

**Gemini Compute-In-Memory
Power and Performance Application Advantages**



Contents

Introduction.....	3
Compute-In-Memory (CIM) Review.....	3
GSI Technology Associative Processing Unit CIM Architecture	5
APU applications	6
GSI’s APU-CIM Architecture Advantages and Business Value	7
Flexibility	8
Sustainable Application Performance	8
Scalability	9
Comparisons	9
High Performance Compute example: Synthetic Aperture Radar Processing.....	9
Custom search application: Small molecule drug discovery	10
High Performance Compute: Salted Response-based cryptography comparison (SHA-1).....	10
HNSW Index Build Acceleration.....	11
Conclusion	11

Introduction

In the paper [Compute-in-Memory Computational Devices](#), we compared compute-in-memory with alternate architectures, even some often mis-interpreted as compute-in-memory. These included compute-near-memory and process-in-memory architectures and devices that included these structures. In this paper, we will concentrate on a purely digital compute-in-memory architecture, discuss applications the technology is well suited for, and validate the advantages purported to CIM in the previous paper in these applications.

Compute-In-Memory (CIM) Review

GSI's Compute-In-Memory (CIM) Gemini® associative processing technology, also referred to as In-Memory Compute (IMC), does compute directly **in** the memory. Gemini-I® and Gemini-II® are digital implementations of this and are devices that compute efficiently by incurring very little data transfer. This results in higher performance and much lower power. The special memory structures allow for native operations, such as Tanimoto distance measure, logical shifts, and simple Boolean operations within the memory structures. These characteristics make it very beneficial for AI inference and high-performance computing workloads where bit-slice flexibility can provide advantages of low power and high density for optimum dynamic precision processing. Such structures can then be amended with compute structures as used in compute-near-memory devices to provide special purpose acceleration capabilities, for example, cases where specialized processing is required on longer words on a regular basis for specific signal processing workloads.

This is different from traditional Von Neumann architectures that multiply the cores they have internally and then use cache memory per core. This is just an integrated version of system level separation of compute and memory chips. In traditional von Neumann and near-memory architectures, data movement between compute and memory at board level or chip level becomes costly in terms of time and energy. In recent data-centric applications, such as Artificial Intelligence/Machine Learning (AI/ML), this problem is aggravated with large amounts of data movement.

Figure 1 is a side-by-side conceptual illustration of the three architecture types:

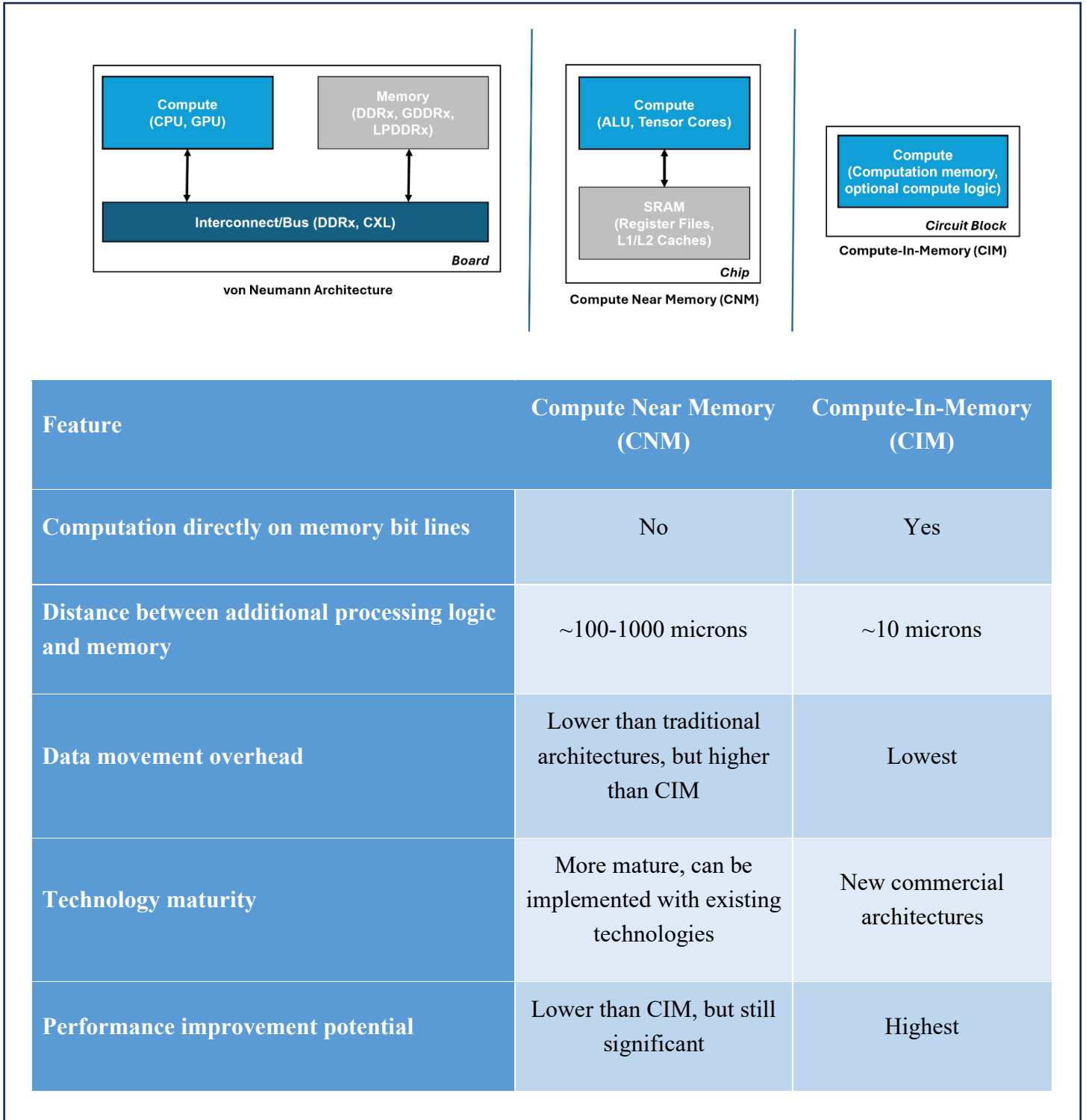


Figure 1: Computing Architectural Types

GSI Technology Associative Processing Unit CIM Architecture

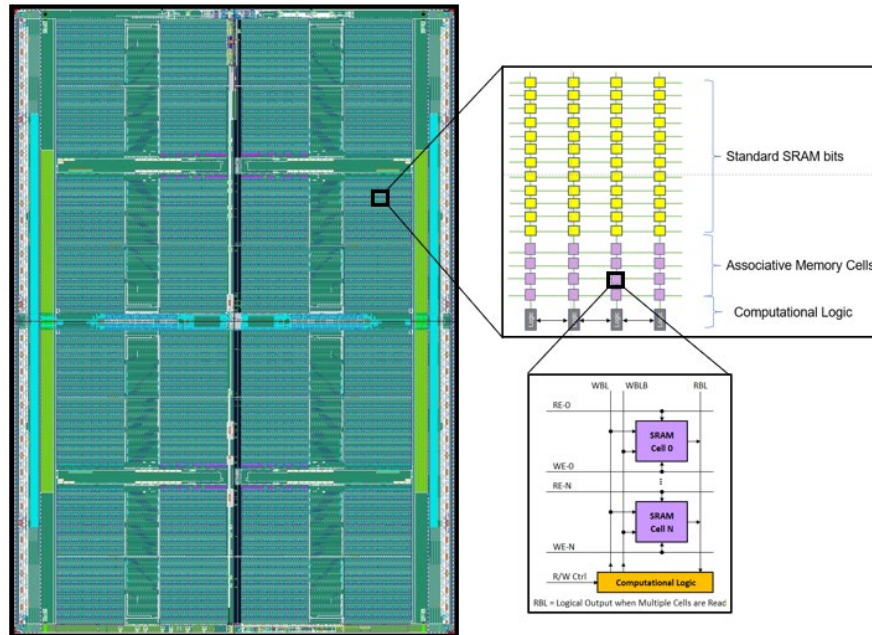


Figure 2: APU (Associative Processor Unit) CIM Architecture Overview
Compute logic is part of associative memory with a tightly coupled SRAM array.

