

General Purpose DSP 기반의 멀티미디어 스트리밍 시스템 구현

김동환, 문재필, 오화용, 이은서, 장태규
중앙대학교 전자전기공학부

A General Purpose DSP based Multimedia Streaming System

Dong-Hwan Kim, Jae-Pil Moon, Hwa-Yong Oh, Eun-Seo Lee, Tae-Gyu Chang
School of Electrical and Electronics Engineering, Chung-Ang University

Abstract - 본 논문에서는 인터넷을 통한 멀티미디어 스트리밍 서비스 환경에서 다양한 표준으로 압축된 콘텐츠의 디코딩을 지원하기 위하여 general purpose DSP (Digital Signal Processor) 기반의 멀티미디어 서비스 플랫폼을 구현하였다. 다양한 표준 방식으로 압축된 멀티미디어 콘텐츠를 재생하기 위하여 Host 프로세서와 DSP 구조의 하드웨어를 설계하고, 멀티미디어 코덱을 DSP에 다운로드하는 소프트웨어적인 기법을 적용하였다. 설계한 플랫폼의 동작을 검증하기 위하여 리눅스 기반에서 DSP를 제어하는 네트워크 클라이언트 소프트웨어를 구현하고, TI의 TMS 320C6416을 대상으로 구현한 MPEG-2 비디오와 AC-3 오디오 코덱을 적용하여 스트리밍 환경에서 멀티미디어 데이터가 원활하게 재생되는 것을 보였다.

MPEG-2 비디오 / AC-3 오디오와 같은 멀티미디어 코덱을 적용하여 동작을 검증하고자 한다.

2. 본 론

본 절에서는 본 논문에서 구성한 멀티미디어 서비스 환경과 다양한 표준의 멀티미디어 데이터를 재생할 수 있는 클라이언트 시스템의 구현에 대해 기술한다.

2.1 멀티미디어 스트리밍 서비스 구조

중앙의 서버에 의한 멀티미디어 스트리밍은 서버에 발생하는 과중한 부하와 네트워크 대역폭의 제약으로 인하여 다수의 사용자를 대상으로 고품질의 멀티미디어 서비스를 제공하는 것이 어렵다[2]. 이에 본 논문에서는 성능, 확장성 및 비용 등의 측면에서 네트워크 대역폭의 공유에 용이한 P2P 방식을 기반으로 하는 멀티미디어 스트리밍 서비스 구조를 구성하였으며 이를 그림 1에 나타내었다.

1. 서 론

현재 대부분의 멀티미디어 디코더 시스템은 성능과 전력 소모 등에 우수한 전용 하드웨어를 사용하는 것이 일반적이다. 그러나 이러한 전용 하드웨어는 개발이 어려울 뿐 아니라, 유연성이 적어 추후 알고리즘의 변경이 매우 어렵다. 이에 비해 DSP 기반의 플랫폼은 코드를 DSP로 다운로드 하는 방식으로 동작하므로 유연성이 우수하다. 또한 특정 알고리즘의 모듈화를 통해 알고리즘의 선택적 재구성에 의한 개발 기간의 단축 효과를 가져올 수 있으며 알고리즘의 유지 및 보완이 용이하다. 또한 DSP 기반의 디코더 시스템은 새로운 코덱이 개발되어도 하드웨어의 변경 없이 코덱을 다운로드 함으로써 새로운 코덱을 지원할 수 있다. 이러한 DSP 기반의 플랫폼은 전용 하드웨어에 비해 상대적으로 낮은 성능이 단점이지만 DSP 기술이 발달함에 따라 성능의 차이는 점차 줄어들고 있다[1].

한편, 네트워크 기술의 발달과 고속 네트워크 망의 보급으로 멀티미디어 스트리밍 서비스의 규모는 갈수록 커져가는 추세이다. 또한 디지털 방송이 시작과 함께 PDP, LCD 등의 고품질 TV 보급화로 고품질의 멀티미디어에 대한 사용자의 요구가 증대하고 있다. 그러나, 서버 용량, 네트워크 대역폭 등의 한계는 다수의 사용자를 대상으로 고품질의 멀티미디어 서비스를 하는 데 있어 큰 제약조건이 되고 있다.

