The Evolution of the Alpha AXP PC

by

David G. Conroy, Thomas E. Kopec, and Joseph R. Falcone

ABSTRACT

The DECpc AXP 150 personal computer is not only the first in Digital's line of Alpha AXP PC products but also the latest in a line of experimental low-cost systems. This paper traces the evolution of these systems, which began several years ago in Digital's research and advanced development laboratories. The authors reveal some of the reasoning behind the engineering design decisions, point out ideas that worked well, and acknowledge ideas that did not work well and were discarded. Chief among the many lessons learned is that combining Alpha AXP microprocessors and industry-standard system components is within the abilities of any competent digital design engineer.

INTRODUCTION

The DECpc AXP 150 system is Digital's first Alpha AXP personal computer (PC) product that supports the Microsoft Windows NT operating system. This product is the latest member of an evolutionary series of low-cost systems that take advantage of PC components and standards. Work on these systems began several years ago in Digital's research and advanced development laboratories.[1] By tracing the evolution of the Alpha AXP PC from the Beta demonstration system (which pioneered the concept of the Alpha AXP PC) through the Theta system (which incorporated an Extended Industry Standard Architecture [EISA] bus) to the DECpc AXP 150 product, this paper shows how experimental systems solved many problems in anticipation of products.

The Alpha AXP PC design philosophy is summed up by one of the DECpc AXP 150 advertising slogans: It's just a PC, only faster. By being culturally compatible with industry-standard PC systems, Alpha AXP PC systems can exploit the huge infrastructure of low-cost component suppliers supported by the high volumes of the PC marketplace and be cost competitive in that marketplace.

Alpha AXP PC systems typically include little functionality in the base system. Many additional capabilities are provided by option cards via the Industry Standard Architecture (ISA) or EISA bus. Such capabilities can be upgraded easily to keep pace with technological developments. In addition, the increasing fragmentation of the desktop computer market has made it virtually impossible to design a single product that addresses the needs of all market segments. By providing option slots,

systems can be configured to meet a wide variety of customer requirements.

BETA DEMONSTRATION SYSTEM

In the early days of the Alpha AXP program (1990), conventional wisdom said that the DECchip 21064 microprocessor could not be used to build low-cost computer systems--it consumed too much power, was difficult to cool, and was optimized for large, high-performance systems.[2,3] Therefore, Digital was not designing any low-cost systems; all product groups were waiting for a low-cost Alpha AXP CPU, the DECchip 21066 microprocessor (which was announced in September 1993).

In December 1990, several members of the Semiconductor Engineering Advanced Development Group (the same group that began the Alpha AXP architecture development and designed the DECchip 21064 microprocessor) decided to investigate the feasibility of producing low-cost systems. In February 1991, with the help of Digital's Cambridge Research Laboratory, they began designing and building a demonstration system called Beta.

Although the Beta system used the same enclosures, power supplies, expansion bus option cards, and peripherals as industry-standard PCs, true PC compatibility was never a goal. The intent was simply to demonstrate the feasibility of building low-cost systems; standard PC components were used because doing so eliminated some design work.

Hardware Design

Figure 1 is a block diagram of the Beta demonstration system. The hardware design was completely synchronous. The DECchip 21064 CPU operated at 100 megahertz (MHz) and generated the 25-MHz clock that ran most of the logic. The ISA bus interface operated at 8.33 MHz, generated by dividing the 25-MHz clock by three.

[Figure 1 (Block Diagram of the Beta Demonstration System) is not available in ASCII format.]

The Beta system was packaged in an attractive, low-profile, tabletop enclosure. The compact size forced the physical arrangement to be cramped, which made cooling the CPU more difficult. Because the CPU operated at 100 MHz and thus dissipated only about 16 watts, it could be cooled by a large, aluminum heat sink (6.9 centimeters [cm] wide by 8.1 cm long by 2.5 cm deep) and the 20 linear centimeters per second of available airflow.

A three-terminal linear regulator produced the required 3.3-volt power. Cooling this regulator was almost as difficult as cooling the CPU chip but was accomplished with an off-the-shelf heat sink

and the available airflow.

The backup cache and memory systems in previous Alpha AXP designs were absolute-performance driven, that is, designed to meet specific performance requirements. Usually, these requirements were set at the level of performance needed to outperform competitors. Workstation performance requirements tended to emphasize the SPEC benchmark suite.

The Beta system designers took a different approach. They proposed various backup cache and memory systems, estimated the performance and cost of each one, and selected the one that had the best performance per unit cost (beyond a minimum absolute-performance requirement).

