

이동통신시스템의 프로세서간 통신성능향상을 위한 넉아웃 스위치의 구조설계

박 상 규[†] · 김 재 홍^{††} · 이 상 조^{†††}

요 약

현재 이동통신시스템의 내부 프로세서간 통신망은 메쉬 토폴로지 형태의 단일 버스를 이용하기 때문에 앞으로 B-ISDN과 연계될 대용량의 트래픽 처리에 한계가 있다. 또한, 이동통신시스템에서는 일반 ATM망에서의 고정 길이가 아닌 가변 길이 패킷을 사용하고 있으므로 ATM스위치를 그대로 이용할 수는 없다. 일부의 구현에서 가변길이를 지원하는 스위치를 제시하고 있으나, 내부 동기화를 위한 전처리, 비트 지연 등의 문제가 있다. 본 논문에서는 추가적인 처리없이 가변 길이의 패킷을 처리할 수 있는 경합집속기를 설계하였다. 또한 제안된 경합집속기는 입력 인터페이스에서 패킷 시작 신호를 위한 지연이 없다. 따라서, 효율적으로 패킷을 처리해 줄 수 있으며, 기존의 경합집속기에서 N 비트 시간의 지연이 걸리던 것을 $\lceil \log_2 N \rceil + 1$ 비트 시간 정도로 감소시켰다. 이에 따라 가변 길이 넉아웃 스위치를 적용한 부분 메쉬 토폴로지의 버스 구조의 이동통신시스템은 B-ISDN망과 연계된 대용량의 트래픽을 처리할 수 있을 것으로 기대된다.

The Design of Knockout Switch Structure For Improving Performance of Inter-Processor Communication in Mobile Communication System

Sang Kyu Park[†] · Jae Hong Kim^{††} · Sang Jo Lee^{†††}

ABSTRACT

There are limitations to process high bandwidth traffic in B-ISDN with mesh-topology single bus architecture of current mobile communication system. And, it is impossible to import ATM switch using fixed length packet rather than variable length packet. Some implementations are able to process variable length packet, but there are some problems such as pre-processing for synchronization and bit delay. In this paper, we design a concentrator that can manipulate variable length packet without additional pre-process. There is no bit delay for packet starting signal in input interface. So it is more efficient to process packets, such that the concentrator can reduce the processing time as $\lceil \log_2 N \rceil + 1$ bit-time rather than N bit-time delay in ordinary concentrator. It is expected that the mobile communication system with partial mesh topology bus adopting the knockout switch architecture can process high bandwidth traffic in B-ISDN.

† 정 회 원: 연암공업전문대학 컴퓨터정보기술과 조교수
†† 정 회 원: 경북대학교 대학원 컴퓨터공학과 박사과정
††† 정 회 원: 경북대학교 컴퓨터공학과 교수
논문접수: 1995년 9월 23일, 심사완료: 1996년 1월 12일

1. 서 론

초고속 통신망이란 단어가 이제 흔하지 않게 거론되는 현실에서 개인 휴대 통신에서도 음성뿐 아니라 다양한 멀티미디어 데이터의 전송 요구를 신속하게 처리할 수 있는 이동통신시스템을 필요로 할 때가 곧 오게 될 것이다. 이는 현재 연구되고 있는 B-ISDN망과의 연계를 위해서도 이동통신시스템의 처리능력, 그 중에서도 빠른 트래픽을 위한 내부 처리 능력 향상이 조속히 이루어져야 할 것으로 생각된다.

세계 각국에서는 이러한 요구를 만족시키기 위해 다양한 방식의 이동통신시스템을 개발, 사용 중이며, 국내에서는 디지털 방식 중의 하나인 CDMA 방식을 개발하였다[3]. 이러한 이동통신시스템 내에서는 기지국 내, 제어국 내 및 교환기 내부를 구성하는 프로세서 간 통신을 위한 망이 필요하며, 이 프로세서 간 통신망은 다수의 고성능 마이크로프로세서로 구성되는 다중 프로세서 구조로 개개의 프로세서들은 내부 통신망을 통하여 서로 연결되어 있다.

이러한 프로세서간 통신망의 성능이 곧 시스템 전체의 성능을 크게 좌우한다는 점에서 통신망의 적절한 설계가 더욱 중요해지고 있다[2, 5, 7].

현재의 이동통신시스템 내의 통신망은 그 단위 통신망이 가상 토큰 버스의 형태로 구성되어 있으며[1], 이 구조는 하나의 버스를 여러 개의 프로세서가 공유하는 형태를 가진다. 현재 시스템의 운용에서 많은 최적화 작업을 거쳐 운영되고 있으나, 이미 이러한 토폴로지 위에서는 다가오는 통신 수요에 적절히 대처하기에는 트래픽의 수용 능력의 한계에 다다른 것 같다. 따라서, 이러한 구조로는 B-ISDN망과의 연계에 따른 다양한 사용자 요구를 서비스해 줄 수 없으며 이의 해결을 위한 새로운 프로세서 간 통신망의 설계가 요구되고 있다.

이러한 기존 단위 통신망의 성능의 한계를 극복하기 위해 이 논문에서는 스위치 구조를 프로세서 간 통신망으로 사용하는 방법을 제안한다. 스위치는 N 개의 입력과 N 개의 출력으로 이루어지고, 적절한 제어로 내부의 스위칭 구조를 통해 프로세서 간 통신망으로 사용할 수 있다. 스위치 구조로 넉아웃, roxanne, coprin, athena, starlite, moonshine 등 여러가지가 제안되었으며[16], 이 중에서 넉아웃스위치는 구조가 단

순하고, 내부 블록킹이 발생하지 않으며, broadcasting 및 multicasting을 쉽게 구현할 수 있다는 장점을 가진다. 또한, 고장이 발생했을 때 피해를 최소화할 수 있고, 출력 버퍼형의 구조로 되어 있어 최고의 성능을 가지는 것으로 평가되고 있다[11, 12, 16, 17]. 그리고, 넉아웃 스위치는 버스 구조를 가지므로 기존 시스템의 단위 통신망과의 호환성 측면에서 유리하여 기존의 기술을 활용할 수 있다.

넉아웃 스위치에는 버스 인터페이스 부분에 경합 문제를 해결하기 위해 경합집속기가 있으며, [11]에서는 가변 길이의 패킷을 처리하기 위해 패킷 유효 신호만을 가지고 경합집속을 한 후에 데이터 스위치를 통해 데이터를 보내는 방법을 제안하였다. 그러나, 이 구조는 데이터 스위치가 별도로 필요하므로, 이 논문에서는 데이터 스위치를 따로 두지 않고 가변 길이의 패킷을 처리할 수 있는 경합집속기를 설계하였다. 제안된 경합집속기는 가변 길이의 패킷을 처리해 줄 수 있을 뿐만 아니라, 입력 인터페이스에서 패킷 시작 동기를 위한 지연없이 패킷을 버스로 입력받아 처리할 수 있다. 따라서, 효율적으로 패킷을 처리해 줄 수 있으며, 기존의 경합집속기에서 N 비트 시간의 지연이 걸리던 것을 $\lceil \log_2 N \rceil + 1$ 비트 시간 정도의 크기로 감소시켰다.

