

An Overview of the UltraSPARC[®] III Cu Processor

The UltraSPARC III Processor Moves to Copper Technology

Version 1.1

June 2002

A White Paper

This document provides a brief overview of the important new technology in the UltraSPARC III processor generation of SPARC[®] processors, specifically as it serves to maintain compatibility, increase scalability, accelerate real performance, and improve RAS (Reliability, Availability, Serviceability) in SPARC-based systems designed for enterprise network computing.



901 San Antonio Road
Palo Alto, CA 94303-4900 USA
800-681-8845

Copyright © 2002 Sun Microsystems, Inc. All Rights reserved.

THE INFORMATION CONTAINED IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY EXPRESS REPRESENTATIONS OR WARRANTIES. IN ADDITION, SUN MICROSYSTEMS, INC. DISCLAIMS ALL IMPLIED REPRESENTATIONS AND WARRANTIES, INCLUDING ANY WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

This document contains proprietary information of Sun Microsystems, Inc. or under license from third parties. No part of this document may be reproduced in any form or by any means or transferred to any third party without the prior written consent of Sun Microsystems, Inc.

Sun, Sun Microsystems, the Sun Logo, Java, Jini, SunFire, Netra, Solaris, Jiro, Sun Enterprise, Ultra, Write Once, Run Anywhere, SunNet Manager, and The Network is the Computer are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the United States and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc. UNIX is a registered trademark in the United States and other countries, exclusively licensed through x/Open Company Ltd.

The information contained in this document is not designed or intended for use in on-line control of aircraft, air traffic, aircraft navigation or aircraft communications; or in the design, construction, operation or maintenance of any nuclear facility. Sun disclaims any express or implied warranty of fitness for such uses.

Table of Contents

Table of Contents	iii
List of Tables and Figures	v
UltraSPARC III WINS MICROPROCESSOR INDUSTRY AWARD.....	vi
Introduction	1
UltraSPARC® III Cu Processor Goals	2
Compatibility	2
SPARC V9 Instruction Set Architecture	2
Performance on Old Code	2
Scalability	2
Scaling Over Time and Generations of Technology	3
Scaling Over Implementations	3
Scaling Over Number Of Processors	3
Real Performance	3
Reliability, Availability, Serviceability (RAS)	3
The Four Goals Sum To Two Basic Points	4
Investment Protection	4
Total Cost of Ownership (TCO)	4
UltraSPARC III Cu Processor Features	5
Features to Enhance Real Performance	5
Peak Instruction Issue Rate	5
Pipelining for High Clock Rates	5
Managing Branches	6
Eliminating Memory Bottlenecks	6
Memory Organization	6
On-Chip Memory Management Units:	8
On-Chip L1 Data and Instruction Caches	8
On-Chip Instruction Prefetch Buffer	8
On-Chip L2 Cache Address Tags	8
On-Chip L2 Cache Controller:	8
On-Chip Prefetch Cache	8

On-Chip Write Cache	8
On-Chip Main Memory Controller:	8
On-Chip System Interface Controller	8
Memory Latency	10
On-Chip L1 Data Cache	10
On-Chip L1 Instruction Cache	10
On-Chip Address Tags for L2 Cache	10
Off-Chip L2 Cache Data	10
On-Chip Memory Controller	10
Speculative Execution of Loads	10
Software Prefetching	11
Memory (and I/O) Bandwidth	11
1,368-pin CLGA package	11
256-bit Data Bus to L2 Cache	11
128-bit Data Bus to System (Memory, I/O, any remote CPUs)	11
On-Chip Memory Controller	11
Coherency Checking	11
8-Entry Store Queue	11
Write Cache	11
Accelerating Computations via Special Instructions	13
New Instructions for Processing Media-centric Code	13
New Instructions for Processing Scientific Code	13
New Instructions for Processing Java Code	13
Instructions for Processing Network-centric Code	13
Features to Enhance RAS	14
Error Detection and Correction (EDC)	14
Diagnostic Bus	14
Error Removal And Recovery	14
Remote Diagnosis and Repair	14
Simplicity and Integration	15
UltraSPARC III Cu Implementation	17
Elements of Chip Performance	17
TI's 4-way Parlay	17
Summary	18

List of Figures and Tables

Figure 1: Memory Hierarchy of UltraSPARC III Cu Processors.....	7
Figure 2: Performance Features of UltraSPARC III Cu Processors.....	12
Figure 3: RAS Features of UltraSPARC III Cu Processors.....	16
Table 1: Instruction, Data, and Address Storage in UltraSPARC III Cu Processors ..	9

1. Introduction

This white paper provides a brief overview of Sun Microsystem's latest SPARC® (Scalable Processor **AR**Chitecture) microprocessor, the 1050 MHz (1.05 GHz) 0.15 micron (0.15 μ) copper metal implementation of the UltraSPARC® III processor. Sun's commitment to the advancement of microprocessor technology plays an integral role in the company's overall strategy for expanding the technical boundaries of enterprise network computing, while at the same time lowering a customer's Total Cost of Ownership (TCO) for computer systems. The UltraSPARC III Cu processor provides Sun with the basic hardware foundation on which to build genuinely innovative, highly effective, and truly open solutions to real computing problems.

2. UltraSPARC® III Cu Processor Goals

The design of the latest generation of SPARC® processor, the UltraSPARC III processor, was driven by four major goals:

- (1) Compatibility
- (2) Scalability
- (3) Real Performance
- (4) RAS (Reliability, Availability, Serviceability)

2.1. Compatibility

Since the SPARC/Solaris™ platform supports the largest pool of quality UNIX® software in the world (nearly 13,000 compiled applications), the first and most basic requirement for any new SPARC processor is that it both preserve and extend the SPARC legacy, keeping the programming model intact.

