

논문 2004-41TC-12-7

# DAB 기반의 EPG Provider 시스템 개발

(Development of the EPG Provider System based on DAB)

진 현 준\*, 박 노 경\*, 황 윤 재\*\*

(Hyun-Joon Jin, Nho-Kyung Park, and Woon-Jae Hwang)

## 요 약

디지털 오디오 방송은 기존의 AM/FM 방송에 이은 고품질의 차세대 라디오 방송으로 CD 수준의 음질, 다양한 데이터 서비스, 양방향성 및 우수한 이동 수신 품질 등을 제공하며 지상파 방송, 위성 방송, 케이블 TV 및 인터넷 등 다양한 매체를 통하여 서비스가 가능한 뉴미디어 서비스이다. 본 논문에서는 DAB(Digital Audio Broadcasting) 시스템이 오디오 방송뿐만 아니라 다양한 멀티미디어 서비스가 가능하다는 장점과 인터넷 단말기로서의 PC를 결합한 새로운 형태의 DAB 기반 EPG(Electronic Program Guide) 응용 기술에 대하여 연구하였다. 개발된 EPSD(EPG Provider System on DAB)시스템은 웹 기반의 서버/클라이언트 구조를 가지며 서버에서 클라이언트 PC로 다양한 EPG 기능을 제공한다. 따라서 DAB 수신기가 보다 저렴하고 소형화될 수 있으며 인터넷과 연결되어 풍부한 데이터 서비스의 개발이 가능하게 된다. 특히 고품질 영상 서비스가 가능하고 기존의 디지털 장비와의 연결 또한 수월하여 추후 홈 네트워크의 중요한 단말 장치로서의 역할을 담당할 것이다.

## Abstract

DAB(Digital Audio Broadcasting) is a new media service that can provide CD quality audio, various data service, interactive and high quality mobile communications through popular media such as terrestrial broadcasting, satellite, cable TV, and internet. In this paper, a new EPG(Electronic Program Guide) application model is proposed. The model is based on DAB and combines a DAB receiver and PCs so that it can take advantages of using various multimedia services and plenty of internet contents. The developed EPSD(EPG Provider System on DAB) has Web-based Server/Client structure and provides EPG functionalities to client PCs over internet. Therefore, the DAB receiver can be smaller and cheaper, and can develop abundant data services on internet. It can also provide high quality video services and be expected to become an important component in future home network systems.

**Keywords :** DAB, EPG, EPSD, Web-based, PC-based

## I. 서 론

21세기 정보화 사회로 접어들면서 고품질의 다양한 디지털 콘텐츠를 즐기고자 하는 소비자의 욕구가 늘어나고 있으며, 세계적으로 방송·통신 및 컴퓨터를 결합하는 멀티미디어 서비스 개발 경쟁이 가속화 되고 있

다. 이동 통신, 위성 통신, 초고속 인터넷 망 등의 다양한 유·무선 통신 시스템들은 디지털화에 성공하여 광대역 멀티미디어 서비스를 제공하고 있으나, 가장 손쉽게 접할 수 있는 텔레비전과 라디오 등의 방송 통신분야는 타 분야에 비해 디지털 시스템으로의 전환이 상대적으로 늦어진 상태이다<sup>[1]</sup>. 이러한 시점에서 CD 수준의 고품질 오디오 서비스, 교통정보 서비스, 일기예보 서비스 등 다양한 부가 데이터 서비스는 물론, 7" 미만 화면에서는 고속 이동 중에도 선명한 화질의 영상정보 수신 이 가능한 차세대 디지털 오디오 방송인 DAB(Digital Audio Broadcasting)가 부각되고 있다<sup>[2][3]</sup>.

DAB 수신기는 자동차용 수신기, Hi-Fi용 또는 Portable 형태의 stand-alone 방식과 PC나 PDA등에 장

정회원, 호서대학교 전기정보통신공학부  
(Department of Electrical, Information and  
Communication Engineering, Hoseo University)

\*\* 정회원, 고려대학교 정보수학과  
(Department of Information & Mathematics, Korea  
University)

※ 이 논문은 2004년도 호서대학교 학술연구조성비에  
의하여 연구되었음

접수일자: 2004년10월4일, 수정완료일: 2004년11월19일

착 할 수 있는 PC/PDA-based 방식으로 구분 할 수 있다. 이중 PC-based 형태는 PC의 메모리와 오디오/비디오 디코더 등 다양한 리소스를 이용할 수 있는 장점을 가지고 있다<sup>[4]</sup>.

본 논문에서는 PC-based DAB 수신기를 구현하였다. 구현된 DAB 수신기는 EPSD(EPG Provider System based on DAB) 형태로 구현되었는데 EPSD는 EPG(Electronic Program Guide)와 결합된 DAB 콘텐츠 응용의 새로운 시스템 모델로 고품질·다채널·양방향 서비스를 디지털 방송을 통해 실현할 수 있다. 따라서 사용자는 정보 선택의 폭을 확장할 수 있으며, 정보수집의 신뢰성을 높일 수 있다. EPSD는 차세대 디지털 방송 프로그램용 인터페이스로 대다수 사용자들의 정보 활용 효율성에 매우 중요한 영향을 미치게 될 것이다.

## II. PC-based DAB 수신기 설계

### 1. Eureka-147

본 논문에서 구현한 DAB 시스템은 Eureka-147 방식으로 ITU-R의 권고안 가운데 Digital System A로 분류되어 있다. Eureka-147은 약 2MHz의 대역폭을 사용하며, CD 수준의 고품질 오디오 서비스가 가능하도록 MPEG Audio Layer II에 기반한 오디오 압축 기술을 사용한다<sup>[3]</sup>. Eureka-147 시스템은 4가지의 전송모드로 구분된다. 현재 우리나라에서의 지상파 DAB 시스템은 전송모드 I 방식을, 위성 DAB 시스템은 전송모드 IV 방식으로 결정하였으며, 본 논문에서 구현한 DAB 수신기는 전송모드 I 방식의 지상파 DAB 시스템을 모델로 채택하였다.