Figure 2 (above) is an actual die photo of the GSI APU Gemini-I chip. If you think it looks a lot like standard computer memory, you are right! GSI has leveraged its twenty-eight years of experience in developing high performance DRAM and SRAM memory products for designing the Gemini products. You’ll also notice that with this architecture there are not the usual disparate sections found in CPU and GPU to move data around depending on what kind of processing you need.

The APU bit processors are groups of specialized SRAM bits that can be read simultaneously onto bit lines. This results in the ability to do Boolean operations on data during the process of reading and latching the result. Additional logic computational units are also embedded on the same bit lines enhancing the operations that can be performed. There are 2 million of these bit-engines in the first-generation APU (Gemini-I) and each can be programmed to compute simple functions on the data “in-memory.” That’s a massive number of simultaneous computations! In fact, it amounts to a little over 2 peta-operations per second at just 500 MHz. And by computing on the data where it lives, the APU eliminates many of the memory transfer bottlenecks that exist in the alternative compute architectures.

Note, the minimum measurable block of computations that includes all transfer and processing in a traditional von Neumann architecture is what can run in the computation unit and the local register memory that provides the inputs for it to operate on. In this CIM architecture, it is just this bit processor array because computation and storage occur in-place in the memory.

Compute-in-memory architecture provides the best cost and performance benefit by exploiting the circuit techniques and inherent spatial architecture capable of using SRAM (static random-access memory) technology and which can be quantified with total cost of ownership (TCO).

In addition to these benefits, the APU architecture also has a closely coupled distributed massive register array that can support the bit processing structures. The Gemini-I has 96Mb of register memory supporting the bit engines just 100microns apart. The second-generation APU, Gemini-II, has 768Mb of register memory. At that close proximity, the energy expended to transfer additional information from the registers into the compute-in-memory engines is 0.05pJ/bit for the 28nm Gemini-I part.

APU Applications

Compute-in-memory has been shown to provide differentiable benefits in a wide range of applications, such as AI/ML, vector search, real-time analytics, image classification, and a plethora of HPC applications, such as SAR, genomics, and cryptographic processing.

The Gemini-I is in production in Biovia Pipeline Pilot small molecule generic drug search where it is used to do Tanimoto-based distance measure from bit vectors that are based on ECFP fingerprints.¹ This lowers search times from hours to seconds and can add additional features to the search at the same time.

GSI also provides a Fast Vector Search (FVS) library and API for traditional vector search with application in a number of different industries. The technology has application for corporate password recovery where massive iterations of SHA processing can occur without constant data retrieval and storage.

Systems are in production for doing high performance computation of Synthetic Aperture Radar image generation on 1U mobile servers. Such systems provide mobile processing capability, such as real-time in a plane, to what hitherto was only an enterprise grade processing workload.

¹ <https://gsitechnology.com/wp-content/uploads/sites/default/files/files/GSIT-Weizmann-Case-Study.pdf>

University of Northern Arizona has published the benefits of the flexible massively parallel structure for cryptographic processing.²

Cornell University published a detailed circuit-based expose on the benefits of the architecture for bit-matching by targeting it for seed location filtering for DNA Read Mapping.³

The APU technology is particularly relevant in industries where processing large data sets in real-time with performance/watt/\$ efficiency is important. The solution provides flexible hardware programming, and the lack of a required fixed bit-level framework provides futureproofing, even when the technology is deployed in field, providing a competitive advantage.

