# ECE3700J Introduction to Computer Organization
# Lab 7 – Virtual Memory

Purpose

In computing, virtual memory is a memory management technique that provides an "idealized abstraction of the storage resources that are actually available on a given machine" which "creates the illusion to users of a very large (main) memory". [Wikipedia]

In order to support this technique, a virtual address to physical address translation mechanism is created on top of the memory hierarchy we modeled in the last lab, as shown in the following structural diagram.
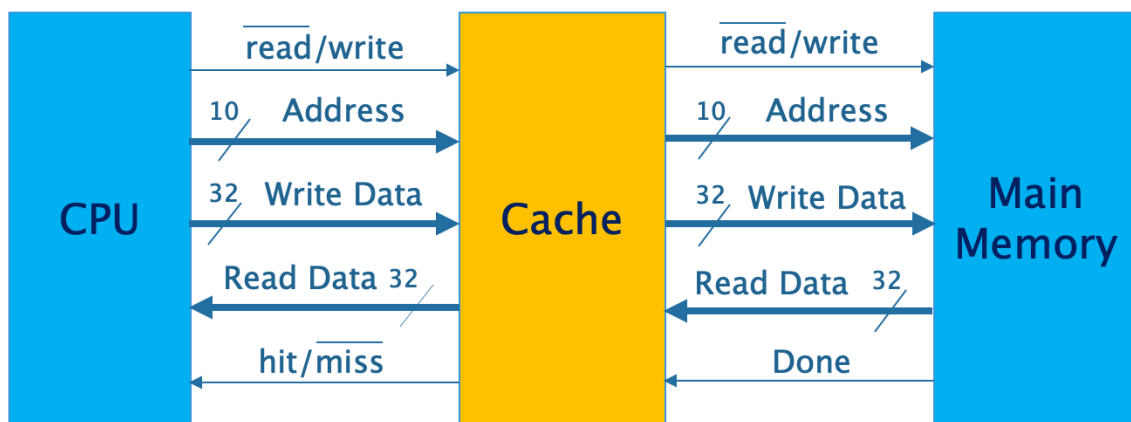
The virtual to physical translation mechanism typically includes a Page Table and a Translation Look-aside Buffer (TLB), as shown in the following diagram. As discussed in lectures, the relationship between TLB and Page Table resembles that between the cache and main memory.
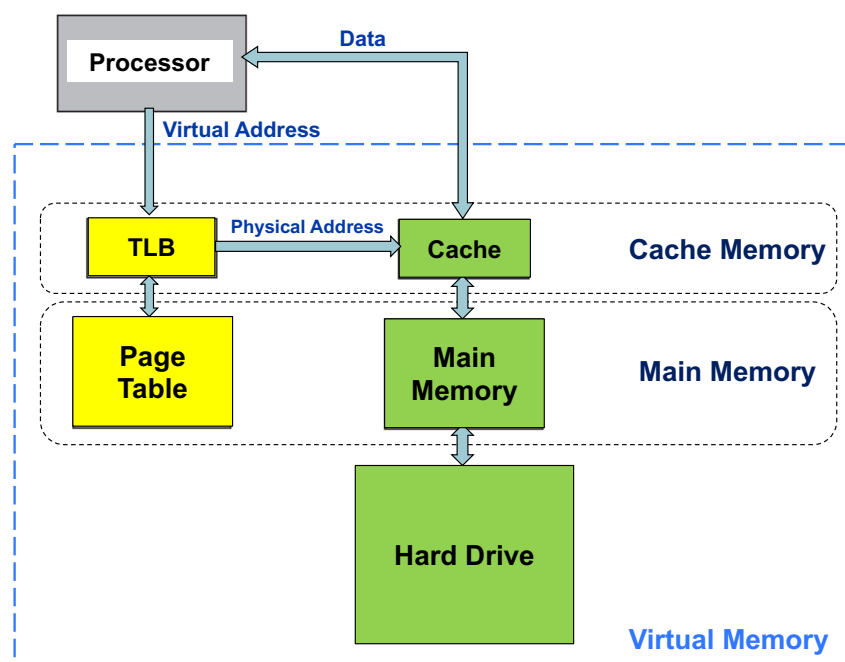
Tasks

Assume the following properties of the memory:
- Byte addressable
- Size of virtual address: 14 bits
- Size of physical address: 10 bits
- Size of the main memory: 1024 bytes
- Size of the cache: 64 bytes
- Size of a block: 4 words
- Cache associativity: 2-way associative
- Write technique: write back
- Cache replacement policy: Least Recently Used (LRU)
- Page size: 256 bytes
- Page Table entry size: 4 bytes each
- Size of TLB: 4×4 bytes
- TLB associativity: fully associative
- TLB Write technique: write back
- TLB replacement policy: LRU

When CPU needs to access a data/instruction in the memory, it sends a 14-bit virtual address to the TLB. If there is a hit in TLB, TLB then sends the corresponding physical address to the cache memory. If there is a TLB miss, TLB should then get the physical address from the Page Table, then send the corresponding physical address to the cache memory. If TLB fails to get the physical address from the Page Table, meaning the requested data is not currently in the physical memory, report a page fault.

Model the memory hierarchy and translation mechanism in Verilog HDL. Write a testbench to act like a CPU to provide a sequence of virtual addresses for reading or writing. Pre-load the main memory with randomly generated data by your team. Simulate the functions of the memory hierarchy with a Verilog simulator of your choice.

Notes about the design:
1. Clock signal is not needed.
2. You do not need to implement a FSM.
3. You are free to add some new features or new signals, if you think it is necessary.
4. For output of main memory, only showing the part whose value has been changed during the simulation process is enough.
5. On page fault, just indicate.
6. You are free to create your own design, but it should be reasonable, and you need to give a clear explanation to us during demonstration.
7. FPGA implementation is not required.

Team Organization

This lab is a team effort. Each team should consist of 3 students, randomly grouped. The work should be appropriately divided and distributed among all team members. Students are not allowed to switch teams without permission of the instructor.

Deliverables

- **Demonstration –** Every team should demonstrate to the teaching group the following before your lab session ends:
    1) Simulation results of the top-module of your design showing significant events and changes of your memory
    2) RTL schematic of your Verilog model generated with Xilinx Vivado software

    Each team member should be prepared for an oral exam on this lab during the demonstration.

- **Peer Evaluation** – Each team member is required to provide a peer evaluation for the team effort in this lab. The marks of the peer evaluation should be integers ranging between 0 to 10, inclusively, with 10 indicating the biggest contribution. A mark should be given to each team member including yourself according the team member's contribution based on your observation. A brief description of contribution of each team member should also be provided, as shown in the following table.

Name	Level of contribution (0 ~ 10)	Description of contribution
(yourself)		
(your lab partner)		
(your lab partner)		

- **Source Files –** All your Verilog source files and any other supporting files.

This is a 2-week lab. The full score for this lab is 300 points.

All required documents should be submitted on Canvas before **22:00pm, December 10, 2022**.

Grading

- Demonstration: 80%
    - Working Verilog model (simulation): 50%
    - Individual oral exam: 20%
    - RTL Schematics: 10%
- Source files and peer evaluation: 20%