HIBS – Novel Inter-layer Bus Structure for Stacked Architectures

Masoud Daneshtalab, Masoumeh Ebrahimi, and Juha Plosila
Department of Information Technology, University of Turku, Turku, Finland
{masdan, masebr, juhplo}@utu.fi

Abstract- Three-dimensional integrated circuit (3D IC) technology has emerged as a viable candidate to overcome the interconnections scaling and integration complexity in next generation digital system designs. In addition, combining the benefits of 3D ICs and Networks-on-Chip (NoCs) schemes provides a significant performance gain for 3D architectures. In recent years, through-silicon-via (TSV), employed for inter-layer connectivity (vertical channel), has attracted a lot of interest since it enables faster and more power efficient inter-layer communication across multiple stacked layers. The bus-based organization, a hybrid between packet-switched network and a bus, is a dominant architecture for utilizing TSVs as inter-layer communication channel in 3D architectures. In this paper, we propose a novel bus structure for inter-layer communication to improve the performance by reducing the delay and complexity of traditional bus arbitration.

I. INTRODUCTION

Network-on-Chip (NoC) can address many of on-chip communication design issues such as the delay and power consumption of global interconnections of high performance Multi-Processor Systems-on-Chip (MPSoCs) [1][2]. However, two-dimensional (2D) chip fabrication technology is facing several challenges in the deep submicron regime even when utilizing NoC architectures [3], e.g. designing the clock-tree network for a large chip, limited floor-planning choices, increasing the wire delay and power consumption, integrating diversity components that are digital, analog, MEMS RF, etc. 3D ICs can deal with these problems due to reducing the interconnect delay (shorter global interconnects) by stacking vertically active silicon layers as well as lower interconnect power consumption caused by wire-length reduction.

In this paper we focused on wafer stacking technology where dies are vertically stacked and Through Silicon Vias (TSVs) are exploited for Inter-layer communication. The distance between wafers can range from 5µm to 50µm [6], which is much shorter than the wire length between cores on a tier, and the pitches of TSVs can range from 1µm to 10µm square [6]. That is, the wire delay, power consumption and chip form factor are significantly reduced [8][11].

Combining the benefits of 3D ICs and Networks-on-Chip (NoCs) provides a significant performance gain for 3D architectures. 3D NoC topologies not only enable scalable networks to provide communication requirements in 3D ICs [3]-[6] but also are a crucial factor of 3D chips in terms of performance, cost, and energy consumption [3]. Various on-chip network topologies have been studied for 3D NoCs [3]-[7][8][9][12]. 3D-Symmetric NoC and 3D NoC-Bus Hybrid (stacked mesh) structures are popularly used in 3D systems, because their grid-based regular structure is intuitively considered to match the 2D VLSI layout for each layer [3][4][5][6][8]. The 3D-symmetric NoC structure is an extension of 2D mesh by adding two additional physical ports to each baseline-router (one for up and one for down) in the popular 2D mesh-based system [3][6]. Adding two additional ports requires larger crossbar incurring significant area and power overhead and increases the blocking probability occurring inside the router.

Since TSVs are shorter and wider than intra-layer interconnects, they have lower resistance and can support higher signaling speeds [6][12]. As router latencies may dominate the fast vertical interconnects, this has led the researchers to propose 3D NoC-Bus Hybrid structures using a bus with a centralized arbiter for each vertical channel, which allows single hop latency for packets between any layers [3]-[6]. On-chip routers in this structure have at most 6 ports, one to the IP-core, one to the bus, and four for cardinal directions. According to [6] the 3D-hybrid structure was observed to be better than the 3D symmetric for the vertical interconnection as long as the number of device layers was less than 9. This has motivated us to propose an efficient bus structure to overcome drawbacks of the conventional bus that has been employed for inter-layer communication. In order to reduce the delay and complexity of conventional buses, a novel bus structure, High-performance Inter-layer Bus Structure (HIBS), is presented for inter-layer communications. HIBS allows simultaneous transmissions without using centralized bus arbitration, which considerably reduces arbitration complexity and improves bandwidth. In the

presented bus structure, a single blocked packet might obstruct subsequent packet transmissions so that increasing the communication latency. To solve this problem, a non-blocking mechanism is introduced for the proposed bus structure. In addition, the presented bus employs bi-synchronous FIFO for synchronization between stacked layers, if each layer is fabricated by different technologies.