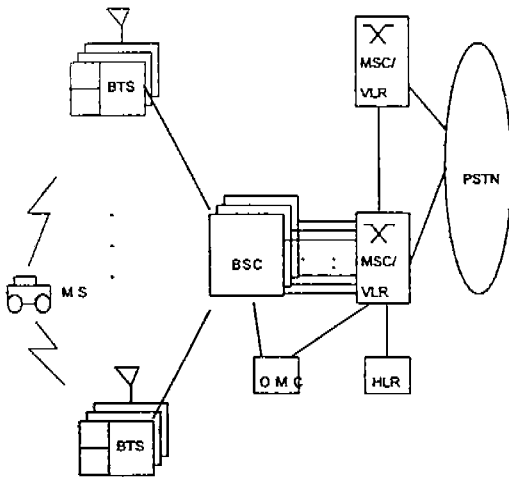
제안된 경합집속기를 사용한 넉아웃 스위치를 이동통신시스템의 상호 연결 통신망의 단위 통신망으로 사용하여, 전체 망의 구조는 기존의 부분적인 폐쇄 토폴로지를 그대로 사용하여 대용량의 트래픽을 처리할 수 있을 것으로 기대된다.

2장에서는 기존 시스템의 프로세서 간 통신망의 구조를 살펴보고, 수학적 해석과 시뮬레이션을 통해 기존 망의 성능의 한계를 알아본다. 3장에서는 이동통신 시스템에서 사용될 수 있는 프로세서 간 통신망의 구조를 제안하고, 프로세서 간 통신망의 근간으로 사용될 넉아웃 스위치에 대해 살펴본다. 그리고, 가변 길이의 패킷을 효율적으로 처리해 주는 경합집속기를 설계한다. 4장에서는 제안된 시스템의 성능을 경합집속기에서의 패킷 손실률과 지연을 중심으로 분석하고, 5장에서 결론을 맺는다.

2. 기존 시스템의 구조 및 성능의 한계

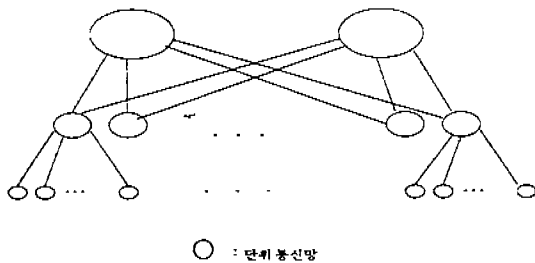
2.1 기존 시스템의 구조

CDMA 이동통신시스템은 그림 1과 같이 이동국, 기지국, 제어국, 이동통신 교환기, 홈 위치 등록국, 방문자 위치 등록국, 운용보전국으로 구성된다[6, 7].

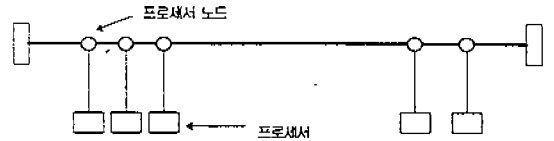


(그림 1) 이동통신시스템의 구성
(Fig. 1) The Configuration of Mobile Communication System

이들 구성요소 중 기지국, 제어국 및 교환기에서는 다수의 프로세서 간 통신을 위한 통신망이 제공된다. 이동통신시스템 내에 있는 프로세서 간 통신망은 이동교환기의 통신망 및 제어국 내의 CIN, 기지국 내의 BIN이 있다. 이들 상호 연결 통신망들은 단위 통신망들이 그림 2와 같이 부분적인 메쉬 토폴로지 형태로 연결되어 있으며, 각각의 단위 통신망은 그림 3과 같이 하나의 버스를 N개의 프로세서가 공유하는 가상 토큰 버스 형태로 구현되어 있다[1, 2, 5].



(그림 2) 단위통신망의 상호 연결 구조
(Fig. 2) The Structure of Interconnection Network



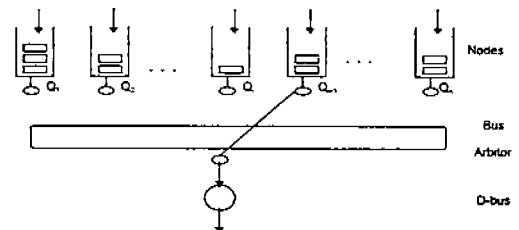
(그림 3) 단위 통신망의 구조
(Fig. 3) The Structure of a Communication Network Unit

2.2 기존 단위 통신망의 성능

이 절에서는 기존 단위 통신망의 구조를 다수의 큐와 단일 서버를 갖는 사이클릭 서비스 시스템으로 모델링하고 해석적 방법과 시뮬레이션을 통하여 성능을 분석한다.

일반적으로 사이클릭 서비스 시스템은 하나의 서버가 여러 큐를 서비스하는 형태를 가지고 있으며, 서버가 큐를 방문하였을 때 큐에 있는 서비스 가능한 패킷에 대한 서비스 정책에 따라 다음 3가지로 분류할 수 있다. 첫째, 소모성 서비스 정책은 큐에 있는 모든 패킷을 서비스하고 다음 큐로 넘어가는 서비스 정책이다. 둘째, 게이트형 서비스 정책은 서버가 큐에 도착한 시점 이전에 도착한 패킷만 서비스하는 정책이다. 마지막으로, 제한된 서비스 정책은 큐에 존재하는 패킷 중 몇 개의 패킷만을 서비스하고 다음 큐로 진행하는 서비스 방식이다. 제한된 서비스 정책 중에서 한개의 패킷만을 서비스하고 다음 큐로 이동하는 방식이 I-limited 또는 비소모성 서비스 정책이다[2].

기존 시스템은 스위치 오버헤드를 가지는 다중 큐 사이클릭 서버로 모델링이 가능하고, 이 모델은 M/G/1 FCFS 서버 모델이라고 가정할 수 있다. 기존 시스템의 큐잉 모델은 그림 4와 같이 표현될 수 있으며, 비소모성 서비스 방식을 구현한다. 비소모성 서비스 정



(그림 4) 단위 통신망의 큐잉 모델
(Fig. 4) Queueing Model of a Communication Network Unit

책은 특히 실시간 처리와 관련한 시스템에서 노드간 평균 대기 시간의 분산을 최소화할 필요가 있는 경우에 적합한 서비스 정책이다.

스위치 오버헤드를 가지는 사이클릭 시스템의 경우 서비스 전략에 따라 성능이 크게 좌우되며, 이 시간을 최소화하기 위해 기존 시스템에서는 기본적인 사이클릭 시스템을 수정한 두개의 토큰을 사용한 버스 중재를 구현하였다[1].

