

# Modified MOBAS에 대한 고장 감내기법 및 새로운 ATM 스위치 구조의 제안

권 세 동<sup>†</sup> · 박 현 민<sup>††</sup> · 최 병 석<sup>†††</sup> · 박 재 현<sup>††††</sup>

## 요 약

MOBAS는 규칙적인 모듈로 구성되어 있어 확장이 용이하며, VLSI 구현 시 고집적화 할 수 있고, 각 모듈간에 동기를 맞추기 쉬울 뿐 아니라, 단일 종류의 칩으로 중앙 스위치 구조를 구성할 수 있다. Modified MOBAS(Multicast Output Buffered ATM Switch)는 MOBAS와 유사한 구조를 가지지만 스위치 모듈(SM: Switch Module)의 구조에서 차이를 보이며 적은 스위치소자(SWE: Switch Element)를 사용한다. 위성 통신에서 스위치의 크기뿐 아니라 고장감내 특성도 스위치를 디자인하는데 필요한 중요한 요소이다. 본 논문에서는 Modified MOBAS의 고장 특성을 분석하고 이에 적합한 Detection 기법 및 Location 기법을 제안하였다. 또한 스위치 모듈구조를 변형하여 Modified MOBAS의 장점을 그대로 수행하면서 고장감내 특성을 가지는 새로운 스위치를 제안하였다. 제안된 스위치는 Modified MOBAS에 비해 약간의 스위치 소자를 더 사용하지만 MOBAS에 비해서는 적은 스위치 소자를 사용할 뿐 아니라, MOBAS와 같이 단일 고장 하에서 성능의 저하가 거의 없다.

## Fault-Tolerance of Wang's Modified MOBAS and A New Fault-Tolerant ATM Switch Architecture

Se-Dong Kwon<sup>†</sup> · Hyun-Min Park<sup>††</sup> · Byeong-Seog Choe<sup>†††</sup> · Jae-Hyun Park<sup>††††</sup>

## ABSTRACT

The architecture of the MOBAS (Multicast Output Buffered ATM Switch) has a regular and uniform structure and, thus, has the advantage of easy expansion, high integration density for VLSI implementation, relaxed synchronization for data and clock signals and building the center switch fabric with single type of chip. Modified MOBAS has a similar modular distributed structure to MOBAS, and has fewer switch elements than the MOBAS. The reduction of a switch size and the fault tolerance are important issues in designing an ATM switch for reliable satellite communications. In this paper, we analyze the fault-tolerance characteristics and propose the detection and location schemes of the Modified MOBAS. And, we propose a new ATM switch architecture with fault tolerant characteristics by slightly changing Modified MOBAS. The proposed switch has an easy fault reconfiguration scheme with graceful performance degradation like MOBAS and employs fewer switch elements than the MOBAS.

키워드: MOBAS, Modified MOBAS, 고장 검출(Fault-Detection), Fault-Location, 재구성(Reconfiguration)

## 1. 서 론

광대역 종합 정보 통신망(B-ISDN: Broadband Integrated Service Digital Network)은 매우 높은 전송률을 지원하며, 패킷 스위칭(packet switching) 방식을 기반으로 하고 있다. 광대역 종합 정보 통신망에서는 비동기 전송 모드(ATM: Asynchronous Transfer Mode) 방식[1]의 네트워

크를 공유하여 음성, 저속 또는 고속의 데이터, 이미지, 비디오 등 여러 가지 다양한 서비스를 제공하므로, 광범위한 트래픽 특성과 기능, 그리고 다양한 서비스 품질의 요구를 충족시킬 수 있다.

따라서, 비동기 전송 모드는 광대역 종합 정보 통신망의 효율적인 구축을 위해 전송 속도의 다양성 및 교환 기술의 구현을 위한 모드이다. ATM은 송신 측에서 수신 측으로 보내는 정보를 48bytes씩 나누고, 수신 측의 정보를 가진 5bytes의 헤더를 덧붙여, 총 53bytes의 고정 길이를 가진 셀(cell)이라고 하는 단위로 정보를 보낸다. B-ISDN에 대한 교환 기법과 멀티플렉싱(multiplexing) 기법으로 매우 적합하며, 높은 수율(throughput), 낮은 셀 손실률(cell loss rate), 짧은 지

\* 본 논문은 정보통신부의 대학기초연구지원사업에 의해 수행된 연구 결과의 일부임.

† 준 회원: 명지대학교 대학원 컴퓨터공학과

†† 종신회원: 명지대학교 컴퓨터학부 교수

††† 정 회원: 동국대학교 정보통신공학부 교수

†††† 정 회원: 명지대학교 전자정보통신공학부 교수

논문접수: 2000년 10월 18일, 심사완료: 2001년 3월 20일

연 등의 특징을 가진다[2-4]. 그리고, 다양한 서비스에 대해 각각의 QoS(Quality of Service)를 보장해주므로 서로 다른 대역폭을 요구하는 다양한 서비스들을 효율적으로 수용할 수 있다[3].

최근에는 유니캐스트(unicast)와 멀티캐스트(multicast)를 포함하는 ATM 스위치가 주요 연구 분야가 되어 왔다[5,6]. 초창기에는 높은 수율, 낮은 지연과 지연의 편차(deviation), 낮은 셀 손실률(cell loss), 그리고 ATM 스위치의 확장성을 얻기 위한 연구가 주요 목적이었다. 그러나, 현재에는 시스템 신뢰성(reliability)과 고장 감내(fault tolerance)가 중요한 관심사가 되고 있다[7-14]. 고장 감내를 위한 내부 상호 연결망(interconnection network)[15]이 제안되었으며, 특별히 위성 통신의 경우에 있어서 스위치 소자의 수를 줄이는 것과 고장 감내의 특성은 신뢰성을 보장하는 가장 중요한 요소이다.

MOBAS(Multicast Output Buffered ATM Switch)[16-19]는 분산된 연산 방식을 사용하고 규칙적인 모듈로 구성되어 있어 대용량 ATM 스위치로의 확장이 용이하며, 멀티캐스트 기능을 수행하기 위해 멀티캐스트 킥아웃 원리(multicast knockout principle)[20,21]을 적용하고, 짧은 지연과 높은 수율을 제공하기 위해 셀 복제 능력을 가지는 출력 큐잉(output queueing) 방식을 사용한다. 셀 복제(cell duplication), 셀 필터링(cell filtering), 셀 경쟁(cell contention)을 해결하기 위해 분산연산 방식을 사용하며, 멀티캐스트 어드레싱을 해소하기 위해 다단계(multi-stage) 방식을 사용한다.

또한, MOBAS는 입력단과 출력단 사이에 다중 경로(multiple path)를 제공한다. MOBAS의 이러한 다중 경로의 특성은 높은 신뢰성을 제공한다. 게다가, MOBAS의 규칙적 모듈 구성은 고장 검출(detection) 및 위치(location), 그리고 시스템 재구성을 위한 동적 진단(dynamic diagnosis) 능력을 제공하며, 심지어 스위치 소자의 고장이나 스위치 내부의 링크 고장의 경우에도 높은 수율과 낮은 셀 손실률을 제공한다.

최근에 MOBAS의 장점을 살리면서 스위치소자(Switch Element)의 수를 줄여 크기를 줄일 수 있는 스위치가 [22]에서 제안되었으며, 본 논문에서는 Modified MOBAS라 칭한다. 위성 탑재 ATM 스위치의 설계에 있어 위성의 무게와 부피를 줄이기 위해 스위치의 크기를 작게 하는 것이 중요하다. 이와 더불어 고장감내 특성을 가져야 한다. 이는 위성 환경에서 동작하므로 고장이 발생하여도 그 보수가 매우 어렵기 때문이다. 이에, 본 논문에서는 Modified MOBAS의 고장 특성을 분석하고, Modified MOBAS를 위한 새로운 재구성 방법을 제안한다. Modified MOBAS의 고장 감내 특성은 여러 가지 고장의 경우 하에서 성능 분석을 통하여 보여진다. Modified MOBAS는 MOBAS보다 적은 스위치 소자를 사용하면서 MOBAS에 비해 약간의 성능 저하를 나타낸다.

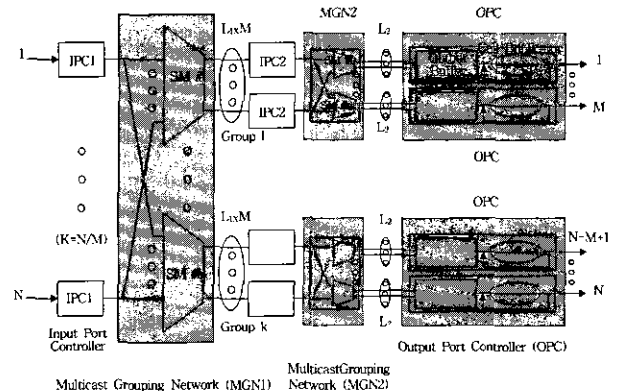
본 논문에서는 Modified MOBAS의 고장특성을 분석하고

이에 적합한 Detection 기법 및 Location 기법을 제안하였다. 또한, 스위치모듈 구조를 변형하여 Modified MOBAS의 장점을 그대로 수용하면서 고장감내 특성을 가지는 새로운 스위치를 제안하였고, 제안된 스위치에 사용된 스위치소자를 MOBAS 및 Modified MOBAS와 비교함으로써 위성탑재 스위치로서의 적합여부를 판별하였다.

제2장에서는 Modified MOBAS의 정상 상태에서의 셀 라우팅(cell routing)에 대하여 설명하고, 제3장에서는 고장의 종류와 여러 고장 경우에 따른 셀 라우팅에 대하여 설명한다. 제4장에서는 Modified MOBAS에 대해 제안한 고장감내 기법에 대하여 설명한다. 그리고, 제5장에서는 본 논문에서 제안한 새로운 ATM 스위치에 대하여 설명하고, 제6장에서는 제안한 스위치와 기존의 스위치를 스위치 크기와 셀 손실률 측면에서 비교 분석한다. 마지막으로, 제7장에서는 결론을 맺고 있다.

## 2. Modified MOBAS의 구조

### 2.1 스위치 구조



(그림 1)  $N \times N$  Modified MOBAS의 전체 구조

(그림 1)은 input port controllers(IPC1, IPC2), multicast grouping networks(MGN1, MGN2), 그리고 output port controllers(OPCs)를  $N \times N$  가지는 Modified MOBAS의 전체 구조를 나타낸다. MOBAS에서는 이러한 IPC와 OPC가 동일한 소자이다. IPC는 도착한 셀을 차단하여 번역 테이블(translation table)에 있는 정보를 살핀 후에, 필요한 정보를 셀에 추가하여 MGN에서 정확하게 라우트(route) 될 수 있도록 한다. OPC는 출력 포트에 향하는 셀들을 출력 버퍼에 저장하고, cell duplicator(CD)에서 멀티캐스트를 위해 다중 복제 셀(multiple copies)을 생성한다. 복제된 각각의 셀에 번역 테이블에 의해 얻어지는 새로운 VCI 값을 할당하고, 내부 셀 형태(format)를 표준 ATM 셀로 변환하며, 마지막으로 다음 스위칭 노드(switching node)나 목적지로 셀을 전송한다.

