
원격 모니터링 시스템을 위한 Linux 실장 지능형 제어기 구현

송근영* · 박세현*

*국립안동대학교

Implementation of Embedded Linux Intelligent Controller for Remote Monitoring System

Keun-young Song* · Se-hyun Park*

*Andong National University

E-mail : j5id@pyunji.andong.ac.kr

요 약

원격 모니터링 시스템 위한 실장형 리눅스 지능형 제어기를 구현한다. 실장형 리눅스 지능형 제어기의 하드 코어는 32비트 cpu로서 구성되었고 실시간 모니터링과 FFT을 수행 할 수 있도록 설계되었다. 그리고 모니터링 시스템은 자바에 의한 인터넷과 GUI 환경에서 수행되도록 설계되었다. 상세 설계와 기능적 해석을 시스템 기반에서 수행되었다.

ABSTRACT

In this paper, we implement embedded Linux intelligent controller for remote monitoring system. Embedded controller as the hard core is consisted of 32 bit CPU and is designed to have processing of real time monitoring and FFT. The prototype monitoring system can operate with world wide web in GUI environment by java. Detailed design and functional analysis for monitoring system are performed by systems approach.

키워드

임베디드 리눅스(embedded linux), 원격 모니터링, 자바(java), 자바애플릿(java applet), FFT

1. 서 론

현재 산업 전반에 걸쳐 다양한 분야에서 실장형 시스템이 사용되고 있다. 특히 원격 감시 분야에 있어서는 실시간 지능형 시스템의 역할이 중요시되고 있다.[1] 현저하게 드러나는 영역에서의 감시와 미시영역의 변화 등을 장기간 감시함으로써 감시대상의 이상 징후를 파악하는 것이 중요하다. 그리고 오랜 감시기간동안의 발생할 수 있는 오류에 대한 보정 기능도 필요하겠다.

본 논문에서는 원격 모니터링 시스템을 위한 Linux 실장 지능형 제어기 구현에 대하여 기술한다. 본 논문이 제안하는 Linux 실장 지능형 제어기는 주위 환경을 실시간으로 감시하고 이를 인터넷을 통해 확인을 할 수 있도록 네트워크 환경이 구축된 장치이다. 제안된 실장형 리눅스 지능형 제어기의 하드코어는 32bit cpu로서 구성되었고 실시간 모니터링과 FFT을 수행 할 수 있도록

설계되었다. 그리고 모니터링 시스템은 자바에 의한 인터넷과 GUI 환경에서 수행되도록 설계되었다.

II. 리눅스 실장 지능형 제어기의 설계

리눅스 실장 지능형 제어기의 하드코어는 32bit 마이크로 프로세서가 사용되었고 운영체제로는 Linux가 사용되었으며 리눅스 운영체제에 의해 입출력장치와 네트워크 장치가 구동된다.

그림 1은 제안된 리눅스 실장 지능형 제어기의 전체적인 구성도이다.

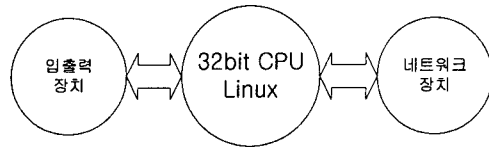


그림 1. 제안된 시스템 구성도

입출력 장치를 통해 각종 정보를 수집하여 프로세서에서 이를 분석, 처리하여 네트워크 장치를 통해서 전송할 수 있다.

그림2는 입력장치의 구성도이다. 주위 환경의 감시를 위해 다양한 종류의 센서를 사용하여 취득된 아날로그 신호를 증폭 및 필터링을 한 후 이를 디지털 신호로 변환하여 cpu에 전달한다. 이때 주변 장치 제어와 cpu로의 데이터 전달을 위해 FPGA가 사용되었으며 이 FPGA는 프로그래밍이 가능한 논리소자이므로 외부장치 변화에 따라 변경이 가능하다.

센서수신부는 잡음제거를 위한 필터링 회로와 소신호를 증폭하기 위한 증폭기 회로, 이렇게 증폭된 신호를 디지털로 변환하는 A-D변환 회로로 구성되어 있다.

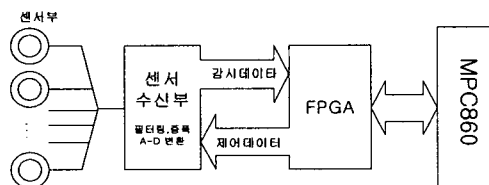


그림 2. 입력장치의 구성도

이렇게 취득된 정보는 리눅스 기반에 수행되는 프로그램에 의해 적절한 데이터로 가공되어 DB화되어 저장되고, 또한 네트워크를 통해 전송한다.

