### **GRAPE-DR:**

2-Pflops massively-parallel computer with 512-core, 512-Gflops processor chips for scientific computing

### Jun Makino

Center for Computational Astrophysics National Astronomical Observatory of Japan and Kei Hiraki and Mary Inaba University of Tokyo

### Talk structure

- Problem with microprocessors
- Possible alternatives
  - FPGA, GPGPU, Specialized hardware
  - One-chip massive SIMD architecture
- GRAPE-DR concept
- GRAPE-DR project status
- Comparison with other approaches
- Summary

# **Problem with microprocessors**

### **Evolution of Microprocessors**



- Transistors: 1000 times in last 15 years
- FPUs: 8 times more in the same period
- a factor of 100 "lost"

## Why FPUs have not increased?

- Memory Wall
  - The bandwidth of off-chip main memory limits the performance. Added transistors need to be used for cache memory and other logic
- Amdahl's Law
  - Parallel speedup is limited by non-parallel part of the program
  - Certainly true for many business applications on Windows PC.

# Is Memory Wall really a problem?

No, for many important applications.

- Any application which requires dense matrix computation
- Many particle-based simulations (astronomy, molecular dynamics, gridless hydro etc)
- $\bullet$  Quantum Chemistry application with local basis or O(N) method
- Yes, for some other applications.
  - Anything which requires large-scale FFT
  - CFD with explicit time integration

# Approaches which ignore MW

- Reconfigurable computing
- GPGPU
- Special-purpose systems
- SIMD chips

# **Reconfigurable computing**

- Commercial FPGAs cannot do FP operations faster than microprocessors.
  - Has been so in last decade
  - Will remain so for foreseeable future
- Custom reconfigurable processors cannot compete with commercial FPGAs in price-performance ratio.
  - Longer development cycle
  - Far smaller quantity

Good for applications with short-wordlength

### GPGPUs —What manufacturers show:

#### GFLOPS



"GPUs beat Moore's Law!"

### GPGPUs —Today



#### Hmm...

### GPGPUs —Same data in log plot



- Faster-than-Moore period ended in 2005
- Microprocessors are catching up
- DP performance?
- Design limit with memory bandwidth

# Special-purpose systems GRAPE and MDGRAPE

- Specialized hardware for the calculation of interaction between particles
- Other operations are done in general-purpose PCs



## **GRAPE-1** to **GRAPE-6**







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

# **Processor LSI**



- 0.25  $\mu$ 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	<b>32.4Gflops</b>	48Gflops
Power	10W	$120 \mathrm{W}$
$\operatorname{Perf}/W$	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\$	$1 \mu { m m}$
1997	<b>GRAPE-6</b>	1M\$	$250 \mathrm{nm}$
2004	<b>GRAPE-DR</b>	4M\$	<b>90</b> nm
2008?	GDR2?	$\sim 10 \mathrm{M}\$$	65nm?

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

# In short...

- FPGAs: not enough FPUs because of reconfigurability
- GPUs: not enough FPUs because of the application requirements
- Special-purpose: We can't pay the initial cost
- We need something with
  - Very large number of FPUs
  - reasonable amount of programmability

# SIMD processor without local memory



- Without external memory
- Without communication network
- Can do almost everything GRAPE or FPGA can do
- Can have much larger number of FPUs than FPGA or GPU

# **GRAPE-DR**

- Planned peak speed: 2 Pflops
- New architecture wider application range than previous GRAPEs
- No force pipeline. SIMD programmable processor
- Planned completion year: FY 2008 (early 2009)
- "Greatly Reduced Array of Processor Elements with Data Reduction"

# **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).

# 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
$\operatorname{Clock}(\operatorname{GHz})$	<b>0.5</b>	1.5	0.8
# FPUs	512	112	<b>320</b>
SP peak(GF)	512	336	512
DP peak(GF)	<b>256</b>		?
$\operatorname{Power}(W)$	<b>65</b>	70?	150?

