Hardware Design and Implementation of PC Cluster RWC Technical Report P-96-017

Atsushi Hori, Hiroshi Tezuka

Real World Computing Partnership
Tsukuba Research Center

Abstract

We have developed a PC cluster comprising 32 Intel Pentium processors connected by Myrinet, a giga-bit-per-second network. We built the PC cluster using off-the-shelf components and commercially available electronic components. These industrial standard processor board and its backplane make our PC cluster easy to upgrade and maintain. Unique chassis design makes it feasible to put 32 processors in one rack. Each PC in the cluster is connected by Myrinet, 160 MByte/s. high-speed network. We developed a Myrinet software driver, called PM which achiving 7.2 micro second latency and a bandwidth of 117 MByte/s. bandwidth. This technical report describes why we chose this design and how we built the PC cluster.

1 Introduction

The PC is a hardware commodity, so the idea of connecting PCs through a high-speed network is nothing new. It is a natural stage of enovation. The easiest way to build a PC cluster is merely to pile off-the-shelf PCs and connect them with an Ethernet. This method makes possible a cheap parallel machine composed of commodity hardware components.

Although this method is inexpensive, it has some diadvantages. First, (Fast-) Ethernet has lower a bandwidth than networks used in parallel machines. Second, pile of PCs takes up a lot of space. Also electro-magnetic theory dictates that the longer signals have to travel, the longer it takes. Third, most PCs are not designed for easy maintenance. For example, just to ad memory to each PC requires that the PCs be unstacked, the covers removed, the memory modules installed, then the PCs reassembled and restacked. Doing this on tens of PCs is quite a chore. Fourth, you must install software and configure each PC. Beyond that software upgrades or patches have to be to each PC. Finally, monitoring facilities are needed for bootup, shutdown, and fault detection.

To overcome those disadvantages, we designed and prototyped a PC cluster, rather than just a pile of PCs. We chose a PICMG (PCI Industrial Computer Manufacturers Group [5]) standard single-board-computer (SBC) and PICMG passive backplane. Using the PICMG stndard, we successfully build a PC cluster that is easy-to-maintain, easy-to-upgrade, and compact. We chose Myrinet[4] to interconnect the PCs. Myrinet has 160 MByte/s bandwidth per link. This band-

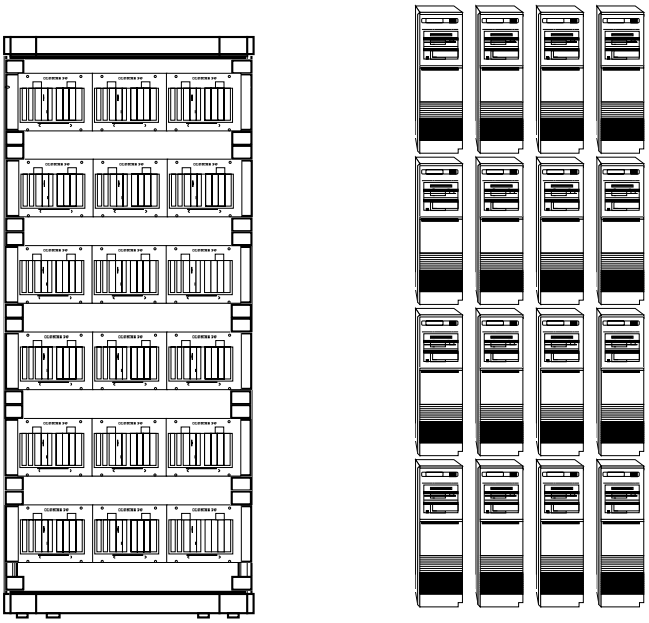

Figure 1: PC Cluster and Pile of PCs

width is ten times more than the bandwidth of Fast Ethernet. With this combination of PCs and Myrinet, our PC cluster approaches the performance of commercial parallel machines.

Our PC cluster offers nothing new in hardware. All the electronic components are commercially available. We developed the PC cluster to pursue parallel software research for parallel language, a runtime system, and a parallel operating system.

The question is how much performance can the PC cluster deliver. The PC cluster is obviously cheap. If performance of the PC cluster is comparable to that of commercial parallel machines, then the PC cluster will form a category of parallel machines. If performance does not reach that of parallel machines, then the PC cluster can form a low-end category. In either case, PC cluster will survive.

