Multicore and Compiler Codesign for Performance and Power

Hironori Kasahara, Ph.D., IEEE Life Fellow

IEEE Computer Society President 2018

Professor, Dept. of Computer Science & Engineering

Senior Executive Vice President (2018-2022)

Waseda University, Tokyo, Japan

Member: Eng. Academy of Japan (Director 2020-24), Science Council of Japan

1980 BS, 82 MS, 85 Ph.D. , Dept. EE, Waseda Univ.
1985 Visiting Scholar: U. of California, Berkeley
1986 Assistant Prof., 1988 Associate Prof., 1997, Waseda Univ., Now Dept. of Computer Sci. & Eng.
1989-90 Research Scholar: U. of Illinois, Urbana-Champaign, Center for Supercomputing R&D
2004 Director, Advanced Multicore Research Institute, 2017 member: the Engineering Academy of Japan and the Science Council of Japan

1987 IFAC World Congress Young Author Prize
1997 IPSJ Sakai Special Research Award
2008 Intel AsiaAcademic Forum Best Research Award
2010 IEEE CS Golden Core Member Award
2014 Minister of Edu., Sci. & Tech. Research Prize
2015 IPSJ Fellow, 2017 IEEE Fellow, Eta Kappa Nu
2019 IEEE CS Spirit of Computer Society Award
IPSJ Contribution Award, 2023 IEEE Life Fellow

Reviewed Papers: 237, Invited Talks: 252,
Granted International Patents: 72, Articles in News Papers, Web News, Medias incl. TV etc.: 718

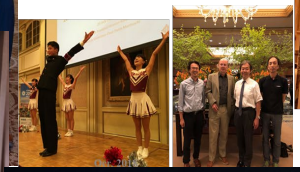
Committees in Societies and Government: 304
IEEE Computer Society President 2018, BoG, Multicore STC Chair, IEEE F. Allen Medal Chair
【METI/NEDO】 Project Leaders: Multicore for Consumer Electronics, Advanced Parallelizing Compiler, Chair: Computer Strategy Committee
【Cabinet Office】 CSTP Supercomputer Strategic ICT PT, Japan Prize Selection Committees, etc.
【MEXT】Info. Sci. & Tech. Committee, Supercomputers (Earth Simulator, K) Committees, National Univ. Evaluation Committee,
【JST】PhD Stipend Project Chair, SBIR Phase1 Chair, Venture Support Governing Board
ACM/IEEE ISCA'25 G. Co-Chair, PACT'27 G. Chair

IEEE Computer Society

The first President from outside North America in 72 years history of IEEE CS

Bjarne Stroustrup: Morgan Stanley & Columbia Univ.
2018 IEEE Computer Society Computer Pioneer Award
IEEE COMPSAC2018 Keynote & Award Ceremony

July 26, 2018, Keynote, Hitotsubashi Hall



July 25, 2018 Award Ceremony Rihga Royal Hotel Tokyo

215
International Conferences

12 Magazines

35 Journals

47 Total Publications

847,000+
Articles in CSDL

6
New Standards

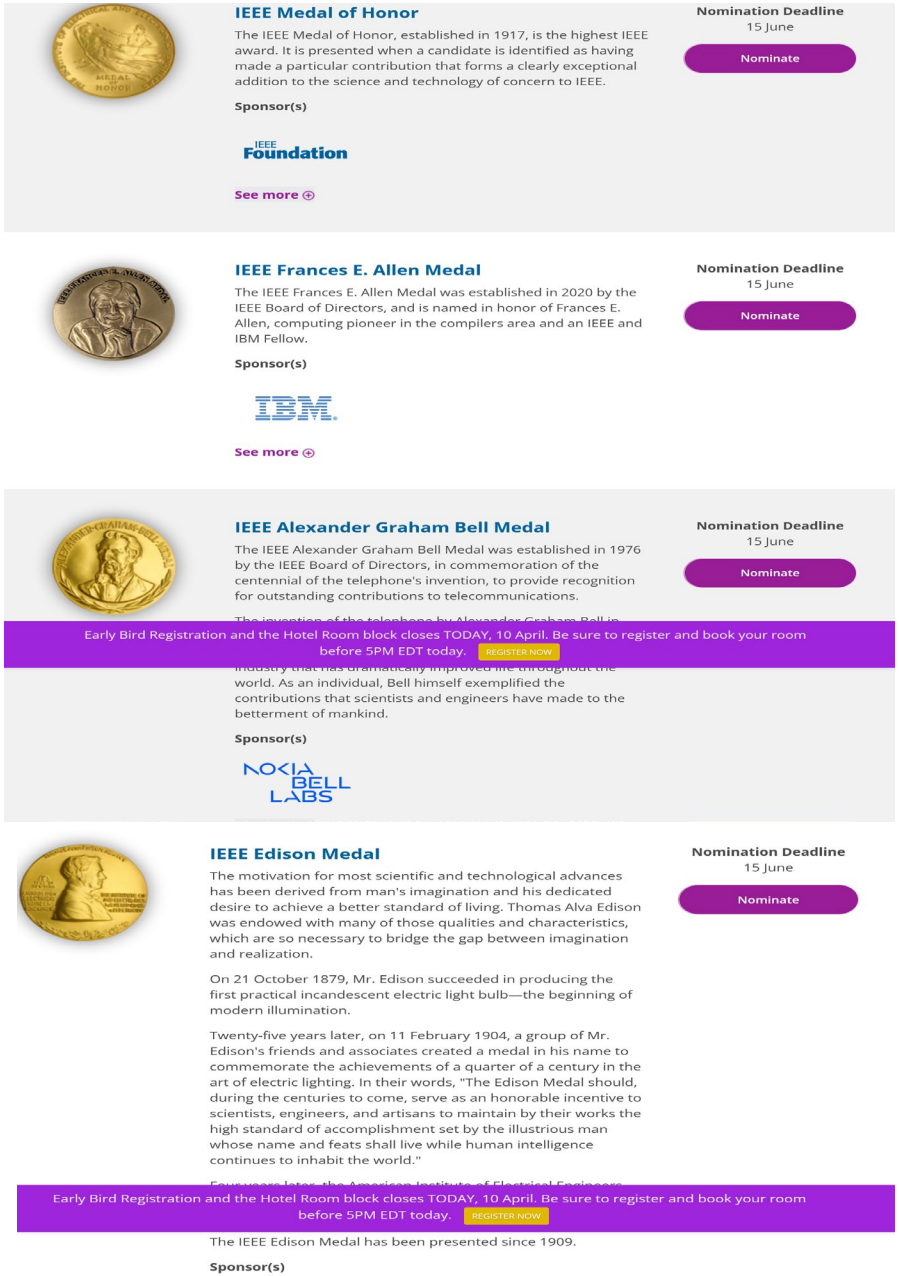
230
Active Standards

IEEE754, 802

IEEE Awards
@IEEEAwards

Congratulations to @IllinoisCS' David J. Kuck on his 2024 @IEEEorg Frances E. Allen Medal, sponsored by @IBM, for pioneering work in vector and #ParallelComputer architecture, s#oftware, and compilers that enables many performance-sensitive applications: bit.ly/IEEEAwards-Rec...

ポストを翻訳



https://corporate-awards.ieee.org/corporate-awards/

Recipients of the IEEE Medal of Honor

Prestigious IEEE Medal of Honor Prize: US $2 Million

IEEE Medal of Honor

NVIDIAFounder

Jensen Huang

"For leadership in the development of graphics processing units and their application to scientific computing and artificial intelligence."

(2025) IEEE Medal of Honor

Broadcom Founder /UCLA/ UCI Donation

Henry Samueli

"For pioneering research and commercialization of broadband communication and networking technologies, and promotion of STEM education."

(2024) IEEE Medal of Honor

Father of Internet

Robert E. Kahn

"For pioneering technical and leadership contributions in packet communication technologies and foundations of the Internet."

(2012) Medal of Honor

RISC MIPS, Tensilica Ex-Stanford President, Chairman of Alphabet (Google)

John L. Hennessy

"For pioneering the RISC processor architecture and for leadership in computer engineering and higher education."

(2011) Medal of Honor

TSMC Founder

Morris Chang

"For outstanding leadership in the semiconductor industry."

(2008) Medal of Honor

Semiconductor Moore's Law

Gordon E. Moore

"For pioneering technical roles in integrated-circuit processing, and leadership in the development of MOS memory, the microprocessor computer and the semiconductor industry."

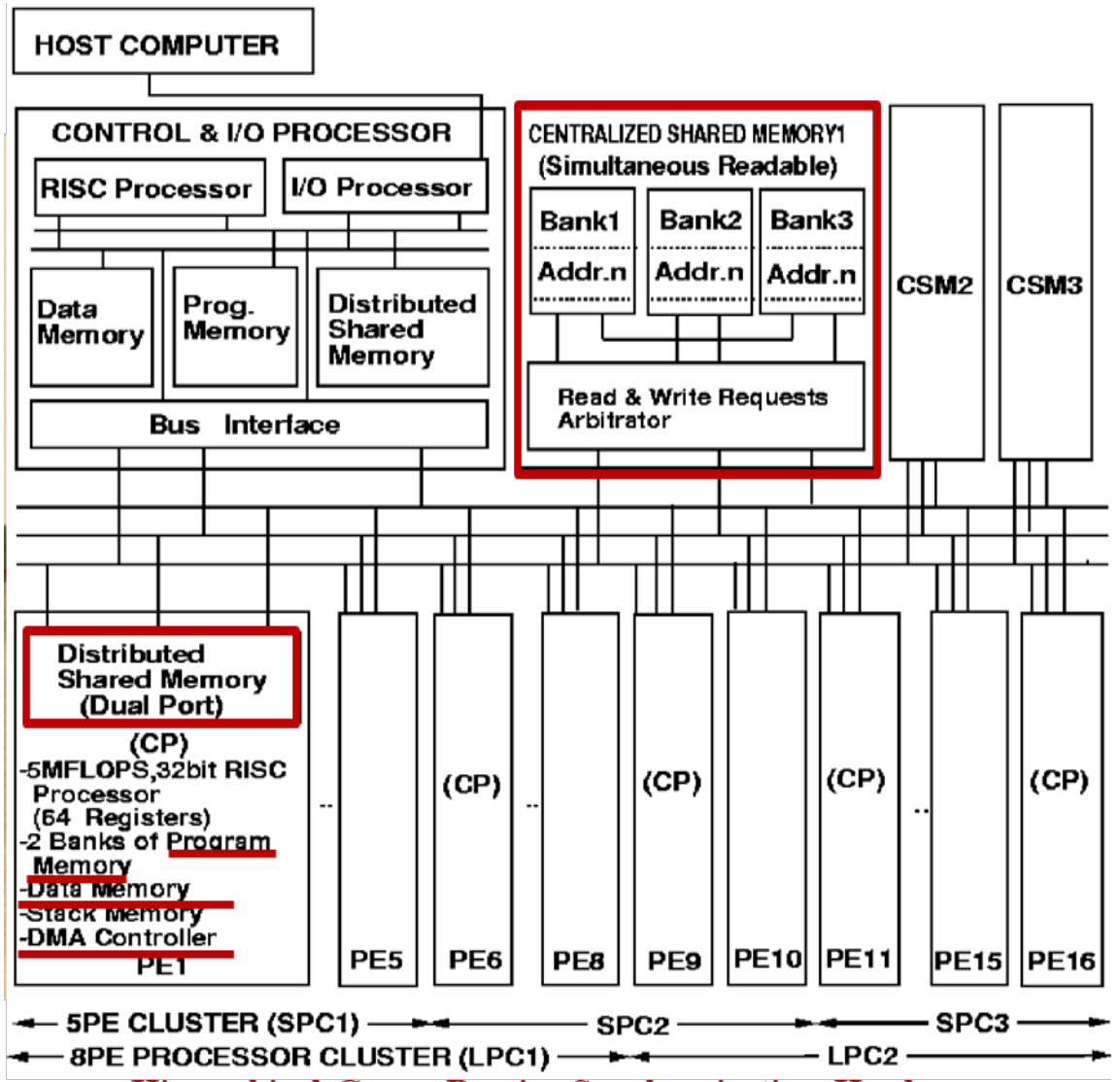
https://corporate-awards.ieee.org/recipients/ieee-medal-of-honor-recipients/

The First Compiler Codesigned Multiprocessor

OSCAR (Optimally Scheduled Advanced Multiprocessor) in 1987

AMD29325 32-bit Floating-point unit

H. Kasahara, "OSCAR Fortran Multigrain Compiler", Stanford University, Hosted by Professor John L. Hennessy and Professor Monica Lam, May. 15. 1995.



AMD29325 32-bit Floating-point unit

Hierarchical Group Barrier Synchronization Hardware

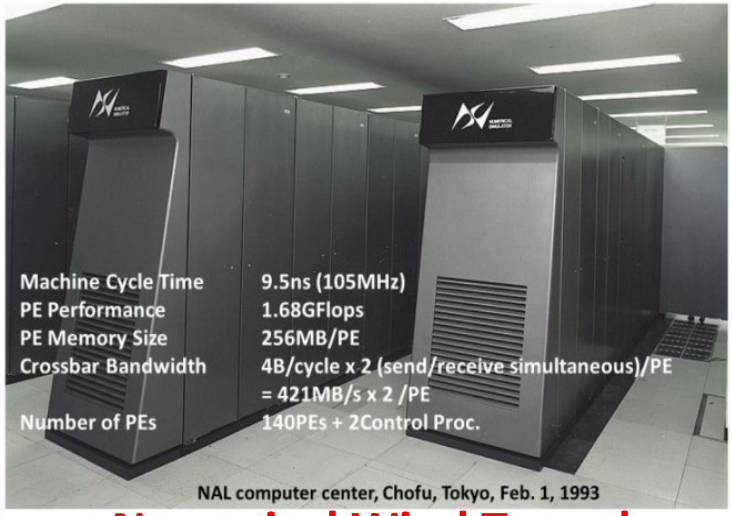
H. Kasahara was a member of 3 World No.1 Supercomputers
"NWT：Kasahara's OSCAR Architecture", "Earth Simulator", and "K"

Mr. Hajime Miyoshi

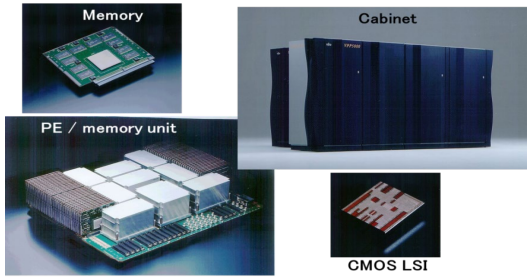
National Aerospace Laboratory

Father of Japanese Supercomputer

Waseda Alumnus, Leader of NWT, Earth Simulator



NWT（Numerical Wind Tunnel）
Kasahara's OSCAR Architecture,
1993,1.68GFLOPS
<Fujitsu VPP 500, 5000>



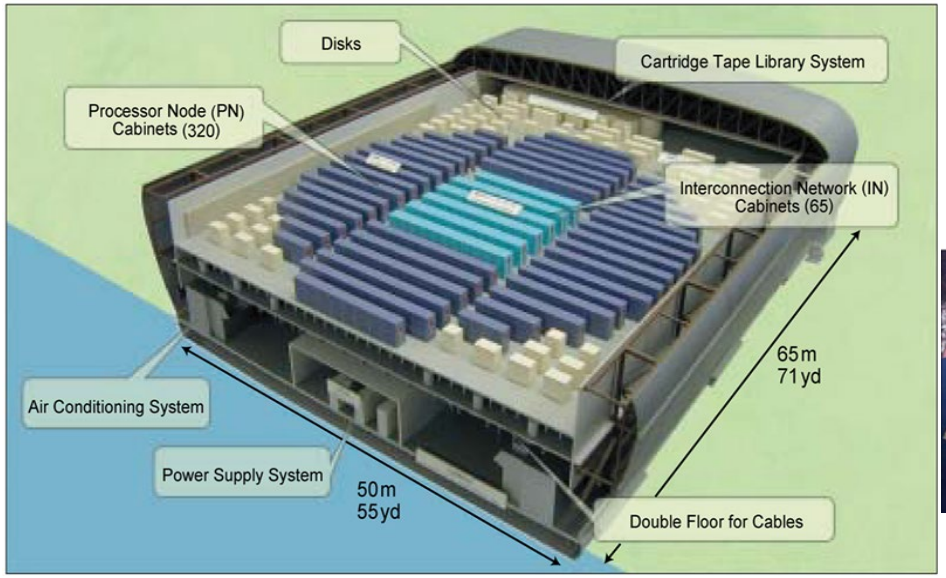
Numerical Wind Tunnel

Earth Simulator, 2002

40 TFLOPS Peak ($40*10^{12}$)

35.6 TFLOPS Linpack, 3.2MW

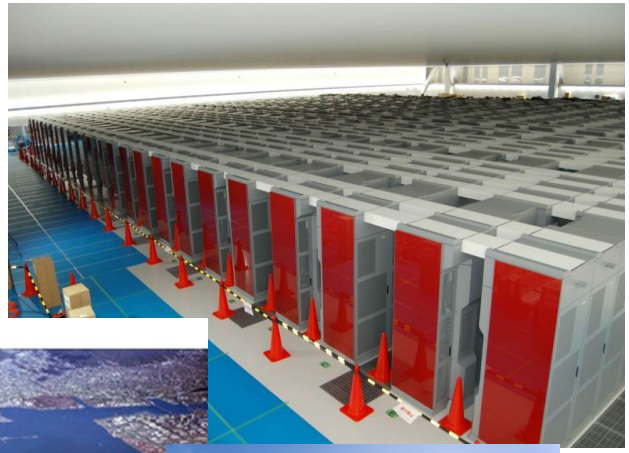
Image of Earth Simulator



K「京」, 2011

10PFLOPS

11.3MW



OSCAR Parallelizing Compiler

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

Multigrain Parallelization(LCPC1991,2001,04)

coarse-grain parallelism among loops and subroutines (2000 on SMP), near fine grain parallelism among statements (1992) in addition to loop parallelism

Data Localization

Automatic data management for distributed shared memory, cache and local memory
(Local Memory 1995, 2016 on RP2,Cache2001,03)
Software Coherent Control (2017)

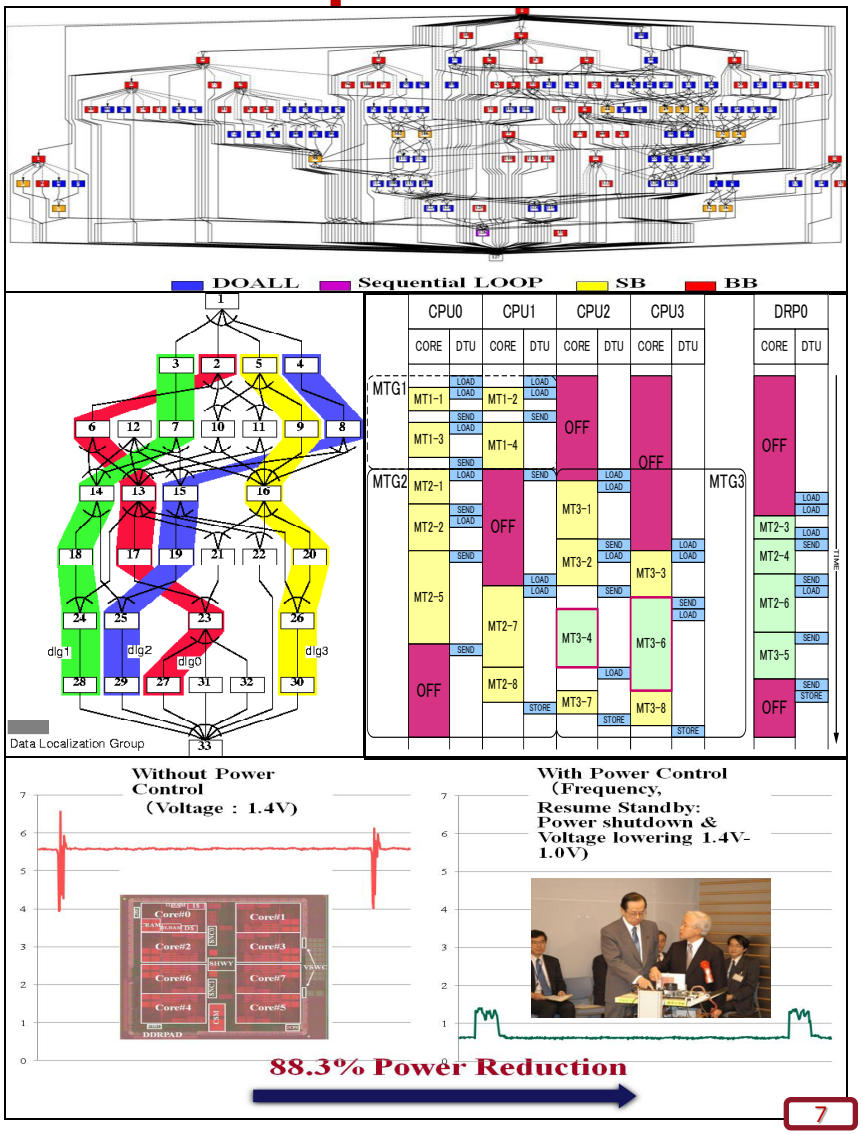
Data Transfer Overlapping(2016 partially)

Data transfer overlapping using Data Transfer Controllers (DMAs)

Power Reduction

(2005 for Multicore, 2011 Multi-processes, 2013 on ARM)

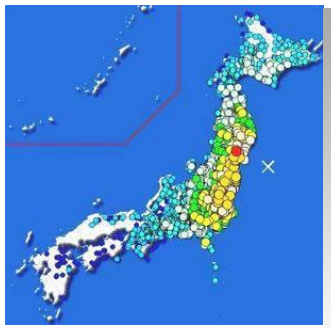
Reduction of consumed power by compiler control DVFS and Power gating with hardware supports.



