Packet-Switched On-Chip Interconnection Network for System-On-Chip Applications

Se-Joong Lee, Kangmin Lee, Seong-Jun Song, and Hoi-Jun Yoo

***Abstract*— Increasing complexity of a system-on-chip design demands efficient on-chip interconnection architecture such as on-chip network to overcome limitations of bus architecture. In this brief, we propose a packet-switched on-chip interconnection network architecture, through which multiple processing units of different clock frequencies can communicate with each other without global synchronization. The architecture is analyzed in terms of area and energy consumption, and implementation issues on building blocks are addressed for cost-effective design. A test chip is implemented using 0.38-μm CMOS technology, and measured its operation at 800 MHz to demonstrate its feasibility.**

***Index Terms*—Communication architecture, interconnection network, network on chip (NoC), on-chip network (OCN), system on chip (SoC).**

I. Introduction

As THE NUMBER and functionality of processing units (PUs) in systems on chips (SoCs) increase, complexity of interconnection architectures of the SoCs have also been increased. Point-to-point interconnection maximizes chip performance [1], but its design complexity takes high development cost. Bus architecture has been improved to come up with requirements of high-performance SoCs [2], however, its inherently poor scalability limits the enhancement.

Recently, the on-chip network (OCN) has been actively studied as a new on-chip communication architecture [3]–[11]. Based on a flexible network architecture, the OCN can provide not only sufficient bandwidth but also additional functionalities required for high-performance SoCs. Discussions and proposals on networking on a chip have been reported regarding architecture and implementation factors [3], [4]. Some other previous works like [5]–[7] visualized the OCN architectures providing specifications of network topology, packet format, and bitwidth of link wires.

However, most of the previous works lack implementation issues because they did not touch practical issues regarding chip implementation. Their architecture specifications like topology, and link bitwidth are determined without considerations of implementation constraints like low-power consumption and small footprint. There are some reports of chips with network communication architectures [8]–[11]. However, the network architecture of [8] is a static interconnection rather than a dynamic packet-switched network, and that of [9] covers synchronous multiprocessors in a limited architecture rather than a general communication architecture for an SoC design. Even though some implementation issues like high-performance OCN design, guaranteed quality of service, and low-power system design are described in [12]–[14], physical design issues and quantitative analyzes of OCN core architecture are remained untouched.

Manuscript received March 26, 2004; revised August 27, 2004. This work was supported in part by the Ministry of Information and Communications, Korea, under the ITRC Support Program. This paper was recommended by Associate Editor T. S. Rosing.

The authors are with the Semiconductor System Laboratory, Department of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 305-701, Korea (e-mail: shocktop@ee-info.kaist.ac.kr).

Digital Object Identifier 10.1109/TCSII.2005.848972

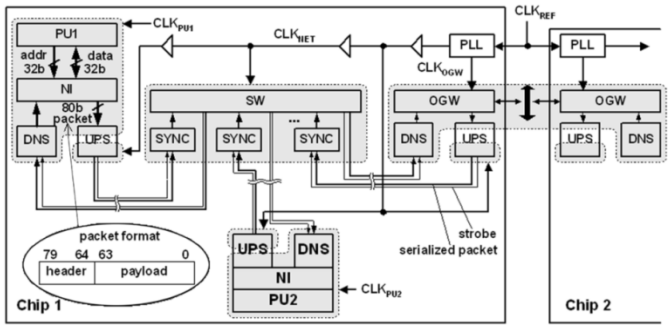

Fig. 1. Overall architecture of the OCN.

In this brief, we propose a dynamic packet-switched OCN which aims at interconnecting multiple PUs of different clock frequencies. The OCN is designed to remove the burden of global synchronization to facilitate designing an SoC using high-speed clock. Architectural factors are optimized and building blocks are designed, considering the physical design factors like area and power consumption. A test chip is fabricated to demonstrate the feasibility of the OCN architecture of this work.

The remainder of this paper is organized as follows. In Section II, overall architecture and operation of the OCN are explained. In Section III, architecture optimization is described. In Section IV, building block design is shown. The implemented chip and measurement results are provided in Sections V, and Section VI concludes the paper.

