AccessionIndex: TCD-SCSS-V.20221015.004
Accession Date: 15-Oct-2022
Accession By: Dr.Brian Coghlan
Object name: Multimax Technical Summary
Vintage: 1989
Synopsis: Encore Computer Corporation, 726-01759, Rev E, Jan-1989.

Description:
Encore Computer Corporation was established in Marlborough, Massachusetts in 1983 by some very experienced, senior and notable computer professionals: Kenneth Fisher (ex-CEO, Prime Computer), Gordon Bell (Vice President, Digital Equipment Corporation [DEC]) and Henry Burkhardt III (co-founder of Data General and Kendall Square Research). The intention was to commercialise parallel machines using commodity microprocessors.

The Encore Multimax commercial symmetric multiprocessor (SMP) minicomputer was introduced in 1985, based on a 10MHz National Semiconductor NS32000 series microprocessor chipset, which included a 32-bit CISC CPU, a floating-point coprocessor, a virtual memory coprocessor, and several other support chips, very attractive for its clean and orthogonal instruction set and the simple interconnection of its CPU-group hardware. Up to ten processors shared a proprietary 100Mhz snoopy cache-coherent 32-bit address and 64-bit data *Nanobus*. It ran symmetric ports of the BSD, UNIX System V or Mach operating systems.

Unfortunately the NS32000 had lost its market momentum due to many design bugs in its original NS16032 / NS32016 chipsets. Later Multimax models employed faster NS32332 and NS32532 chipsets, but National Semiconductor National halted further development of the NS32000 series in 1989. Encore then pivoted to the Motorola 88000 RISC microprocessor, but then in the early 1990s its further development was halted too. By 1994 Encore pivoted yet again to the DEC Alpha 21064, but with little success.

Their main competitor was initially the Sequent Balance, also NS32000-based, but in 1987 Sequent switched to i80386 processors for a successor, the Sequent Symmetry. Both companies were pioneers of large-scale symmetric multiprocessing, which with contemporary advances in cache coherency protocols had become a hot topic. Encore still exists as an entity, albeit not an independent company anymore.

The homepage for this catalog is at: https://www.scss.tcd.ie/SCSSTreasuresCatalog/
Click *'Accession Index'* (1st column listed) for related folder, or *'About'* for further guidance. Some of the items below may be more properly part of other categories of this catalog, but are listed here for convenience.

Accession Index	Object with Identification
TCD-SCSS-V.20221015.004	Multimax Technical Summary. Encore Computer Corporation, 726-01759, Rev E, Jan-1989. 1989.
TCD-SCSS-V.20150617.001	Sequent Symmetry S27 Documentation, Sequent Symmetry S27 Logbook, Technical Summary, Dynix System Administration Guide, Diagnostic Supervisor User's Guide, SCED Power-Up Monitor Guide p.1-39/40/41, and brochures on Campus Timesharing and Scientific Research, S27 and S81 Parallel Computing Systems, ATS FORTRAN Compiler, X Window System, and Academic and Research Accounts. c.1987.
TCD-SCSS-T.20121208.064	Sequent Symmetry S27. Second-generation commercial symmetric multiprocessor (SMP) minicomputer, i80386-based, running Dynix, 1987.