The following sections describe how we designed and developed our PC cluster. Since our PC cluster is a prototype, conservative tradeoffs were made.

2 Design Goal

We decided to run license-free UNIX on each PC, beacuse, its source code is available and there are no license fees. The open source code is very important, as the operating may have bugs that need to be fixed or may need modifications to tune performance for parallel computation or a parallel operating system. With the open source code, we can overcome these problems. We are developing a parallel operating system, called SCore-D[1], which runs on top of UNIX. SCore-D is written in MPC++[2, 3], a parallel C++ implementa-

tion. Using this parallel operating system and MPC++ runtime system development, we are able to find hardware or UNIX operating system bottlenecks in the development of parallel software.

We decided to use Myrinet, which is not a commodity hardware product. Network performace plays a very important role in parallel computing and network characteristics affect how we program for higher performance. We are familiar with Myrinet through our experience connecting 36 SparcStation 20s in a workstation cluster. We had already developed a Myrinet software driver, called PM[6]. Using Myrinet and PM, we achieved in user-to-user, onw-way latency, of 7.2 microseconds and a bandwidth of 117.6 MByte/s on PCs. Those latency and the bandwidth performance is far superior to that of conventional Ethernet.

We decided to avoid the development of new complex electronic components as much as possible. The PC world is changing so fast, taking one year to develop a PC cluster would put us behind, and the cost-effectiveness of the PC cluster would be lost. Using conventional hardware, we can use large software resources, which frees us to devote our energies to the software research itself. Each PC should be fully compatible with PC/AT standard.

Scalability is also important. A rack-mounted PC cluster should be easily expanded merely by connecting additional PC cluster(s). The PC cluster should contain a monitoring facility to diagnose PCs and should boot-up by itself, so that the PC cluster will behave as a complete parallel machine. With this configuration, we can compare the cost of the PC cluster with other parallel machines.

Table 1: Comparison of various network

Network	Bandwidth	Reliability	Message Order
FastEthernet	100 [Mbps]	unreliable	not preserved
ATM	155 [Mbps]	unreliable	not preserved
Myrinet	1280 [Mbps]	reliable	preserved

Finally, the hardware should be easy to maintain. Since our PC cluster will also be used as a test bench, we expected that some hardware components will need to be replaced. We should configure the clusters so that replacement is easier than that of pile of PCs.

3 Design

3.1 Networks

We decided to have three types of network, serial, Ethernet, and a high speed network in the PC cluster. In table 1, some available network interface cards for PCI are compared. We paid attentions not only the bandwidth, but also packet loss and messagin order. If a network hardware does not guarantee reliable transmission, or does not preserve messagin order, then the software should take care of those tasks. This means extra software overhead and resulting larger latency and lower bandwidth. We abandoned the idea of an extra network other than Myrinet, such as ATM or FiberChannel, because of the limited number of PCI slots on the passive backplane we chose. Figure 2 is a connection diagram of our PC cluster.

