

내장형 다중 스레드 서버 기술

김연정*, 조동섭*

*이화여자 대학교 과학기술 대학원 컴퓨터학과
e-mail: chuwoo@ewha.ac.kr

Embedded Multi-Threaded Server Technology

Yeun_jung Kim*, Dong-Sub Cho*

*Dept of Computer Science and Engineering
Ewha Womans University

요 약

서버란 클라이언트에게 서비스를 제공하는 프로그램을 말한다. 클라이언트에게 서비스를 효과적으로 제공하기 위해서 여러 가지가 있지만 그 중 다중 스레드 서버의 장점을 지닌 임베디드 서버를 개발하려고 한다. 이를 위하여 임베디드이란 무엇이며 개발시 고려 사항과 개발 단계를 알아보고 범용 컴퓨터에서 주로 사용되는 다중 스레드 서버의 장점을 알아본 후 임베디드 시스템에 다중 스레드 서버를 구현하는데 가장 효율적인 운영체제와 다중 스레드 방식을 제안하려고 한다.

1. 서론

PC의 보급률이 거의 포화 상태에 이른 지금 임베디드 시스템을 새로운 디지털 정보 기기로 각광받기 시작하였다. 1999년까지는 Internet에 연결되는 장치의 95%가 PC였으나 지금은 그 판도가 바뀌어가고 있다. 지금의 PC와 Post PC의 비율은 95%대 5%이지만 2005년쯤에는 45%대 55%로 될 것이라는 예측에서 알 수 있듯이 앞으로 4년 이내에, 특히 정보 가전의 시장이 PC보다 더 커질 것으로 보여진다.

이렇게 임베디드 시스템이 각광을 받는 이유에는 여러 가지 이유가 있지만 그중 하나는 임베디드 시스템의 이동성이다. 임베디드 시스템은 동작에 꼭 필요한 자원만을 가지고 있으므로 자원을 낭비하지 않고 휴대가 편하다는 장점을 가지고 있다.

이 이점을 서버에 적용시켜 이동성이 좋고 서비스에 꼭 필요한 장비만을 가지고 클라이언트가 요청을 서비스 해주는 서버를 개발하려고 한다. 또한 여러 클라이언트의 요청을 효과적으로 처리하기 위해

범용 컴퓨터에서 사용되는 다중 스레드 서버 기법을 적용 시켜보려 한다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 임베디드 시스템란 무엇이면 임베디드 시스템을 설계하고 개발할 때 고려되어야 할 점과 개발 단계에 대해서 알아보고 3장에서는 범용 컴퓨터에서 서버를 다중 스레드로 구현할 때 필요한 스레드 내에서 데이터 무결성과 다중 스레드 서버 장점 및 사용된 예를 알아보겠다. 4장에서는 임베디드 시스템에서 다중 스레드 서버를 만드는데 가장 효율적인 운영체제와 다중 스레드 방식을 제안할 것이다. 마지막으로 5장에서는 결론과 향후 연구 과제를 알아보겠다

2. 임베디드 시스템

2.1 임베디드 시스템이란

임베디드 시스템이란 더 큰 시스템의 구성요소를 이루거나 사람의 개입 없이 동작하도록 기대되는 하드웨어와 소프트웨어이다. 즉 특정한 기기에 주어진 작업을 수행하도록 구성시키는 시스템이라 할 수 있다. 임베디드 시스템은 운영체제를 포함하거나 단일

본 연구는 2003년도 두뇌한국 21산업에 의하여 지원되었음.

프로그램으로 작성할 수 있을 만큼 단순하게 구현되기도 한다. 범용 컴퓨터가 어떤 목적으로 사용할지 모르기에 다양한 주변장치를 갖추고 있는 반면 임베디드 시스템에는 동작에 반드시 필요한 주변장치만을 가지고 있다[2][3].

