Application Exploration for 3-D Integrated Circuits: TCAM, FIFO, and FFT Case Studies

W. Rhett Davis, Member, IEEE, Eun Chu Oh, Ambarish M. Sule, and Paul D. Franzon, Fellow, IEEE

Abstract—3-D stacking and integration can provide system advantages. This paper explores application drivers and computer-aided design (CAD) for 3-D integrated circuits (ICs). Interconnect-rich applications especially benefit, sometimes up to the equivalent of two technology nodes. This paper presents physical-design case studies of ternary content-addressable memories (TCAMs), first-in first-out (FIFO) memories, and a 8192-point fast Fourier transform (FFT) processor in order to quantify the benefit of the through-silicon vias in an available 180-nm 3-D process. The TCAM shows a 23% power reduction and the FFT shows a 22% reduction in cycle-time, coupled with an 18% reduction in energy per transform.

Index Terms—Fast Fourier transform (FFT), first-in first-out (FIFO), ternary content-addressable memory (TCAM), 3-D integrated circuit (IC).

I. INTRODUCTION

The combination of the technologies of wafer bonding and through-silicon vias (TSV) promises to enable scaling of electronic system performance beyond that provided just by Moore’s law. This paper explores the applications that might benefit from 3-D integrated circuit (IC) design and some of the advances in computer-aided design (CAD) needed to deliver such designs. When is it advantageous to go vertical and when is it not? Stacking two wafers together and integrating them with vertical vias is not cheap. As a rough rule of thumb, the additional processing cost is about equivalent to that of adding two additional layers of metal interconnect. This cost is even higher if individual die are stacked. This cost must be justified through performance gains or cost savings elsewhere in the system.

This cost is much greater than simply that of even “high-end” sophisticated packaging. When might it possibly be justified? Fortunately, there is a growing consensus that there are several, mainstream, circumstances which justify 3-D integration. A list of potential drivers for 3-D integration is provided in Table I.

The first and most obvious potential motivation is miniaturization. However, through-silicon 3-D integration is rarely justified by the desire for miniaturization alone. For most circumstances, if volume reduction is the only goal, then it is much more cost effective to stack and wire-bond. This technology is already in widespread use in cell phones, and continues to grow.

TABLE I

<table>
<thead>
<tr>
<th>Potential Drivers for 3-D Integration</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Driving Issue</strong></td>
</tr>
<tr>
<td>Miniaturization</td>
</tr>
<tr>
<td>Interconnect Delay</td>
</tr>
<tr>
<td>Memory Bandwidth</td>
</tr>
<tr>
<td>Power Consumption</td>
</tr>
<tr>
<td>Mixed Technology (Heterogeneous) Integration</td>
</tr>
</tbody>
</table>

Manuscript received February 12, 2008; revised June 18, 2008. Current version published March 18, 2009. This work was supported in part by the DARPA 3D IC Program through Parametric Technology Corporation (PTC), Needham, MA, and by the National Science Foundation under Grant 643700.

The authors are with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695 USA (e-mail: rhett_davis@ncsu.edu; ecoh@ncsu.edu; amsule@ncsu.edu; paulf@ncsu.edu).

Digital Object Identifier 10.1109/TVLSI.2008.2009352
shows a 22% increase in speed when implemented in three tiers, progressively improve the speed of an 8 K-point FFT. This FFT improvements in delay and energy. Section V describes an effort to achieve similar improvements in the design of a first-in first-out (FIFO) that yielded only 8% and 5% improvement from 3-D integration in three tiers. Section IV shows an attempt to achieve similar improvements in the design of a ternary content addressable memory (TCAM) that shows an impressive 23% power improvement. Because long interconnects often result from a need to access more memory, we begin our study with a focus on the memory array itself, then expand the scope to organize interconnect power [5], and a decrease in chip area. However, experience shows that many designs do not realize this in practice. Fortunately, with careful choice appropriate design applications can be found. For example, field-programmable gate arrays (FPGAs) are very interconnect bound and can achieve substantial performance and power improvements when recast in 3-D [3]. Another study of an low-density parity-check (LDPC) decoder in three tiers [6] showed a power reduction of 60% due to the reduced interconnect lengths and number of repeaters. Results obtained using two practical examples explored at North Carolina State University (NCSU) are summarized in Fig. 1, [8]. Fig. 1 compares data taken from two designs in the Lincoln Labs 3-tier 180-nm process [13] and shrinks to scaled 90- and 45-nm processes. One is a fast Fourier transform (FFT), identical to the one presented in [1]. The other is a dual core open RISC processor system-on-chip (ORPSOC) [9]. In this study, the performance benefits of 3-D integration were compared with those of technology scaling. In these examples, 3-D integration provided about the same performance advantage of two generations of technology scaling—a very compelling case.

