MegaProto: 1 TFlops/10 kW Rack Is Feasible Even with Only Commodity Technology

Hiroshi Nakashima* Hiroshi Nakamura† Mitsuhisa Sato‡ Taisuke Boku‡
Satoshi Matsuoka§ Daisuke Takahashi‡ Yoshihiko Hotta‡

* Toyohashi University of Technology, nakasima@tutics.tut.ac.jp
† University of Tokyo, nakamura@hal.rcast.u-tokyo.ac.jp
‡ University of Tsukuba, {msato,taisuke,daisuke,hotta}@hpcs.cs.tsukuba.ac.jp
§ Tokyo Institute of Technology, matsu@is.titech.ac.jp

Abstract

In our research project "Mega-Scale Computing Based on Low-Power Technology and Workload Modeling", we claim that a million-scale parallel system could be built with densely mounted low-power commodity processors. "MegaProto" is a proof-of-concept low-power and high-performance cluster build only *with commodity components to implement this claim. A one-rack system is composed of 32 motherboard "cluster units" of 1 U-height and commodity switches to interconnect them mutually as well as with other racks. Each cluster unit houses 16 low-power dollar-bill-sized commodity PC-architecture daughterboards, together with a high bandwidth, 2 Gbps per processor embedded switched network based on Gigabit Ethernet. The peak performance of a one-rack system is 0.48 TFlops for the first version and will improve to 1.02 TFlops in the second version through a processor/daughterboard upgrade. The system consumes about 10 kW or less per rack, resulting in 100 MFlops/W power efficiency with a power-aware intra-rack network of 32 Gbps bisection bandwidth, while additional 2.4 kW will boost this to sufficiently large 256 Gbps. Performance studies show that even the first version significantly outperforms a conventional high-end 1 U server comprised of dual power-hungry processors in a majority of NPB programs. It is also investigated how the current automated DVS control could save power for the HPC parallel programs along with its limitation.*

SC|05 November 12-18, 2005, Seattle, Washington, USA

1. Introduction

Our research project "Mega-Scale Computing Based on Low-Power Technology and Workload Modeling" aims to establish fundamental technologies for million-scale parallel systems to achieve Peta-Flops computing. Our research focuses on the feasibility, dependability and programmability of Mega-Scale computing systems.

Central to our claim is that, sharing ideologies with other recent work on low power and power aware HPC efforts, it is possible to fully utilize recent breed of commodity power aware CPU and other components, to achieve high ratios in compute/power and compute/density metrics. For that purpose, we need to employ components that exhibit high compute/power ratio in the first place.

Now, to achieve the best power savings, we additionally claim that, it is best to employ a processor whose (a) difference in power consumption between the minimal and maximal DVS and CPU states are significant, much greater than conventional CPUs such as AMD Opteron, and (b) response to changing workload is quite rapid, in the order of under millisecond or less. This will allow precise, fine-grain tracking of workload compared to coarse-grained methods as proposed in[6, 8].

We attempt to demonstrate the viability of our approach by constructing an exemplar of such a low-power system using currently available commodity technologies. The prototype system is named *MegaProto*[16], in which large number of low-power processors that (almost) fulfill the requirements (a) and (b) above are interconnected with highly reliable and high-speed commodity network. MegaProto is also used as our platform to implement our software research results on low-power compilation, dependable cluster management, and scalable programming, the combinations by which we aim to achieve *Mega-Scale*.

The building block of MegaProto is a 1 U-high 19 inch-

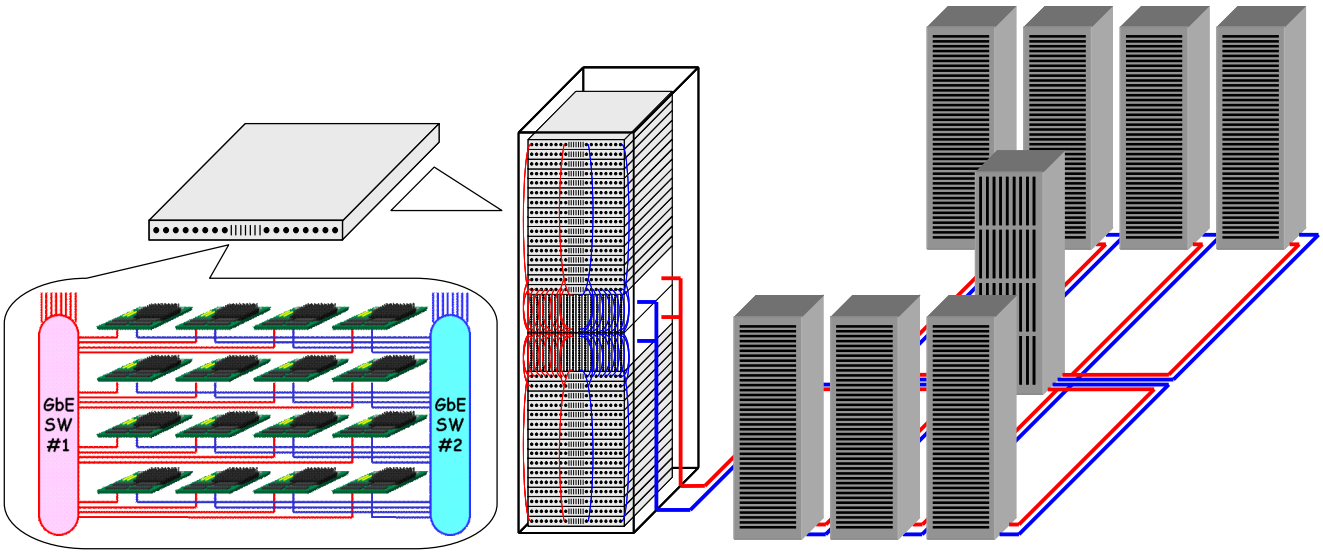


Figure 1. System Configuration

rack mountable motherboard *cluster unit* on which 16 low-power, a dollar-bill-sized, commodity PC-architecture daughterboards are mounted along with a high bandwidth, 2 Gbps per processor embedded switched network based on Gigabit Ethernet on the motherboard. The peak performance of each unit is 14.4 GFlops for the first version and will improve to 32.0 GFlops in the second version through a processor/daughterboard upgrade. The intra- and inter-unit network bandwidths are 32 Gbps and 16 Gbps respectively. As for power consumption, the entire unit idles at less than 150W and consumes 300-320 W maximum under extreme computational stress; this is comparable to or better than conventional 1 U servers comprised of dual high-performance, power-hungry processors. An aggregation of 32 cluster units comprises a one-rack system that exerts 0.48 TFlops and 1.02 TFlops in the first and second versions respectively, while the system consumes 9.6 to 12.6 kW depending on the version and inter-unit network configuration.

We show that, with MegaProto we actually attain considerable power savings without sacrificing speed on a variety of NAS parallel benchmarks. In fact, the power/density characteristics is much greater than traditional 1U servers, sometimes by over a factor of two, even for the older version of our prototype employing a processor that is half the speed (Transmeta Crusoe TM5800 vs. Efficeon 8820). We also illustrate how some components such as networks that are not *power-aware* other than the CPU will become more dominant as we succeed in saving CPU power, suggesting future efforts towards making every aspect of the system power aware while still being commodity.