2.2 임베디드 시스템 설계 및 개발 시 고려 사항

임베디드 시스템을 설계하거나 개발할 때 고려사항으로는 임베디드 시스템은 각종 전기적, 또는 기계적으로 동작하는 기기들을 제어하기 때문에 최악의 상황에서도 정해진 시간 내에 동작하도록 개발되어야 하며 작은 용량의 기기에 맞도록 설계되어야 하기 때문에 각 기기의 특성에 맞도록 가볍고 효율적으로 설계되어야 한다. 또한 임베디드 운영체제는 운용할 기기에 맞게 작고, 안정성 있고, 신뢰할 수 있게 설계되어야 한다. 많은 위험을 내포하고 있는 불안정한 환경에서도 오류 없이 안정적으로 동작하도록 설계되어야 하며 저가의 비용으로 높은 효율을 발휘하는 시스템을 개발할 수 있도록 설계하는 것이 매우 중요하다.

2.3 임베디드 시스템의 개발단계

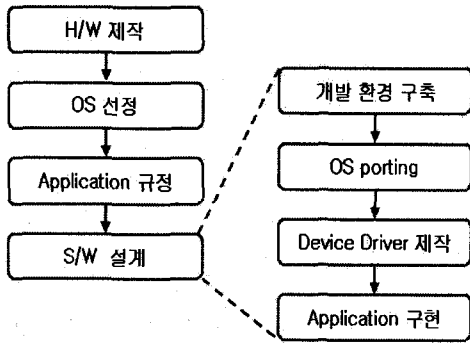


그림 1 임베디드 시스템 개발 단계

임베디드 시스템이 다양한 만큼 다양한 운영체제가 존재한다. 여기서는 임베디드 시스템의 운영체제를 리눅스를 사용하였을 때의 개발과정을 알아보도록 하겠다. 그림1에서 보는 것과 같이 먼저 용도에 적합한 하드웨어를 제작한다. 하드웨어 설계가 끝나면 먼저 적합한 운영체제를 선정하고 목적하는 애플리케이션을 규정한다. 그리고 소프트웨어를 설계한다. 소프트웨어 작업의 시작은 교차 개발 환경을 구축하는 것이다. 임베디드 시스템은 다른 시스템과는 달리 큰 용량의 저장장치를 가지고 있지 못하므로

개발하기 위해서는 Host System 이란 개발환경이 필요하다. 교차 개발 환경이란 호스트 시스템에 타겟 CPU에 맞는 툴체인(toolchain)을 구축하고 memory fusing 방식을 결정하며 부트로더를 제작하는 것이다. 교차 개발환경에서 아키텍처관련 부분을 코딩하게된다. 이 모든 작업이 끝난 후 원하는 애플리케이션을 제작하게 된다[6].

3. 다중 스레드 서버

다중 스레드 서버는 클라이언트의 요청을 다중 스레드 방식으로 서비스하는 서버이다.

3.1 다중 스레드에서의 무결성 보장

프로그램이 자신의 프로세스의 레지스터 context를 자기 자신의 주소 공간 내의 데이터 구조로 복사하고 나서, 실행을 계속하기 위해 이미 저장되어 있던 다른 레지스터 context를 로드하는 같은 프로세스가 계속해서 실행되는 것이다. 이와 같이 실행중인 프로세스 내의 연속적인 제어 흐름을 "스레드(thread)"라고 한다.

여러 명의 클라이언트의 요청을 들어주기 위해 다중 스레드를 사용하기 위해서는 스레드에서의 데이터 무결성 보장이 되어야 한다. 데이터 무결성 보장을 위한 상호 배제를 위한 4가지 요구조건은 첫째로 두개 이상의 프로세스들이 동시에 임계영역에 있어서 안 되며 임계구역 바깥에 있는 프로세스가 다른 프로세스의 임계구역 진입을 막아서는 안된다. 그리고 어떤 프로 세서도 임계구역으로 들어가는 것이 무한정 연기 되어서는 안 된다. 프로세스들의 상대적인 속도에 해해서는 얼마나 가정도 하지 않는다. 멀티프로세싱 환경에서 동시에 두개 이상의 프로세서/스레드가 메모리 번지를 접근하여 연산을 할 경우 데이터의 무결성이 보장되지 않을 것을 방지하기 위한 상호 배제의 기본적인 요구 조건들로 이것들을 고려하여 프로그램을 설계해야한다[6].

