

論文2003-40TC-4-2

IBM NP4GS3 DASL인터페이스와 CSIX-L1인터페이스의 연동구조 및 패킷 제어방안

(A Packet Control method of Interconnection between
IBM NP4GS3 DASL and CSIX Interface)

金光玉*, 崔昌植*, 朴完基*, 崔炳哲*, 郭東鎔*

(Kwang-Ok Kim, Chang-Sik Choi, Wan-Ki Park, Byeoun-Chul Choi, and
Dong-Yong Kwak)

요약

최근의 광 가입자 정합모듈은 매우 빠른 속도로 출현되는 다양한 서비스를 쉽고 효율적으로 수용하기 위해 고성능의 상용 네트워크 프로세서 칩을 대부분 사용하고 있다. 지금까지 많은 벤더들이 2.5Gbps급 네트워크 프로세서를 상용화하였지만, IBM NP4GS3만 2.5Gbps 이상의 우수한 패킷 처리성능을 지원한다. 그러나 IBM NP4GS3는 스위치 인터페이스로 고속 DASL(Data-Aligned Synchronous Link)인터페이스를 사용하기 때문에, NP Forum에 의해 표준화된 CSIX-L1인터페이스를 사용하는 스위치 패브릭과는 정합 할 수가 없다. 이에 따라 본 논문에서는 IBM NP4GS3를 이용한 광 가입자 정합모듈이 표준화된 스위치 패브릭과 효율적으로 정합할 수 있도록 IBM 상용 UDASL칩과 UTOPIA-L3와 CSIX-L1인터페이스를 상호 변환하는 FPGA를 이용한 연동 구조 및 패킷 제어방법에 대해 고찰해본다.

Abstract

Recently, the optical subscriber interface module uses the high performance network processor to quickly develop new application services such as MPLS, VPN, RPR and EPON with a short time-to-market. Although a number of vendors are developing the network processor at 2.5Gbps, only the IBM NP4GS3 can provide packet processing with wire-speed at 2.5Gbps. IBM NP4GS3, however, uses its unique speed DASL interface instead of CSIX-L1 interface, which has standardized by NP Forum currently. Therefore, we implement an interconnection mechanism to use the switch fabric with CSIX-L1 interface. In this paper, we suggest the architecture and a packet control mechanism supporting interconnection between IBM NP4GS3 DASL and CSIX-L1 switch interface using the common IBM UDASL ASIC and XILINX FPGA.

Keywords : CSIX-L1, DASL 인터페이스, UTOPIA-L3, IBM NP4GS3, Switch Interface

* 正會員, 韓國電子通信研究院

(Electronics and Telecommunications Research Institute)

接受日字:2002年8月26日, 수정완료일:2003年3月18日

I. 서론

멀티미디어형 고속 인터넷 및 영상회의, 주문형 비디오, 인터넷게임, HDTV등 다양한 광대역 서비스들을 가

입자들에게 효율적으로 제공하기 위해서는 새로운 광 가입자 기술 개발이 요구된다. 광 가입자 기술 개발은 대부분 다양한 멀티 서비스들을 효율적으로 수용할 수 있을 뿐만 아니라 높은 대역의 패킷 프로세싱 파워를 낼 수 있는 네트워크 프로세서를 이용한다. 현재 네트워크 프로세서는 Agere, Intel, Motorola, Vitesse, AMCC, IBM 등 많은 벤더들에 의해 개발되거나 이미 상용화를 시작하였다^[7].

특히, IBM NP4GS3 상용 네트워크 프로세서는 MPLS 및 VPN, Security, VLAN, IPv6 같은 다양한 서비스와 TBI, SMII, GMII, POS bus 등 다양한 가입자 인터페이스를 지원하며, 또한 L2 기반에서 2.5Gbps 이상의 고성능 패킷 처리를 제공한다. 따라서 다양한 플랫폼 지원을 위해 광 가입자 기술 개발에 IBM 네트워크 프로세서가 많이 사용된다.

현재 NP Forum에서는 2.5Gbps급 네트워크 프로세서와 스위치 패브릭간에 CSIX-L1 (Common Switch Interface Level 1) 스위치 인터페이스 통해 정합 되도록 표준화하였다. 그러나 IBM NP4GS3 네트워크 프로세서는 스위치 인터페이스로 IBM 고유의 고속 DASL 인터페이스를 사용한다. 따라서 IBM NP4GS3를 사용하는 광 가입자 모듈은 스위치 벤더들이 개발한 CSIX-L1 스위치 패브릭과는 정합할 수 없는 문제점이 발생한다^[6]. 현재 IBM은 CSIX-L1 스위치 인터페이스를 지원하는 네트워크 프로세서를 지원하지 않으며, 직접 DASL 스위치 인터페이스와 CSIX-L1 인터페이스를 정합 할 수 있는 모듈도 지원하지 않는다^[5].

FPGA를 이용한 DASL/CSIX-L1 상호 변환 모듈은 DASL 스위치 인터페이스가 400~500MHz의 클럭 속도로 동작하기 때문에 현재 FPGA 기술로는 구현이 불가능하다. 고성능의 FPGA를 사용해서 구현한다 하더라도 FPGA 칩 단가가 너무 비싸 비효율적인 가입자 모듈 개발이 이루어지게 된다. 따라서 현재 DASL 스위치 인터페이스는 ASIC으로 구현된 모듈을 사용해야 한다. 따라서 DASL 스위치 인터페이스는 ASIC으로 구현되어야 한다.

현재 IBM의 UDASL 칩은 ASIC으로 구현된 DASL 스위치 인터페이스 제공한다. 그러나 DASL 스위치 인터페이스가 네트워크 프로세서가 아닌 스위치 패브릭과 정합 되도록 고안되었기 때문에, UDASL 칩의 DASL 인터페이스를 NP4GS3에 정합해 CSIX-L1 스위치 패브릭과 연동하기 위해서는 방향성에 따른 DASL 스위치 인

터페이스 패킷 제어가 요구되며, UDASL의 UTOPIA-L3 패킷을 CSIX-L1으로 변환하는 FPGA 모듈 개발이 요구된다^[1-3].

반면에 IBM 네트워크 프로세서와 달리 IBM PRS Q64G 스위치 패브릭은 스위치 인터페이스로 DASL, CSIX-L1, UTOPIA-L3 3가지의 인터페이스를 제공하여, CSIX-L1 인터페이스를 사용하는 네트워크 프로세서와 정합을 제공한다^[1, 5].

따라서 가입자에게 다양한 서비스 제공 및 고속의 패킷 처리를 수행하기 위해서, 가입자 정합 모듈은 2.5Gbps 이상의 패킷 처리를 수행하는 IBM NP4GS3 네트워크 프로세서를 사용해야 하며, 표준화된 스위치 인터페이스와 정합하기 위해 NP4GS3 DASL 스위치 인터페이스를 CSIX-L1 스위치 인터페이스와 정합 해야 한다. 그리고 이에 따른 패킷 흐름제어가 필요하게 된다. 그러므로 본 논문에서는 DASL 스위치 인터페이스를 CSIX-L1 스위치 인터페이스와 정합하기 위한 구조 및 패킷 제어방법을 제시한다.

본 논문의 구성에서 II장에서는 상용 IBM NP4GS3 칩의 스위치 인터페이스 구조 및 스위치 패브릭에서 지원하는 인터페이스 구조를 고찰하고, III장에서는 본 논문에서 제안한 UDASL 칩을 이용한 CSIX-L1 인터페이스 정합 구조 및 셀 제어방법에 대해 논의한다. IV장에서는 UTOPIA-L3 인터페이스를 CSIX-L1 인터페이스로 변환하는 FPGA 내부 구조에 대해서 살펴보고, 마지막으로 결론을 맺는다.

