Combining Data Remapping and Voltage/Frequency Scaling of Second Level Memory for Energy Reduction in Embedded Systems

Sudarshan K. Srinivasan, Jun Cheol Park and Vincent J. Mooney III
Georgia Institute of Technology
{darshan, jcpark, mooney}@ece.gatech.edu
Outline

- Introduction
- Motivation
- Related Work in Power Modeling
- Experimental Setup
- Data Remapping
- Voltage/Frequency Scaling of Off-chip Memory and Bus
- Experimental Results
- Conclusion
Introduction

- Power/energy is a major issue in embedded systems
- Mobile devices require longer usage time
Introduction (Cont.)

- Memory consumes up to 45% of the total system power*
- Memory is a main target for power/energy reduction

Motivation

Data Remapping Software Technique

Voltage/Frequency Scaling Hardware Technique

Embedded System

- Processor+L1 cache
- Off-chip Bus+L2 cache

Reduction in E.T. & Energy

Energy Reduction

Jun Cheol Park
Georgia Institute of Technology
Related Work in Power Modeling

- Simplescalar/ARM PowerAnalyzer*
  - Cycle level power/performance simulator

- SimplePower**
  - Architectural power estimation tool
  - Does not capture the energy of control unit of processor, clock generation

* http://www.eecs.umich.edu/~jringenb/power/
** http://www.cse.psu.edu/~mdl/software.htm
Experimental Setup

- Processor Core Energy
- Off-chip Bus Energy
- L1 and L2 caches Energy

System Energy
Experimental Setup (Cont.)

- Processor core power

Diagram:

- Benchmark Program (C/C++)
- Binary Translation
- ARM9 Based System Architecture
- RTL Description (Verilog)
- Functional Simulation (VCS)
- Toggle Rate (Activity) Generation
- Synthesize Verilog Model
- Processor Core Power
Experimental Setup (Cont.)

- Processor core power
  - MARS (Michigan ARM Simulator)
    - A cycle accurate verilog model of a RISC processor
    - Capable of running ARM instructions

Diagram:

- Benchmark Program (C/C++)
- Binary Translation
- ARM9 Based System Architecture
- RTL Description (Verilog)
- Functional Simulation (VCS)
- Toggle Rate (Activity) Generation
- Synthesize Verilog Model
- Processor Core Power
Experimental Setup (Cont.)

- **Processor core power**
  - Collect toggle rate of internal logic signals using Synopsys VCS simulation
  - Synthesize verilog model using TSMC .25μ library
Experimental Setup (Cont.)

- Processor core power
  - Estimate power using Synopsys Power Compiler

```
Benchmark Program (C/C++)

Binary Translation

ARM9 Based System Architecture

RTL Description (Verilog)

Functional Simulation (VCS)

Toggle Rate (Activity) Generation

Synthesize Verilog Model

Processor Core Power
```
Experimental Setup (Cont.)

- Off-chip bus power
  - Bus capacitance obtained from actual board
  - PCB board with SA110 processor (Skiff board)

Diagram:

- Benchmark Program (C/C++)
- Binary Translation
- ARM9 Based System Architecture
- RTL Description (Verilog)
- Functional Simulation (VCS)
- Skiff board
- Toggle Rate (Activity) Generation
- Off-chip bus parameters
- Off-chip Bus Power
Experimental Setup (Cont.)

- L1 and L2 caches energy
  - TRIMARAN*
    - Integrated compilation and performance monitoring infrastructure
    - ARM-like processor simulator
    - TRICEPS
      - Generate ARM code
    - SMACS (Smart Memory and Cache Hierarchy Simulator)
      - cache activity statistics
  - Kamble and Ghose model**

*TRIMARAN http://www.trimaran.org
Experimental Setup (Cont.)

Processor Core Power

Off-chip Bus Power

L1 and L2 caches Energy

System Energy

TRIMARAN

Execution Time

+=

+=

+=

+=
Data Remapping*

- A compile time technique for performance enhancement and energy reduction
- Remapping data into new set such that data items that are more likely to be used together are grouped together into the same cache block
- Enhancing spatial locality

Data Remapping (Cont.)

Amount of data fetched before and after remapping
(Traveling salesman problem in Olden Suite)
An item in memory is accessed by initiating a load of the contents of a memory location or address.

Since a memory access is expensive, a set of adjacent memory locations are loaded at the same time and stored in a cache.

- The set of adjacent memory locations is known as a memory block.
  - Blocks do not overlap and have the same size.

Each address can be mapped to a block in memory.
Data Remapping (Cont.)

- Data reorganization is the relocation of data objects in memory

![Diagram showing data reorganization]
Data Remapping (Cont.)

- Analyze application memory access pattern then remap data

![Diagram showing memory address and blocks]

Address

Memory Block
Voltage/frequency scaling of off-chip memory and bus*

- Scaling down supply voltage of off-chip bus and memory (L2 cache)
  - $P$ is proportional to $V^2$
- Significant energy saving in L2 cache
- Doubling the memory access latency
- L2 cache miss rate affects system performance significantly

Voltage/frequency scaling of off-chip memory and bus (Cont.)

100 Mhz, 2.75 V  100 Mhz, 3.3 V  100 Mhz, 2.75 V  50 Mhz, 2.0 V
Experimental Results

- Two Olden benchmarks (Health and Perimeter) are used
- The supply voltage for L2 cache and buses are scaled down to 2V, 50Mhz
- The benchmarks are remapped and simulated with 50Mhz L2 cache
- Half size L1 and L2 cache system is simulated
  - Data remapping can achieve same execution time with half cache resources
Experimental Results (Cont.)

Energy delay with frequency/voltage scaling of memory (FVM) and data remapping (DR) for health benchmark (L1 32KB 16B/line, L2 1MB 32B/line)

<table>
<thead>
<tr>
<th>Execution Cycles</th>
<th>Before DR, FVM</th>
<th>After DR</th>
<th>After FVM</th>
<th>After DR+FVM</th>
<th>After DR+FVM 1/2 size L1</th>
<th>After DR+FVM 1/2 size L2</th>
<th>After DR+FVM 1/2 size L1,L2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delay (Execution Time)(s)</td>
<td>8.036</td>
<td>4.796</td>
<td>8.926</td>
<td>5.78</td>
<td>6.033</td>
<td>7.112</td>
<td>7.363</td>
</tr>
<tr>
<td>Energy*Delay</td>
<td>137.231</td>
<td>49.687</td>
<td>127.778</td>
<td>53.608</td>
<td>57.118</td>
<td>79.35</td>
<td>74.618</td>
</tr>
<tr>
<td>% Energy Reduction</td>
<td>0</td>
<td>39.33</td>
<td>16.16</td>
<td>45.69</td>
<td>44.55</td>
<td>34.66</td>
<td>40.65</td>
</tr>
<tr>
<td>% Energy*Delay Reduction</td>
<td>0</td>
<td>63.79</td>
<td>6.89</td>
<td>60.94</td>
<td>58.38</td>
<td>42.18</td>
<td>45.63</td>
</tr>
</tbody>
</table>
Experimental Results (Cont.)

- Maximum of 46% of energy reduction
- Energy consumption of the cache reduced by half after halving L1 and L2 cache without performance loss

Energy delay with frequency/voltage scaling of memory (FVM) and data remapping (DR) for health benchmark (L1 32KB 16B/line, L2 1MB 32B/line)

- % Energy Reduction
- % Energy*Delay Reduction
Conclusion

- Combine of two techniques (HW & SW) to maximize energy reduction
- Achieve 46% of energy reduction without performance loss
- Achieve 1/2 energy consumption with half size cache, same performance
Thank you.