

# 개선된 MP3 오디오 전송을 위한 SPH/RPH 모듈 구현

권장우\*, 김수진\*\*, 김익형\*\*, 박부곤\*\*, 우동훈\*\*  
\*동명정보대학교 컴퓨터공학과, \*\* 팜캐스트

## Implementation of SPH/RPH Module for Improved MP3 Audio Streaming

JangWoo Kwon\*, SuJin Kim\*\*, IgHyeong Kim\*\*, BuGon Park\*\*, Dong Hoon Woo\*\*  
\*Dept. of Computer Engineering, Tongmyong Univ. Of Information Technology  
\*\*Palmcast Corporation

### 요 약

최근의 인터넷 음악방송은 MP3 오디오를 기반으로 하는 TCP 프로토콜의 전송방식이 일반적이다. TCP방식의 전송은 HTTP 프로토콜을 이용한 파일 전송방식으로 네트워크의 부하가 급증할 경우 TCP의 특성으로 인해 음악의 끊김 현상이 발생하여 QoS 문제가 발생한다. 본 논문은 실시간 전송방식의 RTP(Real-time Transfer Protocol) 프로토콜을 이용하여 MP3 오디오 기반의 생방송 시스템 개발에 대한 연구로서, 기존의 TCP 방식의 음악의 끊김 현상을 개선하기 위한 모듈 구현을 목적으로 한다. 본 연구에서는 MP3 오디오 전송에 따른 QoS(Quality Of Service) 개선을 위하여 인터리빙 기법을 이용한 SPH/RPH(Send Payload Handler/ Receive Payload Handler) 모듈을 구현하였다.

### 1. 서론

네트워크의 속도와 인터넷의 대중화로 인해 멀티미디어 컴퓨팅 환경이 급속히 성장하고 있으며, 오디오와 비디오를 인터넷으로 실시간 전송하는 생방송이나 주문형 방송과 같이 고급 멀티미디어의 전송이 일반화 되었다. 개인이나 기관은 인터넷을 이용한 사설 방송국을 운영하기 시작하였으며 TV, 라디오와 같은 기존 방송국과 유사한 멀티미디어 매체로 자리를 잡기 시작했다. 특히 오디오를 이용한 인터넷 음악 방송국이 최근 활성화 되고 있는 추세이다. 가장 일반적인 인터넷 오디오 방송 프로토콜은 TCP방식의 HTTP프로토콜[1]을 이용한 실시간 파일 전송방식을 가장 많이 사용하고 있다. 네트워크 속도의 한계와 음악방송을 위한 오디오 품질의 측면에서 MP3 포맷이 가장 많이 사용되고 있으며, 국내의 실시간 오디오 방송국은 Null Soft사

의 Winamp 플레이어와 Shoutcast 서버 소프트웨어를 이용하여 생방송을 수행하는 형태이다[2]. TCP방식의 파일 전송형태의 음악방송은 네트워크에 과부하가 걸릴 경우 청취자는 음악이 끊어지는 현상으로 인해 생방송 청취시 올바른 의미 전달을 받기 어려운 단점이 있다.

본 연구에서는 MP3 데이터의 전송에서 TCP방식의 음악의 끊김 현상을 개선하기 위해서 실시간 미디어 전송을 목적으로 개발된 RTP 프로토콜을 이용한 전송방식을 채택한다[3]. RTP는 UDP 방식의 전송으로 TCP와는 달리 QoS(Quality Of Service)를 보장하지는 못하지만 실시간 전송의 장점이 있으며, QoS를 보장하기 위해서 RTCP 프로토콜을 이용한다[4]. RTP만을 이용할 경우 네트워크 과부하가 발생하면 RTP 패킷의 손실로 인해 음질 저하 현상이 유발되게 된다. 본 논문에서는 음악의 단절 현상을 막고

