Green Multicore Computing and Industry Collaboration

Hironori Kasahara

IEEE Computer Society President Elect 2017, President 2018

Professor, Dept. of Computer Science & Engineering

Director, Advanced Multicore Processor Research Institute

Waseda University, Tokyo, Japan

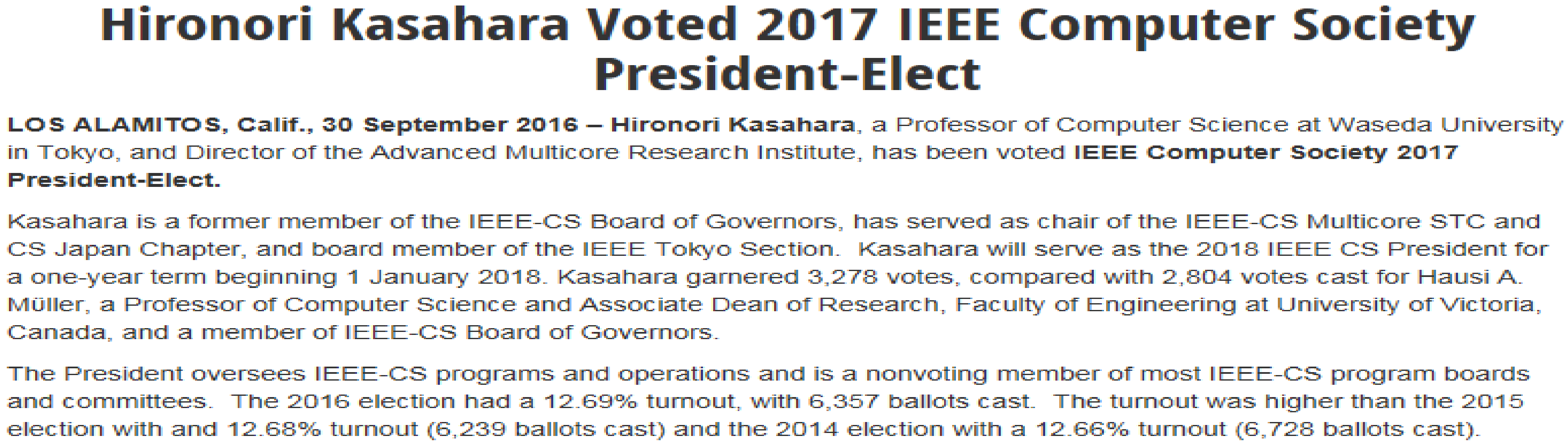
URL: http://www.kasahara.cs.waseda.ac.jp/

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 Prof., Dept. EECE, Waseda U. Now Dept. Comput. Sci. & Eng.
1989-90 Research Scholar: U. of Illinois, Urbana-Champaign, Center for Supercomputing R&D
2017 Member: Science Council of Japan, Engineering Academy of Japan, IEEE Eta-Kappa-Nu

1987 IFAC World Congress Yo ung Author Prize
1997 IPSJ Sakai Special Research Award
2005 STARC Academia-Industry Research Award
2008 LSI of the Year Second Prize
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

Reviewed Papers: 214, Invited Talks: 145, Published Unexamined Patent Application:59 (Japan, US, GB, China Granted Patents: 30), Articles in News Papers, Web News, Medias incl. TV etc.: 572

Committees in Societies and Government 245
IEEE Computer Society President 2018, BoG(2009-14), Multicore STC Chair (2012-), Japan Chair (2005-07), IPSJ Chair: HG for Mag. & J. Edit, Sig. on ARC.
【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, HPCI Promo., Next Gen. Supercomputer K) Committees, etc.

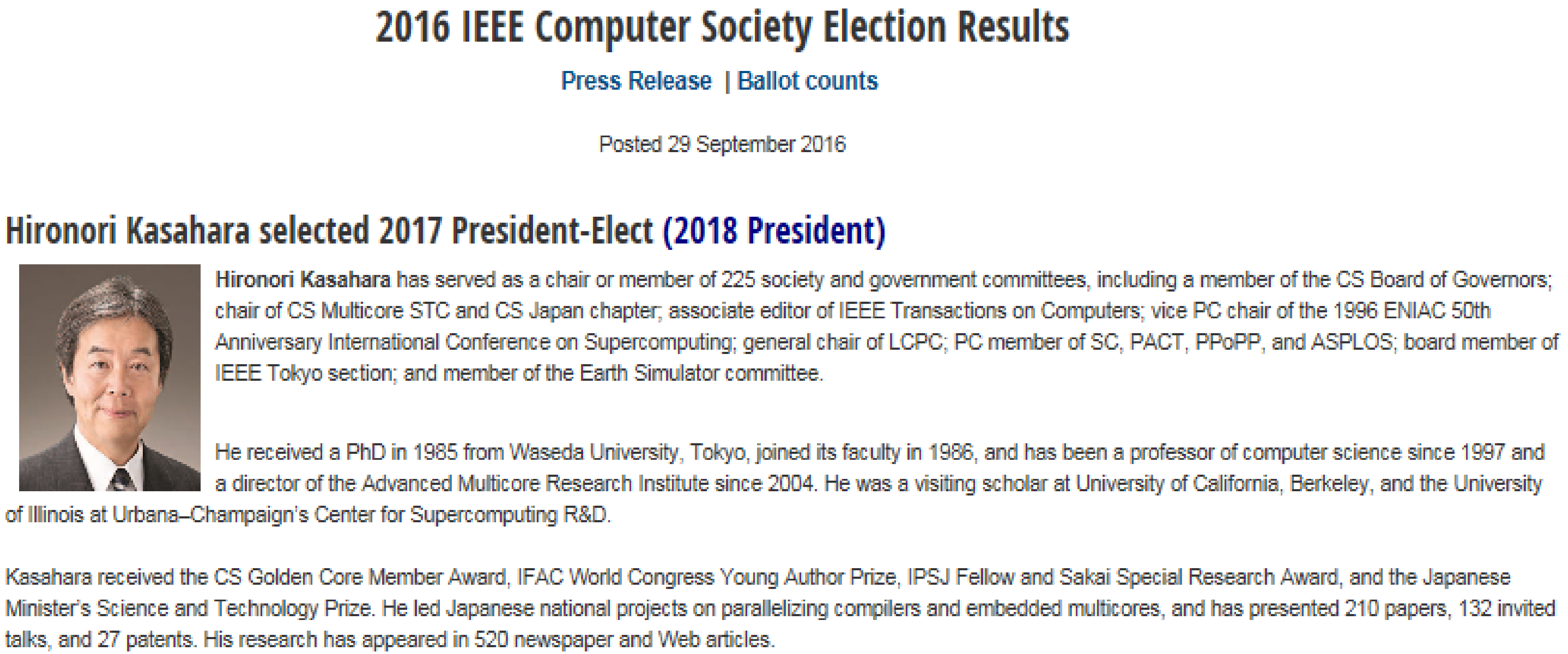


Hironori Kasahara Voted 2017 IEEE Computer Society President-Elect

LOS ALAMITOS, Calif., 30 September 2016 – Hironori Kasahara, a Professor of Computer Science at Waseda University in Tokyo, and Director of the Advanced Multicore Research Institute, has been voted **IEEE Computer Society 2017 President-Elect.**

Kasahara is a former member of the IEEE-CS Board of Governors, has served as chair of the IEEE-CS Multicore STC and CS Japan Chapter, and board member of the IEEE Tokyo Section. Kasahara will serve as the 2018 IEEE CS President for a one-year term beginning 1 January 2018. Kasahara garnered 3,278 votes, compared with 2,804 votes cast for Hausi A. Müller, a Professor of Computer Science and Associate Dean of Research, Faculty of Engineering at University of Victoria, Canada, and a member of IEEE-CS Board of Governors.

The President oversees IEEE-CS programs and operations and is a nonvoting member of most IEEE-CS program boards and committees. The 2016 election had a 12.69% turnout, with 6,357 ballots cast. The turnout was higher than the 2015 election with and 12.68% turnout (6,239 ballots cast) and the 2014 election with a 12.66% turnout (6,728 ballots cast).



2016 IEEE Computer Society Election Results

Press Release | Ballot counts

Posted 29 September 2016

Hironori Kasahara selected 2017 President-Elect (2018 President)

Hironori Kasahara has served as a chair or member of 225 society and government committees, including a member of the CS Board of Governors; chair of CS Multicore STC and CS Japan chapter; associate editor of IEEE Transactions on Computers; vice PC chair of the 1996 ENIAC 50th Anniversary International Conference on Supercomputing; general chair of LCPC; PC member of SC, PACT, PPoPP, and ASPLOS; board member of IEEE Tokyo section; and member of the Earth Simulator committee.

He received a PhD in 1985 from Waseda University, Tokyo, joined its faculty in 1986, and has been a professor of computer science since 1997 and a director of the Advanced Multicore Research Institute since 2004. He was a visiting scholar at University of California, Berkeley, and the University of Illinois at Urbana–Champaign's Center for Supercomputing R&D.

Kasahara received the CS Golden Core Member Award, IFAC World Congress Young Author Prize, IPSJ Fellow and Sakai Special Research Award, and the Japanese Minister's Science and Technology Prize. He led Japanese national projects on parallelizing compilers and embedded multicores, and has presented 210 papers, 132 invited talks, and 27 patents. His research has appeared in 520 newspaper and Web articles.

IEEE Computer Society BoG (Board of Governors) Feb.1, 2017

https://www.computer.org/web/cshistory/officers-2017

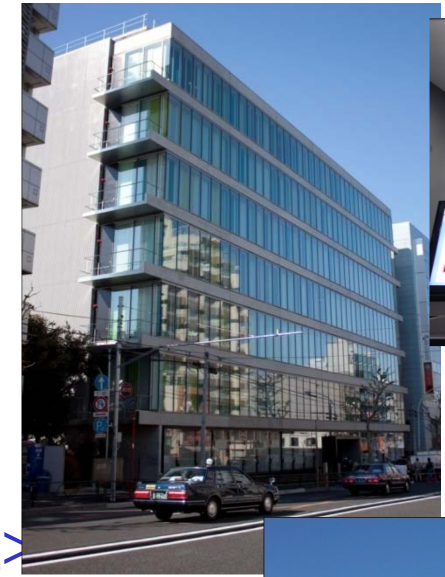
Toward 2018

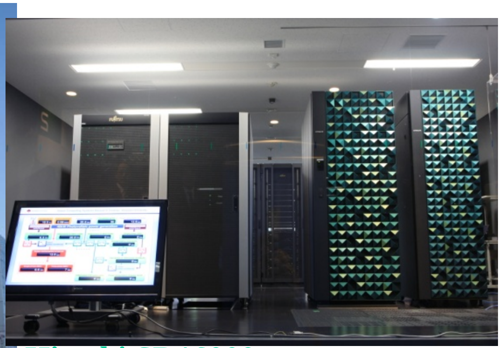
1. Refining content and services to further improve the satisfaction of CS members;
2. Considering an incentive for volunteers to further accelerate CS activities and promptly provide technical benefits for people around the globe;
   To express appreciation to volunteers:
   CS Point (Mileage) System: Annual & Life Time Honor,
   Premier Seating, Premier Registration, Distinguished Reviewer, etc
3. Offering more attractive services for practitioners in industry;
4. Providing the world's best educational content and historical treasures for future generations, which only the CS can create with our pioneering researchers (for example, the Multicore Compiler Video Series found at www.computer.org/web/education/multicore-video-series);
5. Thinking about sustainable membership fees while considering the diversity of economic situations within the 10 regions;
6. Cooperating with other IEEE societies and sister societies in a timely and efficient manner;
7. Intelligibly introducing the latest computer-related technologies to younger generations, including children, so that they can realize their technological dreams.

Green Computing Systems R&D Center

Waseda University

Supported by METI (Mar. 2011 Completion)





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

<R & D Target>

Hardware, Software, Application for Super Low-Power Manycore

- **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, OSCAR Technology, etc

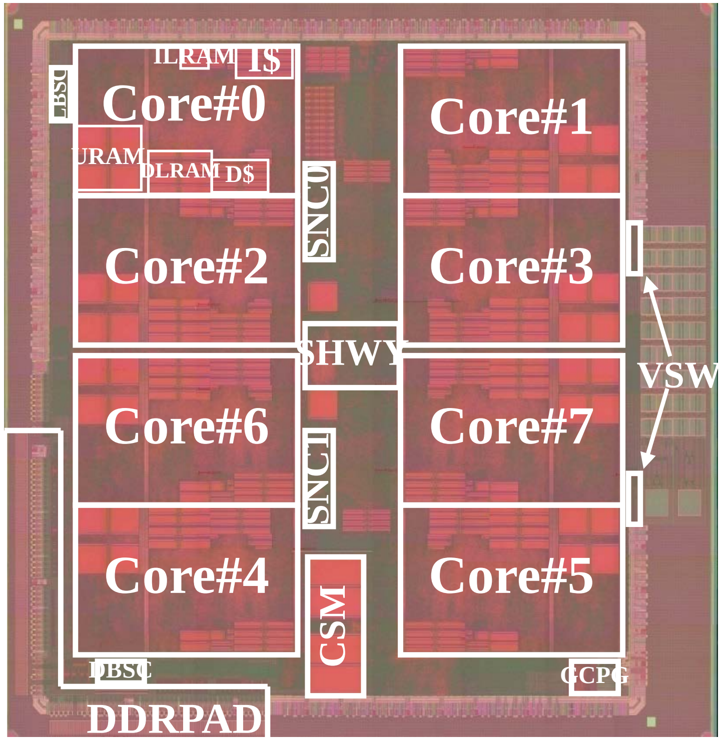
<Ripple Effect>

- **Low CO_2 (Carbon Dioxide) Emissions**
- **Creation Value Added Products**
- **Automobiles, Medical, IoT, Servers**

Beside Subway Waseda Station,
Near Waseda Univ. Main Campus

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”

Power $\propto$ Frequency * Voltage^2

(Voltage $\propto$ Frequency)

➡ **Power $\propto$ 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>

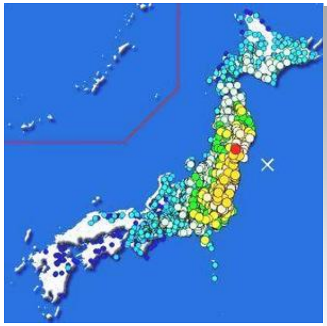
If 8cores are integrated on a chip,

Power is still 1/8 and

Performance becomes 2 times.

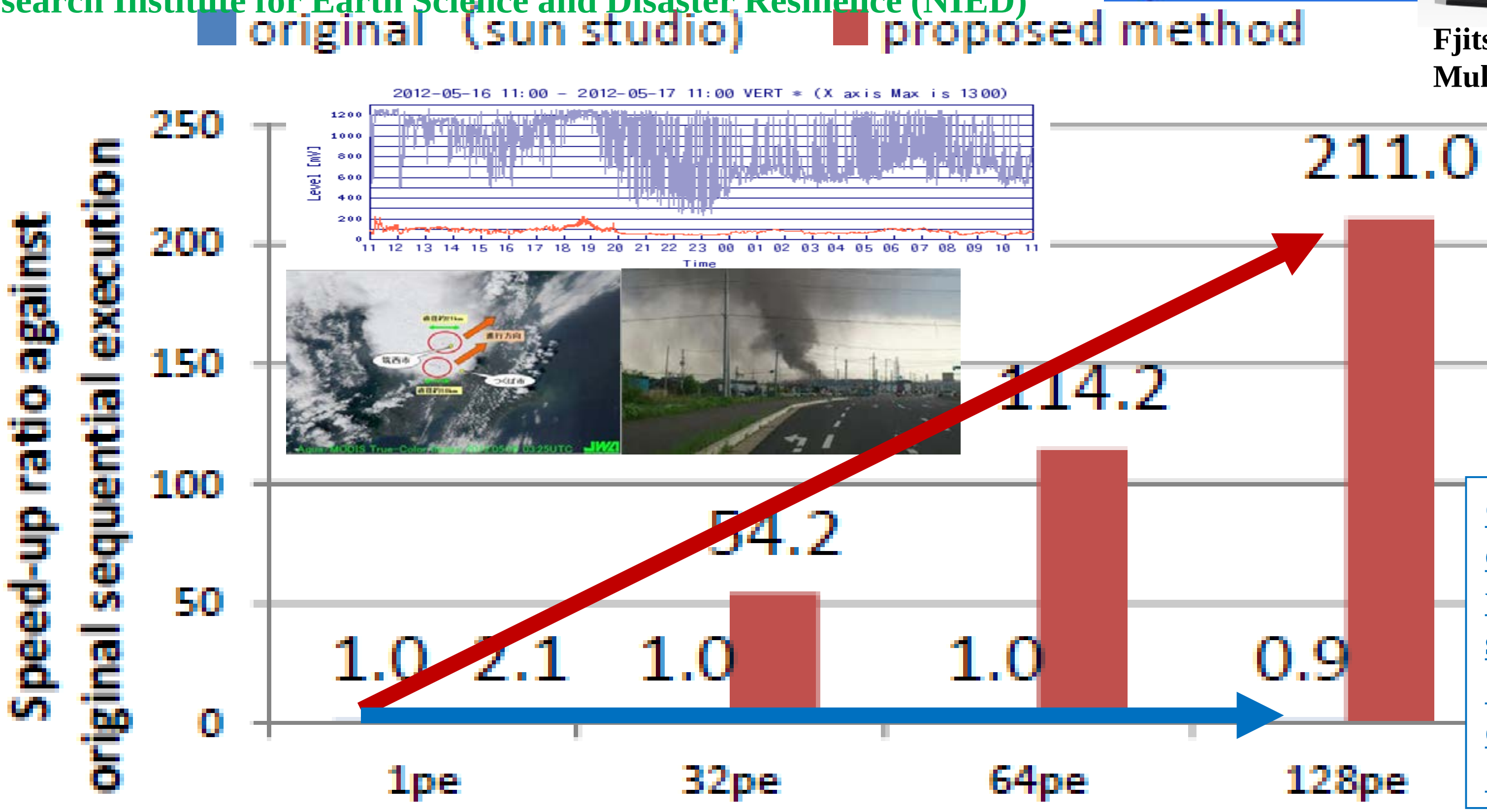
Parallel Soft is important for scalable performance of multicore (LCPC2015)

- 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



Fjitsu M9000 SPARC Multicore Server

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

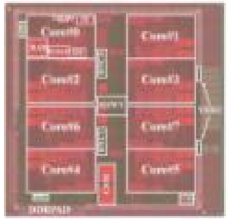


OSCAR Compiler gives us 211 times speedup with 128 cores

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

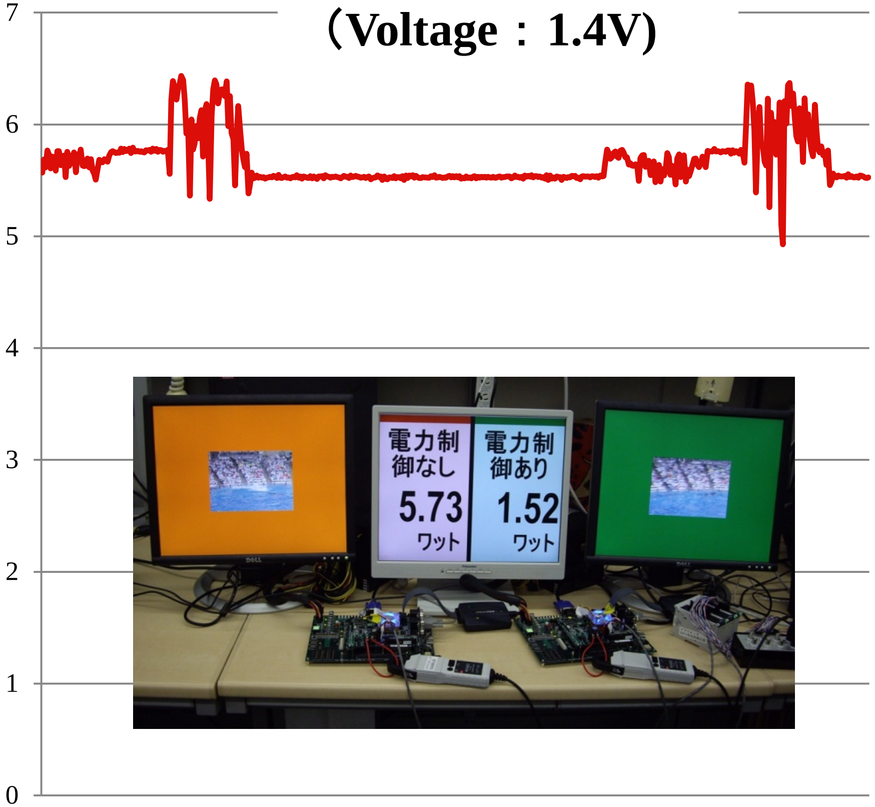
- Automatic parallelizing compiler available on the market gave us no speedup against execution time on 1 core on 64 cores
  - Execution time with 128 cores was slower than 1 core (0.9 times speedup)
- Advanced OSCAR parallelizing compiler gave us 211 times speedup with 128cores against execution time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

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

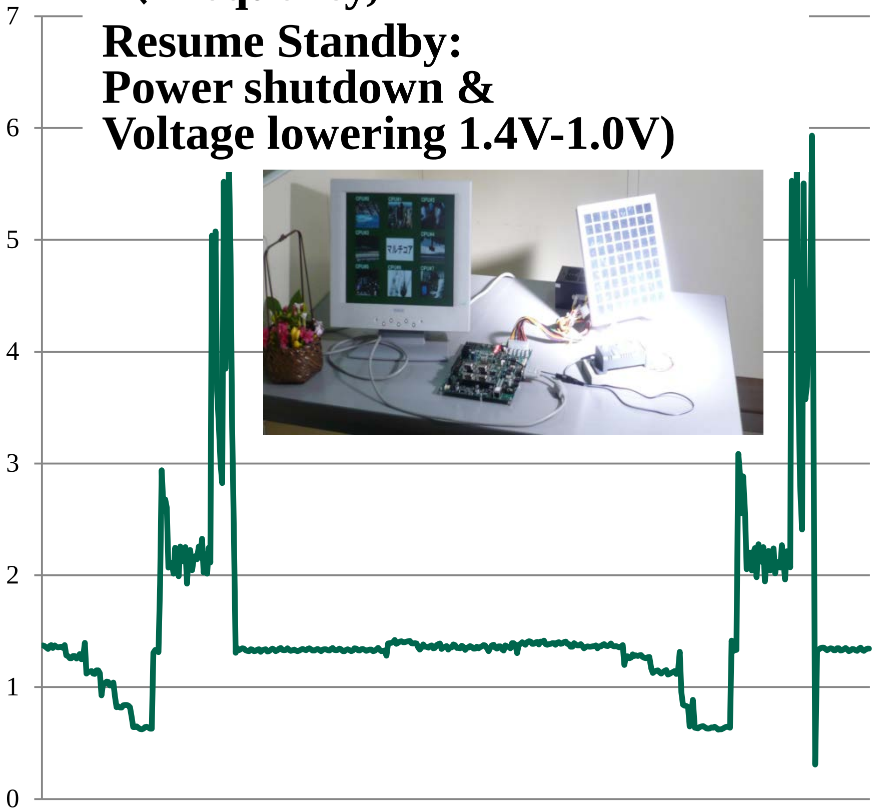


MPEG2 Decoding with 8 CPU cores

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]