Eureka-147 모드 I의 전송 프레임 구조는 그림 1과 같이 SC(Synchronization Channel), FIC(Fast Informa-

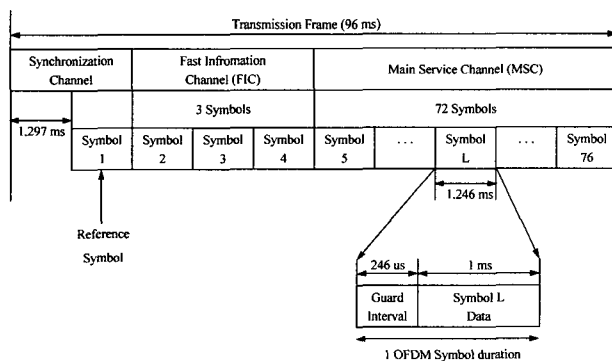


그림 1. Eureka-147 모드 I의 전송 프레임 구조  
Fig. 1. Structure of transmission Mode I for Eureka-147.

tion Channel), MSC(Main Service Channel)로 구성되어 있다. SC는 OFDM의 동기 정보를 가지고있으며, FIC는 MSC의 다중화 정보와 기타 제어용 부가 정보를 가지고 있다. MSC는 오디오 데이터와 부가 데이터 정보로 구성되어 있으며 하나의 전송 프레임 길이는 96ms 이다.

### 2. DAB 수신기 구조

DAB 수신기는 Independent Portable Receiver 로서 Stand-alone 방식과 다른 기기와의 통합을 통한 PC/PDA-based 방식, 예를 들어 PC/PDA 혹은 핸드폰과의 결합으로 구현 될 수 있다.

PC에 기반을 둔 DAB 수신기는 PCI와 USB 인터페이스를 이용 할 수 있는데 사용자의 편의성을 고려하여 본 논문에서는 그림 2와 같이 USB 인터페이스를 이용한 PC-based DAB 수신기를 구현하였다.

PC-based DAB 수신기는 Stand-alone 방식과 비교할 때 PC의 메모리와 오디오/비디오 디코더 등 다양한 PC의 리소스를 이용 할 수 있는 장점이 있다<sup>[4]</sup>. 즉, Stand-alone 방식의 DAB 수신기에 비하여 보다 저렴하고 소형화 될 수 있으며 수신된 디지털 신호를 PC의 고성능 CPU를 통하여 고속으로 처리할 수 있다. 또한 대용량 하드디스크에 방송중인 오디오 파일을 저장 할 수 있으며 인터넷과 연결하여 풍부한 데이터 서비스의 개발이 가능하다<sup>[8]</sup>.

### 3. USB Interface Bridge

PC-based DAB 수신기는 PC의 USB 인터페이스와 DAB 수신기 사이의 통신을 위한 USB 인터페이스 브리지가 필요하다. USB 인터페이스는 신뢰성을 가지고 고속으로 데이터를 전송하는 4가지 방식으로 구별되어진다. 우선순위는 낮지만 오류발생시 재전송을 할 수

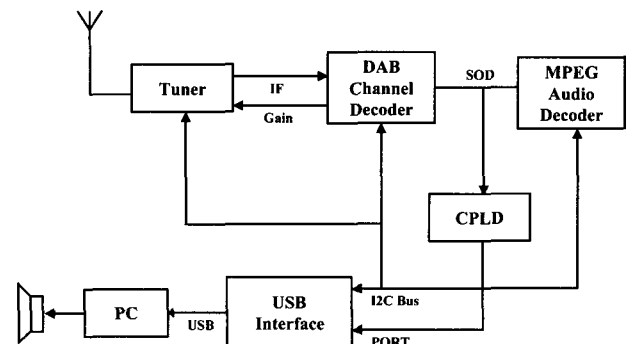


그림 2. PC-based 방식의 DAB 수신기  
Fig. 2. PC-based DAB receiver.

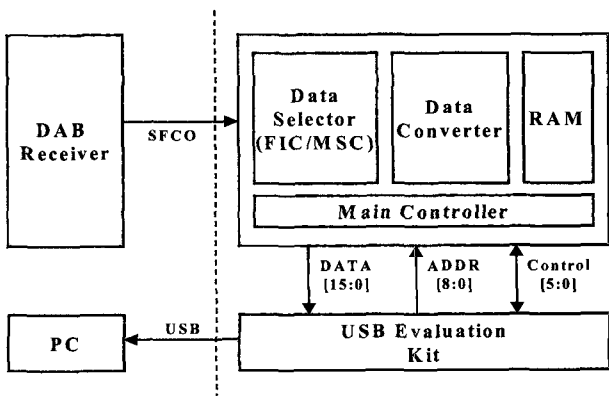


그림 3. 설계한 USB 인터페이스 브리지 구조  
Fig. 3. Structure of designed USB interface bridge.

있는 BULK 방식, 일정 주기에 일정량의 데이터를 전송할 때 사용되는 Isochronous 방식, 주기적으로 소량의 데이터를 처리할 때 사용하는 Interrupt 방식, 그리고 디바이스가 설정 정보를 호스트에 전송하는 Control 방식이 있다<sup>[5][6]</sup>.

DAB 수신기를 통하여 수신된 실시간 데이터를 PC에 전송하기 위해서 USB 인터페이스 전송 방식중 Isochronous 방식으로 선택하였다. 그림 3은 설계된 USB 인터페이스 브리지를 나타낸 것이다.

DAB 수신기로는 Atmel사의 DRK-026 개발 키트가 사용되었는데 이 장비는 Stand-alone 방식만 지원하기 때문에 별도의 외부 데이터 포트가 없으므로 DAB 수신기 내의 채널 디코더 출력 인터페이스 중에서 SFCO(Simple Full Capacity Output) 출력 신호를 이용하여 DAB 수신기에서 디코딩된 데이터를 얻었다<sup>[7]</sup>. 이 데이터는 USB 인터페이스 브리지를 통하여 USB 인터페이스 데이터 형태에 맞게 포매팅 된 후 PC의 USB 인터페이스로 전송된다.

### III. EPSD 시스템 설계

#### 1. EPSD 시스템 구성

EPSD 시스템 구성은 크게 서비스를 요청하는 클라이언트와 서비스를 수행하는 서버로 구분된다. 아래의 그림 4은 EPSD 시스템의 전체 구성도를 나타낸 것이다. 그림에서와 같이 클라이언트는 수신되는 미디어 매체에 따라 모바일장치(휴대폰, PDA)와 일반 PC로 구분할 수 있다.