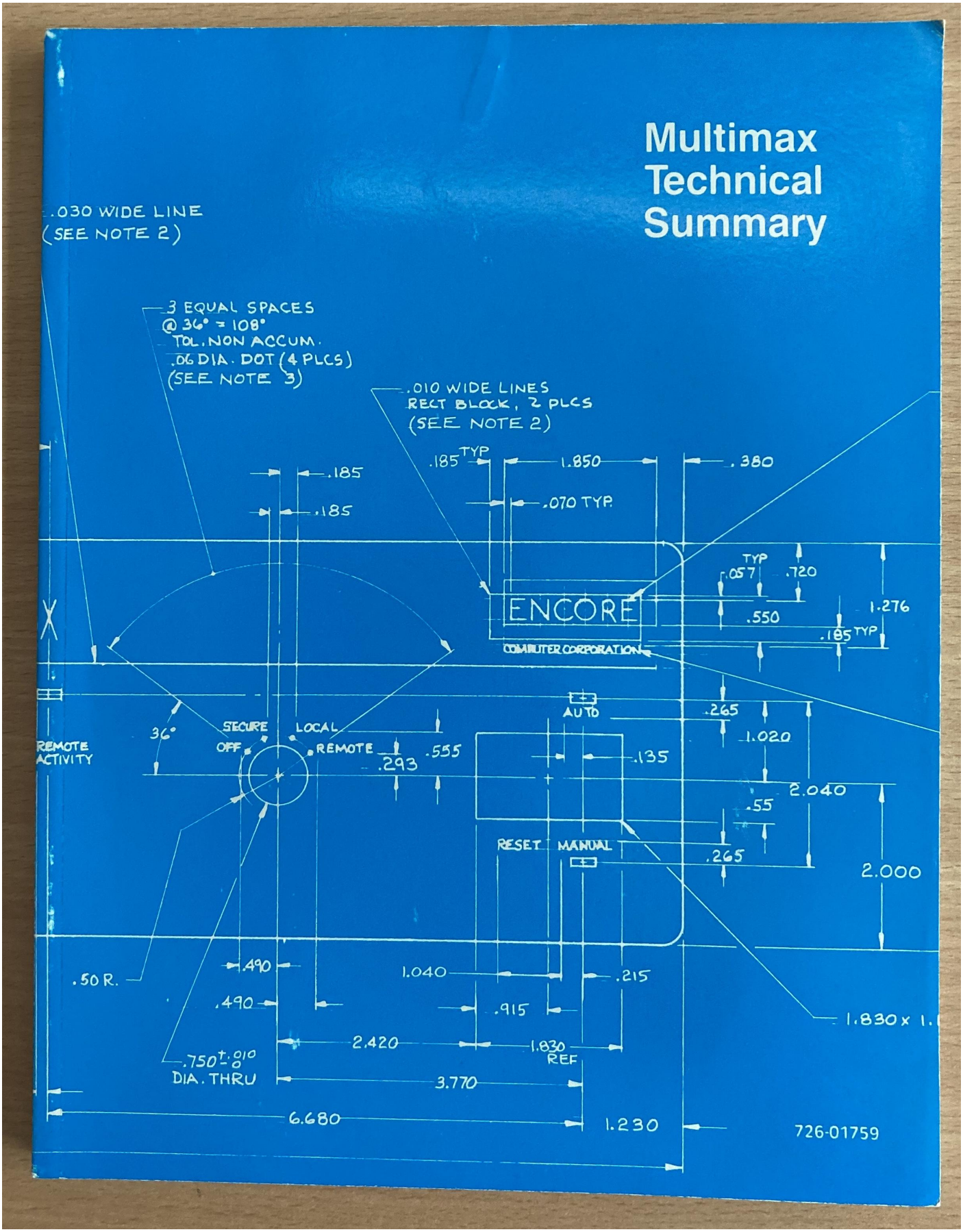

Figure 1: Encore Multimax Technical Summary, front cover

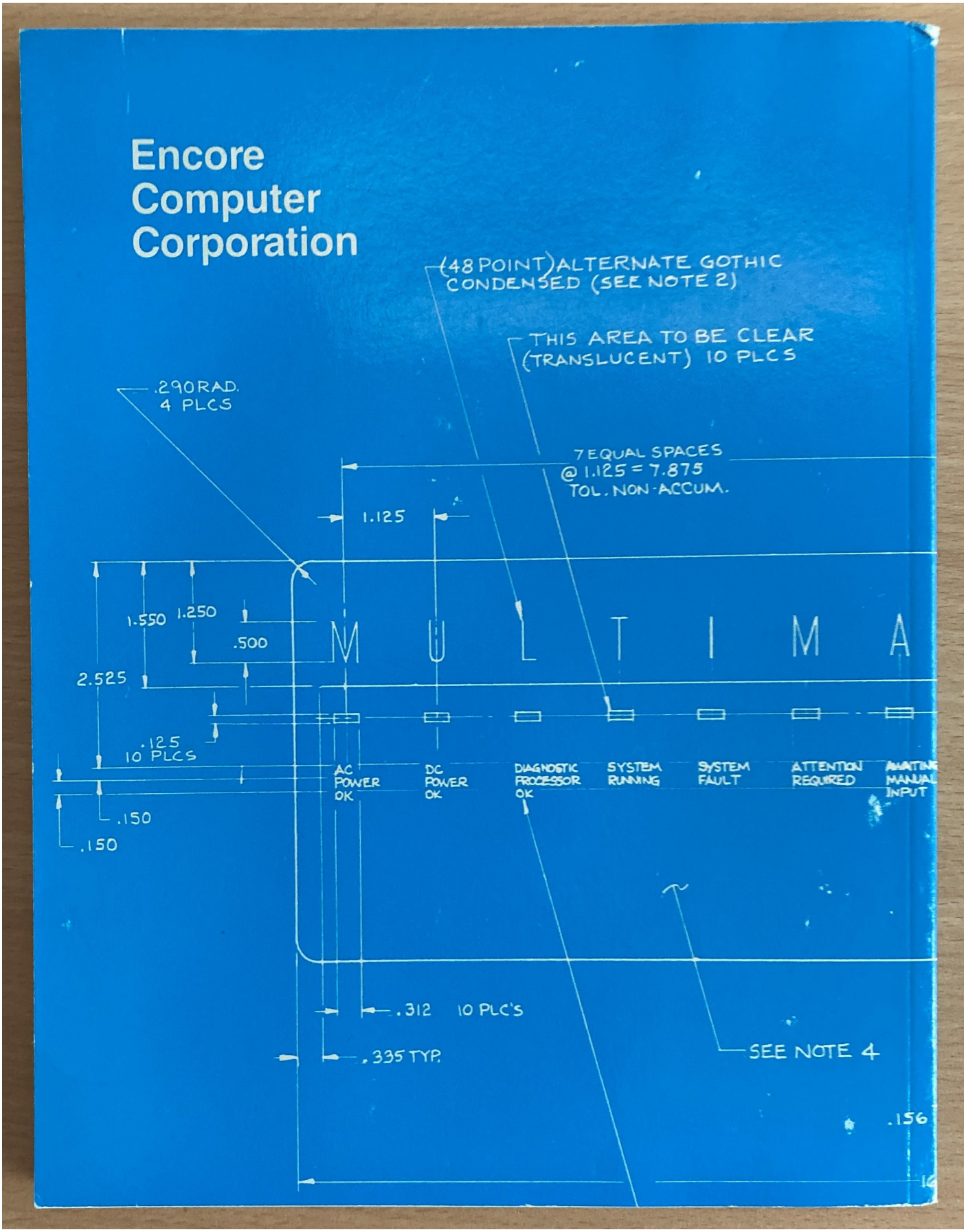

Figure 2: Encore Multimax Technical Summary, rear cover

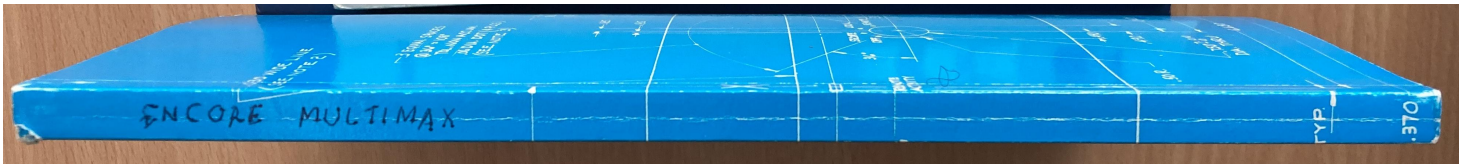

