# Memory Optimization in OSCAR Parallelizing Compiler


Prof. Hironori Kasahara, IEEE Life-Fellow, IPSJ Fellow
Past Senior Executive Vice President, Waseda University
IEEE Computer Society President 2018
Board Member: The Academy of Engineering of Japan & COCN
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., 1989-90 Research Scholar: U. of Illinois, Urbana-Champaign, Center for Supercomputing R&D, 1997 Prof.,
2004 Director, Advanced Multicore Research Institute,
2017member: the Engineering Academy of Japan (2020- Board Mem) and the Science Council of Japan
2018 IEEE Computer Society President
Senior Vice President, Waseda Univ. (2018 Nov.-2022 Sept.)

AWARD: 1987 IFAC World Congress Young Author Prize
1997 IPSJ Sakai Special Research Award,
2005 STARC Academia-Industry Research Award,
2008 LSI of the Year Second Prize,
2008 Intel Asia Academic 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 Spirit of IEEE Computer Society Award,
2020 IPSJ Contribution Award,

Reviewed Papers: 232, Invited Talks: 230,
Granted Patents: 67 (Japan, US, GB, DE, China),
Articles in News Papers, Web News, TV etc.: 697

Committees in Societies and Government 287
IEEE Computer Society: President 2018, Executive Committee(2017-2019), BoG(2009-14), Strategic Planning Committee Chair 2018, Multicore STC Chair (2012-), Japan Chair(2005-07),
IPSJ Chair: HG for Magazine. & 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
JST Moonshot Project G3 Robot & AI Vice Chair,
【COCN】Board Member in Council of Competitiveness Nippon, etc.

# 早稲田大学

USA President:
Bill Clinton

USA Ambassador:
Caroline Kennedy

Chinese President:
Hu Jintao

# WASEDA University


Tokyo, Japan

## 1882 》

Okuma Shigenobu founded Tokyo Semmon Gakko (College)

The "Group of Four" who contributed to the development of Waseda University

## 1903 》

Start of the Waseda-Keio baseball match (Jubilee)

## 《 1922

Visit by physicist Albert Einstein to Waseda University

Archaeological excavation of the Amarna site

## 1928 》

Japan's first gold medalist

## 1993 》

Visit to Waseda University by then U.S. President Bill Clinton

## 《 1940

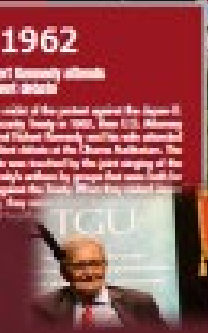
"Vows for life"
Free diplomat Chiune Sugihara

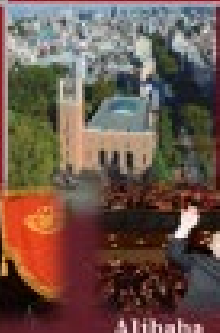
## 1956 》

The beginnings of the Ishibashi Cabinet, first alumnus of Waseda to become Prime Minister

## 《 1962

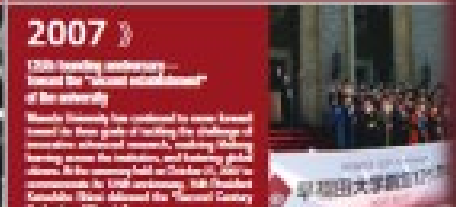
Robert Kennedy attends student awards





## 《 2012

Formulation of Waseda Vision 150



## 2007 》

150th founding anniversary... toward the "second establishment" of the university



Microsoft:
Dr. Bill Gates

British Prime Minister:
Boris Johnson

WASEDA UNIVERSITY



Alibaba
Mr. Jack Ma

# Waseda University
# 早稲田大学

## Number of International Students
7,942 from 125 countries and territories

## Alumni CEOs in Japan
10,606

9 Prime Ministers

Founder Shigenobu OKUMA

## Graduate Employability
#1 in private university of Japan
(#2 in Japan, #27 in the world)
QS Graduate Employability Rankings 2019

0% Schools

| ALUMNI [卒業生] | PARTNER INSTITUTIONS [協定大学・機関] |
|---|---|
| 630,000 | 848 (93 countries) |


Hiroshi YAMAUCHI | Masaru IBUKA | Tadashi YANAI

https://www.kantei.go.jp/jp/rekidaina/kaku/index.html

mercari | HEROZ JAPAN

| FACULTY [教員] | ENROLLMENT [学生数] | UNDERGRADUATE STUDENTS [学部生] | GRADUATE STUDENTS [大学院生] |
|---|---|---|---|
| 5,468 | 49,436 | 41,051 | 8,385 |

* Entire Year 2018

NUMBER OF BOOKS [図書館蔵書]
5,800,000

| Prime Ministers | Business Leaders | Business Leaders |
|---|---|---|
| | Founders of global companies | CEOs of global companies |
| 8th Shigenobu Okuma | | ANA (All Nippon Airways) |
| 17th Shigenobu Okuma | Sony | HONDA |
| 55th Tanzan Ishibashi | Samsung | Nintendo |
| 74th Noboru Takeshita | Casio | UNIQLO |
| 76th Toshiki Kaifu | LOTTE | Shiseido |
| 84th Keizo Obuchi | | Nomura Securities Co., Ltd. |
| 85th Yoshiro Mori | | Tokio Marine & Nichido Fire Insurance Co., Ltd. |
| 91st Yasuo Fukuda | | Olympus Corporation |
| 95th Yoshihiko Noda | | |
| 100th Fumio Kishida | | |

**Aiji TANAKA**
President International Political Science Association (IPSA) President 2016



**Hironori KASAHARA**
Senior Executive Vice President IEEE Computer Society President 2018. The first president from outside USA and Canada in 72 years CS history. CS has 84,000 members from 168 countries.

**Toshio FUKUDA**
The University Professor Waseda, Waseda Alumnus, Prof. Emeritus Nagoya Univ., Prof. Meijo Univ. IEEE President 2020. The first from Asia in 135 years history. IEEE has 420,000 members.

Haruki MURAKAMI

Hirokazu KOREEDA

Yuzuru HANYU

S. ARAKAWA

Yui Suzaki

# 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

IEEE COMPUTER SOCIETY
**BY THE NUMBERS**

**12,000+**
Volunteers

**615**
Committees/ Boards

**2,352+**
Meetings/ Teleconferences

**6**
New Standards

**230**
Active Standards

**IEEE754, 802**

**373,100+**
Community Members

**168**
Countries with CS Members

**634**
Chapters

# ACM/IEEE SC (SuperComputing) 19, Denver, Nov.17-22, 2019


Cornel Univ. Prof. Steven Squyres: Mars Exploration、CalTech. Dr. Katie Bouman: Visualization of Black Hole

# 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)

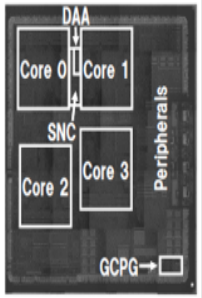
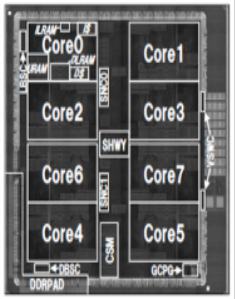
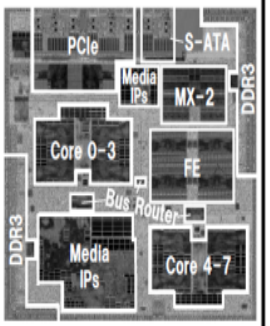
第74回総合科学技術会議の様子(2)


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)

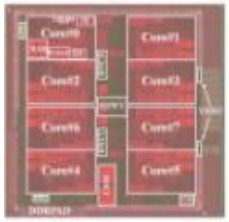
**Codesign of Compiler and Multiprocessor Architecture since 1985**

**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) |

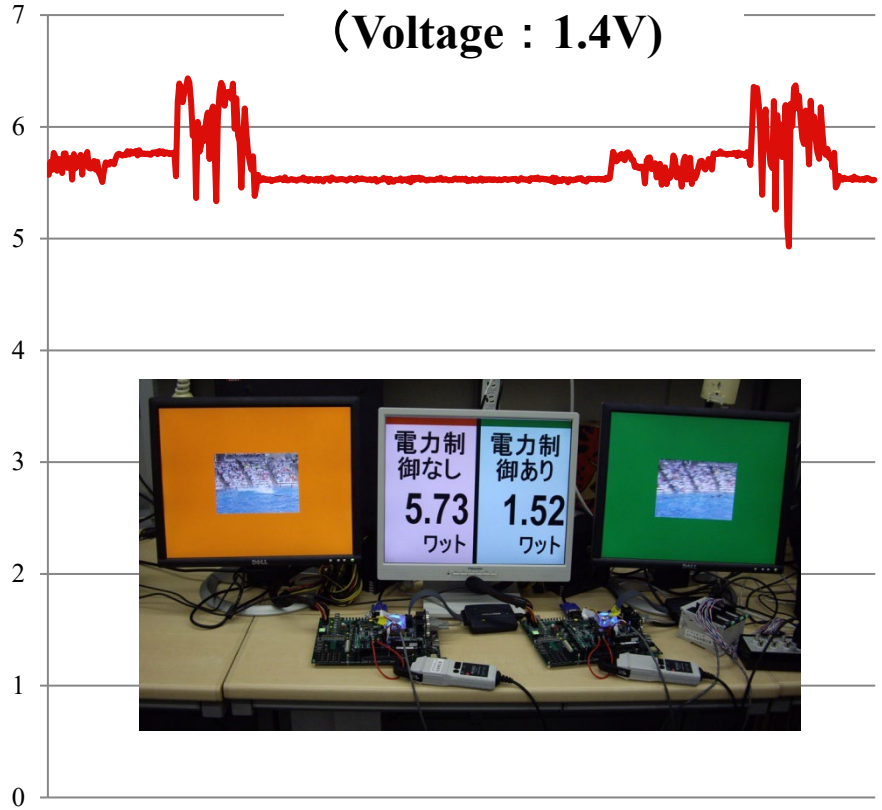
**Prime Minister FUKUDA is touching our multicore chip during execution.**

# 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

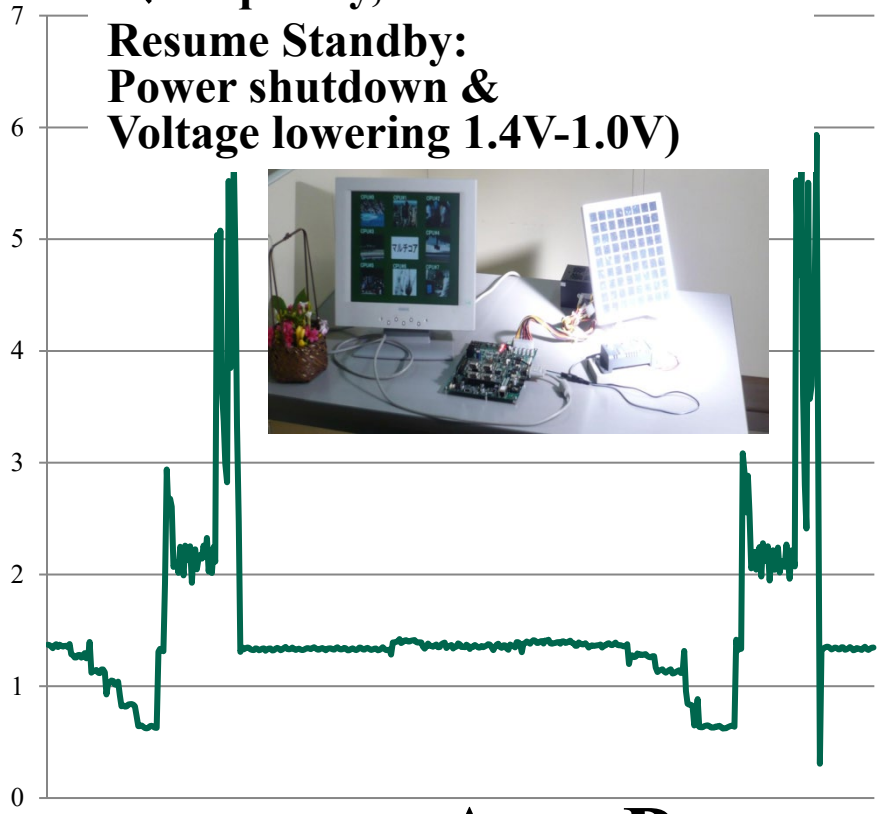
**Without Power Control (Voltage：1.4V)**







**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]**

# Green Computing Systems R&D Center

## Waseda University

**Established by Prof. Kasahara supported by METI (Mar. 2011)**

### ＜R & D Target＞

**Hardware, Software, Application for Super Low-Power Manycore**

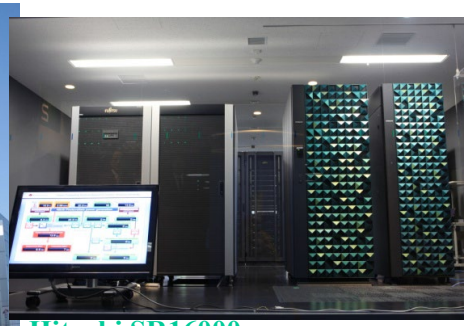
- **More than 64 cores**
- **Natural air cooling (No fan)**
  **Cool, Compact, Clear, Quiet**
- **Operational by Solar Panel**

### <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**

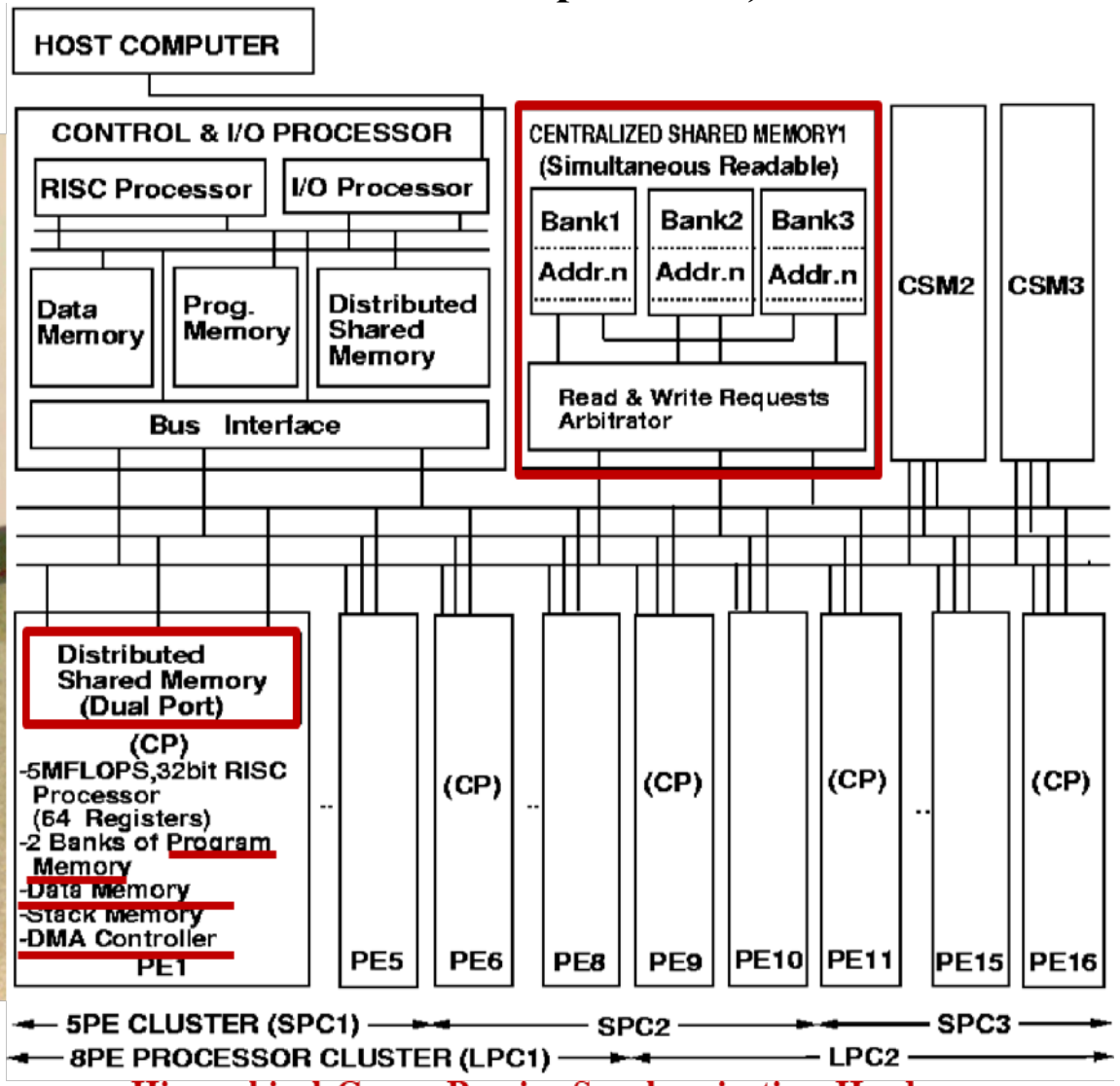


**Hitachi SR16000:**
**Power7 128coreSMP**
**Fujitsu M9000**
**SPARC VII 256 core SMP**

**Beside Subway Waseda Station, Near Waseda Univ. Main Campus**

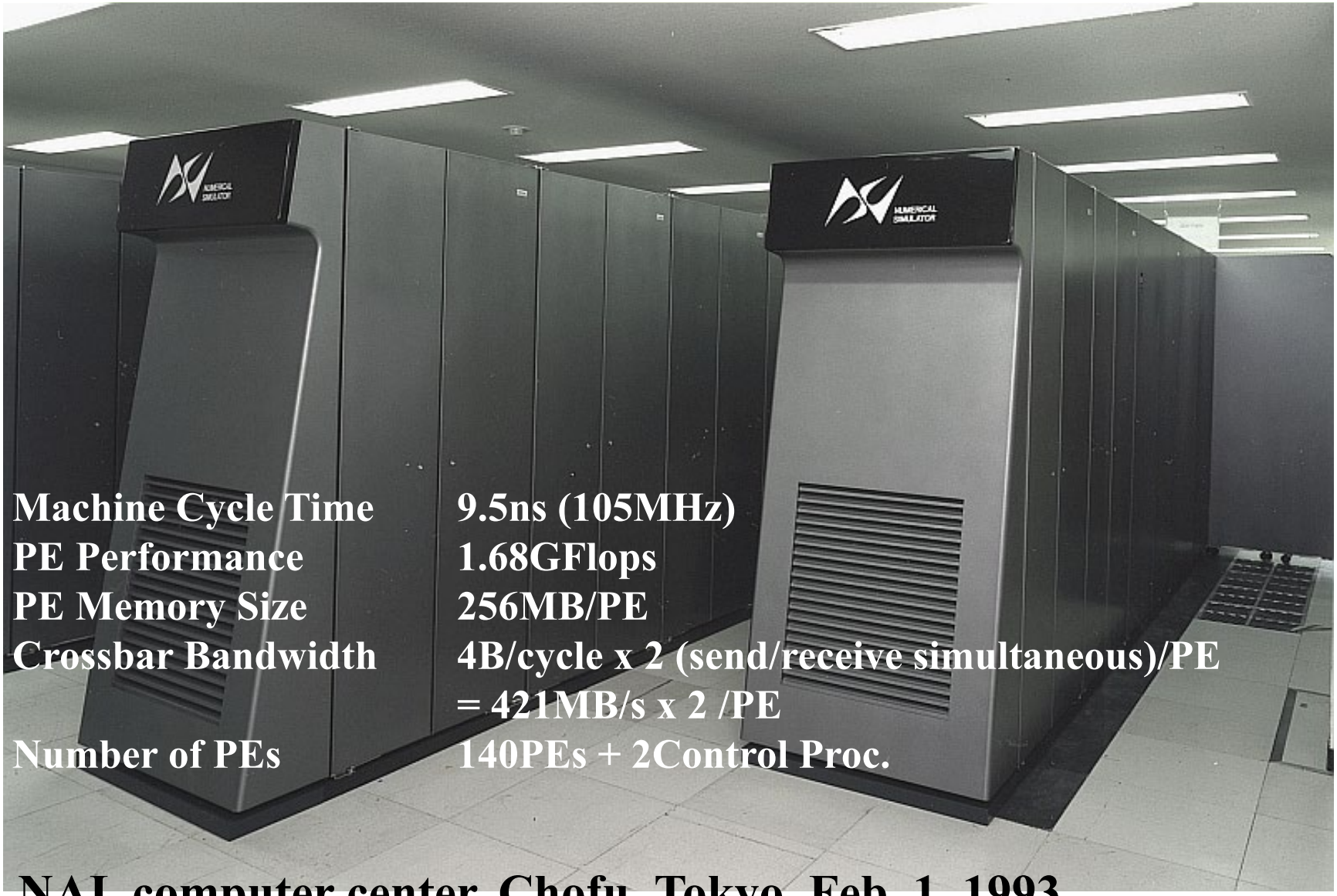
# The First Codesigned Architecture with Compiler

## OSCAR (Optimally Scheduled Advanced Multiprocessor) in 1987


Hierarchical Group Barrier Synchronization Hardware

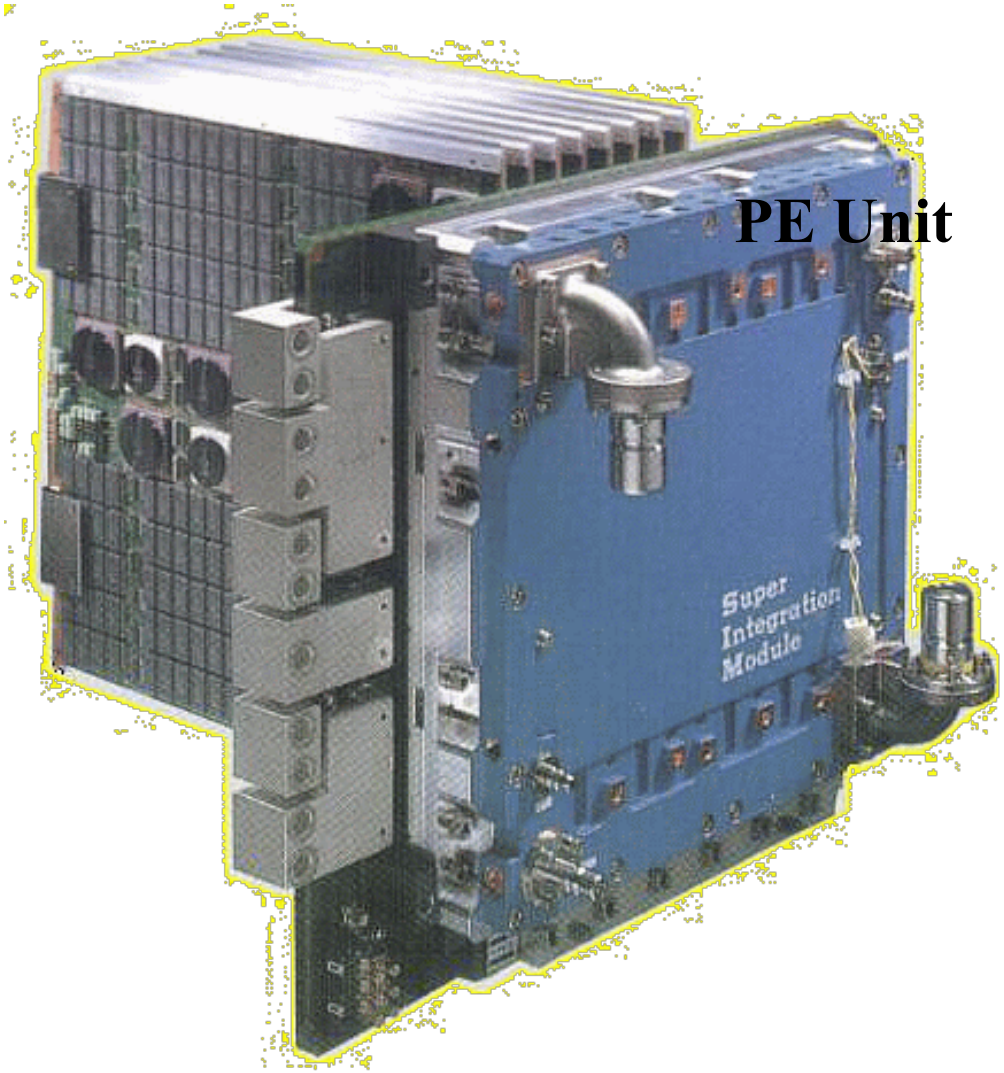
# NWT




| | |
|---|---|
| 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<br>= 421MB/s x 2 /PE |
| Number of PEs | 140PEs + 2Control Proc. |

