PROCEEDINGS OF SCIENCE

Status and physics plan of the PACS-CS Project

PACS-CS Collaboration: A. Ukawa[*a,e†], S. Aoki[a,b], K.-I. Ishikawa[d], T. Ishikawa[e], N. Ishizuka[a,e], K. Kanaya[a], Y. Kuramashi[a,e], K. Sasaki[e], N. Tsutsui[c], M. Okawa[d], Y. Taniguchi[a,e], T. Yoshié[a,e],

[a] *Graduate School of Pure and Applied Sciences, University of Tsukuba, Tsukuba, Ibaraki 305-8571, Japan*
[b] *Riken BNL Research Center, Brookhaven National Laboratory, Upton, New York 11973, USA*
[c] *High Energy Accelerator Research Organization (KEK), Tsukuba 305-0801, Japan*
[d] *Department of Physics, Hiroshima University, Higashi-Hiroshima, Hiroshima 739-8526, Japan*
[e] *Center for Computational Sciences, University of Tsukuba, Tsukuba, Ibaraki 305-8577, Japan*

PACS-CS is a large-scale cluster system with a peak speed of 14.3Tflops for lattice QCD developed at the Center for Computational Sciences, University of Tsukuba, as a successor to the CP-PACS computer. The system consist of 2560 nodes connected by a $16 \times 16 \times 10$ three-dimensional hyper crossbar network. Each node has a single low-voltage 2.8GHz Xeon processor with 2GBytes of memory and an aggregate network bandwidth of 750MByte/sec. With PACS-CS, we plan to pursue the $N_f = 2+1$ dynamical simulation with the Wilson-clover quark action, lowering the up-down quark masses as much as possible using the domain-decomposition idea of Lüscher. The status of the machine as well as the physics plan is described.

XXIVth International Symposium on Lattice Field Theory
July 23-28, 2006
Tucson, Arizona, USA

[*] Speaker.
[†] E-mail: ukawa@ccs.tsukuba.ac.jp

PoS(LAT2006)039

1. Introduction

The PACS-CS Project is a three-year plan, spanning the Japanese fiscal years 2005 to 2007, to develop a massively parallel cluster system suitable for computational science applications and carry out frontier research in several areas of computational science with it, lattice QCD being one of them. The name of the system is an acronym for Parallel Array Computer System for Computational Sciences. It is the seventh of parallel computer systems developed and used for scientific calculations at University of Tsukuba.

The planning for the project was started in the summer of 2003. The proposal of the project to the Ministry of Education, our funding agency, and evaluations by Government Committees extended from early summer to fall of 2004. The approval came in the winter of 2004, and the project formally started in April 2005.

The vendor for developing the system was selected by a formal bidding process. We have exchanged contract with Hitachi Ltd. for the system hardware and software development in July 2005. Another contract was made with Fujitsu Ltd. in August 2005 for developing the network driver software. Since then we have being closely working with these vendors to build the PACS-CS system.

In June 2006 the PACS-CS system was installed in the machine room of Center for Computational Sciences of University of Tsukuba. Science computations started in July 2006.

In this report, we describe the status of the PACS-CS system development, and the physics plan we wish to pursue with it over the next several years.

2. Status of PACS-CS

2.1 Machine specification

PACS-CS is a massively parallel system built from commodity components. The specification of the system is given in Table 1. The unique features of the system are: (i) contrary to commercial PC clusters, the PACS-CS node consists of a single processor with a moderate frequency in order to achieve a balance of floating point capability and memory throughput while keeping the power consumption low, and (ii) each node has six Ethernet ports such that the full 2560 nodes of the system are connected into a $16 \times 16 \times 10$ array through a three-dimensional hyper-crossbar network, each link consisting of a dual trunk of Gigabit Ethernet lines. The design considerations which led to these specifications were described in our report for Lattice 2005 last year[1]. In Fig.1 we present the block diagram of the PACS-CS node, which shows how the design ensures the bandwidth between processor and memory, and between memory and the Gigabit Ethernet ports.

2.2 Implementation

Each node of PACS-CS requires 8 Gigabit Ethernet ports, six for the hyper-crossbar network, and two more for external I/O and system diagnostics and maintenance. There are no commercial boards available which has eight Ethernet ports. We also wish to place two nodes per standard 1U shassis so that the processor density remains the same as common commercial clusters. This means we have to develop a new mother board for the node. Figure 2 shows the top view of the board which is about 20cm by 40cm in size. The eight Ethernet ports are at the left edge of the board.