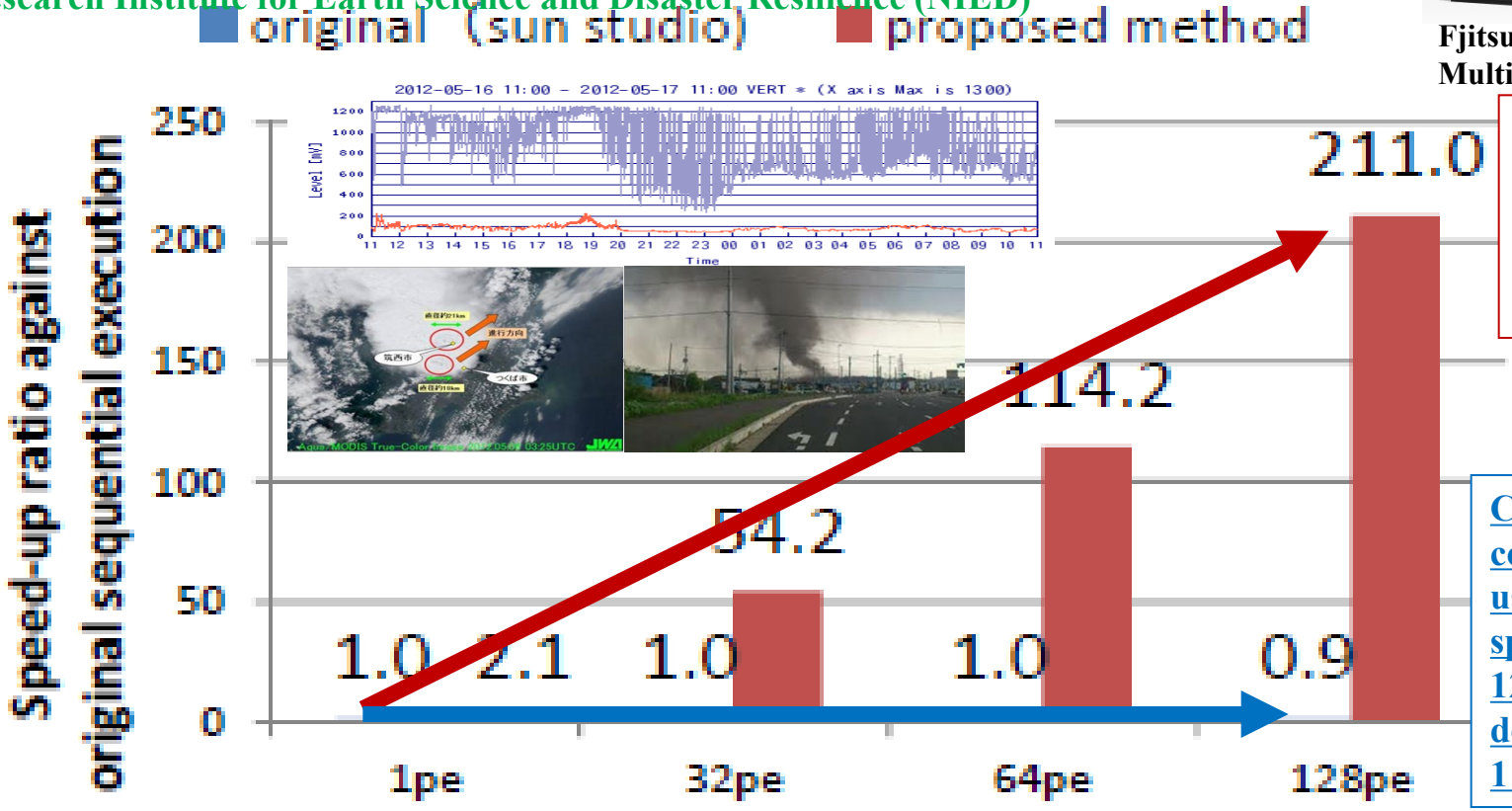
Parallel Soft is important for scalable performance of multicore (LCPC2015)

- Just more cores don't give us speedup
- Development cost and period of parallel software are getting a bottleneck of development of embedded systems, eg. IoT, Automobile

Earthquake wave propagation simulation GMS developed by National Research Institute for Earth Science and Disaster Resilience (NIED)



Fjitsu M9000 SPARC Multicore Server



OSCAR Compiler gives us 211 times speedup with 128 cores

Commercial compiler gives us 0.9 times speedup with 128 cores (slow-downed against 1 core)

- Automatic parallelizing compiler available on the market gave us no speedup against execution time on 1 core on 64 cores
  - Execution time with 128 cores was slower than 1 core (0.9 times speedup)
- Advanced OSCAR parallelizing compiler gave us 211 times speedup with 128cores against execution time with 1 core using commercial compiler
  - OSCAR compiler gave us 2.1 times speedup on 1 core against commercial compiler by global cache optimization

INDUSTRY https://www.ans.org/news/article-5842/amazon-buys-nuclearpowered-data-center-from-talen/

Amazon buys nuclear-powered data center from Talen

Thu, Mar 7, 2024, 10:01PM | Nuclear News

Susquehanna nuclear plant in Salem Township, Penn., along with the data center in foreground. (Photo: Talen Energy)

Talen Energy announced its sale of a 960-megawatt data center campus to cloud service provider Amazon Web Services (AWS), a subsidiary of Amazon, for $650 million.

The data center, Cumulus Data Assets, sits on a 1,200-acre campus in Pennsylvania and is directly powered by the adjacent Susquehanna Steam Electric Station, which generates 2.5 gigawatts of power.

Laramie County approves construction of what could become the largest data center in US

Wyoming is poised to become an artificial-intelligence powerhouse after Laramie County commissioners earlier this month unanimously voted to move forward with the construction of a 1.8 gigawatt data center designed to eventually scale up to 10 gigawatts, which would be the largest single AI campus in the U.S.
January 20, 2026

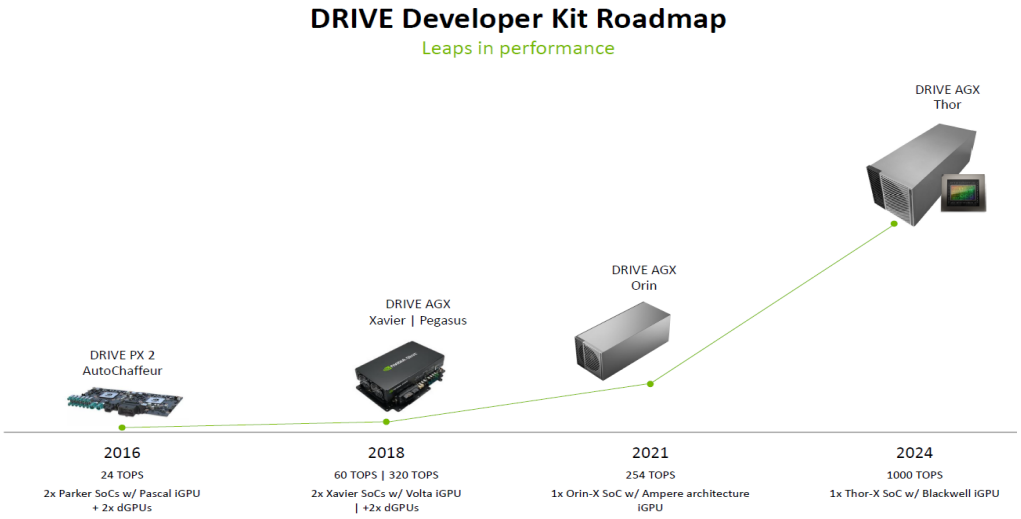
https://wyofile.com/laramie-county-approves-construction-of-what-could-become-the-largest-data-center-in-us/

Self Driving Cars (自動運転)

NVIDIA DRIVE AGX Thor Development Platform

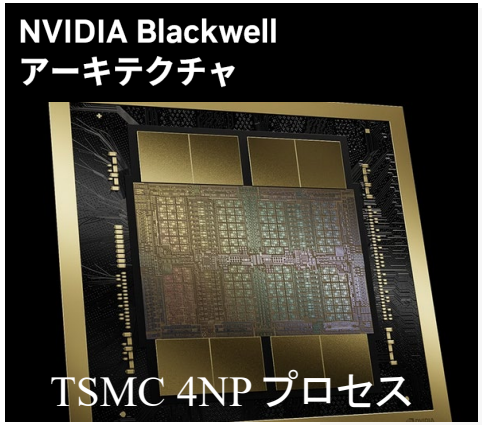
December 2025

Deep Leaning （多層ニューラルネット）
推論（Reasoning）

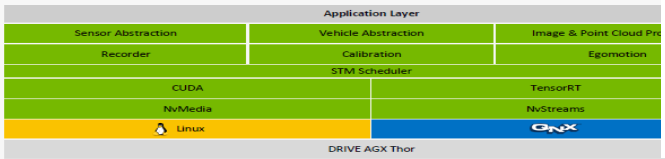


Spec Overview

Components		
Thor SoC	GPU	Integrated Blackwell CUDA Tensor Core GPU
	Accelerators	Programmable Vision Accelerator (PVA) Optical Flow Accelerator (OFA)
	CPU	ARM Neoverse V3AE. Arm64 (v9.2-A), SMP
Safety MCU		Renesas U2A16
Storage		256 GB UFS
Power Supply		Built-In
Vehicle Wiring Harness		Additional Accessory
Performance		
DL Inference		Up to 1,000 INT8 TOPS | 2,000 FP4 FLOPS
Memory Bandwidth		273 GB/s
System RAM		64 GB LPDDR5X at 4266 MHz
Operating Parameters		
Temperature		0 to 35°C (SKU10) | 0 to 45°C (SKU12)
System Power		350 W
Voltage		9 V to 16 V (Static), 7 V to 32 V (Transient)



https://www.nvidia.com/ja-jp/data-center/technologies/blackwell-architecture/

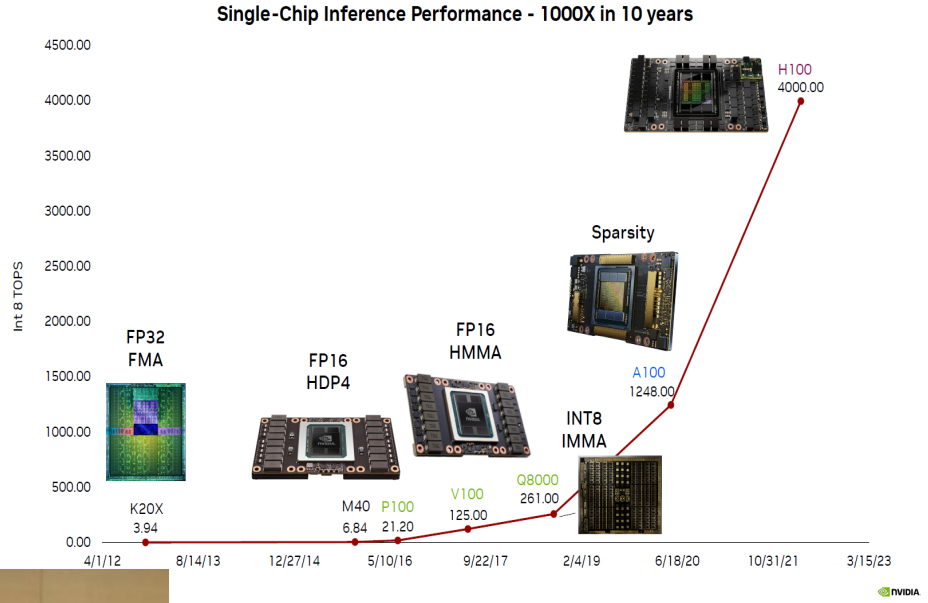


https://hc2023.hotchips.org/

Hardware for Deep Learning

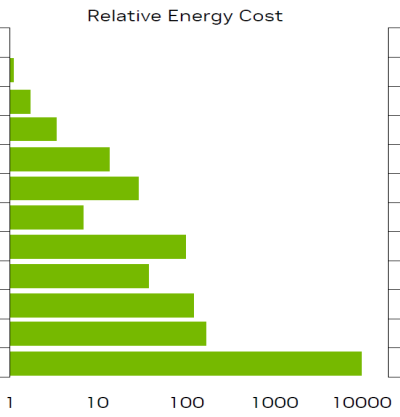
Hot Chips
August 29, 2023

Bill Dally
Chief Scientist and SVP of Research, NVIDIA Corporation
Adjunct Professor of CS and EE, Stanford

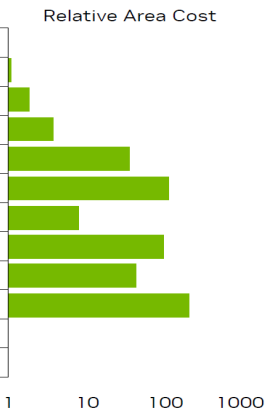


Cost of Operations

Operation:	Energy (pJ)
8b Add	0.03
16b Add	0.05
32b Add	0.1
16b FP Add	0.4
32b FP Add	0.9
8b Mult	0.2
32b Mult	3.1
16b FP Mult	1.1
32b FP Mult	3.7
32b SRAM Read (8KB)	5
32b DRAM Read	640

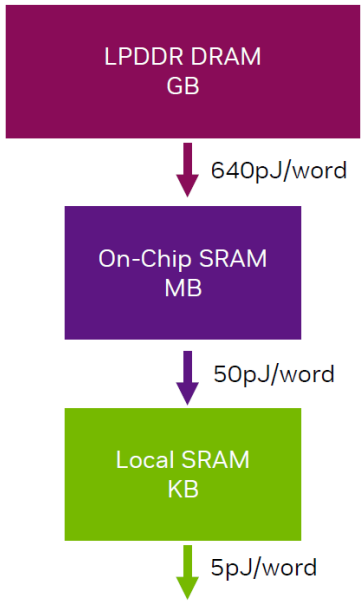


Area (μm^2)
36
67
137
1360
4184
282
3495
1640
7700
N/A
N/A



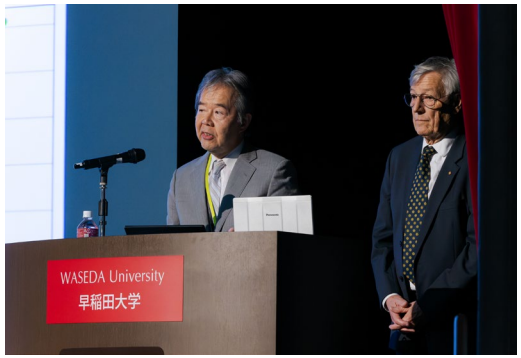
Energy numbers are from Mark Horowitz "Computing's Energy Problem (and what we can do about it)", ISSCC 2014
Area numbers are from synthesized result using Design Compiler under TSMC 45nm tech node. FP units used DesignWare Library.

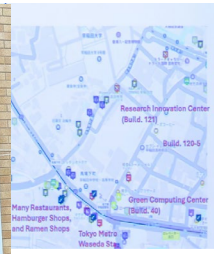
The Importance of Staying Local



ACM/IEEE ISCA'25 General Co-Chairs

Prof. Jean-Luc Gaudiot (U. California, Irvine) & Prof. Hironori Kasahara (Waseda U.)



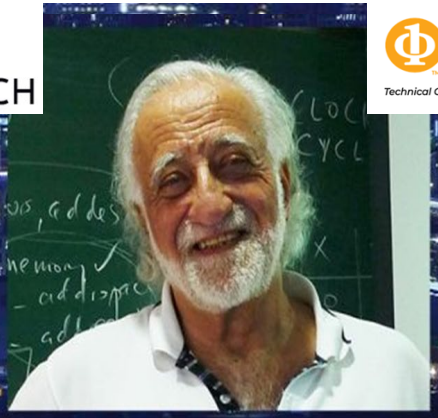


Prof. Keiji Kimura:
Japan Operation Chair

ACM/IEEE ISCA'25 Super Panel, June 22, 2025 @Waseda

ACM SIGARCH

IEEE COMPUTER SOCIETY TCCA
Technical Committee on Computer Architecture



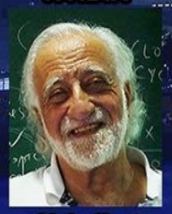
Special Invited Talk on Sunday
Time: 3:30 PM - 4:30 PM, June 22nd JST
Venue: Okuma Auditorium, Waseda University
Speaker: Prof. Yale Patt (UT Austin)
Title: "Solving the Sustainability Problem -- What must come first. ...and 2nd. ...and 3rd."

Dejan Milojicic
HPE
(Moderator)

Wen-Mei Hwu
NVIDIA

Norm Jouppi
Google

Hironori Kasahara
Waseda Univ.



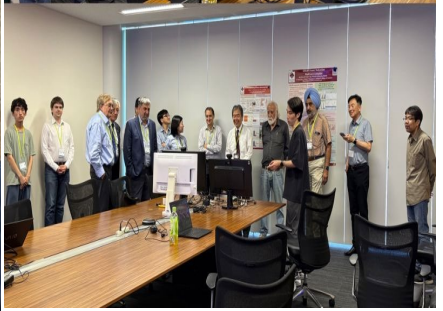
Yale Patt
UT Austin

Guri Sohi
Wisconsin Univ.

Carole-Jean Wu
Meta

Special Panel on "Sustainable Computer Architecture"
Time: 4:45 PM - 6:15 PM, June 22nd JST
Venue: Okuma Auditorium, Waseda University

General Co-Chairs
Prof. Jean-Luc Gaudiot (U. California, Irvine)
Prof. Hironori Kasahara (Waseda U.)



ACM/IEEE PACT

(International Conference on Parallel Architectures and Compilation Techniques)

Nov., 2027 @Waseda University

General Chair: Hironori Kasahara

PACT 2025 November 3-6, 2025

Home | Submit | AE | Attend | Program | Tutorials | ACM SRC | Sponsoring | Organization

PACT 2025

November 3-6, 2025

UC Irvine Beall Innovation Center

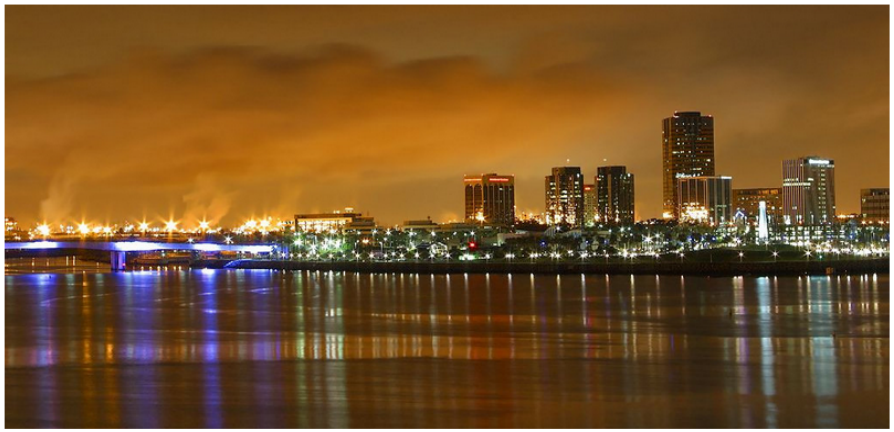
Irvine, California, USA

IEEE COMPUTER SOCIETY

ACM SIGARCH

The International Conference on Parallel Architectures and Compilation Techniques (PACT) is a unique technical conference sitting at the intersection of hardware and software, with a special emphasis on parallelism. The PACT conference series brings together researchers from computer architectures, compilers, execution environments, programming languages, and applications, to present and discuss their latest research results.

PACT 2025 will be held as an in-person event in Irvine, California, USA. At least one of the authors of accepted papers will be required to attend the conference, and we encourage all the authors to participate.



ACM チューリング賞 コンピュータ分野のノーベル賞

A.M. TURING CENTENARY CELEBRATION WEBCAST

MORE ACM AWARDS

Prof. Yoshua Bengio
モントリオール大学
2024年3月7日
大川賞授賞式
Okawa Prize

Turing Award › Winners

Jeffrey Ullman
2020 アルゴリズムと プログラミング言語

Geoffrey Hinton
2018 AI・ディープ・ラーニング
2024ノーベル物理学賞

John L. Hennessy
2017 RISC:スマホ-スパコン
2016大川賞
スタンフォード大前学長・Alphabet(Google親会社)会長

Alfred Aho
2020 アルゴリズムと プログラミング言語

Yoshua Bengio
2018 AI・ディープ・ラーニング
2023大川賞

Tim Berners-Lee
2016 World Wide Web

ディズニー・アニメーション・スタジオ&ピクサーの元社長
Edwin Catmull アニメーションと 3Dグラフィックス
2019
トイ・ストーリー,モンスターズ・インク

Yann LeCun
2018 AI・ディープ・ラーニング

Whitfield Diffie
2015 公開鍵暗号

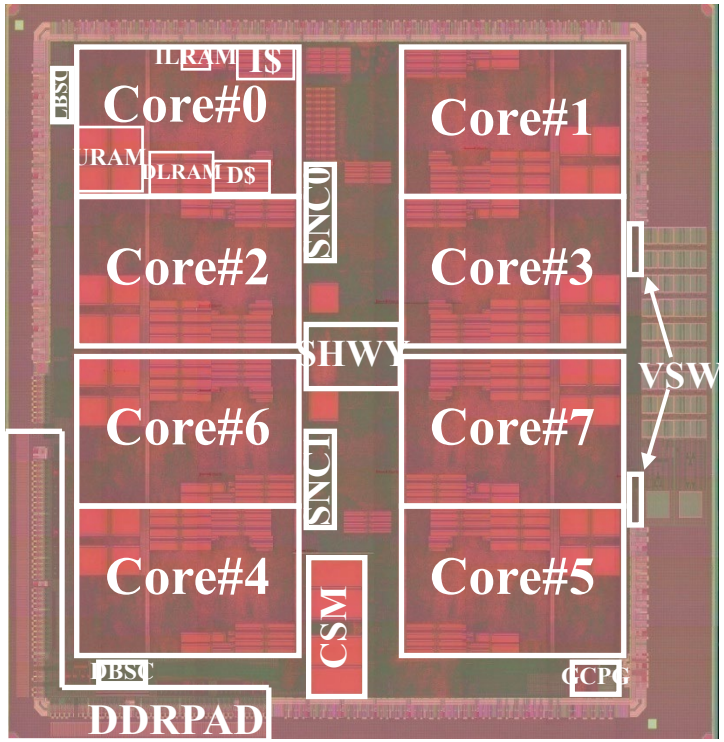
Pat Hanrahan
2019 アニメーションと 3Dグラフィックス

カリフォルニア大バークレー名誉教授, ACM元会長
David A Patterson
2017 RISC:スマホ-スパコン

Martin Hellman
2015 公開鍵暗号

Multicores for Performance and Low Power

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



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”

$$\text{Power} \propto \text{Frequency} * \text{Voltage}^2$$

(Voltage ∝ Frequency)

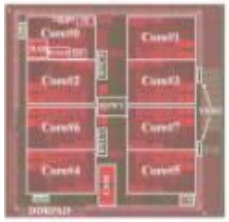
$$\Rightarrow \text{Power} \propto \text{Frequency}^3$$

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

<Multicores>

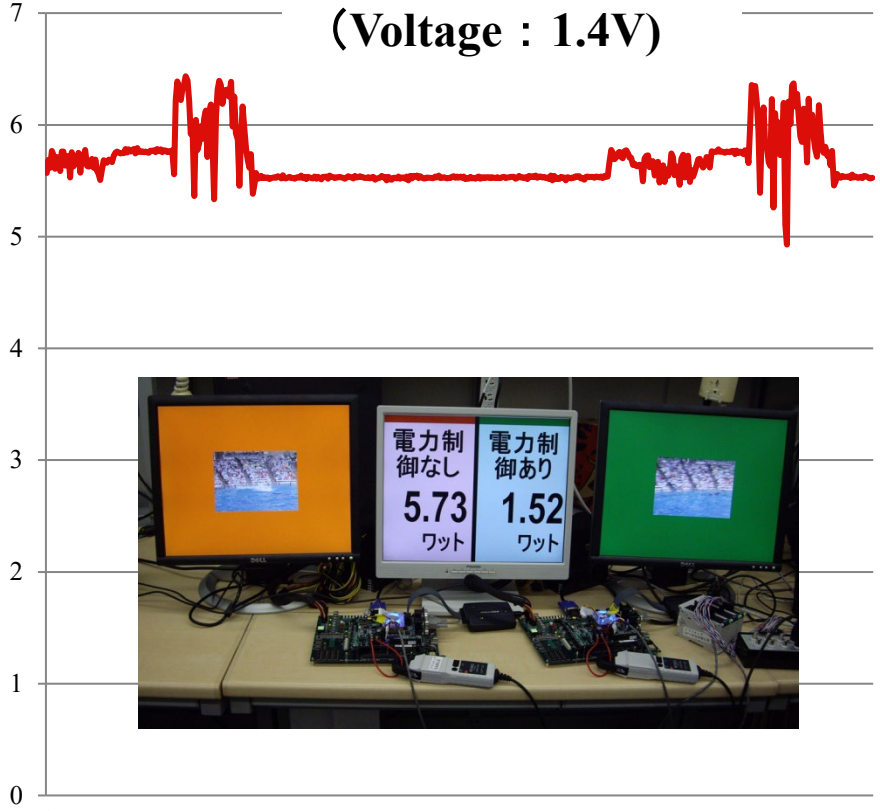
If 8cores are integrated on a chip, Power is still 1/8 and Performance becomes 2 times.