The rest of the paper is organized as follows. In Section II, a brief review of related works is discussed and the proposed bus structure is described in Section III. The experimental results are discussed in Section IV while the summary and conclusion are given in the last section.

II. RELATED WORK

Design techniques and methodologies for 3D architectures have been investigated to efficiently exploit the benefits of 3D technologies. Several NoC topologies for 3D systems have been exhaustively investigated in [3]-[6][9][10][13][15]. The authors in [3] demonstrate that besides reducing the footprint in a fabricated design, 3D systems provide a better performance compared to traditional 2D systems. They have also demonstrated that both mesh and tree topologies for 3D systems achieve better performance compared to traditional 2D systems. However, the mesh topology shows significant performance gains in terms of throughput, average latency, and energy dissipation with a small area overhead [3]. In [13] different 3D mesh-based architectures have been compared in the zero-load latency, but the performance of the network with different traffic patterns and loads is also necessary to be evaluated.

To construct an optimistic 3D mesh-based system, several 3D structures have been presented. Baseline-routers in 2D mesh-based systems have 5 ports, i.e. 4 ports to adjacent routers and one for the resource node. The straightforward extension for 3D mesh-based systems (*3D-symmetric NoC*) is to utilize routers with two additional inter-layer links by adding two physical ports to baseline-routers (one for up and one for down) [3][6][8][9][10]. As mentioned earlier, the 3D structure using such routers, not only increases the area and power overhead of the routers but also contention in the routers may arise. The electrical behavior of the relatively short and wide TSV, i.e. the low resistance, and supporting much higher signaling speeds led the authors of [6] to propose the 3D-hybrid structure. This 3D structure exploits the Dynamic Time Division Multiple Access (dTDMA) bus [14] with a centralized arbiter for the vertical communication link. Thus, moving from one layer to any of the other layers takes only one hop. However, contention issues in the bus limit the attainable performance gains [3]. That is, such structures inherently suffer from the limitation of buses since only one transmission is allowed each time over a vertical bus.

In [5], the DimDe router for 3D architectures has been proposed. The presented router uses a full 3D crossbar and a simple bus structure spanning all layers of the chip and fusing them into a single router entity. This router can minimize vertical traversal to one hop between any layers, but requires huge number of vertical connections and significantly complicates the control and arbiter of the router.

A multilayered 3D router architecture, named MIRA, is introduced for 3D systems by D. Park et al [4]. The router components are classified as separable components (buffers, crossbar, and inter-router links) and non-separable components (arbiter and routing modules). The separable components are laid out across multiple layers to save chip area and reduce power by dynamically shutting down some inactive layers. However, such routers are too aggressive in the current technology [16]. To reduce the area footprint of TSVs, a serialization scheme for vertical channels has been presented in [12], but this scheme increases the latency of inter-layer communications which can degrade the system performance.

Due to the above concerns, in this paper, we have focused on both the 3D-symmetric structure (7-port switch design) and the 3D-hybrid structure (bus-based vertical interconnect). As described in [9], the 3D-hybrid structure is shown to perform the worst compared to the other structures in terms of scalability under local traffic. Although shown to be weak in [9][16], the bus may be appropriated for hot spot traffic injection where many packets may need to be sent through several layers to a hot spot frequently. This may be akin to a processor on one layer, and a memory stack directly above it. Hence, in 3D architectures, the 3D-hybrid structure performance degrades as the number of layers and number of processing nodes increase [9], thereby the 3D-symmetric structure is more feasible, mature, and more efficient than the 3D-hybrid structure as network size increases [17]. In this work, we present a streamlined bus structure which can be utilized in both Hybrid and Symmetric NoC structures with considerable performance gain and lower hardware cost.

III. HIGH-PERFORMANCE INTER-LAYER BUS STRUCTURE (HIBS)

Traditionally, a bus is described as a shared link which can be owned by one attached subsystem at a time. Parallelism can be added to the structure by partitioning the bus into segments with bridges and allowing these segments to operate concurrently [18]. However, on one side, the overall system performance in such designs is still limited by the lack of parallel bus transactions, and on the other hand, because of using many control wires for the central arbitration in such segmented buses, it is not a suitable approach for vertical bus in 3D ICs.