Figure 3: Encore Multimax Technical Summary, spine

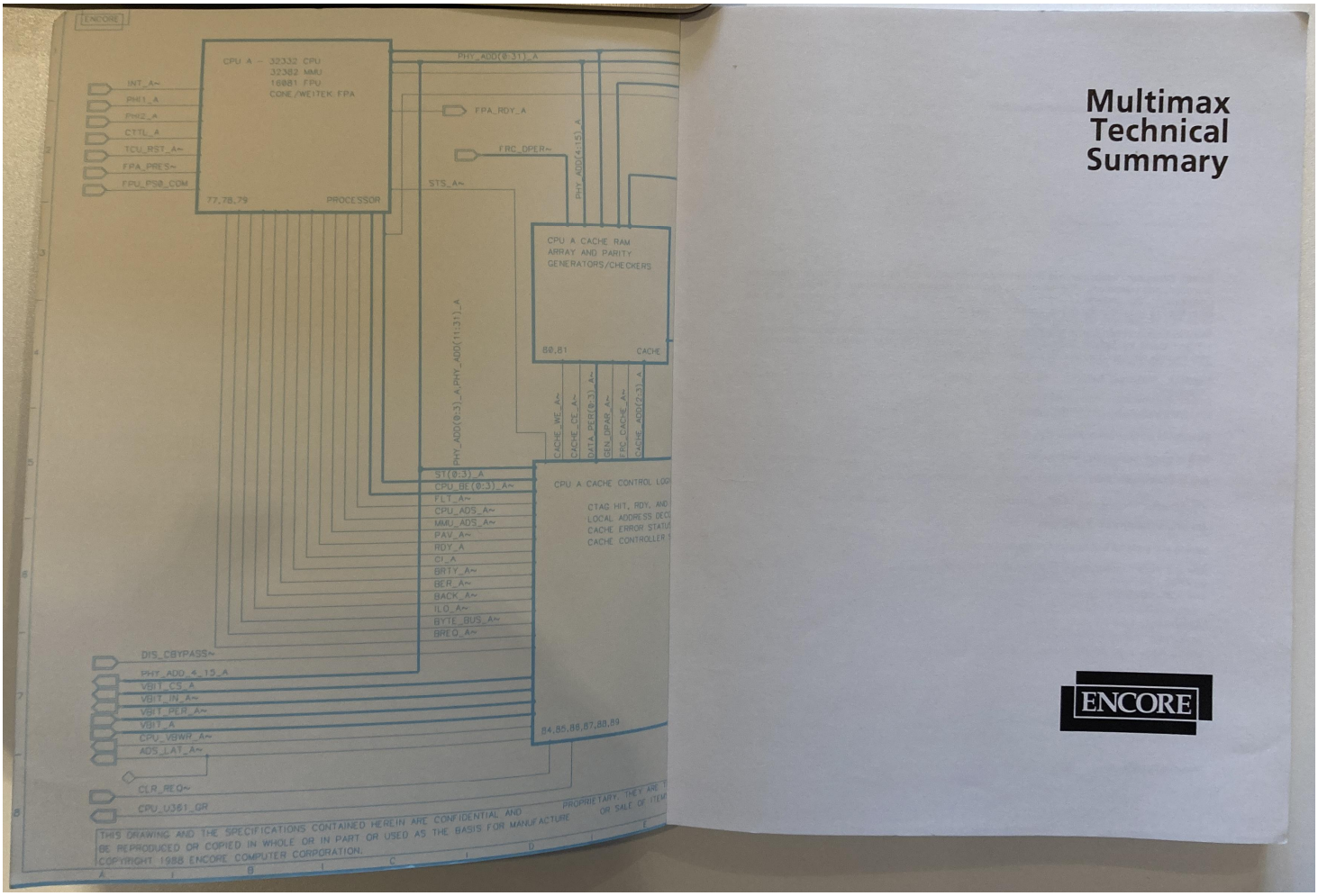

Figure 4: Encore Multimax Technical Summary, inside-front-cover and page i

NOTICE

Encore Computer Corporation has prepared this manual for use by Encore Computer personnel, licensees, and customers. The information contained herein is the property of Encore Computer Corporation. The contents of this manual shall not be reproduced in whole or in part nor used other than as allowed in the software license agreement.