This paper focuses on the driving issue of reducing interconnect lengths. Given that many designs do not show improvement from 3-D integration, it is important to understand why some do and others do not. This paper presents physical-design case studies (not fabricated) of 3-D-integrated designs that illustrate some of the best gains claimed to date from 3-D integration, as well as some designs that did not show significant improvement. Because long interconnects often result from a need to access more memory, we begin our study with a focus on the memory array itself, then expand the scope to organize interconnect power [5], and a decrease in chip area. However, experience shows that many designs do not realize this in practice. Fortunately, with careful choice appropriate design applications can be found. For example, field-programmable gate arrays (FPGAs) are very interconnect bound and can achieve substantial performance and power improvements when recast in 3-D [3]. Another study of an low-density parity-check (LDPC) decoder in three tiers [6] showed a power reduction of 60% due to the reduced interconnect lengths and number of repeaters. Results obtained using two practical examples explored at North Carolina State University (NCSU) are summarized in Fig. 1, [8]. Fig. 1 compares data taken from two designs in the Lincoln Labs 3-tier 180-nm process [13] and shrinks to scaled 90- and 45-nm processes. One is a fast Fourier transform (FFT), identical to the one presented in [1]. The other is a dual core open RISC processor system-on-chip (ORPSOC) [9]. In this study, the performance benefits of 3-D integration were compared with those of technology scaling. In these examples, 3-D integration provided about the same performance advantage of two generations of technology scaling—a very compelling case.

This paper focuses on the driving issue of reducing interconnect lengths. Given that many designs do not show improvement from 3-D integration, it is important to understand why some do and others do not. This paper presents physical-design case studies (not fabricated) of 3-D-integrated designs that illustrate some of the best gains claimed to date from 3-D integration, as well as some designs that did not show significant improvement. Because long interconnects often result from a need to access more memory, we begin our study with a focus on the memory array itself, then expand the scope to organize interconnect power [5], and a decrease in chip area. However, experience shows that many designs do not realize this in practice. Fortunately, with careful choice appropriate design applications can be found. For example, field-programmable gate arrays (FPGAs) are very interconnect bound and can achieve substantial performance and power improvements when recast in 3-D [3]. Another study of an low-density parity-check (LDPC) decoder in three tiers [6] showed a power reduction of 60% due to the reduced interconnect lengths and number of repeaters. Results obtained using two practical examples explored at North Carolina State University (NCSU) are summarized in Fig. 1, [8]. Fig. 1 compares data taken from two designs in the Lincoln Labs 3-tier 180-nm process [13] and shrinks to scaled 90- and 45-nm processes. One is a fast Fourier transform (FFT), identical to the one presented in [1]. The other is a dual core open RISC processor system-on-chip (ORPSOC) [9]. In this study, the performance benefits of 3-D integration were compared with those of technology scaling. In these examples, 3-D integration provided about the same performance advantage of two generations of technology scaling—a very compelling case.

This paper focuses on the driving issue of reducing interconnect lengths. Given that many designs do not show improvement from 3-D integration, it is important to understand why some do and others do not. This paper presents physical-design case studies (not fabricated) of 3-D-integrated designs that illustrate some of the best gains claimed to date from 3-D integration, as well as some designs that did not show significant improvement. Because long interconnects often result from a need to access more memory, we begin our study with a focus on the memory array itself, then expand the scope to organize interconnect power [5], and a decrease in chip area. However, experience shows that many designs do not realize this in practice. Fortunately, with careful choice appropriate design applications can be found. For example, field-programmable gate arrays (FPGAs) are very interconnect bound and can achieve substantial performance and power improvements when recast in 3-D [3]. Another study of an low-density parity-check (LDPC) decoder in three tiers [6] showed a power reduction of 60% due to the reduced interconnect lengths and number of repeaters. Results obtained using two practical examples explored at North Carolina State University (NCSU) are summarized in Fig. 1, [8]. Fig. 1 compares data taken from two designs in the Lincoln Labs 3-tier 180-nm process [13] and shrinks to scaled 90- and 45-nm processes. One is a fast Fourier transform (FFT), identical to the one presented in [1]. The other is a dual core open RISC processor system-on-chip (ORPSOC) [9]. In this study, the performance benefits of 3-D integration were compared with those of technology scaling. In these examples, 3-D integration provided about the same performance advantage of two generations of technology scaling—a very compelling case.

This paper focuses on the driving issue of reducing interconnect lengths. Given that many designs do not show improvement from 3-D integration, it is important to understand why some do and others do not. This paper presents physical-design case studies (not fabricated) of 3-D-integrated designs that illustrate some of the best gains claimed to date from 3-D integration, as well as some designs that did not show significant improvement. Because long interconnects often result from a need to access more memory, we begin our study with a focus on the memory array itself, then expand the scope to organize interconnect power [5], and a decrease in chip area. However, experience shows that many designs do not realize this in practice. Fortunately, with careful choice appropriate design applications can be found. For example, field-programmable gate arrays (FPGAs) are very interconnect bound and can achieve substantial performance and power improvements when recast in 3-D [3]. Another study of an low-density parity-check (LDPC) decoder in three tiers [6] showed a power reduction of 60% due to the reduced interconnect lengths and number of repeaters. Results obtained using two practical examples explored at North Carolina State University (NCSU) are summarized in Fig. 1, [8]. Fig. 1 compares data taken from two designs in the Lincoln Labs 3-tier 180-nm process [13] and shrinks to scaled 90- and 45-nm processes. One is a fast Fourier transform (FFT), identical to the one presented in [1]. The other is a dual core open RISC processor system-on-chip (ORPSOC) [9]. In this study, the performance benefits of 3-D integration were compared with those of technology scaling. In these examples, 3-D integration provided about the same performance advantage of two generations of technology scaling—a very compelling case.

