VLSI Design in the 3rd Dimension

Stephen Strickland
Erhan Ergin
David R. Kaeli
Paul Zavracky

Department of Electrical and Computer Engineering
Northeastern University
Boston, Massachusetts

Abstract

Recently, the need for increased circuit complexity has outpaced our ability to perform efficient routing and placement, while still maintaining small die sizes. Part of this problem can be attributed to the limits imposed by designing in two dimensions. Three-dimensional VLSI circuits, obtainable through using a transferred thin-film process, can provide a path for realizing complex structures, while reducing route lengths and die sizes.

In this paper we report on our new three-dimensional design technology, devices which we have fabricated, and our move to an automated 3-D design path using Cadence and Synopsys design tools. We present both discussion of functional 3-D devices, and on-going work on VLSI design tools and the implementation of a 3-D microprocessor.

keywords: 3-D microelectronics, VLSI CAD, commericial tools, layout and routing

1 Introduction

The desire for increased circuit complexity is currently pushing two-dimensional (2-D) VLSI technology to its limits, causing problems such as large die sizes and non-negligible delays introduced by long power rails and data busses. Because of this, several schemes have been suggested to minimize the effects of interconnects on circuit performance. Some of these schemes include:

1. exploiting performance-driven placement and routing [5, 11, 13, 16, 17],
2. utilizing multiple interconnect layers [18],
3. providing interconnect scaling [14],
4. using interconnect materials with very low resistivity [15], and
5. using better interlayer dielectrics with low permittivity [15].

One promising technology, which has received a lot of attention in the past in Japan is 3-D electronics [6, 9, 19]. If we allow designers to stack active device layers on top of one another, and if we provide the capability of locating vias between layers of devices anywhere on the design, we can overcome many of the barriers encountered with designing in 2-D.

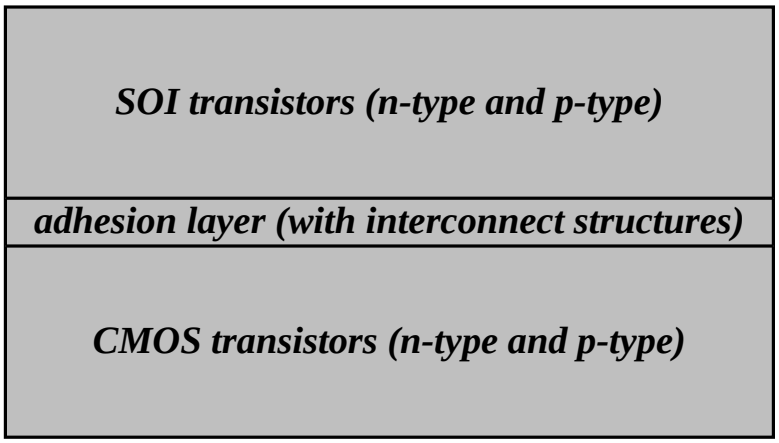

Figure 1: CMOS & SOI Transistors on a 3-D I.C.

In this paper we present a new design technology which minimizes the effects of long routing wires and at the same time, increases transistor density. Three-dimensional (3-D) interconnects are used between stacked transistor layers, thus utilizing the parallelism which is inherent in 3-D topologies. 3-D VLSI designs can effectively reduce or eliminate routing delays by providing for large, compact, circuits, and short, direct, interconnections. We have already been successful in fabricating functional devices using this technology.

Several different approaches have been taken to designing in 3-D. Some investigators are thinning and stacking silicon die [2, 9]. Using this approach, interconnections between levels are accomplished at the perimeter of the die. Akiyama et al. used laser recrystallization to create layers of single crystalline silicon above completely fabricated circuits [1]. Each subsequent layer would be processed to create an additional circuit function. In this type of circuit, interconnects between layers are not limited to the perimeter of the die. In the 1980's, Fan et al. began to commercially develop thin film transfer processes. These techniques were applied to the development of devices with mixed technologies (GaAs and Si) and were proposed as the basis for creating a three-dimensional circuit [4, 20].

We have chosen to pursue the development of three-dimensional circuits based on the transferred thin film approach. The basis of this technique is the ability to align and transfer a completed circuit from one substrate to another. The process involves fabricating a part of the circuit on a Bulk silicon substrate and a second part on an SOI substrate. The SOI circuit is removed from its associated oxide layer, inverted and bonded to the Bulk silicon wafer using epoxy. Once transfer is complete, an insulating oxide layer is grown. The bonding and SOI layers are etched to create vias to the metal-2 layer on the Bulk wafer. Contact to the SOI metal-2 layer is accomplished by etching the upper oxide layer. This process can be continued on additional layers, if desired. Figure 1 provides an illustration of our 3-D approach.

In this paper we will discuss the design of devices using this technology. In Section 2 we review some of the limits imposed by 2-D design. In Section 3, we discuss the issues involved when designing in 3-D, as well as our current design flow. In Section 4 we present modeling results, illustrating the merits of 3-D design. In Section 5 we summarize the important outcomes of this work.

