3D Network-on-Chip with on-chip DRAM: an empirical analysis for future Chip Multiprocessor

Thomas Canhao Xu, *Member, IEEE,* Bo Yang, *Member, IEEE,* Alexander Wei Yin, *Member, IEEE,*
Pasi Liljeberg, *Member, IEEE,* and Hannu Tenhunen, *Member, IEEE*

Abstract—With the increasing number of on-chip components and the critical requirement for processing power, Chip Multiprocessor (CMP) has gained wide acceptance in both academia and industry during the last decade. However, the conventional bus-based on-chip communication schemes suffer from very high communication delay and low scalability in large scale systems. Network-on-Chip (NoC) has been proposed to solve the bottleneck of parallel on-chip communications by applying different network topologies which separate the communication phase from the computation phase. Observing that the memory bandwidth of the communication between on-chip components and off-chip memory has become a critical problem even in NoC based systems, in this paper, we propose a novel 3D NoC with on-chip Dynamic Random Access Memory (DRAM) in which different layers are dedicated to different functionalities such as processors, cache or memory. Results show that, by using our proposed architecture, average link utilization has reduced by 10.25% for SPLASH-2 workloads. Our proposed design costs 1.12% less execution cycles than the traditional design on average.

Keywords—3D Integration, Network-on-Chip, Memory-on-Chip, DRAM, Chip Multiprocessor.

I. INTRODUCTION

The concept of CMP enables to integrate more than one core on a single physical chip. Intel Pentium-D [1], one of the earliest manufactured CMP, has embedded two dies on a processor chip. The integration of more cores on a chip is under intensive research. AMD has announced its twelve-core x86 processor with two dies on a chip, each of which has six cores with an area of 346mm^2 [1]. It is predictable that in the near future, more and more cores will be integrated on a chip. However, the current communication schemes in CMPs are mainly based on the shared bus architecture which suffers from high communication delay and low scalability. Therefore, NoC has been proposed as a promising approach to integrate a large number of components on a single chip by leveraging the well developed computer network concepts [2]. In 2007, Intel has demonstrated an 80 tile, 100M transistor, 275mm^2 2D NoC prototype under 65nm processing technology [3]. An experimental CMP containing 48 x86 cores on a chip has been manufactured for research using 4×6 network-based 2D mesh topology with 2 cores per tile [4].

There is a great concern about memory bandwidth, in which the number of memory requests are growing with core

T.C. Xu is with the Turku Center for Computer Science (TUCS), Jouka-haisenkatu 3-5 B, 20520, Turku, Finland and Department of Information Technology, University of Turku, 20014, Turku, Finland, e-mail: canxu@utu.fi.

B. Yang, A.W. Yin, P. Liljeberg and H. Tenhunen are with TUCS and University of Turku.

[1] Intel and Pentium are trademarks or registered trademarks of Intel or its subsidiaries. Other names and brands may be claimed as the property of others.

TABLE I: Processor and memory bandwidth for one channel

Processor	Core	Typical memory	Typical BW
Pentium 3	1	PC-133 SDRAM	1.066 GB/s
Pentium 4	1/2	PC-1600 DDR	1.6 GB/s
Core 2 Duo	2	PC2-3200 DDR2	3.2 GB/s
Core 2 Quad	4	PC2-6400 DDR2	6.4 GB/s
Core i7 980X	6	PC3-8500 DDR3	8.5 GB/s

numbers. In the era of Pentium 3, the processor has only one core, memory bandwidth requirement is thus not so high. As the number of processor core grows, the requirement of memory bandwidth grows as well. As it is shown in Table I, Core 2 Duo doubles the requirement of memory bandwidth to fit the requests of two cores. The system performance will decline if memory bandwidth cannot sustain the rate requested by processor cores. By plugging two identical Dual In-line Memory Modules (DIMMs) on the motherboard, dual channel can be configured to provide double bandwidth. In the dual channel, data is transfered in a 128-bit flavor instead of conventional 64-bit in one cycle. Triple channel is introduced with Double-Data-Rate 3 Synchronous DRAM (DDR3 SDRAM) memory, providing 192-bit data transfer in a clock cycle. Configured with triple channel DDR3-8500 memory, the maximum theoretical memory bandwidth for Intel Core i7 980X is thus 25.6GB/s [5].

Increasing the memory bandwidth by using DDR4 seems to be a solution, quadrupling or even quintupling the number of memory channels is another solution. However, as mentioned earlier, triple channel configuration requires at least three DIMMs, which increases cost, fault rate and power consumption. Another constraint is the pin count limitation. It is predicted by the ITRS roadmap that pin count will increase by about 10% each year only, comparing with the number of cores that is expected to double every 18 months [6].

