

*Energy-Efficient SuperComputers Workshop*

*Date: Mon, June 14, 2007*

*Place: Meeting Room at SGI Japan*

*Title:*

*“Can 10 Peta-Scale Computing be Built up  
Without Cool Chips and Cool Software?”*

*Tadao Nakamura*

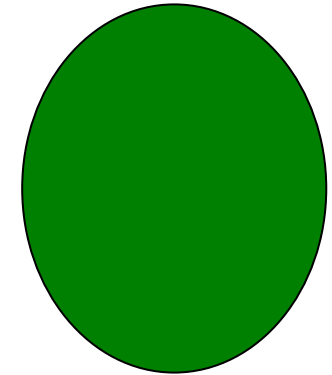
*Professorial Fellow of Imperial College, London*

*(to be inducted)*

*Visiting Full Professor of Stanford University*

## 昔の設計・製造対象

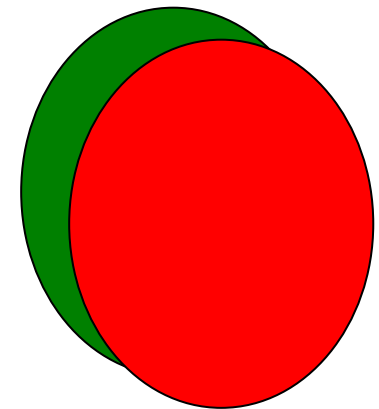
- **ハードウェア**のみ
- **物理で目に見える**



ハードウェア

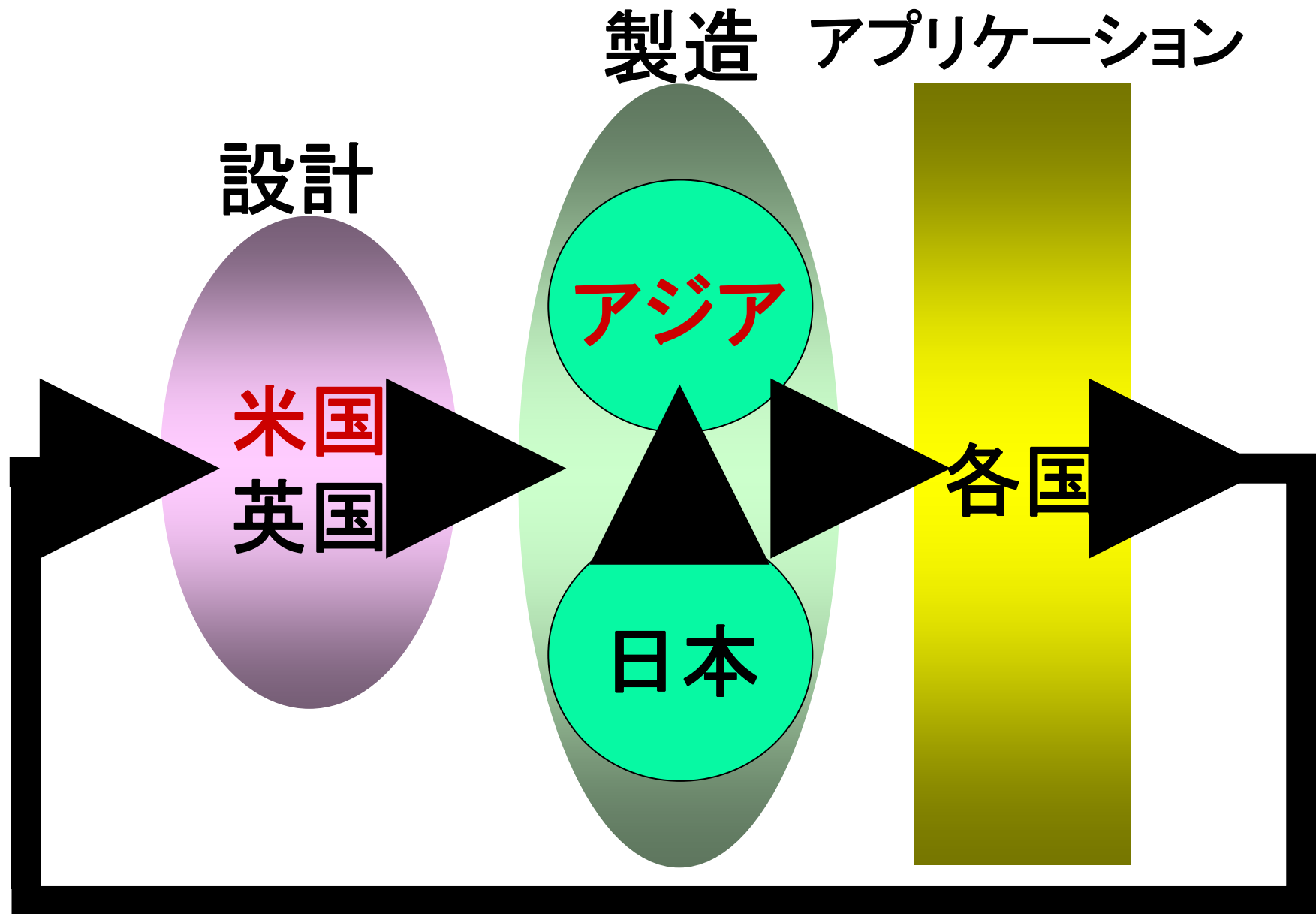
## 今の設計・製造対象

- ハードウェア上の  
**ソフトウェア**が主
- **論理で目に見えない**



ソフトウェア

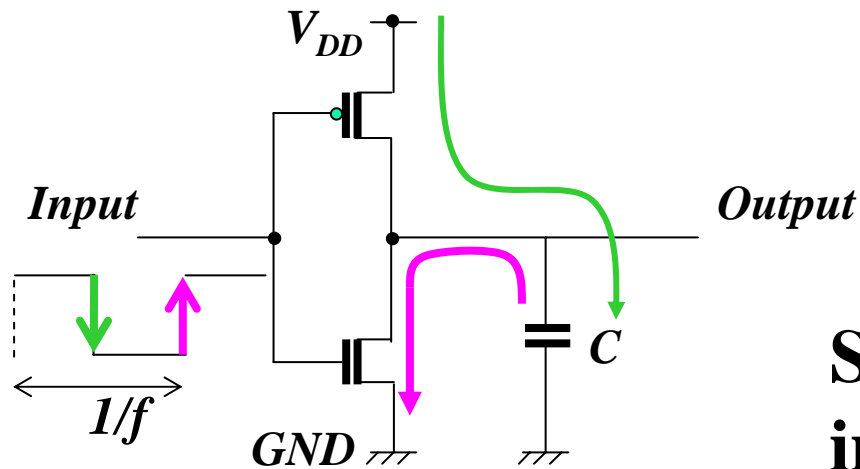
# コンピュータに関する大問題



# *Dynamic Power Consumption of a CMOS Transistor (Gate) - Switching Power -*

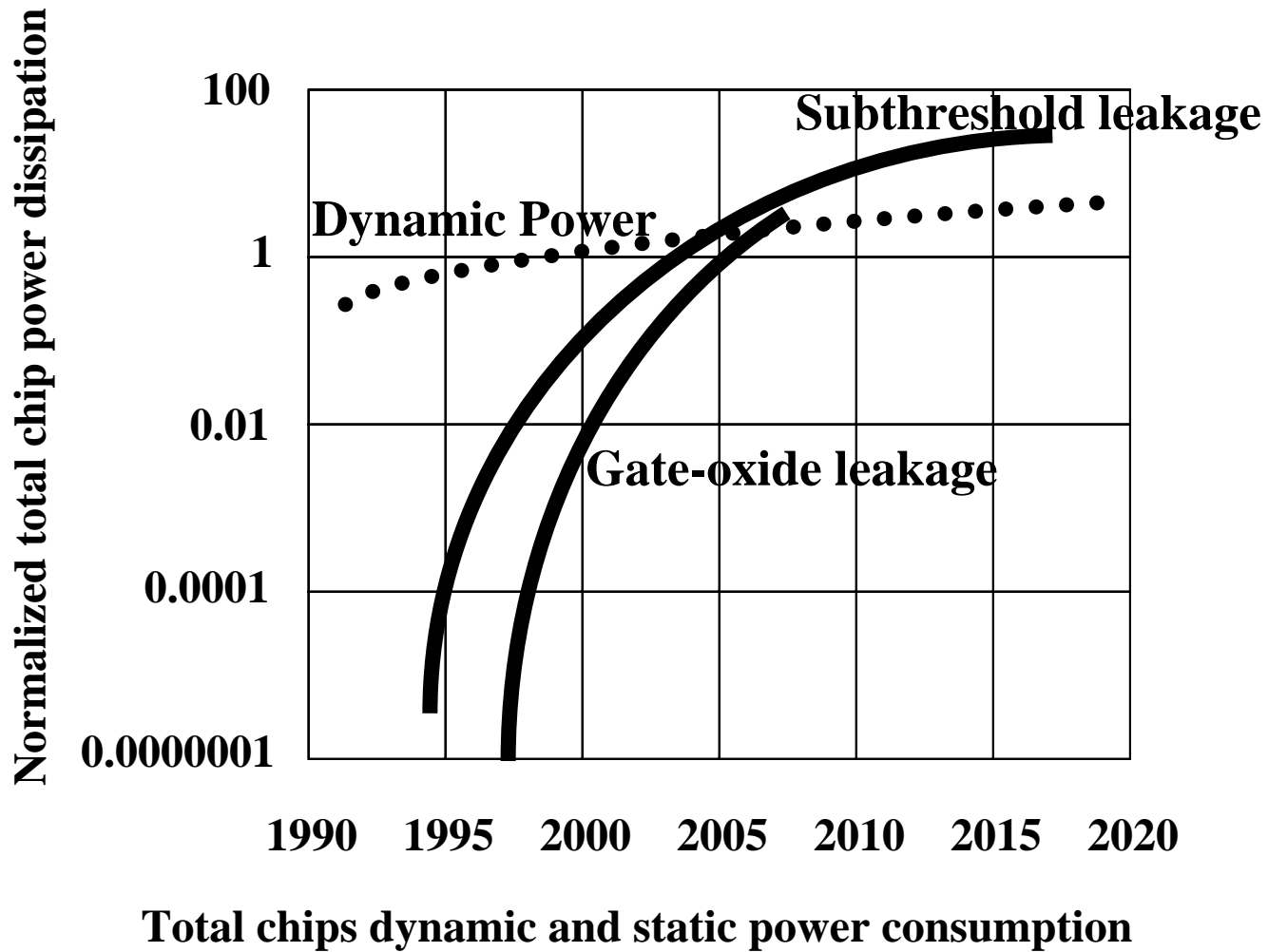
**Switching power in  
one clock cycle**

$$P = f c V_{DD}^2$$



**Switching power at a trigger  
in one clock cycle**

$$P = 1/2 * f c V_{DD}^2$$



From IEEE Computer, p. 69, December 2003

# *Dynamic Switching Power Consumption of a Microprocessor*

$P = \frac{1}{2} \alpha f C V^2$  : The Whole Power [**Joules per second**] of a Microprocessor During One Second

$C = \sum_x \sum_y c$  : The Whole Capacitance of a Microprocessor

$f$  : Clock Frequency [MHz-GHz]

# *Source of Static Power Consumption of a Microprocessor*

$$I_{leak} = I_{sub} + I_{ox}$$

$I_{sub}$  : *Subthreshold Leakage as a Function of Threshold Voltage and Supply Voltage*

