PURDUE
UNIVERSITY

CS250: Computer Architecture
Review

Prof. Chris Clifton

April 23-27, 2007

Grading

- 10%: Written assignments (equally weighted, lowest score dropped)
- 33%: Labs (equally weighted, lowest score dropped)
- *16%: Midterm Exam*
- 5%: Project 1
- 12%: Project 2
- 22%: Final Exam
- 2%: Evaluation of instructors based on in-class/lab contributions, discussions, and overall performance

Sea Change in Architecture (Patterson 2002)

Old Conventional Wisdom	*New Conventional Wisdom*
• Reliable internally, errors at pins	• <65NM soft+hard error rates
• Power free, transistors expensive	• Transistors free, power expensive
• Multiples slow, loads fast	• Loads slow, Multiplies fast – 200 cycles to DRAM, 4 for FP mult
• Instruction-level parallelism	• Diminishing returns from ILP
• 2*/1.5 years improvement	• 2*/5 years improvement
• Uniprocessor/Cluster	• Multi-core

Digital Hardware

- Transistors, RTL and TTL logic
  - Recognize/construct basic gates
- Gates
  - And/or/xor/not
  - Schematic representation
  - Multi-input and/or
- Flip-Flops
  - Recognize/understand feedback circuits
  - Construct simple circuits (e.g., S-R latch)

Digital Logic

- Truth Table
  - Karnaugh Map
- Boolean Algebra
  - Basic properties
  - DeMorgan's Theorem
- Understanding Feedback / Clocked Circuits

Data Representation

- Binary, Hex, Decimal
- Integer Representations
  - Unsigned
  - Approaches to signed
  - Implications for arithmetic
- Character Representations
  - ASCII chart might be handy
- IEEE Floating Point
  - <sign> × $2^{<8 \text{ bit exponent}>-127}$ × 1.<23-bit mantissa>
- Recognizing in Memory Dump

Processor Organization

- Von Neumann Architecture
  - Processor/memory
  - Stored Program Computer
- CPU
  - Register / ALU / controller
  - Microcode
  - Processor modes / protection

CPU

- ALU
  - Operations
- Registers
  - Banks, windows, special purpose
- Instruction processing
  - Fetch/Execute Cycle
  - Pipelines
- Instruction Set
  - Arithmetic, Logical, Memory, Control Flow, Floating Point, Processor Control

Assembly Language

- Instruction format
  - [Label:]
  - Instruction
  - Operands
    - Register, Immediate, Memory
- Addressing Modes
  - Specified vs. Computed
  - Register / Register+offset
  - Direct vs. Indirect

Assembly Language (cont.)

- Flow of control
  - Branch vs. jump
  - Conditional branches and condition codes
- Subroutines
  - Argument passing
  - Function results
  - Return address handling

Assembly Language (cont.)

- Data
  - Data vs. code sections
  - Alignment issues
- Macros
- Microcoded Architectures
  - “Instruction set” directly mapped to hardware
  - Horizontal vs. Vertical microcode

Memory

- Types of memory / hardware issues
  - Static vs. dynamic ram
  - Performance characteristics
- Addressing
  - Processor vs. Memory word
  - How this affects alignment
- Memory Controller
  - Banks / interleaving

Virtual Memory

- Virtual vs. Physical address
  - Base/bound
  - Page table
  - Translation Lookaside Buffer
- Paging / disk storage
- Cache and interaction with Virtual Memory
- Performance
  - Estimating performance of multi-level memory

I/O Devices and Drivers

- Interface types
  - Parallel
  - Serial
- Bus Architectures
  - Fetch/store operations
  - Memory-mapped I/O
- Interrupts vs. Polling
- Buffers

Parallelism

- Parallel architectures
  - Multicore
  - Multiprocessor
  - Cluster
- Pipelining
  - Instruction-level
  - As a paradigm for using parallel architectures
- Performance Assessment

Exam Hints

- Be There!
  - 4/30 19:00-21:00 SC239
- Open Book / Open Note
  - No electronic aids
- Time will be tight
  - You won't be able to look everything up – need to know it
  - **Don't get stuck – do what you know well first!**
- If something doesn't make sense, speak up
- Read last year's final exam at the web site