

# IP data packet 을 처리하기 위한 AAL type5 프로토콜 프로세서 구현

박재현\*, 최명렬\*  
\*한양대학교 전자전기제어계측공학과  
e-mail : jhpark@asic.hanyang.ac.kr

## Implementation of AAL type5 protocol processor for processing of IP data packet

Jae-Hyeon Park\*, Myung-Ryul Choi\*  
\*Dept. of EECE, Hanyang University

### 요 약

본 논문에서는 ATM 망에서의 통합 네트워크 구현을 위한 IP data packet 를 처리하기 위한 AAL type5 프로토콜 프로세서를 설계 및 구현하였다. AAL 계층의 중요 기능들은 ITU-T Recommendations I.363 과 I.363.5 에 근거하여 설계하였다. AAL 계층의 주요한 역할은 데이터의 Segmentation 및 셀의 Reassembly 를 하는 것으로, Segmentation 과정에서는 상위 계층의 연속적인 데이터를 Segmentation 하여 53-byte 크기의 ATM 셀을 구성하는 기능이다. Reassembly 과정에서는 들어오는 셀들을 연속적인 데이터로 만들어 AAL 계층 보다 상위 계층으로 전달하는 것이다. 이 과정에서 셀의 Header 를 확인한 후 crc-32 를 통한 오류 검정을 거치게 되며, 데이터에 오류가 있을 경우에는 해당 셀을 버리고 오류가 없을 시에만 상위 계층으로 전달한다. 본 논문에서 구현한 AAL Type 5 프로세서는 향후 모든 Type 의 data 를 수용하는 칩 개발에 유용할 것으로 사료된다. 본 논문에서 원할한 테스트를 위해 데이터의 loop back 신호 DLB 를 사용했다. VHDL 해석기로는 Synopsys 사의 VHDL Analyzer 를 사용하였고, Design Compiler 로 회로를 합성하였다.

### 1. 서론

최근 통신망의 발전과 더불어 통신 사용자의 다양한 서비스 욕구를 만족시킬 수 있는 종합 정보 통신망(B-ISDN)은 성숙한 고속 전송기술· 교환기술· 신호처리기술· 컴퓨터기술· 소프트웨어기술 등을 활용해 각종 광대역 서비스를 통합해 제공할 수 있는 디지털 네트워크를 구축하는 데 있다. 이와 같은 기본원칙에 입각해 B-ISDN 을 실현하는 방안으로 강구된 것이 ATM(Asynchronous Transfer Mode) 통신방식으로 ATM 은 비동기식 시분할 다중화를 사용하는 특수한 형태의 패킷형 전달방식이다. B-ISDN 에서는 일정한 크기를 갖는 패킷의 연속적인 흐름으로 서비스 정보가 전달되는데, 이 고정된 크기의 패킷을 ATM 셀이라고 한다. 따라서 입력되는 정보들은 AAL 계층에서 일정한 크기(53 바이트)로 절단된 후 ATM 셀에 매핑되고, 이어서 다른 ATM 셀들과 비동기식 시분할 다중화

(ATDM : Asynchronous Time Division Multiplexing)함으로써 내부 전송신호가 만들어지는 것이다. ATM 방식은 사용하면 성질이 다른 각종 서비스(음성전화서비스· 데이터전송서비스· 영상전송서비스· 케이블 TV 서비스 등)를 통합할 수 있다. ATM 방식은 체계적이고 융통성 있는 정보전달을 위해 계층화된 프로토콜 기준 모형을 규정하였다. 이때 규정되는 프로토콜 계층은 물리계층· ATM 계층· ATM 적용계층(AAL:ATM Adaption Layer)과 상위계층 등인데, AAL 은 서비스 신호들을 ATM 셀에 매핑시키는 기능을 수행하고, ATM 계층은 셀 헤더 관련기능을 수행하며, 물리계층은 셀들을 전송 비트열로 바꾸어 전달하는 기능을 한다. 이 과정에서 기존의 AAL(ATM Adaption Layer)칩을 고속 대용량 통신 서비스에 적합하도록 하는 AAL type 5 칩의 개발은 제 3 세대 이동 통신인 IMT2000 네트워크에서 기지국, 제어국, 교환국 간의 음성 패킷들을 실시간으로 처리하면서 망 자원의 활용도를 높이기

매우 적합한 프로토콜 구조를 가지고 있다. 또한, 유선구간의 망 형상 및 프로토콜은 서비스의 품질, 네트워크 구축 및 유지에 있어서 매우 중요한 요소가 된다. 이러한 AAL type 5 시스템의 구현은 IMT2000 통신망의 핵심 기술이라 할 수 있다.