Number of nodes	2560 ($16 \times 16 \times 10$)
Peak performance	14.3Tflops
Node	single CPU + memory + HDD + 8 Gigabit Ethernet ports
CPU	Intel LV Xeon EMT64T@2.8GHz(5.6Gflops) with 1MByte L2 cache
Memory	2GB/node(DDR400 2-way interleave), 5.12TByte/system
Network	3-dimensional hyper-crossbar network (uses dual Gigabit Ethernet/link)
Network throughput	250MByte/sec/direction
	750MByte/sec/node (3-dim simultaneous send/receive)
Local HDD	160GByte$\times$ 2 (RAID-1), 410TByte$\times$2/system
Number of racks	59
System footprint	$100m^2$
Power consumption	545kW

Table 1: PACS-CS specifications

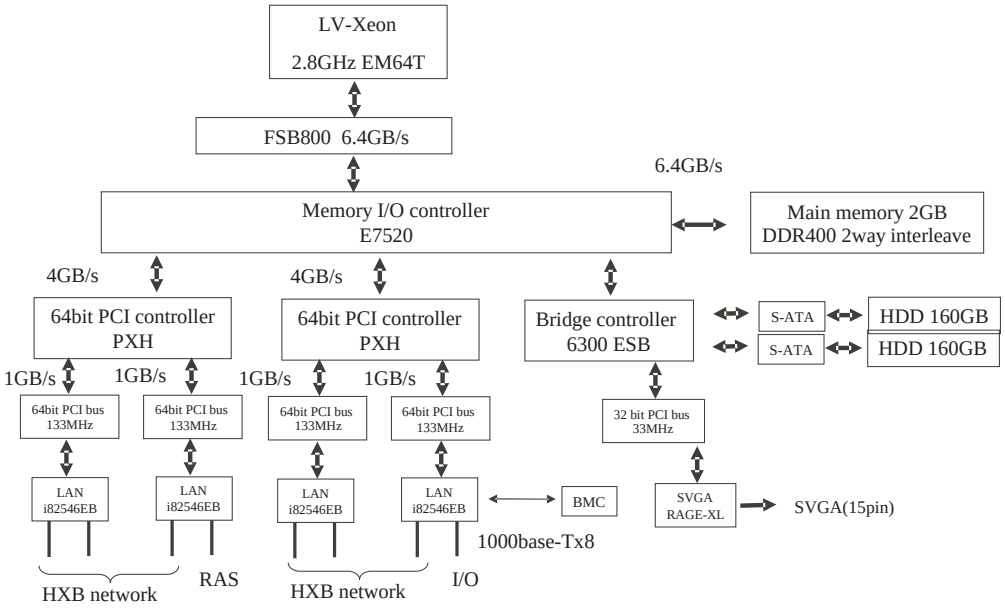

Figure 1: Block diagram of PACS-CS node

Since there are 2560 nodes, there are $2560 \times 8 = 20480$ Gigabit Ethernet cables. The total length of the cables is over 400km. The cables are colored: red for X crossbar, blue for Y crossbar, green for Z crossbar, white and yellow for I/O and system diagnostics. See Fig.4

Figure 3 is the top view of the shassis. Two mother boards are placed on the two sides of the power supply in the middle on the right side of the shassis. The front side of the shassis on the left is filled with 4 units of 3.5" disk, 2units for each node to supply 160GByte of disk space in the RAID-1 mode. We can also see cooling fans placed after the disk units.