2. MegaProto Architecture

2.1. Overview

As shown in Figure 1, a MegaProto system consists of one or more 40-42 U 19-inch standard racks, each of which contains 512 low-power commodity processors mounted on 32 *cluster unit* boards of 1 U size, and up to 32 commodity Gigabit Ethernet (GbE) switches of 24 ports which occupy 8 U packaging space in total. A 1 U cluster unit has 16 processor cards on which a TM5800 (Crusoe, 1st ver.) or TM8820 (Efficeon, 2nd ver.) is mounted together with 256/512 MB memory and other peripherals. It also embodies a pair of *on-board* GbE switched subnetworks to connect processors on it through their dual GbE ports and to provide 16 GbE uplinks for inter-board connection.

The first-version one rack system exhibits 479 GFlops peak performance, while the second version will achieve 1.02 TFlops. Both versions operate with considerably low power, 9.6–10.2 kW with minimum inter-board network configuration and 12.0–12.6 kW with maximized configuration of a 16-child/8-parent fat tree. The overall system will show a good performance/power ratio, 39.8–49.8 MFlops/W in the first version, and 81.3–100.0 MFlops/W in the second version.

The following subsections discuss the detail of the system components in a bottom-up manner.

2.2. Processor Card

A processor card of 65mm $\times$ 130mm (Figure 2, a little smaller than a dollar bill) contains an MPU, 256 MB/512 MB main memory, flash memory, PCI(-X) interface, and other peripheral components. We designed MegaProto

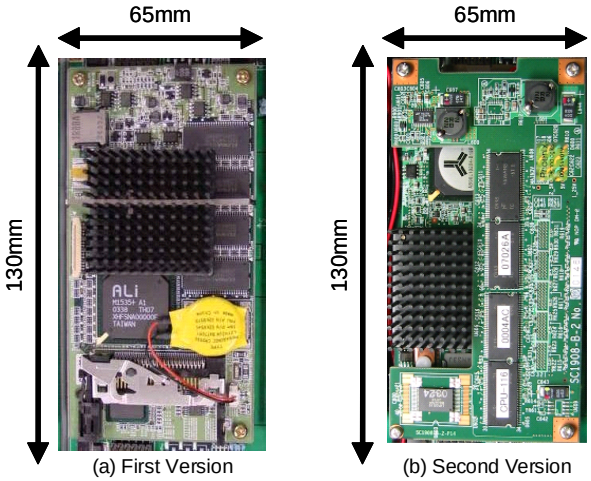


Figure 2. Processor Card

	1st version	2nd version
MPU	TM5800 (0.93GHz)	TM8820 (1.0GHz)
TDP	7.5 W	3 W
Peak Power/Perf	124.0 MFlops/W	666.7 MFlops/W
Caches	L1=64KB(I)+64KB(D)	L1=128KB(I)+64KB(D)
	L2=512KB(D)	L2=1MB(D)
Memory	256 MB SDR-133	512 MB DDR-266
Flush	512 KB	1 MB
I/O Bus	PCI (32 bit, 33 MHz)	PCI-X (64 bit, 66 MHz)

Table 1. Processor Card Specification

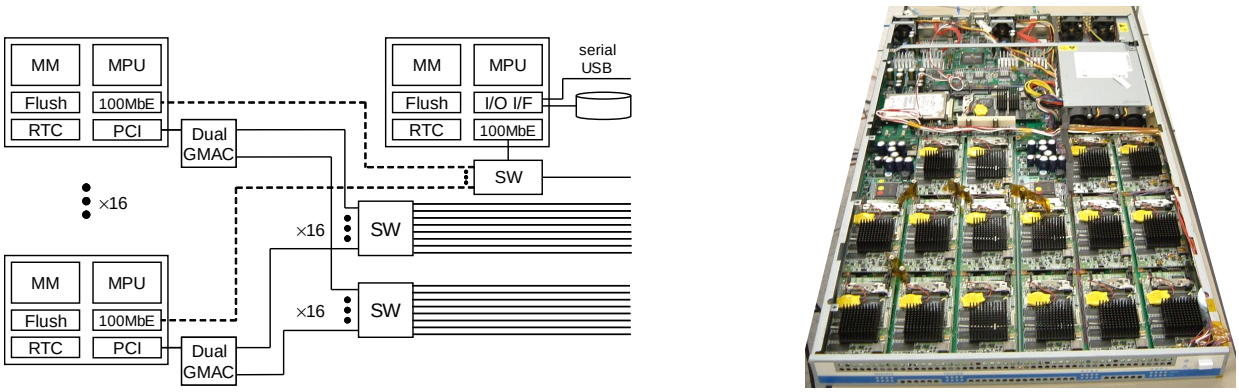


Figure 3. Cluster Unit

to be *evolutional* so that we can stay in sync with the progress of the mobile processor technology. In order to actually experiment whether such evolutional performance improvements would be feasible, we designed two versions of processor cards while the cluster unit motherboard is common to both versions.

As shown in Table 1, the major revision from the first to second version is the microprocessor replacement from TM5800 to TM8820. Both processors are superior to other commodity processors in their 130 nm and 90 nm generations with respect to power efficiency. In particular, TM8820 exhibits 10-fold advantage over high-end processors at 667 MFlops/W, and 3- to 5-fold even when compared with other mobile processors[16, 21].

The improvement of power efficiency for TM8820 also makes it possible to enhance the processor peripherals so that we may allocate a larger portion of our 10 W power budget alloted each processor card. For example, we enlarge the memory capacity from 256 MB to 512 MB also enhancing the memory bandwidth using DDR-266 in place of SDR-133.

Another important improvement is I/O bus performance. In the first version, due to the limited power budget and processor performance, we used 32 bit/33 MHz PCI that is only just enough to fill the bandwidth of a single GbE network. The second version, however, has a sufficiently large bandwidth of 64 bit/66 MHz PCI-X bus to fully utilize the GbE link pair. These improvements not only of the processor but also of its peripherals ensure that both versions exhibit good balance between computation, memory access and communication.

2.3. 1 U Cluster Unit

As shown in Figure 3, the 1 U cluster unit has 16 processor cards to have 14.9 GFlops peak performance in the first version, while the second achieves 32.0 GFlops. The cluster unit is packaged in a $432\text{mm(W)} \times 756\text{mm(D)} \times 44\text{mm(H)}$ chassis. About half of the chassis (front half of the picture) is for 16 processor cards, while the other half is occupied

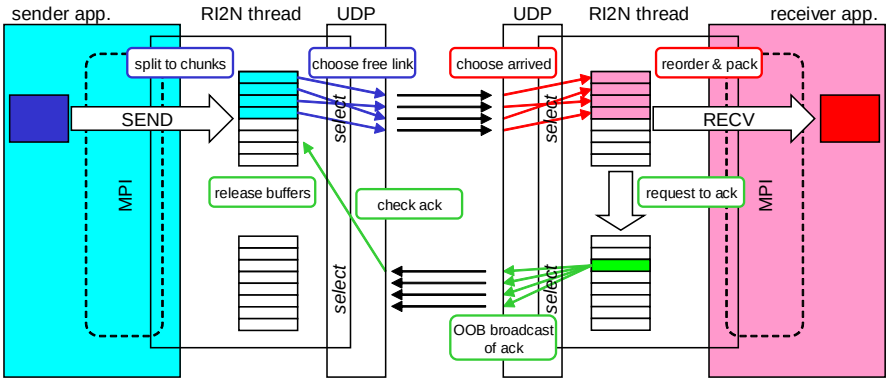


Figure 4. Architecture of RI2N

by network switches, a management processor and a power supply. The total power consumption of a cluster unit at the maximum rating is 300 W for the first version and 320 W for the second, both of which can be sufficiently air-cooled using four low-speed fans under normal operating conditions.

