Exploring a Design Space of 3-D Stacked Vector Processors

Ryusuke Egawa, Jubee Tada, and Hiroaki Kobayashi

Abstract Three dimensional (3-D) technologies have come under the spotlight to overcome limitations of conventional two dimensional (2-D) microprocessor implementations. However, the effect of 3-D integrations with vertical interconnects in future vector processors design is not well discussed yet. In this paper, aiming at exploring the design space of future vector processors, fine and coarse grain 3-D integrations that aggressively employ vertical interconnects are designed and evaluated.

1 Introduction

Modern vector processors play important roles in high performance computing due to the significant advantages over commodity-based scalar processors for memory-intensive scientific and engineering applications. However, vector processors still keep a single core architecture, though chip multiprocessors (CMPs) have become the mainstream in recent processor architectures. Twelve-cores CMPs are already in the commercial market, and an 80-cores CMP is prototyped by Intel to overcome power and performance limitations of single core architectures [30]. On the other hand, CMP-based vector processors have not been found as real products. However, the CMP architecture is also promising for vector processor design, because recent scientific and engineering applications running on a vector supercomputer are well

R. Egawa (✉) · H. Kobayashi
Cyberscience Center, Tohoku University/JST CREST, 6-3 Aramaki-aza-aoba, Aoba, Sendai 980-8578, Japan
e-mail: egawa@isc.tohoku.ac.jp; koba@isc.tohoku.ac.jp

J. Tada
Graduate School of Science and Engineering, Yamagata University, 4-3-16 Jonan, Yonezawa 992-8510, Japan
e-mail: jubee@yz.yamagata-u.ac.jp

M. Resch et al. (eds.), *Sustained Simulation Performance 2012*,
DOI 10.1007/978-3-642-32454-3_4, © Springer-Verlag Berlin Heidelberg 2013

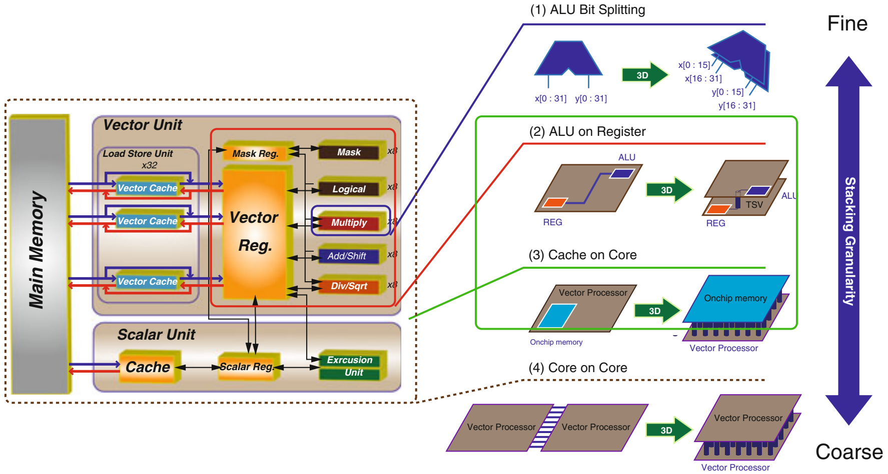

Fig. 1 Stacking granularities

parallelized by vector compilers and/or OpenMP. Under this situation, a Chip Multi Vector Processor (CMVP) architecture has been proposed by Musa et al., and its potential has been clarified [22].

Essentially, vector processors need a plenty of hardware resources compared to scalar processors. This is because a vector processor installs many vector pipelines, large vector registers and vast I/O logics to simultaneously process a huge amount of data provided at a high-memory bandwidth [9]. In addition, CMVP has a large shared on-chip memory named a *vector cache*. Therefore, even if technology scaling were to advance as ever, it would be difficult to implement CMVP with many vector cores by the conventional 2-D implementation technology due to the area limitation.

Recently, 3-D integration technologies have come under the spotlight to overcome the limitations of conventional 2-D microprocessor implementations. 3-D integration technologies are expected to greatly increase transistor density while providing faster on-chip communication. Three dimension integration technologies are not a brand new, and various 3-D integration technologies have been explored, such as micro-bump, wire bonding and through via vertical interconnect etc [16]. Among these various technologies, a vertical interconnect with though-silicon-via (TSV) is assumed as the most promising one to expand the design space of future high-performance and low-power microprocessors [13]. Thus computer architects and circuit designers are re-attracted to 3-D integration technologies by an appearance of TSVs with high feasibility.

To introduce 3-D Die stacking technologies into the future vector processors design, several granularities can be considered as shown in Fig. 1. Aiming at clarifying the potential of 3-D integration technologies, this paper examines fine grain and coarse grain 3-D designs. The fine grain denotes logic level 3-D integrations such as 3-D stacked arithmetic units designs. On the other hand, the coarse grain

stacks more large chunks such as cores or memories. As examples of the fine grain 3-D integrations, we focus on the floating point arithmetic units, which are the key components of vector processors. On the other hand, CMVP architecture is selected as a target of the coarse grain 3-D integrations. To realize the 3-D integration of CMVP, CMVP is modified to exploit the potential of 3-D integration technologies. In this paper, through these designs and early evaluations, a design space of future vector processors is explored.

