

吳 吉 祿

韓國電子技術研究所  
컴퓨터 연구부장 / 工博

# IEEE 896과 새로운 32Bit Backplane Bus

“  
새로운  
32-Bit Bus는  
Bus의 데이터 전송이  
포화되었던 문제를 해결했고  
Processor間的 정보교환  
문제(데이터의 호환성,  
Synchronization, etc)  
등을 해결하고  
있다.”

이제부터 컴퓨터 기술 개발 강좌 6호

## 1. 서 언

IEEE Computer Society에서는 마이크로 프로세서와 관련한 분야의 표준화를 위하여 1977년에 Microprocessor Committee를 조직하였다. Committee에서는 컴퓨터 시스템 개발에 관련한 여러 가지의 표준을 발표하였는데, 그중 IEEE 696(S-100) (9)와 IEEE 796(Multibus) (10)이 Backplane bus의 표준이다. IEEE 796을 발표한 후, Backplane bus의 표준화를 위하여 Subcommittee가 따로 조직되었으며, 향후 5~10년 동안 마이크로 프로세서를 중심으로 한 시스템에 사용할 버스의 표준을 연구하기 시작했다. 같은 해 유럽의 EDISG (European Distributed Intelligence Study Group)에서도 Subgroup을 조직하여 Backplane Bus의 연구에 함께 참여하였다. IEEE 896이라고 명령된 이 표준은 종래의 Bus가 지닌 여러 문제점의 해결과 새로운 시스템에서 Bus가 갖추어야 할 기능의 추가 등을 위한 많은 제안들이 만들어졌다. 현재, 어떤 System Architecture 에든지 사용할 수 있는 32-Bit Bus로 어드레스와 데이터의 Multiplexing, Handshake 전송방법, 분산 Arbitration, Multiprocessor 지원, Serial Interprocessor Link 제공 등의 특성을 갖는 것으로 굳어지고 있다. 그러나 전송 Protocol에 따라 Futurebus 라고 부르는 비동기형 (asynchronous Type)과 동기형 (Synchronous Type)인 IEEE 896.2의 두가지가 제안되고 있다.

Backplane Bus는 ISO의 Computer Communication을 위한 7 Layer Model의 Physical Layer와 Data Link Layer의 구현이며, 또한 그위의 소프트웨어에 관련되는 Layer의 구현에도 관련이 있으므로 Bus의 기능이 시스템 전체에 미치는 영향은 매우 중요하다. Bus의 데이터 전송 속도는 시스템의 성능에 직접적인 영향을 미치며, 전송 Protocol과 Arbitration 방법은 사용하는 프로세서의 신호 특성과 밀접한 관계

를 갖는다. 또한 전송 데이터의 폭은 프로세서의 데이터 형과 메모리의 Alignment에 영향을 준다. Backplane Bus 상의 Serial Data Link는 프로세서간의 새로운 Message 전송 통로로서, Message의 Protocol과 기능 등을 정의하여 ISO의 소프트웨어에 관련된 Layer 까지 구현 가능하므로 Multiprocessor 환경에 많은 지원을 할 수 있다. 버스 신호의 전기적인 특성은 신뢰도에 밀접한 영향을 주며, 기계적인 특성(Bus 모듈의 크기 및 Connector 특성 등)은 시스템의 신뢰도와 외형적 형태를 결정한다.

本稿에서는 Bus를 중심으로 하는 시스템을 설계할 때 고려해야 할 사항을 중심으로 Bus의 특성을 설명하며, 새로운 32-Bit Bus의 표준화에서 고려되었던 문제들과 그 해결 방안에 대해서 설명한다. 현재 32-Bit Bus의 표준을 겨냥하고 등장하고 있는 32-Bit Bus들의 개별적인 특성과 비교를 통해서 앞으로 Bus를 중심으로 하는 시스템의 방향을 전망해 본다.

## 2. Bus의 특성

### 2.1 데이터 전송

Bus의 가장 주된 기능은 데이터를 전송하는 것이다. 어떤 방법으로 주어진 시간 안에 많은 데이터를 전송할 수 있느냐 하는 것이 Bus의 성능을 좌우한다. 전송 Protocol은 데이터를 전송할 때, 버스 Master와 Slave가 임의의 Clock에 의해서 제어되느냐 아니면 양쪽 제어 신호의 Edge에 의해 되는가에 따라 前者를 동기형, 後者를 비동기형 방식이라고 부른다. 비동기형 방식은 Bus상의 각 모듈의 Access Time이 전송 Protocol과 전송 속도에 제한을 주지않고, 앞으로 기술(Bus에 사용되는 부품의 성능 향상 등)이 발전함에 따라 더욱 빠르고 안전한 전송 속도를 얻을 수 있다는 장점을 갖는다. 반면에 동기형 방식은 장래성은 없지만 현재 사용중인 Bus의 모듈속도를 최대한으로 이용할 수 있으며, Clock에 의해 정해진 순간에만 Switching이 일어나므로 Noise 발생이 각

제어 신호의 Edge에서 Switching이 일어나는 비동기형보다 적다. 또 모든 제어가 Clock에 동기하기 때문에 State Trace Tool을 사용하기 쉽기 때문에 Debugging이 용이한 장점을 가지고 있다. 그러나 Clock을 구동하는 모듈 등 Bus를 전체적으로 제어해야 하는 모듈이 따로 필요하기 때문에 Clock Skew 방지를 위한 Bus 모듈의 위치에 대한 제약성과 신뢰도, 확장성의 단점이 있다.

Block 전송 모드는 Bus의 데이터 전송 속도를 극대화 시키기 위한 것으로 우선 Bus 상의 각 모듈은 자신의 Local 메모리가 필요하다. Bus의 제어를 얻은 Master는 일정한 시간동안 Bus를 독점하여 Block 단위의 전송을 한다. 이때 Master는 시작점의 어드레스만 주고 Slave에서 매 데이터 전송마다 순차적으로 어드레스를 증가시키는 방법을 사용하여 전송속도를 증가시킨다. Block 전송 모드의 장점은 공통 Bus가 갖는 제약점인 Arbitration Time, 데이터와 어드레스의 Setup Time 등을 줄임으로 전송속도를 올릴 수 있다.

### 2.2 Arbitration

공통 Bus를 사용하는 경우 데이터의 전송을 할 때 항상 하나의 Master가 전송을 주도해야 한다. 그러므로 Bus 상에 전송을 원하는 Master가 2개 이상 존재할 때는 제어를 맞을 한개의 Master를 Arbitration 과정을 거쳐 고르게 된다.

