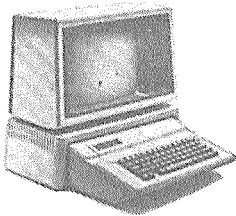


吳 吉 祿  
 韓國電子通信研究所  
 컴퓨터연구부장 / 工博

# Multiprocessor 컴퓨터 동향



本稿는 한국전자통신연구소 컴퓨터 개발부 박 승규 박사가 작성, 「전자통신 동향분석」지에 실린 것임을 밝힌다.

## I. 서 론

몇년전까지만 해도 대학의 실험실에만 머물러 있던 다중 프로세서를 사용한 컴퓨터 시스템들이 최근에 일반 컴퓨터 시장에 나타나기 시작했다. 이것은 그동안 눈부신 반도체의 발전에 따른 강력한 마이크로 프로세서의 출현과, 값싼 CPU를 여러개로 묶어 더 강력한 시스템을 구성할 수 있게 되었기 때문이다.

이와 같은 추세를 가속화시키는 또 하나의 원인은 시스템들의 표준화와 개방성에 기인한다. 즉, 예전에는 미니 이상급의 컴퓨터엔 CPU 하드웨어에서부터 운영체제 등 전통적인 기술을 대기업만이 축적하였으므로 신흥 작은 기업에서는 도전이 쉽지 않았다.

그러나 현재는 UNIX계열을 비롯한 소프트웨어에서부터 하드웨어의 여러부분이 표준화됨으로써 모범기업들이 한 부분만 개발하여도 전체 시스템을 구성할 수 있게 되었다. 그러므로 후발주자 업체가 미니 이상급의 컴퓨터 개발을 시도할 경우, 자연스럽게 마이크로 프로세서들을 다중으로 엮는 시스템 개발방법을 시도하게 되는 것이다.

요사이 활발히 진행되고 있는 다중 프로세서 시스템은 최신의 마이크로 프로세서의 채택과 고속 시스템 버스 설계 그리고 OS에 대개 UNIX 계열을 채택하고 있다.

그러나 아직 UNIX 자체가 이러한 다중 프로세서 시스템에 적합한가의 여부, 또 이와 같은 다중 프로세서를 잘 이용할 수 있는 환경소프트웨어의 정립이 아직 제대로 되어 있지 않다. 이와 같이 하드웨어에서부터 시작되고 있는 다중 프로세서 계열의 시장에서는 성능표시의 혼돈, 용어의 혼돈 등 많은 혼란이 동시에 초래되고 있어, 현재 이 분야의 기술은 모두 준비되었음

에도 불구하고 용어의 혼란 및 사용자의 이해부족 등으로 실제 상품이 제대로 보급되는 것은 2~3년후일 것이라는 예측도 있는 실정이다.

이러한 혼란을 극복하기 위해 미국의 표준국(NSF) 과 국방연구소(DARPA) 등에서는 용어의 표준화와 성능표시를 위한 Benchmark 프로그램의 표준화 등을 시도하고 있다.

그러나 아직 정립해야 할 부분이 남아있음에도 불구하고 이와 같은 다중프로세서 시스템의 개발추세는 값싼 마이크로 프로세서를 이용하여

기존 중형, 대형급 수준의 컴퓨터에 도전하려는 새로운 움직임을 보이고 있어 우리나라와 같이 표준화 및 개방성을 따라가야 하는 후발주자의 입장에서는 매우 중요하고 또 빨리 이러한 추세에 동승해야 하는 분야이다.

