A Survey of Techniques for Energy Efficient On-Chip Communication

Vijay Raghunathan[†], Mani B. Srivastava[†], and Rajesh K. Gupta[‡]
[†]Department of Electrical Engineering, UC Los Angeles, CA 90095
[‡]Department of Computer Science and Engineering, UC San Diego, CA 92093

ABSTRACT

Interconnects have been shown to be a dominant source of energy consumption in modern day System-on-Chip (SoC) designs. With a large (and growing) number of electronic systems being designed with battery considerations in mind, minimizing the energy consumed in on-chip interconnects becomes crucial. Further, the use of nanometer technologies is making it increasingly important to consider reliability issues during the design of SoC communication architectures. Continued supply voltage scaling has led to decreased noise margins, making interconnects more susceptible to noise sources such as crosstalk, power supply noise, radiation induced defects, *etc*. The resulting transient faults cause the interconnect to behave as an unreliable transport medium for data signals. Therefore, fault tolerant communication mechanisms, such as Automatic Repeat Request (ARQ), Forward Error Correction (FEC), *etc.*, which have been widely used in the networking community, are likely to percolate to the SoC domain.

This paper presents a survey of techniques for energy efficient on-chip communication. Techniques operating at different levels of the communication design hierarchy are described, including circuit-level techniques, such as low voltage signaling, architecture-level techniques, such as communication architecture selection and bus isolation, system-level techniques, such as communication based power management and dynamic voltage scaling for interconnects, and network-level techniques, such as error resilient encoding for packetized on-chip communication. Emerging technologies, such as Code Division Multiple Access (CDMA) based buses, and wireless interconnects are also surveyed.

Categories and Subject Descriptors

B.4.3 [**Input/Output and Data Communications**]: Interconnections (Subsystems); C.5.4 [**Computer System Implementation**]: VLSI Systems

General Terms

Design, Performance, Reliability

Keywords

System-on-Chip Design, Low Power Design, Energy Efficient Design, Communication Architectures, Power Management

DAC 2003, June 2–6, 2003, Anaheim, California, USA.

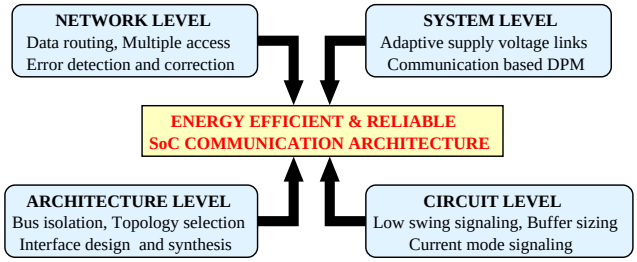

Figure 1: SoC interconnect architecture optimization involves all layers of the design hierarchy

1. INTRODUCTION

Interconnect wires account for a significant fraction (up to 50% [1]) of the energy consumed in an integrated circuit, and this fraction is only expected to grow in future. In fact, it is projected that, as technology scales to the nanometer regime, the delay and energy consumption of global interconnect structures will prove to be a major bottleneck for SoC design [2, 3]. With designers striving to improve the lifetime of battery operated, personal computing devices [4], minimizing the energy consumed in on-chip interconnects becomes crucial. However, doing so requires the use of a structured, interconnect oriented, design methodology at all layers of the hierarchy from the system-level down to physical design.

Interconnect oriented design poses a whole new set of challenges for designers. The advent of deep sub-micron technology also brings several signal integrity issues to the forefront [5, 6]. Supply voltage scaling, used extensively to reduce energy consumption, has led to decreased noise margins, making interconnects less immune to the vagaries of power supply noise, inter-wire crosstalk, radiation induced soft errors, electromagnetic interference, *etc*. With next generation SoCs expected to have multi-billion transistors, ensuring efficient and reliable transport for data signals becomes a daunting task indeed. Existing low power system design methodologies are ill-equipped to tackle this problem because they follow an *error avoidance* paradigm, where the aim is to eliminate all noise related errors through fine-tuned circuit design. Clearly, this will become very costly, if not impossible, given the complexity of future SoCs. Therefore, the paradigm has to change to one of *error tolerance* in which, after a moderate amount of design time optimization, the focus is on error resilience techniques to combat run time errors and ensure correct system functionality.

