GRAPE-DR

Jun Makino
Center for Computational Astrophysics
and
Division Theoretical Astronomy
National Astronomical Observatory of Japan

Talk structure

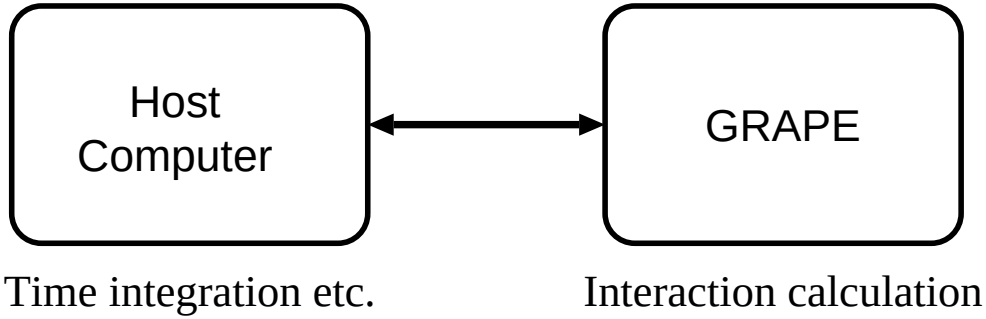
- GRAPE hardwares
  - GRAPE machines
  - GRAPE-DR
- How do they compare with GPGPU?
- GRAPE-DR project status

