

웹 브라우저를 이용한 홈 네트워크 관리 시스템의 설계 및 구현

오 봉 진[†] · 김 채 규^{††}

요 약

본 논문은 웹 브라우저를 통해 관리되는 홈네트워크 관리 시스템의 설계 및 구현에 대해 기술한다. 디지털 가전기기들은 IEEE1394 버스에 의해 퍼스널 자바 환경을 갖춘 셋톱박스과 연결된다. 셋톱박스에서 동작하는 내장형 웹 서버는 HTTP1.1의 세션 유지 기능과 JSDK2.1을 지원 하는 서블릿 엔진을 포함한다. 홈 네트워크 관리 프로그램은 서블릿 형태로 개발되며 IEEE1394 디바이스 드라이버 기능을 접근하기 위해 JNI 를 사용한다. 각 가전기기를 위한 사용자 인터페이스 애플릿은 홈 네트워크 관리 웹 문서에서 관련 아이콘을 선택한 경우 클라이언트로 다운로드 되고, TCP/IP API를 통해 관련 서블릿과 통신한다.

The Design and Implementation of a Home Network Management System through a Web Browser

Bong-jin O[†] · Chae-kyu Kim^{††}

ABSTRACT

This paper describes the design and implementation of a home network management system through a web browser. All digital devices are connected to a set-top box on which personal java environment is installed through an IEEE1394 serial bus. The embedded web server on the set-top box supports persistent-connection of HTTP1.1 and servlet engine for JSDK2.1. The home network management system is developed as the form of a servlet and uses JNI to access IEEE1394 device driver's APIs. When a client selects an icon related with a device the user interface applet of the device is downloaded into a client's web browser and communicates with a related servlet through TCP/IP APIs.

키워드 : 홈 네트워크(Home Network), IEEE1394, 내장형 소프트웨어(Embedded Software), 셋톱박스(Set-top box)

1. 서 론

1960년대 말 최초로 인터넷이 등장하고 두 대의 컴퓨터를 연결하여 정보를 교환할 수 있음을 보여줬을 때, 그것은 많은 컴퓨터 전문가들에게조차 놀라운 일로 받아들여졌다. 그러나 국방이나 학술 등의 전문적인 목적으로 사용되던 인터넷이 월드 와이드 웹(WWW)의 등장으로 일반 대중에게 알려진 후, 인터넷 사용자와 서비스 분야는 상상할 수 없을 정도로 양과 질적인 측면에서 급속한 성장을 이루고 있다. 이는 텍스트 위주의 초기 인터페이스에서 탈피하여, 이미지와 동영상 그리고 사운드가 어우러진 멀티미디어로 표현된 인터페이스를 제공함으로써 일반인도 쉽게 인터넷을 접근할 수 있게 되었기 때문이다. 그 결과로 컴퓨터

관련 직업에 종사하지 않는 사람도 자신의 홈페이지와 전자메일 계정을 보유하는 것이 일반화되었고, 웹 브라우저를 이용하여 쇼핑을 하거나 영화, 음악 등의 서비스를 제공하는 등, 일상 생활에서 일어나던 일들의 대부분이 사이버 세상이라고 불리는 인터넷 위에서 쉽게 접할 수 있게 되었다.

이러한 인터넷의 대중화에 힘 입어 포털 사이트, 인터넷 폰, 온라인 게임 그리고 인터넷 방송국 등 인터넷을 통한 멀티미디어 콘텐츠를 제공을 목적으로 하는 서비스가 폭발적 증가 추세에 있고, 장소의 제약에서 벗어나 이동중이나 다양한 장소에서 인터넷 서비스 수신을 가능하게 하기 위해, 무선 전화, PDA, IMT 2000 단말기 등의 멀티미디어 휴대 단말기 상에 웹 브라우저를 탑재하는 것이 일반화 되었다. 또한 가정에서 인터넷 서비스를 수신하기 위해 아파트 단위나 CATV 망이 들어오는 주택에 초고속 통신망이 가능하도록 하여, 셋톱박스나 케이블 모뎀이 설치된 컴퓨터를 통해 서비스를 받도록 하고 있다.

이렇듯 웹 브라우저는 인터넷 서비스에 있어서 사실상의

† 정 회 원 : 한국전자통신연구원 컴퓨터·소프트웨어기술연구소 선임 연구원

†† 정 회 원 : 한국전자통신연구원 컴퓨터·소프트웨어기술연구소 인터넷 정보기술연구부장

논문접수 : 2000년 7월 1일, 심사완료 : 2001년 8월 27일

기본 인터페이스로써 위치를 확고히 하고 있다. 따라서 자기만의 고유 접속 프로그램을 갖고 이루어지던 서비스들이 웹 브라우저라는 기본 인터페이스로 제공될 수 있도록 변환하는 작업이 많이 이루어지고 있는데, Java Web-Camera, 웹을 통한 전자 결제 시스템, 대형 시스템의 원격 모니터링이나 네트워크 관리 서비스 등이 그 예로 볼 수 있다. 더욱이 HAVi, Jini 그리고 UPnP 등과 같이 가전 제품에 컴퓨터의 일부 기능을 탑재하여 인터넷 서비스가 각 가전기기 수준에서 일어날 수 있도록 하는 연구가 진행되고 있는데, 기본 인터페이스로써 웹 브라우저와 웹 서버가 탑재되거나 게이트웨이를 통한 HTTP 프로토콜 지원을 기본으로 하고 있다[3-6, 15].

한국전자통신연구원에서도 웹 인터페이스를 기반으로 한 인터넷 관련 프로젝트가 진행되고 있는데, 웹 브라우저를 탑재한 휴대형 단말기 개발, 홈 네트워크와 외부 인터넷과의 연동을 위한 홈 게이트웨이 개발, 가전기기나 셋톱박스에 탑재될 수 있는 실시간 커널 연구 그리고 디지털 셋톱박스 개발을 위한 연구가 그것이다. 더욱이 이를 하나로 통합한 솔루션 개발을 위해 인터넷 정보가전 기술 개발을 위한 연구가 진행될 것이다.

