

EXHIBIT 803-A-06

Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)

Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”) anticipates and/or renders obvious (alone or in combination with other references), including at least under Complainant’s apparent infringement theory:

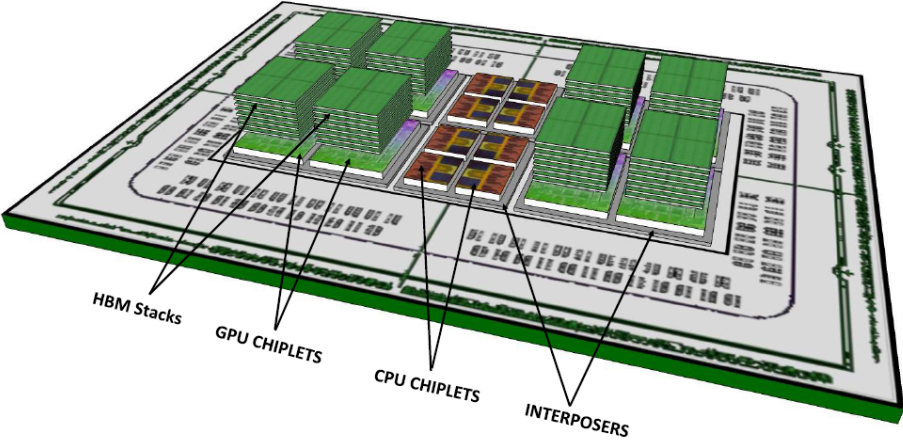
- claims 1-3, 7-10 of U.S. Patent No. 11,841,803 (“the ’803 patent”) under 35 U.S.C. §§ 102 and/or 103.

NVIDIA has charted the portions of the reference that correspond to each element, including at least under Complainant’s apparent interpretation of the claims as reflected in Complainant’s infringement contentions. Where the chart below states that the reference “discloses,” “teaches,” or “suggests” a claim limitation, such disclosure, teaching, or suggestion may be express, implied, or inherent. The cited evidence is merely illustrative, and NVIDIA reserves the right to cite alternative or additional evidence.

No part of this exemplary chart construes, or is intended to construe, the specification, file history, or claims of the asserted patents. In addition, nothing stated in this chart shall be treated as an admission or suggestion that NVIDIA agrees with Complainant regarding either the scope of any of the asserted claims or Complainant’s apparent interpretation of the scope of the asserted claims, or that NVIDIA’s accused products meet any limitations of the claims.

To the extent that Complainant contends that Vijayaraghavan does not disclose one or more limitations of the asserted claims, it would have been obvious to combine the teachings of Vijayaraghavan with (1) the knowledge of one of ordinary skill in the art, and/or (2) the teachings of any of the prior art set forth in NVIDIA’s other invalidity charts and cover pleading, to show all the limitations of the asserted claims, including for the reasons set forth in NVIDIA’s Invalidity Contention Cover Pleading, which are incorporated herein.

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
1[Pre]: A system, comprising:	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p><i>See</i> evidence cited for ’803 patent claims 1-3, 7-10.</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
1[A]: a central processing unit (CPU) communicably coupled to a first graphics processing unit (GPU) chiplet of a GPU chiplet array,	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p>Vijayaraghavan at Fig. 2.</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	 <p data-bbox="926 824 1640 862">Figure 2. Exascale Heterogeneous Processor (EHP)</p> <p data-bbox="682 906 1003 943">Vijayaraghavan at Fig. 3.</p>

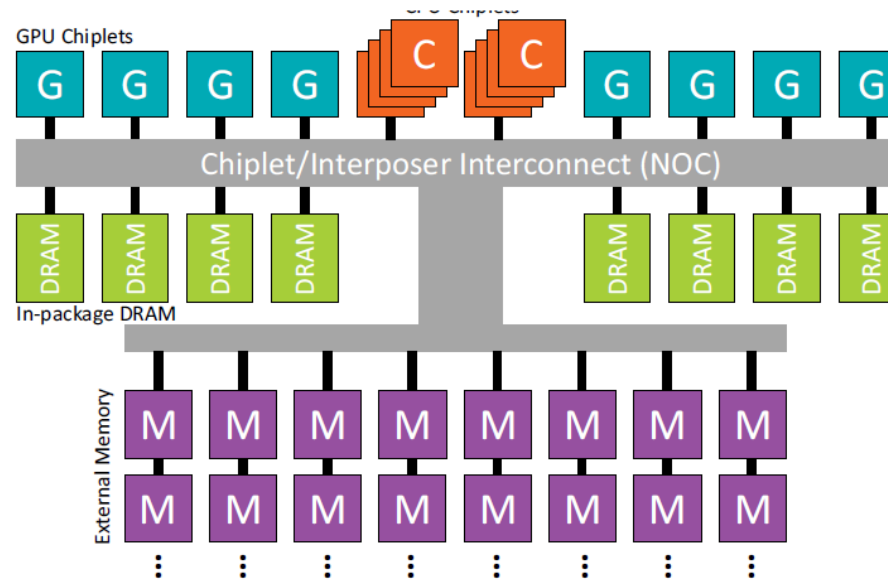


Figure 3. Block diagram of the ENA memory system

Vijayaraghavan at Fig. 7.

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)												
	<div><p>The bar chart displays two metrics for three benchmarks: XSBench, SNAP, and CoMD. The y-axis represents percentages from 0% to 100%. The legend indicates that light blue bars represent the 'Percentage of out-of-chiplet traffic' and dark blue bars represent the 'Percentage of EHP perf. relative to monolithic EHP'.</p><table><thead><tr><th>Benchmark</th><th>Percentage of out-of-chiplet traffic</th><th>Percentage of EHP perf. relative to monolithic EHP</th></tr></thead><tbody><tr><td>XSBench</td><td>~95%</td><td>~85%</td></tr><tr><td>SNAP</td><td>~75%</td><td>100%</td></tr><tr><td>CoMD</td><td>~65%</td><td>100%</td></tr></tbody></table></div> <p>Figure 7. Out-of-chiplet traffic and impact on performance</p> <p>Vijayaraghavan at Abstract (“Abstract—The challenges to push computing to exaflop levels are difficult given desired targets for memory capacity, memory bandwidth, power efficiency, reliability, and cost. This paper presents a vision for an architecture that can be used to construct exascale systems. We describe a conceptual Exascale Node Architecture (ENA), which is the computational building block for an exascale supercomputer. The ENA consists of an Exascale Heterogeneous Processor (EHP) coupled with an advanced memory system. The EHP provides a high-performance accelerated processing unit (CPU+GPU), in-package high-bandwidth 3D memory, and aggressive use of die-stacking and chiplet technologies to meet the requirements for exascale computing in a balanced manner. We present initial experimental analysis to demonstrate the promise of our approach, and we discuss remaining open research challenges for the community.”</p> <p>Vijayaraghavan at 85 (“Although there are potentially many paths to realize exascale computing [5]–[7], this paper presents one vision for how to construct an exascale machine using advances in key technologies including low-power and high-performance CPU cores,</p>	Benchmark	Percentage of out-of-chiplet traffic	Percentage of EHP perf. relative to monolithic EHP	XSBench	~95%	~85%	SNAP	~75%	100%	CoMD	~65%	100%
Benchmark	Percentage of out-of-chiplet traffic	Percentage of EHP perf. relative to monolithic EHP											
XSBench	~95%	~85%											
SNAP	~75%	100%											
CoMD	~65%	100%											

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>integrated energy-efficient GPU compute units (CUs), in-package high-bandwidth 3D memory, aggressive use of die-stacking and chiplet technologies, and advanced memory systems. Our coordinated research that spans from circuits to software enables a very efficient and tightly integrated processor architecture suitable for exascale systems.”)</p> <p>Vijayaraghavan at 85 (“We introduce an Exascale Node Architecture (ENA) as the primary building block for exascale machines. The ENA addresses the exascale-computing requirements through:</p> <ul style="list-style-type: none"> • A high-performance accelerated processing unit (APU) that integrates high-throughput GPUs with excellent energy efficiency required for exascale levels of computation, tightly coupled with high-performance multi-core CPUs for serial or irregular code sections and legacy applications • Aggressive use of die-stacking capabilities that enable dense component integration to reduce data-movement overheads and enhance power efficiency • A chiplet-based approach [8], [9] that decouples performance-critical processing components (e.g., CPUs and GPUs) from components that do not scale well with technology (e.g., analog components), allowing fabrication in disparate, individually optimized process technologies for cost reduction and design reuse in other market segments • Multi-level memories that enhance memory bandwidth with in-package 3D memory, which is stacked directly above high-bandwidth-consuming GPUs, while provisioning high-capacity memory outside of the package • Advanced circuit techniques and active power-management techniques which yield energy reductions with little performance impact • Hardware and software mechanisms to achieve high resilience and reliability with minimal impact on performance and energy efficiency • Concurrency frameworks that leverage the Heterogeneous System Architecture (HSA) [10] and Radeon Open Compute platform (ROCm) [11] software ecosystem to support new and existing applications with high-performance and high programmer productivity”).