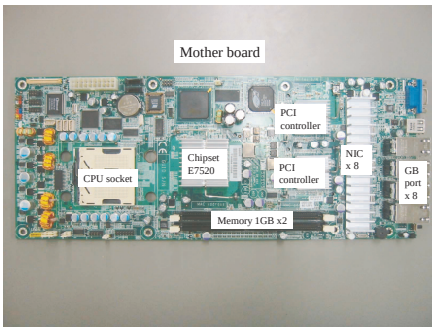

Figure 2: Mother board

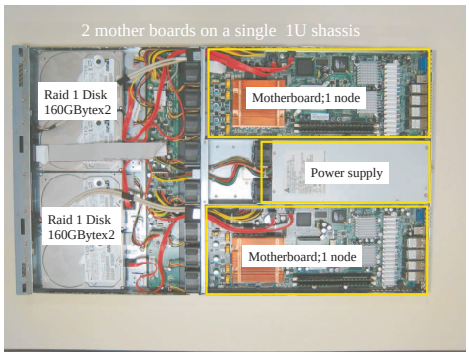

Figure 3: 1U shassis

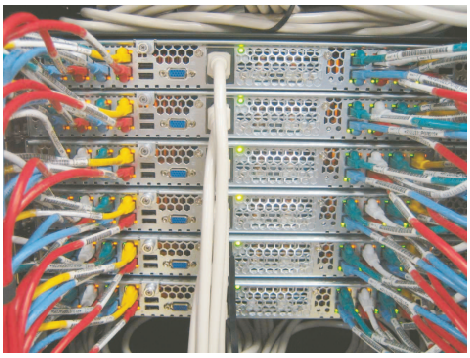

Figure 4: Rear side of 1U shassis. 16 Gigabit Ethernet cables, 8 each per node, are visible.

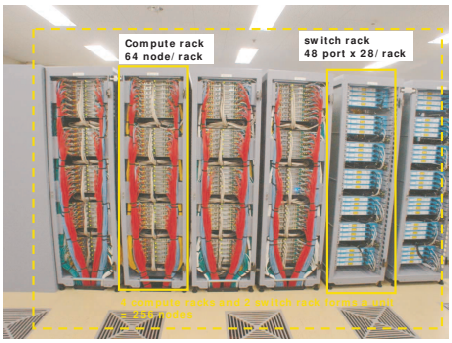

Figure 5: Unit of 256 nodes in 4 racks and switches in 2 racks. The unit at left is power supply.

The system rack is in two varieties, one being the compute rack which houses 32 shassis or 64 nodes per rack, and the other is the switch rack which houses a number of 48 port Gigabit Ethernet switches to form the 3-dimensional hyper-crossbar network. Since the system consists of 2560 nodes, 40 compute racks are needed. The number of switch racks are 18 out of which 15 are used for the hyper-crossbar network, and the remaining 3 are needed for switches for forming the network for external I/O and system diagnostics and control. Finally there is one more rack for a file server with 10TByte of RAID-5 disk directly attached to the PACS-CS system. Thus the PACS-CS system consists of 59 racks in all. As shown in Figs. 5 and 6, these racks are grouped into 10 units, each unit consisting of 4 compute racks and 2 switch racks. The footprint of the entire system is about $100m^2$.

2.3 Software

The operating system of PACS-CS is LINUX, and SCore [2] is used as the cluster middleware. We need to develop a driver for data communication between nodes over the hyper-crossbar network. This software has been developed with Fujitsu Ltd.[3] based on the PMv2 driver available

Figure 6: Total view of the PACS-CS system

from SCore.

The programming language is Fortran90, C and C++. Communication is handled by MPI which will call the hyper-crossbar driver explained above.

2.4 Job execution

For job execution, the PACS-CS system is divided into multiple and overlapping partitions. Starting with the whole system of 2560 nodes, the sizes for partitions are 2048, 1024×2, 512×5, 256×10 etc. Some of the partition sizes come in several topologies, *e.g.*, a 512 partition is either $8 \times 8 \times 8$ or $16 \times 16 \times 2$.

Jobs are executed with a batch queuing system using PBS. A simple set of scripts are utilized to control the job flow including file I/O. In particular, input/output files are stored in the file server of the PACS-CS system and either sent in or retrieved at the start or end of jobs.

2.5 Preliminary performance tests

As a basic test of the system, the Linpack benchmark was executed with the whole system. The result was 10.35Tflops (72.20% of peak) for the matrix size $N = 722944$. This turned out to be the 34th on the Top500 list published in June 2006.

After the PACS-CS system became available for scientific computations in July, we have made several preliminary tests on the network performance. Table 2(a) lists the results for sending 8MB of data from a node to neighboring nodes along the crossbar directions simultaneously in all three directions. Compared to the peak throughput of 750MB/s, roughly 80% is achieved.

(a) 3-dimensional simultaneous send of 8MB of data		
	256 node	512 node
average	587 MB/s (78.2%)	582 MB/s (77.6%)
minimum	559 MB/s (74.6%)	434 MB/s (57.9%)
maximum	619 MB/s (82.6%)	630 MB/s (84.0%)
(b) Global sum over 256 nodes(in milliseconds)		
	8B/node	800kB/node
average	0.420	257
minimum	0.344	52
maximum	0.727	491

Table 2: Preliminary test of network performance

In Table 2 (b) time in milliseconds for a global sum using MPI ALL-Reduce command is given. These are also reasonable numbers given 10–20 microseconds of latency of the network.

