

Low-Power Bus Architecture Composition for AMBA AXI

Sangkwon Na*, Sung Yang**, and Chong-Min Kyung*

Abstract—A system-on-a-chip communication architecture has a significant impact on the performance and power consumption of modern multi-processors system-on-chips (MPSoCs). However, customization of such architecture for a specific application requires the exploration of a large design space. Thus, system designers need tools to rapidly explore and evaluate communication architectures. In this paper we present the method for application-specific low-power bus architecture synthesis at system-level. Our paper has two contributions. First, we build a bus power model of AMBA AXI bus communication architecture. Second, we incorporate this power model into a low-power architecture exploration algorithm that enables system designers to rapidly explore the target bus architecture. The proposed exploration algorithm reduces power consumption by 20.1% compared to a maximally connected reduced matrix, and the area is also reduced by 20.2% compared to the maximally connected reduced matrix.

Index Terms—Low power, architecture composition, AMBA AXI

I. INTRODUCTION

The improvement of the process technology has enabled more and more functionality to be integrated into a single chip, which has increased the amount of on-chip communication between the integrated components. In a highly integrated system, the on-chip communication

architecture becomes a critical factor affecting overall system performance and power consumption [1]. Designers therefore must give special emphasis on the choosing and optimizing of the on-chip communication architecture early in the design flow, preferably at the system level.

The on-chip communication architecture, such as the bus matrix [2], has customizable topologies and parameters, which create a vast exploration space [3]. Different configurations in this space may have various power budgets and performance characteristics. Hence, it is crucial to find the rapid exploration technique which minimizes the power consumption of the target bus architecture under the given performance constraints, especially for mobile devices that have a limited energy budget.

In this paper, we propose the rapid bus architecture composition algorithm for the AMBA AXI [4] bus to meet the power consumption requirement of modern MPSoC systems under the given performance constraint. Several different types of communication architectures, such as shared buses [18,19], bus matrix [2] and Network-on-Chips (NoCs) [20], have been adopted as the main on-chip communication architecture in many MPSoC systems. While shared bus architecture, proposed by AMBA [18] and CoreConnect [19] have been widely, it was hard to meet the high bandwidth requirements of modern high-performance applications on these architectures. Network-on-chips that can overcome these bandwidth limitations require large area and consume even more power than the shared bus-based designs in current lithographic process [21]. Recently, bus matrix or crossbar switch became increasingly popular for on-chip communication due to high bandwidth, such as ARM PL300/301 [22], Synopsys Design Ware AMBA3 fabric [23], and Sonics SonicsMX [24]. In this paper, AMBA AXI is adopted for low-power bus architecture composition since AMBA

Manuscript received Jun. 2, 2009; revised Jun. 11, 2009.

* Dep. EE., Korea Advanced Institute of Science and Technology

** System LSI Division of Samsung electronics

E-mail : skna@vslab.kaist.ac.kr

AXI provides a lot of advanced features as follows: 1) ability to issue multiple outstanding addresses, 2) out-of-order transaction completion, and 3) easy addition of register stages to provide timing closure.

This paper is organized as follows: in section II, we introduce related works about architecture composition algorithms; power models are described in section III; the proposed bus synthesis flow is presented in section IV; section V shows the experimental results; and concluding remarks are given in section VI.

II. RELATED WORKS

Although there are a lot of works dealing with hierarchical shared bus architecture synthesis [6-10] and NoC architecture synthesis [11-15], few efforts have focused on bus matrix synthesis. Ogawa *et al.* [16] proposed a transaction-based simulation environment which allows designers to explore and design a bus matrix. Since designers have to manually specify the communication topology, arbitration scheme and memory mapping, it is hard to apply this approach to the complicated systems.

The automated synthesis approach for STBus crossbar proposed by Murali *et al.* [17] is a similar work to ours. While their work primarily deals with automated crossbar topology synthesis, the communication parameters that heavily affect the system performance are not considered. In addition, they assumed that critical data streams cannot overlap on the same bus, limited the maximum number of components that can be attached to a bus. Furthermore designers have to determine threshold values of traffic overlap as an input based on which components are allocated to separate busses.

The bus matrix synthesis flow suggested by Pasricha *et al.* [3] comes closest to our goal. They synthesized both the topology and communication architecture parameters for the bus matrix, selected arbitration schemes that can allow multiple constraints. For this approach, designers does not have to specify threshold values about data traffic or limit the number of components on a bus. However, their clustering method just deals with slave components rather than whole busses. In other words, if there is a slave component which connected several busses and each bus has variable amounts of traffic, we have to determine whether include this slave component in clustering or not. it may cause overestimated or under-

estimated designs. In order to solve this dilemmatic problem, our approach clusters each bus connected with slave components in terms of a power gain without the performance degradation.