This paper focuses on the driving issue of reducing interconnect lengths. Given that many designs do not show improvement from 3-D integration, it is important to understand why some do and others do not. This paper presents physical-design case studies (not fabricated) of 3-D-integrated designs that illustrate some of the best gains claimed to date from 3-D integration, as well as some designs that did not show significant improvement. Because long interconnects often result from a need to access more memory, we begin our study with a focus on the memory array itself, then expand the scope to organize interconnect power [5], and a decrease in chip area. However, experience shows that many designs do not realize this in practice. Fortunately, with careful choice appropriate design applications can be found. For example, field-programmable gate arrays (FPGAs) are very interconnect bound and can achieve substantial performance and power improvements when recast in 3-D [3]. Another study of an low-density parity-check (LDPC) decoder in three tiers [6] showed a power reduction of 60% due to the reduced interconnect lengths and number of repeaters. Results obtained using two practical examples explored at North Carolina State University (NCSU) are summarized in Fig. 1, [8]. Fig. 1 compares data taken from two designs in the Lincoln Labs 3-tier 180-nm process [13] and shrinks to scaled 90- and 45-nm processes. One is a fast Fourier transform (FFT), identical to the one presented in [1]. The other is a dual core open RISC processor system-on-chip (ORPSOC) [9]. In this study, the performance benefits of 3-D integration were compared with those of technology scaling. In these examples, 3-D integration provided about the same performance advantage of two generations of technology scaling—a very compelling case.
blockages for transistors and all metal layers on the top two tiers. Therefore, even though the TSVs are very small due to SOI processing, they can create significant congestion. A via-first approach could show larger gains than the one reported in this paper. However, because the TSVs account for only 2%–6% of the total area in the FIFO and FFT case studies, we think that this effect is minimal, and our results are likely to be similar, even with a via-first approach.

III. TCAM CASE STUDY

Here, we present a case-study of a TCAM array, starting first with an overview of TCAM design, followed by a discussion of 3-D TCAM designs. Next, simulation results are presented, and in conclusion, we attempt to generalize these results to other types of arrays.

A. TCAM Overview

In a CAM, data is accessed based on its content rather than its location specified by an address. A fully parallel CAM provides fast search functionality that is widely used in various applications such as lookup tables, processor caches, translation lookaside buffers (TLB), data compression, databases, and artificial intelligence [14]–[16]. Recently, CAMs with larger capacities and higher speeds have been in demand in especially high speed network applications including packet routers using longest prefix matching routing algorithms. In such applications, packets need to be classified by the most specific match from a large routing table, with possible "0" and "1," each TCAM cell stores two bits to represent these three states.

As shown in Fig. 3, a general TCAM architecture consists of a core memory array, an address decoder, a comparand data driver, a bitline (BL) data driver, a searchline (SL) pre-discharger, a matchline (ML) precharger, ML sense amplifiers, and a priority logic encoder. In a data searching operation, the search data is sent to the TCAM core to compare the comparand data with all the stored data simultaneously, and the address of the matched data is sent to the output if there is a match. In case multiple matches are found, the priority logic encoder prioritizes the addresses and sends the one with highest priority. Due to the fully parallel nature of the TCAM search operation, power dissipation is high, and lower power TCAM design is very challenging.

The high power consumption of TCAM is largely due to the comparison function used in the search operation. As shown in Fig. 4, the TCAM cell consists of a storage circuit part and a comparison circuit part used in the search operation. The TCAM core is implemented as an array of TCAM cells with horizontal running wordlines (WLs) and MLs, and vertically running BLs and SLs. All the cells in the same row share the same WLs and MLs, and all the cells in the same column share the same BLs and SLs.

TCAM write operations are implemented in a fashion similar to SRAM implementations. When WL is high, data is sent through BLs and stored in a latch. Search operations are performed when WL is low. In a fully parallel TCAM, search data is broadcast through SLs, and each TCAM cell compares the search data against the stored bit. Before every search, MLs are precharged high. If the word matches, ML remains high indicating that the data is stored in the CAM. It is considered a word match when all the cells in the word match. If any of the cells in the word do not match, the word is considered a mismatch and ML discharges to ground through the comparison logic, which is the case for most of the words in lookup table, as only a few words in the table match with the search data. Also, it should be noted that SLs are predischarged to ground before every search to avoid static current through comparison logic. In fact, approximately 50% of the SLs are charged back during evaluation. The frequent transition of MLs and SLs with large capacitances is the major source of power consumption in TCAMs [17].

