

# **Homogeneous and Heterogeneous Multicore / Manycore Processors, Parallelizing Compiler and Multiplatform API for Green Computing**

**Hironori Kasahara**

**Professor, Dept. of Computer Science & Engineering  
Director, Advanced Multicore Processor Research Institute**

**Waseda University, Tokyo, Japan**

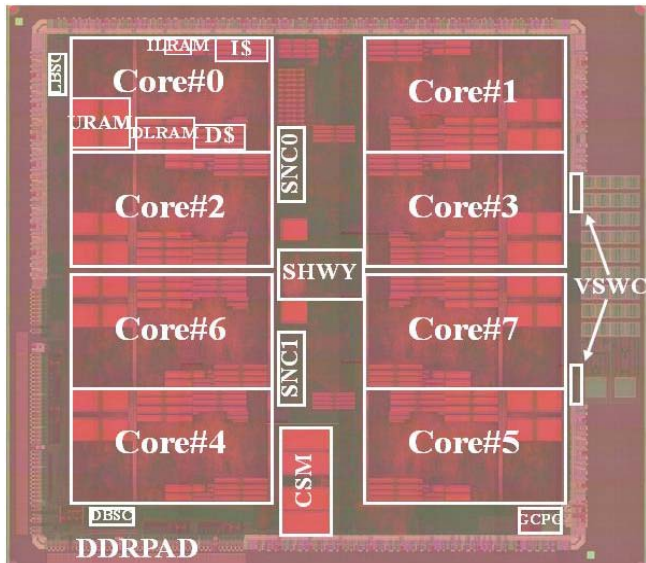
**IEEE Computer Society Board of Governors**

**URL: <http://www.kasahara.cs.waseda.ac.jp/>**

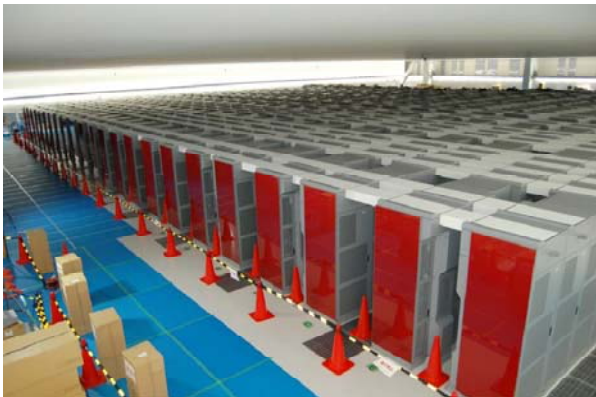
**MPSoC2011, July7, 2011**

# Multi/Many-core Everywhere

## Multi-core from embedded to supercomputers



OSCAR Type Multi-core Chip by Renesas in METI/NEDO Multicore for Real-time Consumer Electronics Project (Leader: Prof.Kasahara)



The 27th Top 500 (20.6.2011),  
**No.1**, Fujitsu “K” 548,352 cores  
 (Current Peak 8.774 PFLOPS)  
 LINPACK 8.162 PFLOPS (93.0%)

### ➤ Consumer Electronics (Embedded)

**Mobile Phone, Game, TV, Car Navigation, Camera,**

IBM/ Sony/ Toshiba Cell, Fujitsu FR1000,

Panasonic Uniphier, NEC/ARM MPCore/MP211/NaviEngine,

Renesas 4 core RP1, 8 core RP2, 15core Hetero RP-X,

Plurality HAL 64(Marvell), Tiler Tile64/ -Gx100(->1000cores),

DARPA UHPC (2017: 80GFLOPS/W)

### ➤ PCs, Servers

Intel Quad Xeon, Core 2 Quad, Montvale, Nehalem(8cores),

Larrabee(32cores), SCC(48cores), Night Corner(50 core+:22nm),

AMD Quad Core Opteron (8, 12 cores)

### ➤ WSs, Deskside & Highend Servers

IBM(Power4,5,6,7), Sun (SparcT1,T2), Fujitsu SPARC64fx8

### ➤ Supercomputers

Earth Simulator:**40TFLOPS**, 2002, 5120 vector proc.

BG/Q (A2:16cores) Water Cooled20PFLOPS, 3-4MW (2011-12),

BlueWaters(HPCS) Power7, 10 PFLOP+(2011.07),

Tianhe-1A (4.7PFLOPS,6coreX5670+ Nvidia Tesla M2050),

Godson-3B (1GHz40W 8core128GFLOPS) -T (64 core,192GFLOPS:2011)

RIKEN Fujitsu “K” 10PFLOPS(8core SPARC64VIIIfx, 128GGFLOPS)

**High quality application software, Productivity, Cost performance, Low power consumption are important**

**Ex, Mobile phones, Games**

**Compiler cooperated multi-core processors are promising to realize the above futures**

# 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.

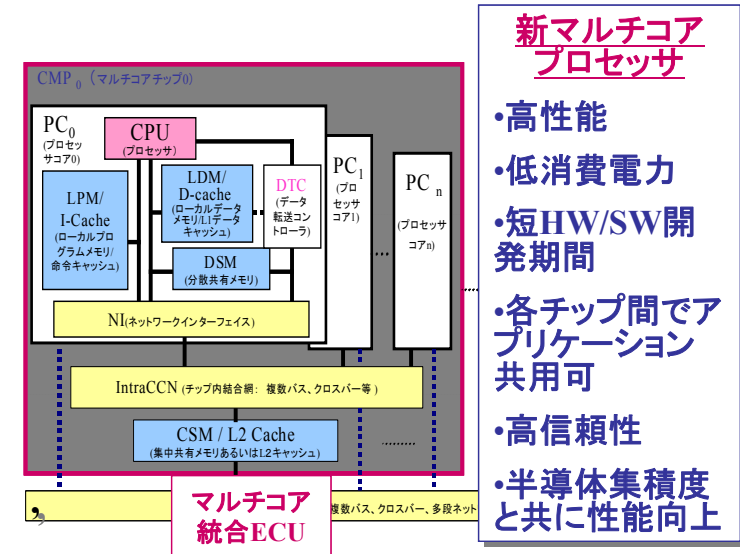
<Period> From July 2005 to March 2008

<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)\*\*



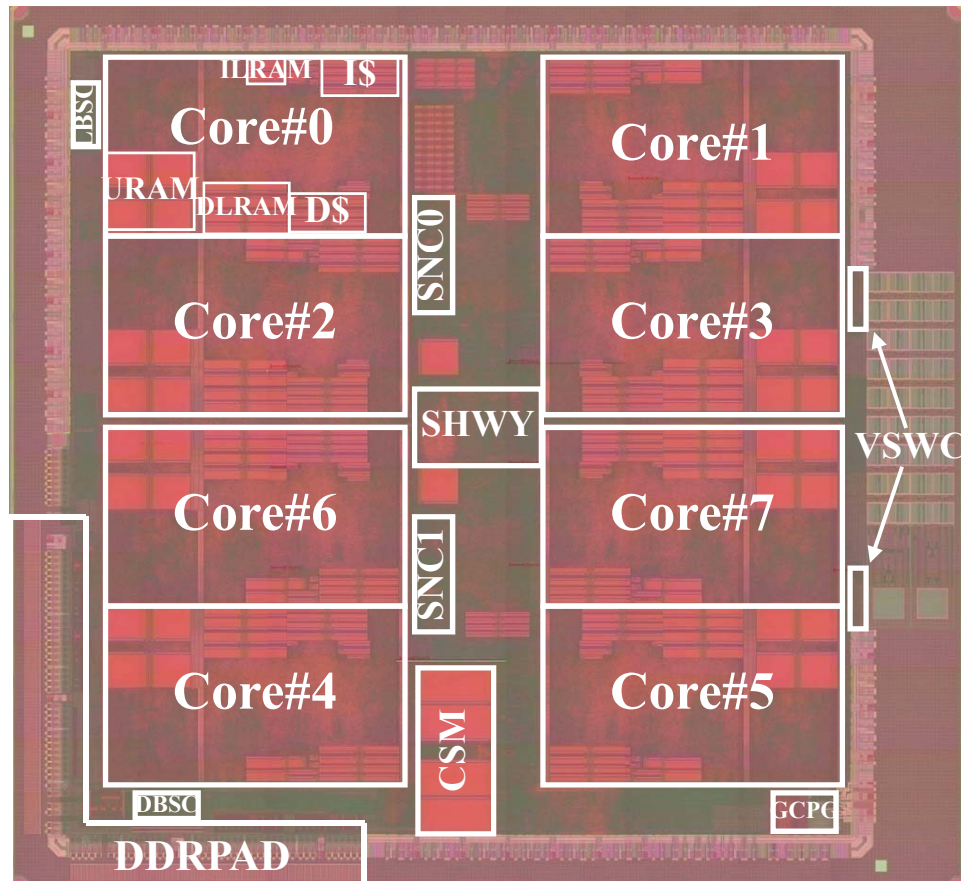
開発マルチコアチップは情報家電へ



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

# Renesas-Hitachi-Waseda Low Power 8 core RP2

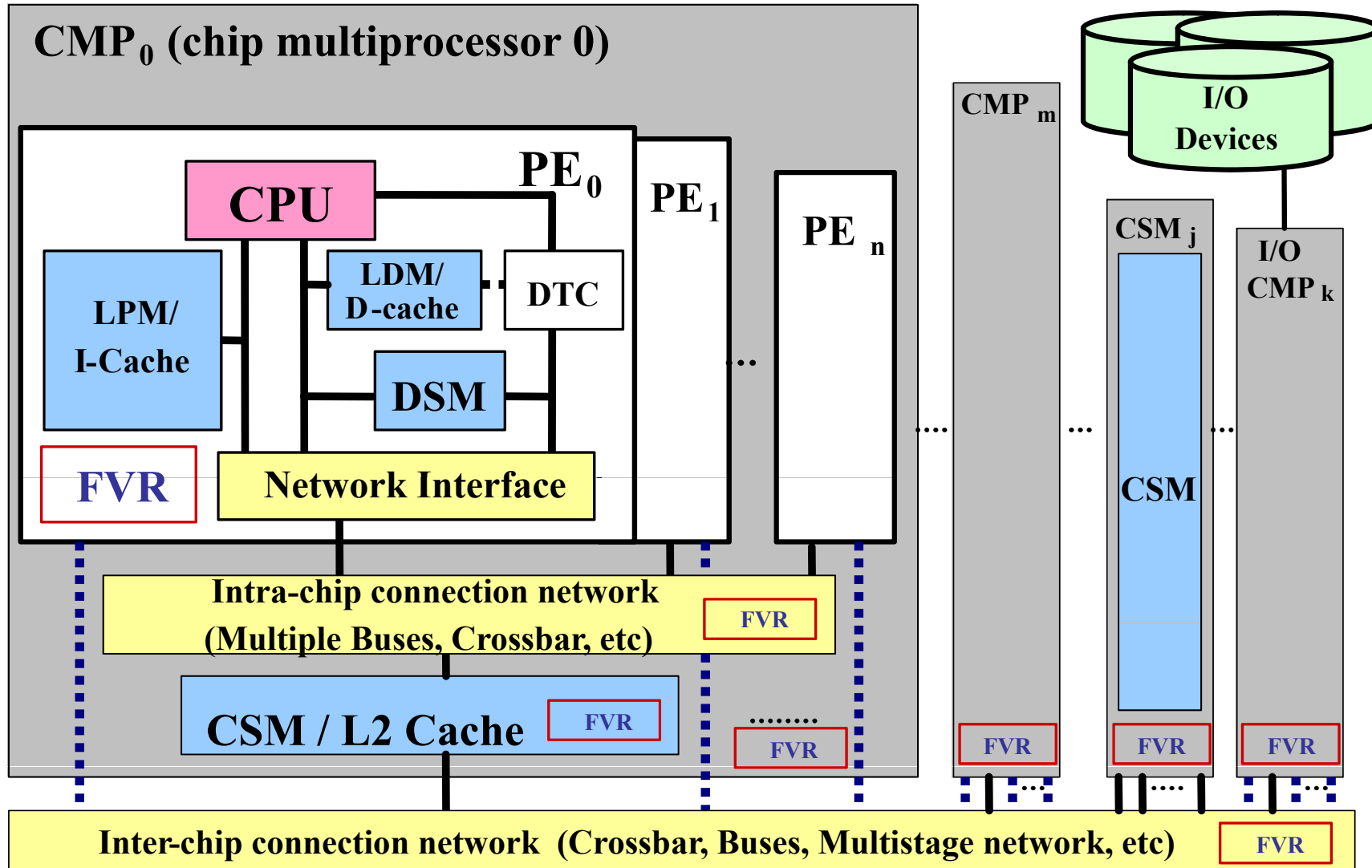
## Developed in 2007 in METI/NEDO project



Process Technology	90nm, 8-layer, triple-Vth, CMOS
Chip Size	104.8mm <sup>2</sup> (10.61mm x 9.88mm)
CPU Core Size	6.6mm <sup>2</sup> (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)

**IEEE ISSCC08: Paper No. 4.5, M.ITO, ... and H. Kasahara, “An 8640 MIPS SoC with Independent Power-off Control of 8 CPUs and 8 RAMs by an Automatic Parallelizing Compiler”**

# OSCAR Multi-Core Architecture



CSM: central shared mem.

DSM: distributed shared mem.

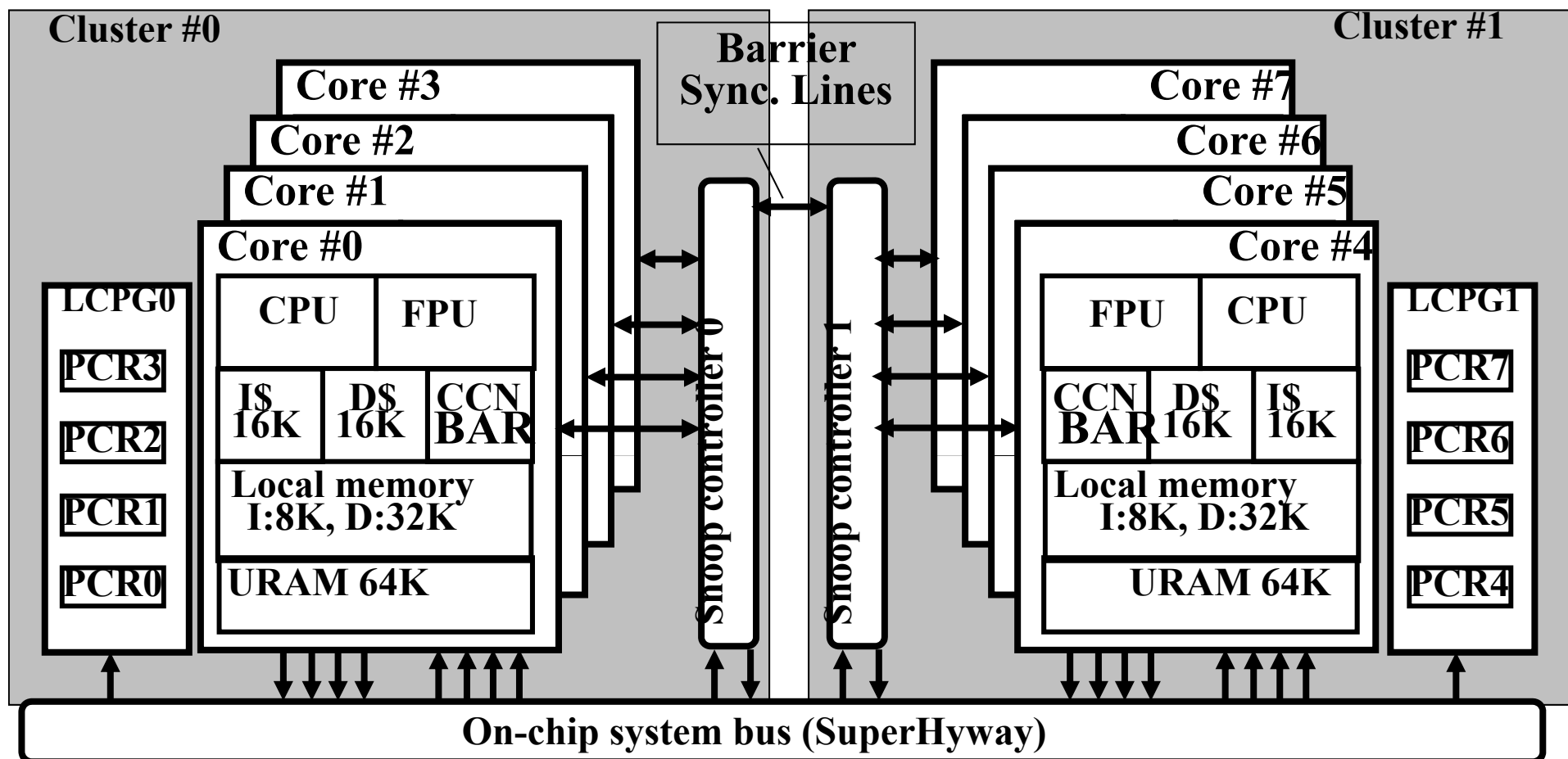
DTC: Data Transfer Controller

LDM : local data mem.

LPM : local program mem.

FVR: frequency / voltage control register

# 8 Core RP2 Chip Block Diagram



**LCPG:** Local clock pulse generator  
**PCR:** Power Control Register  
**CCN/BAR:** Cache controller/Barrier Register  
**URAM:** User RAM (Distributed Shared Memory)

# Demo of NEDO Multicore for Real Time Consumer Electronics at the Council of Science and Engineering Policy on April 10, 2008

第74回総合科学技術会議【平成20年4月10日】



第74回総合科学技術会議の様子(1)



第74回総合科学技術会議の様子(2)



第74回総合科学技術会議の様子(3)



第74回総合科学技術会議の様子(4)

