푸싱 방식에 의한 윈도우 입력 버퍼 스위치의 성능 향상에 관한 연구

양 승 헌, 조 용 권, 곽 재 영, 이 문 기
연세 대학교 전기,컴퓨터 공학부
Tel: (02) 361 - 2867 / fax : 312 - 4584

Window input buffer switch performance progressing by pushing police

Sung Hen Yang Youg Kwon Cho, Che Young kyak, mun ki lee
School of Electrical and computer Engineering Yonsei University
E-mail : hana@spark.yonsei.ac.kr

Abstract

In this paper, we are proposed to pushing window input buffer A.T.M Switch that is not use memory read and write of general window police. Pushing window switch is superior to general window switch in performance but is large to general window switch in cross point number. Max throughput and Cell occupying probability results are verified by analysis an simulation. The evaluation of performance is max throughput and cell loss probability and mean queue length.

Ⅰ. 개 요

현재의 네트 워크는 높은 속도와 실시간 처리를 요구하는 B-ISDN이 각광을 받고 있다. B-ISDN 체재에서 근간으로 하는 것은 A.T.M을 이용한 정보의 전송이다. A.T.M 은 53 byte의 셀과 155.2 Mbps, 625 Mbps 속도를 기본으로 하며, 다른 Packet 전송과 달리 VPI, VCI를 이용하여 빠른 속도와 대역의 확장을 가져오고 있다. 본 논문에서는 A.T.M 프로토콜에서 사용하는 A.T.M 스위치에 대한 내용을 제시하고자 한다. 스위치는 버퍼의 위치에 따라 분류를 하면 입력 부에 버퍼를 두는 입력 버퍼 스위치, 출력부에 버퍼를 두는 출력 버퍼 스위치, 중앙에 버퍼를 두는 공유 버퍼 스위치가 있다.
버퍼를 사용하지 않는 스위치는 셀프 라우팅 스위치, 크로스바 스위치 등이다. 스위치에서 버퍼를 두는 이유는 크로스바 스위치 경우 가고자하는 목적지가 동일하여 내부 충돌이 일어 날 경우 버퍼에 저장하여 다음 번에 전송을 하여 충돌을 감소시킨다. 본 논문에서 제시하는 윈도우 입력 버퍼 스위치는 입력 버퍼 스위치 계통에 속하며, 입력 버퍼 스위치에서 HOL(Head of Line) blocking 이 발생 할 수 있는데 윈도우 방식을 사용하여 HOL blocking 효과를 감소시키고 처리율을 향상시킨다. 그러나, 기존의 윈도우 방식은 Memory Read/Write 방식을 제시하여 내부 속도의 증가가 요구되며, Hardware 제작 시 복잡 도가 요구된다. 그러나, 본 논문에서 제시한 Pushing에 의한 윈도우 방식은 기존의 윈도우 방식 특성을 사용하였으나, Memory Read/Write 방식 대신 라인(LINE)과 크로스 포인트를 확장하여 윈도우 방식을 대체 하였다. 본 주제의 Pushing은 밀어 낸다는 의미인데, 내부 처리 속도를 빨리 한다는 것이다. 각 장별 내용을 보면 1 장에서는 개요를 소개하였고, 2 장에서는 윈도우 방식과 푸싱윈도우 방식의 구조와 동작 원리를 설명한다. 3 장에는 성능 해석으로서 푸싱윈도우 와 윈도우 방식의 최대 처리율을 수학적으로 검증하였다. 4 장에서는 시뮬레이션 결과를 제시하였는데 최대 처리율,셀 손실 율, Mean Queue Length를 제시하였다.

Ⅱ. 윈도우 방식과 푸싱 윈도우 방식

스위치

윈도우 방식은 입력 버퍼 A.T.M 스위치에서 입력 버퍼부를 윈도우 형태의 테이블로서 사용하는 것이다[5].