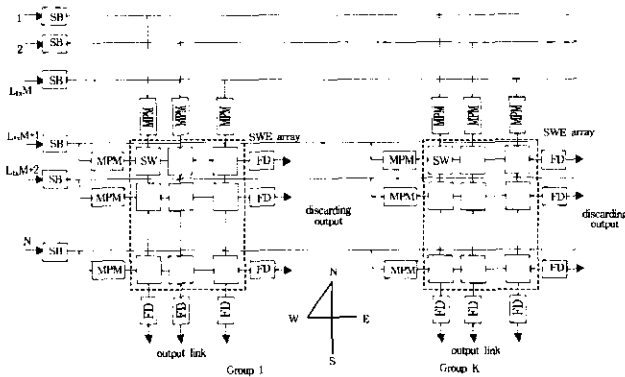
Modified MOBAS에서의 MGN은 MOBAS와는 조금 다

른 구조를 가지고 있다. (그림 2)는 MGNI(multicast grouping network)의 구조를 보여준다. 처음 단계(stage)에서 MGNI는  $K(K = N/M)$ 개의 스위치모듈과  $N$ 개의 SB(skew buffer)들로 구성되어 있다. 각 스위치 모듈은 2차원의 SWE array와  $N$ 개의 multicast pattern maskers(MPM)으로 구성된다. SWE array는  $(L_1 \times M) \times (N - L_1 \times M)$ 개의 SWE들로 구성되어 있다.  $L_1$ 은 MGNI에서의 확장비(expansion ratio)를 나타낸다.

스위치모듈은  $N$ 개의 브로드캐스트 버스(broadcast buses)로부터 최대  $N$ 개의 셀을 입력 받는다. 첫번째부터  $L_1 \times M$ 까지는 SWE array의 북쪽으로 입력되고, 나머지 입력들은 SWE array의 서쪽으로 입력된다. 셀의 유효여부와 우선순위에 따라  $L_1 \times M$ 개는 출력링크(남쪽)로 출력되어 MGNI2로 보내지며, 나머지 셀들은 폐기출력(동쪽)으로 출력되어 폐기된다.

MPM은 입력 셀의 유효여부를 판단하여, 유효 셀에 대해서는 유효비트(V)를 '0'로 설정하고 무효 셀에 대해서는 '1'로 설정하여 우선순위필드에 첨가한다. 1셀 시간(cell time) 동안 셀이 도착하지 않으면 빈 셀을 SWE array로 보내며, 고장의 Detection 및 Location을 위해 임의의 셀을 발생시킨다.

FD(fault detector)는 고장의 Detection 및 Location을 위해 셀의 라우팅 정보와 내용을 검사한다.



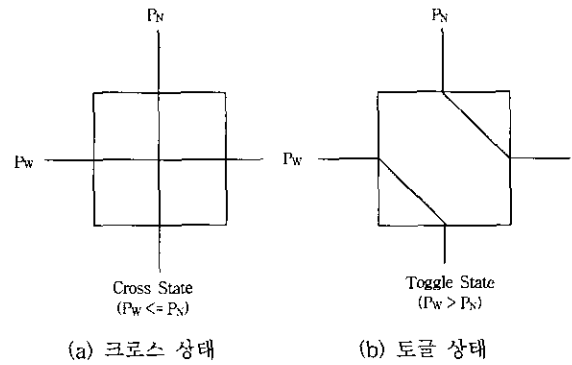
(그림 2)  $N \times N$  Modified MOBAS의 MGNI 구조

2.2 스위치모듈에서의 셀 라우팅

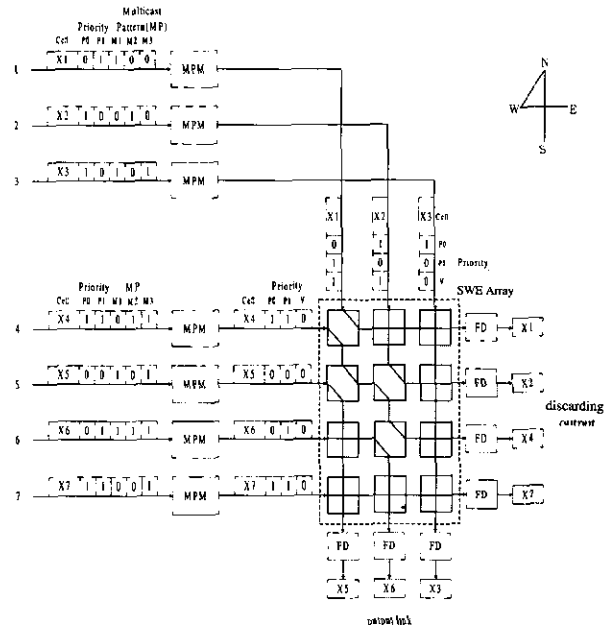
(그림 3)과 같이 SWE는 크로스 상태(cross state)와 토글 상태(toggle state)를 가진다. SWE는 각각 서쪽 입력과 북쪽 입력의 셀을 받아들인다. SWE의 상태는 북쪽입력 셀 ( $P_N$ )과 서쪽입력 셀 ( $P_W$ )의 우선순위비교에 의해 결정된다. SWE는 두 입력 셀(북쪽입력 셀과 서쪽입력 셀)의 우선순위필드( $VP_i P_0$ ) 값을 비트 순( $V \rightarrow P_1 \rightarrow P_0$ )으로 비교하여 자신의 상태를 결정하고, 해당 셀이 모두 통과할 때까지 그 상태를 유지하게 된다. 이는 SWE의 동작을 매우 간단히 하여 하드웨어적 구현을 쉽게 한다. 우선 순위가 높을수록 작은 값을 가지며, 우선 순위가 낮을수록 높은 값을 가진다. 따라서, 최고 우선순위 값은 '000( $VP_i P_0$ )'이고, 최저 우선순위 값은 '111( $VP_i P_0$ )'이다. SWE의 초기상태는 크로스

상태로 설정되어 있다. 북쪽입력 셀의 우선순위가 서쪽입력 셀의 우선순위보다 같거나 더 높으면 SWE는 크로스상태로 설정되고, 서쪽입력 셀의 우선순위가 북쪽입력 셀의 우선순위보다 높은 경우에만 SWE는 토글 상태로 설정된다.

각각의 SWE는 셀이 통과할 때 1비트 지연을 야기한다. 셀 타이밍을 조절하기 위하여, 번째 입력포트로 들어오는 셀은 SB에서  $i-1$ 비트 ( $i \leq L_1 \times M$ ) 혹은  $i - (L_1 \times M + 1)$  비트 ( $i > L_1 \times M$ ) 지연된 후 MPM으로 보내진다.



(그림 3) 스위치소자의 동작



(그림 4) 스위치모듈에서의 셀 라우팅

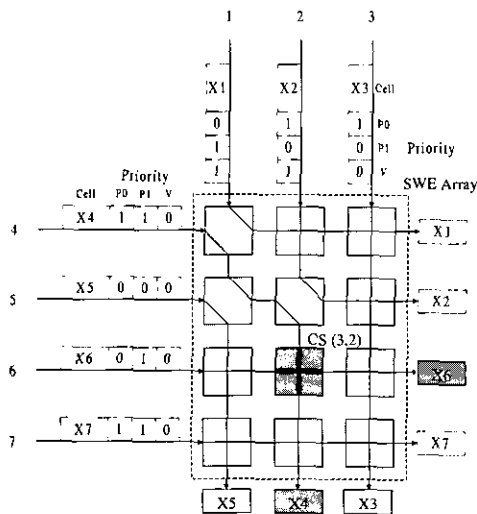
(그림 4)는 고장이 발생하지 않았을 때의 셀 라우팅을 보여준다. 스위치모듈은 7개의 입력( $N=7$ )과 3개의 출력링크 ( $L_1 \times M=3$ )를 가진다. 입력 셀들의 우선순위는  $X5 > X3 > X6 > X4 = X7 > X2 > X1$ 이다. 높은 우선순위를 가지는 3개의 셀들은( $X5, X3, X6$ ) 출력링크로 출력되어 MGNI2로 보내지고, 나머지 셀들은( $X4, X7, X2, X1$ ) 폐기출력으로 출력되어 폐기된다.

### 3. 고장의 종류

이 장에서는 SWE에서의 고장의 종류와 SWE array에서 여러 고장 경우에 따른 셀 라우팅에 대하여 설명한다. SWE의 고장은 SWE내부의 고장이나 링크의 결함에 의해 유발된다. SWE내부고장에 의한 고장은 SWE가 크로스 상태나 토글 상태로 고정되는 경우를 말하며, 전자를 CS(cross-stuck) 고장, 후자를 TS(toggle-stuck) 고장이라 한다. 링크결함에 의한 고장은 SWE의 수평링크나 수직링크가 단락되거나 개방되어 셀이 'all 1'(ST1 고장 : stuck at one) 혹은 'all 0'(ST0 고장 : stuck at zero) 값으로 출력되는 경우를 말하며, 수평링크의 결함을 HS(horizontal-stuck) 고장, 수직링크의 결함을 VS(vertical-stuck) 고장이라 한다.

설명을 위해 다음과 같이 표기를 정의한다. SWE(i,j)는 SWE array에서 i번째 행과 j번째 열에 위치한 SWE를 나타내며, CS(i,j), TS(i,j), HS(i,j), VS(i,j) 등은 각각 SWE(i,j)에서 발생한 CS 고장, TS 고장, HS 고장, VS 고장을 나타낸다.

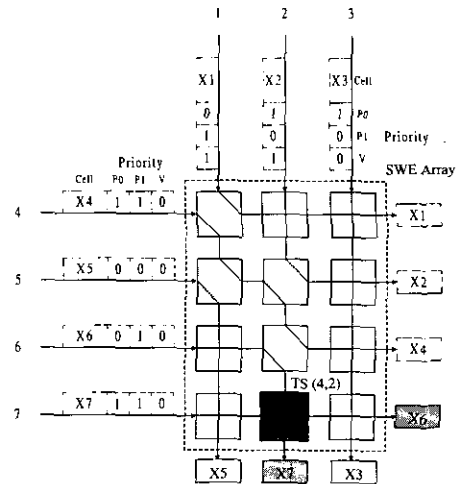
#### 3.1 Cross-Stuck 고장



(그림 5) CS(3,2) 고장에 의한 셀 손실

(그림 5)는 CS(3,2)가 일어난 SWE array에서의 셀 라우팅을 나타낸다. CS 고장은 SWE가 크로스상태로 고정되는 것을 말한다. 따라서, 북쪽 입력 셀 들은 항상 남쪽 출력으로 라우트되고, 서쪽 입력 셀들은 항상 동쪽 출력으로 라우트된다. (그림 5)에서 SWE array내의 빗금 친 부분은 고장난 SWE를 나타내며, 출력링크와 폐기출력의 빗금 친 부분은 고장으로 인해 잘못 라우팅 된 셀을 나타낸다. 위 표현은 모든 예에서 동일하게 적용된다. 그림에서 CS(3, 2)로 인해 3번째 우선순위를 가지는 X6가 폐기출력으로 출력되어 폐기되므로 문제가 야기된다.