This paper is organized as follows. Section 2 briefly outlines the basis of 3-D integration technologies and related works. Then 3-D integrated arithmetic units are designed and evaluated in Sect. 3. In Sect. 4, a 3-D stacked CMVP is introduced, and its early performance evaluations are carried out. Section 5 concludes this work.

2 3-D Die Stacking Technologies

2.1 Die Stacking with TSVs

3-D integration is emerging fabrication technology that vertically stacks multiple integrated chips. The benefits include an increase in device density, the flexibility of routing, and the ability to integrate disparate technologies [4, 16]. Although there are a lot of technologies such as wire bonding to realize a vertical stacking of two or more integrated circuits, this study focuses on 3-D integrated circuits with vertical interconnects due to its short delay with high density. So far, many researches have reported processing technologies for thin and long TSVs with high feasibility [1, 3, 8, 12, 18]. In addition, some researchers have clarified the potential of TSVs by analyzing its electrical characteristics [11, 23].

Two topologies can be conceived to bound two silicon dies: *face-to-face* and *face-to-back*, where face is the side with the metal layer and back is the side with the silicon substrate as shown in Fig. 2. In the face to face bonding, die to die (D2D) vias are processed and deposited on top of metal layers as the conventional metal etching technologies. Although face to face bonding can provide higher D2D via density and lower area overhead than face to back, it can just allow to stack two active silicon layers. On the other hand, face to back bonding can stack any number of multiple active silicon layers by TSVs that go through silicon bulk with lower via density. As noted in earlier studies, the dimension for TSVs vary from 2 to 5 μm in recent real implementations and pitch of 3-D via only requires a modest overhead. More details description about processing techniques of TSVs are described in [12, 18]. Thus, to realize aggressive stacking of multiple layers, face to back seems preferable. In this paper, the following circuits and processors are designed using two to eight silicon layers, henceforth, we focus on the face to back wafer bonding technique.

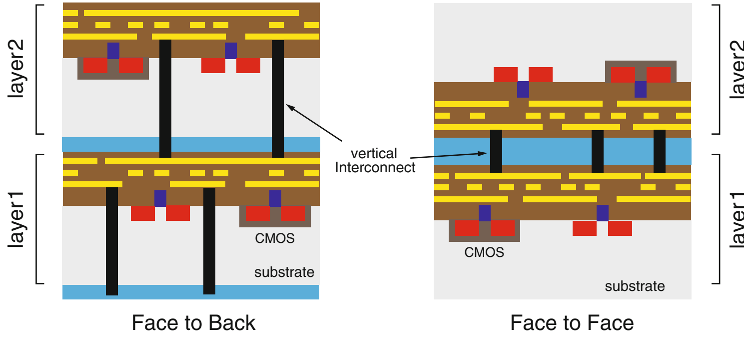

Fig. 2 Face to back vs. Face to face

2.2 Related Work

According to the trend of recent microprocessor designs, several researches have combined 3-D memory stacking and CMPs to supply data to a chip with a massive number of cores at an enough bandwidth [10, 15]. Black et al. [2] have explored the memory bandwidth benefits using Intel Core2 Duo processor. By stacking an additional cache memory layer, the on-chip cache capacity is increased, and the performance is improved by reducing off-chip memory accesses. Loh et al. [17] have discussed DRAM stacking on CMPs. This paper has tried to fully take advantages of the benefit of 3-D integration technologies with TSVs, memory organizations are optimized for many purposes such as the main memory or the last level caches. In addition, stacking layers of non-volatile cache memory such a Phase Change Random Access Memory (PRAM) [31] and a Magnetic Random Access Memory (MRAM) [5] are also studied to mitigate the effects of the processor-memory speed gap with low power consumption. However, these researches are carried out based on multi-core or many-core scalar microprocessors. In this paper, we focus on the chip multi-vector processor, which cannot be implemented by conventional 2-D technologies with a high memory bandwidth. To realize a high sustained performance by improving the memory bandwidth, the 3-D integration technology is suitable for the vector architecture. Thus this paper targets on designing a 3-D stacked CMVP as an example of a coarse grain die stacking implementation.

On the other hand, there are few studies that try to clarify the effects of fine grain 3-D integrations. Mayage et al. [19] have designed a 3-D carry look ahead adder with face to back implementation. Puttaswamy and Loh also have explored the effects of employing the vertical interconnect in combinational logic design [24, 25]. They have designed and evaluated three kinds of parallel prefix adder and barrel shifter by the face to face implementation. In [29], they have designed and evaluated the performance of several arithmetic and control units based on face to back implementation, and reported that 3-D integrated circuits can improve

their power efficiency and performance in the future CMOS and 3-D integration technologies. Due to the small number of researches in this field, these explorations are limited to small-scale arithmetic units, though 3-D integration seems effective in a large-scale circuit design. Hence, this paper selects floating point arithmetic units, which are the most largest and important combinational logic circuits in the recent high-performance microprocessors.

3 3-D Stacked Arithmetic Units

In recent microprocessors, floating-point arithmetic units play important role to achieve a high computational performance. Especially, vector processors and GPUs such as processors with SIMD/Vector instructions install a lot of floating-point arithmetic units to achieve a high computational performance. Thus, in this section, floating point arithmetic units are design in a 3-D fashion and evaluated. Through these approaches, the effectiveness and potential of 3-D integration technologies in floating point unit design are clarified. To design 3-D stacked arithmetic units, first, an arithmetic unit is partitioned into some sub-circuits, based on a circuit partitioning strategy. Next, each sub-circuit is placed on one layer, and data-transfers between sub-circuits are done through TSVs. TSVs require Keep-out zone to avoid electrical effect caused by TSVs to transistors.

