상용 응용 서버용 클러스터 연결망 : Xcent-Net

한국전자통신연구원 한우종 · 박 경

1. 서 론

컴퓨터 성능이 비약적 발전을 거듭하고 있음에도 더 나은 성능을 제공하는 컴퓨터에 대한 요구는 계속되고 있다. 이러한 요구에 부응하기 위하여 병렬처리 기술은 발전해 왔으며 주로 과학계산 및 공학분야에 사용되어 왔다[1,2]. 하지만 근래에 들어서 상업용 분야에서도 컴퓨터 시스템의 성능요구가 커짐에 따라 상업용 컴퓨터 시스템 분야에서도 병렬처리 기술이 사용되기 시작하였다[3].

초기 상업용 컴퓨터 시스템들이 주로 사용하는 병렬처리 기술은 대칭형 다중처리 구조(SMP: Symmetric Multiprocessing)로서 공유버스를 통하여 다수의 프로세싱 노드를 연결하는 구조이다. 대칭형 다중처리 구조는 사용자에게 단일 주소 공간을 제공하여 프로그램 개발 및 관리의 편의성을 제공하는 반면 공유버스에서 발생하는 병목현상으로 인하여 확장성에 있어서 명백한 한계를 갖는다.

슈퍼컴퓨터 분야에서 사용되기 시작한 초병렬처리 구조(MPP:Massive Parallel Processing)는 다수의 프로세싱 노드를 고속 연결망을 사용하여 연결하고, 노드간에 메시지 패싱 기법을 사용하여 통신하는 구조로서 높은 확장성을 제공하지만 응용프로그램 개발이 어려운 단점이 있다.

분산 공유 메모리 구조(DSM:Distributed Shared Memory)는 대칭형 다중처리 구조의 편의성과 초병렬처리 구조의 확장성을 동시에 보장하는 구조이지만 전송 지연 시간 해결 및 데이터 일관성 유지 문제 등과 같은 기술적인 문제를 해결해야 하는 부담이 있으며 구현 비용이 높은 단점이 있다.

프로그램 개발 및 관리의 편의성과 확장성을 동시에 보장하면서 우수한 가격 대 성능비를 갖는 병렬처리 기술이 연구되기 시작하였으며, 메시지 패싱 기술과 공유 가상 메모리(SVM: Shared Virtual Memory) 기법을 사용하는 클러스터링 기술이 각광 받기 시작하였다.

클러스터링 기술의 가능성을 보여준 NOW (Network of Workstations)는 초병렬처리 구조에 비해 우수한 가격 대 성능비를 제공하면서 성능 요구에 따라 자연스러운 확장성을 제공하는 장점을 갖는다. 또한 상업용 컴퓨터 시스템으로서 필수적으로 제공하여야 하는 고가용도 보장이 낮은 비용과 노력으로 구현 가능하다는 장점을 가짐으로써 초병렬처리 구조에 비해서 OLTP등 상업용 분야(commercial application)에서 두각을 나타낼 수 있다[4].

클러스터링 기술에서 가장 중요한 기술은 노드간 연결을 제공하는 연결망으로 Fast Ethernet, ATM, SCI(Scalable Coherency Interface) 등이 주로 사용되어 왔다. Fast Ethernet과 ATM 같은 표준 통신 프로토콜을 사용하는 연결망은 대부분의 클러스터 시스템이 사용하는 기술이지만 고성능 클러스터 시스템을 위한 전용 연결망으로 사용하기에는 전송 지연 시간이 큰 단점을 가진다. SCI는 Fast Ethernet이나 ATM에 비해 낮은 전송 지연 시간을 갖지만 작은 메시지 전송에 알맞게 설계되어 있어 노드간에 분산된 입출력 데이터를 전송하는 데 단점을 가지고 있다. 따라서 고성능 클러스터 시스템은 시스템 특성을

고려한 고유의 SAN(System Area Network) 또는 CAN(Cluster Area Network)을 개발하고 이를 이용하여 시스템을 구성하는 것이 바람직하다.

클러스터링 기술을 발전시키게 된 원동력 중 하나는 대량 생산되는 PC보드나 워크스테이션들이 높은 성능을 경제적으로 제공하게 된 것을 들 수 있다. 특히 SHV(Standrad High Volume) 노드라 불리우는 인텔의 4-way 대칭형 다중처리 보드를 사용한 클러스터링 시스템은 구현 용이성 및 높은 가격 대 성능비를 제공하여 상업용 컴퓨터 서버 분야에서 두각을 나타내고 있다.

고가용성과 고확장성을 유지하면서 우수한 가격 대 성능비를 보장하기 위해서는 SHV노드와 같은 상용 노드를 고속 연결망으로 연결하는 클러스터 시스템이 가장 경쟁력 있는 구조이다. 하지만 각 노드가 UNIX와 같은 독립적인 통합형 운영체제에 의해서 동작하면서 별도의 시스템으로 인식되는 클러스터 시스템에 일반적인 상용 응용 프로그램을 이식하기란 매우 어려운 작업이다. 클러스터 시스템에서 운영체제 수준에서 단일 시스템 이미지를 제공하는 운영체제 기술은 클러스터 시스템의 유연성 및 편의성을 제공하는 중요한 기술이다. 각 노드는 독립적인 커널을 운영하지만 서버 프로세스는 각 노드에 분산되어 클러스터 시스템 전체에 걸쳐 단일 시스템 이미지를 제공한다.

