

IEEE 802.15.4 기반 센서 네트워크를 위한 저전력 실시간 플랫폼의 설계 및 구현

*황태호, 송병철, 김성동
전자부품연구원

e-mail : taeo@keti.re.kr, songbc@keti.re.kr, sdkim@keti.re.kr

Implementation of the low power platform for sensor network based IEEE 802.15.4

*Tae-ho Hwang, Byung-chul Song
Korea Electronics Technology Institute

Abstract

The sensor network that may be deemed to fall in the field of ubiquitous computing performs the basic function of transmitting sensing data through the autonomous sensing and the Ad hoc network. In order to collect and treat various sensing data at the time of application and manage extremely limited system resources, the sensor network requires the embedded operating system that uses low power, a small cord size and the least hardware resources.

In this paper, The operating system having a new structure for constructing the IEEE 802.15.4 MAC and Zigbee sensor network is suggested and can be formed by reviewing the characteristics and the core structural requirements of the operating system for the sensor network based on operating systems, which have been formed under existing similar conditions, and applying such features and core structural requirements to the design of the operating system for achieving the features and the requirements.

I. 서론

기존의 컴퓨팅 플랫폼과 달리 무선 센서 노드들은 컴퓨팅에 필요한 성능, 파워와 메모리, 배터리 등의 모든 자원이 극도로 제한적이다. 또한, 센서 네트워크를 수

만개의 작은 자율적인 센싱노드들로 구성된 분산 컴퓨팅 플랫폼으로 보는 데이터 중심형 (Data-centric) 프로그래밍 스타일이 일반적이다.

센서 노드들의 이러한 특징 때문에 극도의 저전력 경량 장치에서 동작할 수 있으면서 동시에 환경과 응용 프로그램의 변화에 대처하기 위해 동적인 재구성을 지원할 수 있는 특별한 구조의 플랫폼이 필요하다. 또한, 센서 노드를 위한 운영체제는 네트워크와 MAC을 위해 동시에 여러 개의 비동기적인 이벤트를 다룰 수 있어야 하며 응용을 통한 분산형, 데이터 중심형 프로그래밍 모델을 지원해야 한다. 이렇게 상반된 것처럼 보이는 요구조건을 모두 만족시키기 위해서는 새로운 구조의 운영체제 및 프로그래밍 모델을 설계하고 구현하는 것은 매우 어려운 일이다.

본 논문에서는 컴퓨팅 분야에서 제안되었던 Dual Priority Scheduling Algorithm에 근거하여 센서노드에 적합하게 단순화된 새로운 구조를 제안한다. 설계한 시스템 소프트웨어 구조는 상용 8051 코어의 8bit MCU에서 구현하여 기능 및 성능을 검증하였고, 자체 개발한 8051 호환 코어와 RF&MODEM 칩에 포팅하여 IEEE 802.15.4 MAC 프로토콜을 통한 네트워킹을 구현하였다.

II. 관련 연구 분석

2.1. Low power Operation

센서 플랫폼에서 저전력을 위한 주요한 내용은 다음과 같다.

- $P \approx p_t \cdot C_L \cdot V_{dd}^2 \cdot f_{clk}$
 P : Power Dissipation
 p_t : The probability of switching in power transition
 C_L : Loading capacitance
 V_{dd}^2 : Voltage supply
 f_{clk} : Clock Frequency
- Use of the Power Saving Mode supported by MCU core

시스템의 전원 소모(P)는 위의 비례식에서와 같이 공급되는 전압(V_{dd})와 클럭(f_{clk})에 비례한다. 공급되는 전압과 클럭의 관리가 저전력을 위한 전원관리에 있어서 중요한 리소스가 될 수 있다. 원칙적으로 시스템에서 수행되는 작업들의 우선순위에 따라 수행시간 동안 필요한 만큼의 차별화된 전원 및 클럭을 공급하는 것이 가장 이상적이다. 현재 대부분의 MCU-core에서는 이러한 기능을 제공하기 어렵다. 이러한 기능을 위해 MCU-core의 논리 게이트부터 다시 설계를 해야 하는 문제점 때문에 최근의 대부분 상용 MCU-core에서는 몇 단계의 discrete한 레벨의 전압과 클럭을 차별화 할 수 있는 기능을 지원하고 있다.