B. 3-D TCAM Design

This work focuses on the reduction of the ML capacitance, which is the part of the system that is most responsible for the high power consumption. How should bitcells be partitioned in three dimensions to minimize the ML capacitance? Fig. 5 illustrates the MLs and bitcells for several partitioning options in the MITLL 3-Tier 180-nm FD-SOI process used in this work. Fig. 5(a) illustrates a cross-section of a single-tier TCAM, in which the bit-cell active areas lie below the metal match line. Fig. 5(b) and (c) show 3-D options. The upper two tiers are flipped relative to the bottom tier, which reflects the fabrication approach of this technology.

One option is to partition at the block-level, leaving the TCAM array unchanged and connecting MLs outside the array,
with ML length. HSPICE Simulations were then manually assembled with these resistances and capacitances, and compared with a simulation of a single-tier version. The results show that the ML energy per transition is reduced by 28%, with a 23% overall reduction in the array switching energy.

**D. Conclusion**

The results of this study show an impressive reduction in switching energy for this array, but how general is the result? The energy savings is due to the reduction in wire capacitance, which is due to the reduction in wire length. Simulations presented in [1] show that the capacitance per unit of length for a metal-2 line varies little between the tiers (12% difference for isolated wires, and less than 1% difference for shielded wires). Therefore, developing expressions for wire-length in block-level and bit-level partitioned designs will illustrate the energy savings. For block-level partitioning, the total ML length is equal to the number of bits times the width of each bitcell. Therefore, the total length of the ML per bit ($L_{tot}$) as a ratio of the bitcell width ($W_{bit}$) does not change with the number of tiers ($N_{tiers}$). Which can be expressed as follows:

$$L_{tot}/W_{bit}/N_{tiers} = 1.$$  \hspace{1cm} (1)

For bit-level partitioning, $L_{tot}$ will equal $W_{bit}$ plus some additional length due to the width of the TSV ($W_{via}$), and the length of the TSV itself, which we may call $L_{exp,via}$, which is the length of a wire having the same capacitance as the via

$$L_{tot} = W_{bit} + \frac{W_{via}}{2} + \frac{L_{exp,via}}{2}N_{tiers}. \hspace{1cm} (2)$$

The factor of 2 comes from the fact that each via is shared with two cells. Finally, we may rearrange this expression in terms of $L_{exp,via}/W_{via}$. This term remains relatively constant from one technology to another, because the via aspect ratio is typically limited to a value of around 5. Field solver solutions shown in [1] show this value to be about 4 for the MITLL technology. Therefore, the expression becomes

$$L_{tot}/W_{bit}/N_{tiers} = \frac{1}{N_{tiers}} + \frac{W_{via}}{2W_{bit}} \left( \frac{1}{N_{tiers}} + \frac{L_{exp,via}}{W_{via}} \right). \hspace{1cm} (3)$$

Fig. 7 shows a plot of (1) and (3). This plot shows that the bit-level partitioning scheme offers a different savings relative to block-level partitioning, depending on the width of the via relative to the width of the bitcell. This work has a $W_{via}/W_{bit}$ ratio of 0.2, which offers a savings of 24% (very close to the 28% reported in Table II). What is most notable about this plot is that it shows that designs for which the $W_{via}/W_{bit}$ ratio is greater than 0.3 or 0.4 are worse than block-level partitioning. This means that these designs will show no capacitance reduction from 3-D integration. An SRAM WL or BL, for example, which has a ratio of around 0.6 to 0.8 in this technology, is not appropriate for bit-level partitioning. It is the relatively large size of the TCAM cell that gives bit-level partitioning an advantage.

As shown in the previous section, an SRAM array is not suited for bit-level partitioning, and 3-D integration offers no capacitance reduction. However, block-level partitioning does offer
the potential for reduction of the distributed resistance-capacitance (RC) time-constant for WL and bit-lines BL by reducing the path resistance of the line. Assuming a square array and the three tiers available in the MITLL 3-D process, how close can we get to the optimum speed improvement of $\sqrt{3}$ or 42%? As shown in this section of the paper, the answer depends largely on how much design-effort we are willing to spend.

IV. FIFO CASE STUDY

We decided to explore this question in the context of a FIFO memory, because it is a building block in large digital signal processing applications, such as the FFT system described later. In order to provide the fairest comparison of the speed of 3-D-integrated FIFOs to traditional single-tier FIFOs, we decided to use the fastest FIFO architecture known, which is the multi-banked FIFO architecture using single-port memories. In order to perform the study quickly, a standard-cell methodology was used to implement the control and decode logic, with a custom layout assumed for the bit-cell array. This section continues with a review of multi-bank FIFO design, followed by a discussion of the 3-D FIFO design, simulation results, and conclusions.

A. Multi-Banked FIFO Architectures

In FIFO memories, the accesses can be controlled by special write and read pointers that increment when data is either written or read into the memory. Thus, the address decoders present in conventional RAMs can be replaced with simpler shift registers that control the Column and Row Enables for the memory, to model serial accesses of the FIFO. Specialized address pointers have been previously developed [26] for controlling the read and write accesses to the FIFO storage memory.