마이크로 커널 기반 운영체제를 사용하는 클러스터 시스템은 노드별로 독립적인 통합 운영체제를 사용하는 클러스터 시스템에 비하여 프로그램 개발 및 관리의 편의성을 제공하는 반면 노드간에 빈번한 낮은 전송 지연 시간의 빈번한 통신 요구가 발생하며, 따라서 고성능 클러스터 연결망이 요구된다.

SPAX(Scalable Parallel Architecture based on Xbar) 시스템은 SHV 보드를 전용 클러스터 연결망인 Xcent-Net으로 연결한 SMP 클러스터 시스템으로 마이크로 커널 기반형 운영체제를 사용하여 전체 시스템에 걸쳐 단일 시스템 이미지를 제공한다. OLTP(On-Line Transaction Processing)과 같은 상업용 응용 프로그램 처리를 목적으로 개발되었으며 128개의 SHV노드를 연결할 수 있다[6].

본 고에서는 SPAX 시스템의 구성과 SPAX 시스템을 위하여 개발된 Xcent-Net의 설계와 구현에 대하여 기술하고자 한다.

2. SPAX 시스템

SPAX 시스템은 공유버스 구조의 SMP시스템인 TICOM III[7]의 후속 기종으로 초고속 통신망의 정보 처리 서버로 개발되었다. SHV 노드를 8개 연결하는 클러스터를 기본 단위로 하여 128개까지 확장가능한 시스템으로 SMP 시스템과 클러스터링 시스템의 특징을 동시에 만족시키는 계층구조를 갖는다. 각 클러스터는 SHV 노드로 구성되는 공유 디스크 구조의 클러스터 시스템이며 각 노드는 SHV 노드를 용도에 따라 4-way SMP형 사용자 프로세싱 노드 또는 단일 프로세서의 I/O 노드로 사용할 수 있다.[1] 표준 모델은 단위 클러스터당 4개의 SMP노드와 4개의 I/O노드로 구성하며 시스템 전체에 걸쳐 최대 64개의 SMP 노드와 64개의 I/O 노드로 구성된다. 사용자의 요구 및 시스템 사용 목적에 따라 SMP 노드와 I/O 노드의 구성비를 유연하게 구성할 수 있다. 그림 1은 SPAX 시스템의 전체 구성도이다.

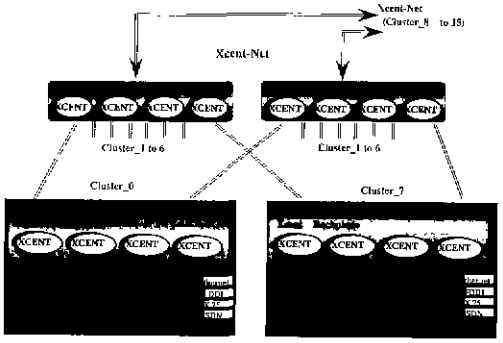

그림 1 SPAX 시스템 구조

각 노드는 PCI버스를 통하여 Xcent-Net에 연결되며 노드 당 두 개의 Xcent-Net 연결 포트를 갖는다. Xcent-Net은 10X10 크로스바 라우

1) I/O 노드도 사용자 프로세싱 노드와 기본 구조를 공유하고 있으나 CPU-bound 특성을 갖지 않으므로 가격을 고려하여 단일프로세서만 장착하도록 하였다

터를 사용한 계층구조 크로스바 연결망으로 64바이트 데이터 패킷을 전송하는 패킷 스위칭 네트워크이다. Xcent-Net의 기능 및 구현에 대한 사항은 3장과 4장에서 자세히 기술한다.

SPAX 시스템은 프로세서, 노드, 연결망, 디스크등 시스템을 구성하는 각 요소에서 단일점 결함을 허용하는 고가용성 시스템이다[8]. 각 노드는 프로세서 결함을 허용할 수 있도록 FRC (Functional Redundancy Check)기능을 제공하며, 각 노드는 이중화된 연결망에 연결되어 있고 디스크 접근 경로도 이중화 할 수 있다. 특히 이중화된 연결망은 결함이 발견되면 결함이 발견된 네트워크를 시스템에서 분리할 수 있으며 정상시에는 사용자의 요구에 따라 개별 연결 통로로 사용되어 높은 대역폭을 제공한다.

SPAX 시스템은 MISIX라는 마이크로 커널 기반형 운영체제를 탑재한다. 각 노드는 동일한 마이크로 커널을 복제하여 동작하며 프로세스 관리자, 프로세서간 통신 관리자, 일관성 유지 관리자등과 같은 서버 프로세스는 각 노드의 마이크로 커널위에 분산되어 탑재되어 시스템 전체에 걸쳐 사용자에게 SVM과 단일 시스템 이미지를 제공한다[9].