III. POWER MODELS

To estimate the bus power consumption at system level, we develop the power model of AMBA AXI bus. The power model consists of a switching activity and a load capacitance. In this mode, the product of estimated switching activity and load capacitance is proportional to power consumption.

The switching activity is the probability of signal switching per one clock cycle. Therefore, we build a switching activity model in relation to an average bit switching activity, i.e., α , as follows:

$$\begin{aligned} K_{utilize} &= \frac{1}{size_{TR} + 2} \\ SW_{add_ch} &= num_{TR} \cdot 2K_{utilize} \cdot \alpha \\ SW_{data_ch} &= num_{TR} \cdot (1 - K_{utilize}) \cdot \alpha \\ SW_{decode_ch} &= num_{TR} \cdot K_{utilize} \cdot \alpha \\ SW_{response_ch} &= num_{TR} \cdot 2K_{utilize} \cdot \alpha \end{aligned} \quad (1)$$

where $size_{TR}$ denotes the size of burst transfer, and num_{TR} denotes the number of transfer.

The load capacitance is evaluated with the complexity of logic, i.e., the number of input/output signals; the load capacitance model is built according to the channel direction: the write channel and the read channel as follows:

$$\begin{aligned} C_{WR} &= \min(2^{\lceil \log_2 n \rceil}, n + 1) \\ C_{RD} &= n \end{aligned} \quad (2)$$

where n is the number of signals of each logic.

IV. BUS SYNTHESIS FLOW

Fig. 1 shows the proposed bus synthesis flow. First, we simulate the target system with fully connected matrix at transaction level to analyze communication characteristics for the target bus system. Next, we remove the unused channels, and define it as a maximally

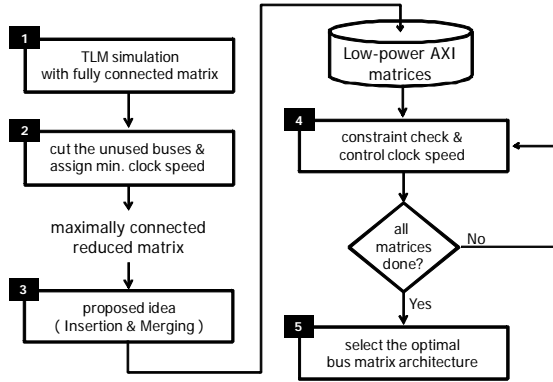


Fig. 1. Bus synthesis flow.

connected reduced matrix. Then, the proposed algorithm that consists of Insertion and Merging method is applied to that matrix; the modified target bus architecture is verified at transaction level under the given performance constraint. We assume that hardware and software partitioning of the target system is already completed and functionalities are mapped onto each IP appropriately.

As described in Fig. 2, the basic idea of insertion and merging is to distribute concentrated heavy traffic and relieve complicated bus logic. Insertion depicted in Fig. 3 is the first phase that reduces the load capacitance of interfaces, and is independently applied to each interface. As listed in Table 1, Insertion leads to reduce the load capacitance.

Merging is the second phase of the power and area reduction caused by clustering newly inserted interfaces. For all inserted interfaces, we cluster them by using branch and bound-based algorithm. As listed in Table 2, the branching algorithm starts by clustering two slaves at

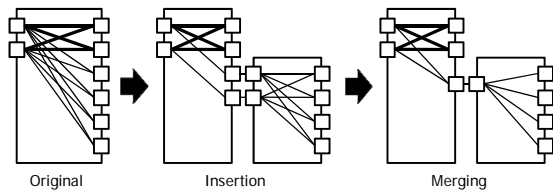


Fig. 2. Insertion and merging algorithm.

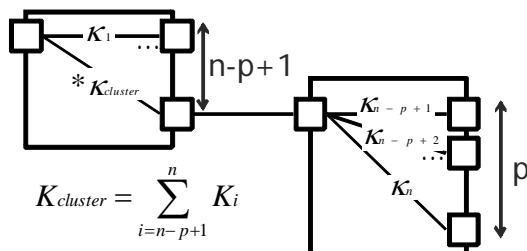


Fig. 3. Insertion.