II. DASL 스위치 인터페이스와 스위치 패브릭 인터페이스 구조

1. IBM NP4GS3 DASL 스위치 인터페이스 구조

스위치와 네트워크 프로세서 사이는 크게 두 가지 방법으로 정합된다. 첫째, 네트워크 프로세서와 스위치를 공동으로 개발하는 Intel과 IBM, Vitesse 등의 업체들은 추가적인 Glue Logic 없이 회사 고유의 스위치 인터페이스나 표준에 따라 CSIX-L1 인터페이스로 정합한다. 둘째, 네트워크 프로세서만을 개발하는 Agere나 EZchip 등 업체들은 자신의 스위치 인터페이스를 CSIX-L1과 정합하기 위해 추가적인 Glue Logic을 사용한다. 네트워크 프로세서는 FOCUS-32, POS-PHY III, UTOPIA-L3, CSIX-L1 등을 스위치 인터페이스로 사용한다. 그중 IBM NP4GS3 네트워크 프로세서는 고유의 DASL 스위

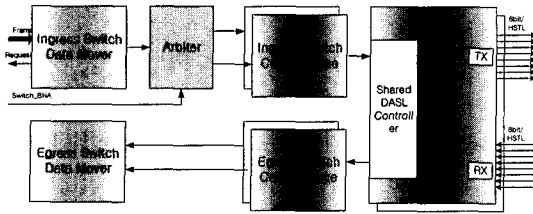


그림 1. NP4GS3스위치 인터페이스 구조
Fig. 1. An architecture of NP4GS3 switch interface.

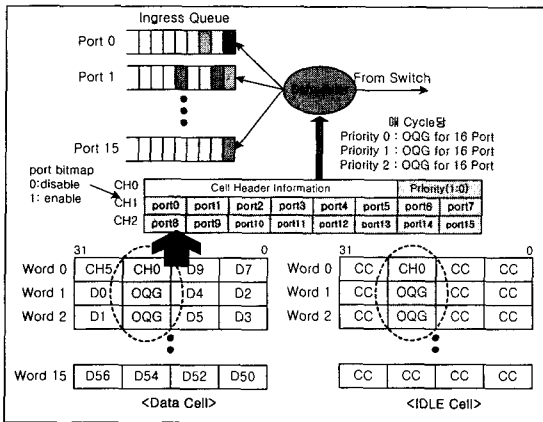


그림 2. VOQ 처리를 위한 OQG 포맷
Fig. 2. OQG format for VOQ processing.

스위치 인터페이스를 사용한다. IBM NP4GS3 네트워크 프로세서는 고유의 DASL스위치 인터페이스를 사용한다. DASL스위치 인터페이스는 8비트의 Differential 데이터 버스를 이용하여, 스위치로부터 50~62.5MHz 클럭을 받아 네트워크 프로세서에서 8배로 분주해 400~500MHz의 속도로 패킷을 전송한다.

NP4GS3의 스위치 인터페이스 구조는 <그림 1>과 같다^[2]. 룩업을 수행한 프레임은 64바이트 단위의 셀로 분해되어 DASL스위치 인터페이스로 입력된다. Ingress Switch Data Mover는 입력되는 프레임과 스위치 패브릭에 전송되는 셀 사이의 완충역할 및 제어기능을 수행한다. 또한 Egress Switch Cell Interface 블록에서 입력되는 OQG(Output Queue Grant)정보를 기반으로 VOQ (Virtual Output Queuing)을 수행한다. 하나의 프레임은 첫 번째 64바이트 셀 전송에서 라우팅을 위한 6바이트의 셀 헤더와 셀 재조립에 사용되는 10바이트의 프레임 헤더 그리고 48바이트의 데이터로 전송하며, 그 다음 셀부터는 6바이트의 패킷 헤더와 58바이트의 데이터로 전송한다.

Ingress Switch Cell interface 블록은 DASL Macro

에게 연속적인 셀을 전송하며, Power On/Reset시 동기화 셀을 전송한다. 동기화가 끝나면 데이터 셀이나 idle 셀을 전송한다. 또한 1개의 Master LU(Logical Unit)와 3개의 Slave LU로 구성된 스위치 패브릭 포맷에 맞게 셀을 변환하는 기능을 수행한다. DASL Macro블록은 Multi-bit Serialization/Deserialization기능을 수행하며, Bit Positioning을 수행하여 스위치 인터페이스 상에 패킷 에러가 있는지 검사한다. Egress Switch Cell Interface 블록은 스위치로부터 입력되는 셀 헤더의 패리티 체크를 수행하며, idle 셀이면 OQG정보만 추출하고 Discard한다. <그림 2>와 같이 idle이나 data 셀로부터 추출된 OQG정보는 Ingress 셀 흐름제어를 위해 사용된다. Egress Switch Data Mover 블록은 프레임 헤더 분석 및 입력되는 셀을 프레임으로 생성하는 기능을 수행한다. 매 클럭당 스위치 16포트에 대한 Priority별 상태를 알려준다.

2. IBM Q64G 스위치 패브릭 인터페이스 구조

현재 IBM Q64G 스위치 패브릭은 네트워크 프로세서와 스위치 정합을 위해 DASL, CSIX-L1, UTOPIA-L3 3가지 인터페이스를 제공한다. 이 3가지 인터페이스를 위한 구조는 <그림 3>과 같다^[3,5]. DASL스위치 인터페이스는 IBM 네트워크 프로세서와 직접 정합 한다. DASL스위치 인터페이스를 사용하지 않은 네트워크 프로세서를 위해 IBM 스위치는 UTOPIA-L3와 CSIX-L1 인터페이스를 지원한다.

UTOPIA-L3인터페이스는 IBM의 상용 UDASL칩을 사용하여, 네트워크 프로세서는 UTOPIA-L3 인터페이스로 125MHz/32Bit 데이터 버스를 통해 정합 되고 스

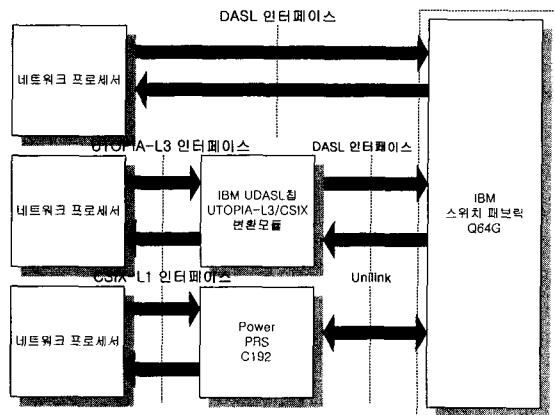


그림 3. IBM Q64G 스위치 인터페이스 구조
Fig. 3. An Architecture of Q64G Switch Interface.

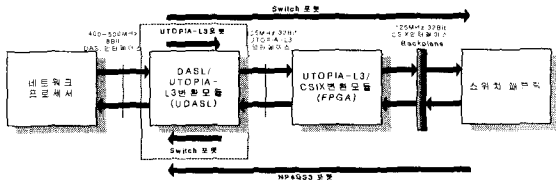


그림 4. 제안한 DASL 인터페이스와 CSIX-L1 인터페이스 정합구조

Fig. 4. An architecture of Interconnection between DASL and CSIX-L1 interface.

위치 패브릭과는 ASIC으로 구현된 DASL 인터페이스를 통해 500MHz/8Bit로 정합 된다. CSIX-L1 인터페이스는 IBM 상용 PowerPRS C192칩을 이용하여 4개의 CSIX-L1 인터페이스와 정합할 수 있다. 네트워크 프로세서는 CSIX-L1 인터페이스 중 1개와 125MHz/32Bit 데이터 버스를 통해 정합 되고 스위치 패브릭과는 Unlink 인터페이스를 통해 정합 된다^[4].

