새로운 Dynamic GSMP V3 구조의 VLSI 설계

김 영 철[†] · 이 태 원[††] · 김 광 옥[†††] · 이 명 옥[††††]

요 약

본 논문에서는 ATM 기반 MPLS 망에서 효율적으로 IP 서비스를 전송하기 위한 동적 버퍼관리 방식의 Dynamic GSMP V3(General Switching Management Protocol Version 3)의 VLSI 구현을 위한 하드웨어 구조를 제안하고 설계하였다. 또한 현재 표준화중인 GSMP와 동적 버퍼관리 방식을 수용한 GSMP를 셀 손실률 측면에서 비교 분석하였다. ATM 스위치 상에 연결 제어의 성능 향상을 위해 스위치에서 연결설정 및 제어를 수행하는 Dynamic GSMP V3의 Slave 블록을 삼성 SoG 0.5μm 공정으로 설계하였다. 기존의 방식과 제안한 방식의 성능 평가를 위해 확률 랜덤 변수에 의해 발생된 셀과 최소 버퍼 알고리즘을 이용하여 모의 실험을 하였으며, 이때 셀 손실률이 향상되었음을 알 수 있었다.

VLSI Design of a New Dynamic GSMP V3 Architecture

Young-Chul Kim[†] · Tae-Won Lee[††] · Kwang-Ok Kim[†††] · Mike Myung-Ok Lee[††††]

ABSTRACT

In this paper, the hardware architecture of Dynamic GSMP V3 (General Switch Management Protocol Version 3), an open interface protocol with resource management functions for efficient IP service on ATM over MPLS, is proposed and designed for VLSI implementation. And we compare and analyze the proposed GSMP with the GSMP under standardization process in terms of CLR (Cell Loss Rate). We design the Slave block of the Dynamic GSMP V3 using SAM-SUNG SoG 0.5μm process, which performs functions for switch connection control in the ATM Switch. In order to compare difference performanaces between the proposed method and the conventional one, we conducts simulations using the minimum buffer search algorithm with random cell generation. The exponential results show that the proposed method leads to performance enhancement in CLR.

키워드 : GSMP, VLSI, 동적버퍼관리(Buffer management), ATM, MPLS

1. 서 론

최근 인터넷 수요의 증가로 통신망에서 음성, 영상 및 데이터를 복합적으로 지원할 수 있는 멀티서비스의 교환기술의 요구 및 인터넷 트래픽의 증가에 따른 새로운 서비스의 요구를 수용하기 위해, 현재의 인터넷을 확장한 새로운 인터넷 백본망을 구축할 필요성이 요구되고 있다. 이에 따라 IETF(Internet Engineering Task Force)와 ATM 포럼을 중심으로 MPLS (Multi-Protocol Label Switching)라는 새로운 스위칭 기술을 개발 중에 있다. 특히 MPLS 망에서는 ATM의 서비스 품질(QoS : Quality of Service)의 보장 및 트래픽 관리 능력을 이용하여 ATM 망에서 IP(Internet Protocol) 서비스를 효율적으로 제공하기 위해 개방형 구조의 인터페이스를 사용한다[1, 2]. 현재 통신인프라는 하나의 서비스 제어 방식과 운용 방식의 틀로 멀티서비스를 수용하는 방식으로 서비스 별로 서로 상이한 서비스 제어 방식과 관리 방식의 특성을 그대로 반영할 수 없다. 그러나 개방형 멀티서비스 교환 기술은 통신망 사업자로 하여금 가격 및 제품 경쟁력이 있는 여러 벤더들의 제품으로 망 구축을 가능하게 하며, 필요에 따라서 망 장비 일부 기능의 추가, 교체, 기능 향상을 가능하게 한다.

GSMP 프로토콜은 1997년 Ipsilon사에서 개발된 개 방형 구조의 범용 프로토콜로서, 스위치 레벨의 접속을 설정, 해제 및 관리하는 인터페이스로 사용된다. 개방형 멀티서비스 교환 시스템에서는 개방형 개념에 따라 스위치와 제어기 사이의 표준 인터페이스(GSMP, VSI)를 사용함으로서 스위치와 제어기의 독립적인 개발이 가능하도록 한다.

ATM 기반 MPLS 구현을 위해 IETF에서는 직접 ATM 스위치에 IP 서비스를 수용하여 QoS 보장형 인터넷 서비스

※ 이 논문은 정보통신부 지원 리눅스 시스템 보안 연구센터의 지원을 받아 연구되었음.
† 정 회 원 : 전남대학교 전자공학과 교수
†† 준 회 원 : 전남대학교 대학원 전자공학과
††† 정 회 원 : ETRI 네트워크 기술연구소 인터넷기술연구원
†††† 정 회 원 : 동신대학교 전기전자공학부 교수
논문접수 : 2001년 3월 5일, 심사완료 : 2001년 5월 26일

를 제공할 수 있는 연구를 진행하고 있으며, ATM MPLS 라우터에서 IP 서비스와 ATM 서비스를 수용하기 위해 GSMP WG에서는 기존의 각 벤더들이 만든 GSMP 인터페이스를 표준화시키는 작업 및 QoS와 여러 서비스를 수용하기 위한 구조를 연구 중에 있다[3-6]. 국내에서는 정부 주도의 차세대 정보통신 사업의 일환으로 MPLS 방식을 채택 후 ETRI를 중심으로 각 기업체에서 MPLS망에서 인터넷 서비스를 효율적으로 전송하기 위해 전달망의 핵심인 ATM 교환기를 통해 인터넷 서비스를 전달하기 위한 구조를 개발하고 있다. 또한 ATM 포트가 두 가지 서비스(IP, ATM)를 위해 자원(VCI/VPI, BandWidth, Buffer, Queue, etc)을 효율적으로 공유해야 하는데, 현재 IETF GSMP WG에서는 qGSMP가 지원하는 버퍼관리 정책에 대한 고려가 되지 않아, 망 상황에 따라 버퍼의 사용 범위가 넘게 되면 서비스를 제공할 수 없는 문제점이 발생한다. 따라서 각각의 서비스를 효율적으로 전송하기 위해서는 동적 버퍼상태에 따른 채널 설정제어가 요구된다.

따라서 본 논문에서는 qGSMP에서 지원하는 임계치를 이용한 동적 버퍼관리 방식을 현재 표준화중인 GSMP V3 프로토콜에 적용함으로서 IP측면이나 ATM측면에서 버퍼를 효율적으로 관리하여, 버퍼상태에 따라 채널을 설정할 수 있게 시스템을 구현하였다. 버퍼 이용률을 버퍼관리기(Buffer Manager)가 관리하여 특정 버퍼가 셀을 수용할 수 없을 때 최소 버퍼 경로를 찾아 셀전송을 함으로서 버퍼 이용률을 높이고, 셀손실률을 개선할수 있는 Dynamic GSMP V3 알고리즘을 제안하여 현재 표준화중인 방식과 비교 분석하였다. 또한 고속 인터페이스 기능 및 GSMP 성능 향상을 위해 Dynamic GSMP V3에 대한 하드웨어 구조를 제안하였으며, IP 서비스에 대한 멀티캐스트 연결과 하드웨어 기반의 고속 패킷 전송이 가능하도록 설계하였다[7].

본 논문의 구성은 다음과 같다. Ⅱ장에서는 MPLS 개요와 GSMP를 이용한 MPLS 방식에 대해서 설명하고, Ⅲ장에서는 제안한 GSMP V3 알고리즘 및 특징을 설명하고, Ⅳ장에서는 C++를 이용한 Dynamic GSMP V3의 성능 평가를 위한 시뮬레이션 및 분석을 수행하였으며, 하드웨어 시뮬레이션 및 합성 결과를 고찰하였다. 마지막으로 Ⅴ장에서는 결론을 내린다.