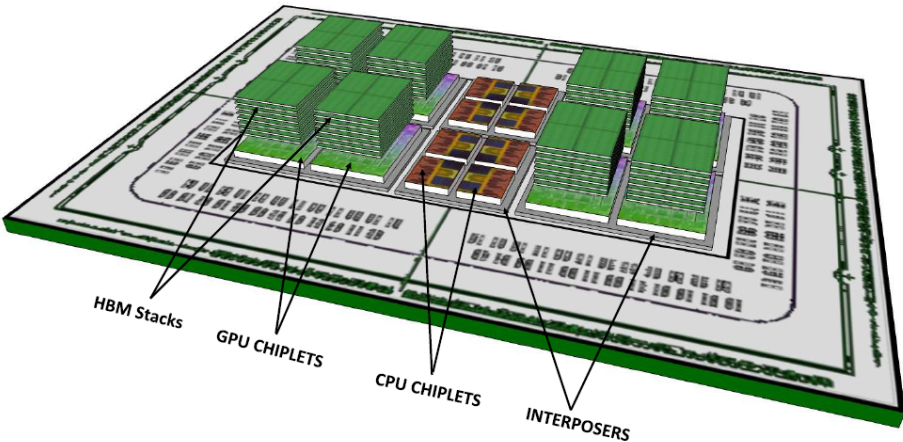
U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>Vijayaraghavan at 86 (“The EHP is an accelerated processing unit (APU) consisting of a balanced mix of CPU and GPU compute resources integrated with in-package 3D DRAM. The overall structure makes use of a modular “chiplet” design, with the chiplets 3D-stacked on other “active interposer” chips. The aggressive use of advanced packaging technologies enables a large amount of computational and memory resources to be located in a single package.”)</p> <p>Vijayaraghavan at 86 (“Fig. 2 shows a conceptual rendering of the EHP. The EHP consists of two primary types of resources. In the center of the EHP are two CPU clusters, each consisting of four multi-core CPU chiplets stacked on an active interposer base die. On either side of the CPU clusters are a total of four GPU clusters, each consisting of two GPU chiplets on a respective active interposer. Upon each GPU chiplet is a 3D stack of DRAM (e.g., some future generation of JEDEC high-bandwidth memory (HBM) [12]). The DRAM is directly stacked on the GPU chiplets to maximize bandwidth (the GPUs are expected to provide the peak computational throughput) while minimizing memory-related data movement energy and total package footprint. CPU computations tend to be more latency sensitive, and so the central placement of the CPU cores reduces NUMA-like effects by keeping the CPU-to-DRAM distance relatively uniform. The interposers underneath the chiplets provide the interconnection network between the chiplets (i.e., network on chip (NOC)) along with other common system functions (e.g., external I/O interfaces, power distribution, system management). Interposers maintain high-bandwidth connectivity among themselves by utilizing wide, short-distance, point-to-point paths.”)</p> <p>Vijayaraghavan at 87 (“The EHP uses eight GPU chiplets. Our initial configuration provisions 32 CUs per chiplet. Each chiplet is projected to provide two teraflops of double-precision computation, for a total of 16 teraflops. Based on the projected system size of 100,000 nodes, this would provide a total of 1.6 exaflops (we over-provision because real applications do not achieve 100% utilization).”)</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>Vijayaraghavan at 87 (“The EHP also employs eight CPU chiplets (four cores each), for a total of 32 cores, with greater parallelism through optional simultaneous multi-threading. The number of CPU cores was carefully chosen to provision enough single-thread performance for irregular code sections and legacy applications without exceeding the package’s area budget.”)</p> <p>Vijayaraghavan at 87 (“The CPU and GPU must be architected in a cohesive manner from an energy efficiency as well as programmability standpoint. To this end, Heterogeneous System Architecture (HSA) compatibility is one of the major design goals of the APU. HSA provides a system architecture where all computing elements (CPU, GPU, and possibly other accelerators) share a unified coherent virtual address space. This enables efficient programming and computation via several mechanisms like free exchange of pointers by both CPU and GPU code, eliminating expensive data copy operations, transparent management of CPU and GPU caches via cache coherence, task offloads by both CPU and GPU to each other or other CPU/GPU units [13], and efficient synchronization mechanisms. These features are supported by AMD’s Radeon Open Compute platform (ROCm) to improve the programmability of such heterogeneous systems.”)</p> <p>Vijayaraghavan at 87 (“2) Modular Chiplet Design: The performance require-ments of the exascale node require a large amount of compute and memory to be integrated into a single package. Rather than build a single, monolithic system on chip (SOC), we propose to leverage advanced die-stacking technologies to decompose the EHP into smaller components consisting of active interposers and chiplets. Each chiplet houses either multiple GPU compute units or CPU cores. The chiplet approach differs from conventional multi-chip module (MCM) designs in that each individual chiplet is not a complete SOC. For example, the CPU chiplet contains CPU cores and caches, but lacks memory interfaces and external I/O.¹ There are multiple benefits to this decompositional approach to SOC construction:</p> <p>Die Yield: Building a single monolithic SOC that provides the equivalent capabilities of what we propose for the EHP would result in an impractically large chip with prohibitive costs. Smaller chiplets have higher yield rates due to their size, and when combined with known-good-die (KGD) testing techniques, can be assembled into larger systems at reasonable cost.</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>This approach has already started garnering interest in both academia [9] and industry [8], [19]. Process Optimization: A monolithic SOC imposes a single process technology choice on all components in the system. With chiplets and interposers, each discrete piece of silicon can be optimized for its own functions. For example, the CPU chiplets can use performance-optimized devices and metal layers, while the GPU chiplets use density-optimized devices and metal. The interposer layers can use a more mature (i.e., less expensive) process technology node as the I/O components likely do not need transistors in the cutting-edge technology node nor as many metal routing layers as the compute chiplets.</p> <p>Re-usability: A single, large HPC-optimized APU would be great for HPC markets, but may be less appropriate for others. The decomposition of the EHP into smaller pieces enables silicon-level reuse. For example, one or more of the CPU clusters could be packaged together to create a conventional CPU-only server processor.”)</p> <p>Vijayaraghavan at 87 n.1 (“Contrast this to typical MCM parts that take identical CPU chips that are each fully functional SOC’s, and then place them in the same package. With chiplets, there is no option to take a single chiplet by itself and convert it into a complete product without additional silicon.”)</p> <p>Vijayaraghavan at 87 n.2 (“The primary point of differentiation, which is not a fundamental issue, is that most past 3D stacking research considered vertical stacks of identical/similar-sized chips, whereas active interposer configurations involve different-sized chips (i.e., the interposer vs. the chiplets) and some chips may also be stacked side by side (e.g., neighboring chiplets) rather than in a strictly vertical arrangement.”)</p> <p>Vijayaraghavan at 89 (“Traditional cycle-level simulators like gem5 [36] and GPGPU-Sim [37] may not be suitable for exploring the large and complex design space of the EHP (e.g., heterogeneous processors, 3D die stacking, multiple memory levels). Their execution times make it difficult to quickly gather first-order insights across a wide range of design candidates. Instead, we leverage an in-house high-level simulator for design-space exploration and use cycle-level simulation to account for the impacts of key microarchitecture choices.</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>Our high-level simulator [38] measures an application’s execution on current hardware and uses a variety of scaling models to estimate the same application’s performance and power on future hardware designs. For instance, we designed analytic performance [39] and power [40] scaling models for CPUs and we use data-movement performance counters combined with distance-based energy values for interconnect power calculations [41]. We use in-house technology-scaling models to estimate how power and our voltage-frequency curves will change.</p> <p>We use machine-learning models to estimate how measured GPU power and performance will scale as the hard-ware configurations change [42]. These models are trained on numerous applications on a wide range of existing hardware configuration points [43].</p> <p>These CPU and GPU models are effective for estimating the impact of high-level hardware-resource changes, but they assume that the underlying CPU and GPU microarchitectures remain the same as the hardware on which the initial measurements were taken. We use the AMD gem5 APU simulator [44] to account for differences from our EHP design (e.g., multi-chiplet organization) and adjust the high-level simulation results accordingly.”)</p> <p>Vijayaraghavan at 91 (“The benefits of a chiplet-based design were covered in Section II, but a multi-chip organization imposes additional latency for inter-chiplet communication (whether for coherence or main-memory access). Messages to a remote chiplet traverse from a source chiplet to the lower interposer layer through through-silicon vias (TSVs), go across the interposer, and go up to the destination chiplet through TSVs [9]. Thus, two extra vertical-communication hops are required compared to a hypothetical monolithic EHP. To assess the performance impact of these overheads, we compare a monolithic EHP design against our proposed multi-chiplet design. The findings below show only a small performance impact, suggesting that a chiplet-based design is feasible.</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>Finding 1: The out-of-chiplet traffic dominates the total traffic, ranging from 60-95% across kernels representing a wide range of behavior. This traffic includes traffic between GPU CUs on different chiplets as well as between the CPU cores and the GPU CUs. Given that there are eight GPU chiplets on the EHP, the remote-traffic rate demonstrates a fairly even distribution of accesses across chiplets. Finding 2: The performance impact of the large out-of-chiplet traffic is relatively small. The largest performance degradation compared to the monolithic EHP is 13%, while some application kernels, such as SNAP, have a negligible impact, as shown in Fig. 7.</p> <p>Takeaway: The small performance impact despite the large out-of-chiplet traffic is due to a variety of reasons. First, the GPU’s massive parallelism is effective at latency hiding. Kernels with abundant parallelism tend to be able to hide much of the latency to other chiplets. Second, some applications (especially those with cache-friendly access patterns) do not stress the memory system and are not sensitive to a few cycles of additional memory latency. Thus, our chiplet organization makes a compelling trade-off between longer network latency and improvement in die yield and cost.”)</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
1[B]: wherein the GPU chiplet array includes: the first GPU chiplet communicably coupled to the CPU via a bus; and	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p><i>See evidence cited for ’803 patent limitation 1[A].</i></p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
<p>1[C]: a second GPU chiplet communicably coupled to the first GPU chiplet via a passive crosslink,</p>	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p>Vijayaraghavan at Fig. 2.</p>  <p>Figure 2. Exascale Heterogeneous Processor (EHP)</p> <p>Vijayaraghavan at Fig. 3.</p>