2.1.1. SPARC V9 Instruction Set Architecture

Like the two previous generations of 64-bit UltraSPARC chips, the UltraSPARC III processor complies with Version 9 (V9) of the SPARC Instruction Set Architecture (ISA). V9 processors are upwardly compatible with earlier 32-bit versions of the SPARC ISA.

2.1.2. Performance on Old Code

New generations of the UltraSPARC processor are reviewed with the requirement that old code not only must continue to run, it must run most code substantially faster than it did on the previous generation of machine. True, for the most performance-critical applications, it may be desirable to recompile to take advantage of all the latest performance features available in UltraSPARC III. But with SPARC, the decision to recompile is a customer's choice not a vendor's requirement.

2.2. Scalability

SPARC is literally the “**Scalable Processor ARChitecture.**” SPARC is designed to scale not just in one but in three important respects: over time, across implementations, and up in MP capability (number of processors). In addition, SPARC-based systems readily scale from stand-alone configurations into high-availability clusters.

Scaling Over Time and Generations of Technology. SPARC processors have scaled from a relatively simple initial 32-bit, 16 MHz implementation of SPARC in 1987 to the latest 64-bit, 1050 MHz implementation of the UltraSPARC III Cu processor in 2001. A number of forward-looking choices made in the UltraSPARC III processor design will enable this generation of processor to derive full benefit from the coming 0.13 μ generation of process technology, featuring smaller and faster gates.

Scaling Over Implementations. At a given point in time, SPARC processors scale over a wide range of implementations, from very powerful chips intended for servers to very inexpensive chips intended for performance-intensive embedded applications.

Scaling Over Number Of Processors. Systems using exactly the same UltraSPARC III processor implementation can scale from uniprocessor desktop machines to the most powerful Multi-Processing (MP) servers available today, with a near-linear increase in performance for each new CPU added to a system. As a result, Sun systems can meet increased demands for compute power just by adding more CPUs, as necessary.

2.3. Real Performance

The UltraSPARC III processor has been designed to deliver not just great performance running highly optimized benchmark programs on carefully tuned machines under precisely controlled laboratory conditions, but the highest possible *real performance* when used by customers to do actual work in normal business environments, running applications like:

- Enterprise intranets
- eCommerce
- On-line Transaction Processing (OLTP)
- High Performance Computing (HPC)

This means that the UltraSPARC III processor design addresses a variety of key performance issues, extending well beyond a high clock rate and wide parallel instruction dispatch. These include efficient branch management, minimizing the time needed to access memory, high data throughput, and new specialized instructions to accelerate key applications.

2.4. Reliability, Availability, Serviceability (RAS)

For Sun's customers, it's not enough simply to build a system able to produce an accurate answer quickly. Equally important, the system must continue to operate reliably for weeks, months, even years at a time, despite being subjected to the most stressful loads. It must be available on a 24 x 7 basis, to everybody that needs access, at any time they need access. And if the system should happen to go down for any reason, it must be possible to put it back into service quickly.

2.5. The Four Goals Sum To Two Basic Points

Add them together, and the four goals of compatibility, scalability, real performance, and RAS sum up to two fundamental points. For customers who commit their business to Sun systems, UltraSPARC processors:

- (1) provide *investment protection*, and thereby help
- (2) lower their *Total Cost of Ownership* (TCO) for computing resources.

2.5.1. Investment Protection

It makes sense to start doing business with Sun because Sun systems are not only easy to live with in the present, but easy to stay with in the future – regardless of how rapidly business grows and evolves, or what new capabilities may be demanded of network computing. Specifically, customers of Sun’s UltraSPARC-based servers and workstations can:

- Buy systems today with leading-edge performance on the applications that matter most to their business.
- Depend on these systems for their most mission-critical computing needs, comfortable in the knowledge that they will bear up under the most stressful conditions, deliver outstanding uptime, and be easy to service and upgrade.
- Scale up their servers to meet even the sharpest increases in workload as business grows, simply by adding more CPUs as needed.
- Upgrade to new generations of machines with greatly enhanced capabilities, when a hardware upgrade makes good business sense.
- Be confident not only that their current applications will run, but that they will run much faster than they did on the older generation of systems.

2.5.2. Total Cost of Ownership (TCO)

The bottom line about Sun’s ability to protect the investment that its customers make in computer systems is literally the bottom line. When measured by overall TCO, Sun systems are among the most cost-effective in the industry today because, for each new generation of UltraSPARC processor, Sun makes sure that:

- Systems at a wide range of price/performance points are built around a core pipeline. This makes it possible to achieve economies of scale, by consolidating operations from the desktop through the datacenter around a single, binary-compatible architecture, able to share the same software, tools and learning.
- Different versions of a basic design (like UltraSPARC III A1 and UltraSPARC III Cu) preserve the same “footprint.” This makes it relatively easy and inexpensive to upgrade systems by swapping or adding boards.
- Processor boards with both slower and faster CPUs can be integrated into the same chassis.
- Above all else, as mentioned earlier, UltraSPARC carefully preserves not just *binary compatibility* from generation to generation of processor, but what is much harder, *near-optimal performance on old binaries*.

3. UltraSPARC III Cu Processor Features

Sun began shipping systems based on the UltraSPARC III processor in 2000. The initial version of the UltraSPARC III processor was fabricated by Texas Instruments (TI) in a 0.18 μ process technology with aluminum (Al) metal layers. Implemented in this technology, the UltraSPARC III processor operates at frequencies of 600 MHz and 750 MHz. In 2001, the UltraSPARC III processor was upgraded to take advantage of a new TI process technology featuring 0.15 μ features with Copper (Cu) metal layers. Implemented in this more advanced generation of technology, the UltraSPARC III processor operates at frequencies of 900 and 1050 MHz.

The features of the UltraSPARC III processor design intended to improve *real performance* and meet the needs of enterprise-class systems for the highest levels of *RAS* are discussed below. In the following account, all descriptions apply equally to both implementations of the UltraSPARC III processor, unless the Cu part is specifically mentioned.