음질 저하 문제를 해결하기 위한 방법으로 RTP 프로토콜 이용할 경우 인터리브 처리를 통해서 QoS 보장 문제를 해결한다. 인터리브 처리에는 크게 두 가지 구현 방법이 존재하는데, 하나는 비트열을 고정길이의 블록에 차곡차곡 채워 넣은 후 그 블록에서 읽어내는 순서를 바꾸어 출력하는 비트 인터리브'이고, 또 하나는 바이트마다 서로 다른 지연량을 주어 비트열의 순서를 변화시키는 길쌈 인터리브' 방식이다[13]. 본 논문에서는 MP3의 재생 단위인 MP3프레임을 인터리빙(Interleaving)하여 패킷 손실이 발생할 경우 연속적인 MP3프레임 손실을 막는 방식인 후자의 방법을 택하여 구현하였다[5].

본문의 구성은 MP3의 RTP 전송시 음질 저하를 개선하기 위한 인터리빙 기법[5]에 대해서 소개하고 음악 끊김 현상과 음질 저하 문제를 해결하기 위한 방법으로 MP3의 SPH/RPH(Send Payload Handler/Receive Payload Handler) 기법에 대해서 설명한다. MP3의 SPH/RPH의 구현 모듈을 설명하고, 실험 및 응용 시스템의 적용방법을 살펴본 후 본 연구에 대한 결론을 맺는다.

## 2. MP3의 SPH/RPH 모듈

### 2.1 RTP (Real-time Transfer Protocol)

RTP(Real-time Transfer Protocol)는 오디오, 비디오의 데이터를 실시간으로 멀티캐스트(Multicast) 및 유니캐스트(Unicast) 방식으로 전송하는데 적합한 전송 프로토콜이다[3]. RTP는 네트워크 자원의 예약을 하지 않으며 실시간 전송 서비스에 대한 QoS 역시 보장하지 않는다. QoS 보상은 데이터의 전송 감시 및 데이터 인식 등의 최소한의 제어를 지원하는 제어 프로토콜인 RTCP(Real-time Transfer Control Protocol)에 의해 결정된다[2].

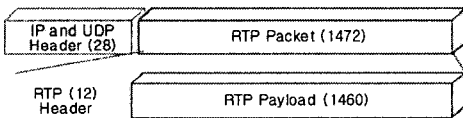


그림 1. RTP Packet의 구성

그림 1은 RTP Packet의 구조를 보여주고 있으며 RTP 패킷은 RTP 헤더와 RTP 페이로드로 구분한다. RTP 패킷은 UDP방식으로 전송되고 한번에 전송할 수 있는 미디어 자료를 RTP 페이로드(Payload)라고 부르며, 최대 크기는 1460Bytes로 제약을 받는다. 각 미디어의 종류에 따라 RTP 전송을 수행할 수 있는 페이로드 구성에 대한 표준안이 IETF에 RFC문서로 존재하며 본 연구에서는 RFC3119

방식의 인터리빙 기법을 이용한다[5].

### 2.2 MP3 Frame 구조

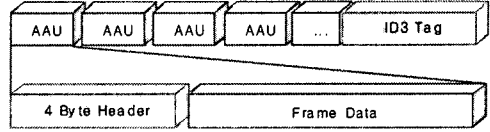


그림 2. MP3 파일과 MP3 Frame의 구성

그림 2는 MP3의 파일 구조와 하나의 MP3 프레임에 대한 구조를 나타낸다. AAU(Audio Access Unit)는 MP3의 프레임을 가리키며, 재생의 기본 단위가 된다. 각각의 프레임 데이터는 RTP 패킷으로 감싸져서 실시간 전송된다.

