The Architecture of Multiprocessor Systems on a Chip

Trevor Mudge

Electrical Engineering & Computer Science

The University of Michigan
Ann Arbor

Outline

- Summary of current “mainstream” computer architecture research
- Current focus of CA research
- Relevant to SOC community
- How can the two communities exchange ideas?

Summary of current “mainstream” computer architecture research

Computer Architecture Research

- A personal view – can't speak for the rest
- Milieu: ISCA, MICRO, HPCA, ASPLOS, PACT, ICS, PLDI ...
- Themes:
  - Performance
  - Performance
  - Performance

Themes

- Performance
  - presumption of unlimited transistor budgets
  - benchmarks – SPEC2000
  - cycle-accurate simulators
- CPU centric
  - processor – memory rather than memory-disk
  - sequential model of processing
- Generality
  - general purpose processing
  - “integer” codes
- Heterogeneity
  - identical processors
  - SMP, CMP, etc

Audience

- Intel, IBM, Sun, Motorola, AMD, Compaq
- Desktops and servers
  - Wintel
  - Linux servers
- Increasingly smaller part of computing and communications
  - embedded ahead by dollar volume
  - unit sales ahead long ago

Hypothesis

- Innovation (in computers) follows low power and volume
  - Mainframes
  - Minicomputers
  - Microprocessors
  - *Hand help communicators*
    - 3G phone – the next desktop
    - merged functionality – wireless mobile computing
- Each step characterized by:
  - Reinventing earlier era
  - Adding its own inventions

Corollary

- Focus of innovation in computer architecture and systems will move away from the Wintel desktop to embedded computing
- Easy bet because everything else is embedded ☺

Idiosyncrasies

- Computations without I/O
- Interrupt structures missing
  - Context switching as a result of multitasking/threading
- Real-time not a priority
- Functionality limited
  - + - * /
  - special purpose usually not examined
    - Graphics
    - NIC, etc
- Time-to-market not a problem - Itanium
- DSPs rarely discussed
- x86 (IA32) rarely discussed – MIPS

Current Focus of CA Research

Research Areas

- Memory hierarchy performance
  - Caches
  - New memory interfaces
  - Raid a special case
- Branch prediction
  - Value prediction
- Instruction level parallelism
  - Pipelining
  - Multi-issue
    - Superscalar
    - VLIW

Research Areas (cont.)

- Compiler-Architecture trade-offs
  - In hardware or software?
- Technology limits – wirelength
  - Often misunderstood
  - Si technology is moving faster than ever
- Small Multi-Processors - SMP
  - Memory (cache) coherence
  - Models of consistency

Past and Future

- Past
  - Supercomputers
    - only one manufacturer ever profitable
    - national flagships
  - Massively parallel machines
    - cube machines
    - programming limits
    - few customers
  - Interconnection networks
    - cross-bars
    - nlogn nets
    - multiple buses

Past and Future

- Future
  - Power/energy consumption
  - Simultaneous Multi-Threading
  - Memory gap
  - Binary translation
    - Transmeta – cycle simulator history
    - Java machines – comes in various forms
  - Servers – SMP, CMP
  - Network processors

