
THE AZUSA 16-WAY ITANIUM SERVER

AZUSA IS A PROTOTYPE 16-WAY, OPTIMIZED, ITANIUM-PROCESSOR-BASED SERVER SUITED FOR ENTERPRISE-LEVEL COMPUTING. THE AUTHORS DESCRIBE THE SERVER'S ARCHITECTURE AND FEATURES, WHICH ARE AIMED AT PERFORMANCE SCALABILITY FOR DIVERSE WORKLOADS.

..... NEC's Itanium prototype server (see Figure 1), code-named AzusA after a river flowing through mountains west of the company's design labs—boots IA-64 operating systems and uses a custom-designed chip set. The chip set supports up to 16 Itanium processors for optimum 16-way performance. In combination with IA-64 Itanium processors, AzusA provides a powerful platform solution for the backbone of the Internet and enterprise computing.

The chip set has very low latencies for a 16-CPU system and has well-balanced memory and I/O bandwidths. As a result, we expect this 16-way server to exhibit consistently high performance in a wide spectrum of workloads ranging from e-commerce and data mining to scientific computing.

The server can be partitioned into a maximum of four domains, each constituting an isolated and complete computer system. This feature aids consolidation of smaller servers into a fewer number of larger servers.

AzusA has an integrated service processor for platform management including preboot configuration, platform error handling, domain management, and other features. The combination of the chip set's high-reliability design and the service processor provides capabilities required for enterprise-level computing.

AzusA 16-way architecture

Figure 2 shows the block diagram of the AzusA 16-way configuration. The modular construction is composed of four 4-CPU cells interconnected via a data crossbar chip and address snoop network. The 16-way box can be hard-partitioned into a maximum of four domains by fully or partially disconnecting the crossbar and the address interconnect at natural boundaries.

Figure 3 shows a prototype cell. Each cell has

Fumio Aono
NEC
Masayuki Kimura
NEC Kofu



Figure 1. The AzusA prototype system.

one system bus that supports up to four Intel Itanium microprocessors with power pods, the AzusaA chip set's north bridge, main memory DIMMs, and four connections to peripheral component interconnect (PCI) adapters via proprietary Gigastream Links (GSLs). Figure 4 (next page) shows the interrelations among those components. Two of the four microprocessors and their associated power pods are located on each side of the cell.

AzusaA's distributed, shared-memory¹ architecture provides each of the four cells with a portion of the main memory. Each cell has 32 DIMM sites, half of which are located on an optional memory daughterboard. The chip set supports up to 128 Gbytes of physical address space.

As is the nature of a distributed, shared-memory machine, AzusaA's memory has a cache-coherent, nonuniform memory access (NUMA)² model. However, its latency characteristics are very close to those of a symmetric multiprocessing machine, with a remote-to-local memory access latencies ratio of approximately 1.5. Apart from latencies, AzusaA provides a flat 16-way model to software.

The four cells share the service processor and the base I/O including the legacy south bridge, also shown in Figure 2. When the server is partitioned into two or more domains, additional PCI, add-on base I/O cards are inserted into designated PCI slots, except for the primary domain, which is serviced by the original base I/O attached to the service processor board. The shared service processor serves all domains simultaneously.

Each PCI adapter has two 64-bit PCI buses that are configurable as either two-slot 66-MHz buses or four-slot 33-MHz buses, as shown in Figure 5 (next page). All of the PCI slots are hot pluggable. Each PCI adapter has two GSL ports; both ports may be used concurrently for performance, or alternatively for redundancy. The maximum length of the GSL copper cable is 5 meters (16.4 feet) with a half Gbyte/s of I/O bandwidth. The maxi-

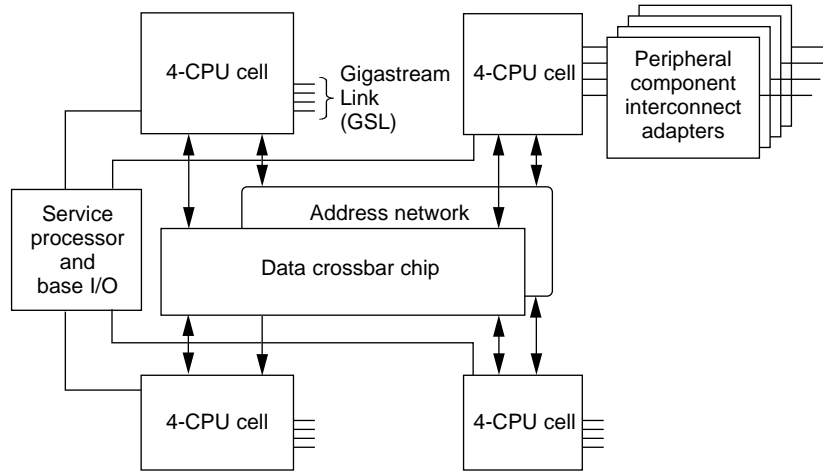


Figure 2. AzusaA system block diagram.