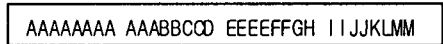


그림 3. MP3 파일과 MP3 Frame의 구성

그림 3은 4Bytes로 구성되는 MP3 프레임 헤더(MP3 Frame Header)를 표현 한 것으로 각각의 비트 정보를 이용하여 MP3의 프레임 크기와 재생시간 정보 등을 알 수 있다. IETF의 RFC3119 방식의 인터리빙을 이용한 전송을 수행하기 위해서 MP3 프레임 헤더의 정보 중 A'로 표시되는 11 비트의 프레임 싱크 비트(Frame Sync bit)를 이용하여 된다. 프레임 싱크 비트는 재생을 위한 동기 비트로 항상 1'로 세팅 되어 있으나 RTP를 이용하여 전송할 때는 11bit를 이용하여 사이클 정보와 인덱스 정보로 변경하여 전송하게 된다[5].

아래는 RTP 캡슐화를 위한 패킷 크기와 RTP 전송 시간에 사용될 MP3의 프레임 크기 및 프레임 시간을 계산하는 식을 설명한다.

#### 가. MP3 프레임 크기 계산

Layer 1:  

$$nFrameSize = (12000 * nFrameBR / nSampleRate + padding) * 4;$$

OTHER Layer:  

$$nFrameSize = 144000 * nFrameBR / nSampleRate + padding;$$

#### 나. MP3 프레임 시간 계산

$$nDuration = FrameSize * 8 / (BitRate * 1000);$$

### 2.3 MP3 Interleaving

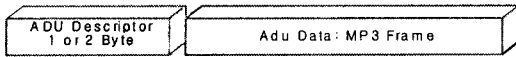


그림 6. ADU(Application Data Unit) 구조

그림 6은 ADU(Application Data Unit)를 설명하고 있다. RFC 3119 방식의 인터리빙(Interleaving)을 수행하고자 할 때 MP3 Frame의 앞쪽에 추가적인 바이트(Byte)를 덧붙여서 인터리빙 정보를 저장하는 공간을 확보하는 방법이다. ADU는 ADU Descriptor와 ADU Size로 분리되며, ADU는 MP3 Frame의 크기에 따라 1Byte 혹은 2Byte 두 가지로 나누어진다[5].

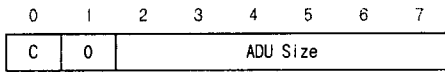


그림 7. ADU Descriptor 구조

그림 7은 ADU Descriptor를 설명하고 있는 것으로 C bit는 Continuous 정보이며, C bit 다음의 Bit가 0으로 세팅되면 ADU Descriptor 크기가 1Byte로, 1로 세팅되면 2Byte로 크기가 정해진다. ADU Descriptor의 Byte 크기가 가변적인 것은 네트워크 전송에서 데이터 크기를 최대한 줄이기 위함이다[5].

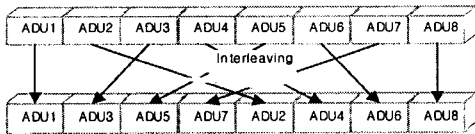


그림 8. MP3 Frame의 Interleaving

그림 8은 MP3로부터 ADU를 생성하고 난 뒤의 인터리빙(Interleaving)을 수행하는 방법을 설명하고 있다. 8개의 ADU를 기준으로 홀수와 짝수로 나눈 뒤 인터리빙을 수행하고 있다. Interleaving 방식은 ADU Frame을 그룹 단위로 생성하고 그림 8과 같이 존재할 때, ADU Index No, Cycle No와 같은 방식으로 그룹화 하면 아래와 같이 묶을 수 있다.

Cycle 0:  
 [(1,0),(3,0),(5,0),(7,0),(2,0),(4,0),(6,0),(8,0)]  
 Cycle 1:  
 [(1,1),(3,1),(5,1),(7,1),(2,1),(4,1),(6,1),(8,1)]  
 ...  
 Cycle n:  
 [(1,n),(3,n),(5,n),(7,n),(2,n),(4,n),(6,n),(8,n)]

RTP 전송을 위해서 RTP 페이로드는 최대한 캡슐화 할 수 있는 ADU Frame을 한번에 묶어서 생성하고 RTP로 전송하게 된다.

