ECE 571 – Advanced Microprocessor-Based Design Lecture 26

Vince Weaver

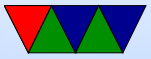
http://web.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

4 November 2020

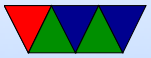
Announcements

- Homework 8 was poster a bit late (sorry about that)
- Review midterm
- Don't forget project ideas



HW#7 Review

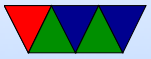
- Number of TLB events, 6 predefined 50 vendor. Tedious to count? Could grep/pipe into wc
- Naive matrix multiply, STLB
  - regular: 23kload 9kstore 0major, 1609 minor
  - swapped: 576kload 70kstore, 0 major, 1638 minor
  - Why not more pagefaults? Kernel stat wrong? Pre-faulting by kernel?
    Odd major faults always 0
    `/usr/bin/time -v`



major only means goes to disk, in our case we are allocating memory and filling it so no going to disk
So why not paging of executable in? Well the first time you run it after boot there should be, but after that likely in disk cache so not need to go to disk.

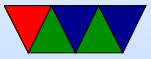
  - Note a TLB miss does not always equal a page fault

- Number of pages to cover 24MB, true that in theory also needs some 2nd-level pages too (hard to know how many without knowing where in memory things are): 6000. Note anything about minor page faults we saw? (program is doing a 1024x1024 multiply, 8 byte doubles,



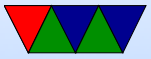
3 of them needed for MMM, so we'd expect more page faults than we got)

- On Linux can use /proc/pid/maps and /proc/pid/pagemap to actually look at the page table mappings



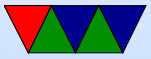
Why not have large SRAM

- SRAM is low power at low frequencies but takes more at high frequencies
- It is harder to make large SRAMs with long wires
- It is a lot more expensive while less dense (Also DRAM benefits from the huge volume of chips made)
- Leakage for large data structures

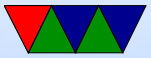


Saving Power/Energy with RAM

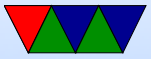
- AVATAR: A Variable retention time aware refresh for DRAM systems by Qureshi et al.
  - JEDEC standard: cell must have 64ms retention time
  - Why refresh bad? Block memory, preventing read/write requests
  - Consume energy (6,28,35)
  - The bigger DRAM gets, more refresh needed
  - predict that in 64Gb chips 50% of Energy will be in refresh



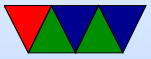
- Multi-rate refresh possible – detect which cells need more and refresh them more often (can be a 4-8x difference)
- VRT (variable retention rate) a problem. Some cells switch back and forth between. So when you probe it might check fine, but then fail later.
- They find that addition of cells stabilized to one new cell/15 mins over time
- Use ECC to catch these errors, though relying on ECC in this case can lead to uncorrectable error every 6 months



- They propose using ECC to adjust the VRT at runtime based on errors that are found
- They find on a 64Gb chip improves perf by 35% and **Energy-Delay** by 55%
- "Refresh-wall"
- Memory controller keeps track of this info
- VRT first reported in 1987. Fluctuations in GIDL (gate-induced drain leakage) presence of "trap" near the gate region
- Intel and Samsung say VRT one of biggest challenge in scaling DRAM

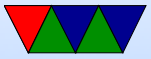


- VRT not necessarily bad – can cause retention to get better!
- Test – use FPGA to talk to 24 different DRAM chips, at controlled temperatures.
  Why do they use an FPGA?
- Actually it's just 3 chips from different vendors, each with 8 chips (for 24)
- Look into ECC. Soft-error rate is 200-5000 FIT/Mbit. Every 3-75 hours for 8GB DIMM. Soft errors happen 54x-2700x lower rate than VRT
- Downside of ECC ... have to scrub memory to check



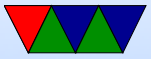
for errors. Also has energy/perf overhead. Energy to refresh DIMM 1.1mJ, energy to scrub 161mJ (150x) but if you scrub every 15 minutes it's a win.

– Use memory system simulator USIMM



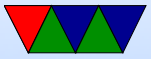
Cryogenic Memory

- Dip DIMMS in liquid nitrogen
- Low power? Faster? Interface with quantum circuits?

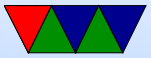


Rowhammer

- Been observed for years, adjacent rows discharging can affect nearby rows
- Particularly bad in DDR3 from 2012-2013
- Accessing same row over and over can make voltage fluctuations in nearby rows, causing faster leakage than normal
- Mitigations? Refresh more often? ECC? Refresh nearby lines if a lot of row hammering going on?

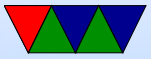


- Can cause exploit. Google NaCl disable “cflush” exploit (need to force access to row)
- Can also trigger just with lots of cache misses
- If you can flip bits of kernel/trusted pointers to point to something you control, then you win.



DRAM – Mobile DRAM

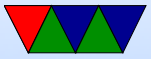
- From Micron: "TN-46-12: Mobile DRAM Power-Saving Features", 2009
- Temperature-Compensated Self Refresh (TCSR) – Auto adjust refresh timings based on temperature
- Partial Array Self Refresh (PASR) – only refresh parts of RAM that have data in them
- Deep Power Down (DPD) – enable turning off the voltage generators when maintaining DRAM not needed
- Has equations for estimating power usage



DRAM – Elsewhere

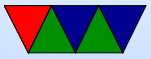
- Tom's Hardware. 2010. "How Much Power Does Low-Voltage DDR3 Memory Really Save?" Using low-voltage (1.25 or 1.35 rather than 1.5) DDR3 DRAM can reduce power by 0.5-1W. Slower performance settings, but not really noticeable.

- Linus Torvalds Rant from 2007: DRAM Energy not a prime concern. Just don't use FBDIMMs if you want low-power.



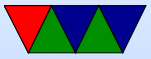
DRAM – Recent Academic

- "Rethinking DRAM Power Modes for Energy Proportionality", Malladi et al, Micro 2012.
  - DRAM spends lots of time idle, but latency is so high for wakeup it cannot utilize powerdown modes
  - Reference 25% of data-center energy usage is DRAM?
  - Use LPDDR2 trades bandwidth for efficiency
  - Current modes involve turning off DLLs (Delay-locked loops?) which are slow to turn on again, 700ns+
  - some background on DRAM operation

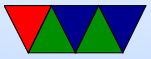


- Low-power mode sounds good, but then it takes 512 memory cycles of power to re-start (a lot of energy)
- Propose MemBLAZE. Moves clock generation out of DIMM and into memory controller, allowing fast wakeup

- "Towards Energy-Proportional Datacenter Memory with Mobile DRAM", Malladi et al, ISCA 2012.
  - Look at using LPDDR2 in servers rather than DDR3.
  - DDR3 often in "Active-idle" as many workloads do not allow sleep.



- "A Predictor-based Power-Saving Policy for DRAM Memories", Thomas et al, EuroMicro 2012.
  - Use a history based predictor to pick when to powerdown.
  - Say up to 20% of mobile devices and 25% of data center is DRAM
- "Rethinking DRAM Design and Organization for Energy-Constrained Multi-Cores", Udipi et al., ISCA 2010
  - DRAMs "overfetch" which hurts energy
- "A Comprehensive Approach to DRAM Power



Management", Hur and Lin, HPCA2008.

- Throttling and Power Shifting – slowing down to fit in power budget
- Put DRAMs in low power mode – available commercially but no one seems to use this yet
- Simulate for Power5 and DDR2-533
- Modify the memory controller