UltraSPARC™ Ultra Port Architecture (UPA): The New-Media System Architecture

The latest family of microprocessors from Sun Microsystems, the UltraSPARC™, delivers leading-edge performance to meet the exacting demands of networking, the rise of multimedia on the desktop, and high-bandwidth applications such as real-time video conferencing and other workgroup applications. Although UltraSPARC in and of itself produces outstanding results—a 167 MHz device yields 350 SPECfp92 and 240 SPECint92 with a sustained block load and store rate of 600 Mbytes/sec—in real-world applications it does not perform in a vacuum. Instead, it must interact within a system environment, relying on a system architecture to support its exceptional throughput and capabilities.

Along with promoting processor performance, system design for UltraSPARC demands a flexible architecture that can easily scale to meet a range of requirements—from desktop systems to large servers that incorporate multiple multiprocessing (MMP). Moreover, to meet competitive market demands, this system architecture must be a cost-effective solution and be easy to implement and support.

Leveraging the system design expertise of Sun Microsystems, Inc., the Ultra Port Architecture (UPA) is an advanced hardware architecture that complements the UltraSPARC microprocessor’s performance. It is fully compliant with the Net.Core system architecture specification standard. The UPA hardware architecture centralizes critical functionality, reducing memory and shared data latencies to an absolute minimum. UPA delivers outstanding throughput (a peak rate of 1.3 GBytes/second), while its scalable structure makes it a
viable architecture for a broad range of applications. To lower system design cost, the UPA interconnect architecture is tightly coupled with mainstream, volume technologies. It also integrates important capabilities, further reducing system design complexity and cost.

**WHAT IS UPA?**

UPA is the processor interface definition to the interconnect describing the logical and physical specification of the port interface and the requirements imposed on the interconnect. It also delineates the behavior of the system controller and the I/O interface to the system interconnect. Designed for processors like UltraSPARC I that adhere to the V9 processor architecture, UPA incorporates system design considerations for efficient low latency interface specifications that will support a complete generation of desktop and server systems.

UPA supports a wide range of ports (32, 64, 128, etc.) and includes four interface properties. An optional *master* interface issues read/write physical address transactions to the interconnect using a distributed arbitration protocol for driving the address bus. A UPA master may also contain a physically addressed coherent cache that has no minimum or maximum size requirement. The mandatory *slave* interface receives read/write transactions from UPA masters, maintaining strong ordering for transactions from the same master class as well as transactions to the same device address. A UPA port may be slave-only, for example, for a graphics frame buffer. The two other optional interfaces a UPA port can assume is an *interrupter* and an *interrupt handler*. UPA interrupters generate interrupt packets to UPA interrupt handlers.

Unlike conventional, global broadcast-based multiprocessor systems that are cache coherent and globally share a snoopy address bus, the UPA interconnection architecture relies on point-to-point packet switched messages. To maintain cache coherency, it issues these messages from a centralized system controller for desktop workstations with one to four processors or a distributed system controller for larger servers. The UPA can maintain a duplicate set of all the cache tags in the system and performs duplicate tag lookup and main memory initiation in parallel pipelines for each coherent transaction. This is substantially different from directory-based multiprocessor systems that maintain coherency states for each data block in main memory and incur a read-modify-write operation penalty for each read transaction that reaches main memory.
LOW LATENCIES YIELD MAXIMUM THROUGHPUT

The departure from conventional snoopy-bus based and directory-based multiprocessor systems results in significant improvements for minimizing latencies. Lower latencies directly translate to minimal cycles per instruction for the processor by reducing the cache miss penalty to as close to memory cycle time as practical. The result is an interconnect architecture that fully exploits the outstanding throughput of UltraSPARC. A system incorporating the UltraSPARC and UPA will deliver a sustained bandwidth of 1 Gbytes/sec-twice that of leading competitive architectures announced this year-at one-half or one-third the clock rate of the processor.

Specific structures within UPA promote minimal latencies. For example, UPA supports UltraSPARC’s separate data and address bus architecture. These large buses—the address bus is 64 bits wide to directly support the 64-bit, V9 architecture, while the data bus is 144 bits wide (128 bits for data and 16 bits for error checking)—allowing for peak, achievable bandwidth. With two buses, dead cycles are not created when switching a shared bus between data and address. There are no turnaround penalties on switching penalties between address and data drivers. And the threat of bus contentions caused by competing address and data transactions disappears.

Not only does UPA support distinct address and data buses, but it can accommodate multiple point-to-point buses. Typically there are multiple interfaces to handle I/O, graphics and the processor in most systems. In multiprocessing systems, further interfaces are needed to the multiple CPUs. Instead of having one set of data and address buses for all these interfaces, UPA allows for an unlimited number of buses.

