A Technical Overview of the Tandem TXP Processor

Robert Horst
Sandy Metz

A Technical Overview of The Tandem TXP Processor

Robert Horst and Sandy Metz

April 1984

Tandem Technical Report 84.1

A Technical Overview of The Tandem TXP Processor

Robert Horst and Sandy Metz

ABSTRACT

The TXP processor was introduced in 1983 as the the fastest member of the compatible Tandem NonStop Processor family. The TXP obtains its performance through parallel data paths, three-stage pipelining, 64Kbyte cache, hardware support for 32 bit virtual addressing, an 83 nanosecond microcycle, and a large control store. This paper sketches these features and describes how a hardware performance monitor was used to evaluate and optimize the design. Various measures of cpu performance are discussed.

New system manages hundreds of transactions per second

Parallel data paths, pipelining, large cache memory, and 32-bit hardware combine to increase transaction system performance

by Robert Horst and Sandra Metz, *Tandem Computers Inc., Cupertino, Calif.*

☐ Computer systems for on-line transaction processing have a unique set of requirements that pose an enormous challenge to designers. These systems have to be fault-tolerant, expandable through the addition of modules, and able to process multiple transactions at a reasonable cost, while maintaining data integrity. The coming generation of transaction-processing systems must also address a fast-growing need for very high-volume applications that require the processing of more and more transactions per second.

Designed to handle very high-volume transaction processing, the 32-bit NonStop TXP system reaches two to three times the speed of the NonStop II system it supercedes, while retaining complete software compatibility. Without reprogramming, a TXP system can grow from a single system containing from 2 to 16 processors, to a local cluster of up to 224 processors linked with fiber-optic cables, to a worldwide network of up to 4,080 processors.

Many of the problems in designing the TXP processor had already been solved in the NonStop II processor and system design. The NonStop II extended the instruction set of the NonStop 1+ system to handle 32-bit addressing but did not efficiently support that addressing in hardware. The existing 5-megabyte input/output bus and 26-megabyte Dynabus, Tandem's proprietary bus structure, had more than enough bandwidth to handle a processor with two to three times the performance. The existing packaging had an extra central-processing-unit card slot for future enhancements, and the existing power supplies could be reconfigured to

handle a higher-power CPU.

The main problems involved designing a new microarchitecture that would efficiently support the 32-bit instructions at much higher speeds, with only 33% more printed-circuit-board real estate and an existing backplane. This involved eliminating some features that were not critical to performance and finding creative ways to save area on the pc board, including clever uses of programmable array logic and an unusual multilevel control-store scheme.

Since the new TXP processor was to be object-code-compatible with the Nonstop II system yet have a significant price-performance advantage, it was expected that soon after announcement much of the company's produc-
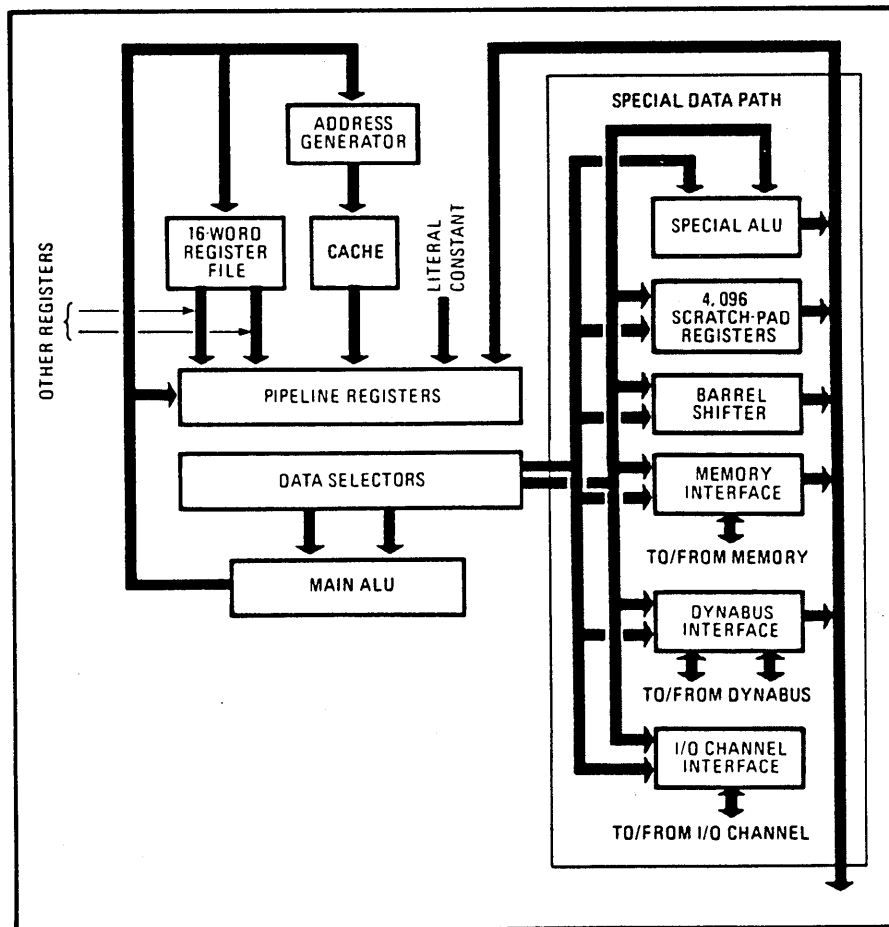