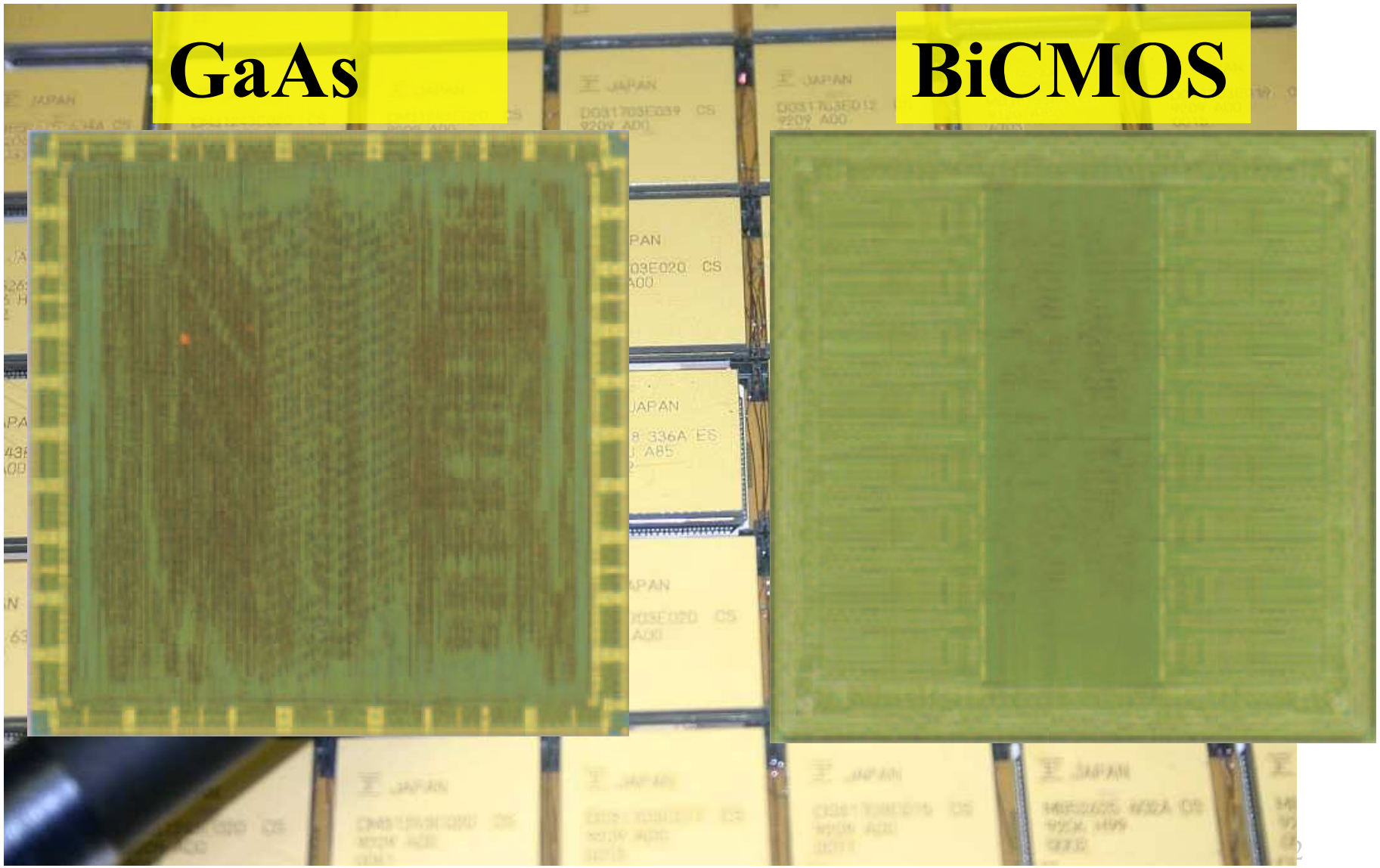
**NAL computer center, Chofu, Tokyo, Feb. 1, 1993**

# VPP500/NWT

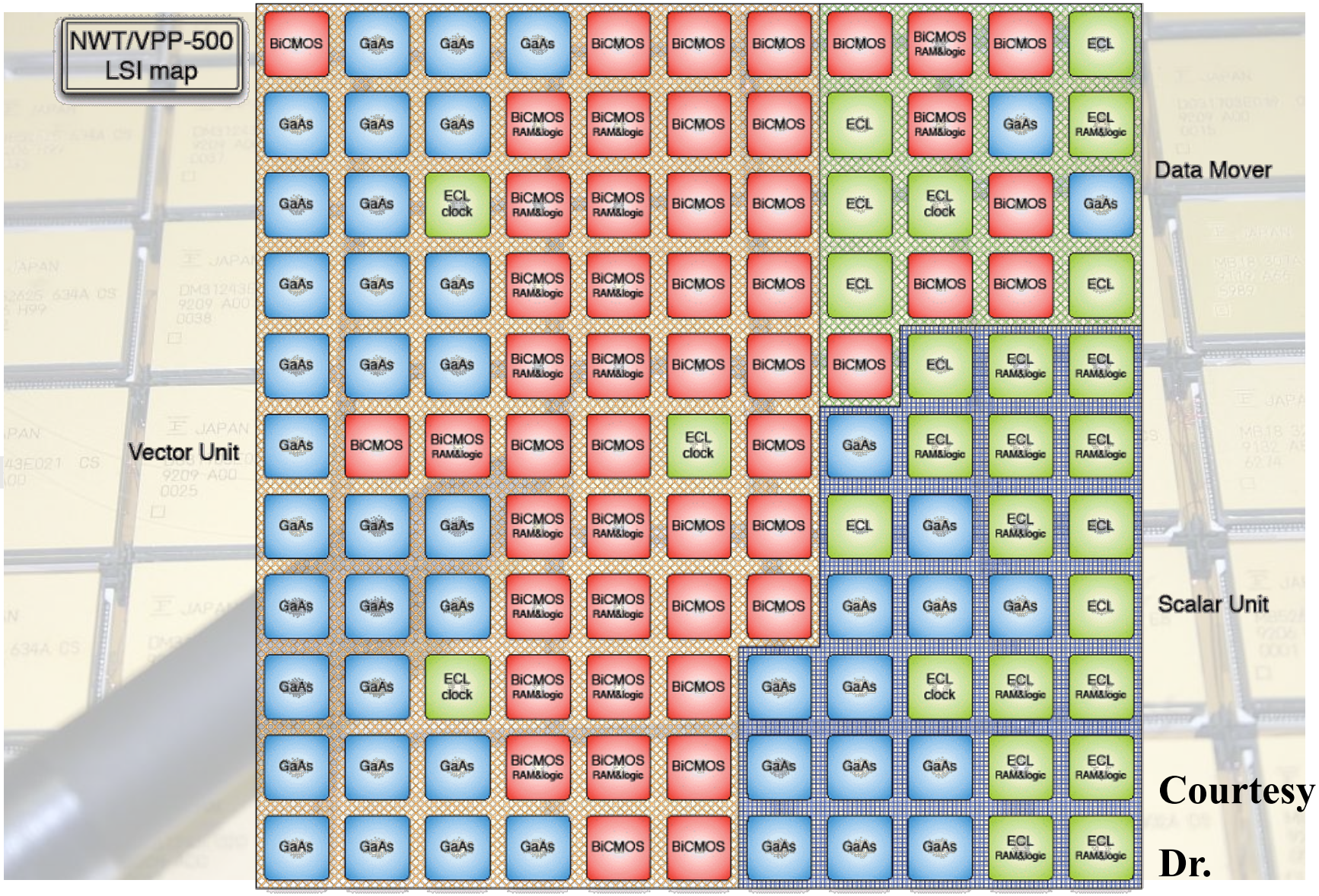



**PE Unit**

# VPP500/NWT



# VPP500/NWT

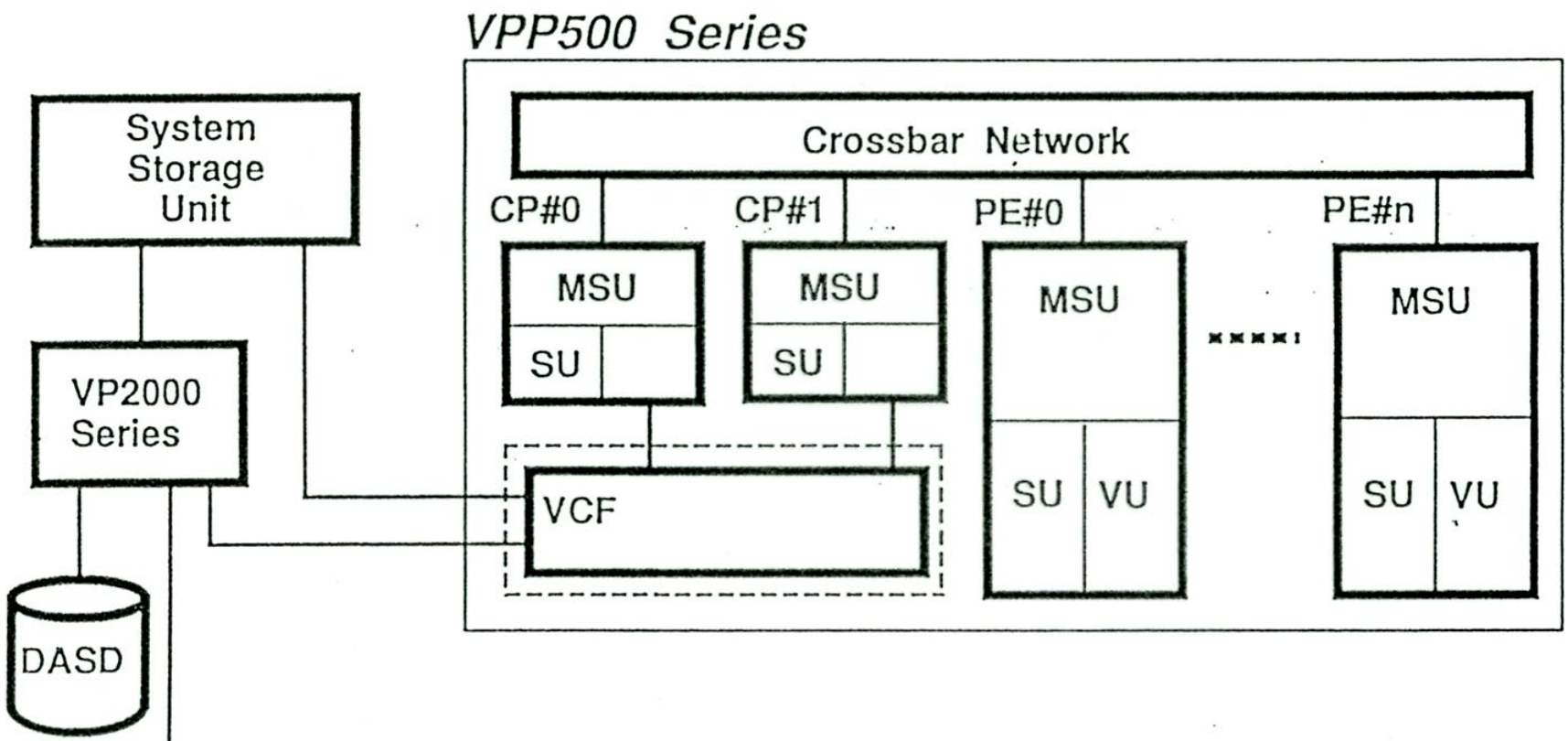



Courtesy

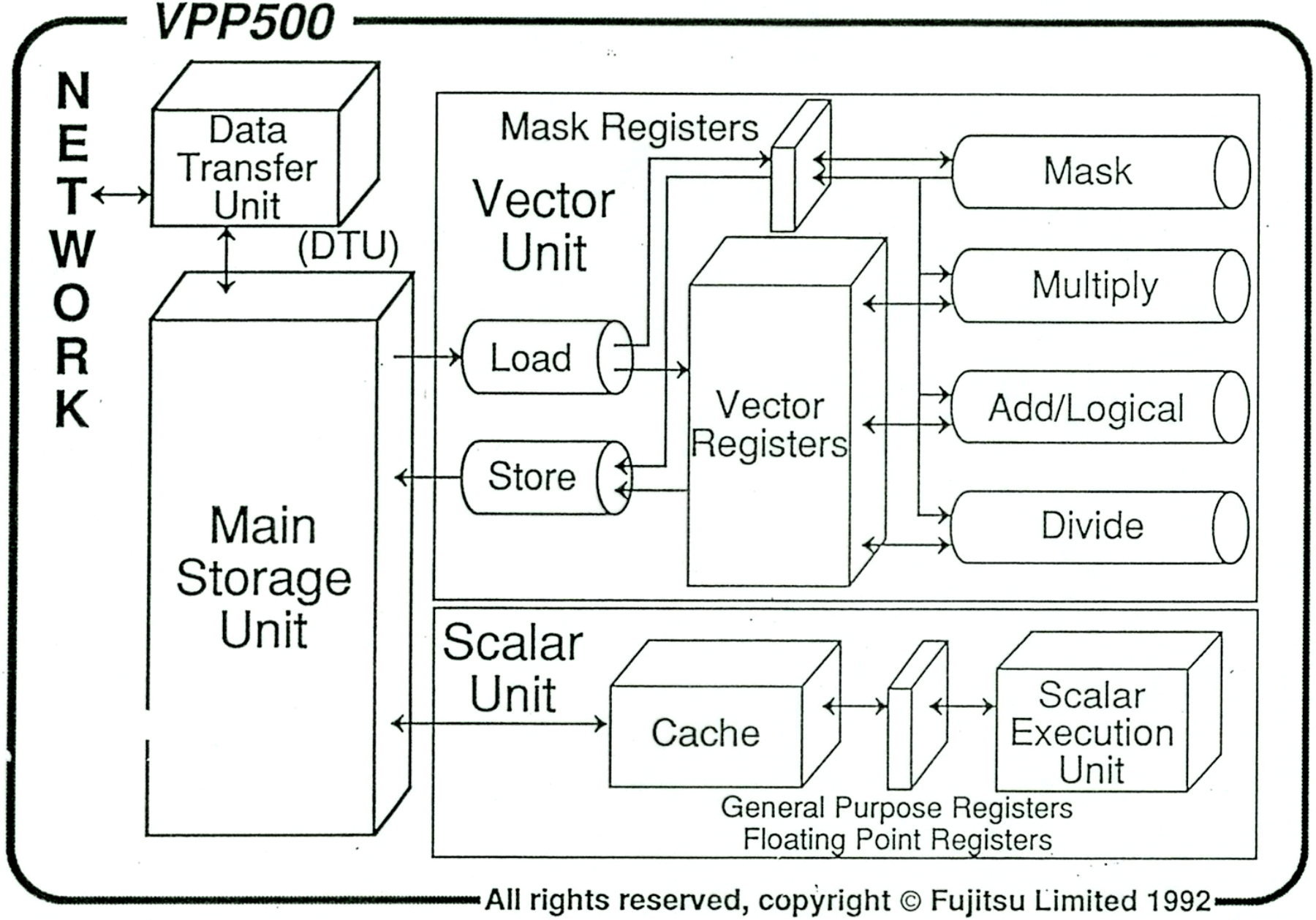
Dr. T.Kitamura

VPP500

# System configuration




CP : Control Processor
PE : Processing Element
VCF : VP/VPX Connection Feature
SU : Scalar Unit
VU : Vector Unit
MSU : Main Storage 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)

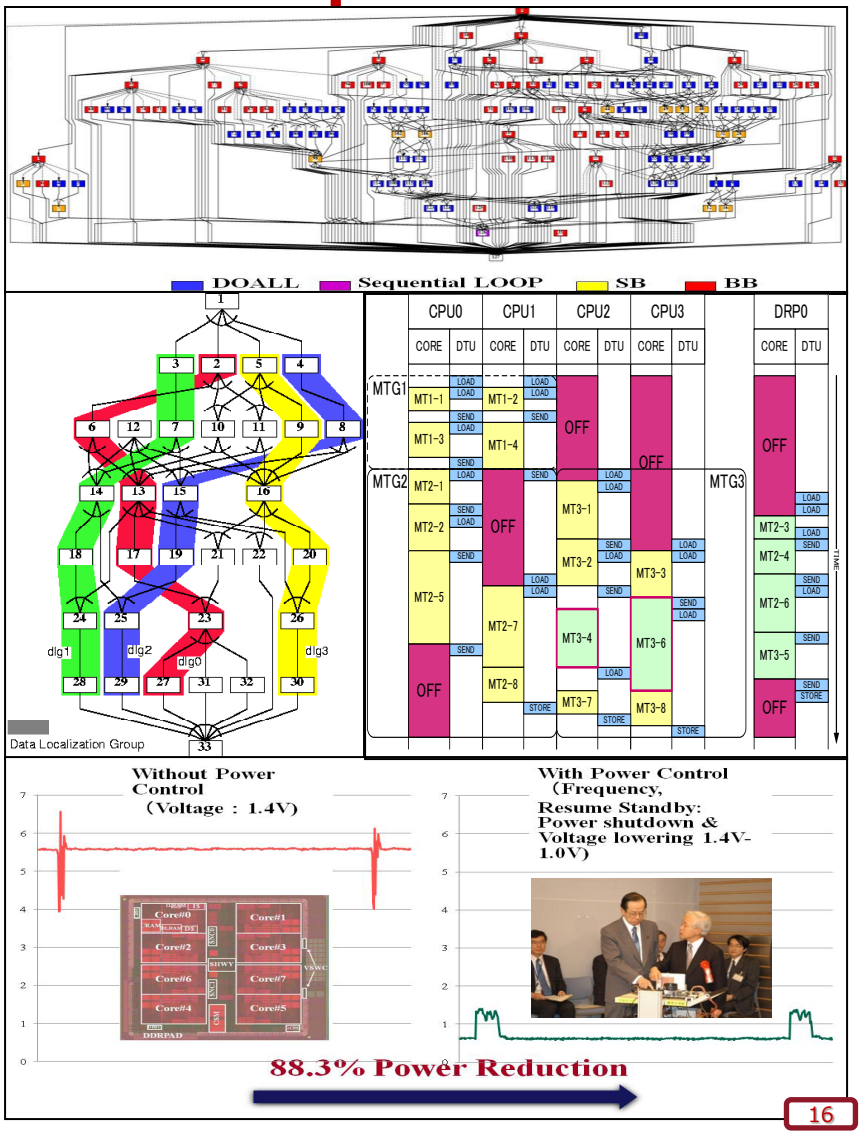
## Data Transfer Overlapping(2016 partially)

Data transfer overlapping using Data Transfer Controllers (DMAs)

## Power Reduction

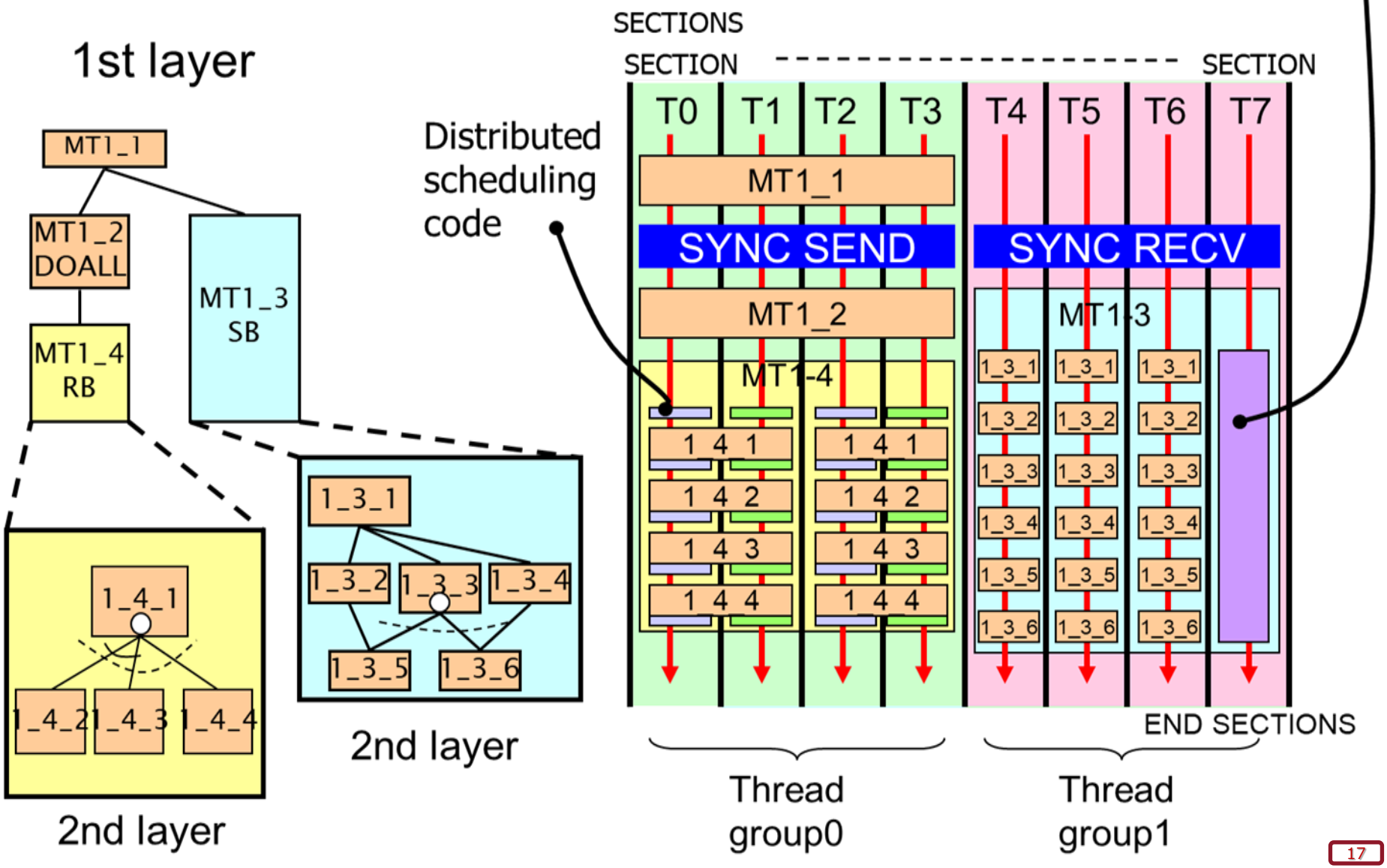
(2005 for Multicore, 2011 Multi-processes, 2013 on ARM)

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



# Generated Multigrain Parallelized Code

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



# 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

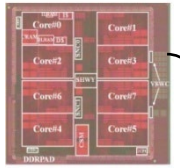
Accelerator 2
Code

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

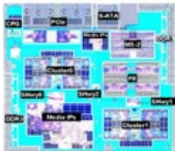
**Low Power Heterogeneous Multicore Code Generation**
API Analyzer (Available from Waseda) | Existing sequential compiler

**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

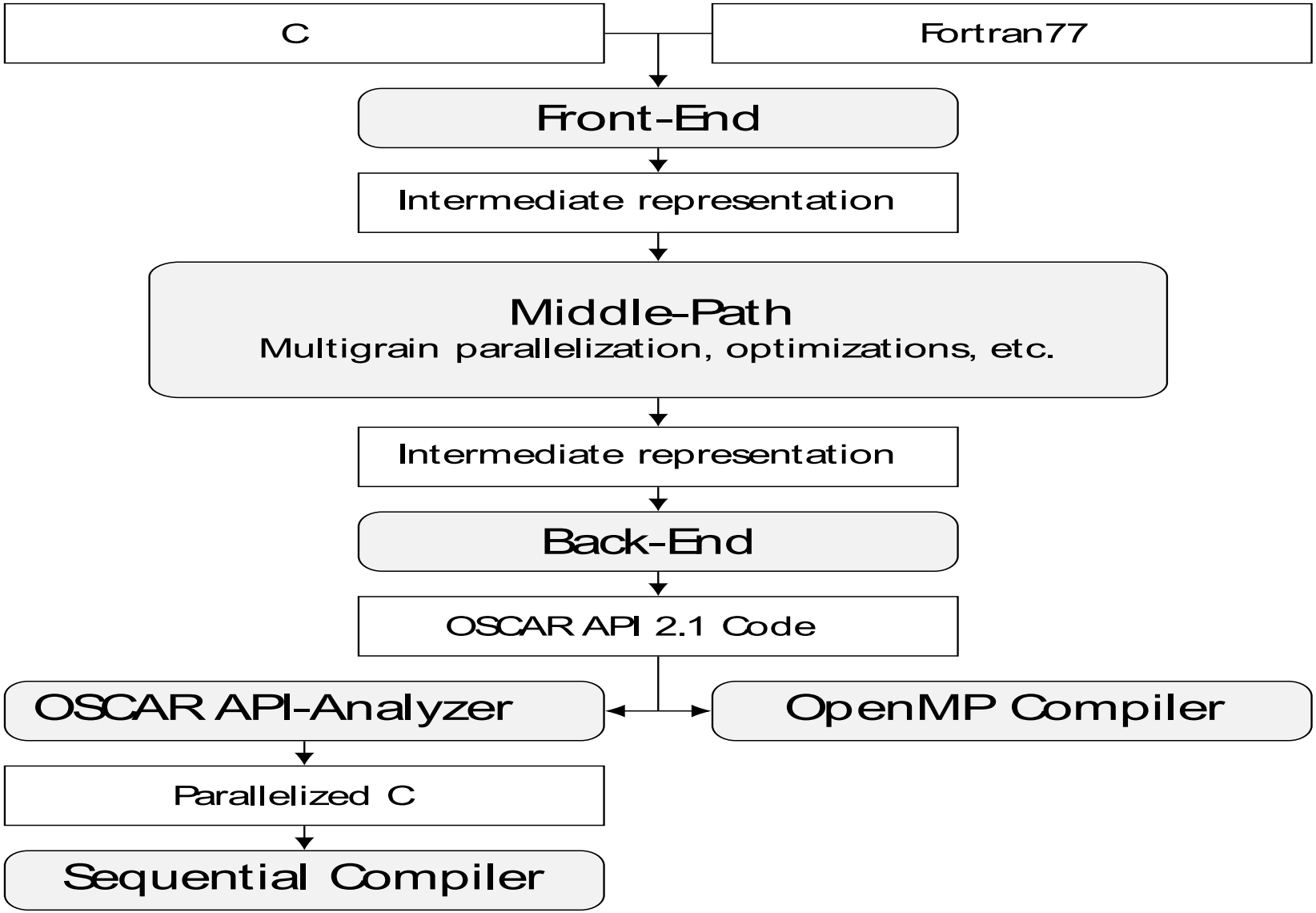
Shred memory servers

Executable on various multicores

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

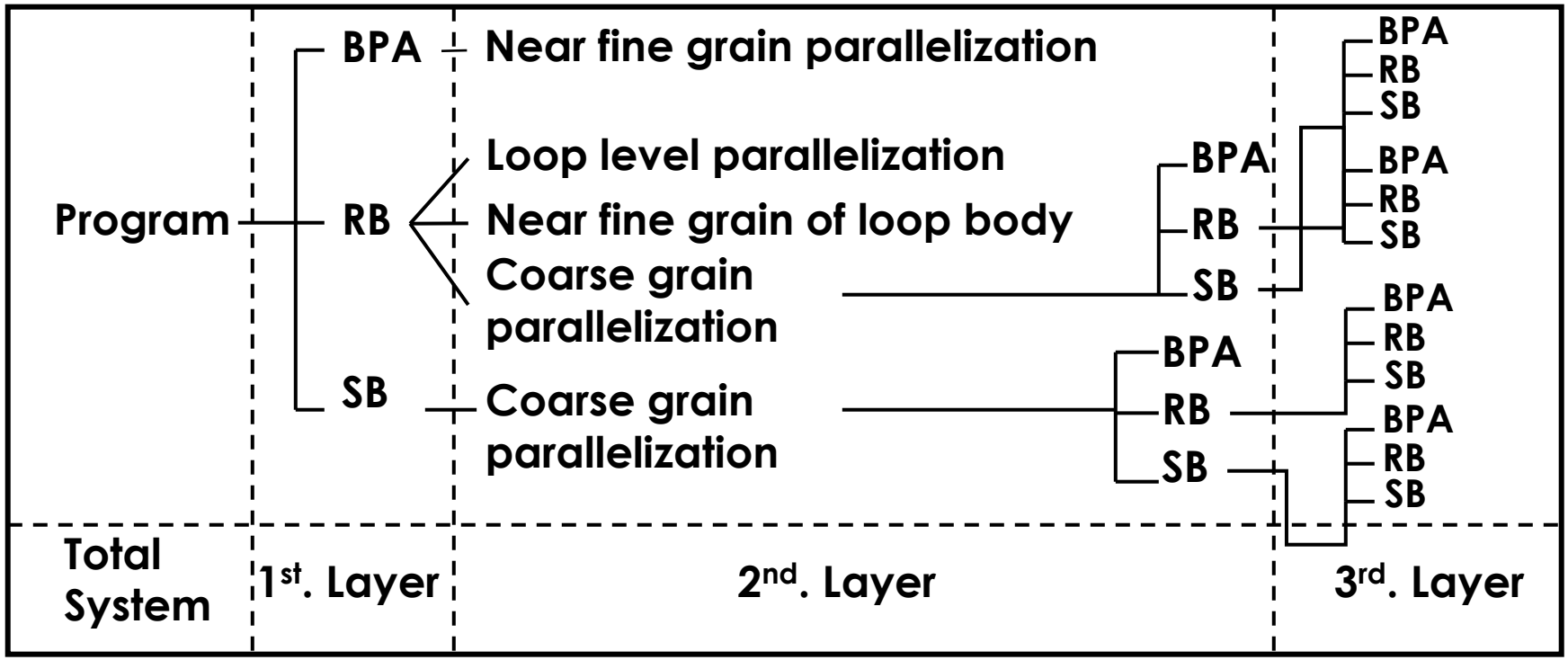
OSCAR: Optimally Scheduled Advanced Multiprocessor
API : Application Program Interface

