組込み向けマルチ・メニーコア用並列化コンパイラ技術

早稲田大学　情報理工学科　教授　笠原博徳

アドバンスト・マルチコア・プロセッサ研究所　所長

IEEE Computer Society President 2018

1980 早大電気工学科卒，1982同修士了
1985 早大大学院博士課程了　工学博士
　　カリフォルニア大学バークレー客員研究員
1986 早大理工専任講師，1988年 助教授
1997 教授、現在 理工学術院情報理工学科
1989～1990 イリノイ大学Center for
　　Supercomputing　R&D客員研究員
2004　アドバンストマルチコア研究所所長
2017 日本工学アカデミー,日本学術会議連携会員

1987 IFAC World Congress Young Author Prize
1997 情報処理学会坂井記念特別賞
2005 半導体理工学研究センタ共同研究賞
2008 LSI・オブ・ザ・イヤー 2008 準グランプリ，
Intel Asia Academic Forum Best Research Award
2010 IEEE CS Golden Core　Member Award
2014 文部科学大臣表彰科学技術賞研究部門
2015 情報処理学会フェロー，
2017 IEEE Fellow，2017 IEEE Eta-Kappa-Nu

査読付き論文216件，招待講演165件，
特許取得45件(日本・米国・英国・中国等)，
新聞・Web記事・TV等メディア掲載 594件

政府・学会委員等歴任数　245件
IEEE Computer Society President 2018, Executive Committee委員長，理事(2009-14)，戦略計画委員会委員長，Multicore STC 委員長，規約委員会委員長，IEEE CS Japan 委員長 (2005-07) 等
【経済産業省・NEDO】 情報家電用マルチコア&アドバンスト並列化コンパイラプロジェクトリーダ，NEDOコンピュータ戦略委員長等
【内閣府】スーパーコンピュータ戦略委員，政府調達苦情検討委員，総合科学技術会議情報通信PT　研究開発基盤領域&セキュリティ・ソフト検討委員，日本国際賞選定委
【文部科学省・海洋研】地球シミュレータ(ES)中間評価委員、情報科学技術委員，HPCI計画推進委員,次世代スパコン(京)中間評価委員・概念設計評価委員，地球シミュレータES2導入技術アドバイザイリー委員長等

2016 IEEE Computer Society Election Results

Hironori Kasahara selected 2017 President-Elect (2018 President)

IEEE CS 70年の歴史の中で初めて、北米以外から会長に選出

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.

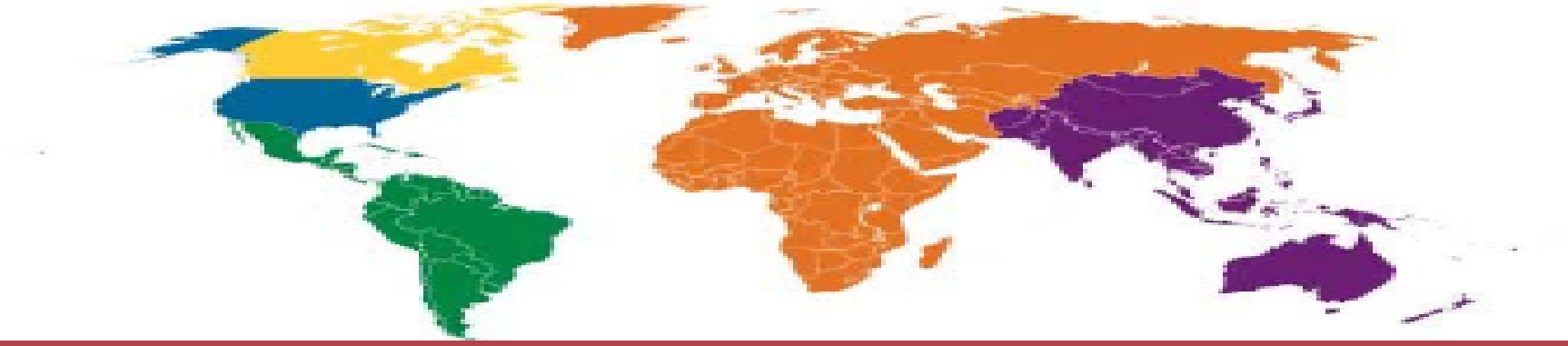
IEEE Computer Society 2018 BoG (理事)Feb.1, 2018

IEEE Computer Society

60,000+ members, volunteer-led organization,
200 technical conferences, industry-oriented "Rock Stars" ,
17 scholarly journals and 13 magazines, awards program,
Digital Library with more than 550,000 articles and papers ,
400 local and regional chapters, 40 technical committees,

- Region 1 (Northeastern US)
- Region 2 (Eastern US)
- Region 3 (Southern US)
- Region 4 (Central US)
- Region 5 (Southwestern US)
- Region 6 (Western US)
- Region 7 (Canada)
- Region 8 (Africa, Europe, Middle East)
- Region 9 (Latin America)
- Region 10 (Asia and Pacific)

- IEEE-USA (Regions 1-6)



COMPSAC2018, July 23-27, Tokyo, Japan

COMPSAC2018, July 23-27, Hitotsubashi Hall & NII, Tokyo, Japan

https://ieeecompsac.computer.org/2018/

Very successful Conference having about 500 participants with 8 CS Presidents

COMPSAC2018 Keynote Speakers

July24

Margaret Martonosi
2018 Computer Society Technical Achievement Award, Princeton Univ.

July25

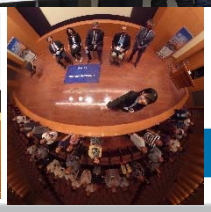
Dejan Milojicic
CS President 2014
IEEE Director, HP Labs
CS 2022 Report

July25

Hironori Kasahara
CS President 2018
Waseda Univ.

July26

Bjarne Stroustrup
2018 Computer Society Computer Pioneer Award
Morgan Stanley, Columbia Univ.



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

IEEE

IEEE CS Awards are presented in June BoG Meeting or COMPSAC

June 7, 2018, BoG in Phonix

IPSJ/IEEE-CS Young Computer Researcher Award

COMPSAC2018, July 25, Rihga Royal Hotel Tokyo, Japan

The First 3 Awardees with
IPSJ President Prof. Shoichiro Nishio and CS President Prof. Hironori Kasahara

https://www.ipsj.or.jp/release/IEEE-CS_Award2018.html

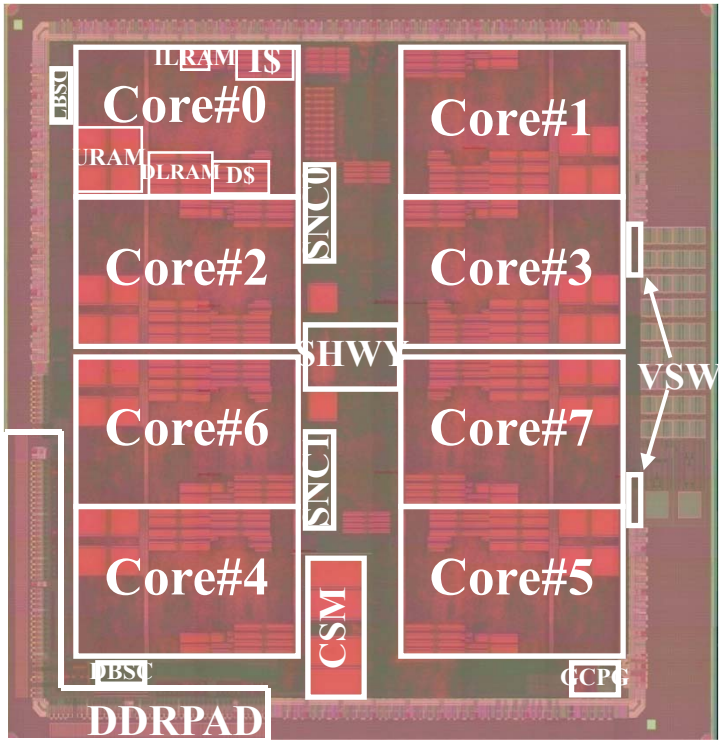
Prof. Yutaka Arakawa

Prof. Akira Kawai

Prof. Yukihiko Shigesada

July 25, 2018 Award Ceremony

ムーアの法則の終焉
コンピュータの高性能化と低消費電力化にはマルチコアが必須



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 ∝ Frequency * Voltage^2

(Voltage ∝ Frequency)

Power ∝ Frequency^3

周波数 Frequency を1/4にすると

(Ex. 4GHz→1GHz),

消費電力は 1/64に削減

性能は 1/4に低下 .

<マルチコア>

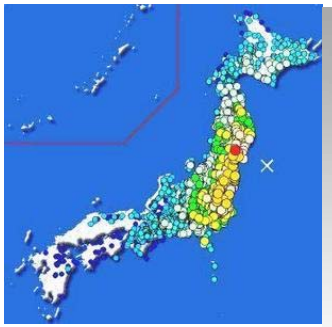
8cores をチップに集積すると,

電力は 依然1/8で性能 は2 倍向上

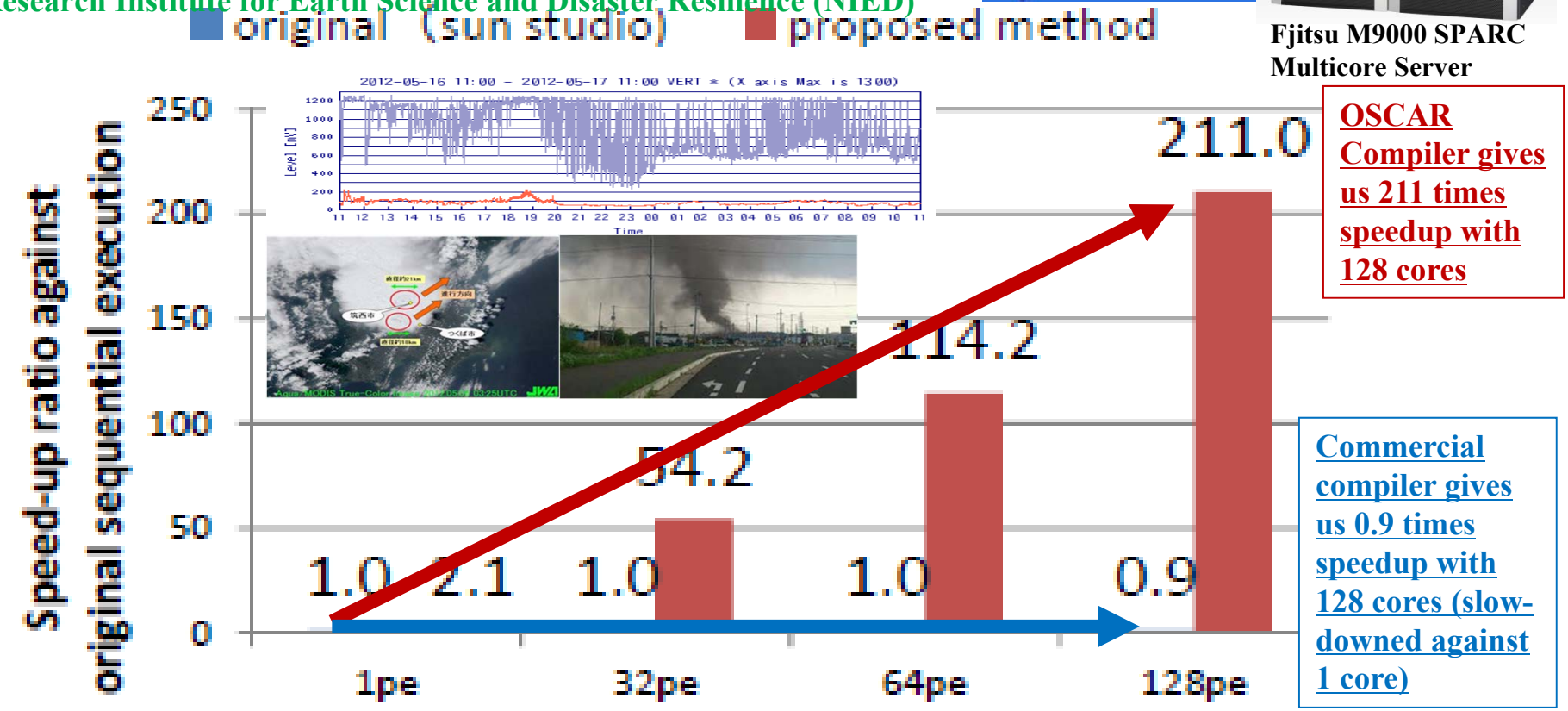
Parallel Soft is important for scalable performance of multicore (LCPC2015)

- 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

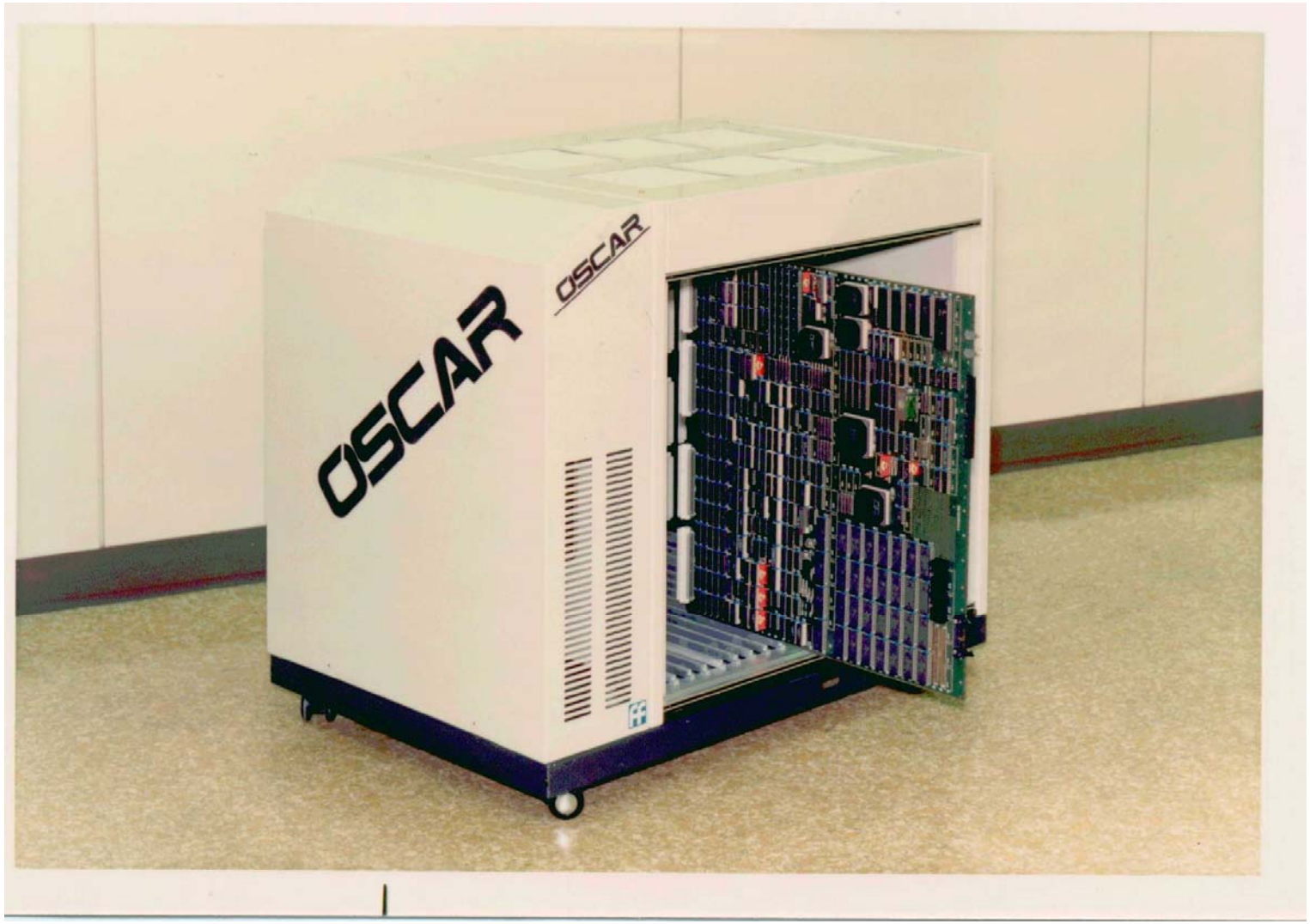
Commercial compiler gives us 0.9 times speedup with 128 cores (slow-downed against 1 core)

- 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

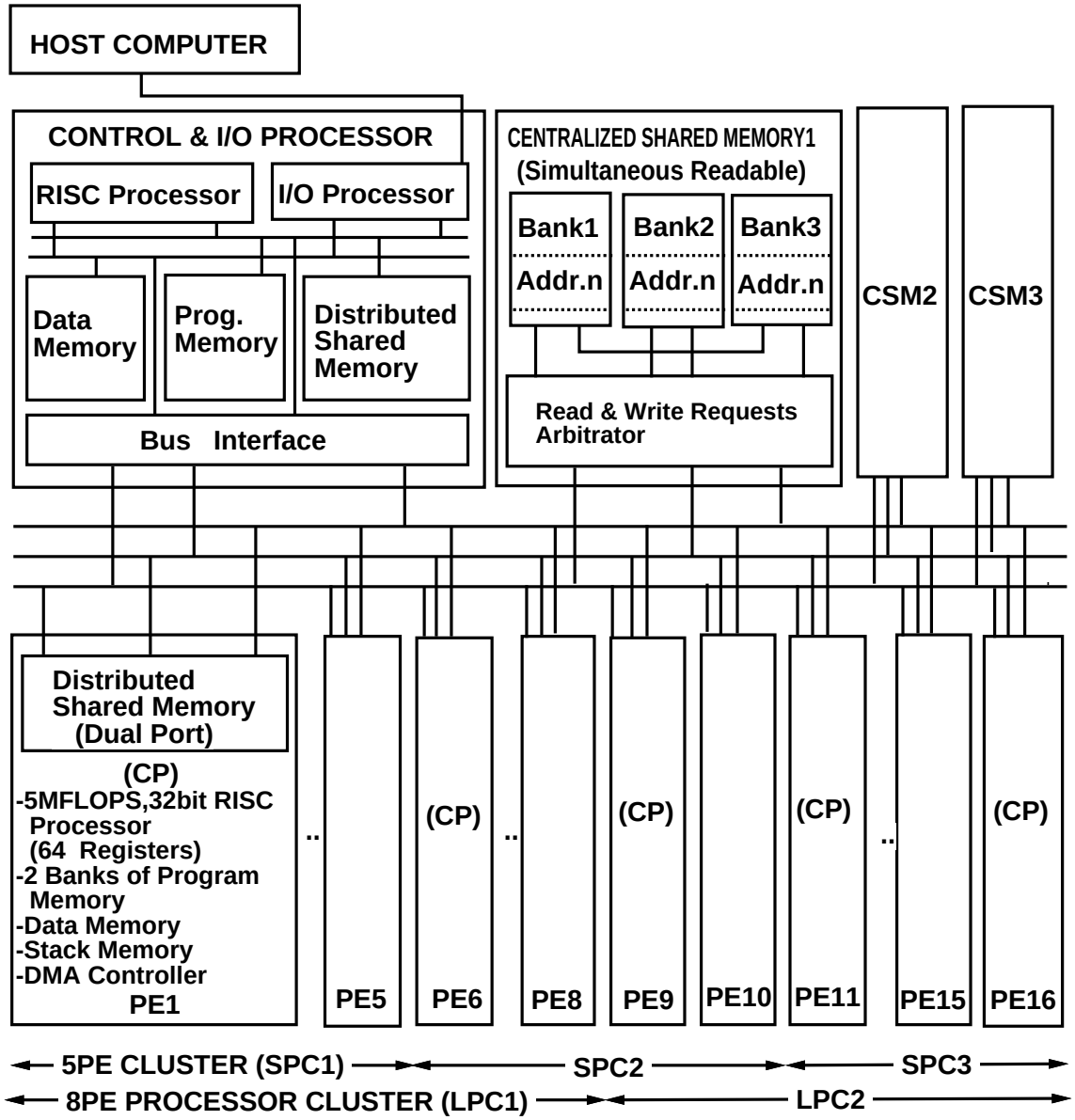
1987 OSCAR(Optimally Scheduled Advanced Multiprocessor)