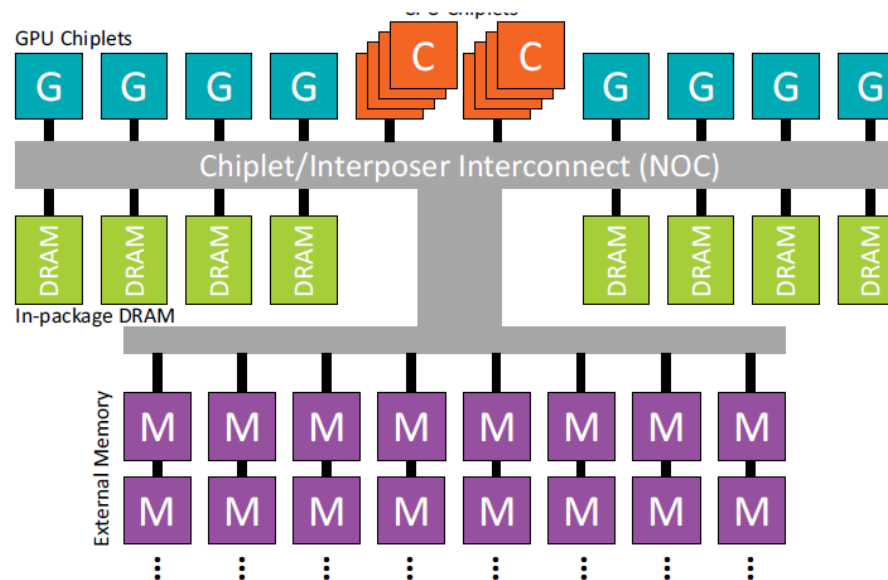


Figure 3. Block diagram of the ENA memory system

Vijayaraghavan at 86 (“The EHP is an accelerated processing unit (APU) consisting of a balanced mix of CPU and GPU compute resources integrated with in-package 3D DRAM. The overall structure makes use of a modular “chiplet” design, with the chiplets 3D-stacked on other “active interposer” chips. The aggressive use of advanced packaging technologies enables a large amount of computational and memory resources to be located in a single package.”)

Vijayaraghavan at 86 (“Fig. 2 shows a conceptual rendering of the EHP. The EHP consists of two primary types of resources. In the center of the EHP are two CPU clusters, each consisting of four multi-core CPU chiplets stacked on an active interposer base die. On either side of the CPU clusters are a total of four GPU clusters, each consisting of two GPU chiplets on a