# The OSCAR Compiler – Compile flow

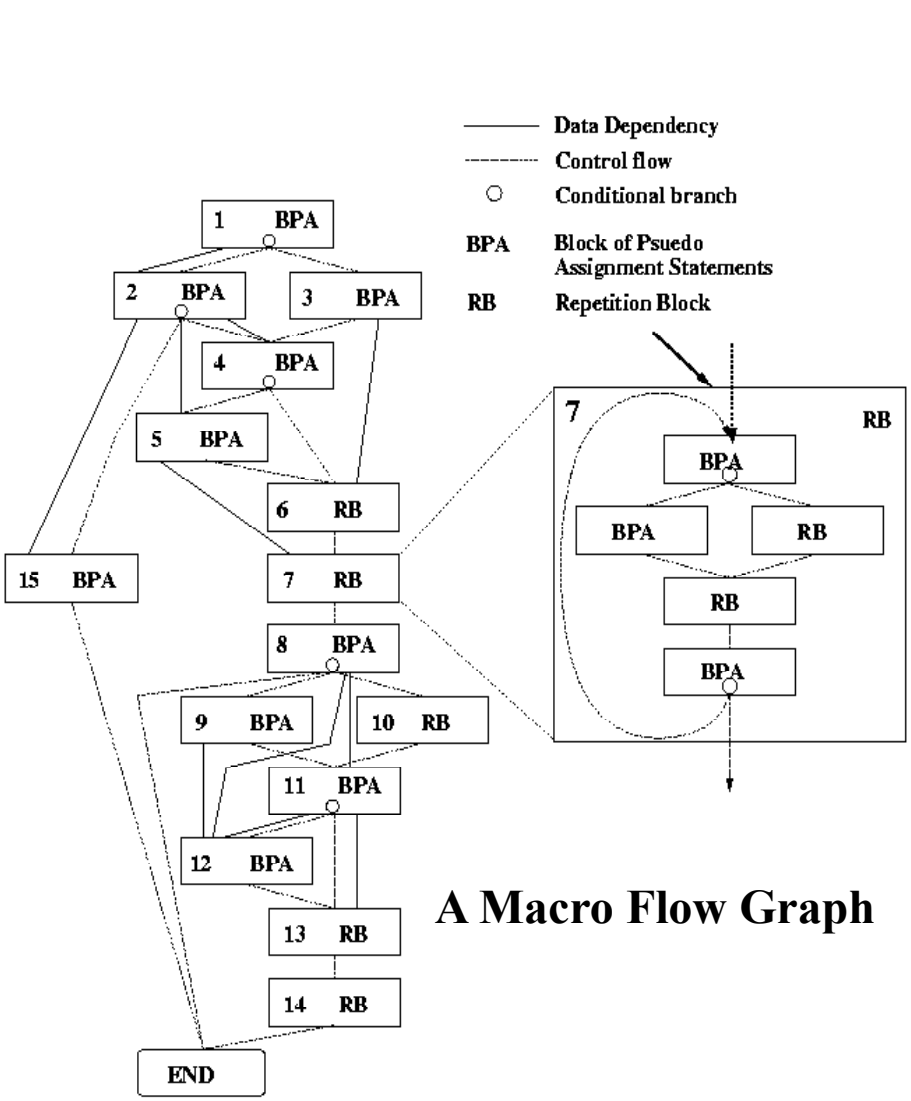


# Generation of Coarse Grain Tasks

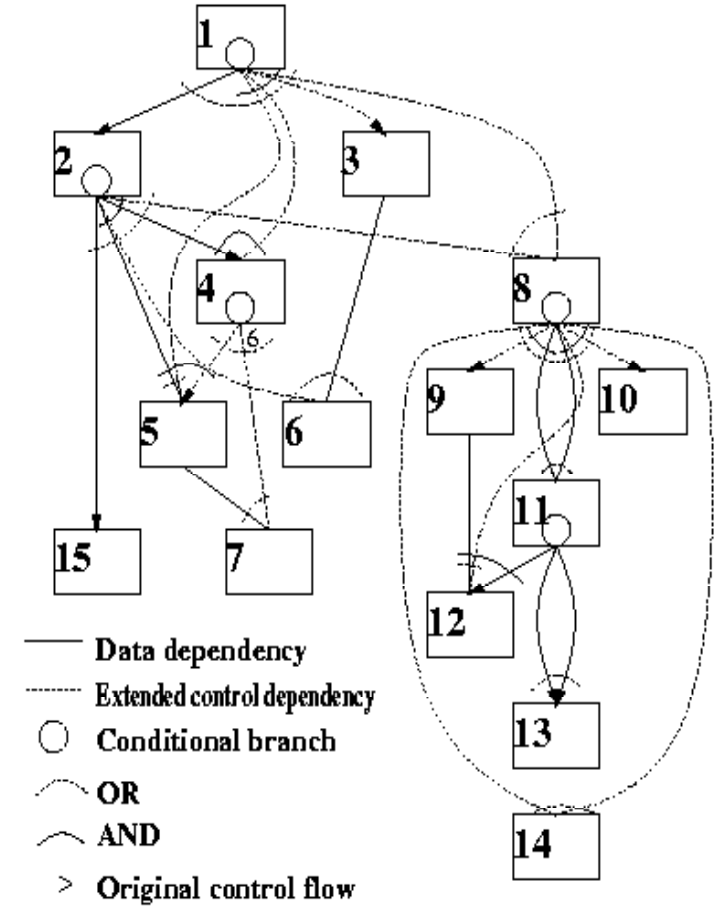
- **Macro-tasks (MTs)**
  - Block of Pseudo Assignments (BPA): Basic Block (BB)
  - Repetition Block (RB) : natural loop
  - Subroutine Block (SB): subroutine



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




A Macro Flow Graph

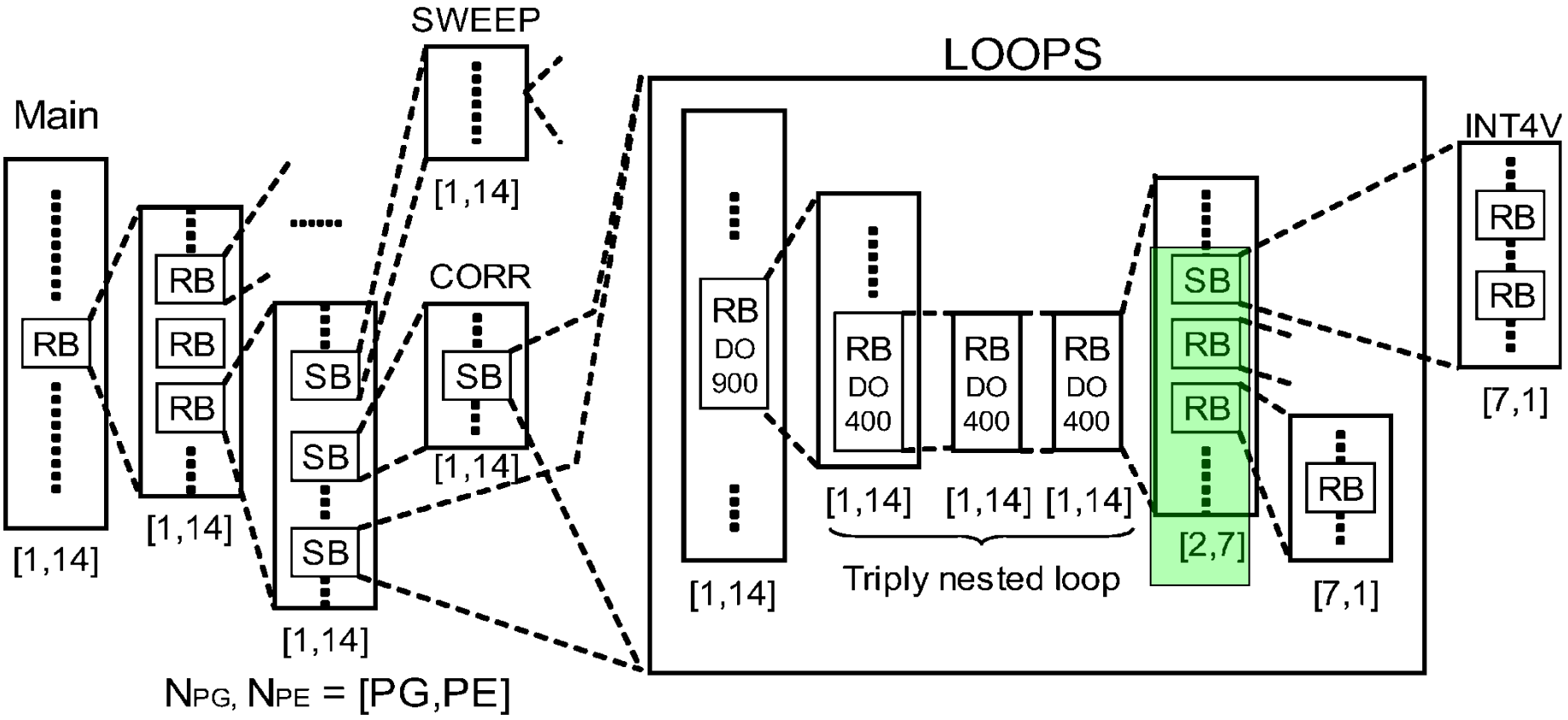



A Macro Task Graph

# Automatic processor assignment in 103.su2cor

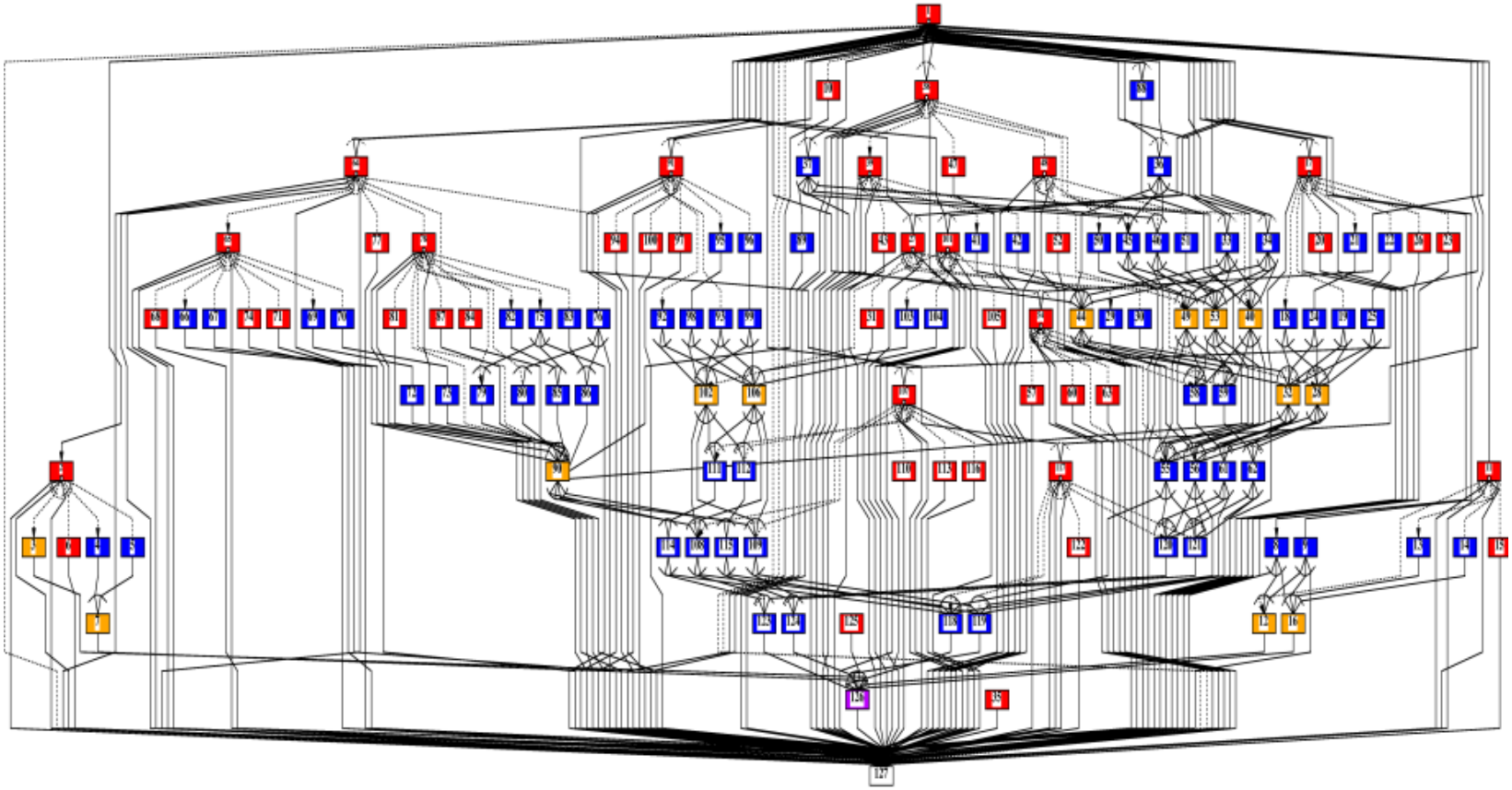
- **Using 14 processors**

**Coarse grain parallelization within DO400**



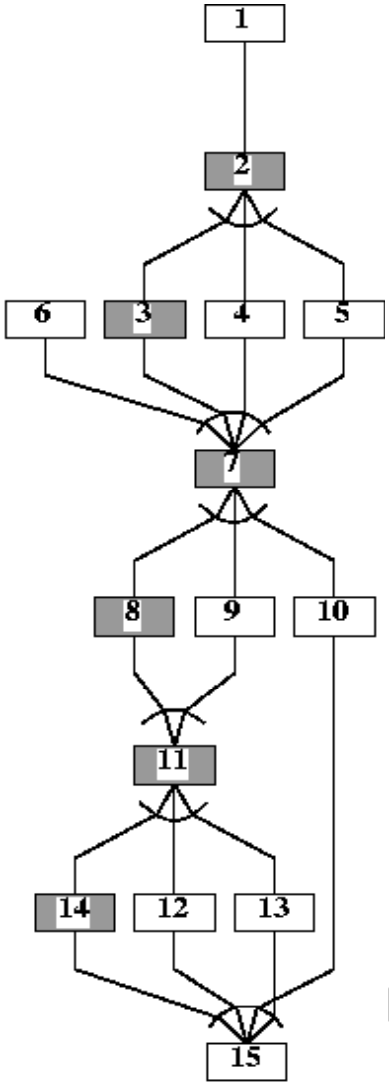
# MTG of Su2cor-LOOPS-DO400

- **Coarse grain parallelism PARA_ALD = 4.3**

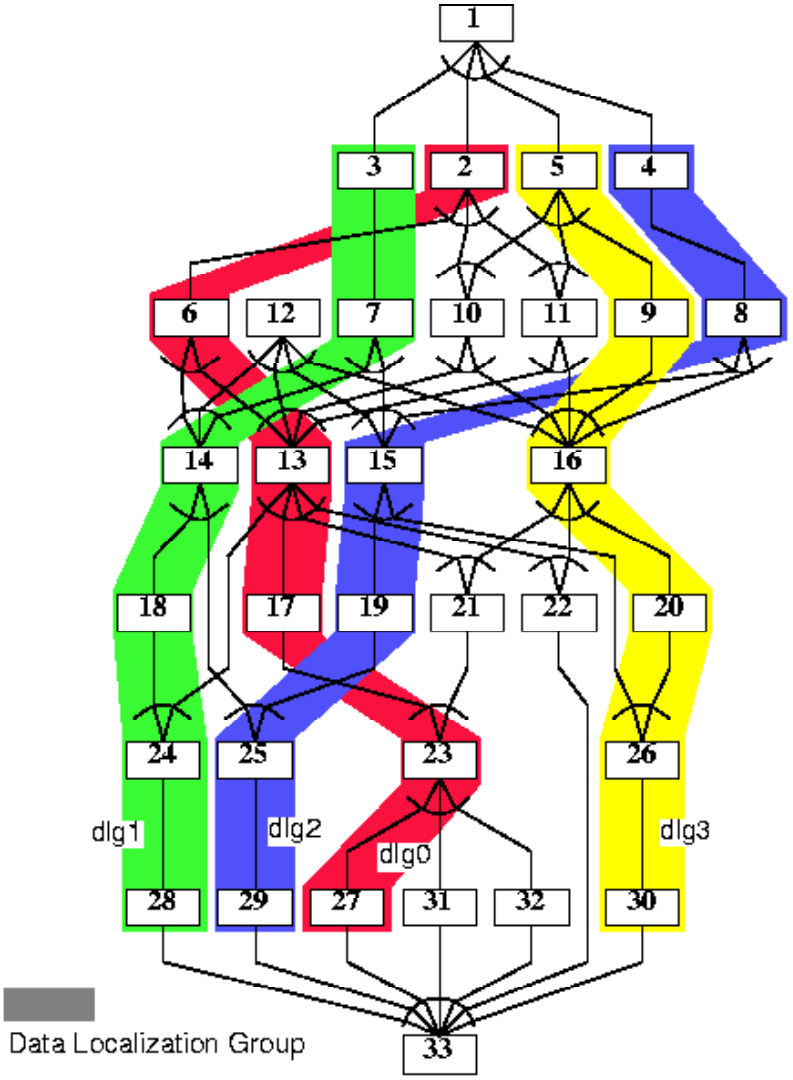


**DOALL** **Sequential LOOP** **SB** **BB**

# Data Localization




MTG

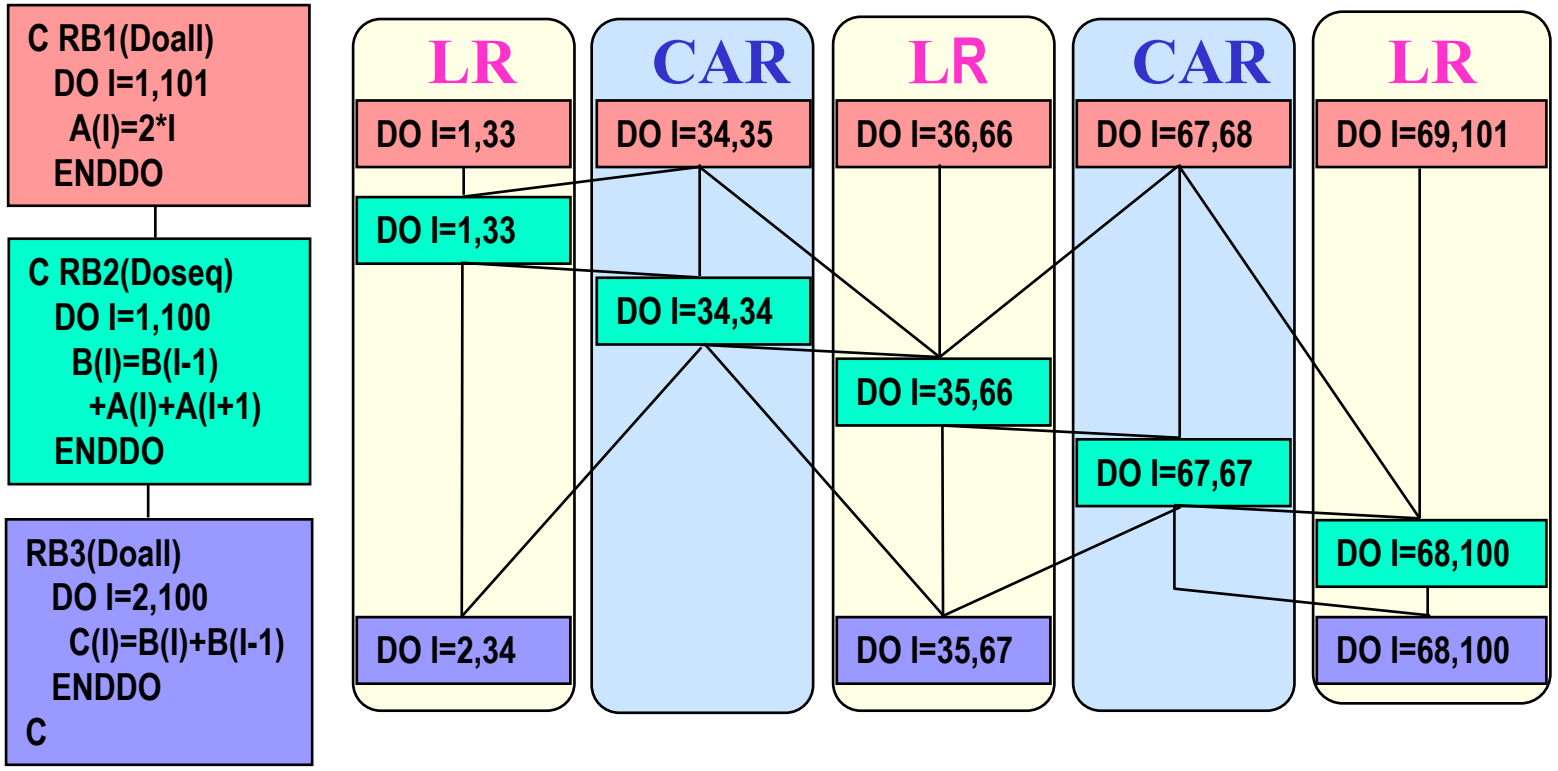



MTG after Division


A schedule for two processors

# 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**

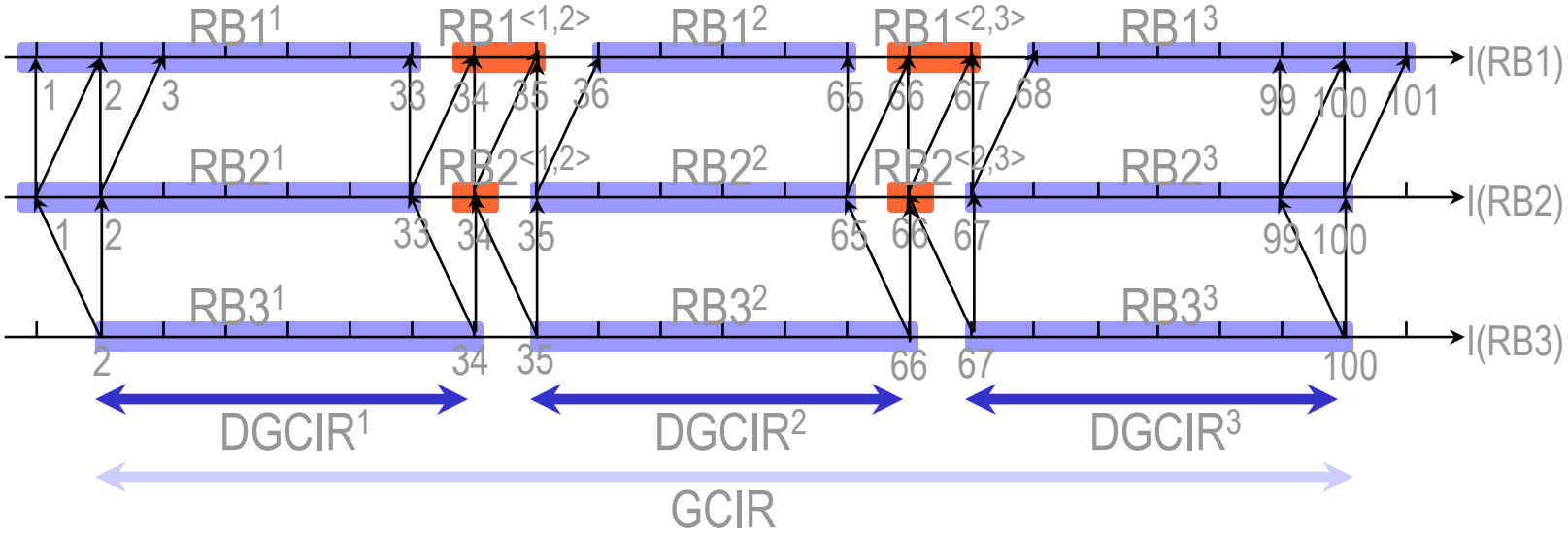





25

# Decomposition of RBs in TLG

- Decompose GCIR into $DGCIR^p (1 \leqq p \leqq n)$
  - n: (multiple) num of PCs, DGCIR: Decomposed GCIR
- Generate CAR on which $DGCIR^p$&$DGCIR^{p+1}$ are data-dep.
- Generate LR on which $DGCIR^p$ is data-dep.



