Spring 2001 Midterm Exam Review

When / Where

Friday, 26 October, 13:40-14:30 CDT

CEBA 3138 (Here)

Conditions

Closed Book, Closed Notes

Bring one sheet of notes (both sides), 216 mm × 280 mm.

No communication devices.

Format

One or two medium-sized problems, short answers.

Resources

Solved tests and homework:. `http://www.ece.lsu.edu/ee4720/prev.html`

Study Recommendations

Solve Old Problems

Memorizing solutions is not the same as solving.

Following and understanding solutions is not the same as solving.

Use the solutions for brief hints and to check your own solutions.

Topics

Introductory Material

ISA v. Implementation.

Technological Factors: Transistor speed and quantity, memory speed and size.

Different factors influencing ISA and implementation.

Design principles: Amdahl's law, locality.

CPU Performance Equation

Benchmark types.

Instruction Set Design

Data Types: What to include, what to leave out.

Basic integer and floating point

Packed types: BCD, integer, saturating integer.

Size choices.

Memory and Register Organization: Why $\approx$ 32 registers?

Stack and accumulator architectures.

Memory/Memory, Register/Memory.

Addressing Modes: What they do, which ones to include.

Register, Immediate, Direct, Register Deferred (Register Indirect), Displacement, Indexed, Memory Indirect, Autoincrement, Autodecrement, Scaled.

Control Transfer Instructions: Types, when to use.

- Branch, Jump, Jump & Link, Call, Return
- Format of displacements in instruction.
- Specification of condition: condition code registers, integer registers, loop counter.
- Delayed and predicated instructions; prediction hints.
- Instruction Coding.
  - Fixed-length, variable-length, and bundled instructions.
  - Splitting of opcode field (as in DLX type-R instructions).
- ISA Classifications: RISC, CISC, VLIW, Stack, Accumulator

Synthetic Instructions

DLX

Classification: RISC

Goals: ISA should allow simple, high-speed implementation.

Instruction types.

Know how to read and write DLX programs.

Chapter-3 (Static Scheduled) DLX Implementations

Unpipelined Implementation

Pipelined Implementations

- Basic (3-cycle branch penalty).
- One-cycle branch penalty.
- Bypassed.

Hazard Definitions

For a Given Pipelined Implementation

Show pipeline execution diagrams.

Show register contents at any cycle.

Determine control hardware.

Determine CPI.

Interrupts, Exceptions, Traps

Difference between interrupt, exception, trap.

Causes of exceptions, role of handler.

Privileged Mode.

Pipeline activity leading to execution of handler.

Vectored traps (using trap table).

Precise exceptions, achieving with floating-point operations.

Long Latency Operations

Types of operations. (Floating point and maybe load.)

Degree of pipelining: Initiation interval, latency.

Detecting functional unit structural hazards.

Detecting WB structural hazards: reservation register.

Detecting and handling RAW hazards: ID-stage v. pre-WB stall.

Handling WAW hazards.

Loop Unrolling