The information in this manual is subject to change without notice, and should not be construed as a commitment by Encore Computer Corporation. Encore Computer Corporation assumes no responsibility for any errors that may appear in this document.

GigaMax, Multimax, Nanobus, Parasight, and UMAX are trademarks of Encore Computer Corporation.

UNIX is a trademark of AT&T Bell Laboratories.

Ethernet is a trademark of Xerox Corporation.

NFS is a trademark of Sun Microsystems, Inc.

Ada is a trademark of the U.S. Government (Ada Joint Program Office).

DECnet, ULTRIX, VAX, and VMS are trademarks of Digital Equipment Corporation.

Annex is a trademark of Xylogics, Inc.

Verilog is a trademark of Design Automation Corporation.

IBM and VM are trademarks of International Business Machines, Inc.

726-01759
Rev E
January, 1989

Printed in the USA

Preface

The Beginnings

Founded in May, 1983 by Kenneth Fisher and others, Encore Computer Corporation spent its early months exploring possible company directions. By the end of the summer of 1983, the company's executives and engineers had decided that computer technology had reached an evolutionary stage justifying the commercial design and implementation of a parallel processing architecture. This decision was based on two key facts:

- The price/performance ratio of microprocessors had grown, and promised continued growth, at a much faster rate than that of discrete component processors. Already, in 1983, low-cost single microprocessor chips were available with performance approaching that of low-end superminicomputers.
- Operating systems had for some years supported multiprogramming, allowing uniprocessors to emulate the execution of multiple processes in parallel. Encore engineers agreed that modifying an industry standard operating system such as UNIX to allow simultaneous execution of multiple processes on separate processors ought to be a feasible task – provided that the system hardware was designed to assist in the complex coordinations required.

For these reasons, Encore decided to sponsor the development of a new computer that would take maximum advantage of existing hardware and software standards, existing and developing microprocessor and peripheral chips, and an as yet undetermined bus design. Marketing and engineering personnel agreed that a successful machine must deliver an unprecedented price/performance ratio, must be modular and expandable, must provide implicit as well as explicit opportunities for parallel program execution, and must provide customers access to all major languages, tools, and applications available to the industry.

Early Multicomputer Architectures

As a concept, the multiple-processor computer system was not new: the Burroughs B5000, a dual symmetrical processor, had been introduced in 1961. In the following

Figure 5: Encore Multimax Technical Summary, pages ii and iii

decade, a variety of approaches had been explored in the academic community, and a number of loosely-coupled commercial machines had been built that housed more than one processor, bus, and memory structure in a single cabinet and provided limited ability to share work loads. But such systems were expensive: in the pre-LSI era, processors were large, complex, and power hungry, and few economies resulted from putting multiple independent computing structures in the same cabinet.

In the late 1970's, a number of special-purpose systems had been built utilizing array processors, dedicated parallel function units, or connection-based architectures (in which all processor units were directly connected to private memory units and in which processor units communicated with one another via messages, not shared memory). But the latter approaches promised to be either too specialized or too difficult to program to meet Encore's requirements. Encore was seeking an architecture that could provide straightforward support of traditional computer applications on the one hand and allow efficient creation of parallel applications on the other. The special-purpose systems promised neither characteristic.