EPSD 시스템에서 전체 시스템을 구성하는 관리자 객체는 세션관리를 위한 CM (Connector Manager), 프로토콜 분석 및 처리를 위한 PAM (Protocol Analysis

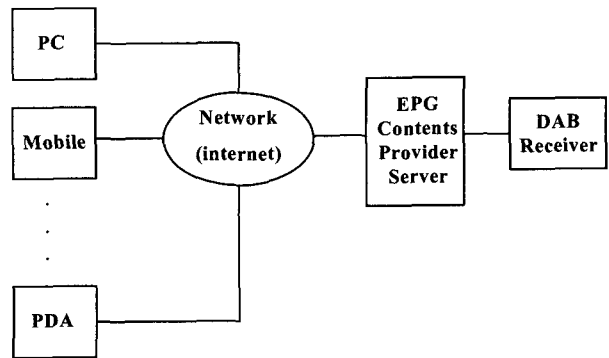


그림 4. EPSD 시스템 구성도  
Fig. 4. Structure of EPSD system. w 6.

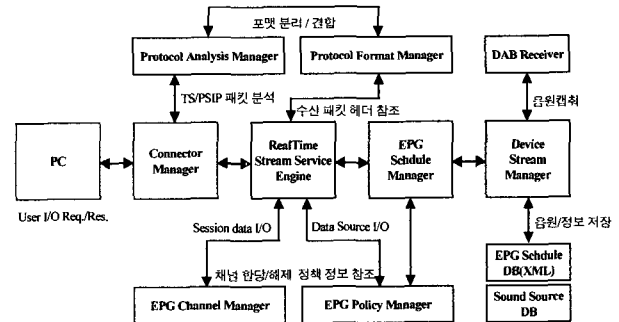


그림 5. EPG Contents Provider 서버의 전체 구성도  
Fig. 5. Server structure for EPG provider system.

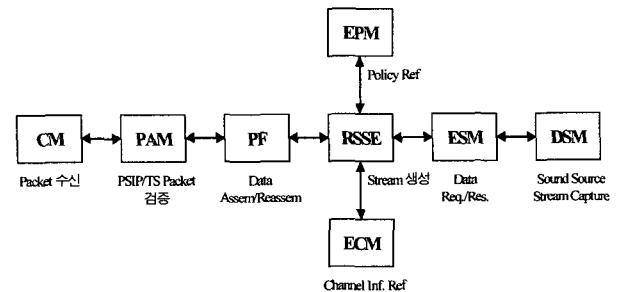


그림 6. EPSD 시스템 전체 처리 과정  
Fig. 6. Process flow of EPSD system.

Manager), PFM(Protocol Format Manager)과 스트리밍 입·출력 및 제어를 위한 RSSE (RealTime Stream Service Engine), 그리고 프로그램 스케줄 및 EPG 서비스의 정보 관리를 위한 ESM (EPG Schedule Manager), ECM(EPG Channel Manager), EPM(EPG Policy Manager)로 구성된다. 그리고 음원 정보를 제공 및 관리하기 위한 DSM(Device Stream Manager)은 실제 음원을 캡처하기 위해 DAB 수신기와 캡처된 음원 정보를 저장·관리하고 있는 EPG Schedule DB, Sound Source DB와 연동된다. 그림 5는 이러한 EPG Contents Provider Server 시스템 전체 구성도를 나타내고 있다.

EPSD 시스템은 추상화된 장치 관리자 DSM으로부

```
//서비스를 위한 서버 포트의 사용유무를 스캔한다.
if(!PortScan(iPort))
{
    registerErrLog("[오류]현재 서비스 포트는 사용중입니다.");
    return;
}
//End of if
socServer = new ServerSocket(iPort);
createConnectorVector(10); //커넥터 처장을 위한 벡터를 선언한다.
//접속자 연결 처리를 수행한다.
while(true)
{
    try {
        socConnector = socServer.accept();//소켓 연결 설립
        while(!CMMM.getEnableAllocateObject())
            //스레드 할당이 불가능하다면
        {
            KillZombiThread(); //좀비 스레드 검색 및 제거
            OptMemory(); //가비지 컬렉터 호출 및 메모리 풀 최적화
        }
        //End of while
        //RSSE 엔진 할당 및 가동
        RSSE rsse = new RSSE(eServer, socConnector); //RSSE 생성
        rsse.begin(); //RSSE 가동
        //RSSE 엔진 벡터 등록
        eServer.addRSSE(rsse);
    }catch(Exception ecp)
    {
        registerErrLog("[Exception]" +ecp);
    }
}
//End of try-catch
}
//End of while
```

그림 7. CM의 클라이언트 접근 제어  
Fig. 7. Connection management of CM.

터 음원정보를 수신하여 RSSE를 이용해서 스트리밍 처리를 수행한다. 그림 6은 EPSD 시스템의 전체 처리 과정을 나타내고 있다.

2. EPSD 시스템의 구성 모듈

효과적인 유지 보수를 위해 EPSD 시스템을 관리자 객체로 설계했다. EPSD 서비스를 위한 클라이언트 및 서버의 내부 구성은 모듈간의 결합도를 최소화시켜 서버의 각 처리 모듈들이 병행 처리되도록 하였다. 또한 동일 기능을 수행하는 객체는 필요에 따라 상속받아 적용 가능하게 함으로써 이식성을 극대화 하였다.

가. CM 모듈

CM은 접속자 세션 관리를 위해 스레드 생성 및 제거를 한다. 이러한 스레드의 생성 및 제거는 자바의 JVM(Java Virtual Machine)이 이행한다. 그러나, 메모리 풀에서 해제되어 반환되지 않은 Zombi 스레드가 발생하여 시스템 메모리 누수의 주요 요인이 되기도 한다. 일반적인 네트워크 응용에 비해 많은 트래픽을 필요로 하는 EPSD 시스템은 서버와 연결된 각 클라이언트의 접속자 세션 스레드를 직접 감시할 필요가 있다.

