

# 리눅스 환경을 위한 IEEE1394 기반 멀티미디어 스트리밍

조상길, 최창열  
강원대학교 컴퓨터·정보통신공학과

## Implementation of IEEE1394 based Multimedia Streaming for Linux Environment

Sang-Gil Cho, Chang-Yeol Choi  
Dept. of Computer, Information and Communications Eng.  
Kangwon National University

### 요약

본 논문은 리눅스 환경에서 IEEE1394의 동시 전송 프로토콜을 이용하는 멀티미디어 스트리밍 시스템의 구현에 대해 기술한다. 구현된 시스템은 DV 데이터의 스트리밍과 재생, 저장은 물론 로컬 재생과 장치 제어 기능을 갖는 사용자 프로그램과 dv1394 디바이스 드라이버로 구성된다. 스트리밍을 위한 메모리할당, 버퍼관리, 전송 데이터 형식 선언, IEEE1394 헤더 및 CIP 처리 기능을 가진 dv1394 드라이버를 사용하여 드라이버와 OHCI1394 카드 사이의 스트리밍에 대한 투명성을 제공함으로써 기존의 스트리밍 시스템이 가졌던 응용 프로그램 개발의 복잡성을 줄였다. 이 시스템은 실시간 원격 모니터링이나 화상 회의 시스템, 그리고 비선형 비디오 편집 시스템에 응용될 수 있다.

### 1. 서론

홈 네트워크에서 디지털 오디오와 비디오 장비를 연결하여 대용량의 멀티미디어 데이터를 송수신하는 연구가 활발히 진행되고 있다[3][4]. 홈 네트워크와 HDTV, 디지털 캠코더와 같은 AV 정보가전기기, 위성방송, 디지털 비디오 방송(Digital Video Broadcasting : DVB), 인터넷 방송 등 다양한 스트리밍 서비스들이 결합되면서 멀티미디어 스트리밍에 대한 관심이 더욱 증가하고 있다. 홈 네트워크에서 멀티미디어 스트리밍을 지원하기 위해서는 대용량의 멀티미디어 데이터를 손실 없이 전송할 수 있는 높은 대역폭과 고속의 전송속도를 제공하는 프로토콜이 필요하며 이를 위해 IEEE1394 표준이 제안되었다.

IEEE1394는 100, 200, 400Mbps의 높은 대역폭과 비동기 전송 방식 뿐 아니라 동시 전송 방식을 제공함으로써 동시 채널을 이용한 실시간 멀티미디어 데이터 전송이 가능하다[1]. 그리고 IEC61883 표준의 CIP(Common Isochronous Packet)를 적용하여 디지털 장비 간의 다양한 멀

티미디어 데이터를 스트리밍 할 수 있도록 정의하고 있다. IEC61883-1은 디지털 캠코더의 데이터 압축 형식인 DV와 같은 AV 스트리밍을 위한 프로토콜을 정의하고[5], IEC61883-4는 DVB, 위성방송, DVD, VCD에서 사용하는 데이터 압축 형식인 MPEG2-TS 스트리밍에 대한 프로토콜을 정의한다[6].

IEEE1394 기반의 멀티미디어 데이터 스트리밍 시스템의 대표적인 연구로는 Linux1394[7]의 DV 스트리밍 시스템과 NEC의 MPEG over 1394[8], TI의 ATV(Advanced TV : HDTV) 인터페이스로서 IEEE1394를 적용하는 연구[9]가 있다. Linux1394는 리눅스 플랫폼 상에서 video 1394 디바이스 드라이버를 사용하여 IEEE1394로 DV 데이터를 스트리밍하기 위한 리눅스 커널 모듈과 libraw1394와 같은 사용자 API를 제안하고 구현하였다. NEC는 시스템 칩(System IC)상에 스트리밍 기능을 구현한 셋톱박스를 사용하여 IEEE1394를 통해 MPEG2-TS를 TV로 재생할 수 있도록 하였다. TI는 ATV의 멀티미디어 스트리밍 인터페이스로서 IEEE1394를 적용하는 시스템을 제안하였다. 이 시스템은 EIA에서 제안한 ATV 프로토콜인

HyperLynx[10]를 기반으로 하며 멀티미디어 스트리밍은 HyperLynx 프로토콜의 CIP 층을 통한 MPEG2-TS 데이터의 전송으로 이루어진다.