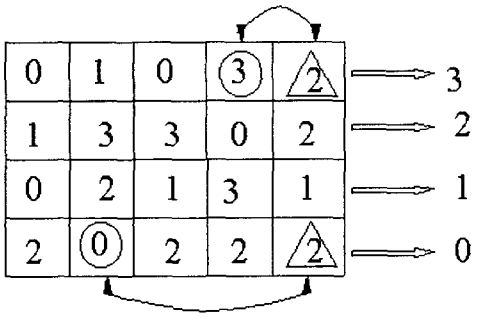

그림 1. 윈도우 입력 버퍼 스위치

그림 1.은 윈도우 방식 스위치를 나타내고 있는데, 내용은 입력 버퍼의 첫(first)단에서 유효 셀의 목적지가 2, 2, 1, 2 인 경우 포트 1, 2 의 목적지 2, 1 셀은 출력부로 보낼 셀로 정하고, 포트 0, 3 의 목적지 2, 2 셀은 충돌 셀이므로 윈도우를 확장하여 포트 1의 목적지 2 셀은 목적지 3 셀과 교체하고, 포스 3의 목적지 2 셀은 목적지 0의 셀과 교체를 한다. 결과적으로 윈도우 방식은 위의 결과와 같이 출력되는 셀은 목적지 0, 1, 2, 3 으로 출력부에서 충돌을 막을 수 있고, 셀 처리율 향상을 이룰 수 있다. 대신 문제점은 위의 포트 0, 3 과 같이 first 단 셀을 뒤로 보내어야하고, 뒤 단의 셀을 앞으로 보내어야 하므로 Memory Read/Write 방식이 요구된다.

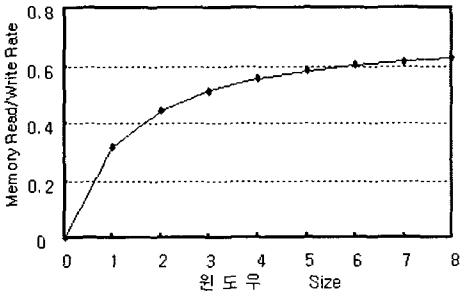

그림 2. 윈도우 사이즈 별 memory read/write rate.

그림 2.는 윈도우 방식을 사용하였을 때 Memory Read/write 의 Rate를 보여 주고 있는데, 윈도우 사이즈 별로 Rate가 지수적으로 증가하고 있는 것을 보여 주고 있다. 그림2. 윈도우 사이즈 8에서 메모리 Read/write Rate는 약 0.63 정도로 매우 높음을 보여 주고 있다. 본 논문에서 제안한 푸싱 윈도우 스위치는 윈도우 방식에서 문제점으로 제시한 Memory Read/Write 방식을 사용하지 않았으며, 대신 그림 3. 과 같이 크로스 포인트 증가와 Pushing(밀어내기)에 의해 윈도우 효과를 이루었다.

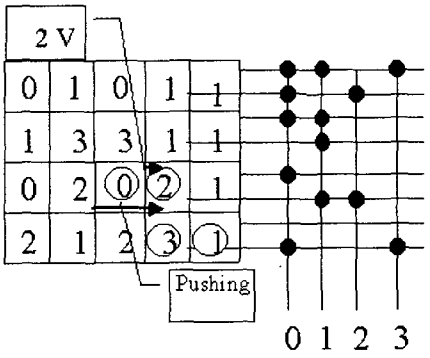

그림3. 푸싱 윈도우 2 스위치

그림3 은 푸싱 윈도우 2 입력 버퍼 스위치를 보여 주고 있는데 첫(first)단에서 유효 셀의 목적지가 1, 1, 1, 1 이므로 포트 4 의 목적지 1을 출력부로 보낼 셀로 정하고, 윈도우를 확장하여 다음 단(second) 의 목적지 2, 3을 출력부로 보낼 셀로 정하고, 2 단의 포트 3에서 셀을 보내고 푸싱에 의해 목적지 0을 내어 보낸다. 이 방식에서는 윈도우 확장에 사용하는 Memory Read/write 대신 Pushing에 의해 윈도우 확장효과를 이루었다. 문제점은 푸싱으로 인한 내부 속도 증가가 요구되며, 크로스 포인트 수의 증가와 Line(선) 의 증가가 요구된다.

Ⅲ. 성능 해석

윈도우 입력 버퍼 스위치에서 최대 처리율은 M/D/1에서 Mean Queue Length와 포화 상태에서 입력 버퍼 스위치의 특성을 이용하여 구할 수 있다[1][2][3][4]. 결과 식은 다음과 같다.

$$W - \rho = \frac{\rho^2}{2(1-\rho)} \quad (\text{W: 윈도우 사이즈}) \qquad (1)$$

윈도우 3에서 최대 처리율은 (1) 식에서 W = 3을 대입하여 정리하면 0.8377을 얻을 수 있다.