3.1. Features to Enhance Real Performance

Features of the UltraSPARC III processor design specifically intended to enhance the real performance delivered by this processor in networked enterprise computing environments include a high peak instruction issue rate, a complete redesign of the pipeline to enable much higher clock rates, a better scheme for handling branch decisions, a systematic and comprehensive attack on memory bottlenecks, and new instructions to accelerate specific computing tasks.

3.1.1. Peak Instruction Issue Rate

Like earlier generations of UltraSPARC, the UltraSPARC III processor maintains a sustainable issue rate of 4 instructions every clock cycle (the maximum number of instructions that can be fetched from cache in a clock cycle). A total of 16 fetched instructions can be queued up, waiting for an appropriate execution unit to become available. There are six parallel execution units: 2 integer ALUs (both identical), 1 branch unit, 1 load/store unit, and 2 floating-point units (1 for add/subtract, 1 for multiply/divide operations).

3.1.2. Pipelining for High Clock Rates

To achieve high clock rates, the new UltraSPARC III processor execution pipeline is segmented into 14 separate stages. As a consequence, this pipeline design is able to scale up from its initial frequency of 600 MHz to its current frequency of 1050 MHz to well past 1500 MHz in the next generations of semiconductor technology.

3.1.3. Managing Branches

The UltraSPARC III processor implements an advanced branch prediction mechanism based on 4 KBs of stored branch history that accurately predicts whether or not a branch will be taken about 95% of the time. Wrong predictions have a penalty of 7 cycles, required to back up, fetch the branch instruction again, and start over.

However, the UltraSPARC III processor takes steps to mitigate the impact of the occasional misprediction. Specifically, on predictions a branch will be taken, while the branch target instructions are being fetched, the “fall-through” instructions are readied for issue in parallel, using a special 4-entry Branch Miss queue. In the unlikely case the branch taken prediction turns out to be wrong, this precaution reduces the penalty for error to just 3 cycles. Although predictions that a branch will be not-taken, if mistaken, still must pay the full 7 cycle penalty, when the two misprediction cases are averaged together (weighted by relative frequency), the overall branch misprediction penalty is lowered to about 4.5 cycles.

3.1.4. Eliminating Memory Bottlenecks

The UltraSPARC III processor incorporates a thorough and systematic attack on memory-related performance issues, starting with the design of the memory hierarchy itself, and extending on to considerations of memory latency (access time) and bandwidth.

3.1.4.1. *Memory Organization*

Like other 64-bit processors, the UltraSPARC III processor lets programs operate in a huge virtual address space of 16 quintillion bytes, 4 billion times larger than the 4 gigabyte address limitation of 32-bit processors. To devices outside the chip, it provides a generous 43 bits of address data – sufficient to address 8 trillion bytes (8 TeraBytes) of physical memory.

A comprehensive assault on memory-related performance issues starts with an efficient memory hierarchy, able to systematically shift the instructions and data needed for computations as close to the CPU as possible, and hold them there for as long as necessary. In addition to main memory, the UltraSPARC III processor supports two levels of cache memory. The first level (L1) consists of four separate caches, two large and two small. The second level (L2) consists of a single large cache.

The two large L1 caches hold instructions and data, respectively. The two small L1 caches are a prefetch cache, mainly used by speculatively executed load instructions to hold floating-point data, and a write cache, that serves to greatly reduce the amount of storage bandwidth required. All four L1 caches are on-chip.

The one large L2 cache is a unified instruction and data cache. The contents of the L2 cache are stored off-chip (in SRAM), but the L2 address tags are kept on-chip for faster access. Figure 1 shows the memory organization of an UltraSPARC III processor system and the associated performance strategy. Table 1 shows the details

about the various elements of the UltraSPARC III processor memory system listed below.

On-Chip Memory Management Units: The instruction MMU contains two TLBs that are accessed in parallel during address lookup: a small TLB with 16 entries, and a large TLB with 128 entries. The data MMU in UltraSPARC III Cu contains three TLBs that are accessed in parallel: a small 16 entry TLB, and two identical 512-entry TLBs. (Each TLB entry holds a virtual address, the corresponding physical address, and some associated page protection and usage information.)

On-Chip L1 Data and Instruction Caches: The L1 data and instruction caches both use Sum-Addressed Memory (SAM) for address calculations, a technique that combines two separate calculations into one. The data cache also uses micro tags to store 8-bit virtual indexes. Both techniques serve to speed up cache access, improving performance.

On-Chip Instruction Prefetch Buffer: This small buffer is used when a miss in the L1 instruction cache hits in the first 32-bytes of a 64-byte line or sub-block in the L2 cache. In this case, the first 32-bytes of the line are filled into the I-cache, and the second 32-bytes (needed if processing continues sequentially) are stored in the IPB.

On-Chip L2 Cache Address Tags: The primary performance benefit of keeping the L2 tags on-chip is a reduction in the latency of L2 address calculations. See section on memory latency (3.1.4.2.) for further discussion.

On-Chip L2 Cache Controller: The L2 cache controller supports a large 8 MB external cache, as well as smaller 1 MB and 4 MB sizes (for systems that need less performance and prefer, instead, to optimize for cost).

On-Chip Prefetch Cache: This small 2 KB cache is accessed in parallel with the L1 data cache for floating-point loads. Data may be placed in this cache by floating-point load misses, or hardware or software prefetches. The prefetch cache provides a holding place on-chip for off-chip data that can be fetched well ahead of its use.

On-Chip Write Cache: The primary performance benefit of this small 2 KB cache is its ability to conserve bandwidth. See section on memory bandwidth (3.1.4.3) for further discussion.