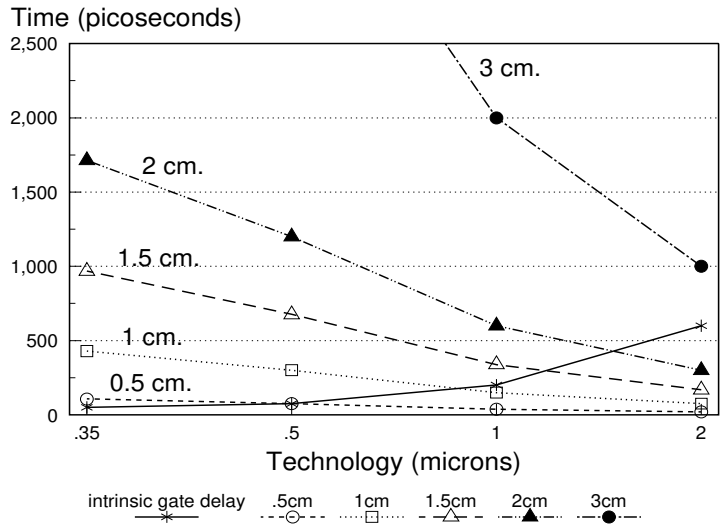

Figure 2: Routing Delays vs. Technology: routing delays for route lengths of .5cm, 1cm, 1.5cm, 2cm and 3cm, as computed for 2μ, 1μ, $.5\mu$ and $.35\mu$. technologies.

2 2-D vs. 3-D VLSI

To better understand the limitations of 2-D VLSI on today's state of the art submicron technologies, we need to look at the effects of scaling on circuit characteristics. Miniaturization of MOS device technologies has been, and continues to be, a popular method to improve circuit performance. One of the most apparent effects of miniaturization is on the source of circuit delays. In larger technologies (i.e., $> 1\mu$), the delays due to routing wires are negligible since gate delays dominate the overall circuit performance. As technology is scaled down, gates become faster. At the same time the interconnect delays increase due to the increased effects of parasitic elements. As a result, routing delays become the dominant factor in circuit performance. In Figure 2 we plot the intrinsic gate delay for 2μ, 1μ, $.5\mu$ and $.35\mu$ technologies, and compare this to the routing delays encountered as a function of route length.

As can be seen, when we push the limits of VLSI to sub-micron levels and as we get closer to the limits of miniaturization schemes, different avenues are being employed to further increase the performance of integrated circuits. 3-D VLSI allows us to stack transistors on top of one another, separated by an insulating material. Electrical connection between the stacked layers is provided by 3-D vias which run along the z-direction.

There are a number of apparent advantages of 3-D VLSI over 2-D VLSI. The first is that if a design is partitioned properly, then interconnect lengths can be greatly reduced. Utilizing the third dimension, buses which would normally be routed across the design to connect two functional units can now be routed in the z-direction, up to another design level. By reducing interconnect lengths, the overall delay introduced by routing is also reduced. Second, partitioning the design on two or more levels can effectively reduce the overall chip footprint. This reduction in chip area can lead to increased yield during the fabrication process (assuming we can provide adequate yield for our 3-D technology). This should also allow us to implement larger and more complex designs without the limitations

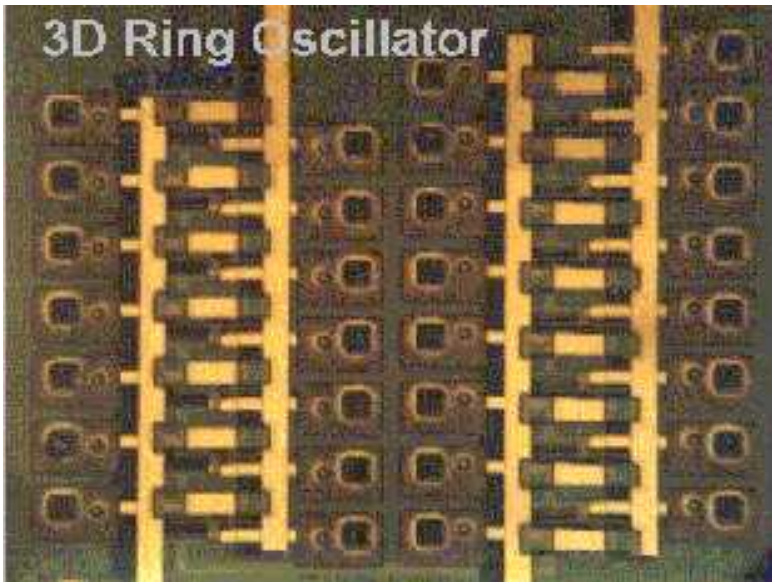

Figure 3: SEM photograph of a 31 stage ring oscillator

introduced by designing in 2-D.

A third advantage of this technology is its ability to combine two different technologies on a single chip. In our current design we are using bulk silicon and SOI, and combining these on a single chip. One can envision combining designs from multiple vendors into a multi-level, single chip design. By defining fixed locations for interconnects, each level can be fabricated using a different process and then transferred to make a single chip (this is somewhat oversimplified, but in concept, is an attractive idea).

To date, we have fabricated a 31-stage ring oscillator on a test chip, and have working devices. This has allowed us to make a number of measurements which were critical when developing a standard cell library for our current design of a more complex device. We are near completion of a 32-bit DLX microprocessor named *YIFAN* [1] which is implemented using our 3-D technology. In the next section we will discuss issues which were considered in our 3-D VLSI and describe the design flow used to develop our design.

3 Implementation

We began this project by first designing a simple test chip which would allow us to characterize the performance of this technology. To this end, we have successfully fabricated a 31-stage ring oscillator, as shown in Figure 3.

In order to exploit the benefits of a new technology, we need to first understand its limitations as well as its advantages. We list here some of the key issues which need attention during 3-D design. In our current design technology (1μ):

- the minimum sized 3-D vias are 6μ by 6μ. This suggests that a 3-D via will occupy at least $36\mu^2$ on both layers (excluding the alignment margins). As the number of 3-D vias used in a design increases, the area overhead introduced by these vias will be

[1] YIFAN Is Fabricated At Northeastern

significant. In some cases this overhead can limit the area reduction expected from 3-D VLSI. As our via technology matures, this will become less of an issue.