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)

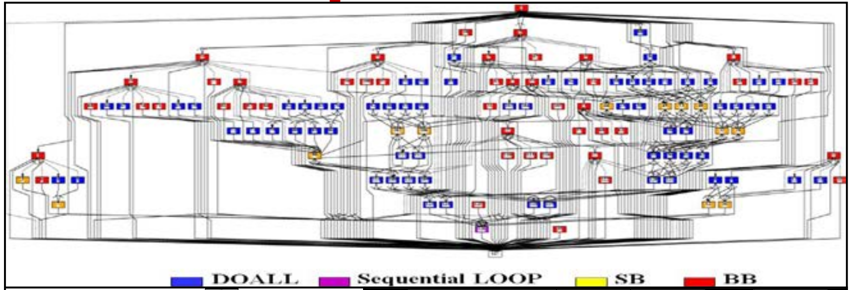
Data Transfer Overlapping(2016 partially)

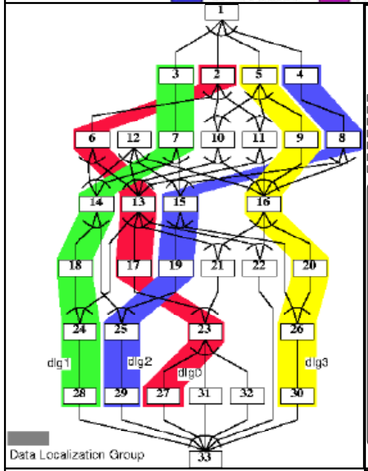
Data transfer overlapping using Data Transfer Controllers (DMAs)

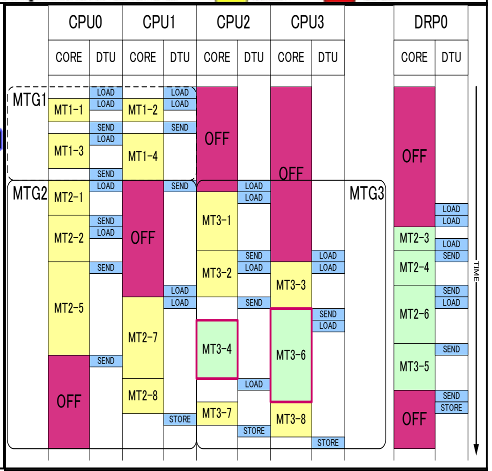
Power Reduction

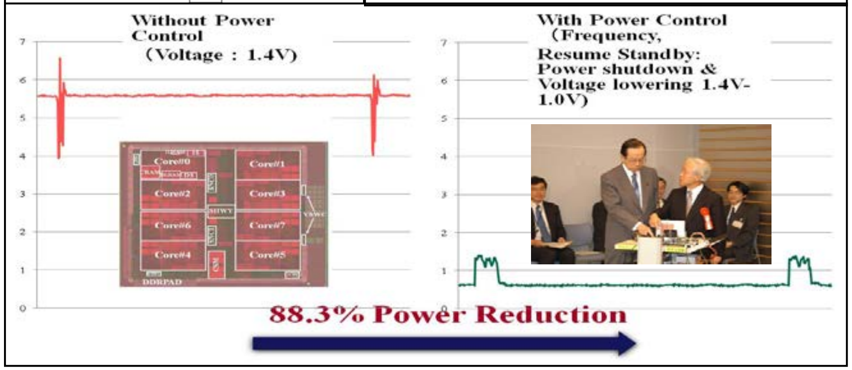
(2005 for Multicore, 2011 Multi-processes, 2013 on ARM)

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









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

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

⋮

Generation of parallel machine codes using sequential compilers

Low Power Homogeneous Multicore Code Generation

API Analyzer | Existing sequential compiler

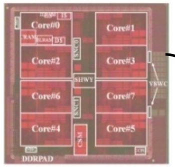
Low Power Heterogeneous Multicore Code Generation

API Analyzer (Available from Waseda) | Existing sequential compiler

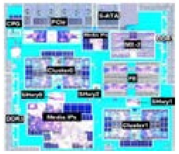
⋮

Server Code Generation

OpenMP Compiler



Homegeneous Multicore s from Vendor A (SMP servers)



Heterogeneous Multicores from Vendor B

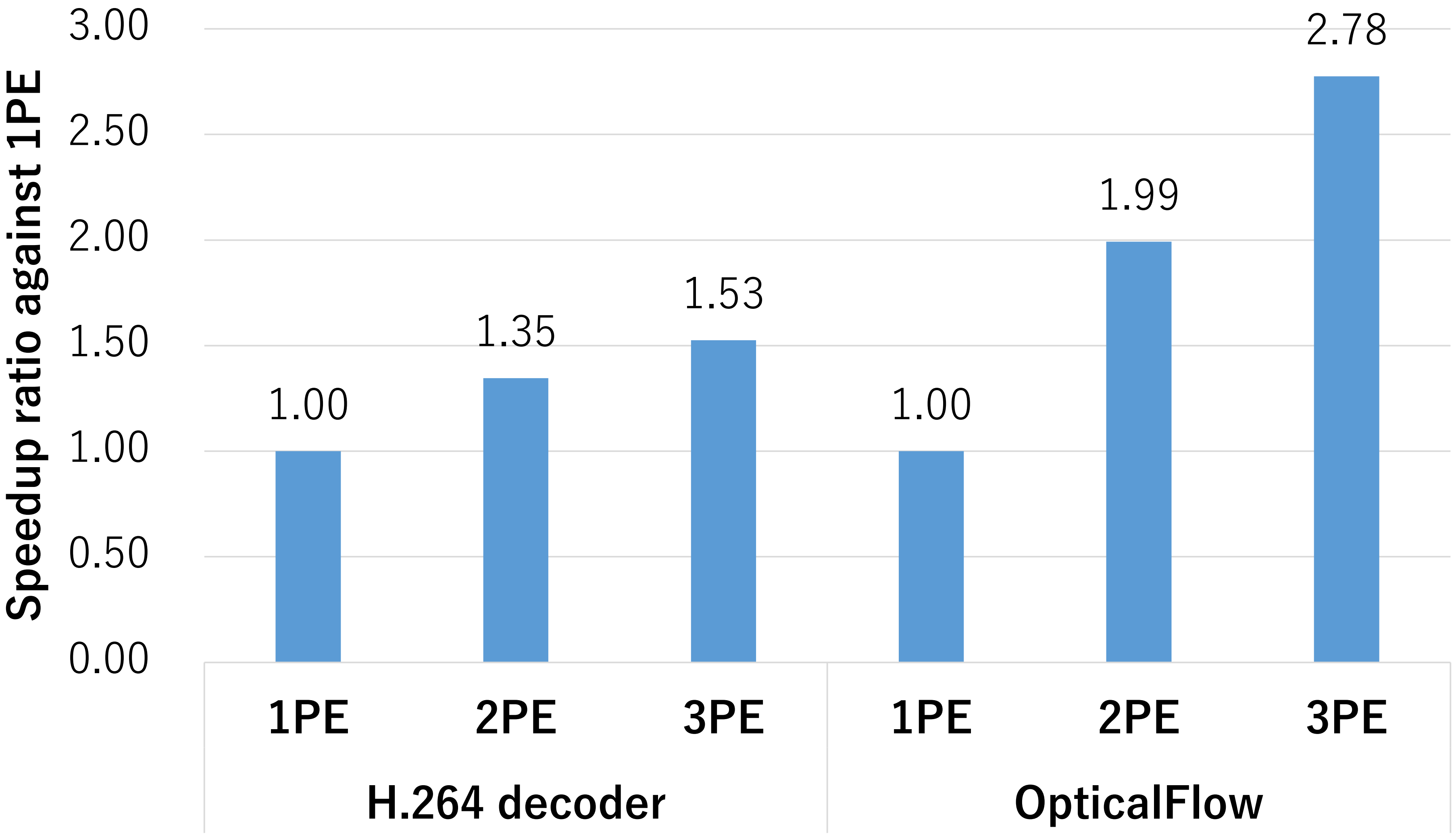
Shred memory servers

Executable on various multicores

OSCAR: Optimally Scheduled Advanced Multiprocessor

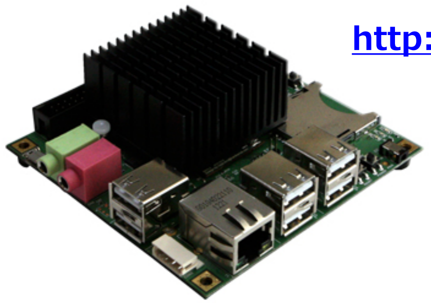
API : Application Program Interface

Speedup ratio for H.264 and Optical Flow on ARM Cortex-A9 Android 3 cores by OSCAR Automatic Parallelization



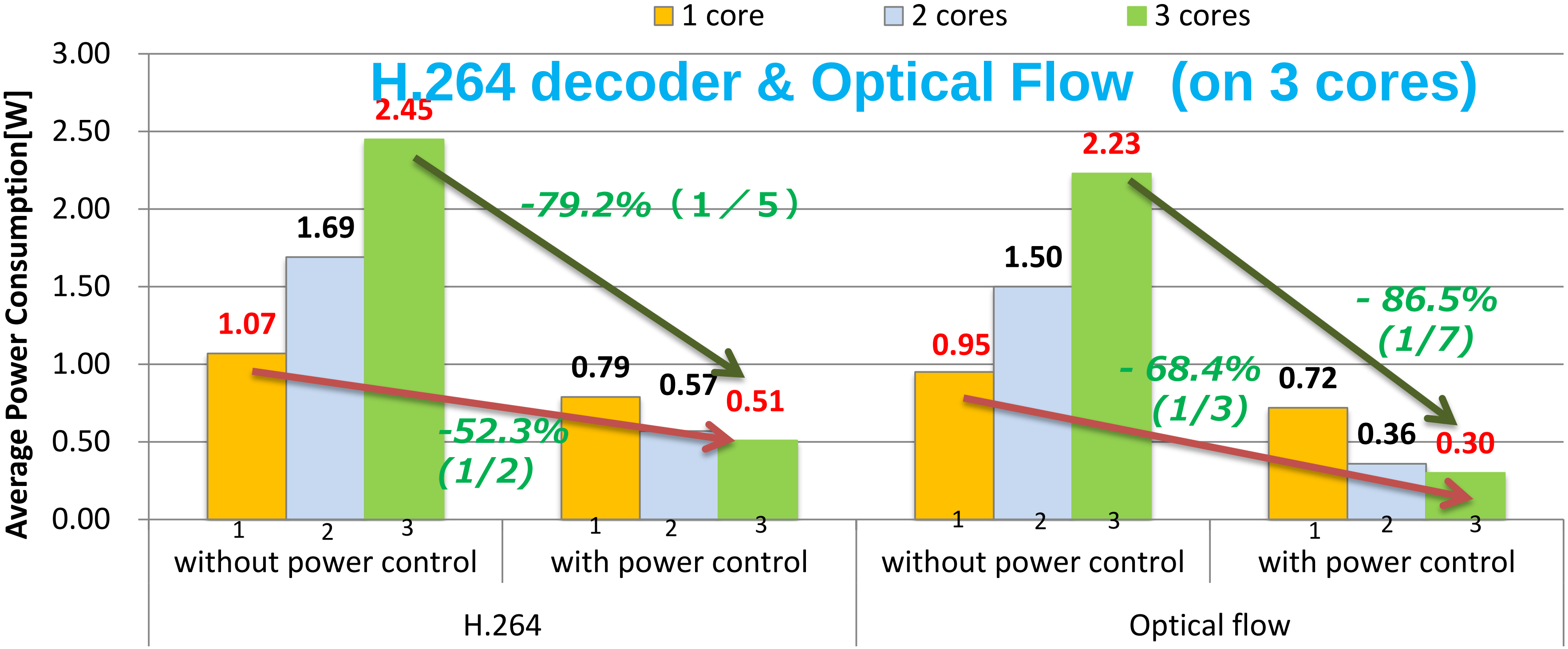
Automatic Power Reduction on ARM CortexA9 with Android

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



ODROID X2

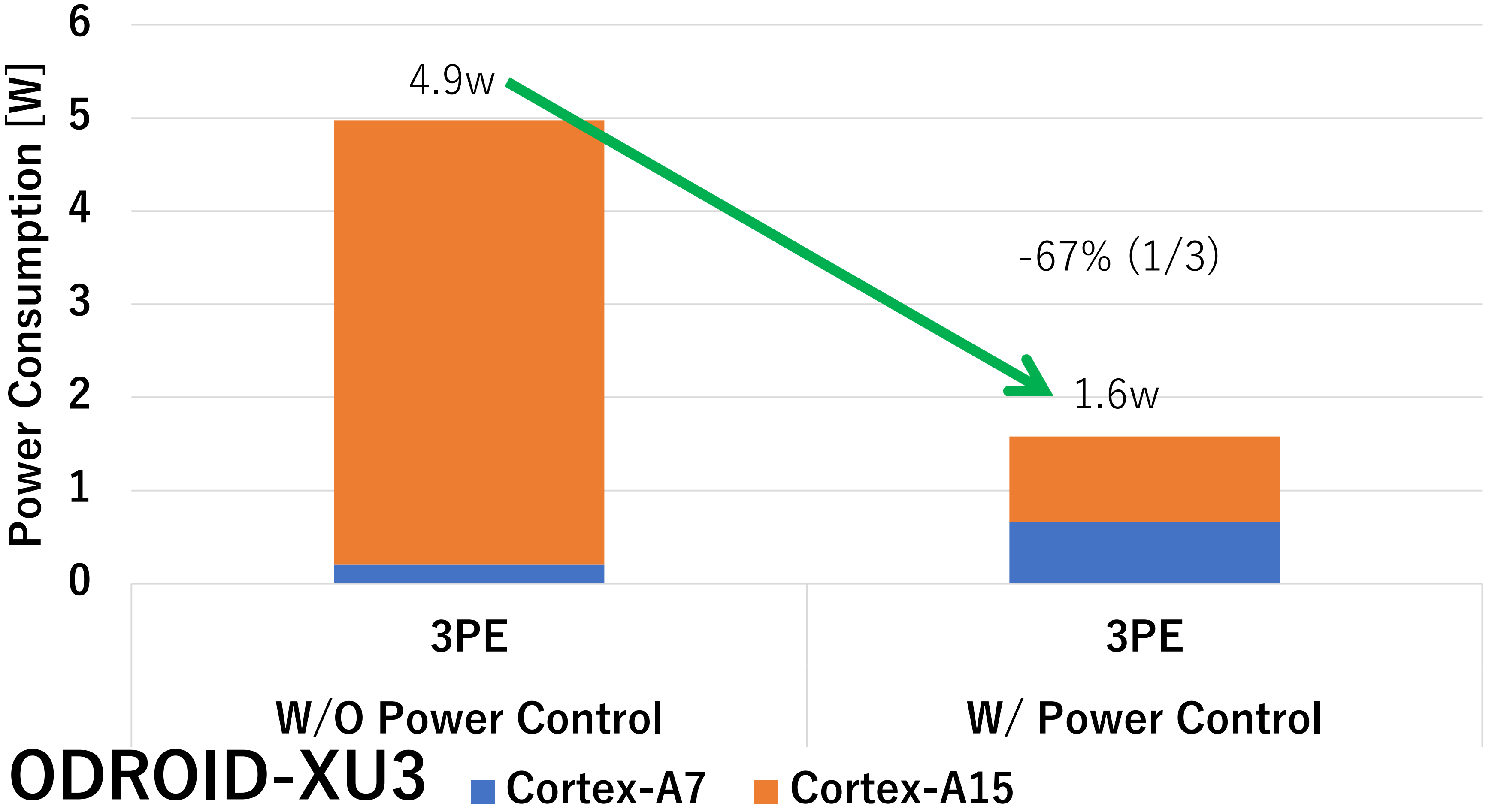
Samsung Exynos4412 Prime, ARM Cortex-A9 Quad core 1.7GHz～0.2GHz, used by Samsung's Galaxy S3



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

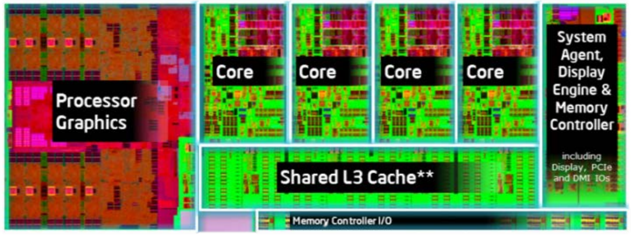


- ODROID-XU3
  - Samsung Exynos 5422 Processor
    - 4x Cortex-A15 2.0GHz, 4x Cortex-A7 1.4GHz big.LITTLE Architecture
    - 2GB LPDDR3 RAM

Frequency can be changed by

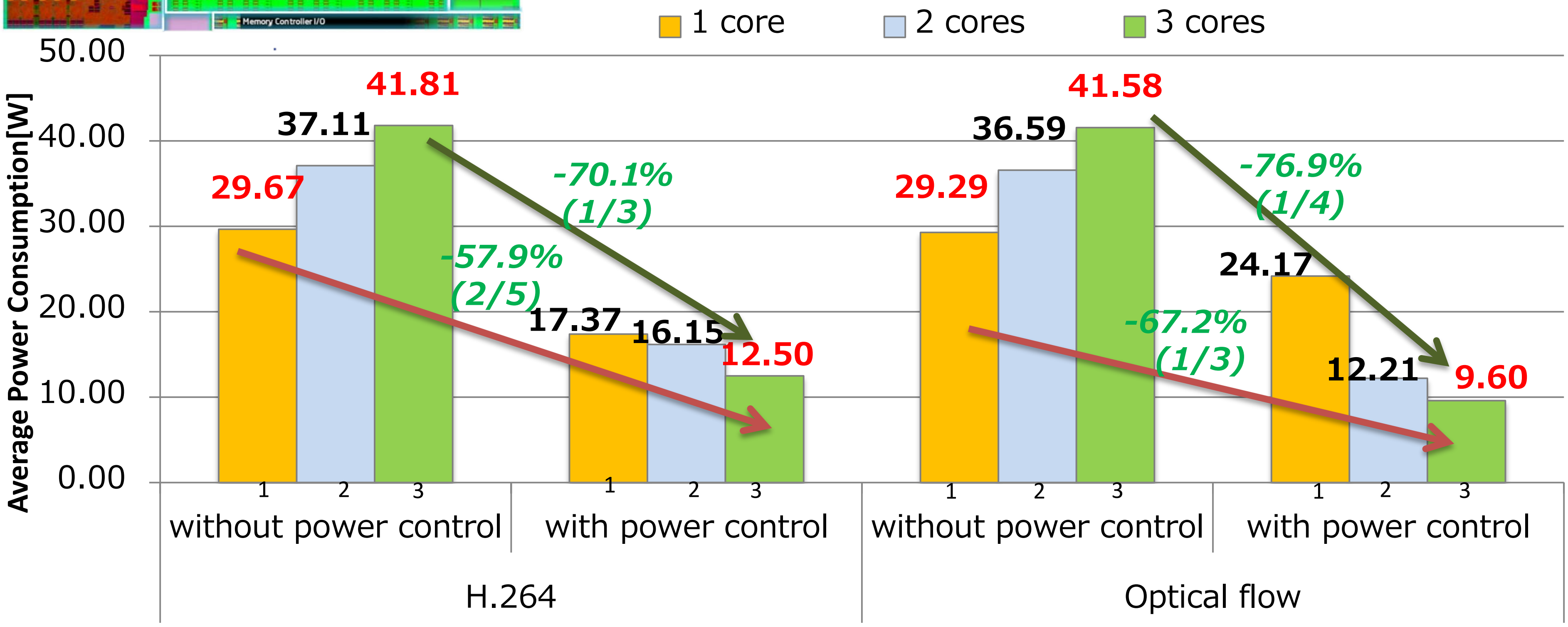
Automatic Power Reuction on Intel Haswell

H.264 decoder & Optical Flow (3cores)


Processor Graphics
Core
Core
Core
Core
System Agent, Display Engine & Memory Controller
including Display, PCIe and DMI IOs
Shared L3 Cache**
Memory Controller I/O

H81M-A, Intel Core i7 4770k

Quad core, 3.5GHz～0.8GHz

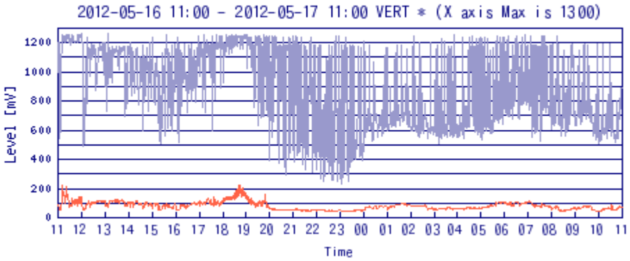


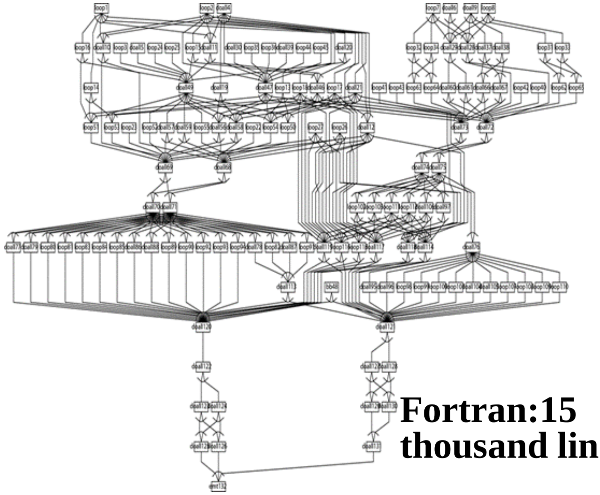
Power for 3cores was reduced to 1/3～1/4 against without software power control

