

# DOS에 기반한 실험적 내장형 웹 서버의 설계

양희재\*

\*경성대학교 전기전자.컴퓨터공학부

## Design of a DOS-Based Experimental Embedded Web Server

Heejae Yang\*

\*Dept of Elec. & Computer Engineering, Kyung Sung University

E-mail : hjyang@star.kyung Sung.ac.kr

### 요 약

웹 브라우저를 사용하여 각종 장치에 대한 감시 및 제어를 할 수 있게 하는 내장형 웹 시스템에 대한 관심이 최근 급격히 증가하고 있다. 이 시스템의 핵심인 내장형 웹 서버는 장치에 부착되는 특성상 하드웨어 자원 사용에 큰 제한이 있으며, 고전적인 유닉스 등의 일반 목적 대형 운영체제가 아니라 한정된 목적의 소형 운영체제상에서 운영된다. 따라서 내장형 웹 서버는 여태까지 연구 되어온 기존의 Desktop 웹 서버에 비해 매우 큰 차이를 보인다.

본 논문에서는 실험적 내장형 웹 서버의 설계 및 구현에 대해 고찰해 본다. 여기서는 특히 x86 계통 시스템에 가장 널리 사용되고 있는 DOS 운영체제 기반의 웹 서버에 대해 연구하였다. DOS 가 제공하는 운영체제 서비스 상에서 소규모 다중태스크 실행환경을 구축하였으며, KA9Q NOS 의 TCP/IP 프로토콜 스택을 이식하여 인터넷 접속환경을 실현했다. 이 기반 위에서 HTTP 서버를 설치하였으며, 별도의 응용 프로그램과 연동되어 장치에 대한 제어 및 감시가 이루어진다. 실험은 80486DX-2 단일기판 컴퓨터 상에서 이루어졌으며 제한된 하드웨어 자원하에서 매우 우수한 서비스 특성을 나타내는 것을 확인하였다.

### I. 서 론

내장형 웹 시스템 (embedded Web system) 은 프린터, 복사기, 통신장비 등에 웹 서버를 내장하여 외부에서 웹 브라우저를 사용하여 이들 장치에 대한 제어 및 감시를 할 수 있게 하는 새로운 형태의 웹 시스템이다 [3]. 현재 대부분의 웹 서버는 주로 대규모 텍스트 자료나 이미지, 동화상 등 멀티미디어 자료를 제공하는 데스크톱 (Desktop) 시스템으로서, 대표적인 서버로는 CERN HTTPD, NCSA Server, Apache Server 등을 들 수 있다. 이들은 대개 유닉스, Windows NT 등 일반목적의 대형운영체제상에서 운영되고 있으며, 수많은 사용자들에게 대량의 자료를 제공하는 목적으로 사용된다. 이에 비해 내장형 웹 시스템은 장치에 대한 제어 및 감시를 위한 소규모 제어 데이터를 제공하며, 제한된 하드웨어 자원을 갖는 한정목적의 소형운영체제상에서 운영된다. 사용자 범위 및 숫자도 비교적 제한적이며 작은 크기의 제어자료 제공을 목적으로 사용된다. 이갈

은 차이로 인해 기존의 데스크톱 시스템에서 사용되던 서버를 내장형 웹 시스템에 바로 적용하는 것은 불가능하거나 매우 비효율적이며, 따라서 새로운 시스템의 개발이 요구된다.

본 논문에서는 이런 배경에서 내장형 웹 응용을 위한 실험적 웹 서버를 설계하였다. 여기서는 특히 내장형 시스템의 프로세서로 통용되고 있는 x86 계통 시스템에 가장 널리 사용되고 있는 DOS 운영체제 기반의 웹 서버에 대해 연구하였다. DOS 가 제공하는 운영체제 서비스 상에서 소규모 다중태스크 실행환경을 구축하였으며, KA9Q NOS 의 TCP/IP 프로토콜 스택을 이식하여 인터넷 접속환경을 실현했다. 이 기반 위에서 실험적 HTTP 서버를 설치하였으며, 별도의 응용 프로그램과 연동되어 장치에 대한 제어 및 감시가 이루어진다. 실험은 80486DX-2 단일기판 컴퓨터 상에서 이루어졌으며 제한된 하드웨어 자원하에서도 매우 우수한 서비스 특성을 나타내는 것을 확인하였다.

