전력전자학술대회 논문집 2004. 7.12 ~ 7.15

ARM Processor를 이용한 UPS의 인터넷 감시

이동근*, 오성진*, 김경환*, 김욱동*
이화전기공업(주)*

Internet Monitoring System for UPS using ARM Processor

DongKeun Lee, SungJin Oh, KyungHwan Kim, UkDong Kim
Ehwa Technologies Information

ABSTRACT

본 논문에서는 여러 산업의 전원계통에서 정전 및 순간정전 등의 사고로 발생할 수 있는 피해를 예방하는데 중요한 역할을 하고 있는 무정전전원공급장치(UPS: Uninterruptible Power Supply)의 인터넷 감시시스템에 대하여 기술한다.

별도의 네트워크 인터페이스장치를 설치할 필요가 없이 UPS를 인터넷에 직접 연결할 수 있도록 개발한 임베디드 시스템(Embedded System)의 하드웨어 구성과 다양한 네트워크 통신 프로토콜을 구현할 수 있도록 설계된 소프트웨어의 구성 및 특징을 설명하고, 인터넷에 연결된 UPS의 고장을 예방하거나 신속하게 처리할 수 있도록 개발하여 사용 중인 EMS(Ehwa Management Service)의 특징을 기술한다. 또한 EMS 서버를 이용하지 않고 간단하게 웹브라우저를 통해 감시할 수 있도록 Java 기술을 이용하여 구현한 웹감시프로그램에 대해서 기술한다.

1. 서 론

UPS는 은행, 방송, 증권거래소, 전화교환국, 병원 그리고 반도체 공장 등 여러 산업에서 전원계통의 정전, 순간정전, 전압강하에 대한 대비책으로 주로 사용되고 있으며 특히 일부 반도체 제조 설비용은 정전에 대한 대비뿐만 아니라 양질의 전원 공급 장치로써 UPS의 사용이 필수적이다. 이와 같이 UPS는 중요한 부하의 안정된 전원공급을 보장하는 장치이므로 고장이 발생한 경우 그 파급효과가 상당하기 때문에 신속한 처리가 필요하며, 정상동작중이더라도 축전지의 충전상태 등의 중요한 정보를 항상 감시하여 사고발생의 요인을 사전에 예방하여야한다. 그러므로 UPS의 신뢰도 향상에 있어 감시시스템의 역할이 매우 중요하다.

본 논문에서는 전국에 걸쳐 설치, 운전 중인 UPS를 인터넷을 이용하여 온라인으로 감시하는 UPS 원격감시 시스템 개발에 대해서 기술한다.

기존에 UPS를 인터넷에 연결하기 위해서는 상품화 되어 있는 별도의 네트워크 인터페이스장치를 직렬통신으로 UPS의 제어보드와 연결하는 것이었다. 이러한 네트워크 인터페이스장치들은 대부분 SNMP(Simple Network Management Protocol) 또는 TCP/IP(Transmission Control Protocol/Internet Protocol) 등의 프로토콜 중에서 한 가지만 지원하기 때문에 두개의 프로토콜 또는 새로운 프로토콜이 필요할 경우에 유연하지 못한 단점을 가지고 있다.

본 논문에서는 LINUX 운영체제를 탑재(Porting)한 임베디드 시스템을 사용하여 매우 간단하고 효율적으로 인터넷에 접속할 수 있는 네트워크 인터페이스 방법을 제시하며, 이것을 이용하여 전국에 설치되어 있는 UPS를 인터넷으로 통합, 관리하고 있는 시스템에 대해서 기술한다.

2. 통신시스템 구성

일반적으로 UPS는 장비전면에 UPS의 동작상태 표시 및 제어를 담당하고 있는 장치인 제어디스플레이장치(CDU: Control & Display Unit)를 가진다. 논문에서는 CDU 기능을 담당하는 마이크로프로세서에 네트워크통신용 프로토콜을 동시에 지원할 수 있도록 설계하여 별도의 통신 인터페이스장치가 필요 없도록 구성하였다. 또한 새로운 프로토콜의 지원이 필요할 경우가 발생하더라도 기존의 시스템처럼 주프로그램 자체를 변경할 필요 없이 임베디드 시스템에 탑재한 LINUX 운영체제에서 실행되는 응용프로그램을 새로이 작성하여 추가해주기만하면 되는 구조로 설계하여 응용성 및 확장성을 좋게 하였다.