이에 본 논문에서는 제한된 네트워크 대역폭의 제약 조건을 극복하기 위하여 네트워크 대역폭의 공유가 용이하고 서버의 제어 기능을 고려하여 Hybrid P2P 기반의 스트리밍 서비스 구조를 구성하였다. 또한, 다양한 표준의 멀티미디어 압축 알고리즘을 지원할 수 있도록 Host PC와 general purpose DSP 기반의 멀티미디어 디코더 시스템을 설계하였다. 설계한 DSP 기반의 디코더 시스템은 리눅스 플랫폼을 대상으로 스트리밍 서비스를 위한 네트워크 모듈과 멀티미디어 스트림의 DSP 전송 모듈을 포함하는 클라이언트 소프트웨어를 구현하고, 여기에

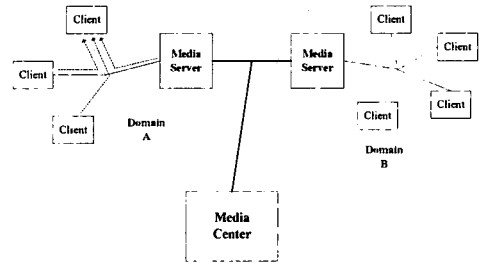


그림 1. 본 논문에서 구성한 P2P 기반의 멀티미디어 스트리밍 서비스 구조.

미디어 서버는 적정 단위의 서비스 지역에 위치하며 미디어 센터로부터 콘텐츠를 공급받아 해당 지역에 서비스 한다. 클라이언트는 미디어 서버에 콘텐츠를 요청하고, 서버와 콘텐츠를 갖고 있는 다른 클라이언트로부터 스트림을 수신 받는다. 서버는 최소한의 데이터를 클라이언트로 전송하며, 데이터를 제공하는 클라이언트는 자신의 프로세스를 수행하는데 영향을 미치지 않을 정도의 부하 내에서만 스트림을 제공한다. 이로 인하여 서버는 데이터 전송 부담이 줄어들게 되어 서버에서의 네트워크 대역폭을 줄일 수 있게 된다. 이와 같이 서버를 둔 Hybrid P2P 방식의 멀티미디어 서비스 구조를 적용함으로써 서버에서 각 클라이언트 간의 통신 품질을 모니터링 하여 데이터 신뢰성 보장과 서비스의 품질 제어가 가능하다[4].

2.2 스트리밍 서비스를 위한 멀티미디어 클라이언트 구현

다양한 표준 방식으로 압축된 멀티미디어 스트림을 재생할 수 있는 클라이언트 시스템을 구현하기 위하여 DSP 코덱을 다운로드 함으로써 멀티미디어 데이터의 디코딩이 가능한 general purpose DSP 기반의 하드웨어를 설계하고, DSP 코덱으로 멀티미디어 스트림을 전송하는 리눅스 기반의 클라이언트 소프트웨어를 구현하였다.

2.2.1 General Purpose DSP 기반의 H/W 구현

그림 3은 TI의 TMS320C6416 DSP를 사용하여 구성된 디코더 시스템이다[3]. 멀티미디어 클라이언트 시스템은 압축된 멀티미디어 데이터의 디코딩을 위한 DSP 디코더 시스템과 서버에 접속하여 스트림을 받아 DSP로 전송하는 호스트 PC로 구성된다.

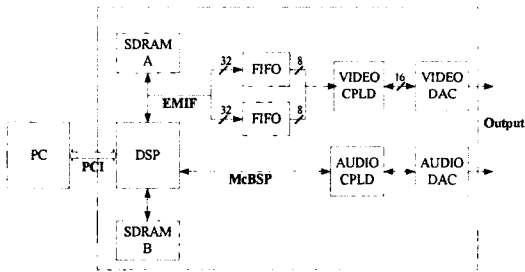


그림 2. TI TMS320C6416 DSP 기반의 멀티미디어 클라이언트 시스템 구성도

DSP 디코더 시스템은 Host PC와 PCI 버스를 통해 멀티미디어 스트림을 전송받아 DSP에서 디코딩을 수행한다. SDRAM은 기본적인 Demux, 비디오 디코더 용도 외에 비디오와 오디오 출력에 사용된다. FIFO는 EDMA (Enhanced Direct Memory Access) 전송을 통해 받은 64bit의 데이터를 4:1로 Mux 하여 16bit RGB 출력을 내보내고 비디오 CPLD에서는 디코딩된 결과를 모니터에 출력할 수 있도록 하기 위하여 Blank와 Sync를 생성하여, 비디오 DAC로 Hsync, Vsync, RGB data를 내보낸다. 오디오 CPLD는 McBSP (Multi-channel Buffered Serial Port)를 통해 받은 데이터를 demux하여 오디오 DAC로 전송한다. 표 1에 본 논문에서 구현한 DSP 기반의 디코더 시스템의 설계 사양을 요약하였다. 그림 3은 DSP 기반의 멀티미디어 디코더 시스템을 구현 결과를 나타내었다.