Therefore, to relax this effect, it is assumed that the area of TSVs and the area of sub-circuits are separated, and TSVs are placed around the sub-circuits as shown in Fig. 3. More detailed design flow can be confirmed in [6].

Based on the discussions in our previous work, there are two requirements for a partitioning strategy for 3-D integration. First, sub-circuits should have almost same size for minimize the footprint, and the critical-path delay should not be enlarged. To fulfill these requirements, we use a circuit partitioning strategy proposed in [26]. The concept of this strategy is shown in Fig. 4, and the features of the strategy are as follows:

- Gates on the critical-path are packed in single layer.
- The area of each layer is equalized as much as possible.
- Small components should be packed together on one layer.

If the critical-path in a circuit is divided into some sub-circuits, TSVs are inserted on the critical-path, and it will increase the maximum delay of a circuit due to its large capacitance. However, it is difficult to partition small components into more small sub-circuits without dividing the critical-path. Therefore, large components should be partitioned into sub-circuits without dividing its critical-path, and small components should be packed together into one layer. To keep the footprint of 3-D stacked VLSIs small, the size of each layer should be equalized. If one sub-circuit is larger than other sub-circuits, it enlarges the footprint of 3-D stacked VLSI. This equalization is controlled and achieved by partitioning large components.

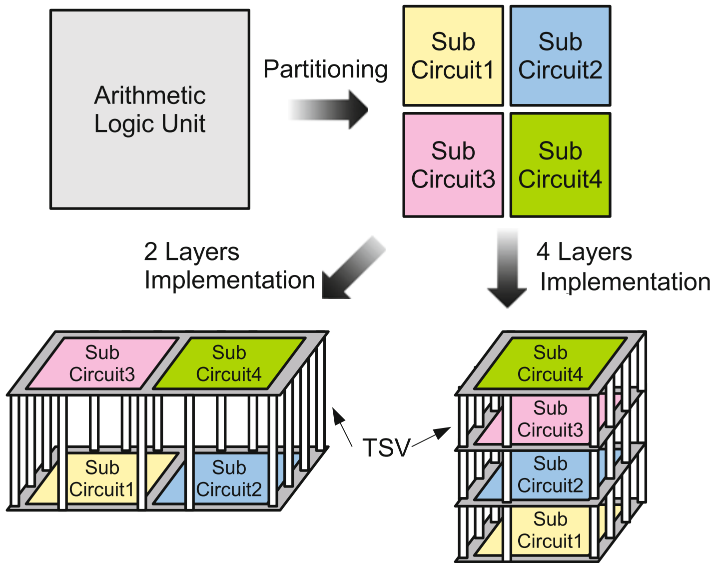

Fig. 3 Design flow of 3-D stacked arithmetic units

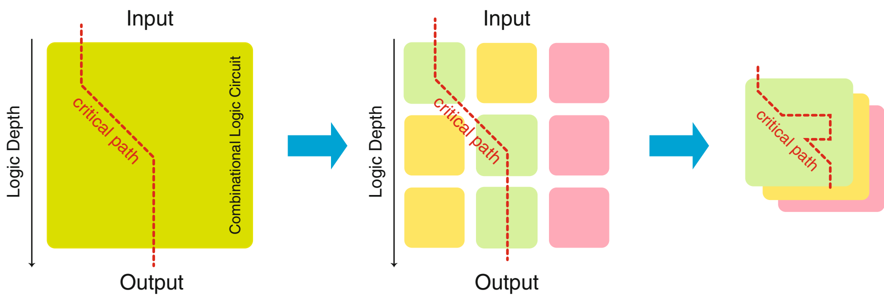

Fig. 4 Circuit partitioning for 3D die stacking

Figure 5 shows a circuit partitioning for a floating-point multiplier. The layer 1 includes the lower-bits part of a booth-encoder and a wallace-tree, and other components of the floating-point multiplier. Other layers have the rest parts of the booth-encoder and the wallace-tree. This is because a significant multiplier, which consists of the booth-encoder, the wallace-tree and the final adder occupies large part of the circuit area. Thus, the booth-encoder and the wallace-tree are partitioned into some sub-circuits. Since the final adder is not partitioned due to avoid dividing the critical-path, the adder is implemented into a single layer.

To evaluate the effects of 3-D Die stacking technologies in arithmetic units designs, single and double precision 3-D stacked floating multipliers are designed using the 180 nm CMOS technology with TSVs. The parameters of TSVs are

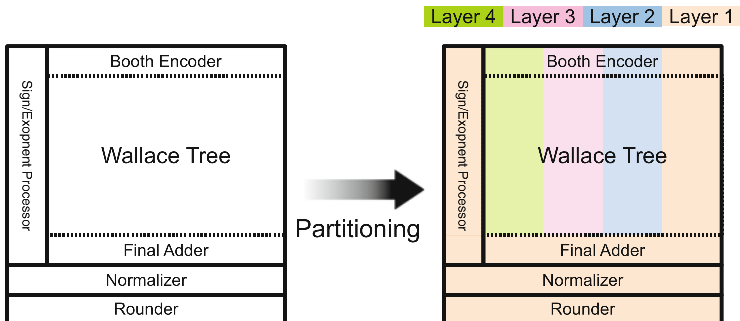

Fig. 5 Design of 3-D stacked floating multiplier (4 layers)