1. Parallel data paths. The NonStop TXP's architecture lets the main arithmetic and logic unit operate in parallel with either a special ALU, one of 4,096 scratch-pad registers, a barrel shifter, the memory interface, the Dynabus interface, or the input/output channel.

TABLE 1: COMPARE-BYTE INSTRUCTIONS (INNER LOOP)		
Clock cycle	**NonStop TXP**		**Traditional architecture**
	Main ALU	**Special ALU**	
1	extract byte 1	extract byte 2	extract byte 1
2	compare bytes	—	extract byte 2
3	(repeat)	(repeat)	compare bytes
4	—	—	(repeat)

TABLE 2: DYNABUS-RECEIVE MICROCODE INSTRUCTIONS (INNER LOOP)		
Clock cycle	**NonStop TXP**		**Traditional architecture**
	Main ALU	**Special data path**	
1	compute checksum on previous word	read next word from bus queue	compute checksum on previous word
2	address next memory location	write data to cache and memory	read next word from bus queue, increment address
3	(repeat)	(repeat)	write data to cache and memory
4	—	—	(repeat)

tion would have to shift quickly from the NonStop II system to the TXP system. This required that efficient board-testing procedures be in place by the time the product was announced and precluded the use of traditional functional board testers, which need months of programming after the design is finished. Instead, scan logic was designed into the processor and a scan-based board-test system using pseudorandom test vectors was developed.

Performance improvements

The performance improvements in the NonStop TXP system were attained through a combination of advances in architecture and technology. The NonStop TXP architecture uses dual 16-bit data paths, three levels of macro-instruction pipelining, 64-bit parallel access from memory, and a large cache (64 kilobytes per processor). Additional performance gains were obtained by increas-

ing the hardware support for 32-bit memory addressing.

The machine's technology includes 25-nanosecond programmable array logic, 45-ns 16-K static random-access-memory chips, and Fairchild Advanced Schottky Technology (FAST) logic. With these high-speed components plus a reduction in the number of logic levels in each path, a 12-megahertz (83.3 ns per microinstruction) clock rate could be used.

The system's dual-data-path arrangement increases performance through added parallelism (Fig. 1). A main-arithmetic-and-logic-unit operation can be performed in parallel with another operation done by one of several special modules. Among them are a second ALU that performs both multiplications and divisions, a barrel shifter, an array of 4,096 scratchpad registers, an interval timer, and an interrupt controller. Other modules provide interfaces among the CPU and the interprocessor bus system, I/O channel, main memory, and a diagnostic processor.

The selection of operands for the main ALU and the special modules is done in two stages. In the first, data is accessed from the dual-ported register file or external registers and placed into two of the six registers. During the same cycle, the other four pipeline registers are loaded with cache data, a literal constant, the results of the previous ALU operation, and the result of the previous special-module operation.

In the next stage, one of the six pipeline registers is selected for each of the main ALU inputs and one for each special-module operand. Executing the register selection in two stages, so that the registers can be two-rather than four-ported, greatly reduces the cost of multiplexers and control storage, while the flexibility in choosing the required operands is unimpaired.