그러나 현재 네트워크 프로세서에서 UTOPIA-L3 인터페이스나 CSIX-L1 인터페이스를 사용하여 2.5Gbps로 트래픽을 처리하는 칩은 없다. 따라서 2.5Gbps로 트래픽을 처리하기 위해서는 IBM NP4GS3 네트워크 프로세서를 사용하며, 표준화된 CSIX-L1 인터페이스를 사용하는 스위치 패브릭과 정합할 수 있는 방법이 요구된다. 따라서 본 논문에서는 IBM 네트워크 프로세서가 표준화된 CSIX-L1 인터페이스의 스위치 패브릭과 정합 할 수 있는 구조 및 셀 제어 방법을 제시하였다.

III. 제안한 DASL 스위치 인터페이스와 CSIX-L1 인터페이스 정합구조

1. IBM NP4GS와 CSIX-L1 인터페이스 정합구조

본 논문에서는 2.5Gbps급 가입자 정합모듈에서 IBM NP4GS3 네트워크 프로세서를 사용하는 경우 DASL 스위치 인터페이스를 표준화된 CSIX-L1 인터페이스로 변환하기 위해 상용 UDASL 칩과 UTOPIA-L3 인터페이스를 CSIX-L1 인터페이스로 정합 하는 FPGA를 개발해 구성하였다. UDASL 칩을 사용한 이유는 ASIC으로 구현된 고속 DASL 인터페이스를 사용하기 위해서이다. 본 논문에서 제안한 구조에서는 <그림 3>에서와 같이 IBM에서 고안된 연결방식과 달리 UDASL 칩의 DASL 인터페이스가 스위치 패브릭이 아닌 네트워크 프로세서와 연동되도록 구성하고, UTOPIA-L3 인터페이스를 네

트워크 프로세서가 아닌 UTOPIA-L3/CSIX 변환모듈 FPGA와 정합 하였다. 또한 UTOPIA-L3/CSIX 변환모듈 FPGA가 CSIX-L1 인터페이스를 통해 스위치 패브릭과 연동되도록 구성하였다. 여기서 UTOPIA-L3/CSIX 변환모듈은 32Bit/100~125MHz로 구현되기 때문에 FPGA 기술로 충분히 구현 가능하다^[8].

DASL 인터페이스를 CSIX-L1 인터페이스와 연동하기 위한 기본 구조는 <그림 4>와 같다. 상용 UDASL 칩은 DASL 인터페이스를 통해 입력된 네트워크 프로세서 셀을 UTOPIA-L3 패킷으로 변환하여 FPGA 모듈에 전달하는 기능을 수행하며, 또한 FPGA로부터 입력되는 UTOPIA-L3 패킷을 DASL 셀 포맷으로 변환하여 네트워크 프로세서에 전달하는 기능을 수행한다. FPGA 모듈은 64바이트의 UTOPIA-L3 패킷을 CSIX-L1 인터페이스에 사용되는 Base Header와 Extension Header를 추가하여 76바이트의 프레임으로 변환하는 기능을 수행한다. 여기서 UTOPIA-L3 패킷은 CSIX-L1 프레임의 페이로드에 위치한다. 또한 스위치로부터 입력되는 프레임에서 프레임 헤더를 Discard하고 페이로드에서 UTOPIA-L3 패킷을 꺼내 스위치 포트 상태에 따라 네트워크 프로세서가 VOQ를 수행할 수 있도록 OQG 정보를 삽입하여 전송한다. 그러나 UDASL 칩은 스위치 패브릭과 연동되도록 고안되었기 때문에 DASL 인터페이스로 입력되는 셀 포맷을 스위치에서 입력되는 셀 포맷으로 인식하게 된다. 특히 VOQ를 지원하기 위해 스위치에서 OQG 정보를 패킷 헤더에 포함하는데, 이때 DASL 인터페이스의 방향성 문제로 UDASL 칩에서 잘못 처리하게 된다. 따라서 잘못 처리된 OQG 정보를 FPGA에서 무시하고, 반대로 CSIX-L1 인터페이스를 통해 입력되는 제어 CFrame의 정보를 바탕으로 OQG 정보를 생성해 네트워크 프로세서까지 Discard되지 않고 전송해야한다. 이후 부터는 제안한 구조의 동기화 방법 및 패킷 제어방법에 대해서 고찰해 본다.

2. 제안한 구조의 동기화 설정 방법.

본 논문에서 제안한 구조에서는 DASL 인터페이스와 CSIX-L1 인터페이스에서 동기화가 이루어진다. CSIX-L1 인터페이스는 표준화된 방식대로 In-Band 파라미터인 Ready Bit를 이용해 동기화를 설정한다. 반면 UDASL 칩의 DASL 인터페이스를 스위치가 아닌 네트워크 프로세서에 사용하기 때문에 방향성에 따른 동기화 방법이 고려되어야 한다.

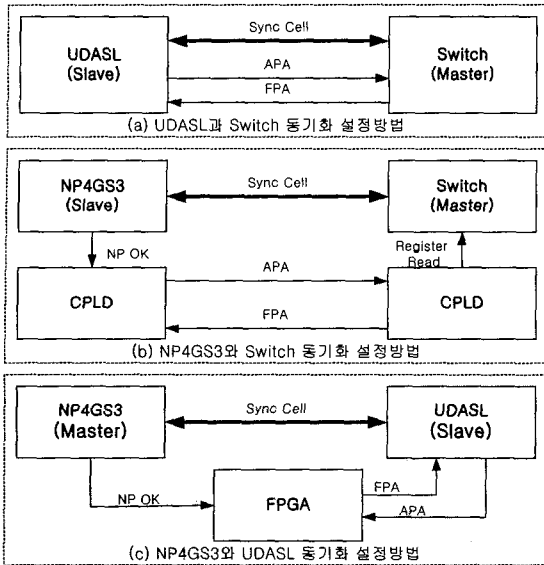


그림 5. 동기화 설정방법 및 구조
Fig. 5. An architecture of synchronous method.

<그림 5>는 현재 IBM에서 사용하는 동기화 방식 및 본 논문에서 제안한 동기화 구조를 보여준다. <그림 5(a)>에서처럼 IBM UDASL칩은 스위치 패브릭과 APA (Adapter Port Available)와 FPA(Fabric Port Available)신호를 이용해 동기화를 설정한다. 즉 스위치 패브릭이 마스터가 되어 SYNC 셀을 이용해 동기화가 수행되면 FPA신호를 통해 동기화가 완료됨을 알려주고, 슬레이브 역할을 수행하는 UDASL칩은 APA신호를 통해 동기화가 수행됨을 알려준다.

또한 <그림 5(b)>와 같이 현재 IBM 네트워크 프로세서 칩에는 FPA와 APA 하드웨어적인 핀이 존재하지 않기 때문에 CPLD를 이용해 동기를 맞춘다. 이때도 역시 스위치 패브릭이 마스터가 되어 동기화가 수행되면 FPA 신호를 통해 동기화가 완료됨을 알려주고, 슬레이브 역할을 수행하는 네트워크 프로세서가 동기화가 완료되면 CPLD에게 알려 APA신호를 전달하게 함으로서 동기화를 수행한다.

<그림 5(c)>는 본 논문에서 제안한 구조에서의 동기화 방법을 보여준다. UDASL칩을 스위치 패브릭이 아닌 네트워크 프로세서에 DASL인터페이스로 정합 하였기 때문에 스위치에서 생성되어야할 FPA신호를 네트워크 프로세서가 생성해야 한다. 또한 FPA신호를 생성하기 위해 기존에 방법과 달리 네트워크 프로세서가 마스터가 되어 동기화가 끝나면 NP OK신호를 FPGA에게 전달해 네트워크 프로세서가 동기화가 끝났음을 FPA신

호를 통해 알려준다. FPA신호를 받은 UDASL칩은 동기화가 끝나면 APA신호를 통해 알려준다.

