
면적 효율적인 독창적 ATM 스케줄러의 설계

손승일*

A Design of an Area-efficient and Novel ATM Scheduler

Seung-il Sonh*

이 논문은 2006년도 한신대학교 연구비 지원에 의해 수행되었음.

요 약

최근 입력 큐 방식의 ATM 스위치에 관한 연구는 가장 활발한 연구 분야 중의 하나이다. 입력 큐 방식의 스케줄러에 관한 연구에서도 많은 발전이 이루어져 왔으며, 상업적으로 응용되고 있다. 스케줄링 알고리즘은 쓰루풋을 향상시키고, QoS를 만족하면서, 공정하게 서비스를 제공하는 특성을 가져야 한다.

본 논문에서는 입력 큐 방식의 ATM 스위치 패브릭을 효과적이고, 빠르게 중재할 수 있는 스케줄링 알고리즘의 구현에 관해 연구하였다. 제안한 스케줄러는 랜덤 트래픽에서 100%에 수렴하는 스케줄링 성능을 제공하고 있다. 제안한 알고리즘은 4회의 반복 매칭을 통해서 N 포트 VOQ 스위치의 중재를 완료할 수 있다. 또한 제안한 알고리즘은 가장 널리 사용되는 iSLIP 알고리즘과 비교하였을 경우 1/2의 면적만을 사용하고 구현이 용이한 장점을 가지고 있다. 4회의 반복 매칭을 수행할 경우에는 iSLIP 알고리즘보다 더 우수한 성능을 보여주었다. 제안한 스케줄링 알고리즘은 FPGA로 구현되었으며, 보드 레벨에서 검증되었다.

ABSTRACT

Currently the research on input-queued ATM switches is one of the most active research fields. Many achievements have been made in the research on scheduling algorithms for input-queued ATM switches and also applied in commerce. The scheduling algorithms have the characteristics of improving throughput, satisfying QoS requirements and providing service fairly.

In this paper, we studied on an implementation of scheduler which arbitrates the input-queued ATM switches efficiently and swiftly. The proposed scheduler approximately provides 100% throughput for scheduling. The proposed algorithm completes the arbitration for N-port VOQ switch with 4-iterative matching. Also the proposed algorithm has a merit for implementing the scheduling algorithm with 1/2 area compared to that of iSLIP scheduling algorithm which is widely used. The performance of the proposed scheduling algorithm is superior to that of iSLIP in 4-iterative matching. The proposed scheduling algorithm was implemented in FPGA and verified on board-level.

키워드

ATM Switch, Scheduler, VOQ, iSLIP, Throughput

I. 서론

ATM(Asynchronous Transfer Mode)은 데이터, 비디오, 음성 등과 같은 다양한 트래픽을 전송하는 차세대 멀티미디어 모바일 통신 시스템인 IMT-2000에서 응용될 수 있을 것으로 예견되는 분야이다[1]. ATM 스위치 중에서도 특히 입력 버퍼 방식을 채용한 스위치 구조가 널리 채용되고 있는 실정이다. 이러한 입력버퍼 스위치 구조는 각 입력 포트를 구현하는 방식에 따라 단일 버퍼 방식과 다중 버퍼 방식으로 구분된다. 그러나 단일 버퍼 방식의 스위치는 HOL 블로킹(Head-Of-Line Blocking) 문제로 최대 성능이 58.6%로 한정되기 때문에 고성능 스위치로는 적합하지 않은 것으로 알려져 있다[2]. 단일 입력 큐의 헤드 셀이 전송하고자 하는 목적지 출력 포트가 다른 포트의 셀을 출력하기 위해 이미 할당되어 있을 때 해당 출력 셀은 출력되지 못하여 전송되지 못하고 차단되거나 같은 큐에 저장되어 차단된 셀 뒤에서 대기하는 현상을 HOL 현상이라 한다. 입력 버퍼 스위치는 이러한 HOL 블로킹 문제를 해결하기 위해 하나의 입력 포트 당 2개 이상의 FIFO 큐를 지원하는 방향으로 발전하였다. 그리고 현재는 VOQ(Virtual Output Queuing) 방식으로 명명되는 입력 버퍼 스위치 구조를 대부분의 스위치 설계에서 채용되고 있다. 이 방식은 각 입력 포트는 각 출력 포트에 대해 분리된 개별적인 FIFO 큐를 제공해 줌으로서 HOL 블로킹 현상을 완전히 제거할 수 있도록 하였다. 그러나 많은 입력 버퍼가 각각 현재 전송 시간 슬롯에서 다중으로 하나의 출력 포트로 셀을 전송하고자 하는 경우에는 하나의 출력 포트는 하나의 입력 포트로부터 셀을 전송받을 수 밖에 없으므로 모든 입력 버퍼들에게 공정한 전송 기회를 할당하기 위한 셀 스케줄링 알고리즘이 필요하다. 셀 스케줄러는 신속하게 출력 포트 경쟁을 해결해주고 높은 쓰루풋(Throughput)을 제공해주며, QoS 요구를 만족시켜야 한다.

