

DGPS 간이 기준국용 임베디드 웹서버 설계

강희열*, 김정원*, 오상현*, 황동환*

*충남대학교 전자공학과, **충남대학교 전기정보통신공학부

Embedded Webserver Design for a Simple DGPS Reference Station

Huiyul Kang*, Jeong Won Kim*, Sang Heon Oh*, Dong-Hwan Hwang**,

*Dept. of Electronics Eng. Chungnam Natl. Univ., **School of Electrical and Computer Eng. Chungnam Natl. Univ.

Abstract - 임베디드 시스템을 인터넷에 접속할 수 있도록 하는 초소형의 임베디드 웹서버는 산업 현장에서 가정용에 이르기까지 널리 활용되고 있다. 본 논문에서는 인터넷을 통해 DGPS 간이 기준국의 보정 정보를 항체에 제공하기 위한 임베디드 웹서버를 제안하였다. 제안한 간이 기준국용 임베디드 웹서버는 HTTP 1.0과 1.1 표준에서 제시한 구조를 기반으로 설계 가능하다. HTTP 1.0 표준을 기반으로 한 서버는 클라이언트의 요청을 순차적으로 처리하는 반면, HTTP 1.1 표준을 기반으로 한 서버는 클라이언트에 대하여 독립된 프로세스를 할당하여 처리한다. 간이 기준국에 적합한 서버 구조를 결정하기 위하여 각 서버 프로그램의 구조 및 특징을 비교하고 네트워크 인터페이스를 가지는 GPS 수신기에 상용의 실시간 운영체제를 기반으로 구현하였다. PC 내부에 탑재한 상용의 웹 브라우저를 통하여 제안한 임베디드 웹서버에 접속하여 DGPS 간이 기준국의 보정 정보를 수신하는 실험을 수행하였다. 실험 결과 제안한 임베디드 웹서버의 구조가 DGPS 간이 기준국에 적합함을 확인하였다.

1. 서론

GPS(Global Positioning System)란 항체와 각 위성 사이의 거리 측정치와 위성의 궤도 정보를 이용하여 항체의 위치를 결정하는 위성 항법 시스템이다[1]. GPS 응용의 한가지인 DGPS(Differential GPS)는 이미 정확하게 측정된 위치에 설치된 기준 수신기(reference receiver)에서 각 위성의 측정치에 대한 오차 보정 정보를 생성하고 데이터 링크를 통하여 이를 원거리에 있는 사용자에게 제공함으로써 사용자가 GPS 항법 시 자신의 위치 정확도를 향상시키는 원리를 기본으로 한다[1]. 현재 국내에서는 해안을 따라 설치한 DGPS 기준국에서 283.5~325 kHz의 주파수 대역을 사용하는 radio beacon을 통하여 DGPS 정보를 제공하고 있으며 한국천문연구원에서는 인터넷 웹사이트를 통한 GPS 데이터 제공 서비스를 실시하고 있다. 또한 미국 및 유럽에서는 통신 위성을 이용한 광역 DGPS 서비스 제공에 관한 연구를 진행하고 있다[1]. 이중 인터넷을 이용한 DGPS 서비스는 전용의 데이터 링크를 구성할 필요 없이 기존의 인터넷 망을 전송 매체로 이용할 수 있으므로 현재 많은 연구가 진행 중이며 상용 GPS 수신기와 PC 기반의 웹서버로 구성된 DGPS 기준국의 구현에 관한 연구들도 발표되고 있다[2][3].

최근 홈 네트워킹(home networking) 및 공장 자동화 등에 널리 적용되고 있는 임베디드 웹서버(embedded webserver)는 임베디드 시스템(embedded system)을 인터넷에 접속할 수 있도록 하는 초소형의 웹서버로서 사용자로부터 하여금 웹 브라우저를 통하여 임베디드 시스템의 동작 상태를 모니터링하고 제어할 수 있도록 한다[4][5]. 이러한 임베디드 웹서버를 기준국용 GPS 수신기에 통합할 경우 데이터 제공을 위한 별도의 웹서버를 필요로 하지 않으므로 DGPS 기준국의 구조를 간단히 할 수 있는