Table 1. Insertion algorithm

$$P_{before} = C(n) \cdot \sum_{i=1}^n K_i$$

$$P_{after} = C(n-p+1) \cdot \sum_{i=1}^n K_i + C(p) \cdot K_{cluster}$$

- 1: **while** $P_{after} < P_{before}$
- 2: **find** p which minimizes P_{after}
- 3: **if** p exists **then**
- 4: insert new slave interface to solution queue
- 5: **else**
- 6: exit
- 7: **end if**
- 8: go to the next interface
- 9: **end while**

Table 2. Merging algorithm

- 1: **while** solution queue is not empty
- 2: pop two inserted slave interfaces from solution queue
- 3: calculate maximum path delay for the merged interface
- 4: **if** maximum path delay < clock period **then**
- 5: merge two slave interfaces
- 6: **end if**
- 7: **end while**

a time, and verifies two bound conditions as below:

- Whether there are no shared slave interfaces.
- Whether a critical path delay is longer than a maximum feasible clock period, which is given from the specification.

V. EXPERIMENTAL RESULTS

For the verification of the power model accuracy, we performed gate-level simulation with power analysis. Fig. 4 shows the comparison between the gate-level simulation and the power model for the write channel. The maximum error is below 2.6% and the average error is about 0.96%. While the power model provides significantly accurate power estimation, it allows to obtain the power consumption 1,000 times as fast as the gate-level simulation.

To validate proposed bus synthesis framework, we applied it to the example system which is a multimedia SoC embedding an H.264 decoder and other processing elements. For this experiment, we built a simulation

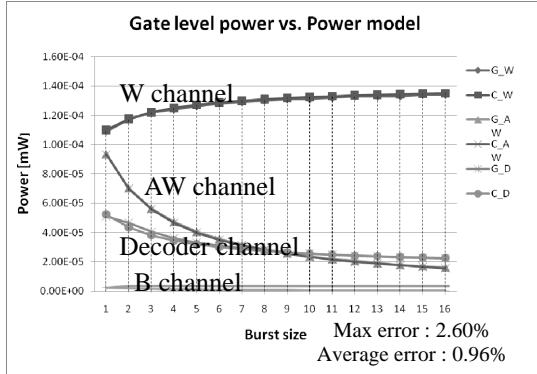


Fig. 4. The accuracy between gate-level simulation and the proposed power model; G_x denotes the gate level power of x channel, C_x denotes the power model of x channel.

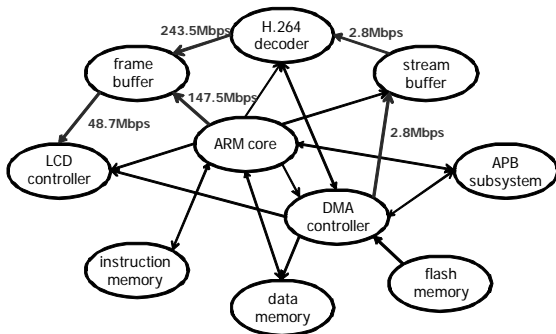


Fig. 5. Communication graph for the example system.

environment using TLM IPs in SoCDesigner [5]. Fig. 5 shows the communication graph and the throughput constraint for the given application.

We synthesized three bus matrices with TSMC 0.18 μm process technology, and listed the area in table III. The total area is reduced by 20.2% compared to the maximally connected reduced matrix. We also analyzed the power consumption of each bus matrix using PrimeTime. As listed in Table 3, the proposed algorithm reduces power consumption by 20.1% compared to the maximally connected reduced matrix.

Table 3. The area and power comparison

	Fully	Maximally Connected	Proposed
Area (gate)	150997(1.00)	49710(0.33)	39753(0.26)
Power (mW)	2.96(1.00)	1.23(0.42)	0.98(0.33)

VI. SUMMARY

The goal of this paper is to present a rapid system-level architecture composition method for AMBA AXI bus-based system. In order to reduce the estimation time

of the bus power consumption, we develop the power model for AMBA AXI bus with an analytic method. We also suggest the insertion and merging method to reduce the bus power consumption; the proposed approach inserts an intermediate interface to distribute concentrated traffic and merges inserted interfaces to reduce the complexity of logic. We reduce the power consumption by 20.1% compared to the maximally connected reduced matrix with a 20.2% area reduction.

