Cost-driven 3D Integration with Interconnect Layers *

Xiaoxia Wu†, Guangyu Sun,
Xiangyu Dong, Reetuparna Das,
Yuan Xie, Chita Das
Pennsylvania State University
University Park, PA 16802
{xwu}@cse.psu.edu

Jian Li
IBM Austin Research Laboratory
Austin, TX 78758
{jianli}@us.ibm.com

ABSTRACT

The ever increasing die area of Chip Multiprocessors (CMPs) affects manufacturing yield, resulting in higher manufacture cost. Meanwhile, network-on-chip (NoC) has emerged as a promising and scalable solution for interconnecting the cores in CMPs, however it consumes significant portion of the total die area. In this paper, we propose to decouple the interconnect fabric from computing and storage layers, forming a separate layer called *Interconnect Service Layer* (ISL), in the context of three-dimensional (3D) chip integration. Such decoupling helps reduce the die area for each layer in 3D stacking. ISL itself can integrate multiple superimposed interconnect topologies. More importantly, ISL can be designed, manufactured, and tested as a separate Intellectual Property (IP) component, which supports multiple designs in the computing and storage layers. The resulting methodology also helps support different manufacturing volume in each die of 3D to reduce the overall manufacturing cost. We demonstrate the proposed methodology with an ISL design example and compare to its 2D and 3D counterparts without ISL support. The results show that 3D design with ISL not only provides significant cost reduction, but also achieves power-performance improvement thanks to the efficient usage of ISL.

Categories and Subject Descriptors

C.0 [**Computer Systems Organization**]: GENERAL—*Systems specification methodology*

General Terms

Design

Keywords

Three-dimensional Integrated Circuit, Network-on-Chip, Interconnect Service Layer

*We thank Lixin Zhang and Steve Vander Wiel for their guidance and support. This work was supported in part by NSF 0905365, 0903432, 0702617, and SRC grants.

†Xiaoxia Wu is now with Qualcomm, San Diego, CA 92121.

DAC 2010, June 13-18, 2010, Anaheim, California, USA.

1. INTRODUCTION

The diminishing return of endeavors to increase clock frequencies and exploit instruction level parallelism in a single processor have led to the advent of chip multiprocessors (CMPs). As the number of cores in CMPs increases aiming for higher computation throughput, die size gradually increases as well. Consequently, the manufacturing yield suffers, which leads to higher manufacturing cost. Meanwhile, networks-on-chip (NoC) has emerged as a promising and scalable solution for interconnecting the cores in CMPs.

A rich collection of NoC literature exist. Nonetheless, challenges for future many-core CMP design remain. For example, current NoC designs lack flexible support for cost-effective power-performance improvement of future many-core CMPs. First, it consumes much chip area and power with increased number of cores in CMPs, which makes the chip bigger, more power hungry, and constraining the number of cores (computing) and the capacity of on-chip cache memory (storage) [9]. Second, the current interconnect fabric is typically fixed for one chip design, since it is integrated with processors and cache memory within one single die. Reuse of the interconnect fabric for future chip generations is difficult, resulting in both design and manufacturing overhead.

On the other hand, three-dimensional (3D) integration technology has become a promising means to mitigate the power-performance related problems in conventional 2D chips, such as dominant interconnect delay and power consumption [7, 17]. In 3D ICs that are based on TSV technology, multiple active device layers are stacked together (through wafer stacking or die stacking) with direct vertical TSV interconnects [17]. One important benefit of 3D ICs is that it provides the opportunity of stacking dies with different technologies, processes, and vendors [17]. 3D technology also reduces the die area in each layer and may provide cost efficiency benefit [8].

Putting these together, we propose to decouple the interconnect fabric from the computing (core) and storage (cache) layers as a separate layer, called *Interconnect Service Layer* (ISL), in 3D stacking. This decoupling can provide reduced manufacture cost since each layer has smaller die area in 3D. It can offer more reliable and flexible interconnect layer compared to its traditional 2D counterparts. The decoupled ISL has the real estate for more than one on-chip network, e.g., it can support multiple on-chip networks in a single die such as mesh, ring, hierarchical topologies, etc. With ISL, the constraints on the router area and link bandwidth in 2D can be relaxed. It also supports different manufacture volume for each die in 3D to reduce the overall cost. For example, our proposed ISL can be manufactured with much larger volume than the other computing and storage layers. Then it can be bonded to those with varied designs, such as

different number of cores and storage capacity.