## II. 용어 및 분류

많은 기술잡지 등에서 그 나름대로의 분류 등을 제시하고 있다. 이 글에서는, 소프트웨어 프

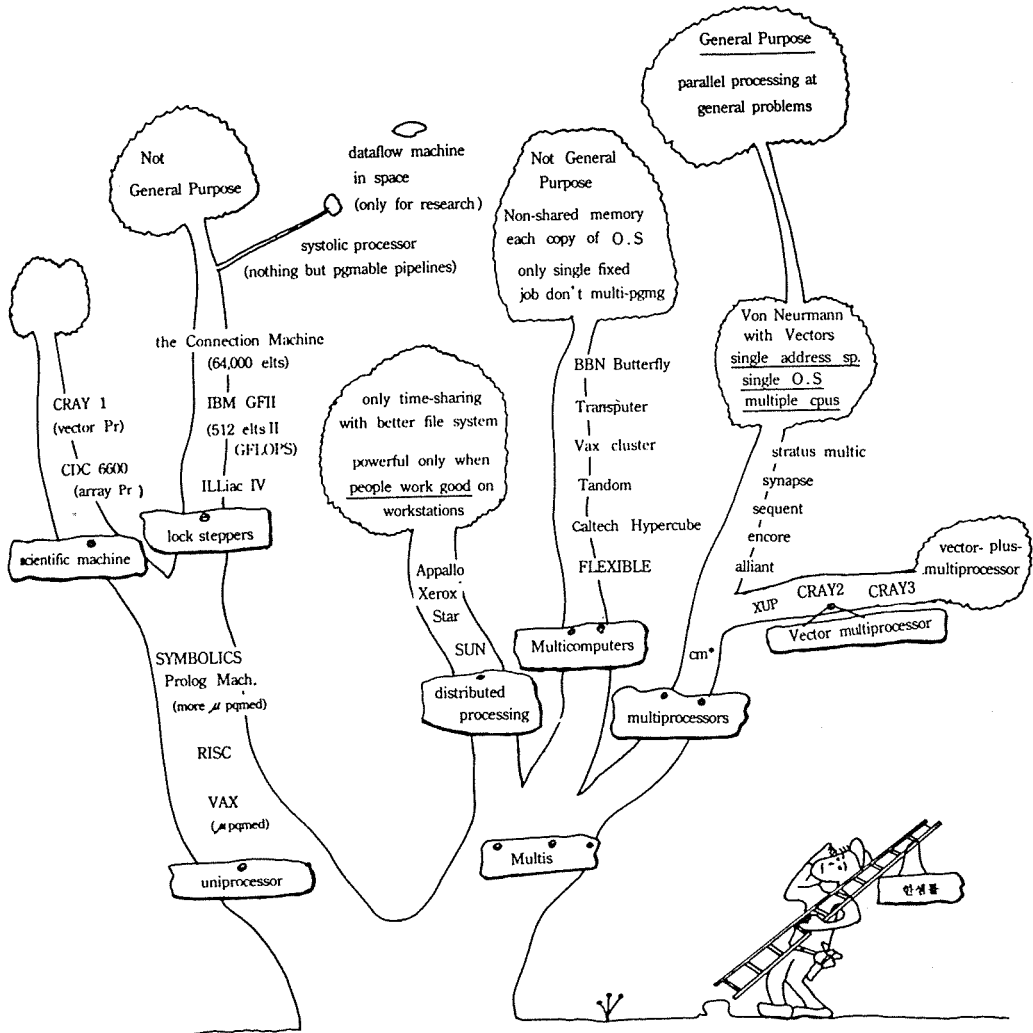


그림 1. UNIX REVIEW

로세싱의 입장에서와 컴퓨터 구조의 입장에서 분류를 한다.

소프트웨어 프로세싱의 입장은 UNIX World 지의 1986년 1월호 기사의 예를 따른다. 즉, parallel processing과 multiprocessing의 구분이다.

Parallel processing이란 logical한 하나의 문제를 많은 작은 문제로 나누어 다중 프로세서에 의해 동시에 해결함으로써 한 문제의 푸는 속도를 높이는 것이다. 즉, 만약 어떤 문제가 세개로 나누어질 수 있고 CPU가 세개 이상 있으면 한개 CPU 시스템보다 약 세배 빨리 풀 수 있다는 것이다.

Multiprocessing은 한 문제 자체를 빨리 풀지는 않으나 서로 관계없는 많은 다른 문제들이 있을때, 각 문제를 여러 다중 프로세서가 나누어 해결함으로써 전체의 throughput을 높이는 것이다. 그러므로 한 문제의 입장에서 보면 CPU가 많아도 빨리 해결되지는 않으나 여러 문제 입장에서 보면 몇배 빨리 해결이 되는 것이다.

Parallel processing의 경우는 한 문제가 서로 연관을 갖는 작은 문제들로 나뉘어지고 이것을 각 CPU에 분배하여 해결하게 되는데, 이와 같은 분배작업에 연관된 일 등은 프로그래머가 시스템의 실제모양(system configuration), 특히 프로세서의 수 및 연결모양(topology) 등을 알아서 응용 code에 기입하여야 한다. 그러므로 시스템 configuration이 바뀌면 적어도 다시 한번 recompilation을 해야 한다.

Multiprocessing의 경우, 잘 설계된 범용성(general purpose)환경 경우는 응용 프로그램 경우 실제 시스템 configuration을 전혀 의식하지 않게 되기 때문에 시스템의 가감 변경 등을 소프트웨어의 변동없이 가능하게 할 수 있다.

다시 정리하면, parallel processing은 하나의 같은 문제를 여러 subtask로 나누어 동시에(concurrently) 수행하는 것으로 run-time issue를 풀기 위한 singletask-oriented기계구조를 한다. 그 응용으로는 과학, 수학 엔지니어링 및 인공지능 등에 관한 문제를 풀기위한 구조이다.

Multiprocessing경우는 서로 독립적으로 수행되는 여러가지 task들의 throughput을 올리기

위한 것으로 Multiple-task-oriented 구조를 한다. 응용으로는 online transaction processing 분야를 주종으로 하고 있다.

위와 같은 분류로 볼 때 parallel processing을 위해서는 subtask간의 긴밀한 관계에 의해 CPU간에 load balancing이 가능한 tightly-coupled multiprocessor 시스템이어야 함을 알 수 있다. 반면에 multiprocessing 응용에서는 file 등을 job들이 공유하나 CPU간에 긴밀한 load balancing은 필요없게 된다. 그러므로 대개 loosely-coupled multiprocessor 시스템의 구성을 하고 있다. 이러한 컴퓨터의 구조 분류는 다음 장에서 살펴본다.

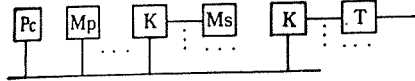
### Ⅲ. 컴퓨터 구조 분류

앞 장에서 언급한 tightly-coupled 및 loosely-coupled 구조와, parallel processing 및 multiprocessing의 관계를 보면 다음과 같다. 즉, tightly-coupled 구조는 parallel processing을 허용함과 동시에 multiprocessing도 가능하다. Gordon Bell의 주장에 의하면 shared memory에 의한 다중 프로세서 시스템은 범용 컴퓨터가 된다는 것이다.

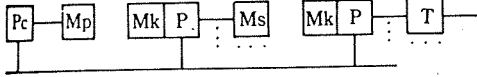
반면에 loosely-coupled된 컴퓨터 구조에서는, CPU간의 load-balancing이 어렵기 때문에 multiprocessing은 허용해도 parallel processing은 불가능하게 된다.

이와 같은 다중 processor들의 분류를 UNIX Review지 2월호에는 여러가지로 분류(Gorden Bell의 분류)하였는데 이는 그림 1과 같다.

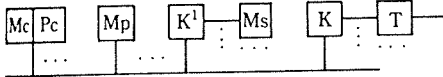
즉, 다중 컴퓨터중에서도 shared memory에 근거하여 다중 프로세서를 갖되 OS는 한 copy만 갖는 multiprocessor분류에 Alliant, Encore, Sequent, Synapse 등등의 회사들이 있다. 또 다른 분류군은 multicomputer라는 그룹으로 여러가지의 CPU간의 연결방법을 가지고 각각의 CPU가 자체 local memory를 갖고 각각의 copy에 OS를 갖는 것으로서, Gordon Bell은 이것도 이미 범용성을 잃는 것으로 보고 있다. 예로는 Flexible, Hypercube, Tandem Vax cluster,



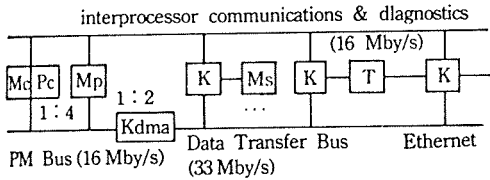
(a) "Unibus" type uniprocessor (PDP-II Unibus, Qbus, Multibus).



