Compiler and API for Low Power High Performance Multicores

Hironori Kasahara

Professor, Department of Computer Science

Director, Advanced Chip-Multiprocessor Research Institute

Waseda University

Tokyo, Japan

http://www.kasahara.cs.waseda.ac.jp

June 24, 2008, MPSoC 2008

METI/NEDO National Project

Multi-core for Real-time Consumer Electronics

<Goal> R&D of compiler cooperative multi-core processor technology for consumer electronics like Mobile phones, Games, DVD, Digital TV, Car navigation systems.

<Period> From July 2005 to March 2008

<Features> ・Good cost performance

・Short hardware and software development periods

・Low power consumption

・Scalable performance improvement with the advancement of semiconductor

・Use of the same parallelizing compiler for multi-cores from different vendors using newly developed API

API：Application Programming Interface

(2005.7〜2008.3)＊＊

新マルチコア
プロセッサ
・高性能
・低消費電力
・短HW/SW開発期間
・各チップ間でアプリケーション共用可
・高信頼性
・半導体集積度と共に性能向上
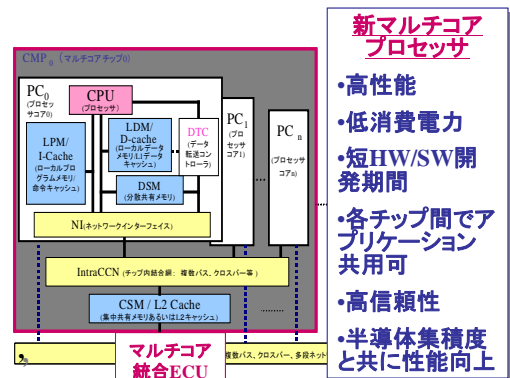
開発マルチコアチップは情報家電へ

**Hitachi, Renesas, Fujitsu, Toshiba, Panasonic, NEC

# OSCAR Parallelizing Compiler

- **Improve effective performance, cost-performance and productivity and reduce consumed power**
  - **Multigrain Parallelization**
    - Exploitation of parallelism from the whole program by use of coarse-grain parallelism among loops and subroutines, near fine grain parallelism among statements in addition to loop parallelism
  - **Data Localization**
    - Automatic data distribution for distributed shared memory, cache and local memory on multiprocessor systems.
  - **Data Transfer Overlapping**
    - Data transfer overhead hiding by overlapping task execution and data transfer using DMA or data pre-fetching
  - **Power Reduction**
    - Reduction of consumed power by compiler control of frequency, voltage and power shut down with hardware supports.

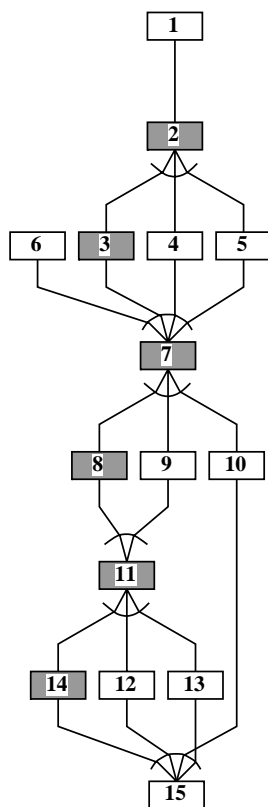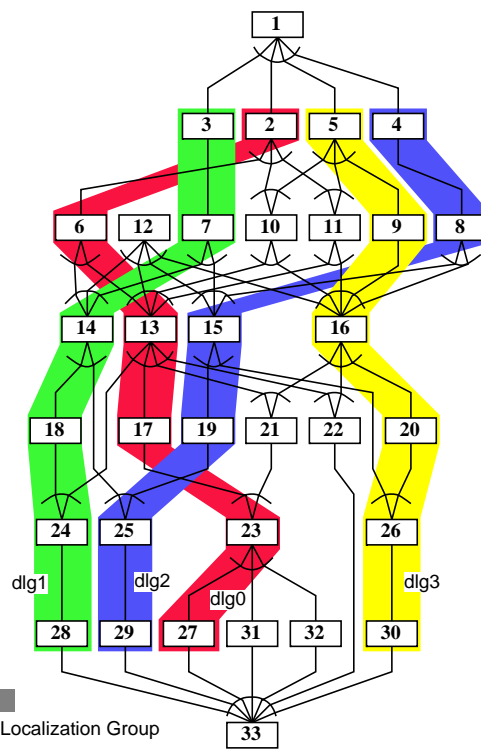# Earliest Executable Condition Analysis for coarse grain tasks (Macro-tasks)

A Macro Flow Graph

A Macro Task Graph
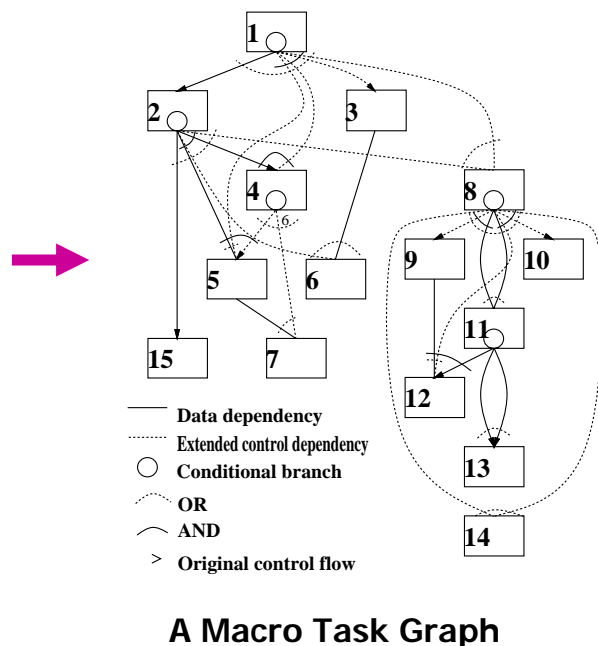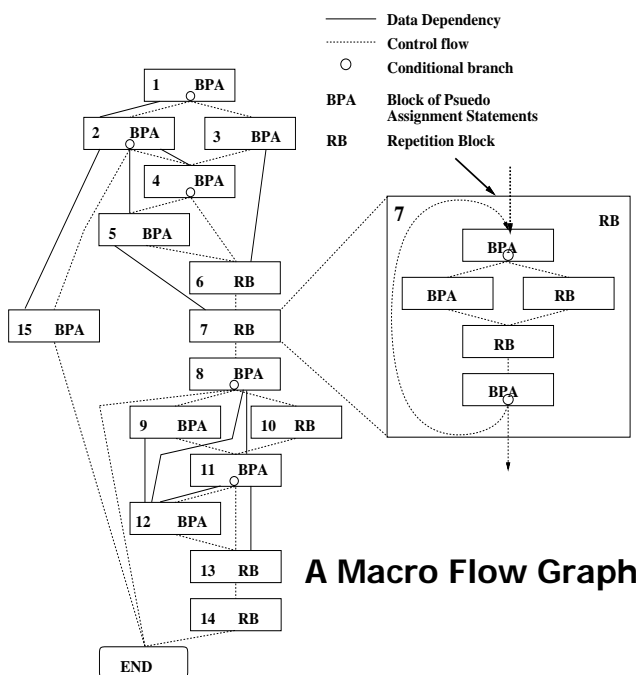
MTG of Su2cor-LOOPS-DO400