Co-design of Compiler and Architecture

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

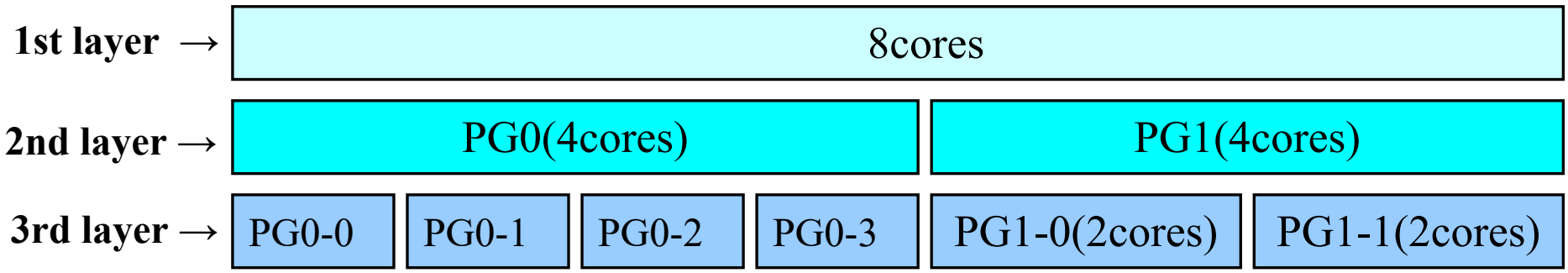


OSCAR(Optimally Scheduled Advanced Multiprocessor)

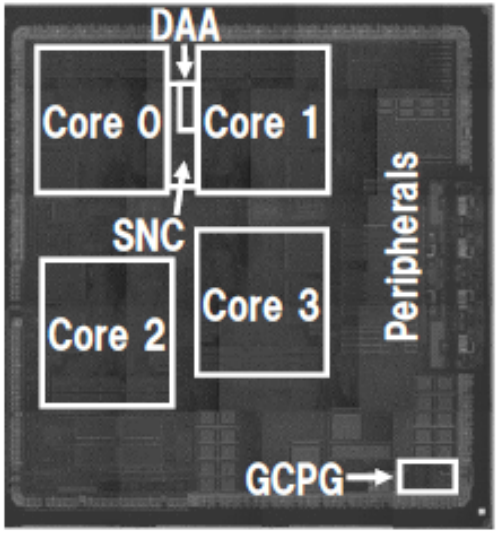
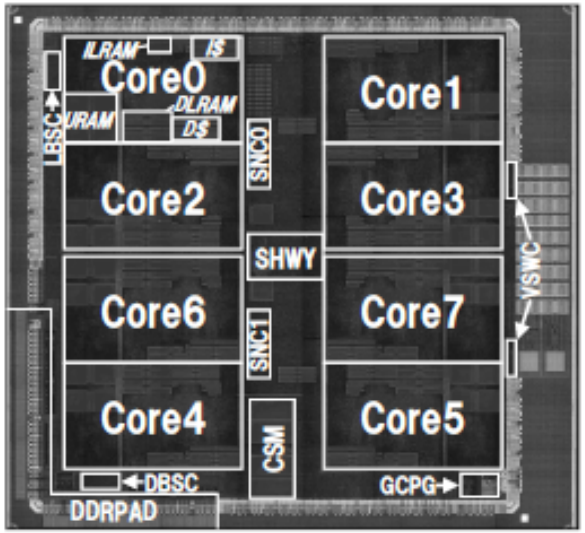
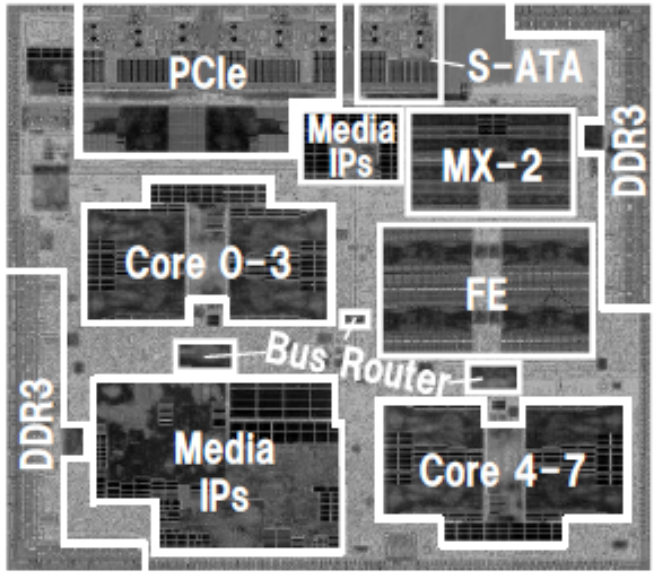


Hierarchical Barrier Synchronization

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



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

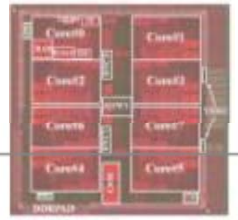
RP-1 (ISSCC2007 #5.3)	***RP-2*** (ISSCC2008 #4.5)	***RP-X*** (ISSCC2010 #5.3)
		
90nm, 8-layer, triple-Vth, CMOS	90nm, 8-layer, triple-Vth, CMOS	45nm, 8-layer, triple-Vth, CMOS
97.6 mm^2 (9.88 x 9.88 mm)	104.8 mm^2 (10.61 x 9.88 mm)	153.8 mm^2 (12.4 x 12.4 mm)
1.0V (internal), 1.8/3.3V (I/O)	1.0-1.4V (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）

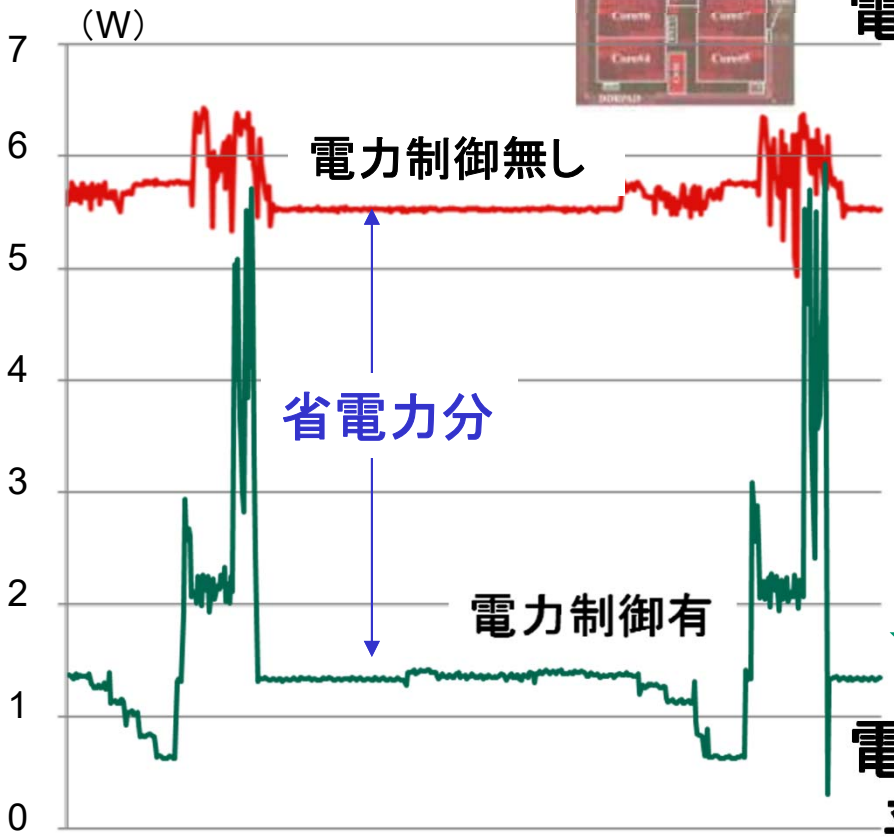
太陽光電力で動作する情報機器

コンピュータの消費電力をHW&SW協調で低減。電源喪失時でも動作することが可能。

リアルタイムMPEG2デコードを、8コアホモジニアス
マルチコアRP2上で、消費電力1/4に削減

世界唯一の差別化技術



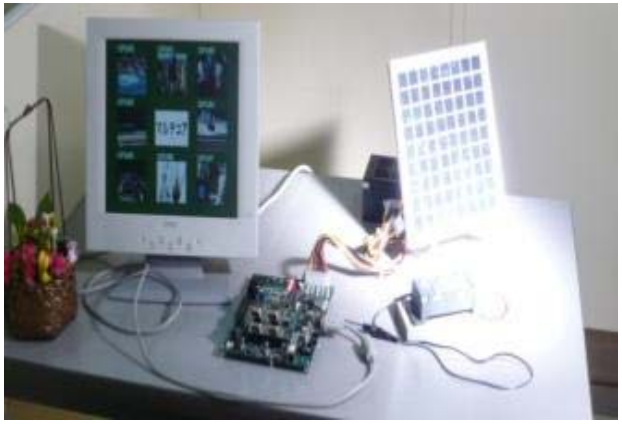


電力制御無し
平均電力
5.73 [W]

電力を
ソフトで
1/4に
削減

電力制御有
平均電力
1.52 [W]

周波数/電圧・電源遮断制御



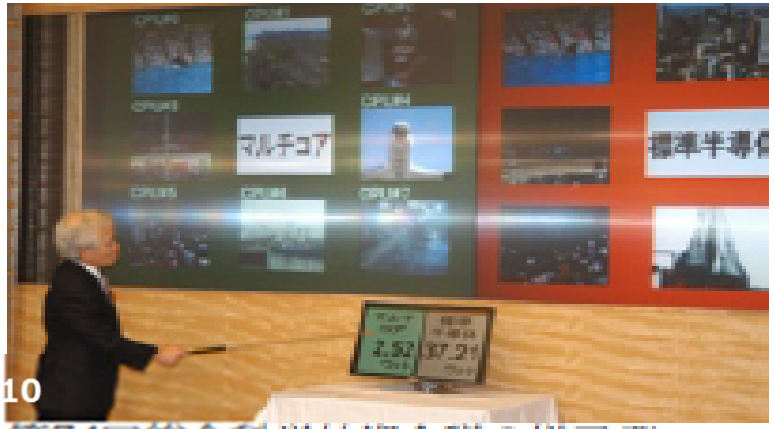
太陽電池で駆動可

NEDOリアルタイム情報家電用マルチコアチップ・デモの様子

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)

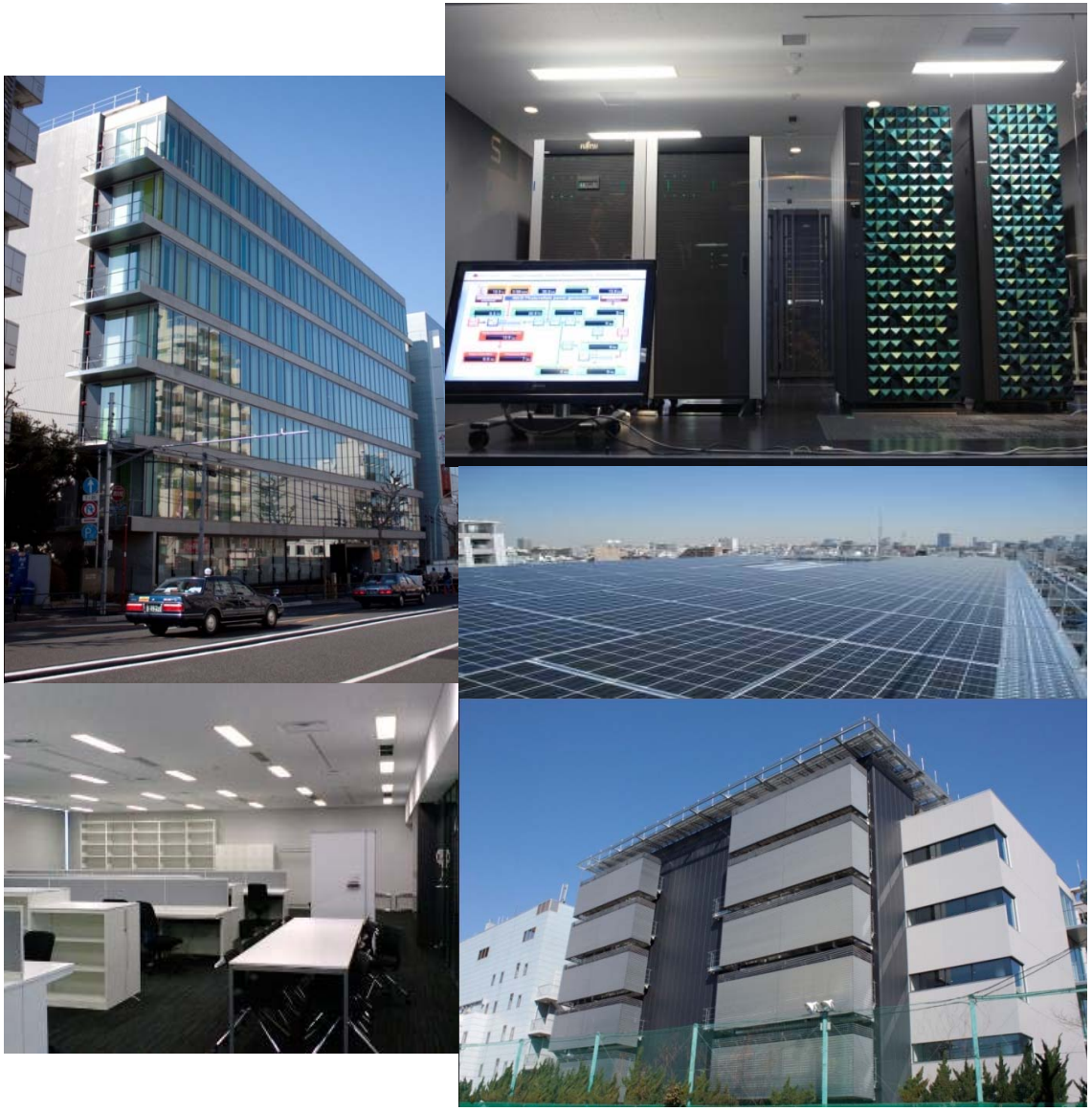
グリーン・コンピューティング・システム研究開発センター　2011年5月13日開所
経済産業省支援：低消費電力マルチコア産官学連携研究　7F笠原・木村, 5F学生

助手: 見神広紀, 島岡 護,大木吉健

客員教授:
内山日立技師長,枝廣名大教授,北村オスカーテクノロジーFellow, 吉田明大教授,
Prof. David Padua (Univ. Illinois),
Prof. Michelle Strout(Arizona Univ.),

客員研究員:
Drs. Shirako & Hayashi (Rice大),日立,NEC,デンソー,オリンパス,ルネサス, オスカー等の企業から約30名
博士課程2名，M2　4名，　M1　4名

＜2017年産学連携＞
日立,デンソー, デンソーヨーロッパ,
ルネサス,NEC,富士電機,
オリンパス，　三菱電機，　NTTデータ,
オスカーテクノロジー（早稲田大学出資ベンチャー）等



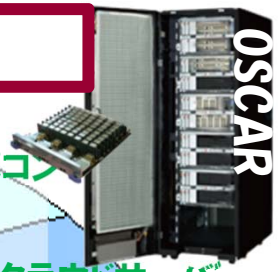
笠原・木村研究室：アドバンストマルチコアプロセッサ研究所

制御

交通シミュレーション・信号制御　NTTデータ・日立

環境を守る

命を守る



グリーンスパコン

車載（エンジン制御・自動運転Deep Learning・ADAS・MATLAB/Simulink自動並列化）デンソー，ルネサス,NEC

アドバンストマルチコアプロセッサ研究所

OSCARマルチコア/メニーコア&コンパイラ　オスカー

ホーム・産業系メニーコア

産業

OS

OSCAR Many-core API

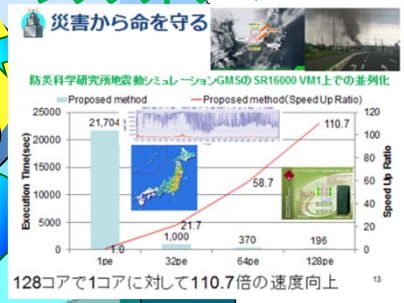
医療

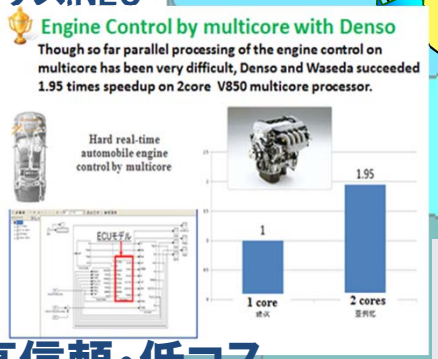
生活

災害

サーバ系メニーコア

グリーンクラウドサーバ



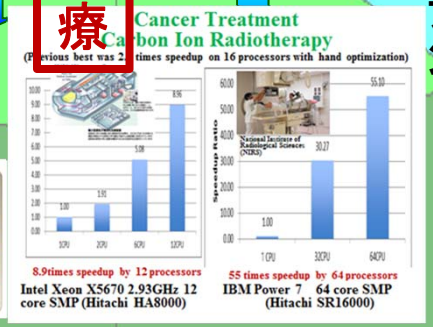


高信頼・低コスト・ソフト開発

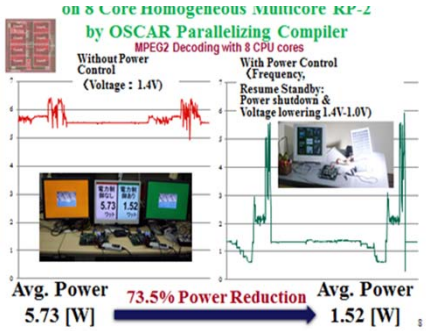
カプセル内視鏡オリンパス

スマートフォ

デスクトップスパコン



首都圏直下型地震火災延焼、住民避難指示



カメラ

FA　三菱

太陽電池駆動・週1以下の充電

医用画像処理等専用サーバ（医療・重粒子線照射計画）日立

新幹線車体設計・ディープラーニング・日立

持続的高付加価値製品の開発

企業

大学

高速化

低消費電力化

産業競争力を守る

世界をリードするマルチコア用コンパイラ技術

プロセッサ高速化における3大技術課題の解消

1.半導体集積度向上（使用可能トランジスタ数増大）に対する速度向上率の鈍化

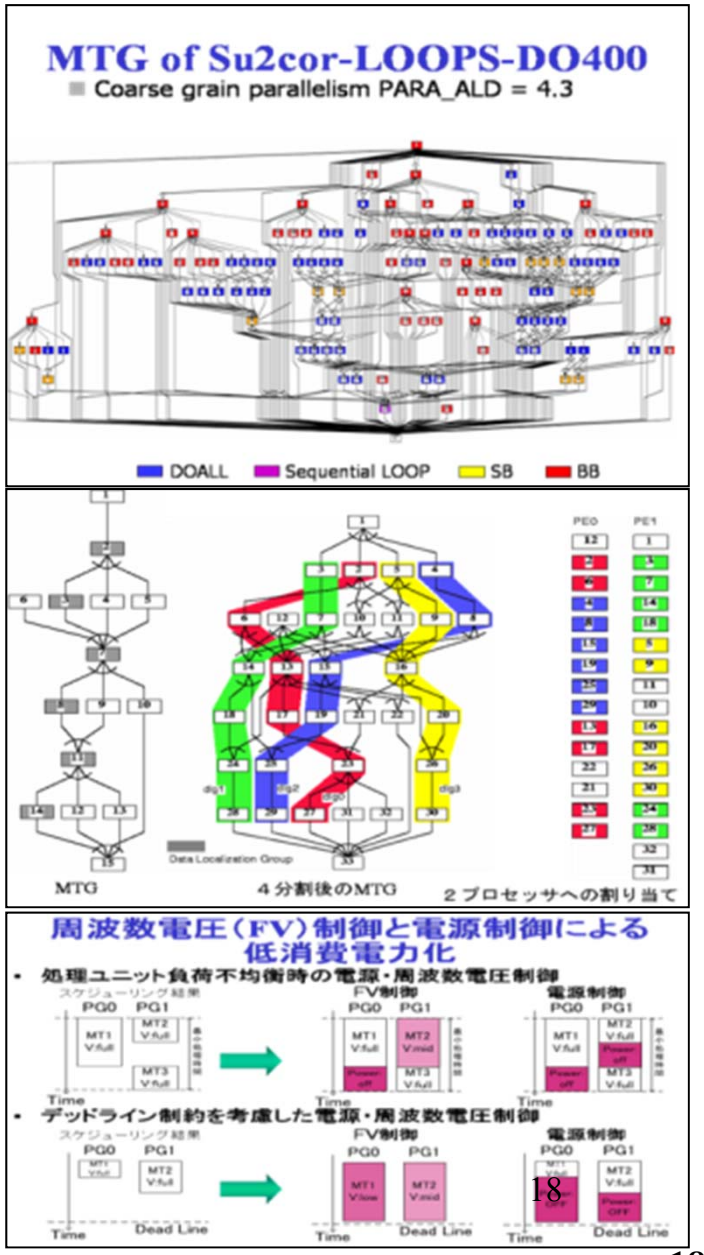
- 粗粒度タスク並列化、ループ並列化、近細粒度並列化によりプログラム全域の並列性を利用するマルチグレイン並列化機能により、従来の命令レベル並列性より大きな並列性を抽出し、複数マルチコアで速度向上