#### 3.2 Toggle-Stuck 고장

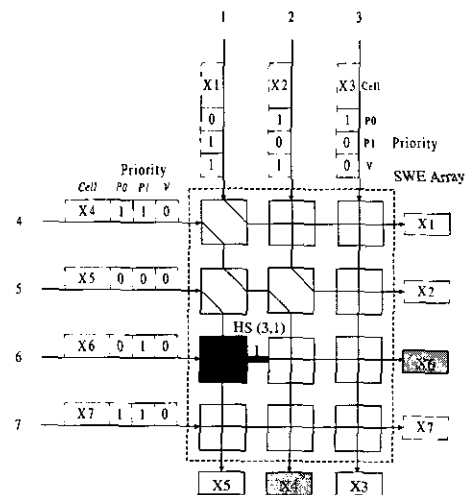


(그림 6) TS(4,2) 고장에 의한 셀 손실

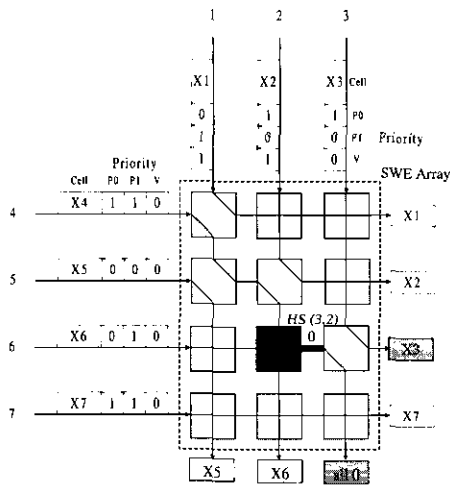
(그림 6)은 TS(4,2)가 일어난 SWE array에서의 셀 라우팅을 나타낸다. TS 고장은 SWE의 상태가 토글 상태로 고정되는 것을 말한다. 따라서, 서쪽 입력 셀 들은 항상 남쪽 출력으로 라우트되고, 북쪽 입력 셀들은 항상 동쪽 출력으로 라우트된다. 그림에서 TS(4,2)로 인해 3번째 우선순위를 가지는 X6가 폐기되므로 문제가 야기된다.

#### 3.3 Horizontal-Stuck 고장

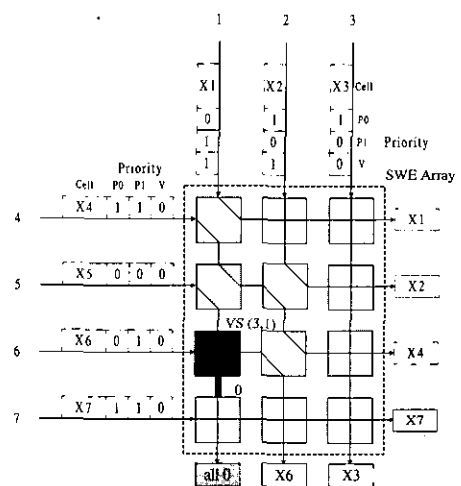
HS 고장은 SWE의 수평링크가 단락되거나 개방되어 모든 데이터 값이 'all 1'이나 'all 0' 값으로 출력되는 것을 말한다. (그림 7)에서는 ST1 HS(3,1)로 인해 3번째 우선순위를 가지는 X6가 폐기되므로 문제가 야기되고, (그림 8)에서는 ST0 HS(3,2)로 인해 2번째 우선순위를 가지는 X3가 폐기되고 'all 0'이 출력링크로 출력되므로 문제가 야기된다.



(그림 7) ST1 HS(3,1) 고장에 의한 셀 손실



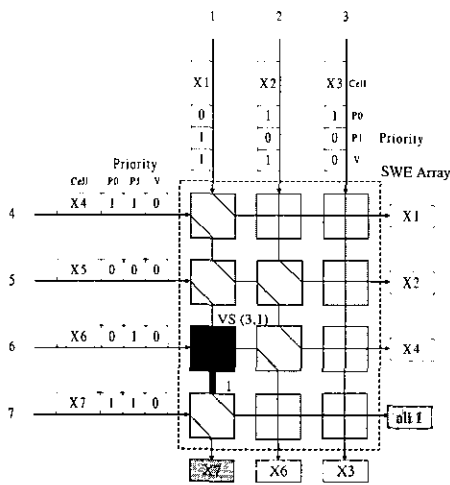
(그림 8) ST0 HS(3,2) 고장에 의한 셀 손실



(그림 10) ST0 VS(3,1) 고장에 의한 셀 손실

3.4 Vertical-Stuck 고장

VS 고장은 SWE의 수직링크가 단락되거나 개방되어 모든 데이터 값이 'all 1'이나 'all 0' 값으로 출력되는 것을 말한다. (그림 9)에서는 ST1 VS(3,1)로 인해 1번째 우선순위를 가지는 X5가 폐기되므로 문제가 야기되고, (그림 10)에서는 ST0 VS(3,1)로 인해 1번째 우선순위를 가지는 X5가 'all 0'으로 바뀌어 출력링크로 출력되므로 심각한 문제가 야기된다.



(그림 9) ST1 VS(3,1) 고장에 의한 셀 손실

4. Fault Detection 기법과 Fault Location 기법

4.1 Fault Detection 기법

고장의 Detection은 SWE array내에서 고장의 발생여부와 종류를 알아내는 과정이다. 고장 검출을 위하여, IPC2와 OPC 그리고 SWE array의 수직 출력은 간단한 논리 회로인 fault detector(FD)를 갖는다. 출력링크와 폐기출력에 위치한 FD는 셀들의 라우팅 정보와 내용을 검사하여 이상유무를 판단한다.

4.1.1 CS 고장과 TS 고장의 경우

CS 고장이나 TS 고장이 발생하면, 폐기 출력의 셀이 출력 링크의 셀 보다 우선 순위가 높은 경우가 발생한다. 즉, 우선순위가 높은 셀이 전송되지 못하기 때문에 심각한 문제가 야기된다. CS 고장과 TS 고장을 검출하기 위해 다음과 같은 방법을 제안하였다. 폐기 출력과 출력 링크에 있는 FD가 출력링크로 출력된 셀들 중 가장 낮은 우선순위를 가진 셀과 폐기출력으로 출력된 셀들 중 가장 높은 우선순위를 가진 셀을 비교하여, 후자의 우선순위가 더 높으면 고장으로 규정한다.

Detection 단계에서는 고장의 발생여부는 알 수 있으나, CS 고장인지 TS 고장인지는 구분할 수 없다. 이는 다음 장에 나오는 fault location 단계에서 알 수 있다.

4.1.2 HS 고장과 VS 고장의 경우

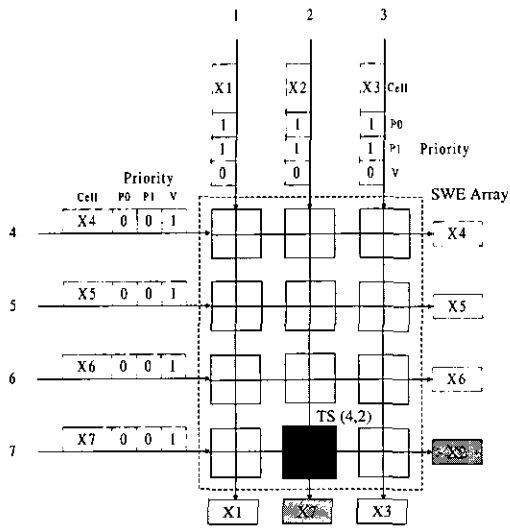
HS 고장이나 VS 고장이 발생하면 'all 1'이나 'all 0'가 출력된다. 따라서 FD에서 'all 1'이나 'all 0'가 검출되면 고장으로 간주한다. Detection 단계에서는 고장의 발생여부는 알 수 있으나, HS 고장인지 VS 고장인지는 구분할 수 없다. 이는 다음 장에 나오는 fault location 단계에서 알 수 있다.

4.2 Fault Location 기법

고장이 검출되면 고장 난 위치를 알아내기 위하여 좀 더 많은 정보가 필요하다. 고장 검출 과정에서 고장 난 SWE의 위치에 대한 부분적인 정보를 알 수 있다. 고장이 검출된 스위치를 정확하게 재구성하기 위하여 고장 난 SWE의 정확한 위치를 알아야 한다. CS 고장과 TS 고장의 경우, TS 고장에 대한 Location을 먼저 수행한다. TS location에 의해 고장이 발견되지 않으면 CS 고장에 대한 Location을 수행한다. 이는 TS Location이 CS Location보다 간단하기 때문이다. HS 고장과 VS 고장의 경우, 출력측에서 'all 1'이

나 'all 0' 값이 발견되면 각각 ST1 고장과 ST0 고장으로 규정하고 각각에 대한 오프라인 테스트를 수행한다.

4.2.1 TS 고장의 경우



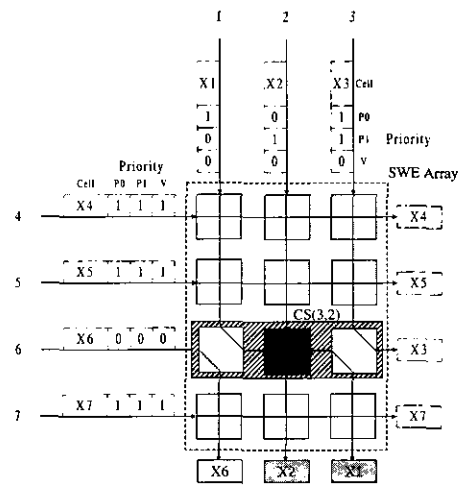
(그림 11) TS(4,2) 고장에 대한 Location

(그림 11)은 TS(4,2)에 대한 Location을 보여준다. 북쪽 입력 셀의 우선순위 값을 '011(VP<sub>1</sub>P<sub>0</sub>)'로 설정하고 서쪽입력 셀의 우선순위 값을 '100'으로 설정하여 입력하면, SWE array내의 모든 SWE들이 크로스상태로 동작하게 된다. 따라서 우선순위 값이 '011'인 셀이 출력링크로 출력되고, 우선순위 값이 '100'인 셀이 폐기출력으로 출력된다. 이와 같은 입력패턴을 'TP1(test pattern)'으로 규정한다. 만일 고장이 발생하면, 두 셀의 출력위치가 바뀌게 된다. 따라서 SWE(출력 셀이 바뀐 행, 출력 셀이 바뀐 열)에서 TS 고장이 발생했음을 알 수 있다. 그림에서 TS(4,2)로 인해 정상 상태에 비해 X2와 X7의 위치가 바뀌었다. 따라서 SWE(X2가 출력된 행=4, X7이 출력된 열=2)에서 TS 고장이 발생했음을 알 수 있다.