We believe that designing a high-performance, reliable, and energy efficient SoC communication architecture will require a judicious interplay of error-avoidance and error-tolerance techniques spanning all levels of the design hierarchy, as shown in Figure 1. This paper presents a survey of several such techniques that can

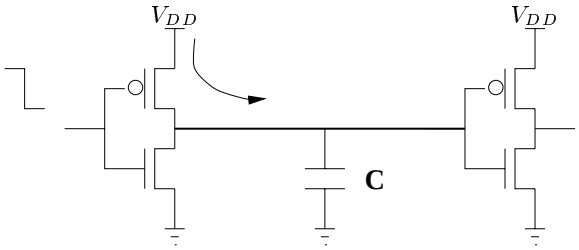

Figure 2: Simple two inverter signaling configuration

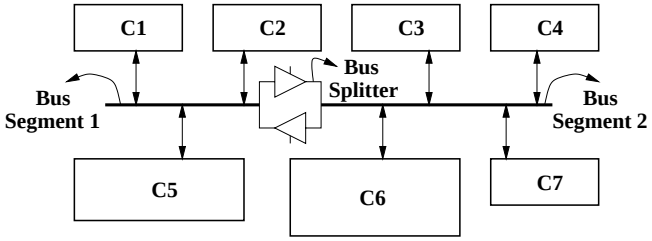

Figure 3: A split bus communication architecture

be used to design next generation SoC communication architectures. The remainder of this paper is structured as follows. We start off by describing circuit-level optimizations in Section 2. Section 3 describes architecture-level techniques, such as communication architecture selection, and on-chip networks. Section 4 describes system-level optimizations, including the recently proposed adaptive supply voltage interconnects. Section 5 deals with network-level issues for packetized on-chip communication, including dynamic error resilience techniques. Section 6 presents two novel communication architectures for SoC design, based on Code Division Multiple Access (CDMA) buses, and Radio Frequency (RF) interconnects. Section 7 presents the conclusions.

2. CIRCUIT LEVEL TECHNIQUES

The most widely used form of signaling is the classical two-inverter configuration with rail-to-rail signal swing [7]. This configuration is illustrated in Figure 2. As shown in the figure, both the transmitter and receiver are CMOS inverters. Intermediate repeaters are often used to improve signal characteristics [8]. Such a full swing signaling scheme is inefficient in terms of energy consumption. The energy consumption of an interconnect wire in one clock cycle is given by [9]:

$$E_{wire} = \alpha \times C \times V_{DD} \times V_{swing} \quad (1)$$

where α is the switching activity (per clock cycle) of the signal that is being transmitted, C is the physical capacitance switched during signal transitions, V_{DD} is the supply voltage, and V_{swing} is the voltage swing across the wire.

Since the two inverter configuration uses a voltage swing of V_{DD}, lowering the supply voltage leads to a quadratic reduction in the energy. In addition, reducing the voltage swing also results in lower wire delays [22]. Several low swing signaling techniques are proposed in [10]. However, low swing sensing decreases the noise margin, which is a concern in low supply voltages (below 1.8V). Circuit reliability can be improved through the use of complex receiver circuitry [5] that, in turn, has an adverse effect on power consumption. An alternative signaling scheme is differential signaling [5], in which information is conveyed as the difference in voltage between two signal wires. Differential signaling is attractive due to its high common-mode noise rejection, allowing for a further reduction in the signal swing. However, differential signaling has a high overhead due to additional interconnect and its routing. To ammortize this overhead, pseudo-differential (PD) signaling schemes have been proposed, where the reference signal is shared among multiple bus lines and receivers. The low voltage differential signaling (LVDS) standard [11] provides designers with guidelines on optimizing interconnect energy consumption through the use of differential signaling techniques. Finally, the use of current-mode signaling has also been explored, and holds promise, for global on-chip interconnects [12].

3. COMMUNICATION ARCHITECTURE SELECTION

The choice of system-level communication architecture has a significant impact on the system performance and energy consumption. A commonly used SoC communication architecture is a shared bus, due to its simple topology, low area cost, and extensibility. However, a monolithic shared bus is often inefficient from an energy viewpoint, because, (i) the load capacitance of the entire bus has to be driven during each data transfer, and (ii) since all the traffic shares a single path, only one component pair can communicate at any given time, requiring high frequency bus operation to achieve the required throughput. To alleviate these problems, several alternative communication architectures have been developed.