2.メモリウオール問題

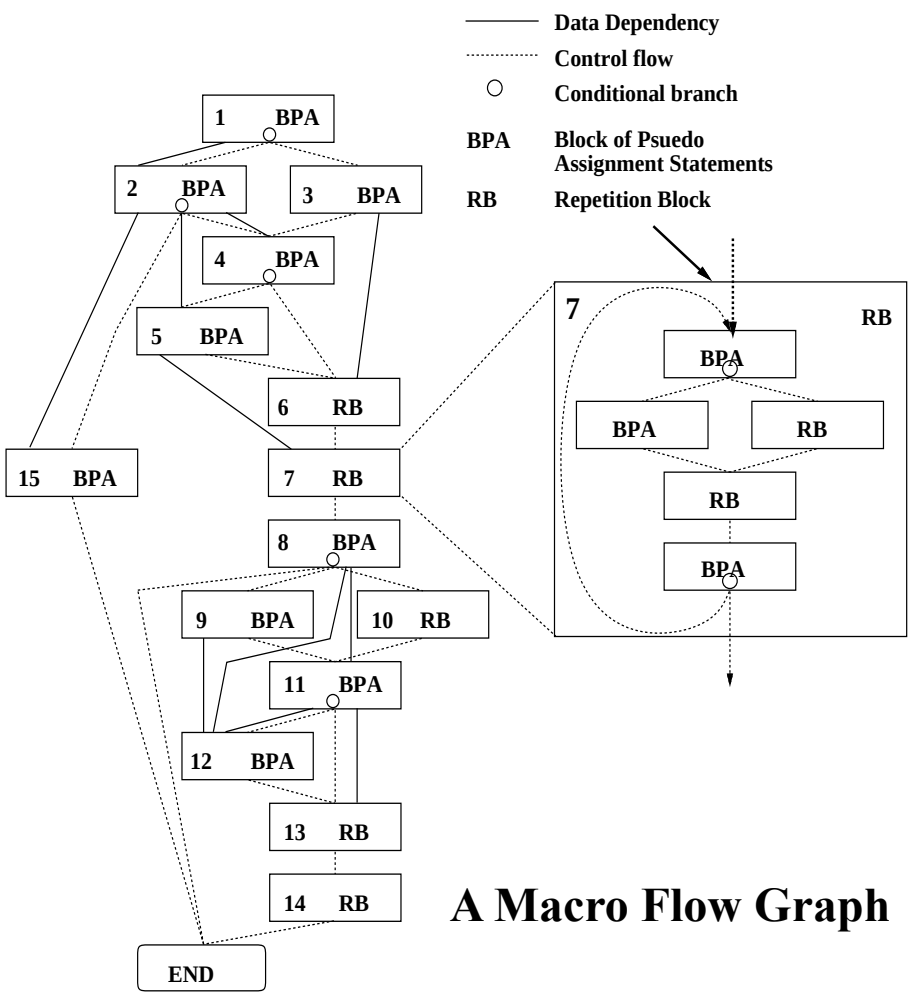
- コンパイラによるローカルメモリへのデータ分割配置、DMAコントローラによるタスク実行とオーバーラップしたデータ転送によりメモリアクセス・データ転送オーバーヘッド最小化

3.消費電力増大による速度向上の鈍化

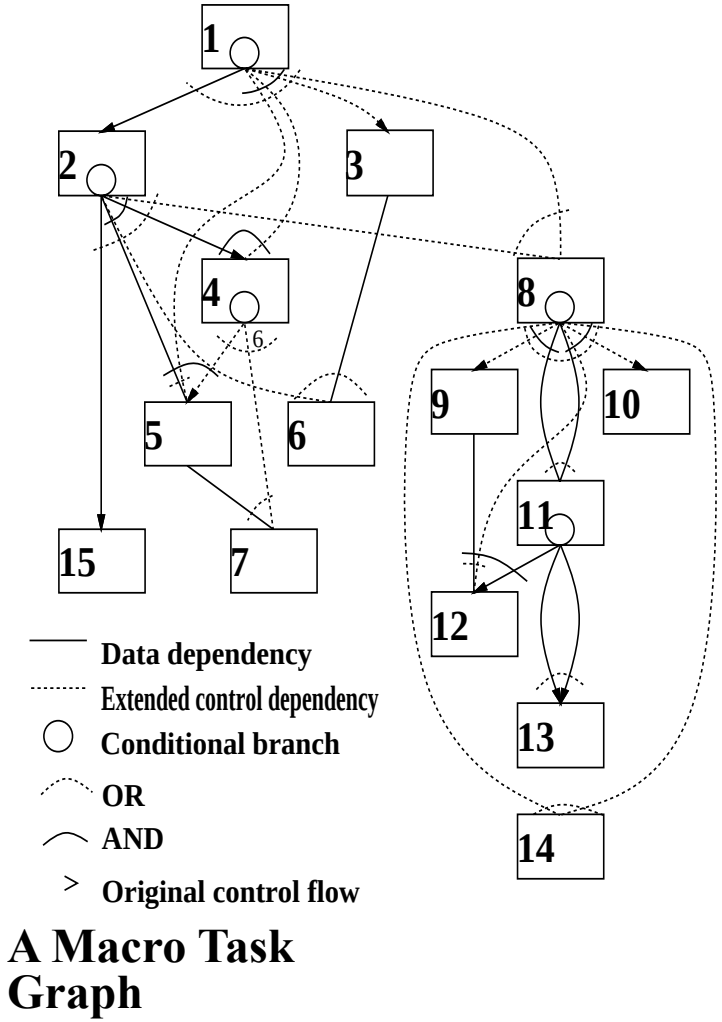
- コンパイラによる低消費電力制御機能を用いたアプリケーション内でのきめ細かい周波数・電圧制御・電源遮断により消費電力低減



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



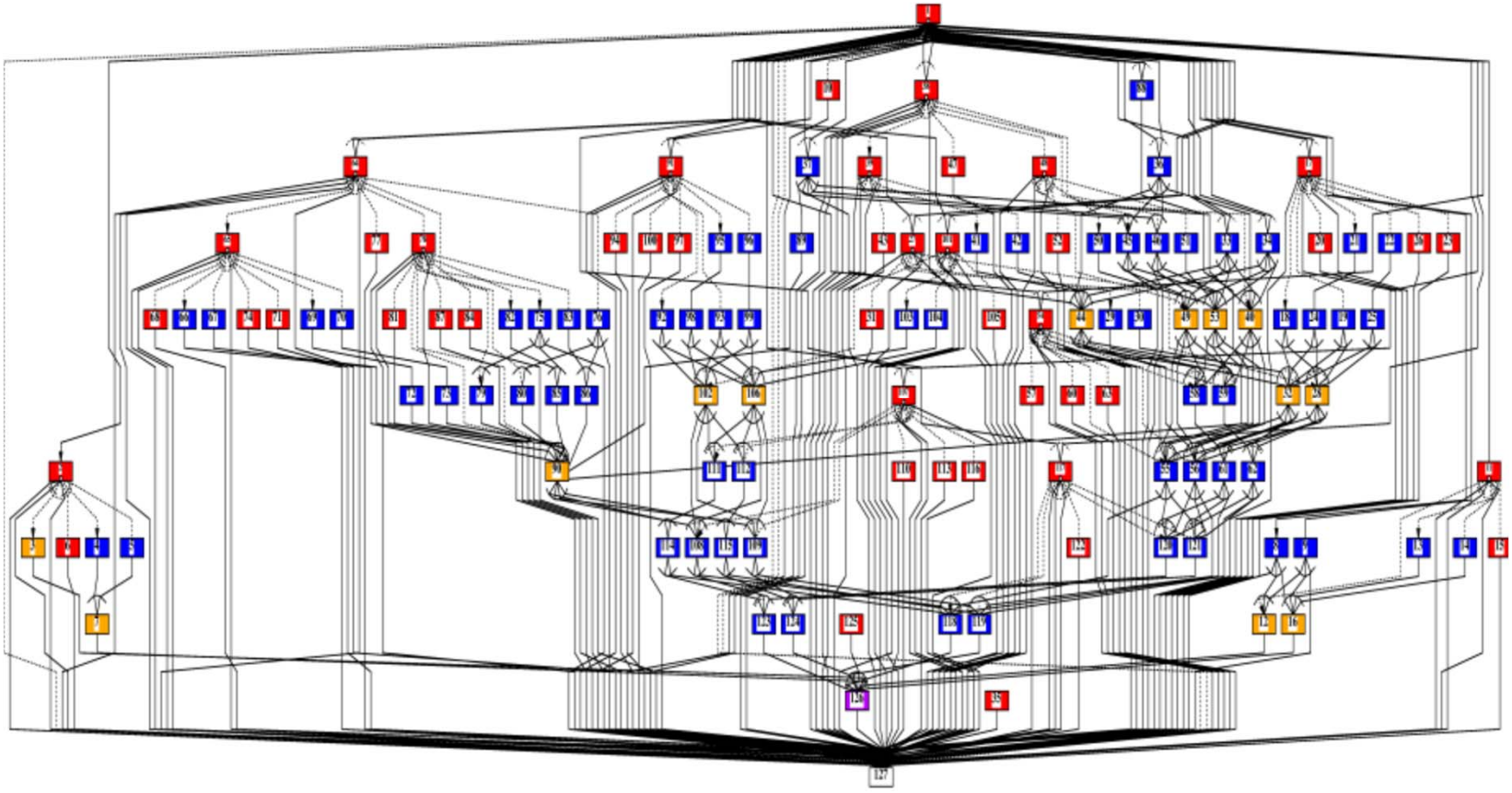
A Macro Flow Graph



A Macro Task Graph

MTG of Su2cor-LOOPS-DO400

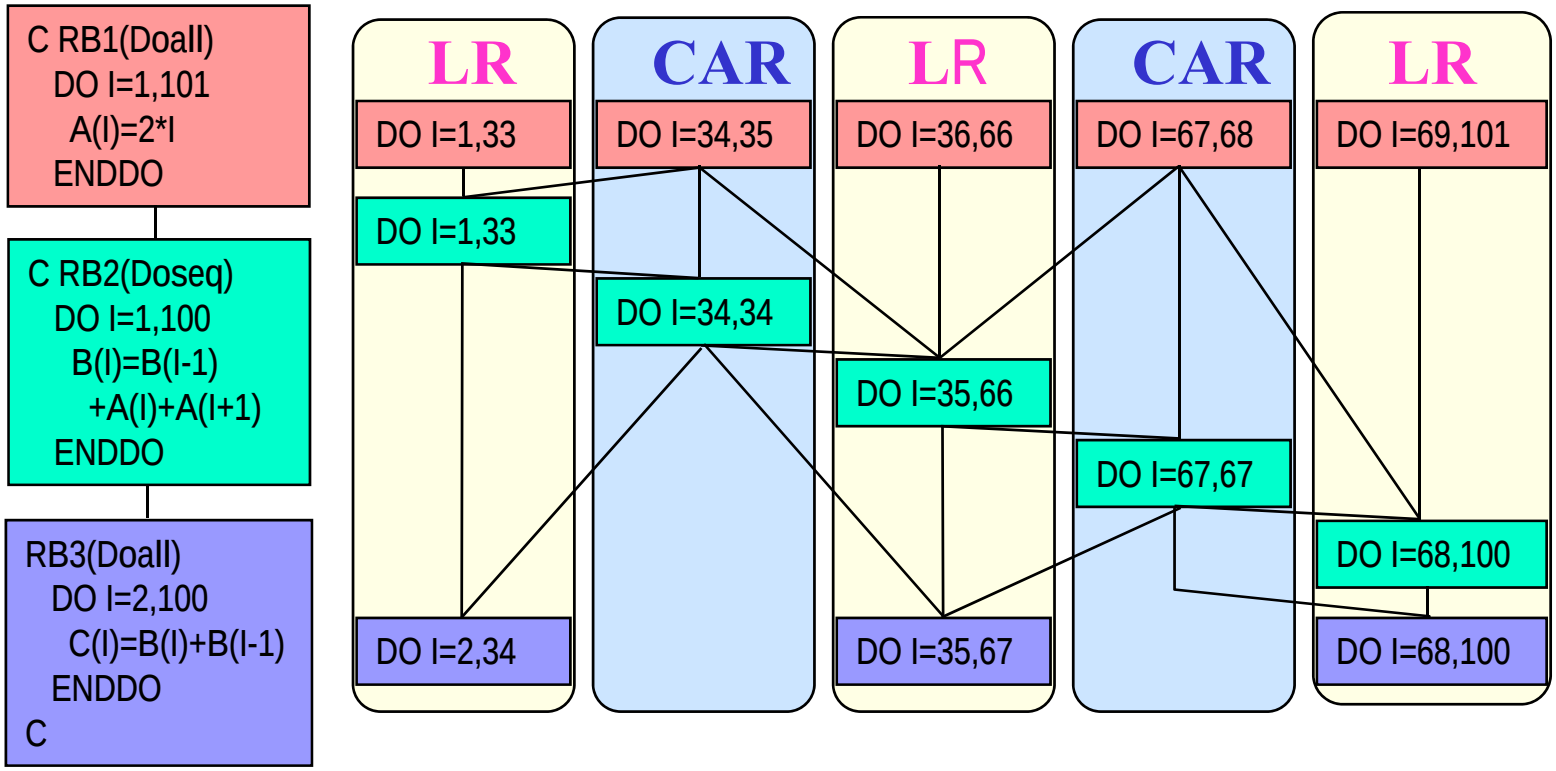
- **Coarse grain parallelism PARA_ALD = 4.3**



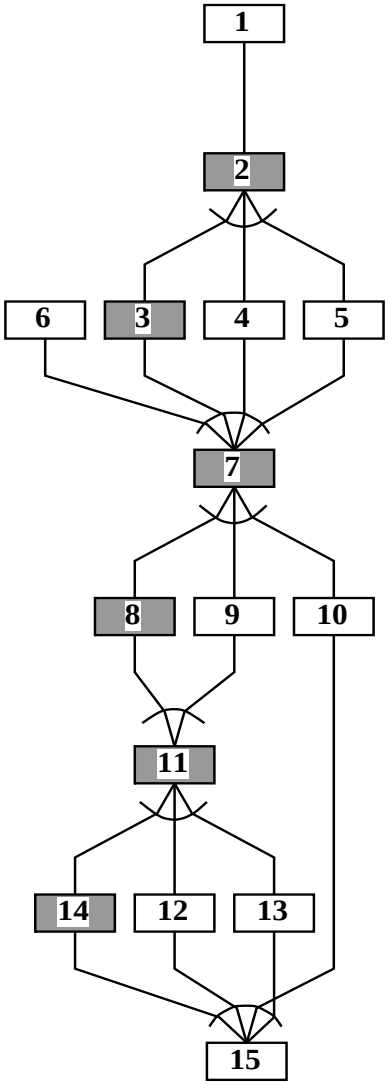
DOALL **Sequential LOOP** **SB** **BB**

Data-Localization: Loop Aligned Decomposition

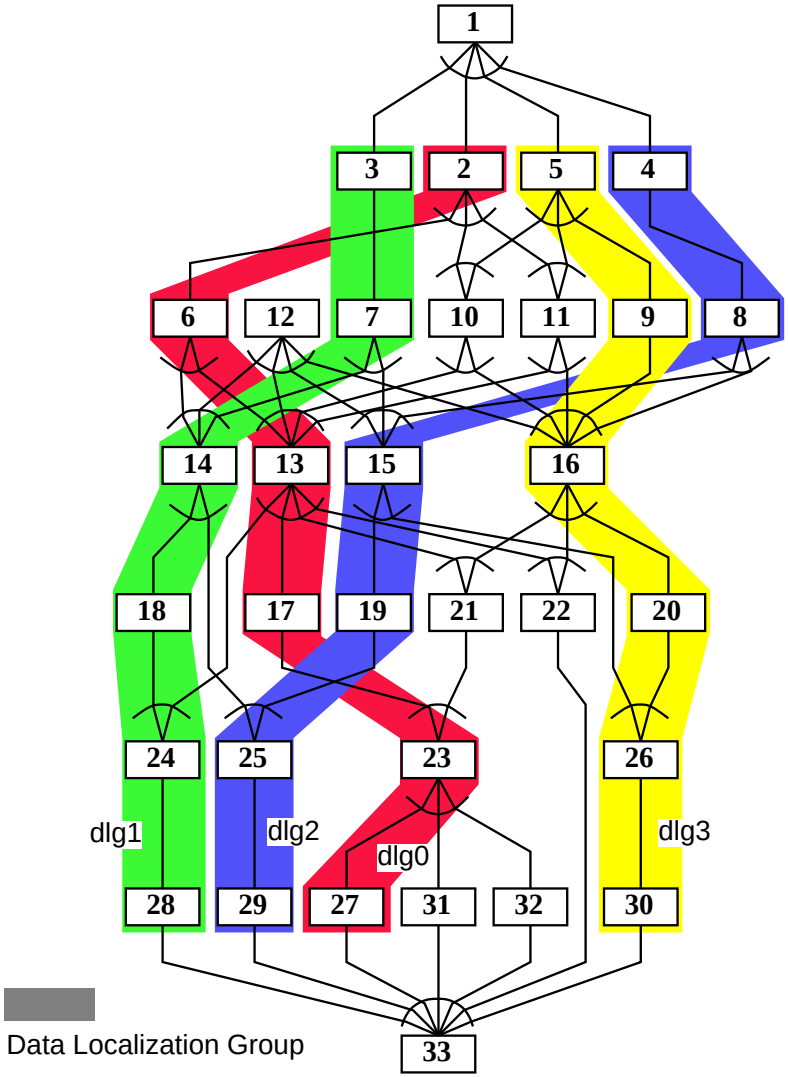
- **Decompose multiple loop (Doall and Seq) into CARs and LRs considering inter-loop data dependence.**
  - **Most data in LR can be passed through LM.**
  - **LR: Localizable Region, CAR: Commonly Accessed Region**