There have been several researches in the field of processor memory bandwidth. Brian M. Rogers et. al. [7] developed a mathematical model to evaluate the impact of memory bandwidth on CMP scaling in different technologies. However, the authors focus only on the theoretical studies in this work. In [8], the organization and performance of 3D memory in NoC are analyzed. They assumed a simple NoC model with uniform random traffic and local traffic. Gabriel H. Loh presented a novel 3D-stacked memory architecture for CMP [9]. It is claimed that a 1.75x speedup is achieved over previous approaches. Nevertheless the paper presumed a conservative quad-core configuration.

In our paper, however, we investigate the empirical design

of 3D NoC with memory on chip. By integrating the memory module on chip using 3D IC technology, overall system performance is expected to improve due to reduced latency and increased bandwidth. We model a 64-core 3D NoC with 3D on-chip DRAM memory, analysis the memory bandwidth and latency with different memory sub system implementations, present the performance with our proposed approach and traditional system using a full system simulator. To the best of our knowledge, this is the first paper about empirical study of stacked DRAM memory architecture for 3D NoC.

II. MODELING OF THE 3D NoC

NoC brings network communication methodologies into on-chip communication. Figure 1 shows a CMP with 4×4 mesh topology. The underlying network is comprised of network links and routers (R), each of which is connected to a processing element (PE) via a network interface (NI). Each PE is a core in the CMP. The basic architectural unit of a NoC is the tile/node (N) which is consisted of a router, its attached NI and PE, and the corresponding links. Communication among PEs is achieved via the transmission of network packets.
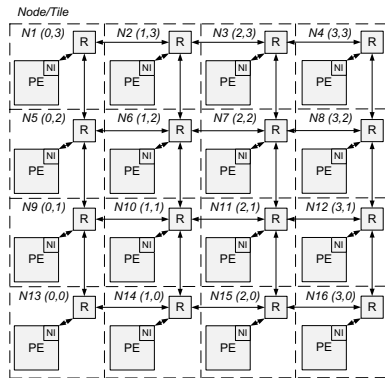

Fig. 1: An example of 4×4 NoC using mesh topology.

The interconnection of traditional 2D chip connection results in long global wire lengths, which further causes high delay, high power consumption and low performance [10]. This situation becomes worse in NoC, since usually a NoC has a larger number of processors compared with traditional CMPs. To solve this problem, 3D integration technology is introduced by stacking multiple dies vertically. Layers with different functions, e.g. processor layer, cache layer, controller layer and memory layer can be implemented in a 3D NoC.
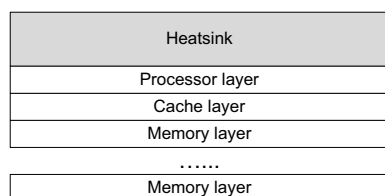

Fig. 2: Schematic diagram of a 3D chip with multiple stacked layers. The heatsink is attached with processor layer.

It is expected that since the processors consume overwhelming majority of power in a chip, stacking multiple processor layers could be unwise for heat dissipation. According to [11], heat dissipation is a major problem by stacking multiple processor layers even if processors are interlaced vertically. Without direct contact with heatsinks, the peak chip temperature of 3D design raises by 29°C comparing with the 2D design, which is unfeasible for some applications [11]. However, by stacking more memory layers instead of processor layers, the thermal constraint is supposed to be alleviated (Figure 2). Gian Luca Loi et. al. shows that, even for 18 stacked layers (1 of processors, 1 of cache and 16 of memory), the maximum temperature for a 3D chip increases only 10°C comparing with 2D chip [12]. It is estimated that 15% lower core frequency of a 3D chip could compensate the thermal drawback [12].

