DESIGNING HIGH-PERFORMANCE MICROPROCESSORS IN 3-DIMENSIONAL INTEGRATION TECHNOLOGY

A Thesis
Presented to
The Academic Faculty

by

Kiran Puttaswamy

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
December 2007

DESIGNING HIGH-PERFORMANCE MICROPROCESSORS IN 3-DIMENSIONAL INTEGRATION TECHNOLOGY

Approved by:

Professor Gabriel H. Loh, Advisor
College of Computing
Georgia Institute of Technology

Professor Sung Kyu Lim
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Hsien-Hsin S. Lee
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Milos Prvulovic
College of Computing
Georgia Institute of Technology

Professor Sudhakar Yalamanchili
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Douglas Yoder
School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: September 13, 2007

Family,

for being who they are

and

Boy(s) Who Survived,

for not blinking during the duel and for teaching me how to make lemonade

ACKNOWLEDGEMENTS

I express my sincere gratitude to Professor Gabriel H. Loh for his energy and patience in mentoring me and guiding my thesis. I feel immensely fortunate to have had a chance to work with him and consider my research collaboration with him as one of my best career decisions. I will forever remember him for his enthusiastic and passionate attitude and his superb mentoring skills. I thank Professor Hsien-Hsin S. Lee for agreeing to be my co-advisor and guiding me on various occasions. His energy seems boundless and he has always been a source of information to me on various technical and career-related issues. I thank Professor Sung Kyu Lim for research collaboration and for serving as a reader on my thesis committee. He has provided valuable feedback on various aspects of my research and has helped enrich the content of my dissertation. I thank Professor Sudhakar Yalamanchili for serving as chairman of my thesis proposal committee and for always responding promptly to my requests. I thank my thesis committee members, Professors Douglas Yoder and Milos Prvulovic for their valuable feedback and suggestions to improve my work.

I thank my friends Samantika, Guru, Ioannis, Mrinmoy, Richard, Dong Hyuk, Dean, Eric, Michael, Dae Hyun, Nive, and Raghu for their friendship over the years. I thank Susie and Deborah in the College of Computing for assistance in administrative matters. I thank Marilou in the ECE Graduate Affairs Office for helping and providing information during various points of my graduate student life. I thank the various program committees and reviewers for their constructive feedback on my research papers. I acknowledge Intel Corporation and FCRP for research equipment and funding.

My mom Vanaja, my dad Puttaswamy, my sister Kanchan, and my brother-in-law Suresh have encouraged and supported me to attend graduate school and for that, I express my deepest gratitude and appreciation. Finally, I thank my wife, Rohini for making a tremendous difference to my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Thesis statement: 3D-integration technology provides simultaneous performance and power benefits to build high-performance microprocessors, while keeping the worst-case temperature under control.

The main contribution of this dissertation is the demonstration of the impact of a new emerging technology called 3D-integration technology on conventional high-performance microprocessors. 3D-integration technology stacks active devices in the vertical dimension in addition to the conventional horizontal dimension. The additional degree of connectivity in the vertical dimension enables circuit designers to replace long horizontal wires with short vertical interconnects, thus reducing delay, power consumption, and area.

To adapt planar microarchitectures to 3D-integrated designs, we study several building blocks that together comprise a substantial portion of a processor's total transistor count. In particular, we focus our attention on three basic circuit classes: static random access memory (SRAM) circuits (e.g., caches, register files), associative/CAM logic circuits (e.g., instruction schedulers, load/store queues), and data processing circuits (e.g., adders, shifters, multipliers) in conventional high-performance processors. We propose 2-die-stacked and 4-die-stacked 3D-integrated circuits and demonstrate different designs to deal with the constraints of the conventional planar technology.

Based on the data and the insights gained from the 3D-integrated circuits, we propose high-performance 3D-integrated microprocessors and evaluate the impact on performance, power, and temperature. We propose 3D-integrated microprocessor designs based on the Alpha architecture and demonstrate two different approaches to improve performance: clock speed (3D-integrated processors with identical microarchitectural configurations as the corresponding planar processor run at a higher clock frequency), and IPC (3D-integrated processors accommodate larger-sized modules than the planar processors for the same frequency). These processors demonstrate the simultaneous benefits of the 3D-integration and highlight the power density and thermal issues related to the

3D-integration technology.

Next, we propose 3D-integrated microprocessor designs based on the Intel Core microarchitecture. We propose novel microarchitectural techniques based on significance partitioning and data-width locality to effectively address the challenges of power density and temperature. We demonstrate that our microarchitecture-level techniques can effectively control the power density and temperature issues in the 3D-integrated processors.

The 3D-integrated processors provide a significant performance benefit over the planar processors while simultaneously reducing the total power. The simultaneous benefits in multiple objectives make 3D-integration a highly desirable technology for use in building future microprocessors. One of the key contributions of this dissertation is the temperature analysis that shows that the worst-case temperatures on the 3D-integrated processors can be effectively controlled using microarchitecture-level techniques. The 3D-integration technology may extend the applicability of Moore's law for a few more technology generations.

Part I

"A beginning is the time for taking the most delicate care that the balances are correct."

– Frank Herbert, *Dune*, 1965.

CHAPTER I

INTRODUCTION

1.1 Motivation

Relentless technology scaling has posed some new challenges to the semiconductor industry. Some of the technology challenges in the modern era include poor scaling of interconnect delays [18, 122], increasing power consumption [22, 138], and manufacturing challenges [64, 93, 150, 19]. The semiconductor industry must overcome such challenges to keep pace with Moore's law [102] and industry projections [64, 129]. 3D-integration technology is a new technology that has the potential to address many of the challenges facing the semiconductor industry. In a conventional planar (2D) technology, floorplanning and layout constraints may force two connected circuits to be physically separated, thus requiring global wires for communication. In a 3D organization, these circuits may be stacked on top of each other, thus replacing long global wires with short vertical interconnects. The 3D interconnects may be realized at different levels of the design hierarchy: at the package level (3D system-on-a-package) [84], at the chip level (3D system-on-a-chip) [77], and at the circuit level (3D integrated circuit or 3D IC) [37].

3D integration provides increased device density, reduced latency, and lower power [120, 109, 148, 113, 159, 110, 157, 112]. Each transistor can access a greater number of adjacent transistors (due to vertical connectivity) leading to higher bandwidth [67]. The relative benefits of the 3D-integration technology will increase in future technology generations, making it a very attractive option for future designs. There has recently been a great deal of interest in 3D-integrated circuits. Prior research includes studies on 3D-integrated caches [120, 109, 148, 95, 14], 3D-integrated register files [111], 3D-integrated arithmetic units [95, 112, 154, 114], 3D-integrated CAM circuits [110, 157], clocking schemes for 3D-integrated circuits [101], 3D-integrated processors [13, 157, 90, 88, 115, 105, 72], 3D-integrated systems-on-a-chip [39, 118], 3D-integrated FPGAs [86, 1, 85] and design automation tools for 3D-integrated designs [27, 32, 36, 45, 118, 157].

The microprocessor industry is evaluating the 3D-integration technology for feasibility and applicability [13, 120, 15, 50, 51]. The embedded processor industry is already offering products such as 3D-integrated SRAMs, 3D-integrated DRAMs, and 3D-integrated micro-controllers with SRAMs [145, 125]. 3D-integration also enables heterogeneous technologies (CMOS, DRAM, analog) to be integrated on different layers of a 3D stack, thus providing superior form-factors and greater functionality [88, 72, 128].

The rest of the chapter is organized as follows: Section 1.2 describes some of the current challenges to the semiconductor industry and shows the potential of the 3D-integration technology in addressing these challenges. Section 1.3 provides the background on the 3D-integration technology. Section 1.4 sets up the scope of this dissertation and concludes the chapter.

1.2 Current Technology Challenges

1.2.1 Interconnect Delay

Interconnect delay has become a limiting factor in the integrated circuit performance. In keeping with Moore's law, (1) the transistor sizes decrease, and (2) the transistor switching speeds increase with successive technology scaling generations. The reduced size of the transistor enables higher integration density by providing more transistors in the same area as previous technology generations. The increased switching speed of the transistor enables a higher frequency of operation. The improved integration density and the increased speed together enable higher functionality and consequently higher overall performance of the integrated circuit.

To provide higher functionality, the transistors need to communicate through interconnects, thus increasing the interconnect complexity. Interconnects are required to communicate the clock-, data- and control-signals, and distribute power to the various transistors on an integrated circuit. Unfortunately, the interconnect delays have not improved at the same rate as the transistor delays with technology scaling [18, 122]. The performance improvement gained by transistor scaling may be diminished by the negative effects of interconnect scaling.

Table 1 shows the International Technology Roadmap for Semiconductors (ITRS) projections on delays for three classes of interconnects, namely, local, intermediate, and global interconnects. The interconnect delays continuously increase with technology scaling. As we go from a 65 nm

Table 1: 2006 ITRS projections of interconnect delays for a local interconnect (0.1 mm), intermediate interconnect (1 mm), and global interconnect (10 mm)

Year of Production	2005	2006	2007	2008	2009	2010	2011	2012	2013
Technology (nm)	80	70	65	57	50	45	40	36	32
Delay (ps) local interconnect	44.0	61.2	76.7	104.4	138.8	179.2	239.2	285.7	345.1
Delay (ps) intermediate interconnect	355	527	682	1039	1413	1825	2436	2784	3504
Delay (ps) global interconnect	1110	1650	2090	3160	4100	5230	6870	7870	9770

technology to a 32 nm technology, the delays increase by more than $4\times$ for each of the local, intermediate, and global interconnects. Thus, the performance of the planar integrated circuit is increasingly limited not by the transistor delay, but rather by the interconnect delay [43, 119, 18, 122]. To ensure the highest performance possible out of the integrated circuit, we need to improve both the resistance and the capacitance of the interconnect.

1.2.2 Power Consumption

Power consumption has become one of the biggest priorities for today's integrated circuits. The total power consumption determines the maximum reliable operating frequency, power-supply sizes, and cooling requirements of the integrated circuit and hence plays a major role in determining its overall performance and reliability. Power consumption depends not only on the technology/circuit parameters (e.g., device sizes, circuit styles, oxide thicknesses) but also on the implementation (e.g., microarchitecture, frequency of operation) [38].

The total power consumption in a CMOS circuit consists of two parts: dynamic power consumption P_{dyn}, and static power consumption P_{stat}. The CMOS integrated circuit dissipates dynamic power when the circuit's transistors switch from logic-low to logic-high and vice versa [117]. The dynamic power depends on supply voltage, switching activity of transistors, load capacitance, and frequency of operation. During high-to-low and low-to-high transitions, dynamic power is consumed due to two current flows: (1) switching current flows to/from capacitive loads to charge (or discharge) the loads, (2) short-circuit current flows on the low-impedance path from power supply to ground. Dynamic power continues to increase due to the aggressive pursuing of performance with increasing frequencies and increasing number of transistors.

The CMOS integrated circuit dissipates static power due to leakage currents that flow even while the circuit is inactive. Static power has become a critical issue that gets worse with every new

technology generation [64, 123, 55]. Static power is dependent on supply voltage, threshold voltage, and temperature. The reducing supply voltage due to technology scaling requires a corresponding reduction in the threshold voltage. This decrease in the threshold voltage results in an exponential increase in the leakage current. In addition to voltage, static power is dependent on temperature and vice versa. Increasing static power increases the temperature of the integrated circuit, which in turn increases the static power. This creates a leakage-temperature feedback loop that might lead to thermal runaways and cause the circuit to fail in functionality.

1.2.3 Power Density

In addition to the total power, power density is a growing problem in which small, high-activity resources consume a large amount of power, causing hotspots on the processor [157, 13, 113]. Power dissipation can be unevenly distributed in modern microprocessors leading to hotspots with significantly greater temperatures than surrounding regions. Black et al. [13] report the hottest ($88\,^{\circ}\mathrm{C}$) and the coolest spot ($59\,^{\circ}\mathrm{C}$) on a high-performance planar processor with a temperature differential of $29\,^{\circ}\mathrm{C}$. High temperatures not only degree performance and reduce reliability but also increase static power and can lead to catastrophic failure of circuits. Power density continues to increase with technology generations as clock speeds, switching and leakage currents, and device counts push the limits of cooling mechanisms. Higher temperatures and increasing power densities have brought heat removal and power distribution to the forefront of the problems facing the semiconductor industry.

1.2.4 Manufacturing Process

Apart from the interconnect delay and the power concerns, maintaining the current rate of performance improvement faces increasingly difficult challenges from a manufacturing perspective. Current transistor sizes are already less than the wavelength of light used for photolithography. Advances in optics and shorter wavelength radiation may provide a few more doublings of transistor density [12]. Process variations are becoming increasingly non-deterministic in current and future technology generations. Process variations increase the variance of the circuit delays from their expected (mean) delays, thereby reducing the yields of the integrated circuits [150, 19, 68]. Lithography and etching processes cause variations in gate length (L). Chemical-mechanical-polishing

(CMP) processes causes variations in interconnect width and height [93] leading to variations in the interconnect resistance and capacitance. Random dopant fluctuations and poly line-edge-roughness cause threshold voltage variations [152].

1.3 Three-dimensional Integration Technology

3D-integration technology [68, 67] greatly reduces the impact of interconnect delays by placing the transistors in stacked layers and providing vertical connectivity. Two functional units connected by a long global wire in a planar circuit can instead be vertically stacked and connected in the third (vertical) dimension, thus drastically reducing the interconnect length. Wire-dominated functional blocks can be stacked on top of themselves to reduce the effects of intra-block wiring. 3D-integration technology provides new ways to design the various processor blocks and even the entire processor microarchitecture. Reducing the amount of interconnect also has a significant impact on power consumption as interconnect power is already estimated to consume about one half of a chip's power [91]. 3D-integration technology provides an alternative means of increasing integration density.

1.3.1 State of the Art in 3D-Integration

This section describes the state of the art in the 3D-integration technology, and outlines the critical parameters.

There are currently several proposed methods for vertically integrating multiple circuit layers such as multi-layer buried structures (MLBS) [158, 69] and 3D bonding technologies (e.g., wafer-to-wafer, die-to-wafer, and die-to-die bonding) [76, 104, 121]. Figure 1 shows a multi-layer buried structures (MLBS) 3D design. In the MLBS 3D technology, multiple device layers are successively (sequentially) fabricated in a stacked fashion. Layer-to-layer connections are made from either inter-layer interconnects (vias) or from direct source-drain/source-drain contacts. The advantage of the MLBS technology is that the 3D vias can potentially scale down with the transistor sizes due to the use of local poly-silicon wires for connection. However, the MLBS technology requires extensive changes to the existing manufacturing processes [157].

The 3D bonding technology requires fewer changes in the manufacturing process than the

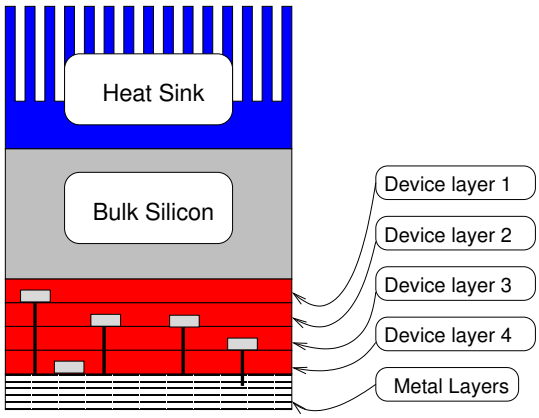

Figure 1: A MLBS 3D IC with four device layers

MLBS technology [95]. The 3D bonding technology processes the integrated circuits with conventional fabrication processes and provides vertical connectivity between the circuits on stacked layers using *bonding* techniques [104, 121, 65, 76]. The bonding technology could be oxide-to-oxide bonding [53], copper-to-copper bonding [13, 104, 121], and dielectric adhesive bonding [53]. In a 3D wafer-bonding technology proposed by Koyanagi et al. [76], a wafer is glued to a supporting material (handle wafer), thinned from the backside by mechanical grinding, and polished to a thickness of a few microns. The thinned wafer is aligned and bonded to another wafer and the handle wafer is removed from the thinned wafer. In another 3D wafer-bonding technology proposed by Lu et al. [65], fully processed wafers (with multilevel on-chip interconnects) are aligned and bonded with a dielectric glue, followed by top-wafer thinning and inter-wafer interconnection. The wafer thinning involves mechanical grinding, chemical mechanical polishing, and wet etching. The advantage of the wafer bonding process proposed by Lu et al. is that it does not require a supporting wafer (handle wafer). Another 3D wafer bonding technology processes the wafers with conventional planar fabrication processes and uses metal vias to bond the planar wafers vertically [13, 104, 121].

In wafer bonding technologies, the yield of the 3D integrated circuit is heavily influenced by the yields of each of the wafers [75]. Thus, wafer-level integration is desirable when the yield of each wafer is very high. 3D integration using die-to-wafer bonding or die-to-die bonding are other promising technologies in which known good dies [73, 11] can be stacked, thus increasing the overall yield. However, wafer-level stacking potentially provides a more cost-effective process compared with die-level stacking [103]. Our proposed designs assume a copper bonding based

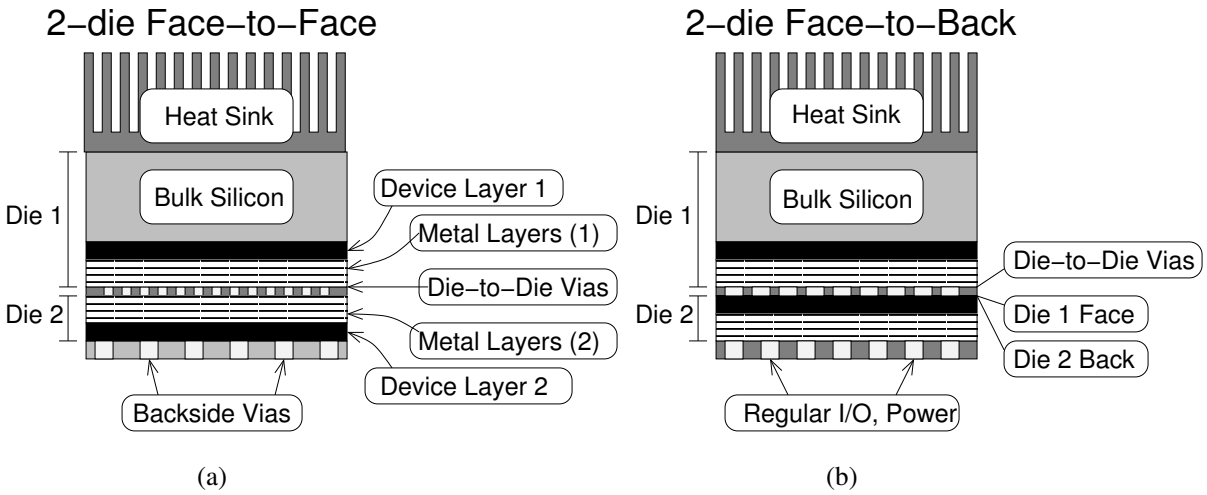

Figure 2: A 2-die-stacked 3D integrated-circuit with (a) face-to-face, and (b) face-to-back bonding topologies. (Figures not drawn to scale).

wafer-level stacking technology. We describe the details of the copper bonding 3D technology in Section 1.3.2.

1.3.2 Copper Bonding 3D-Integration Topologies

1.3.2.1 Two Layer Stacks

Figure 2(a) shows a 3D-integrated circuit using a face-to-face (F2F) wafer-stacking technology with copper metal bonding [13, 104, 121]. The bonding approach of Figure 2(a) involves depositing vias on the top metal layers of each of the two wafers, aligning the two wafers, and bonding them together. Under thermo-compression, the vias fuse together providing both the die-to-die interconnects as well as a physical mechanism to hold the die together. After bonding, one wafer is thinned with chemical-mechanical polishing (CMP) down to $\sim$10μm allowing low impedance backside vias to be etched through, which provide input/output and power/ground connections.

Figure 2(b) shows a 3D-integrated circuit using a face-to-back (F2B) bonding topology. The face-to-back bonding requires etching vias through the backside of the silicon (backside vias). The backside vias are challenging to manufacture for two reasons. First, etching through the backsides of the silicon will cause the cross-sectional area and the length of the backside vias to increase (relative to the face-to-face via). Second, the backside vias must pass through the active region of the silicon die, which may disrupt the layout of transistors. Hence, the face-to-back bonding

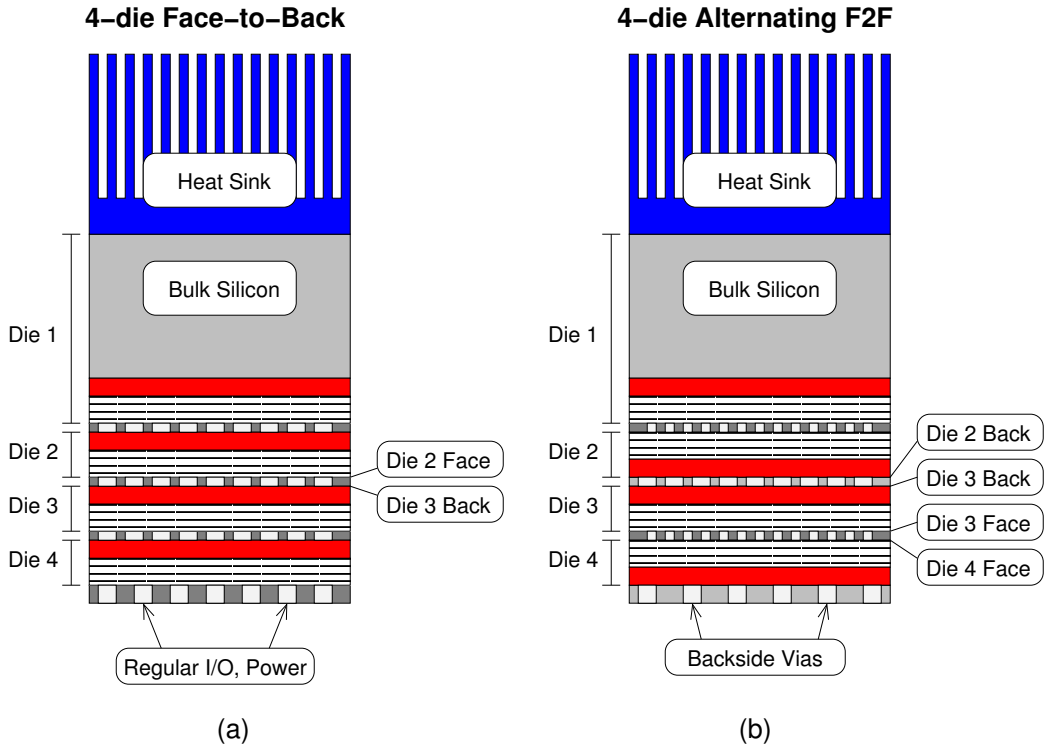

Figure 3: A 4-die-stacked 3D integrated circuit with (a) face-to-back , and (b) alternating face-to-face bonding topologies (Figures not to scale).

topology may not be able to provide as dense of a via interface as the F2F bonding topology.

1.3.2.2 Stacks with More than Two Layers

The 3D-integration technology will likely extend beyond stacking two layers to continue scaling performance and integration density. There are many organizations for integrating multiple layers in a 3D stack using both face-to-face (F2F) and backside vias. The bonding process may be repeated in combinations of face-to-face, face-to-back and back-to-back organizations.

Figure 3(a) shows a 4-die face-to-back (F2B) bonding topology. The advantage of the F2B bonding topology is the uniformity and repeatability of the fabrication process since each additional layer requires identical processing steps. Figure 3(b) shows a 4-die-stack that combines two F2F 2-die stacks with a back-to-back (B2B) interface between the pairs of die. After stacking two die in a face-to-face organization, coarser (less dense) die-to-die vias are required at the backside interface. The alternating F2F topology shown in Figure 3(b) may be desirable because the microarchitects and circuits designers can use the denser F2F vias at half of the die-to-die interfaces.

1.3.3 Die-to-Die Vias

The die-to-die (d2d) vias are perhaps one of the most critical 3D design parameters. The trade-offs between the different organizations have to do with the quality and pitch of the vias implementable at the different interfaces, and whether the vias interrupt the device layer. The pitch and latency of the d2d vias dictate the granularity at which a circuit can be partitioned across the different die.

The distance between the top metal layers on the adjacent die is very small [140, 121], and the size of the d2d vias is of the same order as the top level metal [39]. The thinning of the die reduces the distance that a d2d via must cross to connect the die. As mentioned earlier, the individual die are thinned to $\sim$10μm, and in a F2F organization the d2d vias only need to cross the distance separating the two top metal layers. Depending on the technology, the d2d via height may be $<$5μm to $\sim$20μm [36]. A d2d via is much smaller than the planar interconnect it replaces, and thus reduces both the resistance (R) and the capacitance (C). The signal propagation delay between the die is drastically reduced due to the reduced interconnect RC characteristics of the d2d vias. The delay to drive a signal through a d2d via from one die to another is less than one fan-out-of-four (FO4) delay [109], which makes the cost of cross-die communication similar to a short length of traditional metal. Note that the interface between the two d2d vias may present a disruption in the copper lattice structure, thus potentially increasing contact resistance.

Current implementations of the 3D-integrated designs support d2d via sizes from $\sim$3μm to 10μm [36]. The embedded industry has reported manufacturing d2d vias of size 2.4μm and expects second-generation d2d vias of size 1.46μm [51]. As alignment technologies continue to improve, via sizes smaller than 1μm may soon be practical. As a point of reference, a recent study on the 3D-integrated caches pointed out that a state of the art 6T SRAM cell takes up 0.7μm^2 (in a 65nm technology) [148] which provides a d2d via density on the order of two d2d vias for every three SRAM cells [109]. IBM has announced d2d vias with a size of 0.2μm [82]. The backside vias require etching through the bulk silicon substrate and as a result require larger structures in the current technology. The embedded processor industry currently manufactures backside vias of size 6μm, and has reported that the next technology generation will provide backside vias $<$4μm [51].

Apart from providing a dense d2d via interconnect, the F2F organization in Figure 2(a) does

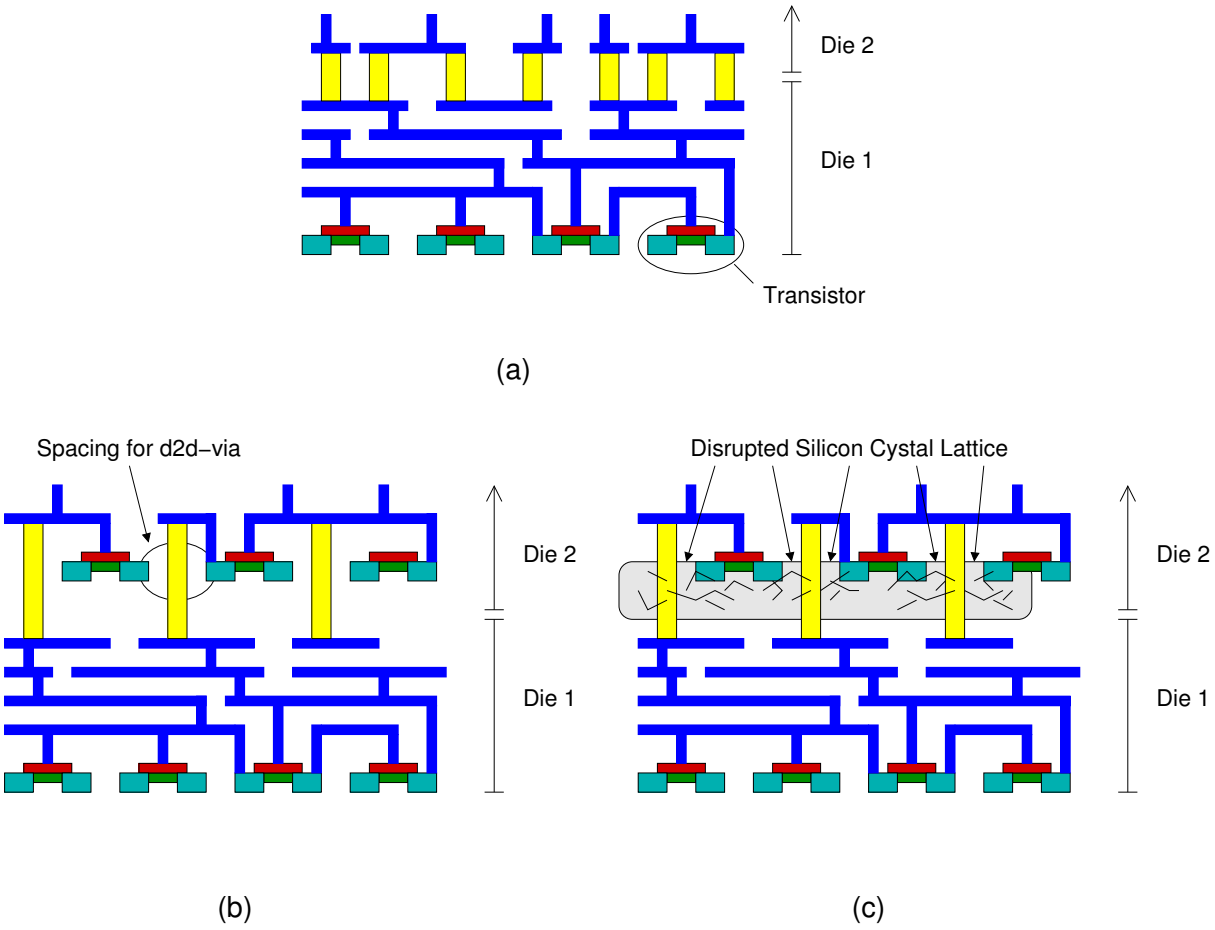

Figure 4: (a) Placement of F2F vias may not affect transistor placement. (b) Placement of backside vias interrupt transistor placement. (c) Backside vias may disrupt the crystal structure of the device layer degrading performance.

not require the vias to disrupt the active device layer. Figure 4(a) shows how the d2d vias can be built above the top-level metal so that they do not directly affect the floorplanning of the underlying logic. On the other hand, Figure 4(b) shows te bakside vias passing through the device layer which implies that space must be explicitly allocated for these vias. Etching the vias may also disrupt the local crystal structure of the silicon substrate as shown in Figure 4(c). This in turn may degrade the performance of transistors that are adjacent to the backside vias. To guard against this effect, one could consider the performance degradation of the transistors around these backside vias and limit the usage of such transistors to only non-critical path circuits, or define a "keep-out" region and avoid placing the transistors in that region. Note that Morrow et al. [103] have tested individual transistors in the thinned silicon of bonded wafers and have demonstrated both n- and p-channel transistors to preserve their electrical characteristics after bonding, thinning, and etching backside vias.

1.3.4 Benefits of 3D-Integration

The 3D-integration reduces interconnect delays because of the additional degree of routability in the vertical dimension. The 3D-integrated circuit designs increase the number of transistors that can be accessed in a single clock cycle. Figure 5(a) shows a planar circuit with a global wire connecting two blocks. Figure 5(b) shows a 3D implementation of the planar circuit where the planar global interconnect has been replaced by a short die-to-die (d2d) via. The additional degree of freedom in the vertical dimension enables a reduction in the interconnect requirements for connectivity.

By shortening the wire lengths, the 3D-integrated circuit can have a dramatic impact not only on improving the performance but also in reducing the power consumption. The power reduction comes from the reduced resistances and capacitances of the shorter wires as well as the reduced repeaters on the global wires. The drastic reduction in wire-length reduces the total power consumption of the 3D-integrated circuit as compared to the planar circuit.

The 3D-integration technology also enables a reduction of circuit footprints as shown in Figure 5. The 3D-integrated design has a reduced footprint compared to the planar design. The reduction in the footprint enables reduced form factor. Reduced form factor is an attractive option to the embedded processor industry, where the market trends continue toward miniaturized products for

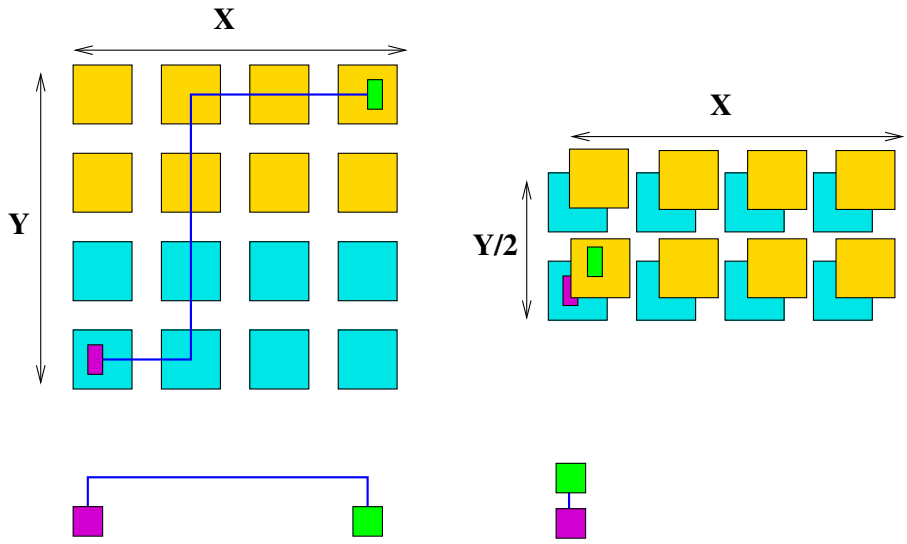

Figure 5: Benefits of 3D technology (a) Planar circuit (b) 3D-integrated circuit.

mobile users. This reduction in the footprint has implications in the yield of the integrated circuits. The smaller the footprints, the larger the number of die that can be obtained out of a certain sized wafer, and hence, the higher the yield. Figure 6(a) shows a wafer that yields two large planar die out of four (50% yield) due to manufacturing defects at two spots on the wafer. Figure 6(b) shows an identical wafer with the defects located at the same spots, which yields 23 small die out of 26 die (88% yield). Assuming each of the small die contains a quarter of the planar circuit, we need to stack four such die as shown in Figure 6(c) to obtain the complete 3D-integrated circuit. When the wafers are aligned and bonded, defective die on each wafer may align with good die on the other wafers causing a loss of overall yield. In the worst case, the 3D-integration yields fourteen 4-die-stacked 3D integrated circuits, while the planar fabrication provides eight (2×4) functional planar integrated circuits out of the same number (four) of wafers. Note that this is a qualitative argument that demonstrates that the yields of the 3D-integrated circuits need not be worse than the yields of the planar integrated circuits, and that the yield analysis of the 3D-integrated circuits requires further careful consideration and research. We consider the yield analysis to be outside the scope of this dissertation.

Another benefit of the 3D-integration technology is the ability to exploit best-of-breed technologies for each layer of the die-stack. Rather than optimizing a single process technology to fabricate both memory and logic elements on the same die, logic and memory processes can be optimized

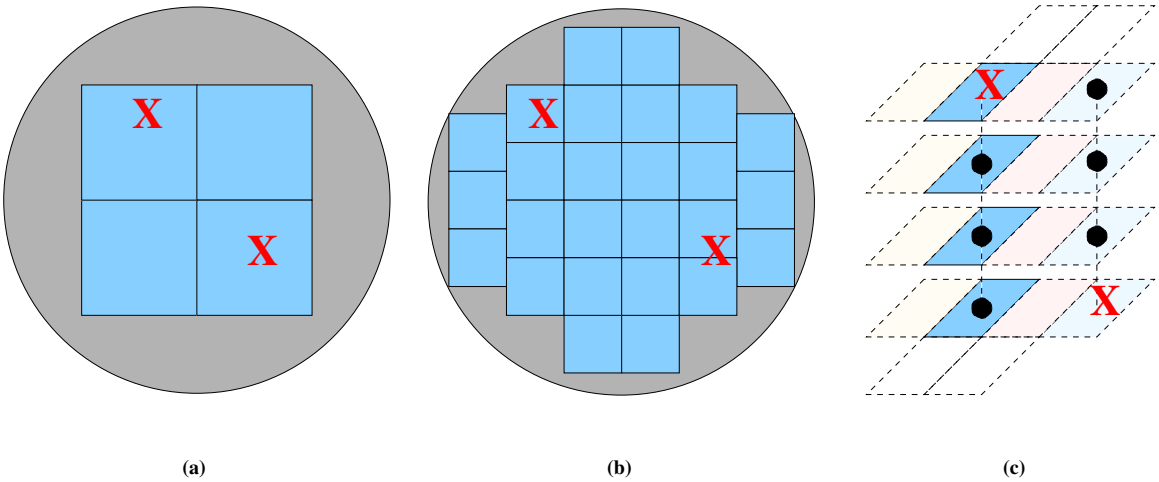

Figure 6: Wafer with (a) planar die (b) reduced footprint die (due to 3D-integration) (c) In wafer bonding, defective die may stack on good die, thus reducing yield.

individually and stacked using 3D-integration [88, 72, 128].

1.3.5 Some Challenges to 3D-Integration

1.3.5.1 Power Density

The 3D-integration technology increases the number of transistors in a volume, which may exacerbate the thermal profiles by increasing the thermal resistance [113]. Additionally, the 3D-integration reduces the footprint of the die, hence reducing the contact area between the heat sink and the die [115, 13]. The reduction in the contact area decreases the ability of the heat sink to remove heat from the die. Heat degrades performance, reduces reliability and increases static power. The increased power density may require more aggressive cooling mechanisms [108], adding manufacturing cost.

By using techniques such as wafer-thinning [76, 65] and thermal vias [46], we can reduce thermal resistances in the 3D-integrated circuits. Using state-of-the-art cooling solutions [108] such as direct liquid cooling can provide better cooling capability. The microarchitecture can also influence the thermal characteristics. In Part III (Chapter 6) of this dissertation, we describe some of the microarchitectural techniques we have proposed to address the thermal challenge in high-performance 3D-integrated microprocessors.

1.3.5.2 Power Integrity

Power integrity is already a reliability concern in the semiconductor industry [100, 99]. The 3D-integration may aggravate some of the power integrity issues. The smaller footprints of the 3D-integrated circuits reduce the area available for power distribution and result in lower pin counts for power supply and ground. In addition to the reduced pin counts, stacking may also cause multiple circuits to draw power from the same pins thus causing abrupt changes in current demands. In high-speed integrated circuits, such abrupt changes in current demands may lead to reliability issues due to inductive (Ldi/dt) noise in the power supply network [100, 99]. Also, some of the die may be further from the decoupling capacitors in 3D-integrated circuits, thus increasing the inductive noise [98, 156]. Hence, power integrity issues may pose a significant challenge to the performance and reliability of the 3D-integrated systems [97].

There has been some recent work in addressing the power integrity issues with respect to the 3D-integrated circuits. Researchers have proposed physical design [97] and microarchitecture design techniques [100] to mitigate the power integrity concerns related to the 3D-integrated circuits. More research is required to address the power integrity issues with special focus on the 3D-integrated circuits.

1.3.5.3 Design Automation Tools

There is a pressing need for electronic design automation (EDA) tools and methodologies to develop new architectures using the 3D-integration technology. Xie et al. [157] have identified two different categories of design tools essential for 3D microarchitecture designs: early design analysis tools and physical design tools.

There has been a great deal of on-going research in the academic community in the 3D EDA tools. Researchers have proposed different algorithms for early-design-phase automated microarchitectural floorplanning tools [56, 31, 58]. Given a microarchitectural description of a processor (blocks, sizes, interconnections), the floorplanning algorithm devises a 3D organization to optimize a given objective (e.g., performance, power, area, wirelength). Related 3D CAD/EDA work includes automated via placement in 3D-integrated circuits for heat dissipation [33], automated routing in 3D-integrated circuits [32], 3D power grid design, and 3D decoupling capacitor insertion

for controlling power supply (di/dt) noise in the 3D-integrated circuits [98].

1.3.5.4 Testing

Testing is another substantial challenge in the 3D-integrated circuit designs. Before bonding the wafers, the circuit on each wafer may exist in an incomplete state depending on the partitioning granularity [81], hence posing a challenge to the testing. Lewis et al. [81] have identified the testing challenges to the 3D-integrated designs and present a technique to enable pre-bond testability at a minimal area cost.

Some 3D-integrated design approaches provide an opportunity to test and debug parts prior to assembly [73, 11]. The ability to integrate proven parts shortens the design testing time. The use of redundancy techniques, dynamic repair, and real-time fault detection/isolation may enable low-cost, high-volume 3D-integrated circuits.

1.3.5.5 Manufacturing Cost

3D integration may increase the cost of manufacturing because of the additional processing steps. Multiple wafers have to be manufactured for a given integrated circuit thus requiring multiple masks. After fabrication, the wafers need to be polished and planarized, have their d2d interconnects deposited, thinned and then bonded. Apart from the planar fabrication processes, new technologies such as precision alignment systems and bonding apparatus are needed to successfully build 3D-integrated circuits. The 3D-integration may also affect yields because an n-die stack can be rendered inoperational by a single bad die, even if the other $n-1$ are fine (refer to Section 1.3.4).

A full analysis of the economic viability of the 3D-integration is beyond the scope of this dissertation. An open research question is to determine and evaluate the benefits that can be obtained from this new technology. Note that, some of the additional costs of the 3D-integration may be partially recovered. It is possible to share global routing and/or clock layers between multiple die (e.g., run the global portion of the clock tree on only one die, and use local clock trees on each die to complete the clock distribution), thus reducing the number of metal layers required for clock distribution. Each individual die in a 3D-stacked processor has a smaller footprint than in a planar/2D implementation, which in turn may enable more effective packing of the rectangular die on a circular wafer as shown in Figure 6(b).

1.4 Scope of This Dissertation

3D-integration technology has the potential to address many of the challenges facing the semiconductor industry. The increased device density and the ability to place and route in the vertical dimension provide new opportunities for microarchitecture design.

The thesis of this dissertation is that the 3D-integration provides simultaneous performance and power benefits to build high-performance microprocessors, while keeping the worst-case temperature under control.

We support this thesis by studying the microarchitectural impact of the 3D-integration technology on the design of high-performance processors and quantifying the benefits. Note that the results presented in this dissertation represent a *conservative estimate* of the value of the 3D-integration technology. We start with conventional planar microarchitectures and adapt to the 3D-integration technology, whereas a microarchitecture explicitly designed to target the 3D-integration may likely provide more benefits.

Figure 7 shows one possible path from current planar designs to future 3D-integrated microarchitectures such as those proposed in this work. Figure 7(a) shows a planar/2D processor configuration that has two CPU cores and a L2 cache. The early incarnations of 3D-integrated processors may primarily leverage the technology for device density. A likely design (b) would 3D-integrate multiple cores on top of a large on-chip cache [13]. This design can provide performance benefit from reducing the wire delay between the cores and the cache, but may not realize the full benefits of the 3D-integration (i.e., the clock speed, area, and power of the individual CPU cores are identical to those of the planar implementation). As the technology matures, more sophisticated 3D-integrated processors will evolve by first adding more layers of cache (Figure 7(c)) thus increasing the overall size of the cache, and then implementing the circuit blocks on multiple die leading to the implementation of 3D-integrated processor cores (Figure 7(d)). Li et al. [82] explored the use of the 3D-integration to implement a multi-core system with a 3D-distributed cache organization. Liu et al. [88] evaluated the impact of integrating a system's main memory directly on top of the processor. In both of these studies, the 3D-integration technology is employed at the coarser interfaces (cpu-to-cache/cpu-to-DRAM) and so the CPU core remains as a planar/2D structure. We

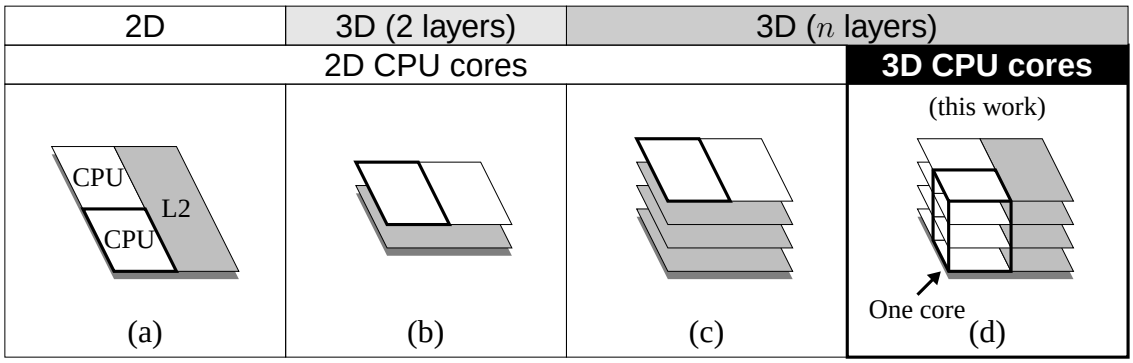

Figure 7: A progression from planar/2D processors to 3D-integrated processors with varying numbers of layers and CPU implementation styles.

categorize microarchitectures such as the 3D CMP [82] and the IntroSpective 3D processor [105] as belonging to Figure 7(b) and (c) since each core is still a conventional planar structure. The scope of our research is 3D-integrated processors (Figure 7(d)). In this dissertation, we focus on the 3D-integrated designs of high-performance processors using 2-die- and 4-die-stacks. We assume a copper-bonding based wafer-level stacking 3D technology. We assume a face-to-face topology for 2-die-stacks and an alternating face-to-face topology for 4-die-stacks without loss of generality. We limit the scope of our exploration to homogeneous die-stacks of MOS (static and dynamic) circuits. We consider the integration of heterogeneous technologies (e.g., DRAM [88, 72], analog [128]) to be orthogonal to our techniques and outside the scope of this work.

We have organized the rest of this dissertation into two parts. In Part II, we present our designs of the 3D-integrated processor components and evaluate the performance and power benefits of these 3D-integrated designs. Chapter 2 presents the designs of SRAM based components. Chapter 3 presents the designs of CAM based components. Chapter 4 presents the designs of data processing components. Together, these three classes of circuits account for a large portion of circuits in the processor. Each of the individual chapters demonstrates that the 3D-integrated circuits provide simultaneous performance and power benefits in comparison to their planar counterparts.

After describing the 3D-integrated designs of processor components, we put them all together in Part III, where we propose and evaluate the 3D-integrated versions of existing high-performance processors. Although existing planar microarchitectures may not derive the full benefit of the 3D-integration, we chose to use existing microarchitectures as our baseline so as to identify the benefits

that are solely due to the 3D-integration. Chapter 5 presents the 2-die-stacked and the 4-die-stacked designs of 3D-integrated high-performance processors based on the planar Alpha 21364 processor. Our 3D-integrated processors demonstrate the simultaneous benefits of the 3D-integration and highlight the power density and thermal issues related to the 3D-integration technology. Chapter 6 presents our 3D-integrated high-performance processor based on the Intel Core microarchitecture. Our 3D-integrated processor addresses the thermal challenges in high-performance using novel microarchitectural techniques. Our results indicate that the temperature increase experienced by the 3D-integrated processors can be effectively controlled using microarchitectural techniques.

Overall, we show that it is possible to keep the temperature under control in the 3D-integrated processors through a combination of reducing total processor power, local power density, and effective thermal resistance while simultaneously increasing performance by a significant amount.

Part II

"Creating a new theory is not like destroying an old barn and erecting a skyscraper in its place. It is rather like climbing a mountain, gaining new and wider views, discovering unexpected connections between our starting point and its rich environment. But the point from which we started out still exists and can be seen, although it appears smaller and forms a tiny part of our broad view gained by the mastery of the obstacles on our adventurous way up."

- Albert Einstein

3D-Integrated High-Performance Components

Thesis statement: 3D-integration provides simultaneous performance and power benefits to build high-performance microprocessors, while keeping the worst-case temperature under control.

