

BROWN

Computer System Design Lecture 4: More Trends, Instruction Set Introduction

Prof. R. Iris Bahar
EN164
February 2, 2007

Reading: Chapter 1, section 1.4-1.5, Appendix B, section B.1-B.2

BROWN


Reminder

- I am planning to hand out information for lab 1 next week.
- Please have a lab partner selected.
- If you haven't found a partner yet, let me know at the end of class and I will try and hook you up with someone.

EN164
Lecture 4-2

BROWN

Growing the Silicon Ingot



EN164
Lecture 4-3

BROWN

Desktop Prices

Vendor	Model	Processor	Clock speed MHz	Price
Dell	Precision 380	Intel Pentium 4 Xeon	3.8GHz	\$3346
HP	ProLiant BL25p	AMD Opteron 252	2.6GHz	\$3099
HP	ProLiant ML350 G4	Intel Pentium 4 Xeon	3.4GHz	\$2907
HP	Integrity rx2620-2	Itanium 2	1.6GHz	\$5201
Sun	Java Workstation W1100z	AMD Opteron 150	2.4GHz	\$2145

- All systems have similar configurations
 - Price variations due to expandability, disks/memory/processor/OS configuration, and commoditization
- On average, the processor is less than 10% of the total cost of the system
- Around 70% of the cost is due to the disk storage

EN164
Lecture 4-4

BROWN

Server Prices

System	CPUs	Price
IBM eSeries p5 595	64 IBM POWER 5	\$16.7 M
IBM eSeries p5 595	32 IBM POWER 5	\$8.4 M
HP Integrity rx5670 Cluster	64 Intel Itanium 2	\$6.5 M
Dell PowerEdge 2800	1 Intel Xeon	\$39 K
HP ProLiant ML350	1 Intel Xeon	\$28 K

- Generally cost much more than desktop systems
- Still, the processor accounts for less than 25% of the system, on average
- Most of the cost is still in the disk storage (~50%)

EN164
Lecture 4-5

BROWN

Processor Technology Trends

- Shrinking of transistor sizes: 250nm (1997) → 130nm (2002) → 90nm (2006) → 65nm (2008) → 45nm (2010)?
- Transistor density increases by 35% per year and die size increases by 10-20% per year
 - How are the extra devices being utilized?
- Transistor speed improves linearly with size (complex equation involving voltages, resistances, capacitances)
 - How does this affect cycle time?
- Wire delays don't scale at the same rate as logic delay
 - The Pentium 4 has pipeline stages for wire delays

EN164
Lecture 4-6

BROWN

Technology Trends

- DRAM density increases by 40-60% *per year*, but latency has only reduced by 33% in *10 years*
 - Leads to the “memory wall”
 - Bandwidth improves twice as fast as latency decreases to help compensate for this discrepancy
- Disk density, until recently, improved by 100% every year. Now it’s more like 60%. 50-100 times cheaper per bit than DRAM.
- Networks: primary focus on bandwidth; 10Mb → 100Mb in 10 years; 100Mb → 1Gb in 5 years

EN164
Lecture 4-7

BROWN

Power Consumption Trends

- Dynamic power $P_{dyn} = V_{dd}^2 \times C_{eff} \times f$

total capacitance of switching nodes
- Capacitance per transistor and voltage are decreasing, but number of transistors and frequency are increasing at a faster rate
- Leakage power is also rising and will soon match dynamic power
- Power dissipation is already between 100-150W in high-performance processors today

➔ Many architectural “enhancements” may become infeasible once power constraints are considered

EN164
Lecture 4-8

BROWN

A Few Key Points...

- Complexity-effective design is important:
 - ➔ a complex design takes longer to build, verify, and consumes more power
- Software cost should be taken into account while evaluating a system’s cost-performance
- Looking at power-performance of a single component can be misleading
- Don’t use CPI or IPC alone when comparing different ISAs
- Don’t rely on peak performance metrics or on results obtained with synthetic benchmarks

EN164
Lecture 4-9

BROWN

The Effect of Clock Speed

Benchmark	Relative Performance (Pentium 4 vs Pentium III)
SPECintRate	~1.25
SPECintRate	~1.75
Multimedia	~1.45
Game benchmark	~1.40
Web benchmark	~1.40

- Even with the same instruction set, performance does not closely track clock speed
 - depends on the benchmark set and processor functionalities
- Even within the same processor family, performance improvements are slower than clock speed improvements

© 2009 Elsevier Science (USA). All rights reserved.
Relative performance comparison between 1GHz Pentium III and 1.7GHz Pentium 4