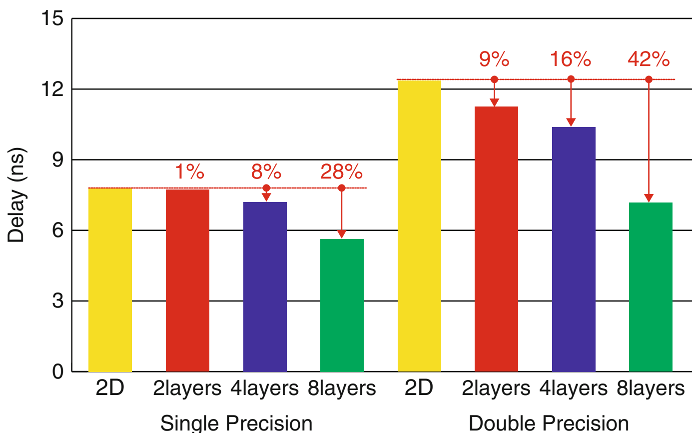

Fig. 6 Effects of 3-D die stacking

assumed based on the ITRS reports, and the resistance and capacitance of the TSVs are set to 7 mΩ and 27 fF, respectively. The number of layers is varied from two to eight. Figure 6 shows the maximum delay of 3-D stacked floating point multipliers. Compared to 2-D implementations, 3-D implementations achieve significant delay reductions. Both in the cases of single and double precision designs, reductions in the maximum delay are boosted as the number of layers increases. This is mainly because the size of layers becomes small as the number of layers increases. This also eliminates the extremely long wires and shortens the average wire length in each design, hence 3-D integrations achieve up to 42 % delay reduction. From these results, we can confirm that 3-D integration technologies with TSVs have enough potential to reduce the maximum delay of floating point arithmetic units by selecting appropriate circuit partitioning strategies.

In addition the maximum delay reduction in the double precision multipliers are larger than those of single-precision multipliers. In the double-precision floating

point multipliers, long wires occupy the many parts of total wire length compared to the single-precision multiplier. Our 3-D implementation reduces the number of these long wires, and it contributes to a reduction in the maximum path delay. Therefore, this result also indicates that 3-D integration technologies are more effective in large-scale arithmetic units designs. More details of the fine grain 3-D stacked circuit designs are described in [26].

4 3-D Stacked Chip Multi Vector Processors

4.1 An Overview of 3-D Stacked CMVP

Recently, it is getting harder to further improve the performance and energy efficiency of vector processors due to several limitations such as the die area and the number of I/O pins on a chip. The most severe problem is the decrease in the ratio of memory bandwidth to the floating-point operation rate (Bytes/Flop, B/F). A high B/F ratio is essential to achieve a high computational efficiency, i.e., efficient use of the computing power. If the B/F ratio of a vector processor decreases to be as the same level as that of a scalar one, the vector processor will no longer be able to keep its superiority over the scalar processor in terms of the sustained performance. To compensate for a low B/F ratio, an on-chip memory for a vector processor named a vector cache has been proposed [21]. The on-chip memory can provide data to vector registers of the processor at a high bandwidth because the data transfer does not need I/O pins. Hence, the B/F ratio of a vector processor is improved by storing reusable data in the on-chip memory to achieve high sustained performance. In addition, an on-chip memory is expected to decrease the number of off-chip main memory accesses, resulting in decreasing the energy consumption in the processor I/O, memory network and off chip memory components. Therefore, the vector cache is also introduced into the 3-D stacked CMVP. The vector cache is not private to each processor core but shared by multiple cores because scientific simulations such as difference schemes often have a high locality of memory reference among threads.

Figure 7 shows the basic structure of the 3-D stacked CMVP. The 3-D stacked CMVP is composed of three kinds of layers; I/O layer, core layer, vector cache layer. The I/O layer contributes to keep off-chip memory bandwidth, and the core layer realizes implementation of many cores on a die. The vector cache layer works for increasing the capacity of on-chip memory to compensate for insufficient memory bandwidth of each core.

The core layer includes two vector cores, and each vector core is designed based on the NEC SX-8 processor. The vector core has four parallel vector pipe sets, each of which contains six types of vector arithmetic pipes (Mask, Logical, Add/Shift, Multiply, Divide, Square root), and 144 KB vector registers as shown in Fig. 8.

