Amending Moore’s Law for Embedded Applications

29 Sep 2004

Dr. Richard W. Linderman
Senior Scientist, Advanced Computing Architectures
AFRL Information Directorate
**Amending Moores Law for Embedded Applications**

**AFRL Information Directorate**

**See also ADM001742, HPEC-7 Volume 1, Proceedings of the Eighth Annual High Performance Embedded Computing (HPEC) Workshops, 28-30 September 2004.** The original document contains color images.
Contribution of Moore’s Law to Improvements of Embedded Systems

• **Price/Performance:** Gigaflops/$M affordability
• **Memory Capacity:** programming simplifications
• **Steep memory hierarchy:** programming inefficiencies and complexities
• **New flexibilities:** e.g. reconfigurable hardware
• **New complexities:** software and parallelism
• **Dramatic new system capabilities**
Extending the Affordability Trend

CHALLENGE:
Develop and Incorporate the Most Affordable Embedded Information Technology Available

APPROACH:
- Leverage Commercial Investments In Computer Architectures
- Develop Portable Embedded DoD Applications using middleware standards
- Leverage DARPA, and Other DoD Efforts In Emerging Architectures

Projection:
2007: 60 trillion flops/$M
2010: 360 trillion flops/$M
• **Price/Performance**: Gigaflops/$M affordability
• **Memory Capacity**: programming simplifications
• **Steep memory hierarchy**: programming inefficiencies and complexities
• **New flexibilities**: e.g. reconfigurable hardware
• **New complexities**: software and parallelism
• **Dramatic new system capabilities**
Airborne COTS HPC

25 Dual-Xeon Compute Nodes

- Dual GbE Network (80 MB/s per link)
- 1U Rack-mounted
- $87K approximate cost - 3.1 DP TFLOPS/$1M

Commodity Compute Node:
- Dual 2.66 GHz Intel Pentium 4 Xeon CPU
  - 512KB L2 Cache
  - 21.3 SP GFLOPS peak per node
- Dual GbE network interfaces
- 6 GB DDR SDRAM

Peak Performance:
- 532.5 GFLOPS (IEEE 32 bit)

Swathbuckler Cluster
(Coyote)
Contribution of Moore’s Law to Improvements of Embedded Systems

- **Price/Performance**: Gigaflops/$M affordability
- **Memory Capacity**: programming simplifications
- **Steep memory hierarchy**: programming inefficiencies and complexities
- **New flexibilities**: e.g. reconfigurable hardware
- **New complexities**: software and parallelism
- **Dramatic new system capabilities**
Can we preserve historical improvement rates?

Azimuth Processing

Seconds per Block

<table>
<thead>
<tr>
<th>Date</th>
<th>Seconds per Block</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aug 14, 03</td>
<td>4050</td>
</tr>
<tr>
<td>Aug 21, 03</td>
<td>2267</td>
</tr>
<tr>
<td>Aug 27, 03</td>
<td>2007</td>
</tr>
<tr>
<td>Aug 28, 03</td>
<td>641</td>
</tr>
<tr>
<td>Sep 02, 03</td>
<td>333</td>
</tr>
<tr>
<td>Sep 04, 03</td>
<td>322</td>
</tr>
<tr>
<td>Feb 01, 04</td>
<td>281</td>
</tr>
<tr>
<td>Feb 04, 04</td>
<td>135</td>
</tr>
<tr>
<td>Feb 27, 04</td>
<td>99</td>
</tr>
<tr>
<td>Mar 02, 04</td>
<td>71</td>
</tr>
<tr>
<td>Mar 10, 04</td>
<td>49</td>
</tr>
<tr>
<td>Mar 12, 04</td>
<td>43</td>
</tr>
<tr>
<td>Mar 25, 04</td>
<td>37</td>
</tr>
<tr>
<td>Mar 29, 04</td>
<td>25</td>
</tr>
</tbody>
</table>
Contribution of Moore’s Law to Improvements of Embedded Systems

• **Price/Performance:** Gigaflops/$M affordability
• **Memory Capacity:** programming simplifications
• **Steep memory hierarchy:** programming inefficiencies and complexities
• **New flexibilities:** e.g. reconfigurable hardware
• **New complexities:** software and parallelism
• **Dramatic new system capabilities**
Heterogeneous HPC Hardware

• 48 Nodes in 2 cabinets
• Server product leverage
• Each node with: Dual 2.2 GHz+ Processors
  – 4 Gbyte SDRAM
  – Myrinet 320 MB/sec Interconnect
  – 80 GB disk
  – 12 M gate Adaptive Computing Board
• 34 TOPs demonstrated
• Online FEB 2003 supporting HIE, TTCP and SBR projects
Contribution of Moore’s Law to Improvements of Embedded Systems

- **Price/Performance**: Gigaflops/$M affordability
- **Memory Capacity**: programming simplifications
- **Steep memory hierarchy**: programming inefficiencies and complexities
- **New flexibilities**: e.g. reconfigurable hardware
- **New complexities**: software and parallelism
- **Dramatic new system capabilities**
Software Complexity & Cost

- **Traditional Embedded SW Development**
  - Low Level Programming
  - Ad Hoc Approaches
  - Stand-alone, Static Implementations
  - Custom Systems
  - Little Code Re-use
  - V & V is labor intensive (e.g. F/A-22)

- **Result is**
  - Prolonged Design Schedules
  - Excessive Cost
  - Difficulty in Maintenance/ Upgrade/ Retrofit
  - Limits on Functionality

- **Looking Ahead**
  - Increased Functionality => Complexity
  - Move Towards Networking, Interconnecting Systems
  - Distributed Computation Models
  - Reconfigurability, Dynamic Modifications
  - Code Correctness/ Safety Concerns
  - Security
  - Shortage of Skilled Programmers

Advanced tools and techniques are needed to address increasing complexity!
Can we preserve historical improvement rates?

Of necessity, innovate in the broader architecture design space

slowing VLSI motivates more architectural exploration

Invest hardware to simplify software

Encapsulate sophisticated software to achieve performance with portability