3.1 Bus splitting

Bus splitting or bus isolation, analyzed in [15], involves splitting a monolithic bus into multiple segments, with dual port drivers connecting adjacent segments, as shown in Figure 3. The dual port drivers, when enabled, relay the data from one bus segment to the other, thus retaining the functionality of the monolithic bus. Both AMBA [13] and IBM CoreConnect [14], two widely used on-chip buses, have a split bus architecture, with a low-speed peripheral bus attached to the main high-speed system bus through a bridge.

A split bus architecture has obvious energy-related circuit advantages such as reduced capacitive load during bus transfers. It was reported in [15] that the use of a split bus architecture yields energy savings of 16%-50% over a monolithic bus, depending on the traffic characteristics, and configuration of the split bus. More importantly, a split bus offers several architectural advantages, such as increased concurrency and energy efficiency during bus transactions. Data transfers can proceed in parallel in different bus segments due to the isolation provided by the bridges. Although the basic idea of bus splitting is well known, there are several interesting design issues involved, which merit further investigation. For example, the split could be vertical (*i.e.,* across the width of the bus) as shown in Figure 3, or could even be horizontal (*i.e.,* along the length of the bus), in which case a single bus gets split into multiple smaller bitwidth buses. The decision of whether, and how often, to split a bus is strongly dependent on the energy overhead of the drivers, and the logic and wires required to generate and propagate the enable signals. Therefore, techniques that provide architectural insights to guide the bus splitting, and tools that automate the process, also need to be developed to facilitate the adoption of bus splitting into standard SoC design flows. Finally, the effect of floor planning decisions on the system energy consumption is evident from the above discussion. Restricting high traffic communication to a single bus segment, by appropriately placing the components involved in such communication, leads to improvements in both the throughput and energy consumption of the communication architecture.

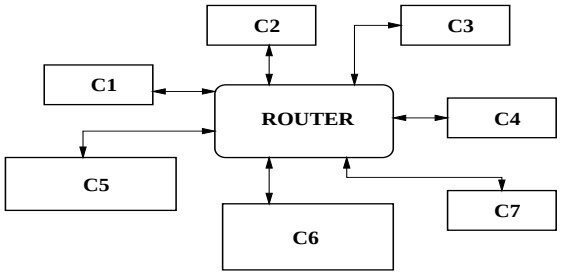

Figure 4: A router based communication architecture

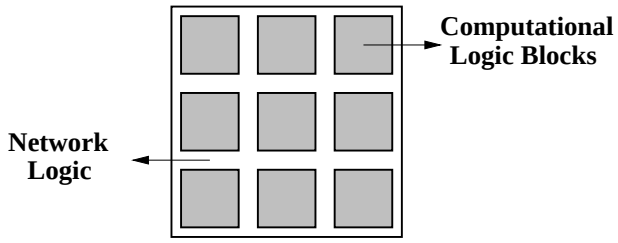

Figure 5: A tile based on-chip interconnection network

3.2 Router based communication architectures

An alternative approach is to use a router based on-chip interconnect architecture, as shown in Figure 4. Such an approach has been adopted at the board level for interconnecting components in a wireless multimedia node [16]. The router could, in theory, be a fully connected crossbar, though it may be optimized for reduced complexity by exploiting the fact that not every component on the chip will need to talk to every other component. Further, optimizations such as those presented in [17] can be used to reduce latency, and improve crossbar utilization. Similar to a segmented bus, a router based architecture has several energy-related advantages:

- The isolation of components from each other leads to reduced capacitive loading during data transfers, which results in lower energy consumption.
- The increased parallelism enabled by the router results in a higher throughput, which can be traded off for additional energy savings through the use of frequency/supply voltage scaling, or other power-performance trade off techniques.

A router based architecture also provides flexibility in design composition, which is of increasing importance in upcoming platform-based designs composed of intellectual property cores. It was shown in [16] that the energy benefits of a router based communication architecture increase as the number of components present in the system increases. With next generation SoCs expected to have a large number of components [18], adopting a router based communication architecture can potentially lead to significant energy savings. Researchers have just begun to investigate the merits/demerits of such an approach by analyzing and modeling the power consumption of routers and switches [19, 20]. Efforts are also underway to develop modeling and simulation frameworks for on-chip communication architectures [21]. Such a framework will enable SoC designers to quickly and efficiently explore the communication architecture design space.