The floorplan of modern multi-core chips such as third-generation Sun SPARC [13], IBM Power 7 [14], AMD Istanbul [1] show the possibility of 3D NoC. The total area of Sun SPARC chip is 396mm^2 with 65nm fabrication technology. Scaled to 32nm technology, each core has an area of 3.4mm^2. We simulate the characteristics of a 64MB, 64 banks, 64-bit line size, 4-way associative, 32nm cache by CACTI [15]. Results show that the total area of cache banks is 204.33mm^2. Each cache bank, including data and tag, occupies 3.2mm^2. We also simulate the characteristics of a 1GB, 8 banks, 32nm DRAM memory by CACTI [15]. It is revealed that the total area of the memory is 212.79mm^2.
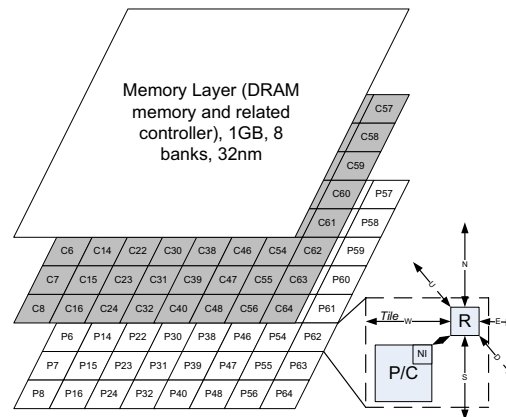

Fig. 3: 3D NoC with one processor layer (Px), one cache layer (Cx) and one memory layer, layers are fully connected by through silicon vias (TSVs, not shown in figure).

On account of the aforementioned analysis, we use a 3D NoC model based on 32nm fabrication technology, with one layer of processor, one layer of cache and several layers of memory. In consideration of heat dissipation, the processor layer should be close to the heatsink. The top layer is a 8×8 mesh of Sun SPARC cores. The cache layer has a 8×8 mesh of cache banks. It is noteworthy that routers are quite small compared with processors and cache banks, e.g. scaled to 32nm, as we calculated, a 7-port 3D router is estimated to be only 0.096mm^2. Furthermore, not all routers in a 3D NoC

require seven ports, e.g. router of P8 in Figure 3 has only East, North, Local PE and Up ports. The total area of the chip is supposed to be around 230mm². Figure 3 shows the above-mentioned 3D NoC with three layers, however more layers of memory can be stacked.

III. Analysis of the impact of on-chip DRAM memory to NoC

Traditional off-chip memory designs are shown in Figure 4. In Figure 4a, both the memory controller and the memory are off-chip with only one memory channel. This is a default configuration with early CMP systems. When reading data from or writing data to the memory, a transmission delay is incurred. The delay consists of two parts: the delay of processor and memory controller, and the delay of memory controller and DRAM module. For modern systems these delays are usually hundreds of cycles (e.g. 200-300). Figure 4b illustrates a CMP system with on-chip memory controller and dual channel DRAM memory. The latency between processor core and memory controller is reduced significantly while the memory bandwidth is doubled. By increase the number of memory controller, as shown in Figure 4c, the performance of memory sub system can be improved further, notwithstanding this configuration is used rarely due to pin count limitations.
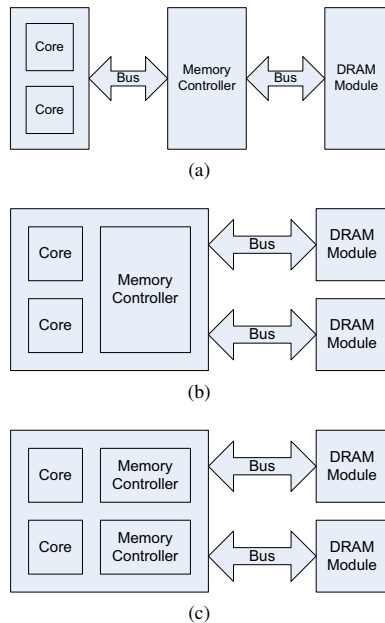

Fig. 4: Compare of different processor and memory sub system organization.

To analyze the effect of memory architecture to a NoC, we first consider a smaller 5×5 mesh with the focus on network latency which is one of the most important measuring factors for NoCs. A SystemC based cycle accurate NoC simulator Noxim [16] has been extended. We use workload trace of FFT from SPLASH-2 [17]. The trace has 2.11M packets, with 78.79M cycles executed. We gather the trace from Simics [18] configured as a 5×5 NoC. The NoC has 25 PEs, in which each

PE has a private L1 cache and a shared L2 cache, the two memory controllers are attached in the center. Other detailed parameters can be found in Table V.

TABLE II: Maximum network latency for a NoC node

64	36	45	31	40
44	32	25	39	54
45	98	83	48	28
43	50	46	51	66
45	56	62	44	65

Table II shows the maximum network latencies. Obviously, the two central nodes have the highest network latency (98 and 83, compared with 25 to 66 of other nodes), due to the concentrated memory traffic from all nodes. The performance of the NoC is degraded with higher latencies. The limitation of system scalability is consequently on the memory sub system. Memory-on-Chip is a feasible way to break the bottleneck of the memory sub system. In this paper, we explore the following approaches.