표 1 TI TMS320C6416 기반의 멀티미디어 디코더 시스템 설계 사양

분류	Chip	Spec.
DSP	TMS320C6416	600MHz
FIFO	IDT72V3680	166MHz, 16,384×36bit
CPLD	Altera MAX3000	116MHz
DAC[V]	THS8200	205MSPS, RGB/YCbCr
DAC[A]	AD74111	48KHz, 24bits
Memory	SDRAM A/B	256MB/64MB

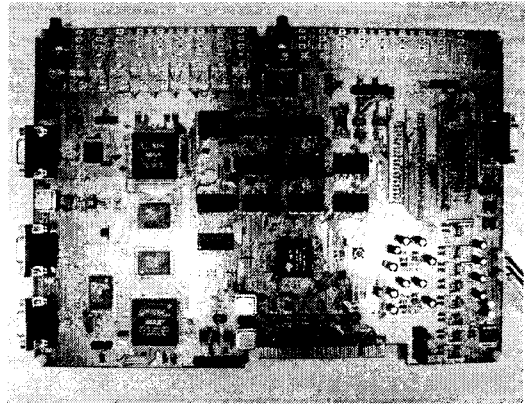


그림 3. TI TMS320C6416 DSP 기반의 멀티미디어 디코더 시스템

2.2.2 클라이언트 S/W 설계 및 구현

본 논문에서의 멀티미디어 스트리밍 방식은 일정 세그먼트 단위로 나뉘어 있는 데이터를 클라이언트의 요청에 의해 미디어 서버 혹은 다른 클라이언트로부터 전송받는 형태로 이루어진다. 이에 클라이언트 소프트웨어는 스트림을 수신하는 네트워크 모듈과 암호화 되어 있는 스트림을 복호화하는 쓰레드와 복호화된 스트림을 DSP로 전송하는 쓰레드로 설계하였다.

그림 4는 멀티미디어 스트리밍 서비스를 위한 클라이언트 소프트웨어의 동작 흐름도이다. 클라이언트는 미디어 서버나 다른 클라이언트로부터 받은 스트림에서 헤더를 추출한다. 추출한 헤더로부터 현재 전송 받은 스트림이 유효한 데이터인지 체크를 한다. 블록 단위로 암호화 되어 있는 스트림은 복호화 과정을 수행하여 DSP 전송을 위한 버퍼에 저장한다.

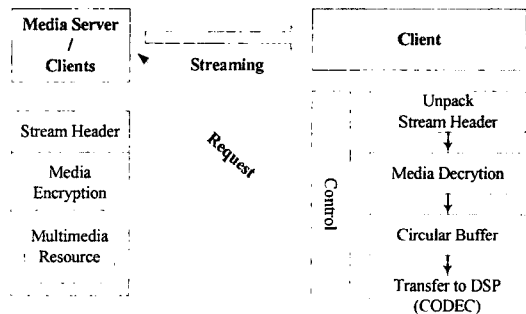


그림 4. 스트리밍 서비스를 위한 멀티미디어 클라이언트 소프트웨어 흐름 설계도

Host PC에서 DSP로의 데이터 전송은 DSP에 의한 PCI 인터럽트를 통해 이루어진다. 압축된 멀티미디어 데이터를 처리하는 DSP는 디코더 버퍼에 데이터가 부족하게 되면 PCI 인터럽트를 발생시키고, Host PC는 메모리에 저장된 데이터를 DSP로 전송한다. 멀티미디어 데이터는 시간에 따라 소비하는 데이터의 양이 일정하지 않기 때문에 데이터를 일정한 크기로 전송할 때 인터럽트가 발생하는 시점 또한 일정하지 않게 된다. 따라서 DSP에서 디코딩에 필요로 하는 데이터 크기 이상을 Host PC에서 저장하기 위한 버퍼를 두어 버퍼 언더플로우를 막아야 한다. 이를 위하여 네트워크로부터 받은 스트림을 circular 버퍼 저장하고, circular 버퍼로부터 데이터를 DSP로 전송하도록 circular 버퍼를 설계하였다. 또

한 네트워크로부터 스트림을 수신하는 쓰레드와 PCI 인터럽트에 의해 DSP로 스트림을 전송하는 쓰레드를 뮤텍스를 사용하여 동기화하여 버퍼 오버플로우를 방지하였다.

그림 5는 embedded 리눅스 환경에서 멀티미디어 클라이언트 소프트웨어를 구현한 구성도를 나타내었다. 클라이언트 소프트웨어는 리눅스 OS, 자바 VM(Virtual Machine)과 네트워크, 스트리밍 등의 미들웨어 모듈 그리고 UI 어플리케이션의 세부분으로 설계하였다. OS에서는 DSP와의 PCI 인터페이스를 위하여 리눅스 DSP 하드웨어 드라이버를 구현하였다. 리눅스 상에 자바 VM을 포팅하여 PBP (Personal Basis Profile)를 통해 UI를 DSP의 출력으로 이루어지도록 설계하였으며, JNI (Java Native Interface)를 사용하여 C/C++로 구현된 스트리밍 모듈, 복호화 모듈, DSP 전송 모듈을 자바 어플리케이션에서 호출하여 사용할 수 있도록 구현하였다[5].

