

논문96-1-2-05

# LAN을 이용한 VOD 시스템의 구현

김 윤 범\*, 이 준 형\*\*, 최 윤 식\*\*, 이 정 수\*\*\*

## Implementation of VOD System Using LAN

Youn-Beom Kim\*, Joon-Hyoung Lee\*\*, Yoon-Sik Choe\*\*, and Jeong-Su Lee\*\*\*

### 요 약

본 논문에서는 차세대 영상 서비스 기술인 VOD(Video-on-Demand)시스템을 LAN을 이용하여 구현하였다. LAN을 이용한 VOD 시스템은 네트워크 병목현상과 이에 따른 시간 지연 때문에 기존의 네트워크 구현으로는 원활한 서비스가 되지 못하므로, 그러한 문제점을 해결하기 위한 방안으로, 본 논문에서는 토폴러지의 변경과 그에 따른 네트워크 장비의 선정, 개선된 프로토콜의 제정 등이 제시되었다. 결론적으로 구현된 시스템은 세그먼트 당 최대 3~4명의 클라이언트를 수용할 수 있었고, 전송하고자 하는 영상 데이터의 크기에 따라 미리 네트워크 버퍼의 크기를 최적화하는 것이 서비스의 품질을 높이는데 필수적이라는 것을 알 수 있었다.

### Abstract

In this paper, a VOD(Video-on-Demand) system is implemented using LAN(Local Area Network). The use of LAN causes the time-delay problem due to its narrow channel limitation, as a result, the service using the conventional LAN used not to be popular. In this paper, to solve these problems, we develop the modified topology, the selection methodology of the network equipment, and the dedicated(improved) protocol. Consequently, we can serve at most 3 or 4 clients at the same time, and we conclude it is necessary to optimize the network buffer size depending on the size of video data.

### I. 서 론

하드웨어 기술의 발달로 컴퓨터 기억용량이 커져 영화 비

디오나 프로그램의 전자 저장소를 만들 수 있게 되었고, 데이터의 압축기술의 발전과 광 네트워크에 의해 압축된 작은 용량의 비디오 데이터를 빠른 속도로 전송하고 새로운 스위칭 장비로써 빛과 동등한 속도로 라우트 하는 것이 가능하게 됨으로써 대화형 미디어 서비스의 구현이 실현되게 되었다. 이러한 대화형 미디어 서비스 중에서 가장 먼저 구체화되고 있는 서비스가 바로 주문자 요구형 비디오, 즉 Video On Demand(VOD)이다[1].

VOD 시스템의 구현에 있어 데이터의 전송 미디어의 선택은 다양해 질 수 있다. 전송되는 데이터의 종류에 따라 저속의 WAN망에서 고속의 ATM망에까지 여러 가지 네트워크

\*삼성전자 마이크로사업부  
Samsung Electronics Co., Division of Microprocessor  
\*\*연세대학교 전자공학과  
Dept. of Electrical Eng., Yonsei University  
\*\*\*한국통신 연구개발단  
Korea Telecom.

상에서 구현될 수 있으며 실제로 많은 시스템들이 다양한 네트워크 플랫폼 상에서 구현되고 있다.

본 논문에서 제안하는 VOD 시스템의 Prototype은 LAN 환경 하에서 동작하게끔 설계되어 있다. LAN상에서 구현시 대용량의 스트림을 연속적으로 전송해야 하는 부담에 따라 네트워크상의 문제점은 크게 두 가지로 나누어 진다. 첫째, 네트워크상의 병목현상에 관한 문제이다. 네트워크상의 병목현상은 하나의 비디오 서버가 다수의 클라이언트에게 서비스를 동시에 제공함으로써 발생하는 문제점이다. 또한 네트워크가 이더넷 방식인 경우 버스 방식인 CSMA/CD방식을 채택하고 있으므로 전송 매체의 공유에 따른 상당한 병목현상이 발생할 수 있다. 둘째, 네트워크 프로그램상의 문제이다. 네트워크 프로그램상의 문제는 프로토콜의 제정과 하드웨어의 속도에 따라 좌우된다[2,3]. 즉, 만약 네트워크 상에서 일정한 전송대역폭을 제공하지 못한다면 구현에 있어 문제점이 발생할 수 있다.

본 논문에서는 II장에서 VOD 시스템의 요건에 대해서 알아보고, III장에서는 위에서 말한 문제점을 네트워크의 구성과 프로토콜을 제안하여 해결방안을 보였으며, IV에서는 실제로 VOD 시스템을 구현해보았다.

