

Embedded Device 기능과 최적화 방안

Embedded Device Implementation and Optimization Technique

*김한규¹, #장주수²

*H.G. Kim¹, #J.S. Jang(jsjang@moassoftware.co.kr)²

¹(주)신명정보통신, ²(주)모아소프트

1. 서론

본고는 산업원천융합기술 개발사업의 “신속 대응 가능한 BIS (Built-In Sensor) 기반 자율 지능형 사출성형시스템 개발” 과제의 연구내용이다.

IT 융합의 관점에서 Embedded Device (M2M Device)는 IT Convergence 의 기본요소로 IT 화 되어있지 않아 IT 화를 추구하는 모든 분야에 중요한 역할을 한다. M2M Device 의 기능을 구현하는 Embedded System 은 간단하게 “사람 대 사물, 사물대 사물간의 지능화된 서비스를 실시간으로 이용 가능한 IT 인프라”라고 정의 한다.

본고에서는 마이크로 사출성형 현장에서 M2M Device 의 역할과 기능, 필요성 및 구현에 있어서의 최적화 방안과 대하여 설명하고자 한다.

2. Embedded Device 기능

M2M Device 는 마이크로 사출성형 현장에 존재하는 시스템 구성장치들인 사출성형기와 금형에 Built-In 된 센서로부터 발생되는 데이터를 센서노드를 통하여 수집한다. 제어요소로서 수집된 데이터를 분석하여 운영자가 원하는 설비의 제어를 위하여 스마트폰이나 운영자 단말기로부터 전달되는 제어 명령을 다양한 인터페이스를 통하여 수행한다. 폐쇄적인 시스템 (Fanuc, 스미토모 사출 성형기)과 통신기능을 제공하지 않아 정보화 되지 않은 설비 (금형 자체)들로부터 Raw Data 의 발생과 수집기능을 지원하여 독립적인 설비들을 유기적으로 연동시켜 운용 설비 및 시스템 간의 정보 통합화를 가능하게 하여 가동률을 향상시키고 상태추론 알고리즘을

탑재하여 고장예지보전기능을 구현하여 정밀도 향상시킨다..

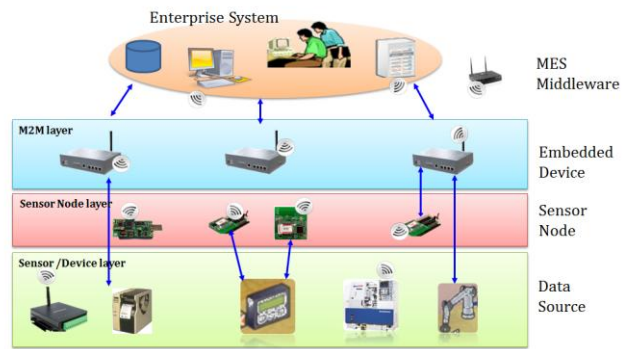


Fig. 1 M2M Device 를 통한 IT 융합의 Layer 구조

3. 최적화 필요성

M2M Device 는 제한된 메모리와 제한된 저장장치 그리고 디바이스의 Flash Memory 에 탑재되어 있는 Embedded 운영체제를 사용하므로 일반 컴퓨터 시스템의 경우와 같이 다양한 기능들을 수행하지 않고 사용하는 Application 에 dedicated 된 기능만을 수행한다. 그러므로 다양한 통신기능, 상태추론기능, 데이터 가공 등의 기능을 완벽히 수행하기 위하여서는 최적화 과정을 반드시 거쳐야 한다.

M2M Embedded Device 의 최적화에는 여러 가지 기법이 있다. M2M Device 의 최적화 방안에는 하드웨어와 소프트웨어 개발 영역의 최적화뿐 아니라 각 기능을 어떻게 최적의 성능을 낼 수 있도록 구현할 것인가 하는 비교기준을 구현하는 것을 포함한다.