3. Xcent-Net 구조

3.1 요구사항

SPAX 시스템 개발의 가장 중요한 부분 중 하나는 고성능 클러스터 연결망인 Xcent-Net 개발이었다. 시스템 요구사항으로부터 추출된 Xcent-Net 설계 요구 사항은 다음과 같다.

3.1.1 비균등 밴드폭 할당

각 단위 클러스터는 충분한 연산능력과 디스크 공간을 확보하고 있으므로 클러스터 간 연결망보다 클러스터 내부 연결망의 통신 수요가 훨씬 높다. 또한 대부분의 상용 응용 프로그램은 전체 연결망을 동시에 사용할 정도로 높은 데이터 병렬성을 가지고 있지 않다. 따라서 클러스터 내부 연결망의 밴드폭을 클러스터 간 연결망의 밴트폭보다 높게 할당하여야 하며, Xcent-Net에서는 4대 1의 비율로 밴드폭을 할당하였다.

3.1.2 응용프로그램에 대한 낮은 민감성

다중처리 시스템은 연결망 구조에 따라서 응용 프로그램이 영향을 받는 경우가 발생하는데 예를 들면 공유버스 구조는 응용 프로그램에 별 영향을 주지 않는 반면, 링 구조의 경우에는 통신하는 노드 상호간의 위치에 따라 전송 지연 시간이 다르므로 응용 프로그램 특성에 따라 성능차이가 날 수 있다. 상용 응용 프로그램의 특성상 특정 시스템에 최적화되도록 제 설계하는 것이 현실상 어려운 점을 감안하면 전송 지연 시간이 연결망 상에서 동일한 것이 응용 프로그램에 영향을 적게 미친다. 이점은 단위 크로스바 연결망의 크기를 결정하는 중요한 요소가 된다.

3.1.3 단순하고 전역적인 단일 전송 프로토콜

Xcent-Net 설계에 있어서 중요한 설계 사항 중에 하나는 클러스터 내부 연결망과 클러스터 간 연결망이 동일하게 사용할 수 있는 단순한 전송 프로토콜 설계이다. 클러스터 내부 연결망과 클러스터 간 연결망이 서로 다른 전송 프로토콜을 사용할 경우에는 프로토콜 변환에 따른 오버헤드로 인하여 전송 지연 시간이 커지며 시스템의 복잡도가 증가하는 단점이 있다.

3.1.4 시스템 패키징의 용이성

대부분의 학술적인 연구에서는 무시되는 경향이 있으나 시스템 패키징에 대한 요구사항은 구현에 있어서 매우 중요한 사항이다. 실제 시스템 구현에 있어서는 보드 레이아웃, 커넥터 핀 배치, 케이블 연결 방법 등 고려해야 한다. 또한 ASIC 구현에 있어서 핀과 패드의 제약은 연결망의 구조 및 연결 포트의 전송폭을 제한하게 되며, 핀의 배치 및 전송 속도, 전송 폭은 신호 전송에 있어서 전기적인 신호 충실도를 고려하여 설계되어야 한다.

3.1 5 연결망 분할 및 분리 기능

대부분의 상용 응용 프로그램은 독립적인 I/O 작업을 갖는 작은 작업들이 유기적으로 결합되는 형태이다. 따라서 연결망은 각 타스크들이 할당되는 노드를 또는 클러스터를 기준으로 I/O 장치에 공간적인 공유를 가능하게 해 주어야 하며 또한 연결망의 분할도 가능해야 한다. 효과적인 연

결망 분할을 위하여 I/O 장치들은 전체 연결망에 골고루 분산될 수 있어야 하며 연결망은 노드 형태에 대하여 중립적이어야 한다.

3.1.6 고가용성

상용 응용 프로그램은 신뢰도가 높은 컴퓨팅 환경을 요구하는 바 연결망에서 단일점 결함을 허용하는 것은 시스템의 가용성을 높이기 위해서 필수적으로 요구된다. 하지만 연결망의 경우 고장감내를 위하여 예비 연결망을 갖는 것은 시스템 구현 비용을 올리는 직접적인 요인이 된다 따라서 효과적인 고장감내 연결망 구조가 필요하며 Xcent-Net은 전송 경로의 중복성을 제공하되 stand-by에 따르는 비용 부담을 피할 수 있도록 embedded redundancy 구조를 갖는 연결망으로 설계하였다.

