy efficiency of the codecs as a function of bus length (capacitive load). Despite the fact that adaptive schemes could obtain up to 40% savings, the bus lengths required to reduce the overall power consumption are not realistic for on-chip buses.*

## 1. Introduction

The power dissipation of integrated circuits is becoming a limiting factor for SoCs [11]. Meeting power objectives is becoming as important as meeting performance targets [12].

A large portion of power is dissipated on system buses due to the wire capacitance. Since the coupling capacitances between adjacent wires increase as a result of technology scaling, wire capacitances even growingly influence the system power consumption. Therefore power minimization is an issue for system buses. The power dissipated on system buses can be calculated by Equation (1):

$$P_{bus} = \frac{1}{2} V_{dd}^2 f C \alpha, \quad (1)$$

where  $f$  is the clock frequency,  $V_{dd}$  the operating voltage,  $C$  the capacitance of the bus and  $\alpha$  the average transition activity of the bus lines respectively. In order to save power by reducing activity, a huge number of encoding techniques were published which exploit different characteristics of the data streams. The idea is to displace tran-

sitions from the high-capacitive bus into the less-capacitive encoder-decoder (codec) circuitry. Thereby, techniques of the first generation such as in [1], [6], [9], [10] focus on reducing the transitions per line (self-activity) while schemes published later [5], [14] take into account the activity between neighbouring lines (coupling activity) in order to reduce the charge of coupling capacitances.

Quite often the reduction in power dissipation claimed is based on the savings on the bus. In contrast to that we explore the efficiency of transition-minimizing bus encoding schemes for on-chip buses with respect to the total power balance. Therefore, we analysed the power dissipation of coder-decoder systems for several encoding schemes (implemented in a 0.13  $\mu\text{m}$  CMOS technology), and related their power dissipation with the savings they obtained for on-chip buses.

The paper is structured as follows: Section 2 gives an overview of related work. In Sect. 3 we define a set of parameters in order to judge the efficiency of encoding schemes. For selected schemes the reduction in activity and the power consumption are presented in Sect. 4. In Sect. 5 a model for bus power dissipation is developed while the efficiency with respect to total power dissipation is explored in Sect. 6.1. We investigated the impact of technology scaling in Sect. 6.2 and summarize the results in Sect. 7.

## 2. Related Work on Bus encoding

A huge number of bus encoding schemes were published in the literature, which exploit application-specific parameters of the data streams to be transmitted. The overview given here is not complete but tries to mention schemes of all classes. Depending on the way the encoding *rule* is implemented, bus encoding schemes are either: static or adaptive. Static schemes can only be optimized at design time based on an application-specific data stream. Adaptive techniques observe the statistics of the data stream *during operation* and adjust their encoding rules accordingly.

According to the application domain the schemes perform differently. Address bus data streams are characterized

\*This work is funded by the Deutsche Forschungsgemeinschaft DFG within the VIVA research initiative under MU 1024/5-2 and by the European community within FET-open, IST-2001-38930

by the high in-sequence portion of subsequently transmitted words. Static schemes such as Gray encoding [13], the Beach solution [2] and combined schemes such as T0-BI, Dual-T0 and Dual-T0-BI [3] are tailored on this property.

Streams transmitted on control or data buses have different characteristics and require therefore general-purpose techniques. A well known example of a static scheme is Bus-Invert (BI) [10] which does not need a priori knowledge of the data streams and has a rather simple codec implementation. BI limits the maximum Hamming distance between two successive bus states to one half of the number of bus lines by inverting the data word if more lines would switch. The inversion is indicated on an additional line. In case the activity is not equally distributed over bus lines inverting the complete data word can cause additional transitions on lines with a low activity. Therefore the Partial Bus-Invert Encoding scheme (PBI) [9] excludes such lines from BI encoding. Since the statistics is application-specific the Adaptive Partial Bus-Invert (APBI) encoding scheme [6] periodically adapts the subset of lines to be included into encoding. An adaptive scheme which we refer to as IAEB was published in [1]. This technique analyzes the number of state changes on each bus line within a sampling window of fixed size and selects the most efficient out of four possible encoding schemes.