$I_{ox}$  : *Gate-Oxide Leakage as a Function of Supply Voltage*

# *Overall Power Consumption*

$$P = \underbrace{\frac{1}{2} \alpha f C V^2}_{\text{Dynamic Power}} + \underbrace{V I_{\text{leak}}}_{\text{Static Power}}$$

*Electrical Energy Consumption*

*To Thermal Energy (HEAT)*

*⇒Microprocessors' Energy Efficiency*

*⇒Related to Batteries' Lifetime*

*Thermal Energy Dissipation*

*⇒Heat Problem related to Reliability of Processors*

*In order to process an application, the most important  
is not POWER but ENERGY*

## **THE MOST IMPORTANT METRIC for Supercomputing**

***FLOPS / POWER***

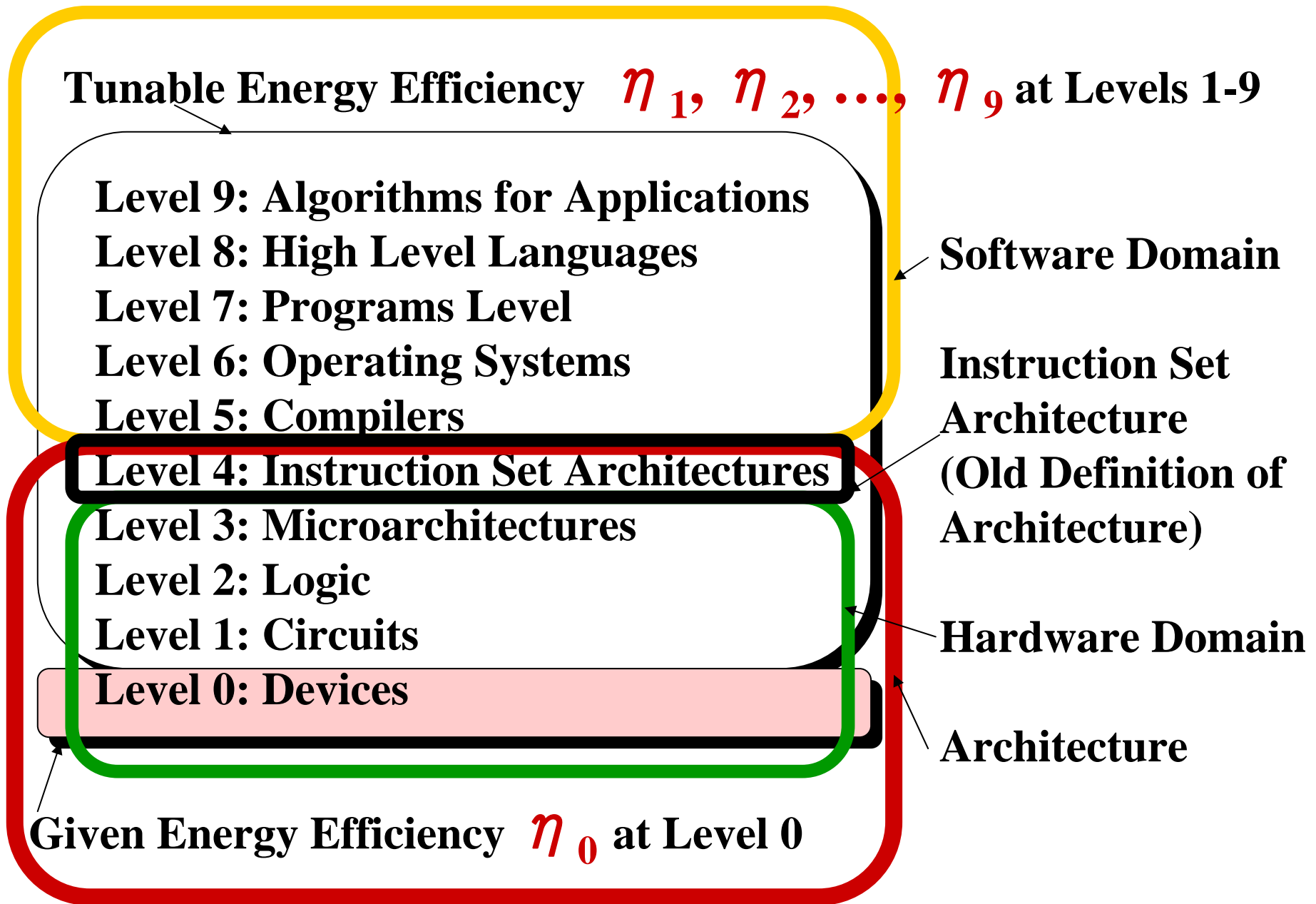
***= Floating Point Operations PER second / Joules PER second***

***= Floating Point Operations PER Joule***

# Total Energy Efficiency $\eta$ ( $0 \leq \eta \leq 1$ )

$$\begin{aligned}\eta &= E_{\text{net}} / (E_{\text{net}} + E_{\text{overhead}}) \\ &= E_{\text{net}} / E_{\text{application}}\end{aligned}$$

*where net energy  $E_{\text{net}}$  without overhead energy  $E_{\text{overhead}}$  means ideal energy  $E_{\text{ideal}}$  to obtain the results of an application in the shortest processing time. Here an application corresponds to a job. Therefore, the reasonable case is totally at  $\alpha = 1$  and  $\eta = 1$ .*