본 논문에서 제안한 푸싱 윈도우 스위치의 처리율은 윈

도우 확장이 근사적으로 W + ρ 이므로 다음 식으로 나타낼 수 있다

$$(W + \rho) - \rho = \frac{\rho^2}{2(1-\rho)} \quad (2)$$

$$W = \frac{\rho^2}{2(1-\rho)} \quad (W : \text{pushing 윈도우 사이즈})$$

푸싱윈도우 3 에서 최대 처리율은 식(2)에 의해 0.873 이다. 도표 1. 은 푸싱윈도우 스위치의 최대처리율을 시뮬레이션과 식(2)의 Analysis를 하여 결과를 보여 주고 있다. 시뮬레이션은 C를 사용하였으며 스위치 사이즈 128의 푸싱 윈도우 스위치를 사용하였다.

도표 1. 푸싱 윈도우 Max Throughput simulation and analysis 비교.

N=128, 푸싱 윈도우 최대 처리율		
윈도우 크기	simulation	analysis
1	0.7347	0.7347
2	0.8295	0.8284
4	0.8997	0.8989
7	0.9347	0.9373
10	0.9543	0.9544
20	0.9756	0.9761

셀 점유 확율(Cell Occupying Probability)은 직관에 의해 다음과 같이 놓을 수 있다.

$$L = \frac{\rho^2}{2(1-\rho)} \quad (\text{M/D/1 Mean queue Length})$$

$$\Pr(Q=n) = \frac{L^n e^{-L}}{n!} \quad (n = 0, 1, 2, 3, \ldots)$$

시뮬레이션과 Analysis 는 다음 도표와 같다. 도표 2. 시뮬레이션에서 사용한 스위치는 푸싱 윈도우 20 (Max Throughput 0.975) 이며 스위치 사이즈는 N = 256, Utilization 은 0.778 이다.

도표 2. N = 256, 푸싱 윈도 20, Cell Occupying Probability(utilization 0.778).

셀 점유 확율		
셀 수	simulation	analysis
0	0.33385	0.25582
1	0.31063	0.34875
2	0.22194	0.23772
3	0.10119	0.10802
4	0.02743	0.03682
5	0.00044	0.00228
6	0.00002	0.00044

Ⅳ . 시뮬레이션

본 실험에서는 일반 PC를 사용하였으며, C 언어를 사용하였다. 사용 트래픽은 Poision Process(Random Normal Bursty Length Cell Size, Random Exponential Interarrival time)를 사용하였다.
셀의 목적지는 Random Uniform을 유지하였으며, 처리 단위는 슬롯으로 처리하여 한 슬롯이 끝나면 다음 슬롯을 입력하였다. 성능 평가 요소는 최대 처리율(Max Throughput), 셀 손실 율(Cell Loss Probability), 평균 큐 길이 (Mean Queue Length)를 사용 하였다.

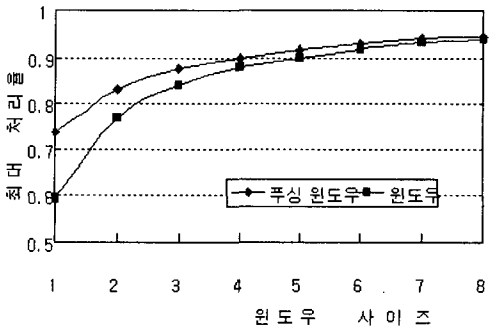

그림 4. 윈도우, 푸싱 윈도우 스위치 최대 처리율 비교.

그림 4. 는 푸싱윈도우 스위치와 윈도우 스위치의 최대 처리율을 보여 주고 있다. 최대 처리율은 입력 셀 유효 비율이 1.0 인 경우 출력되는 유효 셀 비율을 나타낸다. 그래프에서 최대 처리율은 푸싱윈도우 스위치가 윈도우

스위치보다 처리율이 높음을 보여 주고 있는데 윈도우 사이즈가 2 인 경우 윈도우 스위치는 0.78인 반면 푸싱 윈도우는 0.835 정도 나타내고 있다. 처리율은 윈도우 사이즈가 커짐에 따라 근접하고 있으나 조금은 푸싱 윈도우 스위치가 우수함을 나타내고 있다.