저전력 동작을 위한 추가적인 방법은 시스템의 하드웨어 블록 별로 차별적인 재구성이다. 사용하지 않는 주변 장치들의 전원은 차단하는 방식으로 수행할 작업에 따라 센서 플랫폼 상에서 차별적인 시스템을 구성하도록 하는 기능이다.

이러한 저전력을 위한 전원관리에 있어서 센서 노드에서 수행되는 작업들에 대한 수행 주기, 필요한 하드웨어 리소스, 수행시간 등의 명확한 정의가 필요하며, 잘 정의된 작업들에 대해서는 예측 가능한 효율적인 구조의 스케줄러를 통해 실행할 수 있는 구조로 Abstraction되어야 한다.

2.2 IEEE 802.15.4

IEEE 802.15.4의 개략적인 특징은 다음과 같다.

- Low-rate : 250 kbps, 40 kbps, and 20 kbps
- Start or peer-to-peer operation
- Allocated 16 bit short or 64 bit extended addresses
- Allocation of guaranteed time slots (GTS)
- Slotted CSMA-CA channel access in a Beacon-Enabled Network
- Fully acknowledged protocol for transfer reliability
- Low Power consumption
- Energy detection (ED)
- Link quality indication (LQI)

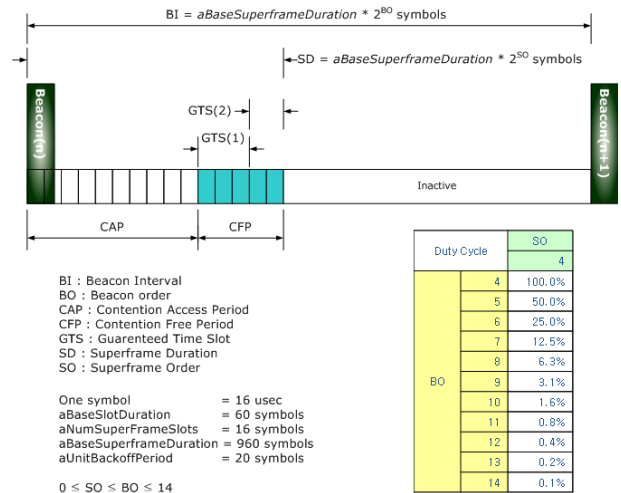


그림 1. IEEE 802.15.4 Super-frame Structure

IEEE 802.15.4는 저전력의 낮은 전송률의 무선 통신을 위해 설계되었다. 동작에서의 특징은 그림 1와 같은 Super Frame의 구조를 가지고 있어서 주기적으로 전송되는 Beacon Frame을 통해 동기화되어 구분된 타임슬롯 구간 내에 데이터의 전송 및 연결의 관리와 같은 모든 작업들 수행되는 특징을 가지고 있다. 따라서, 네트워크 상에서 발생하는 모든 이벤트들은 사전에 예측 가능한 주기적인 작업들로 구성 될 수 있다.