리눅스 실장 지능형 제어기는 수집된 각종 정보를 효과적으로 원격지에 전송하기 위해 네트워크 장치부를 가진다. 기존에 사용된 느린 전송속도를 가진 RS232 대신에 이더넷을 사용함으로써 10/100Mbps의 빠른 전송속도를 지원, 범용네트워크를 통해 실시간으로 정보를 원격지에 전송할 수 있다. 또한 리눅스 운영체제에 의해 다양한 전송 프로토콜을 통해 정보를 전송할 수 있으며 시스템에 각종 서버를 구축하여 원격지에서 시스템으로의 효율적 접근이 가능하다.

리눅스 실장 지능형 제어기는 운영체제에 의해서 동시에 다양한 용도로 사용될 수 있다. 실제로 시스템은 센서로부터 데이터를 취득하는 동시에 웹서버로서 동작하도록 하였다. 그림3은 웹 브라우저를 통해 리눅스 실장 지능형 제어기에 구축된 웹서버에 접근한 모습이다.

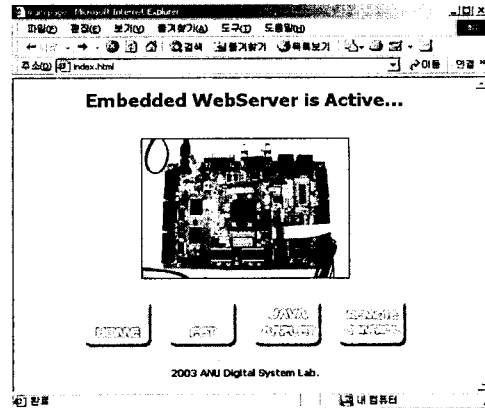


그림 3. 시스템에 구축된 웹서버

III. 원격 모니터링 시스템 설계

본 논문에서는 리눅스 실장 지능형 제어기에 웹 서버를 구축하여 원격지에서 제어기로의 접근 가능 하도록 설계하였다. 웹 서버를 통해서 제어가 취득한 각종 정보를 브라우저를 통해 확인할 수 있다. 그러나 html문서를 통해 얻는 정보는 매 순간 변화하는 정보를 확인하기에는 한계가 있다. 실시간 원격 감시를 위해서는 다른 방식의 기술이 요구되는데 이를 가능케 하는 것이 자바 애플릿이다.[2]

본 논문에서는 원격 모니터링 시스템을 인터넷과 Java를 이용한 GUI 환경으로 구축하였다. 실시간 원격 감시를 위해 자바 애플릿이 사용되었다.

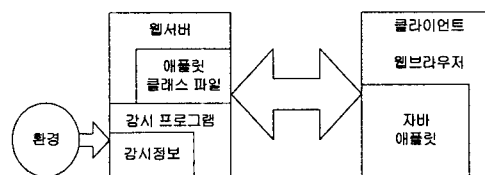


그림 4. 제안된 원격 모니터링 시스템의 구성도

자바 애플릿은 웹 브라우저 상에서 실행 가능한 작은 프로그램이다. 이러한 자바 애플릿의 프로그램 본체는 웹 서버로서의 리눅스 실장 지능형 제어기에 저장되어 있으나, 웹 브라우저로 접근하여 이를 웹 브라우저 상에서 실행할 수 있다.

그림4와 그림5는 각각 원격 모니터링 시스템의 구성도와 감시정보 전달과정이다. 리눅스 실장 지능형 제어기는 웹 서버로 수행되는 동시에 별도 실행되는 감시 프로그램으로 주위의 환경을 감시하여 정보를 수집하고, 웹 브라우저는 웹 서버에서 애플릿 클래스 파일을 받아서 애플릿을 실행

시킨다. 실행된 자바 애플릿은 리눅스 실장 지능형 제어기에 수행되는 감시 프로그램으로부터 실시간 감시정보를 취득하여 화면에 적절한 형태로 표시한다.