4.2.2 CS 고장의 경우

TS 고장에 대한 Location을 수행하여 아무런 이상이 발견되지 않으면 CS 고장으로 간주하며, CS 고장에 대한 Location을 수행한다. 우선 첫째 행에 대해 테스트 블록을 설정하고, 테스트 블록 내의 모든 SWE들은 토글 상태로 동작하고 블록외부의 SWE들은 크로스상태로 동작하도록 테스트 패턴을 입력한다. 블록의 서쪽입력으로부터 시계방향으로 우선순위를 갖도록 우선순위를 설정하며, 나머지 입력은 가장 낮은 우선순위를 부여한다. 정상상태에서는 우선순위 순으로 출력링크로 출력되며, 나머지 셀들은 폐기출력으로 출력된다. 만일 CS 고장이 존재하면 고장 난 열로 입력된 셀은 동일한 열로 출력된다. 따라서 SWE(현재 테스트 블록이 설정된 행, 입력과 동일한 열로 출력된 셀이 입

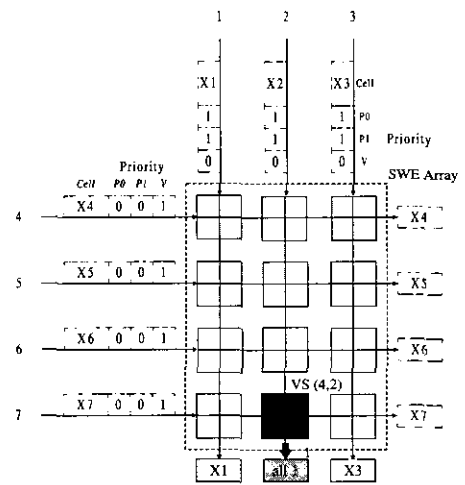
력된 열)에서 CS 고장이 발생했음을 알 수 있다. 고장 난 SWE가 발견될 때까지 행을 하나씩 증가시키면서 동일한 과정을 반복한다. (그림 12)는 CS(3,2)에 대한 Location을 보여준다. 3번째 행에서 고장이 발생한 경우이므로, 첫째 행과 둘째 행에 대한 테스트 블록 내에서는 아무런 이상이 발견되지 않는다. 따라서 셋째 행에 대해 테스트 블록을 설정한다. X6 > X1 > X2 > X3의 우선순위를 가지도록 우선순위를 설정하고 나머지 셀 들(X4, X5, X7)은 가장 낮은 우선순위를 가지도록 설정한다. 정상상태에서는 우선순위 순(X6, X1, X2)으로 출력되지만, CS(3,2)가 존재하면 2열로 입력된 셀은 2열로 출력된다. 따라서 SWE(3, 2)에서 CS 고장이 발생했음을 알 수 있다.



(그림 12) CS(3,2) 고장에 대한 Location

4.2.3 ST1 고장의 경우

ST1 고장은 3가지 유형으로 분류할 수 있다. 첫번째 유형은 VS 고장에 의해 'all 1'이 출력링크로 출력되는 형태로서 ST1 TYPE 1으로 정의한다. 두 번째 유형은 VS 고장에

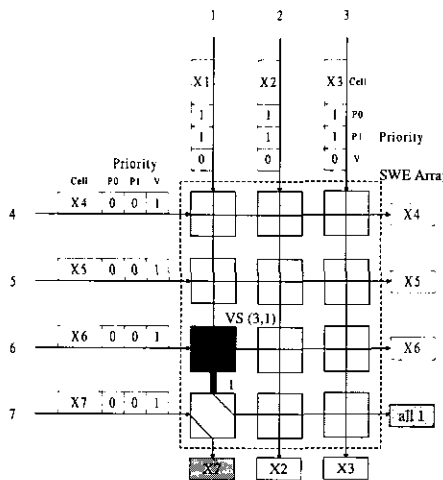


(그림 13) ST1 고장에 대한 Location (ST1 TYPE 1)

의해 'all 1'이 폐기출력으로 출력되는 형태로서 ST1 TYPE II로 정의한다. 세 번째 유형은 HS 고장에 의해 'all 1' 값이 폐기출력으로 출력되는 형태로서 ST1 TYPE III로 정의한다.

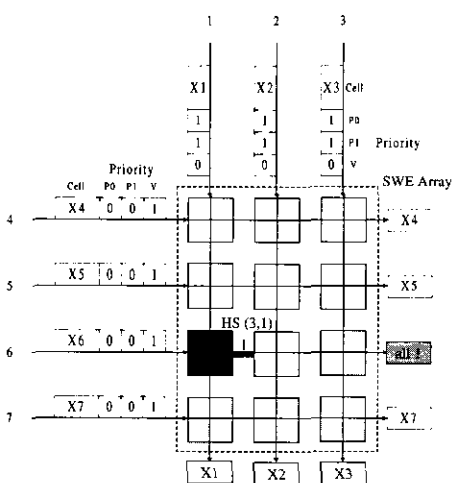
ST1 고장에 대해 TPI(test pattern 1)을 입력하면 3가지 형태로 출력된다. 첫 번째는 출력링크에서 'all 1'이 출력되는 형태이다(그림 13), ST1 TYPE I). 마지막 행에서 VS 고장이 존재하는 경우에만 나타나며, 고장 난 열에서 'all 1'이 출력된다. 따라서 SWE(마지막 행, 'all 1'이 출력된 열)에서 VS 고장이 발생했음을 알 수 있다.

두 번째는 폐기출력에서 'all 1'이 출력되고 출력링크에서 정상상태에 비해 출력 셀이 달라진 형태이다(그림 14), ST1 TYPE II). (고장 난 행 + 1)번째 행에서 'all 1'이 출력되고 고장 난 열에서 다른 셀(X1 → X7)이 출력된다. 따라서 SWE('all 1'이 출력된 행 1, 출력 셀이 달라진 열)에서 VS 고장이 발생했음을 알 수 있다.



(그림 14) ST1 고장에 대한 Location (ST1 TYPE II)

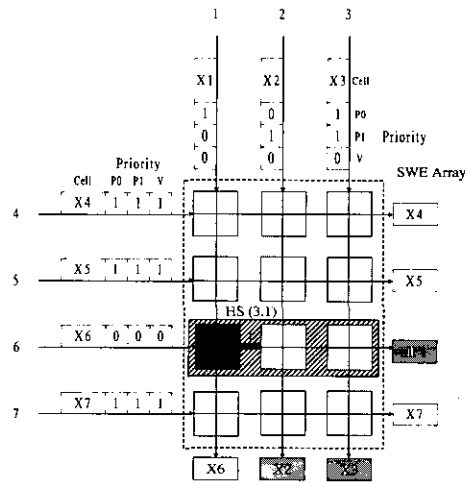
세 번째는 폐기출력에서 'all 1'이 출력되고 나머지 출력



(그림 15) ST1 고장에 대한 Location (ST1 TYPE III)

값에는 변화가 없는 형태이다(그림 15), ST1 TYPE III). 1 단계 오프라인테스트를 통해 SWE('all 1'이 출력된 행, ?)에서 HS 고장이 발생했음을 알 수 있다. 열 값을 알아내기 위해서는 2단계 오프라인테스트를 수행해야 한다.

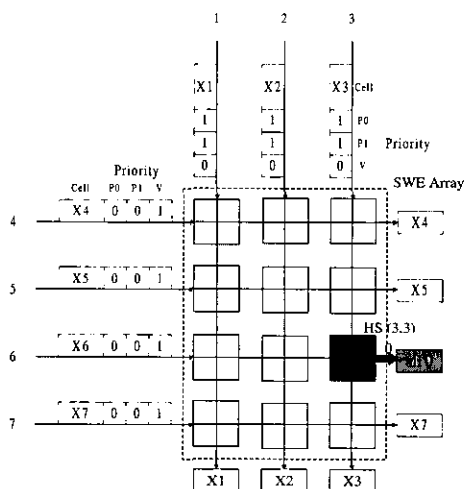
(그림 16)은 ST1 TYPE III에 대한 2단계 오프라인테스트를 보여준다. 이것은 CS Location과 비슷하다. 고장 난 행(3행)에 대해 테스트 블록을 설정하고, X6 > X1 > X2 > X3 순으로 우선순위를 설정하고 나머지 셀들(X4, X5, X7)의 우선순위 값은 '111'로 설정한다. 그러면 블록내부의 SWE들은 토글 상태로 동작하고 나머지 SWE들은 크로스상태로 동작하게 된다. ST1 HS(3,1)으로 인해 X1이 출력링크에서 검출되지 않는다. 따라서 SWE(테스트 블록이 설정된 행, 테스트 블록으로 입력된 셀들 중 출력에서 검출되지 않는 셀이 입력된 열)에서 HS 고장이 발생했음을 알 수 있다.



(그림 16) ST1 TYPE III에 대한 2단계 오프라인테스트

#### 4.2.4 ST0 고장의 경우

ST0 고장은 3가지 유형으로 분류할 수 있다. 첫 번째 유

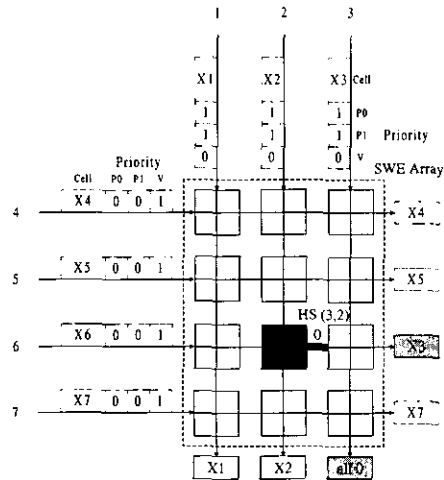


(그림 17) ST0 고장에 대한 Location

형은 HS 고장에 의해 'all 0'가 폐기출력으로 출력되는 형태로서 ST0 TYPE I으로 정의한다. 두 번째 유형은 HS 고장에 의해 'all 0'이 출력링크로 출력되는 형태로서 ST0 TYPE II로 정의한다. 세 번째 유형은 VS 고장에 의해 'all 0' 값이 출력링크로 출력되는 형태로서 ST0 TYPE III로 정의한다.

