A Novel Dimensionally-Decomposed Router for On-Chip Communication in 3D Architectures [*]

Jongman Kim [†] Chrysostomos Nicopoulos [†] Dongkook Park [†] Reetuparna Das [†]
Yuan Xie [†] N. Vijaykrishnan [†] Mazin S. Yousif [‡] Chita R. Das [†]

[†]Dept. of CSE, The Pennsylvania State University
University Park, PA 16802
{jmkim,nicopoul,dpark,rdas,
yuanxie,vijay,das}@cse.psu.edu

[‡] Corporate Technology Group
Intel Corp.
Hillsboro, OR 97124
mazin.s.yousif@intel.com

ABSTRACT

Much like multi-storey buildings in densely packed metropolises, three-dimensional (3D) chip structures are envisioned as a viable solution to skyrocketing transistor densities and burgeoning die sizes in multi-core architectures. Partitioning a larger die into smaller segments and then stacking them in a 3D fashion can significantly reduce latency and energy consumption. Such benefits emanate from the notion that inter-wafer distances are negligible compared to intra-wafer distances. This attribute substantially reduces global wiring length in 3D chips. The work in this paper integrates the increasingly popular idea of packet-based Networks-on-Chip (NoC) into a 3D setting. While NoCs have been studied extensively in the 2D realm, the microarchitectural ramifications of moving into the third dimension have yet to be fully explored. This paper presents a detailed exploration of inter-strata communication architectures in 3D NoCs. Three design options are investigated; a simple bus-based inter-wafer connection, a hop-by-hop standard 3D design, and a full 3D crossbar implementation. In this context, we propose a novel partially-connected 3D crossbar structure, called the 3D Dimensionally-Decomposed (DimDe) Router, which provides a good tradeoff between circuit complexity and performance benefits. Simulation results using (a) a stand-alone cycle-accurate 3D NoC simulator running synthetic workloads, and (b) a hybrid 3D NoC/cache simulation environment running real commercial and scientific benchmarks, indicate that the proposed DimDe design provides latency and throughput improvements of over 20% on average over the other 3D architectures, while remaining within 5% of the full 3D crossbar performance. Furthermore, based on synthesized hardware implementations in 90 nm technology, the DimDe architecture outperforms all other designs − including the full 3D crossbar − by an average of 26% in terms of the Energy-Delay Product (EDP).

Categories and Subject Descriptors

B.4 [**Input/Output and Data Communications**]: Interconnections (Subsystems); B.8.2 [**Performance and Reliability**]: Performance Analysis and Design Aids.

General Terms

Design, Performance.

Keywords

Network-on-Chip (NoC), 3D Integration, 3D Architecture.

1. INTRODUCTION

Interconnects play a dominant role in shaping the power and performance profiles of processors designed using deep submicron technologies. The trend towards integrating multiple cores onto the same chip across a spectrum of devices − from high-end server chips to embedded cores − is further accentuating the importance of on-chip interconnect design. The lack of a good interconnect fabric can be envisioned to result in problems similar to traffic chaos in a large city without a proper roadway infrastructure. Technology scaling effects aggravate interconnect problems [1], especially those of global wires. While gate delays have reduced constantly, the increased resistance of the wires in newer technologies has increased global wire delays [25]. Consequently, wire delays have become quite significant, requiring multiple clock cycles for traversal across the edges of a microprocessor, and requiring architectural innovation such as Non-Uniform Cache Access (NUCA) architectures. Furthermore, signal integrity and reliability concerns such as inter-wire crosstalk and electromigration effects motivate the need for a structured design approach to the interconnect problem. The Network on-Chip (NoC) paradigm has been proposed as a scalable and structured approach for interconnect design [16, 10]. The design of 2D on-chip interconnects has been examined from various aspects, such as performance, power and reliability [30, 36, 24, 31, 42, 35] and some commercial offerings already deploy such networks [2, 3]. The advent of three-dimensional (3D) stacked technologies provides a new horizon for on-chip interconnect design.

3D chip technology promises to reduce interconnect delays by stacking multiple layers on top of each other, and by providing shorter vertical connections [44]. 3D technology has matured and demystified some of the concerns on thermal viability and reliability of inter-wafer vias. In addition, it promises to enable integration of heterogeneous technologies on the same chip − such as having layers of memory stacked on top of processor cores − and is even attractive for placing analog and digital components on the same chip, as this avoids common substrate noise problems. Interconnect

[*]This research was supported in part by NSF grants, EIA-0202007, CCF-0429631, CNS-0509251, CAREER 0093085, and a grant from DARPA/MARCO GSRC.

ISCA'07, June 9–13, 2007, San Diego, California, USA.
Copyright 2007 ACM 978-1-59593-706-3/07/0006 ...$5.00.

architecture design across the layers in a 3D architecture requires careful attention for the components on different layers to communicate effectively. Furthermore, there is a need for an integrated approach to interconnect design in the 2D planes and the vertical direction. Currently, there exists no systematic effort at exploring the interconnect architecture for 3D chips. Recently, researchers have started examining some tradeoffs, such as the influence of bandwidth variation of inter-layer interconnects between processor and memory subsystems [28], and combining vertical interconnects with an NoC fabric for chip multiprocessor caches [32].

In this work, we investigate various architectural options for 3D NoC design. Interconnect design in 3D chips imposes new constraints and opportunities compared to that of 2D NoC design. There is an inherent asymmetry in the delays in a 3D architecture between the fast vertical interconnects and the horizontal interconnects that connect neighboring cores, due to differences in wire lengths (few tens of μm in the vertical direction as compared to few thousand μm in the horizontal direction). Consequently, extending a traditional NoC fabric to the third dimension by simply adding routers at each layer (called the Symmetric NoC in this work, due to the symmetry of routing in all directions) is not a good option, as router latencies may dominate the fast vertical interconnect. Hence, we explore two alternate options; a 3D NoC-bus hybrid structure and a true 3D router fabric for the vertical, inter-strata interconnect. A key challenge with 3D NoC routers is limiting the arbitration complexity due to the large path diversity resulting from the additional interconnects in the third dimension.

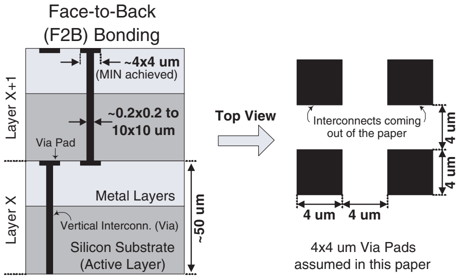


Figure 1: Face-to-Back (F2B) Bonding and the Assumed Vertical Via Layout in this Paper

Vertical interconnects also impose a larger area overhead than corresponding horizontal wires due to the requirement for bonding pads, and can compete with device area as the inter-strata vias punch through the wafer when Face-to-Back (F2B) bonding (see Figure 1) is used. Therefore, the desired number of vertical interconnects used in the 3D router architecture needs to be investigated.

In exploring these tradeoffs in a 3D router design, we developed a new 3D NoC router architecture that we call the 3D Dimensionally-Decomposed (DimDe) Router. The name is a direct corollary of the fact that communication flow through the DimDe router is classified according to the three axes in Euclidean space: X (corresponding to East-West intra-layer traffic), Y (corresponding to North-South intra-layer traffic), and Z (corresponding to inter-layer traffic in the vertical dimension). The idea of decomposing traffic in two dimensions in a 2D environment was introduced in [38, 27] and revisited more recently in [30]. While our proposed DimDe router was inspired by the work in [30], our contribution goes well beyond the introduction of a new traffic dimension. *The DimDe router fuses the crossbars of all the routers in the same vertical "column" (i.e. same X, Y coordinate but different Z coordinate) into a unified entity which allows coordinated concurrent communication across different layers through the same crossbar*. This design amounts to a *true physical* 3D crossbar (unlike the mere stacking of 2D routers in multiple wafer layers). It is important to note that 3D topologies have long been in existence in the macro-network field (e.g. k-ary n-cube), but these rely on 2D routers connected in such a way as to form a *logical* 3D topology. However, 3D chip integration is now enabling the creation of a true physical 3D topology, where the router is itself a three-dimensional entity.

DimDe exhibits the following characteristics that make it a desirable interconnect structure for 3D designs:

(1) *DimDe supports a true 3D crossbar structure which spans all the active layers of the chip*. Irrespective of the number of layers used in the implementation, the 3D crossbar allows a single-hop connection between any two layers, treating all strata as part of a single router structure.

(2) The DimDe design-space provides options for varying the number of vertical connections from one to four to emulate anything between a segmented bus and a full crossbar. Through design space exploration, DimDe was selected to support two vertical interconnects to strike a balance between the path diversity and high bandwidth offered by a full 3D crossbar and the simplicity of a bus. Most importantly, *DimDe's partially-connected crossbar achieves performance levels similar to those of a full 3D crossbar, with substantially reduced area and power overhead and orders of magnitude lower control logic complexity*.

(3) *DimDe supports segmented vertical (i.e. inter-strata) links in the partially-connected crossbar to enable concurrent communication between the different layers of the 3D chip*. This simultaneous data transfer in the vertical dimension significantly increases the vertical bandwidth of the chip as compared to a 3D NoC-bus hybrid structure.

(4) *The DimDe design employs a hierarchical arbitration scheme for inter-strata transfers that reduces area and delay complexity*, while still efficiently enabling simultaneous data transfers. The first stage arbitrates between all requests for vertical communication from within a single layer and the second stage accommodates as many simultaneous requests from the winners of the first stage arbitration.