Arbitration을 담당하는 부분은 제한된 신호를 이용하여 Bus의 요구와 동시에 빠르고 정확하게 하나의 Master를 선택하여야 한다. 중앙 집중형(Centralized)은 Bus 상에 한 곳에 제어 Logic이 모여있는 형태이고, Bus를 요청할 수 있는 모듈(Bus Master) 각각이 제어 Logic을 갖고 함께 동작하는 분산형(Distributed)이 있다. 각각은 자신의 장단점을 갖고 있지만, 표준 Bus에서는 제어회로가 고장을 일으켰을 때 전체 시스템에 미치는 영향이 작은 분산형을 많이 이용하고 있다. 중앙 집중형으로는 Versabus와 Unibus 등이 있고, Multibus와 IEEE 896 등은 분산형에 속한다.

제어 Logic 의 분포와는 별도로 Bus 요구 신호의 구성에 따라 Serial과 Parallel로 구분할 수 있다. Serial 형태는 요구 신호를 Daisy Chain 형태로 연결하여 Backplane 의 Slot 위치에 따라 Request Priority가 결정되도록 구성한 것을 말한다. 적은 신호 수로 구현할 수 있는 장점이 있으나, Priority가 높은 모듈이 Bus를 독점할 수 있는 단점을 지니고 있어 작은 시스템(Bus Master가 1~3개 정도의 시스템)에서 많이 이용한다. Parallel 방식은 각 모듈이 자신의 요구 신호를 갖고 Bus를 요구하는 방법으로 line에 Priority가 주어지는 방법(Priority)과 각 Line에 공평한 기회가 주어지는 방법(Fairness)이 있다. 현재 대부분의 버스들은 Serial과 Parallel, Priority와 Fairness를 동시에 사용할 수 있도록 설계되어 있다. 이 밖에 몇 개의 모듈까지 제공할 수 있는가도 Arbitration Logic에 따라 결정된다.

Local 메모리의 크기는 Bus의 전송 속도와 모듈의 기능에 따라 전체 시스템의 성능에 큰 영향을 미치므로 시스템의 응용에 따라 적절히 결정해야 한다. 또 한편, 버스의 독점에 의한 Deadlock이 발생할 수 있으므로 Time-out이나 Priority의 순환 등의 방지책이 강구되어야 한다.

Mutual Exclusion(Process Synchronization) 문제 해결을 지원하기 위해 Read-Modify-Write를 한 데이터 전송 주기에 수행할 수 있

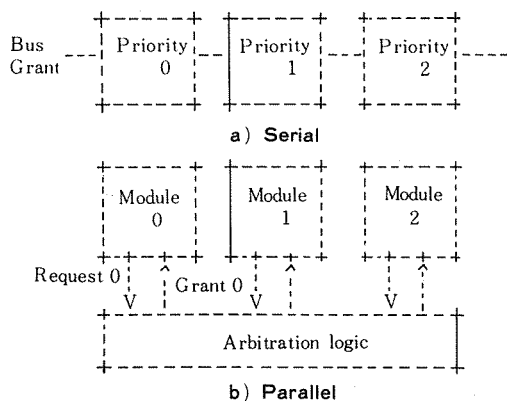


그림 1. Arbitration Scheme

는 전송모드를 제공하는 Bus가 많다. 즉 Test-and-Set 명령을 수행중 다른 Bus 모듈이 끼어 드는 것을 방지하는 전송 모드이다. Broadcast/Broadcall 전송 모드는 하나의 Master와 여러 개의 Slave가 동시에 데이터를 전송할 수 있는 기능으로 시스템의 Initialization이나 Diagnostic 등 이용 범위가 넓다.

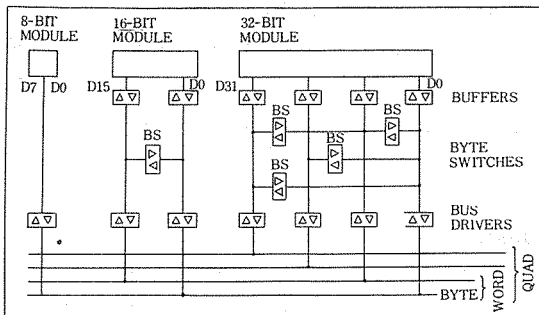
### 2.3 Bus Justify

Bus의 크기는 Bus Width와 Bus Data Unit로 나타낼 수 있다. Bus Width는 그 Bus가 한 번의 데이터 전송 주기에 전달할 수 있는 가장 긴 데이터를 말하고, Bus Data Unit는 가장 작은 것을 말한다. 결국 Data Unit의 정수 배가 Bus Width가 되는데, Bus Width보다 작은 데이터를 전송할 때 그 데이터를 Bus 데이터 통로의 왼쪽이나 오른쪽으로 맞추어서 항상 지정된 통로만을 이용하도록 하는 것이 Bus Justification이다. 이런 제한이 없이 데이터의 크기에 무관하게 전송할 데이터의 번지수에 맞추어 Bus의 통로를 선택할 경우를 Non-Justify 혹은 Straight Bus라고 한다.

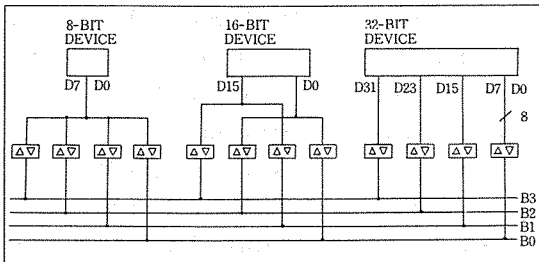
Multibus(IEEE 796)는 Byte Justify Bus이며, 32-Bit Versa/VME Bus는 Word Justified Bus이다. 그러나 16-Bit Versa/VME Bus는 Byte 전송에 지정된 통로를 사용하지 않는 Non-Justified(Straight)이다.

