CMP SCI 635: Modern Computer Architecture

Density Trade-offs of Non-Volatile Memory as a Replacement for SRAM based Last Level Cache

Name: Kunjal Panchal Date: 1st Oct, 2019

Student ID: 32126469 Paper: 8 – Cache [Korgaonkar18]

Strengths:

1. The proposed architecture can tolerate high asymmetry in write latency and delivers significant performance improvements over a traditional SRAM LLC. It mitigates this write latency induced performance degradation and improves the performance of a 4 core system with an 8MB of STTRAM based exclusive LLC by an average of 26%.
2. To mitigate the performance impact of write latency as well as reduce write energy; spin-hall-effect memory is another spintronic memory candidate that can reduce the write latency but at the cost of lower density. While technologists have been exploring material and device options to reduce the performance gap between emerging memories and SRAMs, circuit and architectural techniques have also been investigated. The write latency can be reduced by upsizing the write access transistors or using differential bits at the circuit level. However, these will result in higher overall power and lower density, thereby negating the density advantage offered by the emerging NVM technologies. The techniques of "Write Congestion Aware Bypass" and "Virtual Hybrid Cache" handles these shortcomings to some extent.

Weaknesses:

1. Virtual Hybrid Cache used between L2 SRAM and NVM LLC is just another level of cache in the hierarchy. And that will increase overall cost and on-chip area requirements. Furthermore, it still doesn't do much in the case of programs with poor temporal locality.
2. For traditional LLCs and SRAMs; the proposals are focused on increasing the hit rate by predicting which cache lines will be used again, the ones which are less likely won't be retained, giving room to cache lines which will be fetched again. This just reduces the cache miss rate, still doesn't solve the problem of write congestions. And bypassing the writes can improve the write latency but will have higher miss rates.

   {Question Topic related to this point: } Can we not leave STTRAM LLC to higher rate of write bypasses and keep the sum of main memory reads and those writes to upper level caches faster than what it would take for a STTRAM LLC write? That way, larger number of LLC misses are compensated by faster main memory -> L1 cache load.
3. The way to get the optimal thresholds for write bypassing is never mentioned.

Questions/Assertions:

1. "*A hypothetical SRAM-like write latency STTRAM LLC (+0 ns) would gain 15% over the 4 MB SRAM baseline, which drops to 13% and 5% as the write latency is increased to 5 ns and 10 ns respectively. But when the write latency is further increased to 20 ns, all the capacity benefits are lost, and overall*

the performance drops by 15% when compared to the 4 MB SRAM baseline." How are the write latencies changed in a simulation? Do they just change the circuit parameters for it?

2. How does an Inclusive LLC simplify coherence flows? Doesn't updating two, rather than only one, cache line is more error-prone and cycle expensive?
3. Suggestion for a Hybrid LLC between Inclusive and Exclusive LLCs to improve space versus write cycles trade-off:

   When a cache line comes first in LLC from main memory, we store two copies of it. When higher level caches demand a line, we send one copy to them and keep only the other one in LLC. Now if that line has been changed, the updated copy will come to LLC and we can then store it back and update the already existing dirty copy of it.

   Between L1 and L2, the copy will be the only one and when we need update it, we store it just in LLC.

   If a new cache line is getting inserted, we target 2 different cache lines which have 2 copies of the same line in the LLC, we remove each copy and insert 2 copies of the new cache line.

   If the upper level caches need the previous lines, we still have one copy and if that gets updated, we can just write two updated copies by the same technique of targeting two lines which have duplicate copies.

   This technique sort of "juggles" the cache lines between the caches, but has potential to decrease main memory access.

4. If the surface area is not smooth, the electric/magnetic characteristics of the MTJ will degrade. On the paper "*Title: Demonstration of yield improvement for on-via MTJ using a 2-Mbit 1T-1MTJ STT-MRAM test chip. Journal: Proceedings of 2016 IEEE International Memory Workshop*"; Tetsuo Endoh's group has tackled the issue by developing a special polishing process technology to prevent any interference between the MTJ and its lower electrode. To further test the success of this development, a 2-Mbit STT-MRAM test chip integrating the new technology has been designed to verify the space needed for the integrated circuits - this includes more than 1million MTJs. "*Not only does this test chip show a 70% improvement in its memory bit yield compared to standard STT-MRAM, but its memory cell area is reduced by 30%,*" says Endoh.
5. Can we use STT RAM for L1 or L2 caches? "*System level exploration of a STT-MRAM based Level 1 Data-Cache*" proposes using STT-RAM for the L1 data cache while keeping all other caches SRAM-based. Also, the instruction cache is mostly read-only, except when there is a miss, in which case the line must be fetched and filled into the cache. This means that, when using STT-RAM (or really any other NVRAM technology), the expensive write operations occur less frequently compared to using STT-RAM for the data cache. The paper shows that by using an SRAM loop cache (like the LSD in Intel processors) and an STT-RAM instruction cache, energy consumption can be significantly reduced, especially when a loop is being executed that fits entirely in the loop cache.