표 1 Xcent-Net 설계 사항

Xcent-Net	
Topology	Dual hierarchical crossbar
Router	10 X 10 Crossbar router(8*2)
Nodes	128 Nodes(16 Clusters)
Channel Width	32 bits/channel
Operation Freq	33.3 MHz(long), 66.6MHz(short)
Bandwidth	33.8 Gbytes/sec at 33.3MHz
Max Distance	33 feet between routers
Fault Tolerancy	Embedded redundancy Node isolation
Routing	Virtual cut-through Adaptive path control
Clock scheme	Plesiochronous clocking Source synchronous transmission

3.2 Xcent-Net 기능

앞절에서 설명된 요구사항에 따라 Xcent-Net은 크로스바 토폴로지를 갖는 연결망으로 설계되었다 클러스 내부 연결망과 클러스터 간 연결망이 동일한 전송 프로토콜을 사용할 수 있으며 이중화 된 계층 크로스바 연결망으로 설계되었다. 그림 1에서 나타난 바와 같이 Xcent- Net은 라우터 클라우드(router cloud)로 구성되어 있으며 라우터 클라우드는 Xcent라 명명된 4개의 8비트, 10X10 크로스바 라우터로 구성되어 32비트 전송로를 제공한다. 각 클러스터는 이중화된 연결망을 구성을 위하여 두개의 라우터 클라우드로 구성되며 각 라우터 클라우드는 8개의 클러스터 내부 연결망 포트와 2개의 클러스터 간 연결망 포트를 제공한다. 표 1은 Xcent-Net의 설계 사항을 정리한 것이다.

3.2.1 메시지

Xcent-Net에서 사용되는 메시지는 용도에 따라 제어 메시지와 데이터 메시지로 구분된다. 제어 메시지는 최소 4바이트에서 최대 64바이트로 구성되며 노드간 자원 교환 및 프로세스 제어 메시지로 사용된다. 데이터 메시지는 최대 1 Mbytes까지 구성되며 페이지 규모의 데이터 전송에 사용된다.

서로 다른 두 종류의 메시지를 지원하기 위하여 DMA 기반형과 메모리 맵 방식의 메시지 처리 방식을 모두 지원한다 제어 메시지 처리를 위하여 메모리 맵의 일정부분을 메시지 영역으로 지정하고 사용자 프로그램은 이 영역에 쓰기 동작을 하거나 검색을 통하여 제어 메시지 전송 및 수신을 수행한다. 사용자 프로그램에 의해서 메시지 영역에 쓰기가 수행되면 연결망 접속장치는 자동으로 해당 메시지를 목적지로 전달한다[10]. 또한 대용량 데이터 전송에 사용되는 데이터 메시지 처리를 위하여 주기억 장치에서 직접 읽어서 패킷 형태로 바꾸어 전송할 수 있는 DMA방식도 제공한다[10,11].

3.2.2 패킷

Xcent-Net은 패킷을 기본 전송단위로 하는 패킷 스위칭 연결망으로 메시지는 패킷으로 분할되어 연결망을 통하여 전송된다. 패킷은 최소 3플릿에서 최대 21플릿으로 구성되며 최대 64바이트의 데이터 페이로드(payload)를 제공한다. 패킷은 태그 플릿과 제어 플릿 그리고 데이터 플릿으로 구성되며 태그 플릿은 전송 경로 라우팅에 사용되고 제어 플릿은 패킷 시퀀스 번호와 같이 연결망 정합장치에서 사용하는 정보로 구성된다. 데이터 플릿은 데이터 페이로드를 제공하기 위한 것으로 최대 길이는 16플릿(64바이트)이다.

3.2.3 라우팅

Xcent-Net은 전송 지연 시간을 최소화하고 연결망상에서 발생하는 블로킹 현상을 제거하기 위하여 Virtual Cut-through 라우팅 방식을 사용한다. 라우팅 방식은 구현 환경하에서 효율성과 전송 지연 시간을 고려하여 선택되었다[12].

Xcent-Net은 Source-based Routing 방식에 따라 전송 경로를 제어한다. 패킷 전송시에 송신단이 패킷의 전송 경로를 미리 결정하여 필요한 태그를 구성하여 전송한다. 라우터는 다중 태그 중 최상위 태그 정보를 해독하여 경로를 설정하고 사용된 태그를 패킷에서 제거하는 방식을 사용한다[13]. 그림 2는 다중 태그를 사용하는 Source-based Routing 방식을 보여준다.

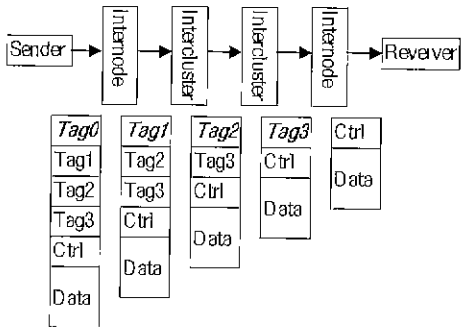

그림 2 다중 태그를 이용한 Source-based Routing

3.2.4 버퍼링과 흐름제어

Xcent-Net에서 사용하는 Virtual Cut-through 라우팅 방식은 패킷 버퍼와 Clear-to-send 흐름제어 방식으로 설계되었다. 패킷 수신단은 수신되는 플릿을 패킷 버퍼에 저장함과 동시에 태그 플릿을 사용하여 경로를 설정한다. 경로 설정이 완료되면 패킷 버퍼에 전체 패킷이 모두 입력되지 않았어도 플릿을 읽어내어 송신단으로 송신하기 시작한다.