Power for 3cores was reduced to 2/5～1/3 against ordinary 1core execution

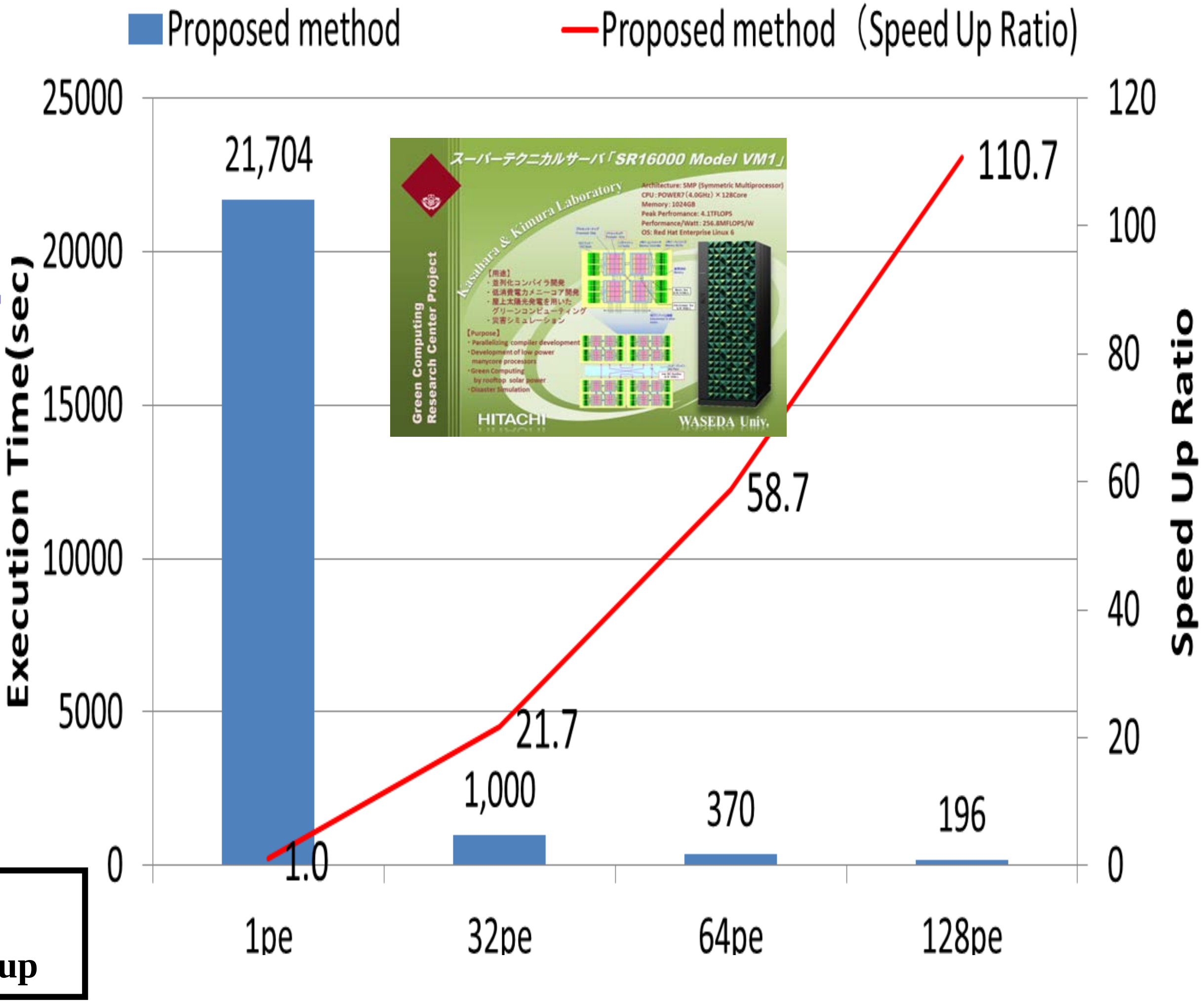
110 Times Speedup against the Sequential Processing for GMS Earthquake Wave Propagation Simulation on Hitachi SR16000

（Power7 Based 128 Core Linux SMP） (LCPC2015)





Fortran:15 thousand lines



スーパーテクニカルサーバ「SR16000 Model VM1」

Kasahara & Kimura Laboratory

Architecture: SMP (Symmetric Multiprocessor)
CPU : POWER7（4.0GHz）×128Core
Memory : 1024GB
Peak Perfromance: 4.1TFLOPS
Performance/Watt : 256.8MFLOPS/W
OS: Red Hat Enterprise Linux 6

【用途】
・並列化コンパイラ開発
・低消費電力メニーコア開発
・屋上太陽光発電を用いたグリーンコンピューティング
・災害シミュレーション

【Purpose】
・Parallelizing compiler development
・Development of low power manycore processors
・Green Computing by rooftop solar power
・Disaster Simulation

Green Computing Research Center Project

HITACHI

WASEDA Univ.

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

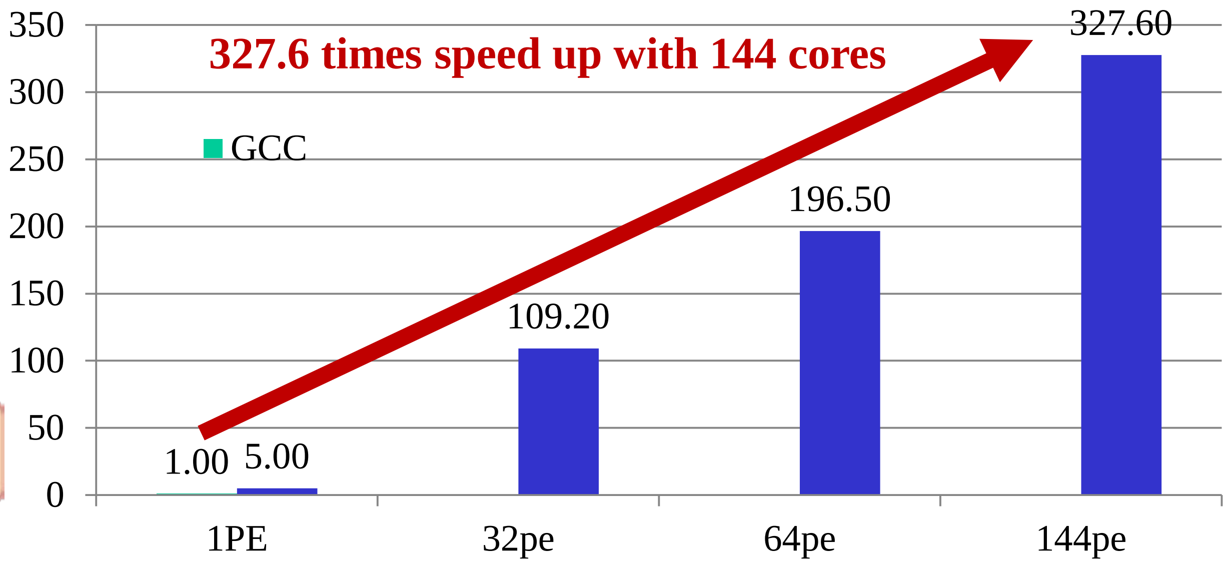
Performance on Multicore Server for Latest Cancer Treatment Using Heavy Particle (Proton, Carbon Ion)

327 times speedup on 144 cores

放射線医学研究所

施設の費用：120億円

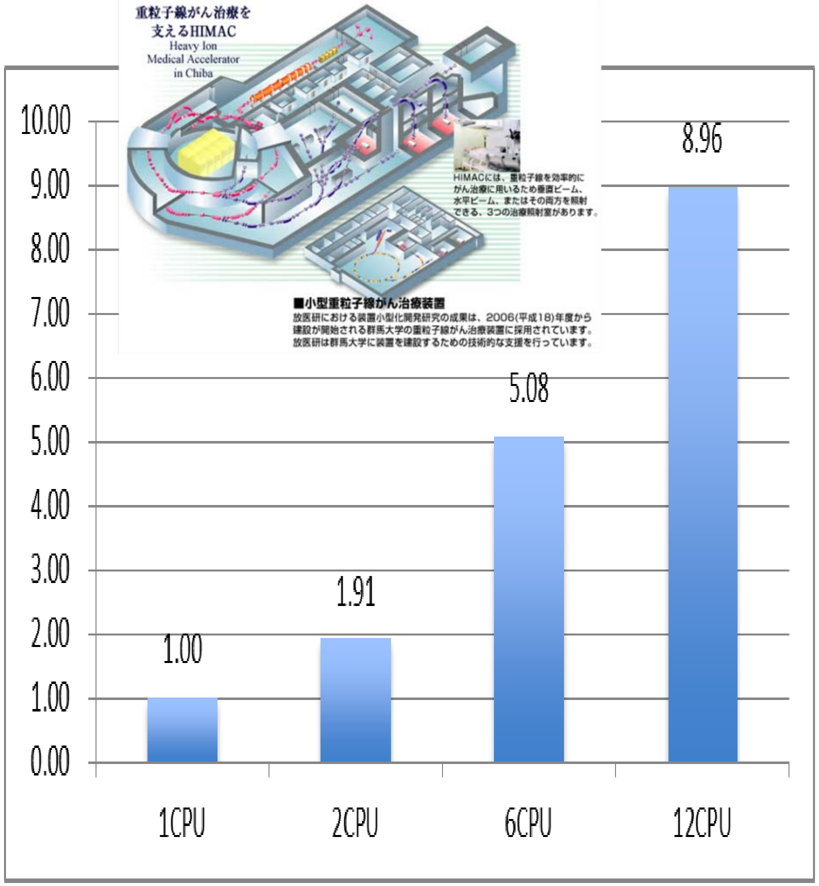
Hitachi 144cores SMP Blade Server BS500:
Xeon E7-8890 V3(2.5GHz　18core/chip) x8 chip

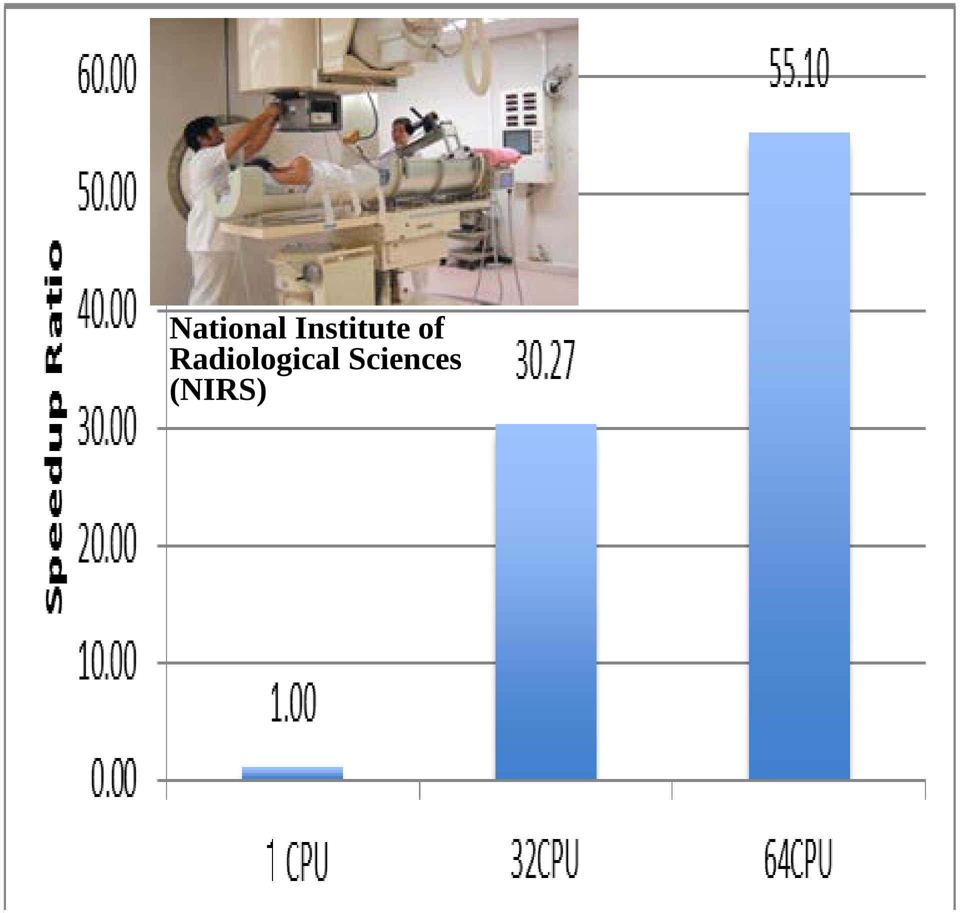


- ➢ **Original sequential execution time 2948 sec（50 minutes）using GCC was reduced to 9 sec with 144 cores（327.6 times speedup）**
  - ➢ **Reduction of treatment cost and reservation waiting period is expected**

Cancer Treatment
Carbon Ion Radiotherapy

(Previous best was 2.5 times speedup on 16 processors with hand optimization)





8.9times speedup by 12 processors

Intel Xeon X5670 2.93GHz 12 core SMP (Hitachi HA8000)

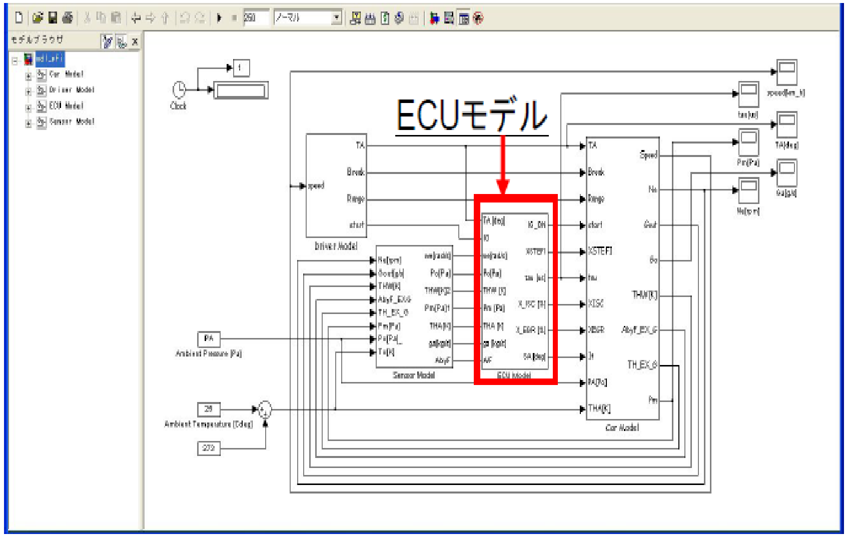
55 times speedup by 64 processors

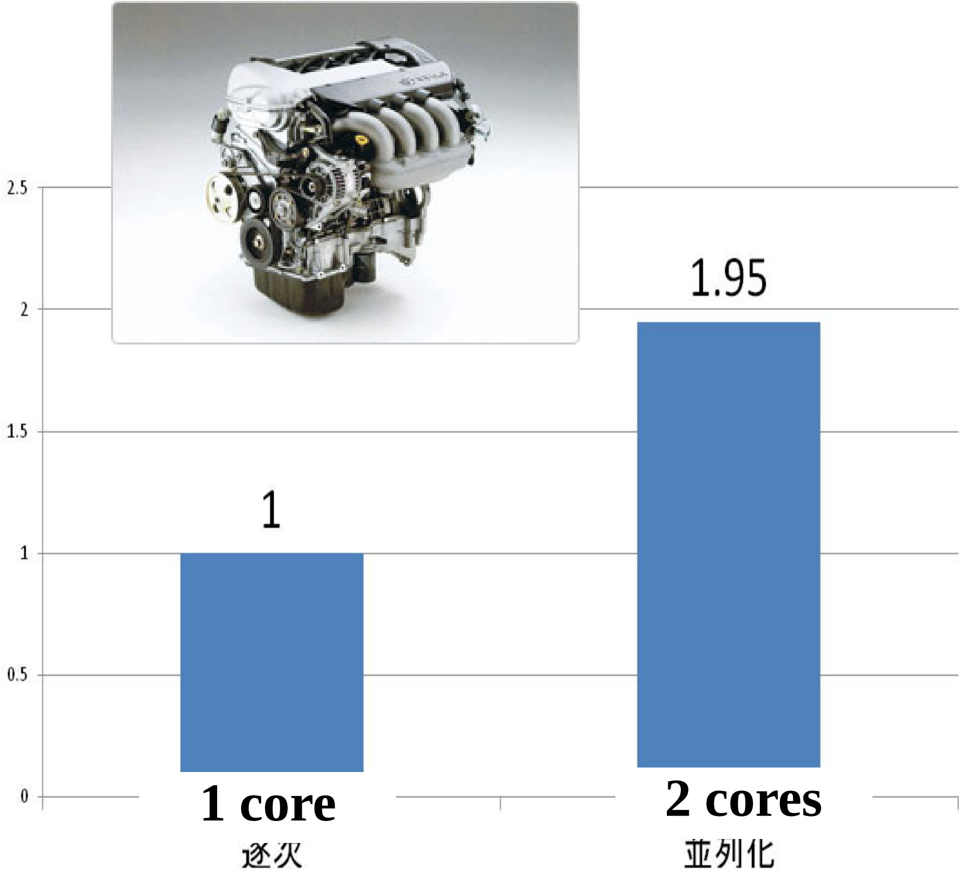
IBM Power 7 64 core SMP (Hitachi SR16000)

Engine Control by multicore with Denso

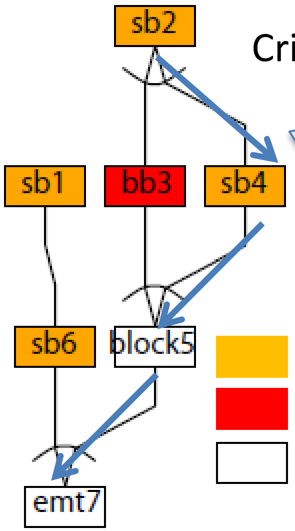
Though so far parallel processing of the engine control on multicore has been very difficult, Denso and Waseda succeeded 1.95 times speedup on 2core V850 multicore processor.

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





MTG of Crankshaft Program Using Inline Expansion

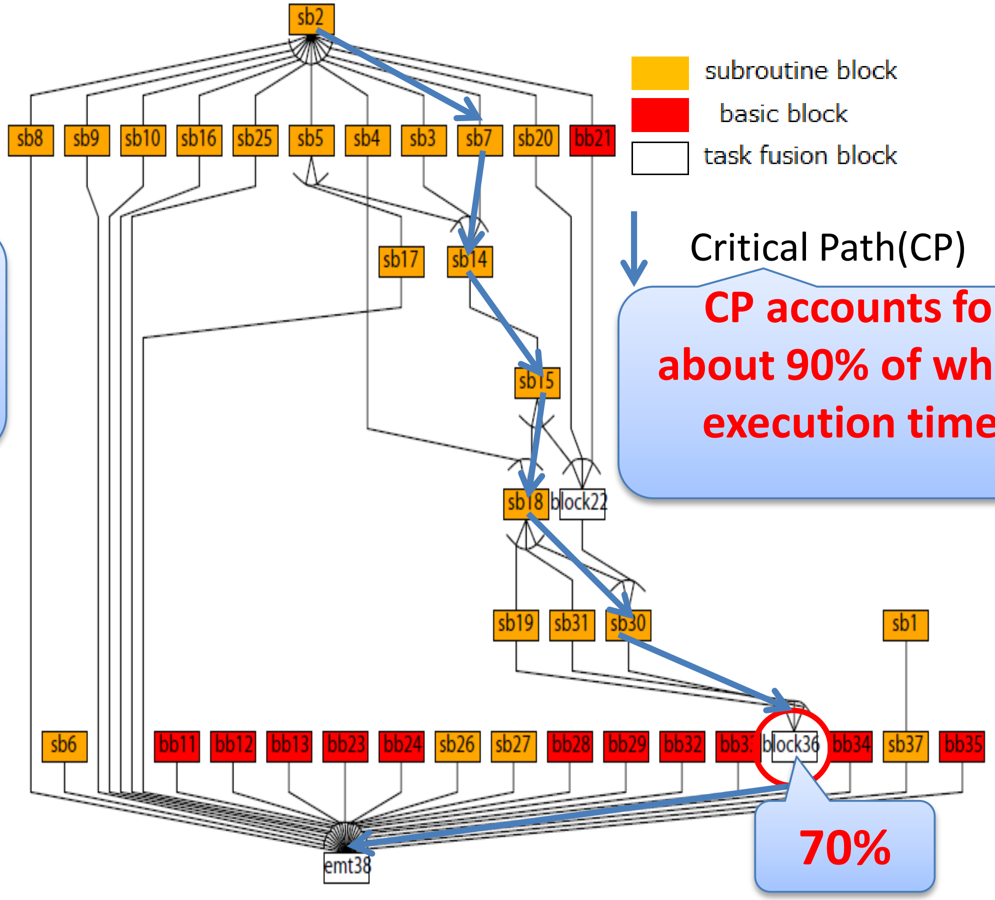


Critical Path(CP)

CP accounts for about 99% of whole execution time.

🟧	subroutine block
🟥	basic block
□	task fusion block

MTG of crankshaft program before restructuring



Critical Path(CP)

CP accounts for about 90% of whole execution time.

70%

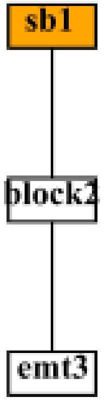
Not enough coarse grain parallelism yet!