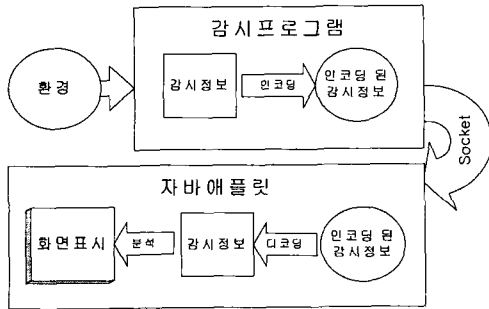


그림 5. 제안된 원격 모니터링 시스템의 감시정보 전달과정

웹 브라우저에서 웹 서버의 감시 페이지로 접근하여 자바 애플릿이 실행되면 애플릿은 리눅스 실장 지능형 제어기의 감시프로그램이 열어둔 소켓에 접속한다. 감시프로그램과 애플릿 간의 소켓이 연결되면 애플릿은 감시 프로그램이 실시간으로 수집하는 감시정보를 별도로 코딩된 형태로 실시간으로 수신할 수 있다. 이렇게 수신된 정보를 디코딩하여 분석하여 그래프, 데이터 리스트 등의 다양한 표현방법으로 감시정보를 분석할 수 있다.

IV. 실험 및 결과

본 논문에서는 원격 모니터링을 위한 리눅스 실장 지능형 제어기를 구현하고 이를 실험하였다. 그림 6은 실험에 사용된 HW이다. 사용된 HW의 중앙처리장치는 32bit 프로세서인 mpc860이 사용되었으며 운영체제는 리눅스를 사용하였다.

설계된 HW에는 실장형 리눅스 운영체제가 실장 되어 있으며 운영체제는 다양한 네트워크 기능을 제공한다. 그리고 일반적인 리눅스와 마찬가지로 실장형 리눅스 역시 파일 시스템을 가지며 또한 각종 서버로서 동작이 가능하다. 본 실험에서는 리눅스 운영체제 하에 웹서버를 구동하였으며 동시에 별도의 감시 프로그램을 실행하였다. 실장된 리눅스 운영체제는 멀티 태스킹을 웹서버와 감시프로그램 등 여러 프로그램을 동시에 운용 가능함을 알았다.

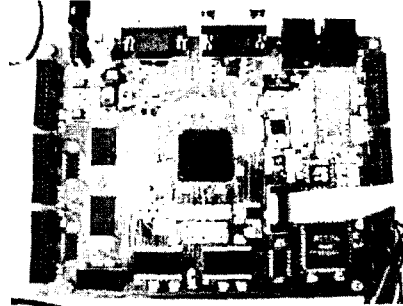
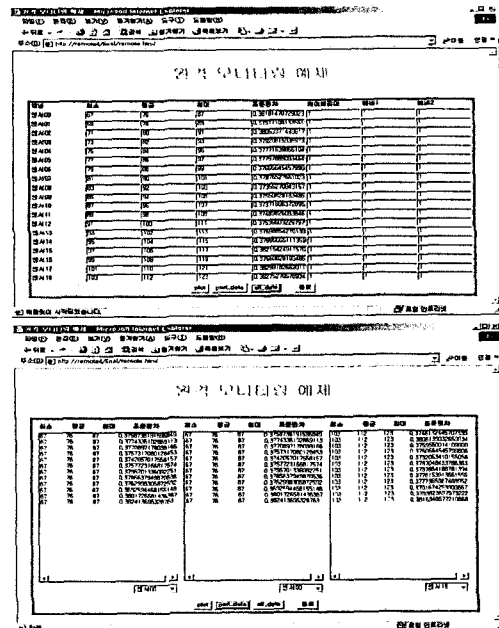


그림 6. 리눅스 실장 지능형 제어기의 모습

본 실험에서는 여러 가지 감시 정보는 각종 센서를 통해 그 값을 취득된 값이 최종 사용자의 브라우저 상에 제대로 나타나는지를 살펴보았다. 취득된 정보는 FPGA에 의해 제어 하에 mpc860 프로세서에 전달되고 프로세서는 적절한 형태 주기억장치와 보조기억장치에 보관한다. 보관된 정보는 전 처리 과정을 거쳐 즉시 송신 가능한 형태의 데이터로 코딩되어 네트워크를 통해 수신 측에 전달된다.

그림 8은 웹 브라우저 상에 실행된 원격 모니터링 프로그램이 실행된 상태이다. 실행된 프로그램은 HW에서 취득한 여러 센서 정보를 그래프 형태, 테이블 형태 등 다양한 표시 방식으로 화면에 표시된다. 그리고 임의의 센서 정보만을 FFT 적용하여 주파수 영역에서의 분석도 가능하다.