## II. VOD시스템을 위한 요건

VOD 시스템의 비디오 데이터는 텍스트나 수치적 타입의 기존의 데이터와는 다른 특성을 지닌다. 비디오·오디오 같은 멀티미디어 데이터들은 특히 비디오 데이터의 경우 그 크기가 매우 크다. 예를 들어 DVI 미디어 세그먼트는 10분에 90Mbytes의 용량이 필요하다. 이 멀티미디어 데이터는 대부분 갱신이 거의 필요없는 읽기 전용 데이터베이스(read-only database)의 성격을 지닌다. 또한 네트워크를 거치는 비디오 데이터는 끊어지지 않고 이어지는 스트림(stream) 형태로 발생해야 한다. 이 데이터들은 미디어 형태에 의해 요구되는 대역폭에 맞게 계속적이고 실시간(real-time)으로 재생되어야 한다. 그리고 요구 즉시형(on demand)이므로 동시에 여러 사용자들을 지원할 때 새로운 스트림에 대하여 어떤 특정한 한계내의 시작 지연(start-up latency)이 보장되어야 한다.

이러한 데이터의 특성과 많은 사용자를 지원하기 위한 경제성 있고 확장 가능한 VOD 시스템의 구성을 위해 서버는 대용량의 저장장치를 사용해야 한다[5,6]. 즉, 대용량의 비디오 데이터의 저장을 위해 저장장치는 대단히 많은 수의 디스크로 구성될 것이다. 비압축된 데이터를 저장할 때 필요한 디스크는 매우 방대하므로 여기서는 MPEG으로 압축된 데이터를 저장하는 시스템을 가정한다. MPEG은 손실 가능한 압축방법이다. 그러므로 MPEG으로 압축된 데이터는 압축

으로 인한 에러가 중대한 영향을 끼칠 수 있는 의학/과학용 데이터에는 적합하지 않을 수 있지만, MPEG 압축율은 1:25 이상이고 1.5Mbps의 데이터 전송율을 지원하여 기존의 압축 알고리즘보다 좋은 성능을 지니므로 비디오 데이터에는 적합하다.

## III. 네트워크상의 문제점과 해결방안

앞에서 말한 네트워크상의 문제점 중에서 먼저 네트워크상의 병목현상에 관하여 알아보면, 네트워크의 구성은 10Mbps 대역의 이더넷으로 구성되어 있고, 이더넷은 IEEE 802.3 표준을 따르므로 CSMA/CD 통신망 접근방법을 이용한다. LAN에 연결된 장치들은 망의 전송 능력을 공유하게 되는데, 매체의 전송 능력을 공유하기 위해서는 전송 매체에 대한 접근을 제한하는 방법이 필요하게 되며, 이러한 방법을 규정한 것을 매체 접근 제어(MAC) 프로토콜이라 한다. CSMA/CD는 이러한 매체 접근 제어의 한 방식으로 전송을 원하는 노드는 다른 노드가 매체를 사용하지 않을 때 데이터 프레임 전송할 수 있다. 이러한 CSMA/CD 방식에서 매체를 접근하기 위한 알고리즘은 다음과 같다[2].

① 매체가 유휴 상태이면 전송을 개시하고, 그렇지 않으면 ②, ③으로 간다.

② 매체가 이용 상태이면 유휴 상태가 감지될 때까지 감시를 계속하고, 유휴 상태가 감지되면 즉시 전송을 개시한다.

③ 전송하는 동안에 충돌이 검출되면, 모든 노드에 프레임 충돌을 알리는 신호를 보내고 전송을 중단한다.

④ 일정 시간이 지나면 재전송을 시도한다(①로 감).

그러나 버스방식인 CSMA/CD 접근 방식을 채택하면 전송 매체의 공유에 따른 상당한 병목현상이 발생할 수 있다.

다음은 네트워크 프로그램상의 문제인데, 네트워크를 통하여 전송되는 데이터는 끊어지지 않고 연속적으로 이어지는 스트림의 형태이어야 하므로 일정한 전송대역폭을 제공하여야 한다.

따라서, 이러한 문제점들을 해결하기 위하여 다음과 같은 해결방안을 제시하고자 한다.

### 1. 네트워크 상의 병목현상

네트워크 상의 병목현상에 대한 해결 방안으로는 토팔러지의 변경과 그에 따른 네트워크 장비의 선정이 하나가 될 수 있다.

기존의 LAN 환경에서는 버스 형태의 토팔러지를 사용한다 [그림 1a]. 버스형 토팔러지는 공용 케이블에 부착된 여러 개의 노드로 구성된다. 한 노드에서 송신한 데이터가 버스 상에 있는 모든 노드에 동시에 전달되며, 한 순간에 한 노드만이 매체에 데이터를 송신할 수 있다. 전송되는 패킷에

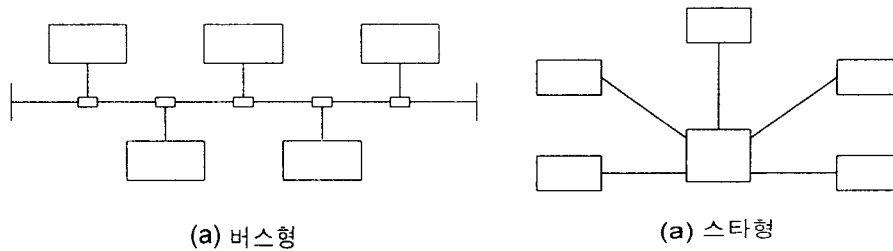


그림 1. LAN의 망 구성 형태  
Fig. 1. Network topology of LAN

는 목적지 주소와 발신지 주소가 포함되어 있으며, 각 노드는 매체를 감시하여 자신의 주소를 가지고 있는 패킷을 복사한다.

