Die Stacking (3D) Microarchitecture

Bryan Black, Murali Annavaram, Ned Brekelbaum, John DeVale, Lei Jiang, Gabriel H. Loh[1], Don McCauley, Pat Morrow, Donald W. Nelson, Daniel Pantuso, Paul Reed, Jeff Rupley, Sadasivan Shankar, John Shen, and Clair Webb
Intel® Corporation
Email: bryan.black@intel.com

Abstract

3D die stacking is an exciting new technology that increases transistor density by vertically integrating two or more die with a dense, high-speed interface. The result of 3D die stacking is a significant reduction of interconnect both within a die and across dies in a system. For instance, blocks within a microprocessor can be placed vertically on multiple die to reduce block to block wire distance, latency, and power. Disparate Si technologies can also be combined in a 3D die stack, such as DRAM stacked on a CPU, resulting in lower power higher BW and lower latency interfaces, without concern for technology integration into a single process flow. 3D has the potential to change processor design constraints by providing substantial power and performance benefits. Despite the promising advantages of 3D, there is significant concern for thermal impact. In this research, we study the performance advantages and thermal challenges of two forms of die stacking: Stacking a large DRAM or SRAM cache on a microprocessor and dividing a traditional microarchitecture between two die in a stack.

Results: *It is shown that a 32MB 3D stacked DRAM cache can reduce the cycles per memory access of a two-threaded RMS benchmark on average by 13% and as much as 55% while increasing the peak temperature by a negligible 0.08ºC. Off-die BW and power are also reduced by 66% on average. It is also shown that a 3D floorplan of a high performance microprocessor can simultaneously reduce power 15% and increase performance 15% with a small 14ºC increase in peak temperature. Voltage scaling can reach neutral thermals with a simultaneous 34% power reduction and 8% performance improvement.*

1. Introduction to 3D

3D die stacking is an emerging technology that eliminates wire both within a microprocessor die and between disparate die. Wire is a primary latency, area and power overhead in computing systems. Wire can consume more than 30% of the power within a microprocessor. With 3D die stacking, dies of different types can be stacked with a high bandwidth, low latency, and low power interface. Additionally, wire elimination using 3D provides new microarchitecture opportunities to trade off performance, power, and area.

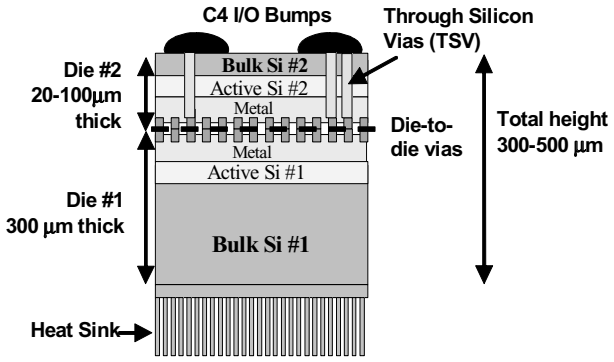

Figure 1. A 3D structure

A basic 3D structure is illustrated in Figure 1. Without any loss of generality, this work assumes a face-to-face bonding because it provides a very dense interface between adjacent die, enabling many options for 3D processor organizations. There are several other methods for die stacking and alignment including wafer-to-wafer bonding [11][15], die-to-die bonding, die-to-wafer bonding, die-partial wafer bonding, partial wafer-partial wafer bonding, and others. There are also many bonding technologies dependent on the bonding materials. It is also possible to stack many die; however, this work limits the discussion to two die stacks. In Figure 1 two die are joined face-to-face with a dense die-to-die via interconnect. The die-to-die (d2d) vias are placed on the top of the metal stack of each die and are bonded after alignment. It is important to note that the d2d vias are not

[1] Gabriel Loh contributed to this work while working at Intel® Corporation prior to becoming faculty at the Georgia Institute of Technology.

like traditional I/O pads; the d2d vias have size and electrical characteristics similar to conventional vias that connect on die metal routing layers. In face-to-face bonding, through-silicon-vias (TSVs) are required to connect the C4 I/O to the active regions of the two die. Power is also delivered through these backside vias. Die #2 is thinned for improved electrical characteristics and physical construction of the TSVs for power delivery and I/O. Good discussions of these processing details can be found in [8][11][14][15][26][27].

Recently 3D die stacking is drawing a great deal of attention, primarily in embedded processor systems. Prior work examines system-on-chip opportunities [4][5][10] [16][18][24], explores cache implementations [15][28] [30], designs 3D adder circuits [14][21], and projects wire benefits in full microprocessors [1][4][5][17][29]. In order to transform 3D design research ideas into products Technology Venture sponsors a dedicated forum for "3D Architectures for Semiconductor Integration and Packaging." At this forum [33] it is clear that the embedded industry considers emerging 3D technology a very attractive method for integrating small systems. Furthermore, existing 3D products from Samsung [32] and Tezzaron [34] corporations demonstrate that the silicon processing and assembly of structures similar to Figure 1 are feasible in large scale industrial productions. This work hence, focuses on power, performance and thermal issues of 3D stacking without delving into the feasibility details.

This paper explores the performance advantages of eliminating wire using 3D on two fronts:

(1) Shorten wires dedicated to off die interfaces connecting disparate die, such as off die wires connecting CPU and memory. Section 3 evaluates the performance potential of stacking memory on logic (Memory+Logic) [5][7] [12][13]. We quantify the performance and power benefits of stacking a large SRAM or DRAM caches on a microprocessor. Our results show that dramatically increasing on die storage increases performance and reduces required off die bandwidth while simultaneously reducing power. A key difference between our work and previous studies is that the prior work assumes that all of main memory can be integrated into the 3D stack. We consider RMS applications that target systems with main memory requirements that cannot be incorporated in a two-die stack, and instead we use the 3D-integrated DRAM as additional high-density cache.

(2) The second approach is to shorten wires connecting blocks within a traditional planar microprocessor. In this approach it is possible to implement a traditional microarchitecture across two or more die to construct a 3D floorplan. Such a Logic+Logic stacking, takes advantage of increased transistor density to eliminate wire between blocks of the microarchitecture [1][17][25]. The result is shorter latencies between blocks yielding higher performance and lower power. Section 4 takes a microprocessor from the Intel® Pentium® 4 family and converts it to a Logic+Logic 3D stacking to quantify the performance and power benefits of reduced wire delays in 3D.