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>respective active interposer. Upon each GPU chiplet is a 3D stack of DRAM (e.g., some future generation of JEDEC high-bandwidth memory (HBM) [12]). The DRAM is directly stacked on the GPU chiplets to maximize bandwidth (the GPUs are expected to provide the peak computational throughput) while minimizing memory-related data movement energy and total package footprint. CPU computations tend to be more latency sensitive, and so the central placement of the CPU cores reduces NUMA-like effects by keeping the CPU-to-DRAM distance relatively uniform. The interposers underneath the chiplets provide the interconnection network between the chiplets (i.e., net-work on chip (NOC)) along with other common system functions (e.g., external I/O interfaces, power distribution, system management). Interposers maintain high-bandwidth connectivity among themselves by utilizing wide, short-distance, point-to-point paths.”)</p> <p>Vijayaraghavan at 87 (“2) Modular Chiplet Design: The performance require-ments of the exascale node require a large amount of compute and memory to be integrated into a single package. Rather than build a single, monolithic system on chip (SOC), we propose to leverage advanced die-stacking technologies to decompose the EHP into smaller components consisting of active interposers and chiplets. Each chiplet houses either multiple GPU compute units or CPU cores. The chiplet approach differs from conventional multi-chip module (MCM) designs in that each individual chiplet is not a complete SOC. For example, the CPU chiplet contains CPU cores and caches, but lacks memory interfaces and external I/O.¹ There are multiple benefits to this decompositional approach to SOC construction:</p> <p>Die Yield: Building a single monolithic SOC that provides the equivalent capabilities of what we propose for the EHP would result in an impractically large chip with prohibitive costs. Smaller chiplets have higher yield rates due to their size, and when combined with known-good-die (KGD) testing techniques, can be assembled into larger systems at reasonable cost. This approach has already started garnering interest in both academia [9] and industry [8], [19]. Process Optimization: A monolithic SOC imposes a single process technology choice on all components in the system. With chiplets and interposers, each discrete piece of silicon can be optimized for its own functions. For example, the CPU chiplets can use performance-optimized devices and metal layers, while the GPU chiplets use density-optimized devices and</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>metal. The interposer layers can use a more mature (i.e., less expensive) process technology node as the I/O components likely do not need transistors in the cutting-edge technology node nor as many metal routing layers as the compute chiplets.</p> <p>Re-usability: A single, large HPC-optimized APU would be great for HPC markets, but may be less appropriate for others. The decomposition of the EHP into smaller pieces enables silicon-level reuse. For example, one or more of the CPU clusters could be packaged together to create a conventional CPU-only server processor.”)</p> <p>Vijayaraghavan at 87 (“3) Active Interposers: There are several options for the assembly of multiple discrete pieces of silicon within the same package. For example, MCMs have been used for years, and they have even been proposed in the context of chiplets as advocated by Marvell’s MoChi (Modular Chips) concept [8]. Passive silicon interposers are also already in volume production, in particular for the integration of GPU and 3D memory as demonstrated by AMD’s RadeonTM R9 Fury GPU [20]. However, due to the sheer amount of compute, memory, and other logic that we aim to integrate into the EHP package, we found true 3D stacking on top of active interposers to be desirable. Active interposers are fundamentally no different than other 3D stacking approaches considered by past research², and preliminary prototypes have even been successfully constructed and demonstrated [21].”)</p> <p>Vijayaraghavan at 87 n.2 (“The primary point of differentiation, which is not a fundamental issue, is that most past 3D stacking research considered vertical stacks of identical/similar-sized chips, whereas active interposer configurations involve different-sized chips (i.e., the interposer vs. the chiplets) and some chips may also be stacked side by side (e.g., neighboring chiplets) rather than in a strictly vertical arrangement.”)</p> <p>Vijayaraghavan at 91 (“The benefits of a chiplet-based design were covered in Section II, but a multi-chip organization imposes additional latency for inter-chiplet communication (whether for coherence or main-memory access). Messages to a remote chiplet traverse from a source chiplet to the lower interposer layer through through-silicon vias (TSVs), go across the interposer, and go up to the destination chiplet through TSVs [9]. Thus, two extra vertical-</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>communication hops are required compared to a hypothetical monolithic EHP. To assess the performance impact of these overheads, we compare a monolithic EHP design against our proposed multi-chiplet design. The findings below show only a small performance impact, suggesting that a chiplet-based design is feasible.</p> <p>Finding 1: The out-of-chiplet traffic dominates the total traffic, ranging from 60-95% across kernels representing a wide range of behavior. This traffic includes traffic between GPU CUs on different chiplets as well as between the CPU cores and the GPU CUs. Given that there are eight GPU chiplets on the EHP, the remote-traffic rate demonstrates a fairly even distribution of accesses across chiplets. Finding 2: The performance impact of the large out-of-chiplet traffic is relatively small. The largest performance degradation compared to the monolithic EHP is 13%, while some application kernels, such as SNAP, have a negligible impact, as shown in Fig. 7.</p> <p>Takeaway: The small performance impact despite the large out-of-chiplet traffic is due to a variety of reasons. First, the GPU’s massive parallelism is effective at latency hiding. Kernels with abundant parallelism tend to be able to hide much of the latency to other chiplets. Second, some applications (especially those with cache-friendly access patterns) do not stress the memory system and are not sensitive to a few cycles of additional memory latency. Thus, our chiplet organization makes a compelling trade-off between longer network latency and improvement in die yield and cost.”)</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
1[D]: wherein the passive crosslink is dedicated for inter-chiplet communications.	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p><i>See evidence cited for ’803 patent limitations 1[A]-1[C].</i></p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
2. The system of claim 1, wherein the passive crosslink comprises a passive interposer die.	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p><i>See evidence cited for ’803 patent limitation 1[D].</i></p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
3[A]: The system of claim 1, wherein the first GPU chiplet comprises a first PHY region including conductor structures for chiplet-to-chiplet communications,	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>Vijayaraghavan at 86 (“While ongoing hardware and software research continues to further improve specific aspects of the ENA, this paper focuses on the overall physical node organization that integrates these aggressive technologies. We describe the proposed ENA concept and the rationale behind the different design decisions. We then present an initial quantitative evaluation of the ENA and discuss remaining challenges and opportunities for further research.”)</p> <p><i>See also</i> evidence cited for ’803 patent claims 1-2.</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
3[B]: and further wherein the second GPU chiplet comprises a second PHY region including conductor structures for chiplet-to-chiplet communications.	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p><i>See</i> evidence cited for ’803 patent limitation 3[A].</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
7. The system of claim 1, wherein the passive crosslink communicably couples all GPU chiplets in the GPU chiplet array.	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p><i>See, e.g.,</i></p> <p><i>See</i> evidence cited for ’803 patent limitation 1[D].</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
<p>8[A]: The system of claim 1, further comprising: a first cache memory hierarchy at the first GPU chiplet,</p>	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p>Vijayaraghavan at Fig. 1.</p>

U.S. Patent No. 11,841,803

Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)

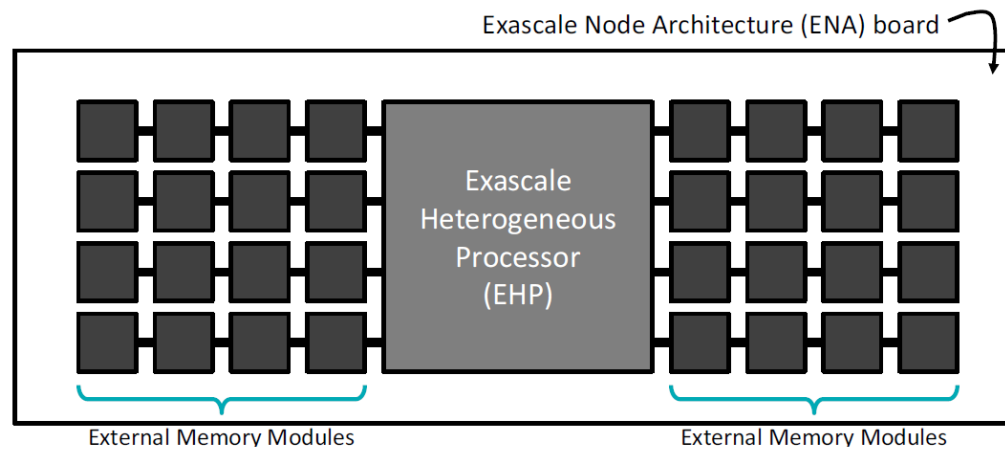
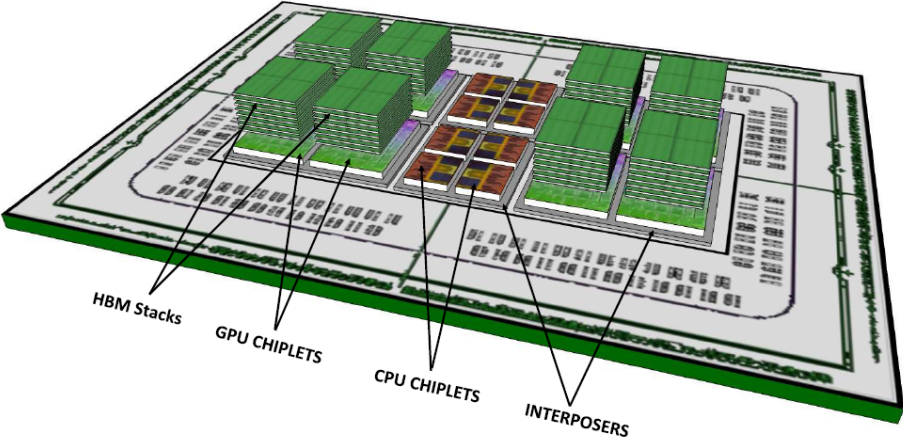


Figure 1. Exascale Node Architecture (ENA)

Vijayaraghavan at Fig. 2.