Ultimately, the Beta demonstration system used a 128K-byte (128KB) write-back cache that was 128 bits wide and built with the 25-nanosecond (ns), 8K-by-18-bit static random-access memories (SRAMs) designed to be used with the Intel 82385 cache controller. At 25 ns, the SRAMs were slow, but by using 48-milliampere (mA) address drivers, incident wave-switching, and AC parallel termination, it was possible to both read and write the cache in 40 ns (four CPU cycles).[4] Writing the cache SRAMs was simplified by the fact that, in addition to the normal system clock (sysclk1), the CPU chip generated a delayed system clock (sysclk2). This delayed clock had exactly the right timing to be used as a cache SRAM write pulse. Enabling logic, built using a 7.5-ns programmable array logic (PAL) device, ensured that the delayed clock was sent to the cache SRAM only when a write pulse was actually needed. Variations on this design, shown in more detail in Figure 2, have been used in several other Alpha AXP systems, including the EB64 (an evaluation board for the DECchip 21064 microprocessor) and an experimental DECstation 5000 Model 100 daughter card, as well as the Theta and DECpc AXP 150 systems.

[Figure 2 (Details of the Beta System Cache) is not available in ASCII format.]

Main memory used 8 or 16 standard, 9-bit-wide, dynamic random-access memory (DRAM), single in-line memory modules (SIMMs). Although the cache was 128 bits wide, the memory was only 64 bits wide and cycled four times, in page mode, to deliver a 32-byte cache line. Main memory was protected by 32-bit longword parity that was generated and checked by the CPU and copied without interpretation to and from the backup cache. The memory controller was built from PALs clocked at 25 MHz. Registered devices generated all the control signals, so none of the PALs needed to be very fast. The long minimum delay of these slow PALs made clock skew management easy.

The fact that the I/O system needed to be able to perform partial longword direct memory access (DMA) writes complicated the backup cache and memory system because both the cache and memory were

protected by longword parity. After rejecting the options of (1) storing byte parity in memory with no parity in the cache and (2) performing read-modify-write cycles on partial longword writes, the designers selected the Intel 82380 multifunction peripheral chip. This chip can perform longword assembly and disassembly for narrow DMA peripherals. Partial longword DMA writes never worked properly, however, because this chip did not function exactly as described in its data sheet. The chip was not used in future designs.

The I/O system was built around an I/O bus that imitated the signaling of an Intel386 DX CPU chip (since the Intel 82380 device was compatible with the Intel386 DX chip). Logic between the memory bus and the I/O bus translated DECchip 21064 cycles into cycles that mimicked the style of Intel386 DX cycles. Figure 3 illustrates the translation.

[Figure 3 (Beta System I/O Addressing) is not available in ASCII format.]

Complete imitation of the signaling of an Intel CPU requires the generation of memory-space reads and writes with byte granularity, I/O-space reads and writes with byte granularity, and several special-purpose cycles. However, only a small subset of these cycles (exactly eight) actually needed to be generated in the Beta system. Thus, DECchip 21064 address bits [32..30] were used to encode the details of the cycle; a single PAL expanded these 3 bits into the Intel cycle--type signals. An alternative scheme that stored unencoded Intel cycle--type signals in a control register was rejected because the control register would have to be saved and restored in interrupt routines.

The external interface of the DECchip 21064 microprocessor is strongly biased toward the reading and writing of 32-byte cache lines; external logic has difficulty gaining access to DECchip 21064 address bits [04..03] and cannot access address bit [02]. Therefore, the Beta system used DECchip 21064 address bits [29..27] to supply Intel address bits [04..02]. This odd positioning, which was chosen because it saved a few parts in the address path, seemed harmless; the CPU was so much faster than the I/O that repositioning the low-order address bits did not reduce Beta system performance. It was a bad trade-off, however, because the odd addressing scheme made writing low-level software for devices containing buffers (e.g., a PC-style video adapter) painful and error prone.

The ISA bus interface connected to the I/O bus. Since all DMA control was inside the Intel 82380 chip, the ISA interface functioned as a simple slave that translated Intel386 DX cycles into ISA cycles.[5] Once again, the Beta system used address bits to encode the details of the cycles. Many programmable options were included because the sensitivity of ISA cards to bus timing was unknown. In fact, the ISA cards that were used could tolerate

wide variations in bus timing and never required the unusual bus cycle options.