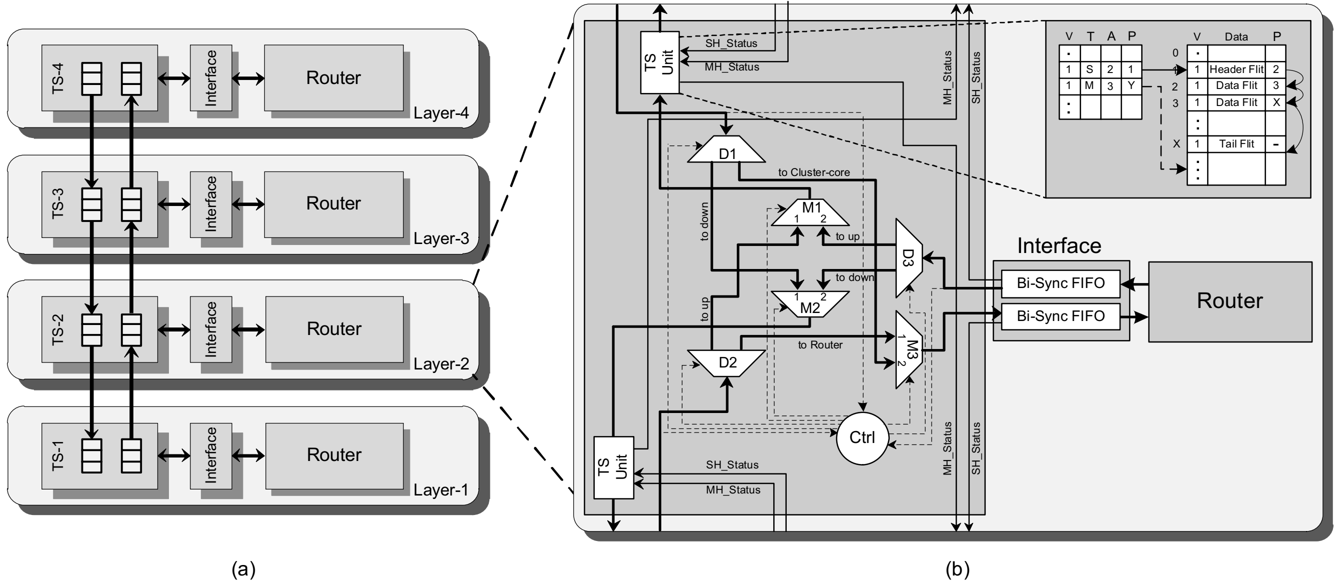

Fig. 1. (a) Proposed bus architecture and (b) the micro-architecture of the transfer stage.

Our solution for these bottlenecks for vertical buses is to consider the system bus with a bidirectional pipeline which is capable of transferring data concurrently from one or more sources to several destinations. As the proposed architecture is illustrated in Fig. 1(a), the system is partitioned into a set of modules each of which is used to connect the corresponding layer to the pipeline bus. As the system is based on GALS design paradigm, the layers can internally operate at different clock frequencies. The layers are independent of each other, in case there are some inter-layer transactions, the layers exchange data synchronously or asynchronously through the pipelined system bus, a segmented communication link which allows simultaneous transfer in both directions. The layers can concurrently access the bus without waiting for any grant signals, because of the pipelined structure of the proposed bus architecture. The interface module acts as a synchronizer between the router and the pipeline bus. To construct the pipelined bus, the physical wires that implement the bus are divided into a set of segments separated from each other by Transfer Stages (TS), one attached to each layer (Fig. 1(a)). Each transfer stage contains internal FIFO queues for pipelining the data flow, and a bus segment between adjacent stages consists of two separate unidirectional point-to-point interconnects which transfer data synchronously (or asynchronously) between the stages in opposite directions. These two links of a segment can operate in parallel, and due to pipelining, all segments of the bus can transfer data simultaneously. Each layer has a unique address for inter-layer communication. Furthermore, each IP-core/memory in a layer has its own address which makes addressing of a specific module in a given layer possible. Hence, a datagram propagating along the bus has a header containing both the layer address and the IP-core/memory address. The former is analyzed at each transfer stage, and the latter is decoded by routers in each layer. The micro-architecture of the transfer stage is illustrated in Fig. 1(b) where it includes two identical pipelines transferring data to the opposite directions. Each pipeline contains multiple slots to pipeline packets between slots. Apart from the pipelines the interface contains FIFO queues used as input and output buffers of the host port. Their capacity has to be chosen according to the speed of the bus interface and the estimated data rate of the attached router. Each transfer stage also contains three multiplexers (M1, M2 and M3) and three de-multiplexers (D1, D2 and D3) to establish a communication between inputs and outputs of the transfer stage. On top of that, each transfer stage has the following functions:

1- It forwards incoming packets from the preceding stage to the next stage through a buffer, in both directions. That is, if the incoming packet from the upper stage (lower stage) is intended to be forwarded to the lower stage (upper stage), D1 and M2 (D2 and M1) will provide the required connections.

2- If an incoming packet from an adjacent transfer stage is intended to be processed by the router, the transfer stage delivers the packet to the interface module of the layer through a FIFO queue, i.e. D1 and M3 (D2 and M3) establishes the required connections to deliver the incoming packet from the upper stage (lower stage) to the router.

3- When a data is sent to another layer, the transfer stage operates as an output buffer. This means that it takes care of first receiving data from the interface module of the attached layer through a FIFO queue and then sending this data to one of the two adjacent transfer stages, depending on the direction in

which, the target layer is located. Namely, D3 and M1 (D3 and M2) will be responsible for the required connections when the router decides to send a packet to upper layer (lower layer). When a packet arrives at a transfer stage, the header flit is sent to the controller unit to determine in which direction the packet should be sent. Based on the controller decision, it will be either forwarded to the next stage or transferred to the host router via the interface. Also, an arbitration in the controller module has to be performed to prevent the two parallel operating pipelines from writing simultaneously to the FIFO in the interface. In addition, because the electrical behavior of short and wide TSVs provides much higher signaling speeds, the credit-based flow control [1] has been implemented for the transmission protocol on a segment between transfer stages.

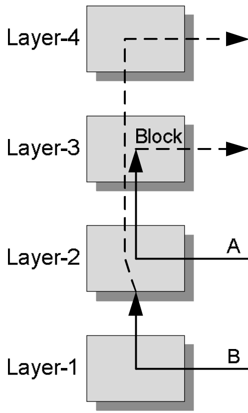

Fig. 2. A blocking situation.

A. Non-blocking Scheme

The proposed pipeline structure allows simultaneous transmissions without using centralized bus arbitration, which considerably reduces arbitration complexity and improves bandwidth. However in the pipeline bus architecture, a single blocked packet might obstruct the subsequent packets so that increasing the communication latency. A blocking situation is shown in Fig. 2 where the packet A and packet B destined for the layer 3 and layer 4, respectively. When the packet A is blocked, the packet B can be obstructed behind the packet A. In order to prevent this blocking situation, we introduce two types of packet, Single-Hop (SH) packet and Multiple-Hop (MH) packet. SH packet destinations are located in one of the neighboring layers while MH packets require passing several layers. However, a MH packet changes its type to SH once the destination is one layer away. In each transfer stage, the incoming packets are de-multiplexed into two separate paths: one path delivers the SH packets to the interface (SH path) while the other path forwards the MH packets to the next stage (MH path). That is, packets are de-multiplexed to either the TS unit or interface. The point is that if one of these two paths gets blocked, the remaining flows from the upstream stage cannot pass through the other path if it is idle. This blocking probability can be considerably reduced by considerably increasing the size of both the interface and TS unit buffers, which is an expensive solution for such systems. The idea is to reduce the blocking probability with a low hardware cost. Therefore, each stage adopts congestion condition of its downstream transfer stage buffers (interface and TS unit buffers) so that it can decide to deliver a packet to the less congested path of the next stage. The congestion condition of SH and MH paths, indicating the stress value of the interface and TS unit buffers, can be transmitted from one layer to another through two separate inter-layer signals (MH_Status and SH_Status). These signals are employed by the TS unit of the upstream transfer stage to forward a non-blocking packet, i.e. send a MH packet if the SH path is congested/blocked or send a SH packet if the MH path is congested/blocked.