(5) Similar to the Row-Column (RoCo) Decoupled Router of [30], DimDe completely separates East-West and North-South intra-layer traffic through a pre-sorting operation at the input. However, inter-layer traffic cannot be completely isolated in its own module. A true 3D crossbar requires inter-layer traffic to merge with intra-layer traffic in a seamless fashion; this would allow incoming packets from different layers to continue traversal in the destination layer. DimDe facilitates this tight integration by augmenting the Row (East-West) and Column (North-South) modules with a Vertical Module which fuses with the other two. *The Vertical Module then extends to all other layers and unifies them in a single operational entity*. The Vertical Module assumes the double role of "gluing" all the layers together and blending inter- and intra-layer traffic through unidirectional connections to the Row and Column modules of all layers. It will be demonstrated that this approach dramatically reduces the 3D crossbar complexity, while still allowing concurrent communication between different layers through the switch.

We compare our proposed 3D router design to four different interconnect architectures: a 2D NoC, a 3D Symmetric NoC, a 3D NoC-Bus Hybrid, and a Full 3D Crossbar[1] implementation. To provide as comprehensive an evaluation as possible, we employed a two-pronged simulation environment: (a) a stand-alone, cycle-accurate NoC simulator running synthetic workloads, and (b) a hybrid NoC/cache simulator running a variety of commercial and scientific workloads within the context of a shared, multi-bank NUCA

[1] Our interpretation of a "full" 3D crossbar is presented in Section 3.4 and subsequently formalized in Section 5.3.

L2 cache in an 8-CPU Chip Multi-Processor (CMP) scenario. This double-faceted evaluation process ensures exposure to several traffic patterns, including request/reply memory traffic.

The proposed DimDe design consistently provides the lowest latency for different traffic patterns and it saturates at higher workloads compared to other considered architectures. Our synthetic workload results show that, for high traffic loads, the recently proposed 3D NoC-Bus Hybrid Architecture [32] exhibits the worst latency and throughput for all traffic patterns (even worse than the 2D topology), as the bus saturates first with higher workload. In terms of throughput behavior, the DimDe architecture provides 18% average improvement over the other designs, while remaining within around 3% of the Full 3D Crossbar's throughput. The real workload results indicate that DimDe provides an average improvement of 27% over the 3D Symmetric and 3D NoC-Bus Hybrid designs, and remains within 4% of the Full 3D Crossbar's performance. However, with the Energy-Delay Product (EDP) as the metric, *DimDe significantly outperforms all other designs, including the Full 3D Crossbar, by 26% on average*. Hence, when accounting for both performance and power consumption, the DimDe design is superior to all other 3D router architectures analyzed in this paper. To the best of our knowledge, this is the first systematic exploration and analysis of 3D interconnect architectures and their ramifications on overall system performance.

The rest of this paper is organized as follows. The next section discusses related work. Section 3 provides details of the different 3D interconnect architectures. Section 4 delves into the proposed DimDe architecture. Section 5 presents experimental results, and the conclusions are drawn in section 6.

2. RELATED WORK

The work related to this paper is summarized in three sub-sections: Networks-on-Chip, 3D Technology, and 3D Architectures.

2.1 Networks-on-Chip

The design of efficient on-chip router architectures has been the main focus of many researchers in the past few years. Specifically, micro-architectural optimizations aimed at reducing the pipeline depth have been developed in [39, 30, 36]. The use of speculative switch allocation led to a 3-stage router design in [39]. By using look-ahead routing [22], whereby the routing decision for the current node is performed in the previous node, the pipeline can be reduced to two stages [30]. Recently, a single-stage router has been proposed which utilizes extensive pre-computation techniques [36]. In our design a 2-stage pipeline serves as the base architecture, without loss of generality.

In addition to these pipeline-reducing techniques, several researchers have proposed optimizations of the functional modules. For example, [29] proposed a hierarchical crossbar switch that logically and hierarchically separates the control logic to increase performance and reduce area consumption. Also, [30] proposed a decomposed crossbar to reduce contention and thereby achieve energy-efficient architectures. The RoCo work in [30] introduced the idea of decoupling the NoC router operation into two functionally independent modules, each with its own compact 2×2 crossbar. Incoming packets are sorted into path sets in the two separate modules. Our approach in this work builds upon this philosophy of dimensional decomposition, as described in section 4. The work in [21] uses the NoC router to aid cache coherence in CMPs. By keeping track of cache accesses within each router, the on-chip network now becomes an integral part of the cache coherence protocol.

2.2 3D Integration Technology

Three-dimensional integration technology [19] is an attractive option for overcoming the barriers in interconnect scaling, offering an opportunity to continue the CMOS performance trend. In a three-dimensional (3D) chip, multiple device layers are stacked together. Various 3D integration vertical interconnect technologies have been explored, including wire bonded, microbump, contactless (capacitive or inductive), and through-via vertical interconnect [19]. Through-via interconnection has the potential to offer the greatest vertical interconnect density and therefore is the most promising one among these vertical interconnect technologies. There are two different approaches to implementing through-via 3D integration: the first one involves sequential device process, in which the front-end processing (to build the device layer) is repeated on a single wafer to build multiple active device layers, before the interconnects among devices are built. The second approach processes each active device layer separately, using conventional fabrication techniques, and then stacking these multiple device layers together using wafer-bonding technology. The latter approach requires minimal changes to the manufacturing steps and is more promising; therefore, it is adopted in our proposed architecture. Wafers can be bonded Face-to-Face (F2F) or Face-to-Back (F2B). The through wafer via in F2F wafer-bonding does not go through the thick buried silicon layer and can be fabricated with smaller via sizes. However, for 3D Integrated Circuits (IC) with more than two active layers, F2B stacking provides better scalability, and, therefore, is adopted in our architecture.

Thermal considerations have been a significant concern for 3D integration [13]. However, various techniques have been developed to address thermal issues in 3D architectures such as physical design optimization through intelligent placement [23], increasing thermal conductivity of the stack through insertion of thermal vias [13], and use of novel cooling structures [18]. Further, a recent work demonstrated that the areal power density is the more important design constraint in placement of the processing cores in a 3D chip, as compared to their location in the 3D stack [26]. Consequently, thermal concern can be managed as long as components with high power density are not stacked on top of each other. Architectures that stack memory on top of processor cores, or those that rely on low-power processor cores have been demonstrated to not pose severe thermal problems [11]. In spite of all these advances, one can anticipate some increase in temperature as compared to a 2D design, and also a temperature gradient across layers. Increased temperatures increase wire resistances, and consequently the interconnect delays. To capture this effect, we study the impact of temperature variations on the 3D interconnect delay to assess the effect on performance.

2.3 3D Architectures

Modern System-on-Chip (SoC) designs, such as CMPs, can benefit from 3D integration as well. For example, by placing processing memory, such as DRAM and/or L2 caches, on top of the processing core in different layers, the bandwidth between them can be significantly increased and the critical path can be shortened [33]. In this context, [32] proposed a 3D Network-in-Memory architecture and explored the challenges of managing 3D CMPs together with L2 cache design-space issues. They also proposed the use of an NoC-Bus Hybrid structure for the 3D interconnect. In this paper, we use this structure as one of the comparison points and demonstrate that our proposed architecture is superior. In [28], a CMP design with stacked memory layers is proposed. The authors show that the L2 cache can be removed due to the availability of wide low-latency inter-layer buses between the processing core layer and DRAM layers, and the area saved from this can be recycled for additional cores. Also, [40] has proposed a multi-bank uniform on-chip cache structure using 3D integration. The notion of adding specialized system analysis hardware on separate active layers stacked vertically on the processor die using 3D IC technology is explored in [37]. The modular snap-on introspective layer collects system statistics and acts like a hardware system monitor.

3. THREE-DIMENSIONAL NETWORK-ON-CHIP ARCHITECTURES

This section delves into the exploration of possible architectural frameworks for a three-dimensional NoC network. A typical 2D NoC consists of a number of Processing Elements (PE) arranged in a grid-like mesh structure, much like a Manhattan grid. The PEs are interconnected through an underlying packet-based network fabric. Each PE interfaces to a network router through a Network Interface Controller (NIC). Each router is, in turn, connected to four adjacent routers, one in each cardinal direction.

Expanding this two-dimensional paradigm into the third dimension poses interesting design challenges. Given that on-chip networks are severely constrained in terms of area and power resources, while at the same time they are expected to provide ultra-low latency, the key issue is to identify a reasonable tradeoff between these contradictory design threads. Our task in this section is precisely this: to explore the extension of a baseline 2D NoC implementation into the third dimension, while considering the aforementioned constraints.

3.1 The Baseline 2D NoC Architecture

A generic NoC router architecture is illustrated in Figure 2. The router has P input and P output channels/ports. As previously mentioned, P=5 in a typical 2D NoC router, giving rise to a 5×5 crossbar. The Routing Computation unit, RC, operates on the header flit (a flit is the smallest unit of flow control; one packet is composed of a number of flits) of an incoming packet and, based on the packet's destination, dictates the appropriate output Physical Channel/port (PC) and/or valid Virtual Channels (VC) within the selected output PC. The routing can be deterministic or adaptive. The Virtual channel Allocation unit (VA) arbitrates between all packets competing for access to the same output VCs and chooses a winner. The Switch Allocation unit (SA) arbitrates between all VCs requesting access to the crossbar. The winning flits can then traverse the crossbar and move on to their respective output links. Without loss of generality, all implementations in this work employ two-stage routers.