This paper makes the following contributions:

1) We map computing, communication and storage (including cache) functions to different layers in 3D stacking. Particularly, we extract communication functions as a single layer called ISL for flexible 3D integration.

2) The ISL can be designed, manufactured and tested as a separate IP component. This allows designs of more flexible and reconfigurable interconnect fabric. Particularly, we propose an architecture with *multiple superimposed heterogeneous networks* for ISL. The ISL can potentially consist of M multiple networks, each network providing a separate degree of flexibility and communication capacity. One or more of these M networks can be active simultaneously at runtime.

3) We evaluate one specific example of ISL and demonstrate the cost benefit and performance improvement. The evaluation results show that the cost reduction of our proposed architecture could be up to 40% compared to 2D case. The performance improvements are 21% and 6.5% in average compared to 2D and 3D without ISL design, respectively.

4) We extend existing 3D cost models by modeling the number of TSVs for power delivery, differentiating cost models between different functional layers, and addressing product volume factor of each layer in 3D integration.

2. INTERCONNECT SERVICE LAYER

Typical 3D stacking architectures include logic-to-logic stacking [4] and logic-to-cache/memory stacking [16]. Figure 1 (a) illustrates a logic-to-cache structure. In this structure, the cache layer is stacked to the computing (processor) layer, while the interconnect network is integrated in computing and cache layers. Figure 1(b) illustrates the proposed 3D stacking structure with *interconnect service layer* (ISL). The interconnect layer consists of routers and links to connect the computing layer and cache layer. Since the whole layer is dedicated for the interconnect it has room to support multiple networks of different granularity and topologies, such as mesh, ring, hierarchical topology, etc.

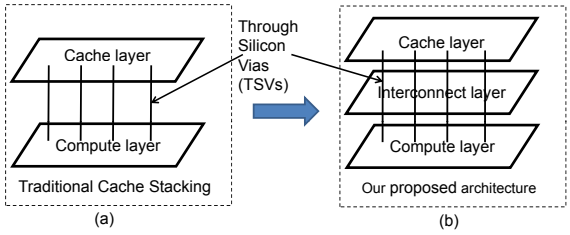

Figure 1: (a) Logic-cache 3D stacking. (b) 3D stacking with interconnect service layer.

2.1 Advantages

A separate *interconnect service layer* (ISL) decouples the active logic components (computing and storage) from interconnect fabric, reducing greatly the pre-fabrication and post-fabrication verification complexity, hence cost. The ISL can be designed, manufactured and tested as a separate IP component. This allows network architects to design more flexible, adaptive and reconfigurable interconnect fabric. We propose *multiple superimposed heterogeneous networks* in ISL. The ISL can potentially consist of M multiple networks, each providing a separate degree of flexibility. One or more of these M networks can be active simultaneously at runtime. We enumerate the benefits of ISL in the following:

Latency. One or more of the M multiple networks can be optimized for providing low latency for latency critical applications. For example, concentrated and richly connected topologies (e.g., Flattened Butterfly and Hierarchical) are oriented best towards providing low latency.

Bandwidth. One or more of M multiple networks can be optimized for providing high bandwidth. For example, a flat topology with many routers and many buffers can provide high throughput. The throughput provided by mesh can be 2X higher than butterfly or hierarchical topology.

Power. One or more of M multiple networks can be optimized for low power to meet the stringent power constraints imposed by computing and storage layers. The power efficiency of a network can be programmed by using different frequency domains, as well as using small data path widths for routers and channels.

2.2 ISL Example

In this section, we describe one specific example of the ISL design with two meshes of different granularity. We assume a range of CMP designs with up to 36 processor cores. Figure 2 illustrates a possible configuration for the proposed 3D stacking architecture. There are two superimposed meshes in the interconnect layer: 6x6 fine-grained mesh and 3x3 coarse-grained mesh, which are highlighted in dark and light, respectively. The reason we choose mesh topologies is because of its simplicity and its scalability for global network [6]. In a 36-core design, each router in the 6x6 fine-grained mesh is connected to one core. Four cores are in a cluster in the 3x3 coarse-grained mesh. A bus, cross-bar or point-to-point interface can be used to connect the two meshes. Each mesh is self-sufficient and supports typical communication traffic. With this interconnect layer, we can stack a processor layer with 36 small cores (6x6) and their caches, 9 big cores (3x3) with their caches, 3 big cores and 24 small cores and their caches, etc, all of which flexibly use the same interconnect layer for chip stacking.