II. OCN Architecture

A. Overall Architecture

Fig. 1 shows an overall block diagram of the OCN. It consists of the network interface (NI), up_sampler (UPS), link wires, first-in-first-out (FIFO) synchronizer with a queuing buffer (SYNC), switch, down_sampler (DNS), and off-chip gateway (OGW). Using an address and/or a data output(s) of a PU, a corresponding NI generates a packet. A packet is an 80-b bit stream consisting of a 16-b header, and a 64-b payload. In the header, routing information is encoded into 16 b. The packet also can be transferred to another chip through the OGW. An NI

has an address map which defines a destination PU according to an address, and a header of a packet is generated based on the address map. The packet header size is fixed to reduce latency and hardware complexity of header parsing logic.

The UPS, link wires, SYNC, switch, and DNS are called *core network*. The core network delivers a packet to a correct destination PU. The core network is serialized in order to lower wiring complexity by reducing the number of link wires. Instead, it uses its own high-frequency clock to sustain bandwidth of the network system. The UPS and the DNS interface the high-speed serialized core network to the NI. Design optimization in the serialization technique of the core network will be described in Section III. And, analysis of the core network topology also will be shown in Section III.

Clock domains of the OCN are illustrated as a dotted gray box in Fig. 1. Without global synchronization, each PU can use its dedicated clock source constituting a distinct clock domain. The clock for OGWs $(\mathrm{CLK_{OGW}})$ is slow enough to do chip-to-chip signaling, and it is synchronized with the reference clock $(\mathrm{CLK_{REF}})$ so that all the OGWs connected together are synchronized with each other.

The $\mathrm{CLK_{NET}}$ is distributed without any skew compensation, and packet transmission is performed using mesochronous communication. When a packet is transmitted from one clock domain to another, a strobe signal (STB) is transmitted together with the packet as a timing reference. And, the DNS and the SYNC compensate phase difference between the STB signal and local clock. This architecture eliminates the burden of global synchronization, thus enables use of a high-speed clock [15], i.e., 800 MHz in this paper, which facilitates performance enhancement of an on-chip system.

An implementation issue on the packet transmission using the STB will be addressed and our approach to solve the issue will be described in Section IV.

B. Packet Routing Scheme

A packet is transferred to a destination according to route information in the packet header. And, a response packet of the destination, if necessary, returns to the source along the exactly reverse path.

In a switch, each switch port has its own port index as shown in Fig. 2(a). A packet header contains a series of the port indexes which are called output port indexes (OPIs) as depicted in Fig. 2(b). An OPI_k represents an index of a switch port to which the packet must be routed by a kth switch. When a packet arrives at a switch, the switch fetches an OPI at the head of the packet header. Then, the header is left shifted to locate the next OPI at the most-significant-bit part of the packet header, and the input port index (IPI) is attached to the tail as shown in Fig. 2(c). Using this packet header modification, each switch can find its OPI at the head of the packet header, and an NI of a destination PU obtains a header for the return path by bit-wise flipping the header. Since an IPI is a flipped index of an input port, the bit-wise flipping outputs correct OPIs.

Using this routing scheme, switches do not have routing tables which contain topology-dependent data. As a result, the core network design is decoupled from system design, so that can be easily ported to a certain system.

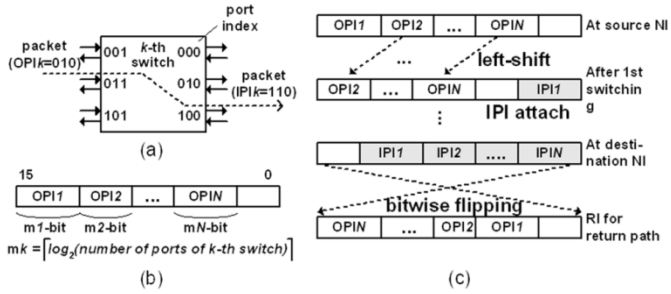

Fig. 2. (a) Switch port index. (b) Header format. (c) Header modification.

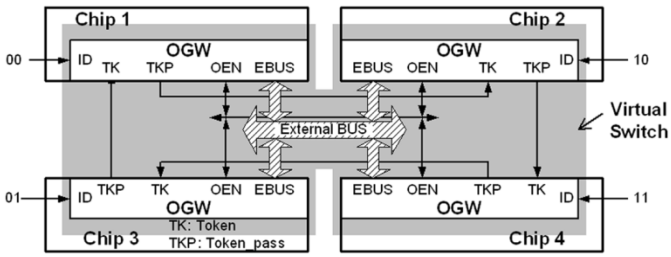

Fig. 3. Chip-to-chip connection using OGWs.