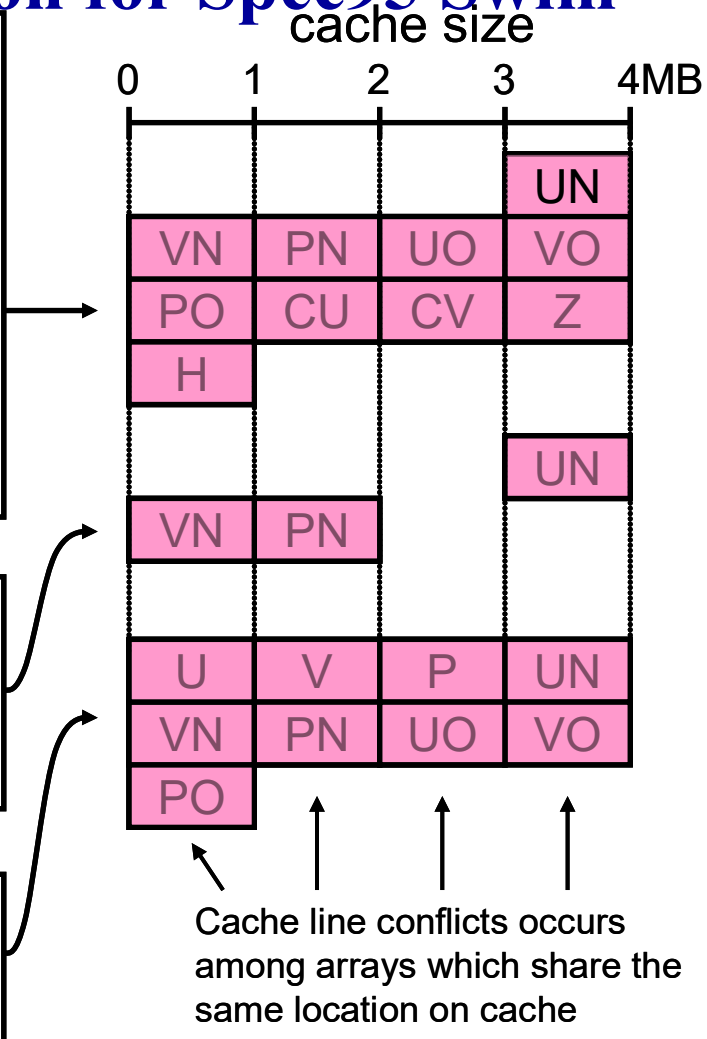
# An Example of Data Localization for Spec95 Swim

```
      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

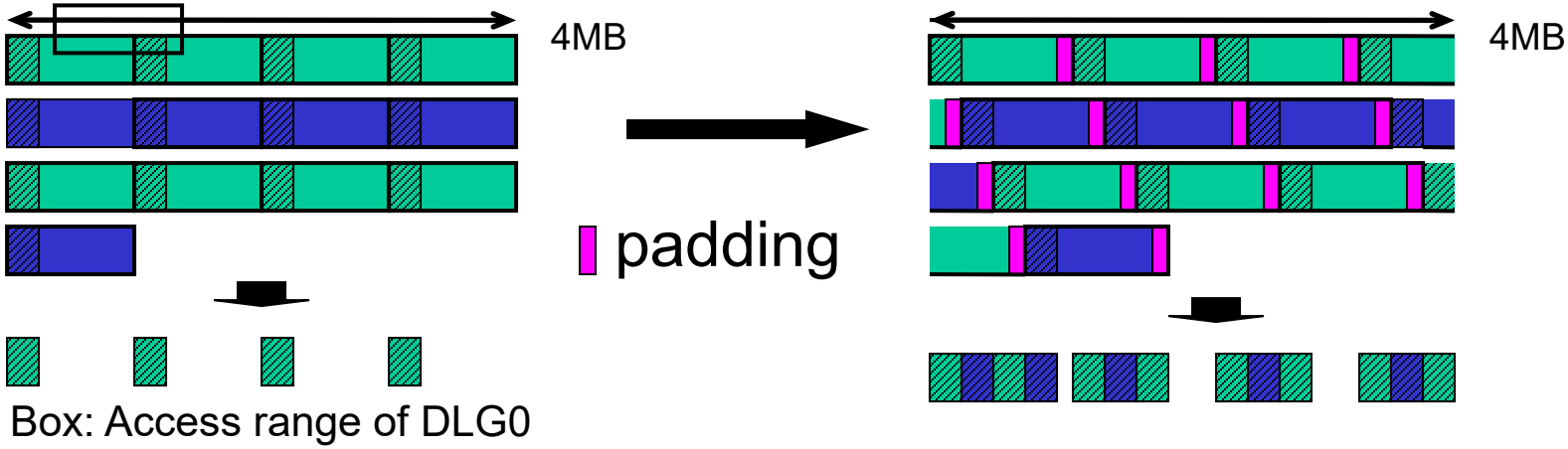
Declaration part of arrays in spec95 swim

**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)
```

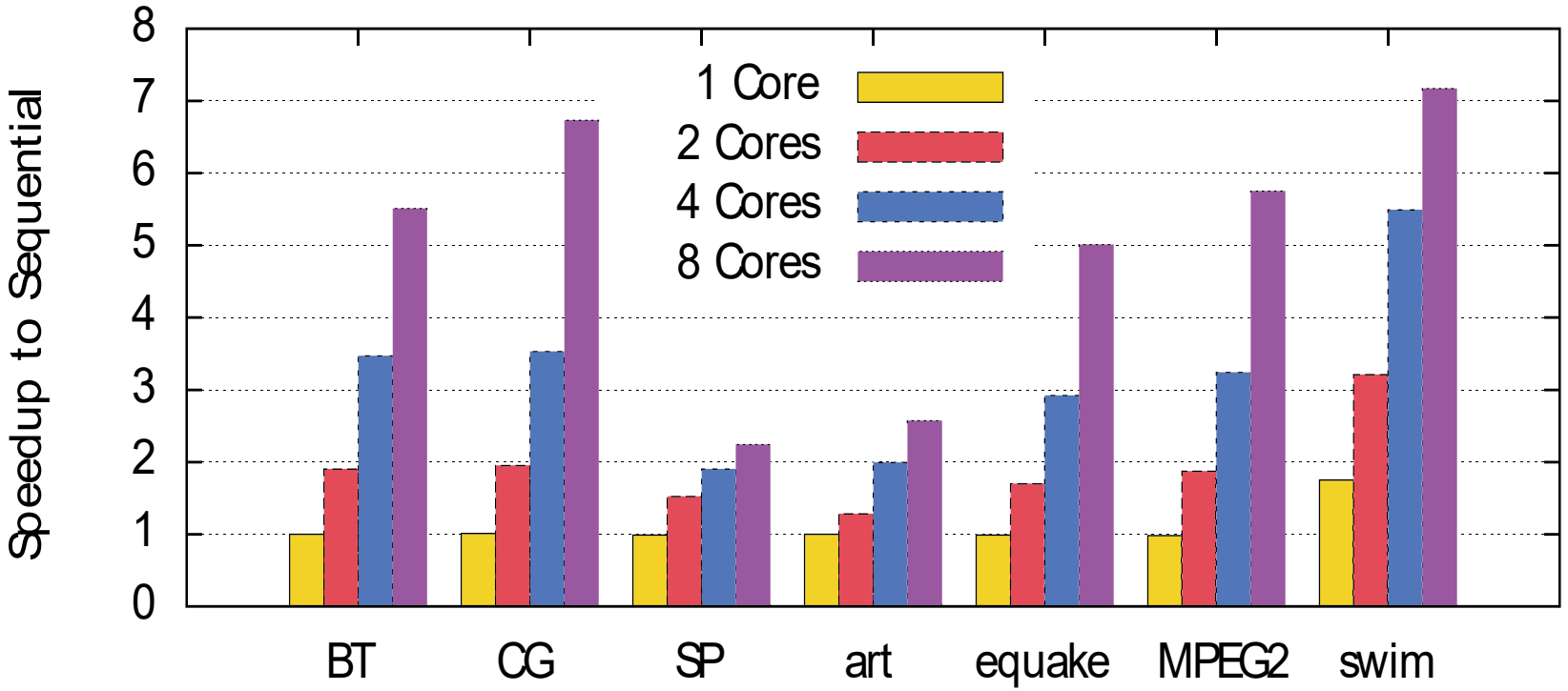



Box: Access range of DLG0

# Intel Xeon E5-2650v4 – Benchmark results on upto 8 cores

- x86-64 based Architecture, 12 Cores, 2.2 GHz – 2.9 GHz
- 30 MiB shared L3 cache,
- L3 Cache: Shared by all cores
- speedup to sequential version
- gcc as backend

- **NAS parallel benchmark suite**
  - **BT: Block Tri-diagonal solver**
  - **CG: Conjugate Gradient computation**
  - **SP: Scalar Penta-diagonal solver**
- **SPEC2000**
  - **art: Image recognition / Neural networks**
  - **equake: Seismic wave propagation simulation**
  - **swim: Shallow water modelling - Fortran 77**
- **MediaBench II : MPEG2 encoding**



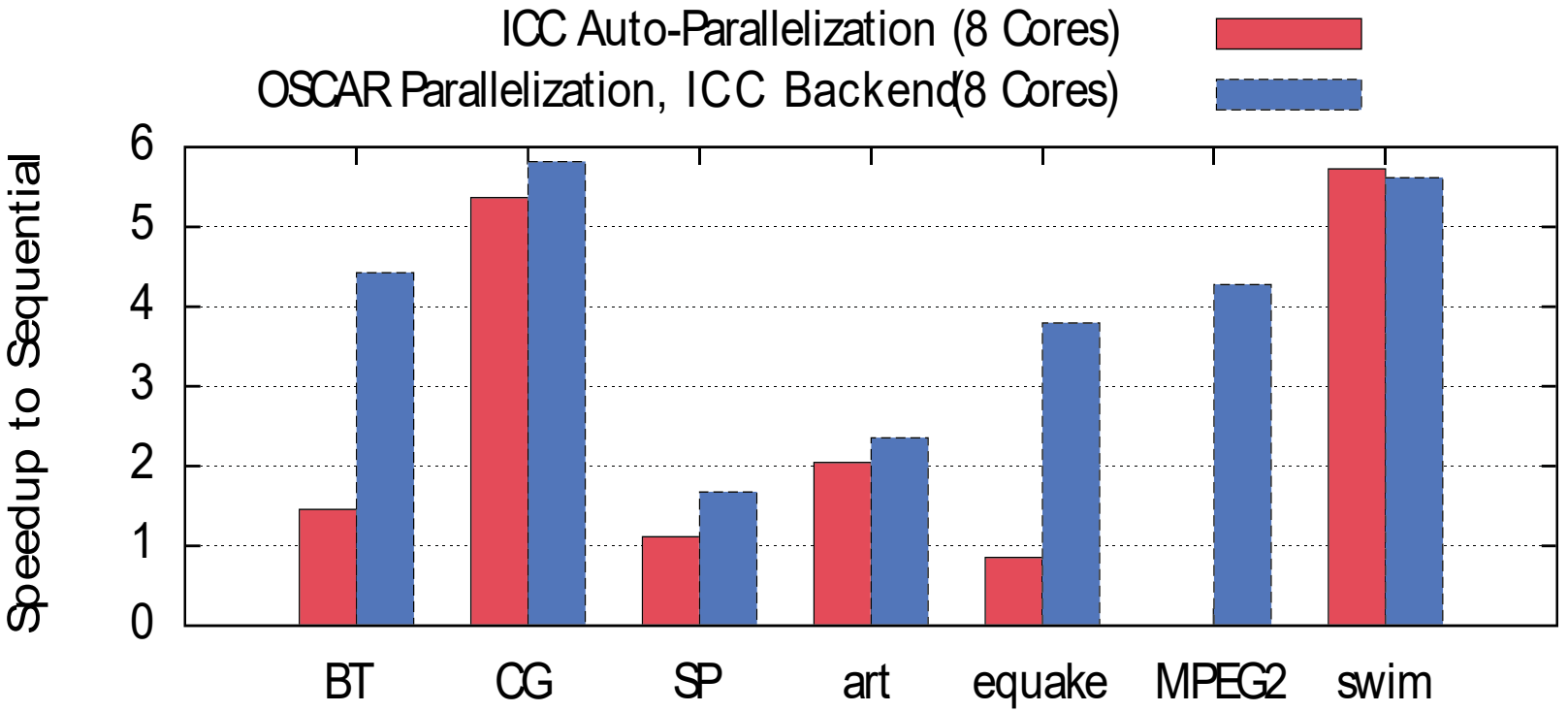

- swim shows superlinear speedup and 1 core speedup
  - seq.: 58.1 s, 1 core OSCAR: 33.2 s, 4 core OSCAR: 10.5 s

# Intel Xeon E5-2650v4 – ICC backend compiler on 8cores

- x86-64 based Architecture, 12 Cores, 2.2 GHz – 2.9 GHz
- 30 MiB shared L3 cache,
- L3 Cache: Shared by all cores

- NAS parallel benchmark suite
  - BT: Block Tri-diagonal solver
  - CG: Conjugate Gradient computation
  - SP: Scalar Penta-diagonal solver
- SPEC2000
  - art: Image recognition / Neural networks
  - equake: Seismic wave propagation simulation
  - swim: Shallow water modelling - Fortran 77
- MediaBench II : MPEG2 encoding

- speedup to sequential version



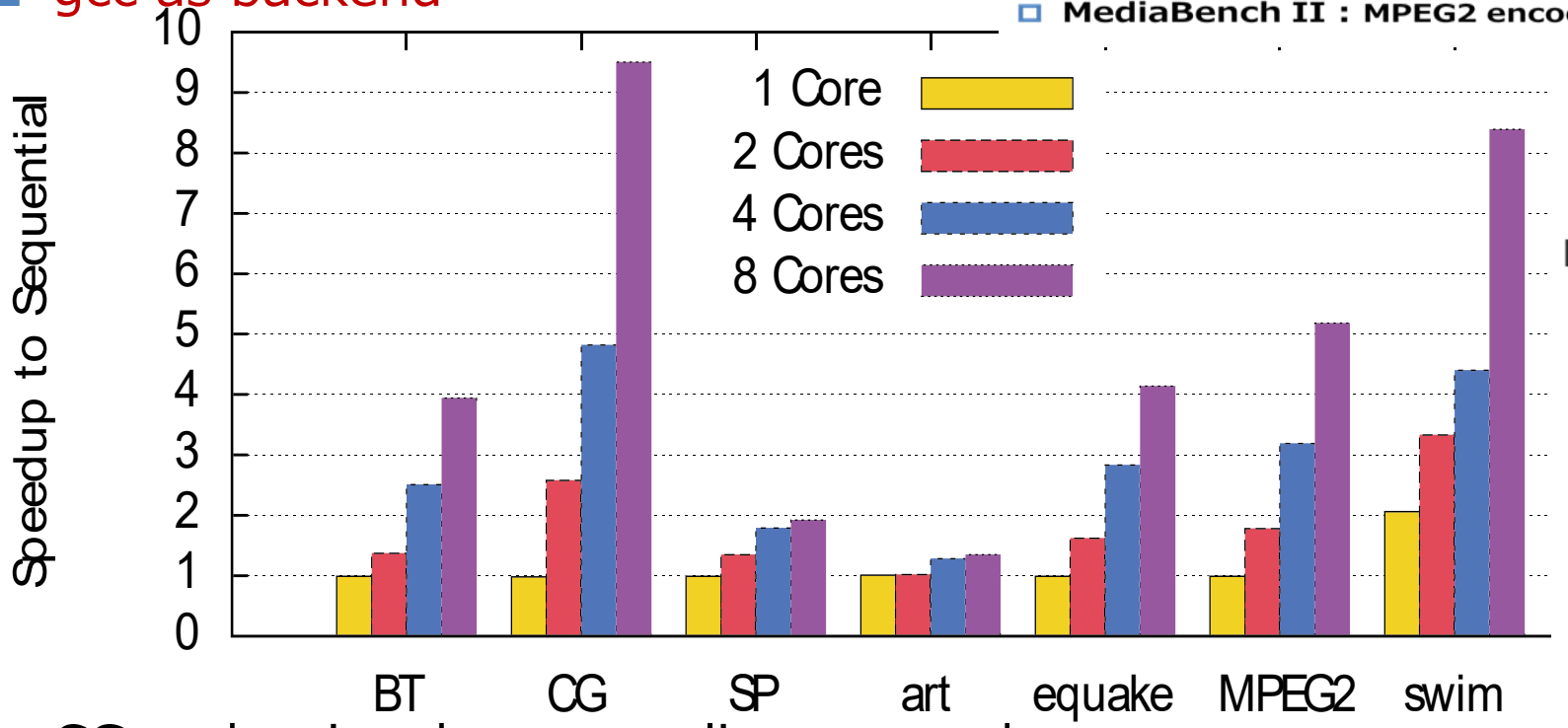

- OSCAR can be used with different backend compilers
  - slightly lower relative speedups are due to better sequential performance

# AMD EPYC 7702P – Benchmark results on upto 8 cores

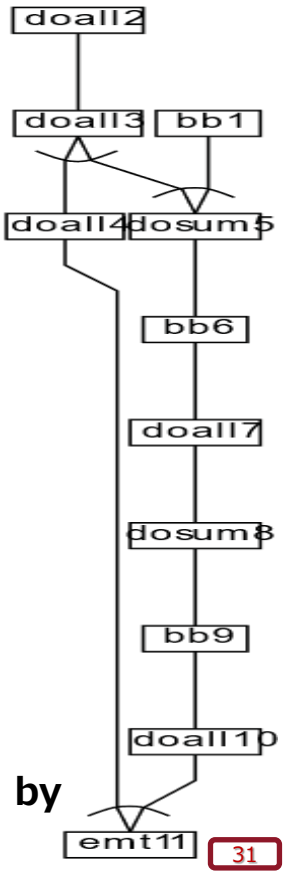
- x86-64 based Architecture, 64 Cores, 2.0 GHz – 3.35 GHz
- 16 MiB L3 cache per 4 core cluster,
- shared within the cluster

- NAS parallel benchmark suite
  - BT: Block Tri-diagonal solver
  - CG: Conjugate Gradient computation
  - SP: Scalar Penta-diagonal solver
- SPEC2000
  - art: Image recognition / Neural networks
  - equake: Seismic wave propagation simulation
  - swim: Shallow water modelling - Fortran 77
- MediaBench II : MPEG2 encoding

- speedup to sequential version
- gcc as backend




- CG and swim show superlinear speedup
  - CG: seq.: 0.86 s, 8 core OSCAR: 0.09 s

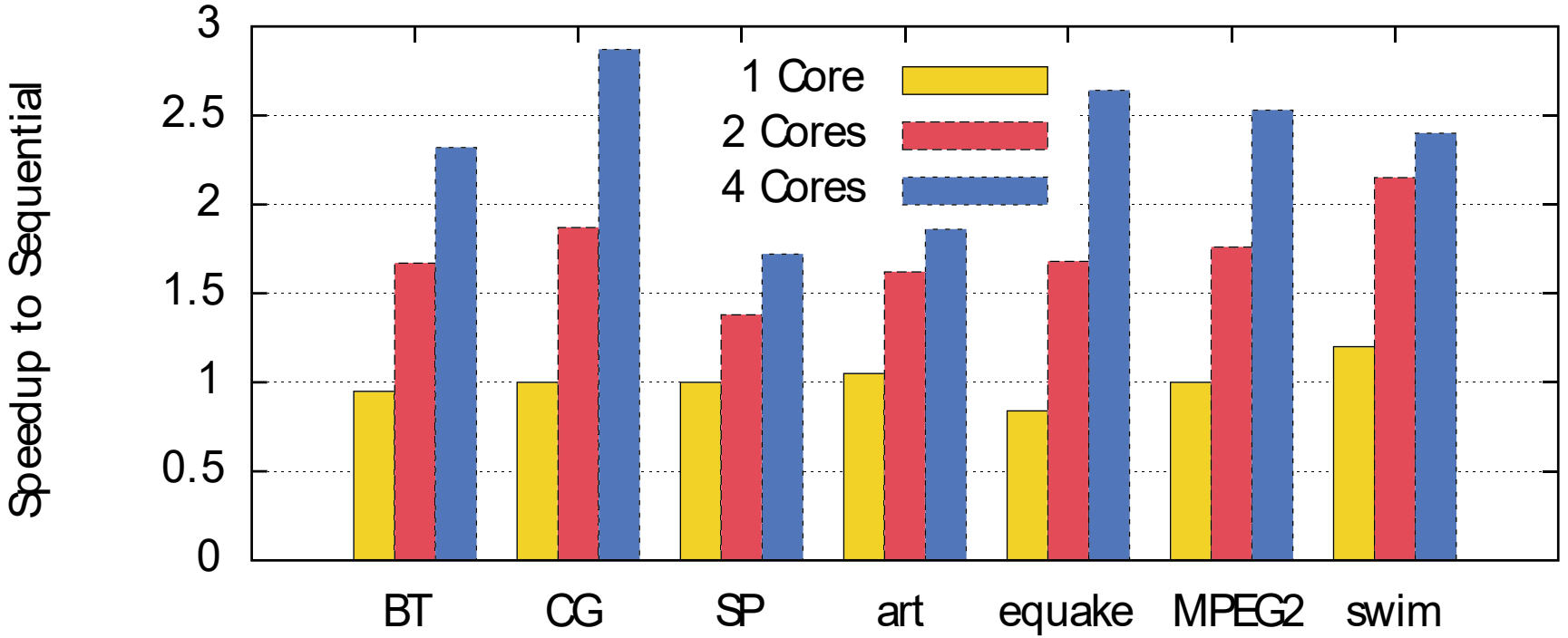



**CG macrotask graph**
**L3 cache Optimization by Data Localization**

# NVIDIA Carmel ARMv8.2 – Benchmark results on upto 4 cores

- Arm v8.2 based Architecture, 6 Cores, 1.4 GHz
- 4 MiB shared L3 cache,
- L3 Cache: Shared across all cores
- speedup to sequential versior
- gcc as backend

- NAS parallel benchmark suite
  - BT: Block Tri-diagonal solver
  - CG: Conjugate Gradient computation
  - SP: Scalar Penta-diagonal solver
- SPEC2000
  - art: Image recognition / Neural networks
  - equake: Seismic wave propagation simulation
  - swim: Shallow water modelling - Fortran 77
- MediaBench II : MPEG2 encoding



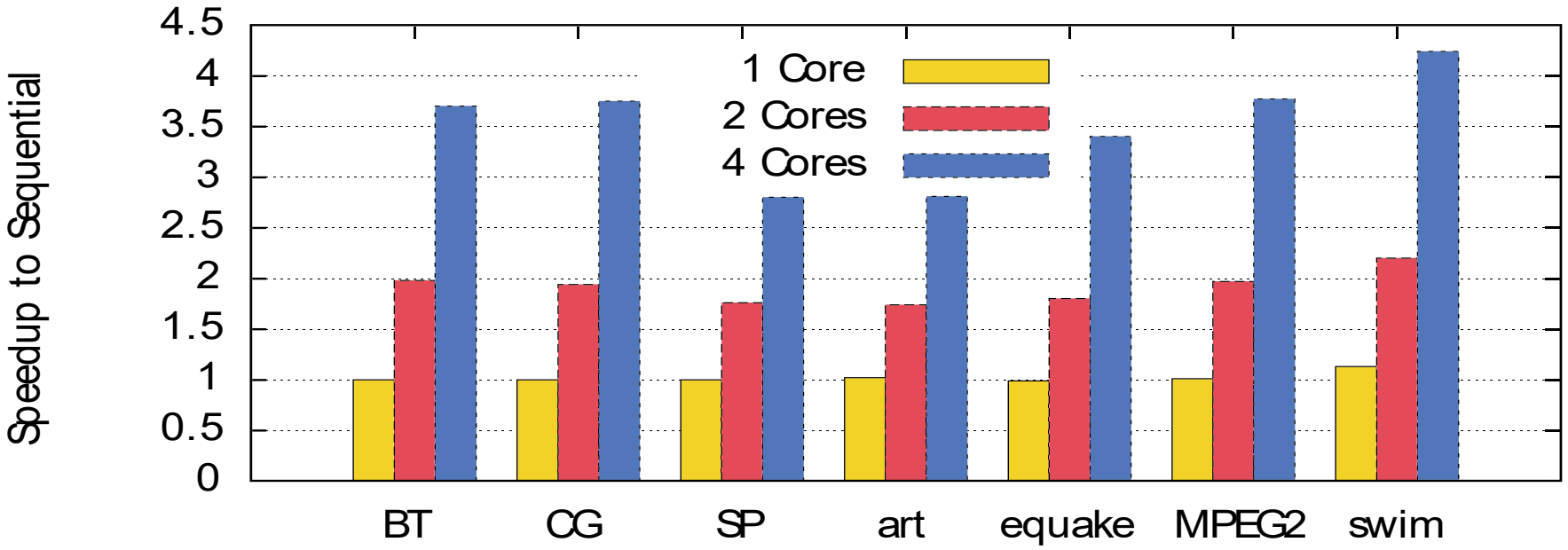

- overall good speedup is observed
  - equake: seq.: 19.0 s, 4 core OSCAR: 7.18 s

# SiFive Freedom U740 – Benchmark results on upto 4 cores

- RISC-V based Architecture, 4 Cores, 1.2 GHz
- 2 MiB shared L2 cache
- L2 Cache: Shared across all cores

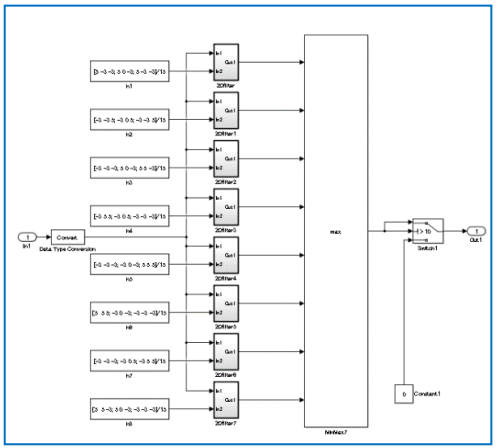
- NAS parallel benchmark suite
  - BT: Block Tri-diagonal solver
  - CG: Conjugate Gradient computation
  - SP: Scalar Penta-diagonal solver
- SPEC2000
  - art: Image recognition / Neural networks
  - equake: Seismic wave propagation simulation
  - swim: Shallow water modelling - Fortran 77
- MediaBench II : MPEG2 encoding

