

Lab Assignment 2: Main Memories

Objective

The purpose of this assignment is to understand the functionality of main memories, and to get an insight into various trade-offs related to the design of systems with main memories with a focus on DRAMs.

Helpful References

- Lecture notes about main memories
- MCSim paper is a helpful source for understanding the MCSim simulator used in Part B of the lab:
Mirosanlou, Reza, et al. "MCSim: An Extensible DRAM Memory Controller Simulator." IEEE Computer Architecture Letters 19.2 (2020): 105-109.
- MCXplore paper and documentation can be a good source for understanding various DRAM address patterns needed for Part A of the Lab
 - *Hassan, Mohamed, and Hiren Patel. "MCXplore: Automating the validation process of DRAM memory controller designs." IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 37.5 (2017): 1050-1063.*
 - <https://gitlab.com/FanusLab/mcxplore/-/blob/master/Docs/Manual.pdf>

Part A: DRAM Traces [50 pts]

- Write a C/C++ program that takes as input the following parameters:

```
-w [address_width] -m [address_mapping] -n [num_requests] -p [address_pattern] -t [type_pattern]
```

And generate a DRAM trace with the specified patterns. The trace should look like

Address : type, where address is in hex and type is either R (for read) or W (for write).

For example:

```
0x12345680 R
0x4cbd56c0 W
0x35d46f00 R
0x696fed40 W
0x7876af80 R
```

- DRAM Architecture: *you have to match the structure of the “DDR3_2Gb_x8” in the MCsim simulator to be able to also use your traces for PartB. Figure out the number of ranks, channels, banks, ..etc for this structure from MCsim.*
- Your program should be able to generate the following address and type patterns:
 - Address patterns:
 - Sequential
 - Random
 - All row hit
 - All row conflicts
 - Type patterns:
 - All R
 - All W
 - Switching: R, W, R, W,..etc.
- You should also consider the following mappings (where RW: Row, RNK: Rank, BNK: Bank, CL: Column):
 - RW-RNK-BNK-CL
 - RW-CL-RNK-BNK
 - RNK-BNK-CL-RW

Part B: Simulating Off-chip Memory [50 pts]

1) [10 pts] Use your tool from Part A to generate the following traces, where each trace should have 10K requests and the address mapping should be row:bank:column (assume a single channel and single rank).

- All Read, Sequential Trace
- All Read, Random Trace
- All Read, Conflict Trace
- All Read, Hit Trace
- Switching, Hit Trace

2) Familiarize yourself with MCsim simulator. Start with getting MCSim from here: <https://github.com/uwuser/MCsim> and read the README file.

3) [15 pts] Configure MCsim to use FR-FCFS and run all the 5 traces from step 1. Make sure that the address mapping in MCsim are set correctly. In MCsim the address mapping is determined in the ini file where each segment is coded as follows: Rank[0], BankGroup[1], Bank[2], SubArray[3], Row[4], Col[5]. So, AddressMapping=425 means row:bank:column.

Draw a graph that compares the execution time of all the five traces.

4) [10 pts] Change the FR-FCFS ini file to use close-page policy and repeat the experiments and redraw the graph similar to part 3.

5) [15 pts] Use the trace “All Read, Random Trace” to compare all the controller systems supported by MCsim. Plot the results in a bar-chart.