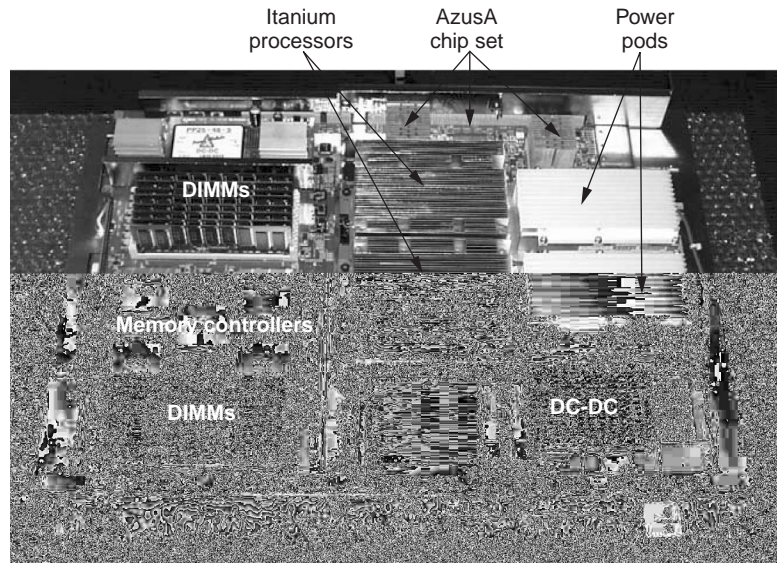


Figure 3. AzusaA cell card photo.

imum configuration of an AzusaA system is 128 PCI slots or 32 PCI buses, with a maximum of 16 PCI adapters. The resulting aggregate I/O bandwidth is approximately 8 Gbytes/s.

Chip set architecture

Figure 4 shows the chip set components and their interconnections for each cell. The 16-way configuration has four sets of components plus the external data crossbar. The chip set design is optimized for 16-way or 4-cell configurations and employs a snoop-based coherency mechanism for lower snoop latencies. The chip set uses 0.25-micron process

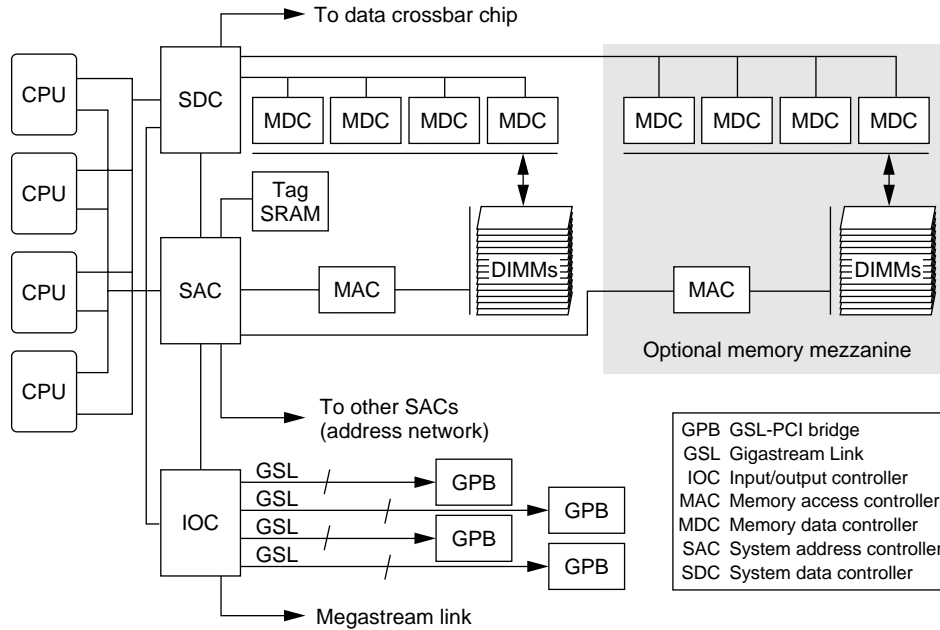


Figure 4. AzusaA chip set components.

