

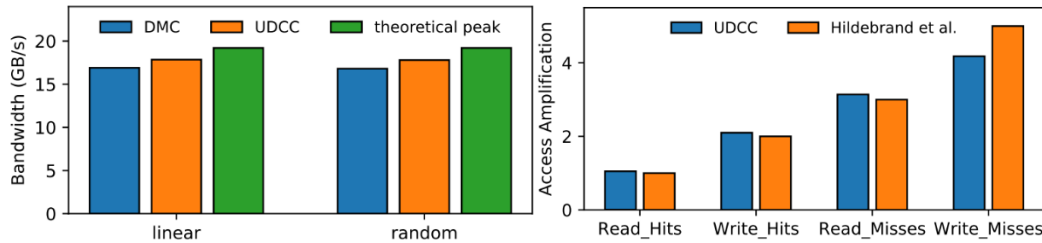
# A Cycle-level Unified DRAM Cache Controller Model in gem5

Maryam Babaie, Ayaz Akram, and Jason Lowe-Power  
Computer Science Department, University of California Davis

The increasing growth of applications' memory demands has led the CPU vendors to deploy large DRAM caches, backed by large non-volatile memories like Intel Optane (e.g., Intel's Cascade Lake). Previous work has explored many aspects of DRAM cache design in simulation such as the caching granularity,<sup>1</sup> dram cache tag placement,<sup>2</sup> etc. to improve performance. However, these works do not provide an open-source DRAM cache modeling platform for a detailed micro-architectural and timing analysis.

In this presentation we will describe a cycle-level unified DRAM cache and main memory controller (UDCC) for gem5. The protocol is inspired by the actual hardware providing DRAM cache, such as Intel's Cascade Lake, in which a DRAM cache is backed by an NVRAM as the off-chip main memory sharing the same bus. We leverage the cycle-level DRAM and NVRAM models in gem5. Our model implements the timing and micro-architectural details enforced by the memory interfaces including the DRAM timing constraints, scheduling policies, buffer sizes, and internal queues. We implement a DRAM cache model that is direct-mapped, insert-on-miss, and write-back to mirror Intel's Cascade Lake design.

**Evaluation:** We verified this model against the performance analysis of Intel's Cascade Lake studied by Hildebrand et. al.<sup>3</sup> First, we performed a comparison of the effective memory bandwidth observed by last level cache (LLC) with gem5's default memory controller (DMC) and UDCC. We used a synthetic read memory traffic pattern such that (nearly) all requests will be hits in the DRAM cache in the case of a system where we use UDCC. Figure 1 (left) provides both controllers' read bandwidth and compares these numbers to the theoretical peak bandwidth possible with the DDR4 device used as the DRAM cache. We observe the bandwidth attained by DMC and UDCC are similar. The scheduling policy implemented in both controllers can explain the slight difference. Moreover, we calculate the access amplification values for UDCC by dividing the effective bandwidth by the sum of the average bandwidth of DRAM and NVRAM devices for a particular run. The comparison of the two access amplification values is shown in Figure 1 (right). Our results match the actual hardware in all cases with only one exception, write misses. In actual hardware, on a write miss in DRAM cache, the block is first allocated by reading it from NVRAM and then writing it into DRAM. The actual data is then written into the DRAM. We merge these two DRAM writes, and thus, our implementation leads to one less access for a write miss compared to the actual hardware.



**Figure 1.** Comparison of observed bandwidth by LLC in DMC vs. UDCC (left) and the comparison of access amplification of UDCC protocol vs Intel's Cascade Lake (right).

In our presentation we will further describe the performance of UDCC with different DRAM technologies as the DRAM cache. We will show the performance of UDCC with main memories slower and faster than regular NVRAM, and the effect NVM wear leveling on UDCC. We believe this new tool can be used to explore new unified DRAM cache and memory controller designs as part of an agile and full-system simulation platform. Using this model which implements the microarchitectural details of realistic DRAM caches on a simulator, can help find any potential improvement for the next generation of memory systems.

<sup>1</sup> M. K. Qureshi and G. H. Loh, "Fundamental latency trade-off in architecting dram caches: Outperforming impractical sram-tags with a simple and practical design," in 2012 45th Annual IEEE/ACM International Symposium on Microarchitecture. IEEE, 2012, pp. 235–246.

<sup>2</sup> G. H. Loh and M. D. Hill, "Efficiently enabling conventional block sizes for very large die-stacked dram caches," in Proceedings of the 44th Annual IEEE/ACM International Symposium on Microarchitecture, 2011, pp. 454–464.

<sup>3</sup> M. Hildebrand, J. T. Angeles, J. Lowe-Power, and V. Akella, "A case against hardware managed dram caches for nvram based systems," in 2021 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS). IEEE, 2021, pp. 194–204.