

Fundamentos de Sistemas de Operação

MIEI 2018/2019

Laboratory session 6

Overview

TLB size and memory access times.

Measuring memory access times

In this work you will measure your computer TLB size and its memory access time vs. memory access times when of TLB misses. This work is based on the homework from chapter 19 of the OSTEP book.

The basic idea is to measure the time to access some defined number of pages. Those pages can be from some data structure like an array, that your program accesses at several positions, where each one corresponds to a different page. You should experiment with different numbers and compare the different results. While the number of pages fit the TLB you only get TLB hits and the corresponding memory access time. As soon as the defined number of pages doesn't fit the TLB, you start getting TLB misses and bigger access times.

Start by revisiting the program `timing.c` from Lab01 (see `timing2.c` from CLIP). Implement a `do_something` function that repeatedly accesses a limited number of pages, passed as argument. It's enough to read/write several ints from an int array, one (or a few) at each memory page, but your code should repeat that access pattern for a requested number of trials. As an example, the following can be the pseudocode for `do_something` function (see also the end of chapter 19):

```
Do_something( numtrials, numpages ):  
    jump = PAGESIZE/sizeof(int)  
    Forall numtrials:  
        For i=0; i<numpages*jump; i+=jump :  
            array[ i ] +=1
```

Your program must receive from command line (via main's `argv[]`) the parameters that define the number of pages to consider and the number of trials. As a result, it should calculate and print the average access time.

Finding TLB size

Execute your program with a low number of pages and trials. Increase number of trials until there is no significant changes to the page access time and to get a relevant result. After that, increase the number of pages to test and try to identify your computer TLB size. Probably, only when going above L2 TLB size will these be noticed.

For those tests construct a shell script that allows you to execute your program with increasing number of pages as argument. For example, test the sequence: 30, 60, 120, ... (doubling until 4000) and draw the results graph.

Hint: try "`help for`" at your command line.