- speedup to sequential version
- gcc as backend




- overall good speedup is observed, swim superlinear
  - BT: seq.: 2041 s, 4 core OSCAR: 551 s

# OSCAR Compile Flow for MATLAB/Simulink



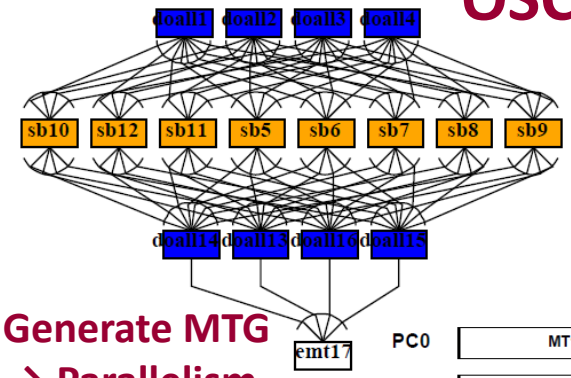
Simulink model

Generate C code
using Embedded Coder

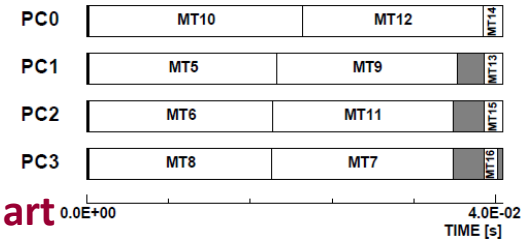
```
/* Model step function */
void VesselExtraction_step(void)
{
  int32_T i;
  real_T u0;

  /* 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



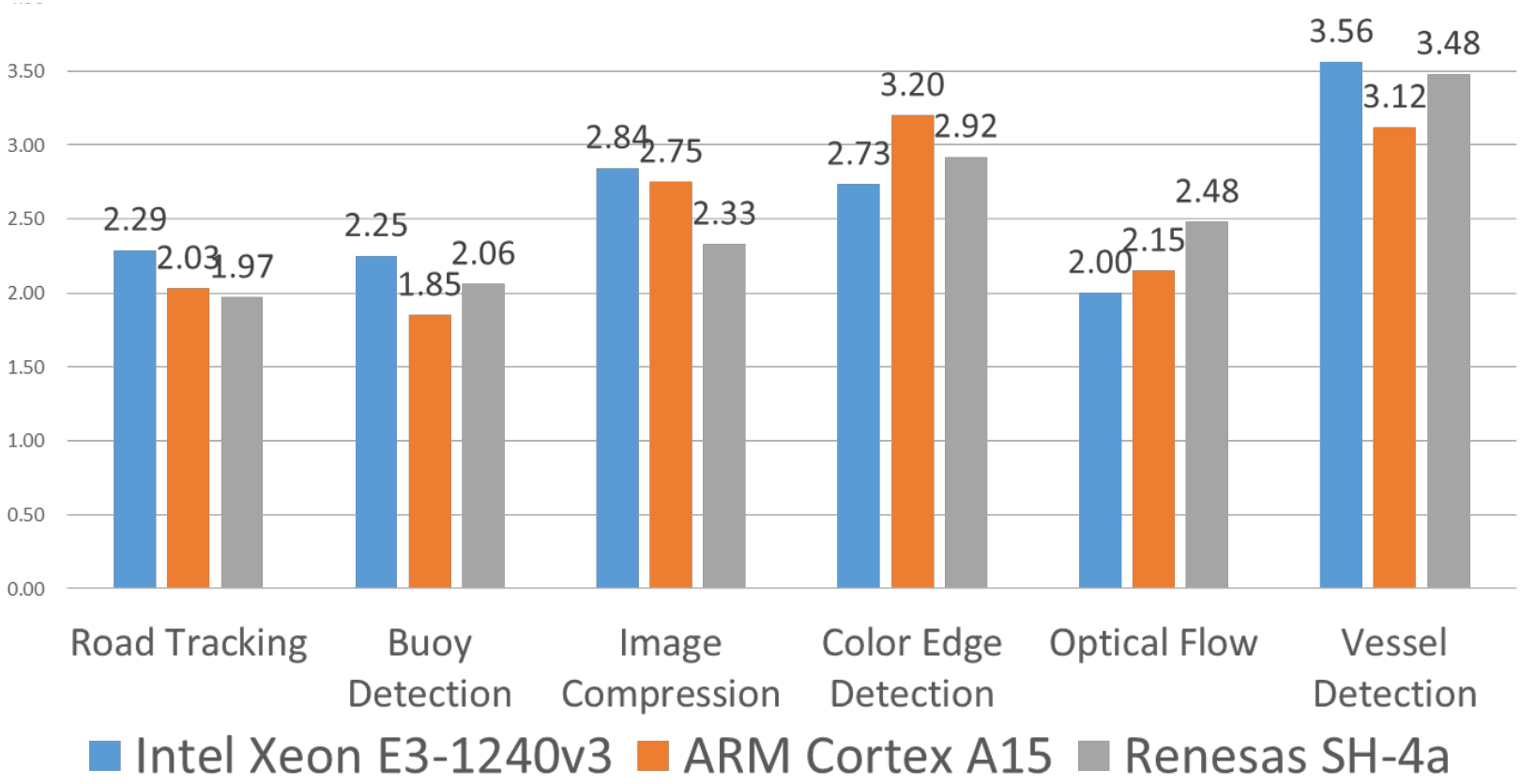

(2) Generate gantt chart
→ Scheduling in a multicore

```
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)

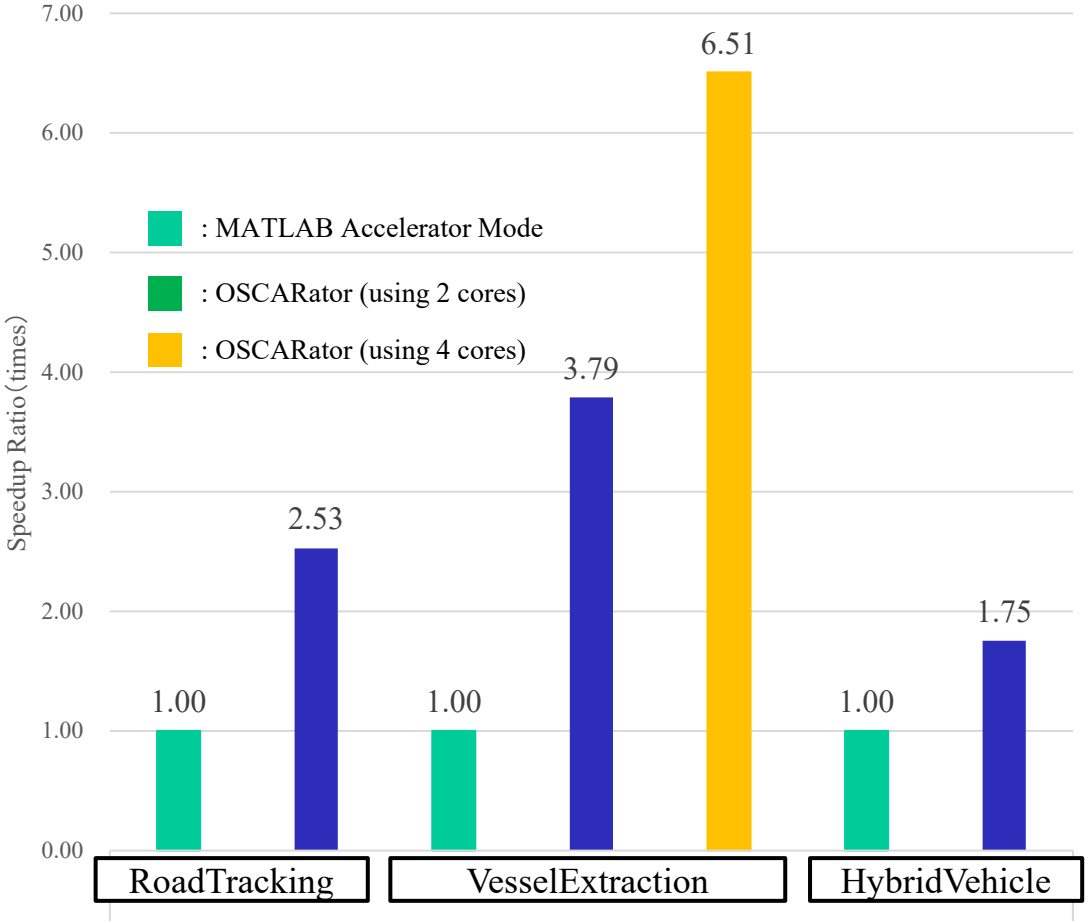



Road Tracking, Image Compression : http://www.mathworks.co.jp/jp/help/vision/examples
Buoy Detection : http://www.mathworks.co.jp/matlabcentral/fileexchange/44706-buoy-detection-using-simulink
Color Edge Detection : http://www.mathworks.co.jp/matlabcentral/fileexchange/28114-fast-edges-of-a-color-image--actual-color--not-converting-to-grayscale-/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

# Speedup of Simulink Models by OSCARator on 4 cores Intel Core i5 Processor

**https://www.oscartech.jp/en/**

## 6.51 times speed up on 4 cores against 1 core MATLAB Accerelator Mode for VesselExtraction




Intel Core i5 7400T 2.4GHz (4 cores)
16GB (SODIMM 2400MHz)
Windows 10 Pro (1903)
MATLAB R2019a Update 5
MinGW GCC 6.3

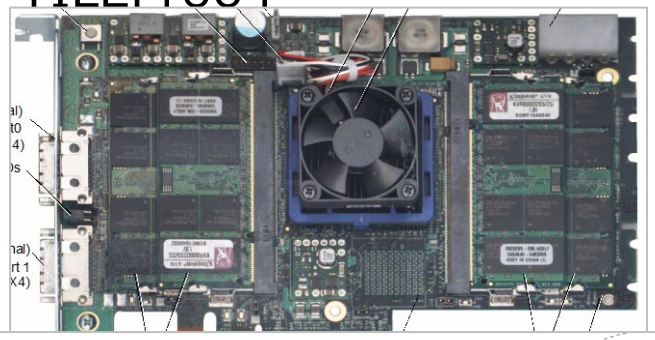
- RoadTracking
  - from Computer Vision Toolbox
  - https://jp.mathworks.com/help/vision/examples/color-based-road-tracking.html
- VesselExtraction
  - from MATLAB Central
  - modified for Simulink Model
  - https://www.mathworks.com/matlabcentral/fileexchange/24990-retinal-blood-vessel-extraction
- HybridVehicle
  - Hybrid Vehicle Powertrain
  - developed by Kusaka Lab. Waseda University
  - http://www.f.waseda.jp/jin.kusaka/

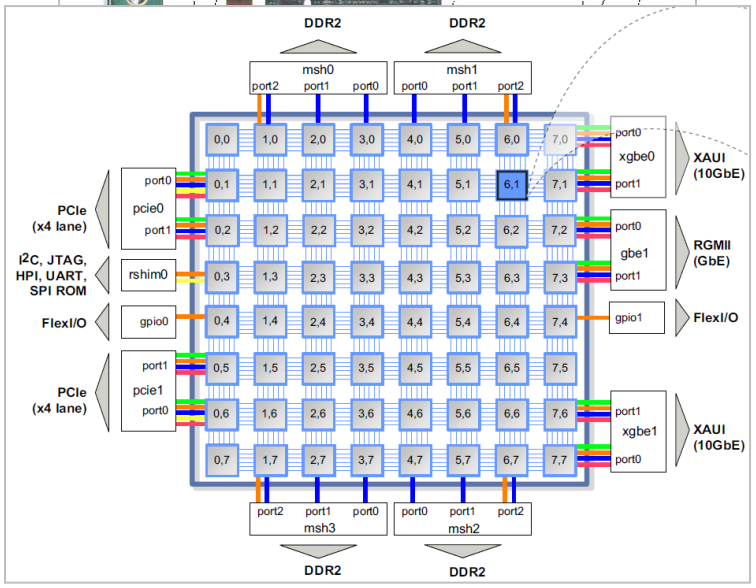
**(Compared with MATLAB Accelerator Mode Simulation)**

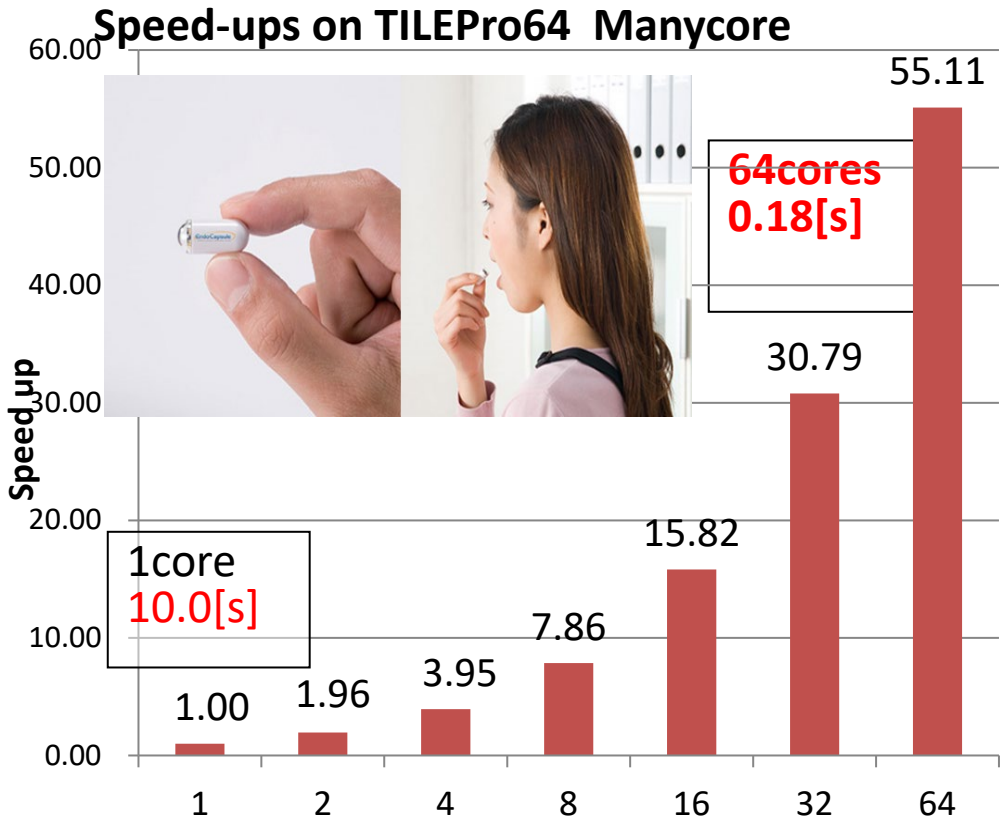
# Automatic Parallelization of JPEG-XR for Drinkable Inner Camera (Endo Capsule)

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

- TILEPro64



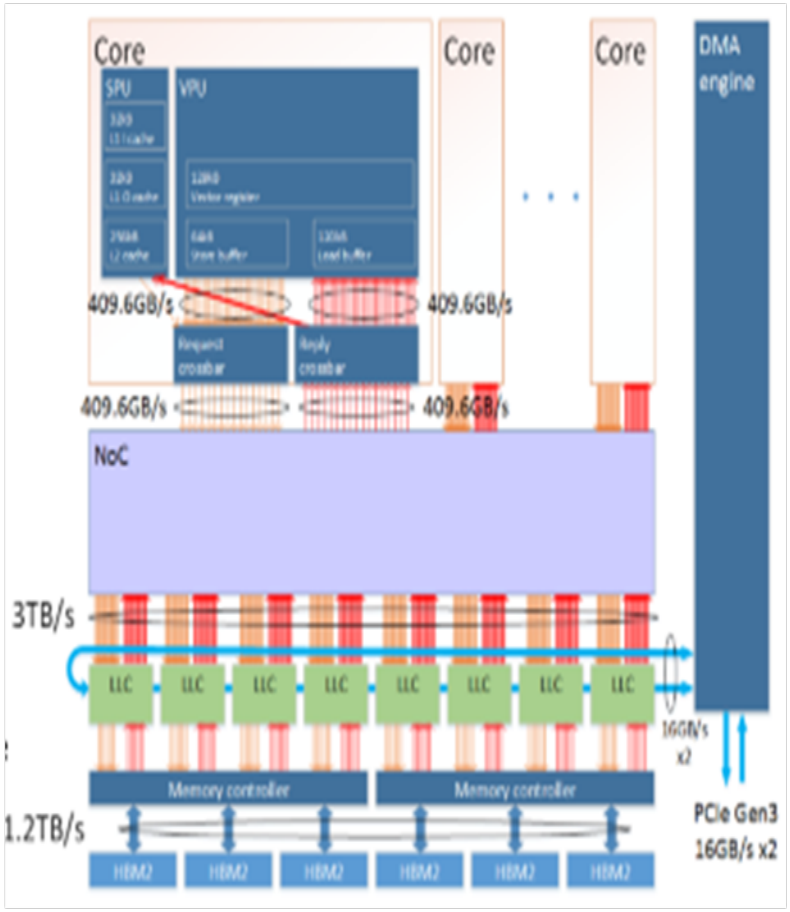


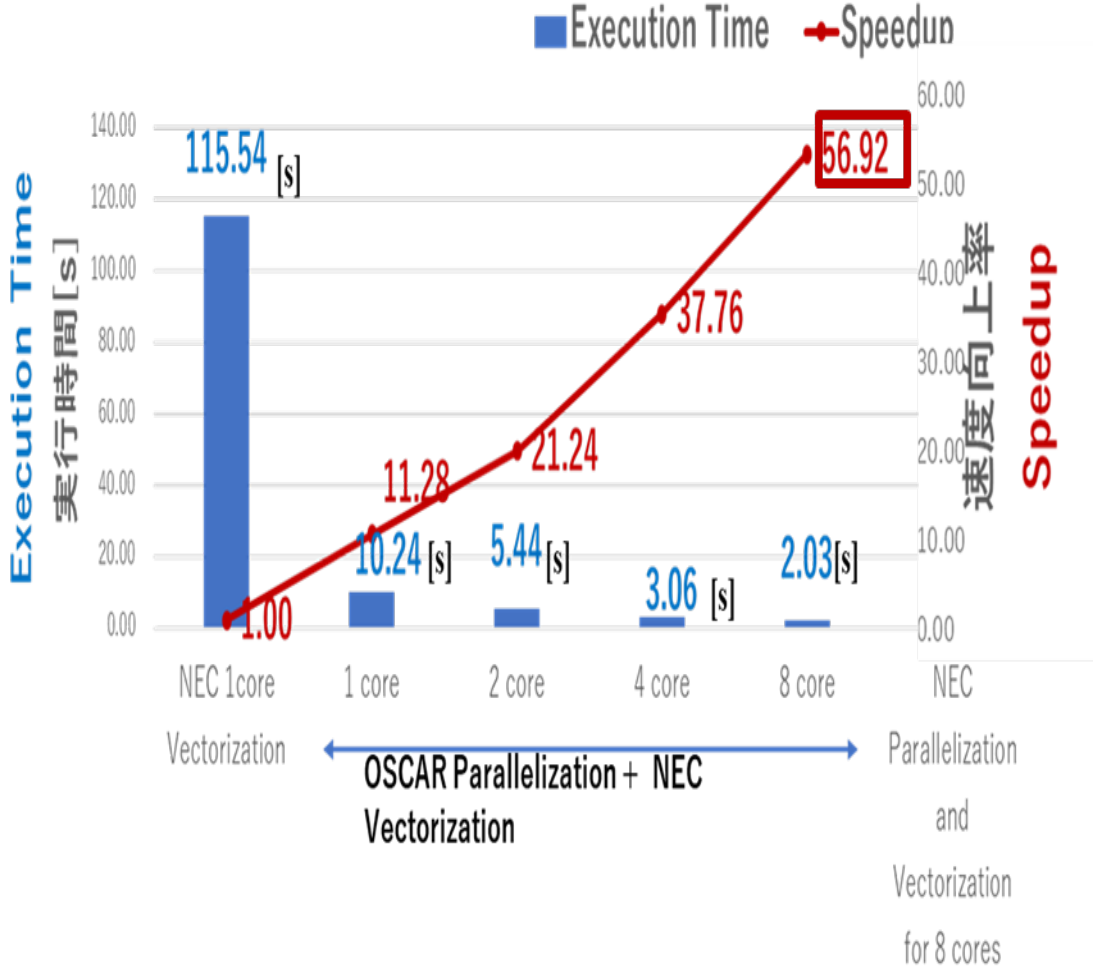




55 times speedup with 64 cores

# Speedups of NPB/CG Scientific Code by OSCAR Compiler on NEC SX-Aurora TSUBASA A100-1 8 cores 10C VE

## 57 times speedup for 8 vector cores by OSCAR Parallelization & NEC Vectorization against NEC 1 core Vectorization



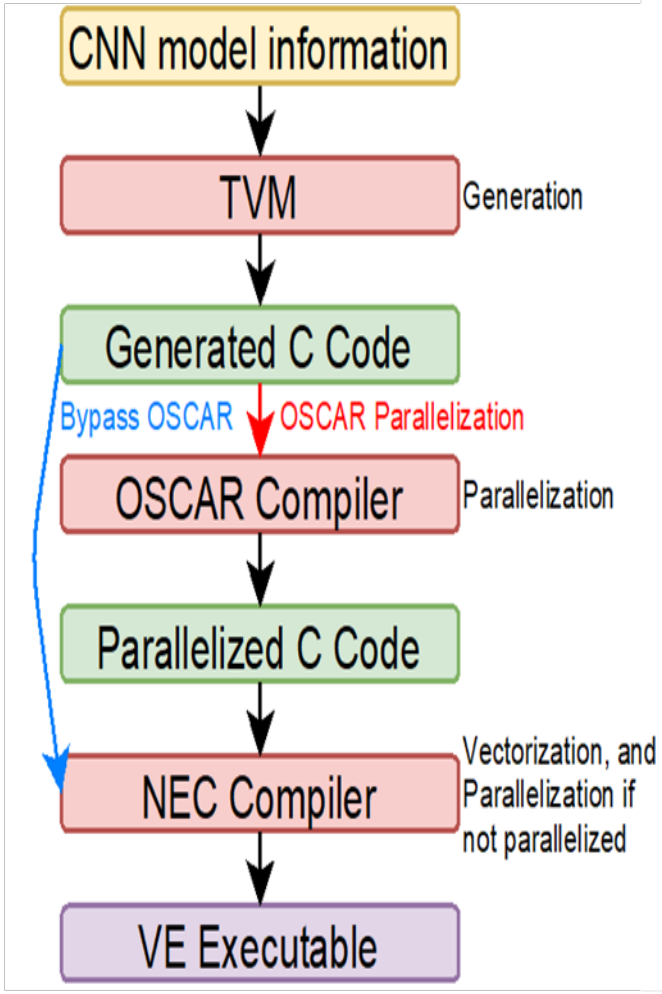



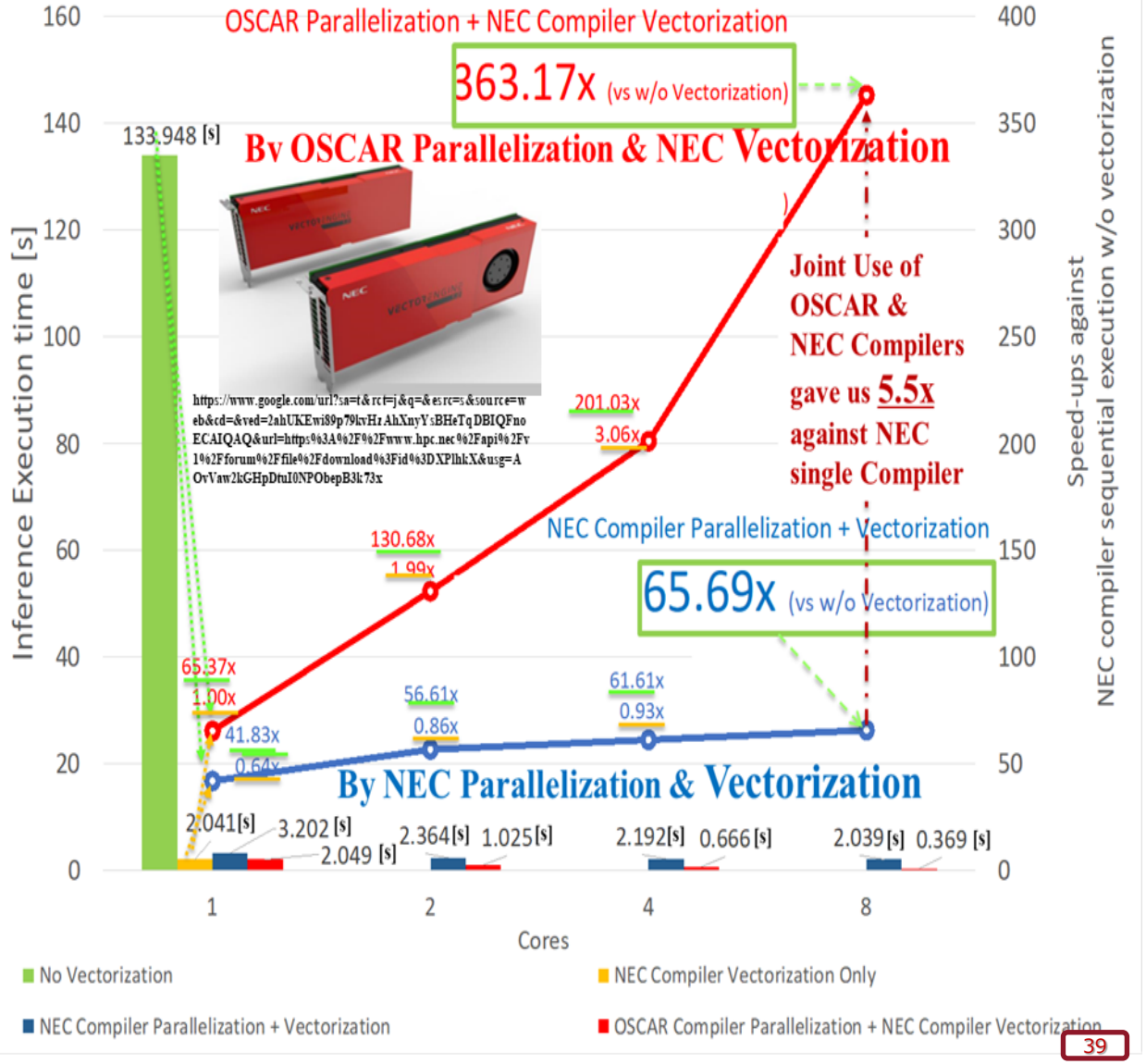

NEC Corporation. (2021) SX-Aurora TSUBASA Performance Tuning Guide. [Online]. Available: https://www.hpc.nec/documents/guide/pdfs/AuroraVE TuningGuide.pdf