장점이 있다. 본 논문에서는 실험실 수준에서 매우 간편하게 운용 가능한 DGPS 간이 기준국용 임베디드 웹서버를 제안하였다. 제안하는 임베디드 웹서버는 네트워크 인터페이스를 갖는 GPS 수신기 상에서 동작하는데 상용의 웹서버에 비하여 상대적으로 적은 메모리와 낮은 처리 속도를 가지는 제한된 하드웨어에서 동작하므로 DGPS 간이 기준국 서비스에 필수적인 기능만을 갖도록 설계한다. 먼저 2절에서는 DGPS 간이 기준국의 하드웨어 및 소프트웨어 구조에 대하여 설명하였으며 3절에서는 제안한 임베디드 웹서버의 구조 및 기능에 대하여 서술하였다. 4절에서는 제안한 DGPS 간이 기준국용 웹서버의 실험 결과를 제시하였으며 끝으로 결론을 맺었다.

2. DGPS 간이 기준국 구조

간이 기준국의 웹서버에 접속하는 항체들은 항법을 수행하는 동안 자신의 위치를 보정하기 위해 주기적으로 간이 기준국의 보정 정보를 필요로 한다. 정보 전송을 위하여, DGPS 간이 기준국은 유선 링크를 이용해 인터넷에 접속하고 항체는 무선 링크를 이용하여 간이 기준국 웹서버에 접속한다[6]. 일반적인 웹서버는 대용량의 파일 시스템과 모든 HTTP 메시지 헤더를 처리할 수 있는 루틴을 포함하지만, 저장 공간의 제한을 갖는 임베디드 웹서버는 필요에 따라 단순한 데이터 처리로 파일 시스템을 대신할 수 있으며 대부분 임베디드 시스템의 동작에 반드시 필요한 처리 루틴만을 포함한다[5].

2.1 DGPS 간이 기준국 하드웨어 구조

그림 1은 DGPS 간이 기준국용 하드웨어의 구조를 나타낸다. 간이 기준국 하드웨어는 GPS 신호 처리부와 네트워크 인터페이스부 및 CPU부로 구성된다. GPS 신호 처리부는 일반적인 GPS 수신기 구조를 따른다[5].

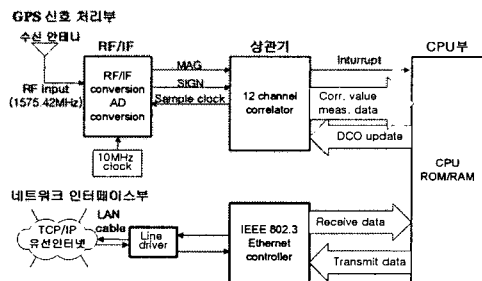


그림 1. DGPS 간이 기준국용 하드웨어 구조

네트워크 인터페이스부는 DGPS 간이 기준국을 인터넷에 접속하는 기능을 수행하는데 상용의 이더넷(Ethernet) 컨트롤러를 사용하여 설계할 수 있다. CPU부는 전체 시스템을 관장하는 기능을 수행하는데 GPS 신

호 처리부에서 처리한 데이터를 이용하여 DGPS 보정 정보를 생성하고 제한한 임베디드 웹서버 프로그램을 실행함으로써 인터넷을 통하여 간이 기준국에 접속하는 사용자에게 DGPS 서비스를 제공하는 기능을 수행한다.

2.2 DGPS 간이 기준국 소프트웨어 구조

그림 2는 DGPS 간이 기준국의 소프트웨어 구조를 나타낸다. 간이 기준국 소프트웨어는 상용의 실시간 운영체제를 기반으로 멀티태스킹(multitasking) 방식으로 설계할 수 있다. 기능상 간이 기준국 소프트웨어는 DGPS 관련 태스크와 임베디드 웹서버 관련 태스크로 나뉜다. 각 태스크는 실시간 운영체제에 의해 주기적 또는 비주기적으로 실행된다. 운영체제에서 각 태스크 실행의 기준이 되는 시간을 틱(tick)이라 한다. 일반적으로는 시스템에 내장된 타이머에서 발생하는 주기적인 인터럽트를 틱으로 사용하는데 DGPS 간이 기준국 소프트웨어의 경우 위성 신호의 획득 및 추적을 담당하는 태스크는 GPS 수신 처리부의 상관기에서 데이터를 CPU부에 출력하는 시점에 동기되어 수행되어야 한다. 따라서 DGPS 간이 기준국에서는 타이머 인터럽트가 아닌 상관기의 인터럽트 신호를 운영체제의 틱으로 사용한다. 각 태스크는 선점형(preemptive) 방식으로 동작하는데 수행 빈도가 높으며 수행 시간이 짧은 태스크에 높은 우선 순위(priority)를 두며 상대적으로 수행 빈도가 낮으며 수행 시간이 긴 태스크에 낮은 우선 순위를 둔다. 간이 기준국 소프트웨어에서는 10 Hz의 주기로 수행되는 GPS 측정치 생성 태스크를 가장 높은 우선 순위로 두며 사용자(클라이언트)의 요청이 있는 경우에만 비주기적으로 수행되는 임베디드 웹서버 태스크를 가장 낮은 우선 순위로 둔다.