시스템 안에서의 Justify와 Straight의 차이는 그림 2에서 알 수 있듯이 데이터의 크기가 다른 Bus 모듈이 같은 Bus 위에 있을 때, 서로의 정보교환을 위해서 필요한 Bus Data Unit(보통 Byte) Switch가 어느 곳에 있느냐로 집약할 수 있다. Justify는 데이터 크기가 작은 모듈들로 구성된 시스템에 큰 데이터를 갖는 새로운 모듈을 첨가하려고 할 때, 기존의 것의 변형없이 사용할 수 있다는 장점을 갖고 있지만, Byte Switch에 필요한 제어 신호를 출력하지 않는 프로세서(TI9900 and LSI-11)와는 같이 사용할 수 없다. 시스템을 이루는 모듈들의 구성에 따라 Justify와 Straight Bus를 선택해야 하는데, 작은 데이터 크기를 갖는 모듈을 중심

으로 확장하려고 하는 시스템의 경우 前者가 유리하며, 큰 데이터를 갖는 모듈이 주를 이루고 있는 시스템에서 특별한 필요에 의해 소량의 작은 데이터를 갖는 모듈을 같이 사용할 경우는 後者が 유리하다. 또한 시스템에서 주로 다루는 데이터의 크기에 따라 Justify와 Straight를 골라야할 필요도 있다. 또한, 버스의 Justify는 Bus 상의 데이터 뿐만이 아니라 메모리의 Align에도 영향을 주기 때문에 다음에 설명할 메모리를 공유하는 Multiprocessor 시스템에서 데이터의 호환성 문제에도 중요한 영향을 미친다.



a) Interfacing 8-, 16-, and 32-Bit Modules to a Justified Little-endian Bus



b) Interfacing 8-, 16-, and 32-Bit Modules to a Straight Little-endian Bus

그림 2. Justified bus and Straight Bus

## 2. 4 Multiprocessor 지원

IEEE 896 Bus는 Tightly Coupled Multiprocessor System을 위한 표준 Bus로써 서로 다른 프로세서들이 메모리를 공유할 때 원활한 정보교환을 할 수 있게 하는 것을 기본 설계목표로 하고 있다. Multiprocessor 시스템을 위한 Bus의 규격은 여러 모듈들이 다른 기술적인 개념을 갖는 다른 제조사들이 만든 프로세서들을 사용할 때 그들의 호환성(Different Speed, Signal Protocol, Data Format 등)을 보장해야 한다.

2. 1과 2. 2의 Protocol Layer에서는 어떻게(how) 정보를 교환할 것인지를 정의하여, 모듈 사이의 다른 속도와 다른 신호의 문제를 해결할 수 있는 Bus의 Protocol에 대해 언급했다. Multiprocessor 환경에서의 호환성을 유지하려면 어떤(what) 정보를 교환해야 하는지, 즉 프로세서 사이에 교환할 Data Format을 정의해야 한다. 그러나 현재까지 프로세서들 사이에 공통된 데이터 표현 형식이 없고, 또한 각 제조사들은 자기의 것을 고집하고 있기 때문에 여러 가지의 다른 프로세서를 사용하는 Multiprocessor 시스템에서 메모리 상의 데이터는 그 데이터를 저장한 프로세서를 알기 전에는 해석이 불가능하다. Multiprocessor 시스템에서 데이터의 호환성 문제를 해결하기 위해 각각의 프로세서는 자기 자신의 Compiler를 갖으며 프로세서 사이에 정보교환을 할 때는 Mailbox나 Port와 같은 공통 Format을 갖는 Data Structure를 이용하는데, 이 방법은 Compiler끼리 서로 다른 것의 특성을 알지 못하기 때문에 서로의 호환성을 고려하다 매우 비효율적인 구조를 갖기 쉽다. (2) Bus 인터페이스 수준에서 데이터형의 호환성에 대한 지원을 하여 향상된 성능의 Multiprocessor 시스템을 보장하는 것이 IEEE 896을 비롯한 최근의 32-Bit Bus의 시도이다. Bus 인터페이스 수준에서 데이터의 호환성을 유지시키기 위해서 고려할 점은 사용할 프로세서들이 Instruction으로 직접 다룰 수 있는 데이터형과 메모리 Align 문제, 그리고 Bus 상의 데이터 형(Justify) 등이 있다.

다음은 프로세서의 데이터 형과 Bus 상의 데이터형, 메모리의 데이터 형 등에 대해 알아보고 이들 문제를 해결하는 방향들에 대해 설명한다.

프로세서의 데이터 형(Data Type) Instruction에서 취급하는 데이터 형태는

Instruction에서 취급하는 데이터 형태는

Instruction에서 취급하는 데이터 형태는

Byte, 16-Bit Integer, 32-Bit Integer, 어드레스, Floating-point, BCD 그리고 복잡한 형태의 갖은 Array, String 등이 있다. 위에 열거한 데이터 형들은 8080은 2개의 데이터 형(Byte, Word) 밖에 없었던 것이 프로세서의 기능이 다양해짐에 따라 늘어나 VAX는 10개, 최근의 68020도 7개의 데이터 형을 갖고 있다.

표현 형식은 크게 LE (Little Endian) 형과 BE (Big Endian)로 나누어지며 기준은 작은 쪽 어드레스에 Least/most Significant Byte (bit)를 위치시키느냐에 따라 구분된다. 그림 3은 LE 형 프로세서의 Bit 표현 방법이며, 마이크로 프로세서에서 많이 사용되고 있다. 그림 4는 16-Bit Integer 가 메모리에 저장되는 것을 나타낸 것으로 그림에서 알 수 있듯이 프로세서마다 다른 표현 방법을 사용하고 있다. 32-Bit Integer 나 Floating-Pointer Data, BCD 들은 더욱 표현 방법들이 다양하여 LE와 BE의 차이 뿐만 아니라 같은 제조사의 프로세서들끼리도 일치성을 잃은 것도 있다. Coprocessor의 등장 등 날로 다양해지는 데이터 Format의 표준화는 시급히 고려되어야 할 문제이다.

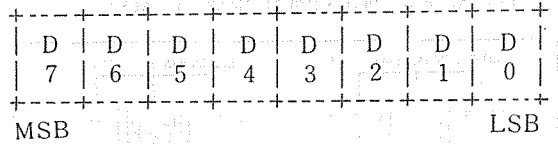


그림 3. One byte representation in little-endian notation

Data format on parallel bus

메모리 상의 데이터는 프로세서의 데이터 형태 뿐만 아니라 Bus의 특성과 프로세서의 Architecture에도 좌우된다. Bus의 Data Size와 Justify 그리고 Memory Alignment를 고려해야 한다. Memory Alignment는 메모리와 Bus 그리고 프로세서 사이의 전송을 간단히 하기 위한 메모리 상의 데이터 표현에 대한 프로세서의 제한이다. Byte Align된 메모리를 사용하는 프로세서의 경우, 모든 데이터는 그 크기에 관계없이 어떤 번지수에도 위치할 수 있지만 Fully-Align된 경우는 각 데이터들은 자신의 크기의 배수가 되는 번지수에만 저장되어야 한다.

현재 NS16000과 8086을 제외한 마이크로 프로세서들은 Word-Align된 메모리를 갖는다.