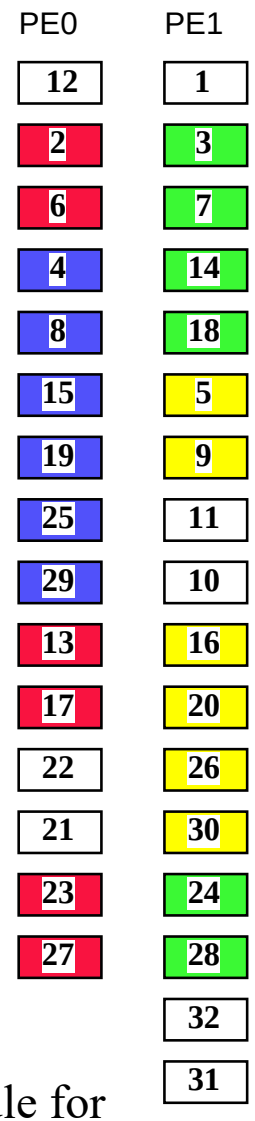
Data Localization



MTG



MTG after Division



A schedule for two processors

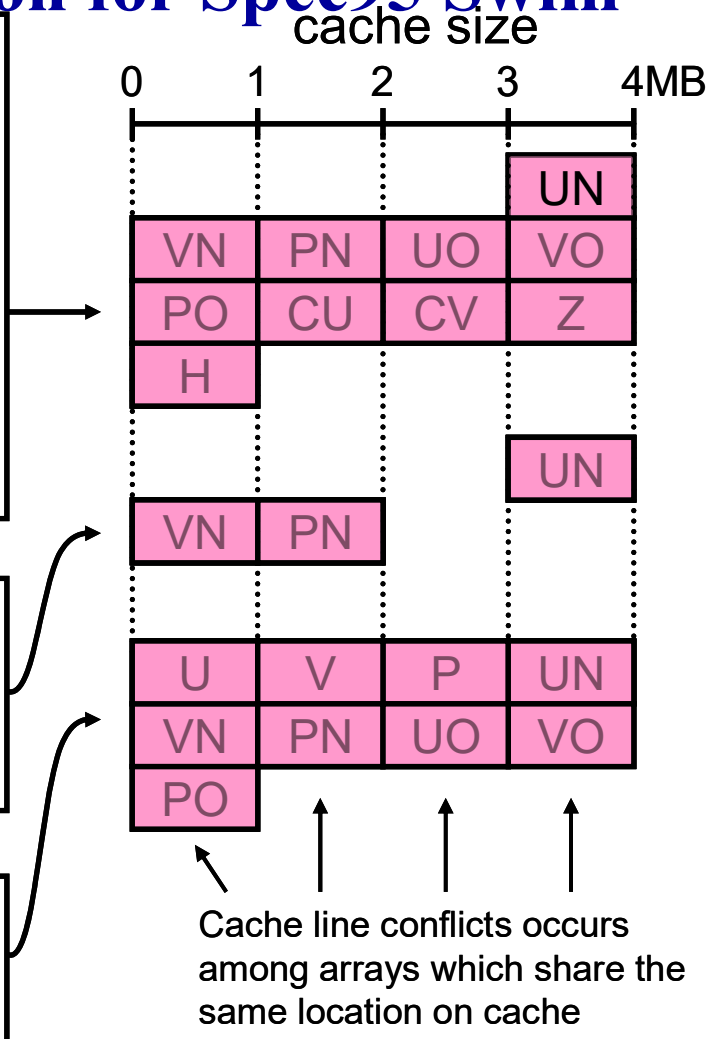
An Example of Data Localization for Spec95 Swim

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

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

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

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



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

Data Layout for Removing Line Conflict Misses by Array Dimension Padding

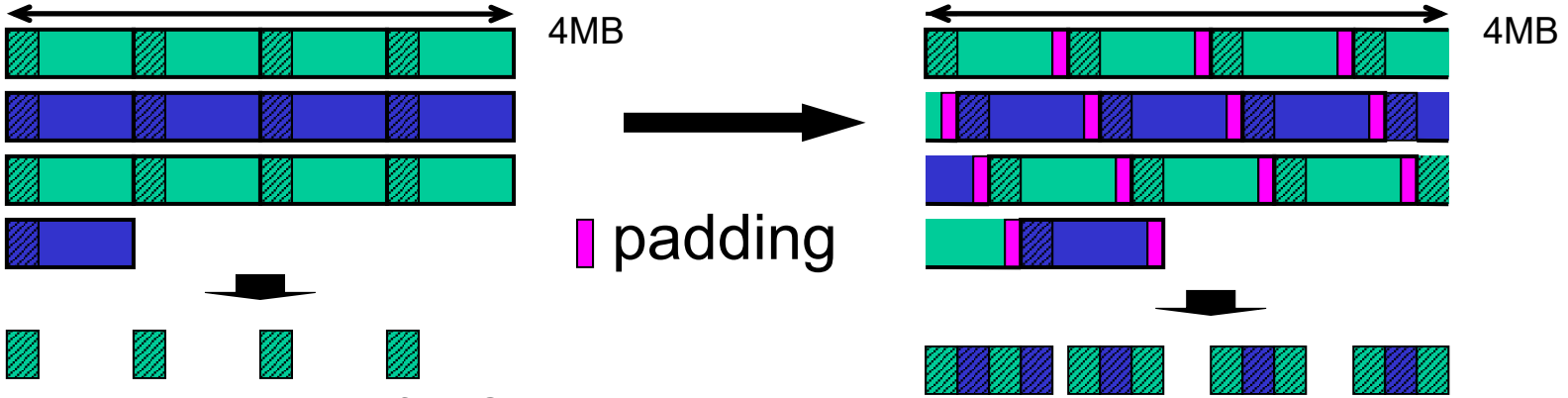
Declaration part of arrays in spec95 swim

before padding

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

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

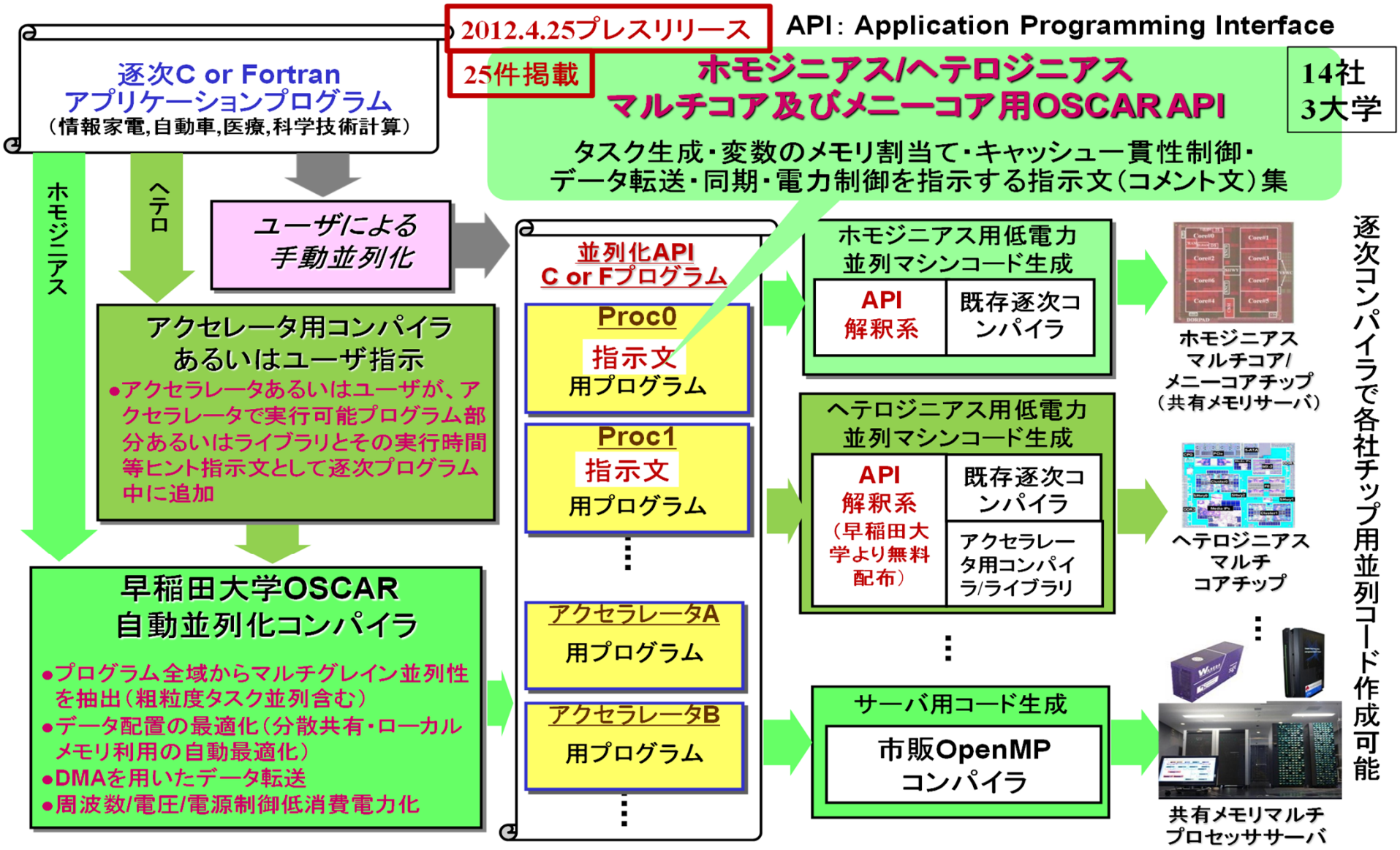
after padding

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

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



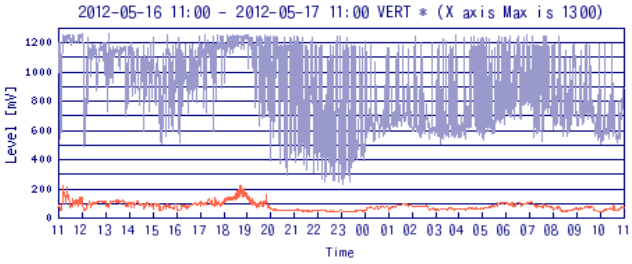
Box: Access range of DLG0

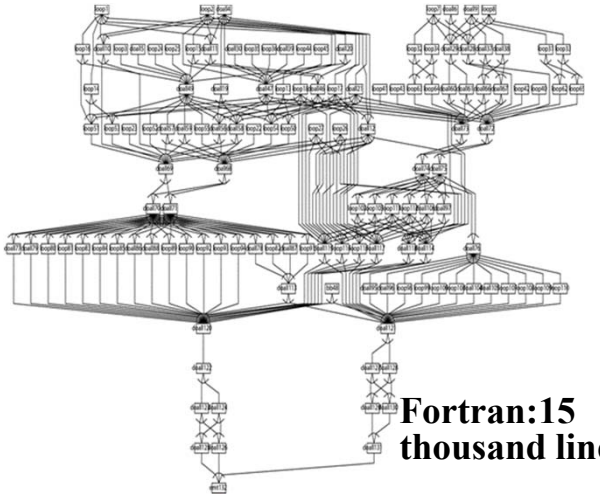
マルチプラットフォームOSCAR API: e.g. Renesas, arm, Infineon, Intel, IBM, AMD



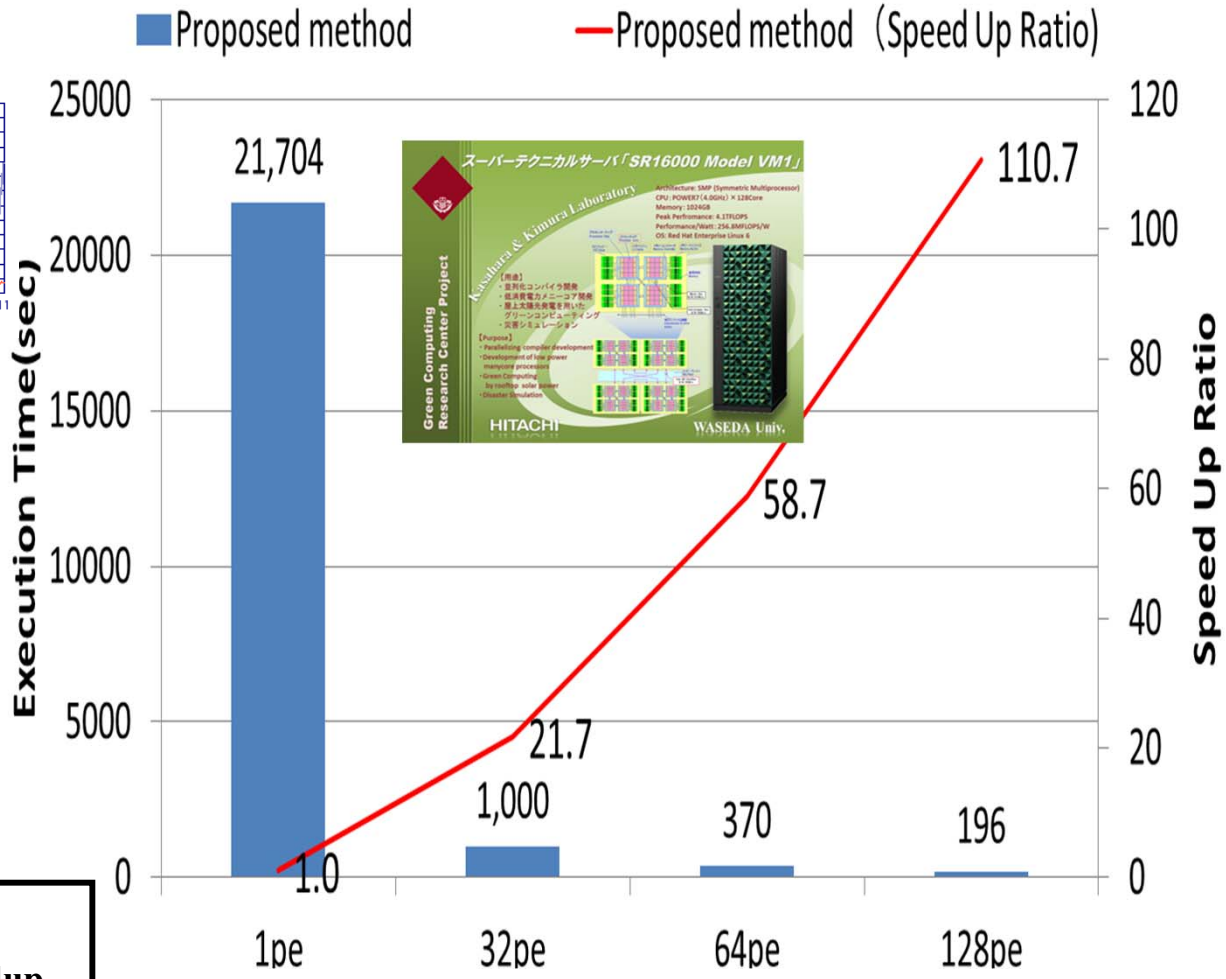
110 Times Speedup against the Sequential Processing for GMS Earthquake Wave Propagation Simulation on Hitachi SR16000

（Power7 Based 128 Core Linux SMP） (LCPC2015)



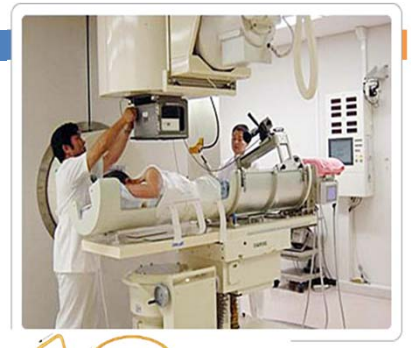


Fortran:15 thousand lines



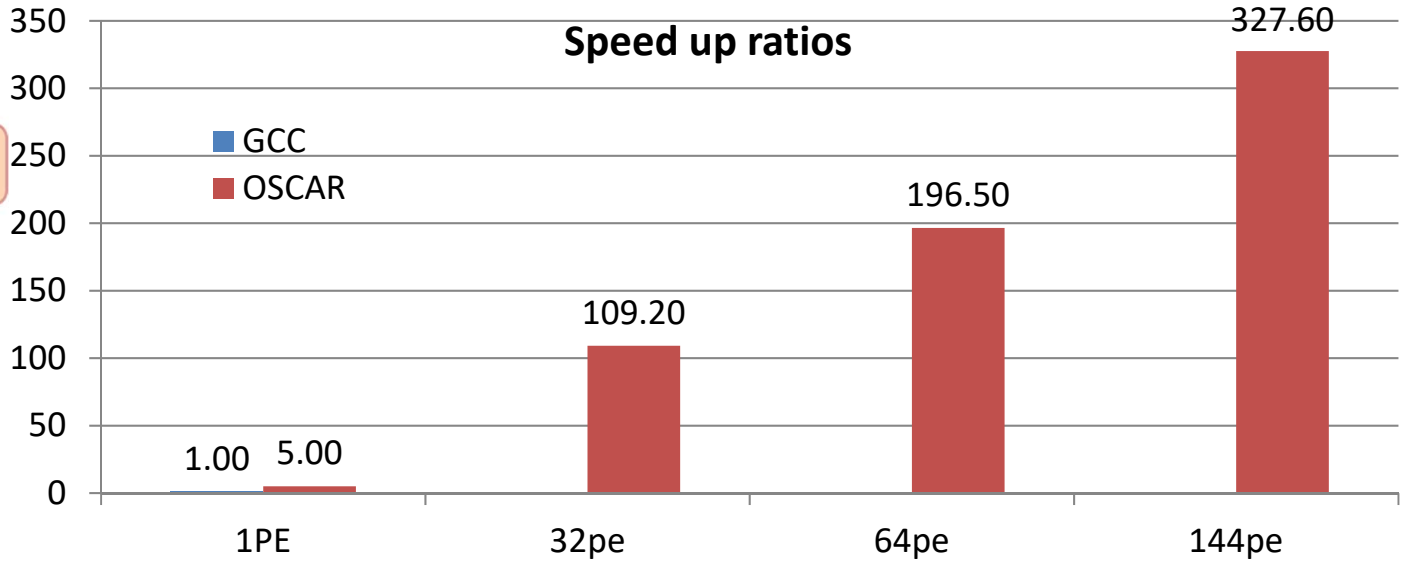
First touch for distributed shared memory and cache optimization over loops are important for scalable speedup

重粒子線がん治療計算の日立BS500ブレードサーバ上での並列化



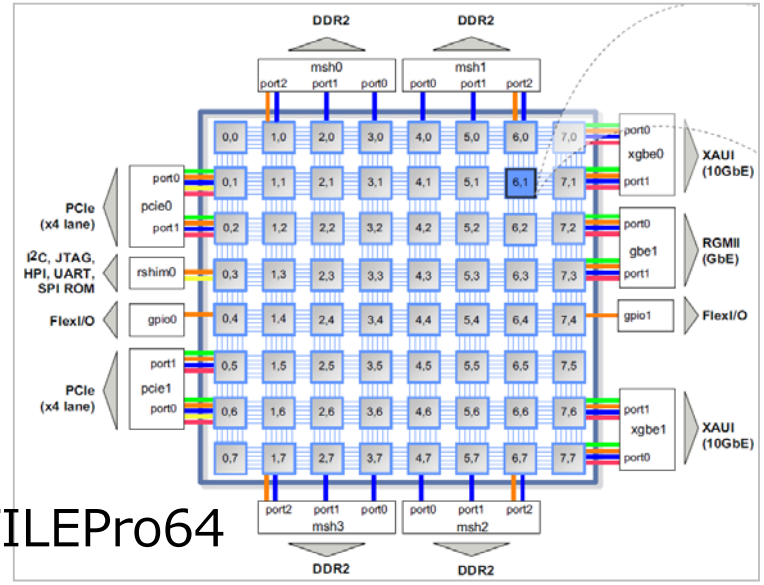
放射線医学総合研究所サイトより
http://www.nirs.qst.go.jp/rd/cpt/index.html

