Automatic Parallelization of MATLAB/Simulink on Multicore Processors
-- Parallel processing of automobile engine control
C code generated by embedded coder --

Hironori Kasahara
Professor, Dept. of Computer Science & Engineering
Director, Advanced Multicore Processor Research Institute
Waseda University, Tokyo, Japan

IEEE Computer Society Board of Governors
IEEE Computer Society Multicore STC Chair
URL: http://www.kasahara.cs.waseda.ac.jp/

Waseda Univ. Green Computing Systems R&D Center
Green Computing Systems R&D Center
Waseda University
Supported by METI (Mar. 2011 Completion)

<R & D Target>
Hardware, Software, Application for Super Low-Power Manycore Processors
- More than 64 cores
- Natural air cooling (No fan)
  Cool, Compact, Clear, Quiet
- Operational by Solar Panel

<Industry, Government, Academia>
Hitachi, Fujitsu, NEC, Renesas, Olympus, Toyota, Denso, Mitsubishi, Toshiba, etc

<Ripple Effect>
- Low CO₂ (Carbon Dioxide) Emissions
- Creation Value Added Products
  - Consumer Electronics, Automobiles, Servers

Hitachi SR16000: Power7 128 core SMP
Fujitsu M9000 SPARC VII 256 core SMP

Beside Subway Waseda Station, Near Waseda Univ. Main Campus
Industry-government-academia collaboration in R&D and target practical applications

For smart life
- Robots
- On-board vehicle technology (navigation systems, integrated controllers, infrastructure coordination)
- Consumer electronic Internet TV/DVD
- Camcorders
- Capsule inner cameras
- Cameras
- Solar Powered Smart phones
- Operation/recharging by solar cells
- Medical servers
- Heavy particle radiation planning, cerebral infarction
- Non-fan, cool, quiet servers designed for server

Protect environment
- Green supercomputers
- Super real-time disaster simulation (tectonic shifts, tsunami, tornado, flood, fire spreading)
- Green cloud servers
- Stock trading

Industry
- Supercomputers and servers
- Waseda University : R&D
- Many-core system technologies with ultra-low power consumption
- OSCAR many-core chip
- Many-core Chip
- Compiler, API

Protect lives
- Protect lives
- Protect environment
- Protect

Intelligent home appliances
- National Institute of Radiological Sciences
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.
Multicore Program Development Using OSCAR API V2.0

Sequential Application Program in Fortran or C
(Consumer Electronics, Automobiles, Medical, Scientific computation, etc.)

Manual parallelization / power reduction

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

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

Hitachi, Renesas, NEC, Fujitsu, Toshiba, Denso, Olympus, Mitsubishi, Esol, Cats, Gaio, 3 univ.

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
- Accelerator 1
  Code
- Accelerator 2
  Code

Low Power Homogeneous Multicore Code Generation
- API Analyzer
- Existing sequential compiler

Low Power Heterogeneous Multicore Code Generation
- API Analyzer
- Existing sequential compiler

Server Code Generation
OpenMP Compiler

Generation of parallel machine codes using sequential compilers

Executable on various multicores

HOMEOGENEOUS MULTICORES FROM VENDOR A (SMP SERVERS)

HETEROGENEOUS MULTICORES FROM VENDOR B

SHRED MEMORY SERVERS
Cancer Treatment
Carbon Ion Radiotherapy
(Previous best was 2.5 times speedup on 16 processors with hand optimization)

8.9 times speedup by 12 processors
Intel Xeon X5670 2.93GHz 12 core SMP (Hitachi HA8000)

55 times speedup by 64 processors
IBM Power 7 64 core SMP (Hitachi SR16000)
92 Times Speedup against the Sequential Processing for GMS Earthquake Wave Propagation Simulation on Hitachi SR16000 (Power7 Based 128 Core Linux SMP)

GMS: Ground Motion Simulator from National Research Institute for Earth Science and Disaster Prevention (NIED)
Power Reduction of MPEG2 Decoding to 1/4 on 8 Core Homogeneous Multicore RP-2 by OSCAR Parallelizing Compiler

**MPEG2 Decoding with 8 CPU cores**

Without Power Control
(Voltage: 1.4V)

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

**Avg. Power**

<table>
<thead>
<tr>
<th>Without Power Control</th>
<th>With Power Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg. Power</td>
<td>Avg. Power</td>
</tr>
<tr>
<td>5.73 [W]</td>
<td>1.52 [W]</td>
</tr>
</tbody>
</table>

73.5% Power Reduction
Automatic Power Reduction for MPEG2 Decode on Android Multicore

ODROID X2 ARM Cortex-A9 4 cores

http://www.youtube.com/channel/UCS43lNYElkC8i_KigFZYQBQ

- On 3 cores, Automatic Power Reduction control successfully reduced power to 1/7 against without Power Reduction control.
- 3 cores with the compiler power reduction control reduced power to 1/3 against ordinary 1 core execution.
OSCAR Compile Flow for Simulink Applications

Generate C code using Embedded Coder

Simulink model

C code

(1) Generate MTG → Parallelism

(2) Generate gantt chart → Scheduling in a multicore

(3) Generate parallelized C code using the OSCAR API → Multiplatform execution (Intel, ARM and SH etc)
Engine Control by multicore with Denso

Though so far parallel processing of the engine control on multicore has been very difficult, Denso and Waseda succeeded 1.95 times speedup on 2core V850 multicore processor.
Parallel Processing of Automotive Engine Control Applications on Two Cores using the OSCAR compiler

![Bar chart comparing Renesas SH-4a and Renesas V850 for Basic Engine Control and Fuel Injection Control.]
Speedups of MATLAB/Simulink Image Processing on Various 4core Multicores

(Intel Xeon, ARM Cortex A15 and Renesas SH4A)

Road Tracking, Image Compression : [http://www.mathworks.co.jp/jp/help/vision/examples](http://www.mathworks.co.jp/jp/help/vision/examples)
Parallel Processing on Simulink Model

• The parallelized C code can be embedded to Simulink using C mex API for HILS and SILS implementation.

Call sequential C code from the S-Function block

Call parallelized C code from the S-Function block
Conclusions

- This talk introduced an automatic parallelization method using OSCAR Compiler for automobile engine control and image processing designed by using MATLAB/Simulink.

- The OSCAR parallelizing compiler allows us to parallelize C codes generated by the Embedded Coder for various multicore processors including ARM, Intel, AMD, Qualcomm, Freescale, IBM, Fujitsu, Renesas and so on.

- Performance evaluation showed
  - 1.91 and 1.79 times of speedups on 2 cores using Renesas SH4A and V850 respectively against on 1 core for Automobile Fuel Injection program,
  - 3.56, 3.12 and 3.45 times of speedups on 4 cores Intel Xeon, ARM Cortex A15 and Renesas SH4A respectively against on 1 core for Vessel Detection program.