To substantiate our thesis statement, we start by studying several individual components that together comprise a large portion of the planar high-performance processors. We focus our attention on three broad classes of circuits, namely, the static random access memory (SRAM) components, the associative logic (CAM) components, and the data processing components. Based on our analysis of the planar components, we propose 2-die-stacked and 4-die-stacked 3D-integrated designs of the processor components. Using representative examples from each class of circuits, we demonstrate different 3D-integrated circuits to address the challenges facing the conventional technology.

Our 3D-integrated designs are partitioned at the circuit level to take advantage of the dense die-to-die (d2d) via interface, thereby increasing the potential benefits of the 3D-integration. We illustrate the additional benefits provided by the circuit-level 3D-integration using a motivating example of a planar processor with a large on-chip cache. Figure 8(a) shows a conventional (planar/2D) processor core with a large on-chip last-level (L3) cache. An intuitive 3D-integration is to the stack the large L3 cache on top of the processor core as shown in Figure 8(b). The benefit of such an organization is an approximate footprint reduction of 50%. But this design fails to utilize the dense die-to-die via interface and the full flexibility of vertical routing to further reduce interconnects and shorten the critical paths within the circuits. Figure 8(a) shows the critical path for accessing the planar cache. In the cache-on-processor implementation in Figure 8(b), the circuit structure of the cache is identical to the planar case and hence the critical path within the cache has not reduced. As a result, the latency and power have not significantly improved by this organization. Furthermore, stacking the cache on top of the processor may prevent other components within the processor from taking advantage of the 3D-integration. Figure 8(c) shows a 3D organization using circuit-stacked 3D components. With circuit-stacked components, the critical path has reduces significantly. The

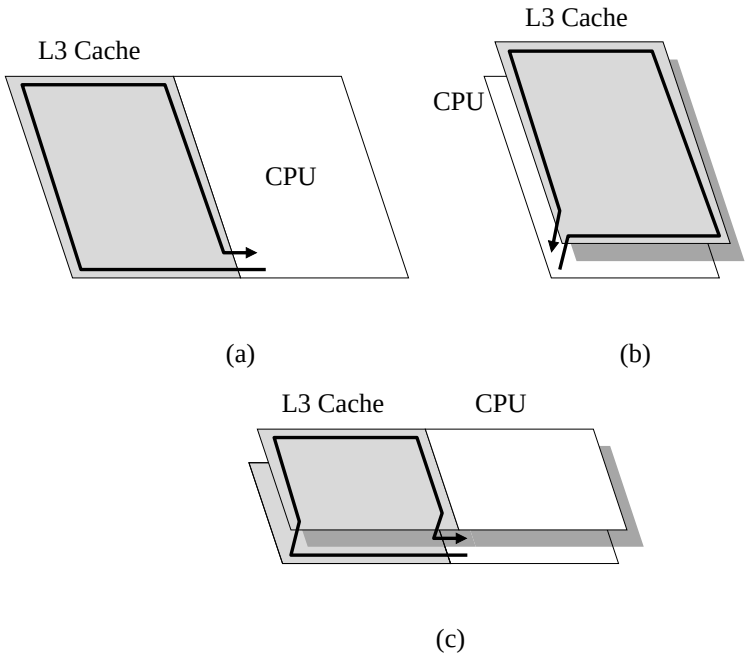

Figure 8: (a) A planar layout of a processor and L3 cache with the L3 critical path, and a 3D implementation with (b) the cache stacked on top of the processor, and (c) circuit-stacked processor and cache.

reduction in the critical paths reduce not only the latencies of the circuits but also their power consumption, thus providing simultaneous benefits.

In the next three chapters, we demonstrate the simultaneous benefits of the 3D-integrated SRAM-, CAM-, and data-processing-components of conventional processors. We use HSpice to obtain the critical path latencies and the energy consumptions of the various processor components. The critical path latency represents the worst-delay path through the circuit, whereas the energy includes the energy consumption from all active circuits and wires. Our HSpice simulations use 65nm predictive technology models [26]. Based on 130nm wire parameters from Intel [147], we extrapolated the wire parameters for 65nm. Our circuit implementations primarily make use of static CMOS gates with some exceptions such as the comparator logic in the associative logic (CAM) based components. To optimize our designs, we sweep through a range of transistor sizes and use the transistor sizes that minimize the overall delay. We use a distributed RC-ladder model for all wires in the circuits. While our absolute results are dependent on the technology parameters and assumptions, the general trends and benefits of the 3D-integrated circuits are likely to hold for other technology parameters as well.

Our 3D-integrated circuit designs use d2d via sizes of 1μm for the face-to-face (F2F) and 2μm for the backside interfaces. While the size of the current manufacturable d2d vias is already only 2.4μm and 4μm for F2F and backside interfaces respectively [51], we assume that the d2d sizes will continue to scale at least for a few more generations. We target a copper-bonding wafer-level 3D-integration technology (refer to Section 1.3.2), where the d2d vias are made with the same material (copper metal) as the traditional on-die interconnects, and therefore have similar per-unit resistance and capacitance. We model the die-to-die vias to be 9μm to cross between the F2F interface and 12μm to cross the backside interface, based on published data [13, 140, 121, 39]. Note that the current 3D-integration technologies already thin the die down to 12μm [146].

Due to higher frequencies, reduced footprints, and/or higher integration, the 3D-integrated circuits may cause the power density to increase substantially, leading to hotspots and causing thermal reliability issues [136]. Researchers have suggested using "dummy" thermal vias that do not carry signals, but provide low thermal resistance paths to remove heat from potential hotspots [28]. A naive thermal analysis may lead one to believe that the 2-die-(4-die-) stack might double (quadruple) the power density. However, such an analysis assumes that the total power consumption of the 3D-integrated circuit is identical to the planar integrated circuit. Our 3D-integrated components substantially shorten the wire lengths and hence consume less power, thus reducing the thermal impact. In this part (Part II) of the dissertation, we demonstrate that the total power/energy consumption of our 3D-integrated component is substantially lower than that of the corresponding planar component. While the power density may increase, the total power of the 3D-integrated circuit decreases. The total power has been shown to have a great impact on the temperature [83]. In some 3D-integrated designs, the latency benefit may be traded using circuit techniques for further power reduction (refer Chapter 4). As another example, some 3D implementations of planar dynamic logic circuits may provide enough latency benefit to allow the power-hungry dynamic logic circuits to be replaced with lower-power-consuming static logic circuits. A thermal analysis of the individual 3D-integrated circuits in isolation is ineffective since the temperature depends not only on the power consumption and the layout of the 3D-integrated circuit but also on the adjacent components and their power consumptions.

In a later part of this dissertation (refer to Part III), we analyze the hotspots on the 3D-integrated

processors [113, 115] and propose microarchitectural techniques [115] to control the worst-case temperature on the 3D-integrated processors.

CHAPTER II

SRAM COMPONENTS

2.1 Overview of This Chapter

This chapter focuses on the planar and the 3D-integrated designs of the circuits that use static random access memory (SRAM) arrays. In order to prove our thesis, we start by proving the assertions made in the thesis statement to hold true for each component of the processor. This chapter presents our designs and results for the processor components based on the SRAM arrays.

We show that the dense die-to-die vias enable the 3D-integrated SRAM components that are partitioned at the level of individual wordlines, bitlines, or ports. The 3D circuit partitioning results in a wire length reduction within the SRAM array, and a reduction in the footprint, which reduces the wires required for global routing. The wire length reduction provides simultaneous latency and energy benefits. As planar designs adapt high-performance techniques such as hierarchical wordlines to improve performance, the corresponding 3D-integrated circuits provide even higher benefits, making it a desirable technology for high-performance designs.

The rest of the chapter is organized as follows: Section 2.2 describes the planar SRAM components. Section 2.3 describes the 3D-integrated designs of large, banked SRAM arrays. Section 2.4 describes the 3D-integrated designs of multi-ported SRAM arrays. Section 2.5 presents our results and analysis. Section 2.6 presents some conclusions from this chapter.

2.2 Planar SRAM Components

Conventional (planar) static random access memory (SRAM) arrays are regular-structured and wire-dominated circuits that show great promise for 3D implementations [109, 120, 148]. Many components on the processor are variations of conventional SRAM arrays. These modules include caches, register files, branch predictor history tables, branch target buffers, physical and architected register files, reorder buffers, and payload memory portions of the instruction issue logic. These components differ in the capacity (number of entries and bits per entry), timing constraints, and bandwidth requirements for reading and writing the arrays.

On-chip cache memories are typically large SRAM components that occupy a vast portion of the processor die area. Caches reduce off-chip memory accesses by exploiting the concepts of temporal and spatial locality. Caches have a large capacity and require both tag and data arrays. Due to their large capacities, caches tend to be organized as banks in order to increase bandwidth while consuming less power. Caches are often subbanked to save power by sharing the sense amplifier circuitry among the subbanks. The basic cache design parameters are cache size, block size, and associativity.

Register files are structurally similar to on-chip caches in that both consist of regular arrays of 6T SRAM cells. Register files have lower capacity requirements and do not have a tag array. But they are typically multi-ported (multiple read ports and multiple write ports) in order to satisfy the bandwidth requirements of the data processing components. The register files are critical components of modern processors in terms of impact on clock frequency and instructions per second (IPC) rates.

The planar SRAM array based components have a regular array of memory cells that are easy to partition across multiple die. The main SRAM array can be viewed as a set of wordlines running horizontally and a set of bitlines running vertically. A row decoder drives the wordlines, which control the access transistors of the data storage cells. The bitlines are read by sense amplifiers at the bottom of the array. Figure 9 shows the critical path of a SRAM array read operation. In case of tagged SRAM components such as caches, a similar array stores the tags and a set of comparators perform the tag match operation.

We describe four existing techniques to enhance performance and save power in planar SRAM arrays. Three of these techniques, namely banking, subbanking, and hierarchical wordlines (HWL) [4], minimize the latency and power of accessing large SRAM arrays. A fourth technique called multi-porting increases the bandwidth of SRAM arrays.

2.2.1 Memory Banking

Memory banking technique divides the memory array into multiple sub-modules called banks. By dividing the array into multiple banks and accessing only the bank that contains the required data, significant power saving can be realized. Memory banks can also be used to enhance bandwidth,

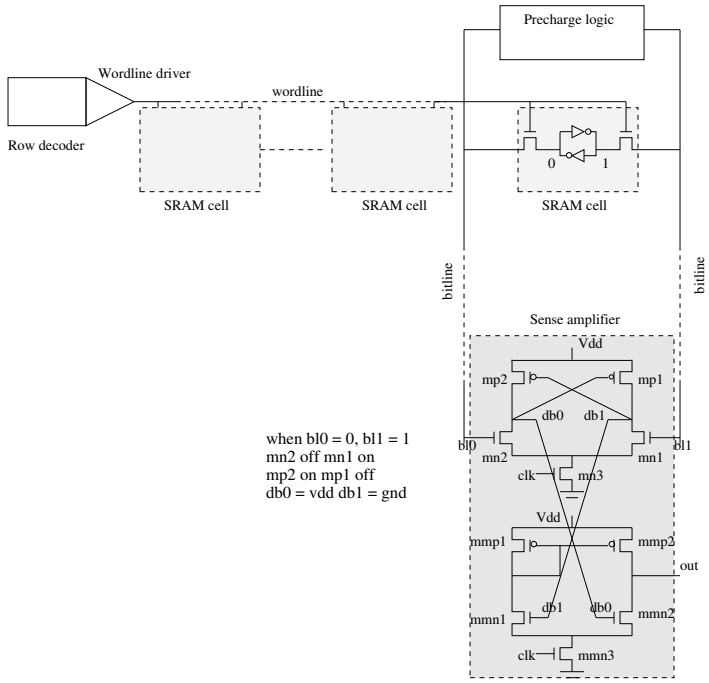

Figure 9: Critical path of a cache access (read) operation

while not requiring a significant wire area increase as in the case of multiple ports. Figure 10(a) shows a single memory array with addresses ranging from 0 to $2^N - 1$. Figure 10(b) shows the same array divided into two banks using higher-order address interleaving technique. Higher-order address interleaving divides the memory array into banks based on the higher order address bits. If the array contains 2^N locations (0 to $2^N - 1$), then one bank contains addresses from 0 to ($2^{N-1} - 1$), and the other bank contains addresses from 2^{N-1} to $2^N - 1$. Figure 10(c) shows the array divided into four banks using higher-order interleaving. Figure 10(d) (Figure 10(e)) shows the memory array divided into two (four) banks based on lower-order address interleaving. Lower order address interleaving uses the lower order address bits to identify the banks. Hence, this technique divides the odd and even addresses into separate banks. If the requested data resides in only one of the multiple banks, the other banks need not be accessed and do not consume dynamic power.

Banking can also be used to enhance bandwidth. If the requested data values reside in different banks, the processor can simultaneously obtain values out of multiple banks, thus increasing the input/output bandwidth. Note that it is possible for two or more simultaneously requested data to reside in the same bank. When multiple addresses target the same bank, the cache detects a bank conflict. In case of bank conflicts, a buffer mechanism can store and reissue the requests to the target

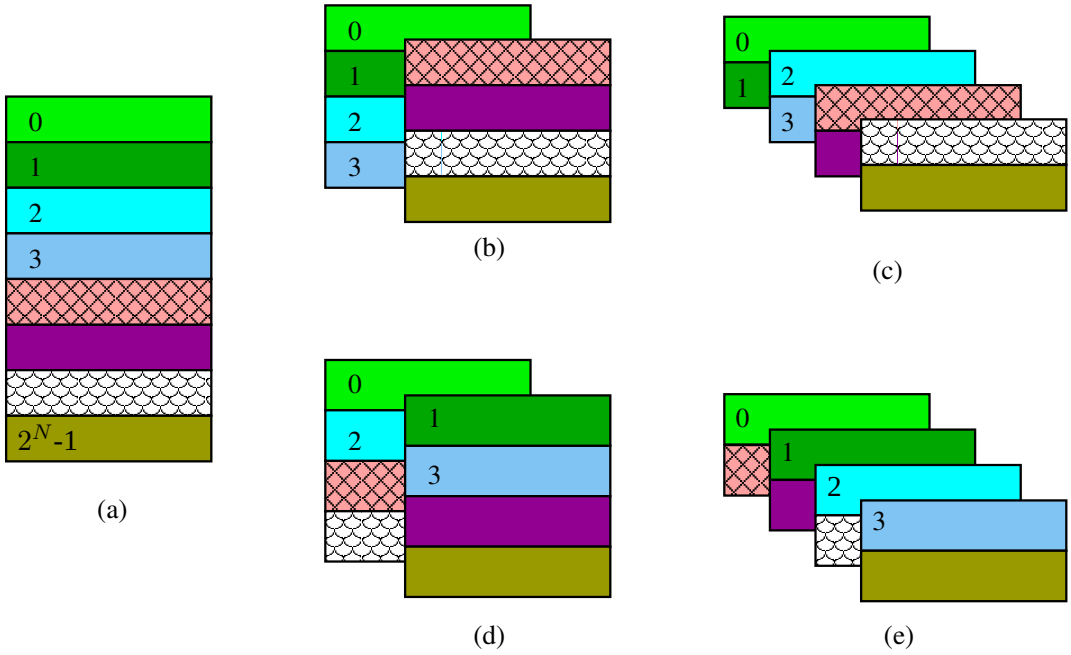

Figure 10: Memory banking technique (a) one monolithic array (b) two banks with higher-order interleaving (c) four banks with higher order interleaving (d) two banks with lower order interleaving (e) four banks with lower order interleaving.

bank.

2.2.2 Memory Subbanking

Memory subbanking saves power in the bitline and the sense-amplifier network [44]. In subbanked cache memory arrays, cache blocks are divided into subbanks. A portion of the cache address (block offset portion) is used to select the subbank that contains the required data. Since data is read out from only one subbank at a time, a common set of sense amplifiers can be shared across the subbanks, thus reducing the cache power. Figure 11(a) shows the data array of a cache with each cache block consisting of four data words. In the absence of subbanking, all four data words are read out and sensed using sense-amplifiers. The required word is then chosen using the block offset bits in the address. Figure 11(b) shows the cache blocks divided into two subbanks, with each subbank consisting of two data words. In the case of subbanking, the subbank selector logic selects between the two subbanks and feeds the data from only one subbank into the sense-amplifier circuitry. Hence, the subbanks can share the sense-amplifier circuitry and save significant amount of power. Subbanking also helps in saving the bitline pre-charge power, since only the selected

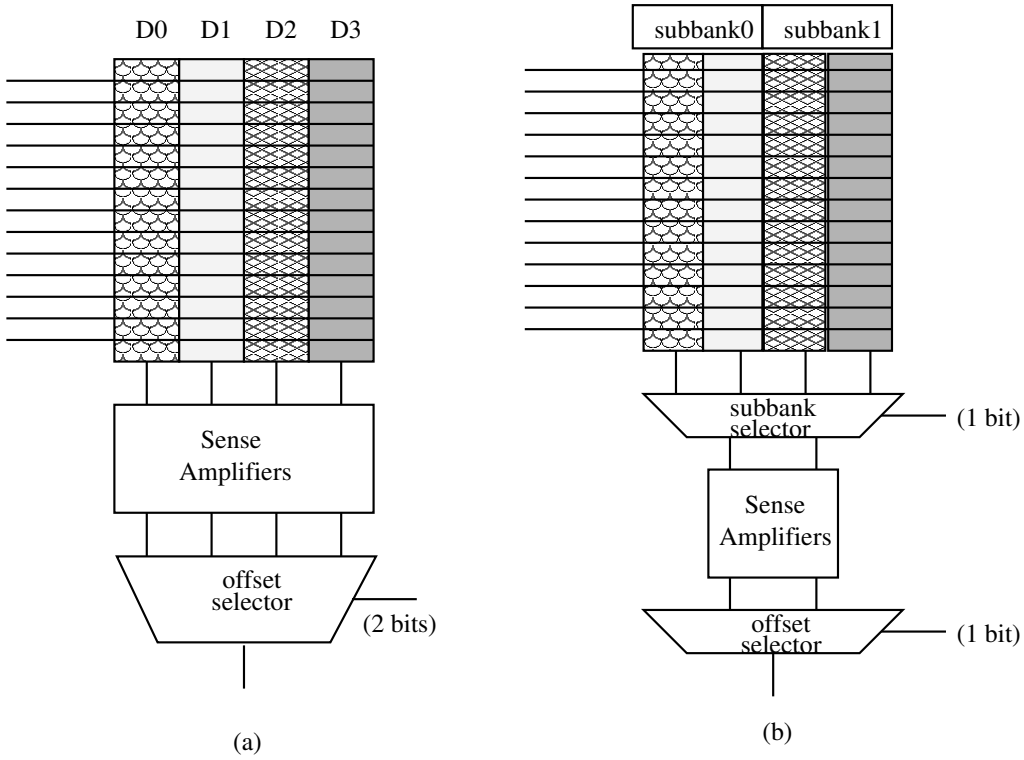

Figure 11: Memory subbanking technique: (a) original data array with each cache block consisting of four data words (b) subbanked data array with two subbanks (two data words per subbank).

subbank needs to be pre-charged.

2.2.3 Hierarchical Wordlines

Wordlines in the SRAM arrays are heavily loaded by the access transistors, which are two per SRAM cell as shown in Figure 9. Hence, the wordlines contribute significantly to the overall latency of the SRAM circuits. Hierarchical wordline technique [4] improves performance by reducing the latency of driving the wordline. The hierarchical word line (HWL) structure uses a global wordline (GWL) to drive multiple shorter sub-word (local) lines. The row decoder output is used as the global word line. Each global word line drives several groups of sub-word lines distributed across the SRAM array. With the HWL structure, the wordline loading is reduced, thus enhancing the overall performance. As an unintended side effect, the HWL technique also worsens the wire complexity of the wordlines since the wiring requirement of wordlines is essentially doubled.

2.2.4 Multi-Porting Technique

Modern processors capable of issuing many instructions per cycle require a large number of read and write ports in components such as register files. The size of an SRAM cell in such multi-ported SRAM components increases dramatically with increasing port requirements. Figure 12 shows SRAM cells with (a) one port, (b) two ports, and (c) four ports. While Figure 12 illustrates a simplistic SRAM cell design (e.g., read-port isolation transistors have been omitted for clarity), it demonstrates the quadratic increase in area with respect to the port count. The area explosion forces the wordlines and bitlines to increase in length that in turn increases both access latency and power consumption. Many microarchitecture-level proposals have been made to deal with the size, latency and power of the register files, including register caching [9] and register file banking [151]. While these techniques can reduce the average latency of register file access, they significantly complicate the processor's data- and control-paths. Increases in processor clock frequency and the relative decrease of the speed of wires exacerbate the problem. Another example of the poor scaling of the register file latency can be found in the Alpha 21264's 4-issue integer execution unit which would normally require an 8-read port, 4-write port register file. Instead, the architects chose to duplicate the entire contents of the register file such that each copy only needs half as many read ports [71]. Two full copies of a moderately ported register file proved to be smaller and faster than a single highly-ported structure. Since the register file is dominated by wire, the 3D-integration may provide an effective means for controlling the latency and power of the large register files required by the modern high-performance processors.

Now, we propose our 3D-integrated SRAM components. For illustration purposes of the planar baseline and our 3D-integrated SRAM circuits, we consider caches and register files as respective examples of banked SRAM and multi-ported SRAM designs.

2.3 3D-Integrated Large SRAM Components

Long metal wires used to route global signals in banked SRAM arrays such as caches suffer from large wire delays and power consumption. The address bus that routes from the edge of the SRAM array to a bank is an example of such wires. Figure 13(a) shows an eight-bank SRAM array with the critical wire path from the bottom left corner of the cache to the farthest bank. One option for a

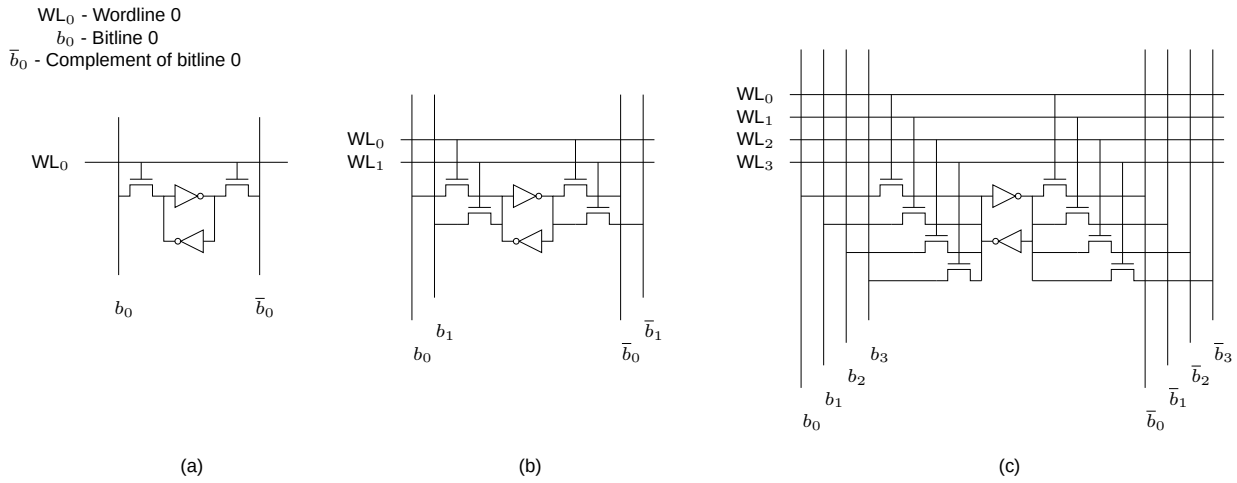

Figure 12: Quadratic area increase of an SRAM cell as the number of ports increases from (a) one to (b) two to (c) four ports.

3D-integrated SRAM array design is to stack banks on top of each other (bank-stacked 3D circuits). Another option is to split the arrays on multiple die (array-split 3D circuits).

2.3.1 Bank-Stacked 3D SRAM Circuits

Figure 13(b) shows a 3D bank-stacked organization with a 2-die-stack. A 3D bank-stacked design was first proposed by Reed et al. [120]. There are two possible orientations for bank stacking. The banks can be stacked left-to-right as shown in Figure 13(b), or top-to-bottom. Figure 13(b) shows how the bank stacking results in a 67% reduction ($3x \rightarrow x$, where x is the bank width) in the horizontal component of the wiring to and from the banks. Because the stacking of the banks occurs in only one direction, the vertical component of the bank wiring is unaffected, thus reducing the wire length savings to 50% (assuming $x = y$, where y is the bank height). Overall, the reduction in wire length translates into a reduction of latency, power, and footprint.

When we have more than two die on the 3D-integrated circuit, we can combine the bank splitting strategies to increase the benefits of the 3D-integration. The fastest bank-stacked 3D SRAM organizations may use different bank partitioning across the stacked layers. The application of a particular bank-stacking technique may reduce the latency of a critical wire by a significant amount such that it may no longer be the worst delay path in the circuit. A different style of partitioning can then address the new worst delay. With a 4-die-stack, we can stack the banks top-to-bottom across

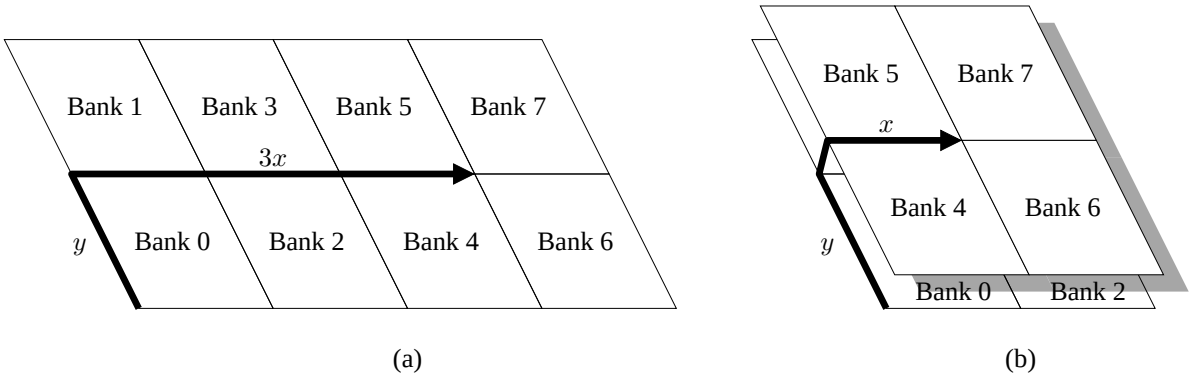

Figure 13: (a) A planar 8-banked array showing the worst-case distance to the farthest bank, and (b) a 3D bank-stacked SRAM organization.

the F2F interfaces and left-to-right across the B2B interface. Figure 14(a) shows a 16-banked planar SRAM array design, that is stacked left-to-right on a 2-die-stack as shown in Figure 14(b). With a 4-die-stack option, the next 3D bank-stacking happens top-to-bottom, thus producing a 4-die footprint that is more evenly balanced in both X- and Y- dimensions.

2.3.2 Array-Split 3D SRAM Circuits

We evaluate splitting the individual rows and columns of the SRAM arrays within a bank on multiple die. Array splitting can reduce the lengths of the wordlines and/or the bitlines depending on the number of die and the orientation of the split. Figure 15(a) shows the details of a small SRAM data array. With bank-stacking, the decision for a top-to-bottom or left-to-right stacking largely depends on which dimension provides larger wire reduction. At the array level, the orientation of the split has a greater impact on the circuit implementation of the array.

The first array-split configuration stacks columns on columns. Figure 15(b) illustrates the column-stacked 3D SRAM array. In this design, the single long wordline has been replaced by a pair of parallel wordlines. The row decoder must drive the wordlines on both the die. This requires one d2d via per wordline. At the bottom of the array, the column select multiplexors have been split across the two die thus requiring additional die-to-die vias. This organization reduces latency and power due to reduced wordline lengths. As shown in Figure 15(c), these benefits are in addition to reductions in the bank-level wire lengths.

The second array-split configuration stacks rows on rows. Figure 15(d) shows a row-stacked

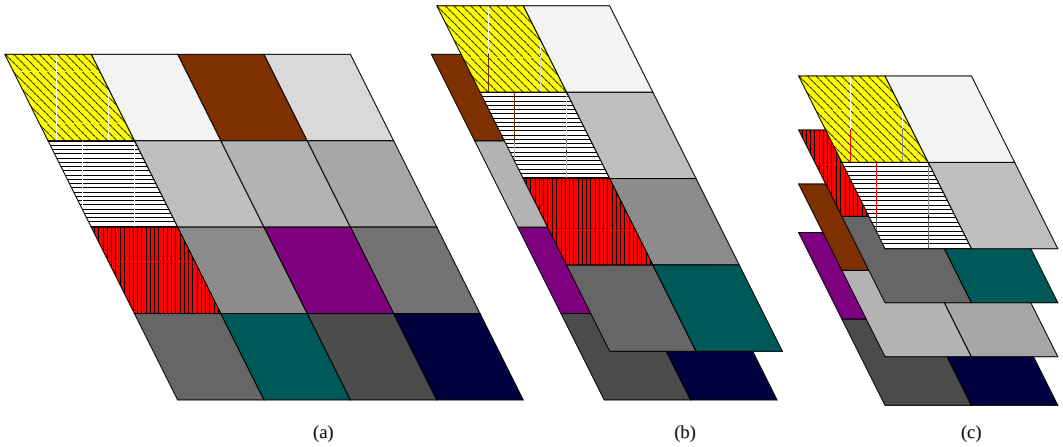

Figure 14: (a) A banked planar array with 16 banks, (b) a 2-die-stacked 3D array with left-to-right stacking, (c) a 4-die-stacked 3D array with top-to-bottom stacking across the F2F interfaces and left-to-right stacking across the B2B interface

3D SRAM array. The row-stacked array requires the row decoder to be partitioned across the two die. We first decompose the 1-to-n decoder into a 1-to-2 decoder followed by two 1-to-$\frac{n}{2}$ decoders. Even though the two 1-to-$\frac{n}{2}$ decoders are stacked on top of each other, the 1-to-2 decoder will only activate one of them which avoids stacking of thermally active components. The lengths and loading of the wordlines remain the same as in the planar organization, but the lengths of the bitlines reduce by half. Similar to the column-stacked organization (Figure 15(b)), there are latency and power benefits due to wire reduction at both the array- and bank-levels.

2.4 3D-Integrated Multi-Ported SRAM Components

There are many possible designs for multi-ported SRAM arrays in the 3D-integration technology. We propose four different strategies for partitioning the multi-ported SRAM arrays across multiple die.

2.4.1 Register-Partitioning (RP) 3D SRAM Circuits

A register-partitioned (RP) 3D SRAM array with a 2-die-stack splits half of the entries and places them on the vertically stacked die. Figure 16 illustrates a 32-entry register file implemented on a 2-die-stack where the bottom die contains registers R0 through R15, and the top die contains R16 through R31. A result of this topology is that the vertical distance (along the bitlines) has been halved, which can greatly reduce the latency and power associated with toggling the bitlines. The

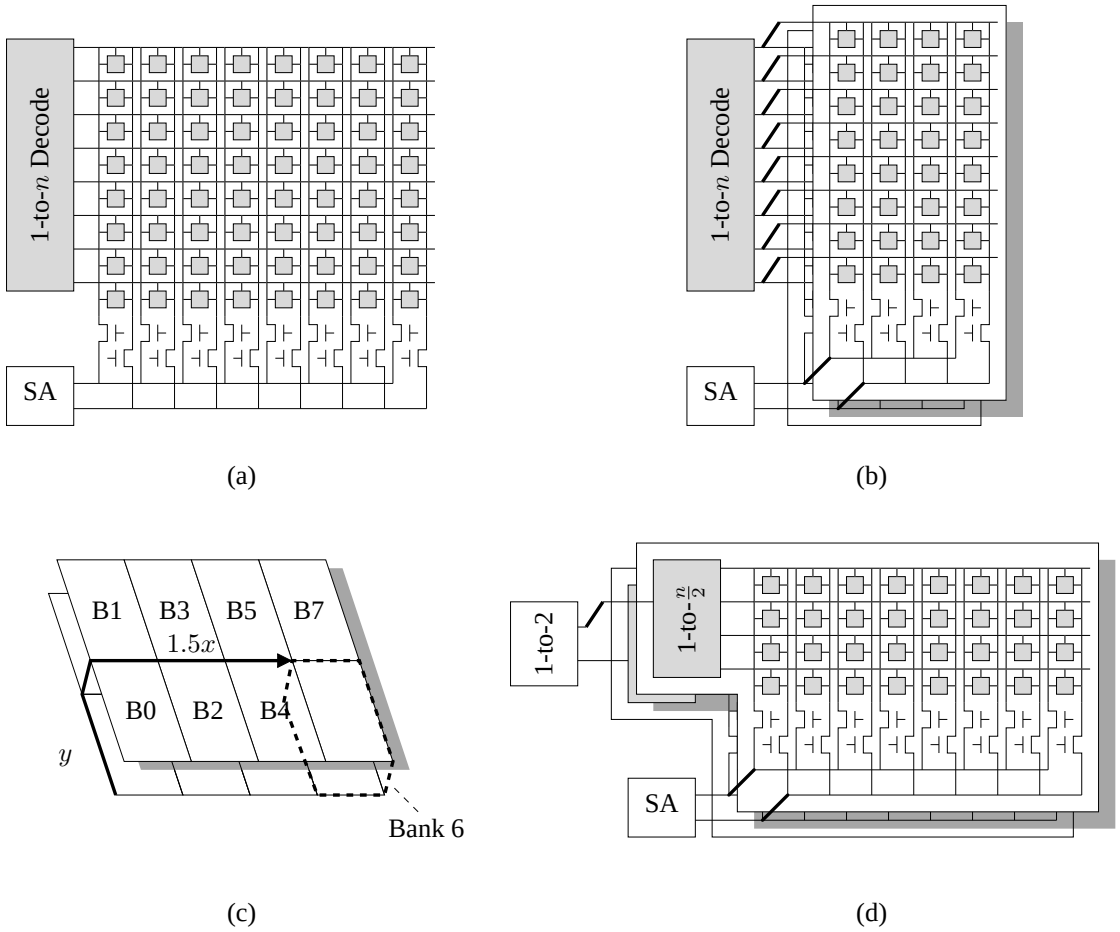

Figure 15: (a) A planar SRAM array, (b) a 3D-integrated array with column-on-column array-splitting, (c) the bank-level organization using column-split arrays, (d) a 3D-integrated array using row-on-row array-splitting.

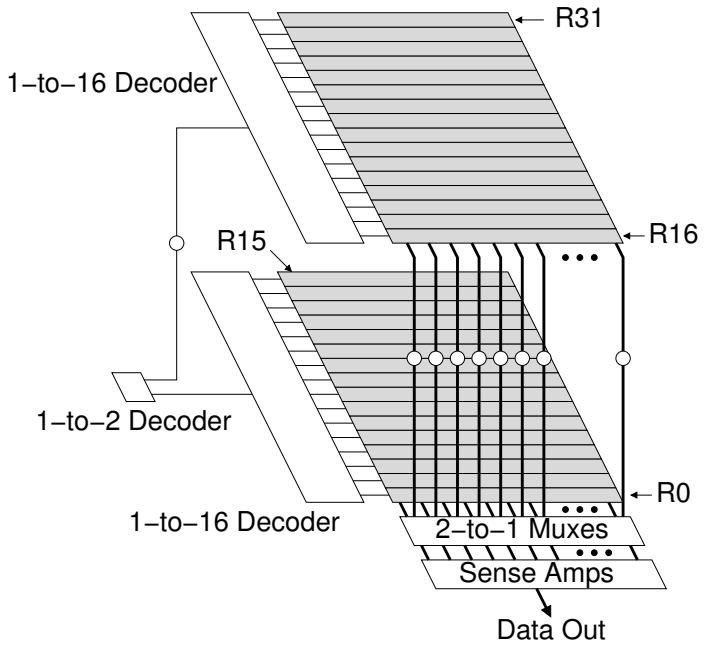

Figure 16: Register-partitioning 3D register file. A ○ represents a die-to-die via.

row decoder height has also been halved, which shortens the critical path in the register file. The overall footprint of the register file has also been halved, which may enable more compact processor floorplans.

To implement a 4-die RP 3D register file, the register entries would be partitioned such that one quarter of the entries reside on each die. The row decoder can be further decomposed in a manner similar to the 2-die-stack.

Note that while Figure 16 shows a 32-entry register file with a single read port, the register file of modern high-performance processors may have 80 [71] or 128 [57] entries. As the number of simultaneously executing instructions increase, the register file size requirement also increases accordingly. The large number of read- and write-ports also exacerbate the area and wire lengths of the register file. Furthermore, the data-widths of modern processors has increased over time from 32 bits to 64 bits, and even 128 bits for some instruction set architectures (e.g., Intel's SSE3 ISA extension), thus requiring correspondingly wider register files.

2.4.2 Bit-Partitioning (BP) 3D SRAM Circuits

The bit-partitioned (BP) 3D register file with a 2-die-stack stacks higher order and lower order halves of the same register across different die. The BP register file can be viewed as the dual of

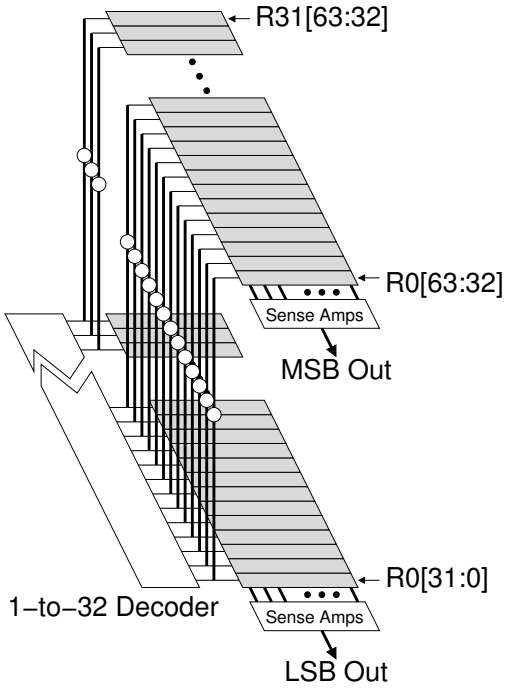

Figure 17: Bit-partitioning 3D register file. A ○ represents a die-to-die via.

the RP organization: RP folds the register file upon itself in the X- direction while BP folds in the Y- direction. Figure 17 shows a 2-die-stacked, bit-partitioned register file where the bottom die stores the least significant half and the top die stores the most significant half of the register. The bit-partitioned register file reduces the wire length and gate loading on the wordline, which provides both latency and energy benefits.

While Figure 17 shows the bits of each register partitioned by significance, one could instead store the bits in odd positions on one die and the bits in even positions on the other die. Choosing one over the other does not impact the latency or power of the BP 3D register file. However, the choice should be made to match the datapaths throughout the rest of the processor. For example, if the ALU circuit designer implements a 3D-integrated integer arithmetic unit partitioned by significance (X[0:31]+Y[0:31] on one die, X[32:63]+Y[32:63] on the second die) [95], then the register file bit-partitioning should also be arranged by significance to avoid unnecessary d2d routing between the outputs of the register file and the inputs of the 3D-integrated arithmetic unit.

The 2-die-stacked BP 3D register file requires the row decoder outputs to be fanned out to both the die. This extra communication incurs a small overhead, but the latency reduction due to the halving of the wordline length provides a significant overall benefit. For a 4-die-stacked BP 3D

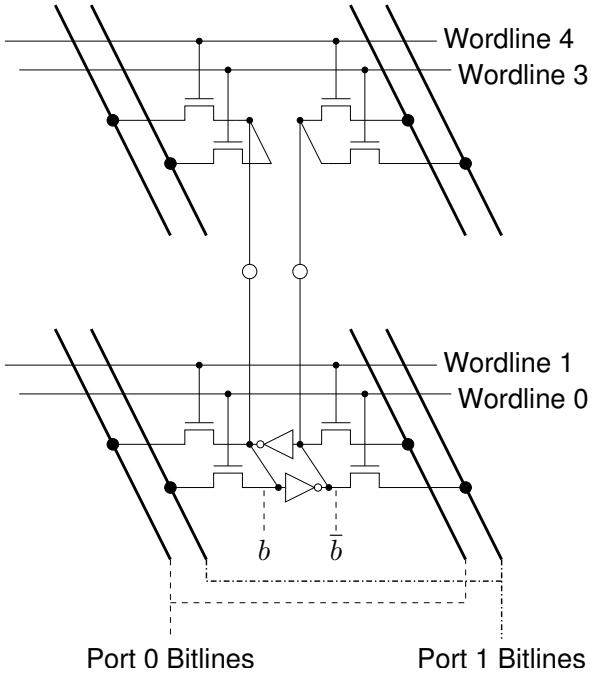

Figure 18: Port-split 3D register file with 2 die-to-die vias per bitcell. A ○ represents a die-to-die via.

register file, the fanout overhead of row decoder output increases since it has to be fanned out to four die. But this latency overhead is insignificant compared to the benefits offered by the 4-die-stacked 3D circuits.

2.4.3 Port-Splitting (PS) 3D SRAM Circuits

The individual SRAM cells in the caches are designed to occupy a small area so as to increase the capacity of the caches, while the area of the SRAM cells in the register files is dominated by the area of the wordlines and the bitlines for implementing multiple read- and write-ports. The relative size of a 6T SRAM cell and a d2d via may make it difficult to implement an individual 6T cell across multiple die [109, 148]. However, SRAM cells in multi-ported SRAM components have a substantially larger footprint (due to the high port count) which may provide the opportunity to allocate one or two d2d vias for each cell. Figure 18 shows a 2-die port-split (PS) SRAM cell where each die contains the bitlines, wordlines and access transistors for half of the ports (either read or write). Two d2d vias are required per bit-cell to route the outputs of the chained inverters to the second die.

The PS 3D register file provides substantial benefits in terms of footprint reduction. Stacking the

wordlines and the bitlines provide a 50% reduction in each of the dimensions leading to an overall footprint reduction of $\sim$75% for the SRAM array. The total footprint reduction may be slightly less than 75% because structures such as row decoders and sense amplifiers may not observe as large of a compaction benefit. This wire reduction translates into substantial latency and energy benefits.

Depending on the multi-ported SRAM cell physical dimensions and the relative sizes of the d2d vias, it may or may not be possible to allocate two d2d vias per SRAM cell. Figure 19 shows an alternative implementation of a 2-die port-split (PS) 3D SRAM cell where only a single d2d via is used to route the data bit b to the second die. On the upper die, an extra inverter recomputes the complement bit $\overline{b}$. This shows that logic duplication may be used to trade-off against excessive inter-die communication. A limitation of the single-via configuration is that the ports on the top die can only support read operations because there is no path to access the "true" $\overline{b}$ storage node.[1] This limitation is likely not critical as the number of write ports is typically much less than the number of read ports.

With two d2d vias per cell, another design alternative would be to split the inverters in the SRAM cell across the two die. This would place all of the b bitlines on (say) the bottom die, and all of the $\overline{b}$ bitlines on the top die. We do not evaluate this configuration as it has several disadvantages. First, the wordlines must be replicated across both die (similar to the BP register file configuration) which eliminates the wirelength reduction in one dimension. Second, splitting the differential bit-lines across more than one die may require designing sense amplifiers that are themselves partitioned across more than one die.

2.4.4 Hybrid Partitioning with More Than Two Die

There are multiple ways to partition the SRAM arrays based on a combination of the previously described implementations of the 3D-integrated SRAM arrays. In this section, we describe some of the examples of hybrid strategies. Hybrid SRAM arrays implemented using more than two die can use a combination of the partitioning strategies. This may be particularly useful in an alternating F2F/B2B die-stacking organization where the d2d via density changes between pairs of die. In a

[1] One could conceivably build a single-ended write port that only changes the value of node b and relies on the SRAM cell itself to override the $\overline{b}$ node, but this would increase the write latency and substantially increase the duration of the short-circuit interval where both PMOS pull-up and NMOS pull-down circuits are active.

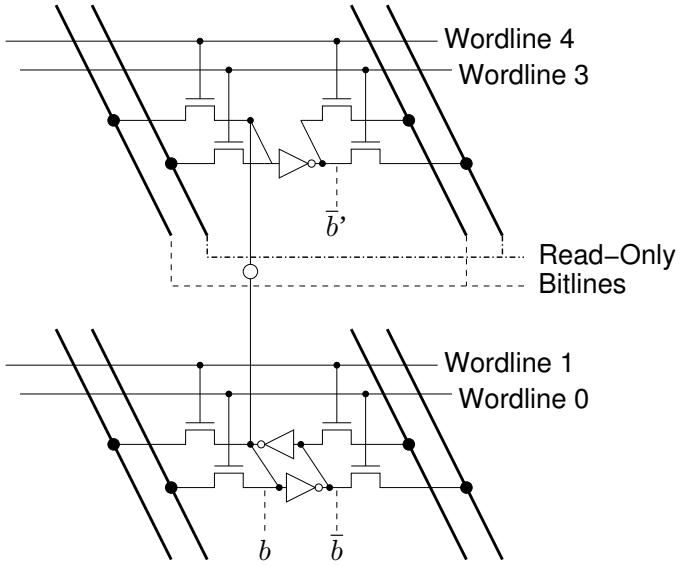

Figure 19: Alternate port-split 3D register file that uses only one die-to-die via per bitcell. A ○ represents a die-to-die via.

4-die-stack with alternating F2F interfaces, one could first use register-partitioning to assign half of the registers to the die-pair 0/1 and the other half to the die-pair 2/3, which limits the usage of the coarser B2B vias to the periphery of the main SRAM array. Then among each pair of F2F die, port-splitting could be employed to exploit the denser F2F interface within the SRAM array. Another possible 3D-integrated design is to use the bank-stacking strategy across the B2B interface and array-splitting strategy across the F2F interfaces.

We illustrate the hybrid partitioning technique with a branch prediction table design. Control flow predictors such as branch predictors constitute a significant portion of the frontend in modern processors. In particular, we consider a branch direction predictor table based on two-bit saturating counters as shown in Figure 20(a). The branch direction predictor table shown in Figure 20(a) is an SRAM array structure that consists of a column of least significant bits called the hysteresis bits and a column of most significant bits called the direction bits. The processor needs the direction bit to make the initial prediction as well as during the update/training phase, while the hysteresis bit is needed only during the update phase. For such branch direction predictors, we can first partition the counters into two separate arrays: one array to store the direction bit and the other to store the hysteresis bit [131]. We implement the two arrays by partitioning them across two die each, as shown in Figure 20(b). Such a partitioning may not only result in more evenly balanced footprints

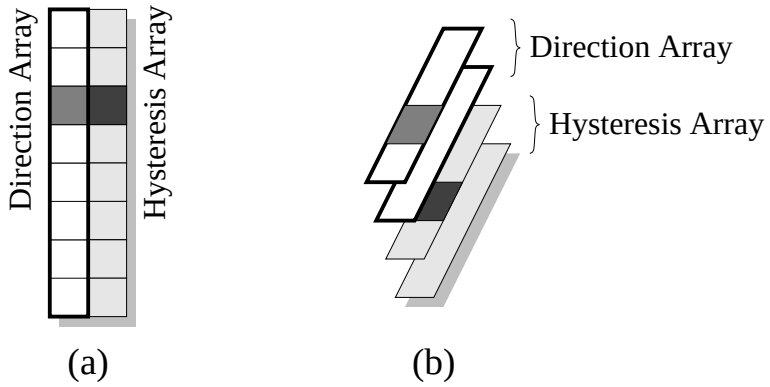

Figure 20: (a) A planar branch direction predictor array, and (b) a 3D-integrated branch predictor array partitioned into two separate sub-tables

Table 2: Impact of the 3D-integration technology on the SRAM array latency.