최적화를 구현하기 위하여서는 먼저 M2M Device 를 구성하는 각각의 하드웨어와 소프트웨어의 리소스에 대하여 그 기능과 역할 등 상세한 내용이 파악되어야 한다. 여기에는 프로세서의 성능, Flash 메모리의 크기 및 쓰기 횟수제한, SDRAM 크기와 Application Firmware 예상크기, 각종 인터페이스 성능 등 하드웨어적인 요인과, 운영체제의 실시간 지원여부, 사용언어의 선택, 디버깅 방법, 컴파일러를 통한 최적화, 변수, 함수, 자료 구조 및 형태, 연산자, 메모리 할당, 분기문과 루프의 처리 등 소프트웨어적인 최적화를 포함한다.

최적화를 통하여 얻고자 하는 것은 원하는 기능의 최고 성능을 만들어 내는 것이다. 성능과 제한된 리소스 사이에는 항상 Trade-off 가 존재한다. 성능을 최고로 하기 위해서는 많은 리소스(메모리, 프로세서 점유율)의 확보가 필요하나, 과도한 리소스의 사용은 제한된 Embedded Device 에서 오히려 성능을 저하시키는 원인이 된다.

4. 최적화 기법

최적화는 주어진 리소스와 기능에 대한 상세한 지식에서부터 시작된다. Embedded Device 와 일반컴퓨터와 가장 큰 차이점은 데이터 저장장치이다. Flash Memory 는 쓰기의 제한이 있다. 일반적으로 한 어드레스에 10~100 만회 정도 쓸 수 있다. 한 어드레스에 쓰기 제한 이상으로 모니터링을 위한 데이터를 0.1 초 단위로 계속 쓴다면 곧 프로그램에 더 이상 쓸 수 없는 문제가 발생 할 것이다. 이러한 문제는 프로그램의 구현 방법으로 풀어나가야 한다. 개발 환경에 있어 C 와 C++ 가 Embedded 개발 언어로 사용되고 있으나 실행 프로그램 사이즈가 작고 속도가 가장 빠르며 하드웨어 제어가 손쉬운 C 언어를 사용하는 것이 추천된다. Embedded System 에서는 일반적인 데스크 탑 컴퓨터에서 사용하는 표준 라이브러리 함수들을 모두 다 사용할 수는 없다. 예를 들어 Sleep, printf, time 등 표준함수를 지원하지 않는

컴파일러가 대부분이다. 속도의 최적화와 리소스 의 최적화는 반비례하므로 성능의 최적화를 위해서는 절충점을 찾아야 한다. Native Type 의 변수를 사용하고, 분기문과 I/O 를 줄여서 속도를 높일 수 있으며, 데드코드를 제거하고, 전역변수 대신 지역변수를 쓰고, 자동으로 초기화 되는 전역변수의 코드는 초기화하지 않으며, 표준 라이브러리 함수의 사용을 자제하여 리소스인 ROM 을 절약하는 최적화를 할 수 있다. 또한 역어셈블러 기능을 사용하여 실제의 개발 언어 소스코드와 어셈블러로 변환된 대응 코드를 비교 하여 최적화의 기준을 정할 수 있다. 매트릭스 등 수학 계산을 위한 디버깅 방법은 사용자가 직접 만들어야 한다.

5. 결론

최적화된 Embedded Device (M2M Device) 는 IT Convergence 의 기본 요소로 IT 화 되어있지 않아 독립적으로 운영 될 수 밖에 없는 설비와 기기들을 다양한 방식의 인터페이스를 통하여 연결시키며 정보통합을 이루는 역할을 한다. 마이크로 사출성형 현장에서 이루어지는 본 과제의 시스템 구성 장치들인 사출 성형기와 금형에 Built -In 된 센서로부터 데이터수집 및 제어 그리고 고장, 예지, 보존 을 위한 추론 시스템 등 탑재, 비 표준 성형기와 이 통신기능들은 M2M Device 를 통하여 구현된다..

후기

본 논문은 지식경제부에서 수행하는 산업 원천융합기술 개발사업에 의해 수행 되었으며 이에 감사 드립니다.

참고문헌

1. Caroline Yao, "Real Time Concepts For Embedded Systems" p53-64.
2. Soto,, "Embedded Sketches", p15-29
3. ETRY "Standards Research Center "M2M Standardization and its Prospects", 2011