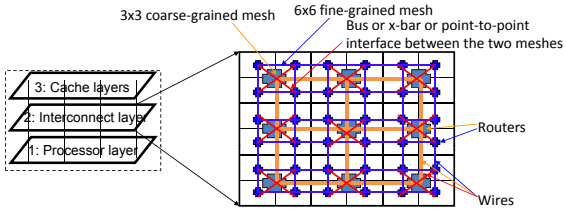

Figure 2: An ISL example for 3D chip designs of up to 36 (6x6) cores.

The two meshes can be used coordinately to improve the performance. For 6x6 multi-core integration, 3x3 coarse-grained mesh provides fast-path cross-chip interconnect. For example, if there is communication between upper left corner core and lower right corner core, 10 hops are needed if no 3x3 coarse-grained mesh is integrated. With the coarse-grained mesh, only 6 worst-case hops are needed. Figure 3 shows an elaboration on supporting 3x3 cores with 6x6 cache banks. Each core has its corresponding 4 cache banks. The fine-grained interconnect fabric supports local data communication between neighbor cores/caches, whether or not the cores belong to the same cluster under a router in the coarse-grained mesh. The coarse-grained interconnect supports faster global data communication between cores/caches that are further apart. For example, if there is communication between Core 0 and Core 8, there are fewer hops and higher link bandwidth between them.

2.3 Architecture

In this section, we present the design of the ISL such as the router design, the TSVs connections, and the routing

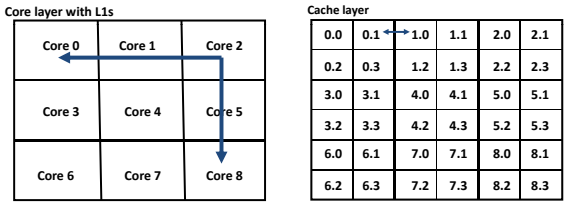

Figure 3: Elaboration on supporting 3x3 cores with 6x6 cache banks.

scheme to support the superimposed mesh topology.

2.3.1 Router Microarchitecture

The key concept is to use NoC routers for communications within the interconnect layer, while using a specific through silicon bus (TSB) for communications among different layers. Figure 4 illustrates an example of the structure. There are 4 cores located in the core layer, 4 routers in the interconnect layer, and 16 cache banks in the cache layer and all layers are connected by through silicon bus (TSB) which is implemented with TSVs. This interconnect style has the advantage of short connections provided by 3D integrations. It has been reported the vertical latency of traversing a 20-layer stack is only 12ps [12], thus the latency of TSB is negligible compared to the latency of 2D NoC routers. Consequently, it is feasible to have single-hop vertical communications by utilizing TSBs. In addition, hybridization of 2D NoC routers with TSBs requires one (instead of two) additional ports on each NoC router, because TSB can move data both upward and downward [10].

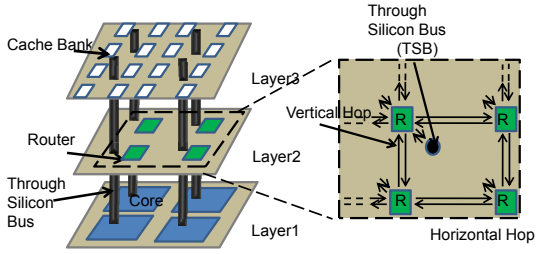

Figure 4: Elaboration of TSBs and routers in a setup of 2x2 cores with 4x4 cache banks.

2.3.2 Routing

The interconnect service layer provides routing both *within* each superimposed network and *between* them. The need for routing between the independent networks is to facilitate flexibility and improve utilization of network resources. Intra-network communication can be supported by simple baseline routing schemes. Simple extension to default dimension-ordered-routing can be implemented if the network supports a regular topology. For example, our previous example has the mesh topology, which we evaluate in this paper. If any one of the networks has irregular topology, application-specific architecture or flow-based topology mapping for guaranteed services, the router needs to support table-based routing and arbitration. To enable inter-network communication, special inter-network input/output ports are provided at *specific routers* of each independent network which connect it to other networks. Thus each independent superimposed network has fewer designated "interface routers" which connect it to separate networks. To avoid possibility of deadlock at interface routers, egress and ingress traffic have dedicated virtual channels in these routers. In addition to operating the superimposed networks independently, the interface router's routing tables can also be programmed for fusion of multiple networks into a larger monolithic network. We skip the rich details of this design space due to limited space.