The DMA controllers in the Intel 82380 handled ISA-bus DMA transfers and generated the appropriate read and write requests. The cache and memory system responded with the appropriate read and write cycles. The refresh controller in the Intel 82380 handled memory refresh and generated the appropriate refresh requests. The memory system responded with column address strobe--before--row address strobe (CAS-before-RAS) refresh cycles. In both cases, the Beta system used the CPU's holdReq/holdAck protocol (in which the CPU stops and tristates most of its pins) to avoid conflicts between the CPU and DMA or refresh on the cache and/or memory system.

During DMA read cycles, the backup cache read data (in anticipation) and performed the tag compare operation in parallel with the RAS-to-CAS delay of the DRAMs. If the reference was a miss, then CAS was asserted and the data came from the DRAMs. If the reference was a hit, then the buffers between the cache data bus and the memory data bus were enabled and the data came from the cache; the cycle on the DRAMs became a RAS-only refresh cycle.

During DMA write cycles, the data was written to the DRAMs. The cache performed the tag compare operation in parallel with the RAS-to-CAS delay of the DRAMs. If the reference was a hit, the data was written into the backup cache as well (without changing the state of the dirty bit), and the appropriate internal CPU cache line was invalidated.

The local peripheral interface also connected to the I/O bus. This interface included (1) an 82C106 PC Combo I/O chip from VLSI Technology, Inc., with a keyboard interface, a mouse interface, two serial lines, a parallel printer port, a time-of-year clock, and a small SRAM with battery backup; (2) a control register that contained a few random control bits; and (3) a 64KB erasable programmable read-only memory (EPROM) that contained the boot and console code. Code in the DECchip 21064 serial read-only memory (ROM) copied the EPROM to main memory for execution, thus eliminating the need for hardware that could read 32 bytes from the byte-wide EPROM and assemble the data into a cache line.

The hardware was completed in a very short time--about eight weeks from the start of the design to releasing the databases to the printed circuit board and assembly houses. One person did the logic design, the logic simulation, and the timing verification. A second person designed the physical layout and solved all the mechanical and cooling problems. A set of quick-turnaround, computer-aided design (CAD) tools developed by Digital's Western Research Laboratory in Palo Alto, California, allowed concurrent design verification (with modifications) and physical design.

Software Design

The Beta system was debugged using a simple console program, which drove an ordinary terminal plugged into one of the serial ports and allowed the designers to peek and poke at memory and I/O devices. Both the hardware and the simple console program were debugged in one day, and most of that day was spent looking for a single software bug in the CPU chip initialization code.

The designers augmented the simple console program to perform more extensive diagnostics, drive a standard Intel PC keyboard and ISA bus display, load programs by name from a file system on a small computer systems interface (SCSI) disk using a standard ISA bus disk controller, and load programs by name via BOOTP over the Ethernet using a standard ISA bus network controller. All this code fit into the 64KB EPROM, albeit compressed.

Eventually, the designers built a fairly complete version of the UNIX operating system for the Beta system, starting from the port of the Berkeley Software Development (BSD)-based ULTRIX system built for the original Alpha AXP Development Unit.[6] This UNIX system included a port of X11R5, believed to be the first 64-bit X11 server ever built.[7]

Performance

The CPU performance of the Beta system was estimated to be 70 for integer (SPEC dhrystone, eqntott, espresso) and 65 for floating point (SPEC fpppp, matrix300, tomcatv). A system with a 128-bit-wide cache and a 128-bit-wide memory would have had a performance of 70 integer and 75 floating point. A system with a 64-bit-wide cache and a 64-bit-wide memory would have had a performance of 65 integer and 55 floating point. All these estimates were obtained from a trace-driven performance model of the DECchip 21064 CPU, the backup cache, and the memory system.

The ISA bus limited the I/O performance of the Beta system to approximately 4 megabytes per second (MB/s). Thus, Beta is the most unbalanced Alpha AXP system ever designed.

Outcome

Digital ultimately built 35 Beta demonstration systems, which were used in many presentations, including a stockholders' meeting and a private demonstration for Bill Gates at Microsoft Corporation. The Beta machines were proof that the DECchip 21064 microprocessor could be used to build low-end systems and prompted a number of low-end system projects to be started. Beta systems were used in the early development of the Alpha AXP version of the Windows NT operating system.

Although many aspects of the Beta design worked well, dealing

with the PC option cards did not go smoothly. The designers knew that there was not much low-level programming documentation available and took care to select option cards based on very large-scale integration (VLSI) chips for which good documentation could be obtained. They were often astounded, however, at how difficult these devices were to program and how often they did not work exactly as described. By the end of the project, it was clear to the designers that they could use PC components to build interesting systems. It was equally evident that PC option cards could be difficult to use, since little or none of the low-level software provided by the option cards' supplier (basic I/O system [BIOS] ROM code, MS-DOS device drivers, or user code that manipulated the option card directly) was usable. Video graphics array (VGA) cards proved to be particularly troublesome, since they tended to conform to the VGA programming standard only after code in their BIOS ROM performed a module-specific initialization sequence.