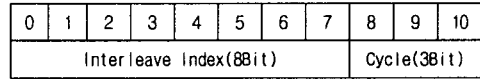


그림 9. MP3 Frame Sync Bit를 이용한 Cycle과 Index

그림 9는 MP3 Frame Sync Bit를 인터리빙을 위한 용도로 활용할 때 사이클과 인덱스 번호 영역을 나타낸 것이다. 8Bit는 인터리빙 인덱스로 사용하고, 3Bit는 사이클 번호로 이용한다[5].

### 2.4 MP3 RTP Payload

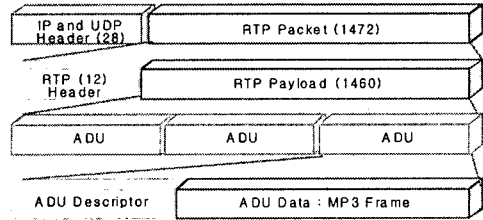


그림 10. MP3 RTP Payload 포맷

그림 10은 MP3를 RTP로 전송하기 위한 Payload를 설명한다. MP3를 ADU로 변환하고 인터리빙을 수행한 후 RTP Payload의 최대 크기인 1460Bytes 이하 크기 만큼의 ADU를 묶어서 하나의 MP3 RTP Payload로 구성한 것이다. MP3 RTP Payload는 하나 혹은 여러 개의 ADU가 이용 될 수 있다[7].

### 2.5 MP3 SPH(Send Payload Handler) 설계

MP3 SPH(Send Payload Handler)는 기존의 MPEG System 전송을 위한 IETF의 RFC2250[6]의 PAD(Packet Assembler/ Disassembler) 모듈을 개량한 것이다[8]. 생방송을 위해 MP3 CODEC에서 출력되는 MP3를 전송하는 방식과 AOD(Audio On Demand)를 위해서 MP3 파일을 통한 전송 방식으로 나눌 수 있다.

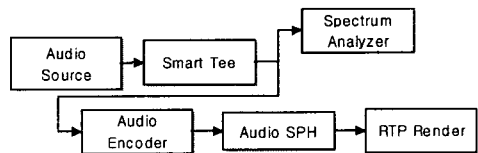


그림 11. MP3 SPH를 위한 Direct Show Filter모듈

그림 11은 생방송 클라이언트를 위한 MP3 SPH 모듈로 Direct Show[10] 기반에서 동작하도록 하기 위한 구성도이다. Direct Show는 컴포넌트 기반의 Filter들을 연결할 수 있는 구조이다. RTP SPH Filter는 MP3를 인터리빙하여 RTP Payload를 구성하는 모듈을 말한다.

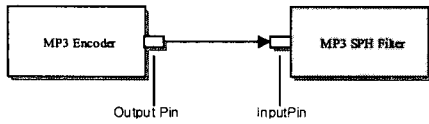


그림 12. MP3 SPH를 CODEC과 연결

그림 12는 CODEC에서 출력되는 MP3 Frame 자료를 SPH Filter로 연결하는 방법을 보여주고 있으며, SPH는 변환 필터 형태이다[11]. RTP SPH Filter는 RTP Render Filter를 통해서 RTP 네트워크로 전송하게 된다.

### 2.6 MP3 RPH(Receive Payload Handler) 설계

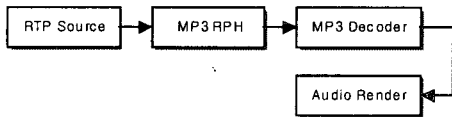


그림 13. MP3 RPH를 CODEC과 연결

그림 13는 RTP로부터 전송되는 패킷을 수신 받아 재생하는 RPH모듈의 구성도를 나타낸 것이다. MP3 RPH Filter는 MP3 Decoder로 원래의 MP3 Frame 자료를 넘겨주고 재생하게 된다.