- Heat must be removed from the Bulk layer which is buried under the SOI layer. This is the main reason we chose to use a Bulk wafer for the lower level instead of an SOI wafer. For an SOI wafer, the insulator will not transfer the heat as efficiently as a Bulk wafer can. If two SOI wafers were used then the transistors would be functioning between two insulators and could overheat. If this occurs, some 3-D vias might be used as cooling vias. This is an important issue to investigate as we extend the design to more than two layers.

- The 3-D vias have to be aligned carefully during placement, as well as during processing. This suggests that new placement and routing algorithms might be needed to optimize the design for area and performance, which take into account the alignment technology employed.

- The placement of the pads can be placed on the Bulk or SOI wafer. If they are placed on the Bulk, then the equivalent area on the SOI wafer must be left unused, and contacts can be made to the pads. An alternative choice is to use the same location on the SOI. This would leave the area on the Bulk wafer available for devices instead of pads. Placement of a separate wafer which would only contain pads could also be implemented. This would allow pads to be placed over the whole wafer which would reduce the chip size needed for a pad frame.

Evaluating these key issues, we find that the Bulk/SOI 3-D process can either be used to maximize performance or density. If performance is given priority, the designer can take advantage of the parallelism which 3-D vias provide, using as many vias as needed. On the other hand, if the die size is the limiting factor, we can vary the number of 3-D vias and evaluate the resulting chip size.

With the help of these preliminary observations, we designed *YIFAN*, the first 3-D microprocessor. Our design objective was to optimize area as well as performance. In the next section, we will show the design flow used for *YIFAN*.

3.1 Design Flow

Our objective in this project was to go one step further than present designs and implement a large, complex circuit in 3-D. The reasons behind selecting a large design were two-fold. First, we want to provide a design which would fully exploit the potential of our new technology. Second, we wanted a device which would establish the feasibility of this new technology for the design and manufacture of real devices (versus implementing a few gates or latches).

With these goals in mind we designed a 32-bit, fully pipelined, RISC microprocessor for the DLX instruction set described in [8] and [10]. In our DLX implementation, we chose not to include an on chip cache, nor a floating point unit, for the first generation microprocessor. These units take up considerable area, and at a 1μ design point, we would quickly run out of area.

We chose the DLX instruction set so that we would have compilers and assemblers to test our processor once it is completed. The DLX instruction set is complete, concise and powerful. We partitioned the design between the SOI and Bulk layers, taking into

consideration the critical paths of the design and the number of 3-D vias. We placed the ALU and the general purpose register file on the SOI layer in order to take advantage of the speed benefits of SOI devices. We have two internal busses (S1 and S2) running between the layers combining the elements of the datapath. The control signals are generated at the Bulk level and carried to the SOI level through 20 3-D vias. There are a total of 200 3-D vias, excluding power and ground vias. The I/O pin count of the chip is 135, excluding extra pins added for testability. The transistor count is 111,450. We targeted a maximum die size of 1 cm^2. This allows us to include features which will provide testability.

To implement our design we used Cadence and Synopsys design tools. Our design flow progressed as follows:

1. Code up the design in an HDL (we have used VHDL in this project). This step will include the preliminary 3-D partitioning of the design. The structural and behavioral HDL is then simulated to establish functionality (RTL level simulation).

2. Next, synthesize the HDL code down to a gate level netlist. Simulate the gate level netlist (using gate level simulation). The standard cell library is well characterized with timing views. This will allow us to check for correctness and performance of the synthesized design.

3. Reevaluate the 3-D partitioning according to timing information. Repartition the design if the schematics do not meet design specifications.

4. At this point the netlist for the design is complete and can be handed off to the backend. This is where the actual layout will be generated. There will be multiple netlists generated, one per layer.

5. We then floorplan the dictating layer (the layer which will determine where the 3-D vias will be placed). This involves the placement of functional units and 3-D interconnects. In our YIFAN design, the SOI layer is chosen as the dictating layer. This choice is based on the fact that the 3-D vias have to pass completely through this layer, which places heavy restrictions on routing and placement for this layer. Once the locations of the 3-D vias are fixed they are used as constraints in the dictated layers.

6. After initial floorplanning is done for the dictating layer, the 3-D via locations are fixed. SDF (Standard Delay Format) backannotation is performed and resimulation is done with estimated wire delay information. Although this information is estimated, it will be fairly accurate due to the fact that 3-D via locations are fixed, thus fixing the wiring scheme. We rerun floorplanning until the simulations meet the design specifications.

7. The placement of the 3-D vias are then extracted. This extracted information is then run through a preprocessing program to simulate the flipping of the dictating layer and generates a new floorplan file (the SOI is "flipped" as part of the transfer process).

8. The floorplanning of the second layer is performed as soon as the constraint file (the result of previous step) is available. This step is similar to the dictating layer floorplanning, with the exception that the 3-D interconnects are already placed.

9. Placement of each individual layer is then performed in parallel as the 3-D vias are already placed. We need each placement algorithm to keep the locations of the 3-D vias fixed.

10. At this point, clock distribution can take place if needed (clock tree synthesis) and this information is also backannotated to the frontend where it is simulated for correctness.