2. MPLS 기술개요와 GSMP 알고리즘

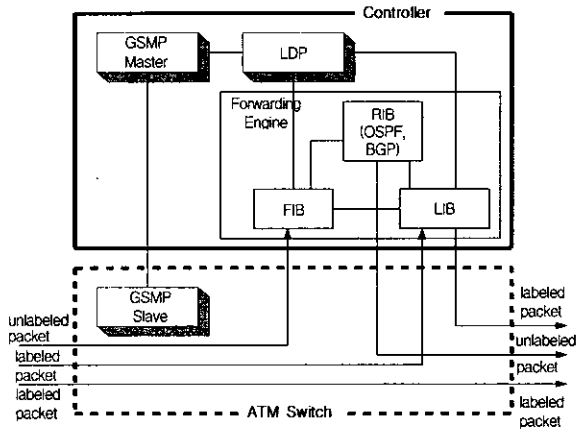
2.1 MPLS 기술 개요

MPLS의 패킷 전송(Forwarding) 방식은 레이블 교체(Label Swapping) 전송방식이라고 할 수 있다. 입구 레이블 스위치 라우터에 패킷이 도착하게 되면, 패킷의 목적지 주소나 레이블 매핑 정책에 의하여 패킷에 레이블을 붙이게 된다. 이렇게 레이블이 붙여진 패킷은 다음 홉으로 전송되고 각 레이블 스위치 라우터에서는 패킷에 실려온 레이블을 Index로 사용하여 다음 홉과 새로운 레이블을 결정하고 패킷의 레이블을 새로운 레이블로 교체한 후 다음 홉으로 전송하게 된다. 따라서 기존의 IP 라우팅 방식에서처럼 각 패킷의 목적지 주소에 따라 오랜 시간이 걸리는 테이블을 찾지 않아도 됨으로서 각 홉에서 패킷을 라우팅 하는데 걸리는 시간을 절약할 수 있다. 또한 ATM 망에서는 VPI/VCI 값을 레이블로 사용하기 때문에 IP 라우팅 판단을 위하여 필요한 각 라우터에서 ATM 셀을 재조립하여 IP 패킷으로 만들 필요가 없다. 따라서 MPLS를 사용하면 종단간 ATM 연결설정 절차 없이 ATM에서 높은 전송 처리율을 얻을 수 있다[8].

2.2 ATM 기반 MPLS 구조 및 전송방식

ATM망을 이용한 인터넷 패킷 전송에 관한 연구는 오래 전부터 되어왔다. 1997년까지는 대용량 트래픽을 처리할 수 있는 라우터가 없었기 때문에 대규모 인터넷 사업자들은 ATM을 코어로 하여 라우터가 에지에 위치하는 형태의 인터넷 백본망을 구축해 왔다. ATM 기반의 인터넷 백본망은 에지 라우터간을 ATM VC (Virtual Channel)로 연결하며 각 VC를 통과하는 트래픽을 감시하고 대역폭 및 경로를 제어함으로서 최상의 망 성능을 발휘할 수 있도록 하는 트래픽 엔지니어링이 용이한 구조를 가지고 있다. 따라서 망 차원의 트래픽 감시가 용이하고 운용자 제어 성이 뛰어나므로 라우터 기반의 백본망에 비해 최적의 망 성능을 발휘할 수 있는 망 환경을 제공해 준다. ATM 환경에서 MPLS 도메인이 다른 MPLS 도메인과 계층적 구조를 가지지 않으면 MPLS 도메인 내에 있는 LSR들은 3가지 경우에 대해 패킷을 포워딩 할 수 있다. 먼저 레이블이 붙은 패킷을 포워딩 할 수 있다[9, 10].

MPLS 도메인으로 들어오는 트래픽의 FEC를 분류한 후, 레이블을 트래픽에 추가시켜 포워딩 한다. MPLS 도메인을 떠나는 트래픽에 대해서는 포함된 레이블 정보를 제거하여 L3 패킷을 만든 후 포워딩 한다. 이런 능력을 가진 LSR 라우터의 모델은 (그림 1)과 같다.

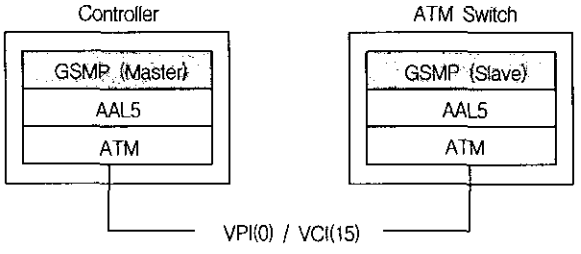


(그림 1) ATM 환경에서 MPLS 라우터 구조

LSR은 크게 제어기와 ATM 스위치로 구분된다. 제어기는 LDP와 포워딩 엔진, GSMP로 구성되어 있으며 ATM 환경에서 레이블은 ATM 셀 헤더의 VPI/VCI 값으로 의미를 부여한다. 전송방식은 기존의 인터넷 프로토콜인 OSPF나 BGP를 이용해 이웃해 있는 라우터들의 정보를 RIB에 저장한 후 레이블이 없는 패킷이 MPLS 라우터에 입력되면 라우팅 프로토콜에 의해 설정된 FIB(Forwarding Information Base)를 참조해 포워딩을 해서 전송하거나 LIB를 이용해 다음 홉으로 가는 레이블을 붙여준다. 레이블이 붙은 패킷이 들어오면 레이블을 삭제하거나 다음 홉으로 가는 레이블을 교환하여 전송한다.

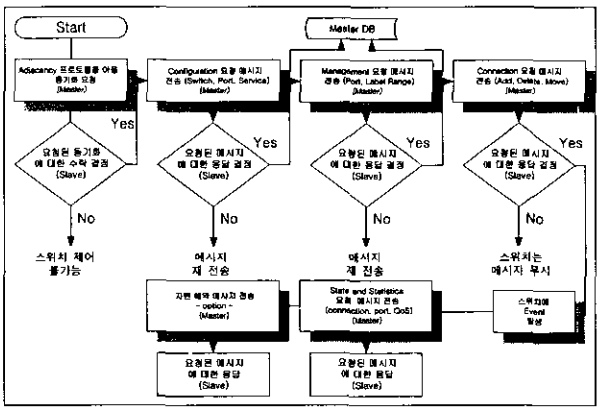
2.3 GSMP V3 프로토콜 구조

GSMP는 스위치 외부의 제어기가 ATM 스위치를 제어할 수 있도록 하기 위한 범용 프로토콜이다. GSMP V3는 현재 IETF에서 마지막 표준화 작업을 진행하고 있는 프로토콜로서 GSMP v1.1이나 v2.0에서 지원되지 않는 QoS보장이 가능하며, 표준화가 완료시 개방형 스위치 구조의 인터페이스로 적용된다. 기존의 ATM 스위치에서는 가상 채널의 설정과 해제를 위해 ATM 시그널링 프로토콜(Q.2931)을 이용하여 설정하였다. 하지만 MPLS 에서는 LDP(Label Distribution Protocol)를 이용하여 ATM 환경에서의 레이블인 VPI/VCI을 부여하고 분배하게 되며 이렇게 할당된 VPI/VCI를 이용하여 ATM 스위치내의 GSMP를 이용해서 가상 채널을 설정하고 해제한다[11].



(그림 2) GSMP 기본 프로토콜 구조

GSMP 프로토콜은 제어기가 Master, 스위치가 Slave로 동작하는 비대칭형 프로토콜이다. GSMP V3는 제어기가 스위치를 통해 연결을 설립, 해제와 멀티캐스트 연결에서 추가 및 삭제, 스위치 포트관리, 구성 정보요청, 통계 정보를 요청하게 한다. 또한 스위치는 링크 오류와 같은 비동기 적인 사건들을 제어기에게 알린다. 그래서 제어기는 항상 스위치에 요구 메시지를 보내게 된다. GSMP의 구조는 (그림 2)와 같다. 하나의 제어기는 여러 스위치를 제어할 수 있고, 또한 하나의 스위치는 분할 기법을 이용해 하나 이상의 제어기에 의해 제어될 수 있다. MPLS 라우터에서 GSMP 프로토콜은 제어부와 스위치부 사이에서 동작한다. GSMP를 이용하면 교환 시스템의 제어부와 스위치부는 상호 독립적인 개발이 가능하게 되어, 개방형 멀티서비스 교환 시스템을 구축할 수 있다. 하나의 스위치는 다수의 레이블 타입을 지원한다. 주어진 포트에 의해 지원된 레이블 타입은 포트 구성 메시지에서 스위치에 의해 제어기에 나타내어진다.



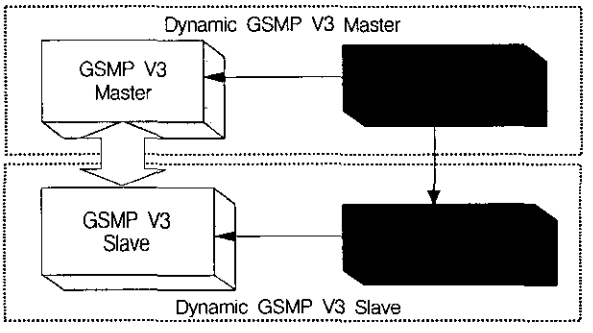
(그림 3) GSMP V3의 동작 개요도

레이블 타입은 ATM, Frame Relay, MPLS Generic Label를 포함한다. GSMP V3는 Point-to-point와 Point-to-multipoint 연결을 지원한다. GSMP V3는 디폴트 QoS 구성과 추가적으로 다른 방법의 협정과 Optional QoS 구성을 허가한다. GSMP 기본 동작개요도는 (그림 3)과 같다.

