COMPSAC 2017 Plenary Panel
Future of Computing: Exciting Research in Computers, Software and Applications

Green Multicore Computing

Hironori Kasahara
President Elect 2017, President 2018
IEEE Computer Society
Professor, Dept. of Computer Science & Engineering
Director, Advanced Multicore Processor Research Institute
Waseda University, Tokyo, Japan
URL: http://www.kasahara.cs.waseda.ac.jp/

July 7, 2017 (Friday)
Performance and Low Power are Key Issues

Power consumption is one of the biggest problems for performance scaling from smartphones to cloud servers and supercomputers (“K” more than 10MW).

Power \propto Frequency \cdot Voltage^2 
(Voltage \propto Frequency)

Power \propto Frequency^3

If Frequency is reduced to 1/4 
(Ex. 4GHz \rightarrow 1GHz), 
Power is reduced to 1/64 and 
Performance falls down to 1/4 .

<Multicores>

If 8cores are integrated on a chip, 
Power is still 1/8 and 
Performance becomes 2 times.
With 128 cores, OSCAR compiler gave us 100 times speedup against 1 core execution and 211 times speedup against 1 core using Sun (Oracle) Studio compiler.
OSCAR Parallelizing Compiler

To improve **effective performance**, **cost-performance** and **software productivity** and reduce power

**Multigrain Parallelization**

coarse-grain parallelism among loops and subroutines, near fine grain parallelism among statements in addition to loop parallelism

**Data Localization**

Automatic data management for distributed shared memory, cache and local memory

**Data Transfer Overlapping**

Data transfer overlapping using Data Transfer Controllers (DMAs)

**Power Reduction**

Reduction of consumed power by compiler control DVFS and Power gating with hardware supports.
Power can be reduced by software control: MPEG2 Decoding to 1/4 on 8 Core Multicore by OSCAR Parallelizing Compiler

Without Power Control
(Voltage : 1.4V)

With Power Control
(Frequency,
Resume Standby:
Power shutdown & Voltage lowering 1.4V-1.0V)

Avg. Power
5.73 [W] 73.5% Power Reduction Avg. Power
1.52 [W]
Power of Multicores with DVFS can be Reduced by Software: Intel Haswell

For HD 720p (1280x720) moving pictures 15fps (Deadline 66.6 [ms/frame])

Power was reduced to 1/4 by compiler on 3 cores

Intel CPU Core i7 4770K
- without power control
- with power control

Real-time Optical Flow

Average power consumption [W]

<table>
<thead>
<tr>
<th>number of PE</th>
<th>1PE</th>
<th>2PE</th>
<th>3PE</th>
</tr>
</thead>
<tbody>
<tr>
<td>29.29</td>
<td>24.17</td>
<td>36.59</td>
<td>41.58</td>
</tr>
<tr>
<td>12.21</td>
<td>9.60</td>
<td>1/3</td>
<td>1/4</td>
</tr>
</tbody>
</table>

1 core Power (29.3W) was reduced to 1/3 (9.6W) with 3 cores by OSCAR compiler.
An Image of Static Schedule for Heterogeneous Multi-core with Data Transfer Overlapping and Power Control
33 Times Speedup Using OSCAR Compiler and OSCAR API on RP-X
(Optical Flow with a hand-tuned library)

Y. Yuyama, et al., “A 45nm 37.3 GOPS/W Heterogeneous Multi-Core SoC”, ISSCC2010
Power Reduction in a real-time execution controlled by OSCAR Compiler and OSCAR API on RP-X (Optical Flow with a hand-tuned library)

Without Power Reduction

With Power Reduction by OSCAR Compiler

70% of power reduction

Average: 1.76[W]  →  Average: 0.54[W]

1 cycle : 33[ms]  →  30[fps]
Automatic Parallelization of JPEG-XR for Drinkable Inner Camera (Endo Capsule)

10 times more speedup needed after parallelization for 128 cores of Power 7. Less than 35mW power consumption is required.

TILEPro64

Speed-ups on TILEPro64 Manycore

55 times speedup with 64 cores

1core 10.0[s]

1.00 1.96 3.95 7.86 15.82 32 64

64cores 0.18[s]

Waseda U. & Olympus
Architecture Design to Support for Parallelization and Power Reduction by Compiler

**Vector Multicore for Embedded to Servers**

- Solar Powered with compiler power reduction.
- Fully automatic parallelization and vectorization including local memory management and data transfer.
Summary

- Software can further reduce the power consumption of low power processor hardware.
  - To develop the parallel software with low development cost and period, automatic paralleling compiler is required.

- Co-design of compilers and architectures will be more important.
  - For example, designing compiler looking at applications first and designing multicore system architectures would be promising.
Sequential Application Program in Fortran or C
(Consumer Electronics, Automobiles, Medical, Scientific computation, etc.)

Low Power Heterogeneous Multicore Code Generation
API Analyzer
Available from Waseda

OSCAR API for Homogeneous and/or Heterogeneous Multicores and manycores
Directives for thread generation, memory, data transfer using DMA, power managements

Parallelized API F or C program
Proc0
Code with directives
Thread 0
Proc1
Code with directives
Thread 1

Low Power Homogeneous Multicore Code Generation
API Analyzer
Existing sequential compiler

Low Power Heterogeneous Multicore Code Generation
API Analyzer (Available from Waseda)
Existing sequential compiler

Server Code Generation
OpenMP Compiler

Parallelized Code with directives
Accelerator 1
Code with directives
Thread 1
Accelerator 2
Code

Waseda OSCAR Parallelizing Compiler
- Coarse grain task parallelization
- Data Localization
- DMAC data transfer
- Power reduction using DVFS, Clock/Power gating

Accelerator Compiler/ User
Add “hint” directives before a loop or a function to specify it is executable by the accelerator with how many clocks

Homegeneous Multicores from Vendor A
(SMP servers)

Heterogeneous Multicores from Vendor B
Hitachi, Renesas, NEC, Fujitsu, Toshiba, Denso, Olympus, Mitsubishi, Esol, Cats, Gaio, 3 univ.

EXECUTABLE ON VARIOUS MULTICORES

Generation of parallel machine codes using sequential compilers

Manual parallelization / power reduction

Parallelization
Data Localization
DMAC data transfer
Power reduction using DVFS, Clock/Power gating

OSCAR: Optimally Scheduled Advanced Multiprocessor API: Application Program Interface

OpenMP

Shred memory servers