エクストリームスケールコンピューティング時代の省電力技術

東京大学大学院情報理工学系研究科
東京大学情報基盤センター
近藤 正章

謝辞

- 本チュートリアル資料の作成にあたり、以下の方々を始めとして、多くの方にご助言・ご協力を頂きました。ここに感謝いたします。
  - 中村 宏 教授（東京大学情報理工学系研究科）
  - 中田 尚 特任助教（東京大学情報理工学系研究科）
  - Cao Thang 特任研究員（東京大学情報理工学系研究科）
  - He Yuan 特任研究員（東京大学情報理工学系研究科）

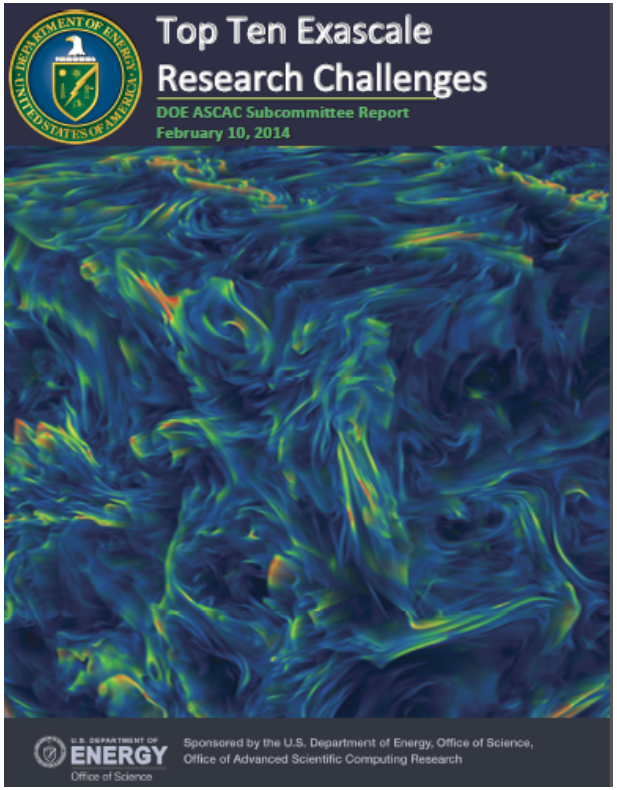
エクストリームスケールコンピューティングへの壁

- エクストリームスケールシステム
  - 社会的・科学的課題解決のためExaflops級の性能を持つシステム
  - 日本では「フラグシップ2020プロジェクト」として開発
  - 米国/欧州/中国でもエクサスケールシステムの開発が計画・構想中
- エクサスケールシステムへの課題
  - 信頼性の壁ーエクサスケールHPLの実行には1週間近くを要する
  - 電力の壁ー20MWでエクサには10倍近い電力効率向上が必要
  - 低B/F、少メモリ容量、深いメモリ階層、プロダクティビティ・・・

その中でも消費電力は特に重要な課題

Top Ten Exascale Research Challenges [DOE2014]

1. Energy efficiency: Creating more energy-efficient circuit, power, and cooling technologies
2. Interconnect technology: Increasing the performance and energy efficiency of data movement
3. Memory Technology: …
4. Scalable System Software: Developing scalable system software that is power- and resilience-aware
5. Programming systems: …
6. Data management: …
7. Exascale Algorithms: …
8. Algorithms for discovery, design, and decision: …
9. Resilience and correctness: …
10. Scientific productivity: …



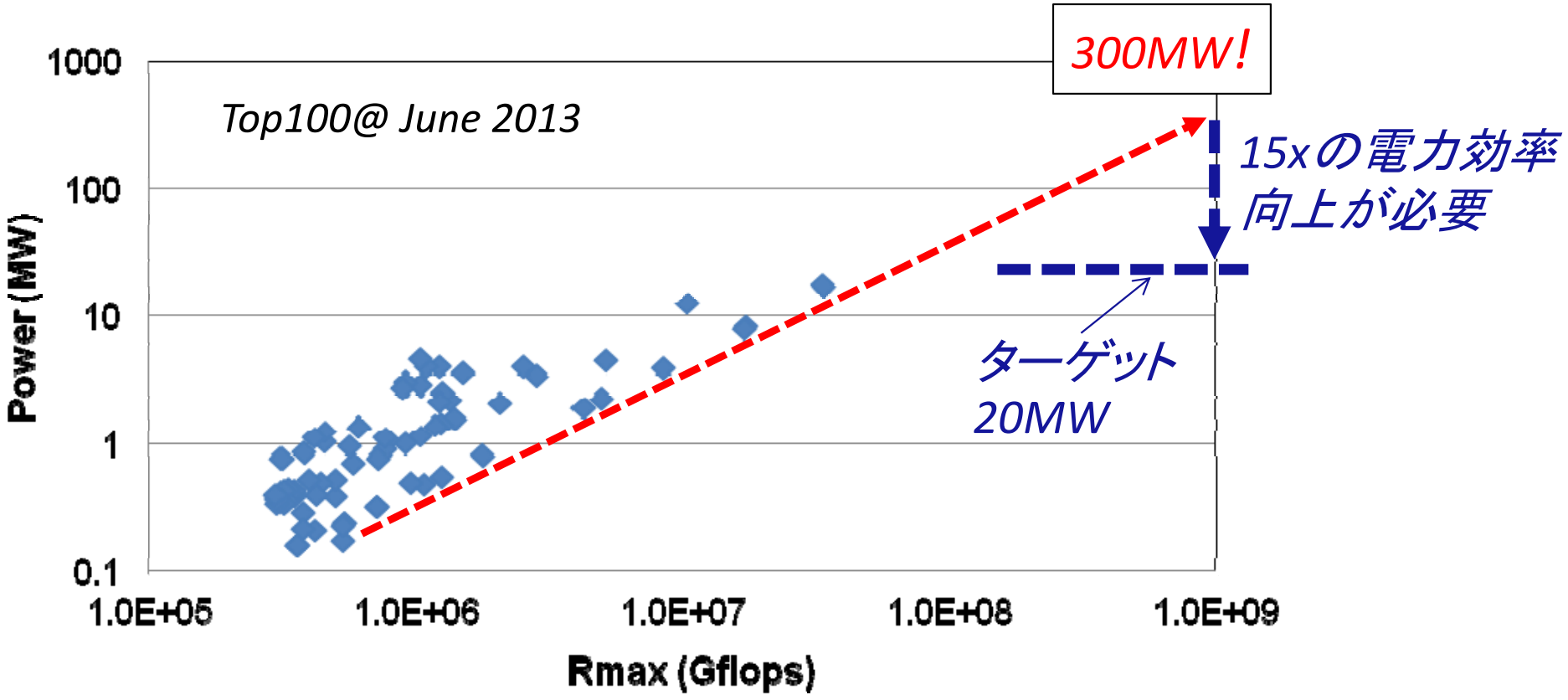
省電力・省エネ関係が最重要課題として挙げられている

本チュートリアルの構成

- 高性能計算機の消費電力トレンド
- 計算機システムにおける電力消費の基礎
- 省電力・省エネ技術
  - Dark-silicon問題とプロセッサの省電力・省エネ化技術
  - メモリの省電力化技術
  - インターコネクションネットワークの省電力化技術
  - システムソフトウェアレベルでの電力制御技術
- 将来展望

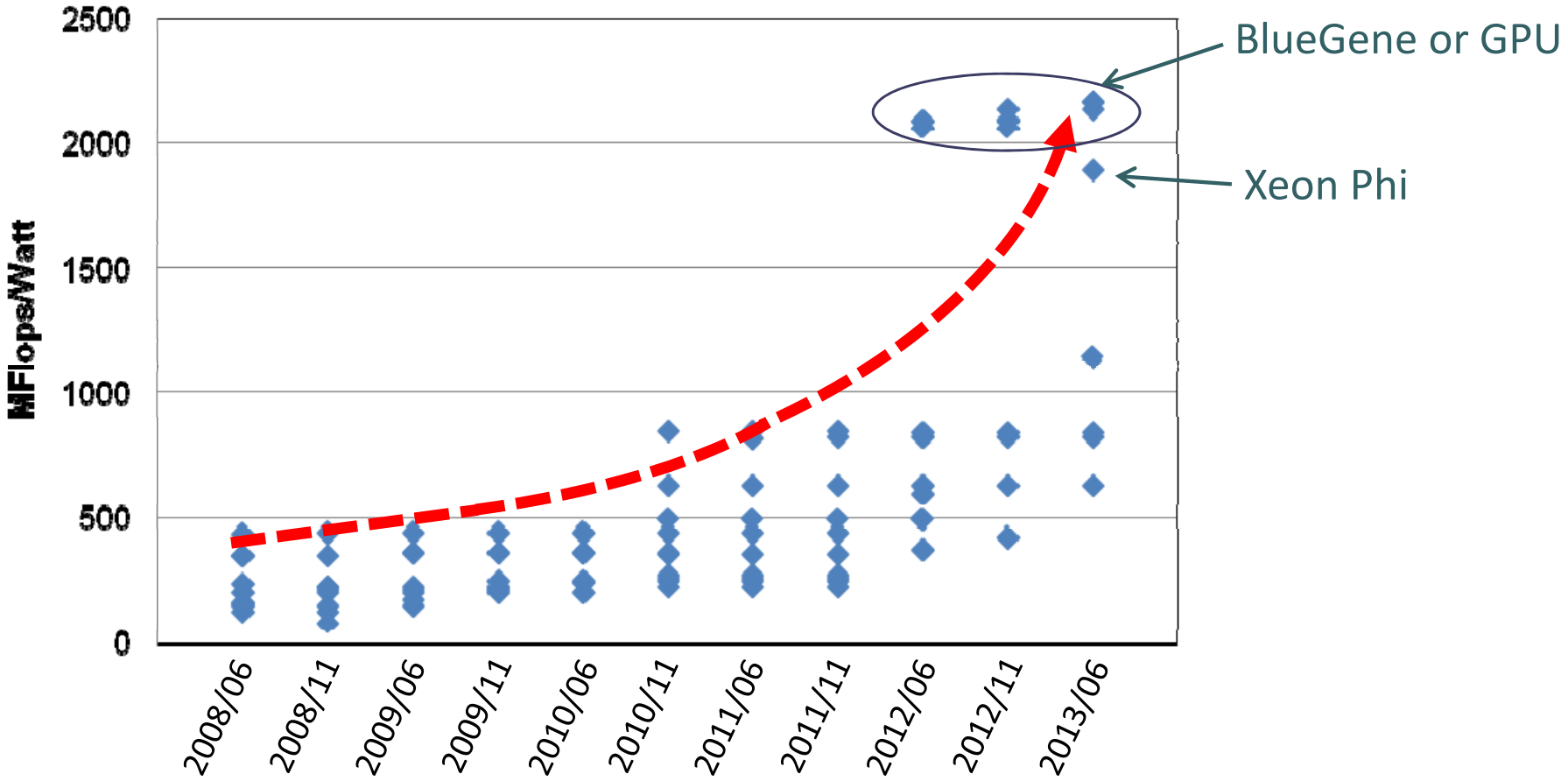
現在のスパコンの電力効率

- Top100スパコンのLinpack性能と消費電力
  - Top100中で最も電力効率が良いスパコン: 3186MFlops/Watt
- 20MWでExaFlops実現には15倍の電力効率向上が必要



システムレベルの電力あたり性能(MFlops/Watt)

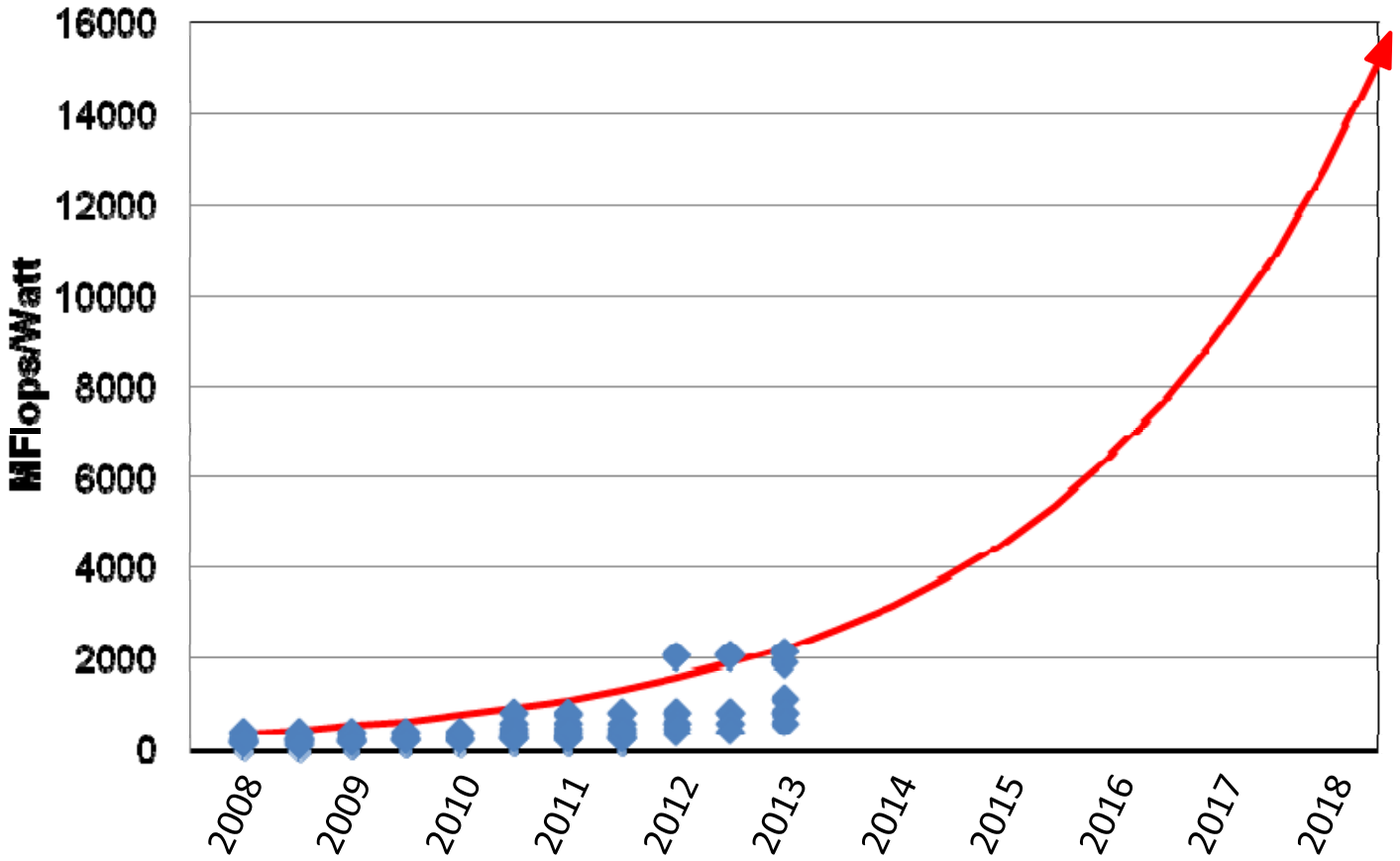
- Top10スパコンのMFlops/Watt
  - 2年(1プロセス世代)でおよそ2倍の向上
  - 今後は電力効率向上ペースは鈍化すると予想されている



システムレベルの電力あたり性能の将来トレンド

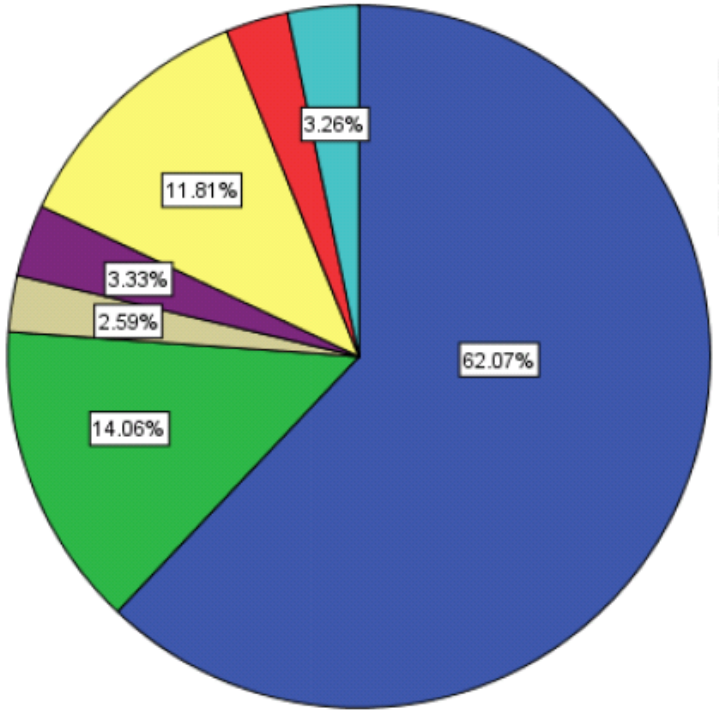
- 2年で2倍のFlops/Watt向上が続くとしても・・・
  - 2018年では10～20GFlops/Watt → エクサシステムで50MW

→ 電力効率向上のためのさらなる技術開発が必須



現在のシステムの消費電力の内訳

- BlueGene/Qの消費電力の内訳　出典：[Wallance2013]



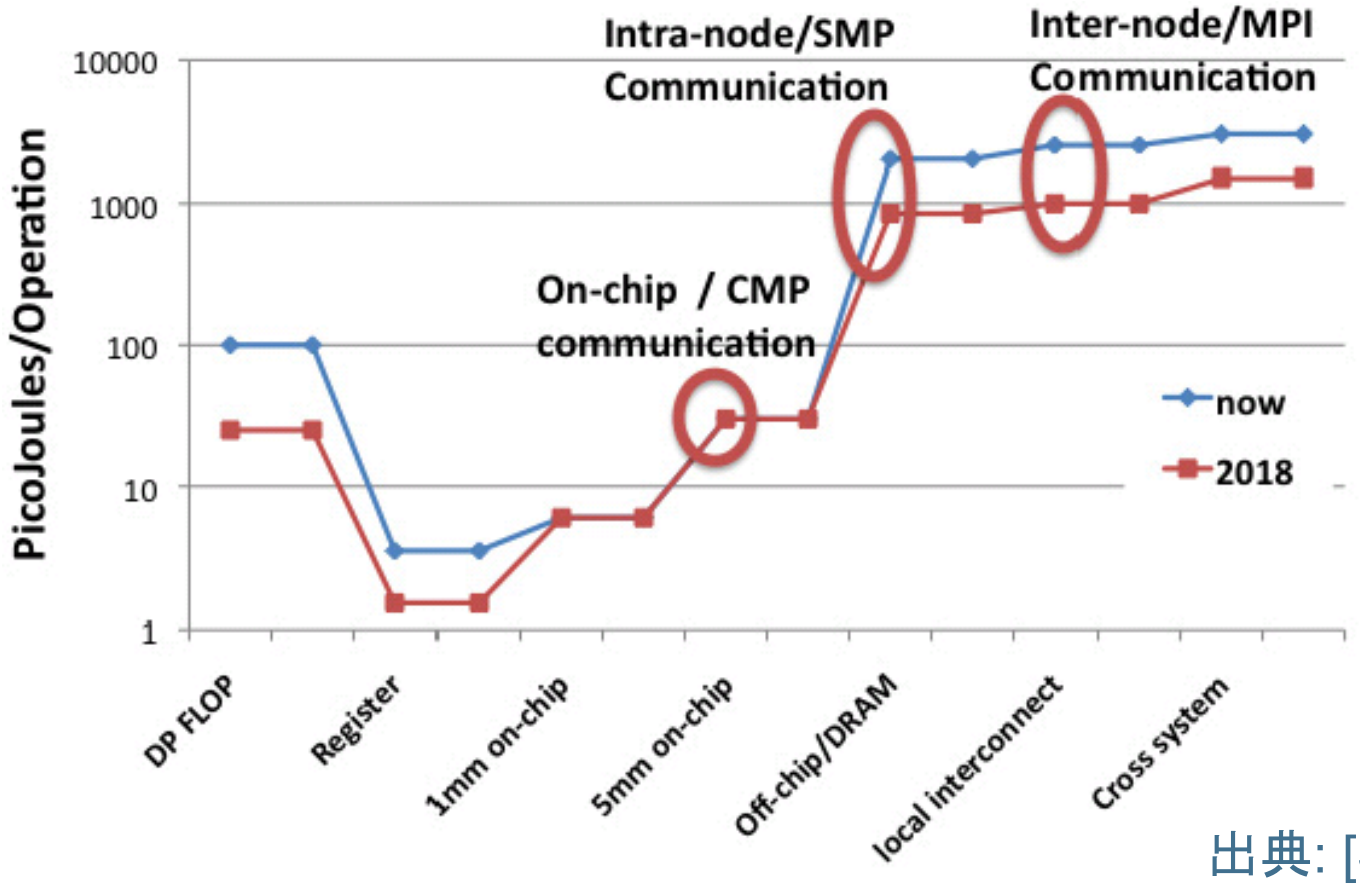
- BGQのenvironment databaseを通じて取得
- プロセッサ+SRAMの電力: 65%
- DRAM電力：約14%
- ネットワーク電力：約21%

Fig. 10. Pie chart showing relative percentages of total power usage consumed by each of the 7 power domains. Intense network activity largely contributing to optics percentage.

- BlueGeneでもプロセッサ・コア以外の消費電力は大きい

データ移動の電力コスト

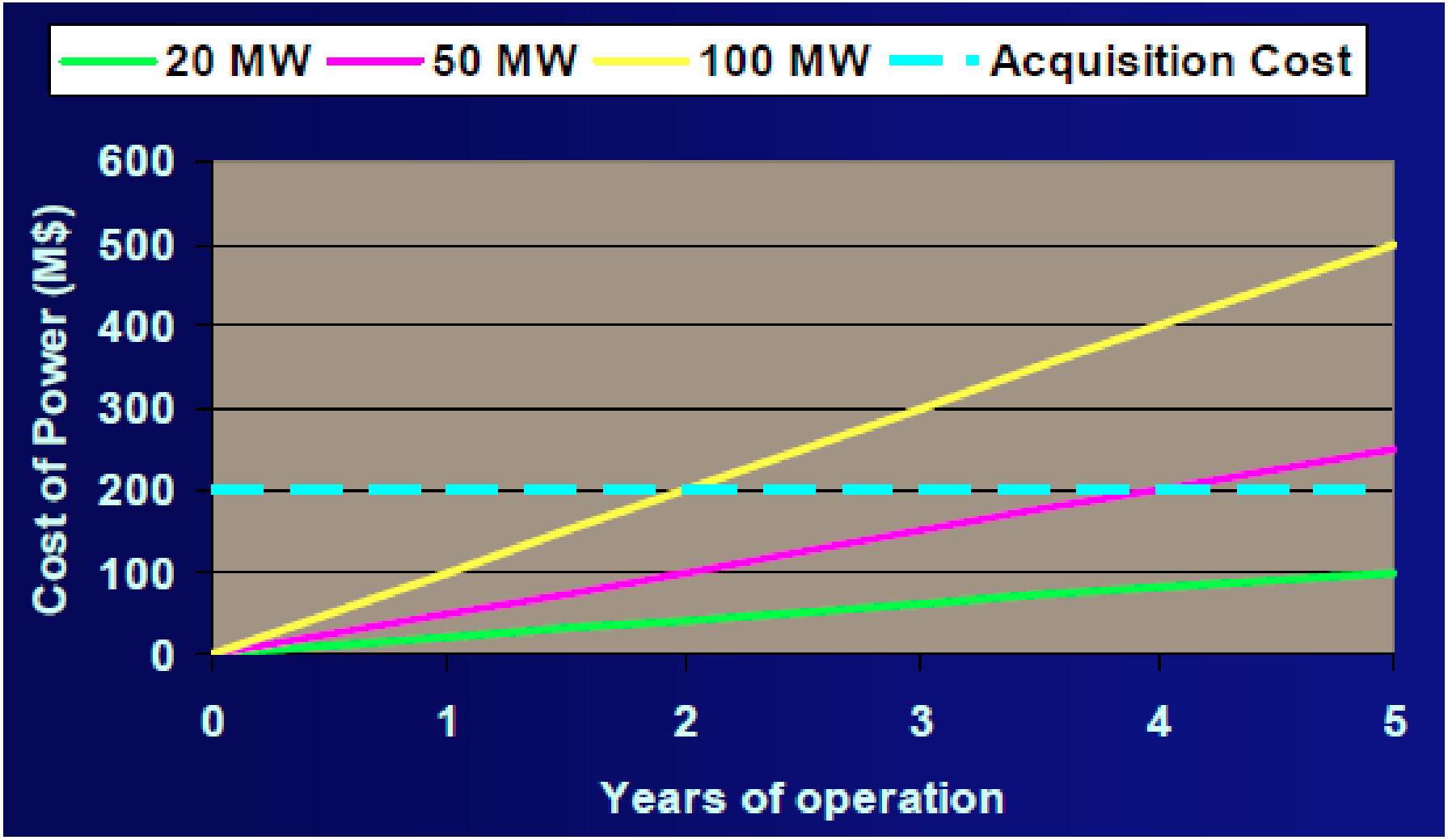
- オフチップデータ移動の電力コストは計算よりも高い
  - 今後はメモリやインターコネクトの電力削減が重要
  - データ移動を抑えるソフトウェア技術(局所性の活用)も重要に



出典: [Stevens09]

電気のコスト

- エクサシステムでは電気代がシステムコストに匹敵



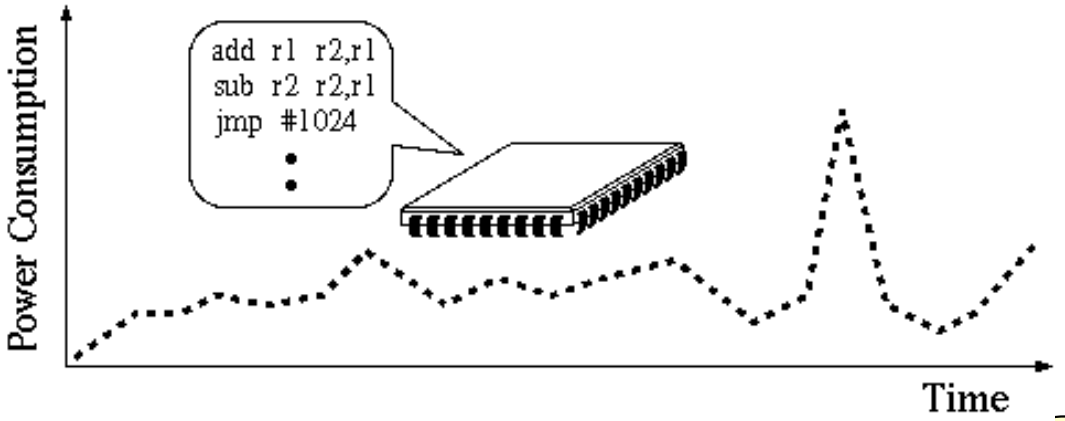
出典: [Nair2014]

本チュートリアルの構成

- 高性能計算機の消費電力トレンド
- 計算機システムにおける電力消費の基礎
- 省電力・省エネ技術
  - Dark-silicon問題とプロセッサの省電力・省エネ化技術
  - メモリの省電力化技術
  - インターコネクションネットワークの省電力化技術
  - システムソフトウェアレベルでの電力制御技術
- 将来展望

「消費電力」と「消費エネルギー」

- 消費エネルギーは消費電力の時間積分
  - 「消費電力」を削減すれば、かならず「消費エネルギー」を削減できるわけではない
  - 消費電力を1/2にしても、プログラム実行時間が2倍になれば消費するエネルギーは同じ！



$$E_{Chip} = \int_0^t P_{Chip}$$

E_{Chip}：チップの消費エネルギー
P_{Chip}：チップの消費電力
t：プログラム実行時間

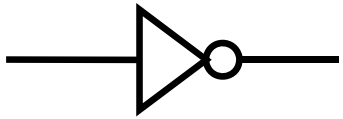
- 消費電力
  - 仕事率（瞬間発熱量）
  - Watt [W]、VA
  - パッケージ、冷却装置、電源、信頼性に影響

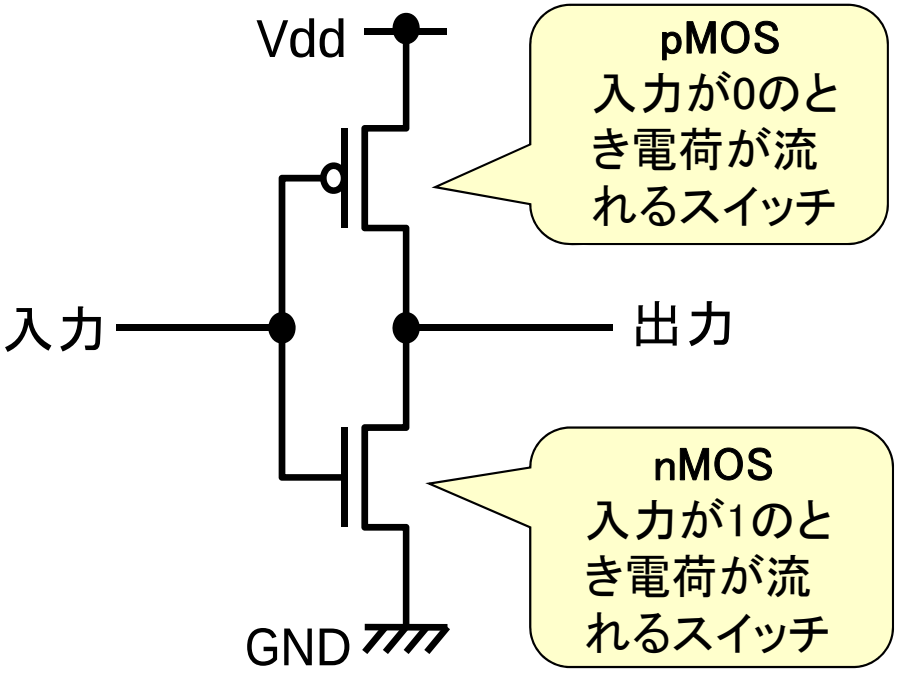
⇔

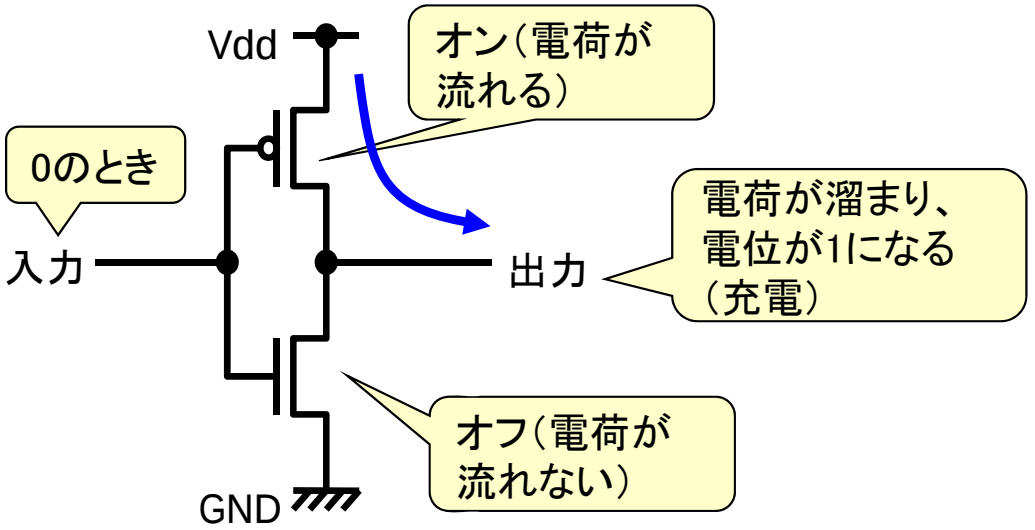
- 消費エネルギー
  - 仕事量
  - Joule [J]
  - バッテリ寿命に影響
  - 電気代

