

Nano-Q+에서 센서의 자동슬립을 지원하는 전력 매니저

최복동*, 은성배*
*한남대학교 정보통신공학과
e-mail:qtisrael@hanmail.net

A Power Manager supporting automatic sleep for Sensors in Nano-Q+

Bokdong Choi*, Seongbae Eun*
*Dept of Information and Communication Engineering, Han-Nam University

요 약

배터리를 사용하는 센서노드의 전력 소모를 줄이기 위해 많은 방법들이 제안되어 있다. 본 논문에서는 MCU 및 센서의 전력을 관리하는 전력 매니저를 제안한다. 센서의 타입을 설정하고 제안되어 있는 센서의 추상화를 추가한 매니저를 제안한다. Nano-Q+가 스케줄링할 때 센서의 타입을 판단하여 전원을 관리 할 수 있도록 한다.

1. 서 론

USN(Ubiquitous Sensor Network)은 다수의 센서노드들을 이용하여 우리의 생활을 편리하게 만드는 환경이다. 작은 크기의 센서노드는 상시전원이 아닌 배터리를 사용하는 경우가 많다. 전력 소모를 줄이기 위해서는 네트워크 통신 알고리즘을 개선하거나, 저전력 모듈을 만들어야 한다. 기존의 센서노드 OS는 저전력 소모를 위해 많은 종류의 관리 방법과 매니저를 제공한다.

기존의 센서 네트워크 OS는 저전력 관리를 위한 방법으로 MCU의 자동슬립 모드를 제공한다. 하지만 OS가 센서의 자동슬립을 제고하지 못한다. 센서의 종류가 매우 많고 특성이 서로 다르기 때문이다. 그래서 MCU가 슬립하고 있는 동안에도 센서는 전력을 사용함으로써 불필요한 전력 낭비를 가져온다. 센서의 종류나 특성에 상관없이 관리하기 위해서는 센서의 추상화[1]가 필요하다.

본 논문에서는 표준화된 센서노드 플랫폼 기반에서 MCU와 센서의 자동슬립 매니저를 제시한다. 모든 태스크가 수행을 완료하고 새로운 태스크를 기다릴 때 MCU와 센서를 함께 자동 슬립 시킨다. 센서의 종류가 다양해 관리하기 힘들었던 부분은 센서의 추상화에서 보완한다. 기존의 센서 네트워크 운영체제인 Nano-Q+[2]에 센서의 추상화가 제안 되었다. 제안된 추상화 위에 MCU와 센서를 자동슬립 시키는 전력 매니저를 제안한다.

2. 배경

2.1. Tiny-OS

기존 센서노드 운영체제로서 가장 먼저 개발된 Tiny-OS[3]는 하드웨어의 제약을 고려하여 매우 소형으

로 개발되었다. 저전력 기능을 위주로 개발되었으며 컴포넌트 기반의 프로그래밍 패러다임을 지원한다. 컴포넌트의 명령은 이벤트 처리기에 의해 상태변화를 감지하고 수행된다. 센서노드가 일을 하지 않는 시간에는 슬립 모드로 전환하여 저전력 파워 소비를 지원한다.

2.2. Nano-Q+

Nano-Q+ 운영체제는 에너지 소모를 최소화하고자 저전력 슬립모드를 제공하고, 제한된 메모리 사용을 최소화하도록 멀티쓰레드 간의 스택을 공유한다. C 기반의 프로그램을 지원하며 멀티 쓰레드 스케줄러 방식 등을 특징을 가진다. Nano-HAL이라는 디바이스 드라이버 영역이 있어 하드웨어를 추상화하지만 사용자는 API를 다 알아야 하는 단점이 있다.

본 논문에서는 제안되어 있는 센서 추상화와 함께 기존 센서노드 운영체제의 문제점을 해결하려 한다. 센서의 추상화는 많은 센서에 상관없이 통일된 API를 제공한다. 이러한 API를 이용하여 센서의 전원을 관리하는 전력 매니저를 제안한다.

2.3. 소형 임베디드 시스템의 전력관리

소형 임베디드 시스템의 전력관리 기법[4][5]들은 기존의 많은 연구에서 다양한 방법으로 소개되고 있다. 먼저 회로 설계 단계에서부터 장치를 저전력으로 동작하도록 설계하는 하드웨어 측면에서의 관리 기법이 있다.

운영체제 혹은 시스템 소프트웨어 측면에서는 프로세서의 공급 전압을 조절하여 전력 소모량을 줄이는 동적 전압조절 기법(DVS), 장치의 상태에 따라 적절한 파워 상

태로 전이하여 전력 소모를 줄이는 동적 전력관리 기법(DPM) 등이 있다.

운영체제에서는 전력관리를 위하여 APM, APM의 진보된 형태인 ACPI[6]를 제공한다. APM 혹은 ACPI를 이용하여 여러 가지 전력관리를 할 수 있지만 이는 바이오스와 운영체제에서 모두 이 기능을 지원해야 사용할 수 있다.