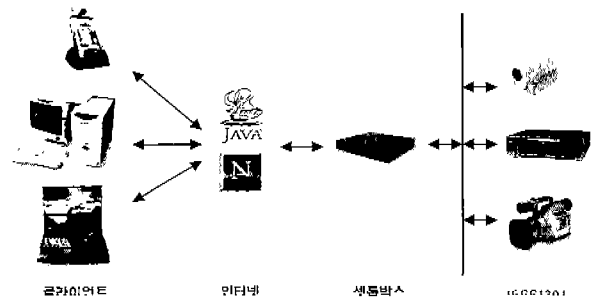
본 논문에서는 이러한 기술들의 기반 기술로 구현된 홈 네트워크 관리 시스템에 대해 기술하며 예제로 원격 카메라 관리 시스템의 동작을 보여준다. 본 논문에서 개발한 시스템은 홈 게이트웨이로 사용될 셋톱박스 위에 IEEE1394 디바이스 드라이버와 내장형 웹 서버를 구현하여 탑재하고, IEEE1394 버스로 연결된 디지털 가전기기들을 관리할 수 있도록 사용자 인터페이스와 관리 모듈을 자바 애플릿과 서블릿을 이용하여 제공한다.

본 논문의 구성을 살펴보면 우선 2장에서 홈네트워크 관리 시스템의 개발 환경에 대해 설명하고, 3장에서는 기반 기술로써 사용된 IEEE1394와 웹 서버에 대해 기술한다. 4장에서 시스템 구현 및 실행 화면에 대해 상세히 설명하며, 마지막으로 5장에서 결론 및 향후과제에 대한 얘기를 한다.

2. 개발 환경

2.1 동작 환경

홈 네트워크 관리 시스템의 전체 구성은 (그림 1)과 같다. 셋톱박스는 멀티미디어 가전기와 IEEE1394 버스에 의해 연결되고, 원격지의 사용자가 접속할 수 있도록 인터넷 망에 연동된다. 사용자는 웹 브라우저를 통해 셋톱박스로 접속하게 되고, 접속 단말기로 PC 뿐만 아니라 PDA, IMT-2000 단말기 등의 휴대형 단말기를 사용할 수 있다. 각 단말기에는 HTTP 1.1 프로토콜과 자바 수행 환경을 지원하는 웹 브라우저를 탑재하고 있어야 한다.



(그림 1) 홈 네트워크 관리 시스템 구성

사용자가 홈 네트워크 관리를 위해 접속하면 셋톱박스에 내장된 웹 서버는 홈 네트워크 구성 정보를 검색하기 위해 서블릿(servlet)을 수행한다. 수행 결과를 하나의 HTML 문서로 구성하여 사용자의 웹 브라우저로 전송하고, HTML 문서에 표현된 정보를 보고 사용자가 관리하고자 하는 가전기기를 선택하게 되면, 관리를 위한 사용자 인터페이스를 구현한 애플릿(applet)이 다운로드 되어 수행된다. 가전기기 관리 애플릿은 셋톱박스에서 동작하는 가전기기 관리 서블릿과 TCP/IP 프로토콜을 통해 정보를 교환하며 사용자의 요구를 처리한다.

2.2 개발 환경

홈 네트워크 관리 시스템 개발에 사용된 개발 플랫폼은 <표 1>과 같다.

<표 1> 개발 환경

분류	사양
하드웨어 플랫폼	메인 보드 : EBSA 285 CPU : Strong Arm SA110 메인 메모리 : 16MB 플래쉬 메모리 : 8MB IEEE1394 칩 : AIC-5800
소프트웨어 플랫폼	운영체제 : VxWorks (Tornado 2.0) 자바환경 : Personal JWorks + JSDK 2.1 클래스

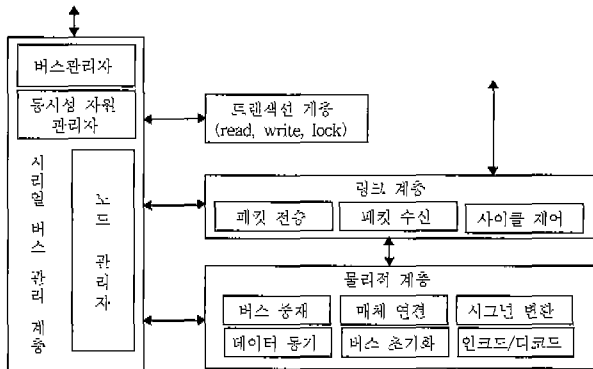
하드웨어 플랫폼은 Strong Arm사의 이벌류에이션(Evaluation) 보드인 EBSA285 보드를 사용했으며, 16MB 크기의 RAM과 8MB 크기의 플래쉬 메모리를 갖는다. 커널과 같은 핵심 모듈은 플래쉬 메모리에 펌웨어로 내장되며, 프로그램 수행에 필요한 모듈들은 타겟(target) 서버로부터 적재(load) 되어 주메모리에서 수행되게 된다. IEEE1394 카드는 Adaptech사의 AHA-8940 카드를 사용했는데, AIC-5800 칩을 내장하고 있고 200Mbps 전송속도를 지원한다. 소프트웨어 플랫폼은 WindRiver사의 VxWorks를 운영체제로 사용하였고 Tornado 2.0 개발도구로 개발하였는데, Tornado 2.0에 포함된 Personal JWorks은 퍼스널 자바 표준을 따르는 것이다. 실현상에서는 서블릿을 이용하여 서비스를 수행하므로 썬(Sun)사의 JSDK 2.1을 위한 핵심 클래스 파일인 servlet.jar를 추가로 사용하였다. 서블릿 클래스들은 순수

자바로 구현되어 있으며, 퍼스널 자바에 포함된 클래스만을 사용하여 수행하므로 퍼스널 자바 환경에서도 정상적으로 동작한다.