	Latency (% Benefit), ps			Cycles (3 GHz)		
Cache (KB)	Planar	2-die 3D	4-die 3D	Planar	2-die 3D	4-die 3D
16	458	449 (2)	450 (2)	2	2	2
32	752	635 (16)	584 (22)	3	2	2
64	1232	885 (28)	731 (41)	4	3	3
128	1716	1381 (20)	1233 (28)	6	5	4
256	2732	1929 (29)	1513 (45)	9	6	5
512	3663	2864 (22)	2461 (33)	11	9	8
1024	5647	3945 (30)	3066 (46)	17	12	10
2048	7456	5774 (23)	4875 (35)	23	18	15
4096	11415	7854 (31)	6096 (47)	35	24	19

but also other benefits such as placement of the more frequently accessed arrays on the die closer to the heatsink, resulting in latency, power, and thermal advantages.

2.5 Results

2.5.1 Bank- and Array-Stacked 3D SRAM Benefits

We simulate a cache read critical path including the bank-level and the array-level circuit details. The simulations account for all of the wires and drivers to reach from the origin (array boundary) to the farthest bank. The request propagates through the decoder tree, the wordline, the SRAM cell, the bitlines, the column multiplexor, the sense amplifier and the way-multiplexor. This signal is then driven back through a series of wires and bank multiplexors to return to the origin. While the critical path involves reading a single SRAM entry, our power simulations include the activity associated with *all* of the SRAM cells and their bitlines in the selected row. We perform a similar analysis for the critical path through the tag array and use the worst-case timing path through either

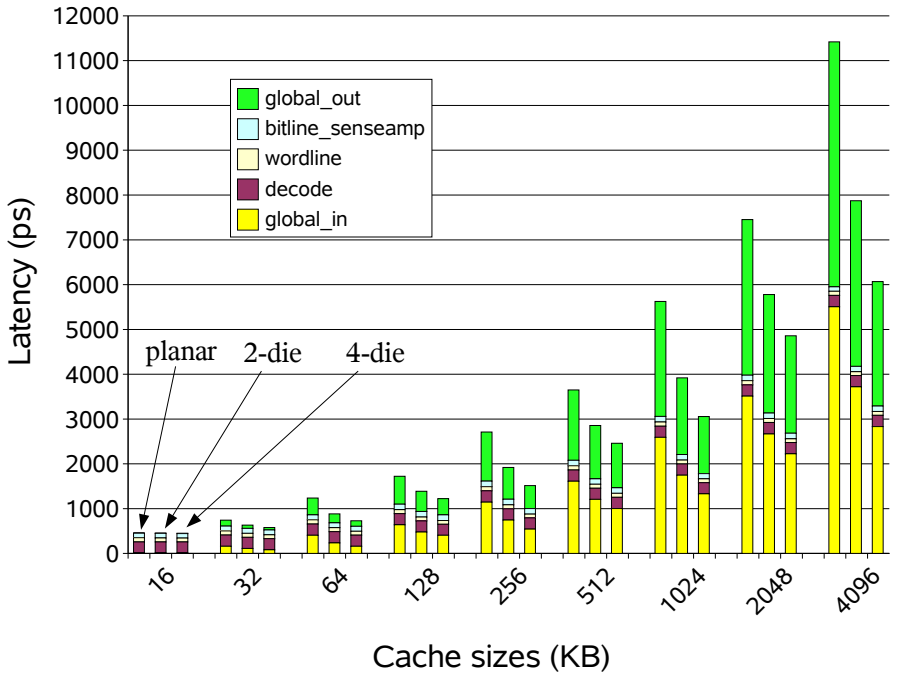

Figure 21: Component-wise breakdown of the SRAM array latencies.

array.

We evaluate our 3D-integrated SRAM circuits against the conventional planar SRAM circuits. We consider array sizes varying from small 16KB level-one (L1) caches up to large 4MB last-level (L2/L3) caches. Our baseline for comparison are the planar caches with transistor sizings and bank configurations chosen to minimize latency. Table 2 reports the size and latency of the baseline planar SRAMs, the latency of the 3D-integrated SRAMs, the relative latency reduction, and the latency in terms of 3GHz clock cycles. The observed latency benefit varies by the SRAM array size. The 3D organization of the larger SRAM arrays provide more benefits because these components have substantially longer global wires to route signals between the array edge and the farthest bank. The latency improvement in Table 2 does not increase monotonically because the best configuration between the planar/2D and the 3D-integrated SRAM arrays may involve different number of banks, which in turn changes the relative benefit. For many 3D configurations, the latency reduction is sufficient to reduce the overall number of clock cycles for the access as shown in the last three columns of Table 2. For the smaller arrays, the wire delays comprise a smaller relative fraction of the overall latency and therefore the effect of reducing these delays is less.

Table 3: Impact of the 3D-integration technology on the SRAM array energy.

	Energy, nJ			Percent	
Cache (KB)	Planar	2-die 3D	4-die 3D	2-die 3D	4-die 3D
16	76.93	74.12	75.37	4	2
32	86.37	86.90	84.06	-1	3
64	87.92	86.49	83.69	2	5
128	102.35	99.17	100.30	3	2
256	106.39	102.70	100.05	3	6
512	108.64	106.38	105.55	2	3
1024	120.75	114.10	107.28	6	11
2048	129.41	121.94	117.86	6	9
4096	150.62	133.32	123.78	11	18

To understand where the 3D-integration has the greatest benefit, Figure 21 illustrates the latency contribution of the different components that comprise an SRAM array. Depending on the array size, different 3D topologies may be required to provide the best latency improvements. As an example, the bank-level routing (global) latency for the 2MB planar cache takes up over 55% of the total latency. Correspondingly, the fastest 3D-integrated 2MB organization exhibits the greatest latency reduction in these timing components, as shown in Figure 21. On the other hand, moderate-sized (64KB-512KB) cache delays are not as dominated by the global routing. In these instances, a 3D organization that targets the intra-SRAM delay provides more benefit. Figure 21 shows that, in particular, the delay reduction associated with the global routing accounts for a substantial part of the overall benefit, while other timing components also observe some improvement. In general, circuit paths dominated by wire delays have the greatest potential for improvement from the 3D-integration. In contrast, the row decoder consists primarily of logic, and the results in Figure 21 are consistent with the expectation that the 3D-integration would not provide much benefit for this component of the overall cache latency.

The 3D organization of the cache structure reduces critical wire lengths, which reduces the total wire capacitance providing both performance *and* energy benefits. For the same cache configurations discussed in the previous section, Table 3 shows the energy consumed per read access for both the planar and the 3D SRAMs. The overall savings range from 2-18%, varying due to differences in the optimal banking configurations, transistor sizings, and SRAM aspect ratios. Note that the energy benefits are in addition to the latency benefits as reported in Table 2. Thus we show that the 3D-integrated SRAM array designs can provide simultaneous latency and energy benefits compared

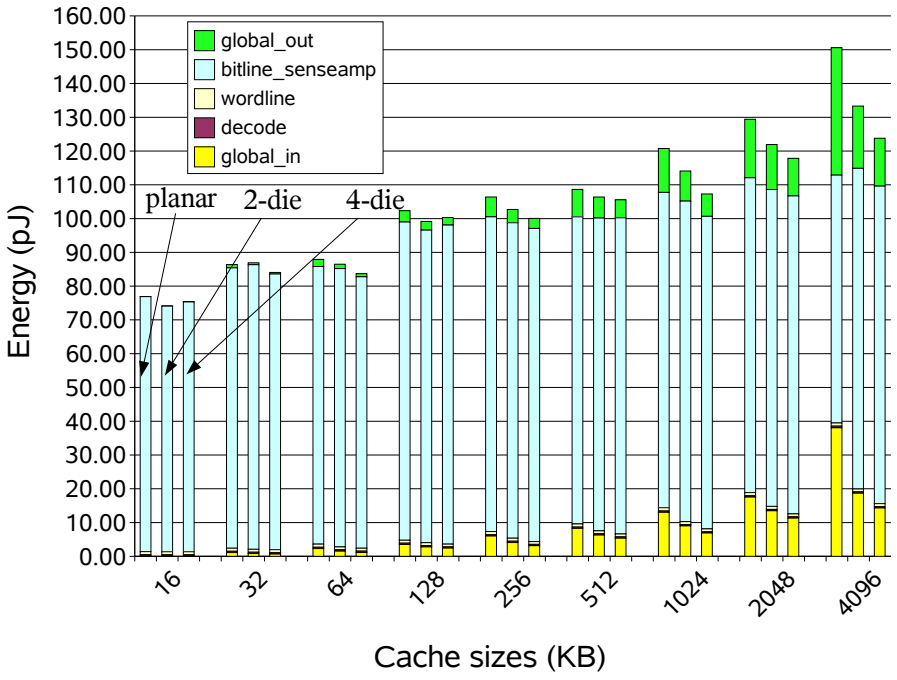

Figure 22: Component-wise breakdown of the SRAM energy (cache read operation).

to the corresponding planar designs.

Figure 22 shows the energy distribution of the SRAMs for a read operation. A component's relative contribution to total latency does not always reflect its impact on energy. The row decoder logic accounts for a considerable amount of latency, but it only needs to switch a single wordline and so it consumes very little energy. On the other hand, a single bitline switch and the corresponding sensing of the bit by the sense amplifier incur a modest energy cost, but this cost is multiplied by the number of accessed bits, which adds up to account for a substantial portion of total access energy. The 3D organization reduces the energy in the most wire-dominated portions of the arrays, namely the bank-level routing and the bitlines.

Figure 23 shows the impact of the hierarchical wordline (HWL) technique on the benefits of the 3D-integrated SRAM circuits. Each bar in Figure 23 shows the percent latency benefit of the 3D-integrated design compared to the corresponding planar baseline design. For example, the first bar shows the latency benefit of the 2-die-stacked 3D cache with no HWL, compared to the planar design with no HWL. The second bar shows the latency benefit of the 2-die-stacked 3D cache with HWL, compared to the planar cache with HWL. For every cache size, the 3D-integration benefits

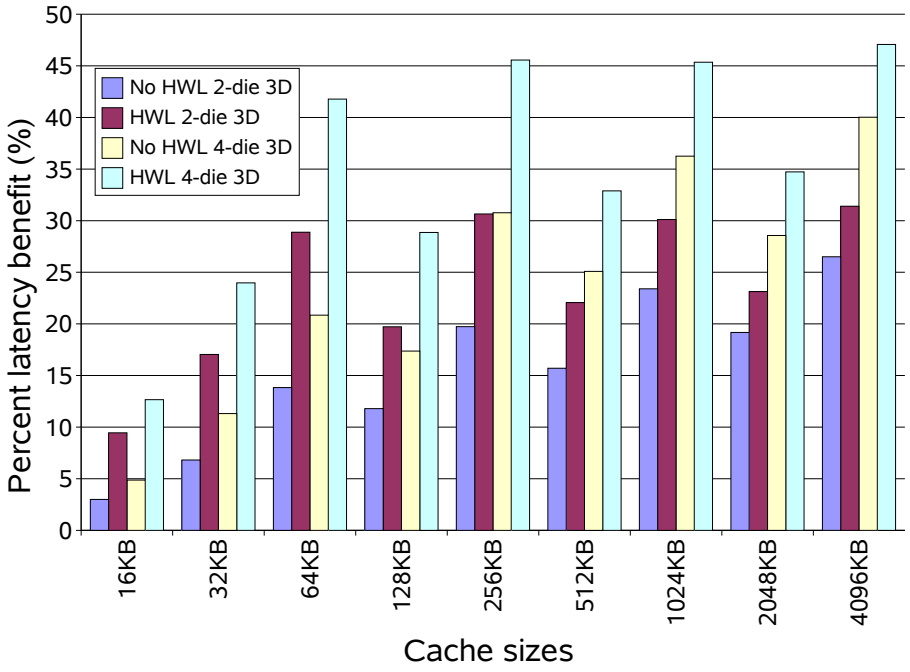

Figure 23: 3D-integration benefits with hierarchical wordlines technique

are even larger for the designs with HWLs. Since HWLs enhance performance, the 3D-integrated SRAM designs are even more attractive in high-performance designs which use techniques such as HWL to improve the performance.

Figure 24 shows the effect of increasing the block size on the benefits of the 3D-integrated caches. For smaller cache sizes (up to 512KB), as the block size increases, the 3D-integrated benefits decrease significantly. For smaller cache sizes, the global wire delays are already a small fraction of the total delays (intra-array delays dominate the total delays) and thus provide less benefits from the 3D-integration. When the block size increases, intra-array delay further increases, and the global wire delay further decreases, thus reducing the 3D benefits. For larger cache sizes, the total delay is dominated by the global wire delays and not the intra-array delays. Hence, for large caches, as the block sizes increase, we do not observe as much reduction in the latency benefits.

2.5.2 Multi-Ported 3D SRAM Benefits

We simulate critical path netlists for different register file configurations based on parameters such as the number of entries, bit-width per entry, and number of ports. The 3D implementations of the

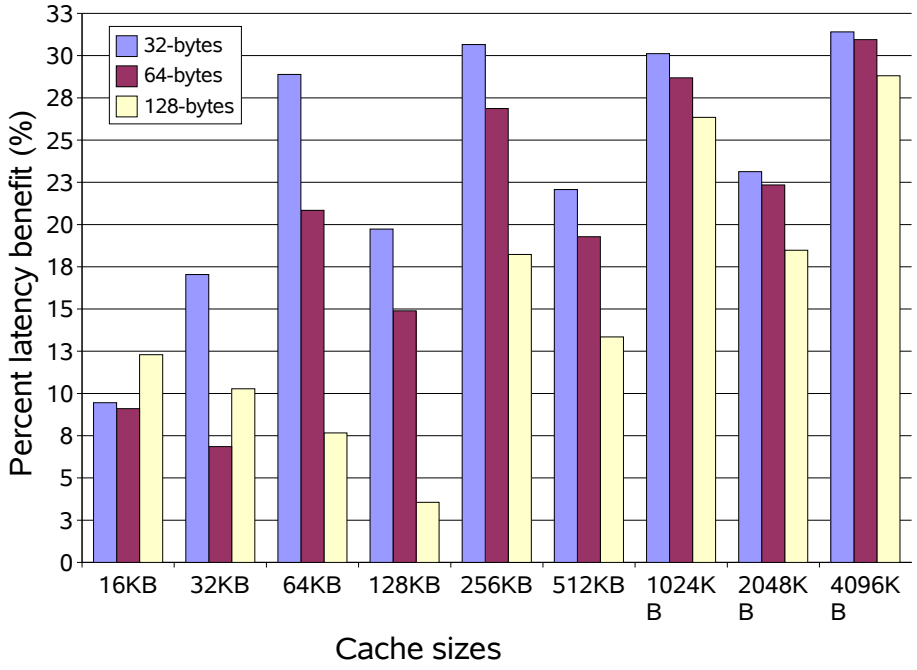

Figure 24: 3D-integration benefits with increasing block sizes.

register file substantially reduce both the latency and the energy. We compare the baseline planar implementations of register files to the 3D organizations explained earlier in Section 2.4.

2.5.2.1 Effect of Increasing the Number of Entries

We evaluate the 3d-integrated register file designs in a high-performance superscalar processor. In particular, we consider a high-performance superscalar processor which can execute four instructions per cycle and can operate on 64-bit operands. Each of the instructions requires two read ports and one write port. Furthermore, to handle multiple instructions per cycle, additional read ports are needed to read the register contents before updating the architected (committed) register file. Overall, the total port requirement is twelve read ports and four write ports for a four-wide issue machine. We simulate the register files with sizes ranging from 16 entries up to 192 entries to model the register file capacity demands of the current and future processors. Table 4 tabulates the planar latencies of the register files and percent latency benefits for each of the 3D configurations considered. Table 4 shows that, for 2-die-stacks, the bit-partitioning (BP) 3D design provides the largest latency benefit when compared to the corresponding RP 3D and PS 3D designs. The wordline is

Table 4: Benefits of 3D multi-ported SRAM configurations with increasing number of entries.

RF		% 2-die		
entries	Planar (ps)	BP	RP	PS
16	258	38	3	10
32	307	34	4	12
64	334	27	7	13
96	383	25	5	11
128	398	23	9	14
160	442	24	14	19
192	458	24	15	19

RF		% 4-die				
entries	Planar (ps)	BP-BP	RP-RP	PS-PS	BP-RP	BP-PS
16	258	43	4	17	40	43
32	307	37	8	17	37	40
64	334	32	12	20	38	43
96	383	25	13	21	32	39
128	398	26	14	22	33	36
160	442	25	20	28	35	40
192	458	27	21	28	35	58

heavily loaded by the two access transistors per bitline column. Hence, splitting the wordline across two die reduces a major component of the wire latency. Since the BP design reduces the wordlines, it provides a large latency benefit. Note that, as the number of entries increase, the latency benefits of the BP 3D register files decrease. This is because, as the number of entries increase, the height of the overall structure increases, thus making the row decoder and bitline/sense-amplifier delay to become increasingly critical. Bitlines become more heavily loaded thus reducing the contribution of the wordline load to the overall latency. The 2-die-stacked RP 3D designs do not provide as much latency benefit as the corresponding BP 3D designs. But, as the number of register entries increases, the benefits of the RP 3D design also increase. Although the PS 3D designs have a substantially smaller footprint, they do not provide the fastest performance due to the fact that the access transistor loading on the wordlines has not been reduced. With increasing number of entries, the wire-complexity keeps increasing. Hence, the 2-die-stacked PS 3D designs also exhibit increasing savings with increasing number of entries.

In case of 4-die-stacks, there are more design options when implementing 3D-integrated circuits. The notation of X-Y for 4-die-stacks means that the B2B interface has a X partitioning and the F2F interfaces have a Y partitioning. A RP-BP organization places one half of the registers on the die-pair 0/1 and the other half on the die-pair 2/3 (RP across the B2B interface and BP across the F2F

interfaces), and then within each pair of die each register's bits are split as half on each die. With 4-die-stacked multi-ported SRAM components, the BP-BP configuration saves more latency than either of the RP-RP and the PS-PS designs. However, with increasing number of entries, the RP-RP and the PS-PS designs provide comparable benefits as the BP-BP designs owing to higher load reduction and wire reduction respectively. Of particular interest are the hybrid configurations, the BP-RP and the BP-PS, which consistently provide large benefits as compared to the non-hybrid 4-die-stacked configurations. This makes sense since one 3D partitioning technique may reduce the latency of a critical wire delay by a significant amount such that it is no longer the worst delay in the circuit. A second different 3D partitioning can then address the new worst delay. In fact, the maximum benefit for each register file size among all the 3D designs is provided by the BP-PS design. The PS configuration is via-intensive and thus an unsuitable candidate for the B2B interface that require vias etched through silicon. However, the PS design reduces the wires in both X- and Y- dimensions resulting in higher wire savings and can be implemented beneficially across the F2F interfaces. This latency reduction may potentially be converted into a reduction in the number of cycles necessary to access the register file.

The latency benefits of the hybrid configurations highlight the generality of our 3D-integration techniques. For different processor configurations, the critical wire delays within the register files will likely be different. The benefits of the 3D-integration are not limited to specific processor microarchitectural parameters; for each planar circuit, the circuit designer can choose the appropriate 3D-integration circuit.

Table 5 shows the best absolute latencies of the planar, the 2-die-stacked 3D, and the 4-die-stacked 3D register files and the corresponding percent benefits. From the table, we can see that the 2-die-stacked circuits provide 25-40% latency benefit over the planar circuits while the 4-die-stacked designs provide 36-58% latency benefit over the planar circuits. From the data, we can conclude that the 4-die-stacked design provides an additional 10% benefit as compared to the 2-die-stacked design. The latency benefit is not doubled for the 4-die-stack as compared to the 2-die-stack because the 4-die-stacked designs require backside vias to go through silicon, which adds area overhead and reduces the latency benefits.

Table 5: Access latencies of register files for a 4-wide, 64-bit superscalar processor, and the percentage benefit compared to the baseline planar implementation, for increasing entries.

RF entries	Planar (ps)	2-die 3D (ps)	4-die 3D (ps)	% 2-die	% 4-die
16	258	159	146	38	43
32	307	202	185	34	40
64	334	242	190	27	43
96	383	286	234	25	39
128	398	306	253	23	36
160	442	337	265	24	40
192	458	349	193	24	58

2.5.2.2 Effect of Increasing the Data-widths

In some microarchitectures, the physical register file also contains some instruction status information such as instruction pointer value, which is required for maintaining in-order retirement of instructions; each of the entries in such physical registers may contain as many as 160 bits of data [133]. We explore a 96-entry register file with data-widths increasing from 64 bits to 192 bits in increments of 32 bits and report the benefits in Figure 25. From the figure, we can deduce that the 2-die-stacked ciruits provide 25-50% latency benefit over the planar ciruits while the 4-die-stacked 3D circuits provide 40-75% latency benefit over the planar circuits. Note that the latency savings are higher in this case as compared to increasing number of entries since increasing data widths aggravate the already heavily loaded wordlines.

2.5.2.3 Effect of Increasing the Port Requirements

We explore a 96-entry, 64-bit wide register file with increasing issue-widths from 2-issue to 6-issue as shown in Figure 26. Since every instruction requires at least two read ports to read the source operand registers and one write port to write the destination register, each increment in issue-width adds two additional read ports and one additional write port to the register file. Note that the planar circuit latency degrades at a much faster rate than the corresponding 3D circuit latency. With increasing issue-widths, the latency benefits of the 3D-integrated circuits continuously increase compared to the corresponding planar circuits.

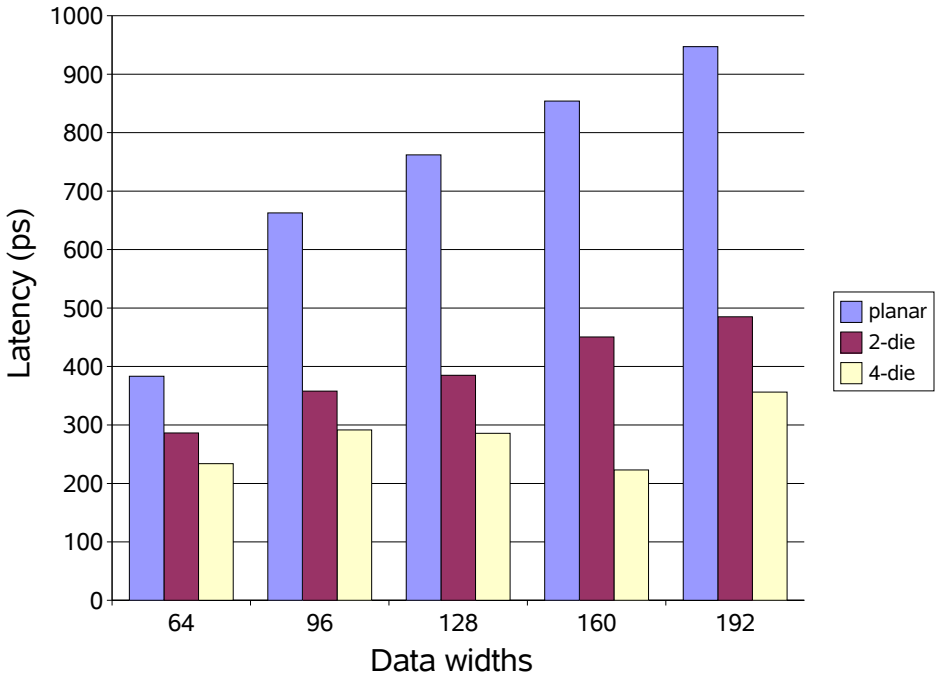

Figure 25: Access latencies of a 96-entry register file (4-issue processor) with increasing data widths.

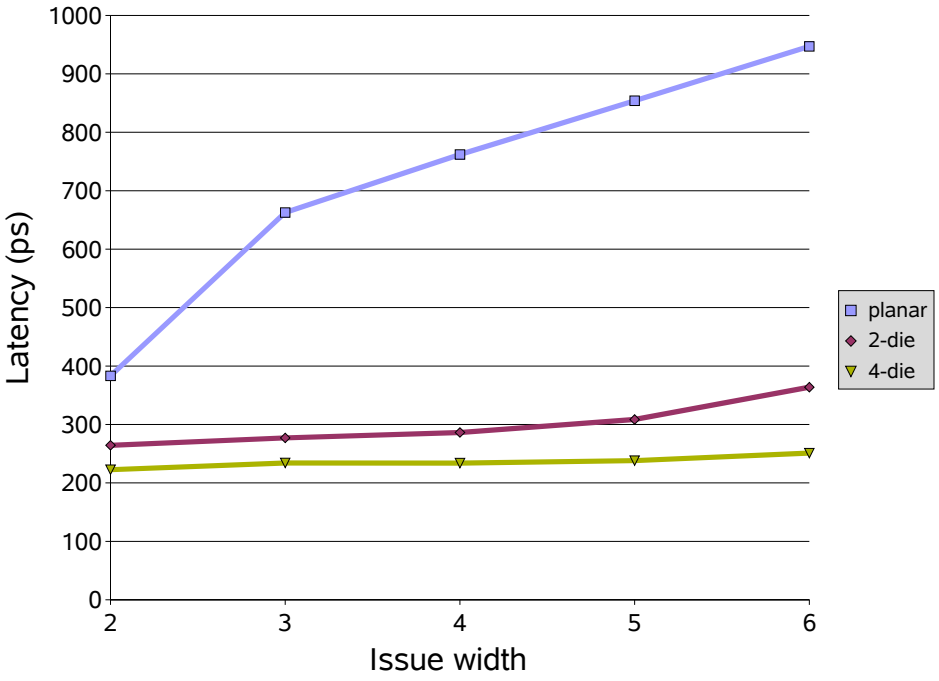

Figure 26: Access latencies of a 96-entry, 64-bit register file with increasing issue-widths.

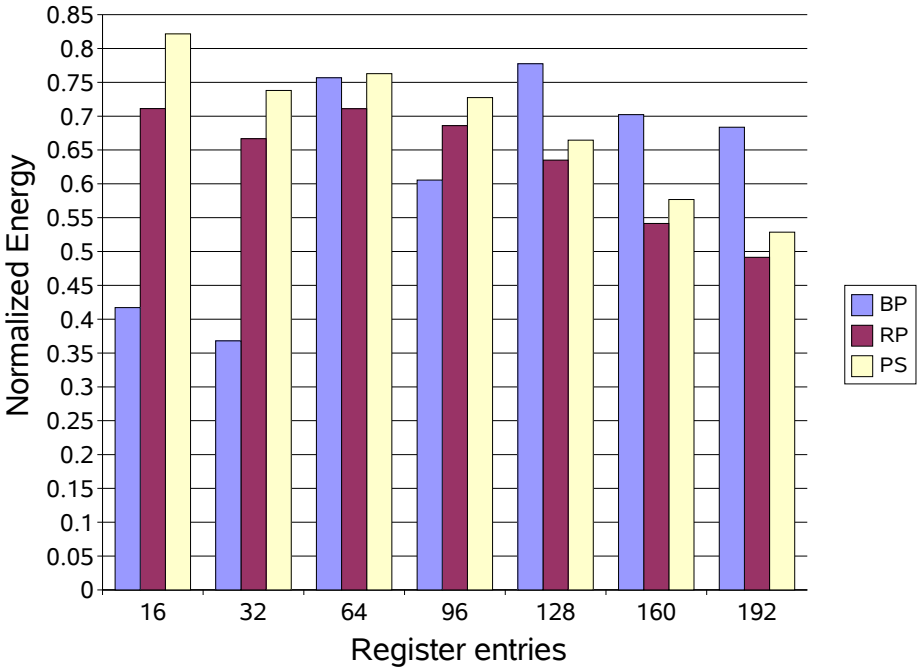

Figure 27: Energy benefits of the 2-die-stacked 3D register file designs.

2.5.2.4 Energy Benefits of the Multi-Ported 3D SRAM circuits

In addition to reducing the latency of the multi-ported SRAMs, the 3D organization also reduces the energy consumption. Figure 27 shows the normalized energy (normalized with respect to the corresponding planar circuit) required to access the 2-die-stacked 3D register files. The 3D configuration that minimizes energy consumption is not necessarily the configuration that has the lowest latency. For smaller number of entries, the BP 3D circuit requires the least energy. As the number of entries increase to more than 64 entries, the RP 3D circuit provides the most energy benefit. The RP 3D circuit effectively halves the bitline length for each doubling of the number of stacked die. The shorter bitlines greatly reduce the loading on the sense amplifiers, which in turn may be sized smaller to consume even less energy.

Figure 28 shows the normalized energy required to access the 4-die-stacked register files. In the case of the 4-die-stacks, either the RP-RP or the BP-RP organizations provide the most energy benefits among all the 3D circuits.

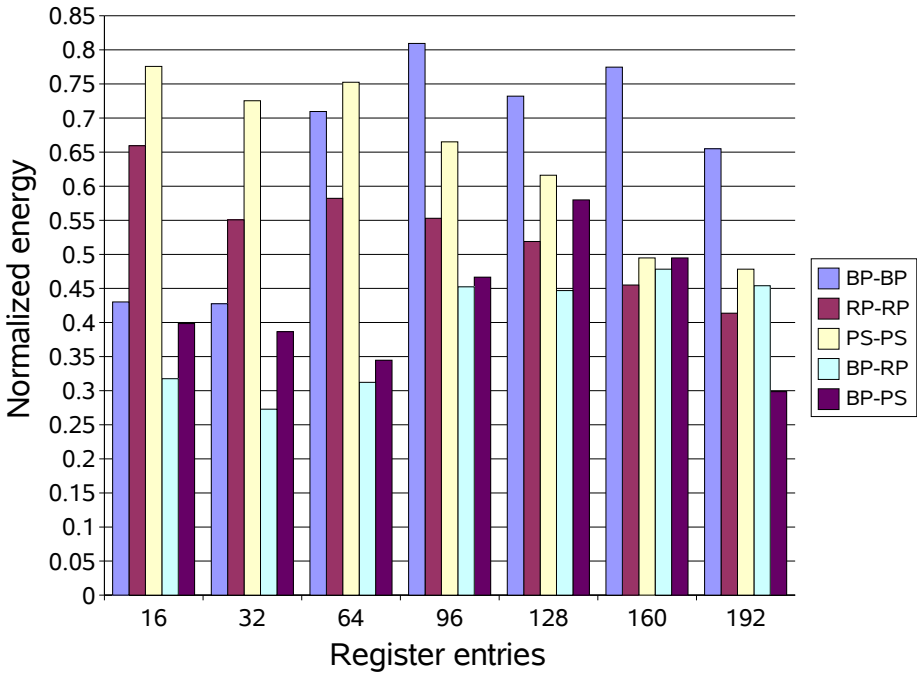

Figure 28: Energy benefits of the 4-die-stacked 3D register file designs.

2.6 Summary of the 3D-Integrated SRAM Components

In this chapter, we focused on the SRAM array based components of the processor. We described large-capacity SRAM components that use banking, subbanking, and hierarchical wordlines as well as small-capacity, multi-ported components that are latency-critical. We described the designs of the 3D-integrated SRAM components based on different partitioning strategies. We analyzed the SRAM circuits in terms of latency and energy benefits for both the 2-die and the 4-die implementations.

We demonstrated that the 3D-integrated SRAM components reduce the lengths of critical wires. We showed that the 3D-integrated SRAM components provide significant performance and energy benefits simultaneously, thus supporting our thesis statement for the SRAM components in the high-performance processors.

When the planar SRAM components scale up in size or adapt circuit and microarchitectural techniques to enhance performance, the corresponding 3D-integrated circuits provide even higher benefits, making the 3D-integration technology well-suited for high-performance processor designs.

Due to the increasing wire delays with technology scaling, the benefits of the 3D-integration technology will increase in the future technology generations, making 3D-integration very attractive for future designs.

Other table-like components such as the decoder PLAs, the microcode ROM in x86 processors, and the lookup table for SRT dividers [54] may also observe similar benefits from a 3D-integrated implementation.

CHAPTER III

ASSOCIATIVE LOGIC COMPONENTS

3.1 Overview of This Chapter

Having supported the performance and power claims of our thesis statement for the SRAM components in Chapter 2, this chapter extends our claims to be applicable to the CAM-based (associative logic) components as well. This chapter focuses on the planar CAM components and our designs of the 3D-integrated CAM components. We make use of the dense die-to-die vias to propose 3D-integrated CAM components that are partitioned at the level of individual CAM entries and tags. Our 3D-integrated CAM designs result in significant wire length reduction and footprint reduction and hence, provide simultaneous latency and power benefits.

The rest of the chapter is organized as follows. Section 3.2 describes the conventional planar CAM components. Section 3.3 explains our designs of the 3D-integrated CAM components. Section 3.4 presents the results and analysis of the planar and the 3D-integrated CAM components. Section 3.5 summarizes this chapter.

3.2 Planar CAM Components

Associative logic or content addressable memory (CAM) circuits are high speed logic circuits used in operations that involve search/match operations in a microprocessor. A CAM circuit consists of an array of associative cells, each of which contains both memory and processing elements. Each associative cell in the array stores a key (tag) and a value (operand) in the memory. The operation of finding the value associated with a tag is called a lookup. For the purposes of lookup, a requestor broadcasts the tag on a common bus. Each associative cell compares the broadcast tag with its own tag. If there is a match, the operand value of the matched tag is provided to the requestor. If there is no match, the requestor is notified that there is no match.

The CAM circuits search the entire array in a single operation and hence extremely fast in servicing a lookup request. However, there are cost disadvantages to the CAM-based circuits. To provide associative search functionality, the CAM cell requires an SRAM cell, a comparator, and

some additional logic. Often, the comparator is implemented using dynamic logic to accomplish the search quickly. Hence, the CAM circuits have not only a very high wire-complexity but also very high power dissipation. For a w-issue CAM design, there are w tag broadcast buses, and each entry must compare its tags with those on the buses. This requirement forces the height of the CAM cell to be $\Omega(w)$ and the width of the broadcast buses to be $\Omega(w)$, and so the broadcast buses alone consume $\Omega(w^2)$ area. Consequently, the CAM circuits are usually used in performance-critical circuits in the modern processors.

Several components in modern processors require content addressable memories (CAMs) to perform fully-associative searches. These circuits are notorious for their poor scaling due to wire delays [106] Addressing this problem for the dynamic instruction scheduler in particular has generated a large amount of research [41, 20, 96, 116, 25, 139, 47]. In the following discussion, we will use the dynamic instruction scheduler (also known as the issue logic) as a representative example of CAM components. The techniques presented for implementing 3D-integrated designs of the conventional instruction scheduler easily translate to the other CAM structures.

3.2.1 Planar Instruction Scheduler Circuit

The dynamic instruction scheduler is responsible for exposing instruction level parallelism by identifying instructions that can be executed in parallel. The capacity of the scheduler to identify opportunities for instruction level parallelism increases with the number of entries in the scheduler. Unfortunately, the latency and energy characteristics of the scheduler do not scale well with increasing number of entries [106]. Given that technology scaling has created an ever-widening gap between the relative delay of logic and wires [3, 122], increasing the number of entries in the dynamic scheduler worsens the wire delay, thus reducing the overall performance of the processor.

The dynamic instruction scheduler consists of two components, namely the wakeup logic and the select logic. The wakeup logic consists of a broadcast bus, a set of comparator circuits and buffers called reservation stations. The reservation station (RS) entries contain source operand addresses (called tags) of the instructions that have been dispatched from the in-order front-end of the processor. The instruction in each valid RS entry is ready for execution as soon as all its source operands become available. The wakeup logic requests functional units to execute the ready

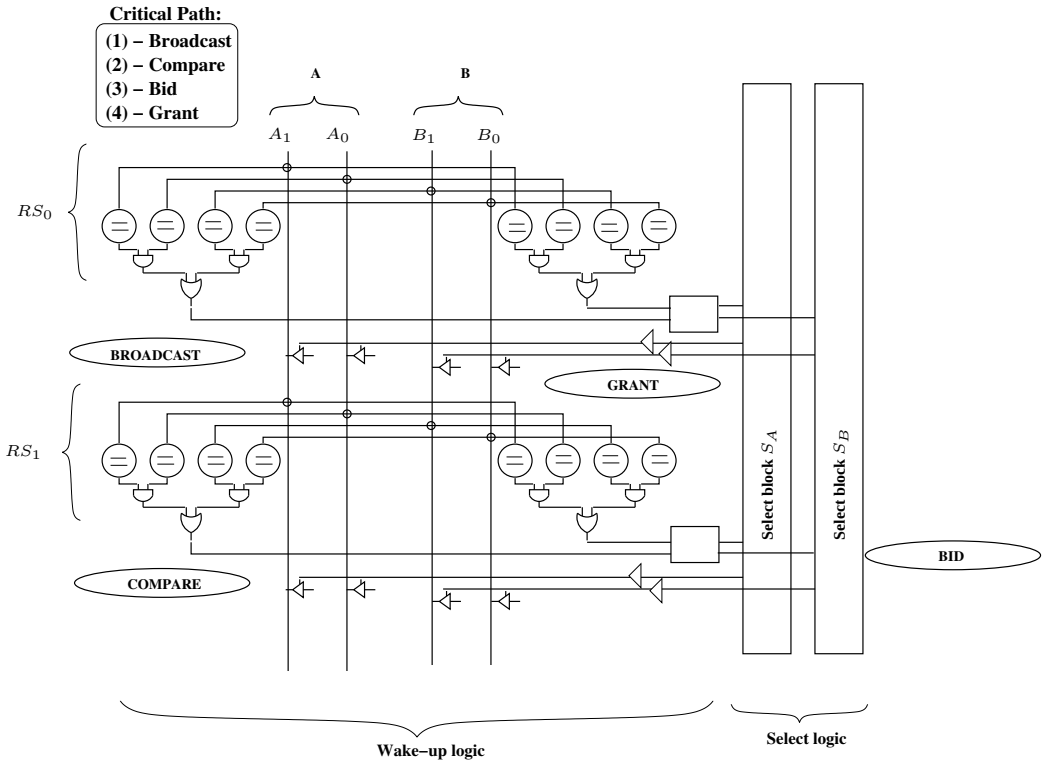

Figure 29: A planar dynamic instruction scheduler circuit

instructions. The select logic matches the requests from the wakeup logic to the available functional units and grants the functional units to the selected incoming requests. The selected instructions issue to the functional units and broadcast their destination tags on the broadcast buses. The wakeup logic simultaneously compares the source operand tags in the RS entries to the broadcast tags and generates a new set of ready instructions based on the tag comparison match. The new set of ready instructions along with all unsuccessful (unselected) instructions from the previous cycle repeat their requests during the current cycle. Every cycle, all ready instructions participate in a bidding process by placing requests for functional units to the select logic and the select logic grants the available functional units to the selected requesting instructions. This chain of events that repeats every cycle is called the wakeup-select loop.

Figure 29 shows a small, 2-issue dynamic scheduler with 2-bit tags and two RS entries. A and B denote the tag broadcasts for the two issues, with A_1A_0 and B_1B_0 being the 2-bit tags. The select logic consists of two blocks, S_A and S_B with each select logic block in charge of one issue (and one resultant tag broadcast). Figure 29 also shows the critical path through the wakeup-select

loop starting at tag broadcast (1), compare logic (2), bid logic (3) and grant logic (4). For every additional RS entry, the length of the longest tag broadcast wire in the wakeup logic increases. Since the select logic height is pitch-matched to the wakeup logic height, the select logic suffers increased wire delays in addition to the increased logic circuitry to accommodate the requests originating from the additional entries. Thus, for increasing number of RS entries, the overall latency of the scheduler also increases.

The latency of the wakeup logic is influenced by two sources. One, the comparators connected to the tag broadcast bus increase the load capacitance, thus increasing the delay. Two, the critical path wire-length of the broadcast bus increases thus increasing both resistance and capacitance of the wire, thus contributing to the total delay. These two factors cause the wakeup-select loop to experience a rapidly diminishing frequency of operation as the scheduler size increases, thus negating the gains of a larger scheduler.

3.2.2 Other CAM Logic Circuits

In addition to the conventional instruction scheduler, CAMs are central to the implementation of other critical components such as the load and store queues, fully-associative translation lookaside buffers, certain register renaming implementations [87], and several other smaller buffers and queues (e.g., stream buffers and victim caches [66]). The register alias table (RAT) portion of the register renaming logic [106] holds the logical register to physical register mappings. In order to check the rename dependencies without creating latency bottlenecks, the RAT is often implemented as CAM logic. The load and store queues (LQ/SQ) are CAM logic components that support out-of-order memory scheduling and store-to-load forwarding [142, 141] in modern high-performance processors. The load queues and store queues maintain the program order of memory operations by keeping track of all the instructions requiring memory accesses. They track both the data and addresses of memory instructions.

The various 3D-integrated designs illustrated using the instruction scheduler extend naturally to these CAM components. We evaluate three CAM components, namely, register alias table, load queue, and store queue, and present the results in Section 3.4.

3.3 3D-Integrated CAM Components

We propose the designs of the 3D-integrated CAM components using the instruction scheduler as the representative CAM component. We reduce the broadcast wire-lengths and the comparator loads of the CAM mechanisms (e.g., wakeup-select loop).

3.3.1 Entry-Partitioned (EP) 3D CAM Circuits

We propose our first 3D-integrated scheduler circuit to reduce both the comparator loading and the wire-length of the longest tag broadcast wire as compared to the planar scheduler circuit. We place half the entries on the top die and the other half on the bottom die. In Figure 30, entries RS_0 and RS_1 are partitioned across the two die such that RS_0 is on the top die and RS_1 is on the bottom die. This design halves the wire-length and the comparator loading of the tag broadcast bus by halving the number of RS entries per die. The reduction in the comparator loading and the broadcast wire lengths speed up the tag broadcast, thus providing latency and power benefits. Note that the select logic is partitioned across the two stacked die such that S_A is on the top die and S_B is on the bottom die. Since the select logic height is pitch-matched to the wakeup logic height, the select logic height also reduces, providing additional latency/energy savings.

There is a logic overhead of one extra driver per die. In addition to the additional driver, the 3D-integrated CAM circuit incurs a minor overhead of two die-to-die (d2d) vias on the critical path, one each for the bid and the grant signals as shown in Figure 30. However, the additional latency and power consumption due to the additional driver and the d2d vias are more than compensated by the large reduction in the wire-length and the comparator loading on the critical path.

3.3.2 Tag-Partitioned (TP) 3D CAM Circuits

With our tag-partitioned (TP) 3D scheduler circuit, we partition half of the broadcast tags on the top die (A_1A_0 in Figure 31) and the other half on the bottom die (B_1B_0). This halves both the vertical wire-length and the horizontal wire-length of the tag broadcast bus. Since both the height and the width reduce by half, the wakeup logic area gets reduced to a quarter of the planar wakeup logic. The comparator loading on the tag broadcast bus remains the same as in the planar circuit. The select logic circuits are also stacked, further reducing the overall area footprint and providing

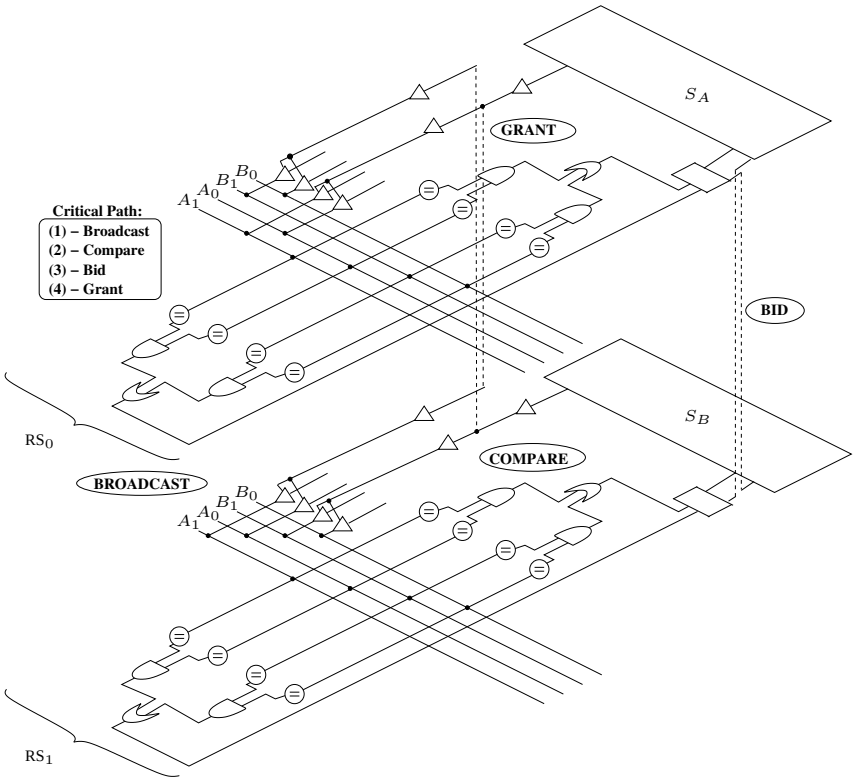

Figure 30: An entry-partitioned 3D CAM circuit

additional benefits in latency and energy.

The TP design incurs a minor latency overhead of two d2d vias on the critical path. However, the latency overhead of the two d2d vias is insignificant when compared to the benefits obtained from reducing the critical path wire-lengths. The comparator match signals are routed through d2d vias to each of the die. The bid signals are routed through d2d vias to enable the wakeup logic to bid on either of the select blocks S_A and S_B. However, the grant signals do not need to be vertically routed since the select logic block that enables the tag broadcasts on a particular die is co-located on the same die.

Note that it is also possible to create an alternate partition of the planar CAM circuit in the 3D-integration technology by bit-slicing the individual tags into two halves and placing each half on one of the two vertically stacked die. If A_1A_0 and B_1B_0 are the two 2-bit tags, we can route A_1 and B_1 on the top die and A_0 and B_0 on the bottom die. While this design has a different d2d via requirement than our proposed TP design, it provides identical latency/area benefits as compared to our TP 3D design.

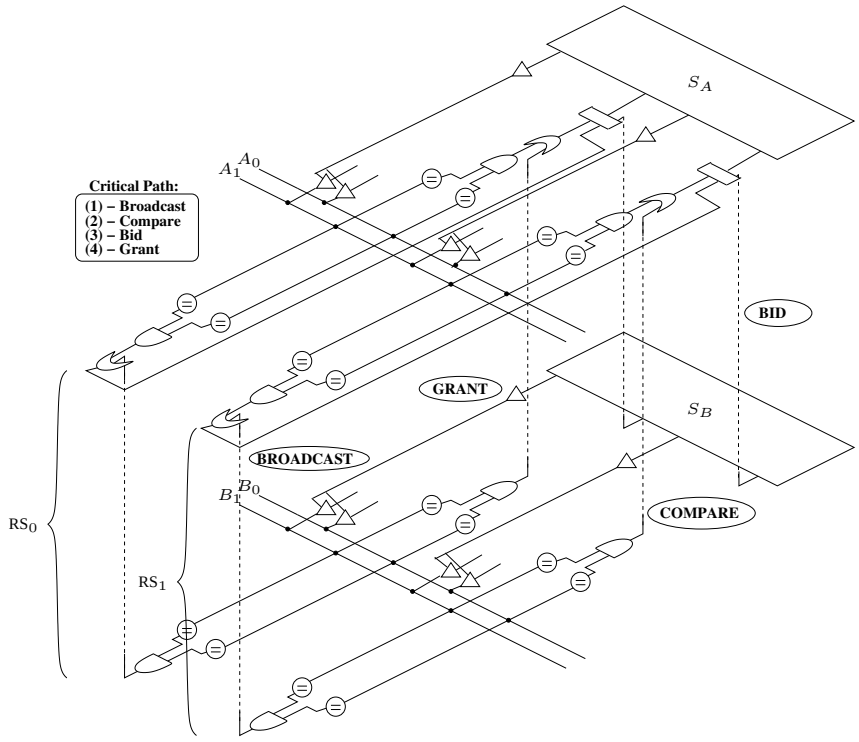

Figure 31: A tag-partitioned 3D CAM circuit

3.3.3 Extending to More Than Two Die

Without loss of generality, we discuss 4-die-stacked 3D scheduler circuits using entry-partitioning and tag-partitioning techniques. The 4-die entry-partitioned (EP) scheduler places one quarter of the RS entries on each die. Additional backside vias are required for routing the bid/grant signals to/from the select logic. Due to the keep-out regions of the backside vias, a naive routing can lead to substantial area overhead within the RS entries. However, Figure 32 shows a technique to insert space between the RS entries and the select logic blocks to avoid disrupting the devices in either the RS entries or the select logic. This may increase the wire-length slightly, however this increase is more than offset by the wire length reduction of the long tag broadcast buses.

