

실시간 운영체제 DSP/BIOS를 이용한 3상 무정전전원공급장치(UPS) 소프트웨어 설계

오성진, 김경환, 이동근
이화전기공업 기술연구소

Software design of three phase uninterruptible power supply(UPS) using the realtime operating system DSP/BIOS

Seongjin Oh, Kyunghwan Kim, Dongkeun Lee
Ehwa Technologies Infomation

ABSTRACT

대부분의 전력전자 시스템과 마찬가지로 3상 UPS 시스템 역시 전력전자 알고리즘 개발자가 응용프로그램을 작성할 때 하드웨어를 조작하고 초기화하는 하위 수준 코드를 작성해 왔다. 이와 같은 응용 소프트웨어와 하드웨어의 밀접한 통합은 시스템 성능차원에서는 부족함이 없으나 이식성과 유지보수 측면에서 많은 문제점을 보여왔다. 또한, 다양화 된 고객요구에 의하여 시스템이 점점 더 복잡해짐에 따라서 기존 소프트웨어 구성은 한계를 이르렀다.

본 논문에서는 TI사의 실시간 운영체제인 DSP/BIOS를 이용하여 3상 UPS 시스템 하드웨어와 응용프로그램을 분리하는 추상화 작업을 통해서 이루어진 성과에 관하여 제시한다. 분업화되고 상호 연동하는 방식의 개발 패러다임의 변화는 전력전자 알고리즘 개발자가 더 많은 시간을 알고리즘 개발에 투입 할 수 있게 하였고 다른 분야 개발자가 병렬로 소프트웨어 개발을 진행함에 따라서 전체적으로 개발기간의 단축을 가져왔다.

1. 서론

전통적으로 DSP 응용 프로그램은 매우 단순한 구조인 인터럽트를 포함한 라운드 로빈 스케줄링(Round-robin Scheduling) 구조를 주로 이용하여 왔다. 이 구조는 하나의 무한 루프와 비동기적 이벤트를 처리하는 인터럽트 그리고 타이머를 이용한 주기적인 태스크 처리루틴들로 이루어져 있다. 이 구조의 장점은 직관적으로 이해하기 쉽고 간편하며 프로그램 크기를 최소한으로 유지 할 수 것이다. 특히, 적은 개발 인원으로 효과적인 결과를 얻을 수 있다. 단점으로는 요구되는 기능 수가 많아지면 프로그램이 복잡해지고 유지 보수가 어려워진다. 또한, 이 구조의 큰 단점은 여러 프로그램 개발자간 협업이 쉽지 않다는 점이다.

대부분의 임베디드 시스템이 이미 직면한 도전적 과제들은 더 이상 전력전자 시스템에서만 예외일 수 없다. 3상 무정전전원공급장치에 대한 고객들의 요구 사항이 점점 복잡해지고 있으며 동종업체간 경쟁도 치열해 지고 있다. 이에 따라서 시장 출시까지의 기간(Time-to-market)의 단축이 가속화되고 있는 추세이고

제품의 가격은 더 낮아지고 있다. freescale, Intel, TI 등의 임베디드 시스템 프로세서를 생산하고 있는 주요 공급사들은 급속하게 신제품을 출시하고 있다. 이와 같은 외적 변화에 대응하기 위하여 3상 무정전전원공급장치 개발자들은 제품 개발에 필요한 총 비용은 줄이면서 개발 속도는 빠르게 가져가야 한다. 이 목적을 달성하기 위해서는 시스템 설계단계에서 충족되어야 할 중요한 원칙들이 있다. 첫 번째는 하드웨어 호환성(Flexibility)이다. 빠르게 변화되고 있는 프로세서에 독립적인 시스템 설계가 필요하다. 두 번째는 소프트웨어 재사용성(Reusability)이다. 마지막으로 손쉬운 개발(Development Ease)이다. 개발 자체가 쉽게 이루어 질 수 있는 여러 가지 툴들이 구비되어 있는가를 조사하고 적극적으로 이 툴들을 사용하는 것이다.