3. Dynamic GSMP V3 알고리즘 구조

3.1 Dynamic GSMP V3 모델링 구조

현재 GSMP에서는 qGSMP와 같이 버퍼관리 정책을 지원하지 않는다. 그러나 ATM 스위치 링크가 IP와 ATM 셀을 동시에 서비스할 때 효율적인 셀 전송을 위해서는 동적으로 버퍼를 관리할 수 있는 버퍼관리 정책이 요구된다.



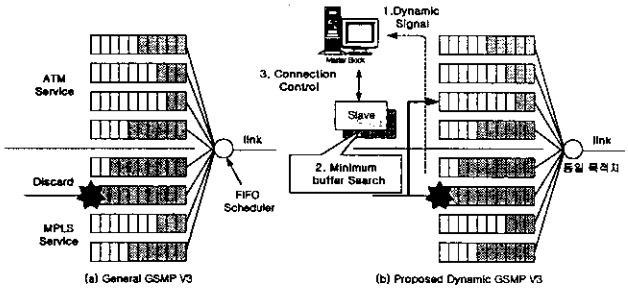
(그림 4) Dynamic GSMP V3 모델링 구조

따라서 본 논문에서는 버퍼관리 기능을 수행할 수 있는 GSMP V3 방식을 제안하였다. 제안한 방식의 기본 구조는 (그림 4)와 같다. 전체구조는 스위치를 제어하는 Master 블록과 Master 블록으로부터의 제어 명령을 처리하는 Slave 블록, 그리고 버퍼관리를 통해 셀 전송률을 증가시키는 버

퍼관리 블록, 그에 따른 연결관리 제어 블록으로 구성된다.

3.2 버퍼관리에 따른 채널 설정 방식 및 구조

본 연구에서 제안한 버퍼관리 알고리즘은 qGSMP에서 사용되는 방식을 적용하였다. 기본적으로 하나의 링크는 동일 목적지로 셀을 전송한다고 가정한다. 버퍼 관리 정책으로는 자신의 서비스를 최대한 효율적으로 보장하면서, 트래픽이 집중될 때 (그림 5)처럼 버퍼의 여유공간의 낭비 없이 서비스를 보장하는 임계치를 이용해 버퍼를 제어한다[12, 13].



(그림 5) 임계치를 이용한 동적 버퍼관리 모델

임계치 값 이하 일 때는 동적으로 버퍼 적용을 수행할 수 있도록 하여 다른 서비스에 대한 셀을 효율적으로 전송함으로서 QoS를 보장하고, 이상일 때는 고정적으로 버퍼를 제어함으로서 자신의 서비스에 대한 QoS를 보장하게 한다. 버퍼 관리 정책은 ATM과 MPLS 자원을 동일 포트에서 이용 시 ATM이나 MPLS 셀에 대한 버퍼 이용률이 폭주할 때 기존의 GSMP V3에서는 폭주된 버퍼로 들어오는 셀 들은 (그림 5) (a)에서처럼 계속 버리게 되지만, 제안한 구조에서는 스위치 제어기에게 새로운 경로 설정을 위한 제어 신호와 함께 Slave가 찾은 최소 버퍼 경로를 알려주면 스위치 제어기는 포트가 Connection Replace 메커니즘을 지원할 경우 이 메커니즘을 이용해 연결을 재 설정하고, 지원이 되지 않으면 Move Output branch 메시지를 이용해 (그림 5) (b)처럼 버퍼 용량이 가장 적은 버퍼로 셀을 전송한다.

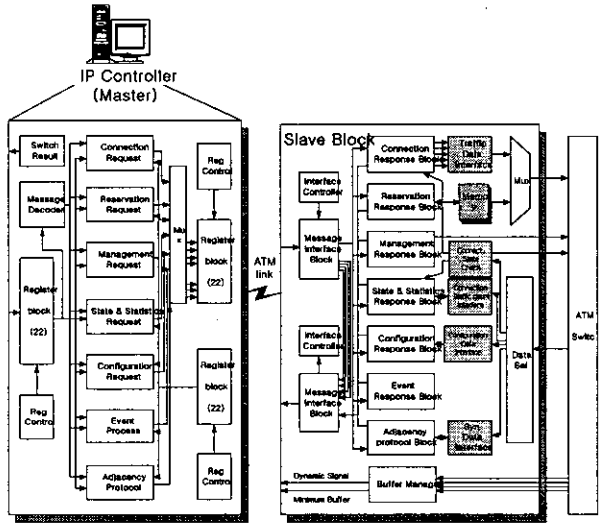
주요 동작은 위와 같다. 여기서 buffer_ threshold는 시뮬레이션을 통해 버퍼사이즈의 70%일 때 가장 좋은 결과를 나타내었다.

3.3 Dynamic GSMP V3의 하드웨어 구조

(그림 5)와 같은 버퍼관리 정책을 수용한 Dynamic GSMP V3 하드웨어 구조는 (그림 6)과 같다. Master 블록은 IP 제어기에서 Connection 정보를 GSMP 메시지 패킷으로 만들어 Slave에게 하나 이상의 연결을 설정할 수 있도록 해주는 기능을 수행한다. 또한 Slave 블록으로부터 입력되는 IP 셀을 전송하기 위한 ATM 스위치의 구성 정보나 기타 연결에 관련된 정보를 IP 제어기에게 알려주는 기능을 수행한다. 기본적인 구성은 GSMP V3에서 지원하는 크게 7개의 메시지를 처리하는 블록과 최대 22Byte 메시지 길이를 32bit 단위로 전송하기 위한 메시지 입. 출력 인터페이스 블록으로 구성되어 있다.

```
if (slot_number > buffer_threshold)
    Dynamic signal <= '1' ;
    /* Finding the Minimun Buffer */
else if (All slot_number > buffer_threshold)
    Static signal <= '1' ;
else
    Static signal <= '1' ;
end ;
```

주로 수행되는 기능은 메시지를 전송하기 위한 채널에 대한 동기화 설정 기능과 Connection 설정을 위한 연결 제어 기능을 주로 수행한다. 연결관리 메시지는 스위치 상에 연결을 설정, 해제, 수정하는 메시지로서 7개의 메시지 블록으로 구성된다. 각 메시지의 기능은 아래와 같다.



(그림 6) Dynamic GSMP V3 하드웨어 모델링 구조

① Add Branch : 기존의 Connection에 추가적인 연결을 설정하거나 새로운 연결을 설립하는 기능.
② Delete Tree : 메시지가 지정하는 연결설정 제거기 능.
③ Delete All Input and Output : 입출력 상에 연결을 모두 제거하는 기능, 입력되는 모든 셀 제거.
④ Delete Branch : 하나이상의 Branch를 제거하는 기 능.
⑤ Move Input and Output : 기존의 설정된 연결을 새로운 연결로 이동 설정하는 기능.

Reservation 메시지는 GSMP V3에 새롭게 추가된 기능으로서 연결을 설정하기 전에 스위치 자원들(대역폭, 버퍼, 큐, 레이블)에 대해 특정 연결 예약 기능을 수행하며, Management 메시지는 포트관리와 레이블 범위 관리의 메시지로 구성되며, 포트관리 메시지는 포트에 서비스를 가져오거나 서비스를 없애는 기능, Loop Back 기능, Reset 기능,

Transmit Data Rate 조정 기능을 수행한다. 또한 레이블 범위 메시지는 한 포트가 지원하는 레이블의 범위를 수정하는 기능을 수행한다. State와 Statistics 메시지는 스위치 입력과 출력포트 그리고 Connection과 관련된 다양한 트래픽 카운터의 값을 요청하는데 사용한다. Configuration 메시지는 스위치, 포트, 서비스 구성 메시지가 있으며, 스위치 구성 메시지는 망 사업자가 망에 맞는 최적의 스위치를 추가 시 스위치 정보를 컨트롤러에게 알리는 기능을 수행하고, 포트 구성 블록은 포트의 구성 정보를 아는데 사용한다. 또한 서비스 구성 블록은 포트나 Connection이 지원하는 서비스 정보를 알기 위해 사용한다. Event 메시지는 스위치가 비동기 적인 어떠한 사건(Port Up, Port Down, … etc)에 대해서 제어기에게 알리기 위해서 사용한다.