For the 3D-integrated designs of more than two die, it is important to optimize the backside via requirements. Using the backside vias for the bid and the grant signals minimizes the overhead because each of the bid and the grant signal requires a single bit; whereas fanning out the actual tags across the backside interface would require substantially more backside vias.

The 4-die tag-partitioned (TP) scheduler is more challenging than the EP 3D scheduler since

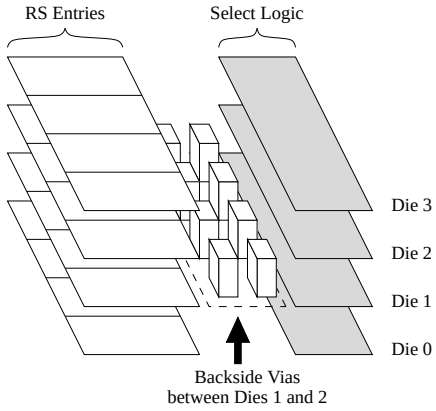

Figure 32: A 4-die-stacked EP scheduler with extra space allocated for backside vias. Dimensions not to scale.

the inter-die via locations are spread over a large region. To avoid disruptions to the RS layout, one could place all of the backside vias between the RS entries and the select logic in a fashion similar to the EP scheduler. This incurs some additional wire delay to communicate between the RS entries and the backside vias.

Another alternative design would be a hybrid approach where entry partitioning is used to place half of the entries on one die-pair (0/1), and the other half of the entries on the other die-pair (2/3). This hybrid technique uses the EP design across the backside via interface. At the F2F interfaces, either the TP or the EP can be used depending on benefits and feasibility.

3.4 Results

We present the results for the 2-die-stacked and the 4-die-stacked 3D implementations in this section. For all simulations, we assume a scheduler that can broadcast up to four 7-bit tags and issue up to four instructions, per cycle. Such a configuration models the scheduler of a 4-issue processor with 128 registers. We present our results for schedulers ranging from 20-entries to 120-entries to chart the trends for both the current and the future high-performance processors.

3.4.1 2-Die-Stacked 3D CAM Benefits

Table 6 shows the latency results for the planar and the 2-die-stacked 3D configurations for scheduler entries increasing from 20 to 120 entries. From Table 6, the 3D scheduler circuits have smaller overall latencies than the planar schedulers for identical number of RS entries. For large instruction schedulers, the overall latency is dominated by the tag broadcast component. Consequently, the 3D

Table 6: Scheduler latencies for planar, entry partitioned (EP), and tag partitioned (TP) 2-die-stacked 3D organizations.

RS entries	Latency (ps)			Saving	
	Planar	EP(2-die)	TP(2-die)	% EP	% TP
20	577	526	571	8.9	1.1
30	623	562	611	9.7	1.8
40	665	582	645	12.5	3.0
50	703	602	681	14.4	3.0
60	739	627	714	15.1	3.4
70	776	653	743	15.8	4.2
80	804	671	771	16.5	4.1
90	839	702	802	16.3	4.4
100	871	706	828	18.9	5.0
110	902	730	855	19.1	5.2
120	927	747	883	19.4	4.8

technology provides greater relative benefit for large sized schedulers. The latency benefits of the EP 3D designs are consistently greater than those of the TP 3D designs. This is due to the fact that the EP has the advantage of both reduced wire-length and reduced comparator loading while the TP is benefited only by reduced wire-length. Although the TP provides drastic reduction in the wire-lengths (the TP 3D circuit footprint being a quarter of the original planar circuit footprint), it does not reduce the comparator loading per broadcast wire. In fact, the benefits of the TP 3D circuit begins to deteriorate as we increase the number of entries beyond 100 due to the dominating effect of the comparator loading. The results listed in Table 6 show that the latency can be reduced by 9-19% depending on the number of entries in the scheduler. If the instruction scheduling logic is the limiting factor of the overall processor frequency [106], then this latency improvement can be directly translated into performance by increasing the 3D-integrated processor clock speed. Alternatively, the 3D-integrated processor frequency can be fixed to be the same as the planar processor frequency, while increasing the number of entries in the scheduler to expose more instruction level parallelism. In most situations, a scheduler with approximately twice as many entries can be implemented within the same latency as the original planar circuit. This is largely due to the fact that the tag broadcast latency scales quadratically with increasing wire length.

We measured the energy consumption of the scheduler circuits using activity profiles collected from benchmark suites such as SPEC2000 [34] and MediaBench [80]. By combining the scheduler activity over the period of benchmark program execution and the circuit latency and power data

Table 7: Scheduler latencies for planar and entry-partitioned (EP) 4-die 3D organizations.

RS entries	Latency (ps)		Saving
	Planar (1-die)	EP (4-die)	% EP
20	577	506	12.3
30	623	521	16.3
40	665	525	21.1
50	703	548	22.0
60	739	563	23.7
70	776	573	26.2
80	804	578	28.0
90	839	596	29.0
100	871	603	30.8
110	902	617	31.6
120	927	630	32.1

obtained by Hspice circuit simulations, we calculate the energy. For a 20-entry scheduler, we found that the EP 3D circuit reduces the total energy consumption by 18% over all the benchmarks. For larger configurations such as the 120-entry case, the energy reduction can be as much as 45%. The 3D designs reduce the energy due to the reduced resistances and capacitances of the wires.

3.4.2 4-Die-Stacked 3D CAM Benefits

We consider only the entry-partitioned (EP) organization because of its superior latency and energy benefits. Table 7 shows the latencies of 4-die schedulers over a range of sizes. The planar latencies are repeated here for reference. Overall, the latencies further improve as compared to the 2-die-stack, ranging from 12-32%. Empirically, it appears that the doubling in the number of die provides $1.5\times$ the benefit. The energy benefits for using the 4-die-stack are similar, with the 20-entry scheduler circuit providing up to 25% energy reduction and the 120-entry scheduler circuit providing up to 67% reduction.

Table 8 shows the results of 3D-integrating other CAM components namely, register alias tables (RATs), load queues (LQs), and store queues (SQs) using 4-die-stacks. These CAM components scale poorly in terms of the wire-complexity in modern processors [142]. Note that the latency benefits of all the CAM circuits in Table 8 are substantial even though they are not as large as some of the SRAM components such as the L1 caches.

Table 8: Latency benefits of the 3D-integration technology for other CAM components

Component	Size	Planar latency (ps)	3D latency (ps)	% latency benefit
RAT	594.0	96 entries	378.6	36.3
LQ	345.7	32 entries	253.1	26.8
SQ	336.2	20 entries	248.6	26.1

3.5 Summary of the 3D-Integrated CAM Components

In this chapter, we focused on the CAM based components in the high-performance processors. 3D-integrated CAM components can provide significant benefits in terms of both performance and power consumption. We demonstrated how the 3D-integration technology can be applied to the design of the CAM components to reduce the lengths of the critical wires. The reduction in the wire lengths leads to substantial improvement in both the latency and the energy, thus supporting our thesis statement from the perspective of the CAM components in the high-performance processors.

CHAPTER IV

DATA PROCESSING COMPONENTS

4.1 Overview of This Chapter

In Chapter 2 and Chapter 3, we provided evidence to support our thesis from the perspective of the SRAM components and the CAM components respectively. After supporting our thesis statement from the perspective of the SRAM components and the CAM components, in this chapter, we focus on the planar and the 3D-integrated designs of the data processing circuits such as arithmetic and logic units. This chapter presents our designs and results for the data processing components in the high-performance processors.

In particular, we study several arithmetic circuits such as adders, shifters and multipliers implemented as 2-die-stacks and 4-die-stacks. These arithmetic units represent a broad spectrum of logic-dominated versus wire-dominated structures and allow us to compare and contrast the benefits of the 3D-integration technology for different circuits.

Modern microprocessors execute multiple parallel arithmetic operations in a clock cycle to exploit instruction level parallelism in an application. For example, some of the Intel processors based on the Core microarchitecture are capable of issuing and executing up to four instructions at a time and are reported to have about eighteen functional units [63, 62, 126]. The latencies of the arithmetic units and the number of simultaneously executing arithmetic units (i.e., issue-width) have a significant influence on the performance of the processor. Thus, designs of arithmetic units such as adders, multipliers, and shifters are critical in deciding the overall performance of high-performance microprocessors.

In this chapter, we evaluate the latency and power benefits of both 2-die-stacked and 4-die-stacked arithmetic units. We demonstrate that the benefits of the 3D-integration technology are the largest when applied to wire-dominated circuits such as the shifter, and large circuits such as the multiplier. We investigate the scalability of the 3D die-stacked arithmetic units by comparing their performance with the corresponding planar circuits. We explore the behavior of the 3D-integrated

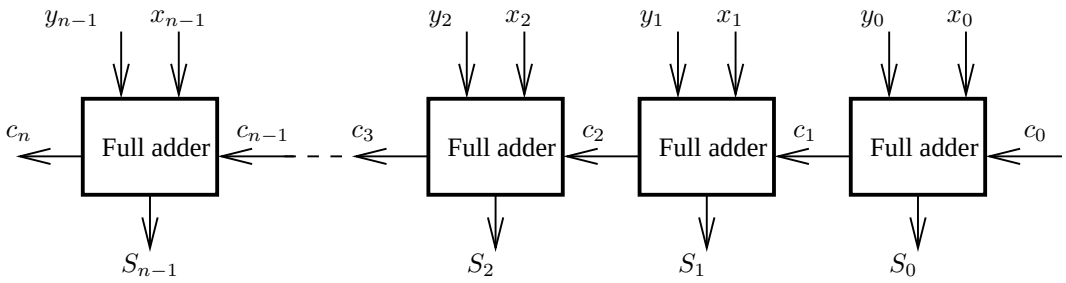

Figure 33: Critical path of an n-bit planar ripple-carry adder

arithmetic circuits with issue-width (number of simultaneously executing functional units in a superscalar processor), transistor sizing, and circuit operating temperature. We demonstrate that the 3D-integrated circuits have better scalability than the planar circuits.

The rest of the chapter is organized as follows: Section 4.2 describes the conventional planar data processing circuits. Section 4.3 details our implementations of the 3D-integrated data processing circuits. Section 4.4 provides the results and analysis. Section 4.5 summarizes our contributions and provides some conclusions.

4.2 Planar Data Processing Components

The key to achieving high performance in data processing components is to identify and optimize the critical (longest delay) paths in the circuits. For example, in an adder circuit, the critical path is through the carry chain. In this work, we evaluate three types of high-performance functional units: adders, shifters, and multipliers. In particular, we consider the Brent-Kung (BK) adder [21], the Sklansky (SK) adder [137] and the Kogge-Stone (KS) adder [74]. We consider a classic barrel shifter and a carry-save array based multiplier [117]. The comparison of circuits that span a broad spectrum of wire-delay characteristics helps to illustrate the relative benefits of the 3D-integration for different circuits. Our analysis also provides insights into the other components in the high-performance processor that might benefit from the 3D-integrated designs.

4.2.1 Adder Circuits

Adder circuits form the core of many operations such as addition, subtraction, multiplication, division, and address generation. Hence, enhancing the performance of the adder is of great interest.

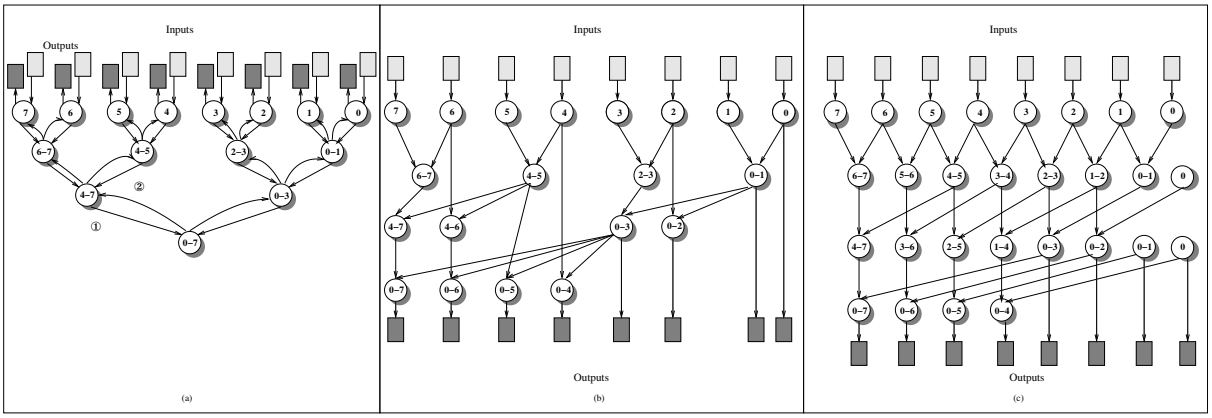

Figure 34: 8-bit planar adders (a) Brent-Kung, (b) Sklansky, and (c) Kogge-Stone. The nodes $\circ$ represent the propagate-generate PG components of the parallel-prefix computation for the adder's carry logic, while the wires communicate the different partial-prefix computations between nodes.

Figure 33 shows the carry chain of a n-bit ripple carry adder. The ripple-carry adder implemented using full-adder cells is usually the smallest and slowest adder. The critical path through the ripple-carry adder circuit is dominated by the logic delay rather than the wire delay. One of the serious drawbacks of the ripple-carry adder is that the delay of the carry chain increases linearly with the operand width, making adders operating on larger bit-widths to be ineffective. Quite a few classic fast adders such as the carry-skip adder, the carry-select adder, and the carry-lookahead adder [117] have been proposed in the past to overcome the carry-chain limitation of the ripple-carry adders. Each of them represents a unique area-time trade-off in the design space. Ladner et al. [78] showed that the carry-propagation in binary addition is a prefix problem and can be calculated using prefix structures. Besides the straightforward serial-prefix structure (implemented by the ripple-carry adder) many different parallel-prefix structures exist, which speed up carry-propagation at the cost of increased area requirements. Identifying the area-delay trade-offs of the parallel-prefix adders is an interesting problem that has received much research attention.

In the parallel-prefix adders, addition is carried out in three steps, namely pre-processing, carry generation and post-processing. In the pre-processing step, special signals called generate G and propagate P signals are extracted from the inputs A and B as shown in Equation 1.

$$\begin{aligned} P_i &= A_i + B_i \\ G_i &= A_i B_i \end{aligned} \tag{1}$$

In the carry generation step, the generate and the propagate signals are used to compute multiple carry bits simultaneously. The carry bits are computed in multiple stages by recursively combining the generate G and the propagate P bits as shown in Equation 2.

$$\begin{aligned} C_i &= G_i + P_i C_{i-1} \\ &= G_i + P_i G_{i-1} + P_i P_{i-1} C_{i-2} \\ &\vdots \\ &= G_i + P_i G_{i-1} + \ldots + P_i \ldots P_0 C_{in} \end{aligned} \tag{2}$$

In the post-processing step, the carry bits are used to compute the final sum bits as shown in Equation 3.

$$S_i = P_i + C_i \tag{3}$$

Different parallel-prefix adders differ in the ways the generate and the propagate bits are combined in the carry generation step. In terms of hardware implementation, the different parallel-prefix algorithms differ from each other in terms of depth (circuit speed), size (circuit area), and maximum fanout, which can be bounded (constant) or unbounded (dependent on the operand width). We chose three parallel-prefix adders with varying logic requirements and speed characteristics as compared to the ripple-carry adder. Table 9 summarizes the characteristics of the serial-prefix (Ripple Carry) and the parallel-prefix adder designs.

Figure 34 illustrates the carry generation logic for each of the adders. While Figure 34 shows 8-bit versions of the adder circuits, all of our results explore larger bit-width adders such as 32-, 64-, and 128-bit adders. Figure 34(a) shows the carry generation tree of an 8-bit Brent-Kung adder. Brent-Kung (BK) adders share processing nodes and require propagation twice through the tree:

Table 9: Characteristics of planar adder designs.

Adder	Area	Speed	Maximum fanout
Ripple-Carry (RC)	lowest	lowest	bounded
Brent-Kung (BK)	medium	medium	bounded
Sklansky (SK)	high	highest	unbounded
Kogge-Stone (KS)	highest	highest	bounded

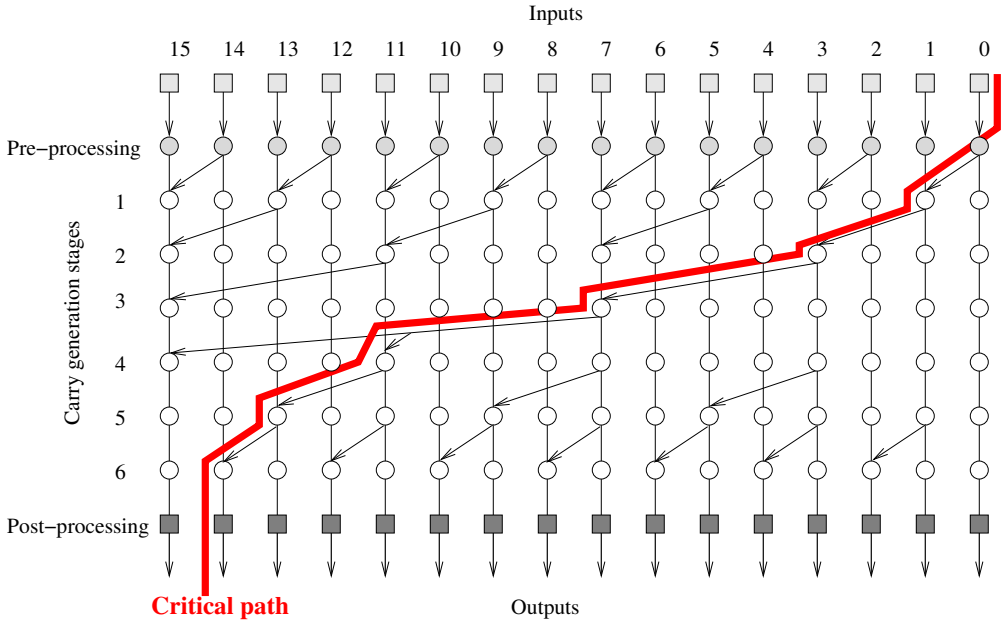

Figure 35: Parallel-prefix graph of a 16-bit planar Brent-Kung adder

first in a downward direction when the prefixes are synthesized (① in Figure 34), and then in an upward direction (② in Figure 34) when the carries are generated. The Brent-Kung parallel-prefix adder provides a regular and area-efficient design [21]. Figure 35 shows the parallel-prefix graph of a 16-bit Brent-Kung adder and highlights the critical path. Note that the carry generation requires traversal through six stages on the critical path.

Figure 34(b) shows an 8-bit Sklansky (SK) adder. The SK adder reduces the number of logic levels that need to be traversed at the cost of more logic fanout and wiring. Each successive level of the SK adder has exponentially increasing fanout and uses the parallel-prefix structure first proposed for conditional-sum adders [137]. As shown in Figure 34(b), the Sklansky adder has a parallel-prefix structure of minimal depth and therefore is among the fastest adder architectures. Its unbounded-fanout (dependent on the operand width) property helps reduce circuit area (fewer prefix nodes) but adds extra delay for driving the high-fanout nodes. Figure 36 shows the parallel-prefix graph of a

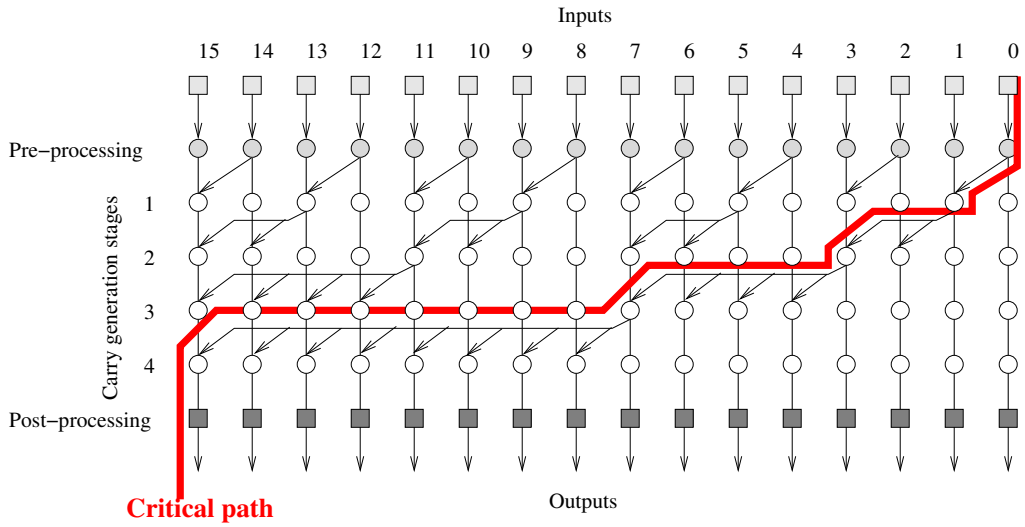

Figure 36: Parallel-prefix graph of a 16-bit planar Sklansky adder

16-bit Sklansky adder and highlights the critical path through four carry generation stages.

Figure 34(c) shows an 8-bit Kogge-Stone (KS) adder. The KS adder reduces the loading on the critical path by using additional logic to limit the fanout per processing node to only two. This in turn decreases the overall circuit latency while increasing the area due to the logic overhead and the wiring overhead. Similar to the Sklansky adder, the Kogge-Stone adder also has a minimal depth parallel-prefix structure. Its bounded-fanout property eliminates the need for driving high-fanout nodes, making it one of the fastest adders in most technologies, but at the cost of more area (more prefix nodes) and wiring. Figure 37 shows the parallel-prefix graph of a 16-bit Kogge-Stone adder and highlights the critical path through four carry generation stages. Note the successive increasing of the wire lengths on the critical path at the deeper stages. This effect is even more pronounced in higher bit-widths such as 32- and 64-bit adders since N-bit adders have lgN stages.

4.2.2 Barrel Shifter Circuit

The barrel shifter enables shifting the bit positions of an input N-bit number by any number of positions up to N. Besides a shift operation being a part of most instruction set architectures, shifting is also used in floating-point arithmetic to align exponents and fractions.

The barrel shifter is a wire-dominated structure since any input bit has to be capable of being routed to any of the N positions within one or two clock cycles. The barrel shifter accomplishes the shifting operation in stages. Each stage performs a conditional shift of magnitude equal to the

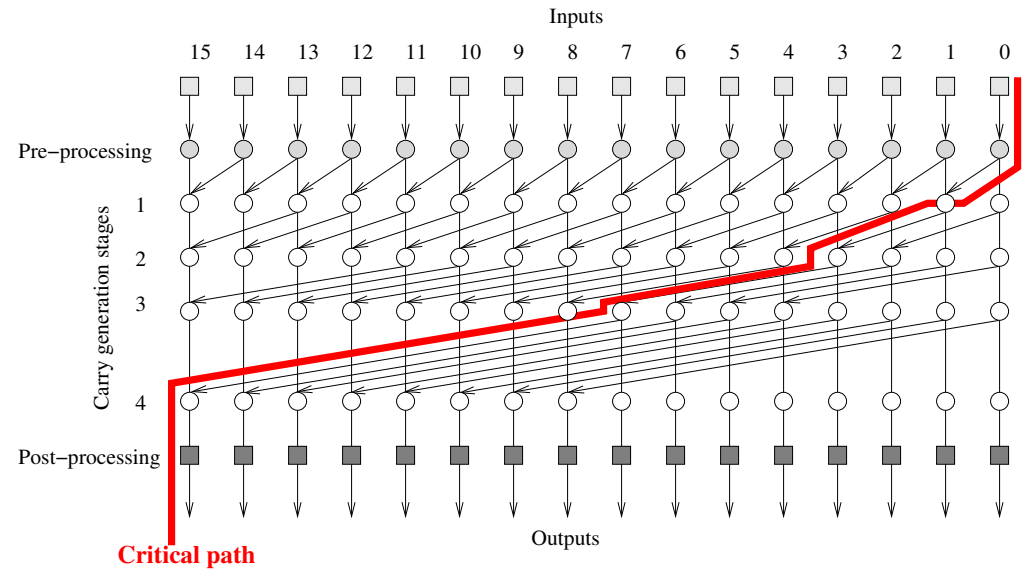

Figure 37: Parallel-prefix graph of a 16-bit planar Kogge-Stone adder

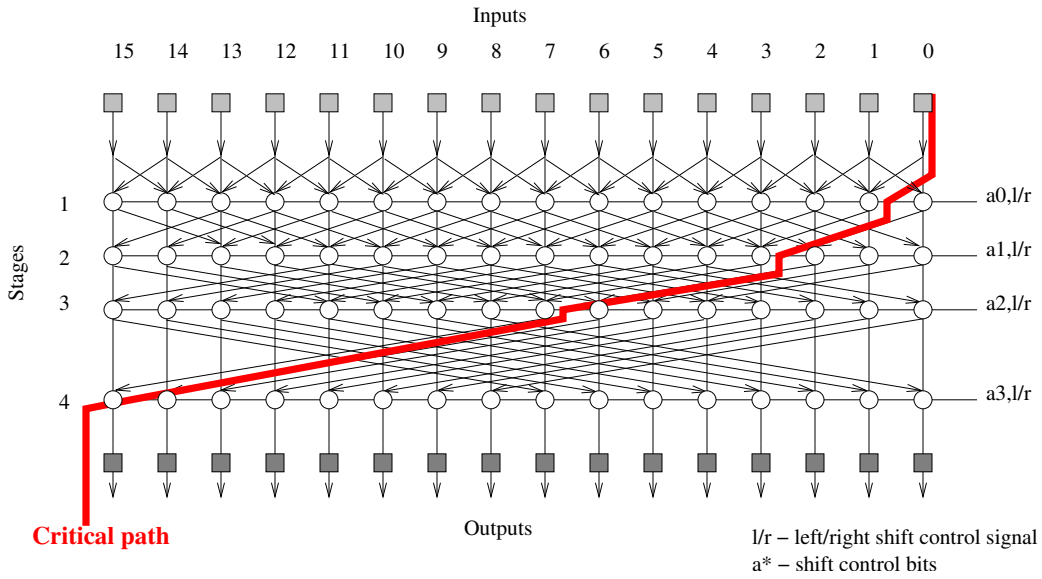

Figure 38: Planar implementation of a 16-bit barrel shifter

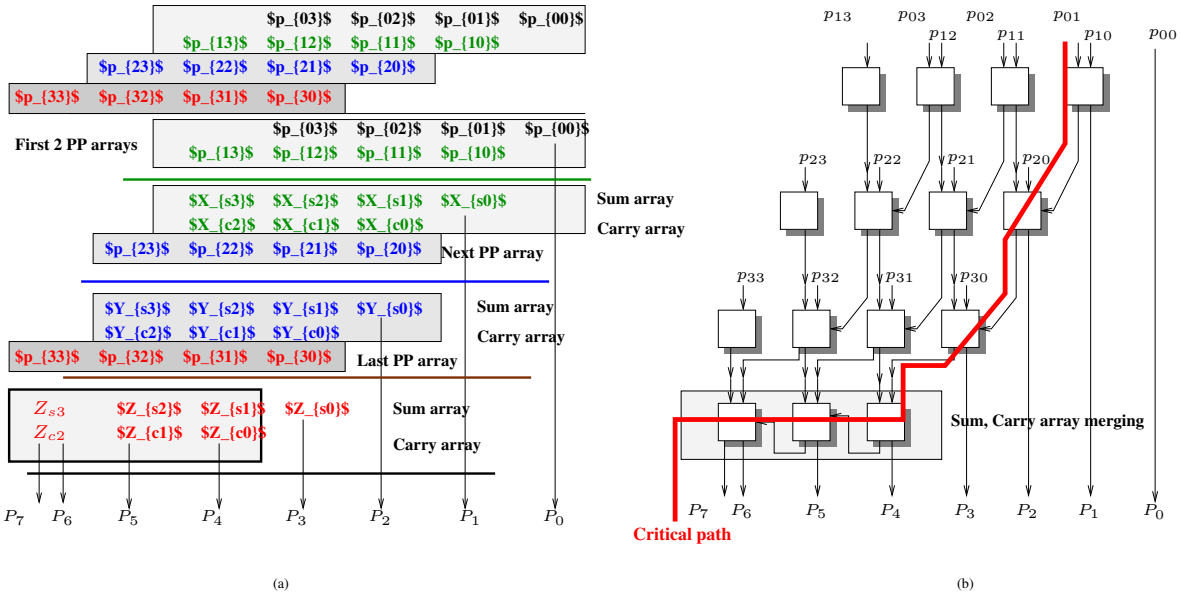

Figure 39: Carry-save array (CSA) (a) multiplier algorithm (b) design (critical path highlighted).

corresponding power of two. For example, the barrel shifter in Figure 38 has four stages; the first stage shifts the input by one bit position, the second stage shifts the value from the first stage by two bit positions, the third stage shifts the value from the second stage by four bit positions, and the last stage shifts the value from the third stage by eight bit positions. The shifter is designed to perform either right shifts or left shifts based on a control input. The worst-case wire length for each successive stage doubles, as illustrated by the critical path in Figure 38. Furthermore, the height of each successive stage also increases exponentially as more wiring tracks are required to route the signals to the appropriate stage inputs. The increase in wire length in both X- and Y-directions leads to a significant wire-related latency and power at the deeper stages of the shifter.

4.2.3 Multiplier Circuit

Multiplication is an important operation in the datapaths of modern processors. In addition to conventional multiplication, some modern instruction set architectures such as the IBM PowerPC [59] and the Intel X86 [61] provide support for a multiply-accumulate operation. The most common multiplier architectures used in high-speed datapaths belong to the class of array multipliers [2, 92]. Some of the challenges of the array multipliers are their high power consumption [92] and large die

area. An array multiplier may operate on either signed or unsigned numbers and may have either non-recoded or recoded structure [16, 42].

An array multiplier typically has dense-logic as well as dense-wiring, thus making it an interesting design point for the 3D implementation. Note that the dense-logic, dense-wire characteristics of the array multiplier is in direct contrast to the logic-dominated adder circuit and the wire-dominated shifter circuit. Thus, the underlying benefits and trade-offs might be different for the array multiplier than the adder and the shifter circuits [112, 114]. Our array multiplier uses a well-known multiplier algorithm called the carry-save algorithm. Figure 39(a) shows a 4×4 carry-save array (CSA) algorithm implementation. In the carry-save technique, the carry information from adding the rows of partial products is not combined with the sum information until the very last step. Note that the sum arrays and the carry arrays are separately propagated until the end, where a carry-completing adder (e.g., carry-lookahead adder) merges the final sum array and the carry array to produce the final product bits. Figure 39(b) shows the array multiplier circuit and highlights the critical path. The critical path is dominated by propagation between the different rows and a rippling component of carry in the last row.

Figure 40 shows the nodes of a 16-bit planar carry-save array multiplier and highlights the critical path. Compared to the size of the adders and the barrel shifter, the multiplier circuit has a large area as well as uniformly dense wires. Note that the critical path has uniform wirelengths between successive logic nodes unlike the adders and the shifter, which have successively increasing wirelengths with deeper stages on the critical path.

4.3 3D-Integrated Data Processing Components

We propose designs of the 3D-integrated circuits to reduce the critical wirelengths among the processing nodes. By vertically stacking nodes on two (four) die, we reduce the area footprint of the designs to approximately half (quarter) of the original size.

There are a variety of ways in which the planar critical paths can be partitioned in the 3D-integration technology. Figure 41(a) shows an 8-bit planar arithmetic circuit with two inputs X_{0-7}

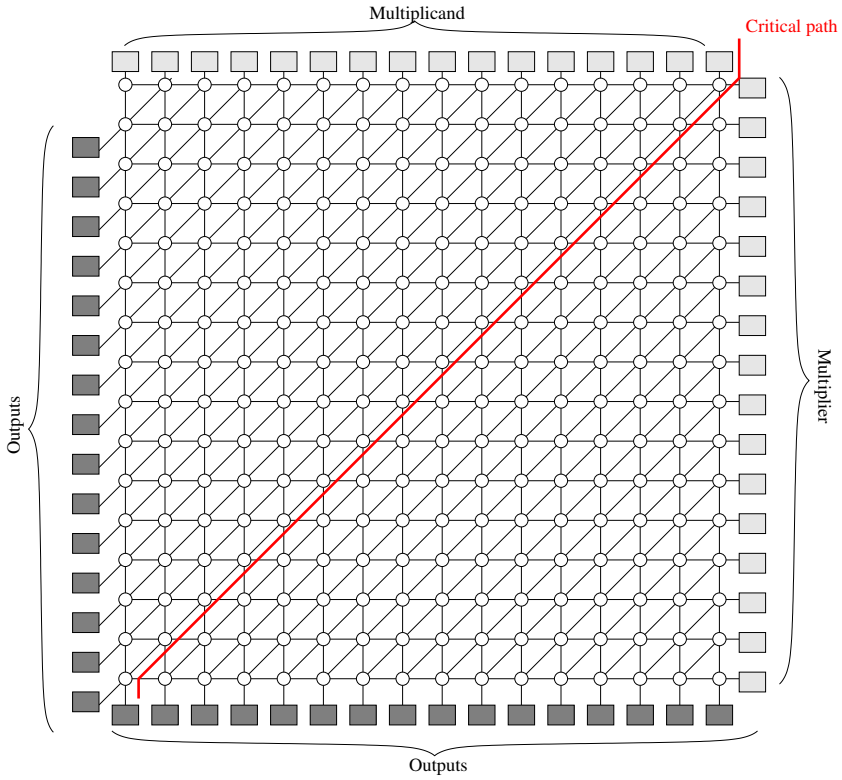

Figure 40: Planar graph of a 16-bit carry-save array multiplier

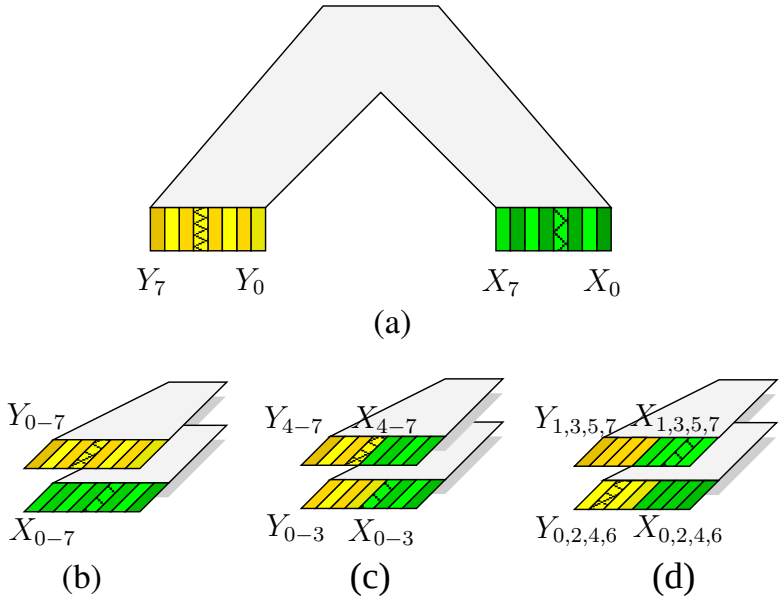

Figure 41: (a) An 8-bit planar adder circuit (b) input-partitioned (c) significance-partitioned (d) odd-even partitioned.

and Y_{0-7}. Figure 41(b) shows the 8-bit arithmetic circuit partitioned in the 3D-integration technology such that the input X_{0-7} is on one die and the other input Y_{0-7} is on the other die (input-partitioned 3D). Depending on how the rest of the datapath is 3D-integrated, the input-partitioned 3D designs of arithmetic units may require additional d2d vias to communicate the input values among the stacked die, in addition to the vias on the critical paths of the arithmetic circuits. In Figure 41(c), the bits are significance-partitioned such that the lower order bits X_{0-3} and Y_{0-3} are on one die and the higher order bits X_{4-7} and Y_{4-7} are on the other die (significance-partitioned 3D). In Figure 41(d), the adjacent bits of the planar circuit are placed on different die such that the even bits $X_{0,2,4,6}$ and $Y_{0,2,4,6}$ are on one die and the odd bits of the inputs $X_{1,3,5,7}$ and $Y_{1,3,5,7}$ are on the other die (odd-even partitioned 3D). The odd-even partitioned 3D circuits shown in Figure 41(d) may require more d2d vias than the significance-partitioned 3D circuits shown in Figure 41(c) but may also provide larger benefits by replacing more wires on the critical path with the d2d vias.

We evaluate the significance-partitioned and the odd/even bit-partitioned 3D circuit designs. Both the designs reduce the wire lengths by replacing many of the wires in the circuit with the d2d vias. The shortening of the wires on the critical paths reduce the circuit delays. The wirelength reduction also decreases wire capacitances and hence, reduces the power consumption of the circuit. The significance-partitioned 3D designs of the arithmetic units may not be able to reduce wires as much as the odd-even partitioned designs. However, they may provide a global benefit by reducing the global wires that pass over those blocks, thus providing latency and power benefits for other circuit paths within the processor.

4.3.1 3D-Integrated Adder Circuits

For each of our 3D-integrated adders, we perform 3D-partitioning with either significance-partitioning or odd-even bit partitioning.

Figure 42 shows an 8-bit significance-partitioned 3D BK adder. The most significant and least significant halves of the parallel-prefix tree logic are stacked on top of each other as shown in Figure 42. The significance-partitioning design has a sparse d2d via requirement since it requires only one via at the last stage. Note that the d2d via replaces the longest wire segment on the critical path.

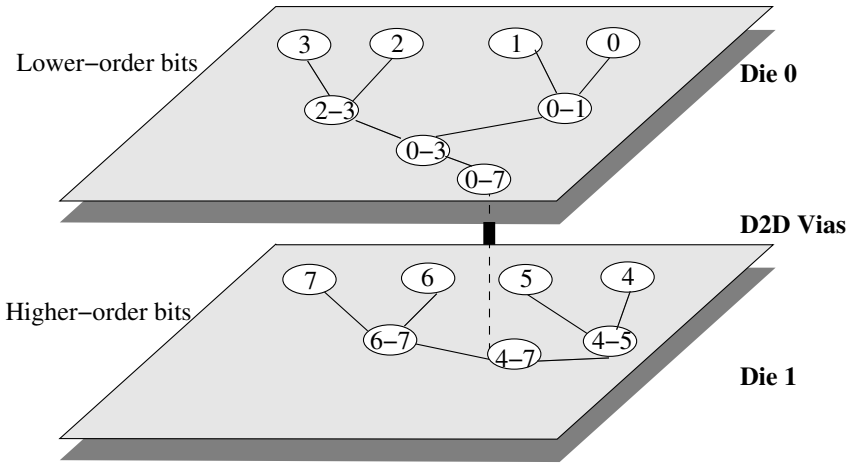

Figure 42: Carry generation graph of an 8-bit significance-partitioned 3D Brent-Kung adder

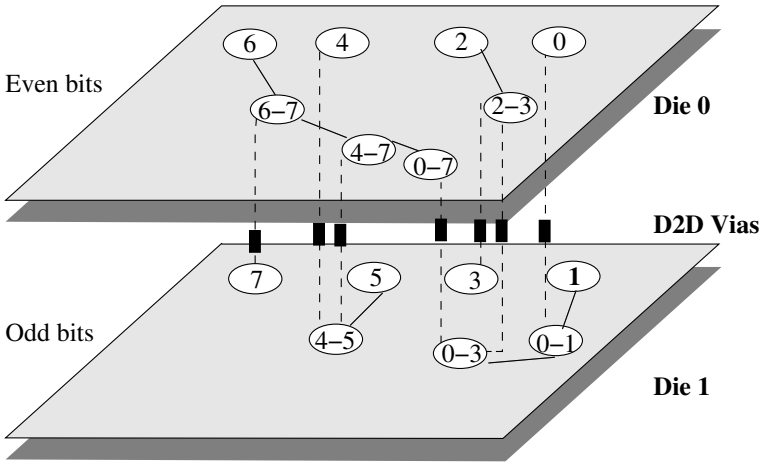

Figure 43: Carry generation graph of an 8-bit odd-even partitioned 3D Brent-Kung adder

Figure 43 shows an 8-bit odd-even partitioned (OEP) 3D BK adder. Every other node of the propagate-generate logic is stacked on top of an adjacent node. This in turn reduces the total width of the circuit by one half, which reduces the length of each inter-node wire by one half as well. While all of the critical wires have been halved in length, some have been replaced by the d2d vias. The additional latency overhead of the d2d via is relatively small and is more than offset by the corresponding wire length reduction. The odd-even partitioned 3D BK adder needs $O(N)$ d2d vias for an N-bit adder, which is easily satisfied by the dense d2d interface of the current 3D-integration technologies.

For the odd-even partitioned (OEP) designs of the Sklansky and the Kogge-Stone adders, we similarly stack adjacent processing nodes which in turn reduces overall footprint and critical wire

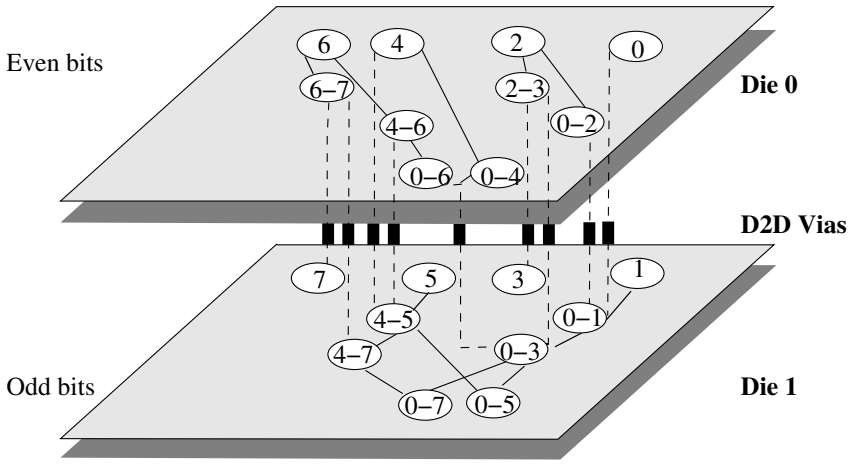

Figure 44: Carry generation graph of an 8-bit odd-even partitioned 3D Sklansky adder

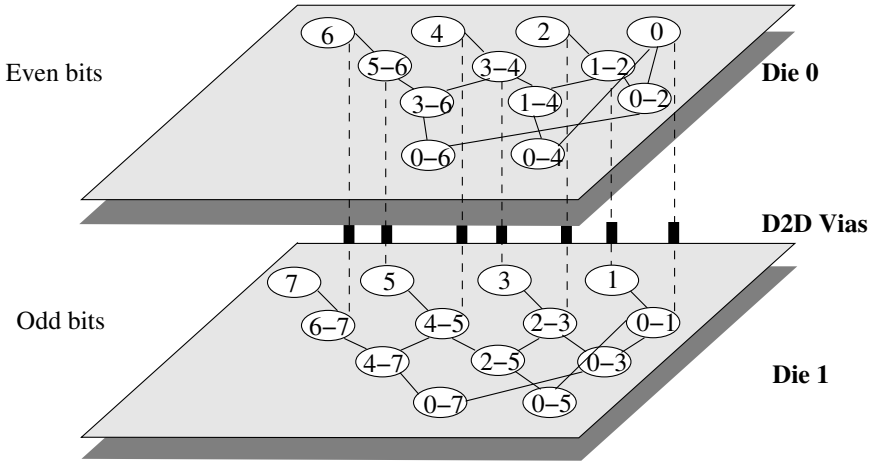

Figure 45: Carry generation graph of an 8-bit odd-even partitioned 3D Kogge-Stone adder

lengths by one half. Figure 44 shows an 8-bit SK adder in OEP-3D and Figure 45 shows an 8-bit KS adder in OEP-3D. The OEP-3D SK adder requires $O(N \lg N)$ d2d vias since the vias are required at each of the $\lg N$ levels of the adder. The OEP-3D SK adder requires the most d2d vias of the three adders we consider. The OEP-3D KS adder requires $O(N)$ d2d vias only for the first level of the propagate-generate logic. Since the d2d vias are required only in the first level, they can be positioned adjacent to each other along a row. Such regular positioning of the d2d vias may have benefits in the floorplanning of the 3D-integrated circuits.

4.3.2 3D-Integrated Barrel Shifter Circuit

Similar to the adders, we may 3D-integrate the barrel shifter in either significance-partitioned or odd-even partitioned designs. In odd-even partitioning, the adjacent nodes are vertically stacked on

different die, thus shrinking the circuit to half its original dimension in each direction. A partitioning of the odd and the even bit positions on the two die reduces the lengths of wires in every successive multiplexing stage, which in turn provides greater latency and power savings with every additional stage of the circuit.

4.3.3 3D-Integrated Multiplier Circuit

We partition the planar multiplier such that the alternate rows of the array are on different die. Figure 46(a) shows the processing nodes of a planar 4×4 multiplier array. The nodes have been shaded differently for identification purposes. Figure 46(b) shows our proposed design of the 3D-integrated CSA multiplier. Note that the 3D-integrated multiplier is partitioned differently than the 3D-integrated adders and shifter. The 3D-integrated multiplier has alternate rows on different die while the 3D-integrated adders and shifter have alternate bit-columns on different die. We observed that the row partitioning increases the savings on the multiplier critical path. Hence we preferred row partitioning over column partitioning for the multiplier circuit. The 3D-integrated CSA multiplier replaces wires on the critical path with the d2d vias as shown in Figure 46, thus reducing the circuit latency and power.

4.3.4 Extending to More Than Two Die

Note that the 2-die-stacked designs can be naturally extended as stacks of more than two die. In this section, we discuss 4-die-stacked designs without loss of generality. A 64-bit arithmetic unit can be significance-partitioned as thirty-two bits per die in the 2-die-stacked design, and as sixteen bits per die in the 4-die-stacked design. With 4-die-stacking, splitting the alternate bits on adjacent die makes the design to be modulo-4 partitioned (M4P).

Figure 47 shows a significance-partitioned radix-4 BK adder implemented as a 4-die stack. Note that the bits are significance-partitioned such that the lower-order bits are on the top die and the higher-order bits are on the bottom die. With 4-die stacks, the advantage of the modulo-4 partitioned (M4P) 3D-integrated designs over the significance partitioned 3D designs may not be large due to the higher backside via overhead of the modulo-4 partitioned designs.