GSI's APU-CIM Architecture Advantages and Business Value

GSI's APU represents a paradigm shift in computer architecture, offering an unprecedented blend of cutting-edge features, including associative, massively parallel, true compute-in-memory, bit-flexible framework processing capabilities. The following is a summary of key benefits the APU offers for a wide range of AI, Search, and HPC applications.

High power efficiency and low power consumption:

The APU architecture can compute on data that is in memory. Moving data among the bit processors achieves Boolean operations at single clock rates that can be used to build up complex functions with reduced instructions. As this processing does not require data to be moved from cache, the system speeds are immense. Power is orders of magnitude lower than traditional von Neumann GPUs, which rely on cache loads for operations because even adding ALUs in an APU architecture along the bit-lines involve memory transfers that are only 20 microns apart. This tremendously fast processing array can also be loaded from register memory. As this memory is very large, larger than the in-place computation memory, and it is distributed and about 100 microns away from the processing memory, again we see about two to three orders of magnitude less movement of data resulting in faster processing and much lower power consumption. This data transfer on the first-generation APU amounts to about 0.05pico-Joules per bit.

² https://jan.ucc.nau.edu/mg2745/publications/Lee_DUAC_ICPPW2023.pdf

³ <https://gsitechnology.com/wp-content/uploads/sites/default/files/files/Accelerating-Seed-Location-Filtering-in-DNA-Read-Mapping-Using-Commercial-Compute-in-SRAM-Architecture.pdf>

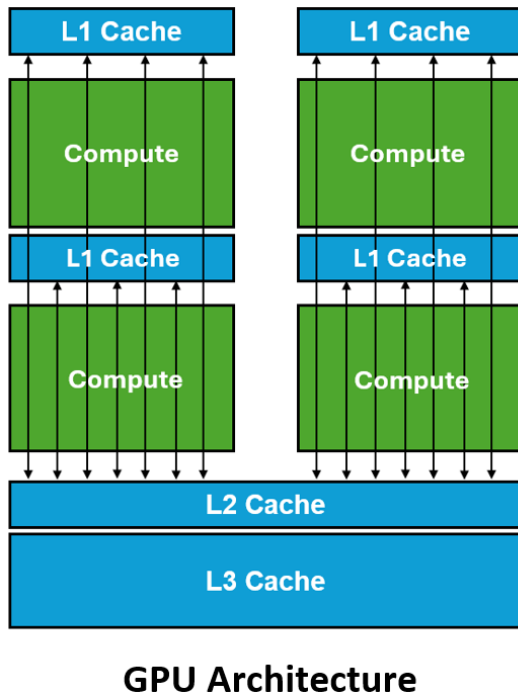


Figure 3: Traditional Von Neumann GPU Core Structure

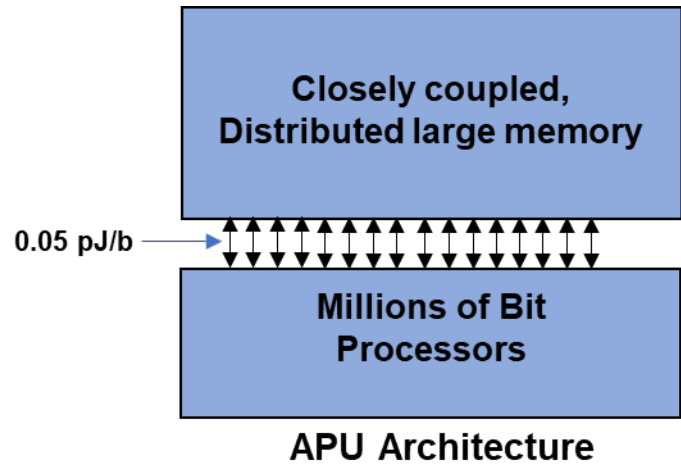


Figure 4: APU CIM Bit Processors with Large Working Memory Support

Flexibility

The APU architecture does not inflict a pre-defined framework on the programmer. Instead, any framework, including custom bit widths down to single bits, can be used natively in the device. Multiple different bit widths can then be devised using this methodology to allow even cycle-by-cycle dynamic precision to conserve compute and memory resources without sacrificing accuracy. This provides for futureproofing in the ability to create algorithms on bit widths that may not even be considered yet. An example of natively supporting new architectures is the ability to support the Microscaling Format (MX Format). This architecture can also handle Binary Neural Network (BNN) frameworks now.