REFERENCES

- [1] R.Ho, K.W.Mai, and M.A, "The Future of Wires", Proc. IEEE, vol. 89, April 2001.
- [2] AMBA AHB Interconnection Matrix, www.synopsys.com/products/designware/amba_solutions.html
- [3] S.Pasricha, N.Dutt, and M. Ben-Romdhane, "Constraint-Driven Bus Matrix Synthesis for MPSoC", ASPDAC 2006.
- [4] AMBA AXI Protocol v1.0 Specification. <http://www.arm.com/products/solutions/AMBA3AXI.html>
- [5] ARM SoC Designer <http://www.arm.com/products/DevTools/SoCDesigner.html>
- [6] M. Gasteier and M. Glesner, "Bus-based communication synthesis on system level," *ACM TODAES* 1999.
- [7] S. Pasricha, N. Dutt, and M. Ben-Romdhane, "Fast Exploration of Bus-based On-chip Communication Architectures," *CODES+ISSS* 2004.
- [8] S.Pasricha, N.Dutt, E.Bozorgzadeh, and M.Ben-Romdhane, "Floorplan-aware Automated Synthesis of Bus-based Communication Architectures," *In Proc. of DAC* 2005.
- [9] A. Pinto, L. Carloni, and A. Sangiovanni-Vincentelli, "Constraint driven communication synthesis," *DAC* 2002.
- [10] D. Lyonnard, S. Yoo, A. Baghdadi, and A. A. Jerraya, "Automatic generation of application-specific architectures for heterogeneous multiprocessor system-on-chip," *DAC* 2001.
- [11] K. Srinivasan, et al, "Linear Programming based Techniques for Synthesis of Network-on-Chip Architectures," *ICCD* 2004.
- [12] D. Bertozzi, et al., "NoC synthesis flow for customized domain specific multiprocessor systems-on-chip," *IEEE TPDS* 2005.

- [13] U. Ogras and R. Marculescu, "Energy- and Performance-Driven NoC Communication Architecture Synthesis using a Decomposition Approach," *DATE* 2005.
- [14] A. Pinto, L. P. Carloni, and A. L. Sangiovanni-Vincentelli, "Efficient Synthesis of Networks On Chip," *ICCD* 2003.
- [15] A. Jalabert, S. Murali, L. Benini, and G. De Micheli, "xpipesCompiler: A Tool for instantiating application specific Networks on Chip," *DATE* 2004.
- [16] O. Ogawa, et al, "A Practical Approach for Bus Architecture Optimization at Transaction Level," *DATE* 2003.
- [17] S. Murali and G. De Micheli, "An Application-Specific Design Methodology for STbus Crossbar Generation", *DATE* 2004.
- [18] ARM AMBA Specification and Multi layer AHB Specification, (rev2.0), <http://www.arm.com>, 2001.
- [19] IBM On-chip CoreConnect Bus Architecture, www.chips.ibm.com/products/coreconnect/index.html
- [20] Benini and G.D.Micheli, "Networks on Chips: A New SoC Paradigm", *IEEE Computers*, Jan. 2002.
- [21] F. Angiolini, et al., "Contrasting a NoC and a Traditional Interconnect Fabric with Layout Awareness", *DATE* 2006.
- [22] ARM documents - System-on-Chip, <http://www.arm.com/documentation/SoC/index.html>
- [23] Synopsys Design Ware IP - AMBA solutions, http://www.synopsys.com/products/designware/amba_solution.html
- [24] SonicsMX datasheet, http://www.sonicsinc.com/documents/SMX_Data_Sheet.pdf



Sangkwon Na received the dual B.S degrees in Electrical Engineering and Computer Engineering from Ajou University, Suwon, Korea, in 2003. Since 2004, he has been pursuing the unified course of the M.S and the Ph.D. degree in the Department of Electrical Engineering and Computer Science at KAIST. His research interests include multimedia/video compression algorithm, scalable video coding and low power design methodology.



Sung Yang received the B.S. degree in electrical engineering from the Chonbuk University, Chonju, Korea, in 2006 and the M.S. degree in electrical engineering and computer science from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2008. Since 2008, he has been System LSI Division of Samsung electronics. His research interests include implementation methodology for high-performance on-chip CPU.



Chong-Min Kyung received the B.S. degree in Electronics Engineering from Seoul National University in 1975, the M.S. and Ph.D. degree in Electrical Engineering from KAIST in 1977 and 1981, respectively. From April 1981 to January 1983, he worked at Bell Telephone Laboratories, Murray Hill, New Jersey as a postdoc. Since he joined KAIST in 1983, he has been working on System-on-a-Chip design and verification methodology, processor and graphics architectures for high-speed and/or low-power applications including mobile video codec.

He received the Most Excellent Design Award, and Special Feature Award in the University Design Contest in the ASP-DAC 1997 and 1998, respectively. He received the Best Paper Awards in the 36th DAC held in New Orleans, LA, the 10th ICSPAT(International Conference on Signal Processing Application and Technology), Orlando, FL, in September 1999, and the 1999 ICCD (International Conference on Computer Design), Austin, TX. He was General Chair of A-SSCC(Asian Solid-State Circuits Conference) 2007, and ASP-DAC 2008. In 2000, he received National Medal from Korean government for his contribution to research and education in IC design. He is a member of NAEK(National Academy of Engineering Korea) and KAST(Korean Academy of Science and Technology). He is an IEEE fellow and Hynix Chair Professor at the KAIST.