The major components, besides processor cards, mounted on the cluster unit motherboard, are for dual GbE networks, which our RI2N (Redundant Inexpensive Interconnection Network)[15] technology utilizes for both high bandwidth and fault tolerance as is discussed in Section 2.4. From each processor card, a PCI-X bus is extended to the motherboard to connect a dual-port GbE media access controller chip (GMAC). Each GbE port is connected to a 24-port Layer-2 GbE switch of 20 Gbps backplane bandwidth from which 8 uplinks of 1000Base-T are extended to outside of the cluster unit. Since we have two GbE networks with individual switches, the total bandwidth within a cluster unit is 32 Gbps, while inter-cluster bandwidth of two uplink bundles is 16 Gbps.

Another important component on the motherboard is a management processor whose configuration is very similar to a processor card[1], but it also has I/O interfaces for a 60 GB hard disk, a USB port and a serial port. All the processor cards and the management processor are connected by a 100 Mbps Fast Ethernet management network through which Linux operating system is booted from the hard disk of the management processor by each diskless processor. The main file system of the cluster unit may reside externally and be accessed through the GbE networks. The management processor is also responsible for housekeeping tasks including health check of processors and configuration management of GbE switches.

The cluster unit is the building block of the MegaProto large-scale cluster, but, at the same time, it can be considered as a small-scale 16-node cluster. Each processor card acts as an independent PC node with a full Linux software stack and cluster middlewares such as MPI. Although a system of multiple cluster units will have physical *boundaries* of networking between units and in higher level interconnection, those boundaries are transparent for programmers in logical sense so that they may assume MegaProto is a *flat* cluster, logically.

2.4. Networking

As described in Section 2.3, each processor has two GbE ports, each port is connected to one of dual GbE network switches on a cluster unit motherboard, and each switch is a terminal member of one of the dual inter-board system-wide network with 8 uplinks. This duality is managed and utilized by our high-bandwidth and fault-tolerant networking technology named RI2N[15].

As shown in Figure 4, RI2N is a user-level middleware to bridge applications (and messaging library such as MPI) and UDP communication layer. On the sender side, it decomposes a message from an application into *chunks* of a few kilo-bytes and transmits each chunk through one of the ports chosen by `select()` system call. This simple mechanism works both for high-bandwidth and fault-tolerance because `select()` chooses a ready-to-send port and avoids congested and/or faulty paths. Then the receiver-side RI2N reorders transmitted chunks by the serial numbers attached to them and sends an out-of-band acknowledgment back to the sender through all the ports periodically. The acknowledgment is used to release chunk buffers in the sender whose contents would be resent through a healthy path if the acknowledgment indicates that one or more chunks are lost.

In order to be tolerant of faults not only at links but also in switches, both of intra- and inter-board networks have completely separated two subnetworks as shown in Figure 5. An inter-cluster subnetwork may be configured with variety so that we trade off its bandwidth and cost.

1 The management processor for both first and second version is TM5800.

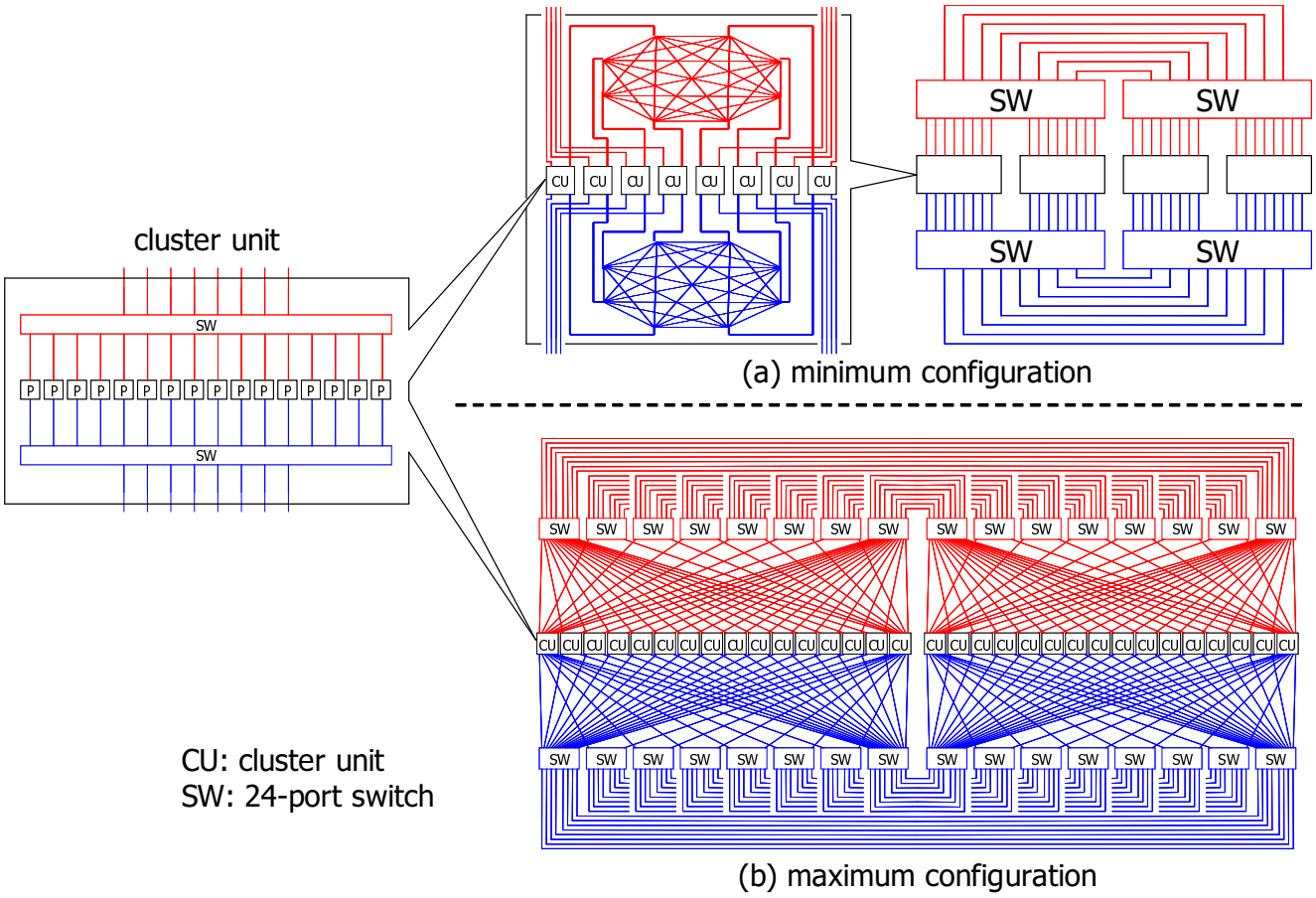


Figure 5. System-Wide Network Configuration

For example, the minimum configuration for a one-rack 32-unit system (Figure 5(a)) has only four 24-port commodity switches (two for each subnetwork) to connect 32 cluster units in a rack but its bisection bandwidth is limited to 32 Gbps[2].