Relevant to SOC Research

Power/Energy Consumption

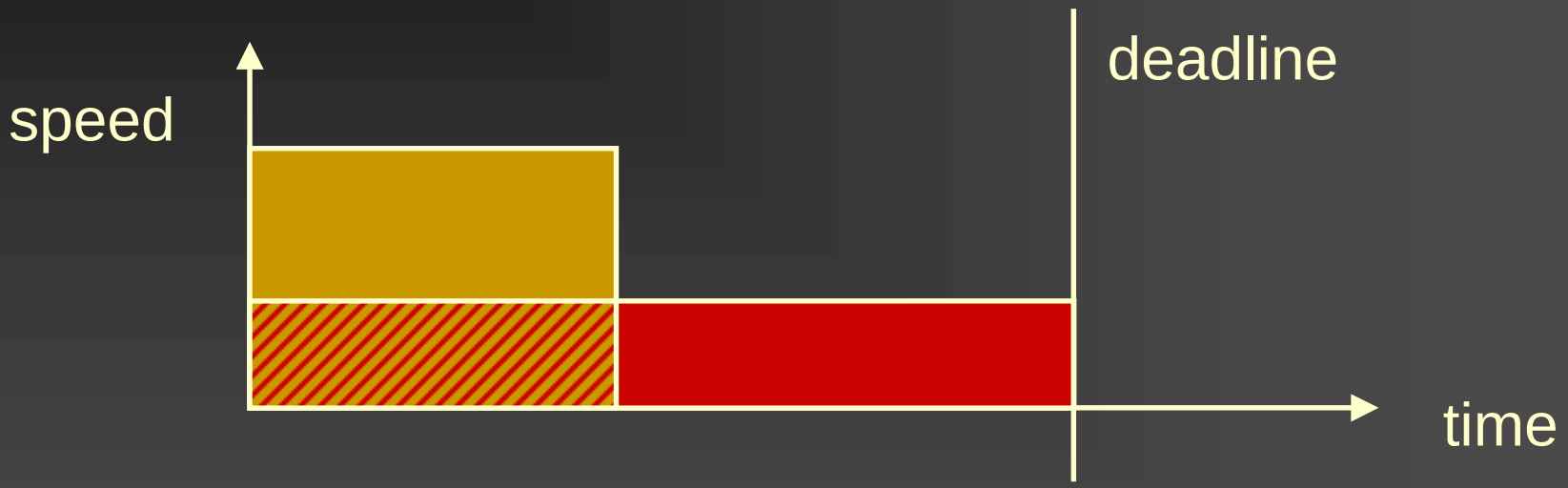
- CA research is late to this party
- Is the leading restriction to performance now
  - solutions are coming fast
- Push from Intel
  - not just portables
- ISP - heavy duty factory
  - ~2,000,000 Watts
  - lead cost driver is power ~25%
    - stats from Intel (Deo Singh) talk at CoolChips Tutorial

Power: Basic Ideas for Architects

- Power = $k\ A\ V^2\ C\ f\ +\ V\ I_{leak}$
  - A = activity level - how many nodes toggled
  - V = power supply voltage
  - C = capacitance of signal lines
  - f = toggle frequency
  - Why not reduce V to a small value?
  - $f_{max} \propto (V - V_t)^2/V$
    - can't simply reduce V
  - $I_{leak} \propto \exp(- qV_t/kT)$
    - future problem – 0.10 um and below
- Corollary: parallelism is good

Further Corollary: Just-in-time Computation

- Red profile saves saves power



- Useful when process is rate determined
  - Mpeg requires 30 frames / second and no more

Observation: Idle Time in Desktop Applications

Application	Idle Time
Windows Media Player - AVI	73.50%
Windows Media Player - MP3	92.80%
QuickTime - Video	44.40%
QuickTime - MP3	92.80%
HotJava	78.00%
Internet Explorer	90.50%
Cloudscape	29.70%
Image/J	42.20%
PowerTranslator	29.40%

*Done with Kris Flautner UM and Rich Uhlig Intel Microprocessor Research Lab using a 4-processors Pentium III system. See: K. Flautner, S. Reinhardt, and T. Mudge. Thread-level parallelism and interactive performance of desktop applications. ASPLOS-IX, Nov. 2000, pp. 129-138.

- How do we take advantage of this?
  - Dynamic voltage scaling work

Trends Against Low Power

- Speculation for high performance
  - branch prediction
  - trace caches
  - prefetching
  - runahead
  - etc.
- Interpretation more popular again
  - JVMs
  - can compilers help us with JITs?
  - can hardware assists help?