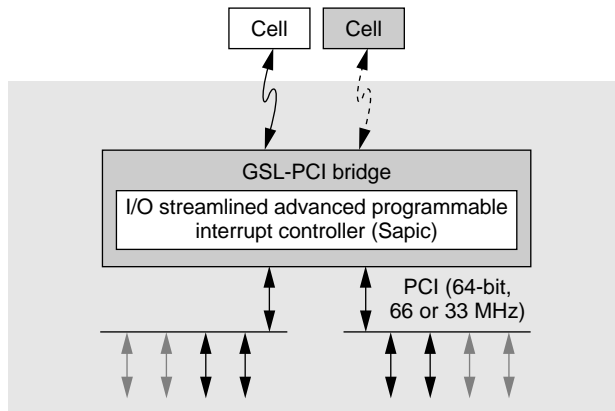


Figure 5. GSL-PCI bridge and PCI buses.

technology and operates at a multiple of the system bus clock frequency.

At the heart of the chip set is the system address controller, an ASIC that handles system bus, I/O, and intercell transactions; internal and external coherency control; address routing; and so on. Figure 6 is a high-level block diagram of the system address controller. The system address controller controls the system data controller and transfers data to and from the system bus, main memory, I/O, and other cells. The I/O controller has signal connections to both the system address controller and system data controller. The I/O

controller also has four GSLs to the I/Os as well as a Megastream Link to the legacy south bridge and service processor. I/O translation look-aside buffers are integrated in the I/O controller chip and convert a 32-bit address issued by a single-address-cycle PCI device into a full 64-bit address.

There are two memory chip sets with one located on the cell and one on the optional memory daughter-board. Each consists of an intelligent memory address controller and four interleaving memory data controllers. It supports a chip-kill feature as well as a memory scan engine that performs memory initialization and test at

power-on and periodic memory patrol and scrubbing.

GSLs connect the I/O controller to host bus PCI bridges, as shown in Figure 5. The same GSL-PCI bridge chip can support both 66-MHz and 33-MHz 64-bit PCI buses. NEC is also planning a PCI-X version of the GSL-PCI bridge. For better I/O performance, the GSL-PCI bridge has write-combining buffers to combine subline inbound stores. The bridge chip also has a data prefetch mechanism to maximize DMA read performance. In addition, the GSL-PCI bridge can handle systemwide, peer-to-peer transactions as well as legacy sideband signals. The GSL-PCI bridge integrates an on-chip programmable interrupt controller, or I/O Sapic (streamlined advanced programmable interrupt controller), that can support various interrupt modes from 8259 compatibles to Sapic.

Cells are interconnected tightly and directly for addresses to form the address network, and via the data crossbar for data. In a two-cell configuration, the data crossbar chip component may be omitted by direct wiring between the two cells.

To effectively reduce the snoop traffic forwarded to the system bus, each cell has a snoop filter (tag SRAM) that keeps track of the cache contents in the four CPUs on the

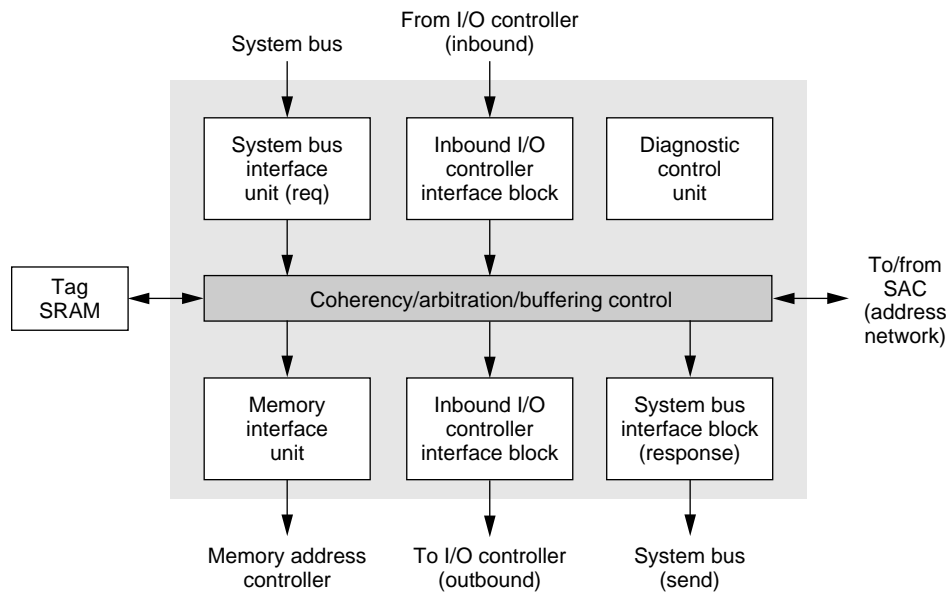


Figure 6. The system address controller.