## CSTP Members

**Prime Minister:**

**Mr. Y. FUKUDA**

**Minister of State for  
Science, Technology  
and Innovation  
Policy:**

**Mr. F. KISHIDA**

**Chief Cabinet  
Secretary:**

**Mr. N. MACHIMURA**

**Minister of Internal  
Affairs and  
Communications :**

**Mr. H. MASUDA**

**Minister of Finance :**

**Mr. F. NUKAGA**

**Minister of  
Education, Culture,  
Sports, Science and  
Technology:**

**Mr. K. TOKAI**

**Minister of  
Economy, Trade and  
Industry:**

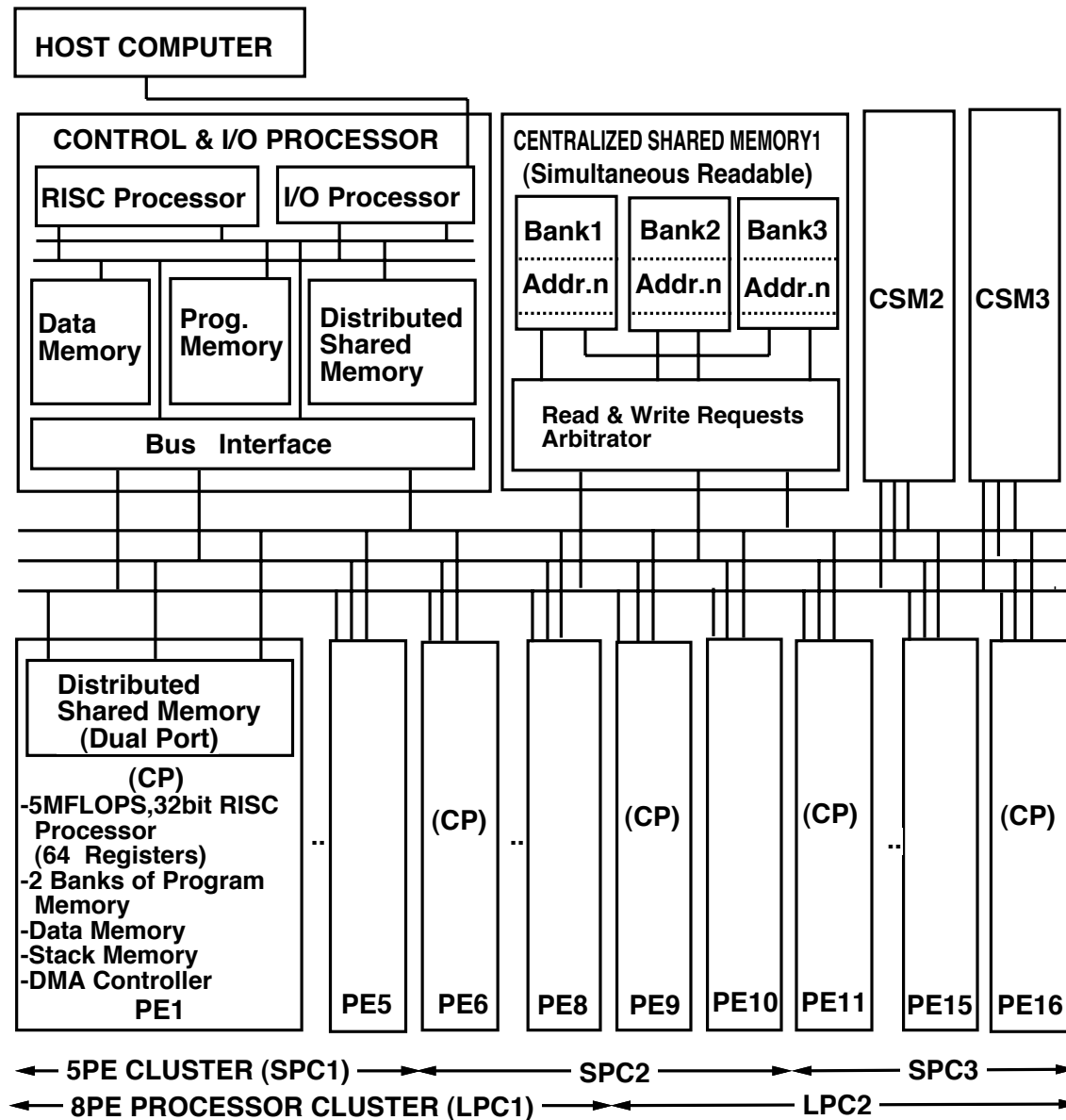
**Mr. A. AMARI**

# 1987 OSCAR(Optimally Scheduled Advanced Multiprocessor)



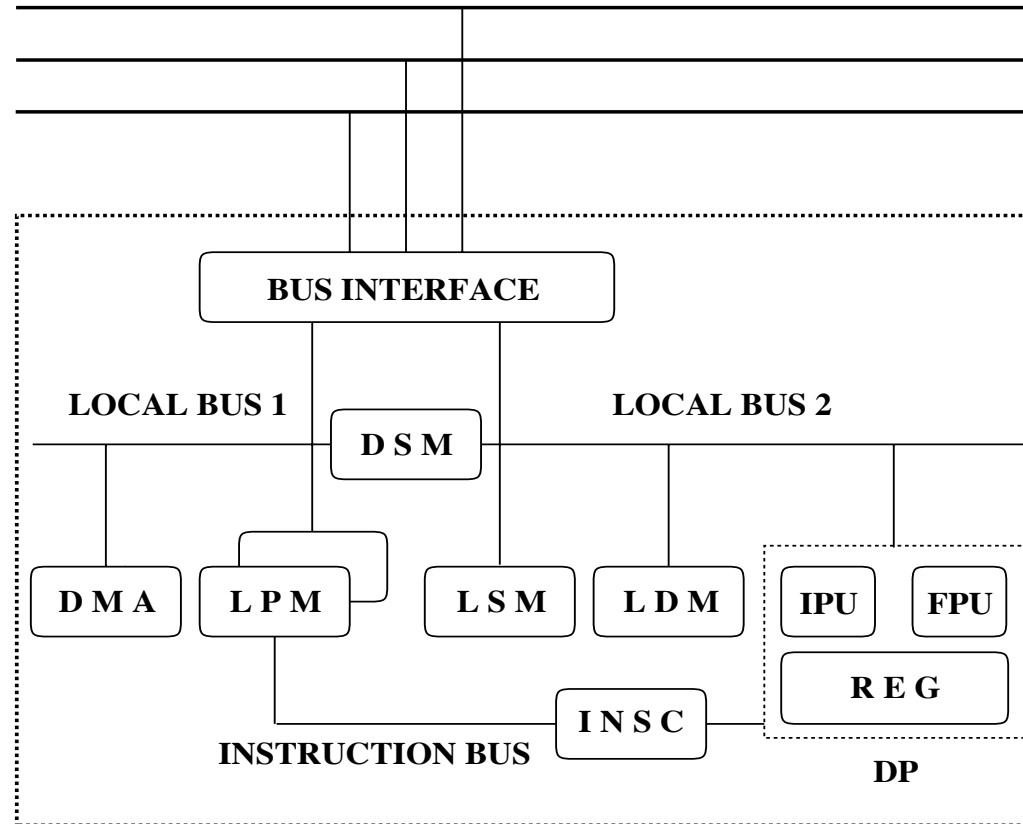


# OSCAR(Optimally Scheduled Advanced Multiprocessor)



# OSCAR PE (Processor Element)

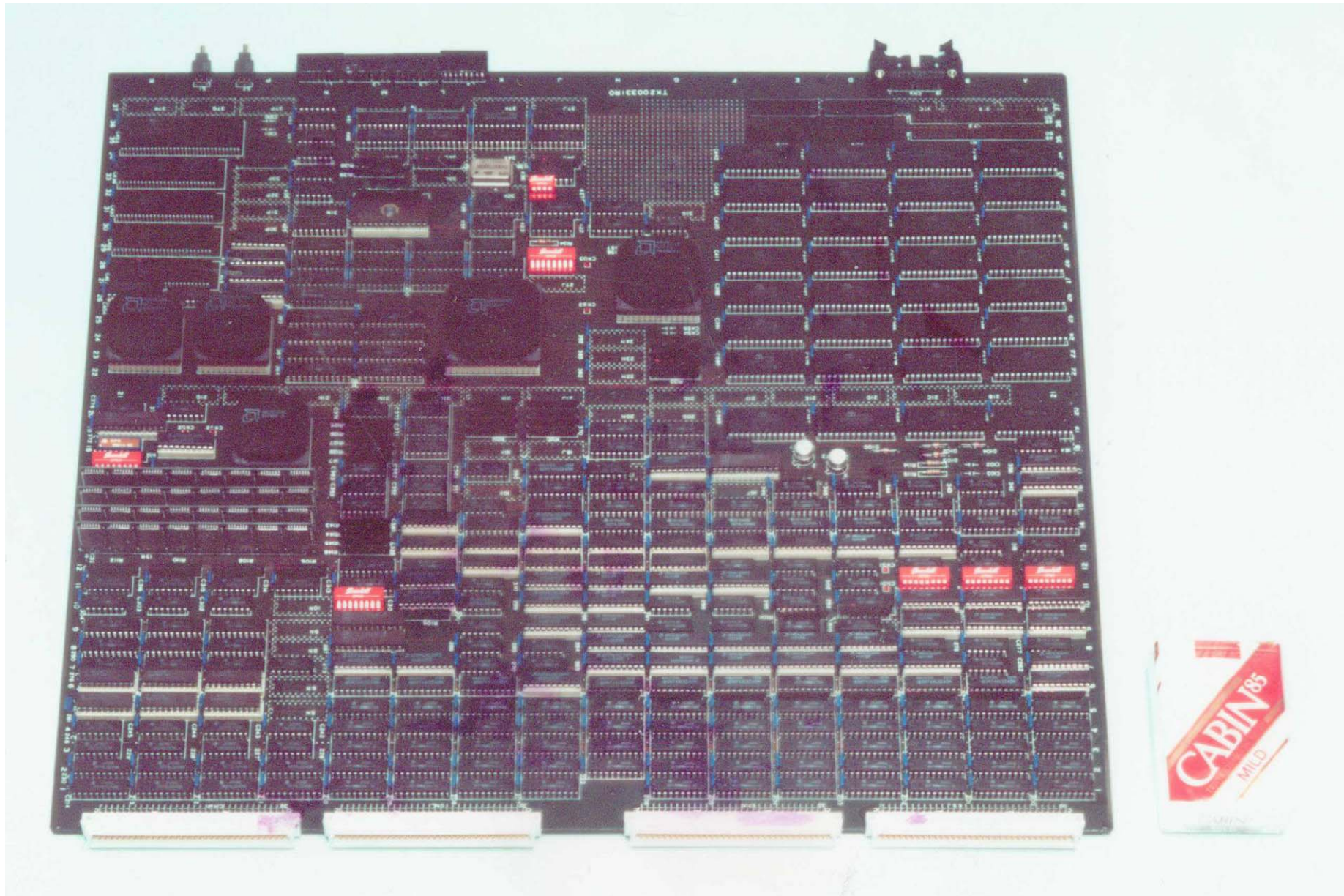
SYSTEM BUS



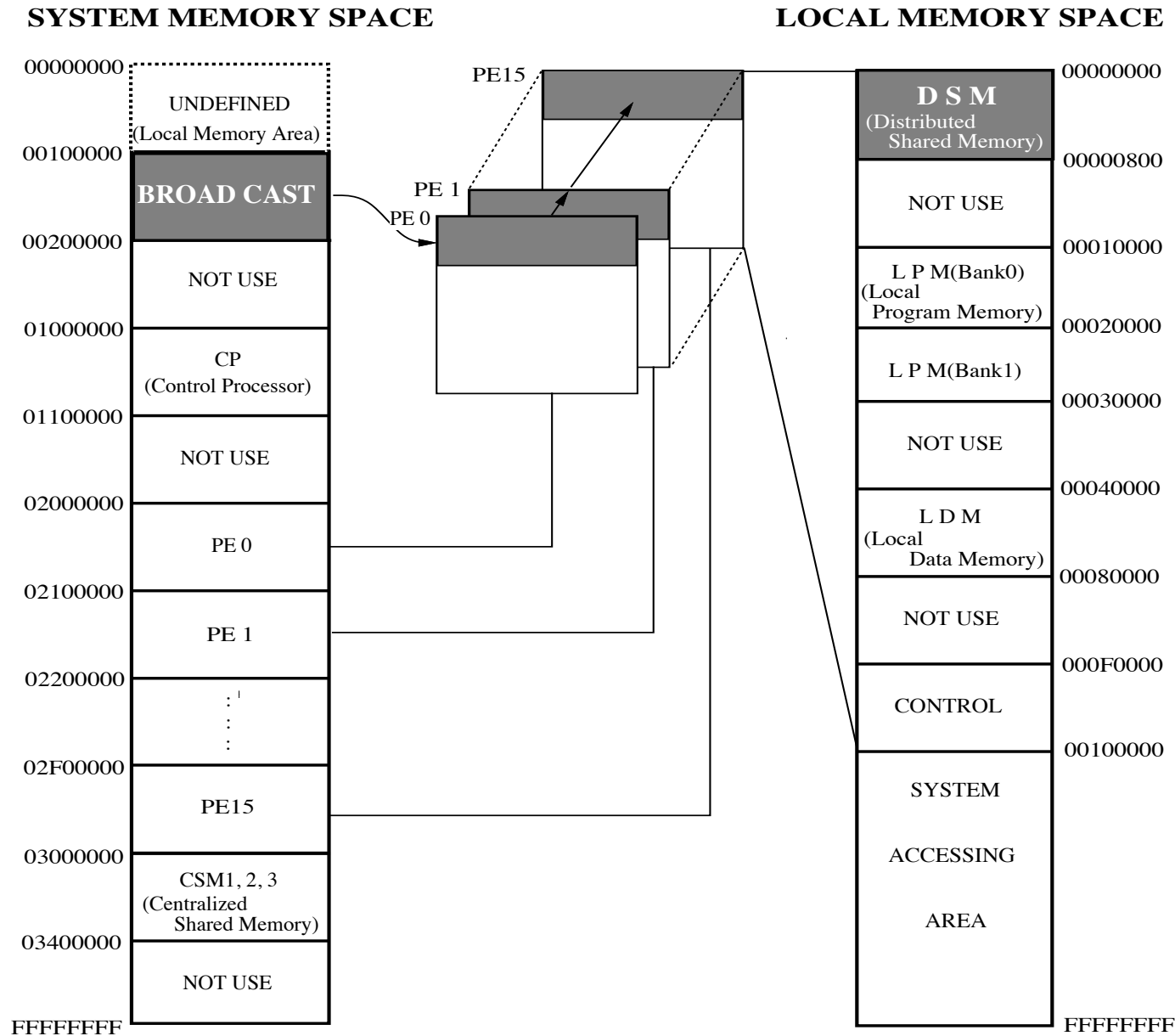
**DMA** : DMA CONTROLLER  
**LPM** : LOCAL PROGRAM MEMORY  
 (128KW \* 2BANK)  
**INSC** : INSTRUCTION  
 CONTROL UNIT  
**DSM** : DISTRIBUTED  
 SHARED MEMORY (2KW)  
**LSM** : LOCAL  
 STACK MEMORY (4KW)

**LDM** : LOCAL DATA MEMORY  
 (256KW)  
**DP** : DATA PATH  
**IPU** : INTEGER  
 PROCESSING UNIT  
**FPU** : FLOATING  
 PROCESSING UNIT  
**REG** : REGISTER FILE  
 (64 REGISTERS)

# 1987 OSCAR PE Board



# OSCAR Memory Space (Global and Local Address Space)



# 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

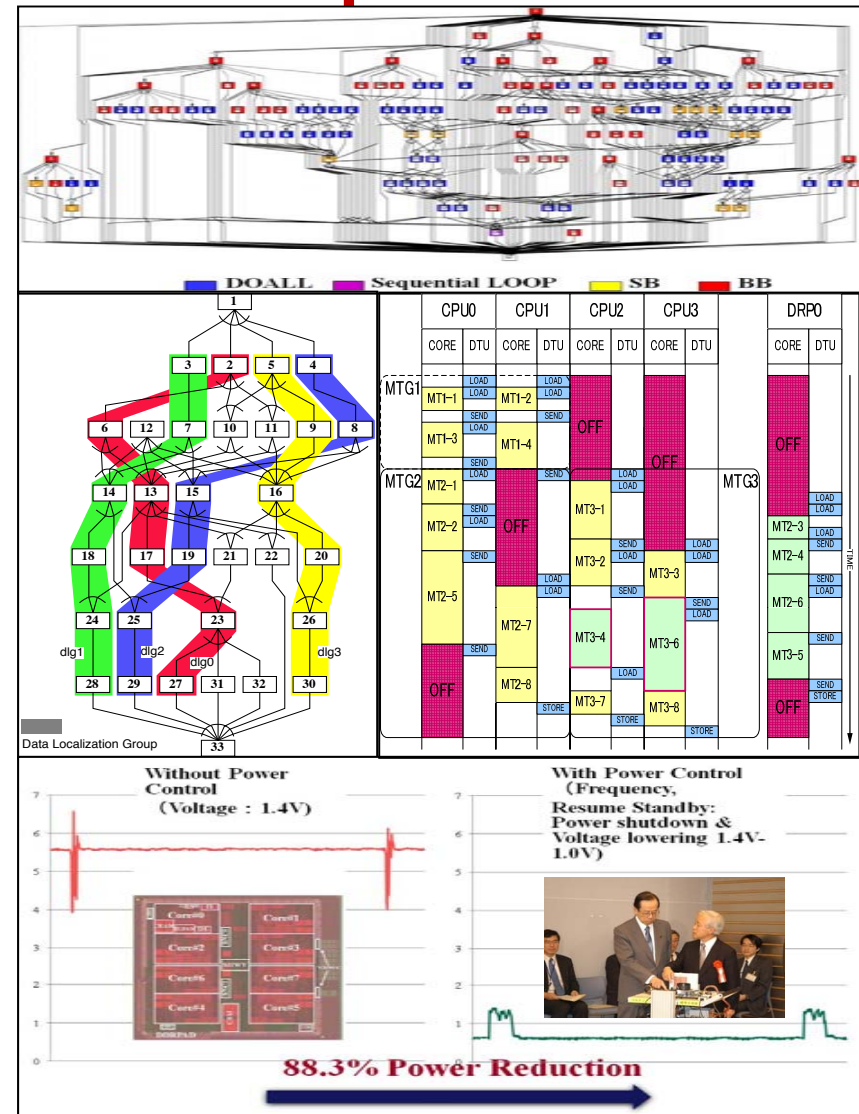
Automatic data management for distributed shared memory, cache and local memory

## Data Transfer Overlapping

Data transfer overlapping using Data Transfer Controllers (DMAs)

## Power Reduction

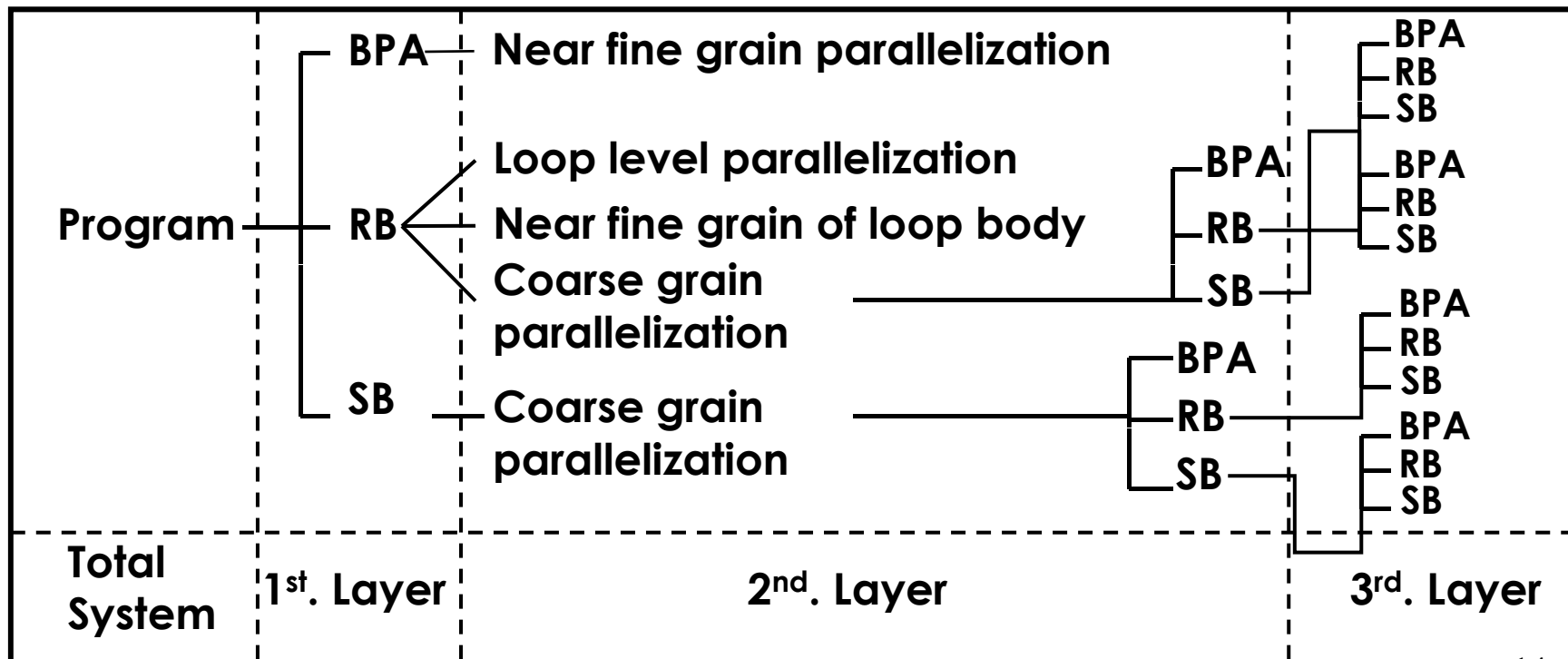
Reduction of consumed power by compiler control DVFS and Power gating with hardware supports.



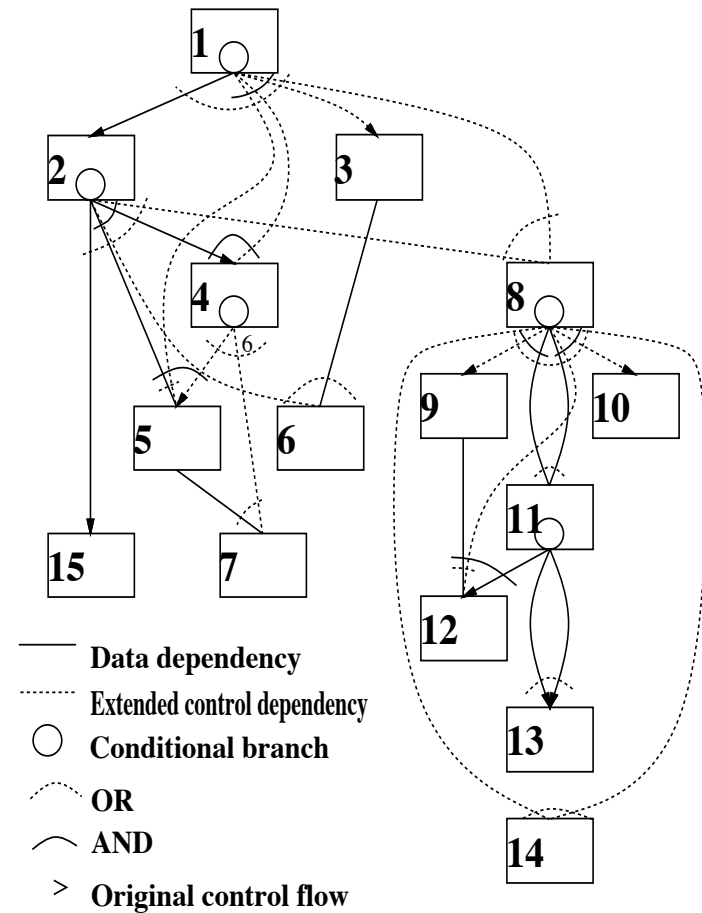
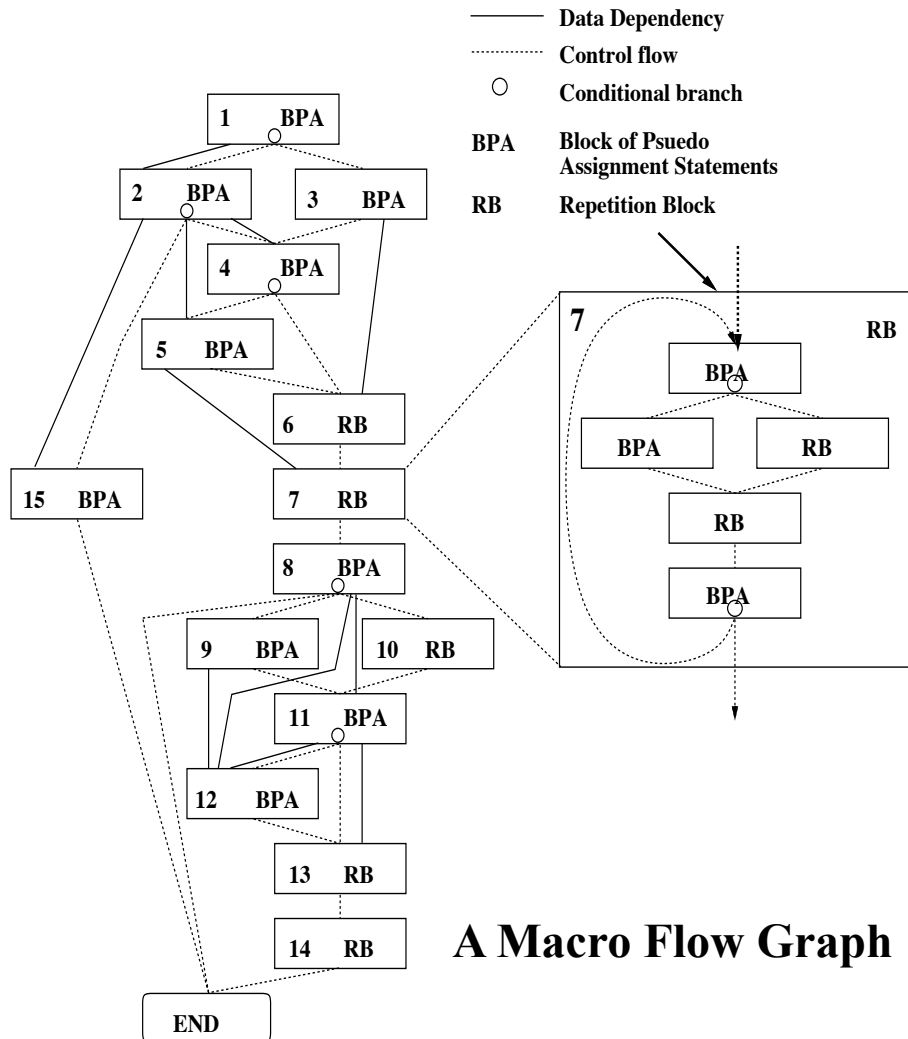
# Generation of coarse grain tasks

