Area-Efficient Multiplier Designs Using a 3D Nanofabric Process Flow

Authors Omitted for Double Blind Review
Affiliations Omitted for Double Blind Review
Email Omitted for Double Blind Review

***Abstract*—In the past few years, the demand for computationally intensive applications, such as digital signal processing or convolutional neural networks, has grown exponentially. As they often rely on a significant number of multipliers, it is crucial to optimize their performances, and more particularly, their area. Recently, a 3D Nanofabric flow has been proposed, where logic circuits are designed by stacking N identical vertical tiers on top of each other. They are then processed in a similar fashion as the Vertical-NAND flash, where several layers can be patterned at once. While the 3D Nanofabric flow presents several layout constraints (single metal routing and identical vertical layers), it can decrease the area by around one order of magnitude, leading to area-efficient and cost-effective circuits. In this paper, we propose to use the 3D Nanofabric process flow to design low-area multipliers. As multipliers can be designed using a regular array organization, we show how they can be spread across multiple vertical layers using the 3D Nanofabric flow, while respecting the different layout constraints. We then provide thorough circuit-level evaluations, including parasitics, to showcase the benefits of our proposed 3D multipliers at the circuit-level. In particular, we show that by stacking up to 64 layers to build a 64-input bit multiplier, the area and area-delay-product can be decreased by $35.1\times$ and $31.4\times$, respectively, when compared to a conventional 12nm FinFET implementation, with only a 35% power consumption overhead. We also show that our proposed multipliers are thermally reliable through 3D cooling simulations.**

I. Introduction

In the past 50 years, the semiconductor industry has been defined by transistor scaling, making it possible to pack more devices within the same area and thus resulting in integrated circuits of higher functionality and complexity. However, conventional scaling has become increasingly difficult due to physical, technological, and economic limitations [1], [2]. As a result, this is leading designers and technologists to explore alternative routes.

Recently, 3D logic integration has been perceived as a viable candidate to alleviate conventional scaling limits, and two schemes have been proposed: parallel 3D, where wafer or dies are separately processed and then stacked on top of each other [3]–[5] and sequential 3D [6]–[10], in which stacked devices are fabricated sequentially on top of each other on a common substrate. Exploiting the third dimensions can improve the footprint without requiring costly feature size reduction, while also decreasing the interconnect length. However, parallel 3D is limited by the large *Through Silicon Vias* (TSVs) pitch ($\sim 1\mu m$ [11]). On the other hand, sequential 3D enables smaller vertical interconnect pitches, but the number of stacked vertical tiers is limited by the manufacturing cost. To this date, state-of-the-art sequential 3D works [6]–[10] are employing 2 to 4 active tiers. In order to stack many more tiers, a recent work [12] proposed an innovative 3D integration scheme, called 3D Nanofabric. Aimed at logic applications, the flow consists of N identical vertical tiers. It is inspired by the *Vertical Nand* (V-Nand) process [13], where multiple layers can be patterned at once, decreasing the manufacturing costs. While a significant area reduction can be achieved using the 3D Nanofabric, it presents several physical design constraints (identical tiers and single routing layer). Hence, it is well suited for regular logic circuits, such as multiplier arrays.

Multipliers are a crucial operator in modern digital systems as they are heavily used in *Convolutional Neural Networks* (CNNs) [14]) or *Digital Signal Processing* (DSP) applications [15]. As multipliers are usually large blocks, improving their area can significantly impact the total chip area, especially for machine learning accelerators where large arrays of multiply-and-accumulate units are used [16], [17]. In this paper, we introduce area-efficient multipliers using the 3D Nanofabric process recently proposed in [12]. We first show how multipliers can be spread across several vertical layers using the 3D Nanofabric flow while respecting the different layout constraints. We then provide thorough circuit-level evaluations, considering parasitics, to showcase the benefits of our proposed 3D multipliers. In particular, we show that by stacking up to 64 layers to build a 64-input bit multiplier, the area and area-delay-product can be decreased by $35.1\times$ and $31.4\times$, respectively, when compared to a conventional 12nm FinFET implementation, with only a 35% power consumption overhead. Besides, we show through 3D thermal simulations that our proposed multiplier is reliable, even for 64 layers.

The rest of this paper is organized as follows: Section II provides some background on different 3D integration schemes. Section III introduces our proposed 3D multiplier. Section IV present our experimental results. Section V concludes this paper and briefly discusses our future works.

