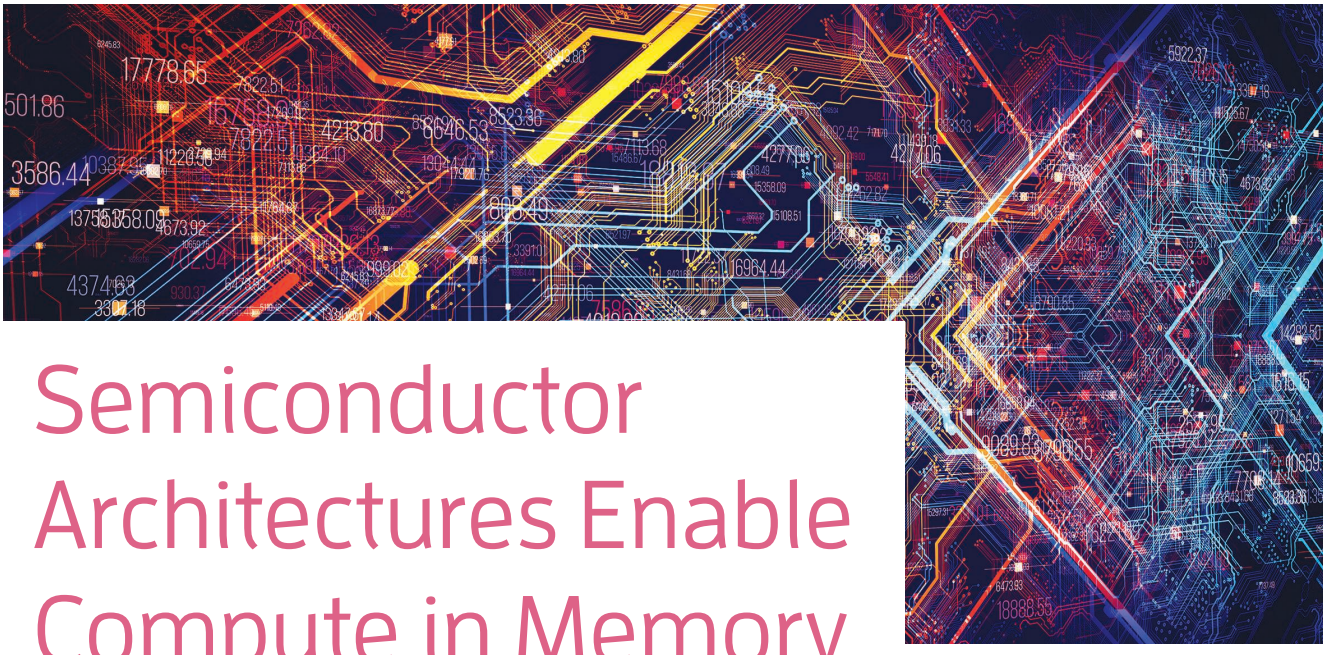

Semiconductor Architectures Enable Compute in Memory

Jim Handy, Objective Analysis
Tom Coughlin, Coughlin Associates, Inc.

Compute in memory (CIM) promises faster and lower power processing of data. Recently presented papers at the 2023 IEEE ISSCC gave some examples of how various semiconductor architectures can enable CIM devices for various computing applications.

Compute in memory (CIM) was a strong theme at IEEE's International Solid-State Circuits Conference (ISSCC) in February.[1] A total of 19 CIM papers were chosen for presentation at this very selective event, which carefully picks only those research efforts that share the most innovative thinking. The developments in semiconductor technologies, such as CIM, will have a great impact on future computing technology and the software that manages that technology.

Digital Object Identifier 10.1109/MC.2023.3252099
Date of current version: 3 May 2023

COMBINING MEMORY AND PROCESSING

All but two of the CIM chips highlighted in this article have been designed to push data-intensive inference tasks like image recognition to the edge. While this will lead to phenomenal reductions in the communication bandwidth between the edge device and the central server, conventional computing architectures would vastly increase endpoint energy consumption, and this would be impractical for most applications (and even impossible for many mobile devices). Neural network inference engines, particularly those that combine memory and processing, provide a compelling alternative since they perform basic recognition tasks while consuming very little energy.

An earlier article, in the May issue of *Computer,*[2] explained the need for CIM but didn't go into concrete examples. This article highlights papers that show the difficulty of architecting a quality CIM chip for edge inference, where most of these devices are expected to be used, and gives some remarkable ideas of how to address them.

0018-9162/23©2023IEEE

EDITOR **TOM COUGHLIN**
Coughlin Associates; tom@tomcoughlin.com

ANALOG OR DIGITAL?