Power Reduction of MPEG2 Decoding to 1/4 on 8 Core Homogeneous Multicore RP-2 by OSCAR Parallelizing Compiler

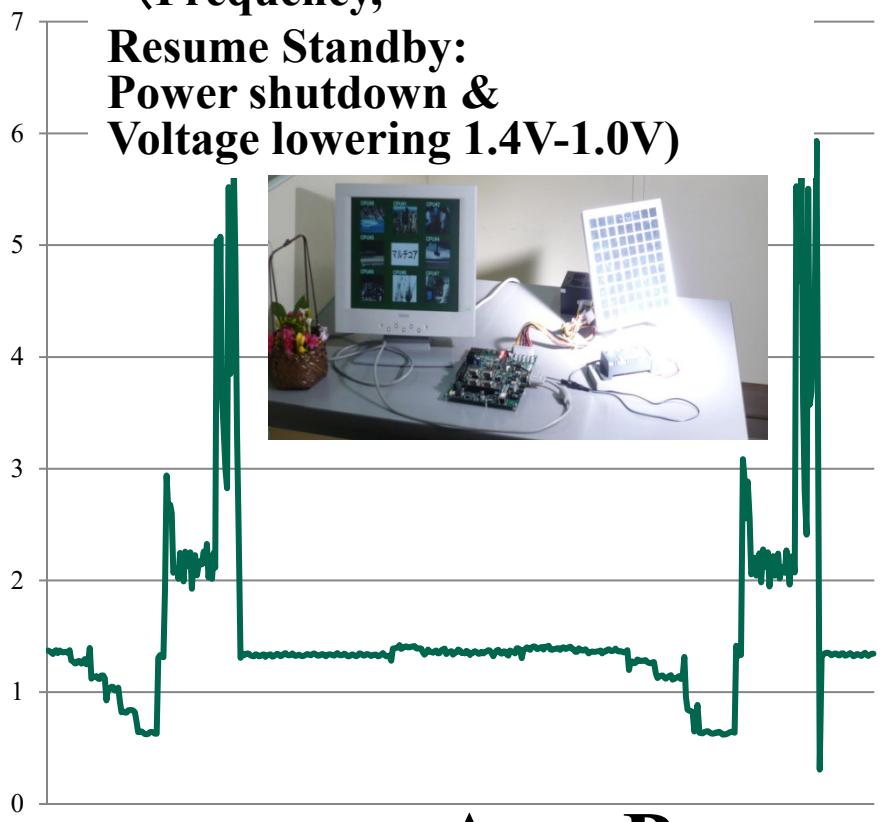


MPEG2 Decoding with 8 CPU cores

Without Power Control (Voltage : 1.4V)



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



Avg. Power 5.73 [W]

73.5% Power Reduction

Avg. Power 1.52 [W]

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

http://www8.cao.go.jp/cstp/gaiyo/honkaigi/74index.html

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)

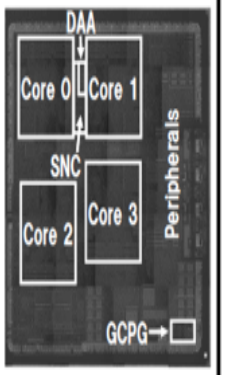
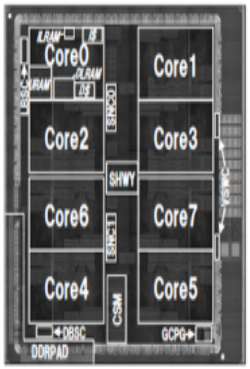
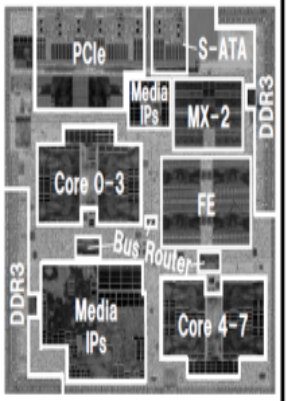
第74回総合科学技術会議の様子(2)

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)

Codesign of Compiler and Multiprocessor Architecture since 1985

4 core multicore RP1 (2007), 8 core multicore RP2 (2008) and 15 core Heterogeneous multicore RPX (2010) developed in NEDO Projects with Hitachi and Renesas

RP-1 (ISSCC2007 #5.3)	*RP-2* (ISSCC2008 #4.5)	*RP-X* (ISSCC2010 #5.3)
		
90nm, 8-layer, triple-Vth, CMOS	90nm, 8-layer, triple-Vth, CMOS	45nm, 8-layer, triple-Vth, CMOS
97.6 mm^2 (9.88 x 9.88 mm)	104.8 mm^2 (10.61 x 9.88 mm)	153.8 mm^2 (12.4 x 12.4 mm)
1.0V (internal), 1.8/3.3V (I/O)	1.0-1.4V (internal), 1.8/3.3V (I/O)	1.0-1.2V (internal), 1.2-3.3V (I/O)
600MHz ,4.32 GIPS,16.8 GFLOPS	600MHz , 8.64 GIPS, 33.6 GFLOPS	648MHz, 13.7GIPS, 115GOPS, 36.2GFLOPS
11.4 GOPS/W(32b換算)	18.3 GOPS/W(32b換算)	37.3 GOPS/W(32b換算)

Prime Minister FUKUDA is touching our multicore chip during execution.

A Strategic Initiative of Computing: Systems and Applications (SISA)- Integrating HPC, Big Data, AI and Beyond, Jan.18-19, 2017

A Strategic Initiative of Computing:Systems and Applications (SISA) --Integrating HPC, Big Data, AI and Beyond-- Jan. 18-19, 2017

Opening: Prof. Gao, Prof. Kasahara

Waseda VP Shuji Hashimoto

I. Architecture and Applications

Keynote: William J. Dally,
NVIDIA and Stanford University, USA

- Kimihiko Hirao, RIKEN, Japan
- G. W. Yang, Tsinghua Univ. China
- J. Sexton, IBM, USA

II . System Software and Applications

Keynote： Rick. Stevens ANL, USA

- S. Mikhail Smelyanskiy Intel USA
- Fred. Streitz, LLNL USA
- R. Govind, IIS, India
- H. Hironori Kasahara, Waseda Univ,

III. Extreme Scale and Beyond

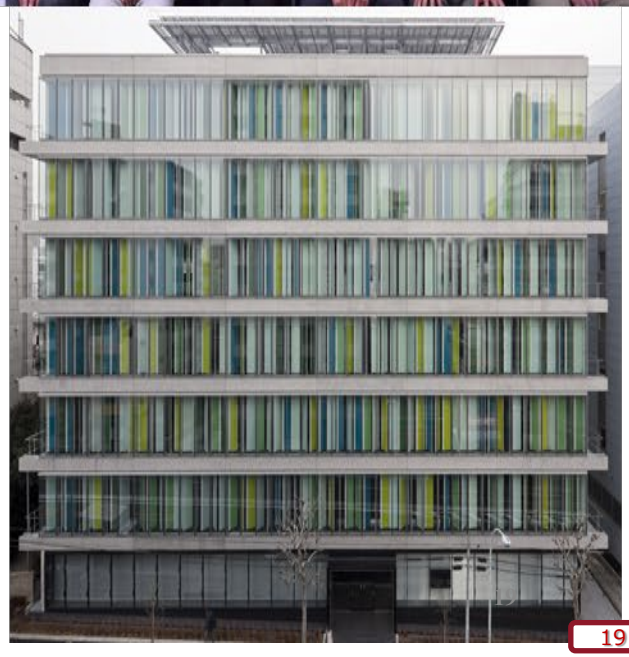
Keynote: Paul Messina ANL, USA

- Motoaki Saito, PEZY, Japan
- Eiji Ishida, MEXT, Japan
- Depei Qian, BUAA, China
- Toshiyuki Shimizu, Fujitsu, Japan

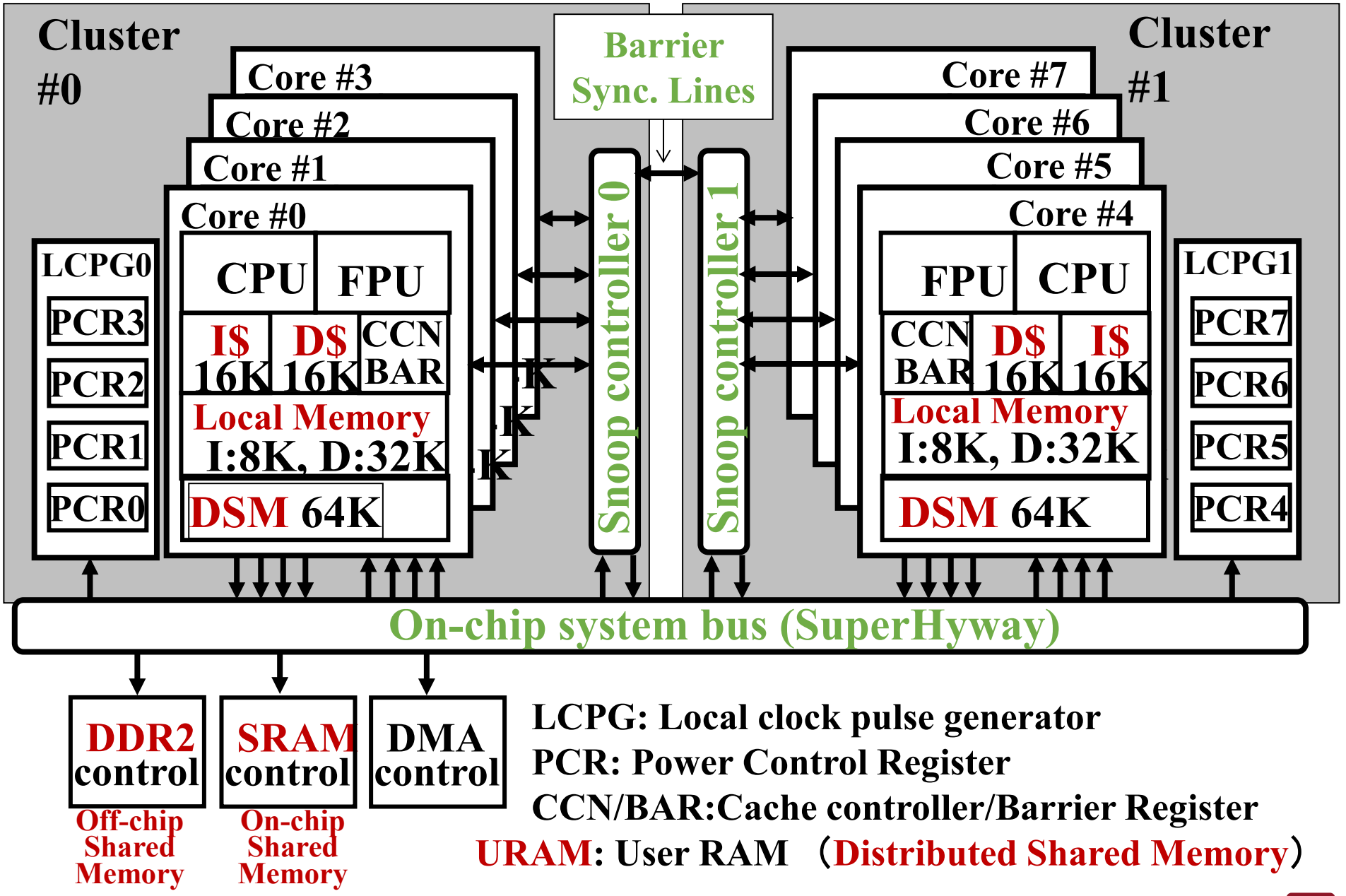
IV. Integration of HPC, Big Data, and AI

Keynote: Thomas Sterling, Indiana Univ., USA

- Masaru Kitsuregawa, NII and Univ. of Tokyo, Japan
- Thomas Schulthess, ETH, Swiss
- Moriyuki Takamura/Toshiaki Kitamura, Oscar Tech, Japan



Codesigned 8 Core RP2 Chip Block

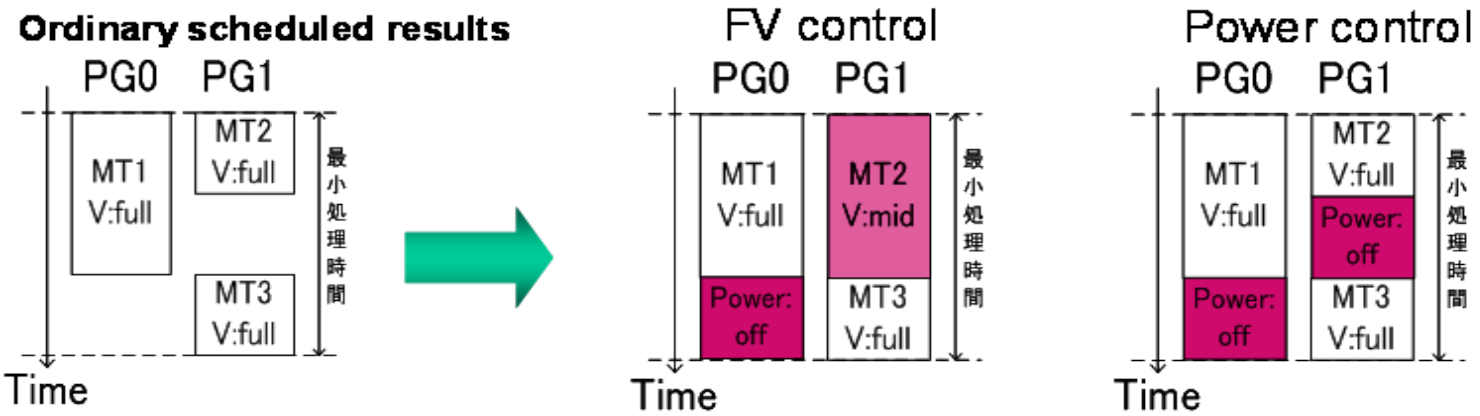


LCPG: Local clock pulse generator
PCR: Power Control Register
CCN/BAR:Cache controller/Barrier Register
URAM: User RAM （Distributed Shared Memory）

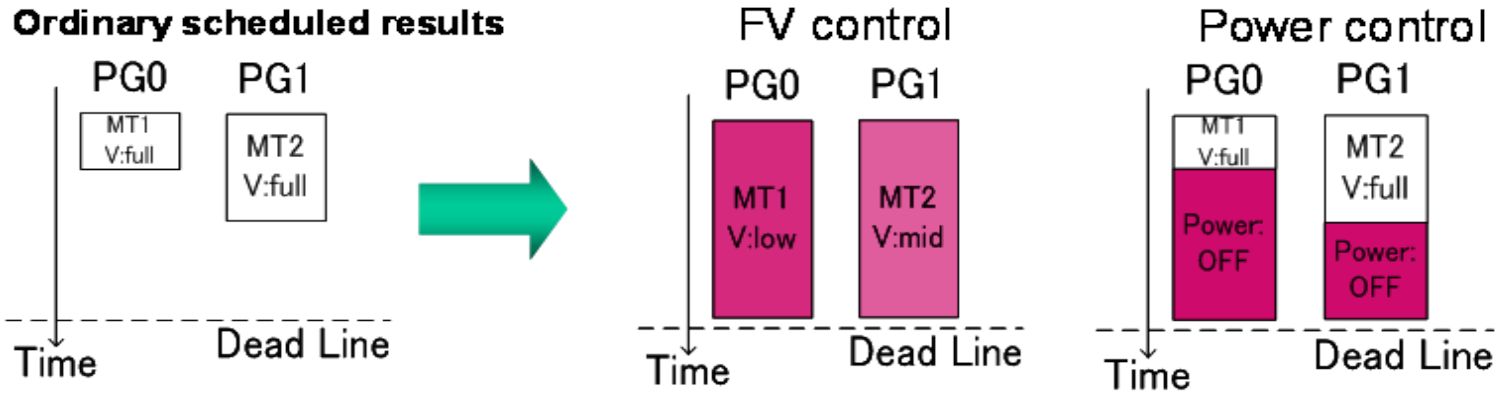
Power Reduction by Power Supply, Clock Frequency and Voltage Control by OSCAR Compiler

Frequency and Voltage (DVFS), Clock and Power gating of each cores are scheduled considering the task schedule since the dynamic power proportional to the cube of F (F^3) and the leakage power (the static power) can be reduced by the power gating (power off).

- **Shortest execution time mode**



- Realtime processing mode with dead line constraints



In this Fig. Frequency Full, Mid, Low

Power OFF: Power Gating

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**

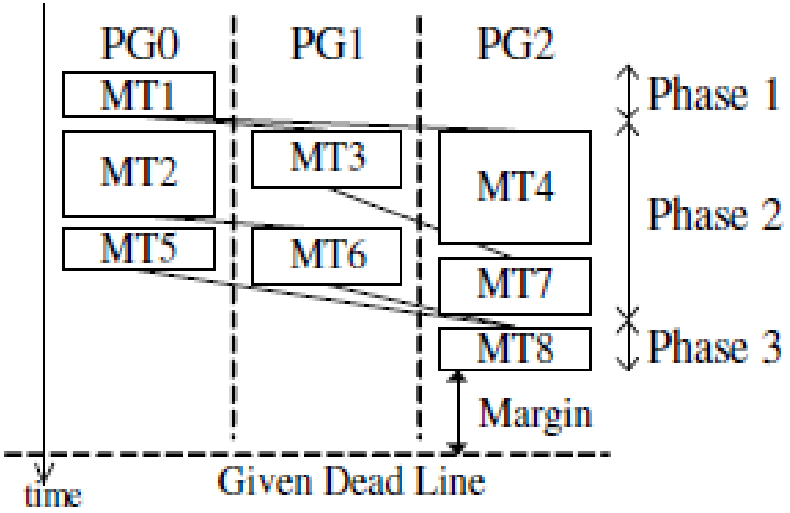
state	FULL	MID	LOW	OFF
FULL	0	40k	40k	80k
MID	40k	0	40k	80k
LOW	40k	40k	0	80k
OFF	80k	80k	80k	0

delay time [u.t.]