리눅스 환경을 위한 기존의 IEEE1394 멀티미디어 스트리밍 시스템은 데이터 송수신 기능만을 제공하는 video 1394 디바이스 드라이버를 사용하여 메모리 할당, 버퍼관리, 전송 데이터 형식 선언, CIP 헤더 및 IEEE1394 헤더 처리 등 스트리밍에 필요한 작업은 사용자 영역에서 이루어짐으로 응용프로그램의 개발이 복잡해진다. 이 문제를 해결하기 위해 NEC는 스트리밍을 위한 작업을 시스템 칩상에 구현하였고 TI는 기존의 HyperLynx 프로토콜 계층에 CIP 층을 추가하였다.

본 논문은 리눅스 환경에서 IEEE1394에 연결된 장치들 간에 멀티미디어 데이터를 스트리밍 할 수 있는 시스템의 구현 내용을 다루며, video1394가 제공하는 데이터 송수신 기능에 사용자 처리 작업들을 추가한 dv1394 디바이스 드라이버를 사용한다. 제안된 시스템은 DV 스트리밍의 재생, 저장, 송수신 기능을 제공하는 사용자 프로그램과 스트리밍을 위한 dv1394 디바이스 드라이버로 구성된다. 그리고 기존의 스트리밍 시스템과는 달리 IEEE1394의 물리층, 링크층 사이의 스트리밍은 디바이스 드라이버가 사용자에게 투명하게 기능을 대신해줌으로써 물리층, 링크층의 멀티미디어 스트리밍에 대한 구현 부담이 줄어든다.

1절에 이어 2절에서는 스트리밍 시스템의 설계 및 세부 구성에 대해 기술한다. 3절에서는 2절의 설계에 기초하여 시스템을 구현하고 4절에서 결론과 향후과제에 대해 다룬다.

## 2. 멀티미디어 스트리밍 시스템 설계

### 2.1 시스템 구성 및 동작

리눅스 환경에서 IEEE1394를 통한 멀티미디어 데이터 스트리밍을 위한 시스템 구성은 그림 1과 같다. 리눅스가 탑재된 컴퓨터 2대는 OHCI1394 어댑터를 통해 IEEE1394에 연결되며 각각 서버와 클라이언트 기능을 수행한다. 그리고 외부로부터 입력을 받아 DV 형식의 데이터를 제공하는 디지털 캠코더와 데이터를 저장하는 외장형 하드디스크나 시디 R/W는 IEEE1394에 직접 연결된다. 리눅스 클라이언트와 여기에 연결된 TV는 1394 저장장치에 저장된 데이터는 물론 스트리밍 데이터를 저장과 동시에 재생한다. 즉 리눅스 서버가 외장형 저장장치에 저장된 DV 파일을 1394 네트워크로의 방송을 통해 클라이언트나 TV에서 재생이 가능하다.

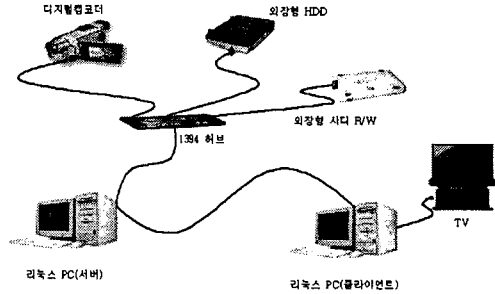


그림 1. IEEE1394 기반의 멀티미디어 데이터 송수신 시스템 구성

### 2.2 리눅스 시스템 구조

멀티미디어 데이터의 송수신 기능을 바탕으로 스트리밍을 하기 위한 리눅스 시스템은 그림 2와 같이 구성되어, 멀티미디어 데이터의 송수신은 물론이고 여러 IEEE1394 장치들의 연결과 제어, SBP(Serial Bus Protocol)2를 따르는 외장형 저장장치의 연결을 가능하게 한다.