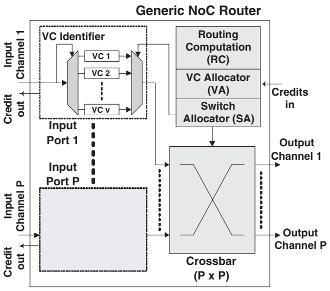
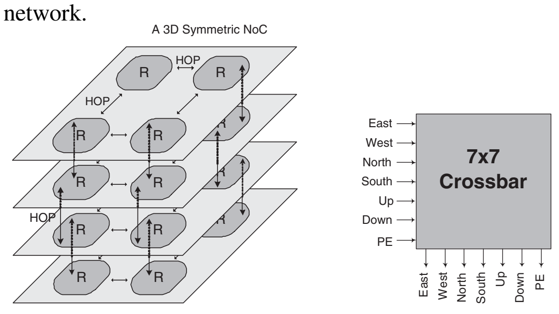


Figure 2: A Generic NoC Router Architecture

3.2 A 3D Symmetric NoC Architecture

The natural and simplest extension to the baseline NoC router to facilitate a 3D layout is simply adding two additional physical ports to each router; one for Up and one for Down, along with the associated buffers, arbiters (VC arbiters and Switch Arbiters), and crossbar extension. We call this architecture a 3D Symmetric NoC, since both intra- and inter-layer movement bear identical characteristics: hop-by-hop traversal, as illustrated in Figure 3(a). For example, moving from the bottom layer of a 4-layer chip to the top layer requires 3 network hops.

This architecture, while simple to implement, has two major inherent drawbacks: (1) It wastes the beneficial attribute of a negligible inter-wafer distance (around 50 μm per layer) in 3D chips, as shown in Figure 1. Since traveling in the vertical dimension is multi-hop, it takes as much time as moving within each layer. Of course, the average number of hops between a source and a destination does decrease as a result of folding a 2D design into multiple stacked layers, but inter-layer and intra-layer hops are indistinguishable. Furthermore, each flit must undergo buffering and arbitration at every hop, adding to the overall delay in moving up/down the layers. (2) The addition of two extra ports necessitates a larger 7×7 crossbar, as shown in Figure 3(b). Crossbars scale upward very inefficiently, as illustrated in Table 1. This table includes the area and power budgets of all crossbar types investigated in this paper, based on synthesized implementations in 90 nm technology. Details of the design and synthesis methodology are given in Section 5.2. Clearly, a 7×7 crossbar incurs significant area and power overhead over all other architectures. Therefore, the 3D Symmetric NoC implementation is a somewhat naive extension to the baseline 2D network.



(a) Overall View (b) Crossbar Configuration

Figure 3: A 3D Symmetric NoC Network

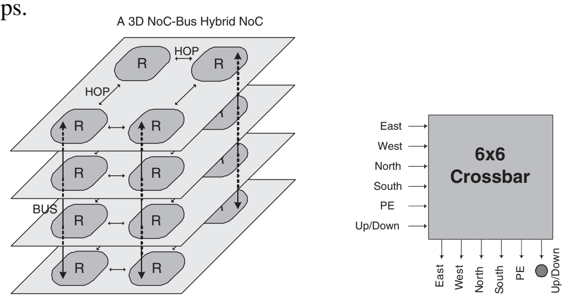
Crossbar Type	Area	Power with 50% switching activity at 500 MHz
4×2 Crossbar(for 3D DimDe)	3039.32 μm^2	1.63 mW
5×5 Crossbar(Conventional 2D Router)	8523.65 μm^2	4.21 mW
6×6 Crossbar(3D NoC-Bus Hybrid)	11579.10 μm^2	5.06 mW
7×7 Crossbar(3D Symmetric NoC Router)	17289.22 μm^2	9.41 mW

Table 1: Area and Power Comparisons of the Crossbar Switches Assessed in this Work

3.3 The 3D NoC-Bus Hybrid Architecture

The previous sub-section argues that multi-hop communication in the vertical (inter-layer) dimension is not desirable. Given the very small inter-strata distance, single-hop communication is, in fact, feasible. This realization opens the door to a very popular shared-medium interconnect, the bus. The NoC router can be hybridized with a bus link in the vertical dimension to create a 3D NoC-Bus Hybrid structure, as shown in Figure 4(a). This approach was first introduced in [32], where it was used in a 3D NUCA L2 Cache for CMPs. This hybrid system provides both performance and area benefits. Instead of an unwieldy 7×7 crossbar, it requires a 6×6 crossbar (Figure 4(b)), since the bus adds a single additional port to the generic 2D 5×5 crossbar. The additional link forms the interface between the NoC domain and the bus (vertical) domain. The bus link has its own dedicated queue, which is controlled by a central arbiter. Flits from different layers wishing to move up/down should arbitrate for access to the shared medium.

Figure 5 illustrates the side view of the vertical via structure. This schematic depicts the usefulness of the large via pads between the different layers; they are deliberately oversized to cope with misalignment issues during the fabrication process. Consequently, it is the large vias which ultimately limit vertical via density in 3D chips.



(a) Overall View (b) Crossbar Configuration

Figure 4: A 3D NoC-Bus Hybrid Architecture

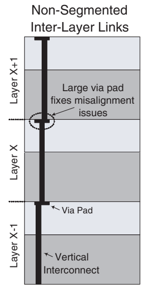

Non-Segmented Inter-Layer Links
Layer X+1
Large via pad fixes misalignment issues
Layer X
Via Pad
Layer X-1
Vertical Interconnect

Figure 5: Side View of the Inter-Layer Via Structure in a 3D NoC-Bus Hybrid Structure

Despite the marked benefits over the 3D Symmetric NoC router of Section 3.2, the bus approach also suffers from a major drawback: it does not allow concurrent communication in the third dimension. Since the bus is a shared medium, it can only be used by a single flit at any given time. This severely increases contention and blocking probability under high network load, as will be demonstrated in Section 5. Therefore, while single-hop vertical communication does improve performance in terms of overall latency, inter-layer bandwidth suffers.

3.4 A True 3D NoC Router

Moving beyond the previous options, we can envision a true 3D crossbar implementation, which enables seamless integration of the vertical links in the overall router operation. Figure 6 illustrates such a 3D crossbar layout. It should be noted at this point that the traditional definition of a *crossbar* - in the context of a 2D physical layout - is a switch in which each input is connected to each output through a single connection point. However, extending this definition to a physical 3D structure would imply a switch of enormous complexity and size (given the increased numbers of input- and output-port pairs associated with the various layers). Therefore, in this paper, we chose a simpler structure which can accommodate the interconnection of an input to an output port through more than one connection points. While such a configuration can be viewed as a multi-stage switching network, we still call this structure a *crossbar* for the sake of simplicity.

The vertical links are now embedded in the crossbar and extend to all layers. This implies the use of a 5×5 crossbar, since no additional physical channels need to be dedicated for inter-layer communication. As shown in Table 1, a 5×5 crossbar is significantly smaller and less power-hungry than the 6×6 crossbar of the 3D NoC-Bus Hybrid and the 7×7 crossbar of the 3D Symmetric NoC. Interconnection between the various links in a 3D crossbar would have to be provided by dedicated connection boxes at each layer. These connecting points can facilitate linkage between vertical and horizontal channels, allowing flexible flit traversal within the 3D crossbar. The internal configuration of such a Connection Box (CB) is shown in Figure 7(a). The horizontal pass transistor is dotted, because it is not needed in our proposed 3D crossbar implementation, which is presented in Section 4. The vertical link segmentation also affects the via layout, as illustrated in Figure 7(b). While this layout is more complex than that shown in Figure 5, the area between the offset vertical vias can still be utilized by other circuitry, as shown by the dotted ellipse in Figure 7(b).

Hence, the 2D crossbars of all layers are physically fused into one single three-dimensional crossbar. Multiple internal paths are present, and a traveling flit goes through a number of switching points and links between the input and output ports. Moreover, flits re-entering another layer do not go through an intermediate buffer; instead, they directly connect to the output port of the destination layer. For example, a flit can move from the western input port of layer 2 to the northern output port of layer 4 in a single hop.

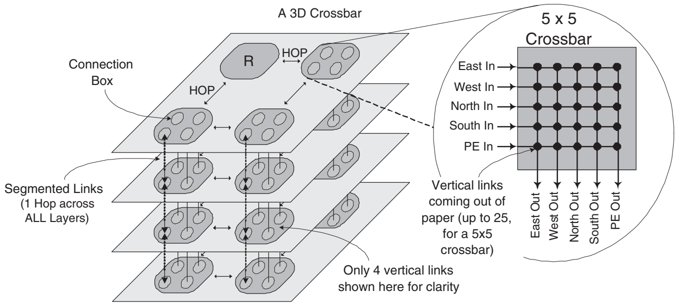
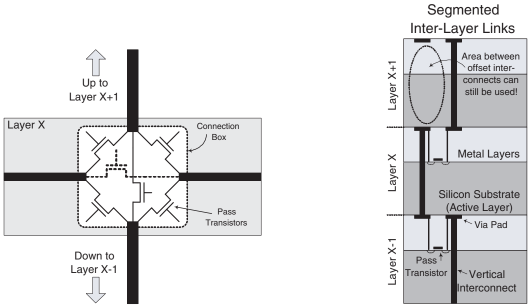

A 3D Crossbar
Connection Box
HOP
HOP
R
5 x 5 Crossbar
East In
West In
North In
South In
PE In
East Out
West Out
North Out
South Out
PE
Segmented Links (1 Hop across ALL Layers)
Vertical links coming out of paper (up to 25, for a 5x5 crossbar)
Only 4 vertical links shown here for clarity

Figure 6: NoC Routers with True 3D Crossbars



(a) Internal Details of a Connection Box (CB) (b) Inter-layer Via Layout

Figure 7: Side View of the Inter-Layer Via Structure in a 3D Crossbar

It will be shown in Section 4 that adding a 128-bit vertical link, along with its associated control signals, consumes only about 0.01 mm^2 of silicon real estate.