11. After placement, a final 3-D partitioning overview is performed to analyze 3-D via placement.

12. After the cells are placed at each layer, another SDF back annotation is performed, now considering information from both layers. The two timing data files are also back annotated to the synthesizer, and simulations are rerun. The two separate netlists are then combined to form a single design with the interconnect wire loads adjusted to represent the complete wire. At this point, timing information can be manipulated in order to test the performance of the design for various 3-D via schemes (e.g., highly resistive and capacitive vias, vias which behave as regular wires).

13. Finally, each design is routed independently. 3-D vias remain at fixed locations.

14. A third SDF back annotation is performed. The output of this step should be very close to the previous SDF output because the wiring scheme is pre-determined by the 3-D via locations.

15. As a last step, we perform Electrical Rule Checking (ERC), Design Rule Checking (DRC) and run Layout Versus Schematic (LVS). A mask set is then generated.

All simulations are performed using Verilog-XL, which is found in Cadence. SDF is obtained using the following methods. For less accurate information, data can be extracted directly from the floorplanning tools. Because this data is less accurate (i.e., no wire information has been produced yet) the simulations run faster. This level of information provides the designer with initial feedback on placement and floorplanning. For more detailed information, we need to extract the parasitics of the design. The formats which are used to represent these parasitics are: Standard Parasitic Format (SPF) which is very close to Spice format, Reduced Parasitic Format (RSPF), and RC format which is a basic RC network. The SPF model is the most precise model, whereas the RC network presents the least accuracy. Parasitic extraction can either be performed by the floorplanner, once routing information is available, or by a stand-alone parasitic extractor like Dracula, Vampire, Diva or Arcadia. This extracted data can be fed into delay calculators like Central Delay Calculator (CDC) to get SDF data, which in turn can then be used by static timing analyzers like Perl, Motif (or Motive), or, in our case, by a simulator like Verilog-XL.

4 2-D versus 3-D Area and Routing

Next we present results from 2-D and 3-D layouts, studying how technology scaling affects chip and routing area using these two layout schemes. We chose three designs to evaluate this question. Our choices were based on the complexity of the designs, the amount of routing required, the size, and the availability of the designs. The designs are:

1. a 32-bit ripple-carry adder,

Technology	2μ	1μ	0.5μ	0.25μ
metal1 width	3λ	3λ	4λ	4λ
metal2 width	3λ	3λ	4λ	5λ
metal1 separation	3λ	3λ	4λ	4λ
metal2 separation	4λ	4λ	5λ	6λ

Table 1: Design rules used for metal wires

2. a 32-bit integer ALU (used in *YIFAN*), and

3. a general purpose register file (also use in *YIFAN*).

The 32-bit adder is the smallest and simplest design in the set. The 32-bit ALU is the most complex design. The register file is the largest design. To get a better understanding of how 3-D VLSI will reduce the routing lengths, we used a channel router with no feedthru cells. This helps us isolate the routing area used in the design. To calculate the area used for routing, we calculated the total area occupied by the standard cells, then subtracted this from the total chip area. We exclude the area used for the I/O pads in our calculations. This is a reasonable method to calculate the routing area because in channel routers such as Cell Ensemble from Cadence, the design is divided between cell regions and channels. The channels are used exclusively for routing. Therefore, since we did not allow thru-cell routing, this would represent the routing area.

The first layout is implemented using a 2μ standard cell library. We scaled this design down to 1μ using MOSIS lambda rules, which simply follows ideal scaling. We also layed out the design in 0.5μ and 0.25μ technologies. For these sub-micron technologies, we needed to use a non-ideal scaling scheme. Metal1 and metal2 wires are made wider and their separation is increased to reduce parasitic resistance and capacitance [3]. This type of scaling is necessary for submicron technologies due to the increase in side-wall capacitances and crosstalk. When scaling 3-D vias, we could not maintain lambda-based rules (i.e., these features are not scalable). Taking into account alignment, deep-etch and via-fill technologies, we estimated the sizes of 3-D vias for 1μ, 0.5μ, and 0.25μ technologies. During this estimation, we tried to keep the contact hole sizes as large as possible, and scaled the alignment margins. Large 3-D interconnects have negligible parasitic resistances and capacitances. Table 1 shows the design rules used for the metal1 and metal2 wires for all the technologies that we used. The design rules for our 3-D vias can be seen in Figure 4.

4.1 32-Bit Ripple-Carry Adder

The first functional block we will look at is a 32-bit ripple-carry adder. In this adder, we used 288 2-input nand gates. The total number of transistors, excluding the I/O drivers, is 1152. We kept the design as simple and as uniform as possible.

As we look at the percent of chip area used for routing in these technologies, we see that there is a steady increase as we go from larger device technologies to smaller ones. This result was expected as routing wires are scaled non-ideally. As we can see from the areas in Table 2, a significant percent of the chip area is used for routing (even though these numbers go down by about 15% when we allow over the cell routing and introduce feedthru