THETA SYSTEM

The next step in the evolution of the Alpha AXP PC took place in the winter and spring of 1992 with the design of the Theta system. This machine was based on the Beta design but used the EISA bus, driven by the Intel 82350DT chip set.[8] Figure 4 is the block diagram of the Theta system. In addition to replacing the ISA I/O system with an EISA I/O system, the Theta design stored its firmware in flash EPROM and used an I/O bus that imitated the signaling of an Intel486 DX CPU chip.

[Figure 4 (Block Diagram of the Theta System) is not available in ASCII format.]

The Theta designers made several mistakes because they lacked a complete understanding of the nuances of industry-standard PC architecture. For example, the EISA bus uses special memory read and write signals when accessing the first 1M byte (1MB) of memory. The designers were not aware that one chip in the EISA chip set was checking address ranges for another chip in the set. Consequently, they did not realize that the address map they had chosen for their system would cause the range check to fail. This failure would block the generation of the special memory read and write signals and thus make the buffer memory in a network card inaccessible. Further, it is not clear that the Theta designers could have determined that this would occur from the chip set data sheet alone.

Digital built few Theta systems, and little system software ever ran on these systems. The Theta project, however, provided an environment for learning how to use the Intel 82350DT EISA chip set in an Alpha AXP system. The project team developed an extensive set of exercisers for popular EISA cards, which were very useful in the development of future systems.

DECpc AXP 150 PRODUCT

The charter for the design and manufacture of the DECpc AXP 150 product, code-named Jensen, belonged to the Entry Level Solutions Business in Ayr, Scotland, which had previously designed the successful line of MicroVAX 3100 systems. However, Digital felt that it was important to incorporate the knowledge gained through the Beta and Theta system development efforts in Massachusetts into this new product. The "tiger team" that was assembled in March 1992 to define and design the product gave rise to a unique partnership among development, qualification, and manufacturing groups in Hudson and Maynard, Massachusetts, and in Ayr, Scotland. The fact that the system was designed by a tiger team faced with a short, fixed schedule had a powerful effect on the architecture. The amount of time available for design work was determined by calculating backward from the time when the system was needed. The system implemented the best architecture that could be designed in the 12-week period available for design.

The schedule was extremely tight, as the following list of milestones indicates:

Date (1992)	Milestone
Mar 26	Assemble tiger team
May 29	Complete preliminary schematics
Jun 1	Begin layout
Jun 29	Complete schematics, begin parts sourcing
Jul 7	Begin prototype build
Jul 28	Complete prototype build, begin debug
Aug 5	Ship first prototype (actual date)
Aug 10	Ship first prototype (scheduled date)

The Microsoft Windows NT operating system was selected as the design center for the product. This decision made designing to an aggressive schedule even more difficult. The port of the Windows NT system to the Alpha AXP architecture was beginning at about the same time, and there were still many unknowns. This problem was solved by quickly establishing a close working relationship with key technical contributors in the Windows NT group, resulting in a design team that spanned eight time zones.

The tiger team's original intention was to base the product on the Theta design. A careful design review, however, showed that the Theta design was not well suited to high-volume manufacture. In fact, the Theta design did not implement some of the more esoteric aspects of the EISA standard, such as the asynchronous timing of translated ISA direct master cycles, and could not be easily modified to do so. Therefore, a new I/O system design was needed. This requirement caused a schedule problem for the Windows NT group. To solve the problem, the tiger team quickly defined the software-visible characteristics of the system,

designed a special version of the Theta machine (which implemented this definition) and built 10 machines. The design team delivered these systems, called Theta-II, to the Windows NT group in early June 1992. The group used Theta-II systems until actual DECpc AXP 150 machines arrived in August 1992.

The DECpc AXP 150 product uses the same enclosure as the DECpc 433ST Intel-based PC. Since this enclosure was intended to hold a multiboard system, the designers' original intention was to build a multiboard system as well. To save time, however, the designers put the entire CPU, cache, memory, and core I/O system onto the mother board and placed all the complicated I/O devices (disk, network, display) on EISA option cards. Placing these devices on option cards had an additional advantage. Many system-level decisions (such as which video controller to use) were removed from the critical path and were, in fact, changed several times as the project evolved.