While 3D provides power and performance advantages in both the above approaches, the most significant concern to 3D design is that 3D designs may increase the thermal hotspots. We evaluate the thermal impact of 3D design in these two scenarios and show that while 3D design does increase the temperature, the growth in temperature is negligible or can be overcome by an overall reduction in power consumption. Our results demonstrate that thermals are not an inexorable barrier to 3D design as generally believed.

2. Modeling Environment

This section describes our 3D performance and thermal evaluation infrastructure. The Memory+Logic stacking evaluation presented in Section 3 requires us to evaluate the performance of adding large caches to a microprocessor. In order to evaluate large cache benefits it is necessary to have long running benchmarks that have large data footprints to exercise the cache structures. On the other hand evaluating Logic+Logic stacking of a microprocessor requires a detailed microarchitecture simulator that can model the interconnection delays of logic blocks accurately. Hence, the goals of the two infrastructures are conflicting forcing us to use two different simulators which are described in Section 2.1 and Section 2.2, respectively. For both sceneries we use a general thermal simulation infrastructure, which is described in Section 2.3.

2.1. Modeling Memory+Logic Performance

For evaluating Memory+Logic stacking we use a trace driven multi-processor memory hierarchy simulator that can run billions of memory references to exercise large caches. This internally developed research tool is designed to model all aspects of the memory hierarchy including DRAM caches with banks, RAS, CAS, page sizes, etc. The input to this simulator is a novel memory address trace generated from a multi-threaded application running on a full system multi-processor simulator. The trace generator module runs alongside the full system simulator and keeps track of dependencies between instructions. The trace generator outputs one trace record for each memory instruction executed by the full system simulator. In addition to the usual trace fields such as cpu id, memory access address, and instruction pointer address, every trace record contains the unique identification number of an earlier trace record this record is dependent upon. The memory hierarchy simulator in turn honors all the dependencies specified in the

Name	Description
Conj	Solids Conjugate Gradient Solver
dSYM	Dense Matrix Multiplication
gauss	Linear Equation Solver using Gauss-Jordan Elimination
pcg	Preconditioned Conjugate Gradient Solver using Cholesky Preconditioner, Red-Black Reordering
sMvm	Sparse Matrix Multiplication
sSym	Symmetrical Sparse Matrix Multiplication
sTrans	Transposed Sparse Matrix Multiplication
sAVDF	Structural Rigidity Computation with AVDF Kernel
sAVIF	Structural Rigidity Computation with AVIF Kernel
sUS	Structural Rigidity Computation with US Kernel
Svd	Singular Value Decomposition with Jacobi Method
Svm	Pattern Recognition Algorithm for Face Recognition in Images

Table 1. The RMS workloads used for analysis in Section 3

trace and issues memory accesses accordingly. For instance, if a load address Ld2 is dependent on an earlier load Ld1, then Ld1 is first issued to the memory hierarchy to obtain the memory access completion time of Ld1. Then Ld2 is issued to the memory hierarchy only after Ld1 is completed.

To demonstrate highly parallel and memory intensive activity, we selected the RMS (Recognition, Mining, and Synthesis) benchmarks [31], shown in Table 1. The RMS workloads can be roughly characterized into two groups: applications, and kernels. Application benchmarks represent a complete solution to performing some task, while the kernels attempt to represent important iterative mathematical algorithms that are becoming more common in emerging applications. Application benchmarks include complex financial models, data mining, physics models, ray tracing for graphics rendering and production, and security focused image recognition algorithms. The math kernels attempt to focus on the basic building blocks of matrix oriented data manipulation and calculations that are being increasingly utilized to model and process complex systems.

For the results presented in this paper we ran two threaded RMS benchmarks on a full system simulator that simulates a two processor SMP system. We marked each benchmark to skip the data initialization phase and start collecting address traces during the computation phase. We ran each benchmark and collected 1 billion total memory references in a trace file, which correspond to roughly 2.5 billion executed instructions. These traces are then fed to the memory hierarchy simulator to obtain the cycles per memory access (CPMA). CPMA metric measures the total cycles spent from when a memory reference starts L1D access to the time when the request is satisfied by the memory hierarchy.

2.2. Modeling Logic+Logic Performance

For evaluating the Logic+Logic stacking, we used a traditional single threaded microarchitecture performance simulator. This performance simulator was developed by the Pentium® 4 design team and was used by the team during the pathfinding and design of the baseline microprocessor used in our study. Apart from modeling all traditional microarchitecture logic blocks, this simulator also accurately models the wire delays due to block interconnections. Due to the fact that this simulator was used by the product design team, we had the ability to run a much broader range of single-threaded applications that are also used in product design evaluations. In all we ran over 650 single thread benchmark traces including SPECINT, SPECFP, hand written kernels, multimedia, internet, productivity, server, and workstation applications.

2.3. Thermals

Thermals are an important part of any 3D microarchitecture because die stacking can dramatically increase power density if two highly active regions are stacked on top of each other. Heat dissipation is also challenged by the fact that each additional die is stacked farther and farther from the interface to the heat sink. This physical distance results in higher thermal resistances and potentially creates thermal isolation leading to self-heating of additional die.

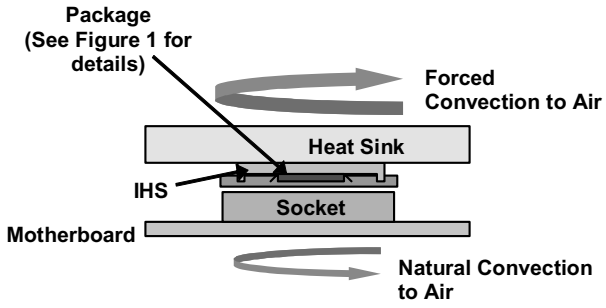

Figure 2. Cross section of system components

A complete 3D die stacking thermal modeling tool derived from silicon-validated production thermal tools was developed internally. A detailed thermal analysis of stacked die architectures requires the implementation of 3D models in order to account for the interactions of mul-

Name	Function	Value
Si #1 thickness	The thickness of the bulk Si of the die next to the heat sink	750 um
Si #2 thickness	The thickness of the bulk Si of the die next to the bumps	20 um
Si ther cond	The conductivity of bulk Si	120 W/mK
Cu metal thickness	The thickness of the Logic metal layers	12 um
Cu metal ther cond	The thermal conductivity of the Cu metal layers; This value accounts for the low-k insulating layers and via occupancy	12 W/mK
Al metal thickness	The thickness of the DRAM metal layers	2 um
Al metal ther cond	The thermal conductivity of the Al metal layers; This value accounts for the low-k insulating layers and via occupancy	9 W/mK
Bond thickness	The thickness of the bonding layer between the two die in the stack	15 um
Bond ther cond	The thermal conductivity of the bonding layer between the two die in the stack; This value accounts for air cavities and die to die interconnect density	60 W/mK
Heat sink ther cond	The thermal conductivity of the heat sink	400 W/mK
Ambient temperature		40 C

Table 2. Thermal constants and definitions for the 3D structure in Figure 1

tiple components in the stacked-die/package/mother-board system and non-symmetric nature of the resulting temperature distribution due to non-uniform die power dissipation and thermal gradients in other directions besides the stacking direction.