본 논문에서는 CM이 연결된 세션에 대한 정보를 저장하며, 생성 및 해제 스레드를 자바 벡터를 이용하여

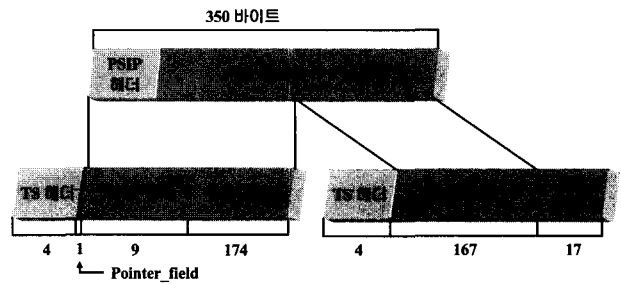


그림 8. TS/PSIP 패킷의 구성  
Fig. 8. Structure of TS/PSIP packet.

```
//TS 헤더, TS의 Payload
byte[] buf_TShead, buf_TSbody;
buf_TShead = getTSHeader(buf);
//첫번째 TS
if((getProtocol(buf_TShead) == "TS") && (iPayload_unit_start_indicat
or == 1)) {
    //시작 Payload 카운터 초기화
    iContinuity_counter = 0;
} else if((getProtocol(buf_TShead) == "TS")
    && (iPayload_unit_start_indicat == 0)) //연속되는 TS
{
    //이전 TS의 PSIP 다음 순서라면
    if(((iPrevContinuity_counter+1)%16) != iContinuity_counter)
    {
        eServer.registerErrLog("TS 패킷이 누락되었습니다.");
        return -1;
    }
} else
    return 0; //다른 TS 프레임
//TS 내부의 PSIP 스트림을 PFM에게 전달
buf_TSbody = getTSBody(buf);
//PFM에게 PSIP 프레임 전달
eServer.pfm.setTSBody(new PSIPSection(buf_TShead, buf_TSbody);
return 0;
```

그림 9. PAM 클래스 멤버  
Fig. 9. Members of PAM class.

관리하며, Zombi 스레드나 Idle 스레드를 검색하여 제거함으로써 메모리 손실을 최소화 하였다.

CM은 자바 클래스로 형상화 하였다. 그림 7는 CM 클래스의 접속자 관리 스레드 실행 중 접근 제어의 처리 과정을 나타낸 것이다.

나. PAM과 PFM 모듈

EPG는 ATSC A/65 표준 스펙에 나와 있으며, PSIP (Program and System Information Protocol)로 표현이 된다. 이러한 PSIP를 이용하여 EPG는 TS(Transport Stream)를 생성하게 되고, 형성된 TS는 PSIP 패킷의 헤더를 통해 제어하게 된다<sup>[9][10]</sup>. 그림 8은 TS/PSIP 패킷의 구성을 간단히 나타낸 것으로 PSIP의 데이터의 크기가 180 바이트 보다 커지면 1 개 이상의 TS를 전

```

psipHead = new PSIPHead(payload); //PSIP 헤더 획득
//TS의 PID가 "1ffb", TS의 Table_id가 "c7"이면 MGT 테이블
while(psipHead.nextTS()) { //188 바이트 TS 획득
    if(psipHead.getPID() == eServer.Math.DEC("1ffb")) {
        if(psipHead.getTableID() == eServer.Math.DEC("c7")) {
            //version_number_check
            if(CheckVersionNumber(psip) == true)
                vPayload.addElement(makeMGT()); //MGT 테이블 구성
        } else if(psipHead.getTableID() == eServer.Math.DEC("ce"))
        {
            vPayload.addElement(makeSTT()); //STT 테이블 구성
        } else if(psipHead.getTableID() == eServer.Math.DEC("c8"))
        {
            vPayload.addElement(makeVCT()); //VCT 테이블 구성
        } else if(psipHead.getTableID() == eServer.Math.DEC("ca"))
        {
            vPayload.addElement(makeRRT()); //RRT 테이블 구성
        }
    }
}
//End of while
else {
    if(psipHead.getTableID() == eServer.Math.DEC("cb"))
        vPayload.addElement(makeEIT()); //EIT 테이블 구성
    else if(psipHead.getTableID() == eServer.Math.DEC("cc"))
        vPayload.addElement(makeETT()); //ETT 테이블 구성
}
//End of else
}
//End of while
    
```

그림 10. PAM의 테이블 생성  
Fig. 10. Creation of PCM tables.

송해야 한다.

PAM과 PFM은 이러한 EPG 시스템의 프로토콜 분석 및 처리를 위한 모듈로서, PAM에서는 PSIP와 TS의 헤더 정보를 분석하여 EPG 시스템의 정보 표현을 위한 데이터로 판단이 되면 PFM으로 수신된 데이터를 전송하게 된다. 1개 이상의 TS로 나누어져 전송된 PSIP 데이터는 PFM에서 조립하거나 분해를 통하여 필요한 정보를 추출하게 되는 것이다. 아래의 그림 9와 10은 이러한 PAM의 클래스 멤버와 각 PFM의 동작 처리 과정을 보여주고 있다.

다. RSSE 모듈

RSSE는 음원 데이터를 수신받아 동기화를 수행한다. 동기화 시간이 초과된 패킷은 폐기처분한다. 트래픽 수준과 샘플링 수준에 따라 자율적으로 스트리밍 크기를 조절하며, 서비스를 위한 보안 정책 적용 및 EPM과 연동하여 스트리밍의 적절성 여부를 결정한다. 그림 11은 RESS와 관련모듈의 연동처리과정을 간략히 나타내고 있다.

RSSE는 요청된 서비스의 현재 채널 사용 여부 및 예약 할당 여부를 분석하는 기능을 수행하는 시스템의 스트리밍 서비스를 위한 핵심 처리 객체이다. 그림 12는 RSSE의 수행 과정을 나타내고 있다.