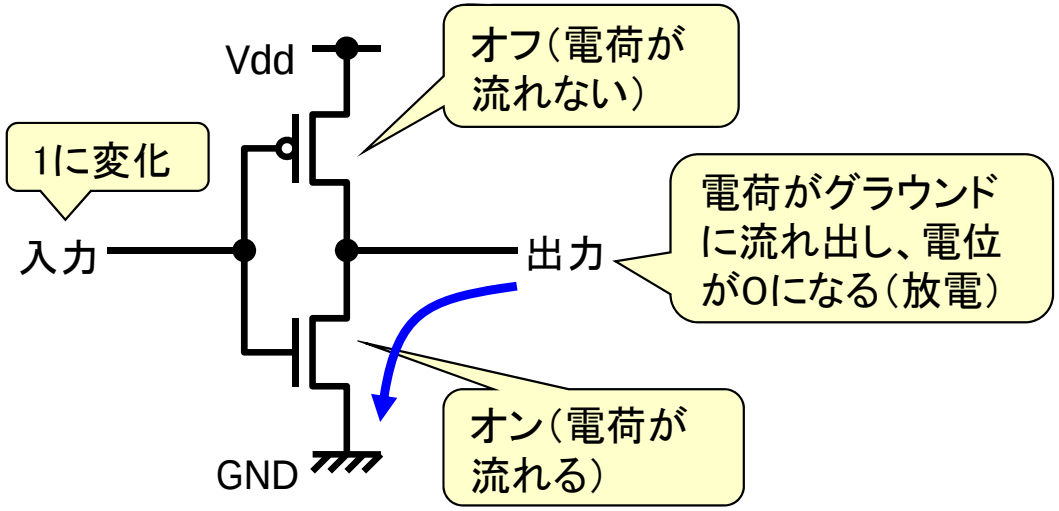
CMOS回路の基本構造：否定ゲート

- CMOS回路の動作原理と電荷の流れ
(否定ゲート)









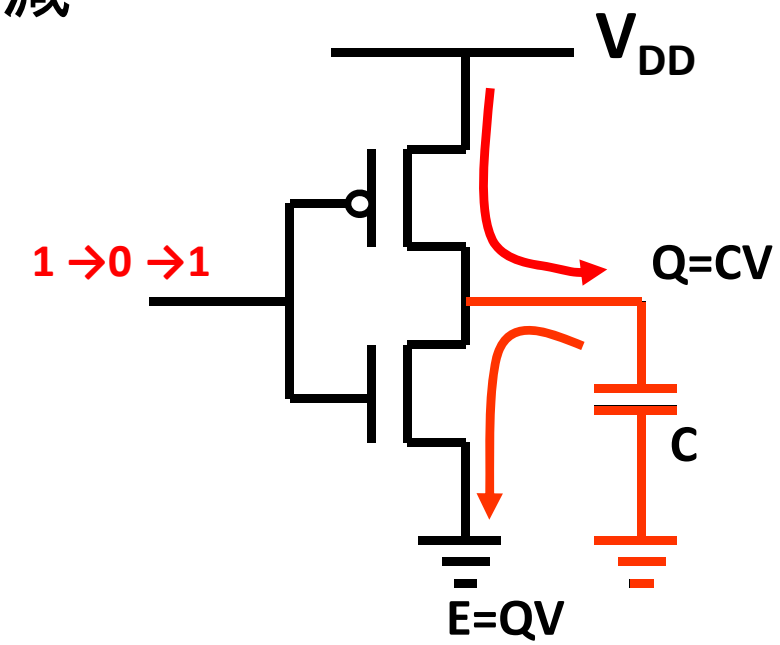
低電力化の基礎

- CMOS回路の消費電力

$$P = \alpha C V^2 f + V I_{leak}$$

α : switching activity
C : load capacitance
V : supply voltage
f : clock frequency
I_{leak} : leakage current

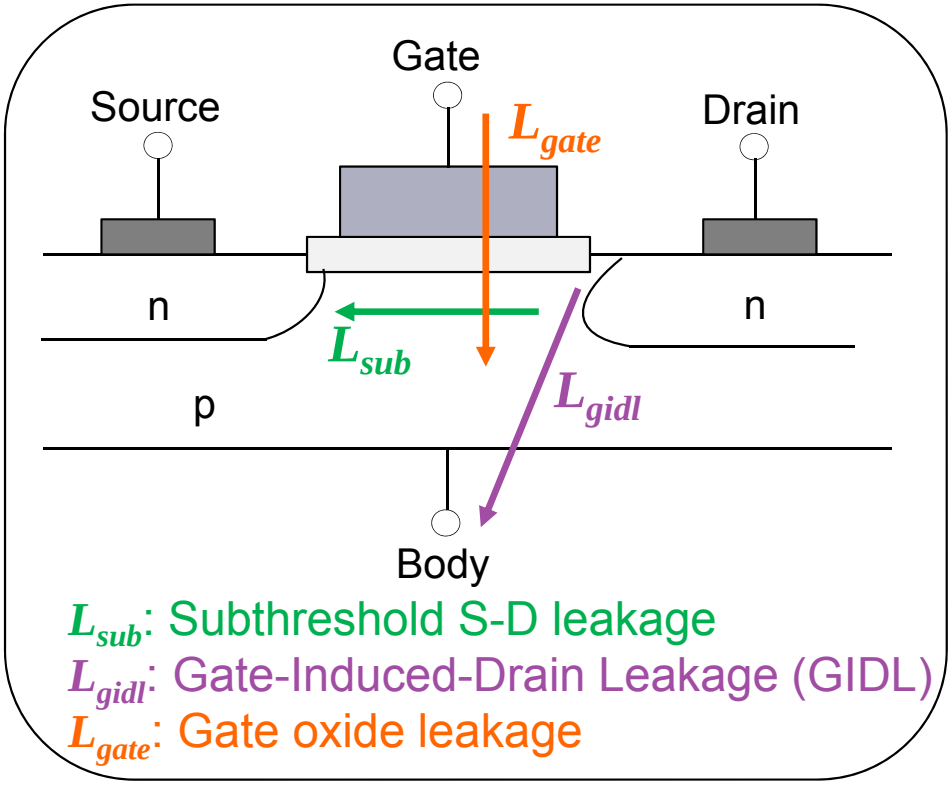
- 消費電力を削減するためには…
  1. トランジスタの遷移確率の削減
  2. 負荷容量の低減
  3. 電源電圧の低減
  4. 周波数の低減
  5. リーク電流の削減

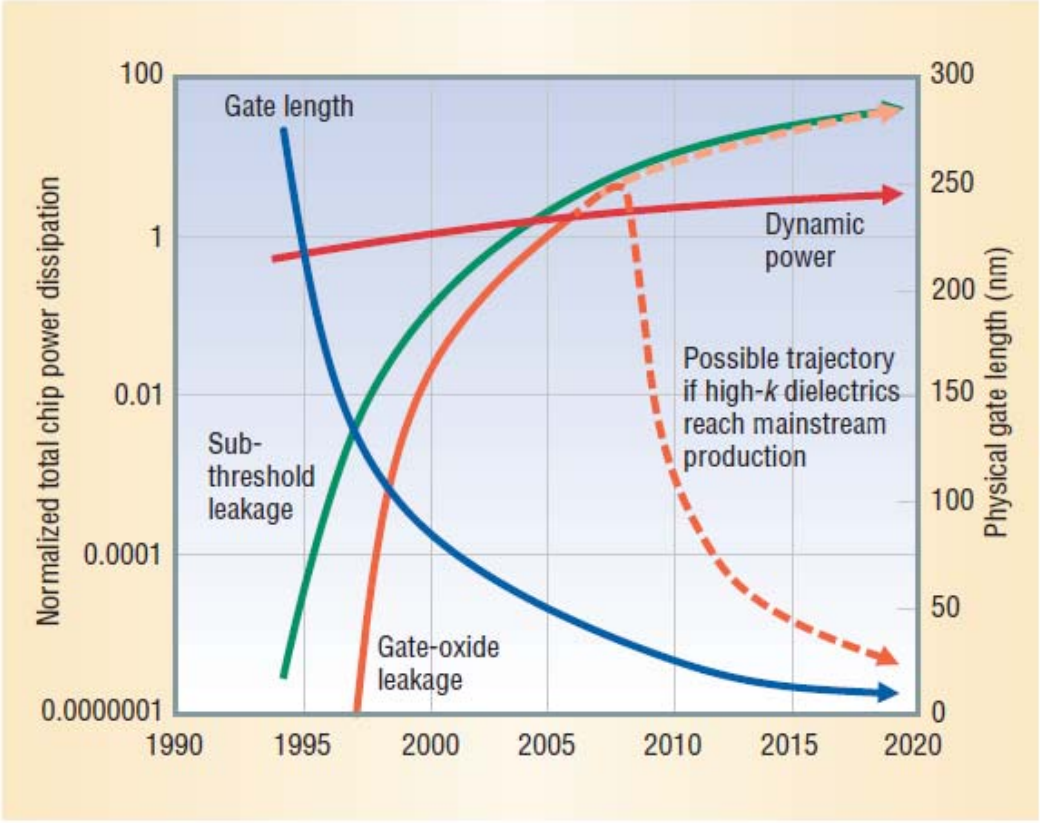


リーク電流

- リーク電流 (Leakage Current)
  - 非動作時にもトランジスタに流れてしまう電流
  - 半導体プロセスの微細化が原因
  - プロセス世代の進展にともない増大

出典: [Kim2003]





電力効率を向上させる技術の例

- プロセッサ
  - (既存技術の延長) DVFS、Power-gating、Clock-gating
  - 3次元積層技術、FinFET、トライゲート、FDSOI、SOTB、・・・
  - Low (Near-Threshold) Voltage Computing
  - メニーコア化、SIMD幅拡大、アクセラレータの効率的利用
  - 用途特化型回路の利用
- メモリ
  - 3次元積層技術(Hybrid Memory Cube、 Wide I/O、HBM)
  - 不揮発性メモリの利用
- インターコネクト
  - 動作モード制御(リンク幅/リンク速度のスケーリング)
  - Silicon Photonics
- システムソフトウェア、システムレベル電力マネージメント
  - Power-Capping、電力性能比を最適化するアルゴリズム/ライブラリ
  - 電力モニタリング／制御インタフェースの提供
  - Hardware Overprovisioning

Dynamic Voltage Frequency Scaling

- Dynamic Voltage Frequency Scaling (DVFS)
  - 動作電圧(クロック周波数)を落として実行
  - 時間に余裕がある場合、対象回路以外が性能ボトルネックの場合有効
    - 一般的にはゆっくり実行した方が消費エネルギー効率が良い
- 電源電圧を1/2に下げる場合(周波数も1/2)の例

$$P \propto Vdd^2 f + VddI_{leak}$$

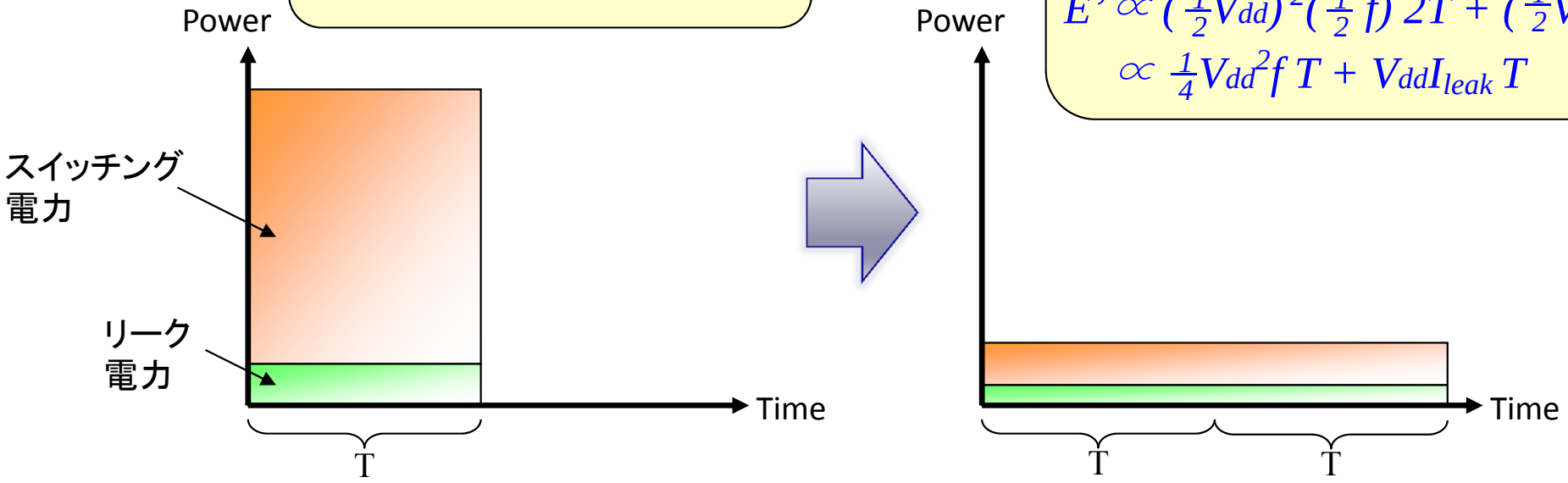
$$E \propto Vdd^2 f\,T + VddI_{leak}\,T$$

$$P' \propto (\tfrac{1}{2}Vdd)^2(\tfrac{1}{2}f) + (\tfrac{1}{2}Vdd)I_{leak}$$

$$\propto \tfrac{1}{8}Vdd^2 f + \tfrac{1}{2}VddI_{leak}$$

$$E' \propto (\tfrac{1}{2}Vdd)^2(\tfrac{1}{2}f)\,2T + (\tfrac{1}{2}Vdd)I_{leak}\,2T$$

$$\propto \tfrac{1}{4}Vdd^2 f\,T + VddI_{leak}\,T$$



代表的なリーク電力削減技術

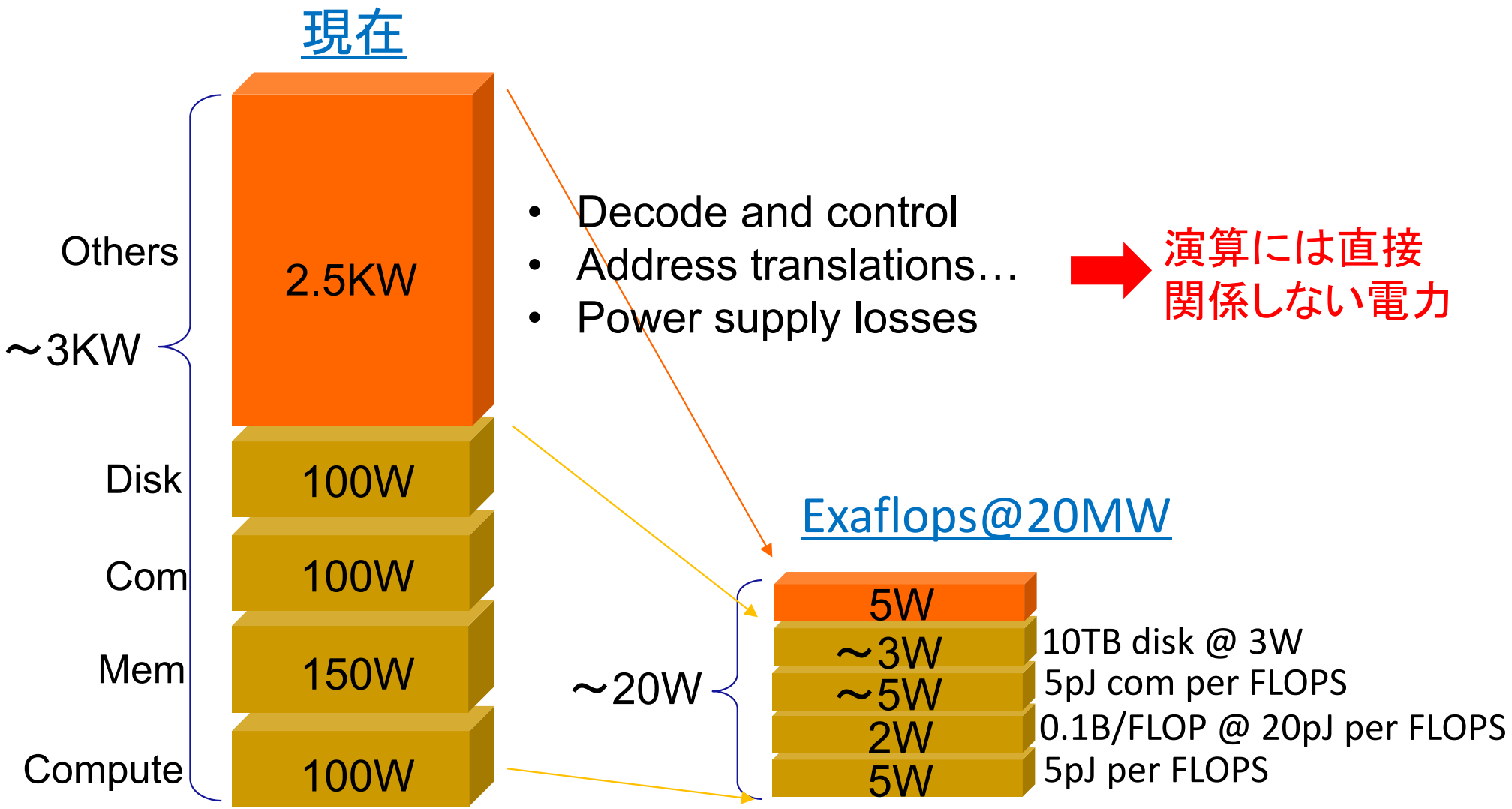
- 閾値電圧の変更（高閾値電圧化）
  - Dual-Vth（Dual Threshold Voltage）
  - VTCMOS（Variable-Threshold-voltage CMOS）
- 電源電圧の変更（電源供給の停止または低電源電圧化）
  - MTCMOS（Multi-Threshold-voltage CMOS） or Power Gating
  - MSV（Multi-Supply Voltage）
  - DVS（Dynamic Voltage Scaling）
- 入力データの設定
  - IVC（Input Vector Control）
- デバイス／プロセス技術の向上
  - Multi-gateトランジスタ
- 不揮発性メモリの利用

本チュートリアルの構成

- 高性能計算機の消費電力トレンド
- 計算機システムにおける電力消費の基礎
- 省電力・省エネ技術
  - Dark-silicon問題とプロセッサの省電力・省エネ化技術
  - メモリの省電力化技術
  - インターコネクションネットワークの省電力化技術
  - システムソフトウェアレベルでの電力制御技術
- 将来展望

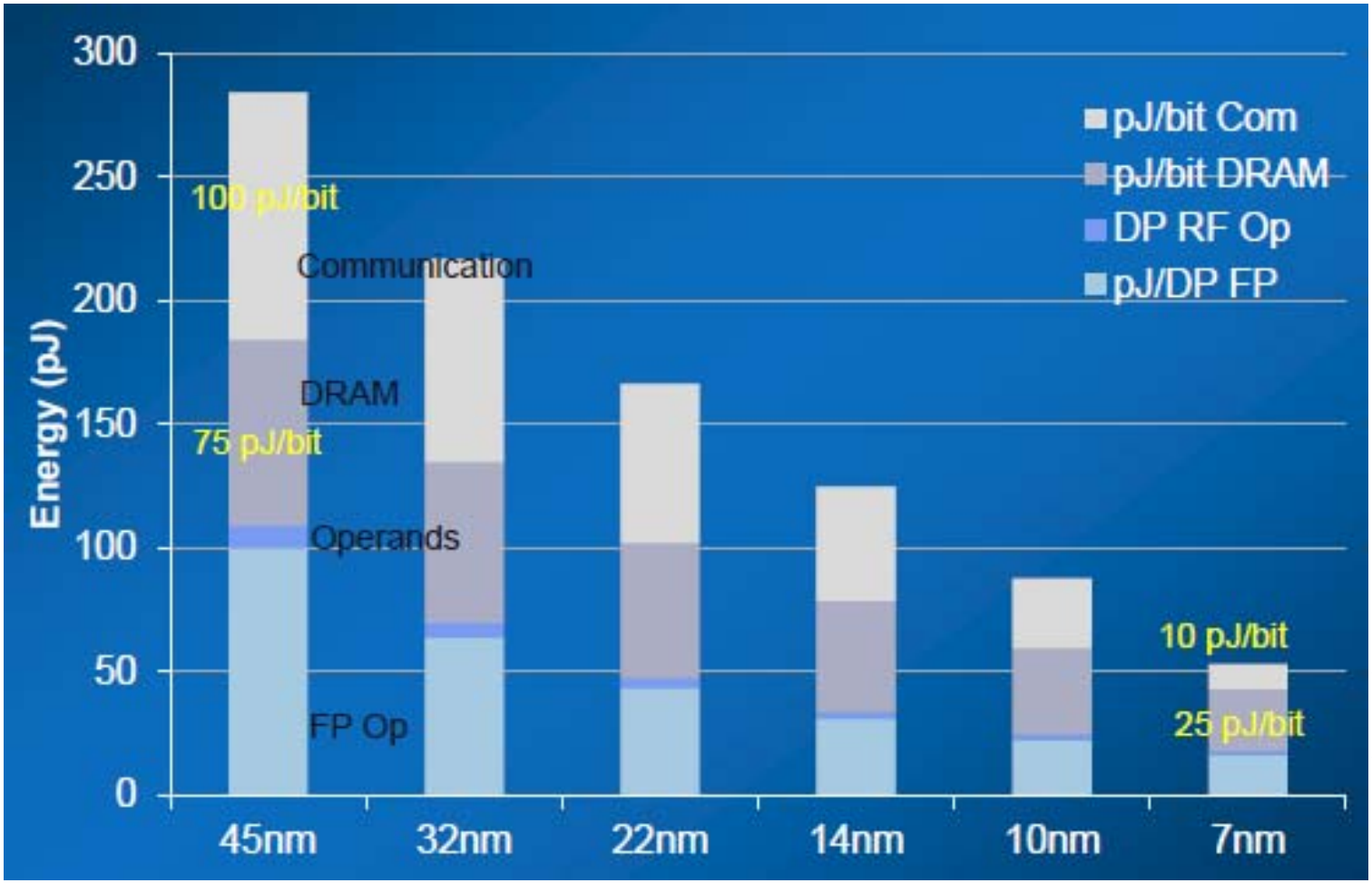
システムレベルでの電力消費の内訳

- Tera-FLOPSシステムの電力　参考：[Borkar2013]



Energy per Compute Operation

- 浮動小数点演算あたりのエネルギーのトレンド予測



20MWで達成可能な性能

← 0.1EFLOPS

← 0.13EFLOPS

← 0.2EFLOPS

← 0.4EFLOPS

← 1.0EFLOPS

出典：[Borkar2013]

Dark Silicon問題

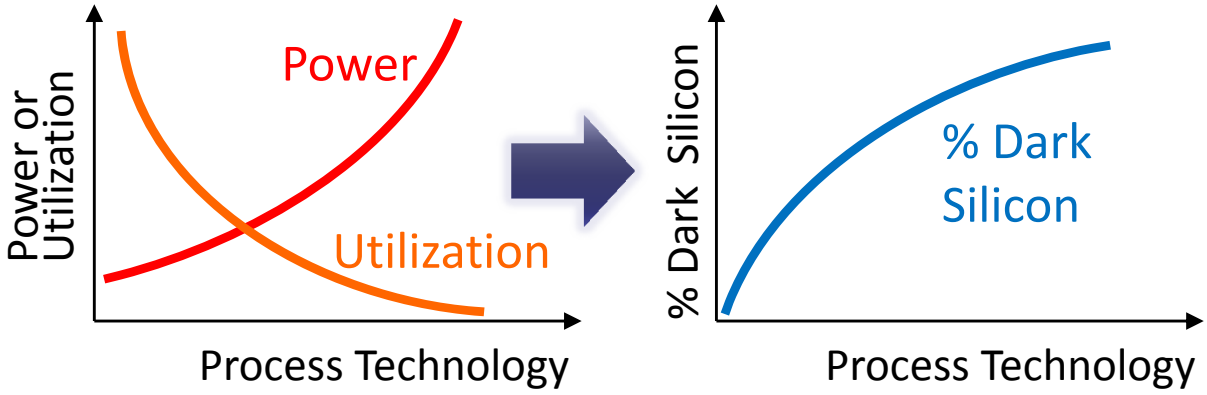
参考：[Taylor2013]

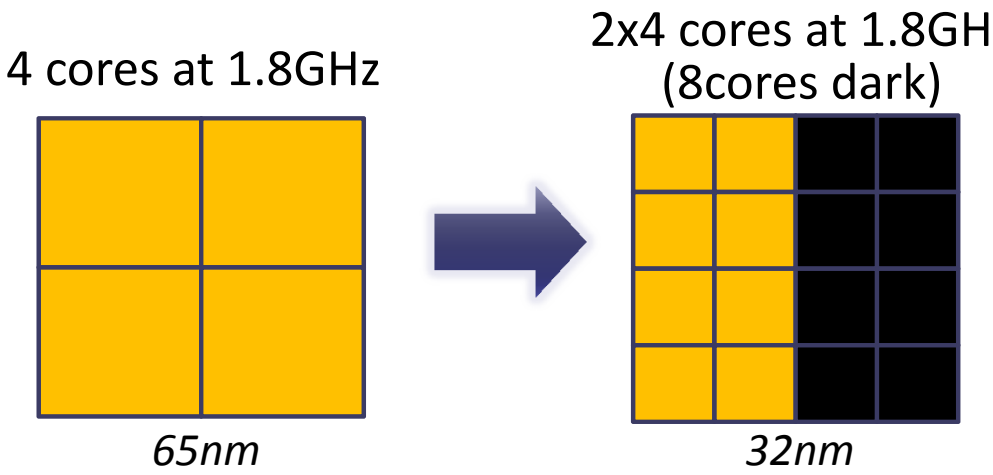
- 将来的にチップ上の領域をフルに稼働させることが不可に
  - 部分的に稼働/周波数を下げて実行、あるいはキャッシュへつぎ込む

→ Multicoreスケーリングも難しくなる（Utilization Wall）

Post-Dennard Scaling

Transistor Property	Dennard	Post-Dennard
Δ Quantity	S^2	S^2
Δ Frequency	S	S
Δ Capacity	$1/S$	$1/S$
Δ V^2	$1/S^2$	1
Δ Power	1	S^2
Δ Utilization (1/Power)	1	$1/S^2$

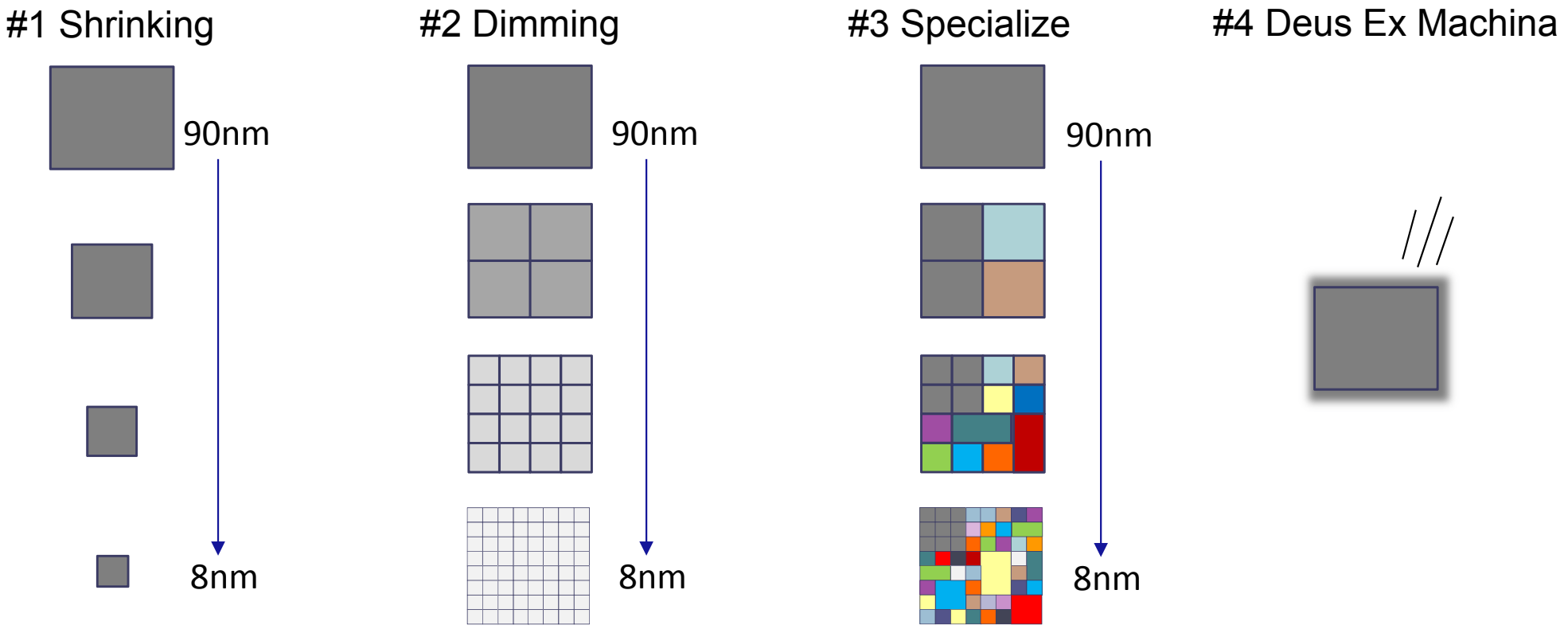




Multicore Scalingの限界

Dark Silicon問題の対処

- Four horsemen　参考：[Taylor2012]



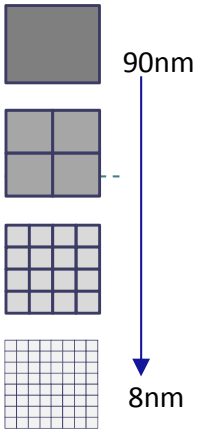
- 単に小面積チップを作る
- ☹チップ単価下落
- ☹I/O Padはスケールしない
- ☹発熱密度の増加

- 低性能コア・低周波数利用
- Near Threshold Voltage
- 空間的: Manycore, SIMD
- 時間的: Intel Turbo Boost, ARM big.LITTLE

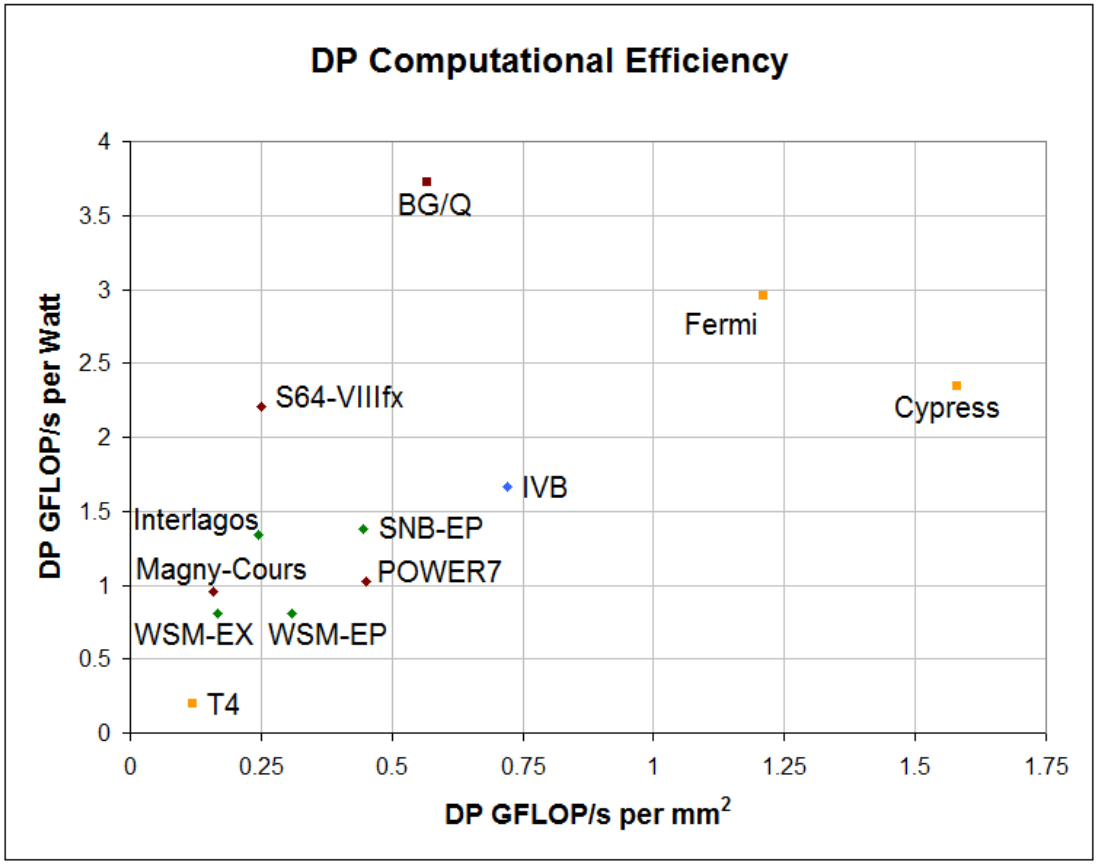
- 機能特化のコアを利用
- チップ面積より電力が重要
- ☺専用回路は汎用プロセッサよりも10-1000x 高効率