EN164
Lecture 4-10

BROWN

ISA for Different Segments

- Instruction sets for all three segments are very similar
- **Desktops:** equal emphasis for integer and FP applications, with little regard for code size and power
- **Servers:** little need for high floating-point performance
- **Embedded:** emphasis on low cost and power
 - code size is important, floating-point may be optional
- Desktops and embedded also care about multimedia applications
 - use special media extension instructions
 - use some of these in lab 1

EN164
Lecture 4-11

BROWN

Anatomy of a Modern ISA

- Operations
 - simple ALU op’s, data movement, control transfer
- Temporary Operand Storage in the CPU
 - Large General Purpose Register (GPR) File
- Number of operands per instruction
 - triadic $A \leftarrow B \text{ op } C$
- Operand location
 - load-store architecture with register indirect addressing
- Type and size of operands
 - 32/64-bit integers, IEEE floats
- Instruction-to-Binary Encoding
 - Fixed width, regular fields

Exceptions: Intel x86, IBM 390

EN164
Lecture 4-12

BROWN

Machine Instructions

- Language of the Machine
- Lowest level of programming that directly controls the hardware
- Assembly instructions are *symbolic* versions of machine instructions
- More primitive than higher level languages
- Very restrictive
- Programs are stored in memory, where instructions can be fetched and later executed

EN164
Lecture 4-13

BROWN

Stored Program Concept

- Instructions are groups of bits
- Programs are stored in memory
 - to be read (or written) just like data

- Fetch & Execute Cycle
 - Instructions are fetched and put into a special register
 - Bits in the register "control" the subsequent actions
 - Fetch the "next" instruction and continue

EN164
Lecture 4-14

BROWN

Assembly Language vs. Machine Language

- Assembly provides convenient symbolic representation
 - much easier than writing down numbers
- Machine language is the underlying reality
- Assembly can provide "pseudo-instructions"
 - e.g., "move \$t0, \$t1" exists only in assembly
 - would be implemented using "add \$t0,\$t1,\$zero"
- When considering performance you should count real instructions

EN164
Lecture 4-15

BROWN

Accessing Internal Storage

- How should memory be accessed?
 - Directly or through registers?
- How should ALU operations be expressed?
 - Implicit or explicit operands?
 - compact or flexible?
- Example: How should we represent $C = A + B$?

Stack	Accumulator	Reg (reg-mem)	Reg (load-store)
Push A	Load A	Load R1, A	Load R1, A
Push B	Add B	Add R3, R1, B	Load R2, B
Add	Store C	Store R3, C	Add R3, R1, R2
Pop C			Store R3, C

- Registers: fast, exploit locality, reduced memory traffic, easier to re-order

EN164
Lecture 4-16

BROWN

Register Architectures

Type	Advantages	Disadvantages	Examples
Register-Register (0 mem, 3 ops) RISC	Simple, fixed-length, simple code-generation, easy pipelining and parallelism extraction	High instr count and code size	Alpha, MIPS, ARM, PowerPC, SPARC
Register-Memory (1 mem, 2 ops) CISC	Can access data without doing a load, small code size	One of the operands is destroyed, instr latency is variable	Intel 80x86, Motorola 68000
Memory-Memory (2 mem, 2 ops) or (3, 3) CISC	Most compact code size, doesn't waste registers	Variation in instr size (hard to decode), frequent memory accesses, variable instr latency	VAX

EN164
Lecture 4-17

BROWN

RISC vs. CISC ISAs

- Reduced Instruction Set Computer (RISC): by using a few simple instruction primitives, the hardware is simpler
 - easy to extract parallelism
 - easy to effect high clock speeds
- Complex Instruction Set Computer (CISC): if you do it in hardware, it's fast → hence, implement every functionality in hardware
 - Rich instruction set
 - Complex decoding
 - Complex analysis to identify dependences
 - Danger is a slower cycle time and/or a higher CPI
 - Goal is to reduce number of instruction executed
 - *Why is this important??*

EN164
Lecture 4-18

Characteristics of RISC ISAs

- Common characteristics of all RISCs
 - Single cycle issue
 - Small number of fixed length instruction formats
 - Load/store architecture
 - Large number of registers
- Additional characteristics of most RISCs
 - Small number of instructions
 - Small number of addressing modes
 - Fast control unit
- Virtually all new instruction sets since 1982 have been RISC based

Example Instruction Sets

- MIPS
 - Popular example of a RISC instruction set
 - Relatively easy to learn and use
- X86
 - CISC instruction set developed at Intel (1st version in 1978)
 - Why is it still popular today, despite the trend to RISC?
- How is Intel able to incorporate RISC characteristics into its processor design while still implementing the X86 ISA?