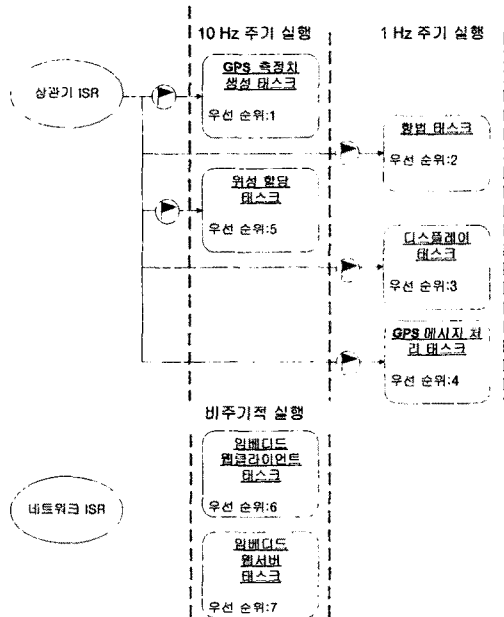


그림 2. DGPS 간이 기준국 소프트웨어 구조

3. DGPS 간이 기준국용 임베디드 웹서버

3.1 서버 프로그램의 구조

서버 프로그램의 구조는 전송 계층(transport layer) 프로토콜과 클라이언트의 요청을 처리하는 방식에 따라 그림 3과 같이 구분한다[7].

Iterative connectionless	Iterative connection-oriented
Concurrent connectionless	Concurrent connection-oriented

그림 3. 서버 프로그램의 네 가지 기본적인 유형

Connectionless 구조와 connection-oriented 구조는 전송계층의 프로토콜에 따라 구분하는데 connectionless 구조는 UDP(User Datagram Protocol)를 사용하고 connection-oriented 구조는 TCP(Transmission Control Protocol)를 사용한다. TCP를 사용하는 서버 및 클라이언트 프로그램은 데이터 패킷의 송수신 전후에 반드시 연결(connection)을 생성하고 해제하는 과정이 추가로 필요하지만 상위 계층의 메시지를 분할하고 재조립하는 기능을 제공하며, 데이터 패킷의 흐름 제어(flow control)와 패킷의 중복, 손실 및 전달 순서 변경에 대한 오류를 정정할 수 있으므로 신뢰할 수 있는 통신이 가능하다.

Iterative 구조와 concurrent 구조는 클라이언트의 요청을 처리하는 방식에 따라 구분한다. Iterative 구조는 하나의 클라이언트의 요청을 처리하는 중에 다른 클라이언트의 요청은 대기상태로 두고 하나의 클라이언트에 대한 처리를 완료한 이후에 새로운 클라이언트의 요청을 처리한다. 반면, concurrent 구조는 서버에 접속하는 각 클라이언트의 요청을 처리하기 위하여 독립적인 프로세스(또는 태스크)를 생성하며 각 클라이언트마다 생성된 프로세스들은 시분할(time-slicing) 방식으로 동작한다.

3.2 HTTP 1.0과 HTTP 1.1 서버 구조

HTTP란 웹서버와 웹클라이언트 사이에 데이터를 전송하기 위한 전송 규약으로 클라이언트가 요청하면 서버가 응답을 수행하는 요청/응답 방식의 응용 계층(application layer) 프로토콜이다. Iterative 구조를 기반으로 하는 HTTP 1.0 표준을 따르는 웹서버는 각 클라이언트의 요청 메시지마다 새로운 연결을 생성하고 해제한다. 이러한 방식은 서버 프로그램의 구조가 간단한 장점이 있으나 접속하는 클라이언트의 수가 증가할수록 실제 데이터 패킷을 송수신 하는 과정 이외에 연결을 생성하고 해제하는 과정이 추가되므로 네트워크 트래픽(traffic)을 불필요하게 증가시킨다[7][8].