본 논문의 구성은 다음과 같다. 2절에서는 내

장형 웹 시스템과 고전적인 데스크톱 웹 시스템과의 차이점을 분석하고 일반적인 내장형 웹 시스템의 구조에 대해 설명한다. 3절에서 본 연구에서 구현한 DOS 기반 내장형 웹 시스템의 설계에 대해 소개하며, 이것의 특성 및 성능에 대해 4절에서 알아본다. 5절에서는 결론과 함께 향후 연구 방향에 대해 소개한다.

## II. 내장형 웹 시스템

1992년 이래 웹은 인터넷의 대표적 정보도구로서 널리 사용되어왔다. 웹은 처음에는 단순한 텍스트나 이미지 자료 정도만을 다루었으나 이후 소리, 동영상, 데이터베이스, 가상현실 등도 다룰 수 있는 보편적 도구로 발전했으며, 이에 따라 모든 정보에 대한 접근이 웹 브라우저를 통해 이루어지는 통합환경으로 바뀌고 있다. 사용자들의 입장에서 보면 모든 서비스를 웹 브라우저라는 단일 인터페이스를 통해 접근할 수 있으므로 특정 서비스를 받기 위해 특정 프로그램을 새로 설치하고 배워야 할 필요가 없다는 점에서 매우 효과적이라 할 수 있다.

최근 들어서는 프린터나 복사기, 통신장치 등에 소규모 웹 서버를 내장시켜 외부에서 웹 브라우저를 통해 이들 장비에 대한 제어 및 감시를 할 수 있게 하는 내장형 웹 시스템이 큰 관심을 끌고 있다. 내장형 웹에 대한 연구는 최근 1, 2년 사이에 급격히 이루어지고 있으며, 향후 엄청난 시장을 형성할 것으로 기대되고 있다 [5].

그림 1은 일반적인 내장형 웹 시스템의 구조를 보여주고 있다. 장치 내부에 내장형 웹 서버와 응용 프로그램이 놓인다. 응용 프로그램은 장치에 대한 실질적인 제어 및 감시 기능을 담당하며, 웹 서버는 응용 프로그램과 웹 브라우저 간의 인터페이스 역할을 한다. 웹 서버와 응용 프로그램은 대개 분리된 태스크로 별도로 동작하며 일반적으로 다중쓰레드를 지원하는 실시간운영체제(RTOS) 상에서 운영된다.

내장형 웹 시스템(e-web)은 기존의 데스크톱 웹 시스템(d-web)과 몇가지 점에서 큰 차이를 보인다. 먼저 웹 서버가 운영되는 하드웨어면에서 d-web은 주로 고성능의 워크스테이션급 이상의 컴퓨터가 사용되는데 비해 e-web은 미미한 성능의 프로세서를 사용하는 간단한 단일기판 컴퓨터가 사용된다. 주기억장치의 용량도 불과 수백 KB에서 많아야 수 MB를 넘지 않으며, 하드디스크와 같은 보조기억장치는 아예 없는 경우가 많다.

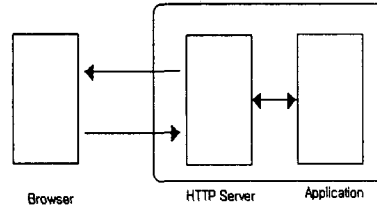


그림 1 일반적인 내장형 웹 시스템의 구조

능의 프로세서를 사용하는 간단한 단일기판 컴퓨터가 사용된다. 주기억장치의 용량도 불과 수백 KB에서 많아야 수 MB를 넘지 않으며, 하드디스크와 같은 보조기억장치는 아예 없는 경우가 많다.