이러한 버스형 토폴러지를 스타형 [그림 1b] 으로 전환하면 각 클라이언트에게 최대의 전송대역폭을 보장해 주게 되어 어느 정도의 병목현상을 극복할 수 있다. 이것은 버스 방식에서 병목현상의 원인이 되는 전송매체의 공유를 피할 수 있기 때문이다. 스타형에서는 중앙의 교환 장치가 망 내의 모든 노드를 상호 연결하는 역할을 한다. 전송을 원하는 노드가 중앙의 교환 장치에 접속을 요구하면, 교환 장치는 두 노드를 연결시켜 주기 위한 교환 기능을 수행한다. 일단 접속이 이루어지면, 두 노드 간은 점 대 점 전용 회선으로 접속한 것과 같게 된다.

이러한 구성으로 소규모 로컬 네트워크 상에서 다수의 클라이언트에 대해 VOD 서비스를 구현할 수 있으며 특히 서버쪽의 노드를 별도의 고속 노드(예를 들어 100Mbps급 고속 이더넷)로 하면 훨씬 많은 클라이언트에게 서비스를 제공할 수 있게 된다.

2. 네트워크 프로그램상의 문제

네트워크 상에서 일정한 전송대역폭을 제공하기 위한 네트워크 프로그램상의 문제는 프로토콜의 제정에 따라 좌우되므로 네트워크 상의 계층별 프로토콜에 대해서 알아보자. 네트

워크 계층에서의 프로토콜은 TCP/IP를 기본으로 한다. TCP/IP의 계층적 구조와 OSI 7 Layer RM(Reference Model)과의 관계는 표 1과 같다.

이제 각 계층에서의 프로토콜에 대해서 알아보면, 응용계층의 프로토콜에서는 전송계층에서 TCP의 신뢰성외에도 응용계층에서의 신뢰성 확보와 제어 신호에 따른 동기화 및 handshaking에 주안점이 있다. 우선 신뢰성 측면에서는 TCP에서 윈도우 시스템을 통한 성능의 향상 및 재전송 알고리즘을 통한 신뢰성을 보장하고 있으나 데이터의 종류가 MPEG 스트림이고 일반 데이터와는 다른 관리를 해야 하며 시간적 측면에서 연속성을 갖기 위해 일반 데이터보다 높은 신뢰성을 요구하고 있으므로 이에 따라 응용계층에서 재전송 및 윈도우 시스템을 통한 Acknowledgement를 수행하여 신뢰성을 보장하고자 한다. 우선 각각의 기능에 대한 요구를 전송한 다음 데이터를 전송받게 되고 전송이 안되었을 경우 다시 요구를 전송하여 재전송하게 된다. 현재의 시스템에서는 한 블록당 한 요구가 전송되므로 네트워크 성능에 따라 여러개의 블록당 한 요구를 전송하게끔하여 효율의 증대를 기대할 수 있다[그림 2].

MPEG 데이터의 전송은 일반 데이터와 달리 파일의 크기에 시간의 단위가 추가된다. 즉 MPEG 스트림에 대해 버퍼의 크기로 분할해서 전송할 경우 버퍼 관리상의 문제점 때문에 분할 기준을 시간단위로 처리해야 한다. 따라서 전송시와 수신시 데이터의 내용을 분석하여 I 프레임간의 간격으로 내용을 분할해서 처리한다. 이를 위하여 전송계층에서 MPEG 전송 전용 프로토콜을 제안할 필요가 있으며 Presentation 계층에서 MPEG 데이터의 분석기능을 추가해야 한다.

전송계층에서 사용되는 프로토콜로서 TCP(Transmission Control Protocol)를 사용하는데 이는 접속형 네트워크 프로토콜이므로 비디오 데이터에 대한 전송의 신뢰성이 보장된다. 스트림 형태의 데이터 전송이 가능하므로 대용량의 데이터 전송에 적합하며 구현이 용이하고 기종에 상관없이 이식이 가능하다. 이 밖에도 사용자 측에서 프로토콜의 변형이

표 1. TCP/IP의 계층적 구조

Table 1. Hierarchical Structure of TCP/IP

TCP/IP	OSI
Application Level	Application Layer Presentation Layer Session Layer
TCP Level	Transport Layer
IP Level	Network Layer
Network Level	Datalink Layer Physical Layer