Because the delivery schedule was tight, the designers needed to ensure that the first-pass boards were nearly production quality (since many systems would be built using first-pass boards). The designers had electromagnetic compatibility (EMC), thermal, assembly, and test experts critique the design and incorporated as many of their suggestions as possible into the first-pass boards. Dealing with these design issues early in the schedule delayed the release of the first system boards but saved time overall because the usual flurry of changes required by regulatory testing and manufacturing was avoided.

Hardware Design

The DECpc AXP 150 hardware design is completely synchronous. The DECchip 21064 CPU chip operates at 150 MHz and generates the 25-MHz clock that runs most of the logic. Sections of the I/O system run at 8.33 MHz, generated by the EISA chip set. The original plan included a 256KB backup cache built with the 16K-deep version of the SRAMs used in the Beta and Theta systems. The designers discovered, however, that the 32K-by-9-bit SRAMs used to build caches for Intel486 DX systems were so inexpensive that building a 512KB backup cache was less expensive than building a 256KB cache. The designers reused the same basic cache design used in the Beta and Theta systems, although two copies of the cache address are needed because there are twice as many SRAMs. This design, with 17-ns SRAMs, allows the CPU to read the cache in 32 ns and write the cache in 40 ns. Figure 5 is a block diagram of the DECps AXP 150 product.

[Figure 5 (Block Diagram of the DECpc AXP 150 Product) is not available in ASCII format.]

The memory system was redesigned to be 128 bits wide (to increase performance), to use 36-bit-wide SIMMs (to save space), to handle both 1M- and 4M-deep SIMMs, and to handle both the single- and

double-banked versions of the SIMMs. Because system software strongly prefers to have contiguous memory, the design includes some hardware so that configuration software can arrange the available memory into a dense block starting at location 0.

Main memory is protected by longword parity, as in the Beta and Theta systems. The memory controller transforms partial longword DMA writes into read-modify-write cycles. The latching and merging functions are performed in the 74FCT652 transceivers situated between the memory bus and the I/O bus. The registers in the transceivers make it possible to check the parity of the old data at the same time as the new data is written, making the read-modify-write cycle faster.

The Windows NT operating system requires that a block of physically contiguous addresses on an I/O bus need only be mapped into a single block of virtually contiguous addresses in the virtual address space. Thus, schemes that place low-order Intel address bits and/or byte enables in high-order Alpha AXP address bits (like the ones used on the Beta and Theta systems) are unacceptable. The DECpc AXP 150 product, therefore, uses a new scheme, developed jointly by the hardware and software designers. This scheme places the low-order Intel address bits and the width of the cycle in low-order Alpha AXP address bits. Figure 6 illustrates the translation.

[Figure 6 (DECpc AXP 150 I/O Addressing) is not available in ASCII format.]

The DECchip 21066/21068 microprocessors and the DECchip 21070 chip set use a similar mapping but make two small improvements. First, they shift the Alpha AXP address 2 bits to the right, which makes the Intel addressing window 128MB in size. Second, they add a special check to make accessing the lowest 1MB of the Intel memory space more efficient, since a number of important peripherals have hardwired address assignments between 640KB and 896KB.

The local I/O systems in the DECpc AXP 150 and Theta systems are similar. The only exception is that the Alpha AXP 150 system has the additional flash EPROM needed to support the console interfaces for the Windows NT operating system (thus conforming to the Advanced RISC Computing [ARC] specification) and the DEC OSF/1 AXP and OpenVMS AXP operating systems (thus conforming to the Alpha AXP console architecture). The designers considered rearranging the addressing and placing the industry-standard peripherals (those in the VLSI Technology Combo I/O chip) at their usual places in the EISA address space. This plan was rejected to ensure that the console firmware could communicate with the serial lines and be used to debug the EISA I/O system.

The EISA I/O system was tricky to design because the Intel 82350DT EISA chip set was intended to be used in a system with a very different architecture. After careful analysis of the

problem, only three truly difficult issues were evident.

1. Clock skew. The EISA bus clock is generated by synchronous logic (inside the Intel 82350DT chip set) running on the system clock; however, the delays between the EISA bus clock and bus control signals, combined with the delays in the EISA bus clock generator itself, make it impossible to use EISA bus control signals as inputs to synchronous logic running on the system clock. This problem was solved by implementing the EISA control logic as two interlocked state machines. The first machine runs on the EISA bus clock, and the second machine runs on the system clock sampling the outputs of the first one. Intel uses a similar scheme in their 82350DT example designs.