## ■ Macro-tasks (MTs)

- **Block of Pseudo Assignments (BPA): Basic Block (BB)**
- **Repetition Block (RB) : natural loop**
- **Subroutine Block (SB): subroutine**

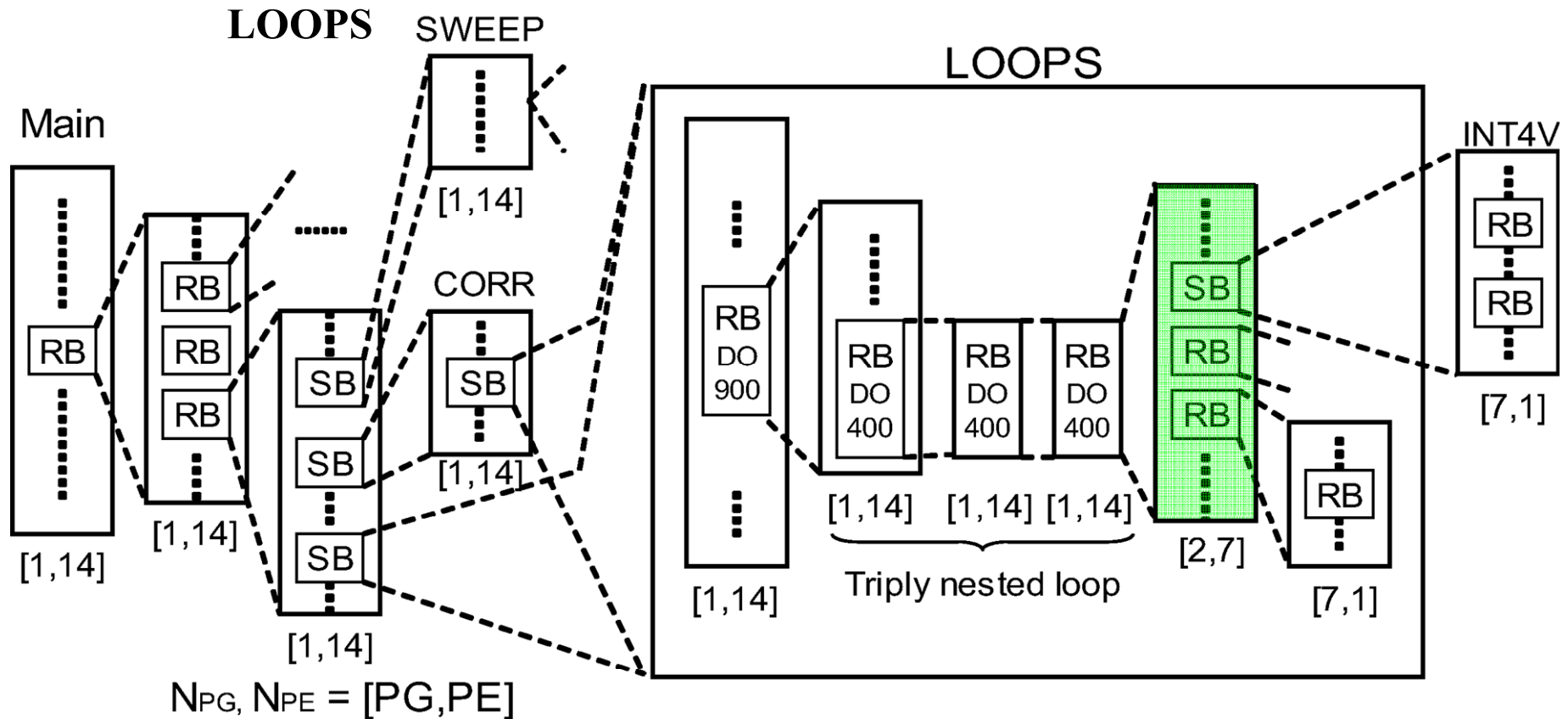


# Earliest Executable Condition Analysis for coarse grain tasks (Macro-tasks)



# Automatic processor assignment in su2cor

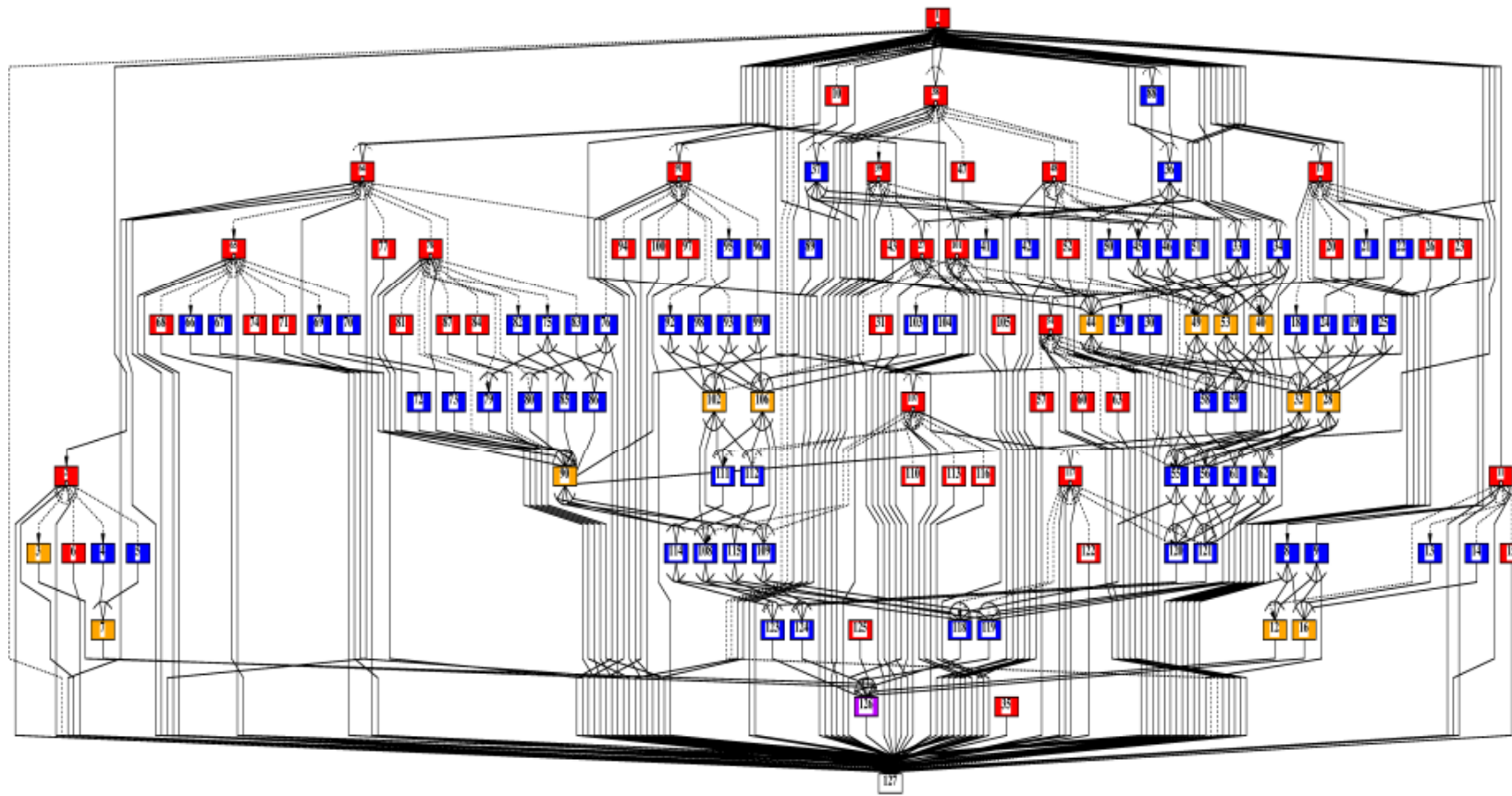
- Using 14 processors
  - Coarse grain parallelization within DO400 of subroutine





# MTG of Su2cor-LOOPS-DO400

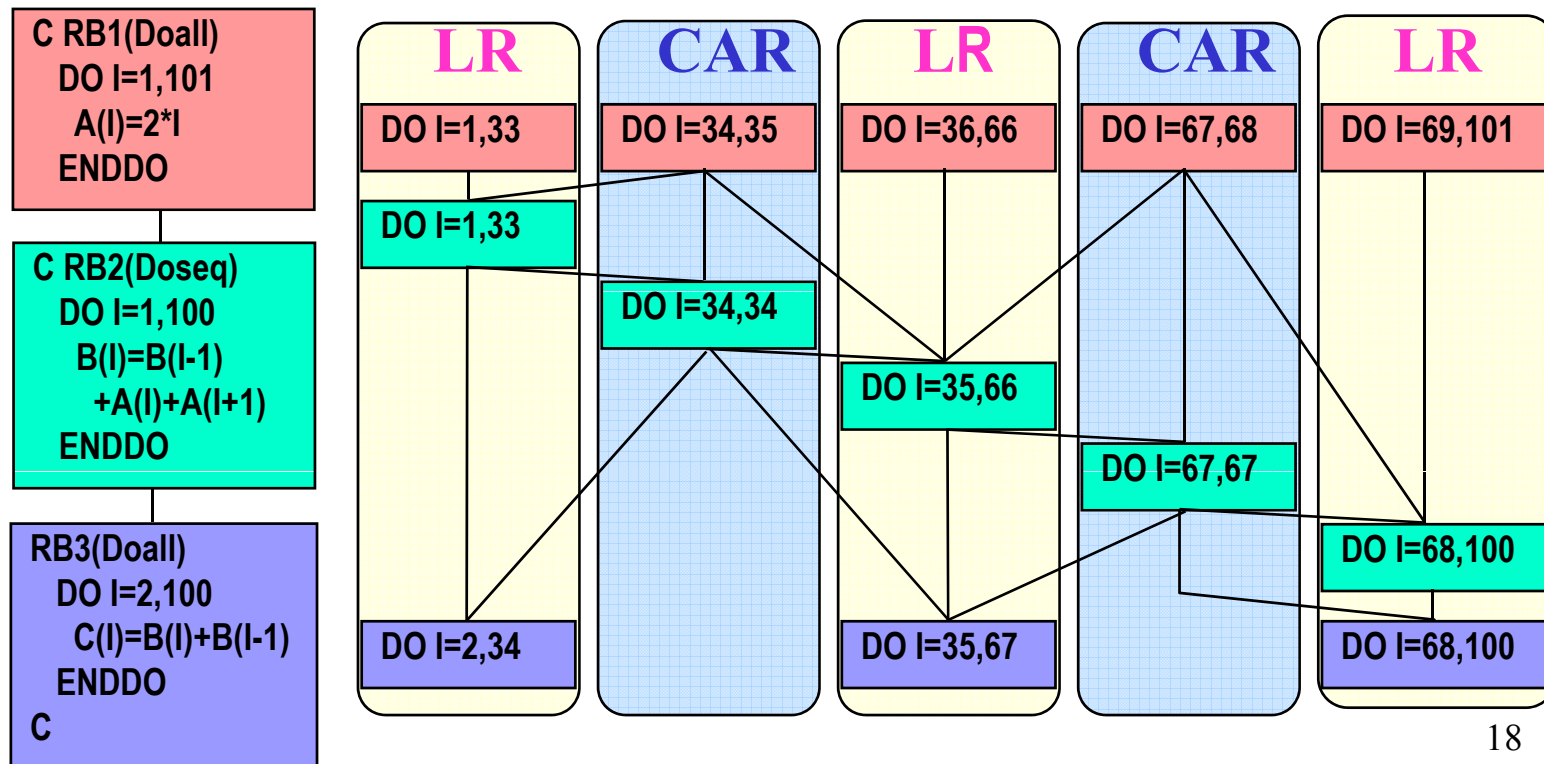
- Coarse grain parallelism  $\text{PARA\_ALD} = 4.3$



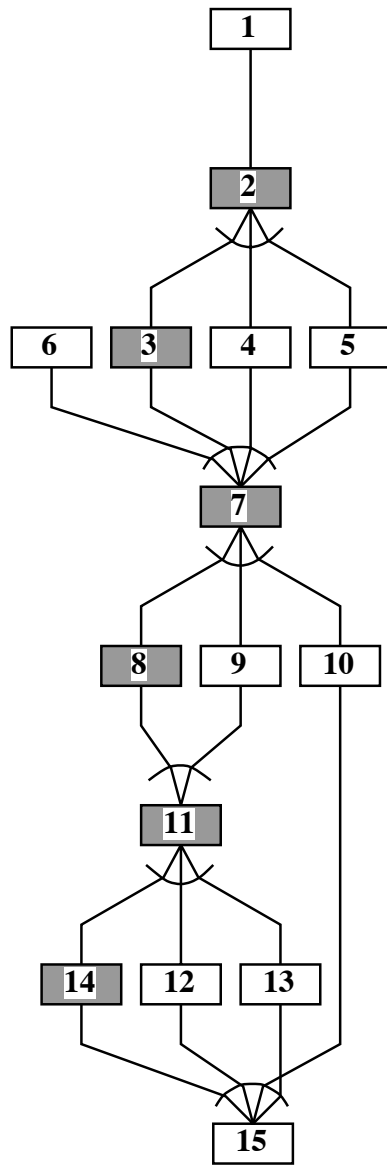
■ DOALL ■ Sequential LOOP ■ SB ■ BB

# Data-Localization Loop Aligned Decomposition

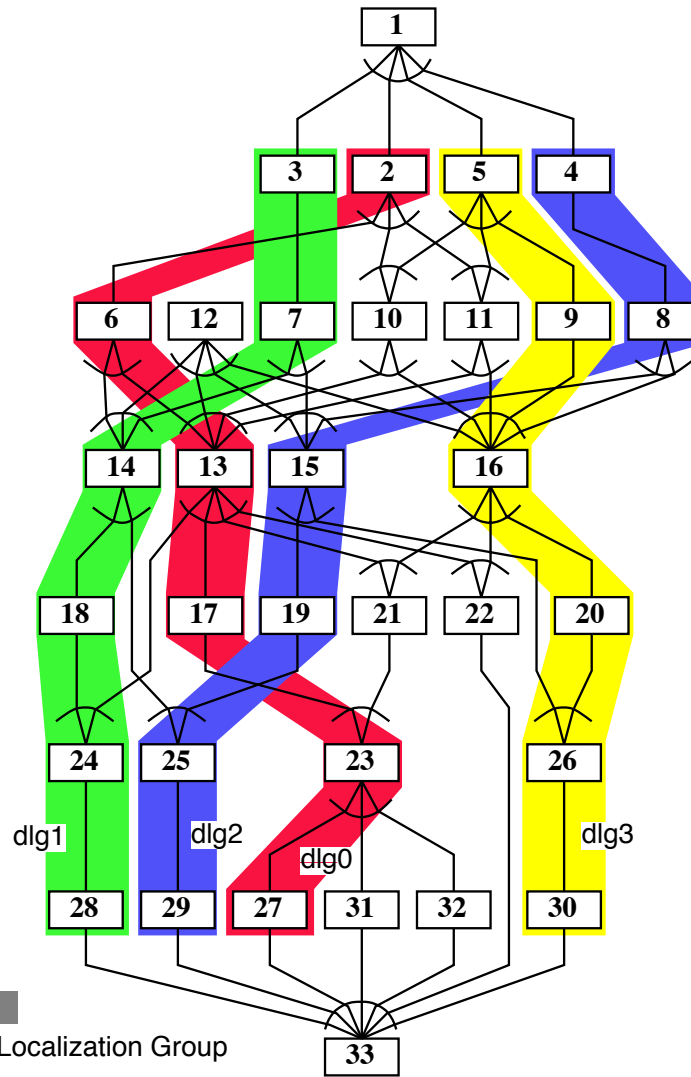
- Decompose multiple loop (Doall and Seq) into **CARs** and **LRs** considering inter-loop data dependence.
  - Most data in **LR** can be passed through LM.
  - LR**: Localizable Region, **CAR**: Commonly Accessed Region



# Data Localization



MTG



MTG after Division

PE0	PE1
12	1
2	3
6	7
4	14
8	18
15	5
19	9
25	11
29	10
13	16
17	20
22	26
21	30
23	24
27	28
	32
	31

A schedule for two processors

# An Example of Data Localization for Spec95 Swim

```

DO 200 J=1,N
DO 200 I=1,M
  UNEW(I+1,J) = UOLD(I+1,J)+
1  TDTSDY*(Z(I+1,J+1)+Z(I+1,J))*(CV(I+1,J+1)+CV(I,J+1)+CV(I,J)
2  +CV(I+1,J))-TDTSDX*(H(I+1,J)-H(I,J))
  VNEW(I,J+1) = VOLD(I,J+1)-TDTSDY*(Z(I+1,J+1)+Z(I,J+1))
1  *(CV(I+1,J+1)+CV(I,J+1)+CV(I,J)+CV(I+1,J))
2  -TDTSDX*(H(I,J+1)-H(I,J))
  PNEW(I,J) = POLD(I,J)-TDTSDY*(CV(I+1,J)-CV(I,J))
1  -TDTSDX*(H(I,J+1)-H(I,J))
200 CONTINUE
  
```

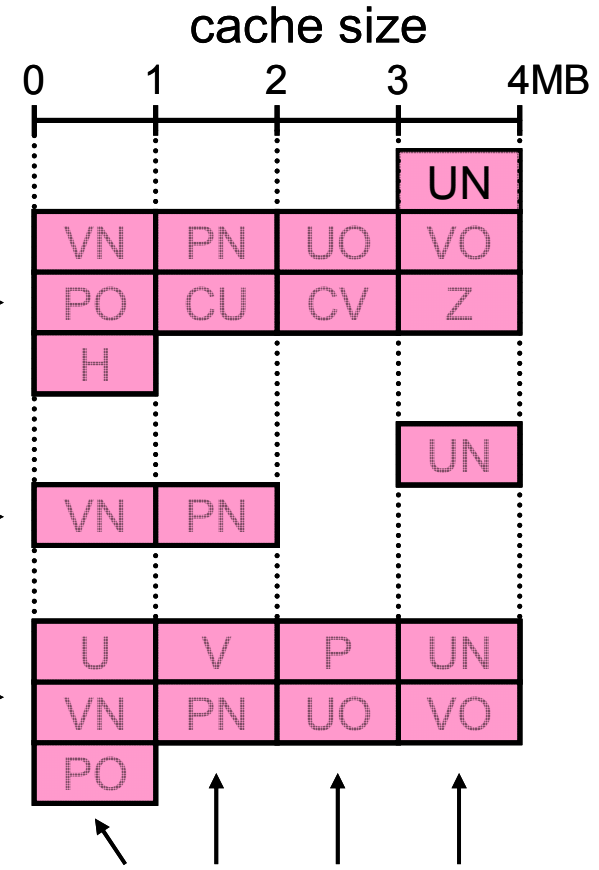
```

DO 210 J=1,N
  UNEW(1,J) = UNEW(M+1,J)
  VNEW(M+1,J+1) = VNEW(1,J+1)
  PNEW(M+1,J) = PNEW(1,J)
210 CONTINUE
  
```

```

DO 300 J=1,N
DO 300 I=1,M
  UOLD(I,J) = U(I,J)+ALPHA*(UNEW(I,J)-2.*U(I,J)+UOLD(I,J))
  VOLD(I,J) = V(I,J)+ALPHA*(VNEW(I,J)-2.*V(I,J)+VOLD(I,J))
  POLD(I,J) = P(I,J)+ALPHA*(PNEW(I,J)-2.*P(I,J)+POLD(I,J))
300 CONTINUE
  
```

(a) An example of target loop group for data localization



Cache line conflicts occurs among arrays which share the same location on cache

(b) Image of alignment of arrays on cache accessed by target loops

# Data Layout for Removing Line Conflict Misses by Array Dimension Padding

Declaration part of arrays in spec95 swim

before padding

after padding

PARAMETER (N1=513, N2=513)

PARAMETER (N1=513, N2=544)