3. Physics plan

Our primary physics plan for PACS-CS is the $N_f = 2+1$ full lattice QCD simulation using the fully $O(a)$ improved Wilson-clover quark action and the Iwasaki RG gluon action. This program has been pursued by the joint CP-PACS and JLQCD collaborations over the past several years. So far, the lightest up and down quark mass is limited to 65MeV or so corresponding to $m_\pi/m_\rho \approx 0.6$. While the results on the hadron spectrum and light quark masses estimated for the physical point in the continuum limit are interesting and encouraging[4, 5], large uncertainties remain due to the long chiral extrapolation toward the physical up and down quark mass of 3.5MeV.

In order to advance the simulation, we take advantage of the algorithmic developments offered by the Lüscher's idea[6] to combine domain-decomposition with the multi-time step molecular dynamics evolution to accelerate the hybrid Monte Carlo algorithm. Our preparatory study of the algorithm[7] shows that the acceleration and robustness of the algorithm for light quark masses observed for the $N_f = 2$ flavor case holds for the $N_f = 2+1$ case as well. In fact, the shrinking of minimum eigenvalue distribution with volume [8] appears even more prounced perhaps due to dynamical strange quark. The polynomial Hybrid monte carlo algorithm for the strange quark has also been accelerated recently with a UV preconditioner technique[9].

We are therefore hopeful that the up and down quark mass may be lowered close to the physical value, thus allowing us to minimize or even avoid the use of effective theories such as chiral perturbation theory to extract physics from lattice QCD simulations.

Studies in the light hadron sector has to be augmented with those of the heavy quark sector. We plan to use the relativistic heavy quark formalism[10], which has the virtue of allowing the continuum limit to be taken. Our extensive tests for charm[11] and bottom[12] systems suggest that the continuum limit could be controlled even for the bottom system using the lattice spacing range of 2 to 4 GeV.

Toward these goals, we have started the first set of production runs at $\beta = 1.9$ on a $32^3 \times 64$ lattice using 256 node partitions of the PACS-CS system. Three values of the up and down hopping parameter are now simulated, $\kappa_{ud} = 0.13754, 0.13770, 0.13781,$ for which we estimate the AWI quark mass to be 25MeV, 15MeV and 7MeV. The runs are still in the thermalization stage, but the algorithm appears to be functioning well.

Acknowledgments

This work is supported in part by Grants-in-Aid for Scientific Research from the Ministry of Education, Culture, Sports, Science and Technology (Nos. 13135204, 13135216, 15540251, 16540228, 16740147, 17340066, 17540259, 18104005, 18540250, 18740130).

References

[1] PACS-CS Collaboration, S. Aoki *et al.,* PoS LAT2005 (2005) 111.

[2] `http://www.pccluster.org/index.html.en`

[3] S. Sumimoto, K. Kumon, T. Boku, M. Sato, A. Ukawa, Proc. of IPS SIGHPC report 2005-HPC-103 (in Japanese) (August 2005).

[4] CP-PACS and JLQCD Collaborations, T. Ishikawa *et al.,* these proceedings, PoS LAT2006 181.

[5] CP-PACS and JLQCD Collaborations, T. Yoshie *et al.*, these proceedings, PoS LAT2006 204..

[6] M. Lüscher, JHEP **0305** (2003) 052; Comput. Phys. Commun. **165** (2005) 199.

[7] PACS-CS Collaboration, Y. Kuramashi *et al.,* these proceedings, PoS LAT2006 029.

[8] L. Del Debbio *et al.,* JHEP **0602** (2006) 011.

[9] PACS-CS Collaboration, K.-I. Ishikawa *et al.,* these proceedings, PoS LAT2006 027.

[10] S. Aoki, Y. Kuramashi and S. Tominaga, Prog. Theor. Phys. **109** (2003) 383.

[11] CP-PACS Collaboration, Y. Kayaba *et al.,* Nucl. Phys. B(Proc. Suppl.) **140** (2005) 479.

[12] CP-PACS Collaboration, Y. Kuramashi *et al.,* PoS LAT2005 (2005) 226.