The CP-PACS Parallel Computer

Akira Ukawa[a]

for the CP-PACS Collaboration

[a] *Institute of Physics, University of Tsukuba, Tsukuba, Ibaraki 305, Japan*

The CP-PACS computer is a massively parallel computer consisting of 2048 processing units and having a peak speed of 614GFLOPS and 128GByte of main memory. It was developed over the four years from 1992 to 1996 at the Center for Computational Physics, University of Tsukuba, for large-scale numerical simulations in computational physics, especially those of lattice QCD. The CP-PACS computer has been in full operation for physics computations since October of 1996. In this article we describe the chronology of the development, the hardware and software characteristics of the computer, and its performance for lattice QCD simulations.

Key words: parallel computer; lattice QCD; Monte Carlo simulation.

1 Introduction

Development of parallel computers for lattice field theory simulations by now has over a decade of history. In Japan the first effort in this direction was made by the QCDPAX Project (1987–1989)[1], which built a 480 processor parallel system with a peak speed of 14GFLOPS. The QCDPAX was actually the fifth of the PAX series of computers which pioneered the development of parallel computers for scientific applications in Japan[2].

The CP-PACS Project was conceived as a successor of the QCDPAX Project in the early summer of 1991. The project name is an acronym for *Computational Physics by Parallel Array Computer System.* The goal was to develop a parallel computer capable of a few hundred GFLOPS of speed not only for lattice field theory simulations but also for a wider area of applications in computational physics, and to carry out research with the computer upon completion of its development.

The Project was proposed to the Ministry of Education, Science and Culture in

Table 1

Members of the CP-PACS Project.

computer science		computational physics		
hardware	software	particle physics	astrophysics	solid state physics
K. Nakazawa[a]	I. Nakata[e]	Y. Iwasaki[c,k]	M. Miyama[n]	S.Miyashita[p]
H. Nakamura[b]	Y. Yamashita[e]	A. Ukawa[k]	T. Nakamura[o]	M. Imada[q]
T. Boku[c,e]	Y. Oyanagi[f]	K. Kanaya[c,k]	M. Umemura[c,k]	K. Nemoto[r]
T. Hoshino[d]	T. Kawai[g]	S. Aoki[k]	Y. Nakamoto[c,k]	A. Oshiyama[k]
T. Shirakawa[d]	M. Mori[h]	T. Yoshie[c,k]		S. Gunji[c,k]
K. Wada[e]	Y. Watase[i]	M. Fukugita[l]		
M. Yasunaga[e]	S. Ichii[j]	M. Okawa[m]		
S. Sakai[c,e]		N. Ishizuka[c,k]		
		H. Kawai[m]		

[a] Dept. of Comp. Science, Univ. of Electro-Communications
[b] Center for Adv. Sci. and Tech., Univ. of Tokyo
[c] Center for Comp. Physics, Univ. of Tsukuba
[d] Inst. of Engineering Mechanics, Univ. of Tsukuba
[e] Inst. of Information Sciences, Univ. of Tsukuba
[f] Dept. of Information Sciences, Univ. of Tokyo
[g] Dept. of Science and Technology, Keio University
[h] Dept. of Engineering, Univ. of Tokyo
[i] Computer Center, KEK
[j] Computer Center, Univ. of Tokyo
[k] Inst. of Physics, Univ. of Tsukuba
[l] Inst. of Cosmic Ray Res., Univ. of Tokyo
[m] Theory Division, KEK
[n] National Astronomical Observatory
[o] Yukawa Institute, Kyoto University
[p] Dept. of Physics, Osaka University
[q] Inst. of Solid State Physics, Univ. of Tokyo
[r] Dept. of Physics, Hokkaido Univ.

the winter of 1991-92. It was approved in the early spring of 1992 as one of the "Program for New Development of Academic Research" of the Ministry, to be executed over five years from April 1992 till March 1997. The funding, totaling 2.2 billion Yen over this period, came from a special category of Grants-in-Aid of the Ministry allocated for innovative fundamental research. Simultaneously with the start of the Project, the Center for Computational Physics[3] was founded at University of Tsukuba as a inter-university research facility for carrying out the Project and for promoting research in computational physics.