Next Challenge: Leakage

- $I_{leak} \propto \exp(- qV_t / kT)$
  - the trend is smaller devices
  - need a lower V to avoid large electric fields
  - limits V_t — a 10-15% decrease doubles I_{leak}
- Leakage current will dominate power
  - activity based solutions will not work
  - further reduced as we see less head room for V
- Dual (multiple) V_t
  - low (fast) V_t only on critical paths
  - shield caches with low V_t and L1 with high (slow) V_t

Activity-based Approach Breaks Down

- Dual thresholds, etc.
- Turn off unused subsystems
- Loss of state
- Selective state savings to speed up restart
  - fast vs slow flip-flops
  - caching

Heterogeneous Multis for SOCs

- Functional multiprocessing makes power management easier
  - particularly leakage
- Dynamic voltage scaling useful
  - not really a CA development
- DSP tradition of slower clocks with higher memory bandwidths – multiple sources
  - allows lower V
  - implicit just-in-time power?

Power & SOCs

- Reduce chip I/O
  - Improved packaging can diminish this
  - MCM/flip-chip etc
- Ideal for embedded DRAM
  - 3 – 6x improvements in density over SRAM
    - more cache and other memory structures
  - not that slow – < 10 nS
    - why not match to cpus – don’t sell GHz

Simultaneous Multi-Threading

- Makes sense for complex high end processors – Alphas, P4s
  - thread management can be absorb into out of order logic
    - Alpha team claim 2% increase in chip area
- Intra process threads don’t show much parallelism
  - code doesn’t use threads for parallelism
    - chicken and egg problem – easing
    - data parallel a different story – software needs rewrite
    - make an argument for response time
- Independent threads – multiprogramming
  - servers – makes sense

Memory Gap & SOCs

- Memory technology focused on density
  - 4x per two years
- Processor technology focused on speed
  - 2x per two years
- Partly an artifact of MHz (now GHz)
  - desktops sold on clock speed – 1.7 GHz Pentium4

New Memory Technologies

- Rambus, DDR SDRAM
  - Capitalize on the inherent bandwidth of RAM
- Suitable for streaming data
  - Bandwidth rather than latency
- Obvious candidate for
  - Video
  - Networking

Binary Translation & SOCs

- Long tradition
  - IBM emulation of 709 on the 360
  - Recently
    - FX32 – Alpha-to-x86
      - strong technology – price didn't make sense
    - Transmeta's morphing
      - on-the-fly translation
    - ARM's Jizelle
      - microcode support
- ISA CPU-to-DSP to leverage legacy code

Java & SOCs

- Endorsed by Nokia
  - Jizelle etc.
- Java processors have not been a success
  - microJava & picoJava
- Real-time Java
  - notion of time
  - garbage collection

Server Technology & SOCs

- SMP - small multiprocessor technology
  - extensive work on coherence
    - memory consistency models - weak, strong etc.
- Small can be as large as 250
  - Interconnect becomes critical
    - “factored” crossbars
- CMP – chip multiprocessor technology
  - shared cache
  - embedded DRAM makes this interesting

Why Multiprocessors?

- Better off using a uniprocessor for programmability
  - cost and physical limit
- Power considerations suggest MP
  - Leakage may reverse this view
- Mainstream world
  - Processors developed for uniprocessors can be naturally used for servers
    - independent job streams
    - database queries

Multiprocessors (cont.)

- Less success for parallel processing
  - programming
  - coherence scaling
  - but data parallel works
  - heterogeneous works too
- Key enabler
  - software
  - development environments for SOCs
- Our research
  - SimpleScalar for SOCs – SRC project
  - first example – ARM + C30

Conjecture: Convergence Architectures

- How much heterogeneity can be covered with a general purpose processor
  - future performance
  - future power
- Programming model is known
  - extensions necessity
    - MAC, mmx, bit ops, etc.
    - compiler support – Tensilica model
- There will always be a cost point favoring some special coprocessors
- IO structures will differentiate