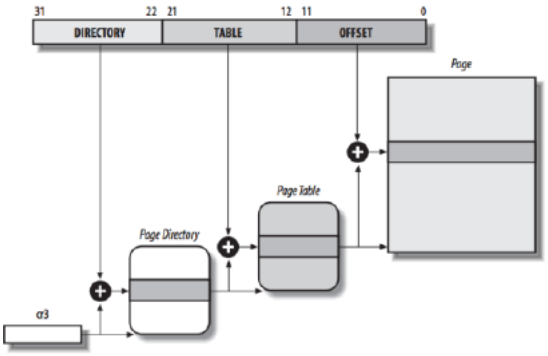
CS6600: Computer Architecture
Assignment #3

Submission deadline: 19th Sept, 2021

Consider a computer system that has 4MB main memory with 4KB page-frames. The first 1MB of main memory is used for kernel space and the remaining space is used for user processes. The computer system supports virtual memory and the page size is 4KB. Assume that the applications generate 32-bit virtual address. The computer system implements memory management unit (MMU) for translating virtual address into physical address. The MMU consisting a free page-frame list (which provides the list of all free page frames in the main memory) and a paging logic (which provides the mapping details of virtual pages to page-frames). The paging logic is implemented in a hierarchical structure, consisting a page directory and a page table. Each entry in the page directory/page table takes 4B space. For every process, there is a separate page directory/page table is maintained. The base address of a page directory is stored in CR3 register. The address translation happens as shown in the figure.



Initially, the page table is empty and the number of entries in the free page-frame list is equal to the number of page-frames in the main memory. Whenever an application generates a virtual address, the OS will check the page table to see whether the page containing the virtual address is mapped to a page-frame. If so, the page hit count is incremented. If the required page is not mapped to a page-frame, the page miss count is incremented and a victim page-frame is identified to map to the virtual page. We consider LRU policy for victim page-frame selection. The replacement policy first checks whether the free page-frame list is empty or not. If the list is not empty, the replacement policy selects the page-frame from the front of the list and map it to the virtual page. If the list is empty, the replacement policy selects the least recently used mapped page-frames as a victim. After selecting the victim page-frame, the mapping details are updated in the page table. Whenever a page-frame is deallocated, it will be appended at the end of the free page-frame list.

Design an MMU simulator (using your preferred programming language) that works as described above. The simulator needs to provide the statistics such as process-wise number of requests generated, the total number of read requests, the total number of write requests, page hit rate, page miss rate, total number of dirty page evictions, the total amount of space utilized for paging logic, etc.