Short history of GRAPE

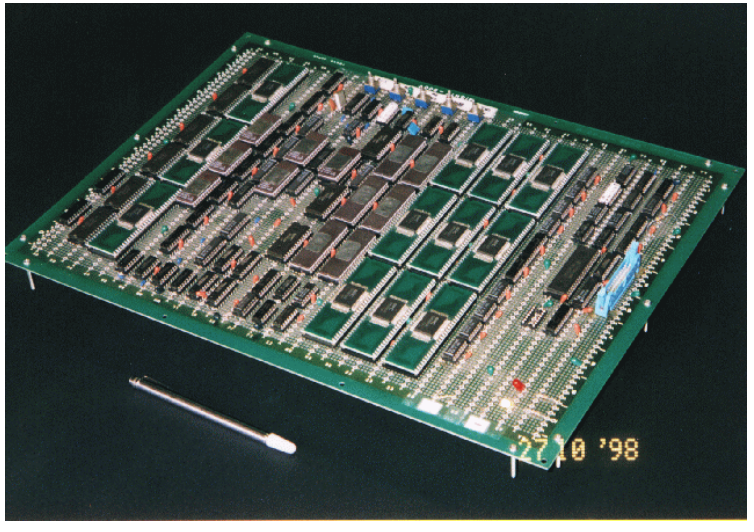
- Basic concept
- GRAPE-1 through 6

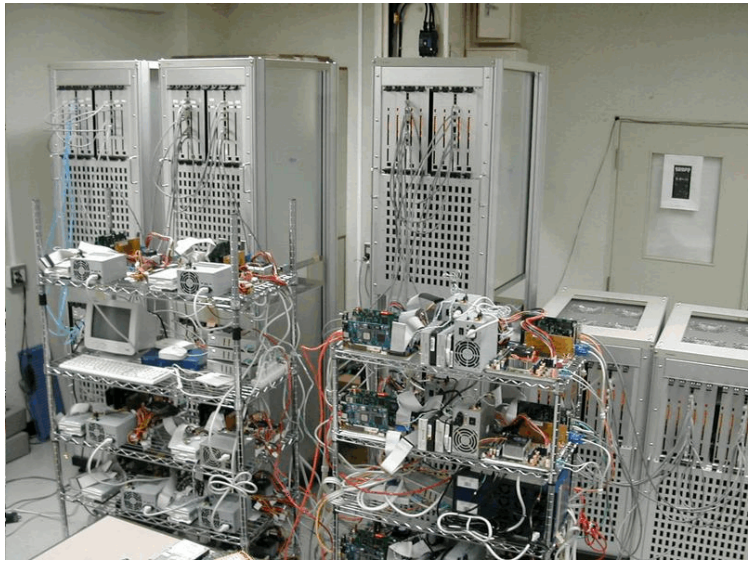
Basic concept

- **With N-body simulation, almost all calculation goes to the calculation of particle-particle interaction.**
- **This is true even for schemes like Barnes-Hut treecode or FMM.**
- **A simple hardware which just calculates the particle-particle interaction can greatly accelerate overall calculation.**