However, despite this encouraging result, there is an opposite side to the coin which paints a rather bleak picture. Adding a large number of vertical links in a 3D crossbar to increase NoC connectivity results in increased path diversity. This translates into multiple possible paths between source and destination pairs. While this increased diversity may initially look like a positive attribute, it actually leads to a dramatic increase in the complexity of the central arbiter, which coordinates inter-layer communication in the 3D crossbar. The arbiter now needs to decide between a multitude of possible interconnections, and requires an excessive number of control signals to enable all these interconnections. Even if the arbiter functionality can be distributed to multiple smaller arbiters, then the coordination between

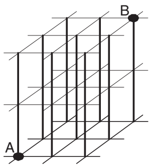
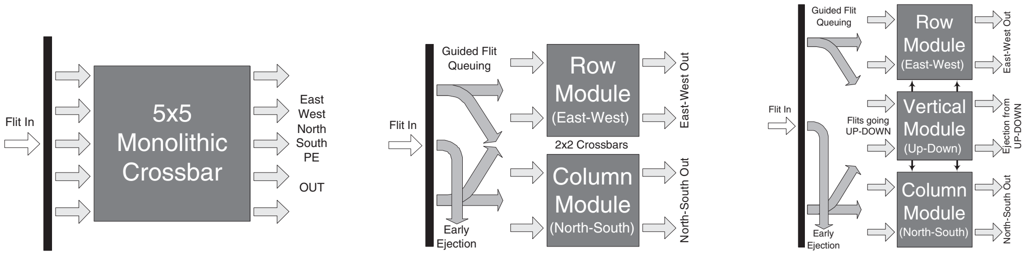


Figure 8: A 3D 3×3×3 Crossbar in Conceptual Form



(a) Conventional 2D NoC Router Overview

(b) The 2D Row-Column (RoCo) Decoupled Router

(c) The Proposed 3D DimDe Router Architecture

Figure 9: Different NoC Router Switching Mechanisms

these arbiters becomes complex and time-consuming. Alternatively, if dynamism is sacrificed in favor of static path assignments, the exploration space is still daunting in deciding how to efficiently assign those paths to each source-destination pair. Furthermore, a full 3D crossbar implies 25 (i.e. 5x5) Connection Boxes (see Figure 7(a)) per layer. A four-layer design would, therefore, require 100 CBs! Given that each CB consists of 6 transistors, the whole crossbar structure would need 600 control signals for the pass transistors alone! Such control and wiring complexity would most certainly dominate the whole operation of the NoC router. Pre-programming static control sequences for all possible input-output combinations would result in an oversize table/index; searching through such table would incur significant delays, as well as area and power overhead. The vast number of possible connections hinders the otherwise streamlined functionality of the switch. Note that the prevailing tendency in NoC router design is to minimize operational complexity in order to facilitate very short pipeline lengths and very high frequency. A full crossbar with its overwhelming control and coordination complexity poses a stark contrast to this frugal and highly efficient design methodology. Moreover, *our experimental results will show that the redundancy offered by the full connectivity is rarely utilized by real-world workloads, and is, in fact, design overkill.*

To understand the magnitude of the path diversity issue in a true 3D crossbar (as shown in Figure 8 for a 3×3×3 example), one can picture the 3D crossbar itself as a 3D Mesh network. For the 3D 3×3×3 crossbar of Figure 8, the number of minimal paths, k, between points A and B is given in [17] as

$$k = \binom{\Delta_x + \Delta_y + \Delta_z}{\Delta_x} \binom{\Delta_y + \Delta_z}{\Delta_y} = \frac{(\Delta_x + \Delta_y + \Delta_z)!}{\Delta_x! \Delta_y! \Delta_z!} \tag{1}$$

where Δ_x, Δ_y and Δ_z are the numbers of hops separating A and B in the X, Y, and Z dimensions, respectively. In our example, $\Delta_x = \Delta_y = \Delta_z = 2$. Thus, the number of minimal paths between A and B is 90. For a 3D 4×4×4 crossbar, this number explodes to 1680. If non-minimal paths are also considered, then path diversity is practically unbounded [17].

Hence, given the tight latency and area constraints in NoC routers, vertical (inter-layer) arbitration should be kept as simple as possible. This can be achieved by using a limited amount of inter-layer links. The question is then: *how many links are enough? Our experiments in Section 5 demonstrate that anything beyond two links per 3D crossbar yields diminishing returns in terms of performance*.

3.5 A Partially-Connected 3D NoC Router Architecture

The scalability problem in vertical link arbitration highlighted in the previous sub-section dictates the use of a partially-connected 3D crossbar, i.e. a crossbar with a limited number of vertical links. The arbitration complexity can be further mitigated through the use of hierarchical arbiters. Two types of arbiters should be employed: intra-layer arbiters, which handle local requests from a single layer, and one global arbiter per vertical link to handle requests from all layers. This decoupling of arbitration policies can help parallelize tasks; while flits arbitrate locally in each layer, vertical arbitration decides on inter-layer traversal. These design directives were the fundamental drivers in our quest for a suitable 3D NoC implementation. As such, they form the cornerstones of our proposed architecture, which is described in detail in the following section.

4. THE PROPOSED 3D DIMENSIONALLY-DECOMPOSED (DIMDE) NOC ROUTER ARCHITECTURE

The heart of a typical two-dimensional NoC router is a monolithic, 5×5 crossbar, as depicted abstractly in Figure 9(a). The five inputs/outputs correspond to the four cardinal directions and the connection from the local PE. The realization that the crossbar is a major contributor to the latency and area budgets of a router has fueled extensive research in optimized switch designs. Through the use of a preliminary switching process, known as Guided Flit Queuing [30], incoming traffic may be decomposed into two independent streams: (a) East-West traffic (i.e. packet movement in the X dimension), and (b) North-South traffic (i.e. packet movement in the Y dimension). This segregation of traffic flow allows the use of two smaller 2×2 crossbars and the isolation of the two flows in two independent router sub-modules, as shown conceptually in Figure 9(b). The resulting two compact modules are more area- and power-efficient, and provide better performance than the conventional monolithic approach.

Following this logic of traffic decomposition in orthogonal dimensions, we propose in this work the addition of a third information flow in the Z dimension (i.e. inter-layer communication). An additional module is now required to handle all traffic in the third dimension; this component is aptly called the Vertical Module. On the input side, packets are decomposed into the three dimensions (X, Y, and Z), and forwarded to the appropriate module. However, as previously mentioned, simply adding a third independent module cannot lead to a true 3D crossbar, because inter-layer traffic must be able to merge with intra-layer traffic upon arrival at the destination chip layer. A totally decoupled Vertical Module would force all packets arriving at a particular layer and wishing to continue traversal within that layer to be re-buffered and re-arbitrate for access to the Row/Column modules.

Hence, the Vertical Module must somehow fuse the Row and Column modules to allow movement of packets from the Vertical Module to the Row and Column Modules. An abstract view of the proposed 3D DimDe implementation is illustrated in Figure 9(c). The diagram clearly shows the Vertical Module linking with the Row and Column Modules. Also notice that the communication link is one-way, i.e. from the Vertical Module to the Row/Column Modules. There is no need for the Row/Column Modules to communicate with the Vertical Module, since intra-layer traffic wishing to change layer is pre-directed to the Vertical Module at the input of the router.

The streamlined nature of a dimensionally decomposed router lends itself perfectly for a 3D crossbar implementation. The simplicity and compactness of the smaller, distinct modules can be utilized to create a crossbar structure which extends into the third dimension without incurring prohibitive area and latency overhead. The high-level architectural overview of our proposed 3D DimDe router is shown in Figure 10. As illustrated in the figure, the gateway to different layers is facilitated by the inclusion of the third, Vertical Module. The 3D DimDe router uses vertical links which are segmented at the different device layers through the use of compact Connection Boxes (CB). Figure 7(a) shows a side view cross-section of such a CB. Each box consists of 5 pass transistors which can connect the vertical (inter-layer) links to the horizontal (intra-layer) links. The dotted transistor is not needed in DimDe, because the design was architected in such a way as to avoid the case where intra-layer communication needs to pass through a CB. The CB structure allows simultaneous transmission in two directions, e.g. a flit coming from layer X+1 and connecting to the left link of Layer X, and a flit coming from layer X-1 connecting to the right link of layer X (see Figure 7(a)). The inclusion of pass transistors in the data path adds delay and degrades the signal strength due to the associated voltage drop. However, this design decision is fully justified by the fact that inter-layer distances are, in fact, negligible. To investigate the effectiveness and integrity of this connection scheme, we laid out the physical design of the CB and simulated it in HSpice using the Predictive Technology Model (PTM) [4] at 70 nm technology and 1 V power supply. The latency results for 2, 3 and 4-layer distances are shown in Table 2. Evidently, even with a four-layer design (i.e. traversing four cascaded pass transistors), the delay is only 36.12 ps; this is a mere 1.8% of the 2 ns clock period (500 MHz) of the NoC router. In fact, the addition of repeaters will increase latency, because with such small wire lengths (around 50 μm per layer), the overall propagation delay is dominated by the gate delays and not the wiring delay. This effect is corroborated by the increased delay of 105.14 ps when using a single repeater, in Table 2.

