Design and Test of 3D-MAPS, a 3D Die-Stack Many-Core Processor

Dean L. Lewis, Michael B. Healy, Mohammad M. Hossain, Tzu-Wei Lin, Mohit Pathak

Hemant Sane, Sung Kyu Lim, Gabriel H. Loh, Hsien-Hsin S. Lee

School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332
TEL: (404) 894-9483
{dean,mbhealy,mhossain7,twlin,mohitp,hsane3,limsk,loh,leehs}@gatech.edu

ABSTRACT

3D-MAPS is a test vehicle for evaluating the architectural implications of microprocessors designed using 3D integration technology. The 3D-MAPS processor is a five-layer stack consisting of logic, cache, and DRAM layers. Testing such a 3D design presents several unique challenges. Our test architecture is a custom design, borrowing from the IEEE 1149.1 and 1500 standards. The design goals were to minimize pin count, maximize graceful degradation, and ensure complete diagnostic capability of the chip.

1. INTRODUCTION

3D integration is an exciting new manufacturing process that allows designers to stack silicon chips vertically. The chips are connected with *through silicon vias* (TSVs)—small, dense wires that are etched straight through the active silicon to create electrical connections with the neighboring layer. Proper design with these TSVs can significantly increase the on-chip device count while simultaneously reducing delay, power consumption, and chip footprint [1, 2, 3].

Arguably the hottest topic in 3D processor design is the memory-on-logic stack. Recent work has shown that a design as simple as integrating main memory on-chip significantly increases performance [4, 5, 6, 7, 8]; going a step further and rearchitecting the memory hierarchy to better exploit the bandwidth available in a 3D memory system produces even more speedup [9]. Our team at the Georgia Institute of Technology has designed and will be fabricating a many-core memory-on-logic processor for evaluating the benefits of a 3D integrated memory.

But many challenges come along with all the promised benefits of 3D integration, particularly in test [10, 11]. Lewis and Lee first presented a general test architecture for module-partitioned 3D ICs [12] and then extended the work to circuit-partitioned 3D ICs [13]. Wu et al. considered optimal 3D scan-chain ordering [14]. Zhao et al. proposed an algorithm for designing optimal 3D clock trees that enable pre-bond test [15]. Noia et al. designed test wrappers for use in 3D SOCs [16].

In this paper we present 3D-MAPS with a particular focus on the DFT capabilities of the chip. The DFT architecture itself is an implementation of the that proposed in [12]. Section 2 gives a brief overview of our chip architecture. Section 3
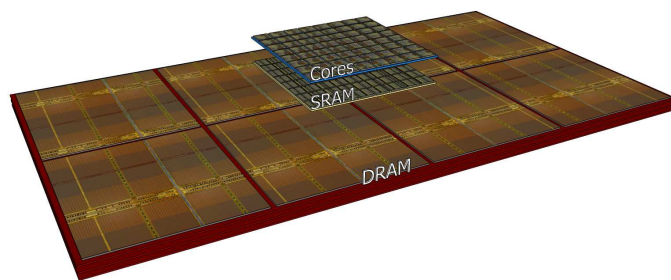

Figure 1: The 3D MAPS chip stack (width and length drawn to scale).

describes our DFT Architecture and how it addresses the challenges of 3D test. Section 4 goes into detail about implementing this test architecture in the first version of the 3D-MAPS chip. Section 5 details our methodology for verifying the DFT architecture. Finally, we conclude with Section 6.

2. ARCHITECTURE

The 3D Massively Parallel Processor with Stacked Memory (3D-MAPS) chip is a small-volume test vehicle for evaluating the benefits of 3D fabrication. The design goal was to produce a processor that could consume as much 3D bandwidth as possible and demonstrate the performance improvements expected of applications running on such a system.