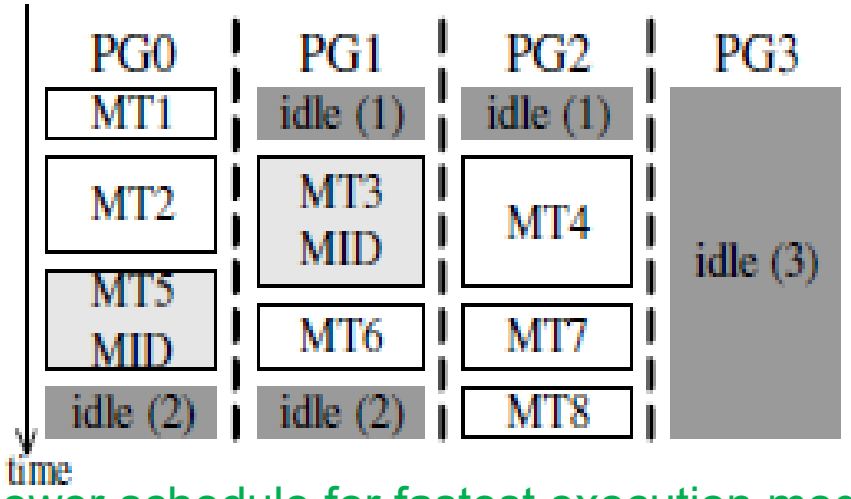
state	FULL	MID	LOW	OFF
FULL	0	20	20	40
MID	20	0	20	40
LOW	20	20	0	40
OFF	40	40	40	0

energy overhead [μJ]

Power Reduction Scheduling



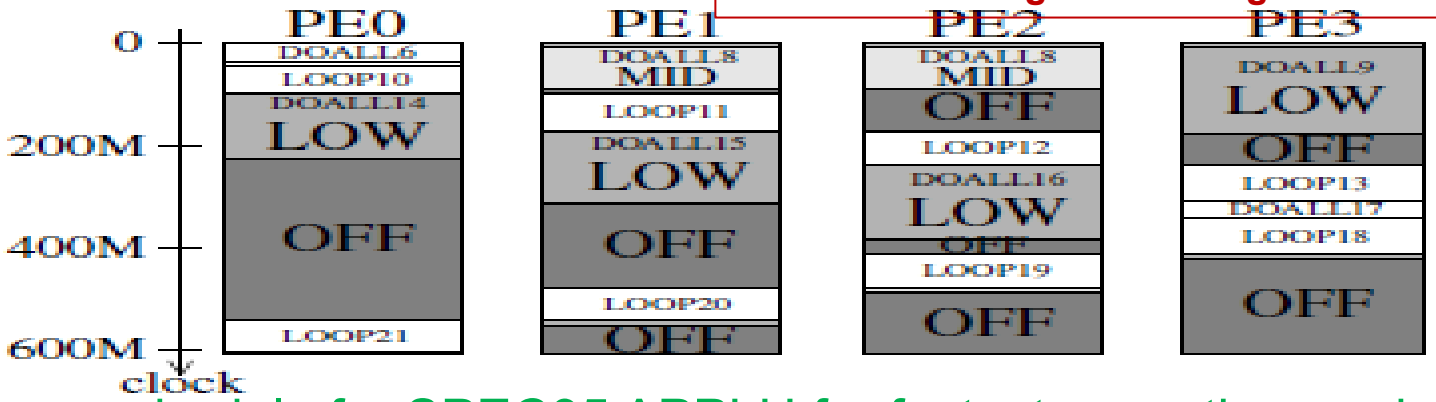
A macrotask graph assigned to 3 cores



A power schedule for fastest execution mode

Realtime scheduling mode
MTs 1,4,7,8 are on Critical Path (CP)

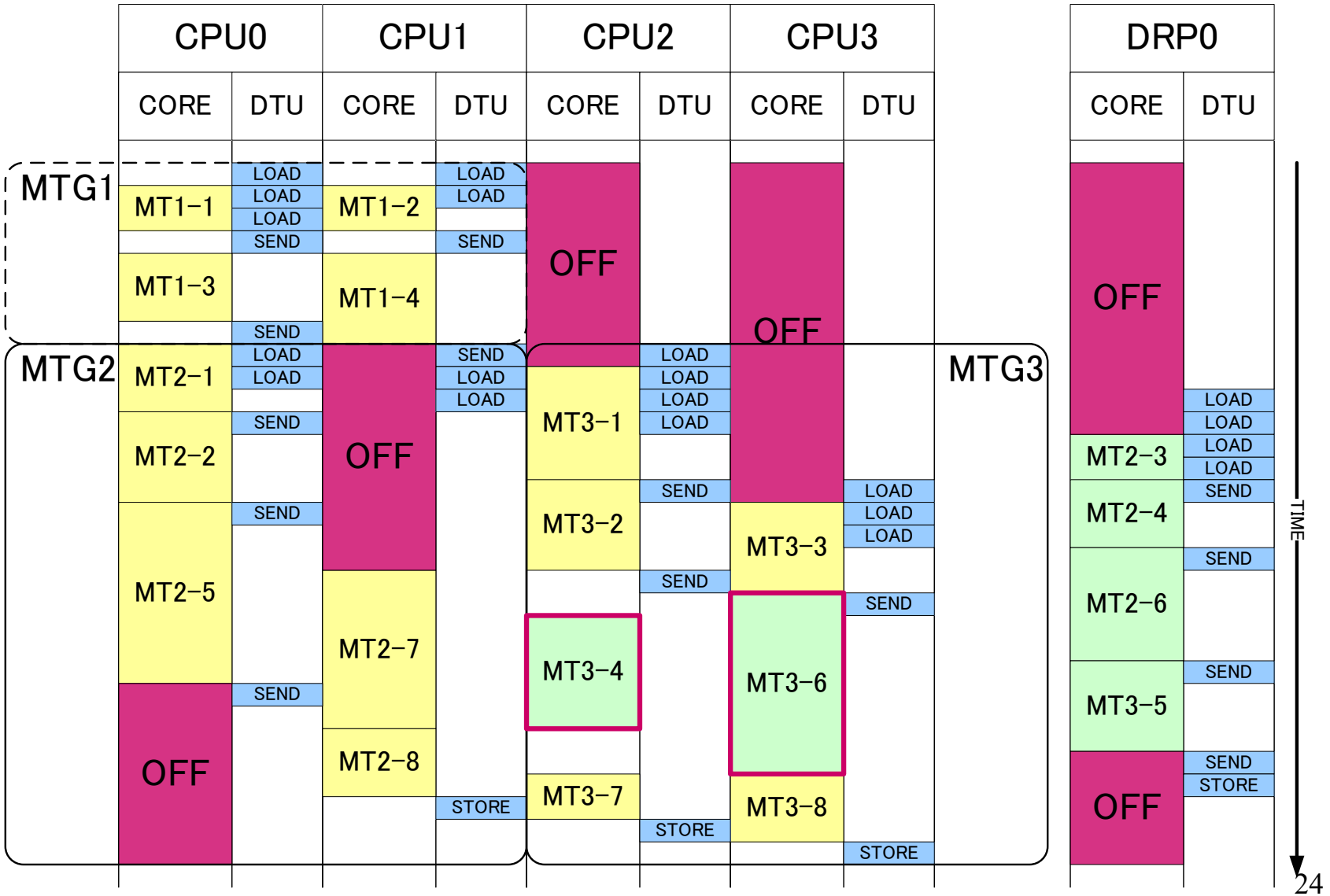
1) **Reduce frequencies (Fs) of MTs on CP considering dead line.**
2) **Reduce Fs of MTs not on CP. Idle: Clock or Power Gating considering overheads.**



A power schedule for SPEC95 APPLU for fastest execution mode

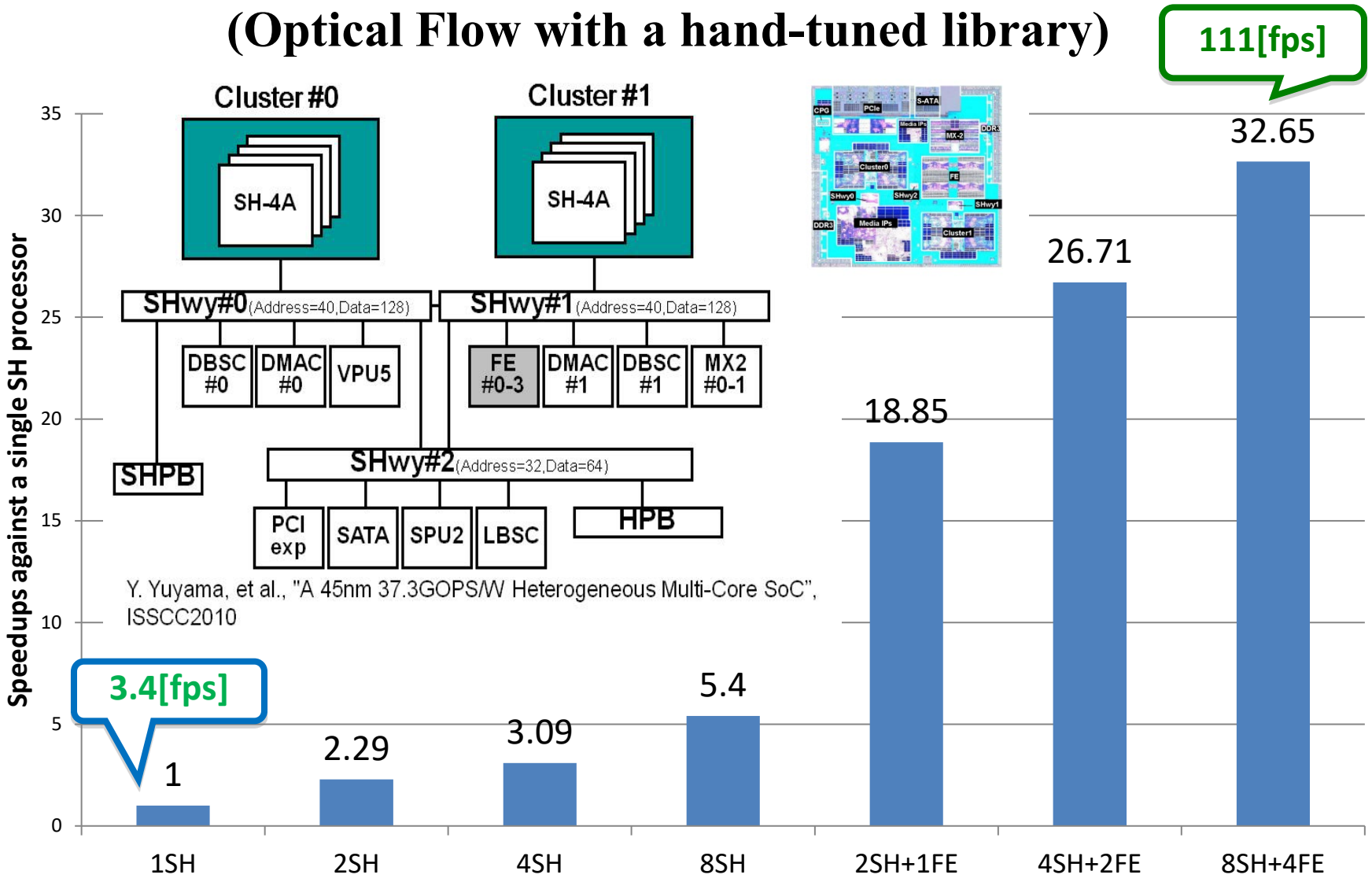
Doall6, Loop 10,11,12,13, Doall 17, Loop 18,19,20, 21 are on CP

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)



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

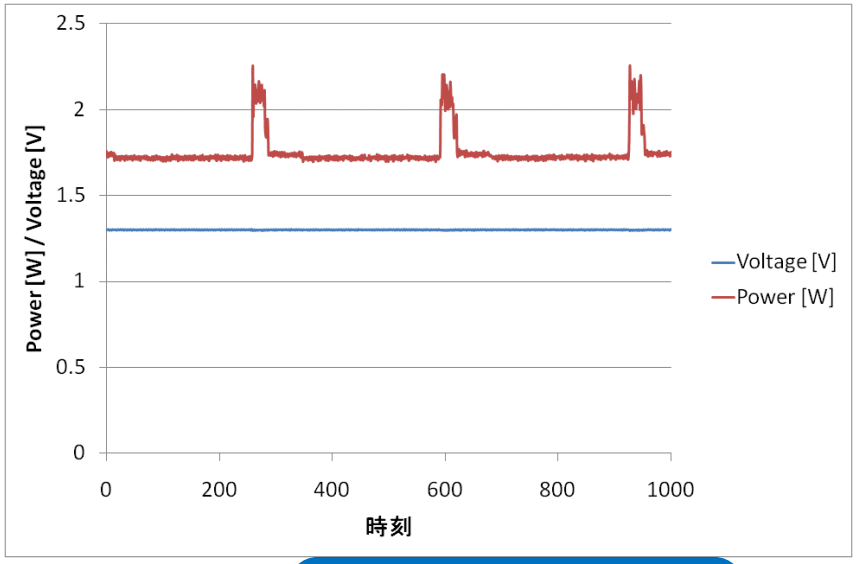
Without Power Reduction

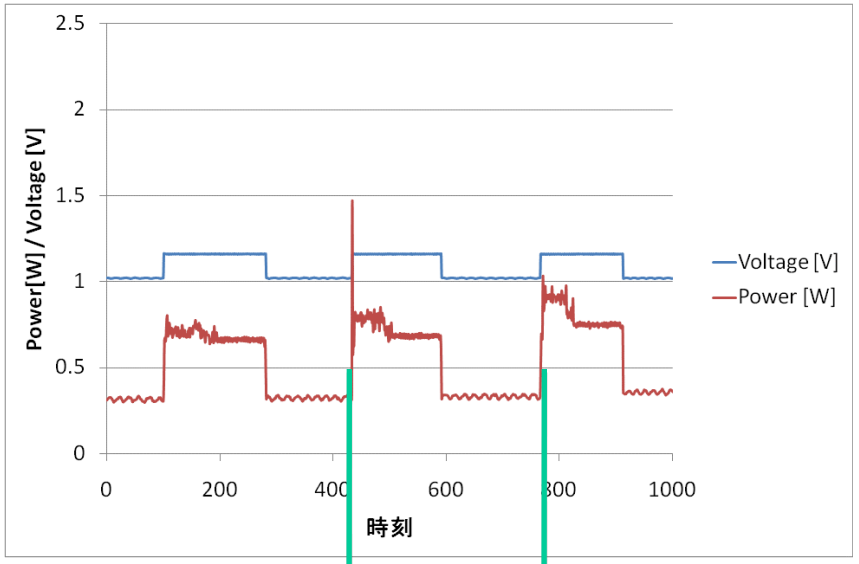
With Power Reduction by OSCAR Compiler

70% of power reduction

Average:1.76[W]

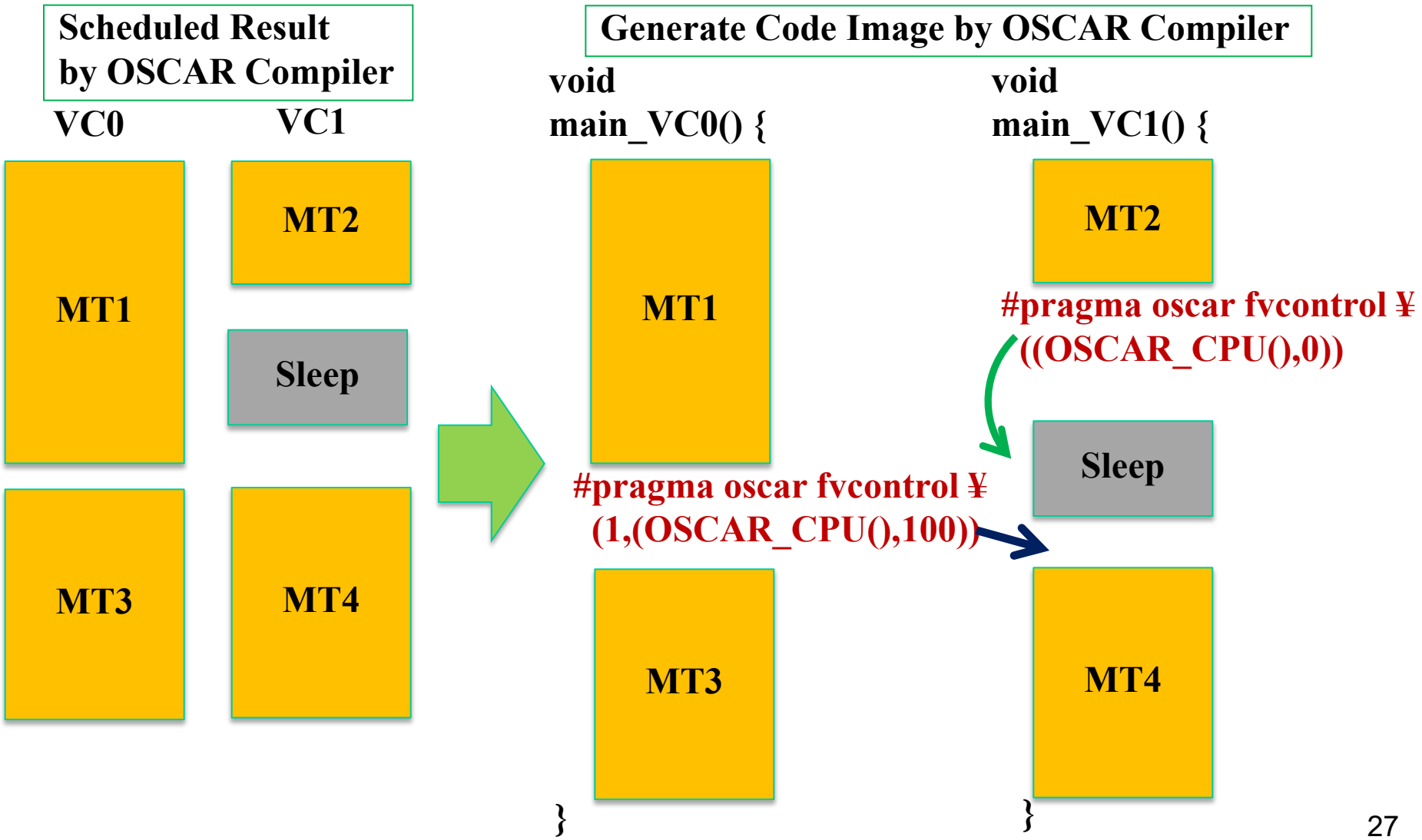
Average:0.54[W]





1cycle : 33[ms]
→30[fps]

Low-Power Optimization with OSCAR API



OSCAR API Ver. 2.0 for Homogeneous/Heterogeneous Multicores and Manycores

(LCPC2009 Homogeneous, 2010 Heterogeneous)

Specification: http://www.kasahara.cs.waseda.ac.jp/api/regist.php?lang=en&ver=2.1

List of Directives (22 directives)

- Parallel Execution API
  - parallel sections (*)
  - flush (*)
  - critical (*)
  - execution
- Memoay Mapping API
  - threadprivate (*)
  - distributedshared
  - onchipshared
- Synchronization API
  - groupbarrier
- Data Transfer API
  - dma_transfer
  - dma_contiguous_parameter
  - dma_stride_parameter
  - dma_flag_check
  - dma_flag_send

(* from OpenMP)

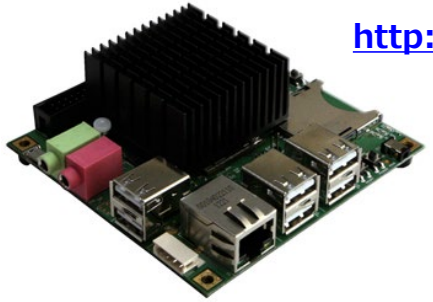
- Power Control API
  - fvcontrol
  - get_fvstatus
- Timer API
  - get_current_time
- Accelerator
  - accelerator_task_entry
- Cache Control
  - cache_writeback
  - cache_selfinvalidate
  - complete_memop
  - noncacheable
  - aligncache

2 hint directives for OSCAR compiler
- accelerator_task
- oscar_comment

from V2.0

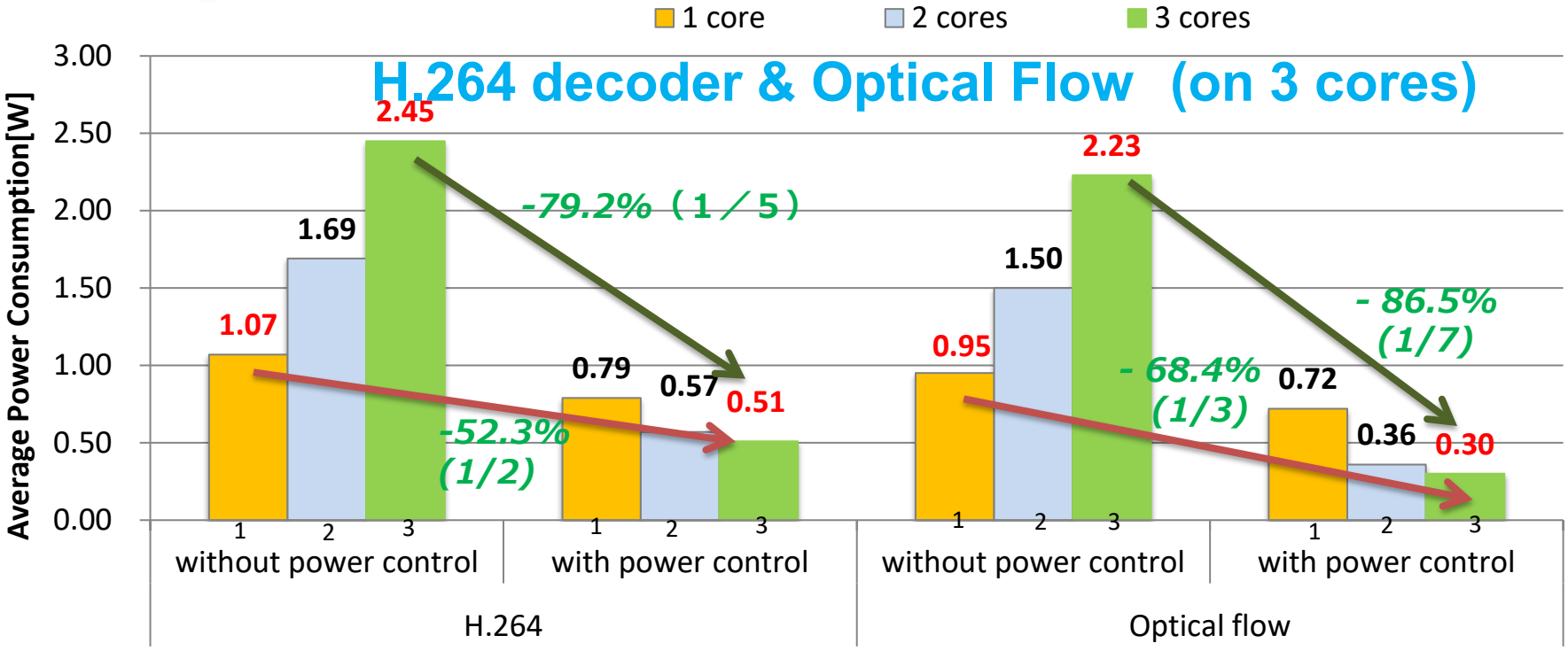
Automatic Power Reduction on ARM CortexA9 with Android

http://www.youtube.com/channel/UCS43lNYEIkC8i_KIgFZYQBQ



ODROID X2

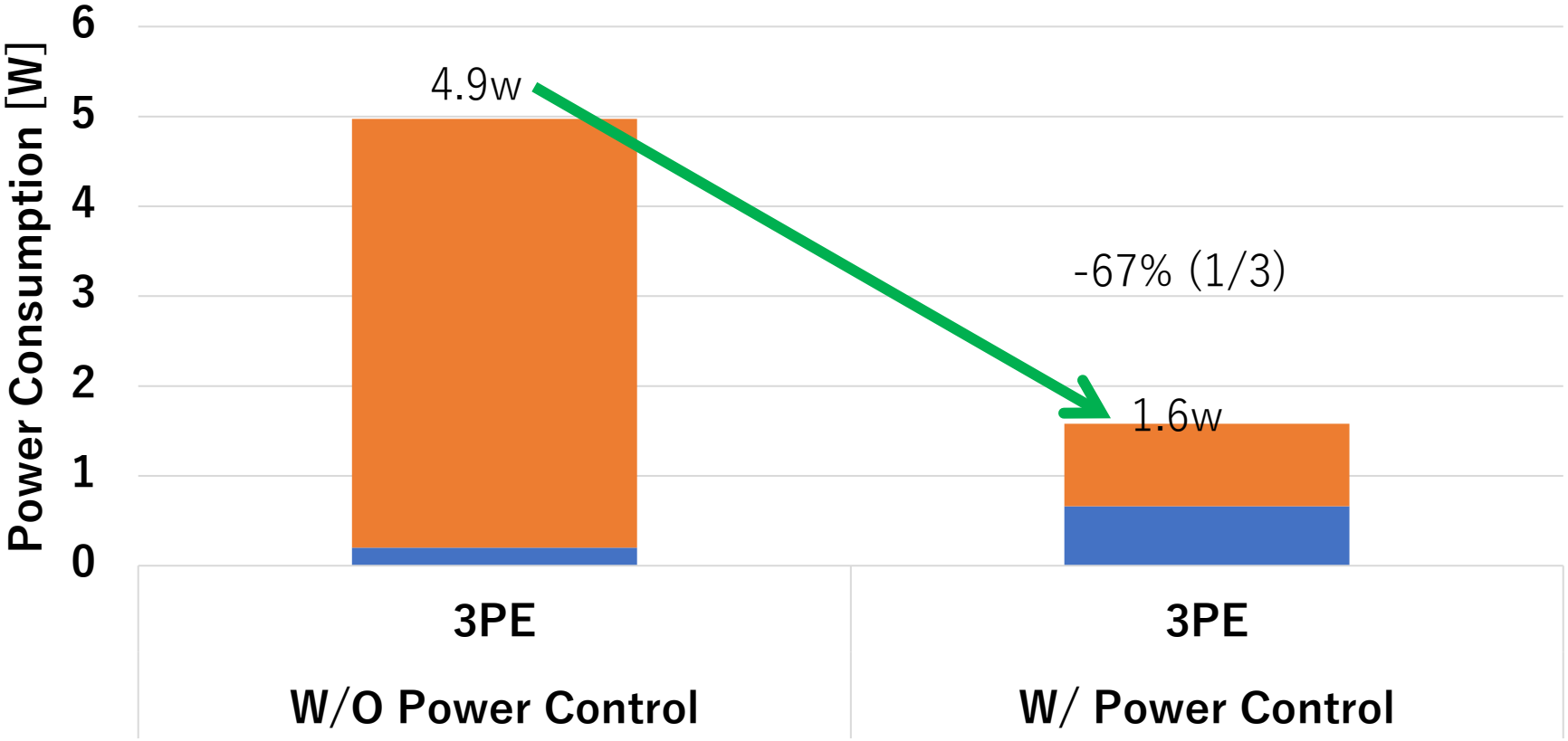
Samsung Exynos4412 Prime, ARM Cortex-A9 Quad core 1.7GHz～0.2GHz, used by Samsung's Galaxy S3