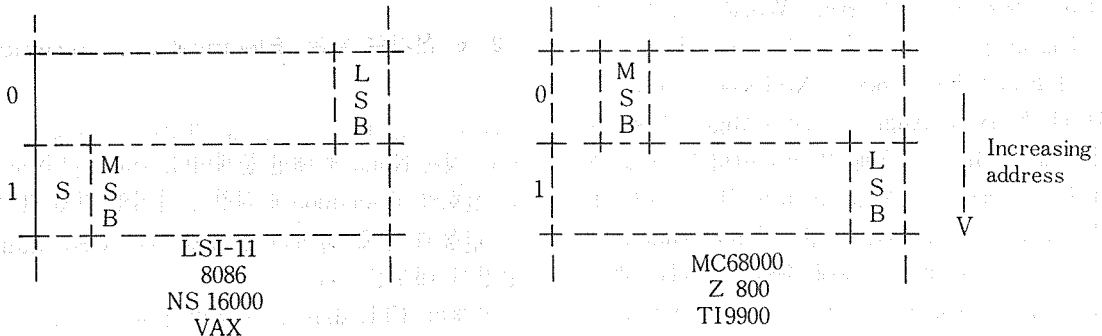


그림 4. Comparison of the several microprocessor.s a word representation in memory

Align된 시스템의 경우 메모리의 낭비가 생기고 Compiler와 Assembler가 복잡해지는 단점이 있지만, 인터페이스 회로가 간단하고 수행 속도가 빠른 장점과 특히 Multiprocessor 환경에서의 데이터 전송의 호환성 문제에 있어서 해결하기 쉬운 장점이 있다.

Processor에서 나오는 데이터는 모두 Non-

Justified (Straight) 형태를 갖으며 Straight Bus는 Bus의 데이터 통로가 메모리의 직접적인 확장으로 Fully-aligned 프로세서를 사용할 경우 LE와 BE를 동시에 접속시킬 수 있다. 이때 BE 측은 그림 5와 같이 Byte Switch가 있어야 하고 TI9900과 LSI-11은 Byte Switch를 제어할 수 있는 신호가 없기 때문에 같은 데

이터 형태를 메모리에서 얻을 수 없다.

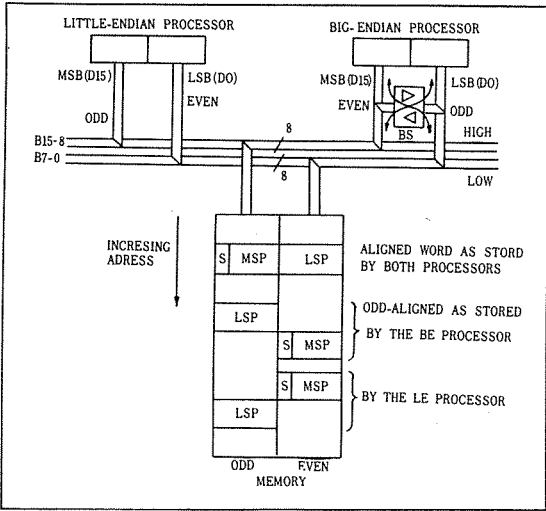


그림 5. To store words consistently over a straight bus

지금까지 프로세서의 Data-Type, Memory-Alignment 그리고 Bus Justification 을 알아본 결과 16-Bit BE 프로세서들 (MC68000, Z8000, 그리고 TI9900) 은 Word-Align 되어 있고 TI 9900을 제외하고 모두 Byte/Word 신호가 있어 LE 환경에 쉽게 적응할 수 있게 되어 있지만, LE 프로세서들 (8086, NS16000, 그리고 LSI-11) 은 Byte-Align (Pseudo-Align) 되어 있기 때문에 BE 시스템에 적응이 어렵기 때문에 여러 프로세서를 지원할 수 있게 하기 위해 LE 형의 Bus가 적절하다. 또한, 종전의 Bus 들이 Bus 상에서의 데이터 형태 뿐만이 아니라 메모리의 데이터까지 정하는 필요 이상의 규격을 배제하고 여유있게 (Loosely) 정의해야 한다. 이렇게 함으로써 여러 프로세서가 메모리를 공유하는 Multiprocessor 시스템에서의 데이터 호환성을 유지할 수 있다.

### 2.5 Serial Data Link

마이크로 프로세서를 중심으로 하는 Multiprocessor 시스템이 일반화됨에 따라 Backplane Bus 상에 Serial Data Link의 필요성을 인식

하게 되었다. IEEE 896을 비롯한 새로운 32-Bit Bus들이 제공하는 Serial data link는 Parallel Bus와는 독립적인 Clock 에 동기시켜 1~3 Mbit/sec의 전송속도를 가지며, 작은수의 Line (Clock과 데이터가 각각 1개씩)으로 데이터의 전송이 가능하다. Arbitration을 위한 별도의 제어 Line이 없이 CSMA/CD (Multibus II)나 Collision Tolerate (VMEbus) 등의 방법을 사용한다.

주로 프로세서 간의 정보교환의 통로로 사용되는데, 메모리를 공유하는 Tightly Coupled인 경우 Message의 Pointer가 전송되며 Loosely Coupled인 경우는 Message가 Packet 단위로 나누어져 전송될 수 있다. 소프트웨어에 관련되는 주요 Protocol Layer는 사용자 (System Designer)가 정의할 수 있도록 되어 있지만 Multibus II의 경우 미리 Bus 규격에 정의된 것도 있다. 사용자가 정함에 따라 Fault-tolerant 시스템에서의 Fault-isolation Path, Intelligent Semaphores, Broadcasting 그리고 Simultaneous Polling 기능 등 다양하게 이용할 수 있다.

### 2.6 전기적 특성 (Electrical Characteristics)

Bus의 전기적인 특성 중 가장 중요하게 취급되는 것은 Noise에 대한 문제이다. Bus상의 Noise는 신호의 Transition에 의한 순간적인 전압 전압의 변동과 신호 사이의 간섭에 의한 Crosstalk 현상이 대부분이다.

종래의 TTL driver는 용량성 부하를 구동할 때 신호의 Transition이 있을 때, 소비 전력의 변화에 의한 Noise의 발생과 반사파에 의한 Ringing 그리고 Crosstalk 등이 발생하기 때문에 안전한 제어 신호와 데이터의 전송을 위해서는 신호가 안정될 때까지 기다린 후 사용해야 한다. Noise는 빠른 회로를 사용할 때 더욱 많이 발생하므로 요즘 새로 설계되는 Bus나 기존의 Bus를 개량하는 Bus들은 Noise를 줄이는데 많은 고려를 하고 있다.