Some examples of the way microcode uses the parallel data paths are shown in Tables 1 and 2. The first example shows the inner loop of the compare-bytes instruction. Each of the dual ALUs in the TXP system extracts one byte; then the extracted bytes are compared. This operation takes two clock cycles on the TXP system
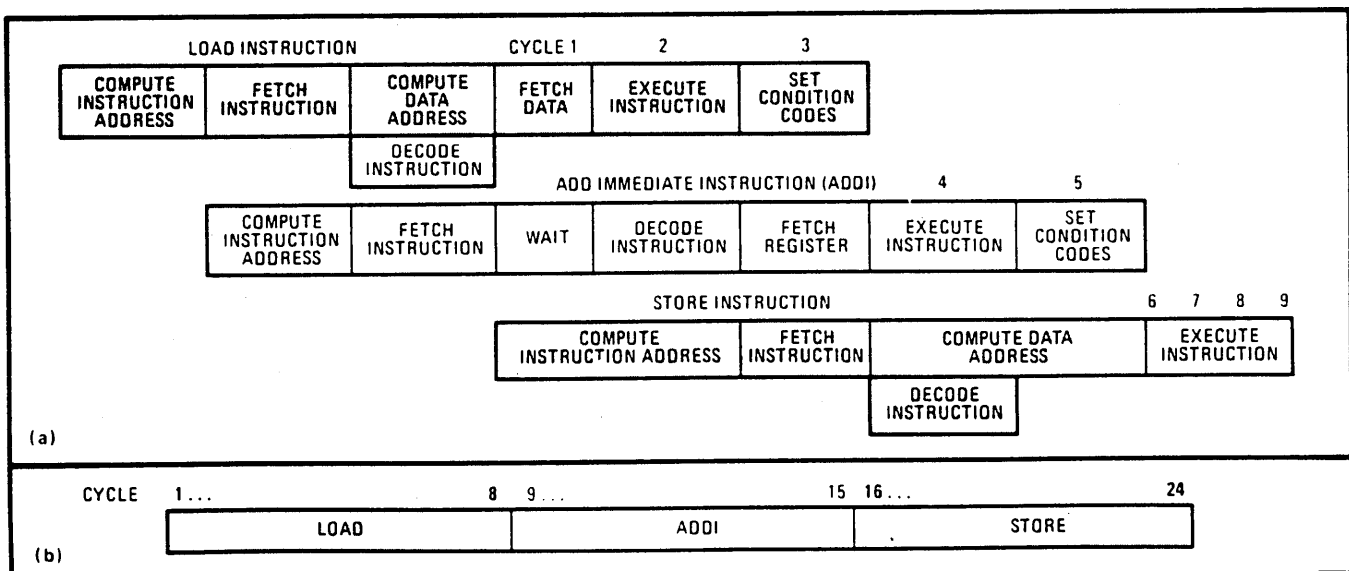
2. Pipelined. The instruction pipeline of the NonStop TXP system allows parts of several instructions to be processed simultaneously (a)—nine cycles are required to execute three typical instructions. Without pipelining (b), 24 clock cycles would be required.

Hardware-performance monitor helps optimize design

While new architectural concepts were being developed for the TXP system, a hardware-performance monitor was built to record measurements of the software-compatible Non-Stop II processor. Xplor consists of two large Wire-Wrap boards plus a small board to interface to the processor under test. It has approximately 800 Schottky TTL components and took more than two years to develop.

This general-purpose tool is capable of capturing 64 bits of data every 100 nanoseconds and reducing that data to usable form. The 256 kilobits of internal memory can be configured in many different word lengths to record, for instance, a 64-bit count of 4,096 different events, a 32-bit count of 8,192 different events, or a single flag for 256-K events. In addition, Xplor has programmable state machines with which data can be captured based on complex sequences of events; it includes hardware for the emulation of various cache organizations.

Two different Xplor configurations were developed to gather data for the TXP processor. The first was an instruction histogram measurement that records the frequency with which each instruction occurs, the percentage of time spent in each instruction, and the average number of code and data reads and writes performed by each instruction. The data is recorded in 64-bit counters, so in effect an unlimited amount of real-time data can be taken before the counters overflow.

The second Xplor configuration monitors memory addresses and emulates the tag store of a cache. Hit ratios for many different cache organizations can be determined by varying the effective cache size, associativity (one-, two, or four-way), block size, and replacement algorithm. Because the data is taken in real-time and reduced on-line, the hit-ratio measurements are much more accurate than the traditional technique, in which short address traces are recorded on tape for later analysis. This is especially important in transaction processing, since a large amount of process switching takes place; some individual transactions can last several seconds, during which millions of memory references take place.

