

# 국산 신기술이 하루속히 출현해야

실시간 운영체제의 시장동향은 CPU, 하드웨어, 소프트웨어 개발툴 시장동향, 그리고 상용 하드보드 시장의 동향까지 두루 포함해야만 정확하게 파악이 가능할 것이다. 그러나 일단 이 글에서는 상용 RTOS의 동향만을, 특히 국내 및 북미 시장 동향을 중심으로 서술하고 검토하도록 하겠다. 물론 여기서 상용 RTOS 시장동향이 주이므로 다른 시장동향은 철저히 상용 RTOS 공급 업체의 시각일 수 밖에 없으며 다소 다른 시각이 있을 수 있음을 미리 밝혀둔다.

■ 이두원/㈜두울정보기술 대표(<http://www.doall.co.kr>, [dwlee@doall.co.kr](mailto:dwlee@doall.co.kr))

## 연재 순서

- 1 실시간 운영체제(RTOS) 개요
- 2 실시간 운영체제 시장동향
- 3 실시간 운영체제와 임베디드 시스템
- 4 적용기술과 적용사례 - 이번호
- 5 상용 실시간 운영체제 가이드

## 서론

임베디드 시스템이나 실시간 운영체제(RTOS)의 특징으로는 메모리와 CPU 파워 등 제한적인 환경에서 원하는 기능을 최적으로 구현해야 하기 때문에 프로그래밍 기술 뿐만 아니라 하드웨어에 대한 폭넓은 지식이 요구된다. 이러한 특징으로 인하여 실시간 처리를 포함한 운영체제에 관한 전반적인 기술과 지식을 바탕으로 한 다른 여러 기술을 응용하여 적용될 수 있는 분야는 광범위하게 이루어 질 수 있다.

여기서는 먼저 실시간 운영체제와 임베디드 시스템에 관한 적용 기술들에 대해 이론적인 내용과 제품의 예로 알아보고 현재 이러한 기술을 응용한 실제 제품이나 적용 사례를 알아 보도록 한다.

## RTOS 및 임베디드 시스템의 기술적 내용

일반적인 운영체제는 기껏해야 ms(밀리초) 단위의 정확성으로 제어를 하지만, 통신기구나 정밀 제어의 경우 수십( $\mu$ s(마이크로초)) 단위로 정확하게 시간을 측정해야 하는 경우가 많다.

또한 인터럽트가 발생하였을 때 해당 프로세스나 태스크가 수십 마이크로 초 이내로 동작해야 하는 경우에는 일반적인 범용 운영체제로는 이러한 조건을 만족할 수가 없다. 그래서 나온 것이 바로 실시간 운영 체제인데 이 실시간 운영체제의 기술적 구

현 원리에 대하여 알아보도록 한다.

### Real-Time과 운영체제의 관계

운영체제는 모든 태스크의 실행을 통합하고 지원하는 요소를 갖추어야 하고 인터럽트를 다루는데 있어 명확한 기능을 수행해야 하고 다음과 같은 기능을 제공해야 한다.

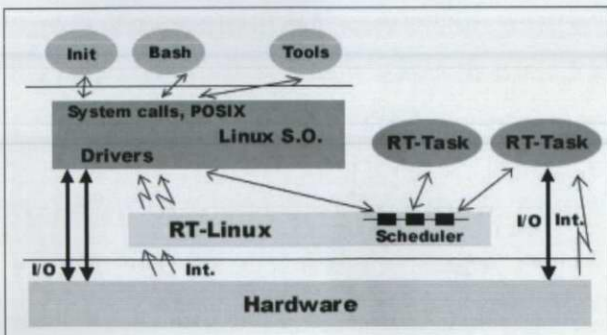
- 스케줄링 알고리즘
- IPC(세마포어, 메시지 등) 메커니즘
- 인터럽트
- 각 태스크의 주기적 활성화

범용 운영체제와는 반대로 실시간 운영체제의 목표는 복잡성을 최소화하는 것이다. 이것은 실시간 운영체제에서 많은 작업을 수행하는 운영체제가 필요하다는 것이 아니라 태스크가 예측 가능하고 빠르게 실행되는 것이 중요하게 여겨진다는 의미이다.

그러나 종종 범용 운영체제보다 실제로 실시간 운영체제가 느리게 보여지는 경우를 경험할 수 있다. 예로서 예측 가능한 값을 얻기 위하여 캐시 메모리 사용을 사용하지 않고 개발하는 경우가 있는데 이때에는 실행 속도가 떨어지는 현상이 나타날 수 있다. 즉 캐시 메모리, 파이프라인을 통한 프로세서 및 분기 알고리즘은 실시간 운영체제에서 예측 가능한 요소를 어렵게 하는 요인이 된다(참고 Site-<http://linux-embedded.com>).