Sustainable Application Performance

The combination of millions of bit processors in the APU array and the large, closely coupled distributed memory means that the architecture is capable of eliminating application memory bottlenecks for models that can be housed in its array. This leads to 100% core utilization, highest throughput, and lowest power consumption. The second-generation Gemini-II incorporates 96MB of

register memory allowing single devices to address larger models at the edge, or building out datacenter systems that can address very large models at full core utilization.

Scalability

Because compute-in-memory devices are memories in one aspect, systems utilizing these components can scale as you would add memory to a system—by merely adding more components and without the need for complex connectivity. Using this methodology, it becomes quite normal to have a single low-cost server system with 8 or 16 cards in place. Massive scaling now becomes possible with commodity 10gE server interconnects.

These benefits compound to allow users to build systems from the edge to datacenter deployments with the best TCO (performance/W/\$).

Comparisons

High Performance Compute example: Synthetic Aperture Radar Processing

Distinguished by: Varying “random” variable precision, parallel processing, one-to-all sensor data input to processors

5 Yr. Power Cost Comparison, 10k x 10k Pixels, 0.5m Resolution, 1 second Processing			
	Intel CPU	Nvidia V100	GSI APU Gemini-I
Form Factor (x)	20.8x 2U server (16x)	9.6 2U server (7x)	2.8 1U server (1x)
Total Power cost (5 years)	\$503,000	\$241,000	\$46,000

Custom Search Application: Small Molecule Drug Discovery

Distinguished by: Desired low threshold comparison, large vector lengths: tested to 8000-bit widths

Biovia Pipeline Pilot Customer Benchmark		
	Intel CPU (1 server)	GSI APU Gemini-I (1 card)
Bit vector sizes	Up to 512	Tested to 8192
Power consumption	355W	226W (Incl CPU)
Threshold limit (lower better)	0.8	0.1
Processing time (512 bit, 38M entries)	Several minutes	1.1seconds

High Performance Compute: Salted Response-based Cryptography Comparison (SHA-1)

Distinguished by: Cryptography processing

Salted Response-based cryptography comparison (SHA-1)			
	Dual AMD EPYC 7542	Nvidia A100	GSI APU Gemini-I (1 card)
Search time (Exhaustive)	12.09s	1.56s	1.62s
Search time (average)	6.04s	0.85s	0.83s
Power consumption (search only)	Not completed, >>253W	253W	83W

HNSW Index Build Acceleration

Distinguished by: Parallelism and hardware graph build acceleration

Index Build Times for subsets of Deep1B			
	Pgvector (workers = 32)	Hnswlib	GXL (APU=15)
50M	2.0 hrs	12.8 mins	6.7 mins
100M	4.5 hrs	27.5 mins	12.5 mins
250M	11.1 hrs	74.0 mins	28.7 mins

Conclusion

GSI’s in-memory computing represents a significant shift in computing paradigm with proven SRAM technology that is ready for high volume deployment. As the cost of traditional compute and memory in terms of energy, cooling, and diminishing returns on scale, continues to increase, in-memory computing has been proven to be able to more efficiently take on traditional von Neumann processing workloads and becoming an increasingly vital component of modern IT infrastructure, enabling applications and actions that were previously economically unattainable.

We’ve covered several application types that this technology has shown differentiable performance—sometimes orders of magnitude—better than traditional compute. Of particular importance is this technology to three types of processing: Inference workloads that are accelerated with softmax can be handled natively with Tanimoto classification; high performance computations based on or that can be efficiently transformed into binary representative processing; and processing that can be categorized by high repetitive loops, particularly with one or both of the previous differentiators.

For more information, please contact associativecomputing@gstechnology.com.