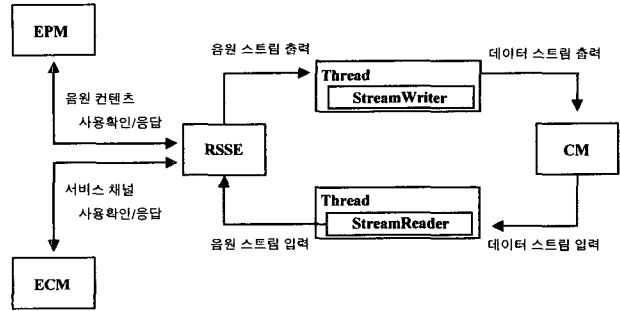


그림 11. RSEE와 ECM/EPM 연동 처리 구성도  
Fig. 11. Processing of RSEE with ECM/EPM.

```

//RSSE 초기화
initRSSE():

//EPG 서비스 채널이 사용 가능한지 확인
if(eServer.ecm.checkEnableChannel(iServiceChannel) == false)
{
    eServer.registerErrLog("서비스 채널이 사용 불가능합니다.");
    return;
}
//End of if

//EPG 정책상 사용 가능한 상태인지 확인
if(eServer.epm.checkEnableService(rsseID,iServiceChannel) == false)
{
    eServer.registerErrLog("현재 채널은 시스템 정책상 제한되고 있습니다.");
    return;
}
//End of if

//I/O 스트림 초기화
streamReader = new StreamReader(soc.getInputStream());
streamWriter = new StreamWriter(soc.getOutputStream());

//I/O 스트림 스레드 기동
streamReader.start();
streamWriter.start();
    
```

그림 12. RSSE의 I/O 처리  
Fig. 12. I/O processing of RSSE.

라. ECM, EPM, ESM 모듈

ECM, EPM, ESM은 EPG 시스템의 프로그램 스케줄 및 서비스 정보 관리를 하는 모듈이다. EPG 시스템에서는 네트워크상에서 사용자들에게 음원 재생 서비스를 프로그램 가능하도록 양방향 서비스를 제공한다. 그리고 다수의 논리적 채널을 두어 서로 다른 콘텐츠를 선택적으로 동일한 시간대에 제공해 주는 역할을 담당한다.

표 1은 ECM 클래스의 주요 멤버들을 나타내고 있다. ECM은 다수의 논리적 채널을 효과적으로 관리하기 위해 사용자 지정시 해당 시간대 음원 재생 스케줄의 재생 시간대를 모니터링하고, 지정된 요청에 유효한 재생 위치를 검색하여 음원 스트림을 송신하는 역할을 한다.

표 2는 EPM 클래스 구성 멤버를 나타낸 것으로 EPG 시스템의 효과적인 자원 관리와 사용자의 무한정

표 1. ECM 클래스 구성 멤버

Table 1. Members of ECM class.

클래스 멤버	내용
public EPGSERVER eServer;	EPG 서버 인스턴스
public boolean[] bChannelEnable;	EPG 서비스 채널 가용 상태
public int iChannelNum;	전체 가용 채널 수
public ECM (EPGSERVER eServer, iChannelNum)	ECM 객체 생성자
setUseChannel (int iServiceChannel)	iServiceChannel 채널 사용 설정
public boolean checkEnableChannel (int iServiceChannel)	iServiceChannel 채널 사용 여부 확인

표 2. EPM 클래스 구성 멤버

Table 2. Members of EPM class.

클래스 멤버	내용
public SchduleDBManager schDBManager;	EPG Schdule DB 관리자 객체
public SourceDBManager srcDBManager;	Sound Source DB 관리자 객체
public DSMInfo dsmInfo;	Device Stream Manager I/O 정보 저장 객체
public PolicyInfo pInfo;	EPG 정책 플래그 리스트 추상화 객체
public void connectDB()	시스템 DB 연결
public void loadDSMInfo()	추상화 장치 관리자 정보 로딩
public boolean initEPM()	EPM 정책 정보 로딩
public String getEPMFlag (String sFlagName)	정책 결정 플래그 참조
public boolean setEPMFlag (String sFlagName, String sValue)	정책 결정 플래그 설정

표 3. ESM 클래스 구성 멤버

Table 3. Members of ESM class.

클래스 멤버	내용
public EPGChannelSchdule[] channelSchdule	각 채널별 스케줄 정보
public XMLSchduleDoc xmlDoc	XML 스케줄 Document
public boolean makeXMLSchduleDoc()	XML 음원 정보 문서 생성
public Vector getChannelSchdule (int iServiceChannel)	해당채널의 서비스 스케줄 일정 반환
public String searchChannelSchdule (int iServiceChannel, SchduleItem si)	해당채널의 서비스 일정 검색
public boolean setConnectStream (int SourcePath)	음원 정보 참조를 위한 스트림 연결 설정
public boolean loadChannelSchdule (int iServiceChannel)	채널별 스케줄 정보 로딩

한 자원 접근을 막기 위한 제약 정보를 가지고 있다. 즉, RSSE에서는 사용자에게 서비스 요청을 받은 음원 재생 스트림을 개방하기 전에 EPM으로부터 처리 가능 여부를 검증 받아야 한다. 권한을 벗어난 요청이거나 시스템의 환경 제약에 적절하지 않은 요청인 경우에는

표 4. EPG 스케줄 DB의 테이블

Table 4. Table of EPG schedule DB.

필드명	자료형	키옵션	필드내용
sid	int	Primary Key	스케줄 구분자
srequester	varchar(50)	Not NULL	스케줄 요청자
sourceid	varchar(50)	Not NULL	음원 구분자
sdate	date	Not NULL	스케줄 수행 일자
stime	time	Not NULL	스케줄 수행 시간
splaytime	long	Not NULL	음원 재생 제한시간(단위:분)
schannel	smallint	Not NULL	서비스 채널
sposition	varchar(50)	Not NULL	서비스 위치
smedia	varchar(50)	Not NULL	서비스 매체

표 5. Sound Source DB의 테이블

Table 5. Table of sound source DB.

필드명	자료형	키옵션	필드내용
sourceid	int	Primary Key	음원 구분자
sourcename	varchar(50)	Not NULL	음원 이름
company	varchar(50)	Not NULL	음원 제작사
ownership	varchar(500)	Not NULL	음원 소유권
copyright	text	Not NULL	음원 저작권 정보
sourcelevel	smallint	Not NULL	음원 등급
firstplaydate	date	Not NULL	음원 방영일자
firstplaytime	time	Not NULL	음원 방영시간
fullplaytime	long	Not NULL	총방송시간
sourceformat	varchar(50)	Not NULL	음원포맷
summary	varchar(50)	NULL	음원 정보 요약

EPM이 RSEE의 스트림 I/O 동작을 통제하게 된다.