Power for 3cores was reduced to 1/5～1/7 against without software power control

Power for 3cores was reduced to 1/2～1/3 against ordinary 1core execution

Automatic Power Reduction of OpenCV Face Detection on big.LITTLE ARM Processor



- ODROID-XU3
  - Samsung Exynos 5422 Processor
    - 4x Cortex-A15 2.0GHz, 4x Cortex-A7 1.4GHz big.LITTLE Architecture
    - 2GB LPDDR3 RAM

Frequency can be changed by each cluster unit

OSCAR Parallelizing Compiler

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

Multigrain Parallelization(LCPC1991,2001,04)

coarse-grain parallelism among loops and subroutines (2000 on SMP), near fine grain parallelism among statements (1992) in addition to loop parallelism

Data Localization

Automatic data management for distributed shared memory, cache and local memory
(Local Memory 1995, 2016 on RP2,Cache2001,03)
Software Coherent Control (2017)

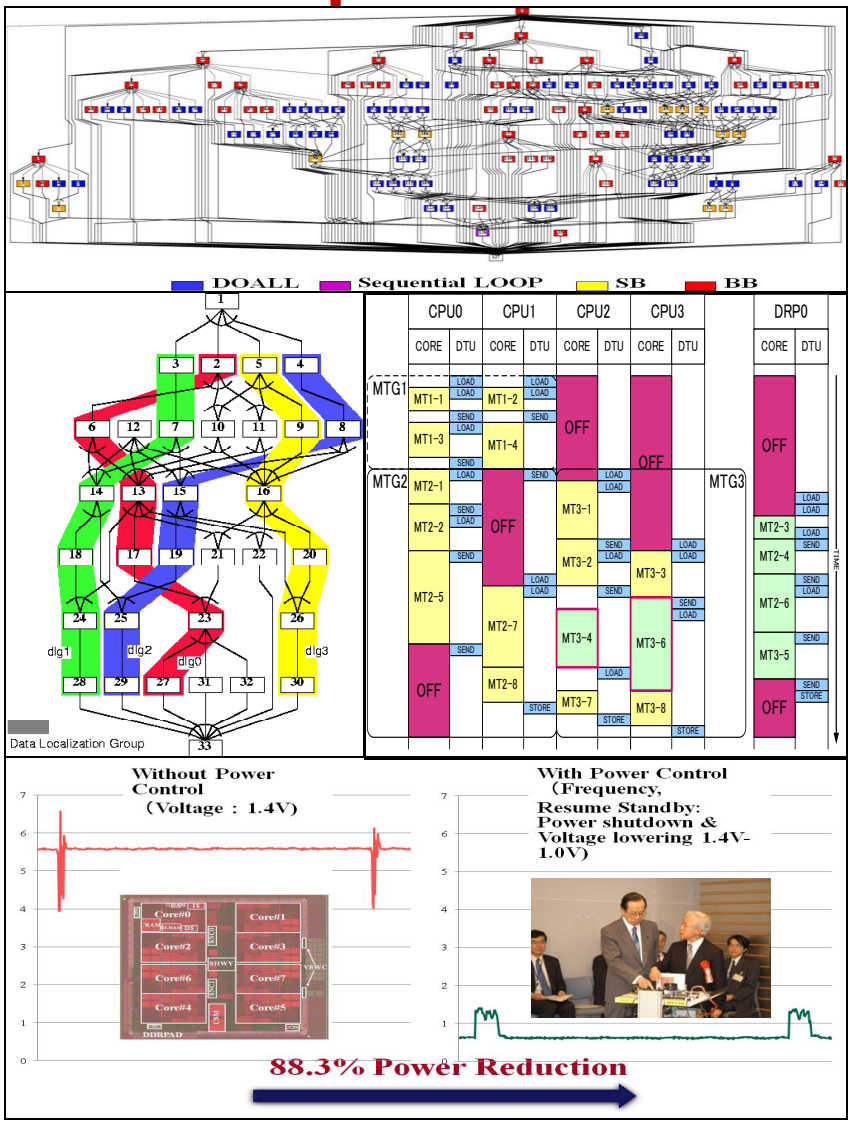
Data Transfer Overlapping(2016 partially)

Data transfer overlapping using Data Transfer Controllers (DMAs)

Power Reduction

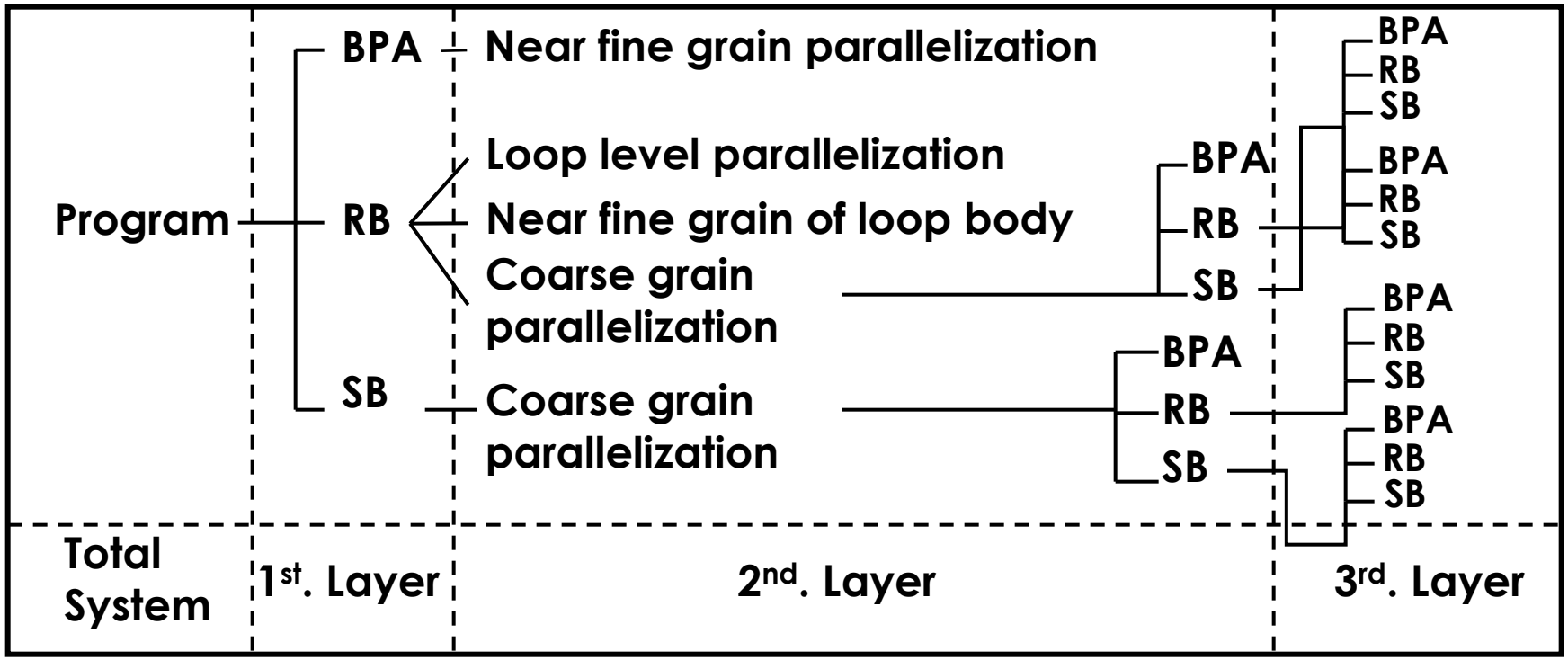
(2005 for Multicore, 2011 Multi-processes, 2013 on ARM)

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

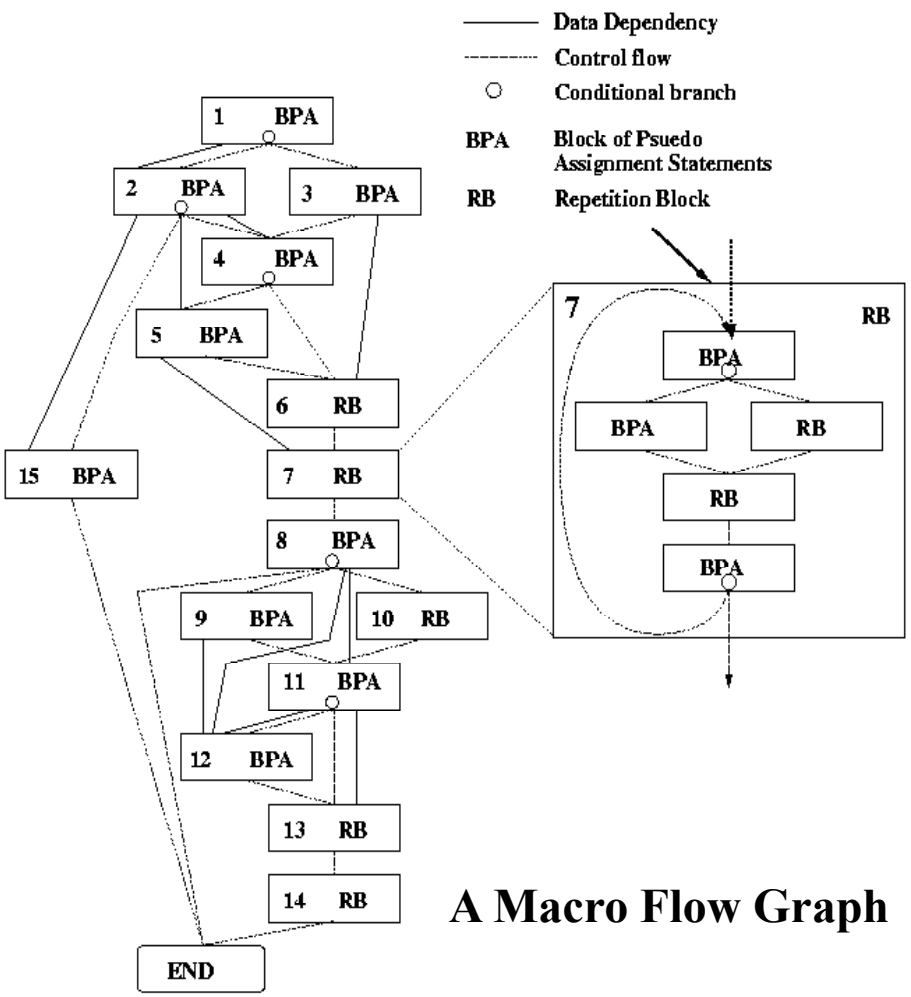


Generation of Coarse Grain Tasks

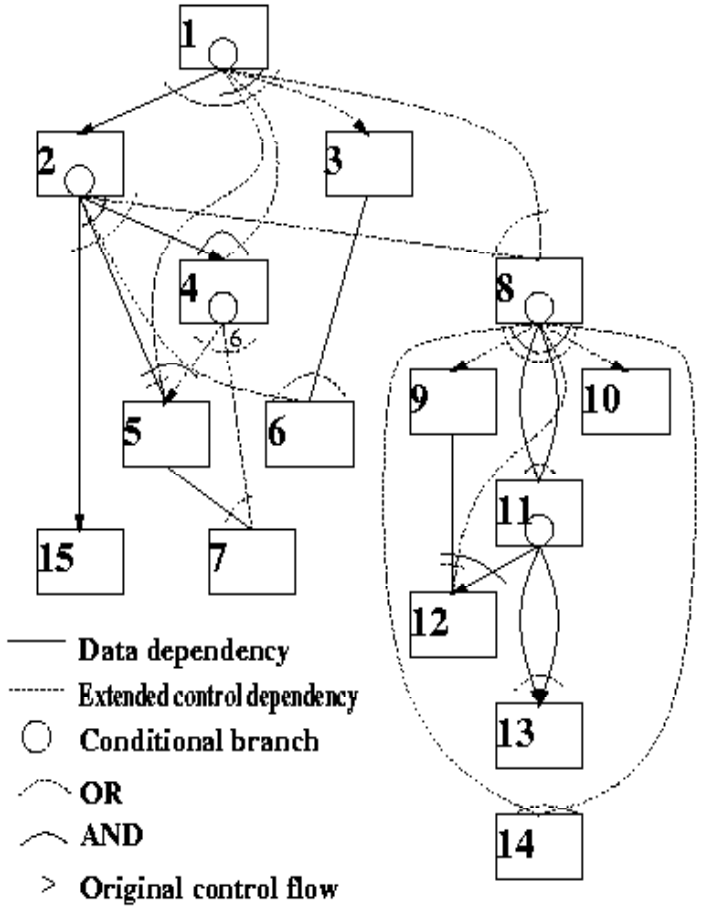
- **Macro-tasks (MTs)**
  - Block of Pseudo Assignments (BPA): Basic Block (BB)
  - Repetition Block (RB) : natural loop
  - Subroutine Block (SB): subroutine



Earliest Executable Condition Analysis for Coarse Grain Tasks (Macro-tasks)



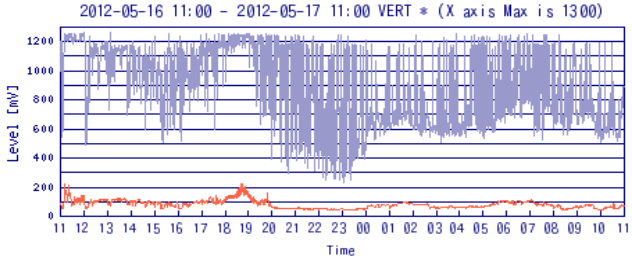
A Macro Flow Graph

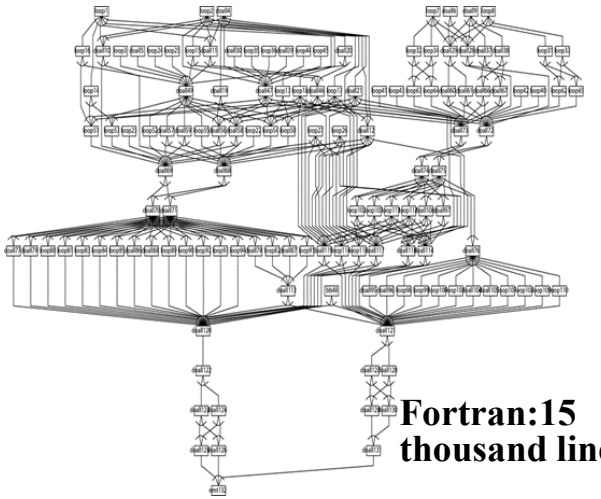


A Macro Task Graph

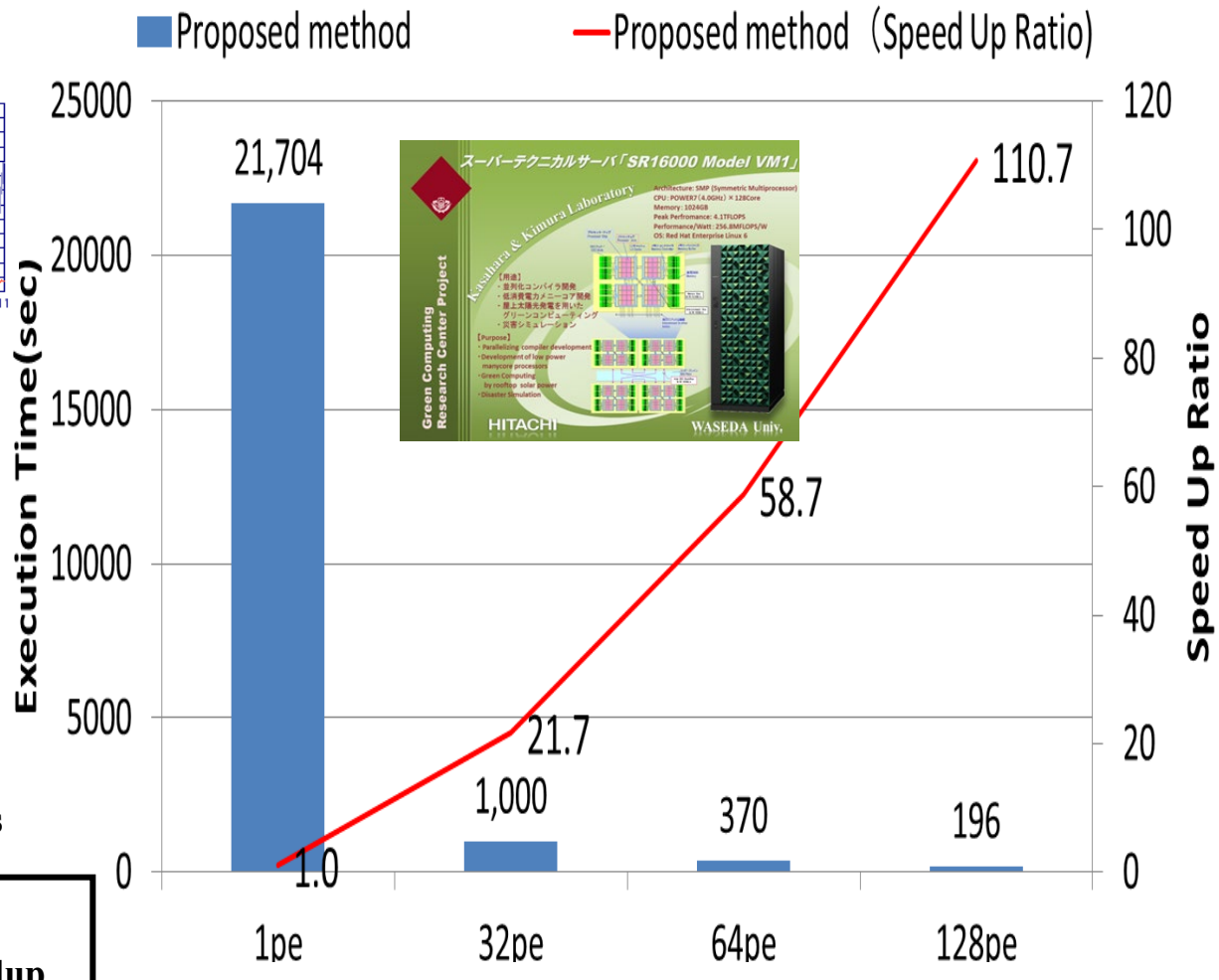
110 Times Speedup against the Sequential Processing for GMS Earthquake Wave Propagation Simulation on Hitachi SR16000

（Power7 Based 128 Core Linux SMP） (LCPC2015)



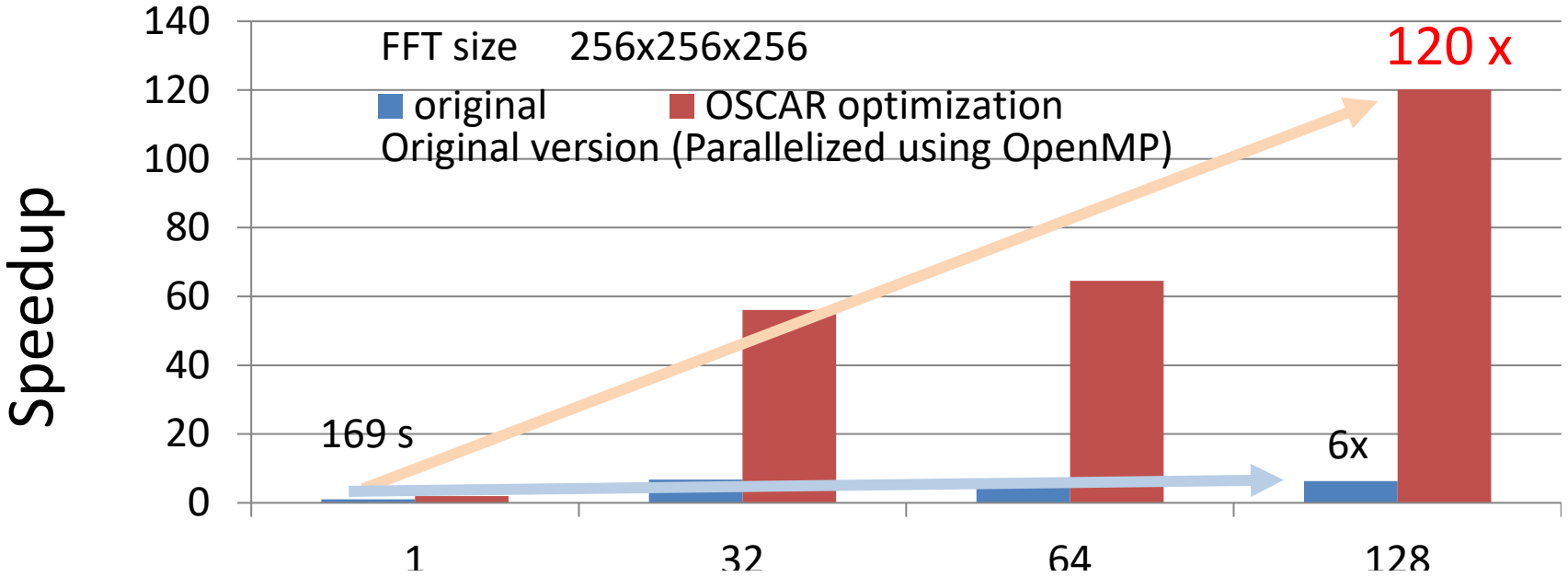


Fortran:15 thousand lines



First touch for distributed shared memory and cache optimization over loops are important for scalable speedup