A. Memory data bus

A standard single-channel DDR2 SDRAM has a bus width of 8 bytes. Dual-channel technology utilizes two memory channels which result in a 16 bytes bus width, and double the memory bandwidth. Intel Core i7 brings triple-channel architecture, with 24 bytes bus width. It is noteworthy that pin-count grows with channel-count, 373 of 1366 pins in the Intel Core i7 processor are dedicated to one memory controller with three channels [19]. By taking the bus completely on-chip, a much wider bus, e.g. 64 bytes, with the same size of cache line, is possible and the bandwidth improves significantly.

B. Frequency of processor-memory bus

The frequency of off-chip memory bus is quite slow comparing with common processor frequencies. The bus is used for the communication between on-chip memory controller and memory. In the era of Intel Pentium, the frequency of the processor was 66 to 200 Mhz, and at the same time the SDRAM itself is 66 to 133 Mhz. However, the frequency of a modern processor could be over 3Ghz, while even with DDR2/3 SDRAM, the clock does not grow so much. The typical frequency of a DDR2/3 SDRAM is 100 to 266 Mhz (200 to 533 Mhz for DDR2 and 400 to 1066 Mhz for DDR3, due to dual/quadruple clock rate). Higher bus clock rate is not feasible due to power and signal noise limitations. It is possible to achieve core clock frequency for the processor-memory bus, with 3D stacked memory design. More than ten times of bus bandwidth is predicted.

C. Memory access latency (MAL)

By stacking multiple layers of DRAM onto a 3D chip, access latencies are expected to reduce due to shorter wire lengths. DRAM is organized into a grid of single-transistor bit-cells, and the grid is divided into rows and columns. On the higher level, a DRAM bank consists of the grid and accompanying logic. A DRAM rank consists of several banks.

When the memory controller accesses data in a DRAM, t_{RCD} (the number of clock cycles needed between a row address strobe and a column address strobe), t_{CAS} (the number of clock cycles needed to access a column) and t_{RP} (the number of clock cycles needed to precharge a row) are major factors. For a DDR-400 memory, the bus frequency is 200Mhz, each cycle takes 5ns, the typical number of clock cycles for t_{RCD}, t_{CAS} and t_{RP} are 3, 3 and 3 respectively (15ns each). By stacking the DRAM ranks in a 3D fashion, the length of internal buses and bitlines are reduced, and hence the access latencies of the memory are reduced.

Definition 1: $t_{MAL} = t_{Bus\_delay} + t_{DRAM\_delay}$

As aforementioned, the area of a 1GB DRAM module is about 212.79mm^2 under 32nm technology, therefore the length of a side for the square module is 14.58mm. Figure 5 depicts that a request traveling from module 1 to 4 in 2D off-chip memory will take at least 29.17mm of wire length, by going through module 2 and 3. In 3D on-chip memory, since the distance between stacked layers are so small, around $50\mu m$, the wire delay between multiple layers can be neglected. Researches have shown that based on this architecture, memory access time has improved by 32% [9].

The latency for an off-chip DRAM is typically 200-300 cycles. Assuming a 2Ghz processor, the time is 100-150ns. By bringing the DRAM on-chip, this latency can be reduced to a very small value, thus we ignore this latency. According to Definition 1, the total latency from memory controller to DRAM will be reduced from $(250+9\times10) = 340$ to $(0+9\times10\times0.68) = 61.2$.
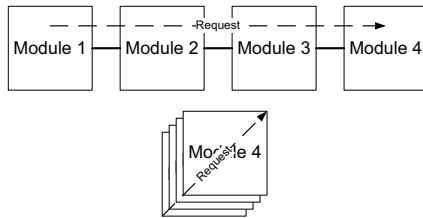

Fig. 5: 2D off-chip memory and 3D on-chip memory organizations.

D. Memory controllers

Many conventional architectures employ a limited number of memory controllers due to pin count limitations. With only one memory controller, 373 of 1366 pins in the Intel Core i7-900 processor are dedicated to that [19]. The ratio between core and memory controller is 6:1 (Table III). The Tilera Tile64 processor [20] implemented a 2D 8×8 mesh with four on-chip memory controllers and off-chip memory architecture. The ratio between core and memory controller is thus 16:1 (Table III). It is not realistic to have a memory controller for each PE in 2D architectures. However, for 3D stacked DRAM NoC, since die layers can be connected with layer-layer TSVs [21], one memory controller per core is feasible. The number of transistors required for a memory controller is quite small

compared with billions of total transistors for a chip. It is presented that a DDR2 memory controller is about 13,700 gates with application-specific integrated circuit (ASIC) and 920 slices with Xilinx Virtex-5 field-programmable gate array (FPGA) [22].

