The PA 7300LC Microprocessor: A Highly Integrated System on a Chip

A collection of design objectives targeted for low-end systems and the legacy of an earlier microprocessor, which was designed for high-volume cost-sensitive products, guided the development of the PA 7300LC processor.

by Terry W. Blanchard and Paul G. Tobin

In the process of developing a microprocessor, key decisions or guiding principles must be established to set the boundaries for all design decisions. These guiding principles are developed through analysis of marketing, business, and technical requirements.

Several years ago, we determined that we could best meet the needs of higher-volume and more cost-sensitive products by developing a different set of CPUs tuned to the special requirements of these low-end, midrange systems. The PA 7100LC was the first processor in this line, which continues with the PA 7300LC.

This article will review the guiding principles used during the development of the PA 7300LC microprocessor. A brief overview of the chip will also be given. The other PA 7300LC articles included in this issue will describe the technical contributions of the PA 7300LC in detail.

Design Objectives

Although the PA 7300LC was targeted for low-end systems, cost, performance, power, and other design objectives were all given high priority. With the design objectives for the PA 7300LC we wanted to:

- Optimize for entry-level through midrange high-volume systems (workstations and servers)
- Provide exceptional system price and performance
- Roughly double the performance of the PA 7100LC
- Provide a high level of integration and ease of system design
- Provide a highly configurable and scalable system for a broad range of system configurations
- Tune for real-world applications and needs, not just benchmarks
- Emphasize quality, reliability, and manufacturability
- Provide powerful, low-cost graphics capabilities for technical workstations
- Use the mature HP CMOS14C 3.3-volt 0.5-µm process
- Use mainstream, high-volume, and low-cost technologies while still providing the necessary performance increases
- Emphasize time to market through the appropriate leverage of features from previous CPUs.

Meeting Design Goals

We began by leveraging the superscalar processor core found in the PA 7100LC processor. First we investigated the value of high integration. Next we added a very large embedded primary cache, now feasible with the 0.5-µm technology. Then we enhanced the CPU core to take advantage of the new on-chip cache by reducing pipeline stalls. We also ensured high manufacturing yields by adding cache redundancy.

We found that integration supported our design goals in many positive ways. Because the primary cache, the secondary cache controller, and the DRAM controller could be on the same chip (see Fig. 1), we had an opportunity to design and optimize them together as a single subsystem. This was a large factor in allowing us to achieve such an aggressive system price and performance point. The high-integration approach also yielded much simpler system design options for our system partners. To further support these partners, we designed the integrated DRAM, level-2 cache, and I/O bus controller with extensive configurability (see ***Subarticle 6a*** "Configurability of the PA 7300LC"). This configurability enabled a wide variety of system options ranging from compact and low-cost systems to much more expandable, industrial-strength systems.

We were careful not to take a cost-first approach to this design. We believe that performance is just as important for customers of HP's lower-cost systems. We took a total system approach in optimizing performance while emphasizing application performance over benchmarks in making design trade-offs. The highly optimized memory hierarchy shows dramatic improvement for the memory-intensive programs found in technical and commercial markets.