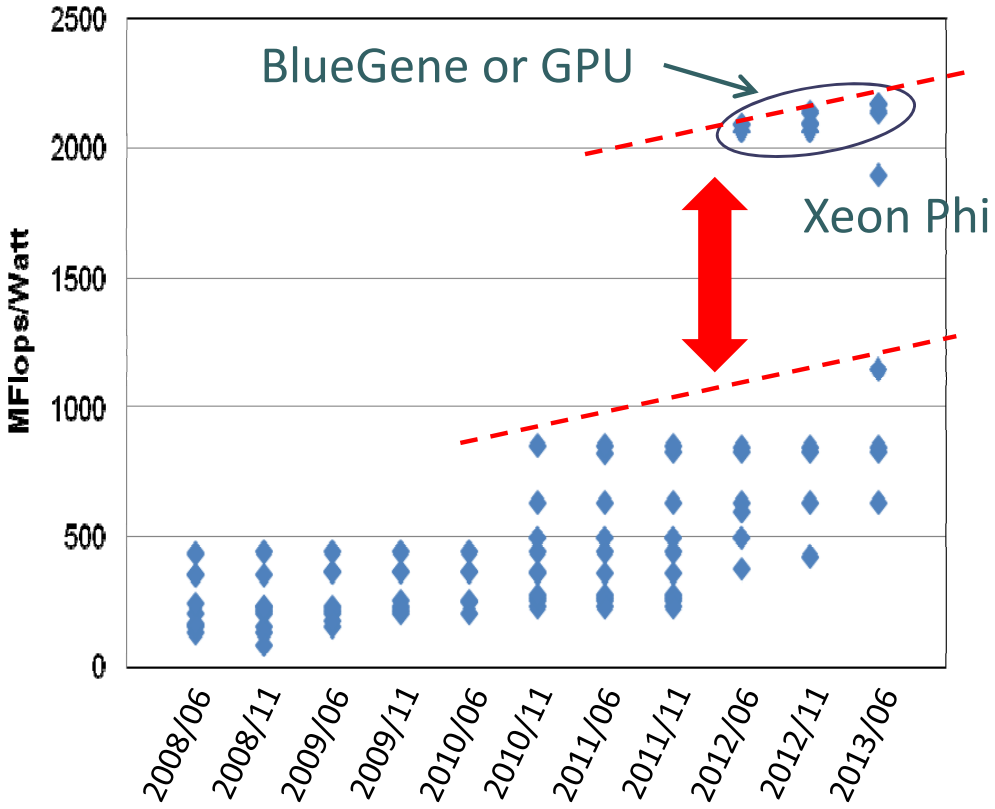
- 破壊的技術革新を待つ
- 新デバイス：nanotubes SFQ, TFET, photonics, …
- 新計算方式：Quantum, 脳型コンピュータ, …

メニーコア化



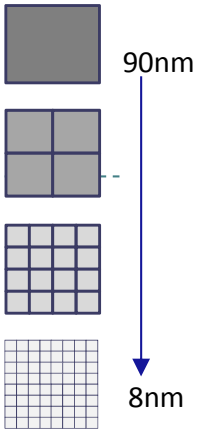
- メニーコアプロセッサによる高エネルギー効率化
  - シンプルで比較的高速でないプロセッサコアを多数用いる
  - 例) GPU、Xeon Phi、PEZY-SC、・・・





出典: [Taylor2013]

プロセッサ・コア内のSIMD幅拡大

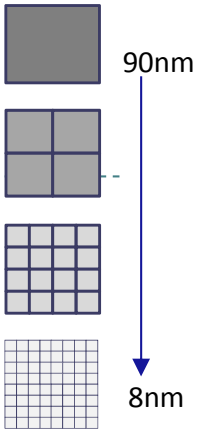


- SIMD幅を拡大することで電力性能効率が向上
- 多くのプロセッサがSIMD幅を増加させる傾向
  - e.g.) Intel SSE(128bit) → AVX(256bit) → AVX-512(512bit)
- 高い実効性能を出すための ソフトウェア環境が重要に

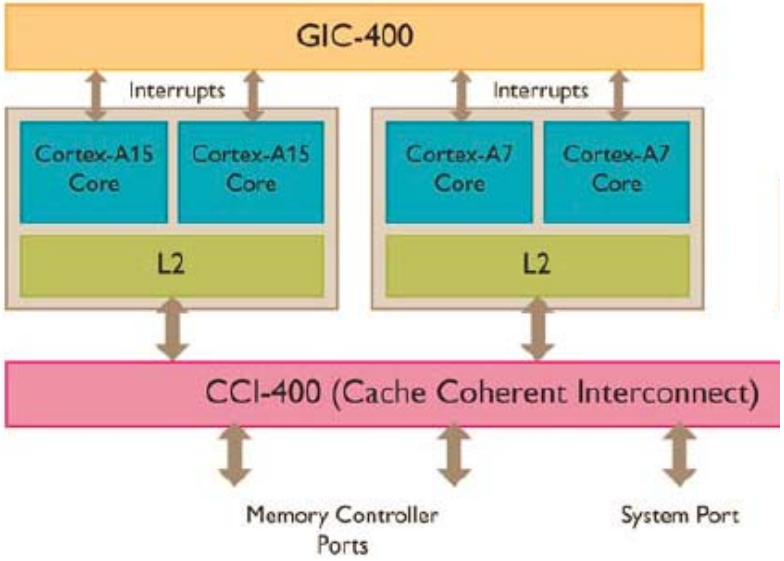
SPARC64 IXfxコアをSIMD拡張したときの特性

出典: [追永2013]

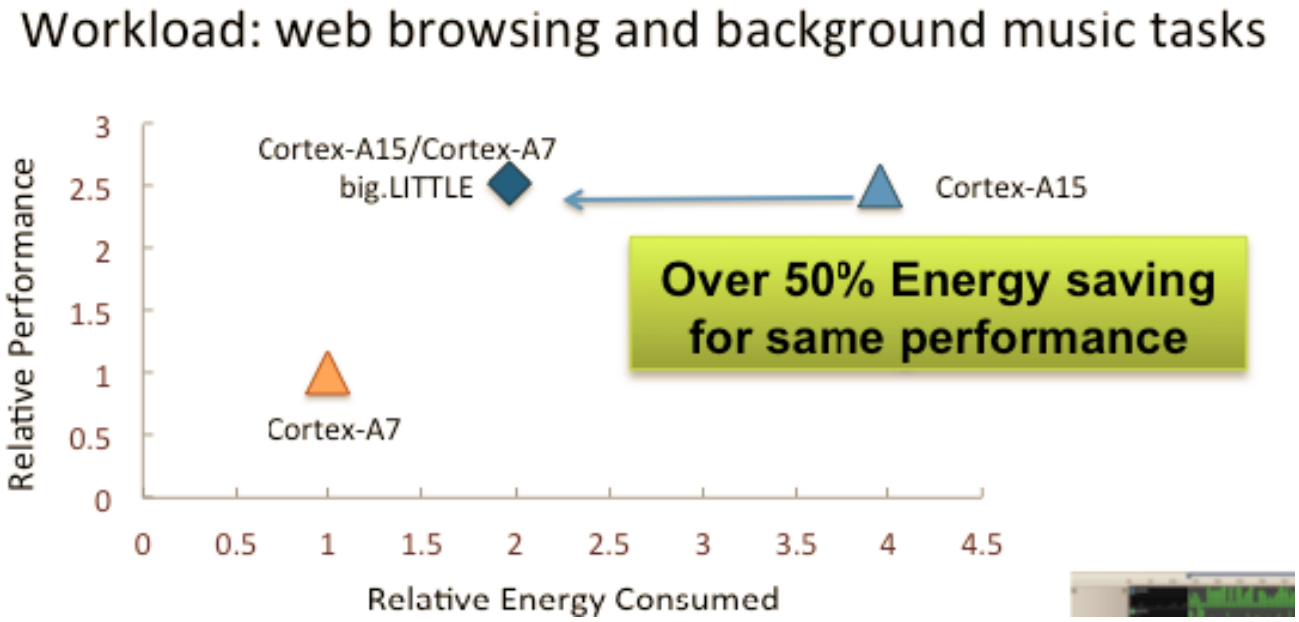
ヘテロジニアスマルチコアプロセッサ



- 高性能と低消費電力化を両立
- 例) ARM big.LITTLEアーキテクチャ
  - 高性能コア(Coretex-A15)と省電力コア(Cortex-A7)を組み合わせ
  - 負荷に応じてタスクを振り分け／マイグレーション

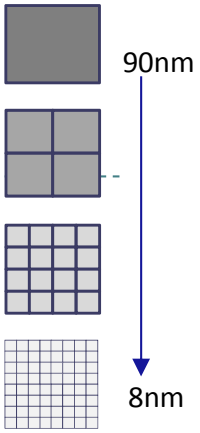


出典: [Jeff2012]

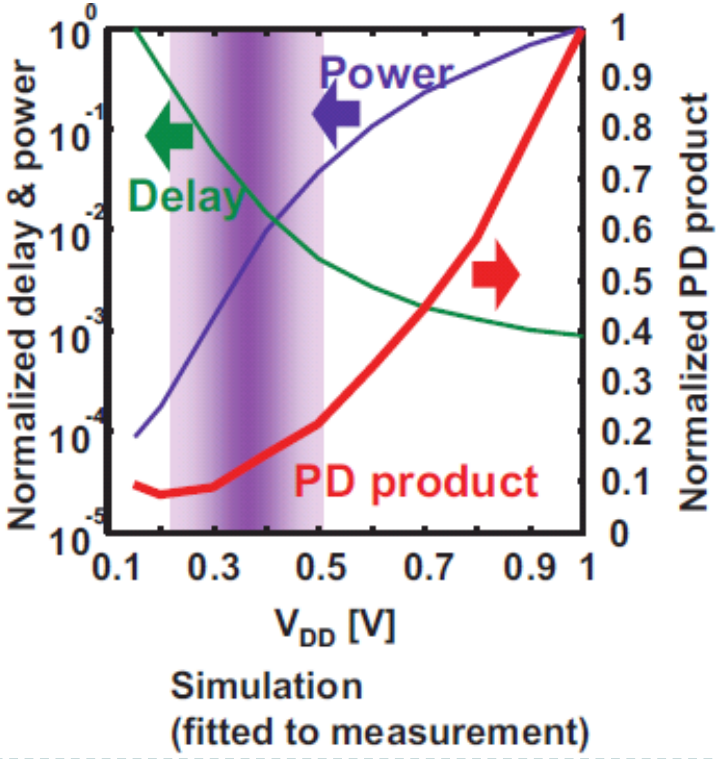


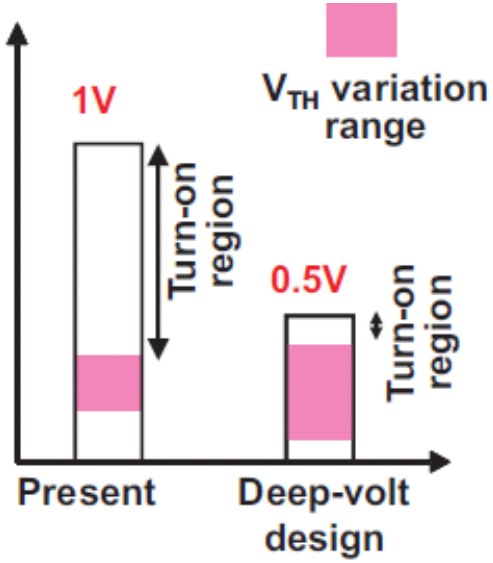
出典: [ARM2013]

Low (Near-Threshold) Voltage Computing



- Near-Threshold Voltage Computing (NTV or NTC)
  - 閾値電圧近くの低電圧領域で回路を動作
  - 最大クロック周波数は悪化 → コアあたりの性能は低下
  - エネルギー効率は最大になる
- 低電圧動作の利点と課題

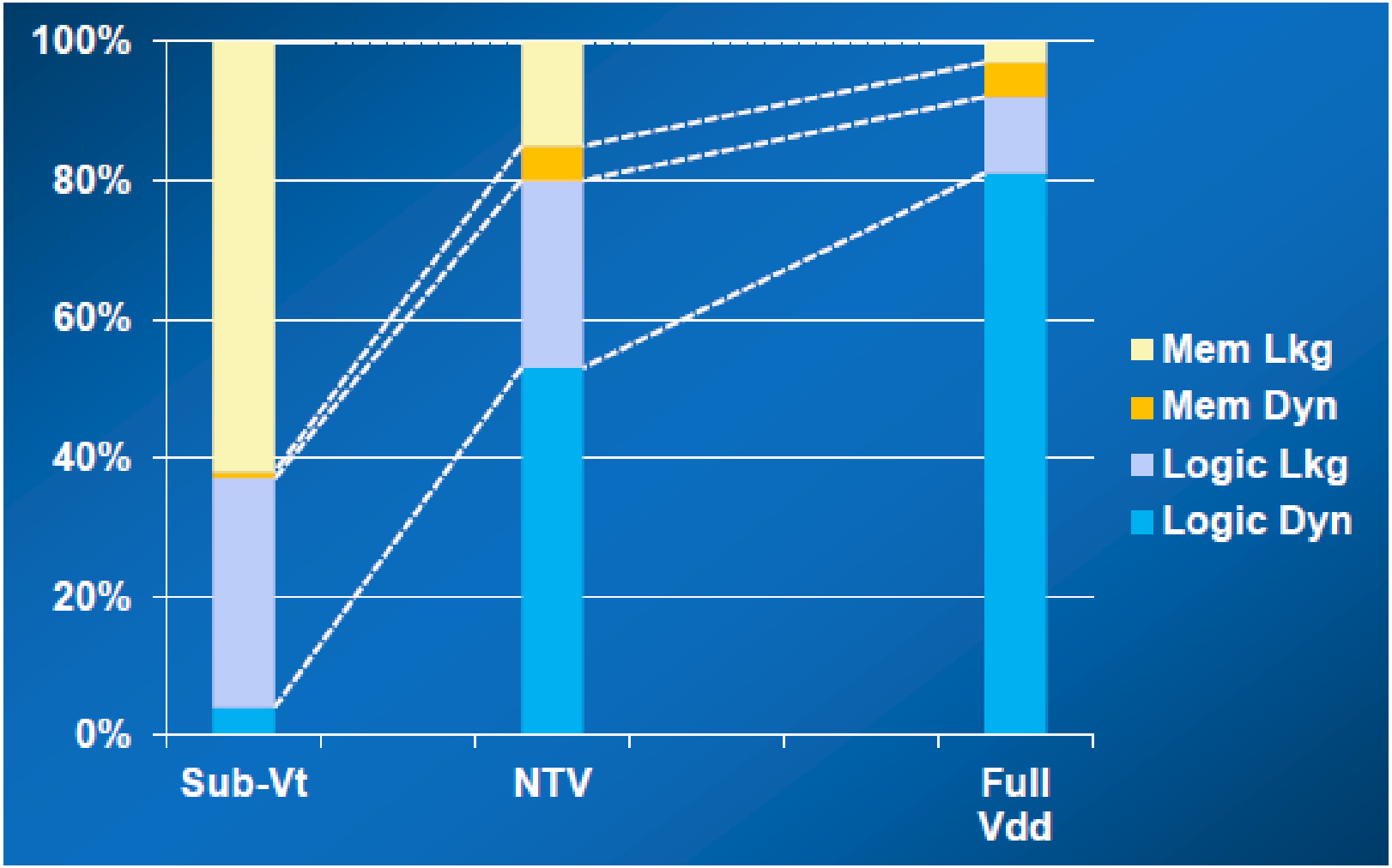




出典: [Sakurai2011]

NTCの電力内訳の変化

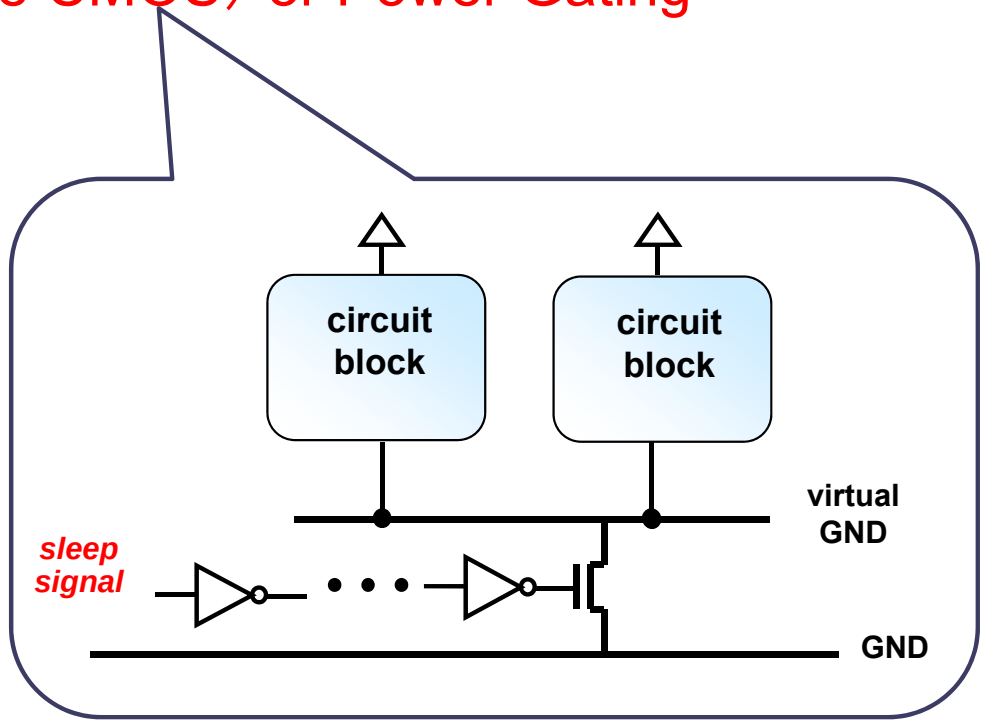
- NTV領域での動作ではリーク電力が支配的に



出典：[Borkar2013]

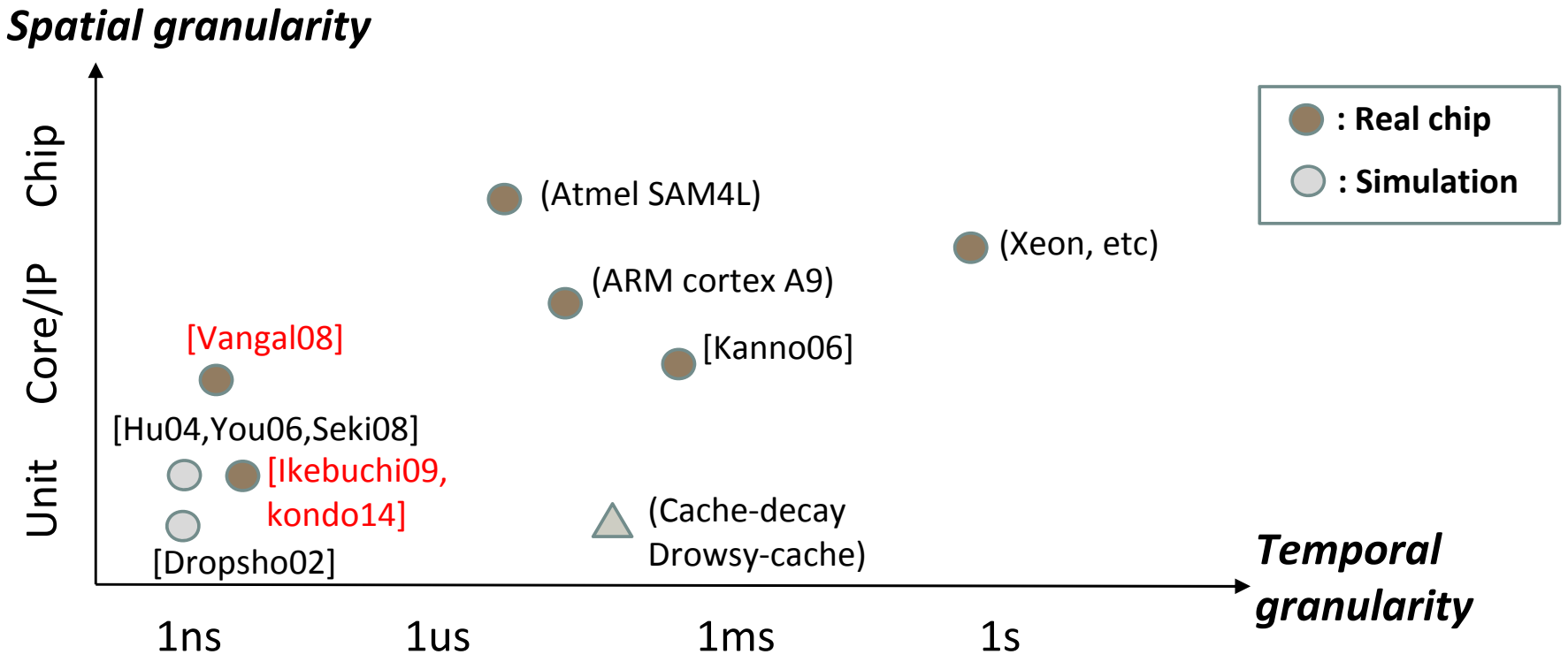
代表的なリーク電力削減技術(再掲)

- 閾値電圧の変更(高閾値電圧化)
  - Dual-Vth(Dual Threshold Voltage)
  - VTCMOS(Variable-Threshold-voltage CMOS)
- 電源電圧の変更(電源供給の停止または低電源電圧化)
  - MTCMOS(Multi-Threshold-voltage CMOS) or Power Gating
  - MSV(Multi-Supply Voltage)
  - DVS(Dynamic Voltage Scaling)
- 入力データの設定
  - IVC(Input Vector Control)
- デバイス/プロセス技術の向上
  - Multi-gateトランジスタ
- 不揮発性メモリの利用



細粒度Power-Gating技術

- Power-Gatingに関する種々の実装



[Dropsho02] S. Dropsho et al., "Managing Static Leakage Energy in Microprocessor Functional Units", *Proc. MICRO*, 2002.
[Hu04] Z. Hu et al., "Microarchitectural Techniques for Power Gating of Execution Units", *Proc. ISLPED, 2004.*
[You06] Y.-P. You, C. Lee, and J. Lee, "Compilers for Leakage Power Reduction", *ACM TODAES*, 2006.
[Seki08] N. Seki et al., "A Fine Grain Dynamic Sleep Control Scheme in MIPS R3000", *Proc. ICCD*, 2008.
[Ikebuchi09] D. Ikebuchi *et al.*, .Geyser-1: A Mips R3000 CPU core with Fine grain Runtime Power Gating,. *Proc. ASSCC2009*, pp.281-284, 2009.
[Kanno06] Y. Kanno et al., "Hierarchical Power Distribution with 20 Power Domains in 90-nm Low-Power Multi-CPU Processor", *ISSCC*, 2006.
[Vangal08] S.R. Vangal, "An 80-Tile Sub-100-W TeraFLOPS Processor in 65-nm CMOS", JSSC, Jan. 2008.
[kondo14] M.Kondo et al., "Design and Evaluation of Fine-Grained Power-Gating for Embedded Microprocessors", DATE2014, March 2014.

細粒度なStandbyモードの例

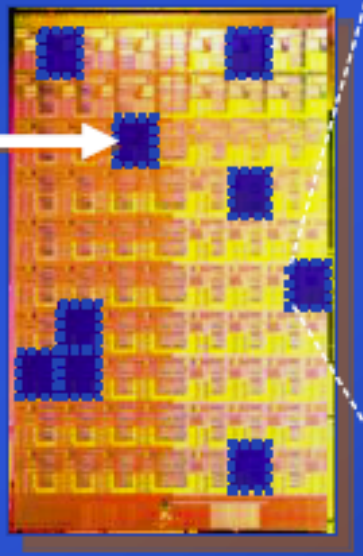
- Intel 80-Tileプロセッサの細粒度電力モード制御

Mode		Power Saving	Wake up
Normal	All active	-	-
Standby	Logic off Memory on	50%	Fast
Sleep	Logic and Memory off	80%	Slow

Dynamic Chip Level

STANDBY:
- Memory retains data
- **50%** less power/tile

FULL SLEEP:
- Memories fully off
- **80%** less power/tile



Dynamic, within a core
21 sleep regions per tile

Data Memory
Sleeping:
57% less power

Instruction Memory
Sleeping:
56% less power

Router
Router
Sleeping:
10% less power
(stays on to pass traffic)

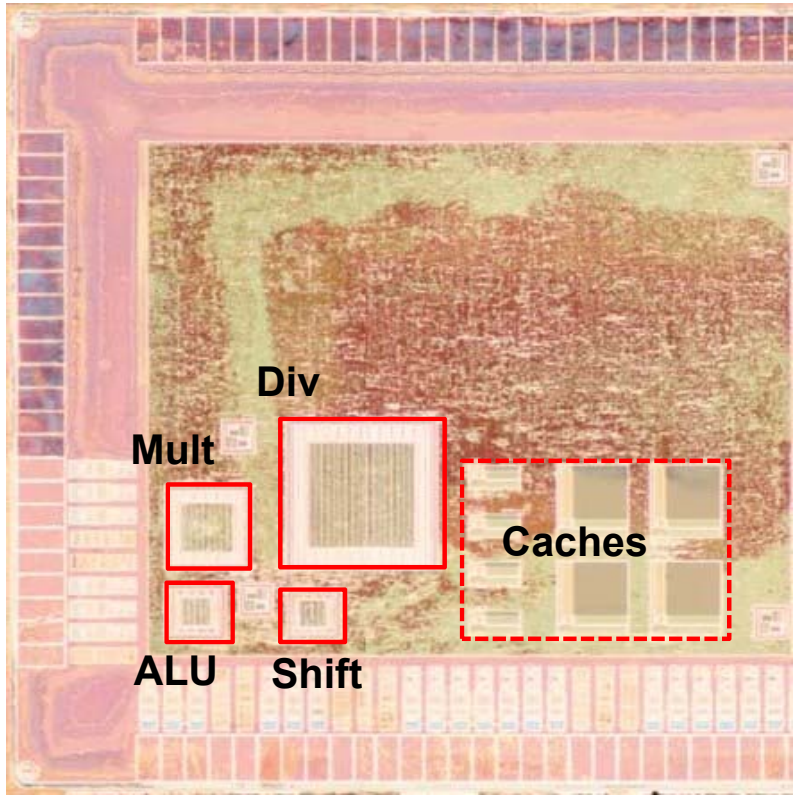
FP Engine 1
Sleeping:
90% less power

FP Engine 2
Sleeping:
90% less power

出典：[Borkar2013]

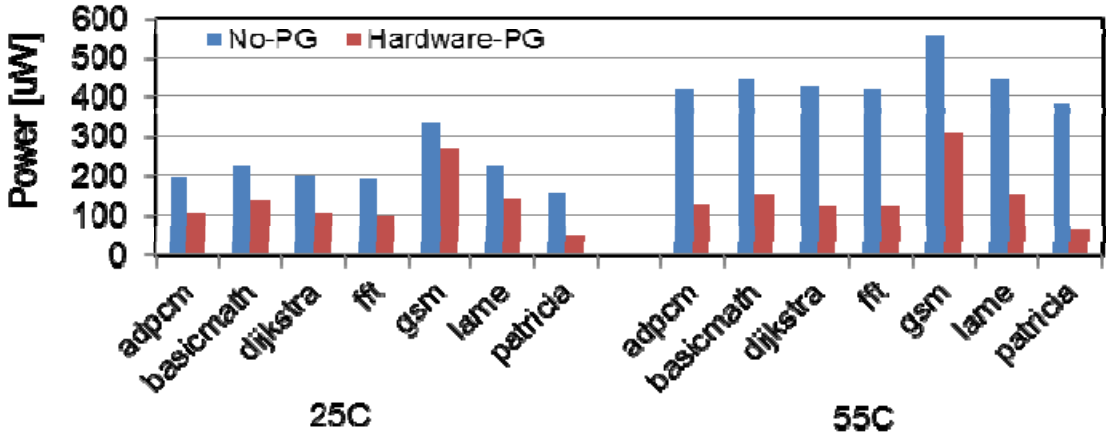
走行時細粒度PGを実装したプロセッサGeyser

- Geyser (MIPS R3000互換、65nm CMOSプロセスで実装)
  - 演算器単位の走行時Power-Gatingを実装、HWによるスリープ制御
  - 省エネ化の損益分岐時間を解析しコンパイラからスリープ制御も可能

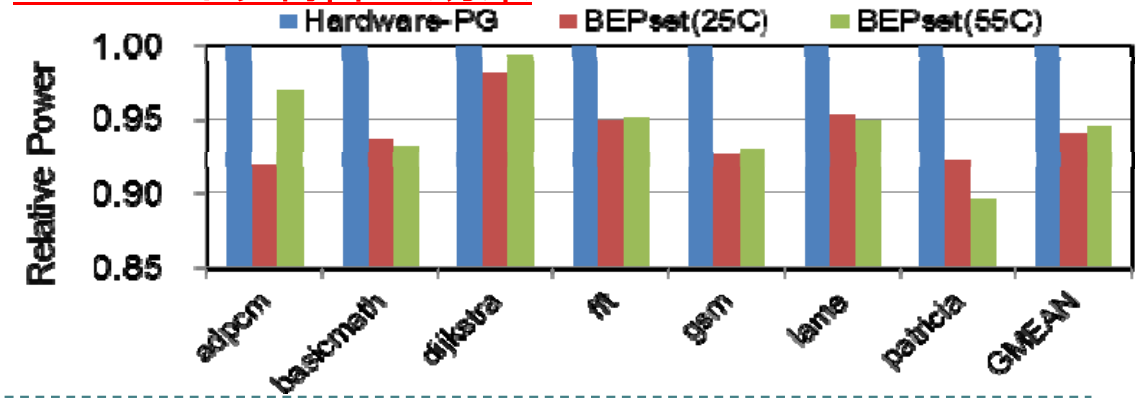


出典：[Kondo2014]