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	 <p data-bbox="926 824 1640 862">Figure 2. Exascale Heterogeneous Processor (EHP)</p> <p data-bbox="680 906 1003 943">Vijayaraghavan at Fig. 3.</p>

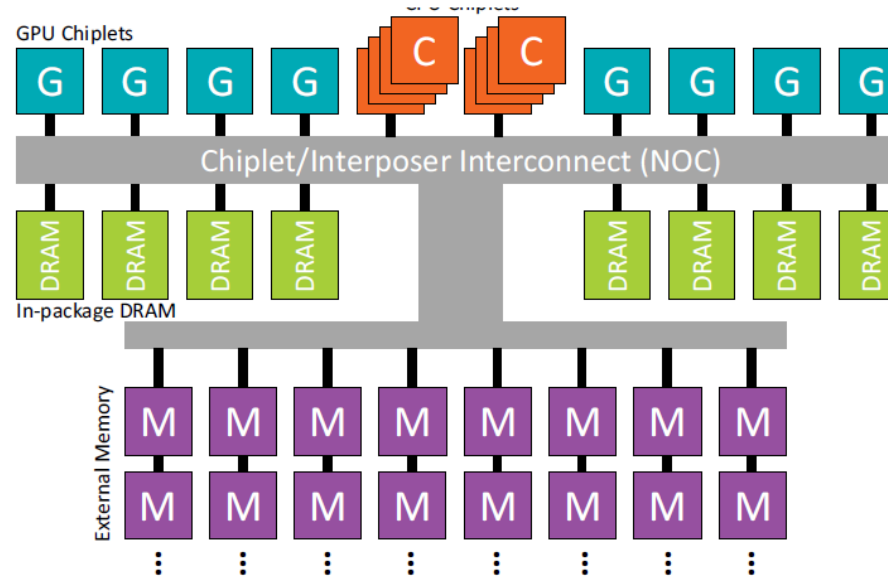


Figure 3. Block diagram of the ENA memory system

Vijayaraghavan at Fig. 8.

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<div><div><div><div><div></div><div></div><div></div><div></div><div></div></div><div>Perf. normalized to perf. with no misses</div><div><div><div></div><div></div><div></div><div></div><div></div></div><div>100%</div><div>80%</div><div>60%</div><div>40%</div><div>20%</div><div>0%</div></div><div><div>LULESH</div><div>MiniAMR</div><div>XSBench</div><div>SNAP</div><div>CoMD</div><div>CoMD-LJ</div><div>HPGMG</div><div>MaxFlops</div><div>Mean</div></div><div><div>20%</div><div>40%</div><div>60%</div><div>80%</div><div>100%</div></div></div><div>Figure 8. Performance impact of miss rates in the in-package DRAM</div><div>Vijayaraghavan at Fig. 11.</div></div></div>

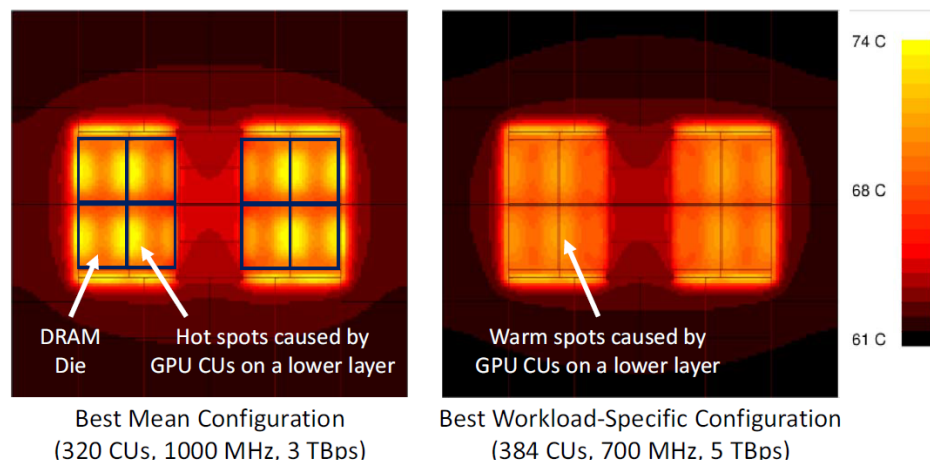


Figure 11. Heat map of the bottom-most in-package 3D-DRAM die for SNAP

Vijayaraghavan at 86 (“The EHP is an accelerated processing unit (APU) consisting of a balanced mix of CPU and GPU compute resources integrated with in-package 3D DRAM. The overall structure makes use of a modular “chiplet” design, with the chiplets 3D-stacked on other “active interposer” chips. The aggressive use of advanced packaging technologies enables a large amount of computational and memory resources to be located in a single package.”)

Vijayaraghavan at 86 (“The Exascale Node Architecture (ENA) is the building block for a single compute node (shown in Fig. 1). The ENA’s computational capabilities are provided by a high-performance accelerated processing unit called the Exascale Heterogeneous Processor (EHP), which we will later describe in greater detail. The EHP makes use of high-bandwidth, in-package DRAM, but it is also coupled with a network of external-memory devices to provide large per-node memory capacity to handle very large application problem sizes. Given the performance goal of 1 exaflop and a power budget of 20MW for a 100,000-node exascale

