Ronald Luijten – Data Motion Architect
lui@zurich.ibm.com

IBM Research - Zurich
14 March 2012

Co-design perspectives from a Data Motion Architect
acknowledgement

- I work with a great team
- Many of the charts in this preso are from Phillip Stanley-Marbell and Victoria Caparros
DCO team 3 year horizon

- Datacenter disruption ahead due to 5 Walls, SCM and 3D packaging
- The datacenter challenge is shifting from compute to data
  - MemcacheD; Purescale; Text Analytics
  - PByte scale memory servers needed
  - Single address space, not cache coherent
  - Small access granularity (few bytes)
  - Cache unfriendly access patterns
    - Large scatter / gather

- Need I/O architecture that satisfies above
  - Address translation bottleneck
  - Power (P9) I/O architecture mission

- Need new, holistic modeling methodology to allow design space exploration
Perspectives on Co-design

- Not co-optimization but holistic optimization is needed

- It was mentioned a few times yesterday:
  - need to holistically select physics model, numerical methods, algorithm, programming model and hardware

- Moore’s paradox was mentioned yesterday:
  - Cost of DRAM bit, BW becomes larger wrt. Cost of compute

- I mentioned this the last two SOS meetings: Data is the problem – compute has been solved

- We are developing a design space exploration environment, where aim to capture from algorithm to transistor based on first principles
  - High risk – high potential payoff endeavor

- We are establishing new insights between SW and HW (examples follow)

- For exascale, all components / building blocks are understood today – however, not how they are to be put together!
  - This will be dictated by energy constraints
“core is the most important part of the system”

- Do whatever you can to keep it busy
  - Utilization is paramount, don’t care how much area and energy is spent doing that.
  - Speculation is good – in case it works, you gain, in case it does not work, nothing is lost, right?

Moving Data is a sin

- Avoid data motion as much as possible
- Compute is free – put it wherever it needs to be done – even when utilization is almost zero

On Data Locality:

- Temporal
- Spatial
- Geographical
You get what you measure (Lord Kelvin)

- Corollary:
  Don’t expect to get what you don’t measure

- Key is to decide what to measure (seems trivial, right?)

"When you can measure what you are speaking about and express it in numbers, you know something about it, and when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory kind."
Ronald’s “More’s law” – the result of Moore’s law
Performance and Energy measurements in 3 ISAs

- **Candidate processors** for scale-out systems: Atom, PowerPC, ARM
  - Three hardware reference designs; **all processors implemented in 45nm CMOS**
  - All three systems running Linux distributions based on kernel 2.6.32

<table>
<thead>
<tr>
<th>Platform</th>
<th>General-Purpose</th>
<th>CPU</th>
<th>Cache Hierarchy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atom™ D510MO</td>
<td>2 x86-64</td>
<td>1.66 GHz</td>
<td>32 K/24 K L1 I/D (per core) and 1 M L2</td>
</tr>
<tr>
<td>Freescale™ P2020RDB</td>
<td>2 Power Architecture® e500</td>
<td>1.0 GHz</td>
<td>32 K/32 K L1 I/D (per core), 512 K shared L2</td>
</tr>
<tr>
<td>TI DM3730 (Beagleboard-xM)</td>
<td>1 ARM® Cortex™-A8</td>
<td>1.0 GHz</td>
<td>32 K/32 K L1 I/D, 256 K L2</td>
</tr>
</tbody>
</table>

**Common Properties:**
- Identical 4GB flash disk
- Lab-grade power meter, 1 measurement per second, sub-mA resolution
- Power measured for whole platform
Zero-Load System Whole-System Power Analysis

- Linux running, no apps – idle process
- Significant difference in zero-load power dissipation
  - > 3× difference in zero-load power dissipation, from lowest (3.8W), to highest (14.5W)
Workload

- 16 applications
  - Broad range of domains
  - Range from small data-analytics kernels with poor locality (KME), to large compute-intensive applications (ART, EQK)