To provide stall-free access to the FIFO memory, it conventionally has dual-port access [18], [19], [24], thus allowing writes and reads to happen simultaneously, as shown in Fig. 8. Dual-ported memory is generally slower than corresponding identical capacity single-ported memory and also typically increases the area requirements by roughly 33% compared to the single-port memory. Increasing the number of ports also leads to increased power consumption. Thus, it is better to use single-ported memories to design FIFO structures.

Generally, using single-ported memory would cause stalling in the FIFO accesses. One method to avoid this problem is to prefetch or cache the next memory read access into a read buffer when a write access is not taking place. This can be easily achieved by multi-banking the available storage memory into 2 or more smaller memories, the combination of whose capacities would give the capacity of the original double-ported memory. The memory accesses can then be interleaved, thus simultaneously utilizing one bank of (single-port) memory for writing and the other for reading, if the corresponding read buffer is not filled. A bypass mechanism can be used to fill the read buffer directly from the write port, when the FIFO is empty, thereby allowing one read and one write per cycle.

Fig. 9 shows the schematic for such an arrangement. The write and read buffers and the bypass mechanism shown are independently present for each of the memory banks. The Controller uses the Write Demux to route the data being written into the FIFO to the proper memory bank. During each cycle, a maximum of one of the $N$ banks is written while the others can take this opportunity to refill their individual read buffers if needed. The Read Mux routes the proper Memory data to be read to the output port in the same order that it was written. The advantage of increasing the number of banks is that each memory has to be of smaller size and hence will be faster. The disadvantage is that it increases the size of the input write demultiplexer and the output multiplexer and would cause an increase in congestion and the length of interconnecting wires.

We can see in Fig. 9 that we are already eroding the benefit that we can expect from 3-D integration. By splitting the single-tier design into multiple banks, we lose the $\sqrt{3}$ performance improvement that we were expecting from shorter WLs and BLs. Still, the three-tier design will have shorter wire-lengths and should show some improvement.

B. 3-D FIFO Design

We designed two FIFO buffers of 2 and 16 kB storage capacity using the MITLL process in different configurations to
Fig. 9. Multi-banked interleaved single-port RAM FIFO.

illustrate the difference between 3-D and single-tier implementations. The first configuration uses 2 banks of 8 kb single-port memories on a single tier (2-D design). Such a configuration is as shown in Fig. 9 with two banks of memories. The second configuration uses 3 banks of 5.5 kb of single-port memories on 3 tiers. As the major portions of the design on each tier are identical and are placed right on top of one another, the lengths of the wires between different tiers is quite small. The critical path of the design lies through the memory and the maximum delay is during the read cycle of the memory. We use the 3-D IC flow described in [22] to analyze this configuration.

In each of the configurations, we use the same standard cell library to synthesize the design and get the timing, area and energy estimates. Estimates for the various bit-cell arrays were obtained from the CACTI [25] tool from HP for the 0.18-μm technology node. We used the estimates from CACTI, rather than our own bit-cell array design, because CACTI’s estimates were smaller and faster than our own design. Because most commercial bit-cell arrays are aggressively designed and ignore design rules, we felt that the CACTI estimates would give a better estimate of what was possible if we had more time and resources to refine our bit-cell array. Dummy physical memories having the aspect ratio, area, delays, and power reported by CACTI were created and used in the ASIC Design flow.

During standard-cell synthesis of the design using Synopsys Design Compiler, we constrained the arithmetic critical paths in the design (incrementing/decrementing the counters) to be less than the critical path through the memory. For the sizes of the memories and the arithmetic components used, a custom design flow would have very easily met this condition. The synthesized design was then placed and routed.

Fig. 10 shows the physical layouts and relative sizes of the last two configurations. The layouts are drawn to scale so it is easy to compare the areas. In each configuration, most of the area is occupied by the memory array. Also seen is the supporting logic made of standard cells.

C. Simulation Results

Parasitic capacitances and resistances were extracted using Cadence Encounter. The timing was then analyzed using Synopsys PrimeTime and power using PrimePower. Table III summarizes the analysis for the three configurations for the 16 and 2 kb capacity FIFOs. The energy shown is that which is consumed for one write and one read to the FIFO.

Comparing the 3-D design to the two-banked, single-tier design, the total area expanded a small amount, due to the overhead in partitioning the design into three tiers. This overhead was larger for the smaller memory. The speed improvement is not very significant: the 16 kb FIFO was 8% faster and used 6% less energy. The smaller design showed less improvement, due to the increased area overhead of partitioning. This improvement comes from the shorter wire-lengths in the 3-D design.

The speed improvement shown for the FIFO is disappointing, because clever banking was used to transform the design so that wire delays were less significant. In order to see more improvement, we need to look at designs in which wire delays play a larger role.