On the other hand, the maximum configuration with 32 switches (16 for each) packaged in 8 U space has a considerably large bisection bandwidth of 256 Gbps as shown in Figure 5(b). It may also be extended to a multiple-rack network to form a fat-tree with node switches having 16 children and 8 parents by connecting top-level links of the figure to the switches outside of the rack. For example, an 8-rack system of 256 units (4096 processors) can be connected by a 2-level fat-tree using 128 commodity switches (64 for each) with 16 ports which will need only one additional rack.

In each configuration of inter-board networking, the routing to exploit multiple shortest paths is managed utilizing VLAN of layer-2 switches[14]. For example, the maximum configuration subnetwork consists of 64 tree-shaped VLAN partitions from which we choose a tree and assign its path to a sender/receiver pair in order to distribute traffic as evenly as possible.

3. Performance Evaluation

This section shows preliminary results of our performance evaluation of the first version (i.e. TM5800 version) cluster unit[3]. First Section 3.1 shows the performance numbers measured by five class A kernel benchmarks of NPB 3.1[2] (IS, MG, EP, FT and CG) and HPL 1.0a[18] with the matrix size parameter $N = 10,000$. The benchmark programs are compiled with gcc/g77 version 3.3.2, linked with LAM-MPI version 7.1.1, and executed under Linux kernel version 2.4.22mmpu. We also measured the performance of a dual-Xeon 1U server with similar software configuration of gcc/g77 3.4.3, LAM-MPI 6.5.6 and Linux kernel 2.4.20-20.7smp. The performance compari-

2 8 links times 2 (bi-) directions times 2 subnetworks makes 32 of 1 Gbps bandwidth.

3 Although the second TM8820 version is now available, it is still in testing/tuning phase.

# of proc.	NPB (class A)[Mop/s] IS	MG	EP	FT	CG	HPL [GFlops]
2	10.1	153.1	5.0	(*1)	95.6	(*1)
4	17.4	262.6	10.0	257.9	115.7	2.07
8	29.6	507.9	19.9	476.4	163.4	3.61
16	52.3	831.6	39.8	923.9	217.5	5.62

(*1) N/A due to memory shortage.

Table 2. Performance of NPB and HPL

son between the MegaProto cluster unit and the dual-Xeon server is shown in Section 3.2. Then, after discussing our power measurement environment in Section 3.3, we show the power measurement result and our analysis on it emphasizing the effect of automated DVS in Section 3.4. Finally based on the result, Section 3.5 discusses the power management issues for the low-power and high-performance computing.

3.1. Performance and Speed-up

Table 2 shows the performance numbers of NPB and HPL from 2 to 16 processors[4]. The speedup relative to four-processor performance is also shown in Figure 6.

The results of NPB indicate that the MegaProto cluster unit behaves as if it were a reasonably well-tuned conventional PC cluster using commodity networking: good speedup for EP, FT and MG; not excellent but still large speedup for IS; and relatively small parallel efficiency for CG due to the lack of scalability. As for HPL, the 16-processor performance 5.62 GFlops or 38 % of the peak is a little bit lower than expected. This is mainly due to the relatively small memory space, 256 MB per processor, which limits the problem size before processors exert full computational power. Therefore, it is anticipated that the second version with 512 MB memory will show much better peak/sustained ratio.

3.2. Comparison with Dual-Xeon Server

Although we showed the performance of the MegaProto cluster unit is scalable in terms of their *absolute* performance, how would it compare relatively to ordinary servers and clusters with high-end processors? As such, we measured the performance of a 1U dual-Xeon server, Appro 1124Xi, which has two Xeon processors at 3.06 GHz and 1 GB DDR memory. Since its total TDP and peak performance of processors, 170 W and 12.2 GFlops respectively[12], and its maximum AC power rating of the entire 1U system, 400 W, are all comparable to our cluster unit with 16 processors, the dual-Xeon server is a good benchmark for performance comparison.

4 Single processor performance was also measured but is omitted because it is extremely low due to frequent memory swap and distorts the speedup results.

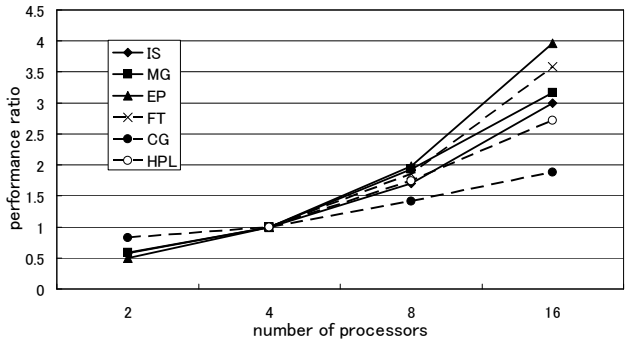


Figure 6. Speedup of NPB and HPL

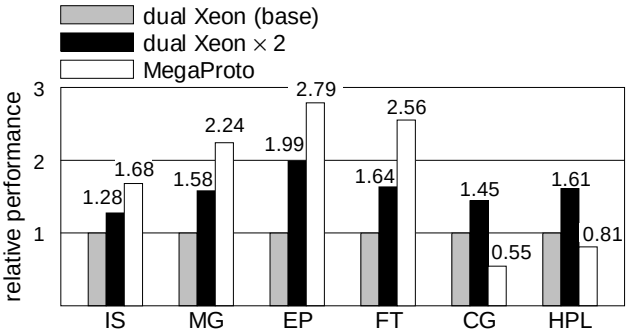


Figure 7. Performance Relative to Dual-Xeon Server

The comparison results shown in Figure 7 demonstrate the advantage of our approach. Our cluster unit of 16 processors greatly outperforms the dual-Xeon server in four NPB kernels, IS, MG, EP and FT. The most significant result is EP in which our cluster unit is 2.8 times as fast as the dual-Xeon server. More notably, it is also significantly faster than the two-node GbE connected dual-Xeon server (i.e. a four-processor Xeon cluster) by 30 % for IS, and more than 40 % for other three benchmarks. These results clearly support our claim that a large scale aggregation of low-power processors could be much more efficient than a cluster of high-end but (or thus) smaller number of processors.