3. 3D COST MODEL

In 3D integration, the manufacturing cost reduction may come from smaller die area of each layer as well as reduced number of metals for routing. The number of metals is predicted by a 3D routability model, which is based on the wire length distribution [8]. As described in [8], when a large 2D chip is partitioned into multiple smaller dies in 3D, the cost could be reduced due to fewer number of metal layers needed for each smaller die although extra bonding cost is needed in 3D stacking.

Our cost model is based on a few prior art, particularly the one by Dong et al. [8] with several improvements: (1) It models the number of TSVs for power delivery; (2) It differentiates cost models for different layers (logic/cache/interconnect); 3) It adds mask cost, design cost, addresses (product) volume factor of each layer, and addresses (production) time factor of each layer.

It is important to estimate the number of TSVs and its impact on the die area in 3D stacking since the area overhead caused by TSVs could be significant depending on the TSV pitch and the design. The TSVs modeled in [8] includes only signal TSVs while the TSVs used for power delivery is not considered. However, for example, the power grid distribution via TSVs in 3D is important to leverage on-chip power density. Therefore, we take into account the TSVs for power delivery.

The estimation on the number of TSVs for power delivery is based on voltage drop caused by TSVs. Assume the allowed maximum voltage drop is $d\%$ and the resistance of one single TSV is R_{tsv}, the number of TSVs is estimated by

$$N_{tsvp} = R_{tsv}/(d\% * V/(P/V)) = R_{tsv} * P/0.01 * d * V^2$$

where N_{tsvp} is the minimum number of TSVs needed for power delivery, P is the total power consumption, V is the power supply voltage. Based on the power and the voltage, the total resistance caused by TSVs can be estimated. Since all the resistance of TSVs are connected in parallel, the number of TSVs needed is obtained from the total resistance and the resistance of one single TSV.

In addition to adding TSVs for power, we also distinguish the cost for different layers such as logic layer (core and interconnect) and cache layer. The first reason is that cache layer may use fewer number of metal layers than the logic layer due to its regularity. Second, different layers may not have the same die area so that it is necessary to differentiate the cost for each single layer.

Another improvement is that we address (product) volume factor of each layer. In our proposed architecture, the interconnect layer "ISL" can be reused in different 3D designs because it is designed, manufactured and tested as a separate IP component and it could provide multiple superimposed heterogeneous networks. This allows ISL to be stacked with different functional units (different number of cores) and various capacity of storage depending on different applications/designs to reduce the total cost. After considering volume, the cost of each die (layer) is defined as follows [14]: $C = NRE/Volume + Cost_{die}$, where NRE stands for non-recurring engineering, including mask and design cost, $Cost_{die}$ is calculated using the model from [8], which mainly considers the die cost including the wafer cost, the wafer yield, the defect density, and the extra cost caused by 3D bonding. We estimate the design cost and the mask cost based on [5], in which these two metrics at different technology nodes are provided.

The overview of our 3D cost model is shown in Fig. 5, with our improvements indicated in the three ovals in the right side. Similar to [8], the key factor of the die cost model is the die area. We assume that the wafer cost, the wafer yield, and the defect density are constant for a specific foundry using a specific technology node. The extra fabrication steps required by 3D integrations consist of TSV forming, thinning, and bonding. The entire 3D cost model also depends on the volume of each single die and some design options, such as Die-to-Wafer/Wafer-to-Wafer bonding, Face-to-Face/Face-to-Back bonding, and Known-Good-Die cost in addition to the wafer cost model and the bonding cost model.

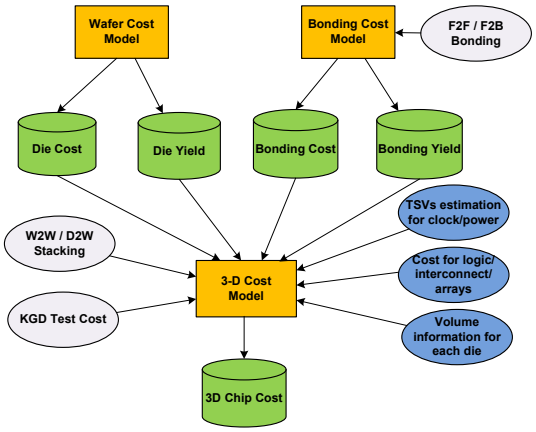

Figure 5: Overview of the proposed 3D cost model.