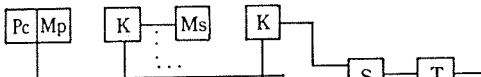
(b) Direct Pc Mp connection using modified "Unibus" with secondary memory and I/O control processors (Multibus, NCR Tower Plexus).



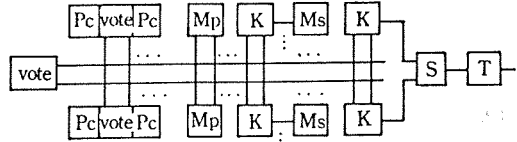
(c) Single bus for symmetrical multiprocessors, in which each processor has a dedicated cache (VME, Multibus II, Nubus, Elexsibus).



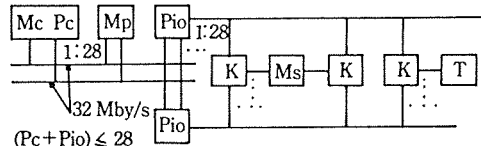
(d) Arete multiprocessor



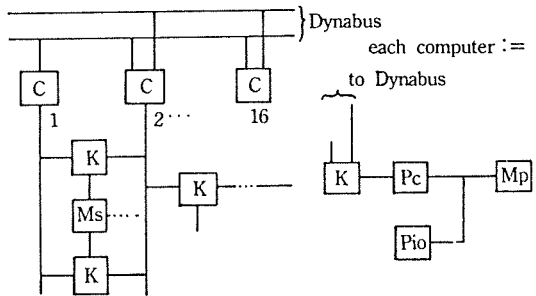
(e) Parallel computers (duplexed computer).



(f) Stratus fault-tolerant computer with voting



(g) Synapse N+1 symmetrical multiprocessor for high-availability and incremental, high-performance computation.



(h) Tandem Non-Stop II multicomputer.

Pc = central processor, Pio = i/o processor ;  
 K = control ; Mp = primary memory ;  
 Mc = cache ; Ms = secondary memory ;  
 and T = transduce (terminal).

그림 2. Common micro-and mini-computer structures.

Transputer 등이다.

다른 또 하나의 분류는 LAN 등의 cable connection에 의한 분산처리 그룹으로 Appollo Xerox Star, SUN 등이 있다. 이것은 대개 더 좋은 file 시스템을 share하기 때문에 그 곳에 붙어있는 workstation을 사용자가 얼마나 잘 쓰는가에 효율이 달려있다.

그러나 CPU의 load balancing만 가지고 시스템을 평가할 수는 없다. 예를 들면 데이터 베이스의 점증적인 증가가 요구되는 경우도 있기 때문이다. 이와 같은 것은 loosely-coupled 분산처리 시스템이 더 쉽게 적응되는 경우가 대부분이다. 이러한 경우는 Tolerant 시스템의 SBB (System Building Block) 등에 데이터 베이스 등이 증가하여도 시스템의 소프트웨어에 아무

변동을 필요로 하지 않는 경우이다. 이와 같은 것은 file 등 일반 resource를 하나의 view로 볼 수 있게 하는 분산 OS의 기술이 요구된다.

이와 같은 분류는 processing unit, memory, 그리고 controller간의 interconnection의 형태 (그림 2) 및 전송속도 등에 따라 이루어질 수도 있다.

Tightly-coupled 시스템에 대해서는 multiprocessor라는 말 대신 parallel processor라는 용어를 많이 쓰고 있는데 전형적인 시스템들의 종류 및 가격모습 등은 그림 3과 같다.

이와는 다르게 수직으로는 가격 등급을, 수평으로는 interconnection 방법 등으로 분류한 것도 있는데 이 분류에 의한 예는 그림 4와 같다.

이와 같이 많은 회사들이 그들 나름대로 많은

제품을 발표하고 있지만 그동안 사업적으로 실패하여 이미 문을 닫은 회사들이 있고, 또한 기술적인 우위가 사업상 우위와 항상 같은 것은 아니기 때문에 여러 모험회사들의 앞날은 예측하기가 어렵다.

아직 안정된 소프트웨어들이 대체로 부족하기 때문에 여러 다중 프로세서 하드웨어 시스템 및 기본 시스템 소프트웨어만 갖추고, 그 외의 소프트웨어는 다른 제 3의 공급자에게 위탁하거나 또는 그것을 개발하여 대량으로 쓸 손님을