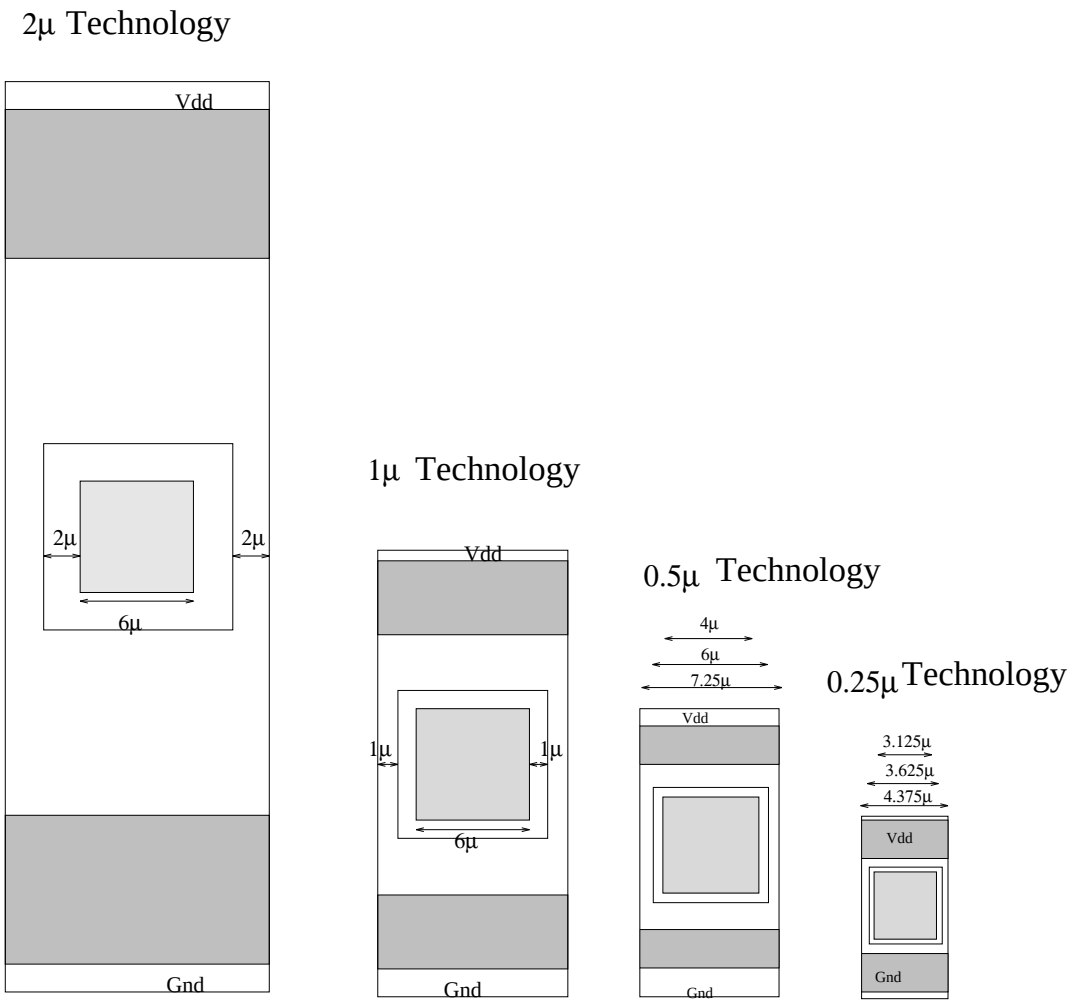

Figure 4: Estimated 3-D via Design Rules

Technology	2μ	1μ	0.5μ	0.25μ
Total Area (μ^2)	2234959.75	611823.90	186706.75	71164.96
Total Cell Area (μ^2)	800640.00	200160.00	53167.50	17614.18
Total Routing Area (μ^2)	1434319.75	411663.90	133539.25	53550.78
% Used for Routing	64.18 %	67.28 %	71.53 %	75.25 %

Table 2: Results of 32-bit Ripple-carry Adder 2-D Layouts

Technology	2μ	1μ	0.5μ	0.25μ
Total Area (μ^2)	1100611.62	276042.20	80387.74	28810.86
Total Cell Area (μ^2)	400320.00	100080.00	26583.75	8807.09
Total 3-D Via Area (μ^2)	49623.00	17375.00	6253.13	2105.47
Total Routing Area (μ^2)	650668.62	158587.20	47550.87	17898.30
% Used for Routing	59.1 %	57.45 %	59.15 %	62.12 %

Table 3: Results of 32-bit Ripple-carry Adder 3-D Layouts (single layer)

Technology	2μ	1μ	0.5μ	0.25μ
Effective Chip Size Reduction	50.75 %	50.73 %	56.9 %	59.52 %
% Reduction in Design Area	1.51 %	1.47 %	13.89 %	19.03 %
% Reduction in Routing Area	9.27 %	10.62 %	28.78 %	33.15 %
% Overhead (introduced by 3-D vias)	4.44 %	5.68 %	6.70 %	5.92 %

Table 4: Comparison of 2-D and 3-D Adder Layouts

cells). These results suggest that interconnection technology is certain to become one of the bottlenecks in VLSI design in the near feature.

In order to compare 2-D VLSI and 3-D VLSI, we laid out the 32-bit ripple-carry adder for the Bulk/SOI 3-D process. We partitioned the design into two 16-bit adder blocks, and included 3-D vias which provide electrical connection between the two layers. During partitioning, our first priority was to minimize design size by reducing routing area. To achieve this, we used a limited number of 3-D vias (50 to be exact). In cases where performance is the first priority, the number of vias can be increased (for instance the ripple carry bits can be routed between layers adding an extra 16 3-D vias). We used the scaling scheme discussed earlier for wire width, separation, and 3-D vias. Table 3 shows the results of our 3-D layouts for the adder. These area measurements are obtained from a single layer of the layout. The second layer has identical characteristics as the design is partitioned into two identical parts. The only difference between the upper and lower layers is that, the I/O pads are placed on the lower layer. As can be seen from these figures, the die size is reduced by more than 50% in all technologies, and the percent of chip area used for routing is reduced on average by 20%. Furthermore, the percent reduction in routing area increases as we scale down the device technology. These results support our initial observations on the advantages of 3-D VLSI.