Parallelization of 3D-FFT for New Magnetic Material Computation on Hitachi SR16000 Power7 CC-Numa Server



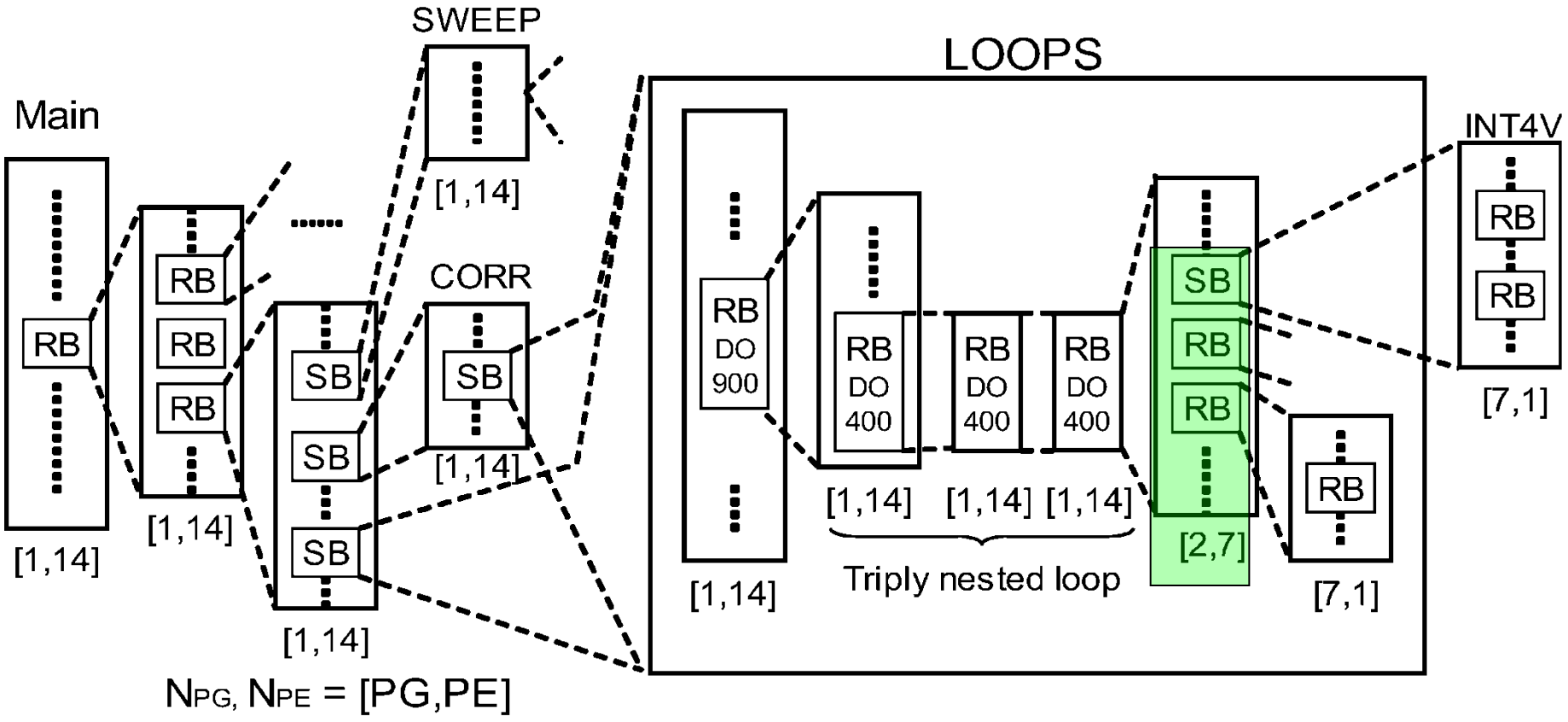
OSCAR optimization

- reducing number of data transpose with interchange, code motion and loop fusion

Automatic processor assignment in 103.su2cor

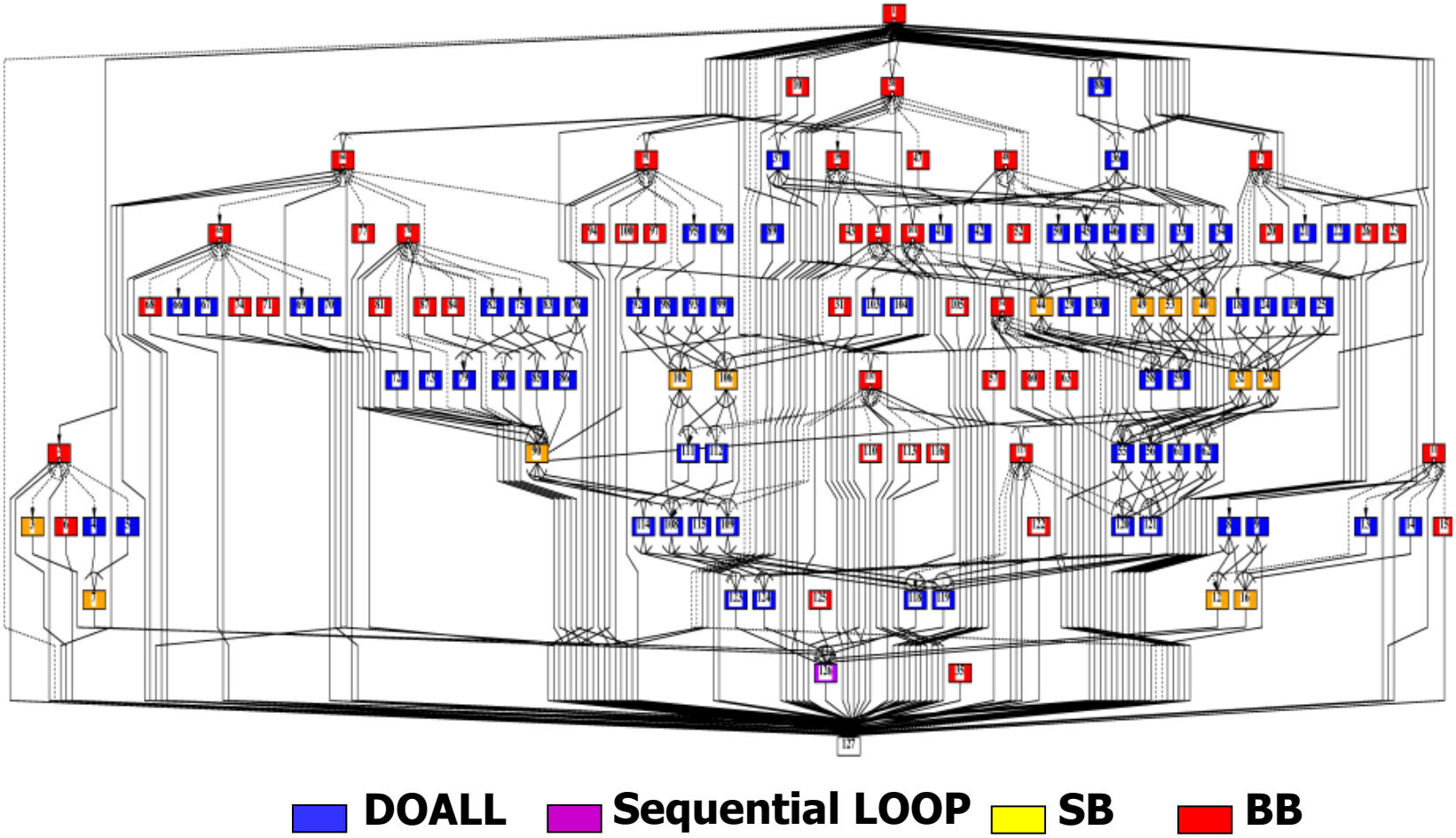
- **Using 14 processors**

Coarse grain parallelization within DO400



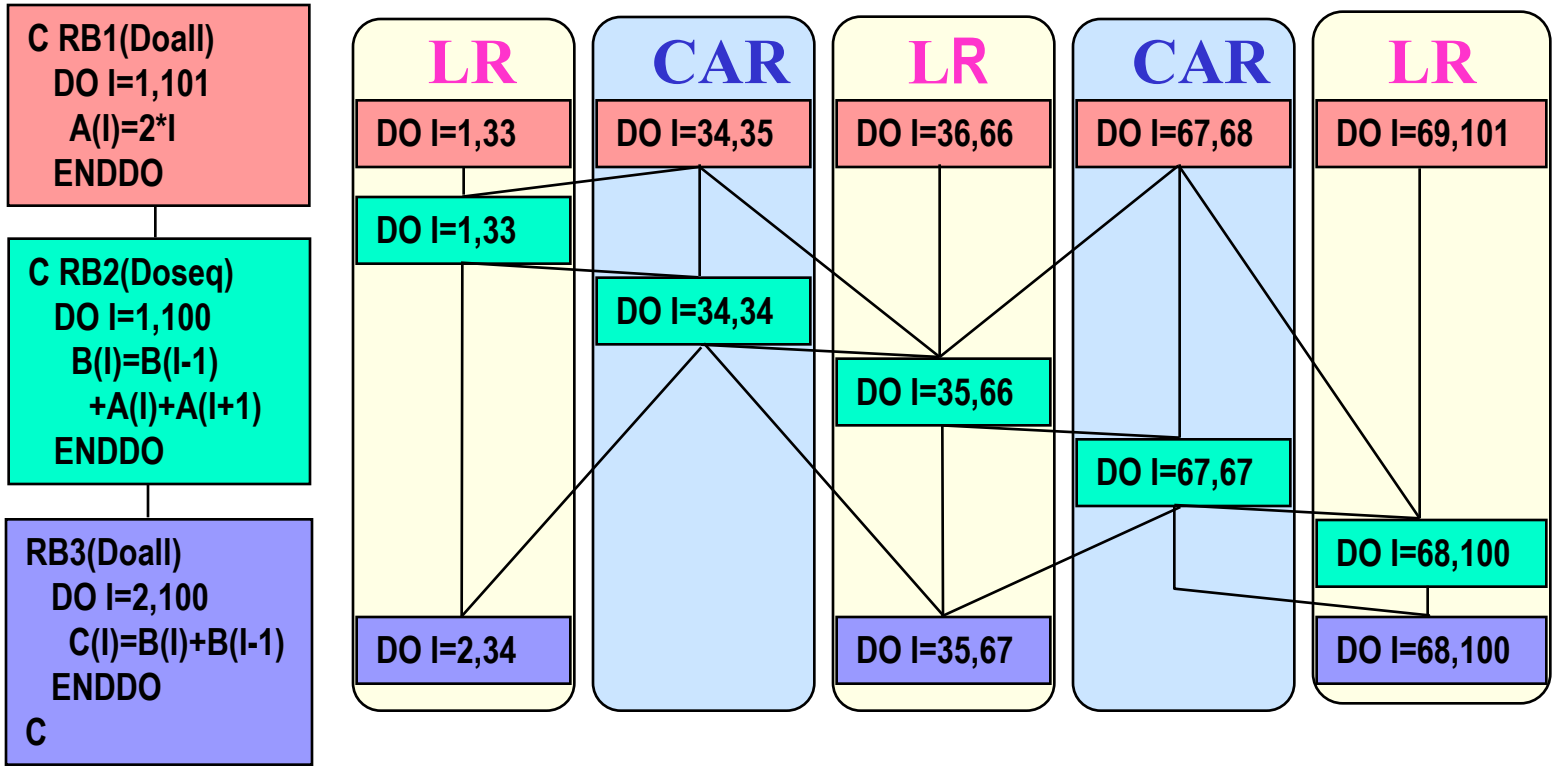
MTG of Su2cor-LOOPS-DO400

- **Coarse grain parallelism PARA_ALD = 4.3**

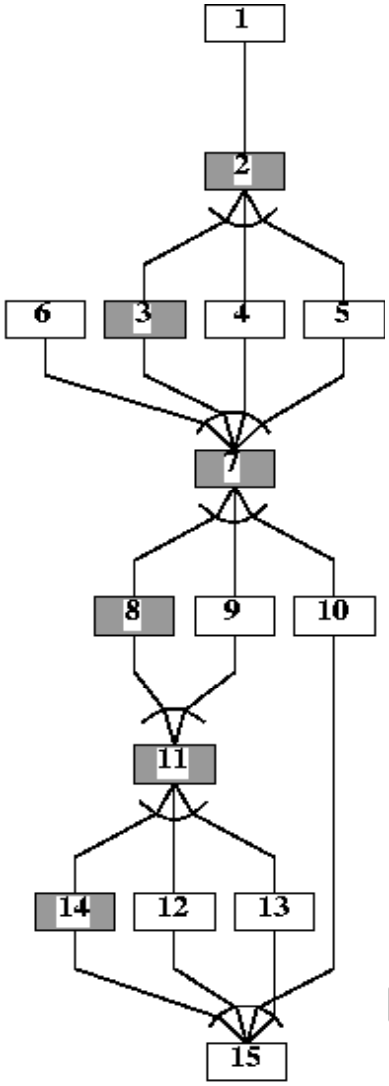


Data-Localization: Loop Aligned Decomposition

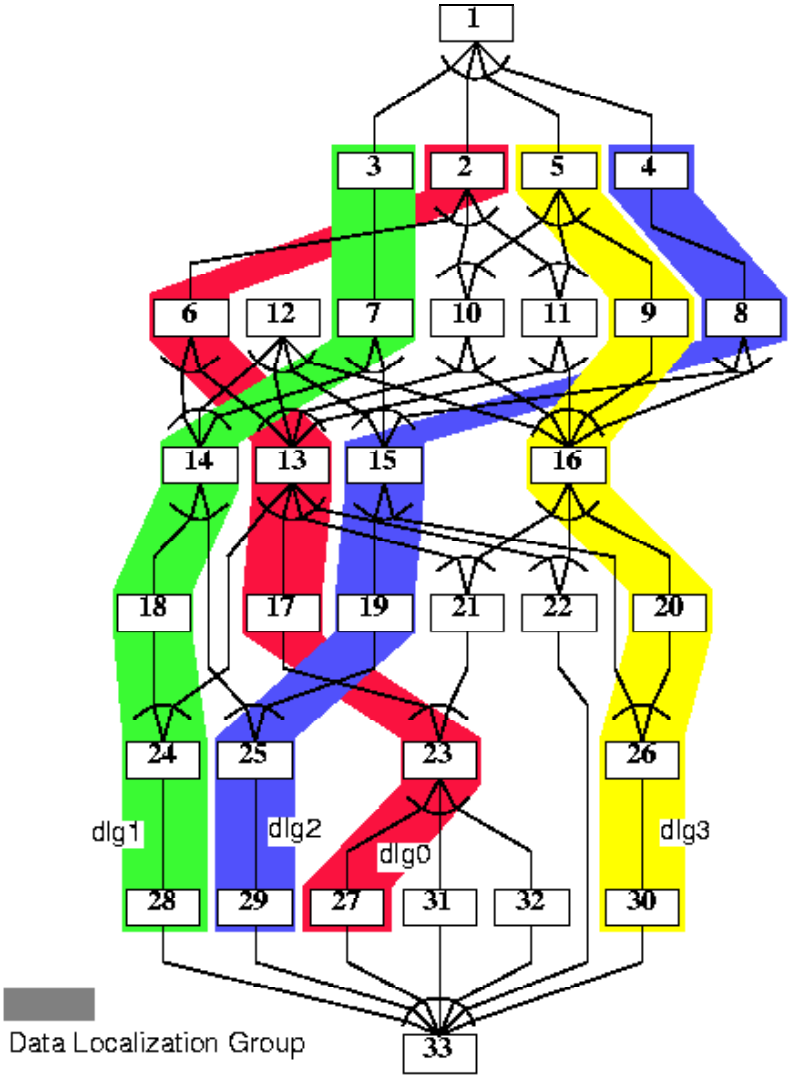
- **Decompose multiple loop (Doall and Seq) into CARs and LRs considering inter-loop data dependence.**
  - **Most data in LR can be passed through LM.**
  - **LR: Localizable Region, CAR: Commonly Accessed Region**



Data Localization



MTG



MTG after Division

A schedule for two processors

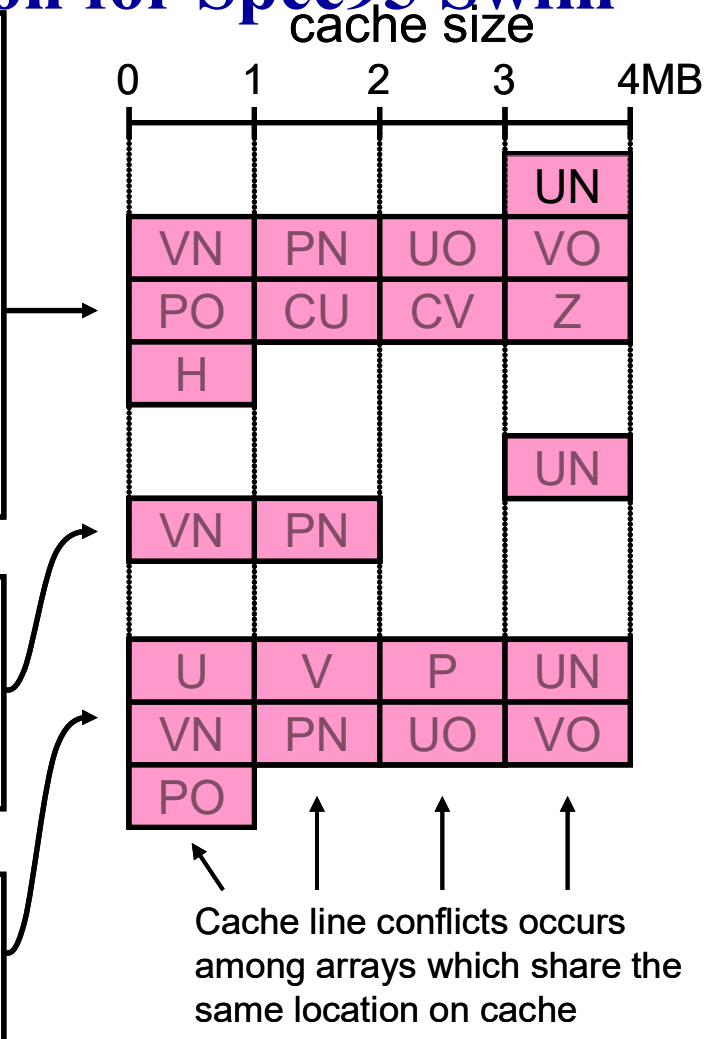
An Example of Data Localization for Spec95 Swim

```
      DO 200 J=1,N
      DO 200 I=1,M
         UNEW(I+1,J) = UOLD(I+1,J)+
     1    TDTS8*(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)-TDTS8*(Z(I+1,J+1)+Z(I,J+1))
     1      *(CU(I+1,J+1)+CU(I,J+1)+CU(I,J)+CU(I+1,J))
     2      -TDTSDY*(H(I,J+1)-H(I,J))
         PNEW(I,J) = POLD(I,J)-TDTSDX*(CU(I+1,J)-CU(I,J))
     1      -TDTSDY*(CV(I,J+1)-CV(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



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

Data Layout for Removing Line Conflict Misses by Array Dimension Padding

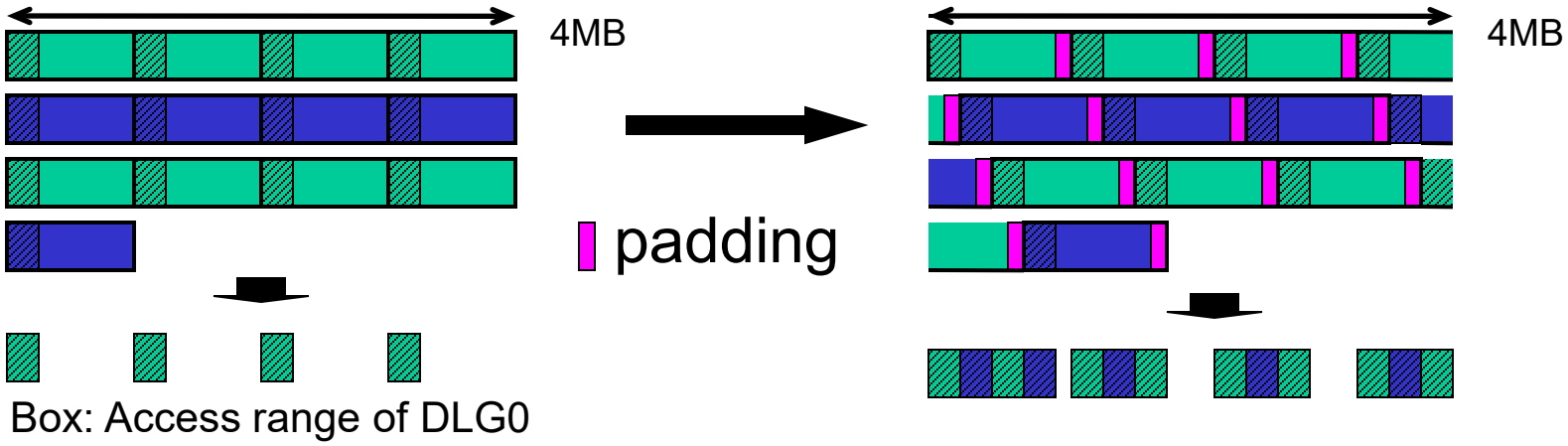
Declaration part of arrays in spec95 swim

before padding

```
PARAMETER (N1=513, N2=513)

COMMON  U(N1,N2), V(N1,N2), P(N1,N2),
*       UNEW(N1,N2), VNEW(N1,N2),
1       PNEW(N1,N2), UOLD(N1,N2),
*       VOLD(N1,N2), POLD(N1,N2),
2       CU(N1,N2), CV(N1,N2),
*       Z(N1,N2), H(N1,N2)
```

after padding

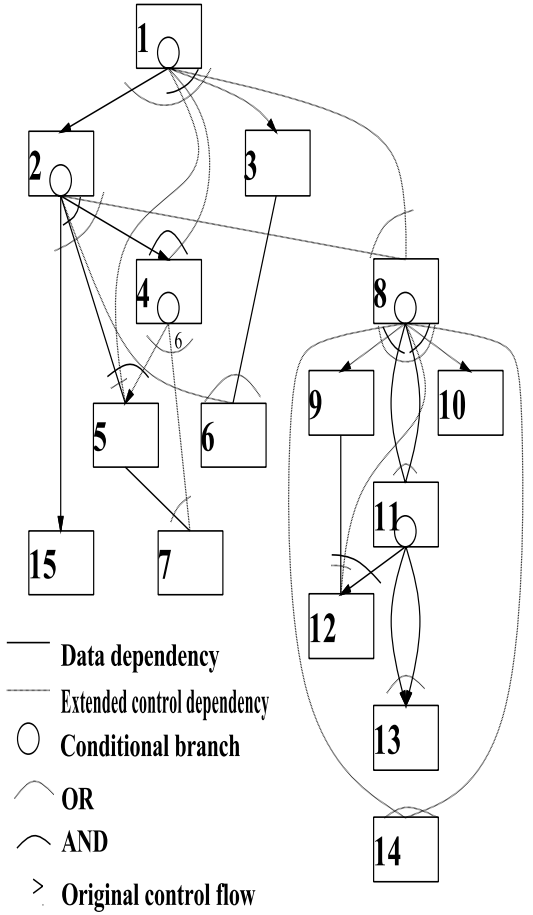
```
PARAMETER (N1=513, N2=544)

