Reed Electronics Group

In-Stat MDR

Alpha EV7 Processor: A High-Performance Tradition Continues

April 5, 2002

Kevin Krewell
Senior Analyst
Senior Editor,
Microprocessor Report
Phone: 408.345.1624
Email: kkrewell@mdr.cahners.com

In-Stat/MDR
1101 S. Winchester Blvd., Building N, San Jose, CA 95128
www.MDRonline.com

Alpha EV7 Processor: A High-Performance Tradition Continues

Executive Summary

The Compaq EV7 is a processor that defines state of the art. A goal of EV7 processor design was to take the existing Alpha processor core (EV6) and feed it with lots of bandwidth and low latencies. Although the EV6 is already a highly acclaimed CPU, it is capable of greater achievements, and the EV7 achieves the bandwidth and latency goal by incorporating two on-chip RDRAM memory controllers and a very large 1.5MB L2 cache.

A second key goal for the processor was scalability, which poses challenging issues for designers of server system processors, separating the men from the boys. The EV7's memory bandwidth scales the addition of more processors, and it already has the highest memory bandwidth available in any server processor. The low-latency links between processors provide glueless processor scaling to 128 processors. These interprocessor (IP) links are controlled by an integrated network interface that routes the request to the appropriate node in the network, using one of the four ports (which are given compass-point notations of N, S, E, and W).

The large on-chip L2 cache, integrated cache controllers, and IP network contribute to an EV7 die size of roughly 400mm^2, well within manufacturing capabilities. The EV7, in the 0.18-micron semiconductor process, will be followed by the EV79 in a 0.13-micron/SOI process. The EV79 will shrink the die to a very cost-effective 300mm^2 and allow clock speeds greater than the EV7.

With its glueless, low-latency, packet-based interprocessor connections, on-chip RDRAM memory controllers, extensive fault tolerance, and four-issue superscalar core, the EV7 easily fits the definition of a state-of-the-art server processor. Its clock speeds are competitive with those of other 64-bit processors.

The address fields allow for up to 128 nodes, with 32GB per node. Fully built out, a 21364 system could support 4TB of memory, which should be sufficient for some time to come.

Introduction

The original Alpha architecture was conceived in 1988 to last 25 years. We are only 14 years into that timeline, and the Alpha processor is now in its prime. The current Alpha CPU, referred to as EV6, represents the present state of the art in processor core design, with a four-way superscalar core, and was first described at the 1996 Microprocessor Forum. The out-of-order processor supports up to 80 instructions in flight. Renaming registers number 80 for integer operations and 72 for floating-point operations. The integer register file is duplicated to allow faster access, with each integer ALU having a file close by. The two files can transfer and share data with a one-clock penalty.

The primary (L1) caches have substantial 64KB sizes, with a two-cycle access latency. This size cache trades off access time for higher hit ratios. A smaller L1 cache could reduce the access to one cycle, but a smaller L1 cache also has a lower hit ratio.

The Alpha processor had a reasonably long pipeline at seven stages. To avoid bubbles in that pipeline caused by instruction discontinuity, branch prediction is an essential part of a high-performance processor. The misprediction penalty on the 21264 is 11 or more clock cycles, due to the time required to refill the buffer and the time spent in the instruction queue. The 21264 branch predictor is based on a branch history table and combines a local and a global predictor. To speed taken-branch instruction fetches, the 21264 has a "next line" prediction in the cache.

The 21364 processor, code-named the EV7, was described at the 1998 Microprocessor Forum; Figure 1 shows a die photo. The origins of the system design can be traced back to a 1996 column written by Alpha processor inventor Dick Sites entitled "It's the Memory Stupid." The article's title was a play on a political catchphrase, but the message behind it was that memory bandwidth and latency—not just CPU core frequency—were the essential problems that needed to be solved in future generations of server processors. In that article, Sites related a performance analysis that showed that the 21164 core, running at 400MHz back then, was spending about three cycles out of every four just waiting for main memory. The problem that needed solving was not achieving faster core performance but keeping fast cores fed with data and instructions.