One perennial question continues to polarize the research community. Should data (for example, from various sensors and cameras) be processed in linear circuits (analog), or should they be digitally processed using standard CMOS logic? Valid arguments were presented for both sides of this argument by various papers, with one paper extending the digital argument to say that floating point math was preferable to the use of integers.

Digital proponents favor digital's precision, determinism, and freedom from corruption due to process, voltage, and temperature, while the advocates of analog processing share their opinion that this approach reduces complexity, which improves power consumption, performance, and die area. Although we feel unqualified to take part in this debate, we found it intriguing that one paper[3] combined digital math, for those weighted features with the highest values, with analog processing for the weighted features with the lowest values, using place values to select between the two approaches.

Quite interestingly, those architectures based on digital integers in neural networks often used very narrow data words of only four to five bits. Although this may sound highly imprecise to most computing professionals, we have heard time and again that neural networks tend to be very tolerant of low resolution, noise, and imprecision. This is a matter best left to those who deeply understand the math behind these devices.

SPARSE DATA

Neural networks are configured to accept dense feature inputs and multiply them by dense weights, but real-world data don't work this way. When data are collected from a variety of sources, they are often called *sparse*. There will be features that are redundant, unimportant, or even totally absent. This can also be true for data from a single source. For example, a video image is likely to contain the same background information frame after frame with only a fraction of its pixels representing something of real interest, like the image of a moving pedestrian. A lot of unnecessary computing resources and energy can be saved by removing these redundant data prior to them being processed, and a number of the papers mentioned novel ways to handle this task. While the authors didn't share much detail in the ways that this was performed, it is quite clear that this was a key design criterion for these systems, which had been architecturally optimized for cost, power consumption, and performance.

THE I/O CONUNDRUM

One of the most challenging issues, and the one that is most problematic for the von Neumann architecture, is the high cost in both the speed and power of managing I/O traffic between the processor and memory. As was illustrated in the May article,[2] data traffic consumes more than its fair share of processing energy, and one important CIM goal is to dramatically reduce this consumption. This requires very careful judgment to architect the CIM system to keep the most important data within the CIM chip for the longest time possible.

All of the ISSCC papers focused on this, with different approaches to solving the problem. For the most part, these were centered around storing the frequently used data in the on-chip memory.

One paper stood out.[4] One of Tsinghua University's five papers described a chip that used conventional caching techniques to determine which features should remain on-chip for later use. A look-ahead policy further helps by preloading values that are likely to be used in the near future.

Caching is a well-understood approach to reducing communication between a processor chip, even a CIM chip, and a larger external memory. It provides faster access while minimizing power-hungry bus traffic, all while improving processing speeds.[5]

One of Tsinghua University's five papers described a chip that used conventional caching techniques to determine which features should remain on-chip for later use.

TRAVELING SALESMEN

Although the vast majority of the conference's CIM papers described inference engines, two papers[6,7] gave examples of Ising machines, which are designed to solve *combinatorial optimization* problems. For those unfamiliar with this term, it is often referred to as the *Traveling Salesman Problem*. These problems can be enormously computationally intensive in a von Neumann architecture, but other computer architectures handle them well.

In the case of the Traveling Salesman Problem (Figure 1), the basic problem is to find the most efficient route for the salesman to visit all of his many accounts out of the numerous possible options. A brute force approach would be to sequentially evaluate every possible option. With an Ising machine, all solutions are examined in parallel, with stronger solutions naturally rising to the top. The Ising machine is a

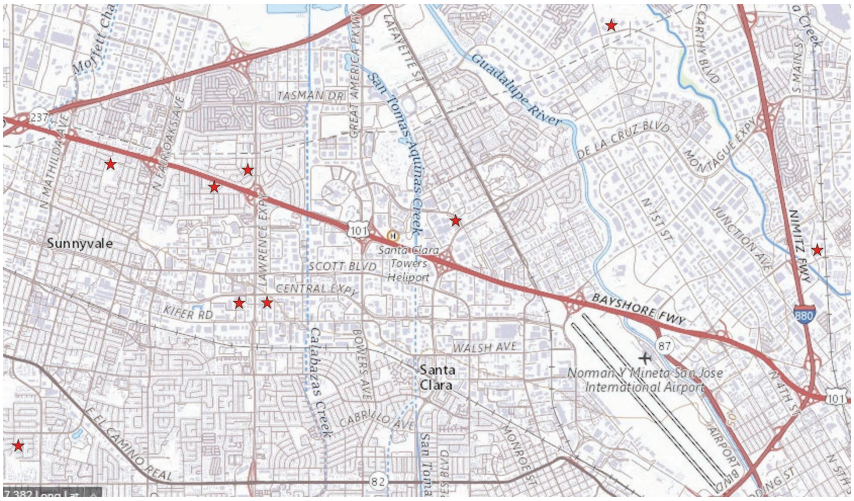

FIGURE 1. The Traveling Salesman Problem determines the optimum route for the salesperson to visit all of their accounts. (Source: USGS Map, public domain.)