**Energy Efficiency at All the Levels**

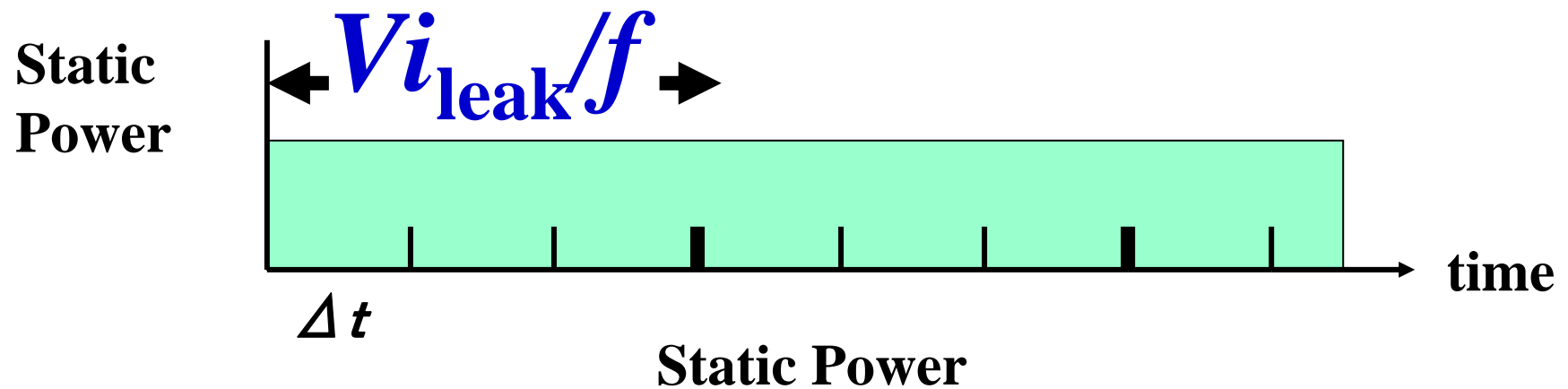
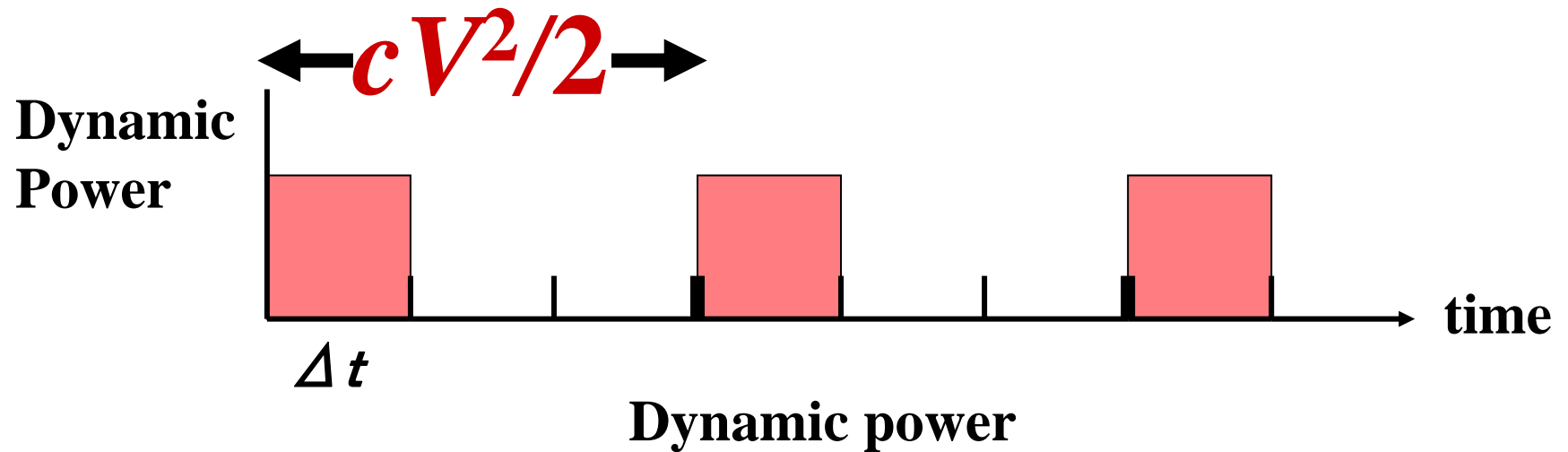
# *Overall Power Consumption*

$$P = \frac{1}{2} \alpha f C V^2 + V I_{leak}$$

*Dynamic Power*                      *Static Power*

- All of the letters except for  $\alpha$  are basically depending on CMOS technologies. The energy efficiency of a microprocessor is therefore basically related to level 0.
- $\alpha$  is related to each energy efficiency at level 1 to level 9. This is changeable/tunable based on the efforts of low energy consumption.
- $\alpha = 1$  means full activity of the gate, namely the switching rate of 1.

*Case of  $g=3$  and then  $f = 1/(g \Delta t)$ , where  $g$  is the number of gates within a stage of the pipeline we have here in RTL.*