Company	Product	Price	Parallelism	Connectivity	Memory	Processor
Accelerated Processors	Model 10	\$88,000	4 to 12 groves of 8 ALUs	reconfigurable	N.A	N.A
Alliant Computer Systems	F×18	\$270,000	up to 20	bus	global, up to 64 megabytes	64-bit CMOS gate array
Ametek	System 14	\$75,000 up	16 to 256	Hypercube	local, up to 256 megabytes	16-bit 80286/80287
Bolt, Beranek & Newman	Butterfly	\$40,000 up	1 to 256	switching system	shared and local	16-bit MC 68000
Deneicor	HEP 1	\$1 Million to \$3 million/execution module	1 to 16 execution modules	shuttle network	global	64-bit ECL
ELXSI	6400	\$600,000	up to 12	bus	global, with local cache, up to 800 magabytes	64-bit ECL gate arrays
Encore Computer	Multimax	\$114,000	up to 20	bus	global, local, up to 32 megabytes	32-bit NS 32032
Flexible Computer	Flex 32	\$150,000 up	up to 20/box, up to 2,480 total	bus	global, 98 megabytes/cabinet	32-bit NS 32032
Gemini Computers	Trusted Multiple	\$47,500 up	1 to 8	bus	shared, local, up to 128 megabytes	16-bit 80286
Intel Scientific Computers	IPSC	\$150,000 to \$520,000	32 to 128	Hypercube	local 288 magabytes	16-bit 80286/80287
International Parallel Machines	IP 1	\$50,000	1 mater processor, up to 8 processors	cross-bar like switch	global, up to 40 megabytes	32-bit
Loral Instrumentation	Dataflo	\$ 65,000 up	5 to 256 oata flow processors	bus	shared, local, up to 14.5 megabytes	16-bit NS32016
Meiko	Computing Surface	\$ 220,000 to \$ 300,000	up to 128	four nearest neighbors	local, 48-K bytes/ 4 processors	32-bit inmos T414 transputer
Multiflow Systems	N.A	VAX range	multiple register	N.A	global, more than 1 gigabyte	gate arrays
Ncube	Ncube/Teg	\$100,000 up	16 to 1,1024	Hypercube	local, up to 160 megabytes	32-bit custom VLSI
Saxpy Computer	Saxpx 1M	\$2 million	32 processors	parallel systolic	shared, up to 512-K bytes	32-bit coutom
Sequent Computer Systems	Balance, 8000	\$60,000	up to 12	bus	global, up to 28 megabytes	32-bit NS 32032
Sequoia System	Sequoia Systems	\$200,000	up to 64	bus	global, up to 252 megabytes	16-bit MC 68010
Thinking Machines	Connection Machine	N.A	64,000 to 1,000,000	Hypercube	global, 500 megabytes	1-bit custom

그림 3. MANY ARCHITECTURES CROWD PARALLEL PROCESSOR MARKET

System starting price	Bus-connected, Bus-connected, shared memory few big processors	Bus-connected shared memory, moderate to large number of medium-sized processors	Hypercube interconnection local memory, moderate number to many processors	Networked computers and work stations, local memory, small to potentially very large numbers of computers	others: such as data-flow, switched-connected processors and transputer-based machines
\$10,000 to \$100,000	Concurrent Computer	Masscomp 5700 5800  Sequent Balance 8000 Balance 21000	Ametek System 14	Apollo DW Series Sun SUN Series	BBN Butterfly Loral Data Flo LDF-100
\$100,000 to \$1 million	Alliant FX/8  Culler Culler 7 Digital Equipment VAX 8300 VAX 8800 VAX Clusters Ebsi 6400	Encore Multimax Flexible Flex-32 Teradata DBC/10/2	Intel Scientific Computers iPSC iPSC-VX Ncube Ncube/ten	Digital Equipment Networked VAXes and MicroVAXes	Floating-Point Systems  T Series Meiko Computing Surface Multiflow (Unamed)
\$1 million to \$10 million	Cray Research XMP Cray-2 Cray-3 Control Data Cyber Plus IBM 3090 Model 400	None available now	None available now	None available now	Thinking Machines The Connection Machine Myrias Research (Unamed)

그림 4. COMMERCIAL PARALLEL-PROCESSING COMPUTERS BY ARCHITECTURE

구하고 있는 경우가 많다. 즉 아직은 이 제통 회사들이 OEM(Original Equipment Manufacturing)으로 사업을 요구하고 방향을 추진하는 편이다.

그러므로 UNIX에 입각하여 tightly-coupled multiprocessor를 만드는 회사의 경우, 소프트웨어를 설계 실험하기 위한 학교, 연구소 또는 자사의 시스템을 쓰기로 한 특별회사 등에 공급을 많이 하고 있다.

반면에 특별 응용분야의 시장을 겨냥하여 특별시스템을 개발하는 경우도 많은데 On-line transaction 분야가 대표적이라 할 수 있다. 이 경우는, 기존 대형의 IBM이 차지했던 영역을 같은 성능이면서도 가격이 저렴한 다중 프로세서 시스템으로 경쟁하고자 하는 노력이다.

그러므로 기존 UNIX의 file 시스템 등을 많이 변경하고 DBMS 패키지도 그 시스템에 적합하게 변경함으로써 많은 transaction을 다중 CPU가 분담 처리하도록 설계하고 있다.

Parallel processing이 multiprocessing 보다 더 많이 나오는 이유, 즉 다중 프로세서 시스템

이 상업응용보다 과학응용 계산용이 더 많은 이유에 대해서는 여러 설이 있으나 기존 대형컴퓨터 메이커들이 이미 차지하고 있는 상업응용 시장을 침투하기보다는 과학계산용 시장을 겨냥하는 것이 우선은 더 쉽다는 의견이 많다. 즉, 기술적인 면에서는 거의 비슷한 입장이라는 것이다. 그러나 몇 년후 사업성이 보이면 상업응용면에서도 더 많은 다중 프로세서가 나올 것이라는 전망도 있다.

## 사례조사

이상 위에서 서술한 여러형태 중 몇가지 시스템의 예를 들어 구조를 살펴본다.

우선 bus에 입각하여 shared memory를 가지고 있는 tightly-coupled 시스템을 보자.