4. METHODOLOGY

System Configuration: Our baseline configuration is a 36-core in-order processor using the Ultra SPARC-III ISA. We use McPAT [11], an integrated power, area, and timing modeling framework, to estimate the area of the cores in 45nm technology. The area of one core is estimated to be $6.8mm^2$. By using CACTI [15], we further obtain that one cache layer fits to approximately 36MB SRAM L2 cache, assume the cache layer has similar area to the core layer. The configurations are detailed in Table 1. We use the Simics toolset [13] for performance simulations. We also evaluate 9-core processor with 9MB L2 cache configuration for the performance using different network topologies. The parameters will be described in Section 5.

Table 1: System configuration.

Processor	36-core, in order, 2GHz
L1	32KB DL1/IL1 per core, 128B, 2-way, 2 cycles
L2	36MB shared cache, 1MB per bank, 8-way, 10 cycles
Memory	400 cycles latency, 16MB large page
Router lat.	5 cycles

Workloads: We use a set of workloads from SpecOMP2001[1] and PARSEC [3]. For each benchmark, we fast forward the benchmark to the program phase of interest and warm up the caches, then 3 billions of cycles are simulated in detailed mode. The instruction throughput of all the cores are used as the metric of performance.

5. EXPERIMENTS AND RESULTS

In this section, we evaluate ISL, with the example of superimposed mesh networks (Section 2), by comparison against its 2D counterpart and a 3D design without ISL, in chip area, cost, and performance.

5.1 Area estimation

In 3D stacking, multiple layers may or may not have the same die area. As we mentioned in Section 3, the total cost of 3D stacking is related to the area of each layer and we distinguish the cost for different layers. For our ISL design, we can distribute the routers aligned with the cores (one router is vertically below one core). Or, we can minimize the area of this layer by centralizing/shrinking the routers with redistribution interconnect layer in both core layer and cache layer, if the two mesh networks consume less real estate than other two layers. In the latter, one extra metal may be added to the core and cache layers for the horizontal routing. The area of ISL includes router area and link area for two mesh topologies, the interface area and link area to connect two networks, and TSV area occupied for signal and power delivery.

We estimate the router area A_{router} by using McPAT [11] based on 45nm technology. In order to support layer-to-layer communication in 3D stacking, hybridization of 2D NoC routers with Through Silicon Buses (TSB, which consists of many TSVs) requires one additional link on each NoC router, because TSB can move data both upward and downward [10]. We feed related parameters into McPAT and obtain the router area $0.93mm^2$. The area of the link depends on the wire width and space as well as wire length, which can be changed in ISL depending on the area of routers and the interface design between two mesh topologies.

In our ISL example, there are 45 signal TSBs (6x6 + 3x3 = 45) pierce through all three layers. We provide one TSB for each core in the structure. Consider the range of TSV pitch size of $1 - 100\mu m$, the area of one TSB A_{TSB} for a 1024-bit TSB (one cache line) would consume different area overhead depending on the TSV pitch. Note that we assume face-to-back bonding in our 3D design so that TSVs are formed by punching through the silicon substrate, resulting in extra area. Table 2 summarizes the area for a single TSB (1024 TSVs) depending on different TSV pitch. If the TSV pitch is smaller than $10\mu m$, the TSB area overhead may be negligible. However, when the TSV pitch is larger than $60\mu m$, the area of one TSB is even comparable to the area of one core. We assume the maximum TSV pitch in our cost evaluation is $40\mu m$. The estimation of TSVs for the power delivery is based on the model proposed in Section 3. Assume P is $100W$, supply voltage is $1Volt$, the resistance of one single TSV is $40m\Omega$ [2], and the voltage drop is 1%, then the number of TSVs needed is 400. Considering larger TSV resistance and higher power consumption, the number of TSVs for power delivery is increased but we expect it will not exceed 2 TSBs used for signals (2048). To summarize, we consider the area overhead caused by TSVs to be 47 TSBs.

Table 2: The area of one TSB (1024 TSVs).

TSV pitch(μm)	1	10	20	40	60	100
TSB area(mm^2)	0.001	0.103	0.411	1.645	3.699	10.28

We use a bus to connect the two meshes, due to its high bandwidth and low latency for local communications [6]. If we assume the link/bus can be routed over the routers then the minimum area for the ISL is only the area of routers plus some routing overhead. Nonetheless, a range of ISL area is evaluated in the cost analysis.