본 시스템의 하드웨어 및 소프트웨어의 구성은 다음과 같다.

2.1 하드웨어 구성

CDU의 CPU(Central Processing Unit)는 현재 휴대전화 및 PDA 등에 널리 사용되고 있는 범용 마이크로프로세서인 ARM(ARM7TDMI)을 사용하여 저전력, 고성능의 기능을 구현하였다.

EMIF(External Memory InterFace)로 16MByte의 SDRAM과 8MByte의 Flash ROM을 사용하며, CS-8900으로 Ethernet을 구현하여 웹감시를 실행한다. 그 밖에 디버깅용도의 RS-232와 UPS의 주제어를 담당하는 DSP(Digital Signal Processor)와 통신하기 위한 RS-485 직렬통신 포트를 갖는다.

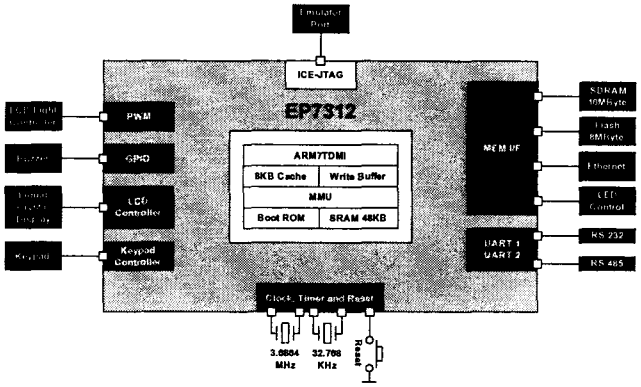

그림 1 CDU 하드웨어 구성도
Fig. 1 CDU Hardware Configuration Diagram

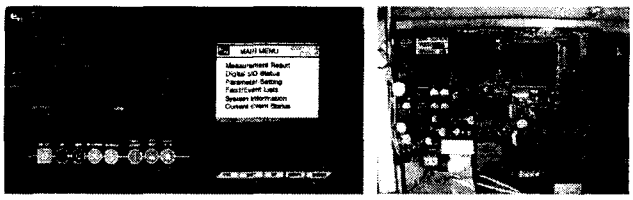

그림 2 CDU 외관 및 보드
Fig. 2 CDU Exterior & Board

2.2 소프트웨어 구성

모두 4가지의 응용프로그램으로 구성되어 있으며 그림 3과 같다. 각 프로그램은 LINUX 운용체제하에서 공통된 데이터를 이용하여 서로 다른 역할을 수행하도록 제작되었다. 새로운 프로토콜을 지원할 필요가 있는 경우, 공통데이터를 이용할 수 있도록 해당 프로토콜을 지원하는 프로그램만 새롭게 작성하여 그림 3의 Application영역에 추가하는 것으로 간단하게 구현할 수 있다.

프로그램의 전체적인 동작은 다음과 같다.

- 주프로그램은 UPS의 각종 정보들을 실시간으로 수집하여 공유메모리에 저장 또는 갱신한다.
- 나머지 모든 프로그램들은 공유메모리에 저장된 정보를 이용하여 해당기능을 수행하는데 필요한 데이터로 사용한다.

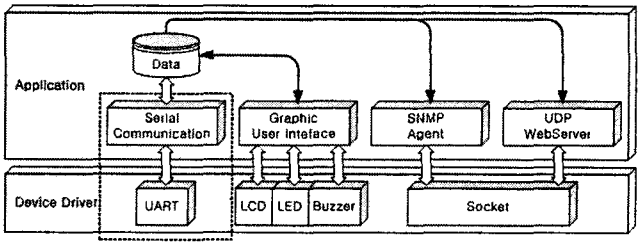

그림 3 CDU 소프트웨어 구성도
Fig. 3 CDU Software Configuration Diagram

각 프로그램들의 동작은 다음과 같다.