As depicted in Fig. 1(b), each TS unit is composed of a table and a buffer. Each row of the table corresponds to a packet and includes a valid tag (v), a packet Type (T), a packet age (A) and a header pointer (P). In the buffer, the flits of each packet are stored with a linked list structure providing high resource efficiency with a little hardware overhead. Fig. 1(b) exhibits a pointer field adopted to indicate the next flit position in the buffer. As multiple packets might be stored in the buffer, an arbitration mechanism is needed to determine which packet is allowed to be transmitted. The TS unit arbitration decision is based on the stress value of MH and SH paths. This arbiter selects the oldest packet (highest age) requesting an available path with the lowest stress value. Afterward, the age value of each packet having the same type as the selected one is increased by the arbiter to prevent starvation.

IV. Experimental Results

In this section, in order to explore the impact of the presented bus, we compare the presented structure with the conventional buses in terms of performance, hardware cost, and temperature. Hence, a cycle-accurate 3D NoC simulation environment is developed to assess the efficiency of the proposed structure. It accurately models the routers and the interconnection links as well as all the architectural features of the various inter-layer bus architectures.

The simulator inputs include the array size, the router operation frequency, the routing algorithm, the link width length, and the traffic type. The experiments are performed on a 64-node (4×4×4) 3D stacked architecture using wormhole switching with a constant packet size of 8 flits while we use the XYZ deterministic routing algorithm in all experiments. As mentioned earlier we consider the hybrid scheme as it is more

efficient than the symmetric scheme [6][12]. The hybrid network is formed by a typical state-of-the-art router structure including input buffers, a routing unit, a switch allocator, and a crossbar as well as an interface unit connecting the router to a vertical channel (bus) interface. For all routers, the data width is set to 32 bits (the maximum bandwidth at each link is 1 flit per cycle) and each input channel has a buffer (FIFO) size of 12 flits. As mentioned earlier, to compensate the performance loss due to using the bus as the vertical interconnect, each vertical channel is composed of two unidirectional channels in opposite directions to propagate the inter-layer data. Thus, 32 bits of the channel is allocated to upward direction and the other 32 bits of the channel is employed for the downward direction. Each channel has its arbiter module and bus controller [6][14]. The depth of buffers in the transfer stage is 5 flits with the congestion thresholds at 80% of the total buffer capacity. Two synthetic traffic profiles including uniform and hotspot, along with SPLASH-2 [19] application traces are used to evaluate the HIBS scheme. Two conventional bus structures, dTDMA [6][14] and SAMBA [18], are considered as baselines for inter-layer communication. As performance metrics, we use throughput and average delay. In SAMBA, multiple compatible bus transactions can be performed simultaneously without introducing additional arbitration complexity while dTDMA is a transaction-less, time-division-based bus architecture with dynamic timeslot allocation. However, both of them utilize a centralized bus arbiter module. Throughput is measured as the fraction of the maximum load that the network is capable of physically handling. Latency defined as the number of cycles between the initiation of a packet transmission issued by a source node and the time when the packet is completely delivered to the destination node.

Three different stacked configurations are considered for experiments: 3D-hybrid HIBS, 3D-hybrid SAMBA, and 3D-hybrid dTDMA, i.e. each configuration is based on the 3D NoC-Bus Hybrid but with different inter-layer communication structures.

A. Performance Evaluation

For each traffic profile, the average communication latency and throughput with various packet injection rates are computed for each configuration.

Uniform Traffic Profile:

In uniform traffic, a node sends the packet to each other node with the same probability [20]. Fig. 3 shows the results obtained from the network under the uniform traffic profile. As can be seen from the results, the configuration using HIBS outperforms the other schemes because of the following reasons. First, this is achieved due to having a small local arbiter in the transfer stage such that the arbitration delay is reduced significantly. Second, since the non-blocking approach of HIBS can reduce the blocking probability, as illustrated in Fig. 3(b), the throughput of the corresponding configuration is considerably higher than the configurations exploiting SAMBA and dTDMA. Decreasing the blocking rate is obviously beneficial to diminish the delay between layers.

Hotspot Traffic Profile:

Hotspot is considered to be a more realistic traffic profile since processors often communicate with a part of the total number of other processors like memory modules. In the hotspot traffic pattern, one or more nodes are designated as hotspot nodes receiving an extra portion of the traffic in addition to the regular uniform traffic. Newly generated packets are directed to each hotspot node with an additional *H* percent probability. We simulate hotspot traffic with four hotspot nodes. Four hotspot nodes are chosen at the center of the mesh, (2, 2, 1), (3, 3, 2), (2, 3, 3), (3, 2, 4), with equal probability of H=20%. The average latency and throughput results are illustrated in Fig. 4. As the figures show, the configuration using HIBS achieves both higher throughput and lower latency.