3 기반 기술

3.1 IEEE1394 시리얼 버스

본 논문에서 각 디지털 가전기기를 연동하기 위해 사용되는 버스는 IEEE1394 시리얼 버스이다. IEEE1394 시리얼 버스는 디지털 가전기기를 연동하여 홈 네트워크를 구성할 수 있도록 다음과 같은 특징을 갖고 있다. 우선 8V-40V 전원이 통합된 케이블을 사용하며, 100Mbps-400Mbps 전송률과 핫 플러그인/아웃(Hot Plug-In/Out) 기능을 제공하여 파워-온 상태에서 다른 기기에 영향을 주지 않고 파워-오프될 수 있다. 제어 명령을 전송하기 위한 비동기(Asynchronous) 전송 프로토콜과 멀티미디어 스트림의 실시간 전송을 위한 동시성(Isochronous) 전송 프로토콜을 독립적으로 지원하여 AV 기기를 위한 표준으로 적합하도록 설계되었다. (그림 2)는 IEEE1394-1995 표준을 따르기 위해 기기에 구현되어야 할 버스 구조이다[8].



(그림 2) IEEE 1394 시리얼 버스 구조

버스 구조는 기본 계층인 물리적 계층, 링크 계층 그리고 트랜잭션 계층의 3계층과 버스 관리를 위한 버스 관리 계층으로 구성된다.

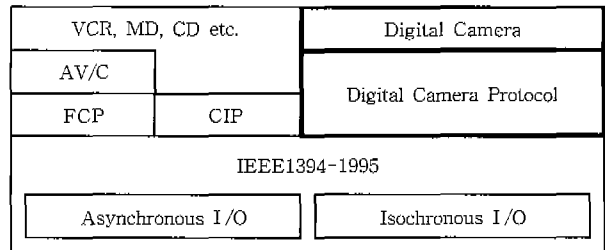
트랜잭션 계층은 ISO/IEC CSR 1212-1994 표준을 따르기 위해 구현되어야 할 read, write, lock 세가지 트랜잭션을 위한 요청(request)-응답(response) 프로토콜을 구현한 계층이다. CSR 구조는 기기의 동작이나 상태를 일련의 레지스터들로 표현하며 메모리의 한 영역을 레지스터를 위해 할당하고, 각 레지스터들의 상태를 변경하고 검색함으로써 원하는 기능을 수행할 수 있도록 설계된다. Read, write, lock은 각 레지스터 값을 변경하거나 검색할 수 있도록 제공되는 트랜잭션이다.

링크 계층은 트랜잭션 처리를 위한 비동기 및 동시성 패킷들을 송/수신하는 기능을 구현한 계층이다. 즉, 패킷 전달을 위한 주소 지정(addressing), 데이터 오류 검사, 데이터

프레이밍(framing) 기능이 포함된다. 또한 동시성 데이터 전송을 위한 동기화를 제어하기 위해 클럭 사이클을 발생하는 기능도 포함된다. 비동기 패킷은 트랜잭션 계층과 연동하여 처리하고, 동시성 패킷의 경우는 상위 응용 프로그램과 연동하여 처리한다.

물리적 계층은 전체 버스에 연결된 노드들간의 버스 소유권을 할당하기 위한 중재(arbitration)기능, 링크 계층에서 전달된 한 프레임의 데이터를 전기적 신호로 변환(signal)하여 전송하는 기능 그리고 시리얼 버스와 물리적 연결을 위한 인터페이스 정의 부분으로 구성된다. 인터페이스 부분에서 토폴로지(Topology) 변화에 따른 버스의 초기화와 전송되는 데이터의 동기화(Resynch) 그리고 데이터 신호 및 전원을 증폭하여 전달(Repeat) 하는 기능을 담당한다.

버스 관리 부분은 버스 관리자와 동시성 리소스 관리자, 노드 관리자로 구분되어지며, 노드 관리는 각 노드별로 탑재된 기본 계층의 제어 기능을 담당한다. 버스 관리자는 전체 버스 관리 기능을 처리하며 동기화를 위한 마스터 클럭 발생, 토폴로지 정보 및 노드 간 전송 속도 정보 유지 등이 포함된다. 동시성 리소스 관리자는 동시성 전송을 위해 필요한 채널이나 버스 대역폭의 할당을 담당한다.



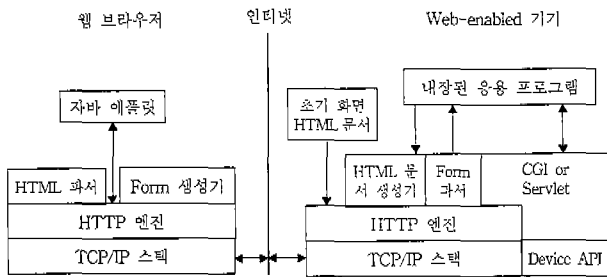
(그림 3) IEEE1394 프로토콜 스택

IEEE1394-1995 기본 프로토콜 스택 위에 각 기기의 종류별로 전용 프로토콜이 탑재되는데, 개략적인 프로토콜 스택을 기술하면 (그림 3)과 같다. IEEE1394-1995의 비동기 전송 프로토콜과 동시성 전송 프로토콜을 기반으로 하여 FCP, CIP, AV/C, Digital Camera 등이 기기의 기능에 따라 선택적으로 탑재된다. FCP는 캠코더, 미니 디스크, 오디오 등의 AV 기기를 제어하기 위한 명령어를 비동기 모드로 제공하기 위해 정의한 표준이다. CIP는 AV기자들이 IEEE1394 버스를 통해 멀티미디어 스트림을 전송하기 위한 패킷을 정의한 것으로 IEEE1394-1995 동시성 전송 표준을 기반으로 한다. AV/C는 최종 단말기의 기능상의 분류를 통해 CSR에 부여될 기능들을 정의한 것으로 VCR이나 캠코더를 위한 AV/C Video Recorder와 미니 디스크(MD)나 콤팩트 디스크 플레이어(CD)를 위한 AV/C Disk 그리고 디지털 카메라를 위한 AV/C Camera 등의 프로토콜이 있다. 본 논문에서 사용된 DS-250 모델은 AV/C와는 달리, IEEE1394-1995의 비동기 전송과 동시성 전송 계층을 그대로 사용하고, 디지털 카메라를 위해 CSR 명령어를 정의한 Digital

Camera 프로토콜을 따른다[9, 10].

3.2 내장형 웹 서버

최근에는 웹 브라우저가 네트워크 프로그램의 기본적인 사용자 인터페이스로서 자리를 잡고 있는 추세다. 특히 Post PC 시대가 열리면서 기존의 Workstation, PC 뿐만 아니라 휴대형 단말기(PDA, Cellular Phone 등)에서 웹 브라우저가 탑재되는 것을 쉽게 볼 수 있다. 더욱이 인터넷 TV, 디지털 셋톱박스, 인터넷 냉장고 및 Java Web-Camera 등 웹 서비스가 가능하도록 웹 서버를 내장하는 기기가 속속 나타나고 있다.