여러 방법들이 제시되고 있는데 인터페이스 회로에 들어가는 Driver를 특별하게 설계하여 특성을 개량하는 방법이 많은 효과를 얻고 있다. 특성의 개선은 신호 전송을 하지 않을 때 Reverse-bias Capacitance를 작게 하여 Bus의 용량성 부하를 줄여서 Ringing을 줄이고, 소비 전력을 감소시키며, 구동 능력을 향상시키는 데 주력한다. Futurebus와 Multibus II에서 사용하는 방법으로 어드레스/데이터를 Multiplex하여 동시에 Transition이 일어나는 Device (Transmitter/Receiver) 수를 줄이며, 제어 신호를 Clock에 동기시켜 일정한 순간에만 Transition이 발생하게 하여 Noise Immunity를 높이는 방법(Multibus II) 등을 사용하고 있다. TTL Driver를 사용할 때 Voltage Swing이 0 ~ 5V가 되는데 반해 Future Bus는 새로운 인터페이스 Chip (National Semiconductor's DS 3896)을 사용하여 1~2V 정도로 Swing 폭을 줄임으로써 전력 소비를 줄여서 전원 Transition에 의한 Noise를 줄이는 방법을 사용하고 있다.

## 2.7 기계적 특성 (Mechanical Characteristics)

Bus의 기계적 특성은 Connector와 Board의 크기 등으로 생각할 수 있다. 현재 등장하는 32 Bit Bus들은 IEEE 696과 IEEE 796의 Edge Connector보다 신뢰도가 높은 DIN Connector를 많이 사용하고 있다. Board의 크기는 IEEE 796보다 작은 Eurocard (100mm×150mm)를 기준으로 Modular하게 확장할 수 있게 하여 보드 크기에 대한 유연성을 최대한 보장하고 있다. (IEEE 896). 그러나 전체적으로 Board의 크기는 줄어들어 VLSI Technology 등 집적도의 향상 등의 기술적인 배경을 대변한다. 또한 32-Bit Bus이지만 16-Bit 시스템의 호환성 및 전망을 고려하여 16-Bit 시스템이 32-Bit의 반절의 크기로 인터페이스할 수 있도록 하는 것도 고려하고 있다.

## 3. Case study

### 3.1 Futurebus

IEEE Futurebus Committde에서 연구하고 있는 32-Bit Parallel Backplane Bus로써, IEEE 896 표준 중에서 비동기형 Bus를 말한다. 현재 Draft 6.2가 발표되었으며 가장 기본적인 설계 목적은 기능면에서 획기적인 향상과 성능면에서 Multiprocessor를 지원하기에 충분하도록 하는 것이다 [14]. 32-Bit 데이터와 어드레스가 Multiplex되어 있고 32-Bit 데이터 전송에 최적화되어 있다 (Non-Justification). 8-, 16-, 24-, 32-Bit 데이터 전송이 가능하며, Interlocked Operations, Broadcast Transfer 그리고 Block-Transfer 모드를 가지고 있다. 전송 Protocol은 Fastbus (IEEE 960, U.S. NIM Committee for high-energy Physics Community)의 변형으로 유연성을 고려한 Single-Transfer Mode와 속도를 고려한 Block-transfer Mode 그리고 여러 개의 Slave와 동시에 정보 교환이 가능한 Broadcasting Mode가 5 Bit Command Code로 제어된다.

Single Transfer Mode는 Read-Modify-Write Cycle을 제공하고 있으며, Block-transfer Mode는 시작 어드레스만 Master에서 제공되며 Slave에서 매 전송 때마다 어드레스를 증가시킨다. 비동기 방법을 사용하고 있기 때문에 현재는 40Mbyte/sec의 전송 속도를 갖지만, 곧 60 Mbyte/sec가 가능하며 그 이상의 속도도 인터페이스 기술이 발전함에 따라 가능하다. Broadcasting Mode는 몇 개의 Slave와 동시에 전송이 가능한 것으로 Read할 때는 마스터에서 각 Slave의 데이터가 논리 OR된다. Multiprocessor 환경을 지원하기 위하여 버스의 각 line에 Byte 순서를 정의하지 않고 LE와 BE 프로세서를 모두 지원하고 두 개의 프로세서 간의 정보 교환 시에는 LLE Format을 추천하고 있다. Serial Data Link도 지원하지

만 아직 Message의 Format과 기능의 정확한 규격은 발표되지 않고 있다. 32-Bit 모듈에 최적화 하기 위해 Nonjustification(Straight) 형태를 사용 8-이나 16-Bit 모듈에 Byte-Switch를 달게 하였다. Locking Mechanism은 Multi-

processor 환경에 또 하나의 중요한 기능으로 Mutually Exclusion 동작을 확실하게 할 수 있다. 이 기능은 Test-and-Set을 위한 Read-Modify-Write와 하나의 프로세서가 여러 개의 버스 모듈을 동시에 필요한 일을 수행할 경우,

