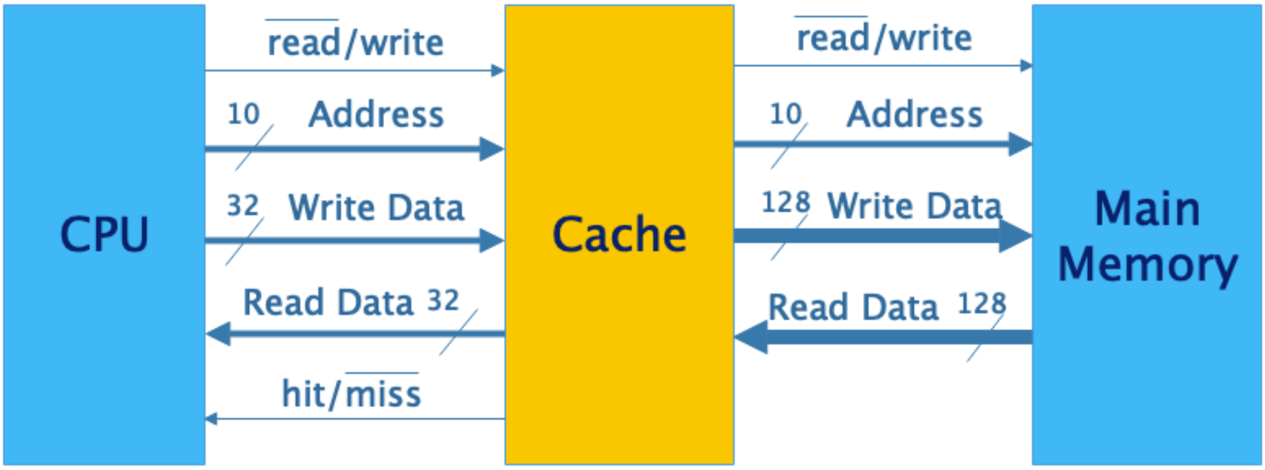
JOINT INSTITUTE
交大密西根学院

Ve370 Introduction to Computer Organization
Project 3

PROJECT DESCRIPTION

Cache memory is the top 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.



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: (a) Direct mapped; (b) 2-way associative
- Write technique: (1) write through; (2) write back
- Cache replacement policy: Least Recently Used (LRU)

When CPU needs to access a data/instruction in the memory, it sends a 10-bit address to the cache. If there is a hit in cache, CPU then reads/writes the cache memory. If there is a cache miss, cache should request the missing block from the main memory by sending the 10-bit address to the main memory. Then CPU should resend the same address to try again. For simplicity purpose, assume the main memory always has hit and latency of the main memory access is not considered.

Model the cache memory and main memory in Verilog HDL with combinations of (1)(a), (1)(b), (2)(a), and (2)(b) for cache associativity and writing mechanism. 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 an appropriate Verilog simulator.

DELIVERABLES

Written report is OPTIONAL for this project. The entire project including all Verilog modules must be submitted in a zipped folder and clearly indicate which Verilog simulator has been used. Simulations for all four different combinations (for cache associativity and writing mechanism) must be separated and clearly indicated. All Verilog modules must be clearly commented to receive full marks. Screen shots and explanations of simulation results can be submitted in a PDF file if you think it will be helpful.

This is a group assignment. One submission for each group is needed. Your work must be submitted electronically to Canvas before the specified due date.

GRADING POLICY
Correctness, completeness, clarity of the program: 80%
Source code: 20%

Late submission will result in 0 point for the source code part and deduction of 20% per day in the other part until all 100% is deducted.

DUE DATE
The project report is due by **11:59pm, November 21, 2019**