GRAPE-DR

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

Talk structure

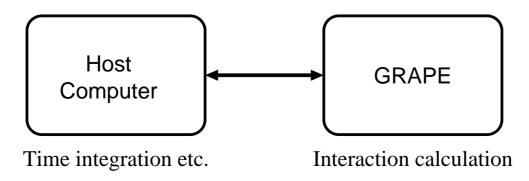
- GRAPE hardwares
 - GRAPE machines
 - GRAPE-DR
- How do they compare with GPGPU?
- GRAPE-DR project status

Short history of GRAPE

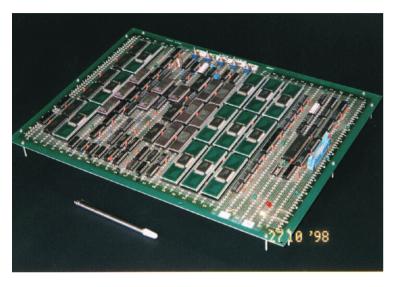
- Basic concept
- GRAPE-1 through 6

Basic concept

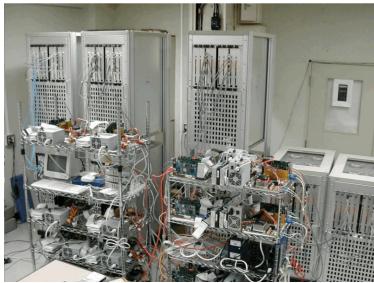
- With *N*-body simulation, almost all calculation goes to the calculation of particle-particle interaction.
- This is true even for schemes like Barnes-Hut treecode or FMM.
- A simple hardware which just calculates the particle-particle interaction can greatly accelerate overall calculation.



GRAPE-1 to **GRAPE-6**

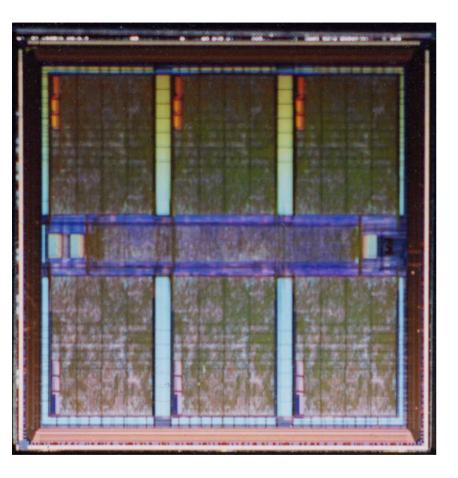






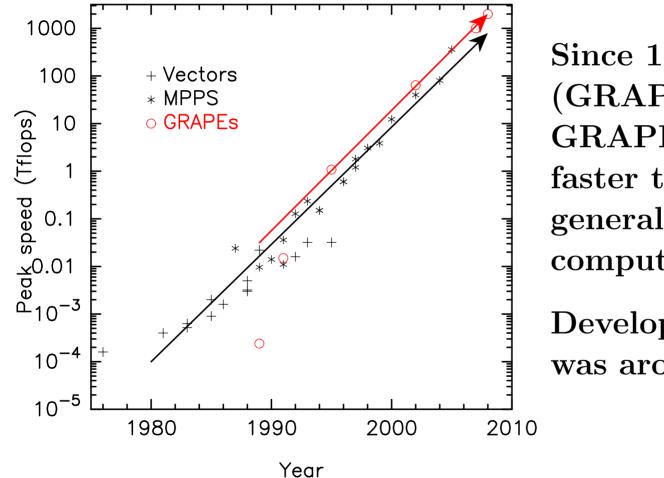
GRAPE-1: 1989, 308Mflops GRAPE-4: 1995, 1.08Tflops GRAPE-6: 2002, 64Tflops

Processor LSI



- 0.25 μ m 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	$250 \mathrm{nm}$	$65 \mathrm{nm}$
Clock	$90 \mathrm{MHz}$	$3 \mathrm{GHz}$
Peak speed	32.4Gflops	48Gflops
Power	10W	$120 \mathrm{W}$
Perf/W	3.24Gflops	0.4 Gflops

"Problem" with GRAPE approach

• Chip development cost becomes too high.