日立 SMPブレードサーバ BS500:
Xeon E7-8890 V3(2.5GHz　18core/chip) x8 chip 計144cores

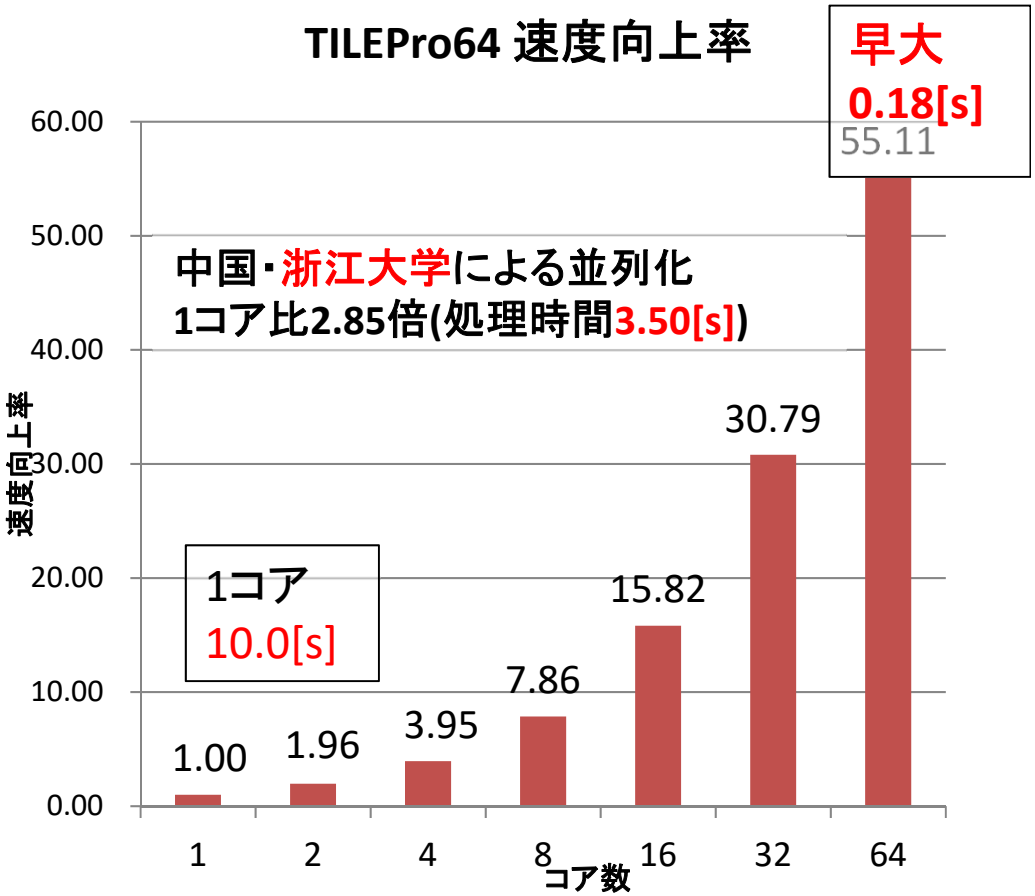


- オリジナル逐次実行時間2948秒（約50分）が、OSCARコンパイラによる144コア並列処理で、9秒に短縮され、327.6倍の速度向上

次世代カメラ・カプセル内視鏡のための静止画圧縮JPEG XRエンコーダ技術の高速低電力処理実証

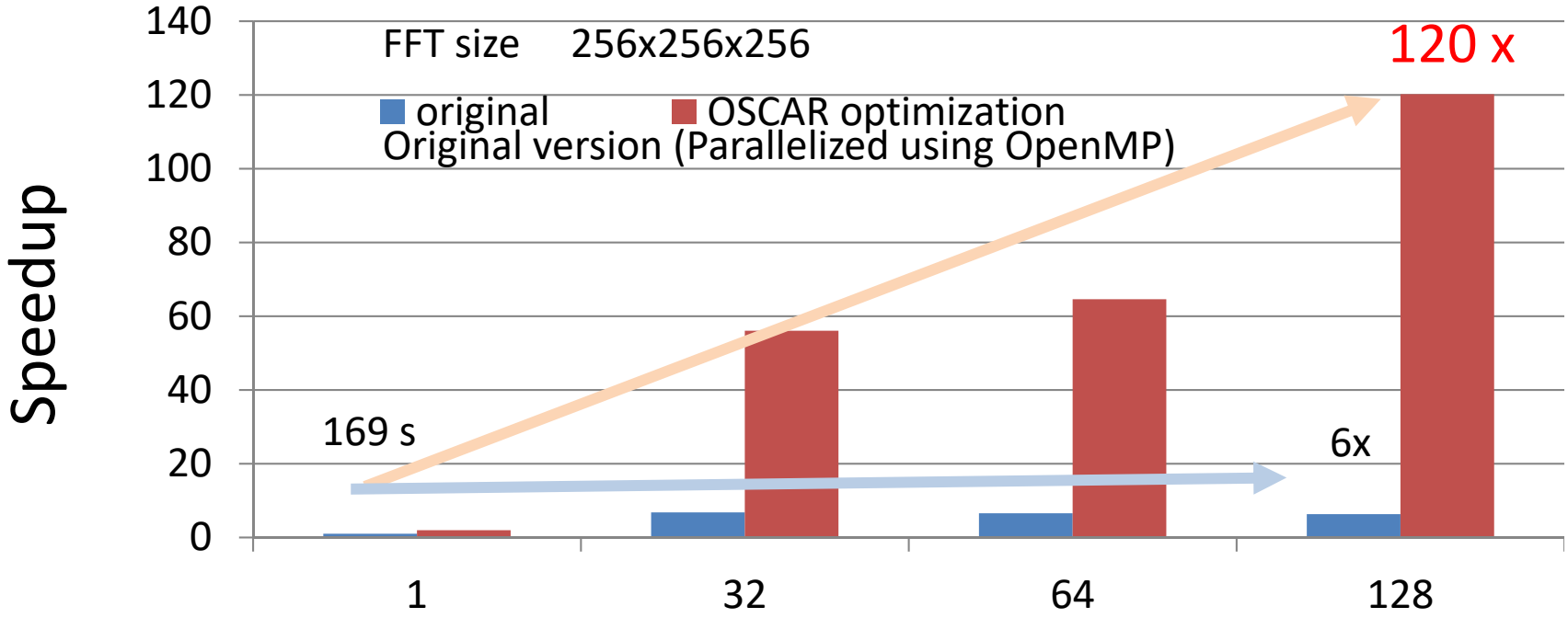


TILEPro64



逐次に対し64コアで55倍の速度向上を達成

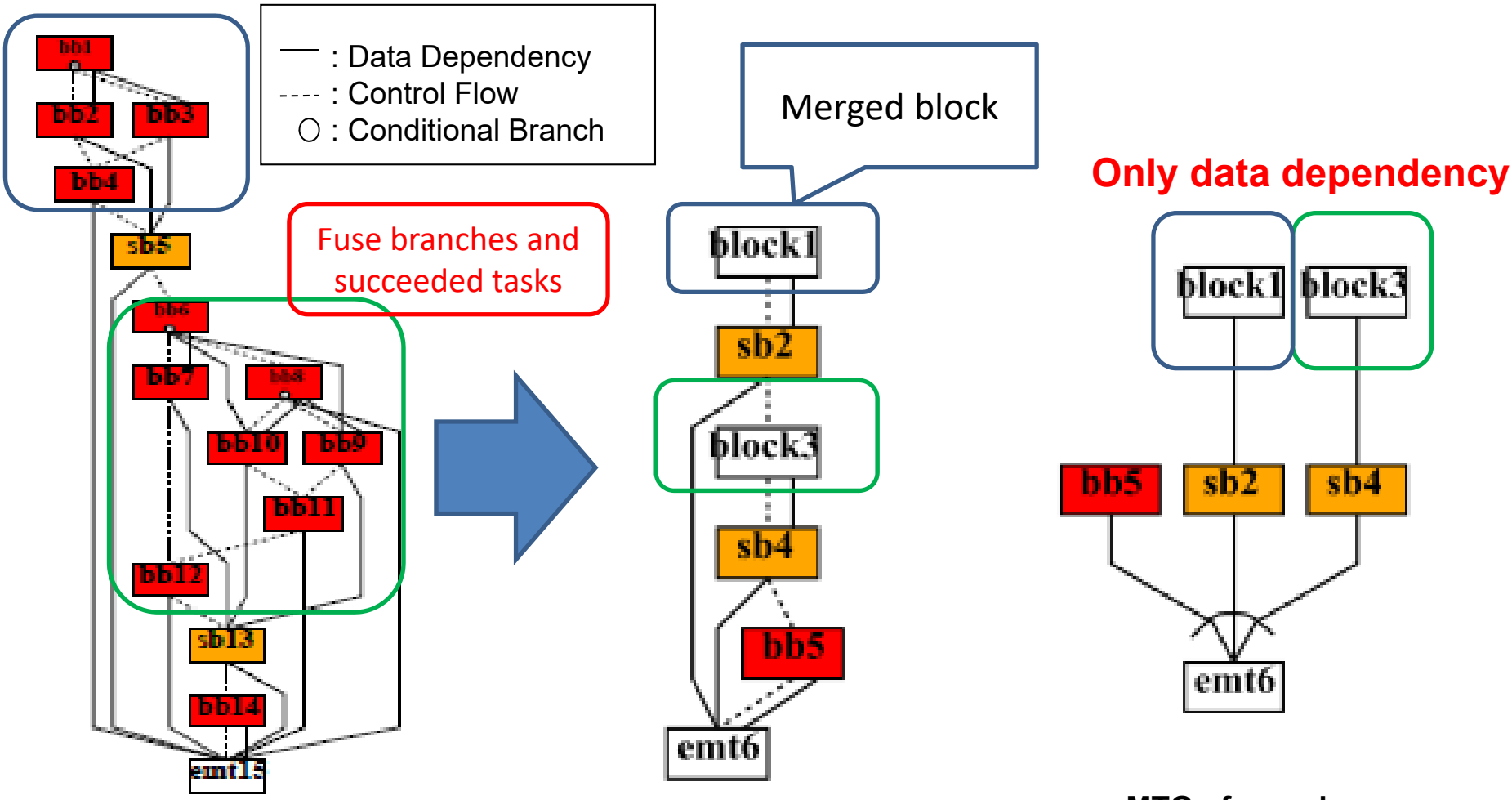
Parallelization of 3D-FFT for New Magnetic Material Computation on Hitachi SR16000 Power7 CC-Numa Server



OSCAR optimization

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

Parallel Processing of Automotive Engine Control: ECU Macro Task Fusion for Static Task Scheduling



MFG of sample program before maro task fusion

MFG of sample program after macro task fusion

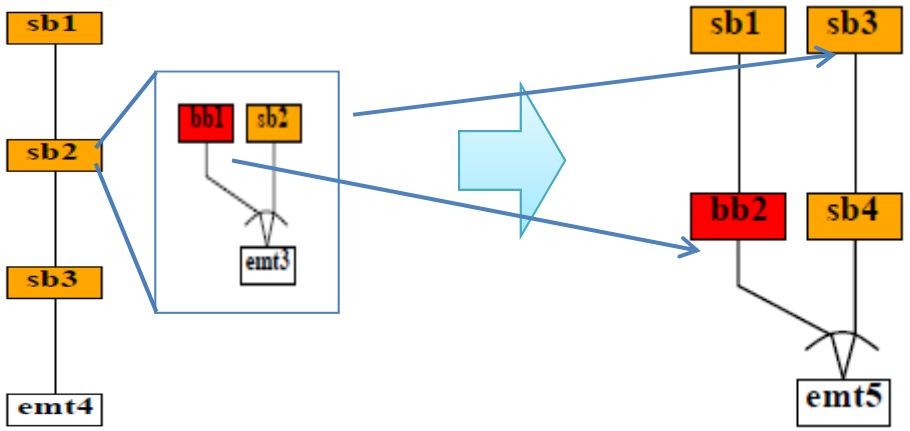
MTG of sample program after macro task fusion

ECU

3.1 Restructuring : Inline Expansion

- Inline expansion is effective
  - To increase coarse grain parallelism
- Expands functions having inner parallelism

Improves coarse grain parallelism



MTG before inline expansion

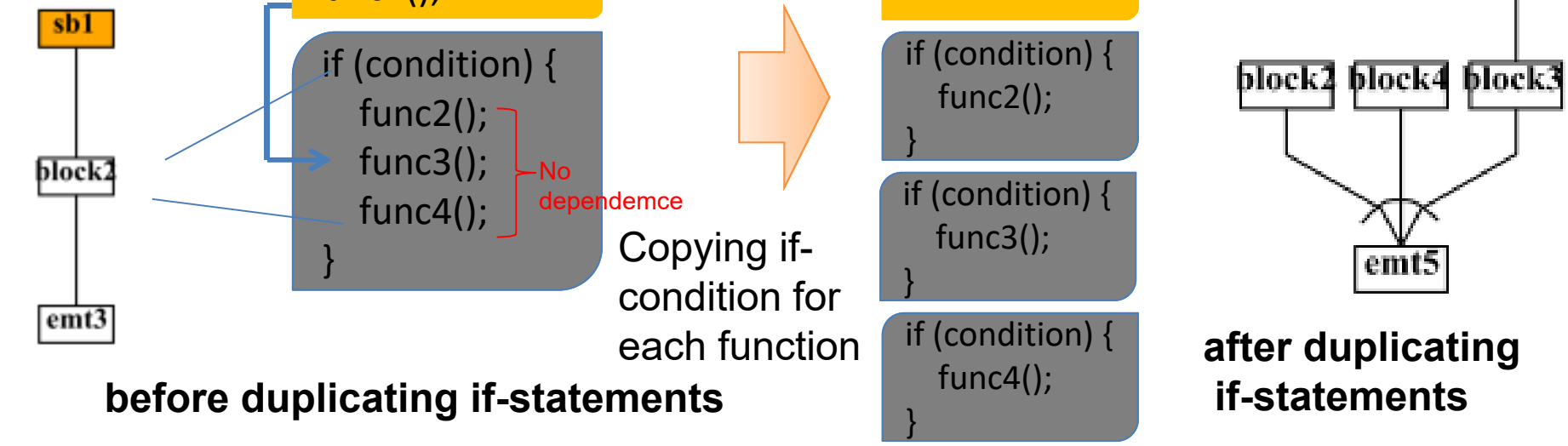
MTG after inline expansion

ECU
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



before duplicating if-statements

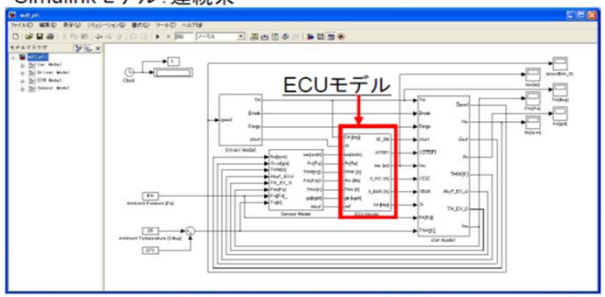
after duplicating if-statements

国際産業競争力を高める

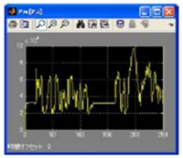
自動走行車（衝突防止含む）、次世代低燃費エンジン制御

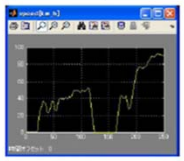
マルチコアによるエンジン制御

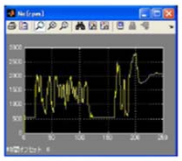
Simulinkモデル：連続系

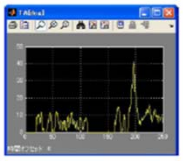


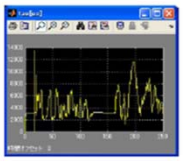
エンジンスペック
-直列4気筒：2000cc
-4AT
-実エンジンデータ





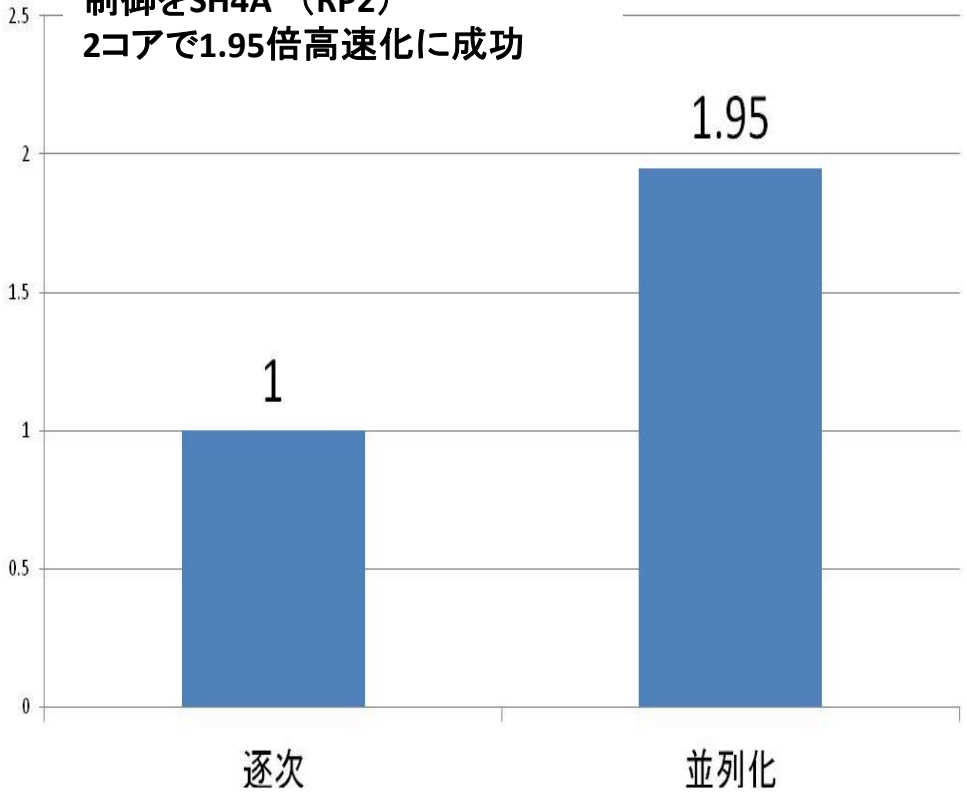






デンソーと共同研究

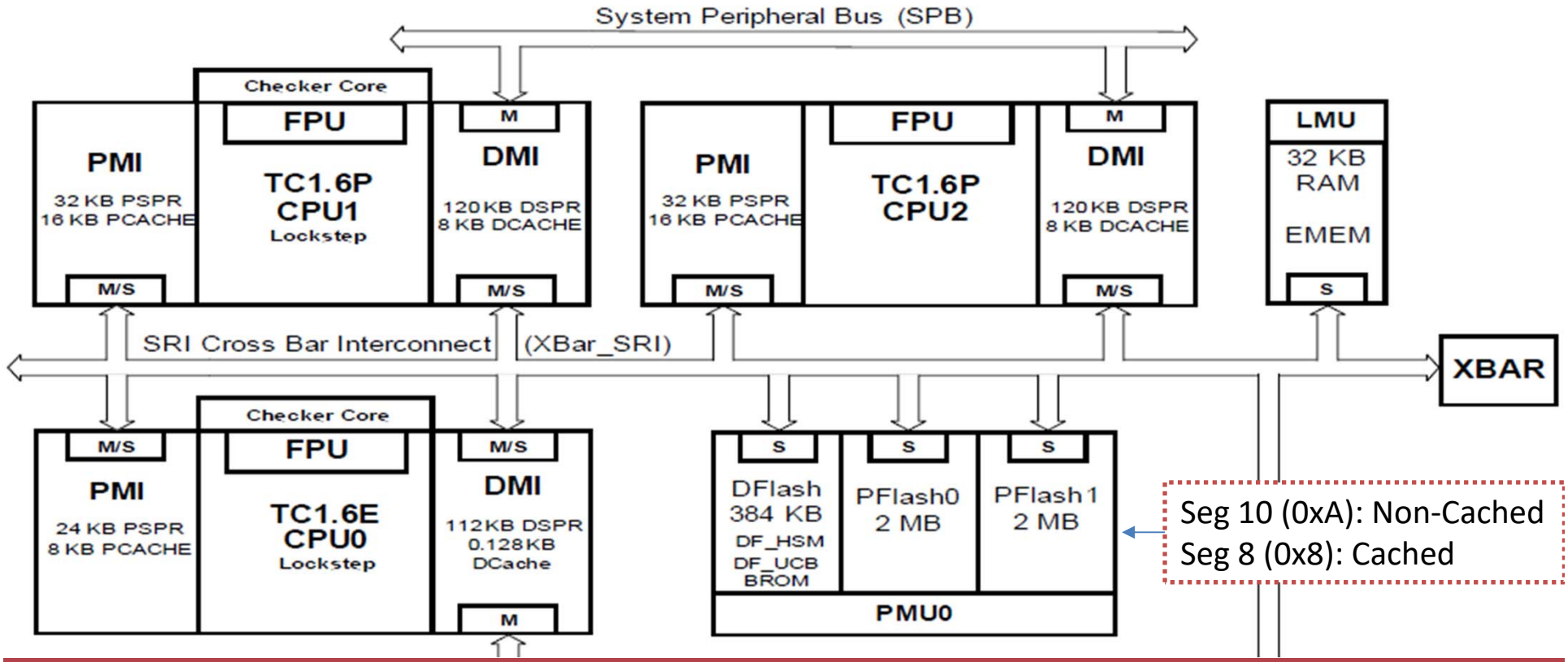
従来並列化できなかったエンジン
制御をSH4A （RP2）
2コアで1.95倍高速化に成功



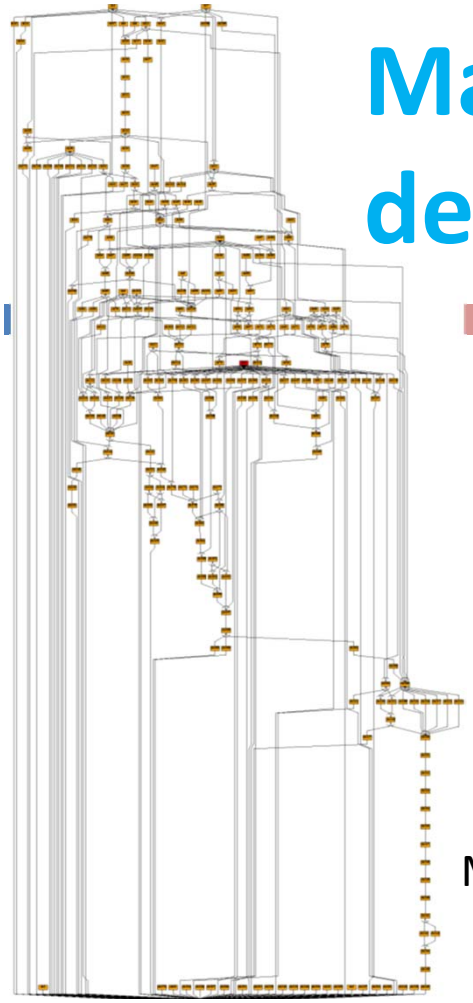
Infineon AURIX TC277

Abbreviations :