Although the processor core is carried forward from the 21264, the EV7's system bus, IP network, on-chip memory controller, and large L2 caches are all new. The 21364 (EV7) will start out in a 0.18-micron semiconductor process at about 1.2GHz and will migrate to a 0.13-micron silicon-on-insulator (SOI) process in 2003/2004. In that process, the processor is known as the EV79 and will feature higher clock frequencies and lower power. The new process will also shrink the die size of the processor up to 33%, making it easier to manufacture. The EV7 has a hefty, but manufacturable, 400mm^2 die size, so the shrink to 300mm^2 will make its manufacture easier and cheaper. Along with the core clock frequency, the processor's RAMBUS memory channels and IP channels can scale up in frequency and bandwidth.

A Great Server Starts With a Great Core

The EV7 processor starts with the core of the 21264 (EV6). This core is largely untouched, as it was already one of the most advanced processor cores available. To enhance its ability to process an increased instruction load, the number of outstanding cache block fills will be increased from 8 to 16.

The EV7 memory hierarchy is straightforward. Misses to the 64KB L1 caches will first access the 1.5MB L2 cache, and if there is a cache hit, the data will be returned on a 128-bit-wide bus. References that miss the L2 cache will access the local memory and return data to the core. Memory locations not located in the local memory will traverse the network to reach the memory of another processor in the system.

The 21264 core has an eight-entry victim buffer that is currently used for both L1 and L2 victims. The new EV7 design will increase the size of the victim buffer to 16- x 64- byte blocks for L1-to-L2 victims. A new 16- x 64-byte victim buffer will be used to hold victims, leaving the L2 cache for the local memory or the network. The buffers allow the caches to quickly flush cache lines and free access for the processor core. Larger buffers are required to support the increased number of processors and interprocessor communications.

The EV6 processor core has a seven-stage pipeline with a two-cycle L1 cache access. Four instructions are fetched from the 64K two-way set-associative instruction cache (I-cache) each cycle and delivered to the integer and floating-point mappers. Integer instructions proceed through the map stage, which will map the 32 virtual register designations into the 80 physical registers available and then insert them into the issue queue.

During every cycle, the integer instruction issue queue will issue up to four instructions (oldest first), press out empty slots (eliminating holes left by previously issued instructions), and accept new instructions from the mapper. In the next cycle, operands are fetched from the register file. To improve register bandwidth, the four integer units share two independent copies of the register file.

The 64K two-way set-associative L1 data cache (D-cache) can process two loads or stores every clock cycle. Cache-miss references are merged into cache block requests that access the L2 cache. Modified data is displaced during the fill and is buffered in a 16-entry victim buffer.

The EV7 processor wraps a sizable 1.5MB, six-way set-associative L2 cache around the processor core, as seen in Figure 1. One side effect of reusing the

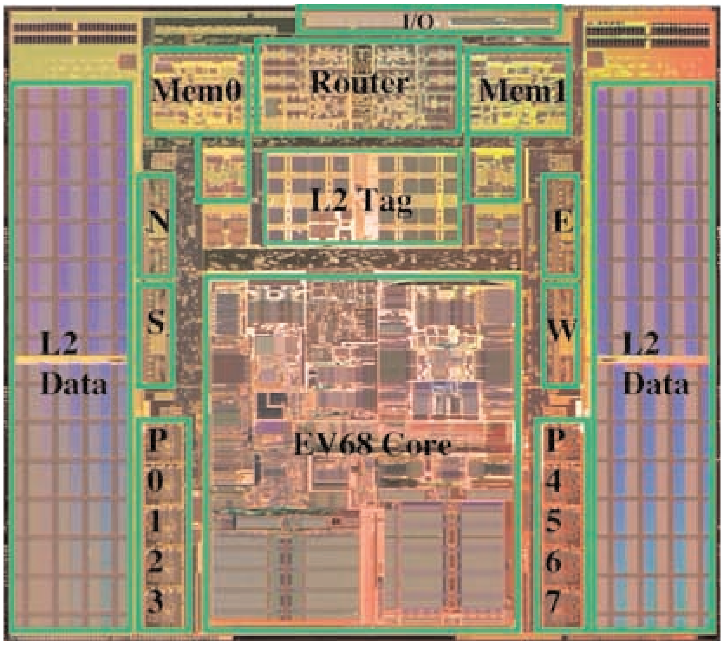


Figure 1. The die photo of the EV7 shows the L2 cache on either side of the die.