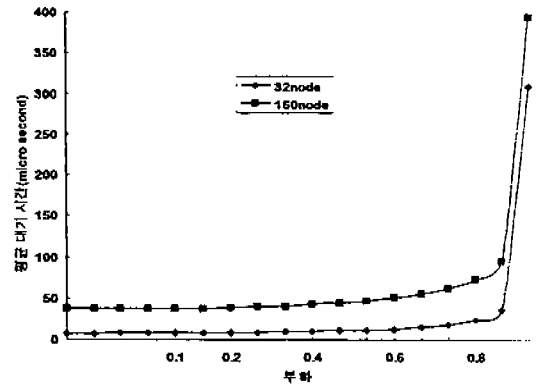
비소모성 사이클릭 서비스의 경우 몇 가지 연구가 있었으나 Boxma와 Meister의 방법이 시뮬레이션 결과와 가장 근사하므로, 이 논문에서는 Boxma와 Meister의 방법을 사용하여 근사식을 구한다. Boxma와 Meister에 의한 임의의 큐에서의 평균 대기 시간은 다음과 같이 표현된다[8, 10].

$$E_{w_i} = \frac{1 + \rho + \rho_i}{1 - \rho - \lambda S} \cdot \frac{1 - \rho}{(1 - \rho)\rho + \sum_{j=1}^N \rho_j^2} \cdot \left[\frac{\rho}{2(1 - \rho)} \sum_{j=1}^N \lambda_i \beta_i^2 + \frac{\rho}{2S} \sum_{j=1}^N \Psi_j^2 + \frac{S}{2(1 - \rho)} \sum_{j=1}^N \rho_j (1 - \rho_j) \right] \quad (1)$$

식 1에서 λ 는 큐 i 에 대한 패킷 도착률을 나타내고, N 은 시스템에 있는 큐의 수를 나타낸다. 그리고, β_i 는 큐 i 에 있는 패킷의 서비스 시간으로 버스의 속도와 메시지의 길이로부터 구할 수 있다. 큐 i 의 버스 사용률 ρ_i 는 $\lambda_i \beta_i (1 \leq i \leq N)$ 로 표현되고, 전체의 버스 사용률 ρ 는 $\sum_{i=1}^N \rho_i$ 또는 $\sum_{i=1}^N \lambda_i \beta_i$ 로 나타낼 수 있다. 그리고, β_j^2 과 Ψ_j^2 는 각각 서비스 시간의 분산과 큐 j 에서 $j+1$ 까지의 idle 스위치 오버 시간의 분산을 나타내며, S 는 한 사이클 동안의 스위치 오버 시간의 합을 나타낸다.

식 1에 의한 수학적 해석과 시뮬레이션을 통해 기존 시스템의 성능을 분석해 보면 그림 5와 같은 결과를 얻을 수 있다. 그림에 나타난 값은 시뮬레이션 결과이며 수학적 해석에 의한 결과도 이와 비슷한 값을 가진다. 그림 5는 패킷의 길이, 버스의 속도, idle 스위치 오버 시간, 서비스 스위치 오버 시간을 각각 48 바이트, 80Mbps, 0.4 μ sec, 0.2 μ sec로 하여 구한 값이다. 결과로 나타난 그래프는 패킷이 버스에 도착하여 버스를 사용하기까지의 대기 시간을 부하의 변화에 따라 나타낸 값으로, 노드의 수가 많아지면 이 값은 증

가하게 된다. 이 결과는 부하가 같더라도 노드의 수가 많으면 스위치 오버 시간이 증가하여 패킷이 서버스 받기까지의 시간이 길어짐을 말해준다.



(그림 5) 부하에 따른 메시지의 평균 지연시간 (Fig. 5) Message Mean Waiting Time According to Load

기존 시스템은 이와 같이, 같은 부하일지라도 노드의 수가 증가하면 패킷의 대기 시간이 길어지게 되어 망의 성능이 저하된다. 또한, 부하 0.8~0.9 근처에서 패킷 대기 시간이 급격한 증가로 병목현상을 나타낸다. 근본적으로 기존의 시스템은 하나의 버스를 다수개의 프로세서가 공유하고 있는 형태이므로, 이 구조를 그대로 유지하면서 버스 중재 알고리즘을 보완하고, 버스의 속도를 빠르게 하더라도 트래픽 처리능력에 한계가 있다. 따라서, 단일 버스를 사용하고 있는 현재의 이동통신시스템의 단위 통신망을 대신할 새로운 통신망이 설계되어야 한다.

3. 넉아웃 스위치를 이용한 구조

이 장에서는 기존의 이동통신시스템의 단위 통신망이 가지는 처리 용량의 한계를 극복하기 위한 새로운 단위 통신망의 구조를 제안한다. 기존의 단위 통신망이 하나의 버스를 다수개의 프로세서가 공유하는 데서 근본적인 성능의 한계를 가지고 있으므로, 이 한계를 극복하기 위해 수 내지 수십 Gbps의 성능을 가지는 넉아웃 스위치를 프로세서 간 통신망으로 사용하는 방법을 제안한다. 스위치는 N 개의 입력과 N 개의 출력으로 이루어져, 입력이 들어왔을 때 이를

내부의 처리를 통해 목적한 출력으로 보내주는 역할을 하는 것이므로 프로세서 간 통신망으로 사용할 수 있다. 넥아웃 스위치에는 버스 인터페이스 부분에 폭주 문제를 해결하기 위해 경합집속기가 있으며, [11]에서는 가변 길이의 패킷을 처리하기 위해 패킷 유효 신호만을 가지고 경합집속을 한 후에 데이터를 데이터 스위치를 통해 보내는 방법을 제안하였다. 그러나, 이 구조는 데이터 스위치가 필요하므로, 이 논문에서는 데이터 스위치를 따로 두지 않고 경합집속을 하면서, 데이터도 함께 보낼 수 있는 경합집속기를 설계한다.

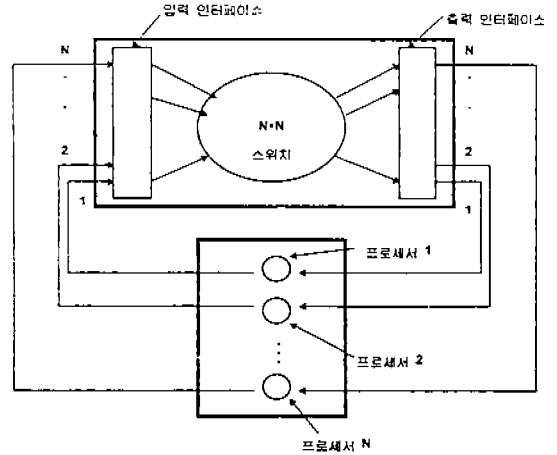
3.1절에서는 스위치를 이용한 프로세서 간 통신망의 구조에 대해 설명하고, 3.2절에서는 기존의 넥아웃 스위치의 기본 구조와 개념에 대하여 설명한다. 마지막으로 3.3절에서 가변 길이의 패킷을 입력 인터페이스에서 패킷 시작 동기를 위한 지연없이 처리하기 위한 새로운 경합집속기를 설계한다.

