Extending the Life of Old Systems with More Memory

Euna Kim[*]
Georgia Institute of Technology
Atlanta, Georgia, USA
euna.kim@gatech.edu

Andrei Bersatti[*]
Georgia Institute of Technology
Atlanta, Georgia, USA
abersatti@gatech.edu

Hyesoon Kim
Georgia Institute of Technology
Atlanta, Georgia, USA
hyesoon@cc.gatech.edu

ABSTRACT

Extending the lifetime of computing systems is definitely one of the ways to reduce CO_2. In this paper, we explore and suggest the options of extending memory in old systems by utilizing CXL technologies.

CCS CONCEPTS

• **Computer systems organization → Processors and memory architectures**; • **Hardware → Impact on the environment**.

KEYWORDS

CO_2, green computing, memory, reuse, CXL

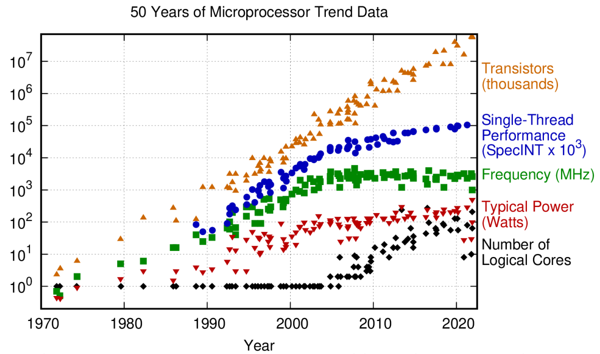
1 INTRODUCTION

As we approach the end of Dennard Scaling, the performance increases of single-thread processor performance have been slowed down. As Figure 1 shows, for the last decade (since 2010), typical power consumption and the frequency of the processor has plateaued.

However, modern workloads face a dual challenge. One challenge is that the size of the data to process is increasing, as shown in [17][15]. However, the attained results would not have been possible without the huge amounts of processed data. Importantly, the sizes of the graphs are increasing at a dizzying rate [8][1]. This growth in the amount of data is an enabler of new applications, but it is also a bottleneck in their progress because of exponentially growing demand for memory and the poor and slowing scaling properties of DRAM [22].

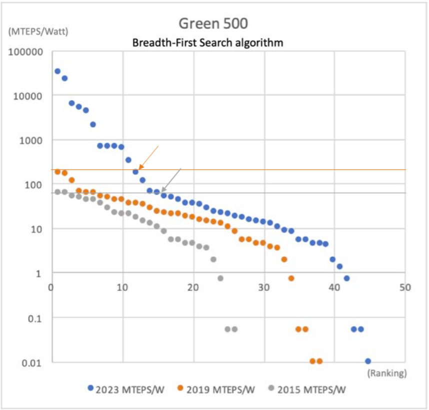
We have observed that the power efficiency of high performance computing systems has increased continuously despite the observation in Figure 1. Figure 2, the Green500 ranking based on the ratio of Millions of TeraFloatingPoint Operations Per Second (MTEPS) to Watt (MTEPS/Watt) is presented for three points in time: the current year (2023), as well as 4 and 8 years ago. Notably, it has been observed that the top 10 systems in 2019 maintain their dominance, remaining at the top 20% of the 2023 ranking. Similarly, the leading 2015 systems retain their importance, occupying the upper 40% of the 2023 ranking. This intriguing pattern signifies that even after a decade, the supercomputer systems that were at the forefront in earlier years continue to hold a position of median performance in 2023.

The other challenge is that the world is warming up, and this is in large part a result of Greenhouse Gas (GHG) emissions. This climate change leads to severe and devastating weather-related events [7]. To respond to this problem, we propose to leverage the potential to reuse technologies as part of the US Environmental Protection Agency's (EPA) "reduce, reuse, and recycle" mantra and strategy [3, 13]. We propose to increase the memory in the high-performance computing systems to extend the life time of computing systems. In particular, we explore the option of using the recent CXL technology [21].



Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2021 by K. Rupp

Figure 1: 50 Years of Microprocessor Trend Data. Image Source: [20]. This chart is provided under the permissive 'Creative Commons Attribution 4.0 International Public License.'



Green500 website data available from 2013 - 2023 (https://graph500.org)

Figure 2: Green500 $\frac{MTEPS}{Watt}$ ranking.

2 BACKGROUND OF CXL TECHNOLOGY

CXL is a high-speed CPU interconnect technology standard developed and maintained by the Compute Express Link consortium. We provide the following background as a brief summary of the more in-depth material found in [2, 9, 19, 21].

[*] Both authors contributed equally to this research.

The notable characteristic of CXL is that its Flex Bus leverages PCIe PHY for its physical layer. What this means is that any computer or device that supports compatible PCIe versions can support CXL.

CXL structure: The CXL Link layer is responsible for CRC checks and flow control through a credit system. The CXL transaction layer is responsible for Flit packing and unpacking and for processing and storing the transactions in queues. CXL is comprised of three sub-protocols, namely CXL.io, CXL.cache, and CXL.memory, which have the function of supporting I/O, cache coherency, and memory semantics, respectively.

CXL versions: CXL 1.0 (released March 2019) and CXL 1.1 (released September 2019) introduced the suit of sub-protocols that constitute the CXL standard. CXL maintains coherency in memory spaces between the CPU and any attached device. CXL 1.1 defined three types of devices based on differing use cases: Type 1, Type 2, and Type 3. Type 1 devices leverage CXL.cache and are caching devices or accelerators such as smart NICs. Type 2 devices are accelerators with memory such as GPGPUs with local memory that is mapped to a cacheable system memory and leverage CXL.cache and CXL.memory. Type 3 devices leverage CXL.memory and have the purpose of serving as memory buffers by expanding memory bandwidth or capacity and can be used to connect together different memory types. In addition, all three types make use of CXL.io. CXL 2.0 (released November 2020) enhanced the protocol by introducing support for, among the most relevant, switch capability and memory pooling. CXL 3.0 (released Q3 2022) doubles the bandwidth and introduces multiple level switching, it also enables non-tree topologies and peer-to-peer communication, and it introduces Global Fabric Attached Memory (GFAM) devices which disaggregates the memory from the processing unit and implements a memory pool.

3 MACHINE EXTENSIONS WITH CXL MEMORY

We begin the following analysis assuming that the trivial solution of simply adding more memory in the high-performance computing systems without any further change is not an option as we assume that the current motherboard is maxed out on available memory slots and capacity. Hence, we try to answer the following simple question: *Can we increase the memory of old systems using CXL technologies, since CXL is based on PCIe?*

3.1 About PCIe Versions

Unfortunately, the earlier versions of CXL require at least PCIe 5.0 [2], which debuted in 2019. It is important to note that previous versions of CXL, namely CXL 1.1 and CXL 2.0 are compatible with 32 GT/s PCIe 5.0 and CXL 3.0 is compatible with 64 GT/s PCIe 6.0. Notably, all CXL versions are backward compatible. AMD Epyc (2022) and Intel Xeon (2023) servers started supporting PCIe 5.0 starting with their 4th generations. AMD Epyc supported PCIe 4.0 starting with its 2nd (2019) and 3rd generation, and Intel Xeon supported PCIe 4.0 starting with its 3rd Generation (2020) [4, 14]. Unfortunately, PCIe supports are also part of CPU specification as I/O design is a part of the protocols. Consequently, to enable the utilization of older systems with CXL, it becomes necessary to replace both the microprocessor and the chipset. This implies that merely swapping out motherboards for the purpose of expanding memory is not a feasible solution for upgrading legacy systems. An alternative would be to be able to use PCIe 4.0 systems with CXL. The downside of this would be that PCIe 4.0 only has half as much bandwidth as PCIe 5.0, which means that the memory bandwidth with CXL would be half as well. However, CXL with PCIe 5.0 is expected to provide up to 64 GB/s in each direction over a 16-lane link and the half of that is 32 GB/s in each direction over a 16-lane link. Compared to the bandwidth of SSD (300-600 MB/s), the performance with CXL with PCIe 4.0 would still provide higher performance than using storage as an extension of memory systems.