### 임베디드 시스템 프로그래밍

임베디드 시스템 프로그래밍이라고 일반 프로그래밍과 다를 것이 없지만 디버깅이나 분석에 있어서 세심한 작업을 필요로 한다. 대상이 되는 하드웨어가 제한적이기 때문에 이러한 자원을 충분히 활용하여 프로그래밍을 해야 하며 개발 환경 구축에 많은 노력이 요구된다.



〈그림 1〉 Real-Time 기능이 구현된 RTLinux 설계 구조

개발에 있어서 함수관련 명령을 수행하기 위해서는 런타임 라이브러리를 만들어야 하고 그 밖의 적당한 개발 상태 표시를 위해 하드웨어 디버거를 붙이는 방법

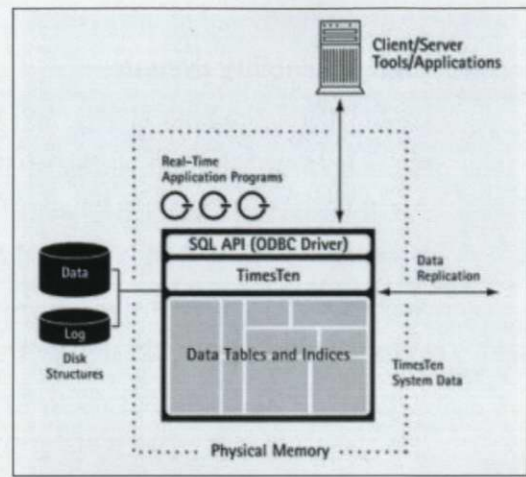
도 있는데 이 경우에는 RTOS가 지원하는 디버거를 이용하거나 혹은 디버거에 맞추어 코드를 작성하는 방법이 있다. 보통 임베디드 시스템은 한 번 전원이 들어오면 지속적인 동작을 요구하기 때문에 메모리의 효율적인 관리가 필요하다. 초기 시작 부분에서 프로그램이 사용할 메모리를 미리 할당해버리고 수행 중에는 동적으로 메모리를 할당받지 않는다. 이렇게 함으로써 속도와 안정성을 높일 수 있다. 임베디드 시스템 프로그래밍의 가장 중요한 기술은 마이크로 프로세서 종류에 따른 메모리 사용량을 줄이는 코딩 방법이 핵심이 될 것이다(참고 site-<http://www.embedded.com>).

### 적용기술

앞서 실시간 운영체제와 임베디드 시스템에 대한 이론적 기술 내용에 대하여 알아 보았다. 이러한 운영체제 및 시스템 내용을 바탕으로 다른 기술들을 포함하는 하나의 응용 기술에 대하여 알아본다.

### 메모리 상주형 데이터베이스

애플리케이션 개발에 있어서 데이터베이스 지원에 대한 부분이 중요한 위치를 차지하고 있으며 실시간 운영체제에서는 실시간 기능과 임베디드 기능을 지원하는 데이터베이스가 필요로 된다. 이러한 요구에 맞는 기술이 바로 메모리 상주형 데이터베이스이다. 가장 큰 특징은 메모리에 상주하여 기능을 수행하기 때문에 성능과 속도면에서 많은 장점을 가지고 있으며 비용과 안정성면에서도 적용가치가 충분한 점을 가지고 있다. 음성 데이터 전송, 인터넷 어플리케이션, PDA 분야 등에 적용되고 있다(참고 Site-<http://www.timesten.com>).



〈그림 2〉 메모리 상주형 DB제품 구성

### High-Availability System

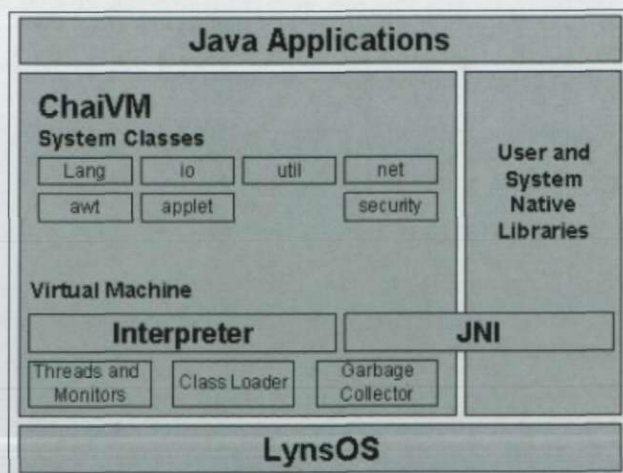
현재의 컴퓨터 시스템들은 예전과는 달리 대형화되고 각 시스템의 중요성도 증대되고 있다. 비즈니스적이나 서비스적인 측면으로 볼 때 이러한 시스템들의 연속적이고 안정적인 유지가 가능하도록 해야 할 것이다. 이때 적용되는 기술이 바로 HA(High Availability)기술이며 실시간, 임베디드 분야에도 함께 적용되어 이루어진다. 현재 HA관련 하드웨어 장비들이 계속 개발되고 있으며 이에 맞는 소프트웨어적인 개발도 함께 이루어지고 있다. 물론 실시간 운영체제도 기본으로 이러한 하드웨어에 포팅 되고 실제 개발 환경이나 시스템에 적용되고 있다.