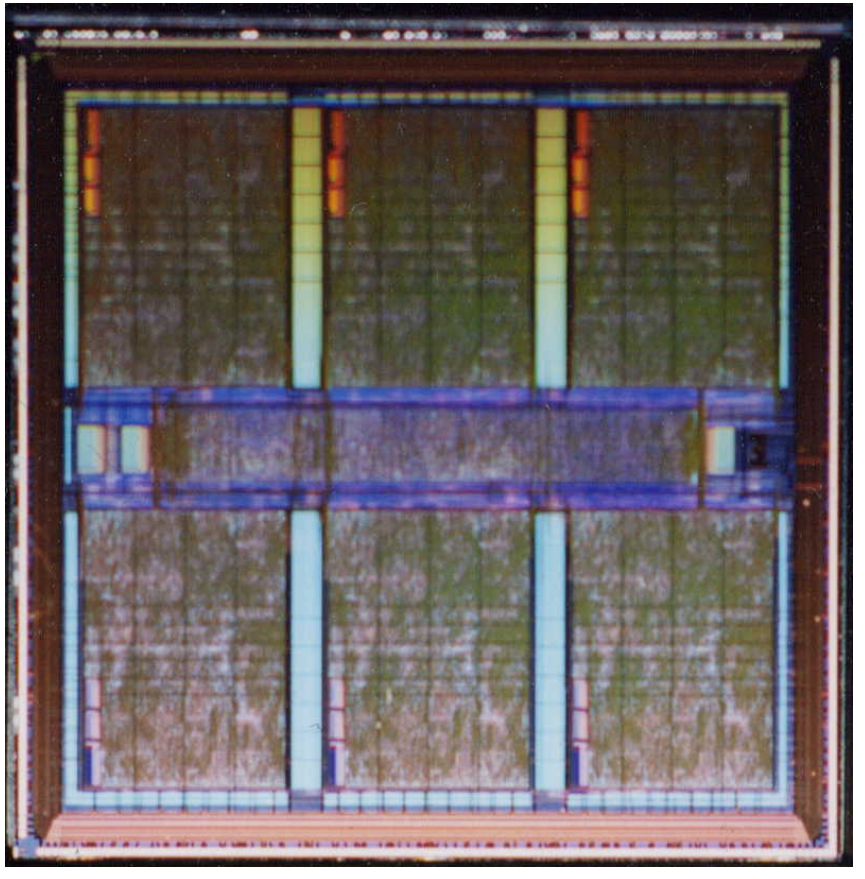
GRAPE-1 to GRAPE-6





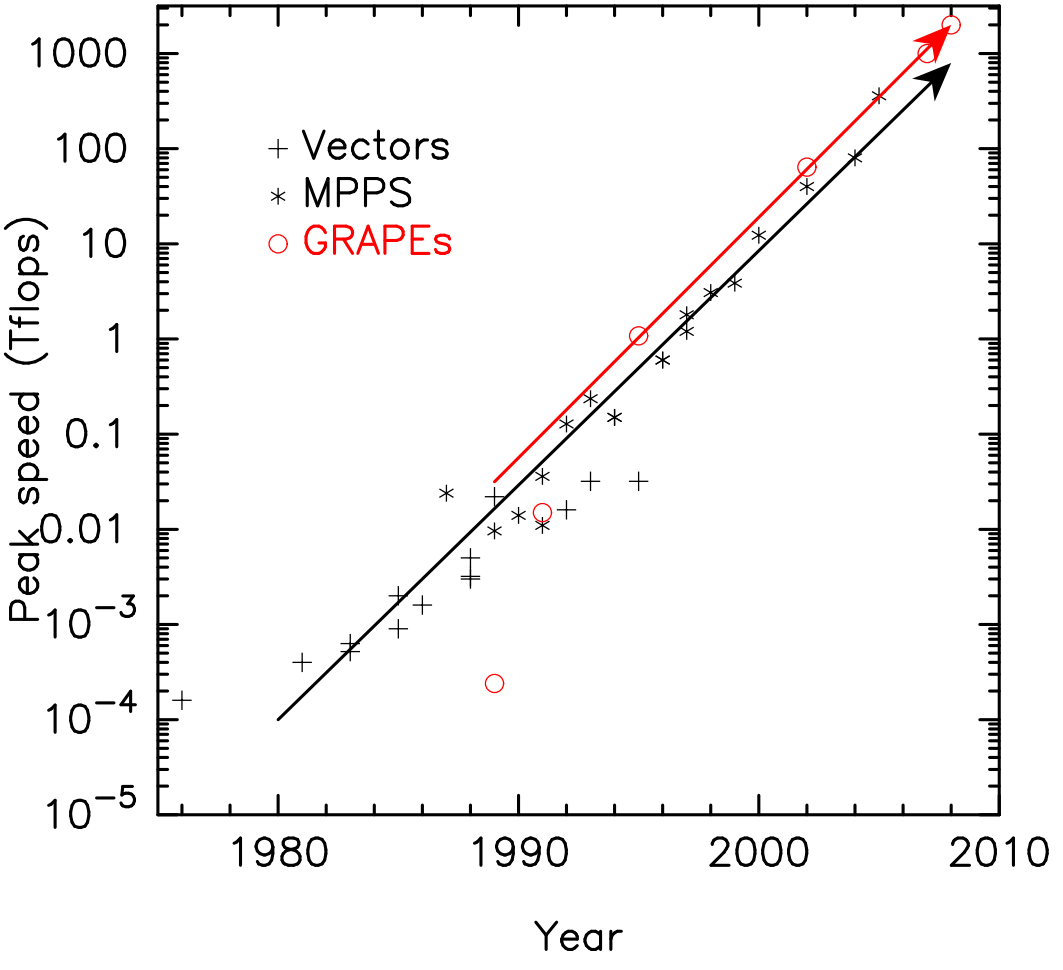
GRAPE-1: 1989, 308Mflops
GRAPE-4: 1995, 1.08Tflops
GRAPE-6: 2002, 64Tflops

Processor LSI



- 0.25 μm design rule (Toshiba TC-240, 1.8M gates)
- 90 MHz Clock
- 6 pipeline processors
- 32.4 Gflops / chip

Performance history



Since 1995 (GRAPE-4), GRAPE has been faster than general-purpose computers.

Development cost was around 1/100.

Comparison with a recent Intel processor

	GRAPE-6	Intel Xeon 5365
Year	1999	2006
Design rule	250nm	65nm
Clock	90MHz	3GHz
Peak speed	32.4Gflops	48Gflops
Power	10W	120 W
Perf/W	3.24Gflops	0.4 Gflops

"Problem" with GRAPE approach

- Chip development cost becomes too high.