This routing scheme provides fixed routing instead of adaptive routing, which reduces design complexity of transport layer: Adaptive routing utilizes multiple paths for packet transmission, and this incurs packet-ordering issue which may require large queuing buffers at each NI.

C. Chip-to-Chip Connectivity

OGWs provide chip-to-chip packet transaction without any additional external component. And, the inter-chip communication does not require specific header type, but it uses the same format used in the core network.

Fig. 3. shows a configuration of multichip interconnection using OGWs. The OGWs are connected together constituting a large virtual switch, and inter-chip packet transaction is performed through the virtual switch. An OGW acts like a switch port, and the external bus performs the functionality of a switch fabric. Each OGW has its own identification index (ID) which is used like a port index of a switch. Accesses of the bus are arbitrated based on token-ring methodology.

If an OGW having a packet to transmit gets a token, it outputs the packet on the bus with assertion of the output-enable (OEN) signal. Other OGWs monitor the OEN, and if it is asserted, check whether the OPI of the packet is matched with their IDs. If they are matched, the OGW accepts the packet and performs header modification.

Using the bus architecture with token-based arbitration, chip-to-chip interconnection can be accomplished only by wires without any external switch or arbiter.

III. Architecture Optimization

A. Topology Selection

When the number of PUs are given, the numbers of switches and SYNCs are determined by OCN topology. Most of the previous works like [7] and [9] utilize Mesh topology or its

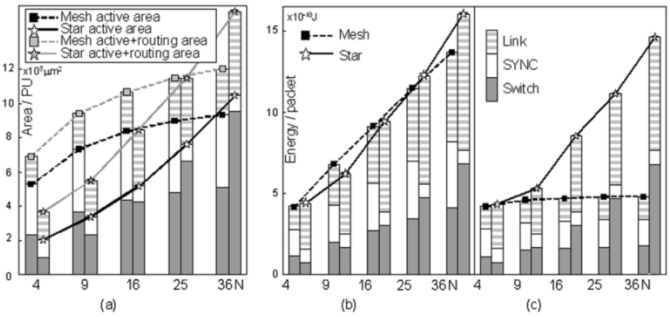

Fig.4. (a) Area, and (b) and (c) energy consumption comparison.

derivative which are widely used and studied for parallel computing architectures [16]. On the contrary, Star topology has not been popularly used because it has a limitation of scalability. In on-chip situation, however, the number of PUs (N) to be connected is limited to a few tens. Moreover, PUs may be placed irregularly to minimize chip area, therefore, the regular structure of Mesh topology may not be made in an SoC design.

In this part, we estimate area and energy consumptions of Mesh and Star topologies to choose cost-effective topology for the OCN. The total area of SYNCs, switches, and routing wires according to N in the Mesh architecture is given by

$$\text{Mesh area} \approx (5N - 4\sqrt{N})A_{\text{SYNC}} + (25N - 36\sqrt{N})A_{1\text{SW}} + 4(N - \sqrt{N})A_{\text{LNK}} \quad (1)$$

where $A_{\text{SYNC}}, A_{\text{kSW}}$, and A_{LNK} are areas of SYNC, $k \times k$ switch, and 80-b 1-mm metal wires, respectively. Distance between adjacent PUs is assumed to be 2 mm. Switch area is almost proportional to the square of the number of ports, so that the A_{kSW} is approximated by $k^2 \times A_{1\text{SW}}$. Similarly, the Star network area can be expressed as

$$\text{Star area} \approx NA_{\text{SYNC}} + N^2A_{1\text{SW}} + N\sqrt{N}A_{\text{RT}}. \quad (2)$$

The energy consumption of a network is evaluated by average energy consumed by one packet transmission from an arbitrary PU to another PU, which is represented by

$$\text{Average energy} \approx H_{\text{AVG}}(E_{\text{SYNC}} + E_{\text{SW}}) + L_{\text{AVG}}E_{\text{LNK}} \quad (3)$$

where H_{AVG} and the L_{AVG} mean average switch hop count and average link length between two PUs, respectively. The E_{SYNC} and the E_{SW} represent the energy required to queue and switch one packet, respectively. The E_{LNK} is the energy consumed by 1-mm 80-b link wires for one packet transmission.