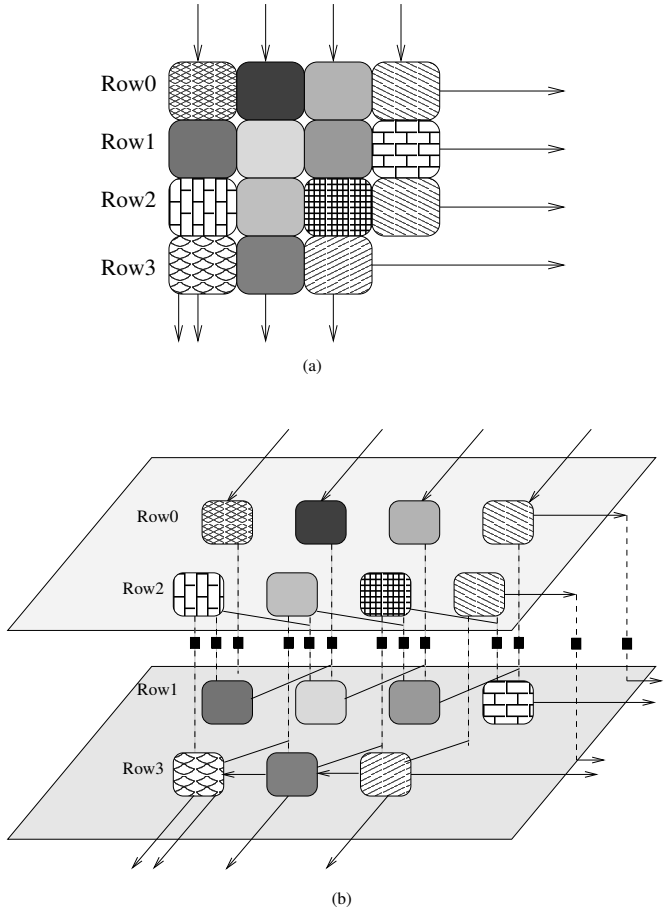

Figure 46: (a) Processing nodes and input/output ports of a planar 4×4 multiplier array (wires within the array not shown for clarity) (b) 4×4 3D-integrated carry-save array (CSA) multiplier with the rows partitioned on different die.

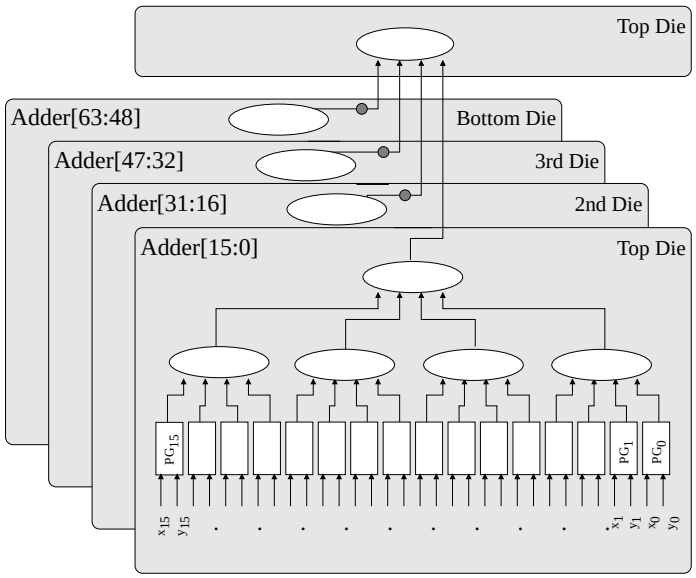

Figure 47: Significance partitioned 3D BK adder using a 4-die stack

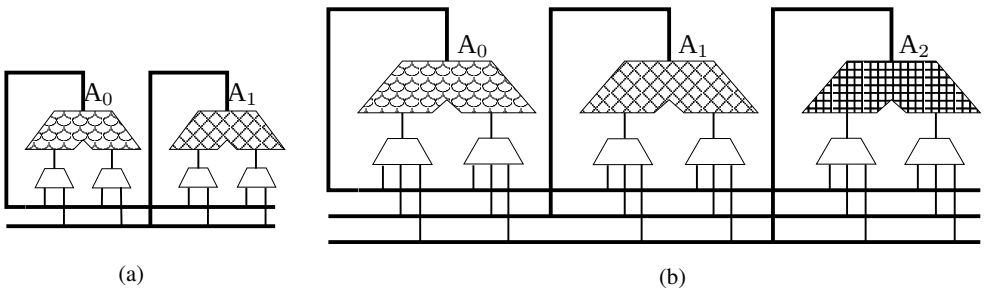

Figure 48: Bypass wiring complexity (a) Issue-width IW = 2 (b) IW = 3

4.3.5 Scalability Studies

In this section, we describe the parameters used in our scalability studies, namely issue width, transistor sizing and operating temperatures.

4.3.5.1 Issue-width

Large issue widths enable microprocessors to execute more arithmetic operations simultaneously. Unfortunately, increasing the issue-width causes the wire complexity and the area of the microprocessor to increase. Typically, an arithmetic operation (such as addition) requires at least two source operands. The source operands of a particular arithmetic operation can themselves be the results generated by other arithmetic operations. If an instruction C has one of its source operands generated by instruction A and another source operand generated by instruction B, then C is said to be dependent on A and B. The results generated by A and B may be bypassed to the arithmetic unit executing C to speed up the execution. As the issue-width IW of the microprocessor increases, a given source operand for a given arithmetic operation can be generated by any of the IW arithmetic units. The circuit required to bypass the results from each arithmetic unit to all the arithmetic units quickly dominates the delay. Figure 48 shows the increasing wire complexity and the increasing area when the issue width IW is increased from two (A_0 and A_1) to three (A_0, A_1, and A_2). This rapid increase in the wiring complexity and the area of the bypass circuit increases the latencies of the individual arithmetic units, since each individual arithmetic unit has to drive longer wires both within the unit and in between the units.

Given that technology scaling has created an ever widening gap between the relative delay of logic and wires [64, 18], increasing the issue-width may increase the wire delay of the bypass

network, thus reducing the benefits of multiple arithmetic units on the overall performance of the processor.

4.3.5.2 Transistor Sizing

Traditionally, arithmetic unit designers identify one or more critical paths through the arithmetic circuits and speed up the critical paths with innovative designs and/or transistor sizing [74, 2, 92]. Transistor sizing involves enlarging (or reducing) the width of the channel of a transistor. When the width of the channel is increased, the current drive capability of the transistor increases, thus reducing the delay of the output. The undesirable effect of increasing transistor sizes is the increase in gate capacitances and leakage current. Increased gate capacitances lead to increased switching power. Also, leakage power has become a critical issue that gets worse with every new technology generation [64, 123, 55]. Since leakage power is sensitive to temperature and vice versa, increasing leakage power also increases temperatures. This creates a leakage-temperature feedback loop that might lead to thermal runaways. In summary, transistor sizing for performance may lead to increased power consumption and worsening power density issues. Hence, transistor sizing has to be balanced with power budget and thermal considerations. In this exploration, we compare the latency and the energy profiles of the 3D-integrated circuits with those of the planar circuits for different transistor widths.

4.3.5.3 Operating Temperature

When the circuits operate at higher temperatures, they experience a latency degradation due to reduced carrier mobility and increased wire resistances. We study the impact of increased operating temperature on the latency degradation of the planar and the 3D circuits. We explore whether the 3D-integrated circuits experience a worse latency degradation than the planar circuits due to their higher thermal profiles.

4.4 Results

4.4.1 3D-Integrated Adder Benefits

Table 10 shows the latencies and energy consumptions for the planar and the 2-die-stacked 3D adder circuits, without including the bypass wiring complexity. The top portion of Table 10 shows

Table 10: Latency and energy benefits of the 2-die-stacked 3D circuits compared to the planar circuits. The columns marked '%' show the relative reduction in latency and energy. Note that these results are for stand-alone functional units and do not take bypass wiring into account. (The best 64-bit 3D-integrated configurations are shown in bold font).

	Planar Latency (ps)			Significance partitioned			Odd-even partitioned		
Bits	BK	SK	KS	Latency %					
32	333.82	266.12	193.40	0.01	3.58	4.35	2.21	4.39	5.25
64	392.38	440.84	238.74	0.26	4.12	**5.48**	0.90	4.85	**7.37**
128	449.33	784.59	299.33	0.03	4.51	9.47	0.44	6.05	13.05
	Planar Energy (pJ)			Significance partitioned			Odd-even partitioned		
Bits	BK	SK	KS	Energy %					
32	4.00	3.21	4.49	-0.26	5.50	5.27	1.95	7.40	5.78
64	9.49	9.90	11.09	0.53	**8.37**	6.13	0.64	**8.31**	7.12
128	22.03	39.37	27.32	0.81	12.05	9.21	0.44	14.64	11.22

the planar latencies and the percent latency benefits of the 3D-integrated adder circuits. The bottom portion of Table 10 shows the planar energy consumption and the percent energy benefits of the 3D-integrated adder circuits. Since the adders are logic-dominated structures, the benefits are modest.

Overall, the odd-even partitioned circuits provide larger latency benefits than the significance-partitioned circuits. This is expected since the odd-even partitioned circuits reduce more wires than the significance-partitioned circuits. With increasing bit widths, the number of levels in the carry generation tree and the overall size of the circuit also increase, hence increasing the wires used by the design. As we see from Table 10, with increasing bit-widths, the relative latency benefit increases, thus illustrating that the 3D-integration technology can effectively target wire delay.

Figure 49 shows the percent latency distributions of the planar and the odd-even partitioned 3D implementations of the 64-bit versions of the three adders. The BK adder, being the least wire-dominated adder, derives little benefit from the 3D-integration. The corresponding stages in both the planar and the 3D circuits have similar contributions to the overall delay, thus making the overall savings of the 3D BK adder to be small. The 3D SK adder provides larger benefit than the 3D BK adder. In Figure 49, the later stages of the 3D SK adder have gradually increasing latency savings. Although the SK adder has longer wires at the later stages of the carry generation tree, it also has increasing logic complexity due to the exponentially increasing fan-outs of the later stages. The 3D KS adder provides the highest benefit since it has a higher wire-complexity as well as bounded fan-out.

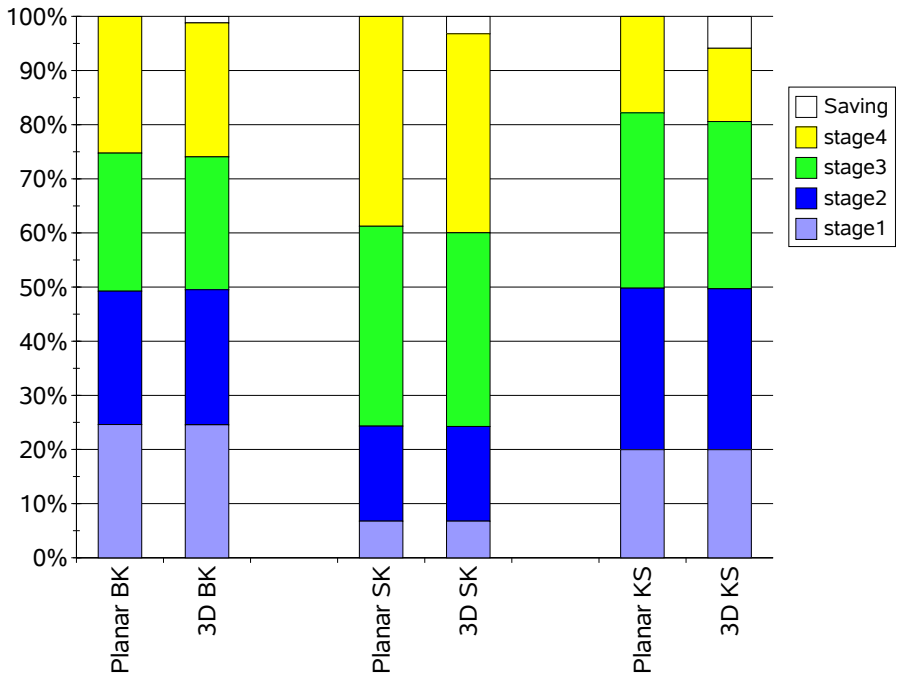

Figure 49: Latency distributions and savings for various 64-bit adders.

Note that the 3D-integrated circuits provide simultaneous benefits in both latency and energy. The extent of simultaneous benefits provided by the 3D-integrated circuits depend on the wire-length characteristics of the planar design. The BK adder, being less influenced by the 3D-integration, derives less energy benefit from the 3D implementation. Although the 3D KS adder provides the largest benefit in terms of latency, it does not provide as large a benefit as the 3D SK adder in terms of energy. The overall energy reduction in the KS adder is less than the SK adder because the large amount of logic duplication in the KS adder reduces the relative power contribution of the wires. This demonstrates that different 3D-integrated arithmetic unit designs may be used to optimize for different design objectives such as latency and energy.

Figure 50 shows the latencies of both significance-partitioned (SP) and modulo-4 partitioned (M4P) 4-die-stacked 3D adder implementations. Note that the latency benefits of the modulo-4 partitioned 3D circuits are comparable to those of the significance partitioned 3D circuits. With 4-die implementations, the d2d vias are required to be etched through the backside and hence incur area penalty. Due to the larger d2d via overhead, 4-die-stacked M4P 3D designs provide similar latency benefits as the 4-die-stacked SP 3D designs.

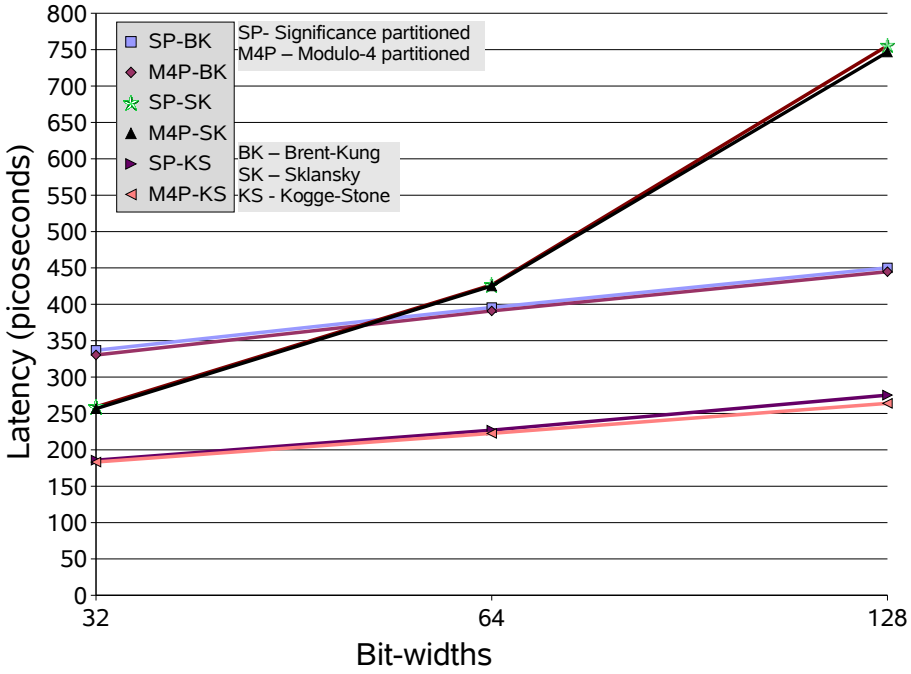

Figure 50: Latency for 4-die-stacked 3D implementations of various adders.

Table 11: Latency and energy benefits of the 3D-integrated shifter compared to the planar shifter circuit. (results do not take bypass wiring into account).

Bits	Latency			Energy		
	Planar (ps)	3D (ps)	Benefit %	Planar (pJ)	3D (pJ)	Benefit %
32	256.65	240.45	6.31	2.87	2.74	4.71
64	328.33	291.79	**11.13**	7.12	6.55	**7.98**
128	452.68	362.26	19.97	18.25	15.81	13.37

4.4.2 3D-Integrated Barrel Shifter Benefits

The results for the 3D barrel shifter in Table 11 demonstrate that the 3D-integration can provide even greater benefits when the wire-delay component dominates the overall circuit delay. The 64-bit 3D-integrated barrel shifter exhibits a 11% latency improvement with a simultaneous 8% benefit in energy. We also evaluated 4-die-stacked 3D implementations of the barrel shifter. We found the 64-bit, 4-die-stacked 3D implementation of the barrel shifter to provide 16% latency improvement with a simultaneous 12% benefit in energy.

Table 12: Latency and energy benefits of the 3D-integrated multiplier compared to the planar multiplier circuit. (results do not take bypass wiring into account).

Bits	Latency			Energy		
	Planar (ps)	3D (ps)	Benefit %	Planar (pJ)	3D (pJ)	Benefit %
32	986.08	840.09	14.81	4.50	3.98	11.72
64	2234.00	1654.50	**25.94**	12.41	9.85	**20.64**
128	5721.10	3278.10	42.70	36.03	23.51	34.73

4.4.3 3D-Integrated Multiplier Benefits

Table 12 shows the benefits of the 3D-integrated carry-save array (CSA) multiplier compared to the planar CSA multiplier. The benefits of the 3D-integrated CSA multiplier are larger than either the 3D-integrated adders or the 3D-integrated shifter due to the large size of the multipliers resulting in long wirelengths on the critical path. For example, a N-bit multiplier has $O(N)$ wires on the critical path as compared to a N-bit barrel shifter that has $O(lgN)$ wires on the critical path. Replacing each of the wires on the critical path with d2d vias provides a large overall benefit. A 64-bit 3D-integrated multiplier provides a 26% latency improvement with a simultaneous 21% benefit in energy. Even though the CSA multiplier does not have successively increasing wire segments, it derives a large benefit due to the large number of wires getting replaced by the d2d vias. We have already seen in Chapter 2 that the 3D-integration offers significant benefits in implementing large components such as on-chip caches, register files, and branch-prediction tables. The benefits provided by the 3D-integrated multipliers offer further proof that the 3D-integration effectively addresses the wire-delay issues in large components.

4.4.4 Scalability Results

We explore the scalability of the planar and the 3D-integrated circuits with increasing issue widths, transistor sizing, and temperature.

4.4.4.1 Issue-width

Figure 51 shows the latency impact on the planar KS adders due to increasing issue-widths. Note that with larger bit-widths, the rate of degradation of latency is also higher. With larger bit-widths, the overall area of the circuit increases, thus increasing the wirelengths. We observed a similar trend

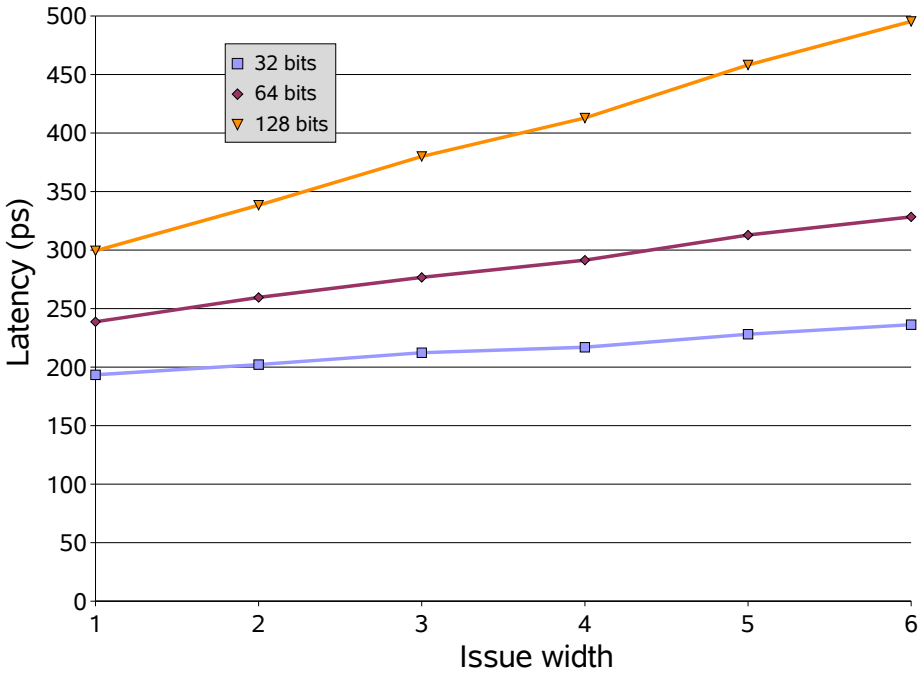

Figure 51: Latency versus issue width for the 64-bit planar KS adder.

across all the arithmetic units that we analyzed.

Table 13 shows the percent delay improvements of various 2-die-stacked OEP-3D arithmetic circuits (64-bits) for increasing issue-widths. As issue-widths increase, the shifter, being already a wire dominated circuit with successively increasing wire delays at deeper levels, is impacted greatly by the additional area required by the bypass circuit. Hence, with increasing issue-widths, the shifter derives the largest benefit from 3D-integration. The adder and the multiplier circuits suffer less wire-increases due to the increasing issue-widths, and hence derive less benefit than the shifter. Note that the multiplier circuits are larger than the adder circuits (longer wires) and hence provide larger benefits. Thus the benefits of the 3D technology are dependent on the logic- and wire-densities of the original designs. As the issue-width increases to six, the latency benefits raise to $\sim$48% for the barrel shifter.

Figure 52 shows the comparison of energy consumption for the 64-bit planar KS adder circuit and the 64-bit 3D-integrated KS adder circuit with increasing issue-widths. Note that the planar circuits have a much higher increase in the total energy consumption compared to the 3D-integrated circuits, as the issue-width increases. The planar circuits have a much higher rate of increase in

Table 13: Percent improvement in delays of various 64-bit, 2-die 3D-integrated arithmetic circuits (odd-even partitioned)

Issue width	% KS Adder	% Barrel Shifter	% CSA Multiplier
2	15.43	30.13	26.51
3	18.16	36.23	30.54
4	21.07	40.74	34.61
5	24.63	44.2	37.28
6	25.68	**47.27**	40.61

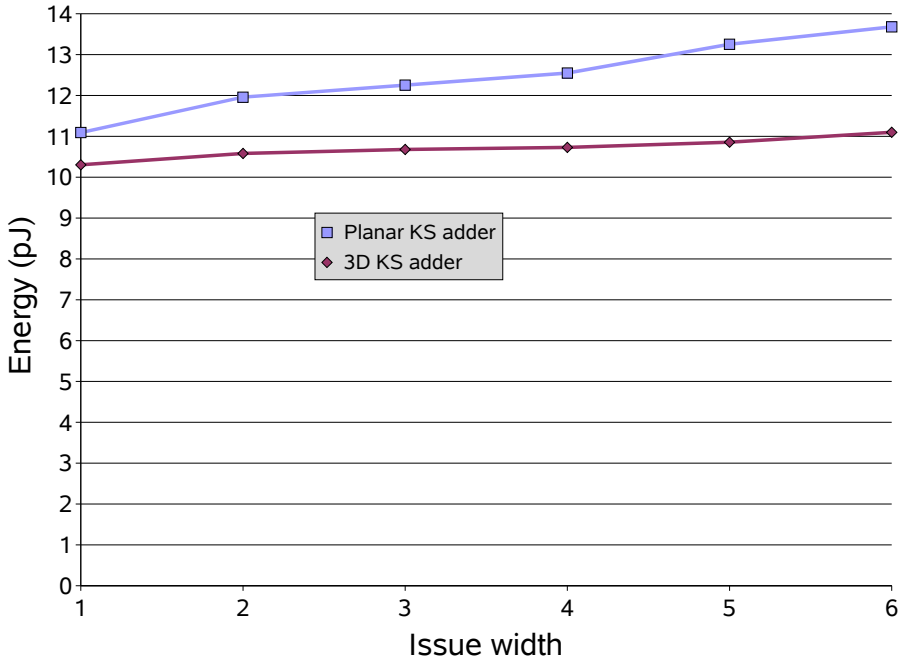

Figure 52: Energy comparison of the 64-bit planar and the 64-bit 3D-integrated KS adder circuits with increasing issue-widths.

delay than the 3D-integrated circuits, thus causing their energy to increase at a faster rate. Thus, the 3D-integrated circuits have better energy scalability with increasing issue-widths than the planar circuits.

4.4.4.2 Transistor Sizing

Figure 53 shows the latency trends of the 64-bit planar KS adder and the 64-bit 3D-integrated KS adder due to transistor sizing in a 4-issue processor configuration. The drive-strengths as well as the gate loads increase with increased transistor sizing. Until the drive-strength reaches the optimal capacity to drive the load, the overall latency decreases for both the planar and the 3D-integrated

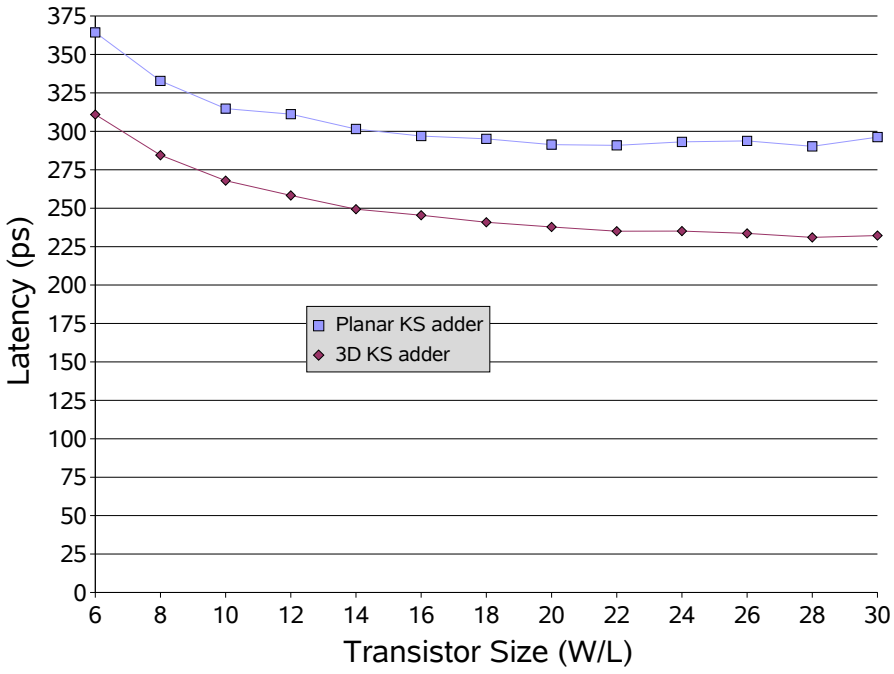

Figure 53: Latency versus transistor sizing for the planar and the 3D-integrated KS adder circuits (64-bit, 4-wide issue).

adders. Once the circuit has achieved sufficient drive-strength, further increases in the transistor sizes do not decrease latency and may even lead to latency degradation due to the increasing gate load. Given that the latency trends are similar for both the planar and the 3D-integrated circuits, we conclude that the same design procedures that are used to size the transistors in the planar circuits can be used for the transistor sizing in the 3D-integrated circuits.

Next, we consider the impact of transistor sizing on the energy consumption of the 64-bit planar KS adder and the 64-bit 3D-integrated KS adder circuits in Figure 54. As the transistor sizes increase, the energy consumption also increases. With increased transistor sizes, switching capacitances increase, thus increasing the energy consumption. Increased transistor sizes also increase the leakage currents. As the transistor sizes increase, the energy consumption of the 3D-integrated adder increases slower than that of the planar adder due to the reduced wires.

For power-conscious designs, the latency benefits of the 3D circuits can potentially be traded to further reduce the power consumption. The circuit designers may be able to replace fast, leaky transistors in the planar designs with slow, less leaky transistors in the 3D-integrated designs and still

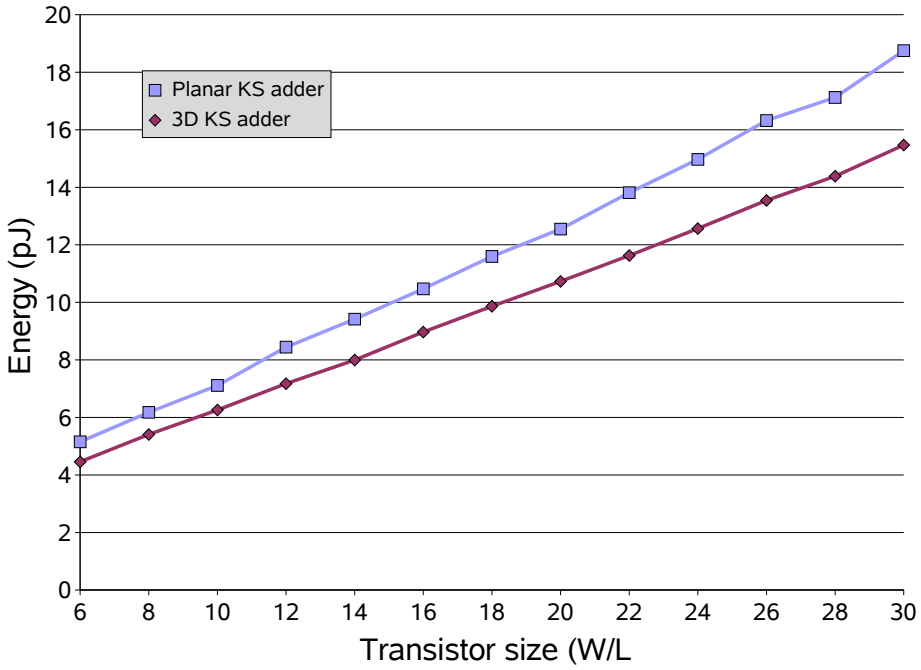

Figure 54: Effect of transistor sizing on the energy consumption of 64-bit, planar and 3D-integrated KS adders.

Table 14: Power benefit due to downsizing the transistors on the 3D-integrated KS adder circuits

Bit-width	Planar latency (ps)	3D latency (ps)	Power benefit (%)
32	216.93	211.18	38.09
64	291.37	284.48	55.87
128	412.80	405.11	71.65

meet the latency target for the overall design. One particular example is to reduce the transistor sizes in the 3D-integrated circuits such that the 3D-integrated circuits operate within the same latency as the corresponding planar circuits, but at a lower power consumption. Table 14 shows the potential power benefits, when the transistors on the 3D KS adder circuits are resized until their latencies are approximately the same as the corresponding planar circuits. The last column in Table 14 shows the percent power saving that can be realized when the 3D-integrated circuits operate at the same latency as the corresponding planar circuits.

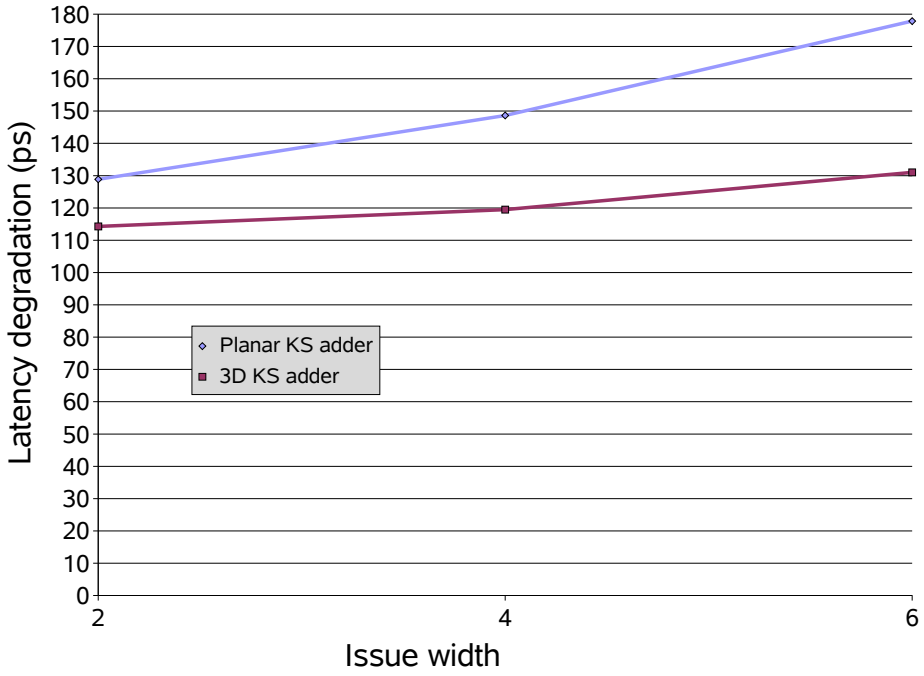

Figure 55: Latency degradation of the planar and the 3D KS adder circuits due to temperature increase from 25C to 100C.

4.4.4.3 Operating Temperature

Next, we look at the latency degradation due to increased operating temperature of the circuits. Figure 55 shows the latency degradation of the planar and the 3D-integrated circuits when the operating temperature increases from 25C to 100C. When the circuits operate at higher temperatures, they experience latency degradation due to reduced carrier mobility and increased wire resistances. Since the 3D-integrated circuits reduce the wires, they experience a lower rate of latency degradation than the planar circuits for increasing operating temperatures and issue-widths.

4.5 Summary of the 3D-Integrated Data Processing Components

In this chapter, we showed that the 3D-integration provides simultaneous reductions in wire lengths, latency, and power in the design of data processing components, thus supporting our thesis statement for the data processing components in the high-performance processors. For wire-dominated circuits such as barrel shifters and large circuits such as array multipliers, the 3D-integration can provide large simultaneous benefits in both performance and power consumption. Logic-dominated

circuits such as adders may not gain as much from 3D implementations, but the 3D-integration may still provide benefits by reducing the bypass wires between the circuits rather than within the circuits. We demonstrated that the 3D-integrated circuits have better scalability than the planar circuits in the face of increasing issue-widths, transistor sizing, and operating temperatures. The relative benefits of the 3D technology will increase in future technology generations, making it a very attractive option for future designs. The superior scalability of the 3D-integrated circuits may play a crucial role in extending the silicon road-map for a few more technology generations.

While the SRAM components, the CAM components and the data processing components account for a large portion of the circuits in modern processors, there are many other miscellaneous components that we have not explicitly modeled in any of the chapters so far. Some of such components are decoder programmable logic arrays (PLAs), microcode read-only-memories (ROMs) in x86 processors, lookup tables for SRT dividers, and dependence-checking logic for register renaming.

Components such as the decoder programmable logic arrays (PLAs), the microcode read-only-memories (ROMs) in x86 processors, and the lookup tables for SRT dividers have underlying array structures and can be classified to have similar circuit profiles as the SRAM components. We believe that these components would observe similar benefits from a 3D implementation as the SRAM components depending on their sizes.

Logic-dominated components such as the dependence-checking logic have similar circuit profiles as the adder circuits and may derive only a moderate benefit from the 3D implementations. However, the 3D implementation of such logic-dominated components may still provide a global benefit even if their own latency and/or power remain unaffected. By reducing the footprints of such components, the lengths of the global wires that pass over those components may be reduced, thus providing latency and power benefits for other circuits. Such floorplan-related wire reduction may be able to remove entire pipeline stages, such as the stages in the Intel NetBurst microarchitecture that are primarily for driving a signal from one part of the chip to another [57].

Part III

"The truth. It is a beautiful and terrible thing, and should therefore be treated with great caution."

– J. K. Rowling, *Harry Potter and the Sorcerer's Stone*, 1997.

3D-Integrated High-Performance Processors

Thesis statement: 3D-integration provides simultaneous performance and power benefits to build high-performance microprocessors, while keeping the worst-case temperature under control.

Previously, in Part II, we studied several processor components and demonstrated the simultaneous performance and power benefits of the 3D-integrated components in support of our thesis statement. Note that the latency/power benefits of the individual processor components may not necessarily translate directly into overall processor performance benefits. Some critical circuits such as the wakeup-select loop [106], branch resolution loop, and load resolution loop [17] may influence the performance more significantly than the other components. In this part of the dissertation, we evaluate the performance and power of 3D-integrated processors using a detailed experimental evaluation procedure.

We demonstrate that the 3D-integrated processors can provide simultaneous performance and power benefits. While the 3D-integration technology provides simultaneous benefits in performance and power, stacking multiple die increases power density. The increased power density may exacerbate existing hotspots and may even create new hotspots [157]. Performance benefits due to the 3D-integration can potentially be nullified due to frequent thermal emergencies. Thermal management is already a significant problem in current planar processors and researchers are focusing efforts on designing temperature-aware microarchitectures [135, 136]. In the previous chapters, we deferred the thermal analysis of the 3D-integrated components since the temperatures depend not only on the power consumption of the component but also on other factors such as the floorplan, power dissipation of the adjacent components, and application-dependent activity factors.

In contrast to Part II of the dissertation, this part (Part III) focuses on the power density and temperature challenges in addition to the performance and power issues of the high-performance

processor designs. We address the thermal challenges in high-performance 3D-integrated processors [115, 113], besides evaluating the simultaneous performance and power benefits at the processor level. We propose microarchitectural techniques [115] to control the worst-case temperature on 3D-integrated processors.

To enable detailed thermal analysis for the planar and our 3D-integrated processors, we create the floorplans for the processors. We collect the activity factors while the processors run various benchmark suites. This performance analysis procedure provides us with relevant data to perform the thermal analysis. While our absolute results depend on our assumptions related to the floorplans, the general trends will hold for other floorplans and microarchitectures.

We conclude this dissertation by demonstrating that the 3D-integrated processors can simultaneously provide both performance and power benefits, while keeping thermals under control.

CHAPTER V

3D-INTEGRATED PROCESSORS

5.1 Overview of This Chapter

We describe the 3D-integrated processors based on a planar high-performance processor and evaluate their performance and power benefits. We perform temperature analysis of the planar and the 3D-integrated processors. Our 3D-integrated processors provide simultaneous performance and power benefits, furnishing further evidence in support of our thesis statement. Our 3D-integrated processors demonstrate two different approaches to improve performance: clock speed improvements (3D-integrated processors with identical microarchitectural configurations as the baseline planar processor run at a higher clock frequency due to wire reduction), and IPC improvements (3D-integrated processors accommodate larger-sized modules than the planar processors for the same clock delay and run at the planar clock frequency) Our 3D-integrated processors demonstrate the simultaneous benefits of the 3D-integration and highlight the power density and thermal issues related to the 3D-integration technology.

The rest of this chapter is organized as follows. Section 5.2 introduces our planar and 3D-integrated processor configurations. Section 5.3 describes our evaluation framework for circuit latency, power, and temperature estimations. Section 5.4 presents our latency, power, and temperature results. Section 5.5 summarizes our results and concludes the 3D-integrated processors.

5.2 Planar and 3D-Integrated Processors

5.2.1 Baseline Planar Processor

We model our planar processor based on the Alpha 21364 [10, 48, 71, 94, 49, 134] processor shown whose details are shown in Figure 56. Figure 56(a) shows a die photograph and the floorplan of the Alpha 21364 processor. The Alpha 21364 processor [49] consists of an Alpha 21264 core flanked by L2 cache on three sides. Figure 56(b) shows the block level details and floorplan of the Alpha 21264 core [71]. Figure 56(c) shows the details of the integer execution core (EBox) of the Alpha 21264 processor. Note that the integer execution core consists of two clusters, namely Cluster 0 and

Table 15: Parameters of our baseline planar processor

Parameter	Value
Instr/Data L1 caches	64 KB
L2 Cache	1 MB
Branch Predictor	4 KB (tournament predictor)
DTLB	128 entries
ITLB	128 entries
Integer Register File	80 entries
FP Register File	80 entries (4r+2w)
Integer RAT	80 entries
FP RAT	80 entries
Integer Scheduler	20 entries
FP Scheduler	16 entries
Load/Store Queue	32/32 entries
Reorder Buffer	80 entries
Integer ALUs	12
FP ALUs	4

Cluster 1. The Alpha 21264 splits the integer register file into two clusters that contain duplicates of the 80-entry register file [71]. The Alpha architects used the clustered approach to overcome the poor scaling of register file latency and area limitation posed by the large multi-ported register file. Alpha 21264's 4-issue integer execution core which would normally require an 8-read port, 4-write port register file. Instead, the architects chose to duplicate the entire contents of the register file such that each copy only needs half as many read ports [71], but the same number of write ports. Two full copies of a moderately ported register file proved to be smaller and faster than a single highly-ported structure.

Table 15 shows the parameters of our baseline planar processor.

We model the 3D processors by either partitioning and stacking the planar components such that a part of each component resides on each stacked die or by stacking the whole units on one of the die, depending on benefits and feasibility. We evaluate both 2-die-stacked and 4-die-stacked 3D implementations of the baseline planar processor.

5.2.2 2-Die-Stacked 3D-Integrated Processors

For the 2-die-stacked 3D implementation, we partition and stack the SRAM-based components such as caches and register files with half the entries each die (Chapter 2). Similarly, we stack the CAM-based components such as instruction schedulers with half the entries on each die (Chapter 3). Combinational logic components such as arithmetic and logical units are split such that either half

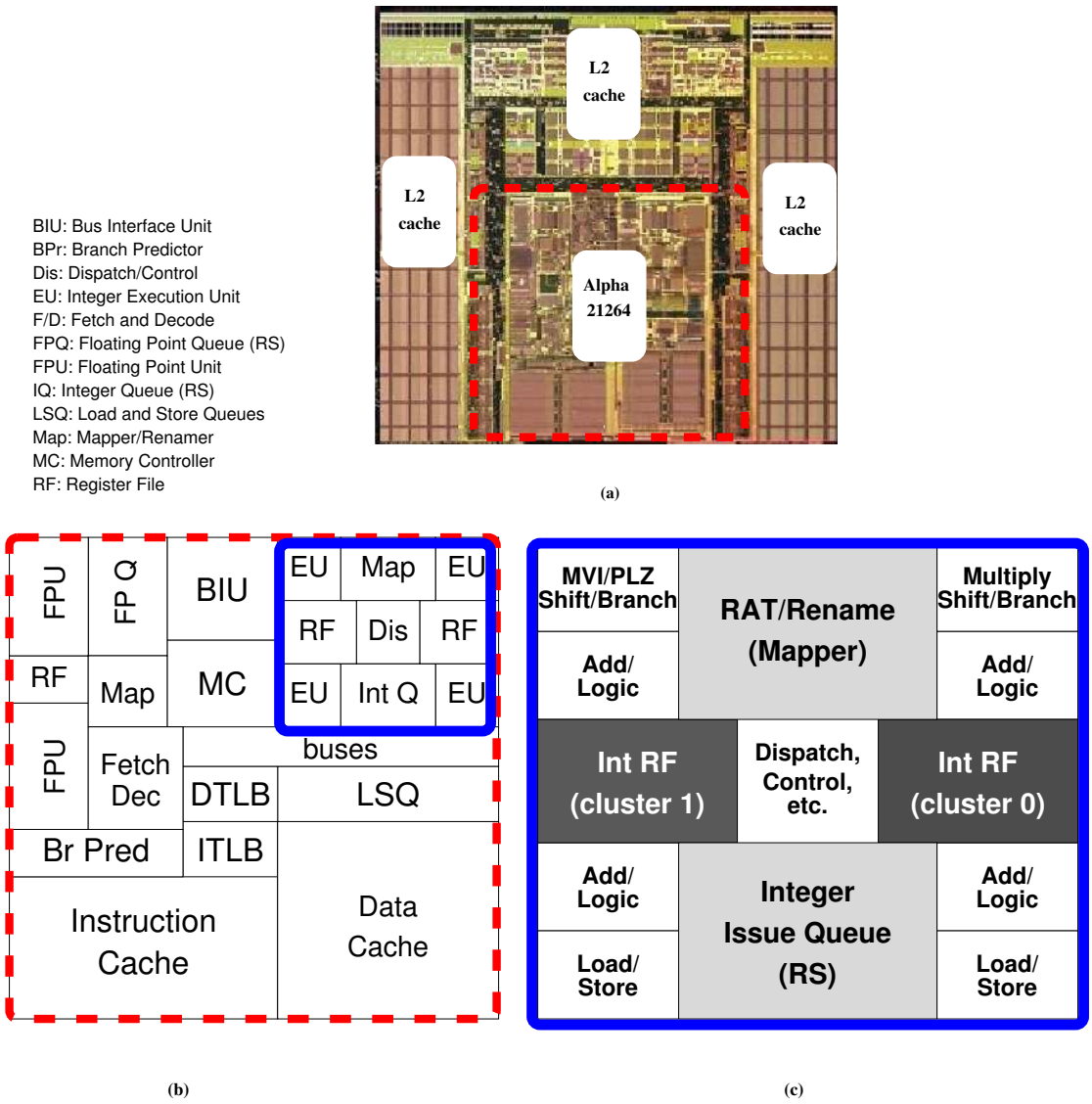

Figure 56: Baseline planar processor (a) Die photograph and floorplan of the Alpha 21364 [49] The floorplan of (b) the 21264 core [71], and (c) Our floorplan of the 21264 integer execution core (EBox).

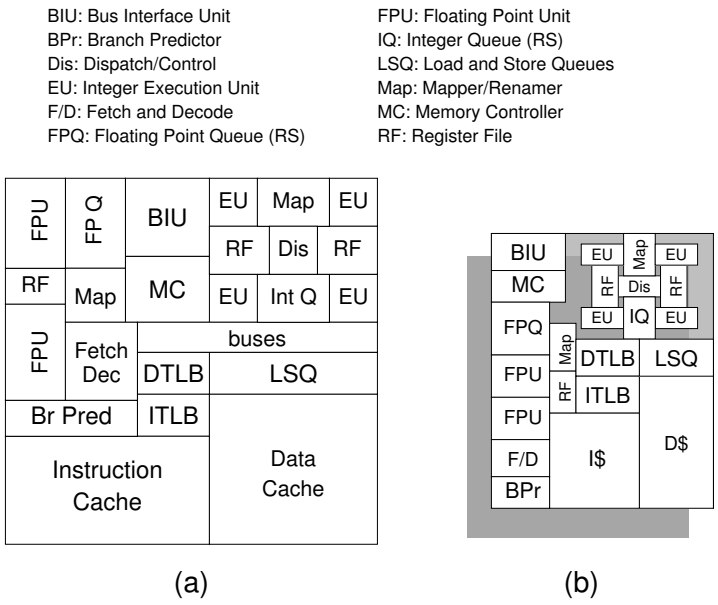

Figure 57: (a) Our baseline planar floorplan for the 21364 core, L2 cache not shown, (b) a compacted 3D-integrated 2-die-stacked floorplan.

the operands are processed on each of the die or whole units are placed on one of the stacked die depending on benefits and feasibility (Chapter 4).

The first 3D-integrated processor that we evaluate is loosely based on the Alpha 21364 as shown in Figure 56(a). Its architectural configuration is identical to the planar configuration shown in Table 15.

Figure 57(a) repeats the original planar processor floorplan for convenience, and Figure 57(b) shows our 2-die-stacked 3D processor. After reducing the footprints of each of the components, we manually compacted the processor floorplan. Since the original floorplan was not designed with 3D-integration in mind, we end up with some unutilized regions on the floorplan. Most of the refloorplanning and compaction happened in a fairly straightforward manner.

The 3D-integrated processor illustrated in Figure 57(b) provides performance benefits by enabling a faster clock frequency. We call this 2-die-stacked 3D processor $21364f_2$ since it is the identical microarchitecture as the original planar baseline, but with faster frequency. To achieve an overall frequency improvement, the latency of all critical paths must be reduced. While the 3D-integration may be able to accomplish this, it is also reasonable to argue that for many processor designs it is difficult to retime all of the critical paths.

An alternative use of the 3D-integration is to keep the original planar floorplan as shown in Figure 57(a), but make use of the additional integration capacity to increase the processor resources (e.g., more instruction queue entries, larger register files, branch predictor tables). We call this 2-die-stacked 3D processor 21364++$_2$ since it is identical in floorplan as the original planar 21364, but with larger resources.

5.2.3 4-Die-Stacked 3D-Integrated Processors

For the 4-die-stacked 3D-integrated processor, we partition the various microarchitectural components on each of the four stacked die. We combine various stacking strategies to achieve a balanced floorplan. For example, by making the first 3D-partition in the X- direction and then making the second partition in the Y- direction, and repeating it for all the components, we obtain a 4-die-stacked 3D floorplan that is identical to the planar floorplan except for a halving of length in both the X- and Y- directions, which makes the footprint to be a quarter of the baseline planar processor footprint. With such partitioning, the 2-die-(4-die-)stacked 3D designs will have approximately half (quarter) the footprint as the planar design.

We consider the 4-die-stacked 3D versions of the baseline planar processor. In this design, the floorplan is similar to that of the 2-die-stacked processor shown in Figure 57(b), except that the components are now split across four layers. As a result, the footprint of each component is further decreased and the overall processor floorplan can be further compacted, as depicted in Figure 58(b). The 3D-integrated processor illustrated in Figure 58(b) provides performance benefits by enabling a faster clock frequency, that is even higher than the 21364f$_2$. We call this 4-die-stacked 3D processor 21364f$_4$ since it is the identical microarchitecture as the original planar baseline, but with faster frequency. Similarly, we also evaluate a 4-die-stacked version of 21364++$_4$.