IEEE1394를 통한 데이터 전송은 크게 비동기 전송과 동기 전송으로 나뉘어지며, 그림 2의 최하위에서 물리적 연결을 지원하는 OHCI1394 카드 내에 있는 비동기 전송 버퍼와 동기 전송 버퍼를 통해 구분된다.

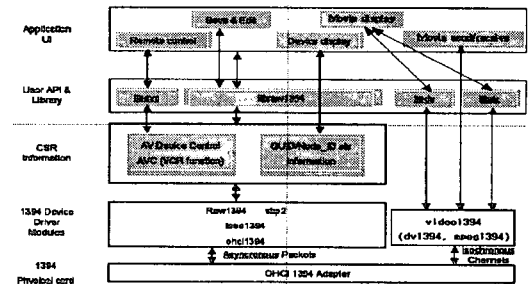


그림 2. 데이터 송수신을 위한 리눅스 시스템 구조

비동기 데이터 전송은 대부분 IEEE1394 장치들을 제어하기 위한 명령어 전달에 사용되며 이벤트 발생을 통해 통신이 이루어진다. 비동기 전송을 위한 드라이버 계층은 커널 스페이스 모듈로서 최하위의 OHCI1394 카드와 인터페이스하는 ohci1394, 1394 서브시스템의 핵심으로서 상하위 드라이버의 관리, 트랜잭션 처리, 그리고 이벤트 트리거링 메카니즘을 제공하는 ieee1394, 사용자 애플리케이션에게 raw1394 버스로의 접근 인터페이스를 제공하는 raw1394, 그리고 IEEE1394를 SCSI처럼 PC의 외부 인터페이스로 사용할 수 있게 하여 저장 장치를 연결하는 SBP2로 구성된다.

동시성 데이터 채널은 오디오, 비디오와 같은 멀티미디어 데이터의 송수신을 위해 사용된다. 사용자 프로그램의 전송 요청에 의해 DV 데이터는 dv1394 디바이스 드라이버를 통해 OHCI1394 카드로 동시성으로 전송된다. dv1394는 IEEE1394 헤더와 CIP를 포함하는 옵션을 자체적으로 지원하므로 사용자에게 dv1394 드라이버와 OHCI1394 카드 사이의 스트리밍에 대한 투명성을 제공한다. 즉 CIP 층을 별도로 두거나 사용자가 직접 선언해 줄 필요가 없다. dv1394 드라이버는 그림 3과 같이 메모리와 버퍼의 관리, 스케줄링, 1394 버스의 동시 채널과 사용자 프로그램 사이에 멀티미디어 데이터를 스트리밍하기 위한 DMA 메카니즘, 프레임 동기화 기능을 제공한다.

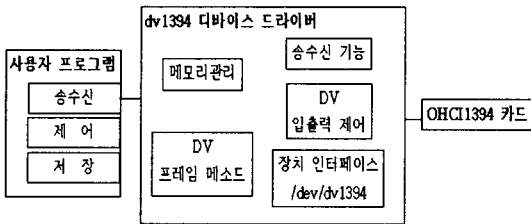


그림 3 dv1394 드라이버의 기능

### 2.3 dv1394 드라이버의 스트리밍

사용자 응용프로그램이 DV 데이터 스트리밍을 위해 DV 프레임을 읽고 쓰는 과정은 그림 4와 같으며, DV 데이터를 전송하기 위한 송신 과정은 다음과 같다.