2. 3상 무정전전원공급장치 시스템 설계

2.1 설계 목표

가장 중요한 목표는 전력전자 응용 소프트웨어와 시스템 소프트웨어의 분리이다. 그림 1과 같이 소프트웨어가 분리됨에 따라서 전력전자 엔지니어들은 제어 알고리즘(Control algorithm) 개발에 집중할 수 있었으며 시스템 소프트웨어 개발 엔지니어들과 동시 작업이 가능하게 되어 개발 기간을 단축을 가져왔다.

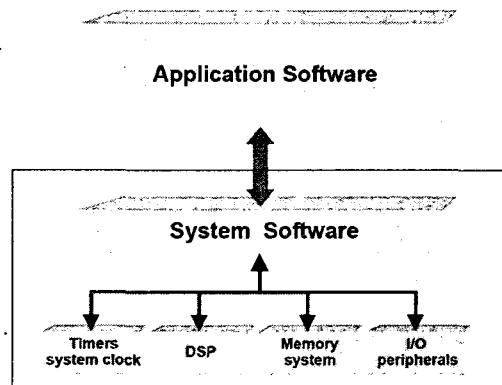


그림 1 3상 무정전전원공급장치 소프트웨어 컴포넌트
Fig. 1 Three phase uninterruptible power supply software component

두 번째로, 시스템 소프트웨어의 설계 목표는 손쉬운 이식(Migration)이다. 이 목표는 미래에 발생하는 하드웨어의 변경에 보다 빠르게 대처하기 위하여 반드시 필요하다. 세 번째 목표는 보드 지원 패키지(BSP)를 구축하고 그 책임을 하드웨어 개발 엔지니어의 몫으로 남기는 것이다. 이로써 하드웨어 개발자와 소프트웨어 개발자간 분리를 이루어 낼 수 있었다.

2.2 실시간 운영체제(Real-time OS) 선정

설계 목표들을 이루어 내기 위해서는 태스크들 간의 동기화와 통신 체계가 매우 중요한 요소이다. 이것은 물론 실시간 운영체제 없이도 가능하지만 개발의 용이성과 효율성을 높이기 위해서는 적절한 실시간 운영체제를 선택하는 것이 바람직하다. 서론에서 언급한 바와 같이 호환성, 재사용성, 개발 용이성을 적절히 만족할 수 있는 실시간 운영체제의 선택이 필요하다. 이런 목적에 맞는 실시간 운영체제 중 하나는 TI사에서 개발된 DSP/BIOS이다. DSP/BIOS는 경성 실시간 커널(Hard real-time kernel)로 TI사의 C6000, C5000 DSP 제품군과 C2000의 일부 DSP에서 이용 가능하여 호환성이 매우 높고 쉽고 다양한 개발툴을 포함하고 있다.

2.3 하드웨어 구조

그림 2에서 보는 것과 같이 주 프로세서는 TI사의 C6711(250MHz)이고 보조 프로세서로 F2812(150MHz) 2개를 이용하고 있으며 디지털 로직처리를 위하여 사이링스사의 Spartan3(XC3S400)를 채택하였다.

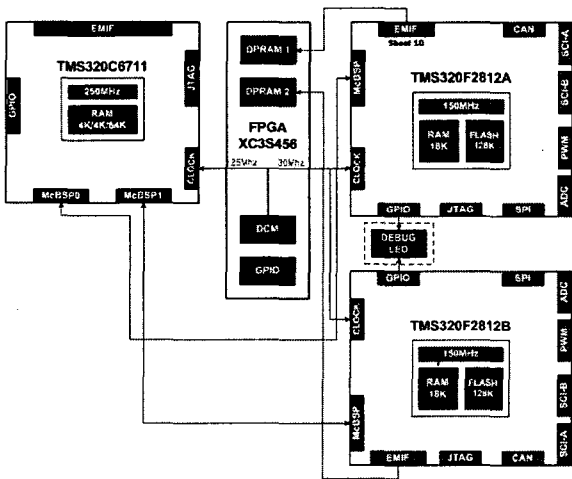


그림 2 3상 무정전전원공급장치 DSP 보드 블록도
Fig. 2 Block diagram of the DSP board of three phase uninterruptible power supply