3.2 Restructuring: Duplicating If-statements

- Duplicating if-statements is effective
  - To increase coarse grain parallelism
- Duplicates fused tasks having inner parallelism

Improves coarse grain parallelism

Func1 depends on func3



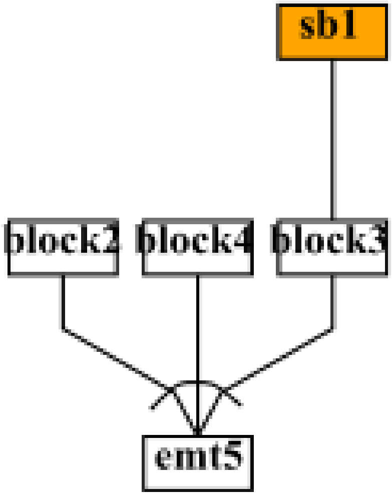
```
func1();
if (condition) {
    func2();
    func3();
    func4();
}
```

No dependemce

Copying if-condition for each function

```
func1();
if (condition) {
    func2();
}
if (condition) {
    func3();
}
if (condition) {
    func4();
}
```

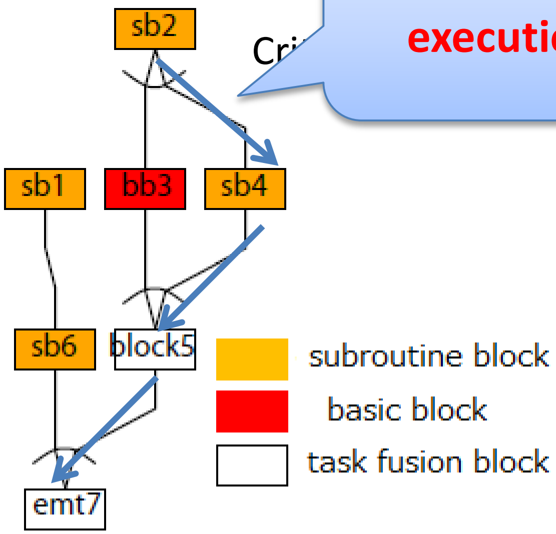
before duplicating if-statements



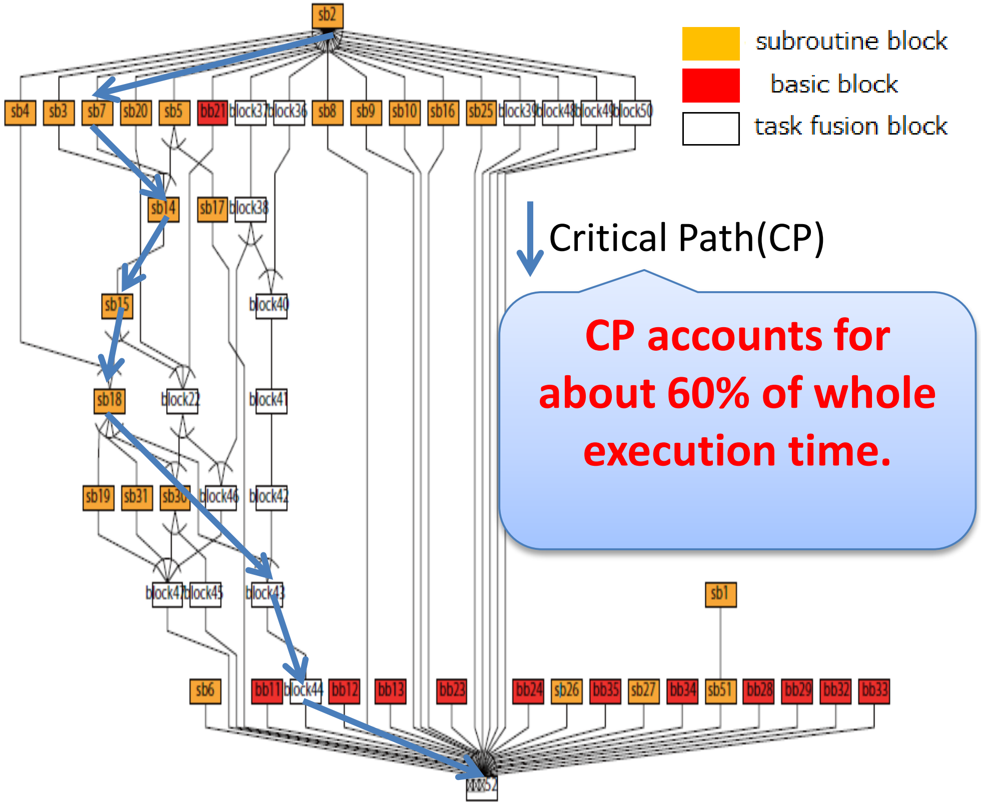
after duplicating if-statements

MTG of Crankshaft Program Using Inline Expansion and Duplicating If-statements

CP accounts for over 99% of whole execution time.



MTG of crankshaft program before restructuring



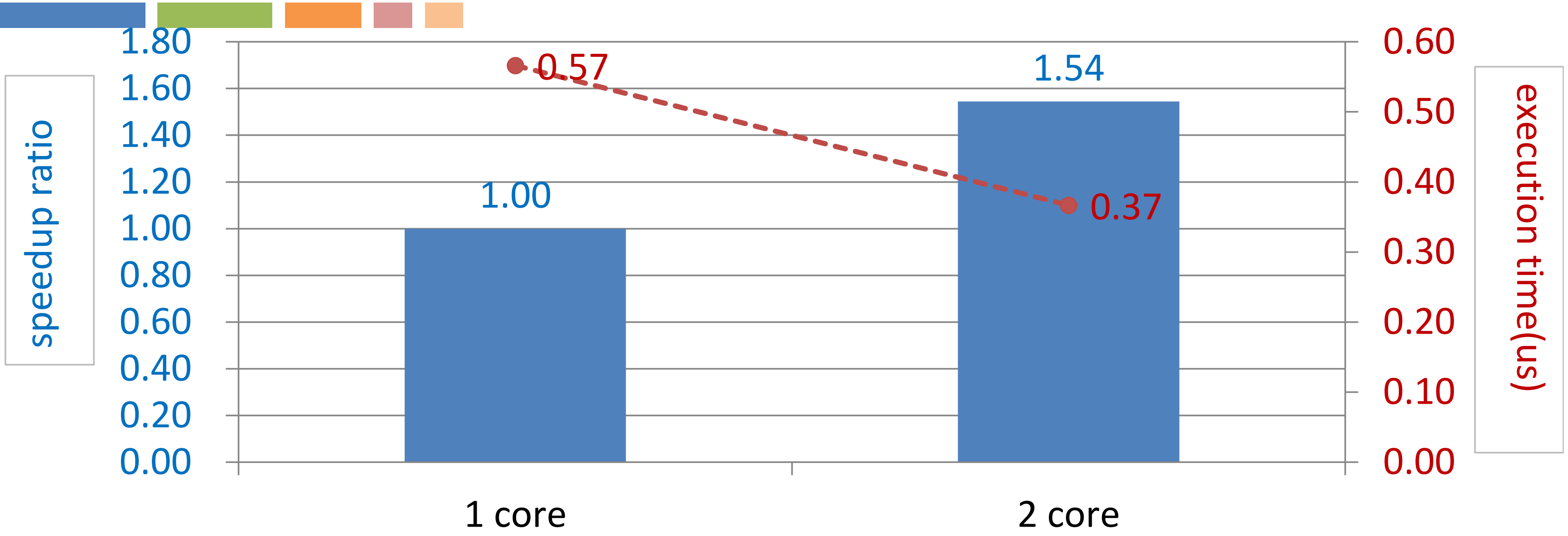
CP accounts for about 60% of whole execution time.

MTG of crankshaft program after restructuring

- Succeed to reduce CP
  - 99% -> 60%

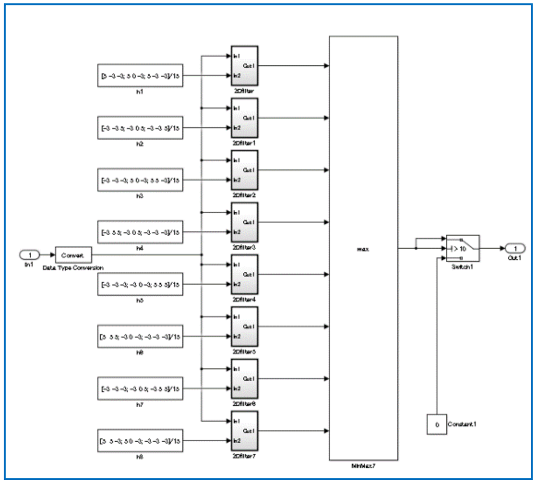
Successfully increased coarse grain parallelism

Evaluation of Crankshaft Program with Multi-core Processors



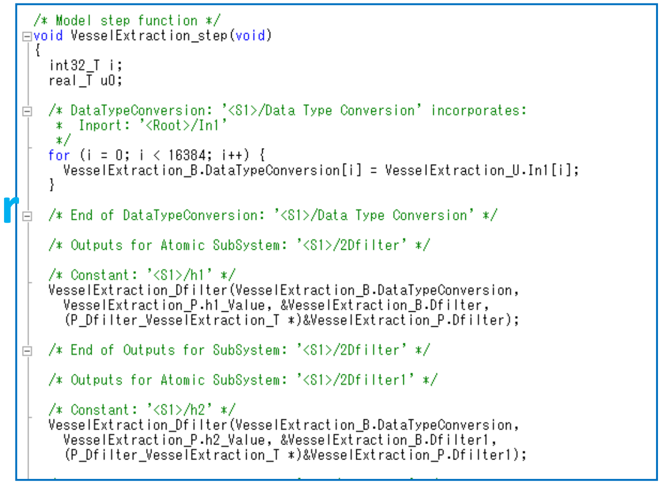
- Attain 1.54 times speedup on RPX
  - There are no loops, but only many conditional branches and small basic blocks and difficult to parallelize this program
- This result shows possibility of multi-core processor for engine control programs

OSCAR Compile Flow for Simulink Applications



Simulink model

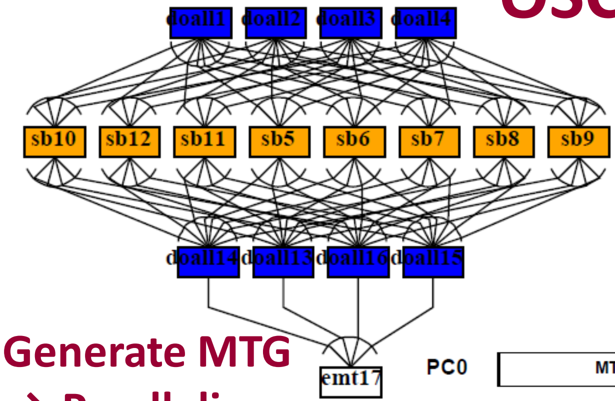
Generate C code using Embedded Coder



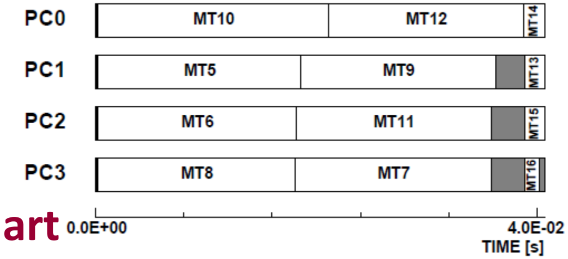
```
/* Model step function */
void VesselExtraction_step(void)
{
  int32_T i;
  real_T u0;

  /* 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

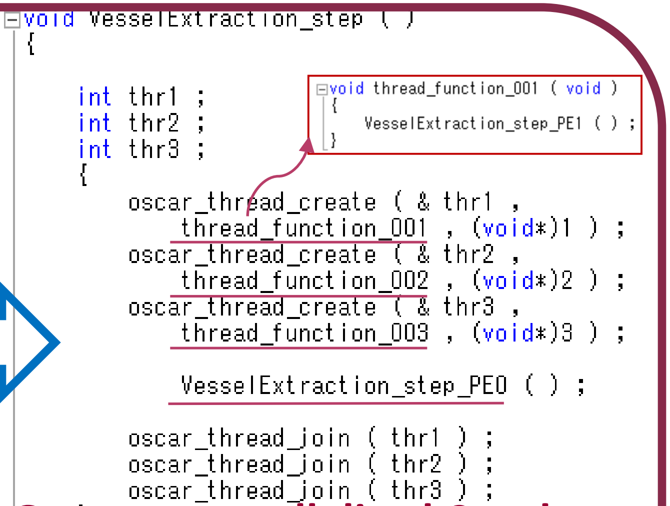
OSCAR Compiler



(1) Generate MTG
→ Parallelism



(2) Generate gantt chart
→ Scheduling in a multicore



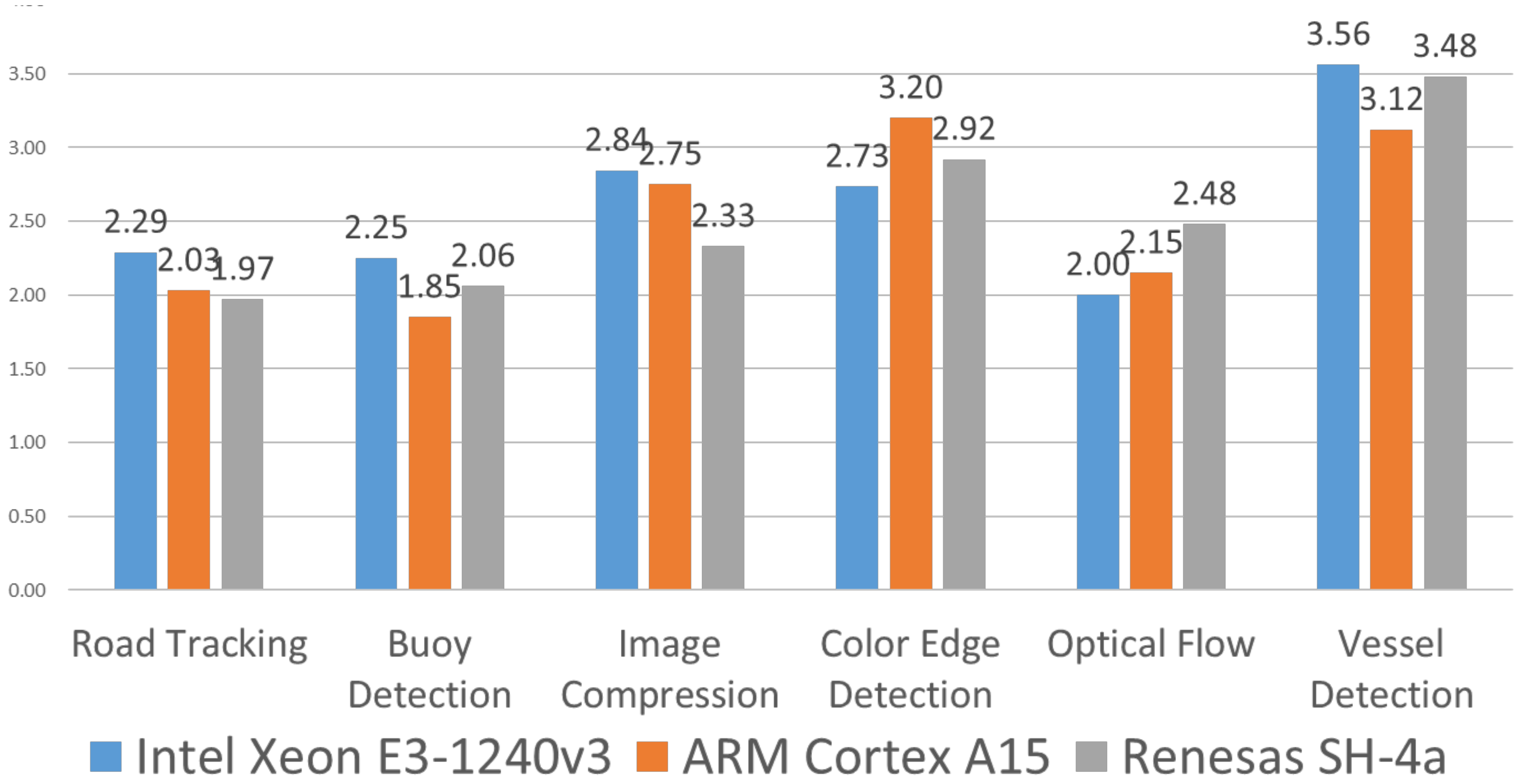
```
void VesselExtraction_step ( )
{

    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)

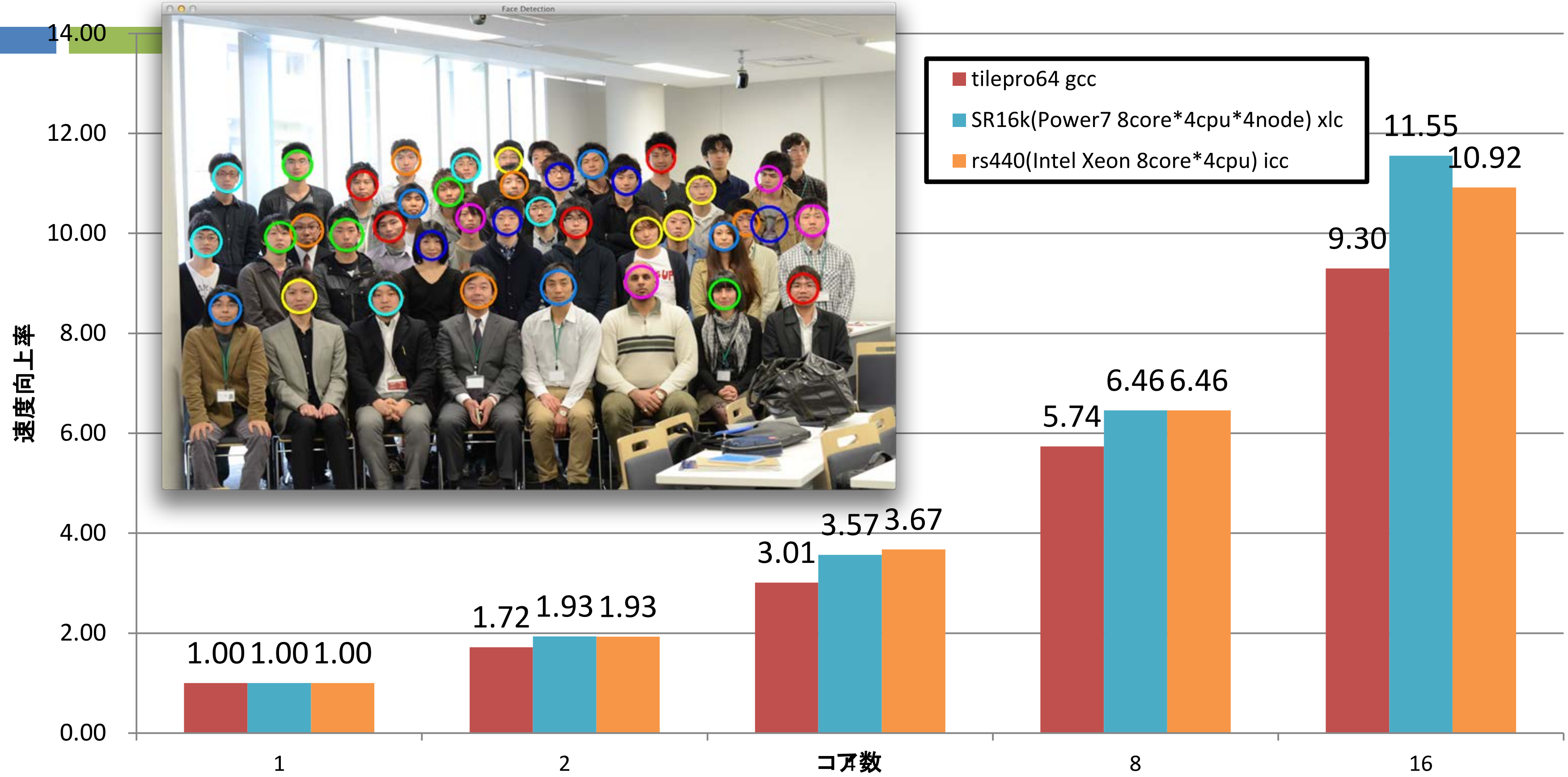
Speedups of MATLAB/Simulink Image Processing on Various 4core Multicores

(Intel Xeon, ARM Cortex A15 and Renesas SH4A)



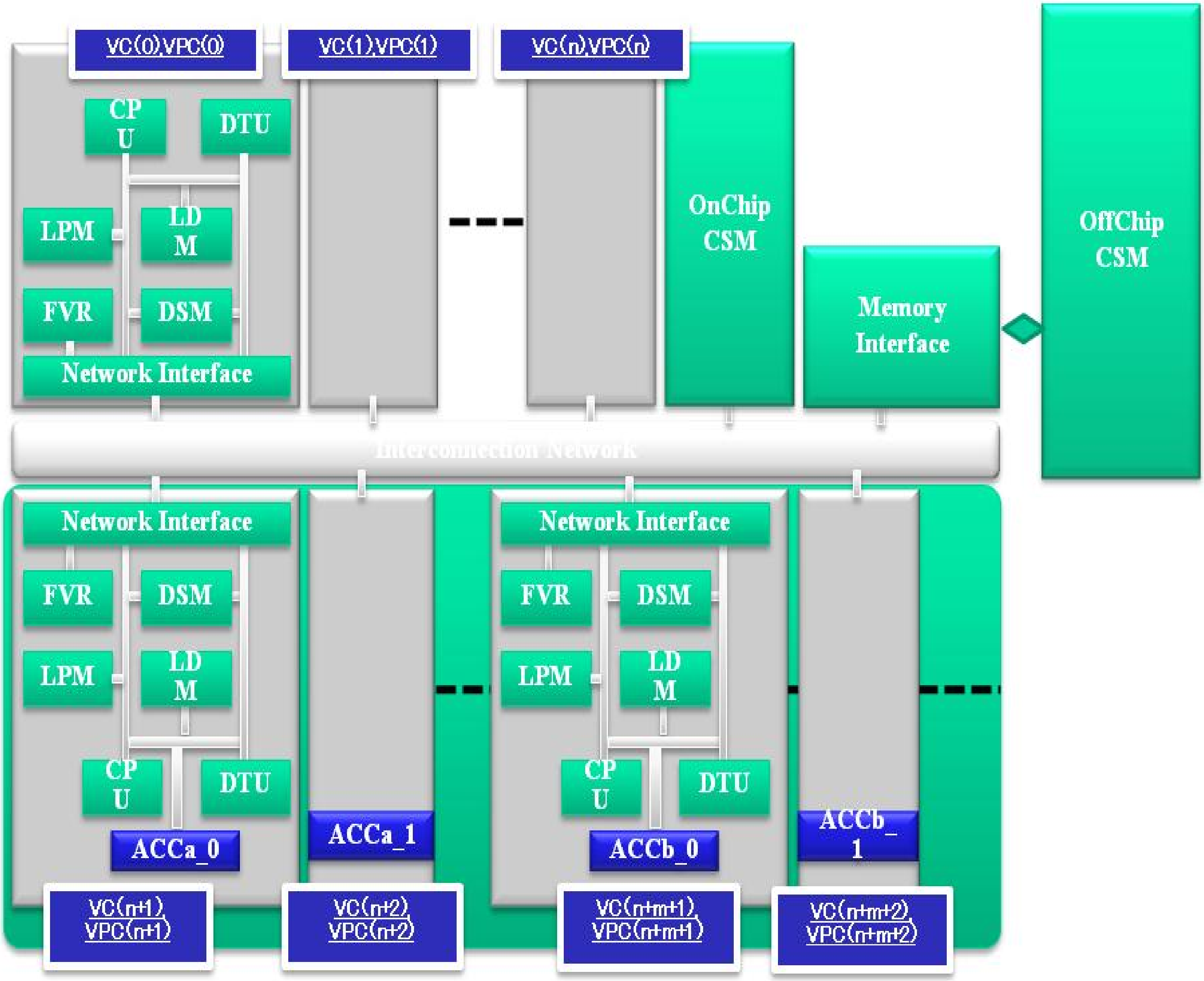
Road Tracking, Image Compression : http://www.mathworks.co.jp/jp/help/vision/examples
Buoy Detection : http://www.mathworks.co.jp/matlabcentral/fileexchange/44706-buoy-detection-using-simulink
Color Edge Detection : http://www.mathworks.co.jp/matlabcentral/fileexchange/28114-fast-edges-of-a-color-image--actual-color--not-converting-to-grayscale-/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

Parallel Processing of Face Detection on Manycore, Highend and PC Server



□ OSCAR compiler gives us 11.55 times speedup for 16 cores against 1 core on SR16000 Power7 highend server.

