Thermal Modeling, Analysis and Management of 2D Multi-Processor System-on-Chip

Prof. David Atienza Alonso
Embedded Systems Laboratory (ESL)
Institute of EE, Faculty of Engineering

ECOFAC 2010, Lannion, 29/03 – 2/04 2010

Outline

- MPSoC thermal modeling and analysis
- HW-based thermal management for MPSoCs
- SW-based thermal management for MPSoCs
- Conclusions

MPSoC Thermal Modeling Problem

- Continuous heat flow analysis
- Capture geometrical characteristics of MPSoCs
- Explore different packaging features and heat sink characteristics
- Time-variant heat sources
- Transistor switching depends on MPSoC run-time activity (software)
- Dynamic interaction with heat flow analysis

Very complex computational problem!
MPSoC Thermal Modeling
State-of-the-Art

- MPSoC Modeling and Exploration
  1. SW simulation: Transactions, cycle-accurate (~100 KHz)
     [Synopsys Realview, Mentor Primecell, Madsen et al., Angiolini et al.]
     At the desired cycle-accurate level, they are too slow for thermal analysis of real-life applications!
  2. HW prototyping: Core dependent (~50–100 MHz)
     [Cadence Palladium II, ARM Integrator IP, Heron Engineering]
     Very expensive and late in design flow, no thermal modeling, only used for functional validation of MPSoC architectures!

- Heat Flow Modeling:
  1. Software thermal/power models [Skadron et al., Kang et al.]
     Too computationally intensive and not able to interact at run-time with inputs from MPSoC components!

Orthogonalizing
MPSoC Thermal Modeling and Analysis

- MPSoC Modeling and Exploration
  1. SW simulation: Transactions, cycle-accurate (~100 KHz)
     [Synopsys Realview, Mentor Primecell, Madsen et al., Angiolini et al.]
     At the desired cycle-accurate level, they are too slow for thermal analysis of real-life applications!
  2. HW prototyping: Core dependent (~50–100 MHz)
     [Cadence Palladium II, ARM Integrator IP, Heron Engineering]
     Very expensive and late in design flow, no thermal modeling, only used for functional validation of MPSoC architectures!

- Heat Flow Modeling:
  1. Software thermal/power models [Skadron et al., Kang et al.]
     Too computationally intensive and not able to interact at run-time with inputs from MPSoC components!
Orthogonalizing
MPSoC Thermal Modeling and Analysis

Chip and Package Heat Flow Modeling

Model interface
- Input: power model of MPSoC components, geometrical properties
- Output: temperature of MPSoC components at run-time

Thermal circuit: 1st order RC circuit
- Heat flow = Electrical current; Temperature = Voltage
- Heat spreader and IC composed of elementary blocks

Temperature change
\[ C \Delta T = - G (I_k) + P_k \quad k = 1..m \]

power consumption vector
**Chip and Package Heat Flow Modeling**

- **Model interface**
  - Input: power model of MPSoC components, geometrical properties
  - Output: temperature of MPSoC components at run-time

- **Thermal circuit**: 1st order RC circuit
  - Heat flow ~ Electrical current
  - Heat spreader and IC composed of elementary blocks

**Thermal conductance matrix**

\[
\begin{bmatrix}
G_{1,2} & G_{1,3} \\
G_{2,1} & G_{2,3}
\end{bmatrix}
\]

**Temperature vector at instant \( k \)**

\[T_k = \text{C} \cdot t_k\]

**Thermal capacitance matrix**

\[
\begin{bmatrix}
C_1 & C_{1,2} \\
C_{2,1} & C_{2,2}
\end{bmatrix}
\]

**Circuit for non-linear thermal estimation**

\[C \cdot T_{k+1} = -G \cdot T_{k+1} + P_k; \quad k = 1..m\]

**SW Thermal Estimation Tool for MPSoCs**

- Creating linear approximation while retaining variable Si thermal conductivity:
  - Time step chosen small enough for convergence
  - Complexity scales linearly with the number of modeled cells (simulated on P4@ 3GHz)

\[C \cdot T_{k+1} = -G \cdot T_k + B; \quad k = 1..m\]

**Complexity scales linearly with the number of modeled cells (simulated on P4@ 3GHz)**
Case Study: HW 4-Core MPSoC

- MPSoC Philips board design:
  - 4 processors, DVFS: 100/500 MHz
  - Plastic packaging
- Software:
  - Image watermarking, video rendering
- Power values for 90nm:

<table>
<thead>
<tr>
<th>Element</th>
<th>Max Power (mW) 100 MHz</th>
<th>Max Power (mW) 500 MHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor</td>
<td>2.6x10^7</td>
<td>1.0x10^7</td>
</tr>
<tr>
<td>D-Cache</td>
<td>1.4x10^7</td>
<td>7.1x10^6</td>
</tr>
<tr>
<td>F-Cache</td>
<td>1.4x10^7</td>
<td>7.1x10^6</td>
</tr>
<tr>
<td>Pin Mem</td>
<td>9.8x10^6</td>
<td>2.75x10^6</td>
</tr>
<tr>
<td>AMBA</td>
<td>6.3x10^6</td>
<td>6.8x10^6</td>
</tr>
</tbody>
</table>

Average temperature of emulated 4-core MPSoC

Results: Thermal Validation 4-core MPSoC

- MPARM: Cycle-accurate SW architectural simulator
  - Complete power/thermal models tuned to Philips/IMEC figures
  - Simulations too slow: 2 days for 0.18 real sec (12 cells)
  - HW thermal emulation able to validate policies at run-time
    - Dynamic Voltage and Frequency Scaling (DVFS) based on thresholds

Emulation time 45 sec (128 cells)
**Results: Thermal Validation 4-core MPSoC**

- **MPARM**: Cycle-accurate SW architectural simulator
- Complete power/thermal models tuned to Philips/IMEC figures
- Simulations too slow: 2 days for 0.18 real sec (12 cells)
- HW thermal emulation able to validate policies at run-time
- Dynamic Voltage and Frequency Scaling (DVFS) based on thresholds

**Figure:** Very fast validation of MPSoC run-time thermal behavior and management.

**Outline**

- MPSoC thermal modeling and analysis
- HW-based thermal management for MPSoCs
- SW-based thermal management for MPSoCs
- Conclusions

---

**Temperature Management is Power Control under Thermal Constraints**

- Power consumption of cores determines thermal behavior
- Power consumption depends on frequency and voltage
- Setting frequencies/voltages can control power and temperature

**Figure:** Power consumption of cores and related temperatures.

---

**Optimization problem:** frequency/voltage assignment in MPSoCs under thermal constraints

- Meet processing requirements
- Respect thermal constraint at all times
- Minimize power consumption
**HW-Based Thermal Management State-of-the-Art**

- Static approach: thermal-aware placement to try to even out worst-case thermal profile [Sapatnekar, Wong et al.]
  - Computationally difficult problem (NP-complete)
  - Not able to predict all working conditions, and leakage changing dynamically, it is not useful in real systems

- Dynamic approach: HW-based dynamic thermal management
  - Clock gating based on time-out [Kie et al., Brooks et al.]
  - DVFS based on thresholds [Chaparro et al., Mukherjee et al.]
  - Heuristics for component shut down, limited history [Donnat et al]

Techniques to minimize power, they only achieve thermal management as a by-product

**Formalization of Thermal Management Problem in MPSoCs**

- Control theory problem
  - Observable: Geometrical properties and behavior
  - Performance counters
  - Controlable: Max. throughput under thermal constraints

- Tuning knobs: frequencies/voltages of the system (DVFS)

Techniques to minimize power, they only achieve thermal management as a by-product

**Pro-Active HW-Based Thermal Control: Phase 1 – Design-Time**

- Predictive model of thermal behavior given a set of frequency assignments

- Design-time phase: Find optimal sets of frequencies for the cores for different working conditions
- Run-time phase: Apply one of the predefined sets found in phase 1 for the required system performance
Pro-Active HW-Based Thermal Control: Phase 1 – Design-Time
- Predictive model of thermal behavior given a set of frequency assignments

Optimization problem: \( \min \sum_{i=1}^{n} p_i \)

Constraints:
- Power constraint on average, freq. is \( f_k \)
- Power equation based on frequency
- Thermal equation

Method: Table of cores frequencies assignments

Making Power and Thermal Constraints Convex
- Power constraint adaptation
  - Change non-affine (quadratic equality):
    \( p_{\text{max}} \left( f_{\text{avg}} \right)^2 / \left( f_{\text{max}} \right)^2 = p_{\text{avg}}; i = 1, \ldots, n, \forall k \)
  - To convex inequality:
    \( p_{\text{max}} \left( f_{\text{avg}} \right)^2 / \left( f_{\text{max}} \right)^2 \leq p_{\text{avg}}; i = 1, \ldots, n, \forall k \)

- Thermal constraint adaptation
  - Use worst case thermal conductivity in the range of allowed temperatures, and iterate (if needed) to optimum
**Pro-Active HW-Based Thermal Control:**

**Phase 2 - Run-Time, Putting It All Together**

- Use table of frequencies assignments and index by actual conditions at regular run-time intervals.