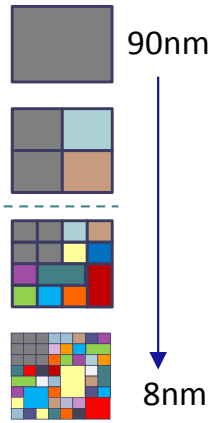
・細粒度PGの評価結果



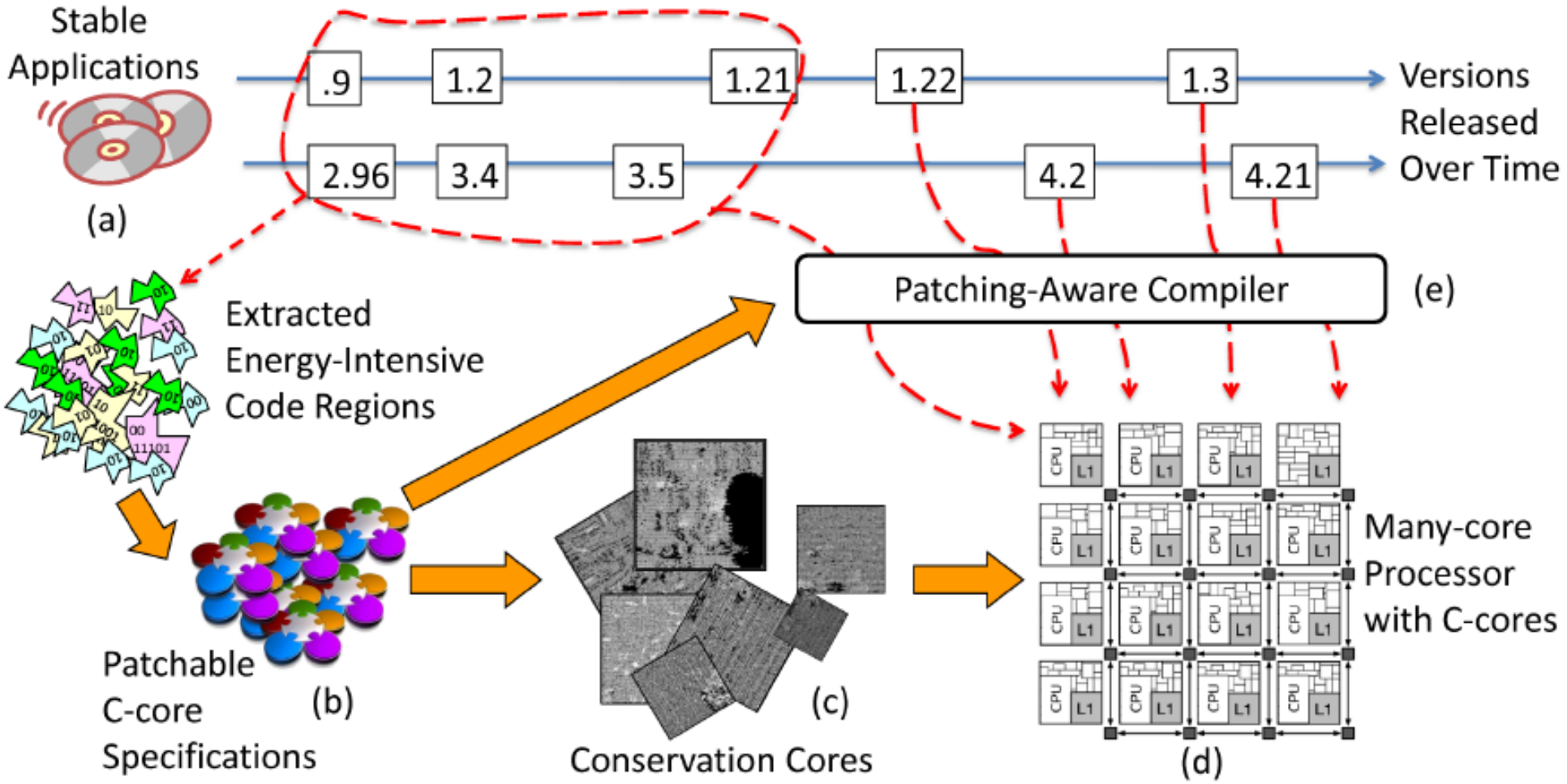
・コンパイラ制御の効果



機能特化型(専用)コア利用の例 1/2



- Conservation cores (C-cores)
  - プロファイリングから機能特化の回路をコアに合成

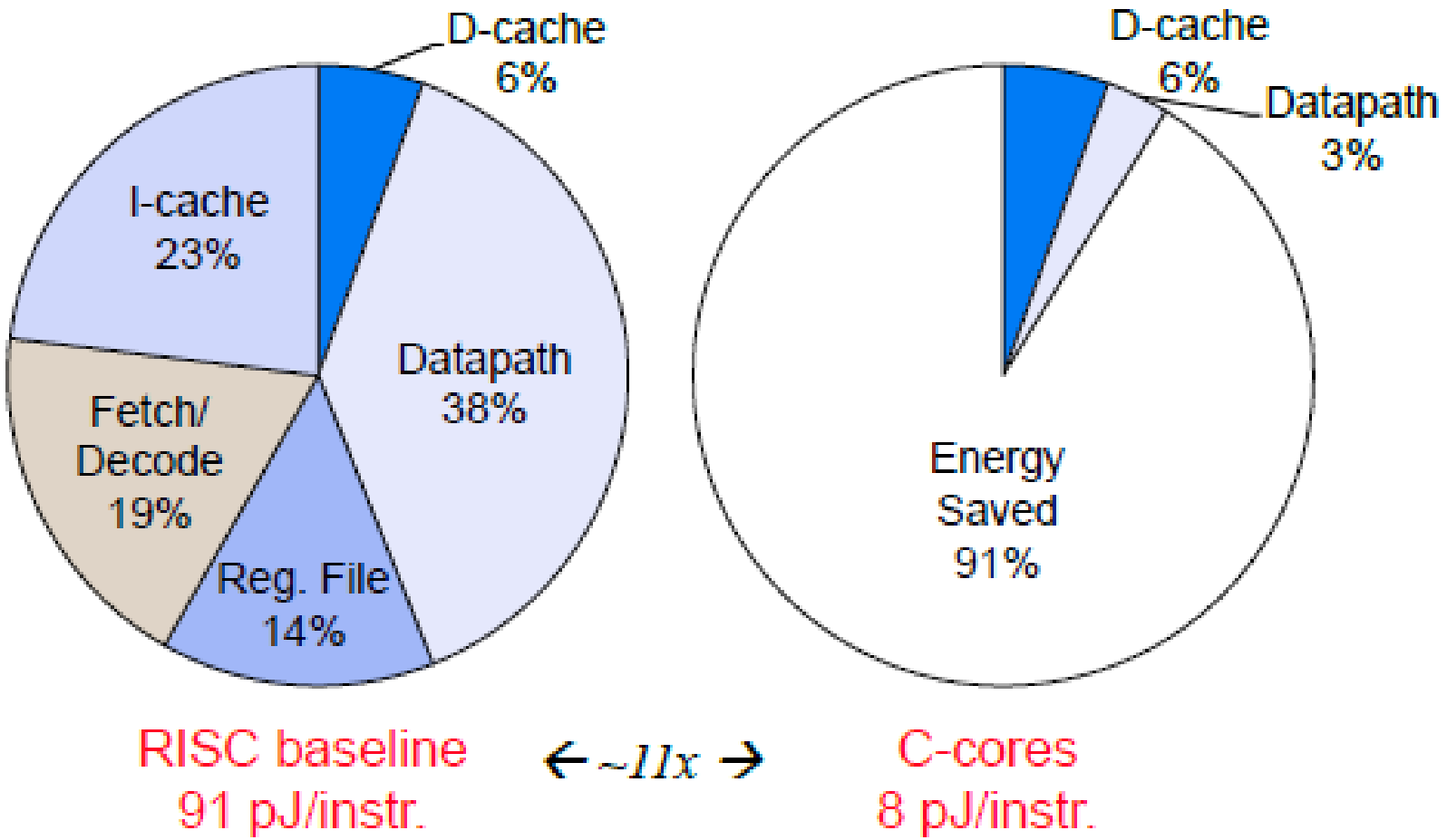


出典: [Venkatesh2010]

機能特化型(専用)コア利用の例 2/2

- Conservation cores (C-cores)のエネルギー効率

出典: [Taylor2012]



FPGAの利用

- 多くの分野で高処理効率のためFPGAを採用する流れ
  - Intel：データセンター向けとしてXeonにFPGA搭載を検討
  - Microsoft：検索処理の効率化のためブレードにFPGAを統合

Microsoft Catapult

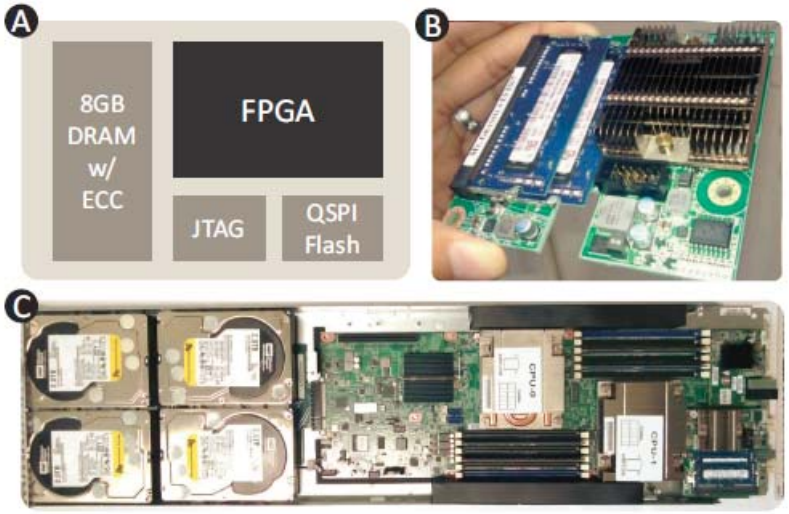
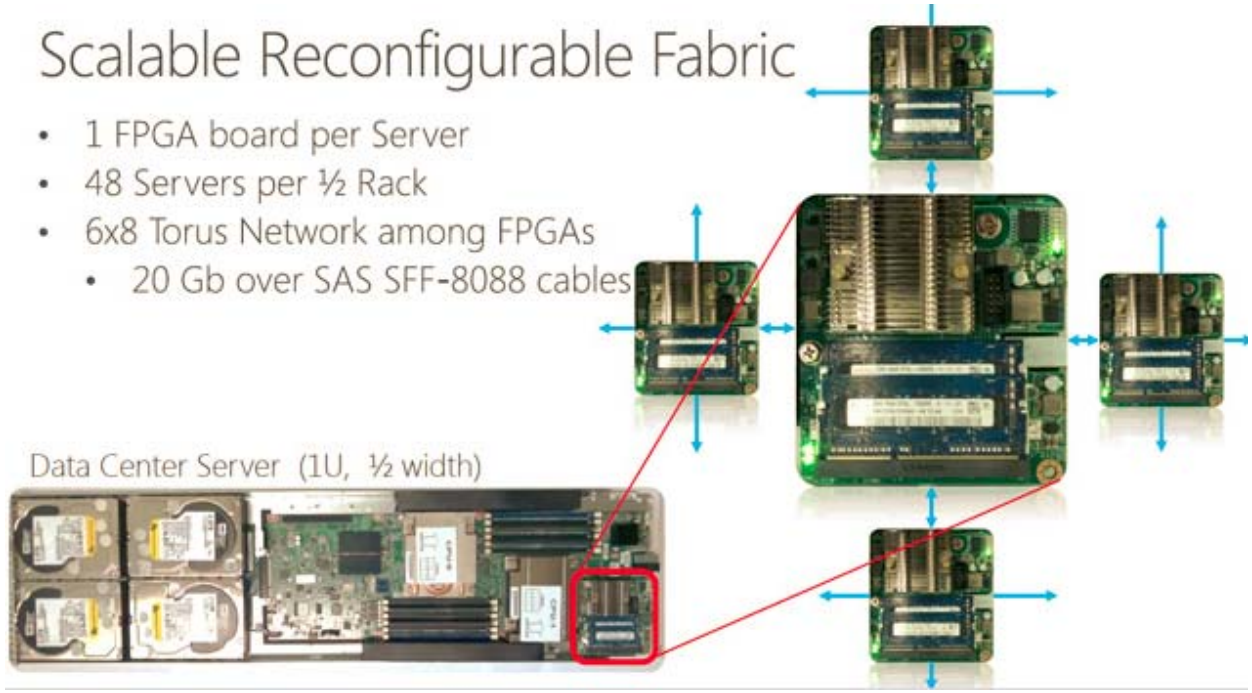


Figure 1: (a) A block diagram of the FPGA board. (b) A picture of the manufactured board. (c) A diagram of the 1 U, half-width server that hosts the FPGA board. The air flows from the left to the right, leaving the FPGA in the exhaust of both CPUs.

出典：[Putnam-ISCA2014]



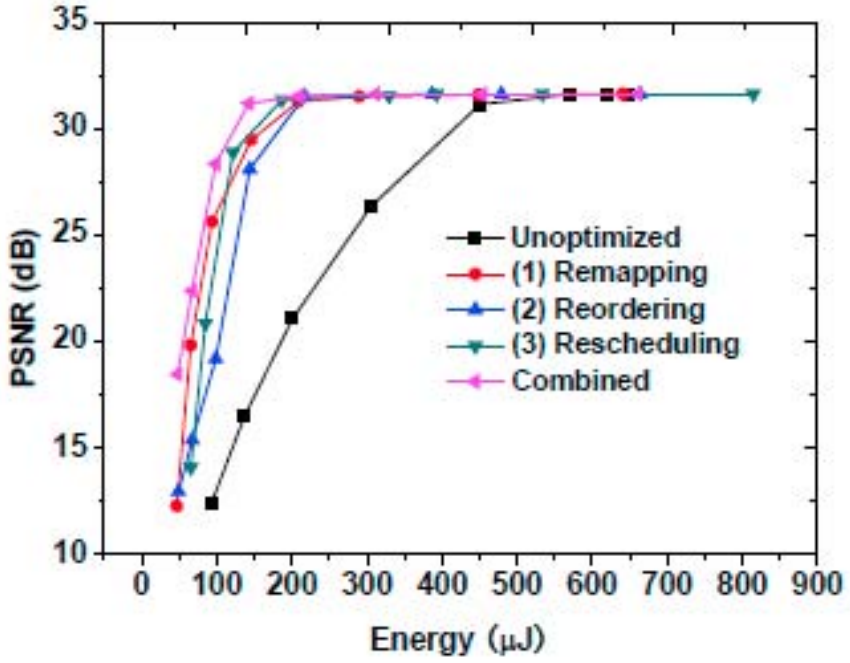
出典：[Putnam-HotChips2014]

Approximate Computing

- 多くの場合100%の計算精度は必要ではない
  - 信号処理、マルチメディア、検索、機械学習、データマイニング・・・
- 計算エラーをある程度許容することで電力削減が可能
  - 低電源電圧の利用、ECCの省略、など

画像のデコード（IDCT）への適用例

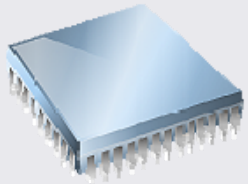
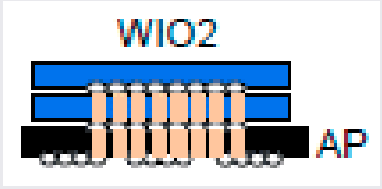
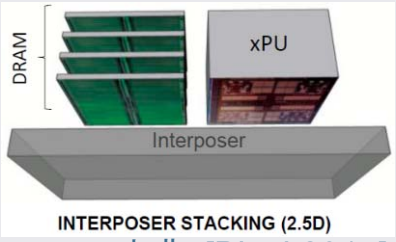
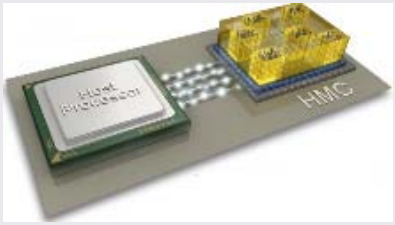
出典：[Han2013]



本チュートリアルの構成

- 高性能計算機の消費電力トレンド
- 計算機システムにおける電力消費の基礎
- 省電力・省エネ技術
  - Dark-silicon問題とプロセッサの省電力・省エネ化技術
  - メモリの省電力化技術
  - インターコネクションネットワークの省電力化技術
  - システムソフトウェアレベルでの電力制御技術
- 将来展望

メモリ技術の比較　参考：[Allan2014,Brennan2014]

Memory	LPDDR3/4	Wide IO2	HBM	HMC
DRAM Interface	Traditional Parallel Interface	Wide Parallel Interface	Wide Parallel multi-channel IF	Chip-to-Chip SERDES
System Interface	Point-to-point, DIMM	プロセッサ上に3次元積層	2.5D積層(Silicon interposer)	Point-to-Point SERDES
Interface Width	16,32,64	256,512	128/channel	4links(16-lane)
Max Bandwidth	34GBps	68GBps	256GBps	240GBps
Size	0.5-4GB	1-2GB	1-8GB/Cube	2-4GB/Cube
JEDEC Standard	Yes	Yes	Yes	No
Application	Mobile	High end Smartphone	Graphics, HPC	High end server, HPC
		 出典：[Brennan2014]	 出典：[Black2014]	 出典：[Baxter2014]

HMCを用いたシステムの実例

- SPARC64 XIfx
  - 富士通株式会社が開発したHPC計算機向けCPU
  - PRIMEHPC FX100へ搭載

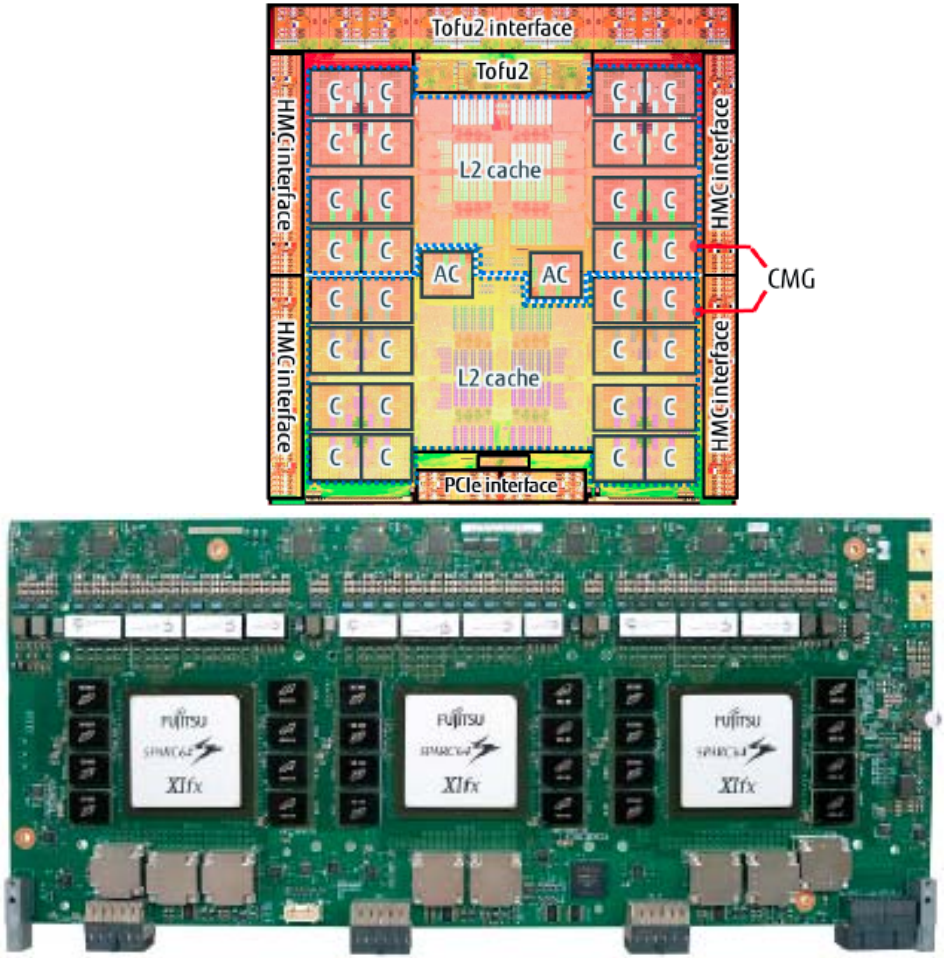


Table 2 SPARC64™ XIfx specifications	
Number of cores	32 + 2
Number of threads per core	1
L2 cache capacity	24 MiB
Peak performance	> 1 Tflops
Theoretical memory bandwidth	240 GB/s x 2 (in/out)
Theoretical interconnect bandwidth	125 GB/s x 2 (in/out)
Process technology	20 nm CMOS
Number of transistors	3.75 billion
Number of signal pins	1,001
HMC SerDes	128 lanes
Tofu2 SerDes	40 lanes
PCIe Gen3 SerDes	16 lanes

出典：[SPARC64XIfx2014]

メモリ(DRAM)の消費電力

- 各世代のメモリの比較

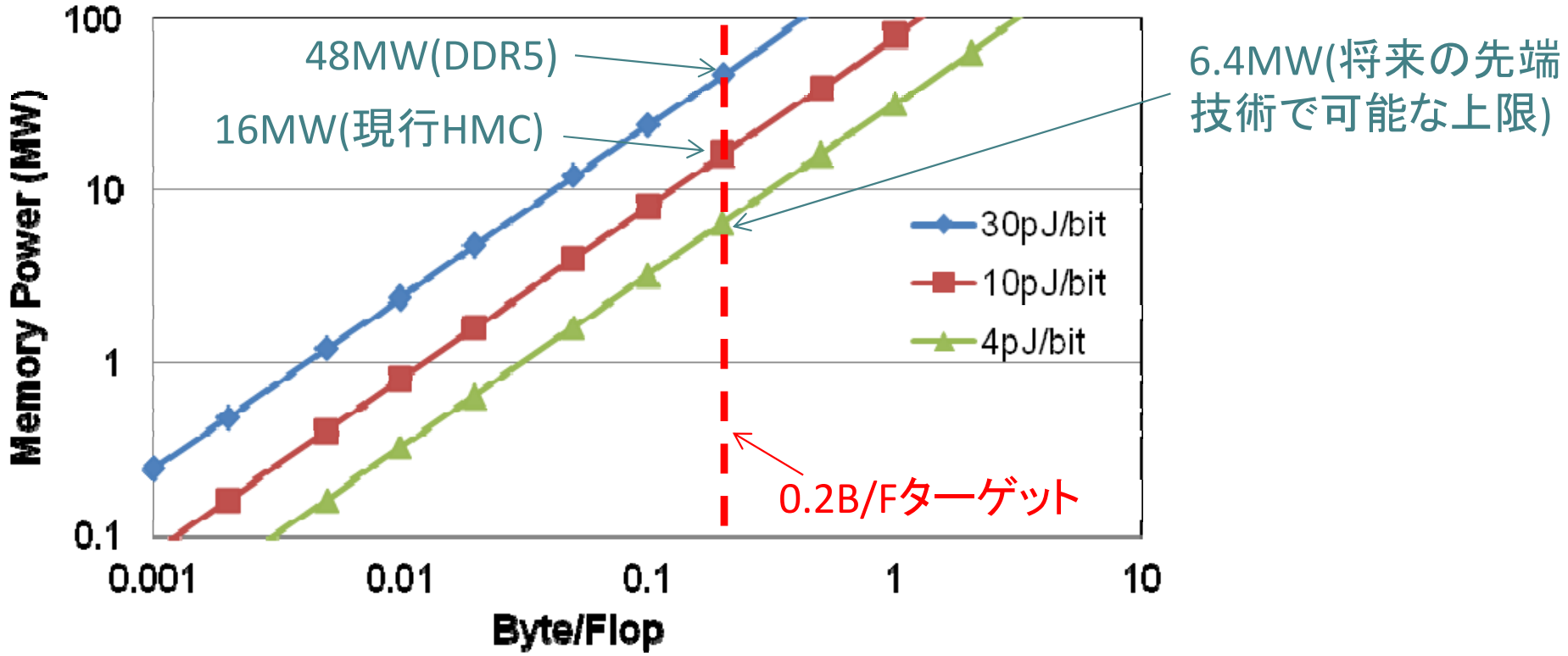
Technology	VDD	IDD	BW GB/s	Power (W)	mw/GB/s	pJ/bit	real pJ/bit
SDRAM PC133 1GB	3.3	1.50	1.06	4.96	4664.97	583.12	762
DDR-333 1GB	2.5	2.19	2.66	5.48	2057.06	257.13	245
DDRII-667 2GB	1.8	2.88	5.34	5.18	971.51	121.44	139
DDR3-1333 2GB	1.5	3.68	10.66	5.52	517.63	64.70	52
DDR4-2667 4GB	1.2	5.50	21.34	6.60	309.34	38.67	39
HMC 4DRAM w/ Logic	1.2	9.23	128.00	11.08	86.53	10.82	13.7

出典：[Pawlowski2011]

- 世代が進むにつれてDRAMの電力効率(pJ/bit)は改善
- DRAMモジュールの消費電力は増加傾向
  - 実装密度の向上、バンド幅の向上
- 従来トレンドを外挿すると2018年のDDRテクノロジ (DDR5)では30pJ/bitと予想されている

メモリ(DRAM)の消費電力

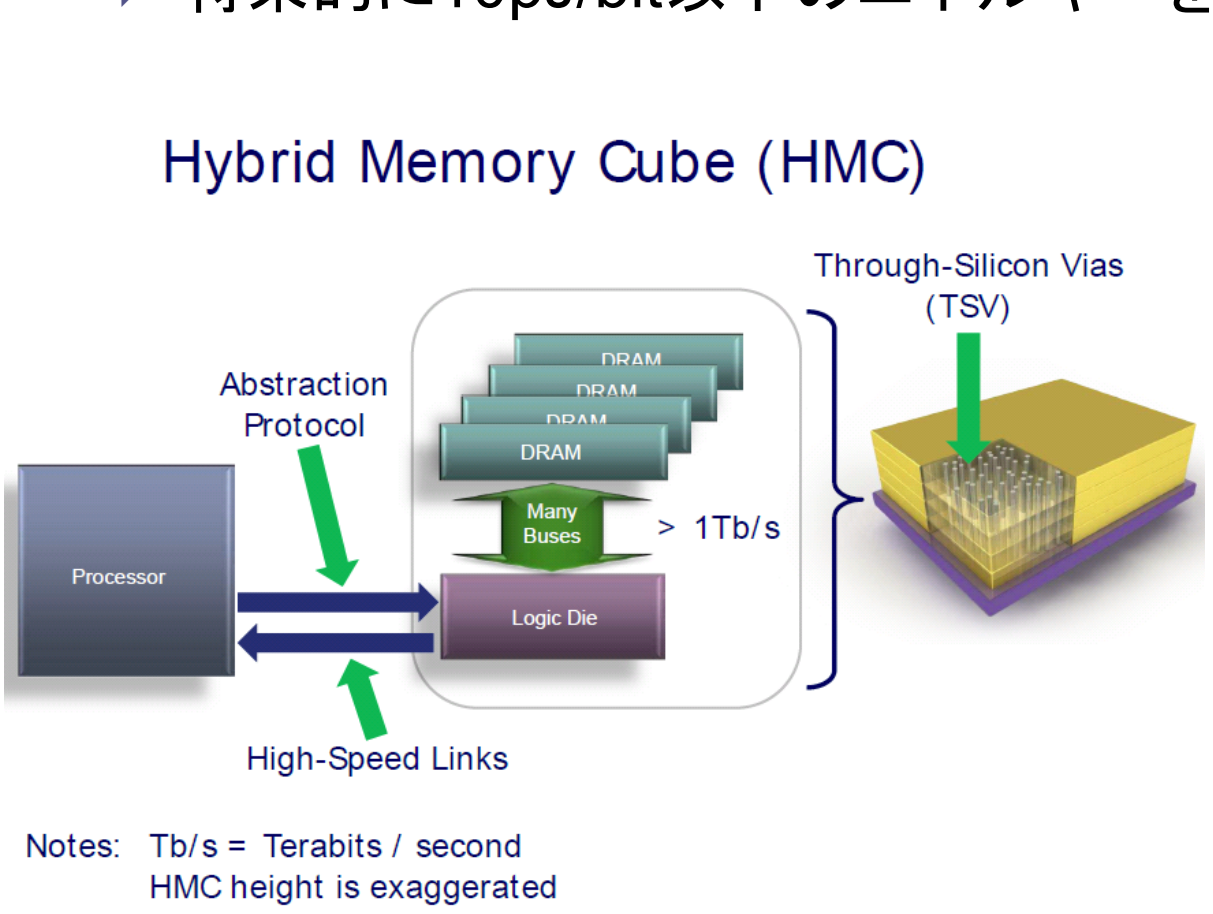
- 要求メモリバンド幅(Byte/Flop)に基づくDRAM電力の推定



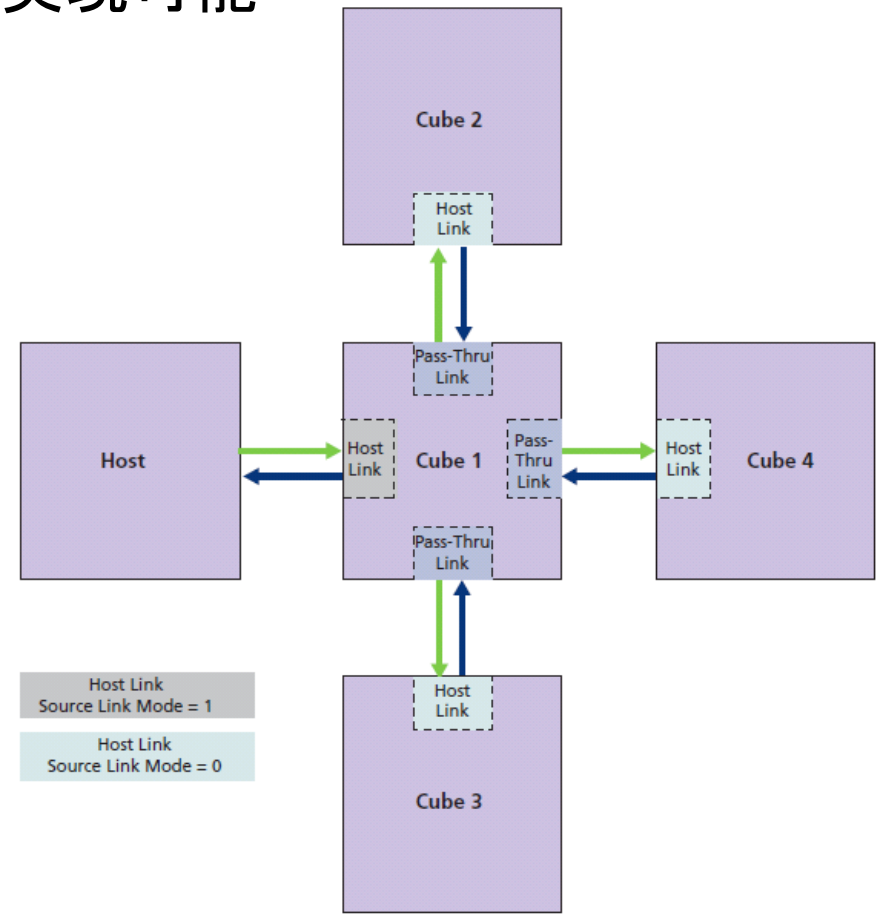
- 30pJ/bitでは0.2B/Fの実現に48MWもの電力が必要
- より電力効率の良いメモリ技術の開発が必須
  - 先端テクノロジで7pJ/bit（最小で4pJ/bit）と予想[Stevens2009]

Hybrid Memory Cube (HMC)

- DRAMダイとロジックダイを3次元積層
- プロセッサ・HMC間、HMCモジュール間を高速なシリアルリンクで接続
- 将来的に10pJ/bit以下のエネルギーを実現可能



出典：[Pawlowski2011]

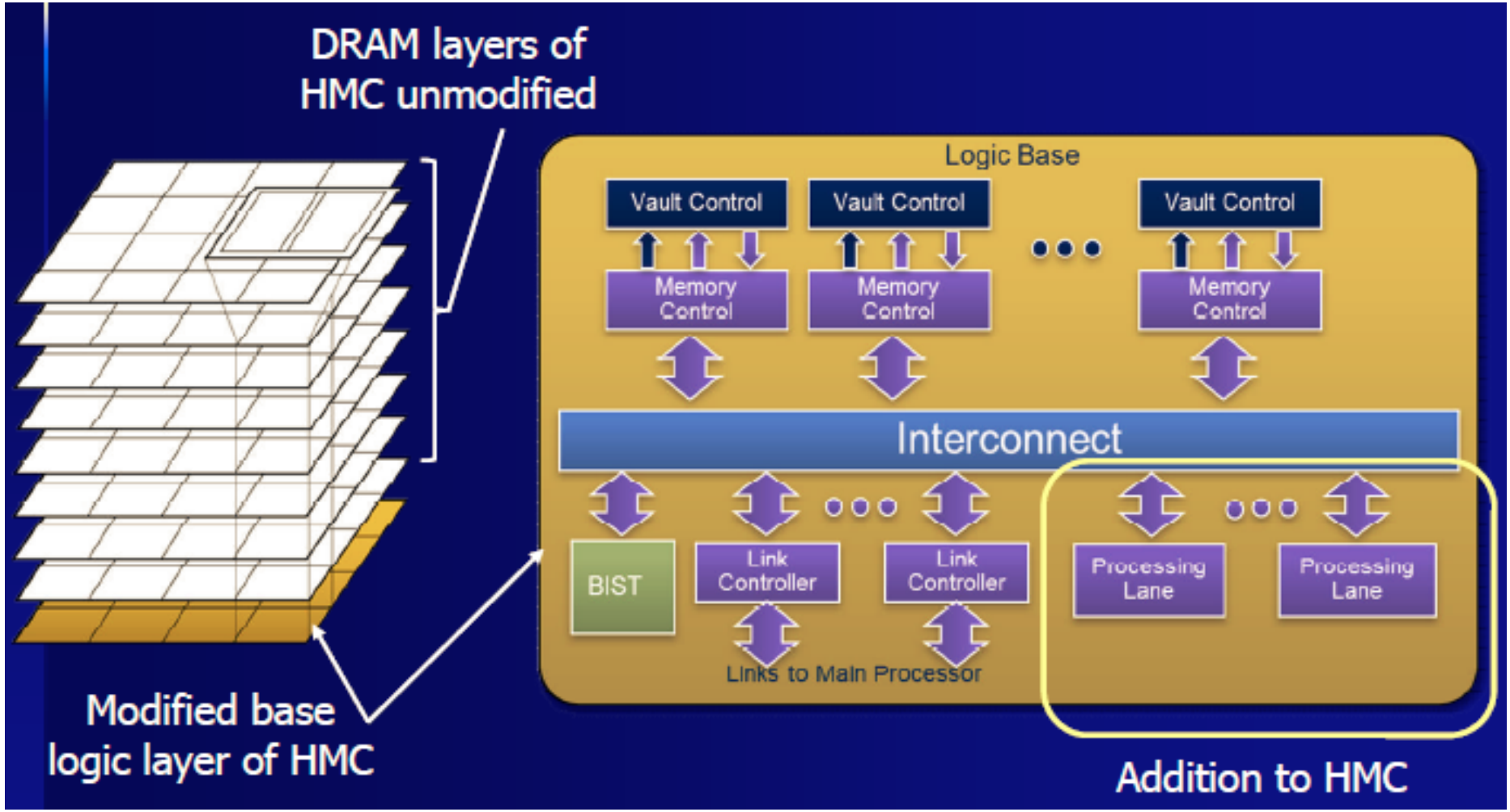


出典：[HMC2013]