COMMON U(N1,N2), V(N1,N2), P(N1,N2),

COMMON U(N1,N2), V(N1,N2), P(N1,N2),

\* UNEW(N1,N2), VNEW(N1,N2),

\* UNEW(N1,N2), VNEW(N1,N2),

1 PNEW(N1,N2), UOLD(N1,N2),

1 PNEW(N1,N2), UOLD(N1,N2),

\* VOLD(N1,N2), POLD(N1,N2),

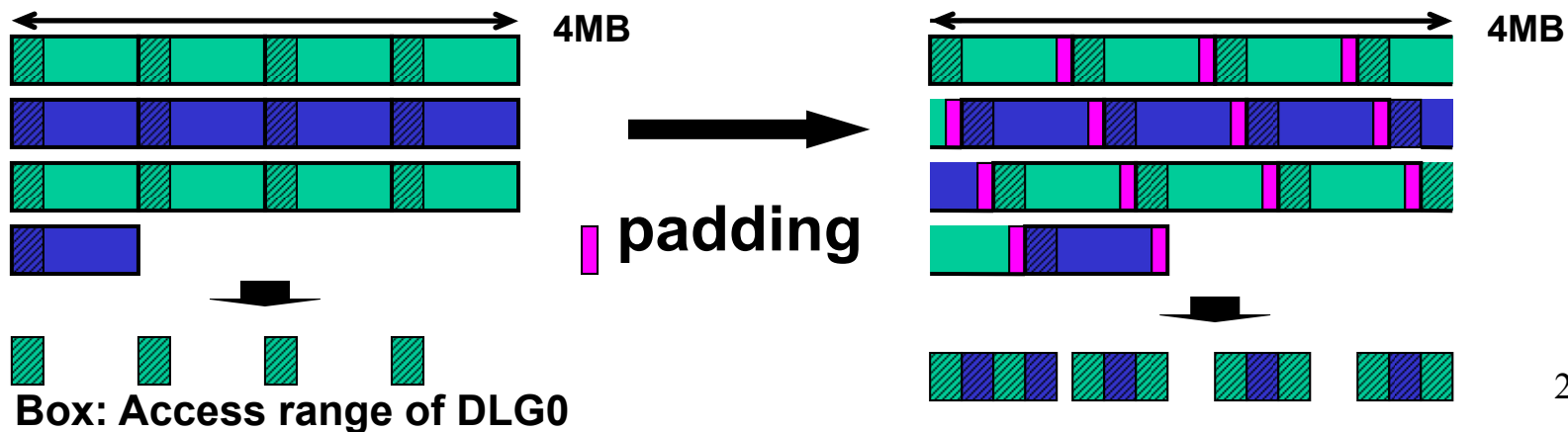
\* VOLD(N1,N2), POLD(N1,N2),

2 CU(N1,N2), CV(N1,N2),

2 CU(N1,N2), CV(N1,N2),

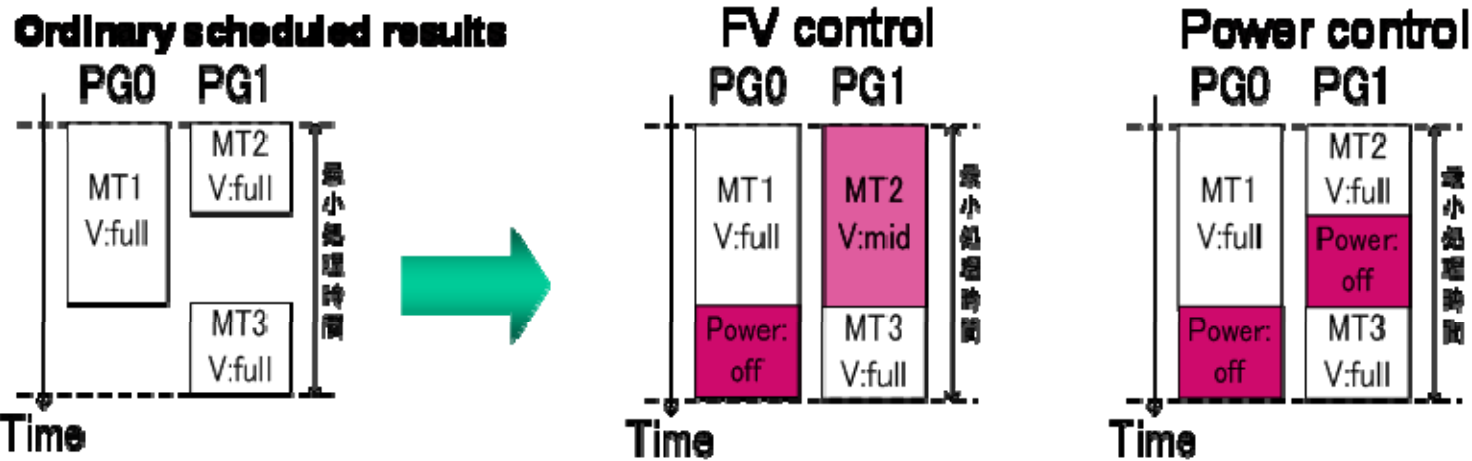
\* Z(N1,N2), H(N1,N2)

\* Z(N1,N2), H(N1,N2)

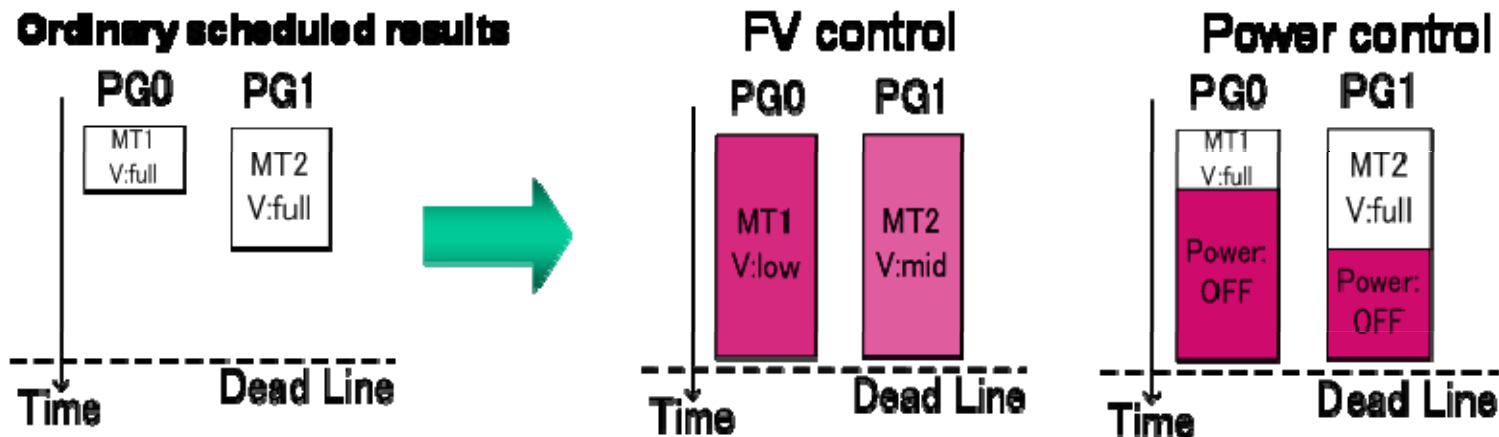


# Power Reduction by Power Supply, Clock Frequency and Voltage Control by OSCAR Compiler

- Shortest execution time mode



- Realtime processing mode with dead line constraints



# An Example of Machine Parameters for the Power Saving Scheme

- **Functions of the multiprocessor**

- Frequency of each proc. is changed to several levels
- Voltage is changed together with frequency
- Each proc. can be powered on/off

state	FULL	MID	LOW	OFF
frequency	1	1 / 2	1 / 4	0
voltage	1	0.87	0.71	0
dynamic energy	1	3 / 4	1 / 2	0
static power	1	1	1	0

- **State transition overhead (Example: not for RP2)**

state	FULL	MID	LOW	OFF
FULL	0	40k	40k	80k
MID	40k	0	40k	80k
LOW	40k	40k	0	80k
OFF	80k	80k	80k	0

delay time [u.t.]

state	FULL	MID	LOW	OFF
FULL	0	20	20	40
MID	20	0	20	40
LOW	20	20	0	40
OFF	40	40	40	0

energy overhead [ $\mu$ J]

# Power Reduction Scheduling

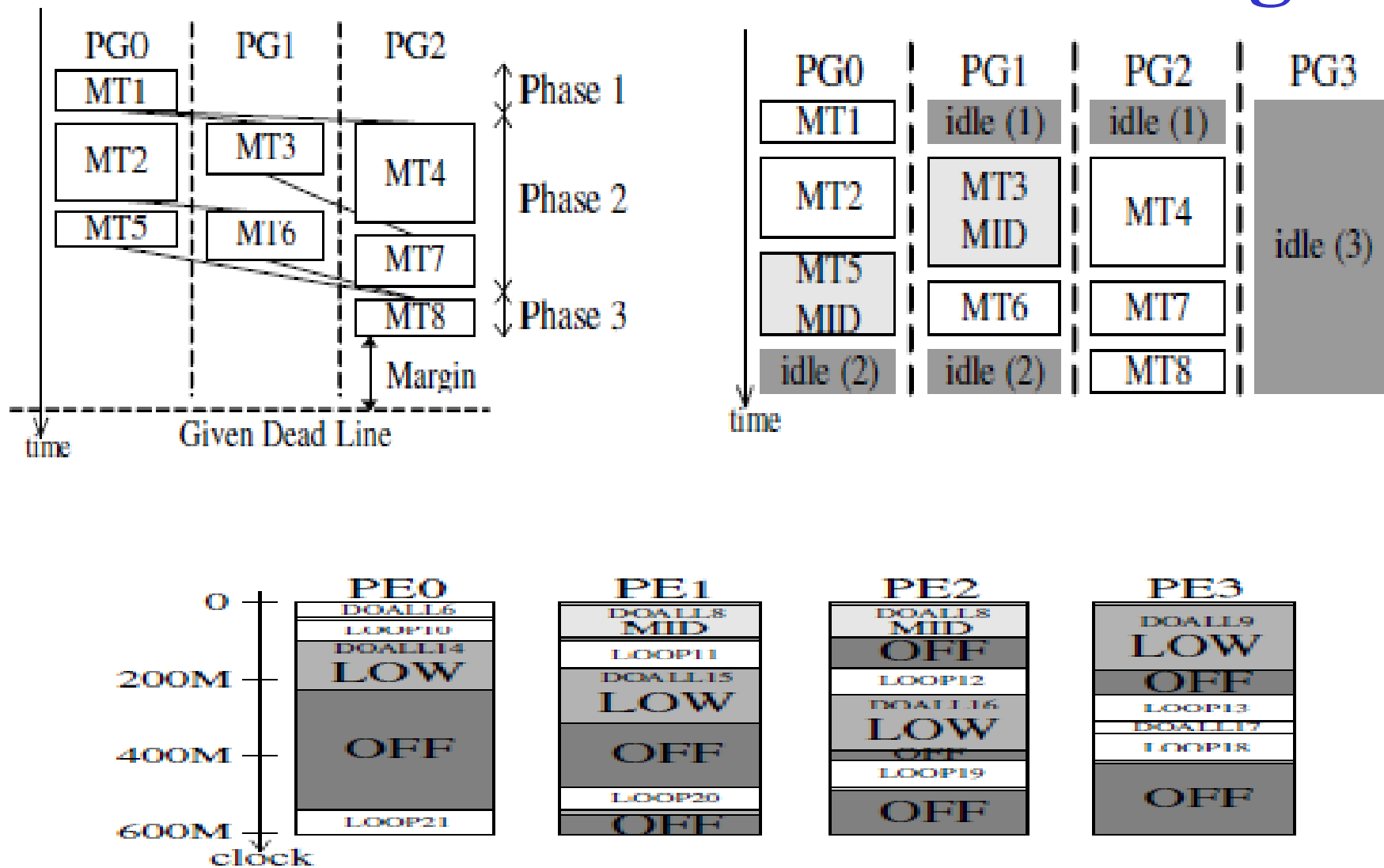


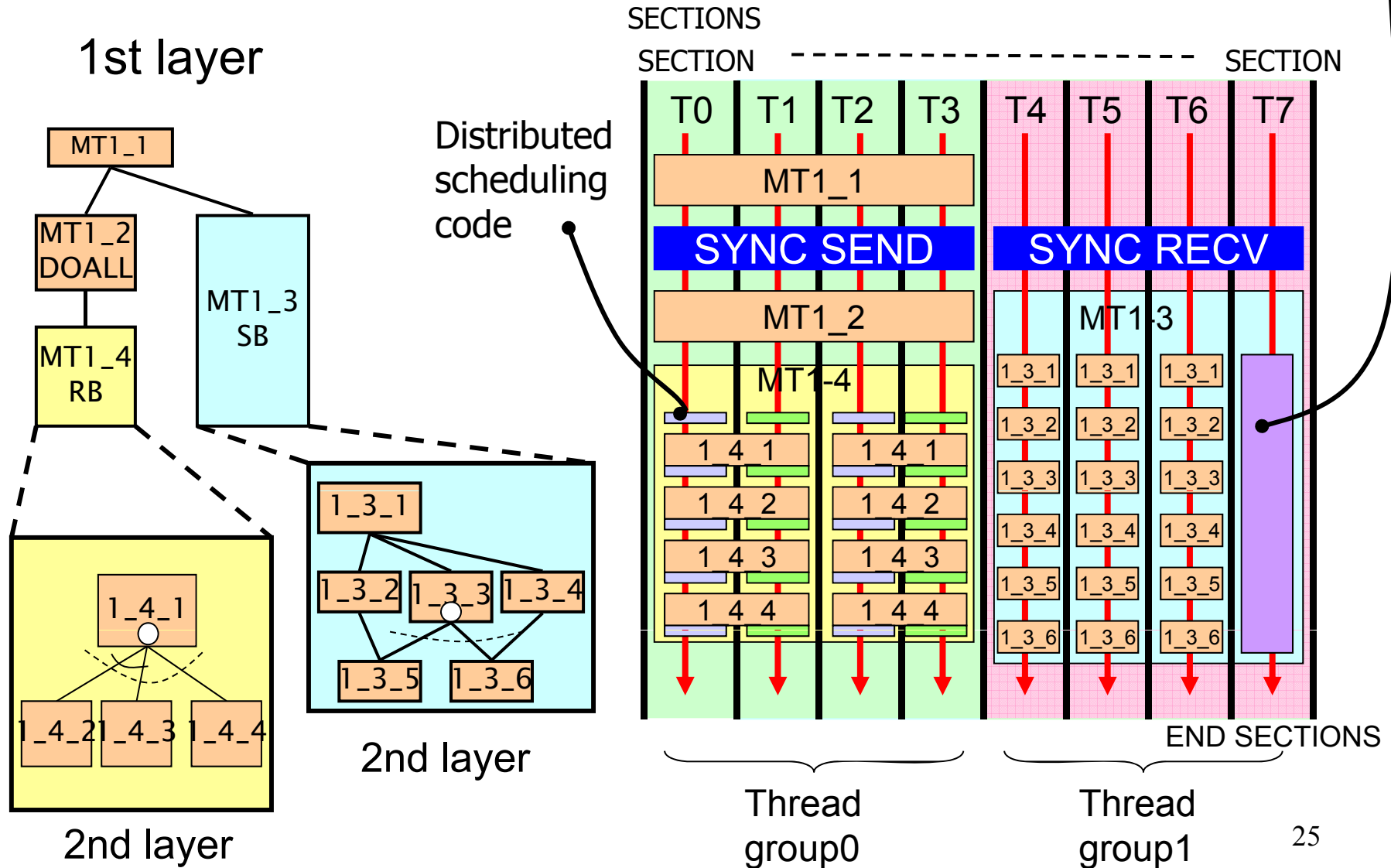
Fig. 6. V/F control of applu(4proc.)



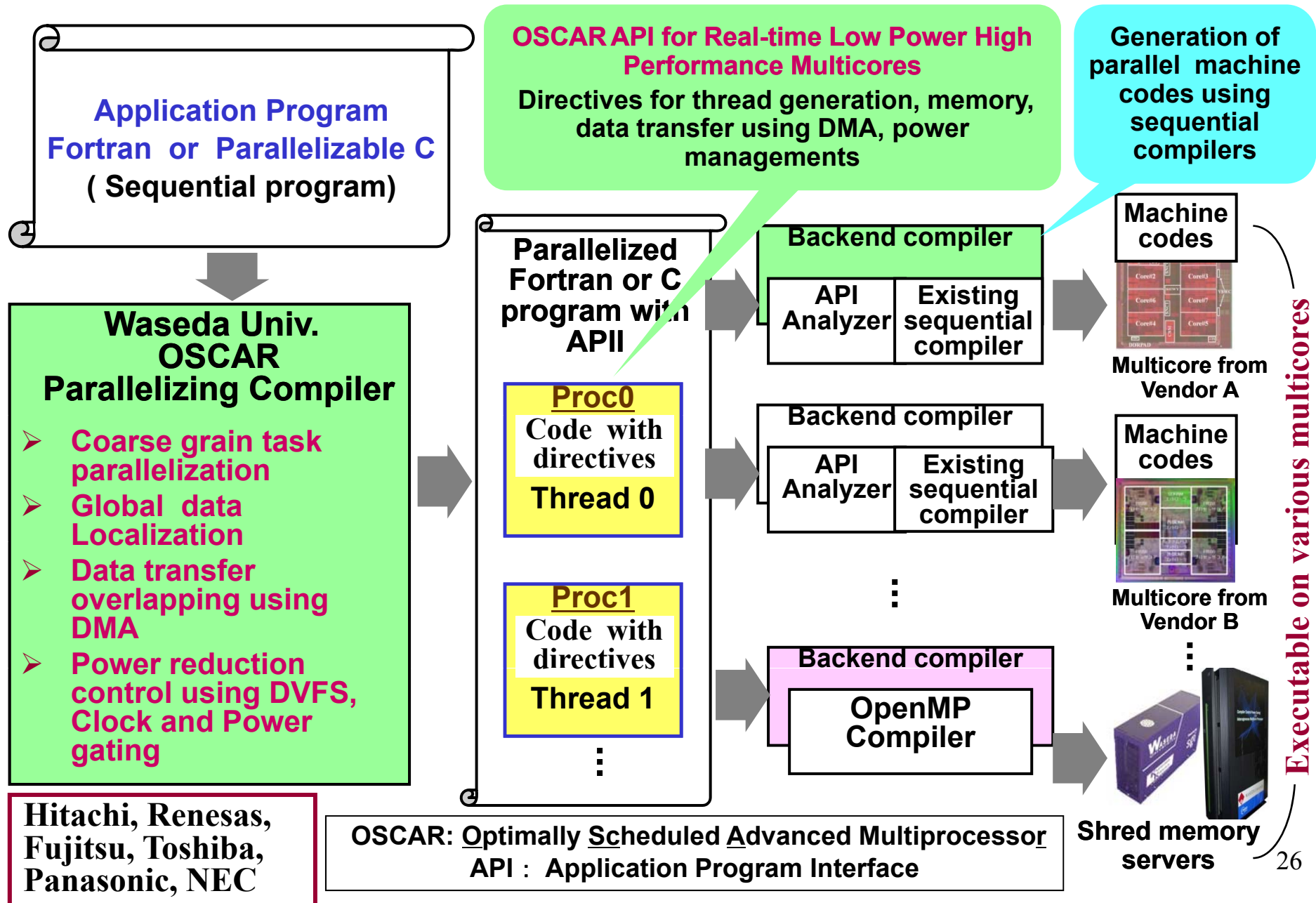
# Generated Multigrain Parallelized Code

(The nested coarse grain task parallelization is realized by only OpenMP “section”, “Flush” and “Critical” directives.)