ST0 고장에 대해 TP1을 입력하면 3가지 형태로 출력된다. 첫번째는 폐기출력에서 'all 0'가 출력되는 형태이다(그림 17), ST0 TYPE I). 마지막 열에서 HS 고장이 존재하는 경우에만 나타나며, 고장 난 행에서 'all 0'가 출력된다. 따라서 SWE('all 0'가 출력된 행, 마지막 열)에서 HS 고장이 발생했음을 알 수 있다.

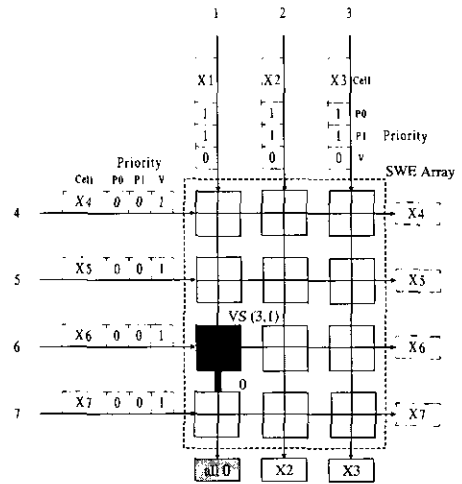
두 번째는 출력링크에서 'all 0'가 출력되고 폐기출력에서 정상상태에 비해 출력 셀이 달라진 형태이다(그림 18), ST0 TYPE II). 고장 난 행에서 다른 셀(X6 → X3)이 출력되고, (고장 난 열 + 1)번째 열에서 'all 0'가 출력된다. 따라서 SWE(출력 셀이 달라진 행, 'all 0'가 출력된 열 1)에서 HS 고장이 발생했음을 알 수 있다.



(그림 18) ST0 고장에 대한 Location (ST0 TYPE II)

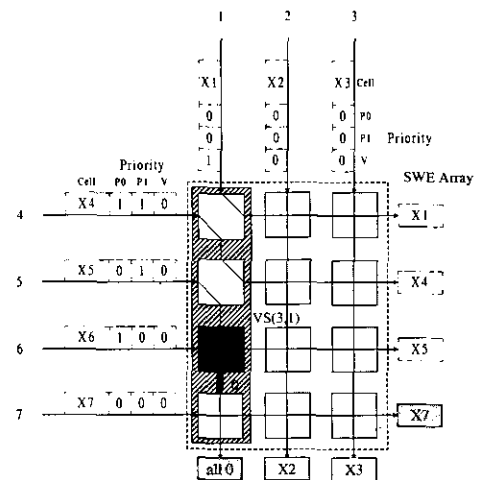
세 번째는 출력링크에서 'all 0'가 출력되고 나머지 출력 값에는 변화가 없는 형태이다(그림 19), ST0 TYPE III). 1 단계 오프라인테스트를 통해 SWE(?, 'all 1'이 출력된 열)에서 VS 고장이 발생했음을 알 수 있다. 행 값을 알아내기 위해서는 2단계 오프라인테스트를 수행해야 한다.

(그림 20)은 ST0 TYPE III에 대한 2단계 오프라인테스트를 보여준다. 이것은 CS Location과 비슷하다. 고장 난 열(1열)에 대해 테스트 블록을 설정하고, X7 > X6 > X5 > X4 > X1 순으로 우선순위를 설정하고 나머지 셀들(X2, X3)의 우선순위 값은 '111'로 설정한다. 그러면 블록 내부의 SWE들은 토글 상태로 동작하고 나머지 SWE들은 크로스 상태로 동작하게 된다. ST0 VS(3,1)으로 인해 X6가 'all 0'로 출력되어 X6가 검출되지 않는다. 따라서 SWE(테스트



(그림 19) ST0 고장에 대한 Location (ST0 TYPE III)

블록으로 입력된 셀들 중 출력에서 검출되지 않는 셀이 입력된 행, 테스트 블록이 설정된 열)에서 VS 고장이 발생했음을 알 수 있다.



(그림 20) ST0 TYPE III에 대한 2단계 오프라인테스트

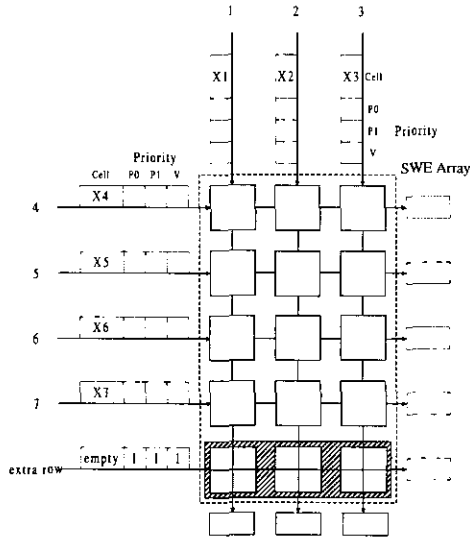
## 5. 새로운 스위치 구조 및 고장 감내 기법

### 5.1 스위치 구조

(그림 21)은 제안된 스위치의 스위치모듈구조를 보여준다. 제안된 구조는 출력링크 수만큼의 SWE를 더 사용하여 입력링크를 하나 늘린 것으로, 늘어난 행을 'EXTRA ROW'라 칭한다. 정상상태에서 EXTRA ROW에는 빈 셀('111')이 입력된다. 이때 EXTRA ROW의 SWE들은 모두 크로스상태로 되어 빈 셀은 폐기출력의 EXTRA ROW로 출력되고 각 SWE의 북쪽입력 셀들은 출력링크의 동일한 열로 출력된다. 따라서 EXTRA ROW는 스위치의 성능저하를 가져오지 않는다. 만일 고장이 발생하면 고장이 발생한 라인으로의 입력을 EXTRA ROW에 연결하고 고장이 발생한 라



인의 모든 SWE들을 크로스상태로 강제하여 고장이 발생한 라인을 제거함으로써 고장을 극복할 수 있다.

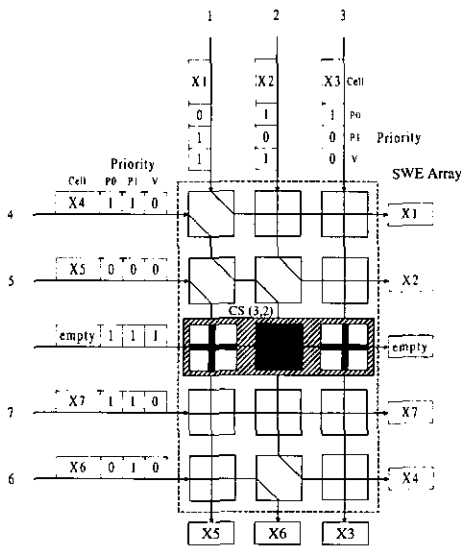


(그림 21) 제안된 스위치의 스위치 모듈구조

5.2 고장 감내를 위한 Reconfiguration 기법

5.2.1 CS 고장의 경우

(그림 22)는 CS(3,2)에 대한 Reconfiguration을 보여준다. 3행에서 CS 고장이 존재하므로 3행으로의 입력을 EXTRA ROW에 연결하고 3행의 모든 SWE들을 크로스상태로 강제하며 3행에는 빈 셀을 입력한다. 그러면 빈 셀은 폐기출력의 3행으로 출력되어 3행이 제거된 것과 같은 효과를 가진다. 따라서 CS(3,2)는 스위치의 동작에 아무런 영향을 주지 않는다. 즉 입력으로 들어오는 모든 셀들이 출력 링크나 폐기 출력으로 향하게 된다. 즉, ROW를 스위치에 하나 더

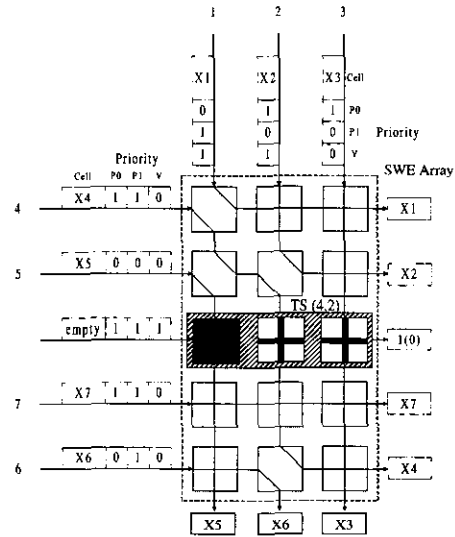


(그림 22) CS(3,2) 고장에 대한 Reconfiguration

연결함으로 고장이 발생한 라인을 제거한 것과 같은 효과를 가지게 된다. 따라서, Reconfiguration 수행 후 스위치의 성능저하가 없다.

5.2.2 TS 고장의 경우

TS 고장에 대한 Reconfiguration은 CS 고장의 경우와 마찬가지로 고장 난 행의 모든 SWE들을 크로스상태로 만들어주고 빈 셀을 입력해줌으로써 고장 난 행이 제거되는 효과를 가지도록 한다. (그림 23)은 TS(4,2)에 대한 Reconfiguration을 보여준다. 4행으로의 입력은 EXTRA ROW에 연결하고 4행의 모든 SWE를 크로스상태로 강제하여 4행을 제거하는 효과를 가진다. 즉 입력으로 들어오는 모든 셀들이 출력 링크나 폐기 출력으로 향하게 된다. 즉, ROW를 스위치에 하나 더 연결함으로써 고장이 발생한 라인을 제거한 것과 같은 효과를 가지게 된다. 따라서, Reconfiguration 수행 후 스위치의 성능저하가 없다.



(그림 23) TS(4,2)에 대한 Reconfiguration

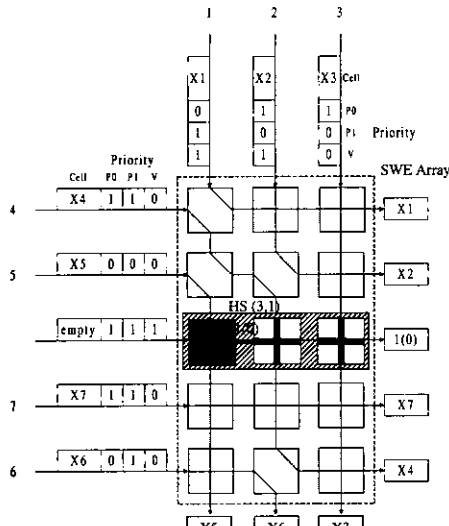
5.2.3 HS 고장의 경우

HS 고장은 CS 고장이나 TS 고장처럼 고장난 행을 제거함으로써 고장을 극복할 수 있다. (그림 24)는 HS(3,1)에 대한 Reconfiguration을 보여준다. 3행에서 HS 고장이 존재하므로 3행으로의 입력은 EXTRA ROW에 연결하고 3행의 모든 SWE들을 크로스로 강제하여 'all 1' 혹은 'all 0' 값이 폐기출력의 3행으로 출력되어 폐기되도록 한다. 이는 3행이 제거되는 것과 같은 효과를 가지며, Reconfiguration 수행 후 스위치의 성능저하가 없다.