The advantage of our first version MegaProto is not unanimous, however. It is almost twice as slow on CG and also loses by a close margin in HPL. As discussed in Section 3.1, however, some of this is attributed to rather small memory capacity of each node, a significant part of which is occupied by OS kernels and system daemons, leaving about 220 MB for actual applications. In the second version, memory capacity will be doubled, as well as the peak floating point performance of 90 nm TM8820 being twice as fast as our current TM5800 (0.93 GFlops vs. 2.0 GFlops),

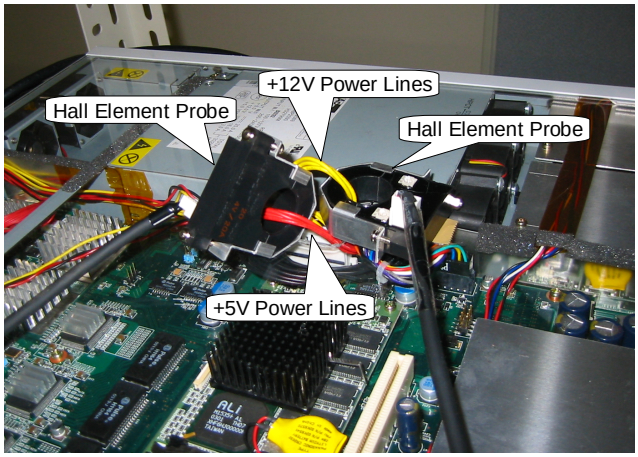


Figure 8. Power Measurement using Hall Element Probes

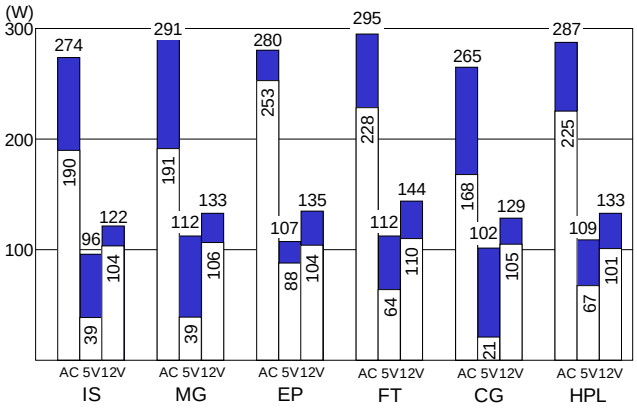


Figure 9. Peak and Average Power Consumption

plus the fact that I/O network performance will scale accordingly as well. By comparison, the speed improvements of 90 nm Xeons compared to 130 nm counterparts are minimal; as such, we expect that our second version will exhibit landslide victory in 90 nm generation comparison.

3.3. Power Measurement

For the MegaProto system, its power consumption is as important as its execution speed. A detailed power profile showing when and where it consumes power is also necessary to analyze its behavior for improved design in future. Thus we need a tool to measure the power consumption as precisely as possible in terms of time and space resolution.

Our requirement is satisfied by an electric current measurement tool with Hall elements[10]. This tool has ring-shape Hall element probes through which power lines to be measured are *threaded* as shown in Figure 8 without any electrical contacts nor interferences. The tool also has A/D converters to digitize measured power currents and to transfer them to a PC with a fine-grained time resolution as fine as 10 μs.

Using this tool, we measured the power current of 100 V AC input of the power supply together with its +5 V and +12 V DC outputs. The +5 V power line is provided to processors via voltage regulation modules, while +12 V and its converted branches drive processor peripherals including memories and network devices which are the major consumers.

3.4. Power Consumption

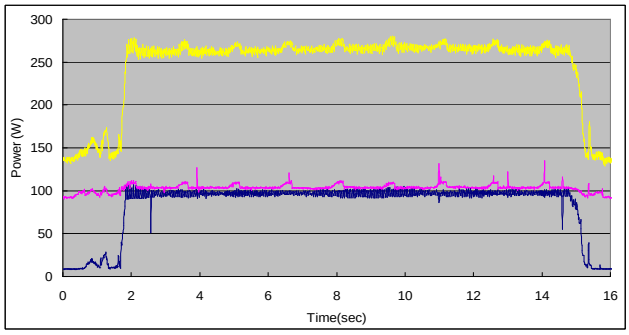
Another important evaluation result, measured power consumption, is summarized in Figure 9. The figure shows the peak (dark bars) and average (white) power consumption of AC and two DC sources of each benchmark executed with 16 processors[5]. Since all benchmarks do exhibit periods of maximum computational intensity (albeit with varying durations), the peak power consumption values are by and large similar and are at nearly maximum rating. The average of +5 V power for processors, however, varies from 21 W of CG to 88 W of EP and AC power follows the trend.

This variance is more clearly observed from the power profiles shown in Figure 10. For example, the profile of EP (a) shows nothing interesting because the program almost thoroughly concentrates on the computation of pseudo random numbers followed by a few reduction communications. Thus its execution gives almost *flat* CPU power consumption at nearly maximum rating.

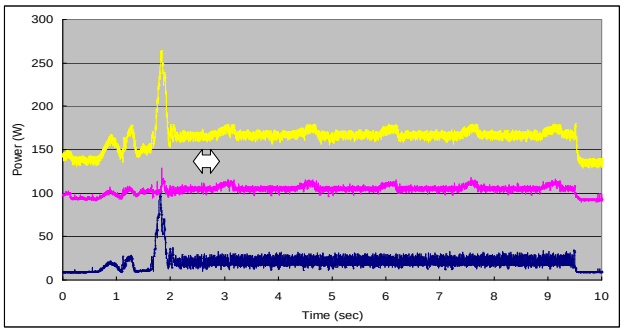
Another flat CPU power profile is observable in CG as shown in Figure10(b), but the reason of the flatness is quite different. As stated before, CG is a communication bound program and about 60 % of its execution is spent for communication. After a power peak corresponding to the initialization of the sparse matrix whose largest eigenvalue is to be found by the program, the CPU power is stably low in the 15 iterations of the conjugate gradient (CG) method to solve the linear system. Figure 11(a)[6] shows a representative execution profile of a processor in one of CG method procedure (indicated by a double-headed arrow in Figure 10(b)), in which the white part corresponds to computation while communication parts are painted dark. The profile of one of the main 25 iterations in the procedure is also shown in the lower part of the figure, from which we can observe the it-

5 The sum of DC powers are not equal to the AC power because of approximately 20 % loss of AC/DC conversion and small power consumed by processor peripherals.

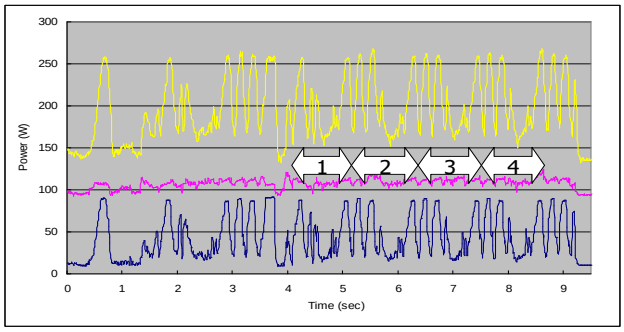
6 Due to a minor technical problem, this profile is obtained by an execution with LAM-MPI version 6.5.9.