5.3 Experimental Procedure

5.3.1 Circuit Latency and Energy

Using critical-path Hspice simulations as explained in Part II, we evaluate the latency and energy for the planar, the 2-die-stacked and the 4-die-stacked 3D processors. Note that the fast 3D-integrated processors (21364f$_2$ 21364f$_4$) have reduced footprints than the baseline planar processor, with the reduction being proportional to the number of stacked die. We utilize the latency saving of the

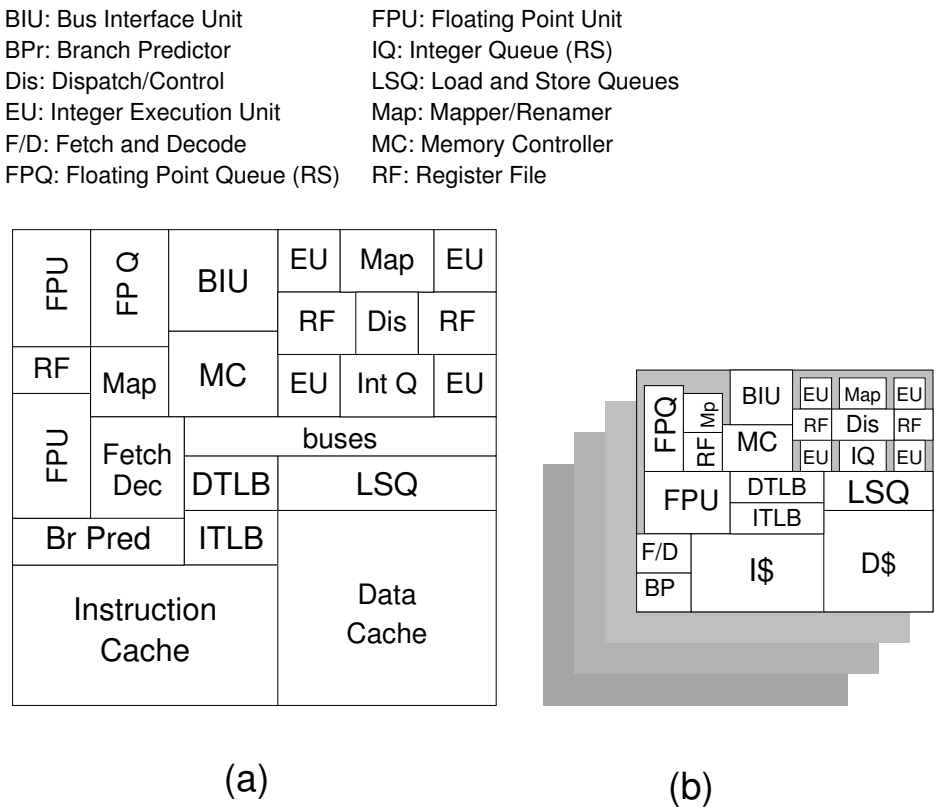

Figure 58: (a) Our baseline planar floorplan for the 21364 core, L2 cache not shown, (b) a compacted 3D-integrated 4-die-stacked floorplan.

3D-integrated circuits to proportionally increase their clock frequency which in turn increases the power consumption.

The ++ 3D-integrated processors (21364++$_2$ 21364++$_4$) have identical footprints as the baseline planar processor. These 3D processors run at the same frequency as the planar processor, but provide higher performance due to larger-sized resources. We run Hspice simulations to find the largest possible sizes of the 3D circuits, that can be implemented without exceeding the corresponding latency of the planar circuit.

Using the Hspice framework, we obtain the latency and power data for the various components. After obtaining the power data, we perform detailed temperature simulations of the planar and the 3D-integrated circuits using a temperature simulation tool called HotSpot (version 3.0.2) from the University of Virginia [136].

5.3.2 Temperature Analysis

HotSpot can model multiple layers of silicon and metal required to represent a 3D processor. HotSpot requires power consumption data and floorplan data as inputs and generates the temperatures for various functional blocks. Using HotSpot, we simulate each die on the 3D stack as three layers: bulk silicon, active silicon, and the metal layer. The planar and the 3D models include the physical and the electrical details of metal routing (A), active devices (B), thinned bulk silicon substrate (C), d2d vias between each adjacent pair of die (F2F and B2B), heat spreader (D), and heat sink (E).

Figure 59(a) shows our planar integrated circuit modeled as five layers: metal layers (A), active silicon (B), bulk silicon (C), heat spreader (D) and heat sink (E). The heat spreader is attached to the bulk silicon with a thermal interface material [124]. Figure 59(b) and (c) show our 2-die-stacked and 4-die-stacked 3D-integrated circuits respectively.

We compute an average specific heat capacity (SHC) and thermal resistivity (TR) of the metal layer taking into account the proportion of the metal and inter-layer dielectric on each of the layers. We obtained the material properties, listed in Table 16 from the CRC handbook [35]. For the interface between adjacent die, we compute an average SHC and an average TR based on the fraction of the interface that is occupied by copper versus air. We model the D2D via width to be half of

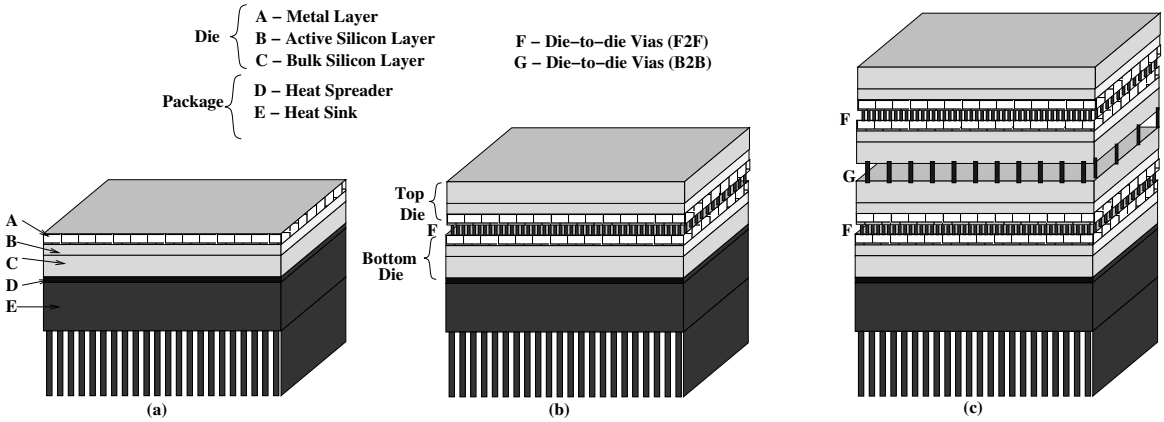

Figure 59: (a) A planar integrated circuit (b) A 2-die-stacked 3D-integrated circuit (c) A 4-die-stacked 3D-integrated circuit (Figures not to scale)

Table 16: Material properties

Material	Specific heat capacity (SHC) $J/m^3/K$	Thermal resistivity (TR) $(W/m/K)^{-1}$
Cu	3.49E+6	2.53E-3
Si	1.75E+6	0.01
SiO_2	1.79E+6	1.69
Air	1.51	40

the via pitch, which results in a 25% copper occupancy (75% air) at the die-to-die interface. The heat spreader is attached to the bulk silicon with a thermal interface material [124]. We use a phase change metallic alloy [124] for the thermal interface material (TIM) between the bulk silicon layer of the last die and the heat spreader. We present our results using this methodology in the rest of this chapter.

5.4 Results

5.4.1 Latency

Table 17 shows the latencies of our baseline planar components as well as the 2-die-stacked and the 4-die-stacked processors. We also list the largest-sized 3D circuit that can be implemented without exceeding the corresponding latency of the planar circuit. For example, the planar L1 cache is of size 64 KB and requires 1536 ps for a read access. The 2-die-stacked 64 KB L1 cache (present in $21364f_2$) requires only 1159 ps for the read access and the 4-die-stacked 64 KB L1 cache (present in $21364f_4$) requires only 979 ps. With 2-die-stacked implementation, we can implement a 128 KB

Table 17: HSpice timing results for various microarchitectural modules for planar, 3D 2-die-stacked and 3D 4-die-stacked implementations, and the largest size implementable for each module without exceeding the latency of the corresponding planar implementation.

Module/ Circuit	2D	3D 2-layer		
	Latency (ps)	Latency (ps)	% Speedup	Largest Size
L1 Cache (64KB)	1536	1159	24.5%	128KB
L2 Cache (1MB)	3551	2834	20.2%	1MB
BPred - (Local/Global/Meta:2/1/1KB)	760	667	12.3%	(2/2/2)
Load/Store Queue (32 each)	252	185	26.5%	44 entry
Int RAT (80 regs)	347	241	30.5%	120 regs
Int Issue Queue (20 entry)	467	416	11.0%	40 entry
Intra-Cluster Int Bypass	224	201	10.3%	—
Cross-Cluster Int Bypass	447	310	30.7%	—

Module/	3D 4-layer		
Circuit	Latency (ps)	% speedup	Largest Size
L1 Cache (64KB)	979	36.3%	256KB
L2 Cache (1MB)	2129	40.1%	2MB
BPred - (Local/Global/Meta:2/1/1KB)	615	19.1%	(2/4/4)
Load/Store Queue (32 each)	154	38.8%	80 entry
Int RAT (80 regs)	189	45.6%	160 regs
Int Issue Queue (20 entry)	396	15.1%	80 entry
Intra-Cluster Int Bypass	190	15.3%	—
Cross-Cluster Int Bypass	246	45.0%	—

cache to run within 1536 ps (planar L1 cache latency). So the $21364{+}{+}_2$ contains a 128 KB L1 cache running within the same latency as the planar 64 KB cache.

Most of our circuits are implemented in static CMOS logic. Hence, some of the absolute latency values may be slower than expected. This results in conservative speedup estimates since wire delay would consume a larger fraction of the clock period in dynamic logic implementations. More aggressive custom logic design would likely result in an even greater relative 3D-integration benefit.

Different modules in Table 17 exhibit different amounts of latency improvements due to differences in the impacts of wire delays. Some circuits are more wire-dominated than the others due to the differences in capacity and port requirements. In determining the clock frequency of a processor, not all of the components are limiting factors. For example, since there are no single-cycle floating point instructions, the FP issue queue can be pipelined and may not directly help nor hinder the overall cycle time. Palacharla [106] identified the integer scheduling logic and the bypass network as critical cycle-time limiters [106]. We focus our attention on these two modules (for both the planar and 3D-integrated cases), and use these as an approximation of the overall processor clock

frequency benefit[1]. We take the smaller of the two improvements as the expected cycle time reduction. The 2-die-stacked 3D processor provides a 10.3% frequency boost, and the 4-die-stacked 3D processor provides a 15.1% improvement.

5.4.2 Power Consumption

3D-stacked circuits can reduce power consumption because the shorter wire lengths present less capacitance to the driving logic. Furthermore, it is important to note that this power reduction is *in addition to* any latency benefits from having shorter wires. When we use the 3D-integration to increase the sizes of the components, we may expect to see power increases.

It is important to note that in our higher frequency floorplans, we are not adding capacitance in the same fashion as repipelining (increasing the pipeline stages) of a planar processor. In the planar processor, the pipeline's total non-latch latency is constant, and this latency is redistributed over a larger number of pipeline stages due to repipelining. This adds power in the form of more pipeline latches and their associated microarchitectural changes to maintain performance in the deeper pipeline (e.g., aggressive speculative scheduling and the associated replays, and deeper and more complex bypass networks). This results in both C and f increasing in the $P = \frac{1}{2}CV^2f\alpha$ equation. When using 3D-integration to achieve a higher clock speed, we reduce the total non-latch latency required to execute the instruction by eliminating a portion of the total wire delay, but the microarchitecture remains largely unchanged. In 3D processors, f increases but C decreases, which may help the processors to extract more power benefits.

The overall power benefit varies depending on the 3D configuration. When considering the reduction in energy consumption of the entire processor, $21364f_2$ achieves a 8.52% overall energy reduction. However, the *power* consumption is affected by the frequency increase, and the net effect is that the power increases by 0.9%. Despite the fact that many components in $21364{+}{+}_2$ are larger, there is actually an overall power reduction of 3.46%. One of the reasons for the overall power reduction is the increased hit rate of the larger L1 cache that results in a lower access frequency of the L2 cache. For the 4-die-stacked 3D configurations, the overall power benefits are even better: 16.7%

[1]We do realize that the *absolute* latencies differ which is a result of a variety of assumptions that we made in the designs, layout and floorplanning of the circuits. We believe that the relative benefit is still representative of the cycle time improvements achievable in practice.

for 21364f$_4$ (including the effects of the increased frequency) and 12.87% for the 21364++$_4$. The shorter wire lengths reduce the amount of capacitance that must be switched, but perhaps just as importantly, they reduce the total resistance as well. The reduction in both resistance and capacitance of the wire results in a significant overall power reduction.

While our focus is on using the 3D-integration to extract more performance, another alternative is to use the 3D-integration to implement low-power processors. The wire-length reduction of the 3D-integrated circuits can provide latency savings, and this latency reduction can be traded for further power reduction. Besides simple voltage scaling (which is becoming more difficult as the on-chip supply voltages are already 1V or less), other options include replacing power-hungry dynamic logic with CMOS gates, reducing the transistor sizes (also decreases leakage), using slower transistor implementations such as thicker oxides and higher threshold voltages (both of which are used in the Sony Cell SPE [144]), or using longer channels or stacked transistors (reduces leakage).

5.4.3 Temperature

We demonstrated that the benefits of the 3D processors increase as we stack more die, due to successive reductions in wire lengths. However, as we stack more die, the power density increases due to vertical stacking of the active devices and reduced die footprints, thus causing the on-chip temperatures to increase. Also, the die located further from the heat sink experience a longer heat dissipation path to the heat sink. In this section, we analyze the temperatures on the 3D-integrated processors. We begin by presenting the effect of power density.

5.4.3.1 Power Density

We assume that the total power consumption of each of the components on the 3D processor is uniformly distributed among the corresponding stacked partitions of the block. For example, if the planar arithmetic adder consumes a power of 300 milliwatts and our Spice circuit simulation of the 2-die-stacked 3D adder shows a power reduction of 5% over the planar adder (285 milliwatts), we assign the power consumption of the 2-die 3D adder to be 142.5 milliwatts per die. We similarly assign the power consumption of other components on each of the die. Note that this may be a conservative assumption in terms of power density since it is possible to self-stack some circuit components such that only one of the stacked partitions is active at any given time. For example,

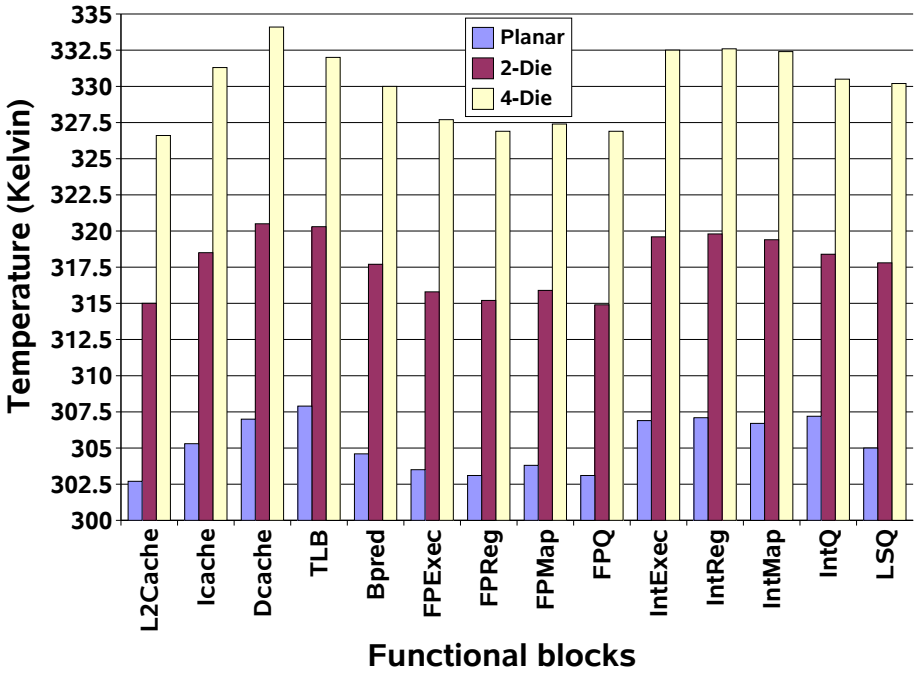

Figure 60: Temperatures of components on the planar and the 3D-integrated processors

consider a planar cache that has 256 lines. The 2-die 3D implementation may stack 128 lines on each die. The address decoding logic of the 3D cache selects only one of the two die based on the input address. Thus, the address decoding logic generates activity and power dissipation on only one of the die, thus cutting the overall switching power roughly in half. This power reduction might offset the halving of the 3D processor footprint and cause the 3D power density to be roughly the same as that of the planar processor. Also, if different functional components are stacked, they may not both be actively dissipating power simultaneously. For example, if an integer adder is stacked on an integer multiplier, only one of them might be active based on the application requirements.

Figure 60 shows the die temperatues of the planar and the 3D-integrated processors based on our temperature analysis. From Figure 60, the maximum temperature on the planar processor is 307.9 K. The corresponding maximum temperatures on the 2-die and the 4-die 3D implementations are 320.5 K (12.6 K increase over the planar processor) and 334.1 K (26.2 K increase over the planar processor), respectively. Note that the rough doubling of the power density experienced by the 3D-integrated processors has lead to only a moderate increase (12.6 K and 26.2 K for 2-die and 4-die processors, respectively) in the maximum on-chip temperatures. The maximum temperatures

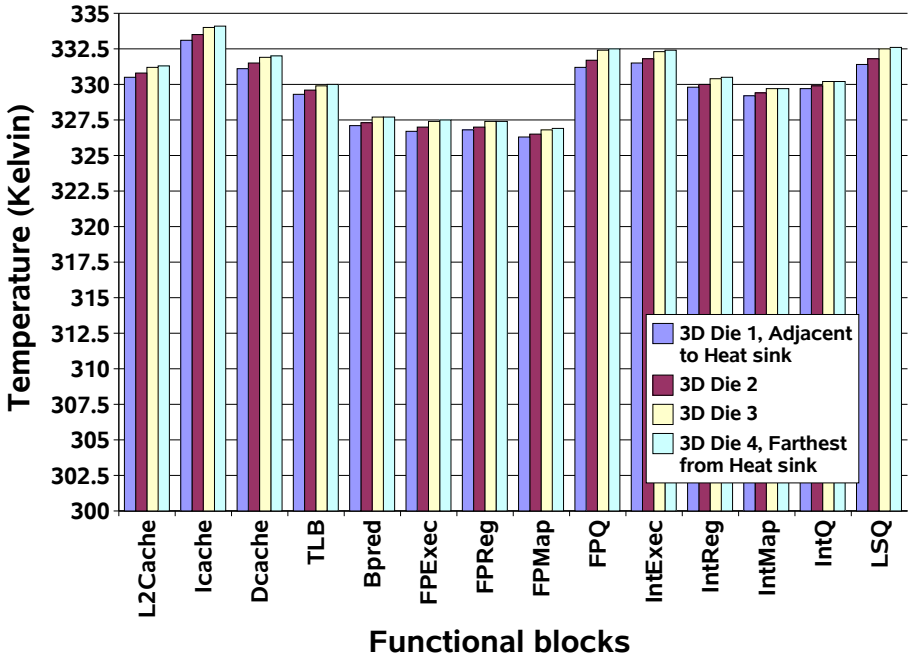

Figure 61: Temperatures on each of the four die on a 4-die-stacked 3D processor

from our experiments is in contrast to the data published by prior research where the 2-die-stacked 3D-integrated processors are reported to increase the on-chip temperatures by as much as 50 K [29, 60]. The large difference in the reported temperatures can be explained by the fact that our model includes the current technology advancements such as wafer thinning, copper metallization, and efficient packaging materials [124].

5.4.3.2 Temperatures on Stacked Die

We compare the temperatures on the individual die of the 4-die 3D processor. Figure 61 shows the temperatures of the components on each of the four die on the 4-die 3D processor. From Figure 61, we see that, for each functional block, the die that are further away from the heat sink have a slightly higher temperature than the die that are closer to the heat sink. But the overall difference in temperatures among the individual die themselves is not large suggesting that the die interfaces are efficient at handling inter-die temperature.

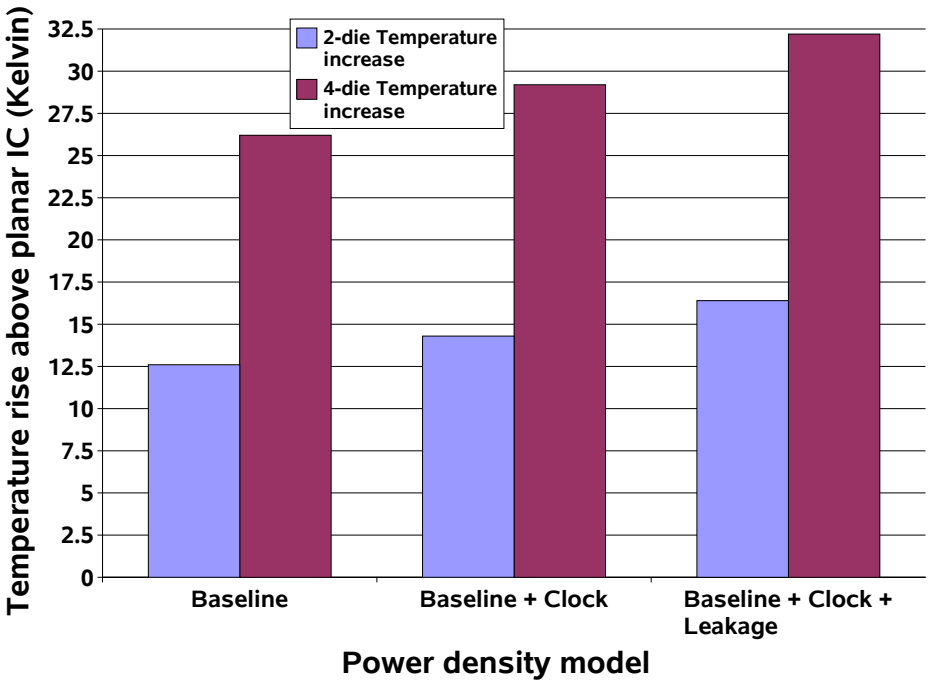

Figure 62: Maximum temperature increase with clock and leakage power modeling

5.4.3.3 Modeling Clock Power and Leakage Power

We add the clock power and the leakage power estimates to our processor power consumption data. The Alpha processor is reported to dissipate 34% of the system power in the clock network [24]. Since our baseline processor is based on the Alpha processor core, we assign the clock power to be 34% of the system power and correspondingly increase the power consumption of each of the components in both the planar and the 3D processors. We model the leakage to be 15% of the system power (65nm technology) based on the leakage power data in [130].

Figure 62 plots the temperature increase for the 3D-integrated processors over the planar processor baseline after taking the system clock power and the leakage power into account. From Figure 62, the maximum temperatures on the 2-die and 4-die 3D implementations increase by 16.4 K and 32.2 K, respectively over the baseline planar processor. The increased temperatures on the 3D-integrated processors may require more aggressive cooling mechanisms [108], thus increasing manufacturing costs.

5.5 Summary of the 3D-Integrated Processors

In this chapter, we showed that the 3D-integrated processors provide simultaneous performance and power benefits. We also showed that the benefits increase as we stack more die. Note that the Alpha processors are traditionally considered speed demons and hence the frequency benefits of 2-die-stacked and 4-die-stacked circuits may seem to be modest ($\sim$10% and $\sim$15% while increasing the temperature by 12K and 26K respectively). The important thing to keep in mind is that we are starting from an architecture that is designed and optimized for a conventional planar technology.

In Chapter 6, we describe our 3D-integrated processors based on the Intel Core microarchitecture. In addition to evaluating the latency and power benefits due to wire reduction, we propose microarchitectural techniques that target the switching activity which is another component in dynamic power. By effectively steering switching activity to enable more efficient heat removal, our microarchitecture techniques are able to control thermals on the 3D-integrated processors.

CHAPTER VI

THERMAL HERDING 3D-INTEGRATED PROCESSORS

6.1 Overview of This Chapter

We describe the 3D-integrated design of a planar dual-core high-performance processor. We propose several microarchitecture-level techniques to address the thermal challenge in 3D-integrated processor cores. While we present a variety of methods, they can all be categorized under the general theme of *Thermal Herding*. Thermal Herding techniques herd or steer the majority of the processor's switching activity to the die that is closest to the heat-sink. Our 3D/thermal-aware microarchitecture contributions include a significance-partitioned datapath (frequently-switching, lower-order 16-bits are placed on the die closest to the heat-sink), a 3D-aware instruction scheduler allocation scheme, an address memoization approach for the load and store queues, a partial value encoding for the L1 data cache, and a branch target buffer that exploits a form of frequent partial value locality in target addresses.

We evaluate the performance, power, and temperature of our Thermal Herding 3D-integrated processor. We demonstrate that our microarchitecture-level Thermal Herding techniques can effectively control the power density and temperature issues in 3D-integrated processors. We show our Thermal Herding 3D processors to have frequency, power and thermal advantages over the coarse-grained 3D-integrated processors (refer to Figure 7(c)).

We explore the design space involving performance, power, and temperature. We vary several design and physical parameters of 3D-integrated processors and explore the impact of those parameters. In particular, we consider the impact of total power consumption, 3D die-to-die via density, and stacked die thickness on 3D processors. We explore the impact of the 3D technology on different market segments (desktop versus servers) by experimenting with different power budgets. We explore the impact of fabrication technology by varying the die-to-die via density and the die thicknesses.

Overall, we show that it is possible to keep 3D thermals under control through a combination of

Table 18: Circuit and microarchitecture parameters of the baseline processor.

Parameter	Value	Parameter	Value
Process technology	65 nm, 8M Copper, CMOS	Inst Fetch Queue	16 entry
Core voltage	1.3 - 1.5 Volts	Issue	Max. 6/cycle
Pipeline	14 stage	*Integer*	3 ALU 2 shifts 1 complex ALU
Fetch/Decode/Commit	4 instructions/cycle	*Floating Point*	1 ALU 1 multiplier 1 divider/sqrt
Unified L2 cache	4MB, 16-way, 12-cycle	*Memory*	1 Load/Store port 1 Load-only port
Instr/Data L1 caches	32KB, 8-way, 3-cycle	Reorder buffer size	96 entries
Branch Predictor	10KB Bimodal/Local/Global hybrid	Scheduler size	32 entries
Branch Target Buffer/iBTB	2K/512-entry, 4-way	LoadQ/StoreQ size	32/20 entries
Instr/Data TLBs	128/256-entry, 4-way	Branch Mispred Latency	Min. 14 cycles

reducing total processor power, local power density, and effective thermal resistance while simultaneously increasing performance by a significant amount. This performance, power and temperature analysis of our final design 3D-integrated processor design provides further evidence in support of our thesis that the 3D-integration provides simultaneous performance and power benefits to build high-performance microprocessors, while keeping the worst-case temperature under control.

The rest of this chapter is organized as follows. Section 6.2 introduces our planar, Thermal Herding 3D, and coarse-grained 3D processor configurations. Section 6.3 explains the details of our Thermal Herding microarchitecture techniques. Section 6.4 describes our evaluation framework and technology assumptions for circuit latency, instructions per cycle (IPC) performance, power, and temperature estimations. Section 6.5 presents our performance, power, and thermal results. Section 6.6 summarizes our work and provides some concluding remarks.

6.2 Planar and 3D-Integrated Dual-Core Processors

6.2.1 Baseline Planar Processor

We model our baseline planar processor loosely based on the Intel Core microarchitecture [63, 62, 70, 126]. Table 18 lists the parameters, and Figure 63(a) shows our assumed floorplan for our baseline planar dual core processor superposed on a die micrograph [63, 62, 126]. Based on our circuit analysis, we assign a clock frequency of 2.66 GHz to the baseline planar processor.

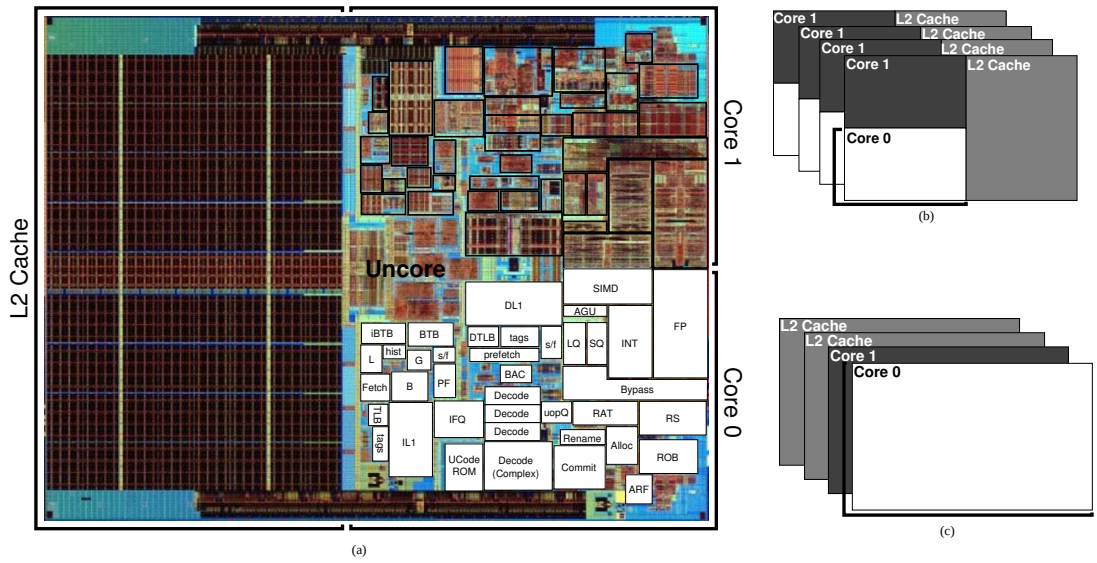

Figure 63: (a) Floorplan for the baseline planar processor. (b) 4-die-stacked Thermal Herding 3D processor, and (c) 4-die-stacked coarse-grained 3D processor, (Not to scale)

6.2.2 4-Die-Stacked Thermal-Herding 3D Processor

For our Thermal Herding 3D processor, we make use of circuit-level partitioned components explained in Part II of this dissertation. Figure 63(b) shows the organization for our Thermal Herding 3D processor. The footprint is approximately one quarter of the baseline planar processor footprint since the logic is distributed over four die. For our Thermal Herding 3D processor, the majority of the layout is identical to the planar processor except for a $\sim 4\times$ footprint reduction due to the partitioned implementation of individual circuit blocks on four die. We manually compacted the floorplan to reduce empty regions (whitespace) and achieve tighter placement of the blocks. Our Thermal Herding 3D processor is based on two concepts, namely significance-partitioned datapath and datawidth prediction.

6.2.2.1 Significance-Partitioned Datapath

Past research has observed that many integer instructions use data that require only a few of their lower order bits [23, 155]. In particular, many 64-bit integer values require only sixteen or fewer bits to represent. Hence, we organize our datapath by assigning each sixteen bit slice of the datapath to a separate die. We place the lower-order (most likely to switch) bits on the die that is closest to the heat-sink. Our 3D microarchitecture partitions a majority of the data and control paths such that the

communication between die is either completely eliminated or isolated to the periphery of modules where space can be allocated for the d2d vias without impacting the critical paths within blocks. The primary focus of this dissertation is the microarchitecture of the 3D-integrated processors, but it is interesting to note that the microarchitecture design can also be used to address d2d communication constraints.

6.2.2.2 Data width Prediction

Past research has observed that an instruction's usage of low-width values is highly predictable [89, 40]. Hence, we make use of the instruction's data width locality property to further save power. For each instruction, the processor makes a prediction whether to use low-width ($\leq$16-bit) or full-width ($>$16 bits) values. Our scheme uses a simple program counter (PC)-indexed 2-bit saturating counter predictor [40]. When the predictor predicts an instruction to be low-width but the data is actually full-width, the result is an *unsafe* misprediction. An unsafe misprediction requires pipeline stalls in relevant pipeline stages. The complementary case of a *conservative* or safe misprediction does not cause pipeline stalls, but it is a missed opportunity to reduce processor switching activity and save power. In order to exploit data width locality in the 3D-integrated processors, additional gating circuitry is required to disable accesses to the die that store the unused portions of the datapath.

6.2.3 4-Die-Stacked Coarse-Grained 3D Processor

Figure 63(c) shows the organization of our coarse-grained 3D (CG-3D) processor. The coarse-grained 3D configuration stacks the two planar cores on one die each, and the L2 cache banks on two additional die. CG-3D can provide a reduced footprint benefit (due to stacking cache on processor) and a small performance benefit (due to reduced wire lengths between the cores and the stacked cache), but it does not fully exploit the benefits and flexibility of 3D-integration (i.e., the clock frequency and power consumption of the individual cores in the 3D implementation are the same as in the planar implementation) Since CG-3D does not change the core clock speed, we assign a clock frequency of 2.66 GHz to the CG-3D processor.

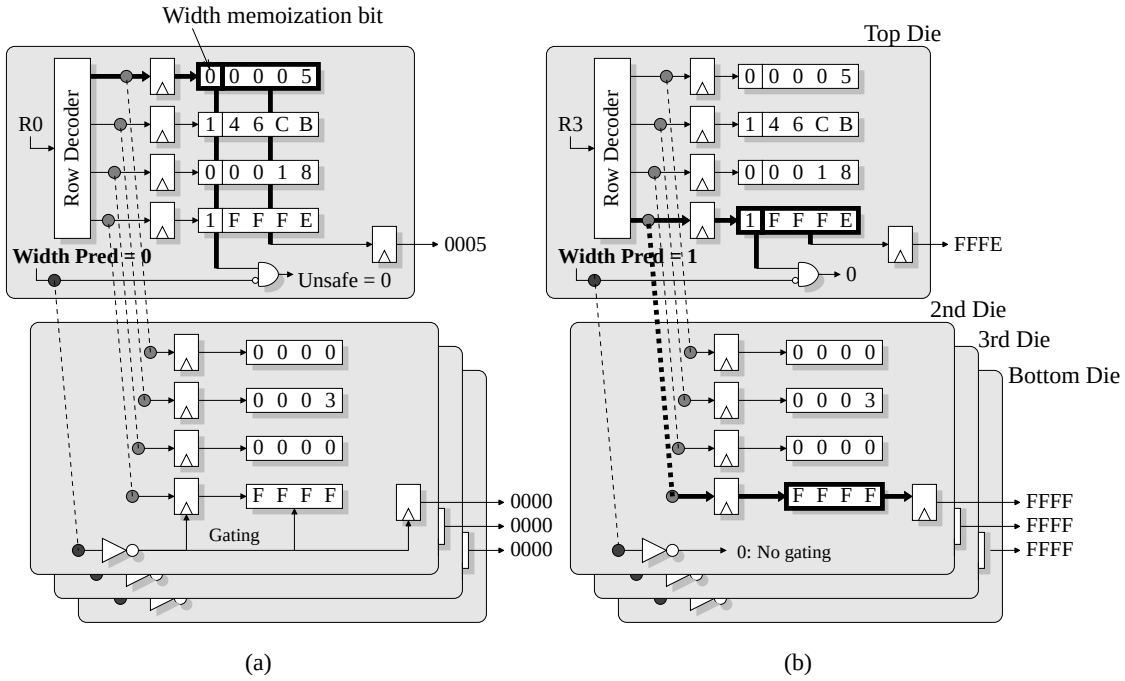

Figure 64: Thermal Herding in register files: examples where (a) a low-width value requires access to only the top die, and (b) a full-width value requires access to all four die.

6.3 Thermal-Herding Techniques for 3D Microarchitectures

This section describes our *Thermal Herding* microarchitecture techniques to reduce the total power consumption and the power density of the 3D processors while still maintaining the performance benefits of the 3D integration. In the next few sections, we discuss some critical Thermal Herding components of our 3D processor.

6.3.1 Register Files

We partition each 64-bit entry in the register file (RF) such that each word (16-bits) resides on a separate die. This word-partitioned 3D register file organization reduces the access latency and the dynamic power consumption [111]. We use width *prediction* to enable early determination of gating control signals in advance of the actual register file access. On a predicted low-width instruction (Figure 64(a) shows such an access for register R0), only the top die portion of the register file is active. When we access only the top die, the power density is similar to that of a planar register file, and the activity is isolated to the top die (adjacent to the heat-sink). In the case of a predicted full-width access (R3 in Figure 64(b)), all four die are active.

The top die (least-significant 16-bit-word slice) contains a *width memoization* bit for each RF entry. The memoization bit indicates whether the remaining data-slices residing on the other three die are non-zero values. The processor datapath tracks the actual widths of each of the register values using the memoization bit. On reading the width memoization bit, the processor compares it to the predicted width. If the width prediction is low and the actual width is full, then the processor performs two actions: (1) it gates (stalls) the previous stages of the register file and enables the logic on the remaining three die, and (2) it corrects the instruction's width prediction to prevent any further stalls in the rest of the pipeline.

In a superscalar processor, the register file provides operands for many instructions in parallel. To maintain the program-order in the front-end, any register file stall due to an unsafe misprediction prevents all later (in program order) instructions from dispatching to the out-of-order back-end of the processor core. All instructions that suffer from unsafe mispredictions in a *group* can be serviced in parallel in the next cycle, and therefore any group of instructions (those accessing the register file in the same cycle) can induce at most one stall for the entire group regardless of whether one or all of them had unsafe mispredictions.

Note that the register file latency impacts not only the number of cycles in the conventional "branch mispredict detection" pipeline but also post-commit latencies such as those required to copy the committed values from the physical register file to the architected register file.

6.3.2 Arithmetic Units

We choose a planar parallel-prefix tree-based integer adder and explain our Thermal Herding design of the 3D integer adder; however, the concepts presented here can be extended to the design of other arithmetic units. Figure 65 shows our 4-die implementation of the tree-based integer adder. In our 3D design, the portion of the adder that adds the lower order 16-bits resides on the top die closest to the heat-sink. Our 3D-integrated adder compacts the physical placement of the upper levels of the adder's carry logic which contain the adder's longest wires.

Even though the register file provides the memoization bit that indicates if an instruction's operands are low-width, a full-width prediction initiates access to the entire adder because two low-width operands may generate a full-width result (e.g., adding two 16-bit values may result in

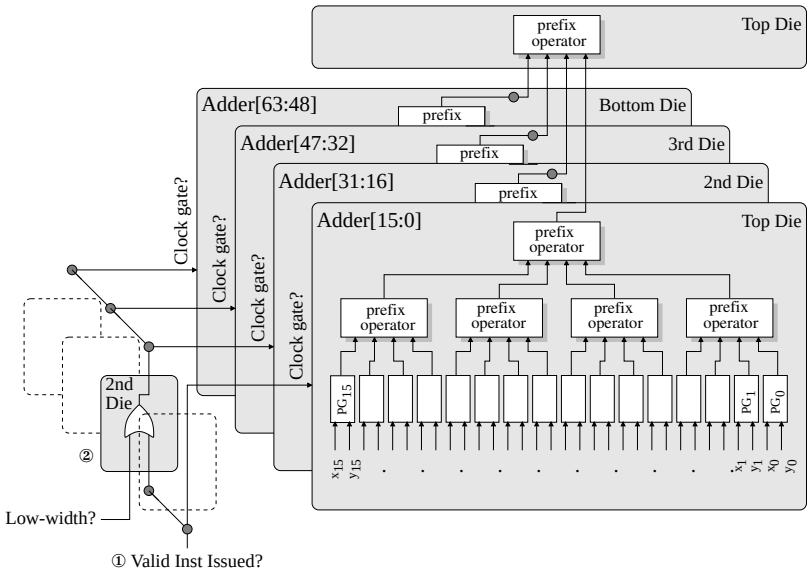

Figure 65: Thermal Herding in an integer adder with the most active (lower order) placed on the top die

a 17-bit sum). In the cycle prior to execution, a conventional processor can decide whether it can clock gate (① in Figure 65) the higher-order bits of the adder and save dynamic power, by using the information residing in the instruction scheduler. Figure 65-② shows the additional input to the clock-gating logic to gate the bottom three die of our Thermal Herding 3D adder.

There are two possible unsafe width-misprediction scenarios. The first is a misprediction on the instruction's input operands. If the width-predictor predicts that the instruction has low-width operands but its operands are actually full width, then the arithmetic unit is not fully enabled at the start of the instruction's execution. This requires a one cycle stall to re-enable the higher order bits of the arithmetic unit. The second type of unsafe misprediction is on the output of the arithmetic unit. In the case of output width misprediction, the width misprediction may not be known until several cycles into the computation (for pipelined functional units) and so we force any instruction with unsafe output width misprediction to re-execute. While these mispredictions can induce a performance penalty, the accuracy of the width predictor prevents this from causing a significant deterioration in the performance.

When our TH-3D adder operates on correctly predicted low-width values, our approach not only reduces the total power but also maintains comparable power density as the planar adder. As demonstrated in Chapter 4, large and wire-dominated arithmetic units, e.g., shifters and multipliers,

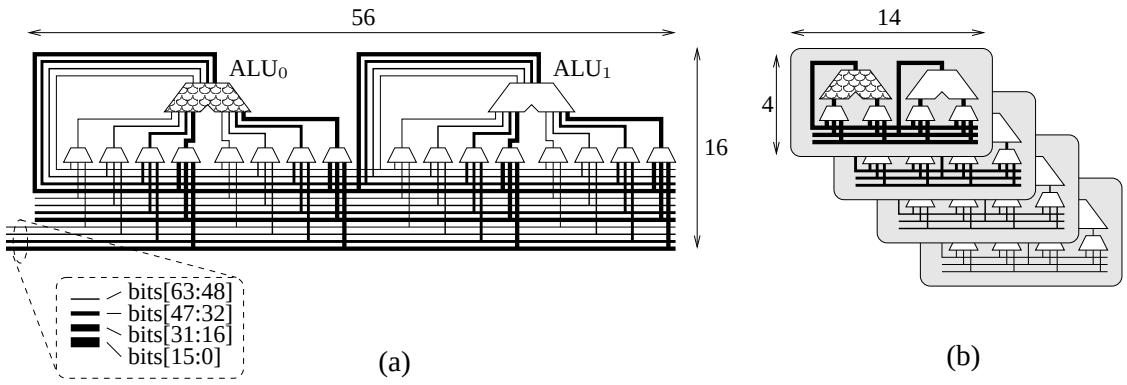

Figure 66: (a) A planar bypass network with a register file path and two ALU outputs, and (b) the equivalent 3D bypass network.

will benefit even more from the wire reduction [112]. Hence, our Thermal Herding 3D circuits are simultaneously faster and lower power while having a similar power density profile as the planar functional units when handling low-width values.

6.3.3 Bypass Network

We organize the bypass network using significance-partitioning with sixteen bits per die. Since the unsafe mispredictions have already been handled by the arithmetic units, the bypass network does not need additional circuitry to handle the width mispredictions. A correctly predicted low-width output will cause only the drivers/wires on the top die to dissipate dynamic power. A full-width output will cause activity on all die. In addition to the power density reduction due to Thermal Herding, the wire-intensive nature of the bypass network allows for a substantial reduction in wire-related area, latency and power. Figure 66(a) shows two ALUs and the bypass network. Figure 66(b) shows the ALUs and the bypass implemented in a 4-die organization. Note that the dimensions of *both* the width and height of the bypass network have been reduced to a quarter of their original sizes.

6.3.4 Instruction Scheduler

We partition the instruction scheduler based on the RS entries, with one quarter of the entries placed on each die [110]. Although there is a slight overhead to fan-out the tag broadcasts to all four die, this organization greatly reduces the lengths of the broadcast buses which results in overall power

and latency benefits. We combine this entry-partitioned (EP) scheduler organization (refer Chapter 3) with a modified allocation algorithm that herds instructions toward the top die to keep the active entries close to the heat-sink. If there are no available entries in the top die, then the allocator starts allocating on the die that is next closest to the heat-sink. To further reduce power consumption, the RS entries can make use of the allocator's information regarding the occupancy of each die. If there are no occupied RS entries on a given die, then the tag broadcast for that die can be gated, leading to further power reductions.

6.3.5 Load and Store Queues

The data loaded from and stored to memory exhibit value-width properties similar to that of the register values. As a result, we propose to significance-partition these queues similar to the main datapath. This significance-partitioning provides the additional advantage that the values propagating between these structures already have their bits located on the appropriate die, avoiding the need for extra d2d vias.

Load and store addresses are almost always full-width values. However, we observe that the upper bits of the addresses do not frequently change. For example, loads and stores to and from the stack are likely to have identical upper address bits. To exploit this phenomenom, we use *partial address memoization* (PAM). On the top die, we always broadcast the low-order 16-bits of a load or store's address. In addition, we broadcast an extra bit that indicates whether the remaining bits are identical to those of the most recent store address. So long as there is sufficient locality in the types of memory references (e.g., stack versus heap), our PAM approach will herd most of the address broadcasts and comparisons to the top die. Our address memoization is inspired from instruction scheduler tag memoization [132], although we use memoization in a different context to target 3D power density.

6.3.6 Data Cache

The values in the data cache have similar data-width locality characteristics as the data values in the register files. As a result, we organize the data arrays of the L1 data cache as significance-partitioned 3D circuits. On the prediction of a low-width load, the processor accesses only the top die. The organization is analogous to the register file in that a small amount of extra state (memoization bits)

on the top die provides fast detection of unsafe width mispredictions. On an unsafe misprediction, we stall the pipeline. Since the tag match occurs in parallel with the misprediction detection, the processor knows the set-associative way of the cache hit and therefore only needs to access a single set-associative way when retrieving the remaining bits. A store in the commit stage already knows its data-width, and therefore stores will not cause unsafe width mispredictions when writing to the cache.

To increase the frequency of low-width values, we broaden the definition of a "low-width" value for load and store instructions. Instead of storing a single width memoization bit, we store two bits that encode the upper 48 bits. When the encoding bits are 00, that means the upper 48 bits are all zeros; 01 means the bits are all ones (encodes negative numbers); 10 means the upper bits are identical to the upper bits of the referencing address, which occurs when heap data structures store pointers to other nearby objects [30]; 11 means the upper bits cannot be trivially encoded using our 2-bit scheme and should be read from the remaining three die. Previous work on *frequent value locality* [160] has observed that there are frequently occurring data values. When we can ignore the lowest sixteen bits, the remaining *partial value* exhibits even stronger frequent value locality.

It is important to note that we only gate the bottom three die on a low-width predicted load or store. For a fill from or a spill to the L2 cache, we do not have a corresponding width prediction. Therefore, all spill/fill interactions between the L1 data cache and the L2 cache always access all four die.

6.3.7 Front-End

Since the functional blocks of the front-end do not deal with data values, a data-centric approach to activity partitioning may not be feasible here. Figure 67(a) shows a four-die 3D organization of the processor front end. We implement the instruction cache (I$) and instruction translation lookaside buffer (ITLB) using previously proposed 3D stacking organizations [109, 148] which provide latency and power benefits, but there is no explicit Thermal Herding.