II. Background

In this section, we provide a brief background on traditional 3D logic integration schemes and the 3D Nanofabric flow.

A. Conventional 3D Integration Schemes

Several works on parallel 3D have been proposed [3]–[5], where wafers or dies are vertically stacked and interconnected.

This is achieved using μ-bumps and *Through-silicon Vias* (TSVs) [18]. μ-bumps mature solutions offer 3D interconnect pitches in the range of $20\mu m$, and are currently used in commercial products such as *High Bandwidth Memory* (HBM) [3] and 2.5D passive interposers for GPUs [4]. A more aggressive TSV pitch, in the range of a few nanometers [5], can be achieved using direct hybrid bonding [19], rather than μ-bumps, and is currently used for smart imaging sensors [5]. However, parallel 3D is limited by the large pitch of TSVs ($\sim 1\mu m$ [11]). On the other hand, sequential 3D consists of stacked devices that are fabricated sequentially on top of each other [6]–[10]. Sequential 3D has several applications, such as stacking logic on logic or memory on logic [7] or stacking emerging technologies on top of CMOS [10]. Nevertheless, only a limited number of vertical tiers have been currently demonstrated [10], as stacking many tiers would result in significant costs of fabricati

B. 3D Nanofabric

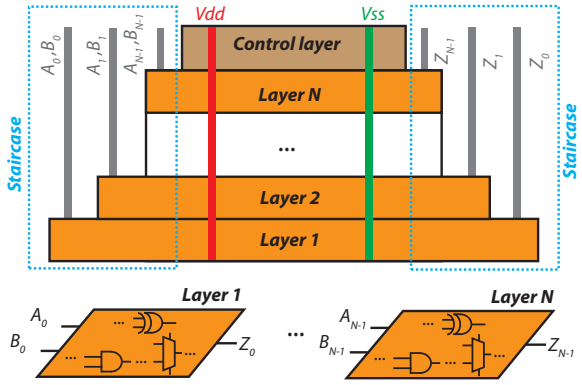

Fig. 1. 3D Nanofabric overview. Note that each vertical layer is physically identical in terms of layout.

Recently proposed [12], the 3D Nanofabric consists of N physically identical vertical layers stacked on top of each other, as illustrated in Fig. 1. As for V-NAND chips [13], primary inputs and outputs (A_i, B_i and Z_i) can be provided by staircases while global signals (V_{ss}, V_{dd}) are coming from a top logic control layer through vertical pillars, spanning among all vertical tiers. Inspired by the V-Nand process flow, the 3D Nanofabric is, however, aimed at logic applications. By stacking many logic tiers, a significant footprint reduction can be obtained (up to $16\times$ for a 32-bit arithmetic logic unit [12]). As it consists of a repetitive stack of layers, they can potentially all be patterned at once, resulting in decreased manufacturi sts. On the other hand, the thors of [12] showed that the 3D Nanofabric presents sever yout constraints: (i) all the vertical layers should be identical from a layout perspective; (ii) due to process flow restrictions, ly one metal routing layer can be used. Thus, we believe that such 3D integration scheme is well suited for multiplier arrays due to their regular organization and simple intra nections. Besides, the authors in [12] only focused on the area benefits of the 3D Nanofabric while omitting delay and power analyses. Our paper intends to fill this gap.

III. Proposed 3D Multiplier

In this section, we show how traditional N-bit multiplier arrays can be designed in 3D, using the 3D Nanofabric flow.