TABLE III: Comparison of processors with memory controller and memory channel

Processor	Core	Memory controllers	Channels
AMD MagnyCours	6	1 DDR3	2 (128bits)
Intel Nehalem	4	1 DDR3	3 (192bits)
IBM Power 7	8	2 DDR3	8 (512bits)
Tilera Tile64	64	4 DDR2	1 (64bits)

E. Mixing of all techniques

With higher bus frequency, wider bus width, shorter wire length and more memory controllers, memory bandwidth can be improved significantly.

Definition 2: Bandwidth = Clock × Data Rate × Rate Multiplier × Bus Width × Channel × Controller

The bandwidth of memory is defined in Definition 2. According to that, the bandwidth of a modern single-channel single-controller DDR2 memory with 200Mhz bus frequency is $200\times2\times2\times8\times1\times1 = 6.4$GB/s. By stacking the memory on-chip, with native clock rate of the core (2Ghz), 64-byte bus width and 64 memory controllers, the theoretical maximum bandwidth would reach: $2000\times1\times2\times64\times1\times64 = 16,384$GB/s! It is noteworthy that the memory runs in synchronous mode, i.e. the memory and the I/O bus are with the same frequency. We observed that the 3D stacked DRAM has a lower power consumption comparing with off-chip DRAMs, due to that 3D on-chip connection is much power efficient than off-chip bus I/Os. Higher frequencies can be achieved with lower power consumption, or lower frequencies for power constrained applications. Table IV shows the comparison of memory sub system of modern systems and our proposed system.

IV. EXPERIMENTAL EVALUATION

In this section, we present the experimental evaluation under different memory configurations. Applications are selected from SPLASH-2 [17].

A. 3D NoC Router and Routing Algorithm

As shown in Figure 1, routers in 2D NoCs have five ports to connect to five directions, namely, North, East, West, South and Local PE. For the vertical communication between different layers, routers in our 3D NoC model have two more ports and the corresponding virtual channels, buffers and crossbars to connect to the Up and Down pillars (Figure 3).

Adaptive routing is used widely in off-chip networks, however deterministic routing is favorable for on-chip networks because the implementation is easier. In this paper, a dimensional ordered routing (DOR) [23] based deterministic routing algorithm is selected and modified to fit the 3D topologies.

TABLE IV: Memory sub system configurations for different processors

Processor	Core	Typical Memory Configuration	Typical Memory BW	Memory BW per core	Memory Latency
AMD MagnyCours	6	1 DDR3, 533Mhz (133x4), 2 channels	17.1GB/s	2.85GB/s/core	250+(7-7-7)
Intel Nehalem	4	1 DDR3, 533Mhz (133x4), 3 channels	25.6GB/s	6.4GB/s/core	250+(7-7-7)
IBM Power 7	8	2 DDR3, 533Mhz (133x4), 8 channels	136.8GB/s	17.1GB/s/core	250+(7-7-7)
Tilera Tile64	64	4 DDR2, 200Mhz (100x2), 1 channel	12.8GB/s	0.2GB/s/core	250+(3-3-3)
Our proposed	**64**	**64 DDR, 2000Mhz sync., 1 channel**	**16,384GB/s**	**256GB/s/core**	**0+(2-2-2)**

When a node N_{source} sends a flit to a node $N_{destination}$, the flit will first travel along the X direction in N_{source} dimension until $Flit_x = Pillar_x$, then it will be routed in the Y direction. As long as the flit reaches the pillar, it will be vertically routed to the layer of the destination node. X-Y deterministic routing is used when the flit reaches the destination layer, in which a flit is first routed to the X direction and last to the Y direction.

B. Experiment Setup

The simulation platform is based on a cycle-accurate 3D NoC simulator which can produce detailed evaluation results. The platform models the routers, horizontal links and vertical pillars accurately. The state-of-the-art router in our platform includes a routing computation unit, a virtual channel allocator, a switch allocator, a crossbar switch and four input buffers. Deterministic routing algorithm has been selected to avoid deadlocks.