- Coarse grain parallelism PARA_ALD = 4.3

	DOALL		Sequential LOOP		SB		BB
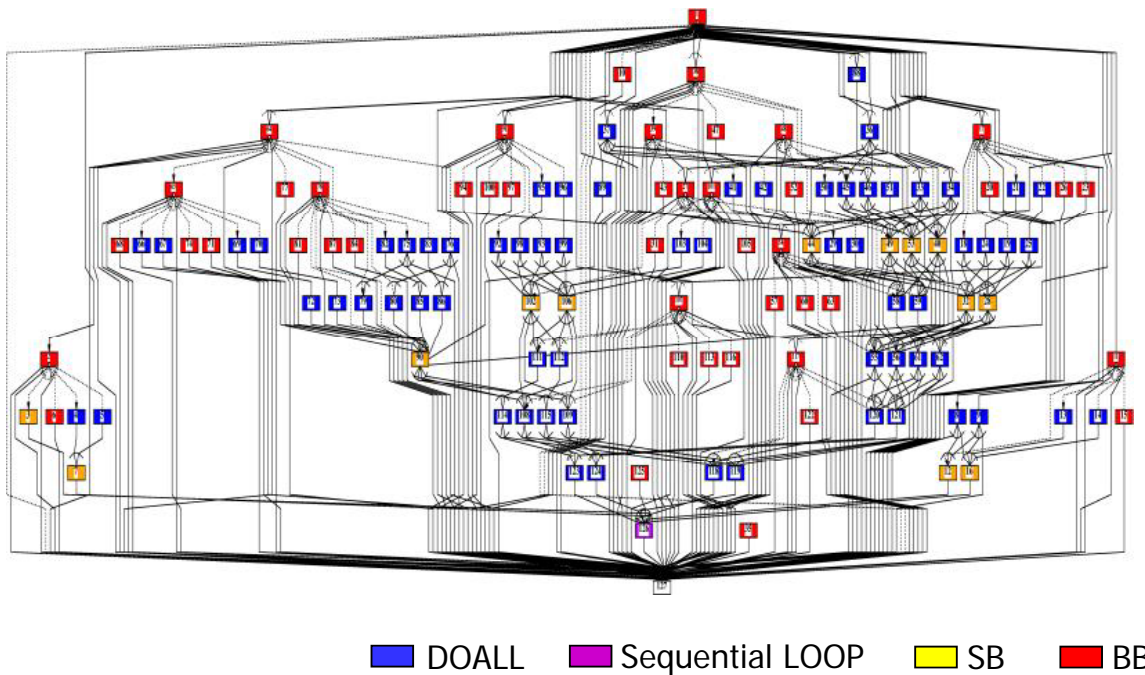
Data Localization
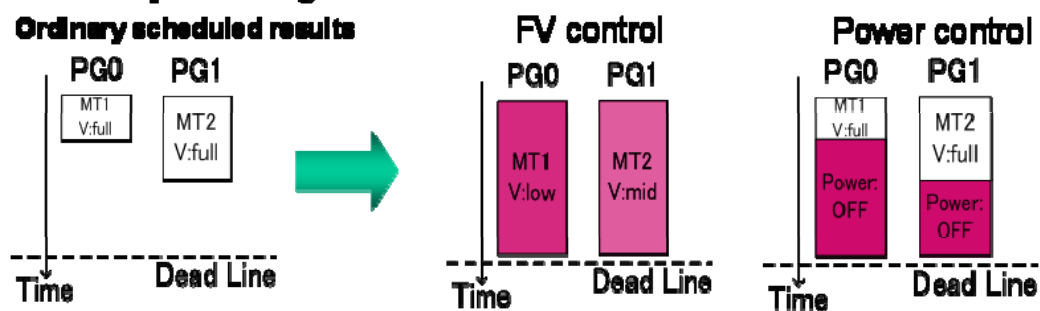

MTG MTG after Division A schedule for two processors

Power Reduction by Power Supply, Clock Frequency and Voltage Control by OSCAR Compiler