# *One Gate Energy Consumption*

$$e_{\text{one gate}} = cV^2/2 + Vi_{\text{leak}}/f$$

*where*

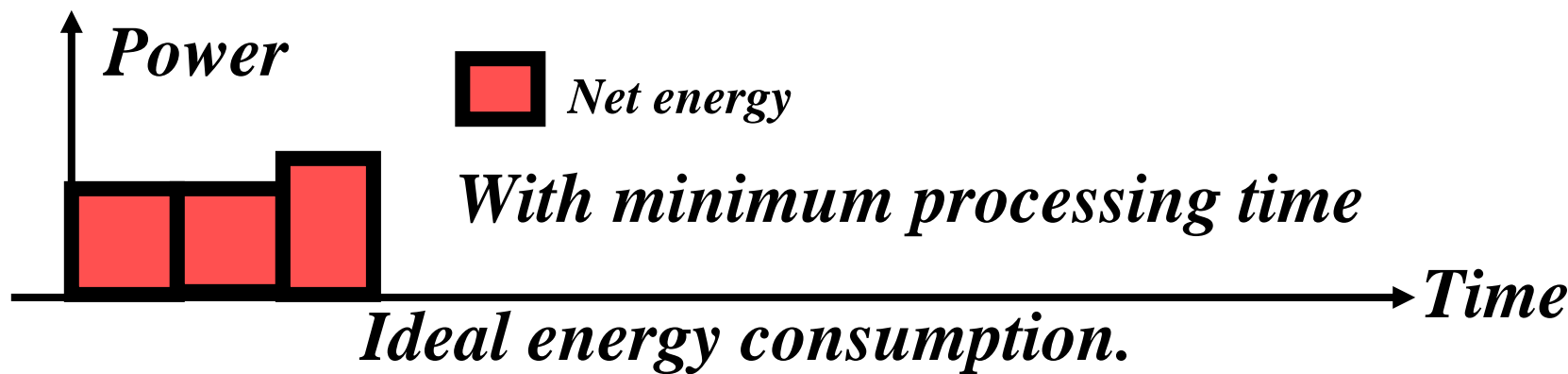
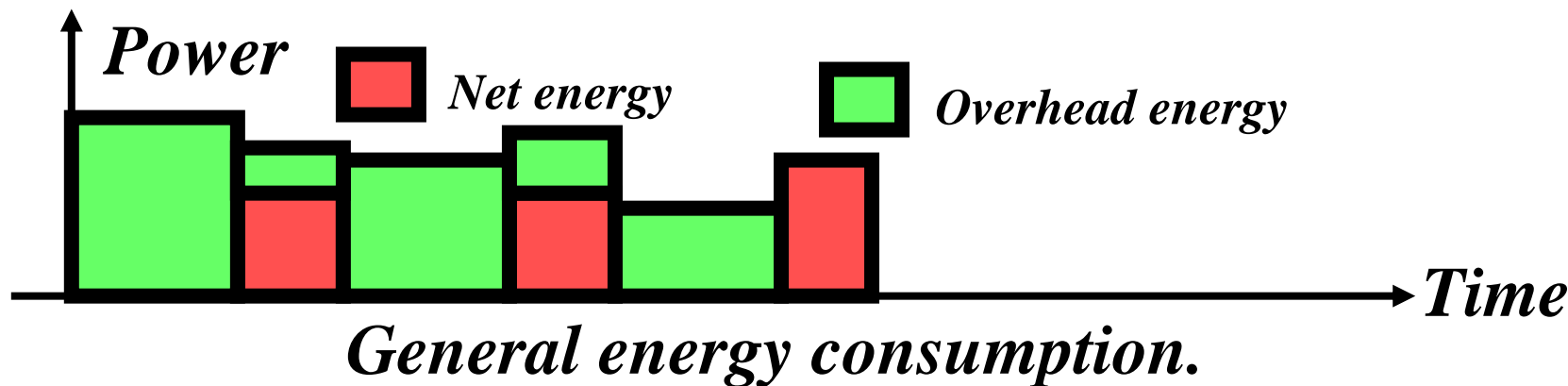
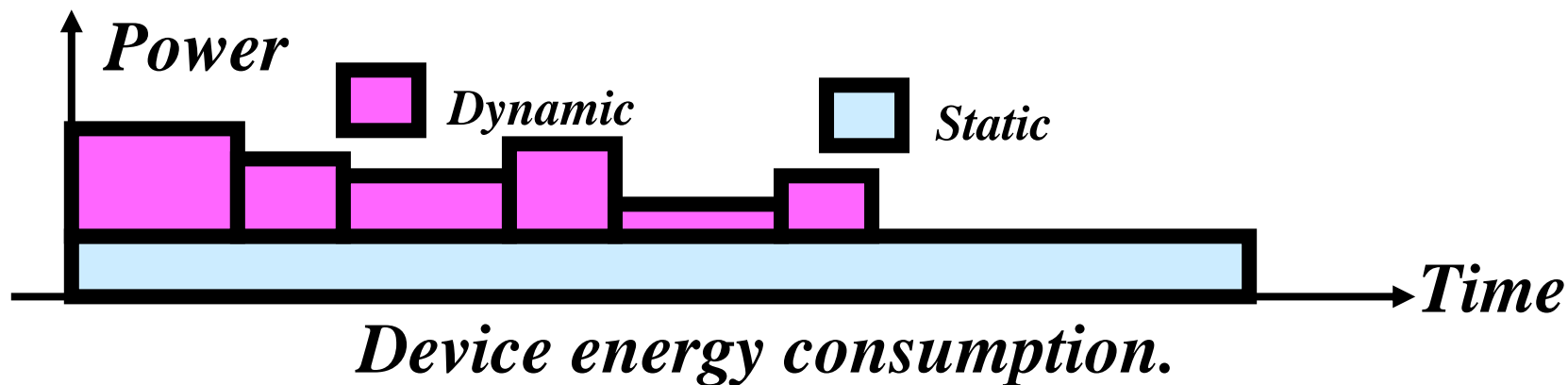
*$cV^2/2$  is an Dynamic Energy and  $Vi_{\text{leak}}$  is a Static Power of a gate during one switch within a pipeline stage.*

*$i_{\text{leak}}$  is leakage current of the gate and  $f$  is a frequency of the stage activity.*