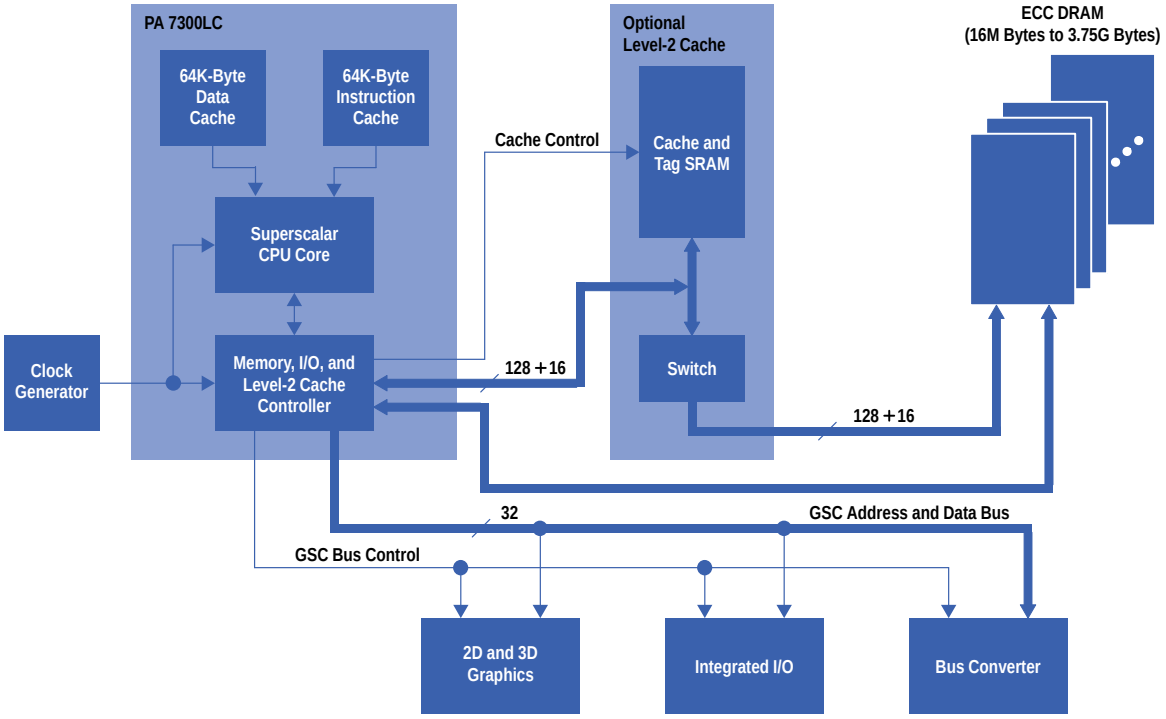

Fig. 1. *PA 7300LC system design.*

Another way of meeting our performance goals was to push the frequency while increasing the level of integration. We focused early on the layout and floor plan of the chip to enable higher-frequency operation. Through this effort, all critical paths were optimized. We tracked and optimized 62,000 individual timing paths during the design phase.

Despite leveraging the design from an existing CPU, the PA 7300LC design team still evaluated a large array of technical features and alternatives to meet our performance goals. Fundamentally, our approach was to build a robust CPU using a simple, efficient microarchitecture. Such a design ran less risk of functional bugs and allowed physical designers more leeway to push their circuits for higher performance.

On-Chip Primary Cache Decisions

It was clear from the beginning that the CMOS14C process would allow an on-chip cache of reasonable size, so a significant investigation was done to determine an optimal cache size and configuration. HP's System Performance Lab in Cupertino, California assisted us by repeatedly running benchmarks and code traces with different cache topologies and memory latencies.

Optimal Cache Size. Finding a balance between instruction-cache and data-cache sizes was difficult. The PA 7300LC was intended for use in both technical markets, where larger data caches are desired, and commercial markets, where programs favor large instruction caches. The standard industry benchmarks can easily fool designers into using smaller instruction caches, trading the space for more data cache or simply keeping the caches small to increase the chip's frequency. HP has always designed computer systems to perform well on large customer applications, so we included them in our analysis. Ultimately, we found that equally sized caches scaled extremely well with larger code and data sets. The typical performance degradation found when a program begins missing cache was mitigated by large cache sizes and our extremely fast memory system.

We could physically fit 128K bytes of cache on the die, so it was split into 64K bytes for the instruction cache and 64K bytes for the data cache. Not only would this provide impressive performance, but we noted that it would be the largest on-chip cache of any microprocessor when it began shipping.