EV6 core design is a 12-cycle load-to-use latency that is set by the existing controls in the core. A benefit of the rather long load latency is that it significantly reduces the power consumption of the L2 array. The L2 cache can read or write 16 bytes/cycle at 1GHz, resulting in phenomenal 16GB/s of read or write bandwidth. The L2 array is protected by ECC logic with a single-error-correct, double-error-detect code for fault tolerance.

Need Memory Bandwidth? Call RDRAM

In the search for increased system and memory bandwidth, one obstacle has been the processor front-side bus. Until recently, the processor bus was the bottleneck where memory accesses, I/O accesses, and cache snoops all had to share a bus running at a fraction of processor’s speed. To access main memory, memory accesses had to route through another chip that ran at DRAM bus speeds. An ongoing goal of microprocessor design has been to get memory references closer to the processor core. Most modern high-performance processors feature on-die L2 caches, with server processors often featuring 1MB or more of secondary cache. The EV7 offers 1.5MB, which balances cache size with die size.

The EV7 contains two integrated Direct Rambus (RDRAM) memory controllers. Direct Rambus provides the highest data rate per pin, with outstanding bandwidth and good access latency. RDRAM is also packet oriented, which works well with the IP network. The pin-to-pin delay for a page hit in the RDRAM is 30ns, and the load to use access latency is 75ns.

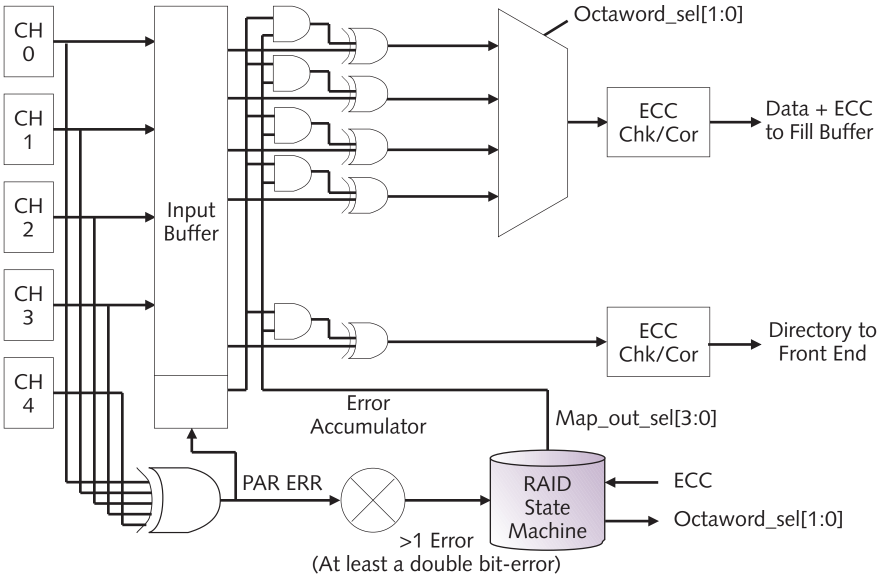


Figure 2. A fifth RDRAM channel is available to be used in a memory RAID configuration. The extra channel protects against the failure of any other channel. If the data ECC exhibits an error greater than one bit, the data access is retried with one channel disabled. The channels are swapped out until a correct packet ECC is obtained.

The choice of RDRAM memory may be considered controversial, but in 1998, it was considered the de facto future of high-performance memory. If we leave all politics of Rambus aside, RDRAM technology offers the most bandwidth per pin of any mainstream memory technology today. The EV7 achieves the highest dedicated memory bandwidth of any server processor by incorporating dual on-chip RDRAM memory controllers. The controllers have been optimized for fast RDRAM performance by supporting 2,048 simultaneous open pages. Each of the two memory controllers on EV7 supports four 18-bit RDRAM channels, a design that offers each processor more than 12GB/s of raw bandwidth. A fifth channel is available for a data RAID configuration, providing exceptional memory fault tolerance.