COMMON  U(N1,N2), V(N1,N2), P(N1,N2),
*       UNEW(N1,N2), VNEW(N1,N2),
1       PNEW(N1,N2), UOLD(N1,N2),
*       VOLD(N1,N2), POLD(N1,N2),
2       CU(N1,N2), CV(N1,N2),
*       Z(N1,N2), H(N1,N2)
```



Box: Access range of DLG0

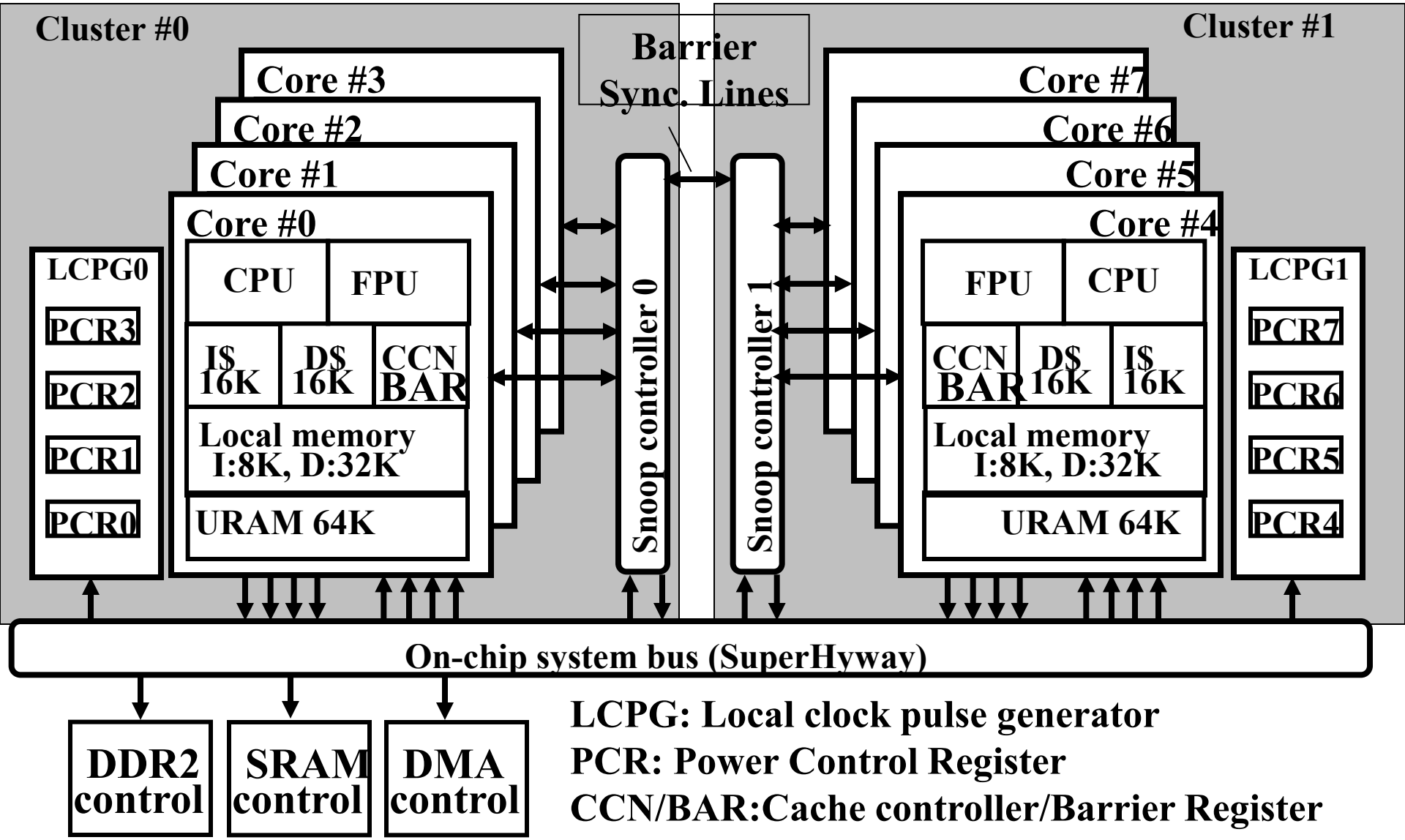
Software Coherence Control Method on OSCAR Parallelizing Compiler

- Coarse grain task parallelization with earliest condition analysis (control and data dependency analysis to detect parallelism among coarse grain tasks).
- OSCAR compiler automatically controls coherence using following simple program restructuring methods:
  - To cope with stale data problems:
    - **Data synchronization by compilers**
  - To cope with false sharing problem:
    - **Data Alignment**
    - **Array Padding**
    - **Non-cacheable Buffer**



MTG generated by earliest executable condition analysis

8 Core RP2 Chip Block Diagram



LCPG: Local clock pulse generator

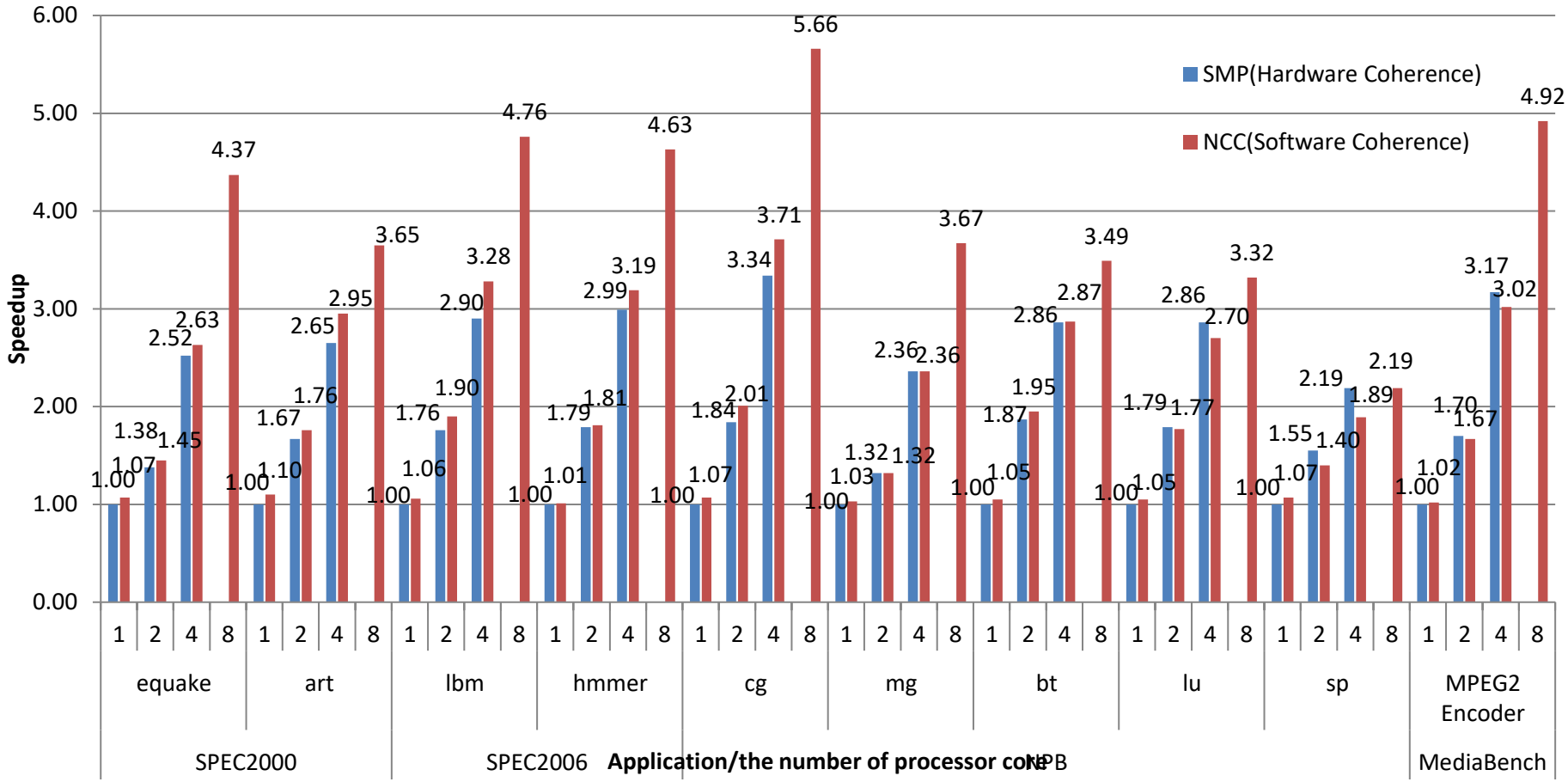
PCR: Power Control Register

CCN/BAR:Cache controller/Barrier Register

URAM: User RAM (Distributed Shared Memory)

Automatic Software Coherent Control for Manycores

Performance of Software Coherence Control by OSCAR Compiler on 8-core RP2

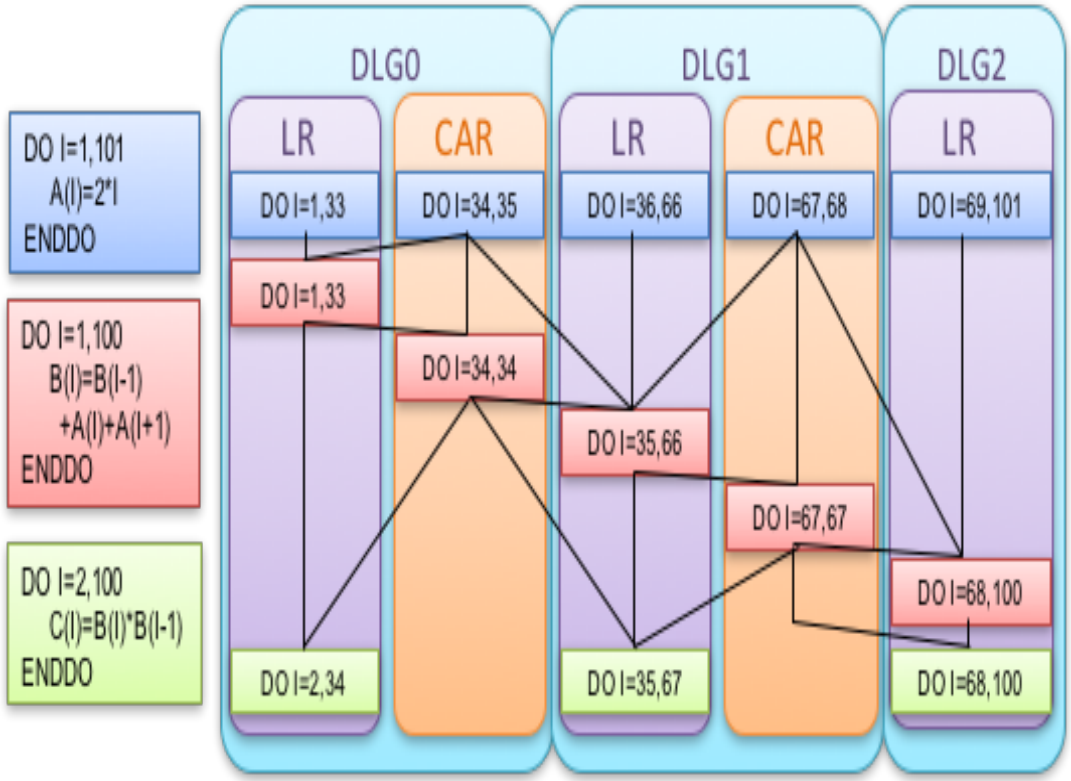


Automatic Local Memory Management

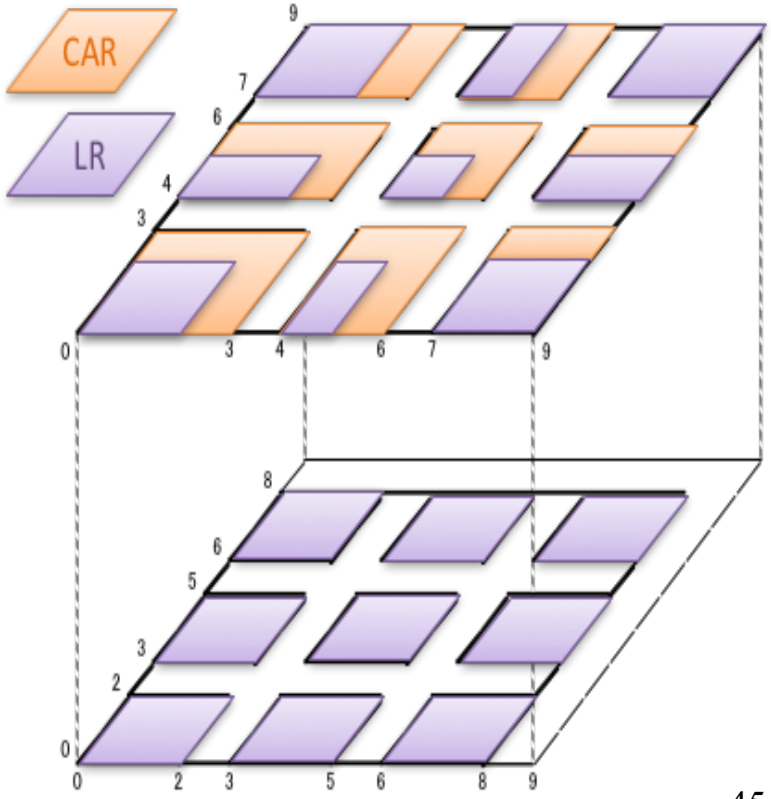
Data Localization: Loop Aligned Decomposition

- **Decomposed loop into LRs and CARs**
  - **LR (Localizable Region): Data can be passed through LDM**
  - **CAR (Commonly Accessed Region): Data transfers are required among processors**

Single dimension Decomposition

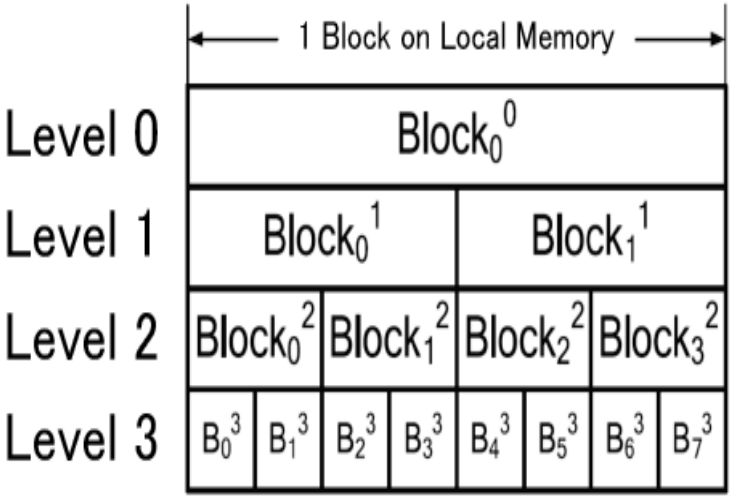


Multi-dimension Decomposition



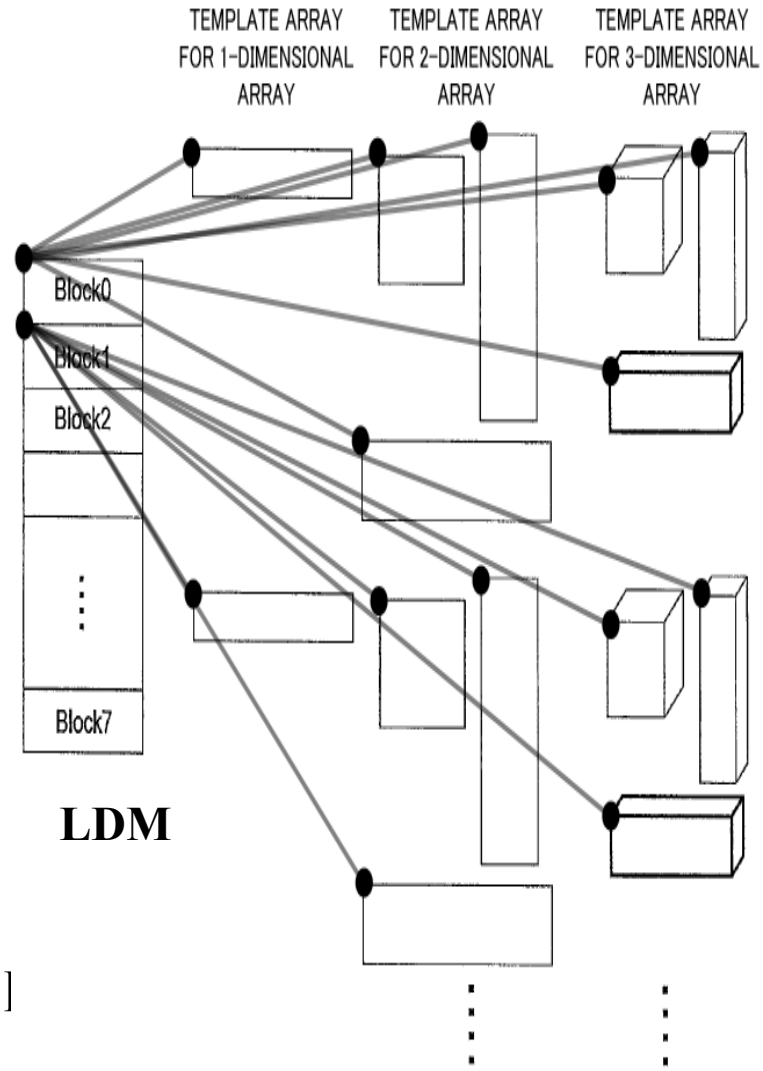
Adjustable Blocks

- Handling a suitable block size for each application
  - different from a fixed block size in cache
  - each block can be divided into smaller blocks with intege $Block_{Number}^{Level}$ small arrays and scalar



Multi-dimensional Template Arrays for Improving Readability

- a mapping technique for arrays with varying dimensions
  - each block on LDM corresponds to multiple empty arrays with varying dimensions
  - these arrays have an additional dimension to store the corresponding block number
    - TA[Block#][] for single dimension
    - TA[Block#][][] for double dimension
    - TA[Block#][][][] for triple dimension
    - ...
- LDM are represented as a one dimensional array
  - without Template Arrays, multi-dimensional arrays have complex index calculations
    - A[i][j][k] -> TA[offset + i' * L + j' * M + k']
  - Template Arrays provide readability
    - A[i][j][k] -> TA[Block#][i'][j'][k']

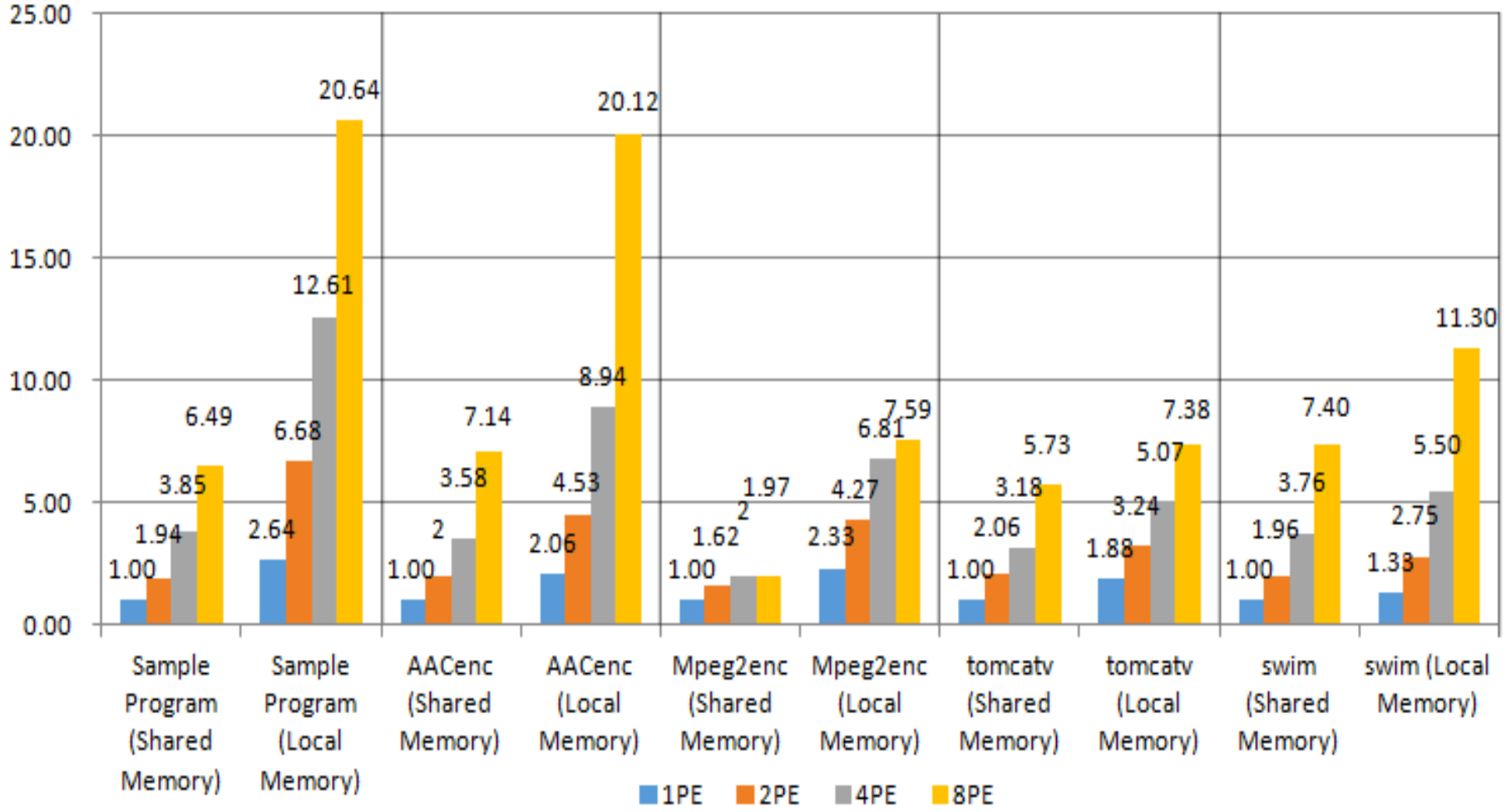


LDM

Block Replacement Policy

- Compiler Control Memory block Replacement
  - using live, dead and reuse information of each variable from the scheduled result
  - different from LRU in cache that does not use data dependence information
- Block Eviction Priority Policy
  1. (Dead) Variables that will not be accessed later in the program
  2. Variables that are accessed only by other processor cores
  3. Variables that will be later accessed by the current processor core
  4. Variables that will immediately be accessed by the current processor core

Speedups by the Proposed Local Memory Management Compared with Utilizing Shared Memory on Benchmarks Application using RP2



20.12 times speedup for 8cores execution using local memory against sequential execution using off-chip shared memory of RP2 for the AACenc

Multicore Program Development Using OSCAR API V2.0

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

Homogeneous

Hetero

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

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

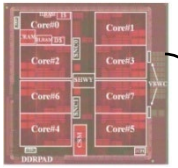
Accelerator 2
Code

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

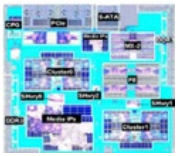
Low Power Heterogeneous Multicore Code Generation
API Analyzer (Available from Waseda) | Existing sequential compiler

Server Code Generation
OpenMP Compiler

Generation of parallel machine codes using sequential compilers



Homegeneous Multicore s from Vendor A (SMP servers)



Heterogeneous Multicores from Vendor B

Shred memory servers

Executable on various multicores

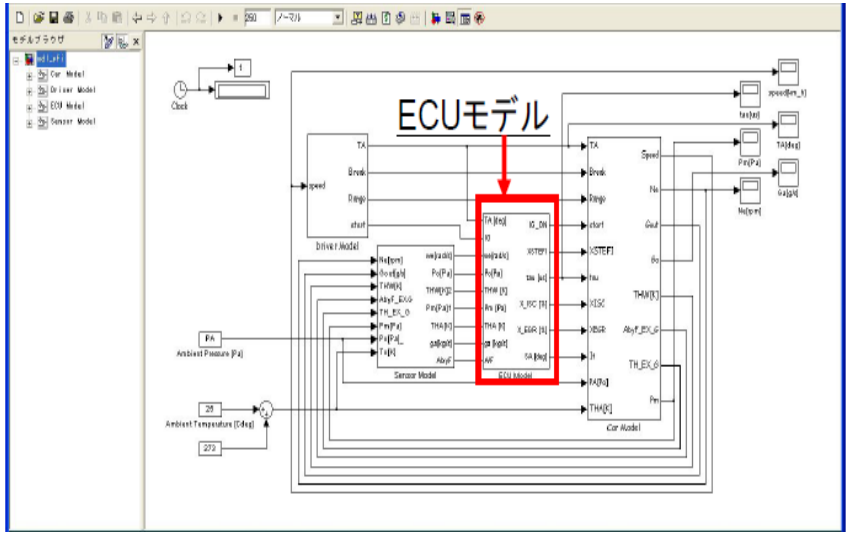
Hitachi, Renesas, NEC, Fujitsu, Toshiba, Denso, Olympus, Mitsubishi, Esol, Cats, Gaio, 3 univ.

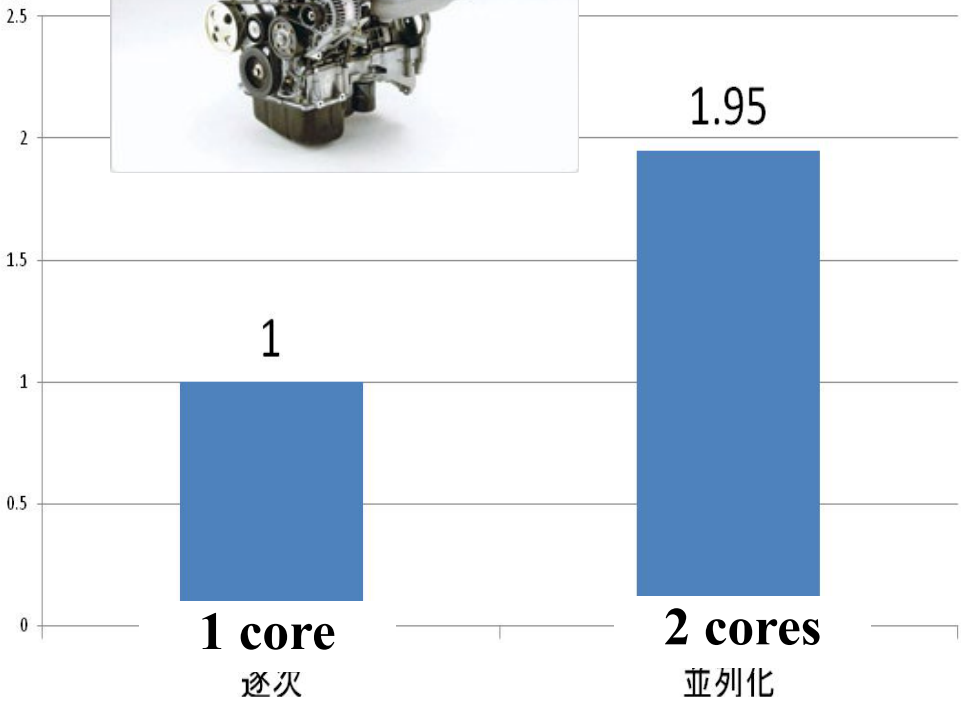
OSCAR: Optimally Scheduled Advanced Multiprocessor
API : Application Program Interface

Engine Control by multicore with Denso

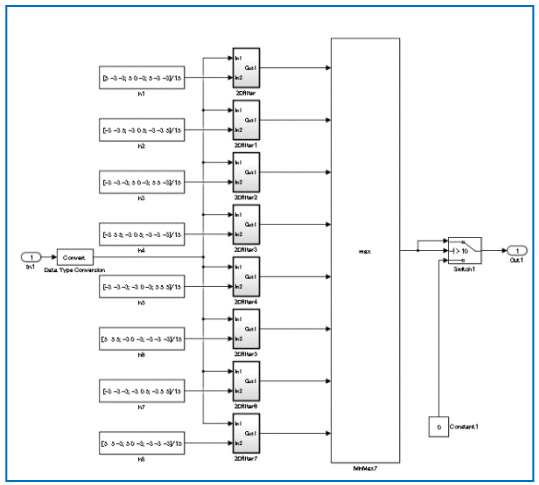
Though so far parallel processing of the engine control on multicore has been very difficult, Denso and Waseda succeeded 1.95 times speedup on 2core V850 multicore processor.

- Hard real-time automobile engine control by multicore using local memories
- Millions of lines C codes consisting conditional branches and basic blocks





OSCAR Compile Flow for Simulink Applications



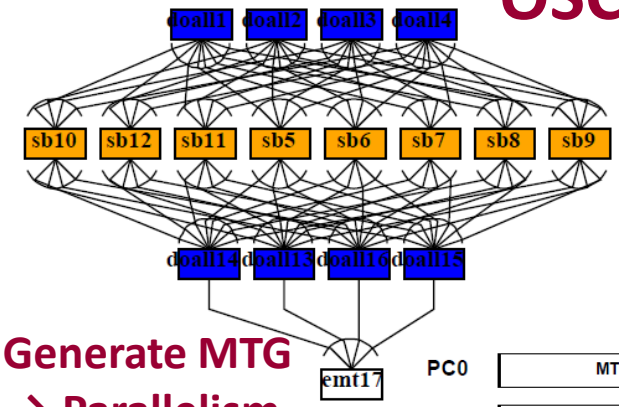
Simulink model

Generate C code
using Embedded Coder

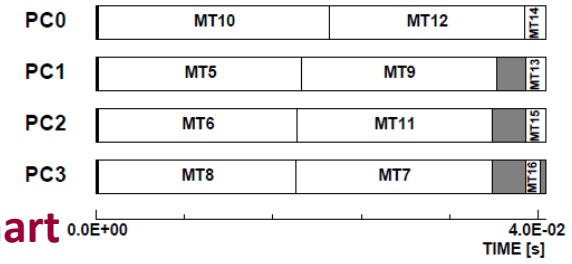
```
/* Model step function */
void VesselExtraction_step(void)
{
  int32_T i;
  real_T u0;