수신단은 송신단으로 패킷 버퍼의 가용 여부를 흐름제어 신호를 통하여 미리 알려주어야 하며, 송신단은 흐름제어 신호를 확인한 후에 패킷 전송을 시작한다.

3.2.5 동기방식

대규모 연결망에서 동일한 전역 클럭의 유지 및 분배는 매우 어려운 일이다. Xcent-Net은 독립 동기 방식을 사용하여 연결망상에서 전역 클럭을 제거하였다[13,14]. 각각의 라우터 클라우드는 독립된 오실레이터를 갖고 지역 클럭을 생성하여 사용하며, 각 지역 클럭은 동일한 주파수를 갖는다 따라서 각 라우터 클라우드는 주파수는 동일하지만 위상은 서로 독립적인 클럭을 유지하게 된다. 따라서 송신단과 수신단간에 패킷을 전송함에 있어서 송신 클럭과 수신 클럭이 서로 다른 위상을 가지게 되며 수신단에서는 수신신호를 래치하기 위한 동기 정보가 별도로 요구된다. 이 문제를 해결하기 위하여 Xcent-Net에서는 근원지 동기 전송 방식을 사용하여 패킷 데이터를 전송한다[13,14]. 그림 3은 근원지 동기 방식을 나타내고 있다.

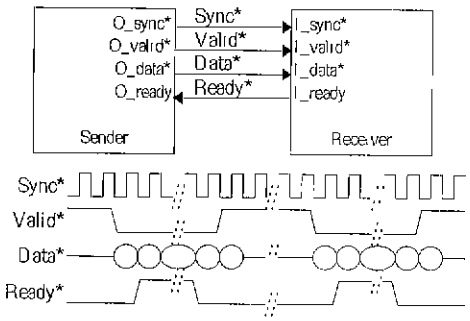

그림 3 근원지 동기 전송 방식

그림 3에서와 같이 송신단에서 수신단으로 패킷 데이터를 전송하는 데는 두 개의 동기 신호가 사용된다. 송신단은 패킷 전송시에 패킷 동기 신호 Valid* 신호와 플릿 동기 신호인 Sync* 신호를 패킷 데이터와 동시에 전송한다 플릿 동기 신호는 송신단의 동작 클럭을 사용한다. 수신단은 플릿 동기 신호를 사용하여 플릿 데이터와 패킷 동기 신호를 래치하고 래치한 데이터를 수신단 동작 클럭으로 동기화하여 사용한다.

4. Xcent-Net 구현

Xcent-Net을 구성하는 소자들은 라우터, 백플레인 그리고 케이블 등이다. 라우터는 버퍼링, 라우팅, 흐름 제어, 패킷 전송 등 Xcent-Net의 주요 기능을 수행하는 기본 소자로 클러스터 내부 연결망, 클러스터 간 연결망 구성을 위하여 백플

레인에 실장된다. 백플레인으로 구성된 클러스터 내부 연결망과 클러스터 간 연결망은 케이블로 연결된다.

4.1 라우터

앞에서 설명된 연결망 기능은 대부분 라우터에 구현되어 있다. 라우터는 그림 4와 같이 패킷 버퍼를 갖는 10개의 입력 포트와 10개의 출력 포트, 10개의 지역 중재기, 하나의 전역 중재기 그리고 10X10 크로스바 패스와 패스 제어기로 구성되어 있다.

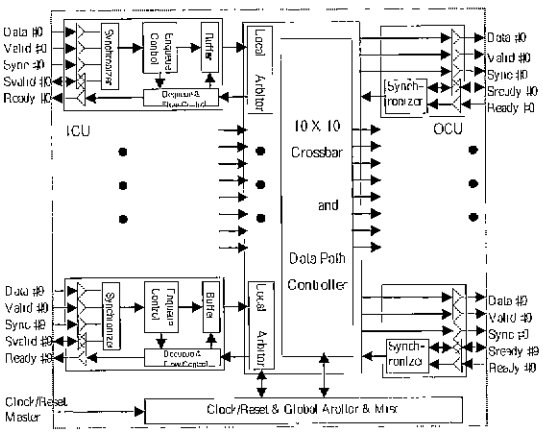

그림 4 라우터 블록도

입력 포트는 동기화기와 패킷 버퍼, 패킷 버퍼 입력 제어기, 패킷 버퍼 출력 제어 및 흐름 제어기로 구성되어 있다. 입력되는 패킷은 플릿 동기 신호에 의해서 래치되어 이중 포트 SRAM으로 구성된 패킷 버퍼에 저장된다. 패킷 버퍼의 쓰기 포트는 패킷 버퍼 입력 제어기에 의해 제어되고 플릿 동기 신호에 의해서 동기되며, 읽기 포트는 패킷 버퍼 출력 제어 및 흐름 제어기에 의해서 제어되며 라우터 동작 클럭에 동기되어 동작한다. 따라서 입력되는 패킷은 패킷 버퍼를 통과하면서 라우터 내부 동작 클럭 위상에 맞도록 동기화 된다. 동기화기는 플릿 동기 신호에 의해서 래치된 패킷 동기 신호를 라우터 클럭으로 동기화 시키며, 동기화 된 패킷 동기 신호는 패킷 버퍼 출력 제어 및 흐름 제어기로 전달된다. 패킷 버퍼 출력 제어 및 흐름 제어기는 동기화기로부터 패킷 동기 신호를 받아 중재를 요청하고 중재 결과에 따라 패킷을 읽어내어 크로스바 패스로 전송하고 흐름 제어 신호를 구동한다.