A. Multiplier Array Overview

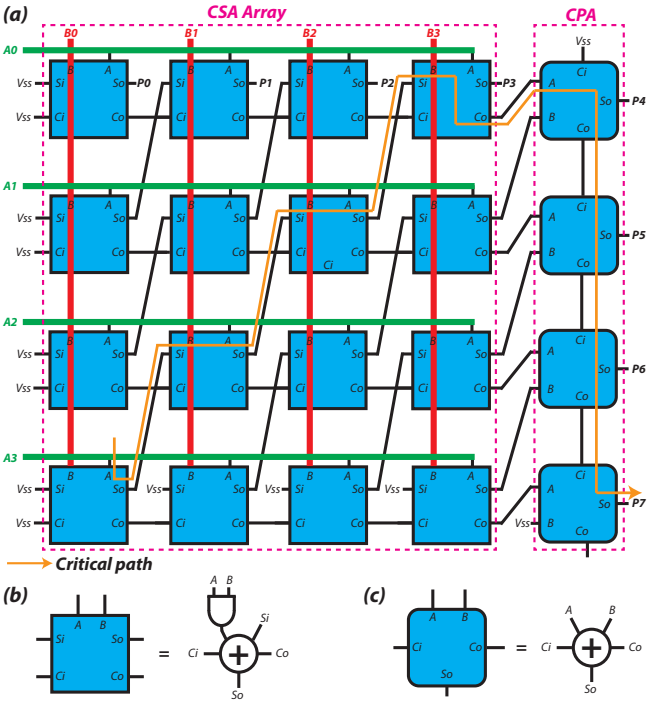

Fig. 2. (a) 3D 4-bit multiplier array organization. Note that each vertical layer on the figure is a physical vertical layer in the 3D plane. Array multiplier internal blocks: (b) CSA and (c) CPA. The rounded + gate represents a FA.

Fig. 2 (a) depicts the organization of a conventional 4-bit input multiplier array. Note that a 4-bit multiplier is shown for the readability of the figure, but the multiplier organi tion can be applied to larger sizes without loss of generali . The multiplier consists of a 4×4 *Carry-Save Adder* (CSA) array and a *Carry-Propagate Adder* ( ). Each CSA is built with an AND2 gate and a *Full Adder* (FA) gate, as shown in Fig. 2 (b). The NAND2 is used to calculate the partial product while the CSA is used to calculate the running sum, based on the partial products. The least significant N output bits are available as sum outputs directly from CSAs. The most significant output bits arrive in a carry-save redundant form and a 4-bit CPA is used to convert them into a standard binary form. The CPA is implemented using a traditional chain of FA gates, as shown in Fig. 2 (c). Note that the first CSA of each vertical layer adds the first partial product to a pair of 0, so its internal structure can be simplified to a single AND2 gate (producing S_o) while C_o can be set to V_{ss} for the next CSA.

B. 3D Multiplier Organization

As shown in Fig. 2 (a), each vertical layer (consisting of 4 CSAs and 1 FA) is identical, and hence can be rearranged as a vertical layer of the 3D Nanofabric. Hence, an N-bit input multiplier requires N vertical tiers. Inputs A_i and outputs P_i communicate with the Nanofabric through staircases. As all layers their share inputs B_i, they can be provided through vertical pillars rather than a staircase [12]. Such multiplier array is well suited for the Nanofabric as: (i) it comprises of a regular organization, where each row is the same, fitting within the 3D Nanofabric requirements; (ii) its internal interconnections are relatively simple, making it easy to comply with the single metal routing rule; (iii) inputs B_i can be shared among all vertical layers, simplifying the internal routing further. Note that within the CSA array, the S_i signals are propagated vertically to the next vertical layer. For the final CPA, the carries are also propagated vertically. As the FA is challenging to design following the 3D Nanofabric physical constraints, we used a gate-level based approach rather than a transistor-level based design, as proposed in [12]. As will be demonstrated in the experimental results section, this will lead to a small delay and power overhead at the gate-level. On the other hand, stacking several layers on top of each other will lead to a very low area and smaller interconnects, compensating the gate-based FA overheads.

IV. Experimental Results

In this section, we demonstrate the benefits of our proposed 3D multipliers. We first introduce our experimental methodology and then evaluate the performances of various multiplier sizes at the circuit-level. Finally, we provide a 3D thermal analysis.

A. Experimental Methodology

For the 2D area evaluations, we described the multipliers through RTL netlists, synthesized, and fed them into a Place & Route flow, using a commercial FDSOI 12nm node. We targeted an 12nm FDSOI node for the 2D area evaluation to compare the proposed 3D implementation to a competitive commercial technology node. For the 3D case area evaluation, we used an in-house PDK based on the 3D Nanofabric rules [12]. Note that the staircase area was also taken into consideration. For the circuit-level evaluations, we considered the same 28nm FDSOI node for both the 2D and 3D cases to show that the 3D multiplier delay and power reduction/overhead are agnostic of the technology node. Using a 12nm FinFET node for the 2D case would lead to better delay but a significantly higher power consumption compared to an FDSOI technology, which would taint the results and favor the 3D case. We chose an FDSOI technology as the layer transfer of crystalline silicon is compatible with the 3D Nanofabric [12]. Note that we consider the interconnect parasitics for both 2D and 3D cases, based on the post Place & Route flows. For the 3D vias, we estimate the resistance and capacitance to be $200m\Omega$ and $0.1fF$, respectively, based on similar monolithic 3D measurements on an FDSOI 28nm node [20].