표 3은 ESM 클래스의 주요 멤버들을 나타내고있다. ESM은 EPG 관련 음원 재생 스케줄 정보를 ESM에서 관리하여, 실시간 음원 재생을 위해 DSM과 연동한다. 그리고 음원 스트림을 직접 개방하고, Sound Source DB에 DSM으로부터 재생된 스트림을 저장한다. EPG Schdule DB에 저장된 음원의 정보는 XML(eXtensible Markup Language) 자료 포맷으로 저장하여 음원 참조 시 음원에 대한 정보를 제공한다.

마. EPG Schedule DB와 Sound Source DB

저장되는 음원 정보는 고속의 정보 검색이 가능해야 하며, 검색된 정보를 가공 할 수도 있어야 한다. EPG Schedule DB는 Sound Source DB의 음원 정보 위치를 저장하고 있으며, 각 음원 정보는 레코드 단위로 저장 된다.

EPG 스케줄 DB 테이블은 표4와 같이, Sound Source DB 테이블은 표5와 같이 정의 하였으며 음원에 대한

세부 정보는 EPG 스케줄 DB 테이블 내에 체계적으로 XML 문서 포맷으로 저장되어 있다. 이러한 XML 정보를 참조하여 스케줄된 음원의 위치와 정보를 신속하게 검색할 수 있다.

#### IV. EPSD System 구현 및 실험 결과

##### 1. PC-based DAB 수신기 구현

본 논문에서는 USB 인터페이스를 이용한 PC-based DAB 수신기를 구현하였다. 구현된 DAB 수신기는 EPG 시스템에서 DSM 역할을 수행하는 것으로 동작을 확인 하였다.

PC에서 USB 인터페이스를 통하여 입력되는 데이터는 6개의 S/W 모듈을 통하여 재생을 하게 되는데 아래의 그림 13은 PC 소프트웨어 구성도를 나타낸 것이다.

S/W Receiver Driver는 EZ-USB 개발 키트로부터 스트리밍 데이터를 ISO 모드로 수신한다. Packet Analysis Manager는 수신된 패킷의 포맷과 형식을 구분하기 위한 객체이다. 스트림 매니저는 ISO 스트림 연결을 관리 및 지능형 버퍼링 기능을 제공하며, 스트림 타이밍 매니저는 제한된 시간 이내에 도착하지 않은 데

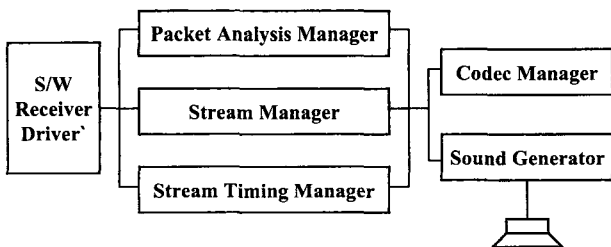


그림 13. PC 소프트웨어 구성도  
Fig. 13. Diagram of PC software.

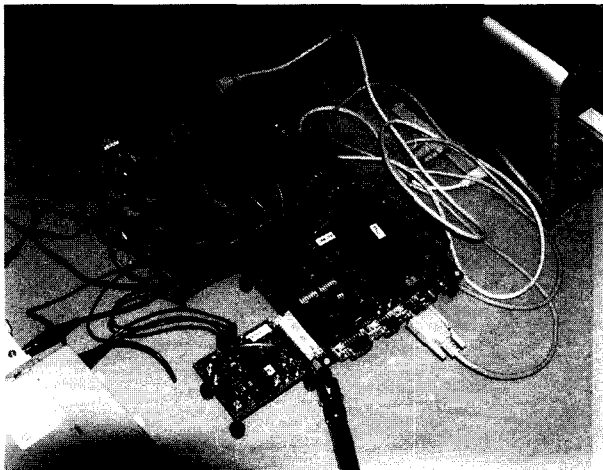


그림 14. 구현된 PC-based DAB 수신기  
Fig. 14. Implemented PC-based DAB receiver.

이터를 폐기한다. Codec Manager는 Packet Analysis Manager에서 분석 및 해석된 데이터를 적합한 코덱으로 링크하기 위해 사용된다. Sound Generator는 수신된 데이터와 Codec Manager에서 링크된 Codec을 이용하여 사운드 재생 및 각종 사용자 인터페이스 기능을 제공하게 된다.

그림 14는 구현된 PC-based DAB 수신기이다. DAB 수신기는 DRK-026 개발 키트로 구현을 하였으며, 수신기의 컨트롤은 DRK-026 개발 키트의 특성상 I2C 인터페이스를 이용하여 PC에서 직접 제어 한다.

##### 2. EPSD System 구현

EPSD의 전송 데이터는 TS 헤더와 PSIP 헤더를 공통적으로 가진다. 이러한 헤더를 효과적으로 표현하기 위하여 그림 15 과 같이 클래스 계층을 구성 하였다.

각 테이블을 추상화한 클래스는 실제 처리를 위해 각 클래스의 멤버를 가지게 되고 각 멤버 변수는 스펙의 규정에 따라 크기와 형이 결정되었다. 위의 클래스를

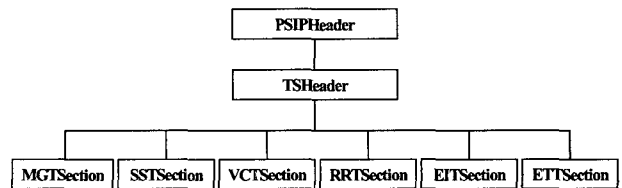


그림 15. 클래스 계층도  
Fig. 15. Diagram of class hierarchy.

```

0  객체 초기화
1  sync 수행(각 바이트의 0x47 이용)
2  while(연결종료가 아니면)
3  {
4      1개의 TS를 획득(188바이트)
5      if(스트림의 끝)
6      {
7          객체 처리를 종료한다.
8      }else if(PID == 0x1FF8)
9      {
10         if(MGT인가)
11         {
12             if(version_number_check)
13             {
14                 MGT 구성
15                 //End of if
16             }else if(STT인가?)
17             {
18                 STT를 구성한다.
19             }else if(VCT인가?)
20             {
21                 VCT를 구성한다.
22             }else if(RRT인가?)
23             {
24                 RRT를 구성한다.
25             }//End of if
26         }else if(EIT인가?)
27         {
28             EIT를 구성한다.
29         }else if(ETT인가?)
30         {
31             ETT를 구성한다.
32         }//End of if
33     }//End of while
    
```

그림 16. PSIP 추출 알고리즘  
Fig. 16. PSIP extraction algorithm.