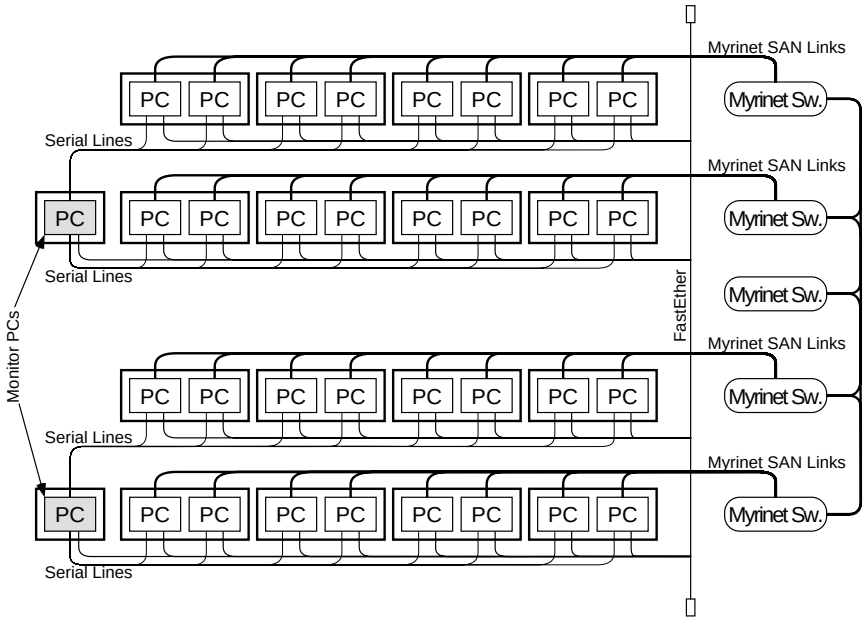

Figure 2: Three Networks in PC Cluster

Serial Lines

Most PCs have two serial ports, so we decided one of them as a console, and the other as a kernel debugger for kernel development. It can also be usedto monitor the PC's activity.

Ethernet

Since the research and development of a parallel file system is not a high priority for us, we will use NFS as a coherent and distributed file system over the cluster. We have also decided that the standpoint of version control, Ethernet is the best choice, as it is a mature system that is readily available.

Myrinet

Our experience in developing workstation clusters indicates that Myrinet is the best network for inter-processor communication. And with the PCI I/O bus, communication performance of the PC cluster is expected to outperform that of workstation cluster.

3.2 Processor Card

We chose the PICMG standard, as the PICMG PCI/ISA processor board is electronically identical to the PC/AT. The PICMG PCI/ISA processor board and I/O cards mount in parallel on the PICMG PCI/ISA passive backplane allowing a more comact PC to be built. This is preferable to the standard PC mother board, which is larger than the PICMG processor board, and whose I/O cards mount vertically on the mother board. We also looked at

single board computers which are much smaller than the PICMG processor card. Most, however, are neither fully compatible with the PC/AT, nor have they PCI I/O bus.

The other possible alternative is the CompactPCI, which is another standard from PICMG. It has some advantages over the PICMG PCI/ISA standard, which include CompactPCI smaller size and higher reliability. At this time, however, the CompactPCI is not fully matured, but may be a strong alternative for PC clusters within the next few years.

The other advantage of using the PICMG standard is that the availability from many manufacturers in the world. Many Pentium and PentiumPro boards are already avilable. DEC has Alpha boards, and some manufacturers offer dual Pentium and dual PentiumPro boards. With this availability, we are able to choose a processor boards that meet any specific requirements.

The PICMG standard also has some disadvantages, i) the processor card is more expensive, and, ii) technology is usually two or three months behind off-the-shelf PC boards. Nevertheless, we chose the PICMG standard.

The pitfall we must watch for in choosing a processor card relates to speed. in choosing a processor card. Since Myrinet is the fastest PCI device among others, the chipsets used in some processor boards cannot handle the high speed data transfer. Therefore, it is important to test a small PC cluster setbefore ordering a large number of processor boards.

3.3 Chassis Design

Production cycles of processor boards, Myrinet and other peripheral cards are fast, so we cannot expect that the same board or card to be available a year from now. Also, because our PC cluster is used for research, the hardware configuration will change more often than a system in normal use. Maintenance must, therefore, must easily.

To make access easier, we designed a steel box, called a module, which contains two passive backplanes, each of which can hold one processor board, two ISA slots and two PCI slots. The module can also hold the +5V, +12V and - 12V power supplys, and a cooling fan (Figure 3). This configuration puts two PCs in a module, and each can operate and be tested as standalone units. Although it would have been possible to use a central power supply system to feed all PCs. Power cables thick enough to conduct hundreds of Amperes would have been. Such thick cables could have made assemble difficult.

The PCI and ISA cards are mounted with their metalic panels facing the front of the modules. This is reveerse of normal mouting. We mounted them the way, so that the LEDs that many of cards contain can be seen from the front of the rack.