The energy and area are evaluated and plotted in Fig. 4 based on implementation data of this work. As shown in Fig. 4(a), Star has less active area when the N is under 25. It still requires less area even if routing area is taken into account when the N is under 16. In the energy analysis, two kinds of traffic pattern are used: uniformly distributed traffic [Fig. 4(b)] and short-distance traffic [Fig. 4(c)] in which traffic exists only inside of local 2×2 PUs. In the case of the first traffic, Star and Mesh show similar energy consumption if $\text{N} < 36$. For the second traffic, however, locality of traffic pattern makes the Mesh energy consumption unchanged while the N increases.

The exact values may depend on process parameters and detail design schemes, but we can observe in this analyzes that the Star occupies smaller area and consumes similar energy when the N is under 9 even in the high-locality traffic condition.

And, when the traffic is random, the Star is can be chosen for cost-effective topology until the N becomes $25 \sim 36$. When N is larger than the criteria, Mesh or hybrid topology can be chosen. In this work, the prototype chip is designed for 4 PUs, and Star topology is used.

B. Physical Transfer Unit Size

The physical transfer unit (phit) is a unit into which a packet is divided and transmitted through the core network. The phit size is the bitwidth of a link. If the phit size is smaller than the packet length, serialization must be performed by the factor of

$$\text{Serialization ratio (SERR)} = \text{packet size}/\text{phit size}. \quad (4)$$

Using large phit size, is inefficient in the respect of the OCN cost, i.e., area and power consumption. On the contrary, use of a small size phit reduces the number of link wires, thus the space between link wires can be widen to decrease coupling capacitance. And, the small phit reduces the switch fabric size, therefore, area and power consumption of a switch can be reduced [10]. In more detail, switch power consumption is the sum of arbiter and switch fabric power, and the switch fabric power is the dominant. Power consumption of a $k \times k$ cross-point switch fabric (P_{kSF}) is expressed as

$$P_{\text{kSF}} \propto k \times \text{port bitwidth} \times (f_D C_D V^2 + f_C C_G V^2) \quad (5)$$

where f_D and f_C mean operation frequencies of data and control signals, respectively. C_D and C_G mean coupling capacitance of a switch fabric data wire and gate capacitance of a cross-point unit switch, respectively. C_D is proportional to the switch fabric size. When serialization is applied, port bitwidth and C_D are divided by SERR, and the f_D is multiplied by SERR, which means the P_{kSF} is proportional to 1/SERR.

Another power factor which can be affected by the serialization is switching activity factor. However, we do not consider the factor in this paper since it can be well controlled using a data coding scheme [17].

When the serialization is used, operation frequency of the core network, CLK_{NET}, must be increased by the factor of the SERR in order to maintain network bandwidth. Considering additional circuitry for the frequency increment, energy consumption and area of UPS, DNS, link, SYNC, and switch are analyzed according to the SERR. Operation frequency without the serialization is set to be 200 MHz, and detail designs of the blocks are based on the implementation results.

Fig. 5. shows the analysis results of energy consumption per packet transmission and area of building blocks in star topology OCN. In Fig. 5(a), the energy consumptions in a switch and links decrease as the SERR increases as mentioned above. On the contrary, those of a UPS, DNS, and SYNC increase due to additional circuitry for the serialization. In the most cases, the SERR of 4 minimizes the overall power consumption. As shown in Fig. 5(b), the serialization also reduces the overall OCN area effectively, and the SERR of 4 is the optimal. The SERR of 8 is not only inefficient in terms of power, but also impractical because it requires 1.6-GHz operation which is beyond physical limitation that the process technology of this work can support. In the implemented chip, in which SERR of 2 is energy optimal

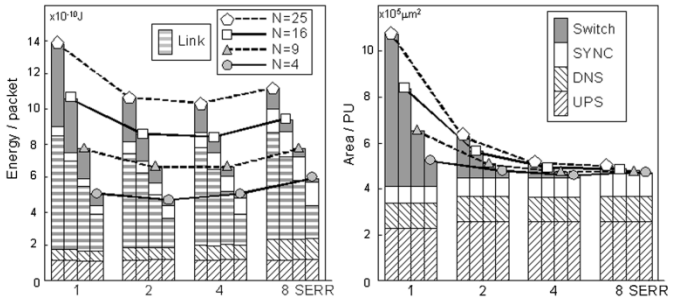

Fig. 5. (a) Energy and (b) area of an OCN according to SERR.