지역 중재기는 각 입력 포트에 연결되어 있으며 두 단계 우선순위와 라운드 로빈 방식을 사용하여 각 입력 포트의 전송 요구를 중재하며 전역 중재기는 브로드캐스트 요구에 대한 중재를 수행한다. 크로스바 패스와 패스 제어기는 입력 포트에서 출력 포트로 전송 경로를 제공하며 중재 결과에 따라 크로스바 패스를 설정한다. 출력 포트는 크로스바 패스를 통해 출력되는 데이터를 전송하며 다음 수신단으로부터 입력되는 흐름 제어 신호를 분석하여 지역 중재기와 전역 중재기에 전달한다.

한 패킷이 라우터를 통과하는 지연 시간은 표 2와 같이 7 클럭이 소요되며 Xcent-Net에서 최장 거리의 노드간 에서 패킷 전체가 전송되는 데 걸리는 전송 지연 시간은 45 클럭이다 임의의 두 노드간에 하나의 패킷 모두가 전송될 때 걸리는 전송 지연 시간은 (식 1)에 따라 계산된다.

표 2 클럭별 라우터 동작

Clock No	Description
1	Latch
2 - 3	*Valid*ª synchronization
4	synchronization of Xcent cloud
5	Arbitration
6	crossbar set up
7	data out

표 3 라우터 ASIC 구현 사항

Target library	LCA300K (LSI)
Technology	0.6 μ CMOS gatearray
Metal layer	2
Gate count	50K gates
Signal count	276
I/O buffer	TTL (4 mA output driver)
Package	383 CPGA(22 X 22)
Operation speed	66 MHz
Operation voltage	5 V + 10%
Power consumption	3.5 W

Tp = (# of stages) X 7 + (# of data flits - 1) (식 1)

라우터는 Verilog-HDL을 사용하여 설계되었다. LSI의 LCA300K 라이브러리를 사용하여 논리 합성을 수행한 후 0.5μ CMOS 게이트 어레이로 제작되었으며, 383핀 세라믹 PGA 패키지에 실장되었다 표 3은 라우터 구현 사항을 정리한 표이고 그림 5는 라우터의 실물 사진이다.

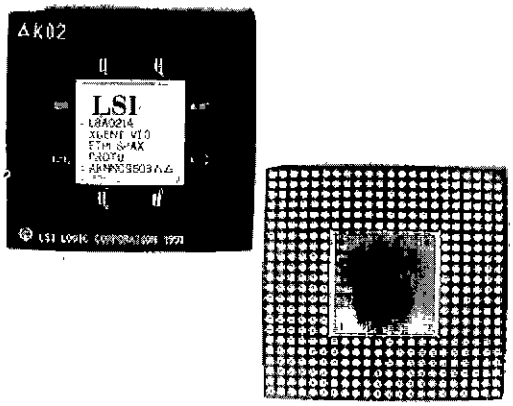

그림 5 라우터 전면과 후면 사진

4.2 백플레인

백플레인은 라우터 클라우드 2세트와 필요한 커넥터를 실장하기 위한 것으로 16층 다층 기판으로 설계, 제작되었다. 근원지 동기 전송 방식을 지원하기 위하여 동일한 포트를 구성하는 데이터 신호간에 정교한 스큐 제어가 요구되었다[15]. 백플레인과 케이블, 각 구동소자등은 정교한 임피던스 제어와 복잡한 신호 분석을 통하여 설계되었다. 백플레인상의 단일점 임피던스는 일온스 6 mil 트래이스의 Dual Stripline으로 구성하여 53Ω으로 제어되어 있다. 그림 6은 임피던스 제어가 이루어진 16층 다중 기판의 구성을 보여준다.

인접한 층간 신호간섭을 줄이기 위하여 층간 신호 배열은 상호 수직관계를 갖도록 설계되었으며, 동일한 층간에 평행한 신호선은 50 mils 이상 거리를 두고 라우팅 하였다. 신호 반사 영향을 줄이기 위하여 신호선의 모서리는 45도 각도로 처리되었으며 동일 포트를 구성하는 신호선간의 스큐는 300ps 이하로 제어되었다. 그림 7은 구현된 백플레인의 모습을 보여 준다.