In Table 4, we compared the results of our 2-D and 3-D adder layouts. The most significant result in this table is the amount of reduction in design area. In these 3-D layouts, the effective design area is twice the total area figures shown in Table 3 (the two layers in our 3-D layouts occupy the same area). When we compare the total design area in 2-D and 3-D layouts, we see that there is a reduction in favor of 3-D layouts (which increases as we move to sub-micron technologies). Considering that the area used by standard cells does not change from the 2-D scheme to the 3-D scheme, and including the overhead introduced by 3-D vias in the 3-D layouts, we can get a better picture on the reduction of design area obtained when moving from one scheme to another. The results for the sub-micron

Technology	2μ	1μ	0.5μ	0.25μ
Total Area (μ^2)	76275319.61	17397053.44	6446785.56	2202920.98
Total Cell Area (μ^2)	8671175.00	2167793.75	576272.72	171434.98
Total Routing Area (μ^2)	67604144.61	15229259.69	5870512.84	2031486.00
% Used for Routing	88.6 %	87.54 %	91.06 %	92.22 %

Table 5: Results of 32-bit Integer ALU 2-D Layouts

technologies exhibit this advantage. Also note the rate of reduction in routing area for sub-micron technologies. Table 4 suggests that 3-D VLSI is more effective as the portion of the design used for routing increases. This result also supports our theory that the benefits of 3-D VLSI will be more apparent in smaller device technologies. The amount of overhead introduced by 3-D vias in sub-micron technologies is caused by non-ideal scaling of these vias. Once our 3-D via process is completely characterized, we expect dramatic changes in these numbers.

4.2 32-Bit Integer Arithmetic Logic Unit

Next, we consider the 32-bit ALU implemented in our *YIFAN* design. This design consists of 3004 standard cells. The total number of transistors in the ALU is 14904. We implemented signed and unsigned add, subtract, and, or, xor, arithmetic left shift, arithmetic right shift, logical left shift, and logical right shift for integers and bit-vectors in our ALU. We consider this design to be the most complex among the designs in our test set, although it is not as large as the register file. We used Synopsys's Design Analyzer synthesis tool to implement the gate level design, and Cadence's Cell Ensemble place and router to layout the design. The area measurements of the 2-D layouts are shown in Table 5. We can see that as the technology is scaled down, similar to the 32-bit adder, the percent of total chip area used for routing increases (except for 2 and 1μ technologies). Another important characteristic we need to pay attention to is the overall increase in the routing area as the design gets more complex. Compared to the ripple-carry adder, total routing area increased by about 20 % in all of the technologies. One reason for this is that the ripple carry adder is a design of repetitive single bit adder components. This allows the placement and routing algorithms to do a reasonable job at minimizing the routing. But with an ALU, the design uses more random logic and therefore the placement and routing of the design is more complex. As we include feedthru cells and pins and use different optimization algorithms, the area devoted to routing will be reduced by 10% on average.

Due to the fact that the gate level representation of the ALU is obtained through synthesis, we could not partition the design with a schematic editor. Instead, we synthesized a 16-bit ALU which implements the same functions as the 32-bit ALU, and added 3-D vias to connect it to the top layer which also contains an identical 16-bit ALU. These two implementations may have slight differences as we do not have total control over the synthesis (behavioral synthesis). However, they are logically identical, and thus, area comparisons between the 2-D and 3-D representations are valid. The results of our 3-D measurements are shown in Table 6.

When we look at Table 6, we see that (as expected) the percent of area used for routing decreased for all 3-D layouts. However, the improvements are consistent across all tech-

Technology	2μ	1μ	0.5μ	0.25μ
Total Area (μ^2)	25313997.00	5752698.38	2275575.30	696031.56
Total Cell Area (μ^2)	4422086	1105521.50	292083.74	86823.91
Total 3-D Via Area (μ^2)	53515.00	10112.50	6628.31	2316.05
Total Routing Area (μ^2)	20838396.00	4628064.38	1976863.25	606891.60
% Used for Routing	82.32 %	80.45 %	86.87 %	87.19 %

Table 6: Results of 32-bit ALU 3-D Layouts (single layer)

Technology	2μ	1μ	0.5μ	0.25μ
Effective Chip Size Reduction	66.8 %	66.93 %	64.70 %	68.4 %
% Reduction in Design Area	33.63 %	33.87 %	29.40 %	36.81 %
% Reduction in Routing Area	38.35 %	39.22 %	32.65 %	40.3 %
% Overhead (introduced by 3-D vias)	0.14 %	0.22 %	0.21 %	0.21 %

Table 7: Comparison of 2-D and 3-D ALU Layouts

nologies, unlike in the adder layouts where smaller technologies benefited more from the 3-D technology. One reason for this is that we did not use any compaction or optimization on any of the layouts, and the smaller device technologies might benefit more from these options. Another reason for this behavior was the fact we could not ideally scale our 3-D vias.

In Table 7, we compare the 2-D and 3-D ALU layouts. The reduction in chip size for this complex design (complex compared to the adder) is significantly higher in all technologies. The percent reduction in design area increased significantly compared with the results for the adder layouts. This data supports our conjecture that as designs get larger and more complex, they can take better advantage of 3-D VLSI. Just like the chip size comparison, the reduction in routing area also supports this theory. Compared with our results from the 32-bit adder section, the percent reduction in routing area increased by an average of 17.13%. One puzzling characteristic that we can observe in this comparison is again the even distribution of improvements throughout the technologies. In fact, the sub-micron technologies did not take as much advantage of 3-D VLSI as larger technologies. Our explanation for this is that even though we split the design onto two levels and tried to reduce the routing, it is not true that we removed all long routes. Since we used Synopsys to generate the two separate schematics it was not possible for us to analyze the routing and divide the ALU by hand over the two layers such that the longer routes were removed first.