(그림 4) Web-enabled device 구성

(그림 4)는 클라이언트 측의 웹 브라우저를 통해 웹 서버가 내장된 기기의 서비스를 제공받는 전형적인 모델을 보여준다. 클라이언트는 PDA, PC 등의 다양한 단말기가 되며, 웹 브라우저가 탑재된다. 자바 애플릿 수행을 위한 JVM (Java Virtual Machine)을 포함한 경우는 서버측에서 제공하는 관리 프로그램을 다운로드 하여 수행할 수 있으며, 그렇지 않은 경우는 HTML 문서를 주고 받음으로써 서비스가 수행된다. 일반적으로 서버는 캠코더 혹은 인터넷 TV, 셋톱

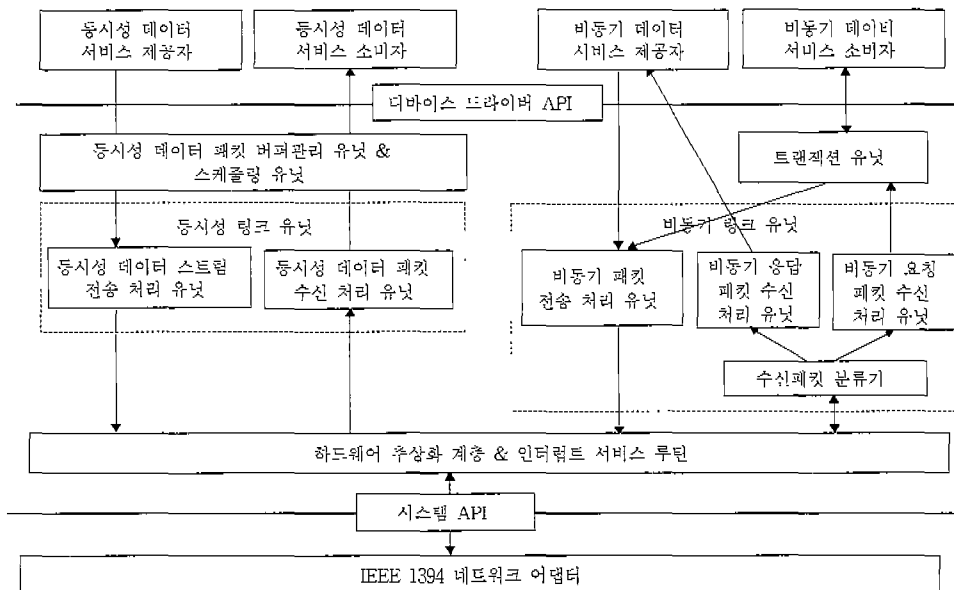
박스나 디지털 카메라 등 서비스를 제공하는 기기나 기기를 관리하는 호스트가 되며, 초기 관리 화면은 고정적인 웹 페이지로써 내장된다. 클라이언트의 접속시 초기 관리 화면을 표현하는 HTML 문서가 전송되고, 사용자의 입력에 따른 명령이 HTTP 프로토콜로써 서버 측의 응용 프로그램으로 전달된다. 응용 프로그램은 Device API를 이용하여 기기를 동작하고 그 결과를 다시 HTML 문서로 표현하여 전송하게 된다. 만약 자바 애플릿과 CGI, Servlet 등을 이용하면, 클라이언트 접속시 애플릿이 초기 관리 화면을 보여주고, 애플릿과 서블릿간에 HTTP 프로토콜 혹은 TCP/IP 등을 통한 고유 프로토콜로 서비스를 수행할 수 있다.

웹 기능을 갖는 기기를 이용한 서비스 모델에서 고려할 점은 클라이언트와 서버간의 네트워크 전송에 필요한 네트워크 트래픽을 줄이는 것과 내장된 웹 서버를 최적화하여 기기의 시스템 자원 소모를 최소화하는 것이다. 일반적으로 HTTP1.0에서의 문제점으로 지적되는 경직된 캐싱 정책과 트랜잭션 단위로 네트워크 세션을 설정하는 단점이 보완된 HTTP1.1을 구현한 웹 서버가 선호되고 있다. 또한 웹 서버의 기능을 단순화하여 일반 웹 사이트 운용을 위해 제공되는 고급 기능들을 제거하고, GET, HEAD, PUT 등의 핵심 기능과 서버측의 시스템 API를 이용하여 기기를 조작하기 위해 필요한 CGI나 서블릿 엔진 정도만을 포함하여, 전체 크기를 10KB에서 50KB 정도로 최적화 하는 것이 일반적이다[12-16].

4 구 현

4.1 IEEE1394 디바이스 드라이버

(그림 5)는 본 논문에서 구현한 IEEE1394 디바이스 드라이버



(그림 5) IEEE1394 디바이스 드라이버 구조

이더의 각 계층 중 기본 계층인 트랜잭션 계층, 링크 계층, 물리적 계층에 걸친 구조를 보여준다. 상위 계층의 응용 프로그램은 디바이스 드라이버 API를 이용하여 read, write, lock 트랜잭션을 처리하고, IEEE1394 어댑터와 디바이스 드라이버는 시스템 API를 통해 연결된다.

하드웨어 추상화 부분은 서로 다른 1394 어댑터의 기능을 추상화하여 상위 계층에서 동일한 방식의 인터페이스를 통해 다양한 벤더의 어댑터를 접근할 수 있도록 구현한 계층이며, 인터럽트 서비스 루틴은 어댑터로부터 발생된 인터럽트의 기능을 처리하는 서비스 루틴들을 구현한 부분이다. 서비스 루틴이 수행되는 동안에는 다른 모듈들이 디바이스 드라이버를 사용할 수 없으며, 다른 인터럽트 발생도 불가능하도록 모든 인터럽트가 마스크(mask) 된다. 따라서 각 서비스 루틴은 최소한의 시간 내에 작업을 끝내도록 구현되어야 한다. 특히 비동기 패킷 수신 처리 작업은 다양한 포맷을 갖는 패킷 헤더 분석, 목적지 결정(addressing) 등에 필요한 절차가 복잡하므로 작업 시간을 줄이기 위한 방법이 필요하다. 본 논문에서는 패킷 수신 인터럽트 발생시 패킷을 비동기 패킷과 등시성 패킷으로 구분하고, 상위 계층의 등시성 데이터 패킷 수신 처리 유닛과 수신 패킷 분류기로 전달하는 역할만 하도록 하여 작업 시간의 최소화를 꾀한다.