Near Data Processing (1/2)

- HMCのロジックレイヤでベクトル演算を行う
  - データ移動のコスト(レイテンシ、電力)を削減

出典: [Nair2014]

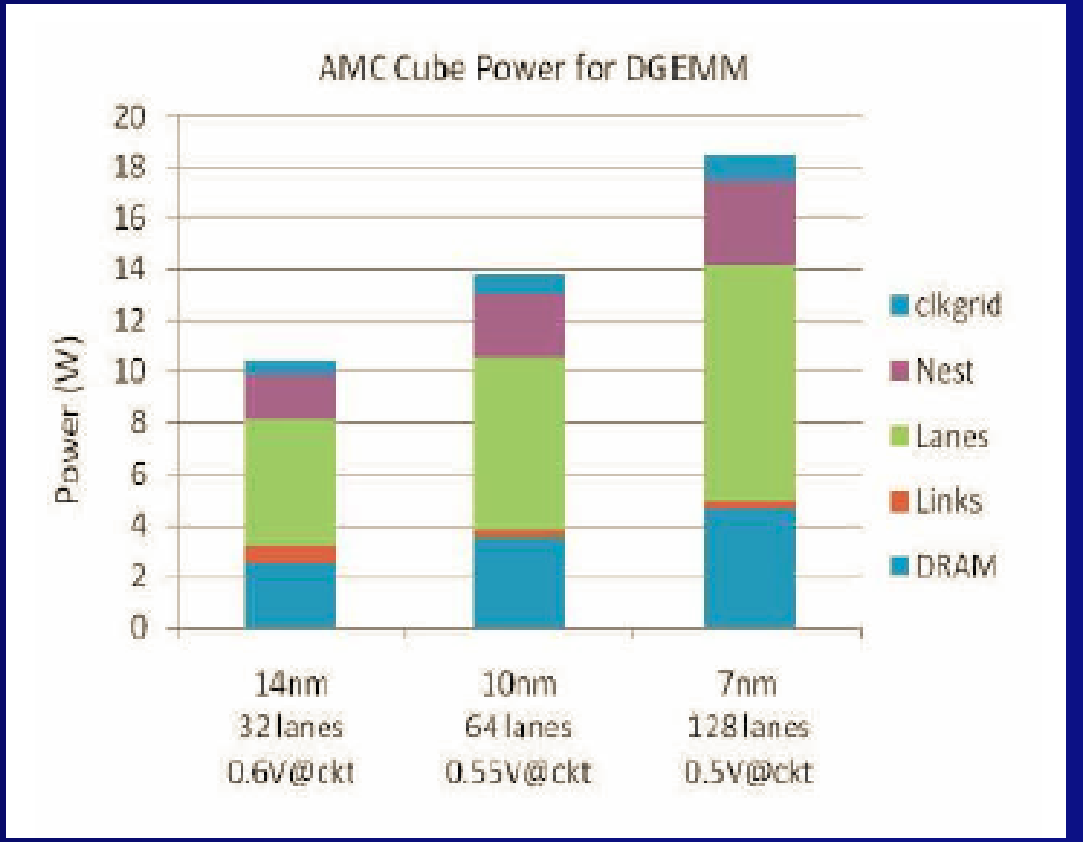


Near Data Processing (2/2)

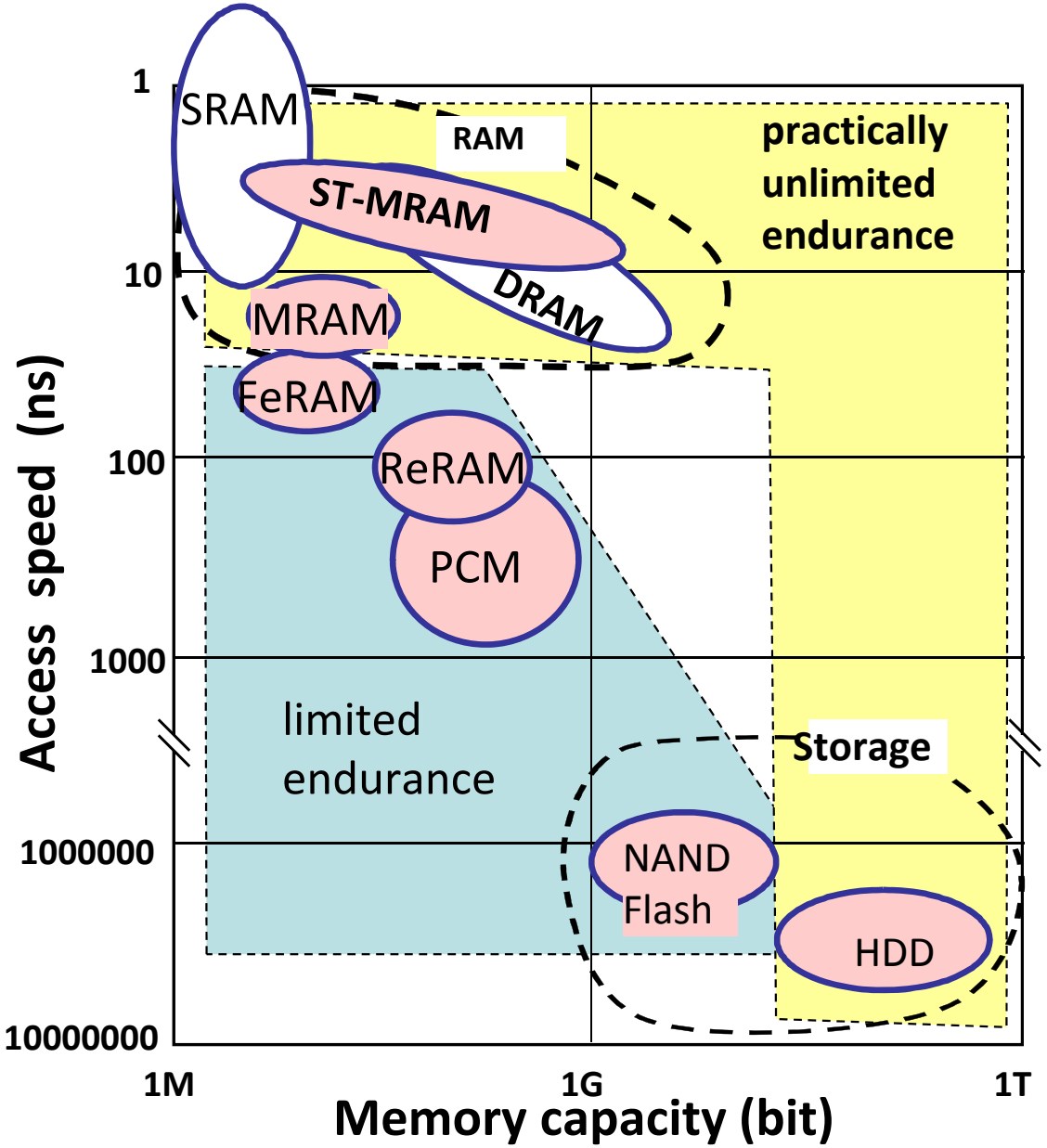
- AMC (Active Memory Cube)の電力あたり性能

出典: [Nair2014]

- DGEMM
  - **C = C – A x B**
  - 83% peak performance: 266 GF/AMC
    - Hand assembled
    - 77% through compiler
  - 10 W power in 14 nm technology
  - Roughly 20 GF/W at system level
  - 7 nm projection:
    - 56 GF/W at AMC
    - Roughly at target at system level



不揮発メモリ

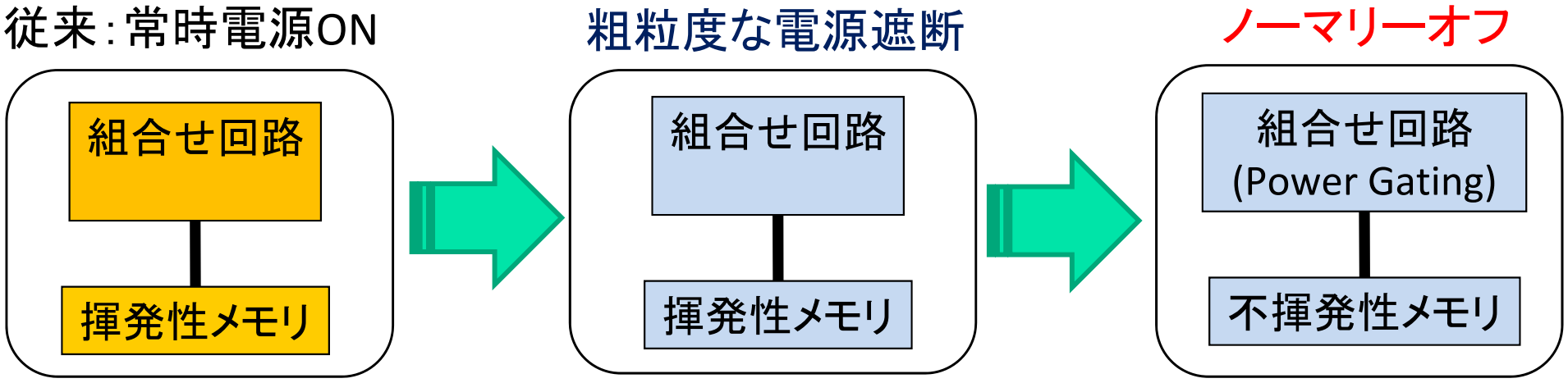


- 電荷ではなく抵抗値として情報を記憶
- リーク電力ゼロ（周辺回路除く）
- アクセス時間、集積度ともに揮発メモリに匹敵

参考: [Ando2012]

ノーマリーオフコンピューティング

- NEDO ノーマリーオフコンピューティング基盤技術開発
  - システムとしては動作中であっても真に動作すべき構成要素以外の電源を積極的に遮断する「ノーマリーオフ」を実現する「コンピューティング」
  - 不揮発性メモリ（電源遮断しても記憶を保持）＆ パワーゲーティング（電源遮断による低電力化）



☹リーク電力大

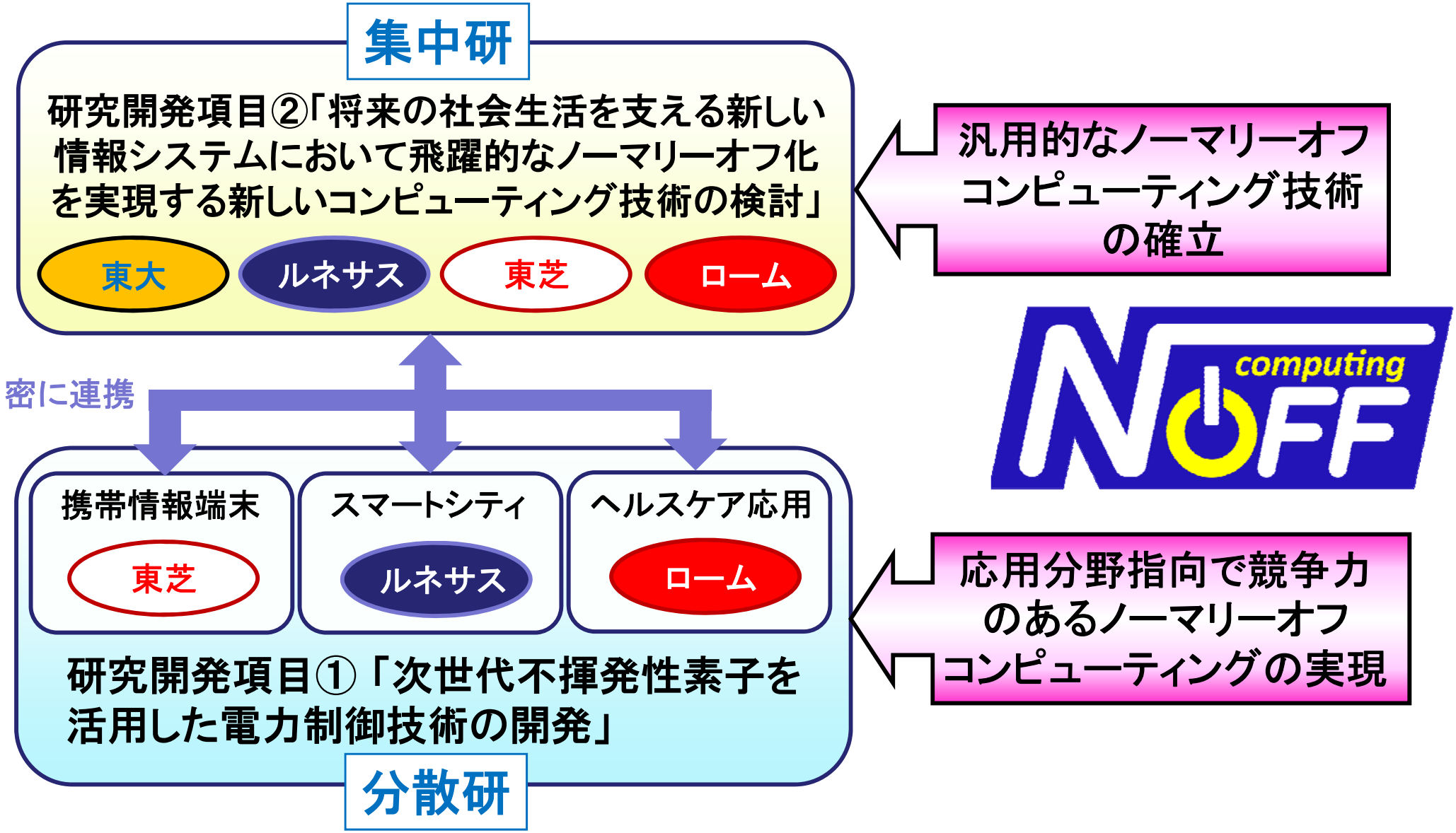
メモリ内容の退避が必要
☹データ保存に長い時間

☺時間的に細粒度な電源遮断

参考: [中村2014]

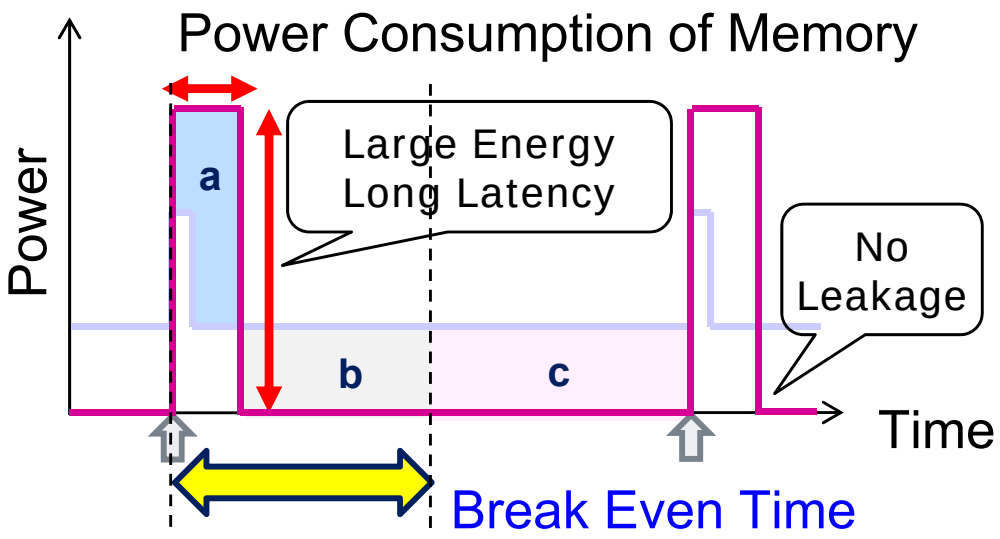
ノーマリーオフコンピューティングプロジェクト

- 実施体制図　出典: [中村2014]

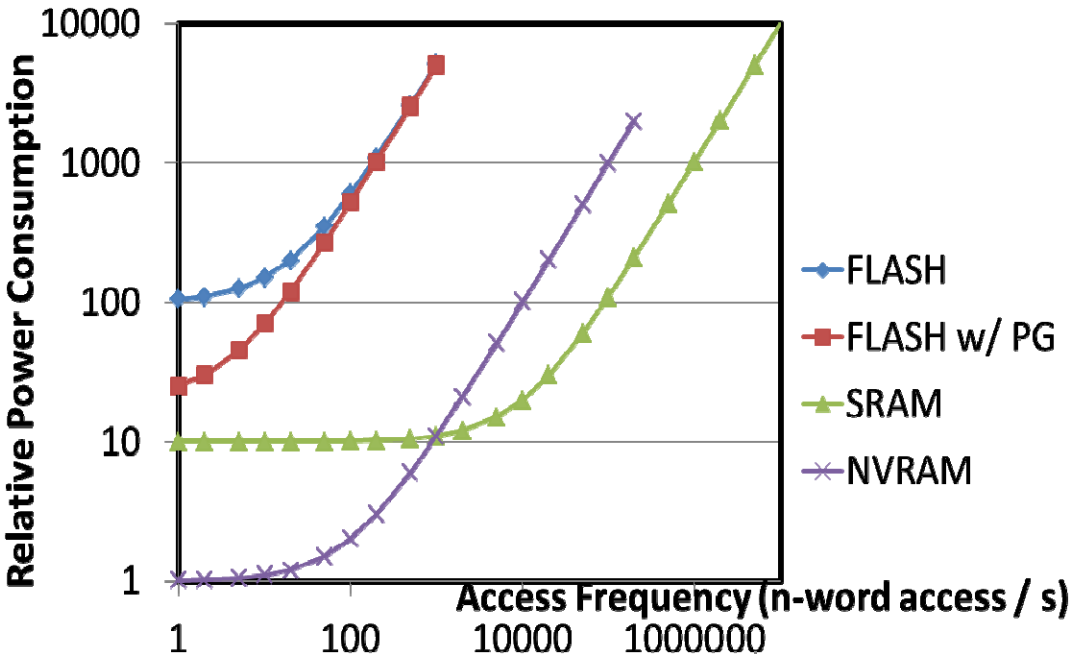


不揮発性メモリの損益分岐点

- Pitfall（落とし穴）: 揮発性メモリを不揮発性メモリで置き換えれば、必ず電力は削減できる ← *これは誤り*
- アクセス時のエネルギー: 不揮発性メモリ ＞ 揮発性メモリ
  → 不揮発性メモリの損益分岐点 (BET) が重要



a : extra access energy
b : reduced leakage energy
c : actual reduced energy

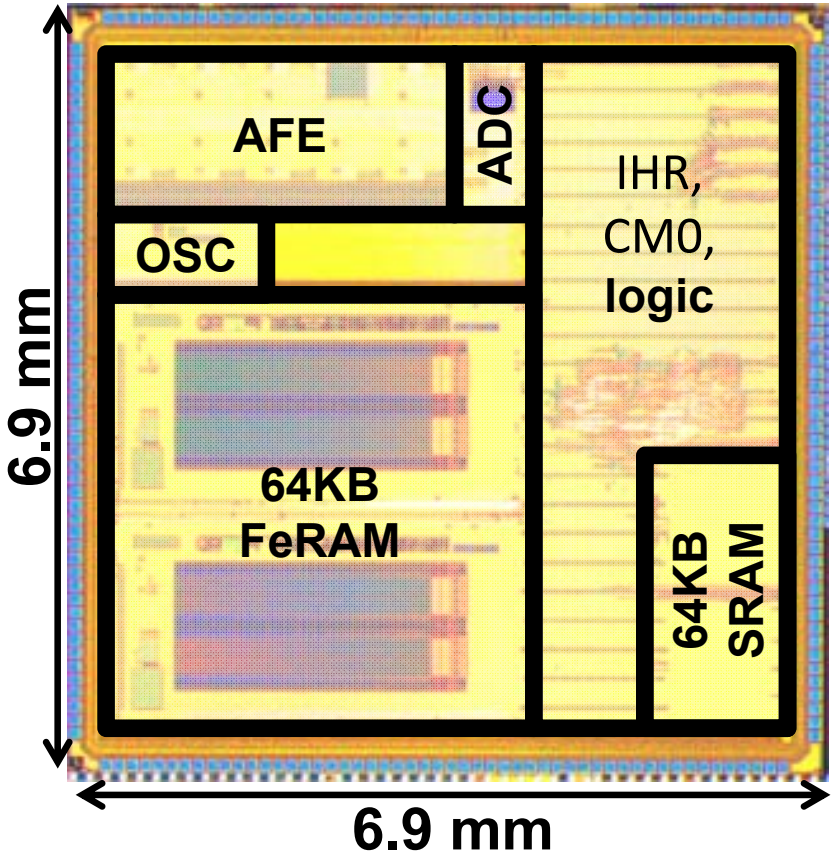


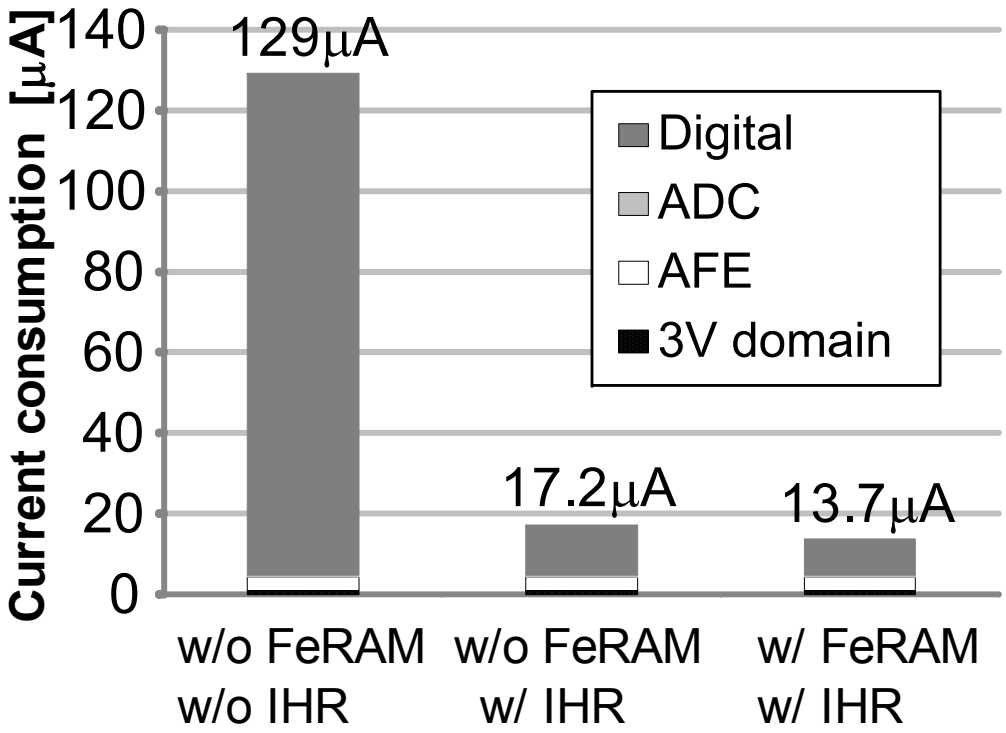
→ 1K words 書き込みで BET=1sec

参考: [中村2014]

不揮発メモリを用いたプロセッサの例

- Bio-information sensor
  (ROHM + OMRON HEALTHCARE + Kobe Univ.)



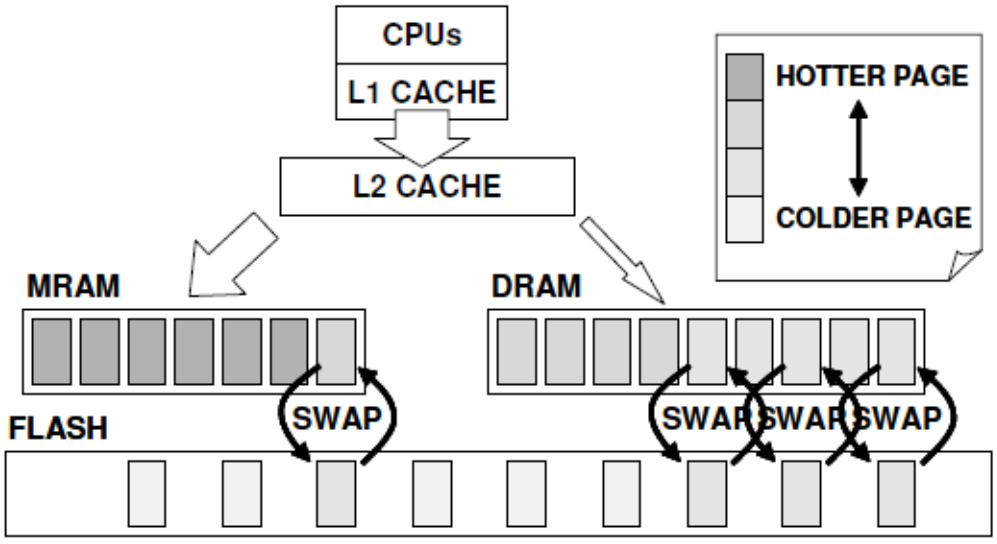


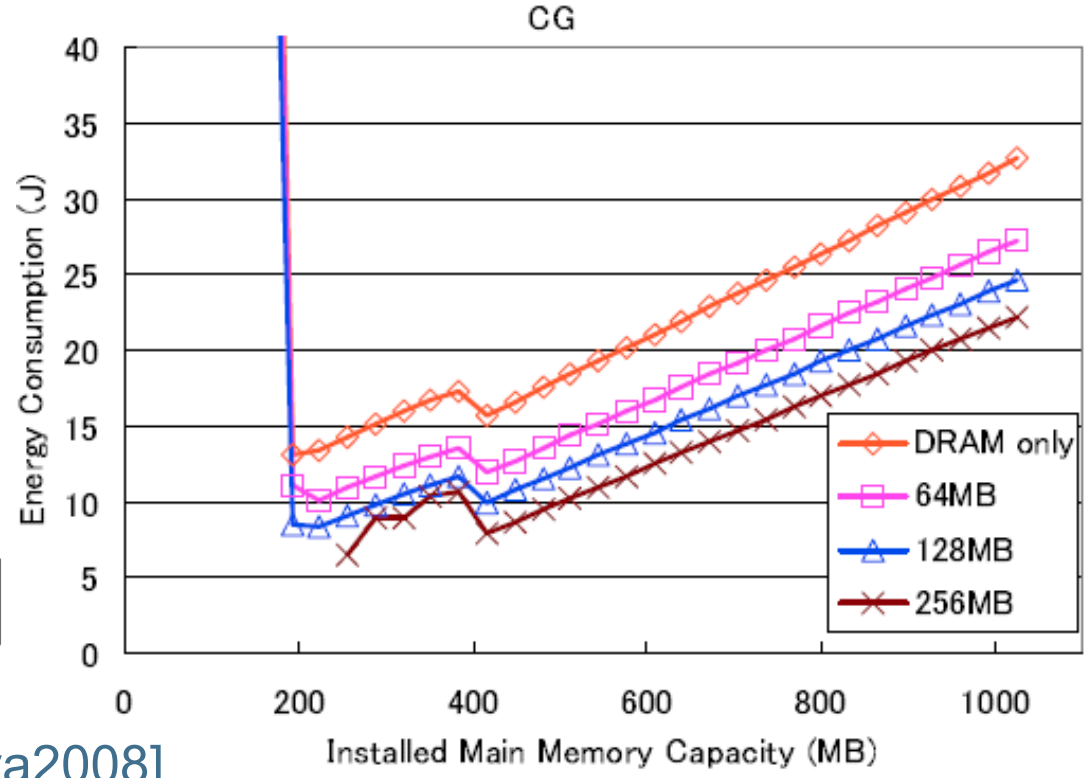
ADC, IHR : 128 sampling/sec
CM0 and FeRAM : Normally-Off,
1 sample /sec

出典: [Izumi2013, Nakada2015]

不揮発性メモリのHPCシステムへの適用

- DRAMとMRAMを主記憶に、FLASHをスワップ領域用デバイスへ適用した例
  - なるべく高速メモリへのアクセスとなるようページングを工夫
  - 最大で25%のメモリシステムの省エネルギー化を達成





出典: [Hosogaya2008]

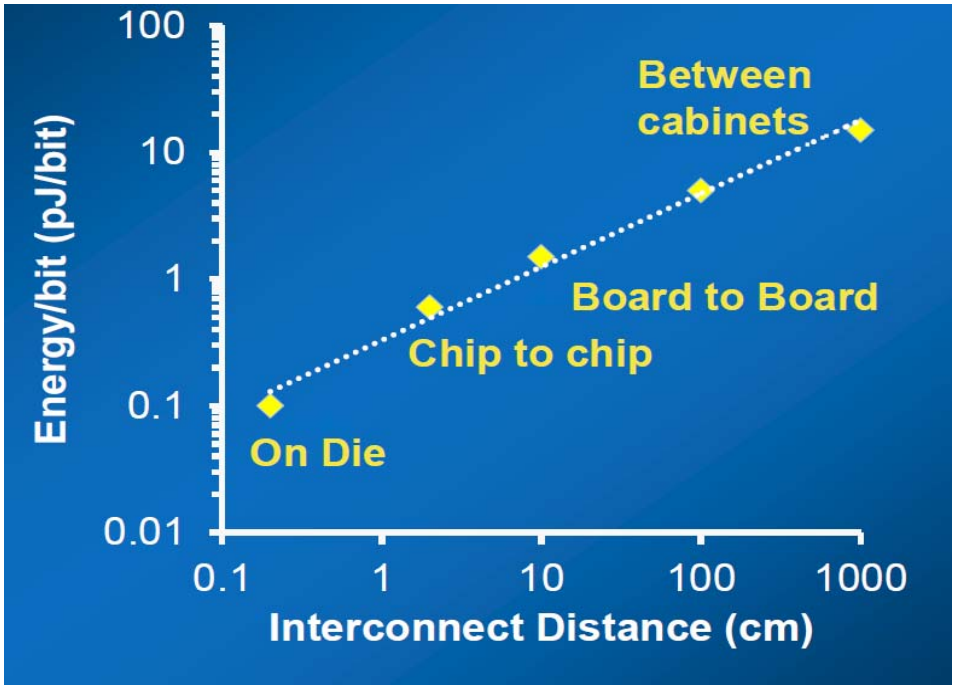
本チュートリアルの構成

- 高性能計算機の消費電力トレンド
- 計算機システムにおける電力消費の基礎
- 省電力・省エネ技術
  - Dark-silicon問題とプロセッサの省電力・省エネ化技術
  - メモリの省電力化技術
  - インターコネクションネットワークの省電力化技術
  - システムソフトウェアレベルでの電力制御技術
- 将来展望