The benefits of multiple bus sets are numerous. Having multiple sets of buses minimizes the number arbitration cycles and lowers the possibility of bus contentions. The system controller understands the responsibilities of the different buses and can process bus requests in parallel for the multiple buses. It also maximizes the use of bus parking algorithms, lowering the latencies associated with gaining ownership of the bus. In essence, multiple address and data buses means less potential masters on each set of buses.

A distributed arbitration protocol that is pipelined is used to ensure the lowest possible latency for bus ownership, while at the same time meeting the minimum cycle time requirements
of UPA. Each UPA port has arbitration integrated into it so every port in the system sees the bus request for each other. This saves at least one cycle as opposed to two or three cycles in competitive systems. Although one cycle may seem a modest saving, when the system tolerance is only ten cycles, this kind of latency reduction can make a big difference in system performance.

ADAPTABLE ARCHITECTURE FOR A RANGE OF SYSTEMS

Along with its outstanding performance, UPA is an extremely flexible architecture, designed to meet the various cost/performance needs of system design. For example, the primary focus when designing a desktop system will be cost. At the other end of the spectrum, the number one consideration for a large server design is performance. Although UPA is designed to deliver optimal cost/performance for one to four processors, it supports a range of system design.

UPA can work with almost any system configuration—from uniprocessor to many, many processor (16, 32, 64, 128, etc.) multiprocessing server—but is specifically optimized for 1- to 4-way multiprocessing. Up to four tightly coupled processors and a system controller can share the same system address bus. The distributed arbitration arrangement eliminates the need for lengthy chip crossings dedicated exclusively to determining bus access. Multiprocessing systems with numerous processors can also be built around UltraSPARC by taking advantage of the rich set of transactions and the coherency protocol supported by the processor memory interface unit. Coherency protocol is point-to-point-write-invalidate using duplicate tags for snooping. The cache state transitions supported in the UltraSPARC processor include MOESI, MOSI and MSI.

Regardless of how many processors are used in the system, the UPA bus structure can also support a variety of configurations. As discussed earlier, the number of separate point-to-point address and data buses that can be used to promote system performance is only limited by the pins on the system controller. But when system costs are paramount, a muxed or shared bus can be used. A near-zero latency arbitration scheme for shared buses ensures optimal bus utilization even when data and address transactions flow over the same bus. UPA allows for up to four ports to share a single address bus to support four-way multiprocessing.
Not only can there be multiple buses-muxed or separate—but the width of the data bus can vary to meet different system price/performance goals. As data flows through a system, the data rate will change depending on what part of the system is being accessed. Data rates will be slower when interacting with commercial memories as oppose to interfacing with UltraSPARC.

With flexible data bus widths, the system designer can target the width of every data bus to match the end port needs. For example, an I/O system might only need 64 bits of data, but the processor would require 128 bits for second level cache, helping it to look like first level cache. Or the data bus to external memory might be wider for higher throughput to slower, more cost-effective memories.

**COST-CONSCIOUS DESIGN**

One of Sun’s goals when designing UltraSPARC and UPA was to create a cost effective system without sacrificing performance. To reach that goal, as much capability as possible was integrated into UltraSPARC and UPA. And compatibility with mainstream technologies was given priority so that system designers could leverage the cost advantages of these technologies.

UltraSPARC integrates crucial capabilities on chip. A complete New-Media instruction set in the UltraSPARC supports the range of encode and decode standards including broadcast-quality MPEG2 decompression. These capabilities eliminate the need for expensive graphics processing, image compression and double buffering circuitry. UltraSPARC also includes full second-level cache support on-chip. Instead of fussing with glue logic, the system designer directly attaches commodity SRAMs to the processor to create the second-level cache.

In the same way, UPA integrates key functionality, further reducing system design complexity and support. Cache coherence protocol processing and other related complexity is off loaded to the system, considerably simplifying the processor system interface. The complexity tradeoff was made to put the burden of cost and complexity of duplicate tag management in the system instead of the processor. Likewise, important error correction circuitry is included with each UPA port, replacing external circuitry. And the universal nature of the UPA port eliminates the need to deal with different types of ports for I/O, graphics and the CPU. Instead, the designer
uses an UPA port for each, simplifying system design and operation.

To further minimize system cost and design complexity, UPA was structured to interact with high-volume, low-cost technologies. For example, any commercially available LVTTL drivers are compatible with UltraSPARC’s 3.3V power supply because UPA is a globally synchronous architecture. Since UPA and UltraSPARC do not require fixed latencies, they can interact with any proprietary interface, eliminating the need to insert additional FIFOs to ensure proper handshaking. And UPA’s ability to support 256-bit data buses enables it to interface to vanilla DRAMs instead of expensive SDRAMs or EDODRAMs.

ARCHITECTED FOR THE ’90S

UPA reflects the depth of systems expertise and knowledge at Sun Microsystems. Its advanced design, exceptional performance and emphasis on cost-effective design make it the interconnect solution of choice for next-generation designs. UPA in conjunction with UltraSPARC delivers the high throughput and performance needed for today’s and tomorrow’s demanding, networked compute environments.