### 1. Sequent사의 Balance 21000

그림 5에서 보듯이 한 보드에는 두개의 CPU를 갖고 있고 이론적으로 15개의 보드까지 허용

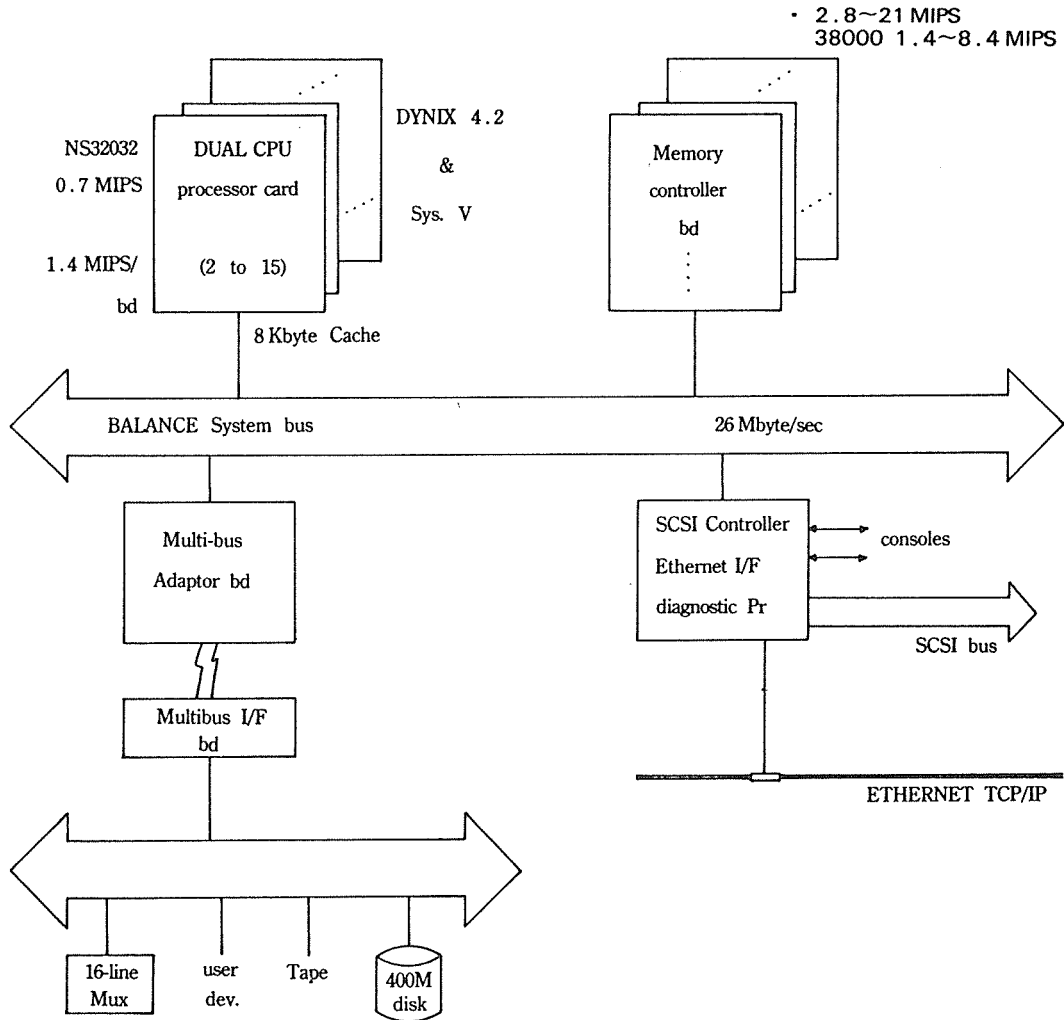


그림 5 Balance 21000

하므로 30개의 다중프로세서를 가질 수 있다. 각 board는 8 Kbyte의 cache를 가지고 있고 system bus의 속도는 20Mbyte/sec이므로 빠르다고 할 수는 없으나 SLIC (System Link Interface Chip)이라는 local bus 형태가 있어 버스의 contention을 줄여주고 있다.

기타 I/O를 위해서는 기존 표준형 multibus를 쓰고 있다. 운영체제는 UNIX 4.2 또는 UNIX System V 모두를 허용하는 DYNIX라는 것을 가지고 있다. CPU는 NS의 32032이다.

## 2. Encore사의 Multimax

위 예와 비슷한 형태를 가진 그림 6의 multi-

max 시스템은 100Mbytes/sec의 높은 전송 속도를 가진 고유의 nanobus를 가지고 있다. 한 보드에 두개의 CPU를 갖고, cache는 32Kbyte이다. 10개의 보드를 허용하므로 이론적으로는 20개의 CPU를 다중프로세서로 가질 수 있다. balance와는 달리 I/O 부분은 LAN cable에 I/O server 형태의 ANNEX라는 것을 가지고 있고, LAN 및 disk 등을 제어하는 카드가 다른 bus에 연결되어 있다. 그러므로 분산시스템으로 확장할 수 있는 더 융통성 있는 형태를 취하고 있다고 볼 수 있겠다.