Once the measurement methods were working, Xplor was attached to an eight-processor NonStop II system. A typical transaction-processing benchmark was brought up on the system, and transactions then were generated by another system, running software that simulated users at a number of terminals. At that point, histogram and cache measurements were taken for several of the central processing units.

The results of the histogram measurements helped determine some of the data-path widths and organizations for the TXP processor. Once the most frequently executed instructions were known, the design was modified to provide more hardware support for them. Since the measurements distinguished different paths through some instructions, tradeoffs could be made in the microcode to make the frequent cases faster.

The results of the cache measurements brought about some major changes in the original cache organization. In one measurement, the hit ratio went from 97% for the original cache to 99% for the final one, for an overall CPU performance gain of over 15%.

---

but would require three if the extract operations could not be done simultaneously.

The dual 16-bit data paths tend to require fewer cycles than a single 32-bit path when manipulating byte and 16-bit quantities and slightly more cycles when manipulating 32-bit quantities. A 32-bit add takes two cycles rather than one, but the other data path is free to use the two cycles to perform either another 32-bit operation or two 16-bit operations.

Time disadvantage

The time disadvantage in performing a single 32-bit operation is partially offset by the cycle-time advantage for 16- versus 32-bit arithmetic (32-bit arithmetic requires more time for carry propagation). Measurements of transaction-processing applications have shown that the frequencies of 32-bit arithmetic are insignificant relative to data-movement and byte-manipulation instructions, which are handled more efficiently by the dual data paths than by a single 32-bit data path. Most instructions have enough parallelism to let the microcode make effective use of both data paths.

To control the large amount of parallelism in the NonStop TXP system processor, a wide control-store word is required. The effective width of the control store is over 100 bits. To reduce the number of RAMs required, the control store is divided between a vertical control

store of 8-K 40-bit words and a horizontal control store of 4-K 84-bit words. The vertical control store controls the first stage of the microinstruction pipeline and includes a field that addresses the horizontal control store, whose fields control the pipeline's second stage. Lines of microcode that require the same or similar horizontal controls can share horizontal-control-store entries.

Unlike microprocessor-based systems that have microcode fixed in read-only memory, the NonStop TXP system microcode is implemented in RAM, so it can be changed along with normal software updates and new performance-enhancing instructions can be added.

The NonStop TXP processor uses three-stage pipelining for both macro- and microinstructions. Figure 2 illustrates the operation of the macroinstruction pipeline for a sequence of three instructions. The first is a load instruction that loads a word into the hardware stack. The second is an add immediate instruction that adds a constant to a register on the hardware stack, and the third is a final store, which stores the result in memory.

With no pipelining, this sequence would require 24 (8+7+9) clock cycles to execute, but because the prefetch and part of the execution of each instruction can be overlapped with previous instructions, the actual execution time is just 9 (3+2+4) clock cycles. Because instructions are pipelined, the TXP processor can execute its fastest instructions in just two clock cycles (167 ns),
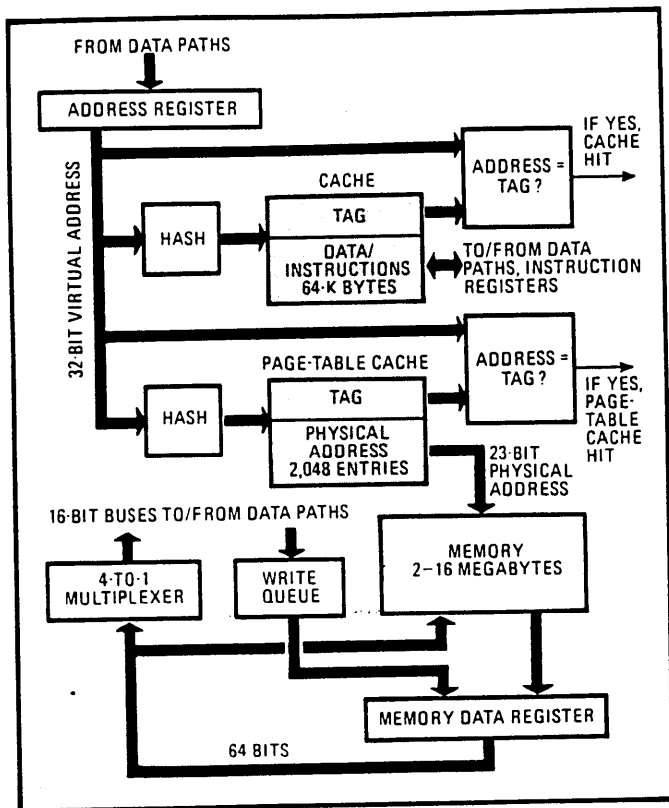
3. Memory access. The simple but extensive organization of the TXP cache provides an average hit ratio of over 96%. With a cache hit, the data is read out of the cache in 83 nanoseconds. When the data requested is not in cache, a cache miss results and the 64-bit-wide access to memory speeds the cache refill.

and it can execute load and branch instructions, which are frequently used, in only three clock cycles (250 ns).