Cache Associativity. Cache associativity was another issue. Recent PA-RISC implementations have used very large directly mapped (off-chip) caches. Associativity would reduce the potential for thrashing in the relatively small 64K-byte caches, but we were worried about adding a critical timing path to the physical design—selecting the right *way** of associativity and multiplexing data to the cache outputs. Increasing the ways of associativity would further reduce the thrashing, but make the timing even worse. The Systems Performance Lab included associativity in their performance simulations, helping us arrive

* Way, or N-way associativity, is a technique used to view a single physical cache as N equally sized logical subcaches. The PA 7300LC caches are two-way associative, so each 64K-byte cache has two ways of 32K bytes each. This provides two possible locations for any cached memory data, reducing the thrashing that can occur in a direct-mapped cache when two memory references are vying for the same location.

at our decision to implement two-way caches. To reduce the impact on timing, we eliminated cache address hashing, which had been used to reduce thrashing in directly mapped cache designs. Once we added associativity, hashing was no longer necessary.

Associative cache designs also need an algorithm for determining which way to update on a cache fill. Again, there are many alternatives, but our simulations showed the easiest approach to be the best. A pointer simply toggles on each fill, so that the ways alternate.

Other Cache Decisions

Many other cache decisions fell out of the same types of analysis. The data cache uses a copy-back rather than a write-through design** and a 256-bit path to the memory controller was included for single-cycle writes of copyout lines as shown in Fig. 2.

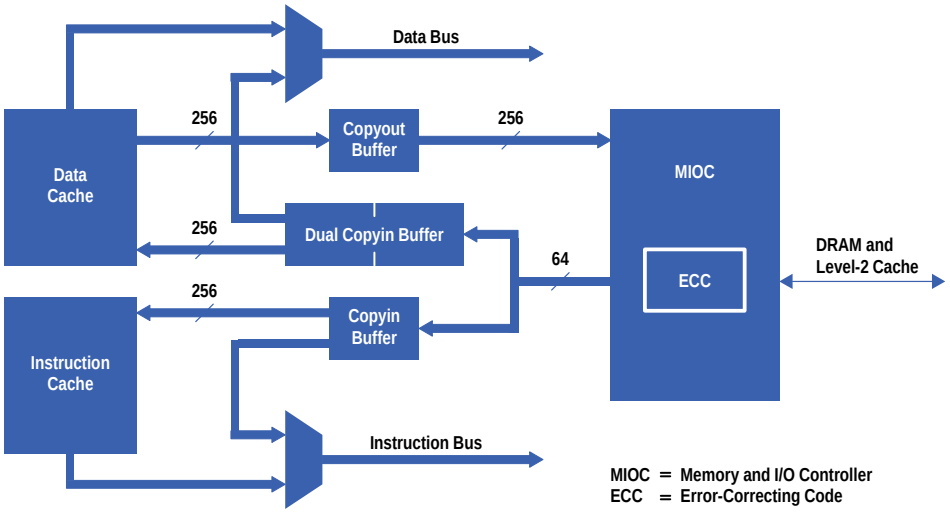

Fig. 2. *Primary cache system on the PA 7300LC chip.*

Moving the caches onto the chip also simplified changing the CPU pipeline to remove the "store-tail" penalty, in which stores on consecutive cycles cause a hang. This made it easier for compilers to optimize code.

Adding Spare Columns to the Cache Arrays. Manufacturability is a big concern for large VLSI memory structures like the PA 7300LC's caches. Dense, regular structures like cache RAM cells are very susceptible to the smallest manufacturing defects, and just one failing bit out of 1,200,992 can make a part useless. To compensate, the cache design team added spare columns to the cache arrays. During the initial wafer test of a CPU die, an internal built-in self-test (BIST) routine runs to check for errors. If a bad RAM cell is found, the BIST signature indicates which column should be swapped out, and a laser is used to blow a special metal fuse on the chip. The bad column is replaced with the spare, fully restoring the chip's functionality. The article on page 1 describes this feature in detail.

Integrated Memory and I/O Controller Decisions. Incorporating the memory and I/O controller (MIOC) onto the PA 7100LC chip was an important performance win, and we worked to make it even better on the PA 7300LC. Simply having the MIOC and CPU on the same die is extremely efficient. An off-chip MIOC would require a chip crossing for each data request and data return. Chip crossings are time-consuming, costing many chip cycles at 160 MHz. Since the CPU stalls on a critical request, chip crossings directly degrade performance.

