Computer Architecture: A Science of Tradeoffs

Outline

The ISA

The Microarchitecture

The System level

The ISA
(visible to the software)

Dynamic/Static Interface

What at compile time, what at run time

Rich instruction set vs. simpler instruction set

Fixed-length, Uniform Decode vs. …

Condition codes vs. …

Load/Store vs. …

Help for the Programmer vs. help for the uarchitect

- ***Addressing modes***
- ***Data types***
- ***Unaligned accesses***

The ISA (continued)

Hardware interlocks vs. …

VLIW vs. …

0,1,2,3 address machine

Compatibility vs. a new ISA

Precise exceptions vs. …

The Microarchitecture (under the hood)

- *CPI vs. cycle time (or, IPC vs. frequency)*
- *in-order vs. out-of-order execution*
- *Speculate vs. stand around and wait*
- *Issue-width*
- *ASIC vs. programmed control*
- *Use of chip real estate*
  - *Better branch predictor*
  - *Accelerators*
  - *Microcode*
- *Pipeline depth*
- *Cache structures*

Microarchitecture (continued)

Fault tolerance

- ***Yesterday: Tandem (Two cores in lock step)***
- ***Tomorrow: repeat to remove soft errors (or two cores in lock ..)***

On-chip latency – near neighbor communication

Branch target buffer – saves a bubble on taken branch

Partitioning – until when?