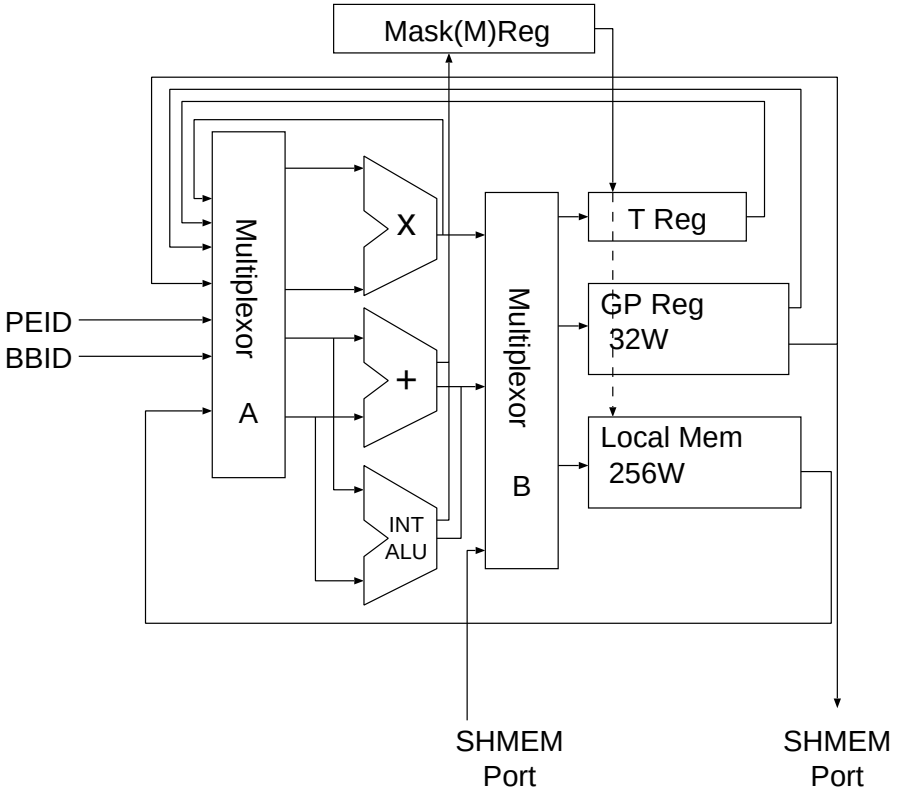
Year	Machine	Chip initial cost	process
1992	GRAPE-4	200K$	1μm
1997	GRAPE-6	1M$	250nm
2004	GRAPE-DR	4M$	90nm
2008?	GDR2?	$\sim$ 10M$	65nm?

Initial cost should be 1/4 or less of the total budget.
How we can continue?