Encore's Preferred Architecture

Encore's engineering staff believed that they should be able to devise a parallel architecture that would circumvent the need for programs to be written expressly for the hardware platform. This meant avoiding the requirements for explicit memory partitioning and complex message and data passing imposed by most of the connection-based architectures. The solution seemed to lie in a machine with a bus-based, symmetric shared-memory architecture that could execute traditional sequential programs as though it were a conventional uniprocessor. In doing so, however, it would reduce context switches and improve overall throughput in multiple-user environments by dynamically allocating independent processes to as many processors as were available. Such a system would not only show time-sharing performance improvements that would be a more or less linear function of the number of processors, it would also allow programmers to create explicitly parallel applications with minimum attention to the design of the hardware.

Problems to be Considered

The central problem that Encore had to address in designing the new machine was the necessarily finite bandwidth of any possible shared bus design. Since a shared bus provides a single pathway between processors and memory, sooner or later it must become a bottleneck as more or faster processors are added and more bus traffic is incurred. The conventional wisdom of the late seventies was that shared bus multiprocessors would encounter this bus saturation point with four to six processors, and that adding processors beyond this point would not materially improve performance.

Encore's response to this problem was that, although unlimited scalability of a parallel processing architecture might be theoretically attractive, it came at too high a price in terms of the agreed-upon objectives. Encore decided that limited scalability was acceptable and could produce very worthwhile improvements in price/performance if the saturation point could be moved high enough.

Bus Design

After some deliberation, Encore engineers agreed that raising the bus saturation point to an acceptable level would involve two fundamental design efforts. The first was to create a wide, very high-speed bus that would have surplus bandwidth to permit the bus to accomodate several generations of microprocessors, each operating at two or three times the speed of its predecessor. Out of this design effort came the Encore Nanobus, a 100 Mbyte/second structure providing independent address (32 bits plus parity), data (64 bits plus parity), interrupt vector (14 bits), and control lines. Since the Nanobus architecture permits pended and pipelined bus transactions, system components can exchange massive amounts of data with minimum bus contention.

Cache Design

The high-bandwidth Nanobus design, however, was only part of Encore's solution to the bus saturation problem. A second component of the solution was to minimize the need for processors to access the bus at all. Since the principal activity of Von Neuman processors is to access instructions and data from memory and to return modified data to memory, it was clear that one way of reducing the load on the bus was to provide each processor with an auxilliary local cache memory that could be used to store the most recently accessed (and most likely to be reaccessed) instructions and data. Because caches would be small relative to the size of main memory, they could be implemented with high-speed static RAM that could be accessed without wait states and thereby serve the additional purpose of speeding up processor execution time while still preserving the shared-memory model of the architecture. The first generation of Encore processor boards allocated 16K bytes of cache memory for each processor, the second generation allocated 64K bytes, and the third 256K bytes. Although the growing power of each processor generation made increasingly severe demands on cache memory, these cache sizes provided estimated hit rates in the neighborhood of 89%, 95.7%, and 97.5%, respectively: once the cache was loaded, only a very small proportion of memory accesses needed to be made across the bus to main memory.

Bus Watching Protocol

Designers have been using cache memory to reduce bus traffic and speed up memory access on uniprocessors for many years. When the technique is applied in a shared memory multiprocessing environment, however, we encounter a problem of data coherency: since memory is shared and can be modified by any processor on the

Figure 6: Encore Multimax Technical Summary, pages iv and v

system, the same data may be replicated in more than one cache and may therefore be subject to change in incompatible ways by more than one processor.

The shared bus becomes an asset in dealing with this problem. In Encore's first and second generation processor boards, memory writes were always performed on main memory, not cache. To maintain cache coherency, each cache structure watched the bus for memory write operations to locally held locations. When such writes were detected, the cache invalidated its corresponding entry and thus forced the next access of that entry to acquire current data from main memory.

This *write through* protocol was quite effective in reducing bus traffic and maintaining cache coherency among processors of moderate speed such as those on Encore's first two generations of processor cards: theoretical aggregate system speeds of 60 million instructions per second (MIPS) were supportable by this procedure, giving comfortable headroom to a system whose maximum speed (in 1986) was approximately 40 MIPS. However, a more sophisticated scheme became necessary with the higher-speed third generation cards. Since these cards would give the largest system a maximum aggregate speed in excess of 170 MIPS, the 60 MIPS ceiling provided by the write-through cache protocol would have been totally inadequate.

