Power and Energy Modelling of Multi-core Processors for System-Level Design Space Exploration

Santhosh Kumar Rethinagiri, Oscar Palomar, Osman Unsal, Adrian Cristal
Barcelona Supercomputing Center
Energy-Aware Computing Workshop 2014

This project and the research leading to these results has received funding from the European Community's Seventh Framework Programme [FP7/2007-2013] under grant agreement n 318693
ParaDIME Consortium
Why ParaDIME?

- **Parallel Distributed Infrastructure for Minimization of Energy**
- Rising cost
  - Hardware cost
  - Programming efficiency
  - Runtime optimization
  - Energy aware data center computing
The ParaDIME Stack

ParaDIME Infrastructure

Data Center

Computing Node/Stack
- Application/BM
- API
- Scala
- Actor Sched
- AKKA
- JVM
- OS
- Simulated HW
- Accelerators
- Cores
- Interconnect
- Future Devices

Computing Node/Stack
- Application/BM
- API
- Scala
- Actor Sched
- AKKA
- JVM
- OS
- VM
- Hypervisor
- Hyper
- Real HW
- VM
- VM
- VM
- VM
- Real HW
- Real HW
- Real HW
- Real HW

Multi Data Center Scheduler

Intra Data Center Scheduler

TuD
Cloud & Heat Technologies

UNINE
BSC
IMEC

EACO Workshop, Sept. 10th 2014, Bristol
Challenges of modelling power of heterogeneous systems

- Estimating power/energy is a **critical design** goal for electronic devices.
- Designers today must evaluate power estimation as early as possible in the electronics design.
- Design changes are easier in the design phase and have the greatest impact on application power estimation at System-Level.
- A platform to use **different processors and components**.
- Functional level is accurate but it’s a **course grain**. Restriction in terms of measuring power from the real board.
- For **fine grain**, we can achieve it from gate level simulation. Restriction applies as we don’t have the tools and RTL sources. Very slow simulation speed.
- Another challenge is power law holds for a simple processor but for **complex processor system** remains debatable?
### Power estimation methodology and tools

<table>
<thead>
<tr>
<th>System level power estimation</th>
<th>Multi-Level Power Analysis</th>
<th>Tools /Methodologies</th>
<th>Inputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithmic level power estimation</td>
<td>Functional Level Power Analysis</td>
<td>CAT / FLPA</td>
<td>AADL Model</td>
</tr>
<tr>
<td></td>
<td>Instruction level</td>
<td>SoftExplorer / FLPA</td>
<td>C code</td>
</tr>
<tr>
<td>Micro-Architectural level power estimation</td>
<td>Cycle level</td>
<td>Jouletrack / ILPA</td>
<td>ASM</td>
</tr>
<tr>
<td>Circuit level power estimation</td>
<td>Gate level</td>
<td>PowerTimer</td>
<td>ASM</td>
</tr>
<tr>
<td></td>
<td>Transistor level</td>
<td>Watchch</td>
<td>Binary</td>
</tr>
<tr>
<td>ILPA: I(prog) = Σ I(instr) + Σ ΔI</td>
<td>QuickPower</td>
<td>SimplePower</td>
<td></td>
</tr>
<tr>
<td>FLPA: parameterized activity</td>
<td>omed models</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Notes:**
- ILPA: Instantaneous Level Power Analysis
- FLPA: Functional Level Power Analysis

**Diagram:**
- Fast but coarse grained
- Slow but accurate
- COMPROMISE

**Tools/Methodologies:**
- CAT / FLPA
- SoftExplorer / FLPA
- Jouletrack / ILPA
- PowerTimer
- Watchch
- SimplePower
- QuickPower
- SPICE

**Simulation and modeling times:**
- Details
- Accuracy
Hybrid design space exploration methodology

First step – power model generation

Second step – system-level power estimation

Third step – design space exploration

MultiProcessor (SMP) architecture

Optimization results for re-mapping/simulation of application and hardware resources

Final DSE results/hardware implementation

Estimated power/energy/Timing

DESSERT Runtime management module

Optimized timing and work-load Inputs

Timing optimization/dynamic slack reclamation

Data or thread level parallelism/workload balancing

Functional Level Power Analysis

Power Models Library

Transaction virtual platform

Design space refinement

EACO Workshop, Sept. 10th 2014, Bristol
First step: FLPA (Functional Level Power Analysis)

- FLPA
  - 3 steps
  - Complex architectures
  - Simples models
  - Precise estimations
  - Processors Models
    - DSP & GPP
    - SoftExplorer Tool
  - FPGA Models
  - Flash Memory Models
  - System ...