3. 제안한 구조의 셀 제어방법

네트워크 프로세서에서 생성된 셀은 <그림 6>과 같이 DASL인터페이스, UTOPIA-L3인터페이스, CSIX-L1 인터페이스, OQG에 따른 네트워크 프로세서 VOQ 제어 등 패킷 흐름제어를 따라 스위치 패브릭까지 셀을 전송하게 된다. <그림 7>과 같이 현재 IBM 네트워크 프로세서는 셀 제어를 위해 Send_Grant신호와 Master_Grant(1:0)신호를 사용한다. 즉 네트워크 프로세서는 상대방에게 셀을 받을 수 없는지의 정보를 Send_Grant신호로 알려주면 이 신호에 따라 상대방에서 셀 Start/Stop을 수행한다. 또한 상대방에서 입력되는 Master_Grant(1:0)신호의 인코딩 정보를 이용해서 해당 Priority에 셀을 전달하게 된다. 네트워크 프로세서는 3개의 Priority를 지원한다. 반면에 스위치 패브릭은 4개의 Priority를 지원하고 있다.

그러나 현재 스위치 패브릭은 각 Priority가 패킷을 받을 수 있는지 없는지의 정보를 Mem_Grant(3:0)신호의 비트맵으로 알려준다. 따라서 스위치 패브릭의 Mem_Grant(3:0)신호와 네트워크 프로세서 Master_

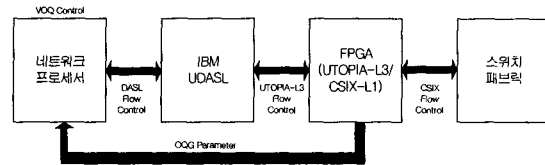


그림 6. 패킷 제어 수행범위
Fig. 6. The scope of packet flow control.

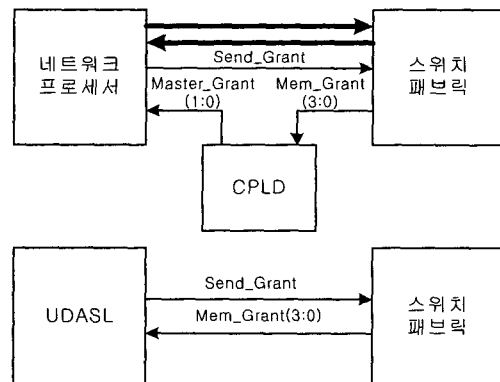


그림 7. IBM 네트워크 프로세서 셀 제어구조
Fig. 7. An architecture of cell flow control using the IBM NP.

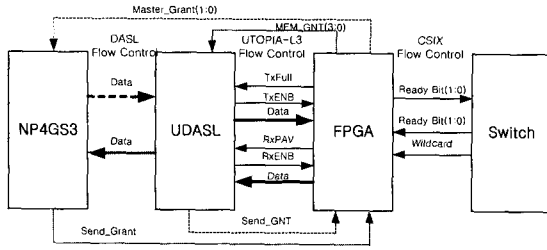


그림 8. 제한한 패킷 제어 구조
 Fig. 8. An architecture of proposed packet flow control.

Grant(1:0)신호가 서로 핀 매칭이 안되기 때문에 <그림 5>에서 사용하는 CPLD를 통해 스위치 패브릭으로부터 입력되는 Mem_Grant(3:0)신호를 인코딩해 네트워크 프로세서에게 알려주어 셀 제어를 수행한다. 또한 UDASL 칩은 스위치 패브릭과 Send_Grant신호와 Mem_Grant(3:0)신호를 이용해 셀 제어를 수행한다.

본 논문에서 제한한 셀 제어는 <그림 8>과 같이 수행된다. 특히 DASL인터페이스의 경우 네트워크 프로세서와 스위치 패브릭이 아닌 네트워크 프로세서와 UDASL칩 사이에서 셀 흐름 제어를 수행한다. 이에 따른 셀 제어를 위해 네트워크 프로세서가 셀을 받을 수 있다고 Send_Grant신호로 FPGA에게 알려주면, FPGA에서는 Control CFrame의 정보를 바탕으로 스위치 상태를 체크하여 Mem_Grant(3:0)신호를 UDASL칩에 전

달한다. 이 신호를 받은 UDASL칩은 셀을 네트워크 프로세서에게 전달한다. 마찬가지로 UDASL칩이 셀을 받을 수 있다고 Sent_Gnt신호를 FPGA에 전달하면 Control CFrame의 정보를 바탕으로 Master_Grant(1:0)신호를 네트워크 프로세서에게 전달해 셀을 UDASL칩에 전송하게 한다. 또한 FPGA칩이 더 이상의 셀을 받을 수 없는 경우 Master_Grant(1:0)신호를 "00"으로 전송해 네트워크 프로세서로부터 더 이상의 패킷이 전달되지 않도록 패킷 제어를 수행한다. UDASL칩과 FPGA 사이에는 UTOPIA-L3인터페이스를 통해 패킷 제어를 수행한다. UDASL의 송수신 UTOPIA-L3인터페이스는 마스터로 동작한다.

UDASL칩에서 패킷을 받을 수 있다고 RXENB신호를 FPGA에 전송하면 FPGA는 패킷의 시작을 나타내는 RXSOP신호와 패킷을 UDASL칩에 전달한다. RXPAV신호는 FPGA에서 패킷을 보낼 준비가 되어있음을 UDASL칩에 알린다. 역시 마찬가지로 FPGA가 패킷을 받을 수 있을 때 TXFULL신호를 UDASL칩에 전달하면, UDASL칩은 TXENB신호를 이용해 전송할 패킷이 있음을 표시하고, 패킷의 시작을 나타내는 TXSOP신호와 패킷을 FPGA칩에 전달한다^[9].

CSIX-L1인터페이스는 프레임 헤더의 Ready bit를 이용해 서로 패킷을 받을 수 있는지 없는지를 표시한다. 이때 패킷 레벨 핸드 셰이킹이 되어야 하므로 최소한

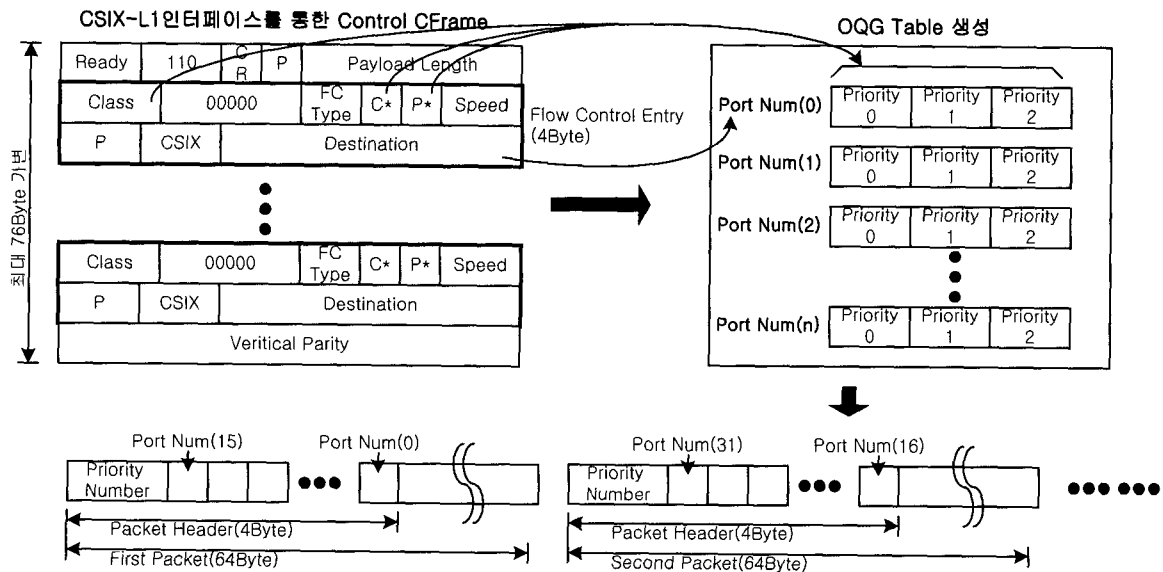


그림 9. 네트워크 프로세서 VOQ지원을 위한 OQG전송방법
 Fig. 9. The method of OQG transmission for supporting VOQ to the network processor.