The rack for the PC cluster holds 18 modules, 3 modules to a row, 6 rows to the rack. Various cables (Myrinet, Ethernet, serial and power) are harnessed on the rack frames . This moduler design makes the PC cluster easy-to-maintain. When a PC fails operation the cluster can be restored by merely replacing the module. Removing a modules is as simple as disconnecting the cables and pulling

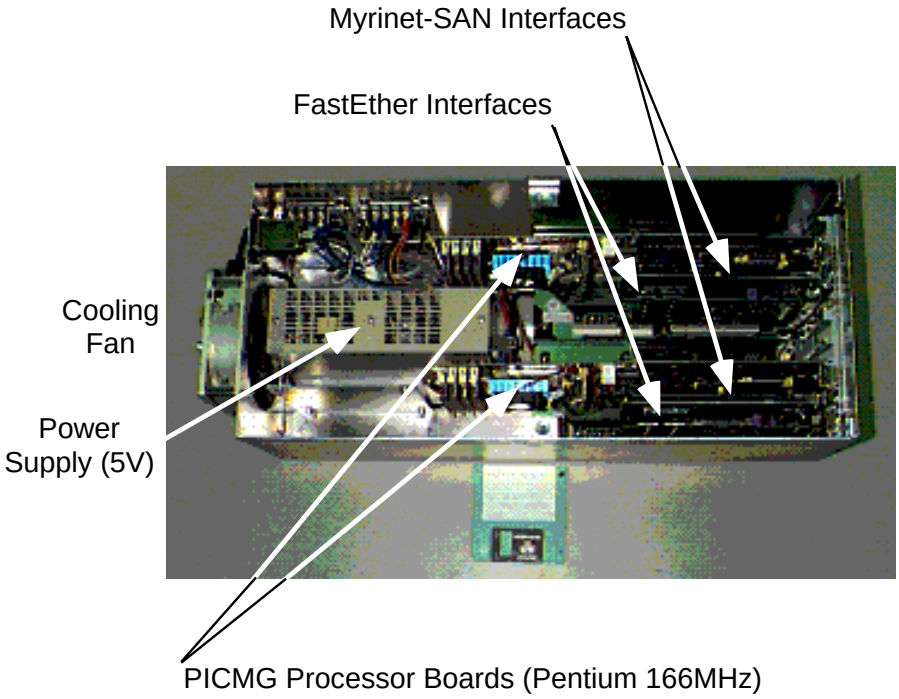

Figure 3: PC Cluster Module

out the module.

3.4 Monitoring Facility

Two of the 18 modules differ somewhat from the others. These two modules each contain just one PC, plus a disk and four serial line ISA cards to connect up to 36 serial lines. We call them monitor PC, because they serve as consoles, boot-up servers and network file servers. Thus, a rack contains32 PCs for computing and two monitor PCs.

3.5 Cooling System

We designed a steel box to hold three modules. The height of the outer box is 4U (7 inches). A 2U (3.5 inches) space is left between any two boxes. There is a number of open slots at the ceiling panel of the box. Since the top of module box is open, a cooling fan on the back pulls incoming air from the slots (Figure 4). This cooling system works fine. The frontal side of the space is used for air intake and cabling, and the back side allows space for a Myrinet switch and a Fast Ethernet hub. Placing Myrinet switches at each gap of rows shortens the length of Myrinet cables.

3.6 Diskless System

Whether to put a disk in each PC was a big issue. The Myrinet latency is a hundred times shorter than disk access time, and the bandwidth is far more than that of a disk. Further, disk reliability does not approach that of the other electronic components. Putting a disk in every PC means that the MTBF deteriorates as the number

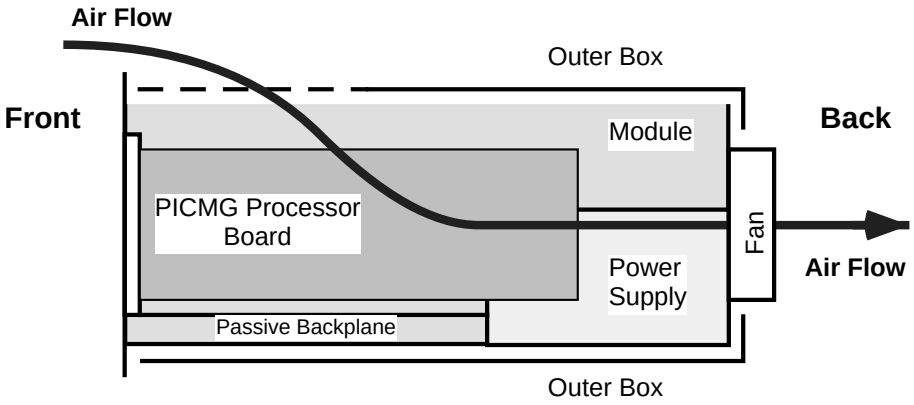

Figure 4: Cooling Air Flow (Side View)

of PC. A disk typically consumes about 10 Watts of power. The surge current that occurs when the disk rotor spins up is significant, so power on sequence might be required.