Next-Generation GRAPE — GRAPE-DR

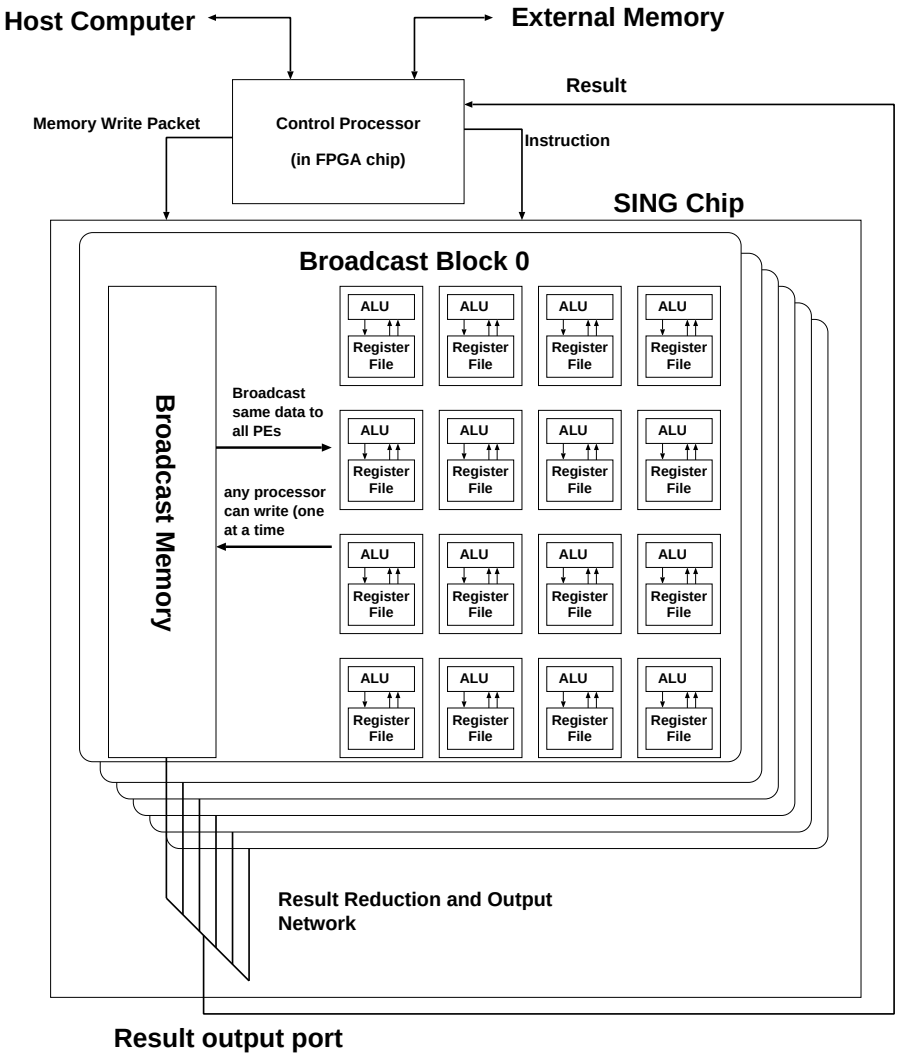
- Planned peak speed: 2 Pflops
- New architecture — wider application range than previous GRAPEs
- primarily to get funded
- No force pipeline. SIMD programmable processor
- Planned completion year: FY 2008 (early 2009)

Processor architecture



- Float Mult
- Float add/sub
- Integer ALU
- 32-word registers
- 256-word memory
- communication port

Chip structure



Collection of small processors.

512 processors on one chip
500MHz clock

Peak speed of one chip: 0.5 Tflops (20 times faster than GRAPE-6).

Why we changed the architecture?

- To get budget (N-body problem is too narrow...)
- To allow a wider range of applications
  - Molecular Dynamics
  - Boundary Element method
  - Dense matrix computation
  - SPH
- To allow a wider range of algorithms
  - FMM
  - Ahmad-Cohen
  - ...

Comparison with FPGA

- much better silicon usage (ALUs in custom circuit, no programmable switching network)
- (possibly) higher clock speed (no programmable switching network on chip)
- easier to program (no VHDL necessary; assembly language and compiler instead)

Comparison with GPGPU

Pros:

- **Significantly better silicon usage (512PEs with 90nm)**
- **Designed for scientific applications reduction, small communication overhead, etc**

Cons:

- **Higher cost per silicon area... (small production quantity)**
- **Longer product cycle... 5 years vs 1 year**

Good implementations of N-body code on GPGPU are coming (Hamada, Nitadori, Portegies Zwart, Harris, ...)

Comparison with GPGPU(2)

	GRAPE-DR	nV G92	AMD FS9170
Design rule	90	65	55
Clock(GHz)	0.5	1.5	0.8
# FPUs	512	112	320
SP peak(GF)	512	336	512
DP peak(GF)	256	—	?
Power(W)	65	70?	150?

How do you use it?