As with the L2 cache, main memory is protected by a single-error-correct, double-error-detect ECC code. Main memory data is transported in a packet that includes a 9-bit ECC code. Errors can be corrected inline without any additional latency or reduction in bandwidth.

If additional memory error protection is needed (and the probability of errors increases as large systems approach terabyte-size memory arrays), a RAID memory configuration provides greater protection, allowing the machine to survive the failure of entire RIMM modules. A small delay is required to discover the error, then the chip resumes full bandwidth and latency while making the correction. The RAID configuration requires an extra RDRAM channel, shown in Figure 2, that is wired as a bit-wise exclusive-or with the other four channels, providing the necessary information to recover from the failure of one of the four channels.

A directory-based cache-coherence protocol is an integral part of the memory controller. Each directory entry is associated with a single 64-byte cache block. The directory entry tells the user the state of the data block, and there are three significant states in the directory. If the data is local, it is either in the DRAM or in the cache of the local processor. The data can be shared, which means multiple CPUs have cached unmodified copies of the block. If there are a small number of shares, the directory can actually record the CPU numbers of the shares, simply recording their IDs. However, if there are a large number of shares, the directory is switched to course vectors, where 20 bits are used to represent regions in the machine. In a 64-processor machine, each bit would then represent a block of four processors. Finally, exclusive means that a processor has an exclusive copy of the block cached; in that case, the address of the processor that cached the block is stored.

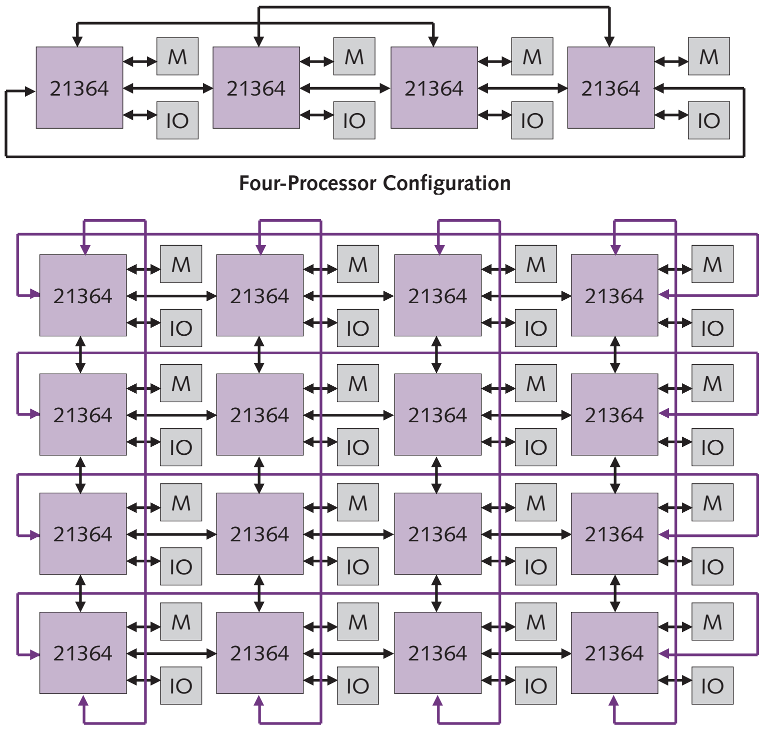
The EV7 physical address bit 36 determines if the sequential addresses will be located on a single processor or interleaved between two processors. This interleaving can be selected on 256MB regions, allowing for some linear storage as well as interleaving. Linear addressing allows for the lowest possible latency when placing the data where the user needs it (for example, replicated code). For data that is accessed by many processors, interleaving spreads the references across two CPUs to reduce the chances that one processor's memory array holds all pertinent data, causing too much traffic to that processor (hot spotting).

The address fields allow for up to 128 nodes, with 32GB per node. The addressing mode must be set at boot time. If fully built out, a 21364 system could support 4TB, a memory size that should be sufficient for some time to come.