The model consists of the heat sink, integrated heat spreader (IHS), die, package, socket, and motherboard with boundary conditions for airflow on both sides as illustrated in Figure 2. The analysis of this heat conduction problem is based on the solution of the conservation of energy equation.

$$\rho_i c_i \partial_t T = K_i \nabla^2 T + Q_D \qquad (1)$$

In Equation (1), ρ, c, and K denote the density, heat capacity and thermal conductivity respectively, whereas the sub-index i indicates different materials. The capital letter T is the temperature and t denotes time. The symbols ∂_t and ∇^2 represent partial differentiation with respect to time and the Laplacian operator. Q_D is the power dissipated during operation which is represented by power maps at the die and package levels in our analysis. In our numerical implementation, Equation (1) is solved using 3D finite element method (FEM). Equation (1) has to be supplemented with appropriate boundary and initial conditions at the heat sink and motherboard as shown in Equation (2).

$$\partial_n T = h(T - T_{amb}) \qquad (2)$$

T_{amb} denotes the ambient temperature and h is a heat transfer coefficient. The symbols ∂_n denotes differentiation with respect to the normal to the interface. The power generation is defined through die and substrate package power maps.

Table 2 enumerates some of the thermal constants for the structure illustrated in Figure 1. It is obvious from these constants that heat dissipation is most sensitive to the metal layers and the bonding layer. Figure 3 illustrates the sensitivity to these constants for a stacked microprocessor. The two lines are the peak temperature on the die as the "Cu metal layer" and "Bonding layer" thermal conductivity vary from 60W/mK to 3W/mK. The "Cu metal layer" is the traditional metal stack on the two die. The "Bonding layer" is the new via interface between the two die. The 3D structure is sensitive to both layers, however the metal layer has a more significant temperature impact and unfortunately has the lower thermal conductivity of 12W/mK. These results demonstrate that the additional 3D process features are not the fundamental thermal limitation and in fact the existing metal layers present the most serious problem for the given system.

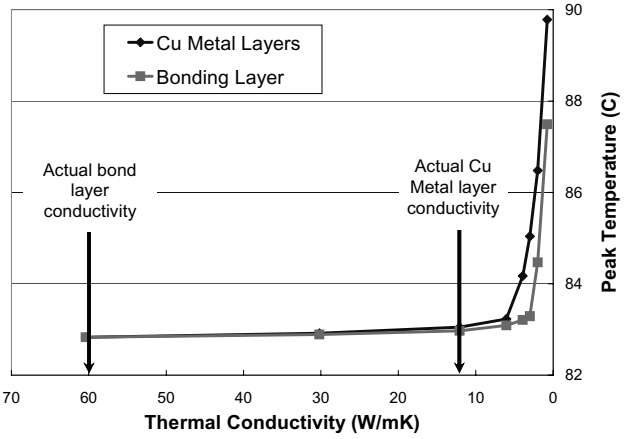

Figure 3. Heat dissipation sensitivity to the Cu metal layers and the bonding layer

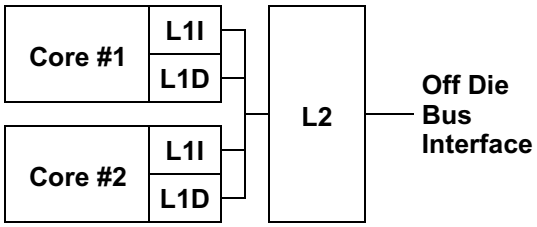

Figure 4. Intel® Core™ 2 Duo (Baseline)

3. Memory+Logic Stacking

Stacking cache memory on a microprocessor is one way to exploit 3D die stacking. Increased on die cache capacity improves performance by capturing larger working sets, reduces off die bandwidth requirements because more instructions and data are found on die, and reduces system power by reducing main memory accesses and bus activity. Figure 4 is a high level diagram of the baseline Intel® Core™ 2 Duo microprocessor used in this study. The microarchitecture configuration parameters of interest are shown in Table 3. The cores have private first level instruction and data caches of 32KB and share a 4MB second level cache. The L2 is connected to main memory through an off die bus interface. Both the banked main memory and stacked DRAM caches are modeled as 16 banked DDR3. Main memory has 4KB pages and the stacked DRAM has 512B pages with 64 byte sectors. Note that the 4MB L2 cache in the baseline occupies approximately 50% of the total die size of the baseline processor.

We explore three options for 3D stacking memory on this baseline processor die (also illustrated in Figure 7.). The first option is to increase the L2 size to 12MB SRAM and place the additional 8MB L2 cache on top of the baseline processor die. Since 4MB L2 is 50% of the total die area, 8MB stacked L2 is roughly the same size as the baseline die. The second option is to replace the SRAM L2 with a denser DRAM L2. Typically well designed DRAM is about 8X denser than an SRAM. Hence, we replace the 4MB L2 with a 32MB stacked DRAM L2. The tags for the stacked DRAM are placed on the processor die and the tag size increases the size of the baseline die by about 2MB depending on the implementation, resulting in a maximum 25% area overhead. Note, however, that in this option we have removed the 4MB L2 cache on the baseline processor die reducing the planar die dimensions by 50%. Hence even after accounting for the growth in the die area due to DRAM tags, the total CPU die dimensions are reduced. Finally, we explore the third option of stacking a 64MB DRAM on top of the baseline processor. This option allows us to stack DRAM without changing the baseline die dimensions. For the third option of stacking 64MB DRAM the tag size is about 4MB, and the existing 4MB cache on the baseline die is used to store the tags. In all these simulations the cache access latencies increase with cache size.