본 논문에서 구현한 AAL type 5 프로토콜 프로세서는 ITU-T Recommendations I.363 과 I.363.5 에 따라 구현하였다[1][2]. 본 논문의 구성은 다음과 같다. 2 장에서는 AAL type 5 의 기능에 대한 설명 및 각각의 블럭에 대한 구현 설명과 특징들을 서술하였다. 그리고 마지막 3 장에서는 결론과 향후에 진행방향 및 연구과제에 대하여 간단하게 기술하였다.

## 2. AAL type 5

AAL type 5 의 구조는 ITU-T 표준 I.363.5 에 규정되어 있는 CS(Convergence Sublayer) 와 SAR(Segmentation and Reassembly Sublayer)라는 두개의 하부구조로 구성되며, 다양한 크기의 IP data packet 을 CLAD(Cell Assembly and Disassembly) 방법으로 여러개의 ATM 셀로 assembly 을 한다. 또한, CPU 가 외부 메모리에 전송할 packet 을 쌓은 후에 AAL type 5 segmentation 블럭에서 ATM 셀로 전달되고 반대로 수신된 ATM 셀로부터 reassembly 과정을 거쳐 외부 메모리에 쌓인다. 이후 CPU 가 이를 읽어간다. SAR 수신 처리 장치에 있어서 ATM 계층으로부터 스타트 신호가 입력되면 48 카운트를 하여 FIFO 제어 신호와 CRC enable 신호 및 완료 신호를 발생하는 카운터 및 제어부와 ATM 계층으로부터 AUU 비트를 입력하여 메시지의 시작과 종료를 판단하는 AUU 처리부, AUU 처리부의 출력과 카운터 및 제어부의 출력에 의해 수신된 메시지의 길이를 계산하는 길이 연산부 카운터 및 제어부의 출력과 AUU 처리부의 출력에 따라 조립 완료 신호를 발생하는 조립 완료 신호 발생부, FIFO 제어 신호에 따라 ATM-FIFO 로부터 데이터를 읽어 AAL-FIFO 에 데이터를 쓰도록 FIFO 들을 제어하는 FIFO 제어부 및 카운터 및 제어부의 출력에 따라 수신된 메시지의 CRC 에러를 검출하는 CRC 처리부로 구성되어있다.

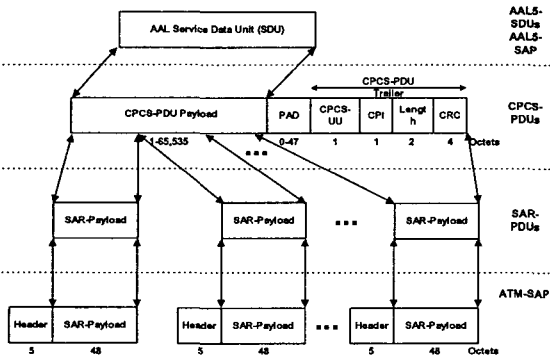


그림 1. AAL type5 Common Part SAR and CS Example

그림 1 은 각각의 AAL Service Data Unit (SDU)이 ATM cell 에 실리는 과정을 보여주고 있다[3].

CPCS packet 와 AAL type 5 cell 의 구조는 그림 2 와 같다. CPCS packet 은 8octets 의 Trailer 와 payload 로 이루어져 있다. Payload 는  $2^{16}-1$  octets 까지 사용이 가능하다. Trailer 는 CPCS-UU(User-to-User Indication : 8bits), CPI(Common Part Indicator : 8bits), Length(CPCS-PDU Length : 16bits), CRC(Cyclic Redundancy Check : 32bits)의 4 가지 필드로 이루어져 있다.

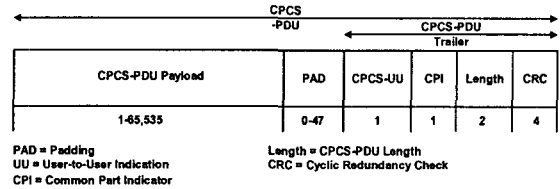


그림 2. Structures of CPCS packet and CPCS-PDU

CPCS packet 이 CPCS\_PDU(CPCS\_Payload Data Unit)에 할당될 때 CPCS-PDU 는 data 의 크기에 따라 알맞게 0-47 octet 의 크기로 Padding 된다.