Slave 블록은 Master 블록으로부터 요청된 정보를 받아 먼저 요청한 메시지에 에러가 없는지 또는 요청 메시지를 수행할 수 있는지를 조사하여, 에러가 있거나, 요청 메시지를 수행할 수 없으면 에러 신호와 에러 코드를 포함한 에러 메시지를 Master에게 전송한다. 그렇지 않으면, 요청메시지의 요청을 연결된 스위치 상에서 수행을 한 후 그 결과를 Master 블록에게 알리는 기능을 수행한다. 메시지 인터페이스 블록은 Master 블록 내에 위치한 메시지 인터페이스 블록과 동일한 기능을 수행한다. Slave 블록내의 Connection 블록은 스위치 상에 연결을 설정하거나 해제하는 기능을 수행하며, Connection 데이터를 스위치에 8bit로 전송하기 위해 Traffic Data 인터페이스 블록이 필요하다. ATM 스위치 상에 하나의 Connection을 설정하기 위한 기본 단계는 다음과 같다.

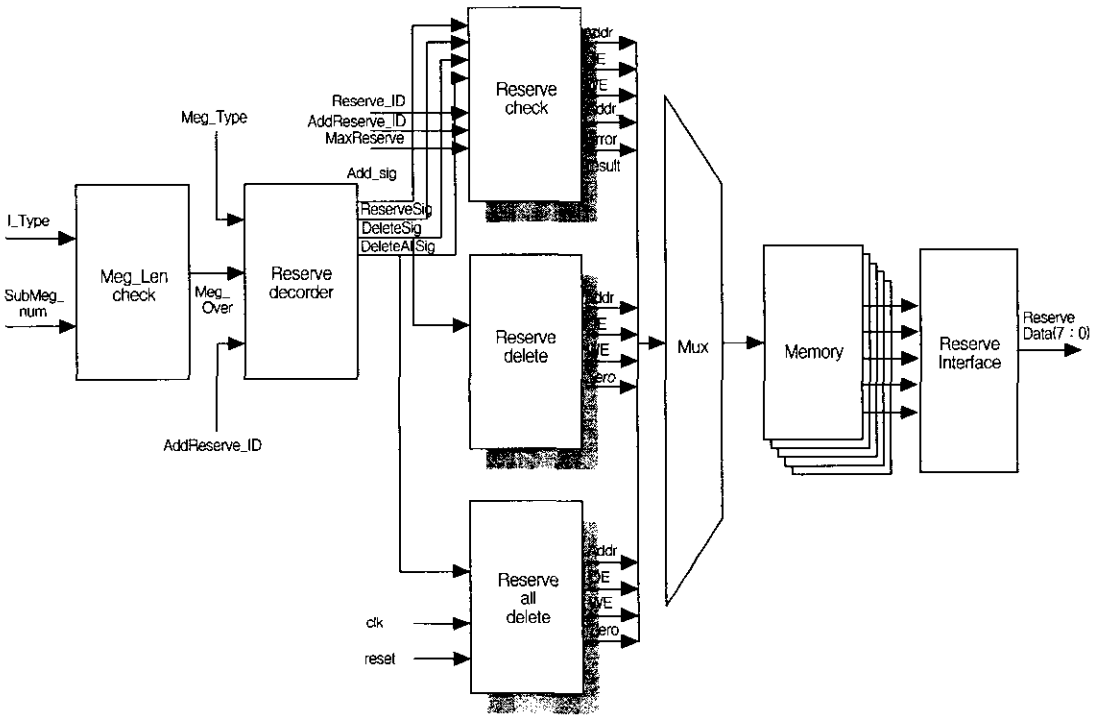
state 1 : (32bit Reservation ID = 0)
Connection Message의 요청에 따른 연결정보를 수행.

state 2 : (32bit Reservation ID /= 0 and
Reservation ID = Reservation memory data)
Reservation 메시지에 의해 예약된 연결정보를 수행.

state 3 : (Mux select (Reservation ID))
입.출력 포트나 레이블 및 트래픽 파라미터 정보를 선택해 스위치에 전달 (50byte).

(그림 7)과 Slave 블록 내에는 최대 10개의 자원을 예약할 수 있는 메모리가 내장되어 있다. Connect_State 블록은 요청메시지에 대한 Connection이 존재하는지 또는 Branch가 존재하는지를 검사하며, 최대 1024개까지 검사할 수 있다. 또한 하나의 연결에 대한 Branch 정보를 State and Statistics 블록에 알려줘 Connection에 연결된 Branch정보를 Report Connection State 메시지를 통해 Master 블록에게 알릴 수 있도록 한다.

하나의 Connection은 입력포트와 레이블 정보에 의해 설정되고, 하나의 Branch는 출력포트와 레이블 정보에 의해 지정된다. 또한 Adjacency 프로토콜은 링크상의 상태를 동기화 시키거나 사용하는 프로토콜의 버전이 일치되는지 아



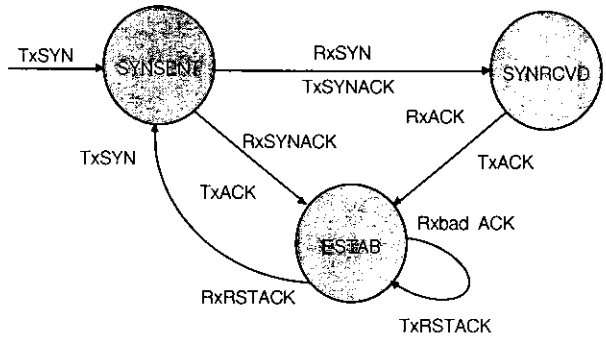
(그림 7) Slave내의 Reservation 블록 구조

는데 사용하거나 링크의 다른 쪽에 있는 Entity의 속성을 발견하는데 사용한다. 또한 Entity의 속성이 변하는 것을 검출하는데 사용한다.

GSMP는 Hard State 프로토콜이므로 스위치와 제어기 사이에 연결의 손실을 검출하는데 그리고 스위치나 제어기의 속성의 변화를 검출하는데 중요하다. 하드웨어 구조는 (그림 8)과 같다.

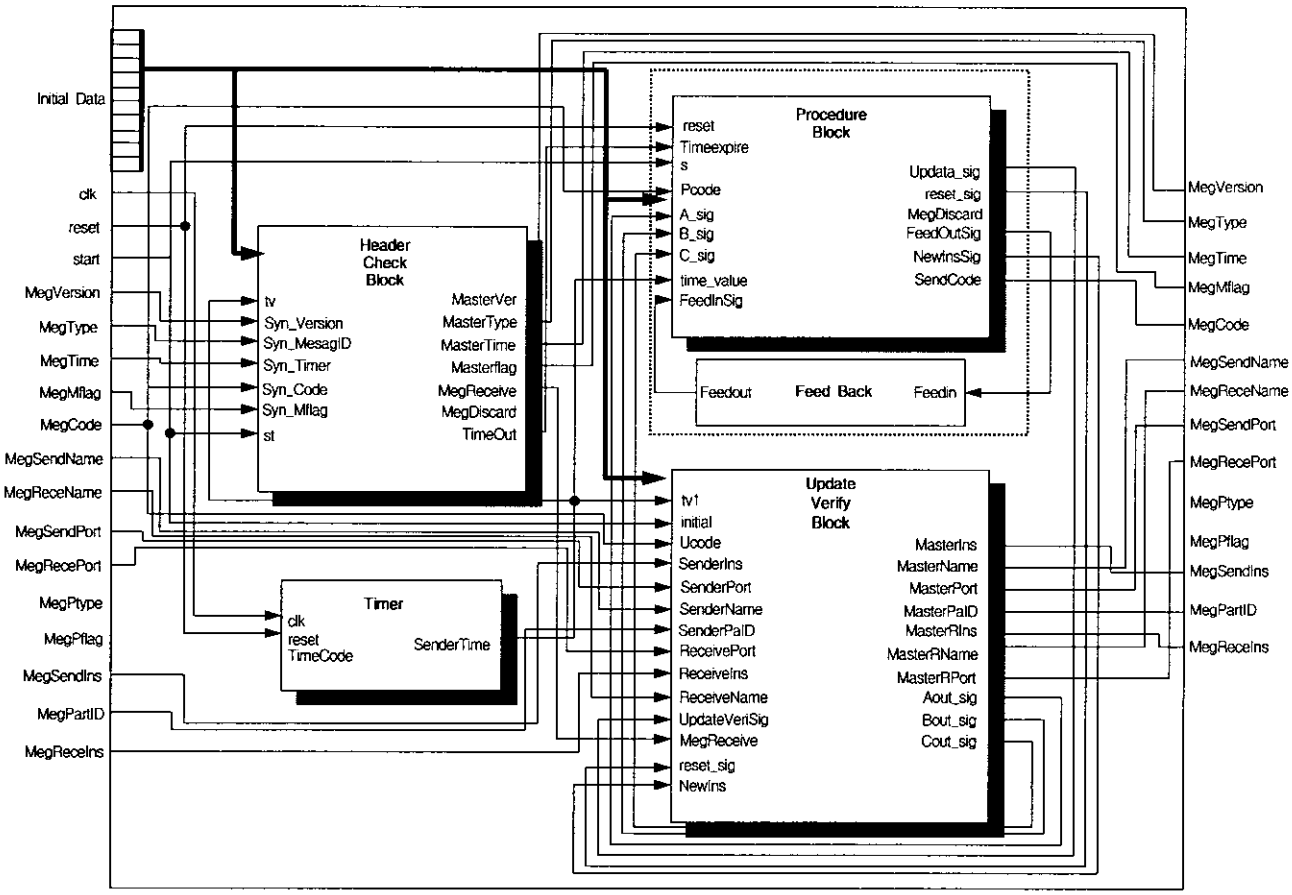
이 블록은 Slave를 제어하는 Master 블록과 스위치 포트를 제어하는 Slave 블록과 동기화를 설정하는 기능을 수행한다. 이 블록에서는 스위치 제어기의 정보와 동기화를 이룬 후 Slave의 정보를 저장하며, 주기적인 동기화 신호를 보낸다. 이 블록은 입력 메시지가 제대로 들어왔는지 확인하는 CheckSum 블록과 Slave 혹은 Master의 주기적으로 업데이트되는 정보를 저장하며 그에 따른 처리를 수행하는 Update 블록, 그리고 동기화 절차를 수행하는 Procedure 블록과 동기화가 설정된 후 주기적으로 동기화를 체크하기 위해 주기적인 신호를 보내기 위한 Timer 블록으로 구성되어 있다.