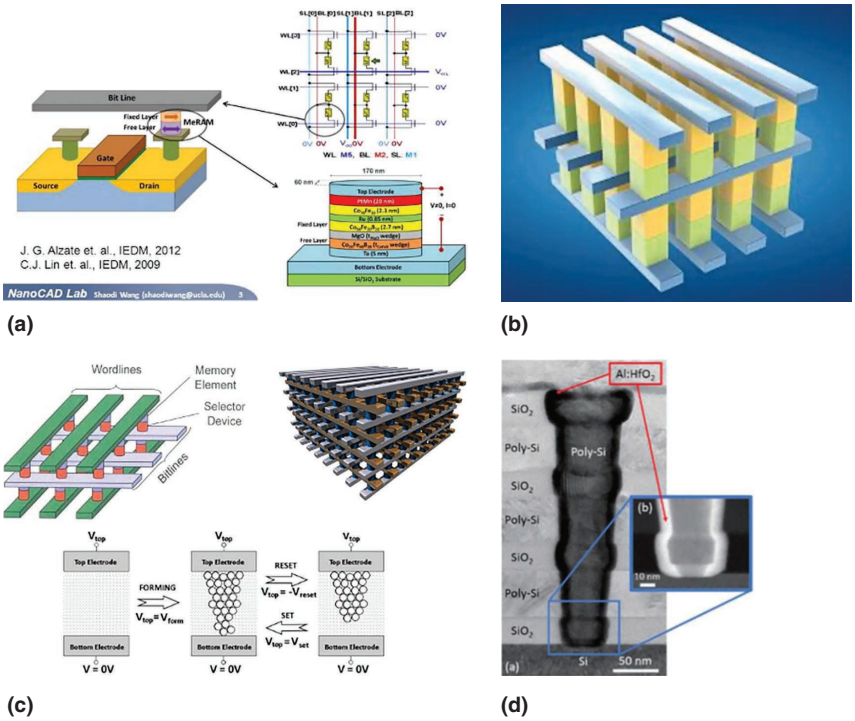

FIGURE 2. Leading emerging memory types. (a) Magnetoresistive RAM (MRAM). (Source: University of California; used with permission.) (b) Phase change memory (PCM). (Source: Intel Corporation; used with permission.) (c) ReRAM. (Source: upper left, Intel; upper right and lower, Technion, Israel Institute of Technology; used with permission.) (d) Ferroelectric RAM (FRAM). (Source: imec; used with permission.) IEDM: IEEE International Electron Devices Meeting; WL: word line; BL: bit line; OV: zero voltage or ground: SL: select level.

hardware accelerator that determines solutions to combinatorial optimization problems using the Ising model's natural inclination to converge to the optimum solution. Physicists use this same approach to model ferromagnetism. The typical Ising model statistically calculates the evolution of atomic spins' dipole moments, which can be in either of two states. The two Ising machines presented at ISSCC used different approaches based on standard CMOS logic.

One of the two papers, from the University of California, Santa Barbara,[7] interconnects 1,440 flip-flops in a way that causes them to wrestle against one another as linear devices until they settle down into a steady state, which can then be read as the solution. Weights guide the interconnections among all of these flip-flops for the problem at hand. The other, the Tokyo Institute of Technology's "Amorphica,"[6] is a completely digital CIM approach that migrates among four different morphologies. This one was found to come to a solution 58 times faster than a commercial 1-GHz GPU while consuming 1/500th as much power to consume only 1/30,000th of the GPU's energy.

MEMORY TYPES

Although the bulk of the papers described CIMs based on static RAM, which is readily available at any CMOS semiconductor fabrication, some used different memory types. KAIST's DynaPlasia chip[8] is based on an embedded dynamic RAM, which cuts the CIM cell's transistor count from 10–18 transistors to three. This results in commensurate savings in the die area to allow more operations in a given die area, hence lowering cost.

Taiwan's National Tsing Hua University introduced[9] a chip based on resistive RAM (ReRAM) that used both single-level cells and multilevel cells to balance performance against cost. Its nonvolatile memory allows the chip to boot faster than one whose weights must be loaded from external memory, and that provides important

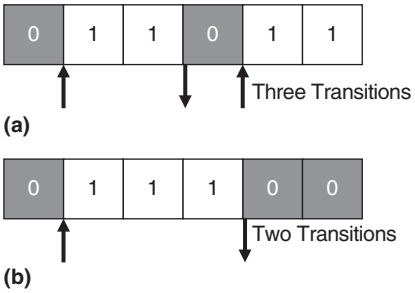

FIGURE 3. Reducing bit transitions by adjusting data values. (a) Original values. (b) Revised values. (Source: Objective Analysis).

energy savings in battery-operated systems at the network edge and in embedded devices.