V. FFT CASE STUDY

FFT structures typically have long wires, to route results from one butterfly circuit to another. Previously in [1], we examined an eight-point FFT processor that showed a mere 2% speedup from 3-D integration. In this study, we wanted to pursue a much more aggressive design to see if there would be a more impressive improvement. We chose a fixed-point, 8192-point FFT primarily because it is well examined in the literature [27]–[29] and many high-speed designs are well known. We chose to implement a slight variant of these designs, which we believed would give us the best speed in the 180-nm MITLL technology. An aggressive design would likely show the best improvement from 3-D-integration, because gate delays would be relatively small compared to the wire delays.

A. Radix 2/4/8 MDC FFT Architecture

We chose the multi-path delay commutator (MDC) architecture [34] as the basis for our FFT, because it is a well understood approach for pipelining the FFT algorithm and tends to achieve the highest number of transforms per second. We chose it over
After settling on the basic MDC architecture, we had to choose the radix of the basic butterfly operations. There are many variants of the Cooley–Tukey radix-2 FFT algorithm described in the literature, each with its own signal flow graph that can be mapped into hardware. The radix-2/4/8 algorithm [27], combines three types of butterflies into one signal flow-graph, and has been used in several high-performance VLSI implementations [27], [29]. We chose the radix-2/4/8 structure, because we believed that it would give us the highest throughput.

The processing elements for the Radix-2/4/8 Butterfly (PE1, PE2, and PE3) are modified slightly from the single-delay feedback versions shown in [33]. The final MDC Radix-2/4/8 Butterfly designed for this study is shown in Fig. 11. Any FFT with a number of points divisible by 8 will require a cascade of Radix-2/4/8 Butterflies. For other FFTs, Radix-2 or Radix-2/4 Butterflies can be padded with the cascade of Radix-2/4/8 Butterflies. For this 8192-point design, we used the structure shown in Fig. 12. All multipliers read twiddle-factor data from individual ROMs which are not shown.

The FIFO memories lie inside the radix-2/4/8 butterflies as shown in Fig. 11. The FIFO capacity required is divided by two at each stage of the pipeline, and so the first stage will have two FIFOs of 2-K words, followed by 1-K words, 512 words and so on until the last stage contains a simple 1-word register of 24 bits. The complex multipliers are the biggest combinational blocks in the design and lie in the critical path. As the entire structure of the design is pipelined, it is functionally very easy to pipeline the complex multipliers and reduce the critical path delay. The complex multiplier we used follows the structure described in [35], which reduces the complexity from four real multipliers and two adders to only three real multipliers and five real adders. The fixed-point VHDL package available from Doulos [36] is used to develop the RTL code for the complex multiplier. As the entire design has a flexible pipeline structure, it is easy to modify the number of pipeline stages if needed. This can be done by adding the appropriate number of registers in the datapath. We add registers at the input and output of every complex multiplier in the design to achieve the highest clock frequency.

During the design, care had to be taken for the twiddle factor table when register stages were added or deleted from the pipeline. As the data from the input samples reaches the complex multiplier one cycle later or earlier (depending on whether a register stage was added or subtracted), the twiddle generator ROMs have to generate the twiddles one cycle earlier or later as the case may be. This can be easily accomplished by shifting the contents of the Twiddle ROMs to match the current value in the data-stream.

B. FFT Physical Design Flow

The design-flow is similar to any register transfer level (RTL)-to-layout flow using system-on-chip (SoC) encounter from Cadence, but with some inter-tier constraints. After the partitioning stage, each of the steps is a multi-tier version of the usual physical design flow. During each stage, the feasibility of the design is checked and if not found feasible, another iteration is done to tweak the design. Examples of non-feasible characteristics include overlapping cell placement, TSVs not aligned correctly between tiers, insufficient routing resources, etc.

The design-flow begins by swapping the FIFO memories in the RTL code with code including three banks of memory, using the same approach described in Section III. A single-tier version was created for comparison, using two banks. The three-tier design is then synthesized and put in to the RTL-to-GDS flow illustrated in Fig. 13. The design is first partitioned into the number of tiers available (three in our case) using K-Metis [37]. Because
K-Metis does not handle the size difference between the standard-cells and memories, the memories are removed from the netlist before partitioning, and added back in afterwards. After partitioning, the top module contains only three sub-modules, corresponding to each tier. No cells are moved between tiers after this point.

These three pseudo-independent designs are then taken through the rest of the 3-D physical design flow. An initial unconstrained placement is done individually on each of the tiers using Encounter. The three placed designs are then manually studied and modified to get the next iterated placement. Modifications involve manually changing the locations of memories to get a better floorplan. Each bank in the FIFO memories is placed in the same X-Y position. This is accomplished by first placing the memories on tier B and copying these locations to Tiers A and C. After the memories are pre-placed, the TSVs are placed on Tier B and again the corresponding locations are copied onto Tiers A and C. The memories already placed in the tiers act as blockages during this placement. Next, the rest of the standard cells are placed individually on each tier. The clock trees are then individually synthesized and the three tiers are independently routed.