CPU로는 위에서와 마찬가지로 NS의 32032, 운영체제로는 UNIX4.2 또는 UNIX System V중

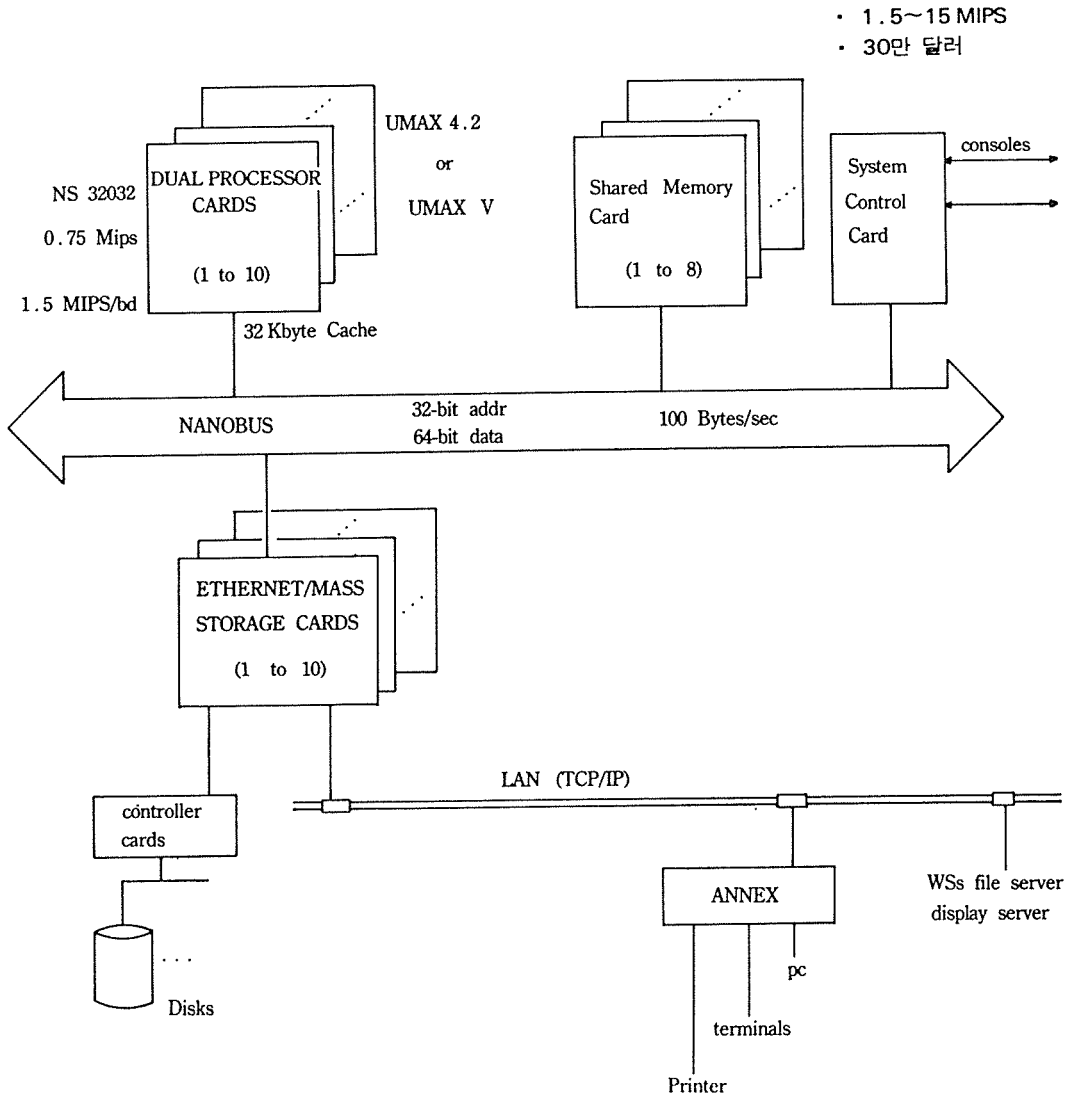


그림 6 Encore Multimax

하나를 선택해야 한다. 이름은 UMAX 4.2 또는 UMAX V라고 한다.

### 3. Masscomp사의 MC5700

위의 두 예가 처음부터 대규모의 다중프로세서 설계에서 시작했다면 Masscomp 시스템들은 단일프로세서로 된 engineering workstation으로부터 시작하여 다중프로세서로 발전한 것이다.

그림 7은 4개의 CPU 시스템을 보여주고 있는데 26.6Mbyte/sec의 memory bus에 8 Kbyte cache를 가지고 CPU는 Motorola의 68020 계열

을 쓰고 있다. 3계열의 계층적 버스를 쓰고 있고, 그래픽 및 numeric 데이터 처리를 위한 I/O들이 붙어 있다. 운영체계로는 UNIX System V 나 4.2 BSD와 호환하는 RTU라는 시스템이다.

### 4. EnMasse 시스템

지금까지의 예와는 달리 transaction processing을 위한 다중 프로세서로 EnMasse 시스템을 들 수 있다. 그림 8은 응용을 수직층으로 나누어 설계된 구조를 보여주고 있다. 즉, 사용자를 위한 터미널 등은 마이크로 프로세서 단위의 IOP



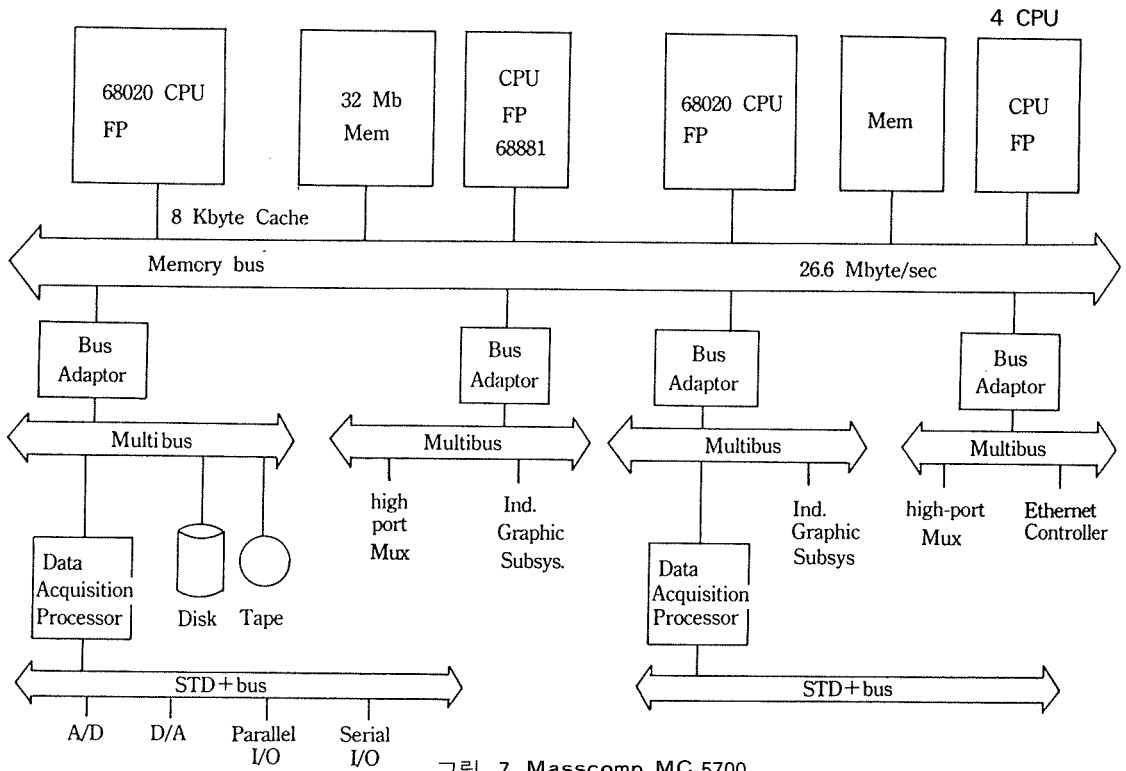


그림 7 Masscomp MC 5700

(I/O Processor)가 담당을 하고, 사용자 프로그램 즉, UNIX의 file 시스템을 제외한 O.S 및 user program 등은 AP (Application Processor)에서 작동을 한다.