TABLE V: System configuration parameters

Processor configuration	
Instruction set architecture	SPARC
Number of processors	64
Issue width	1
Cache configuration	
L1 cache	Private, split instruction and data cache, each cache is 16KB. 4-way associative, 64-Byte line, 3-cycle access time
L2 cache	Shared, distributed in 64 nodes, unified 64MB (64 banks, each 1MB). 64-Byte line, 6-cycle access time
Cache coherence protocol	MOESI
Cache hierarchy	SNUCA
Memory configuration	
Size	4GB DRAM
Access latency	See Section III and Table IV
Requests per processor	16 outstanding
Network configuration	
Router scheme	Wormhole
Flit size	128 bits

We use a 128-node network which models a single-chip CMP for our experiments. The 3D architecture in this paper has one layer for processors, one layer for shared cache memories and five layers of DRAM memory (one layer for logic) (for simplicity, Figure 3 shows only three layers). A full system simulation environment with 64 processors and 64 L2 cache nodes has been implemented. The simulations are run on the Solaris 9 operating system based on SPARC instruction set in-order issue structure. Each processor is attached to a wormhole router and has a private write-back L1 cache. The L2 cache shared by all processors is split into banks. The size of each cache bank node is 1MB; hence the total size of shared L2 cache is 64MB. The simulated memory/cache architecture mimics SNUCA [24]. A two-level distributed directory cache coherence protocol called MOESI based on MESI [25] has been implemented in our memory hierarchy in which each L2 bank has its own directory. The protocol has five types of cache line status: Modified (M), Owned (O), Exclusive (E), Shared (S) and Invalid (I). Orion [26], a power simulator for interconnection networks, is used to evaluate detailed power characteristics. A wormhole router is modeled in Orion, with corresponding input/output ports, buffers and the crossbar. Power consumption of routers is analyzed. We use Simics [18] full system simulator as our simulation platform. The detailed configurations of processor, cache and memory configurations can be found in Table V.

C. Result Analysis

The normalized full system simulation results are shown in Figure 6 and 7. As is shown in Figure 6, our proposed design outperforms the traditional design in terms of average link utilization. Average link utilization is calculated with the number of flits transferred between NoC resources per cycle. Under the same configuration and workload, lower utilization means mitigated network load, which is favorable. Comparing with the traditional design, the average link utilization for our proposed design is reduced by 10.25%, on average. FFT and Cholesky have the most significant reduction of average link utilization, 13.34% and 12.81% respectively.
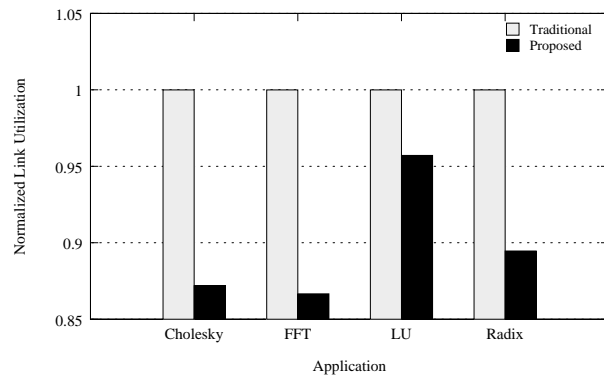

Fig. 6: Normalized average link utilization with different configurations.

The results in Figure 7 show that our proposed design outperforms the traditional design in terms of executed cycles under all workloads. On average, our proposed design costs 1.12% less cycles than the traditional design, and the cycle reduction reaches 2.29% for LU workload and 1.77% for Radix respectively. The improvements of executed cycles can

be interpreted as the result of the increased memory bandwidth and reduced memory access latency which commensurate with the number of memory accesses. The improvement of executed cycles is less remarkable comparing with average link utilization since local operations (e.g. core and cache) are not related with network operations.
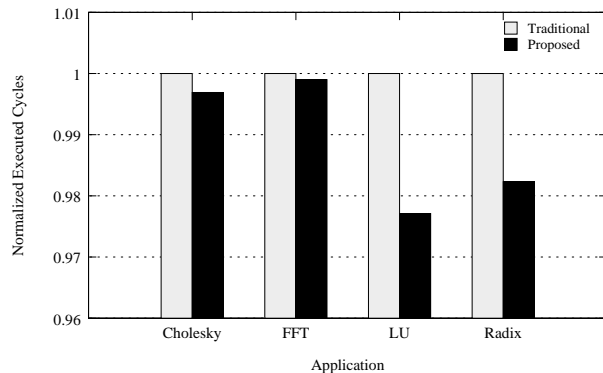

Fig. 7: Normalized executed cycles with different configurations.