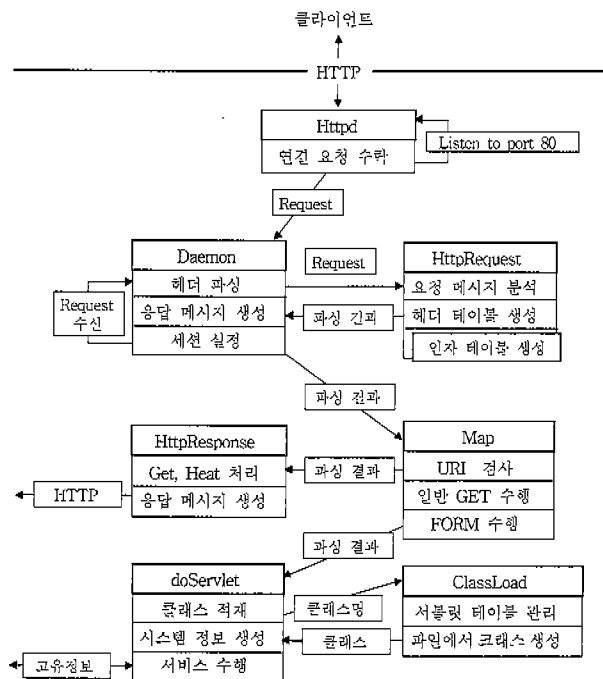
인터럽트 서비스 루틴으로부터 전달된 패킷은 링크 계층에서 분석되고 처리되는데, 비동기 수신 패킷의 경우는 패킷 분류기의 큐로 전달되어 요청 패킷의 경우와 응답 패킷의 경우로 구분되어 목적지로 넘겨진다. 목적지를 결정할 때는 요청 패킷의 경우는 헤더에 포함된 목적 주소 48비트 값을 통해 목적지를 알아내고, 응답 패킷의 경우는 내부적으로 관리되는 룩업 테이블(Lookup Table)을 참조하여 알아내는데, 룩업 테이블은 전송되는 요청 패킷의 노드 아이디(Node Id) 부분과 트랜잭션 코드(tCode) 필드를 키로 사용하여 관리된다. 즉, 다른 노드로 요청 패킷을 보낼 때 트랜잭션 코드 필드와 노드 아이디를 이용하여 송신 태스크를 테이블에 등록한 후 응답 패킷이 수신된 경우 목적지를 알아내기 위해 사용한다.

트랜잭션 유닛은 비동기 패킷 전송 요청이 일어날 때 송신 패킷 큐에 넣고 트랜잭션 정보(트랜잭션 코드 필드, 노드 아이디)를 룩업 테이블에 등록 시킨 후 사용자 태스크를 블록 시킨다. 트랜잭션 결과는 응답 패킷 형태로 수신되고 위에서 설명한대로 패킷 분석기에 의해 룩업 테이블을 통해 목적주소를 파악한 후, 블록된 사용자 태스크를 재개하여 패킷을 처리하도록 전달한다.

등시성 링크 유닛에서는 등시성 패킷의 송/수신과 관련된 작업을 처리하게 되는데, 등시성 패킷은 125s마다 송/수신되며, 일정 크기의 DMA 버퍼 단위로 전송된다. 따라서 등시성 패킷의 수신을 위해서는 버스 대역폭 할당과 네트워크

어댑터의 등시성 패킷 수신을 위한 DMA 버퍼 할당 그리고 채널 할당 작업이 선행되어야 한다. 패킷 수신은 DMA 버퍼가 채워질 때마다 발생하는 인터럽트의 서비스 루틴에 의해 처리되고, 패킷 송신은 미리 할당된 송신 DMA 버퍼에 주기적으로 송신할 패킷을 채워 넣고, 버퍼가 차면 할당받은 채널로 송신하게 된다. 본 논문에서 사용하는 AIC5800 칩의 경우 등시성 데이터를 위한 송/수신 채널이 두개씩 할당되어 있으므로 동시에 두개의 멀티미디어 스트림 처리만 가능하다.

4.2 내장형 웹 서버



(그림 6) 웹 서버의 클래스 구성

본 논문에서 구현한 웹 서버는 셋톱박스에 탑재되며, 외부에서 접속하여 IEEE1394 버스에 연결된 기기들을 접근할 수 있는 게이트웨이 역할을 한다. 일반 웹 사이트 관리를 위한 서비스와는 달리 연결된 각 기기들을 선택하고, 제어하여 서비스를 제공 받는 형태의 작업을 수행하므로, 기기 조작을 위한 사용자 인터페이스 기능과 기기 제어를 담당하는 모듈이 셋톱박스에 포함되어야 한다. 본 시스템에서는 자바의 애플릿과 서블릿을 이용하여 사용자 인터페이스와 로컬 버스에 연결된 기기를 동작시키도록 하였고, 이를 위해 썬(Sun)사의 JSDK 2.1을 지원하는 서블릿 엔진을 구현하였다. 또한 HTTP 1.1의 네트워크 세션유지 기능을 추가하였고, GET, HEAD, PUT 등의 기본 기능만을 구현함으로써 전체 웹 서버의 크기를 57KB(Servlet.jar 24KB 포함)로 최적화 하였다. (그림 6)은 본 논문에서 구현된 내장형 웹 서버의 클래스 구조를 보여준다.

클라이언트와 웹 서버의 통신은 HTTP 프로토콜에 정의된 요청 메시지(Request Message)와 응답 메시지(Response Message)의 교환을 기본으로 한다. 웹 서버는 특정 포트(일반적으로 80번 포트를 사용함)로 들어오는 클라이언트의 요청 메시지를 분석하여 적절한 오퍼레이션을 취하고 결과를 클라이언트로 전송한다[11].

