

Lab 13

[Computer Architecture I ShanghaiTech University](#)

[Lab 12](#) Lab 13

Objectives:

- Explore the workings of virtual memory, specifically the TLB and the Page Table.
- Analyze TLB hit rate and Page Table hit rate and figure out what accesses optimize these values.

Setup

For this lab we will mostly be using the virtual memory simulation features of Camera, a cache and virtual memory simulator. You may also find the cache simulations interesting, however we won't be working with those here. You can build Camera from the [starter file](#) following the user guide in it.

Once Camera opens up, select the virtual memory option to open a visualization of the virtual memory system. In the top left you can see the contents of physical memory. Just below that is a listing of all the pages of virtual memory for this process. To the right of these items are the contents of the TLB and the Page Table. At this point these should all be empty as we haven't done anything yet. Read about the statistics of your memory system in the "PROGRESS UPDATE" box at the bottom of the window. This area will keep you updated on your status through the simulation as it progresses. You can move the simulation forward, backward or start it over from the beginning using the buttons to the right of the "PROGRESS UPDATE" box.

Exercises

Exercise 0 - Sanity Check

Before you continue, **MAKE SURE THAT YOU OPENED THE VM SIMULATOR AND NOT THE CACHE SIMULATOR.**

Exercise 1 - Working with CAMERA

Click the button labeled "Auto Generate Add. Ref. Str." at the right-hand side of the window. This will generate a set of ten address references. You can think of these as a series of RISC-V "load word" instructions reading from the memory address specified. Click the button labeled "Next" to begin the simulation.

For the rest of this exercise you are at the mercy of the "PROGRESS UPDATE" box. After each click of the "Next" button examine the contents of the box and the current state of the memory system. Try to really get an understanding of what is going on in the TLB, the Page Table, and Physical Memory at each step.

Once you have reached the end of the simulation note the number of TLB Hits and Misses and Page Hits and Faults. Write these numbers down, along with the sequence of memory accesses used to show to your TA during checkoff.

Checkoff

- Given the way the address was broken down, how big are the pages in this model?
- How many TLB Hits and Misses did we have for the randomly generated set of ten addresses? What about for Page Hits and Page Faults?
- Did you have any Page Hits? Explain why. Can you think of a set of ten addresses that would result in a Page Hit?
- Explain the process by which we turn a virtual address into a physical address for the very first access (emphasizing on TLB Misses and Page Faults).

Exercise 2 - Misses

Now that you've seen what a random workload looks like in the VM system let's try creating a custom workload with a specific property. Your goal for this exercise is to create a workload of ten memory accesses that will cause ten TLB misses and ten Page Faults. You should be able to come up with such a workload on paper, but then you should run it in CAMERA to verify your work. You can specify a custom workload in CAMERA by clicking the button labeled "Self Generate Add. Ref. Str." and entering in the addresses you want to reference one at a time. When you are satisfied that you've got a valid sequence write it down and be ready to show it to your TA during checkoff.

Checkoff

- Demonstrate that your ten memory accesses results in ten TLB Misses and ten Page Faults. Explain why such behavior occurs.

Exercise 3 - Fixing our Faults

Given your sequence of memory accesses from Exercise 2, can you find a change to a single parameter (e.g. TLB size, page table size, memory size, etc...) that would result in the same number (ten) of TLB misses but result in fewer than ten page faults? Work through this on paper and be ready to show your results to your TA during checkoff.

Checkoff

- Explain the single parameter change that would result in ten TLB misses, but **fewer** than ten page faults.

Schwertfeger, Sören <soerensch AT shanghaitech.edu.cn>

Chundong Wang <wangchd AT shanghaitech.edu.cn>

Last modified: 2020-06-06