Figure 5 shows results from the three options of stacking memory on logic. The first bar in each group shows the baseline CPMA. The remaining bars show the CPMA with

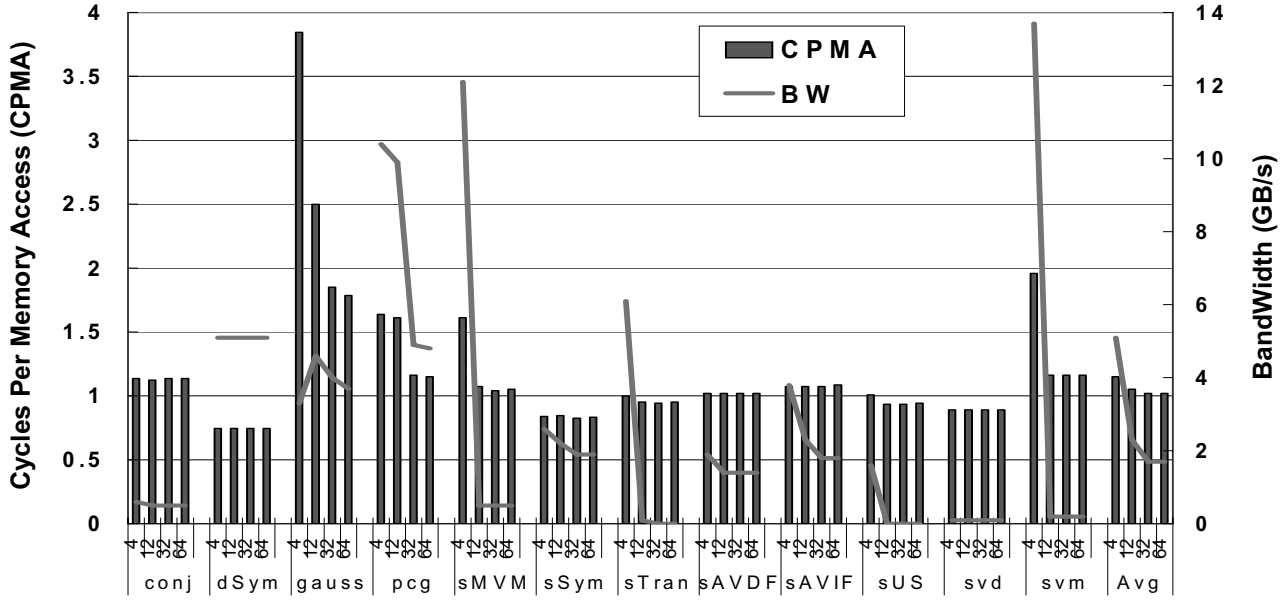

Figure 5. Performance results for 2 threaded RMS benchmarks as cache capacity increases from 4MB to 64MB.

Parameter	Value
Core Parameters	Same as Intel® Core™ 2 Duo
L1D Cache	32KB, 64B line, 8-way, 4 cyc
Shared L2	4 MB, 64B line, 16-way, 16 cyc
Stacked L2	**SRAM:** 12 MB, 24 cyc **DRAM:** 4-64MB, 512B page, 16 address interleaved banks, 64B sectors
DDR Main Memory	16 banks, 4KB page, 192 cyc
Bank delays (stacked L2 & DDR memory)	Page open 50 cyc Precharge 54 cyc Read 50 cyc
Off die Bus BW	16 GB/s

Table 3. Microarchitecture parameters

the three stacking options described above. The bars in Figure 5 show that for several of the RMS benchmarks (gauss, pcg, sMVM, sTrans, sUS, and svm) CPMA decrease dramatically as the last level cache increases from 4 to 64MB. The benchmarks that do not see improvement fit in the 4MB baseline and do not require more capacity. The secondary Y-axis plots the off die bandwidth for all four configurations. The bandwidth lines in Figure 5 show significant reduction in off die bandwidth as the cache capacity increases. The larger caches are effective at converting off-die bus accesses to on-die cache hits. Increasing the last level cache capacity from 4MB to 32MB, on average, reduces bus bandwidth requirements by 3x and CMPA by 13% with peak CMPA reduction of 50%. There is also a 66% average power reduction in average bus power, due to reduced bus activity. Assuming a bus power consumption rate of 20mW/Gb/s, 3D stacking of DRAM reduces bus power by 0.5W.

The performance improvements and bandwidth reductions in Figure 5 are very good; however in a 3D die stack the resulting thermals may not be acceptable. Figure 6(a) illustrates the power density map and Figure 6(b) illustrates the thermal map of the baseline microprocessor with 4MB of shared L2 cache which occupies approximately 50% of the chip area. The power map clearly illustrates the difference in the heat generated within the cores relative to the cache. The total power corresponding to these power maps are from a 92W skew of the baseline processor. The greatest concentration of power is in the FP units, reservation stations, and the load/store unit, pointed to in Figure 6(b). Using our 3D thermal modeling tool assuming standard desktop package cooling and an ambient temperature of 40ºC, the two hottest spots are at 88.4ºC and the coldest spot is 59ºC for the reference planar design.

Figure 7 shows the block diagrams including power consumption of (a) the baseline 4MB processor; (b) an additional 8MB of stacked SRAM with a total of 12MB of L2; (c) 32MB of stacked DRAM with the 4MB SRAM removed; and (d) 64MB of stacked DRAM. In our design 4MB of SRAM consume 7W, 32MB of DRAM consume 3.1W, and 64MB of DRAM consume 6.2W. This 3D DRAM is low power compared to DDR3 because the 3D die to die interconnect is much lower power than traditional off-die I/O. The RC of the all copper die to die interconnect used to interface the DRAM to the processor is comparable to 1/3 the RC of a typical via stack from first metal to last metal. The power of each configuration in Figure 7 is a little different making thermal comparisons challenging. The