2.2.1 Serial Communication Application

RS485 포트를 이용하여 UPS의 DSP 제어보드와 실시간으로 통신하며 각종 정보들을 수집하고 주기적으로 공유메모리에 저장 및 갱신한다. 그림 4와 같이 독자적인 스케줄러에 따라 동작하고 소프트웨어적으로 크게 세 가지 계층으로 구분되며 각각의 역할은 다음과 같다.

- ESCI(ETUS Serial Comm. Interface) Layer:
· 스케줄러에 의해 만들어진 통신패킷을 순서대로 큐(Queue)에 저장한다.
· PACKET층에서 올라온 데이터를 해석한다.
- PACKET Layer:
· DSP로 송신할 데이터를 프로토콜에 맞도록 패킷으로 만들어 준다.
· DSP로부터 수신된 데이터를 프로토콜 SYNTAX만 조사하여 그대로 ESCI층으로 보낸다.
- Read | Write Layer:
· Physical층에 요청한 Byte수만큼 읽거나 쓴다.

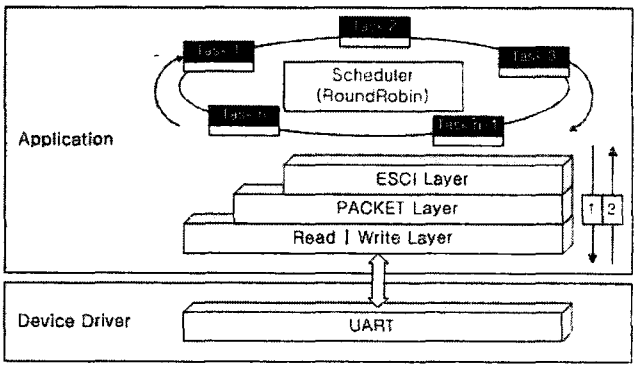

그림 4 직렬통신 응용프로그램의 계층 구조
Fig. 4 Layer Structure of Serial Communication Application

2.2.2 Graphic User Interface Application

메뉴구동방식의 사용자 인터페이스 프로그램으로 공유메모리에 저장된 정보를 읽어와 장비 전면에 있는 CDU의 LCD(Liquid Crystal Display)화면 및 LED(Light-Emitting Diode) 또는 Buzzer를 통해 UPS의 현재 상태를 표시하며, UPS ON/OFF 등의 제어버튼을 눌러 공유메모리의 특정 값을 변화시켜 직렬통신프로그램에 의하여 UPS 제어를 수행한다.

2.2.3 SNMP Agent Application

NMS(Network Management System) 및 인터넷 감시에 사용할 목적으로 제작된 UPS Agent 프로그램으로 네트워크 혹은 인터넷에 연결된 Manager 컴퓨터에서 원격지의 UPS Agent들을 통합하여 관리하는데 사용된다.

이 프로그램은 Manager 컴퓨터에서 관리정보를 요청하면 해당 데이터들을 공유메모리에서 읽어와 응답하거나(Polling) 정전 등의 특정경보가 발생한 경우에는 Manager의 요청이 없어도 UPS의 경보 상태에 대한 정보를 Manager에 자동으로 송신하는 (Trap) 역할을 한다.

2.2.4 Web-Server Application

SUN microsystems의 Java 기술로 만들어진 프로그램으로 UDP(User Datagram Protocol)를 이용한다. 네트워크나 인터넷에 연결된 Java Runtime이 동작하는 PC(Personal Computer)의 웹브라우저에서 요청한 데이터를 공유메모리에서 읽어와 응답한다. 데이터통신은 CDU에 내장된 Java Servlet과 PC의 웹브라우저로 다운로드 된 Java Applet 사이에서 이루어진다.