- Shortest execution time mode
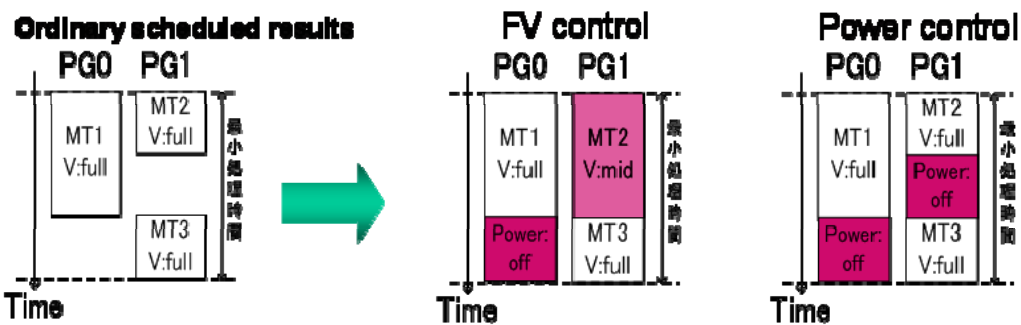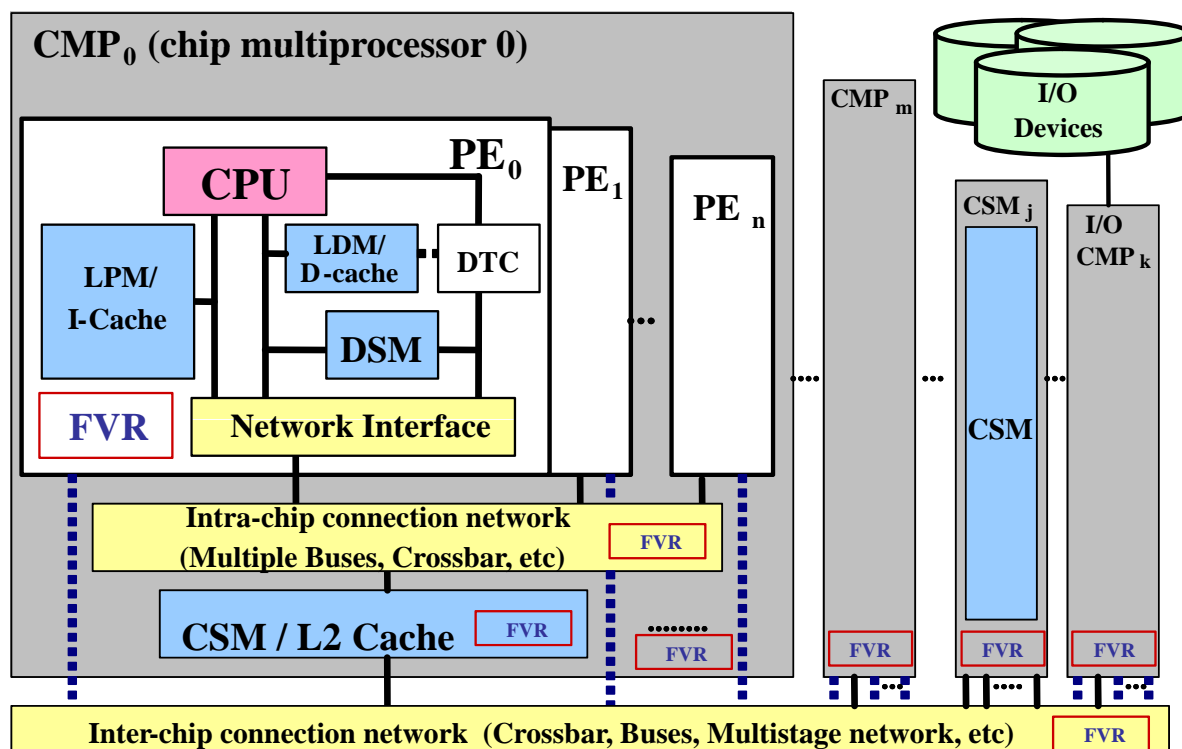

- Realtime processing mode with dead line constraints

OSCAR Multi-Core Architecture

|	
CSM: central shared mem.	LDM : local data mem.
DSM: distributed shared mem.	LPM : local program mem.
DTC: Data Transfer Controller	FVR: frequency / voltage control register

API and Parallelizing Compiler in METI/NEDO Advanced Multicore for Realtime Consumer Electronics Project

