Multicore Computing - Evolution

Performance Scaling

Source: Shekhar Borkar, Intel Corp.

ECE 4100/6100 (2)
Intel

- Homogeneous cores
- Bus based on chip interconnect
- Shared Memory
- Traditional I/O

IBM Cell Processor

Heterogeneous MultiCore

- High speed I/O
- High bandwidth, multiple buses
- Classic (stripped down) core
- Co-processor accelerator

Source: Intel Corp.

Source: IBM
AMD Au1200 System on Chip

PlayStation 2 Die Photo (SoC)
Multi-* is Happening

### Cores and Logical Thread Roadmap

**Current Platforms**

<table>
<thead>
<tr>
<th>Year</th>
<th>Processor</th>
<th>Cores</th>
<th>Threads</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005</td>
<td>Montecito</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2006+</td>
<td>Dispose</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

**Future**

<table>
<thead>
<tr>
<th>Year</th>
<th>Processor</th>
<th>Cores</th>
<th>Threads</th>
</tr>
</thead>
<tbody>
<tr>
<td>2006+</td>
<td>Dispose</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

**MP Servers**

<table>
<thead>
<tr>
<th>Year</th>
<th>Processor</th>
<th>Cores</th>
<th>Threads</th>
</tr>
</thead>
<tbody>
<tr>
<td>2006</td>
<td>Sunni</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2008</td>
<td>Sunni</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

**Source: Intel Corp.**

**Intel’s Roadmap for Multicore**

- Drivers are
  - Market segments
  - More cache
  - More cores

**Source: Adapted from Tom's Hardware**
Distillation Into Trends

- **Technology Trends**
  - What can we expect/project?

- **Architecture Trends**
  - What are the feasible outcomes?

- **Application Trends**
  - What are the driving deployment scenarios?
  - Where are the volumes?

Technology Scaling

- 30% scaling down in dimensions $\rightarrow$ doubles transistor density

- Power per transistor
  - $V_{dd}$ scaling $\rightarrow$ lower power

- Transistor delay $= C_{gate} \frac{V_{dd}}{I_{SAT}}$
  - $C_{gate}$, $V_{dd}$ scaling $\rightarrow$ lower delay
**Fundamental Trends**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>High Volume</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>8</td>
</tr>
<tr>
<td>Integration Capacity</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>Delay (ps/layer)</td>
<td>0.7</td>
<td>0.7</td>
<td>1.7</td>
<td>0.7</td>
<td>1.7</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Energy/Logic Op</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Metal Layers</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>RC Delay</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td></td>
</tr>
<tr>
<td>Integration Capacity</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>Metal Layers</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>RC Delay</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td>Reduce slowly towards 2-2.5</td>
<td></td>
</tr>
</tbody>
</table>

Source: Shekhar Borkar, Intel Corp.

**Moore’s Law**

- How do we use the increasing number of transistors?
- What are the challenges that must be addressed?

Source: Intel Corp.
Impact of Moore’s Law To Date

- Increase Frequency ➔ Deeper Pipelines
- Increase ILP ➔ Concurrent Threads, Branch Prediction and SMT
- Push the Memory Wall ➔ Larger caches
- IBM Power5

Source: IBM

Manage Power ➔ clock gating, activity minimization

Shaping Future Multicore Architectures

- The ILP Wall
  - Limited ILP in applications
- The Frequency Wall
  - Not much headroom
- The Power Wall
  - Dynamic and static power dissipation
- The Memory Wall
  - Gap between compute bandwidth and memory bandwidth
- Manufacturing
  - Non recurring engineering costs
  - Time to market
The Frequency Wall

- Not much headroom left in the stage to stage times (currently 8-12 FO4 delays)
- Increasing frequency leads to the power wall


Options

- Increase performance via parallelism
  - On chip this has been largely at the instruction/data level

- The 1990's through 2005 was the era of instruction level parallelism
  - Single instruction multiple data/Vector parallelism
    - MMX, SSIMD, Vector Co-Processors
    - Out Of Order (OOO) execution cores
    - Explicitly Parallel Instruction Computing (EPIC)

- Have we exhausted options in a thread?
The ILP Wall - Past the Knee of the Curve?

Performance

- Scalar In-Order
- Moderate-Pipe Superscalar/OOO
- Very-Deep-Pipe Aggressive Superscalar/OOO

“Effort”

Made sense to go Superscalar/OOO: good ROI

Very little gain for substantial effort

Source: G. Loh

The ILP Wall

- Limiting phenomena for ILP extraction:
  - **Clock rate**: at the wall each increase in clock rate has a corresponding CPI increase (branches, other hazards)
  - **Instruction fetch and decode**: at the wall more instructions cannot be fetched and decoded per clock cycle
  - **Cache hit rate**: poor locality can limit ILP and it adversely affects memory bandwidth
  - **ILP in applications**: serial fraction on applications

- Reality:
  - Limit studies cap IPC at 100-400 (using ideal processor)
  - Current processors have IPC of only 1-2

Source: G. Loh
The ILP Wall: Options

- Increase granularity of parallelism
  - Simultaneous Multi-threading to exploit TLP
    - TLP has to exist → otherwise poor utilization results
  - Coarse grain multithreading
  - Throughput computing

- New languages/applications
  - Data intensive computing in the enterprise
  - Media rich applications

The Memory Wall

“Moore’s Law”

Processor-Memory Performance Gap: (grows 50% / year)
The Memory Wall

- Increasing the number of cores increases the demanded memory bandwidth
- What architectural techniques can meet this demand?