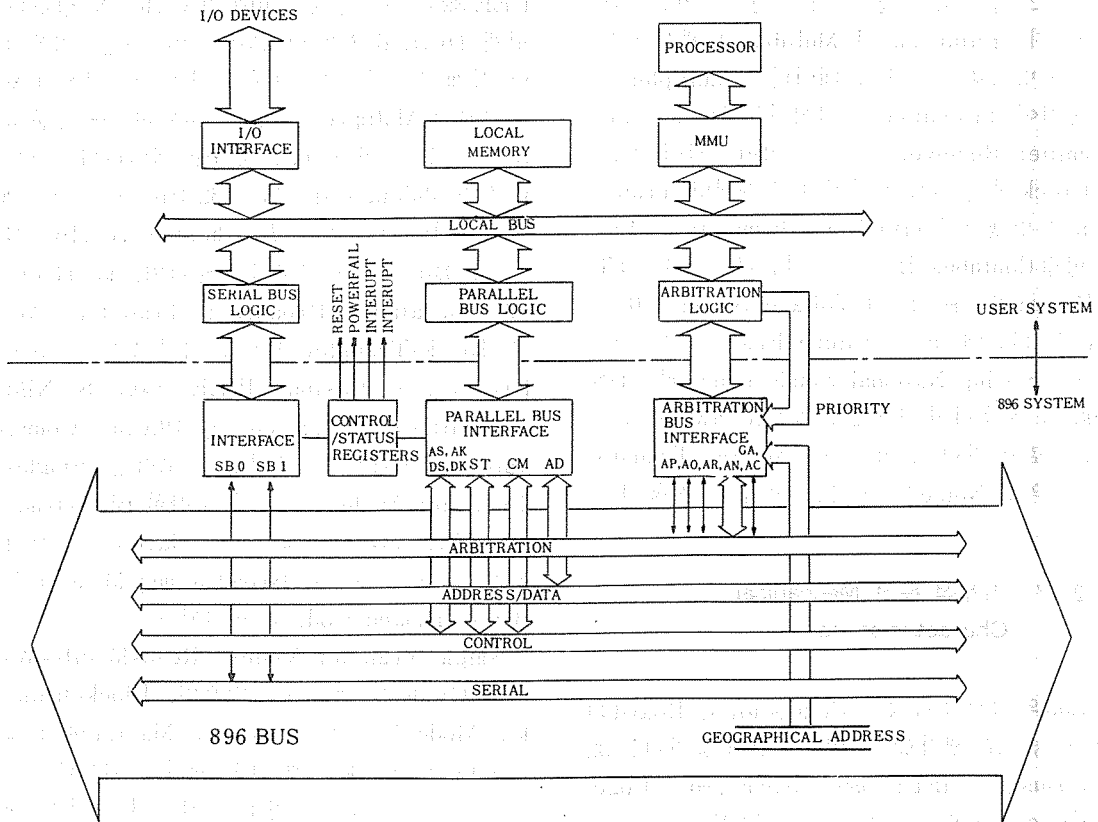


그림 6. Interfaces in a typical IEEE 896 Board

Lock Protocol을 사용하여 일이 끝날 때까지 다른 프로세서의 접근을 막을 수 있게 함으로써 가능하다.

Arbitration은 S-100과 Fastbus와 같으며 완전한 분산형이기 때문에 신뢰도가 높다. National Semiconductor Co.에서 설계한 특별한 Bus Transceiver는 Futurebus의 또 하나의 장점으로 TTL Driver의 문제점을 개선하여, 빠르고 정확한 전송을 보장하고 있다. 이 Tran-

sceiver는 전압의 swing 폭을 1V로 줄임으로 전력의 소비를 줄여서 버스가 차지하는 전력 소비의 비중과 전원 전압의 변동에 의한 Noise를 줄이고 있다. 작은 전압 Swing 폭은 절대 Noise Margin을 줄여서 불리한 것 같지만 Crosstalk 현상을 줄임으로써 상대적인 Noise Margin은 변함이 없고, 전원에 의한 영향을 줄임으로 전체적인 Noise Immunity는 향상된다.

Futurebus는 Tectronix Inc, IBM 그리고,



National Semiconductor Co. 등에서 지원하고 있다.

### 3.2 Multibus II

Multibus II는 동기형 버스로써 많은 신호들이 Multiplex된 32-Bit Bus이다. TI의 Nubus와 함께 IEEE 896.2 Committee에 제출된 Bus로 거의 IEEE 896.2의 규격과 비슷한 것으로 평가된다. 5개의 Bus로 구성되어 하나의 Bus 구조가 갖는 성능의 한계 및 제한된 확장성 등을 배제하여 시스템이 여러 가지 요구에 만족시키는 유연성을 보장한다. 8-, 16-, 24-, 32-Bit 데이터 전송을 할 수 있으며 Justified Bus 형태를 갖는다. 그림 7과 같이 iPSB(Parallel Bus), iLBX II(Local Bus Extension), iSSB(Serial System Bus), iSBX(Input/output Expansion Bus), Multichannel Direct-memory-access bus)로 구성되어 있다. iPSB는 40 Mbyte/sec의 전송 속도를 가지며 iLBX II는 48 Mbyte/sec의 속도를 Block 전송 모드를 사용함으로 가능하다. iSSB는 CSMA/CD 전송 Protocol을 사용하는 Serial Data Link로 2 Mbyte/sec의 전송 속도를 가지며, 주로 Multiprocessor 시스템에서 프로세서 사이의 정보 교환을 지원한다. Interprocessor Communication 방법 중에 전달하고자 하는 정보를 모두 복사하는 Pass-by-value와 정보의 어드레스만을 전달하여 메모리 상에서 공유하는 Pass-by-reference가 있는데, 前者의 경우는 모든 정보를 전송해야 하기 때문에 속도가 늦고 後者は 정보를 공유하기 때문에 Process Synchronization 문제가 발생할 수 있다. Multibus II에서는 Pass-by-value 방법을 이용하면서 iSSB를 이용하여 전송 속도를 보완하고 있어 두 방법의 장점을 모두 취하고 있다.

어드레스와 데이터, Arbitration Line과 Initial Control, 시스템 제어 Line 등이 Multiplex되어 있어서 하나의 96 Pin DIN(IEC 603-2) Connector로 32-Bit의 데이터 전송이 가능하다. 많은 제어 신호가 Multiplex되어 있고 동

기형 Protocol을 사용하기 때문에 Logic Analyzer와 같은 Tool 이용이 용이하여 Debugging이 쉽다.

Geographical Addressing 기능은 각 모듈의 Bus상의 어드레스를 Switch로 하는 것을 Interconnect Register를 사용하여 소프트웨어에서 직접 정할 수 있게 한 것이다. 이 기능은 Diagnostic 능력을 보완한다.

Multibus II는 200, 여 개의 Multibus-I의 제조사들이 지원하고 있고 또한 IEEE 896.2의 규격으로 Nubus보다 접근해 있는 것으로 평가되고 있기는 하지만 Justified Data Transfer ISO의 Upper Layer(Interprocessor Communication)의 고정 등이 문제로 제기되고 있다. 또한 Multibus I과의 소프트웨어의 호환성을 설계 목표로 생각하고 있는 Intel로서는 많은 호환성의 문제를 안고 있다.

Intel에서 개발했으며 Multibus I를 지원하는 제조사와 특히 Hewlett-Packard, Siemens, Tectronix 그리고 AMD 등에서 지원하고 있다.

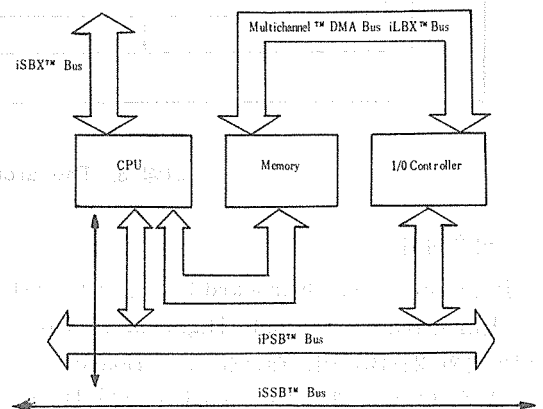


그림 7. The architecture of Multibus II

### 3.3 VME(Versa Module Europa) Bus

Motorola, Mostek 그리고 Signetics에서 1981년에 공동 개발, 발표한 VME Bus에 1982년

다시 Local Bus Extension인 VMX Bus 와 Serial Link인 VMS와 함께 현재의 규격을 완성했다. 비동기형 Protocol, 데이터와 어드레스가 분리되어 있고, 16-Bit 시스템에 최적화되어 있는 Word-Justified Bus이다. Word-Justified Bus 임으로 24-Bit 전송이 없다.