通信の消費電力

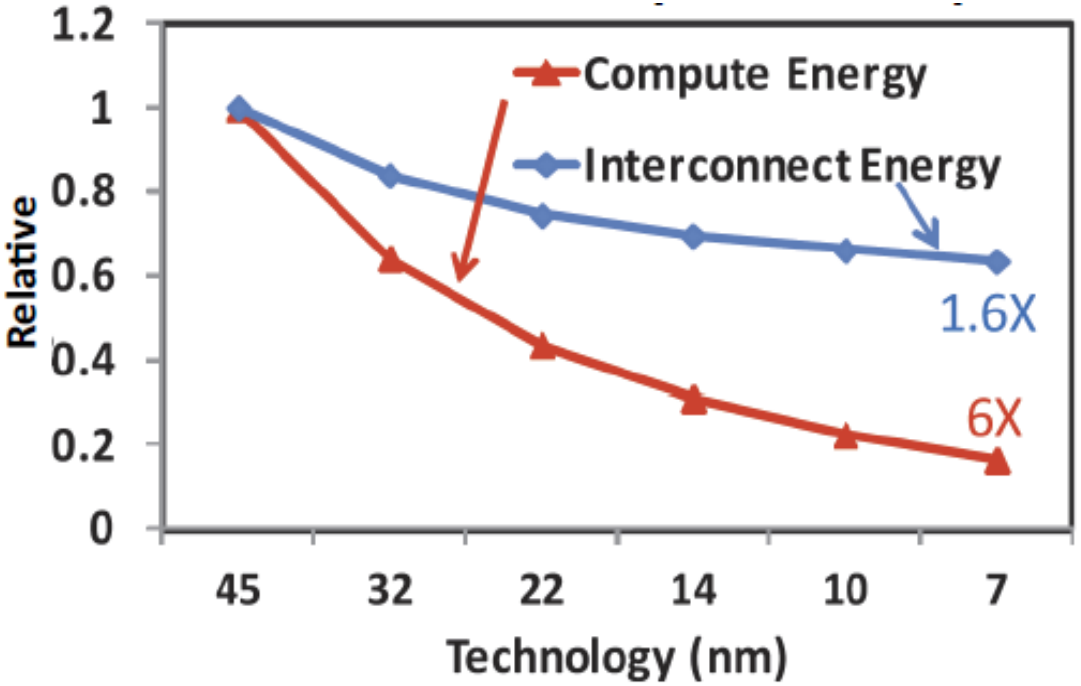
- 距離に応じたデータ移動に必要なエネルギーが増大
  - 局所性の活用が重要
- 通信エネルギーは計算エネルギーほどスケールしない

距離に応じたデータ移動のエネルギー



出典：[Borkar2013]

計算と通信(On-die)のエネルギー

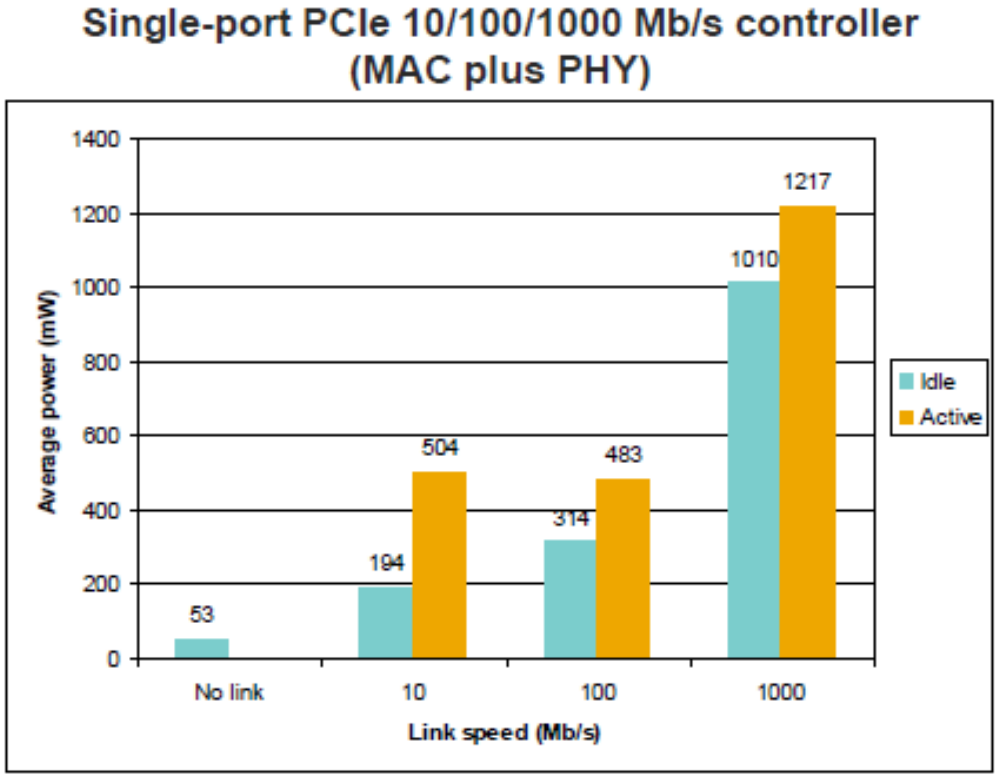


出典：[Borkar-JLT2013]

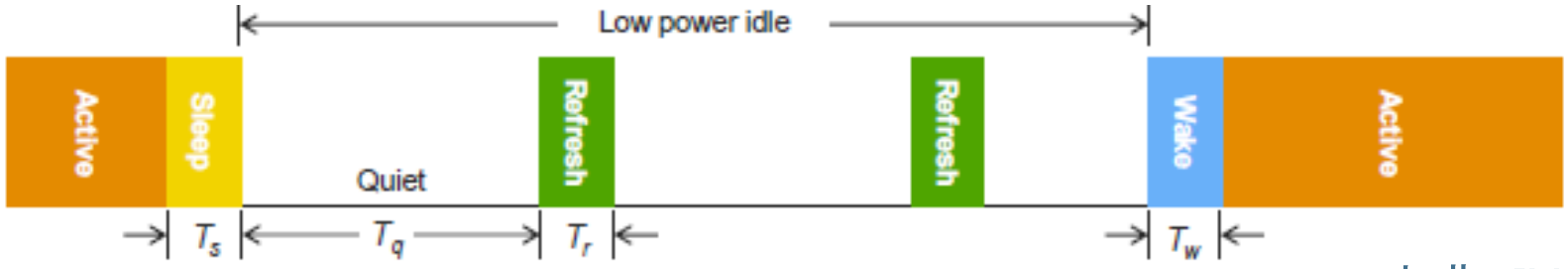
Energy Efficient Ethernet (EEE)

- Rapid PHY selection (RPS)
  - トラフィック量に応じて伝送速度を変更して低電力化
- Low Power Idle (LPI)
  - 通信が行われていない際に部分的に回路の電源をオフ
  - タイミングの調整などのため定期的に信号をやり取り

→Activeモードへ高速復帰



Source: Intel, Intel® 82573L Gigabit Ethernet Controller, 130 nm
"Idle" = no traffic
"Active" = bi-directional, line-rate traffic

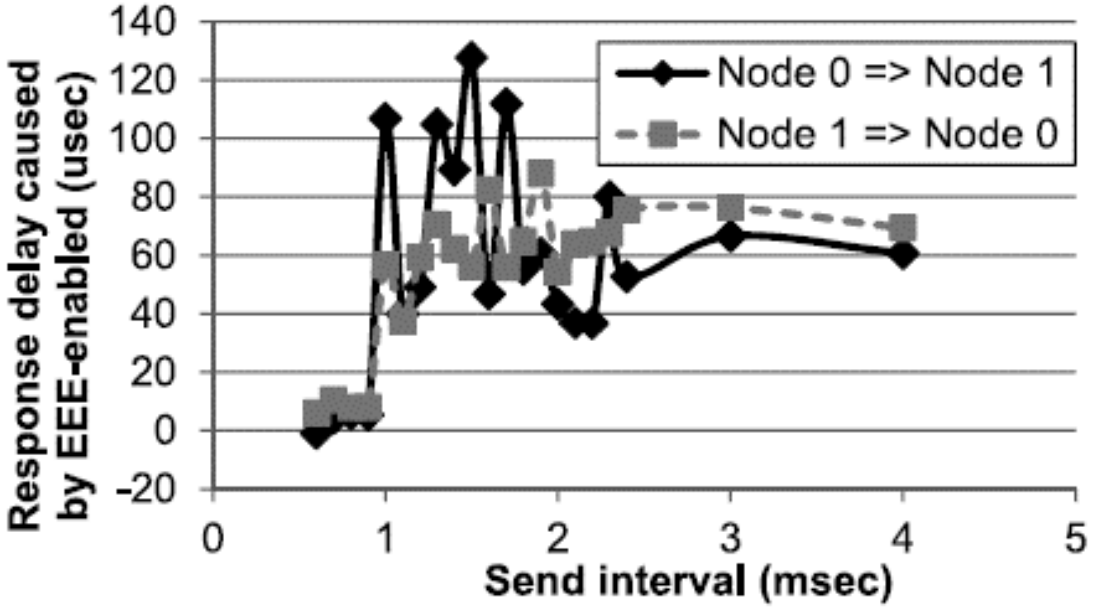


出典:[Healey2010]

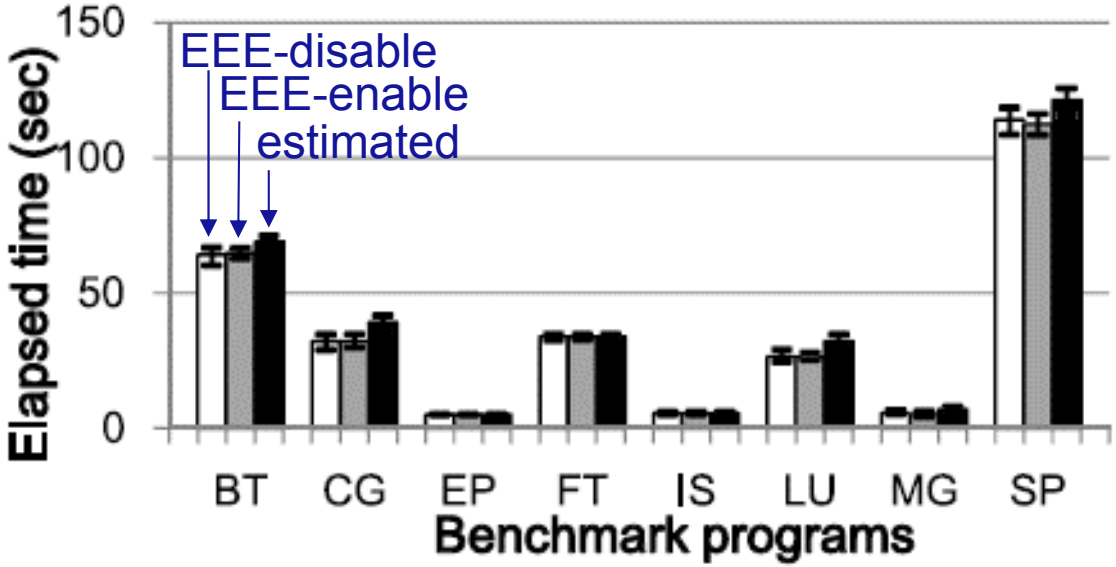
EEE利用時の性能

出典：[Miwa2013]

- EEE利用による遅延
  - Ping-Pongプログラム
  - Switch: Dell PowerConnect5548
  - 1msecタイムアウトインターバルでスリープ

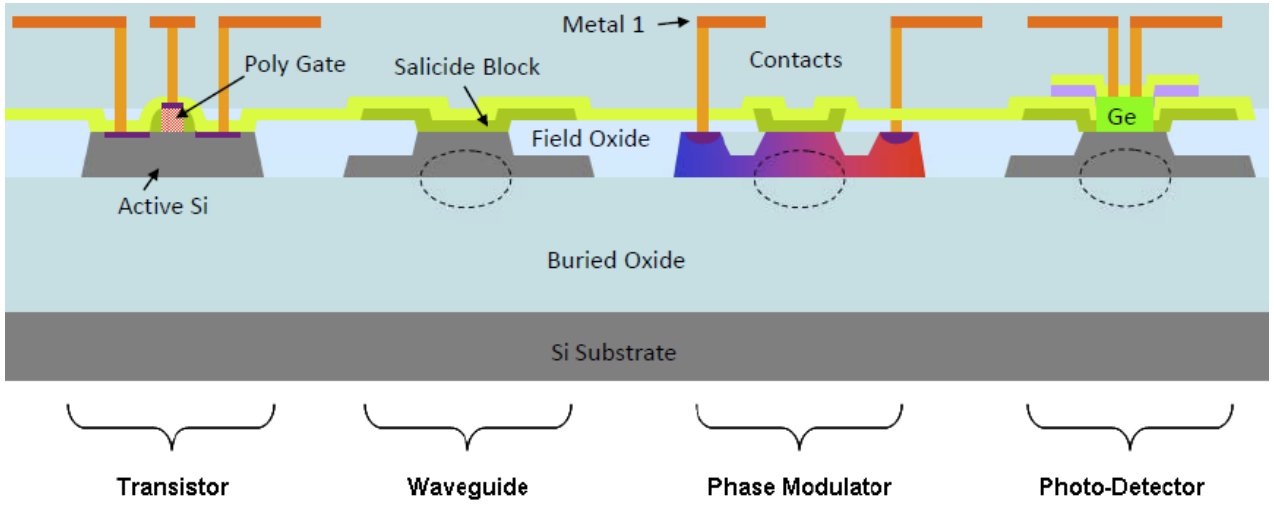


- EEE利用時のNPBの性能
  - 16ノードで実行
  - 性能低下は小さい



Silicon Photonics

- シリコン上に光回路を構成
- Silicon Photonicsの利点
  - 高バンド幅：Wavelength parallelismの利用
  - 低消費エネルギー：距離に非依存、1pJ/bitを達成できる可能性
  - 低コスト：CMOSプロセスで製造
  - 電子回路(エレクトロニクス)との接続性
  - スケーラビリティ: 半導体プロセスの進展の恩恵を享受



出典：[Welch2010]

エレクトロニクス通信と光通信の比較

出典：[Bergman2011]

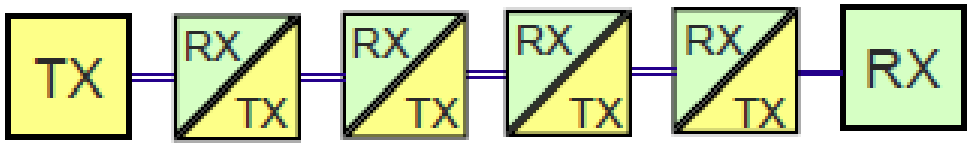
Photonics changes the rules for Bandwidth, Energy, and Distance.

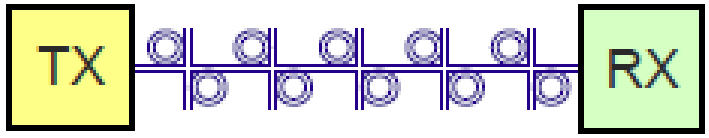
ELECTRONICS:

- Buffer, receive and re-transmit at every router.
- Each bus lane routed independently. ($P \propto N_{LANES}$)
- Off-chip BW is pin-limited and power hungry.

OPTICS:

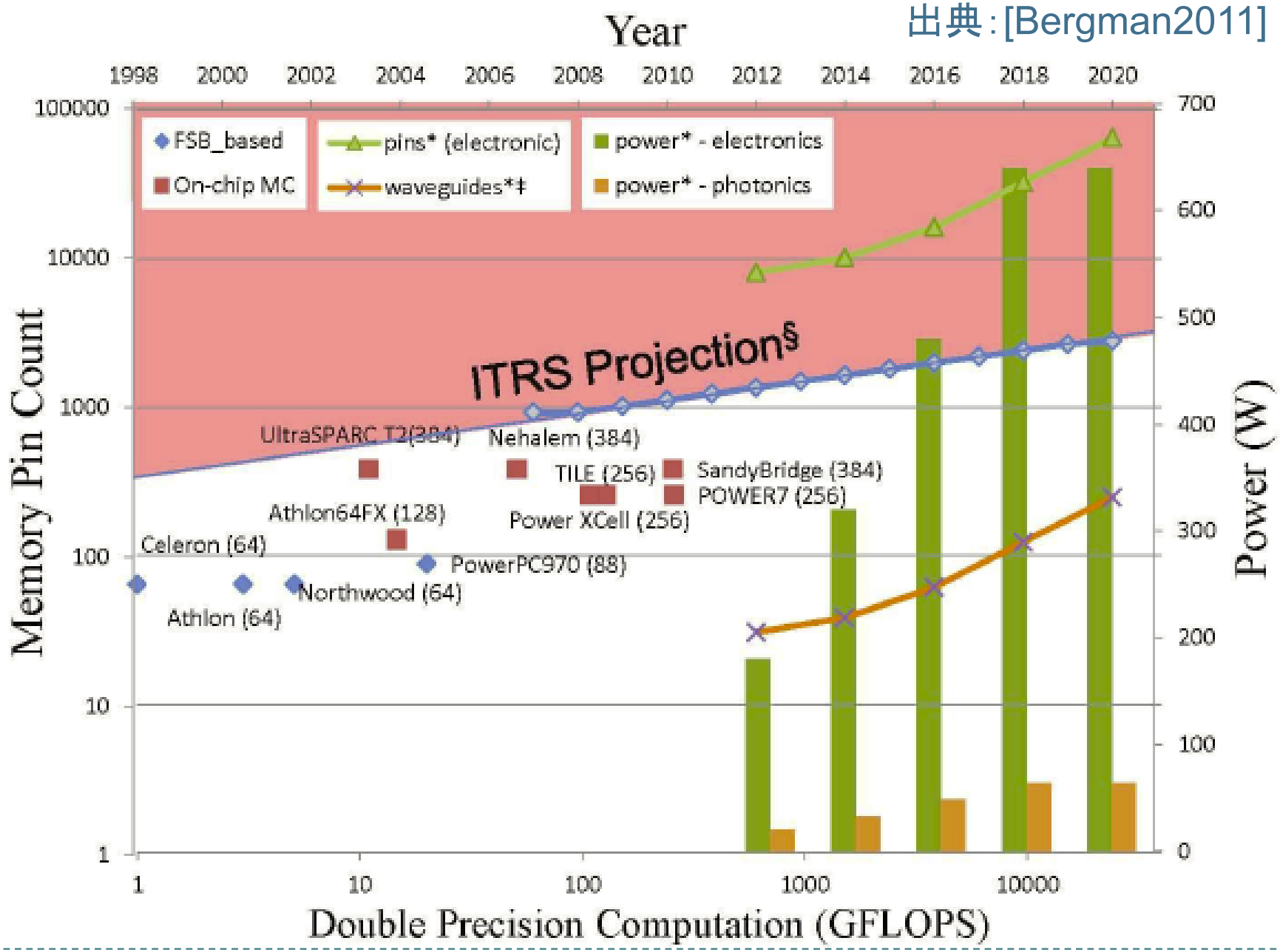
- Modulate/receive high bandwidth data stream once per communication event.
- Broadband switch routes entire multi-wavelength stream.
- Off-chip BW = On-chip BW for nearly same power.





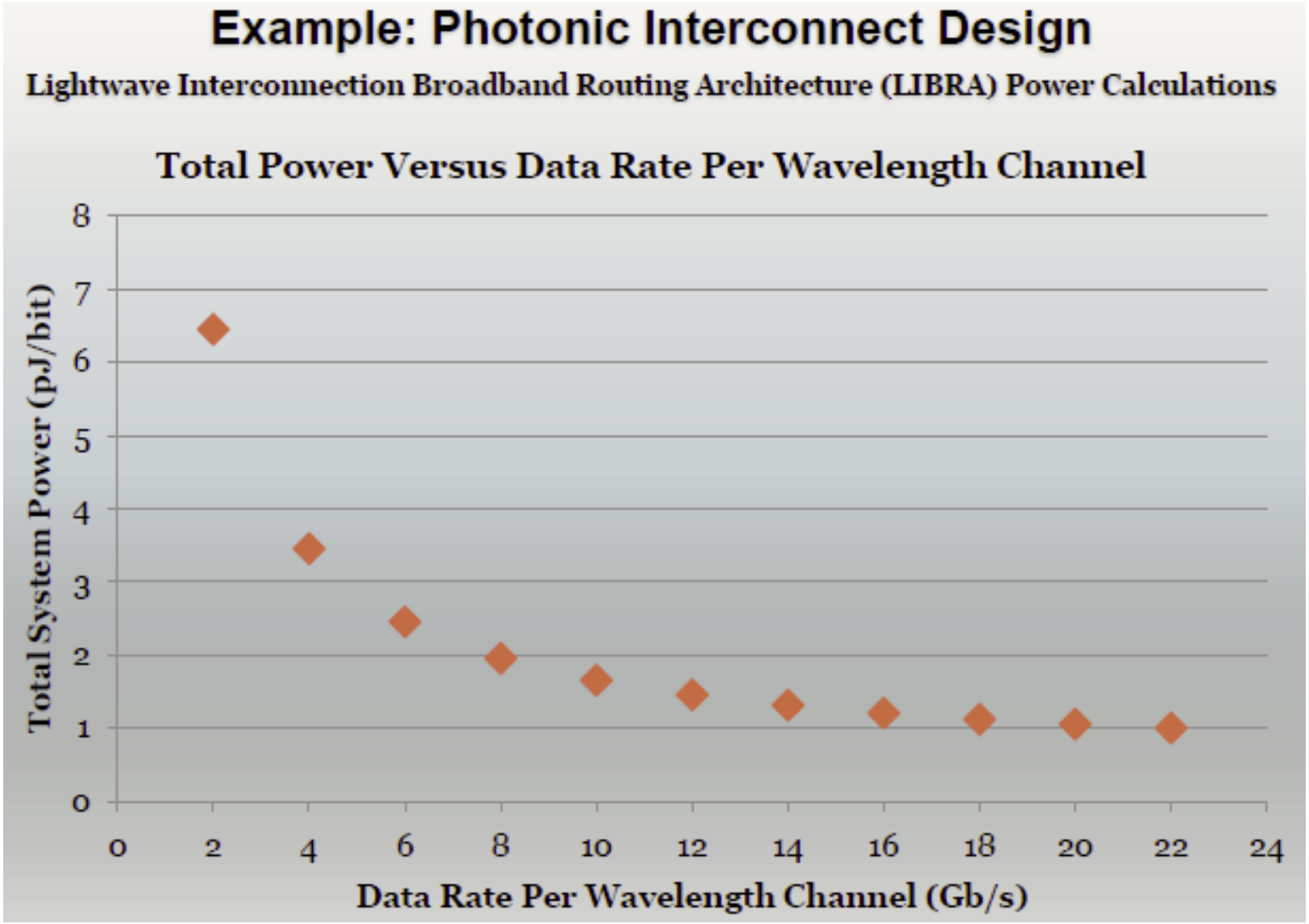
Photonicsの必要性

出典：[Bergman2011]



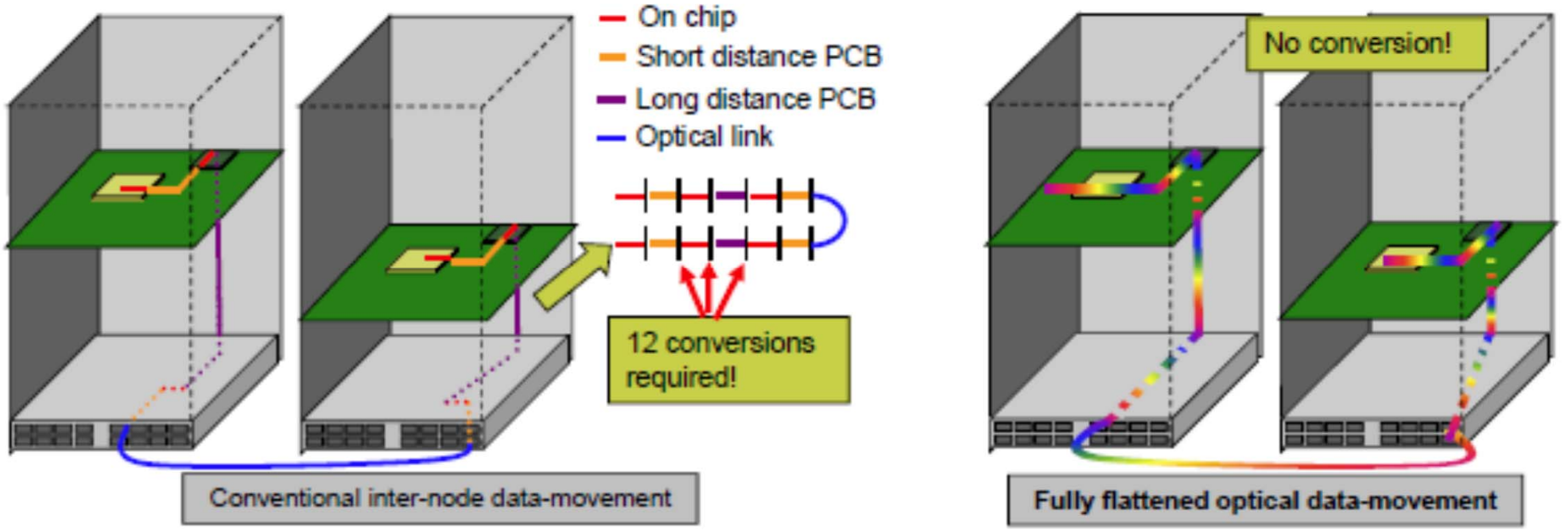
Photonicsによるネットワークの電力効率

出典：[Bergman2011]



HPCシステムへの利用例

- Optically-switched system
  - Cut-through bufferless通信が必要
  - 複数回のフォーマット変換が不必要に
  - 距離に非依存な通信性能

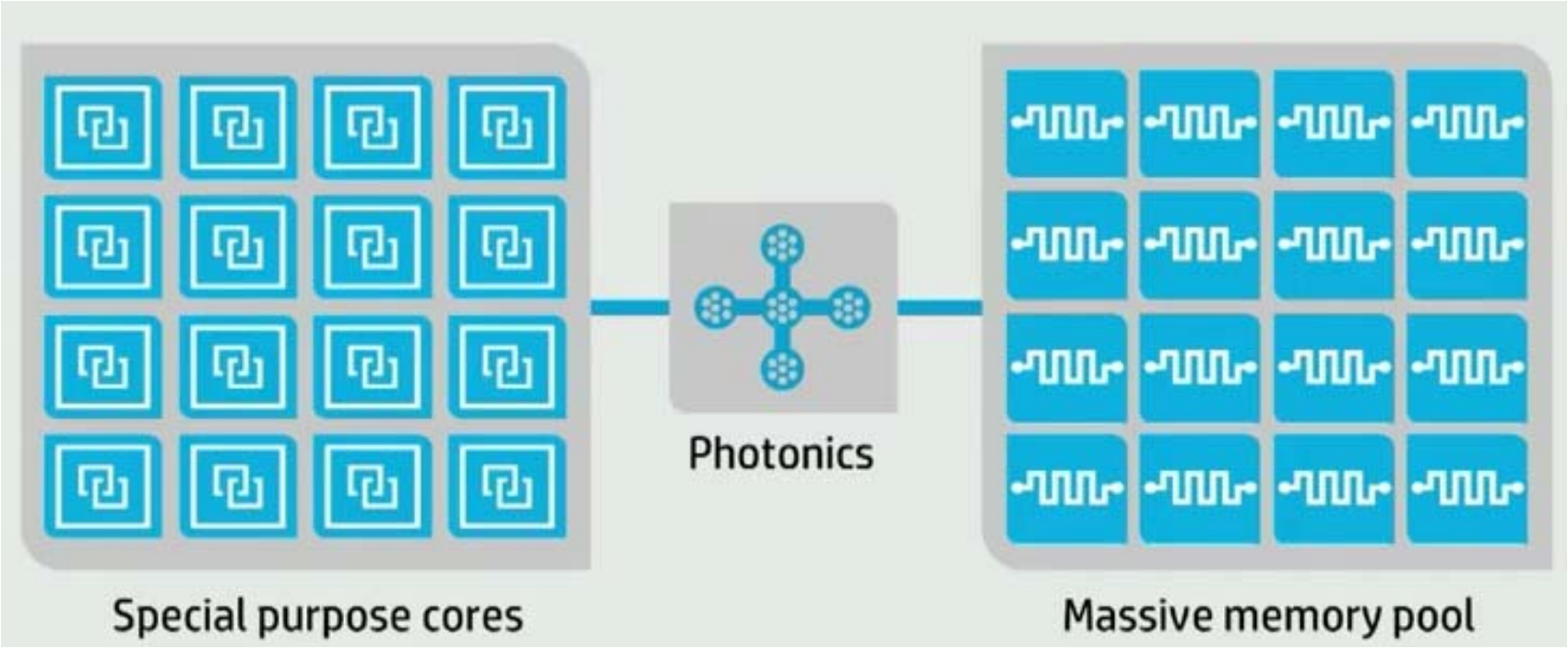


出典：[Bergman-OFC2014]

システムアーキテクチャへの影響

- HP “The Machine”

出典：[HP2014]

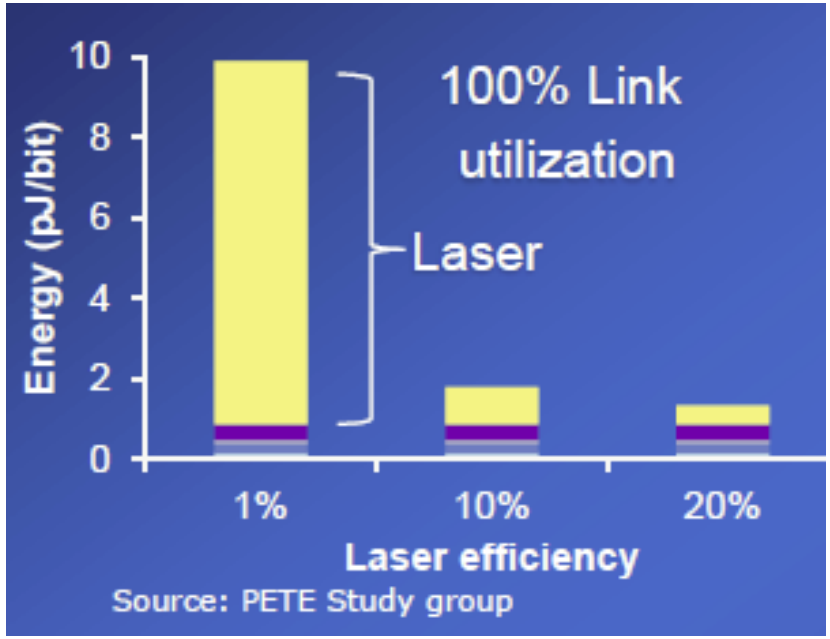


- The Machineアーキテクチャの特徴
  - 用途特化型コア利用による高電力効率
  - 不揮発性メモリを用いたメモリプール(主記憶＆ストレージの融合)
  - それらをつなぐPhotonics技術

Silicon Photonicsの課題

- Circuit Switchingの必要性
  - 光通信はcircuit switchingベース
  - 通信パス制御、セットアップの遅延
- スケーラビリティ
  - 1Kポート以上のラック間通信は簡単ではない
  - Wavelength parallelismでは不十分
- 製造プロセス
  - 大量生産の製造技術の確立
  - ファブや設計ツール等の開発が必要
- レーザーのエネルギー効率の考慮

出典：[Borkar2013]



本チュートリアルの構成

- 高性能計算機の消費電力トレンド
- 計算機システムにおける電力消費の基礎
- 省電力・省エネ技術
  - Dark-silicon問題とプロセッサの省電力・省エネ化技術
  - メモリの省電力化技術
  - インターコネクションネットワークの省電力化技術
  - システムソフトウェアレベルでの電力制御技術
- 将来展望

電力マネージメント技術

- エクサスケールでは電力マネージメントが重要
  - ハードウェア／ソフトウェアの電力管理
  - 各構成要素、各システム階層でのきめ細かな電力制御
  - Power-cappingのもとでの Over-provisioningも有望
- 重要技術項目
  - 電力消費状況の(リアルタイム)モニタリング
    - 電力観測インタフェースを備えるシステムが普及しつつある
    - e.g.) BlueGene/P、BlueGene/Q、Intel Sandy Bridge (RAPL)
  - 電力性能比を最適化するアルゴリズム/ライブラリ
  - 電力制御インタフェースの標準化
  - 電力制御用ノブの適切なモデリングと最適化制御