OSCAR Heterogeneous Multicore



DTU
- Data Transfer Unit

LPM
- Local Program Memory

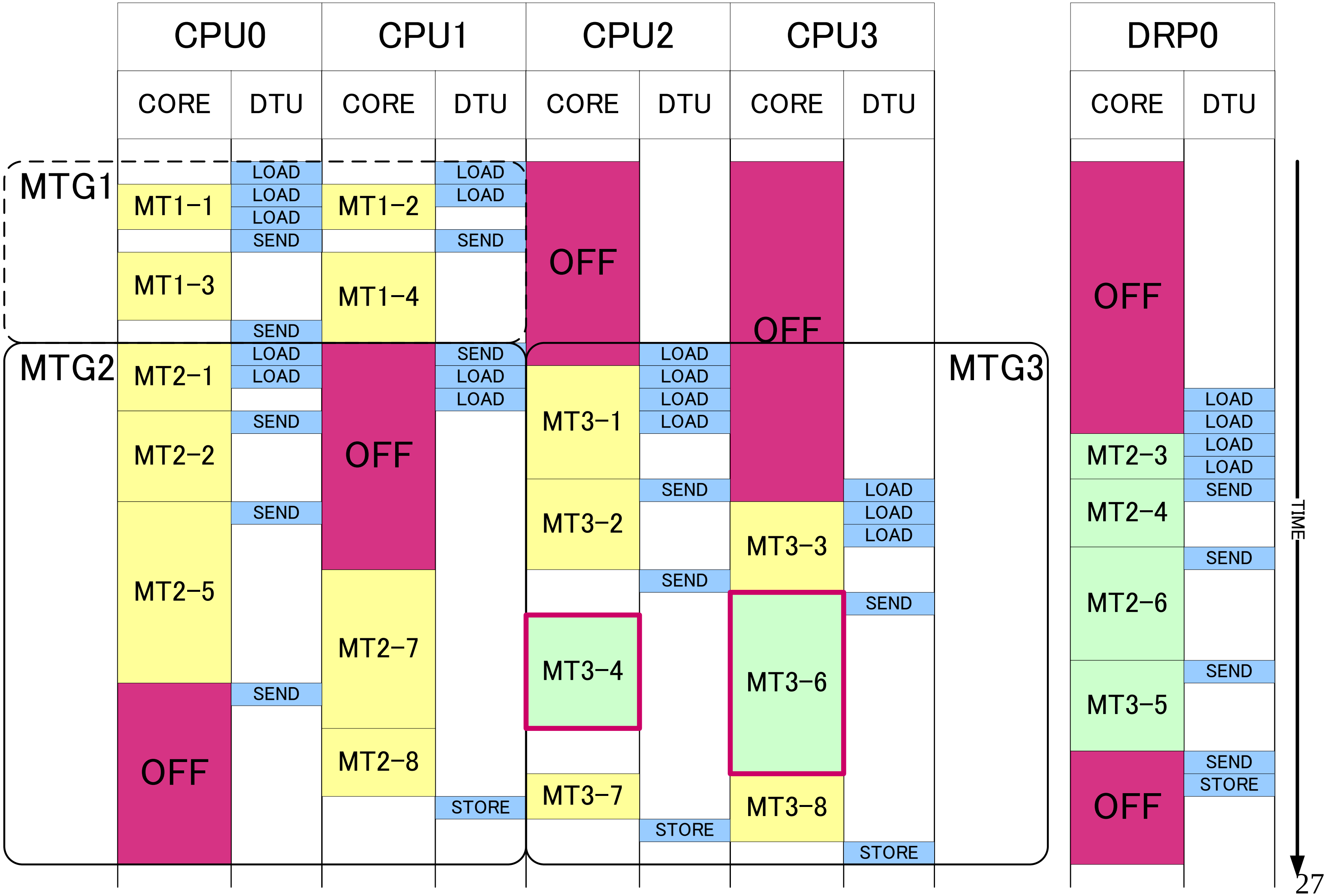
LDM
- Local Data Memory

DSM
- Distributed Shared Memory

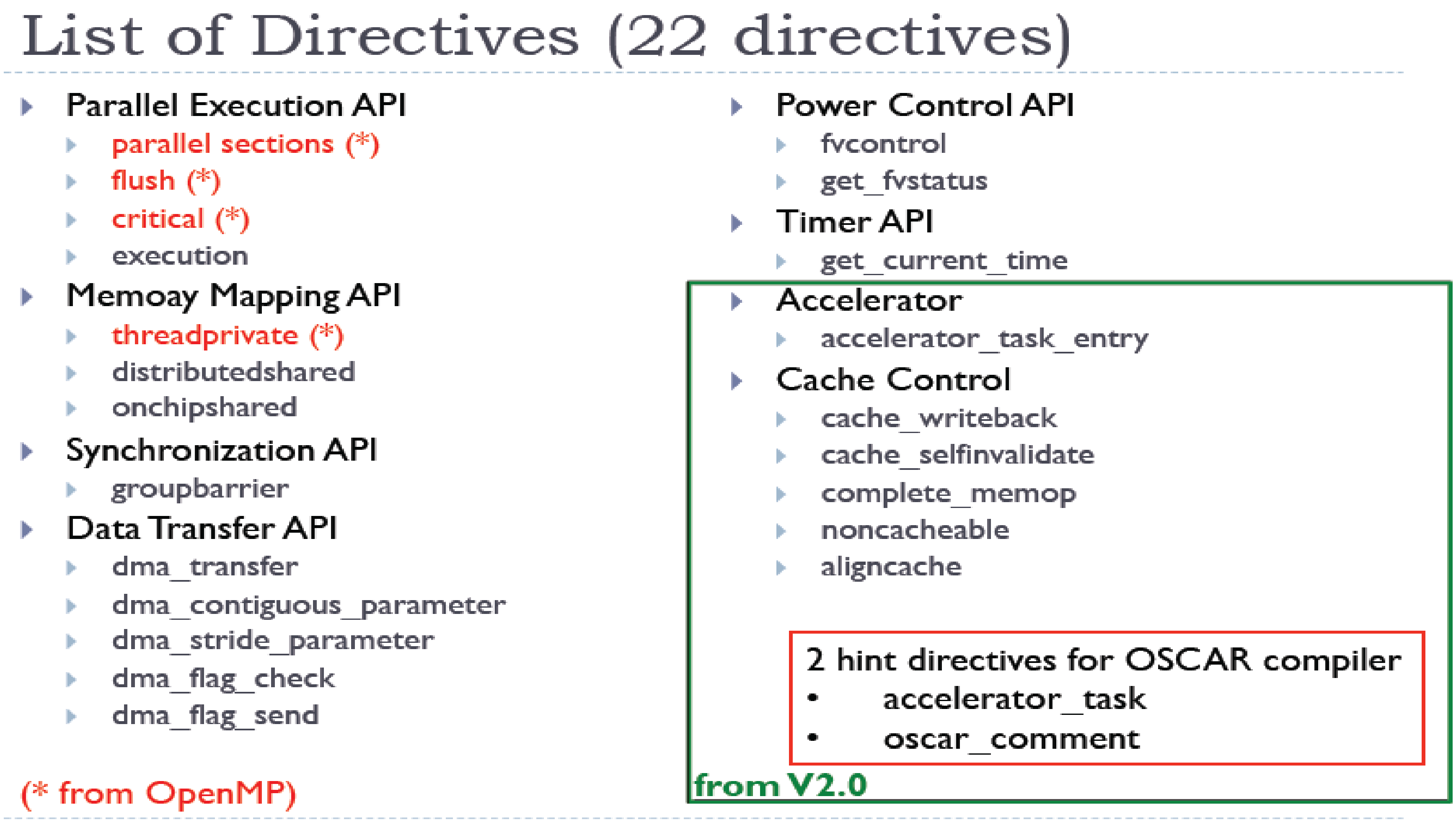
CSM
- Centralized Shared Memory

FVR
- Frequency/Voltage Control Register

An Image of Static Schedule for Heterogeneous Multi-core with Data Transfer Overlapping and Power Control



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)

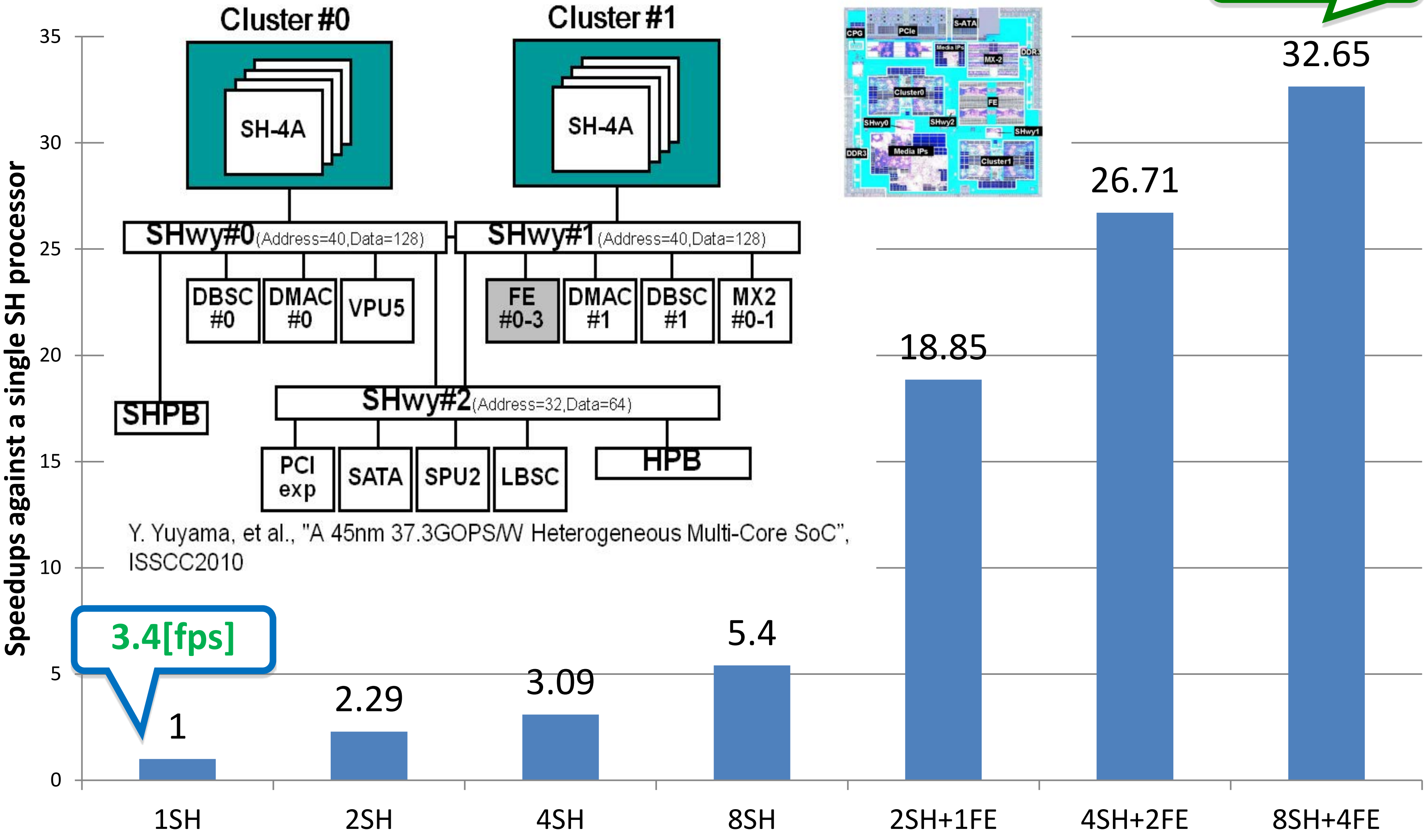
- Power Control API
  - fvcontrol
  - get_fvstatus
- Timer API
  - get_current_time
- Accelerator
  - accelerator_task_entry
- Cache Control
  - cache_writeback
  - cache_selfinvalidate
  - complete_memop
  - noncacheable
  - aligncache

2 hint directives for OSCAR compiler
- accelerator_task
- oscar_comment

from V2.0

33 Times Speedup Using OSCAR Compiler and OSCAR API on RP-X

(Optical Flow with a hand-tuned library)



111[fps]

3.4[fps]

Y. Yuyama, et al., "A 45nm 37.3GOPS/W Heterogeneous Multi-Core SoC", ISSCC2010

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

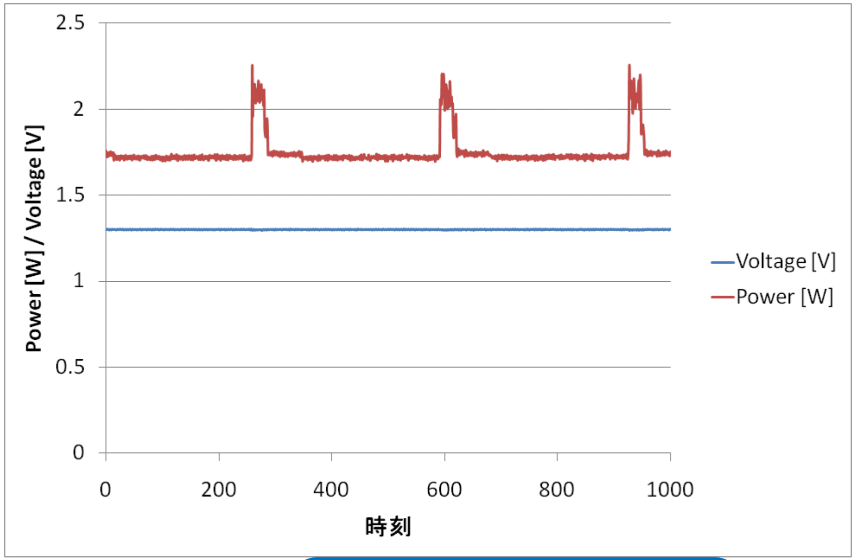
Without Power Reduction

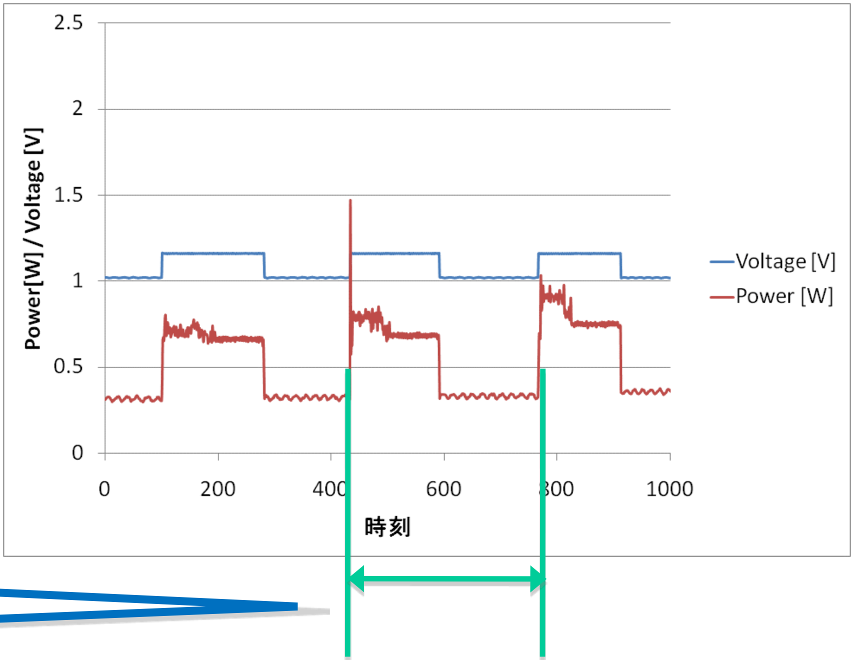
With Power Reduction by OSCAR Compiler

70% of power reduction

Average:1.76[W]

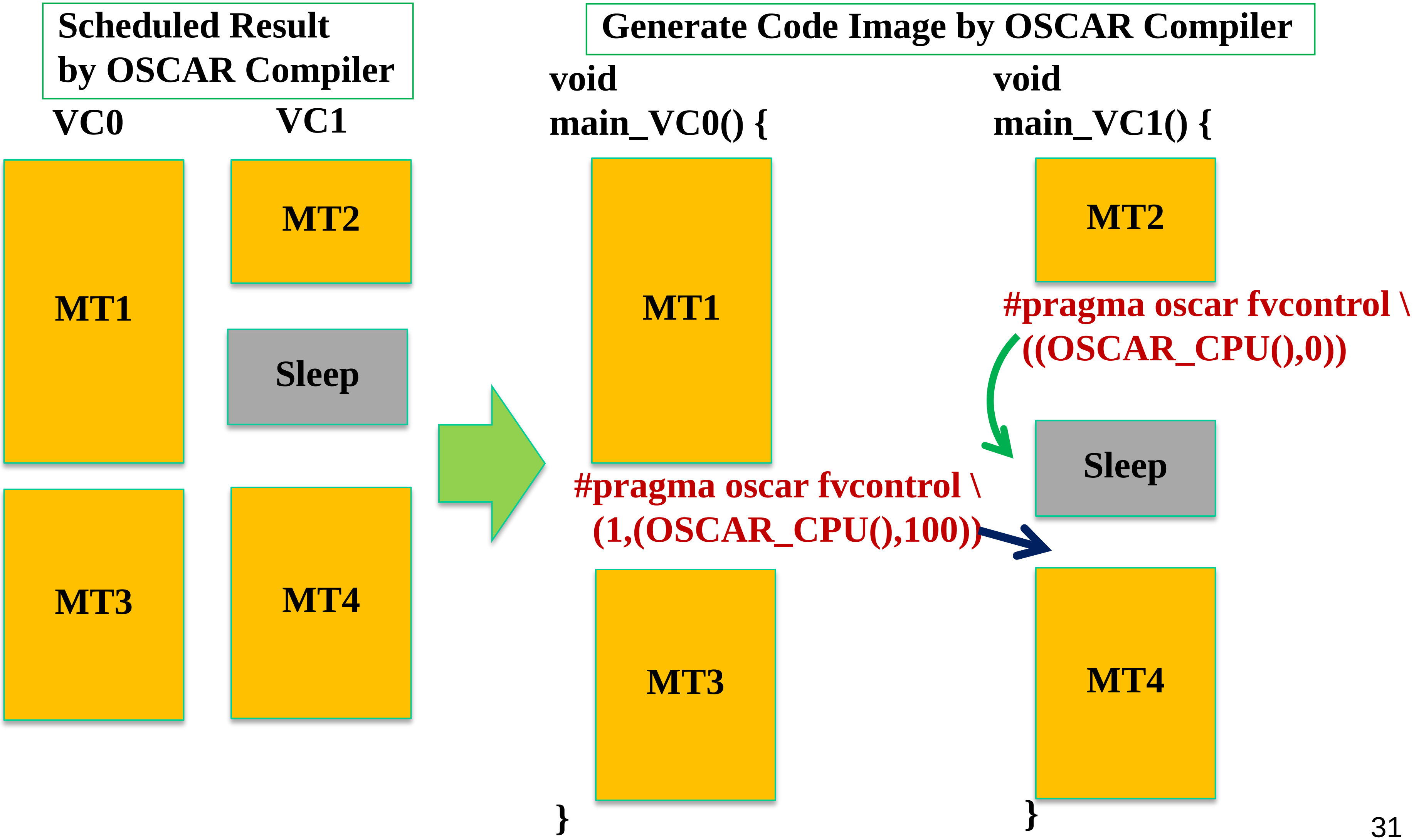
Average:0.54[W]





1cycle : 33[ms]
→30[fps]

Low-Power Optimization with OSCAR API



Scheduled Result by OSCAR Compiler

Generate Code Image by OSCAR Compiler

```
void
main_VC0() {
MT1
#pragma oscar fvcontrol \
 (1,(OSCAR_CPU(),100))
MT3
}
```

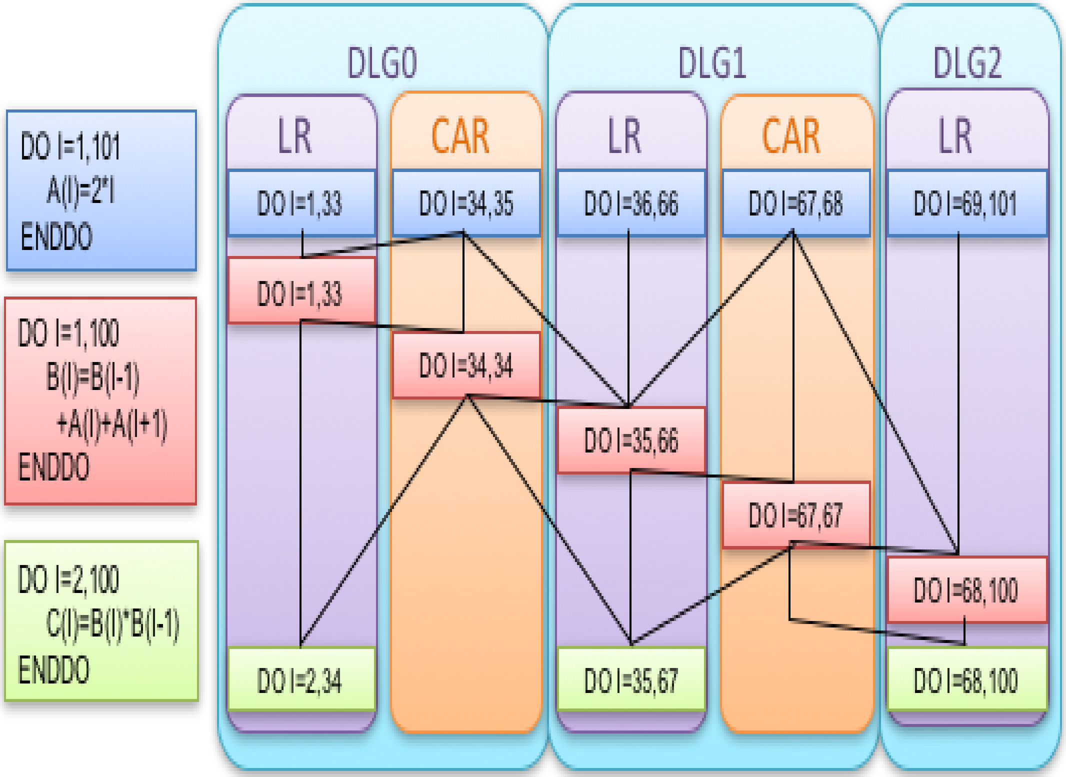
```
void
main_VC1() {
MT2
#pragma oscar fvcontrol \
 ((OSCAR_CPU(),0))
Sleep
MT4
}
```