On-Chip Main Memory Controller: The on-chip Memory Controller Unit (MCU) provides built-in support for 133 MHz SDRAM chips. Depending on the number and density of SDRAM chips used, the size of main memory can range from a minimum of 128 MBs up to a maximum of 16 GBs. The MCU provides control signals only to memory, all the resulting data traffic itself goes through the system data bus, controlled by the system interface unit (see next).

On-Chip System Interface Controller: The on-chip System Interface Unit (SIU) implements all the logic necessary to support efficient protocols for managing bus traffic and cache coherency. As a result, all that's needed to connect an UltraSPARC III processor to a system is a simple data switch chip that provides a crossbar between the UltraSPARC III CPU, main memory, and the interconnect to system data (I/O and, in MP configurations, "distant" memory attached to other processors).

Figure 1: Memory Hierarchy of the UltraSPARC III Cu Processor

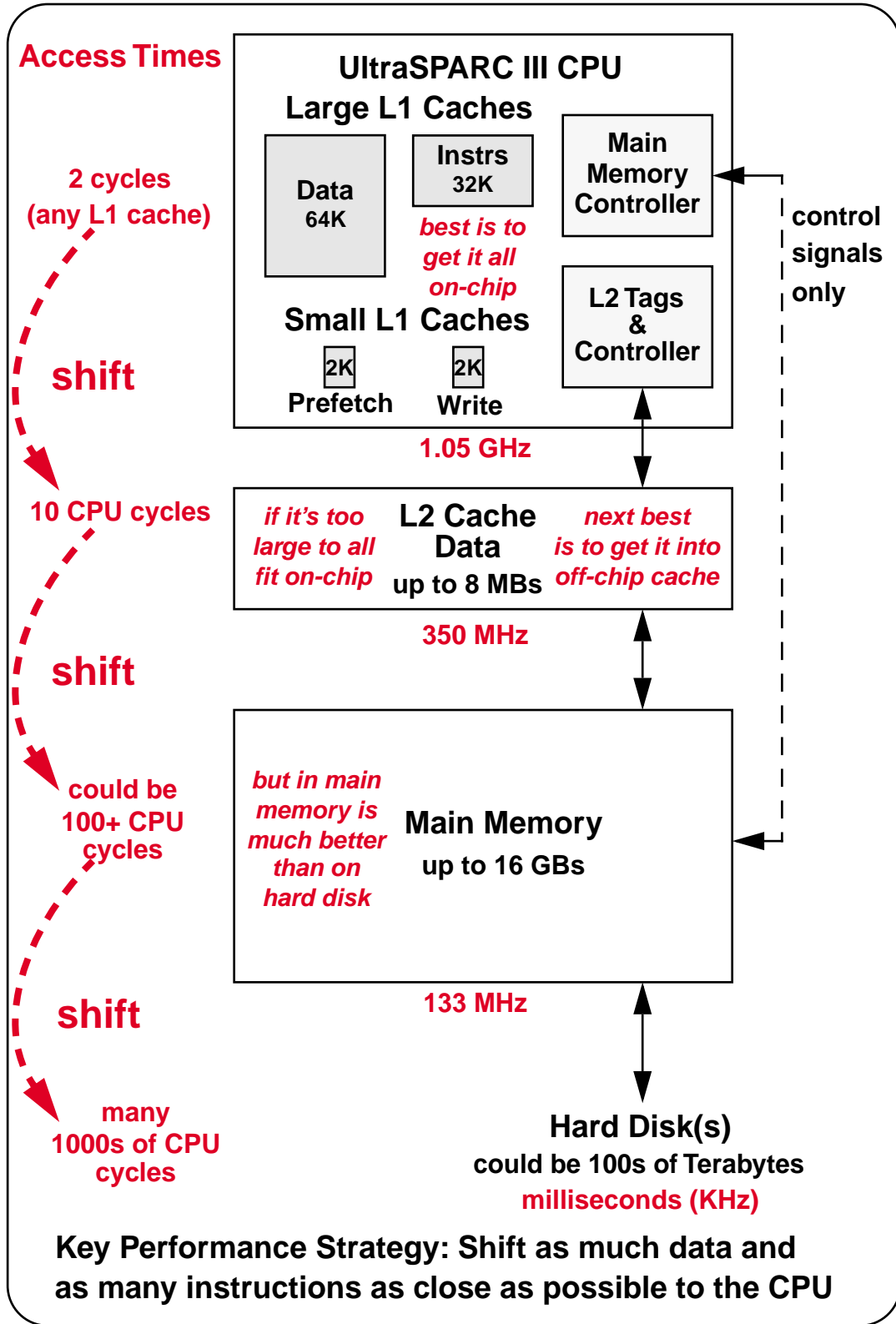


Table 1: Instruction, Data, and Address Storage in the UltraSPARC III Cu Processor

STORAGE BLOCK	LOCATED	LATENCY (CYCLES)	SIZE	SET-ASSOCIATION	LINE LENGTH (BYTES)	LINE TAGS	WRITE POLICY
I-TLBs (2)	on-chip	2	16 entries 128 entries	fully 2-way	N/A N/A	N/A	note ¹ note ²
D-TLBs (3)	on-chip	2	16 entries 512-entries 512-entries	fully 2-way 2-way	N/A N/A N/A	N/A	note ¹ note ³ note ³
L1 I-cache	on-chip	2	32 KB	4-way	32	VI, PT ⁴	write-invalidate
L1 D-cache	on-chip	2	64 KB	4-way	32	VI, PT ⁴	write-through no write-allocate
I Prefetch Buffer	on-chip	about 5	1 line	N/A	32	PI, PT ⁵	N/A
Prefetch Cache	on-chip	2	2 KB	4-way	64 ⁶	PI, VT ⁷	write-invalidate
Store Queue	on-chip	varies	8 entry	N/A	N/A	N/A	N/A
Write Cache	on-chip	varies	2 KB	4-way	64	PI, PT ⁵	write-back ⁸ write-allocate
L2 Address Tags	on-chip	2	90 KB	N/A	N/A	N/A	N/A
L2 Cache	off-chip	about 15	8 MB ⁹	2-way	512 ¹⁰	PI, PT ⁵	write-back write-allocate
Main Memory	off-chip	many ¹¹	up to 16GB	N/A	N/A	N/A	N/A

Note 1: Stores entries for 8 KB, 64 KB, 512 KB, 4 MB, locked or unlocked pages.