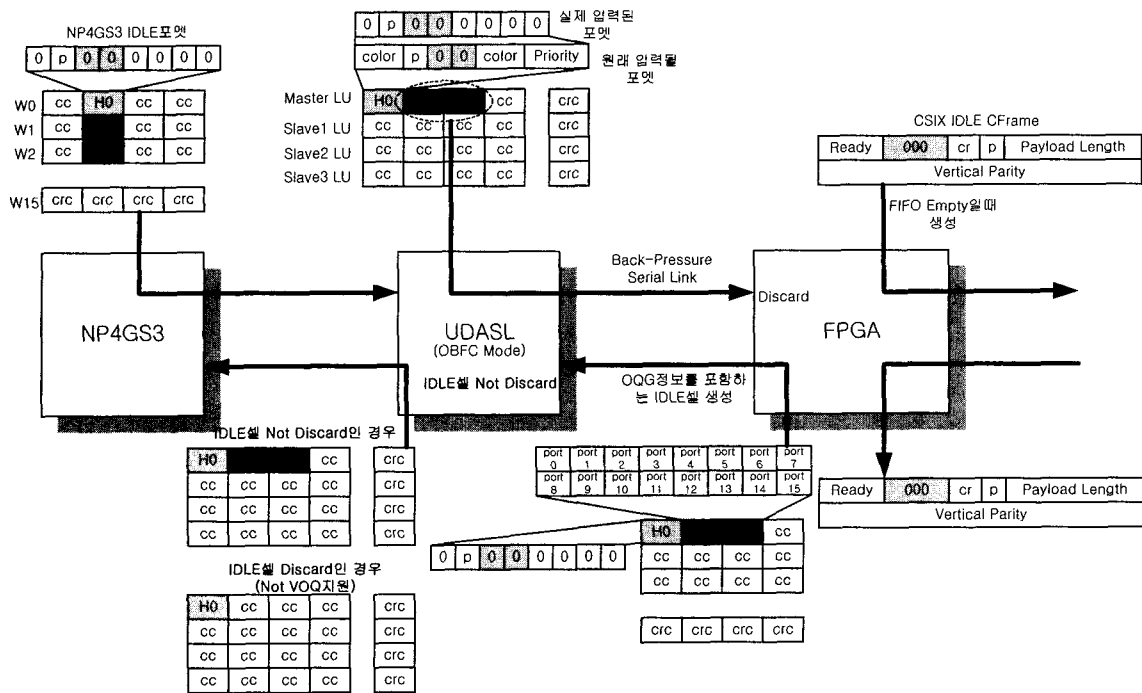


그림 10. 네트워크 프로세서와 스위치 패브릭간의 IDLE패킷 처리방법

Fig. 10. The method of IDLE packet processing between the network processor and switch fabric.

하나의 패킷을 받을 수 있는 여유메모리를 확보해야한다. 위 코드 OQG정보를 기반으로 Priority별 DASL인터페이스 흐름제어를 표시하는 pseudo code를 나타낸다.

```

***** DASL Interface Flow Control *****
if Send_Grant = high
    NP4GS3는 UDASL로부터 패킷 받음
    if Priority(Port Number)(Priority Number)_OQG
        case "000" => Mem_Gnt <= "0000" (NO Grant)
            "001" => Mem_Gnt <= "1000" (Priority0)
            "010" => Mem_Gnt <= "1100" (Priority1)
            "111" => Mem_Gnt <= "1110" (Priority2)
        else
            Mem_Gnt <= "0000" (NO Grant)

if Send_Gnt = high
    UDASL은 NP4GS3로부터 패킷 받음
    if Priority(Port Number)(Priority Number)_OQG
        case "000" => Master_Grant <= "00" (No Grant)
            "001" => Master_Grant <= "01" (Priority0)
            "010" => Master_Grant <= "10" (Priority1)
            "111" => Master_Grant <= "11" (Priority2)
    elsif FPGA_FIFO = Full
        Master_Grant <= "00" (No Grant)
    
```

4. 제안한 구조의 VOQ 지원방법

네트워크 프로세서가 VOQ기능을 수행하기 위해서는 스위치 출력포트의 상태를 알아야한다. 따라서 VOQ를 지원하기 위해 본 제안 모델에서는 CSIX-L1인터페이스를 통해 입력되는 제어정보를 바탕으로 OQG정보를 생성해 네트워크 프로세서에게 알려준다. OQG정보는 Destination Address정보에 삽입해서 전송한다.

이 Destination Address정보 필드는 스위치로 입력될 때는 스위치 라우팅 정보로 사용되지만 스위치에서 출력될 때는 무의미한 필드가 되므로 이 필드를 이용해 계속 네트워크 프로세서에게 각 포트 및 각 Priority에 대한 정보를 전송한다. 기본 구조는 <그림 9>와 같다. OQG 정보 생성과정 및 전송과정은 다음과 같다.

- (1) CSIX-L1인터페이스를 통해 Control CFrame정보가 FPGA에 입력된다. 이 프레임에서 Class정보와 C*/P* 정보 그리고 Destination Address정보를 추출한다.
- (2) C* = "1"이면 Destination Address에 해당하는 포트의 모든 Priority 비트를 "0"으로 설정한다. 이것은 이 Priority 위치로는 데이터를 전송할 수 없음을 의미한다.

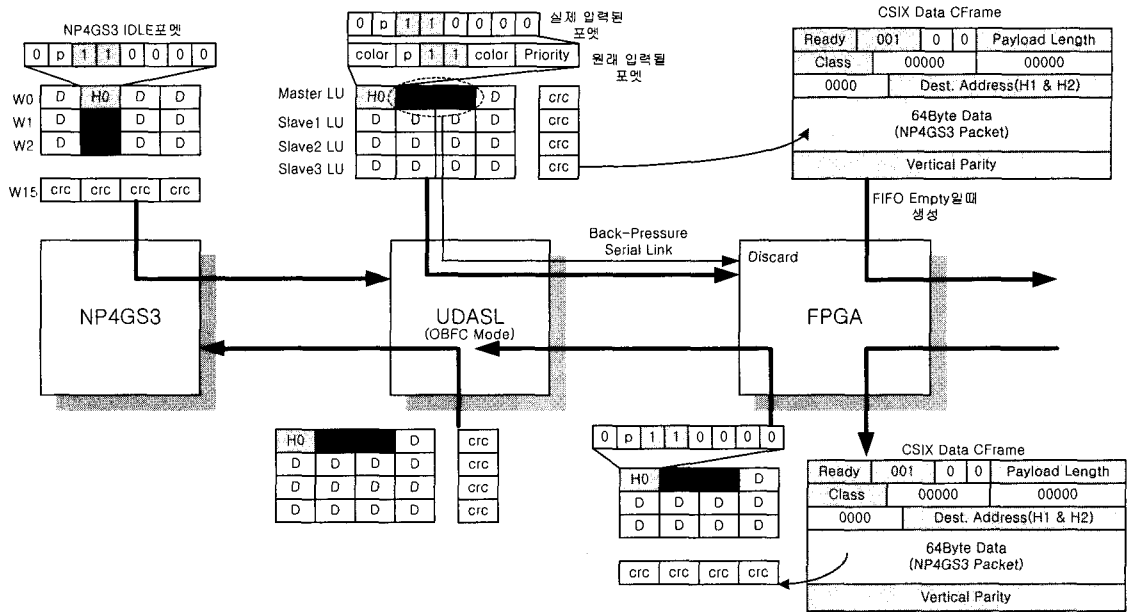


그림 11. 네트워크 프로세서와 스위치 패브릭간의 DATA패킷 처리방법
 Fig. 11. The method of DATA packet processing between the network processor and switch fabric.