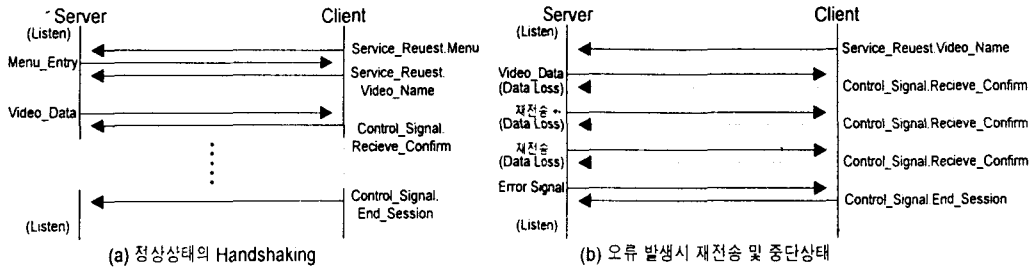


그림 2. 전송 데이터의 흐름 제어  
Fig. 2. Flow Control of Trasfered Data

용이하고 성능의 향상을 위해 헤더의 부하는 줄일 수 있는 장점이 있다. 그러나 지나치게 큰 헤더로 인해 속도가 느리다는 단점을 극복해야 한다. UDP(User Datagram Protocol)는 비접속형 프로토콜로서 비디오 데이터의 제어 신호 전송용으로 사용한다. 비접속형 프로토콜이므로 신뢰성은 떨어지나 제어 신호의 크기가 작으므로 데이터그램으로 전송해도 가능하다. 제어신호의 최대 크기는 150바이트이므로 MTU보다 크지 않은 이상, 하나의 데이터그램으로 전송할 수 있다.

본 논문에서는 클라이언트를 세그먼트당 5대 이하로 제한하고 있기 때문에 네트워크 계층 프로토콜인 IP(Internet Protocol)의 역할을 그리 크지 않다고 할 수 있다. IP의 데이터그램은 Network MTU에 따라 그 크기와 단편화가 달라진다. LAN망을 기반으로 하고 다른 외부의 네트워크와 접속하지 않으므로 MTU를 1500바이트로 고정해서 데이터를 전송한다. 이것은 가능하면 단편화에 대한 부담을 줄여서 속도를 증가시키고자 하는 것이다.

이더넷은 데이터링크 계층의 프로토콜로서 IEEE 802.3 CSMA/CD(Carrier Sense Multiple Access Protocol/Collision Detection)방식을 사용하고 있다.

물리계층의 프로토콜은 이더넷을 사용하고 있는데, 이더넷의 규격중 다음의 3가지가 일반적으로 많이 사용된다. 10BASE-T(Unshielded Twisted Pair Cable)는 허브를 이용한 구성에 이용되며 케이블링이 간편하고 네트워크의 안정성이 뛰어나다. 10BASE-5(Attachment Unit Interface Cable)는 thick ethernet이라고도 부르며 보통 소규모의 백본용으로 사용되고, 케이블링이 어렵다. 10BASE-2(BNC)는 thin ethernet이라고도 하며 케이블링은 간편하나 네트워크의 안정성이 떨어지는 특징을 가진다.

IV. 시스템 구현

LAN상에서 구현한 VOD 시스템은 프로토콜로 TCP/IP를 사용하고 네트워크의 구성은 이더넷을 이용하여 단일 세

그먼트내에서 동작하게끔 되어 있다. 이것은 Dedicated Network Configuration으로서 실제로 구현되는 시스템에서 충분한 전송 대역폭이 나오지 않기 때문에 타 네트워크와의 접속을 끊고 전용 네트워크를 꾸밈으로서 전송 대역의 활용도를 높이기 위한 방안이다.

VOD 시스템은 대량의 디지털 비디오 데이터를 저장·검색하여 실시간내에 사용자에게 전송하는 비디오 서버와 가입자들을 서버에 연결하는 네트워크, 그리고 전송받은 데이터를 디코딩하여 화면으로 보여주는 클라이언트의 세가지 부분으로 나누어진다.

전송데이터로 사용하는 MPEG-1은 1.5Mbps 정도의 전송율을 통해서 VHS급의 화질을 제공할 수 있는데, 본 논문에서는 MPEG-1기법을 이용한 Video-CD 타이틀을 서비스되는 동영상 자료로 선정하였다. Video-CD는 대략 1.3Mbps의 전송율을 요구하므로 LAN환경에서 구축되기에 적절하며, 이미 많은 양의 프로그램들이 공급되고 있기 때문에 쉽게 프로그램을 구비할 수 있다는 장점이 있다.

전송망의 구성은 10Mbps 대역의 이더넷으로 구성되어 있다. 비디오 서버는 사실상 100Mbps급의 고속 이더넷으로 접속되어야 하나 구성상 일반 이더넷으로 접속하였다. 따라서 1.5Mbps정도의 MPEG-1을 전송데이터로 사용하는 경우에는 클라이언트의 개수가 5정도로 제한될 수밖에 없다. 프로토콜은 TCP/IP를 사용하며, 네트워크 상에서의 데이터 전송은 소켓을 사용한다.