3.3 Networks on chip

The work in [22] advocates replacing the global wiring on a chip with a general purpose interconnection network. It is argued that such an approach structures the top level wires in a chip, and facilitates modular design. Structured wiring results in predictable electrical parameters, such as crosstalk, *etc*. In addition, a regular global wiring structure will also enable the use of advanced performance optimization techniques, such as wavefront pipelining [23]. Figure 5 shows an on-chip network, with computational logic partitioned and placed into regular tiles. The tiles interface to the network logic, and communicate by sending packets to each other.

Building on the concept of on-chip networks, [24] proposes a three layer protocol stack model for packetized inter-component data communication. The authors' viewpoint is that the complexity of next generation SoCs, combined with noise effects, will result in system operation not being fully deterministic. As a result, SoC design will require the use of stochastic models and tools, as well as probabilistic metrics of performance, such as those applied extensively in the field of computer networking. Based on a similar argument, the authors of [25] propose to use the standard seven layer ISO/OSI protocol model for on-chip networks. However, it is still unclear if such a full blown protocol stack will be practical, since the latency for transferring information down the transmitter stack, through the channel, and up the receiver stack might be unacceptably high. Recent research has also focused on optimizing the design of application specific on-chip networks by analyzing and exploiting application-level traffic patterns [26]. Finally, note that a large class of on-chip networks, namely direct networks, involve a router based communication architecture, described in Section 3.2.

4. SYSTEM LEVEL TECHNIQUES

4.1 Communication based power management

The communication architecture plays an important role in system-level power management as well. The authors of [27] introduce the concept of *communication architecture based power management* (CBPM), where the communication architecture regulates the execution of system components to yield a battery friendly system-level current (and power) profile. The rationale behind CBPM is threefold, (i) integrating the system-level power management functionality into the communication architecture, which binds the system components together, eliminates the need for a separate power management entity, (ii) due to its connectivity, the communication architecture can gather information (such as the execution states of system components) required to make power management decisions, and also deliver these decisions to the various system components, and (iii) since the communication architecture schedules inter-component communications, it can control the timing of a component's power modes, thus regulating the component's (and therefore, the system's) power profile. In [27], CBPM policies were used to design a battery efficient IEEE 802.11 medium access control (MAC) processor. It was reported that the use of CBPM yields a performance-battery life tradeoff (*e.g.*, a gain of 3.2X in battery life with a performance degradation of 2.3X) that is superior to what can be achieved using conventional power management techniques, such as frequency scaling.

4.2 Adaptive supply voltage links

Adaptive supply regulation, also called dynamic voltage scaling (DVS), has proven to be an effective technique for energy optimization of microprocessors. DVS schemes dynamically adjust the processor clock frequency and supply voltage to just meet instantaneous performance requirement, making the system *energy aware*. In [28], the same concept is applied to serial interconnect links. It

is shown that the power consumption of a link varies by 10X over the range of link frequencies (and corresponding supply voltages).

Based on the adaptive supply link developed in [28], the authors of [29] present a DVS technique for energy efficient interconnection networks. The motivation for the work is that, much like microprocessors, communication architectures too display a wide variance in their utilization, depending on the communication patterns of applications [31]. The DVS algorithm presented in [29] is a variant of previously proposed history based DVS techniques for processors, and adapts the link's frequency and supply voltage in accordance with the instantaneous traffic bandwidth. The authors report power savings of 3.2X on average, with a slight degradation in throughput and latency, compared to fixed supply links. However, the effect that supply voltage reduction has on the link's reliability is not considered in [29] since it is assumed that the interconnect link operates with negligible bit error rates over a wide range of frequencies and supply voltages. While this is often true for inter-chip interconnects due to the use of advanced signaling techniques, it is unlikely to be the case for on-chip interconnect wires. As will be seen in Section 5, reducing the supply voltage of a link increases the bit error rate. Therefore, DVS schemes, in the case of on-chip interconnects, should attempt to maintain the system's fidelity through techniques such as error resilient encoding. Further, combined speed scaling and shutdown techniques for computation systems, such as [30], also impact the timing of system-level communication transactions as well as when architectural components are scheduled for shutdown. Going forward, these algorithms will need to account for the effect of voltage scaling on channel reliability as well. Finally, the practical use of adaptive supply interconnects is likely to be dictated by the feasibility of generating and distributing multiple variable supply voltages on-chip.