The cache-coherency protocol terminology includes the requester, home, shares, and owner. The requester is the node in the network that asks for the data. The home is the location of the directory that represents that data and is also the location of the DRAM storage for it. Potentially, there's an owner of the block, a remote node that contains an exclusive copy of the line in its cache. There can also be shares—remote nodes that contain shared copies of the line that is not modified.

Network Scaling and Reliability Design Issues

The EV7 is a ccNUMA (cache coherent, nonuniform memory architecture) design that has an average memory latency similar to that of many typical server designs, but with superior scalability. The worst-case memory latency for the EV7 is similar to that in many typical processor designs, so can it be treated as a non-NUMA system. Local or nearby references are significantly faster than in other designs, and a NUMA-aware operating system can take advantage of



Four-Processor Configuration

Sixteen-Processor 2D Torus Configuration

Figure 3. The glueless scalability of the EV7 can support from 4 to 64 processors with ease. The physical address protocol is designed to support up to 128 processors. The processors' connections wrap around in a 2D torus shape, providing redundancy and shorter interprocessor paths.

local references to improve system performance.

The heart of the uniqueness of the 21364 is the ease of scalability. The memory scales with the speed and number of processors. The total system bandwidth also scales, using the interprocessor (IP) buses that offer glueless connection between processors, as shown in Figure 3. The 2D topology of interprocessor connections allows for full, closed mesh designs with fault tolerance and minimum hops from the farthest processors. The topology can work in up to 128 processor configurations, and large arrays can be partitioned into multiple smaller ones.

Each processor node is capable of moving 10GB/s of data, and each hop in the network will take an average of 18ns. Because the network is packet based and uses an adaptive routing scheme, it does not guarantee data ordering. The torus topology reduces worst-case node-to-node path delays. The wraparound path can create a deadlock cycle, but that problem is avoided by creating a virtual channel for each path.

The adaptive routing protocol allows the network to detect and avoid hot spots. The routing algorithm employed also must be deadlock free. To accomplish deadlock-free operation, special buffering is allocated at each node. To be dead-

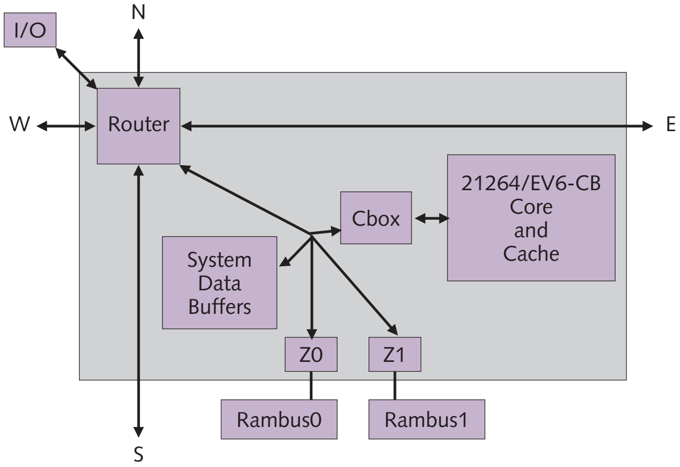


Figure 4. This simplified version of the IP network connections shows the router connections to the local processor, four IP channels, I/O channel, and local memory. Each IP port supports 6.2GB/s bandwidth, and each RDRAM memory array supports 6.2GB/s of data bandwidth.

lock free, packets must always make forward progress on each node hop. The packet can also be subject to deadlock situations because of the two-dimensional square connections. The deadlock-free network has buffers, and the nodes follow a protocol that moves a packet in only one direction (either east-west or north-south) first. This routing scheme is call dimension-order routing. The deadlock-free network is available to all packets, in addition to the adaptive routing network.