PIM, 2DRR, iSLIP, MUCS 알고리즘을 비롯하여 많은 셀 스케줄링 알고리즘이 개발되어 왔다[3-7]. 이러한 알고리즘들 중에서 iSLIP 스케줄링 방식이 구현과 성능 측면에서 우수한 것으로 알려져 있다[8]. 그리고 주요 ATM 스위치 장비에서는 iSLIP과 같은 방식의 스케줄러가 채용되어 상용화가 이루어진 실정이다. 현재까지 제시된 셀 스케줄링 알고리즘은 높은 쓰루풋을 제공하는 경우에는 면적이 커지는 단점이 있고, 혹은 아날로그와 디지털 설

계 기법을 병행하여 구현이 난해한 문제점을 가지거나, 단순하지만 성능이 떨어지는 단점을 가지고 있다. 그리고 iSLIP과 같이 하드웨어 구현이 용이한 기존의 스케줄링 알고리즘에 대한 개선책에 대한 연구 또한 미흡한 실정이다[9-10].

따라서 본 논문에서는 iSLIP 알고리즘의 성능을 증가하면서 적은 면적으로 구현이 가능하여 VLSI 설계에 적합한 새로운 방식의 셀 스케줄러를 제안하고자 한다. 이를 위해 제안하고자 하는 스케줄링 알고리즘과 기존을 스케줄링 알고리즘에 대한 성능을 평가하기 위해 VOQ 스위치에 스케줄링 알고리즘을 적용하였다.

II. 관련 연구

그림 1은 오늘날 대부분의 고성능 ATM 스위치 장비에서 채용되어 사용되고 있는 VOQ 스위치 구조를 보여주고 있다.

4x4 VOQ 구조의 ATM 스위치로 각 입력 큐는 목적지 포트별로 분리된 FIFO를 제공해 주고 있다. 본 논문에서는 VOQ 구조의 ATM 스위치에서 효과적인 스케줄링을 수행하는 알고리즘을 개발하는 것이다. 그런데, VOQ 구조의 ATM 스위치로 발전하기 이전에 사용되었던 입력 버퍼 구조 중에서 각 입력 큐가 2개의 분리된 FIFO를 갖는 구조에 주목하게 되었다. 그림 2는 2개의 분리된 FIFO를 가지는 입력 큐를 사용하여 구현한 ATM 스위치의 구조를 보여주고 있다[11].

그림의 예는 하나의 입력 포트는 2개의 FIFO를 보유하고 있으며, 이중에서 상위 FIFO는 목적지 출력 포트가 홀수 포트에 향하는 ATM 셀을 관리하며, 하위 FIFO는 목적지 출력 포트가 짝수 포트에 연결되는 ATM 셀을 관리하게 된다. 그림 2에서 점선이 아닌 굵은 실선으로 표시된 선은 스케줄러의 중재를 통해 목적지 출력 포트에 셀을 전달하게 되는 개념도를 보여주고 있다. 그림 2는 각 입력 큐가 짝수 목적지 출력 포트와 홀수 목적지 출력 포트별 FIFO를 관리하는 구조이지만, 상위 FIFO는 상위 1/2 목적지 출력 포트에 향하는 셀을 관리하고, 하위 FIFO는 하위 1/2 목적지 출력 포트에 향하는 셀을 관리하는 방식으로 구현하는 것도 가능하다.

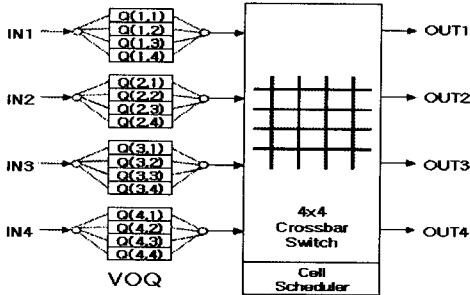


그림 1. 4x4 VOQ 구조의 ATM 스위치
Fig. 1 ATM switch with 4x4 VOQ structure

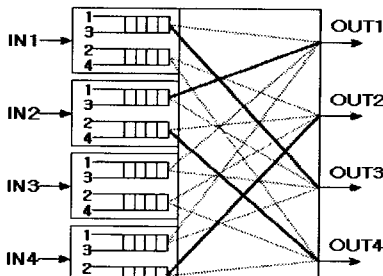


그림 2. 각 입력큐가 2개의 분리된 FIFO로 구현된
4x4 ATM 스위치 구조
Fig. 2 4x4 ATM switch structure with 2 separated
FIFOs for each input queue

입력큐를 짝수 및 홀수 목적지 포트별로 셀을 분리하는 방식의 개념을 응용하여 새로운 방식의 스케줄링 알고리즘 개발에 활용하였다.