초기 상태의 State는 항상 "SYNSENT" 상태이다. 상태가 "ESTAB" 상태일 때 링크에 대한 동기화가 설정된다. 상태 변화는 (그림 9)와 같다.



(그림 9) Adjacency protocol 동기화 상태도

동기화를 위한 주요 동작은 위와 같다. 먼저 Master(IP 제어기)로부터 채널 설정을 위한 메시지가 입력될 때, 코드 상태가 "RSTACK"이면 채널을 재 설정하게 된다. 이때 A는 Master와 Slave간에 채널이 설정된 상태를 나타내며, C는 채널설정과 함께 서로 인접해 있는 Entity간에 정보를 알고 있는 상태를 나타낸다. 만약 채널 리셋을 요구하는 메시지가 입력되더라도, Slave블록과 채널 설정이나 서로의



(그림 8) Adjacency protocol 블록 구조

```
① Timer Expire : Reset Timer
    if state = SYNSENT send SYN
    if state = SYNRCVD send SYNACK
    if state = ESTAB send ACK

② Packet Arrives :
    if (incoming message = RSTACK)
      if (A && C && !SYNSENT)
        "Reset the Link"
      else
        "Discard the message"
      end if ;
    if (incoming message = SYN, SYNACK, ACK)
        "React according to the State Diagram"
    else if (any other GSMP message)
      if (state != ESTAB)
        "Discard incoming message"
      else if (state = SYNSENT)
        "Send SYN"
      else if (state = SYNRCVD)
        "Send SYNACK"
      end if ;
    end if ;
```

Entity정보를 모른다면 채널 리셋을 수행하지 못하게 된다. 여기서 SYN코드는 채널 동기화를 요청하는 신호가 되며, SYNACK는 채널 동기화 요청에 대한 응답신호가 된다. ACK신호는 주기적으로 채널 동기 상태를 체크할 때 사용되는 코드이다. 이런 코드를 가진 메시지가 입력되면 현재 상태와 조건에 따라 채널 설정에 관련된 처리를 수행하게 된다.

4. 성능 분석 및 시뮬레이션 결과

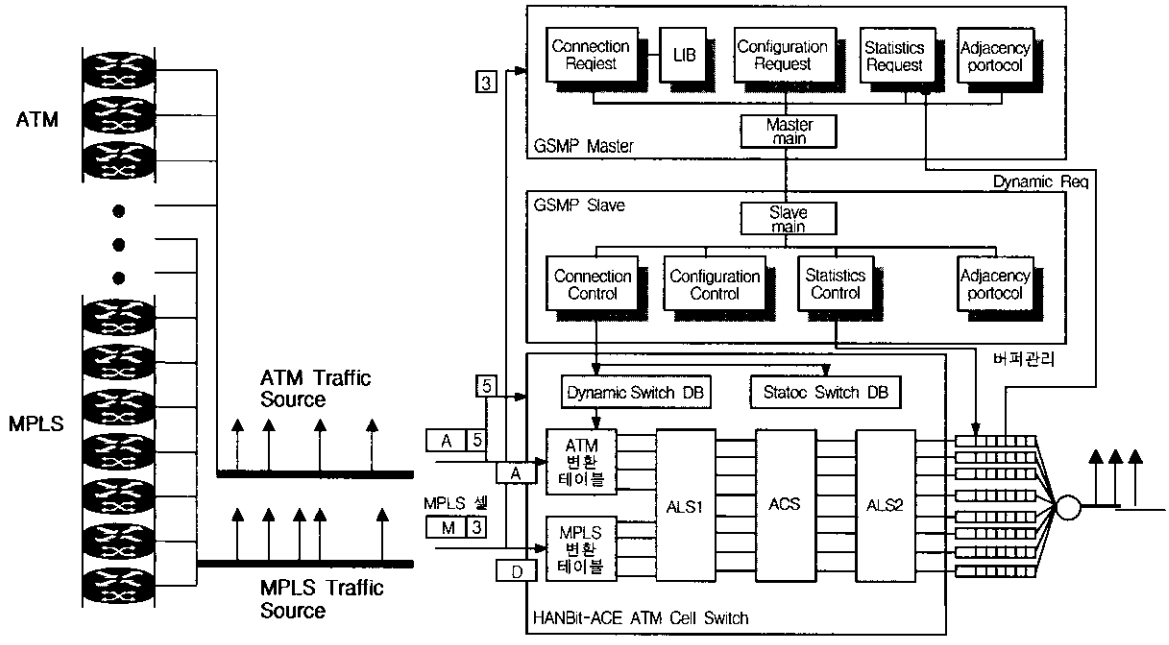
4.1 C++를 이용한 실험 및 성능 평가

본 논문에서는 하드웨어 구현에 앞서 제안된 버퍼 관리 알고리즘을 이용한 Dynamic GSMP V3의 모의 실험을 통하여 제안된 알고리즘의 성능 평가를 수행하고 개선된 알고리즘을 바탕으로 하드웨어 구현을 실시하였다. 본 모의 실험에서는 팬티엄 PC 환경 하에서 MicroSoft Visual C++ 6.0을 이용하였으며 IPP 트래픽 모델 및 On/Off 모델을 이용하여 시뮬레이션을 수행하였다.

〈표 1〉 GSMP의 트래픽 파라미터 조건

Class	Class 1		Class 2		Class 3	
Source	MPLS	ATM	MPLS	ATM	MPLS	ATM
지연 민감	○		○			
손실 민감			○		○	
Modelling	VBR(onoff)		VBR(IPP)		VBR(IPP)	
peak_rate	100 cells/s		200 cells/s		400 cells/s	
active_time	0.3 sec		0.1 sec		0.5 sec	
idle_time	0.9 sec		0.7 sec		0.4 sec	
버퍼사이즈	50					
Threshold	35 (버퍼의 70 %)					
Load	0.1 - 1.0					
링크 용량	2122 cells/sec(45Mbps)					

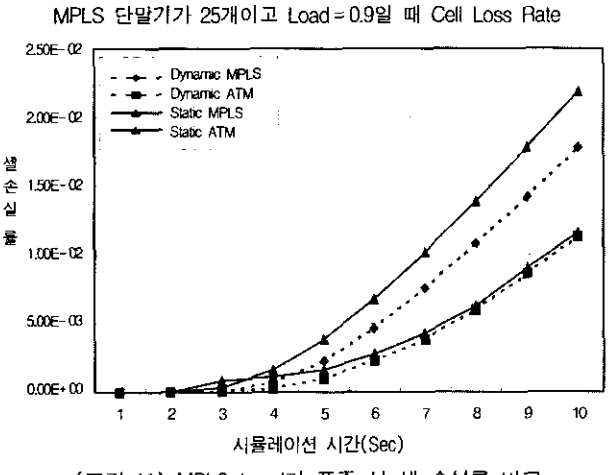
실험을 위해 각 케이스별로 ATM, MPLS 트래픽으로 분류하였으며, 트래픽 파라미터의 특성은 <표 1>과 같고, 모델링 구조는 (그림 10)과 같다. 총 50개의 단말기로 구성하였으며, 각각 25개씩 나누어 MPLS와 ATM 트래픽 모델로 가정