Each NonStop TXP processor has a 64-K-byte cache that holds both data and code. A 16-processor NonStop TXP system has a full megabyte of cache memory. To determine the organization of the cache, a number of measurements were performed on a NonStop II system using a specially designed hardware monitor (see "Hardware-performance monitor helps optimize design," p. 149). The measurements showed that higher cache hit ratios resulted with a large, simple cache (directly mapped) than with a smaller, more complex cache (organized as two- or four-way associative). Typical hit ratios for transaction processing on the NonStop TXP system are in the range of 96% to 99%.

Cache miss

Cache misses are handled in a firmware subroutine rather than by the usual method of adding a special state machine and dedicated data paths for handling a miss. Because of the large savings in cache hardware, the cache can reside on the same board as the primary data paths; keeping these functions proximal reduces wiring delays and contributes to the fast 83.3-ns cycle time.

The cache is addressed by the 32-bit virtual address rather than by the physical address, thus eliminating the extra virtual-to-physical translation step that would otherwise be required for every memory reference. The virtual-to-physical translation, which is needed for refilling

the cache on misses and for storing through to memory, is handled by a separate page table cache that holds mapping information for as many as 2,048 pages of 2-K bytes each (Fig. 3).

A cache memory by itself does not necessarily boost a processor's performance significantly. It is of little use for the cache to provide instructions and data at a higher rate than the rest of the CPU can process. In the TXP processor, the cache's performance was tuned to provide instructions and data at a rate consistent with the enhancements to instruction processing provided by increased pipelining and parallelism.

32 bits and more

The two concerns related to a system's word length are capability and performance. The NonStop TXP system has 32-bit virtual addressing built into the hardware, so is capable of addressing a gigabyte of virtual memory. In addition, the TXP processor can manipulate 32 bits of data at a time through its dual 16-bit data paths. Thus the 32-bit NonStop TXP system has the additional advantage of being able to run software that was originally written for the 16-bit NonStop II system; both systems have been provided with instructions that can operate on 8-, 16-, 32-, and 64-bit data types.

In transaction processing, measurements of instruction frequencies show that data-movement instructions (loads, stores, and moves) occur much more frequently than 32-bit arithmetic instructions. For this reason, the NonStop TXP system is optimized to handle data movement by providing 64-bit access to main memory and 32-bit buses and address registers to make memory addressing as efficient as possible.

The NonStop TXP processor was implemented on four large pc boards using high-speed FAST logic, PALs, and high-speed static RAMs. The CPU's logical and physical partitioning was carefully controlled to ensure that the machine's basic cycle time would not be slowed by long propagation delays. The four CPU boards are:

- SQ: containing the control store and sequencing logic.
- CC: containing the I/O channel and various special modules.
- IP: holding the main data paths and cache.
- MC: providing the memory interface, barrel shifter, and interprocessor bus interface.

Each CPU module also has from one to four memory boards. On the initial release, each memory board contains 2 megabytes of error-correcting memory implemented with 64-K dynamic RAMs. A 16-processor NonStop TXP system can therefore contain up to 128 megabytes of physical memory.

The NonStop TXP system was designed to be easy to manufacture and efficient to test. Data and control registers were implemented with shift registers configured into several serial-scan strings. The scan strings are of value in isolating failures in field-replaceable units. This serial access to registers also makes board testing much faster and more efficient because the tester can directly observe and control many control points. A single custom tester was designed for all four CPU boards and for the memory-array board as well.

The NonStop TXP system is the first product to be

MIPS and transactions per second

Determining relative performance among computer systems has never been an easy task. The often-quoted millions-of-instructions-per-second rate is intended as a way to compare basic central-processing-unit-hardware performance. Comparisons are also made on the basis of benchmarks. CPU-intensive benchmarks measure the performance of the CPU hardware and compiler; more extensive benchmarks measure the entire system performance—including the hardware, compiler, operating system, and data-base-management system. In general, the more extensive benchmarks give a more accurate prediction of actual system performance.