3.1 제안된 프로세서 간 통신망 구조

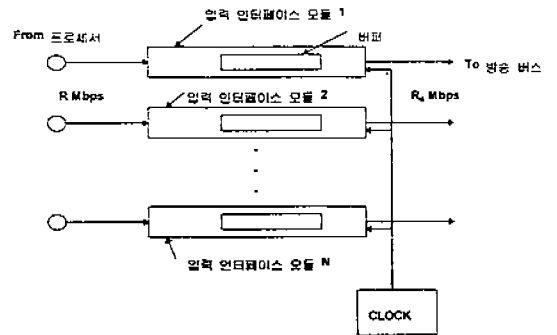
이동통신시스템을 위한 프로세서 간 단위 통신망을 넥아웃 스위치를 이용하여 설계한다. 스위치를 개념적으로 해석해 보면 N개의 입력과 N개의 출력이 있는 시스템으로 볼 수 있다. 입력 부분을 프로세서로부터 나오는 패킷을 받아들이는 부분으로 보고, 출력 부분을 프로세서로 패킷을 전달하는 부분으로 보면 스위치는 모든 프로세서로부터의 패킷을 어느 프로세서에게나 전달할 수 있는 기능을 가지고 있다고 볼 수 있다. 이러한 개념을 그림으로 나타내면 그림 6과 같다. 이 논문에서는 그림 6과 같이 스위치를 프로세서 간 통신을 위한 망으로 사용한다.

그림 6에서 프로세서들은 이동통신시스템 내의 단위 통신망을 이루고 있는 프로세서들이다. 스위치의 각각의 방송 버스로 입력되는 부분에 있는 입력 인터페이스는 스위치 내부와의 속도 차이를 고려하여 최대 패킷 길이의 버퍼를 하나 가지고 있으며, 버퍼에 입력된 패킷을 스위치 내부 클럭에 맞게 방송 버스로 전달한다. 그림 7은 입력 인터페이스를 나타낸 것이다.

프로세서와 입력 인터페이스 사이의 링크의 속도(R Mbps)와 스위치 내부의 클럭 속도(R_0 Mbps)가 다를 수 있으므로 입력 인터페이스에 들어 온 패킷을 스위치 내부 속도로 변환하여 방송 버스로 전달한다.



(그림 6) 스위치를 이용한 통신망의 개념도
(Fig. 6) The Concept Structure of a Communication Network Unit Using Switch



(그림 7) 입력 인터페이스
(Fig. 7) Input Interface

각 입력 인터페이스 모듈로 들어오는 패킷들은 다른 입력 인터페이스 모듈의 패킷과의 동기 없이 바로 방송 버스로 전달된다. 따라서, 입력 인터페이스에서 지연을 가지지 않는다. 원래의 넥아웃 스위치는 경합집속기에서 신호의 동기를 위해 $R_0/27$ 클럭에 패킷 시작 동기를 맞추어 패킷을 방송 버스로 전달하였으나, 이 논문에서는 패킷간의 동기가 필요없는 경합집속기를 사용하므로 입력 인터페이스에서의 지연이 필요없게 된다. 출력 인터페이스는 버퍼로부터 도착한 패킷에 헤더를 붙여 프로세서로 전송한다. 또한, 스위치 내부 속도(R_0 Mbps)를 출력 인터페이스와 프로세서 사이의 링크의 속도(R Mbps)로 변환하는 기능을 가진다.

$N \times N$ 스위치로 사용하는 넉아웃 스위치는 기존의 넉아웃 스위치에서 경합집속기를 수정한 것이다.

3.2 넉아웃 스위치

넉아웃 스위치는 패킷 스위칭을 위해 제안된 구조로, 완전히 상호 연결된 구조를 사용하며, 입력으로 들어온 모든 패킷들은 입력 인터페이스를 거쳐, 각각의 방송 버스를 통하여 스위치 내부에 전달된다. 모든 버스 인터페이스는 이들 각각의 방송 버스를 접근할 수 있도록 링으로 연결되어 있으며, 따라서 하나의 출력 인터페이스는 모두 N 개의 교차점을 가진다. 출력 인터페이스가 모두 N 개이므로 전체적으로 N^2 개의 교차점을 가진다. 패킷이 각각의 입력 인터페이스로 들어가 독립적인 버스를 통해 버스 인터페이스로 보내지므로 내부 블로킹이 발생하지 않는 구조적인 장점을 가진다. 버스 인터페이스를 통과한 패킷은 출력 인터페이스를 통하여 출력이 된다[17].

이러한 구조에서 서로 다른 여러 개의 방송 버스에서 하나의 버스 인터페이스로 동시에 패킷이 도착하면 폭주 문제가 발생하며, 이 문제를 효율적으로 해결해 주기 위해 버스 인터페이스 내부에 경합집속기를 가지고 있다.

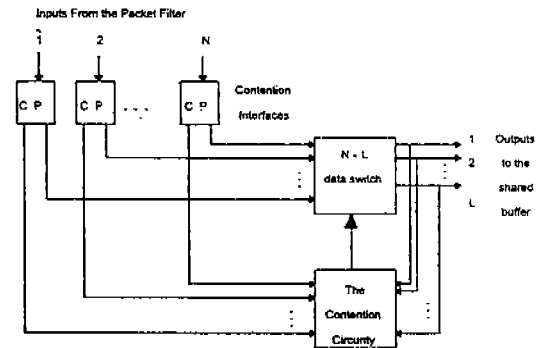
3.3 경합집속기의 설계

3.3.1 기존의 경합집속기

패킷 필터를 통과한 유효한 패킷들은 $N \times L$ 경합집속 기능을 수행하는 경합집속기에 들어가게 된다. 경합집속기는 k 개의 패킷이 들어 왔을 때 $k \leq L$ 이면 모든 패킷을 출력시키고, $k > L$ 일 경우 L 개의 패킷만을 출력시키고 $k-L$ 개의 패킷은 폐기하는 역할을 한다. 이러한 기능은 동시에 다수 개의 패킷이 도착했을 때 모든 패킷을 처리할 수 없으므로 QoS의 범위 내에서 이를 적절히 처리해 주기 위해 필요하다. 기존의 넉아웃 스위치는 가변 길이의 패킷을 처리해 주기 위해 그림 8과 같은 경합집속기를 제공하였다.

그림 8에서 경쟁 인터페이스들의 C신호는 패킷의 유효 여부를 나타내는 신호이고, P는 지연된 데이터이다. C신호들은 경쟁 회로에 입력되어 넉아웃 방식에 의해 L 개의 승자를 선택한다. 이 결과는 데이터 스위치에 입력되어 선택된 패킷이 데이터 스위치를 통해 공유 버퍼로 들어갈 수 있게 된다. 데이터 스위치

에서 출력을 한 후 리셋 신호를 경쟁 회로에 보내 사용이 끝난 버퍼를 다시 사용가능하게 한다. 패킷의 길이가 가변이므로 어떤 데이터는 다음 클럭에 의한 데이터들이 들어올 때에도 전송이 끝나지 않을 수 있다. 이러한 경우는 리셋 신호를 경쟁 회로에 보내지 않게 하여 그 버퍼를 계속 사용하게 함으로써 다음 동기의 데이터들은 그 버퍼를 사용하지 않게 한다. 여기에서 데이터들은 버스 입력 인터페이스에서 일정한 클럭에 동기되어 버스로 전달되며 따라서, 경쟁 회로는 동기된 신호를 가지고 경합집속을 하게 된다.