The 3D-MAPS chip consists of five silicon layers as shown in Figure 1. The first two layers were designed by the Georgia Tech team. They consist of a 64-processor logic layer and a 256-bank SRAM layer. The remaining three layers are Tezzaron Semiconductor's FaStack® 3D Memory system [17]. The chip is fabbed in a 130nm six-metal bulk-Si process from Chartered Semiconductor (now part of Global Foundries) [18]. The TSV process is via-first, and the vias are $1.2\mu m$ wide by $6\mu m$ tall ($2.5\mu m$ pitch). The die-to-die (d2d) bond pads are $3.4\mu m$ wide ($5\mu m$ pitch). Global Foundries manufactures the vias and bond pads, and Tezzaron Semiconductor bonds and thins the wafers.

The instruction set architecture is a custom 32-bit two-way VLIW architecture. Each instruction bundle is 64 bits wide,

1

consisting of two 32-bit instructions. The ISA is a reduced instruction count version of the MIPS ISA. Floating-point arithmetic is not supported due to space constraints.

2.1 Processor Architecture

Each processor is a five-stage in-order VLIW machine. Each instruction bundle consists of two instructions (one arithmetic, one memory) which map to two parallel execution paths, one for arithmetic and one for memory. Keeping the memory path full is key to maximizing bandwidth utilization.

2.2 Memory Hierarchy

Each processor contains a 1.5kB (192 instruction words) instruction memory (IM) on the logic layer and a 4kB data memory (DM) on the SRAM layer. To maximize DRAM bandwidth utilization, we implemented a double-buffering scheme in the data memory. Therefore, each DM consists of four 1kB memory banks, two banks per buffer. The processor-DM data path is 32 bits wide and designed to transfer both characters (8 bits) and words (32 bits). The DM-DRAM data path is 256 bits wide (128 bits from each DM bank).

The FaStack® DRAM memory system is subdivided into eight banks, with eight cores sharing a single bank. Round-robin arbitration is used to share each DRAM channel amongst the associated processors. Each channel, running at 278MHz (DDR), provides 8.3GB/s bandwidth, or 1.0GB/s per processor; that is 66.3GB/s through all eight DRAM ports total. This is the critical number. The rest of the chip has been designed to consume as much of this bandwidth as possible to best illustrate the power of 3D design.

2.3 Manycore Architecture

The 3D-MAPS chip contains sixty-four processors arranged in an eight-by-eight grid. The processors are connected with a two-dimensional mesh network that allows for register-to-register message passing with a bisection bandwidth of 8.3GB/s.

Because this chip is a test vehicle for 3D integrated memory, there are no off-chip memory interfaces. Loading data and instructions into the chip and reading the results out is done through the test access port (TAP), discussed in Section 3.

The chip also features two AND trees: a *barrier* network for synchronizing interprocessor communication (seven cycles minimum to resolve) and a *done* network for signaling the completion of the program.

3. SECTOR DFT ARCHITECTURE

The 3D-MAPS chip is subdivided into four test cores which we call *sectors*. Each sector is composed of sixteen processor cores (in a two-by-eight array) as shown in Figure 2. Each sector is protected by a test wrapper that can isolate it from neighboring sectors in the event of a failure. This allows the chip to easily degrade to a reduced-processor-count system depending to the defects encountered.

In order to maximize the independence of each sector, the test wrapper is extended all the way to the off-chip boundary. Figure 3 illustrates the wrapper logic in the TAP; this logic is called a *sector control unit* (SCU). As shown, the SCU (and so the wrapper) is actually a 3D unit, encompassing both the processor- and SRAM-layer components of each sector. Each SCU consists of a pair of off-chip input (TDI) and output (TDO) pins.