A scheme focussed on adapting words instead of (groups of) lines is the Adaptive Minimum Weight Codes (AMWC) [7]. Based on the statistics, observed in a window of variable size, AMWC maps data words on a non-redundant minimum weight code. A word based approach is also used by the Adaptive code-book method [8]. In a continuously updated table the word with the smallest Hamming distance is searched. The index and the difference are transmitted over the bus.

This scheme also takes into account coupling activity. Two other schemes which focus on the minimization of coupling activity instead of self-activity are the coupling-driven Businvert [5] and the Odd/Even Businvert scheme [14].

### 3. Efficiency Metrics

Encoding schemes differ in their effectivity of reducing the switching activity on the bus. Therefore, the encoding efficiency  $E_\alpha$  which we define by equation (2) is an important parameter to compare encoding schemes.

$$E_\alpha = 1 - \frac{\alpha_{coded}}{\alpha_{uncoded}}. \quad (2)$$

Note that  $E_\alpha$  refers only to activity reduction and is therefore technology independent.

As indicated by equation (1), the power savings on the bus are related linearly with the reduction in switching activity. In order to determine the savings in overall power consumption, the self dissipation of codec circuits has to be

considered as well. The power savings  $P_{saved}$  are defined by Equation (3):

$$P_{saved} = P_{unc} - P_{cod} - P_{Codec}, \quad (3)$$

where  $P_{unc}$  and  $P_{cod}$  represent the power dissipated on an uncoded and a coded bus, respectively and  $P_{Codec}$  represents the power dissipated by the codec system.

From the power balance (3) we derive the energy efficiency  $E_P$  (4) in order to ascertain the percentage of bus power saved due to coding:

$$E_P = \frac{P_{saved}}{P_{unc}} = 1 - \frac{P_{cod} + P_{Codec}}{P_{unc}}. \quad (4)$$

If energy is saved the energy efficiency  $E_P$  is positive. For negative  $E_P$  the total power consumption is increased by this encoding technique. To save energy, the codec power has to be small and/or the power consumption of the encoded bus has to be sufficiently lower than the original uncoded bus.

The power consumed on the uncoded and encoded bus depends on the bus capacitance, which is linearly related to the length  $L$  of the bus as indicated by the extension of equation (1):  $P_{bus} = \frac{1}{2} V_{dd}^2 f \frac{C_{bus}}{|mm|} L \alpha$ . The longer the bus, the less the influence of the codec power dissipation and the stronger the influence of the reduction in switching activity ( $\lim_{L \rightarrow \infty} E_P = E_\alpha$ ) and vice versa. Therefore, a good metric to compare different encoding schemes is the length at which the energy efficiency  $E_P$  is zero. From that length coding starts to pay off. We will call this length the *effective length*  $L_{eff}$ , which can be derived from equation (4) by expanding the bus power consumption as previously mentioned:

$$L_{eff} = \frac{P_{Codec}}{\frac{1}{2} V_{dd}^2 f \left( \frac{C_{bus,unc}}{|mm|} \alpha_{unc} - \frac{C_{bus,cod}}{|mm|} \alpha_{cod} \right)}. \quad (5)$$

The capacitance of the coded ( $C_{bus,cod}$ ) and uncoded bus ( $C_{bus,unc}$ ) can differ due to additional wires required by several schemes. Using the effective length, the efficacy of the codecs in saving power can be compared, incorporating their different encoding efficiencies, their power dissipation and possible extra signaling wires.

### 4. Encoding Efficiency and Power Dissipation of Bus Encoding Schemes

For our investigations we choose a variety of general purpose encoding schemes which do not require a priori knowledge of the data streams. We selected static and adaptive techniques. A well known static scheme with a very simple codec implementation is the Businvert (BI) encoding scheme. APBI, an adaptive scheme, represents schemes which extend BI. In order to demonstrate the impact of optimizing codecs for internal power dissipation we also implemented APBI with extended mask update periods (mup) [6]

from every window ( $APBI_{mup0}$ ) to every second window ( $APBI_{mup1}$ ) up to every 16th window ( $APBI_{mup4}$ ). The logic of the adaptive block is switched off during inactive periods. As a different line-oriented adaptive technique IAEB [1] was selected. AMWC4, a power-optimized variation of AMWC, represents the schemes which compute the encoding rule on word-level basis.