4.3 YIFAN General Purpose Register File

The last design in our test set is the general purpose register file that we designed for YIFAN. This design consists of 16 32-bit registers and the control circuitry needed to multiplex the inputs to the registers. It is the largest design in our test set with 3346 standard cells, and 25512 transistors. It also requires intensive routing. The gate level schematic is generated using the Synopsys VHDL synthesizer similar to the 32-bit integer ALU. Table 8 shows the

Technology	2μ	1μ	0.5μ	0.25μ
Total Area (μ^2)	94333693.35	22835370.28	7634349.00	2772901.45
Total Cell Area (μ^2)	13584292.00	3396073.00	864592.35	242093.26
Total Routing Area (μ^2)	80749401.35	19439297.28	6769756.65	2530808.19
% Used for Routing	85.60 %	85.13 %	88.67 %	91.27 %

Table 8: Results of the General Purpose Register File 2-D Layouts

Technology	2μ	1μ	0.5μ	0.25μ
Total Area (μ^2)	45774062.42	9358428.08	3537017.21	1173817.14
Total Cell Area (μ^2)	6792146.00	1698036.50	432296.18	121046.63
Total 3-D Via Area (μ^2)	60326.00	21545.00	7753.88	2610.78
Total Routing Area (μ^2)	38921590.42	7638846.58	3096967.16	1050159.73
% Used for Routing	85.03 %	81.63 %	87.56 %	89.47 %

Table 9: Results of the General Purpose Register File 3-D Layouts

2-D layout measurements on the register file. One thing to notice is that the area used for routing is approximately the same as the ALU. This suggests that there is an upper limit on the routing area, even as the complexity and size of the design increases. This limit is about 85-88% of the total design area. As we include feedthru cells and pins, the routing area percentage will go down by an average of 15%. This is higher than the ALU because our register file design is an array of common structures and can be placed in such a way as to optimize feedthru usage.

The register file was then split into two equal register files (8 32-bit registers each), along with their required control circuitry. 3-D vias were added to combine the two units to make the equivalent register file as the 2-D implementation. Since, the two layers are logically identical and thus area comparisons between the 2-D and 3-D representations are valid. The results of our 3-D measurements are shown in Table 9.

Table 10 shows us that the register file did not fare as well when moving to 3-D as the other designs did. One reason for this is the way the register file was split to two levels. Splitting the register file into 8 32-bit registers on each level does not reduce the routing significantly. This is because there is no routing between registers. So, by splitting the register file this way the only routing that was reduced was: (a) if any wire was routed around the removed logic or (b) the control signals now go vertical to a second set of registers instead of horizontally.

4.4 Discussion

From the results produced from our test set we can make some observations. The first is that 3-D VLSI technology can reduce the overall routing area of a design substantially. The second, which is more important, is that the overall gain of the technology is heavily dependent on how well the design has been partitioned. The goal is to remove the long routes and to optimize area usage. This can not be done today with the conventional design tools which are optimized for 2-D VLSI technology.

Technology	2μ	1μ	0.5μ	0.25μ
Effective Chip Size Reduction	51.48 %	59.02 %	53.67 %	57.67 %
% Reduction in Design Area	2.95 %	18.04 %	7.3 %	15.34 %
% Reduction in Routing Area	3.6 %	21.41 %	8.51 %	17.01 %
% Overhead (introduced by 3-D vias)	0.13 %	0.19 %	0.20 %	0.19 %

Table 10: Comparison of 2-D and 3-D General Purpose Register File Layouts

Currently we are working on the final steps to completing the *YIFAN* design. Because of tool limitations the design flow was changed slightly to support the 3-D process. Because we are using standard design tools at this time we had to create two separate designs in Cadence which represented the two layers of the microprocessor. This is because the tool does not understand 3-D VLSI technology. In order to complete the design we started with the top layer first, placing and routing this layer. This fixes the interconnect locations, design functionality for each layer, and the minimum design size, even before we have considered other layers.

Our approach was to begin work on the top level design until we were satisfied with the layout and performance. Then we transferred the locations of the interconnects to the bulk wafer and completed the second design. This would align the interconnects on both levels. The only problem is that the interconnects' locations were only optimized for the SOI layer instead of both layers. This was because a channel router was used to place standard cells. The problem this causes is that when you route the design, the route will travel through the channels between each row of standard cells. These routing channels will expand or shrink, as needed for routing. Since each layer is independent of other layers, this movement will not be reflected onto the other layers and therefore would perturb the interconnect alignments. To get the correct output loads on the schematics we extracted the input load on one layer and added this to the output load of the second layer.

4.5 Future Work

We are currently pursuing some new ideas for 3-D routing and placement. Some related issues we plan to pursue include:

1. providing scalable design rules for 3-D vias,
2. mixing different technologies on different layers,
3. improving our 3-D characterization,
4. providing cost analysis for 3-D, and
5. considering 3-D thermal modeling

5 Summary

Our purpose in writing this paper was to introduce a new VLSI technology, look at its advantages from a designer's standpoint, and pass on our experiences with 3-D VLSI to the reader. In order to accomplish this, we identified some limitations of the current technology

and showed how 3-D VLSI can improve upon and eliminate some of these problems. To demonstrate the advantages of 3-D VLSI, we layed out three circuits in both 2-D and 3-D, and compared their layout characteristics. Although most of these characteristics are process and design specific, we can still get a good idea on what to expect from similar technologies. We showed that 3-D VLSI helps reduce chip size by reducing the amount of area used for routing. We also showed that 3-D VLSI improves circuit performance by reducing interconnect wire lengths.