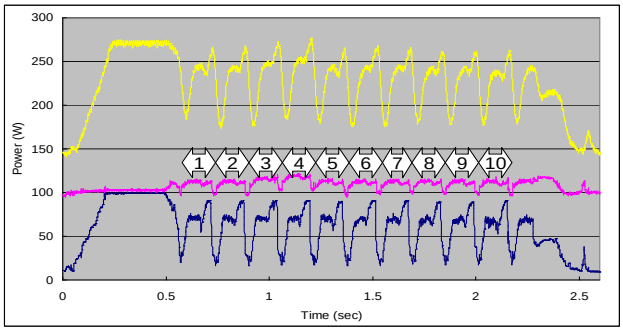
(a) EP



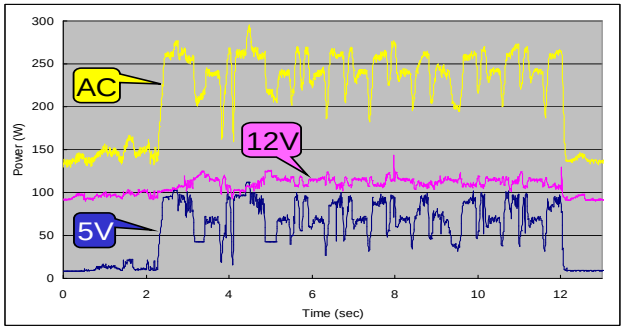
(b) CG



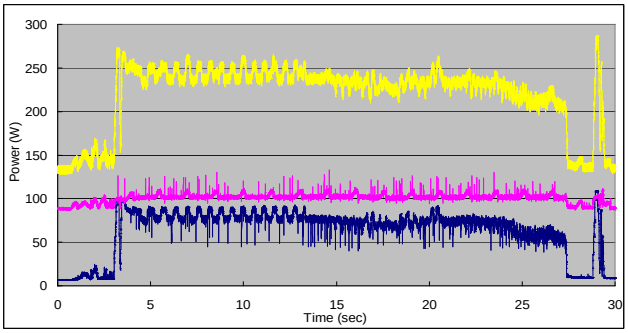
(c) MG



(d) IS



(e) FT



(f) HPL

Figure 10. Power Profile

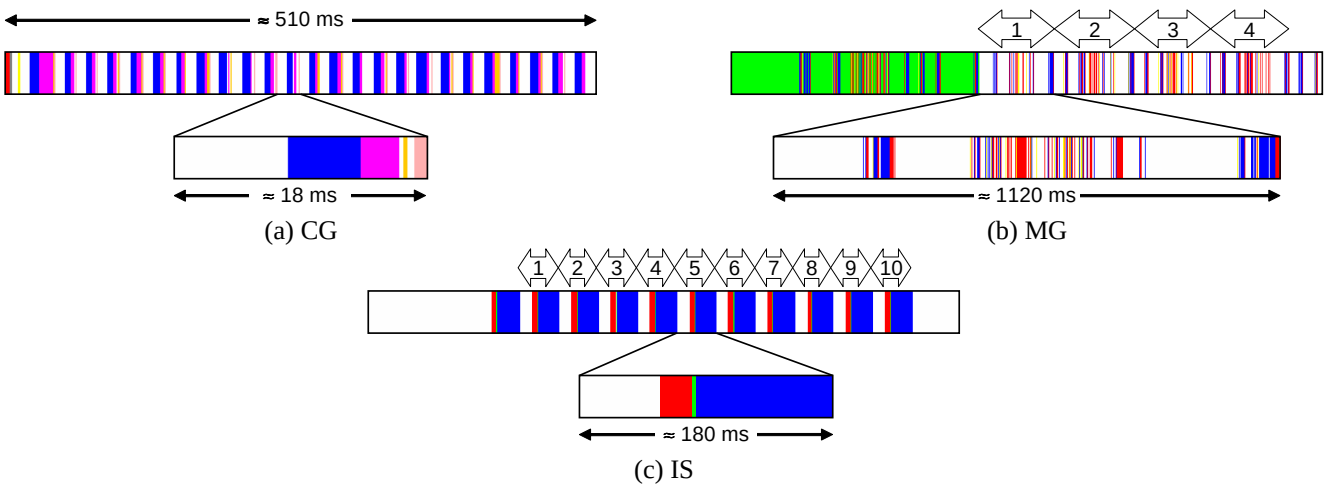


Figure 11. Execution Profile

eration has a computation phase followed by four communication phases three of which are manual all-reduce type communications. Since these communications may wait at length for message arrival, a processor frequently spends idle periods long enough for automated DVS to lower the power to the minimum level. Then, however, since the intermittent computation periods of about 8 ms may be too short for the DVS software to *shift its gear up* to the maximum level[7], the power level stays low when the computation resumes.

Other programs show power consumption behaviors between these two extremes. For example, the power profile of MG shown in Figure 10(c) exhibits frequent shifting of the gear with up on computation and down on communication. By closer look of the profile, we find four iterations of V-cycle multigrid operations indicated by double-headed arrows in the figure. As observed from the execution profile shown in Figure 11(b)[6], an operation has three major neighboring communications through the boundary surface of the finest 3D grid, which correspond to deep *valleys* of the power profile. In this case, however, a computation phase between communications is enough long so that the automated DVS shifts the gear up to the maximum level to exert full computation power.

The profile of IS shown in Figure 10(d) also has twelve peaks. The first and the largest peak is for initialization to generate a data set to be sorted, while the last one corresponds to the verification of the sorting. Other ten peaks represent ten iterations of the bucket sort routine for which DVS gear is automatically upshifted for the local ranking and bucketing. The valleys separating ten peaks correspond to all-to-all communication, which is sufficiently long for DVS gear to be shifted down, for the redistribution of the keys in the local buckets. The execution profile shown in Figure 11(c) has some resemblance to that of CG, but their power profiles are quite different because the computation phase of IS is enough long (about 60 ms) and the all-to-all type communication (darkest bar[8]) involves a significant amount of computation.

Other two profiles for FT and HPL shown in Figure 10(e) and (f) also exhibits frequent gear shifts. Their valleys, however, are relatively shallow and narrow because the amount of communication in these programs, 3D-array transposition in FT and the broadcast of pivot panel in HPL, relative to the computation is smaller than other programs.

These profiles, excepting that of EP, clearly proves that the total energy for the execution of HPC parallel programs is efficiently saved by shifting the DVS gear down automatically in their communication phases. This is good news for low-power high-performance computing because it is inevitable for HPC parallel programs to have some communication phases which often are of significantly long duration. For most parallel SPMD programs it is reasonable and/or sometimes superior to leave DVS control to be automatic, since the workload are memory and communication sensitive, and may change rapidly over time. This is in contrast to previous work such as [6, 8] that set DVS gears manually in a coarse-grained fashion. Thus, *if* network devices saves their power in computation phases, the system level power will be effectively saved in total.

However, the result shown in Figure 9 and 10 reveals the power-unawareness of network devices. From these graphs we find that the +12 V power line is fairly stable unlike +5 V for processors. Since the major consumers of the +12 V source are network devices, it may seem initially odd that neither the frequency nor the data size of communication

7 The hardware-level response time is 20 μsec or less[20].

8 Or blue bar if you view it chromatically

has any effect on the power consumption of the networks, resulting in stable as well as non-negligible power consumption. The reason for this is that the current networking chipsets employed *always* drive network links even when no data is transmitted. This power-unawareness of network devices may be satisfactory for standalone usage as office networking components, but would be a serious impediment for low-power, high-density cluster systems. We will revisit this issue later.

3.5. Power Management for HPC

Our evaluation result of the power consumption suggests the effectiveness of DVS for the low-power high-performance computing. This section gives a further discussion of the issue.

Most SPMD applications embody fairly independent computation and communication phases. Because of this, we claim that, in most cases we only need to establish the *maximal* proper *gear setting* of the DVS so as to maximize some appropriate power-saving metric (such as the energy-delay product) for the computational phase, while for the communication phase control can be relinquished to automated DVS control.