Centralized scheduling code



# Compilation Flow Using OSCAR API

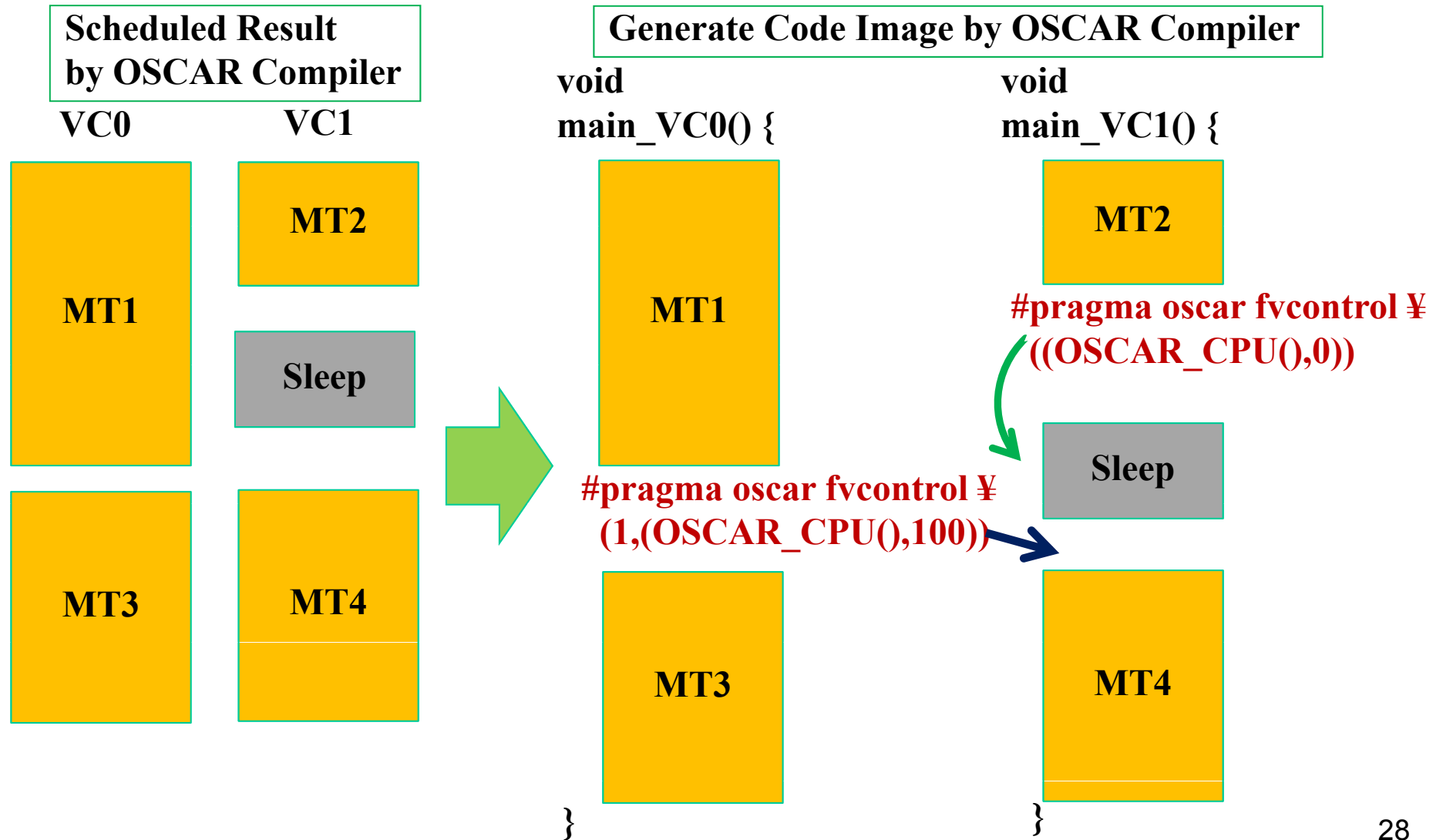


# OSCAR API

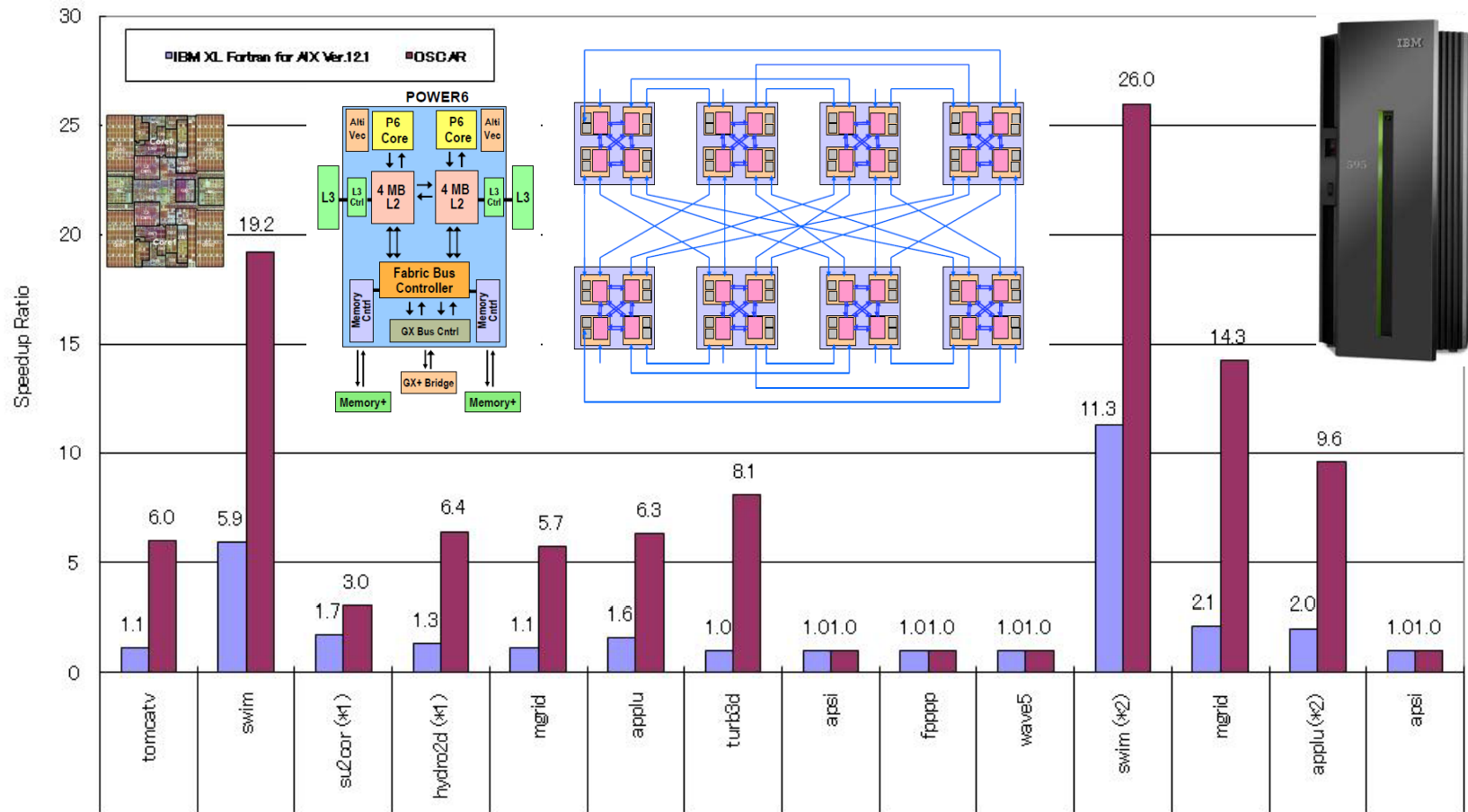
**Now Open! (<http://www.kasahara.cs.waseda.ac.jp/>)**

- **Targeting mainly realtime consumer electronics devices**
  - embedded computing
  - various kinds of memory architecture
    - SMP, local memory, distributed shared memory, ...
- **Developed with Japanese 6 companies**
  - Fujitsu, Hitachi, NEC, Toshiba, Panasonic, Renesas
  - Supported by METI/NEDO
- **Based on the subset of OpenMP**
  - very popular parallel processing API
  - shared memory programming model
- **Six Categories**
  - **Parallel Execution** (4 directives from OpenMP)
  - **Memory Mapping** (Distributed Shared Memory, Local Memory)
  - **Data Transfer Overlapping** Using DMA Controller
  - **Power Control** (DVFS, Clock Gating, Power Gating)
  - **Timer for Real Time Control**
  - **Synchronization** (Hierarchical Barrier Synchronization)

# Low-Power Optimization with OSCAR API



# Performance of OSCAR Compiler on IBM p6 595 Power6 (4.2GHz) based 32-core SMP Server



**OpenMP codes generated by OSCAR compiler accelerate IBM XL Fortran for AIX Ver.12.1 about **3.3 times** on the average**

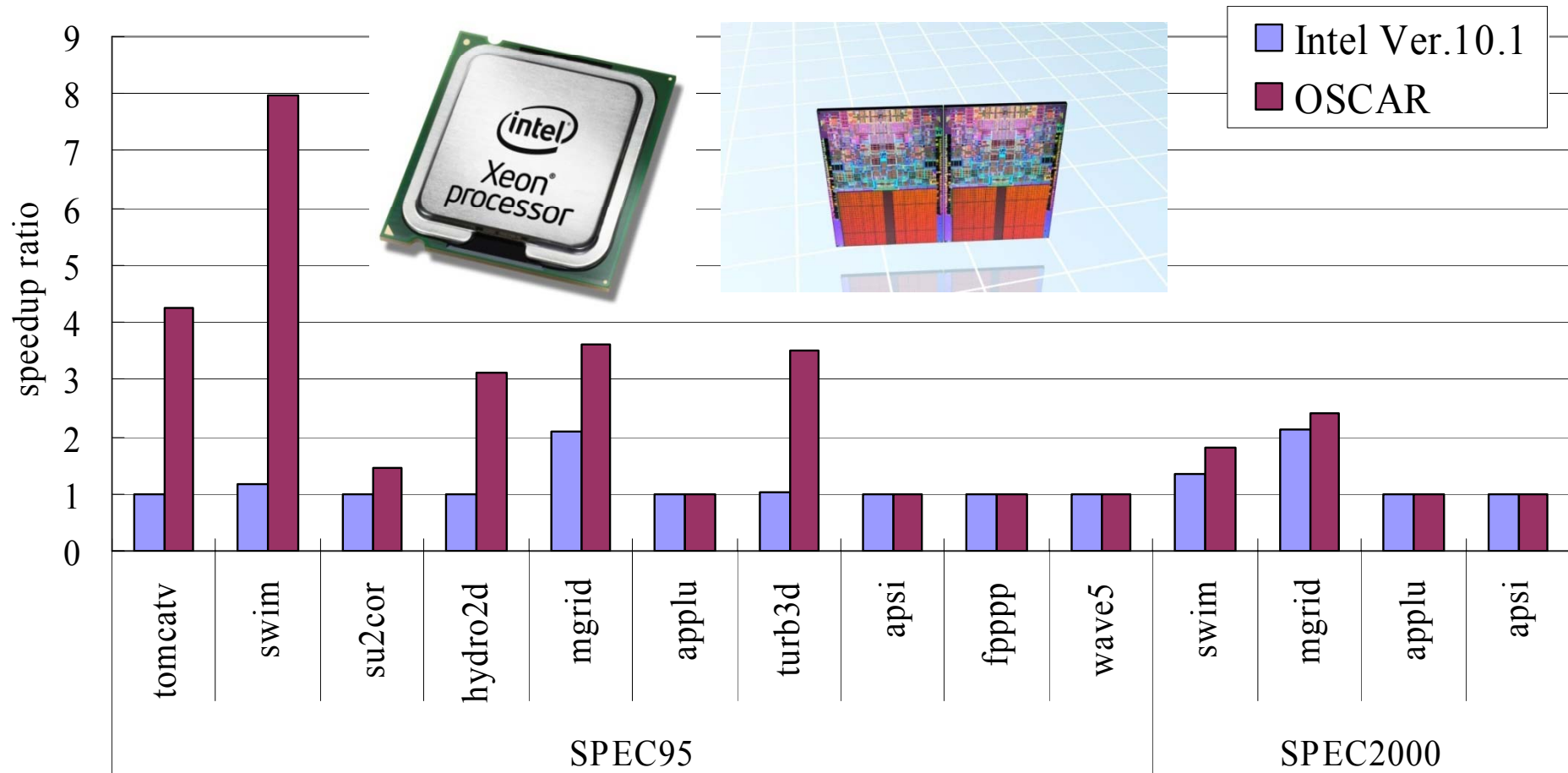
Compile Option:

(\*1) Sequential: -O3 -qarch=pwr6, XLF: -O3 -qarch=pwr6 -qsmp=auto, OSCAR: -O3 -qarch=pwr6 -qsmp=noauto

(\*2) Sequential: -O5 -q64 -qarch=pwr6, XLF: -O5 -q64 -qarch=pwr6 -qsmp=auto, OSCAR: -O5 -q64 -qarch=pwr6 -qsmp=noauto

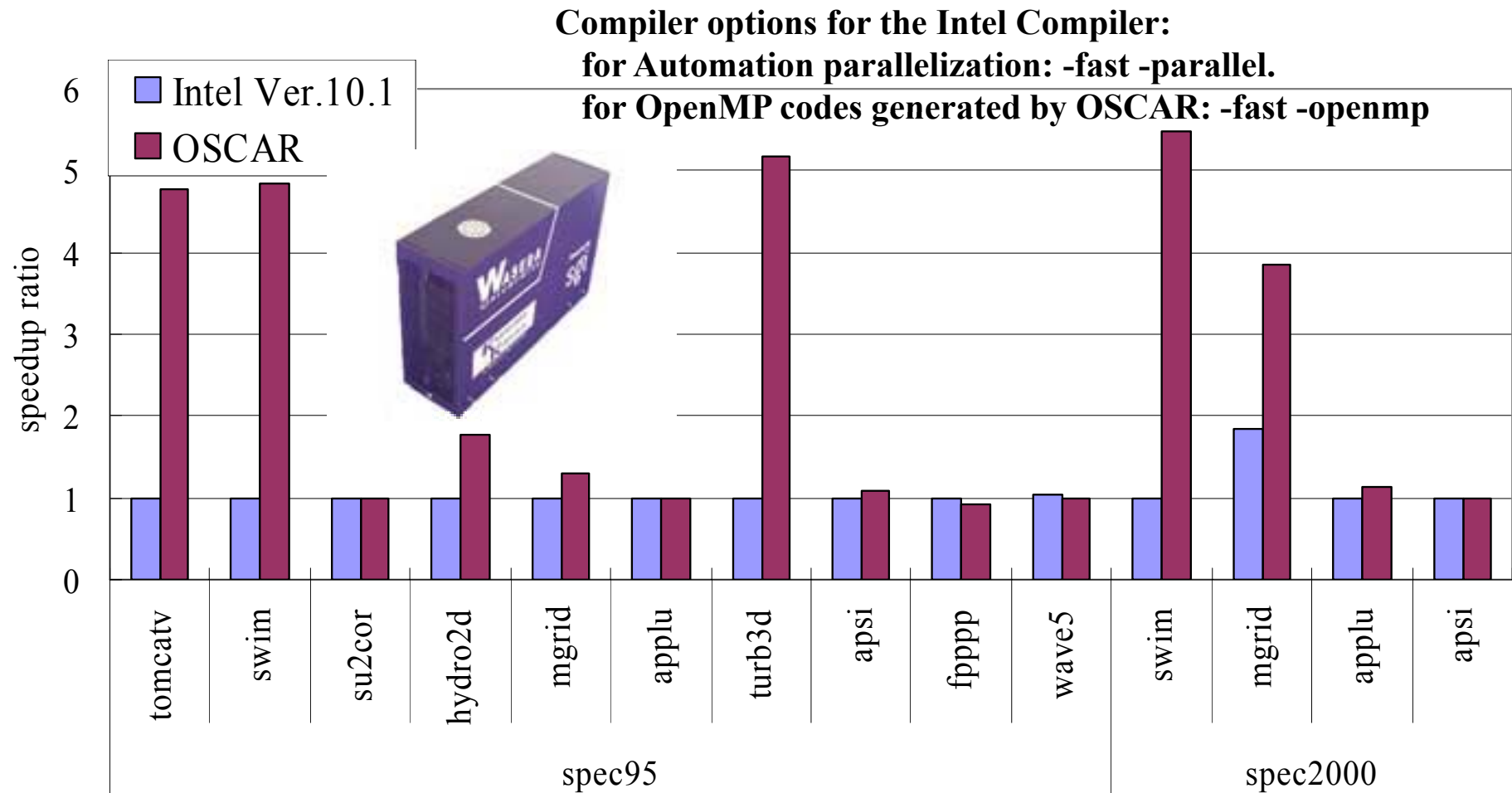
(Others) Sequential: -O5 -qarch=pwr6, XLF: -O5 -qarch=pwr6 -qsmp=auto, OSCAR: -O5 -qarch=pwr6 -qsmp=noauto

# Performance of OSCAR Compiler Using the Multicore API on Intel Quad-core Xeon



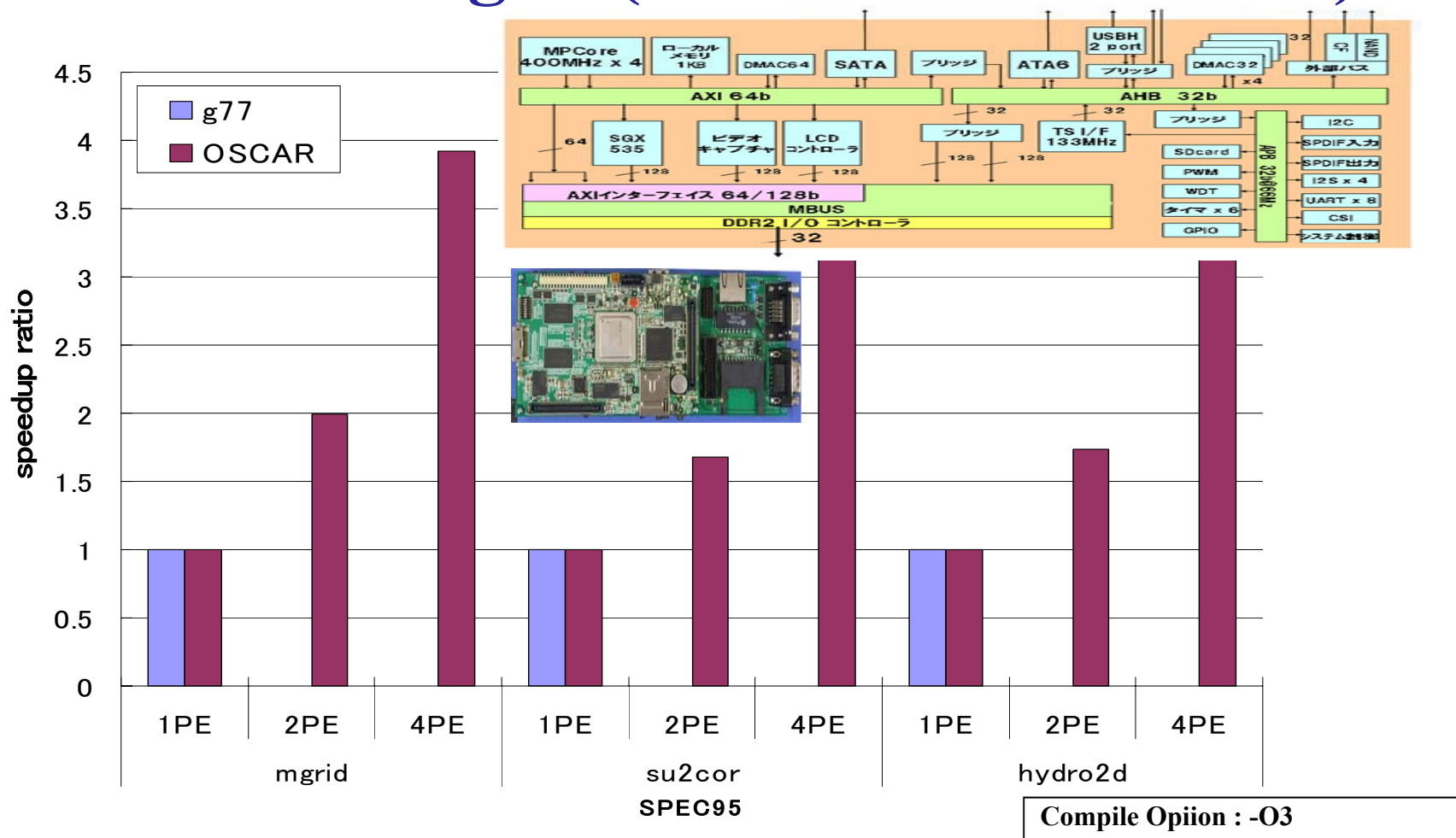
- **OSCAR Compiler gives us 2.1 times speedup on the average against Intel Compiler ver.10.1**

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



- **OSCAR compiler gave us 2.3 times speedup against Intel Fortran Itanium Compiler revision 10.1**

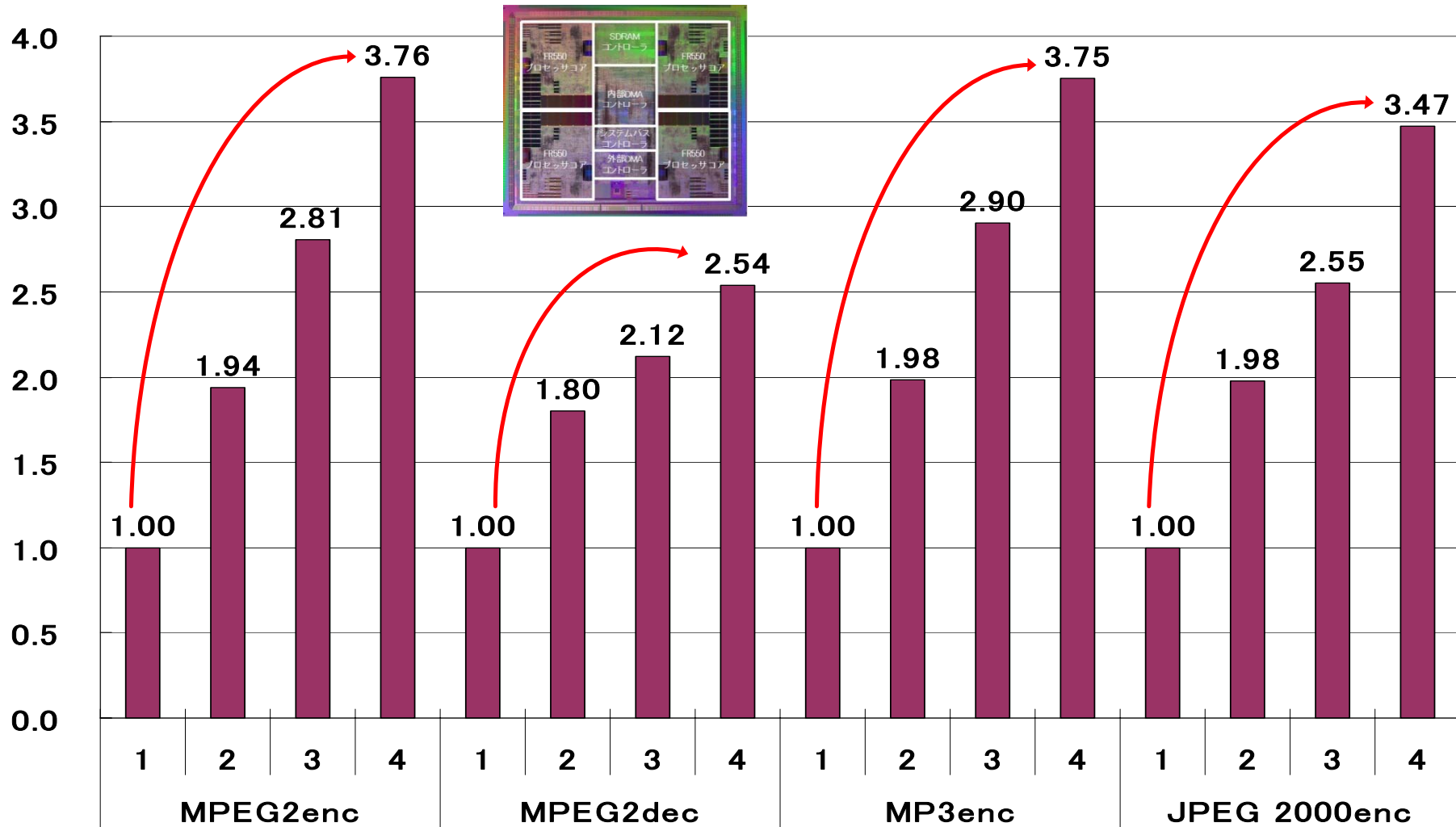
# Performance of OSCAR compiler on NEC NaviEngine(ARM-NEC MPcore)