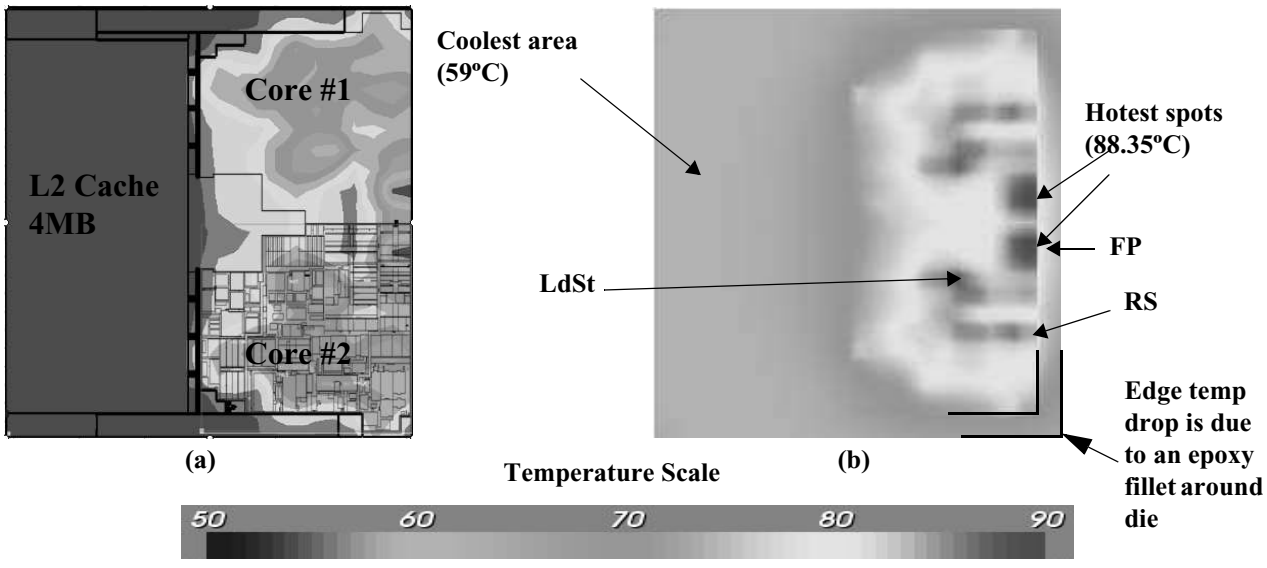

Figure 6. Intel® Core™ 2 Duo planar floorplan: (a) power map; (b) thermal map

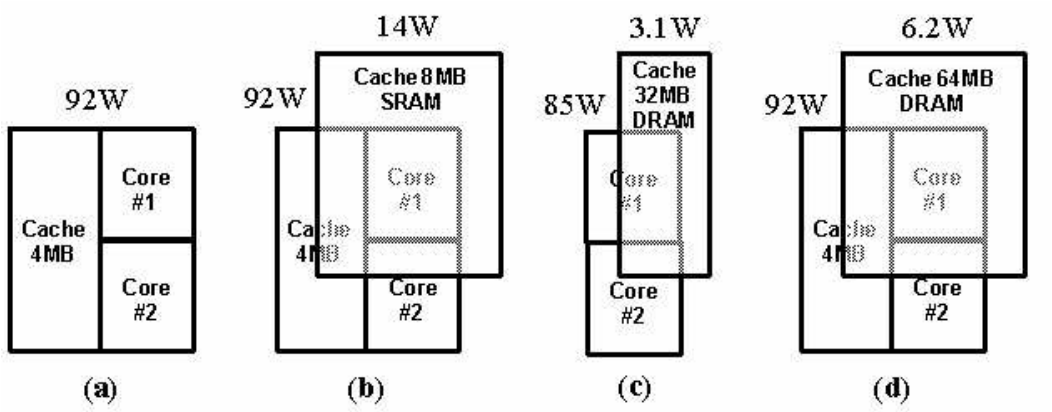

Figure 7. Memory stacked options: (a) 4MB baseline; (b) 8MB stacked for a total of 12MB; (c) 32MB of stacked DRAM with no SRAM; (d) 64MB of stacked DRAM

12MB case adds 200% more SRAM cache and increases the total power by 14W to 106W. The 32MB case is slightly lower power because the DRAM is lower power than the SRAM however the power density is increased due to the stacking. In all cases the highest power die is placed closest to the heat sink.

Figure 8(a) shows the peak temperature for all 3 stacking configurations compared to the baseline. Stacking SRAM results in the greatest thermal increase because of the higher power density of SRAM compared to DRAM. None of the stacking options significantly impact the thermals. In order to contrast to the 2D reference thermals, Figure 8(b) shows the 3D 12MB thermal map. The shape of the thermal behavior is the same between the reference machine and this 3D example because the cache-only die in the stack has uniform power. Notice there is a slight increase in heat density across the die in the 3D case. The results in this section show that the thermal impact of stacking memory is not significant, while there are significant performance and power advantages that can be exploited by stacking memory.

4. Logic+Logic Stacking

This section exploits the transistor density benefits of 3D die stacking by dividing a traditional microprocessor design between two die, called Logic+Logic stacking. The demonstration vehicle is borrowed from the our previous work [1]. This work presents a more complete analysis of the thermal consequences of Logic+Logic stacking. Figure 9 illustrates the planar floorplan of a microprocessor from the family of Intel® Pentium® 4 microprocessors. This is a deeply pipelined microarchitecture with a branch misprediction penalty of more than 30 clock cycles.

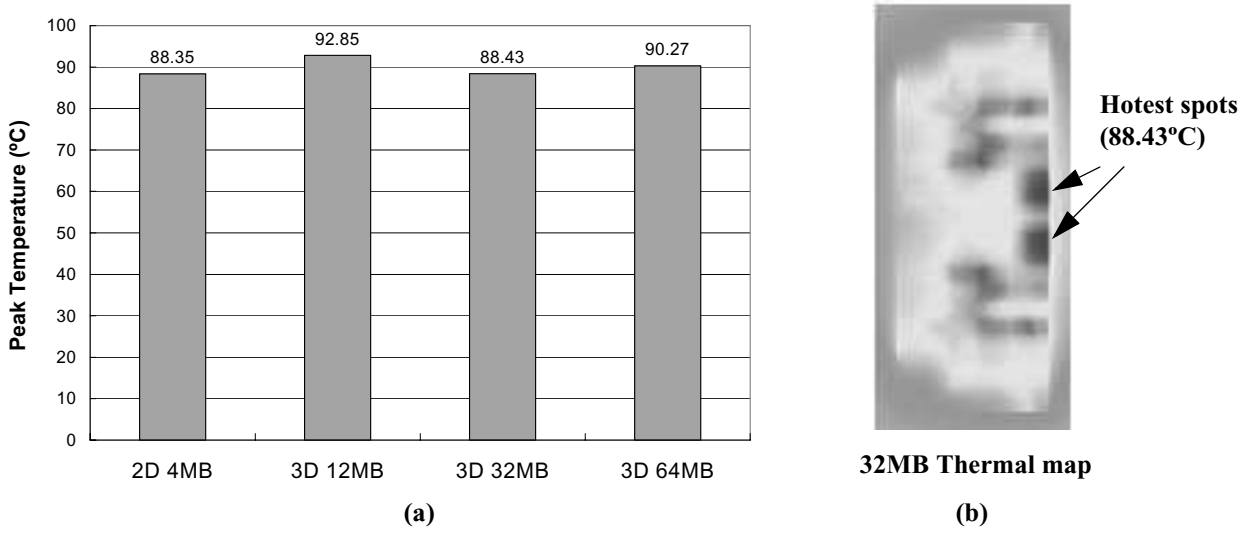

Figure 8. (a) Temperature results for the stacked 12MB, 32MB, and 64MB compared to the baseline 4MB; (b) Thermal map of the 3D stacked 32MBs. (Note: Same temperature scale as Figure 6)

