

# 트래픽 관리를 지원하는 인터페이스 모듈 설계에 대한 연구

박노석, 손승일

한신대학교 정보통신학과

전화 : 031-370-6819 / 핸드폰 : 016-642-5273

## A study on Design of the Interface Module supporting the Traffic management

Noh-Sik Park, seong-il sonh

Dept. of Information and Communication , hanshin University

E-mail : ndock@hanmail.net

### Abstract

This paper has been studied a interworking signalling protocol between two hybrid networks by analyzing Satellite B-ISDN architecture, DSS2 Layer 3 Signalling protocol, B-ISUP protocol, S-BISUP protocol stack and so on. Also in the paper, messages and primitives have been defined for B-ISDN's Connection Type, Ownership and each protocol in order to connect point-to-multipoint.

And then this paper has designed basic call procedures for OBP Satellite B-ISDN NNI interworking protocol, verified and implemented them.

### I. 서론

ATM(Asynchronous Transfer Mode)은 데이터, 비디오, 음성등과 같은 다양한 트래픽을 전송하는 차세대 멀티미디어 모바일 통신 시스템인 IMT-2000에서 응용될 수 있을 것으로 예견되고 있는 분야이다. 이는 높은 대역폭과 다양한 유형의 QoS의 지원이 가능하기 때문에 ATM의 응용은 점점 증가할 것으로 예견되고 있다.[1] 네트워크의 다양한 서비스를 충족시키고 QoS의 지원을 위해서는 효율적인 저장 버퍼의 관리가 필요하다.[2,3]

본 논문은 트래픽 매니저 및 스위치 패브릭 사이에서 데이터를 읽고 쓰기 위한 인터페이스로서 8개의 클래스와 3종류의 패킷 유형(unicast multicast, broadcast)을 분류하여 각 채널별로 FIFO에 저장하는 역할을 수행하며 우선순위에 의한 FIFO 제어 및 데이터 시퀀스 정보를 확인하여 잘못된 데이터 삭제 및 소프트웨어 리셋 기능을 제공하는 트래픽 관리 인터페이스 모듈을 하드웨어 설계 언어로 설계한 후 Xilinx를 사용하여 시뮬레이션 하고자 한다.

### II. 인터페이스 모듈의 구조

트래픽 관리를 지원하는 인터페이스 모듈은 트래픽 매니저로부터 데이터 버스를 통해 32비트의 데이터를 수신하여 각 채널별로 패킷을 분류하여 FIFO에 저장하는 역할을 수행한다. 또한 내부의 흐름제어 정보를 기반으로 트래픽 매니저에 채널 요청 정보를 전송하는 기능을 수행한다. 채널 요청 정보는 버스 사이즈를 최소화 하기위해 1비트만을 사용하여 구현하였다. 8개의 클래스를 제공하고 있으며, 클래스 번호가 낮은 것이 더높은 우선권을 갖도록하였고, 추가적인 컨트롤 채널을 제공하여 내부의 상태정보를 액세스할 수 있는 기능을 제공한다. 트래픽 매니저는 주기적으로 컨트롤 채널을 통해 인터페이스 모듈의 내부 상태를 조사하여 필요한 조치를 취하게 되고 필요할 경우 loop back, 소프트웨어 리셋등의 기능을 수행한다. 그림1은 인터페이스

이스 모듈의 블록도를 보여주고 있다.

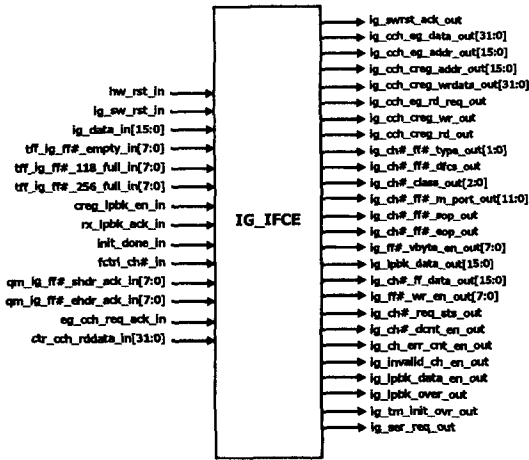


그림 1. 트래픽 관리를 지원하는 인터페이스 모듈의 블록도

### III. 내부 블록간의 인터페이스

트래픽 관리를 지원하는 인터페이스 모듈은 tm2tx블록, ser\_tm\_req블록, ctr\_ch블록, ig\_lpbk블록, nxt\_req\_gen블록, ext\_hdr\_req블록, base\_hdr\_reg 블록, fifo\_sel블록, nor\_route블록, fifo\_dat블록으로 구성되어 있다. 제안하는 블록의 내부 블록간 인터페이스는 그림 2와 같다.