The placement approach described above sacrifices some performance, because no attempt is made to minimize the lengths of wires that span tiers. This performance loss is limited somewhat by the fact that the placement algorithm attempts to keep cells close to the manually placed FIFO banks, which are placed in the same position on each tier. We have not yet been able to quantify the performance loss, because the 3-D placement algorithms under development [38] do not currently support memory blockages.

Next, three sets of parasitics are extracted into three SPEF files. Also, a netlist is produced for each tier that contains the clock and reset tree buffers. The three netlists along with the three SPEF files are then read into Synopsys PrimeTime and timing is reported. Similarly, they are read into Synopsys PrimePower and power is reported.

Fig. 14 shows the final placed version of Tiers A–C. The overall density for Tier A is 63%, for Tier B is 67% and that for Tier C is 66%. Fig. 14 shows the final routed versions of Tiers A–C, respectively.

For the single-tier design, the floorplan was made as close as possible to the multi-tier design. The most significant difference was that for every FIFO memory on a particular tier in the multi-tier design, there were two sub-banks of memories to be placed in the single-tier design. For each of the FIFO memories, the two sub-banks were placed close to each other, in order to reduce the lengths of the wires connecting these two subbanks and the
controllers. As in a usual EDA flow, the single-tier placement is followed by the routing and clock tree synthesis steps in that order and the performance of this design is compared with the three-tier approach.

C. FFT Physical Verification Results

The design-flow used for this study is identical to the flow that was used to fabricate chips in the MITLL three-tier process during the 2006–2007 run. Chips fabricated on that run have been tested and are working properly, but have unfortunately not yet been tested at full speed to check the validity of our timing and power verification flow. Therefore, the performance estimates listed here are useful only for relative comparison of the single-tier and three-tier designs.

The performance characteristics of the single-tier and three-tier designs are compared in Table IV. First, we compare the performance of the single-tier design with previously published work to ensure that we are using a sufficiently aggressive basis for comparing single-tier and three-tier performance. Designs with larger feature sizes are normalized using the method of constant electric-field scaling and the scaling factors shown. As can be seen from this table, the FFT designed here is 8×–10× faster. These other designs were intended for baseband signal processing and were not pipelined as aggressively. Also, the use of the MDC architecture provides a shorter execution time at the expense of area. Also of note in the table is that the use of the MDC architecture provides a shorter execution time in delay and energy, due to the fact that the single-tier design was accelerated with clever use of banks. An 8 K-point FFT was shown that achieves a 22% reduction in cycle time when implemented in 3 tiers, which is perhaps the best improvement in speed shown to date for an end-user application in 3-D integration.

Although these improvements are impressive, they are still not enough to motivate the use of TSVs. Although the semiconductor industry is moving toward 3-D integration, the fabrication of TSVs is costly. If speed and power improvements of 20%–25% are the best that can be achieved with the use of TSVs, then we may find that companies will simply stack die and route signals to the chip periphery. The performance improvement afforded by TSVs is useful only for the highest performance designs, in which memory latency is usually the bottleneck. It remains to be seen whether or not designers of those high-performance systems will think that TSVs are worth the cost.

VI. Conclusion

This paper has shown case studies of circuits design in a 3-D process with through-silicon vias in order to help quantify their benefit. A TCAM was shown that achieved an impressive 23% power improvement from 3-D integration in three tiers. A FIFO memory was shown that yielded only 8% and 5% improvements in delay and energy, due to the fact that the single-tier design was accelerated with clever use of banks. An 8 K-point FFT was shown that achieves a 22% reduction in cycle time when implemented in 3 tiers, which is perhaps the best improvement in speed shown to date for an end-user application in 3-D integration.

The authors would like to thank Cadence, Synopsys, PTC, and Ansoft for generously providing the CAD tools for this work. They would also like to thank MIT Lincoln Labs for providing access to their FD-SOI technology and for their aid in developing our design kit. They would also like to thank J. Stine at Oklahoma State University for generously providing access to the OSU-SoC standard-cell characterization scripts, which provided the basis for their library.