cell. When a coherent memory transaction is issued in one cell, its address is broadcast to all other cells for simultaneous snooping. The snoop filter is checked for any chance that the inquired address is cached in the cell. If it is a possibility, the address is forwarded to the system bus for snooping, and the result is returned to the requester cell. Otherwise, a snoop miss response is returned instantly as a result of the tag lookup. In either case, the snoop filter is updated by replacing or purging the tag entry associated with the CPU cache line that was loaded with the memory data. On a memory read, the local or remote addressed memory line is always read speculatively, whether or not the line may be cached in a CPU.

To fully accommodate Itanium processor system bus bandwidth as well as the I/O traffic, the chip set provides approximately 4.2 Gbytes/s of memory bandwidth per cell (a system total of 16.8 Gbytes/s). The interconnection between the system data controller and the data crossbar chip is also 4.2 Gbytes/s per port (8.4 Gbytes/s bisectional), so the full memory bandwidth is exportable to the other cells. The address snoop bandwidth matches the theoretical maximum transaction issue rate at all four system buses for the best performance.

The system address controller has numerous address range registers to configure, which

present a single flat memory space to the operating system. Similarly, all the PCI buses can be configured—either by the firmware at boot time or dynamically during online reconfiguration—to make a single, systemwide PCI bus tree. For compatibility reasons, these configuration registers are mapped to the chip set configuration space in a manner similar to Intel’s 82460GX chip set. This makes AzusA a natural 16-way extension of an 82460GX-based 4-way system.

From an operating system’s viewpoint, our 16-way platform appears as a collection of 16 CPUs on a single virtual system bus, which is also connected to a single large memory and a large PCI bus tree rooted at the system bus. Although there are certain practical compromises such as limiting external task priority register (XTPR)-based interrupt rerouting within each physical system bus, AzusA’s simple system image and near-uniform memory access latency make it easy to achieve very good scalability without elaborate programming.

The chip set architecture supports compatibility with the Itanium processor and features aimed at reliability, availability, and serviceability. These features include cell hot-plug capability and memory mirroring, data paths protected by error-correcting codes, error containment and graceful error propagation for IA-64 machine check abort recov-

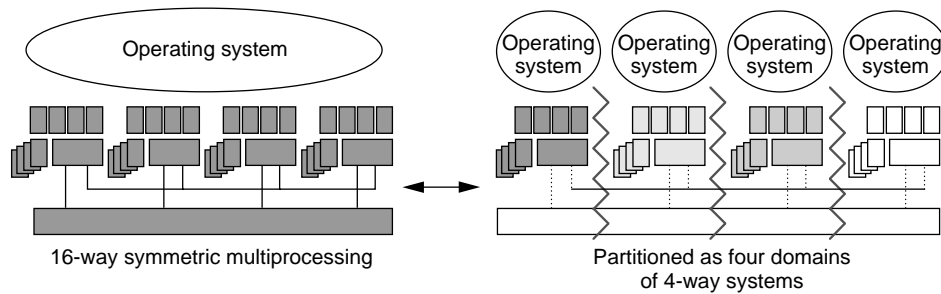


Figure 7. System partitioning example.

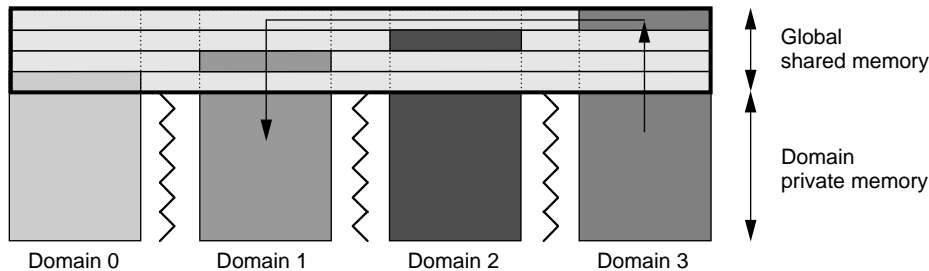


Figure 8. Interdomain communication via shared memory.

ery, parity-protected control logic, hardware consistency checking, detailed log registers, and backdoor access to the chip set.