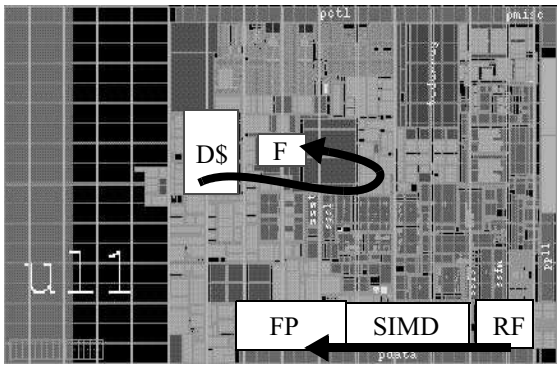

Figure 9. Planar floorplan of a deeply pipelined microprocessor with the load to use and floating point register read to execute paths

Since this microarchitecture has 100s of thousands of nets that would all require optimization to demonstrate a frequency improvement, this work assumes a constant frequency and focuses on eliminating pipe stages in the microarchitecture. Note that the term pipe stage is used to refer to all pipe stages in the microarchitecture including the cache hierarchy, store retirement, post completion resource recovery, etc. The number of pipe stages in this microarchitecture is much greater than the miss-prediction clocks. Power is improved by reducing total metal capacitance, driver strengths, number of pipe stage latches, repeating latches, and repeaters

Using Logic+Logic stacking, a new 3D floorplan can be developed that requires only 50% of the original footprint. The goal is to reduce inter-block interconnect by stacking and reducing intra-block, or within block interconnect through block splitting. The new 3D floorplan is in Figure 10.

Logic+Logic stacking simply moves blocks closer together reducing inter-block latency and power. Much of this effort concentrates on known performance sensitive pipelines. For example load-to-use delay is critical to the overall performance of most benchmarks. The path between the first level data cache (D$) and the data input to the functional units (F) is drawn illustratively in Figure 9. The worst case path occurs when load data must travel from the far edge of the data cache, across the data cache to the farthest functional unit yielding at least one clock cycle of wire delay entirely due to planar floorplan limitations. Figure 10 shows that a 3D floorplan can overlap the D$ and functional units. In the 3D floorplan, the load data only travels to the center of the D$, at which point it is routed to the other die to the center of the functional units. As a result of stacking that same worst case path contains half as much routing distance, since the data is only traversing half of the data cache and half of the functional units, thus eliminating the one clock cycle of delay in the load-to-use delay. This stacking is also favorable for thermals because the D$ is relatively low power and the 3D power density of the D$ on top of the functional units is lower than the planar floorplan's hottest area over the instruction scheduler.

Another example illustrated in Figure 9 is the floating point register file (RF) data out to the floating point (FP) unit input. The single instruction multiple data (SIMD) unit is intentionally between the FP and RF because the planar floorplan is optimized for the more critical SIMD applications. This placement adds two cycles to the latency of all FP instructions. In the 3D floorplan of Figure 10, it is possible to optimize for both FP and SIMD eliminating the two cycles previously allocated for wire delay, thus improving the performance of all FP applications, while not hurting SIMD applications.

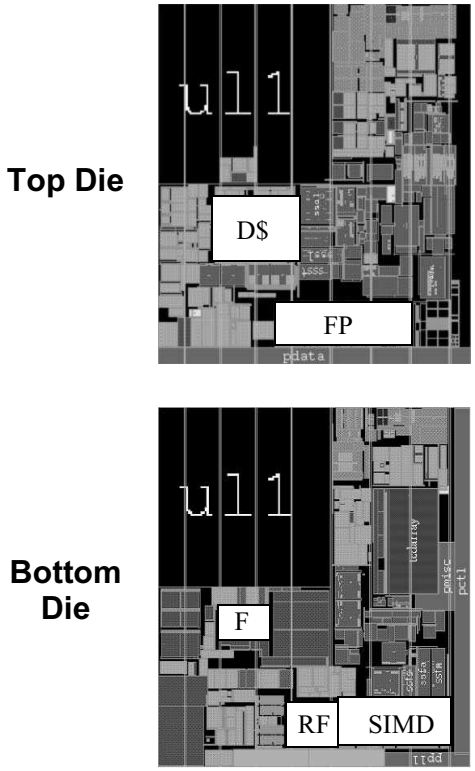

Figure 10. 3D floorplan of the planar microprocessor in Figure 9; The wire delay and resulting latencies within the load to use and FP register read to FP execute are reduced significantly.

Using Logic+Logic stacking, 25% of all pipe stages in the microarchitecture are eliminated with the new 3D floorplan simply by reducing metal runs, resulting in a 15% performance improvement. Table 4 enumerates the macro functional parts of the machine, the percent of the planar pipe stages eliminated with the new 3D floorplan, and the performance improvement of each modification. Only full

pipe stages are eliminated in this study, partial pipe stage improvements will affect frequency but are not reflected here. A notable improvement is a 30% reduction in the lifetime of store instructions after retirement. This greatly reduces the energy per store instruction and improves performance significantly by more efficiently utilizing the limited capacity of the store queues.

Normally an IPC increase causes more activity which increases total power. However, the IPC improvements measured in Table 4 are due to latency reductions from the elimination of wire delay and the subsequent pipestages. Therefore energy per instruction is reduced by removing power consuming metal and latches.

Functionality	% of Stages Eliminated	Perf. Gain (%)
Front-end pipeline	12.5%	~0.2%
Trace cache read	20%	~0.33%
Rename allocation	25%	~0.66%
FP inst. latency	Variable	~4.0%
Int register file read	25%	~0.5%
Data cache read	25%	~1.5%
Instruction loop	17%	~1.0%
Retire to de-allocation	20%	~1.0%
FP load latency	35%	~2.0%
Store lifetime	30%	~3.0%
Total	**~25%**	**~15%**

Table 4. Logic+Logic 3D stacking performance improvement and pipeline changes.