This is especially applicable to processors such as TM5800 Crusoe where the difference in the power consumption level is drastically reduced with DVS (for Crusoe it is almost an order of magnitude). Since most of the work for communication will be performed by the off-CPU DMA and communication engine, the automated DVS should achieve extremely low power CPU operation during communication without sacrificing performance. Various MegaProto benchmarks exemplify this, such as MG and IS in which the energy of computation is limited to less than 40 % of the maximum level.

The important factor is the granularity of DVS control versus the frequency of communication for each SPMD communication loop iterations. If the former is significantly smaller than the latter, upon the transition from the communication to the computational phase there would be very little overhead. On the other hand, if the DVS granularity is relatively significant, it will affect performance as we observe in the CG benchmark. Our evaluation strongly suggests that the response time of the software control of DVS is larger than a few millisecond and to follow the fine-grained computation/communication switching in CG, while the DVS hardware can shift its gear up in 20 μsec or less[20]. If we have no means to make the software response quicker, the only solution here then would be to statically set DVS gear to some fixed level without dynamic control, at least during loop execution.

The longer-term technological trend indicates that DVS control granularity will be significantly reduced to sub-microsecond levels in hardware and thus expectedly to sub-millisecond in software; this could be offset somewhat by finer grained computing elements to achieve low power (more parallelism instead of higher voltage/clock frequency), but we expect that the former will advance much faster than the latter. Moreover, since the processors will at least not slow down compared to the present, unless for some unknown reasons we move to much smaller problem sizes per processor, we expect that that the communication granularity reduction will be slower. One caveat is that, although this observation will be applicable to most SPMD applications, for largely irregular/asynchronous (such as AMR), we may need to perform explicit control of interactions between communication and computation to achieve proper latency hiding, as has been suggested by Chen et al[3].

As has been evidenced, we do need facilities to exert DVS control to networks as well. One advantage here is that, compared to computation where it is harder to predict how much we *will* compute in general, it is easier to estimate how much data are being transferred. Because of this it may be sufficient as well as more effective to perform explicit control rather than resort to automated means, that may sacrifice performance.

4. Related Work

Green Destiny[22] is the first successful attempt to build a high-performance cluster with low-power processors. Its building block is a 3 U-high blade housing a TM5600[9] processor of 667 MHz, DRAM modules of 640 MB and a hard-disk drive of 10 GB. Since 24 blades are mounted in a chassis fitting for 19-inch rack, the packaging density per 1 U is 8-processor, which is half of MegaProto. The peak-performance/power ratio of the blade is 35.6 MFlops/W, which is about 2/3 of the first version of MegaProto and about 1/3 of its second version. Another notable difference between Green Destiny and MegaProto is found in network bandwidth. Since a processor of Green Destiny has a single 100 Mbps Fast Ethernet port[10], its per-processor and per-Flops bandwidth are about 1/20 and 1/7 of MegaProto respectively. Although performance of standard benchmarks of Green Destiny has not been published, its scalability problem due to narrow bandwidth should be much severer than the first version of MegaProto that behaves relatively inefficient with communication bound programs as discussed in Section 3.1 and 3.2[11].

However, the success of Green Destiny made a great impact upon high-performance cluster architecture showing a

9 TM5600 is the predecessor of TM5800 for the first version of MegaProto.

10 A processor has two additional Fast Ethernet ports for management and auxiliary but they are not used for parallel computation.

11 The first version of MegaProto utilizes one GbE port only but its per-Flops bandwidth is still 7-fold of Green Destiny.

System	Green Destiny	BlueGene/L	Altix 3000	Earth Sim.	MegaProto(v.1)	MegaProto(v.2)
Processor	TM5600	custom (PowerPC 440)	Itanium 2	custom (vector)	TM5800	TM8820
Frequency[GHz]	0.667	0.7	1.5	0.5	0.93	1.0
# of Proc.	240	2048	64	16	512	512
Power[kW]	5.2	28.1	12.2	16.0	12.0	12.8
Area[m^2]	0.56	0.84	1.02	1.40	0.54	0.54
Volume[m^3]	1.13	1.64	1.95	2.80	1.08	1.08
Peak Performance						
/proc (GFlops)	0.667	2.8	6.0	8.0	0.93	2.0
/rack (TFlops)	0.16	5.73	0.38	0.13	0.48	1.02
/W (MFlops/W)	30.8	204.0	31.5	8.0	39.7	80.0
/m^2 (TFlops/m^2)	0.29	6.85	0.37	0.09	0.88	1.90
/m^3 (TFlops/m^3)	0.14	3.50	0.20	0.05	0.44	0.95
Memory						
/proc (MB)	640	256	1900	2048	256	512
/rack (GB)	150	512	119	32	128	256
/Flops (B/Flops)	0.94	0.09	0.31	0.25	0.27	0.25

Table 3. Comparison with Other Systems

low-power and high-density system can outperform ordinary high-ends. In fact, Green Destiny outperforms ASCI Q by 3-fold and 14-fold in power and space efficiency respectively with an N-body program.

Another heavy impact was made by BlueGene/L[1, 11] which was ranked No. 1 on the November 2004 TOP500 list[19]. One rack of BlueGene/L has 32 motherboards on which 16 daughterboards are mounted for each. Since two dual-processor chips of 5.6 GFlops reside on a daughterboard, a rack exerts 5.7 TFlops peak performance by 2048 processors while it consumes about 28 kW which results in a high power efficiency of 204 MFlops/W. Its Linpack performance 70.7 TFlops is achieved not only by the high peak performance but also by the 3D torus network with 1.4 Gbps links and the tree network with 2.8 Gbps links. Although BlueGene/L greatly owes its high absolute performance and power efficiency to the non-commodity ASIC processor and network chips and thus it is not a good reference of MegaProto, it is a clear proof of the superiority of low-power systems for high-performance computing.

Now we show various one-rack performance numbers of these two systems to compare them with those of both versions of MegaProto in Table 3. The table also includes the numbers of Altix 3000[17] and Earth Simulator[9] which are ranked second and third of the latest TOP500 list respectively. Note that the numbers of MegaProto are for the maximum network configuration with 32 cluster unit boards and 32 switches for inter-board connection discussed in Section 2.4. As stated above, BlueGene/L has substantial peak performance as well as its ratio to power, footprint and rack volume, which are superior to numbers of the second best Altix 3000 by one order of magnitude or more.

On the other hand, the numbers of the second version of MegaProto are significantly better than Altix 3000, the world fastest commodity processor based system, although they are second to the numbers of BlueGene/L. Its peak performance and power efficiency are about 2.5-fold of Altix, while its space and volume efficiencies are about 5-fold. Thus we may conclude our MegaProto is a top-level efficient cluster build by commodity technologies. As for the memory capacity, it seconds to BlueGene/L but is superior to other systems in the total capacity, and is comparable to Altix 3000 and Earth Simulator from the viewpoint of per-Flops capacity. Therefore MegaProto has an enough large memory as well as a good balance of memory and computation to exert its full performance with large scale problems.

The comparison of the systems clearly shows that those build by low-power processors, such as MegaProto and BlueGene/L, are superior to those with high-ends like Altix and Earth Simulator. Our observation is this superiority will continue or even grow in future. For example, we have already found that the power efficiency of low-power mobile processors has been improved more than high-ends in the 130 nm to 90 nm generation progress[16]. Another support is obtained from ITRS semiconductor roadmap[13] in which power consumption of high-end processors is forecasted to grow in 5 %/year pace while the pace of mobiles is expected to be 2.6 %/year.