For the third generation processor cards, Encore engineers therefore decided to implement a new procedure for maintaining cache coherency and reducing bus traffic. This would be what is known as a *write deferred* cache protocol. This protocol writes data to cache rather than main memory, witholding main memory writes until some system requirement causes the data to be written back. This occurs either when the cache location is required for other data (reallocation), or when some other processor in the system needs to read the memory location in question. Such locations can be written and read many times before being replaced, and for that reason accesses to them practically disappear from the Nanobus. Coherency is maintained across caches by a special system of ownership and sharing whereby the cache to which an entry is being written must first attain ownership of that location, causing any other caches sharing the data to mark their copies invalid. The owning cache must now assume the responsibility of providing the changed data to any cache that subsequently wishes to read it. Finally, the current owner must mark its copy invalid if any other cache requests ownership.

Write deferred cache protocol in conjunction with the 256K byte cache size used on current Encore processor cards allows a theoretical aggregate system speed in excess of 200 MIPS – providing acceptable headroom for the 170 MIPS maximum speed of current Encore systems.

Future Hardware Development

But 200 MIPS is by no means the upper limit of the Nanobus architecture. Encore is currently developing a system, tentatively called GigaMax, under a contract from the Defense Advanced Research Projects Agency (DARPA) that uses a patented, tightly-coupled, multiple bus structure able to maintain cache coherency and greatly increase maximum performance far above 200 MIPS. The Gigamax will consist of up to eight Multimaxes interconnected by bus cables carrying high-speed, differential, emitter-coupled logic (ECL) signals, with a ninth Nanobus acting as a global switch. Each Multimax will use write-deferred protocol to maintain internal cache consistency, and the ninth Nanobus will use the same protocol to maintain coherency between Multimaxes. The Gigamax should easily be capable of over 1000 MIPS of performance.

Multimax Software

One of the principal goals of the shared-memory multiprocessor architecture discussed in the previous paragraphs is to allow system users to achieve near linear performance improvements as processors are added. Meeting this objective requires not only appropriate hardware, but also a parallel operating system that offers symmetric access to system resources. Although the multiprogramming character of the standard distributions of Berkeley and AT&T UNIX systems showed considerable promise as starting points for such an operating system, they share the assumption that only one thread of execution at a time can run in the kernel. If such single-threaded execution is attempted on a multiprocessor, the effort to protect kernel-shared data structures produces a master-slave relationship between processors whereby only one processor (the master) is capable of executing within the kernel context.

Because such exclusiveness would be incapable of delivering the full power of the multiprocessor hardware, Encore made a number of substantial changes in the UNIX operating system kernels. Encore's kernels utilize threads of execution to allow concurrently running processes multiple, simultaneous entry into the operating system. If the threads block due to an I/O or signal wait, their state is saved. When they are again free to run, they can do so on whatever processor is currently available. Similarly, a thread that is blocked for an I/O completion can have its I/O completion interrupt processed by any available processor. This fully symmetric operating system behavior, working in conjunction with the automatic load-balancing features of the Multimax, transparently converts the multiprogramming characteristics of standard UNIX into true multiprocessing characteristics. The result is that multiple-process environments, like those that have always typified UNIX, are parallelized automatically.

Encore's two primary operating systems, UMAX 4.2 and UMAX V (compatible with Berkeley Standard Distribution 4.2 and AT&T's System V.3, respectively), have each been modified and optimized to allow users the maximum amount of concurrency. All of the key user interfaces, networking capabilities, networked file systems, system

Figure 7: Encore Multimax Technical Summary, pages vi and vii

services, and utility programs that are found on other UNIX 4.2 BSD and System V systems are provided by Encore's UMAX 4.2 and UMAX V. These operating systems are compatible with one another across local and wide area networks and are compatible with other UNIX-based systems that support the Internet protocol family.