The current members of the Project is listed in Table 1. A cross-disciplinary collaboration between computer scientists and computational physicists is an important feature of the Project, which has proven crucial for developing a computer design capable of high performance for actual physics applications. The Project is headed by Y. Iwasaki, and the development of the CP-PACS computer has been led by K. Nakazawa.

The development of the CP-PACS computer took four years from 1992 to 1996. For the industrial partner, Hitachi Ltd. was selected by a formal bidding process in the summer of 1992. The company has since worked in a close collaboration with the Project members both on the hardware and the software development. The initial system of 1024 processors was installed at the Center for Computational Physics in March 1996. After a few months of testing and physics runs, an upgrade to a 2048 processor system was started in August 1996, which was completed in late September 1996. The CP-PACS computer

Table 2

Specifications of the CP-PACS computer.

Architecture	MIMD with distributed memory	
Number of nodes	2048 (PU) + 128 (IOU)	
Theoretical peak speed	614.4GFLOPS(64bit data)	
Total main memory	128GByte	
Node processor	PA-RISC-based custom CPU with *PVP-SW feature*	
	Clock cycle	150MHz
	Peak speed	300MFLOPS
	Floating point registers	128(64bit)
	First-level cache	16(I)+16(D) KByte
	Second-level cache	512(I)+512(D) KByte
Network	*3-dimensional Hyper Crossbar* ($8 \times 17 \times 16$)	
	Data transfer throughput	300MByte/sec (per link)
	Latency	3 microsec
	Method	wormhole + *remote DMA*
Software	OS	UNIX micro kernel
	Language	FORTRAN90, C, assembler
Disk system	529GByte (*distributed RAID-5 disk*)	
External connection	HIPPI (peak throughput 100MByte/sec)	
	ETHERNET	
Size	7m(width) $\times$ 4.2m (depth) $\times$ 2.0m (height)	
Power dissipation	275KW maximum	
Cooling system	air-cooling	

has been in full operation for physics computations since October of 1996.

2 The CP-PACS Computer

2.1 Hardware architecture

The specification of the CP-PACS computer is listed in Table 2. The system consists of 2048 processing units (PU's) and 128 I/O units (IOU's) connected together in an $8 \times 17 \times 16$ array by the *Hyper Crossbar* network. The IOU's form an $8 \times 1 \times 16$ subarray at the end of the y direction.

Each PU carries a superscalar RISC processor with a peak speed of 300MFLOPS, a storage controller and 64MBytes of DRAM memory, and a network interface adapter. The processor is custom made with the architecture based on

PA-RISC 1.1. A well-known problem of RISC processors for large-scale scientific computations is the performance degradation occurring when the data size exceeds the cache memory capacity because of long memory access latency. In order to solve this problem, an upward compatible enhancement of architecture called *PVP-SW (Pseudo Vector Processor based on Slide Window Registers)* has been developed for the CP-PACS processor[4].

The PVP-SW enhancement has several elements. The *slide window mechanism* allows use of a large number of physical registers (128 for the CP-PACS processor) through a continuous sliding of the logical register window of 32 registers along the physical registers. The *preload* and *poststore* instructions for floating point data, which can be issued without waiting for the completion of previous load/store operations, allow a pipelined access to main memory, thus resolving the long memory latency problem. Finally the software pipelining technique is employed for vector-like execution of instructions. With these features an efficient vector processing is realized in spite of the superscalar architecture of the CP-PACS processor.

The Hyper Crossbar network of the CP-PACS computer is made of crossbar switches in x, y, and z directions connected together by an exchanger at each of the three-dimensional crossing point of the crossbar array. Each exchanger has a PU or IOU attached to it so that data transfer of any pattern is possible by stepping through a maximum of three crossbar switches. Furthermore a hardware bisection of each crossbar switch is possible so that the system can be divided up to 8 independent subsytems.

