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

ARTIST Summer School 2010, Autrans (France)
Outline

- MPSoC thermal modeling and analysis
- HW-based thermal management for MPSoCs
- SW-based thermal management for MPSoCs
- Conclusions
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!
MPSocC Thermal Modeling
State-of-the-Art

- MPSocC 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. Combination of cycle-accurate MPSocC behavior and IC heat flow modeling at run-time is unheard of

     Very expensive and late in design flow, no thermal modeling, only used for functional validation of MPSocC 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 MPSocC components!
Orthogonalizing MPSoC Thermal Modeling and Analysis

Framework: MPSoC behavioral model on reconfigurable HW interacting with efficient thermal estimation
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

\[
\begin{align*}
C_{\text{si,1}} & \left[ \begin{array}{cc}
G_{1,2} & -G_{1,2} \\
G_{2,1} & G_{2,1}
\end{array} \right] C_{\text{cu,n}} \end{align*}
\]

\[
T_{k} = - G (T_{k})_{1} + p_{k}; \quad k = 1..m
\]

Si thermal conductivity depends on temperature (IMEC & Freescale, 90nm)

Actual value
SW Thermal Estimation Tool for MPSoCs

\[ C \dot{t}_k = - G(t_k) t_k + p_k ; k = 1..m \]

- Creating linear approximation while retaining variable Si thermal conductivity:
  - Si thermal conductivity linearly approx.: \( G_{i,j}(t_k) = 1 + q t_k \)
  - Numerically integrating in discrete time domain the:
    \[ t_{k+1} = A(t_k) t_k + B \]

- Complexity scales linearly with the number of modeled cells (simulated on P4@ 3GHz)

- Thermal library validated against 3D finite element model (IMEC & Freescale)

- Time step chosen small enough for convergence

- Non-linear thermal estimation

- Proposed linear thermal estimation

- 60 sec of MPSoC heat flow analysis

- Heat flow estimation vs time (S.)
  - Linear fit
  - Actual value

- Number of Cells vs Heat flow estimation
  - Linear fit
  - Actual value

Temperature (in Celsius)

Number of Cells
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>$2.92 \times 10^2$</td>
<td>$1.02 \times 10^3$</td>
</tr>
<tr>
<td>D-Cache</td>
<td>$1.42 \times 10^2$</td>
<td>$7.10 \times 10^2$</td>
</tr>
<tr>
<td>I-Cache</td>
<td>$1.42 \times 10^2$</td>
<td>$7.10 \times 10^2$</td>
</tr>
<tr>
<td>Priv Mem</td>
<td>$0.61 \times 10^2$</td>
<td>$2.75 \times 10^2$</td>
</tr>
<tr>
<td>AMBA</td>
<td>$0.31 \times 10^2$</td>
<td>$0.68 \times 10^2$</td>
</tr>
</tbody>
</table>
Results: Thermal Validation 4-core Philips 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

Very fast validation of MPSoC
run-time thermal behavior and management

Emulation time 45 sec (128 cells)!

Package limit (~85°C)
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

- 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

  No formalization of the thermal optimization problem

- Dynamic approach: HW-based dynamic thermal management
  - Clock gating based on time-out [Xie et al., Brooks et al.]
  - DVFS based on thresholds [Chaparro et al., Mukherjee et al.]
  - Heuristics for component shut down, limited history [Donald 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: Geometric and behavioral properties
  - Controlable: Max. throughput under thermal constraints
  - Tunable: frequencies/voltages of the system (DVFS)

Optimal frequency assignment module, 2-phase approach:

1) **Design-time phase**: Find optimal sets of frequencies for the cores for different working conditions

2) **Run-time phase**: Apply one of the predefined sets found in phase 1 for the required system performance

Observed system: MPSoC

- **Run-time HW**: DVFS support
- **Performance counters**: (average frequency)
- **Thermal sensors**: Thermal state estimation
- **Control output**: Cores freqs

Requirements: Max. Throughput
Constraints: Max. temperature

Observer and control system
Pro-Active HW-Based Thermal Control: Phase 1 – Design-Time