Note 2: Stores entries for 8 KB, unlocked pages.

Note 3: Stores entries for unlocked pages. Can only access/fill pages of one size at one time, but can be set to any of the four page sizes, and can handle multiple pages of that size at a time.

Note 4: Virtually Indexed, Physically Tagged.

Note 5: Physically Indexed, Physically Tagged.

Note 6: Has 2 32-byte sub-blocks with separate valid bits.

Note 7: Physically Indexed, Virtually Tagged. On the snoop side, this cache is PI, PT.

Note 8: Maintains dirty bits on a per byte basis.

Note 9: 4 MB and 1 MB sizes also are supported.

Note 10: Line length will vary with cache size, A 4 MB cache requires a 256-byte line while a 1 MB cache has a 64-byte line. Lines longer than 64 bytes are divided into 64-byte sub-blocks; the sub-blocks are used for purposes of both line fill and coherency checking.

Note 11: The precise number of cycles needed will vary with the implementation.

3.1.4.2. *Memory Latency*

A second critical factor in avoiding memory bottlenecks is latency, measured in terms of the number of CPU clock cycles needed to access memory. The UltraSPARC III processor pays considerable attention to the issue of reducing memory latency by as much as possible at every level of the memory hierarchy.

On-Chip L1 Data Cache: The UltraSPARC III processor maintains a fast 2 cycle access to all its on-chip caches. This means the latency penalty for accessing data found in the L1 data cache can be avoided by inserting (just) one instruction issue group between a load instruction and the instruction that uses the loaded data.

On-Chip L1 Instruction Cache: Since the UltraSPARC III processor can execute up to four new instructions every clock cycle, the L1 instruction cache (like the other on-chip caches) uses a technique called *wave pipelining* to enable launch of the address for the next group of four instructions before the preceding group has been received. The result is a steady supply of instruction groups delivered to the (six parallel) UltraSPARC III processor execution engines every clock cycle.

On-Chip Address Tags for L2 Cache: The UltraSPARC III processor integrates into the CPU not just the controller for the off-chip L2 cache but the address tags for the L2 cache itself. The ability to operate the L2 address tags at CPU clock speeds accelerates hit/miss calculations, shaving up to 10 cycles off accesses to main memory on cache misses. In MP configurations, the latency of coherency transactions is likewise greatly reduced.

Off-Chip L2 Cache Data: The external cache can operate at any of several different divisions of the processor clock speed, allowing SRAMs to be selected either for higher performance or lower cost. Using high-performance 350 MHz SRAMs, the best-case latency to the L2 cache for a 1050 MHz UltraSPARC III processor is about 15 clock cycles. After receiving address information, a 350 MHz L2 cache delivers the data requested by a 1050 MHz CPU at a rate of 32 bytes every 3 CPU clock cycles.

On-Chip Memory Controller: Integrating the MCU into the CPU reduces the time needed to access memory by over 30% (relative to an off-chip memory controller). To help minimize latencies to shared data in MP configurations, the memory banks of different processors can be interleaved with one another.

Speculative Execution of Loads: To help manage the inherent latency involved in fetching items from memory, the UltraSPARC III processor provides the ability to execute loads speculatively, i.e., to launch them early, out of sequence. For items in the L1 cache, as mentioned above, this means moving loads up by (just) one issue slot.

For items in main memory, however, many CPU cycles will elapse between the time an item is requested and the time it is received. To support speculation here, UltraSPARC provides special non-faulting load instructions for use by the compilers, which have the large overview required for planning to acquire data far in advance of use. Hardware will initiate speculative loads, too, whenever a regular pattern can be detected in the loads requested by software.

Software Prefetching: Compilers also can get data ahead of need by issuing explicit “prefetch” instructions. Prefetches have considerable latitude in the way they execute. Data can be prefetched only for reading or for reading and writing. Data can be speculatively put in the external cache, or only in the on-chip prefetch cache.

3.1.4.3. *Memory (and I/O) Bandwidth*

In addition to constructing a memory hierarchy that systematically shifts needed data and instructions as close to the execution engine as possible (organization), and lowers the number of clock cycles needed to access every level of the memory hierarchy (latency), a comprehensive assault on memory-related performance issues also must strive to increase bandwidth, the amount of data that can be transferred in a given unit of time. Bandwidth is a function of bus *width* (number of bits that can be transferred in parallel) multiplied by bus *frequency* (rate at which the bus cycles).

1,368-pin CLGA package: The UltraSPARC III processor makes use of a high pin-count Ceramic Land Grid Array (CLGA) package, largely in order to support two separate data buses that total to a width of 384 bits (not counting associated check and address bits). The bandwidth across these two buses aggregates to 13.6 GBs a second.

256-bit Data Bus to L2 Cache: Using 350 MHz SRAMS, the interconnect to the L2 cache allows the CPU to transfer information in and out at a rate of 11.2 GBs a second (moving 32 bytes at a time, 350 million times a second) – enough bandwidth to completely turn over the contents of an 8 MB L2 cache 1400 times in a second.

128-bit Data Bus to System (Memory, I/O, any remote CPUs): The main system bus can be set to cycle at any of several divisions of the CPU clock rate, up to a maximum rated frequency of 150 MHz (1/7 the clock rate of a 1050 MHz CPU). At 150 MHz, the CPU can transfer data and instructions to and from the system at a rate of 2.4 GBs a second (moving 16 bytes at a time, 150 million times a second) – enough bandwidth to completely read or write a 16 GB main memory in 6.67 seconds.