- (3) C* = "0"이고 P* = "0"이면 Class비트와 Destination Address를 바탕으로 해당 포트의 특정 Priority만 Blocking한다.
- (4) C* = "1"이고 P* = "1"이면 모든 포트와 Priority를 Blocking한다.
- (5) C* = "0"이고 P* = "1"이면 모든 포트에서 Class비트에 설정된 Priority만 Blocking한다.
- (6) OQG생성 테이블을 바탕으로 네트워크 프로세서에게 패킷을 전달할 때 첫 번째 4바이트 데이터에 특정 Priority 타입에서 최대 16개 포트의 정보를 실어서 전송한다. 만약 16포트 스위치를 이용할 경우 3번의 패킷전송으로 모든 포트의 Priority Blocking 정보를 전송할 수 있다.

5. 제안한 구조의 IDLE/DATA 패킷 전송방법

UDASL칩은 DASL인터페이스를 통해 스위치의 OQG 정보를 받아 UTOPIA-L3인터페이스를 통해 네트워크 프로세서에게 알려주는 메커니즘으로 구현되어 있다. 그러나 본 논문에서 제안한 구조는 DASL인터페이스를 스위치가 아닌 네트워크 프로세서 NP4GS3에 인터페이스 되도록 설계되어, UDASL칩은 NP4GS3에서 보낸 패킷 정보를 스위치에서 OQG정보를 포함해서 들어오는 패킷으로 잘못 인식하여 잘못된 VOQG정보를 생성하게 된다. 이 잘못된 OQG정보는 스위치 Ingress 방향에서는 사용

되지 않으므로 FPGA에서 무시해버리고, 스위치 패브릭에서 들어오는 제어 프레임을 바탕으로 OQG정보를 삽입해서 전송하게 된다. <그림 10>은 네트워크 프로세서와 스위치 패브릭간에 IDLE패킷 처리 방법을 보여준다. 기본 동작은 다음과 같다.

- (1) 네트워크 프로세서는 전송할 패킷이 없을 경우 IDLE패킷을 전송한다. 이때 PQ헤더를 제외한 나머지 필드는 모두 "CC"값이 할당된다.
- (2) UDASL칩은 IDLE패킷이 입력되면 PQ를 보고 IDLE패킷인지 체크하고 "CC"정보를 이용해 잘못된 OQG정보를 Back-Pressure Serial Link를 통해 FPGA에게 전송한다. IDLE패킷은 UDASL칩에서 Discard된다. 여기서 원래 입력되어야 할 패킷헤더 포맷과 네트워크 프로세서로부터 입력되는 패킷헤더의 포맷에서 Packet Type필드 위치가 동일하므로 UDASL칩은 Packet Type을 확인할 수 있다.
- (3) FPGA에서는 FIFO가 Empty가 되면 CSIX-L1 IDLE CFrame을 전송한다.
- (4) FPGA는 스위치로부터 입력되는 패킷이 없어 FIFO가 Empty가 되면 네트워크 프로세서에 IDLE패킷을 전송한다. 이때 이 IDLE패킷에 OQG정보도 추가된다.
- (5) UDASL칩은 입력단에서 FPGA에서 입력된 IDLE패킷을 Discard하면 출력단에서 "CC" 포함한 IDLE패킷을

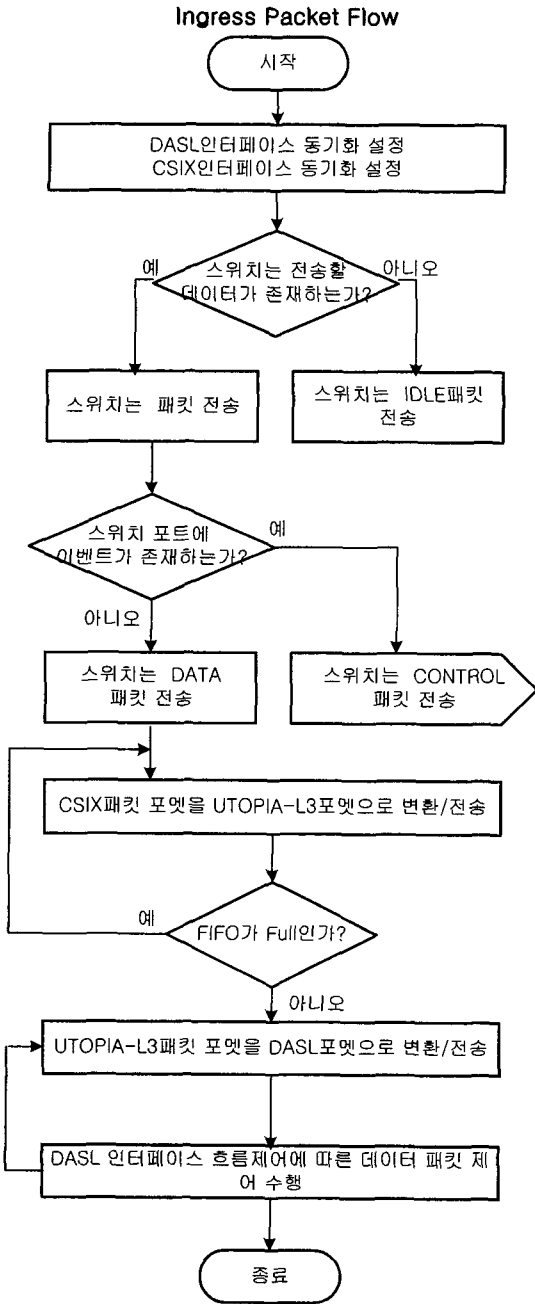


그림 12. Ingress 패킷 흐름도
Fig. 12. Flowchart of ingress packet.

전송하게 되므로 네트워크 프로세서는 잘못된 VOQ를 수행한다. 따라서 UDASL칩에서 IDLE패킷이 입력되더라도 Discard하지 않도록 레지스터를 설정해야한다. (6) 네트워크 프로세서는 UDASL칩으로부터 입력되는 IDLE패킷의 OQG정보를 이용해 VOQ를 수행한다. <그림 11>은 네트워크 프로세서와 스위치 패브릭간