The encoding schemes will perform differently on different data streams since they exploit various characteristics of the data. To obtain a fair comparison, we defined a set of test data streams:

- **art**: A random, segmented data stream with a varying distribution of activity in every segment.
- **eps**: An ascii file in encapsulated postscript format.
- **gzip**: gzip binary (example for an executable file).
- **ppm**: A composed PPM image stream consisting of 4 different images with varying statistics.
- **gauss**: White Gaussian noise.
- **dct<sub>1,2</sub>**: Recorded at the output of the DCT stage of an image processing application.

The encoding efficiency  $E_\alpha$  of the coding schemes is plotted in Fig. 1. The figures were obtained by applying equa-

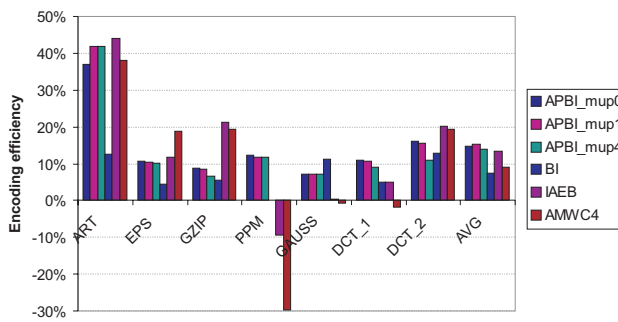


Figure 1. Encoding efficiency

tion (2) on the activity values measured on an uncoded and coded bus. Transitions on extra bus wires required by some encoding schemes (e.g. *BI*, *APBI*), are included in the activity of the coded bus. *AVG* indicates the performance, averaged over all data streams.

As expected the best encoding efficiency is achieved by the adaptive schemes for the *art* data stream while *BI* can not be outperformed for data streams with an over bus lines equally distributed activity (*gauss*). On average (*avg*), adaptive encoding schemes achieve a higher encoding efficiency  $E_\alpha$  than static schemes.

The power dissipation of the codecs for the different data streams form a threshold for the total power savings. Therefore, the same data streams were used to simulatively observe the power consumption of the codecs. The codec circuitries were implemented as a pure digital circuit for a data width of 32 bit and synthesized for a 0.13  $\mu\text{m}$  CMOS technology. For all schemes an optimized layout of coder

and decoder was produced. Netlists (including parasitics) were extracted from these layouts, and the power dissipation of the codecs was determined by power simulations using the in-house power estimation tool “DIESEL”. Since the power consumption depends on the frequency, we expressed the power consumption of the codecs for different data streams in  $\mu\text{W}/\text{MHz}$ , and plotted these values in Fig. 2. The graph shows that the schemes consume a differ-

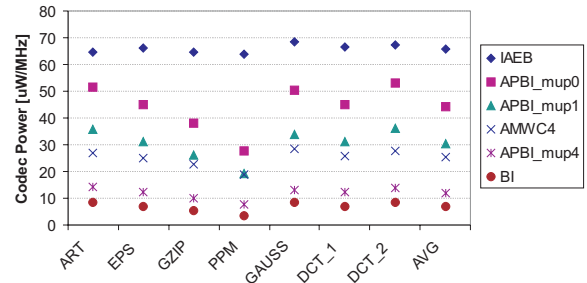


Figure 2. Power consumption

ent amount of power corresponding to the active hardware overhead. Furthermore the power dissipation differs while coding the different data streams which is proportional to the average activity of the stream. Comparing Fig. 1 and 2 shows in most of the cases the relation: the higher the encoding efficiency the more power is dissipated within the codec system.

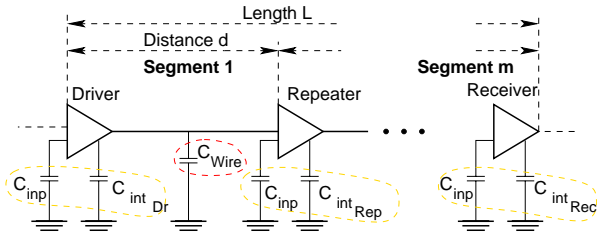
In modern embedded systems, a variety of data is transmitted over the system buses. For fair and simple comparison we therefore use the average encoding efficiency of a codec, and the average power consumption.