Performance

While supporting managability and flexibility, the NEC team also designed the AzusaA chip set with performance features in mind. Its memory latency in a 16-way cache-coherent, NUMA configuration is less than 200 ns for a local memory access or local CPU cache hit, and less than 300 ns for a remote (other cell) memory access or remote CPU cache hit, for coherent accesses measured from the address to the data on the system bus. I/O latency is less than 800 ns.

In real-world environments like online transaction processing in which data is extensively shared among the processors, our study shows that more than half of cache misses might be found in other processors' caches. In such cases, including an additional level of caching to AzusaA would not justify the added cost.

For this reason, the AzusaA design team focused on improving the effective memory latency without resorting to system-level caches. As a result, AzusaA has achieved a very low remote-to-local ratio of approximately 1.5. Various techniques including specula-

tively reading the main memory and the high-speed snoop network helped to reach this ratio. Because of this low ratio, the design team expects AzusaA to demonstrate consistent performance scalability for all types of workloads.

Partitioning and in-box clustering

When AzusaA is hard-partitioned at cell boundaries creating isolated domains (four maximum), each domain constitutes a complete computer system and runs a separate operating system instance (see Figure 7). Each domain can contain an arbitrary number of cells, and each cell may have any number of CPUs.

The integrated service processor configures the domains by programming the chip set registers. Repartitioning may take place either dynamically at user requests, at failures, or at boot time. Needless to say, each domain can be separately booted or brought down without affecting operations of other domains. Although, to alter domain configuration when an operating system is running requires operating system support.

In addition to domain partitioning, the AzusaA chip set supports a true in-box clustering capability; that is, the partitioned domains can communicate with one another through the crossbar, eliminating the need for external interconnects.³

The internal interconnect is actually made up of partially shared physical memory, custom drivers, and optional user-mode libraries. Each participating domain contributes a fraction of its physical memory at boot time to share among the nodes as a communication area. Figure 8 shows the conceptual view of the in-box cluster memory in a 4×4 configuration. Users can program the amount of shared memory, as well as node configurations in the field.

Custom drivers and user-mode libraries provide standard cluster application program interfaces (such as VIPL or Winsock direct)

to the upper layers and use the partially shared memory as the physical medium. This offers main memory latency and bandwidth for cluster node communications while ensuring application compatibility.

For security, the shared part of memory is not available to the operating system so that the main memory space is completely isolated between nodes. Also, the shared memory permits transferred data to always be held in the memory on the sender's physical cell(s); thus, the loss of a node does not affect communication among surviving nodes. The receiver node is gracefully protected from uncorrectable memory data errors propagated from the sender cell.

Features

As mentioned earlier, AzusA has many platform features that support reliability, availability, and serviceability. These include redundant GSL connections, fault-tolerant copy tags, memory patrol, and scrubbing features.

Reliability and data integrity

The chip set protects virtually all logic circuits and paths, including control logic—unlike most existing chip sets that protect only data paths with parity or error-correcting codes. Consider the example of a corrupt data buffer pointer. If the buffer pointer's intermittent failures were left undetected, the wrong data would be sent to the processor, I/O, or memory without flagging errors—breaking the data integrity. In the AzusA chip set, all buffer pointers and most of the other control structures are protected by parity or another scheme, virtually eliminating the risk of silent data corruption.

Availability

As in PCI cards, a cell in a partitioned configuration can be hot swapped while other domains are online and in service. With proper operating system support, a cell in a multi-cell domain (such as a 16-way symmetric multiprocessing machine) can be dynamically hot swapped and serviced for continuous availability. Considering that the main memory is generally harder for operating systems to relinquish, AzusA has a memory-mirroring feature to enable dynamic copying of the main memory to a backup cell. So a cell can be deconfigured only by disabling (performed by

.....
The chip set protects virtually all logic circuits and paths, including control logic—unlike most existing chip sets that protect only data paths with parity or error-correcting codes.
.....

the firmware either dynamically or at reboot) its CPUs and I/Os.

Another availability feature is platform error containment. When possible, the chip set tries to reduce an error to a single transaction and propagate it all the way to the destination. In combination with IA-64's machine check abort (MCA), this can improve the chance that the operating system will survive errors such as irrecoverable error-correcting-code errors.