이 AP는 Motorola 68000계열로 되어 있고, 24개까지 허용함으로써 전형적인 multiprocessing을 하게 되어 있다. 특히 DBMS 등을 처리한 file 시스템 등은 FP(File Processor)에 의해 처리되는데, 4개의 M 68000으로 다중프로세서 구조를 하고 있다. 즉 하나의 FP에 여러개의 AP를 연결하고 있다. 그러므로 bottle neck이 FP에 걸릴 수 있어 DB의 점층적 크기가 제한되는 것은 tightly-coupled 시스템과 같게 된다.

이 시스템은 file 시스템을 UNIX것과는 달리 EnMasse사가 transaction응용에 적합하도록 새로 설계하였고, COBOL과 결합된 응용이 IOP 등에 분산처리되도록 많은 소프트웨어를 개발하여 발전시켰다.

### 5. Tolerant 시스템

Transaction processing을 위한 응용 및 DB의

점층적인 증가 허용과 fault-tolerant 시스템을 위한 것으로 Tolerant사의 Eternity시스템을 들 수 있다. 그림8은 한 모듈이 SBB(System Building Block)으로 되어 있고, 여러개의 SBB가 Dual LAN cable에 연결되어 cluster를 이룬 것을 보여주고 있다.

한 SBB 내에는 NS 32032 CPU에 의한 프로세서들이 기능적으로 여러개 있다. UPU(User Processing Unit)는 일반 사용자 프로그램을 수행하고 RPU(Realtime Processing Unit)는 다른 remote resource(다른 SBB의 resource) 등을 user transparent하게 보여주기 위한 부분의 수행을 담당하고 있다. 한 SBB내에 I/O 처리를 위해 IOP가 있고 I/O bus를 통해 disk 및 터미널이 붙어 있다. 특히 터미널들은 CIP(Communication Interface Processor)에 붙어 있어 CIP의 CPU에 의해 여러기능을 제공받을 수 있다.

이와 같은 SBB들이 서로 Symmetric한 입장에서 그룹을 이루고 TX라는 운영체제(UNIX에 기반)가 모든 자원을 하나처럼 보이게 하는 분

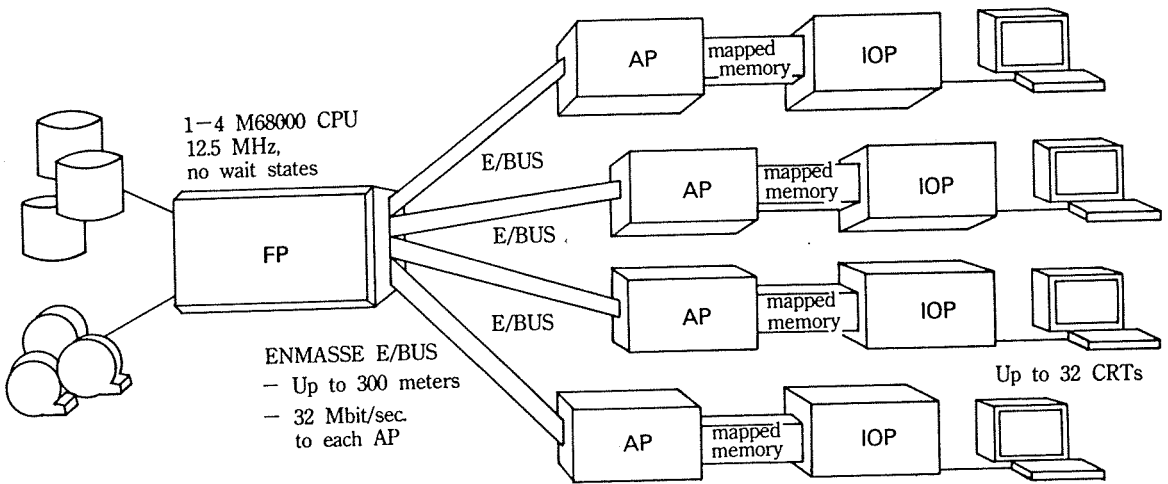


그림 8 ENMASSE Architecture ENMASSE E/BUS

산처리 시스템을 갖추고 있다. 시스템 구조는 fault-tolerant를 위해 이중으로 되어 있는데, 이 회사의 궁극적인 transaction processing은 이와 같은 fault-tolerancy에 의해 이루어질 것으로 믿고 있다.

### 6. 기타

이외에도 메인 프레임급의 다중프로세서로

ELXSI가 있다. 프로세서를 ECL 조직으로 속도를 높이고 시스템 버스도 300Mbyte/sec 이상으로 고속화함으로써 어느 시스템보다도 상위계층을 제공하고 있고 자체운영체제 (EMBOS) 및 UNIX 계열도 제공하고 있다.