- **GRAPE: The necessary software is now ready. Essentially the same as GRAPE-6.**
- **Matrix etc ... RIKEN/NAOJ will do something**
- **New applications:**
  - **Primitive Compiler available**
  - **For high performance, you need to write the kernel code in assembly language (for now)**

Primitive compiler

(Nakasato 2006)

```
/VARI  xi, yi, zi, e2;
/VARJ  xj, yj, zj, mj;
/VARF  fx, fy, fz;
dx = xi - xj;
dy = yi - yj;
dz = zi - zj;
r2 = dx*dx + dy*dy + dz*dz + e2;
r3i= powm32(r2);
ff = mj*r3i;
fx += ff*dx;
fy += ff*dy;
fz += ff*dz;
```

- **Assembly code**
- **Interface/driver functions**
- **SIMD parallel data distribution**
- **Data reduction**

are generated from this "high-level description".
(Can be ported to GPUs)

Interface functions

```
struct SING_hlt_struct0{
  double xi;
  double yi;
  double zi;
  double e2;
};
int SING_send_i_particle(struct SING_hlt_struct0 *ip,
                         int n);
...

int SING_send_elt_data0(struct SING_elt_struct0 *ip,
                        int index_in_EM);
...
int SING_get_result(struct SING_result_struct *rp);

int SING_grape_run(int n);
```

A few more words on software

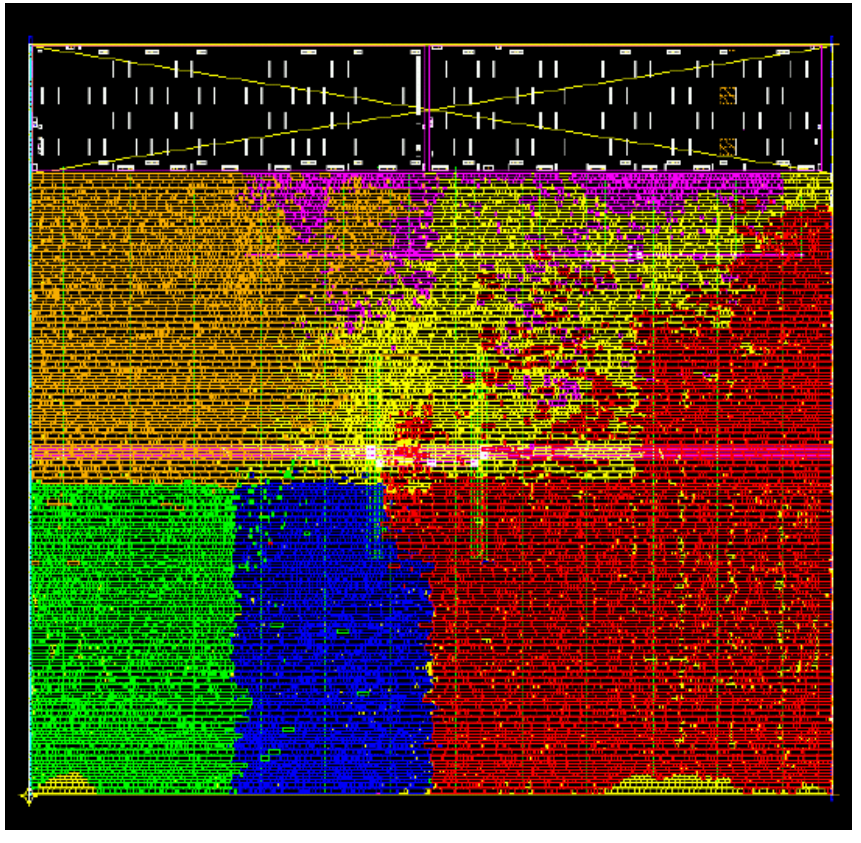
- The right way to separate the task between host CPU and (GRAPE, GRAPE-DR, GPU, FPGA) is the same
- The right way to make efficient use of large number of processors on (GRAPE, GRAPE-DR, GPU, FPGA, CPU) is the same

