Using Cache Memory to Reduce Processor-Memory Traffic

James R. Goodman[1]

[1]Department of Computer Sciences
University of Wisconsin-Madison

ShanghaiTech University, 2013

Outline

CPU and Memory speed mismatch

- Example

  Motorola MC68000 10 MHz CPU clock; 5 MB/s Memory access rate, half its pins tasked with memory connection.

CPU and Memory speed mismatch

- Example

  Motorola MC68000 10 MHz CPU clock; 5 MB/s Memory access rate, half its pins tasked with memory connection.

  - 10x transistors = 30x memory bandwidth. Not feasible to increase pin number 30 fold.

Debates about On-chip Memory

- Dedicated on-chip memory with a relatively slower CPU may outperform a more powerful CPU with conventional memory.

Debates about On-chip Memory

- Dedicated on-chip memory with a relatively slower CPU may outperform a more powerful CPU with conventional memory.
- The chip should contain as much memory as the CPU needs.

Debates about On-chip Memory

- Dedicated on-chip memory with a relatively slower CPU may outperform a more powerful CPU with conventional memory.
- The chip should contain as much memory as the CPU needs.
- Microprocessors in 1983 need 0.25 MiB of memory, more than possible amount.

Debates about On-chip Memory

- Dedicated on-chip memory with a relatively slower CPU may outperform a more powerful CPU with conventional memory.
- The chip should contain as much memory as the CPU needs.
- Microprocessors in 1983 need 0.25 MiB of memory, more than possible amount.
- Higher performance CPUs apparently require more memory.

Debates about On-chip Memory

- Dedicated on-chip memory with a relatively slower CPU may outperform a more powerful CPU with conventional memory.
- The chip should contain as much memory as the CPU needs.
- Microprocessors in 1983 need 0.25 MiB of memory, more than possible amount.
- Higher performance CPUs apparently require more memory.
- Which leads to:

Debates about On-chip Memory

- Dedicated on-chip memory with a relatively slower CPU may outperform a more powerful CPU with conventional memory.
- The chip should contain as much memory as the CPU needs.
- Microprocessors in 1983 need 0.25 MiB of memory, more than possible amount.
- Higher performance CPUs apparently require more memory.
- Which leads to: On-chip memory is clearly not feasible in 1983, nor is it today.

Issues With Using Cache Memory

- Use of cache has aggravated bandwidth problem.

Issues With Using Cache Memory

- Use of cache has aggravated bandwidth problem.
- Cache optimization aspects:
  - Maximizing Hit Ratio
  - Minimizing Data Accessing Time
  - Minimizing Miss Penalty
  - Minimizing Overhead of Updating Memory, Maintaining Multi-cache Consistency

Issues with Using Cache Memory Cont.

- Optimization Usually Results in Larger Burst Bandwidth Requirement.

- Example

  IBM System/370 model 155
  Cache-Memory transfer rate: 100 MB/s
  Cache-CPU transfer rate is less than 1/3 of that.

Issues with Using Cache Memory Cont.

- Optimization Usually Results in Larger Burst Bandwidth Requirement.

- Example

  IBM System/370 model 155
  Cache-Memory transfer rate: 100 MB/s
  Cache-CPU transfer rate is less than 1/3 of that.

  - Reason: To exploit spatial locality, thus data fetched in large blocks, resulting in high memory bandwidth bursts.

Issues with Using Cache Memory Cont. 2

- To lower the bandwidth from backing store to cache:
  - Transfer small blocks from backing store to cache,
  - Experience long delays while a block is brought from backing store to cache.

Issues with Using Cache Memory Cont. 2

- To lower the bandwidth from backing store to cache:
  - Transfer small blocks from backing store to cache,
  - Experience long delays while a block is brought from backing store to cache.
- Explore the effectiveness of exploiting temporal locality, i.e. blocks fetched from backing store are only the size needed by CPU.
- Effective environment: single-board computer running Multibus or Versabus.

Single Board Computer Application

Usage of Buses

- Buses, if needed, are designed for generality and simplicity not for high performance.

Single Board Computer Application

Usage of Buses

- Buses, if needed, are designed for generality and simplicity not for high performance.
- Example

  Multibus by Intel Corporation

Single Board Computer Application

Usage of Buses

- Buses, if needed, are designed for generality and simplicity not for high performance.
- Example

  Multibus by Intel Corporation
  - Applications are severely limited by bandwidth of Multibus.
  - Try to determine whether a cache memory system can be implemented with Multibus
  - Allocation of memory should be handled by the system instead by the programmer

Caches in Single Board Computers

Proposal

Single-Board Computer

- CPU w/o local memory except for cache
- Backing store provided by Multibus

Caches in Single Board Computers

Proposal

Single-Board Computer

- CPU w/o local memory except for cache
- Backing store provided by Multibus

Question

- Can we build a cache that works with Multibus and supports multiple processors?
- How many processors can we support?

Caches in Single Board Computers Cont.

- Important criterion is maximize use of the bus.
- Optimize system performance by optimizing bus utilization, achieving higher performance by minimizing individual processors' bus requirements.
- Prefer individual processors to remain idle periodically over saturating bus traffic with data that will never be used.

Switching Contexts

Context Switch

Task switch results in cache being reloaded and CPU speeds reduced to bus speed.

- Effects may be minimized if task switching frequency is reduced.
- Utilize multiple processors for multiple tasks.
- Interrupt handling optimized so program only uses small portion of cache.

Write-Through or Write-Back

New Strategy: Write-Once

Effect of Write Strategy on Bus Traffic

Cold Start vs. Warm Start

Cache Size

Block Size

Lowering Overhead of Small Blocks

Effect of Large Address Blocks

Summary

- The first main message of your talk in one or two lines.
- The second main message of your talk in one or two lines.
- Perhaps a third message, but not more than that.

- Outlook
  - Something you haven't solved.
  - Something else you haven't solved.

For Further Reading I


A. Author.
Handbook of Everything.
Some Press, 1990.


S. Someone.
On this and that.
Journal of This and That, 2(1):50–100, 2000.