On-Chip Memory Controller: The MCU can handle up to 15 outstanding load/store requests at the same time, with out-of-order completion.

Coherency Checking: Whenever cache snooping is necessary in MP systems, it is done at the maximum speed possible (since all cache tags are kept on-chip).

8-Entry Store Queue: This buffer for store operations serves to decouple store issue from store execution (in just the same way the 16-entry instruction queue decouples instruction fetch from instruction execution). Stores stay in this queue until they complete an update to the Write cache, allowing related stores to coalesce together and greatly reducing the demands on external bandwidth.

Write Cache: This new on-chip cache in UltraSPARC III, while small (2 KBs), is big enough to dramatically impact store traffic. By working with the Store queue to buffer stores, the Write cache often can merge multiple related smaller operations into one larger operation, able to take better advantage of the wide data bus to the L2

cache. The result is a very substantial reduction in write-through traffic from L1, eliminating up to 90% of the separate store operations required.

3.1.5. Accelerating Computations via Special Instructions

A non-faulting load is one example of a specialized instruction supported because of its usefulness in certain computational circumstances, when the need for specific data can be known well before it is ever used. In fact, there are a variety of important computing tasks that involve executing one or more highly specialized operations a very large number of times.

3.1.5.1. New Instructions for Processing Media-centric Code

UltraSPARC I was the first general-purpose processor to include a special set of instructions to accelerate media and graphical applications, called VIS (predating Intel's similar MMX instruction set for Pentium processors by nearly two years). The UltraSPARC III processor extends UltraSPARC's legacy of leadership in this area with a new byte-shuffle instruction that makes it fast and easy, e.g., to extract pixel data from an image. This and other VISible improvements warrant a new version number, to distinguish the 2.0 VIS functionality available with the UltraSPARC III processor from the 1.0 VIS functionality available with the first two generations of UltraSPARC processors.

3.1.5.2. New Instructions for Processing Scientific Code

The set of parallel arithmetic operations at the core of VIS provides the functionality needed to accelerate a wide variety of the technical algorithms typical of High Performance Computing (HPC). VIS 2.0 adds support for interval arithmetic, making it very fast, e.g., to switch rounding modes (from round up to round down, or vice-versa) in a numerical computation. Since HPC applications often involve juggling large numbers of elements from huge memory arrays, the 1184 entries in 5 separate TLBs provided by the UltraSPARC III Cu processor also can accelerate performance, by reducing the number of times memory addresses have to be computed.

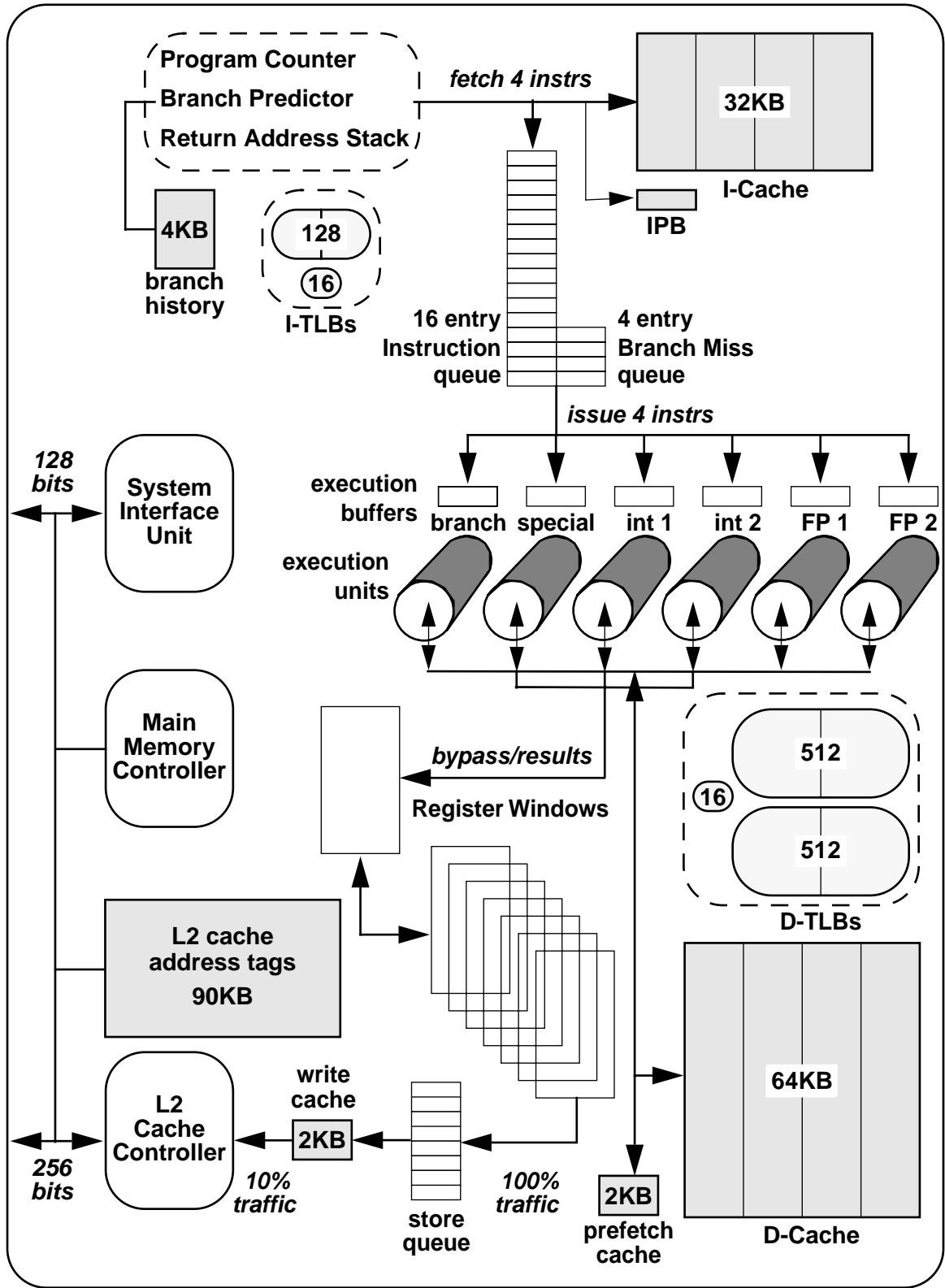
3.1.5.3. New Instructions for Processing Java Code