## Consumed Net Energy $E_{\text{ideal}}$

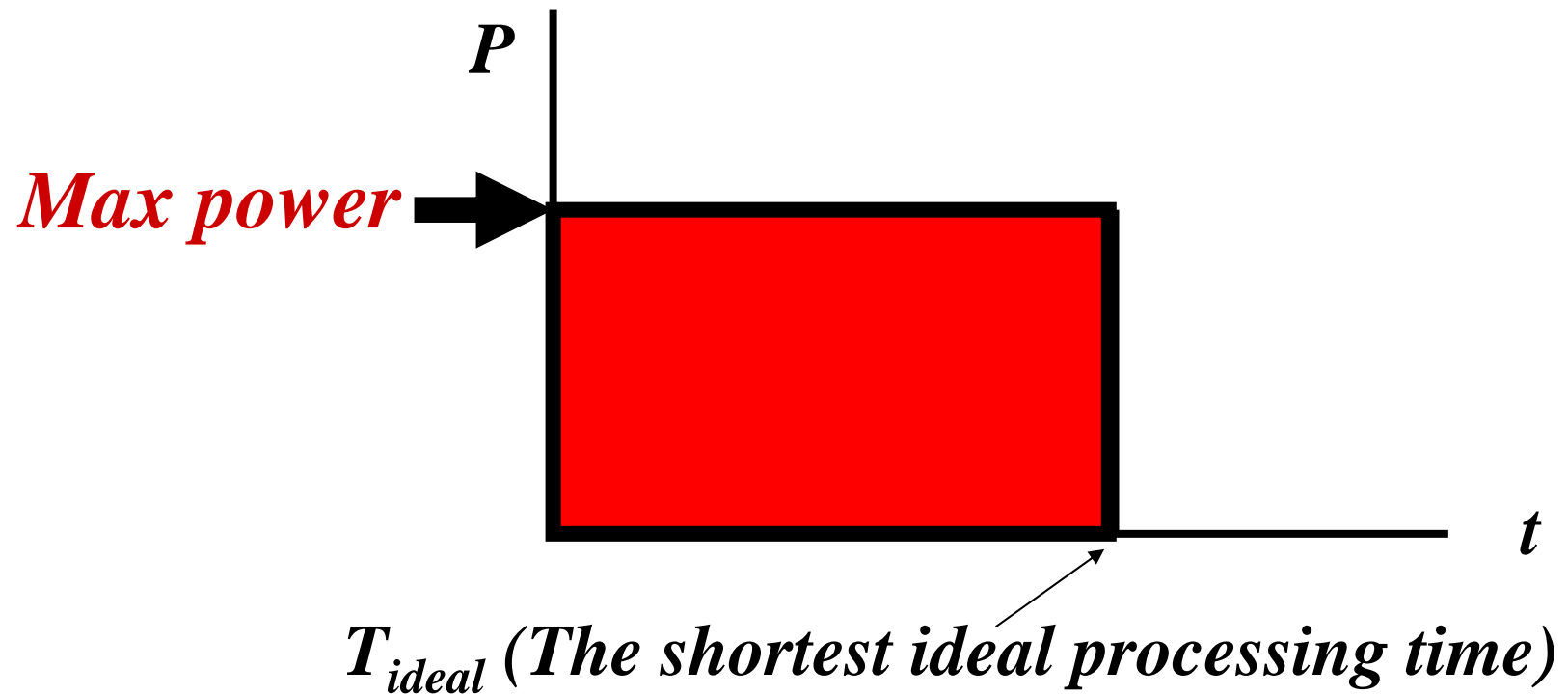
$$e \sum_x \sum_y \sum_n^{n_{\text{ideal}}} u(x, y, n\Delta t) + VI_{\text{leak}} * n_{\text{ideal}} \Delta t = \eta * E_{\text{application}}$$

- $e (= cV^2/2)$  : Energy consumed by one switch in a CMOS Tr.
- $u(x, y, n\Delta t)$  : **Switching activity function** of CMOS Tr's at space(x, y)  
whose value is either 0 or 1.
- $f = 1/(g\Delta t)$  : Clock frequency where g is the number of gates (CMOS Tr's) within a pipeline stage in the microarchitecture.
- $n_{\text{ideal}}$  : The number of switchings for IDEALLY processing an application.
- $\Delta t$  : A switching time of the CMOS Tr (=Gate delay).
- $\eta$  : Efficiency of the total energy for an application completion.
- $E_{\text{application}}$  : Consumed energy [Joules] during an application including overhead energy.



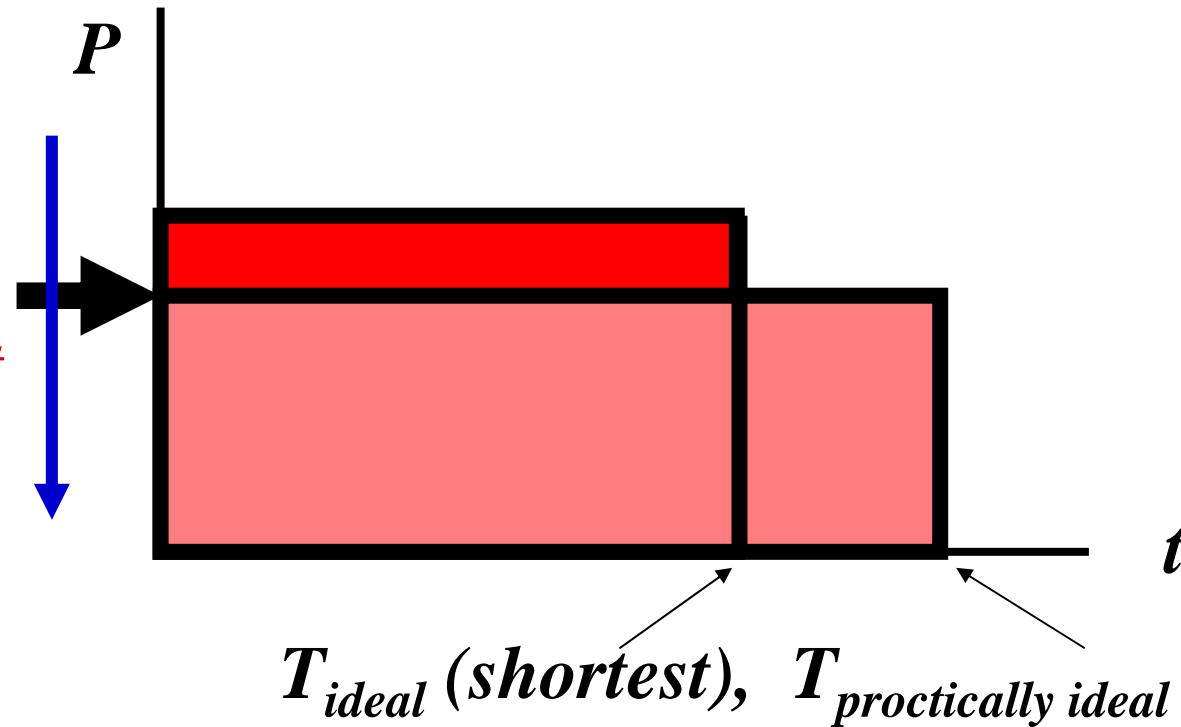
## Case of all gates active in a die during an application