운영체제면에서도 d-web의 경우 유닉스나 윈도우 NT 등 다양한 기능을 갖는 범용 운영체제가 사용되지만 e-web에서는 최소한의 모니터 프로그램 또는 마이크로 커널이 사용된다.

웹 서버와 브라우저간에 유통되는 정보 형식 및 크기면에서도 차이를 보인다. d-web의 경우 텍스트, 이미지, 동영상 등 다양한 자료들이 유통되며, 이들의 크기도 수백 byte에서 수 MB까지 매우 다양하다. 그러나 e-web의 경우 유통 자료들은 장치의 제어 및 감시 정보들이며, 이들의 크기는 일반적으로 수십 byte에서 수 KB 이내로 비교적 작다.

d-web의 경우 대개 정적인 자료들이 많아 브라우저 내의 로컬 캐시 또는 별도의 프록시 서버의 캐시를 사용하는 경우가 많다. 그러나 e-web의 경우 운영되고 있는 장치에 대한 정보는 끊임없이 변화하므로 캐시 사용이 어렵거나 불가능하다.

이상의 이유들로 인해 내장형 웹 시스템은 기존의 데스크톱 웹 시스템과는 매우 다르게 설계 및 구현되어야 한다. 아직까지 내장형 웹 시스템에 대한 연구가 많이 이루어져 있지않고 있으며, 따라서 다양한 실험을 할 수 있는 프로토타입 시스템의 개발이 조속히 요구되고 있다. 본 연구에서는 이에 따라 내장형 시스템의 제어장치로 현재 가장 널리 사용되고 있는 x86 계통 프로세서를 쓴 프로토타입 내장형 웹 서버를 개발하였다. 특히 x86 시스템의 운영체제로 널리 보급되어져 있으면서도 간단한 DOS 운영체제를 사용하여 시스

템을 구현하였다. 본 연구를 통해 얻고자 하는 주요 목적은 다음과 같다.

▲ Feasibility 조사: 내장형 웹 시스템이 갖추어야 할 자원, 즉 하드웨어 (프로세서, 메모리 등) 및 소프트웨어 (운영체제, TCP/IP, 웹 서버 등)의 수준 파악

▲ 성능 평가: 주어진 자원 하에서 시스템의 성능 평가를 통해 병목현상을 파악하고 향후 개선점을 조사

▲ Rapid prototyping: 가장 널리 사용되는 x86 계열 프로세서와 DOS 기반의 운영체제를 사용함으로써 손쉽게 testbed 를 구현하고 빠른 시간 내에 상용화 할 수 있는 기반 마련

### III. 시스템 설계

하드웨어 testbed 로는 Advantech 사의 PCA-6143P 단일기판 컴퓨터를 사용했다 [2]. 80486DX-2 프로세서를 갖는 이 컴퓨터는 32-bit 의 데이터 버스를 갖고 있으며, 입출력 장치는 16-bit 의 ISA 버스를 통해 이루어진다. 4MB 의 DRAM 이 설치되어있으며, 그외에도 1.44MB 의 Flash memory 가 내장되어져서 보조기억장치 역할을 할 수 있게 되어있다. DOS 는 이 flash memory 가 제공하는 화일 시스템 상에 놓여있다.

#### 3.1 운영체제

이 시스템에서는 크게 세가지의 프로세스, 즉 HTTP server 프로세스, Application 프로세스, TCP/IP 프로토콜 스택 등이 각각 실행된다. 이들 프로세스의 효율적 관리를 위한 운영체제의 지원이 필요하다. 기존의 데스크톱 방식의 웹이 실행되는 환경인 Unix 등의 일반적 운영체제는 자원의 제약이 큰 내장형 시스템에서는 적합하지 않으며, 보다 작고 효율적인 운영체제가 요구된다.