# Speedups of Deep Learning Winograd 2D-Convolution generated by TVM on NEC Personal Vector Supercomputer SX-Aurora TSUBASA 8 Core Type 10C

## OSCAR Parallelization and NEC Vectorization gave us 363x Speedup against a Scalar Core

### Parallelization of Deep Leaning C Code generated by TVM

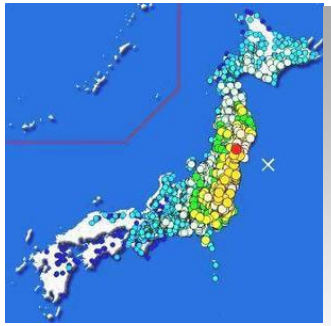


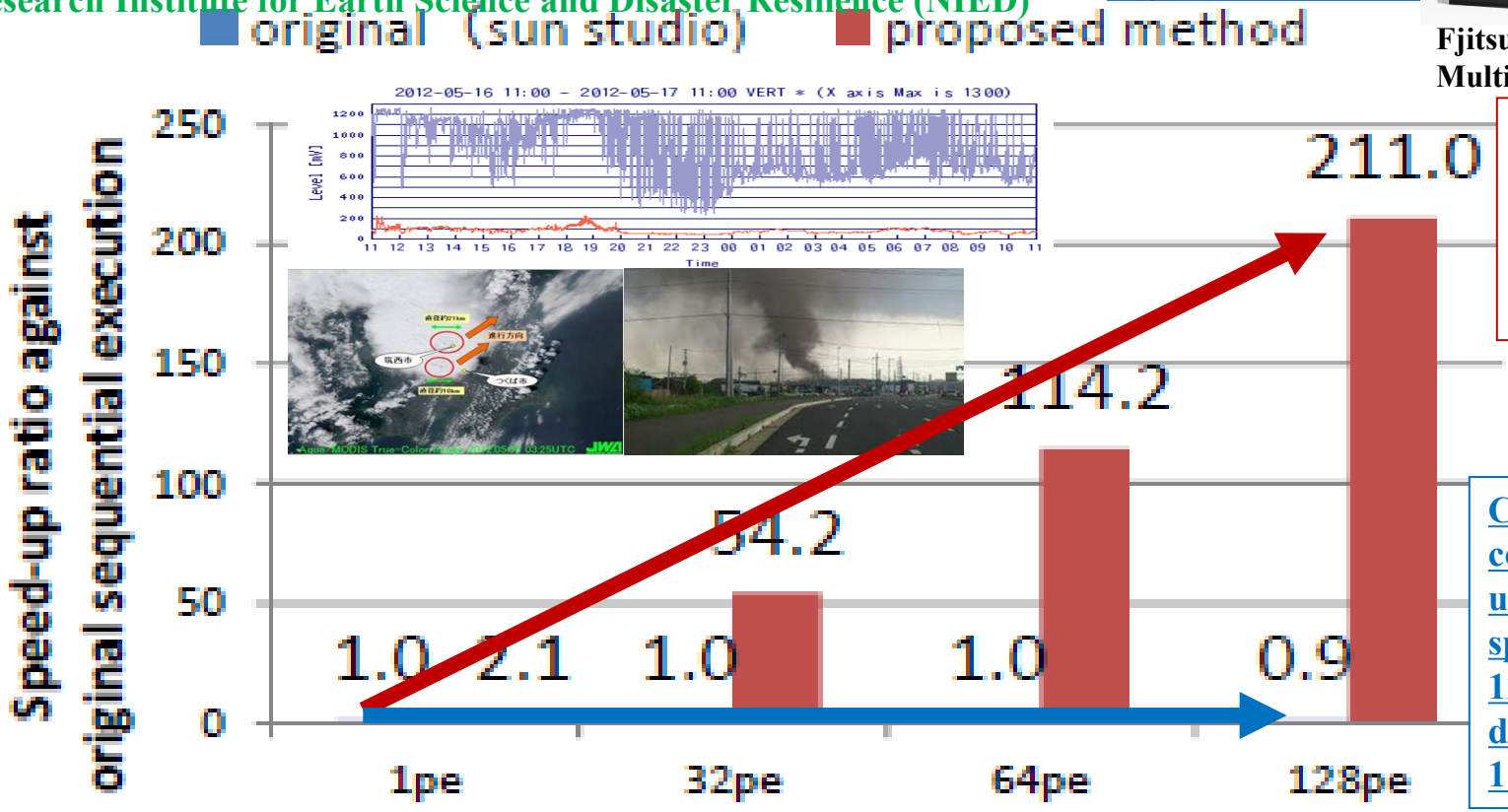
# Parallel Soft is important for scalable performance of multicore (LCPC2015)

- 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



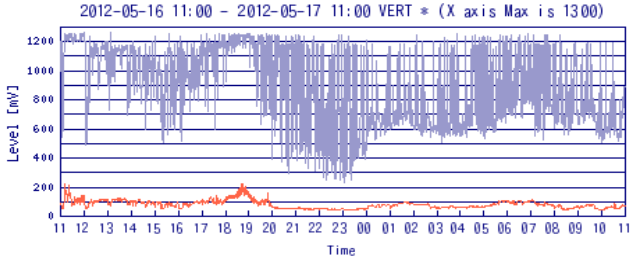

OSCAR Compiler gives us 211 times speedup with 128 cores

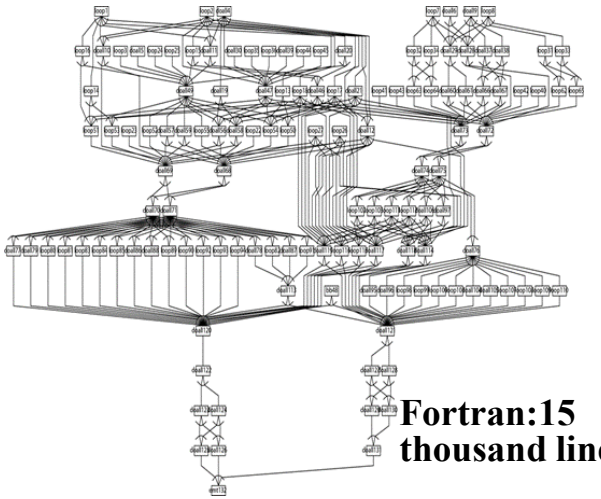
Commercial compiler gives us 0.9 times speedup with 128 cores (slow-downed against 1 core)

- 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

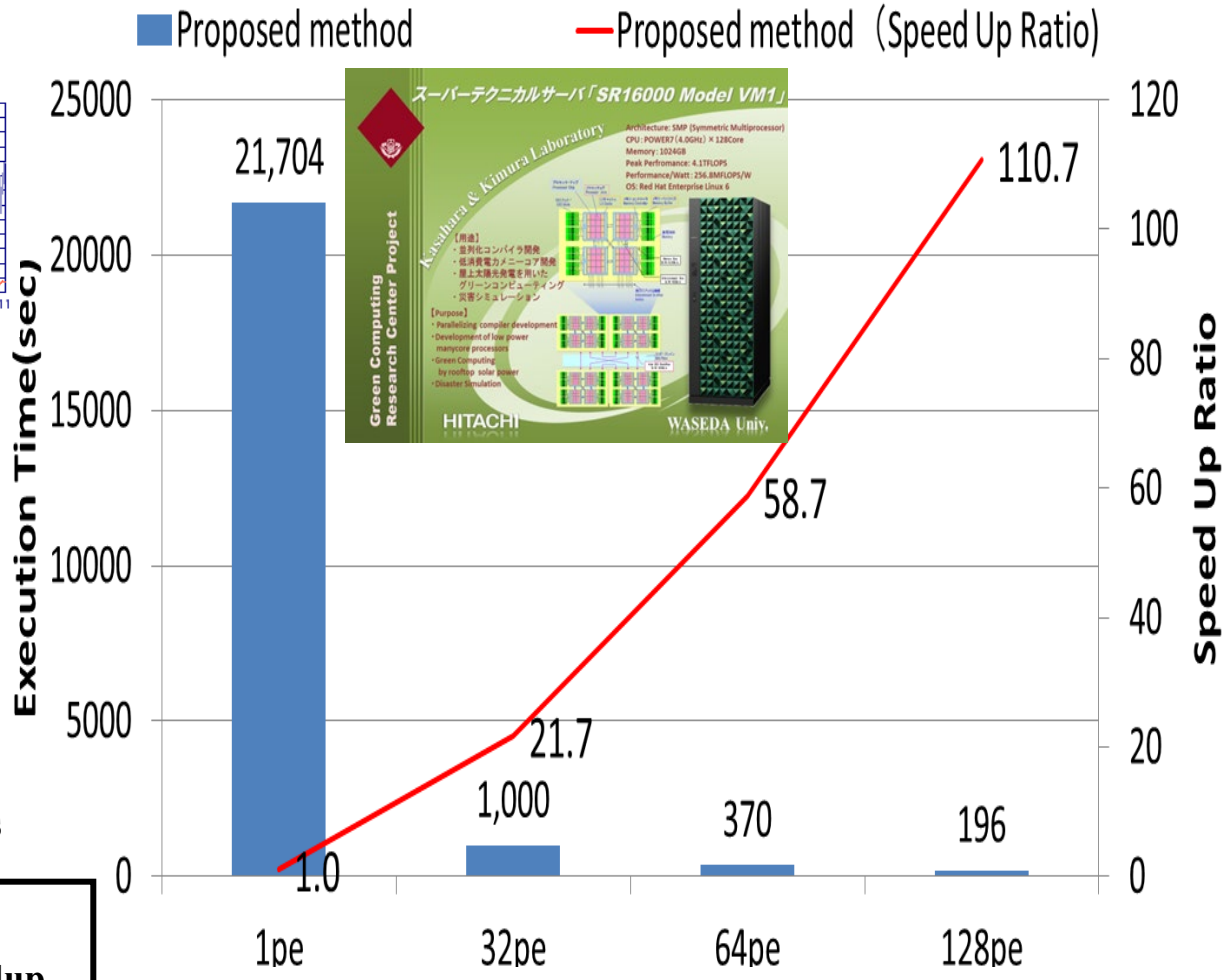
# 110 Times Speedup against the Sequential Processing for GMS Earthquake Wave Propagation Simulation on Hitachi SR16000

## （Power7 Based 128 Core Linux SMP） (LCPC2015)



Fortran:15 thousand lines

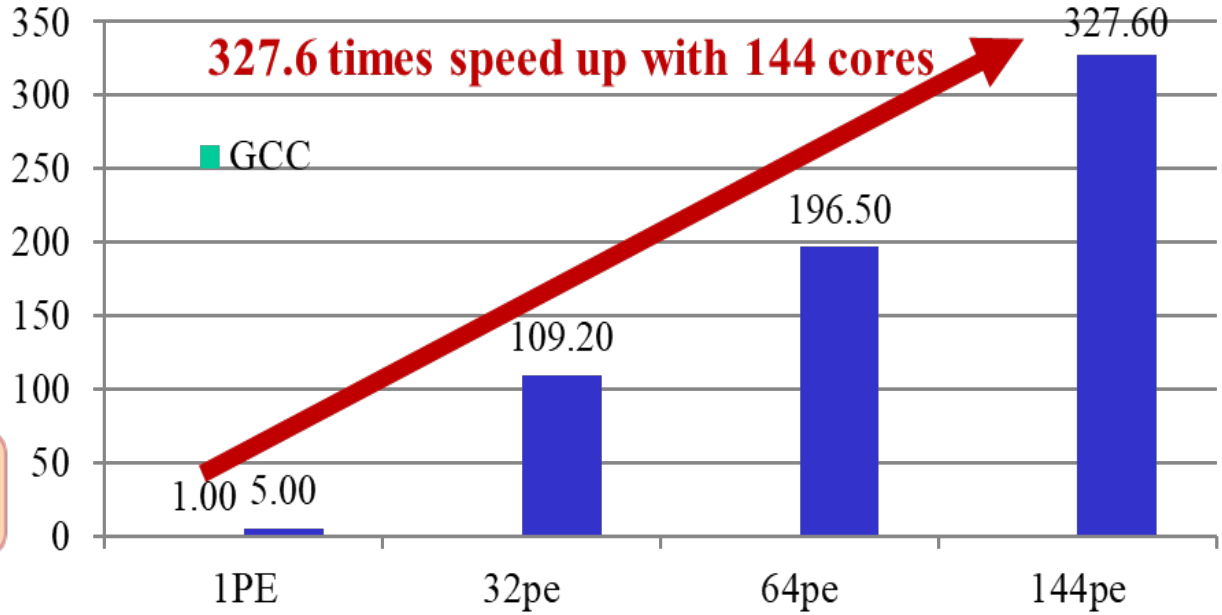



**First touch for distributed shared memory and cache optimization over loops are important for scalable speedup**

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

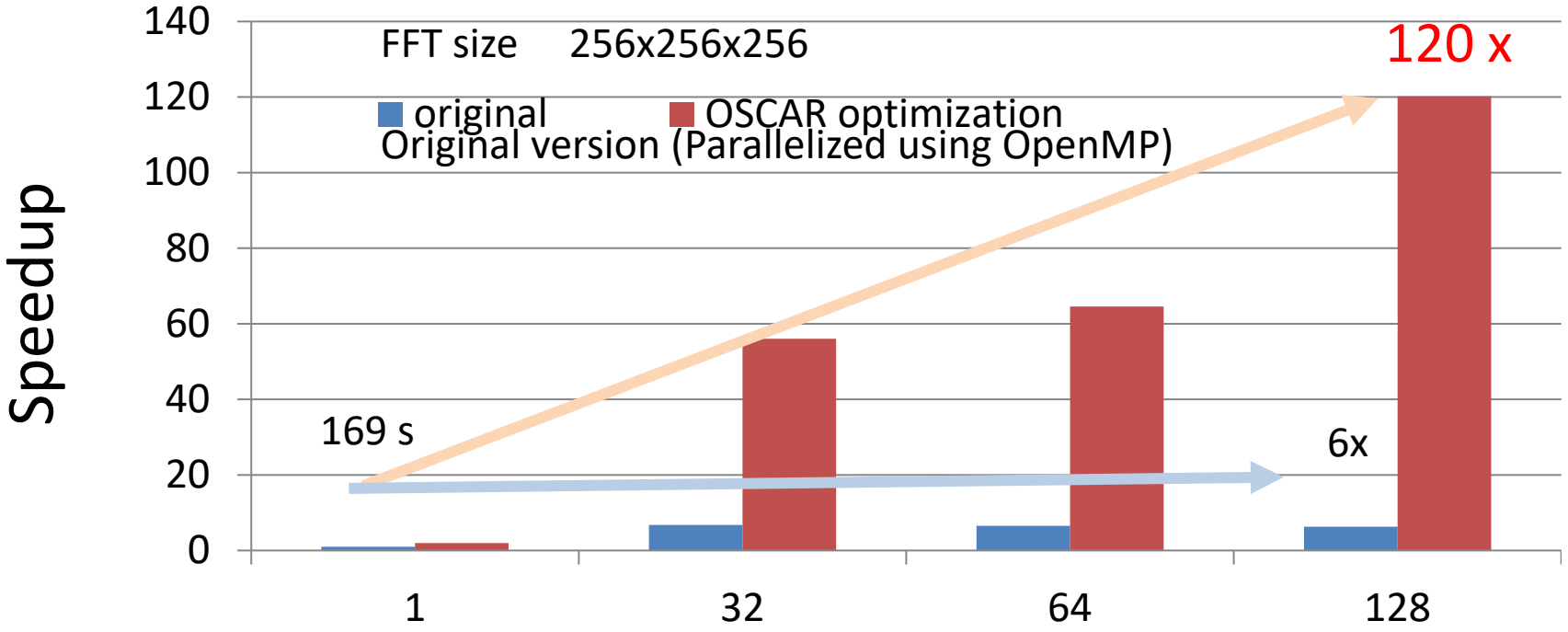
## 327 times speedup on 144 cores


**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

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



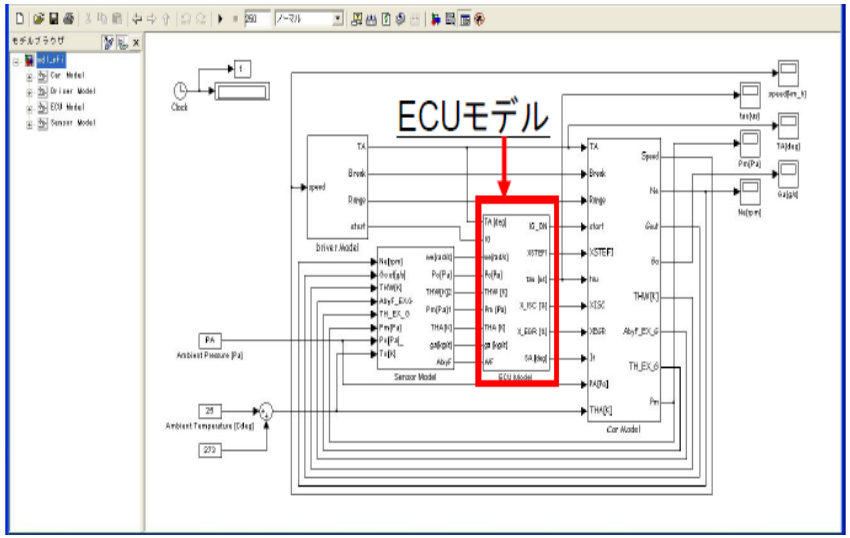

OSCAR optimization

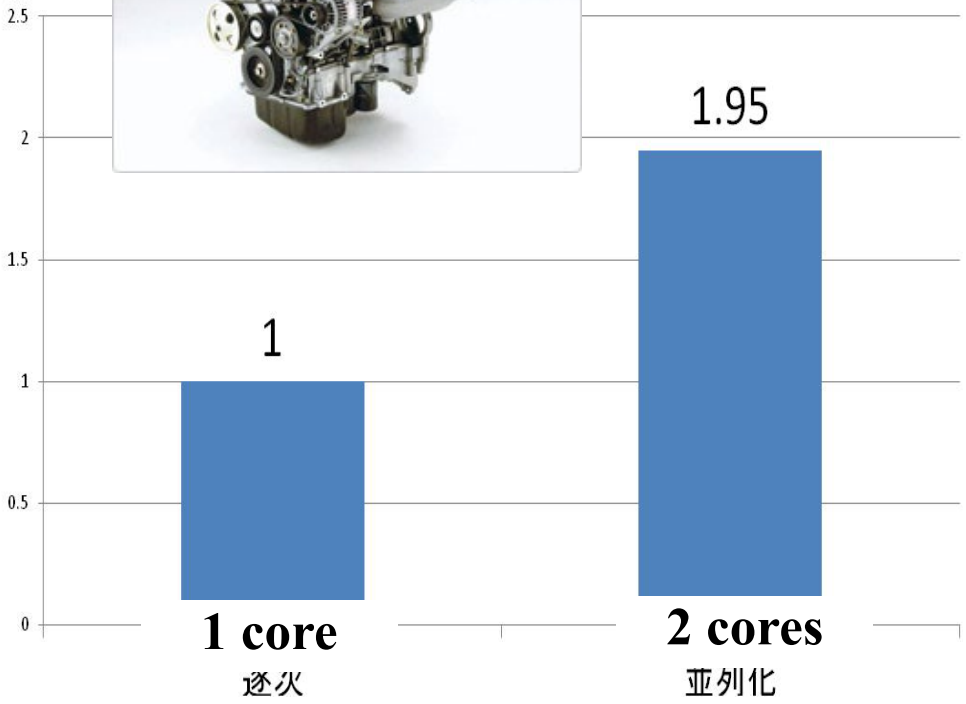
- reducing number of data transpose with interchange, code motion and loop fusion

# Engine Control by multicore with Denso

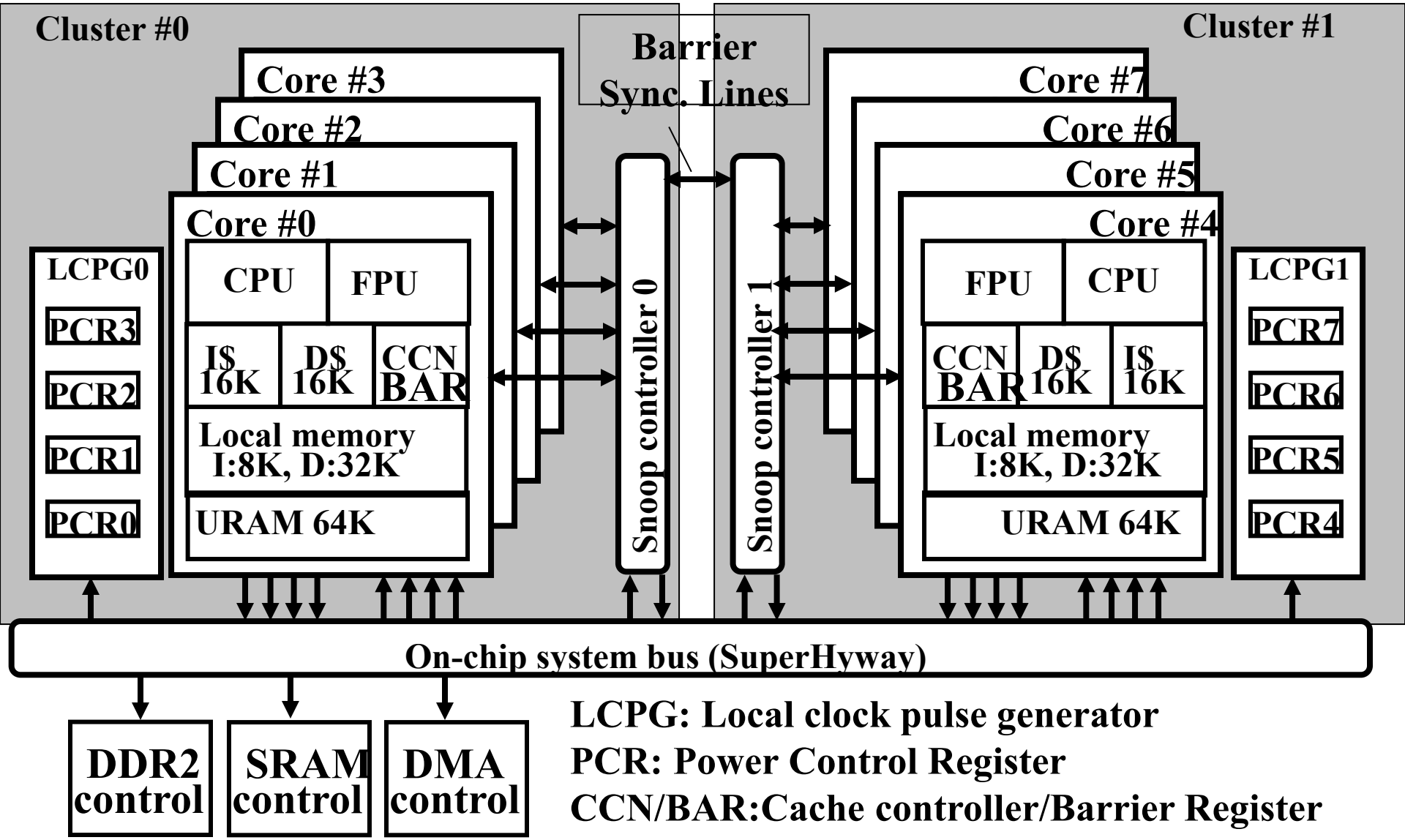
**Though so far parallel processing of the engine control on multicore has been very difficult, Denso and Waseda succeeded 1.95 times speedup on 2core V850 multicore processor.**


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



# 8 Core RP2 Chip Block Diagram



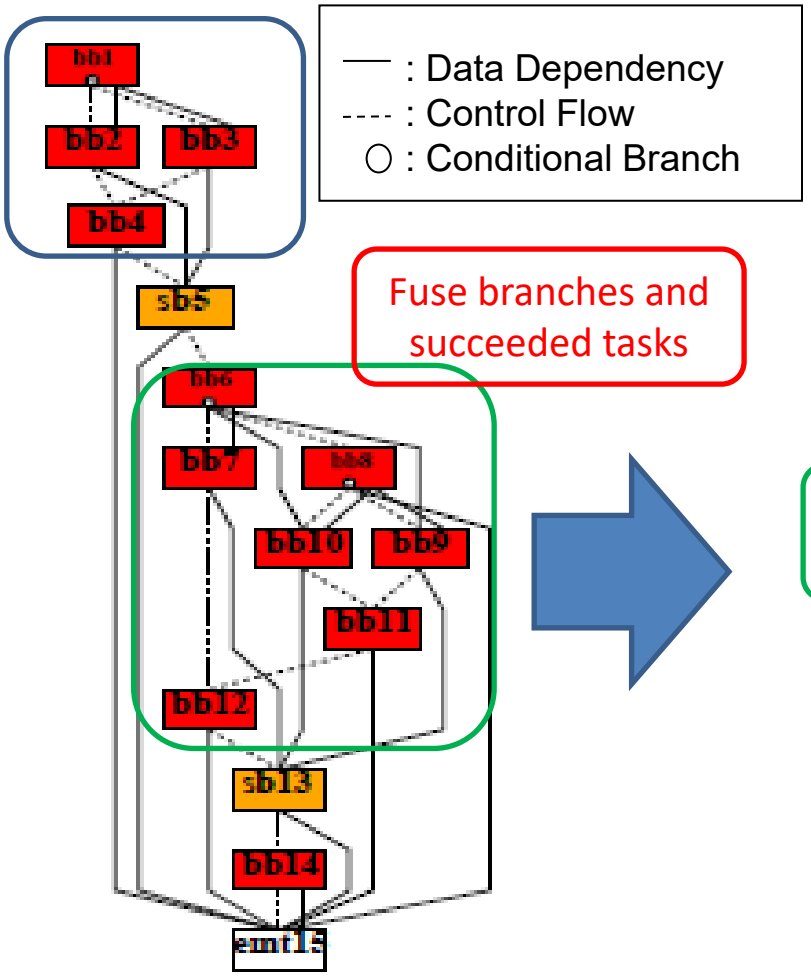

LCPG: Local clock pulse generator

PCR: Power Control Register

CCN/BAR:Cache controller/Barrier Register

URAM: User RAM (Distributed Shared Memory)