*If net energy  $E_{ideal}$  for an application processing is with allowable max power, the ending time  $T_{ideal}$  means the shortest ideal processing time of the application.*



*Even though net energy  $E_{ideal}$  is obtained with the shortest processing time, maximum power must be lowered keeping the net energy constant unless the temperature breaks the device.*

*Max power must be lowered if it is not guaranteed in terms of reliability*



## *Algorithm for Obtaining the Ideal Chips*

- (1) Device (Level 0) technology is given, and then the energy efficiency is also given.**
- (2) Hardware Domain (Levels 1-3) is designed and tuned under the device's dynamic energy and static energy.**
- (3) Instruction set architecture (Level 4) is designed and tuned based on the hardware in (2).**
- (4) Software domain (Level 5-9) is designed and tuned based on the instruction set architecture in (3).**

# Parallel Processor is Lower in Dynamic Power

<<<Pipelined Processor>>>

$$\begin{aligned} P_{d\text{-pipelined}} &= P_{PE} \\ &= \frac{1}{2} f(C + \alpha) V^2 \\ &(> P_{d\text{-parallel}}) \end{aligned}$$

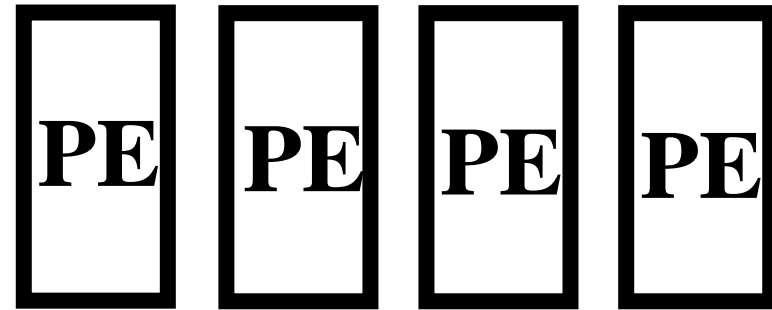
$\alpha$   
Overhead Circuitry for  
Higher Clock Frequency



$$V \geq V$$

<<<Parallel Processor>>>

$$\begin{aligned} P_{d\text{-parallel}} &= 4P_{PE} \\ &= 4\left(\frac{1}{2} (f/4) CV^2\right) \\ &= \frac{1}{2} fCV^2 \end{aligned}$$



## *Parallel Processor is Lower in Temperature*

### **Pipelined Processor**

$$P_{uni} = \frac{1}{2} f(C + \alpha) V^2$$

$$\begin{aligned} \Delta t_{uni} &= \frac{1}{2} f(C + \alpha) V^2 T / (C + \alpha) \gamma S \\ &= \frac{1}{2} f V^2 T / \gamma S \end{aligned}$$

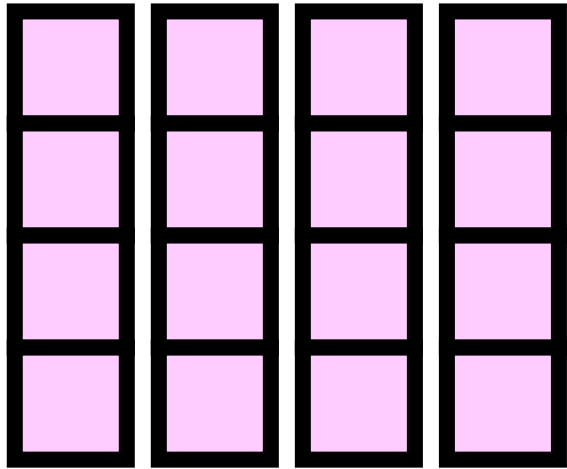
### **Parallel Processor**

$$P_{array} = 4 * \frac{1}{2} (f/4) C V^2$$

$$\Delta t_{array} = \frac{1}{2} f V^2 T / 4 \gamma S$$

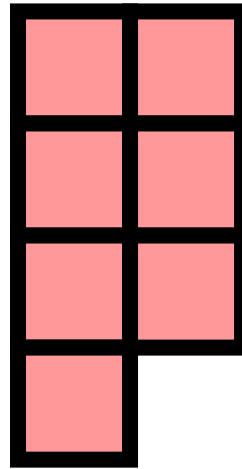
$$\Delta t_{uni} > 4 * \Delta t_{array}$$