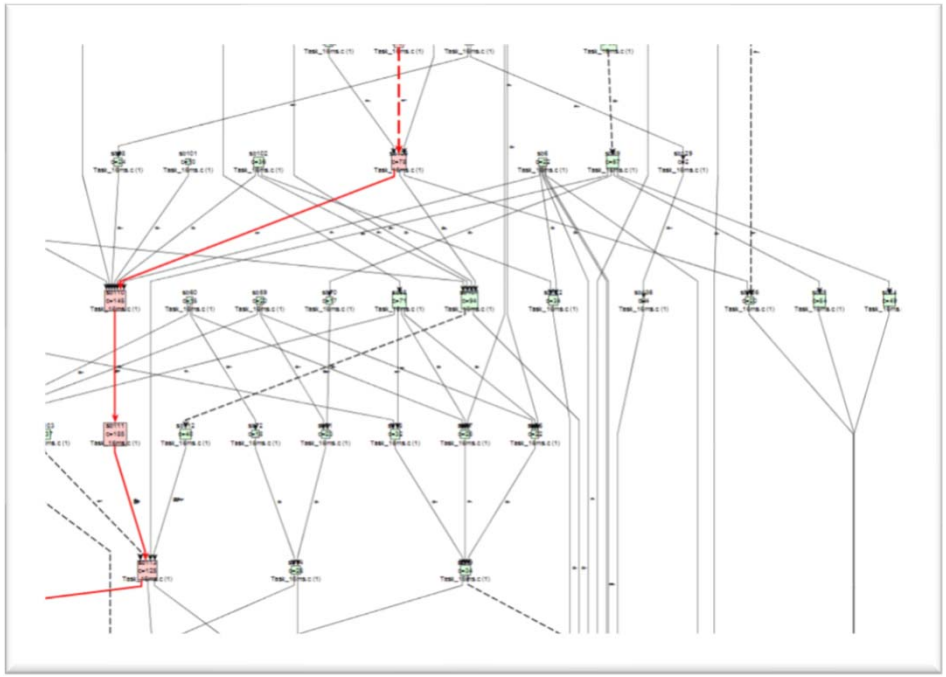
PCACHE:	Program Cache
DCACHE :	Data Cache
DSPR:	Data Scratch-Pad RAM
PSPR:	Program Scratch-Pad RAM
BROM:	Boot ROM
PFlash:	Program Flash
DFlash:	Data Flash (EEEPROM)
S :	SRI Slave Interface
M :	SRI Master Interface

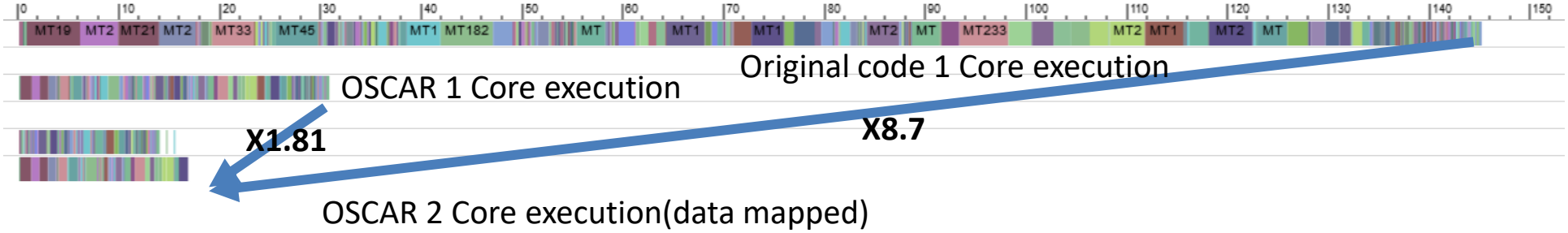


Macrotask Graph, Dependence details and schedules



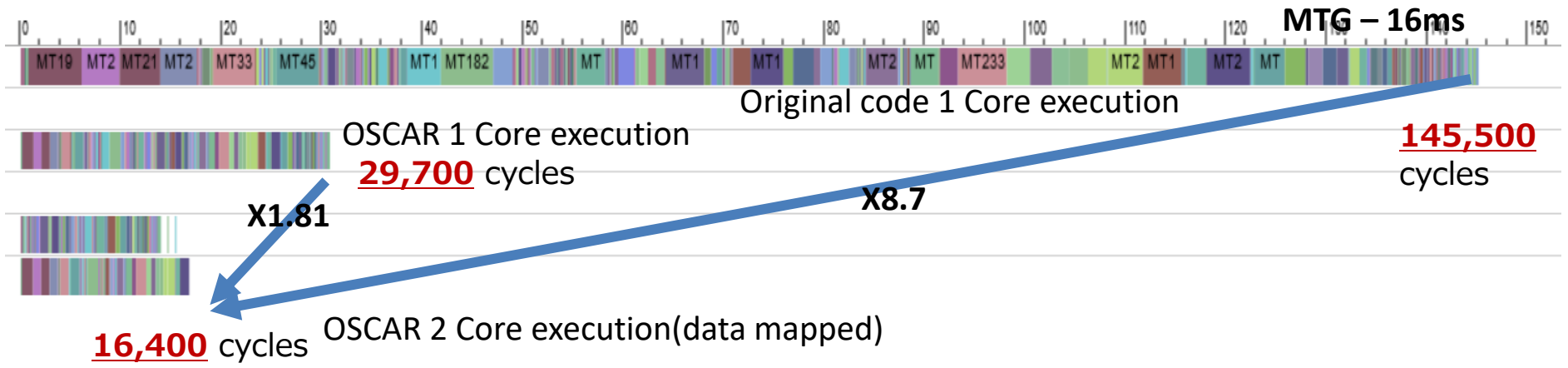
MTG – task_16ms





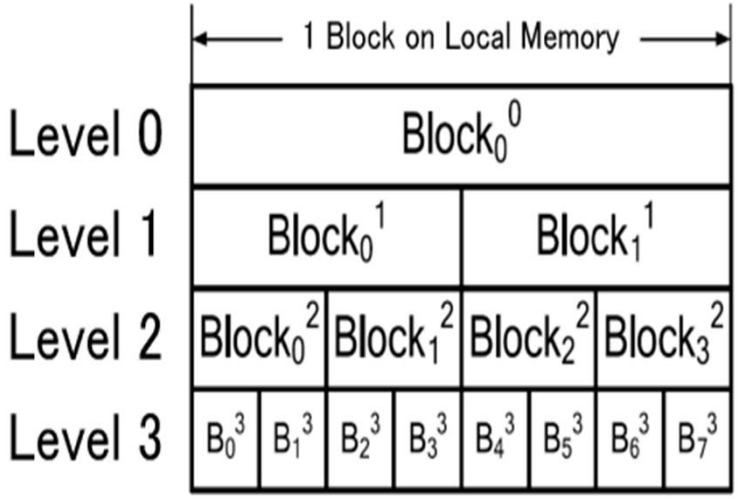
Automatic Parallelization of an Engine Control C Program with 400 thousands lines on AUTOSAR on 2 cores of Infineon AURIX TC277

- **Original sequential** execution time on 1 core: **145,500** cycles
- **Sequential execution time by OSCAR** on 1 core: **29,700** cycles
  - 4.9 times speedup on 1 core against original execution by OSCAR Compilers automatic data allocation for local scratch pad memory, flush memory modules
- **2 core execution by OSCAR** Compiler: **16,400** cycles
  - **1.81 times speedup with 2 core against 1 core execution with OSCAR Compiler**
  - **8.7 times speedup against original sequential execution.**



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 $\text{Block}_{\text{Number}}^{\text{Level}}$ small arrays and scalar

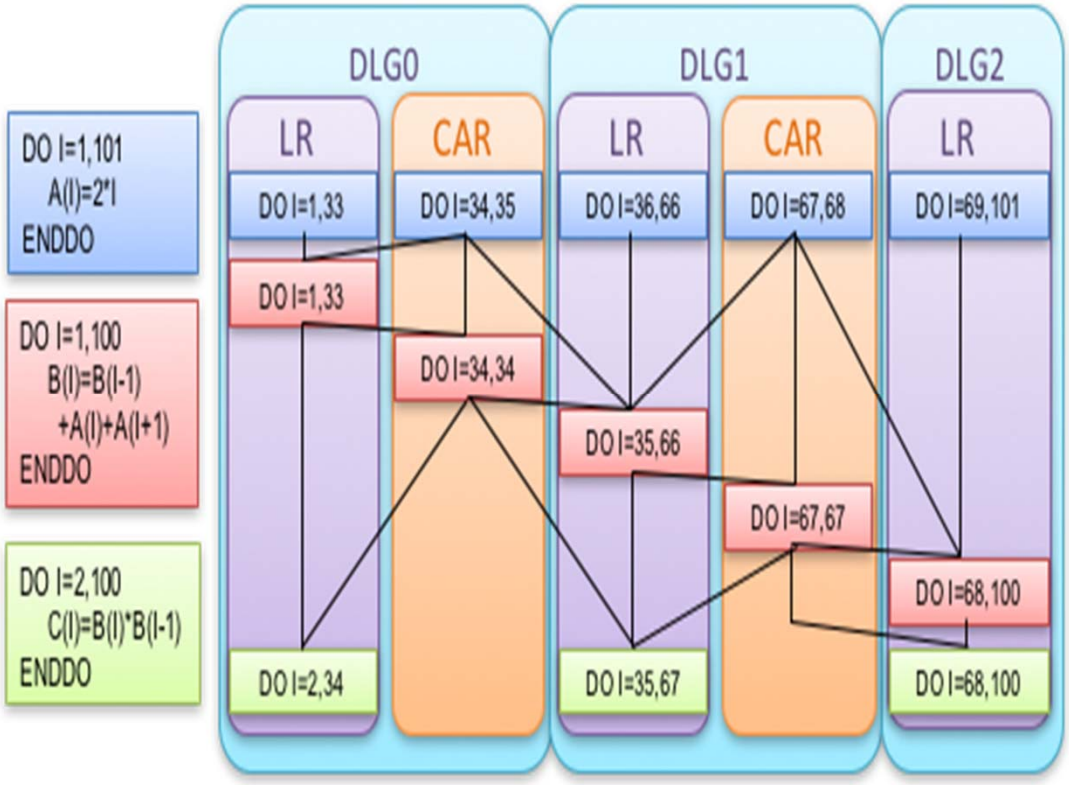


Automatic Local Memory Management

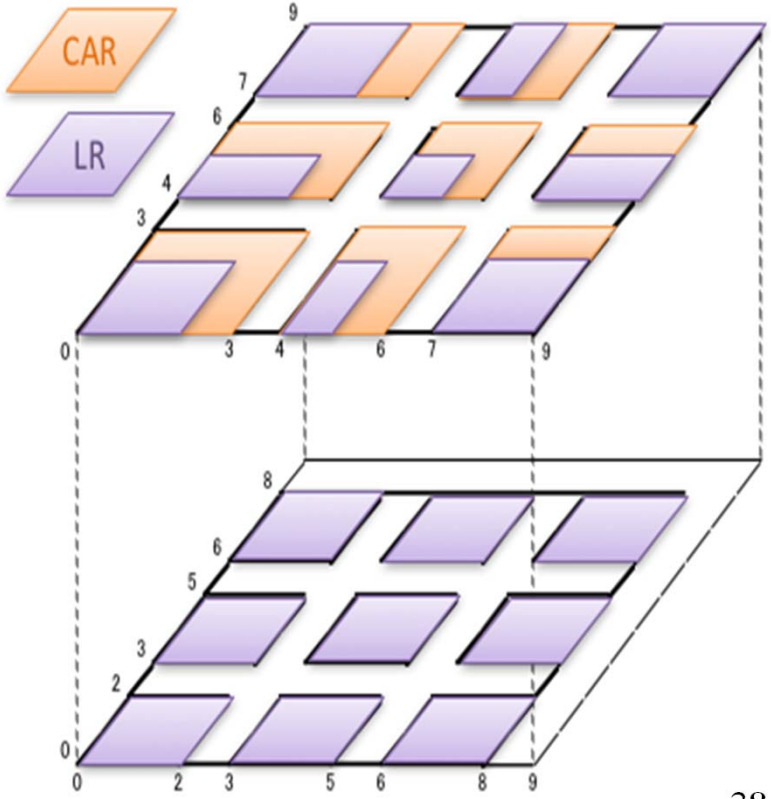
Data Localization: Loop Aligned Decomposition

- **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



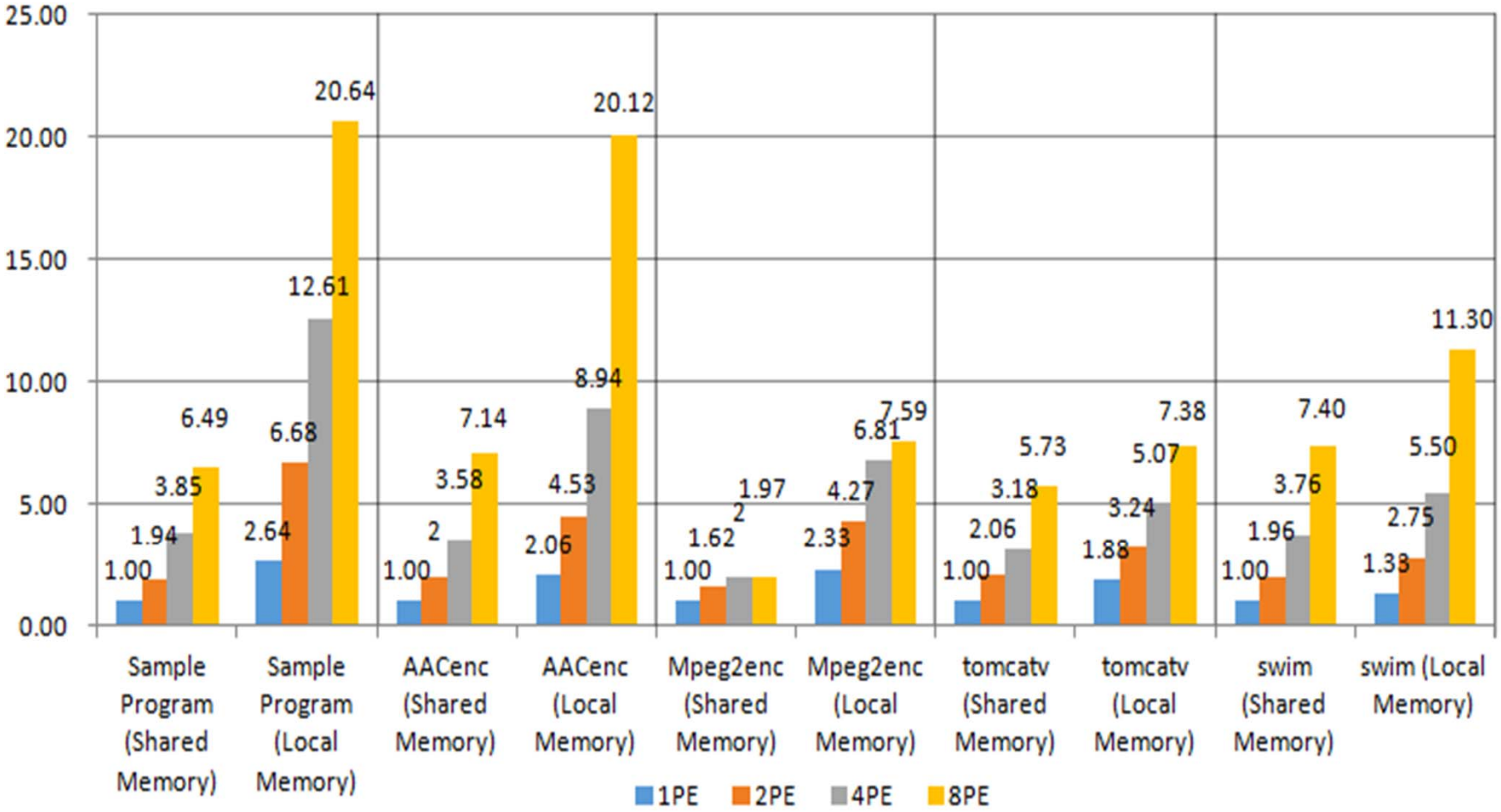
Multi-dimension Decomposition



Block Replacement Policy

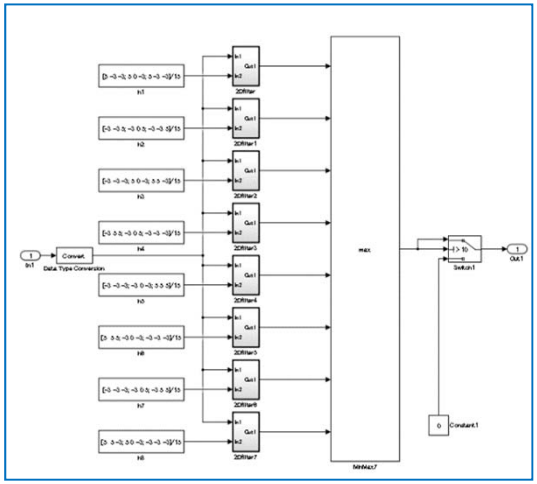
- Compiler Control Memory block Replacement
  - using live, dead and reuse information of each variable from the scheduled result
  - different from LRU in cache that does not use data dependence information
- Block Eviction Priority Policy
  1. (Dead) Variables that will not be accessed later in the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

