Title: Addressing Power and Performance Issues in DRAM-based Main Memory

Area: Computer Architecture - Memory Hierarchy - Main Memory

Extended Abstract

The onset of multicore/many-core era and increasing workload footprints have exacerbated the 'Memory Wall' problem. Even though the size of on-chip caches is increasing, its effectiveness is being limited by the interference among the cores' accesses, resulting in an increased need of a high performance main memory system further. Dynamic Random Access Memory (DRAM) based main memory dominates the present memory landscape because of its highly mature fabrication technology and affordable cost. To meet the demand for increased memory capacity, with the help of advances in fabrication technology, high-density commodity DRAM devices are realized. JEDEC [1] has announced DDR4 standard with device densities as high as 32Gb. As memory capacity and accesses increase, its contribution to the total system power also increases.

In our work, we address both performance and power aspects of high-density DRAM devices. To improve the memory performance we propose three techniques, two of which reduce the impact of the refresh operation on performance, and the third technique increases the available bank level parallelism. These techniques also have the benefit of reduced memory power consumption making them energy-efficient.

DRAM cell stores information as charge in a capacitor. With time, the capacitor loses charge due to leakage. To maintain data integrity, the DRAM devices are periodically refreshed. For the new high-density devices, the time spent in refreshing the device is significant (for example, a $16Gb$ device spends around 6% of the time in refreshing). The major downside of the refresh is that when a DRAM device is refreshing, it is unavailable for the computing cores. Hence, the memory requests are delayed for a much longer time, degrading the performance significantly.

Considering high-density DRAM devices, we propose two techniques that target to reduce the impact of refresh on the overall system performance. The first technique, namely, *Scattered Refresh* [3], is motivated by an observation that a memory array in a DRAM bank is made up of small independent subarrays [2] sharing some peripheral circuitry. Scattered Refresh introduces small changes to this global circuitry to allow access to multiple subarrays during refresh. To utilize multiple subarrays at the same time, the refresh controller on the DRAM device scatters the rows to be refreshed across different subarrays. This allows overlap in activation and precharge of rows in different subarrays. Scattered Refresh effectively removes the precharge operation of one row from the critical path of activation of another row and hence, reduces the refresh cycle time. A direct implication of reduced refresh cycle time is that refreshes complete faster, increasing the availability of the banks to the compute cores. Considering a 4 core system with $8Gb$ devices, we show that Scattered Refresh achieves up to 10.2% improvement in overall system performance. Scattered Refresh, being orthogonal to the existing refresh handling techniques, can be employed along with any of them boosting their effectiveness further.

The second refresh handling technique we propose revisits the refresh process and questions the fundamental design choices involved in it. *Enhanced Fine Granularity Refresh (EFGR)* [4] increases the availability of the DRAM banks by refreshing fewer banks of a rank per refresh. At rank level, EFGR achieves parallel operation of memory request service and the refresh operation. EFGR introduces three optimizations to achieve this parallel

operation. First, EFGR proposes few small hardware changes to the peripheral circuitry of a DRAM device to decouple the refreshing and non-refreshing banks. Second, the memory controller is provided with the maximum number of banks that can be accessed along with a refresh so as to meet the peak power constraint. Finally, EFGR relaxes the pre-refresh condition that all banks need to be precharged before a refresh operation and limits the condition to only those banks to be involved in the refresh. From our experiments with a 4 core system, considering $16Gb$ devices, we observe that EFGR achieves 6.8% improvement in overall system performance along with a reduction in memory system power. EFGR not only outperforms all the state-of-the-art refresh handling techniques but also being complementary, is employable along with them.

Partial Page Activation (PPA) [5], our third techinque targets to improve both performance and energy-efficiency of the memory. To service a memory request, when a DRAM bank is accessed, an entire row (8KB in DDR4) of data is activated and brought in to a temporary storage called row buffer. This design is primarily motivated to exploit locality in memory accesses. Motivated by an observation that access locality is lost due to destructive interference among multiple access streams from various cores, PPA proposes to reduce the row buffer size so that energy consumption during activation and precharge operations reduces significantly. Upon closer study of the DRAM memory arrays, we notice that the memory arrays are made up of several smaller arrays called tiles [2]. By selectively activating the tiles, per access, the row buffer can be effectively reduced to half or one-fourth. Exploiting this, PPA introduces few non-disruptive changes to the DRAM array and significantly reduces the dynamic energy consumption of the memory. Further, by introducing small changes to the peripheral circuitry of a DRAM bank, PPA allows access to multiple smaller banks (with reduced row sizes) so as to increase the bank level parallelism. For a 4 core system, considering $8Gb$ devices, we observe that PPA achieves 7.4% and 11.9% improvement in performance along with 8.8% and 14.8% savings in memory power, by half page and quarter page activation, respectively.

Given that any new design in the DRAM is heavily scrutinized with its demand on area, many memory techniques with high area overhead are not endorsed by the industry. Our proposed techniques introduce minimal changes to the exisiting DRAM devices, making them attractive candidates to be adopted by the industry for future systems.

References

[1] JESD79-4, JEDEC Committee JC-42.3 Std. DDR4, Sept. 2012.

[2] Itoh K, "VLSI Memory Chip Design", Springer, 2001.

[3] Venkata Kalyan Tavva, Ravi Kasha and Madhu Mutyam. 2014. *Scattered refresh: An alternative refresh mechanism to reduce refresh cycle time.* Asia and South Pacific Design Automation Conference (ASP-DAC), pp.598 - 603. doi: 10.1109/ASP-DAC.2014.6742956

[4] Venkata Kalyan Tavva, Ravi Kasha, and Madhu Mutyam. 2014. *EFGR: An Enhanced Fine Granularity Refresh Feature for High-Performance DDR4 DRAM Devices.* ACM Trans. Archit. Code Optim. (TACO) 11, 3, Article 31 (October 2014). DOI=10.1145/2656340 http://doi.acm.org/10.1145/2656340

[5] Venkata Kalyan Tavva and Madhu Mutyam. *Partial Page Activation for a High-Performance and Energy-Efficient Memory.* Work under review, communicated to IEEE Transactions on Computers.