<table>
<thead>
<tr>
<th>Frequency of Cores</th>
<th>Temperature of Cores</th>
<th>Method inputs</th>
</tr>
</thead>
</table>

**Run-time optimal DVFS assignment HW module**

1. Index table output of phase 1 with current working conditions.
2. Compare to current assignment to cores and generate required signaling to modify DVFS values.

**Case Study: 8-Core Sun MPSoC**

- MPSoC Sun Niagara architecture
- 8 processing cores SPARC T1
- Max. frequency each core: 1 GHz
- 10 DVFS values, applied every 100ms
- Max. power per core: 4 W
- Execution characteristics of workloads (Sun Microsystems):
  - Mix of 10 different benchmarks, from web-accessing to multimedia
  - 60,000 iterations of basic benchmarks, tens of seconds of actual system execution

**Results: Thermal Constraints Respected**

- Proposed method achieves better throughput than standard DVFS while satisfying thermal constraints.
- Total run-time of benchmarks: 180 sec vs. 106 sec (45% less execution time).

**Outline**

- MPSoC thermal modeling and analysis
- HW-based thermal management for MPSoCs
- SW-based thermal management for MPSoCs
- Conclusions
Pro-Active Static (Offline) SW-Based Thermal Management

- It cannot really model run-time behavior!
- Need for online management!

Task Migration for Load vs. Thermal Balancing

- Plain load vs. Thermal Balancing
- No improvement in workload distribution possible: no migration
Task Migration for Load vs. Thermal Balancing

- Heat&Run: Load balancing with local knowledge of temperature in MPSoC components

Existent approaches do not consider global thermal dynamics for task migration!

- Migration strategy for thermal balancing
  - Global knowledge of temperature at MPOS level
  - Adjusted to particular thermal dynamics of each platform

- Formalization
  - Dynamic number of tasks, no control theory formalization possible
  - Knapsack problem, move N largest tasks between cores: estimated increase in temperature and minimizing performance penalty

Reduces hot-spots and reaches thermal balancing

Case Study: Freescale MPSoC Board

- Hardware
  - 3 RISC processor cores
  - 16KB caches, 32KB shared mem.
  - AMBA bus, 2GB ext. mem

- Software
  - uCLinux-based MPOS
  - Multimedia applications: audio and video

- Two packaging options
  - Mobile embedded SoCs (slow temperature variations)
  - High performance SoCs (fast temperature variations)
Results and Comparisons

- Good thermal balancing
  - Average: 40.5°C, variations of < 3°C
  - Small performance overhead (2 migrations)

- Comparisons with other policies
  - Load balancing inefficient (>7°C differences)
  - Heat & Run inefficient or causes many deadline misses (40% below performance requirements)
  - Performance requirements met for both types of packaging

~1.2ms @ 400MHz (1% overhead)

Comparisons with other policies

- Load balancing inefficient (>7°C differences)
- Heat & Run inefficient or causes many deadline misses (40% below performance requirements)
- Performance requirements met for both types of packaging

Adapt2D: Combination of HW and SW-Based Pro-Active Thermal Management

- Initial: Large gradients
- New: Thermal balancing

- HW-based management: Convex-based dynamic voltage and frequency scaling (DVFS) exploration
- SW-based management: Proactive task scheduling and migration
- Support of multi-processor operating system: Solaris Multi-Core

Good thermal control in commercial MPSoCs in 90nm, what about 3D integration?

Outline

- MPSoC thermal modeling and analysis
- HW-based thermal management for MPSoCs
- SW-based thermal management for MPSoCs
- Conclusions
Conclusions

- Progress in semiconductor technologies enables new MPSoCs
- Thermal/reliability issues must be addressed for safe human interaction
- Clear benefits of thermal-aware design methods for MPSoCs
- Novel, fast and low-cost thermal modeling approach at system-level
- Formalization of HW-based thermal management problem as convex, and solved in polynomial time
- New SW-based thermal balancing method with very limited overhead
- Validation on commercial 2D- MPSoCs (Sun, Freescale, Philips)
  - Fast exploration of thermal behavior of complex MPSoCs
  - Effective HW- and SW-based pro-active thermal management

Key References and Bibliography

- Thermal modeling and FPGA-based emulation
- Thermal management for 2D MPSoCs