3.2 Adding Memory

There are two basic ways in which memory can be expanded to a system by leveraging CXL technologies. The first option explored is to add CXL Type 3 memory expansion modules such as [23]. This option enables leveraging CXL 2.0 to introduce memory modules in the form of add in cards. The second option explored is to leverage CXL Memory Pooling by using a memory pool controller such as [5]. While the original intent of memory pooling is to leverage memory from a variety of different sources from more than one system in order to avoid over-provisioning memory that can lead to memory stranding [6, 16] and to add to overall memory, we look at memory pooling as a means to compensate for older systems that may be reused [3, 13] that may only need more memory in order to perform at an acceptable level.

4 RELATED WORK

Research on the sustainability of computing systems has been gaining prominence recently, with researchers studying overall environmental sustainability as well as the specific aspects of sustainable computing [10–13, 18]. More directly related to our current research, Junkyard Computing [24] proposes to reuse mobile phones for processing. We propose to use old servers with added memory. [6] carries out an analysis of CXL pooling on Azure cloud and compares and explores CXL memory pool deployment strategies. Further, and more directly relevant to our paper, it suggests that economical costs may be reduced by reusing DDR4 from retired servers.

5 CONCLUSIONS

Based on the analysis provided, it is evident that CXL technology presents a promising opportunity to augment the memory capacity of current systems. However, it is crucial to note that compatibility with existing machines is highly dependent on the specific PCIe version employed.

ACKNOWLEDGMENTS

The authors would like to thank the reviewers for their time.

REFERENCES

[1] [n. d.]. Graph500. http://www.graph500.org/.

[2] Ishwar Agarwal. 2022. CXL Overview and Evolution. http://hc34.hotchips.org/assets/program/tutorials/CXL/Hot%20Chips%202022%20CXL%20Overview%20and%20evolution.pdf

[3] Environmental Protection Agency. [n. d.]. Reduce, Reuse, Recycle. https://www.epa.gov/recycle Accessed on 08.06.2023.

[4] AMD. [n. d.]. AMD EPYC TM 7002 Series Processors. https://www.amd.com/en/processors/epyc-7002-series

[5] Astera Labs 2022. *Leo Memory Connectivity Platform for CXL 1.1 and 2.0*. Astera Labs. https://www.asteralabs.com/wp-content/uploads/2022/08/Astera_Labs_Leo_Aurora_Product_FINAL.pdf

[6] Daniel S. Berger, Daniel Ernst, Huaicheng Li, Pantea Zardoshti, Monish Shah, Samir Rajadnya, Scott Lee, Lisa Hsu, Ishwar Agarwal, Mark D. Hill, and Ricardo Bianchini. 2023. Design Tradeoffs in CXL-Based Memory Pools for Public Cloud Platforms. *IEEE Micro* 43, 2 (2023), 30–38. https://doi.org/10.1109/MM.2023.3241586

[7] Jeff Borland, Peter Dakins, David Johnson, and Ross Williams. 2023. *IPCC, 2023: Summary for Policymakers. In: Climate Change 2023: Synthesis Report. A Report of the Intergovernmental Panel on Climate Change.* Technical Report. Working Groups I, II and III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change.

[8] Avery Ching, Sergey Edunov, Maja Kabiljo, Dionysios Logothetis, and Sambavi Muthukrishnan. 2015. One Trillion Edges: Graph Processing at Facebook-Scale. *Proc. VLDB Endow.* 8, 12 (Aug. 2015), 1804–1815. https://doi.org/10.14778/2824032.2824077

[9] CXL Consortium 2022. *Compute Express Link (CXL) Specification*. CXL Consortium. Rev. 3.0.