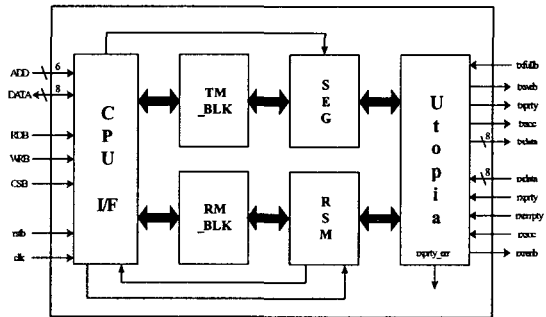


그림 3. AAL type5 프로토콜 블럭도

AAL type5 processor 의 블럭도는 그림 3 와 같다. CPU I/F 블럭은 CPU 와 통신을 통해 데이터를 레지스터에 저장 및 신호를 생성하고 TM\_BLK(TX Message Block), RM\_BLK(RX Message Block)은 전 블럭에서 넘겨진 데이터를 임시로 저장하는 곳이다. segmentation 블럭은 연속적으로 들어오는 1 octet 에서  $2^{16}-1$  octets 의 데이터를 사용자의 용도에 맞게 segmentation 를 담당하며 reassembly 블럭은 헤더의 오류를 검사하며 헤더를 분리한다. 이렇게 분리된 데이터는 사용자가 원하는 데이터로 reassembly 이 된다. Utopia 블럭은 물리 계층과의 연결을 담당하는 인터페이스 부분이다.

### 2.1 CPU I/F 블럭

CPU I/F 블럭의 구성은 그림 4 와 같이 12 개의 레지스터로 이루어져 있다. 각각의 레지스터에는 해당되는 값들이 저장되어 있고, wenb, wr\_done, renb, rd\_done 등의 신호가 생성되는데, 이 신호들은 레지스터 2 번지의 값을 바탕으로 생성된다. AAL type5 processor 가 동

작 상태에서는 CSB 신호가 low 가 되며, 그와 동시에 읽을때는 RDB 가 low, 쓸때는 WRB 가 low 이어야 한다. 그리고 헤더값들은 packet 이 segmentation 및 reassembly 을 위해서 사용되므로 SEG, RSM 블록으로 넘겨진다.

	write	read
cpureg0		version
cpureg1	length&pad&oam	length&pad&oam
cpureg2	msgsel&rd_done&wr_done	rd_avl&wr_avl&msgsel&rd_done&wr_done
cpureg3	tm_data_in	tm_data_out
cpureg4	h1	h1
cpureg5	h2	h2
cpureg6	h3	h3
cpureg7	h4	h4
cpureg8	h1mask	h1mask
cpureg9	h2mask	h2mask
cpureg10	h3mask	h3mask
cpureg11	h4mask	h4mask

그림 4. CPU I/F 레지스터 MAP

## 2.2 Message block

Message block 블록은 송신(Tx), 수신(Rx) 두 부분으로 나누어져 있고, 그림 5 는 message block 의 블록도를 나타낸 것이다.

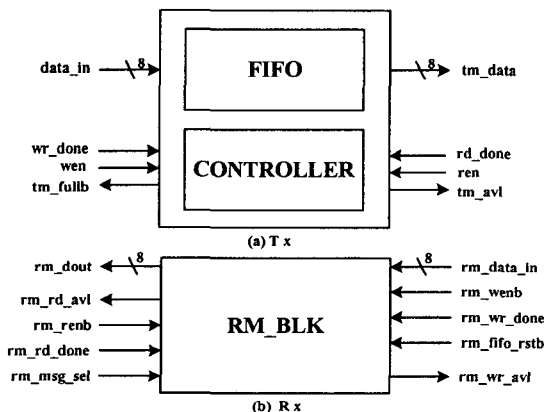


그림 5. MSG\_BLK 의 블록도

우선 Tx 부분은 CPU I/F 블록에서 들어온 데이터를 임시 저장을 했다가 segmentation 블록으로 데이터를 넘겨주는 기능을 하며, 세개의 FIFO 로 이루어져 있다. Tx 부분에서는 데이터의 길이를 나타내는 tm\_length 와 padding 유무를 나타내는 tm\_pad, 그리고 현재의 cell 이 데이터인지 아니면 oam cell 인지를 구별하는 tm\_oam 신호가 생성된다. Tx 에서 데이터를 읽어 들이기 위해서는 우선 tmfullb 신호가 low 상태가 되어야 한다. 이 신호는 현재 세개의 FIFO 중에 최소 하나의 FIFO 가 비어있는 상태일때 low 신호가 된다. FIFO 선택은 wr\_done 신호의 카운트를 통해서 선택을 한다. Tm\_avl 신호는 tmfullb 신호의 반대 역할로 최소 한