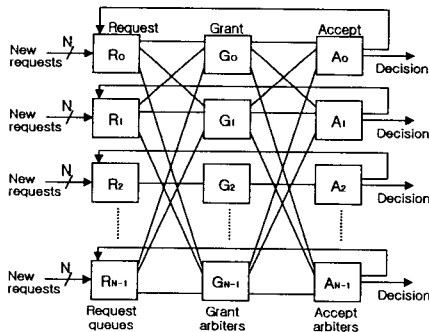


그림 3. iSLIP 스케줄링 알고리즘의 블록도
Fig. 3 Block diagram of iSLIP scheduling algorithm

그림 3은 본 논문에서 비교하고자하는 iSLIP 스케줄링 알고리즘의 블록도를 보여주고 있다. iSLIP 스케줄링 알고리즘은 3단계로 이루어진 반복 매칭을 통해서 입출력

을 중재하게 되는데, 이러한 3단계는 다음과 같이 요약할 수 있다.

- 1) 요청(Request) : 모든 입력단은 그 입력단에 저장되어 있는 셀들의 모든 목적지 출력단으로 요청(request) 신호를 보낸다.
- 2) Grant : 만약 출력단이 1개 이상의 요청을 수신하게 되면, 가장 높은 우선권을 가지는 입력단부터 시작하여 정해진 라운드-로빈 스케줄 방식으로 가장 가깝게 있는 입력단에 그랜트(grant) 신호를 보낸다.
- 3) Accept : 입력단이 1개 이상의 그랜트 신호를 수신하게 되면, 가장 높은 우선권을 가지는 입력단부터 시작하여 정해진 라운드-로빈 스케줄에 의해 먼저 활성화되어 있는 입력단을 수락한다.

위에서 설명한 1)부터 3)까지의 과정을 반복적으로 수행하게 되며, 반복 회수 많아짐에 따라 중재 성능도 향상된다. 보통의 경우에는 2 - 4회 정도의 반복을 수행한다. 랜덤 유니폼(Random Uniform) 셀 생성 환경에서 시뮬레이션 결과 iSLIP 알고리즘을 3회 반복 수행하였을 때의 매칭 성공률은 99.5%를 보였으며, 4회 반복 수행하였을 때의 매칭 성공률은 99.6%를 보였다. 그러나, iSLIP 스케줄링 알고리즘은 높은 중재율을 보이지만, 입력 포트 수 N 이 증가하게 되면 회로의 복잡도는 $O(N^2)$ 으로 급격하게 증가하는 문제점을 가지고 있다.

따라서 본 논문에서는 iSLIP 스케줄링 알고리즘의 높은 중재율을 보장하면서도 이를 회로로 구현할 때 복잡도를 대폭적으로 줄일 수 있는 새로운 구조의 알고리즘을 연구하게 되었다.

III. 새로운 스케줄링 알고리즘의 제안

본 논문에서는 iSLIP의 장점인 100%에 수렴하는 쓰루풋을 보장하면서, 회로 면적을 줄일 수 있는 새로운 구조의 스케줄러를 연구하였다. 그림 4는 본 논문에서 제안하는 새로운 구조의 스케줄러 블록도를 보여주고 있다.

그림 4는 스위치의 포트 크기가 $N=8$ 일 경우를 보여주고 있다. 그림 4를 기본으로 하여 본 논문에서 제안하는

스케줄러의 전체적인 구현 전략을 설명하면 다음과 같다.

한 셀 타임의 한 클럭 사이클당 모든 목적지 포트에 대한 중재를 수행하지 않고, 짝수 포트 및 홀수 포트에 대해 교대로 중재를 수행한다. 결과적으로 한 클럭 사이클당 전체 목적지 포트의 1/2에 대한 중재만을 수행하게 된다.

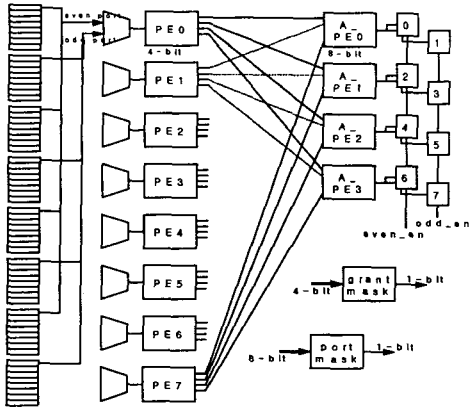


그림 4. 본 논문에서 제안하는 새로운 구조의 스케줄러 블록도(N=8)

Fig. 4 Block diagram of the proposed novel scheduler(N=8)

그러므로 스케줄러에서 사용되는 PE (processing element)의 작업량은 기존의 iSLIP 스케줄러와 비교하여 반으로 줄어들게 된다. Grant 포트를 선별화하는 PEi는 기존의 iSLIP과 동일한 수의 유닛을 필요로 하지만 입력출력 포트의 수를 1/2만을 사용하고 내부 회로 또한 기존의 절반만 사용하여 구현이 가능하다. A-PEi는 최종적으로 중재를 수락할 포트를 선택하는 유닛으로 입력과 출력 신호의 수는 기존의 iSLIP 방식과 동일하지만 A-PEi의 수를 1/2만을 사용하므로 전체적인 회로의 면적이 1/2로 감소하게 된다.