B. Circuit-level Results

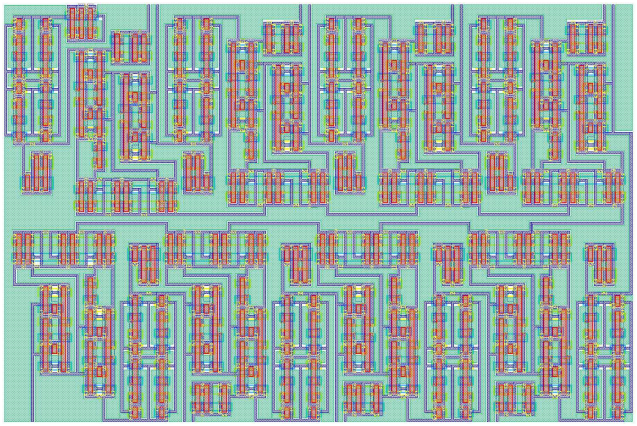

Fig. 3. 8-bit input multiplier layout view using the 3D Nanofabric rules [12].

TABLE I
2D FinFET and 3D Nanofabric multiplier area comparison, when varying the input-bit N from 4 to 64.

Number of input-bit N	Area (in μm^2)		
	2D	3D	Improvement
4	11.32	4.55	2.6×
8	57.14	8.92	6.6×
16	224.89	17.83	12.8×
32	724.89	36.32	20.1×
64	2662.62	75.78	35.1×

1) Area Benefits: Fig. 3 depicts the layout of one of the vertical layer of an 8-bit 3D multiplier, using the 3D Nanofabric flow. Note that in the 3D Nanofabric, there is no standard-cell fixed height due to the physical design constraints [12], hence the irregular logic cell placement. Nevertheless, as suggested in [12], we make sure that the gate to gate distance remains constant to obtain a regular layout in terms of the gate-pitch. As can be seen, several empty spaces are filled with some dielectric (in green), as some extra space is required to route the signals using a single metal layer, resulting in some area overhead. However, when stacking several layers on top of each other, significant area savings are obtained, as shown in Table I. As can be observed, even a small number of stacked layers ($N = 4$) leads to a area reduction of 2.6× (the difference between the number of vertical layers N and the area improvement is due to the area overhead coming from the staircase and the extra space required for the metal routing). As we stack up more vertical layers, this area is significantly decreased, up to 35.1× when $N = 64$. As explained in Section III-B, inputs Bi are shared vertically, so they do not need to be provided through a staircase, reducing its area overhead (for $N = 64$, the staircase area is $\sim 9\%$ of the total area). We believe that stacking 64 vertical layers is a fair assumption, as current V-Nand processes have demonstrated up to 128 vertical layers [21].

2) Delay and Power Comparisons: Fig. 4 shows the delay and power for both the 2D and 3D implementations when varying the input-bit N from 4 to 64. Note that the delay

and power values are normalized to the 2D delay and power, respectively, when $N = 4$. For both cases, the shown delays are the multiplier critical paths (represented by the orange arrow in Fig. 2 for the 3D case). As can be observed, the 3D Nanofabric exhibits a delay and power overhead when compared to the 2D case ($1.6\times$ and $3.6\times$ respectively, when $N = 4$). This can be explained by 2 factors: (i) as presented in Section III-B, gate-level based FAs are used, instead of transistor-level based cells, so their internal delay and power are higher; (ii) the 2D multipliers have been synthesized and hence optimized, while the 3D implementation uses a regular array due to the Nanofabric physical constraints. However, due to its very small area as several layers are stacked on top of each other, the 3D multiplier has significantly shorter internal wires, especially for large N, compensating the delay and power overheads previously described. This effect is more pronounced for large N as the 2D area (and hence the internal parasitics) grows quadratically while the 3D area grows linearly with N. When $N = 64$, the 3D delay, and power are only 10% and 35% higher than the 2D implementation, respectively.