개의 FIFO 에 데이터가 write 되어 있을 때 high 신호가 발생한다. Read 할 FIFO 선택은 rd\_done 신호의 카운트로 이루어 진다. Rx 부분은 Tx 부분과 같은 기능을 담당하는 곳으로 Rx 부분은 reassembly 에서 들어온 데이터를 임시로 보관하는 블록이다. Rm\_blk 은 16 개의 FIFO 로 이루어져 있고 reassembly 블록에서 rm\_fifo\_rstb 신호가 low 로 넘어오면 rm\_blk 블록의 모든 FIFO 에 있는 데이터는 폐기 시킨다. Rm\_fifo\_rstb 신호는 reassembly 에서 데이터가 에러가 발생했을 때 만들어지는 신호이다.

## 2.3 Segmentation block

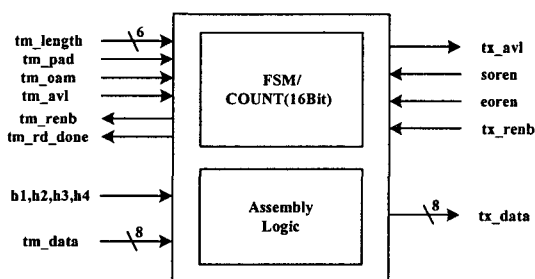


그림 6. Segmentation 블록도

Segmentation 블록도는 그림 6 과 같다. segmentation 블록은 53octets 의 크기로 분해하는 기능을 한다. 그리고 crc\_gen 신호가 high 가 들어오면 CRC-32 를 생성한다. 이렇게 생성된 값들과 읽어 들인 데이터 값을 사용하여 ATM Header 와 payload 로 53 octets 의 cell 을 생성한다.

## 2.4 Reassembly block

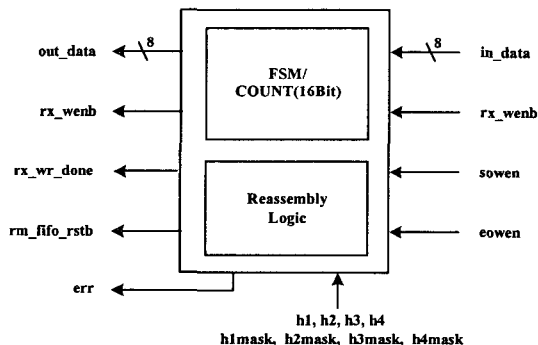


그림 7. Reassembly 블록도

Reassembly 블록도는 그림 7 과 같다. Reassembly 블록은 ATM 계층에서 넘어온 셀에 header 를 제거하여 원래의 packet 으로 만든 기능을 한다. Reassembly 블록은 CPU I/F 블록에서 읽어 들인 Header 값과 HMASK 을 이용하여 Utopia 블록에서 읽어 들인 cell 의 header 값과 비교하여 에러가 발생하면 데이터를 폐기한다. 에러 발생시에는 err 신호가 high 로 되며 rm\_fifo\_rstb

는 low가 되어 rm\_msg\_blk블럭을 RESET 시킨다.

### 2.5 Utopia block

Utopia 블럭도는 그림 8 과 같다. Utopia 블럭은 physical layer 과 연결을 담당하는 블럭이다. 여기에서 사용한 Utopia 블럭은 level 1로서 하나의 physical layer 와 하나의 Utopia 계층의 연결만 가능하다. Utopia 계층도 Rx, Tx 로 나눌수 있다. Rx, Tx 기능은 모두 동일하다. 즉 데이터를 정확하게 읽기 위해서 53 번의 카운트를 한다. 그리고 원활한 테스트를 위해서 DLB(Diagnostic Loopback)신호를 만들었다. 이 신호는 AAL type 5 protocol processor 에의 Utopia 블럭에서 데이터 loop back 을 위한 신호이다.