본 논문의 웹 서버 수행 과정을 보면, 주 클래스인 Httpd에서 80번 포트를 모니터링 하고 있다가 클라이언트의 접속 요청이 들어오면, 새로운 소켓을 열고 요청 메시지를 처리하도록 쓰레드를 하나 생성하여 동작 시킨다. 동작되는 쓰레드 클래스는 Daemon 클래스인데 클라이언트의 요청 메시지의 버전, 헤더, 인자를 분석하기 위해 HttpRequest 클래스를 생성한다. HttpRequest는 요청 메시지를 분석하여 URI(Unified Resource Identifier) 정보와 헤더 정보를 파싱한 헤더 테이블 그리고 인자로써 전달되는 정보들을 저장한 인자 테이블을 생성한다. 분석된 결과는 다시 Daemon으로 전달되고 세션 설정 함수가 호출되는데, 이 부분에서 URI 정보를 검사하여 Form 문서 처리 요청인지, GET이나 HEAD와 같은 일반 HTTP 명령어 수행 요청인지 분석하여 해당 모듈을 호출하게 된다.

Form문서 처리가 아닌 경우는 URI 정보에 명시된 HTML 문서나 이미지와 같은 정보를 전송하는 역할을 하게 된다. 이 작업은 HttpResponse 클래스에서 처리하며, URI 정보에 지정된 정보를 포함하는 응답 메시지를 생성하고 클라이언트로 전송하는 작업을 수행한다.

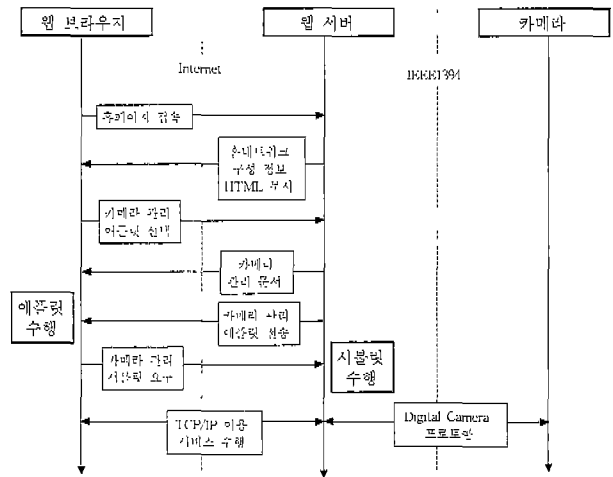
Form문서 처리의 경우는 수신되는 인자 정보를 가지고 특정 작업을 수행할 모듈이 요청 메시지에 포함되어 수신된다. 일반적으로 CGI나 서블릿등이 그런 모듈로써 사용되는데 본 논문의 웹 서버는 서블릿만 지원한다. DoServlet 클래스가 URI 정보에 명시된 서블릿을 생성하고 인자 테이블에 저장된 정보를 이용하여 작업을 수행하는데, 서블릿 클래스를 메모리로 적재하기 위해 ClassLoad 클래스를 이용한다. ClassLoad는 서블릿 클래스를 위한 등록 테이블을 내부적으로 관리하고 있으며, 등록 테이블에 클래스가 검색되는 경우 바로 객체화하여 doServlet 클래스로 전달한다. 테이블에 존재하지 않는 경우는 로컬 디스크에 저장된 파일로부터 적재하며 다음에 요구될 것에 대비하여 테이블에 등록시킨 후 doServlet으로 전달한다. 모든 서블릿은 Service() API를 갖도록 JSDK 2.1에 명시되어 있으며, 서블릿 엔진 정보와 같은 서블릿 수행 환경 정보와 클라이언트로부터 전달된 인자 정보를 이용하여 작업을 처리한다. 클라이언트와 서블릿의 정보 교환 수단은 HTTP 프로토콜일 수도 있으나 둘 간에 정의한 고유 프로토콜을 사용할 수도 있다.

HTTP1.1의 특징인 세션 유지 기능은 Daemon 클래스의 세션 설정 부분에서 지원하는데, 한번 연결된 세션은 수신되는 요청 메시지 헤더의 Connection 필드가 Close로 설정되어 있거나 클라이언트와의 네트워크 설정이 끊어지는 경

우 종료되도록 구현하였다.

4.3 홈 네트워크 관리 시스템

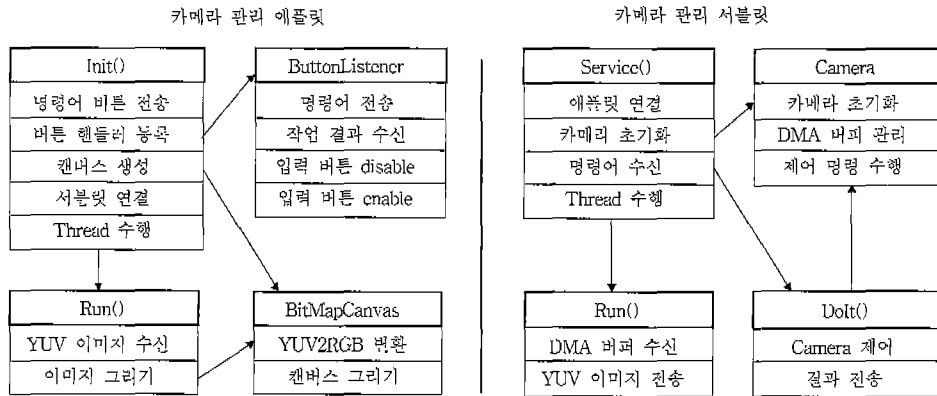
본 논문에서 구현한 홈 네트워크 관리 시스템은 웹 브라우저 통해 접속하여 각 기기를 관리하기 위해 외부 인터넷과 연결된 하나의 셋톱박스에 탑재된다. 홈 네트워크 상태를 보여주는 주 프로그램 모듈과 각 기기를 관리하는 모듈이 셋톱박스에서 동작하고 이 모듈과 정보를 교환하며, 사용자의 요구를 처리할 사용자 인터페이스를 갖는 클라이언트 프로그램이 웹 브라우저 상에서 동작되도록 제공된다. 본 논문에서는 자바의 애플릿과 서블릿을 이용하여 사용자 인터페이스와 가전기기 관리 모듈을 구현하였다. 본절에서는 디지털 카메라 관리를 중심으로 설명한다.