Data transfer through the network is made through *Remote DMA (Remote Direct Memory Access)*, in which data are sent or received directly between the memories of the processors with a minimum of intervention from the operating system. This leads to a significant reduction in the startup latency and high throughput of data transfer.

The distributed disk system of CP-PACS is connected to the 128 IOU's at the end of the y crossbar switches by SCSI-II bus. RAID-5 disks are used for fault tolerance. The IOU's handle file I/O requests issued by the PU's distributively using the Remote DMA through the Hyper Crossbar network.

The CP-PACS computer is controlled by a front-end host through an ETHERNET connection. For external data I/O, a HIPPI connection is set up between the front-end host and one of the IOU's.

2.2 Hardware implementation

The CPU chip is fabricated using 0.3 micron CMOS semiconductor technology. The chip size is 15.7mm x 15.7mm. The CPU, the storage controller (SC) and the network interface adapter (NIA) are mounted on a ceramic multi-chip module of a size 5.7cm x 5.7cm. The SC and NIA chips are fabricated using 0.5 micron CMOS gate-array technology. The PU module also carries the off-chip second-level cache memory chips.

Eight PU modules together with 4Mbit DRAM memory are mounted on a board of size 45.6cm x 62.5cm. Each board has two more chips for the crossbar switches in the x direction, and one chip for the clock distributer.

Sixteen PU boards and one IOU board are placed vertically on a back plane, and two back planes, one on top of the other, are housed in a cabinet. A crossbar switch in the y direction is mounted on the backplane in the cabinet. Crossbar switches in the z direction are mounted on two separate boards, which are housed in two cabinets.

The entire system consists of 8 cabinets for PU/IOU boards, 2 cabinets for z-crossbar switches, and additional 4 cabinets for I/O adapters. The 14 cabinets are arranged in an H shape ; the z-crossbar cabinets occupies the central part, and each leg has 3 cabinets with an I/O adapter cabinet placed at the edge. The system is cooled by air drawn in from beneath the cabinets. The RAID-5 distributed disk system, connected to the I/O adapters, is placed a few meters from the CP-PACS computer.

2.3 Software

The CP-PACS computer is run under the UNIX operating system. Each node processor, however, carries only a kernel based on Mach3.0 which handles memory control, inter-node communications, process scheduling, interruptions and I/O. The full UNIX operating system is implemented on one of the IOU, named SIOU, which controls parallel processes through the network.

The programming language of the CP-PACS computer is FORTRAN90, C, and assembly language. An assembler code can be included as a subroutine in a FORTRAN or C code in order to maximize the performance. The FORTRAN and C compilers employ the technique of modulo scheduling and register coloring to generate codes incorporating the PVP-SW enhancement . In FORTRAN and C programs, Remote DMA data transfer is made by calling special library routines for communications.

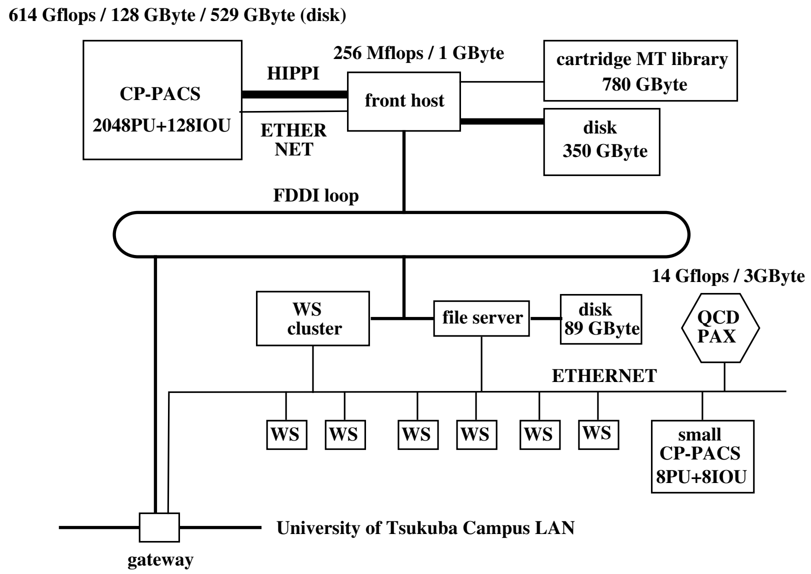

Fig. 1. Computer system of the Center for Computational Physics.