이용하여 TS로부터 PSIP를 추출하고 분석을 해야 하는데 TS는 PES(Packetized Elementary Stream)와 PSIP 섹션으로 구성되어 있다. 본 논문에서 TS로부터 PSIP를 추출하기 위한 알고리즘은 그림 16과 같다.

위의 알고리즘에서 추출되는 정보중 STT, MGT, RRT, VCT는 각각 고유한 PID와 Table\_Id를 이용하여 얻을 수 있지만, EIT와 ETT는 PID 값이 정해져 있지 않기 때문에 우선 MGT를 분석하여 EIT와 ETT의 PID를 추출해야 한다. 그러므로 EIT와 ETT를 구성하기 전에 먼저 MGT 섹션을 처리해야 한다. 그림 17은 MGT 섹션을 처리하기 위한 과정을 나타내고 있다.

EPSD 콘텐츠 제공 서버는 서버내 DSM을 이용하여 DAB 수신기로부터 음원을 얻어 사용자 요청시 직접 서비스 예약 클라이언트에 대해 실시간 멀티캐스팅을 실행하게 된다. 본 논문에서 구현된 EPSD 콘텐츠 제공 서버는 동시에 5개의 실시간 음원 스트림 제공 서비스로 제한된다. 그림 18은 사용자 측에 GUI로 제공되는 EPSD 시스템 인터페이스로 구현된 EPSD 시스템은 실시간 멀티캐스팅, 음원 캡처, 프로그램 검색, 채널 관리

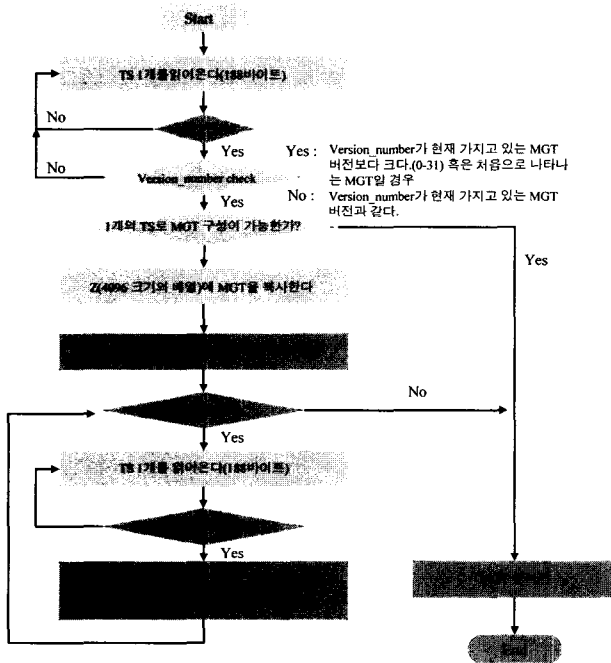


그림 17. MGT 처리 순서도  
Fig. 17. Flowchart of MGT.

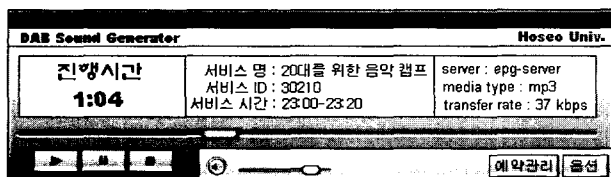


그림 18. 구현된 EPG 시스템 인터페이스  
Fig. 18. Implemented EPG system interface.

등 다양한 기능을 제공한다.

### 3. 실험 결과

#### 가. PC-based DAB 수신기의 실험 결과

설계된 PC-based DAB 수신기의 동작여부 확인은 DRK-026 개발 키트에서 SFCO를 통하여 전송되어진 데이터가 FPGA 모듈에서 패킷 형태로 변환이 되어 EZ-USB의 USB 인터페이스를 통하여 PC에 전송이 되고, 전송된 데이터는 PC에서 디코딩 기능을 수행하는 것으로 확인 하였다. 구현된 FPGA 모듈의 동작 파형은 그림 19, 20과 같다.

그림 19의 파형에서는 DAB 수신기에서 SFCO를 통해 FPGA 모듈에 입력이 되는 데이터를 FIC와 MSC로 구분해 주고 있는 것을 확인 할 수 있다. 그림 20의 파형에서는 EZ-USB로 전송 되어질 FIC와 MSC 신호가 16bit 데이터로 패킷화 되어 있는 것을 나타낸다.

그림 21은 PC에서 구현된 제어 프로그램을 나타낸 것으로 구현된 USB 인터페이스를 통하여 PC에 전송되는 신호를 스트림 형태로 읽고 분석하여 디코딩 기능을 수행하는 것을 확인 할 수 있다.

#### 나. EPSD System 실험 결과

본 논문에서 구현된 EPSD 시스템의 동작여부는



그림 19. FIC와 MSC 신호를 분류  
Fig. 19. Selection of FIC and MSC.



그림 20. 데이터 패킷화 된 출력신호  
Fig. 20. Output signals of packetized data.

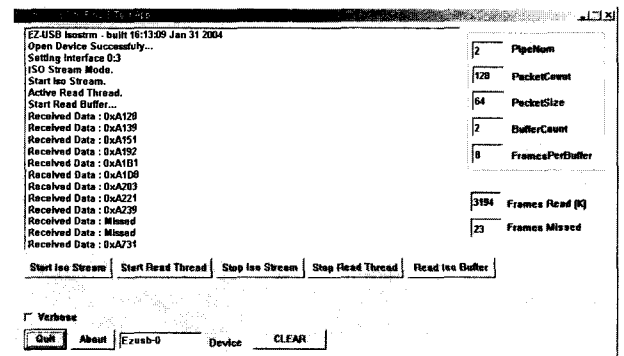


그림 21. 구현된 제어 프로그램  
Fig. 21. Implemented control program.