<table>
<thead>
<tr>
<th>Application</th>
<th>Benchmark Suite</th>
<th>Dwarf</th>
</tr>
</thead>
<tbody>
<tr>
<td>kmeans (KME)</td>
<td>Phoenix MapReduce</td>
<td>Dense Linear Algebra</td>
</tr>
<tr>
<td>matrix multiply (MM)</td>
<td>—</td>
<td>Sparse Linear Algebra</td>
</tr>
<tr>
<td>pca (PCA)</td>
<td>—</td>
<td>Dense Linear Algebra</td>
</tr>
<tr>
<td>basicmath (BM)</td>
<td>MiBench</td>
<td>Sparse Linear Algebra</td>
</tr>
<tr>
<td>bitcount (BIT)</td>
<td>—</td>
<td>Sparse Linear Algebra</td>
</tr>
<tr>
<td>dijkstra (DIJK)</td>
<td>—</td>
<td>Graph Traversal</td>
</tr>
<tr>
<td>JPEG decode (JPGd)</td>
<td>—</td>
<td>Dense Linear Algebra, Structured Grids</td>
</tr>
<tr>
<td>ifft (IFFT)</td>
<td>—</td>
<td>Spectral Methods</td>
</tr>
<tr>
<td>qsort (QSRT)</td>
<td>—</td>
<td>Graph Traversal</td>
</tr>
<tr>
<td>rijndael decode (AES)</td>
<td>—</td>
<td>Combinational Logic</td>
</tr>
<tr>
<td>susan (SUS)</td>
<td>—</td>
<td>Dense Linear Algebra</td>
</tr>
<tr>
<td>164.gzip (GZIP)</td>
<td>SPEC CPU2000</td>
<td>Finite State Machine</td>
</tr>
<tr>
<td>175.vpr (VPR)</td>
<td>—</td>
<td>Backtrack/Branch-Bound</td>
</tr>
<tr>
<td>177.mesa (MESA)</td>
<td>—</td>
<td>MapReduce</td>
</tr>
<tr>
<td>179.art (ART)</td>
<td>—</td>
<td>Backtrack/Branch-Bound</td>
</tr>
<tr>
<td>183.equake (EQK)</td>
<td>—</td>
<td>Unstructured Grids</td>
</tr>
</tbody>
</table>

Instruction mixes:

Intel Atom D510 performance counters across benchmarks (Instr. mix percentiles; 3 repeated runs.)
Application-Driven Whole-System Power Analysis

Serial workload
- Single core, serial application launch
- Essentially tied between ARM (30.8kJ) and Atom (33.1kJ): 7.8% difference
- PPC: due to FPU
“Throughput” workload
- All 16 applications launched simultaneously; on Atom and PPC, utilize 2 cores
- PPC: has limited FPU; perf. limited by floating-point-intensive Equake benchmark

Highest-average-power system is the most energy-efficient
- Atom platform uses least energy (15.4kJ), vs. ARM (30.8kJ) and PPC (86.8kJ)
Stefan-Boltzmann Law: relates energy flux ($\Phi$) to temperature ($T$), area ($A$), and emissivity ($e$), Stefan-Boltzmann constant ($\sigma = 5.67 \times 10^{-8}\text{Wm}^{-2}\text{K}^{-4}$):

$$\Phi = A \cdot e \cdot \sigma \cdot T^4$$
Thermal Analysis: Hotspots

2D temperature map from **14k-element microbolometer array** (thermal radiation meter)

- **Temperature peaks**
  - Hottest components in all three platforms are not the processors
  - **ARM platform**: USB-Ethernet bridge;
  - **Atom platform**: I/O controller;
  - **PowerPC platform**: Ethernet switch subsystem
Thermal Analysis: Estimating Power Apportionment

- **ARM (DM3730)**: 3.3W
- **Atom (D510)**: 16W
- **PPC**: 13W

Diagram showing power distribution across different components for each platform.
Power wall analysis

Assumptions:

- Keep scaling as we have done in past couple of years
- Projection from 30 year trend until 2010
- ITRS roadmap on pins

Power wall analysis: Supply pins

Projection based on historical data
• Contradicts ITRS assumption
  1:1 for ASIC, 2:1 for CPUs – supply:signal)
Power—Interplay Between Power and Bandwidth: Pins

Assumptions:
• 17GB/s mem @ 155 pins (DDR3)
• 1 memory acc / 1k instr / 32 byte fetch
• ITRS pin projections for 2020

If we keep scaling as we have in past, we can support ~1000 threads per chip in 2020

Conclusions

- Energy: ISA does not matter – system design does
  - Packaging is *the* make or break dimension for exascale
- No ARM ISA energy advantage measured – contradictory to rumors
  - The u-server advantage lies in:
    - Simple core leaves room to integrate ‘other stuff’ onto same chip
    - Ethernet, USB, SATA, etc. integrated onto chip
    - Significant energy savings by avoiding chip-crossings (System-on-a-Chip)
- Need to understand workloads, data placement, access patterns in order to optimize system design
- Choice of metrics is key – exaflop as target is missing the point
- Need to understand energy efficient cache design
- Creating holistic design space exploration tool
- Breakthru Innovation needed in system design
  - Start from data – work your way back to processing
  - Putting 100’s of cores on single die most likely wrong design point for HPC
  - I predict accelerators will be key – not sure this will be current GPU thinking
papers (µServer research)