3.2 다중 스레드 서버

서버가 클라이언트의 요청을 다중 스레드 방식으로 서비스할 때에는 다음과 같은 장점을 지닌다. 첫째 다중 스레드 서버는 노드마다 사용할 수 있는 최대 사용자를 증가시켜 서버 사용이 가능한 사용자의 수를 증가시킬 수 있다.

각 요구를 서비스하기 위해 새로운 스레드를 생성하는 것의 오버헤드는 새로운 프로세스를 생성하는 것의 오버헤드보다 상당히 적다. 특히 리눅스의

경우 다른 운영체제와 비교해서 새로운 프로세스를 효과적으로 생성한다. 또한 스레드는 자동적으로 그들이 속한 프로세서의 자원과 메모리를 공유하므로 메모리 사용과 시스템 오버헤드를 줄일 수 있다.

요청에 대한 서비스를 각각의 독립적으로 하는 프로세서를 만들어 클라이언트가 서비스 받기 위해 기다리는 시간을 줄이는 동시에 idle한 서버 프로세스의 수를 줄일 수 있다. 또한 한 프로세서에만 로드가 집중되는 것을 어느 정도 막을 수 있어 로드 밸런싱(Load Balancing)을 이룰 수 있다.

스레드 풀링은 스레드가 만들어질 때 작업이 대기열에 추가되고 자동으로 시작되는 다중 스레딩의 한 형태로 스레드 풀링은 각 스레드의 속성을 개별적으로 설정하지 않은 채로 별도의 여러 작업을 시작하려는 경우에 유용하다. 각 스레드는 기본 스택 크기와 우선 순위로 시작된다. 기본적으로 시스템 프로세서 당 보통 최대 25개의 스레드 풀 스레드가 실행될 수 있습니다. 이 한도를 넘는 추가 스레드는 대기열에 보관될 수 있지만, 다른 스레드가 끝날 때까지는 시작되지 않는 것이다. 스레드 풀링의 한 가지 장점은 상태 개체의 인수를 작업 프로세서에 전달할 수 있다는 점입니다. 이러한 스레드 풀링 이용하여 접속 풀링을 지원하기 위해 설치되어 서버의 접속을 원활하게 해준다.

마지막으로 기존의 애플리케이션을 다중 스레드 서버에서 사용하는데 특별한 수정할 필요로 하지 않는다.

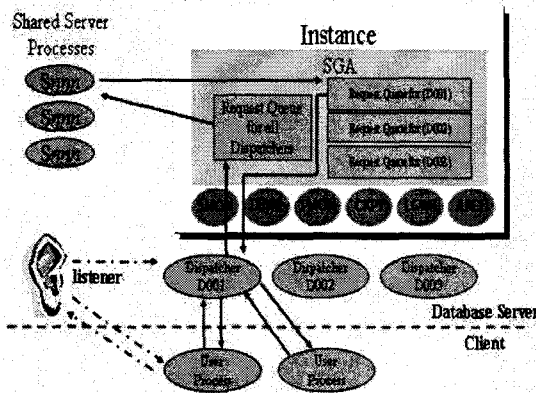


그림 2 다중 스레드 서버[7]

그림2는 실제로 데이터베이스 서버로 구현 다중 스레드 서버의 모습입니다[7].