2. Flow control on partial longword writes. In the DECpc AXP 150 system, the backup cache and main memory are protected by longword parity. This complicates DMA writes of less than a longword. Such writes need to be implemented as read-modify-write sequences, and the timing of the EISA byte-enable signals (which tell the DMA logic if a read-modify-write cycle is needed) make it impossible to extend the cycle using the normal EISA flow control mechanism (EXRDY). The designers solved this problem by stretching the EISA bus clock when needed; the Intel EISA chip set has logic (HSTRETCH) for doing this. The short bus clock stretch does not affect any option that conforms to the EISA bus specification.

3. ISA direct masters. The Intel 82350DT chip set contains logic that translates ISA direct master cycles into ordinary EISA cycles. These EISA cycles, however, have somewhat unusual timing; they contain events that are not synchronized to the EISA bus clock. The most difficult part of dealing with this timing was determining that these events could actually happen. Making it work required simply that some key transceiver control signals be generated combinatorially.

The arbiter in the Intel 82350DT chip set responds to bus requests (e.g., EISA bus masters or the DMA controllers inside the 82350DT chip set), stops the CPU using the DECchip 21064 CPU's holdReq/holdAck protocol, and takes control of the cache and memory system. When the CPU is in hold, the memory controller watches for EISA memory cycles aimed at the low 256MB of memory and performs the appropriate read, write, read-modify-write, or refresh cycles. The cache cycles at the same time as memory, reading and writing as required. The timing is tighter than in previous machines, but making it work required only a careful placement of the critical parts. No attempt was made to allow the processor to run during DMA, partially to keep the design simple (EISA bursts are not required to be sequential) and partially to avoid risk. Intel systems built with the 82350DT chip set stop the CPU during DMA. Introducing parallelism could have revealed a

lingering bug in the chip set that no other system had encountered.

Software Design

The designers used a custom version of the Theta-II firmware to debug the first DECpc AXP 150 systems. The systems were operational in less than half a day after they arrived from the assembly house. Almost immediately, the debugging firmware was replaced by the first version of the production firmware, developed jointly by the DECpc AXP 150 and the Windows NT firmware teams.

The first group to receive shipment of the systems was the Windows NT group, who had been using the Theta-II machines. Their software worked instantly; they only had to fix a single bug that had been concealed by a bug in the Theta-II systems. The group used both systems until enough DECpc AXP 150 systems were available, at which point the Theta-II systems were decommissioned.

Next, the OpenVMS and DEC OSF/1 groups took delivery of systems. Making each of these two systems operational on a DECpc AXP 150 machine was a slow process because this was the first time that many of the developers had dealt with PC hardware. Once operational, however, the systems stabilized rapidly.

The designers discovered few hardware problems as the operating system work progressed. Some hardware problems were discovered in the EISA option cards, when software attempted to use the cards in a manner unlike that of the MS-DOS operating system. These problems were tracked down with the help of option card vendors.

Performance

The CPU performance of the system met project expectations. The original design concept targeted 100 SPECmark89, and simulations indicated that the 150-MHz system with 512KB cache would achieve this rating. As with any new system, performance tuning is an important part of the development activity. Table 1 shows the performance results as of January 1994 for the DECpc AXP 150 system and for some of its competitors, all running industry-standard benchmarks under the Microsoft Windows NT operating system. Descriptions of these benchmarks, as well as of the hardware and software configurations used to make the measurements, can be found in the Alpha AXP Personal Computer Performance Brief--Windows NT.[9]

Table 1 DECpc AXP 150 Benchmark Performance under the Windows NT Operating System

Benchmark	Metric	Digital DECpc AXP 150	Gateway 2000 P5 60 MHz	MIPS Computer Systems Magnum 75/150 MHz
Byte, numeric sort	Sorts/s	36.0	11.4	33.6
Byte, string sort	Bytes/s	136M	40.4M	123M
Byte, bit fields	Ops/s	7.6M	2.2M	7.8M
Byte, emulated float	FLOPS	2977	974	4597
Byte, simple FPU	Bytes/s	5.4M	2.7M	3.9M
Byte, transcendentals	Coeffs/s	867	334	538
Dhrystone V1.1	DMIPS	175.1	68.9	68.7
Dhrystone V2.1	DMIPS	161.5	61.6	59.8
CLinpack 100 x 100	MFLOPS	22.2	8.0	7.4 (double prec.)
CWhetstone	KWIPS	95.5	34.0	36.7 (double prec.)