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>machine, we need to architect the ENA node to provide 10 teraflops of performance in a 200W power envelope. Below, we first describe the EHP and its constituent technologies, and then we discuss the memory system for the ENA.”)</p> <p>Vijayaraghavan at 86 (“Fig. 2 shows a conceptual rendering of the EHP. The EHP consists of two primary types of resources. In the center of the EHP are two CPU clusters, each consisting of four multi-core CPU chiplets stacked on an active interposer base die. On either side of the CPU clusters are a total of four GPU clusters, each consisting of two GPU chiplets on a respective active interposer. Upon each GPU chiplet is a 3D stack of DRAM (e.g., some future generation of JEDEC high-bandwidth memory (HBM) [12]). The DRAM is directly stacked on the GPU chiplets to maximize bandwidth (the GPUs are expected to provide the peak computational throughput) while minimizing memory-related data movement energy and total package footprint. CPU computations tend to be more latency sensitive, and so the central placement of the CPU cores reduces NUMA-like effects by keeping the CPU-to-DRAM distance relatively uniform. The interposers underneath the chiplets provide the interconnection network between the chiplets (i.e., network on chip (NOC)) along with other common system functions (e.g., external I/O interfaces, power distribution, system management). Interposers maintain high-bandwidth connectivity among themselves by utilizing wide, short-distance, point-to-point paths.”)</p> <p>Vijayaraghavan at 87 (“The CPU and GPU must be architected in a cohesive manner from an energy efficiency as well as programmability standpoint. To this end, Heterogeneous System Architecture (HSA) compatibility is one of the major design goals of the APU. HSA provides a system architecture where all computing elements (CPU, GPU, and possibly other accelerators) share a unified coherent virtual address space. This enables efficient programming and computation via several mechanisms like free exchange of pointers by both CPU and GPU code, eliminating expensive data copy operations, transparent management of CPU and GPU caches via cache coherence, task offloads by both CPU and GPU to each other or other CPU/GPU units [13], and efficient synchronization mechanisms. These features are</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>supported by AMD’s Radeon Open Compute platform (ROCm) to improve the programmability of such heterogeneous systems.”)</p> <p>Vijayaraghavan at 87 (“While APUs with HSA compatibility have been available for some time, the existing designs have been targeted more for consumer platforms rather than HPC systems. To this end, we are actively researching several mechanisms to provide heterogeneous computing for HPC. We have created novel mechanisms like the QuickRelease synchronization mechanism [14] and heterogeneous race free memory mod-els (HRF) [15]–[17] to reduce synchronization overhead between GPU threads, and heterogeneous system coherence (HSC) [18] to transparently manage coherence between CPU and GPU caches. These make the APU easier to program and reason about for the application programmers, enabling more codes to be more easily ported to take advantage of the EHP’s resources.”)</p> <p>Vijayaraghavan at 87 (“2) Modular Chiplet Design: The performance require-ments of the exascale node require a large amount of compute and memory to be integrated into a single package. Rather than build a single, monolithic system on chip (SOC), we propose to leverage advanced die-stacking technologies to decompose the EHP into smaller components consisting of active interposers and chiplets. Each chiplet houses either multiple GPU compute units or CPU cores. The chiplet approach differs from conventional multi-chip module (MCM) designs in that each individual chiplet is not a complete SOC. For example, the CPU chiplet contains CPU cores and caches, but lacks memory interfaces and external I/O.1 There are multiple benefits to this decompositional approach to SOC construction:</p> <p>Die Yield: Building a single monolithic SOC that provides the equivalent capabilities of what we propose for the EHP would result in an impractically large chip with prohibitive costs. Smaller chiplets have higher yield rates due to their size, and when combined with known-good-die (KGD) testing techniques, can be assembled into larger systems at reasonable cost. This approach has already started garnering interest in both academia [9] and industry [8], [19]. Process Optimization: A monolithic SOC imposes a single process technology choice on all components in the system. With chiplets and interposers, each discrete piece of silicon can be optimized for its own functions. For example, the CPU chiplets can use performance-</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>optimized devices and metal layers, while the GPU chiplets use density-optimized devices and metal. The interposer layers can use a more mature (i.e., less expensive) process technology node as the I/O components likely do not need transistors in the cutting-edge technology node nor as many metal routing layers as the compute chiplets.</p> <p>Re-usability: A single, large HPC-optimized APU would be great for HPC markets, but may be less appropriate for others. The decomposition of the EHP into smaller pieces enables silicon-level reuse. For example, one or more of the CPU clusters could be packaged together to create a conventional CPU-only server processor.”)</p> <p>Vijayaraghavan at 88 (“In addition, due to vertical integration of the DRAM and the GPU compute units, thermal feasibility is an important design consideration. In Section V-D, we study the thermal feasibility of our envisioned approach.”)</p> <p>Vijayaraghavan at 88 (“The current state of the art for handling transient faults is error-correcting codes (ECC). ECC can be relatively easily deployed for regular structures like DRAM and SRAM arrays, but it still comes with area costs that are more challenging in our space-constrained EHP design. Also, some components may not normally have HPC-class RAS support by default. For example, current GPUs are still over-whelmingly used for graphics and visualization applications where a few errors result in unnoticed aberrations of a couple pixels, and so heavy-duty RAS support is unwarranted. To avoid burdening the GPU with excessive RAS features, which would reduce the re-usability of GPU chiplets in other markets, we have explored software-based solutions such as redundant multi-threading (RMT) [25]. RMT takes advantage of the fact that the GPU is not fully utilized for some workloads, and uses the otherwise idle GPU resources to redundantly perform computations and detect errors (note that this paper does not include quantitative evaluation of RMT).”)</p> <p>Vijayaraghavan at 88 (“The exascale memory requirements are very aggressive in terms of total memory capacity, the desired memory bandwidth, and the energy-related costs. Ideally, the entire system would make use of in-package 3D-integrated DRAM, as that provides the highest performance and capacity at the lowest energy cost. Unfortunately, our projections for</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>the maximum amount of DRAM that we can integrate into a package fall short of the per-node memory targets. We instead make use of a heterogeneous memory architecture consisting of multiple levels of memory [26], [27]. Fig. 3 shows a conceptual view of the EHP’s memory hierarchy.</p> <p>1) Integrated 3D Memory: Many of the scientific work-loads that we have analyzed demand more bandwidth than what can be provided by conventional DRAM outside of the processor package. The stated exascale targets [28] for memory bandwidth and energy efficiency are incredibly challenging for off-package memory solutions. Hence, we integrate 3D-stacked DRAM into the EHP package.</p> <p>To estimate the memory capabilities of 3D DRAM in the exascale timeframe, we started with current JEDEC High-Bandwidth Memory (HBM) [12] and project forward. First-generation HBM provides 1GB capacity at 128 GB/s of bandwidth per stack, and second-generation HBM is projected to yield 8GB at 256 GB/s per stack. By the 2022-2023 time-frame, we estimate that HBM (or a similar equivalent) should have advanced at least two more generations. Assuming a doubling of capacity for each generation, we project that 3D DRAM in the exascale time-frame would provide 32GB per stack. As the interface speed for HBM2 is already 2 Gbps, we do not expect the same per-generation doubling of bandwidth; we instead assume only a single doubling (through wider interfaces or faster speeds). The EHP makes use of a total of eight 3D DRAM stacks (one per GPU chiplet), so the total 3D DRAM capacity is 256GB with a total aggregate bandwidth of 4 TB/s, meeting the exascale memory bandwidth target.”)</p> <p>Vijayaraghavan at 89 (“2) External-memory Network: The exascale target for per-node memory capacity is at least 1TB, which exceeds our in-package projections by a factor of four. As such, the ENA must augment the in-package memory with additional external memory. The ENA makes use of Memory Networks that consist of multiple memory modules interconnected with point-to-point links (as opposed to bus-based topologies for DDR) [29], [30]. A current example of this kind of memory approach is the Hybrid Memory Cube, which</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>uses 3D-stacked DRAM inside each module, and the modules communicate with high-speed serial interfaces [31].</p> <p>The envisioned EHP package provides eight external-memory interfaces, each of which connects to multiple memory packages. Fig. 3 shows each interface supporting a disjoint set of memory devices. A simple chain topology is shown, although support for other topologies is also expected. We assume that the memory interfaces are address-interleaved in some fashion such that there is never a need for one memory interface to send a request to a memory package connected to a different interface (similar to how multiple DDR memory channels are managed today). Optional links (not shown) could be used to cross-connect chains for redundancy purposes, which allow access to memory devices in the event of link failures. Depending on the exact needs of the supercomputer customer, the external-memory network could consist of a mix of both DRAM and non-volatile memory (NVM) devices. NVM provides higher densities and therefore could be useful in scenarios where very large problem sizes are important, or they could be used to reduce the total ENA board-level component count by meeting a given capacity target with fewer higher-capacity packages. The trade-off is that the NVMs are typically slower, consume more dynamic power (especially for writes), and may suffer from write-endurance issues that could impact the system’s MTTF.”)</p> <p>Vijayaraghavan at 89 (“3) Memory-system Management: The ENA’s memory architecture provides (at least) two different levels of memory. There is the in-package 3D DRAM, and there is also the external-memory network (which could provide more than one additional level of memory if both DRAM and NVM modules are used). Such a heterogeneous memory system raises the question of how the different memory resources should be managed. While the ENA is designed to support a configurable memory system with multiple modes of operation, we envision the primary mode to be software-controlled. The ENA’s physical memory address space is interleaved across the different memory resources (the granularity and overall mapping of which would be controlled by the system software). The operating system (OS) would then provide capabilities to monitor and migrate memory between memory resources in an attempt to maximize the fraction of memory requests that can be serviced by</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>the in-package DRAM [26], [27], [32]. The system would also provide user-level APIs so that programmers could explicitly allocate data into specific types of memory [26], [33].”)</p> <p>Vijayaraghavan at 89 (“There has been a significant amount of work in recent years exploring how to make use of 3D DRAM as a large, hardware-managed cache [34], [35]. We envision the ENA supporting 3D-DRAM as a cache, but for HPC applications, this will often not be desirable. For an example ENA with 256GB of in-package DRAM and 1TB of external memory, using the in-package DRAM as a cache would sacrifice 20% of the system’s total addressable memory capacity, which in turn reduces the problem sizes that can be solved. However, if a problem does not require the full memory capacity of the ENA, enabling the hardware-cache mode could provide a performance uplift without any application modifications.”)</p> <p>Vijayaraghavan at 91 (“The benefits of a chiplet-based design were covered in Section II, but a multi-chip organization imposes additional latency for inter-chiplet communication (whether for coherence or main-memory access). Messages to a remote chiplet traverse from a source chiplet to the lower interposer layer through through-silicon vias (TSVs), go across the interposer, and go up to the destination chiplet through TSVs [9]. Thus, two extra vertical-communication hops are required compared to a hypothetical monolithic EHP. To assess the performance impact of these overheads, we compare a monolithic EHP design against our proposed multi-chiplet design. The findings below show only a small performance impact, suggesting that a chiplet-based design is feasible.</p> <p>Finding 1: The out-of-chiplet traffic dominates the total traffic, ranging from 60-95% across kernels representing a wide range of behavior. This traffic includes traffic between GPU CUs on different chiplets as well as between the CPU cores and the GPU CUs. Given that there are eight GPU chiplets on the EHP, the remote-traffic rate demonstrates a fairly even distribution of accesses across chiplets. Finding 2: The performance impact of the large out-of-chiplet traffic is relatively small. The largest performance degradation compared to the monolithic</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>EHP is 13%, while some application kernels, such as SNAP, have a negligible impact, as shown in Fig. 7.</p> <p>Takeaway: The small performance impact despite the large out-of-chiplet traffic is due to a variety of reasons. First, the GPU’s massive parallelism is effective at latency hiding. Kernels with abundant parallelism tend to be able to hide much of the latency to other chiplets. Second, some applications (especially those with cache-friendly access patterns) do not stress the memory system and are not sensitive to a few cycles of additional memory latency. Thus, our chiplet organization makes a compelling trade-off between longer network latency and improvement in die yield and cost.”)</p> <p>Vijayaraghavan at 91, n.5 (“Here, we use the term “miss” to describe a memory access to the external memory. We are not using the in-package DRAM as a hardware-controlled cache.”)