Previous research on 3-D VLSI performance evaluation focused mostly on comparing off-chip interconnection technologies to 3-D vias [7, 12]. We believe that our Bulk/SOI 3-D technology can produce results that should be compared not only with off-chip interconnect performance, but also with on-chip interconnection performance.

We pointed out that sub-micron technologies would benefit more from 3-D VLSI compared to larger technologies. Our basis for this argument was that in smaller device technologies chip delays are dominated by interconnect delays. To avoid long interconnect delays, in some scaling schemes, the interconnection wires are non-ideally scaled. In these technologies, routing area dominates overall chip area. As 3-D VLSI introduces a new dimension in routing, designs that require intense routing will benefit the most. If we look at the current trend in VLSI design, we see that the complexity and size of chips nearly doubles from one generation to the next. This observation alone motivates future exploration using 3-D VLSI.

5.1 Acknowledgements

We would like to acknowledge the contributions of Philip Sailer and Navid Atoofi on the development of a 3-D design library for the Magic toolset, the generation of the 3-D ring oscillator design, and the initial VHDL implementation of YIFAN. We would like to also acknowledge the contributions of Keith Warner, DongMei Li and Piyush Singal for the development and processing used in our 3-D ring oscillator. This research is support by the Office of Naval Research and Kopin Corporation.

References

[1] S. Akiyama, S. Ogawa, M. Yoneda, N. Yoshii, and Y. Terui, "Multilayer CMOS Device Fabrication on Laser Recrystallized Silicon Islands", *Technical Digest, IEEE IEDM*, pp. 352-355, 1983.

[2] "Cubic Memory, Inc., Scotts Valley, CA".

[3] H. B. Bakoglu, "Circuits, Interconnections, and Packaging for VLSI", Addison-Weley Publishing Company, Inc., 1990.

[4] B.D. Dingle, M.B. Spitzer, R.W. McClelland, J.C.C. Fan, and P.M. Zavracky, "Monolithic Integration of a Light-Emitting Diode Array and a Silicon Circuit Using Transfer Processes," *Appl. Phys. Lett.*, 62(22), May 1993.

[5] Alfred E. Dunlop and Brian W. Kernighan, "A Procedure For Placement of Standard-Cell VLSI Circuits," *IEEE Transactions on CAD/I-CAS,*, CAD-4(1), January 1985, pp. 92-98.

[6] Satoshi Fujita, "A Template Matching Algorithm Using Optically-Connected 3-D VLSI Architecture," *Proceedings of the 14th Annual International Symposium on Computer Architecture*, June 1987, pp. 64-70.

[7] Stefan A. Kuhn, et al., "Performance Modeling of the Interconnect Structure of a 2-Dimensionally Integrated RISC-Processor/Cache-System", *Proc. of IEDM*, pp. 592-599, 1995.

[8] J. Hennsessy and D. Patterson, *Computer Architecture a Quantitative Approach*, Morgan Kaufman Publishers, 1996.

[9] Y. Hayashi, S. Wada, K. Kajiyana, K Oyama, R. Koh, S. Takahasi and T. Kunio, "Fabrication of Three-Dimensional IC's Using Cumulatively-Bonded IC (CUBIC) Technology," *Symposium on VLSI Technology*, 1990.

[10] P. Sailer and D.R. Kaeli, *The DLX Instruction Set Handbook*, Morgan-Kaufmann, San Mateo, CA, 1996.

[11] B.W. Kernighan and S. Lin, "An Efficient Heuristic Procedure For Partitioning Graphs," Bell System Technical Journal, 49(2), 1970, pp. 291-308.

[12] M. B. Kleiner, et al., "Performance Improvement of the Memory Hierarchy of RISC-Systems by Application of 3-D-Technology", *Proc. of IEDM*, pp. 645-655, 1995.

[13] C.Y. Lee, "An Algorithm For Path Connections and Its Applications," IRE Transactions on Electronic Computers, EC-10, September 1961, pp. 346-365.

[14] Dong-mei Li, "Interconnect Process for 3-Dimensional IC," Department of Electrical and Computer Engineering, MS Thesis, Northeastern University, May 1995.

[15] Shyam P. Murarka, "Low Dielectric Constant Materials For Interlayer Dielectric Applications," *Solid State Technology*, March 1996, pp. 83-90.

[16] R. Nair, C.L. Berman, P.S. Hauge and E.J. Yoffa, "Generation of Performance Constraints for Layout," *IEEE Transactions on Computer Aided Design*, CAD-8(8), 1989, pp. 860-874.

[17] Carl Sechen and Alberto Sangiovanni-Vincentelli, "The TimberWolf Placement and Routing Package," *IEEE Journal of Solid State Circuits*, SC-20(2), April 1985, pp. 510-522.

[18] M. Stallmann, T. Hughes and W. Liu, "Unconstrainted Via Minimization for Topological Multilayer Routing," *IEEE Transactions on Computer Aided Design*, CAD-9(9), 1990, pp. 970-980.

[19] Sheridan Tatsuno, "Japan's Push Into Creative Semiconductor Research: 3-Dimension IC's," *Solid State Technology*, March 1987 pp. 29-30.

[20] P.M. Zavracky, M.M. Zavracky, J.C.C. Fan, and J.P. Salerno, "Optoelectronic Interconnection for III-V Devices on Silicon", U.S. Patent No. 4,890,895, January 2, 1990.