DRAM controller Research Directions

Created	@January 5, 2025 5:33 PM
Class	

Independent Read Write Request channels for Write Caching

1. Individual Read Write buffer, write buffer uses a high/low watermark. Read prioritizes over write. Issue writes request only after the saturation within Write buffer pass the high watermark.
2. Serve write commands until the saturation counter reaches the low watermark.
3. For every read request, first search the write buffer to find the needed request, if found return the data. Otherwise, access the DRAM

[1] Jacob, B., Ng, S. W., & Wang, D. T. (2008).
Memory Systems: Cache, DRAM, Disk. Morgan Kaufmann. 13.4.1, P512

Adaptive row buffer management

1. Probe the next request address, see if it would become a row buffer conflict, issue the command base on this information. So that next command issue is an auto-precharge

[1] Chang-Hsuan Chang, Ming-Hung Chang and Wei Hwang, "A flexible two-layer external memory management for H.264/AVC decoder," 2007 IEEE International SOC Conference, Hsinchu, Taiwan, 2007

Adaptive DRAM thermal refresh

1. From the information of temperature sensor, adjust the refresh interval of the refresh controller according to current temperature. Important for correctness

[1] C. -Y. Chang, P. -T. Huang, Y. -C. Chen, T. -S. Chang and W. Hwang, "Thermal-aware memory management unit of 3D-stacked DRAM for 3D high definition (HD) video," *2014 27th IEEE International System-on-Chip Conference (SOCC)*

Write Updated Partial Refresh

0. Some rows of DRAM is not valid or not ultilized or can be freed [2],[3]
1. Seperate DRAM into multiple segment, govern each segments with a pointer. Whenever a write command is issued, updates the corresponding segments' counter. Due to the sequential access nature of LLM model
2. For every issuing refresh, check if the refresh row address is greater than the pointer, if greater than the pointer, issue the demmy refresh[4]
3. Use dummy refresh mentioned in [4] to simply increment the inner counter of the DRAM bank through the memory controller

[1] Jafri, S. M. A. H., Hassan, H., Hemani, A., & Mutlu, O. (2020). Refresh triggered computation: Improving the energy efficiency of convolutional neural network accelerators.
ACM Transactions on Architecture and Code Optimization

[2] Shin, H.-S., Park, Y., Jung, J., Lee, J., & Chung, E.-Y. (2018). EXTREME: Exploiting page table for reducing refresh power of 3D-stacked DRAM memory. *IEEE Transactions on Computers*

[3] Isen, C., & John, L. K. (2009). ESKIMO: Energy savings using semantic knowledge of inconsequential memory occupancy for DRAM subsystem. In *Proceedings of the 42nd Annual IEEE/ACM International Symposium on Microarchitecture*

[4] Bhati, I., Chishti, Z., Lu, S.-L., & Jacob, B. (2015). Flexible auto-refresh: Enabling scalable and energy-efficient DRAM refresh reductions. In *Proceedings of the 42nd Annual International Symposium on Computer Architecture*

Bit plane Encoding for data error tolerance

1. FP16 re-encoding to create more zeroes for combating the random VRT effect, add the decoding and encoding logic to the datapath of read and write. uses BX only
2. This exploits the data similarity within the local group to create more conscutive 0s, thus increase the error resilience to DRAM VRT phenomenon
3. For int8 biases and weights, reencoding through historgram before run time, remap the 8-bits data to new value value s.t. more 0s can be created. User specify which data through command he is accessing to enable the decoding logic.
4. For partial results, int8 can ultilize DBX encoding yields better result, fp16 using BX only
5. For weights in fp16, can use BX then perform Histogram remapping using 4-bits for low critical path. Upper user specify the data type they want to fetch to ultilize this function. Decoding logic & encoding logic of BX can be shared.

[1] Kim, J., Sullivan, M., Choukse, E., & Erez, M. (2016). Bit-Plane Compression: Transforming Data for Better Compression in Many-Core Architectures. *Proceedings of the 43rd Annual International Symposium on Computer Architecture* (ISCA)

[2] G. Pekhimenko, V. Seshadri, O. Mutlu, M. A. Kozuch, P. B. Gibbons and T. C. Mowry, "Base-delta-immediate compression: Practical data compression for on-chip caches," *2012 21st International Conference on Parallel Architectures and Compilation Techniques (PACT)*

[3] Khan, S., et al. "The Efficacy of Error Mitigation Techniques for DRAM Retention Failures: A Comparative Experimental Study." *Proceedings of the 2014 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, 2014

Refresh Rescheduling(Refresh Postpone)

- Ultilize the DRAM specification, postpone the refresh if the row buffer is still on. According to the specification of DDR, it allows 8 refreshes to be

postponed. Due to the sequential accesses of the LLM model, one can postpone the refresh until the end of the current DRAM row access. One can actually delay or postpone the refreshes.

- According to DDR3 specification

REFRESH

REFRESH is used during normal operation of the DRAM and is analogous to CAS#-before-RAS# (CBR) refresh or auto refresh. This command is nonpersistent, so it must be issued each time a refresh is required. The addressing is generated by the internal refresh controller. This makes the address bits a "Don't Care" during a REFRESH command. The DRAM requires REFRESH cycles at an average interval of 7.8µs (maximum when $T_C \leq +85°C$ or 3.9µs; maximum when $T_C \leq +95°C$). The REFRESH period begins when the REFRESH command is registered and ends ^{t}RFC (MIN) later.

To allow for improved efficiency in scheduling and switching between tasks, some flexibility in the absolute refresh interval is provided. A maximum of eight REFRESH commands can be posted to any given DRAM, meaning that the maximum absolute interval between any REFRESH command and the next REFRESH command is nine times the maximum average interval refresh rate. Self refresh may be entered with up to eight REFRESH commands being posted. After exiting self refresh (when entered with posted REFRESH commands) additional posting of REFRESH commands is allowed to the extent the maximum number of cumulative posted REFRESH commands (both pre and post self refresh) does not exceed eight REFRESH commands.

The posting limit of eight REFRESH commands is a JEDEC specification; however, as long as all the required number of REFRESH commands are issued within the refresh period (64ms), exceeding the eight posted REFRESH commands is allowed.

ChargeCache

- Reduce the access latency for recently pre-charged rows or recently refreshed rows
  [1] Hassan, H., Mutlu, O., Rippel, E., & Hsieh, H. P. (2016). *ChargeCache: Reducing DRAM Latency by Exploiting Row Access Locality*. In Proceedings of the 22nd IEEE International Symposium on High-Performance Computer Architecture (HPCA) (pp. 581–593). IEEE

Replacing refresh with accesses

- RTC mentioning this, by replacing the refreshes with recent accesses. Some of the refreshes can be skipped. Or by synchronizing the access with refreshes to increase the possibility of skipping the refreshes

Huffman Decoder at (run-time) 2KB boundary

- After trace analysis and the information provided by the upper core, using this to prefetch or enter self refresh mode.
- If the data to fetch are weights, these weights can be pre-encoded is a 2KB boundary using HC or RLE, such that the read latency can be reduced even further
- A prefetch + mode of upper layer + AGU is needed to specify when to prefetch and decode the data.

Prefetcher Design & Analysis

- Choose the best prefetcher accoding to the application, Stream prefetcher and stream buffer should be the best choices.