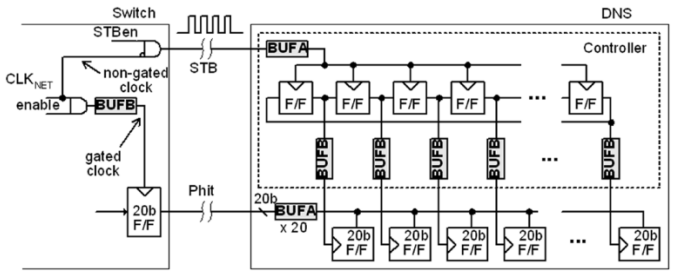

Fig. 6. Matched-delay architecture in a DNS.

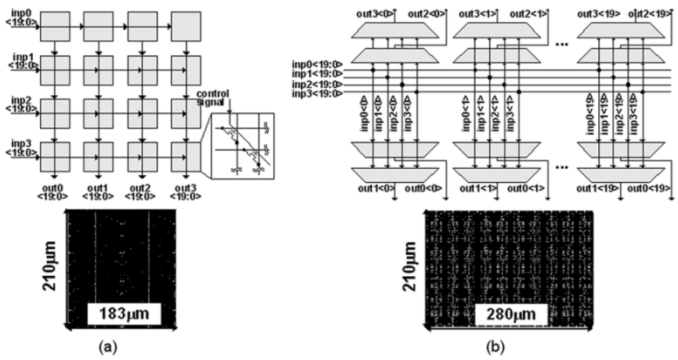

Fig. 7. Schematics and layouts of (a) crosspoint and (b) MUX-tree SFs.

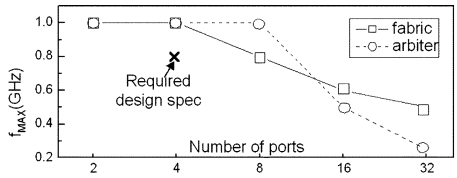

Fig. 8. f_{MAX} versus number of ports of a crosspoint switch.

and 4 is area optimal, the phit size is determined as a quarter packet length to cover design issues on 4:1 serialized network.

IV. Building Block Design

A. Source Synchronous Transmission

Without global synchronization, the OCN transfers phits by source synchronous scheme in which an STB is transmitted with phits. In order to suppress unnecessary power consumption, the STB is activated only when a phit is valid. A receiving part fetches the phit using the STB signal, and synchronizes the phit with the local clock using a FIFO synchronizer [18].

A design challenge in the above scheme is that the delay time for distributing the STB signal over latches of the FIFO is significant, so that the STB skew can possibly result failure of latching the phit data. For example, a DNS has 240 latches to be driven by the STB, and the delay time is 0.6 ns which is about half of the STB cycle time when the $\mathrm{CLK_{NET}}$ frequency is 800 MHz. Use of a delay-locked loop (DLL) or phase-locked loop (PLL) for compensating the skew is not adequate solution because of its area and power overhead.

To solve this problem, the matched-delay FIFO architecture is proposed. Fig. 6 shows the matched-delay scheme in a DNS FIFO. It is assumed that the switch uses clock gating technique to reduce power consumption at the output flip-flops (F/Fs). The DNS FIFO has 12 20-b F/Fs. Each 20-b F/F is enabled by a clock signal generated by the ring counter. The driving buffer A (BUFA) and B (BUFB) drive 12 and 20 F/Fs, respectively. In the matched-delay architecture, the STB is generated by the $\mathrm{CLK_{NET}}$ directly, so that the STB signal leads the phit signals by the amount of the BUFB delay time (t_{BUFB}). As a result, the total delay times of the STB and phit are expressed as

$$\text{STB delay} \approx t_{\mathrm{PD}} + t_{\mathrm{BUFA}} + t_{\mathrm{CQ}} + t_{\mathrm{BUFB}} \quad (6)$$

$$\text{Phit delay} \approx t_{\mathrm{BUFB}} + t_{\mathrm{CQ}} + t_{\mathrm{PD}} + t_{\mathrm{BUFA}} = \text{STB delay} \quad (7)$$

where t_{PD} and the t_{CQ} mean propagation time on the long wire and clock-to-Q delay time of a F/F, respectively.

This architecture provides the matched delay regardless of the number of F/Fs, the serialization ratio, and operation frequency.