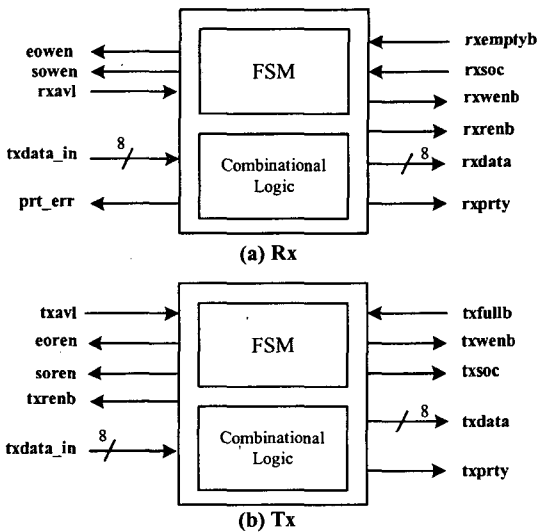


그림 8. Utopia Rx/Tx 블럭도

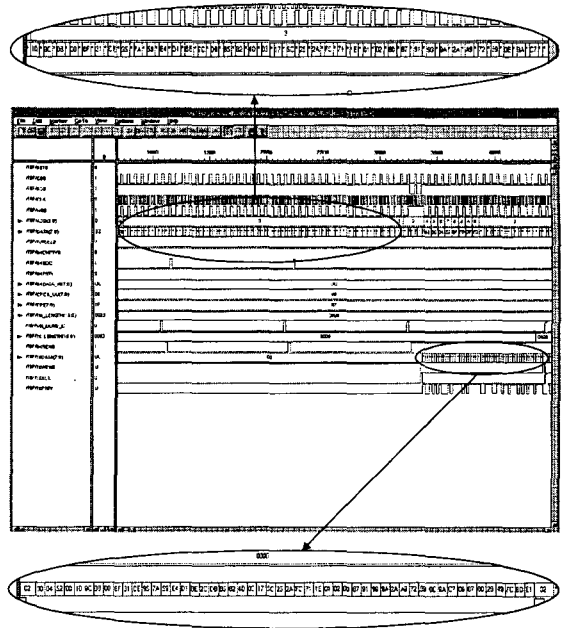


그림 9. AAL type5 simulation result

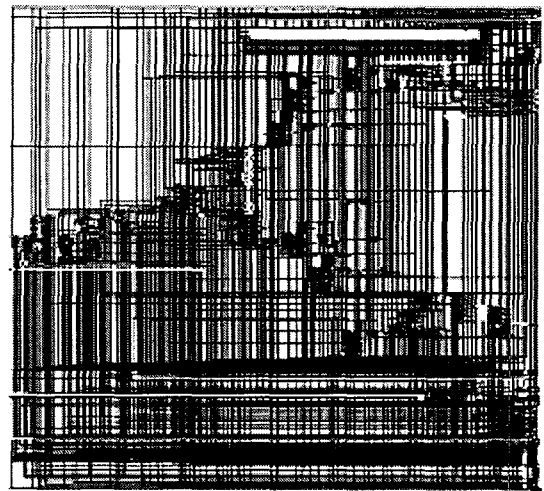


그림 10. AAL type5 Analyzer result

### 3. 결 과 및 향후과제

다양한 크기의 IP data packet 을 처리할 수 있는 AAL type 5 protocol processor 를 VHDL 로 설계하였다.테스트에서는 50 만개의 데이터를 입력으로 사용했다. 그림 9 는 시뮬레이션 결과를 보여주고 있다. 그림에서 윗부분에 있는 송신단의 데이터 입력값과 아래부분에 있는 수신단의 결과값이 같다는 것을 알 수 있다. 즉, 여러가지 다양한 크기의 IP data packet 을 처리하는데 유용한 것을 알 수 있다. 그리고 최종적으로 그림 10 과 같은 합성결과를 나타내었다. VHDL 해석기로는 Synopsys 사의 VHDL Analyzer, VHDL simulator 로는 VHDL Debugger 를 사용하였으며 Design Compiler 로 회로를 합성하였다. 향후에는 어떠한 종류의 데이터에서도 효율적인 처리를 할 수 있도록 AAL type1,2,5 통합칩을 구현할 것이다.

#### 참고문헌

- [1] ITU-T Recommendation I.363 "B-ISDN ATM Adaptation Layer (AAL) Specification", March 1993
- [2] ITU-T Recommendation I.363.5, "B-ISDN ATM Adaptation Layer Specification : Type 5 AAL", August 1996
- [3] David McDysan and Darren Spohn, "ATM Theory and Applications", McGraw-Hill, pp. 366-370, March 2000