Baseline power data for the planar design is gathered using performance model activities and detailed circuit and layout based power roll ups from each block in the physical design. 3D power is estimated from the baseline by scaling according to the proposed design modifications. The removed pipestages are dominated by long global metal. As a result, the number of repeaters and repeating latches in the implementation is reduced by 50%. The two die in the 3D floorplan also share a common clock grid. The 3D clock grid will have 50% less metal RC than the planar design because the 3D floorplan footprint is 50% smaller, yielding a better skew, jitter, and lower power [1][5][7][11][17]. Clock grid details are considered beyond the scope of this work, but the power advantages are relevant. Fewer repeaters, a smaller clock grid, and significantly less global wire yields a 15% power reduction overall. Typically a power reduction results in a performance reduction, but in the 3D design the energy consumed per instruction is reduced without reducing performance therefore the new 3D floorplan has a 15% power reduction while simultaneously increasing performance by 15%. Further power improvement can be found by dividing blocks between die [1][7][25] but that is beyond the scope of this work. It is important to note that our power and thermal results are conservative because our reported 15% power reduction does not account for these additional means of reducing both total power and power density.

A risk of 3D stacking is the accidental doubling of power density and the thermal consequences. In a complex industrial design such as the design used in this study there are already a handful of peak hot areas and many very hot areas. Hence, it is critical to not increase the peak power density when stacking for 3D. This task is challenging but not impossible. A simple iterative process of placing blocks, observing the new power densities and repairing outliers was used in this experiment. The result is a 1.3x power density increase and 14°C temperature increase as illustrated by the second bar in Figure 11. The 3rd bar in Figure 11 labeled "3D Worstcase" shows a 26°C increase if there where no power savings from the 3D floorplan and the stacking were to result in a 2x power density. This bar shows the potential limitations of Logic+Logic stacking.

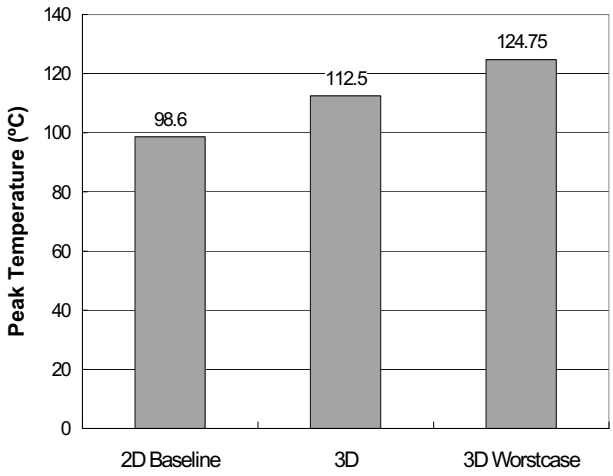

Figure 11. Temperature of the Logic+Logic 3D floorplan compared to the baseline and when there is no power reduction for a worst case 3D

Since there is a performance gain of 15% along with the 15% power reduction, it is possible to voltage and frequency scale the final results to reach a neutral peak temperature for the 3D floorplan. Table 5 enumerates several possible ways to take advantage of the 3D stacking by scaling voltage and frequency. The scaling of performance as a function of frequency is measured by the performance model to be a 0.82% performance improvement for each 1% increase in frequency. Performance and frequency do not scale 1:1 for many reasons of which main memory latency

3D provides 15% added perf and 15% pwr savings at same frequency

	Pwr	Pwr %	Temp	Perf	Vcc	Freq
Baseline	147	100%	**99**	100%	1	1
Same Pwr	147	100%	**127**	129%	1	1.18
Same Freq.	125	85%	**113**	115%	1	1
Same Temp	97.28	66%	**99**	108%	0.92	0.92
Same Perf.	68.2	46%	**77**	100%	0.82	0.82

Table 5. Frequency and voltage scaling the Logic+Logic stacked 3D floorplan; Conversion equations for Power, Vcc, and Frequency are included.

Perf vs. Freq	**0.82% performance for 1% frequency**
Freq vs. Vcc	**1% for 1% in Vcc**

is a significant factor. The frequency for this microprocessor varies 1% for each 1% change in Vcc. This 1:1 relationship and subsequent linear behavior was measured for this processor and is a sufficient approximation for the voltage ranges in Table 5. The thermal results in Table 5 are simulated using the tool described in Section 2.3.

Table 5 demonstrates the significance of a simultaneous 15% performance gain and 15% power reduction. The 3D floorplan can attain neutral thermals by frequency and voltage scaling. The result is a 34% power reduction and 8% performance improvement. Scaling to neutral performance yields a 54% power reduction. From the results presented in this section we demonstrate significant advantage for Logic+Logic 3D stacking of a microprocessor using simple layout modifications. We expect further improvements to be possible by further design optimizations for 3D.

5. Conclusions

This work explores the microarchitecture advantages and challenges of 3D die stacking. Two specific stacking examples are explored to demonstrate different methods of using 3D stacking to reduce wire within a microprocessor implementation and within a system. It is shown that a 32MB 3D stacked DRAM cache can reduce the cycles per memory access of a two-threaded RMS benchmark on average by 13% and as much as 55% while increasing the peak temperature by a negligible 0.08ºC. It is also shown that a 3D floorplan of a high performance microprocessor can simultaneously reduce power 15% and increase performance 15% with a small 14ºC increase in peak temperature. Voltage scaling can reach neutral thermals with a simultaneous 34% power reduction and 8% performance improvement.

6. References

[1] B. Black, D. Nelson, C. Webb, and N. Samra, "3D Processing Technology and Its Impact on iA32 Microprocessors," In Proceedings of the 2004 IEEE International Conference on Computer Design, pages 316-318, October 2004.

[2] J. Cong, J. Wei, and Y. Zhang, "A Thermal-Driven Floorplanning Algorithm for 3D Ics," In Proceedings of the International Conference on Computer-Aided Design, pages 306-313, November 2004.

[3] J. Cong, and Y. Zhang, "Thermal-Driven Multilevel Routing for 3-D Ics," In Proceedings of the Asia South Pacific Design Automation Conference, pages 121-126, January 2005.

[4] Y. Deng and W. Maly, "Interconnect Characteristics of 2.5-D System Integration Scheme," In Proceedings of the International Symposium on Physical Design, pages 171-175, April 2001.

[5] Y. Deng and W. Maly, "2.5D system integration: a design driven system implementation schema," In Proceedings of the Asia and South Pacific Design Automation Conference, pages 450-455, January 2004.

[6] B. Goplen and S. S. Sapatnekar, "Efficient Thermal Placement of Standard Cells in 3D ICs using a Force Directed Approach," In Proceedings of the International Conference on Computer-Aided Design, pages 86-89, November 2003.