클라이언트는 DOS 및 Windows용으로 각기 개발되었으며 DOS용은 네트워크 소프트웨어로서 Netstar를 사용하였다. Netstar는 현재 Microsoft Socket Implementation for DOS 규격을 비교적 잘 준수하고 있으나 네트워크 버퍼의 크기를 조절하는 등의 구성 사항을 변경할 수가 없다는 단점이 있다. 또한 Windows 환경에서 Winsock 규격을 준수하고 있지 않기 때문에 Windows 환경하에서는 Trumpet Winsock v1.0을 사용하였다. Trumpet Winsock v1.0은 현재 Windows Socket v1.1을 잘 만족시키고 있으며 개발자 환경은 Borland C/C++ v4.0에서 제공하는 winsock.h 헤

더 파일을 사용하였다. VOD 시스템의 하드웨어와 소프트웨어 구성을 살펴보면 다음과 같다.

(1) 하드웨어 개발환경

- CPU : Intel 80486DX-66, 66MHz
- Network Interface : NE2000 Compatible, 10BASE-T
- MPEG Decoder : RealMagic Lite
- VGA Card : Standard VGA 이상

(2) 소프트웨어 구성

- OS : MS-DOS v6.2, MS-Windows v3.1
- Network Software : 삼성 Netstar v1.1, Trumpet Winsock v2.0b(Winsock Version 1.1)
- C-Compiler : Borland C/C++ v4.0(for DOS and Windows)
- Packet Driver : Ethernet NE2000
- Open RealMagic API Document Set
- RealMagic Full Motion Driver v2.1 Beta

이제 데이터의 수신 구조를 살펴보면, 데이터의 수신구조는 그림 3과 같다.

블록 다이어그램에서 우선 임시 버퍼의 할당을 위하여 네트워크 큐의 개수만큼 버퍼를 초기화 한다. 이 때 각 버퍼의 크기는 BUF\_SIZE로 정의되며 이 값은 별도로 1Kb로 정의되어 있다.

이제 각 디바이스 드라이버 별 초기화를 해야 하는데 FindDriver 함수를 이용해 드라이버가 메모리 상에 어느 위치에 존재하는가를 알아낸 후 Full Motion Driver를 초기화 한다. 그 다음 socket\_init 함수로 전역변수로 되어 있는 소

켓 디스크립터를 설정하여 보고자하는 데이터 파일을 지정한 후 네트워크를 통하여 스트림 데이터를 전송받게 되는데 네트워크 버퍼 크기만큼 여러번을 전송받아 하나의 임시 버퍼를 채우게 된다. 일단 하나의 임시 버퍼가 준비되면 Full Motion Driver를 FMPIInit라는 매크로를 통하여 초기화를 하고 CallBack Function을 지정한다.

CallBack Function까지 지정되면 이후 스트림을 열어서 그 스트림의 타입이 지정된 MPEG-1 스트림일 경우 화면 크기를 지정하여 재생을 하게 된다. 스트림의 재생은 인터럽트를 통하여 후면작업으로 진행되므로 이후의 흐름은 전면작업으로 진행된다. 이후 계속적으로 버퍼를 네트워크에서 전송 받으며 버퍼의 스위칭은 CallBack Function을 통해 인터럽트를 이용한 후면작업과 재진입의 반복으로 이루어 진다.

CallBack Function은 RealMagic Full Motion Driver에서 제공하는 형태로서 이것은 Decoder에서 인터럽트를 통한 후면작업(Background Job)으로 video overlay를 처리하기 때문에 스트림을 읽어서 decoder로 넘겨주기 위한 방법을 제공하기 위해 사용된다. 현재 RealMagic에서 제공하는 CallBack Function의 종류는 하나의 스트림에 대해 하나의 CallBack Function을 정의할 수 있는데, 현재의 시스템에서는 모든 스트림에 대해 하나의 CallBack Function을 정의하고 있다. 이것은 다음의 명령을 통해서 이루어 진다.

FMPCallback(hStream, Value);

여기서 스트림의 핸들러를 0으로 하게 되면 전역 CallBack Function이 정의되어 모든 스트림에 대해 같은 CallBack Function이 적용되게 된다.