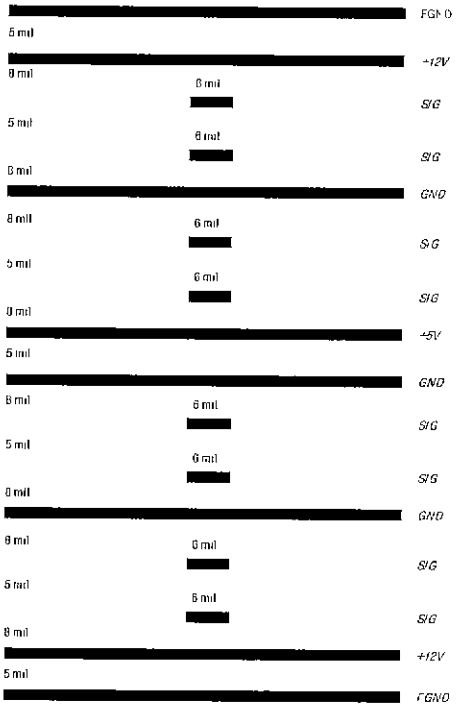

그림 6 16층 다중 기판으로 구성된 백플레인 단면도

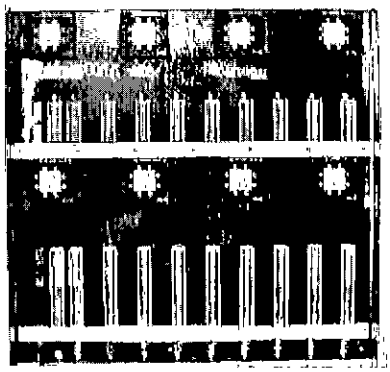

그림 7 Xcent-Net 라우터가 실장된 백플레인 모습

그림 8은 라우터에 의해서 구동되는 4mA 단일점 TTL 신호선 측정 결과로 사각형으로 표시된 것이 라우터의 출력단 신호이고 삼각형으로 표시된 것은 라우터 출력단으로 부터 20 인치 거리인 커넥터 핀에서 측정한 결과이다. 측정 파형

은 하강 에지에서 미세한 흔들림이 보이나 사각형 파형이 안정되게 유지에 되었으며 약 2ns의 전송 지연 시간을 보인다.

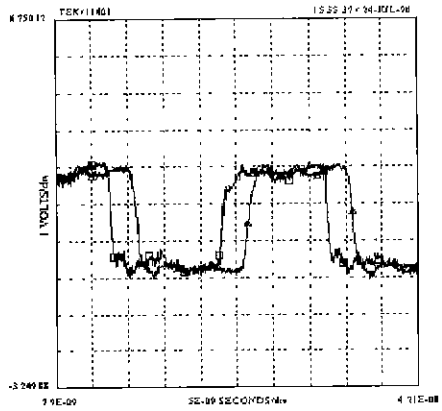

그림 8 백플레인 상에서의 신호 측정 결과

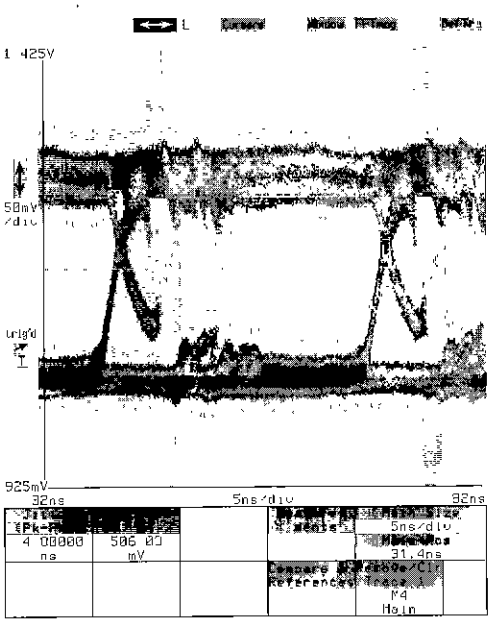

그림 9 LVDS 신호 측정 결과

백플레인과 백플레인을 연결하는 케이블은 LVDS(Low Voltage Differential Signal)을 사용하였다. 라우터에 의해서 구동되는 신호는 케이블 정합 장치에 의해서 LVDS신호로 변환되어 전송된다. 최대 33피트의 케이블 전송에 있어서 신호 감쇠를 고려하여 30 gagues 이상의 굵기를 갖는 STP(Shielded Twisted Pair) 케이블을 사용하였다. 그림 9는 측정한 LVDS 신호선으로 National Semiconductor사의 DS90C031 LVDS 드라이버를 사용하여 33MHz의 신호선을 구동하고 33피트 떨어진 수신단에서 측정한 결과이다. 그림에서 볼 수 있듯이 차동신호 상호간에 전압 분배가 평형을 이루고 있으며 전압차는 약 230 mV로 약 170mV의 전압이 감쇠되었으나 신호 수신이 가능한 수준을 유지하였다.