(그림 7) 홈 네트워크 관리 시스템 실행

홈 네트워크 관리 시스템의 실행 순서는 (그림 7)과 같다. 클라이언트의 웹 브라우저가 최초로 셋톱박스로 접속을 시도하면, 웹 서버는 주 프로그램 관리 서블릿을 동작시키고 여기서 각 기기의 시리얼 번호를 통해 IEEE1394 버스상의 디지털 카메라 그리고 다른 IEEE1394 기기들의 모델을 알아내어 전체 네트워크 구성 정보를 표현하는 HTML 문서를 생성하고 웹 브라우저로 전송한다. 클라이언트 측에서는 문서상에 표현된 정보 중 관리를 원하는 특정 기기를 상징하는 아이콘을 선택한다. 선택된 아이콘은 관리 대상인 가전기기를 위한 사용자 인터페이스를 갖는 애플릿이 포함된 HTML 문서와 연결(Link)되어 있다. 연결된 문서가 전송되고, 애플릿이 클라이언트의 웹 브라우저에서 동작한다. 애플릿은 각 가전기기 관리를 위한 초기 화면을 웹 브라우저 상에 표현하며, 웹 서버로 하여금 대응되는 서블릿을 동작시키도록 요구한다. 웹 서버의 서블릿 엔진에 의해 메모리 상에 객체화 된 서블릿은 애플릿과 TCP/IP를 이용하여 가전기기 관리를 위한 정보를 주고 받게 된다.

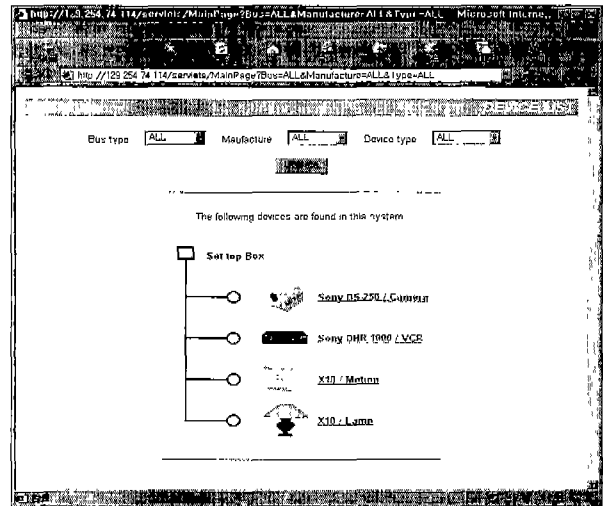
(그림 8)은 원격 카메라 관리 시스템을 위한 애플릿과 서블릿 기능을 개략적으로 표현한 것이다. 애플릿은 클라이언



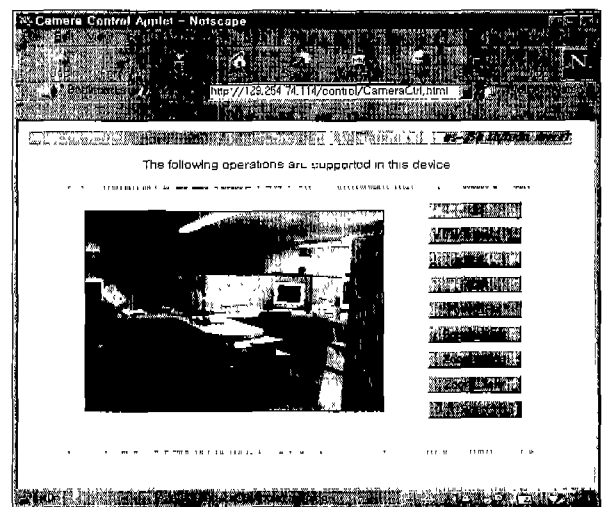
(그림 8) 원격 카메라 관리 애플릿, 서블릿 구조

트로 다운로드 된 후 카메라 관리를 위한 초기 화면을 구성하는데, 명령어 선택을 위한 입력 인터페이스로써 버튼을 생성하여 각각의 핸들러를 등록한다. 또한 이미지를 출력할 수 있도록 320X240 캔버스를 생성한다. 버튼 핸들러는 각 버튼마다 부이 받은 명령어를 문자열로 변환하여 서블릿으로 전달하고 작업 결과를 문자열로 입력 받아 사용자에게 보여주도록 설계되었다. 또한 캔버스는 수신된 YUV 데이터를 RGB로 변환하여 화면상에 출력하는 기능을 갖고 있다. 화면 초기화가 끝나면 서블릿을 연결하기 위한 작업으로 들어간다. 이미지 전달 통로로써 사용될 네트워크 세션을 설정하고 쓰레드를 수행 시킨다. 쓰레드는 서블릿으로부터 전달되는 이미지 데이터를 수신하고 화면에 출력하기 위해 캔버스를 호출하는 작업을 한다. 카메라가 파워-오프되는 경우 발생하는 이벤트를 감시하기 위해 IEEE1394 디바이스 드라이버에 모니터링 태스크를 등록한다. 이 태스크의 역할은 이벤트 발생시에 카메라가 플러그-아웃 된 경우, 사용자에게 그 사실을 알리고 관리 화면에서 명령어 입력 버튼을 사용하지 못하도록 설정한다. 물론 카메라가 플러그-인 되면 다시 명령어 버튼 입력이 가능하도록 설정한다.

카메라 관리 서블릿은 웹 서버에 의해 메모리로 적재된 후, 시스템 포트 중에서 애플릿으로 이미지 전송을 하기 위해 사용될 포트를 선택하여 포트정보를 애플릿으로 전송한다. 카메라 관리를 위한 API를 갖고 있는 Camera 클래스를 생성하고, Digital Camera 프로토콜에 따라 카메라 동작을 위한 초기화 작업을 한다. 이 작업에는 채널 할당, 대역폭 할당(데이터 전송 포맷, 프레임 설정) 그리고 DMA 버퍼 할당 등이 포함된다. 초기화 작업이 끝나고 카메라 동작을 명령하면, 카메라는 캡처된 화면을 할당된 채널로 주기적으로 보내게 되고 DMA 버퍼로 채워진다. 버퍼에 채워진 데이터는 쓰레드에 의해 애플릿으로 전송된다. doIt()은 애플릿으로부터 수신된 명령어를 상징하는 문자열을 분석, 카메라의 동작을 조작한 후 결과를 문자열로 변환하여 애플릿으로 전달한다.