3 Computer system of the Center for Computational Physics

In Fig. 1 a schematic drawing is given of the computer system of the Center including the CP-PACS computer. Program development and test runs are made on the small CP-PACS system consisting of 8 PU's and 8 IOU's. The CP-PACS computer is operated in a batch mode. Job requests are submitted from the front-end host through NQS. Data files are automatically sent and received between the disk system of the front-end host and that of the CP-PACS computer at the start and termination of job requests. Long-term storage of data files uses the automatic tape archive system; the capacity is expected to be increased to 10–20TByte in the near future.

4 Application to lattice QCD

The development of lattice QCD programs started in the fall of 1995. The effort intensified after March 1996 when the 1024 processor system was installed. The performance of programs has steadily improved as the compiler has become better and our experience for optimization has accumulated. A significant step has been the completion of an assembler code for the core part of the minimal residual solver for the Wilson quark matrix; inverting the matrix represents the most CPU time intensive part in lattice QCD simulations. The current performance of basic lattice QCD programs are listed in Table 3.

For the first application of the CP-PACS computer to lattice QCD, we have chosen to carry out a high statistics calculation of the quenched light hadron

Table 3

Performance of lattice QCD programs.

program	MFLOPS/PU calc.	comm.	total	coding	size	# PU
R/B MR solver	191(77%)	(23%)	148	assembler+F90	$48^3 \times 84$	1024
(Wilson action)	99(84%)	(16%)	84	F90	$48^3 \times 84$	1024
CG solver(KS)	139(90%)	(10%)	125	F90	$32^3 \times 128$	256
Heat bath	109(94%)	(6%)	102	F90	$48^3 \times 84$	1024
HMC (full QCD)	151(74%)	(26%)	112	assembler+F90	$16^3 \times 32$	256

Table 4

Quenched spectrum run on the CP-PACS.

$\beta = 6/g^2$	5.9	6.1	6.25	6.47
lattice size	$32^3 \times 56$	$40^3 \times 70$	$48^3 \times 84$	$64^3 \times 112$
# PU used	256	512	1024	2048
CPU time/conf.(hours)	3.0	4.8	6.8	19.0

spectrum with the Wilson quark action. To improve on previous results, we employ a larger spatial size of $La \approx 3$fm, smaller lattice spacings down to $a \approx 0.05$fm, and smaller quark masses reaching $m_\pi/m_\rho \approx 0.4$.

In Table 4 we list the cycle time of the ongoing simulation for calculating hadron propagators. The three smaller lattice sizes are simulated simultaneously by dividing the full array of 2048 processors into four subsystems given by 256+256+512+1024. The quenched run, started in July 1996, is well underway, and we hope to be able to report the results soon.

Acknowledgements

I would like to thank the members of the CP-PACS Project, especially Y. Iwasaki and K. Nakazawa, for valuable discussions. This work is supported in part by the Grants-in-Aid of the Ministry of Education, Science and Culture (Nos. 04NP0801, 08640349).)

References

[1] Y. Iwasaki, T. Hoshino, T. Shirakawa, Y. Oyanagi and T. Kawai, Comp. Phys. Comm. **49** (1988) 449.

[2] T. Hoshino, *PAX Computer, High Speed Parallel Processing and Scientific Computing* (Addison-Wesley, New York, 1989).

[3] For further information on the Center, see, http://www.rccp.tsukuba.ac.jp

[4] H. Nakamura, H. Imori, K. Nakazawa, T. Boku, I. Nakata, Y. Yamashita, H. Wada and Y. Inagami, *Proc. of International Conference on Supercomputing '93*, 298 (1993); H. Nakamura, K. Nakazawa, H. Li, H. Imori, T. Boku, I. Nakata and Y.Yamashita, *Proc. of 27th Hawai International Conference on System Sciences*, 368 (1994).