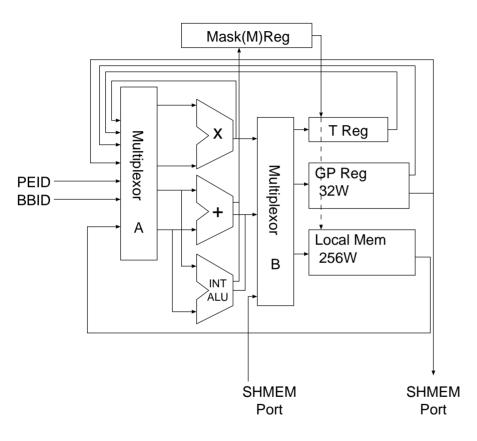
Year	Machine	Chip initial cost	process
1992	GRAPE-4	200K\$	$1 \mu { m m}$
$\boldsymbol{1997}$	GRAPE-6	1M\$	$250\mathrm{nm}$
2004	GRAPE-DR	4M\$	90nm
2008?	GDR2?	$\sim 10 \mathrm{M}\$$	$65 \mathrm{nm}?$

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

Next-Generation GRAPE — GRAPE-DR

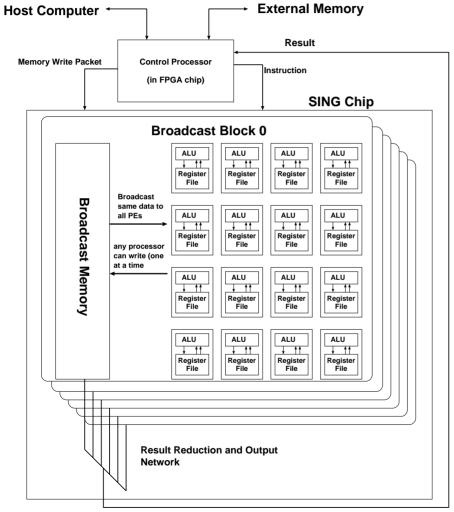
- Planned peak speed: 2 Pflops
- New architecture wider application range than previous GRAPEs
- primarily to get funded
- No force pipeline. SIMD programmable processor
- Planned completion year: FY 2008 (early 2009)

Processor architecture



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

Chip structure



Result output port

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
 - \mathbf{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
${ m Clock}({ m GHz})$	0.5	1.5	0.8
$\# \mathrm{FPUs}$	512	112	320
${ m SP} \ { m peak}({ m GF})$	512	336	512
${ m DP} \ { m peak}({ m GF})$	256		?
$\operatorname{Power}(W)$	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

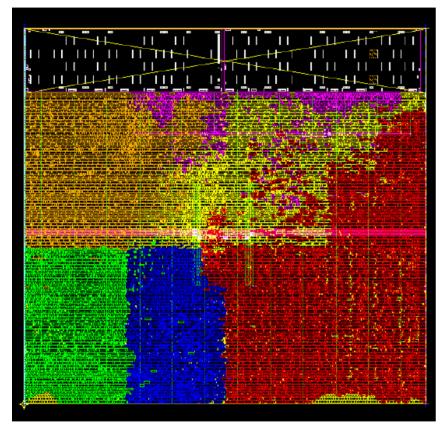
- The right way to separate the task between host CPU and (GRAPE, GRAPE-DR, GPU, FPGA) is the same
- The right way to make efficient use of large number of processors on (GRAPE, GRAPE-DR, GPU, FPGA, CPU) is the same
- We should develop a common software platform for different hardwares