Table 1. System configuration parameters.

8 SPARC cores, issue width=1
L1 cache: private, split instruction and data cache, each cache is 16KB. 4-way associative, 64-bit line, 3-cycle access time.
L2 cache: shared, distributed in 3 layers, unified 28MB (28 banks, each 1MB). 64-bit line, 6-cycle access time, SNUCA MESI protocol.
Main memory: 4GB DRAM, 260 cycles, 16 outstanding.
Router: wormhole, 2 cycles, 32-bit flit

Application Traffic Profile:

In order to know the real impact of the proposed inter-layer scheme, we used traces from some application benchmark suites selected from SPLASH-2. Traces are generated using the GEMS simulator [21]. We used the LU, Radix, Ocean, and FFT applications from SPALSH-2 for our simulation. Table 1 summarizes our full system configuration where the cache coherence protocol is MESI [22] and access latency to the L2 cache is derived from the CACTI [23]. We form a 36-node on-chip network (3×3×4) that four layers are stacked on top of each other, i.e. out of the 36 nodes, 8 nodes are processors and other 28 nodes are L2 caches. L2 caches are distributed in the bottom three layers, while all the processors are placed in the top layer close to a heat sink so that the best heat dissipation capability is achieved [4][7]. The simulator produces, as output, the communication latency and throughput for cache access. Fig. 5 shows the average network latency and throughput of the real workload traces collected from the aforementioned system configurations.

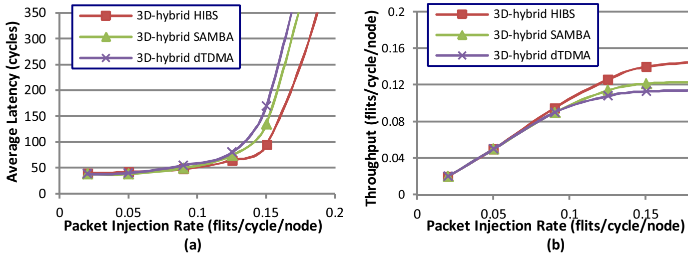

Fig. 3. (a) average latency and (b) throughput under the uniform traffic profile.

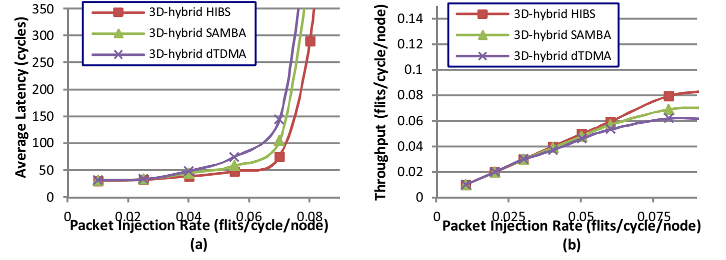

Fig. 4. (a) average latency and (b) throughput under the hotspot traffic profile.

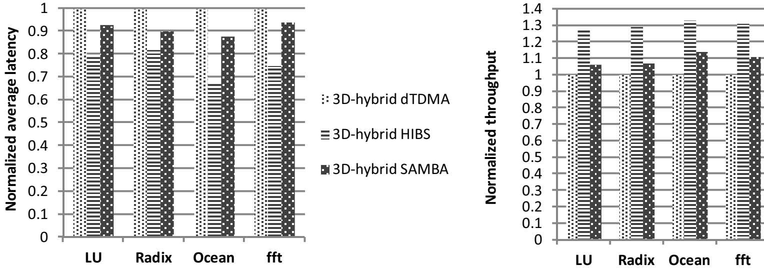

Fig. 5. Average latency and throughput for SPLASH-2 workloads normalized to the configuration using dTDMA.

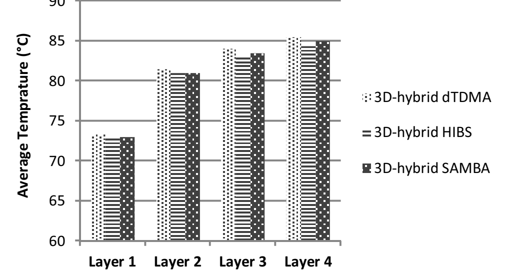

Fig. 6. Layer temperature profiles under the fft application profile.