### Embedded Java

임베디드 시스템에서는 여러 장점을 가진 다른 기술들의 접목이 상당한 중요성을 가지고 있다. 재사용성, 확장성 등의 특징을 가진 자바는 이러한 특징이 요구되는 임베디드 시스템에서 적용하려는 노력이 있었으며 여기서 탄생한 것이 바로 임베디드 자바이다.

임베디드 자바는 임베디드 디바이스에 맞게 설계되어 있어 적은 리소스를 가지고 최적의 성능을 낼 수 있으며, POSIX 기반의 스케줄링, 쓰레드의 활용으로 실시간 운영체제에 적합하게 개발되었다.

이상과 같이 적용기술들을 간략히 살펴 보았는데 여기서 알아본 응용 기술은 특징이 될만한 것들만 알아본 것이고 뿐만 아니라 Hot Swap, Embedded Web-server, Real-Time ORB, Modular Kernel Technology 등 무수히 많은 기술들이 실시간 또는 임베디드 분야에 적용되어 가고 있다.



(그림 3) Embedded Java 구조도(상용 실시간 OS인 LynxOS위의 ChaiVM 구성 예)

(표) 임베디드 시스템 개발 환경

구성 목록	내용
Host PC	● Cross 개발 환경 구성(컴파일러, 유틸리티 등)
Motorola MBX860 Board	● Host PC 및 Target Board는 모두 네트워크망에 연결
Linux 커널 소스	● Host와 Target은 Serial Cable로 연결(터미널 화면)
GCC 및 Binutil	● Linux 커널 부트 이미지만 Target에 직재하고 나머지
그 밖의 유틸리티	관련 파일은 Host PC에서 NFS 부팅 시킴

### 적용사례

디지털 시대로 발전되어가고 있는 현재에 고 성능과 고 가용성을 가진 장비의 개발과 이에 맞는 적용 기술이 요구되고 있다. 실시간 및 임베디드 시스템의 적용 사례로는 임베디드 보드에 특정 운영체제를 올려 애플리케이션을 개발 할 수 있는 환경의 예를 보여 주는 것이 가장 이해하기 쉽고 확실한 예가 될 것이다. 간단한 예로서 Motorola MBX860 보드에 LinuxPPC를 포팅하여 간단한 임베디드 작업환경을 구성할 수 있는 경우가 있다(참고 Site-<http://www.ppc.kernel.org>, <http://rtlinux.to>). Lucent Technologies 사에서 제공하는 멀티미디어 통신 교환 서버이다. 이 장비는 네트워크 상에서 실시간 통신 기능을 향상시키기 위한 인터넷 프로토콜 툴로 구성되어 있다.

유럽 Airbus 항공사의 항공기인 Airbus A340, A330, A320기에 차세대 항법 시스템에 실시간 및 임베디드 시스템이 적용되어 운영되어 있으며, 차세대 보잉 737 비행 시뮬레이터에도 이러한 실시간 기술이 적용되어 졌으며 현재 다양한 비행 훈련용으로 쓰여지고 있다.

간단히 두 가지의 적용 예를 보여 주었지만 이러한 분야 뿐만 아니라 PDA 단말 분야, 무선 통신 분야 등 현재 관심의 대상이 되고 있는 모든 분야에 폭넓게 도입되어 있다.

### 결론

오늘날 실시간 운영체제와 임베디드 시스템에 대한 관심이 점차 높아가고 있고 실제로 기술력을 가지고 개발하는 업체들도 많아 졌다. 그럼에도 불구하고 지금까지 알아본 적용 기술이나 사례가 거의 외국의 것이었다. 가능한 빠른 시일 내에 좋은 솔루션들이 나와 줌으로써 자체적인 기술력을 가진 기술 선진국이 되기를 바란다.