The Router, seen in Figure 4, has four connections to neighboring processors, nominally named after the east, west, north, and south compass points. The router also connects to the IO7 port and the local ports. The local ports include the C box port from the local processor core and the two memory controllers: Z0 and Z1. The data pours into an input queue. One problem with input-queue-type networks is that when latency builds up in the input queue, it's hard to eliminate it. To correct that problem, the EV7 has two read ports on every single queue, so it can read two packets out of a queue and route the packets to any of the possible outputs. All the ports are basically identical, except for the I/O port, which has an output FIFO—a hardware feature that allows the frequency of the I/O port to interface to a less exotic technology.

The EV7 uses source-based routing—taking the physical address bits that represent the processor ID and indexing a router table. The router table includes the east-west and north-south coordinates of the destination node and also the route to access the destination. Once the packet is injected into the network, it will route east-west, in the direction that was indicated, until the coordinate matches, and then it will route north-south until the coordinate matches; at that point, it is ejected from the network (at the destination node). Initial hop bits are

319	283	247	211	247	283	319	355
283	247	211	175	211	247	283	319
247	211	175	140	175	211	247	283
211	175	140	75	140	175	211	247
247	211	175	140	175	211	247	283
283	247	211	175	211	247	283	319
319	283	247	211	247	283	319	355
355	319	283	247	283	319	355	391

Figure 5. This chart shows the distribution of memory access latencies for a 64-processor configuration.

used to avoid holes in the network and provide the ability to recover after an error in the network. For example, if the packet would like to route to the east, but the east link is down for some reason, the initial hop bit will send the packet out the north port, where it will then be able to route to the east. This tactic is used to avoid broken links in the torus.

The chart in Figure 5 is a matrix of numbers that shows how latency builds up in the network. The center is the starting point, so in the middle of the matrix, a node has a 75ns local DRAM memory latency. To read the DRAM on a node that's one hop away, there's the DRAM access (75ns), about 30 nanoseconds of overhead getting in and out of the network, and then 18ns per hop of mostly wire and router delay. The total delay is about 140ns to read the memory on a node that's one hop away. As the memory reference moves additional hops away, it builds up another 35–36ns of delay, on average, per hop. The cell in the upper right-hand corner of the chart represents the node that is farthest away, and roughly 390ns are required to read memory out of that node. If there were uniform distributions of accesses across this matrix, the average memory latency would be approximately 250ns. The average latency for the 64-processor configuration compares favorably with the memory latency on an Alpha (EV6) ES45 quad-processor configuration.

Another feature of the network is that it uses asynchronous clocking between two processors; this eliminates the need to distribute a high-frequency, low-skew clock across a physically large machine, which would be either impossible or extremely expensive. Asynchronous clocking between processors removes the need to distribute a low-skew clock within a large system. Reliability is improved by eliminating a centralized clocking scheme, which could be a single point of failure. Like most processors, EV7 supports a large number of "gear box" (core-to-bus) ratios to match the processor frequency to the RDRAM or IP network link speeds.

EV7 provides an additional I/O port with every processor, so that every time a user adds an EV7 to a configuration, another opportunity to increase I/O is available, as Figure 6 shows. A fifth port, called IO7, provides up to 3GB/s of bandwidth to industry-standard buses, such as PCI, PCI-X, and AGP. The IO7

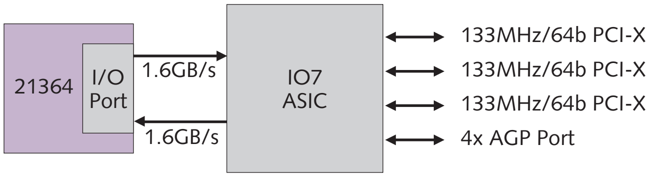


Figure 6. This figure shows a typical I/O ASIC that can be added to an EV7 processor to support PCI-X, AGP, or some other I/O channel. The 3GB/s bandwidth of the IO7 port has sufficient bandwidth to support multiple ports per I/O ASIC.

port is, to a first order, exactly the same as the processor ports; however, it is designed to interface to industry-standard gateways, so the clock speed can divided down.

Conclusions

The EV7 is a stunning system design, with glueless scalability and outstanding memory bandwidth and short access latency. The EV6 processor core provided the stable platform to build this highly advanced memory architecture. The EV7 redefines state-of-the-art and continues a great processor tradition.