We can see that HIBS-based configuration consistently reduces the average network latency across all tested benchmarks. It shows a steady reduction amount: 18%~32% (HIBS/dTDMA) and 8%~22% (HIBS/SAMBA) with the average of 24% 16.7%, respectively. Moreover, the throughput growths from these benchmarks are also significant. The results are 27~33% and 16%~21% for HIBS/dTDMA and HIBS/SAMBA, respectively.

B. Physical Analysis

The number of routers and vertical channels in a chip affects the area and implementation cost. Thus, to compute the network area for each configuration, we estimate the area of routers, vertical channels, and bus interfaces. The network platform of each configuration is synthesized using Synopsys Design Compiler with the UMC 0.09µm technology, while the backend is performed with the Cadence Encounter tool. Depending on the technology and manufacturing process, the pitch of TSVs can range from 1µm to 10µm [6][7][11]. In this work, the pad size for TSVs is assumed to be 5µm square with pitch of around 8µ, the flit-width is set to 32 bits, and each vertical channel requires 3×14 control wires for arbitration in four-stacked layers [6]. Each inter-layer connection, composed of two 32-bit vertical unidirectional channels, of dTDMA amd SAMBA buses, occupies about 6400µm^2 and 6550µm^2, respectively, while the required area for each inter-layer connection using HIBS is 4096µm^2. This shows that the proposed scheme can save more than 35% of the TSV area footprint with a large performance gain. Hence, after considering the TSV footprint area, the hardware overhead of the network using HIBS is approximately 16% and 14% less than that of networks employing the conventional dTDMA and SAMBA buses, respectively.

C. Thermal Analysis

To analysis the impact of the proposed inter-layer bus scheme on the chip temperature of a 36-node on-chip network (3×3×4), we use the Hotspot simulator [24] using the complete layout of the entire chip along with the power numbers for each component. We use the average power consumption, including leakage, area and the floorplan from the SPARC Niagara design (90nm) [25] for the processors while power numbers for the cache memories are obtained from CACTI. For the power consumption of the network, during network simulation, the power trace is obtained from Orion [26] under the *fft* application profile. The results of the thermal simulations for different configurations are shown in Fig. 6. In this figure, we show the average of the steady state minimum and maximum temperatures of each layer where *Layer 1* is considered to be the one which is near the heat sink in our thermal model. According to the results, all configurations have almost the same temperature values showing that the presented scheme is not increasing the chip temperature compared to the conversational schemes. However, it also makes a small temperature improvement.

V. Conclusion

In this paper, a novel bus structure for vertical channels is introduced not only to mitigate the drawbacks of existing bus

structures in terms of delay and throughput, but also to reduce the number of required inter-layer arbiter control signals. Experimental results with synthetic and real benchmarks revealed that the on-chip network formed by the presented bus structure reduces the hardware cost with significant performance gain.

VI. ACKNOWLEDGMENTS

The authors wish to acknowledge the Academy of Finland and Ulla Tuominen Foundation for the partial financial support during the course of this research.