본 설계에서 추가적으로 필요한 부분은 PEi의 입력에 유입되는 입력 신호를 선택하기 위해 사용된 멀티플렉서 블록이다. 이 멀티플렉서 블록에서는 제어 신호에 따라서 짝수 및 홀수 포트를 교대로 선택하는 기능만을 수행하는 단순 멀티플렉서로 구현된다. Even_en 신호가 활성화되어 있을 시에는 0, 2, 4, 6 번 목적지 포트에 대한 중재를 수행하게 되고, odd_en 신호가 활성화되어 있을 시에는 1, 3, 5, 7 번 목적지 포트에 대한 중재를 수행하게 된다.

아울러서 특정 목적지 포트로의 중재가 승인되었다는 것을 알리는 포트(port) 마스크 블록과 특정 VOQ가 최종

적으로 중재된 목적지 포트를 승인받았다는 것을 알리는 그랜트(grant) 마스크 블록이 있다. 포트 마스크 블록 또한 본 설계에서는 기존의 iSLIP 방식의 1/2의 회로 면적으로 구현이 가능하다. 왜냐하면, 각 PEi의 출력이 기존의 1/2이기 때문에 포트 마스크의 수도 또한 1/2만을 필요로 한다. 단지, 짝수 및 홀수의 중재시에 각기 다른 포트에 대한 마스크를 의미하기 때문에 추가적으로 1개의 마스크 레지스터를 보유하고 있어야 한다. 그랜트 마스크 블록은 포트의 크기와 동일한 수의 블록을 필요로 하지만, 입력이 기존의 iSLIP과 비교하여 1/2만을 갖는다. 결과적으로 1/2의 면적으로 그랜트 마스크 블록의 구현이 가능하게 된다.

전체적으로 본 논문에서 설계중인 스케줄러는 기존의 iSLIP과 비교하여 1/2의 회로 면적으로 구현이 가능하며, 성능 또한 100%의 쓰루풋을 얻을 수 있는 장점이 있다. 단, 이 때, 한 가지 추가적으로 지원해야 하는 사항이 있는데, 이는 각 셀 타임의 첫 번째 시작하는 중재는 짝수 및 홀수 포트가 변경되도록 해야 한다는 것이다. 즉, 현재의 셀 타임에서 홀수 포트의 중재를 먼저 시작한다면, 다음 번 셀 타임의 시작에서는 짝수 포트에 대한 중재를 먼저 수행해야 한다는 것이다. 이렇게 해야 각 포트는 공평성을 보장받을 수 있다.

그림 5는 본 논문에서 설계한 스케줄러의 상세한 블록도를 보여주고 있다. 그림 5에서 볼 수 있는 바와 같이 2개의 포인터가 필요하다. 하나는 gnt_pe 블록의 포인터이며,

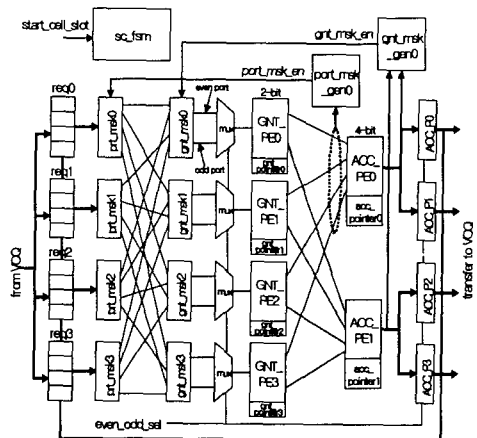


그림 5. 본 논문에서 제안하는 스케줄러의 상세 블록도(N=4)

Fig. 5 Detail block diagram of the proposed scheduler(N=4)

다른 하나는 acc_pe 블록의 포인터이다. 이러한 포인터들은 현재 수행중인 중재에서 가장 우선권이 높은 포트의 주소를 포함하고 있다. 그리고 Sc_fsm 블록에 있는데, 이는 스케줄러의 전체 동작을 제어하는 상태 머신이다. 본 설계의 스케줄러는 10 사이클 이내에 스케줄을 완료하도록 하였으며, 2개의 클럭 사이클에 1번의 중재를 수행하며, 이러한 중재는 4회 반복하여 수행하는 구조이다. 그리고 마지막 9번째 클럭 사이클에서 스케줄러의 전체를 갱신하는 기능을 수행한다.

IV. 성능 평가 및 분석

본 논문에서는 제안하는 스케줄러의 성능을 기존의 iSLIP 스케줄링 알고리즘과 비교하기 위해 두 스케줄링 알고리즘을 C 언어를 사용하여 모델링하였으며, 이에 대해 컴퓨터 시뮬레이션을 수행하였다. 그리고 각 스케줄러의 성능 검증을 위한 스위치는 입력 버퍼 구조가 VOQ 구조를 갖는 8x8 크로스바 스위치 구조를 채택하였다. 시뮬레이션에 사용된 VOQ의 각 목적지별 크기는 512 셀 버퍼를 사용하였다.