V. CONCLUSION

In this paper, we demonstrate a novel 3D NoC architecture which targets at lower power consumption, lower communication delay, and higher system performance. Observing that current on-chip systems suffer from the critical memory bandwidth problems in the communications between on-chip components and off-chip memory, we propose a solution in which memories are integrated on chip by dedicating several 3D layers to on-chip DRAM. Besides, in the proposed architecture, there are two other layers which are dedicated to processors and cache, respectively. Considering the heat dissipation, the processor layer is placed near to the heatsink. In our experiments, we model a 3D NoC where four of SPLASH-2 applications are selected as synthetic benchmarks. Results of the experiments show that, on average, the average link utilization has reduced by 10.25% compared with traditional design. It is also observed that our proposed design costs 1.12% less execution cycles for the workloads.

ACKNOWLEDGMENT

REFERENCES

[1] AMD, "The amd opteron 6000 series platform," May 2010, http://www.amd.com/us/products/server/processors/6000-series-platform/pages/6000-series-platform.aspx.

[2] L. Benini and G. D. Micheli, "Networks on chips: A new soc paradigm," *IEEE Computer*, vol. 35, no. 1, pp. 70–78, January 2002.

[3] S. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz, D. Finan, P. Iyer, A. Singh, T. Jacob, S. Jain, S. Venkataraman, Y. Hoskote, and N. Borkar, "An 80-tile 1.28tflops network-on-chip in 65nm cmos," in *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, Feb. 2007, pp. 98–589.

[4] Intel, "Single-chip cloud computer," May 2010, http://techresearch.intel.com/articles/Tera-Scale/1826.htm.

[5] ——, "Intel core i7-980x processor extreme edition," May 2010, http://ark.intel.com/Product.aspx?id=47932.

[6] S. I. Association, "The international technology roadmap for semiconductors (itrs)," 2007, http://www.itrs.net/Links/2007ITRS/Home2007.htm.

[7] B. M. Rogers, A. Krishna, G. B. Bell, K. Vu, X. Jiang, and Y. Solihin, "Scaling the bandwidth wall: challenges in and avenues for cmp scaling," in *Proceedings of the 36th annual international symposium on Computer architecture*, June 2009, pp. 371–382.

[8] A. Weldezion, Z. Lu, R. Weerasekera, and H. Tenhunen, "3-d memory organization and performance analysis for multi-processor network-on-chip architecture," in *3D System Integration, 2009. 3DIC 2009. IEEE International Conference on*, 28-30 2009, pp. 1 –7.

[9] G. H. Loh, "3d-stacked memory architectures for multi-core processors," in *ISCA '08: Proceedings of the 35th Annual International Symposium on Computer Architecture*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 453–464.

[10] D. Sylvester and K. Keutzer, "Getting to the bottom of deep submicron," in *Computer-Aided Design, 1998. ICCAD 98. Digest of Technical Papers. 1998 IEEE/ACM International Conference on*, Nov 1998, pp. 203–211.

[11] T. C. Xu, A. W. Yin, P. Liljeberg, and H. Tenhunen, "A study of 3d network-on-chip design for data parallel h.264 coding," in *Proceedings of the 27th Norchip Conference*, November 2009.

[12] G. L. Loi, B. Agrawal, N. Srivastava, S.-C. Lin, T. Sherwood, and K. Banerjee, "A thermally-aware performance analysis of vertically integrated (3-d) processor-memory hierarchy," in *DAC '06: Proceedings of the 43rd annual Design Automation Conference*. New York, NY, USA: ACM, 2006, pp. 991–996.

[13] M. Tremblay and S. Chaudhry, "A third-generation 65nm 16-core 32-thread plus 32-scout-thread cmt sparc processor," in *ISSCC 2008*, February 2008, pp. 82–83.

[14] IBM, "Ibm power 7 processor," in *Hot chips 2009*, August 2009.

[15] T. Shyamkumar, M. Naveen, A. J. Ho, and J. N. P., "Cacti 5.1," HP Labs, Tech. Rep. HPL-2008-20.

[16] U. of Catania, "Noxim, an open network-on-chip simulator," http://noxim.sourceforge.net.

[17] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh, and A. Gupta, "The splash-2 programs: Characterization and methodological considerations," in *Proceedings of the 22nd International Symposium on Computer Architecture*, June 1995, pp. 24–36.

[18] P. Magnusson, M. Christensson, J. Eskilson, D. Forsgren, G. Hallberg, J. Hogberg, F. Larsson, A. Moestedt, and B. Werner, "Simics: A full system simulation platform," *Computer*, vol. 35, no. 2, pp. 50–58, February 2002.

[19] Intel, "Intel core i7 processor extreme edition and intel core i7 processor datasheet, volume 1," December 2008, http://download.intel.com/design/processor/datashts/320834.pdf.