4. 임베디드 다중 스레드 서버

앞장에서 살펴본 범용 컴퓨터에서 구현되는 다중 스레드 서버를 임베디드 시스템에서 구현하기 위해서는 시스템 특성상 범용 컴퓨터하고는 달리 최소한의 사양으로 시스템을 가동시키고 있다. 메모리 또한 범용 컴퓨터보다 작기 때문에 꼭 필요한 요소만을 가지고 있다. 그래서 임베디드 운영체제는 데스크탑 컴퓨터에서 볼 수 있는 대부분의 특징을 지원하지 않으므로 크기가 매우 작다. 보통 임베디드 시스템은 한명의 사용자만을 지원하고 임베디드 시스템을 구성하는 모든 태스크는 함께 동작하므로 멀티 유저 오퍼레이팅 시스템에서와 같은 보안 문제를 적용할 필요가 없었다.

그러나 이 시스템을 서버로 사용하기 위해서는 여러 명의 사용자를 지원할 수 있는 기능을 갖춘 애플리케이션이 되어야 한다.

이것을 위해 다중 스레드 서버를 임베디드 시스템에 적용시키는 것을 제안한다.

커널 스레드와 사용자 스레드 중 사용자 스레드를 사용할 것이다. 사용자 스레드는 커널 위에서 지원되면 사용자 수준의 스레드 라이브러리에 의해 구현되는 것으로 커널의 지원 없이 스레드의 생성과 스케줄링, 관리를 한다. 이렇게 하여 스레드 관리와 생성을 빠르게 할 수 있다.

타겟 보드의 운영 체제로는 임베디드 리눅스 선정하겠다. 스레드 기반의 운영체제는 응용 프로그램의 오류가 시스템 전체를 손상시킬 수 있고 새로운 기능 추가 어려움이 있는 등 몇 가지 제약점을 가지고 있지만 프로세스 기반의 임베디드 리눅스를 사용하면 응용 프로그램의 오류로 인한 스레드 기반 운영체제에서와 같은 심각한 손상 없이 복잡하고 다양한 응용 서비스에 사용하는데 사용할 수 있기 때문이다.

임베디드 리눅스 운영체제에 사용자 스레드로 다중 스레드를 구현하는 것이 임베디드 다중 스레드 서버에 가장 효과적일 것이라고 제안한다.

5. 결론

지금까지 임베디드 시스템이란 무엇이며 임베디드 시스템을 개발하는 단계와 고려할 점들에 대해서 알아보았고 범용 컴퓨터에서 구현되는 다중 스레드 서버가 갖는 장점과 다중 스레드로 서비스를 할 경우

스레드 간에 데이터 무결성을 위해 꼭 필요한 점들에 대해 알아보았다. 마지막으로 주로 범용 컴퓨터에서 사용되는 다중 스레드 서버를 임베디드 시스템에서 적용시킬 때 효율적인 운영체제와 다중 스레드 방식으로 임베디드 리눅스 운영체제에서의 사용자 스레드를 제안하였다.

앞으로 제안한 임베디드 다중 스레드 서버를 구현하여 타겟 보드에 실제로 실어 다중 스레드 서버가 임베디드 시스템에서도 잘 돌아가는 지를 알아볼 것이다. 범용 컴퓨터의 다중 스레드 서버와의 클라이언트의 요청을 처리하는 성능과 자원 활용도면을 비교 분석할 예정이다.

참고문헌

- [1] Craig Hollabaugh Ph.D. " Embedded Linux" Addison welsey
- [2] 박재호 "임베디드 리눅스" 한빛 미디어
- [3] Michael Barr "Programming Embedded Systems" O'REILLY
- [4] 다니엘 보에이, 마르코 체사티 "Understanding the Linux Kernel" O'REILLY
- [5] 박철 "Strong ARM(SA1110)을 이용한 board level porting 작업"
- [6] 신일경 "온라인3D슈팅게임 만들기(3) 리눅스를 이용한 게임서버 만들기" 2002년 동계 게임학회
- [7] 김용주 "다중 스레드의 사용법과 구성"