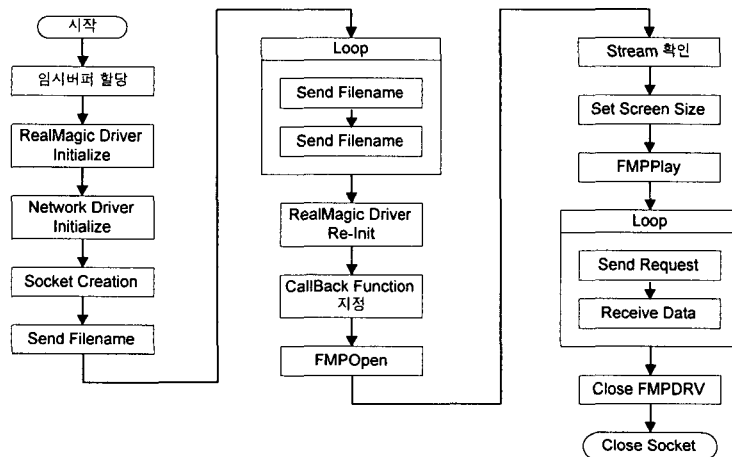


그림 3. 데이터 수신 구조의 블록 다이어그램  
Fig. 3. Block Diagram of Data Receiving Structure

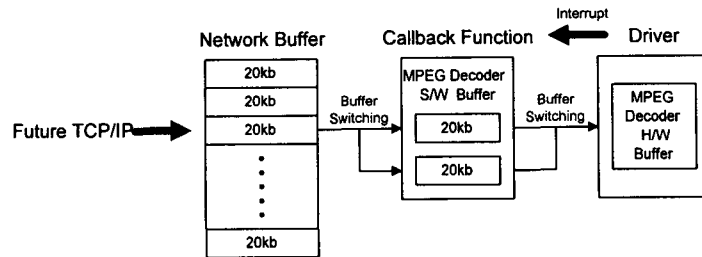


그림 4. VOD 시스템의 버퍼관리  
Fig. 4. Buffer Management of VOD system

일단 CallBack Function이 정의되어 다른 함수들이 호출되어 명령이 전달되면 fmpdrv.h에서 정의된 메시지들이 상황에 따라 전달된다. CallBack Function으로 전달되는 메시지의 종류와 의미는 다음과 같이 정의되어 있다.

- FMPM\_ERROR - 오류 발생
- FMPM\_SIGNAL - 시그널 발생
- FMPM\_COMPLETED - 스트림을 완전히 재생하였을 경우
- FMPM\_CANCELLED - 스트림 재생시 취소하였을 경우
- FMPM\_BUF\_CREATE - 스트림을 버퍼 모드로 오픈하였을 경우
- FMPM\_BUF\_CLOSE - 스트림 닫기
- FMPM\_BUF\_SEEK - 버퍼 내의 지정된 위치 찾기
- FMPM\_BUF\_EMPTY - 버퍼 내용이 비어있을 경우
- FMPM\_BUF\_POS - 버퍼 내에서 지정된 위치로 이동
- FMPM\_MEM\_ALLOC - 메모리 할당
- FMPM\_MEM\_FREE - 메모리 해제

다음은 메모리 요구사항에 대해 알아보자. VOD 시스템은 버퍼 스트림(Buffered Stream)의 형태를 사용하고 있으므로 스트림에 따라 요구되는 드라이버의 메모리 요구량은 없다고 볼 수 있다. 그러나 스트림에 대한 응용 프로그램에서의 버퍼를 할당하여 사용해야 한다. Full Motion Driver에서 요구하는 메모리는 총 82Kb가 요구되고 있다.

그림 4는 VOD 시스템의 버퍼 관리를 나타내는데 크게 DOS 모드에서의 버퍼관리와 RealMagic 드라이버에 의한 인터럽트로 나눌 수 있다.

도스 모드에서의 네트워크 버퍼 관리는 4Kb 크기의 임시 버퍼와 20Kb 크기의 실제 버퍼로 이루어져 있다. 실제 버퍼는 15개를 사용하여 번갈아 가며 사용하게 된다. 현재 4Kb 크기의 임시버퍼를 사용하는 이유는 사용하는 도스용 TCP/IP Kernel의 제한점으로 다른 TCP/IP의 Kernel을 사용할

경우 임시 버퍼를 거치지 않고 바로 5Kb 크기의 실제 버퍼로 데이터를 다운로드 할 수도 있다. 이 임시버퍼를 4Kb보다 적게 하면 스트림을 전송받아 20Kb의 실제 버퍼를 채우는데 걸리는 시간 지연 때문에 이 시간동안 비디오는 끊기게 되므로 4Kb정도가 가장 적합하다.

MPEG 디코더의 소프트웨어 버퍼는 20Kb 크기의 버퍼 2개를 교대로 사용하고 있는데, 버퍼의 구조는 다음과 같다.

```
struct TBuf {
    char Buf[2][20000];
    int BufNb;
}
```

BufNb는 현재 사용하고 있는 버퍼를 가리키고 있으며, Buf는 각 20Kb 크기의 변수 2개로 이루어져 있다. 사용방법은 그림 5과 같다.

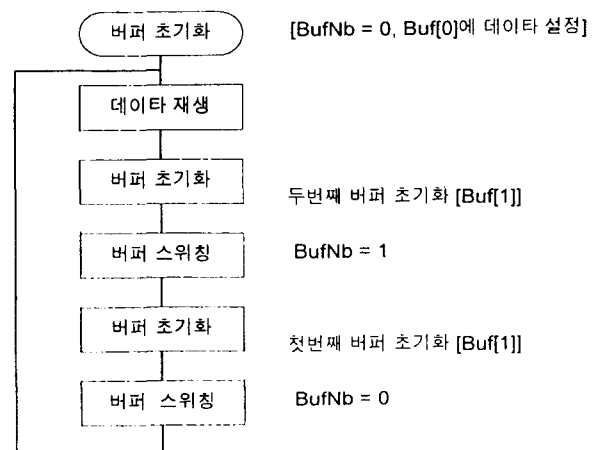


그림 5. 버퍼 전환 구조  
Fig. 5. Buffer Change Structure

이와 같이 두 개의 버퍼를 연속적으로 번갈아 가며 사용하게 되는데 이것은 스트림의 재생 도중에 데이터를 확보한 다음 실제로 하나의 버퍼가 사용된 후 연속적으로 다음 버퍼를 사용할 수 있도록 하기 위해 하나가 아닌 두 개의 버퍼를 사용하는 것이다.