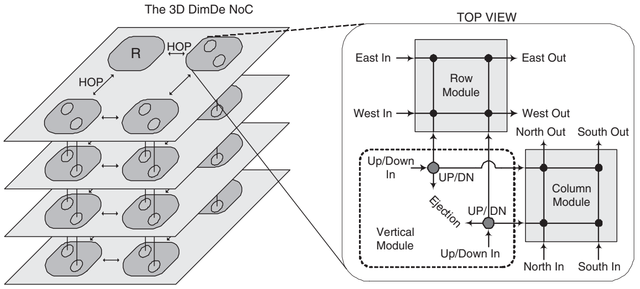


Figure 10: Overview of the 3D DimDe NoC Architecture

Inter-Layer Link Length	Number of Repeaters	Delay
50 μm (Layer 1 to 2)	0	7.86 ps
100 μm (Layer 1 to 3)	0	19.05 ps
150 μm (Layer 1 to 4)	0	36.12 ps
150 μm (Layer 1 to 4)	1 (layer 3)	105.14 ps

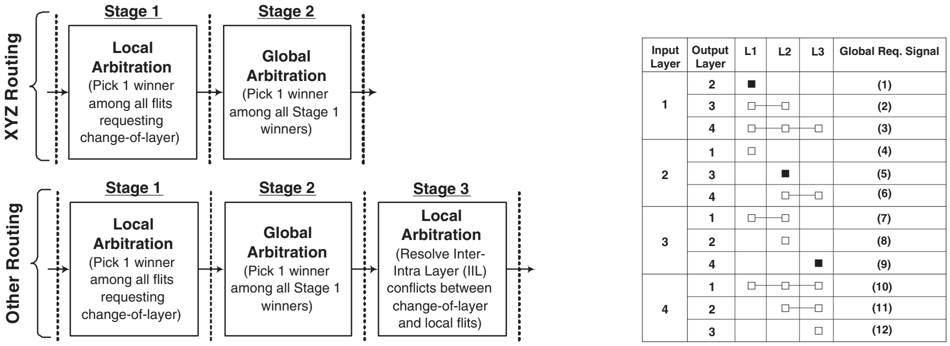
Table 2: The Inter-Layer Distance on Propagation Delay

To indicate the fact that each vertical link in the proposed architecture is composed of a number of wires, we thereby refer to these links as bundles. The presence of a segmented wire bundle dictates the use of one central arbiter for each vertical bundle, which is assigned the task of controlling all traffic along the vertical link. If arbitration is carried out at a local level alone, then the benefit of concurrent communication along a single vertical bundle cannot be realized; each layer would simply be unaware of the connection requests of the other layers. Hence, a coordinating entity is required to monitor all requests for vertical transfer from all the layers and make an informed decision, which will favor simultaneous data transfer whenever possible. Concurrent communication increases the vertical bandwidth of the 3D chip. Given the resource-constrained nature of NoCs, however, the size and operational complexity of the central arbiter should be handled judiciously. The goal is not to create an overly elaborate mechanism which provides the best possible matches over several clock cycles. Our objective was to obtain reasonably intelligent matches within a single clock cycle.

To achieve this objective, we divided the arbitration for the vertical link into two stages, as shown at the top of Figure 11(a). The first stage is performed locally, within each layer. This stage arbitrates over all flits in a single layer which request a transfer to a different layer. Once a local winner is chosen, the local arbiter notifies the second stage of arbitration, which is performed globally. This global stage takes in all winning requests from each layer and decides on how the segmented link will be configured to accommodate the inter-layer transfer(s). The arbiter was designed in such a way as to realize the scenarios which are suitable for concurrent communication.

Figure 11(b) illustrates all possible requests to the global arbiter of a particular vertical bundle, assuming a 4-layer chip configuration using the deterministic XYZ routing. The designations L1, L2, and L3 indicate the different segments of the vertical bundle; L1 is the link between layers 1 and 2, L2 is the link between layers 2 and 3, and so on. As an example, let us assume that a flit in layer 1, which wants to go to layer 2, has won the local arbitration of layer 1; global request signal 1 (see Figure 11(b)) is asserted. Similarly, a flit in layer 2 wants to go to layer 3; global request signal 5 is asserted. Finally a flit in layer 3 wants to go to layer 4; global request signal 9 is asserted. The global arbiter is designed to recognize that the global request combination 1, 5, 9 (black boxes in Figure 11(b)) results in full concurrent communication between all participating layers. It will, therefore, grant all requests simultaneously. All combinations which favor simultaneous, non-overlapping communication are programmed into the global arbiter. If needed, these configurations can be given higher priority in the selection process. The arbiter can be placed on any layer, since the vertical distance to be traveled by the inter-layer control signals is negligible. The aforementioned two arbitration stages suffice only if deterministic XYZ routing is used. In this case, a flit traveling in the vertical (i.e. Z) dimension will be ejected to the local PE upon arrival at the destination layer's router. If, however, a different routing algorithm is used, which allows flits coming from different layers to continue their traversal in the destination layer, then an additional local arbitration stage is required to handle conflicts between flits arriving from different layers and flits residing in the destination layer. The third arbitration stage, illustrated at the bottom of Figure 11(a), will take care of such Inter-Intra Layer (IIL) conflicts. The use of non-XYZ algorithms also complicates the request signals sent across different layers. It is no longer enough to merely indicate the destination layer; the output port designation on the destination layer also needs to be sent. IIL conflicts highlight the complexity involved in coordinating flit traversal in a 3D network environment. An example of the use of a non-XYZ routing algorithm is presented in Figure 12, which tracks the path of a flit traveling from Layer X to the eastern output of Layer X+1. In this case, the flit changes layer and continues traversal in a different layer.

Each vertical bundle in DimDe consists of a number of data wires (128 bits in this work), and a number of control wires to/from a central arbiter which coordinates flit movement in the vertical di-


Stage 1
Stage 2
XYZ Routing
Local Arbitration
(Pick 1 winner among all flits requesting change-of-layer)
Global Arbitration
(Pick 1 winner among all Stage 1 winners)
Stage 1
Stage 2
Stage 3
Other Routing
Local Arbitration
(Pick 1 winner among all flits requesting change-of-layer)
Global Arbitration
(Pick 1 winner among all Stage 1 winners)
Local Arbitration
(Resolve Inter-Intra Layer (IIL) conflicts between change-of-layer and local flits)

(a) Vertical (Inter-Layer) Link Arbitration Stages

Input Layer	Output Layer	L1	L2	L3	Global Req. Signal
1	2	■			(1)
	3	□	□		(2)
	4	□	□	□	(3)
2	1	□			(4)
	3		■		(5)
	4		□	□	(6)
3	1	□	□		(7)
	2		□		(8)
	4			■	(9)
4	1	□	□	□	(10)
	2		□	□	(11)
	3			□	(12)

(b) 3D Global Arbitration Request Scenarios

Figure 11: Vertical (Inter-Layer) Link Arbitration Details

mension. These control signals include: (a) Request signals from all layers to the central arbiter indicating the requested destination layer (and possibly output port, depending on the routing algorithm used), and the corresponding acknowledgement signals from the arbiter. (b) Enable signals from the arbiter to the pass transistors of the Connection Boxes of each layer spanned by the wire bundle. The total number of wires, w, in a vertical bundle is given by

$$w = \begin{cases} b + 2(n-1)^2 + 5(n-1), & if\ XYZ\ algorithm \\ b + 2(n-1)^2 + 6(n-1) + 5(n-1), & otherwise \end{cases}$$

where

$$\begin{aligned} b &= number\ of\ data\ bits/wires, \\ 2(n-1)^2 &= number\ of\ request/acknowledgement \\ &\quad signals\ to/from\ the\ central\ arbiter \\ &\quad assuming\ an\ n-layer\ chip, \\ 6(n-1) &= number\ of\ additional\ signals\ sent \\ &\quad to/from\ the\ arbiter\ for\ output\ port \\ &\quad designation\ (3-bit\ designation\ for \\ &\quad the\ four\ possible\ output\ ports\ and \\ &\quad the\ ejection\ port)\ when\ a\ non-XYZ \\ &\quad routing\ algorithm\ is\ employed, \\ 5(n-1) &= number\ of\ enable\ signals\ for\ the\ pass \\ &\quad transistors\ of\ the\ CB\ of\ each\ layer. \end{aligned}$$

Assuming a 4-layer configuration ($n = 4$), XYZ routing, and 128 data bits (i.e. $b = 128$), the number of wires in a vertical bundle, w, is 161. Based on the square-like layout of Figure 1, the area consumed by the bundle is around 10,000 μm^2 = 0.01 mm^2. This amounts to a vertical via density of around 1.5 million individual wires per cm^2. This result illustrates the fact that increasing the number of vertical vias is, in fact, feasible in terms of area consumption by the wires themselves. However, as explained in Section 3.4, adding extra vertical bundles in the 3D crossbar is prohibitive in terms of arbitration complexity; the area, power and latency increases incurred by a highly-complex arbitration scheme negate any advantages provided by the increased number of inter-layer bundles. Furthermore, it will be demonstrated later on that increasing the number of inter-layer bundles yields rapidly diminishing returns in terms of performance gain under both synthetic and real workloads. A detailed view of the proposed 3D DimDe architecture is shown in Figure 13. DimDe employs Guided Flit Queuing [30] to guide incoming flits to an appropriate Path Set (PS). Guided Flit Queuing is a preliminary switching operation at the input of the router which utilizes the look-ahead routing information present in incoming header flits. This information denotes the requested output path; thus, incoming traffic can be decomposed into the X, Y, and Z dimensions. The Vertical Module adds two extra path sets to the 2D implementation. One path set is used by incoming flits from the East-West (intra-layer) dimension, and the other for flits from the North-South dimension. Just like Guided Flit Queuing, the Early Ejection Mechanism [30] uses the look-ahead routing information to identify packets which need to be ejected to the local PE. This enables such flits to bypass the destination router and be directly ejected to the NIC. The Vertical Module consists of two bidirectional vertical bundles, one for each of the two path sets. Note that the number of vertical bundles can be varied from four to one. Each vertical link has one input connection and three output connections on each layer. The input connection comes from the associated path set's MUX (see dark box in the middle of Figure 13). The three output connections are as follows: (1) One connection to the Row Module Crossbar for flits which arrive from other layers and need to continue traversal in the East-West dimension of the current layer. (2) One connection to the Column Module crossbar for flits which need to continue in the North-South dimension of the current layer. (3) One connection for ejection to the Network Interface Controller (NIC) of the local PE. This configuration implies that the Row and Module crossbars need to grow in size from 2×2 in the 2D case to 4×2 in DimDe to accommodate the two additional connections from the two vertical links. Despite this increase in size, two 4×2 crossbars are still substantially smaller than a single monolithic 6×6 or 7×7 crossbar, as illustrated in Table 1. Once again, it is precisely for this reason that we chose to use this architecture in our 3D NoC implementation.