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

				1.1.1	<u>, i i i</u>	<u> </u>	11.1	uų	<u></u> .									ji i	1				. 1. 1.	
	FEDD	PE01	PE 02	PEOP	PE04	PE D4	PED3	PEOZ	PEQI	FEOD				PE CO	PEQI	PEQZ	PEQO	PEGA	FE D4	FED3	FE 02	PEOI	PEOD	U
PE DS	PEDB	PE 07	PE OB	PEOP	PE 10	PE 10	PEDS	PEDØ	FE07	FEOG	FE05	E	FEOD	FE CO	PE07	PEQE	PEQP	PE 10	PE 10	PEDB	PEOB	PE 07	PE06	PEO
PE11	PE1Z	PEID	PE 14	PEID	PE16	PE 16	PE 15	PE 14	PE13	PE12	PE11	F-	PE11	PE12	PE13	PE14	PE15	PE18	PE 10	PE10	PE 14	PEIQ	PEIZ	FEI
PE 17	PC 16	PE 19		PEZI	PC77	PE 22	PE 21		PE 19	PE 18	PE 17		PE 17	PE18	PE 19		PE21	PE22	FE 77	PE21		PE 19	PE 18	PEI
PE 23	PE 24	PE 25		PE27	PE28	FE28	FE 27		PE 25	PE 24	PE 23		PE 73	PE 24	PE25		FC27	FE28	PE 26	PE 27		FE 25	FE24	PE2
PE 2	9 PE 3	D PES	PE 20	PE 26	1		PEZ6	PEZO	PE31 P	E 30 P	E 79		PEZ	9 PE 3	O PES	FEZD	FE 26	20		FE26	FE20	FE31 P	£ 30 P	τ72
PEZ	e PEO	D PES	PE 20	PEZE			PEZG	PE 20	PE31 P	£20 F	£ ZP		PEZ	9 PES	O PE3	PE ZD	PE ZB			PE26	PE20	PE31 P	E 20 F	T 79
FE 73	FE 74	FE 75		PEZ7	PEZB	PE 75	PE 17		FE 75	FE 74	FE ZO		FE ZO	PE 74	PE75		PC27	PEZB	FE 78	FE 27		PE 25	PE74	PEZ
PE 17	PC 18	PE 19		PE21	PE22	PE 22	FE 21		PE 19	PE 18	PE 17		PE 17	PE 18	PE 19		PE21	PE22	PE 22	PE21		PE 19	PE 15	PE
PE 11	PC12	PE 18	PE14	PE15	PE18	FE 16	FE 15	FE 14	PE18	PE 12	PE11		PE11	PE 12	PE13	PE14	PE15	PE 16	PE 18	PC 15	PE 1.4	PE13	PE 12	PE
PEDS	PEDB	PE 07	PE OB	PEOP	FE 10	PE 10	FEDØ	FEDS	PED7	PEOB	PE 05		PE 05	PE 06	PE07	PEOB	FEO9	FE 1D	PE 10	PEDB	PEOB	PE 07	PEOG	PE
	PEDD	FE01	PE 02	PEO3	PE04	PED4	FED3	FED2	PEOI	PEOD				PE 00	PE01	PE02	FEOD	FEC4	PE D4	PED3	PE 02	PE01	PE00	
	PEDD	FE01	PE 02	PEO3	PE 04	FED4	FED3	FED2	PEOT	PEOD		10.00	NY' N	PE 00	FE01	PE07	FEOD	FECH	PED4	PED3	PE 02	FE01	PEOD	
PEDS	PEDE	PE 07	PE OB	PEOP	PE10	PE 10	FEDR	FEDS	PED7	PEOB	PEOS	Ĩ	PEOS	PE 06	PE07	PEOB	FEO9	FE 1D	PE 10	PEDE	PEOB	PE 07	PEOG	PEC
	PE12	PE13	PE14	_	PE16	PE 16	PE 15	\vdash	PE13	-	PE11	P	PE11	PE12	PE13	PE14	PE15	PE15	PE 16		PE14	PE13	PE12	FE
PE 17	PC 18	PE 19		PEZI	PE 22	PE 22	PE 21		PE 19	PE18	PE 17		PE 17	PE18	PE 18		PE21	PE27	FE 72	PE21		PE 18	PE18	PE
PE 23	PE 24	PE25		PE27	PE28	PE 28	PE 27		PE 25	PE 24	PE 23		PE 23	PE 24	PE25		PE27	PE28	PE 28	PE 27		PE 25	PE24	PE2
PE 2	e FED	D ^I PES	PE 20	PE 26			FE26	PE20	FE31 P	520 P	E 28		PE2	e PES	0 ¹ PE3 ¹	PE 20	PE 26	Ter.		PE26	PE20	PE31 P	5 20 F	1 16 2 P
PE 2	9 PE3	O PEA	PE 20	PE 26					PE31 P	-	-		PE2	9 PE3	0 PE3	PE20	PE 28			PE26		PE31 P	-	
PE 23	PE24	PE 25		PE27	PE2B	FE 28	PE 27		PE 25		PE 23	2 -	2	PE24	PT 75	1	FE27	PERE	PE 26	PE 27		PE 25	PE24	PE2
PE 17	PE 18	PE 19		PEZI	PC 2 2	PE22	PE21		PE 19	PE 18	PE 17			PE18	PE18		PE21	PE20	FEZZ	PE21		-	PE18	⊢
PE11	PE12	PE13	PE14	PE15	PE16	PE 16	PE 10	FE 14	PE1a	PE12	PE11	ः जू	PE 11	PE12	PE13	PE 14		PE 16	PE 16	PE 16	PE14	PE 13	PE12	FET
FE 05	FE 06	FE07	PEOS	PE09	PE 10	PE 10	PED9		PE07		PEOS			PEOB		PEOB	PEOP		PE 10	FE 09	FEOB	PE 07	PE06	PEC
	FEOD	PEDI	PEOZ	PEOD	PE04	PECM	PE03	PEO2	PEO1	PEOD		設い		PEOP	PEO1	PE02	PE03	PECH	FE (14	FEOS	FEOT	PEGI	PEOD	
			-			12	1 4.0		- Line i		6.2	4				1.413		-		e line	i i	-		

- 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 ...
- $\bullet \sim 5 {\rm K}~{\rm USD}$...

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.