VMS Bus의 Lower Layer Protocol은 Multibus II와 비슷하며 Upper Layer는 Bus 규격에 정하지 않고 있다. Message Frame의 크기는 최대 32 Byte로 3종류의 Message Frame이 있다. 위의 특성을 제외하고는 Arbitration; 전송속도, Bus의 기능 등은 Multibus II와 거

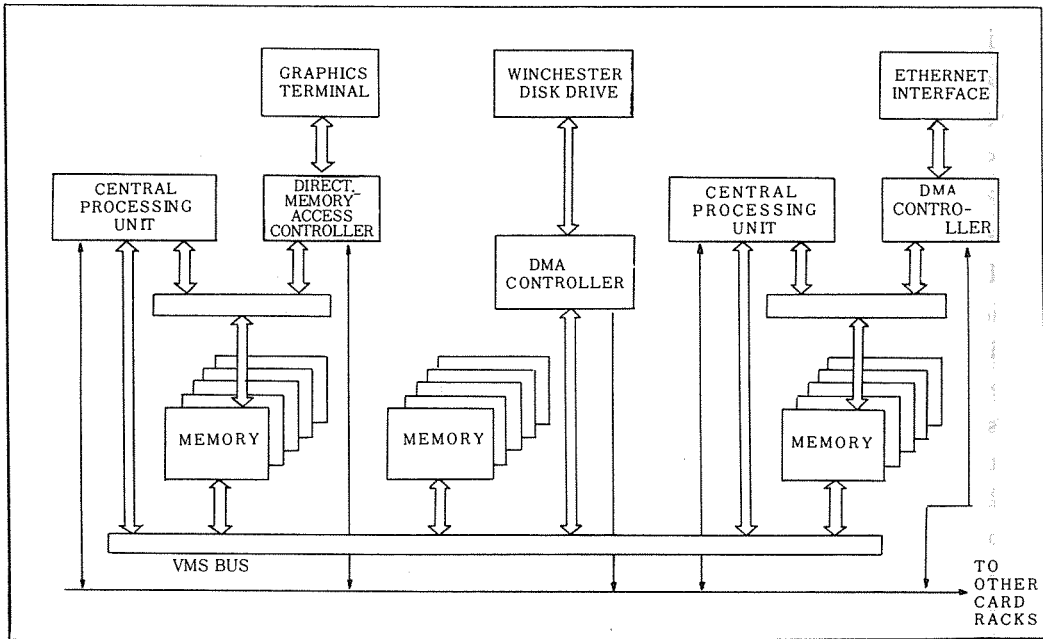


그림 8. The architecture of VMEbus

의 비슷하다.

Board의 크기는 Eurocard를 기준으로 하나의 Eurocard 크기(Single Height)로 16-Bit 데이터와 24-Bit 어드레스를 갖는 Board를 구성할 수 있고 두개의 Eurocard의 크기(Double Height)면 VME, VMS 그리고 VMX를 사용하는 Board를 구성할 수 있다.

먼저 발표된 32-Bit Bus 이고 또한 MC68000 Family와 인터페이스가 용이하기 때문에 MC 68000 Family가 많이 사용됨에 따라 현재 32-Bit Bus 중 가장 큰 시장 점유율을 갖고 있고, 또한 계속 증가하는 추세에 있다. 현재 Force

Computers INC., Motorola, Mostek, Data-sud Philips 순으로 VME 제품의 시장을 점유하고 있다.

Nubus는 MIT에서 Artificial Intelligence, Network, Office Automation을 위한 Workstation 연구에 사용할 목적으로 개발했으며, License가 Western Digital을 거쳐 TI로 왔으며 IEEE 896.2 Committee에 제출되었다. 이 Bus는 어느 프로세서에나 사용할 수 있도록 프로세서에 의존되는 부분을 최대한 줄이는 데 역점을 두고 설계되었다. LISP Machine Inc.의 Lisp Machine은 Bit-sliced 프로세서와 같이 Nubus

를 사용하고 있으며, TI의 Workstation은 MC68010과 같이 사용되고 있다. 동기형 Protocol, Multiplexed, 37.5 Mbyte 전송속도를 가지며 프로세서와의 의존도를 고려하여 ISO의 Layer 1과 2만을 구현하고 있다.

현재 TI에서는 사용자가 보다 쉽게 사용할 수 있도록 규격을 정리하고, 또한 소프트웨어와 하드웨어의 설계를 돕기 위한 응용 방법, 설계 지침 등을 준비하고 있다.

### 3.5 기타

DEC에서는 Minicomputer에서 많이 사용되던 BI-Bus (Bus Interface Bus) 들을 개선한 하나의 32-Bit Bus를 발표하여 증가하는 Micro- 혹은 Supermicro-Computer와 경쟁할 계획을 갖고 있다. 이 버스는 Multibus II와는 보다 근본적인 설계 개념의 차이를 보일 것으로 평가된다.

## 4. 맺음 말

현재 32-Bit Bus가 주목을 받고 있는 것은 32-Bit 마이크로 프로세서의 출현으로 마이크로 프로세서를 이용한 32-Bit 시스템이 본격화될 전망이다기 때문이다. NS32032와 MC68020은 최초의 32-Bit 데이터 버스를 갖는 마이크로 프로세서들로 이들의 기능을 완전히 발휘할 수 있도록 지원할 수 있는 Bus가 꼭 필요하게 되었다.

32-Bit Bus에의 관심은 IEEE 896의 표준화의 방향과 가장 많은 시장 점유율을 갖고 있는 VME Bus와 Multibus II와의 전망 등으로 집약된다.

IEEE 896 표준은 현재 거의 굳어지고 있으나 범용의 32-Bit 시스템에 적합한 동기형 (IEEE 896.2)과 고성능 혹은 Fault-Tolerant 시스템에 알맞은 비동기형 (Futurebus)으로 나누어짐에 따라 두 Bus 간의 호환성 문제를 최종적으로 고려하고 있다. 즉 Address Mapping, Error Control, ISO의 Higher Protocol Layer 등을 구현함에 있어서 두 Bus 간에

소프트웨어의 완전한 호환성을 목표로 하고 있다.