The Vertical Module of the proposed DimDe router uses two Path Sets to group the available Virtual Channels. As shown in Figure 14, the DimDe router requires 5 VCs for correct functionality under a deterministic, deadlock-free algorithm: one VC for injection from each of the four incoming directions, and one for injection from the local PE. The sixth VC can be used as a drain channel for deadlock recovery under adaptive routing algorithms. Moreover, depending on the algorithm used, additional VCs can be added to the two Vertical Module path sets to ensure deadlock freedom. These drain VCs need to operate on deadlock-free algorithms to guarantee deadlock breakup [20]. In this work, we concentrated on deterministic XYZ and ZXY algorithms as a proof of concept of the proposed architecture. Since these algorithms are inherently deadlock-free, the sixth VC buffer was used as an additional injection VC from the local PE.

As previously explained in Section 2.2, thermal issues are of utmost importance in 3D chips. Stacking several active layers with minimal distance in-between favors the creation of hotspots. From a 3D NoC perspective, it was important to investigate the effect of high temperature on the propagation delay of the signals on the vertical (inter-layer) interconnects. To that extend, the propagation delay between the layers was modeled as an RC ladder (Figure 15(b)) to accurately capture the distributed resistance, capacitance, and temperature variations along the inter-strata vias. The resistance of metals is affected by temperature, and it was modeled using equations from [12]. Assuming a T_{Layer1} temperature of 85 °C and a fixed linear temperature gradient between each layer, the propagation delay of these vias was simulated in HSpice with the required

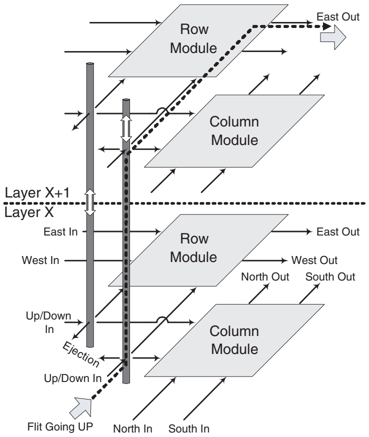


Figure 12: An Example of a Non-XYZ Routing Algorithm

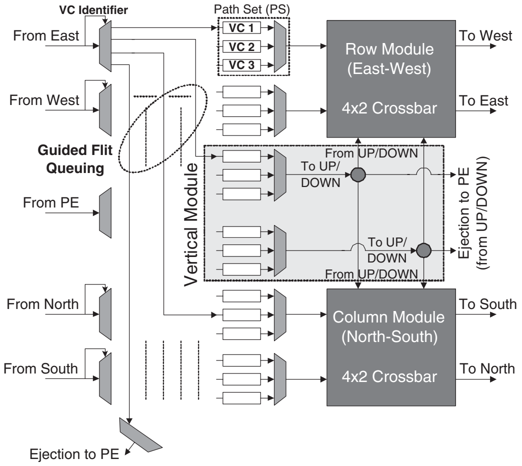


Figure 13: Architectural Detail of the Proposed 3D DimDe NoC Router

temperature annotations. Even in the worst case of a 10 °C temperature increase per layer for 8 layers, the total propagation delay from the lowest to the highest layer was only 0.11 ps and, therefore, considered inconsequential for our work. The results of the thermal analyses are summarized in Figure 15(a).

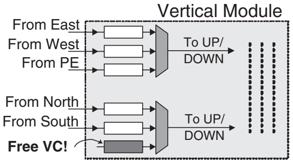
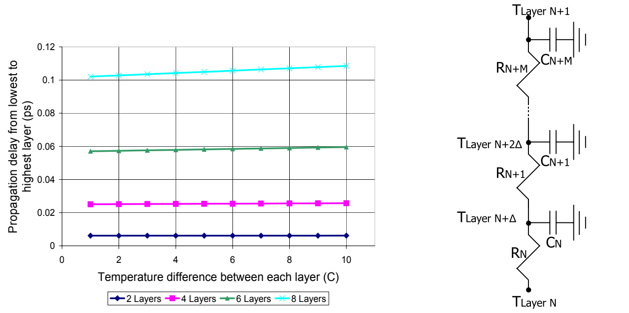


Figure 14: Virtual Channel Assignments in the Vertical Module of DimDe



(a) Inter-Layer Propagation Delay vs. Temperature (b) Modeling of Temperature Effect on Propagation Delay

Figure 15: Thermal Effects on Inter-Layer Propagation Delay

5. PERFORMANCE EVALUATION

In this section, we present simulation-based performance evaluation of our architecture, a generic 2D router architecture, a 3D Symmetric NoC design, the 3D NoC-Bus Hybrid architecture, and the Full 3D Crossbar implementation, in terms of network latency, throughput and power consumption under various traffic patterns. Our experimental methodology is followed by the experimental results.

5.1 Simulation Platform

A double-faceted evaluation environment was implemented in order to conduct a detailed evaluation of the router architectures analyzed in this paper: (a) A cycle-accurate stand-alone 3D NoC simulator was developed, which accurately models the routers, the interconnection links and vertical pillars, as well as all the architectural features of the various NoC architectures under investigation. The simulator was built by augmenting an existing 2D NoC simulator and models each individual component within the router architecture, allowing for detailed analysis of component utilizations and flit flow through the network. The activity factor of each component is used for analyzing power consumption within the network. In addition to the network-specific parameters, our simulator accepts hardware parameters such as power consumption (dynamic and leakage) for each component and overall clock frequency. This leg of the simulation process examines the behavior of all the architectures under synthetic workloads.

(b) To provide a more diversified simulation environment, we also implemented a detailed trace-driven cycle-accurate hybrid NoC/cache simulator for CMP architectures. The memory hierarchy implemented is governed by a two-level directory cache coherence protocol. Each core has a private write-back L1 cache (split L1 I and D cache, 64 KB, 2-way, 3-cycle access). The L2 cache is shared among all cores and split into banks (32 banks, 512 KB each for a total of 16 MB, 6-cycle bank access). An underlying NoC model connects the L2 banks. The L1/L2 block size is 64 B. Our coherence model includes a MESI-based protocol with distributed directories, with each L2 bank maintaining its own local directory. The simulated memory hierarchy mimics SNUCA [9]. The sets are statically placed in the banks depending on the low order bits of the address tags. The network timing model simulates all kinds of messages: invalidates, requests, replies, write-backs, and acknowledgements. The interconnect model is the same as (a) above. The off-chip memory is a 4 GB DRAM with a 260-cycle access time.

Detailed instruction traces of four commercial server workloads were used: (1) TPC-C [8], a database benchmark for online transaction processing (OLTP), (2) SAP [5], a sales and distribution benchmark, and (3) SJBB [7] and (4) SJAS [6], two Java-based server benchmarks. The traces – collected from multiprocessor server configurations at Intel Corporation – were then run through our NoC/cache hybrid simulator to measure network statistics. Additionally, a second set of memory traces was generated by executing programs from SPLASH [43], a suite of parallel scientific benchmarks, on the Simics full system simulator [34]. Specifically, barnes, ocean, water-nsquared (wns), water-spatial (wsp), lu, and radiosity (rad) were used. The baseline configuration is the Solaris 9 Operating system running on eight UltraSPARC III cores. Benchmarks execute 16 parallel threads. Again, the number of banks for the L2 shared cache is 32. Thus, 32 nodes are present in the NoC network, 8 of which are also CPU nodes.

5.2 Energy Model

The proposed components of the 3D router architectures, and a generic two-stage 5-port router architecture were implemented in structural Register-Transfer Level (RTL) Verilog and then synthesized in Synopsys Design Compiler using a TSMC 90 nm standard cell library. The library utilized appropriate wire-load approximation models to reasonably capture wire loading effects. The vertical interconnects were modeled as 2D wires with equivalent resistance and capacitance. The resulting designs operate at a supply voltage of 1 V and a clock speed of 500 MHz. Both dynamic and leakage power estimates were extracted from the synthesized router implementation. These power numbers were then imported into our cycle-accurate simulation environment and used to trace the power profile of the entire on-chip network.

5.3 Performance Results

The proposed 3D DimDe design was compared against four other router architectures (2D NoC, 3D Symmetric NoC, 3D NoC-Bus Hybrid, and a Full 3D Crossbar configuration) using our cycle-accurate simulation environment. Our definition of a "full" 3D crossbar implies that all connection points inside the 2D 5×5 crossbar (i.e. 25 links) extend into the third (i.e. vertical) dimension. It should *not* be confused with a *non-blocking* crossbar where each input can be connected to each output regardless of how the other inputs and outputs are interconnected. Such a configuration would be tremendously complex in a physical 3D setting because the number of possible input-output pairs explodes as the number of layers increases.

In both simulation phases, two deterministic routing algorithms (XYZ routing and ZXY routing) were used to measure the average network latency, throughput, and power consumption in all experiments. For the synthetic workload simulation phase (described in part (a) of Section 5.1), all architectures under investigation were evaluated using a regular mesh network with 64 nodes. In the 3D designs, 4 layers were used, each with 16 nodes (4x4). Wormhole routing [15] based on virtual-channel flow control [14] was employed in all cases. To ensure fairness, all architectures under test had 3 VCs per input port, and a total buffer space of 80 flits per node. Each simulation consists of two phases: a warm-up phase of 20,000 packet injections, followed by the main phase which injects a further one million packets. Each packet consists of four 128-bit flits. The simulation terminates when all packets are received at the destination nodes. Uniform, matrix-transpose (dimension reversal) [41] and self-similar traffic patterns were used.

For the real workload simulation phase (described in part (b) of Section 5.1), the 32 L2 cache banks (nodes) were "folded" into 4 layers, with each layer holding 8 banks (4x2). The 8 CPUs were also split into 2 CPUs/layer. The commercial workloads were simulated for 10,000 transactions per thread, whereas the scientific workloads were simulated for 100 million instructions per core upon commencement of the parallel phase of the code. Data messages were 5-flit packets (64 B cache-line plus network overhead), while control messages were single-flit packets.