먼저 랜덤 유니폼(Random Uniform) 트래픽에 대해서 분석하기로 하자. 그림 6은 본 논문에서 제안하는 스케줄러와 iSLIP 스케줄링 알고리즘을 쓰루풋을 비교 분석한 것이다. 그리고 iSLIP 스케줄링 알고리즘의 반복 매칭 회수를 3회와 4회에 대해 성능 평가를 수행하였고, 제안된 알고리즘 또한 반복 매칭 회수를 3회 및 4회에 대해 성능을 평가를 수행한 것이다. iSLIP 알고리즘은 3회 혹은 4회의 반복 매칭 수행시 거의 유사한 쓰루풋을 제공하고 있지만, 본 논문에서 제안하고 있는 스케줄러는 3회 반복 수행시에는 80% 이하의 중재 성능을 나타내었다. 따라서 3회 반복 매칭을 수행하는 것은 낮은 성능을 제공하기 때문에 비교의 대상에서 제외하였다. 그런데 본 논문에서 제안한 스케줄링 알고리즘을 4회 반복 수행할 경우에는 기존의 iSLIP 알고리즘과 동등한 성능을 발휘하는 것을 알 수 있다. 따라서, 본 논문에서 제안하는 알고리즘은 반드시 4회의 반복 수행을 하여야 한다는 것을 알 수 있다. 결과적으로 본 논문에서 제안하는 스케줄링 알고리즘은 iSLIP 보다 최소한 1회의 추가적인 반복 수행을 제공해야 한다는 것을 알 수 있다.

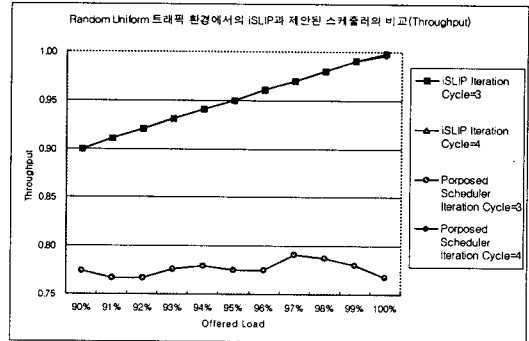


그림 6. 랜덤 유니폼 트래픽 환경에서 스케줄러간 쓰루풋 비교

Fig. 6 Comparison of throughput among schedulers under random uniform traffic

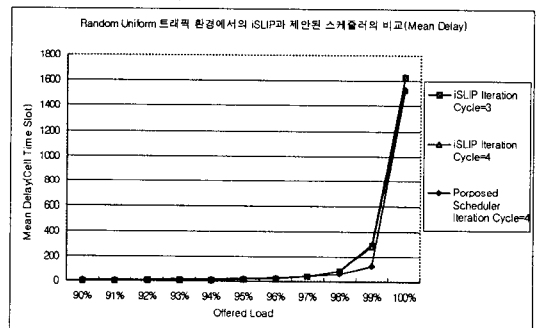


그림 7. 랜덤 유니폼 트래픽 환경에서 스케줄러간 평균 셀 지연 비교

Fig. 7 Comparison of mean cell delay among schedulers under random uniform traffic

그림 7은 기존의 iSLIP과 본 논문에서 제안하는 스케줄러 알고리즘간의 평균 셀 지연을 비교한 것이다. 97% 이하의 offered load에서는 거의 동일한 평균 셀 지연을 가지지만, 98%에서는 본 논문에서는 제안한 스케줄링 알고리즘이 작은 평균 셀 지연을 가지므로 결과적으로 평균 셀 지연에 더 우수한 특성을 나타내고 있음을 알 수 있다.

그림 8은 제안한 스케줄링 알고리즘과 iSLIP 알고리즘의 최대 버퍼 크기를 분석한 것이다.

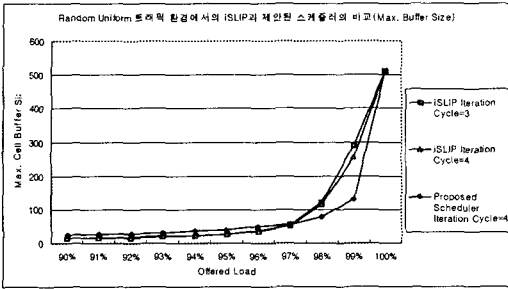


그림 8. 랜덤 유니폼 트래픽 환경에서 각 스케줄러간 최대 버퍼 크기 비교

Fig. 8 Comparison of maximum buffer sizes among schedulers under random uniform traffic

97% 이하에서는 본 논문에서 제안한 스케줄링 알고리즘이 약간 큰 버퍼의 크기를 가지지만, 98% 이상의 offered load에서는 본 논문에서 제안하는 알고리즘이 더 작은 버퍼 크기를 갖는다. 따라서 99%의 offered load가 제공될 경우 본 논문에서 제안하는 알고리즘은 130개의 버퍼 크기를 요구하지만, 4회 반복 수행을 하는 iSLIP 알고리즘은 256개의 버퍼 크기를 요구하므로 iSLIP 알고리즘은 거의 2배의 버퍼 크기를 소비하고 있음을 알 수 있다. 따라서 98% 이상의 offered load가 제공될 경우에는 본 논문에서는 제안하는 알고리즘이 더 작은 버퍼 크기를 사용하므로 더 작은 비용으로 구현이 가능함을 알 수 있다.