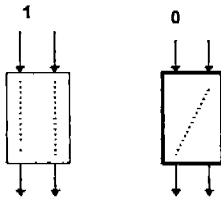


(그림 8) 가변길이 패킷을 지원하는 $N \times L$ 경합집속기 (Fig. 8) $N \times L$ Concentrator for Variable-length Packet

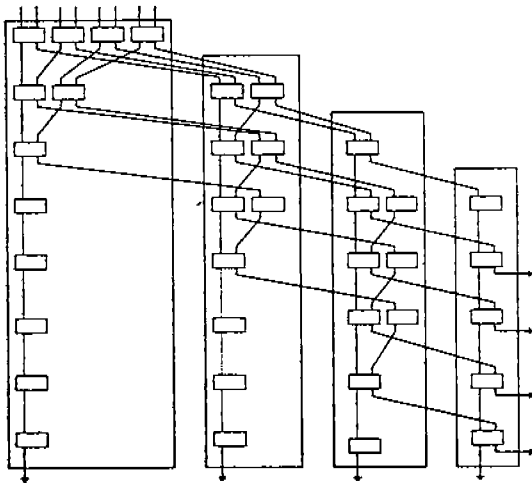
경쟁 회로는 2×2 스위칭을 하는 그림 9와 같은 경쟁 스위치 소자를 기본 단위로 하여, 이들을 그림 10과 같이 토너먼트 방식으로 연결함으로써, 동시에 들어온 패킷들을 폐기할 것인가 아닌가를 결정하고, 폐기하지 않는 패킷들에 대해서는 이 결과를 데이터 스위치로 보내서 패킷이 공유 버퍼로 전송이 되도록 한다.

경쟁 스위치 소자에서의 기본적인 경쟁 원리는 그림 9와 같이 왼쪽의 패킷이 유효한 패킷일 경우에는 오른쪽 패킷의 유효성 여부에 상관없이 각 C신호의 경로를 왼쪽 그림과 같이 제공해 주고, 왼쪽의 패킷이 유효한 패킷이 아닐 경우에는 왼쪽의 C신호는 폐기되고 오른쪽의 C신호를 왼쪽의 경로를 통해 전송되도록 하는 방법을 사용한다.

경쟁 인터페이스에서 들어온 입력은 경쟁 회로의 첫번째 단으로 입력된 후, 경쟁 스위치에서 그림 9에서 나타낸 경쟁 원리로 경쟁을 하며, $N \times L$ 경쟁 회로일 경우 경쟁에서 L 번 지면 그 신호는 폐기가 된다.



(그림 9) 경쟁 스위치 소자에서의 경쟁
(Fig. 9) The Contention in Contention Switch Element



(그림 10) 경쟁회로의 기본 골격
(Fig. 10) The Backbone of Contention Circuitry

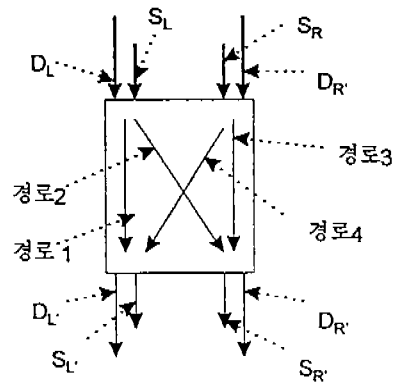
그림에서 하나의 입력이 들어와 하나의 출력이 나가는 것은 지연 스위치로 한 비트 시간 동안 지연을 시키는 역할을 한다. 이 스위치는 신호의 동기를 위해 필요하다.

이러한 방식은 입력 인터페이스에서 패킷 시작 동기를 맞추어 버스에 입력시키기 때문에 입력 인터페이스에서 지연을 가지게 되고, 패킷 유효 신호(C신호)로 경쟁을 하는 동안 N비트의 지연이 발생된다. 또한, 경쟁을 위한 경쟁 회로와 패킷이 전달되는 데이터 스위치가 따로 구성되어 있으므로 추가적인 하드웨어를 가지게 된다. 따라서, 이러한 단점을 개선하기 위해 다음 절에서는 가변 길이의 패킷이 입력되었을 때 입력 인터페이스에서 지연없이 바로 입력되어 경쟁을 하면서 데이터도 함께 전달할 수 있는 경합집속기를 설계한다.

3.3.2 제안된 경합집속기

기존의 경합집속기는 경합집속을 위해 입력 인터페이스 및 경합집속기 내에서 불필요한 지연을 가지며, 경합집속을 위한 회로와 데이터 스위치가 따로 필요하다. 이것은 경쟁 회로가 경쟁 신호의 동기를 맞추어 경쟁을 하기 때문에 발생하는 결과이므로 새로운 경합집속기는 비동기적으로 경쟁을 하면서 데이터도 동시에 전달될 수 있도록 설계한다.

비동기적으로 경쟁을 하면서 데이터도 함께 전달할 수 있도록 하기 위해 기존의 경쟁 스위치 소자의 기능을 다르게 하여, $N \times L$ 경합집속 기능을 구현한다. 새로운 경쟁 스위치 소자는 그림 11과 같다. 각 스위치 소자는 그림에서 보는 바와 같이 4개의 입력이 들어와 4개의 출력이 나가는 형태로 되어 있으며, 데이터가 지나갈 수 있는 경로는 4가지가 존재한다. 그림에서 D_L, D_R 은 입력 데이터, S_L, S_R 은 이 데이터가 유효한 데이터인가 아닌가를 나타내는 신호, D'_L, D'_R 은 출력 데이터, S'_L, S'_R 은 출력 데이터의 유효성 여부를 나타내는 신호이다. 각 입력 데이터는 데이터의 유효성 여부와 경로의 존재 여부에 따라 출력의 상태가 결정되게 된다. 스위치 내부의 경로의 상태를 표현하기 위해 3 비트가 필요하며, 이 경로의 상태와 입력 데이터에 따라 출력 데이터와 다음 상태가 결정된다.

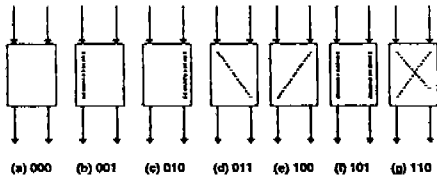


(그림 11) 2×2 경쟁 스위치 소자
(Fig. 11) 2×2 Contention Switch Element

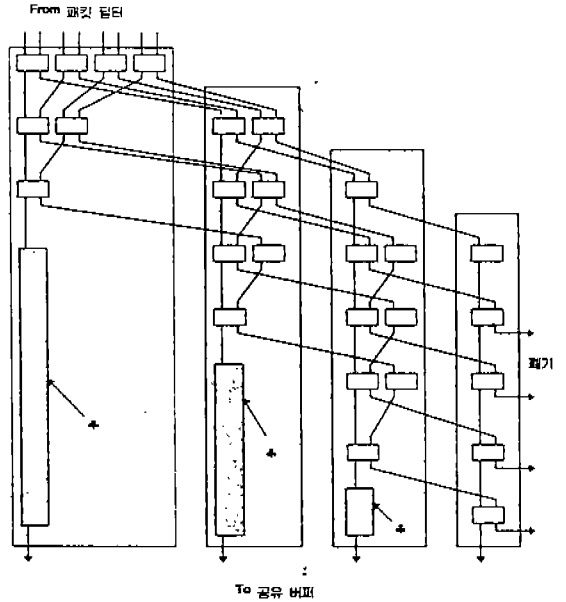
경로의 상태는 그림 12와 같이 7가지가 가능하며, 그림을 간단히 하기 위해 이하의 그림에서 데이터 선과 유효 신호를 나타내는 선을 하나로 나타낸다. 그

림에서 실선은 데이터가 전달되는 경로를 나타내며, 실선이 없는 것은 데이터가 전달되지 않음을 말한다.

