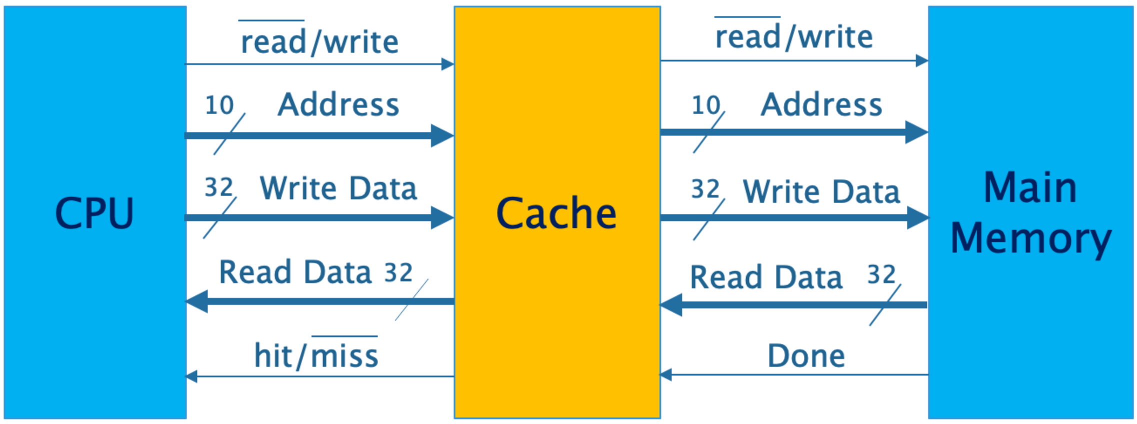
Ve370 Introduction to Computer Organization
Lab 6 – Cache Memory

Purpose

Cache memory is the top level of the memory hierarchy that interacts with the CPU directly. Communications between CPU and the rest of the memory hierarchy are typically through cache. Simplified interfaces between the CPU and cache and between cache and the main memory are shown in the following structural diagram.



This lab is intended to help you better understand the organization of memory hierarchy and the relationship between different levels of memory.

Tasks

Assume the following properties of the memory:
- Byte addressable
- Size of the main memory: 1024 bytes
- Size of the cache: 64 bytes
- Size of a block: 4 words
- Cache associativity: Direct mapped
- Write technique: write back
- Cache replacement policy: Least Recently Used (LRU)

When CPU needs to access a data/instruction in the memory, it sends a 32-bit address to the cache. If there is a hit in cache, CPU then reads or writes the cache memory according to the read/write signal. If there is a cache miss, cache should request the missing block from the main memory by sending the same 32-bit address to the main memory. You may assume the main memory always has hit, and add reasonable latency to the main memory access. Once main memory operation is finished according to its read/write signal, output signal Done should be asserted. Then CPU should resend the same address to try to request the data again.

Model the cache memory and main memory in Verilog HDL. Write a testbench to act like a CPU to provide a sequence of 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, say, using a write buffer, or adding a new signal, 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. 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.
6. 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 signals 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 1-week lab. The full score for this lab is 300 points.

All required documents should be submitted on Canvas before **22:00pm, November 27, 2021**.

Grading

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