다음은 버스티 유니폼(Bursty Uniform) 트래픽 환경에 대해 각 스케줄링 알고리즘간의 성능을 분석해 보았다. 버스트 길이(Bursty length)는 8로 하여 성능을 분석하였다. 먼저 버스티 유니폼 트래픽 환경에서 각 스케줄링 알고리즘간의 쓰루풋을 비교한 것이 그림 9에 나타나 있다.

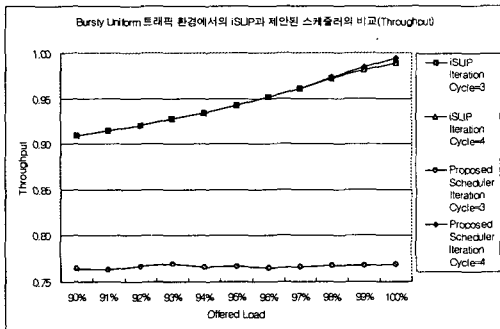


그림 9. 버스티 유니폼 트래픽 환경에서 각 스케줄러간 쓰루풋 비교

Fig. 9 Comparison of throughput among schedulers under bursty uniform traffic

그림에 나타나 있는 바와 같이, 본 논문에서 제안한 스케줄링 알고리즘 중에서 반복 수행이 3회일 경우에는 대략 76%의 쓰루풋을 나타내므로 구현에서 제외하였다. 따라서 본 논문에서 제안하는 4회 수행 스케줄링 알고리즘과 iSLIP 알고리즘 간의 쓰루풋에 대해 분석할 것이다. 그림에 나타나 있는 바와 같이 97% 이하의 offered load에서는 거의 동일한 성능을 나타내고 있음을 알 수 있으며, 98% 이상에서는 본 논문에서 제안한 4회 반복 수행 알고리즘이 약간 높은 쓰루풋을 나타냄을 알 수 있다.

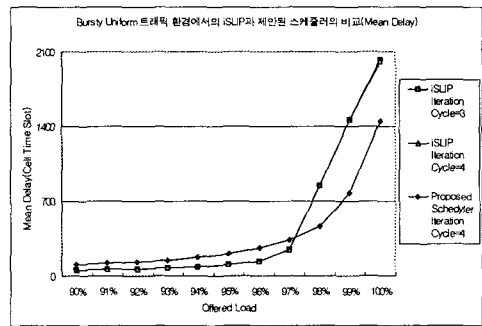


그림 10. 버스티 유니폼 트래픽 환경에서 각 스케줄러의 평균 셀 지연 비교

Fig. 10 Comparison of mean cell delay among schedulers under bursty uniform traffic

그림 10은 버스티 유니폼 트래픽 환경에서 각 스케줄링 알고리즘의 평균 셀 지연을 비교한 것이다. 평균 셀 지연은 97%의 offered load까지는 iSLIP 알고리즘이 약간 우수한 것으로 나타났으며, 98% 이상의 offered load에서는 본 논문에서 제안한 스케줄링 알고리즘이 상당히 우수한 것으로 나타났다.

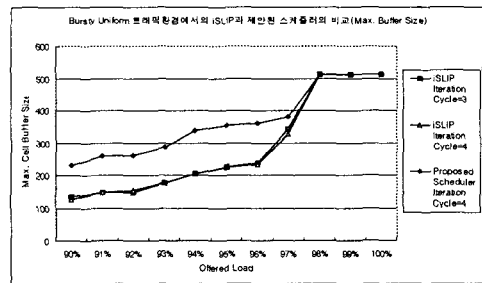


그림 11. 버스티 유니폼 트래픽 환경에서 각 스케줄러의 최대 셀 버퍼 크기 비교

Fig. 11 Comparison of maximum buffer sizes among schedulers under bursty uniform traffic

그림 11은 버스티 유니폼 트래픽 환경에서 최대 버퍼 크기를 비교한 것이다. iSLIP 4회 반복 수행과 본 논문에서 제안하는 스케줄링 알고리즘은 거의 동일한 최대 버퍼 크기를 가짐을 알 수 있으며, iSLIP 3회 반복 수행시에는 다소 많은 버퍼 크기를 소비함을 알 수 있다.

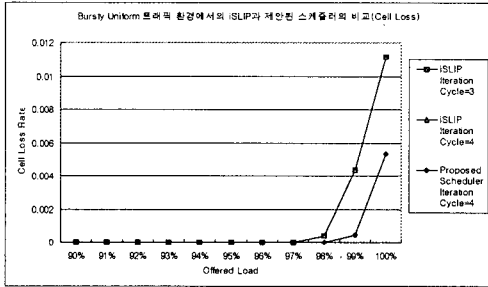


그림 12. 버스티 유니폼 트래픽 환경에서 각 스케줄러의 셀 손실률 비교
Fig. 12 Comparison of cell loss rates among schedulers under bursty uniform traffic

그리고 98% 이상의 offered load에서는 모든 스케줄링 알고리즘에서 버퍼 오버플로우가 발생하고 있음을 보여 주고 있다. 따라서 VOQ에서 셀 손실이 발생하고 있음을 의미한다. 이러한 셀 손실률을 그림 12에 나타내었다. 97% 이상의 offered load에서 iSLIP 스케줄링 알고리즘이 본 논문에서 제안한 스케줄링 알고리즘보다 높은 셀 손실률을 보이고 있음을 알 수 있다.