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

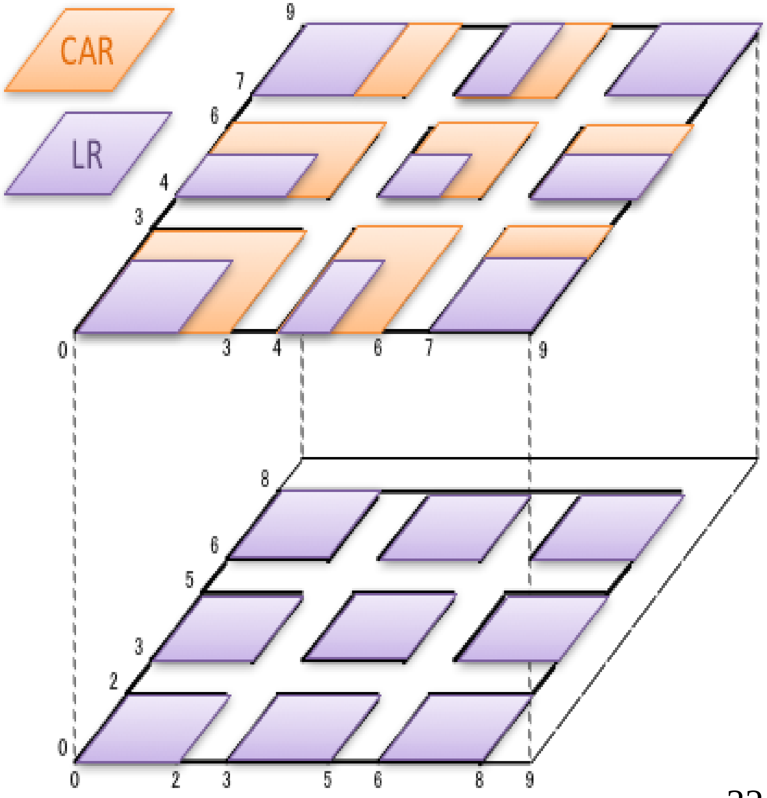


```
DO I=1,101
  A(I)=2*I
ENDDO
```

```
DO I=1,100
  B(I)=B(I-1)
   +A(I)+A(I+1)
ENDDO
```

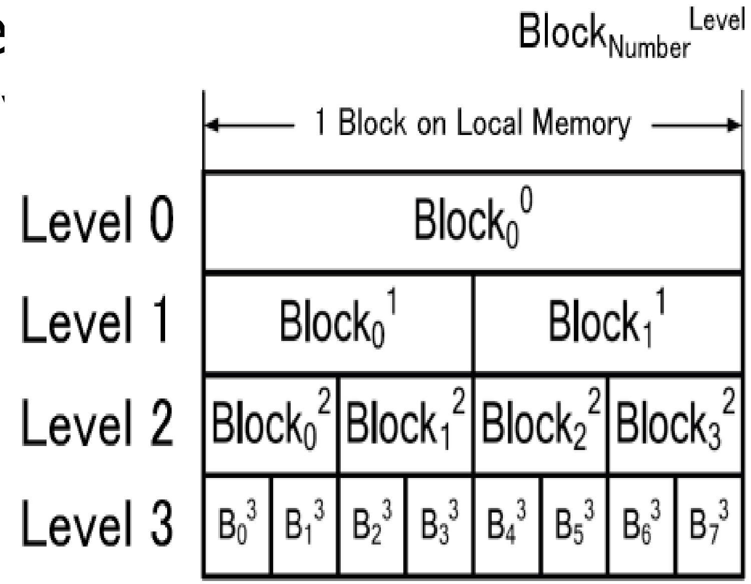
```
DO I=2,100
  C(I)=B(I)*B(I-1)
ENDDO
```

Multi-dimension Decomposition



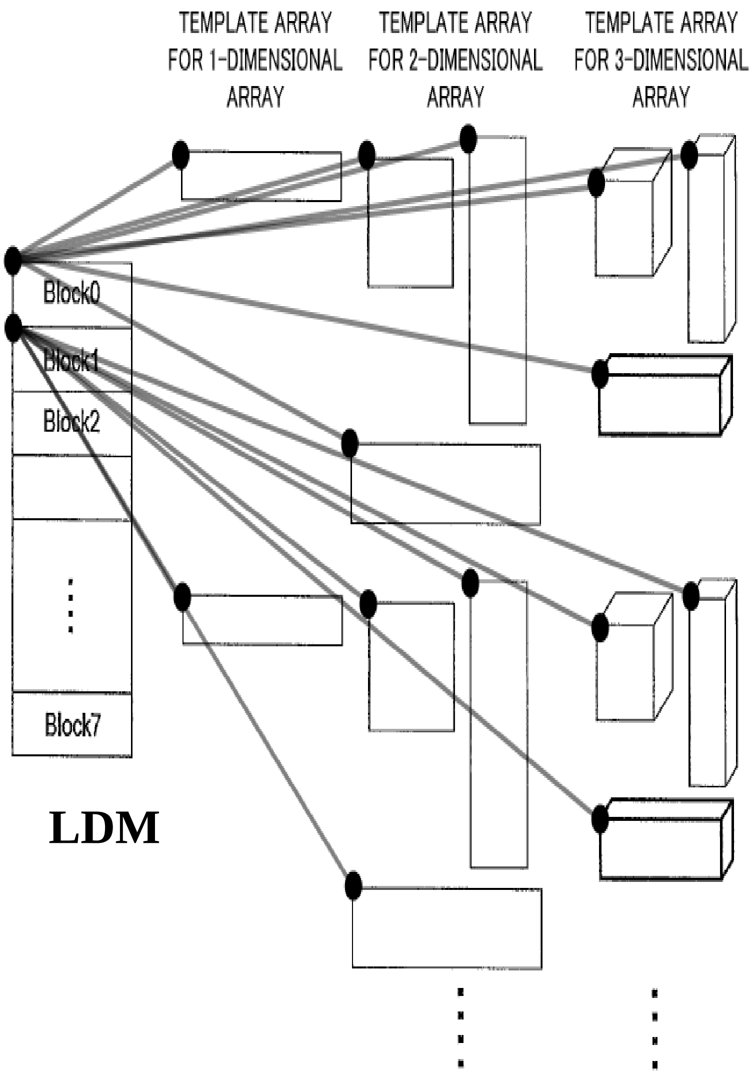
Adjustable Blocks

- 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 small arrays
    and scalar

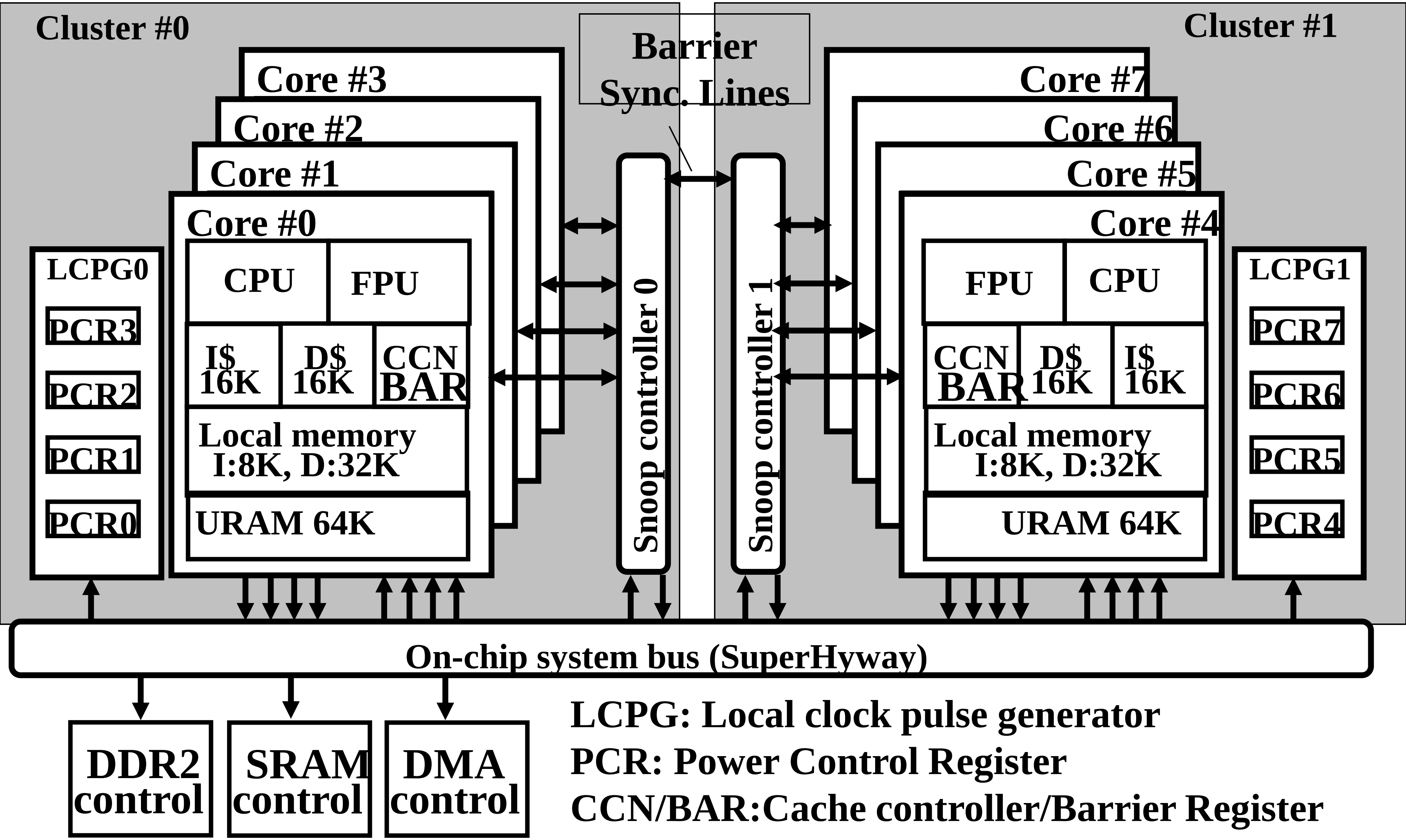


Multi-dimensional Template Arrays for Improving Readability

- 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’]


TEMPLATE ARRAY FOR 1-DIMENSIONAL ARRAY
TEMPLATE ARRAY FOR 2-DIMENSIONAL ARRAY
TEMPLATE ARRAY FOR 3-DIMENSIONAL ARRAY
Block0
Block1
Block2
Block7
LDM

8 Core RP2 Chip Block Diagram



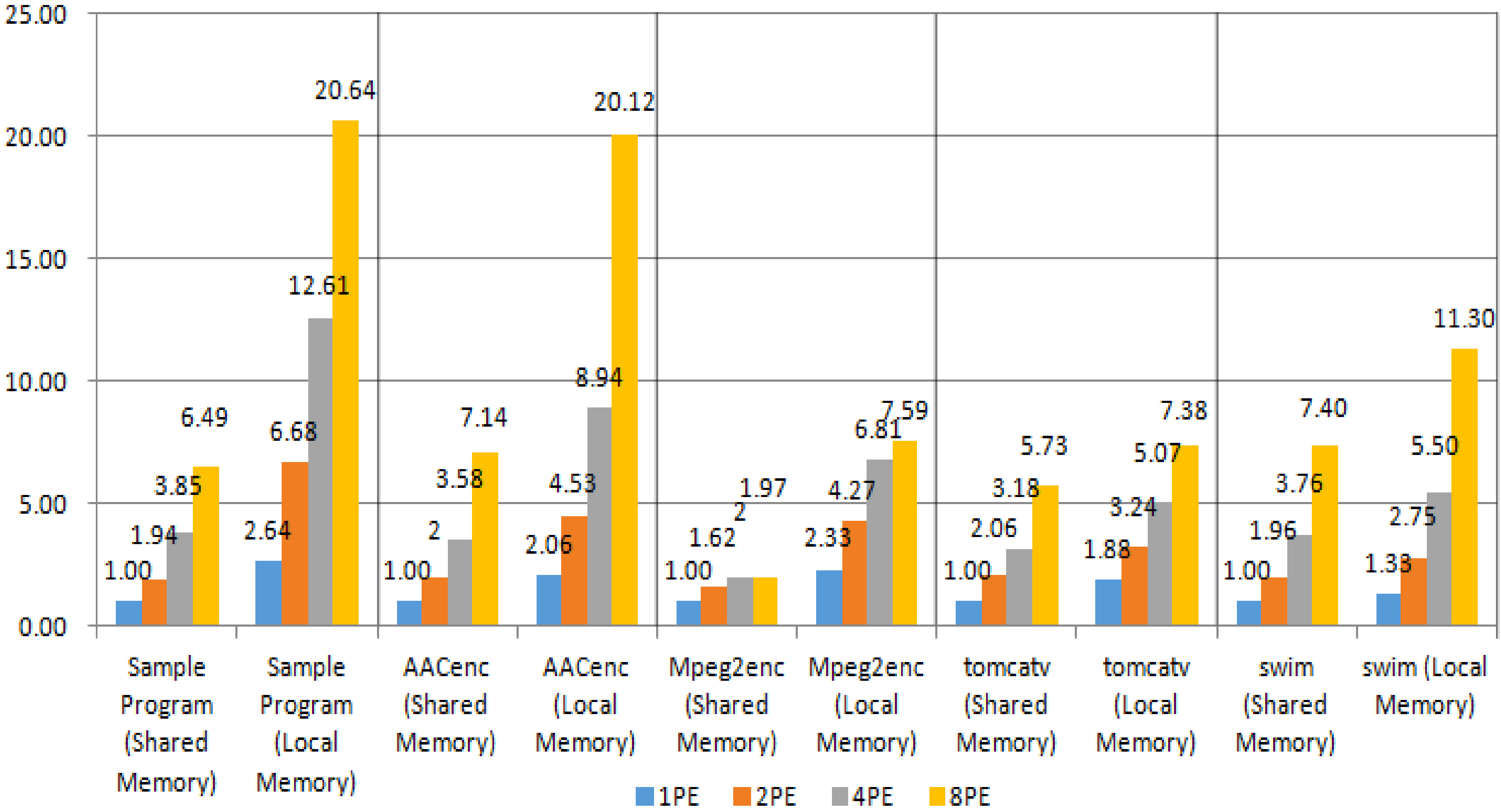
LCPG: Local clock pulse generator

PCR: Power Control Register

CCN/BAR:Cache controller/Barrier Register

URAM: User RAM (Distributed Shared Memory)

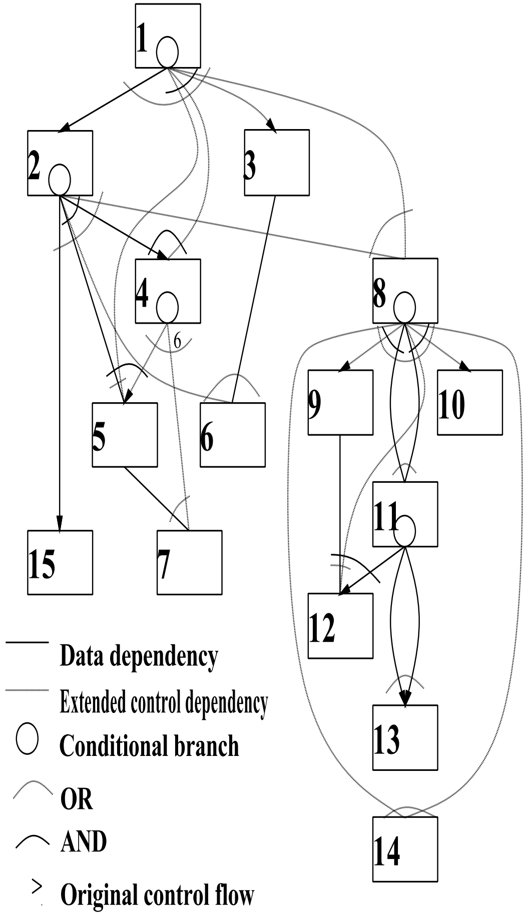
Speedups by the Proposed Local Memory Management Compared with Utilizing Shared Memory on Benchmarks Application using RP2



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

Software Coherence Control Method on OSCAR Parallelizing Compiler

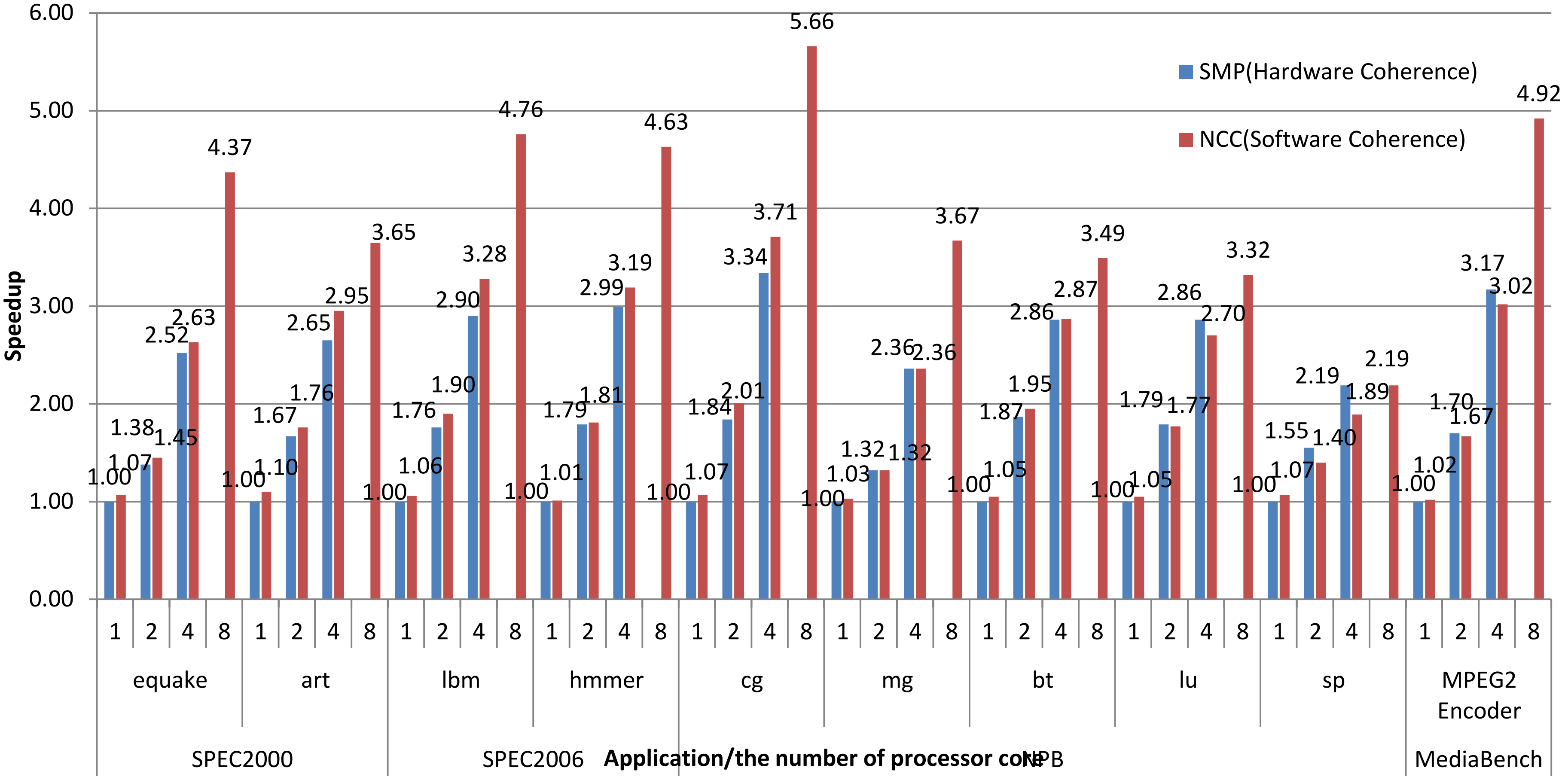
- Coarse grain task parallelization with earliest condition analysis (control and data dependency analysis to detect parallelism among coarse grain tasks).
- OSCAR compiler automatically controls coherence using following simple program restructuring methods:
  - To cope with stale data problems:
  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