1. Functional Analysis
   - Algorithmic parameters
   - Architectural parameters

2. Characterization
   - Scenario: $\alpha = 0 \ldots 1$
   - Architecture: $F = 20 \ldots 200$ MHz

3. Model determination
   - Algorithmic parameters
   - Architectural parameters

$P = f($parameters$)$
Functional block (ARM Cortex-A9)
# Generic Power Model Parameters

The parameters which influence the power in a system.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau$</td>
<td>External memory access rate</td>
</tr>
<tr>
<td>$\gamma_1$</td>
<td>L1 cache miss rate for a processor</td>
</tr>
<tr>
<td>$\gamma_2$</td>
<td>L2 cache miss rate for a processor</td>
</tr>
<tr>
<td>SCU</td>
<td>Snoop control unit counter for ARM Cortex-A9 multi-core</td>
</tr>
<tr>
<td>$\gamma_1$</td>
<td>L1 cache miss rate for a processor</td>
</tr>
<tr>
<td>$\rho$</td>
<td>Parallelism rate for DSP processor fetch stage access</td>
</tr>
<tr>
<td>$\beta$</td>
<td>DSP processor processing rate between instruction memory unit and processing unit</td>
</tr>
<tr>
<td>PSR</td>
<td>Pipeline stall rate between instruction memory unit and processing unit</td>
</tr>
<tr>
<td>IPC</td>
<td>Instruction Per Cycle</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Area utilization for a CLB</td>
</tr>
<tr>
<td>$F_{\text{processor}}$</td>
<td>Frequency of the processor</td>
</tr>
<tr>
<td>$F_{\text{bus}}$</td>
<td>Frequency of the bus</td>
</tr>
<tr>
<td>N</td>
<td>Number of cores</td>
</tr>
</tbody>
</table>
Power measurement environment

APC
Switched PDU
AP7921

Agilent
Power Analyzer
N6705A

Multiplexer
Agilent M9147A
4x1->4

POWER SUPPLY

SNMP

Ethernet Switch

Agilent Probe + Power supply
1141A + 1142A

High Precision
Digitizer
NI PXI-5105

PXI Chassis
NI PXIe-1062Q

DELL
Workstation
Precision T5500

probes
heads

Active differential probes

devices

internet

DMZ

LXI

ETHERNET

JTAG

RS232

PARA
DIME

Courtesey: Open-People project

EACO Workshop, Sept. 10th 2014, Bristol
Variation of Instruction Per Cycle (IPC) in Power for ARM Cortex-A8

![Graph showing variation of instruction per cycle (IPC) in power for ARM Cortex-A8. The x-axis represents IPC, and the y-axis represents power in milliwatts (mW). The graph shows a steady increase in power as IPC increases.]
## Power consumption models generated with FLPA

<table>
<thead>
<tr>
<th>Processors</th>
<th>Power models</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARM Cortex-A7</td>
<td>( P(\text{mW}) = 0.39 \ F_{\text{processor}} + 3.65 \ IPC + 0.66 \ (\gamma 1) + 2.12 \ (\gamma 2) + 8.27 )</td>
</tr>
<tr>
<td>(dual/quad-core)</td>
<td>( P(\text{mW}) = a \ F_{\text{processor}} + b \sum_{i=1}^{4} (\gamma 1 c_i) + c \sum_{i=1}^{4} (IPC c_i) + d \ (\gamma 2) + 5.24 )</td>
</tr>
<tr>
<td>ARM Cortex-A15</td>
<td>( P(\text{mW}) = 0.56 \ F_{\text{processor}} + 8.95 \ IPC + 1.2 \ (\gamma 1) + 3.5 \ (\gamma 2) + 15.93 )</td>
</tr>
<tr>
<td>(single core)</td>
<td></td>
</tr>
<tr>
<td>ARM Cortex-A15</td>
<td>( P(\text{mW}) = a \ F_{\text{processor}} + b \sum_{i=1}^{4} (\gamma 1 c_i) + c \sum_{i=1}^{4} (IPC c_i) + d \ (\gamma 2) + 17.45 )</td>
</tr>
<tr>
<td>(dual/quad-core)</td>
<td></td>
</tr>
<tr>
<td>Heterogeneous architecture</td>
<td>( E(\text{mJ}) = \sum_{i=1}^{n} E_{p_i} + E_{\text{mem}} + E_{\text{sync}} + E_{I/O} )</td>
</tr>
</tbody>
</table>
Second Step: System Level Power Analysis

System-level power estimator

Functional Level
- Processor & application mapping

Task and Data interface

Activity counter Interface

Task 1
- Cortex-A15 MPcore JIT
- Pipeline Stage module
- L1 I
- L1 D
- L2 Cache

Task 2
- Cortex-A7 MPcore JIT
- Pipeline Stage module
- L1 I
- L1 D
- L2 Cache

Power estimator Kernel

Power models Library

Transaction System-level virtual platform

Data
- Memory

External Memory interface

I/O Peripherals
Result Interface

Usage of armCortexA91

- DEMUX1: 23.52%
- IDCT1: 10.51%
- IQZZ1: 5.72%
- LBU1: 4.11%
- IRQ: 13.73%
- Context switch: 11.38%
- Idle: 0.00%
- VLD1: 31.03%
### Result Interface