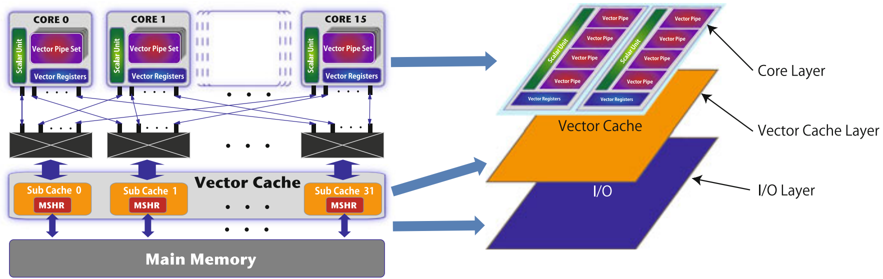

Fig. 7 Basic structure of the 3-D stacked CMVP

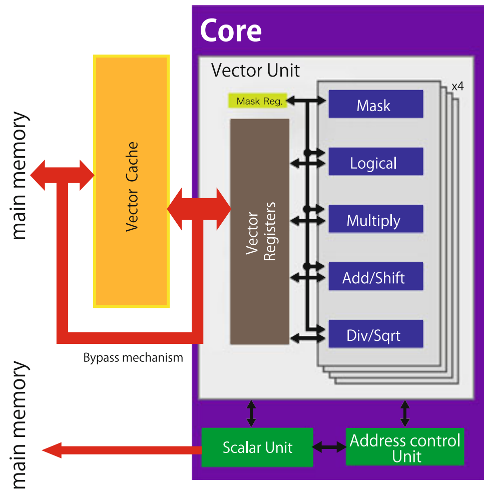

Fig. 8 Block diagram of a core

On the I/O layer, only SERializer/DESerializer(Ser/Des logic) is implemented. As shown in [9], the occupancies of Ser/Des logic in the NEC vector processors SX-8 and SX-9 are quite high. Thus the 3-D stacked CMVP introduces independent I/O layers to keep a high memory bandwidth.

The vector cache layers are put between the I/O layers and the core layers. The vector cache layers consist of 32 2-way set-associative sub-caches with miss status handling registers (MSHR) [14]. A vector load/store instruction of the vector architecture is able to concurrently deal with up to 256 floating-point data. Hence, the vector cache also needs to process up to 256 data in continuity. Furthermore, the vector cache employs a bypass mechanism between the main memory and vector register files. The bypass mechanism makes possible to supply data from both the main memory and the vector cache at the same time. Thus, the total amount of data provided to the vector register files in time is increased by the bypass mechanism. In addition, the vector cache is a non-blocking cache with MSHR. In scientific

computations such as difference schemes, two vector load instructions load the memory regions that are partially overlapped. If the subsequent load instruction is issued right after the preceding instruction, however, the data to be fetched by the preceding instruction have not been cached in the vector cache yet owing to the long latency of main memory accesses. Thus, the subsequent load instruction causes cache misses even though the data to be accessed are in-flight. MSHR is used to avoid this situation, and makes it possible for the subsequent load instruction to reuse in-flight data fetched by the preceding instruction.

4.2 Performance Evaluations

In this section, to clarify the effects of increasing the off-chips memory bandwidth and the number of cores by 3-D integration technologies, we firstly evaluate the performance of the 3-D stacked CMVP without vector cache layers. Then the performance with vector cache layers is examined in terms of sustained performance and energy consumption. Based on the performance evaluation, the tradeoff between performance and energy consumption is discussed to realize effective usage of a plenty of hardware given by 3-D integration technologies.

4.2.1 Evaluation Setup

An NEC SX trace-driven simulator that can simulate the behavior of the 3-D stacked CMVP architecture at the register-transfer level is implemented. The simulator is designed based on the NEC SX vector architecture. The simulator accurately models a vector core of the SX-8 architecture; the vector unit, the scalar unit and the memory system. The simulator takes system parameters and a trace file of benchmark programs as input, and outputs instruction cycle counts. The specification of 3-D stacked CMVP is shown in Table 1. We assume that the I/O layer provides a 64 GB/s memory bandwidth with one layer, the number of cores is possible at the maximum of 16, and the maximum capacity of the vector cache is 32 MB. Since we assume that the TSV has 2 um diameter with a 30 um length [13], the access latency between cores and the vector cache reduces to 70 % of the cache access latency of conventional 2-D implementations. The energy consumed by the vector cache accesses are obtained by CACTI6.5 [20].

The evaluations are performed by using a FDTD code, which simulates the antipodal fermi antenna [27] by using SX-9 of Tohoku University. The number of grids, the vector operation ratio and vector length are $612 \times 105 \times 505$, 99.9% and 255, respectively. The benchmark programs is compiled by the NEC FORTRAN compiler, which can vectorize and parallelize the applications automatically. Then executable programs run on the SX trace generator to produce the trace files.

Table 1 Specification of 3-D stacked CMVP

Parameter	Value
Number of cores	1–16
Vector cache implementation	SRAM
Capacity of the vector cache	512–32 MB
Cache line size	8B
Cache policy	Write-through, LRU replacement
Cache associativity	2
Memory bandwidth (between cache and core)	64 GB/s/core
Off-chip Memory bandwidth	64–256 GB/s
Tr. process technology	90 nm
Number of entries of MSHR	8,192

4.2.2 Evaluation Results and Discussions

First, the performance of the 3-D stacked CMVP with the various number of cores is evaluated. As shown in the previous section, since one core layer includes two vector cores, 4-cores, 8-cores and 16-cores are implemented by two layers, 4-layers and 8-layers, respectively. The off-chip memory bandwidth is also varied from 64 to 256 GB/s by stacking the necessary number of I/O layers. In the case of using a single I/O layer, the off-chip memory bandwidths are 64 GB/s and 128 GB/s by changing the usage of silicon budgets. 64 GB/s is achieved when a half part of the I/O layer is used, and 128 GB/s is achieved by using the whole I/O layer. Doubling I/O layers realizes 256 GB/s. The memory bandwidth per core is decreased as the number of cores increases, thus 4 B/F rate per core is achieved in the cases of 2 cores with 128 GB/s of the off-chip memory bandwidth and 4 cores with 256 GB/s of the off-chip memory bandwidth.