본 연구에서는 기본 운영체제로 DOS (MS-DOS v6.2) 를 사용했다. DOS 는 작은 메모리 요구량에도 불구하고 프로세스 관리, 주기억장치 관리, 입출력장치 관리, 화일 시스템 등 전형적인 운영체제 서비스의 대부분을 포함하고 있는

며 [4], 대부분의 low-end x86 계통 시스템에 가장 널리 보급되어있다는 점과 함께 잘 정리된 documentation 이 제공되고 있다는 장점이 있다.

다만 DOS 운영체제는 단일태스크 환경을 지원하는 매우 제한된 기능을 갖고 있으며, 내장형 웹 시스템과 같은 다중태스크 환경에는 부적합하다. 따라서 본 연구에서는 DOS 운영체제 상에 다중태스크 지원을 위한 새로운 계층을 추가했다. 이 계층은 TSR (Terminate and Stay Resident) 프로그램으로 구축되었으며, booting 시 메모리에 상주한다. 프로세스 관리부분은 비선점 다중 태스킹 환경을 바탕으로 태스크 사이의 동기화를 위한 wait(), signal() 등의 시스템 호출들을 지원하게 하였으며, 특히 TCP/IP 프로토콜 스택 지원을 위한 각종 timer 를 구축했다. 메모리 관리부분은 기존의 MS-DOS 가 실행 중인 단일 태스크에 모든 메모리를 할당하는 것을 고쳐 다수개의 태스크가 필요로 하는 메모리를 할당해 줄 수 있게 했다.

#### 3.2 TCP/IP

기존의 DOS 에 추가하여 KA9Q NOS 가 제공하는 TCP/IP 프로토콜 스택을 본 시스템에 이식하였다. KA9Q NOS [7] 는 원래 아마추어 무선 (HAM)에서 사용되는 패킷 통신을 위해 고안되었으나 코드가 비교적 잘 정리되어있고 그 크기도 작아 이 정도 규모의 시스템에 매우 적합하다고 판단했다. 전체 코드 중 DNS 등 내장형 웹 운영을 위해 꼭 필요한 것이 아닌 것은 모두 생략하고 코드를 최적화하여 메모리 요구량을 최소화했다.

물리적 인터넷 접속을 위해 본 시스템에서는 3COM 의 EtherLink III (3C509B) Ethernet card 를 사용하였으며, ISA 버스상에서 16-bit 입출력이 이루어지도록 설정하였다. TCP/IP protocol stack 과의 연결을 위한 패킷 드라이버 프로그램 (3c5x9pd.com)도 설치했다. 이와함께 ARP 기능을 넣어서 IP 주소와 Ethernet 주소간 변환이 이루어지게 하였다. 설계된 시스템은 서버로 동작하므로 자신이 ARP Request 를 보내기 보다는 대부분의 경우 클라이언트의 ARP Request 에 대해 응답을 하는 역할을 맡는다. 응답을 하면서 클라이언트의 Ethernet 주소 및 IP 주소를 자신의 ARP cache 에 저장한다.

### 3.3 HTTP 서버

내장형 시스템의 HTTP 서버는 기존 데스크톱 시스템의 서버에 비해 매우 간단한 기능을 갖는다. 본 논문에서는 HTTP version 1.0 규격에서 정한 여러가지 method 중 단지 GET method 만을 사용하게 했다. 내장형 웹과 같이 클라이언트에서 서버로 보내는 정보의 양이 비교적 적은 경우 환경변수 (HTTP\_ENV) 를 사용하는 GET method 가 더욱 효율적일 것으로 판단했다.