- OSCAR compiler gave us 3.43 times speedup against 1 core on ARM/NEC MPCore with 4 ARM 400MHz cores

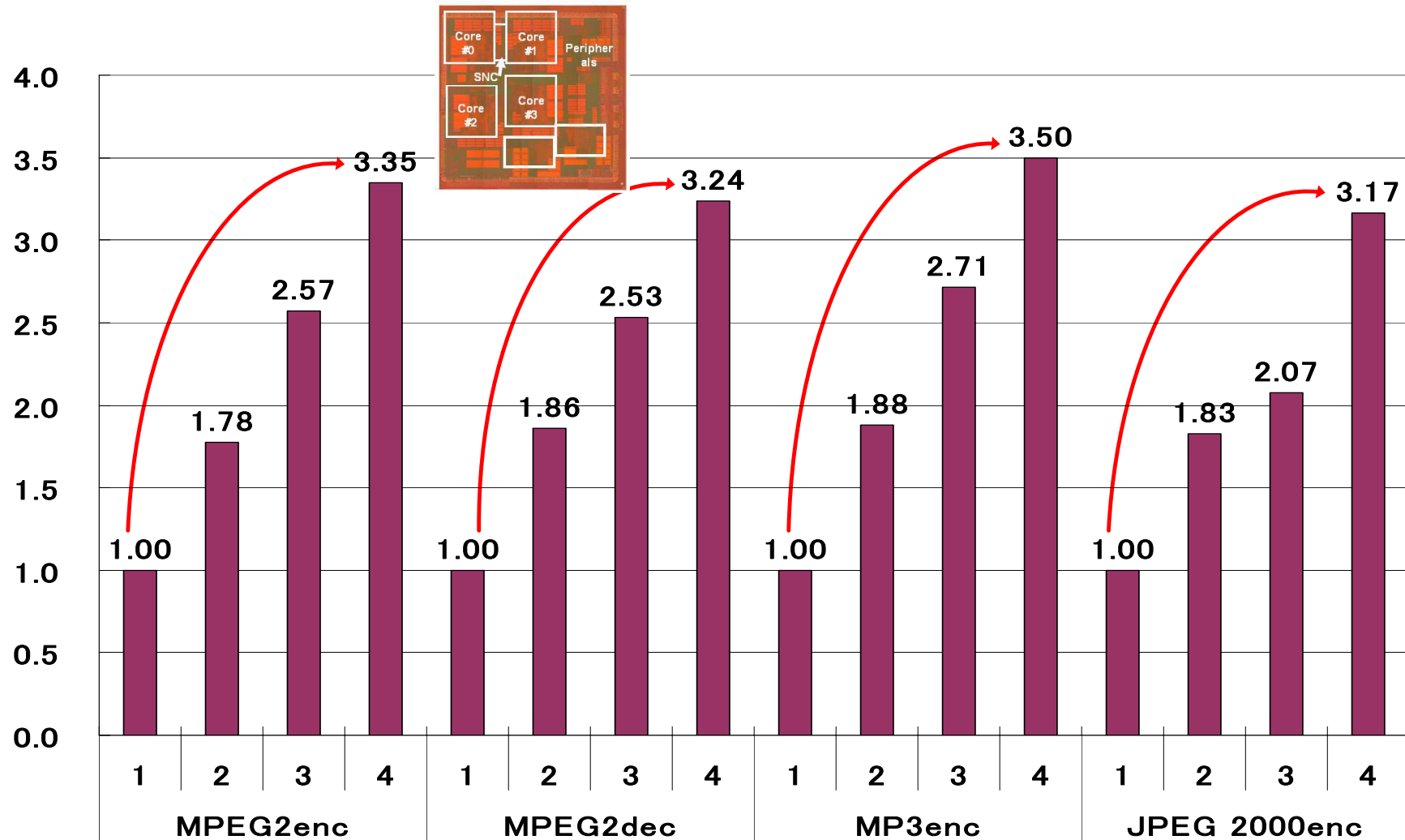


# Performance of OSCAR Compiler Using the multicore API on Fujitsu FR1000 Multicore



**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

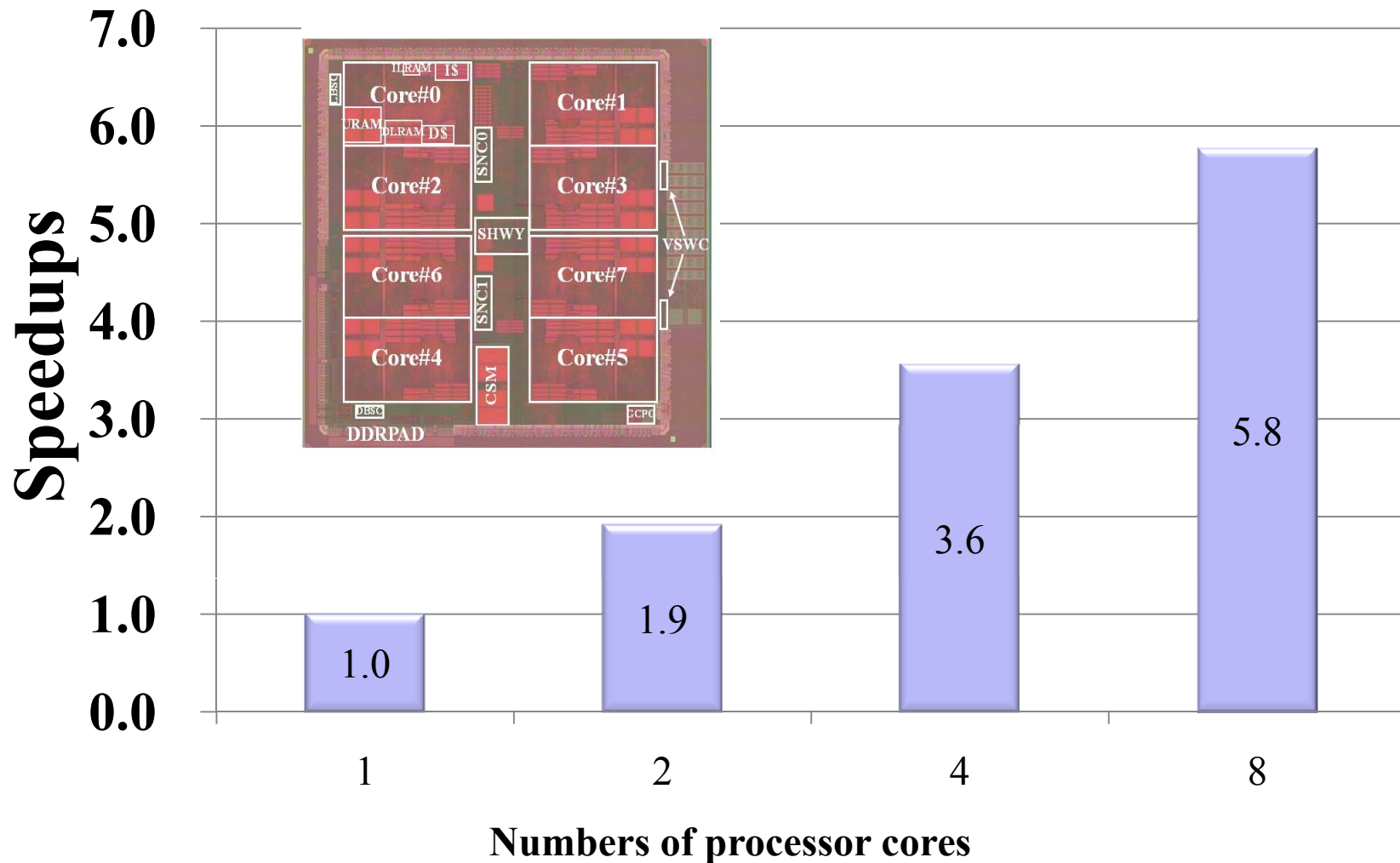


**3.31 times speedup on the average for 4cores against 1core**

# Processing Performance on the Developed Multicore Using Automatic Parallelizing Compiler

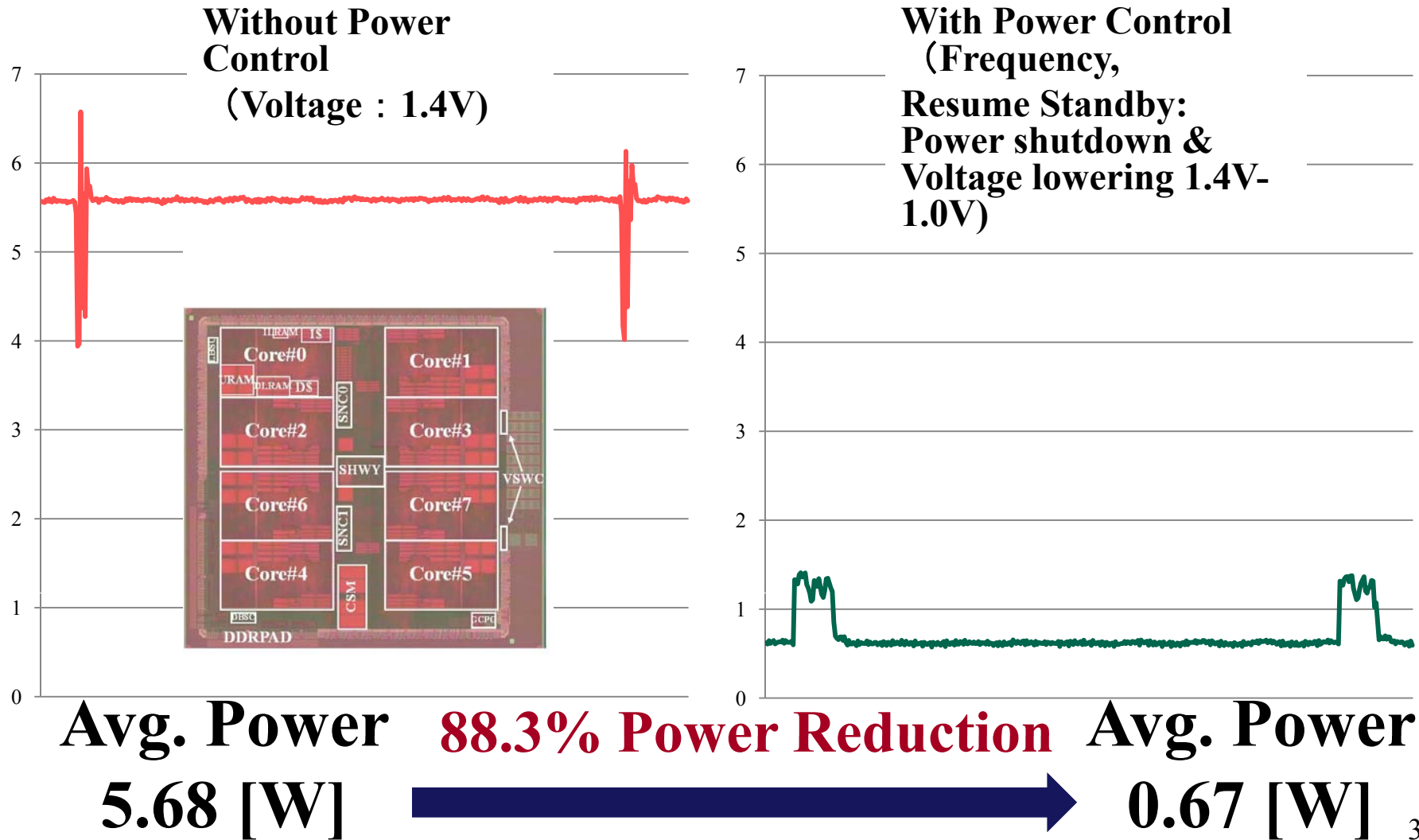
Speedup against single core execution for audio AAC encoding

\*) Advanced Audio Coding



# Power Reduction by OSCAR Parallelizing Compiler for Secure Audio Encoding

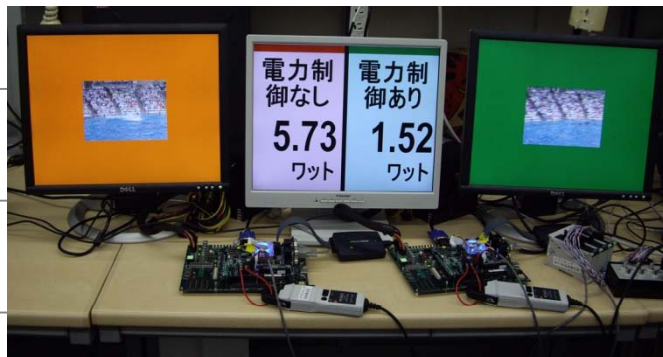
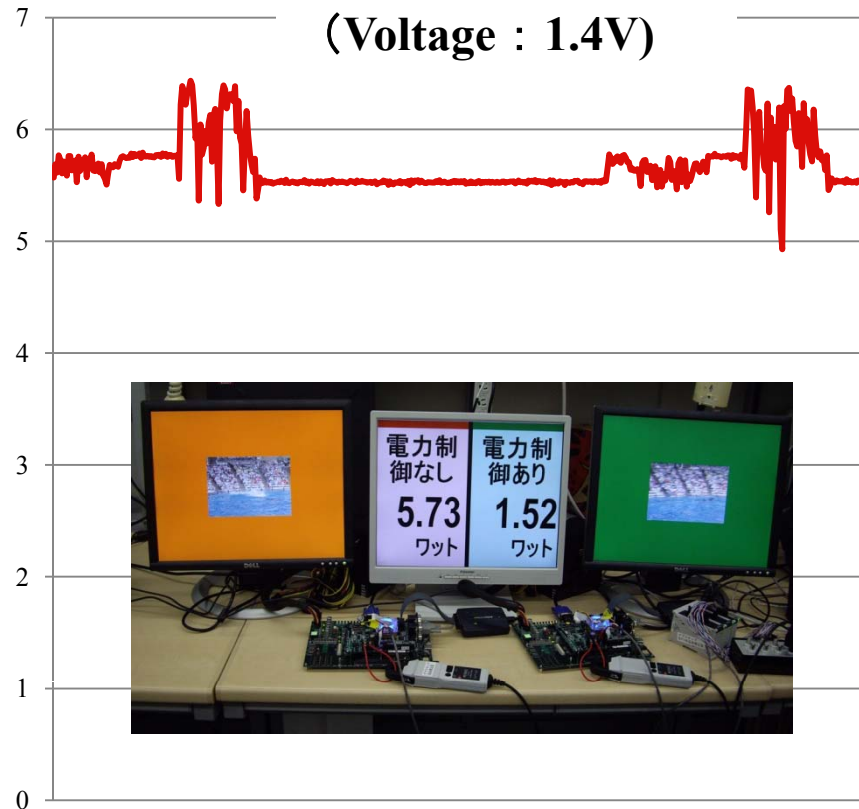
AAC Encoding + AES Encryption with 8 CPU cores



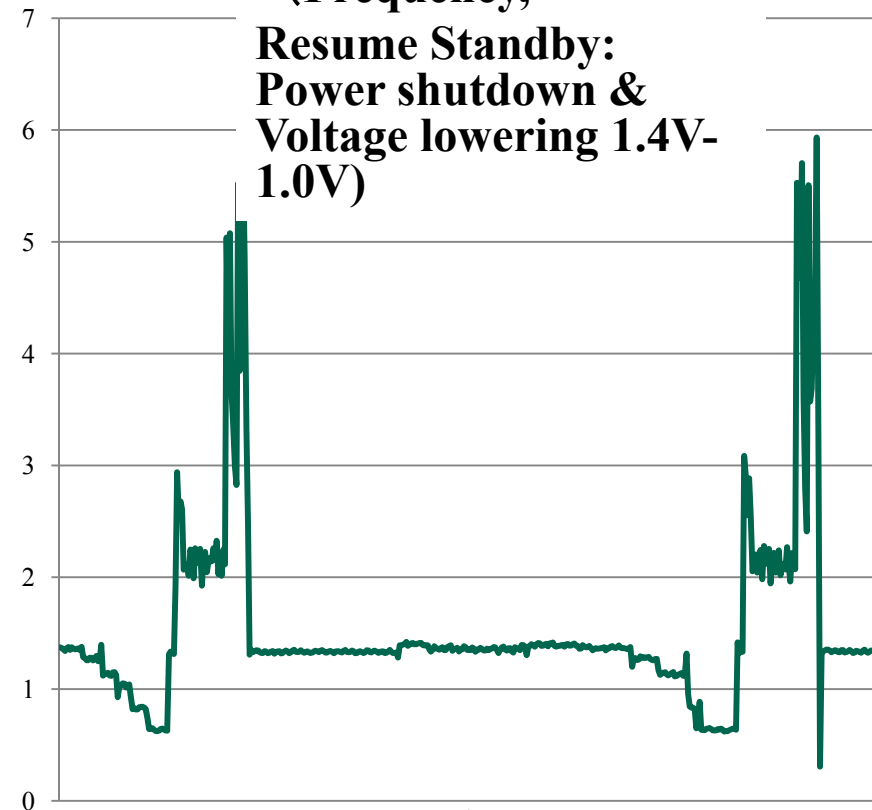
# Power Reduction by OSCAR Parallelizing Compiler for MPEG2 Decoding

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]

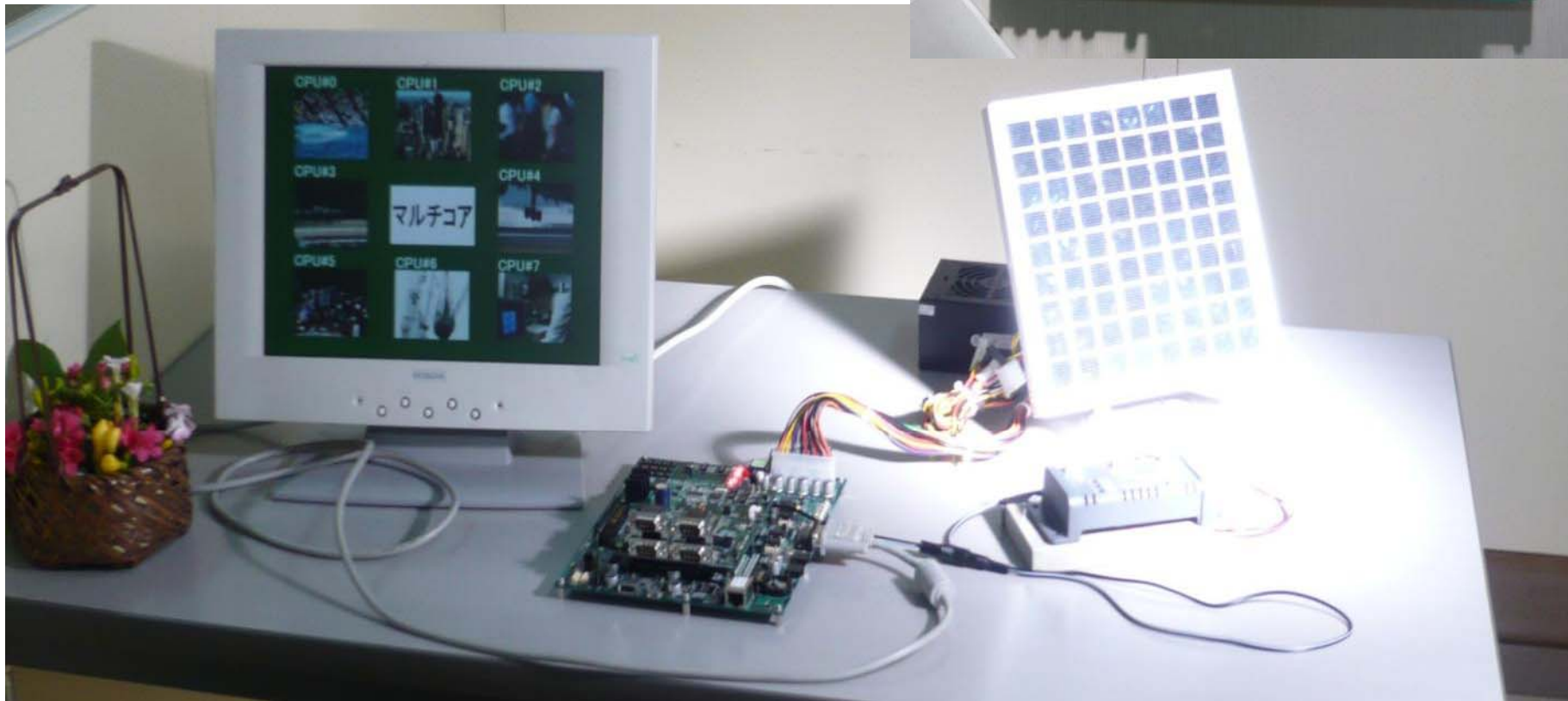
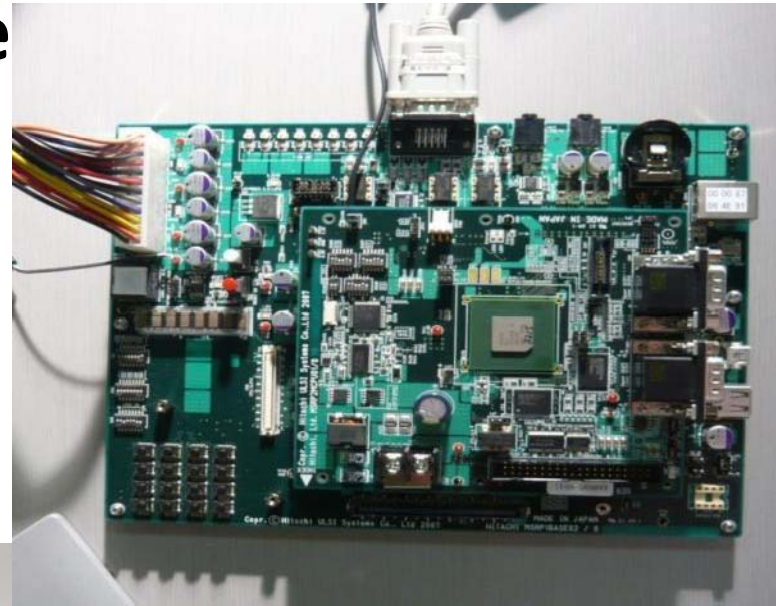
73.5% Power Reduction