## 5. Analytical Model for the Bus

Although the encoding efficiency determines the savings, we need the actual power dissipation of the bus to evaluate whether the coding schemes are effective, and to calculate the effective length. To cover buses of different lengths, with different repeater insertion schemes and with different number of wires, we developed a parameterizable analytical model for the power dissipation on the bus. The model was validated by a power simulation with a lay-out (including parasitics) of a particular bus implementation.

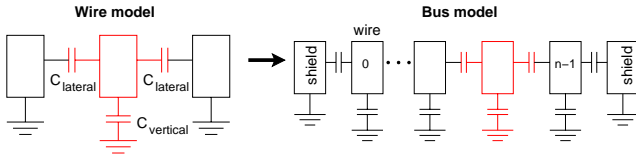
The model of a single bus line is depicted in Fig. 3. The line capacitance  $C_{line}$  covers the bottom and coupling capacitances of the *wires* ( $C_{wire}$ ), as well as internal ( $C_{int}$ ) and input ( $C_{inp}$ ) capacitances of *active elements* such as driver, receiver and potential repeaters.

In deep submicron technologies, the wire capacitance (Fig.4) is not only determined by the vertical capacitance  $C_{vertical}$  but increasingly by the coupling capacitances ( $C_{lateral}$ ) to adjacent wires [4]. Due to the miller effect



**Figure 3. Model for a bus line including driver, repeaters and receiver**

the power dissipation becomes data-dependent. While the coupling capacitance is not charged if neighbouring wires switch in the same direction, twice  $C_{lateral}$  is charged for opposite switches. However, assuming the activity to be independently distributed over bus lines, adjacent lines will switch as often in the same direction as in the opposite direction. On average the coupling capacitance between lines is charged once. Therefore we assume the two adjacent wires to be connected to ground. Each transition from “0” to “1” on the middle wire causes the vertical capacitance and both of the coupling capacitances ( $C_{lateral}$ ) to be charged. The wire capacitance per unit length can then be calculated by:  $C_{wire} = C_{vertical} + 2C_{lateral}$ .



**Figure 4. Model for a bus wire with 2 adjacent wires and extension to the  $n$ -wire bus model**

The capacitance of an  $n$ -bit wide bus, assuming identical drivers and repeaters, can be calculated by Equation (6):

$$C_{bus} = nL(C_{vertical} + 2C_{lateral}) + n\left(\frac{L}{d}C_{int,Dr} + \frac{L}{d}C_{inp,Dr} + C_{int,Rec} + C_{inp,Rec}\right) \quad (6)$$

where  $L$  is the bus length and  $d$  the repeater distance.

Equation (6) is used to calculate the average capacitance of a bus line per unit length based on the values of a  $0.13 \mu\text{m}$  CMOS technology. The wire capacitance  $C_{wire}$  depends on the spacing to adjacent wires and the distance to the metal layers above and below. Therefore we quantify the upper ( $C_{max}$ ) and lower ( $C_{min}$ ) bound of line capacitance to span the whole range of power consumption. The lower bound is given by combining the capacitance for large spacing with a very relaxed repeater insertion scheme: 1 repeater every 5 mm ( $C_{line,min} = 116 \text{fF}/\text{mm}$ ). Likewise, the capacitance for minimum spacing is combined

with a very aggressive repeater scheme (repeater spacing = 1 mm) which provides the upper bound of the capacitance ( $C_{line,max} = 349 \text{fF}/\text{mm}$ ). As a typical example we selected a bus with double width and double spacing and a repeater distance of 2 mm ( $C_{line,typ} = 238 \text{fF}/\text{mm}$ ).

## 6. Results

### 6.1. Effective Length and Energy Efficiency

According to Equation (5) the effective lengths of the encoding schemes for upper and lower bound as well as for the typical line capacitance, each combined with the average encoding efficiency have been calculated. The results are given in Tab. 1.