V. 시뮬레이션 및 FPGA 검증

그림 13은 본 논문에서 제안한 스케줄러의 동작에 대한 시뮬레이션 파형을 보여주고 있다.

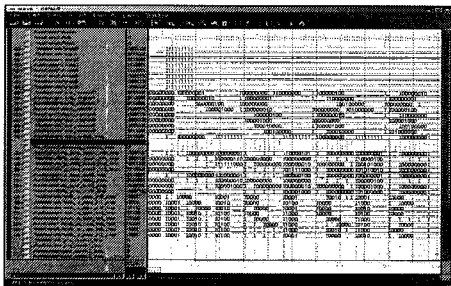


그림 13. 스케줄러 코어의 시뮬레이션 파형
Fig. 13 Simulation waveform for scheduler core

Vreq0 ~ vreq7은 각 VOQ의 셀 증재 요청 신호의 존재 여부를 의미하는 신호이며, acc_vector0 ~ acc_vector7은 스케줄러의 증재를 완료한 다음의 각 VOQ의 증재가 허가된 포트에 해당되는 포트만이 '1'로 세팅되어 나타나게 된다. 모든 포트는 특정 목적지 포트에 대해 단지 1개의 포트만을 스위칭을 허락하게 되므로, acc_vector는 모두 상호 배타적인 값을 가져야 한다. Even_odd_sel_tmp 신호는 본래 내부 신호이지만, 신호의 모니터링을 위해 외부로 출력한 것이다. 본 설계의 스케줄러의 4회의 증재를 수행하도록 하지만, 각 셀 타임 슬롯마다 작수 및 홀수 포트의 우선 증재를 교대로 지원하기 위해 4회의 증재가 끝난 다음 even_odd_sel_tmp 신호는 한 번 더 변경되게 된다. Pointer_update_tmp 신호는 내부의 포인터 갱신 enable 신호이다.

ATM 교환기용 고속 스케줄러의 테스트를 위해 자일링스 XCV1000E 디바이스에 설계한 모듈을 다운로드 하였고, Visual C++ 언어를 이용하여 작성한 테스트 프로그램을 PCI 인터페이스 카드를 통해 전송하고 수신함으로써 설계한 모듈을 검증하였다.

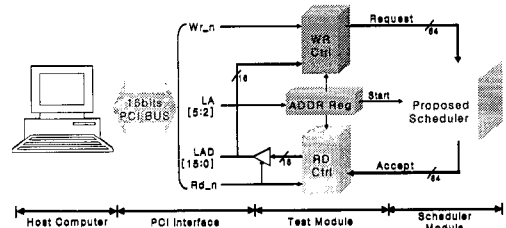


그림 14. 전체 데모 시스템의 인터페이스 블록도
Fig. 14 Interface block diagram for demo system

인터페이스 방식은 호스트 컴퓨터에서 PCI 인터페이스 카드를 통하여 읽기/쓰기 신호와 주소, 데이터 버스를 입력하여준다. 테스트 모듈에서 Wr_ctrl 블록에 입력되어 온 16비트 데이터 4개를 조합하여 64비트의 Request 신호를 생성하게 되고 스케줄러 시작 신호를 확인 후 스케줄러 모듈에 입력한 후 스케줄링 후 최종 Accept 신호 64비트를 Rd_ctrl 블록에 입력한 후 PCI 인터페이스를 통해 각 주소에 16비트씩 PC에 전송한다. 그림14는 설계한 모듈의 개략적인 인터페이스 모듈을 보여주고 있다. 하나의 전송되는 셀의 크기는 256바이트로 각 포트별로 전송 승인된 출력포트에 1개의 셀을 보내어 화면에 출력하게 된다. 그리고 입력되는 셀의 전송 비율을 두어 전송비율에

맞게 스위칭 되도록 구현하였다. 그림15는 데모 프로그램의 실행화면이다.



그림 15. 데모프로그램의 실행화면
Fig. 15 Executed display for demo program

VI. 결과 고찰

본 논문에서 제안한 알고리즘이 랜덤 유니폼 및 버스티 유니폼 트래픽 환경에서 기존의 iSLIP 스케줄링 알고리즘보다 전반적으로 우수한 성능을 보이고 있음을 알 수 있으며, 또한 회로의 면적 또한 1/2로 줄일 수 있는 장점이 있다. 표 1은 입출력 포트 크기가 N인 경우 몇 가지 스케줄링 알고리즘의 성능, 면적, 복잡도 및 특징을 요약한 것이다. 표에서 볼 수 있는 바와 같이, 2DRR 스케줄링 알고리즘의 경우에는 입출력 포트 수가 증가함에 따라 이에 비례하여 반복 매칭의 회수가 증가하는 것이 단점이며, MUCS 스케줄링 알고리즘의 경우에는 아날로그와 디지