</p> <p>Vijayaraghavan at 91-92 (“Our design choice of integrating high-bandwidth 3D memory directly into the EHP package is driven by the exascale high-bandwidth requirement. Due to the limited capacity of the in-package 3D-DRAM, some data may reside in the external memory instead. Requests to addresses not in the first-level in-package memory result in costlier accesses to the external memory through a lower-bandwidth, off-package interface that can degrade performance. To highlight the importance of provisioning high-bandwidth memory closer to the compute and efficient data management for the multi-level memory, we artificially vary the fraction of requests serviced by the in-package DRAM.</p> <p>Finding: Fig. 8 plots the performance reduction due to in-package 3D-DRAM misses⁵ normalized to no misses (i.e., all requests serviced by in-package DRAM). As expected, the degree of reduction depends on the application characteristics. The compute-intensive application MaxFlops retains the same performance level regardless of the in-package memory miss rate due to very infrequent memory accesses. In contrast, other applications observe degradations from 7% to as much as 75% as more misses in the in-package memory cause more external-memory traffic. LULESH, a memory-intensive kernel, has an interesting behavior as it shows lower sensitivity to bandwidth compared to a compute-intensive kernel</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>like CoMD. This can be attributed to the LULESH’s irregular access patterns that make it more sensitive to memory latency than bandwidth. Separate analysis of the applications indicate that 46% to 89% of memory traffic may still need to access off-package memory, mainly due to large data footprints. These rates are derived from previous memory management techniques [27], and will likely be improved as research discovers better multi-level memory management algorithms.</p> <p>Takeaway: The in-package memory is critical for many bandwidth-intensive kernels. Efficient data management via software and/or hardware techniques will continue to be important to ensure that as many requests as possible can be serviced from the in-package memory for both performance and energy concerns.”)</p> <p>Vijayaraghavan at 92-93 (“We simulate the EHP package alone as it has higher power density than the external memory. We used the HotSpot [47] model from the University of Virginia and calibrated it against an internal product model.</p> <p>We use peak DRAM temperature as a metric in this analysis because DRAM is less tolerant to temperature than processors. DRAMs must stay below 85°C to avoid increasing the refresh rate [48]. We assume a high-end air-cooling solution [49] and 50°C ambient temperature in a 2U-server chassis [50].</p> <p>Finding 1: As shown in Fig. 10, EHP’s in-package DRAMs stay below the 85°C limit for all kernels with both the baseline EHP configuration and the best per-kernel configuration⁷. Despite the high CU power, MaxFlops does not stress the memory temperature because of almost no memory accesses. On the other hand, CoMD-LJ approaches the thermal limit due to its modest memory accesses combined with high compute intensity that raises the CU dynamic power in the kernel-optimized EHP configuration. Note that these results do not include any power optimizations discussed later in Section V-E.</p> <p>Finding 2: For most kernels, the best-per-application configuration leads to better performance, higher power, and higher temperature, compared to the best-mean configuration. However, for SNAP and HPGMG, this does not hold. Though their performance and power trends are the same as the other workloads, the temperature with the best-per-application configuration is</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>lower than the mean configuration. This is because more power is shifted from the high-power-density CUs to the lower-power-density in-package DRAM. Fig. 11 shows the temperature difference in the bottom-most in-package DRAM die, with the best-mean configuration versus the best kernel-optimized configuration for SNAP. Takeaway: Our use of aggressive die stacking should be thermally feasible even with air cooling. However, more advanced cooling solutions may become necessary as the hit rate of the in-package DRAM improves, more power from the external memory is shifted to the EHP, or if a design point uses a greater per-node power budget.”</p> <p>Vijayaraghavan at 93-94 (“Below, we briefly describe the techniques we have explored and their estimated power benefits.</p> <p>Near-Threshold Computing (NTC): NTC enables operating near the threshold voltage as long as the underlying circuits are variability tolerant and resilient against errors. Recent advances in our NTC research allows operating the CUs near the threshold voltage at as high as 1 GHz while still obtaining 14% power savings on average across the workloads examined. Given the lower stability of SRAM cells, this paper does not apply NTC to the memory circuits, which would be a potential future research direction. Asynchronous Compute Units: Most computing devices have synchronous implementations, in which data are stored in registers at the edge of a clock. Although synchronous designs have lower design complexity than asynchronous designs, the former typically has higher dynamic power because of high switching activities and clock-tree complexity. We estimate average system-power savings of 4.3% by carefully applying asynchronous-circuit techniques to only the ALUs and crossbars of the GPU SIMD units. Asynchronous Routers: When we extend asynchronous circuits to interconnect routers, our study indicates an average power savings of 3.0% can be attained.</p> <p>Low-Power Links: Operating interconnect links in a low-power mode can lead to average power savings of 1.6%. DRAM Traffic Compression: Our research shows that a substantial portion of the EHP interconnect power is spent on the long-distance interconnects between the last-level cache (LLC) and the in-package memory. One way to reduce the data-movement power is to apply data compression to the network messages. We estimate average power</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>savings of 1.7% from compression. LULESH benefits the most from this optimization, given its high memory intensity.</p> <p>Finding 1: Fig. 12 shows the power saved when each of the above techniques is deployed individually and in combination. The power savings range from 13% to 27% when all techniques are deployed together. Note that the baseline power without optimizations already includes power savings from DVFS.</p> <p>Finding 2: Reduction in power can lead to performance improvements by allowing a higher-performing hardware operating point while staying under the power budget. Fig. 13 shows the improvement in performance per Watt when comparing the new best-mean configuration with optimizations (288 CUs, 1100 MHz, 3 TB/s) to the one before optimizations (320 CUs, 1000 MHz, 3 TB/s). The trend in the efficiency gains across the kernels as seen in Fig. 13 is not the same as the power-savings trend in Fig. 12 because the change in the best-mean configuration (lower CU count and higher frequency) affects different kernels differently. Takeaway: Optimizing power consumption throughout various node components can improve the node energy efficiency substantially, and may also be necessary if the target applications exercise different parts of the node differently.”)</p> <p>Vijayaraghavan at 95 (“High-density and Low-energy Memory: As discussed in Section V-C, the higher read and write access energy of NVM (compared to DRAM) leads to a significant power overhead for memory-intensive applications. However, NVM’s very low leakage power coupled with high memory densities is desirable to reduce the static power of systems with large memory capacities. Further advances in memory technologies are required to improve not only the density but also the energy efficiency. Additionally, there are ample research opportunities for efficient management of heterogeneous memories and multi-level memories to ensure that more frequently accessed data reside in a faster, lower-access-energy memory.</p> <p>Resiliency Solutions: As discussed in Section II, supporting resiliency without significant area and performance overheads is critical given the aggressive exascale performance goals, the low mean-time-to-failure requirement, and the increased rate of silent errors that future technology nodes may cause. In addition, some of the power-saving techniques this paper</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>evaluated substantially reduce the operating voltage, potentially increasing error rates. Further research is needed to explore techniques that have low performance and area overheads while providing robust resiliency.</p> <p>Power Optimizations: This paper considered only a handful of promising power-optimization techniques. Other types of optimizations are also necessary to target the remaining components with high power consumption, such as memory and SerDes links. A management technique must be developed to orchestrate a variety of power optimizations, triggering only those deemed appropriate for a given application phase with negligible performance impact.”)</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
<p>8[B]: wherein a first level of the first cache memory hierarchy is coherent within the first GPU chiplet; and</p>	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p>Vijayaraghavan at 87 (“The CPU and GPU must be architected in a cohesive manner from an energy efficiency as well as programmability standpoint. To this end, Heterogeneous System Architecture (HSA) compatibility is one of the major design goals of the APU. HSA provides a system architecture where all computing elements (CPU, GPU, and possibly other accelerators) share a unified coherent virtual address space. This enables efficient programming and computation via several mechanisms like free exchange of pointers by both CPU and GPU code, eliminating expensive data copy operations, transparent management of CPU and GPU caches via cache coherence, task offloads by both CPU and GPU to each other or other CPU/GPU units [13], and efficient synchronization mechanisms. These features are supported by AMD’s Radeon Open Compute platform (ROCm) to improve the programmability of such heterogeneous systems.”)</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>Vijayaraghavan at 87 (“While APUs with HSA compatibility have been available for some time, the existing designs have been targeted more for consumer platforms rather than HPC systems. To this end, we are actively researching several mechanisms to provide heterogeneous computing for HPC. We have created novel mechanisms like the QuickRelease synchronization mechanism [14] and heterogeneous race free memory mod-els (HRF) [15]–[17] to reduce synchronization overhead between GPU threads, and heterogeneous system coherence (HSC) [18] to transparently manage coherence between CPU and GPU caches. These make the APU easier to program and reason about for the application programmers, enabling more codes to be more easily ported to take advantage of the EHP’s resources.”)</p> <p>Vijayaraghavan at 91 (“The benefits of a chiplet-based design were covered in Section II, but a multi-chip organization imposes additional latency for inter-chiplet communication (whether for coherence or main-memory access). Messages to a remote chiplet traverse from a source chiplet to the lower interposer layer through through-silicon vias (TSVs), go across the interposer, and go up to the destination chiplet through TSVs [9]. Thus, two extra vertical-communication hops are required compared to a hypothetical monolithic EHP. To assess the performance impact of these overheads, we compare a monolithic EHP design against our proposed multi-chiplet design. The findings below show only a small performance impact, suggesting that a chiplet-based design is feasible.</p> <p>Finding 1: The out-of-chiplet traffic dominates the total traffic, ranging from 60-95% across kernels representing a wide range of behavior. This traffic includes traffic between GPU CUs on different chiplets as well as between the CPU cores and the GPU CUs. Given that there are eight GPU chiplets on the EHP, the remote-traffic rate demonstrates a fairly even distribution of accesses across chiplets. Finding 2: The performance impact of the large out-of-chiplet traffic is relatively small. The largest performance degradation compared to the monolithic EHP is 13%, while some application kernels, such as SNAP, have a negligible impact, as shown in Fig. 7.</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>Takeaway: The small performance impact despite the large out-of-chiplet traffic is due to a variety of reasons. First, the GPU’s massive parallelism is effective at latency hiding. Kernels with abundant parallelism tend to be able to hide much of the latency to other chiplets. Second, some applications (especially those with cache-friendly access patterns) do not stress the memory system and are not sensitive to a few cycles of additional memory latency. Thus, our chiplet organization makes a compelling trade-off between longer network latency and improvement in die yield and cost.”)</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
8[C]: a second cache memory hierarchy at the second GPU chiplet,	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p><i>See</i> evidence cited for ’803 patent limitation 8[A].</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
8[D]: wherein a first level of the second cache memory hierarchy is coherent within the second GPU chiplet.	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p><i>See</i> evidence cited for ’803 patent limitation 8[B].</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
<p>9[A]: The system of claim 8, further comprising: a unified cache memory including both a last level of the first cache memory hierarchy and a last level of the second cache memory hierarchy,</p>	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p><i>See</i> evidence cited for ’803 patent Claim 8.</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>
<p>9[B]: wherein the unified cache memory is coherent across all GPU chiplets of the GPU chiplet array.</p>	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p><i>See</i> evidence cited for ’803 patent Claim 8.</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.
<p>10.The system of claim 1, further comprising: a plurality of conductive pillars coupling a circuit board substrate to a first non-PHY region of the first GPU chiplet and a second non-PHY region of the second GPU chiplet.</p>	<p>Including at least under Complainant’s apparent infringement theory, Vijayaraghavan discloses and/or renders obvious this limitation.</p> <p><i>See, e.g.,</i></p> <p>Vijayaraghavan at Fig.1.</p> <div data-bbox="777 662 1785 1105" data-label="Diagram"> </div> <p>Figure 1. Exascale Node Architecture (ENA)</p> <p>Vijayaraghavan at 89 (“2) External-memory Network: The exascale target for per-node memory capacity is at least 1TB, which exceeds our in-package projections by a factor of four. As such, the ENA must augment the in-package memory with additional external memory. The ENA makes use of Memory Networks that consist of multiple memory modules interconnected with point-to-point links (as opposed to bus-based topologies for DDR) [29], [30]. A current example of this kind of memory approach is the Hybrid Memory Cube, which</p>