  /* DataTypeConversion: '<S1>/Data Type Conversion' incorporates:
   *  Inport: '<Root>/In1'
   */
  for (i = 0; i < 16384; i++) {
    VesselExtraction_B.DataTypeConversion[i] = VesselExtraction_U.In1[i];
  }

  /* End of DataTypeConversion: '<S1>/Data Type Conversion' */

  /* Outputs for Atomic SubSystem: '<S1>/2Dfilter' */

  /* Constant: '<S1>/h1' */
  VesselExtraction_Dfilter(VesselExtraction_B.DataTypeConversion,
    VesselExtraction_P.h1_Value, &VesselExtraction_B.Dfilter,
    (P_Dfilter_VesselExtraction_T *)&VesselExtraction_P.Dfilter);

  /* End of Outputs for SubSystem: '<S1>/2Dfilter' */

  /* Outputs for Atomic SubSystem: '<S1>/2Dfilter1' */

  /* Constant: '<S1>/h2' */
  VesselExtraction_Dfilter(VesselExtraction_B.DataTypeConversion,
    VesselExtraction_P.h2_Value, &VesselExtraction_B.Dfilter1,
    (P_Dfilter_VesselExtraction_T *)&VesselExtraction_P.Dfilter1);
```

C code

OSCAR Compiler



(1) Generate MTG
→ Parallelism



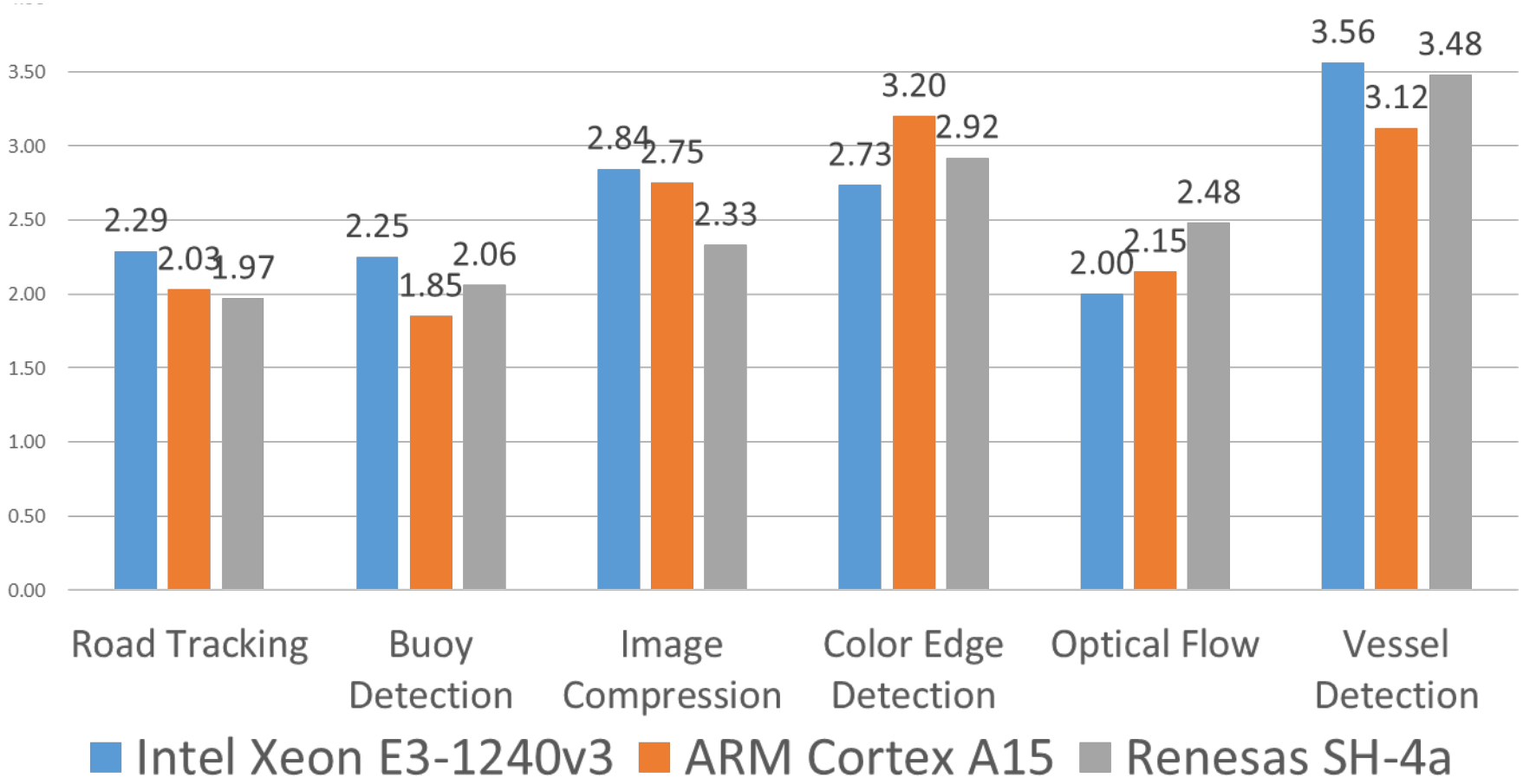
(2) Generate gantt chart
→ Scheduling in a multicore

```
void VesselExtraction_step ( )
{
    int thr1 ;
    int thr2 ;
    int thr3 ;
    {
        oscar_thread_create ( & thr1 ,
            thread_function_001 , (void*)1 ) ;
        oscar_thread_create ( & thr2 ,
            thread_function_002 , (void*)2 ) ;
        oscar_thread_create ( & thr3 ,
            thread_function_003 , (void*)3 ) ;

            VesselExtraction_step_PE0 ( ) ;

        oscar_thread_join ( thr1 ) ;
        oscar_thread_join ( thr2 ) ;
        oscar_thread_join ( thr3 ) ;
    }
}

void thread_function_001 ( void )
{
    VesselExtraction_step_PE1 ( ) ;
}
```

(3) Generate parallelized C code
using the OSCAR API
→ Multiplatform execution
(Intel, ARM and SH etc)

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
Buoy Detection : http://www.mathworks.co.jp/matlabcentral/fileexchange/44706-buoy-detection-using-simulink
Color Edge Detection : http://www.mathworks.co.jp/matlabcentral/fileexchange/28114-fast-edges-of-a-color-image--actual-color--not-converting-to-grayscale-/
Vessel Detection : http://www.mathworks.co.jp/matlabcentral/fileexchange/24990-retinal-blood-vessel-extraction/

OSCAR API Ver. 2.0 for Homogeneous/Heterogeneous Multicores and Manycores (LCPC2009Homo, 2010 Hetero)

List of Directives (22 directives)

- Parallel Execution API
  - parallel sections (*)
  - flush (*)
  - critical (*)
  - execution
- Memoay Mapping API
  - threadprivate (*)
  - distributedshared
  - onchipshared
- Synchronization API
  - groupbarrier
- Data Transfer API
  - dma_transfer
  - dma_contiguous_parameter
  - dma_stride_parameter
  - dma_flag_check
  - dma_flag_send

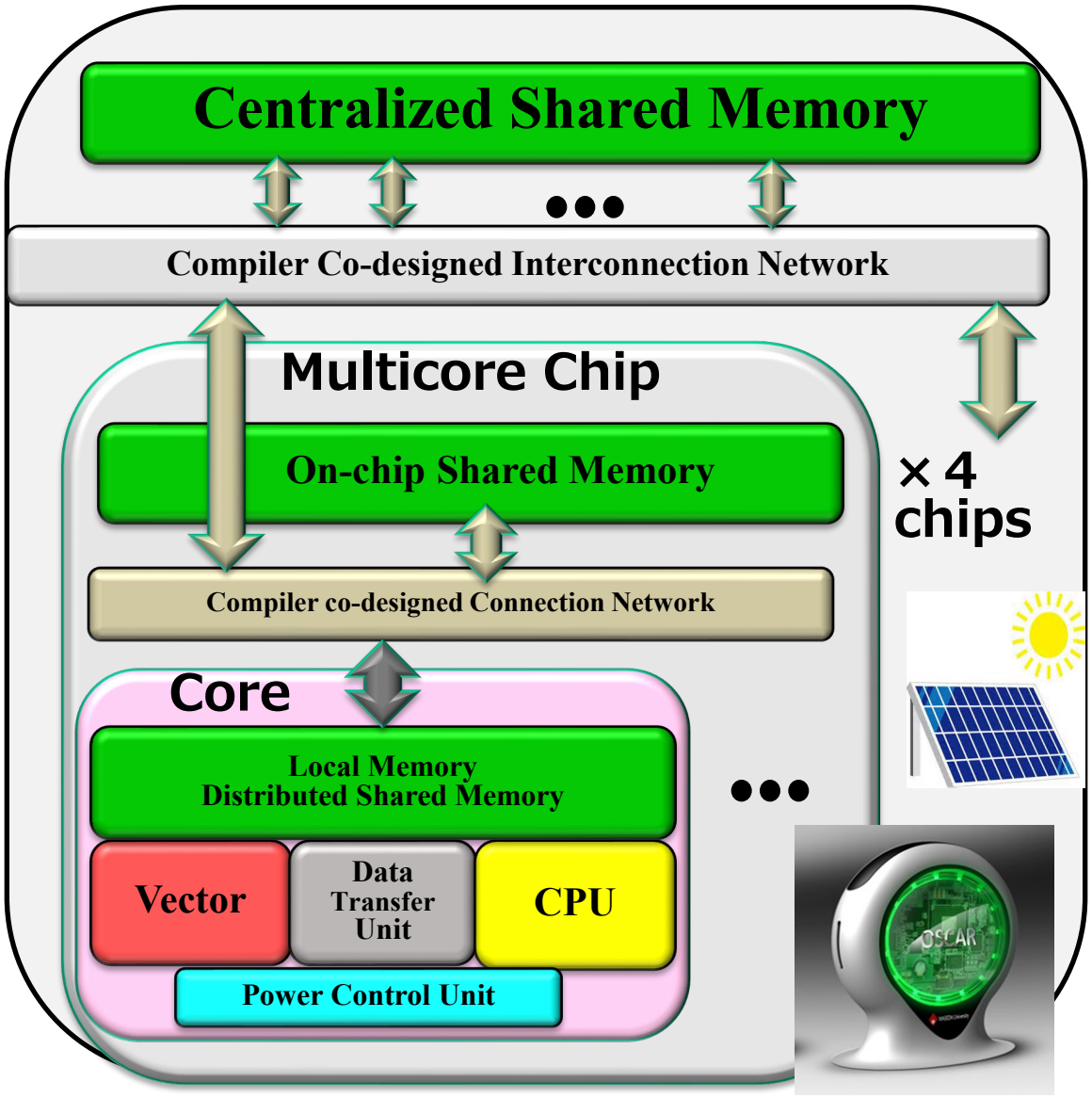
(* from OpenMP)

- Power Control API
  - fvcontrol
  - get_fvstatus
- Timer API
  - get_current_time
- Accelerator
  - accelerator_task_entry
- Cache Control
  - cache_writeback
  - cache_selfinvalidate
  - complete_memop
  - noncacheable
  - aligncache

2 hint directives for OSCAR compiler
- accelerator_task
- oscar_comment

from V2.0

OSCAR Vector Multicore and Compiler for Embedded to Severs with OSCAR Technology



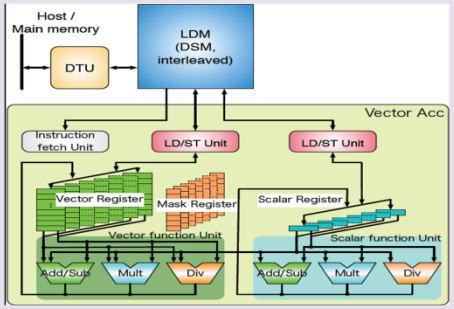
Target:

- **Solar Powered**
- **Compiler power reduction.**
- **Fully automatic parallelization and vectorization including local memory management and data transfer.**

Vector Accelerator

Features

- Attachable for any CPUs (Intel, ARM, IBM)
- Data driven initiation by sync flags



Function Units [tentative]

- Vector Function Unit
  - 8 double precision ops/clock
  - 64 characters ops/clock
  - Variable vector register length
  - Chaining LD/ST & Vector pipes
- Scalar Function Unit

Registers[tentative]

- Vector Register 256Bytes/entry, 32entry
- Scalar Register 8Bytes/entry
- Floating Point Register 8Bytes/entry
- Mask Register 32Bytes/entry

Future Multicore Products with Automatic Parallelizing Compiler

Next Generation Automobiles

- Safer, more comfortable, energy efficient, environment friendly
- Cameras, radar, car2car communication, internet information integrated brake, steering, engine, moter control

Smart phones

-From everyday recharging to less than once a week
- Solar powered operation in emergency condition
- Keep health

Advanced medical systems

Cancer treatment,
Drinkable inner camera

- Emergency solar powered
- No cooling fun, No dust , clean usable inside OP room

Personal / Regional Supercomputers

Solar powered with more than 100 times power efficient : FLOPS/W

- Regional Disaster Simulators saving lives from tornadoes, localized heavy rain, fires with earth quakes

Summary

- Waseda University Green Computing Systems R&D Center supported by METI has been researching on low-power high performance Green Multicore hardware, software and application with industry including Hitachi, Fujitsu, NEC, Renesas, Denso, Toyota, Olympus and OSCAR Technology.
- OSCAR Automatic Parallelizing and Power Reducing Compiler has succeeded speedup and/or power reduction of scientific applications including "Earthquake Wave Propagation", medical applications including "Cancer Treatment Using Carbon Ion", and "Drinkable Inner Camera", industry application including "Automobile Engine Control", "Smartphone", and "Wireless communication Base Band Processing" on various multicores from different vendors including Intel, ARM, IBM, AMD, Qualcomm, Freescale, Renesas and Fujitsu.
- In automatic parallelization, 110 times speedup for "Earthquake Wave Propagation Simulation" on 128 cores of IBM Power 7 against 1 core, 55 times speedup for "Carbon Ion Radiotherapy Cancer Treatment" on 64cores IBM Power7, 1.95 times for "Automobile Engine Control" on Renesas 2 cores using SH4A or V850, 55 times for "JPEG-XR Encoding for Capsule Inner Cameras" on Tilera 64 cores Tile64 manycore.
  - The compiler will be available on market from OSCAR Technology.
- In automatic power reduction, consumed powers for real-time multi-media applications like Human face detection, H.264, mpeg2 and optical flow were reduced to 1/2 or 1/3 using 3 cores of ARM Cortex A9 and Intel Haswell and 1/4 using Renesas SH4A 8 cores against ordinary single core execution.
- Local memory management for automobiles and software coherent control have been patented and already realized by OSCAR compiler.