Figure 9a shows the performance of the 3-D stacked CMVP when changing the off-chip memory bandwidth. The performance is normalized by the case of single core, without vector cache, with baseline off-chip memory bandwidth (64 GB/s). In this graph, yellow, blue and red bars indicate the cases of 64, 128, and 256 GB/s, respectively. From these results, we can confirm that enhancing off-chip memory bandwidth improves the sustained performance with a high-scalability. On the other hand, Fig. 9b shows the performance of the 3-D stacked CMVP with a 8 MB vector cache when changing the off-chip memory bandwidth. We can also confirm that the vector cache has a high potential to improve the performance in the all the cases. These results indicate that there are two choices to improve the performance of the 3-D stacked CMVP by enhancing off-chip.

Next, effects of enhancing off-chip memory bandwidth by introducing I/O layers and employing the vector cache are discussed. Figure 10a shows the normalized performances of 16 cores cases. The performance of using two I/O layers without the vector cache (1 B/F) and using one I/O layer with an 8 MB vector cache (0.5 B/F + Cache) are comparable. Both cases improve the effective memory

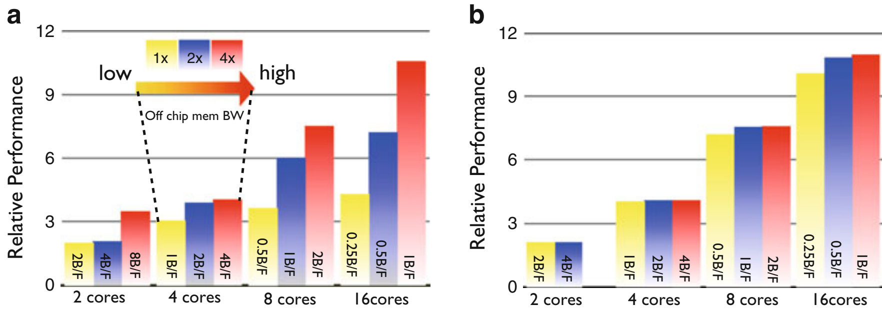

Fig. 9 Effects of enhancing off-chip memory bandwidth and vector cache. **(a)** Effects of off-chips mem. bandwidth. **(b)** Effects of 8 MB vector cache

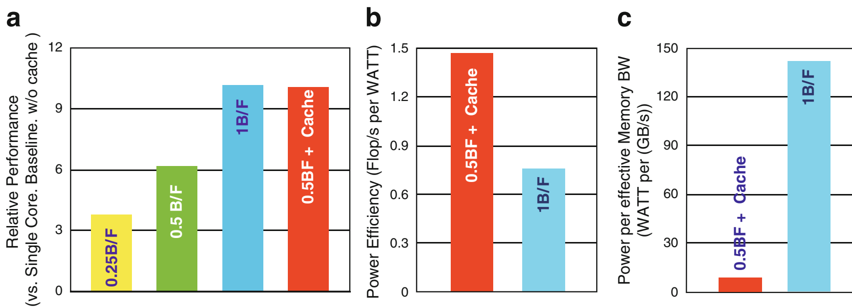

Fig. 10 Off-chip vs. On-chip memory bandwidth. **(a)** Performance. **(b)** Sustained Performance per Watt. **(c)** Watt per Effective Memory BW

bandwidth, and then improve the performance compared to the base-lined case. This result indicates that several architectural designs of the 3-D stacked CMVP can be considered. To clarify the energy efficient configurations of the 3-D stacked CMVP, the power and energy consumption of the 3-D stacked CMVP are evaluated. Figure 10b shows the power efficiency of the 3-D stacked CMVP. The power efficiency is obtained by *performance/Watt*. This result indicates that the vector cache can achieve 49 % higher power efficiency (*performance/watt*) compared to the case of increasing the off-chip memory bandwidth. In addition, the vector cache requires a 91 % smaller power consumption per effective memory bandwidth compared to the case of increasing the off-chip memory bandwidth as shown in Fig. 10c. Therefore, to realize energy efficient computing on the 3-D stacked CMVP, the vector cache is much more power-efficient. Since the power efficiency would strongly depend on applications, more variable evaluations using other benchmarks are needed to determine the best configuration in terms of the performance and the power consumption. This consideration remains as our future work.

In this paper, we set a bandwidth between the vector cache and the vector register as 64 GB/s/core. However, there are several researches designing 3-D stacked cache memories, which realize a high memory bandwidth and huge capacity using

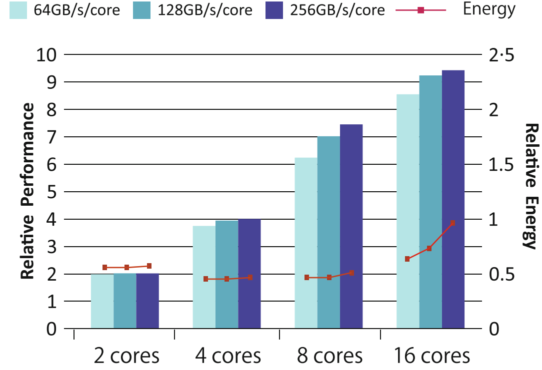

Fig. 11 Effect of an enhanced the vector cache on the performance and energy

through silicon vias (TSVs) [15, 28]. Since TSVs have small RC delay compared to conventional 2-D wire and recent TSVs process technologies allow to implement many TSVs on a chip [8], 3-D stacked cache memories have potential to realize higher memory bandwidth compared to 2-D implementation [7]. Thus, we can assume the future 3-D stacked vector cache have a high memory bandwidth of 128 GB/s/core and 256 GB/s/core.