近年のCPUの電力観測・制御のインタフェース

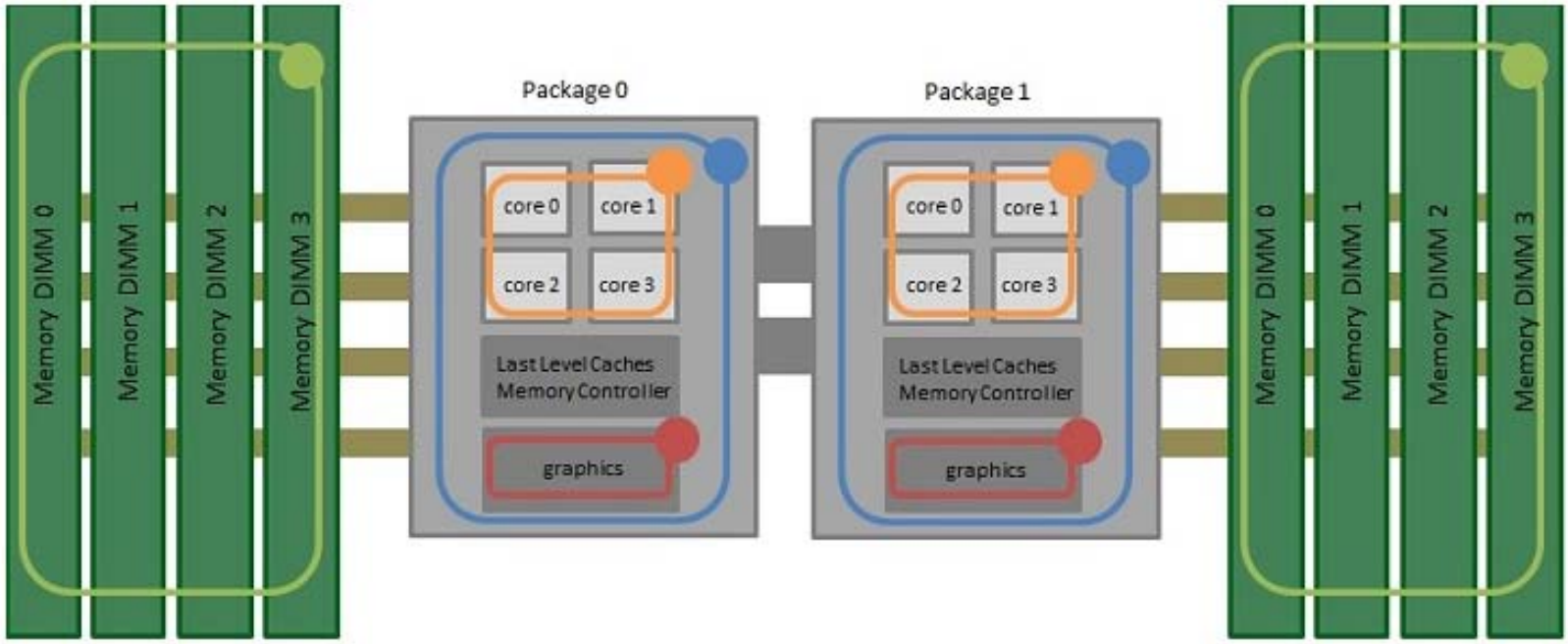
- DVFS: ほとんどのCPUでクロック周波数(電圧)を変更可能
  - Linuxのインタフェース
    - CPUFreq：各CPUの周波数を設定するフレームワーク
      （/sys/devices/system/cpu/cpu<n>/cpufreq/ 以下で種々の設定）
    - governor：クロックや電圧を調節する際の動作モード決定

governor	説明
ondemand	負荷に応じて周波数切り替え（CPU負荷が95%で切り替え）
performance	負荷にかかわらず最大周波数で動作
conservative	負荷に応じて周波数切り替え（CPU負荷が70%で切り替え）
powersave	負荷にかかわらず最低周波数で動作
userspace	ユーザが周波数を指定

- Power Cap機能
  - Intel Xeon Sandy Bridge: Running Average Power Limit (RAPL)
  - AMD Bulldozer Opteron: Power Cap Manager
  - IBM POWER6+: Active Energy Manager

電力観測・制御インタフェースの例：RAPL

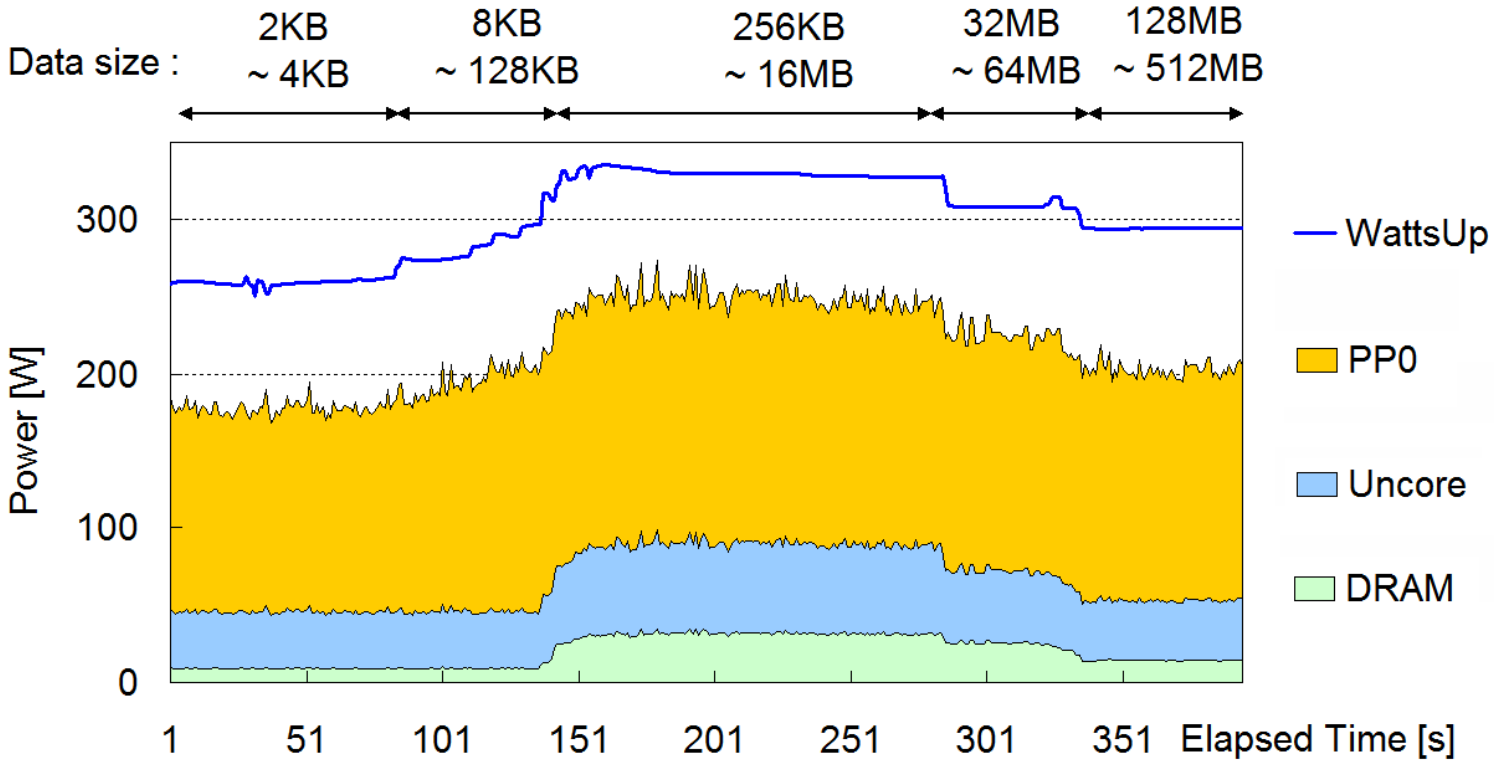
- RAPL (Running Average Power Limit)インタフェース
  - Intel Sandy Bridgeマイクロアーキテクチャより搭載
  - パフォーマンスカウンタや温度等の情報を基に消費電力の見積り・制御
  - MSRを介して消費電力の取得や電力上限設定が可能
  - ドメイン毎に電力を計測



出典：[Intel2012]

RAPLによる電力モニタリングの例 出典:[カオ2013]

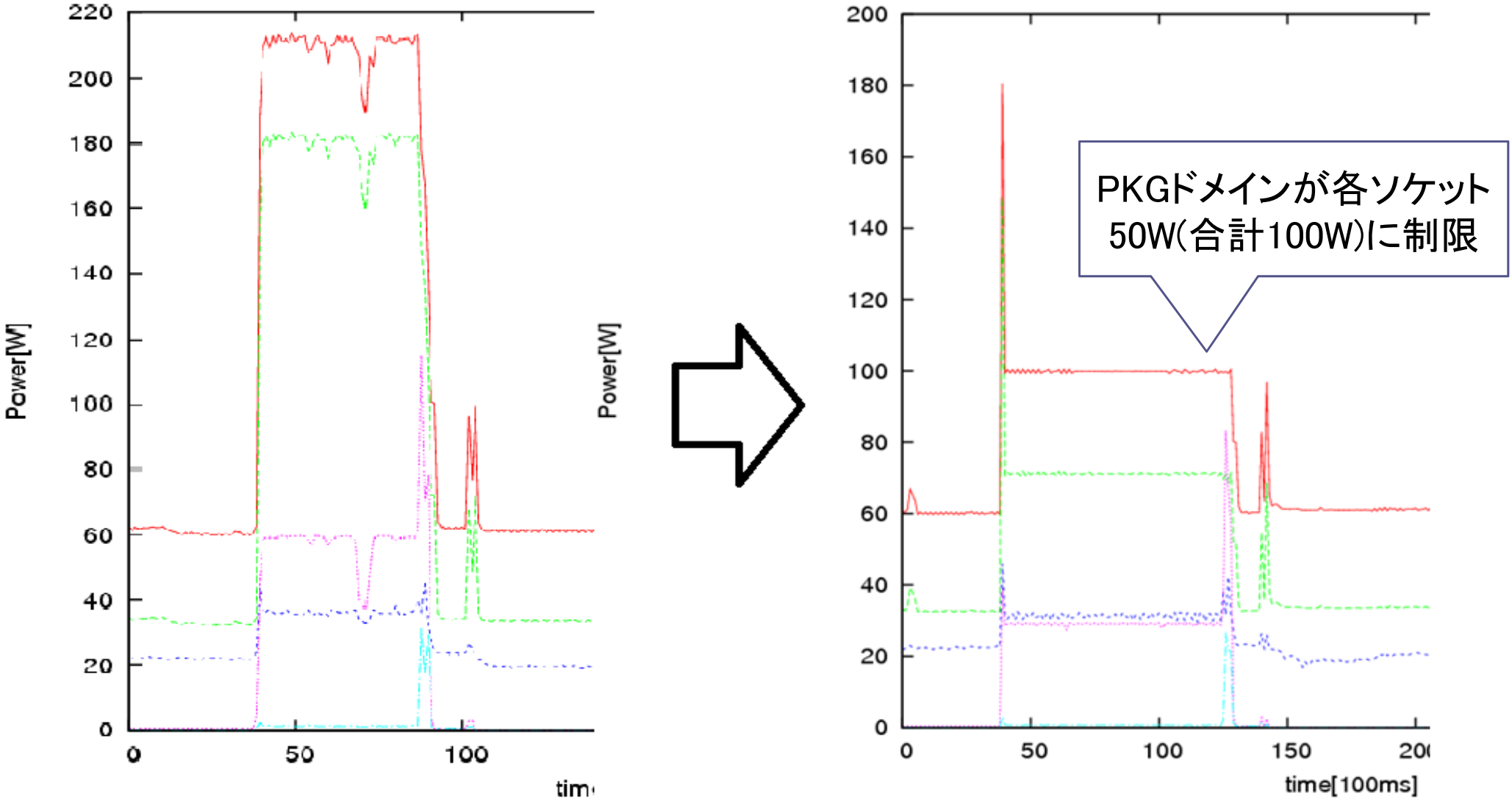
- ストリームアクセスプログラム
  - 16コア(MPI並列)、配列サイズ:2KB～2GB



- RAPLと外部電力計(WattsUp)の消費電力傾向は非常に良く一致
- メモリアクセス頻度の違いにより消費電力が大きく異なる
  - L2サイズ(256KB)以上:電力増加 → キャッシュアクセス増 + DRAMアクセス増
  - L3サイズ(20MB)以上:電力減少 → 単位時間あたりのアクセス頻度減少

消費電力制約の設定例：CPU

- パッケージ電力(PKG)を50Wに設定



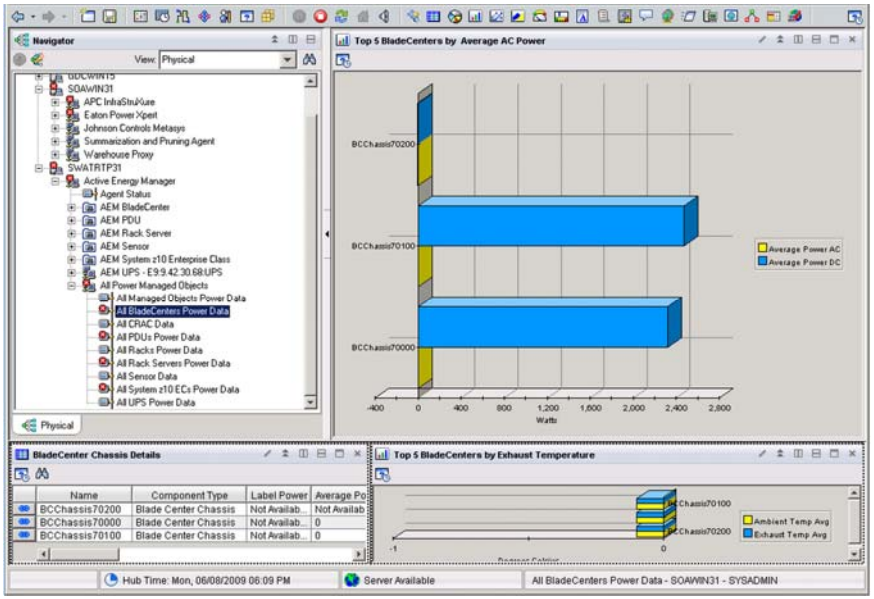
電力を意識したシステム管理ソフトへの要求

- ジョブ・リソーススケジューリング
  - どのジョブをどのノードで実行するか
  - 各ジョブにどれだけの電力資源を配分するか
  - 各ハードウェアの電力ノブをどのように調整するか

- 目的関数・制約
  - システム全体のジョブスループット (#Jobs/hour)
  - 各ジョブの実行性能 (FLOPS)
  - ジョブあたりの消費エネルギー、電力コスト
  - 電力制約の遵守度

電力管理機能を持つ資源管理ツール(1/2)

- IBM Systems Director Active Energy Manager
  - モニター機能
    - ノード、ラック単位での電力計測
    - 電力傾向分析、熱傾向分析など
  - 電力管理機能
    - 省電力オプション設定
    - パワーキャッピング
    - 省エネ度の予測
    - :

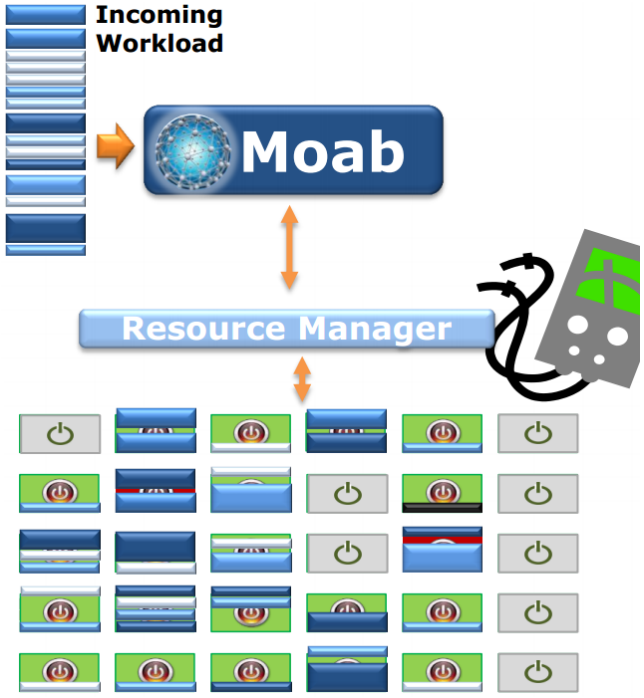


出典:[IBM2014]

- IBM Loadleveler Energy Aware Scheduling
  - ジョブ毎の電力ログ、電力制御
  - ポリシーに応じたCPU周波数の設定
  - :

電力管理機能を持つ資源管理ツール(2/2)

- SLURM Workload Manager
  - 消費電力の計測
    - RAPLによる消費電力/エネルギー計測とリポート(IPMI計測は開発中)
    - 消費エネルギーの計算とデータベース化
  - DVFSを用いたジョブの周波数制御
  - 省エネを考慮したジョブスケジューリング(開発中)
- MOAB
  - ワークロードのconsolidation
  - ノードの電力状態マネージメント
    - 省電力モード、ノード電源オン・オフ
    - パワーキャッピング(開発中)
    - DVFSを用いたジョブの周波数制御
    - 10-30%の省エネ化



出典:[Schott2013]

電力を考慮したジョブスケジューリングの研究

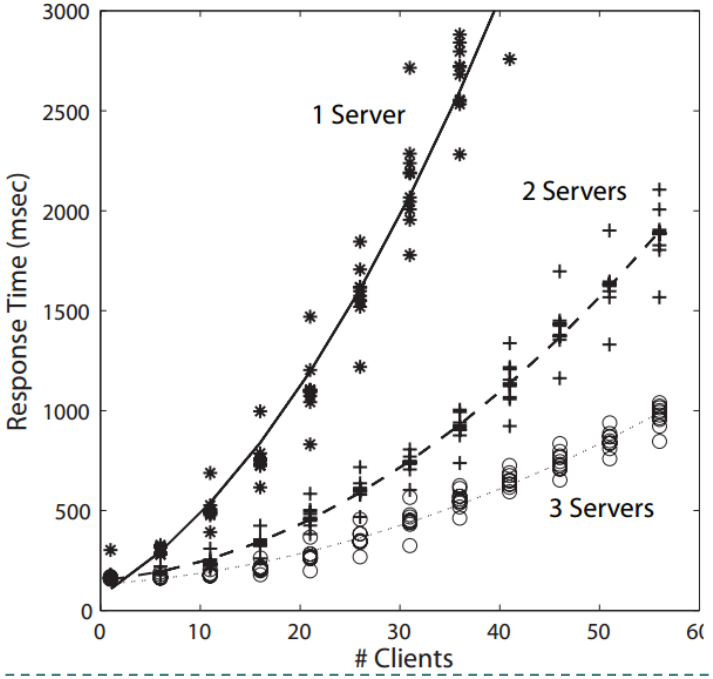
- [Auweter2013] A. Auweter et al., “A case study of energy aware scheduling on SuperMUC”, ISC’14, LNCS vol.8488, pp.394–409, 2014.
  - 履歴を元にLoadLevelerでCPU周波数毎の性能と消費電力を予想
  - 70%のジョブを2.3GHzで動作、最高周波数で全ジョブを実行に比べ6%の省エネ化
- [Sarood2014] O. Sarood et al., “Maximizing Throughput of Over-provisioned HPC Data Centers under a Strict Power Budget”, SC’14, pp.807–818, Nov. 2014.
  - ジョブの電力性能モデルと電力制約に基づき、実行ジョブと電力割り当てを最適化
  - Moldable and malleable ジョブを仮定
- [Etinsk2012] M. Etinski et al., “Parallel job scheduling for power constrained hpc systems”, J. of Parallel Computing, Vol.38 Issue 12, pp.615-630, Dec. 2012.
  - キュー待ち時間を含めた性能をCPU周波数毎に予測、最適なCPUをジョブ毎に設定
  - 電力制約下の下での性能を最大可するための定式化
- [Geogiou2014] Y. Georgiou et al., “Energy Accounting and Control with SLURM Resource and Job Management System”, ICDCN2014, LNCS Vol.8314, pp.96-118, Jan. 2014.
  - SLURMに対して消費エネルギーアカウンティングと周波数制御による省エネ機能を追加

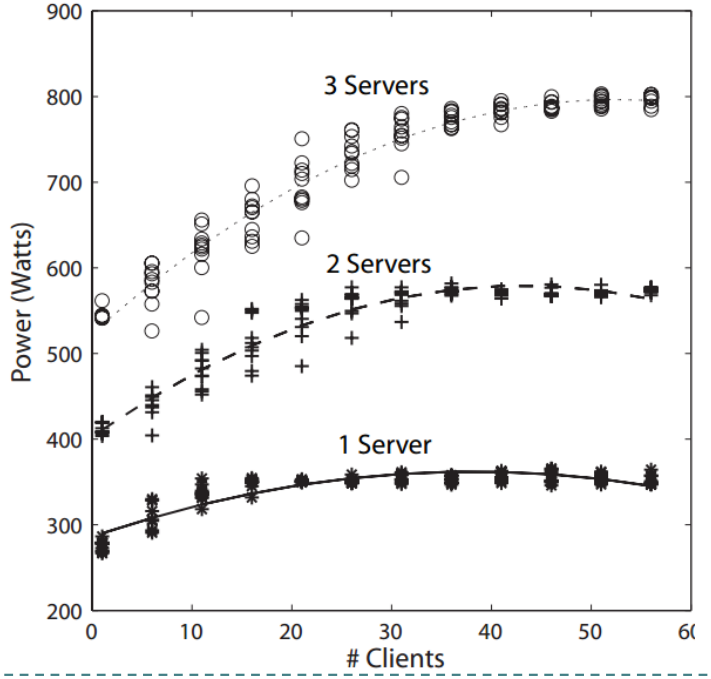
データセンタの電力管理に関する研究(1/2)

- [Das2008] R. Das et al, “Autonomic Multi-Agent Management of Power and Performance in Data Centers”, AAMAS 2008, pp.107-114, May 2008.
  - サーバ管理ツールとセンサを利用
  - エージェントベースの手法で低負荷時に性能を満たす範囲でサーバーの電源をオフ

when $C < 17$, power on one server;
when $17 \leq C < 34$, power on 2 servers;
when $34 \leq C < 57$, power on 3 servers;
when $C > 57$, power on 1 server
where C is the number of clients

出典:[Das2013]





データセンタの電力管理に関する研究(2/2)

- [Urgaonkar2013] R. Urgaonkar et al., “Optimal power cost management using stored energy in data centers”, SIGMETRICS '11, pp.221-232, 2011.
  - 低電力コストの時間にUPSに電力をチャージ
  - 高電力コスト時にUPSから一部電力を供給

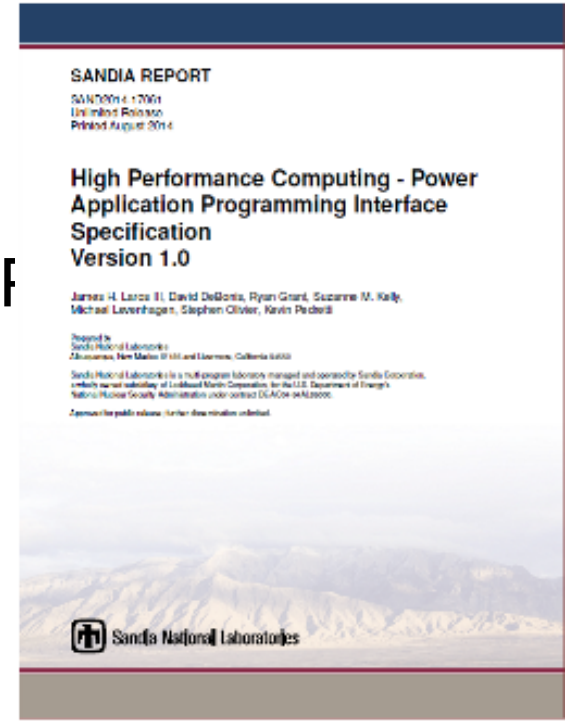
- [Chen2010] Y. Chen et al. “Integrated management of application performance, power and cooling in data centers”, NOMS2010, pp.615-622, 2010.
  - ワークロード毎に性能に応じて電力と冷却能力を割り当て
  - ワークロードのマイグレーション、サーバ統合を考慮

- [Leverich2009] J. Leverich et al., “Power Management of Datacenter Workloads Using Per-Core Power Gating”, Computer Architecture Letters, Vol.8, Issue 2, pp.48-51, 2009.
  - コア単位での電力供給制御とDVFSによりプロセッサ消費エネルギーを削減

電力マネージメント用API

- 標準でスケーラブルな電力観測・制御APIの必要性
  - 様々な電力センサや電力制御ノブ
    - 電力センサー: Intel RAPL、NVIDIA SMI、Smart PDUs、IPMI…
    - 電力制御ノブ: RAPL、CPUfreq、メモリホットプラグ、Low Power Idle、…
  - HPC計算機システムの大規模化・階層化・ヘテロジニアス化
- 電力マネージメントAPIの世界動向
  - Sandia National Laboratories HPC Power Application Programming Interface (PowerAPI)
    - Version 1.0が2014/9にリリース
    - 種々の制御レベル(ユーザ/管理者/OS/・・・)と階層的な対象(CPU/ボード/ケース/・・・)を定義
    - 実装はベンダー等が独自に行う

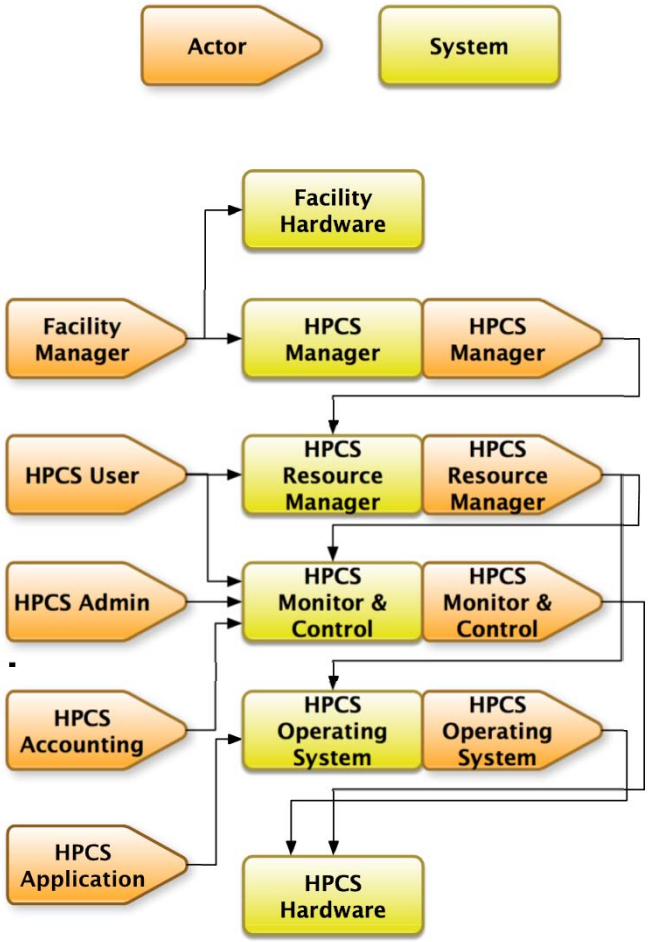
参考: [PowerAPI2014]



Sandia Power APIのインタフェース

- Actorとシステム間のインタフェース
- 以下の10インタフェースを考慮
  - HPCS Manager->Resource Manager
  - User->Resource Manager
  - User->Monitor & Control
  - Administrator->Monitor & Control
  - Accounting->Monitor & Control
  - Application->OS
  - Resource Manager->Monitor & Contrc
  - Resource Manager->OS
  - Monitor & Control->Hardware
  - OS->Hardware

Use Caseダイアグラム



出典:[PowerAPI2014]

各インタフェースで利用する機能の定義

参考：[PowerAPI2014]

- 以下の7カテゴリを提供
  - Initialization
  - Hierarchy Navigation
    - Hierarchy navigation for the objects
  - Group
    - Same operation on multiple objects
  - Attribute
    - Get/set for the attributes
  - Metadata
  - Statistics
  - Version
    - Version of the specification supported by the implementation

Sandia Power APIのAPIの一部

- PWR_GetEnergyByUser()

参考 : [PowerAPI2014]

```
int PWR_GetEnergyByUser( const char* userId,
                         double* value
                         PWR_StatTimes* statTimes );
```

Argument(s)	Input and/or Output	Description
const char* userId	Input	The user Id that the energy value will be collected for.
double* value	Output	Pointer to a double that will contain the energy value.
PWR_StatTimes* statTimes	Input/Output	The user specified window for the energy value (start and stop times must be specified).