The Memory Wall

- On-die caches are both area intensive and power intensive
  - StrongArm dissipates more than 43% power in caches
  - Caches incur huge area costs
- Larger caches never deliver the near-universal performance boost offered by frequency ramping (Source: Intel)
The Power Wall

\[ P = \alpha CV_{dd}^2 f + V_{dd}I_{st} + V_{dd}I_{\text{leak}} \]

- Power per transistor scales with frequency but also scales with \( V_{dd} \)
  - Lower \( V_{dd} \) can be compensated for with increased pipelining to keep throughput constant
  - Power per transistor is not the same as power per area \( \rightarrow \)
  - Power density is the problem!
  - Multiple units can be run at lower frequencies to keep throughput constant, while saving power

Leakage Power Basics

- Sub-threshold leakage
  - Increases with lower \( V_{th} \), \( T \), \( W \)
  \[ I_{\text{sub}} = K W e^{-V_{\text{th}}/nkT} (1 - e^{-V/kT}) \]

- Gate-oxide leakage
  - Increases with lower \( T_{ox} \), higher \( W \)
  - High K dielectrics offer a potential solution
  \[ I_{\text{ox}} = K_w W \left( \frac{V}{T_{ox}} \right)^2 e^{-a T_{ox}/V} \]

- Reverse biased pn junction leakage
  - Very sensitive to \( T \), \( V \) (in addition to diffusion area)
  \[ I_{pn} = J_{\text{leakage, p+\text{n}}} (e^{qV/kT} - 1) A \]
The Current Power Trend

Source: Intel Corp.

Improving Power/Performance

\[ P = \alpha C V_{dd}^2 f + V_{dd} I_{st} + V_{dd} I_{leak} \]

- Consider constant die size and decreasing core area each generation = more cores/chip
  - Effect of lowering voltage and frequency \( \rightarrow \) power reduction
  - Increasing cores/chip \( \rightarrow \) performance increase

Better power performance!
Accelerators

TCP/IP Offload Engine

Opportunities: Network processing engines
MPEG Encode/Decode engines, Speech engines

Source: Shekhar Borkar, Intel Corp.

Low-Power Design Techniques

- Circuit and gate level methods
  - Voltage scaling
  - Transistor sizing
  - Glitch suppression
  - Pass-transistor logic
  - Pseudo-nMOS logic
  - Multi-threshold gates

- Functional and architectural methods
  - Clock gating
  - Clock frequency reduction
  - Supply voltage reduction
  - Power down/off
  - Algorithmic and software techniques

Two decades worth of research and development!
The Economics of Manufacturing

- Where are the costs of developing the next generation processors?
  - Design Costs
  - Manufacturing Costs

- What type of chip level solutions is the economics implying?

- Assessing the implications of Moore’s Law is an exercise in mass production

The Cost of An ASIC

Example: Design with 80 M transistors in 100 nm technology

Estimated Cost - $85 M - $90 M

- Cost and Risk rising to unacceptable levels
- Top cost drivers
  - Verification (40%)
  - Architecture Design (23%)
  - Embedded Software Design
    - 1400 man months (SW)
    - 1150 man months (HW)
  - HW/SW integration

12 – 18 months

The Spectrum of Architectures

- **Customization fully in Hardware**
- **Design NRE Effort**
- **Increasing NRE and Time to Market**
- **Increasing NRE Effort**
- **Decreasing Customization**

Hardware Development:
- Custom ASIC
- Structured ASIC
- FPGA
- Polymorphic Computing Architectures
- Tiled architectures

Software Development:
- Fixed + Variable ISA
- Microprocessor

Synthesis

- LSI Logic
- Leopard Logic
- Xilinx
- Altera
- MONARCH
- SM, RAW, TRIPS
- PACT, PICOChip
- Tensilica
- Stretch Inc.

Interlocking Trade-offs

- **Memory**
  - Bandwidth
  - Latency

- **Power**
  - Leakage power
  - Dynamic power

- **ILP**
  - Instruction-Level Parallelism

- **Frequency**
  - Dynamic frequency

- Improving one property comes at the expense of the other
- We need new approaches to co-optimization!
Multi-core Architecture Drivers

- Addressing ILP limits
  - Multiple threads
  - Coarse grain parallelism \(\rightarrow\) raise the level of abstraction

- Addressing Frequency and Power limits
  - Multiple slower cores across technology generation
  - Scaling via increasing the number of cores rather than frequency
  - Heterogeneous cores for improved power/performance

- Addressing memory system limits
  - Deep, distributed, cache hierarchies
  - OS replication \(\rightarrow\) shared memory remains dominant

- Addressing manufacturing issues
  - Design and verification costs
    \(\rightarrow\) Replication \(\rightarrow\) the network becomes more important!