Figure 11 shows the performance and energy consumption of the 3-D stacked CMVP with a 8 MB enhanced vector cache. In this evaluation, the FDTD code is also used as a benchmark, and every value are normalized by that of a single core without the vector cache. From these results, we can confirm that enhancing the performance of the vector cache is quite effective to improve energy efficiencies of the 3-D stacked CMVP. More detail design of the 3-D vector cache, which exploits the potential of 3-D die stacking technologies should be considered as our future work.

5 Conclusions

To clarify the potential of 3-D Die stacking in the future vector processors design, fine and coarse grain 3-D integrations are examined. As a fine grain 3-D integration, 3-D stacked floating point multipliers are designed and evaluated. By partitioning the floating point multipliers appropriately, 3-D integration technologies can significantly reduce the maximum delay of the floating point multipliers. On the other hand, as a coarse grain 3-D integration, the 3-D stacked CMVP is introduced and evaluated. The effects of the vector cache and enhancing off-chip memory bandwidth by I/O layers are evaluated from the viewpoint of the performance and the energy consumption. Evaluation results show that the vector cache can effectively decrease the power and energy consumption of the 3-D stacked CMVP, while achieve a high performance. From these results, we can confirm that 3-D integration technologies have enough potential to boost the performance and the energy efficiency of future vector processors.

To realize more powerful computing environments with extremely high memory bandwidth, more detailed design of the vector cache should be considered. In addition, designing of the 3-D stacked CMVP the vector cache under the fine and coarse grain 3-D integration technologies should be considered.

Acknowledgements The authors would like to thank Associate Professor Hiroyuki Takizawa, Professor Mitsumasa Koyanagi of Tohoku University, Yusuke Funaya of Hitachi, Ryu-ichi Nagaoka of BOSCH, Dr. Akihiro Musa, Jun Inasaka and Dr. Shintaro Momose of NEC for valuable discussions on this research. This research was partially supported by Grant-in-Aid for Scientific Research (Grant-in-Aid for Young Scientists (B) No. 22 700044) and (Grant-in-Aid for Scientific Research (B) No. 22300013), the Ministry of Education, Culture, Sports, Science and Technology. This research was also partially supported by Core Research for Evolutional Science and Technology (CREST), Japan Science and Technology Agency (JST).

References

1. E. Beyne. Tsv technology overview. In *Semicon Taiwan 2008 CTO Forum*, 2008.
2. B. Black, M. Annavaram, N. Brekelbaum, J. DeVale, L. Jiang, G. H. Loh, D. McCaule, P. Morrow, D. W. Nelson, D. Pantuso, P. Reed, J. Rupley, S. Shankar, J. Shen, and C. Webb. Die stacking (3d) microarchitecture. In *MICRO 39: Proceedings of the 39th Annual IEEE/ACM International Symposium on Microarchitecture*, pages 469–479, 2006.
3. S. Das, A. Fan, K.-N. Chen, C. S. Tan, N. Checka, and R. Reif. Technology, performance, and computer-aided design of three-dimensional integrated circuits. In *ISPD '04: Proceedings of the 2004 international symposium on Physical design*, pages 108–115, New York, NY, USA, 2004. ACM.
4. W. Davis, J. Wilson, S. Mick, J. Xu, H. Hua, C. Mineo, A. Sule, M. Steer, and P. Franzon. Demystifying 3d ics: the pros and cons of going vertical. *Design & Test of Computers, IEEE*, 22(6):498–510, Nov.-Dec. 2005.
5. X. Dong, X. Wu, G. Sun, Y. Xie, H. Li, and Y. Chen. Circuit and microarchitecture evaluation of 3D stacking magnetic RAM (MRAM) as a universal memory replacement. In *DAC '08: Proceedings of the 45th annual Design Automation Conference*, pages 554–559, New York, NY, USA, 2008. ACM.
6. R. Egawa, Y. Funaya, R. Nagaoka, Y. Endo, A. Musa, H. Takizawat, and H. Kobayashi. Effects of 3-D Stacked Vector Cache on Energy Consumption. In *2011 IEEE International 3D Systems Integration Conference (3DIC)*, pages 1–8, 2012.
7. Y. Funaya, R. Egawa, H. Takizawat, and H. Kobayashi. 3D On-Chip Memory for the Vector Architecture. In *2009 IEEE International 3D Systems Integration Conference (3DIC)*, pages 1–8, 2009.
8. S. Gupta, M. Hilbert, , S. Hong, and R. Patti. Techniques for producing 3d ics with high-density interconnect. In *Proceedings of the 21st International VLSI Multilevel Interconnection Conference*, 2004.
9. J. Inasaka and M. Kajita. Techniques for power supply noise management in the SX supercomputers. In *IEICE Tech. Report*, pages 41–46, 2008.
10. T. Kgil, A. Saidi, N. Binkert, S. Reinhardt, K. Flautner, and T. Mudge. Picoserver: Using 3d stacking technology to build energy efficient servers. *J. Emerg. Technol. Comput. Syst.*, 4(4):1–34, 2008.
11. D. Khalil, Y. Ismail, M. Khellah, T. Karnik, and V. De. Analytical model for the propagation delay of through silicon vias. In *ISQED '08: Proceedings of the 9th international symposium on Quality Electronic Design*, pages 553–556, 2008.