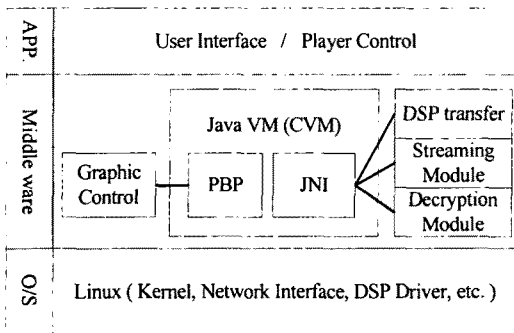


그림 5. 리눅스 환경의 멀티미디어 클라이언트 소프트웨어 구성도

2.2.3 테스트 시스템 구성 및 성능 분석

멀티미디어 스트리밍 시스템의 동작을 시험하기 위한 목적으로 하나의 스트리밍 서버와 DSP 기반의 디코더 시스템을 클라이언트로 테스트 시스템을 구축하였다. 테스트 스트림은 MPEG-2 비디오와 AC-3 오디오를 대상으로 스트리밍 서비스 환경을 갖추어 수행하였다. 클라이언트는 디코딩에 필요한 코덱을 DSP에 다운로드하고, 서버로부터 받은 스트림을 재생한다. General Purpose DSP 기반 클라이언트 시스템의 MPEG-2 비디오 디코딩 성능을 표 2에 요약하였다. 구현한 클라이언트 시스템은 720×480의 SD급 스트림의 경우 약 40%대의 부하율을 보임으로써 DVD 화질의 동영상을 재생할 수 있음을 확인하였으며, 동작 화면의 예를 그림 6에 나타내었다.

	HD Stream (1920×1088)			SD Stream (720×480)		
bitrate (Mbps)	17.7	17.45	17.55	7.63	9.35	9.54
cycles/frame (profile Avg.)	25,945,945	27,906,976	26,519,337	3,689,469	3,735,356	3,924,775
frame/sec (profile Avg.)	18.5	17.2	18.1	130.1	128.5	122.3
frame/sec (board Avg.)	11.2	10.4	10.1	74.2	71.0	69.6
Load (%)	267	288	297	40	42	43

표 2. 구현한 General Purpose DSP(TI, TMS320C6416) 기반 클라이언트 시스템의 MPEG-2 비디오 디코딩 성능 요약



그림 6. DSP 기반 클라이언트 시스템의 MPEG-2 비디오 디코딩 동작 화면 예

3. 결 론

본 논문에서는 멀티미디어 스트리밍 서비스 환경에서 다양한 서비스를 지원할 수 있는 general purpose DSP 기반의 클라이언트 시스템을 구현하였다. DSP 기반의 클라이언트는 컨테츠에 맞는 DSP 코덱을 다운로드 함으로써 다양한 표준으로 압축되어 있는 멀티미디어 스트림의 디코딩이 가능하다. 이는 Host PC와 TI의 TMS320C6416 DSP 기반의 디코더 시스템으로 구성된 클라이언트 하드웨어와 리눅스 기반의 클라이언트 소프트웨어를 구현하고, MPEG-2 비디오와 AC-3 오디오 코덱을 적용하여 동작을 확인함으로써 검증하였다. 설계한 DSP 기반의 멀티미디어 스트리밍 시스템을 실제 인터넷과 같은 환경에 적용하기 위해서는 DSP와 Host PC에 대한 동작 성능에 대한 분석과 이를 바탕으로 P2P 환경에서의 스트리밍 체계를 설계하고 이에 대한 동작 검증이 필요하다.

감사의 글

본 연구는 산업자원부 차세대 디지털컨버전스플랫폼 기술 개발 과제 지원에 의해 수행되었음.

[참 고 문 헌]

- [1] Iain E. G. Richardson, Video Codec Design, John Wiley & Sons, 2002.
- [2] W. H. Ma and D. H. C. Du, "Reducing Bandwidth Requirement for Delivering Video over WAN with Proxy Server", IEEE Trans. on Multimedia, vol. 4, no. 4, pp. 539-550, Oct. 1999.
- [3] Texas Instruments, "TMS320C6000 Technical Brief", SPRU197D, Feb. 1999.
- [4] Peering Portal White Paper, "P2P Market Report", Apr. 2002.
- [5] Sheng Liang, The Java Native Interface, ADDISON-WESLEY, 1999