Figure 3 also highlights the design-for-3D-test mechanisms we employed. The vertical pillars indicate d2d connections
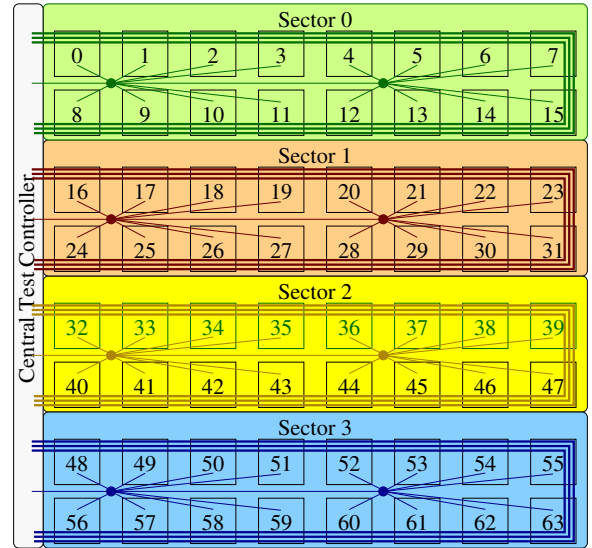

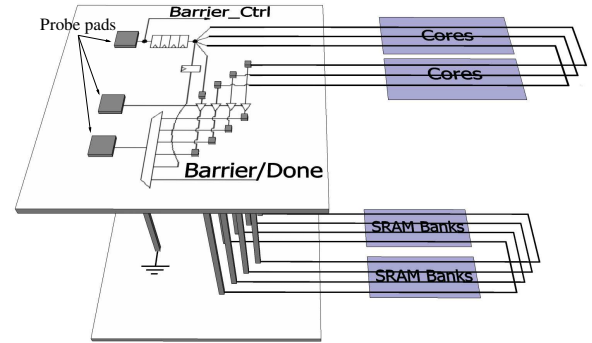**Figure 2: Sector DFT architecture.**

Figure 3: One sector control unit (the CTC contains four).

between the two layers. Testing the processor (top) layer pre-bond is easy. A third probe pad (shown middle) is used to enable tri-state drivers which short the scan chains directly to the TDO pin. Testing the SRAM (bottom) layer is a little harder. The d2d stubs are not large enough to be probed, so dummy probe pads (not shown) are required on this layer. These consist of TDI and TDO pads and mux select pads (mux not shown). Post-bond testing is effectively identical to traditional planar test with the scan chains now looping through both layers.

The SCUs are managed by a central test controller (CTC). This contains the test control state machine (TCSM) and drivers for global signals (like clock, reset, barrier acknowledge, and enable signals).

3.1 The Scan Chains

Each sector has five scan chains as shown in Figure 3. First is the *serial scan chain* (SSC). This is the actual data-carrying chain and traverses both the logic layer and the SRAM layer. It is subdivided into a set of short chains within each processor and bank according to logical devisions. The second scan chain, the *pipeline bypass chain* (PBC), selects which segments

of the SSC are bypassed during a given scan operation. For example, to load the IM and DM, we only enable the input stages to those units and bypass the rest, significantly reducing load times.

The third chain is the *core bypass chain* (CBC) which bypasses all segments of a given processor or SRAM bank in the SSC. The fourth chain is the *timing specification chain* (TSC), used to configure the DRAM controller timing.

The last chain is the one-bit *sector control chain* (SCC). This chain is internal to the SCU and is used to activate the isolation mechanisms within the test wrappers. Because of its short length, it also doubles as a quick test path for the TDI and TDO pins.

3.2   Central Test Controller

The central test controller (Figure 4) controls the operation (both functional and test) of the entire chip. Because this test chip lacks traditional off-chip memory interfaces, the CTC serves as the only connection between the processor and the outside world. Modeled after the IEEE 1149.1 test access port, the CTC contains six logical units: four identical sector control units (SCU), a custom test control state machine (TCSM), and a global control unit (GCU) (for *barrier* and *done*).

The TCSM (Figure 5) is the heart of the entire chip. For this design, instead of the traditional test command register, we chose to hard-encode the various test modes directly into the TCSM. This significantly reduced the complexity of the CTC and improved usability of the TCSM at the cost of interoperability, which was not a design requirement for this test chip.

Other key 1149.1 features remain. First, the TCSM is controlled by a single *test mode select* (TMS) signal. Second, holding TMS low will always return the TCSM to the default RESET state, within four cycles for this design.