5.2 Cost analysis

5.2.1 *Die cost without volume analysis*

In this section, we analyze the die cost (without volume consideration) for ISL design with different area consumption and compare it against its 2D counterpart and a 3D

design without ISL. In our ISL example design, the area of the core and cache layer is $228mm^2$ without TSV area overhead. If the TSV area overhead is considered then the maximum extra area caused by TSVs is $75mm^2$ ($40\mu m$ pitch), which is 1/4 of the total die area of processor die. The area of ISL may be changed depending on the router design, the interface design between two mesh networks, and the TSV area overhead. In order to perform comprehensive analysis, the ISL area is varied from $60mm^2$ to $228mm^2$ without TSV overhead. We also evaluate TSV area overhead with different TSV pitches. Three TSV area overhead cases are evaluated: $0mm^2$, $30mm^2$, and $75mm^2$. The default number of metal layers is 9, 6, and 6 for core layer, cache layer, and ISL, respectively. When ISL area is smaller than core and cache layers, one extra metal layer is needed in core and cache layers for horizontal routing to connect to TSBs. Fig. 6 illustrates the cost comparison for different ISL area and TSV area overhead. The X-axis represents the ISL area for different routers and interface design. We observe that the total cost increases with the increased ISL area. When the ISL area is smaller than the core and cache layers one extra metal is needed for core and cache layer for the routing; however, the cost reduction caused by reduced ISL area offsets the extra metal cost. We also observe that the extra cost caused by TSV area overhead could be significant if TSV pitch is large. This indicates that it is important to take the TSV area into account when we evaluate 3D at early design stage.

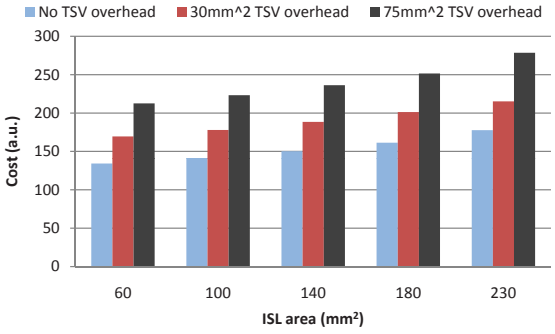

Figure 6: Cost comparison for different ISL area and TSV area overhead.

We also evaluate the cost for 2D design and 3D design without ISL. For 2D design, we place 36 cores, 36 L2 cache banks, and fine-grained mesh network in one layer. Without loosing generality, the area of the mesh network also varies depending on the router design and the link area. The 36 cores and 36 cache banks consume $460mm^2$. We assume the total area ranges from 510 to $600mm^2$ (510, 540, 570, 600 with 9 metals), the total cost is 258.49, 289.80, 322.53, and 359.61 for these four cases, respectively. We see that even with smallest network area, the cost of 2D design is much higher than that of 3D with ISL (maximum area) or comparable to 3D with ISL with maximum TSV area overhead. The cost reduction of 3D with ISL (maximum area) is about 40% compared to 2D even with the smallest interconnect area if no TSV area overhead is considered. However, if maximum TSV area overhead is considered in 3D, the cost is about 6% higher than 2D with smallest network area.

For 3D design without ISL, the routers are integrated with cache layer (or computing layer) so that there are two layers in total. Similarly, the area of these two layers is changed from 280 to $400mm^2$ (6 cases: 280, 300, 320, 340, 360 and 400) depending on the router design and TSV area overhead. The total cost is 164.84, 184.58, 206.57, 229.8, 258.47, 286.09, 313.61 for these 6 cases. We find that 3D with ISL is most cost efficient if TSV area overhead is reasonable but the exact boundary for TSV pitch depends on the design itself. Therefore, it is essential to take the TSV area, including signal and power delivery etc, into account, when we evaluate a 3D design at its early stage.

5.2.2 *Volume analysis*

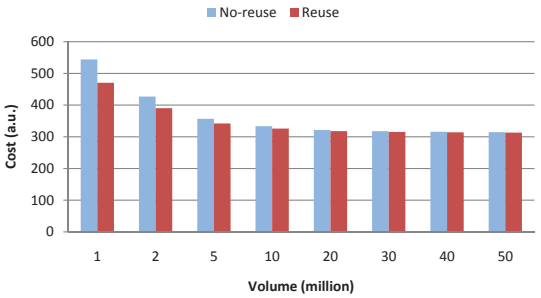

Figure 7: Volume analysis for non-reuse and reuse.