U.S. Patent No. 11,841,803	Thiruvengadam Vijayaraghavan et. al, Design and Analysis of an APU for Exascale Computing, 2017 IEEE International Symposium on High Performance Computer Architecture (“Vijayaraghavan”)
	<p>uses 3D-stacked DRAM inside each module, and the modules communicate with high-speed serial interfaces [31].</p> <p>The envisioned EHP package provides eight external-memory interfaces, each of which connects to multiple memory packages. Fig. 3 shows each interface supporting a disjoint set of memory devices. A simple chain topology is shown, although support for other topologies is also expected. We assume that the memory interfaces are address-interleaved in some fashion such that there is never a need for one memory interface to send a request to a memory package connected to a different interface (similar to how multiple DDR memory channels are managed today). Optional links (not shown) could be used to cross-connect chains for redundancy purposes, which allow access to memory devices in the event of link failures. Depending on the exact needs of the supercomputer customer, the external-memory network could consist of a mix of both DRAM and non-volatile memory (NVM) devices. NVM provides higher densities and therefore could be useful in scenarios where very large problem sizes are important, or they could be used to reduce the total ENA board-level component count by meeting a given capacity target with fewer higher-capacity packages. The trade-off is that the NVMs are typically slower, consume more dynamic power (especially for writes), and may suffer from write-endurance issues that could impact the system’s MTTF.”)</p> <p>This limitation also would have been obvious to a person of ordinary skill in the art based on (1) Vijayaraghavan alone; (2) Vijayaraghavan in combination with the general knowledge of a person of ordinary skill in the art; and/or (3) Vijayaraghavan in combination with the teachings with respect to this claim element as detailed in NVIDIA’s other invalidity charts and cover pleading.</p>