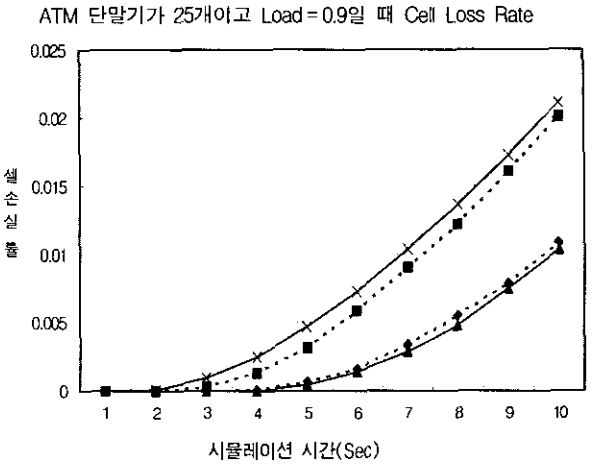
(그림 10) Dynamic GSMP V3 테스트 모델 구조

하였다. 성능 비교와 실험방법의 공정성을 위해서 GSMP V3와 Dynamic GSMP V3도 동일한 트래픽 조건과 임계치를 갖는 상태에서 각각의 로드(Load)값을 변화시켜가면서 비교 실험하였다. 버퍼사이즈나 임계치는 VBR 트래픽 특성에 맞게 시뮬레이션 결과를 통해 설정하였다. MPLS 서비스 폭주시 MPLS 트래픽의 셀전송 효율을 측정하기 위해 MPLS 트래픽에 대한 Load값을 높게 설정하고 ATM 트래픽에 대한 Load는 낮게 설정하여 시뮬레이션을 수행하였으며, ATM 서비스 폭주시 성능 평가를 위해 MPLS 트래픽에 대한 Load을 낮게 설정하고 ATM 트래픽에 대한 Load는 높게 설정하여 시뮬레이션을 수행하였다. 그 결과는 (그림 11)와 (그림 12)는 같다.

(그림 11)에서는 MPLS 셀이 MPLS 버퍼 쪽에 집중될 때



(그림 11) MPLS Load가 폭주 시 셀 손실률 비교



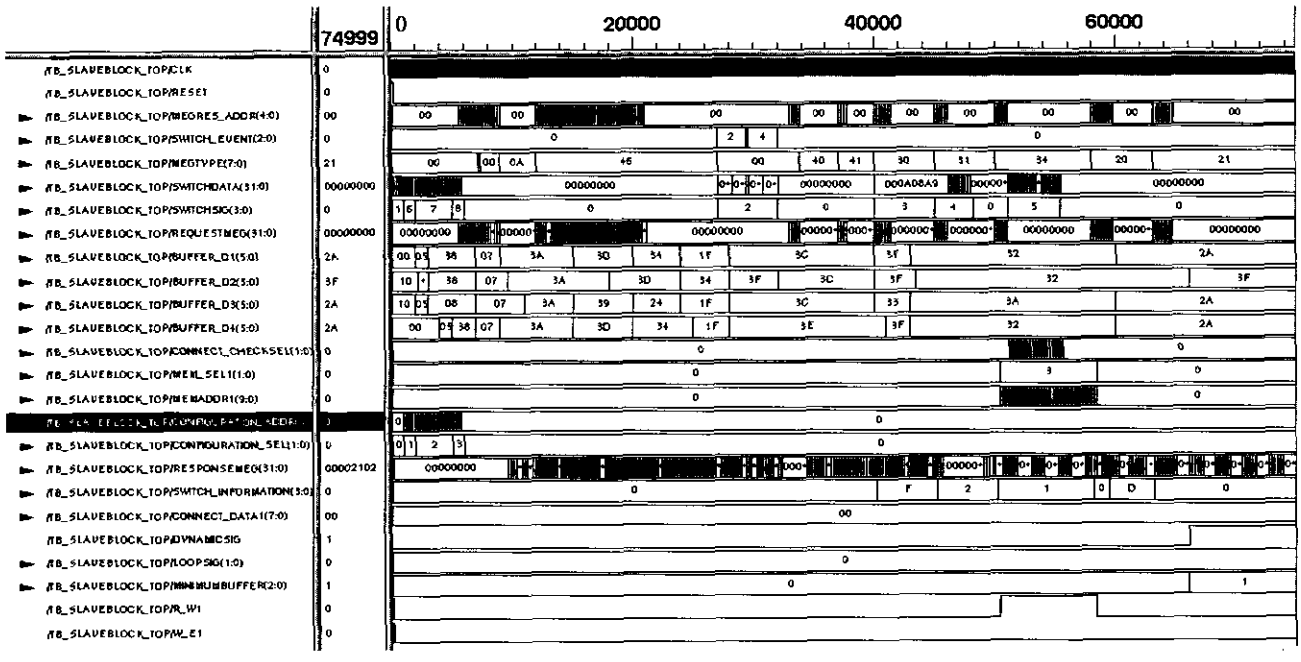
(그림 12) ATM Load가 폭주 시 셀 손실률 비교

버퍼 관리 방식을 사용하지 않은 기존의 GSMP(Static 방식)와 제안한 동적 분산 버퍼관리 방식을 사용한 GSMP(Dynamic 방식)를 비교한 결과 ATM의 셀 손실률에서는 거의 차이가 나지 않고 MPLS의 셀 손실률이 향상되는 것을 볼 수 있다. 이는 ATM 서비스의 QoS를 만족하면서 IP 서비스의 전송률이 향상됨을 알 수 있다.

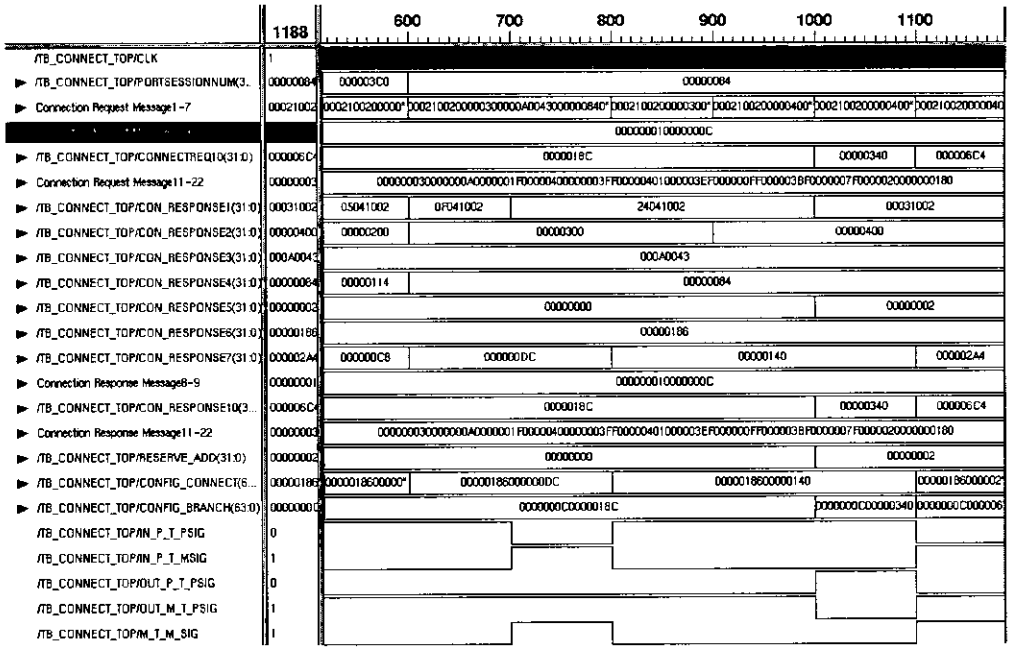
(그림 12)에서는 ATM 셀이 버퍼에 폭주할 경우의 시뮬레이션 결과로서 MPLS 서비스의 QoS를 보장하면서 ATM 셀에 대한 손실률을 향상시키는 것을 확인할 수 있다.