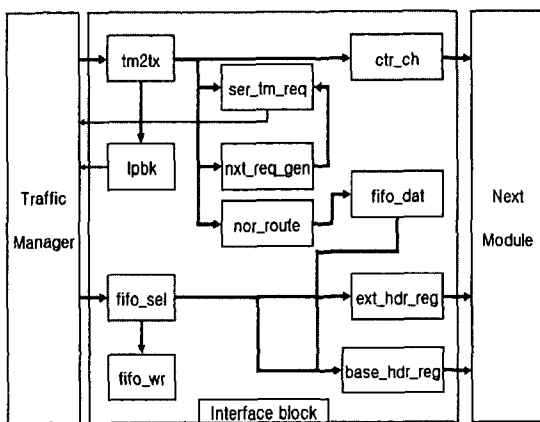


그림 2. 내부 블록간의 인터페이스

#### 3.1 tm2tx 블록

트래픽 매니저로부터 idle 셀을 제외한 유형의

tm\_cell을 수신하게 되면, 이 블록은 정상적인 데이터 인지 컨트롤 데이터인지의 여부에 따라 해당 블록에 수신한 tm\_cell을 전달하는 역할을 수행한다. 정상적인 데이터의 경우에는 nor\_route블록에서, 컨트롤 데이터인 경우에는 ctr\_ch블록에서 tm\_cell 데이터를 처리한다. 이 블록에서는 유효데이터의 수, sop(Start of packet), eop(End of packet) 및 패킷 시퀀스 에러 신호, 패킷 종류등의 신호를 디코딩하여 전달하는 역할을 수행하고, 최종적으로 셀 전송이 종료시 종료 신호를 생성한다. 수신한 데이터를 곧바로 수신단으로 라우팅할수 있는 loop back 모드를 지원하고, loop back 모드는 1 tm\_cell 동안만 수행한다. 그림 3은 tm2tx 블록의 내부구조를 보여주고 있다. 2개의 블록으로 구성되어 있다. tm2tx\_fsm블록은 정상적인 데이터 전송이나 컨트롤 레지스터 블록으로 데이터를 전달하기 위해 사용되고, tm2tx\_dat 블록은 tm2tx\_fsm 블록에 채널 번호와 tm\_cell 유형 정보를 제공하고, ser\_tm\_req 블록에 request 인에이블 신호를 제공한다.

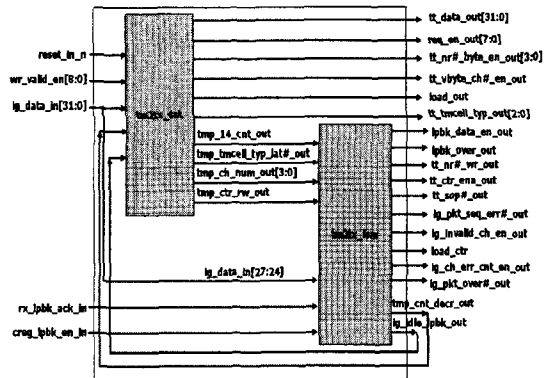


그림 3. tm2tx 블록의 내부 블록간 인터페이스

#### 3.2 nxt\_req\_gen블록

nxt\_req\_gen 블록은 이전의 request를 일단 서비스 하고 나면 FIFO full 및 패킷 종료 신호에 기초하여 각 채널 단위로 next request 인에이블 신호를 생성하는 블록이다. 트래픽 매니저와 데이터 트랜잭션은 패킷 단위로 이루어지기 때문에 동일 채널에 대해 하나의 완전한 패킷을 수신한 이후에 다른 패킷을 수신할 수 있다. 그리고, 하나의 완전한 패킷을 수신하였을 경우에 2개의 FIFO 중에서 최소 하나의 FIFO는 비어 있어야 다음 패킷의 수신을 위한 request 신호를 활성화 시킬수 있다. 이에 대한 전체 블록도와 임의의 채널에 대한 next request 신호 생성 회로를 그림 4와 그림 5에서 보여주고 있다.