5. 결 론

Xcent-Net은 상용응용을 위한 데이터베이스 엔진 탑재용 대규모 클러스터 시스템인 SPAX를 위한 연결망이다. SPAX 시스템은 마이크로 커널 기반형 운영체제를 탑재하여 사용자에게 단일 시스템 이미지를 제공하는 클러스터 시스템으로 대규모 상용 응용 프로그램 처리를 목적으로 개발되었다. 단일 시스템 이미지 제공을 위해서는 낮은 전송 지연 시간을 갖는 연결망이 요구되며, 또한 빈번한 제어 메시지와 페이지 단위의 대규모 데이터 메시지를 동시에 효과적으로 처리할 수 있어야 한다. 상용 응용 프로그램의 특성상 고확장성 및 고가용도 요구사항 또한 필수적이다. Xcent-Net은 상용응용 처리용 클러스터 시스템의 요구사항에 맞게 개발된 클러스터 연결망으로 클러스터 내부 연결망과 클러스터 간 연결망으로 구성되는 계층 크로스바 연결망이다. 이중화 구조를 통하여 연결망 상의 단일점 고장을 감내할 수 있으며 연결망의 분할 및 구성이 유연한 특징을 갖고 있다.

Xcent-Net을 사용하는 SPAX 시스템은 마이크로커널 기반 운영체제가 탑재되어 시험을 마쳤으며, OPS(Oracle Parallel Server)를 시험 탑재하여 응용 서비스 시연을 하였다. Xcent-Net 설계 및 개발 기술은 범용 클러스터 시스템에서 사용 가능한 차세대 SAN(System Area Network) 연결망 개발에 응용할 예정이다.

참고문헌

[1] D J.Torrellas and D. Koufaty, "Comparing the Performance of the DASH and Cedar

Multiprocessor for Scientific Applications", Proc of ICPP'94, vol.2, pp304-308, August, 1994.

[2] L. Barroso and M. Dubois, "Performance Evaluation of the Slotted Ring Multiprocessor", IEEE TOC, vol.44, no 7, pp878-890, July, 1995.

[3] W.J.Hahn, S.H.Yoon, and K.W.Rim, "Design of the New Parallel Processing Architecture for Commercial Applications," Journal of Korean Institute of Telematics and Electronics, vol.31, no.5,pp897-907, 1996.5.

[4] Gartner group, Second Annual Advanced Technology Groups Conference),1992

[5] T.E.Anderson, D.E.Culler and D.A. Patterson, "A Case for NOW", IEEE Micro, vol.15, no.1, pp54-64, 1995.2.

[6] W.J.Hahn, S.K.Yoon, K.W.Lee and M.Dubois, Evalution of a Cluster-based System for OLTP Application, ETRI Journal, Vol. 20, No. 4, pp.301-326, December 1998.

[7] W.J.Hahn, K.W.Rim and S.W.Kim, "A Multiprocessor Server with a New Highly Pipelined Bus", Proc. of 10th IPPS, pp512-517, April, 1996.

[8] B.J.Min, S.H.Shin and K.W.Rim, "Design and Analysis of a Multiprocessor System with Extended Fault Tolerance", Proc. of IEEE 5th Workshop on Future Trends in Distributed Computing Systems, pp301-307, August, 1995.

[9] H.J.Kim, J.K.Lee and K.W.Rim, "Parallel Operating System for MPP System: Design and Implementation", Lecture note in Computer Science, pp413-422, May. 1996.

[10] R. P. Martin, HPAM:An Active Message layer for a Network of HP Workstation, proc. of Hot Interconnect II, pp. 40-58, Aug. 1994

[11] J. C. Hoe, Network Interface for Message-Passing Parallel Computation on a Workstation Cluster, proc. of Hot Interconnect II, pp. 154-163, Aug. 1994.

[12] P. Kermani and L. Kleinrock, Virtual cut-through:A new computer communication switching technique, Computer Networks, vol. 3, pp 267-186, Sept. 1976.

[13] G. A. Boughton, Arctic Routing Chip, proc. of Hot Interconnect II, pp. 164-172, Aug. 1994.

[14] K.Park, J.S.Hahn, S.W.Sim and W.J.Hahn", Xcent-Net' A Hierarchical Crossbar Network for a Cluster Based Parallel Architecture", Proc. of 8th PDCS, October, 1996.

[15] S.W.Sim and A Martinez, "Switching Router Network in a Parallel Computer", Proc. of Design SuperCon'97, January, 1997.

한 우 종

1981 고려대학교 전자공학과 학사
1984 고려대학교 전자공학 석사
1995 고려대학교 전자공학 박사
1985 한국전자통신연구소 연구원
1986~1988 미국 AIT사 파견 연구원
1989 전자계산기 분야 기술사
1997~1998 한국전자통신연구원 프로세서연구실장
1998~현재 한국전자통신연구원 하드웨어구조연구팀장, 책임연구원
관심분야:컴퓨터구조, 마이크로프로세서 구조, 병렬처리 구조, 상호연결망, 계층메모리 구조 등
E-mail:wjhan@computer.etri.re.kr

박 경

1991 건북대학교 컴퓨터공학 학사
1993 전북대학교 컴퓨터공학 석사
1993~현재 한국전자통신연구원 연구원
1994~1997 SPAX 병렬처리 시스템 개발 사업 참여
1996~1998 단일칩 다중프로세서 구조 개발 과제 참여
1998~현재 고성능멀티미디어 서버 개발 사업 참여
관심분야:마이크로프로세서 구조, 병렬처리 구조 등
E-mail:kyong@computer.etri.re.kr