Chip crossings also require additional pins on packages, driving up the cost. As a result, designers strive to keep external data paths narrow. With the MIOC on-chip, we were able to use wider data paths liberally for faster transfers. We placed some of the MIOC's buffers inside the cache and used wider data paths to create a bus that is one cache line wide for blasting cache copyouts to the MIOC in one cycle.

Cost and Performance Decisions. Despite all the performance enhancements, the increased CPU frequency placed a burden on the MIOC to minimize memory latencies and pipeline stalls because of filled request queues. Blocking for an off-chip resource costs more CPU cycles at higher frequencies, so it was paramount that the PA 7300LC MIOC be fast and efficient. The challenge was in achieving this without significantly increasing the system cost.

** In a write-through cache design, data is written to both the cache and main memory on a write. In a copy-back cache design, data is written to the cache only, and is written to main memory only when necessary.

Doubling the external memory data path to 128 bits was a clear performance advantage, but it also increased system cost. Adding 72 (64 data + 8 error correction) pins to the CPU die and package came at a price. We were concerned that system designers would also be forced to create more expensive memory designs. Configurability was the best solution. The increased performance warranted adding pins to the CPU, but the MIOC was designed to support a 64-bit mode for less expensive memory designs in low-cost systems.

Off-Chip Second-Level Cache Performance. In addition to the primary cache, one of the PA 7300LC's most intriguing features is its second-level cache (see Fig. 3). Even with the MIOC's very fast memory accesses, it takes at least 14 CPU cycles for cache miss data to be returned. While this is excellent by industry standards, we had the opportunity to make it even faster by implementing an off-chip second-level cache.