3. 인터넷을 이용한 UPS의 감시

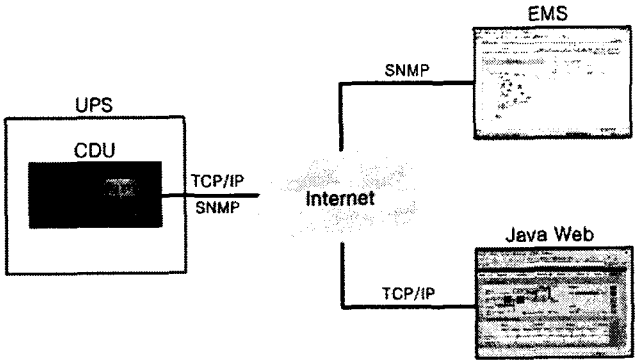

그림 5 인터넷을 이용한 UPS 감시 구성
Fig. 5 UPS Monitoring Configuration using Internet

개발된 네트워크 인터페이스는 그림 5와 같이 전국에 설치된 UPS를 중앙감시센터에서 통합하여 관리하는 EMS와 웹브라우저를 통해 일대일로 연결하여 감시할 수 있는 Java Web으로 구성되어 있다. UPS가 EMS에 연결되어 관리되고 있는 중에도 사용자가 Java Web을 이용하여 웹브라우저를 통해 동시에 접근할 경우에도 충돌 없는 감시가 가능하도록 하였다.

3.1 EMS(Ehwa Management Service)

인터넷을 이용한 UPS의 통합관리서비스로 UPS 전용 NMS라 할 수 있으며, 그림 6과 같이 구성되어 현재 운용 중에 있다.

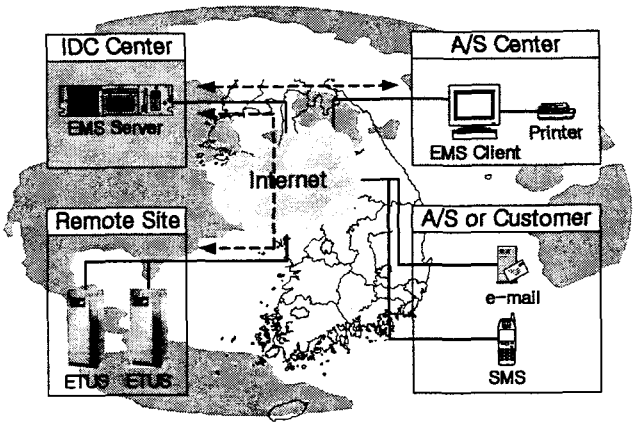

그림 6 EMS 구성 및 개념도
Fig. 6 EMS Configuration & Conceptual Diagram

EMS는 Microsoft의 .NET 기술로 만들어진 프로그램으로 프로토콜은 SNMP를 사용하였고 관리정보기준(MIB: Management Information Base)은 UPS의 표준인 RFC1628(Request For Comments) 문서를 기준으로 제작되었다.

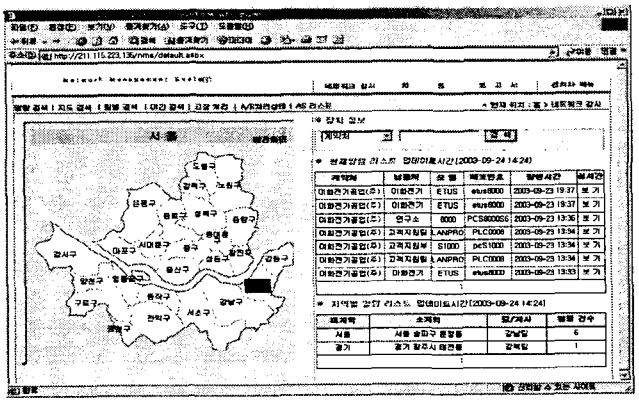

그림 7 EMS 화면 예
Fig. 7 The Example of EMS Window

EMS 서버는 전국에 설치된 UPS의 관리정보를 주기적으로 수집하여 데이터베이스에 저장하고 관리자는 EMS 서버를 통해 직접 관리하거나 외부에

서 인터넷으로 EMS 서버에 접속하여 웹브라우저를 통해 관리할 수 있으며, 관리자가 감시 중이 아닌 상태에서도 경보상황이 발생하게 되면 E-Mail 및 SMS(Short Message Service)로 알려주므로 신속한 처리가 가능하다. 또한 데이터베이스에 저장된 정보는 UPS의 이력으로 관리되거나 각종 보고서, 차트 등으로 표시하여 분석 자료로 활용할 수 있다.

