

**Course Name**: Computer Architecture Lab

**Course Number and Section**: **14:332:361:05**

**Experiment**: Lab #5 - Simulating a Cache

**Lab Instructor**: Christos Mitropoulos

**Date Performed**: 11/17/2016

**Date Submitted**: 12/7/2016

**Submitted by**: Pawel Derkacz - 151005994

--------------------------For Lab Instructor Use ONLY--------------------------

GRADE: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

COMMENTS:

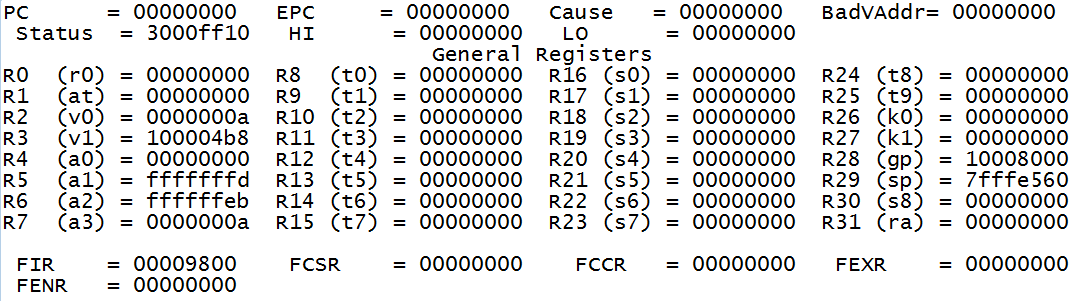
Electrical and Computer Engineering Department

School of Engineering

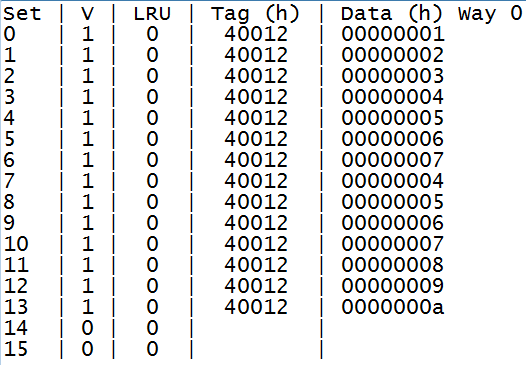
Rutgers University, Piscataway, NJ 08854

Assignment 1:

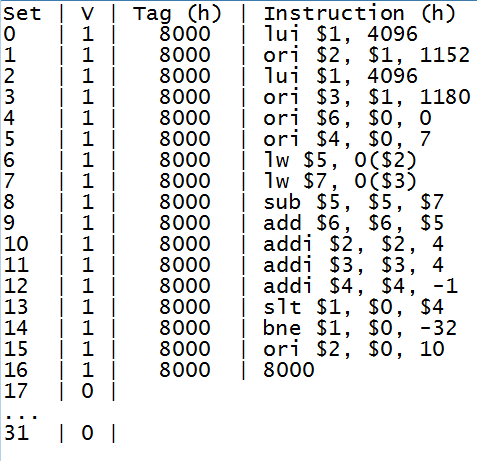
Register Statuses:



Data Cache:



Instruction Cache:



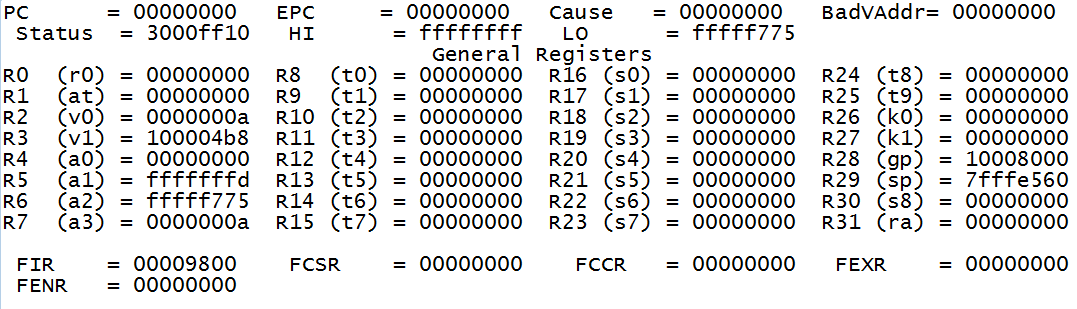
Explanation:

In regards to the data cache, since the cache size was chosen to only be 128 bytes with a block size of 4 bytes in a 2-way set associative manner, therefore there exist 4 selector bits and 16 blocks per set. However, there are only 14 elements in total, thus the data is stored similar to a direct mapping manner in the sense that the second set is not used due to lack of need for it and there is no evicting. The arrays are stored in with the selector bits seemingly to be the last 4 bits of the address.

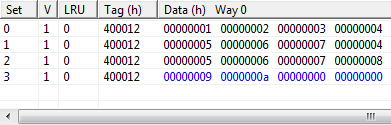
Regarding the instruction cache, the cache ran in direct mapping mode, but no evicting happened due to the fact that less than 32 instructions were present.

Assignment 2:

Register Statuses:



Data Cache:



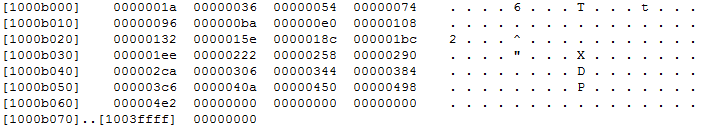
Explanation:

In regards to the data cache, since the cache size was chosen to only be 256 bytes with a block size of 16 bytes in a 4-way set associative manner, therefore there exist 2 selector bits and 4 blocks per set. Since there are 14 elements and only 4, 16 byte, blocks in each set, the 4 elements can be stored in each block with 3 elements having a non-zero offset. It seems that the way the arrays are stored follows that the last two are the selector bits and when 4 words are read, the 5th one has a different set to go to, and this repeats for the rest. For those that would go to the same set, if there is space, the data moves in with offset.

Assignment 3:

Note: Assuming that vector C will have the same form as vector A, since both are vectors.

Matrix C:



Assignment 4:

Code is at the end.

Assignment 5:

1) Cache size 128 Bytes, Block size 4Bytes, direct-mapping

Missed: 0

2) Cache size 256 Bytes, Block size 4 Bytes, direct-mapping

Missed: 0

3) Cache size 128 Bytes, Block size 16 Bytes, direct-mapping

Missed: 6

By having a larger block size on direct mapping caused, in this case, only 8 available lines to map to. Thus, several misses were made. For the first two, the number of lines (32 and 64, respectively) was enough for the program to store the data it needed.