The TCSM has eight modes encoded as shown in Figure 5. From left to right, these are processor test (one cycle), memory test (two cycles), DRAM test (132 or 137 cycles), PBC load, CBC load, SCC load, TSC load, and functional execution. These modes produce various enable and hold signals to manage the operation of the chip.

The GCU manages the final stage reduction in the *barrier* and *done* trees. It simply ANDs the four sector results together based on the SCC mask. It also supports breakpoints, described in the next section.

Finally, the CTC contains four-cycle delay registers on many paths. These delays synchronize the short-run chain signals with the global-run *enable* and *hold* signals, which require four cycles to propagate to the entire array[1]. These delays have the effect of the TCSM, TMS pin, and TDI pins operating four cycles ahead in time of the rest of the chip. This design was extensively tested at the RTL level to ensure the correctness of all this coordination.

3.3   Program Debug Features

When a program produces an erroneous result on an experimental chip like this, it is quite a challenge to determine if the problem is in the hardware or the software. To make failure diagnosis easier, we have provided hardware to support two debug modes: pipeline dumps and breakpoints. Pipelines dumps are simple; the TCSM is transitioned from execution

---

[1]Because SCC is local to the CTC, this chain must pay both local and global cycle penalties. This necessitates the *SCC_wait* state in the TCSM that no other test mode requires.

to scan, which immediately scans out the pipeline contents. This is a very powerful technique for analyzing program behavior in detail, but it only works if the exact cycle of interest is known.

Breakpoints are used when the cycle number is not known. To support breakpoints, pin TDI<0> is included in the barrier tree as mentioned previously. Disabling TDI<0> prevents barriers from resolving, converting all *barrier* instructions into breakpoints. We can then dump the memory contents before enabling TDI<0> and thus clearing the breakpoint. This is expected to be very useful for pin-pointing the problematic cycle for pipeline dumping.

3.4   3D Test Organization

Most of the scan chains and some of the global control signals require 3D implementations. There are two basic routing options for these chains. We could weave between the two tiers as the chains circle the sector, alternately stitching processor chain segments to SRAM chain segments; this would be a style of max-cut partitioning. Alternatively, we could stitch all the processor segments together and then all the SRAM segments, a min-cut partitioning of the chains. We choose the latter design for two reasons. First, minimizing the number of d2d connections in the chains minimizes the chances of the chain being broken by a 3D processing induced fault. Second, complete chains in each layer can be used in pre-bond testing with many fewer dummy probe pads, a significant cost savings.

Several global signals—most notably *clock*—require routing on both layers. The routing trade-offs for these types of signals was explored in [15]. Based on this analysis, we chose to build a fully-connected tree on both layers for because this method increases pre-bond usefulness and reduces risk at an acceptable power and routing cost.

3.5   Pre-bond Testability

For post-bond operation, obviously, all signals and chains source from the CTC. However, for pre-bond test, there are two scenarios. The first scenario applies to the processor tier. The CTC is implemented on this tier, so its functionality is available for pre-bond test. Thus pre-bond test of this layer is performed in the standard way, using the TCSM to direct the bit stream to the appropriate state bits. Tri-state drivers are used to bypass the connections to and from the SRAM layer on the three 3D scan chains as shown in Figure 3—these drivers are permanently disabled after stacking by a hardwired connection to ground.

Testing the SRAM layer pre-bond is a completely different situation because the four chains and global signals are dangling. Because the d2d bonding pads used in the Tezzaron process are too small to probe, additional dummy pads are attached to key nets to allow the ATE to apply the necessary bit streams; these pads are simply buried in the stack post-bond. Just as a scan-based architecture keeps the pin count low post-bond, it also keeps the probe pad count low pre-bond, so these pads are an acceptable cost. On the plus side, this approach allows the test structures on the SRAM layer to be integrated seamlessly with the core layer structures, no fuss required.