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

<table>
<thead>
<tr>
<th>Phase 1 Design</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predictive model of thermal behavior given a set of frequency assignments</td>
<td></td>
</tr>
</tbody>
</table>

**Phase inputs**
- Predictive model of thermal behavior given a set of frequency assignments
- Allowed core power values and frequencies
- Chip floorplan
- Packaging, heat spreader information

**Optimization problem:**
\[
\text{minimize} \sum_{k=1}^{m} 1^T p_k
\]

**Non-linear offline problem**

- Performance constraint: on average, freq. is \( f_{\text{avg}} \)
- Meet temp. constraint at all time points
- Power equation: quadratic dependence on freq.
- Frequency in predefined range

**Thermal equation:** Si conductivity depends on temp

**Power equation based on frequency:**
\[
p_{\text{max}} f_i k / f_{\text{max}}^2 = p_{i,k}, \quad i = 1, \ldots, n, \quad k = 1, \ldots, m
\]

**Method outputs**
- Table of cores frequencies assignments
- Minimize sum of power consumption of cores
Making Power and Thermal Constraints Convex

- Power constraint adaptation
  - Solve convex problem and get table of optimal frequencies for different working conditions in polynomial time (number of processors)

- 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

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]:
  - Mixes of 10 different benchmarks, from web-accessing to multimedia
  - 60,000 iterations of basic benchmarks, tens of seconds of actual system execution

Sun’s Niagara MPSoC
Results: Thermal Constraints Respected

Proposed method achieves better throughput than standard DVFS while satisfying thermal constraints.

Total run-time of benchmarks:
- DVFS: 180 sec
- 2-phase Convex method: 106 sec (45% less exec. time)
Outline

- MPSoC thermal modeling and analysis
- HW-based thermal management for MPSoCs
- SW-based thermal management for MPSoCs
- Conclusions
Can we control the MPSoC thermal profile by controlling software execution?

- SW layers introduced to better exploit the HW of MPSoCs
  - Applications divided in *tasks*: blocks of operations to be executed
  - Multi-processor Operating System (MPOS) distributes the tasks
    - **Load balancing**: equal distribution of work between processors
Task Migration for Load vs. Thermal Balancing

- Plain load balancing

<table>
<thead>
<tr>
<th>LOAD FREQUENCY</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
</tr>
<tr>
<td>50%</td>
</tr>
<tr>
<td>0%</td>
</tr>
</tbody>
</table>

- TASK B
  - FSE LOAD: 40%

- TASK A
  - FSE LOAD: 40%

- TASK C
  - FSE LOAD: 40%

No improvement in workload distribution possible: no migration.
Heat & Run: Load balancing with local knowledge of temperature in MPSoC components.
Task Migration for Load vs. Thermal Balancing

- Heat&Run: Load balancing with local knowledge of temperature in MPSoC components
  - Helping with hot-spots, but no thermal balancing

Existing approaches do not consider global thermal dynamics for task migration.
Task Migration for Load vs Thermal Balancing

- **Contribution**: 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 migrat/s)

- Comparisons with other policies
  - Good performance and uniform temperature adjusting globally to thermal dynamics with MPOS
    - inefficient (>7°C diffs)
  - Heat&Run inefficient or causes many deadline misses (40% below performance requirements)
  - Contribution: performance requirements met for both types of packaging

~1.2ms @ 400MHz (1% overhead)
Adapt2D-MIGRA: Combination of HW and SW-Based Pro-Active Thermal Management

- **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 MPSO Cs
- SW-based thermal management for MPSO Cs
- Conclusions
Conclusions

- Progress in semiconductor technologies enables new MPSoCs
  - Thermal/reliability issues must be addressed for safe human interaction
  - Thermal monitoring and control are key

- 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
Thank you!

QUESTIONS?

Acknowledgements:

- UCSD / Sun Microsystems
- IMEC / Philips
- IBM Zürich
- Bologna / Freescale semiconductors

Swiss National Science Foundation

European Commission