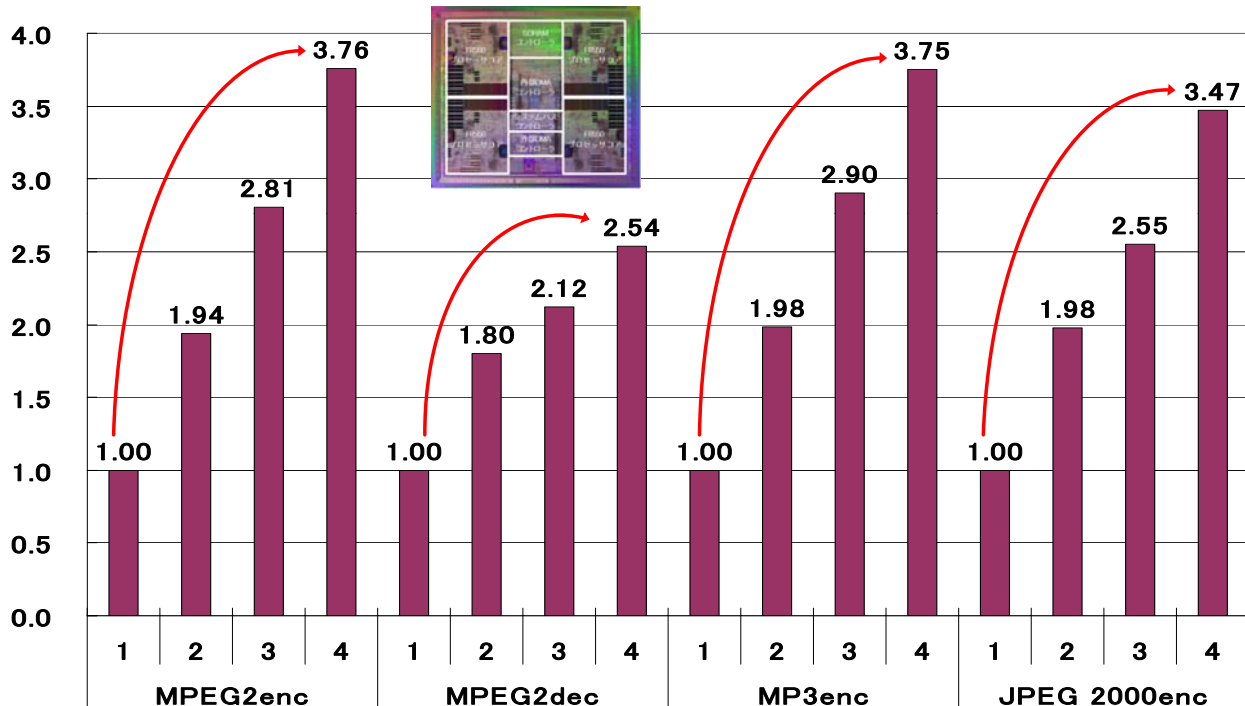Details of API: See http://www.kasahara.cs.waseda.ac.jp/
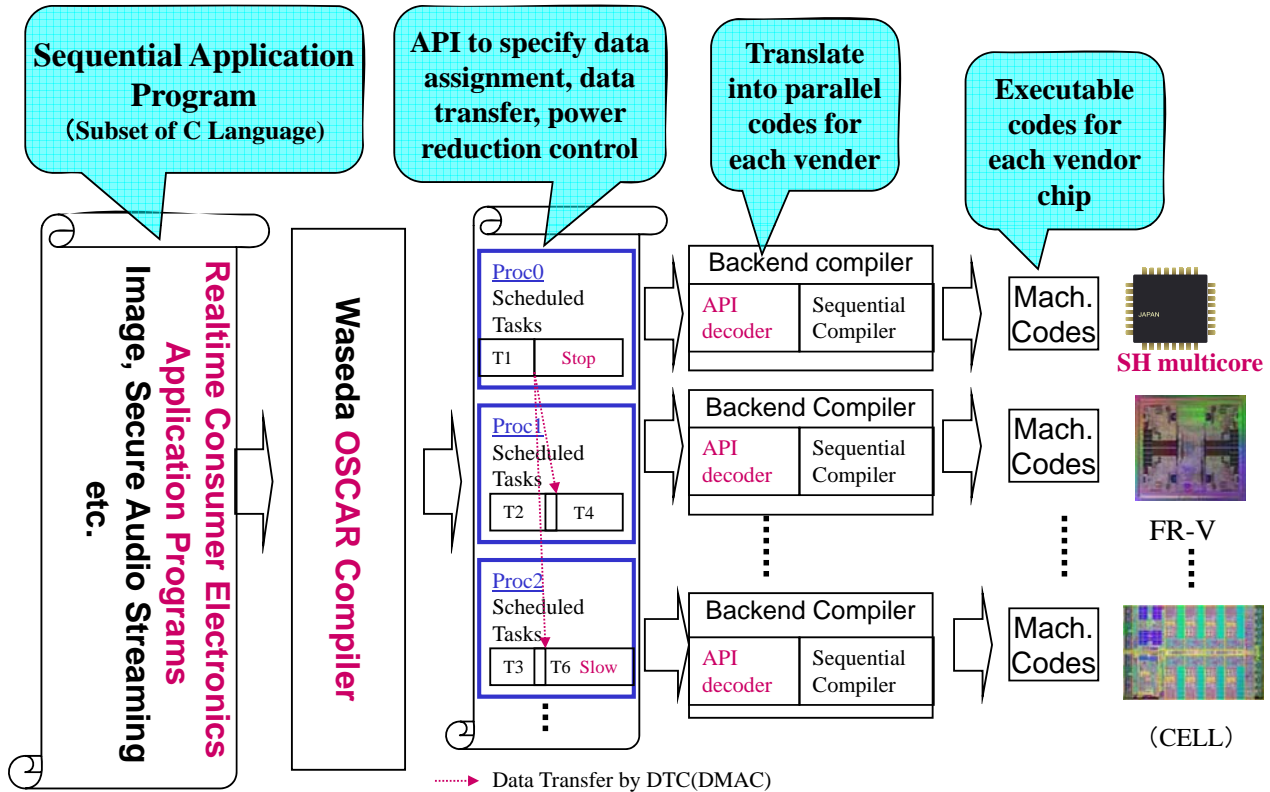