본 논문에서는 MPEG decoder로서 RealMagic을 사용하는데 RealMagic에서는 MPEG 스트림을 decoding하여 화면에 중첩(overlay)시키기 위하여 8Kb의 메모리를 응용 프로그램 버퍼와 별도로 필요로 한다. 이 버퍼는 응용 프로그램에서 조작하지 못하는 것으로서 응용 프로그램의 버퍼를 계속적으로 하드웨어 드라이버의 버퍼로 보내는 작업이 필요하다. 이 작업은 인터럽트를 통해서 이루어지고 있으며 후면 작업으로 진행되기 때문에 하드웨어 버퍼가 비어 응용 프로그램의 버퍼를 다시 읽어들이 필요할 때에는 운영 체제의 재진입이 요구된다.

RealMagic 드라이버는 두 개의 스택을 사용하고 있다. 우선 응용 프로그램 스택이 있는데, 이것은 응용프로그램에서 드라이버로 명령을 전달할 때 사용된다. 보통 최저 2K이상 필요하며 응용프로그램에 따라 5K-8K정도 필요할 수도 있다. 또 다른 스택은 내부 스택인데 이것은 Full Motion 드라이버가 인터럽트를 통하여 후면작업으로 스트림을 재생하거나 기타 작업을 할 때 사용된다.

Netstar를 이용하여 도스상에서 TCP/IP를 사용하려면 우선 패킷 드라이버를 구동시켜야 한다. 그 위에 네트워크 드라이버가 적재되고 그 위에 TCP 커널이 적재되면 TCP/IP가 사용가능한 상태가 된다. 그림 6는 이러한 구조를 나타낸다.

Application	
TCP.EXE	64Kb
SDRV.EXE	8Kb
Packet Driver	6Kb
OS	

그림 6. Netstar TCP/IP의 계층적 구조  
Fig. 6. Hierarchical Structure of Netstar TCP/IP

Netstar의 적재구조는 일반적인 TCP/IP의 계층적 구조를 따르고 있으나, MS-DOS상에서 디바이스 드라이버가 아닌 램상주(TSR)의 형태를 따르고 있기 때문에 상당한 메모리를 차지하게 된다. 단 Conventional Memory가 아닌 Upper Memory에 적재될 수 있으므로 Conventional에서의 메모리의 사용량을 줄일 수 있다. XMS를 사용하지 않는 경우의 메모리 요구량은 총 78Kb정도가 필요하다.

위에서 언급한 데이터 수신 구조와 CallBack Function,

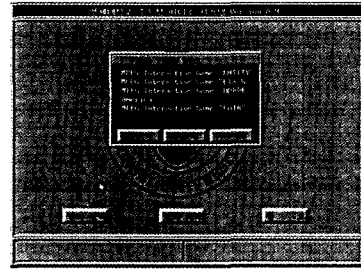


그림 7. 메타 데이터를 전송받는 화면  
Fig. 7. Scene Receiving Meta Data



그림 8. 자료를 선택한 후 스트림을 전송받아 비디오를 보여주는 화면  
Fig. 8. Scene Presenting Transferred Video Data After Selecting Data

메모리 요구량을 고려하여 구현한 VOD 시스템의 화면구성은 그림 7, 8과 같다.

그림 7은 서버에 연결하여 자료선택을 위한 데이터를 전송받는 화면의 구성이다. 서버에 접속을 할 때는 소켓을 이용하여 하는데 데이터 전송을 위해서는 포트 9000에 접속하여 자료선택을 위한 데이터를 전송받고 일단 자료를 선택하면 포트 9001에 접속하여 MPEG 스트림을 전송받는다[9]. 이렇게 메타 데이터의 전달과 MPEG 스트림의 전송 분리하는 이유는 서버가 3개의 독립적인 프로세서로 이루어졌기 때문이다. 즉, 자료를 소비자 프로세서에게 전달하는 생산자 프로세서와 시작 블록 및 생산자 프로세서로부터 전달되는 자료를 실시간으로 클라이언트에게 전달해야 하는 소비자 프로세서, 그리고 자료에 대한 자료인 메타 데이터와 사용자 정보를 가지고 있는 운영센터로 이루어지므로 이를 구현하였기 때문이다. 자료를 선택한 후 비디오를 보여주는 화면은 그림 8과 같다.