MTG generated by earliest executable condition analysis

Automatic Software Coherent Control for Manycores

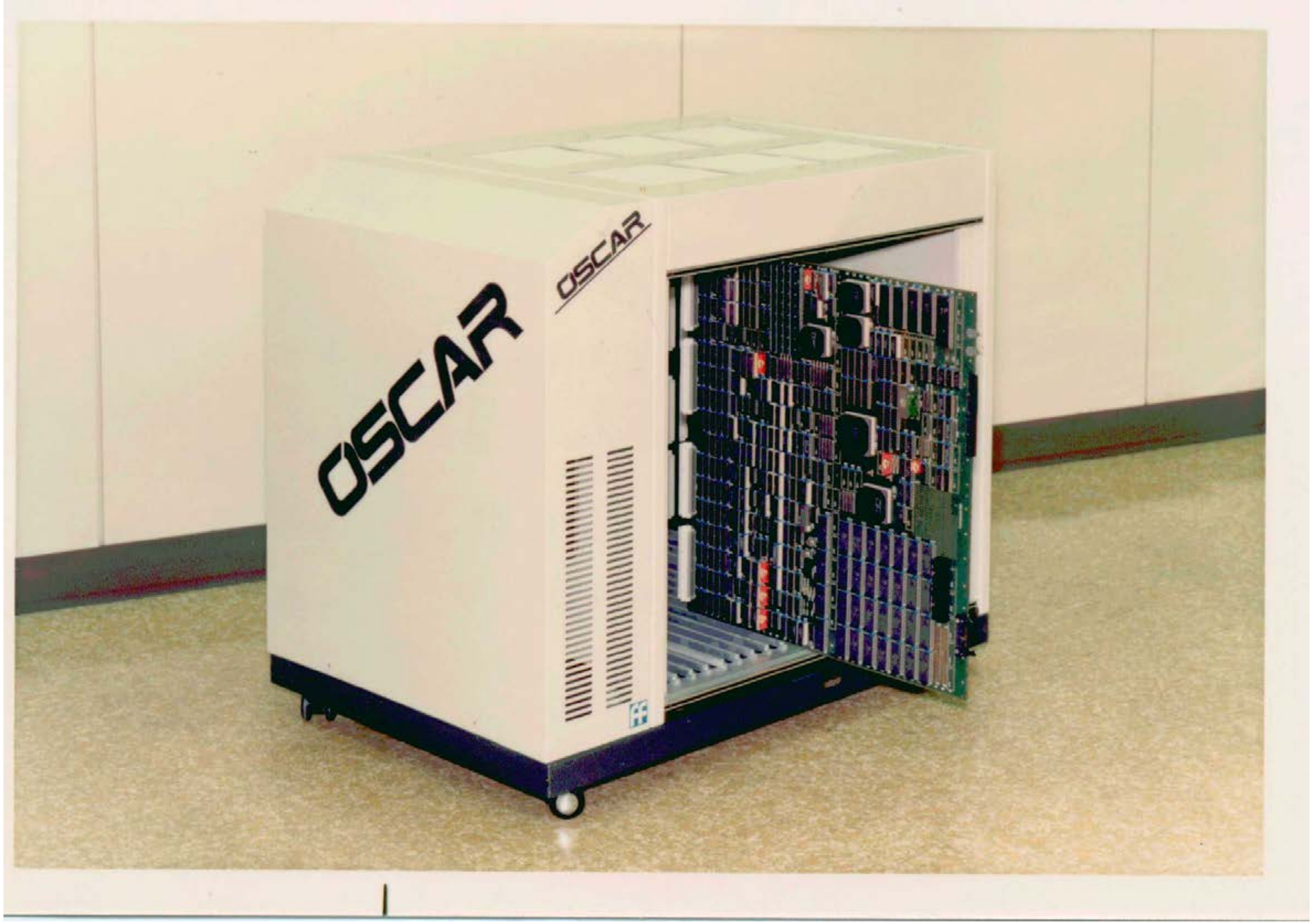
Performance of Software Coherence Control by OSCAR Compiler on 8-core RP2



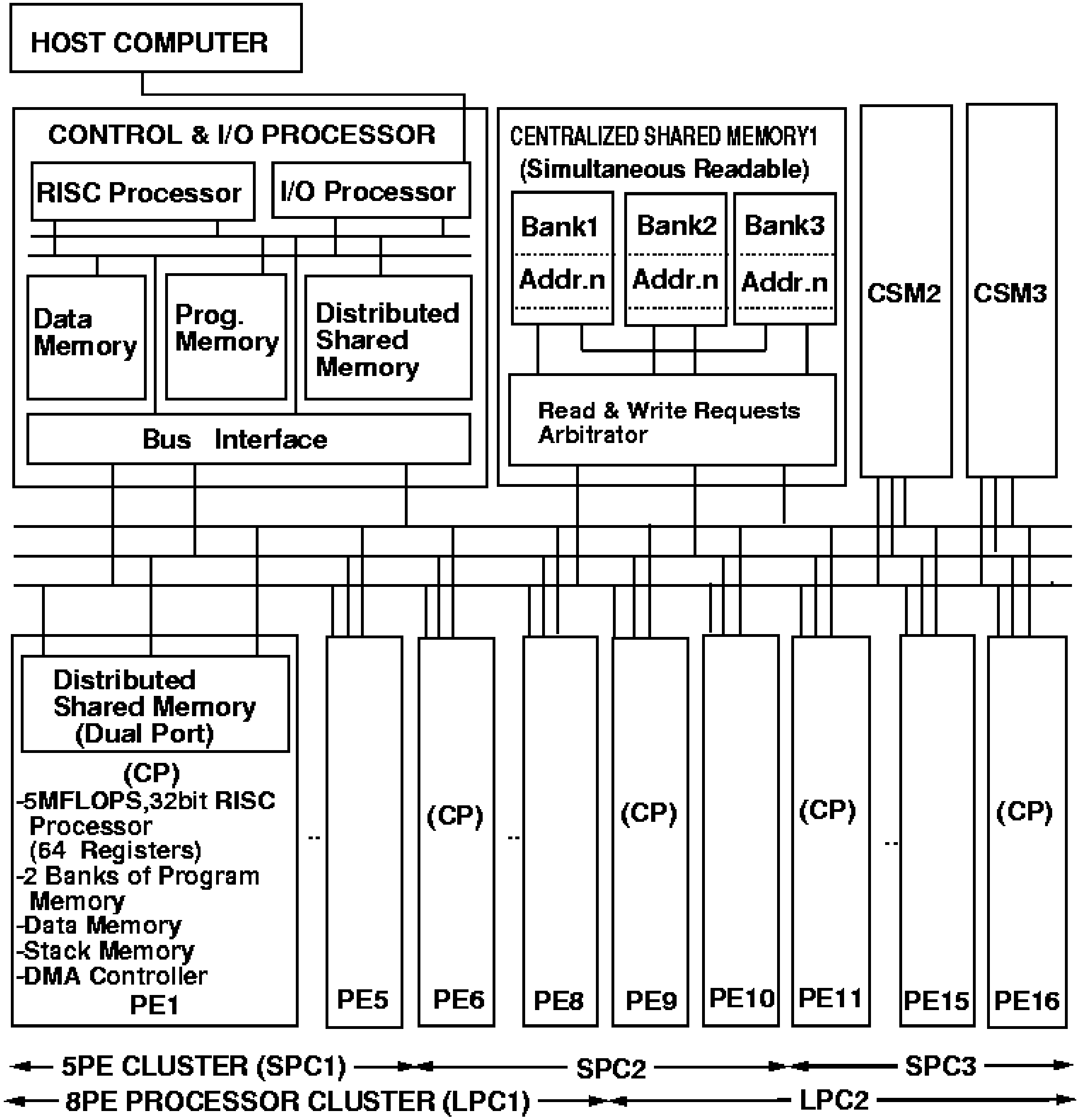
1987 OSCAR(Optimally Scheduled Advanced Multiprocessor)

Co-design of Compiler and Architecture

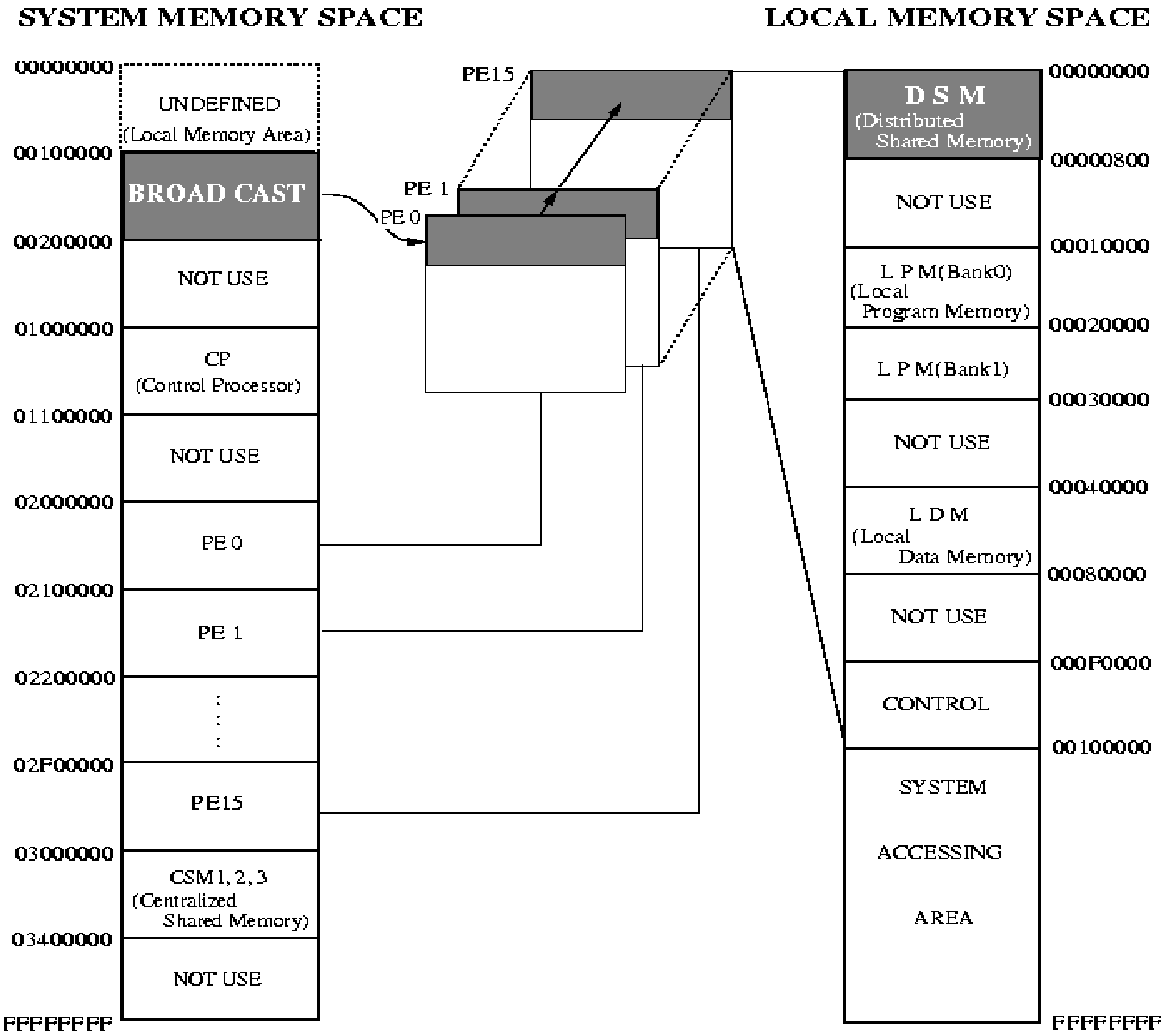
Looking at various applications, design a parallelizing compiler and design a multiprocessor/multicore-processor to support compiler optimization



OSCAR(Optimally Scheduled Advanced Multiprocessor)

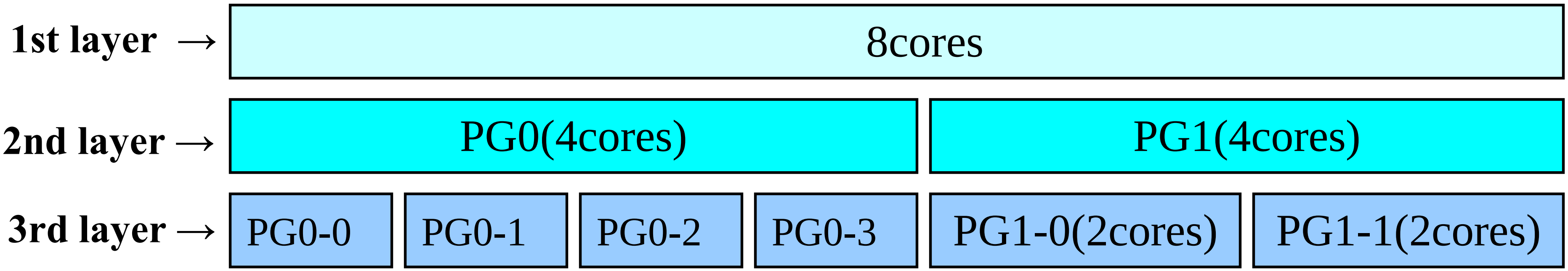


OSCAR Memory Space (Global Address Space)



Hierarchical Barrier Synchronization

- Specifying **a hierarchical group barrier**
  - #pragma oscar group_barrier (C)
  - !$oscar group_barrier (Fortran)

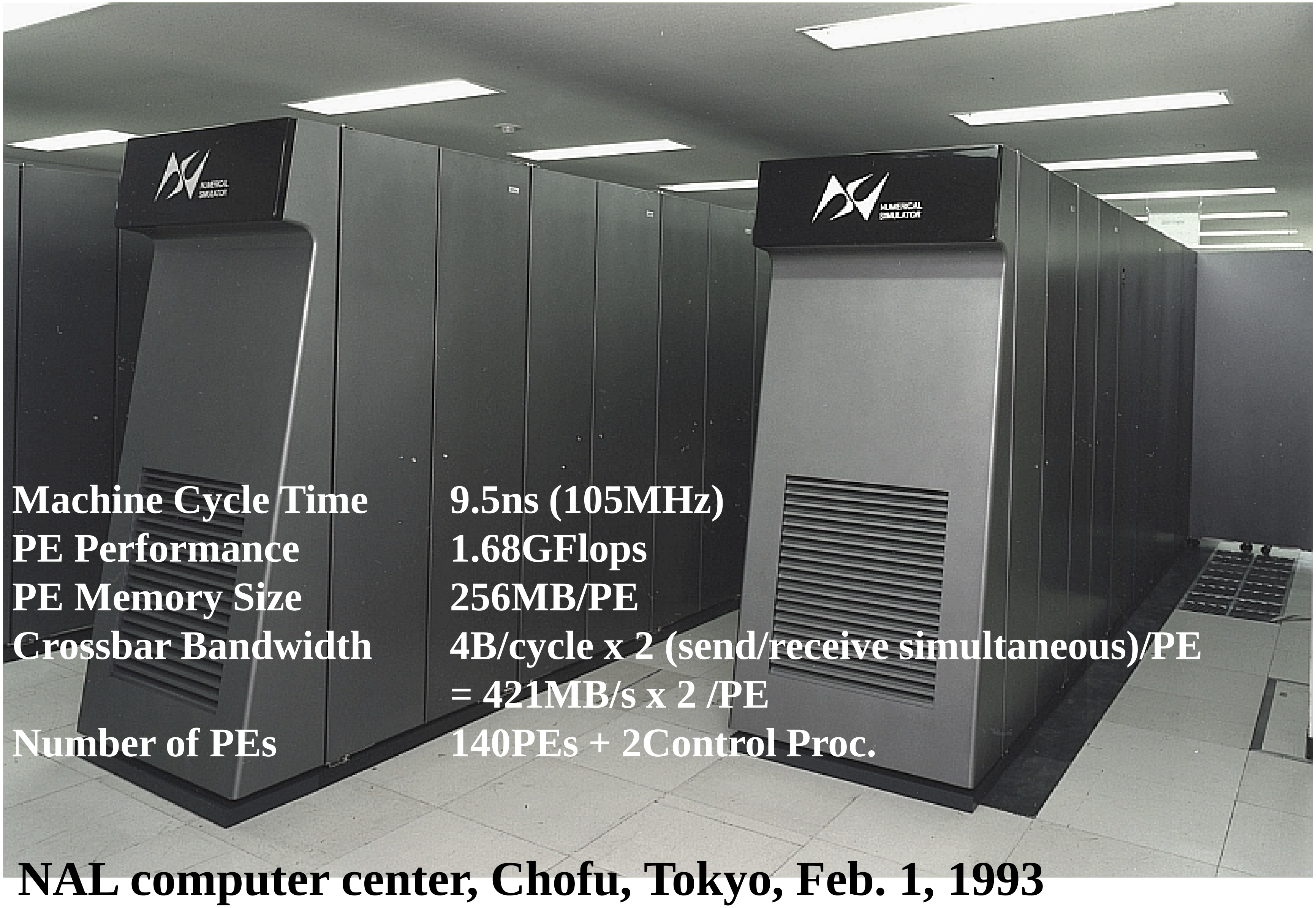


1st layer → 8cores

2nd layer → PG0(4cores) PG1(4cores)

3rd layer → PG0-0 PG0-1 PG0-2 PG0-3 PG1-0(2cores) PG1-1(2cores)

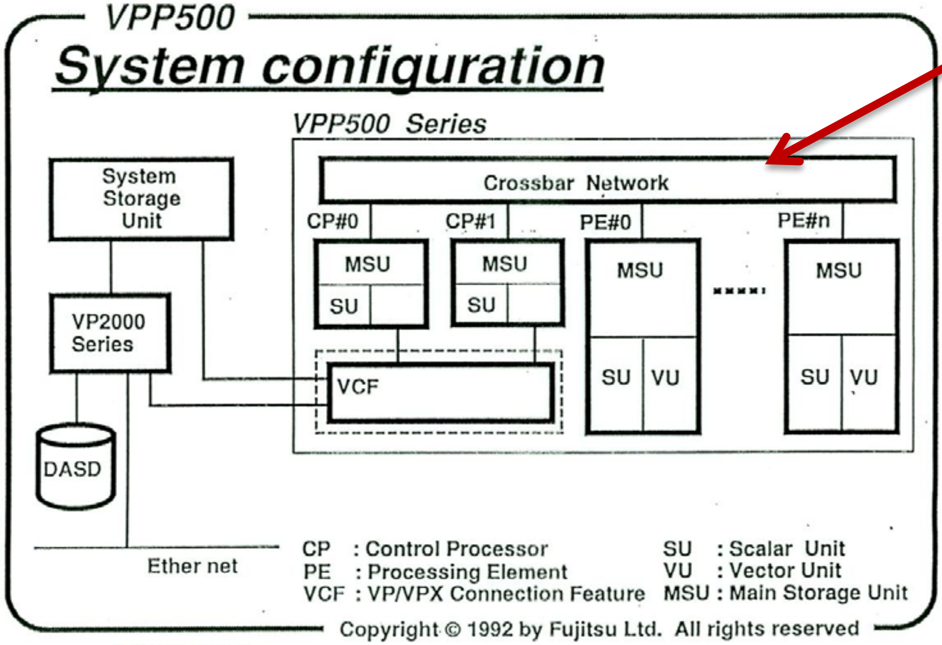
NWT



Machine Cycle Time	9.5ns (105MHz)
PE Performance	1.68GFlops
PE Memory Size	256MB/PE
Crossbar Bandwidth	4B/cycle x 2 (send/receive simultaneous)/PE = 421MB/s x 2 /PE
Number of PEs	140PEs + 2Control Proc.

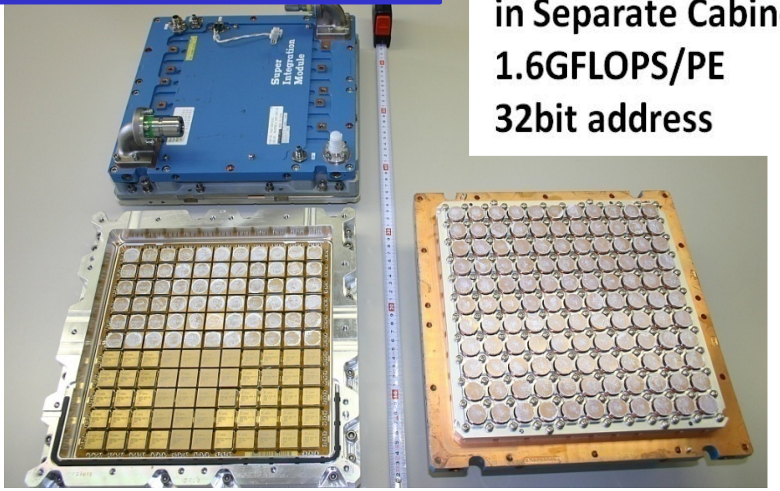
NAL computer center, Chofu, Tokyo, Feb. 1, 1993

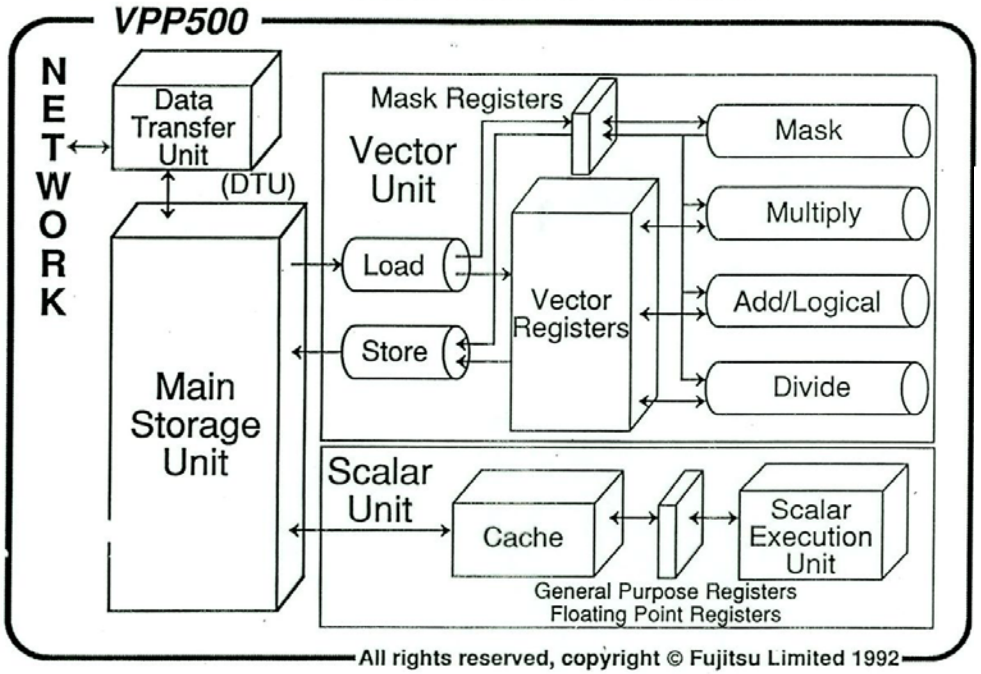
Fujitsu Vector Parallel Supercomputer with Crossbar to a Chip