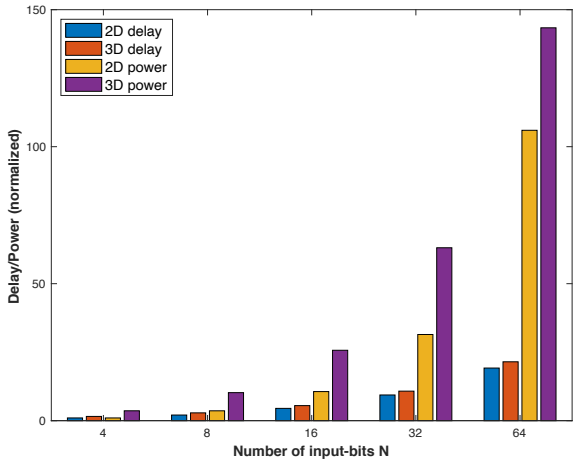

Fig. 4. Post PEX 2D and 3D multiplier delay and power comparisons, when varying the input-bit N from 4 to 64. Note that the delay and power values are normalized to the 2D delay and power, respectively, when $N = 4$.

3) Summary: Table II summarizes the various circuit-level results. Due to a slightly higher delay and power, the 3D multiplier energy is higher ($1.5\times$ for $N = 64$) than its 2D counterpart. However, the main benefit of the 3D multiplier is its very small area and thus lower manufacturing cost [12]. When stacking 64 layers, the area and *Area-Delay-Product* (ADP) can be decreased by 35.1 and $31.4\times$, respectively, compared to a 2D implementation.

C. Thermal Analysis

To study the thermal behavior of our proposed multiplier, we used 3D-ICE [22] based on the linear solver SuperLU library [23]. We assume that a microchannel-based liquid cooling is employed [24]. Fig. 5 shows the temperature map of the last layer of the proposed 64-bit 3D multiplier obtained with 3D-ICE. Most of the heat is concentrated where the

TABLE II
2D AND 3D MULTIPLIER METRIC COMPARISON, WHEN VARYING THE INPUT-BIT N FROM 4 TO 64.

Input-bit N		**4**	**8**	**16**	**32**	**64**
Delay (ns)	**2D**	0.17	0.34	0.75	1.56	3.20
	3D	0.26	0.47	0.92	1.80	3.58
Power (μW)	**2D**	28.2	87.1	256.7	820.3	2561.6
	3D	87.3	247.4	621.1	1525.4	3465.5
Energy (fJ)	**2D**	4.7	29.9	192.0	1281.6	8195.5
	3D	22.6	117.4	568.7	2738.5	12402.8
ADP ($\mu m^2.ns$)	**2D**	1.89	19.61	168.20	1132.57	8518.65
	3D	1.12	4.14	16.08	64.75	271.20

power dissipation is the highest (latest CSA and CPA of the multiplier).However, there is no particular critical hotspot ($T < 57°C$), showing that our proposed 3D multipliers are thermally reliable. Besides, while each layer of the 3D multiplier consumes a bit more power than a 2D implementation, the internal gates are bigger (gate-level based FA), so the current density is, in fact, lower. Note that a similar analysis can be done for smaller N, but the internal temperature would be lower since fewer vertical layers are stacked.

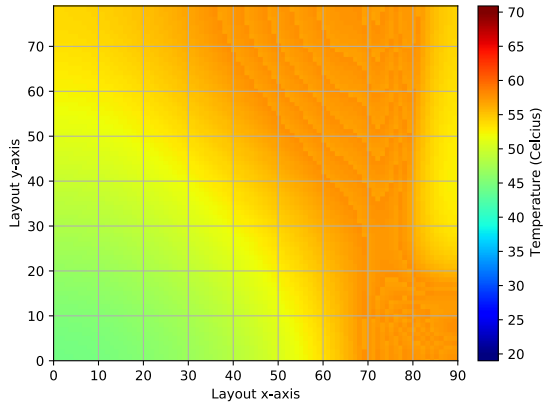

Fig. 5. Temperature map of the top layer of the proposed 64-bit 3D multiplier, in the xy plane.

V. Conclusion

In this paper, we introduced area-efficient 3D multiplier designs based on the recently proposed 3D Nanofabric. As multipliers can be realized using regular arrays, we first showed how they could be spread across multiple vertical layers, while respecting the different layout constraints of the 3D Nanofabric. Through electrical simulations considering parasitics, we showed that by stacking up to 64 layers to build a 64-input bit multiplier, the area is decreased by $35.1\times$ compared to a conventional 12nm FinFET implementation, with only a 35% power consumption overhead. Besides, we showed through 3D thermal simulations that our proposed multiplier is reliable, even for 64 layers. We believe that such kind of low-area multiplier designs can bring significant benefits in the context of machine learning accelerators for CNNs. In future works, we plan on evaluating multiplier arrays at the architectural-level. While the delay and power consumption of the 3D multiplier are slightly higher than its 2D counterpart, proposing an array of such 3D multipliers

(with some shared inputs) could lead to very high-bandwidth and low-area structures, particularly attractive in the context of machine learning to realize convolutions. We also plan on providing an accurate cost-analysis, as the low area and few processing steps of such 3D Nanofabric can significantly reduce the cost of manufacturing [12].