(그림 9) 홈 네트워크 관리 주 화면



(그림 10) DS-250 관리 화면

(그림 9) 사용자가 웹 브라우저를 통해 셋톱박스로 처음 접속했을 때 보여지는 초기 HTML 문서이고, (그림 10)은 DS-250 아이콘을 선택했을 때 전달되는 애플릿이 수행된

모습을 보여준다. 캡춰된 화면의 오른쪽 버튼들은 카메라 제어를 위한 명령어 입력 버튼들로 줌(Zoom), 초점(Focus) 조절과 동영상 모드와 스틸 이미지(Still Image) 모드를 변환하는 명령어를 입력할 수 있다.

5 결론 및 향후 과제

본 논문에서는 인터넷 응용 프로그램을 위한 사실상의 사용자 인터페이스로써 인식되고 있는 웹 브라우저를 이용하여, 원격지에서 IEEE1394 버스로 연결된 가전기기를 관리하는 시스템을 설계하고 구현하였다. IEEE1394-1995 표준을 따르는 인터페이스 카드를 위해, 디바이스 드라이버와 Digital Camera 프로토콜 스택을 Strong Arm용 VxWorks 위에 구현했고, HTTP1.1의 네트워크 세션 유지 기능과 JSDK2.1을 지원하는 서블릿 엔진이 포함된 내장형 웹 서버를 개발하여 탑재하였다. 사용자 인터페이스와 기기 관리 모듈이 자바 애플릿과 서블릿으로 구현되어 PC와 PDA, 무선전화 등의 멀티 미디어 단말기를 통해 어느 위치에서도 쉽게 접근할 수 있다. 또한 디바이스 드라이버를 제외한 모든 계층이 순수 자바로 구현되어 다른 시스템으로의 이식이 쉽다.

본 논문의 원격 카메라 관리 시스템에서는 서블릿 수준에서 IEEE1394 버스에 연결된 카메라 및 가전기기들에 대한 검색과 기기 제어를 위한 작업이 처리되고 있는데, 최근에 많은 관심을 갖고 연구되고 있는 HAVi, Jini, UPnP 등의 홈 네트워크 표준을 살펴보면, 미들웨어 수준에서 룩업 서비스나 이벤트 처리 및 기기 관리 기능을 처리할 수 있도록 시스템을 구성하는 것이 일반적인 연구 방향이다. 따라서 현재 구현된 기반 기술을 바탕으로 미들웨어 구조를 확장하여 보다 유연하고 국제 표준 기술을 수용하는 홈 네트워크 시스템을 구축할 계획이다.

참 고 문 헌

[1] "Java Servlet API Specification version 2.1a," Sun Microsystems, Nov. 1998.
 [2] "Personal Jworks programmer's Guide 3.0 edition 1," WindRiver, 1999.
 [3] J. A. Digirolamo, "The VESA Home Network Initiative : White Paper Update Release 2," VESA Home Network Committee, 1999.
 [4] "Jini Architectural Overview : technical white paper," Sun Microsystems inc., Jan. 1999.
 [5] "Universal Plug and Play Device Architecture Reference Specification Version 0.90," Nov. 1999.
 [6] "The HAVi Specification ver. 1.0," Jan. 2000.
 [7] "IEEE 1212-1991 std. for Control and Status Register(CSR) Architecture for Microcomputer Buses," 1992.
 [8] "IEEE 1394-1995 std. for High Performance Serial Bus,"

1995.
 [9] "1394-based Digital Camera Specification ver. 1.04," Aug. 1996.
 [10] "ISO/IEC 61883-1 std. for Consumer audio/video equipment Digital interfac," 1998.
 [11] "Hypertext Transfer Protocol HTTP/1.1 rev. 06," 1999.
 [12] Peter M. Corcoran, Joe Desbonnet, "Browser-Style Interfaces to a Home Automation Network," IEEE Transactions on Consumer Electronics, Vol.434, Nov. 1997.
 [13] IAN Douglas Agranat, "Engineering Web Technologies for Embedded Applications," IEEE Internet Computing, May-June 1998.
 [14] Robert Itschner, Claude Pommerell and Martin Rutishauser, "GLASS : Remote Monitoring of Embedded Systems in Power Engineering," IEEE Internet Computing, May-June 1998.
 [15] "Design and Implementation of an Embedded-Java Web-Camera," http://www.cse.ucsc.edu/research/embedded/other/www_conf/web_cam.html.
 [16] 김인홍, 이정배, "웹 기반 내장형 응용", 한국정보처리학회지, vol.5, No.4, July, 1998.
 [17] 광병권, 이승호, 이종균, 송종수, "웹 기반 그룹웨어 솔루션", 한국정보처리학회지, Vol.6, No.3, May, 1999.



오 봉 진

e-mail : bjoh@etri.rc.kr

1993년 부산대학교 전자계산학과 졸업 (이학사)

1995년 부산대학교 대학원 전자계산학과 졸업(이학석사)

1995년~1997년 시스템공학연구소 연구원

1998년~1999년 한국전자통신연구원 컴퓨터·소프트웨어기술연구소 연구원

2000년~현재 한국전자통신연구원 컴퓨터·소프트웨어기술연구소 선임 연구원

관심분야 : IEEE1394 프로토콜, 홈 네트워크, 정보가전 등



김 채 규

e-mail : kyu@etri.rc.kr

1978년 고려대학교 수학과 졸업(이학사)

1993년 호주 Univ. of Technology Sydney 진산학(석사)

1997년 호주 Univ. of Wollongong 전산학 (박사)

1977년~1990년 KIST 진산개발센터 연구원, 시스템공학연구소 책임연구원

2000년~현재 한국전자통신연구원 컴퓨터·소프트웨어기술연구소 인터넷정보가전연구부장

관심분야 : 데이터베이스, 홈 네트워크, 정보가전 등