We should develop a common software platform for different hardwares

Development status

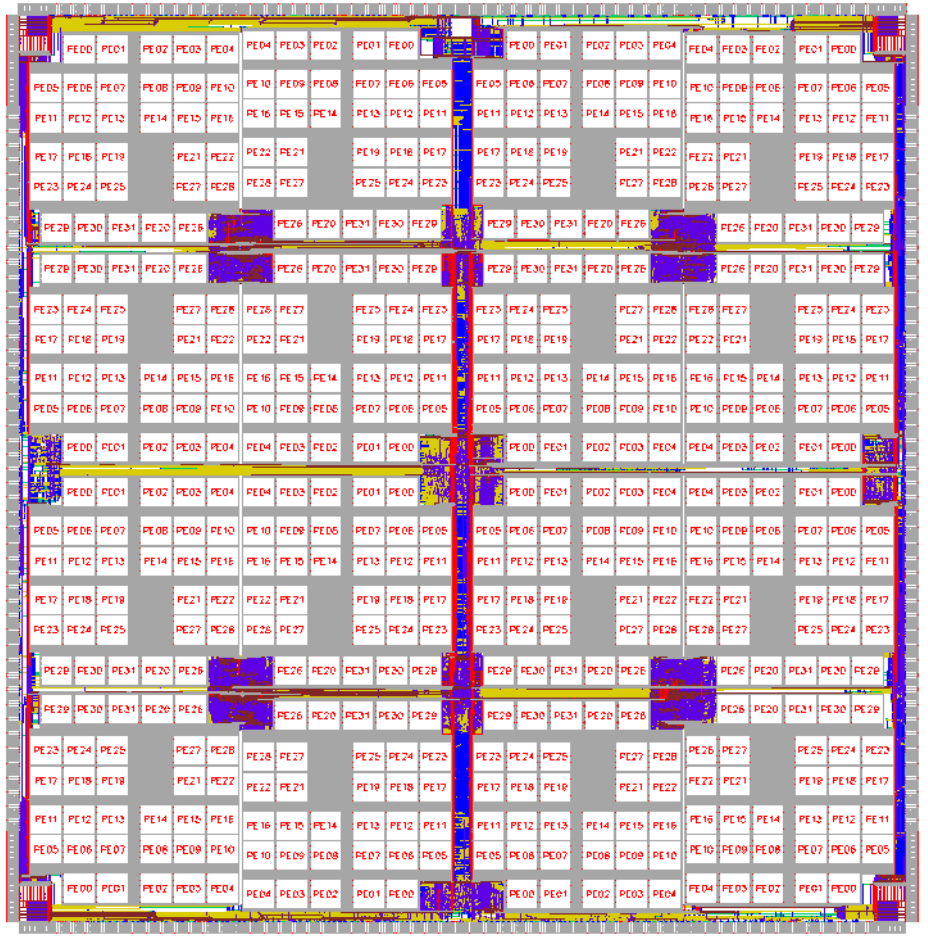
Sample chip delivered May 2006

PE Layout



0.7mm by 0.7mm
Black: Local Memory
Red: Reg. File
Orange: FMUL
Green: FADD
Blue: IALU

Chip layout



- 32PEs in 16 groups
- 18mm by 18mm

Prototype board

2nd prototype. (Designed by Toshi Fukushige)
Single-chip board
PCI-Express x8 interface
On-board DRAM
Designed to run real applications
(Mass-production version will have 4 chips)

Preliminary data for first commercial version

- Prototype board working
- 1 Chip on a board (0.5Tflops peak)
- PCI-Express x4 interface
- 80W ...
- $\sim$ 5K USD ...

GDR-2?

- **We are trying hard to "steal" some money from Japan's "Next-Generation Supercomputer Project"**
- **With 65nm, it is not difficult to achieve**
  - **768 DP Gflops/chip**
  - **1.5 SP Tflops/chip**
  - **On-chip memory (16-32MB)**

Summary

- **GRAPE-DR, with programmable processors, will have wider application range than traditional GRAPEs.**
- **Production version board is now working.**
- **Commercial version should be ready by... sometime early next year**
- **Peak speed of a card with 4 chips will be 2 Tflops.**