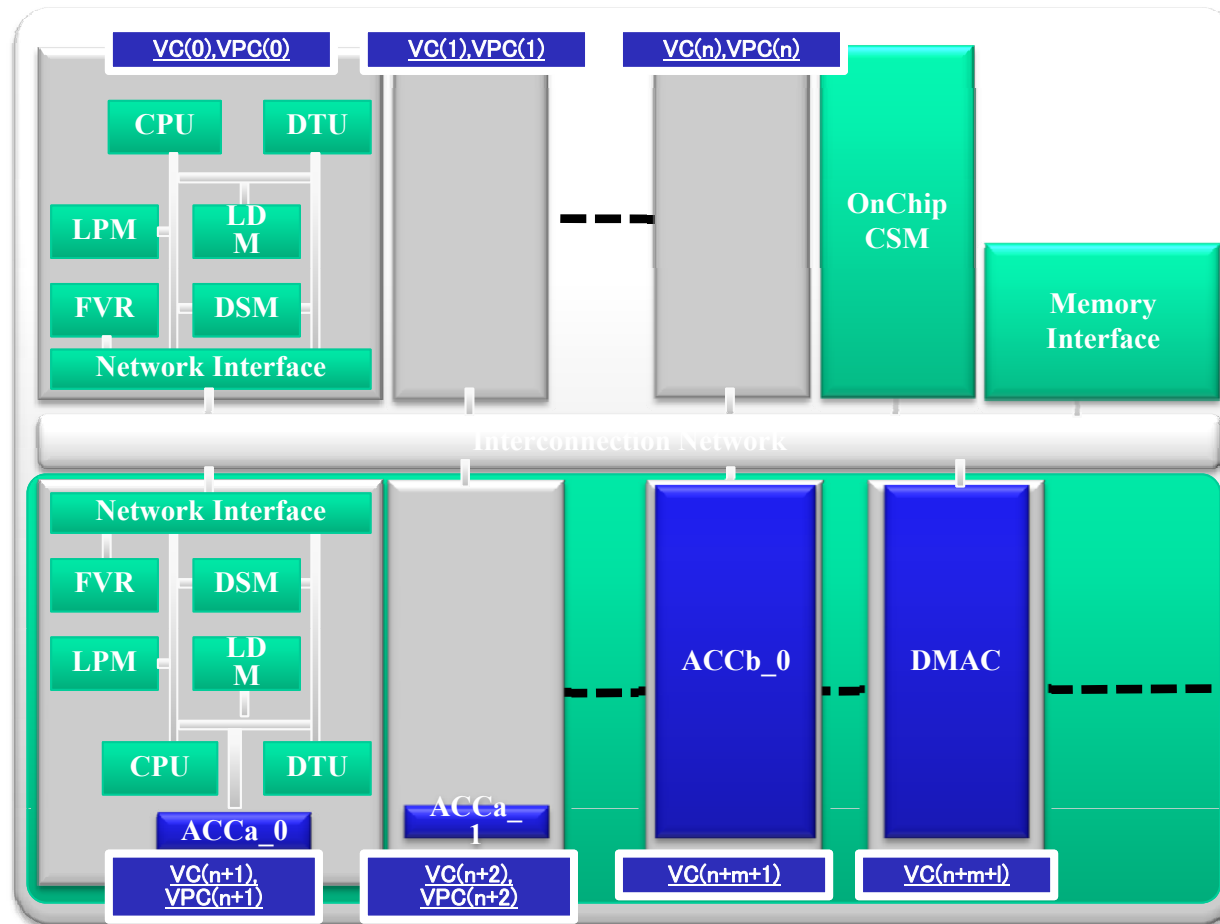
Avg. Power  
1.52 [W]

# Low Power High Performance Multicore Computer with Solar Panel

- Clean Energy Autonomous
- Servers operational in deserts

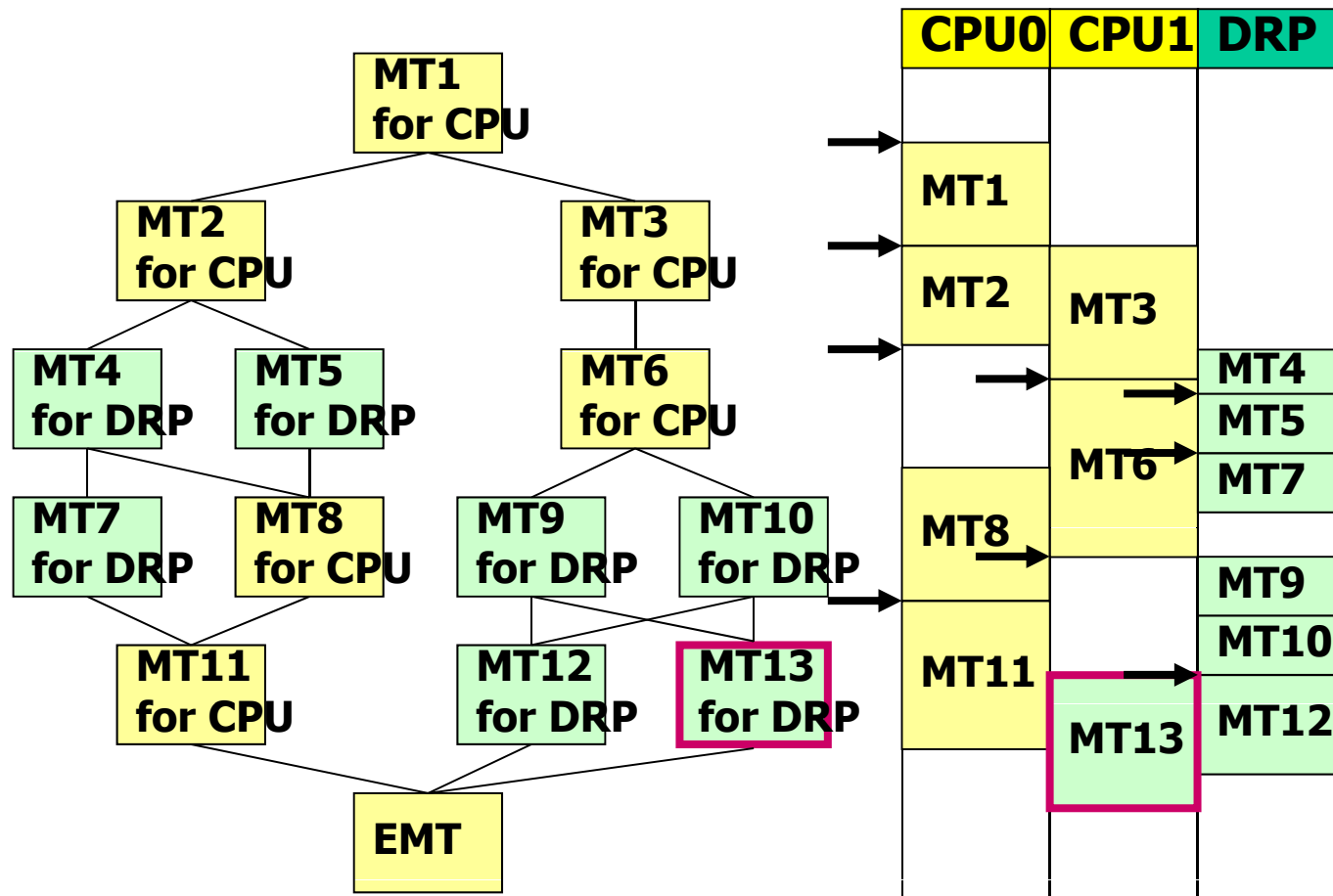


# OSCAR API-Applicable Heterogeneous Multicore Architecture



- DTU
  - Data Transfer Unit
- LPM
  - Local Program Memory
- LDM
  - Local Data Memory
- DSM
  - Distributed Shared Memory
- CSM
  - Centralized Shared Memory
- FVR
  - Frequency/Voltage Control Register

# Static Scheduling of Coarse Grain Tasks for a Heterogeneous Multi-core

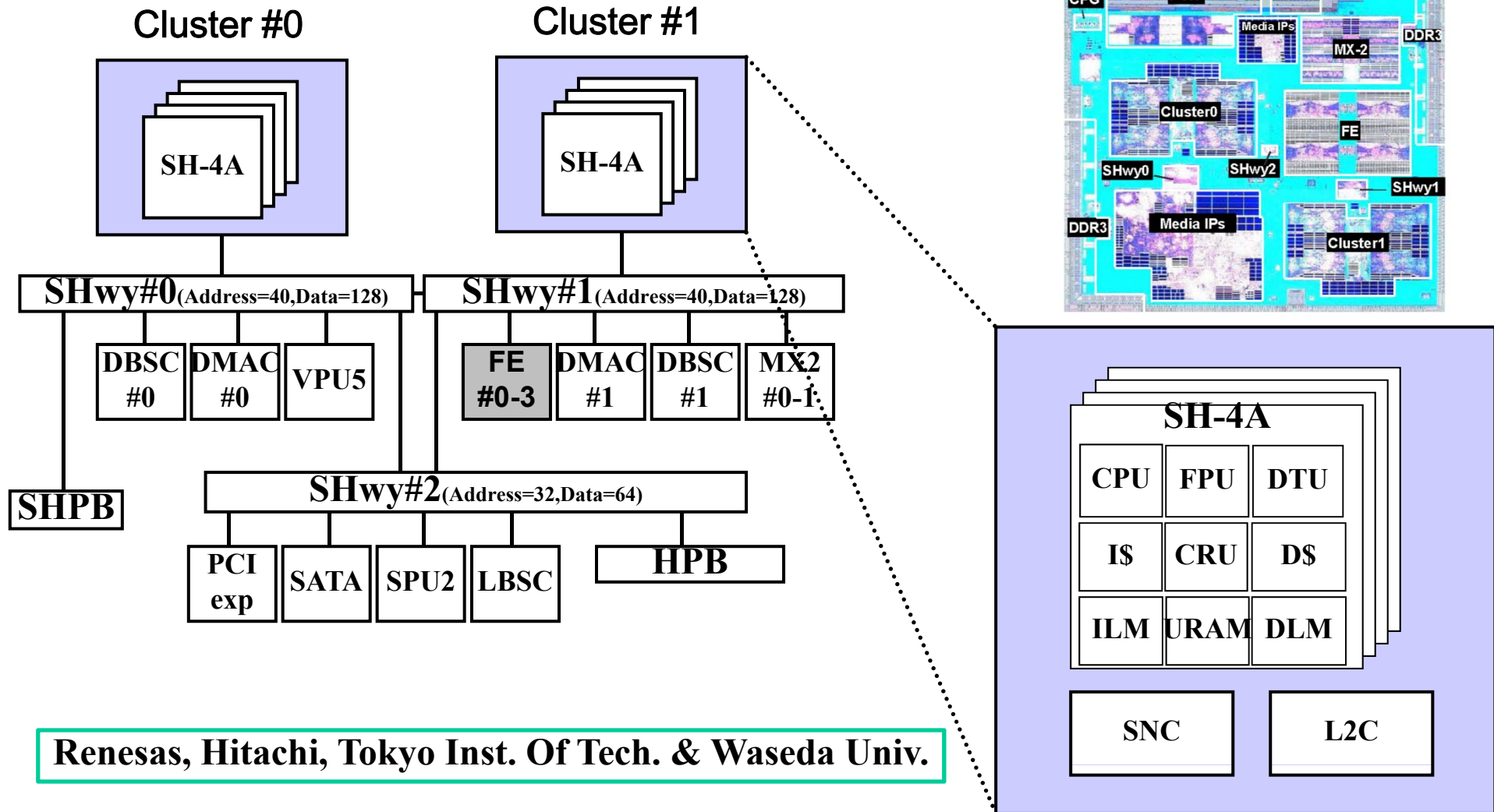






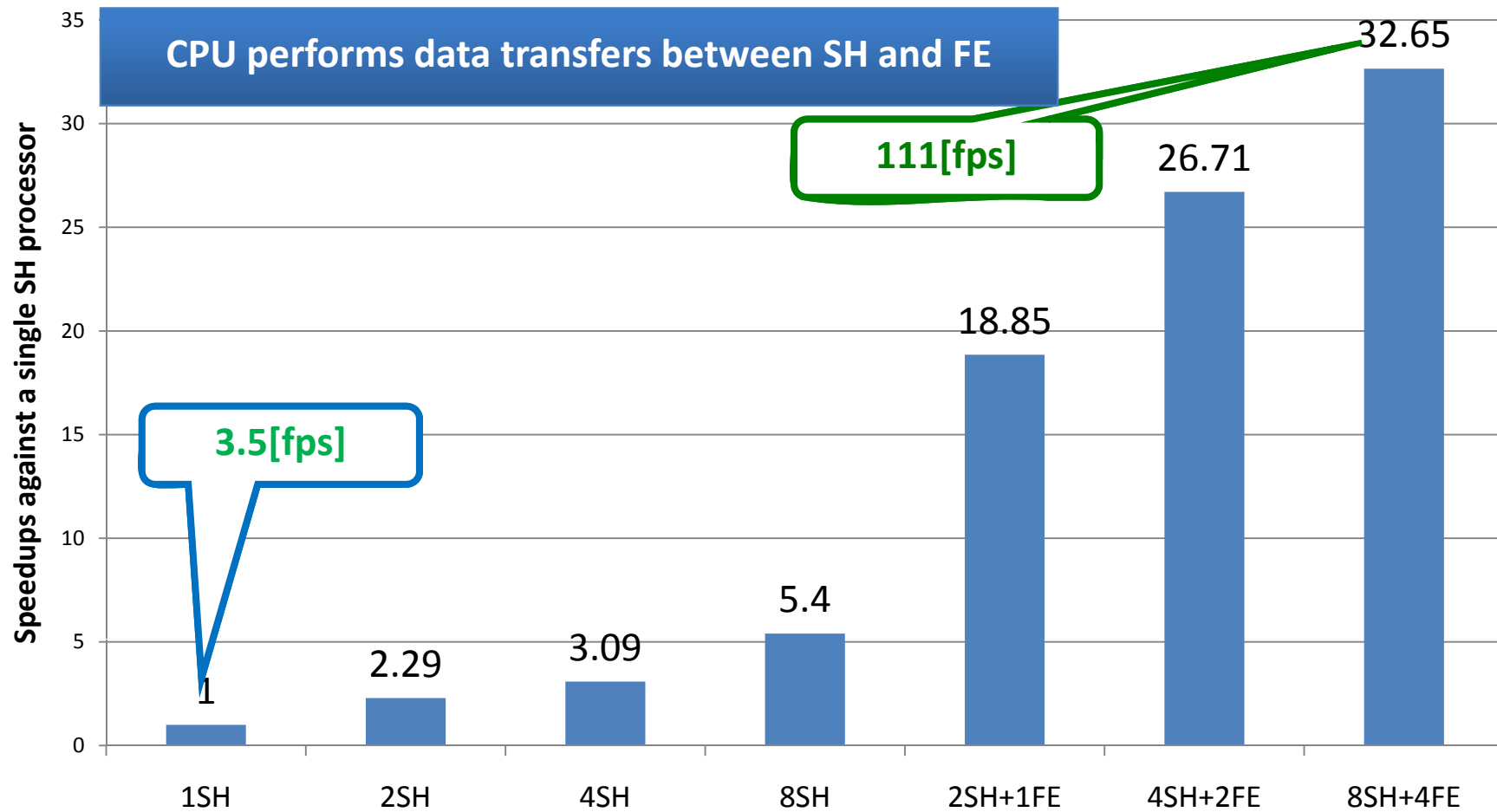
# Heterogeneous Multicore RP-X

presented in SSCC2010 Processors Session on Feb. 8, 2010



Renesas, Hitachi, Tokyo Inst. Of Tech. & Waseda Univ.

# Parallel Processing Performance Using OSCAR Compiler and OSCAR API on RP-X(Optical Flow with a hand-tuned library)

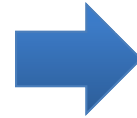


# Power Reduction in a real-time execution controlled by OSCAR Compiler and OSCAR API on RP-X (Optical Flow with a hand-tuned library)

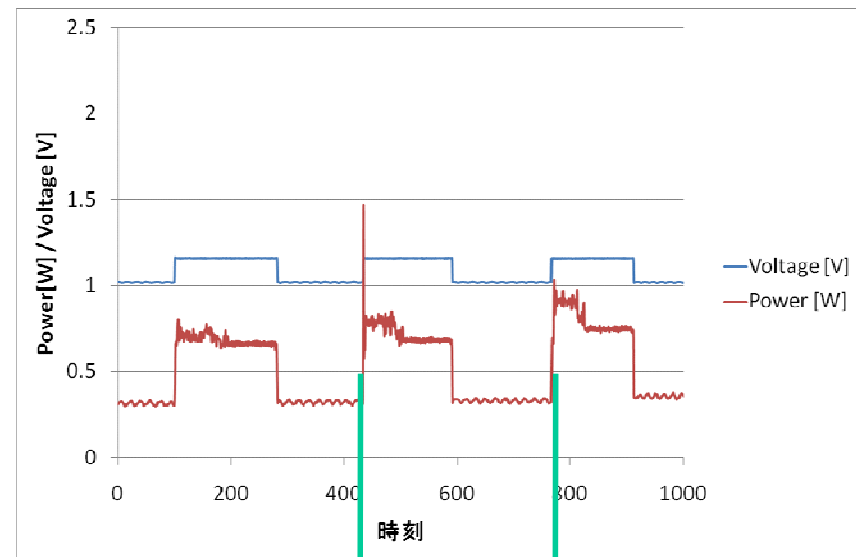
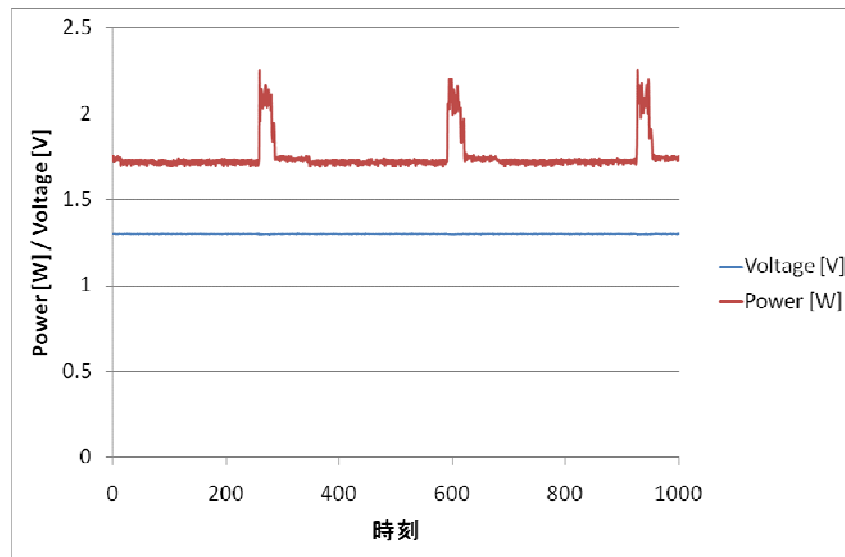
**Without Power Reduction**