<table>
<thead>
<tr>
<th>Estimation results</th>
<th>Cache miss rate (L1)</th>
<th>Cache miss rate (L2)</th>
<th>IPC</th>
<th>Power (mW)</th>
<th>Energy (mJ)</th>
<th>Power optimization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>6.1</td>
<td>0.15</td>
<td>1.28</td>
<td>474.4</td>
<td>860.7</td>
<td>DVFS</td>
</tr>
<tr>
<td>Core details</td>
<td>ARM Cortex-A8</td>
<td>1 core</td>
<td>No multicore</td>
<td>JPEG</td>
<td>Power optimized 20 %</td>
<td>Work Load - NO</td>
</tr>
</tbody>
</table>

---

**Dessert**

- Timing details
- Optimize
- Estimate
- Export .CSV File

---

EACO Workshop, Sept. 10th 2014, Bristol
Results and Comparison (Power estimation)

- Estimates for A7 (single core)
- Measurements for A7 (single core)
- Estimates for A7 (dual-core)
- Measurements for A7 (dual-core)
- Estimates for A7 (quad-core)
- Measurements for A7 (quad-core)

- Estimates for A15 (single core)
- Measurements for A15 (single core)
- Estimates for A15 (dual-core)
- Measurements for A15 (dual-core)
- Estimates for A15 (quad-core)
- Measurements for A15 (quad-core)

Power (mW)

Dithering  Ghost script  Image rotation  Text parsing  Coremark Benchmarks  H.264  JPEG  Perlbench  gcc  mcf
Results and comparison (Energy)

- Estimates for A7 (single core)
- Measurements for A7 (single core)
- Estimates for A7 (dual-core)
- Measurements for A7 (dual-core)
- Estimates for A7 (quad-core)
- Measurements for A7 (quad-core)
- Estimates for A15 (single core)
- Measurements for A15 (single core)
- Estimates for A15 (dual-core)
- Measurements for A15 (dual-core)
- Estimates for A15 (quad-core)
- Measurements for A15 (quad-core)

Energy (J)

- Dhrystone
- FDCT
- IMDCT
- Matrix multi
- Kalman filter
- Parosorting
- MTT radar
- SDR
- JPEG
- MPEG 2
- H.264 Full HD
- FIR
- Downscalar

EACO Workshop, Sept. 10th 2014, Bristol

PARADIME
Third step: Auto optimization

- DVFS
  - Runtime – Inter task DVFS
  - Programmer annotation based DVFS

- Work-load balancing based on task
  - Runtime
  - Programmer based request
Task scheduling
Optimization based on work load balancing
## Optimization (Inter task DVFS)

<table>
<thead>
<tr>
<th>Mixed processor core</th>
<th>Power estimated (mW)</th>
<th>Power measured (mW)</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 processor core</td>
<td>3854</td>
<td>3914</td>
<td>1.5</td>
</tr>
<tr>
<td>4 processor core</td>
<td>4587</td>
<td>4626</td>
<td>0.84</td>
</tr>
<tr>
<td>8 processor core</td>
<td>5327</td>
<td>5398</td>
<td>1.3</td>
</tr>
</tbody>
</table>
Conclusion

- In our tool, we have proved that our estimates are accurate.
- Adaptable for any kind of complex processor system.
- Added advantage, rapid prototyping of the components and porting of the applications made easy.
- Estimating power and designing applications made easy and time efficient.
Hardware Architecture

- Energy-Efficient Message Passing
  - Message passing microarchitecture
  - Message passing accelerator
    - Task passing

- Operation Below Safe $V_{dd}$
  - Automatic HW lowering of $V_{dd}$
  - SW-guided (low-power annotation)
  - Errors?

- Heterogeneous Computing
  - Architectural level
  - Device level
Heterogeneous system-level environment

- ARM Cortex-A9 Quad-core ISS
- ARM Cortex-A8 Quad-core ISS
- DSP C64x ISS
- FPGA Hardware Accelerator
- GPU accelerator

Data

Memory

Virtual Platform

PETS Tool activity counter Interface

Task 1

Task 2

Task 3
Heterogeneous computing results and comparison
Programming model

- Message passing programming model
  - Actor model (Akka+Scala)
- Annotations to provide information to the hardware
  - Operation below Safe Vdd
  - Approximate Computing

```scala
@Storage(Array("precise=false", "VF_relax=true"))
var x = 5
@Calculation(Array("VF_relax=true", "VF_det=DMR", "VF_corr=TM"))
def calc(first:Array[double])
```

- Rewrite/Expand annotated code with Scala Macro Annotations
Power and Energy Modeling of Multi-core Processors for System-Level Design Space Exploration

Oscar Palomar (BSC)
Energy-Aware Computing Workshop 2014

This project and the research leading to these results has received funding from the European Community's Seventh Framework Programme [FP7/2007-2013] under grant agreement n 318693
Computation of power and measurement of voltage for OMAP

\[ I_{vdd1} = V_{dropJ6} = 0.05 \]

\[ I_{vdd2} = V_{dropJ5} = 0.1 \]
Power measurement environment