## V. 결론

본 논문에서는 LAN을 이용하여 대화형 미디어 서비스인 VOD 시스템을 구현해 보았다. 이 시스템의 프로그램 구현이 응용프로그램 수준에서 이루어졌기 때문에 하드웨어나 운영 체제의 동작을 최적화 할 수는 없었다. 네트워크 이더넷(대역폭 10Mbps)인 점과 MPEG-1의 사용자 소비율이 1.5Mbps인 점을 감안하여 5명의 최대 사용자수를 목표로 하였으나 실제로는 3, 4명의 최대 사용자를 지원할 수 있었다. 이는 서버에서 사용된 하드디스크의 대역폭이 작고 앞에서 언급한 하드웨어 수준이나 운영 체제 수준의 제어가 불가능했기 때문이라 생각된다. 또한 에러 콘트롤이나 사용자 인터페이스, 예외 상황 처리 등의 부분이 미흡한 것이 서버의 문제점이라 할 수 있다.

VOD 시스템에서의 실험결과에서는 프레임의 크기에 따라 전송 버퍼의 크기를 최적화 할 필요가 있었다. 이를 위하여 반드시 스트림 전송 전에 Content Analysis가 선행되어 미리 서버와 클라이언트간에 버퍼의 크기를 정하는 절차가 필요하다. 실험 결과에 따르면 프레임 크기가 176×120인 경우보다 352×240인 경우에서 보다 많은 왜곡현상이 발생하였다. 이는 한 프레임에 대한 정보량의 차이 때문으로 예상되는데, 정보량이 적을수록 깨끗한 영상을 얻을 수 있다.

현재 VOD서비스는 MPEG-1 표준에 관한 응용이 대부분이지만 차후의 VOD서비스는 MPEG-2 표준에 관한 응용을 지향할 것이다. 이는 MPEG-2 표준이 미래에 가능한 서비스까지 고려한 동화상 압축 표준이고 HDTV급까지 수용하는 것이기 때문이다. 하지만 MPEG-2 응용의 비트율에 비하면 아직 작은 가입자 선로의 대역폭 문제 때문에

MPEG-2 응용의 서비스보다는 MPEG-1 응용을 포함하면서 MPEG-2 응용에 준하는 응용의 선택이 필요할 것으로 본다.

## 참고 문헌

- [1] "주문형 비디오 VOD," 전자저널, 1994년 8월.
- [2] 이병기, 강민호, 이종희, "광대역 통신 시스템," 교학사, 1993
- [3] M.Morris Mano, *Computer system architecture*, Prentice Hall, 2th edition, 1982.
- [4] Craig S.Freedom and David J.DeWitt, "The SPIFFI Scalable Video-on-Demand System," *ACM SIGMOD* 95.
- [5] Steven Berson, Leana Golubchik, and Richard R. Muntz, "Fault Tolerant Design of Multimedia Servers," *ACM SIGMOD* 95.
- [6] A.L. Narasimha Reddy, "A study of I/O system organizations," *Proc. of Int. Symp. on Comp. Arch.*, May 1992.
- [7] A.L.N.Reddy and J.C.Wyllie, "I/O Issues in a multimedia system," *IEEE Computer*, vol.27, no.3, Mar 1992.
- [8] *Open RealMagic Application Programming Interface*, Sigma Designs, Inc.
- [9] W. Richard Stevens, *UNIX Network Programming*, Prentice Hall, 1994



저 자 소 개



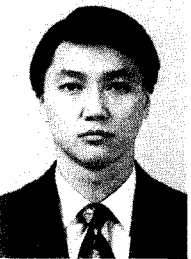
金 闊 帆  
 1971년 12월 24일생  
 1994년 2월 연세대학교 전기공학과 졸업(공학사)  
 1996년 2월 연세대학교 대학원 전기공학과 졸업(공학석사)  
 현재 삼성전자 마이크로 사업부 시스템 팀  
 주관심분야 : VOD system, DSP Chip Design



李 俊 珩  
 1973년 12월 31일생  
 1996년 2월 연세대학교 전기공학과 졸업(공학사)  
 현재 연세대학교 전기공학과 석사과정  
 주관심분야 : VOD system



崔 潤 植  
 1957년 2월 12일생  
 1979년 2월 연세대학교 전기공학과 졸업(공학사)  
 1984년 5월 Case Western Reserve University 시스템공학과 졸업(MSEE)  
 1987년 5월 미국 Pennsylvania State University, University Park 전기공학과 졸업(MS)  
 1990년 12월 미국 Purdue University, West Lafayette 전기공학과 졸업(Ph.D)  
 1990년 11월~1993년 2월 (주)현대전자산업 산업전자연구소 책임연구원  
 1993년 3월~현재 연세대학교 공과대학 전기공학과 조교수  
 주관심분야 : 통계적 신호처리, 컴퓨터 시각



李 晶 洙  
 1980년 2월 한양대학교 공과대학 전자공학과(공학사)  
 1982년 2월 한국과학기술원 전기 및 전자공학과(공학석사)  
 1989년 2월 한국과학기술원 전기 및 전자공학과(공학박사)  
 1990년 2월~1991년 4월 금성통신연구소 선임연구원  
 1992년 10월~1993년 9월 일본 NTT Human Interface 연구소 객원연구원  
 1991년 5월~현재 한국통신 전송기술연구소 가입자전송연구실 VOD 연구팀장  
 주관심분야 : VOD system, 영상처리