하드웨어 개발자들은 시스템 소프트웨어 개발자를 위해서 보드 지원 패키지(BSP)를 구축하고 그 패키지 이용에 필요한 매뉴얼을 작성하여 제공함으로써 소프트웨어의 하드웨어 의존성을 상당히 줄일 수 있었다.

2.4 소프트웨어 구조

시스템의 전체 소프트웨어 구조는 그림 3과 같다. 맨 상위 층은 전력전자 소프트웨어 개발자들에 의하여 개발

된다. 이 층에 포함된 소프트웨어는 전력전자 제어 알고리즘과 시스템의 운전 상태의 변화를 처리하는 유한 상태 머신(FSM, Finite State Machine)이 주를 이룬다. DSP/BIOS 커널과 디바이스 드라이버를 포함하고 있는 시스템 소프트웨어 층은 주로 시스템 자원관리를 담당하고 전력전자 제어 알고리즘 소프트웨어를 위한 기반 구조(Infrastructure)를 이루고 있다. 또한 물리적 하드웨어 층과 절연된 층을 이루고 결과적으로 손쉬운 이식(Migration)이 가능하도록 설계하였다.

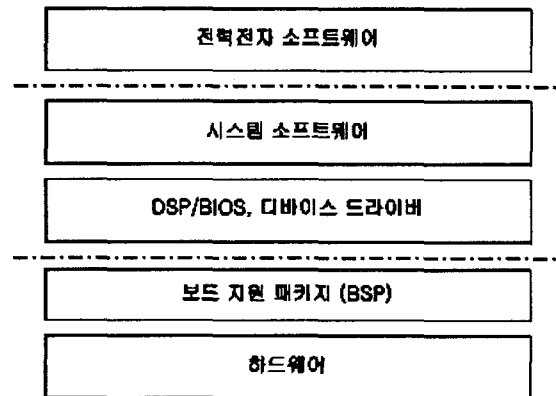


그림 3 3상 무정전전원공급장치 소프트웨어 계층 구조
Fig. 3 Three phase uninterruptible power supply software hierarchy

3. 결론

본 논문에서는 지금까지 3상 무정전전원공급장치 개발에 있어서 종래의 백그라운드 모델 설계 방법에서 벗어나 실시간 운영체제를 이용한 새로운 설계 방법이 갖는 이점에 대하여 살펴보았다. 이와 같은 개발 패러다임의 변화는 모든 임베디드 시스템을 적용하는 거의 모든 분야로 확대되고 있는 중이며 그 적용 속도도 점차 가속화될 것으로 예상된다. 이는 예상의 근거는 모델 기반 설계(Model-Based Design)의 등장과 자동 코드 생성(Automatic Code Generation) 기술의 발전이다. 전력전자 엔지니어들은 앞으로 제어 알고리즘과 운전 상태 변화 처리 모듈을 직접 손으로 프로그램을 작성하는 일이 점차 줄어들게 될 것이고 그 대신 자신의 제어 알고리즘 개발에 더 많은 시간을 투자할 수 있을 것이며 상태처리 모듈은 이와 같은 상황이 전개되기를 진정으로 원하고 있다.

참고 문헌

- [1] David Stegner, "Embedded Application Design Using a Real-Time OD", Automation Conference, Vol. :21-25 pp. 151-156, 1999, June.
- [2] Qing li, Caroline Yau, Real-time concepts for embedded systems: First Edition, CMPBooks, pp. 214-230, 2003.
- [3] Andy The, "How to Get Started With the DSP/BIOS Kernel", TI Application Report, 1999, June.