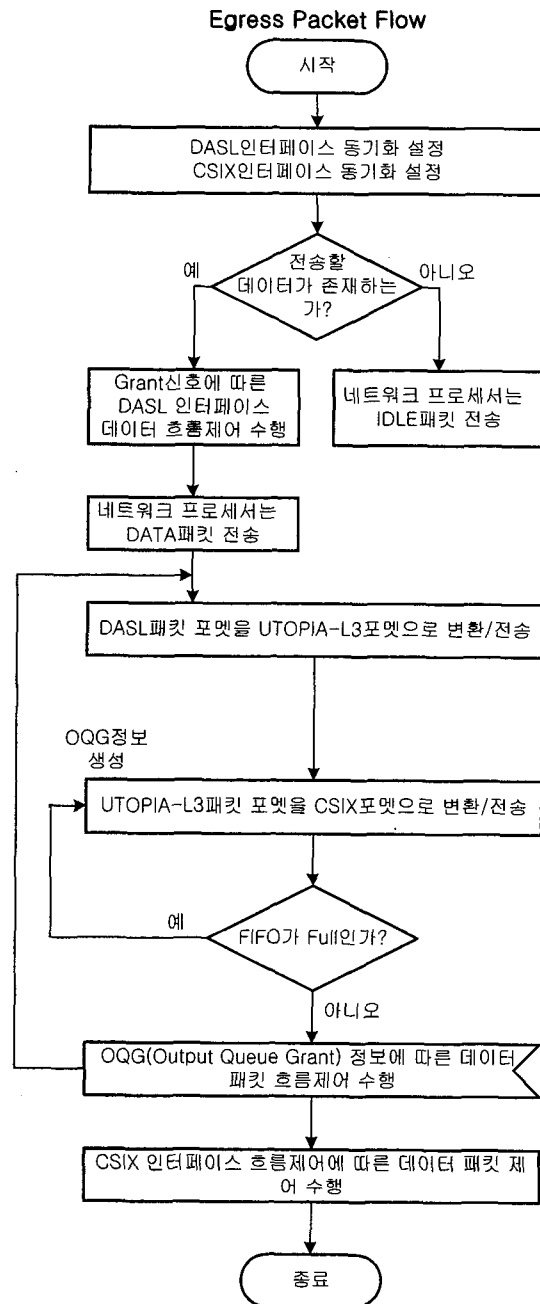


그림 13. Egress 패킷 흐름도
Fig. 13. Flowchart of egress packet.

의 데이터 패킷 처리방법을 보여준다. 기본 동작은 다음과 같다.

- (1) 네트워크 프로세서는 입력된 프레임의 록업을 수행한 후 64바이트 단위로 패킷을 생성해 전송한다. 이때 패킷 헤더에는 데이터의 타입을 나타내는 "11"의 값이 포함되고, 스위치에서 라우팅에 사용되는 목적

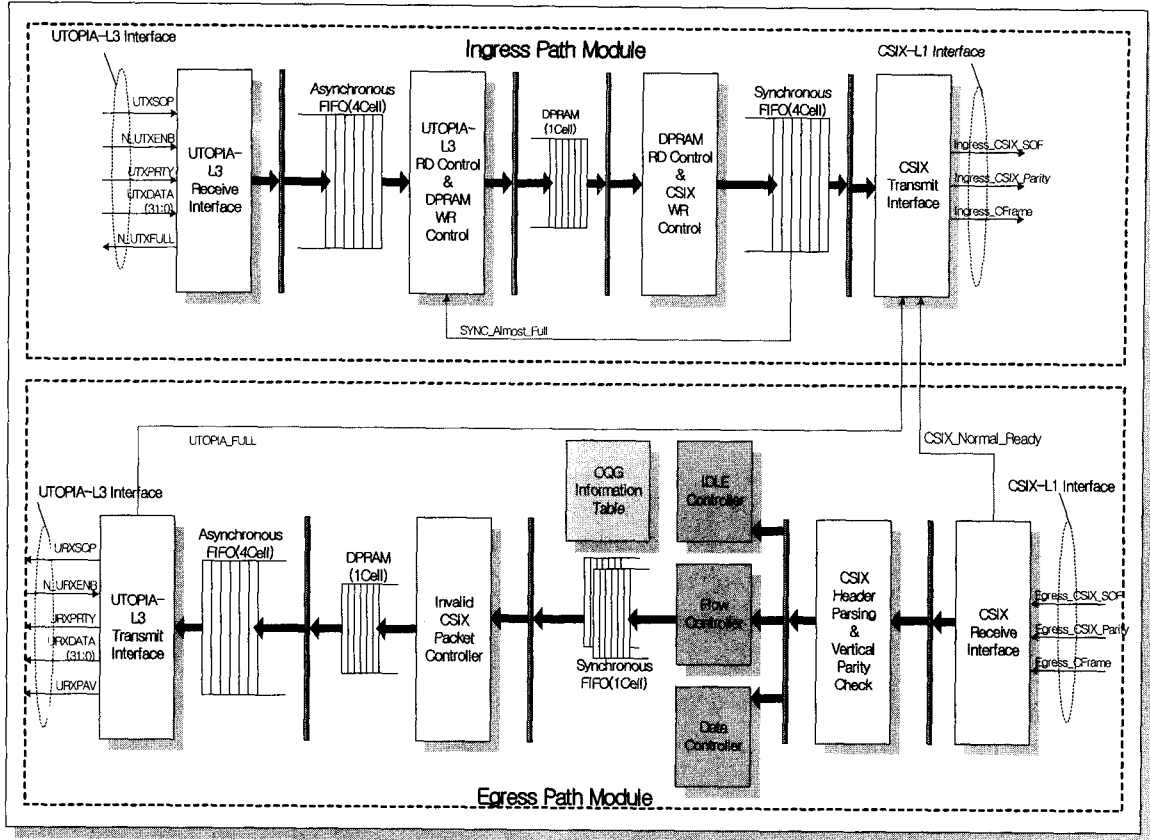


그림 14. UTOPIA-L3/CSIX-L1 변환모듈
 Fig. 14. The Converter between UTOPIA-L3 and CSIX-L1.

- 지 주소가 포함된다.
- (2) UDASL칩은 Data패킷이 입력되면 PQ를 보고 Data 패킷인지 체크하고 이 패킷을 FIFO에 저장한 후 전송한다. 이때도 역시 목적지 주소 필드를 이용해 잘못된 OQG를 Back-Pressure Serial Link를 통해 FPGA에게 전송한다.
 - (3) FPGA에 데이터 패킷이 입력되면 네트워크 프로세서 헤더로부터 목적지 주소, 클래스 정보등을 추출하여 기본헤더 및 확장헤더, Vertical Parity정보를 생성하여 76바이트 프레임을 스위치 패브릭에 전달한다. 이때 패이로드에는 네트워크 프로세서 패킷이 입력된다.
 - (4) FPGA는 스위치로부터 입력되는 데이터 프레임으로부터 64바이트 패이로드 정보만 추출한 후 패킷 헤더속에 OQG정보를 삽입하여 전송한다.
 - (5) UDASL칩은 입력단에서 데이터를 받으면 스위치 포맷으로 변환하여 네트워크 프로세서에 전달한다.
 - (6) 네트워크 프로세서는 입력되는 패킷을 하나의 프레

임으로 조립한 후 물리층 포트로 전달하고, OQG정보를 이용해 VOQ를 수행한다.
 <그림 12>와 <그림 13>은 각각 Ingress/Egress 패킷 제어에 따른 순서도를 나타낸다.