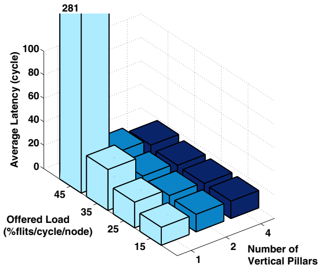


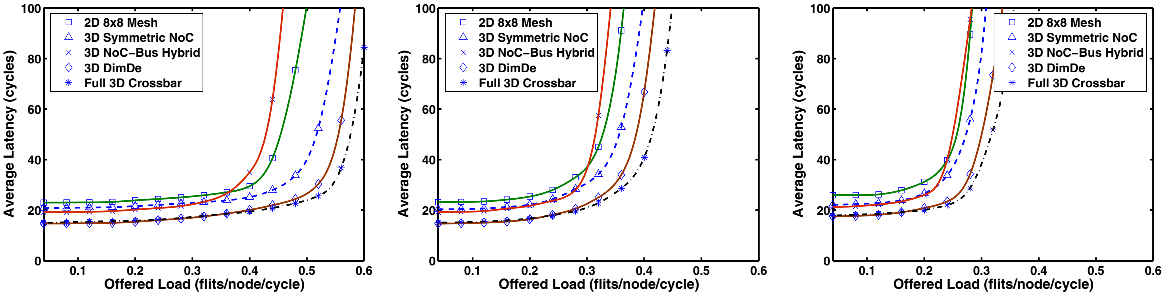
Figure 16: Impact of the Number of Vertical Bundles on Performance

The 3D DimDe architecture design exploration provided different options for the number of pillars in the Vertical Module. Since we are using 2×2 crossbars as the basic building blocks, four pillars would provide a complete crossbar connection, while a single pillar would provide a segmented bus connection. As previously mentioned, the caveat is that more vertical pillars offer more path diversity and complicate the arbiter design. Hence, the number of pillars should be decided based on the performance, energy and area tradeoffs. Figure 16 illustrates the effect of the number of vertical pillars (per node) on average network latency. Interestingly, going from two to four vertical bundles yields rapidly diminishing returns in terms of performance gains. This experiment suggests that the two-pillar DimDe design provides the best compromise in terms of performance, area and energy behavior; employing only two - instead of more - vertical links would lower design complexity and power consumption without adversely affecting network latency. Therefore, in the rest of the evaluations, we use the two-pillar DimDe architecture (as shown in Figure 13). Moreover, to validate our assertion that anything more than two vertical bundles would yield diminishing returns, we will compare our design to a full 3D crossbar configuration (Section 3.4) with 25 vertical bundles. In all experiments, *the Full 3D crossbar was assumed to complete its configuration in a single clock cycle. Therefore, all results for this Full 3D Crossbar design will be very optimistic. However, despite discounting the complexity of the control and arbitration logic of the Full 3D crossbar, the proposed DimDe router will still be able to achieve comparable performance*.

The latency and throughput results of all five architectures for various synthetic traffic patterns (i.e. phase 1 of our simulation experiments) are illustrated in Figures 17 through 18. It can be observed that the proposed DimDe design consistently remains within 5% (on average) of the ideal Full 3D Crossbar's performance, while providing much lower latency and saturating at much higher workloads than the remaining architectures. Compared to the DimDe design, the 3D Symmetric topology suffers from the additional router delay at each inter-layer hop. At low loads (e.g., up to 20% for all traffic patterns with XYZ routing), the NoC-Bus Hybrid provides lower latency compared to the 3D Symmetric NoC as it benefits from the single hop vertical communication. As the load increases, the NoC-Bus Hybrid Architecture exhibits the worst latency and throughput for all traffic patterns (even worse than the 2D topology) as the bus saturates first with higher workload. Consequently, the 3D NoC-Bus Hybrid may be suitable only for 3D architectures where the traffic is mostly confined to the 2D strata and the load on the vertical links is sparse. Clearly, *the proposed 3D DimDe router outperforms the other three designs in all traffic patterns, and it achieves performance very close to that of a full crossbar, using only two (instead of 25) vertical bundles. This soundly resonates our assertions that a full 3D crossbar is design overkill in terms of performance enhancement*. The results with ZXY routing follow the same trends as with XYZ routing; thus, they are omitted for brevity.

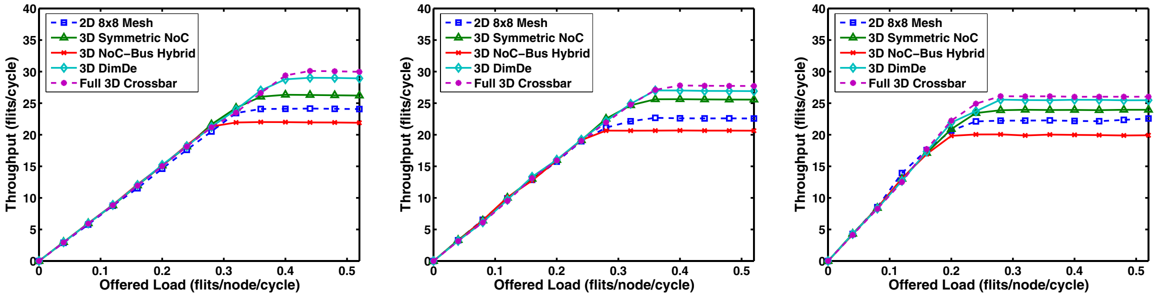
In terms of the throughput behavior (Figure 18), the DimDe architecture provides 18% average improvement over the other designs, while remaining within around 3% of the Full 3D Crossbar's throughput.

Figures 19 and 20 show the results of phase 2 of our simulation experiments, i.e. real commercial and scientific workloads in an 8-CPU CMP environment. Figure 19 depicts the average network latency for the four 3D architectures under test. We do not show the 2D results, since the significantly larger hop count in the 2D case naturally leads to substantially worse results compared to all 3D architectures. Clearly, the proposed 3D DimDe design outperforms all designs except the Full 3D Crossbar. DimDe provides an average improvement of 27% over the 3D Symmetric and 3D NoC-Bus Hybrid designs, and remains within 4% of the Full 3D Crossbar's performance. However, a more complete picture is painted in Fig-



(a) Uniform Traffic (b) Self-Similar Traffic (c) Transpose Traffic

Figure 17: Average Latency with various Synthetic Traffic Patterns (XYZ routing)



(a) Uniform Traffic (b) Self-similar Traffic (c) Transpose Traffic

Figure 18: Throughput with various Synthetic Traffic Patterns (XYZ Routing)

ure 20, which compares the Energy-Delay Product (EDP) of all the architectures. This metric is, in fact, more meaningful since it accounts for both performance and power consumption. Here, the efficiency of the proposed 3D DimDe design shines through. DimDe significantly outperforms all other designs, including the ideal Full 3D Crossbar, by 26% on average. These results underscore the efficiency of the DimDe architecture. *Through the decomposition of incoming traffic into smaller components, the use of a simple, partially-connected 3D crossbar, and reduced arbitration complexity, DimDe can outperform even the optimistic results of a full (i.e. 25 vertical bundles) crossbar structure. This result is of profound significance, because it shows that increasing inter-layer links arbitrarily increases design complexity and overhead without tangible performance benefits.*

6. CONCLUSIONS

3D technology is envisioned to provide a performance-rich, area- and energy-efficient, and temperature-aware design space for multicore/SoC architectures. In this context, the on-chip interconnect in a 3D setting will play a crucial role in optimizing the performance, area, energy and thermal behaviors. In this paper, we have explored several design options for 3D NoCs, specifically focusing on the inter-strata communication. Three possible designs that include a simple bus for the vertical connection, a symmetric 3D hop-by-hop topology, and a true 3D crossbar architecture are investigated. The proposed 3D architecture, called the 3D DimDe router, supports two vertical interconnects to achieve a balance between the path diversity and high bandwidth offered by a full 3D crossbar and the simplicity of a bus. DimDe supports a true 3D crossbar structure spanning all layers of the chip and fusing them into a single router entity. We have investigated the detailed microarchitectural implications of the design, which include the feasibility of the inter-strata vertical wire layout, arbitration mechanism, and virtual channel support for providing deadlock-free routing. The design has been implemented in structural Verilog and synthesized using a TSMC 90 nm standard cell library to analyze the area, energy, and thermal behaviors. It has been shown that it is possible to implement a hierarchical two-stage vertical arbitration mechanism.

To ensure a comprehensive evaluation environment, we utilized a double-faceted simulation process to expose all designs to several traffic patterns, including request/reply memory traffic. Phase 1 of the simulation used a stand-alone, cycle-accurate NoC simulator running synthetic workloads, while Phase 2 used a hybrid NoC/cache simulator running a variety of commercial and scientific workloads within the context of a multi-bank NUCA L2 cache in an 8-CPU CMP environment. In both cases, the proposed DimDe design was demonstrated to offer average latency and throughput improvements of more than 20% over the other 3D architectures, while remaining within 5% of the full 3D crossbar performance. More importantly, the DimDe architecture outperforms all other designs, including the full 3D crossbar, by an average of 26% in terms of the Energy-Delay Product (EDP). *One of the most important contributions of this work is the clear indication that arbitrarily adding vertical links in a 3D NoC router yields diminishing returns in terms of performance, while increasing control and arbitration complexity.*

This paper, to our knowledge, is the first attempt to explore the design of a 3D-crossbar-style NoC for upcoming 3D technology. 3D integration presents the interconnect designer with several new challenges. In the future, we plan to investigate the design of a pipelined arbitration scheme to support adaptive routing within the context of fault tolerance and load balancing.