ReRAM is only one of a number of emerging memory technologies that are being evaluated for their great potential for use with inference engines, usually in the form of neural networks. Other chips that perform CIM are also drawn to emerging memory technologies thanks to their persistence (they are all nonvolatile), their low energy consumption, and other aspects of their performance. The leading emerging memory candidates are illustrated in Figure 2 and are detailed in a report by Coughlin Associates and Objective Analysis.[10]

Both China's Southeast University[11] and Taiwan's National Tsing Hua University[12] described their spin-transfer torque magnetoresistive RAM (MRAM)-based inference engines. The Tsing Hua chip reduces power consumption by minimizing transitions among ones and zeros in its serial data processor. It performs this interesting task by nudging values with several transitions to nearby values with fewer transitions (Figure 3). The resultant errors are reported to be relatively minor, which is in keeping with reports from others about neural networks' error tolerance.

The sheer number and variety of options presented by these highly innovative papers show that the industry is nowhere near reaching a consensus regarding which technology best fits edge and embedded device computing. CIM is a rich field, and it will see many further advancements before it becomes a common part of our everyday lives. But it will become commonplace, and when it does, we will hardly notice that such innovative products have even become a critical part of our everyday lives.

ACKNOWLEDGMENT

Tom Coughlin is the corresponding author.

REFERENCES

1. "ISSCC 2023," in *Proc. Int. Solid-State Circuits Conf.*, San Francisco, CA, USA, Feb. 2023. [Online]. Available: http://www.ISSCC.org
2. T. Coughlin and B. Tonti, "Computing nearer to data," *Computer*, vol. 55, no. 7, pp. 82–87, Jul. 2022, doi: 10.1109/MC.2022.3171354.
3. P.-C. Wu et al., "A 22nm 832kb hybrid-domain floating-point SRAM in-memory-compute macro with 16.2-70.2TFLOPS/W for high-accuracy AI-edge devices," presented at the Int. Solid-State Circuits Conf., San Francisco, CA, USA, 2023, Paper 7.1.
4. F. Tu et al., "TensorCIM: A 28nm 3.7nJ/gather and 8.3TFLOPS/W FP32 digital-CIM tensor processor for MCM-CIM-based beyond-NN acceleration," presented at the Int. Solid-State Circuits Conf., San Francisco, CA, USA, 2023, Paper 16.4.
5. J. Handy, *The Cache Memory Book*. San Mateo, CA, USA: Morgan Kaufmann, 1993.
6. K. Kawamura et al., "Amorphica: 4-replica 512 fully connected spin 336 MHz metamorphic annealer with programmable optimization strategy and compressed-spin-transfer multi-chip extension," presented at the Int. Solid-State Circuits Conf., San Francisco, CA, USA, 2023, Paper 2.39.
7. J. Bae et al., "CTLE-Ising: A 1440-spin continuous-time latch-based Ising machine with one-shot fully-parallel spin updates featuring equalization of spin states," presented at the Int. Solid-State Circuits Conf., San Francisco, CA, USA, 2023, Paper 7.9.
8. S. Kim et al., "DynaPlasia: An eDRAM in-memory-computing-based reconfigurable spatial accelerator with triple-mode cell for dynamic resource switching," presented at the Int. Solid-State Circuits Conf., San Francisco, CA, USA, 2023, Paper 16.5.
9. W.-H. Huang et al., "A nonvolatile AI-edge processor with 4MB SLC-MLC hybrid-mode ReRAM compute-in-memory macro and 51.4-251TOPS/W," presented at the Int. Solid-State Circuits Conf., San Francisco, CA, USA, 2023, Paper 16.6.
10. "Emerging memories enter next phase," Coughlin Associates and Objective Analysis, Atascadero, CA, USA, 2022. [Online]. Available: https://Objective-Analysis.com/reports/#Emerging
11. H. Cai et al., "A 28nm 2Mb 22.4-41.5TOPS/W STT-MRAM computing-in-memory macro with a refined bit cell for AI inference," presented at the Int. Solid-State Circuits Conf., San Francisco, CA, USA, 2023, Paper 33.
12. Y.-C. Chiu et al., "A 22nm 8Mb 46.4-160.1-TOPS/W STT-MRAM near-memory-computing macro with 8b precision for AI-edge devices," presented at the Int. Solid-State Circuits Conf., San Francisco, CA, USA, 2023, Paper 33.2.

JIM HANDY is the general director of Objective Analysis, Los Gatos, CA 95032 USA. Contact him at jim.handy@objective-analysis.com.

TOM COUGHLIN is president of Coughlin Associates, Inc., San Jose, CA 95124 USA. He is a Fellow of IEEE. Contact him at tom@tomcoughlin.com.