HTTP 1.1 표준에서는 concurrent 구조를 기본으로 하는데 클라이언트가 서버에 접속하면 이후 접속을 중단할 때까지 연결을 계속 유지한 채로 클라이언트의 모든 요청에 대하여 응답을 수행한다. 웹서버의 구성이 복잡해지고 클라이언트에 전송할 데이터의 양이 증가함에 따라 현재 대부분의 상용 웹서버는 구조는 다소 복잡하지만 불필요한 데이터 전송을 줄일 수 있는 HTTP 1.1 표준을 기반으로 하고 있다[9].

3.3 간이 기준국용 임베디드 웹서버 구조

DGPS 간이 기준국과 같이 한 클라이언트에 주기적으로 데이터를 전송하는 웹서버의 경우 HTTP 1.0 표준에서 제시한 구조는 불필요한 연결의 생성과 해제를 반복하게 된다. 또한 이동 전화나 무선 랜을 이용한 무선 인터넷 접속은 우선 인터넷 접속에 비하여 전송 속도가 낮으며 또한 패킷 전송 오류의 발생 확률이 상대적으로 높으므로 특정 클라이언트에 대한 처리가 예기치 않게 늦어지는 경우가 발생할 수 있다. 본 논문에서는 이러한 조건을 고려하여 HTTP 1.1 표준을 따르는 임베디드 웹서버를 구성하였다. 그림 4에 제안하는 DGPS 간이 기준국용 임베디드 웹서버의 구조를 나타내었다. 마스터 서버 태스크는 HTTP 1.1 표준의 서버 프로그램 구조를

따라 각각의 클라이언트마다 동일한 우선 순위의 태스크를 생성하여 요청에 응답한다. 클라이언트에서 서버의 접속을 해제하면 해당 처리 태스크는 마스터 서버 태스크에 의하여 삭제된다. 이러한 구조를 채택함으로써 특정한 클라이언트의 요청에 대한 처리 지연으로 다른 클라이언트의 요청에 대한 응답이 지연되는 현상을 방지할 수 있다.

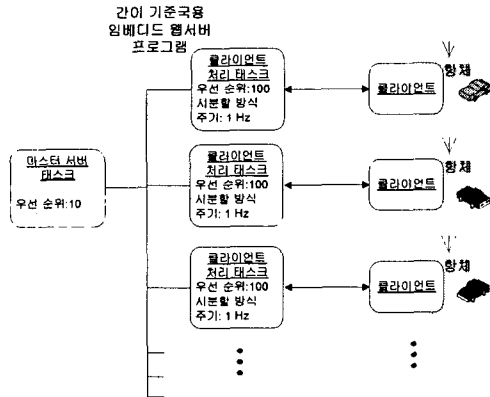


그림 4. DGPS 간이 기준국 임베디드 웹서버 구조

4. 실험 결과

제한한 임베디드 웹서버의 기능을 검증하기 위해 그림 4와 같은 DGPS 간이 기준국용 하드웨어를 설계 구현하였다. GPS 신호 처리부는 상용 칩셋을 사용하였으며 CPU부는 Intel 사의 StrongARM을 사용하였다. 네트워크 인터페이스부는 Crystal 사의 이더넷 컨트롤러 칩을 사용하였으며 실시간 운영체제는 TCP/IP 프로토콜을 지원하는 Windriver 사의 pSOS+를 사용하였다. 클라이언트의 요청 메시지 처리 루틴은 보정 정보 전송 시 반드시 필요한 메시지 헤더만 처리하도록 구현하였는데 추후 필요에 따라 메시지 헤더의 처리 루틴을 추가함으로써 웹서버의 기능을 확장할 수 있다.