Performance of OSCAR Compiler Using the Multicore API on Intel Quad-core Xeon
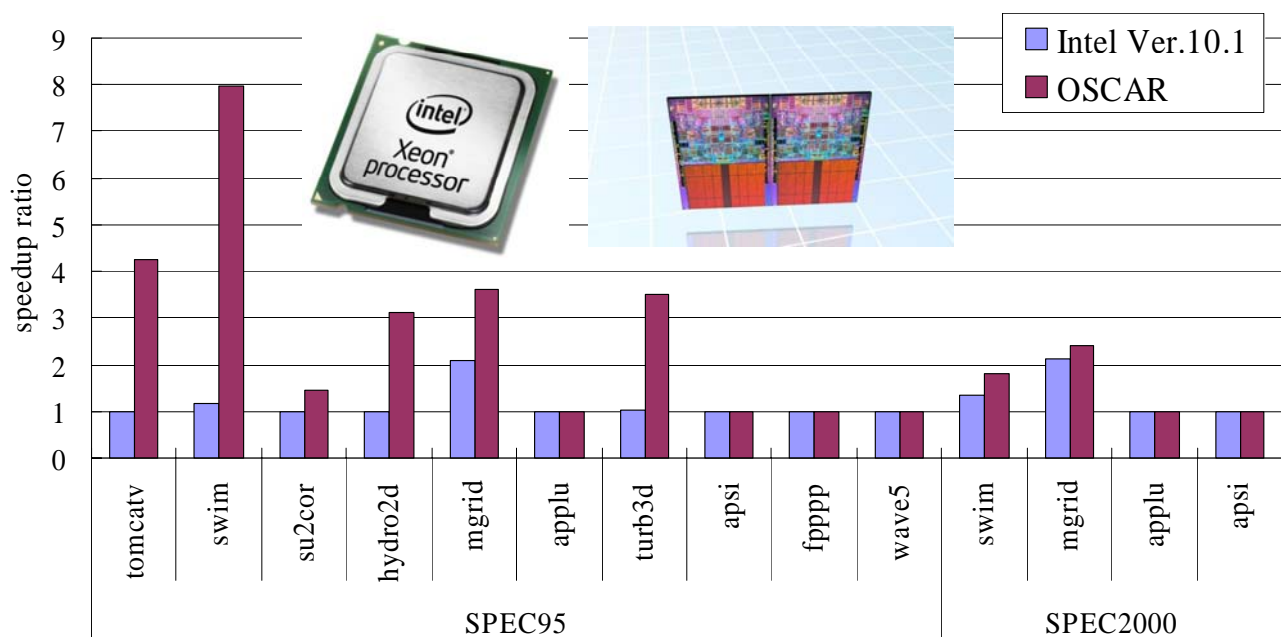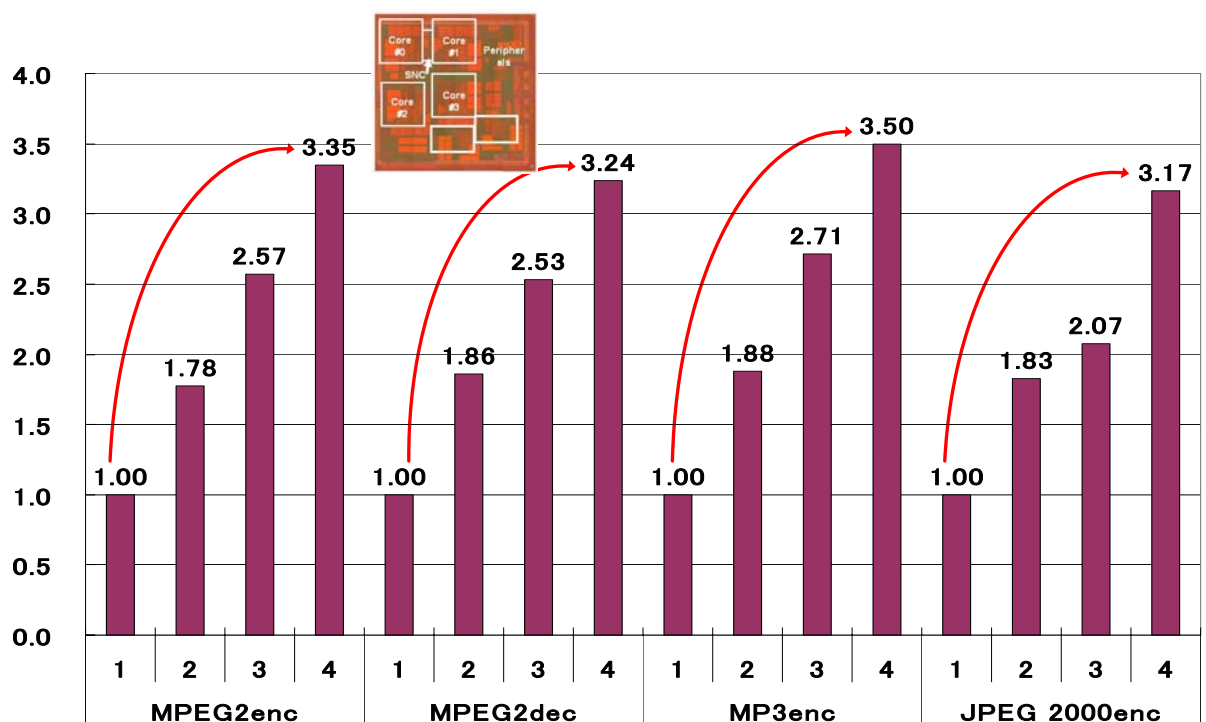