**Table 1. Effective length in mm for 32 bit Codecs in  $0.13 \mu\text{m}$  Technology**

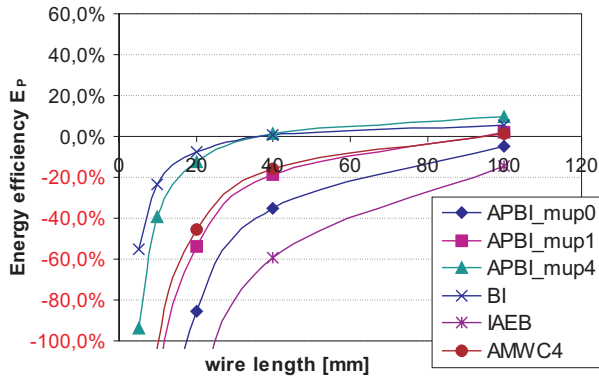
	APBI		BI	IAEB [1]	AMWC <sub>4</sub>	
	$mu0$	$mu1$	$mu4$			
$C_{min}$	264	174	74	76	391	180
$C_{typ}$	130	86	36	37	192	88
$C_{max}$	89	59	25	26	131	61

The lowest effective lengths are obtained for the upper bound ( $C_{max}$ ). The minimum length, 25 mm, is achieved for APBI with the largest mask update interval (APBI $_{mu4}$ ), for which the adaptive parts of the codec logic are switched on only every 16th windows. Striking is that the second lowest effective length (26 mm) is obtained by Bus Invert coding (BI). This encoding scheme requires only 1/11th of the hardware overhead of APBI $_{mu4}$  and is half as efficient, but has almost the same effective length.

The values listed in Tab. 1 indicate that the buses have to be quite long before break-even. For the upper bound capacitances, the length varies between 25 and 131 mm, whereas for the typical case the effective length varies between 36 and 192 mm. Furthermore, we would like to recapitulate that the effective length was defined as the length at which the energy efficiency was 0. To save energy, the buses have to be even longer. It is very unlikely that buses of these lengths will be used on chip, since typical circuits in  $0.13 \mu\text{m}$  technology reach extensions up to 20 mm (according to the ITRS Roadmap).

By plotting the energy efficiency ( $E_p$ ) as a function of bus length, we can obtain a good insight between the efficiency of the coding schemes for different capacitive loads which are proportional to the length.

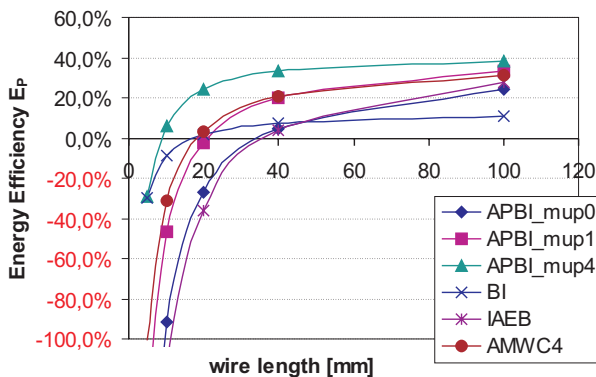
The average energy efficiency for the selected schemes, based on the average encoding efficiency  $E_\alpha$  achieved on a bus with typical wire capacitances are plotted in Fig. 5. Due to the power consumed within the codec system, for



**Figure 5. Average energy efficiency, for typical wires and average  $E_\alpha$**

bus lines shorter than the effective length ( $E_P < 0$ ) the total power dissipation of the system is higher than in the uncoded case. After achieving the break-even length, power is saved. The starting points of the curves are directly related to the power dissipation of the codec systems. The higher portion of adaptive schemes is indicated by the higher offset. Due to the power dissipated in codec systems even for on-chip bus lengths of 100 mm the power savings are less than 10%.

To obtain an upper bound for the energy efficiency of transition minimizing bus encoding schemes we investigated the highest encoding efficiency achievable by the encoding schemes. The upper bound which is depicted in Fig. 6 is yielded with the best encoding efficiency  $E_\alpha$  on a bus with maximum capacitance. The best  $E_\alpha$  is achieved for all schemes by encoding the *art* data stream. Only for BI it is *dct2*. It can be observed that the slope of the effi-



**Figure 6. Energy efficiency - upper bound (maximum wire capacitance, best  $E_\alpha$ )**

ciency curves are steeper and the schemes break-even much

sooner. Corresponding to the higher encoding efficiency the saturation points of adaptive schemes are higher. However, Fig. 6 indicates that it is even for best-case reduction very difficult to reduce the power dissipation of the bus by means of coding. The shortest effective length is still 9 mm, and reached only by one adaptive codec (APBI<sub>mup4</sub>). In order to save approximately 20% in power for this codec a bus length of about 15 mm is necessary. Noticeable too is that the best case BI performs relatively poor.