**With Power Reduction by OSCAR Compiler**  
**70% of power reduction**

**Average: 1.76[W]**

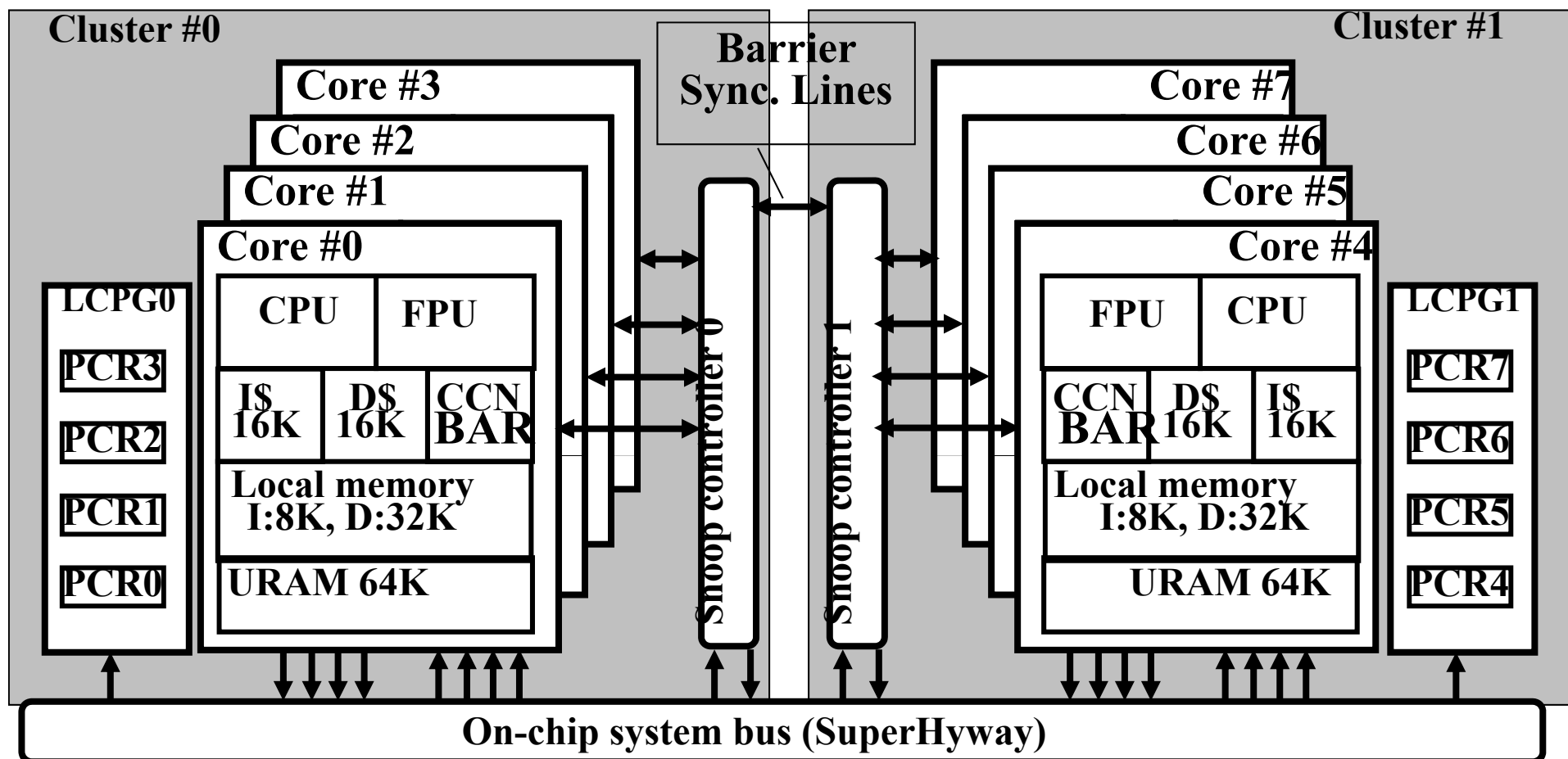


**Average: 0.54[W]**



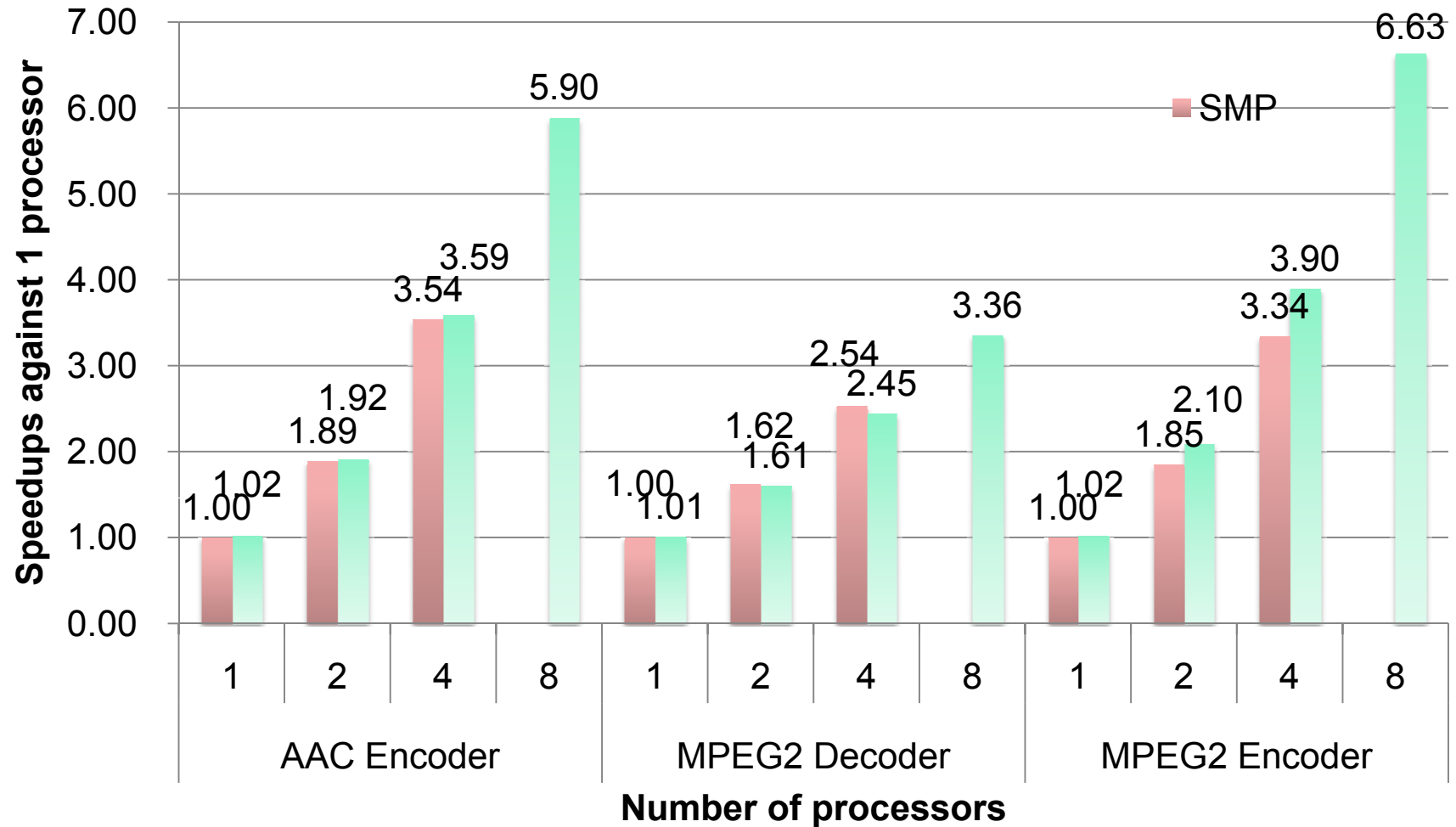
**1cycle : 33[ms]  
→30[fps]**

# 8 Core RP2 Chip Block Diagram



LCPG: Local clock pulse generator  
 PCR: Power Control Register  
 CCN/BAR: Cache controller/Barrier Register  
 URAM: User RAM (Distributed Shared Memory)

# Software Cache Coherent Control by OSCAR Compiler and API on RP2

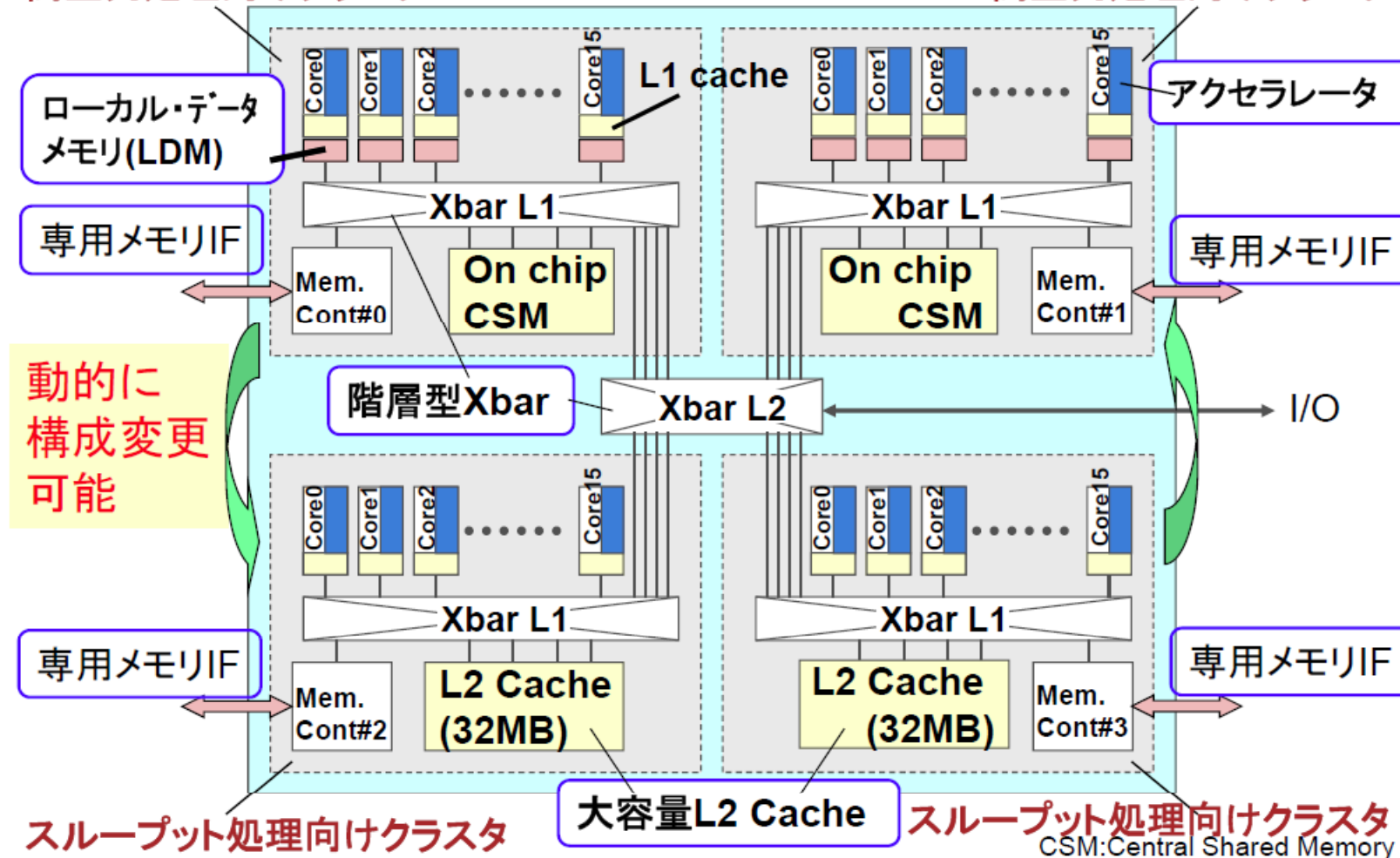


# Fujitsu's Vector Manycore Design Evaluated in NEDO Manycore Leading Research, Feb. 2010

各アプリ領域の処理を効率良く実行するための方式 **FUJITSU**

高並列処理向けクラスター

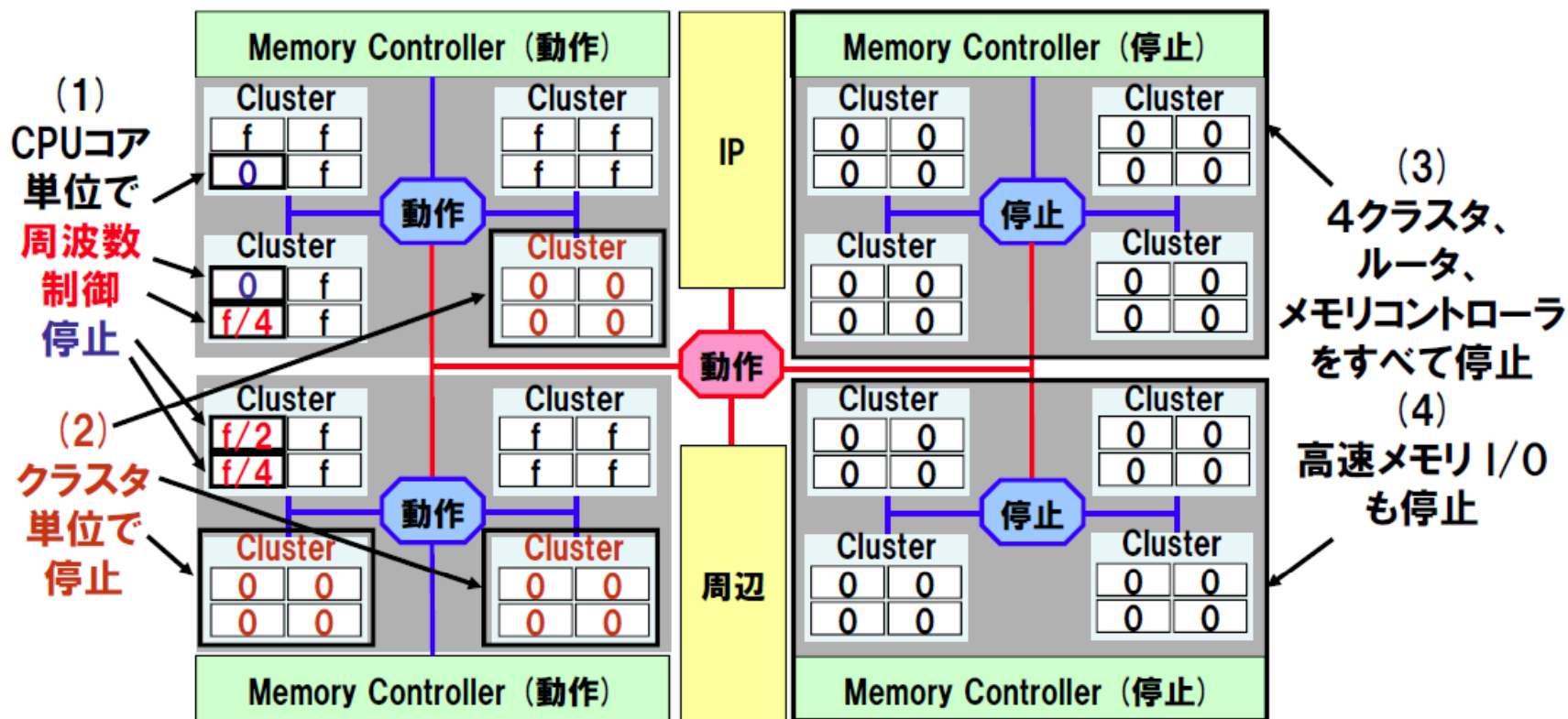
高並列処理向けクラスター



# Renesas's Low Power Manycore Design Evaluated in NEDO Manycore Leading Research, Feb. 2010

## 4.3 低電力マルチコア実装技術 (1)

◆ 提案した64コア、Fat-treeアーキテクチャにて、以下のような処理負荷に連動した階層的な低電力制御を行うことで、メニーコアチップ低消費電力実現



Fat-Tree構成メニーコア低電力制御例



# 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

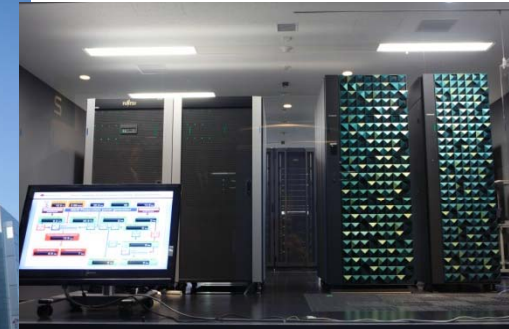
- More than 64 cores
- Natural air cooling (No fan)  
Cool, Compact, Clear, Quiet
- Operational by Solar Panel

## <Industry, Government, Academia>

Hitachi, Fujitsu, NEC, Renesas, Olympus,  
Toyota, Denso, Mitsubishi, Toshiba, etc

## <Ripple Effect>

- Low CO<sub>2</sub> (Carbon Dioxide) Emissions
- Creation Value Added Products
  - Consumer Electronics, Automobiles,  
Servers



Hitachi SR16000:

Power7 128coreSMP

Fujitsu M9000

SPARC VII 256 core SMP



Beside Subway Waseda Station,  
Near Waseda Univ. Main Campus

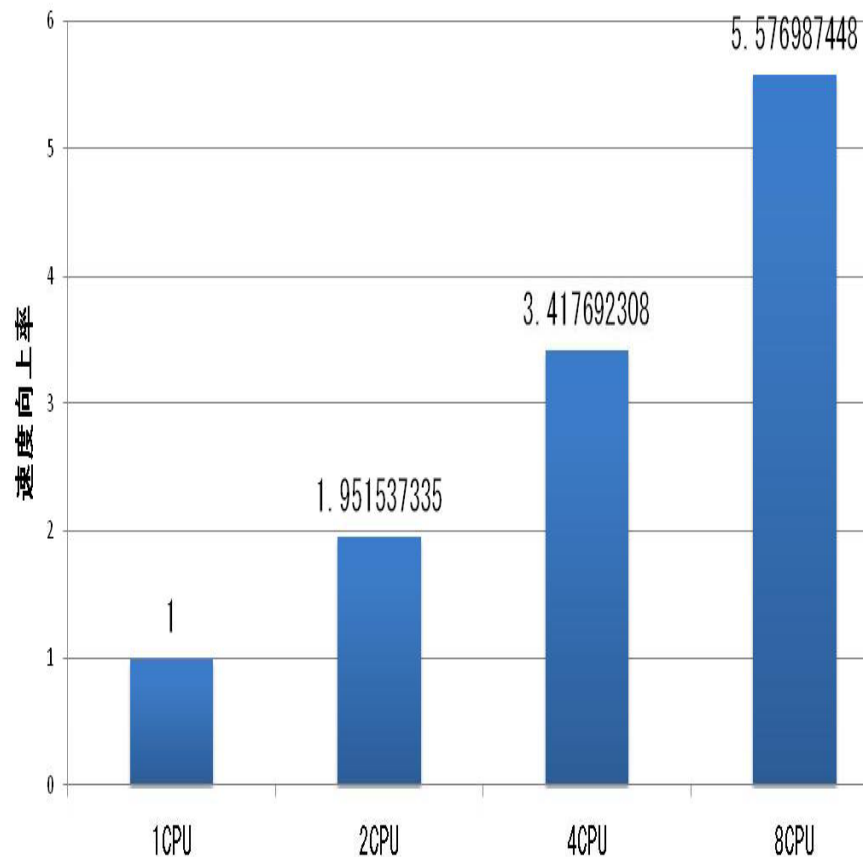
Research, development and practical utilization through industry-government-academia partnerships (spillover effect)

## Environment



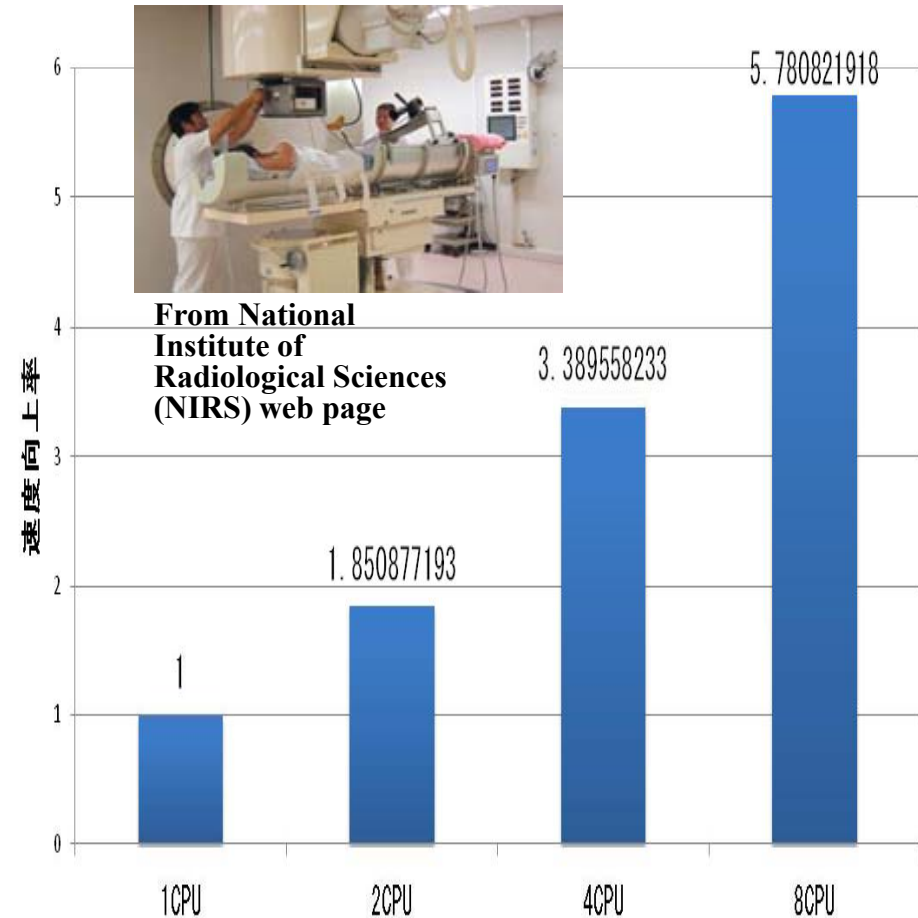
# Cancer Treatment: Carbon Ion Radiotherapy

(This research was performed with cooperation of  
Mitsubishi Electric and Mitsubishi Space Software.)



5.58 times speedup by 8 processors

**Intel Quadcore Xeon 8 core SMP**



5.78 times speedup by 8 processors

**IBM Power 7 8 core SMP  
(Hitachi SR16000)**

# Conclusions

- **OSCAR compiler cooperative real-time low power multicore with high effective performance, short software development period will be important in wide range of IT systems from consumer electronics to automobiles, medical systems, disaster super-realtime simulator (Tsunami), and EX-FLOPS machines.**
- **For industry**
  - A few minutes of compilation of C program using OSCAR Compiler and API without months of programming allows us**
    - **Several times speedups on market available SMP servers.**
    - **Scalable speedup on various multicores like 8 core homogeneous RP2 (8SH4A) , 15 core heterogeneous RPX (8SH4A, 4FEGA, 2MX2 & 1VPU), MPCore, FR1000 and so on.**
    - **70% power reduction on RP2 and RPX for realtime media processing .**
- **OSCAR green compiler, API, multicores and manycores will be continuously developed for saving lives from natural disasters and sickness like cancer in addition to the current activities.**