4.2 하드웨어 시뮬레이션 및 합성 결과

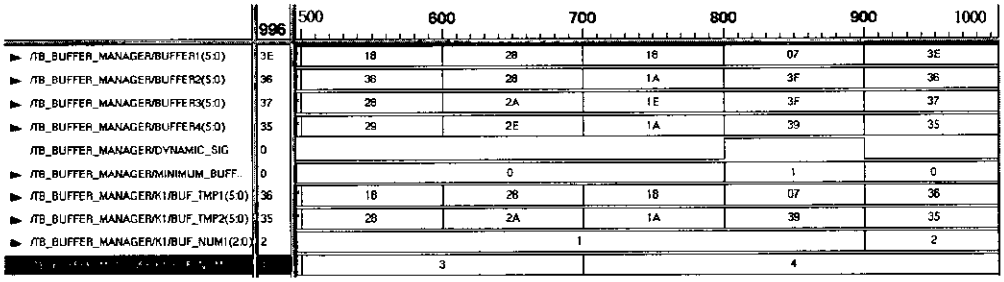
본 연구에서 제안한 Dynamic GSMP V3를 C++를 통한 성능 평가를 바탕으로 제안한 하드웨어 구조 모델을 설계하고, Synopsys 툴을 이용하여 VHDL 모델링한 후 이를 시뮬레이



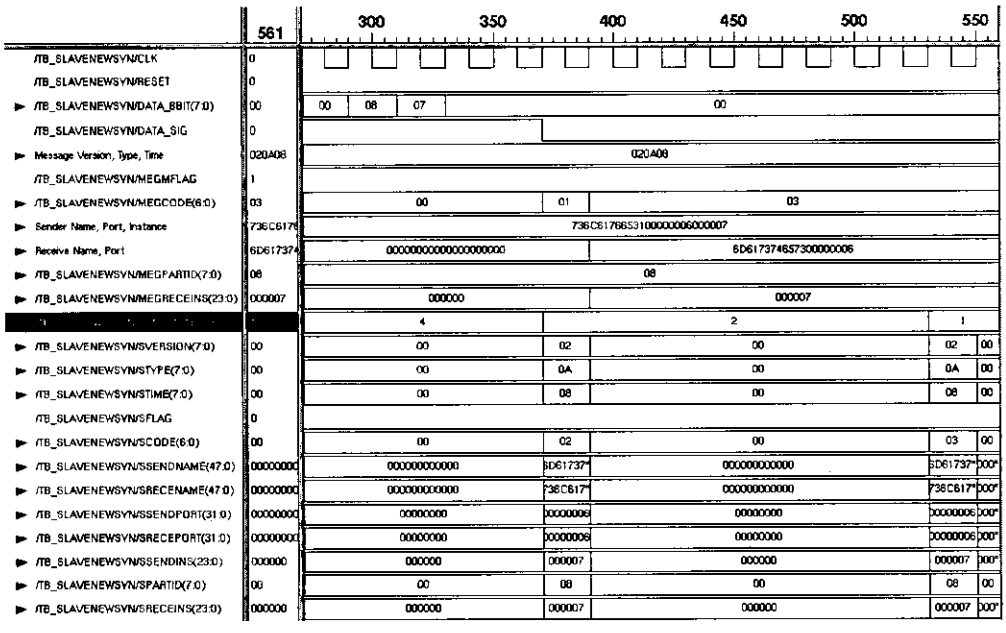
(그림 13) Top Slave 블록 시뮬레이션 결과



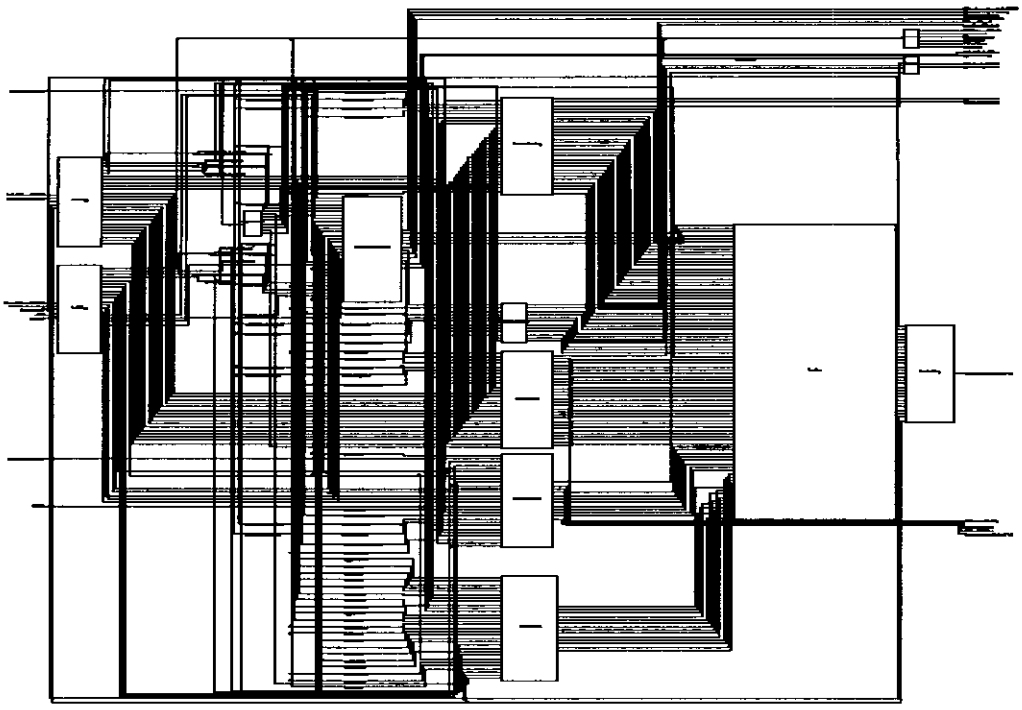
(그림 14) Connection Response 블록 시뮬레이션 결과



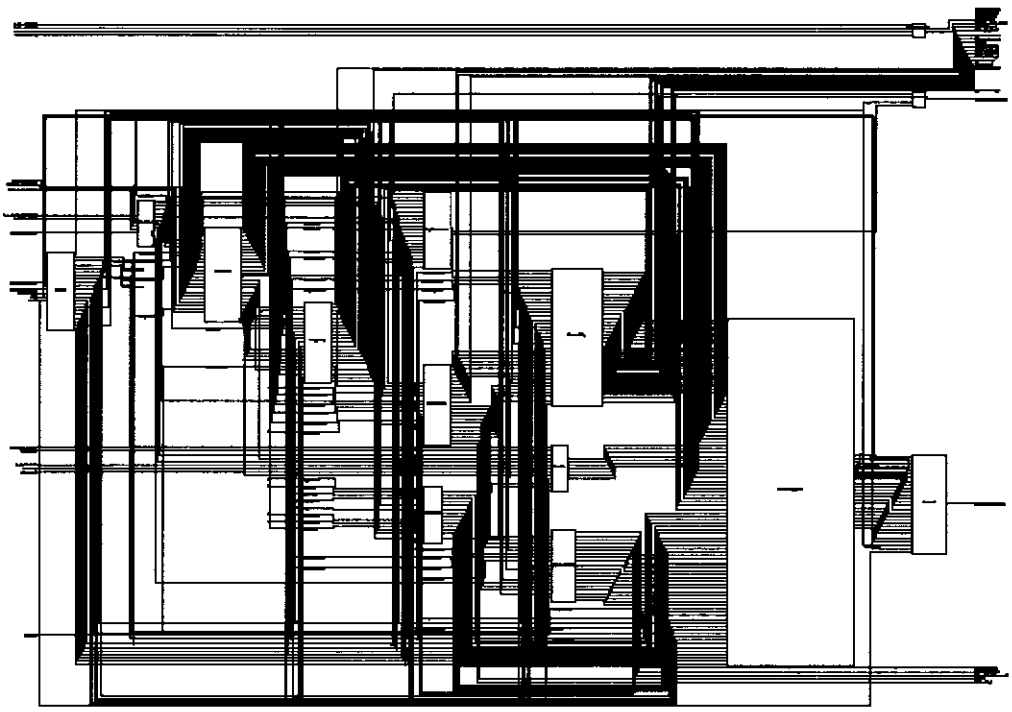
(그림 15) Buffer Management 블록 시뮬레이션 결과



(그림 16) Adjacency Protocol 시뮬레이션 결과



(그림 17) Master 블록 합성 결과



(그림 18) Slave 블록 합성 결과

션을 통해 성능을 검증하고 합성하였다. 각 하위 블록들은 동작적 모델링으로 기술하고, 상위 수준에서는 구조적으로 기술하였다. 구조는 Master 블록 Slave 블록으로 나누어 구현하였으며, 주요 블록들이 하드웨어적으로 제대로 동작하는지의 여부를 검증하였다. 설계는 삼성 KG80 library를 이용하였으며, SADAS 3.2.5 version 툴을 이용하여 레이아웃을 수행하였다. Package는 Copper_208QFP를 사용하였다. Slave 블록의 전체블록 시뮬레이션 결과는 (그림 13)과 같다. 처음 0ns에서부터 100ns까지는 Slave의 동기 데이터를 Adjacency 블록에 입력된다. 그 후에 Slave에 장착된 스위치의 구성을 알기위

해 Switch, 포트, 서비스 각각의 주소에 의해 32비트 해당 데이터가 입력된다. (그림 13)에서 SwitchSig 신호는 입력되는 스위치 데이터가 어떤 데이터인지를 구분해 주는 신호이다. 각각의 데이터가 초기 데이터가 저장된 후 Master로부터 입력되는 동기화 메시지에 따라 채널 상태가 "4(SYNSENT)"에서 "2(SYNRCVD)"에서 "1(ESTAB)" 상태로 되는것을 볼 수 있다. 또한 채널 상태가 ESTAB 상태일 때 동기 메시지를 제외한 다른 메시지가 입력되면 출력되는 결과를 확인할 수 있다. 또한 입력되는 메시지에 따라 해당 신호들이 출력되는 것을 나타내고 있다. 또한 4개의 버퍼에서 입력되는 용량을 비교하여 특정 버퍼의 용량이 버퍼 용량을 초과시 DynamicSig 신호가 1로 설정되는 것을 나타낸다. 본 시뮬레이션 결과를 통해 트래픽 발생이 불규칙적이고 실시간을 요하는 VBR 트래픽의 경우에는 버퍼에서의 셀손실이 예상된다. 그러나 버퍼를 동적으로 제어, 관리함으로서 셀 손실을 줄일 수 있었다.