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



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

OSCAR Compile Flow for Simulink Applications



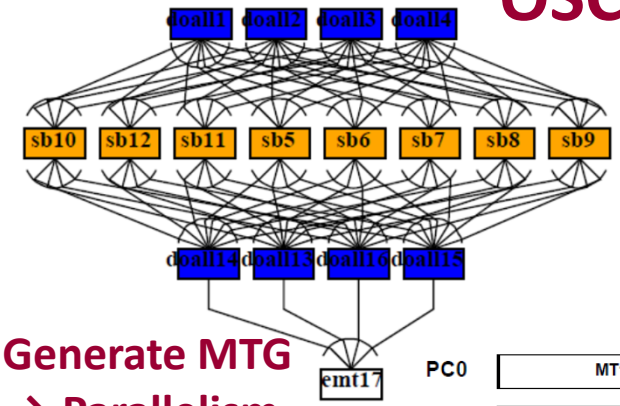
Simulink model

Generate C code
using Embedded Coder

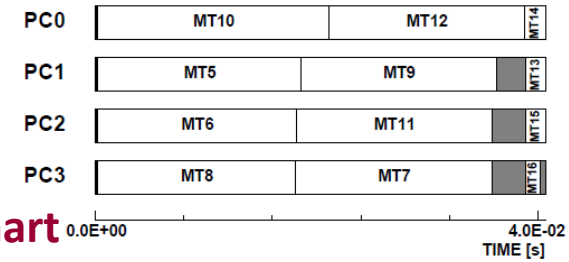
```
/* Model step function */
void VesselExtraction_step(void)
{
  int32_T i;
  real_T u0;

  /* 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



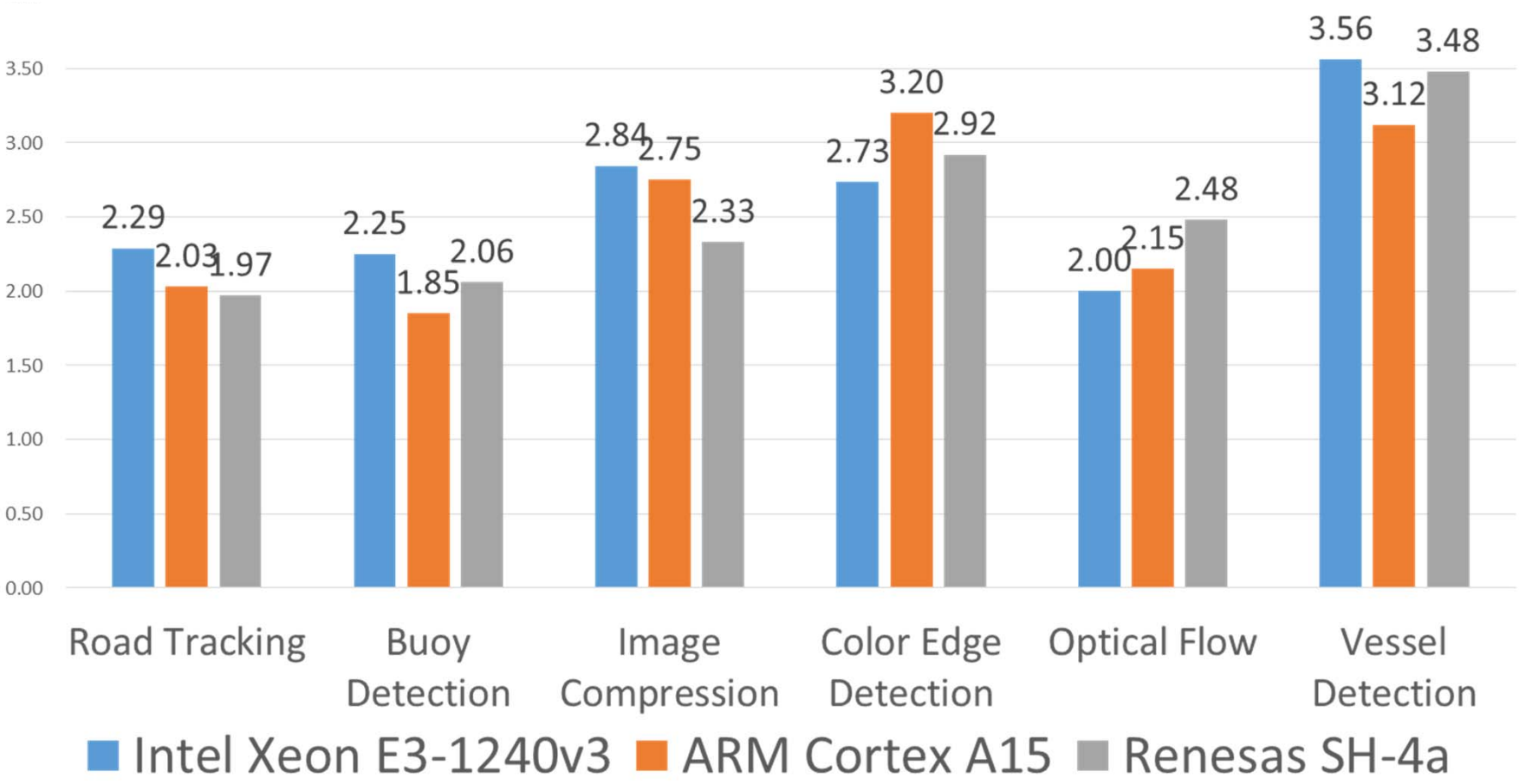
(2) Generate gantt chart
→ Scheduling in a multicore

```
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)

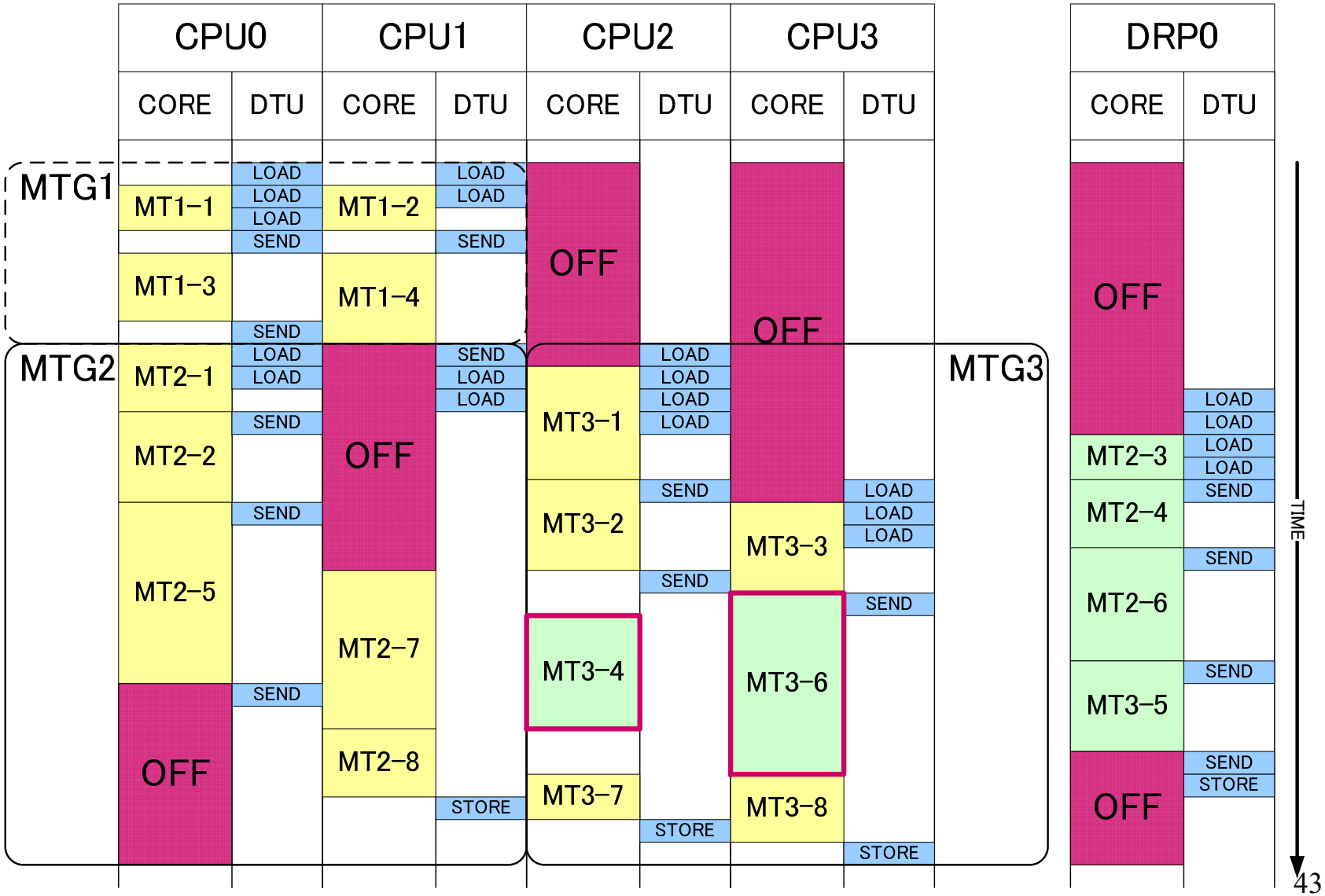
Speedups of MATLAB/Simulink Image Processing on Various 4core Multicores

(Intel Xeon, ARM Cortex A15 and Renesas SH4A)

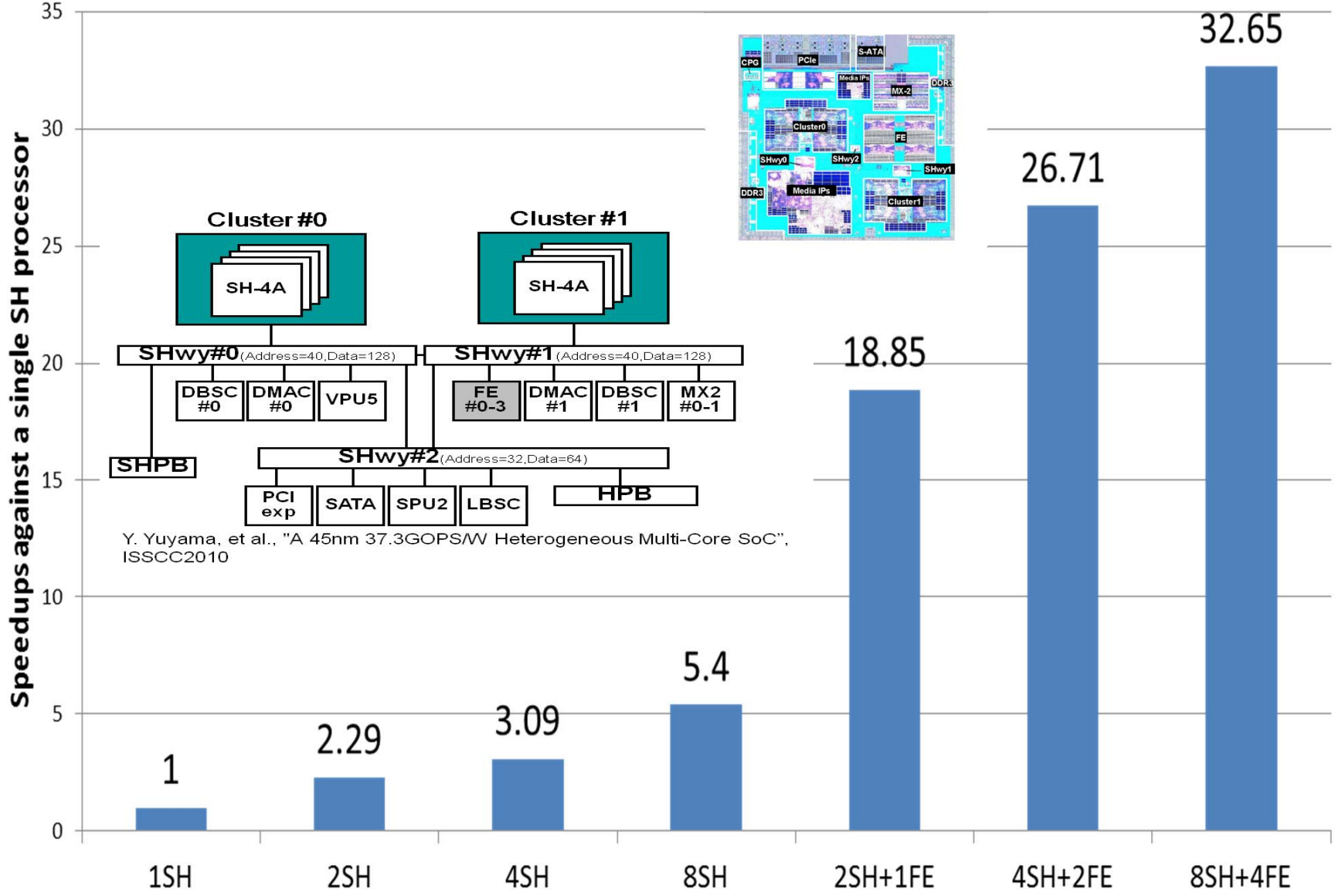


Road Tracking, Image Compression : http://www.mathworks.co.jp/jp/help/vision/examples
Buoy Detection : http://www.mathworks.co.jp/matlabcentral/fileexchange/44706-buoy-detection-using-simulink
Color Edge Detection : http://www.mathworks.co.jp/matlabcentral/fileexchange/28114-fast-edges-of-a-color-image--actual-color--not-converting-to-grayscale-/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

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



RPX上でのオプティカルフロー計算において、8つのSH4Aプロセッサと4つのアクセラレータFEGA（動的再構成可能プロセッサ）利用時に逐次に比べ33倍高速化



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

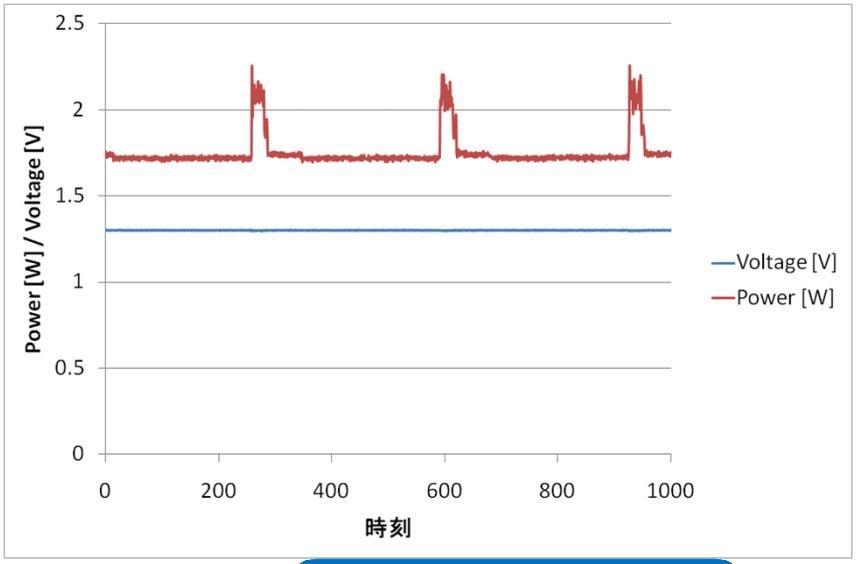
Without Power Reduction

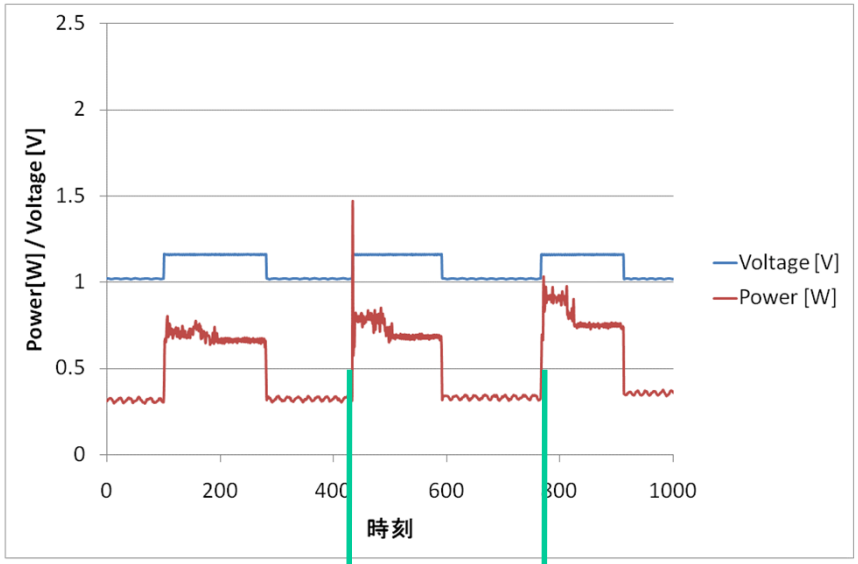
With Power Reduction by OSCAR Compiler

70% of power reduction

Average:1.76[W]

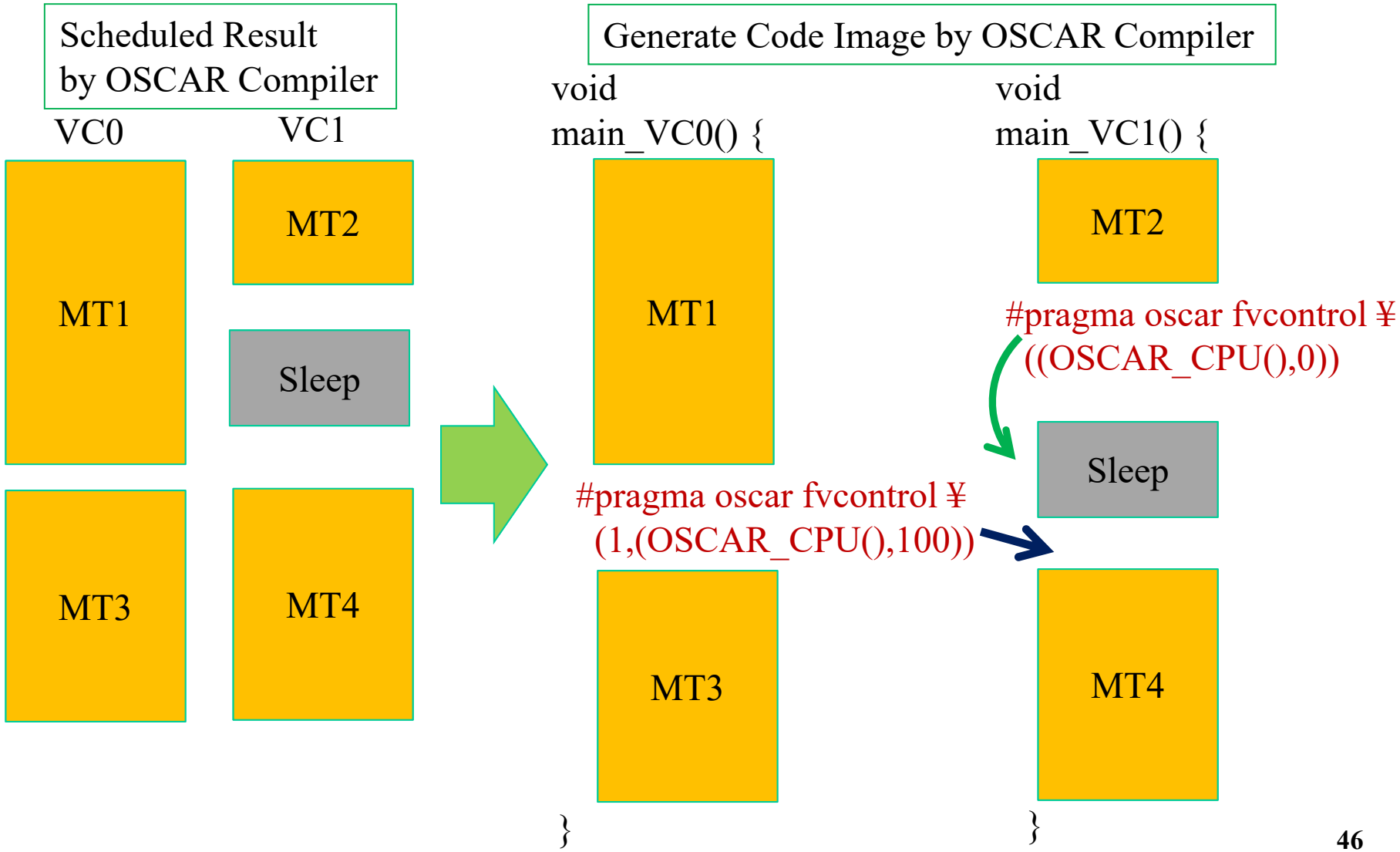
Average:0.54[W]





1cycle : 33[ms]
→30[fps]

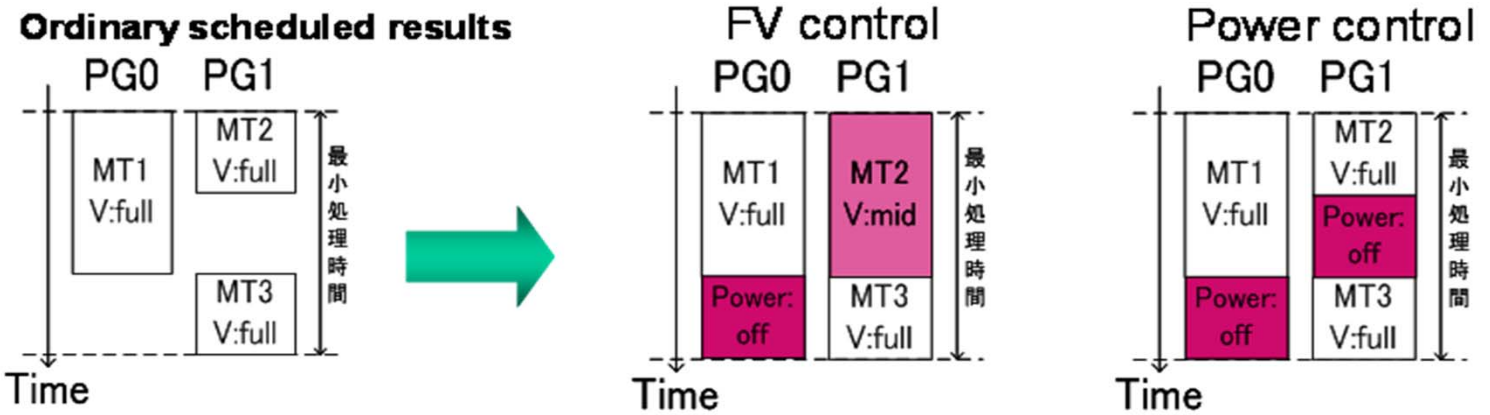
Low-Power Optimization with OSCAR API



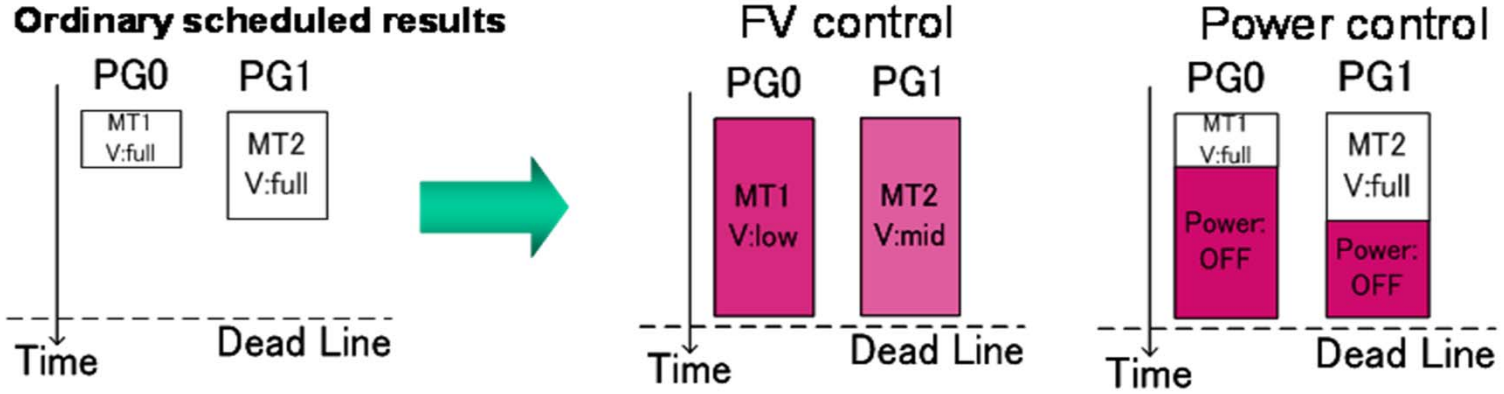
Power Reduction by Power Supply, Clock Frequency and Voltage Control by OSCAR Compiler