### 3. 구현 및 고찰

MP3의 SPH/RPH 모듈을 클라이언트에 적용하기 위해서 Microsoft의 DirectShow기반의 컴포넌트인 Filter[10]를 이용하여 개발하였다. 서버측의 모듈은 PAD 모듈[8]에서 RFC3119방식의 인터리빙 모듈을 추가하여 장착하였다.

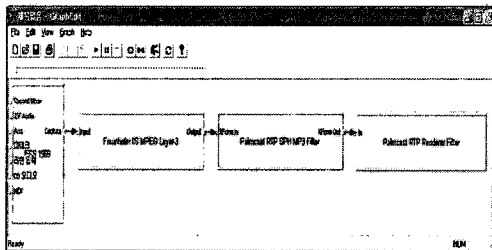


그림 14. MP3 SPH를 위한 Filter Graph Editor 화면

그림 14는 Direct Show Filter Graph Editor를 이용하여 SPH를 연결하여 테스트하는 화면을 나타낸 것이다. 클라이언트측은 타 방송 소프트웨어나 타 종류의 MP3 CODEC과 쉽게 결합 할 수 있는 방식이다.

본 연구에서 개발된 모듈은 현재의 인터넷 환경뿐만 아니라 PDA 등의 Post PC 계열의 디바이스에 장착되어 광범위한 응용 범위를 가진다.

### 4. 결론

본 연구에서는 MP3의 효율적인 전송을 위한 SPH/RPH 모듈을 구현하였다. RFC3119방식의 인터리빙을 이용하여 네트워크의 과부하가 발생할 경우 음질 끊김 현상과 음질 저하 현상을 개선할 수 있었다. 특히 인터넷을 이용한 오디오 생방송이나 과부하의 주문형 오디오 방송에서 효과적으로 활용할 수 있다.

향후 연구 계획으로는 개발된 모듈을 다양한 환경에서의 테스트와 함께 안정성 검증을 위한 실험적 연구가 필요하며, 인터리빙의 인덱스 크기를 변경하여 네트워크의 부하에 따른 인터리빙 인덱스 크기나 방식을 조정하는 방법에 관한 연구와 함께 비상업적인 CODEC인 Vorbis의 Orgg CODEC[12]을 위한 인터리빙에 대한 연구가 진행될 예정이다.

### [참고문헌]

- [1] HTTP Hyper Text Transfer Protocol  
<http://www.w3c.org/Protocols/>
- [2] NullSoft: Shoutcast Server & Winamp Player  
<http://www.nullsoft.com, http://www.shoutcast.com>
- [3] IETF RFC 1889 RTP: "A Transport Protocol for Real-Time Application", January 1996
- [4] IETF RFC 1890 RTCP: "RTP Profile for Audio and Video Conferences with Minimal Control", January 1996
- [5] IETF RFC 3119: "A More Loss-Tolerant RTP Payload Format for MP3 Audio", R. Finlayson, June 2001
- [6] IETF RFC 2250: "RTP Payload Format for MPEG1 / MPEG2 Video", RFC 2250, January 1998
- [7] W.Richard Stevens, TCP/IP Illustrated, Volume 1, Addison Wesley, pp.148~151, 1994.
- [8] RTP기반 실시간 MPEG Audio 전송을 위한 PAD모듈 설계 및 구현", 2002년 한국멀티미디어학회 추계학술발표논문집 pp.771~775
- [9] TOCP와 Multi-Thread를 이용한 RTSP Media Server 개발", 2002년 한국멀티미디어학회 추계학술발표 논문집 pp.767~770
- [10] Direct Show: Microsoft DirectShow™ SDK  
<http://www.microsoft.com/directX>
- [11] 변환필터: DirectShow 멀티미디어 프로그래밍, 신화선, 한빛미디어 2002, pp.572~623
- [12] Vorbis.com Open Free Audio  
<http://www.vorbis.com/>
- [13] 정제창 역, "그림으로 보는 응용 MPEG," 교보문고, pp.143~152,1997