In our ISL architecture design, the ISL could be stacked with different functional units (e.g., different number of cores) and various capacity of storage (different capacity of L2 Cache). With die reuse, the total cost for multiple applications could be reduced. Assume we have two 3D designs, each with 3 layers. The first design has 36 cores as layer 1, interconnect layer as layer 2, 36M L2 Cache as layer 3. The second design has 9 cores as layer 1, interconnect layer as layer 2, 9M L2 Cache as layer 3. If no reuse is enabled, the total cost for these two designs is calculated as

$$Cost_{total} = \sum_{i=1}^{6}(NRE_i/Volume_i) + \sum_{j=1}^{2} 3D_j$$

There are total 6 layers in these two designs and each layer has its own NRE cost and volume. With the reuse, since the NRE part of the total cost is reduced, i.e., two ISL layers use the same design, thus only one design cost and mask cost need to be considered. Here we provide an example to illustrate the volume impact on the total cost for these two applications.

We obtain $3D_j$ from our cost model without volume consideration. For volume related cost at 45nm technology, the design cost and mask cost are either from or scaled from [5]. Note that we assume 3D design does not introduce extra design cost compared to 2D once the 3D design flow is mature. Fig. 7 illustrates the cost comparison between the non-reuse and reuse cases. The result shows that the reuse is very cost effective when the volume is low. However, when the volume is very high, the NRE cost is not dominant then the cost difference is reduced. Note that if the cache layer can also be reused (have the same capacity for these two applications) then the cost reduction will be improved.

5.3 Performance result

We first examine the performance of the example ISL architecture with 36 cores. The cache size is estimated based on the area of cache layer using Cacti [15]. For 36-core system, we assume there is a 36-bank shared L2 cache with 36MB capacity. Each core is connected to a cache bank using TSB. The cache controller is integrated with the processor layer. The floorplans for these three cases are shown in Fig. 8 (only 4 cores are illustrated for the simplicity). Fig. 8(a) illustrates 2D configuration, in which one core and one cache bank is placed together and connected to other

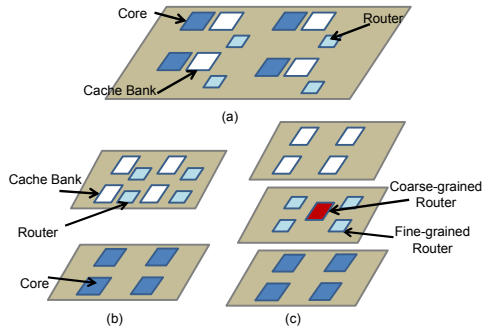

Figure 8: System configurations for 2D, 3DB, and 3DI cases.

cache/bank groups by routers. Fig. 8(b) shows 3D baseline (3DB), in which the fine-grained mesh network is integrated with core layer. Fig. 8(c) shows 3D design with ISL (3DI), in which both coarse-grained and fine-grained mesh networks are integrated in ISL. In 3DI, the coarse-grained mesh reduces the number of hops for global communication. In 2D case and 3DB cases, there is only fine-grained mesh network to support the communication. In 2D case, the link latency is larger due to longer link between two routers. Fig. 9 illustrates the IPC comparison of these three cases, normalized to 2D case. While it varies between applications, the result shows that 3DI achieves 21% and 6.5% average performance improvement over 2D and 3DB, respectively.

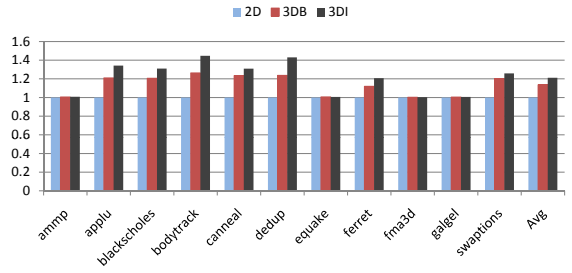

Figure 9: Performance comparison among 2D, 3DB, and 3DI cases.

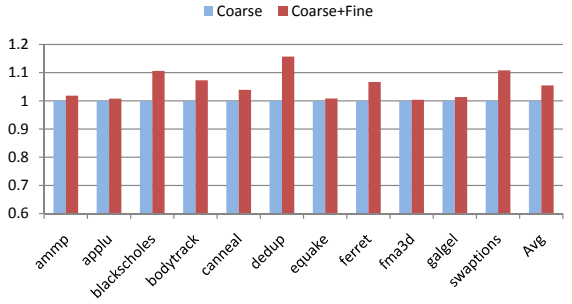

Figure 10: Performance comparison between multiple networks and coarse-grained network.