### How do you use it?

- Particle simulations: The necessary software is now ready. Essentially the same as GRAPE-6.
- Matrix etc ... Libraries will be provided
- New applications:
  - Primitive Compiler available
  - For high performance, you need to write the kernel code in assembly language (for now)

# Primitive compiler (NAOJ/RIKEN)

(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 and FPGAs)

# 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 thoughts 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 Chip and board at Booth # 2133

# **PE** Layout



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

# Chip layout

: E			<b>.</b>	1,1.			L.J.	1.1.1			1						╷╷╷		ji i	1	)       		_î_   	1.1.	
		FEDD	PEO1	PE 07	PE 02	PE04	PE D4	PEDO	PE 02	PEQ1	FEOD				PE CO	PEQI	PEQZ	PEQO	PEGA	FED4	FED3	FEOZ	PE01	PEOD	
	PEDS	PEDB	PE 07	PE 08	PE 09	PE10	FE 10	PEDS	PE 0.5	FED7	FE 06	FEOS	E	FE 00	FE CO	PE07	PEOB	PEQ9	PE 10	PE 10	PEDP	PEOB	PE 07	PE06	PE05
	PE11	PE12	PE10	PE14	PE 1.5	PE16	PE 16	PE 15	PE 14	PE13	PE12	PE11	F-	PE11	PE12	PE13	PE14	PE15	PE18	PE 16	PE 10	PE 14	PEID	PE1Z	FEII
_	PE 17	PC 16	PE 19		PEZI	PEZZ	PE 22	PE 21		PC 19	PE 18	PE17		PE17	PE 18	PE19		PE21	PE22	FEZZ	PE21		PE 19	PE 18	PE 17
	PE 23	PE 24	PE 25		PE27	PE28	PE28	PE 27		PE 25	PE 24	PE 23		PE 23	PE 24	PE25		PE27	PE28	PE 26	PE 27		PE 25	FE24	PE23
-	PE 2	9 PE 3	PE31	PE 20	PE 26	1		PEZ6	PE70	PE91 FI	E30 F	E 79		PEZ	9 PEN	PE31	FE ZO	FE 26	20		FE26	PE20	FE31 P	E 30 P	t 79
	PEZ	e PEO	PERI	PE20	PEZE			PEZE	PE 20	PE31 P	520 P	E ZP	1	PEZ	e PEO	0 PEO	PE ZD	PE ZE			PE26	PE20	PE31 P	E 20 P	E 79
	FE 73	FE 74	FEZD		PE27	PE 28	PETS	PE 17		FE ZD	FE 74	FE ZO		FE ZO	PE 24	PEZ5		PC27	PEZB	FE 78	FE 27		PE ZO	PE74	PEZO
	PE 17	PC 18	PC 19		PE21	PE22	PE 22	FE 21		PE 19	PE 18	PE17		PE 17	PE 18	PE 19		PE21	PE22	PE 22	PE21		PE 19	PE 15	PE17
=	PE11	PE 12	PE13	PE 14	PE15	PE1E	FE 16	FE 15	FE 14	PE13	PE 12	PE11	1.110	PE11	PE12	PE13	PE14	PE15	PE 16	PE 16	PE 15	PE 14	PE13	PE12	PE11
	PEDS	PEDE	PE 07	PE OB	PEOP	PE 10	PE 10	PE DØ	FEDS	PED7	PE 06	PE 05		PE 05	PE 06	PE07	PEOB	PE09	FE 1D	PE 10	PEDB	PEOB	PE 07	PEOE	PE05
		PEDD	PEO1	PE 02	PEO3	PE04	PED4	FED3	PE D2	PE01	PE OD				PE 00	PE01	PE02	PEOD	FEC4	PE D4	PED3	PE 02	PE01	PEOD	
		PEDD	PE01	PE 02	PE 03	PE04	FED4	FED3	PED2	PE01	PEOD			AYY SA	PE 00	PE01	PE02	FEOD	PECH	PED4	PED3	PEO2	PE01	PEOD	