# Macro Task Fusion for Static Task Scheduling

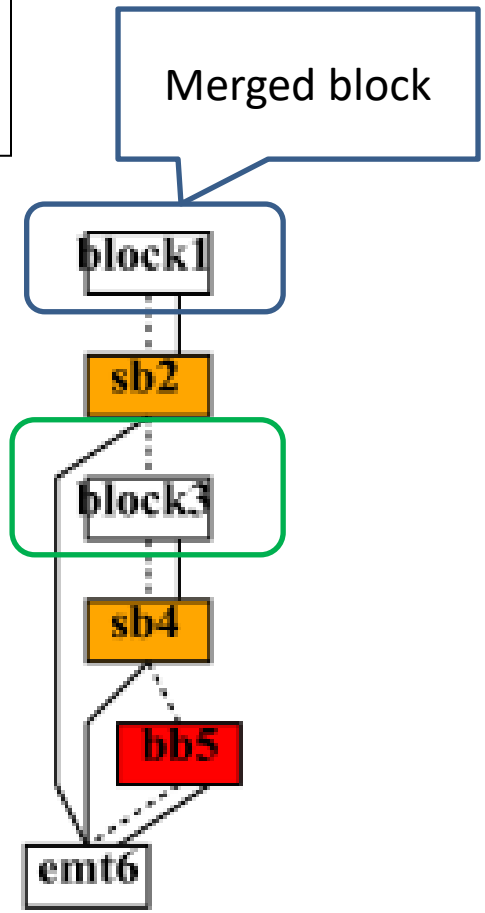



MFG of sample program before maro task fusion

: Data Dependency
: Control Flow
: Conditional Branch

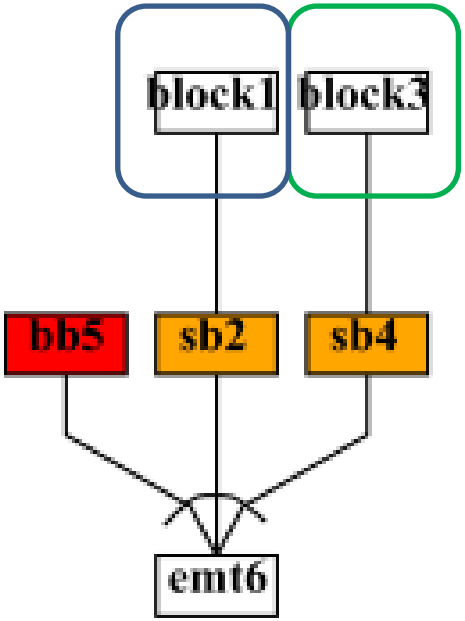
Fuse branches and succeeded tasks

Merged block




MFG of sample program after macro task fusion

Only data dependency



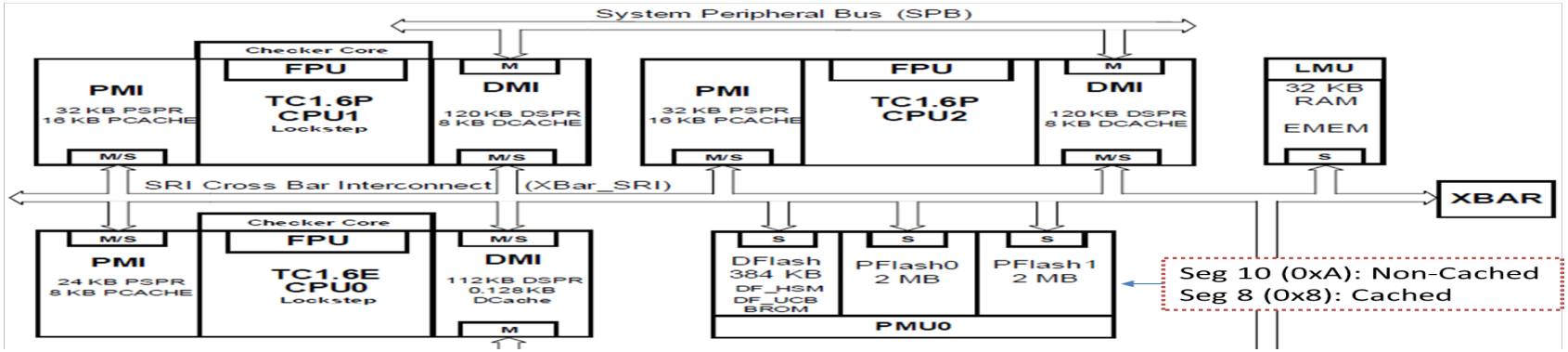

MTG of sample program after macro task fusion

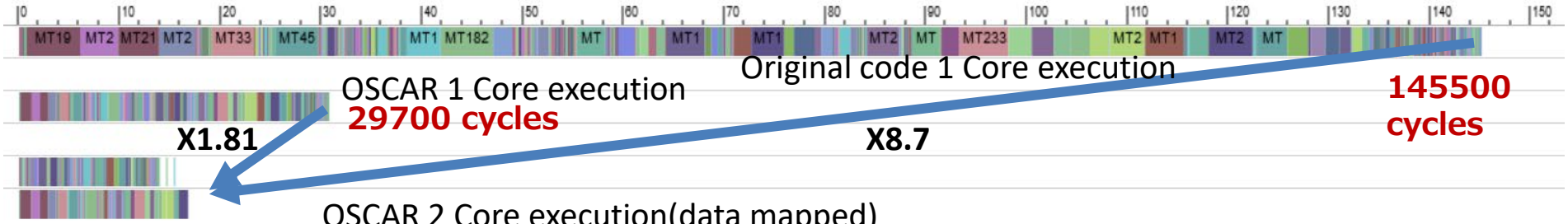
# Automatic Parallelization of an Engine Control C Program with 400 thousands lines on AUTOSAR on 2 cores of Infineon AURIX TC277

## Infineon AURIX TC277

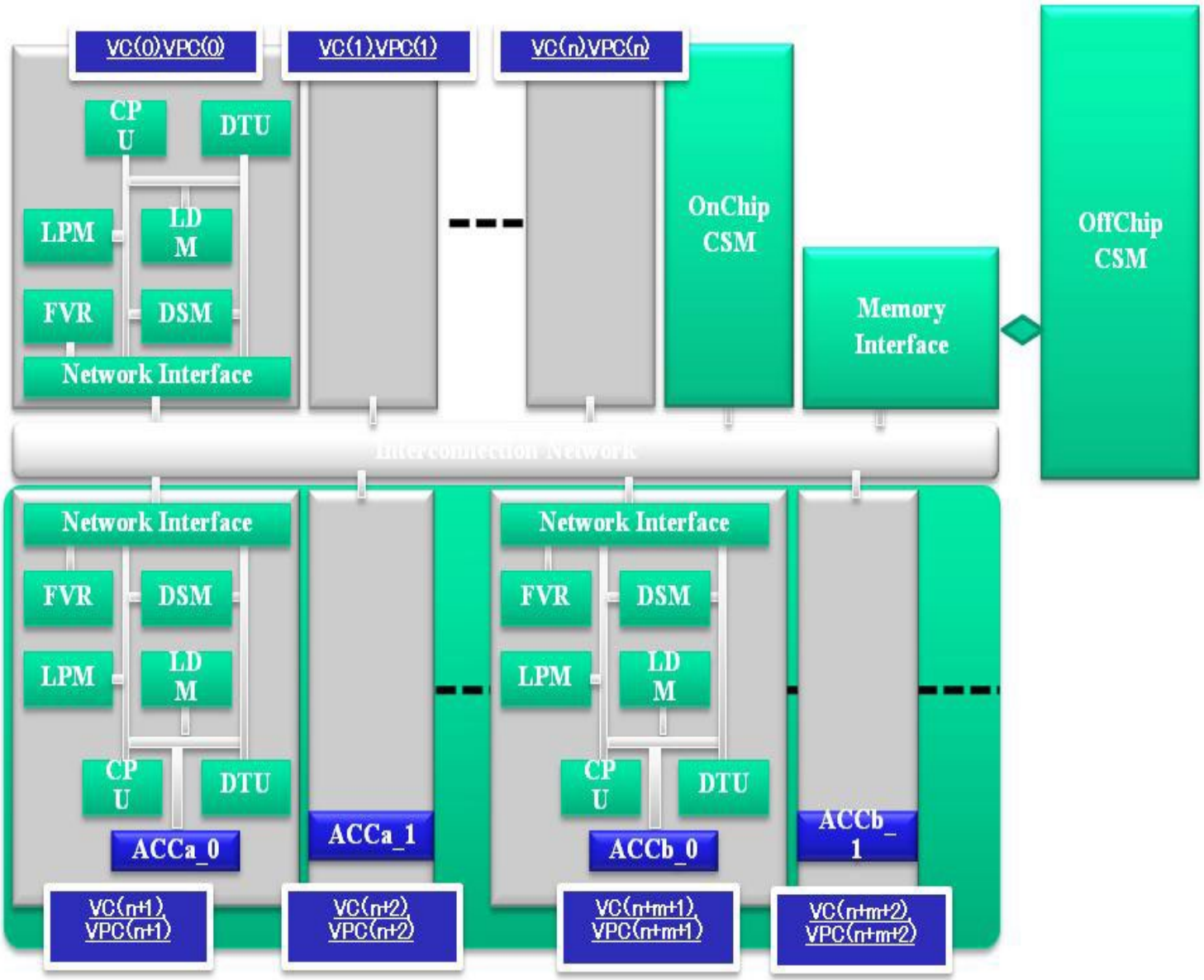
Abbreviations :

| | |
|---|---|
| PCACHE: | Program Cache |
| DCACHE : | Data Cache |
| DSPR: | Data Scratch-Pad RAM |
| PSPR: | Program Scratch-Pad RAM |
| BROM: | Boot ROM |
| PFlash: | Program Flash |
| DFlash: | Data Flash (EEEPROM) |
| S : | SRI Slave Interface |
| M : | SRI Master Interface |



# 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

# 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

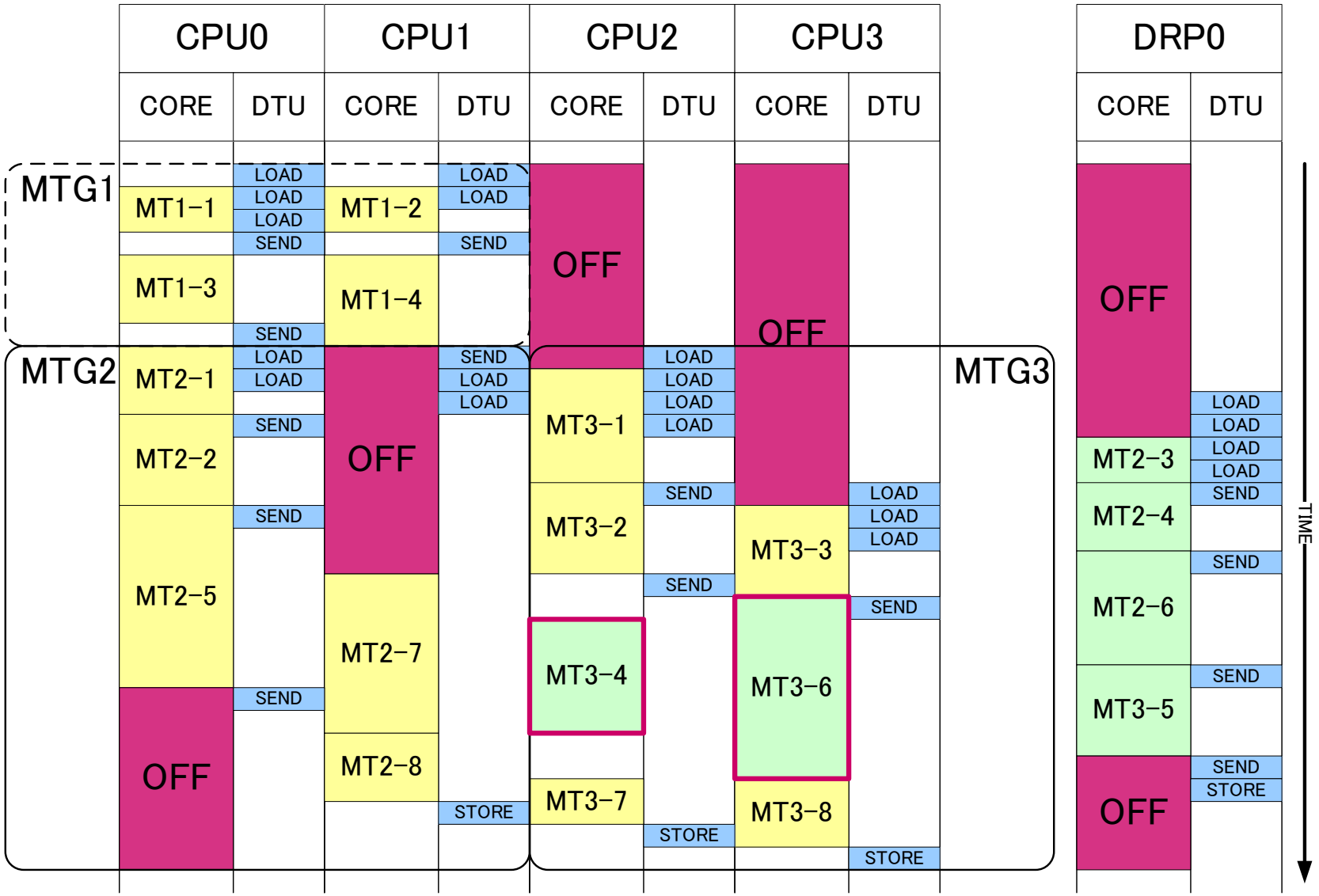
(* from OpenMP)

- 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

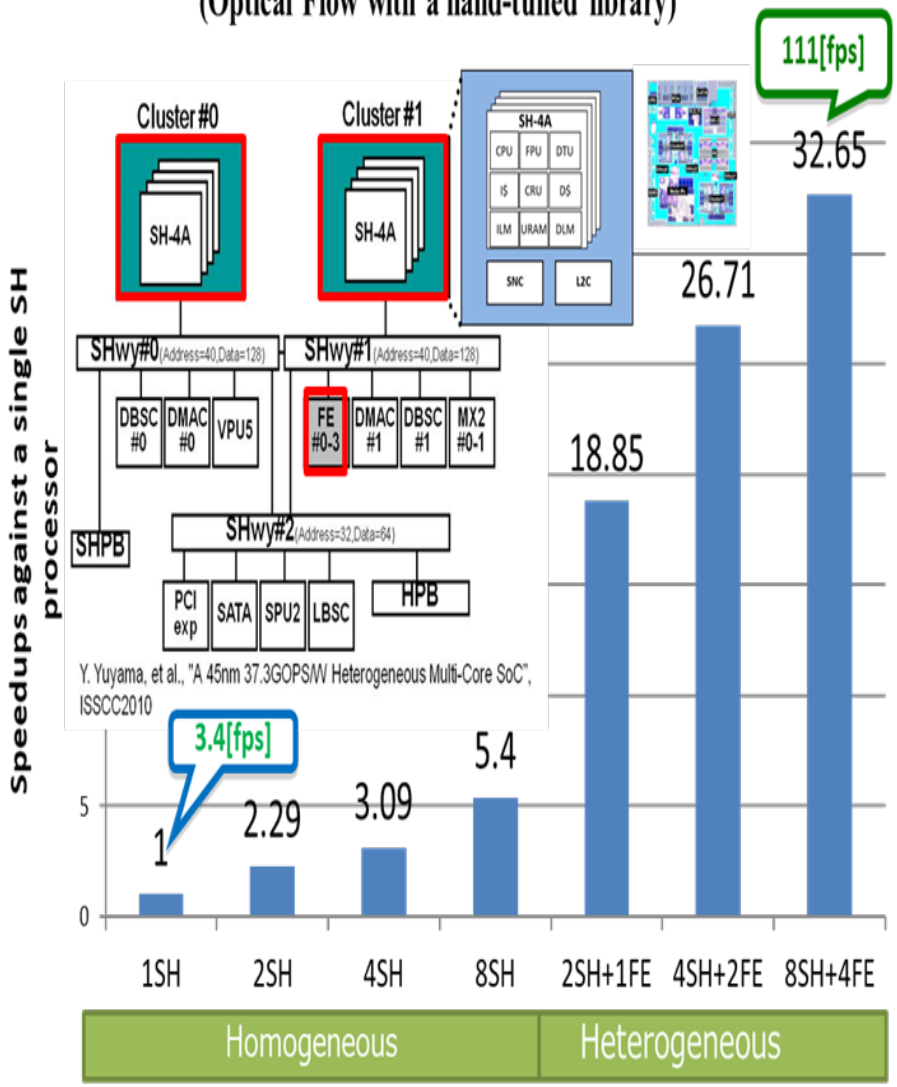
# An Image of Static Schedule for Heterogeneous Multi-core with Data Transfer Overlapping and Power Control



# Speedups & Power Reduction on RP-X Heterogeneous Multicore with 8 CPUs and 4 DRPs

## 33 Times Speedup Using OSCAR Compiler and API on Renesas RP-X with 8 CPUs & 4 DRP Accelerators

(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)

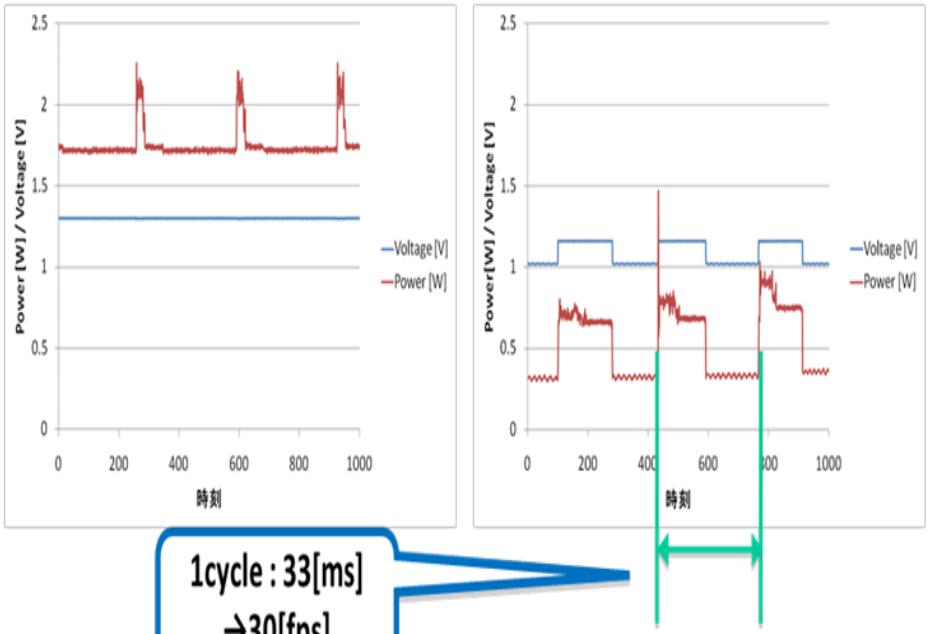
Without Power Reduction

With Power Reduction by OSCAR Compiler

70% of power reduction

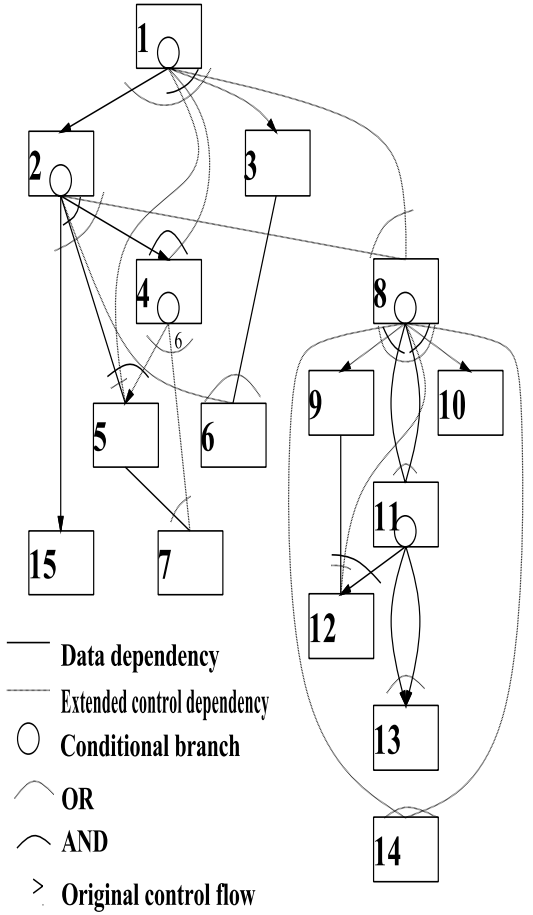
Average:1.76[W]

Average:0.54[W]



# Software Coherence Control Method on OSCAR Parallelizing Compiler

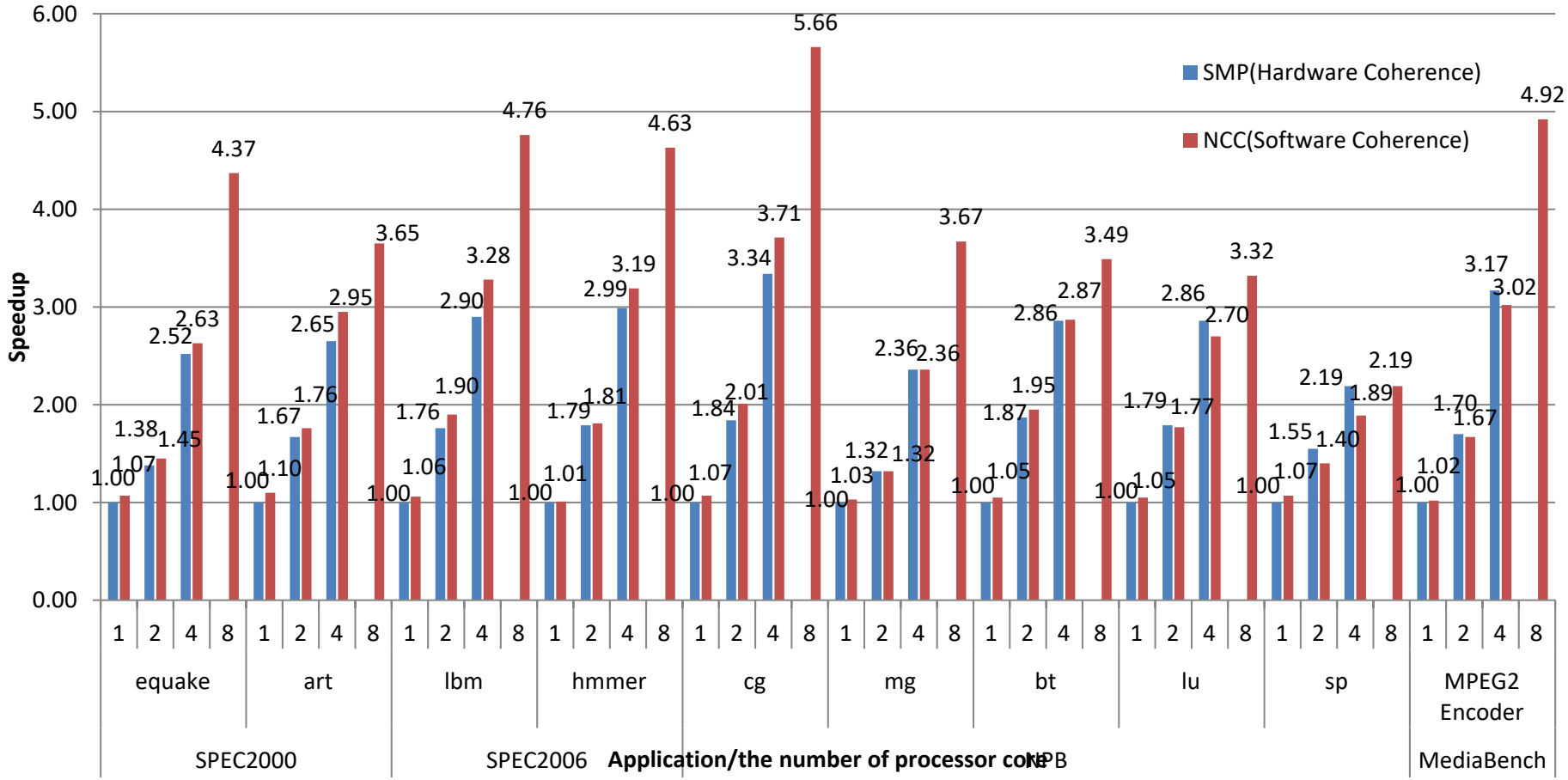
- Coarse grain task parallelization with earliest condition analysis (control and data dependency analysis to detect parallelism among coarse grain tasks).
- OSCAR compiler automatically controls coherence using following simple program restructuring methods:
  - To cope with stale data problems:
  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