Comparing the performance of the DECpc AXP 150 system to DEC 3000 AXP workstation performance is difficult. Most performance measurements have been made under the Windows NT operating system, using the compilers and libraries appropriate for that system, and the Windows NT operating system does not run on DEC 3000 AXP systems. As shown in Table 2, the SPEC benchmark suite, running under the DEC OSF/1 AXP operating system, demonstrates that the DECpc AXP 150 system performs like a midrange DEC 3000 AXP system. These results are not surprising because the only real difference between the two systems is the DECpc AXP 150 machine's slightly slower memory system.[10]

Table 2 DECpc AXP 150 and DEC 3000 AXP Benchmark Performance under the DEC OSF/1 AXP Operating System

System	SPECint92	SPECfp92
DEC 3000 AXP Model 600	114	162
DEC 3000 AXP Model 500	84	128
DEC 3000 AXP Model 400	75	112
DECpc AXP 150	77	110 (V1.3A-4)
DEC 3000 AXP Model 300	66	92

While adequate for many applications, the I/O performance of the DECpc AXP 150 system is limited not only by the EISA bus (which has a peak bandwidth of 33 MB/s) but also by the Intel 82350DT EISA chip set. The chip set is designed to be used with Intel microprocessors, and considerable I/O performance is lost reconciling Intel control signals with DECchip 21064 control signals. For this reason, the peak bandwidth of the EISA bus in the DECpc AXP 150 system is only 25 MB/s. The chip set also wastes EISA bus bandwidth while performing internal operations.

One network adapter could transfer only 16 MB/s because of excessive EISA bus request latency introduced by the 82350DT chip set.

Outcome

The DECpc AXP 150 product was first shown in public on October 28, 1992, by Bill Gates at Windows on Wall Street, a presentation of the Windows NT operating system for more than 1,000 Wall Street analysts. The machine was subsequently shown at the Alpha AXP introduction and in both Digital's and Microsoft's booths at the 1992 Fall COMDEX conference in Las Vegas. There, the product was a finalist for "best system of show," missing the title by only one vote despite limited advertising. Digital formally announced the system at the 1993 Spring COMDEX conference, coincident with Microsoft's rollout of the Windows NT operating system. The DECpc AXP 150 product continues to receive good reviews and awards from the PC media.

Lessons Learned

Engineers involved in the Beta, Theta, and DECpc AXP 150 projects learned many lessons during the evolution of the these systems. Chief among these lessons are the following:

1. Combining Digital's Alpha AXP microprocessors and industry-standard system components was fairly straightforward and certainly within the abilities of any competent digital design engineer.

2. Even though many engineers shy away from evolutionary design because it lacks the glamour of doing things from scratch, evolution is a fine methodology. This is especially true if the cost of each step along the way is small. Each new Alpha AXP PC eliminated the obvious shortcomings of the previous designs, yet the risk of the system not working was small because most of the design was copied from the predecessor.

3. The interchip and system buses designed with Intel CPUs in mind (e.g., ISA, EISA, and Peripheral Component Interconnect [PCI]) can be used in non-Intel systems. Designing the correct interfaces, however, may require multiple iterations. Digital engineers made three iterations before arriving at what seems to be a good mapping from Alpha AXP program I/O cycles to Intel program I/O cycles.

4. Sometimes, the best way to show that something is possible is to build a demonstration unit. When the Beta system was designed, few believed that low-end Alpha AXP systems could be built. Nonbelievers found it difficult

to defend their position in the presence of a working computer.

5. Clear goals can make development proceed faster, since alternatives that run counter to the goals need little analysis--they can simply be rejected. The time-to-market goal of the DECpc AXP 150 project was an extreme example. Designers needed only to consider alternatives that allowed meeting the tight schedule.

6. Qualification, test, and assembly issues must be addressed as part of the design rather than as annoying details to be addressed later. Addressing these issues during the design phase may delay the delivery of the prototype, but doing so helps to ensure that the prototype is of high quality and to avoid the risk of longer delays later in the project.

7. A good CAD system is a valuable asset. Because our CAD system was designed for rapid turnaround, it was possible to make significant changes to designs as new data appeared or when urgent needs arose. The Theta-II board design was released five days after the designers concluded that it was needed.

ACKNOWLEDGMENTS

The contributions of many people made this effort successful. Although not complete, the following list gives credit to those who were the most deeply involved. Rich Witek, Dave Conroy, Tom Levergood, Larry Binder, Larry Stewart, Win Treese, and Joe Notarangelo designed and implemented the Beta hardware and software. Ralph Ware, Lee Ridlon, and Carey McMaster designed and implemented the Theta hardware and software. Joe Falcone first realized that there was a product lurking in the advanced development, transformed it into a real project, named it after a legendary sports car, and did the early system definition work. The Jensen hardware tiger team was led by Tom Kopec and included Jay Osmany, Fraser Murphy, Tom Mann, Dave Conroy, Lee Ridlon, and Carey McMaster. Colin Brench, Sharad Shah, Roy Batchelder, Jim Pazik, and Mike Allen provided consulting services early in the design.