12. M. Koyanagi, T. Fukushima, and T. Tanaka. High-density through silicon vias for 3-d lsis. *Proceedings of the IEEE*, 97(1):49–59, Jan. 2009.
13. M. Koyanagi, T. Nakamura, Y. Yamada, H. Kikuchi, T. Fukushima, T. Tanaka, and H. Kurino. Three-dimensional integration technology based on wafer bonding with vertical buried interconnections. *IEEE Trans. Electron Devices*, 53(11):2799–2808, 2006.
14. D. Kroft. Lockup-Free Instruction Fetch/Prefetch Cache Organization. *ISCA*, pages 81–88, 1981.
15. G. H. Loh. 3d-stacked memory architectures for multi-core processors. In *ISCA '08: Proceedings of the 35th International Symposium on Computer Architecture*, pages 453–464, 2008.
16. G. H. Loh, Y. Xie, and B. Black. Processor Design in 3D Die-Stacking Technologies. *IEEE Micro*, 27(3):31–48, 2007.
17. G. H. Loh, Y. Xie, and B. Black. Processor Design in 3D Die-Stacking Technologies. *Micro, IEEE*, 27(3):31–48, may. 2007.
18. P. Marchal, B. Bougard, G. Katti, M. Stucchi, W. Dehaene, A. Papanikolaou, D. Verkest, B. Swinnen, and E. Beyne. 3-d technology assessment: Path-finding the technology/design sweet-spot. *Proceedings of the IEEE*, 97(1):96–107, Jan. 2009.
19. J. Mayega, O. Erdogan, P. M. Belemjian, K. Zhou, J. F. McDonald, and R. P. Kraft. 3d direct vertical interconnect microprocessors test vehicle. In *GLSVLSI '03: Proceedings of the 13th ACM Great Lakes symposium on VLSI*, pages 141–146, New York, NY, USA, 2003. ACM.
20. N. Muralimanohar, R. Balasubramonian, and N. P. Jouppi. CACTI 6.5. Technical Report HPL-2009-85, HP Labs, 2009.
21. A. Musa, Y. Sato, R. Egawa, H. Takizawa, K. Okabe, and H. Kobayashi. An On-chip Cache Design for Vector Processors. In *MEDEA '07: Proceedings of the 2007 workshop on MEmory performance*, pages 17–23, New York, NY, USA, 2007. ACM.
22. A. Musa, Y. Sato, T. Soga, K. Okabe, R. Egawa, H. Takizawa, and H. Kobayashi. A shared cache for a chip multi vector processor. In *MEDEA '08: Proceedings of the 9th workshop on MEmory performance*, pages 24–29, New York, NY, USA, 2008. ACM.
23. J. S. Pak, C. Ryu, and J. Kim. Electrical characterization of trough silicon via (tsv) depending on structural and material parameters based on 3d full wave simulation. In *Electronic Materials and Packaging, 2007. EMAP 2007. International Conference on*, pages 1–6, Nov. 2007.
24. K. Puttaswamy and G. Loh. The impact of 3-dimensional integration on the design of arithmetic units. In *Proceeding of the IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 4951–4954, May 2006.
25. K. Puttaswamyt and G. H. Loh. Scalability of 3d-integrated arithmetic units in high-performance microprocessors. In *DAC '07: Proceedings of the 44th annual Design Automation Conference*, pages 622–625, New York, NY, USA, 2007. ACM.
26. J. Tada, R. Egawa, K. Kawai, H. Kobayashi, and G. Goto. A Middle-Grain Circuit Partitioning Strategy for 3-D Integrated Floating-Point Multipliers. In *2011 IEEE International 3D Systems Integration Conference (3DIC)*, pages 1–8, 2012.
27. Y. Takagi, H. Sato, Y. Wagatsuma, K. Mizuno, and K. Sawaya. Study of High Gain and Broadband Antipodal Fermi Antenna with Corrugation. In *2004 International Symposium on Antennas and Propagation*, pages 69–72, 2004.
28. Y.-F. Tsai, F. Wang, Y. Xie, N. Vijaykrishnan, and M. J. Irwin. Design space exploration for 3-d cache. *IEEE Trans. Very Large Scale Integr. Syst.*, 16(4):444–455, 2008.
29. B. Vaidyanathan, W.-L. Hung, F. Wang, Y. Xie, V. Narayanan, and M. J. Irwin. Architecting microprocessor components in 3d design space. In *VLSID '07: Proceedings of the 20th International Conference on VLSI Design held jointly with 6th International Conference*, pages 103–108, Washington, DC, USA, 2007. IEEE Computer Society.
30. S. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz, D. Finan, P. Iyer, A. Singh, T. Jacob, S. Jain, S. Venkataraman, Y. Hoskote, and N. Borkar. An 80-Tile 1.28TFLOPS Network-on-Chip in 65nm CMOS. In *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pages 98–589, feb. 2007.
31. X. Wu, J. Li, L. Zhang, E. Speight, R. Rajamony, and Y. Xie. Hybrid Cache Architecture with Disparate Memory Technologies. In *ISCA '09: Proceedings of the 36th annual international symposium on Computer architecture*, pages 34–45, New York, NY, USA, 2009. ACM.