- PWR_ObjAttrGetValue()

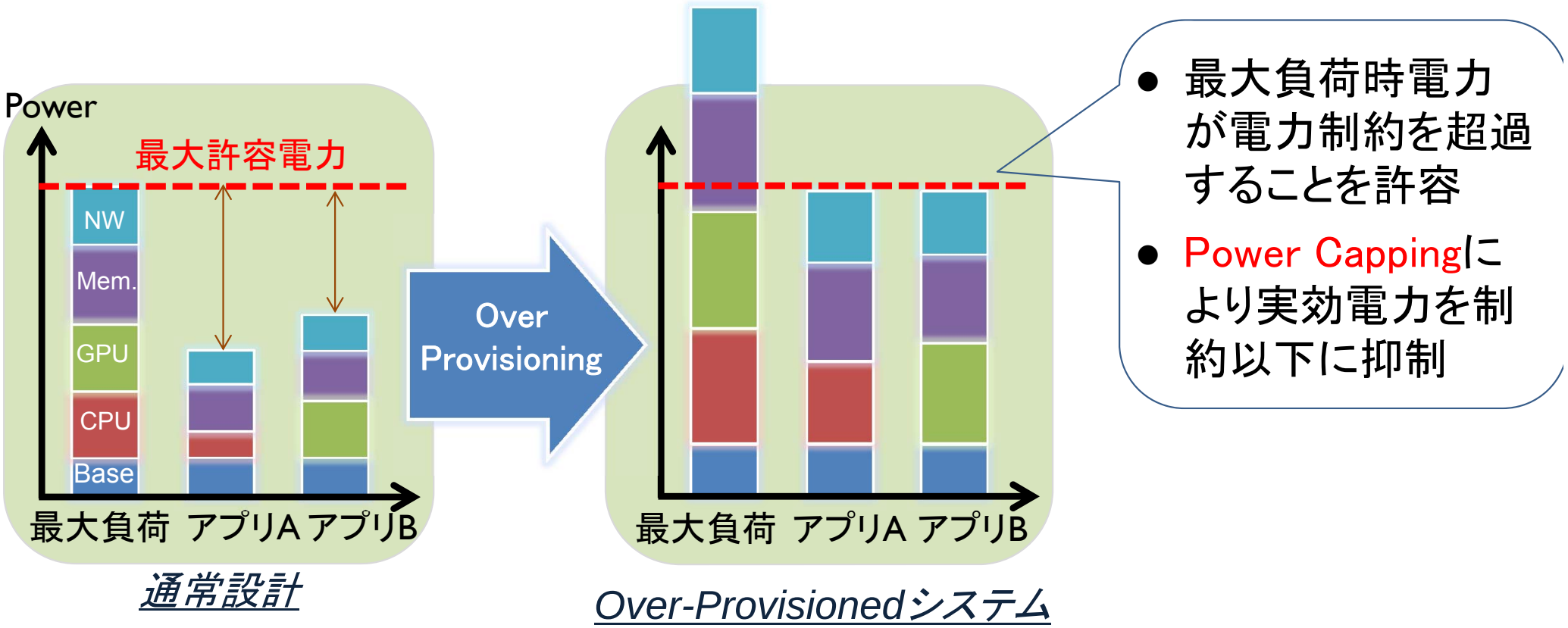
```
int PWR_ObjAttrGetValue( PWR_Obj object,
                         PWR_AttrName attr,
                         void* buf,
                         PWR_Time* ts);
```

Argument(s)	Input and/or Output	Description
PWR_Obj object	Input	The object that the user is acting on
PWR_AttrName attr	Input	The attribute the user wishes to access (get), see the PWR_AttrName type definition in section 3.4.
void* buf	Output	The user provided void pointer to the stor-

Hardware Over-Provisioning システム

- 電力制約以上のハードウェアをシステムに搭載
  - 最大負荷時電力が電力制約を超えることを許容
  - 全てのコンポーネントがフルに動作することはない

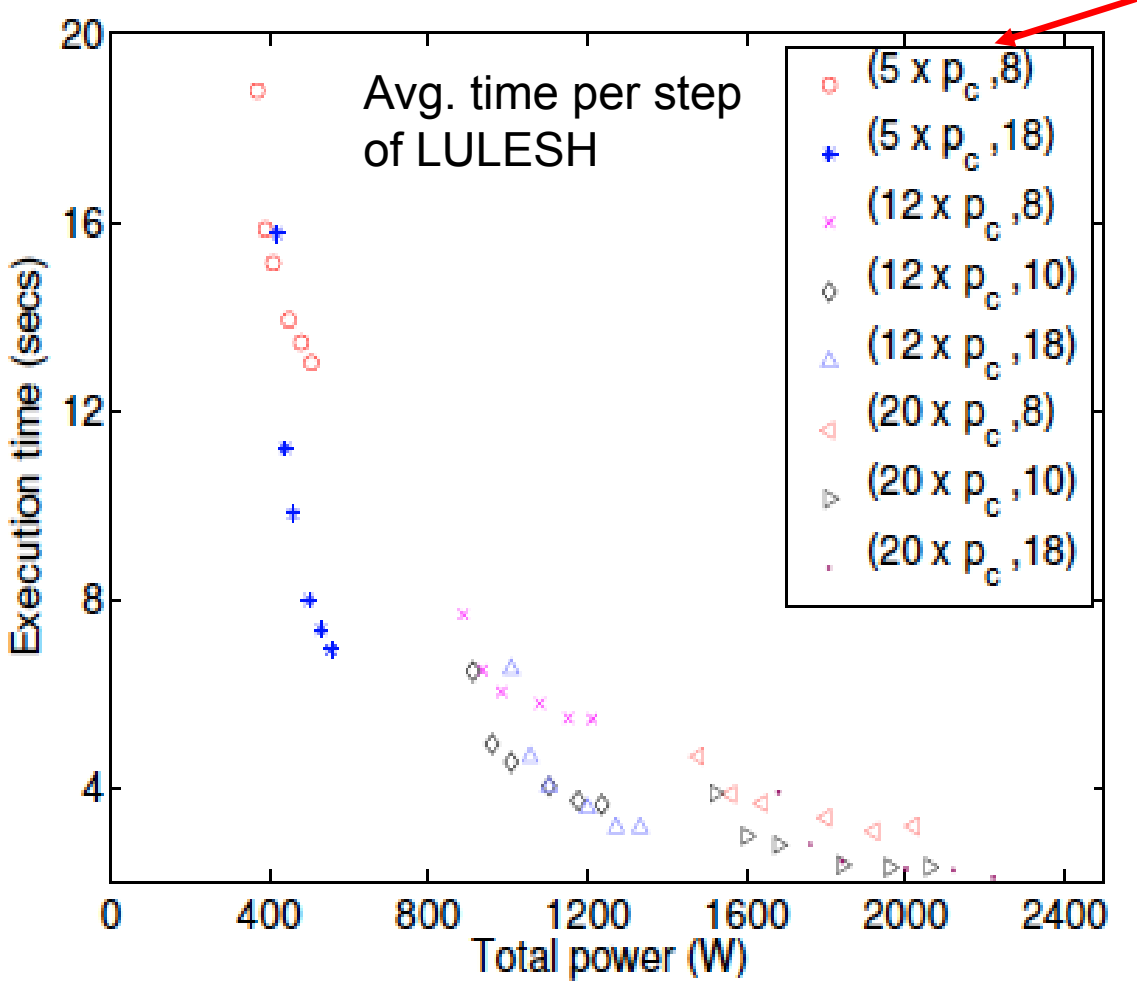
電力資源を各コンポーネントへ適応的に配分することで実効性能向上



CPU-Memoryの電力配分と実行時間の関係

- CPU電力とメモリ電力配分、およびノード数を変更した場合の制約電力下での実行時間

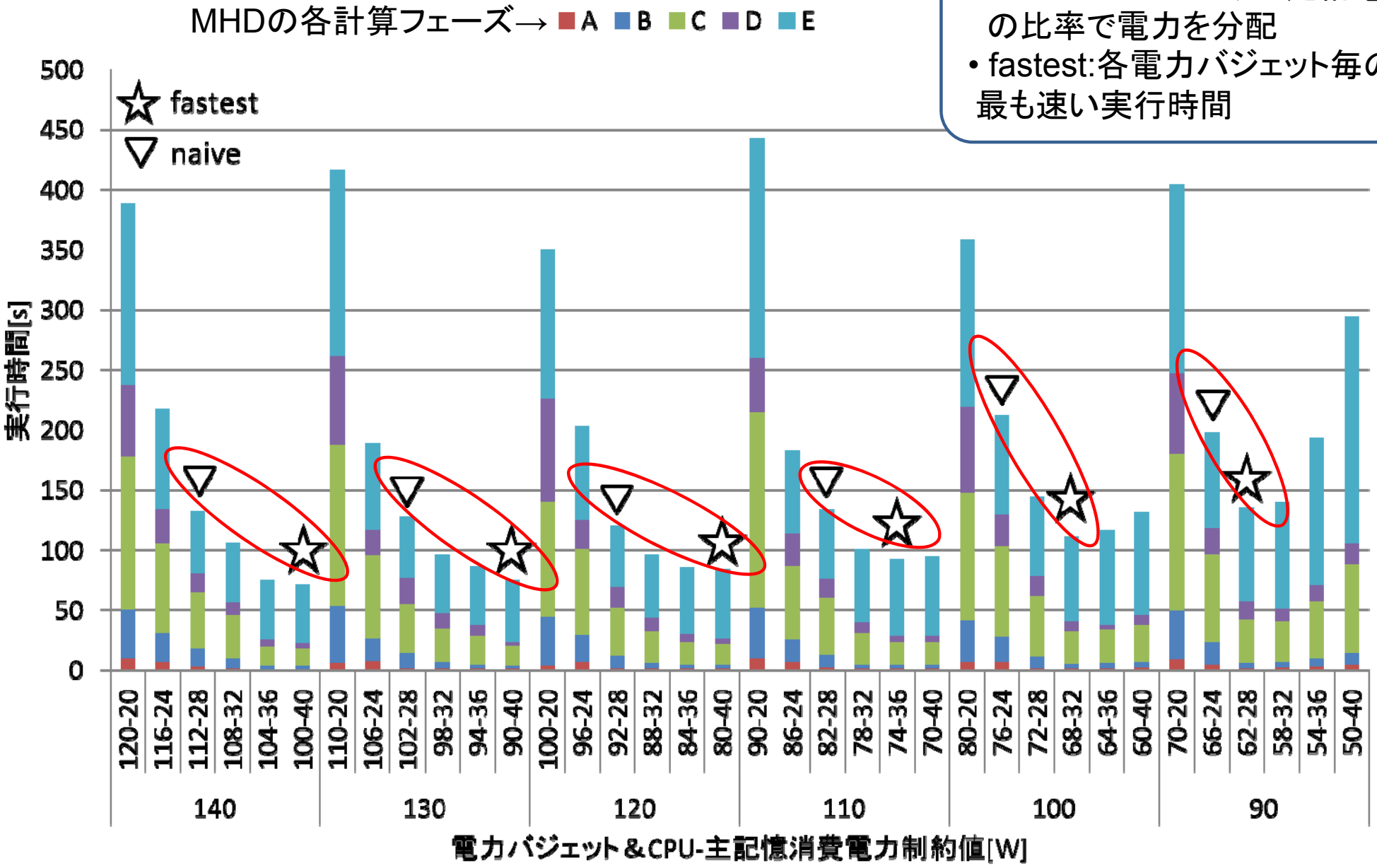
（ノード数 x CPU電力, メモリ電力）



出典：[Sarood2013]

CPU・メモリ間の電力配分実験結果(1/2)　出典:[吉田2013]

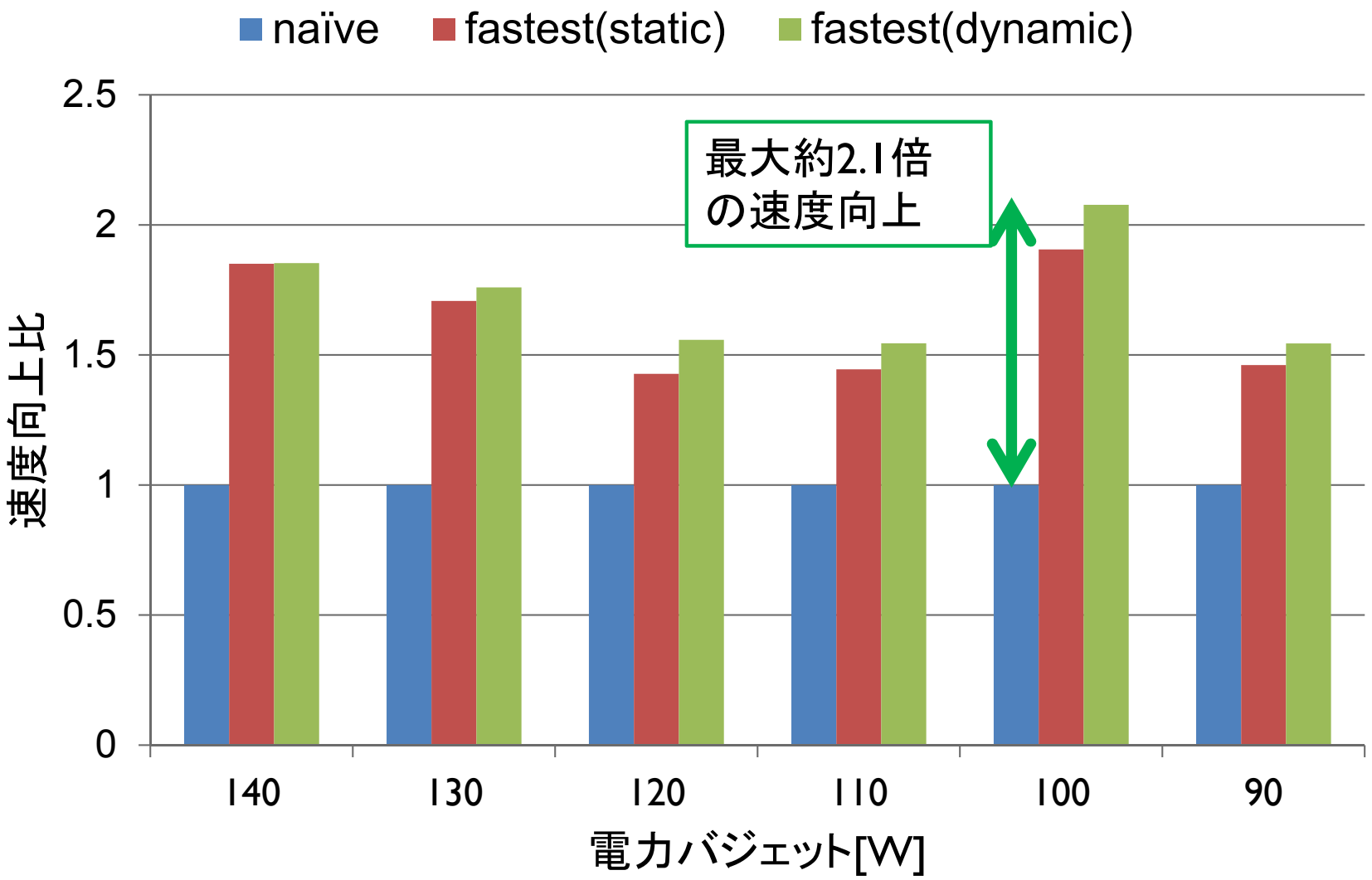
- naive:CPUとメモリの定格電力の比率で電力を分配
- fastest:各電力バジェット毎の最も速い実行時間



CPU・メモリ間の電力配分実験結果(2/2)

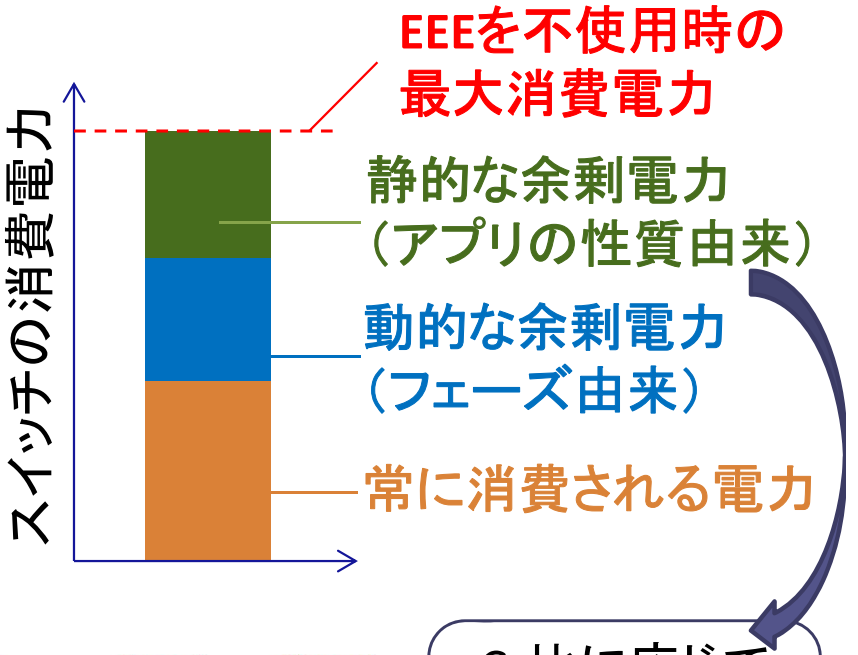
出典:[吉田2013]

- naïveに対する電力配分最適時による速度向上

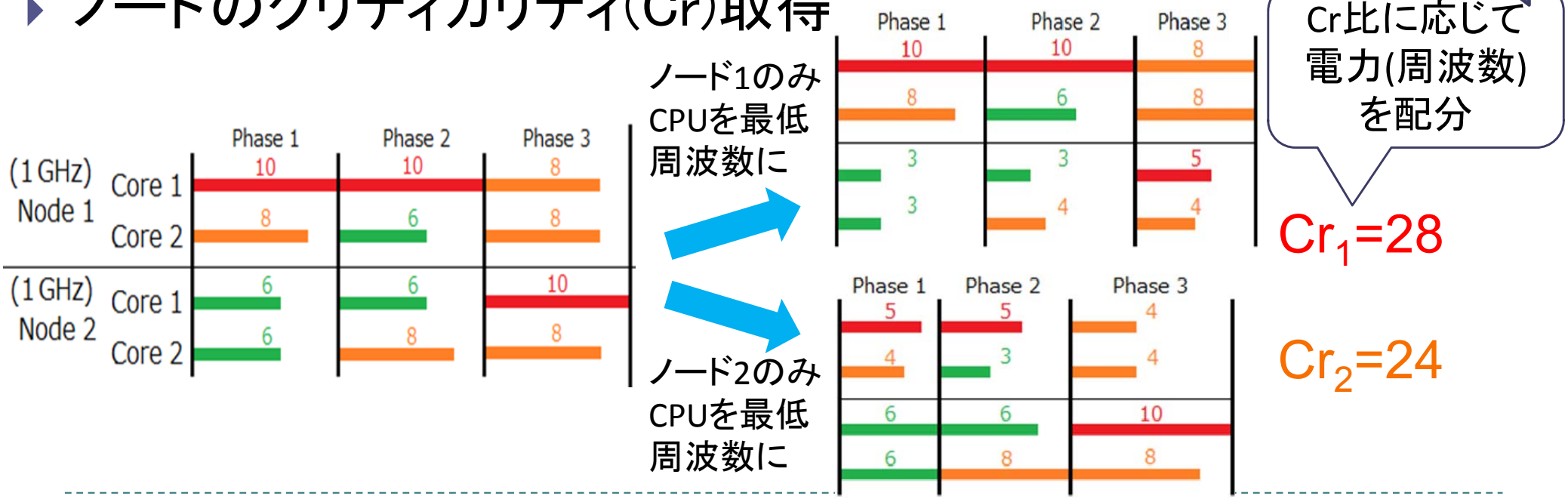


プロセッサ-ネットワーク間の電力分配法 出典:[會田2013]

- Energy Efficient Ethernet (EEE)
  - インタフェース上にデータが流れない期間にリンクを省電力モードへ移行
  - タイムアウトで省電力モード
  - オンデマンドで復帰



- ノードのクリティカリティ(Cr)取得

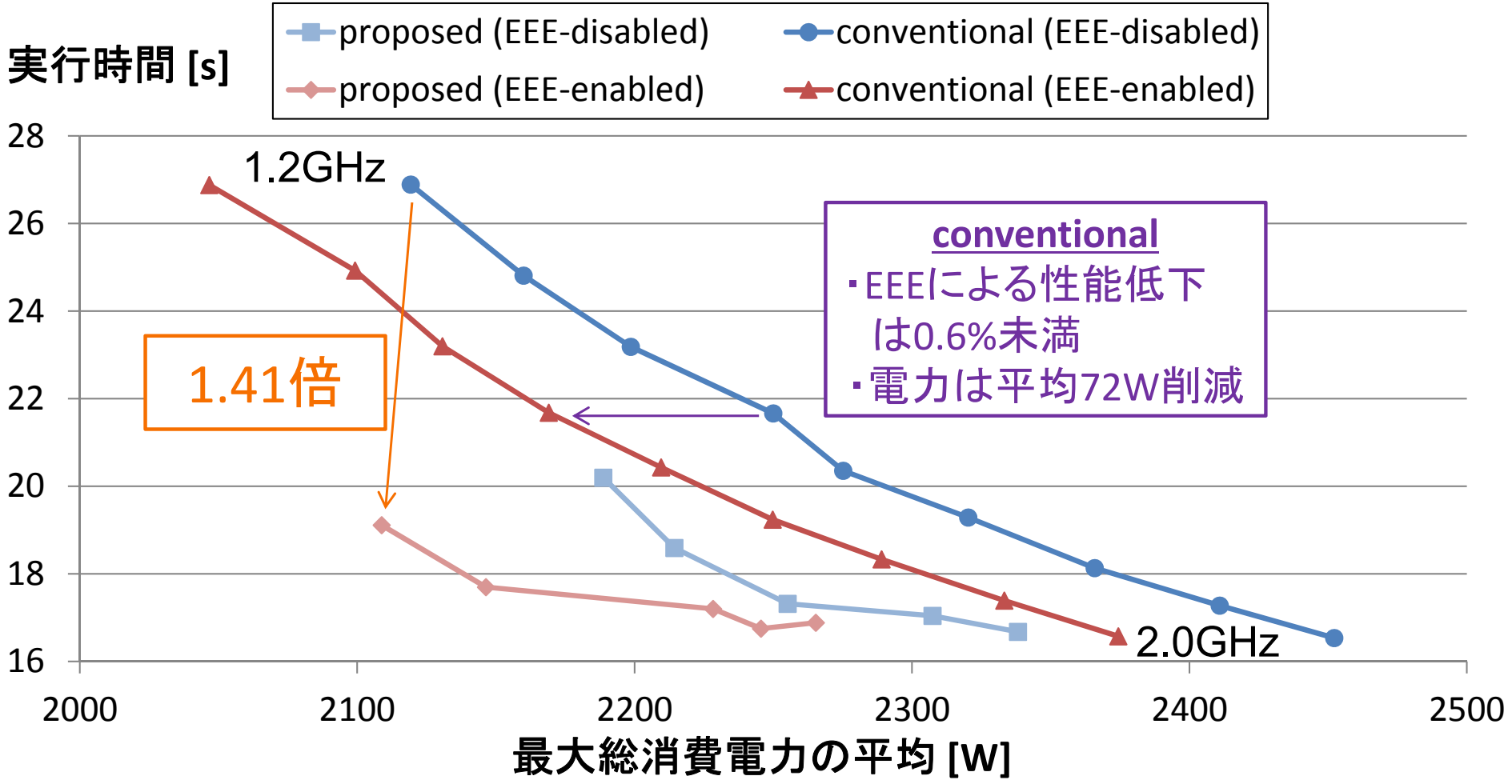


プロセッサ-ネットワーク間の電力配分実験結果

- ネットワークの静的な余剰電力を利用することで1.4倍の性能向上を実現

出典:[會田2013]

ノード: Dell PowerEdge r620 x 16
スイッチ:Dell PowerConnect 8132 x 3
- 10GbE:24ポート, EEE対応)
- Fat-Tree構成



本チュートリアルの構成

- 高性能計算機の消費電力トレンド
- 計算機システムにおける電力消費の基礎
- 省電力・省エネ技術
  - Dark-silicon問題とプロセッサの省電力・省エネ化技術
  - メモリの省電力化技術
  - インターコネクションネットワークの省電力化技術
  - システムソフトウェアレベルでの電力制御技術
- 将来展望

他の省電力技術と国内の研究動向

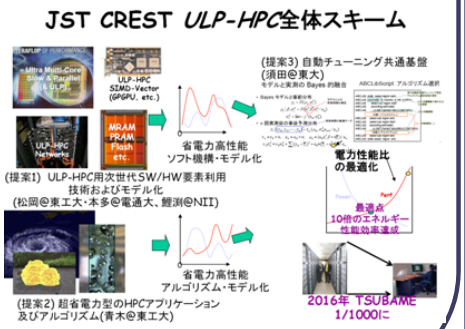
- 本チュートリアルでカバーできなかった技術分野
  - 省電力アルゴリズム/数値計算ライブラリ/自動チューニング
  - ファシリティ、冷却技術、・・・
- 国内の研究動向

超低電力化技術によるディペンダブルメガスケールコンピューティング（代表：中島浩教授）

- メガスケール計算実現のための要素技術
  - HW/SW協調による低電力化技術
  - SW主導のディペンダブル技術
  - グリッド/P2Pに基づくプログラミング技術
- Scale-outクラスタシステムの実証実験

ULP-HPC（代表：松岡聡教授）

- 省電力を目指した次世代HPC HW/SW技術
  - 新システムアーキテクチャ、モデル化
  - SW要素技術
  - グリーンスパコン
- 省電力HPCアプリ・アルゴリズム
- 電力性能比最適化自動チューニング



PomPP CREST（代表：近藤）

2001 ’06 ’07 ’12

今後も省電力を意識した継続的な研究開発が重要

Beyond 1 ExaFlops

- 将来的に2pj/DP-Flopが実現したとしても
  - 20MWの電力制約では10ExaFlopsが限界
  - さらなる高性能計算の進展にはブレークスルーが必要
- More Moore/More than Moore技術
  - SoC、SiP、3次元実装技術、・・・
  - RF、不揮発性メモリ、Photonics、MEMS、・・・
- さらなるブレークスルーへの期待: Post Moore技術
  - CMOSに変わる新デバイス
    - SFQ(単一磁束量子)回路、原子スイッチ、ナノエレクトロニクス、・・・
  - 新計算原理
    - 量子情報処理、Brain-inspired Computing、・・・

まとめ

- エクストリームスケールコンピューティングの実現には省電力化技術が必要不可欠
  - 2020年前後に20MWでExaFlopsの実現は簡単ではない
  - プロセッサ・メモリ・インターコネクトを含む全コンポーネントの省電力化を考える必要あり
  - ソフトウェア、システム管理レベルの省電力化技術も重要
- 今後の展望
  - 科学技術発展のために計算機の高性能化≒省電力化は重要
  - 日本の強みを生かす継続的な技術開発サイクルに期待
  - ポスト・ムーア時代のスパコンに向けた検討も開始する時期に！

参考文献 (1/5)

- [DOE2014] DOE ASCAS Subcommittee, “Top ten Exascale Research Challenges”, DOE ASCAS Subcommittee Report, Feb. 2014.
- [Wallance2013] S. Wallace, V. Vishwanath, S. Coghlan, Z. Lan, M. Papka, “Measuring Power Consumption on IBM Blue Gene/Q”, Proc. 9th HPPAC, May 2013.
- [Stevens2009] R. Stevens, et. al. “Scientific Grand Challenges: Architectures and Technology for Extreme Scale Computing”, Technical report, ASCR Scientific Grand Challenges Workshop Series, Dec. 2009.
- [Nair2014] R. Nair, “Active Memory Cube: A Processing-in-Memory Approach to Power Efficiency in Exascale Systems”, WoNDP-2, Dec. 2014.
- [Kim2003] N.S. Kim, “Leakage Current: Moore’s Law Meets Static Power”, IEEE Computer Vol.36, Issue 12 , pp.68-75, Dec. 2003.
- [Kogge2008] P. Kogge, et. al., “ExaScale Computing Study, Technology Challenges in Achieving Exascale Systems”, IPTO tech. report TR-2008-13, DARPA, Sep. 2008.
- [Borkar2013] S. Borkar, “Exascale Computing – a fact or a fiction?”, IPDPS2013 Keynote, May 2013.
- [Taylor2013] M. B. Taylor, “A Landscape of the New Dark Silicon Design Regime ”, IEEE Micro, Vol.33, Issue 5, pp.8-19, Sep-Oct. 2013.
- [Taylor2012] M. B. Taylor, “Is Dark Silicon Useful? Harnessing the Four Horsemen of the Coming Dark Silicon Apocalypse”, DAC, 2012.
- [Kanter2012] D. Kanter, “Computational Efficiency for CPUs and GPUs in 2012”, real world technologies, 2012. (http://www.realworldtech.com/compute-efficiency-2012/)
- [追永2013] 追永, “FXシリーズの今後の取り組みについて”, SS研HPCフォーラム2013，2013年8月.

参考文献 (2/5)

- [ARM2013] ARM, “big.LITTLE Technology: The Future of Mobile”, ARM White Paper, 2013.
- [Jeff2012] B. Jeff, “Advances in big.LITTLE Technology for Power and Energy Savings”, ARM White Paper, Sep. 2012.
- [Sakurai2011] T. Sakurai, “Designing Ultra-Low Voltage logic”. Proc. ISLPED’11, pp57-58, Aug. 2011.
- [Kondo2014] M. Kondo, et al., “Design and Evaluation of Fine-Grained Power-Gating for Embedded Microprocessors", Design, Automation and Test in Europe Conference and Exhibition (DATE 2014), March 2014.
- [Venkatesh2010] G. Venkatesh, et al., “Conservation Cores: Reducing the Energy of Mature Computations”, ASPLOSXV, pp.205-208, March 2010.
- [Putnam-ISCA2014] A. Putnam et al., “A Reconfigurable Fabric for Accelerating Large-Scale Datacenter Services”, ISCA’14, pp.13-24, June 2014.
- [Putnam-HotChips2014] A. Putnam et al., “ Large-Scale Reconfigurable Computing in a Microsoft Datacenter”, Hot Chips 26, Aug. 2014.
- [Han2013] J. Han and M. Orshansky, “Approximate Computing: An Emerging Paradigm For Energy-Efficient Design“, ETS’13 Embedded Tutorial, May 2013.
- [Allan2014] G. Allan, “The Future of DRAM”, Synopsys Blogs, May 2014. (http://blogs.synopsys.com/committedtomemory/2014/05/28/the-future-of-dram/)
- [Brennan2014] B. Brennan, “New Directions in Memory Architecture”, The Memory Forum, June 2014.
- [Black2013] B Black, “Die Stacking is Happening”, MICRO-46 Keynote, Dec. 2013.

参考文献 (3/5)

- [Baxter2014] R. Baxter, “Developing Scalable and Resilient Memory Systems with Hybrid Memory Cube”, Data Center Conference 2014, Feb. 2014.
- [SPARC64XIfx2014] Next Generation Technical Computing Unit, Fujitsu Limited, “FUJITSU Supercomputer PRIMEHPC FX100 Evolution to the Next Generation”, White paper, 2014.
- [Pawlowski 2011] J. T. Pawlowski, “Hybrid Memory Cube (HMC)”, Hot Chips23, Aug. 2011.
- [HMC2013] Hybrid Memory Cube Consortium, “Hybrid Memory Cube Specification 1.0”, 2013.
- [Ando2012] K. Ando, et al., “Roles of Non-Volatile Devices in Future Computer System: Normally-off Computer”, Energy-Aware Systems and Networking for Sustainable Initiatives, edited by N. Kaabouch and W.-C. Hu, published by IGI Global, June, 2012.
- [中村2014] 中村, “ノーマリーオフコンピューティング基盤技術開発プロジェクト公開シンポジウム プロジェクト概況報告”,2014年6月.
- [Izumi2013] S. Izumi et. al.: “A 14 μA ECG processor with robust heart rate monitor for a wearable healthcare system“, IEEE ESSCIRC 2013, pp. 145-148, 2013.
- [Nakada2015] T. Nakada: “Normally-Off Computing: Synergy of New Non-Volatile Memories and Aggressive Power Management ”, ASP-DAC Tutorial, 2015.
- [Hosogaya2008] Y. Hosogaya, “Performance Evaluation of Parallel Applications on Next Generation Memory Architecture with Power-Aware Paging Method”, IPDPS2008, April 2008.
- [Borkar-JLT2013] S. Borkar, “Role of Interconnects in the Future of Computing”, Journal of Lightwave Technology, Vol. 31, Iss. 24, pp. 3927–3933, Dec. 2013.
- [Healey2010] A. Healey “Introduction to Energy Efficient Ethernet”, Joint T11.2/T11.3 Ad Hoc Meeting March 2010.

参考文献 (4/5)

- [Miwa2013] S. Miwa, “Performance Estimation for High Performance Computing Systems with Energy Efficient Ethernet Technology”, EnA-HPC’13, pp.1-9, Aug. 2013.
- [Welch2010] B. Welch, “Silicon Photonics for HPC Interconnects, 100 Gbps and Beyond”, SC’10 Disruptive Technologies, 2010.
- [Bergman2011] K. Bergman, "Silicon photonics for exascale systems”, 2011 Workshop on Architectures I: Exascale and Beyond: Gaps in Research, Gaps in our Thinking, Aug 2011.
- [Bergman-OFC14] K. Bergman, et al., "Silicon photonics for exascale systems”, Optical Fiber Communications Conference and Exhibition (OFC) Tutorial, March 2014.
- [HP2014] HP Labs, “The Machine: The Future of Computing”, 2014. (http://www.hpl.hp.com/research/systems-research/themachine/)
- [Intel2012] Intel Power Governor, http://software.intel.com/en-us/articles/intel-power-governor, July 2012.
- [カオ2013] カオ，和田，近藤，本多，“RAPLインタフェースを用いたHPCシステムの消費電力モデリングと電力評価”，情報処理学会HPC研究会，2013年10月．
- [IBM2014] http://www-01.ibm.com/support/knowledgecenter/SSLQVT_6.3.0/com.ibm.itmem.doc_6.3/aemagent63_user221.htm%23wq412
- [Schott2013] B. Schott, “Energy aware scheduling in complex heterogeneous environments”, HPCN-Workshop 2013, 2013.
- [Auweter2013] A. Auweter et al., “A case study of energy aware scheduling on SuperMUC”, ISC’14, LNCS vol.8488, pp.394–409, 2014.
- [Sarood2014] O. Sarood et al., “Maximizing Throughput of Over-provisioned HPC Data Centers under a Strict Power Budget”, SC’14, pp.807–818, Nov. 2014.

参考文献 (5/5)

- [Etinsk2012] M. Etinski et al., “Parallel job scheduling for power constrained hpc systems”, J. of Parallel Computing, Vol.38 Issue 12, pp.615-630, Dec. 2012.
- [Geogiou2014] Y. Georgiou et al., “Energy Accounting and Control with SLURM Resource and Job Management System”, ICDCN2014, LNCS Vol.8314, pp.96-118, Jan. 2014.
- [Das2008] R. Das et al, “Autonomic Multi-Agent Management of Power and Performance in Data Centers”, AAMAS 2008, pp.107-114, May 2008.
- [Urgaonkar2013] R. Urgaonkar et al., “Optimal power cost management using stored energy in data centers”, SIGMETRICS '11, pp.221-232, 2011.
- [Chen2010] Y. Chen et al. “Integrated management of application performance, power and cooling in data centers”, NOMS2010, pp.615-622, 2010.
- [Leverich2009] J. Leverich et al., “Power Management of Datacenter Workloads Using Per-Core Power Gating”, Computer Architecture Letters, Vol.8, Issue 2, pp.48-51, 2009.
- [PowerAPI2014] Sandia Report , “High Performance Computing – Power Application Programming Interface Specification Version 1.0”, SAND2014-17061, Aug. 2014.
- [Sarood2013] O. Sarood, et al., “Optimizing power allocation to CPU and memory subsystems in overprovisioned HPC systems”, Cluster2013, pp.1-8, Sep. 2013.
- [吉田2013] 吉田，佐々木，深沢，稲富，上田，井上，青柳，“CPUと主記憶への電力バジェット配分を考慮したHPCアプリケーションの性能評価”，情報処理学会HPC研究会，2013年10月．
- [會田2013] 會田，三輪，中村，“電力制約下におけるCPUとネットワークの電力制御協調手法，SWoPP2013，2013年7月．