B. Switch Design

A switch fabric (SF) of a switch is designed with crosspoint structure rather than MUX-tree to reduce the switch area and power consumption. Fig. 7 shows 4×4 SF schematics and layouts for the two structures. While the MUX-type requires MUX circuits and complex wiring, the crosspoint SF consists of simple nMOSs as node switches and minimum wires. As a result, the layout of the crosspoint SF occupies 65% area compared to that of MUX-structure. In a power consumption simulation, the crosspoint SF consumed about 50% power compared to the MUX-tree SF, by virtue of reduced parasitic coupling capacitances of the fabric wires and simple logic gates.

In some previous works [19], [20] which compare a crosspoint type with a MUX-tree type, their power consumptions are reported to be similar with each other. In [19], however, a switch chip in which output driver consumes most of power is considered rather than an on-chip switch. The [20] analyzes the power consumptions in more detail. However, the size of the crosspoint SF is over estimated: the wire pitch of the SF is assumed to be four times of the minimum pitch, while the SF designed in this work has the minimum pitch. This is because [20] assumes that a switch port is a single-bit line, so that every cross-point of the SF wires has a node switch. However, in this work, the bit-width of a switch port is 20 b so that only one unit switch per 20 cross points is required.

One disadvantage of the crosspoint SF is that its maximum operation frequency (f_{MAX}) is lower than that of MUX-tree SF. The f_{MAX} which is determined by delay time of a critical path for worst case input patterns is calculated for the various number of ports, and shown in Fig. 8. The f_{MAX} of MUX-type arbiter

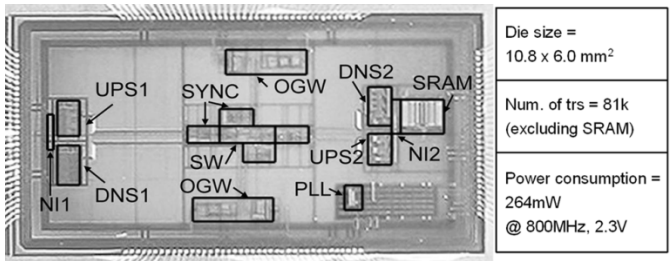

Fig. 9. Die photo and characteristics.

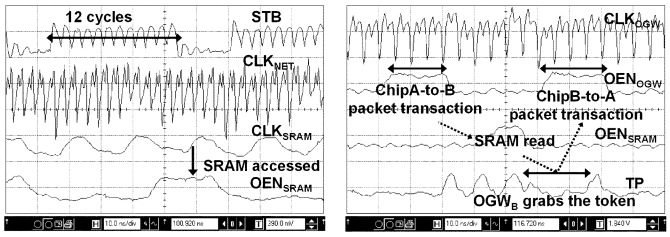

Fig. 10. Waveforms of (a) single chip and (b) chip-to-chip operations.

with single pipeline structure [21] is also evaluated. The graph implies that the performance of the crosspoint switch is enough to satisfy the specification of the prototype chip, i.e., 800 MHz operation of a 4×4 switch.

V. Implementation Results

A. Chip Implementation

To demonstrate feasibility of the OCN architecture, a test chip is implemented using 0.38-μm CMOS technology. Fig. 9 shows a photograph of the chip and its characteristics.

The 4×4 crosspoint switch at the center of the chip constitutes a serialized star-connected network. A 1 kB SRAM is embedded as an IP of an SoC. Clock generation and distribution are made with consideration of modeling an SoC environment in which global synchronization is impossible: all the clock sources are stimulated independently by external equipments, and the $\mathrm{CLK_{NET}}$ is distributed over the entire chip manually without consideration of clock skew problems.

B. Measurement Results

Fig. 10 shows measurement results of a single chip and a chip-to-chip operation. In the single chip test, the $\mathrm{CLK_{NET}}$ is set to 300 MHz, and the SRAM clock is 50 MHz. Three packets, i.e., 12 phits, are transferred to the switch, and the STB signal is probed. Among the packets, the first packet contains an SRAM read command. The measured $\mathrm{OEN_{SRAM}}$, output enable signal of the SRAM, indicates that the SRAM is successfully accessed.

In the chip-to-chip operation, two chips are connected through OGWs. The core network of each chip uses a 800-MHz clock signal, and the OGWs operate at 200 MHz. The measured waveforms show successful chip-to-chip operations including the packet transaction from chip-A to B, SRAM access, token reservation, and returning the SRAM data to the chip-A.