	PEDS	PEDE	PE 07	PE OB	PE 02	PE10	PE 10	FED®	FEDS	PED7	PEOE	PEOS	Ĩ	PE 05	PE 06	PE07	PEOB	FEO9	FE1D	PE 10	PEDE	PEOB	PE 07	PEOF	PE05
	PE11	PE 12	PE18	PE14	PE 15	PE16	FE 16	PE 15	PE 14	PE18	PE 12	PE11	F	PE11	PE12	PE13	PE14	PE15	PE16	PE 16	PE 16	PE 14	PE13	PE12	FE11
	PE 17	PC 18	PE 19		PE21	PEZZ	PE22	PE 21		PC 19	PE 18	PE17		PE 17	PE 18	PEID		PE21	PE27	FE 72	PE21		PE 19	PE18	PE17
	PE 23	PE 24	PE 25		PE27	PE28	PE 28	PE 27		PE25	PE 24	PE 23		PE 23	PE24	PE25		PE27	PE28	PE 28	PE 27		PE 25	PE24	PE23
	PE 2	9 FE3	PE31	PE 20	PE 26	g		PE26	PE 20	PE31 P	E30 P	E 28		PE2	e PES	0 PE31	PE 20	PE 26	Ter.		PE26	PE20	PE31 P	E30 P	£ 28
	PE 2	9 PE3	PE31	PE 20	PE 26			PT 26	PF 20	erat e	530 P	F 70		ero	o PEA	PEM	PE 20	PF 26			PE26	PE20	PE31 P	E 30 P	629
	PE 23	PE24	PE25		PE27	PE28							28 -				1		. Base	PE 26	PE27		PE 25	PE24	PE23
	PE 17	PE 18	PE 19		PE21	PEZZ	PE28	PE 21		PE 19	PE 18	PE 17		PE 17	PE18	PE18		PE21	PE27	FE 72	PE21		PE18	PE18	PE17
	PE11	PE 12	PE 13	PE14	PE15	PE1E	ET 16	ET 15	107.14	PEIN	PE 12	85.11	3	85.11	PE12	PE 13	PEIA	PE 15.	PF 15	PE 16	PE 16	PE14	PE13	PE12	FE11
	FE 05	FE 06	FE07	PE06	PE09	PE 10	PE 10	PED9	PEOS	PE07	PEOB	PEOS		PE 05	PEOB	PE07	PEOB	PEOP	PE10	PE 10	FE 09	FEOB	PE07	PE06	PE05
F		FEOD	PEQ1	PE 07	PEOD	PE04	PEDA	PE03	PE02	PEO1	FE OO	-			PEOP	PE01	PE02	PEO3	PECH	FE (14	FEOS	FEOT	PEGI	PEOD	
							-	1		ller (S	12.11						. AT 31				Taili	i Since bi	الفقر حاد	-	

- 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)

## **Prototype board performance**

- Measured on first board with PCI-X interface (communication limited)
- Assembly code not fully optimized yet
- 50 G flops for  ${\cal N}=1024$
- asymptotic speed 170Gflops

# 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**?

- With 65nm, it is not difficult to achieve
  - 768 DP Gflops/chip
  - -1.5 SP Tflops/chip
  - On-chip memory (16-32MB)
- $\bullet$  Could reach 10P flops with 13,000 chips, 2-3MW
- With 45nm the performance more than doubles

# Summary

- GRAPE-DR is a massively parallel SIMD processor chip with 512 PEs in a chip.
- Can be applied to a fairly wide range of applications.
- The processor chip is completed and is working as designed.
- Peak speed of a card with 4 chips will be 2 Tflops.
- We plan to complete a 1 Pflops (DP) system by the end of FY 2008.
- Chip and board at Booth # 2133 — Please visit us!