Targeted execution enabling increased power efficiency

Anirban Lahiri
Adya Shrotriya
Nicolas Zea

Technology Researchers

John Goodacre
Director, Program Management
ARM Processor Division

August 2009
Interesting System Configurations…

A: Within ARM MPCore™ Cluster
B: System Coherent
C: Accelerator Port Coupled
D: Loosely Coupled
Background

- Smooth transition between energy and performance levels
- Reduced loss due to leakage power as cores can be switched off
- Addresses the application performance diversity

Fig 1a: Power at Peak performance per Operating Voltage

Fig 1b: Power-Performance Diversity of Single Task Workloads

Fig 1c: Diversity of Multitask Workloads

Disclaimer: The plots are indicative of practical architectures and systems.
Analyzing Diversity

- Code compatibility (due to uniform ISA) ensures easy dynamic task migration (Fig 2a).
- Task migration for power efficiency based on required performance (Fig 2b). Example shows a set of tasks $T_1 - T_5$.

![Fig 2a: Single task migrating across cores over time](image)

- Prevents smaller tasks from corrupting high performance task execution. E.g. Task $T_1$ in Fig 2b.
- Important to further analyse temporal effects of SoC power.

![Fig 2b: Task migrations over time based on performance requirement in a Multitask Workload](image)
Methodologies Being Utilized

Flow 1 (High-level Simulation)
- System Generator
  - Simulator RV
    - Execution Trace
  - Cache Models
    - Cache Hit / Miss Ratio
  - OS - Profiler
    - OS Behaviour
  - On-board Execution
    - Execution Trace
  - Statistical Methods

Flow 2 (Hardware Emulation)
- ARMCC / GCC
  - C Code
    - App / OS
      - (Parallel / SMP ?)
  - Netlist
  - Simulation (PowerTheater)
    - Power per H/w Unit
    - Cache Hit / Miss Ratio
  - Using RVT

Flow 3 (Low-level Simulation)
- RTL
  - Emulator (Palladium)
  - Compare
  - Power Estimation Model
  - Accuracy
## Software Model Considerations

<table>
<thead>
<tr>
<th></th>
<th>Power Aware SMP</th>
<th>Big-Switch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of OS modification</td>
<td>Requires affinity to be driven by performance requirement</td>
<td>Potentially no changes required</td>
</tr>
<tr>
<td>Maximum power save</td>
<td>Can operate as big-switch too</td>
<td>Little and big core need performance continuum</td>
</tr>
<tr>
<td>Level of task diversity and peak performance</td>
<td>Enable better scalability</td>
<td>Limited to performance of single CPU</td>
</tr>
<tr>
<td>Implementation complexity</td>
<td>OS needs a speculative understanding of performance demands</td>
<td>Invisible to OS, operates similar to interrupt service routine</td>
</tr>
<tr>
<td>Management Responsibility</td>
<td>OS performance monitor</td>
<td>Application dependent</td>
</tr>
<tr>
<td>Flexibility</td>
<td>SMP / AMP designs</td>
<td>Single CPU only</td>
</tr>
</tbody>
</table>
## Summary Expectations

<table>
<thead>
<tr>
<th>Application Scenario</th>
<th>Power-Aware SMP Scheduled</th>
<th>Big-Switch</th>
<th>Big-Core Only</th>
<th>Little Core Only</th>
</tr>
</thead>
<tbody>
<tr>
<td>Big-Task (700MIPS)</td>
<td>520mW</td>
<td>Big-Core (500mW @ 0.8V)</td>
<td>Big-Core (500mW @ 0.8V)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>+ Little-Core (20mW Leakage)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Small-Task (350MIPS)</td>
<td>250mW</td>
<td>Little-Core (200mW)</td>
<td>Big-Core (500mW)</td>
<td>Little-Core (200mW)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 Big-Tasks + 3 Small Tasks</td>
<td>700mW</td>
<td>Big Core (750mW @ 1.1V)</td>
<td>Big Core (750mW @ 1.1V)</td>
<td></td>
</tr>
<tr>
<td>(1100MIPS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 Big-Tasks + 5 Small Tasks</td>
<td>950mW</td>
<td></td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>(1400MIPS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Operating Voltage (Volts)</th>
<th>0.8</th>
<th>0.9</th>
<th>1.0</th>
<th>1.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Big-Core MIPS at Peak Frequency</td>
<td>700</td>
<td>800</td>
<td>950</td>
<td>1100</td>
</tr>
<tr>
<td>Little-Core MIPS at Peak Frequency</td>
<td>350</td>
<td>400</td>
<td>450</td>
<td>500</td>
</tr>
<tr>
<td>Big-Core Power at Peak Frequency (mW)</td>
<td>500</td>
<td>575</td>
<td>600</td>
<td>750</td>
</tr>
<tr>
<td>Little-Core Power at Peak Frequency (mW)</td>
<td>200</td>
<td>250</td>
<td>300</td>
<td>350</td>
</tr>
</tbody>
</table>

Possible Power savings up to 50%
Performance enhancements up to 30% seen by reducing corruption of high performance tasks
Key to still understand the costs of migration
Thank you