ACKNOWLEDGMENT

TABLE IV

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Radix</td>
<td>12</td>
<td>12</td>
<td>11</td>
<td>12</td>
<td>12</td>
<td>0.0%</td>
</tr>
<tr>
<td>Process (µm)</td>
<td>2/4/8</td>
<td>2/4/8</td>
<td>2/4/8</td>
<td>2/4/8</td>
<td>2/4/8</td>
<td>-0.0%</td>
</tr>
<tr>
<td>Scaling Factor</td>
<td>3.33</td>
<td>2.77</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-15%</td>
</tr>
<tr>
<td>Voltage</td>
<td>3.3</td>
<td>3.3</td>
<td>1.8</td>
<td>1.5</td>
<td>1.5</td>
<td>-15%</td>
</tr>
<tr>
<td>Area (Normalized) (mm²)</td>
<td>107 (9.63)</td>
<td>100 (13.0)</td>
<td>4.84</td>
<td>8.17</td>
<td>10.6</td>
<td>+30%</td>
</tr>
<tr>
<td>Clock Frequency (Normalized) (MHz)</td>
<td>20 (66.6)</td>
<td>20 (55.5)</td>
<td>20</td>
<td>167</td>
<td>214</td>
<td>+28%</td>
</tr>
<tr>
<td>Power (Normalized) (mW)</td>
<td>650 (58.5)</td>
<td>600 (77.7)</td>
<td>25.2</td>
<td>385</td>
<td>404</td>
<td>+5%</td>
</tr>
<tr>
<td>Exec. Time (Normalized) (µs)</td>
<td>400 (120)</td>
<td>400 (144)</td>
<td>717</td>
<td>24.4</td>
<td>19.0</td>
<td>-22%</td>
</tr>
<tr>
<td>Energy/Transform (Normalized) (µJ)</td>
<td>260 (7)</td>
<td>240 (11.2)</td>
<td>18.1</td>
<td>9.39</td>
<td>7.68</td>
<td>-18%</td>
</tr>
<tr>
<td>EDP (Normalized) (µJ-s)</td>
<td>104 (0.84)</td>
<td>96 (1.61)</td>
<td>13.0</td>
<td>0.23</td>
<td>0.15</td>
<td>-36%</td>
</tr>
</tbody>
</table>

This paper has shown case studies of circuits design in a 3-D process with through-silicon vias in order to help quantify their benefit. A TCAM was shown that achieved an impressive 23% power improvement from 3-D integration in three tiers. A FIFO memory was shown that yielded only 8% and 5% improvements in delay and energy, due to the fact that the single-tier design was accelerated with clever use of banks. An 8 K-point FFT was shown that achieves a 22% reduction in cycle time when implemented in 3 tiers, which is perhaps the best improvement in speed shown to date for an end-user application in 3-D integration.

Although these improvements are impressive, they are still not enough to motivate the use of TSVs. Although the semiconductor industry is moving toward 3-D integration, the fabrication of TSVs is costly. If speed and power improvements of 20%–25% are the best that can be achieved with the use of TSVs, then we may find that companies will simply stack die and route signals to the chip periphery. The performance improvement afforded by TSVs is useful only for the highest performance designs, in which memory latency is usually the bottleneck. It remains to be seen whether or not designers of those high-performance systems will think that TSVs are worth the cost.
REFERENCES


W. Rhett Davis (S’92–M’02) received the B.S. degree in electrical and computer engineering from North Carolina State University, Raleigh, in 1994 and the M.S. and Ph.D. degrees in electrical engineering from the University of California at Berkeley, Berkeley, in 1997 and 2002. Since 2002, he has been with the Department of Electrical and Computer Engineering, North Carolina State University, where he is currently an Associate Professor. He has worked briefly with Hewlett-Packard (now Agilent), Boeblingen, Germany, and Chameleon Systems, San Jose, CA. His research interests are centered on developing methodologies, CAD tools, and circuits for systems-on-chip in emerging technologies. His interests include 3-D IC design and low-power and high-performance circuit design for digital signal-processing and embedded systems.

Eun Chu Oh received the B.S. degree (summa cum laude) in electrical and computer engineering and the M.S. degree in electrical engineering from North Carolina State University, Raleigh, in 2000 and 2001, respectively, where she is currently pursuing the Ph.D. degree in electrical engineering. Her research interests include 3-D IC circuit techniques and design, memory design, and low-power and high-performance circuit design for digital systems in emerging technologies. Miss Oh is a member of Phi Kappa Phi, Eta Kappa Nu, and Tau Beta Pi. DAVIS et al.: APPLICATION EXPLORATION FOR 3-D INTEGRATED CIRCUITS 505

Authorized licensed use limited to: North Carolina State University. Downloaded on April 6, 2009 at 23:53 from IEEE Xplore. Restrictions apply.
Ambarish Mukund Sule received the Bachelor of Engineering degree in electronics engineering from University of Mumbai, Mumbai, India, in 1999 and the M.S. and Ph.D. degrees in electrical and computer engineering from North Carolina State University, Raleigh, in 2003 and 2008, respectively. Since 2007, he has been with Qualcomm, Raleigh, NC, where he works on embedded systems. He has worked briefly with Texas Instruments, Bangalore, India.

Paul D. Franzon (SM’99–F’06) received the Ph.D. degree from the University of Adelaide, Adelaide, Australia, in 1988. He is currently a Distinguished Alumni Professor with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh. He was with AT&T Bell Laboratories, DSTO Australia, Australia Telecom, and two companies he cofounded, Communica and LightSpin Technologies. His current research interests center on the technology and design of complex systems incorporating VLSI, MEMS, advanced packaging, and molecular electronics. Application areas currently being explored include novel advanced packaging structures, network processors, SOI baseband radio circuit design for deep space, on-chip inductor and inductance issues, RF MEMS, and moleware circuits and characterization. He has led several major efforts and published over 120 papers in these areas.

Dr. Franzon was a recipient of an NSF Young Investigators Award in 1993, in 2001 was selected to join the NCSU Academy of Outstanding Teachers, and in 2003 was selected as a Distinguished Alumni Professor.