Acknowledgment: The authors gratefully acknowledge Dr. Ravishankar Iyer and Donald Newell of Intel Corp. for their invaluable help with the workload traces and associated experiments.

7. REFERENCES

[1] International Technology Roadmap for Semiconductors (ITRS), 2005 edition, http://www.itrs.net/.

[2] Arteris, http://www.arteris.com/.

[3] STMicroelectronics Spidergon, http://www.st.com/stonline/.

[4] 70nm PTM technology model, http://www.eas.asu.edu/ ptm/.

[5] SAP Sales and Distribution Benchmark. http://www.sap.com/solutions/benchmark/index.epx.

[6] SPECjAppServer Java Application Server Benchmark. http://www.spec.org/jAppServer.

[7] SPECjbb2005 Java Business Benchmark. http://www.spec.org/jbb2005.

[8] TPC-C Design Document. http://www.tpc.org/tpcc/.

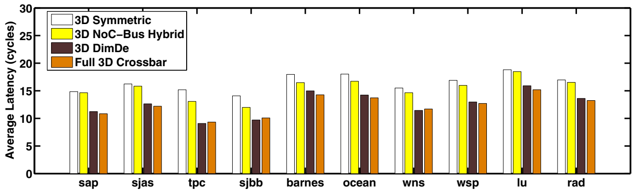


Figure 19: Average Latency with various Commercial and Scientific Workloads

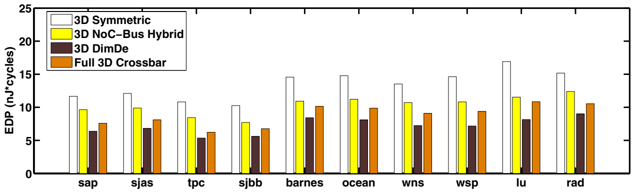


Figure 20: Energy-Delay Product (EDP) with various Commercial and Scientific Workloads

[9] B. M. Beckmann and D. A. Wood. Managing wire delay in large chip-multiprocessor caches. In *MICRO 37: Proceedings of the 37th annual IEEE/ACM International Symposium on Microarchitecture*, pages 319–330, 2004.

[10] L. Benini and G. D. Micheli. Networks on Chips: A New SoC Paradigm. *IEEE Computer*, 35(1):70–78, 2002.

[11] B. Black, M. M. Annavaram, E. Brekelbaum, J. DeVale, L. Jiang, G. H. Loh, D. McCauley, P. Morrow, D. W. Nelson, D. Pantuso, P. Reed, J. Rupley, S. Shankar, J. P. Shen, and C. Webb. Die stacking (3d) microarchitecture. In *MICRO'06: Proceedings of the 39th annual IEEE/ACM International Symposium on Microarchitecture, 2006*.

[12] A. Chandrakasan, W. J. Bowhill, and F. Fox. *Design of High-Performance Microprocessor Circuits*. IEEE Press, 2001.

[13] J. Cong and Y. Zhang. Thermal via planning for 3-D ICs. In *ICCAD '05: Proceedings of the 2005 IEEE/ACM International conference on Computer-aided design*, pages 745–752, Washington, DC, USA, 2005. IEEE Computer Society.

[14] W. J. Dally. Virtual-channel flow control. *IEEE Trans. on Parallel and Distributed Systems*, 3(2):194–205, 1992.

[15] W. J. Dally and C. L. Seitz. The torus routing chip. *Journal of Distributed Computing*, 1(3):187–196, 1986.

[16] W. J. Dally and B. Towles. Route Packets, Not Wires: On-Chip Interconnection Networks. In *Proceedings of the 38th Design Automation Conference*, June 2001.

[17] W. J. Dally and B. Towles. *Principles and Practices of Interconnection Networks*. Morgan Kaufmann, 2003.

[18] B. Dang, P. Joseph, M. Bakir, T. Spencer, P. Kohl, and J. Meindl. Wafer-level microfluidic cooling interconnects for GSI. In *Proceedings of the IEEE 2005 International Interconnect Technology Conference*, pages 180–182, 2005.

[19] W. R. Davis, J. Wilson, S. Mick, J. Xu, H. Hua, C. Mineo, A. M. Sule, M. Steer, and P. D. Franzon. Demystifying 3D ICs: The Pros and Cons of Going Vertical. *IEEE Design & Test of Computers*, 22(6):498–510, 2005.

[20] J. Duato. A New Theory of Deadlock-Free Adaptive Routing in Wormhole Networks. *IEEE Trans. on Parallel and Distributed Systems*, 4(12):1320–1331, December 1993.

[21] N. Eisley, L.-S. Peh, and L. Shang. In-network cache coherence. In *MICRO'06: Proceedings of the 39th annual IEEE/ACM International Symposium on Microarchitecture, 2006*.

[22] M. Galles. Scalable pipelined interconnect for distributed endpoint routing: The SGI Spider chip. In *Proceedings of The Symposium on Hot Interconnects*, pages 141–146, 1996.

[23] B. Goplen and S. Sapatnekar. Efficient thermal placement of standard cells in 3D ICs using a force directed approach. In *International Conference on Computer Aided Design (ICCAD)*, pages 86–89, 2003.

[24] S. Heo and K. Asanovic. Replacing global wires with an on-chip network: a power analysis. In *ISLPED '05: Proceedings of the 2005 international symposium on Low power electronics and design*, pages 369–374, 2005.

[25] R. Ho, K. W. Mai, and M. A. Horowitz. The Future of Wires. In *Proceedings of the IEEE*, pages 490–504, 2001.

[26] W.-L. Hung, G. Link, Y. Xie, N. Vijaykrishnan, and M. J. Irwin. Interconnect and Thermal-aware Floorplanning for 3D Microprocessors. In *7th International Symposium on Quality Electronic Design (ISQED)*, pages 98–104, 2006.

[27] R. Kessler and J. Schwarzmeier. Cray T3D: A new dimension for cray research. In *COMPCON '93: Proceedings of The IEEE Computer Society International Conference*, pages 176–182, 1993.

[28] T. Kgil, S. D'Souza, A. Saidi, N. Binkert, R. Dreslinski, S. Reinhardt, K. Flautner, and T. Mudge. PICOSERVER: Using 3D Stacking Technology to Enable a Compact Energy Efficient Chip Multiprocessor. In *Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-XII)*, 2006.

[29] J. Kim, W. J. Dally, B. Towles, and A. K. Gupta. Microarchitecture of a high-radix router. In *ISCA '05: Proceedings of the 32nd Annual International Symposium on Computer Architecture*, pages 420–431, Washington, DC, USA, 2005. IEEE Computer Society.

[30] J. Kim, C. Nicopoulos, D. Park, V. Narayanan, M. S. Yousif, and C. R. Das. A Gracefully Degrading and Energy-Efficient Modular Router Architecture for On-Chip Networks. In *33rd International Symposium on Computer Architecture (ISCA)*, pages 4–15, 2006.

[31] R. Kumar, V. Zyuban, and D. M. Tullsen. Interconnections in multi-core architectures: Understanding mechanisms, overheads and scaling. In *ISCA '05: Proceedings of the 32nd Annual International Symposium on Computer Architecture*, pages 408–419, Washington, DC, USA, 2005. IEEE Computer Society.

[32] F. Li, C. Nicopoulos, T. Richardson, Y. Xie, V. Narayanan, and M. Kandemir. Design and Management of 3D Chip Multiprocessors Using Network-in-Memory. In *33rd International Symposium on Computer Architecture (ISCA)*, pages 130–141, 2006.

[33] C. C. Liu, I. Ganusov, M. Burtscher, and S. Tiwari. Bridging the Processor-Memory Performance Gapwith 3D IC Technology. *IEEE Design & Test of Computers*, 22(6):556–564, 2005.

[34] P. S. 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*, 35(2):50–58, 2002.

[35] R. Marculescu. Networks-On-Chip: The Quest for On-Chip Fault-Tolerant Communication. In *Proceedings of the IEEE Computer Society Annual Symposium on VLSI (ISVLSI'03)*, 2003.

[36] R. Mullins, A. West, and S. Moore. Low-Latency Virtual-Channel Routers for On-Chip Networks. In *ISCA'04: Proceedings of The 31st Annual International Symposium on Computer Architecture*, June 2004.

[37] S. Mysore, B. Agrawal, N. Srivastava, S.-C. Lin, K. Banerjee, and T. Sherwood. Introspective 3d chips. In *ASPLOS'06: Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems*, 2006.

[38] P. R. Nuth and W. J. Dally. The J-machine network. In *Proc. The International Conference on Computer Design*, pages 420–423, 1992.

[39] L.-S. Peh and W. J. Dally. A Delay Model and Speculative Architecture for Pipelined Routers. In *Proceedings of the 7th International Symposium on High-Performance Computer Architecture*, 2001.

[40] K. Puttaswamy and G. H. Loh. Implementing Caches in a 3D Technology for High Performance Processors. In *ICCD '05: Proceedings of the 2005 International Conference on Computer Design*, pages 525-532, Washington, DC, USA, 2005. IEEE Computer Society.

[41] H. Sarbazi-Azad, M. Ould-Khaoua, and L. M. Mackenzie. Analytical modelling of wormhole-routed k-ary n-cubes in the presence of matrix-transpose traffic. *J. Parallel Distrib. Comput*, 62(4):605–621, 2002.

[42] L. Shang, L.-S. Peh, A. Kumar, and N. K. Jha. Thermal Modeling, Characterization and Management of On-Chip Networks. In *Proc. of the 37th MICRO*, 2004.

[43] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh, and A. Gupta. The splash-2 programs: characterization and methodological considerations. In *ISCA '95*, pages 24–36, 1995.

[44] A. Y. Zeng, J. J. Lü, K. Rose, and R. J. Gutmann. First-Order Performance Prediction of Cache Memory with Wafer-Level 3D Integration. *IEEE Design & Test of Computers*, 22(6):548–555, 2005.