Each of the various measurement techniques has pitfalls. The MIPS rate is perhaps the least accurate way to compare systems. One reason is that there is no easy way to relate the power of one instruction set to another. In addition, vendors vary in the way they measure MIPS: some use it for the speed of the fastest instructions, others measure the speed of the most frequently executed instructions, and still others measure the speed of a "typical" mix of instructions. According to these definitions, each NonStop TXP processor is 6, 4, or 2 MIPS, respectively.

---

developed using Tandem's proprietary computer-aided-design system. The CAD system's capabilities for logic entry, logic simulation, and automated pc-board routing were instrumental in reducing the design time. While most high-performance CPUs require four to five years to develop, the NonStop TXP processor took just 2½ years—six months to complete a written specification, one year to construct a working prototype, and another year to reach volume production.

Performance measurement

Some simple benchmark programs have recently become popular in measuring performance (see "MIPS and transactions per second," p. above). One is the Puzzle benchmark, which is a CPU-intensive program to solve a three-dimensional puzzle. Execution times for Puzzle can vary widely for the same machine, depending on whether the program accesses arrays through subscripts or pointers and whether frequently used variables are assigned to registers. Versions of the Puzzle benchmark with pointers and registers were used to compare relative performance for a TXP processor.

Puzzle was written in TAL (transaction application language, the company's system-programming language); the execution time, using a single TXP processor, was measured at 1.67 s. This compares with 4 s on a VAX-11/780 for Puzzle written in C.[1] Because Puzzle does not measure such system features as support for virtual memory, I/O bandwith, and the ability to do fast context switching, a standard benchmark for comparing transaction-processing systems is still needed.

One transaction-processing benchmark has been developed by a third party, however. The U. S. Public Health Service ran an extensive benchmark in 1981 to determine which system to select for a large on-line medical-information system.[2] In that study, a 15-processor Tandem NonStop system running a 1981 version of Tandem's Encompass DBM system performed the benchmark at a rate of 4.5 transactions/s. An International Business Machines Corp. System 370/168-3 running version 3 of the Adabas DBM system performed the same benchmark at 2 transactions/s.

This benchmark gives a data point for comparisons between Tandem and IBM systems. A 15-processor NonStop system performs the Public Health Service benchmark 2.25 times as fast as an IBM 370/168-3. Though it would be desirable to compare the TXP system directly to one of IBM's newest systems, such as the IBM 4381-2, no competitive benchmarks have been published. However, comparisons of the MIPS rate of different processors within a single family are fairly accurate and can be used to extrapolate to newer systems.

According to market research performed by the Gartner Group,[3] the IBM 4381-2 is rated at 2.7 MIPS, compared with the older IBM 370/168-3's 2.4 MIPS rating—a ratio of 1.125 : 1. Company tests have shown the NonStop TXP to have a MIPS rate approximately three times that of the NonStop processor. The extrapolation of the Public Health Service benchmark performance to the two newer systems is shown in Table 3.

Unlike many shared-memory multiprocessor systems, Tandem systems provide linear growth in transaction-processing power as the system expands. A single system can include up to 16 processors, and clusters with as many as 224 NonStop TXP processors may be configured with Tandem's fiber-optic link. Clusters with up to 60 processors are currently in operation, and their users have verified the linear-performance growth within a cluster of this size.

The largest IBM mainframe today is the IBM 3084, which is rated at approximately 23 MIPS. Extrapolation from the benchmark data suggests that the performance of a cluster of 224 TXP processors is on the order of 10 times as powerful as IBM's top-of-the-line 3084 processor. □

TABLE 3: TANDEM VERSUS IBM PERFORMANCE COMPARISONS	U.S. Public Health Service benchmark: results (transactions per second)	USPHS benchmark: extrapolated results* (transactions per second)
IBM 370/168-3	2	—
Tandem NonStop 15-processor system	4.5	—
IBM 4381-2	—	2.25
Tandem NonStop TXP 3-processor system	—	2.7
		*Not actual measurements

References
[1]Malcolm A. Gleser, Judith Bayard, and David D. Lang, "Benchmarking for the Best," Datamation, May 1981.
[2]Computer Architecture News, 10 : 1, March 1982, p. 29.
[3]Gartner Group Inc., Stamford, Conn., market research surveys.