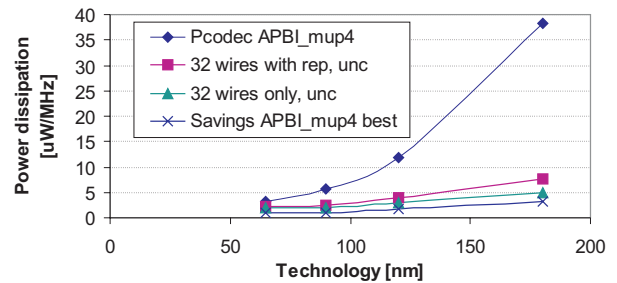
The final efficiencies differ corresponding to the coding efficiency  $E_\alpha$ , since the codec power is negligible for extremely long wires. For the adaptive schemes, close to 40% savings can be obtained for very long bus lengths. This opens up the opportunity of these techniques for highly capacitive loads such as off-chip buses since the capacitances involved are much higher, and quite often the voltage swings are larger as well.

Figures 5 and 6 indicate that the energy efficiency can be improved by increasing the encoding efficiency and/or decreasing the power dissipated in coder-decoder system.

## 6.2. Impact of Technology-Scaling

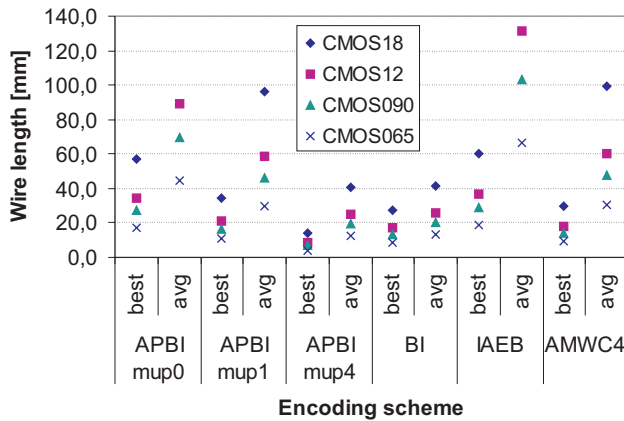
Since transition-minimizing bus encoding schemes implemented in current technologies ( $0.13 \mu\text{m}$  CMOS) do not pay off for on-chip buses we explored how the energy efficiency is influenced by scaling to the next two technology generations.

Internal capacitances as well as supply voltages continuously decrease due to scaling. As a result, the dynamic power consumption of the logic will decrease assuming a stable frequency. In contrast to that the capacitance of interconnects will continuously increase due to the reduced spacing between neighbouring wires. Nevertheless, the power consumed on wires slightly decreases due to the reduced supply voltage. Fig. 7 depicts the codec and bus power dissipation based on technology information.



**Figure 7. Scaling of power dissipation**

Since the power dissipation of the logic is reduced much stronger than the portion consumed by wires, the effective length for the encoding schemes will also decrease as depicted in Fig. 8. Transition-reducing bus encoding schemes



**Figure 8. Scaling of effective wire length**

break-even then much sooner: at 4 mm for best-case encoding efficiency and at 13 mm wire length for average reduction in activity. However, it is very unlikely that such bus lengths will be achieved since also the dimensions of the communicating modules will scale accordingly.

## 7. Conclusions

As our investigations show, transition minimizing bus encoding techniques do not pay off for on-chip buses. The length of the on-chip bus to break even on the power consumption of the codec is already longer than what can be expected for typical designs. The length to obtain significant power savings is even longer. Although technology scaling reduces this minimum length, the required lengths are still longer than what can be expected to be a realistic bus-length for those technologies. Only in some very specific point-to-point connections, where the best case coding efficiency can be achieved, the length of the bus to save a significant amount of power fits within the size of a reasonable chip.