AzusA also supports fault-resilient boots: on a system crash, the service processor analyzes the chip set log to pinpoint the not working field-replaceable unit using the built-in diagnosis dictionary. The service processor can then reboot the system without the suspected field-replaceable unit to prevent the error recurrence. The redundant I/O paths enhance the chances of reconfiguration and contribute to higher availability. Other availability features include redundant and hot-swappable fans and power supplies.

Serviceability and maintainability

The integrated service processor plays an important role in platform management, error handling, and platform maintenance.

The service processor monitors and manipulates the platform via the chip set's backdoor maintenance port, and associated sideband signals. Through the back door, it logs and analyzes chip set errors then reconfigures and reboots in coordination with machine check abort processes whenever possible. It also configures the domain, reconfigures dynamically, and monitors events.

The service processor includes a dedicated service LAN port to provide a secure connec-

tion to a management workstation. The service processor console can be accessed either through the secure connection or via a local serial port.

Other service processor features include automated remote error reporting, and off-line diagnoses of deconfigured cells. The service processor also serves as the Advanced Configuration and Power Interface (ACPI)⁴ specification embedded controller or the Intelligent Power Management Interface (IPMI)⁵ specification baseboard management controller. Despite all this functionality, a stall in the service processor will not affect normal system operation.

Packaging

The main cabinet houses 16 PCI slots and array disks, 16 processors, the memory, service processor, and the server system infrastructure (SSI)⁶ power supply. Additional PCI slots require the addition of an I/O expansion cabinet. The maximum GSL cable length of 5 meters allows for a maximum of four expansion cabinets, up to two on each side of the main cabinet.

Conformance to standards

AzusA design conforms to many industry standards including DIG64,⁷ IPMI, and SSI—which are jointly promoted by NEC—as well as ACPI and PCI, to name just a few.

AzusA will be commercially available shortly after the Itanium processor is released. At the time of its introduction, AzusA will have almost all of the features discussed here. Together with the advantages of the IA-64 architecture and the vast amount of address space it can offer, we expect that AzusA will be widely deployed—in combination with other IA-64 and IA-32 servers—to support the back end of the Internet. We also plan to further enhance the AzusA architecture's features. MICRO

Acknowledgments

Intel has extensively supported AzusA development. Microsoft ported its SQL Server and the TerraServer application to an AzusA prototype. Phoenix Technologies provided the foundation of the AzusA system abstraction layer (SAL).

We thank our many colleagues at NEC, NEC Kofu, NEC Engineering, NEC Soft-

ware Hokuriku, NEC Software Kobe, NEC Software Hokkaido, NEC Software Kyusyu, and NEC Systems (US) for their participation in the design and development of AzusA and in the review of this article.

Special thanks go to Naomichi Yonezawa, who contributed a great deal to the AzusA architecture but unexpectedly passed away.

References

1. J. Proti, M. Tomasevi, and V. Milutinovi, "Distributed Shared Memory: Concepts and Systems," *IEEE Parallel and Distributed Technology: Systems and Applications*, Vol. 4, No. 2, 1996, pp. 63-79.
2. D. Johnson and B. Taylor, "Designing Complex, Interoperable IA-64 Servers," presentation at the Intel Developer Forum Fall 2000, 22 August 2000.
3. Gregory F. Pfister, *In Search of Clusters*, 2nd ed., Prentice Hall PTR, New Jersey, 1998.
4. *Advanced Configuration and Power Interface*; <http://www.teleport.com/~acpi>.
5. *Intelligent Platform Management Interface*; <http://developer.intel.com/design/servers/ipmi/index.htm>.
6. *Server System Infrastructure*; <http://www.ssiforum.org/default.asp>.
7. *The Developer's Interface Guide for IA-64 Servers*; <http://www.dig64.org>.

Fumio Aono is an engineering manager and architect of AzusA system design in the 2nd Engineering Department, Computers Division of NEC. He represented NEC as a promoter at the DIG64 V1.0 working group. Aono received his MSCS from the University of California, Santa Barbara and a BE degree from the University of Tokyo. He is a member of the IEEE Computer Society and of the ACM.

Masayuki Kimura is a principal engineer and chief architect of the AzusA chip set at the 1st Computer Engineering Department of NEC Kofu. Kimura received a ME degree and a BE degree from Yokohama National University. He is a member of the IPSJ.

Direct comments and questions about this article to Fumio Aono, NEC, M/S23-34140, 1-10 Nisshin-cho, Fuchu, 183-8501, Japan; f-aono@bq.jp.nec.com.