- **OSCAR Compiler gives us 2.09 times speedup on the average against Intel Compiler ver.10.1**

Performance of OSCAR Compiler Using the multicore API on Fujitsu FR1000 Multicore

3.38 times speedup on the average for 4 cores against a single core execution
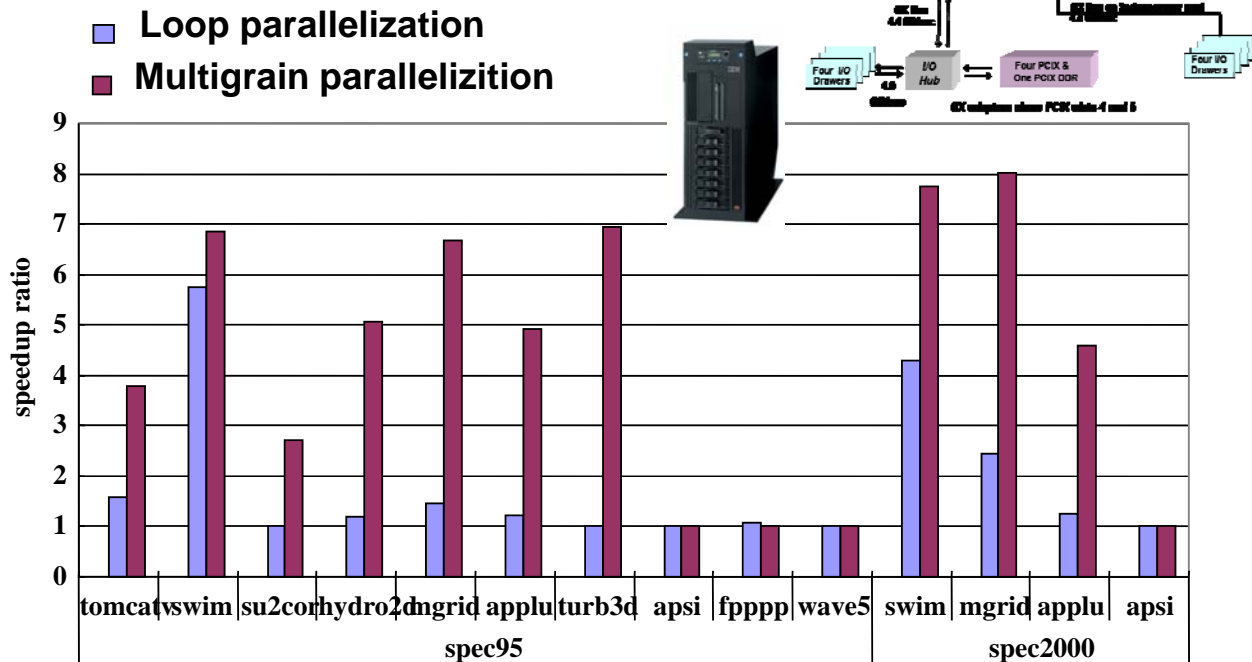
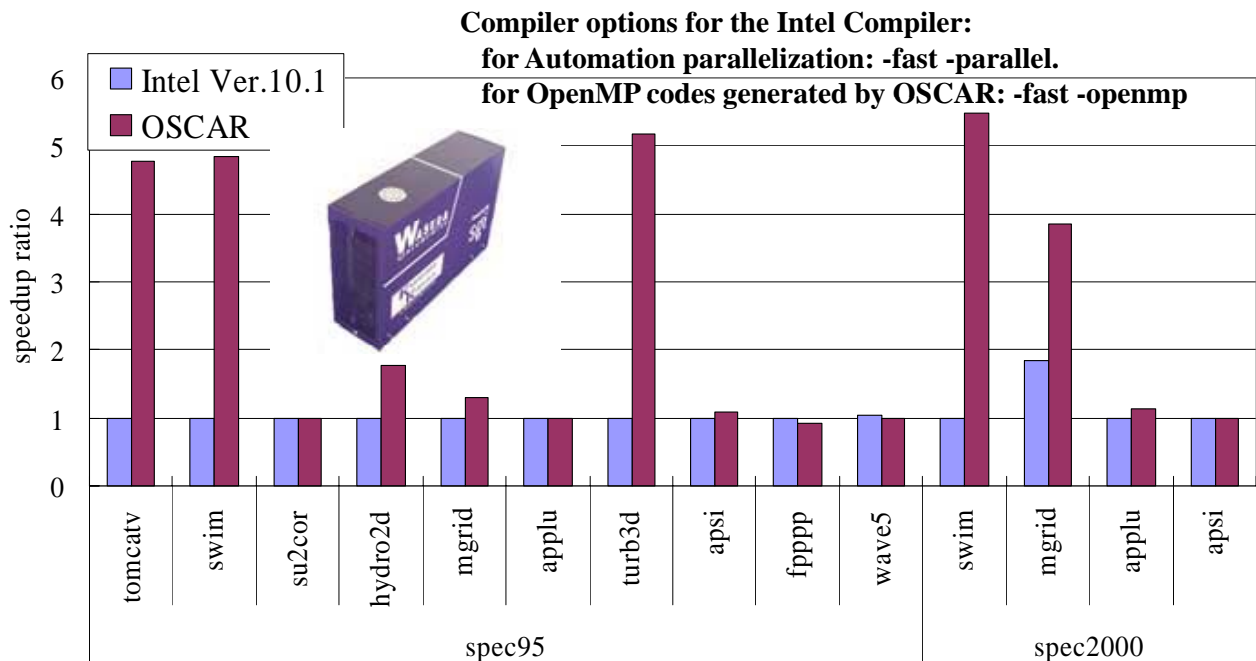# Performance of OSCAR Compiler Using the Developed API on 4 core (SH4A) OSCAR Type Multicore

3.31 times speedup on the average for 4cores against 1core

Performance OSCAR Multigrain Parallelizing Compiler on a IBM p550q 8core Deskside Server

- **2.7 times speedup against loop parallelizing compiler on 8 cores**

Performance of OSCAR compiler on 16 cores SGI Altix 450 Montvale server

Compiler options for the Intel Compiler:
for Automation parallelization: -fast -parallel.
for OpenMP codes generated by OSCAR: -fast -openmp

- **OSCAR compiler gave us 2.32 times speedup against Intel Fortran Itanium Compiler revision 10.1**

RP2 Chip Photo and Specifications
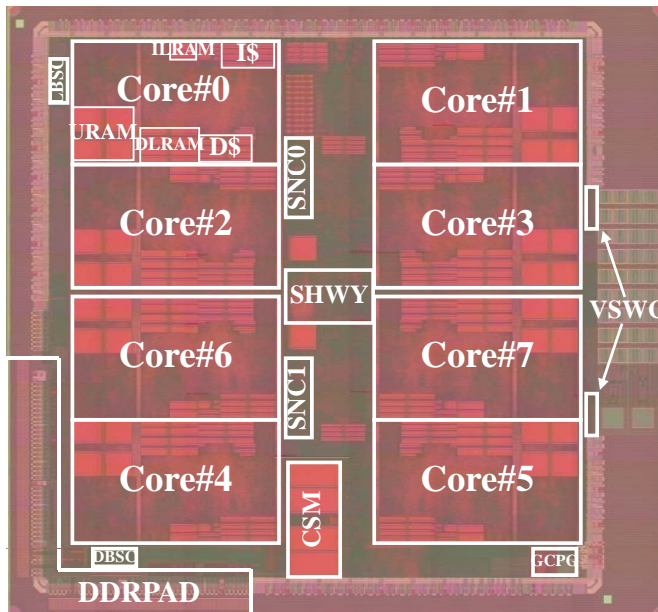
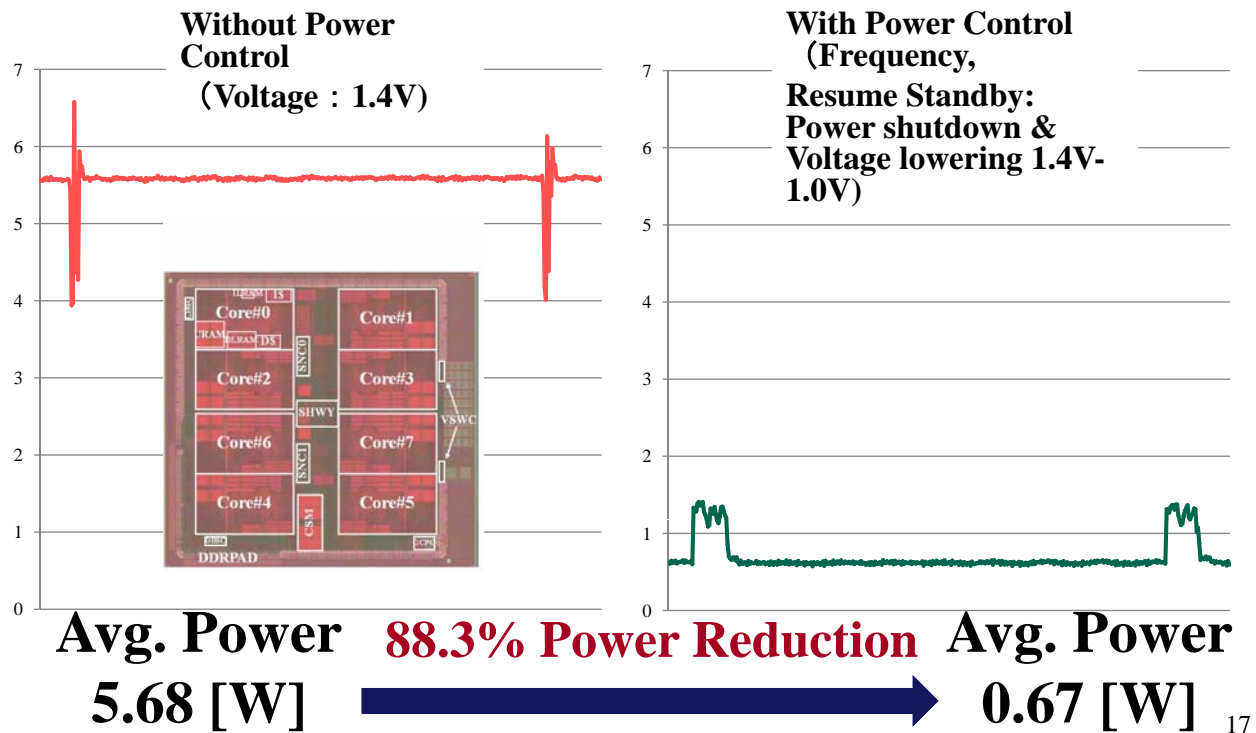| Process Technology	90nm, 8-layer, triple-Vth, CMOS
Chip Size	104.8mm^2 (10.61mm x 9.88mm)
CPU Core Size	6.6mm^2 (3.36mm x 1.96mm)
Supply Voltage	1.0V–1.4V (internal), 1.8/3.3V (I/O)
Power Domains	17 (8 CPUs, 8 URAMs, common)