Further, a disk in every PC dictate a proper power-down sequence. Power failure would also be a problem. Shutting down tens of PCs is no fun, and an operater would have to wait for all PCs to shutdown. Extra hardware would be needed to synchronize shutdown. Because of these problems and the superior Myrinet performance, we decided not to incorporate disks, and to connect to external I/O system through the Myrinet.

To boot up all the PCs, the code of operating system kernel is copied through a network. Network booting via Ethernet is conventional technology, but attempting to boot tens of PCssimultaneously would overload the boot server. To avoid overload, we developed Broadcast Transfer Protocol (BTP). With BTP, the boot server

sends only one copy of the operating system kernel. As the Ethernet offers no-collision operationg in theory, boot time is expected to be short. The BTP code is baked into ROM on the Ethernet card. Booting through Although booting through Myrinet is possible, we took the conservative approach.

4 Summary

It took only three months from design of the PC cluster to assembly. This fast implementation is very important to keeping up with state-of-the-art technologies. The modular design of the PC cluster makes it compact and easy to maintain. When a faster processor board or network card is available, we can easily upgrade the PC cluster by replacing the cards.

Our evaluation of the PC cluster revealed that developing a passive backplane with only 2 PCI slots and a processor slot would allow 64 PCs to be mounted in the same size rack. That means the processor density of this cluster of 64 PCs would be higher than that of CM-5.

Myrinet, as opposed to Ethernet, delivers performance that is expected to compare favorably with commercial parallel machines. We are now installing our SCore operating system, MPC++ runtime system, parallel debugger and MPI message passing library. When these are installed, we will start evaluating performance of the PC cluster.

References

[1] A. Hori, H. Tezuka, Y. Ishikawa, N. Soda, H. Konaka, and M. Maeda. Implementation of Gang-Scheduling on Workstation Cluster. In D. G. Feitelson and L. Rudolph, editors, *IPPS'96 Workshop on Job Scheduling Strategies for Parallel Processing*, volume 1162 of *Lecture Notes in Computer Science*, pages 76–83. Springer-Verlag, April 1996.

[2] Yutaka Ishikawa. The MPC++ Programming Language V1.0 Specification with Commentary Document Version 0.1. Technical Report TR–94014, RWC, June 1994.

[3] Yutaka Ishikawa, Atsushi Hori, Hiroshi Tezuka, Motohiko Matsuda, Hiroki Konaka, Munenori Maeda, Takashi Tomokiyo, and Jörg Nolte. MPC++. In Gregory V. Wilson and Paul Lu, editors, *Parallel Programming Using C++*, pages 429–464. MIT Press, 1996.

[4] http://www.myri.com/.

[5] http://www.picmg.com/.

[6] Hiroshi Tezuka, Atsushi Hori, and Yutaka Ishikawa. PM: A High-Performance Communicatin Library for Multi-user Parallel Environments. Technical Report TR–96015, RWC, November 1996.

Table 2: Specification of PC Cluster

Node Processor			32
	CPU	Pentium 166 MHz	1
	Cache	Asynchronous	512 KB
	Memory		64 MB
	Ethernet	100 bps	1
	Myrinet	1280 bps	1
Monitor Processor			2
	CPU	Pentium 166 MHz	1
	Cache	Asynchronous	512 KB
	Memory		64 MB
	SCSI Disk	2 GB	1
	Ethernet	100 bps	2
	VGA		1
	Serial Port	RS232C	32

Table 3: Components and vender List

	Vender	Type	Note
Rack	Schroff	Comrack	80x80x200 [cm]
PICMG CPU Board	Advantech	PCA-6157	
FastEther	3Com	3c595-TX	
Myrinet NI	Myricom	M2M-PCI32	
Myrinet Switch	Myricom	M2M-DUAL-SW8	
		M2FM-SW8	
Serial Card	Comtrol	RocketPort RP8-J	