일반적인 HTTP 서버와 마찬가지로 3.2 절의 TCP/IP 프로토콜 스택이 제공하는 소켓 인터페이스를 사용하여 서버 프로세스를 구현했다. 클라이언트로부터의 접속 요구가 들어오면 그 클라이언트의 요구를 서비스하기 위한 별도의 프로세스를 새로 만들어 다수의 클라이언트를 동시에 지원할 수 있도록 했다. Apache 서버, NCSA 서버 등은 미리부터 몇개의 프로세스를 만들어 두어 프로세스 fork 에 따른 부담을 줄이도록 하고 있지만, 내장형 웹의 경우 하드웨어 자원 제약이 심하므로 CERN HTTP 서버와 같이 요구가 들어올 때마다 새로운 프로세스를 만들도록 했다.

일반 파일에 대한 요구가 들어오면 HTTP 서버가 직접 자료를 넘겨주고, 장치의 제어 및 감시에 대한 요구가 들어오면 응용 프로그램과의 공유 버퍼로부터 자료를 읽어 클라이언트에게 넘겨준다. 내장형 웹에서는 일반 파일보다는 응용 프로그램이 만들어내는 자료에 대한 접근이 더욱 빈번하므로 요구가 들어올 때마다 게이트웨이 프로그램을 fork 하는 CGI 방식은 매우 큰 부담을 야기시킨다. 따라서 여기서는 서버가 직접 공유 버퍼를 읽어 해당 데이터를 HTML 문서로 변환하여 클라이언트에게 보내게 했다. CGI 방식에 비해 확장성은 떨어지지만 내장형 웹은 원래부터 한정된 목적으로 사용되는 것이므로 별 문제는 없다.

### 3.4 내장 응용 프로그램

내장형 웹 시스템의 응용분야의 한 예로 본 연구에서는 선박의 기관실 상태 감시 시스템(engine-room monitoring system) [1] 에 적용했다. 선박의 기관실 내에 있는 엔진, 발전기, 보일러 등에 대한 각종 정보들이 LSU (Local Signal

Unit) 에 의해 수집되고, 이들 정보들은 다시 내장형 웹 시스템의 응용 프로그램에 의해 주기적으로 모아진다. 응용 프로그램이 수집한 자료는 시스템 내의 공통 버퍼에 기록되며, 이 내용은 HTTP 서버에 의해 HTML 문서로 변환되어 웹 브라우저에게 보내어진다.

## IV. 실험 및 성능분석

### 4.1 메모리 사용도

본 시스템은 x86 real-mode 에서 동작한다. 먼저 기본운영체제인 DOS (MS-DOS v6.2) 커널이 차지하는 영역이 63KB, 명령어 해석기인 command.com 이 6KB, 3c5x9pd.com 패킷 드라이버가 6KB, 인터럽트 백터 영역을 포함한 기타 영역이 16KB 등 도합 91KB 를 차지했다. 이 영역 위에 다중 태스크 모듈 (24KB), TCP/IP 프로토콜 스택 (57KB), HTTP 서버 (4KB), 응용 프로그램 (3KB), DOS 기능 확장 및 대치 모듈 32KB 등 도합 120KB가 소요되었다. 이외에도 초기화된 데이터 18KB 및 초기화되지 않은 데이터 20KB 가 소요되었으며, 따라서 DOS를 포함한 총 메모리 사용량은 249KB 이었다.

### 4.2 성능

그림 2에 본 내장형 웹 서버의 성능을 나타내었다. 이 그림은 요청한 파일의 크기 변화에 따른 처리율을 보여주고 있다. 내장형 웹 서버는 일반적인 데스크톱 방식의 웹 서버와는 응용분야가 다르므로 좀 더 다른 차원에서의 성능평가가 이루어져야한다고 판단되지만 이 부분에 대해서는 향후 연구에 포함하도록 할 예정이다.