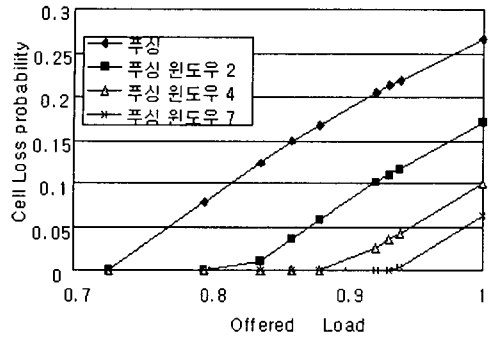

그림 5. N = 128, 푸싱 윈도우 셀 손실 율

그림 5. 는 푸싱 윈도우 스위치의 윈도우 사이즈별 셀 손실 율을 나타내고 있다. 사용 스위치는 N = 128 푸싱 윈도우 스위치이며, 셀 손실 율은 제공된 부하가 높아짐에 따라 증가됨을 알 수 있다. 그림에서 셀 손실 율은 윈도우 사이즈가 증가 될 수록 감소됨을 보여 주고 있는데, 그렇다고, Hardware 제작 시 윈도우 사이즈를 무한히 늘일 경우 버퍼 사이즈나 구조상 많은 제약이 따른다. 푸싱 윈도우 스위치의 셀 손실 율을 구체적으로 보면 푸싱 스위치는 제공 부하 0.74에서, 푸싱윈도우 2 스위치는 0.84에서 급격히 셀 손실 율이 증가됨을 보여 주고 있는데 그것은 제공 부하가 최대 처리율에 달했기 때문이다.

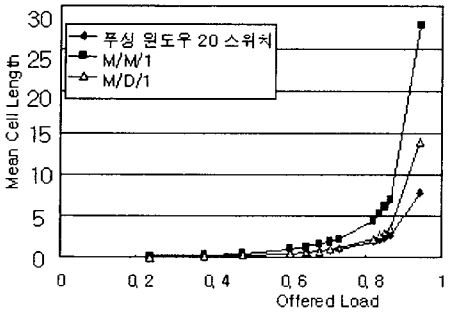

그림 6. N=128, 푸싱 윈도우 20, Mean queue Length

그림 6.은 스위치 사이즈 N = 128, 푸싱 윈도우 20 스위치의 Mean Queue Length를 보여 주고 있다. 그래프 특성은 M/M/1 과 M/D/1을 비교했을 때 M/D/1에 가까운 것을 알 수 있는데 그것은 스위치 사이즈가 클수록 A.T.M 스위치는 출력이 일정한 트래픽을 유지하기 때문이다. Mean Queue Length 는 최대 처리율 근처에서 기하 급수적으로 증가함을 보여주고 있다.

V. 결론

본 논문에서는 푸싱윈도우 스위치와 윈도우 스위치의 구조를 제시했으며, 두 스위치의 특성과 최대처리율을 비교하였다. 푸싱 윈도우 스위치의 최대 처리율과 셀 점유 확율은 시뮬레이션과 수학적 해석으로 검증하였다. 푸싱 윈도우 스위치의 윈도우 스위치에 대한 장점은 처리율이 우수하고, Memory Read/Write 사용하지 않는 것이나, 단점은 크로스 포인트 수가 윈도우 사이즈에 비례하여 늘어나고, 내부속도가 빨라야 한다. 그러나 Hardware 제작 시 푸싱 윈도우 스위치는 윈도우 방식의 Memory Read/write를 사용하지 않으므로 제작이 간편하다.

참고 문헌

[1]. Howard M.Taylor Samuel Karlin, " An Introduction To Stochastic Modeling 3 ",Academic Press 1998

[2] Donald Gross, Carl M.Harris, " FUNDAMENTALS OF Queueing Theory 3 ", WILEY INTER-SCIENCE, 1998

[3] SHELDON M.ROSS, " INTRODUCTION TO PROBABLITY MODELS 6 ", Academic Press 1997.

[4] MARK J.KAROL, MICHAEL G.HLUCHYJ, SAMUEL P.MORGAN, " Input Versus Output Queueing on a Space-Division Packet Switch ", IEEE Trans.Commun, Vol. COM-35, no, 12, pp. 1347 - 1356, December 1987.

[5] Dye-Jyunn MA, " The Maximum Throughput of a Nonblocking Packet Switch with Window Policy ", IELCE TRANS. COMMUN, VOL. E80-B. NO.4. APRIL.