- ① DV 데이터를 전송하기 위해 사용자 프로그램은 가장 먼저 DV1394\_INIT를 통해 채널, 전송 프레임 수, 버퍼를 초기화하고 dv1394 드라이버를 준비시킨다.
- ② 버퍼의 상태와 송수신이 가능한 프레임의 수를 확인하는 부분인 DV1394\_GET\_STATUS는 전송 프레임 수, 수신 가능한 프레임 수, 이미 수신된 프레임 수, 현재 수신된 프레임 수의 값을 보여준다. 그 상태 정보를 이용하여 DV1394\_SUBMIT\_FRAMES는 전송 가능한 수만큼의 프레임을 전송한다. 예를 들어, DV1394\_GET\_STATUS가 n\_frames=4, active\_frame=1, first\_clear\_frame=3, n\_clear\_frames=2이면 DV1394\_SUBMIT는 3개의 새로운 프레임을 전송한다.
- ③ 모든 DV 데이터가 전송되어 더 이상 전송할 프레임이 없을 경우 DV1394\_SHUT\_DOWN을 통해 데이터 전송을 중지하고 버퍼를 비운다.

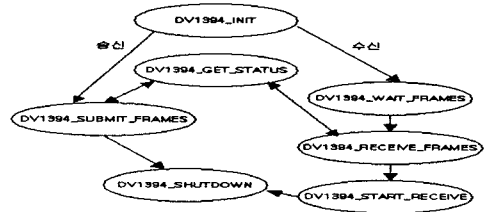


그림 4. dv1394 드라이버에서의 DV 데이터 송수신 과정

DV 데이터의 수신 과정은 다음과 같다.

- ① DV1394\_INIT로 드라이버를 준비하고 버퍼가 차있을 경우에 새로운 데이터를 받기 위해 DV1394\_WAIT\_FRAMES에서 버퍼를 비운다.
- ② DV1394\_RECEIVE\_FRAMES는 DV1394\_GET\_STATUS로부터 수신할 수 있는 프레임의 상태 정보를 얻는다.
- ③ DV1394\_START\_RECEIVE에서 DV 프레임을 수신한다.
- ④ 모든 DV 데이터를 수신하여 더 이상 수신할 프레임이 없으면 DV1394\_SHUTDOWN을 통해 데이터 수신을 중지하고 버퍼를 비운다.

## 3. 구현 및 검토

### 3.1 구현

본 논문에서 제안한 멀티미디어 데이터 스트리밍 시스템은 dv1394 디바이스 드라이버를 사용하여 리눅스 커널 2.4.19 상에서 사용자 프로그램으로 구현하였다. IEEE1394 어댑터 카드는 PYRO의 OHCI1394를 사용하였고 DV 데이터는 SONY의 TRV-110 디지털 캠코더를 사용하여 입력받았다. DV 데이터를 저장하기 위한 저장 장치로는 1394 외장형 하드디스크와 1394 외장형 시디 R/W를 사용하였다. 사용자 프로그램은 Gtk와 C언어로 구현하였으며 그림 2에서 보인 각 모듈로는, 디지털 오디오, 비디오의 제어를 위한 libavc1394-0.3.1, API인 libraw1394-0.9.0, 멀티미디어 데이터 코덱인 libdv-0.98을 사용하였다. 그리고 IEEE1394를 위한 드라이버로는 커널 2.4.19의 ieee1394, ohci1394, raw1394를 사용하였다.

사용자 프로그램은 DV 데이터의 스트리밍과 저장, 전송, 수신 기능을 가진다. 즉 IEEE1394에 연결된 디지털 캠코더로부터 입력되는 DV 영상을 리눅스 서버에 파일로 저장하고, 리눅스 서버가 저장된 파일을 IEEE1394를 통해 전송하면 리눅스 클라이언트가 수신하는 과정을 실험을 통해 확인하였다. 서버가 전송한 DV 데이터를 리눅스 클라이언

트가 수신하여 재생하는 예는 그림 4와 같다.



그림 4. 전송된 dv 데이터의 재생 예

### 3.2 결과 및 검토

dv1394 드라이버를 통한 DV 데이터의 송수신은 기본적으로 OHCI1394 카드에 장착된 메모리 내의 DMA 스페이스를 통해 이루어진다. 스트리밍 시스템의 동작은 dmesg를 통해 DMA가 사용되는 정보를 모니터링하고 printk()를 사용하여 dv1394 드라이버의 동작을 디버깅함으로써 확인하였으며, 전송 과정은 그림 5, 수신 과정은 그림 6과 같다.