5. NETWORK LEVEL ISSUES

The techniques presented thus far do not explicitly account for the effect of noise on the interconnect reliability. Since a noisy interconnect behaves as an unreliable transport medium, and introduces errors in the transmitted signals, the communication process needs to be fault tolerant to ensure correct information transfer. This is done through the use of channel coding [32]. Channel coding schemes introduce a controlled amount of redundancy in the transmitted data, increasing its noise immunity. Next, we describe the impact of error coding schemes on the interconnect reliability and energy consumption.

5.1 Modeling interconnect reliability

An interconnect wire, at a high level of abstraction, can be modeled as a noisy communication channel over which bit streams are transmitted. The error performance of such a channel has been analyzed in [33], and improved in [34]. To understand the interdependence of the error performance of an interconnect link on the supply voltage, we make the following assumptions (as in [33]) about the channel characteristics:

- The channel is memoryless, *i.e.*, the noise process affecting a given bit is independent of that affecting preceding or succeeding bits.
- All the noise sources collectively induce a noise voltage, V_N, on the channel, which follows a Gaussian distribution[1] with zero mean, and variance σ_N^2. This assumption has been shown to be valid for off-chip I/O links [35].

[1] As described in [5], correlated noise sources should be summed up using a worst case analysis before being bundled with the remaining uncorrelated noise sources as a single Gaussian source.

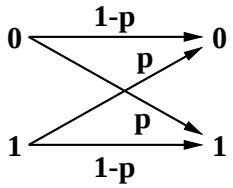

Figure 6: A binary symmetric channel

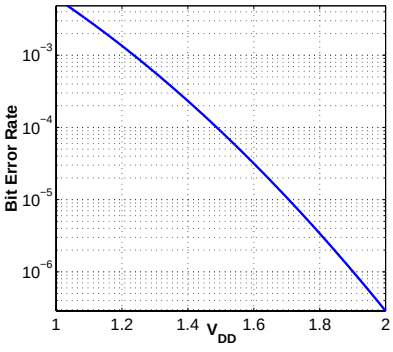

Figure 7: Bit error rate versus supply voltage for $\sigma_N = 0.2V$

- The logic switching threshold of a CMOS inverter is equal to $\frac{V_{DD}}{2}$, where V_{DD} is the supply voltage. If the noise distorts the transmitted signal by more than the logic threshold, an error occurs.

The first and last bullets above imply that the channel is a **Binary Symmetric Channel (BSC)** [32]. In a BSC, the probability that a bit will be received incorrectly is independent of the actual bit value itself, as illustrated in Figure 6. The parameter, *p*, is the probability that a given bit arrives flipped at the receiver, and hence, represents the Bit Error Rate (BER) of the channel. The probability that the noise voltage is V, is given by the Gaussian probability density function:

$$p(V) = \frac{1}{\sigma_N \times \sqrt{2\pi}} e^{\frac{-V^2}{2\sigma_N^2}} \tag{2}$$

Therefore, the BER, which is the probability that the noise voltage exceeds the logic threshold, is given by:

$$BER = \int_{\frac{V_{DD}}{2}}^{\infty} p(V)\, dV \tag{3}$$

$$= \frac{1}{\sigma_N \times \sqrt{2\pi}} \int_{\frac{V_{DD}}{2}}^{\infty} e^{\frac{-V^2}{2\sigma_N^2}}\, dV \tag{4}$$

$$= \frac{1}{\sqrt{2\pi}} \int_{\frac{V_{DD}}{2\sigma_N}}^{\infty} e^{\frac{-V^2}{2}}\, dV \tag{5}$$

$$= Q\left(\frac{V_{DD}}{2 \cdot \sigma_N}\right) \tag{6}$$

where $Q(\cdot)$ is the Gaussian tail function given by:

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{\frac{-u^2}{2}}\, du \tag{7}$$

Equation (6) defines the dependence of channel reliability (in terms of BER) on the supply voltage. This dependence is shown in Figure 7. As the supply voltage decreases, the BER increases, and hence the reliability of the channel decreases. It is clear that any future advances in DVS will need to account for this dependence.

5.2 Error correction versus Retransmission

Linear block codes [32] are commonly used for channel coding. Using an (n, k) linear block code, a data block, k bits long, is mapped onto an n bit code word, which is transmitted on the channel. The receiver examines the received signal, and declares an error if it is not a valid code word.