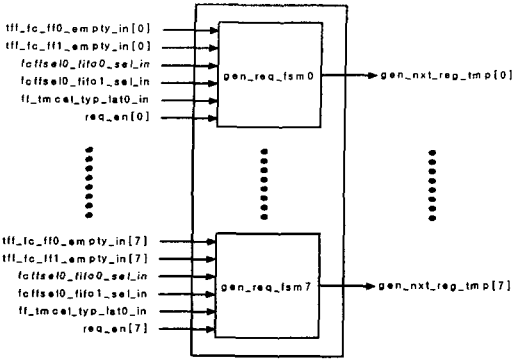


그림 4. next\_req\_gen 블록의 입출력 신호

### 3.4 ctr\_ch블록

컨트롤 채널은 트래픽 매니저가 next module에 명령을 내리거나, 상태 및 통계 레지스터를 액세스하기 위해 사용된다. 컨트롤 채널의 제어 블록은 ctr\_mod 블록과 ctr\_fsm 블록으로 구성된다. ctr\_fsm블록은 컨트롤 채널의 읽기 및 쓰기 신호를 생성하는 블록이며, ctr\_mod 블록은 컨트롤 채널 관련 신호를 수신하여 적절한 주소 및 데이터 값을 출력해 주는 블록이다. 그림 7은 ctr\_ch 블록에 대한 내부 블록간의 입출력 신호의 흐름을 보여주고 있다.

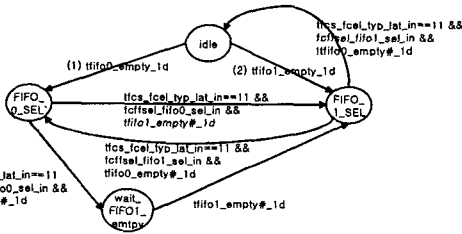


그림 5. next\_req\_fsm블록의 상세 회로도

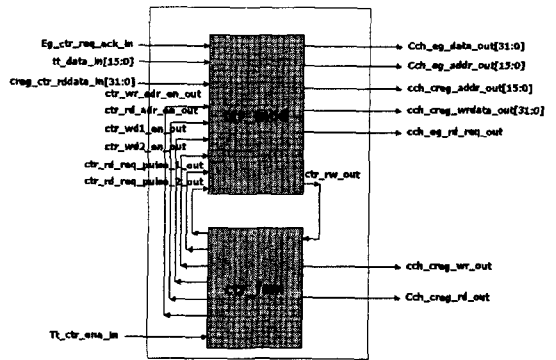


그림 7. ctr\_ch 블록의 모듈간 인터페이스

### 3.3 ser\_tm\_req블록

이블록은 트래픽 매니저에게 향후 수신 가능한 채널의 request 정보를 전송한다. request 신호의 전송 방식은 버스의 사이즈를 줄이기 위해 단일 라인으로 구성하였으며 1개의 채널 request 신호를 전송하는 데는 12 사이클이 소요된다. 트래픽 매니저에 채널 데이터 전송의 request를 보낼수 있는 조건은 해당 채널이 최소한 하나의 비어있는 FIFO가 존재해야하며, 스위치 패브릭으로부터 수신한 흐름제어 정보 신호가 활성화 되어있지 않아야 한다. 그림 6은 ser\_tm\_reqmffhrdm lsoqn 모듈간 인터페이스와 외부 입출력 신호를 보여주고 있다.

### 3.5 Base\_hdr\_reg블록

Base\_hdr\_reg 블록은 기본적으로 tm\_cell의 헤더 정보로부터 얻어지는 sop, eop 및 payload 바이트 카운트 수에 대한 정보를 각 채널당으로 저장하고 있는 제어 레지스터이다. 큐 매니저 블록으로부터 헤더 승인 신호를 수신후 FIFO에 쓸 데이터의 유효 바이트는 인에이블 신호를 tfifo 블록에 전송하는 역할을 수행한다.

### 3.6 ext\_hdr\_reg블록

모든 tm\_cell은 헤더와 페이로드를 포함하고 있다. 프레임 헤더를 생성하는 인터페이스 사양 요구를 만족시키기 위하여 각각의 sop tm\_cell에서 내장된 여분의 정보가 추출되어야 한다. 메시지 유형, 클래스값, 목적지 포트 번호 및 멀티 캐스트 ID와 같은 정보는 ext\_hdr\_reg 블록에 의해 제공 되어진다. 큐 매니저는 이러한 정보를 사용하여 프레임의 헤더를 만든다 각 채널당 각 FIFO에 대해 이러한 레지스터가 제공된다. 그림 8은 ext\_hdr\_reg 블록의 입출력 신호를 보여주고 있다.