표 1. 제안한 알고리즘과 타 알고리즘의 비교
Table 1. Comparison between the proposed algorithm and other algorithms

입출력 포트의 크기 = N				
구분	반복매칭 회수	복잡도	면적	특징
2DRR	N	O(N)	n/a	·포트크기 증가시 적용 불가
MUCS	N	O(N ²)	n/a	·아날로그와 디지털 혼용 구현이 난해, 포트크기 증가시 적용 불가
iSLIP	3,4회	O(N ²)	1	·성능 우수 ·포트 크기 증가시 복잡도 증가
Proposed algorithm	4회	O(N ² /2)	1/2	·성능이 가장 우수 ·iSLIP의 1/2 면적으로 구현

털 설계 방식을 병행하여 회로를 설계하였기 때문에 구현이 난해한 문제점을 가지고 있다. iSLIP 스케줄링 알고리즘 경우에는 반복 회수는 4회 이내로 적절하지만 복잡도가 O(N²)으로 입출력 포트 수가 증가함에 따라 회로 구현이 난해해지는 문제점을 가지고 있음을 알 수 있다. 그리고 본 논문에서 제안한 알고리즘은 반복 회수가 4회이며, 면적은 iSLIP 스케줄링 알고리즘의 1/2밖에 되지 않아 회로로 구현할 경우에 장점이 있으며, 성능 또한 우수하였다.

VII. 결론

본 논문에서는 VOQ 방식의 ATM 스위치를 효과적으로 중재할 수 있는 스케줄러에 대해 연구하였다. 본 논문에서 제안한 알고리즘이 랜덤 유니폼 및 버스티 유니폼 트래픽 환경에서 기존의 iSLIP 스케줄링 알고리즘보다 전반적으로 우수한 성능을 보이고 있음을 알 수 있으며, 또한 회로의 면적 또한 1/2로 줄일 수 있는 장점이 있는 것으로 확인하였다. 성능 평가를 바탕으로 제안한 알고리즘을 FPGA로 구현하였으며, 이를 보드 레벨에서 검증을 완료하였다. 따라서 본 논문에서 제안한 스케줄링 알고리즘은 고속의 ATM 셀 스케줄링 알고리즘으로서 사용이 가능할 것으로 사료된다.

참고문헌

- [1] Masahide Hatanaka, Toshihiro Masaki, Takao Onoye, "VLSI Architecture of Switching Control for AAL Type 2 Switch," IEICE Trans. Fundamentals, Vol.E83-A, No.3, pp435-441, Mar. 2000
- [2] A. Mekittikul, and N. McKeown, "A Practical Scheduling Algorithm to Achieve 100% Throughput in Input-Queued Switches," In Proc. IEEE Inforcom'98. Vol.2, pp.792-799, San Francisco, April. 1998.
- [3] Richard O. LaMaire and Dimitrios N. Serpanos, "Two-dimensional round-robin schedulers for packet switches with multiple input queues," IEEEACM Trans. Networking, vol. 2, no. 5, pp.471-482, Oct. 1994.
- [4] N. McKeown, P. Varaiya, and J. Walrand, "Scheduling cells in an input queued switch," Electron. Lett., vol.29,

no.25, pp.2174-2175, 1993.

[5] Y. Tamir, and G. Frazier, "High performance multi-queue buffers for VLSI communication switches," Proc. of 15th Ann. Symp. on Comp. Arch., pp.343-354, June 1988.

[6] T. Anderson, S. Owicki, J. Saxe, and C Thacker, "High speed switch scheduling for local area networks," ACM Trans. Comput. Syst., vol. 11, no. 4, pp.319-322, Nov. 1993

[7] H. Duan, J. W. Lockwood, and S. M. Kang, "Matrix Unit Cell Scheduler(MUCS) for Input-Buffered ATM Switches," IEEE Commun. Lett., vol. 2, no. 1, pp20-23, Jan. 1998

[8] S. Motoyama, "Cell delay modeling and comparison of iterative scheduling algorithms for ATM input-queued switches," IEE Proc. Commun. Vol.150, No 1, pp11-16, Feb. 2003

[9] I. Radusinovic, M. Pejanovic, Z. Petrovic, "An ILPF Cell Scheduling Algorithm for ATM Input-Queued Switch with Service Class Priority," TELSIC 2003, Vol.1,

pp26-29, Oct. 2003

[10] Zhao Qi, Gu Guanqun, "Design of an Input-queued ATM Switch supporting multicast and Research on its Scheduling Policy," IEEE Workshop on high performance switching and routing, pp331-334, May 2001

저자소개

손승일(Seung-Il Sonh)



1989년 연세대학교 전자공학과(학사)
 1991년 연세대학교 대학원 전자공학과(석사)
 1998년 연세대학교 대학원 전자공학과(박사)

2002년~현재 한신대학교 정보통신학과 부교수

※관심분야: ATM 통신 및 보안, ASIC 설계, 영상신호처리칩