3.6   DRAM Test

The three-tier DRAM system actually consists of two memory layers and one logic layer for test and control. This built-
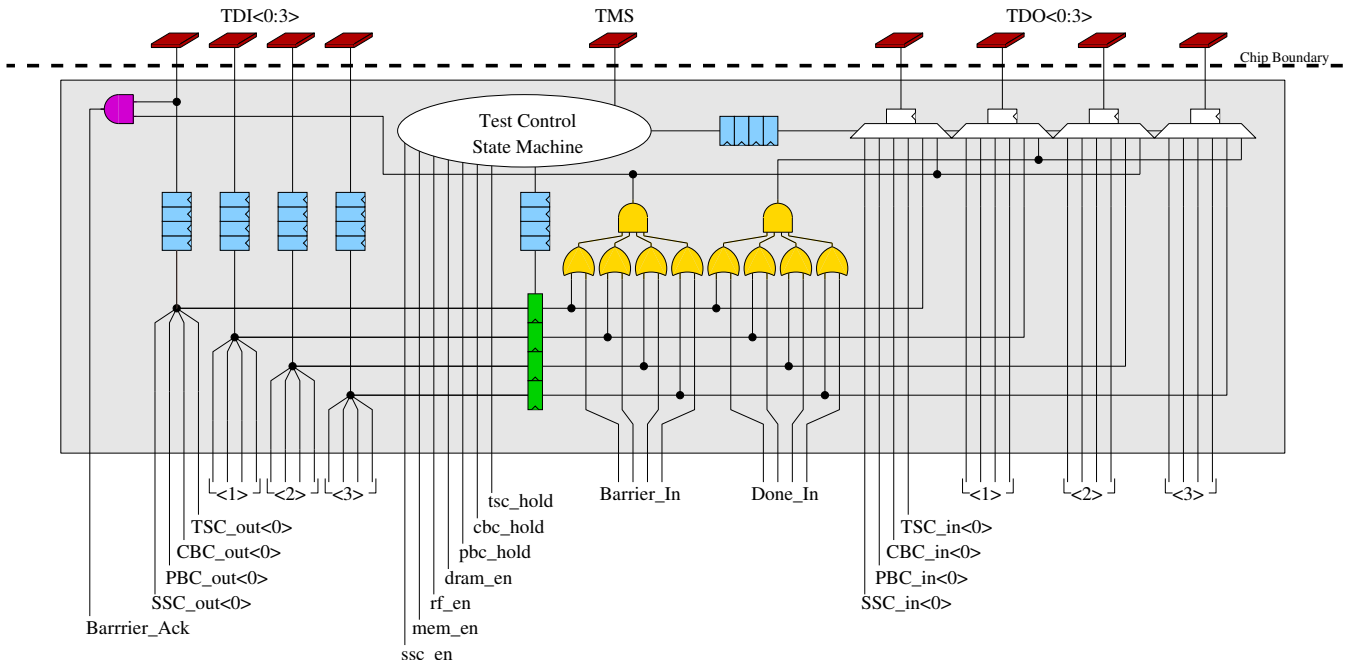
Figure 4: A circuit diagram of chip boundary, including the four SCUs, TCSM, and global control logic.

in test system is called Bi-STAR®and includes power-on self test and repair, online soft-error detection, and online repair. An embedded microprocessor manages these operations transparent to the rest of the system and so requires no support from the CTC. Bi-STAR® provides status information to the SRAM tier; we capture these reports in a segment of the SSC for evaluation off-chip.

4. IMPLEMENTATION

4.1 The First Run

The 3D-MAPS processor is a part of a multiproject wafer run for experimental 3D designs. Because of time constraints in meeting the deadlines for this run, we were not able to implement the complete architecture, as described in Section 2. Instead, we produced a reduced-functionality version (hereafter referred to as the "first run" version); the DFT hardware was also modified as necessary.

The chief architectural component missing from the first run is the DRAM. The DRAM controller and the arbitration logic were too complex to meet the fabrication deadline. As a result, we instead implemented a two-tier logic-on-memory system (Figure 6), where the SRAM scratch pad memory is now the full extent of the memory hierarchy. Additionally, the double-buffering was also removed, giving each core access to the full capacity of its memory tile at all times.