NOTE AND REFERENCES

1. The DEC 2000 AXP Model 300 is the same product using the DEC OSF/1 AXP and the OpenVMS AXP operating systems.

2. R. Sites, ed., Alpha Architecture Reference Manual (Burlington, MA: Digital Press, Order No. EY-L520E-DP, 1992).

3. DECchip 21064 Microprocessor: Hardware Reference Manual (Maynard, MA: Digital Equipment Corporation, Order No. EC-N0079-72, 1992). This document and many others associated with Digital's Alpha AXP microprocessors are available on the Internet and can be obtained via anonymous FTP from gatekeeper.pa.dec.com, in directory pub/DEC/Alpha.

4. H. Johnson and M. Graham, Chapter 6 in High Speed Digital Design: A Handbook of Black Magic (Englewood Cliffs, NJ: Prentice-Hall, 1993).

5. E. Solari, ISA and EISA, Theory and Operation (San Diego, CA: Annabooks, 1992).

6. C. Thacker, D. Conroy, and L. Stewart, "The Alpha Demonstration Unit: A High-Performance Multiprocessor," Communications of the ACM, vol. 36, no. 2 (February 1993): 55-67.

7. J. Gettys, personal communication, 1991.

8. 82350DT EISA Chip Set (Santa Clara, CA: Intel Corporation, Order No. 290377-003, 1992).

9. Alpha AXP Personal Computer Performance Brief--Windows NT, 2d ed. (Maynard, MA: Digital Equipment Corporation, Order No. EC-N2685-10, January 1994).

10. Alpha AXP Workstation Family Performance Brief--DEC OSF/1 AXP, 4th ed. (Maynard, MA: Digital Equipment Corporation, Order No. EC-N2922-10, February 1994).

TRADEMARKS

The following are trademarks of Digital Equipment Corporation: Alpha AXP, AXP, DEC, DECchip, DECpc, DECstation, Digital, MicroVAX, OpenVMS, and ULTRIX.

BSD is a trademark of the University of California at Berkeley.

Gateway 2000 is a registered trademark of Gateway 2000, Inc.

Intel, Intel386, and Intel486 are trademarks of Intel Corporation.

MIPS is a trademark of MIPS Computer Systems, Inc.

OSF/1 is a registered trademark of Open Software Foundation, Inc.

UNIX is a registered trademark of Unix System Laboratories, Inc., a wholly owned subsidiary of Novell, Inc.

Microsoft, MS-DOS, and Windows NT are registered trademarks of Microsoft Corporation.

PAL is a registered trademark of Advanced Micro Devices, Inc.

SPEC, SPECfp, SPECint, and SPECmark are registered trademarks of the Standard Performance Evaluation Council.

BIOGRAPHIES

David G. Conroy Dave Conroy is a senior consulting engineer at Digital's System Research Center in Palo Alto, California. He received a B.A.Sc. degree in electrical engineering from the University of Waterloo, Canada, and moved to the United States in 1980 as a cofounder of the Mark Williams Company. He joined Digital in 1983 to work on the DECtalk speech synthesis system and became a member of the Semiconductor Engineering Group in 1987 to work on system-level aspects of RISC microprocessors, including the DECchip 21064 CPU. Dave is now investigating the design of high-performance memory systems.

Joseph R. Falcone Joseph Falcone is a hardware engineering manager in the Technical Original Equipment Manufacturer Group, which develops Alpha AXP board products for the OEM market. Prior to this, he developed the system architecture and business strategy for the DECpc AXP 150 personal computer. Joe came to Digital from Hewlett-Packard Laboratories in 1983 to work in Corporate Research & Architecture on distributed systems research and development. This work resulted in the DECnet DFS File Service product. He received an A.B.C.S. (1980) from the University of California, Berkeley, and an M.S.E.E. (1983) from Stanford University.

Thomas E. Kopec Principal engineer Tom Kopec is a member of the Assistive Technology Group working on compact speech-synthesis and text-to-speech systems. Previously, he was a member of the Entry Systems Business Group that designed the VAX 4000 Models 200 through 600 and the MicroVAX 3500 and 3800 systems. Tom received a B.S.E.C.E. (1980, with honors) in microwave engineering and signal processing and an M.S.E.C.E. (1985) in image processing and computer graphics, both from the University of Massachusetts at Amherst.