Processing Performance on the Developed Multicore Using Automatic Parallelizing Compiler

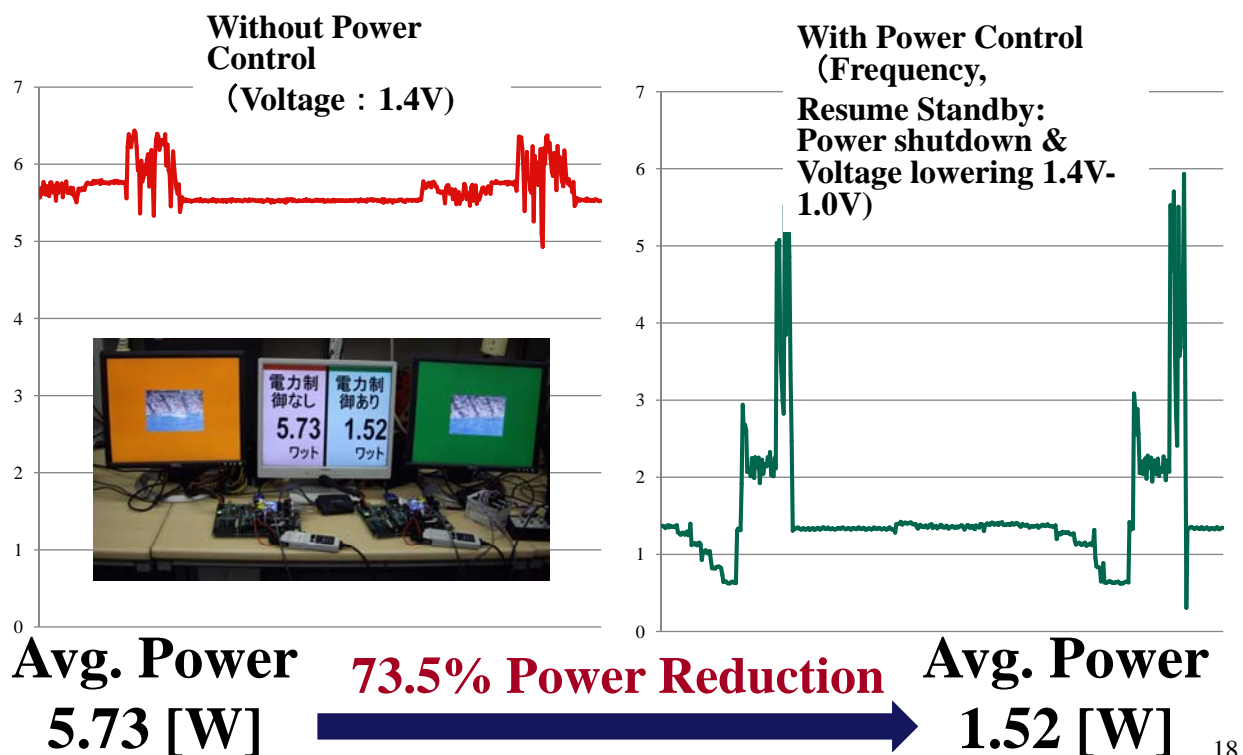### Speedup against single core execution for audio AAC encoding