초기에 모든 경쟁 스위치는 그림 12의 (a)와 같은 상태이며, 이 상태에서 왼쪽에서 유효한 데이터가 들어오면 (b)의 상태로 변하고 데이터는 경로 1을 통해 전달되고 이 데이터가 유효하다는 신호도 함께 전달된다. 이런 식으로 데이터와 유효 신호가 전달되며, 하나의 패킷에서 시작 부분이 사용하면 경로는 동일 패킷의 뒷부분이 계속 사용할 수 있게, 하나의 패킷은 모두 같은 경로를 통해 버퍼로 전달되게 된다. 패킷의 유효 상태 신호가 전달됨으로 인해 하나의 패킷이 그 경로의 사용을 마치면 바로 그 사용 경로는 다른 패킷에 의해 사용될 수 있도록 해제된다. 따라서, 경로 사용의 효율을 최대화할 수 있는 장점을 가진다.



(그림 12) 경쟁 스위치 소자에서의 가능한 경로
(Fig. 12) Possible Path of Contention Switch Element



(그림 13) 8×4 경합집속기의 기본구조
(Fig. 13) The Basic Structure of 8×4 Concentrator

두 번째 버퍼로 가게 된다. 이런 식으로 L개의 패킷의 출력 위치가 결정되게 된다.

원래의 닥아웃 스위치에서는 패킷의 동기를 위해 그림 13의 \bullet 로 표시된 부분에 지연 스위치를 두었으나, 수정된 닥아웃 스위치는 패킷이 각 버셀별로 비동기적으로 독립적으로 들어오며, 패킷의 길이도 가변이므로 지연 스위치를 둘 필요가 없어 이 부분을 제거하였다. 지연 스위치의 제거는 경합집속기 내에서의 지연을 크게 감소시킨다. 예를 들어, 8×4 경합집속기의 경우 지연 스위치를 두었을 때, 모든 패킷은 동일하게 8 비트 지연을 가진다. 그러나, 지연 스위치를 제거하면 첫 번째 버퍼로 가는 경로로 패킷이 전달될 경우 3 비트 지연만을 가지고, 두 번째, 세 번째, 네 번째 경로로 패킷이 전달될 경우에도 각각 5 비트, 7 비트, 8 비트의 지연을 가진다. 패킷이 i번째 버퍼로 가게될 확률은 시스템의 부하가 0.8일 때를 예로 들면, 각각 82.4%, 15.2%, 2.2%, 0.2% 이므로 8×4 경합집속기일 때 지연 스위치를 제거함으로써 57.5% 정도의 지연 감소 효과를 가진다. 지연의 감소는 N이 증가할수록 많아지게 된다. 이 확률을 구하는 식은 4장에서 설명한다.

앞에서 설명한 경쟁 스위치 소자를 기본 스위치로 하여 경합집속기를 구성한다. 그림 13에 8×4 경합집속기의 구성 예를 나타내었다. 그림에서 하나의 선이 입력으로 들어와서 하나의 출력을 내는 기본 스위치는 지연 스위치로 단순히 입력을 1비트 지연시키는 역할을 한다. 그리고, 그림에서 \bullet 로 표시된 부분은 원래의 경합집속기에서 지연 스위치가 있었던 부분이나, 제안된 경합집속기는 비동기적 경합집속을 하므로 지연 스위치를 제거하였다.

패킷 필터에서 들어온 입력은 경합집속기의 첫 번째 단으로 입력된 후, 다른 패킷이 동시에 입력되었을 경우 경쟁 스위치에서 그림 12에서와 같이 경쟁을 하여 경쟁에서 이기면 왼쪽으로, 지면 오른쪽으로 가게 되며, N×L 경합집속기일 경우 경쟁에서 L번 지면 그 패킷은 폐기가 된다. 위의 그림에서는 8×4 경합집속기이므로 4번 경쟁에서 지게 되면 폐기가 된다. 경쟁에서 한번도 지지 않은 패킷은 첫 번째 버퍼로 가게 되고, 한번 지고 나머지를 모두 이긴 패킷은

4. 제안된 구조의 성능 분석

이 장에서는 3장에서 제안한 구조에 대한 성능을 분석한다. 입력 인터페이스로 입력된 패킷은 입력 인터페이스 내부에서는 지연을 가지지 않으며, 버스로 보내진 후 경합집속기를 거쳐 버퍼에 저장된 후, 출력 인터페이스를 거쳐 목적 프로세서로 보내지게 된다. 이러한 구조에서 성능의 기준으로 삼을 수 있는 것은 경합집속기에서의 패킷 손실률, 공유 버퍼에서의 오버플로우 확률, 경합집속기에서의 지연, 공유 버퍼에서의 지연, 그리고 총 지연이 있으며, 이 논문에서는 기존의 턴아웃 스위치에서 경합집속기의 구조를 바꾼 것이므로, 경합집속기에서의 패킷 손실과 지연, 그리고, 총 지연을 중심으로 성능을 분석한다.

4.1 경합집속기에서의 패킷 손실률

스위치에 도착하는 각각의 패킷이 N개의 출력에 도착할 확률이 동일하다고 가정하고, 패킷의 길이는 독립적으로 지수 분포를 따른다고 가정하면, 이 논문에서 제안한 경합집속기를 표준 "L-server loss system"으로 해석할 수 있다[13]. 특별히, N이 클 경우에 어떤 특정한 출력으로 도착하는 패킷을 포아송 프로세스로 볼 수 있다. 이러한 조건에서 도착한 패킷이 L개의 출력을 가진 경합집속기 내에서 손실될 확률은 식 2에 나타낸 Erlang B Formula로 구한다[13].

$$Pr[Packetloss] = \frac{\frac{\rho^L}{L!}}{\sum_{k=0}^L \frac{\rho^k}{k!}} \tag{2}$$

식 2에서 ρ 는 시스템의 부하이며, k는 경합집속기에 동시에 입력된 패킷의 수이고, L은 경합집속기의 출력의 개수이다. 식 2를 사용하여, 부하가 0.5일 때부터 1.0일 때까지 L의 크기 변화에 따른 패킷 손실률을 그림 14에 나타내었다. L이 8일 경우를 보면 패킷 손실률은 부하가 1.0에 이르더라도 10^{-5} 이하의 값을 가진다. 경합집속기에서의 패킷 손실을 버퍼 오버플로우에 의한 패킷 손실 및 전송 매체 자체의 손실보다 작은 값을 유지하기 위해 L을 8에서 10정도의 크기로 하면 충분할 것이다.