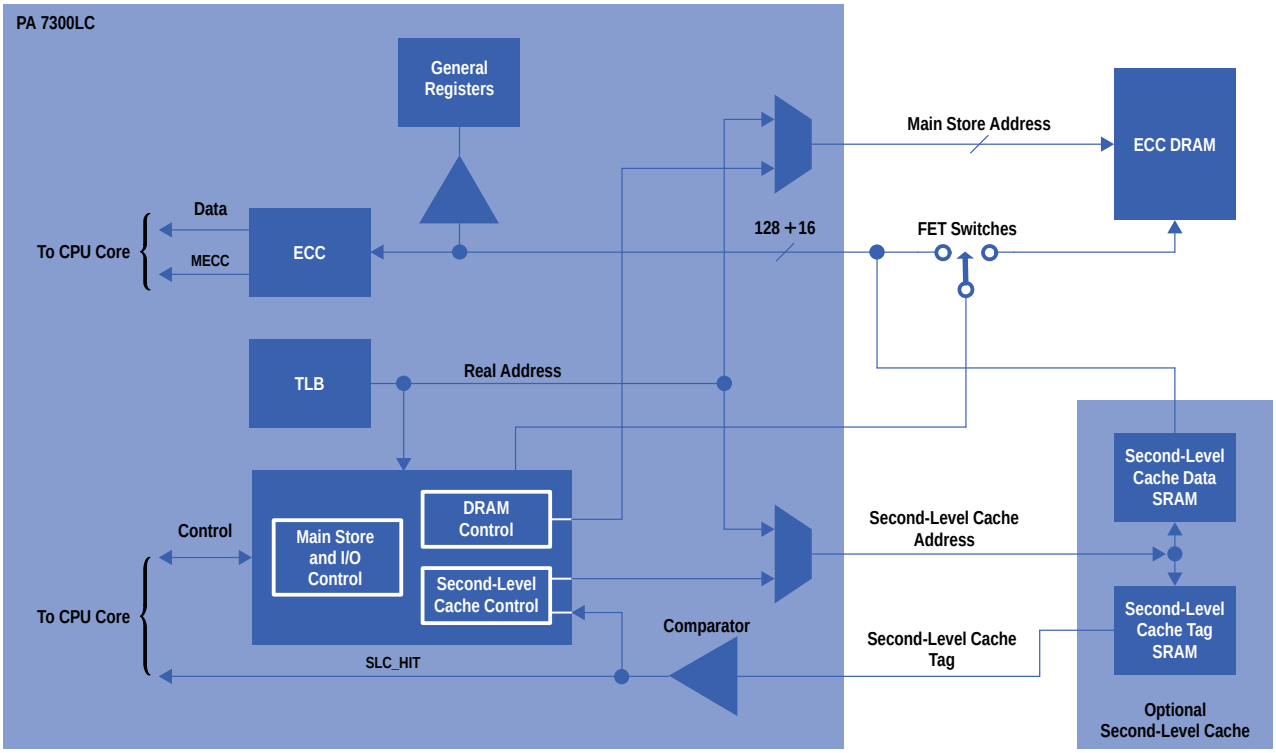

Fig. 3. *Memory subsystem for the PA 7300LC.*

In many cases, the CPU is stalled during the entire memory access. A typical second-level external cache design could drastically reduce the number of stall cycles, but would be expensive. The engineering pros and cons were debated, and a very interesting solution was found. Address pins for a second-level cache were added to the CPU, but the second-level cache and DRAMs share the memory data lines (either 64 or 128). Very fast FET switches are used to shield second-level cache accesses from the heavy DRAM line loads until it is determined that the second-level cache will miss. While adding one cycle to memory accesses, this technique reduces access time to only six cycles on a second-level cache hit. The second-level cache is optional for low-cost systems or for those applications where a second-level cache is not beneficial.

MIOC Design Enhancements. Internally, the MIOC design was enhanced in many areas in the PA 7100LC MIOC. The internal pipeline was split into independent queues for memory and I/O, preventing memory stalls during long I/O operations. Reads can be promoted ahead of memory writes to satisfy CPU requests rapidly, and graphics writes are accelerated ahead of other transactions to increase graphics bandwidth. Finally, the GSC*(general system connect) interface was enhanced to improve graphics bandwidth by well over 200% over the PA 7100LC and to support a broader range of CPU:GSC operating ratios.

CPU Core Decisions

Removing the Phase-Locked Loop. Because of its higher operating frequency, the original PA 7300LC design contained a phase-locked loop circuit to synthesize both CPU and system clocks. Designing a phase-locked loop in a digital CMOS process is challenging and historically has affected yield and robustness in VLSI designs. When an inexpensive external clock part was found, we decided to recover the phase-locked loop circuit area and reduce technical risk by removing it.

* The GSC is the local bus that is designed to provide maximum bandwidth for memory-to-graphics transfers.

Integer and Data Cache Controller Enhancements. The on-chip caches caused both the integer and data cache controllers to be redesigned, and significant enhancements were included in both. The data cache controller added a deeper store buffer, and by also modifying the instruction pipeline, we were able to eliminate completely the store-tail problem mentioned earlier. Also, memory data is bypassed directly to execution units before error correction, with later notification in the rare event of a memory bit error.

The instruction cache controller expanded the instruction lookaside buffer (ILAB) from one entry to four, and improved the performance of bypassing instructions directly from the MIOC to the execution units. Both are very tightly coupled to the MIOC so that memory transfers to and from the caches are extremely fast.

Summary

We developed a set of guiding principles based upon marketing, business, and technical requirements for this system. The guiding principles enabled the design of an exceptional microprocessor targeted to the volume and price/performance requirements of the workstation and server market. A large part of the overall success of this design comes from the well-engineered cache and memory hierarchy. The technology we chose allowed us to develop a high-capacity primary cache and a rich set of performance-improving features.

The PA 7300LC design met its schedule and exceeded its performance goals. Customers are receiving PA 7300LC- based systems today.

Acknowledgments

We wish to thank the rest of the PA 7300LC design team in HP's Engineering Systems Lab in Fort Collins, Colorado for their superb technical contributions. We also express appreciation to our partners in the General Systems Lab and Fort Collins Systems Lab, as well as various HP marketing organizations for providing customer input and requirements. Finally, we express appreciation to the Systems Performance Lab for their efforts in running performance simulations on our behalf. A special recognition is made to Tian Wang (developer of the PA 7300LC performance simulator) who passed away during this development effort. We extend our sympathies to his family.