*) Advanced Audio Coding
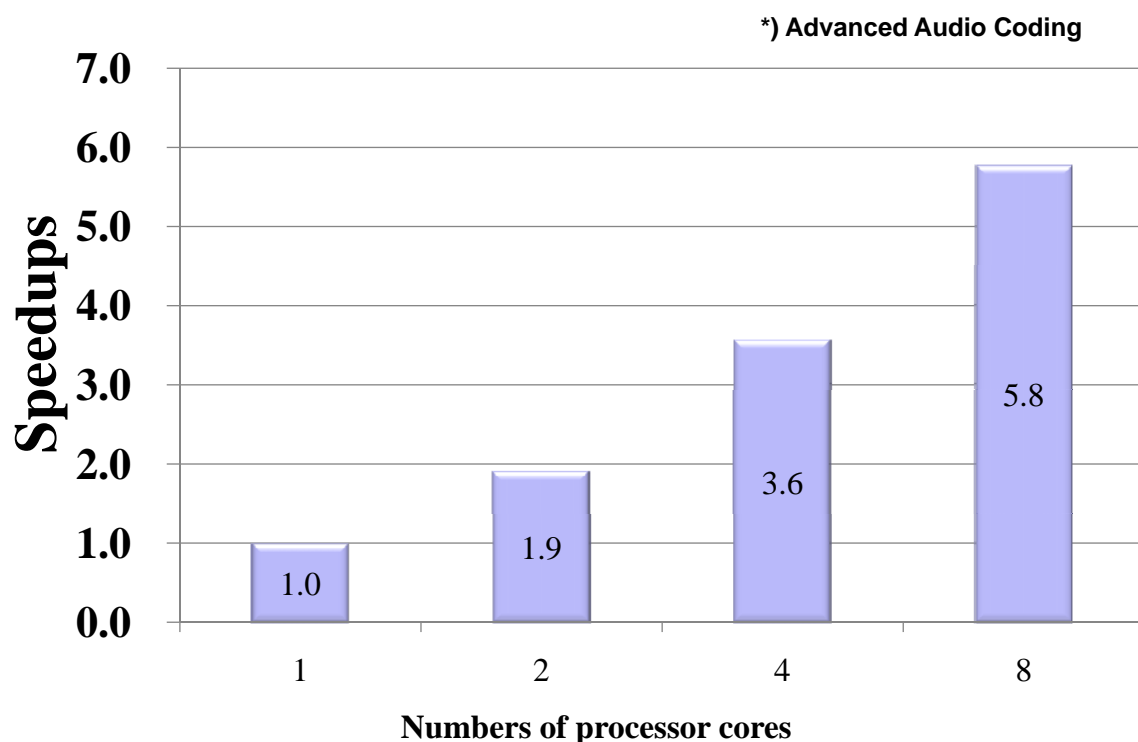
Power Reduction by OSCAR Parallelizing Compiler for Secure Audio Encoding

AAC Encoding + AES Encryption 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
5.68 [W]**

88.3% Power Reduction

**Avg. Power
0.67 [W]**

Power Reduction by OSCAR Parallelizing Compiler for MPEG2 Decoding

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
5.73 [W]**

73.5% Power Reduction

**Avg. Power
1.52 [W]**

Low Power High Performance Multicore Computer with Solar Panel

> **Clean Energy Autonomous**
> **Servers operational in deserts**
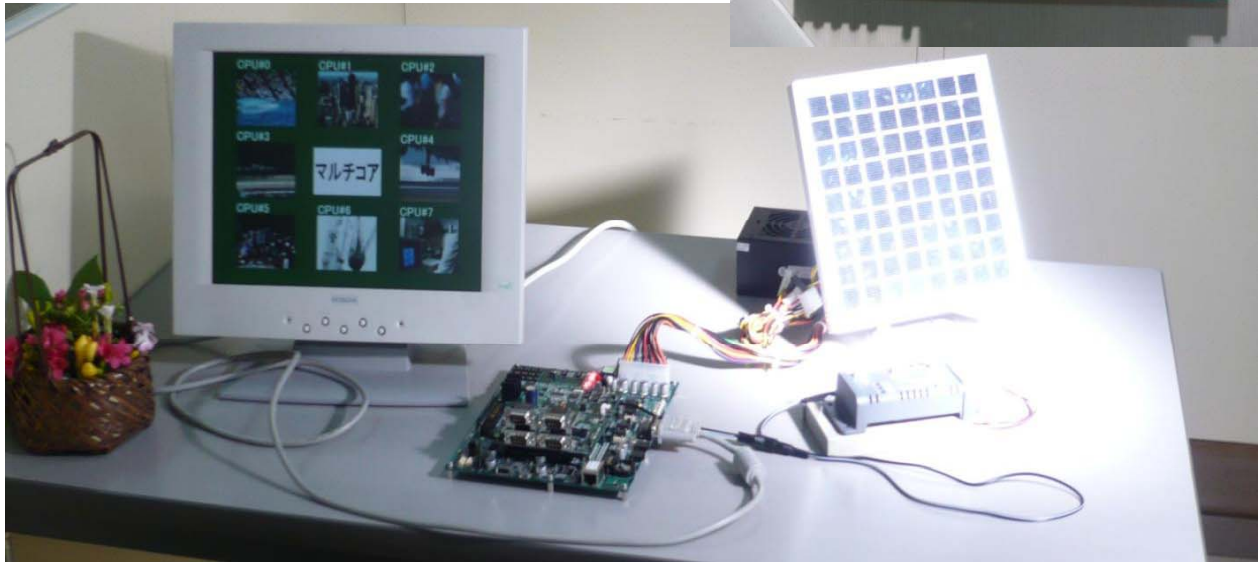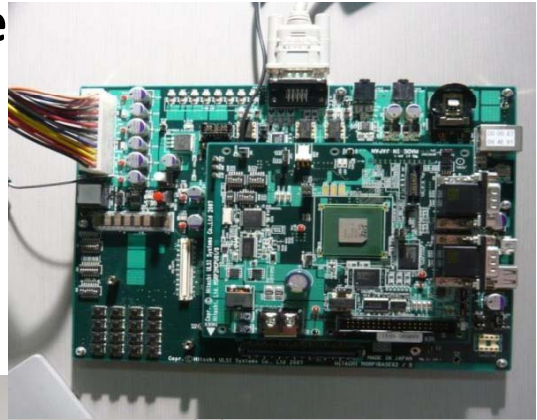
Conclusions

> Compiler cooperative low power, high effective performance, short software development period multi-core processors will be more important in wide range of information systems from embedded applications like games, mobile phones, digital TVs and automobiles to peta-scale supercomputers.

> Automatically generated parallel programs using the developed multicore API by OSCAR compiler give us the following performances:
>> 3.31 times speedup on 4core (SH4A) OSCAR type multicore
>> 3.38 times speedup on 4 core FR1000 against 1 core
>> 88% power reduction by the compiler power control on the developed 8 core (SH4A) multicore for realtime secure AAC encoding
>> 70% power reduction on the multicore for MPEG2 decoding