With these changes, no clocked elements remain on the SRAM tier—the control signals for the SRAM macros are all produced on the logic tier and so these macros do not require a local clock input. Given this, and given that a two-tier design is not at significantly more risk than a one-tier design, we chose to remove the test features of the SRAM tier and forego pre-bond test for the first run. Instead, the SRAM layer is tested through the logic layer post-bond. Finally, DRAM-related functionality was removed from the CTC and TCSM;
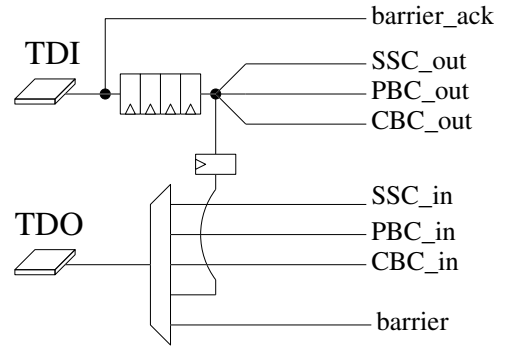

Figure 7: The first run version of the SCU.

the first-run SCU is shown in Figure 7.

Each core contains 772 flipflops, all scannable. 605 standard cells make up the CTC, consuming $3,848um^2$ of silicon. Of this, 97 gates or $820um^2$ comprise the TCSM. There are 116 F2F signal vias for the data memory bus and further 1018 vias for power and ground distribution per core. Testing the signal vias to the DM is done through a scan test of the third and fifth pipeline stages. This method does not allow us to test the vias specifically, but tracking errors to the nets the vias are on was determined to be sufficient for this test chip.

4.2 CAD Tools for DFT

As with the rest of the design process, CAD tools were used heavily to implement the test design. The following is a description of the tool flow we used to create the scan chains.

Design Compiler from Synopsys [19] was used to synthesize the design from behavioral VHDL. This included creating the preliminary scan chain from the state elements as described by
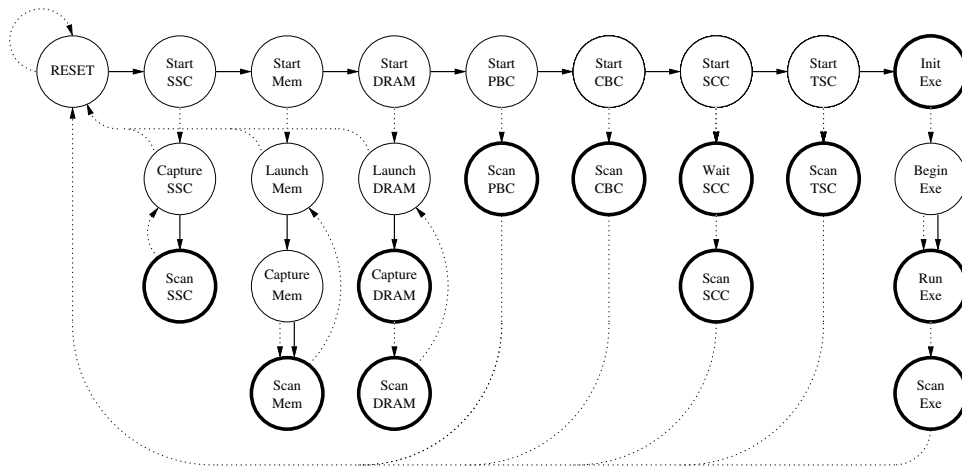
Figure 5: State diagram for the TCSM. Dashed arrows represent TMS='0' transitions, solid arrows and bolded states TMS='1' transitions.

the designer. The tool produced a gate-level HDL description of the design with the scan chains included that was used to verify the correctness of our DFT elements

Design Compiler operates without any knowledge of the physical layout. To optimize the scan chain order for routing congestion and wirelength required the Encounter tool from Cadence [20]. This produced a new HDL description that had to be verified against the older, simpler version.

Finally, the optimized scan chain order is used to produce test data bit streams from the input files to our benchmarks. This was done with a short C program developed in-house. These bit streams were executed on the compiled HDL (discussed in Section 5) to verify the correctness of the entire design.