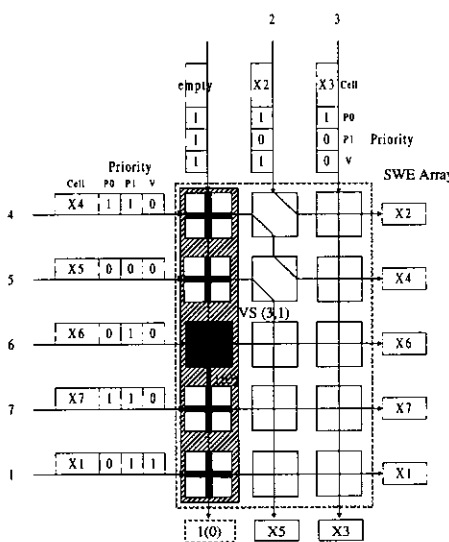
5.2.4 VS 고장의 경우

VS 고장은 다른 고장들과는 달리 고장난 열을 제거함으로써 고장을 극복할 수 있다. (그림 25)는 VS(3,1)에 대한 Reconfiguration을 보여준다. 1열에서 VS 고장이 존재하

로 고장난 열로의 입력을 EXTRA ROW에 연결하고 고장난 열을 제거하기 위해 고장난 열의 모든 SWE를 크로스상태로 강제하고 빈 셀을 입력한다. 출력링크의 고장난 열에는 'all 1' 혹은 'all 0'가 출력되므로 해당 출력링크를 끊어주어야 한다. 그러므로 출력링크가 하나 감소한다. 하지만 Reconfiguration 수행 후 VS 고장은 스위치의 동작에 아무런 영향을 미치지 않는다.



(그림 24) HS(3,1)에 대한 Reconfiguration



(그림 25) VS(3,1)에 대한 Reconfiguration

5.3 성능분석

이 장에서는 본 논문에서 제안한 스위치의 셀 손실 성과 첫 번째와 두 번째 단계(stage: MGN1, MGN2)에서의 결과에 대해 설명한다. 제안된 스위치는 SWE수를 줄이기 위해 MOBAS를 변형한 형태로써, 전체적으로 MOBAS와 유사한 구조를 가진다. 따라서 성능분석을 위해 MOBAS에서 사용된 수식을 그대로 적용할 수 있다.

5.3.1 MGN1에서의 셀 손실률

성능 분석을 위하여, 각 입력에서의 트래픽은 다른 입력에서의 트래픽과 독립적이며, 복제 셀들은 모든 출력그룹에 균일하게 도착한다고 가정한다. 평균 셀 도착율( $\rho$ )은 1 셀 시간동안 하나의 입력포트에 셀이 도착할 확률이다.  $E[F_1]$ 는 MGN1의 평균 셀 복제율,  $E[F_2]$ 는 MGN2의 평균 셀 복제율,  $E[D]$ 는 OPC의 평균 셀 복제율을 나타낸다. 성능 분석시의 가정에 따라, MGN1에서의 출력 셀들은 MGN2로의 입력으로 균일하게 도착한다고 가정된다. 또한 OPC로 향하는 MGN2의 출력 셀들도 같은 가정을 가진다. 따라서,  $F_1, F_2, D$  등은 서로 독립적인 랜덤 변수들로 가정된다.

각 셀 들은 모든 concentrator(SM's)로 브로드캐스트되며, 셀 헤더에 있는 multicast pattern에 따라 각 concentrator에서 정확히 필터링(filtering)된다. 각 concentrator의 입력에서의 평균 셀 도착율은 다음과 같다.

$$\rho = \frac{\rho E[F_1]}{K}$$

$K(= N/M)$ 는 MGN1에서의 concentrator의 수이다. 1셀 시간동안,  $k$ 개의 셀이 MGN1의 어떤 스위치모듈로 향한 확률 ( $A_k$ )은 (5-1)과 같다.

$$A_k = \binom{N}{k} \rho^k (1-\rho)^{N-k} = \binom{N}{k} \left( \frac{\rho E[F_1] \cdot M}{N} \right)^k \left( 1 - \frac{\rho E[F_1] \cdot M}{N} \right)^{N-k}, \quad 0 \leq k \leq N \quad (5-1)$$

$\rho$ 는 MGN1에서의 특정한 concentrator의 입력으로의 셀 도착율이다.  $N \rightarrow \infty (N \geq 1024)$ 이면, (5-1)은 다음과 같이 된다.

$$A_k = e^{-\rho E[F_1] \cdot M} \cdot \frac{(\rho E[F_1] \cdot M)^k}{k!} \quad (5-2)$$

$\rho$ 는 안정된 시스템에서 다음과 같은 조건을 만족하여야 한다.

$$\rho \cdot E[F_1] \cdot E[F_2] \cdot E[D] < 1$$

각 출력 그룹 당  $L_1 \times M$ 개의 라우팅 링크만이 유효하기 때문에, 한 타임 슬롯 동안에 출력 그룹으로 향하는 셀의 개수가  $L \times M$ 을 초과하는 경우에, 초과하는 셀들은 폐기된다. MGN1에서의 셀 손실률 ( $P_1$ )은 (5-3)과 같다.

$$P_1 = \frac{1}{N \cdot \rho \cdot E[F_2] E[D]} \sum_{k=L_1 \times M+1}^N (k-L_1 \times M) \cdot A_k \cdot E[F_2] E[D] \quad (5-3)$$

$$= \frac{1}{\rho \cdot E[F_1] \cdot M} \sum_{k=L_1 \times M+1}^N (k-L_1 \times M) \cdot \binom{N}{k} \left( \frac{\rho E[F_1] \cdot M}{N} \right)^k \left( 1 - \frac{\rho E[F_1] \cdot M}{N} \right)^{N-k} \quad (5-4)$$

(5-3)의 분모에서 나타나는  $N \cdot \rho \cdot E[F_2]E[D]$ 은 한 타임 슬롯동안에 특정 concentrator에 도착하는 평균 유효 셀의 개수이다.

$$\sum_{k=L_1+M+1}^N (k-L_1 \times M) \cdot A_k \cdot E[F_2]E[D]$$

은 특정 concentrator에서 폐기되는 평균 유효 셀의 개수이다. (5-3)에서 나타나듯이 MGN2와 OPC에서 셀 복제를 처리하기 위하여 분모와 분자에  $E[F_2]E[D]$ 를 공통으로 포함한다.  $N \rightarrow \infty (N \geq 1024)$ 이면, (5-4)는 (5-5)와 같이 된다.

$$\begin{aligned} P_1 &= \left[ 1 - \frac{L_1 \times M}{N \times p} \right] \left[ 1 - \sum_{k=0}^{L_1 \times M} \frac{(N \times p)^k e^{-N \times p}}{k!} \right] \\ &+ \frac{(N \times p)^{L_1 \times M} e^{-N \times p}}{(L_1 \times M)!} \\ &= \left[ 1 - \frac{L_1 \times M}{\rho E[F_1] \times M} \right] \left[ 1 - \sum_{k=0}^{L_1 \times M} \frac{(\rho E[F_1] \cdot M)^k e^{-\rho E[F_1] \cdot M}}{k!} \right] \\ &+ \frac{(\rho E[F_1] \cdot M)^{L_1 \times M} e^{-\rho E[F_1] \cdot M}}{(L_1 \times M)!} \end{aligned} \quad (5-5)$$

### 5.3.2 MGN2에서의 셀 손실률

MGN2의 입력 단에 도착하는 셀 도착 패턴은 MGN1에서 상응하는 concentrator를 통과하는 셀의 개수에 의해 결정되므로, MGN2에서 셀 손실률을 분석할 때에는 특별한 주의가 필요하다. MGN2로  $l (l \leq L_1 \times M)$ 의 셀이 도착하면, MGN2의 입력에서 위쪽부터  $l$ 개가 연속적으로 나타난다.  $L_1 \times M$ 보다 많은 개수의 셀이 MGN2로 향하게 되면,  $L_1 \times M$ 개의 셀만이 MGN2의 입력 단(각 포트 당 하나의 셀)에 나타나며 나머지 셀들은 MGN1에서 폐기된다.

MGN2에 있는 concentrator에서의 셀 손실률은  $P_2$ , MGN2에 있는 특정 concentrator에  $l$ 개의 셀이 도착할 확률은  $B_l$ 라 한다.  $P_2$ 와  $B_l$ 는 MGN2의 입력에 도착하는 평균 셀 수에 의존한다(즉, MGN1에서 상응하는 concentrator를 통과하는 셀의 개수). 이것은  $P_2$ 가 (5-1)에 나타난  $A_k$ 의 함수라는 것을 의미한다.  $P_2$ 를 계산하기 위하여 사용하는,  $j$ 개의 셀이 MGN2에 도달할 확률, ( $I_j$ )는 다음과 같다.

$$I_j = \begin{cases} A_j & \text{for } j < L_1 \times M \\ \sum_{k=L_1 \times M}^N A_k & \text{for } j = L_1 \times M \end{cases}$$

MGN2로  $j (j \leq L_1 \times M)$ 개의 셀이 도착하면, MGN2의 입력에서 위쪽부터  $j$ 개가 연속적으로 나타난다. 셀이 MGN2의 어디에서 어떻게 나타나는지는 셀 손실 평가와는 무관하므로, 셀이 MGN2의 어떤 입력에서라도 나타날 수 있다는 가정 하에 간단하게 성능 분석을 할 수 있다.

$j$ 개의 셀이 MGN2에 도달하였을 때,  $l$ 개의 셀이 MGN2의 특정 concentrator의 입력에 도달할 확률 ( $B_{l,j}$ )은 다음과 같다.

$$B_{l,j} = \binom{j}{l} q^l (1-q)^{j-l} \quad 0 \leq j \leq L_1 \times M, \quad 0 \leq l \leq j$$

복제된 셀이 MGN2의  $M$ 개의 concentrator로 균등하게 도착한다고 가정하면  $q$ 는  $\frac{E[F_2]}{M}$  같다.

MGN2로  $L_2$ 와 같거나 적은 수의 셀이 도착하면 MGN2 ( $0 \leq j \leq L_2$ )에서 셀 폐기가 일어나지 않는다. 따라서, 각 concentrator에서는 한 타임 슬롯 동안에  $L_2$ 개까지의 셀을 받아들일 수 있다. MGN2 ( $L_2 \leq j \leq L_1 \times M$ )로  $L_2$ 개 이상의 셀이 도착하면 특정 concentrator에서 셀 손실이 일어나게 된다.