REFERENCES

[1] A. Razavieh *et al. Challenges and Limitations of CMOS Scaling forFin-FET and Beyond Architectures*, IEEE TNANO, 18:999-1004, 2019.

[2] K. Seshan *et al. Handbook of Thin Film Deposition (Fourth Edition), Chapter 2 - Limits and Hurdles to Continued CMOS Scaling*, William Andrew Publishing, 2018.

[3] J. Kim *et al.*, *HBM: Memory solution for bandwidth-hungry processors,* HCS, 2014.

[4] C. Lee *et al.*, *Overview of the Development of a GPU with Integrated HBM on Silicon Interposer*, ECTC, 2016.

[5] S. Lhostis *et al.*, *Reliable 300 mm Wafer Level Hybrid Bonding for 3D Stacked CMOS Image Sensors*, ECTC, 2016.

[6] A. Mallik *et al.*, *The impact of Sequential-3D integration on semiconductor scaling roadmap*, IEDM, 2017.

[7] P. Batude *et al.*, *Advances, Challenges and Opportunities in 3D CMOS Sequential Integration*, IEDM, 2011.

[8] L. Brunet *et al.*, *First demonstration of a CMOS over CMOS 3D VLSI CoolCube™ integration on 300mm wafers*, VLSI, 2016.

[9] F. Andrieu *et al.*, *A review on opportunities brought by 3D-monolithic integration for CMOS device and digital circuit*, ICICDT, 2018.

[10] M. Shulaker *et al.*, *Three-dimensional integration of nanotechnologies for computing and data storage on a single chip*, Nature, 547: 74–78, 2017.

[11] Z. Wan *et al.*, *Low-Temperature Wafer Bonding for Three-Dimensional Wafer-Scale Integration*, S3S, 2018.

[12] E. Giacomin *et al. Layout Considerations of Basic Arithmetic Logic Units Using an N-layer 3D Nanofabric Process Flow*, VLSI-SOC, 2020.

[13] J. Jang et al., *Vertical cell array using TCAT(Terabit Cell Array Transistor) technology for ultra high density NAND flash memory*, VLSI, 2009.

[14] A. Khan *et al. A survey of the recent architectures of deep convolutional neural networks*, Artif Intell Rev (2020). https://doi.org/10.1007/s10462-020-09825-6

[15] L. P. Thakre *et al. Performance Evaluation and Synthesis of Multiplier Used in FFT Operation Using Conventional and Vedic Algorithms*, ICETET, 2010.

[16] Y.-H. Chen *et al. Eyeriss: A Spatial Architecture for Energy-Efficient Dataflow for Convolutional Neural Networks*, ISCA, 2016.

[17] Y. S. Shao *et al. Simba: Scaling Deep-Learning Inference with Multi-Chip-Module-Based Architecture*, MICRO, 2019.

[18] E. Beyne *et al.*, *Through-silicon via and die stacking technologies for microsystems-integration*, IEDM, 2008.

[19] W. Ruythooren *et al.*, *Cu-Cu Bonding Alternative to Solder based Micro-Bumping,* EPTC, 2007.

[20] D. Kumar Nayak *et al. Power, performance, and cost comparisons of monolithic 3D ICs and TSV-based 3D ICs*, S3S, 2015.

[21] C. Siau *et al.*, *A 512Gb 3-bit/Cell 3D Flash Memory on 128-Wordline-Layer with 132MB/s Write Performance Featuring Circuit-Under-Array Technology*, ISSCC, 2019.

[22] A. Sridhar *et al.*, *3D-ICE: Fast compact transient thermal modeling for 3D-ICs with inter-tier liquid cooling,* ICCAD, 2010.

[23] X. S. Li, *An overview of SuperLU: Algorithms, implementation, and user interface,* ACM Transactions on Mathematical Software, 2015.

[24] T. Brunschwiler *et al.*, *Interlayer cooling potential in vertically integrated package,* Microsyst. Technol., 15: 57–74, 2009.