3.2 Java Web-Server

Java 기술을 이용한 감시프로그램으로 다양한 기기의 웹브라우저 환경에서 동작하며, 여러 장소에서 PC를 이용하여 동일한 UPS를 동시에 감시할 수 있다. EMS의 경우와 달리 별도의 감시용 PC가 필요 없으며 또한 별도의 NMS 소프트웨어가 필요 없다.

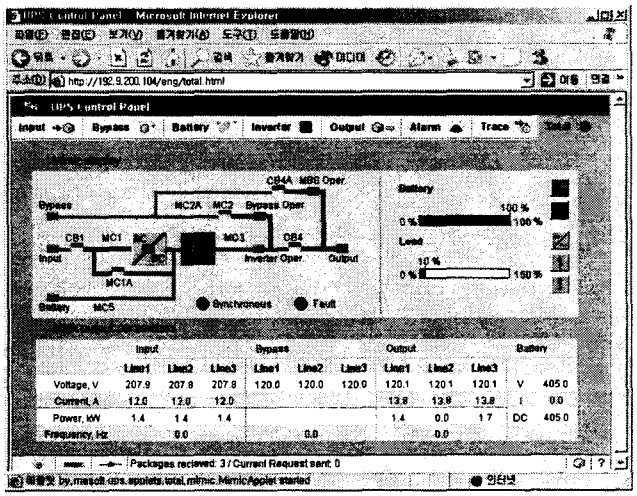

그림 8 Java Web 감시 화면 예

Fig. 8 The Example of Java Web Monitoring Window

4. 결 론

본 논문에서는 UPS의 인터넷 감시를 위하여 개발한 임베디드 시스템의 하드웨어와 소프트웨어의 구성과 동작에 대하여 설명하였고 실제로 운용하고 있는 UPS 인터넷 감시 서비스인 EMS에 대하여 기술하였다.

개발한 시스템의 주요 특징은 다음과 같다.

1) ARM Processor를 사용한 임베디드 시스템의 구성으로 목적에 따라 다양한 응용프로그램의 제작 및 추가가 용이하다.
2) 공유메모리를 이용한 소프트웨어구조로 다양한 통신환경과 프로토콜에 대처하기 쉽다.
3) EMS의 SMS 및 E-Mail 서비스로 UPS의 고장 상태를 신속하게 파악, 처리할 수 있다.
4) EMS의 데이터베이스에 저장된 정보들을 이용하여 UPS의 이력으로 보관할 수 있으며, 이를 통해 앞으로 일어날 수 있는 사고에 대비할 수 있는 중요한 자료로 활용할 수 있다.
5) EMS의 보고서 및 차트 기능으로 데이터분석에 활용할 수 있고 고객에 주기적인 보고서 제출이 가능하다.

현재 사용 중인 EMS는 정전 등의 사고를 신속하게 처리하거나 현재 상태 및 과거의 고장이력 등 UPS 관리에 중요한 데이터를 수집하고 분석하여 발생할 수 있는 사고를 미리 예방하는데 활용되고 있으며, 여러 산업의 전원 계통에서 중요한 구성요소로 사용되고 있는 UPS의 신뢰도를 높이는데 중요한 역할을 할 것으로 기대된다.

참 고 문 헌

[1] John Catsoulis, "Designing Embedded Hardware", November 2002, O'LEILLY®.

[2] Karim Yaghmour, "Building Embedded Linux Systems" April 2003, O'LEILLY®.

[3] 박영환, "임베디드 시스템 임베디드 리눅스", (주)사이텍미디어 2002년 6월 인쇄 및 발행.

[4] 이영준, "SNMP기반 SNMP/OSI 통합 망 관리 시스템 설계 및 구현", 컴퓨터교육학회 논문지, 6권, 3호, pp. 57~64

[5] 안성진, 조영욱, 유승군, "SNMP MIB-II 를 이용한 인터넷 관리 시스템의 웹 인터페이스 설계 및 구현", 정보처리학회 논문지, 6권, 3호, pp. 699~709