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

- **Shortest execution time mode**



- Realtime processing mode with dead line constraints



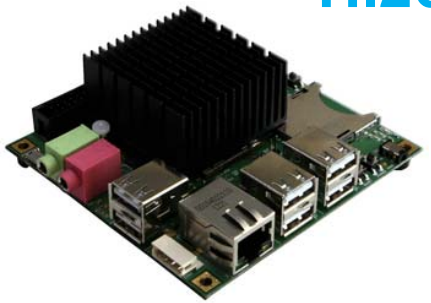
In this Fig. Frequency Full, Mid, Low

Power OFF: Power Gating

ARM CortexA9 4コアAndroid上での電力削減

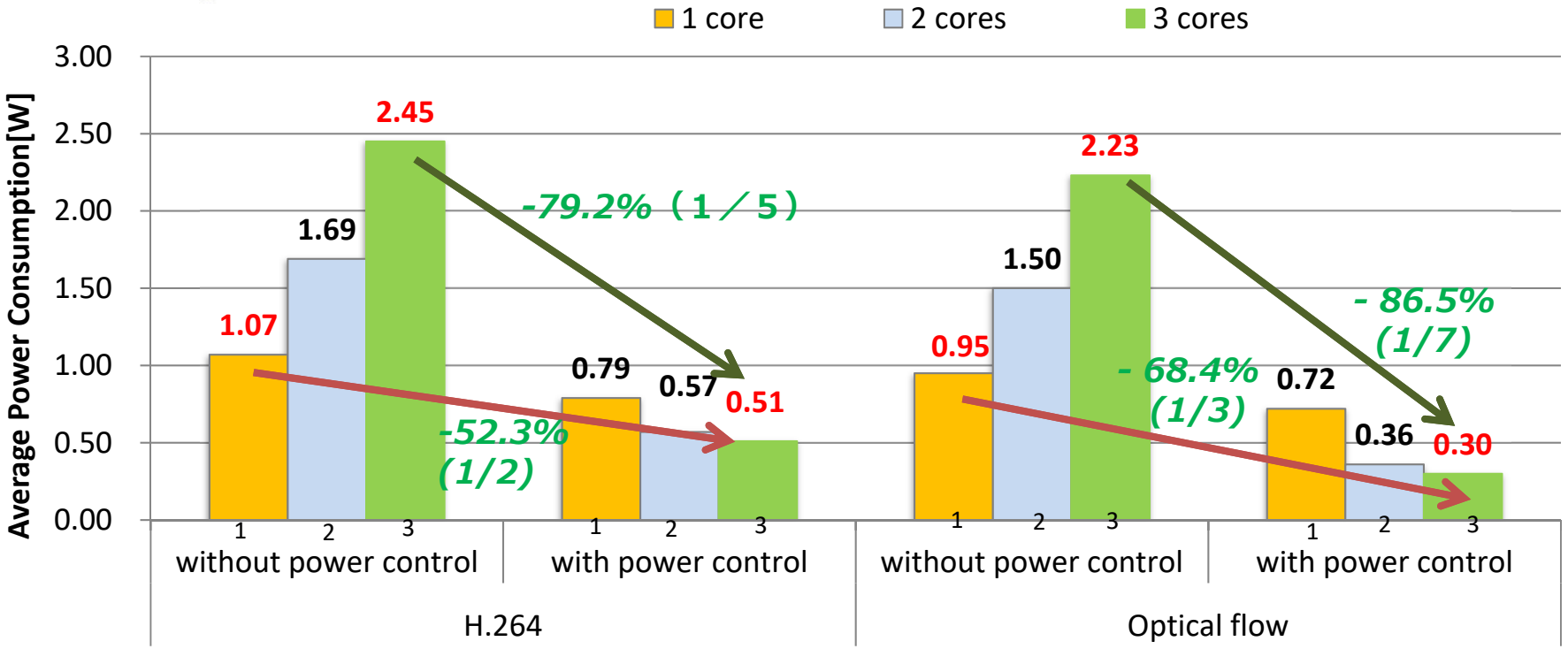
http://www.youtube.com/channel/UCS43lNYEIkC8i_KIgFZYQBQ

H.264 decoder & Optical Flow (3コア使用)



ODROID X2

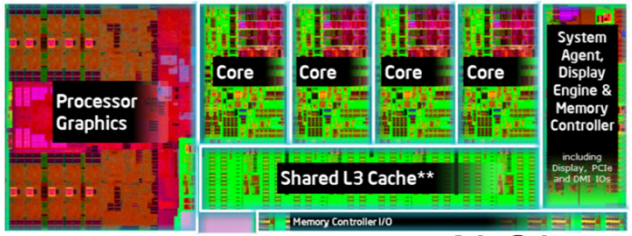
Samsung Exynos4412 Prime, ARM Cortex-A9 Quad core
1.7GHz～0.2GHz, used by Samsung's Galaxy S3



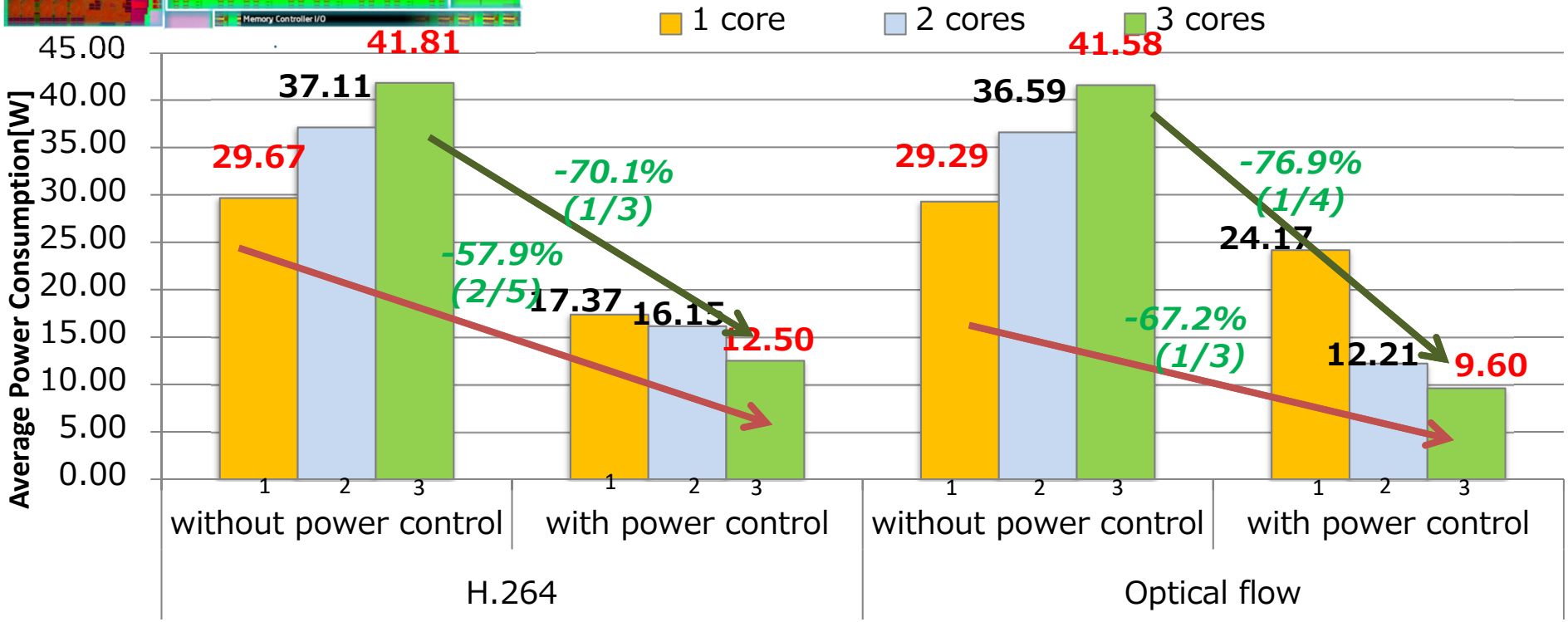
3PE電力制御なしと3PE電力制御ありで電力を1/5～1/7に削減
1PE電力制御なしと3PE電力制御ありで電力を1/2～1/3に削減

Intel Haswell上での電力削減

H.264 decoder & Optical Flow （3コア使用）

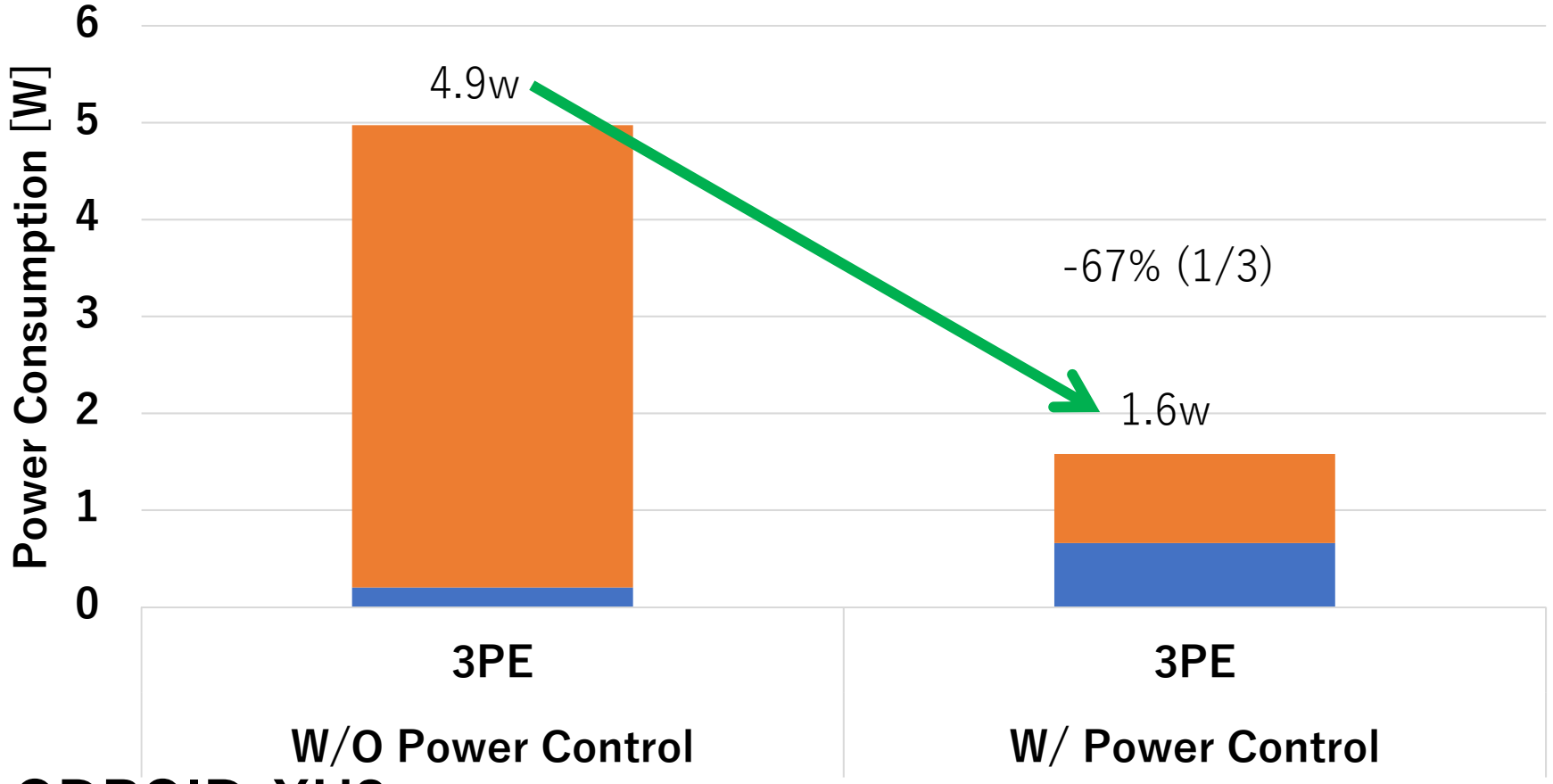


H81M-A，Intel Core i7 4770k
Quad core，3.5GHz～0.8GHz



3PE電力制御なしと3PE電力制御ありで電力を1/3～1/4に削減
1PE電力制御なしと3PE電力制御ありで電力を2/5～1/3に削減

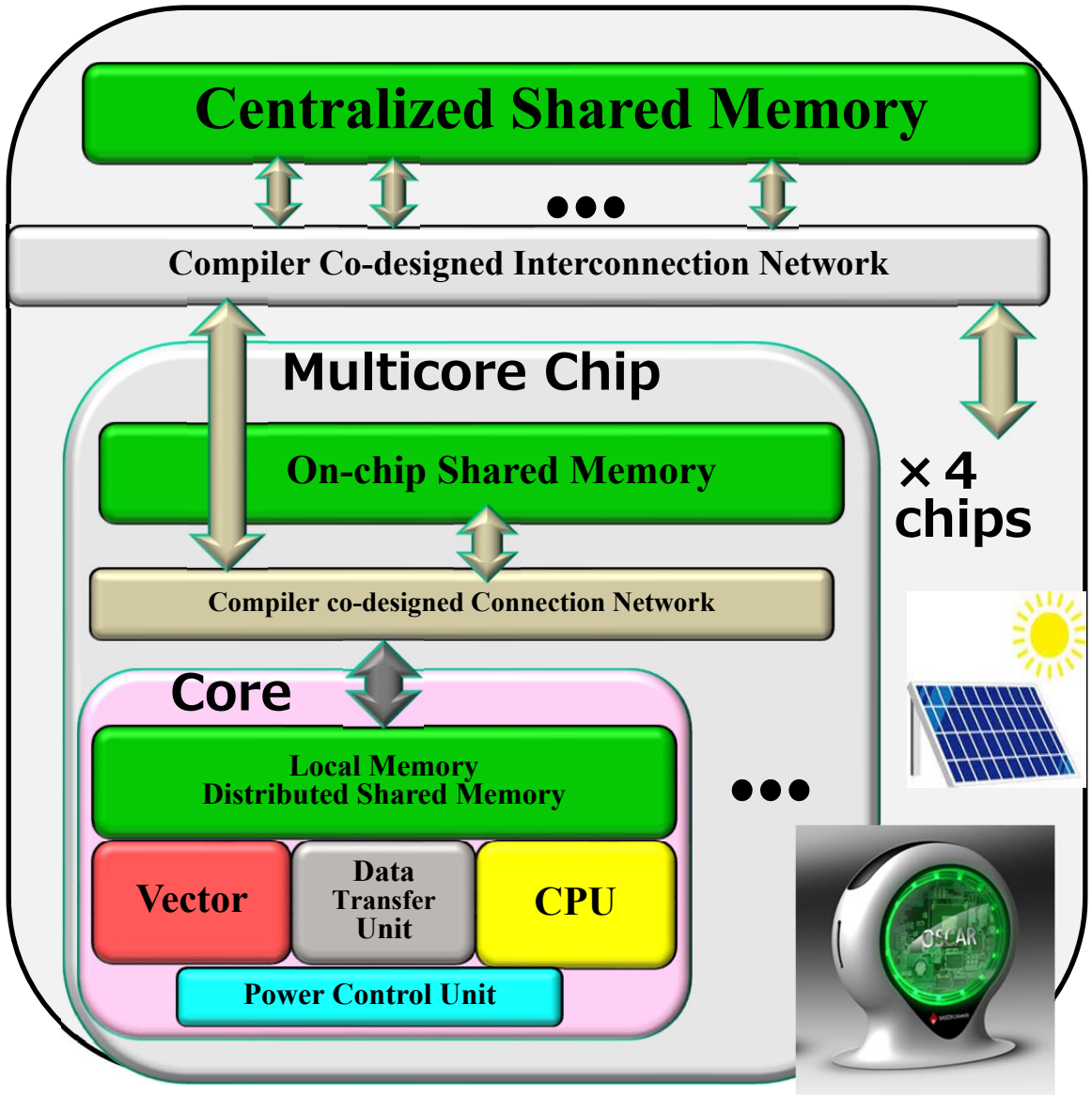
Automatic Power Reduction of OpenCV Face Detection on big.LITTLE ARM Processor



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

Frequency can be changed by each cluster unit

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



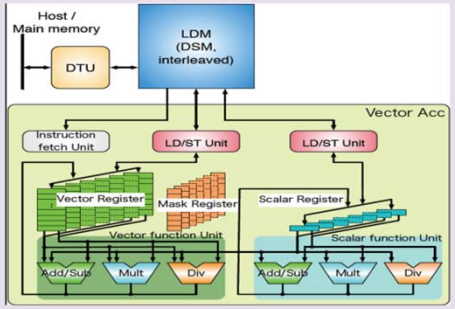
Target:

- **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

まとめ

- 早稲田大学グリーンコンピューティング研究開発センターでは、低消費電力・高性能なグリーンマルチコアコンピューティングシステムのハードウェア、ソフトウェア、応用の研究開発・実用化を産官学連携で行っている。
- OSCARコンパイラは、35年の研究開発により、科学技術計算、医療画像処理、災害シミュレーション、自動車エンジン制御等に使用するホモジニアス・ヘテロジニアスマルチコア用のプログラムの自動並列化・メモリ最適化に世界で唯一成功。
- 自動並列化は、Intel, ARM, IBM, AMD, Qualcomm, Infineon, ルネサス, 富士通等種々のマルチコア用の並列プログラムの自動作成を行え、性能的には重粒子線ガン治療計算で144コアで328倍、地震波伝搬シミュレーションで128コアで110倍、自動車エンジン制御計算で2コアで1.95倍、EV用バッテリ磁性材料3DFFTで128コアで120倍、カプセル内視鏡用画像圧縮処理で64コアで55倍、MATLAB/Simulinkの自動並列化で4コアで3.6倍等の対逐次・速度向上を得ている。
  - コンパイラ実用化及び産学連携実現のためのオスカーテクノロジー社を設立
    - 自動車用製品版コンパイラOSCARTech Compiler Ver.2　が出荷中
    - 自動車・医療・災害時避難指示を目指した超低消費電力・高性能・高ソフト生産性アクセラレータ付きマルチコアも笠原・木村研とともに開発中
- 電力削減では、世界で初めてリアルタイムアプリケーション並列動作中の電力削減に成功し、 ルネサス， ARM， Intel Haswell上で、電力を 3コアで1コアと比べ、1/2から1/3に削減。BigLittleヘテロジニアスでも電力を1/3に自動削減。