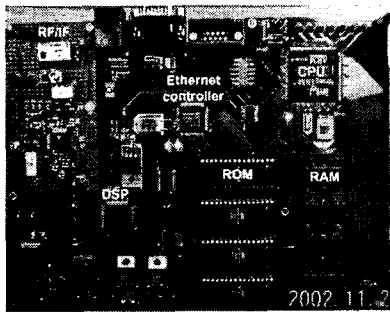


그림 5. DGPS 간이 기준국용 하드웨어

그림 6은 PC에서 웹브라우저를 통하여 간이 기준국의 임베디드 웹서버에 접속한 실험 결과를 나타낸다. 웹서버에 10개의 웹브라우저에서 1 초 간격으로 데이터를 요청하도록 하였다. 이때 클라이언트와의 송수신이 완료되면 생성된 처리 태스크는 삭제되므로 웹서버의 동작을 확인하기 위해 클라이언트 처리 태스크 생성 시 무한 루프로 동작하도록 하여 연결을 해제하지 않도록 하였다. 실험 결과 10개의 웹브라우저가 데이터를 오류 없이 수신하였으며 제한한 웹서버의 클라이언트 처리 태스크가 시분할 방식에 의해 잘 동작함을 확인하였다.

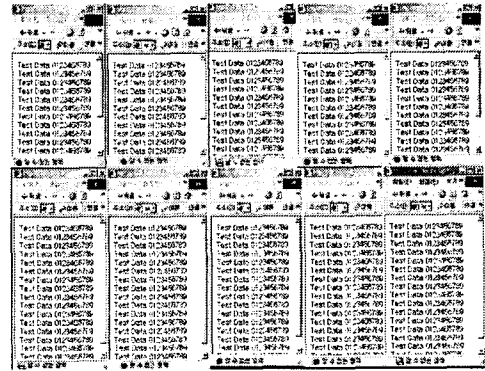


그림 6. 실험 결과

5. 결론

본 논문에서는 DGPS 간이 기준국용 임베디드 웹서버를 제안하였다. 먼저 서버 프로그램의 네가지 기본적인 유형과 HTTP 1.0과 HTTP 1.1 기반의 서버 구조의 특징을 비교하였으며 다수의 클라이언트에 요청을 독립적으로 처리할 수 있는 HTTP 1.1 기반의 웹서버 프로그램을 제안하였다. 제안한 임베디드 웹서버의 기능을 검증하기 위하여 네트워크 인터페이스를 가지는 간이 기준국용 하드웨어에 상용의 실시간 운영 체제를 이용하여 웹서버 프로그램을 구현하였다. PC 상에서 다수의 웹브라우저를 통하여 구현한 간이 기준국용 웹서버로부터 데이터를 수신하는 실험을 수행하였으며 제안한 임베디드 웹서버가 모든 클라이언트의 요청을 정상적으로 처리함을 확인하였다.

후후로 본 논문에서 제안하는 임베디드 웹서버를 이용하여 DGPS 간이 기준국을 구현하고 실제 운용 환경에서의 성능 평가가 이루어져야 할 것이다.

[참고 문헌]

- [1] W. Parkinson and J. J. Spilker Jr., *Global Positioning System: Theory and Applications Volume II*, American Institute of Aeronautics and Astronautics, Inc., Washington DC., 1996.
- [2] S. J. Ko, J. H. Won, J. S. Lee "A Practical Real-Time Precise CDGPS Positioning Using Mobile Phones, Internet and Low Cost C/A-Code GPS Receivers," *ION NTM 2001*, pp. 288-295, 22-24 January 2001, Long Beach, CA
- [3] H. Hada, H. Sunahara, "New Differential and RTK Corrections Service for Mobile Users, Based on the Internet," *ION GPS 99*, pp. 519-527, 14-17 September 1999, Nashville, TN,
- [4] Y. Sheinin, M. Emelianov, M. Ignatiev, "Embedded Real Time Web Servers", *Real-Time Magazine*, No. 1, pp. 84, 86-89, Jan. 1998
- [5] B. McCombie, "Embedded Web Servers Now and in the Future," *Real-time Magazine*, No. 1, pp. 82-83, Jan. 1998
- [6] 강희열, 오상현, 황동환, 이상정, "Internet을 이용한 DGPS 간이 기준국용 GPS 수신기 설계," 제어·자동화·시스템 공학회 대전·충청지부 학회지 pp. 83 87, 2002, 12
- [7] D. E. Comer, D. L. Stevens, *Networking with TCP/IP, Volume III, 2nd ed.*, Prentice Hall, 1996
- [8] T. Berners-Lee, R. Fielding, H. Frystyk, *Hypertext Transfer Protocol - HTTP/1.0, RFC-1945*, Internet Architecture Board, May 1996
- [9] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, *Hypertext Transfer Protocol - HTTP/1.1, RFC-2616*, Internet Architecture Board, June 1999