Another important discussion of future trend is which of commodity or dedicated processors and devices will have advantage as the building blocks of high-performance and

low-power systems. Although the top of TOP500 would be earned by a dedicated technology based system such as BlueGene/L and Earth Simulator, we expected that commodity-based, our choice, will be the majority of the huge mass of high-performance computing which is for midrange systems around or up to 1 TFLops. In fact, many R&D projects are undertaken using low-power commodity processors as the element of medium scale clusters and/or those targeting business computing.

For example, Super Dense Server of IBM Austin Research Laboratory employs Ultra Low Voltage Pentium III of 300 to 500 MHz variable clock with SpeedStep to achieve an excellently low power of 12 W per node resulting in 41.7 MFlops/W, and relatively high density of 4.5 node per 1 U (by mounting 36 nodes in 8 U packaging space)[5]. Another example is Argus developed in University of South Carolina[7]. It has 128 PowerPC 750CXe processors of 600 MHz connected by Fast Ethernet, and has 38.4 MFlops/W power efficiency comparable to the first version of MegaProto and 0.72 TFlops/m^3 volume efficiency which is close to our second version. Commercial products of low-power clusters are also available from Orion Multisystems which puts a "Desktop Cluster" of 12 TM8800 processors[12] and "Deskside Cluster" of 96 processors on the market appealing their power and space efficiencies[4].

5. Conclusion

This paper described the design of our low-power and compact cluster for high-performance computing named MegaProto. The building block of the MegaProto system is a 1 U-high one-board cluster unit embodying 16 commodity low-power processors together with a pair of GbE networks also build by commodity networking technology. We designed two versions of the cluster unit with 0.93 GFlops TM5800 and 2.0 GFlops TM8820, both of which have a very low power consumption 300-320 W. Thus a cluster unit exerts a high peak performance of 14.9 GFlops and 32.0 GFlops in the first and second version respectively, which results in 0.48 TFlops and 1.02 TFlops one-rack system performance with 32 units and a set of commodity switches.

Our preliminary performance evaluation showed even the first version greatly outperforms a traditional 1U server in the execution of four NPB kernels. Since the second version has much higher performance and thus better power efficiency of 100 MFlops/W, it will undoubtedly prove the superiority of the low-power high-performance systems to the traditional high-end hot systems. We also showed the automated DVS of TM5800 successfully saves the energy of the execution of the benchmarks by shifting down the voltage and clock frequency in the communication phases.

Two units of the first version was shipped in March 2003 and have been used as the testbench of the second version design as well as a preliminary platform for our software development. The second version system with 20 units of 320 processors has been manufactured and is now in testing/tuning phase so that we will report its performance soon.

12 TM8800 shares a common processor core with TM8820 used for the second version of MegaProto.

Acknowledgments

The authors would express their appreciation to technical staff of IBM Japan for their contributions and support. This research work is supported by Japan Science and Technology Agency as a CREST research program entitled "Mega-Scale Computing Based on Low-Power Technology and Workload Modeling."

References

[1] N. R. Adiga et al. An overview of the BlueGene/L supercomputer. In *Proc. Supercomputing 2002*, Nov. 2002.

[2] D. H. Bailey et al. The NAS parallel benchmarks. *Intl. J. Supercomputer Applications*, 5(3):63–73, 1991.

[3] G. Chen, K. Malkowski, M. Kandemir, and P. Raghavan. Reducing power with performance contraints for parallel sparse applications. In *Proc. WS. High-Performance, Power-Aware Computing (included in Proc. IPDPS 2005)*, Apr. 2005.

[4] D. Costa. Orion puts a cluster on your desktop. http://www.workstationplanet.com/features/article.php/3437011, Nov. 2004.

[5] W. M. Felter et al. On the performance and use of dense servers. *IBM J. R & D*, 47(5/6):671–688, Sept. 2003.

[6] X. Feng, R. Ge, and K. W. Cameron. Power and energy profiling of scientific applications on distributed systems. In *Proc. Intl. Parallel and Distributed Processing Symp.*, Apr. 2005.

[7] X. Feng, R. Ge, and K. W. Cameron. Supercomputing in 1/10 cubic meter. In *Proc. Intl. Conf. Parallel and Distributed Computing and Networks*, Feb. 2005.

[8] V. W. Freeh, F. Pan, N. Kappiah, D. K. Lowenthal, and R. Springer. Exploring the energy-time tradeoff in MPI programs on a power-scalable cluster. In *Proc. Intl. Parallel and Distributed Processing Symp.*, Apr. 2005.

[9] S. Habata, M. Yokokawa, and S. Kitawaki. The Earth Simulator system. *NEC Res. & Develop.*, 44(1):21–26, Jan. 2003.

[10] Y. Hotta, M. Sato, D. Takahashi, and T. Boku. Measurement and characterization of power consumption of microprocesors for power-aware cluster. In *Proc. COOL Chips VII*, Apr. 2004.

[11] IBM Corp. *IBM e-Server Blue Gene Solution*, Nov. 2004.

[12] Intel Corp. *Intel Xeon Processor with 512-KB L2 Cache at 1.80 GHz to 3 GHz—Datasheet*, Mar. 2003.

[13] International Technology Roadmap for Semiconductors. Executive summary (2003 edition). http://public.itrs.net/Files/2003ITRS/Home2003.htm, 2003.

[14] T. Kudo, H. Tezuka, M. Matsuda, Y. Kodama, O. Tatebe, and S. Sekiguchi. VLAN-based routing: Multi-path L2 Ethernet network for HPC clusters. In *Proc. Cluster 2004*, Sept. 2004.

[15] S. Miura, T. Boku, M. Sato, and D. Takahashi. RI2N—interconnection network system for clusters with wide-bandwidth and fault-tolerancy based on multiple links. In *Proc. Intl. Symp. High Performance Computing 2004*, pages 342–351, Oct. 2003.

[16] H. Nakashima et al. MegaProto: A low-power and compact cluster for high-performance computing. In *Proc. WS. High-Performance, Power-Aware Computing (included in Proc. IPDPS 2005)*, Apr. 2005.

[17] National Aeronautics and Space Administration. NAS computing resources—Columbia supercomputer. http://www.nas.nasa.gov/Users/Documentation/Altix/hardware.html, 2004.

[18] A. Petitet, R. C. Whaley, J. Dongarra, and A. Cleary. HPL—a portable implementation of the high-performance Linpack benchmark for distributed-memory computers. http://www.netlib.org/benchmark/hpl/, Jan. 2004.

[19] TOP500 team. Top500 list for November 2004. http://www.top500.org/lists/2004/11/, Nov. 2004.

[20] Transmeta Corp. *TM5800 Data Book—Version 2.1*, Sept. 2003.

[21] Transmeta Corp. *Transmeta Efficeon TM8820 Processor—Product Sheet*, Jan. 2005.

[22] M. Warren, E. Weigle, and W. Feng. High-density computing: A 240-node Beowulf in one cubic meter. In *Proc. Supercomputing 2002*, Nov. 2002.