**MTG generated by earliest executable condition analysis**

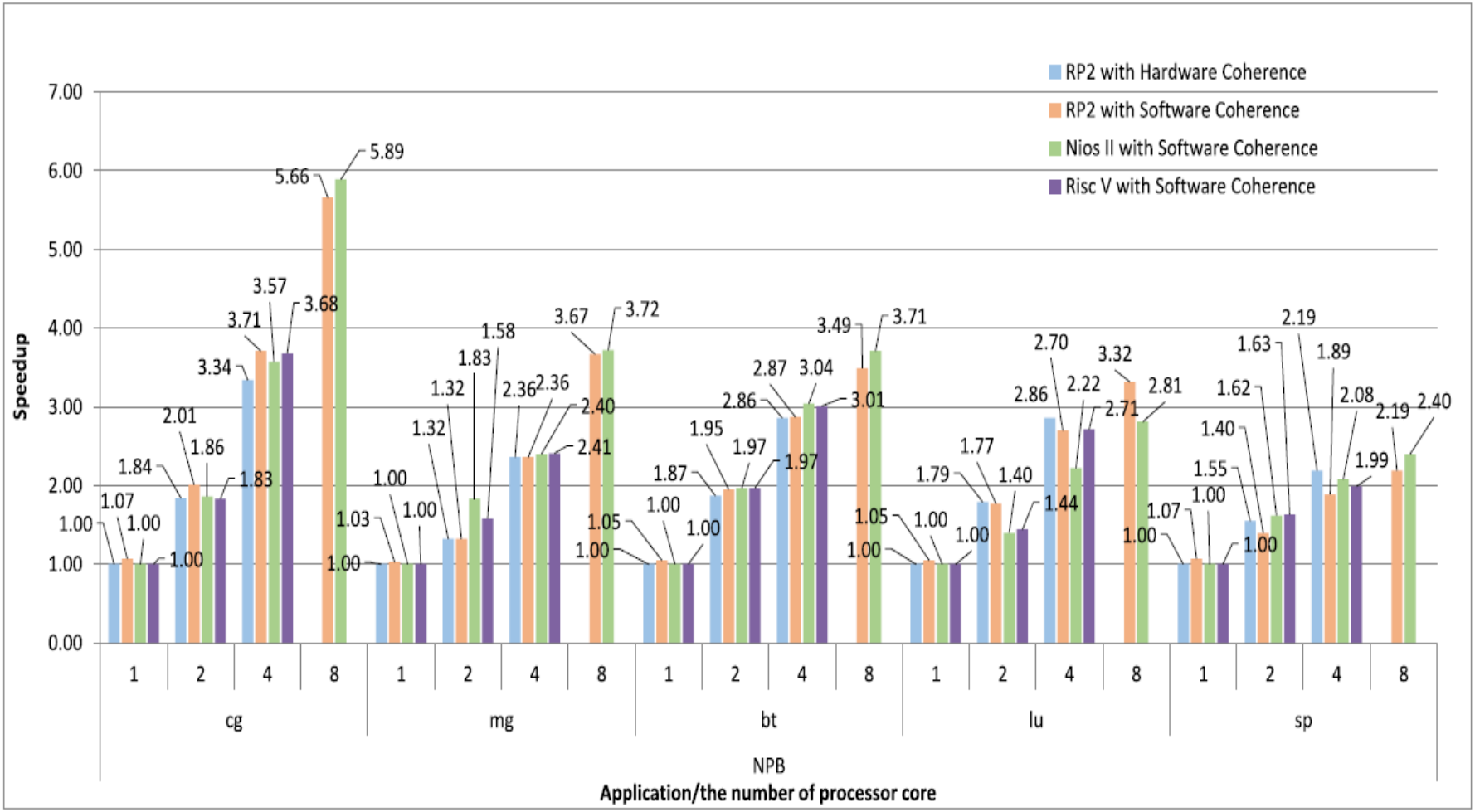
# Automatic Software Coherent Control for Manycores

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



# OSCAR Software Cache Coherent Control for NIOS and RISCV cores on FPGA

## 3.57x Speedups for NIOS and 3.68x for RISCV using 4 cores for NPB CG

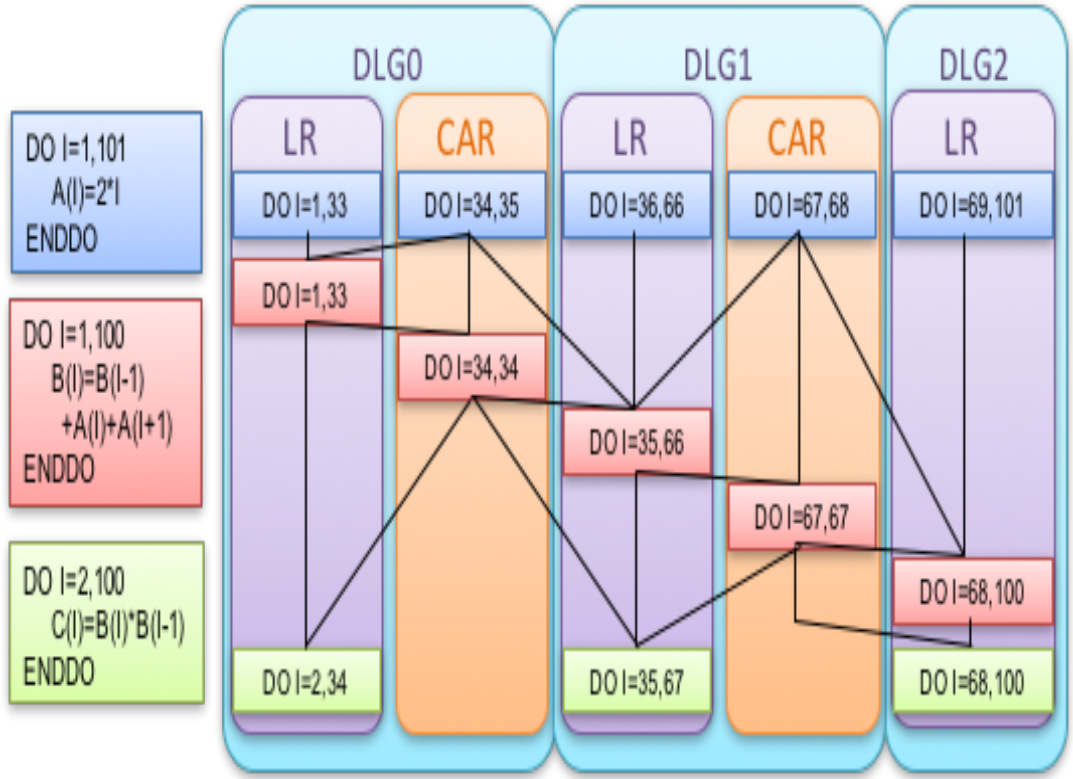


# Automatic Local Memory Management

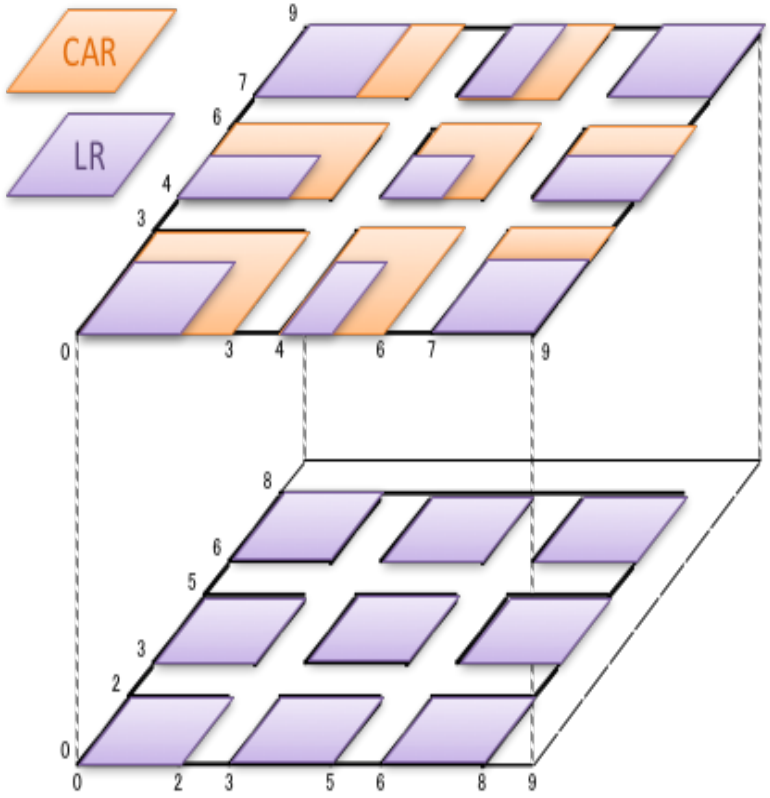
## Data Localization: Loop Aligned Decomposition

- **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**

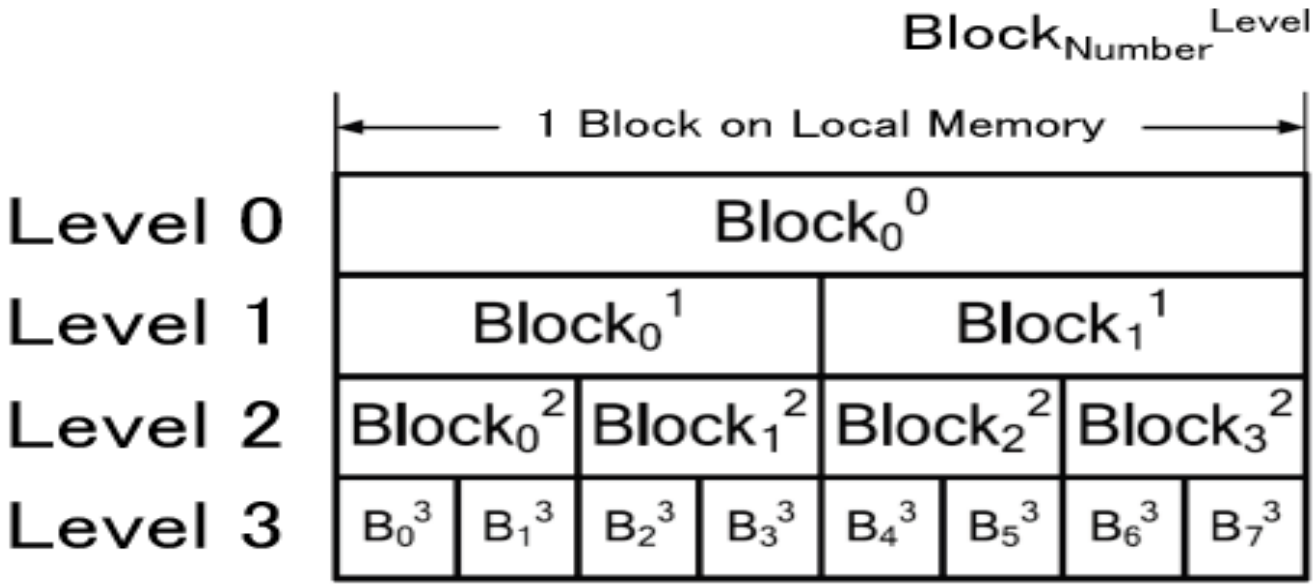



**Multi-dimension Decomposition**



# Adjustable Blocks

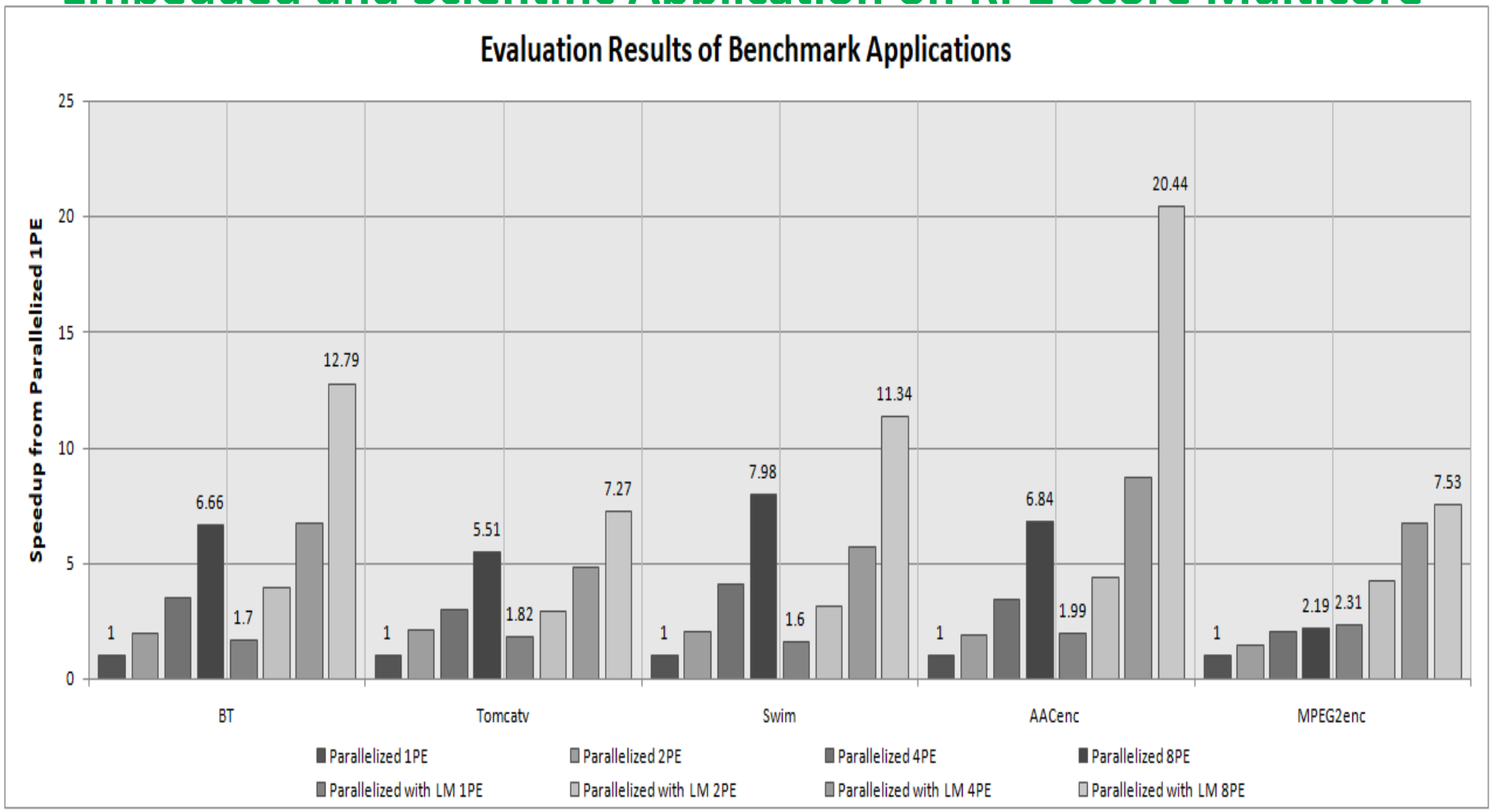
- Handling a suitable block size for each application
  - different from a fixed block size in cache
  - each block can be divided into smaller blocks with integer divisible size to handle small arrays and scalar variables



# 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 OSCAR Automatic Local Memory Management compared to Executions Utilizing Centralized Shared Memory on Embedded and Scientific Application on RP2 8core Multicore

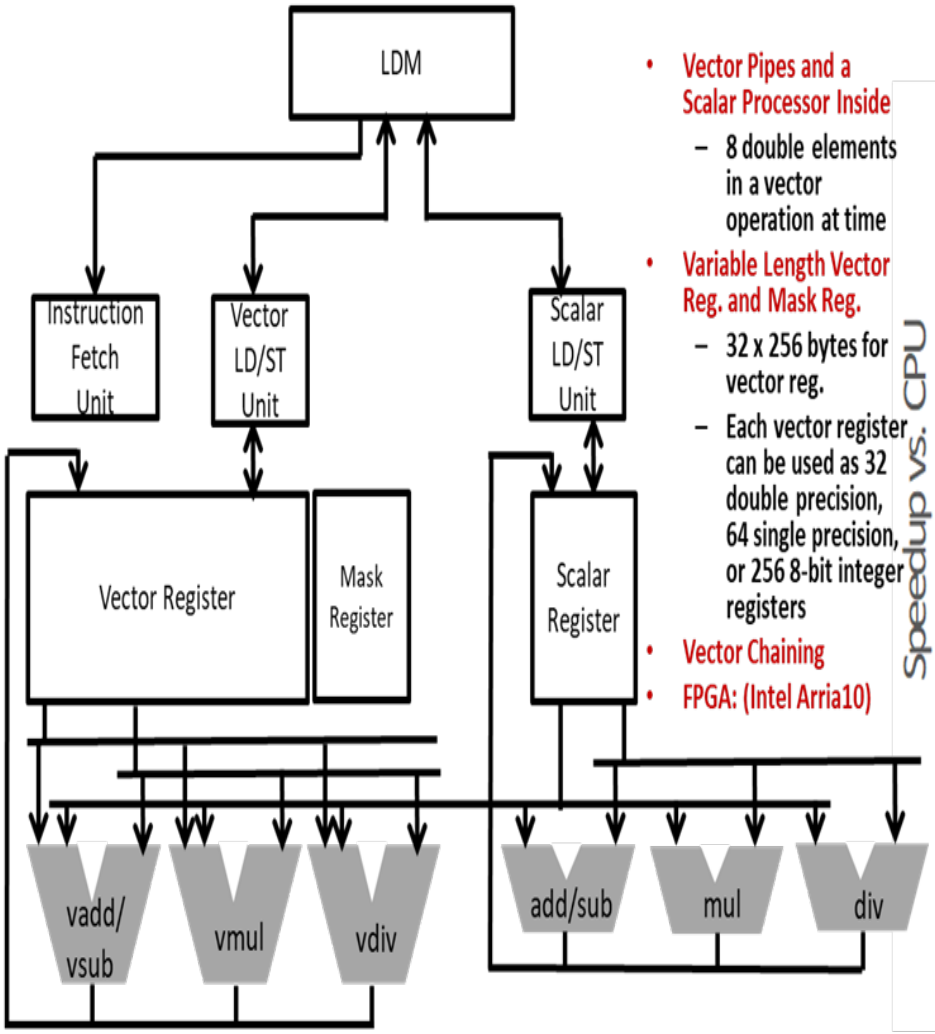



**Maximum of 20.44 times speedup on 8 cores using local memory against sequential execution using off-chip shared memory**

# Performance for Multimedia & Scientific Applications on OSCAR Vector Accelerator

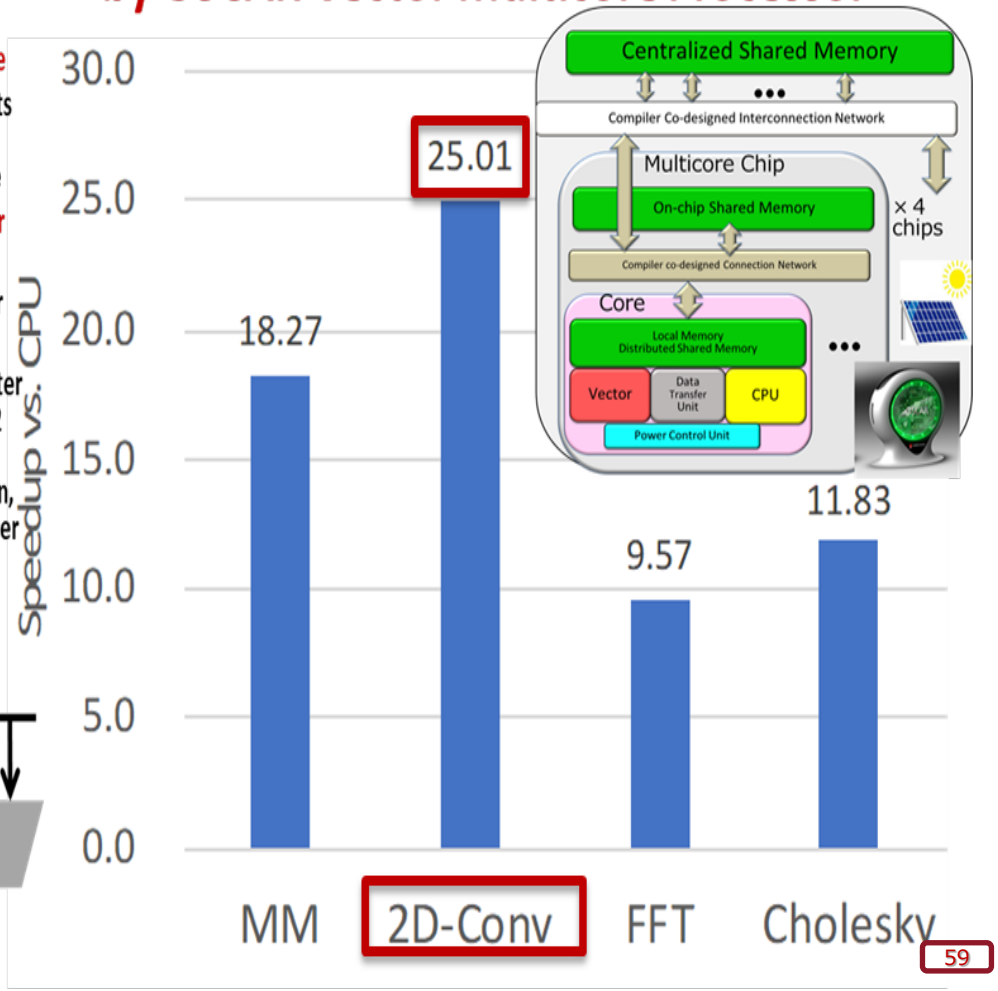
**(A Vector Processor with Local Memory or Distributed Shared Memory and DMA Controller Managed by the OSCAR Compiler Redesigned Improving Japanese Supercomputer Technology in 1980-2000)**

## OSCAR Vector Accelerator on FPGA

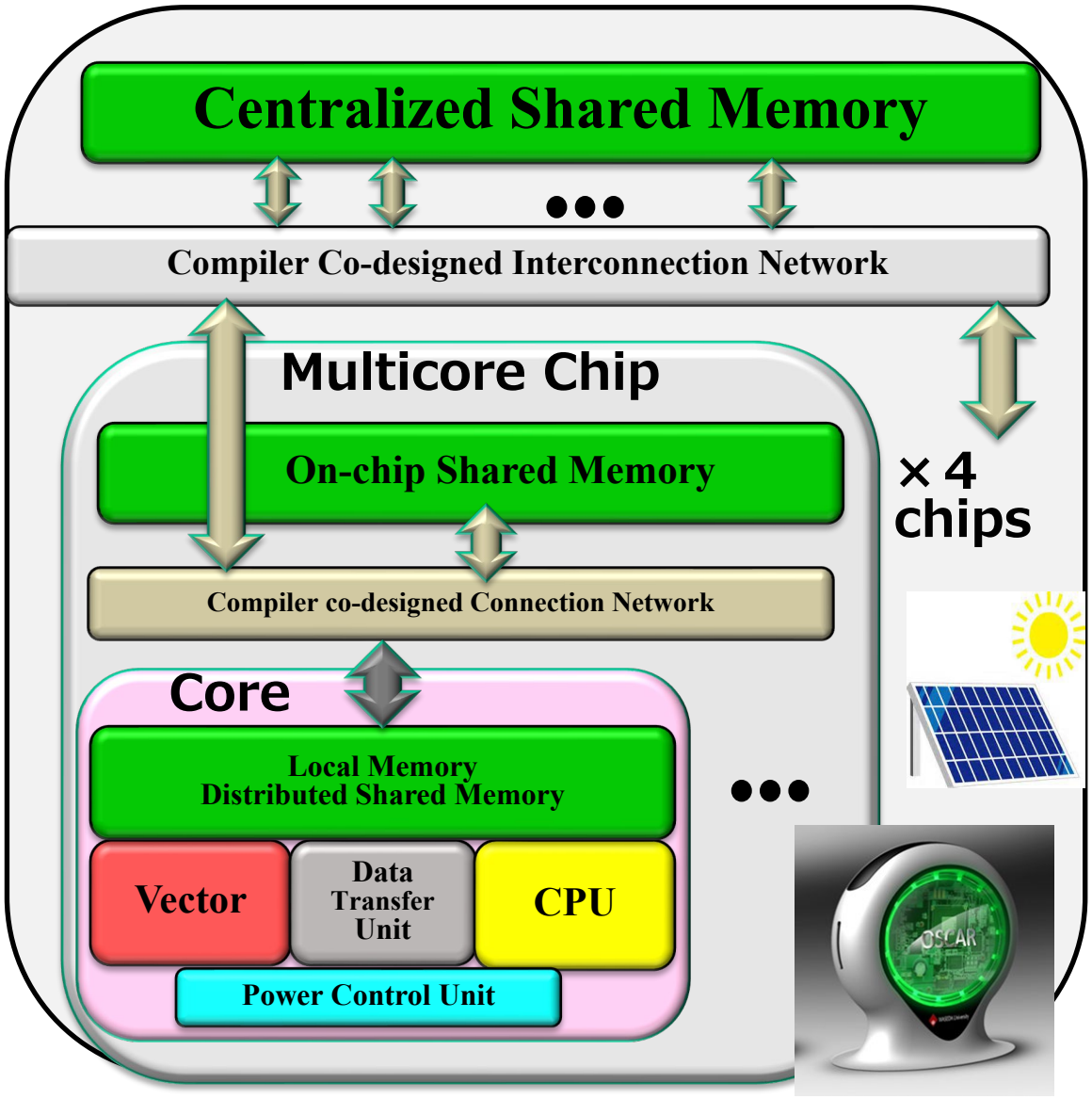



- Vector Pipes and a Scalar Processor Inside
  - 8 double elements in a vector operation at time
- Variable Length Vector Reg. and Mask Reg.
  - 32 x 256 bytes for vector reg.
  - Each vector register can be used as 32 double precision, 64 single precision, or 256 8-bit integer registers
- Vector Chaining
- FPGA: (Intel Arria10)

## Speedups against General Purpose Processor by OSCAR Vector Multicore Processor



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









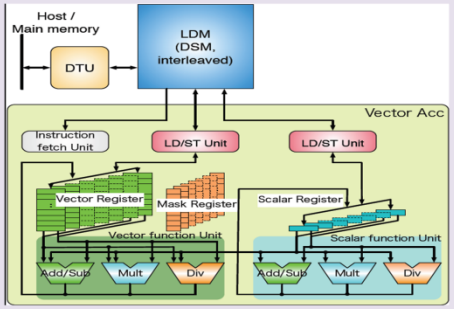
**Target:**

- **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

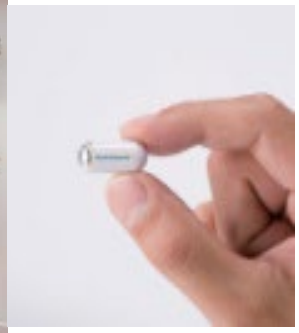
## Next Generation Automobiles

- 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