4.2 경합집속기에서의 지연

$N \times L$ 경합집속기를 사용하였을 경우 이 논문에서 제안한 경합집속기에서의 지연은 동시에 입력되는 패킷의 수에 따라 지연의 값이 다르므로 각 경로로 지나갈 확률을 구해서 지연의 평균을 구할 수 있다. 버스의 부하가 ρ 일 때, k개의 패킷이 동시에 하나의 출력으로 향할 확률은 식 3과 같이 나타낼 수 있다.

$$Pk = \binom{N}{k} \left[\frac{\rho}{N}\right]^k \left[1 - \frac{\rho}{N}\right]^{N-k} \quad k=0, 1, \dots, N \tag{3}$$

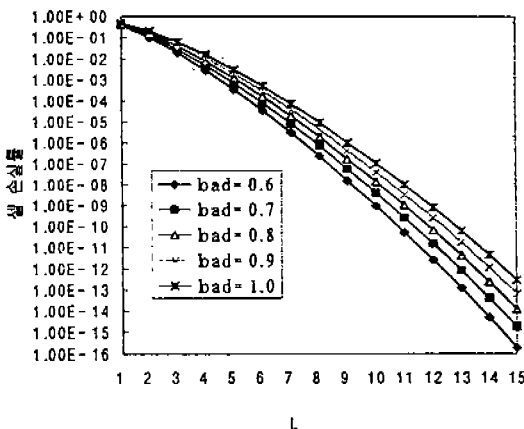
따라서, 버스로 입력되는 패킷들이 각각의 버퍼로 가게 될 확률은 식 4와 같다.

$$P[Loc(i)] = \frac{\sum_{k=1}^N \frac{Pk}{k}}{\sum_{k=0}^N Pk} \quad i=1, 2, 3, \dots, L \tag{4}$$

최종적으로 경합집속기를 통과한 패킷들이 각 버퍼 i로 들어갈 확률은 식 5와 같다.

$$P[Buf(i)] = \frac{P[Loc(i)]}{\sum P[Loc(i)]} \quad i=1, 2, \dots, L \tag{5}$$

이 논문에서 제안한 경합집속기에서의 지연은 기존의 것과 달리 부하에 따라 값이 변하게 된다. 경합집속기의 출력을 1, 2, ..., L로 번호를 부여할 때, 1번 출력으로 가는 패킷은 「 $\log_2 N$ 」 비트의 지연을 가지

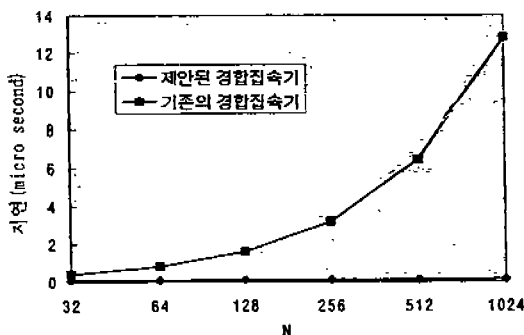


(그림 14) 부하와 L값에 따른 패킷 손실 확률
(Fig. 14) Packet Loss Probability According to Load and L

고, 2번 출력으로 가는 패킷은 $\lceil \log_2 N \rceil + 2$ 비트의 지연을 가지게 된다. L-1번 출력까지는 2비트씩 지연이 증가하며, L번 출력은 L-1번 출력보다 1비트 지연을 더 가지게 된다. 이러한 지연에 각 출력으로 패킷이 가게될 확률을 곱하면 평균 지연이 되며, 이는 식 6과 같이 표현된다.

$$\lceil \log_2 N \rceil \times P\{Buf(1)\} + \sum_{j=1}^{L-2} [(\lceil \log_2 N \rceil + 2 \cdot j) \times P\{Buf(j+1)\}] + (\lceil \log_2 N \rceil + 2 \cdot (L-2) + 1) \times P\{Buf(L)\} \quad (6)$$

이 식에서 $N \geq L, L \geq 2$ 이다. 식 6을 계산해 보면 경합집속기에서의 지연은 $\lceil \log_2 N \rceil$ 비트보다 같거나 크며 $\lceil \log_2 N \rceil + 1$ 비트보다 작은 값을 가진다. 이 값은 원래의 경합집속기에서의 지연인 N 비트에 비해 크게 감소한 것으로 제안된 경합집속기가 지연 면에서 우수함을 보여준다. 그림 15는 기존 경합집속기와 개선된 경합집속기의 지연을 넥아웃 스위치의 입력수에 따른 비로서 보여 주고 있다.



(그림 15) 경합집속기에서의 지연
(Fig. 15) The Delay in Concentrator

4.3 총 지연

시스템 전체의 지연은 패킷 전송 시간, 경합집속기에서의 지연, 공유 버퍼에서의 지연의 합으로 표현된다. 패킷 전송 시간은 버스의 속도와 패킷의 길이에 의해 구할 수 있고, 경합집속기에서의 지연은 2절에서 구한 값을 사용한다. 마지막으로 공유 버퍼에서의 지연은 큐잉 모델을 사용하여 구할 수 있다. 넥아웃 스위치에서 공유 버퍼는 M/M/1/K 큐잉 모델로 해

석이 가능하고, 이 모델에서 어떤 패킷이 도착한 순간의 공유 버퍼의 총 크기는 $\frac{\rho}{1-\rho}$ 패킷으로 표현이 된다[13]. 따라서, 공유 버퍼에서의 지연은 $\frac{\rho}{1-\rho}$ 패킷의 전송 시간이다.

버스의 속도, 패킷의 평균 길이, 프로세서의 수(즉, 버스의 수)를 각각 R_0 Mbps, L 바이트, N이라고 하면 제안된 구조에서의 총 지연은 식 7과 같이 표현할 수 있다. 이 식에서는 경합집속기에서의 지연은 최대 값으로 하였으며, 패킷 전송 시간에서 버스에서의 전파 시간은 값이 작으므로 무시하였다.

Total Delay

$$\cong \left\{ \left[\frac{8 \cdot L}{R_0} \right] + \left[(\lceil \log_2 N \rceil + 1) \cdot \frac{1}{R_0} \right] + \left[\frac{\rho}{1-\rho} \cdot \frac{8 \cdot L}{R_0} \right] \right\} \mu \text{ sec} \quad (7)$$

기존의 경합집속기를 사용하였을 경우의 총 지연은 식 8과 같이 나타낼 수 있다. 기존의 구조에서는 입력 인터페이스에서 입력 패킷의 동기를 위해 지연이 필요하며, 그 값은 평균 $\frac{13.5}{\rho} \cdot \frac{1}{R_0}$ 로 나타낼 수 있다. 그리고, 경합집속기에서의 지연은 N 비트 시간이다.