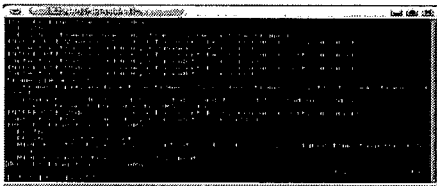


그림 5. dv1394를 통한 스트리밍 데이터 전송

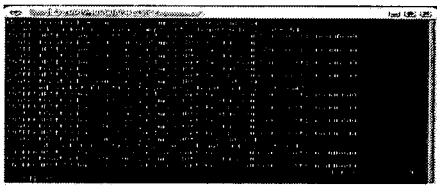


그림 6. dv1394를 통한 스트리밍 데이터 수신

각각의 디버깅을 통해 송수신을 위한 DMA 접근, 2.3절에서 설명한 드라이버 상태정보를 통한 송수신 과정, 전송 패킷 크기, 타임스탬프와 같은 전송 파라미터를 확인하였다

### 4. 결론 및 향후과제

앞으로 홈 네트워크 기술이 더욱 발전하면 가정 내의 모든 정보 가전기기들은 하나의 네트워크로 연결될 것이며 동영상과 음성이 포함된 대용량의 멀티미

디어 데이터가 기기간에 송수신 될 것이다. 본 논문에서는 IEEE1394를 통해 가정 내의 디지털 오디오, 비디오, 디지털 캠코더, 외장형 저장장치와 같은 AV 기기와 저장장치를 연결하고 노드간에 대용량 멀티미디어 데이터를 실시간으로 송수신할 수 있는 멀티미디어 스트리밍 시스템을 리눅스 상에 구현하였다. 이를 통해 사용자는 자신의 PC나 TV앞에서 위성방송을 수신하고 오디오나 캠코더를 제어하며 수신한 멀티미디어 데이터를 실시간으로 저장장치나 VCR에 저장할 수 있어 보다 편리한 생활을 누릴 수 있을 것이다. 이 시스템은 향후 실시간 스트리밍을 요구하는 원격 모니터링 시스템, 화상회의 그리고 비선형 비디오 편집 시스템에 확장 응용될 수 있다.

리눅스 환경에서 IEEE1394 기반의 멀티미디어 스트리밍은 현재 DV는 어느정도 안정적이지만 MPEG2는 지원되지 않고 있다. 따라서 mpeg1394 디바이스 드라이버의 개발[11]과 이를 이용한 MPEG2 스트리밍 기술의 개발은 향후 과제로 남는다.

### [참고문헌]

- [1] P1394 Draft Std for a High Performance Serial Bus, IEEE, March 15, 1998.
- [2] 1394 Trade Association, <http://www.1394ta.org>.
- [3] 송석일, "홈 네트워크 표준화 기술 동향", 전자통신 동향분석, 제15권 제6호, pp.56-64, ETRI.
- [4] 김두현, 김선자, 문경덕, 박광로, 이진우, 마평수, "인터넷 정보가전 기술 개발 및 표준화 동향", 정보통신연구원 간행물 6호, Oct. 2000
- [5] IEC61883-1 : "Consumer audio/video equipment Digital Interface", General, Feb. 1998.
- [6] IEC61883-4 : "Consumer audio/video equipment Digital Interface", MPEG2-TS data transmission, Feb. 1998.
- [7] IEEE1394 for linux, <http://www.linux1394.org>
- [8] MPEG over 1394, "System@IC news", March 2001.
- [9] Adam J, Kunzman, Alan T. Wetzel, "1394 High Performance Serial Bus: The Digital Interface for ATV", IEEE Transaction on Consumer Electronic, vol. 41, no. 3, pp.893-900 August 1995.
- [10] HyperLynx, Proposal for EIA Standard ATV Baseband Digital Interface, February 8, 1995.
- [11] [www.geocrawler.com](http://www.geocrawler.com)