In addition to improved dynamic branch prediction, the UltraSPARC III processor adds a prepare-to-branch instruction, that lets software specify ahead of time the location of branch target instructions to the fetch unit, so this address can be computed in advance. In Java code, e.g., where a byte code interpreter might spend up to half its time jumping to new routines, the performance improvement from this new instruction can be very noticeable.

3.1.5.4. Instructions for Processing Network-centric Code

Like the earlier generations of UltraSPARC, the UltraSPARC III processor offers pipelined block load and store instructions that allow data to be copied from one

Figure 2: Performance Features of the UltraSPARC III Cu Processor



location in main memory to another, without being stored in cache along the way. This speeds the data transfer (and avoids disrupting the cache), greatly increasing the ability of software to effectively utilize available bandwidth – a considerable plus, e.g., in Net-centric applications, that otherwise might bog down under heavy traffic loads.

3.2. Features to Enhance RAS

While better performance is always important, even the fastest systems are of no value when they break down. For mission-critical applications, RAS is a far more basic consideration than peak performance.

3.2.1. Error Detection and Correction (EDC)

To ensure the integrity of stored data, the UltraSPARC III processor guards against the various errors that can occur in large semiconductor memory arrays. The two large L1 caches on-chip are both protected by error recovery code, able to detect any single bit errors due to background radiation or other causes. If a parity error occurs, the erroneous L1 cache line is marked as invalid, triggering a refill of the line with correct data from the lower memory hierarchy on the next attempt to access it.

The off-chip memories (the L2 cache, including its on-chip address tags, and main memory) are protected by Error Correcting Codes (ECC). These can both detect and correct any erroneous single-bit, as well as detect (but not correct) double-bit errors. Data affected by errors than cannot be corrected is marked as bad, to prevent it from being reused or spreading to other processors in an MP system.

3.2.2. Diagnostic Bus

The UltraSPARC III processor includes a special 8-bit bus that operates independently of the main system bus. The CPU can boot from flash PROMs over this bus, as well as access peripherals like serial consoles and scan controllers. This lets an UltraSPARC III processor system be powered on, configured, and tested even if the majority of the system is not operational. During normal operations, the diagnostic bus can be used to sample the processor's internal state in real time, enabling both early detection of failing mechanisms and proactive maintenance to prevent or limit failures.

3.2.3. Error Removal And Recovery

Since each UltraSPARC III CPU controls its own local memory (if it has any local memory), detecting and correcting all single-bit corruption in it, the threat of a data error propagating to other CPUs in an MP system is eliminated in many cases. Where errors cannot be corrected, the ability of the UltraSPARC III processor to mark data as bad at least allows errors to be contained. Further, its ability to identify

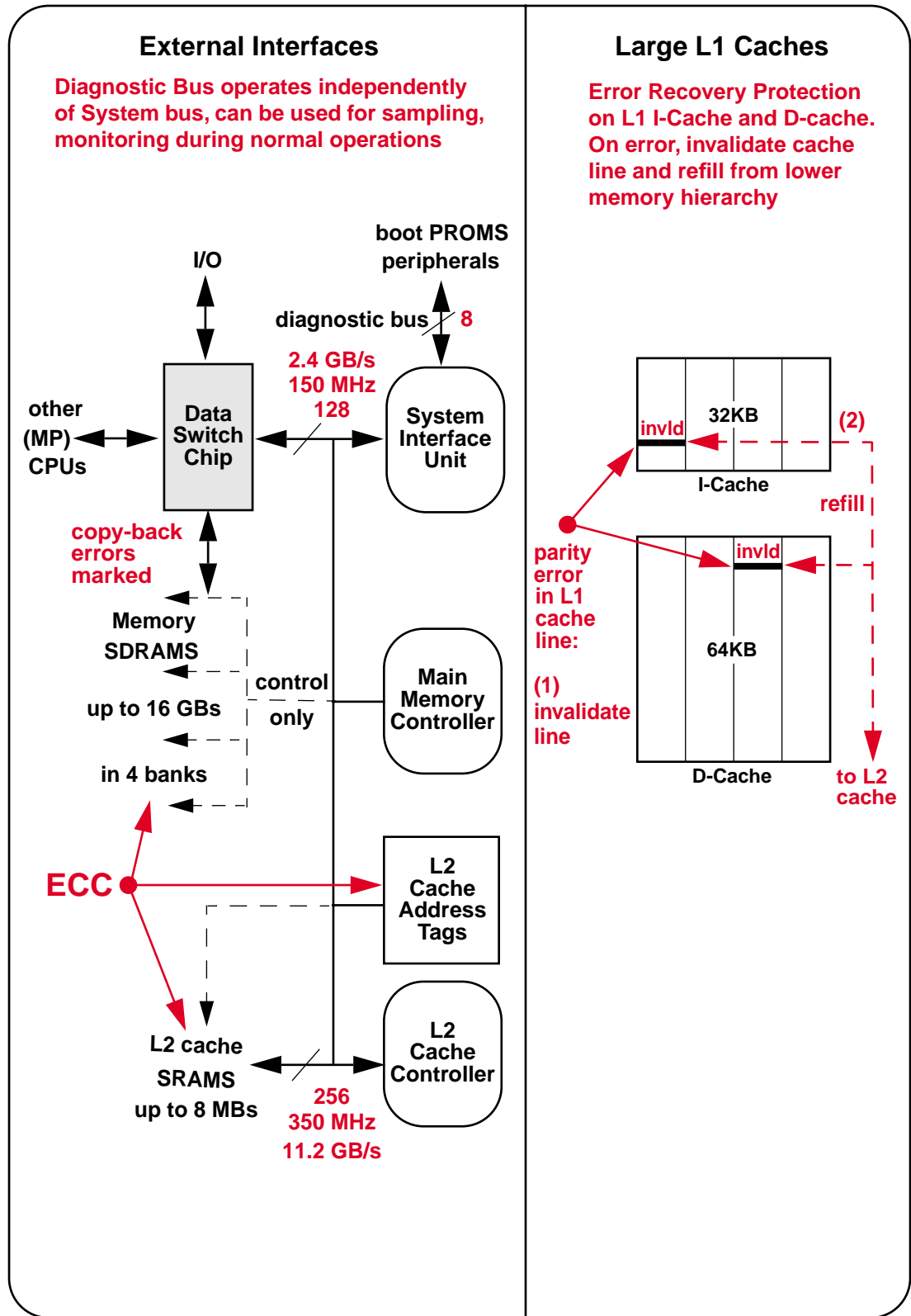
the source of an error, up to and including singling out a processor with uncorrectable problems for replacement, greatly aids in the recovery process.