IEEE 896.2의 규격과 거의 일치하는 Multibus II와 MC68000 Family의 지원을 받는 VME Bus의 대결은 현재 먼저 발표된 VME Bus가 시장 점유율에서 앞서고 있지만, 앞으로 표준화가 발표되면 거의 대등하게 사용될 전망이다. 동기형과 비동기형의 차이는 있지만 근본적으로 기술적인 큰 차이가 없기 때문에 두 Bus의 앞날은 지원하는 회사들과 사용자가 제공받을 수 있는 제품의 발표 시기 등에 따라 좌우된다고 할 수 있다.

새로운 32-Bit Bus는 종래의 Bus (Multibus, S-100, etc)에서 크게 고려하지 않았던 Multiprocessor에 대한 지원이 두드러지고 있다. Local Bus의 확장과 Block 전송 모드 등은 종래의 버스에서는 버스에 삽입되는 모듈이 증가함에 따라 버스의 데이터 전송이 포화되었던 문제를 해결했고, Serial Data Link, Read-Modify-write Cycle 등으로 프로세서 사이의 정보 교환문제 (데이터의 호환성, Synchronization, etc)를 해결하고 있다. 이밖에도 Bus 제어신호의 Protocol이 일부 특수한 프로세서에 의존하지 않는 것, Interrupt 구조 등이 Multiprocessor 시스템에 대한 지원이 두드러진 것은, 앞으로 마이크로 프로세서와 Bus 구조를 중심으로 하는 시스템의 방향을 Bus가 암시하고 있다고 할 수 있다.

종래의 Bus가 시스템 안에서 갖던 의미는 단순한 하드웨어 Board 사이의 인터페이스였지만, 앞으로의 버스는 좀더 시스템 전체에 접근해 있다는 것을 지금까지의 조사로 알 수 있다.

그러므로 시스템 설계자는 시스템 설계 시에 Bus의 중요성을 인식하고 시스템과 Bus 사이의 폭 넓은 이해가 필요하다.

## REFERENCE

1. Danny Cohen, "On Holly Wars and a plea for Peace", COMPUTER, Oct. 1981

2. Hubert Kirmann, "Data Format and Bus Compatibility in Multiprocessors," IEEE MICRO, Aug. 1983
3. Andrew A. Allison, "Status Report On the P896 Backplane Bus," IEEE MICRO, Feb. 1981
4. Robert Rosenberg, Erik L. Keller, "Multiprocessing 32-Bit buses are starting to blossom, Electronics," Mar. 22 1984
5. John Beaston, "Multiprocessor bus is ready to meet 32-bit applications of future," Electronics, Mar. 22 1984
6. Craig Mackenna, Rick Main, John Black, "Backup support gives VMEbus powerful multiprocessing architecture," Electronics, Mar. 22 1984
7. John Theus, Mathew Taub, R. V. Balakrishnan, "Futurebus anticipates coming needs," Electronics, Jul. 12 1984
8. Harvey J. Hindin, "Thirty-two bit system designer face decision time," COMPUTER DESIGN, Feb. 1984
9. Interface Devices (IEEE Standard 696-1983), Institute of Electrical and Electronics Engineers, Inc., New York, 1983.
10. Microcomputer System Bus (IEEE Standard 796-1983), Institute of Electrical and Electronics Engineers, Inc., New York, 1983
11. Versabus Specification Manual, Pub. No. M68KVBS (D 4), Motorola, Inc., Phoenix, AZ, May 1981
12. VME Bus Specification Manual, Pub. No. M68KVMEB (DI), Motorola, Inc., Phoenix, AZ, Oct. 19
13. MC 68020 Technical Summary, Pub. No. BR-243, Motorola, Inc., Phoenix, AZ, 1984
14. Specifications for Advanced Microcomputer Backplane Buses, IEEE Microprocessor Standards Committee P896 Working Group, Nov. 1983

..... (P49에서 계속) .....

도 사용할 수 있다.

美國이 이와 같은 새로운 方法에 더욱 적극적으로 대처하고 있지 않는 데 대해 가볍게 보는 경향도 있으나 그 점은 걱정하지 않아도 될 것 같다. 템포는 늦더라도 계속 앞으로 前進하고 있기 때문이다.

1970년대 후반에 있어서의 오피스 컴퓨터의 導入이 그랬던 것과 같이 학교에서의 컴퓨터 導入이라고 하는 것은 上部로부터의 指示에 의해 이루어졌던 것은 아니고 일부의 관심이 있는 개인을 중심으로 진행되었다.

그들은 使命感에 불타 컴퓨터를 배우기 시작했다. 그리고 司書, 컴퓨터를 약간 만져본 일이 있는 教師, 數學教師, 進歩의인 校長 등, 이와 같은 사람들이 學父母들의 支援이나 PTA의 活動資金에 의해 購入한 적은 수의 컴퓨터를 사용하여 시작했던 것이다. 어쨌든 처음에는 대단한 것이 못되었지만 그 成果가 확실해지면 教育關係者들도 점차적으로 어떤 컴퓨터가 필요한가를 생각하게 될 것이다. 그렇게 되면 컴퓨터 메이커에도 지금까지보다 廣域의 學區나 州레벨에서의 受注가 들어오기 시작하고 있다. Tandy社의 W. Gattis 教育部長에 의하면 「학교에 있

어서의 컴퓨터 購入方法에 커다란 변화가 나타나고 있다. 여러 가지 可能性과 現實性을 띠고 있으며 商談도 물리고 있다」라고 말하고 있다.

教育에는 教職員으로부터 教科書에 이르기까지 여러 가지의 것이 필요하지만 컴퓨터는 그 價格도 점점 떨어지고 있다는 점에서 다른 것들과는 다르다. 하드웨어의 機能이 상승되고 네트워킹化가 되어 각종 터미널이 컴퓨터에 接續되어 디스크나 프린터가 共同 利用하게끔 되었다.

이와 같은 機能 향상에 따라 학교에서의 컴퓨터室이 간단하고 더우기 값싸게 設置하게끔 되고 있어 教師와 학생이 Regular Base로 컴퓨터를 사용하게 될 것이다.

印刷機械가 발명되어 읽기, 쓰기, 藏書라는 것이 大衆化되어 적어도 300年の 歷史를 거치게 되었다. 곧 컴퓨터時代를 맞으려고 하는 時期에 美國의 教育制度가 改革이 진전되지 못하고 있다는 데 대해 조바심을 느끼고 있다면 이것을 다시 한번 상기해 볼 필요가 있다고 본다. 조급한 나머지 方向錯誤를 가져오기보다는 그 속도는 느리지만 침착하게 進行시키는 것이 좋다고 본다.