[20] D. Wentzlaff, P. Griffin, H. Hoffmann, L. Bao, B. Edwards, C. Ramey, M. Mattina, C.-C. Miao, J. Brown, and A. Agarwal, "On-chip interconnection architecture of the tile processor," *Micro, IEEE*, vol. 27, no. 5, pp. 15 –31, sept.-oct. 2007.

[21] T. C. Xu, P. Liljeberg, and H. Tenhunen, "A study of through silicon via impact to 3d network-on-chip design," in *Proceedings of the 2010 International Conference on Electronics and Information Engineering (ICEIE 2010)*, August 2010.

[22] H. Global, "Ddr 2 memory controller ip core for fpga and asic," June 2010, http://www.hitechglobal.com/ipcores/ddr2controller.htm.

[23] H. Sullivan and T. R. Bashkow, "A large scale, homogeneous, fully distributed parallel machine," in *Proceedings of the 4th annual symposium on Computer architecture*, March 1977, pp. 105–117.

[24] C. Kim, D. Burger, and S. W. Keckler, "An adaptive, non-uniform cache structure for wire-delay dominated on-chip caches," in *ACM SIGPLAN*, October 2002, pp. 211–222.

[25] A. Patel and K. Ghose, "Energy-efficient mesi cache coherence with pro-active snoop filtering for multicore microprocessors," in *Proceeding of the thirteenth international symposium on Low power electronics and design*, August 2008, pp. 247–252.

[26] H.-S. Wang, X. Zhu, L.-S. Peh, and S. Malik, "Orion: a power-performance simulator for interconnection networks," in *Proceedings of the 35th Annual IEEE/ACM International Symposium on Microarchitecture*, November 2002, pp. 294–305.

Thomas Canhao Xu received his M.Eng. degree in Software Engineering from Zhejiang University, China in 2007. He has been teaching the National Certification of Information Engineer (NCIE) and Wish certified Network Engineer (WNE) for two and half years. He has authored 4 textbooks for WNE education. Since September 2008, he has been working in the Computer Systems laboratory, University of Turku as a researcher. He is also a Ph.D. student in the Turku Centre for Computer Science (TUCS), Turku, Finland. His research interests include software system support for network-on-chip platforms, system level 3D multiprocessor architecture design and software engineering.

Hannu Tenhunen received the Diplomas from Helsinki University of Technology, Finland, 1982 and Ph.D. from Cornell University, NY, 1986. In 1985, he joined Signal Processing Laboratory, Tampere University of Technology, Finland, as Associate Professor and later served as professor and department director. Since 1992, he has been with Professor in Royal Institute of Technology (KTH), Sweden where he also served as dean. Currently he is director of Turku Centre for Computer Science, Finland and at University of Turku. His current research interests are VLSI architectures and systems, especially Network-on-Chip systems. He has over 600 reviewed publications and 16 patents internationally.
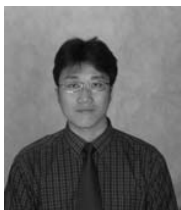
Bo Yang received his M.Sc. degree in Management from the Renmin University of China in 2006. He has carried out several information system projects for five years at China Aerospace Science & Industry Corporation. He has another five years' experience in the management field when he served as the head of HR department in aforementioned company. Since September 2008, he has been working in the Computer Systems laboratory, University of Turku as a researcher. His research interests include power-efficient on-chip application modeling and mapping, reconfigurable network-on-chip platforms.

Alexander Wei Yin received his M.Sc. degree in System-on-Chip design from the Royal Institute of Technology (KTH), Stockholm, Sweden in 2008. Since January 2008, he has been working in the Computer Systems laboratory, University of Turku as a researcher. He is also a Ph.D. student in the Turku Centre for Computer Science (TUCS), Turku, Finland. His research interests include low power techniques, fault tolerant designs and 3D integrated circuit architectures on network-on-chip platforms.

Pasi Liljeberg received his Ph.D. degree from University of Turku, Finland, in 2005. Since January 2010 he has been working in the Computer Systems laboratory as senior lecturer. During the period 2007-2009 he has worked as an Academy of Finland postdoctoral researcher. His current research interests include network-on-chip intelligent communication architectures, on-chip fault tolerant design, 3D multiprocessor system architectures, globally-asynchronous locally-synchronous platforms for nanoscale NoC and formal approaches in embedded system development. He has more than 60 international refereed papers. He has established and is leading a research group focusing on fault tolerant self-timed communication platform for nanoscale systems.