References

[1] A. Jantsch et al., Networks on Chip. New York: Kluwer, 2003.
[2] M. Daneshtalab, M. Ebrahimi, P. Liljeberg, Juha Plosila, and H. Tenhunen, "A Low-Latency and Memory-Efficient On-hip Network," in Proceedings of 4th IEEE/ACM International Symposium on Network-on-Chip (NOCS), pp. 99-106, May 2010, France.
[3] B. S. Feero, P. P. Pande, "Networks-on-Chip in a Three-Dimensional Environment: A Performance Evaluation," IEEE Transactions on Computers, vol. 58, no. 1, pp. 32-45, Jan. 2009.
[4] D. Park, et al., "MIRA: A Multi-Layered On-Chip Interconnect Router Architecture", ISCA 2008, pp. 251-261, Pennsylvania State, USA.
[5] J. Kim, et al., "A novel dimensionally-decomposed router for on chip communication in 3D architectures," in Proc. of the ISCA, pp. 138-149, Boston, USA, 2007.
[6] F. Li, et al., "Design and Management of 3D Chip Multiprocessors Using Network-in-Memory," In 33rd International Symposium on Computer Architecture (ISCA), pp. 130–141, 2006.
[7] I. Loi and L. Benini, "An Efficient Distributed Memory Interface for Many-Core Platform with 3D Stacked DRAM," in Proc. of the DATE Conference, Germany, pp. 99-104, 2010.
[8] I. Loi, et al., "Supporting vertical links for 3D networks on chip: toward an automated design and analysis flow," in Proc. Nanonets, 2007.
[9] A. Y. Weldezion, et al., "Scalability of the Network-on-Chip communication architecture for 3D meshes," In Proc. of International Symposium on Networks-on-Chip (NoCS), pp. 114-123, 2009.
[10] M. Ebrahimi, et al., "Exploring Partitioning Methods for 3D Networks-on-Chip Utilizing Adaptive Routing Model," in Proceedings of 5th ACM/IEEE International Symposium on Networks-on-Chip (NOCS), pp. 73-80, May 2011, USA.
[11] I. Savidis, et al., "Electrical modeling and characterization of through-silicon vias (TSVs) for 3D integrated circuits," Microelectronics Journal, Vol. 41(1), pp. 9-16, 2010.
[12] S. Pasricha, "Exploring Serial Vertical Interconnects for 3D ICs," in Proc. IEEE/ACM DAC, pp. 581-586, 2009.
[13] V.F. Pavlidis and E.G. Friedman, "3D Topologies for Networks-on-Chip," IEEE Transactions on VLSI, 15(10):1081, 2007.
[14] T. Richardson, et al., "A hybrid SoC interconnect with dynamic TDMA-based transaction-less buses and on-chip networks", In Proc. VLSID, pp. 8-15, 2006.
[15] M. Daneshtalab, M. Ebrahimi, P. Liljeberg, J. Plosila, and H. Tenhunen, "High-Performance On-Chip Network Platform for Memory-on-Processor Architectures," in Proceedings of IEEE International Symposium on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC), pp. 1-6, June 2011, France.
[16] Y. Qian, Z. Lu and W. Dou, "From 2D to 3D NoCs: A Case Study on Worst-Case Communication Performance," In. Proc. of the International Conference on Computer-Aided Design (ICCAD), pp. 555-562, 2009.
[17] A. Y. Weldezion, et al., "3D Memory Organization and Performance Analysis for Multi-processor Network-On-Chip Architecture," in Proc. of IEEE International 3D System Integration Conference, USA, 2009.
[18] R. Lu, A. Cao, and C. Koh, "SAMBA-Bus: A High Performance Bus Architecture for System- on-Chips", IEEE Transactions on VLSI Systems, Vol. 15, Issue 1, pp. 69-79, Jan 2007.
[19] S.C. Woo, et al., "The splash-2 programs: Characterization and methodological considerations," in Proc. ISCA, pp. 24-36, 1995.
[20] P. Lotfi-Kamran, et al., "BARP- A Dynamic Routing Protocol for Balanced Distribution of Traffic in NoCs to Avoid Congestion," in Proceedings of 11th IEEE/ACM Design, Automation, and Test in Europe (DATE), pp. 1408-1413, Mar 2008, Germany.
[21] M.M.K. Martin, et al. "Multifacet's general executiondriven multiprocessor simulator (GEMS) toolset," SIGARCH Computer Architecture News, v. 33, No. 4, pp.92-99, 2005.
[22] A. Patel and K. Ghose, "Energy-efficient mesi cache coherence with pro-active snoop filtering for multicore microprocessors," Proc. Low power electronics and design, pp. 247-252, 2008.
[23] N. Muralimanohar, et al., "Optimizing nuca organizations and wiring alternatives for large caches with cacti 6.0," In proc. 40th IEEE/ACM International Symposium on MICRO, pp. 3–14, 1-5 Dec. 2007.
[24] K. Skadron, et al., "Temperature-aware microarchitecture," in Proc. of ISCA, pp. 2-13, 2003.
[25] P. Kongetira, K. Aingaran, and K. Olukotun, "Niagara: a 32-way multithreaded Sparc processor," Micro, IEEE, vol. 25, pp. 21-29, 2005.
[26] A. Kahng, B. Li, L.-S. Peh, and K. Samadi, "Orion 2.0: A fast and accurate noc power and area model for early-stage design space exploration," in proc. of DATE, pp. 423-428, 2009.