3. 제 안

1. 센서의 분류

센서는 주기적으로 데이터를 폴링하는 것과 이벤트가 발생했을 때 데이터를 전송하는 것으로 나눌 수 있다. 폴링형 센서는 주기적인 센서 데이터를 평균하여 처리한다. 필요치 않을 시에는 전원을 차단해서 전력의 낭비를 막을 수 있다. 온도, 조도, 습도 등을 이용한 각종 환경 모니터링, 위치인식서비스 등이 있다.

이벤트형 센서는 필요한 이벤트가 발생했을 때에만 데이터를 전송한다. 신뢰성 있는 데이터 전달이 필수적으로 요구된다. 이벤트 발생 시기를 예측할 수 없어 센서의 전력을 함부로 차단할 수 없다. 그래서 저전력이 요구되지만 구현하기가 어렵다.

2. 센서의 지정

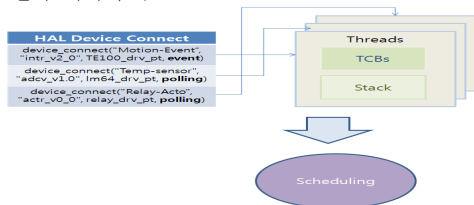
센서 투명성에서 응용 프로그래머는 device_connect()를 이용하여 HW mapping table을 연결한다. 하지만 센서 투명성에서는 센서의 타입을 지정하지 않고 있다. 센서를 polling, event 두 가지 방식으로 지정하고, 기존의 방식에 아래와 같이 추가하고자 한다.

<표 1> 센서의 타입 지정

HAL Device Connect
device_connect("Motion-Event", "intr_v2_0", TE100_drv_pt, event)
device_connect("Temp-sensor", "adcv_v1.0", lm64_drv_pt, polling)
device_connect("Relay-Acto", "actr_v0_0", relay_drv_pt, polling)

쓰레드 생성시 사용하는 센서의 타입을 확인하고 polling 시에만 저전력 매니저를 사용한다. event 시에는 사용자가 지정한 특별한 경우가 아니고는 저전력 매니저를 사용하지 않는다.

3. 전력 매니저 구조



(그림 1) 전력 매니저

디바이스를 연결할 때 센서의 타입을 정해준다. 쓰레드는 생성될 때 센서의 타입을 넘겨받는다. 스케줄러는 MCU를 슬립 시킬 때 센서 타입을 판단하고 polling 시나 사용자의 요구 시 Sensor를 슬립 시킨다.

4. 저전력 스케줄러

Nano-Q+에서 쓰레드 문맥 교환은 현재 쓰레드 문맥의 저장, 쓰레드 스케줄링, 다음 실행될 쓰레드의 문맥 복구로 이루어진다. 이 과정에서 현재 쓰레드는 준비 큐에 들어가고 스케줄러를 통해 준비 큐에 들어있는 쓰레드들 중 다음에 실행될 특정 쓰레드를 선택하게 된다.

쓰레드가 Sleep 상태에 들어가 준비 큐에 들어있는 쓰레드가 없을 때 MCU를 Sleep 시킨다. MCU를 Sleep 시킬 때 센서의 타입을 확인한다. 센서의 타입이 polling 시에는 센서의 전원을 off 한다. 센서의 타입이 event 인데 센서의 슬립을 원하거나, 센서의 타입이 polling 인데 센서의 지속적인 전원이 필요할 때가 있다. 이때는 사용자가 옵션을 줘서 다양한 경우의 수를 지원한다.

4. 결 론

본 논문에서는 센서 투명성을 지원하는 플랫폼 위에서 MCU와 Sensor를 슬립 시키는 전력 매니저를 제시 하였다. Sensor를 제어하기 위해 제안되어 있는 센서 투명성을 이용하였고, 구현하기 위해서는 표준화된 플랫폼이 필요하다.

다양한 특성의 Sensor를 제어하는 방법이 표준화 된다면 H/W 개발과 S/W 개발에 큰 영향을 미친다. 이는 USN 산업에 활기를 가져오고 생산성과 기술력 개발에 좋은 영향을 미칠 것이다.

참고문헌

[1] 은성배, 소선섭, 김병호 / 센서투명성을 지원하는 센서 노드 운영체제 구조 / 한국컴퓨터종합학술대회 논문집 Vol15, No 1

[2] S. Park, J. Kim, K. and D. Kim, / "Embedded Sensor Networked Operating System" Proc. of 9th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing, 2006.

[3] T. Schmid, H. Dubois-Ferriere, and M. Vetterli, "Sensorscope: experiences with a wireless building monitoring sensor network," Proc. of Workshop on Real-World Wireless Sensor Networks, 2005

[4] 이형석, 정영준 / 임베디드 운영체제 커널 기술동향 / 전자통신동향분석 제21권 제1호

[5] 이재동, 허정연 / 태스크 중기화가 필요한 임베디드 실시간 시스템에 대한 효율적인 전압 스케줄링 / 정보과학회논문지: 시스템 및 이론 제 35권 제 6호(2008.6)

[6] 황영시, 정기석 / E-ACPI / 대한전자공학회