Remote Diagnosis and Repair. The ability of the UltraSPARC III processor to uniquely distinguish the source of a failure, together with the ability to access the UltraSPARC III processor across the diagnostic bus even when the system bus is inoperable, allows technicians to remotely identify and possibly repair a failing unit.

3.2.4. Simplicity and Integration

Since simpler mechanisms are inherently more reliable, the UltraSPARC III processor generally favors the simplest approach consistent with its several goals. Thus, the instruction issue unit is static (requiring a relatively simple in-order issue unit) not dynamic (requiring a much more complex out-of-order issue unit). By integrating into the CPU chip vital system components like the L2 cache controller, the main memory controller, and the system interface controller, reliability as well as performance is enhanced. Every chip eliminated from the system design is one fewer point of failure, and one less set of interconnects to go bad.

Figure 3: RAS Features of UltraSPARC III Cu



4. UltraSPARC III Cu Processor Implementation

For high-performance processors, as a rule, each new generation of architecture is matched to the latest generation of process technology, in order to exploit the higher clock rates (performance) and transistor budgets (functionality) associated with more advanced process technologies (using smaller feature sizes).

4.1. Elements of Chip Performance

The two critical issues that control processor frequency for a given semiconductor implementation technology are the speed of the transistors and the speed of the wires provided by that technology. These two key parameters are usually couched in terms of the *delay* occasioned when an electrical signal encounters one or the other of these two chip components, namely:

- For transistors, *gate delay*, or how long a transistor takes to switch (open or close), turning on or shutting off the flow of electricity along a path.
- For wires, *wire delay* (also known as *interconnect delay*), or how long it takes an electrical current to flow from one end of a wire to the other.

4.2. TI's 4-way Parlay

Sun has a long history with Texas Instruments (TI) for fabrication of high-end SPARC processors, stretching back to 1988. The genuine benefit of this partnership to each party is evident from its enduring nature, spanning three decades.

The GHz+ clock speed of the UltraSPARC III Cu processor is largely due to a 4-way combination of advanced technologies used by TI in their 0.15 μ process generation. The first two items below are specific to the issue of reducing gate delay. The second two items are specific to the issue of reducing wire delay. The combination of fast gates and fast wires adds up to a process technology able to deliver outstanding performance.

- **100 Nanometer Gates:** The effective gate length of 100 nm. provides outstanding gate speed (low gate delay), as shown by the 1.05 GHz clock frequency of the UltraSPARC III Cu processor.
- **Nitrided Gate Oxide:** The use of a new nitrided gate oxide (RPNO), in place of silicon dioxide for gate insulation, allows use of a physically thicker layer with greatly decreased current leakage.
- **Copper Metal Layers:** The use of a new highly-conductive metal (copper) for metal layers significantly lowers resistance in the thin wires required by the 0.15 μ generation of process technology.

- ***Low K Metal Layer Insulation:*** The use of a new low *k* dielectric (FSG) to insulate tightly spaced wires lowers their capacitance, improving signal speed. In combination with the new copper metal, it's a double win in terms of maintaining fast signal transmission across wires.

5. Summary

The UltraSPARC III Cu processor has been architected to provide software compatibility with all previous generations of SPARC design, both 32-bit and 64-bit, achieving substantially higher performance on unrecompiled binaries. It provides customers with the highest levels of support for:

- Scalable multiprocessing systems.
- Real performance on enterprise applications.
- RAS vital for mission-critical systems.

In order to address the new challenges and opportunities that will arise in the coming decade, Sun will need to build new kinds of systems, based on new generations of Solaris™ operating environment and the UltraSPARC processor. The gap between those future systems and the systems of today is likely to be as great or greater than the gap between today's 1050 MHz, 64-bit UltraSPARC III Cu processor systems and the 40 MHz, 32-bit SuperSPARC I processor systems of a decade ago.

Nonetheless, despite all the changes that surely will occur over the next decade, there also are some basic things about Sun systems that certainly will not change. Chief among these enduring attributes are the fundamental values that SPARC/Solaris systems deliver to customers. For current customers, this means real protection for the investment already made in Sun hardware and software, including the widest choice of systems built around a single binary-compatible architecture available anywhere, running the world's largest base of quality UNIX applications. For both current and future customers, Sun's key SPARC and Solaris technologies provide the essential foundation for an unmatched combination of very powerful, extremely dependable, and highly scalable solutions to real computing problems, delivered at an extraordinary value when measured by Total Cost of Ownership.