5. SIMULATION

To verify the test architecture, we use RTL simulation and compared the results against a golden model. The RTL is extracted from the final, sign-off layout. This layout has passed a battery of checks including DRC, LVS, signal integrity and noise analysis, timing and process corners, and IR drop. It includes all standard cells inserted by the tools including gates like buffers and delay elements that do not appear in the original behavioral RTL.

The golden model was produced through an iterative process, each version of the architecture verified against the previous version. We started with benchmark kernels and expected output files, then developed a C++ architectural simulator. Next we developed a behavioral HDL model of the architecture, which was used by the CAD team to produce the actual layout and gate-level HDL description.

The layout was produced with commercial place and route tools from Cadence; we forced these tools to use scannable flipflops for *all* state bits in the design to ensure complete access to the system state. We verified the test and execution plans by running simulated tests on the HDL. Unfortunately, our benchmarks are too complex for full simulation (4.8M simulated clock cycles are required just to load the initial memory configuration), so we instead simulated just a few bytes of data, sufficient to verify the process.

6. CONCLUSION

In this paper, we presented the test architecture in 3D-MAPS, an experimental test chip for exploring the architectural impact of 3D integration. Our Sector Test Architecture serves a number of key roles. First, it allows for pre-bond test; each silicon tier can be tested independent of the other. Second, the test structures are hierarchal, so the 3D test architecture is simply a combination of the planar test features without significant additional overhead. Third, the sector architecture enables the chip to keep operating in the presence of most faults; the list of single points of failure is exceedingly small. Lastly, all this functionality is provided through a very small set of package pins. This test strategy has proven ideal for managing our experimental chip and demonstrating that 3D test can be well handled.

7. ACKNOWLEDGMENTS

8. REFERENCES

[1] K. Puttaswamy and G. H. Loh. Thermal Herding: Microarchitecture Techniques for Controlling Hotspots in High-Performance 3D-Integrated Processors. In *Proceedings of the 13th International Symposium on High-Performance Computer Architecture*, 2007.

[2] T. et al. Kgil. PicoServer: Using 3D Stacking Technology To Enable A Compact Energy Efficient Chip Multiprocessor. In *Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems*, 2006.

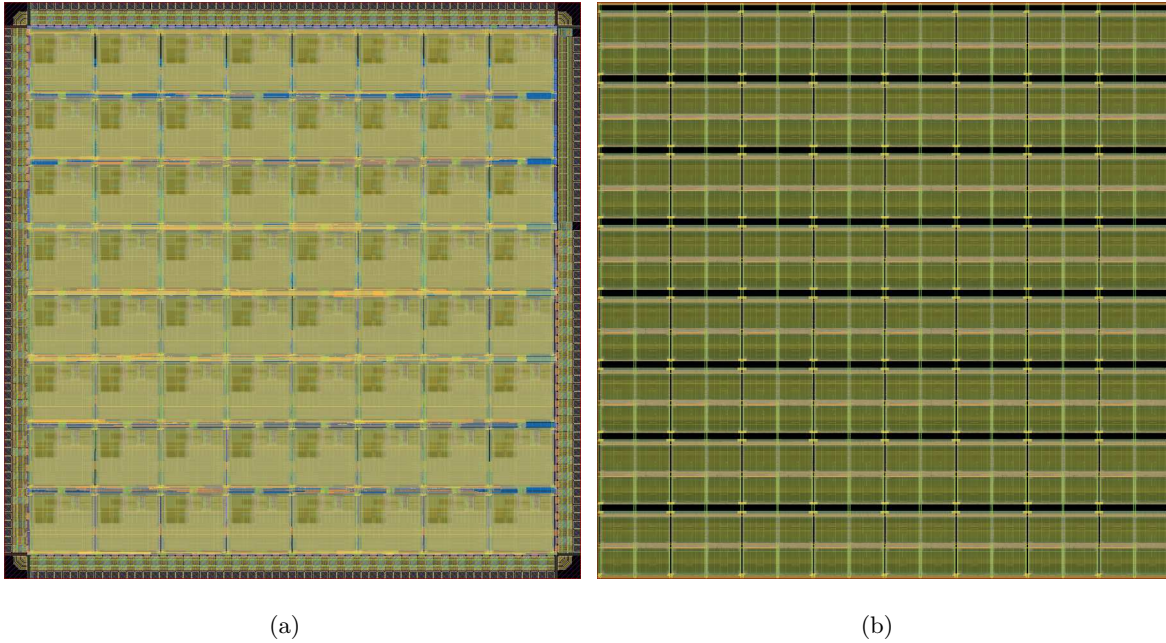[3] J. Kim, C. Nicopoulos, D. Park, R. Das, Y. Xie, V. Narayanan, M.S. Yousif, and C.R. Das. A Novel