In anticipation of the demands that the developing GigaMax will permit users to make on the Multimax, Encore has added MACH to its group of operating systems. MACH is a multiprocessor-oriented operating system and environment that was initially designed at Carnegie-Mellon University and is being further developed as part of the DARPA contract that is funding the development of GigaMax. Compatible with UNIX 4.3, MACH separates a UNIX process into threads of control and a set of resources – for example, virtual address space, interprocess communication facility partitions, and file descriptors. Because MACH allows multiple threads to be associated with a single set of resources and also allows each thread to be dynamically mapped onto a process, it provides the basis of an extremely effective parallel processing paradigm.

Encore has added extensions to its production operating systems for use by parallel programmers. One such extension for UMAX V, *gang scheduling*, permits dedicating a group of processors exclusively to a set of processes, allowing a parallel application to avoid usual UNIX scheduling and its associated latencies and overhead. Gang scheduling is useful for minimizing synchronization delays in time-critical parallel programs or for performing accurate timing of external events such as I/O device latencies.

The preceding paragraphs do no more than hint at the volume of Encore's developments since the company's founding. Not only have operating systems been ported and tuned to the new hardware, but debugging and profiling tools have been developed for the parallel programming environment, compilers (both parallelized and sequential) have been ported and optimized, and many of the most widely used applications programs have found new vitality in the Multimax environment.

The remainder of this book documents the details of this accomplishment. Encore believes that it has fulfilled its original commitment: the Multimax delivers an unprecedented price/performance ratio; it is modular and expandable; it provides implicit as well as explicit opportunities for parallel program execution; it provides customers access to the major languages, tools, and applications available to the industry. The beneficiary of an extremely aggressive research and development effort, it has already undergone numerous enhancements; in the near future, it will undergo even more.

We at Encore Computer Corporation are proud of our contribution to the computer art, and our customers generally acknowledge that we have reason to be. To find out more about what prompts their enthusiasm, read on.

Contents

Figure 8: Encore Multimax Technical Summary, pages viii and ix

Figure 9: Encore Multimax Technical Summary, pages x and xi

Chapter 1

The Systems

Encore Computer Corporation's Multimax systems represent the best of a new computer class characterized by a high-performance architecture that accommodates multiple processors in a large number of different configurations. There are currently two basic platforms in the Multimax family, one providing slots for 11 Nanobus cards, the other providing slots for 20 Nanobus cards. These platforms can be subdivided according to the kind of 32-bit processor cards installed in their backplanes. Since Encore currently offers moderate- and high-speed processor cards, the two card types and the two platforms combine to produce four basic models in the Multimax line:

- The Multimax 310. Supports from 2 to as many as 10 moderate-speed processors, each capable of 2 million instructions per second (MIPS) – resulting in an aggregate performance rating ranging from 4 to 20 MIPS.
- The Multimax 320. Supports from 2 to 20 moderate-speed processors, each capable of executing 2 MIPS – for an aggregate performance rating ranging from 4 to 40 MIPS.
- The Multimax 510. Supports from 2 to 10 high-speed processors, each capable of approximately 8.5 MIPS – for an aggregate performance rating ranging from 17 to 85 MIPS.
- The Multimax 520. Supports from 2 to 20 high-speed processors – for an aggregate performance rating ranging from 17 to 170 MIPS.

All systems have fast shared memory (16 to 64 Mbytes on the Multimax 310/510, 16 to 160 Mbytes on the Multimax 320/520). They also have configurable extended I/O capacity (1 or more network channels and from 1 to 10 or 1 to 12 intelligent mass storage channels). On both systems, the microprocessors, memory, and I/O interfaces are coupled across a wide, high-speed main system bus, providing a single computer product that spans a spectrum of performance from that of minicomputers to that of mainframes.

Figure 10: Encore Multimax Technical Summary, pages xii and 1-1