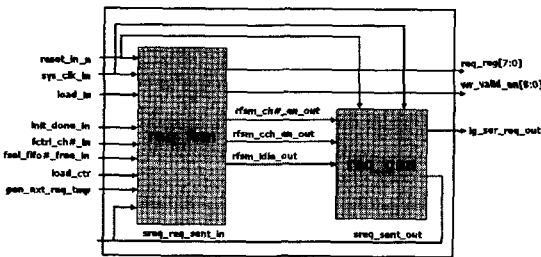


그림 6. ser\_tm\_req 블록의 모듈간 인터페이스

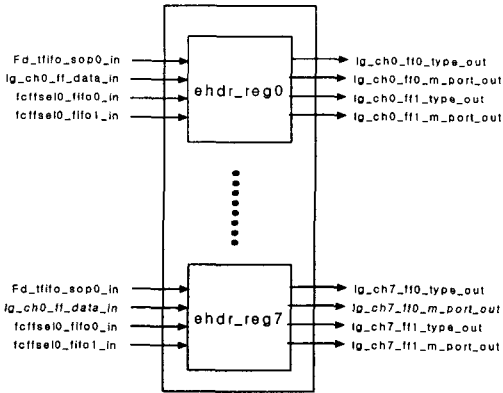


그림 8. ext\_hdr\_reg 블록의 입출력 신호

### 3.7 fifo\_sel블록

fifo\_sel 블록은 각 채널 당 fifo 선택을 위한 상태머신블록인 fsel\_fsm 블록이 존재하며 각 블록의 동작 방식은 동일하다. fifo 뱅크의 변경은 채널의 완전한 패킷을 수신하였을 경우에 변경하게 된다. 또한 fifo\_sel 블록은 트래픽 매니저에게 새로운 request 신호를 발생시킬 수 있다는 정보를 ser\_tm\_req 블록에 알리기 위해 적어도 하나의 FIFO가 비어 있음을 의미하는 신호를 전송한다. 그림 9는 fifo\_sel 블록의 입출력 신호를 보여주고 있다.

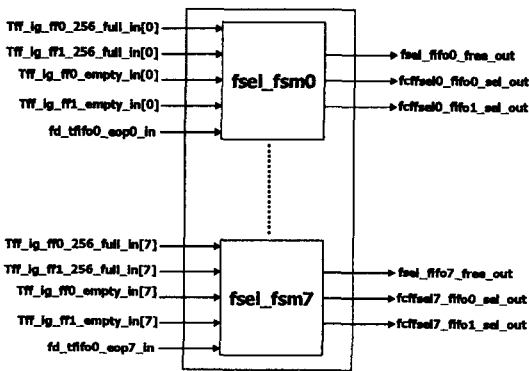


그림 9. fifo\_sel 블록의 입출력 신호

### 3.8 nor\_route블록

nor\_route 블록은 tm2tx 블록에서 유입된 tm\_cell을 디코딩하여 정상적인 데이터 셀인 경우 수신하는 블록이다. 따라서 nor\_route 블록의 모든 입력 신호는 tm2tx 블록으로부터 전달받게 된다. 데이터 바이트 인에이블 신호, sop 신호, eop 신호, 쓰기 인에이블 신호

를 수신하여 단순히 레지스터에 래치하고 fifo\_dat 블록에 전송하는 역할을 수행한다.

### 3.9 fifo\_dat블록

fifo\_dat 블록은 바이트 인에이블 신호에 따라 32비트 데이터로 정렬하는 기능을 수행한다. nor\_route 블록으로부터 수신한 유효 데이터만을 취급하여 tfffo 블록에 데이터를 전송하는 역할을 수행한다. 이 블록에 생성된 데이터는 fifo\_write블록에서 제공하는 쓰기 인에이블 신호를 사용한다.

## IV. 결론

트래픽 매니저로부터 데이터 버스를 통해 16비트의 데이터를 수신하여 각 채널별로 패킷을 분류하며 FIFO에 저장하는 역할을 수행한다. 또한 인터페이스 관리 모듈은 8개의 데이터 클래스를 제공하며, 클래스 번호가 낮은 것이 높은 우선권을 갖도록 하였다. 추가적으로 컨트롤 채널을 통하여 내부의 상태정보를 액세스할 수 있는 기능을 제공한다. 그리고 1~128 바이트의 셀을 가변적으로 수신할 수 있도록 설계되었다. 내부 상태정보를 통한 트래픽 매니저에 특정 채널의 데이터 요청 신호를 생성한다.

본 논문에서 제시한 인터페이스 관리 모듈을 이용하여 ATM 네트워크의 QoS의 지원과 다양한 트래픽의 처리를 효율적으로 수행할 수 있다.

### 참고문헌(또는 Reference)

- [1] Masahide Hatanaka, Toshihiro Masaki, Takao Onoye, VLSI Architecture of Switching Control for AAL Type 2 Switch, IEICE Trans Fundamentals, VolE83-A, No.3 pp435-441, Mar. 2000
- [2] 이형호, 김봉완, 안병준, "테라비트 라우트 기술" Telecommunication Review, 제11권 2호, pp237~247, Mar. 2001.
- [3] 전종암, 변성혁, 안병준, 이형호, "테라비트 라우터 기술 동향", 대한전자공학회지, 제28권, 제9호, pp. 50-59, 2001년 9월