(a)   (b)

Figure 6: Final layout of the core (a) and SRAM (b) dies.

Dimensionally-Decomposed Router for On-Chip Communication in 3D Architectures. In *Proceedings of the International Symposium on Computer Architecture*, 2007.

[4] Gabriel H. Loh. 3d-stacked memory architectures for multi-core processors. In *35th ACM International Symposium on Computer Architecture*, pages 453–464, June 2008.

[5] Guangyu Sun et al. A novel architecture of the 3d stacked mram l2 cache for cmps. In *Proceedings of the 15th International Symposium on High-Performance Computer Architecture*, February 2009.

[6] Niti Madan et al. Optimizing communication and capacity in a 3d stacked reconfigurable cache hierarchy. In *Proceedings of the 15th International Symposium on High-Performance Computer Architecture*, February 2009.

[7] Bo Zhao, Yu Du, Youtao Zhang, and Jun Yang. Variation-tolerant non-uniform 3d cache management in die stacked multicore processor. In *42nd International Conference on Microarchitecture*, December 2009.

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

[9] Dong Hyuk Woo, Nak Hee Seong, Dean L. Lewis, and Hsien-Hsin S. Lee. An optimized 3d-stacked memory architecture by exploring excessive, high-density tsv bandwidth. In *Proceedings of the 16th International Symposium on High-Performance Computer Architecture*, January 2010.

[10] Hsien-Hsin S. Lee and Krishnendu Chakrabarty. Test Challenges for 3D Integrated Circuits. *IEEE Design and Test of Computers, Special Issue on 3D IC Design and Test*, 26(5):26–35, Sep/Oct 2009.

[11] E. J. Marinissen and Y. Zorian. Testing 3d chips containing through-silicon vias. In *International Test Conference*, October 2009.

[12] Dean L. Lewis and Hsien-Hsin S. Lee. A Scan-Island Based Design Enabling Pre-bond Testability in Die-Stacked Microprocessors. In *IEEE International Test Conference (ITC)*, October 2007.

[13] Dean L. Lewis and Hsien-Hsin S. Lee. Testing Circuit-Partitioned 3D IC Designs. In *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, May 2009.

[14] Xiaoxia Wu, Paul Falkenstern, and Yuan Xie. Scan Chain Design for Three-dimensional Integrated Circuits (3D ICs). In *Proceedings of the International Conference on Computer Design*, 2007.

[15] Xin Zhao, Dean L. Lewis, Hsien-Hsin S. Lee, and Sung Kyu Lim. Pre-bond testable low-power clock tree design for 3d stacked ics. In *IEEE International Conference on Computer-Aided Design*, 2009.

[16] B. Noia, K. Chakrabarty, and Yuan Xie. Test-wrapper optimization for embedded cores in tsv-based three-dimensional socs. In *IEEE International Conference on Computer Design*, pages 70 –77, October 2009.

[17] Tezzaron Semiconductor. Fastack® stacking technology, August 2009.

[18] Global Foundaries. http://www.globalfoundries.com/, 2010.

[19] Synopsys. http://www.synopsys.com/, 2010.

[20] Cadence. http://www.cadence.com/, 2010.