After instructions enter the decoding pipeline, they will not move between die until they dispatch into the RS entries. We implement the register alias table (RAT) by placing the ports corresponding to each instruction on different die [111]. A single instruction's RAT read and write ports are all

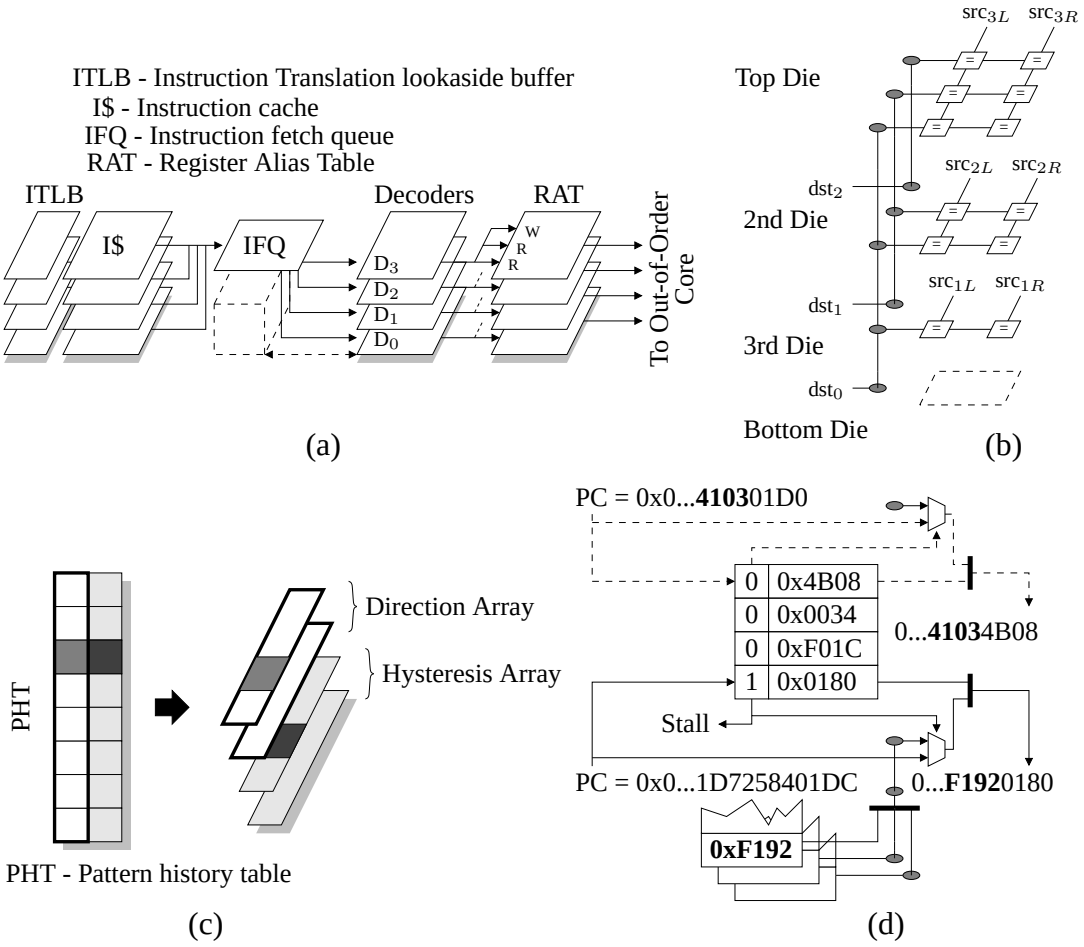

Figure 67: (a) 3D organization of the front-end pipeline components, (b) 3D register rename intra-group dependency checking logic, (c) branch predictor saturating counter array partitioned into two separate 3D sub-tables, and (d) Thermal Herding in the branch target buffer.

located on the same die as the decoded instruction itself, thereby avoiding the need for additional d2d vias (the two R's and one W in Figure 67(a) show these RAT ports for the top die).

Having instructions located on different die forces the intra-group rename dependency checking to use the d2d vias. However, we can partition the logic across multiple die to place more of the activity closer to the top die. A given instruction in a rename group only needs to check whether one of its input operands matches the output of a previous instruction in the same group. This implies that the first instruction does not require any dependency checks, and the last instruction requires the most checks. We place the instruction that requires the most register name comparisons on the top die, thereby herding more of the switching activity to the top die as illustrated by the dependency checking logic in Figure 67(b).

The last major parts of the front-end are the control-flow predictors. In particular, we consider the branch direction predictor and the branch target buffers. For the branch direction predictor based on two-bit saturating counters, we first partition the counters into two separate arrays: one array to store the direction bit (msb) and another to store the hysteresis bit (lsb) [131]. We implement the two arrays by partitioning them across two die each, as shown in Figure 67(c). The processor needs the direction bit for making the initial prediction as well as during the update/training phase, while the hysteresis bit is needed only during the update phase. Therefore, we place the more frequently accessed direction-bit array on the top two die closer to the heat-sink.

For the branch target buffers (BTBs), we observe that most branch targets are located relatively close to the originating branch itself. This is particularly true for PC-relative branch targets. Based on this observation, we organize the BTBs like our data cache, where we store the low-order sixteen bits on the top die along with one additional *target memoization* bit that indicates whether the bits on the remaining three die should be accessed. The top of Figure 67(d) shows a target prediction example (dashed lines) that reuses the higher order bits of the branch's program counter (PC). The target memoization bit causes the selection of the higher order PC bits. The bottom of Figure 67(d) shows an example (solid lines) of the infrequent case where the target memoization bit is set to one. In this case, we need to stall the prediction pipeline for one cycle to retrieve the most significant bits from the remaining three die of the array. Similar to reading the higher order 48 bits of the data cache, the BTB tag match in the first cycle enables the front-end to only access the single

set-associative way that had a hit.

6.3.8 Thermal Herding Microarchitecture Summary

The majority of our 3D components make use of a compact 3D organization to reduce wires which results in simultaneous latency and power reductions. When possible, we attempt to herd the switching activity to the die closest to the heat sink. We observed that 97% of all fetched instructions have their widths correctly predicted, thereby avoiding any severe performance degradation due to our microarchitectural Thermal Herding.

There are other 3D optimizations that yield IPC improvements. In particular, a 3D-integrated L2 cache results in a significantly faster access time which can reduce the number of clock cycles for an L2 access, even at a higher clock frequency. A variety of "global" signals may have shorter wire lengths in a 3D implementation because each of the components now has a reduced footprint, thereby compressing the overall processor floorplan. Some microarchitectures impose an extra clock cycle to load values into floating point registers due to the extra distance required to route from the cache to the floating point units [13, 157]. Another example is the extra pipeline stage(s) that may be necessary to communicate the branch misprediction from the execution stage in the back-end to the processor front-end [57]. The wire reduction due to the 3D-integarted circuits may sufficiently reduce latencies to remove this extra cycle.

6.3.9 Design Space Exploration

In the design of complex systems such as processors, evaluating the impact of design and technology parameters is important in order to come up with the optimal processor configuration [153] for the target technology. In the context of the 3D processors, when we stack the circuits, design and technology issues such as power density and leakage currents may become even more acute. As transistor geometries shrink with technology scaling, worsening relative wire delays might increase the performance and the power benefits of the 3D technology, but fabrication challenges such as the d2d via deposition, wafer alignment and mechanical bonding may prove to be a bottleneck in realizing the full potential of 3D-integration. In order to evaluate the impact of such technology factors, we vary the design parameters and the circuit parameters and explore the design space of the 3D-integrated processors. In particular, we consider the impact of total power consumption, 3D

die-to-die via density, and stacked die thickness on the 3D processors.

6.3.9.1 Impact of Increasing Power Consumption

Processor companies often create different versions of a baseline design to cater to different market segments. The different designs differ in their functionality (e.g., cache size, operating frequency), power consumption, and cost price. For example, the Intel dual core processor's desktop version has a power budget starting from 65W while the server version has a power budget starting from 80W [63, 62, 70, 126]. We explore the thermal impact of the 3D technology on different market segments (desktops versus servers) by experimenting with different power consumptions.

6.3.9.2 Impact of Technology Challenges

Some fabrication processes may pose challenges to the realization of the full potential of the 3D-integration technology. We focus on two of those challenges, namely die-to-die via density and wafer thinning process. Current die-to-die (d2d) via sizes are primarily limited by the accuracy of aligning the wafers prior to bonding. It is quite possible that the scaling of the d2d via sizes may not keep up with technology scaling, thereby increasing the size of the d2d vias relative to transistors over time. Hence, depending on the fabrication process, the usage of such vias could be limited to only non-critical path circuits to avoid performance degradation. We explore the impact of fabrication technology on the thermal profiles by varying the die-to-die via density.

One of the key fabrication procedures in the 3D-integration is the wafer thinning as explained in Section 1.3 of Chapter 1. We explore the effect on temperature, if the 3D die-stacks are not able to achieve the aggressive $\sim$10μm thickness due to future limitations of the fabrication processes.

6.4 Experimental Procedure

Our experimental analysis includes the quantification of performance in terms of clock frequency improvement, IPC rates, power consumption, and the overall thermal impact.

We use HSpice to simulate the processor components and determine their latencies and energy consumptions as explained in Part II.

For our processor configurations, we use SimpleScalar/MASE for the Alpha ISA [79, 6] to collect the access statistics for individual components. We use a collection of 106 application traces

including all benchmarks from SpecInt2000 and SpecFP2000 with the reference inputs, and a variety of programs from MediaBench [80], the Michigan embedded benchmarks [52], the Wisconsin pointer-intensive benchmarks [7], assorted graphics programs from the SimpleScalar website (includes games such as Doom and Quake, ray-tracing, and mpeg and avi video playback), and the BioBench [5] and BioPerf [8] bioinformatics benchmark suites. In all cases, we use SimPoint 2.0 to choose representative simulation points [107]. For each module, we compute the energy per access based on our HSpice results, and combine it with the access factor of the module reported by MASE, to calculate the total energy. We assume that the baseline planar processor dissipates 35% of its power in the clock network [24] and 20% in leakage. We assume that the clock network footprint is reduced by $\frac{1}{4}$ since it is distributed across four die, but we conservatively reduce its power consumption by $\frac{1}{2}$ for the 3D processor configurations. We assume that our 3D-integrated processors do not reduce the leakage energy.

For our thermal analysis, we use HotSpot (version 3.0.2) from the University of Virginia [136] as explained in Chapter 5. For the sake of temperature analysis, we assume that both the cores run the same application. This may be conservative since this means the worst case benchmark is running on both the cores at the same time.

6.5 Results

In this section, we present our experimental results to quantify the impact of our 3D Thermal Herding microarchitecture on performance, power, and temperature.

6.5.1 Performance

3D-integration can influence both the processor's clock frequency as well as its IPC rates. In the following sections, we discuss each of them in detail.

6.5.1.1 Clock Frequency

As discussed in Section 6.3, the 3D implementations of the processor's components reduce the wire-delay internal to those components. By 3D-integrating all of the processor's critical paths, we may be able to increase the overall clock frequency and gain more performance. Table 19 shows the latencies of some of the processor's components for both the planar and the 3D implementations.

Table 19: Critical path latencies for several microprocessor blocks. We consider the Wakeup-Select and the ALU-Bypass loops (in bold) to be the clock limiting paths.

Block/Path	Latency		
Name	Planar (ps)	3D (ps)	(% Reduction)
Wakeup+Select	**347.8**	**235.3**	**32.4%**
ALU+Bypass	**641.8**	**410.9**	**36.0%**
ROB/PRF	787.0	378.4	51.9%
Arch. RF	524.8	248.8	52.6%
L1 Cache	1019.5	707.9	30.6%
L2 Cache	4140.7	2008.7	51.5%
ITLB	663.1	369.9	44.2%
DTLB	788.8	504.6	36.0%
RAT	594.0	378.4	36.3%
Load Queue	345.8	253.1	26.8%

Previous work has identified the instruction scheduling logic (wakeup-select loop) and the arithmetic unit and result bypass loops (highlighted in bold in Table 19) to be particularly important in determining a processor's maximum clock frequency [106]. While we recognize that reducing the processor's cycle time may require speeding up hundreds or thousands of critical timing paths, we believe that these two fundamental loops are representative of the achievable speedups of a 3D microarchitecture. We observe a 32% improvement in the latency of the wakeup-select loop. This benefit derives from the reduction in the wire length and wire loading of the wakeup logic's tag broadcast bus as well as the wire delay in the select logic. Our 3D word-partitioned datapath results in a 36% latency improvement in the *ALU+Bypass* loop. The adder only accounts for 3% out of the 36% benefit because our 3D-partitioned adder only reduces the wire delay of the last levels of the carry logic. However, the compaction of the ALUs results in a substantial reduction in the distance traversed by the bypass network. Overall, wire latency reduction translates into a 47.9% increase in clock frequency, from 2.66 GHz to 3.94 GHz.

While we base our overall clock frequency on the two critical loops, Table 19 demonstrates that the 3D-integration provides substantial latency improvements across a wide variety of other components. As demonstrated in Chapter 2 and prior research on 3D SRAM designs [120, 148, 109, 95, 111], we observe that large components (caches, register files, TLBs) observe substantial latency improvements. Most of these components can be fairly easily pipelined and therefore we do not consider them as frequency limiters.

It is important to understand that the clock frequency increase provided by our Thermal Herding 3D microarchitecture is directly from reducing the wire delay component of the cycle time, whereas the conventional microarchitectural approach for increasing clock frequency is to increase the number of pipeline stages. Adding more pipeline stages increases pipeline complexity (e.g., requiring more inter-stage bypassing) and it usually decreases IPC (e.g., needing aggressive speculative scheduling).The frequency benefits of the 3D technology do not require changes to the overall pipeline organization at the microarchitecture level. With careful design, 3D may be able to *remove* pipeline stages as described in Section 6.3.8 of this chapter, thus providing a microarchitecturally simpler pipeline organization *and* higher clock speeds.

To put this frequency benefit in perspective, we matched Intel Pentium 4 processor clock speeds against the respective official SPEC score reporting dates[1] and interpolated an average frequency increase of 17% per Moore's Law generation (18 months), which extrapolates to $\sim$37% for two generations. A four-die stack provides the device density of two technology generations into the future, and our result of a $\sim$33% frequency boost indicates that the 3D-integration can provide the same approximate level of speed increase as the traditional technology scaling. This comparison provides a useful "order-of-magnitude" comparison and demonstrates that the 3D-integration technology can match the frequency benefits of a conventional planar technology that is two generations into the future.

6.5.1.2 IPC and Overall Performance

Different features of our Thermal Herding 3D (TH-3D) microarchitecture affect overall IPC in different ways. For example, our TH-3D processor can improve IPC by reducing the pipeline depth and reducing the L2 latency in clock cycles (in addition to the frequency increase). However, our Thermal Herding microarchitecture can reduce IPC rates due to the different width-misprediction related stalls and increases in the average number of *cycles* to access main memory due to the frequency increase.

We explore various configurations to better understand the performance benefits and the IPC impact. Figure 68 shows the geometric mean IPC rates for each of our benchmark classes, and

[1] From Q1 2002 to Q4 2004, covering the 130nm and 90nm technology nodes.

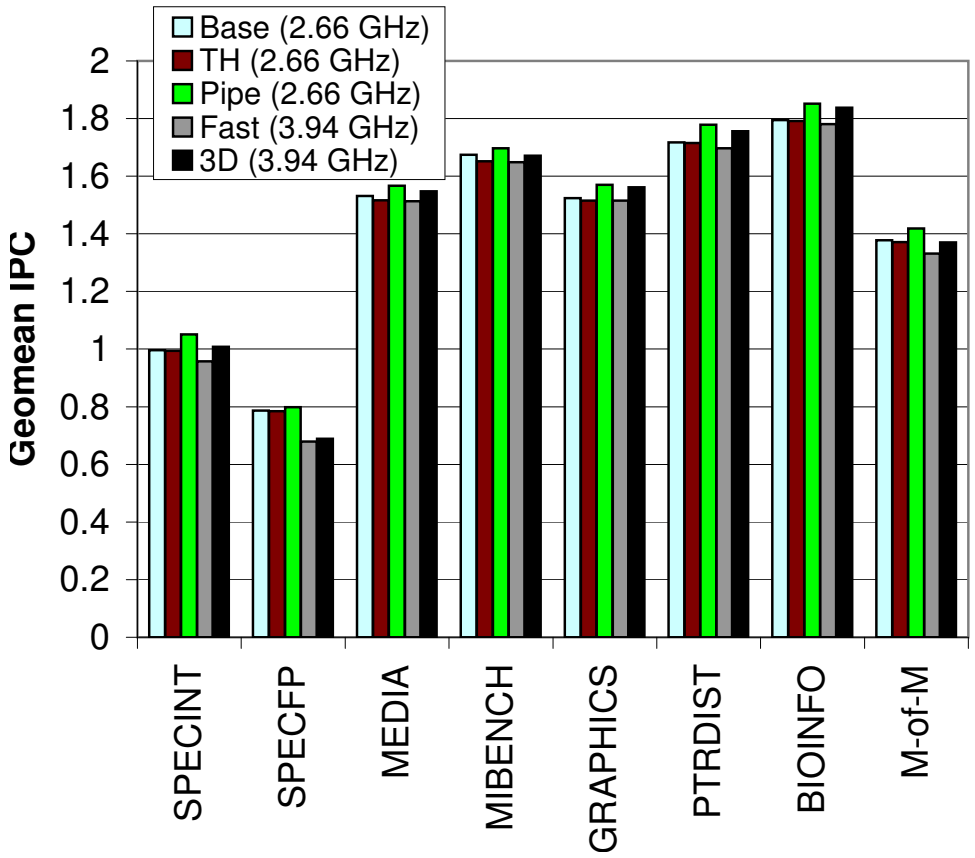

Figure 68: IPC results of the planar and the 3D processors. M-of-M is the "mean of means" (geometric mean across all benchmark groups).

the overall mean of the per-group means (M-of-M). Base is the baseline (2.66 GHz) planar processor, TH applies the data-width prediction technique and Pipe applies the pipeline optimizations described earlier in Section 6.3. For both TH and Pipe, we do not change the clock frequency to isolate the IPC impact directly attributable to these changes. The Fast configuration is microarchitecturally identical to the baseline planar processor, but the higher clock frequency (3.94 GHz) increases the average number of cycles to access main memory which in turn decreases IPC. Finally, the IPC rates for 3D are for our 3D Thermal Herding processor that simultaneously accounts for the impact of our Thermal Herding, pipeline optimizations, and increased clock frequency.

Overall, the pipeline reduction benefits slightly outweigh the effects of width mispredictions and the higher clock speed, and provides a small IPC benefit. However, overall performance is determined by both IPC rates and the clock frequency. Figure 69(a) shows the overall performance in instructions per nanosecond (IPns) that accounts for both of these factors, and Figure 69(b)

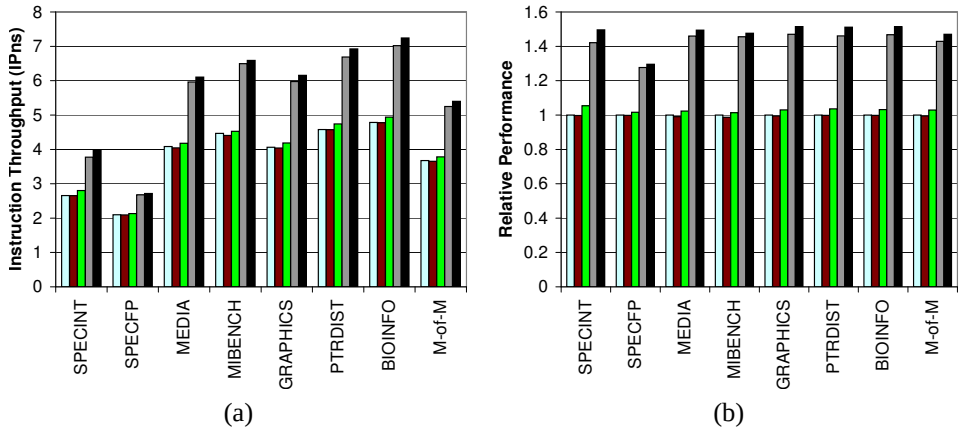

Figure 69: Performance impact of our Thermal Herding 3D techniques (a) throughput in instructions per nanosecond, and (b) overall performance speedup. M-of-M is the "mean of means".

shows the relative performance speedup over the baseline. Because the pipeline optimizations cancel out the IPC degradation of increased clock speeds, performance tends to scale directly with the overall frequency improvement of our 3D processor. Performance improvements ranged from 7% (mcf/SPECInt2000) to 65% (crafty/SPECInt2000) and 77% (patricia/MiBench). Except for SPECFP2000, all of the other benchmark groups exhibit 49.4% to 51.5% mean performance improvements. Our 3D processor provides only a 29.5% benefit for SPECFP2000 because these benchmarks have a large number of accesses to the main memory, and our 3D processor has not reduced the latency (in seconds) of DRAM accesses. One could potentially use 3D-integration to further stack DRAM on top of our already 3D-stacked processor to help reduce main memory latency [88].

6.5.2 Power Consumption

Column I in Table 20 lists five benchmarks that consume the most power among all benchmark suites and Column II shows the respective power consumption of those benchmarks. The power consumption reflects the frequency of access and the switching activity of various components in the processor. Column III in Table 20 shows the power consumed by our Thermal Herding 3D processor. Our Thermal Herding 3D processor provides overall power reduction in two ways: first, the significance-partitioned 3D implementation of the processor components substantially reduces the amount of wires in these components; second, data-width-locality based prediction scheme allows

Table 20: Most power-consuming benchmarks on our planar baseline, and the corresponding power consumed by our Thermal Herding 3D processors

Benchmark	Planar power (W)	Thermal Herding power (W)	Thermal Herding %
mpeg2encode-matrix	90.1	64.3	28.6
susan-smoothing	85.9	60.0	30.2
clustalw	81.5	61.4	24.7
phylip-dnapenny	80.2	59.6	25.7
adpcm-dec	80.2	57.0	28.9

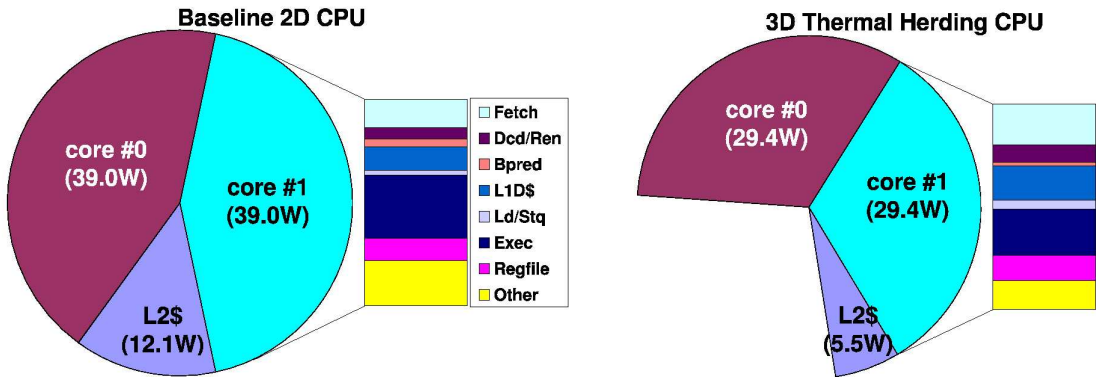

Figure 70: Power consumption distribution (Mpeg2 encoding/MediaBench) of the (a) baseline planar processor, and (b) our Thermal Herding 3D processor.

our TH-3D processor to sometimes reduce approximately 75% of a component's switching activity. However, when we increase clock frequency to improve performance, we also increase power. Note that our Thermal Herding 3D processor provides ∼28% average power saving compared to the planar processor.

For the baseline planar processor, the Mpeg2 encoding benchmark from the MediaBench suite [80] resulted in the highest power consumption of 90.1W. Since the coarse-grained 3D (CG-3D) processor does not take advantage of the 3D technology to either increase the clock frequency of the cores or reduce their power consumptions, it dissipates a similar amount of power as the planar processor for each of the benchmarks. Despite the frequency increase, our Thermal Herding 3D microarchitecture aggressively reduces wires and herds the switching activity, resulting in a power reduction from 90.1W to 64.3W. Note that our Thermal Herding 3D processor consumes less power than the planar processor even while running at a higher frequency because it reduces the overall wire-related resistance and capacitance at the level of individual modules as well as between the modules. In

fact, if the 3D processor were to be operate at the same frequency as the planar processor, we can gain additional power benefits. The total power saving for our TH-3D processor over the planar baseline ranges from 15.0% (Yacr2 benchmark from the pointer applications [7]) to 30.2% (Susan image processing benchmark applying a smoothing filter from MiBench [52]) depending on the characteristics of the application. Applications such as Yacr2 are memory-intensive and thus derive less benefit from our TH-3D processors. Image processing applications are computation-intensive and thus derive larger benefit from our Thermal Herding 3D processors. In the rest of the results section, we report the peak temperatures based on the benchmarks listed in Table 20.

Figure 70(a) shows the power distribution for the Mpeg2 encoder application, running on each of the two processor cores (90.1W total). Each of the two planar cores consumes 39W and the L2 cache consumes 12.1W of power. Figure 70(b) shows the power distribution for our Thermal Herding 3D processor. Note that our Thermal Herding 3D processor provides a large power benefit in the L2 cache (from 12.1W to 5.5W) due to the extensive wire-reduction in the gigantic L2 structure by the 3D circuit partitioning. Note that most of the saving is from the significance-partitioning of the L2 cache, since there is no data-width locality based gating in the L2 cache. Each of the cores experience a reduction in power from 39.0W each to 29.4W each. The saving in each of the cores is due to both significance-partitioning and correctly predicted low-width accesses on the datapath.

Figure 71(a) shows the percentage of accesses that were low-width and full-width for several microarchitectural blocks. Each low-width access consumes less total power and isolates the power to the top die (or the top two die in the case of the branch predictor lookup). Except for the branch target buffer (BTB) addresses and the store queue (SQ) addresses, the general trend is that approximately one half of all activity requires accessing only the top die. Figure 71(b) shows the activity breakdown for some of the blocks that are partitioned based on instructions rather than data. For the decoder and the register alias table (RAT) in the frontend, Thermal Herding provides only a slight shift in the distribution toward the top die. However for the reservation station (RS) entries in the back-end, the top-die-first allocation policy is able to keep one third of all RS tag broadcasts and comparisons corralled to the top die, and limit the bottom die activity to about 15% (compared to an expected 25% if instructions were uniformly distributed across all four die). The combination of reduced power per access and reduced access frequency both play an important role in controlling

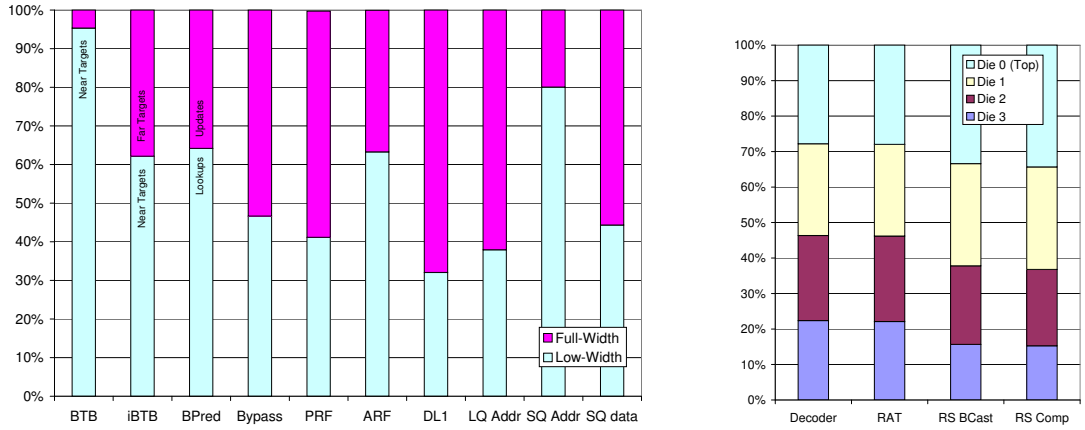

Figure 71: (a) Distribution of full-width versus low-width accesses of various components, and (b) per-die distribution of activity.

thermals in a 3D processor. Li et al. [83] describe how reducing total power can have a substantial effect on reducing chip temperature.

6.5.3 Temperature

The 3D-integrated processor can potentially suffer from thermal challenges due to the stacking of devices as well as the fact that there is less contact with the heat spreader/heat-sink surface for removing heat from the processor.

Figure 72(a-b) shows the thermal maps for the applications that induced the worst-case temperatures among our 106 application traces. Different applications were responsible for these worst-case scenarios (Mpeg2 for planar; Yacr2 for Thermal Herding). Figure 72 includes annotations for the locations and magnitudes of some notable hotspots, with the worst-case hotspot marked by a box around the temperature. Figure 72(a) shows the baseline planar processor, which has a peak temperature of 360K located at the instruction scheduling logic (RS entries). Figure 72(b) shows our Thermal Herding 3D processor which has the hottest spot at 372K in the data cache area (only 12K increase from the planar baseline). Since Yacr2 is a memory-intensive application, the high frequency of data cache accesses combined with additional cache accesses to handle width mispredictions causes the data cache to become the hottest spot.

The overall worst-case temperatures determine the cooling requirements for the system design. However, more insights can be gained by examining the thermal behavior of the processors across a

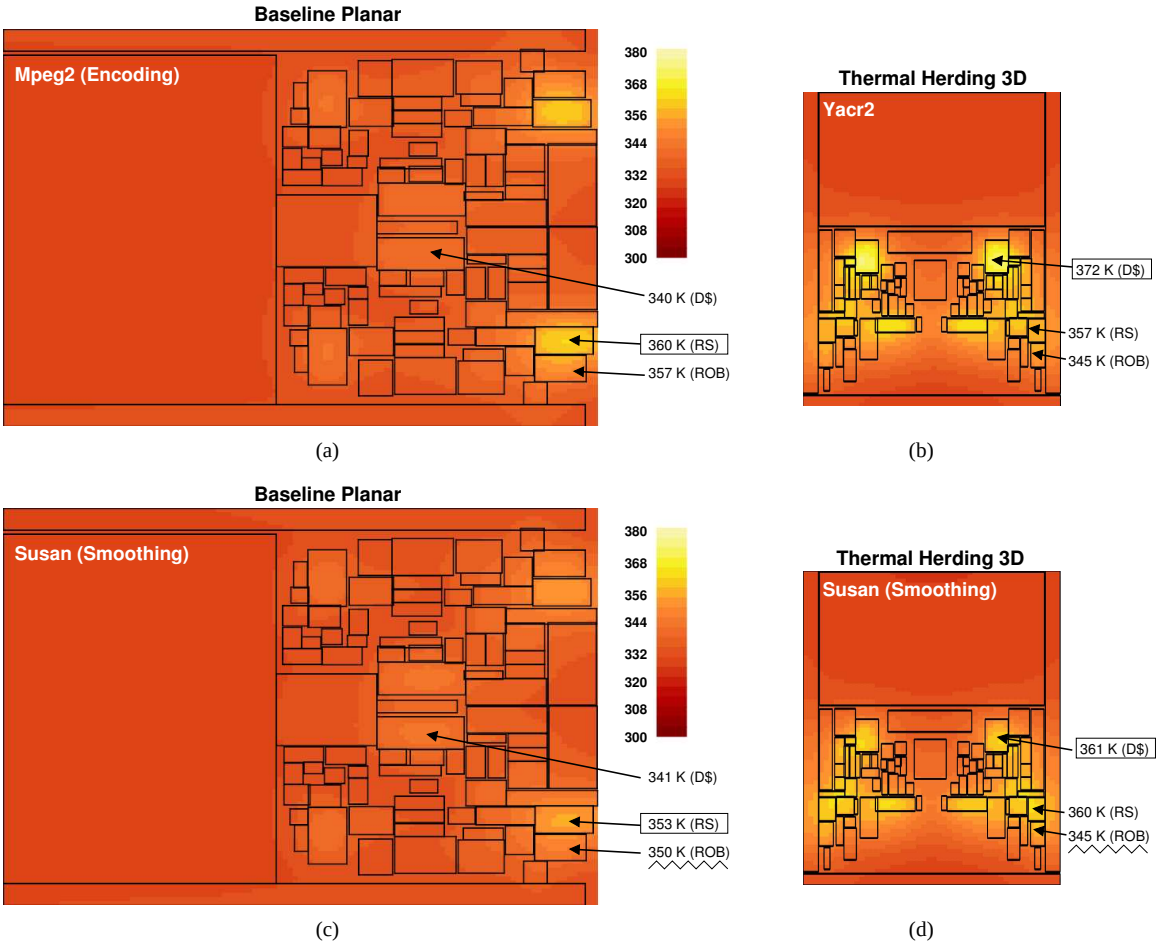

Figure 72: Worst-case thermal plots of (a) the baseline planar processor, and (b) the Thermal Herding 3D processor. (c-d) Thermal plots of the planar and the Thermal Herding 3D processors for the Susan benchmark from MiBench. Boxes indicate hottest blocks.

single benchmark. Figure 72(c-d) shows the same processor configurations, except that each one is now running the same application. For several components, our Thermal Herding techniques provide a substantial reduction in the temperatures. This is largely due to the overall power benefit from our Thermal Herding techniques. In fact, we observed some components in our Thermal Herding 3D processor to have a lower local temperature than the baseline planar processor. For example, the reorder buffer (ROB) which contains the physical registers exhibits a large number of low-width accesses (approximately 5× (2×) more low-width reads (writes) than full-width reads (writes)), resulting in 44% of the ROB's power being isolated to the top die. The net result is a 5K temperature reduction for the ROB over the planar baseline. Note that Black et al. [13] have demonstrated that we can further reduce the temperature by converting a small part of our performance gains into power benefit.

6.5.4 Design Space Exploration

6.5.4.1 Impact of Increasing Power

We investigate the 3D-integration benefits for different processor market segments, (e.g., desktop versus server implementations of the processor) by varying the total power consumption. We increase the total power consumption from 90W to 150W as shown in Table 21 and report the worst case temperatures across all our benchmarks.

As the total power increases from 90W to 150W, the peak temperature on the planar processor rises from 359.9 K to 391.2 K, an increase of 31.3 K. In case of the TH-3D (CG-3D) processor, the peak temperature increases by 39.4 K (47.7 K). Most of the temperature increase in the 3D processors is due to a combination of increased thermal resistances (die further from the heat sink) and reduced area of contact with the heat sink.

Overall, the thermal trends of our 3D processors remain similar to that of the planar processor. This shows that the 3D processor benefits scale favorably with increasing power budgets and do not worsen, in comparison to the planar processors.

As the functionality increases, the peak temperatures of both the coarse-grained 3D (CG-3D) processor and our Thermal Herding 3D (TH-3D) processor increase faster than those of the planar

Table 21: Total power consumption versus peak temperature

Total Power (W)	Planar peak temperature (K)	CG-3D peak temperature (K)	TH-3D peak temperature (K)
90	359.9	384.8	372.3
100	365.2	392.8	378.8
120	375.6	408.6	392.0
150	391.2	432.5	411.7

processor. The temperature degradation is higher in the 3D-integrated processors due to the increasing power density and reduced surface area for heat sink contact as compared to the planar processor. Both the CG-3D processor and the TH-3D processor experience increased temperatures due to the stacking of actively switching circuits right on top of each other. However, our TH-3D processor does not experience as high a temperature increase as the CG-3D processor due to the power benefit. Note that the CG-3D processor does not derive power benefit from the 3D-integration due to stacking planar components on top of each other. The peak temperature on our TH-3D processor is atleast *+12K* less than the peak temperature on the corresponding CG-3D processor and the difference keeps increasing with higher power configurations. Our Thermal Herding 3D processor achieves better thermal profiles than the CG-3D processor in addition to significant improvements in both the performance and the power.

To understand the interaction between the 3D-integration and the temperature, we performed thermal analysis of a 3D processor that runs at the higher 3D frequency (3.94 GHz) but does not provide energy benefits. Note that this 3D configuration mimics a high increase in the power density due to both increased frequency and power. The worst case temperature increased to 418K (+58 degrees over the baseline planar processor). Our 3D processors are not as hot due to the decrease in the *total* power consumption, which provides a large relief to the increase in the power density.

6.5.4.2 Impact of D2D Via Density

We consider the via density of the d2d vias and their potential impact on the peak temperature. Table 22 shows the effect of decreasing the via density from fully populated (100%) d2d vias to one quarter (25%) of the via density. As can be seen from Table 22, the peak temperature on either the CG-3D or the TH-3D does not vary much, except for the CG-3D temperature being consistently higher by about 12 K compared to the TH-3D temperature. The d2d vias are required to bond the

Table 22: Density of d2d vias versus peak temperature (original 3D-integrated processors assume fully-populated via density)

Configuration	100% populated	75% populated	50% populated	25% populated
CG-3D	384.8	384.9	385.0	385.2
TH-3D	372.3	372.4	372.6	372.9

Table 23: Die thinning versus peak temperature (original 3D-integrated processors have a die-thickness of 9μm)

Die thickness	CG-3D	TH-3D
(μm)	peak temperature (K)	peak temperature (K)
9	384.8	372.3
18	383.1	370.4
36	381.2	368.0
72	377.0	364.3

die together and hence need to be populated to the maximum extent.

6.5.4.3 Impact of Wafer Thinning

One of the key manufacturing procedures in 3D fabrication is the wafer thinning process explained in Section 1.3 of Chapter 1. We explore the effect on temperature, if the 3D die-stacks are not able to achieve the 9μm thickness due to fabrication process limitations in the future. Table 23 shows the peak temperatures for thicker die on the 3D-stack with thicknesses ranging from 9μm to 72μm at the d2d interfaces. As the die thickness increases, we observe a slight reduction in the worst-case temperatures (measured across all benchmarks). Even though a thicker die increases the thermal resistance on the path from the die to the heat-sink, it also increases the distance between the stacked devices in the vertical dimension and hence alleviates some power density issues. However, increasing die thicknesses have other implications in terms of increasing the d2d via latency and hence reducing the performance benefits. Overall we observe a slight reduction in the peak temperatures for increasing die thicknesses. Since the temperature impact is within a few degrees for the thinner wafers as compared to the thicker wafers, it is advisable to thin the wafers to the maximum possible extent to obtain the maximum performance.

6.6 Summary of the Thermal Herding 3D-Integrated Processors

The 3D-integration technology has the ability to increase transistor density and perhaps extend Moore's Law for a few more technology generations. Through detailed circuit analysis, we demonstrated that conventional high-performance processors implemented in the 3D-integration technology can provide simultaneous benefits in latency and power, resulting in substantial performance benefits. We demonstrated that our microarchitecture-level techniques can control power density and mitigate 3D thermal issues. The 3D-integration technology provides more performance for less power, thereby providing excellent performance-per-Watt ratios. We showed our Thermal Herding 3D processors to have frequency, power and thermal advantages over the coarse-grained 3D processors. We explored the impact of various parameters such as total power, die thickness, and via density on the peak temperature of the chip. We demonstrated that the temperatures on the 3D-integrated processors can be effectively controlled using microarchitectural techniques.

In this dissertation, we have demonstrated that the 3D-integrated high-performance microprocessors can provide significant value. However, at the heart of this study lies a conventional ROB/RS-based microarchitecture originally designed for a planar fabrication technology.

Other non-traditional architectures such as TRIPS [127] or WaveScalar [143] may also be interesting candidates for 3D implementations. Apart from adapting conventional microarchitectures to a 3D-technology, designing a new processor from the ground-up to exploit the 3D technology may provide even greater performance and power benefits. A new microarchitecture designed from the ground up with the 3D-integration in mind may be much more effective at exploiting the strengths of the 3D-integration.

There are many possibilities for 3D-integrated processor design and there is a great need for more 3D microarchitecture research.

CHAPTER VII

CONCLUSIONS

The 3D-integration technology has the potential to simultaneously address many of the challenges faced by the semiconductor industry. By placing the microprocessor circuits in stacked layers and providing vertical connectivity with short interconnects, the 3D-integration technology greatly reduces wire lengths, and the resulting delay and power consumption.

We proposed 3D-integrated designs of various microprocessor components, such as caches, register files, instruction schedulers, and arithmetic units, based on partitioning the logic and/or the wiring. We proposed designs of high-performance 3D-integrated microprocessors and evaluated the impact on frequency, power, and temperature. 3D-integrated microprocessor designs based on the Alpha 21364 processor demonstrated two different approaches to improve performance (improved speed and improved functionality). 3D-integrated microprocessor designs based on the Intel Core microarchitecture proposed microarchitectural techniques to address the challenges of power density and mitigate the temperatures.

In this dissertation, we have demonstrated that the 3D-integrated high-performance microprocessors can provide significant value. Our experimental analysis of performance, power and temperature of 3D-integrated circuits has provided evidence in support of our thesis that the 3D-integration provides simultaneous performance and power benefits to build high-performance microprocessors, while keeping the worst-case temperature under control. The simultaneous benefits in multiple objectives makes the 3D-integration a highly desirable technology for use in designing future high-performance processors. One of the key contributions of this dissertation is the temperature analysis that shows that the worst-case temperature on the 3D-integrated processors can be effectively controlled with microarchitectural techniques.

Small footprints and power reductions offered by the 3D-integration technology can give rise to more functional and stylishly-sleek products such as cellphones and PDAs in the embedded industry. Various 3D-integration approaches may be combined with each other to offer flexible heterogeneous

integration and/or higher functionality. From the perspective of the embedded industry, the 3D-integration technology can be viewed as an enabler for increased integration, lower power, and smaller form factors.

3D-integration also provides faster operating speeds and higher bandwidths than the current technology generations. From the perspective of the high-performance processor industry, 3D-technology can be viewed as a means of breaking the processor-memory wall. With the rapid proliferation of the multi-core processors, interconnects will become crucial in deciding the performance of the multi-core systems, thus increasing the value and the inevitability of the 3D-integration technology.

Some of the prerequisites for the widespread deployment of the 3D technology include design automation tool support and design flows, yield analysis of high-volume manufacturing processes, and economic feasibility of the additional manufacturing steps. The 3D-integration technology may extend the applicability of Moore's law for a few more technology generations. We believe that this is just the beginning of the 3D revolution.

APPENDIX A

ELECTRIC MODELS OF TRANSISTORS AND WIRES

Figure 73(a) shows a logic gate driving a fanout load through an interconnect. Figure 73(b) shows the corresponding circuit abstraction, that models the driver driving the fanout gate through the interconnect. The driving gate is represented as a voltage source with an output resistance R_{tr}. The interconnect is represented as a distributed RC line, with r and c as resistance and capacitance per unit length and l as the length of the wire. The fanout gate is represented as a lumped capacitance C_L. Using the Sakurai model [149], the overall delay, D is given by Equation 4. From Equation 4, we can see that the interconnect RC delay dominates for long interconnections due to the quadratic dependence of the delay on the length l of the interconnect. Long interconnections are usually avoided by inserting repeaters to eliminate the quadratic length dependence and to regenerate the signal.

$$D = 0.4(r \times c \times l^2) + 0.7(R_{tr} \times C_L + R_{tr} \times c \times l + C_L \times r \times l) \quad (4)$$

Figure 74(a) identifies the physical dimensions of two adjacent interconnects, namely, length l, thickness t, and width w, and describes the calculation of the interconnect resistance R. Figure 74(b) identifies the dimensions of three adjacent interconnect layers and identifies the sources of the interconnect capacitance C.

With technology scaling, width w of the interconnect scales down with the technology factor.

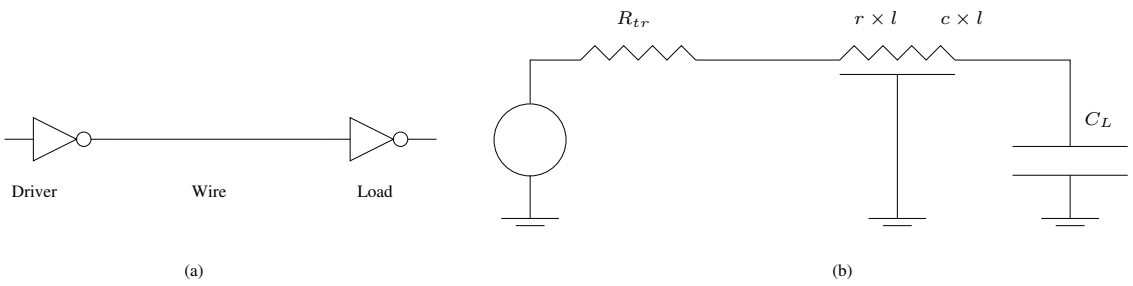

Figure 73: (a) Circuit (b) Circuit electrical model.

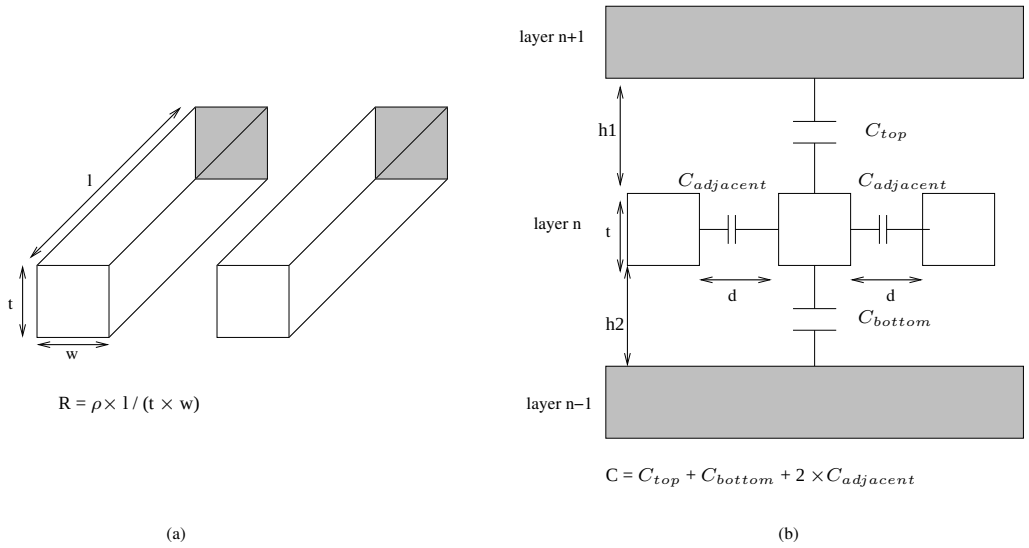

Figure 74: Interconnect layer model.

The interconnect resistance is inversely proportional to the width, hence, scaling of the interconnect width makes the resistance R larger. Another effect of the technology scaling is decreased spacing d between the interconnects and, therefore, increased coupling capacitance. The coupling capacitance adds to the interconnect capacitance C, and increases cross-talk between the coupled interconnect lines.

Overall, as the feature size is scaled down, the interconnect RC-delay does not scale with feature size. As a result, while transistor sizes and speeds continue to improve, the interconnect delay contribution becomes a significant component of the overall delay.

APPENDIX B

DYNAMIC AND STATIC POWER CONSUMPTION

B.1 Dynamic Power Consumption

The total dynamic power dissipation P_{dyn} of an integrated circuit is calculated using Equation 5, where C is the total capacitance, V is the supply voltage, f is the frequency of operation, and α is the activity factor.

Increased number of actively switching transistors, increased wiring complexity, and a higher frequency of operation together cause a significant increase in the dynamic power consumption.

$$P_{dyn} = \frac{1}{2} C V^2 f \alpha \tag{5}$$

B.2 Static Power Consumption

The total static power dissipation P_{stat} of an integrated circuit is calculated using Equation 6.

$$P_{stat} = \sum V \times I_{lkg} \tag{6}$$

With each technology scaling, the supply voltage has to be scaled down to maintain constant electric field from one generation of integrated circuit devices to the next in order to maintain the performance improvement. The reducing supply voltages require a corresponding reduction in threshold voltages. This decrease in the threshold voltage results in an exponential increase in the subthreshold leakage current. Leakage power has already become comparable to dynamic power, and it is projected to dominate the total chip power in nanometer scale technologies. As a result, the total power consumption of the integrated circuits has been increasing rapidly, making power to be a major concern in modern design.