복제된 셀이 MGN2의  $M$ 개의 concentrator로 균등하게 도착한다고 가정하였기 때문에, 한 타임 슬롯 동안에 MGN2의 특정 출력 포트에  $l$ 개의 셀이 도착할 확률,  $B_l$ 는 다음과 같다.

$$B_l = \sum_{j=l}^{L_1 \times M} B_{l,j} \cdot I_j$$

MGN2에서의 셀 손실률( $P_2$ )는 (5-6)과 같다.

$$\begin{aligned} P_2 &= \frac{\sum_{l=L_2+1}^{L_1 \times M} (l-L_2) \cdot B_l \cdot E[D]}{N \frac{\rho E[F_1]}{K} (1-P_1) E[F_2]} E[D] \\ &= \frac{\sum_{l=L_2+1}^{L_1 \times M} (l-L_2) \cdot B_l}{\rho E[F_1] \cdot (1-P_1) \cdot E[F_2]} \end{aligned} \quad (5-6)$$

입력으로부터 MGN1의 특정 concentrator로 향하는 셀의 평균 개수는  $N \frac{\rho E[F_1]}{K}$ 이다.  $N \frac{\rho E[F_1]}{K} \cdot (1-P_1)$ 은 concentrator에서 폐기되지 않는 셀의 평균 개수를 나타내며, 이 셀들은 차례대로 상응하는 MGN2의 평균 셀 도착 개수가 된다. 따라서, (5-6)에 나타나는 분모  $N \frac{\rho E[F_1]}{K} \cdot (1-P_1) E[F_2]E[D]$ 는 특정 출력 포트에 도착하는 평균 유효 셀의 개수이다. 폐기 셀이 OPC에서 복제될 수 있기 때문에, (5-6)에 나타나는 분자  $\sum_{l=L_2+1}^{L_1 \times M} (l-L_2) \cdot B_l \cdot E[D]$ 는 MGN2의 특정 concentrator에서 폐기되는 평균 유효 셀의 개수이다.

## 6. 실험 및 고찰

### 6.1 스위치 크기

스위치의 크기는 사용된 SWE의 수와 관련이 있다. Modified MOBAS는 MOBAS에 비해 적은 수의 SWE가 사용된다. 제안된 스위치는 고장감내를 위해 Modified MOBAS보다 약간의 SWE가 더 사용되지만, MOBAS보다는 적은 수의 SWE가 사용된다.

일반적으로 스위치의 크기는 입력 수  $N$ , 그룹 크기  $M$ , 확

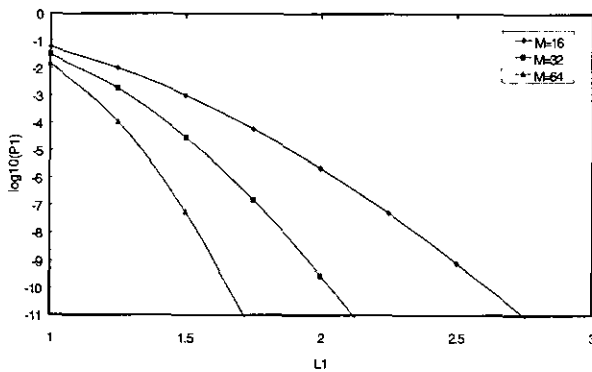
장비  $L_1, L_2$  등에 의해 결정된다. 여기서  $N$ 과  $M$ 은 임의적으로 정해줄 수 있는 값들이고,  $L_1$ 과  $L_2$ 는 요구되는 전체 스위치의 셀 손실률에 따라 계산되어야 하는 값들이다.

MGN1은 ( $K = N/M$ )개의 그룹을 가지고, MGN2는  $M$ 개의 그룹을 가진다. MGN1에서의 각 스위치모듈은  $N$ 개의 수평 입력 라인과  $L_1 \times M$ 개의 수직 라우팅 링크를 가진다. MGN2에서의 스위치모듈은  $L_1 \times M$ 개의 수평 입력과  $L_2$ 개의 출력을 가진다. SM1은  $\{N - (L_1 \times M) + 1\} \times (L_1 \times M)$ 개의 SWE로 구성되며, SM2는  $\{(L_1 \times M) - L_2 + 1\} \times L_2$ 개의 SWE로 구성된다. 한편 MGN1과 MGN2는 각각  $K$ 개의 SM1과  $M$ 개의 SM2를 가지므로 전체 스위치에 사용된 SWE의 수는  $K \times [\{N - (L_1 \times M) + 1\} \times (L_1 \times M)] + K \times M \times [\{(L_1 \times M) - L_2 + 1\} \times L_2]$ 와 같다.

6.2 결과 및 고찰

6.2.1 MGN1에서의 셀 손실률

(그림 26)은 다른 값을 가지는  $L_1$ 에 대한 MGN1에서의 셀 손실률 ( $P_1$ ) 값의 변화를 보여준다. 트래픽 조건은 유효 부하 ( $= \rho \cdot E[F_1] \cdot E[F_2] \cdot E[D]$ )가 0.9인 유니캐스트( $E[F_1] = 1, E[F_2] \cdot E[D] = 1$ )로 가정한다. 유니캐스트에 대한 성능 조건을 만족하는 MGN은 멀티캐스트에 대한 성능조건도 만족한다. 따라서, 본 논문에서는 유니캐스트( $E[F_1] = 1, E[F_2] \cdot E[D] = 1$ )에 대해서만 성능분석을 수행한다. 그룹크기( $M$ )는 16, 32, 64이다.  $M$ 이 증가하면, 동일한 셀 손실률을 만족하는  $L_1$ 이 감소한다. MGN1에서  $10^{-10}$ 이하의 셀 손실률을 가지기 위해서는  $L_1$ 은 2.6( $M = 16$ ), 2.1( $M = 32$ ), 1.7( $M = 64$ ) 이상이어야 한다.

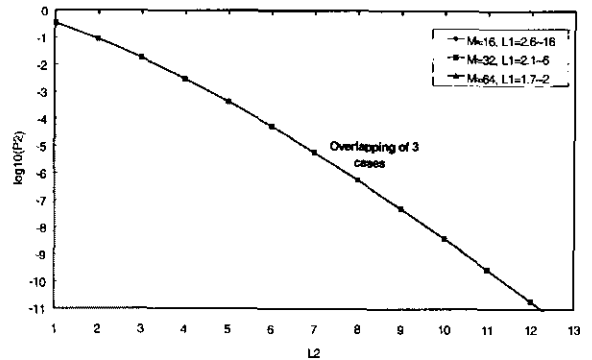


(그림 26)  $L_1$ 에 대한 MGN1에서의 셀 손실률

6.2.2 MGN2에서의 셀 손실률

(그림 27)은  $L_2$ 에 대한 MGN2에서의 셀 손실률 ( $P_2$ )값의 변화를 보여준다. 트래픽 조건은 유효부하가 0.9인 유니캐스트로 가정한다. 그룹크기( $M$ )는 16, 32, 64이며,  $L_1$ 은 2.6~16( $M = 16$ ), 2.1~6( $M = 32$ ), 1.7~2( $M = 64$ )이다. 이는

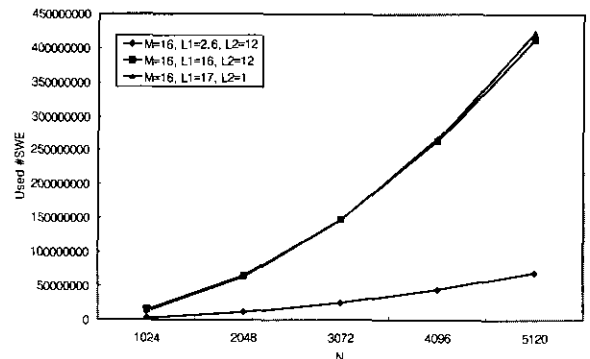
MGN1에서의 셀 손실률이  $10^{-10}$ 이하가 되도록 하는 값이며,  $L_1$ 이 이보다 더 커지면  $L_2(L_2 \geq 1)$ 에 관계없이 MGN2에서의 셀 손실률은 거의 0이 되어 그래프상에 나타나지 않는다. 그래프상의 하나의 선은 실제 3개의 선이 겹쳐진 것으로  $M$ 에 관계없이 동일한 값을 가진다. MGN2에서  $10^{-10}$ 이하의 셀 손실률을 가지기 위해서는  $L_2$ 는 12 이상이어야 한다.



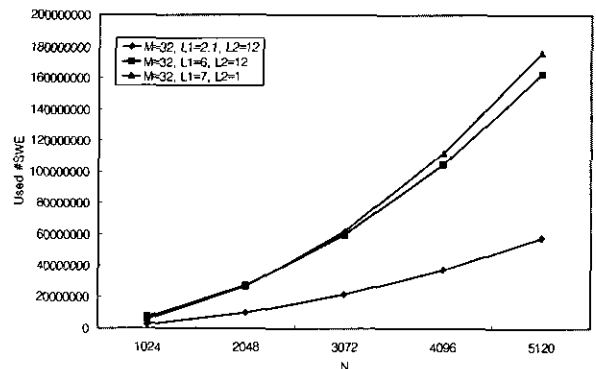
(그림 27)  $L_2$ 에 대한 MGN2에서의 셀 손실률

6.3 사용된 스위치소자수의 비교

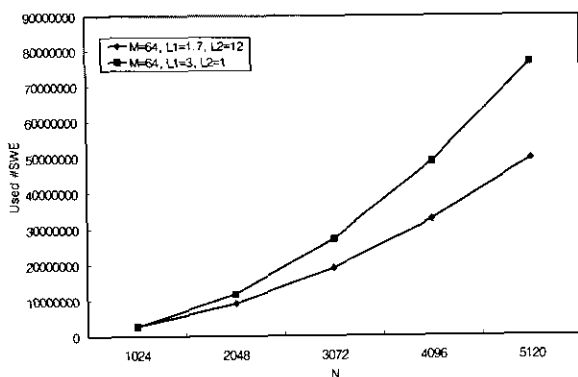
(그림 28)은  $M = 16, 32, 64$ 일 때, 사용된 전체 스위치소자의 수를 나타낸다. ( $L_1, L_2$ )값은 스위치가  $10^{-10}$ 의 셀 손실률을 가지도록 하는 값이며 (2.6, 12)( $M = 16$ ), (2.1, 12)( $M = 32$ ), (1.7, 12)( $M = 64$ )일 때, 가장 적은 스위치소자를 사용한다.