반면에 4 개 정도의 M68020 CPU에 의한 다중 프로세서나 상업응용에 쓰이기 위해 소프트웨어 개발을 주력하고 있는 TI의 BS1500 시스

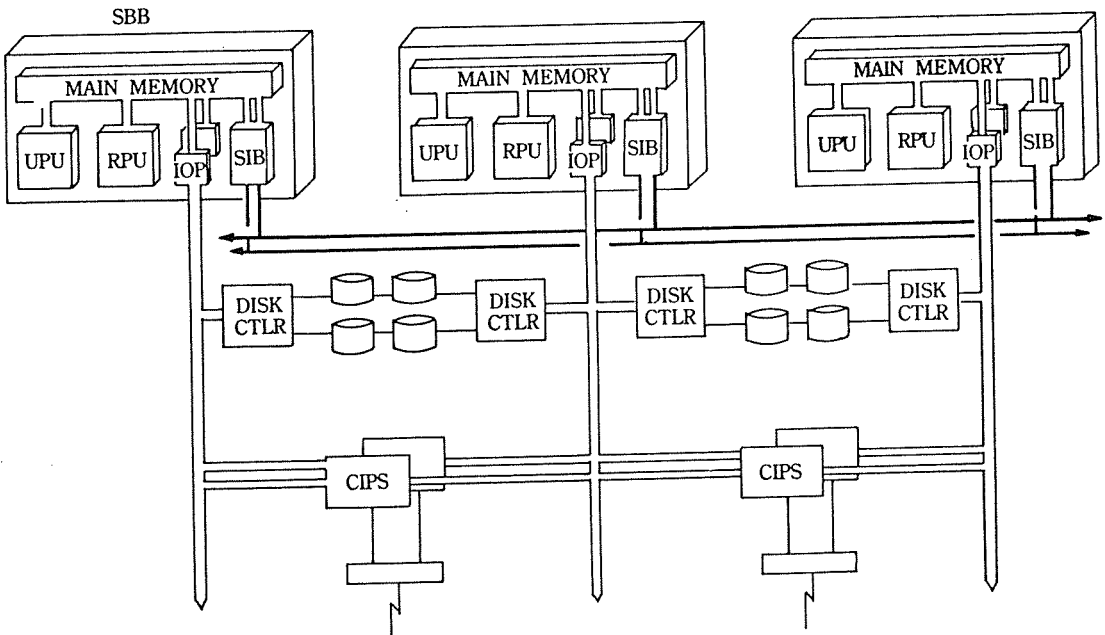


그림 9 DISTRIBUTED CONFIGURATION

(P. 42로 계속)

것이다. 일본적인 「文化」를 日本式 經營으로서 그들에게 강요하는 것이 아니라 오히려 그러한 방법을 취하면 반드시 팔린다는 데에 철저를 기하고 있다. 더우기 이의 실현방법으로서 어떠한 직종에서도 같은 임금이라는 규칙을 강력히 내세운다. 이것이 미국인 종업원으로서는 新鮮한 「文明」으로서 받아들이는 것이다. 따라서 日本式 經營이라는 것도 「文化」로서 추진하는 것이 아니라 「文明」 Level에서 실시해 나간다면 앞으로 충분히 국제적인 모델이 될 수 있다고 생각한다.

美国社会学会에 나가고 심포지움에도 참가했으나 그때 참석자의 한사람인 하버드 대학의 에즐러 보겔 교수는 昨今の 美日經濟關係를 생각한다면 脱工業化社会에서 지금이야말로 미국경

제는 逆轉하여 植民地化되어 버렸다고 개탄했다.

Post Industrial Reverse Neo Colonialism이라는 것이다. 물밀듯이 일본으로부터 자본이 들어오고, 工場 進出 또한 왕성하다. 더우기 미국이 팔 수 있는 것은 農産物과 같이 1 차産品밖에 없는 상황이고 보면 단지 植民地經濟라고 볼 수밖에 없다.

그러한 現實的인 問題는 차치하고라도 아뭏든 日本式 經營이라는 것은 「文明」의 수준에서는 충분히 通한다는 判断이 필요하다고 생각한다. 「文明」은 보다 普編的인 性格을 가진, 物的인 Hard의 側面이며 「文化」쪽은 Soft한 側面인데 이러한 것을 組合하여 생각해 나감으로서 새로운 「文明」으로서의 日本式 經營이 충분히 성립된다고 생각한다.

## P. 36에서 계속

램도 있다. 이것도 UNIX에 기반을 두고 있다.

이와 같은 시스템의 부분별 비교는 그림 9와 같다.

## V. 결론

지금까지 여러개의 프로세서를 가진 시스템도 구조와 응용이 제각기 다르다는 것을 살펴 보았다. 다중프로세서를 이용한 기술자체는 점점 성숙하여 성능 및 가격면에서 기존 컴퓨터 시스템보다 나아지고 있다. 다만 다중프로세서의 시장개척 등에는 사용자들의 인식 및 계몽 등 정립되어야 할 부분이 아직은 남아 있다.

후발주자로서 선진국 수준의 컴퓨터 개발에도 전해야 할 우리나라 입장에서 볼 때, 마이크로 프로세서에 기반을 둔 다중프로세서 시스템분야는 우리에게 전망이 매우 밝다고 할 수 있겠다. 특히 이 분야 기술의 핵심인 버스 및 cache 등의 하드웨어와 다중프로세서 운영체제 기술 및 Concurrent 프로그램 기술 등에서 요구된다.

## 참 고 문 헌

1. Omri Serlin, "Defining Parallel Processing PART 1," UNIX/World, Jan. 1986.
2. Robert Lineback, "Parallel Processing : Why a Shake out Nears, Electronics, Oct. 28. 1985.
3. B. J. Fuller, "Multiprocessing : UNIX Runs Business-Oriented Supermini." UNIX/World, June 1986.
4. Gordon Bell, "The Mini and Micro Industries." IEEE COMPUTER, Oct. 1984.
5. Tom Manuel, "Parallel Processing : The Pace Quickens," Electronics, Sept. 18, 1986.
6. T. J. LeBlanc, "Shared Memory vs Message Passing in a Tightly-Coupled Multiprocessor : A Case Study," Proc. of 1986 Int'l Conf. of Parallel Processing, 1986.
7. "Interview with Gordon Bell." UNIX Review, Feb. 1986.
8. 기타 각 multiprocessor 회사들의 catalog.