(그림 14)는 Connection Response 시뮬레이션 결과를 나타낸다. Connect_Signal 신호를 바탕으로 요청한 포트나 레이블이 있는지 조사하여, 만약 지정된 연결이 없으면 "0003"이 아닌 해당 에러 코드를 Con_Response 데이터를 통해 출력되는 것을 보여준다. 또한 입력되는 메시지가 P-T-P인지 P-T-M인지 M-T-M인지를 조사하여 스위치에게 신호로 알려 주는 것을 볼 수 있다. 또한 5700 ns 지점에서 Connection 요청에 따라 8비트의 데이터 버스를 통해 연결 정보를 출력하는 것을 볼 수 있다.

(그림 15)는 버퍼 관리 블록의 시뮬레이션 결과를 나타낸다. 4개의 버퍼로부터 버퍼사용 용량이 입력될 때 어떤 특정 버퍼가 꽉 차면 버퍼관리 알고리즘을 적용하여 Dynamic 신호와 함께 최소 버퍼경로를 나타내는 것을 볼 수 있다.

그러나 다른 나머지 버퍼도 만약 버퍼용량의 70%이상이면 동적 신호를 발생하지 않는다. (그림 16)은 동기화 블록의 시뮬레이션 결과를 보여준다. 먼저 Master 블록으로부터 동기화를 요청하는 코드 "01(SYN)"이 입력되면 Master의 정보를 Slave의 메모리에 저장하고, Slave의 정보와 함께 "02(SYNACK)"로 응답되는 것을 볼 수 있다.

그 다음 "03(ACK)" 신호가 Master로부터 입력되면 입력되는 메시지내의 Slave 정보가 맞는지 확인한 후 "03(ACK)" 신호와 함께 메시지를 출력하는 것을 보여준다. 또한 동기화가 설정된 이후에는 Slave 블록의 동기화 주기마다 메시지를 출력하는 것을 볼 수 있다.

제안한 Dynamic GSMP V3는 삼성 KG-80 라이브러리를 이용해 합성하였을 때, Master 블록의 경우 총 76,000개 정도의 게이트가 소요되었으며, Slave 블록은 약 135,430개의 게이트가 사용되었으며, Kg8324d_t Master 타입을 사용하였다. 입력 Risc Slope과 Fall Slope는 각각 2ns이며, 출력 Load Capacitance는 75pF이다.

(그림 17)은 Master 블록의 합성을 나타내고, (그림 18)은 Slave 블록의 합성 결과를 나타낸다.

5. 결 론

본 논문에서는 MPLS 망에서 ATM 스위칭 방식을 백본망으로 사용할 경우 ATM 스위치 자원을 통해 IP 서비스를 제공하기 위한 3계층과 2계층의 인터페이스 기능을 수행하는 GSMP V3에 버퍼 관리 알고리즘을 추가한 Dynamic GSMP V3 프로토콜을 구현하였다. Dynamic GSMP V3는 현재 마지막 표준화가 진행중인 GSMP V3에서는 qGSMP가 지원하는 버퍼관리 정책은 제공하지 못하고 있다.

본 논문에서는 ATM 자원(버퍼)을 MPLS 서비스와 ATM 서비스가 공유 시 특정 서비스에 대한 폭주가 발생할 때 발생하는 셀 손실률을 최소 버퍼 알고리즘을 적용하여 버퍼 상태에 따라 채널을 재 설정함으로서, 효율적으로 2가지 서비스(ATM, IP)에 대한 전송 효율이 현재 표준화중인 방식보다 향상됨을 확인하였다. 또한 ATM 기반 MPLS 망에서 ATM 스위치에 고속으로 Connection을 제어를 수행하고, IP 제어기의 CPU 부담을 줄이기 위해 고속 인터페이스가 가능한 GSMP V3를 위한 하드웨어 모델을 제안하고, 구현함으로서 GSMP V3의 성능을 향상 할 수 있었다.

본 논문에서 제안된 Dynamic GSMP V3를 MPLS 망에서 사용할 경우 2가지 서비스에 대한 셀손실률이 보장되므로 다양한 트래픽을 효율적으로 처리할 수 있을 뿐만 아니라 개방형 구조의 GSMP V3 모듈을 통한 다양한 멀티미디어 서비스의 수용을 효율적으로 할 수 있을 것으로 사료된다.

참 고 문 헌

[1] IETF draft : A Framework for Multiprotocol Label Switching, May, 1998.

[2] IETF draft : Multiprotocol Label Switching Architecture, April, 1999.

[3] IETF draft : General Switch Management Protocol V3, August, 2000.

[4] IETF draft : GSMP Applicability, March, 2000.

[5] RFC 2397 : Ipsilon's General Switch Management Protocol Specification, March, 1998.

[6] ATM Forum Technical Committee : Traffic Management Specification Version 4.1, Feb, 1999.

[7] Bruce Davie, paul Doolan, "Switching in IP Networks," Morgan Kaufmann Publishers Inc, 1998.

[8] 한국전자통신연구원, "ATM상의 인터넷 서비스 기술개론", 전한도서, 1999.

[9] 이재섭, 손명희, 이현숙, "MPLS 기반의 ATM IP 교환기에서의 데이터 패치처리 기술", 한국 통신학회 추계종합학술발표회 논문집, 1998.

[10] 정연쾌, 최명철, 김부일, 이현숙, 이재섭. "ATM 교환기 기반 Label Switching Router 구조", 한국통신학회 추계종합학술발표회 논문집, 1998.

[11] P. Newman, T. Lyon, and G. Minshall. "IP Switching : ATM Under IP," IEEE/ACM Transactions on Networking.

[12] Human, J. M., Lazar, A. A. and Pacifici. G., "Real-Time Scheduling with Quality of Service Constraints," IEEE Journal on Selected Areas in Communications, Vol.9, September 1991, pp.1052-1063.

[13] S. G. Min, Y. K. Kim, "Design and Implementation of Multi-service Switch Resource Management Function for Multi-service ATM Switching System," IEICE ITCCSCC'99, July, 1999.

김 영 철

e-mail : yckim@chonnam.ac.kr
1981년 한양대학교 전자공학과 졸업 (공학사)
1987년 University of Detroit, EE, 공학석사
1993년 Michigan State University, EE, 공학박사
1993~현재 전남대학교 전자공학과 부교수
2000~현재 전남대학교 반도체설계교육센터 소장
관심분야 : 초고속통신망, 인터넷 응용, 회로설계, 보안 칩 개발

이 태 원

e-mail : twlee@neuron.chonnam.ac.kr
1993년 전남대학교 전자공학과 졸업(공학사)
1999년 전남대학교 전자공학과 공학석사
1999년~현재 전남대학교 전자공학과 박사과정
관심분야 : 초고속통신망, 회로설계

김 광 옥

e-mail : kwangok@etri.re.kr
1999년 조선대학교 정보통신공학과 졸업 (공학사)
2001년 전남대학교 전자공학과 공학석사
2001년~현재 ETRI 네트워크 기술 연구소 인터넷 기술 연구부 MPLS H/W팀 근무중
관심분야 : GSMP, MPLS, 인터넷 응용

이 명 옥

e-mail : mikelee@hichips.com
1981년 Arizona State University 졸업 (공학사)
1983년 Arizona State University 공학석사
1988년 Airzona State University 공학박사
1996년~현재 동신대학교 전기전자공학부 부교수
2000년~현재 (주)HiCHIPS 부사장 & 기술이사
관심분야 : 저전력, 회로설계, 보안, 인터넷 응용