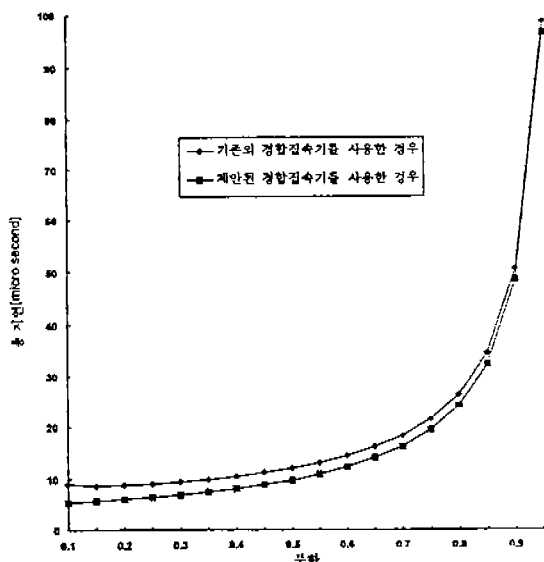
Total Delay'

$$\cong \left\{ \left[\frac{13.5}{\rho} \cdot \frac{1}{R_0} \right] + \left[\frac{8 \cdot L}{R_0} \right] + \left[\frac{N}{R_0} \right] + \left[\frac{\rho}{1-\rho} \cdot \frac{8 \cdot L}{R_0} \right] \right\} \mu \text{ sec} \quad (8)$$

식 7과 8에 따라 $L=48, R_0=80, N=160$ 일 때, 부하의 변화에 따른 지연의 변화를 나타낸 것이 그림 16이다. 식 7에 파라미터로 N이 포함되어 있으나, 새로운 경합집속기를 사용한 경우에 N은 지연에 거의 영향을 미치지 않는다.

따라서, 기존 단일 버스 시스템에서 노드 수의 증가 때문에 발생하던 스위치 오버 시간에 의한 지연의 증가 문제는 제안된 구조에서는 발생하지 않는다. 또한, 제안된 경합집속기를 사용하였을 경우에 기존의

경합집속기를 사용하였을 경우에 비해 경합집속기에서의 지연이 감소하므로, 제안된 경합집속기가 지연 면에서 우수함을 말해준다.



(그림 16) 스위치의 총 지연
(Fig. 16) Total Delay in Switch

5. 결론 및 향후 연구 과제

이 논문에서는 이동통신시스템에서 프로세서 간 통신을 위해 제공되어야 하는 통신망의 구조를 제안하였다. 이 새로운 프로세서 간 통신망은 기존의 단일 버스 형태의 구조가 가지는 한계를 극복하기 위해 다수의 버스를 가지는 구조로 설계되었다. 이 구조는 패킷 스위치로 사용되고 있는 녀아웃 스위치를 이용하여 설계되었으며, 가변 길이의 패킷을 입력과 동시에 지연없이 처리하기 위해 새로운 경합집속기를 설계하였다.

새로 제안된 경합집속기는 기존의 녀아웃 스위치의 경합집속기에서 사용되던 지연 스위치를 제거하여 패킷을 처리할 수 있게 하였으며, 경합집속과 데이터 전달을 동시에 할 수 있게 하였다. 지연 스위치를 제거함으로써 경합집속기 내부에서 발생되던 지연을 기존 경합집속기의 N 비트에서 $\lceil \log_2 N \rceil + 1$ 비트 정도의 크기로 감소시켰다. 또한, 제안된 경합집속기는 비동기적으로 경합집속 기능을 수행하므로, 입

력 인터페이스에서의 지연도 제거할 수 있다.

제안된 경합집속기를 사용한 녀아웃 스위치를 이동통신시스템에서의 프로세서 간 통신을 위한 망구조로 사용한다면, 기존의 구조와 같은 부분적인 메쉬 토폴로지를 그대로 적용할 수 있으며, B-ISDN과 연계된 망에서의 대용량의 트래픽을 처리할 수 있을 것으로 기대된다.

기존의 경합집속기와 제안된 경합집속기는 모두 특정 위치의 패킷에 우선 순위가 주어지도록 되어 있으므로, 우선 순위를 공평하게 할 수 있는 방법이 연구되어야 한다.

참고 문헌

- [1] 김상훈 외, TDX-10에서의 고속 대용량 IPC 기능 구현, 한국정보과학회 봄 학술발표논문집 vol. 20, no. 1, 1993.
- [2] 이충근, 대용량 전자 교환기의 프로세서간 고속 통신 네트워크에 관한 연구, 고려대학교 박사 학위 청구 논문, 1994.
- [3] 이혁재, CDMA 이동통신 시스템의 연구개발 현황, 한국통신학회지, 제10권 제10호, 10월, 1993년.
- [4] 전광일 외, 이중화로 연결된 다중프로세서시스템에서의 신뢰성있는 프로세서간 통신기법, 한국정보과학회 봄 학술발표논문집, vol. 20, no. 1, 1993.
- [5] 최진규 외, 대용량교환시스템의 IPC Network구조 설계 및 성능평가에 관한 연구, 한국전자통신연구소 1993년 연구 중간 보고서, 1993.
- [6] 한국전자통신연구소, 디지털 이동통신시스템 개발, 1993.
- [7] 한국전자통신연구소, 디지털 이동통신시스템 개발, 1994.
- [8] Boxma. O. J and Groenendijk. W. P, Pseudo-conservation laws in cycle-service systems, Journ of Appl. Prob., 24, 1987.
- [9] Boxma. O. J and Groenendijk. W. P, Waiting times in discrete-time cycle-service system, IEEE Trans. on Comm., vol. 36, no. 2, 1988.
- [10] Boxma. O. J and Meister. B. W, Waiting time approximation for cycle-service systems with switch-over times, Performance evaluation 6, 1987.

- [11] Eng. K. Y, Hluchyj. M. G and Yeh. Y. S, A knockout switch for variable-length packets, Proc. ICC 87, June 9, 1987, pp. 22. 6.1-22. 6.6.
- [12] Eng. K. Y, Hluchyj. M. G and Yeh. Y. S, Multicast and broadcast services in a knockout packet switch, Proc. INFOCOM 88, New Orleans, LA, Mar. 1988, pp. 29-34.
- [13] Kleinrock. L, Queuing systems, vol. 1:Theory, John Wiley & Sons, New York, NY, 1975, pp. 94-106.
- [14] Kuehn. P. J, Multiqueue systems with nonexhaustive cycle service, The Bell Sys. Tech. Journ, Mar. 1979.
- [15] Levy. H, Kleinrock. L, Polling systems with zero switch-over periods:A general method for analyzing the expected delay, Performance Evaluation 13, 1991.
- [16] Prycker. M, Asynchronous transfer mode:Solution for broadband ISDN, Ellis horwood limited, 1993.
- [17] Yeh. Y. S, Hluchyj. M. G and Acampora. A. S, The knockout switch:A simple modular architecture for high-performance packet switching, Proc. ISS 87, Mar. 1987.



박 상 규

1983년 경북대학교 전자공학과 졸업(학사)
 1985년~1986년 삼성전자 전산실 근무
 1989년 경북대학교 대학원 전자공학과(공학석사)
 1991년 경북대학교 대학원 전자공학과 박사과정 수료

1992년~현재 연암공업전문대학 컴퓨터정보기술과 조교수

관심분야: B-ISDN/ATM 교환분야 및 프로토콜



김 재 홍

1994년 경북대학교 컴퓨터공학과 졸업(학사)
 1996년 경북대학교 대학원 컴퓨터공학과(공학석사)

1996년~현재 경북대학교 대학원 컴퓨터공학과 박사과정

관심분야: ATM교환분야, 컴퓨터 시뮬레이션



이 상 조

1974년 경북대학교 수학교육과 졸업(학사)

1976년 한국과학원 전산학과 졸업(이학석사)

1994년 서울대학교 컴퓨터공학과 졸업(공학박사)

1976년~현재 경북대학교 컴퓨터공학과 교수

관심분야: 운영체제, 자연언어처리, 기계번역