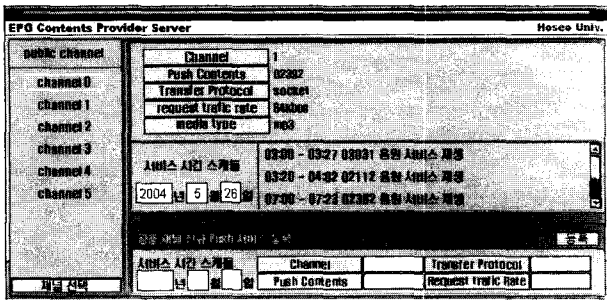


그림 22. 서비스 설정 및 상태 검색 윈도우  
Fig. 22. Service setting and searching window.

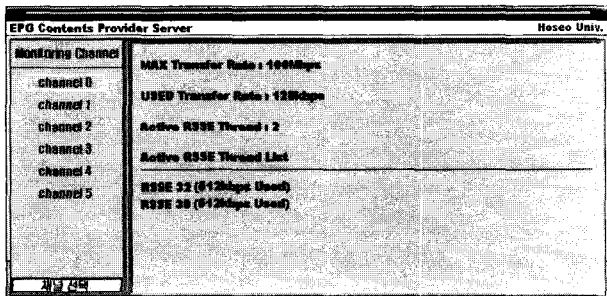


그림 23. 채널 트래픽 모니터링 윈도우  
Fig. 23. Monitoring window of channel traffic.

EPSD 서버로부터 사용자가 콘텐츠를 제공 받는 것을 확인하여 검증하였으며 서버는 현재 접속하고 있는 사용자를 실시간 모니터링 하도록 하였다.

사용자에게 제공되는 서비스는 실시간 멀티캐스팅, 예약된 채널의 데이터를 제공하는 음원 스트림 Push 서비스, 프로그램 검색 등 서비스이다. 위의 그림 18의 EPSD 시스템 인터페이스를 통하여 아래의 그림 22에서와 같이 연동되어 동작되는 것을 확인 하였다.

그림 23은 EPSD 콘텐츠 제공 서버의 서비스 채널의 트래픽을 적절히 배분하기 위해 EPSD 서버에 개방된 각 채널의 트래픽을 실시간 모니터링 한다. EPSD 콘텐츠 제공 서버에 등록된 각 콘텐츠는 사용자로부터 이용된 횟수와 전체 서비스의 이용률을 바탕으로 시스템 운영 정책에 반영 된다.

### V. 결 론

본 논문에서는 USB 인터페이스를 이용한 PC-based DAB 수신기를 구현하고, EPSD 시스템을 개발 하여 DAB 서비스의 새로운 PC 기반의 모델을 제시하였다.

EPSD 시스템은 PC-based DAB 수신기의 성능을 최대한 이용하여 고품질 영상 및 다양한 멀티미디어 서비스, 인터넷을 기반으로 하는 콘텐츠를 제공한다. 특히 DAB 수신기의 간결한 구조와 객체화를 통한 이식성을

극대화 하였다.

현재 DAB 뿐만 아니라 다양한 영상 서비스도 가능한 DMB(Digital Multimedia Broadcasting)로의 전환이 이루어지고 있는 시점에서 EPSD 시스템은 PC-based DMB 수신기의 활용 모델로 제시된다.

향후 연구 과제는 EPSD 시스템의 다양한 디지털 기기들과 연결로 홈 네트워크의 관리 시스템을 개발하는 것이다.

### 참 고 문 헌

- [1] Walter Tuttlebee, "Enabling technologies of Software Defined Radio," John Wiley&Sons, LTD, 2002.
- [2] Walter Tuttlebee, "Origins, Drivers and International Perspectives of Software Defined Radio," John Wiley&Sons, LTD, 2002.
- [3] Wolfgang Hoeg and Thomas Lauterbach, "Digital Audio Broadcasting," John Wiley&Sons, LTD, 2001.
- [4] Darran Nathan, Bernhard Spath, Oliver Faust and Chua Beng Koon, "Design and features of an intelligent PC-Based DAB receiver," *IEEE Transactions on Consumer Electronics*, vol.48, No 2. p.322-328, 2002.
- [5] 김영훈 편저, "USB Guide," Ohm사, p.339-446, 2002.
- [6] "EZ-USB FX2 Technical Reference Manual," Cypress, 2001.
- [7] "DAB Baseband Demoboard Manual," Atmel, 2001.
- [8] Jaehee Cho, Namshin Cho, Keukjoon Bnag, Myunghee Park, Heeyoung Jun, Hyuncheol Park, "PC-Based Receiver for Eureka-147 Digital Audio Broadcasting," *IEEE Transactions on Broadcasting*, vol.47, No 2. JUNE 2004.
- [9] P. Davies, "Guiding light/Technology-EPG," *IEEE Transactions on Broadcasting*, pp.30-34, 2000.
- [10] Marijan, "Internet search for TV content based on TV anytime," *IEEE Transactions on Broadcasting*, pp.70-74, 2003.

저 자 소 개



**진 현 준**(정회원)  
 1984년 고려대학교 전자공학과 졸업(학사).  
 1986년 고려대학교 전자공학과 졸업(석사).  
 1998년 Lehigh University, 전산학(공학박사).

1998년~현재 호서대학교 전기정보통신공학부 부교수  
 <주관심분야: 시스템프로그래밍, 임베디드시스템>



**박 노 경**(정회원)  
 1984년 고려대학교 전자공학과 졸업(학사).  
 1986년 고려대학교 전자공학과 졸업(석사).  
 1991년 고려대학교 전자공학과 졸업(공학박사).

1990년~현재 호서대학교 전기정보통신공학부 교수  
 <주관심분야: VLSI 설계, 멀티미디어>



**황 운 재**(정회원)  
 1985년 서울대학교 수학과 졸업(학사).  
 2000년 SUNY Stonybrook 응용수학과 졸업(이학박사).  
 2002년~현재 고려대학교 정보수학과 조교수

<주관심분야: 응용수학, 수치해석>