Once an error has been detected, it can be handled in one of two ways, (i) Forward Error Correction (FEC), where the properties of the code are used to correct the error, if possible, or (ii) Retransmission, also called Automatic Repeat Request (ARQ), where the receiver asks the sender to retransmit the codeword that was in error. FEC schemes require a more complex decoder[2], while ARQ schemes require the existence of a reverse (or feedback) channel from the receiver to the transmitter. The tradeoffs between the two techniques, as well as a hybrid FEC/ARQ scheme, were first explored in [36], for wireless communications. In the context of on-chip communication, the energy efficiency of FEC based, as well as ARQ based, schemes was studied in [37]. Several variants of Hamming codes [32] (which are a subset of linear block codes) were implemented in the VHDL model of a SPARC V8 compliant embedded processor, and retransmissions were carried out in a manner compliant to the AMBA on-chip bus standard [13]. Trace driven simulations showed that, for the same constraint on system reliability, ARQ schemes enable the use of a lower supply voltage compared to FEC schemes. Further, the decoder in FEC schemes incurs a large energy overhead. As a result, ARQ schemes are more energy efficient than FEC schemes, inspite of the higher number of bits transferred. However, it is expected that as technology scales, the ratio of computation cost to on-chip communication cost will decrease, bridging the gap between the two strategies.

5.3 Protocol considerations

The adoption of packetized on-chip communication makes the organization of the communication protocol stack extremely crucial. The stack should be lightweight to ensure negligible performance and energy overheads during communication. We envision the use of a simple three-layer protocol stack, comprising of physical, data-link, and network layers. While the physical layer would implement techniques such as voltage scaling, the data-link layer would implement medium access control (MAC) and error control techniques and its related decisions. An example of a MAC protocol is the TDMA based two-level arbitration protocol used in the Sonics micronetwork backplane [38]. Data routing algorithms, implemented at the network layer, establish the path a message follows through the network to its final destination. Several energy efficient routing algorithms have been developed for conventional networks (especially wireless) [39], which, in addition to minimizing communication energy, also balance out the spatial distribution of energy consumption in the network. In the context of SoCs, such a routing algorithm will also help avoid local hot spots, thereby simplifying SoC thermal management. Researchers have just started to address the issue of energy efficient data routing in tile based on-chip networks [40, 41].

6. EMERGING TECHNOLOGIES

As the importance of a well designed communication architecture to SoC performance and energy consumption is understood, on-chip communication architecture design is attracting an increasing amount of research attention [42]. Next, we discuss two emerging technologies, which present interesting alternatives to conventional SoC communication architectures.

[2]This is because FEC schemes involve both error detection and correction, whereas ARQ schemes involve only error detection, which is far less complex.

6.1 CDMA based buses

A new bus architecture is proposed in [43], which is based on the Direct Sequence Code Division Multiple Access (DS-CDMA) technique. Unlike CDMA for wireless, on-chip signaling is baseband and does not require upconversion to radio frequency (RF). The basic principle of a CDMA based bus is that each transmitter modulates its data using a code sequence, which is orthogonal to the code sequences used by other transmitters. The modulated data is injected onto the bus, using a charge pump. Orthogonal codes (*e.g.*, PN codes, Walsh codes) have the property that a code sequence has a high correlation with itself, and low (ideally, zero) correlation with other code sequences. Therefore, a receiver that demodulates the received signal using the correct code sequence can successfully retrieve the data, even in the presence of other transmissions. This enables an on-chip communication paradigm where multiple transmissions can proceed, in parallel, over the shared physical medium, unlike conventional architectures where only one component can transmit at a given time.

6.2 Wireless interconnects

Another recently proposed inter/intra-chip communication technique is the use of wireless interconnects [44]. Information transfer between components is done using miniature radio transceivers that transmit signals through a micro strip transmission line that is off-chip, but on the package. Such an approach is motivated by the low cost of the radio transceivers. In effect, such a communication mechanism creates a miniature wireless LAN on the chip. A signal to be transmitted is up-linked to the transmission line using a capacitive coupler, which acts as a near field antenna. Similar capacitive couplers down-link the signal to the input ports of the receiving component. Such an RF interconnect system also allows multiple users to transmit information simultaneously, through the use of either Frequency Division Multiple Access (FDMA) or CDMA techniques. The authors of [44] report that an RF communication architecture provides extremely high bandwidth, and supports up to 50 parallel data transmissions. The work in [45] also proposes to use wireless interconnects for configuring, monitoring, and diagnosing the various components in a computer system.