(그림 28 (a)) 사용된 스위치소자의 수,  $M = 16$



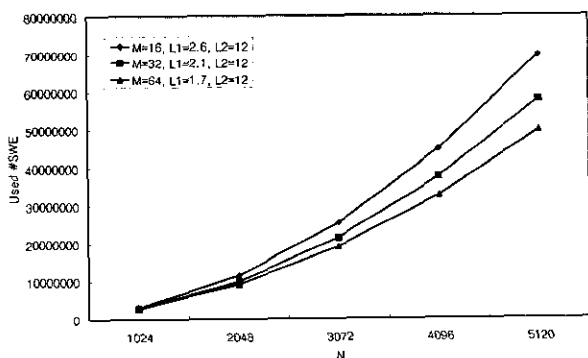
(그림 28 (b)) 사용된 스위치소자의 수,  $M = 32$



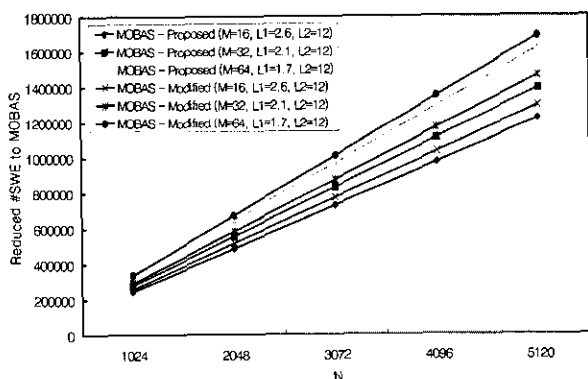
(그림 28 (c)) 사용된 스위치소자의 수,  $M = 64$

(그림 29)는 사용된 전체 스위치소자의 수를  $M(=16, 32, 64)$ 에 대해 비교한 그래프이다. 그룹크기  $M$ 을 늘리면 더 많은 스위치소자를 줄일 수 있다.

(그림 30)은 MOBAS에 비해 줄어드는 스위치소자의 수를 비교한 그래프이다.  $N=1024$ 일 때, 제안된 스위치는 MOBAS에 비해 7.5%( $M=16$ ), 9.2%( $M=32$ ), 10.3%( $M=64$ )의 스위치소자를 줄일 수 있고, Modified MOBAS는 MOBAS에 비해 8%( $M=16$ ), 9.6%( $M=32$ ), 10.8%( $M=64$ )의 스위치소자를 줄일 수 있다. 그룹크기  $M$ 을 늘리면 더 많은 스위치를 줄일 수 있다. 한편, 제안된 스위치는 Modified MOBAS에 비해 대략 0.5%의 스위치소자가 더 사용된다.



(그림 29) 사용된 전체 스위치소자수의 비교



(그림 30) MOBAS에 비해 줄어드는 스위치소자의 수

## 7. 결 론

스위치의 크기뿐 아니라 고장감내 특성은 위성탐재 ATM 스위치의 설계에 있어서 중요한 요소이다. Modified MOBAS (Multicast Output buffered ATM Switch)는 MOBAS와 유사한 구조를 가지지만 스위치모듈의 구조에서 차이를 보이며 MOBAS보다 적은 스위치소자를 사용한다. 스위치의 크기를 줄일 수 있어 위성탐재스위치로서 적합하지만 고장감내특성을 가지고 있지 않으므로 이에 대한 연구가 필요하다. 또한 MOBAS와 다른 고장특성을 가지므로 동일한 고장감내기법을 적용할 수 없다.

본 논문에서는 Modified MOBAS의 고장특성을 분석하고 이에 적합한 Detection 기법 및 Location 기법을 제안하였다. 또한 스위치모듈구조를 변형하여 Modified MOBAS의 장점을 그대로 수용하면서 고장감내특성을 가지는 새로운 스위치를 제안하였다. 제안된 스위치는 Modified MOBAS에 비해 약간의 스위치소자를 더 사용하지만 MOBAS에 비해서는 여전히 적은 스위치소자를 사용한다. 또한, 제안된 스위치는 MOBAS에 비해 스위치크기를 줄일 수 있으며 여전히 MOBAS와 동일한 성능을 가진다. 또한 고장에 대한 Location 및 Reconfiguration이 간단하고 고장감내 수행 후 성능저하가 없다.

제안된 스위치는 MOBAS에 비해 10%( $N=1024, M=32$ )의 스위치소자를 줄일 수 있다.  $M$ 이 증가할수록 스위치소자의 감소율이 더 커진다. 따라서, 제안된 스위치는 Modified MOBAS와 비슷한 크기를 가지며 우수한 고장감내특성을 지니고 있으므로 위성탐재ATM스위치에 적합하다.

## 참 고 문 헌

- [1] CCITT SG XVIII Draft Recommendation I.150 : "B-ISDN ATM Functional Characteristics," Geneva, May 1990.
- [2] R. Handel, M. N. Hunber, and S. Schröber, "ATM Networks : Concepts, Protocols, Applications," Addison Wesley, 1994.
- [3] M. D. Prycker, "Asynchronous Transfer Mode : solution for Broadband ISDN," Prentice Hall, 1995.
- [4] C. Partridge, "Gigabit Networking," Addison Wesley, 1992.
- [5] H. Ahmadi, W. E. Denzel, "A Survey of Modern High-Performance Switching Techniques," IEEE J. Select. Areas Commun., Vol.7, pp.1091-1103, Sep. 1989.
- [6] W. E. Stevens, M. D. Prycker, F. A. Tobagi, and T. Yamaguchi, Special issue on "Large-Scale ATM Switching Systems for B-ISDN," IEEE J. Select. Areas Commun., Vol.9, No.8, Oct. 1991.
- [7] G. B. Adams III, D. P. Agrawal, and H. J. Siegel, "A Survey and Comparison of Fault-Tolerant Interconnection Networks," IEEE Trans. Comput., pp.14-27, Jun. 1987.

[8] A. Itoh, "A Fault-Tolerant Switching Networks for B-ISDN," *IEEE J. Select. Areas Commun.*, Vol.9, No.8, pp.1218-1226, Oct. 1991.

[9] S. C. Yang and J. A. Silvester, "A Reconfiguration ATM Switch Fabric for Fault Tolerance and Traffic Balancing," *IEEE J. Select. Areas Commun.*, Vol.9, No.8, pp.1205-1217, Sep. 1988.

[10] M. Jeng and H. J. Siegel, "Design and Analysis of Dynamic Redundancy Networks," *IEEE Trans. Comput.*, Vol.37, No.9, pp.1019-1029, Sep. 1988.

[11] V. P. Kumar and A. L. Reibman, "Fault Dependent Performance Analysis of a Fault-Tolerant Multistage Interconnection Network," *IEEE Trans. Comput.*, Vol.38, No.12, pp.1703-1713, Dec. 1989.

[12] A. Varma and S. Chalasani, "Fault-Tolerance Analysis of One-Sided Crosspoint Switching Networks," *IEEE Trans. Comput.*, Vol.41, No.2, pp.143-158, Feb. 1992.

[13] T. H. Lee and J. J. Chou, "Fault Tolerance of Banyan Using Multiple-Pass," in *Proc. pp.867-875, INFOCOM '92*.

[14] D. P. Agrawal, "Testing and Fault-Tolerance of Multistage Interconnection Networks," *IEEE Computer*, Vol.15, pp.41-53, Apr. 1982.

[15] R. Jacob, "A Survey of Fast Packet Switches," *ACM Computer Communication Review*, Vol.20, No.1, pp.54-64, Jan. 1990.

[16] H. J. Chao and B. Choe, "A Large-Scale Multicast Output Buffered ATM Switch," *GLOBECOM '93*, Houston, TX, Dec. 1993.

[17] Choe and H. J. Chao, "Performance Analysis of a Large-Scale Multicast Output Buffered ATM Switch," *INFOCOM '94*, San Francisco, CA, Jun, 1994.

[18] H. J. Chao and B. Choe, "Design and Analysis of a Large-Scale Multicast Output Buffered ATM Switch," *IEEE/ACM Trans. Networking*, Vol.3, No.2, pp.126-138, Apr. 1995.

[19] Choe and H. J. Chao, "Fault-Tolerance of a Large-Scale Multicast Output Buffered ATM Switch," *IEEE*, 1994.

[20] M. J. Karol and M. G. Hluchyj, "The Knockout Switch : Principles and Performance," *Proc. 12th Conf. Local Comput. Network*, pp.16-22, Oct. 1987.

[21] K. Y. Eng, M. J. Karol, and Y. S. Yeh, "A Growable Packet (ATM) Switch Architecture : design principles and applications," *IEEE J. Select. Areas Commun.*, Vol.SAC-5, pp.1274-1283, Oct. 1987.

[22] K. Wang and M.-H. Cheng, "Design and Performance Analysis of a Growable Multicast ATM Switch," *INFOCOM '97*, Kobe, Japan, Vol.3, pp.934-940, Apr. 1997.



**권 세 동**

e-mail : sdkwon@wh.myongji.ac.kr  
 1997년 명지대학교 컴퓨터공학과 졸업(학사)  
 1999년 명지대학교 컴퓨터공학과 졸업(석사)  
 1999년 현재 명지대학교 컴퓨터공학과  
 박사과정  
 1999년~현재 (주)거성 텔레콤 연구원  
 관심분야 : ATM 교환기, ATM 트래픽 제어, 컴퓨터 네트워크



**박 현 민**

e-mail : hpark@wh.myongji.ac.kr  
 1985년 서울대학교 전자공학과 졸업(학사)  
 1988년 North Carolina State University  
 Electrical and computer engineering  
 (석사)  
 1995년 North Carolina State University  
 Electrical and computer engineering  
 (박사)  
 1996년~현재 명지대학교 공과대학 컴퓨터학부 부교수  
 관심분야 : ATM 교환기, 네트워크 트래픽 관리 및 보안 기술  
 연구



**최 병 석**

e-mail : bcchoe@cakra.dongguk.ac.kr  
 1985년 서울대학교 전자공학과(학사)  
 1987년 Fairleigh Dickinson University  
 전자공학(석사)  
 1994년 Polytechnic University 전자공학  
 (박사)  
 1994년~1997년 명지대학교 전자공학과 조교수  
 1997년~현재 동국대학교 정보산업대학 정보통신공학부 조교수  
 관심분야 : 위성 ATM 교환기 개발, 광대역 가입자 접속 방식,  
 초고속 위성망



**박 재 현**

e-mail : jhpark@wh.myongji.ac.kr  
 1984년 서울대학교 전기공학과 졸업  
 1986년 Polytechnic University 대학원  
 전자공학(공학석사)  
 1993년 Polytechnic University 대학원  
 전자공학(공학박사)  
 1994년~1996년 삼성전자 마이크로사업부 선임연구원  
 1996년~현재 명지대학교 전자정보통신공학부 조교수  
 관심분야 : 컴퓨터 네트워크, 통신이론, Statistical 신호처리