Furthermore, it has been shown that adaptive schemes outperform static schemes in encoding efficiency and effective length. Our experiments confirmed that methods which reduce internal power consumption as presented for APBI<sub>mup4</sub> efficiently improve the energy efficiency. For high capacitive loads the maximum power savings are determined by the encoding efficiency. Therefore adaptive schemes achieve higher savings after break-even due to the higher encoding efficiency. Since (dedicated) off-chip interconnections have a much higher capacitive load, these maximum power savings seem reachable for these interconnections. For this case, technology scaling, reduces the cost of silicon real estate, and reduces the latency penalty. This favors the application of adaptive transition minimising coding for off-chip interconnect.

## 8. Acknowledgements

The authors would like to thank all members of DD&T at Philips Research, Eindhoven for the excellent cooperation and the helpful and fruitful discussions. Special thanks to A. Vaassen, M. Meijer, S. van Dijk, H. Veendrick, L. Sevati and P. Soulard.

## References

- [1] L. Benini, A. Macii, E. Macii, M. Poncino, and R. Scarsi. Synthesis of Low-Overhead Interfaces for Power-Efficient Communication over Wide Buses. In *36th Design Automation Conference DAC*, 1999.
- [2] L. Benini, G. Micheli, E. Macii, M. Poncino, and S. Quer. System-Level Power Optimization of Special Purpose Applications: The Beach Solution. In *Int'l Symposium on Low Power Electronic Design ISLPED*, pages 24–29, 1997.
- [3] L. Benini, G. Micheli, E. Macii, D. Sciuto, and C. Silvano. Address Bus Encoding Techniques for System-Level Power Optimization. In *Design Automation and Test in Europe DATE*, 1998.
- [4] R. Ho, K. W. Mai, and M. A. Horowitz. The Future of Wires. *Proceedings of the IEEE*, 89(4):490–504, April 2001.
- [5] K.-W. Kim, K.-H. Baek, N. Shanbhag, C. Liu, and S.-M. Kang. Coupling-driven signal encoding scheme for low-power interface design. In *ICCAD*, pages 318 – 321, 2000.
- [6] C. Kretzschmar, R. Siegmund, and D. Mueller. Adaptive Bus Encoding Technique for Switching Activity Reduced Data Transfer over Wide System Buses. In *Workshop on Power and Timing Modeling, Optimization and Simulation PATMOS*, pages 66–75, Goettingen, Germany, September 2000. Springer.
- [7] C. Kretzschmar, R. Siegmund, and D. Mueller. A Low Overhead Auto-optimizing Bus Encoding Scheme for Low Power Data Transmission. In *Int'l Workshop on Power and Timing Modeling, Optimization and Simulation, PATMOS*, pages 342–352, Sevilla, Spain, September 2002. Springer.
- [8] Satoshi Komatsu, Makoto Ikeda, and Kunihiro Asada. Bus Data Encoding with Adaptive Code-book Method for Low Power IP Based Design. In *International Workshop on IP based design and Synthesis*, pages 77–81, December 2000.
- [9] Y. Shin, S. Chae, and K. Choi. Partial Bus-Invert Coding for Power Optimization of System Level Bus. In *Int'l Symposium on Low Power Electronic Design ISLPED*, pages 127–129, 1998.
- [10] M. Stan and W. Burleson. Bus-Invert Coding for Low-Power I/O. In *Transactions on VLSI Systems*, volume 3, pages 49–58, March 1995.
- [11] D. Sylvester and K. Keutzer. Getting to the Bottom of Deep Submicron. In *ICCAD*, 1998.
- [12] D. Tamura, B. Pangrle, and R. Maheshwary. Techniques for energy-efficient SoC design. *EEdesign*, July 2003.
- [13] X.L.Su, X.Y.Tsui, and A. Despain. Saving Power in the Control Path of Embedded Processors. *IEEE Design and Test of Computers*, 11:24–30, 1994.
- [14] Y. Zhang, J. Lach, K. Skadron, and M. Stan. Odd/even bus invert with two-phase transfer for buses with coupling. In *ISLPED*, pages 80 – 83. ACM, 2002.