REFERENCES

[1] ABABEI, C., MOGAL, H., and BAZARGAN, K., "Three-dimensional place and route for fpgas," in *ASP-DAC '05: Proceedings of the 2005 conference on Asia South Pacific design automation*, (Shanghai, China), pp. 773–778, 2005.

[2] ABU-SHAMA, E., MAAZ, M. B., and BAYOUMI, M. A., "A fast and low power multiplier architecture," in *Proceedings of IEEE Midwest symposium on Circuits and Systems*, pp. 53–56, 1996.

[3] AGARWAL, V., HRISHIKESH, M. S., KECKLER, S. W., and BURGER, D., "Clock Rate Versus IPC: The End of the Road for Conventional Microarchitectures," in *Proceedings of the 27th International Symposium on Computer Architecture*, (Vancouver, Canada), pp. 248–259, June 2000.

[4] AHN, J.-H. and JEONG, J.-S., "Hierarchical word line structure." United States Patent Application, February 17 1998.

[5] ALBAYRAKTAROGLU, K., JALELL, A., WU, X., FRANKLIN, M., JACOB, B., TSENG, C.-W., and YEUNG, D., "BioBench: A Benchmark Suite of Bioinformatics Applications," in *Proceedings of the International Symposium on Performance Analysis of Systems and Software*, (Austin, TX, USA), pp. 2–9, March 2005.

[6] AUSTIN, T., LARSON, E., and ERNST, D., "SimpleScalar: An Infrastructure for Computer System Modeling," *IEEE Micro Magazine*, pp. 59–67, February 2002.

[7] AUSTIN, T. M., BREACH, S. E., and SOHI, G. S., "Efficient Detection of All Pointer and Array Access Errors," in *Proceedings of the SIGPLAN Conference on Programming Language Design and Implementation*, (Orlando, FL, USA), pp. 290–301, June 1994.

[8] BADER, D. A., LI, Y., LI, T., and SACHDEVA, V., "BioPerf: A Benchmark Suite to Evaluate High-Performance Computer Architecture of Bioinformatics Applications," in *Proceedings of the*, pp. 163–173, 2005.

[9] BALASUBRAMONIAN, R., DWARKADAS, S., and ALBONESI, D., "Reducing the Complexity of the Register File in Dynamic Superscalar Processors," in *Proceedings of the 34th International Symposium on Microarchitecture*, (Austin, TX, USA), pp. 237–248, December 2001.

[10] BANNON, P., "Alpha 21364: A Scalable Single-chip SMP," *Microprocessor Forum*, October 1998.

[11] BEDDINGFIELD, C. and KOST, D., "Prediction Caches for Superscalar Processors," in *Proceedings of the 47th Electronic Components and Technology Conference*, (San Jose, CA), pp. 643–648, May 1997.

[12] BJORKHOLM, J. E., "EUV Lithography - The Successor to Optical Lithography?," *Intel Technology Journal*, Q3 1998.

[13] BLACK, B., ANNAVARAM, M., BREKELBAUM, N., DEVALE, J., JIANG, L., LOH, G. H., MCCAULEY, D., MORROW, P., NELSON, D. W., PANTUSO, D., REED, P., RUPLEY, J., SHANKAR, S., SHEN, J., and WEBB, C., "Die Stacking (3D) Microarchitecture," in *Proceedings of the International Symposium on Microarchitecture*, (Orlando, FL), December 2006.

[14] BLACK, B., NELSON, D., WEBB, C., and SAMRA, N., "3D Processing Technology and its Impact on IA32 Microprocessors," in *Proceedings of the 22nd International Conference on Computer Design*, (San Jose, CA, USA), pp. 316–318, October 2004.

[15] BLACK, B., NELSON, D., WEBB, C., and SAMRA, N., "3D Processing Technology and its Impact on IA32 Microprocessors," in *Proceedings of the 22nd International Conference on Computer Design*, (San Jose, CA, USA), pp. 316–318, October 2004.

[16] BOOTH, A. D., "A signed binary multiplication technique," in *Quarterly Journal of Mathematics*, vol. 4, 1951.

[17] BORCH, E., MANNE, S., EMER, J., and TUNE, E., "Loose Loops Sink Chips," in *Proceedings of the 8th International Symposium on High Performance Computer Architecture*, pp. 299–310, 2002.

[18] BORKAR, S., "Design Challenges of Technology Scaling," *IEEE Micro Magazine*, vol. 19, pp. 23–29, July 1999.

[19] BOWMAN, K., DUVALL, S., and MEINDL, J., "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 2, pp. 183 – 189, Feb 2002.

[20] BREKELBAUM, E., II, J. R., WILKERSON, C., and BLACK, B., "Hierarchical Scheduling Windows," in *Proceedings of the 35th International Symposium on Microarchitecture*, (Istanbul, Turkey), pp. 27–36, November 2002.

[21] BRENT, R. P. and KUNG, H. T., "A Regular Layout for Parallel Adders," pp. 260–264, March 1982.

[22] BROOKS, D., COOK, P. W., BOSE, P., SCHUSTER, S. E., JACOBSON, H., KUDVA, P. N., BUYUKTOSUNOGLU, A., WELLMAN, J.-D., ZYUBAN, V., and GUPTA, M., "Power-Aware Microarchitecture: Design and Modeling Challenges for Next-Generation Microprocessors," *IEEE Micro Magazine*, vol. 20, pp. 26–44, November 2000.

[23] BROOKS, D. and MARTONOSI, M., "Dynamically Exploiting Narrow Width Operands to Improve Processor Power and Performance," in *Proceedings of the 5th International Symposium on High Performance Computer Architecture*, (Orlando, FL, USA), pp. 13–22, January 1999.

[24] BROOKS, D., TIWARI, V., and MARTONOSI, M., "Wattch: A Framework for Architectural-Level Power Analysis and Optimizations," in *Proceedings of the 27th International Symposium on Computer Architecture*, (Vancouver, Canada), pp. 83–94, June 2000.

[25] BROWN, M. D., STARK, J., and PATT, Y. N., "Select-Free Instruction Scheduling Logic," in *Proceedings of the 34th International Symposium on Microarchitecture*, (Austin, TX, USA), pp. 204–213, December 2001.

[26] CAO, Y., SATO, T., SYLVESTER, D., ORSHANSKY, M., and HU, C., "New Paradigm of Predictive MOSFET and Interconnect Modeling for Early Circuit Design," in *Proceedings of the 2000 Custom Integrated Circuits Conference*, (Orlando, FL, USA), pp. 201–204, May 2000.

[27] CHENG, L., DENG, L., and WONG, M., "Floorplan Design for 3-D VLSI Design," in *Proceedings of the Asia South Pacific Design Automation Conference*, (Shanghai, China), January 2005.

[28] CHIANG, T.-Y., BANERJEE, K., and SARASWAT, K. C., "Compact Modeling and SPICE-Based Simulation for Electrothermal Analysis of Multilevel ULSI Interconnects," in *Proceedings of the International Conference on Computer-Aided Design*, 2001.

[29] CHIANG, T.-Y., SOURI, S. J., CHUI, C. O., and SARASWAT, K. C., "Thermal analysis of heterogeneous 3D ICs with various integration scenarios," in *International IEDM Technical Digest*, pp. 31.2.1–31.2.4, 2001.

[30] COLLINS, J., SAIR, S., CALDER, B., and TULLSEN, D. M., "Pointer Cache Assisted Prefetching," in *Proceedings of the 35th International Symposium on Microarchitecture*, (Istanbul, Turkey), pp. 62–73, November 2002.

[31] CONG, J., JAGANNATHAN, A., MA, Y., REINMAN, G., WEI, J., and ZHANG, Y., "An Automated Design Flow for 3D Microarchitecture Evaluation," in *Proceedings of the 11th Asia South Pacific Design Automation Conference*, pp. 384–389, 2006.

[32] CONG, J. and ZHANG, Y., "Thermal-Driven Multilevel Routing for 3-D ICs," in *Proceedings of the Asia South Pacific Design Automation Conference*, (Shanghai, China), January 2005.

[33] CONG, J. and ZHANG, Y., "Thermal Via Planning for 3-D IC's," in *Proceedings of the International Conference on Computer-Aided Design*, pp. 745–752, 2005.

[34] CORPORATION, S. P. E., "SPEC CPU 2000." from `http://www.spec.org/cpu/`.

[35] CRC PRESS, "CRC Handbook of Chemistry." `http://www.hbcpnetbase.com/`.

[36] DAS, S., FAN, A., CHEN, K.-N., and TAN, C. S., "Technology, Performance, and Computer-Aided Design of Three-Dimensional Integrated Circuits," in *Proceedings of the International Symposium on Physical Design*, (Phoenix, AZ, USA), pp. 108–115, April 2004.

[37] DAVIS, W. R., WILSON, J., MICK, S., XU, J., HUA, H., MINEO, C., SULE, A. M., STEER, M., and FRANZON, P. D., "Demystifying 3d ics: The pros and cons of going vertical," *IEEE Des. Test*, vol. 22, no. 6, pp. 498–510, 2005.

[38] DE, V. and BORKAR, S., "Low power and high performance design challenges in future technologies," in *GLSVLSI '00: Proceedings of the 10th Great Lakes symposium on VLSI*, (Chicago, Illinois, United States), pp. 1–6, 2000.

[39] DENG, Y. and MALY, W., "2.5D System Integration: A Design Driven System Implementation Schema," in *Proceedings of the Asia South Pacific Design Automation Conference*, (Yokohama, Japan), pp. 450–455, January 2004.

[40] ERGIN, O., BALKAN, D., GHOSE, K., and PONOMAREV, D., "Register Packing: Exploiting Narrow-Width Operands for Reducing Register File Pressure," in *Proceedings of the 37th International Symposium on Microarchitecture*, (Portland, OR, USA), pp. 304–315, December 2004.

[41] ERNST, D. and AUSTIN, T., "Efficient Dynamic Scheduling Through Tag Elimination," in *Proceedings of the 29th International Symposium on Computer Architecture*, (Anchorage, AK, USA), pp. 37–45, May 2002.

[42] FADAVI-ARDEKANI, J., "MN Booth encoded multiplier generator using optimized Wallace trees," *IEEE Transactions on VLSI*, vol. 1, pp. 120–125, 1993.

[43] FARKAS, K. I., CHOW, P., JOUPPI, N. P., and VRANESIC, Z., "The Multicluster Architecture: Reducing Cycle Time Through Partitioning," in *Proceedings of the 30th International Symposium on Microarchitecture*, (Research Triangle Park, NC, USA), December 1997.

[44] GHOSE, K. and KAMBLE, M. B., "Reducing Power in Superscalar Processor Caches Using Subbanking, Multiple Line Buffers and Bit-Line Segmentation," in *Proceedings of the International Symposium on Low Power Electronics and Design*, (San Diego, CA, USA), pp. 70–75, August 1999.

[45] GOPLEN, B. and SAPATNEKAR, S., "Efficient Thermal Placement of Standard Cells in 3D ICs Using a Force Directed Approach," in *Proceedings of the International Conference on Computer-Aided Design*, (San Jose, CA, USA), pp. 81–85, November 2003.

[46] GOPLEN, B. and SAPATNEKAR, S., "Thermal Via Placement in 3D ICs," in *Proceedings of the International Symposium on Physical Design*, (San Fransico, CA, USA), pp. 167–174, April 2005.

[47] GOSHIMA, M., NISHINO, K., NAKASHIMA, Y., ICHIRO MORI, S., KITAMURA, T., and TOMITA, S., "A High-Speed Dynamic Instruction Scheduling Scheme for Superscalar Processors," in *Proceedings of the 34th International Symposium on Microarchitecture*, (Austin, TX, USA), pp. 225–236, December 2001.

[48] GOWAN, M. K., BIRO, L. L., and JACKSON, D. B., "Power Considerations in the Design of the Alpha 21264 Microprocessor," in *Proceedings of the 35th Design Automation Conference*, (San Francisco, CA, USA), pp. 732–737, June 1998.

[49] GRODSTEIN, J., RAYESS, R., TRUEX, T., SHATTUCK, L., LOWELL, S., BAILEY, D., BERTUCCI, D., BISCHOFF, G., DEVER, D., GOWAN, M., LANE, R., LILLY, B., NAGALLA, K., SHAH, R., SHRIVER, E., YIN, S.-H., and MORTON, S., "Power and CAD Considerations for the 1.75MByte, 1.2GHz L2 Cache on the Alpha 21364 CPU," in *Proceedings of the ACM Great Lakes Symposium on VLSI*, (New York, NY, USA), pp. 1–6, April 2002.

[50] GUARINI, K. W., TOPOL, A. W., IEONG, M., YU, R., SHI, L., NEWPORT, M. R., FRANK, D. J., SINGH, D. V., COHEN, G. M., NITTA, S. V., BOYD, D. C., O'NEIL, P. A., TEMPEST, S. L., POGGE, H. B., PURUSHOTHAMAN, S., and HAENSCH, W. E., "Electrical Integrity of State-of-the-Art 0.13μm SOI CMOS Devices and Circuits Transferred for Three-Dimensional (3D) Integrated Circuit (IC) Fabrication," in *Proceedings of the International Electron Devices Meeting*, pp. 943–945, December 2002.

[51] GUPTA, S., HILBERT, M., HONG, S., and PATTI, R., "Techniques for Producing 3D ICs with High-Density Interconnect," in *Proceedings of the 21st International VLSI Multilevel Interconnection Conference*, (Waikoloa Beach, HI, USA), September 2004.

[52] GUTHAUS, M. R., RINGENBERG, J. S., ERNST, D., AUSTIN, T. M., MUDGE, T., and BROWN, R. B., "MiBench: A Free, Commerically Representative Embedded Benchmark Suite," in *Proceedings of the 4th Workshop on Workload Characterization*, (Austin, TX, USA), pp. 83–94, December 2001.

[53] GUTMANN, R. J., LU, J. Q., DEVARAJAN, S., ZENG, A. Y., and ROSE, K., "Wafer-level three-dimensional monolithic integration for heterogeneous silicon ICs," in *2004 Topical Meeting on Silicon Monolithic Integrated Circuits in RF Systems*, pp. 45–48, 2004.

[54] HARRIS, D. L., OBERMAN, S. F., and HOROWITZ, M. A., "SRT division architectures and implementations," in *Proceedings. 13th IEEE Symposium on Computer Arithmetic*, pp. 18–25, 1997.

[55] HE, L., LIAO, W., and STAN, M., "System Level Leakage Reduction Considering Leakage and Thermal Interdependency," in *Proceedings of the 42nd Design Automation Conference*, (Anaheim, CA, USA), pp. 12–17, June 2004.

[56] HEALY, M., VITTES, M., EKPANYAPONG, M., BALLAPURAM, C., LIM, S. K., LEE, H.-H. S., and LOH, G. H., "Multi-Objective Microarchitectural Floorplanning for 2D and 3D ICs," *Transactions on Architecture and Code Optimization*, 2006.

[57] HINTON, G., SAGER, D., UPTON, M., BOGGS, D., CARMEAN, D., KYLER, A., and ROUSSEL, P., "The Microarchitecture of the Pentium 4 Processor," *Intel Technology Journal*, Q1 2001.

[58] HUNG, W.-L., LINK, G., XIE, Y., VIJAYKRISHNAN, N., and IRWIN, M. J., "Interconnect and Thermal-aware Floorplanning for 3D Microprocessors," in *Proceedings of the 7th International Symposium on Quality Electronic Design*, (San Jose, CA, USA), March 2006.

[59] IBM CORP., "Power PC ISA Architecture guide (WWW site)." `http://www.ibm.com/developerworks/eserver/articles/archguide.html`.

[60] IM, S. and BANERJEE, K., "Full chip thermal analysis of planar (2-D) and vertically integrated (3-D) high performance ICs," in *International IEDM Technical Digest*, pp. 727–730, Dec 2000.

[61] INTEL, "Intel Pentium Processors with MMX Technology." from `http://www.intel.com/design/intarch/mmx/mmx.htm`.

[62] INTEL CORPORATION, "IA-32 Intel Architecture Optimization Reference Manual." Order Number: 248966-011, 2004.

[63] INTEL CORPORATION, "Intel 64 and IA-32 Architectures Optimization Reference Manual." Order Number: 248966-015, 2007.

[64] ITRS, "International Technology Roadmap for Semiconductors." from `http://www.itrs.net`.

[65] J.-Q. LU AND A. JINDAL AND Y. KWON AND J. MCMAHON AND M. RASCO AND R. AUGUR AND T. S. CALE AND R. J. GUTMANN, "Evaluation procedures for wafer bonding and thinning of interconnect test structures for 3d ics," in *Proceedings of the*, (Troy, NY, USA), pp. 74–76, 2003.

[66] JOUPPI, N. P., "Improving Direct-Mapped Cache Performance by the Addition of a Small Fully-Associative Cache and Prefetch Buffers," in *Proceedings of the 17th International Symposium on Computer Architecture*, (Seattle, WA, USA), pp. 364–373, May 1990.

[67] JOYNER, J. W., VENKATESAN, R., ZARKESH-HA, P., DAVIS, J. A., and MEINDL, J. D., "Impact of three-dimensional architectures on interconnects in gigascale integration," *IEEE Transactions on VLSI*, vol. 9, pp. 922–928, Dec 2001.

[68] JOYNER, J., *Opportunities and Limitations of Three-dimensional Integration for Interconnect Design*. PhD thesis, Georgia Institute of Technology, 2003.

[69] JUNG, S. M., JANG, J., CHO, W., MOON, J., KWAK, K., CHOI, B., HWANG, B., LIM, H., JEONG, J., KIM, J., and KIM, K., "The revolutionary and truly 3-dimentional 25f2 sram technology with the smallest s3 cell, 0.16um2 and sstff for ultra high density sram," *VLSI Techn. Dig. Techn. Papers*, pp. 228–229, 2004.

[70] KANTER, D., "Intel's Next Generation Microarchitecture Unveiled," tech. rep., Real World Technologies. `http://www.realworldtech.com/`.

[71] KESSLER, R. E., "The Alpha 21264 Microprocessor," *IEEE Micro Magazine*, vol. 19, pp. 24–36, March–April 1999.

[72] KGIL, T., D'SOUZA, S., SAIDI, A., BINKERT, N., DRESLINSKI, R., MUDGE, T., REINHARDT, S., and FLAUTNER, K., "Picoserver: using 3d stacking technology to enable a compact energy efficient chip multiprocessor," in *ASPLOS-XII: Proceedings of the 12th international conference on Architectural support for programming languages and operating systems*, pp. 117–128, 2006.

[73] KNICKERBOCKER, J. U., PATEL, C. S., ANDRY, P. S., TSANG, C. K., BUCHWALTER, L. P., SPROGIS, E. J., GAN, H., HORTON, R. R., POLASTRE, R. J., WRIGHT, S. L., and COTTE, J. M., "3-D Silicon Integration and Silicon Packaging Technology Using Silicon Through-Vias," *IEEE Journal of Solid-State Circuits*, vol. 41, pp. 1718–1725, August 2006.

[74] KOGGE, P. M. and STONE, H. S., "A Parallel Algorithm for the Efficient Solution of a General Class of Recurrence Equations," pp. 786–793, August 1973.

[75] KOYANAGI, M., FUKUSHIMA, T., and TANAKA, T., "New three-dimensional integration technology to achieve a super chip," *8th International Conference on Solid-State and Integrated Circuit Technology*, vol. 53, pp. 318 – 321, October 2006.

[76] KOYANAGI, M., NAKAMURA, T., YAMADA, Y., KIKUCHI, H., FUKUSHIMA, T., TANAKA, T., and KURINO, H., "Three-Dimensional Integration Technology Based on Wafer Bonding With Vertical Buried Interconnections," *IEEE Transactions on Electron Devices*, vol. 53, pp. 2799–2808, November 2006.

[77] KURINO, H. and KOYANAGI, M., "Technology for three dimensional integrated system-on-a-chip," in *Proceedings of the7th International Conference on Solid-State and Integrated Circuits Technology*, pp. 599–602, October 2004.

[78] LADNER, R. E. and FISCHER, M. J., "Parallel Prefix Computation," *Journal of the ACM*, vol. 27, pp. 831–838, Oct 1980.

[79] LARSON, E., CHATTERJEE, S., and AUSTIN, T., "MASE: A Novel Infrastructure for Detailed Microarchitectural Modeling," in *Proceedings of the 2001 International Symposium on Performance Analysis of Systems and Software*, (Tucson, AZ, USA), pp. 1–9, November 2001.

[80] LEE, C., POTKONJAK, M., and MANGIONE-SMITH, W. H., "MediaBench: A Tool for Evaluating and Synthesizing Multimedia and Communication Systems," in *Proceedings of the 30th International Symposium on Microarchitecture*, (Research Triangle Park, NC, USA), pp. 330–335, December 1997.

[81] LEWIS, D. L. and LEE, H.-H. S., "A Scan-Island Based Design Enabling Pre-bond Testability in Die-Stacked Microprocessors," in *Proceedings of the International Test Conference*, (Santa Clara, CA), October 2007.

[82] LI, F., NICOPOULOS, C., and OTHERS, "Design and Management of 3D Chip Multiprocessors Using Network-in-Memory," in *Proceedings of the 33rd International Symposium on Computer Architecture*, (Boston, MA, USA), pp. 130–141, June 2006.

[83] LI, Y., LEE, B., BROOKS, D., HU, Z., and SKADRON, K., "CMP Design Space Exploration Subject to Physical Constraints," in *Proceedings of the 12th International Symposium on High Performance Computer Architecture*, (Austin, TX, USA), 2006.

[84] LIM, S. K., "Physical design for 3D system on package," *IEEE Des. Test*, vol. 22, pp. 532–539, November 2005.

[85] LIN, M. and GAMAL, A. E., "A routing fabric for monolithically stacked 3d-fpga," in *FPGA '07: Proceedings of the 2007 ACM/SIGDA 15th international symposium on Field programmable gate arrays*, pp. 3–12, 2007.

[86] LIN, M., GAMAL, A. E., LU, Y.-C., and WONG, S., "Performance benefits of monolithically stacked 3d-fpga," in *FPGA '06: Proceedings of the 2006 ACM/SIGDA 14th international symposium on Field programmable gate arrays*, pp. 113–122, 2006.

[87] LIPASTI, M. H., MESTAN, B. R., and GUNADI, E., "Physical Register Inlining," in *Proceedings of the 31st International Symposium on Computer Architecture*, (München, Germany), pp. 325–335, June 2004.

[88] LIU, C. C., GANUSOV, I., BURTSCHER, M., and TIWARI, S., "Bridging the Processor-Memory Performance Gap with 3D IC Technology," *IEEE Design and Test of Comp.*, vol. 22, pp. 556–564, November–December 2005.

[89] LOH, G. H., "Exploiting Data-Width Locality to Increase Superscalar Execution Bandwidth," in *Proceedings of the 35th International Symposium on Microarchitecture*, (Istanbul, Turkey), pp. 395–405, November 2002.

[90] LOI, G. L., AGRAWAL, B., SRIVASTAVA, N., LIN, S.-C., SHERWOOD, T., and BANERJEE, K., "A thermally-aware performance analysis of vertically integrated (3-d) processor-memory hierarchy," in *DAC '06: Proceedings of the 43rd annual conference on Design automation*, (San Francisco, CA, USA), pp. 991–996, 2006.

[91] MAGEN, N., KOLODNY, A., WEISER, U., and SHAMIR, M., "Interconnect-Power Dissipation in a Microprocessor," in *Proceedings of the International Workshop on System-Level Interconnect Prediction*, (Paris, France), pp. 7–13, February 2004.

[92] MAHANT-SHETTI, S. S., BALSARA, P. T., and LEMONDS, C., "High performance low power array multiplier using temporal tiling," *IEEE Transactions on VLSI*, vol. 7, pp. 121–124, 1999.

[93] MASUDA, H., OHKAWA, S., KUROKAWA, A., and AOKI, M., "Challenge: variability characterization and modeling for 65- to 90-nm processes," in *Proceedings of the IEEE CICC*, pp. 593–9, 2005.

[94] MATSON, M., BAILEY, D., BELL, S., BIRO, L., BUTLER, S., CLOUSER, J., FARRELL, J., GOWAN, M., PRIORE, D., and WILCOX, K., "Circuit implementation of a 600 mhz superscalar risc microprocessor," *International Conference on Computer Design*, pp. 104–110, Oct 1998.

[95] MAYEGA, J., ERDOGAN, O., BELEMJIAN, P. M., ZHOU, K., MCDONALD, J. F., and KRAFT, R. P., "3D Direct Vertical Interconnect Microprocessors Test Vehicle," in *Proceedings of the ACM Great Lakes Symposium on VLSI*, (Washington, DC, USA), pp. 141–146, April 2003.

[96] MICHAUD, P. and SEZNEC, A., "Data-Flow Prescheduling for Large Instruction Window in Out-of-Order Processors," in *Proceedings of the 7th International Symposium on High Performance Computer Architecture*, (Monterrey, Mexico), pp. 27–36, January 2001.

[97] MINZ, J., LIM, S. K., CHOI, J., and SWAMINATHAN, M., "Module Placement for Power Supply Noise and Wire Congestion Avoidance in 3D Packaging," in *Proceedings of the 13th Topical Meeting on Electrical Performance of Electronic Packaging*, (Portland, OR, USA), October 2004.

[98] MINZ, J., WONG, E., and LIM, S. K., "Reliability-aware floorplanning for 3d circuits," *IEEE International SOC Conference*, 2005.

[99] MOHAMOOD, F., HEALY, M., LIM, S. K., and LEE, H.-H. S., "Noise-Direct: A Technique for Power Supply Noise Aware Floorplanning Using Microarchitecture Profiling," in *Proceedings of the 12th Asia South Pacific Design Automation Conference*, (Yokohama, Japan), pp. 786–791, January 2007.

[100] MOHAMOOD, F., HEALY, M. B., LIM, S. K., and LEE, H.-H. S., "A Floorplan-Aware Dynamic Inductive Noise Controller for Reliable Processor Design," in *Proceedings of the 39th International Symposium on Microarchitecture*, pp. 3–14, December 2006.

[101] MONDAL, M., RICKETTS, A. J., KIROLOS, S., RAGHEB, T., LINK, G., VIJAYKRISHNAN, N., and MASSOUD, Y., "Thermally robust clocking schemes for 3d integrated circuits," in *DATE '07: Proceedings of the conference on Design, automation and test in Europe*, (Nice, France), pp. 1206–1211, 2007.

[102] MOORE, G. E., "Cramming More Components Onto Integrated Circuits," *Electronics*, April 1965.

[103] MORROW, P. R., PARK, C.-M., RAMANATHAN, S., KOBRINSKY, M. J., and HARMES, M., "Three-dimensional wafer stacking via Cu-Cu bonding integrated with 65-nm strained-Si/low-k CMOS technology," in *IEEE Electron Device Letters*, vol. 27, pp. 335–337, May 2006.

[104] MORROW, P., KOBRINSKY, M. J., RAMANATHAN, S., PARK, C.-M., HARMES, M., RAMACHANDRARAO, V., MOG PARK, H., KLOSTER, G., LIST, S., and KIM, S., "Wafer-Level 3D Interconnects Via Cu Bonding," in *Proceedings of the 21st Advanced Metallization Conference*, (San Diego, CA, USA), October 2004.

[105] MYSORE, S., AGARWAL, B., SRIVASTAVA, N., LIN, S.-C., BANERJEE, K., and SHERWOOD, T., "Introspective 3D Chips," in *Proceedings of the 12th Symposium on Architectural Support for Programming Languages and Operating Systems*, 2006.

[106] PALACHARLA, S., *Complexity-Effective Superscalar Processors*. PhD thesis, University of Wisconsin, 1998.

[107] PERELMAN, E., HAMERLY, G., and CALDER, B., "Picking Statistically Valid and Early Simulation Points," in *Proceedings of the 2003 International Conference on Parallel Architectures and Compilation Techniques*, (New Orleans, LA, USA), pp. 244–255, September 2004.

[108] PRASHER, R. S., CHANG, J.-Y., SAUCIUC, I., NARASIMHAN, S., CHAU, D., CHRYSLER, G., MYERS, A., PRSTIC, S., and HU, C., "Nano and Micro Technology-Based Next-Generation Package-Level Cooling Solutions," *Intel Technology Journal*, vol. 9, November 2005.

[109] PUTTASWAMY, K. and LOH, G. H., "Implementing Caches in a 3D Technology for High Performance Processors," in *Proceedings of the International Conference on Computer Design*, (San Jose, CA, USA), October 2005.

[110] PUTTASWAMY, K. and LOH, G. H., "Dynamic Instruction Schedulers in a 3-Dimensional Integration Technology," in *Proceedings of the ACM Great Lakes Symposium on VLSI*, pp. 153–158, 2006.

[111] PUTTASWAMY, K. and LOH, G. H., "Implementing Register Files for High-Performance Microprocessors in a Die-Stacked (3D) Technology," in *Proceedings of the International Symposium on VLSI*, 2006.

[112] PUTTASWAMY, K. and LOH, G. H., "The Impact of 3-Dimensional Integration on the Design of Arithmetic Units," in *Proceedings of the International Symposium on Circuits and Systems*, (Kos, Greece), pp. 4951–4954, May 2006.

[113] PUTTASWAMY, K. and LOH, G. H., "Thermal Analysis of a 3D Die-Stacked High-Performance Microprocessor," in *Proceedings of the ACM Great Lakes Symposium on VLSI*, pp. 19–24, 2006.

[114] PUTTASWAMY, K. and LOH, G. H., "Scalability of 3D-Integrated Arithmetic Units in High-Performance Microprocessors," in *Proceedings of the Design Automation Conference*, (San Diego, CA, USA), pp. 622–625, 2007.

[115] PUTTASWAMY, K. and LOH, G. H., "Thermal Herding: Microarchitecture Techniques for Controlling Hotspots in High-Performance 3D-Integrated Processors," in *Proceedings of the International Symposium on High Performance Computer Architecture*, (Phoenix, AZ), pp. 193–204, 2007.

[116] RAASCH, S. E., BINKERT, N. L., and REINHARDY, S. K., "A Scalable Instruction Queue Design Using Dependence Chains," in *Proceedings of the 29th International Symposium on Computer Architecture*, (Anchorage, AK, USA), pp. 318–329, May 2002.

[117] RABAEY, J. M., "Digital integrated circuits: a design perspective," Prentice Hall electronics and VLSI series, 1996.

[118] RAHMAN, A. and REIF, R., "System Level Performance Evaluation of Three-Dimensional Integrated Circuits," *IEEE Transactions on VLSI*, vol. 8, pp. 671–678, June 2000.

[119] RANGANATHAN, N. and FRANKLIN, M., "An Empirical Study of Decentralized ILP Execution Models," in *Proceedings of the Symposium on Architectural Support for Programming Languages and Operating Systems*, (San Jose, CA, USA), pp. 272–281, October 1998.

[120] REED, P., YEUNG, G., and BLACK, B., "Design Aspects of a Microprocessor Data Cache using 3D Die Interconnect Technology," in *Proceedings of the International Conference on Integrated Circuit Design and Technology*, (Austin, TX, USA), pp. 15–18, May 2005.

[121] REIF, R., FAN, A., CHEN, K.-N., and DAS, S., "Fabrication Technologies for Three-Dimensional Integrated Circuits," in *Proceedings of the 3rd International Symposium on Quality Electronic Design*, (San Jose, CA, USA), pp. 33–37, March 2002.

[122] RONEN, R., MENDELSON, A., LAI, K., LU, S.-L., POLLACK, F., and SHEN, J. P., "Coming Challenges in Microarchitecture and Architecture," *Proceedings of the IEEE*, vol. 89, pp. 325–340, March 2001.

[123] ROY, K., MUKHOPADHYAY, S., and MAHMOODO-MEIMAN, H., "Leakeage Current Mechanisms and Leakage Reduction Techniques in Deep-Submicrometer CMOS Circuits," *Proceedings of the IEEE*, vol. 91, pp. 305–327, February 2003.

[124] SAMSON, E. C., MACHIROUTU, S. V., CHANG, J.-Y., SANTOS, I., HERMERDING, J., DANI, A., PRASHER, R., and SONG, D. W., "Interface Material Selection and a Thermal Management Technique in Second-Generation Platforms Built on Intel Centrino Mobile Technology," *Intel Technology Journal*, vol. 9, February 2005.

[125] SAMSUNG ELECTRONICS CORPORATION, "Samsung Electronics Develops World's First Eight-die Multi Chip Package for Multimedia Cell Phones." Press Release from `http://www.samsung.com`, January 10 2005.

[126] SANDPILE, "WWW Site." `http://www.sandpile.org`.

[127] SANKARALINGAM, K., NAGARAJAN, R., LIU, H., KIM, C., HUH, J., BURGER, D., KECKLER, S. W., and MOORE, C. R., "Exploiting ILP, TLP, and DLP with the Polymorphous TRIPS Architecture," in *Proceedings of the 30th International Symposium on Computer Architecture*, (San Diego, CA, USA), pp. 422–433, May 2003.

[128] SCHROM, G., HAZUCHA, P., HAHN, J.-H., KURSUN, V., GARDNER, D., NARENDRA, S., KARNIK, T., and DE, V., "Feasibility of Monolithic and 3D-Stacked DC-DC Converters for Microprocessors in 90nm Technology Generation," in *Proceedings of the International Symposium on Low Power Electronics and Design*, (Newport Beach, CA, USA), pp. 263–268, August 2004.

[129] SEMICONDUCTOR INDUSTRY ASSOCIATION, "The National Technology Roadmap for Semiconductors." 1999.

[130] SERY, G., BORKAR, S., and DE, V., "Life is cmos: why chase the life after?," in *Proceedings of the 39th conference on Design automation*, pp. 78–83, 2002.

[131] SEZNEC, A., FELIX, S., KRISHNAN, V., and SAZEIDES, Y., "Design Tradeoffs for the Alpha EV8 Conditional Branch Predictor," in *Proceedings of the 29th International Symposium on Computer Architecture*, (Anchorage, AK, USA), May 2002.

[132] SHARKEY, J., PONOMAREV, D., GHOSE, K., and ERGIN, O., "Power-Efficient Wakeup Tag Broadcast," in *Proceedings of the International Conference on Computer Design*, (San Jose, CA, USA), October 2005.

[133] SHEN, J. P. and LIPASTI, M. H., *Modern Processor Design: Fundamentals of Superscalar Processors*. McGraw Hill, 2005.

[134] SITES, R., *Alpha AXP Architecture Reference Manual*. Butterworth-Heinemann, 2 ed., October 1995.

[135] SKADRON, K., STAN, M. R., HUANG, W., VELUSAMY, S., SANKARANARAYANAN, K., and TARJAN, D., "Temperature-Aware Microarchitecture," in *Proceedings of the 30th International Symposium on Computer Architecture*, pp. 2–13, May 2003.

[136] SKADRON, K., STAN, M. R., SANKARANARAYANAN, K., HUANG, W., VELUSAMY, S., and TARJAN, D., "Temperature-Aware Microarchitecture: Modeling and Implementation," *Transactions on Architecture and Code Optimization*, vol. 1, pp. 94–125, March 2004.

[137] SKLANSKY, J., "Conditional Sum Addition Logic," *IRE Transactions on Electronic Computers*, vol. 9, pp. 226–231, June 1960.

[138] SRINIVASAN, V., BROOKS, D., GSCHWIND, M., BOSE, P., ZYUBAN, V., STRENSKI, P. N., and EMMA, P. G., "Optimizing Pipelines for Power and Performance," in *Proceedings of the 35th International Symposium on Microarchitecture*, (Istanbul, Turkey), pp. 333–344, November 2002.

[139] STARK, J., BROWN, M. D., and PATT, Y. N., "On Pipelining Dynamic Instruction Scheduling Logic," in *Proceedings of the 33rd International Symposium on Microarchitecture*, (Monterey, CA, USA), pp. 57–66, December 2000.

[140] STRICKLAND, S., ERGIN, E., KAELI, D. R., and ZAVRACKY, P., "VLSI Design in the 3rd Dimension," *Integration: the VLSI Journal*, vol. 25, pp. 1–16, September 1998.

[141] SUBRAMANIAM, S. and LOH, G. H., "Fire-and-Forget: Load/Store Scheduling with No Store Queue At All," in *Proceedings of the 39th International Symposium on Microarchitecture*, (Orlando, FL, USA), pp. 273–284, December 2006.

[142] SUBRAMANIAM, S. and LOH, G. H., "Store Vectors for Scalable Memory Dependence Prediction and Scheduling," in *Proceedings of the 12th International Symposium on High Performance Computer Architecture*, (Austin, TX, USA), pp. 64–75, 2006.

[143] SWANSON, S., MICHELSON, K., SCHWERIN, A., and OSKIN, M., "WaveScalar," in *Proceedings of the 36th International Symposium on Microarchitecture*, (San Diego, CA, USA), pp. 291–302, May 2003.

[144] TAKAHASHI, O., COTTIER, S., DHONG, S. H., FLACHS, B., and SILBERMAN, J., "Power-Conscious Design of the Cell Processor's Synergistic Processor Element," *IEEE Micro Magazine*, vol. 25, pp. 10–18, September–October 2005.

[145] TEZZARON SEMICONDUCTOR, "WWW Site." `http://www.tezzaron.com`.

[146] TEZZARON SEMICONDUCTORS, "Tezzaron Unveils 3D SRAM." Press Release from `http://www.tezzaron.com`, January 24 2005.

[147] THOMPSON, S., ALAVI, M., and OTHERS, "130nm Logic Technology Featuring 60nm Transistors, Low-K Dielectrics, and Cu Interconnects," *Intel Technology Journal*, vol. 6, May 2002.

[148] TSAI, Y.-F., XIE, Y., VIJAYKRISHNAN, N., and IRWIN, M. J., "Three-Dimensional Cache Design Using 3DCacti," in *Proceedings of the International Conference on Computer Design*, (San Jose, CA, USA), October 2005.

[149] T.SAKURAI, "Closed-form expressions for interconnection delay, coupling and crosstalk in VLSI's," *IEEE Transactions on Electron Devices*, vol. 13, pp. 118–124, January 1993.

[150] TSCHANZ, J. W., NARENDRA, S., NAIR, R., and DE, V., "Effectiveness of adaptive supply voltage and body bias for reducing impact of parameter variations in low power and high performance microprocessors," *IEEE Journal of Solid-State Circuits*, vol. 38, pp. 826 – 829, May 2003.

[151] TSENG, J. H. and ASANOVIĆ, K., "Banked Multiported Register Files for High-Frequency Superscalar Microprocessors," in *Proceedings of the 30th International Symposium on Computer Architecture*, (San Diego, CA, USA), pp. 62–71, May 2003.

[152] TUINHOUT, H. P., "Impact of parametric mismatch and fluctuations on performance and yield of deep-submicron cmos technologies," in *ESSDERC*, pp. 95–101, 2002.

[153] VACHHARAJANI, M., "Microarchitecture modeling for design-space exploration," 2004.

[154] VAIDYANATHAN, B., HUNG, W. L., XIE, Y., NARAYANAN, V., and IRWIN, M. J., "Architecting Microprocessor Components in 3D Design Space," in *Proceedings of 20th Intl. Conference on VLSI Design*, (Bangalore, India), 2007.

[155] VILLA, L., ZHANG, M., and ASANOVIĆ, K., "Dynamic Zero Compression for Cache Energy Reduction," in *Proceedings of the 33rd International Symposium on Microarchitecture*, (Monterey, CA, USA), December 2000.

[156] WONG, E., MINZ, J., and LIM, S., "Power supply noise-aware 3D floorplanning for system-on-package," in *Proceedings of IEEE 14th Topical Meeting on Electrical Performance of Electronic Packaging*, pp. 259–262, October 2005.

[157] XIE, Y., LOH, G. H., BLACK, B., and BERNSTEIN, K., "Design space exploration for 3d architectures," *J. Emerg. Technol. Comput. Syst.*, vol. 2, no. 2, pp. 65–103, 2006.

[158] XUE, L., LIU, C., and TIWARI, S., "Multi-layers with buried structures (mlbs): An approach to three-dimensional integration," *IEEE International Conference on Silicon On Insulator*, pp. 117–118, 2001.

[159] ZENG, A., LÜ, J., ROSE, K., and GUTMANN, R. J., "First-Order Performance Prediction of Cache Memory with Wafer-Level 3D Integration," *IEEE Design and Test of Comp.*, vol. 22, pp. 548–555, November–December 2005.

[160] ZHANG, Y., YANG, J., and GUPTA, R., "Frequent Value Locality and Value-Centric Data Cache Design," in *Proceedings of the 9th Symposium on Architectural Support for Programming Languages and Operating Systems*, (Cambridge, MA, USA), pp. 150–159, November 2000.

VITA

Kiran Puttaswamy is a PhD candidate in the School of Electrical and Computer Engineering at the Georgia Institute of Technology and a member of the Superscalar Technology INnovation Group (STING). He is involved in designing high-performance microprocessors using 3-Dimensional integration technology. Professor Gabriel H. Loh is his thesis advisor. He has on-going research collaboration with the Microprocessor Architecture Research Society (MARS) led by Professor Hsien-Hsin S. Lee (co-advisor) and the Georgia Tech Computer Aided Design Laboratory (GT-CAD) led by Professor Sung Kyu Lim. He has conducted graduate research work with the Center for Research on Embedded Systems and Technology (CREST) in the past.

Conference Publications:

- Kiran Puttaswamy and Gabriel H. Loh, Scalability of 3D-Integrated Arithmetic Units in High-Performance Microprocessors, in the Proceedings of the ACM Design Automation Conference (DAC), June 4-8, 2007, San Diego, CA, USA.
- Kiran Puttaswamy and Gabriel H. Loh, Thermal Herding: Microarchitecture Techniques for Controlling HotSpots in High-Performance 3D-Integrated Processors, in the Proceedings of the International Symposium on High-Performance Computer Architecture (HPCA), February 10-14, 2007, Phoenix, AZ, USA.
- Chinnakrishnan Ballapuram, Kiran Puttaswamy, Gabriel H. Loh, and Hsien-Hsin S. Lee, Entropy-based Low Power Data TLB Design, in the ACM/IEEE Conference on Compilers, Architecture and Synthesis for Embedded Systems (CASES), October 23-25, 2006, Seoul, South Korea.
- Kiran Puttaswamy and Gabriel H. Loh, Thermal Analysis of a 3D Die-Stacked High-Performance Microprocessor, in the ACM/IEEE Great Lakes Symposium on VLSI (GLSVLSI), April 30-May 2, 2006, Philadelphia, PA, USA.
- Kiran Puttaswamy and Gabriel H. Loh, Dynamic Instruction Schedulers in a 3-Dimensional Integration Technology, in the ACM/IEEE Great Lakes Symposium on VLSI (GLSVLSI), April 30-May 2, 2006, Philadelphia, PA, USA.
- Kiran Puttaswamy and Gabriel H. Loh, The Impact of 3-Dimensional Integration on the Design of Arithmetic Units, in the IEEE International Symposium on Circuits and Systems (ISCAS), May 21-24, 2006, Kos, Greece.
- Kiran Puttaswamy and Gabriel H. Loh, Implementing Register File for High-Performance Microprocessors in a Die-Stacked (3D) Technology, in the IEEE International Symposium on VLSI (ISVLSI), pp. 384-389, March 1-3, 2006, Karlsruhe, Germany.
- Kiran Puttaswamy and Gabriel H. Loh, Implementing Caches in a 3D Technology for High Performance Processors, in the IEEE International Conference on Computer Design (ICCD), pp. 525-532, October 2-5, 2005, San Jose, California.
- Jinwoo Kim and Kiran Puttaswamy, Possibility and Limitation of a Hardware-Assisted Data Prefetching Framework Using Off-Line Training of Markovian Predictors, in the International Conference on Computer Design (CDES), June 27-30, 2005, Las Vegas, Nevada.

- Kiran Puttaswamy, Kresten McGrath, Satya Vadlamani, and Ravi Kolagotla, High speed high accuracy power estimation methodology for next generation system-on-a-chip (SoC) micro-architectures, in the Proceedings of Intel Design and Test Technology Conference (DTTC), 2004, Portland, Oregon.
- Raghavan Sudhakar, Kiran Puttaswamy, and Ravi Kolagotla, Efficient GSM-AMR channel decoding for next generation DSP by improved Traceback, in the Proceedings of Intel Design and Test Technology Conference (DTTC), 2004, Portland, Oregon.
- Kiran Puttaswamy, Dhinakarraj H. Gantala, and Ravi Kolagotla, Power analysis of a VLIW DSP core on an embedded processor system-on-a-chip (SoC) using instruction set architecture features, in the Proceedings of Intel Design and Test Technology Conference (DTTC), 2003, Portland, Oregon.
- Kiran Puttaswamy, Jun-Cheol Park, Kyu-won Choi, Abhijit Chatterjee, Peeter Ellervee, and Vincent John Mooney III, System Level Power-Performance Trade-Offs in Embedded Systems Using Voltage and Frequency Scaling of Off-Chip Buses and Memory, in the International Symposium on System Synthesis (ISSS), pp. 225-230, October 2-4, 2002, Tokyo, Japan.
- Krishna V. Palem, Rodric M. Rabbah, Vincent John Mooney, Pinar Korkmaz, and Kiran Puttaswamy, Design space optimization of embedded memory systems via data remapping, in the ACM SIGPLAN Joint Conference on Languages, Compilers, and Tools for Embedded Systems and Software and Compilers for Embedded Systems (LCTES-SCOPES), pp. 28-37, June 19-21, 2002, Berlin, Germany.
- Lakshmi N. Chakrapani, Pinar Korkmaz, Vincent John Mooney III, Krishna V. Palem, Kiran Puttaswamy, and W. F. Wong, The Emerging Power Crisis in Embedded Processors: What can a Poor Compiler Do?, in the International Conference on Compilers, Architecture and Synthesis for Embedded Systems (CASES), pp. 176-180, November 16-17, 2001, Atlanta, Georgia.

Journals/Book Chapters:

- Kiran Puttaswamy and Gabriel H. Loh, High-Performance 3D-Integrated Microarchitectures for Controlling Hotspots, in submission to the ACM Transactions on Architecture and Compiler Optimizations.
- Kiran Puttaswamy and Gabriel H. Loh, High-Performance, Efficient, and Scalable 3D-Integrated Arithmetic Units, in submission to the IEEE Transactions on VLSI.
- Kiran Puttaswamy and Gabriel H. Loh, 3D-Integrated SRAM Components for High-Performance Microprocessors, in submission to the IEEE Transactions on Computers.
- Kiran Puttaswamy, Peeter Ellervee, Vincent John Mooney, III, Krishna V. Palem, Weng-Fai Wong, Lakshmi Narasimhan Chakrapani, Kyu-Won Choi, Yuvraj Singh Dhillon, Utku Diril, Pinar Korkmaz, Kyoung-Keun Lee, Jun Cheol Park, and Abhijit Chatterjee, Power-performance trade-offs in second level memory used by an ARM-like RISC architecture, in the Springer Power Aware Computing Series, Ed: Robert Graybill and Rami Melhem, pp. 211 - 224, 2002.

Technical Reports:

- Rodric M. Rabbah and Kiran Puttaswamy, Design and Synthesis of a High Performance and Low Power Routing Switch for a Hypercube Network, Georgia Institute of Technology Technical Report CREST-TR-02-008, October 2002.
- Pinar Korkmaz, Kiran Puttaswamy, and Vincent Mooney III, Energy modeling of processor core and memory hierarchy using Synopsis and Kamble and Ghosh model, CREST Technical Report CREST-TR-02-002, February 2002.
- Krishna V. Palem, Rodric M. Rabbah, Vincent Mooney III, Pinar Korkmaz and Kiran Puttaswamy, Power Optimization of Embedded Memory Systems via Data Remapping, GA Tech Technical Report: GIT-CC-02-010.