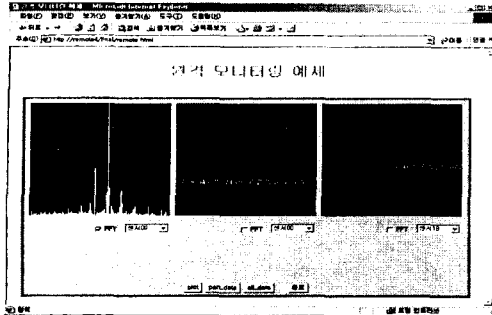


그림 8. 실행된 원격 모니터링 프로그램

센서의 종류에 따라 별도의 설정값을 주어서 데이터 오차의 보정도 가능하다. 이때 설정값은 소켓 통신을 통해 서버 측에 전달되어 변환된 값에 대한 처리를 하며 개별 설정값 이외에도 각종 제어정보를 전송함으로써 서버를 통해 여타 장치를 제어할 수 있다.

여기서 원격 모니터링 프로그램은 자바 애플릿으로 설계되어 있어 클라이언트 PC에 별도의 원격 모니터링의 위한 프로그램을 설치할 필요가 없으며 웹 브라우저만 설치되어 있으면 된다. 원격 모니터링 프로그램의 본체는 웹 서버 상에 존재하며 웹 브라우저를 통해 원격 모니터링 페이지에 접속하면 자동으로 이를 호출하여 실행된다.

표 1 기존의 방법과 제안된 방법 비교

	기존의 방법	제안된 방법
용도	제한된 기능	다양한 용도
감시형태	정적, 단기간	실시간, 장기간
보수,확장	어려움	용이함
네트워크	RS232, 저속	이더넷, 고속
원격지 거리	근거리 장거리시 비용추가	장거리 인터넷을 사용
원격제어	불가능	가능
원격 감시 SW	설치	별도로 설치
	호환성	특정 플랫폼만 지원
	요구 사항	운영체제별로 개발 필요
운영체제	운영체제없음 or 고가의 운영체제	오픈정책을 따르는 리눅스 운영체제
개발비용	고가	저렴한 비용

표 1은 기존의 방법과 리눅스 실장 지능형 방법을 비교한 자료이다. 리눅스 실장 지능형 제어를 이용한 방법은 여러 면에서 기존의 방법보다 장점을 가진다. 용도 면에서 특정한 용도에 구애받지 않고 여러 용도로 사용될 수 있으며, 실시간 감시를 제공하며, 장기간 동안 감시 기능을 수행할 수 있다.

고속의 이더넷 네트워크를 사용함으로써 장소에 제약을 받지 않고 원거리에서도 모니터링이 가능하며 원격제어도 가능하다.

원격모니터링 소프트웨어 역시 다양한 플랫폼을 지원하며 장비에 구애받지 않는다. 비용에 있어서도 고가의 운영체제의 구입 없이 오픈 라이선스정책을 따르는 리눅스를 사용함으로써 저렴한 비용으로 구현이 가능하다.[4]

원격모니터링 소프트웨어 역시 다양한 플랫폼을 지원하며 장비에 구애받지 않는다. 비용에 있어서도 고가의 운영체제의 구입 없이 오픈 라이선스정책을 따르는 리눅스를 사용함으로써 저렴한 비용으로 구현이 가능하다.[4]

V. 결 론

본 논문에서는 원격 모니터링 시스템 위한 실장형 리눅스 지능형 제어를 구현하였다. 실장형 리눅스 지능형 제어기의 하드 코어는 32비트 cpu로서 구성되었고 실시간 모니터링과 FFT를 수행할 수 있도록 설계되었다. 그리고 모니터링 시스템은 자바에 의한 인터넷과 GUI 환경에서 수행되도록 설계되었다. 상세 설계와 기능적 해석을 시스템 기반에서 수행되었다

원격 모니터링 시스템으로서 리눅스 실장 지능형 제어를 구현하고 이를 실험을 통해 검증해 보았다.

제안된 원격 모니터링 시스템은 인터넷이 보편화된 현재의 시점에서 산업 현장 뿐 아니라 가정자동화 등에도 사용 가능하며 특정 플랫폼에 국한되어 개발되어온 기존의 원격모니터링 소프트웨어와 달리 다양한 플랫폼에서도 별도의 소스 코드 변경 없이 그대로 적용할 수 있는 Java language를 사용함으로써 개발에 있어서도 융통성을 제공하였다.

참고문헌

- [1] John Lombardo, Embedded Linux, infobook, p.11, 2002.
- [2] Se-hoon Lee & Chang-jong Wang, Inside JAVA2 programming, daerim, p19, 2000.
- [3] Richard Stones & Neil Matthew, Beginning Linux Programing 2nd Edition, wrox, p671~712, 2000.
- [4] Sang-hoon Kim, Flash Memory File-System For Embedded Linux, Seoul National University Graduate School Electrical Engineering, p7, 2002.