*Parallel Processor*



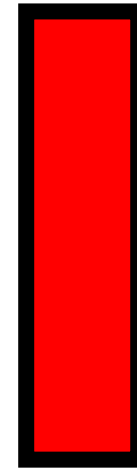
$\frac{1}{4} * f$

*Pipelined Processor*



$f$

*Serial Processor*

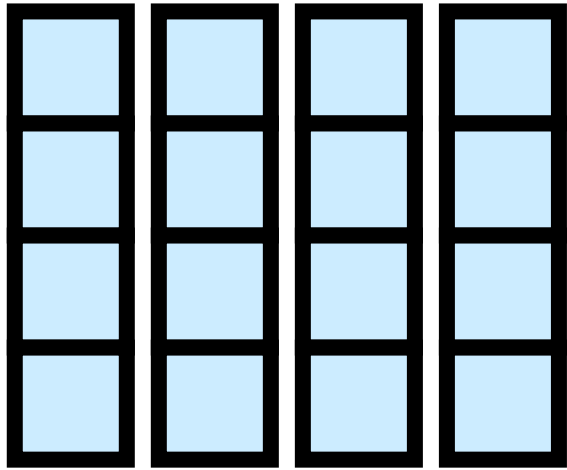


$4f$

$$P_{d\text{-parallel}} < P_{d\text{-pipelined}} < P_{d\text{-serial}}$$

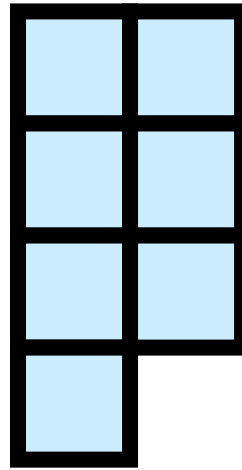
*Dynamic Power*

*Parallel Processor*



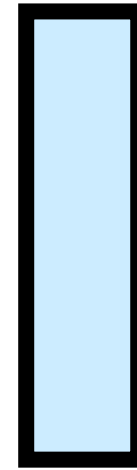
$$\frac{1}{4} * f$$

*Pipelined Processor*



$$f$$

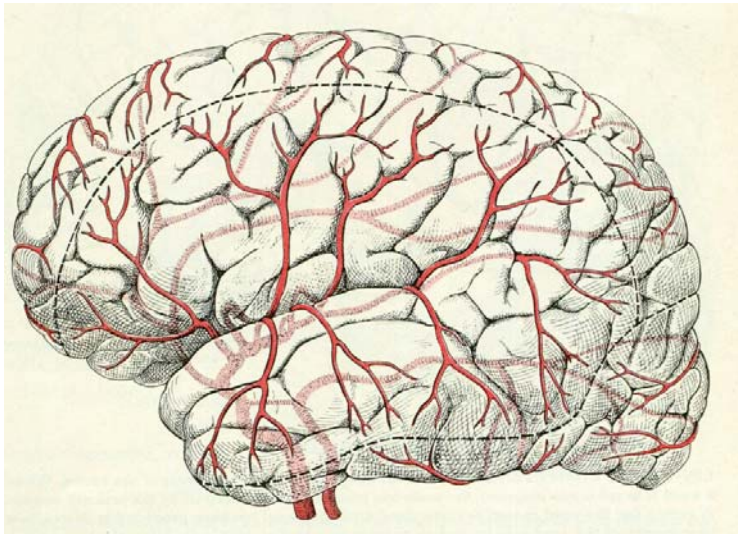
*Serial Processor*



$$4f$$

$$P_{d\text{-parallel}} > P_{d\text{-pipelined}} > P_{d\text{-serial}}$$

*Static Power*



*The Mechanism of Computing in the Human Brain is not so much von Neumann Machine as Quantum Computing, **I propose!***

*Before this, FPGA computing is based on Memory, which means that Computers consist of only memories!*

# *How To Build Up 10 Peta-Scale Supercomputers*



May, 8th 2007

**TOPS Systems Corp.**

# **How To Build Up 10 Peta-Scale Supercomputers**

**We must focus in the following three respects.**

- 1) Power/Energy Consumption**
- 2) Production of Software**
- 3) Reliability**

# Issue 1 : Reducing Power/Energy Consumption

## Earth Simulator Case

$$35.86 \text{ TFLOPS} / 5 \text{ MW} = 7.2 \text{ MFLOPS} / \text{W}$$

## BlueGene Case

$$280.6 \text{ TFLOPS} / 1.6 \text{ MW} = 175 \text{ MFLOPS} / \text{W}$$

If we obtain 10 Peta FLOPS using BlueGene Processors,

$$10000 / 280 * 1.6 = \mathbf{57 \text{ MW}}$$

If we estimate the utility using the rate 100,000,000 Yen / (MW \* Year),

it is 5,700,000,000 Yen / Year

## *Solution*

**If we have 1 Peta FLOPS / MW (= 1 GFLOPS / W) machine,**

$1000 / 175 = \mathbf{5.7 \text{ times}}$  the power performance of BlueGene is obtained and the power consumption of the 10 Peta-Scale machine is **“10 MW.”**

For 1 GFLOP/W, the **more number of small cores** per chip (low power high density), which has good balance between CPU performance and Memory / Network performance, is better than the less number of big cores with high dynamic power on both CPU and network.

## Issue 2 : Producing Parallel Programming Tool

In order to realize **10 Peta FLOPS**, we must provide **1,000,000 CPUs**, where the performance of a CPU is **10 GFLOPS / CPU**.

### *Solution:*

To manage these many number of CPUs, we have to produce a new Parallel Programming Tools to support massive parallelism.

Then, we have to **re-write** application software.

Which means, Basically we must rebuild its Software and development environment **from scratch**.

Ex) Application Software programming for 10 Peta FLOPS is something like, Mapping of “3-D structure of simulation target system” (Earth) onto “3-D network structure of 1,000,000 CPU nodes”.

## Issue 3 : How to improve the reliability

If the MTBF (Mean Time Between Failures) of a **10 GFLOPS CPU** is **10 Years**, the one of a 1,000,000 CPU supercomputer is **315 Seconds** because  $10 * 365 * 24 * 60 * 60 = 315 * 10^6$  and  $315 * 10^6 / 10^6 = 315$ .

### ***Solution:***

We must introduce Fault Tolerant System Architecture.

ex) Flexible network with reconfiguration to dynamically bypass faulty CPU nodes.

# Future Development

Technology Transfer between the 10 Peta-Scale Supercomputers and Home Information Appliances.

If we must have CPUs with high power performance, we must have embedded CPUs with high power performance, and vice versa.