Crossbar Network
VPP500: 256*256
VPP5000:
1024*1024

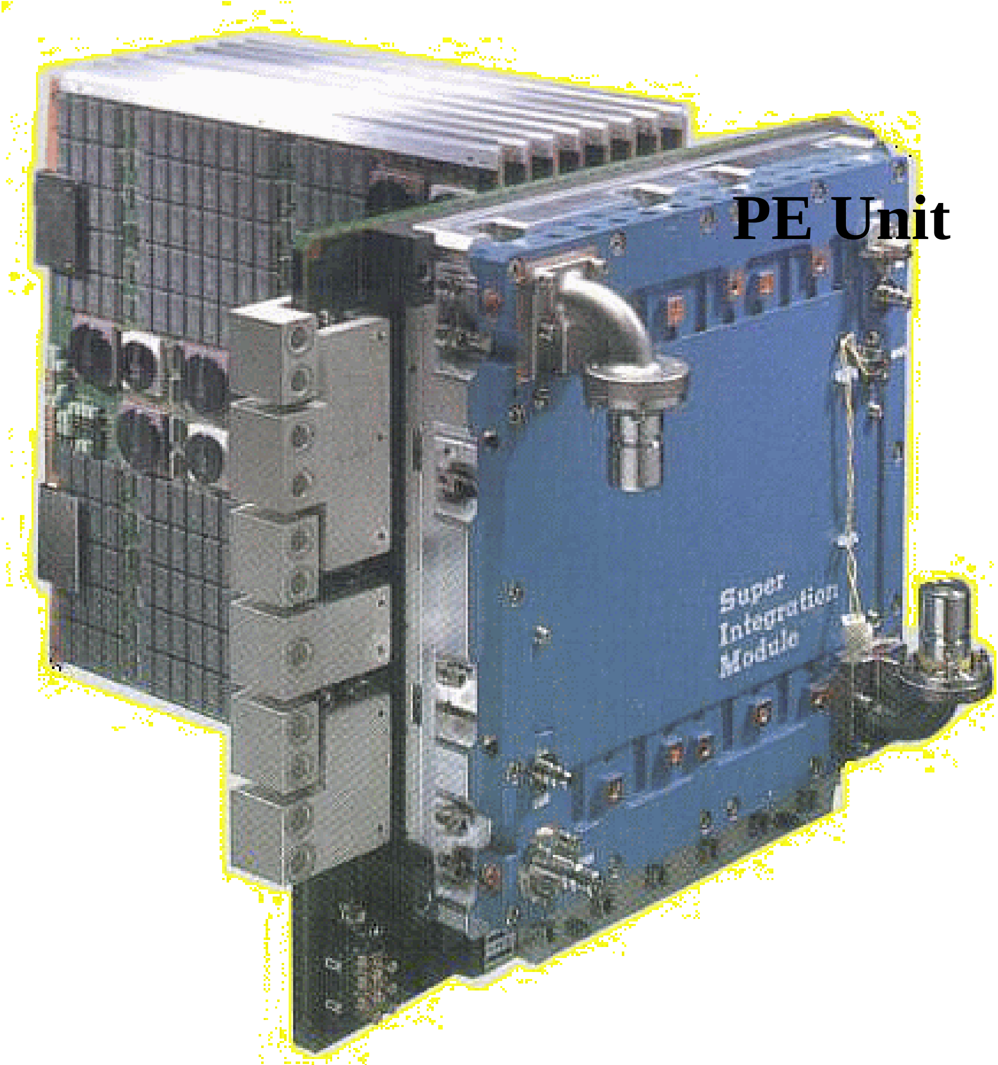
VPP500
GaAs/BiCMOS/ECL
Water Cooling
10ns
1GB/PE
in Separate Cabinet
1.6GFLOPS/PE
32bit address



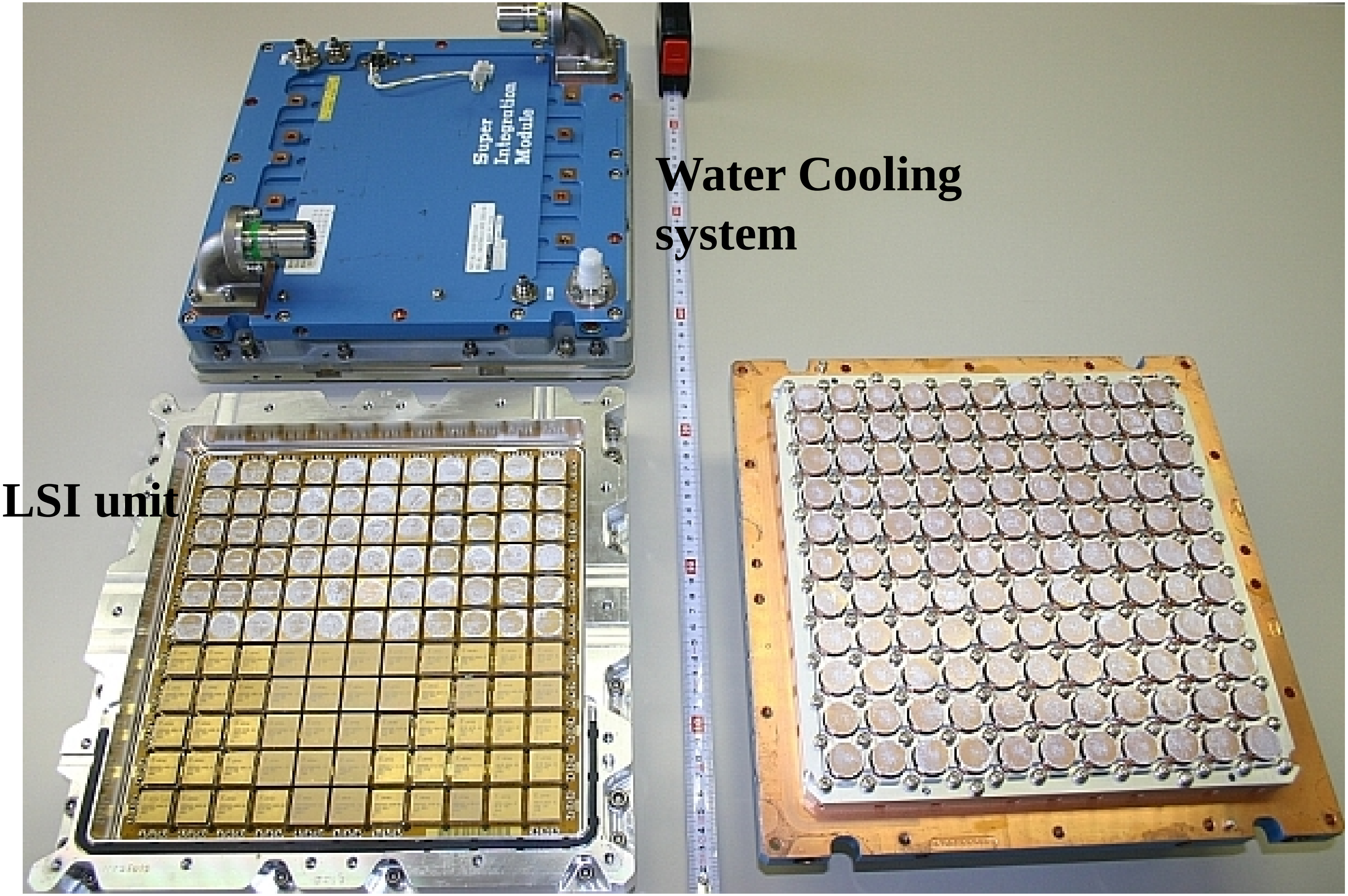


VPP5000
0.25um CMOS
30M Tr./Chip
Forced Air Cooling
3.5ns clock cycle
8GB/PE
on PE Board
9.1GFLOPS/PE
32/64 bit address

VPP500/NWT



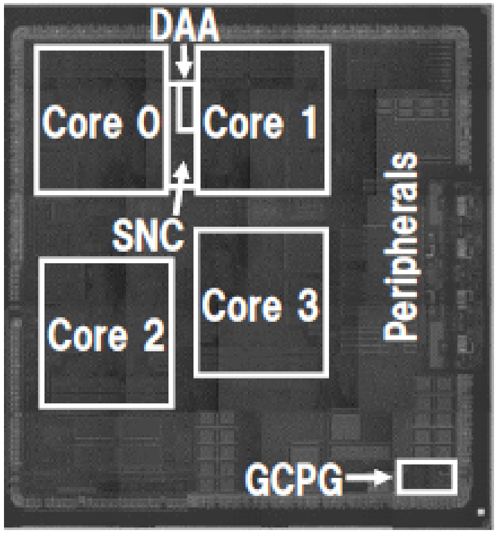
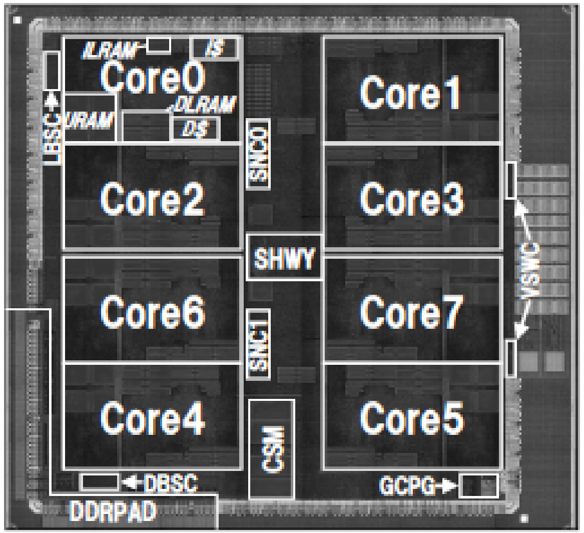
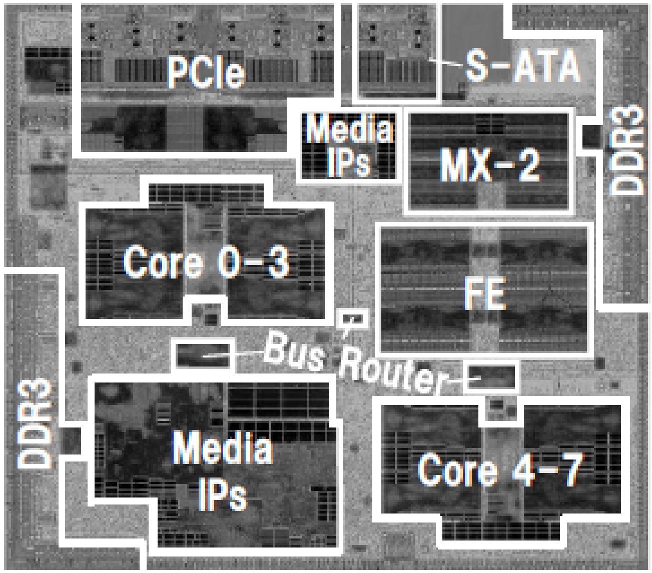
VPP500/NWT



Water Cooling system

PE LSI unit

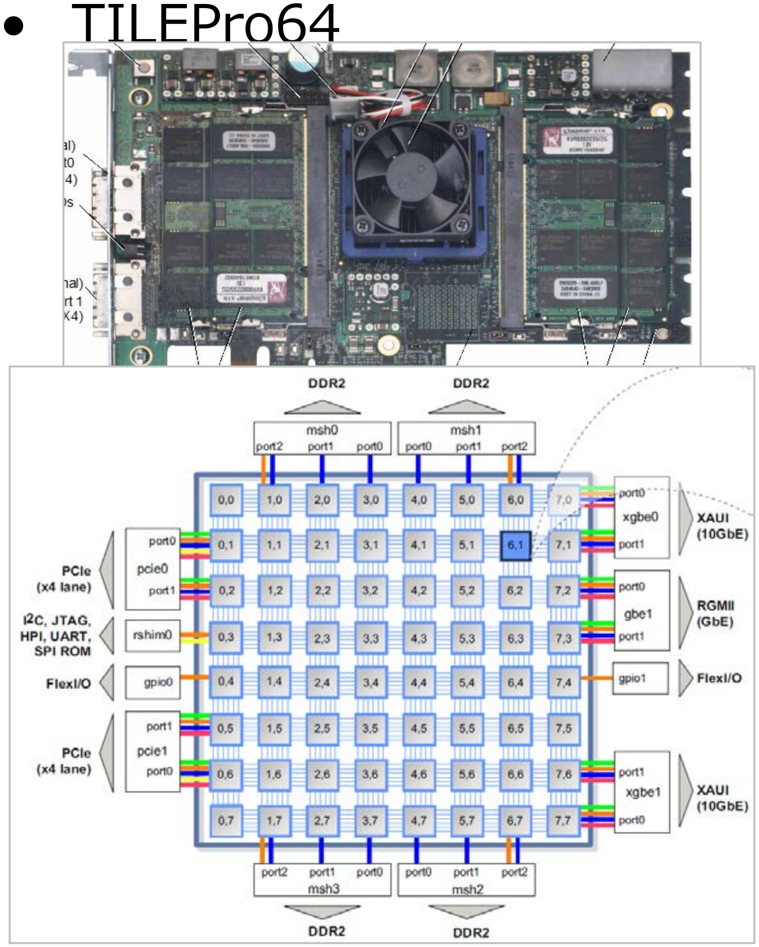
4 core multicore RP1 (2007) , 8 core multicore RP2 (2008) and 15 core Heterogeneous multicore RPX (2010) developed in NEDO Projects with Hitachi and Renesas

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）

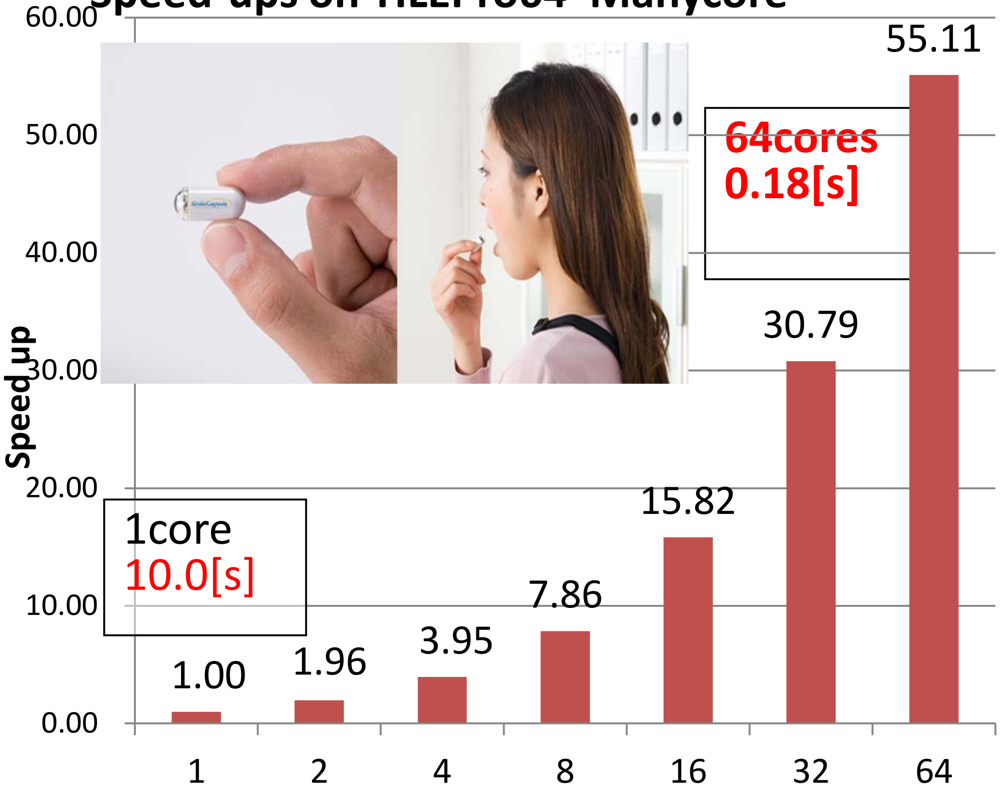
Automatic Parallelization of JPEG-XR for Drinkable Inner Camera (Endo Capsule)

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

- TILEPro64



Speed-ups on TILEPro64 Manycore

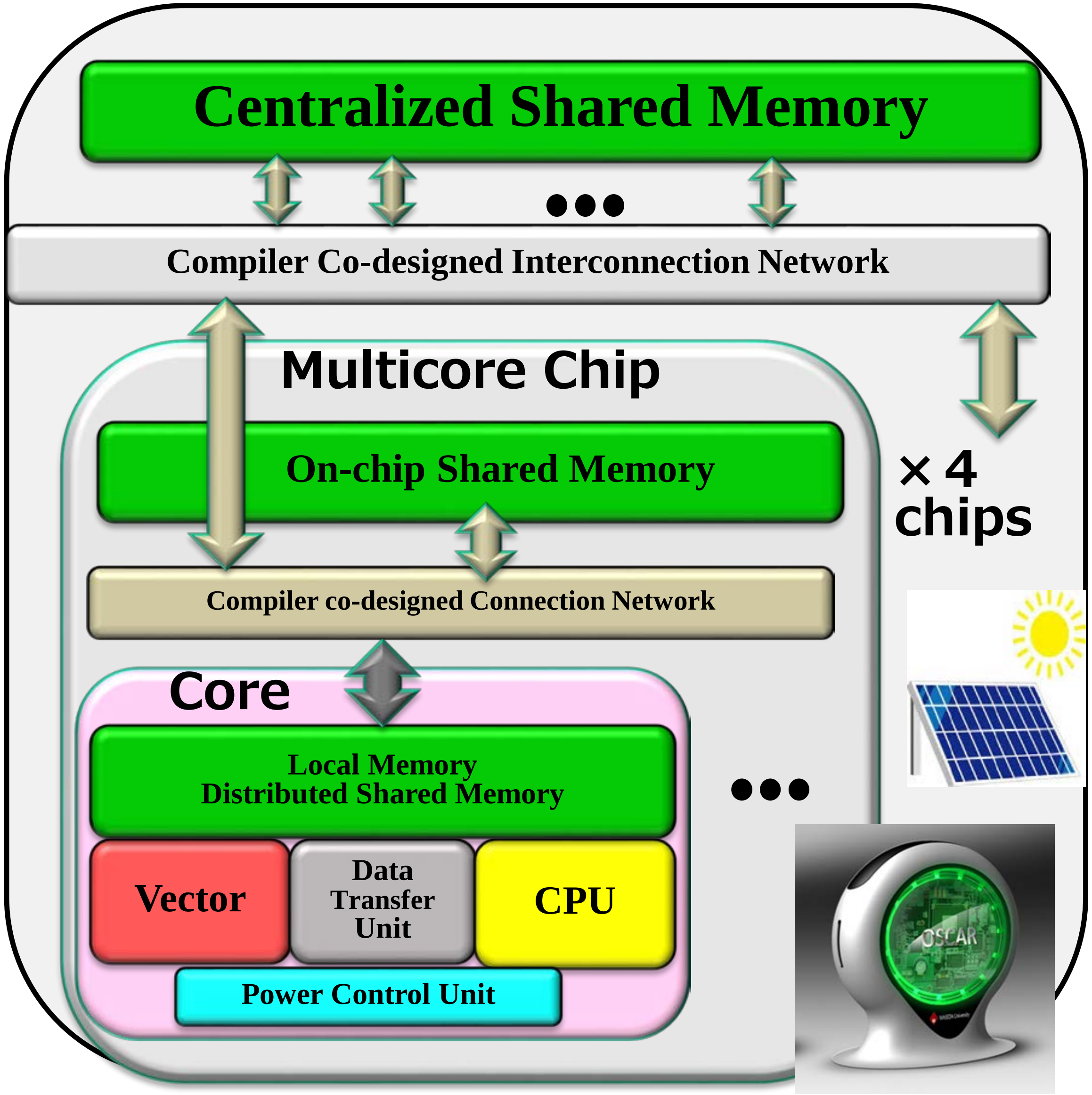


64cores
0.18[s]

1core
10.0[s]

55 times speedup with 64 cores

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



Centralized Shared Memory

●●●

Compiler Co-designed Interconnection Network

Multicore Chip

On-chip Shared Memory

Compiler co-designed Connection Network

Core

Local Memory
Distributed Shared Memory

Vector

Data Transfer Unit

CPU

Power Control Unit

●●●

× 4 chips

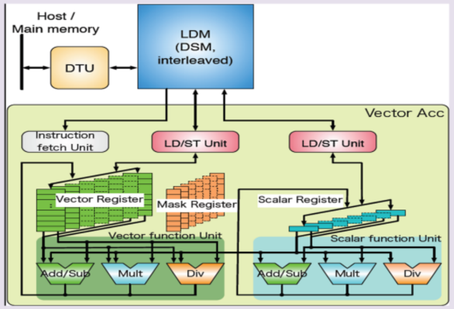
Target:

- 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

OSCAR Technology Corp.

Started up on Feb. 28, 2013:

Licensing the all patents and OSCAR compiler from Waseda Univ.

Founder and CEO: Dr. T. Ono (Ex- CEO of First Section-listed Company, Director of National U., Invited Prof. of Waseda U.)

Executives: Dr. M. Ohashi : COO (Ex- OO of Ono Sokki)

Mr. A. Nodomi : CTO (Ex- Spansion)

Mr. N. Ito (Ex- Visiting Prof. Tokyo Agricult. And Tech. U.)

Dr. K. Shirai (Ex- President of Waseda U., Ex- Chairman of Japanese Open U.)

Mr. K. Ashida (Ex- VP Sumitomo Trading, Adhida Consult. CEO)

Mr. S. Tsuchida (Co-Chief Investment Officer of Innovation Network Corp. of Japan)

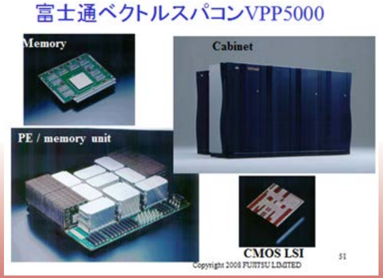
Auditor: Mr. S. Honda (Ex- Senior VP and General Manager of MUFG)

Dr. S. Matuda (Emeritus Prof. of Waseda U., Chairman of WERU INVESTME

Mr. Y. Hirowatari (President of AGS Consulting)

Advisors: Prof. H. Kasahara (Waseda U.)

Prof. K. Kimura (Waseda U.)



Future Multicore Products

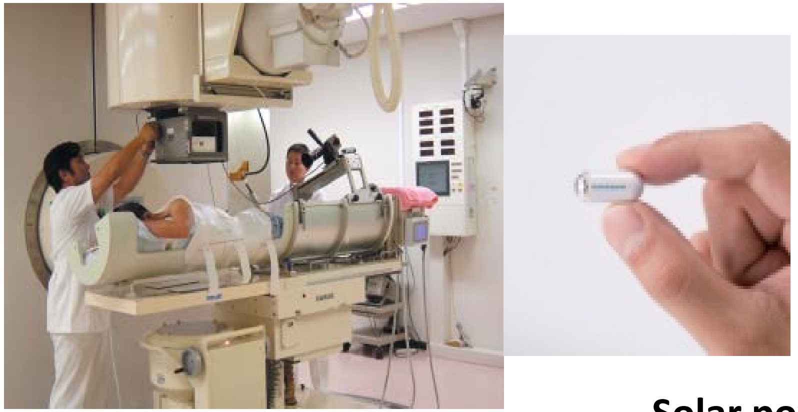
Next Generation Automobiles

- 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.