We also evaluate the performance of the example ISL architecture with 9 cores, and compare the proposed superimposed (coarse+fine grained) networks with coarse-grained mesh network. For 9-core system, we assume there is a 36-bank shared L2 cache with 9MB capacity. The performance comparison is shown in Fig. 10. The result shows that ISL with superimposed (coarse+fine grained) networks has 5.5% performance improvement than the coarse-grained network. In summary, we show that our ISL design improves performance by effectively using these two mesh topologies for different system configurations.

Finally, we have shown the proposed ISL design improve performance by supporting the selection of networks for efficient data communication, e.g., reducing hop counts. As a result, the interconnect power consumption is reduced accordingly. In addition, the thermal is expected not a big concern with reduced power consumption. Due to limited space, we skip the power and thermal evaluation results in this paper.

6. CONCLUSION

We have demonstrated a methodology that decouples the interconnect fabric from computing and storage layers, forming a single layer called ISL, in emerging three-dimensional chip integration technology. This decoupling can reduce manufacture cost thanks to smaller die area for each layer in 3D. It also supports different manufacture volume for each die in 3D to reduce the overall chip cost. As an example, we have proposed to superimpose multiple networks in the interconnect service layer for flexible 3D integration. We have extended the state-of-the-art 3D cost model in this work. Our evaluation shows that a 3D design with ISL not only provides significant cost benefits but also achieves performance-power improvement, compared to its conventional 2D and 3D counterparts.

7. REFERENCES

[1] http://www.spec.org/. 2001.

[2] S. Alam, R. Jones, S. Rauf, and R. Chatterjee. Inter-Strata Connection Characteristics and Signal Transmission in Three-Dimensional (3D) Integration Technology. In *ISQED*, pages 580–585, March 2007.

[3] C. Bienia, S. Kumar, J. P. Singh, and K. Li. The PARSEC Benchmark Suite: Characterization and Architectural Implications. In *PACT*, pages 72–81, October 2008.

[4] B. Black, M. Annavaram, N. Brekelbaum, J. DeVale, L. Jiang, G. H. Loh, D. McCaule, P. Morrow, D. W. Nelson, D. Pantuso, P. Reed, J. Rupley, S. Shankar, J. Shen, and C. Webb. Die Stacking (3D) Microarchitecture. In *MICRO*, pages 469–479, 2006.

[5] A. Chang. Case Study of a 65-nm SoC Design. *Design & Test of Computers, IEEE*, 26(2):14–19, March-April 2009.

[6] R. Das, S. Eachempati, A. Mishra, V. Narayanan, and C. Das. Design and evaluation of a hierarchical on-chip interconnect for next-generation CMPs. In *HPCA*, pages 175–186, Feb. 2009.

[7] 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 and Test of Computers*, 22(6):498– 510, 2005.

[8] X. Dong and Y. Xie. System-level cost analysis and design exploration for three-dimensional integrated circuits (3D ICs). In *ASP-DAC*, pages 234–241, 2009.

[9] R. Kumar, V. Zyuban, and D. M. Tullsen. Interconnections in Multi-Core Architectures: Understanding Mechanisms, Overheads and Scaling. In *ISCA*, pages 408–419, 2005.

[10] F. Li, C. Nicopoulos, T. Richardson, Y. Xie, V. Narayanan, and M. Kandemir. Design and Management of 3D Chip Multiprocessors Using Network-in-Memory. *ISCA*, 34(2):130–141, 2006.

[11] S. Li, J. H. Ahn, R. D. Strong, J. B. Brockman, D. M. Tullsen, and N. P. Jouppi. McPAT: An Integrated Power, Area, and Timing Modeling Framework for Multicore and Manycore Architectures. In *MICRO*, pages 469–480, 2009.

[12] G. 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*, pages 991–996, 2006.

[13] P. S. Magnusson, M. Christensson, J. Eskilson, et al. Simics: A Full System Simulation Platform. *Computer*, 35(2):50–58, 2002.

[14] J. M. Rabaey, A. Chandrakasan, and B. Nikolic. *Digital Integrated Circuits-a design perspective*. Prentice Hall, 2003.

[15] P. Shivakumar and N. Joupp. Cacti 3.0: An integrated cache timing, power, and area model. In *Western Research Lab Research Report*, 2001.

[16] X. Wu, J. Li, L. Zhang, E. Speight, R. Rajamony, and Y. Xie. Hybrid cache architecture with disparate memory technologies. In *ISCA*, pages 34–45, 2009.

[17] Y. Xie, G. H. Loh, B. Black, and K. Bernstein. Design Space Exploration for 3D Architectures. *Journal on Emerging Technologies in Computing Systems*, 2(2):65–103, 2006.