IV. 제안한 UTOPIA-L3/CSIX-L1 변환 FPGA 모듈구조

본 논문에서 제안한 구조에 사용되는 FPGA모듈은 UDASL칩의 UTOPIA-L3인터페이스와 스위치의 표준 CSIX-L1인터페이스 사이를 정합하는 기능을 수행한다. <그림 14>와 같이 FPGA모듈은 송신 UTOPIA-L3인터페이스를 통해 입력되는 유토피아 패킷을 CSIX-L1인터페이스 프레임으로 변환하여 스위치 패브릭에 전달하는 Ingress모듈과 CSIX-L1인터페이스를 통해 입력되는 프레임을 UTOPIA-L3인터페이스 패킷으로 변환하여 UDASL칩에 전달하는 Egress모듈로 구성된다. 고속 패킷 변환을 위해 4단 파이프라인 구조로 설계하였다. 송

수신 UTOPIA-L3 인터페이스는 모두 Slave모드로 동작하며, 32비트 데이터 버스를 사용하며, 100MHz 클럭은 UDASL칩에서 제공된다. UDASL칩이 FPGA로 전송할 패킷이 있으면, FPGA에서 제공된 N_UTX FULL신호에 따라 패킷을 전송한다. 반면에 FPGA가 UDASL에게 전송할 패킷이 있으면, UDASL칩은 패킷을 받을 수 있을 때, N_URXENB 신호를 Low로 하여 FPGA가 URXSOP와 함께 4바이트의 패킷 스트림을 전송하게 한다. Ingress모듈에서 FPGA로 패킷 데이터가 입력되면 패킷이 패리티 에러가 없는지 확인한 후 Asynchronous FIFO에 저장한다. 패킷 단위 흐름제어를 수행하기 때문에 16개의 클럭 주기로 하나의 패킷이 입력된다. 만약 1클럭안에 새로운 SOP가 나타나면 앞의 패킷은 제대로 모두 들어오지 않았으므로 폐기하게 되고 그에따른 메모리는 갱신된다. 이를 위해 1패킷 크기의 DPRAM과 Controller가 사용된다. 하나의 완전한 패킷이 입력되면 이 패킷을 CSIX 프레임의 페이로드 속에 넣게되고, Base헤더 및 Extension헤더, Vertical헤더를 생성하여 전송한다. 추가적인 CSIX헤더를 생성하고 전송하는 동안 패킷은 Synchronous FIFO에 저장된다. 스위치로의 프레임 전송의 제어는 스위치에서 전송된 프레임속의 Ready신호를 이용하여 제어한다. 스위치로 전송할 프레임이 없으면 FPGA는 IDLE프레임을 전송한다. Egress모듈에서 스위치는 데이터 프레임과 제어 프레임 IDLE 프레임을 FPGA에게 전송한다. 프레임을 받으면 프레임의 헤더타입을 보고 어떤 프레임인지 조사한 후 각각 그에 따른 처리를 수행한다. IDLE프레임이면 동기가 맞는지 체크하고 버리게 된다. 제어프레임이면 스위치 포트의 상태를 내부 레지스터에 저장하여 Ingress방향의 프레임 흐름제어에 사용하도록 하고, 또한 OQG정보를 생성한다. 데이터 프레임이면 페이로드에서 패킷만 꺼내어 DPRAM에 저장한 후 64바이트 패킷이 모두 입력되면, 모두 Asynchronous FIFO에 입력한다. 그리고 UDASL칩의 상태에 따라 레지스터 정보를 보고 OQG 정보를 갱신하여 패킷을 전송한다. 만약 UDASL이 패킷을 받을 수 없게 되면 Full신호를 Ingress CSIX에게 알려 데이터 전송시 Redey비트를 '0'으로 전송하여 스위치로부터 다음 프레임들이 전송되지 않도록 한다.

V. 결 론

다양한 광대역 서비스들을 가입자들에게 효율적으로

제공하기 위해서는 2.5Gbps급 가입자 정합 모듈 개발이 요구되며, 다양한 가입자 인터페이스 및 서비스를 효율적으로 지원할 수 있는 기술이 필요하다. 이런 기능들을 수행하기 위해 IBM NP4GS3 네트워크 프로세서를 사용한다. 그러나 IBM NP4GS3는 표준화된 CSIX-L1 스위치 인터페이스를 지원하지 않고 있다.

본 논문에서는 IBM의 NP4GS3 네트워크 프로세서가 표준화된 CSIX-L1 스위치 인터페이스와 정합할 수 있도록 IBM의 UDASL칩과 UTOPIA-L3/CSIX-L1 FPGA 변환모듈을 이용하여 2.5Gbps 가입자 정합 모듈의 스위치 정합구조 및 효율적인 패킷처리 및 패킷변환 방법을 제시했다. 또한 CSIX-L1 제어 프레임에서 우선 순위별로 OQG정보를 만들어 네트워크 프로세서가 우선순위 기반 VOQ를 수행 할 수 있도록 구조를 설계하였다.

본 논문에서 제시한 구조는 IBM네트워크프로세서 뿐만 아니라 UTOPIA-L3 인터페이스를 지원하는 네트워크 프로세서를 UTOPIA-L3/CSIX-L1 FPGA 변환모듈을 이용해 효율적으로 표준 스위치 인터페이스와 정합을 제공한다.

참 고 문 헌

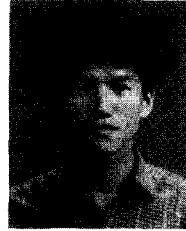
- [1] IBM Doc. "IBM Packet Routing Switch Serial Interface Converter Datasheet", January 14, 2002.
- [2] IBM Doc. "IBM Power Network Processor NP4GS3 Datasheet" Feb 25, 2002.
- [3] IBM Doc. "IBM Packet Routing Switch PRS28.4G Version 1.7 Datasheet", February 6, 2001.
- [4] VITESSE Doc. "GigaStream Intelligent Switch Fabric Design Manual Rev2.1", 2001.
- [5] IBM Doc. "PowerPRS C-192 Common switch interface", 2001.
- [6] NP Forum. "Common Switch Interface Specification-L1", August 5, 2000.
- [7] Linley Gwennap, Bob Wheeler. "A Guide to Network Processors", 2000.
- [8] 김기민, 이형섭, 김용태, 서한석. "CSIX와 스위치 인터페이스를 위한 프로토콜 변환기 설계", NCS2000, 2000
- [9] ATM Forum. "UTOPIA-L3 Physical layer Interface", af-phy-0136.000, Nov, 1999

저 자 소 개



金光玉(正會員)

1999년 : 조선대학교 정보통신공학과 학사. 2001년 : 전남대학교 전자공학과 석사. 2001년~현재 : 한국전자통신연구원 네트워크연구소 연구원 근무. <주관심분야 : 네트워크 프로세서응용, 트래픽 스케줄링 기술>



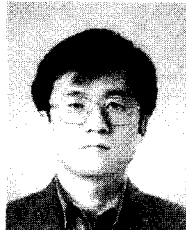
崔炳哲(正會員)

1987년 : 한양대학교 전자공학과 학사. 1997년 : 한남대학교 전자공학과 석사. 1997년~현재 : 부산대학교 컴퓨터공학과 박사과정, 1987년~1993년 : 삼성전자통신연구원 주임연구원 근무, 1993년~현재 : 한국전자통신연구원 네트워크연구소 선임연구원 근무, <주관심분야 : 고속통신망, IP록업, 네트워크프로세서응용>



崔昌植(正會員)

1997년 : 부산대학교 컴퓨터공학과 학사. 1999년 : 부산대학교 컴퓨터공학과 석사. 1999년~현재 : 한국전자통신연구원 네트워크연구소 연구원 근무. <주관심분야 : H/W패킷 포워딩 제어, 임베디드 프로그램>



郭東鎔(正會員)

1983년 : 동국대학교 전산학과 공학사. 1985년 : 동국대학교 전산학과 공학석사. 1998년~현재 : 한국정보통신대학교 공학부 공학박사, 1985년~현재 : 한국전자통신연구원 네트워크연구소 책임연구원 근무, <주관심분야 : 트래픽 스케줄링, 네트워크 프로세서응용>



朴完基(正會員)

1991년 : 충남대학교 전자공학과 학사. 1993년 : 충남대학교 전자공학과 석사. 2001년~현재 : 충남대학교 전자공학과 박사과정. 1993년~2000년 : 국방과학연구소 연구원 근무. 2000년~현재 : 한국전자통신연구원 네트워크연구소 선임연구원 근무. <주관심분야 : MPLS, Gigabit Ethernet, EPON>