7. CONCLUSIONS

On-chip communication architectures play a crucial role in determining the performance and energy consumption of deep submicron SoCs. Designing an energy efficient and reliable interconnection architecture requires careful optimization at all layers of the design hierarchy. This paper presented a survey of techniques that can be used to design low energy, fault tolerant SoC communication architectures. In addition to the techniques discussed in this paper, other well-known techniques such as code-compression, bus encoding to minimize switching activity, *etc.*, can be used to further reduce on-chip communication energy. By adopting an optimization strategy based on these techniques, system designers will be able to alleviate, if not completely avoid, the interconnect related performance and energy problems that are projected to be bottlenecks for future SoC design.

ACKNOWLEDGEMENTS

This paper is based in part on research funded through Semiconductor Research Corporation System Task Thrust ID 899.001.

8. REFERENCES

[1] D. L. Liu and C. Svensson, "Power consumption estimation in CMOS VLSI chips", *IEEE Journal SSC*, volume 29, issue 6, pp. 663–670, June 1994.

[2] R. Ho, K. Mai, and M. Horowitz, "The future of wires", *Proc. IEEE*, volume 89, issue 4, pp. 490–504, April 2001.

[3] D. Sylvester and K. Keutzer, "A global wiring paradigm for deep submicron design", *IEEE Trans. CAD*, pp. 242–252, Feb. 2000.

[4] K. Lahiri, A. Raghunathan, and S. Dey, "Battery-driven system design: a new frontier in low power design", *Proc. IEEE Intl. Conf. VLSI Design*, pp. 261–267, 2002.

[5] W. J. Dally and J. W. Poulton, *Digital Systems Engineering.* Cambridge University Press, NY, 1998.

[6] H. B. Bakoglu, *Circuits, Interconnections and Packaging for VLSI.* Addison-Wesley, 1990.

[7] J. M. Rabaey, *Digital Integrated Circuits: A Design Perspective.* Prentice Hall, NJ, 1996.

[8] V. Adler and E. G. Friedman, "Repeater design to reduce delay and power in resistive interconnect", *IEEE Trans. Circuits Syst. II*, volume 45, pp. 607–616, May 1998.

[9] G. Y. Wei, *et al.*, "Energy-efficient design of high-speed links", Chapter 8 in *Power Aware Design Methodologies*, Editors: M. Pedram and J. Rabaey. Kluwer Academic Publishers, Norwell, MA, 2002.

[10] H. Zhang, V. George, and J. Rabaey, "Low-swing on-chip signaling techniques: effectiveness and robustness", *IEEE Trans. VLSI Systems*, volume 8, issue 3 , pp. 262–272, June 2000.

[11] ANSI/TIA/EIA-644-1995 standard, "Electrical characteristics of low voltage differential signaling (LVDS) interface circuits".

[12] A. Maheshwari and W. Burleson, "Current sensing techniques for global interconnects in very deep submicron (VDSM) CMOS", *Proc. IEEE Wkshp. VLSI*, pp. 66–70, 2001.

[13] AMBA On-chip Bus Standard, ARM Inc., (`http://www.arm.com/armtech/AMBA?OpenDocument`)

[14] IBM CoreConnect bus architecture., (`http://www-3.ibm.com/chips/products/coreconnect/`)

[15] C. T. Hsieh and M. Pedram, "Architectural energy optimization by bus splitting", *IEEE Trans. CAD*, volume 21, issue 4, pp. 408–414, April 2002.

[16] P. Lettieri and M. B. Srivastava, "A QoS-aware, energy-efficient wireless node architecture", *Proc. MoMuC*, pp. 252–261, 1999.

[17] J. Chang, S. Ravi, and A. Raghunathan, "FLEXBAR: a crossbar switching fabric with improved performance and utilization", *Proc. IEEE CICC*, pp. 405–408, 2002.

[18] H. Chang, *et al., Surviving the SOC Revolution: A Guide to Platform-Based Design.* Kluwer Academic Publishers, Norwell, MA, 1999.

[19] H. S. Wang, L. S. Peh, and S. Malik, "A power model for routers: Modeling Alpha 21364 and InfiniBand routers", *Proc. Symp. on High Perf. Interconnects*, pp. 21–27, 2002.

[20] T. Ye, L. Benini, and G. De Micheli, "Analysis of power consumption on switch fabrics in network routers", *Proc. ACM/IEEE DAC*, pp. 524–529, 2002.