[10] Sam Fankhauser, Stephen M. Smith, Myles Allen, Kaya Axelsson, Thomas Hale, Cameron Hepburn, J. Michael Kendall, Radhika Khosla, Javier Lezaun, Eli Mitchell-Larson, Michael Obersteiner, and Lava. 2022. The meaning of net zero and how to get it right. *Nature Climate Change* 12, 1 (January 2022), 15–21. https://doi.org/10.1038/s41558-021-01245-

[11] World Business Council for Sustainable Development and World Resources Institute. 2004. *The Greenhouse Gas Protocol: A Corporate Accounting and Reporting Standard*. World Business Council for Sustainable Development.

[12] Udit Gupta, Mariam Elgamal, Gage Hills, Gu-Yeon Wei, Hsien-Hsin S. Lee, David Brooks, and Carole-Jean Wu. 2022. ACT: Designing Sustainable Computer Systems with an Architectural Carbon Modeling Tool. In *Proceedings of the 49th Annual International Symposium on Computer Architecture* (New York, New York) *(ISCA '22)*. Association for Computing Machinery, New York, NY, USA, 784–799. https://doi.org/10.1145/3470496.3527408

[13] Udit Gupta, Young Geun Kim, Sylvia Lee, Jordan Tse, Hsien-Hsin S. Lee, Gu-Yeon Wei, David Brooks, and Carole-Jean Wu. 2022. Chasing Carbon: The Elusive Environmental Footprint of Computing. *IEEE Micro* 42, 4 (2022), 37–47. https://doi.org/10.1109/MM.2022.3163226

[14] Intel. [n. d.]. 3rd Gen Intel Xeon scalable processors brief. https://www.intel.com/content/www/us/en/products/docs/processors/xeon/3rd-gen-xeon-scalable-processors-brief.html

[15] Jure Leskovec and Andrej Krevl. 2014. SNAP Datasets: Stanford Large Network Dataset Collection. http://snap.stanford.edu/data.

[16] Huaicheng Li, Daniel S. Berger, Lisa Hsu, Daniel Ernst, Pantea Zardoshti, Stanko Novakovic, Monish Shah, Samir Rajadnya, Scott Lee, Ishwar Agarwal, Mark D. Hill, Marcus Fontoura, and Ricardo Bianchini. 2023. Pond: CXL-Based Memory Pooling Systems for Cloud Platforms. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2* (Vancouver, BC, Canada) *(ASPLOS 2023)*. Association for Computing Machinery, New York, NY, USA, 574–587. https://doi.org/10.1145/3575693.3578835

[17] OpenAI. 2023. https://chat.openai.com/ Accessed on 08.06.2023.

[18] David A. Patterson, Joseph Gonzalez, Quoc V. Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David R. So, Maud Texier, and Jeff Dean. 2021. Carbon Emissions and Large Neural Network Training. *CoRR* abs/2104.10350 (2021). arXiv:2104.10350 https://arxiv.org/abs/2104.10350

[19] Rambus Press. 2022. https://www.rambus.com/blogs/compute-express-link/ Accessed on 08.06.2023.

[20] Karl Rupp. 2018. https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/

[21] Debendra Das Sharma. 2022. Compute Express Link®: An open industry-standard interconnect enabling heterogeneous data-centric computing. In *2022 IEEE Symposium on High-Performance Interconnects (HOTI)*. 5–12. https://doi.org/10.1109/HOTI55740.2022.00017

[22] Shigeru Shiratake. 2020. Scaling and Performance Challenges of Future DRAM. In *2020 IEEE International Memory Workshop (IMW)*. IEEE, 1–3. https://doi.org/10.1109/IMW48823.2020.9108122

[23] Smart Modular Technologies 2023. *Introduction to Compute Express Link (CXL) Memory Modules*. Smart Modular Technologies. https://www.smartm.com/api/download/fetch/527 M-WP011.

[24] Jennifer Switzer, Gabriel Marcano, Ryan Kastner, and Pat Pannuto. 2023. Junkyard Computing: Repurposing Discarded Smartphones to Minimize Carbon. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2* (Vancouver, BC, Canada) *(ASPLOS 2023)*. Association for Computing Machinery, New York, NY, USA, 400–412. https://doi.org/10.1145/3575693.3575710