[7] R. J. Gutmann, J.-Q. Lu, Y. Kwon, J. F. McDonald, and T. S. Cale, "Three-dimensional (3D) ICs: A Technology Platform for Integrated Systems and Opportunities for New Polymeric Adhesives," In Proceedings of the First International Conference on Polymers and Adhesives in Microelectronics and Photonics, pages 173-180, October 2001.

[8] R. Islam, C. Brubaker, P. Lindner and C. Schaefer, "Wafer level packaging and 3D interconnect for IC technology," In Proceedings of the Advanced Semiconductor Manufacturing 2002 IEEE/SEMI Conference and Workshop, pages 212-217, April 2002.

[9] J. Joyner and J. Meindl, "Opportunities for Reduced Power Dissipation Using Three-Dimensional Integration," In Proceed-

ings of the IEEE 2002 International Interconnect Technology Conference, pages 148-150, June 2002.

[10] T. Kgil, S. D'Souza, A. Saidi, N. Binkert, R. Dreslinski, S. Reinhardt, K. Flautner, T. Mudge, "PicoServer: Using 3D Stacking Technology to Enable a Compact Energy Efficient Chip Multiprocessor," in Proceedings of the ACM International Conference on Architectural Support for Programming Languages and Operating Systems, October 2006.

[11] P. Lindner, V. Dragoi, T. Glinsner, C. Schaefer, R. Islam, "3D interconnect through aligned wafer level bonding," In Proceedings of the Electronic Components and Technology Conference, pages 1439-1443, May 2002.

[12] L. Liu, I. Ganusov, M. Burtscher, and S. Tiwari, "Bridging the Processor-Memory Performance Gap with 3D IC Technology," in IEEE Design and Test of Computers, Volume 22, No. 6, pages 556-564, Nov/Dec 2005.

[13] G. Loi, B. Agarwal, N. Srivastava, S. Lin, T. Sherwood, and K. Banerjee, "A Thermally-Aware Performance Analysis of Vertically Integrated (3-D) Processor-Memory Hierarchy," in Proceedings of the Design Automation Conference, June 2006.

[14] J. Mayega, O. Erdogan, P. Belemjian, K. Zhou, J. McDonald, and R. Kraft, "3D Direct Vertical Interconnect Microprocessors Test Vehicle," In Proceedings of the 13th ACM Great Lakes symposium on VLSI, pages 141-146, April 2003.

[15] P. Morrow, M. J. Kobrinsky, S. Ramanathan, C.-M. Partk, M. Harmes, V. Ramachandrarao, H.-M. Park, G. Kloster, S. List, and S. Kim, "Wafer-level 3D interconnects via Cu bonding," In Proceedings of the 2004 Advanced Metalization Conference, October 2004.

[16] S. Mysore, B. Agarwal, S. Lin, N. Srivastava, K. Banerjee, T. Sherwood, "Introspective 3D Chips," in Proceedings of the ACM International Conference on Architectural Support for Programming Languages and Operating Systems, October 2006.

[17] D. W. Nelson, C. Webb, D. McCauley, K. Raol, J. Rupley II, J. DeVale, and B. Black, "A 3D Interconnect Methodology Applied to ia32-class Architectures for Performance Improvement through RC Mitigation," In Proceedings of the 21st International VLSI Multilevel Interconnection Conference, September 2004.

[18] R. Patti, "The Design and Architecture of 3D Memory Devices," 3D Architectures for Semiconductor Integration and Packaging, Tech Venture Spring 2004 Forum.

[19] K. Puttaswamy and G. Loh, "Implementing Caches in a 3D Technology for High Performance Processors," In the Proceedings of the IEEE International Conference on Computer Design, pages 525-532, October 2005.

[20] K. Puttaswamy and G. Loh, "Implementing Register Files for High-Performance Microprocessors in a Die-Stacked (3D) Technology," in Proceedings of the IEEE International Symposium on VLSI, pages 384-389, March 2006.

[21] K. Puttaswamy and G. Loh, "The Impact of 3-Dimensional Integration of the Design of Arithmetic Units," In Proceedings of the IEEE International Symposium on Circuits and Systems, pages 4951-4954, May 2006.

[22] K. Puttaswamy and G. Loh, "Dynamic Instruction Schedulers in a 3-Dimensional Integration Technology." In Proceedings of the ACM/IEEE Great Lakes Symposium on VLSI, pages 153-158, May 2006.

[23] A. Rahman, A. Fan, and R. Reif, "Comparison of Key Performance Metrics in Two and Three Dimensional Integrated Circuits," In Proceedings of the IEEE 2000 International Interconnect Technology Conference, pages 18-20, June 2000.

[24] A. Rahman and R. Reif, "System Level Performance Evaluation of Three-Dimensional Integrated Circuits," In IEEE Transactions on VLSI, Volume 8, Issue 6, pages 671-678, December 2000.

[25] P. Reed, G. Yeung, and B. Black, "Design Aspects of a Microprocessor Data Cache using 3D Die Interconnect Technology," In Proceedings of the International Conference on Integrated Circuit Design and Technology, pages 15-18, May 2005

[26] N. Tanaka, T. Sato, Y. Yamaji, T. Morifuji, M. Umemoto, and K. Takahashi, "Mechanical effects of copper through-vias in a 3D die-stacked module," In Proceedings of the Electronic Components and Technology Conference, pages 473-479, May 2002.

[27] N. Tanaka, Y. Yamaji, T. Sato, and K. Takahashi., "Guidelines for structural and material-system design of a highly reliable 3D die-stacked module with copper through-vias," In Proceedings of the Electronic Components and Technology Conference, pages 597-602, May 2003.

[28] Y. Tsai, Y. Xie, V. Narayanan, and M. J. Irwin, "Three-Dimensional Cache Design Exploration Using 3DCacti," in Proceedings of IEEE International Conference on Computer Design, pages 519-524, October 2005.

[29] Y. Xie, G. Loh, B. Black and K. Bernstein, "Design Space Exploration for 3D Architectures," in ACM Journal of Emerging Technologies in Computing Systems, Volume 2, Issue 2, pages 65-103, April 2006.

[30] A. Y. Zeng, J. Lu, R. Gutmann, and K. Rose, "Wafer-level 3D manufacturing issues for streaming video processors," In Proceedings of the Advanced Semiconductor Manufacturing Conference, pages. 247-251, May 2004.

[31] http://www.intel.com/technology/magazine/computing/recognition-mining-synthesis-0205.htm

[32] http://www.samsung.com/PressCenter/PressRelease/PressRelease.asp?seq=20060413_0000246668

[33] http://techventure.rti.org/summer200513/

[34] http://www.tezzaron.com/technology/FaStack.htm