[21] H. S. Wang, *et al.*, "Orion: A power-performance simulator for interconnection networks", *Proc. ACM/IEEE MICRO)*, 2002.

[22] W. J. Dally and B. Towles, "Route packets, not wires: On chip interconnection networks", *Proc. ACM DAC*, pp. 684–689, 2001.

[23] J. Xu and W. Wolf, "Wave pipelining for application-specific networks on chips", *Proc. ACM CASES*, 2002.

[24] L. Benini and G. De Micheli, "Networks on chips: a new SoC paradigm", *IEEE Computer*, volume 35, pp. 70–78, January 2002.

[25] M. Sgroi, *et al.*, "Addressing the system-on-a-chip interconnect woes through communication-based design", *Proc. ACM/IEEE DAC*, pp. 667–672, 2001.

[26] G. Varatkar and R. Marculescu, "Traffic analysis for on-chip networks design of multimedia applications", *Proc. ACM/IEEE DAC*, pp. 795–800, 2002.

[27] K. Lahiri, A. Raghunathan, and S. Dey, "Communication architecture based power management for battery efficient system design", *Proc. ACM/IEEE DAC*, pp. 691–696, 2002.

[28] J. Kim and M. Horowitz, "Adaptive supply serial links with sub-1V operation and per-pin clock recovery", *Proc. IEEE ISSCC*, pp. 268-269, 2002.

[29] L. Shang, L.-S. Peh, and N. K. Jha, "Power-efficient interconnection networks: Dynamic Voltage Scaling with links", *Computer Architecture Letters*, volume 1, May 2002.

[30] S. Irani, S. Shukla, and R. Gupta, "Algorithms for power savings", *Proc. ACM SODA*, pp. 37–46, 2003.

[31] K. Lahiri, A. Raghunathan, and S. Dey, "Evaluation of the traffic-performance characteristics of SOC communication architectures", *Proc. IEEE Intl. Conf. VLSI Design*, pp. 29–35, 2001.

[32] S. Wicker, *Error Control Systems for Digital Communication and Storage.* Englewood Cliffs, Prentice Hall, 1995.

[33] R. Hegde and N. Shanbhag, "Toward achieving energy efficiency in presence of deep submicron noise", *IEEE Trans. VLSI Systems*, volume 8, issue 4, pp. 379–391, August 2000.

[34] F. Worm, *et al.*, "An adaptive low-power transmission scheme for on-chip networks", *Proc. ACM/IEEE ISSS*, 2002.

[35] C.-K. K. Yang, *et al.*, "A 0.5- m CMOS 4.0Gb/s serial link transceiver with data recovery using oversampling", *IEEE Jornal SSC*, volume 33, pp. 713–722, May 1998.

[36] P. Lettieri, C. Fragouli, and M. B. Srivastava, "Low power error control for wireless links", *Proc. MOBICOM*, pp. 139–150, 1997.

[37] D. Bertozzi, L. Benini, and G. De Micheli, "Low power error resilient encoding for on-chip data buses", *Proc. IEEE DATE*, pp. 102–109, 2002.

[38] D. Wingard, "Micro-Network based integration for SOCs", *Proc. ACM/IEEE Design Automation Conf.*, pp. 673–677, 2001.

[39] C. Schurgers and M. Srivastava, "Energy efficient routing in sensor networks", *Proc. Milcom*, pp. 357–361, 2001.

[40] J. Hu and R. Marculescu, "Exploiting the routing flexibility for energy/performance aware mapping of regular NoC architectures", *Proc. IEEE DATE*, pp. 688–693, 2003.

[41] T. Dumitras and R. Marculescu, "On-chip stochastic communication", *Proc. IEEE DATE*, pp. 790–795, 2003.

[42] Session #41, "On-chip communication architectures", *ACM/IEEE DAC*, 2001.

[43] R. Yoshimura, *et al.*, "DS-CDMA wired bus with simple interconnection topology for parallel processing system LSIs", *Proc. IEEE ISSCC*, pp. 370–371, 2000.

[44] M. F. Chang, *et al.*, "RF/wireless interconnect for inter- and intra-chip communications", *Proc. IEEE*, volume 89, issue 4, pp. 456–466, April 2001.

[45] H. Eberle, "A radio network for monitoring and diagnosing computer systems", *IEEE Micro*, pp. 60–65, Jan./Feb. 2003.