VI. Conclusion

An architecture for a packet-switched OCN is proposed, its cost optimization is shown, and implementation issues on building block design are addressed. Clocking, packet routing, and chip-to-chip communication schemes are proposed. For cost-effective OCN architecture, Star topology with serialized link is used. For source synchronous packet transmission, matched-delay architecture is proposed. Switch type and its feasibility are also analyzed. The 800-MHz test chip is fabricated using 0.38-μm CMOS technology, and successfully demonstrates feasibility of the usage of OCN to implement high performance SoC. This architecture can be used as an effective integration platform for heterogeneous PUs.

References

[1] R. Woo *et al.*, "A 210 mW graphics LSI implementing full 3 D pipeline with 264 Mtexels/s texturing for mobile multimedia applications," in *ISSCC Tech. Dig.*, 2003, pp. 44–45.

[2] AMBA™ AXI Protocol Specification (2003). [Online]. Available: http://www.arm.com

[3] M. Sgroi *et al.*, "Addressing the system-on-a-chip interconnection woes through communication-based design," in *Proc. Design Automation Conf.*, 2001, pp. 667–672.

[4] L. Benini *et al.*, "Powering networks on chips," in *Proc. Int. Symp. System Synthesis*, 2001, pp. 33–38.

[5] P. Guerrier *et al.*, "A generic architecture for on-chip packet-switched interconnections," in *Proc. Design Automation Test Eur. Conf. Exhib.*, 2000, pp. 250–256.

[6] S. Kumar *et al.*, "A network on chip architecture and design methodology," in *Proc. Ann Symp. VLSI*, 2002, pp. 117–124.

[7] W. J. Dally *et al.*, "Route packets, not wires: On-chip interconnection networks," in *Proc. f Design Automation Conf.*, 2001, pp. 684–689.

[8] H. Zhang *et al.*, "A 1-V heterogeneous reconfigurable DSP IC for wireless baseband digital signal processing," *J. Solid-State Circuits*, vol. 35, no. 11, pp. 1697–2000, Nov. 2000.

[9] M. Taylor *et al.*, "A 16-issue multiple-program-counter microprocessor with point-to-point scalar operand network," in *ISSCC Tech. Dig.*, 2003, pp. 170–171.

[10] S. Lee *et al.*, "An 800 MHz star-connected on-chip network for application to systems on a chip," in *ISSCC Tech. Dig.*, 2003, pp. 468–469.

[11] K. Lee *et al.*, "A 51 mW 1.6 GHz on-chip network for low power heterogeneous SoC platform," in *ISSCC Tech. Dig.*, 2004, pp. 152–153.

[12] A. Laffely *et al.*, "Adaptive system on a chip (ASOC): A backbone for power-aware signal processing cores," in *Proc. IEEE Int. Conf. Image Processing*, Barcelona, Spain, Sep. 2003.

[13] E. Rijpkema *et al.*, "Tradeoffs in the design of a router with both guaranteed and best-effort services for networks on chip," in *Proc. Design Automation Test in Eur.*, Mar. 2003, pp. 350–355.

[14] D. Bertozzi *et al.*, "Xpipes: A network-on-chip architecture for gigascale system-on-chip," *IEEE Circuits Syst. Mag.*, vol. 4, no. 2, pp. 18–31, Feb. 2004.

[15] S. Kimura *et al.*, "An on-chip high speed serial communication method based on independent ring oscillators," in *ISSCC Tech. Dig.*, 2003, pp. 390–391.

[16] H. J. Siegel *et al.*, *Interconnection Networks for Large-Scale Parallel Processing: Theory and Case Studies*. New York: McGraw-Hill, 1990.

[17] K. Lee *et al.*, "SILENT: Serialized low energy transmission coding for on-chip interconnection networks," in *Proc. Int. Conf. Computer-Aided Design*, 2004, pp. 448–451.

[18] W. J. Dally *et al.*, *Digital Systems Engineering*. Cambridge, U.K.: Cambridge Univ. Press, 1998, ch. 10.

[19] M. Cooperman *et al.*, "CMOS gigabit-per-second switching," *J. Solid-State Circuits*, vol. 28, no. 6, pp. 631–639, Jun. 1993.

[20] T. T. Ye *et al.*, "Analysis of power consumption on switch fabrics in network routers," in *Proc. Design Automation Conf.*, 2002, pp. 524–529.

[21] K. Lee *et al.*, "A distributed on-chip crossbar switch scheduler for on-chip networks," in *Proc. Custom Integrated Circuits Conf.*, May 2003, pp. 671–674.