## V. 결론 및 향후 연구방향

본 논문에서 우리는 DOS 에 기반한 내장형 웹 서버의 설계 및 구현에 대해 소개했다. 내장형 웹 시스템은 장치에 부속되는 특성상 기존의 데스크톱 방식의 웹 시스템과는 큰 차이를 나타낸다. 본 연구에서는 이러한 차이점을 분석하고 현재 내장

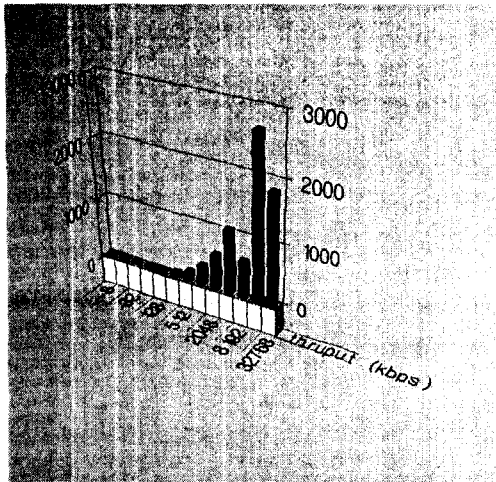


그림 2. 구현된 내장형 웹 서버의 성능

형 시스템의 제어장치로 가장 널리 쓰이고 있는 x86 계통 프로세서를 대상으로 하는 DOS 기반 내장형 웹 서버를 구축하였다. 이와같이 널리 알려져 있고 통용되고 있는 하드웨어 및 운영체제 환경을 기반으로 함으로서 고전적인 내장형 시스템이 손쉽게 웹을 통한 인터넷 접속환경으로 변화될 수 있도록 했다. 간단한 testbed 상에서 실험해 본 결과 내장형 웹 시스템으로서의 충분한 성능을 나타내는 것도 확인할 수 있었다.

내장형 웹 시스템은 [5] 에서 소개한 것과 같이 향후 데스크톱 방식의 웹 시스템을 능가하는 엄청난 시장 잠재력을 지니고 있다. 이에따라 본 연구에서 구현한 것과 같은 내장형 웹 서버 시스템의 개발이 더욱 활발해 질 것으로 보이며, 보다 작고 효율적이며 유연한 특성을 갖는 서버의 개발이 요구된다. 향후 연구방향은 이런 요구에 따라 효율적인 운영체제 커널의 개발과 TCP/IP 프로토콜 스택의 개선으로 나아갈 것이며, 아울러 플랫폼에 독립적인 시스템 구축을 위한 Java 환경으로의 변천이 요구되어진다 [6].

## VI. 참고문헌

[1] KT 전기주식회사, 중소형 선박용 엔진룸 모니터링 시스템 기술개발에 관한 연구, 통상산업부 공업기반기술개발사업 기술개발보고서, 1998. 1.

[2] Advantech Co. Ltd., PCA-6143P Industrial PC User's Manual, January 1996

[3] I. D. Agranat, "Engineering Web Technologies for Embedded Applications," *IEEE Internet Computing*, vol. 2, no. 3, May/June 1998, p.40-45

[4] D. N. Jump, *Programmer's Guide to MS-DOS*, Brady Books, 1984, ISBN 0-13-729096-9

[5] B. Lee, "Embedded Internet Systems: Poised for Takeoff," *IEEE Internet Computing*, vol. 2, no. 3, May/June 1998, p.24-29

[6] D. Mulchandani, "Java for Embedded Systems," *IEEE Internet Computing*, vol. 2, no. 3, May/June 1998, p.33-39

[7] I. Wade, *NOSintro: TCP/IP Over Packet Radio: An Introduction to the KA9Q Network Operating System*, Dowermain Pub., 1992, ISBN 1-897649-00-2

[8] N. Witchey, "An Easy-to-Do Embedded Web Server," *IEEE Internet Computing*, vol. 2, no. 3, May/June 1998, p.100 (An extended version of this article is found in the paper: "Designing an Embedded Web Server," *Internet Computing Online*, <http://computer.org/internet/v2n3/w3design.htm>)