2.3 Zigbee Network

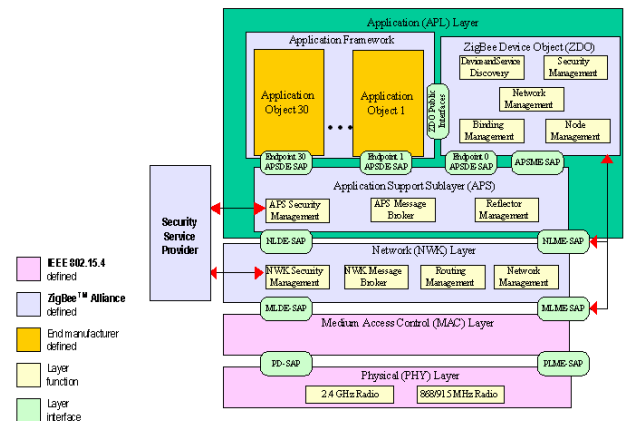


그림 2 . Zigbee stack

하위의 MAC/PHY는 IEEE 802.15.4를 사용하며 상위 네트워크 레이어는 Star, Tree, Mesh Topology를 정의하고 있다. Zigbee Network Layer는 Ad hoc routing을 통한 센서 네트워크 액세스를 기본으로 하고 있다. 시스템의 구조에 있어서 이러한 특징을 반영하기 위해서는 MAC에서의 주기적인 작업에 동기화되어 발생하는 비주기적인 이벤트를 빠른 응답성을 가지면서 저전력의 효율인 처리 구조가 되어야 한다.

III. 플랫폼 구현

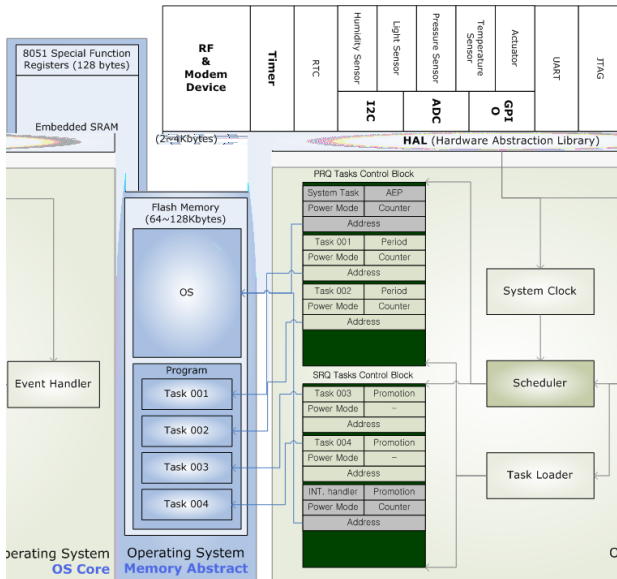


그림 3 . Platform Structure

3.1 HW Platform

운영체제의 구현을 위해 제작한 센서 노드(S-Point)의 하드웨어는 8051-core를 사용하여 원칩화한 모듈이다. 그림 9의 센서 노드를 위한 칩의 내부 구조를 나타낸다. 8051-core를 중심으로 32KBytes의 플래쉬 메모리 가지고 있으며 128 Bytes의 내부 SRAM을 가지고 있다. 8051-core는 여러 장치들의 제어를 위하여 코어 내부에 SRAM과 주소 영역을 공유하는 128Bytes의 SFR(Special Function Register)영역을 가지고 있으며 CPU 코어 외부에 2KBytes의 External SRAM을 가지고 있다. (그림4)

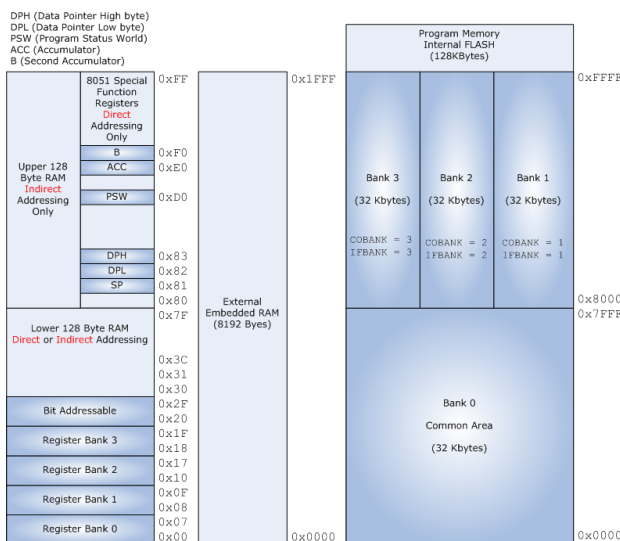


그림 4 . System Memory Abstract

그림4에서 DATA 영역으로 구분된 내부 메모리와 XDATA 영역으로 구분된 External 메모리는 서로 다른 인스트럭션으로 액세스하며 SFR영역은 Indirect Addressing으로 접근한다.

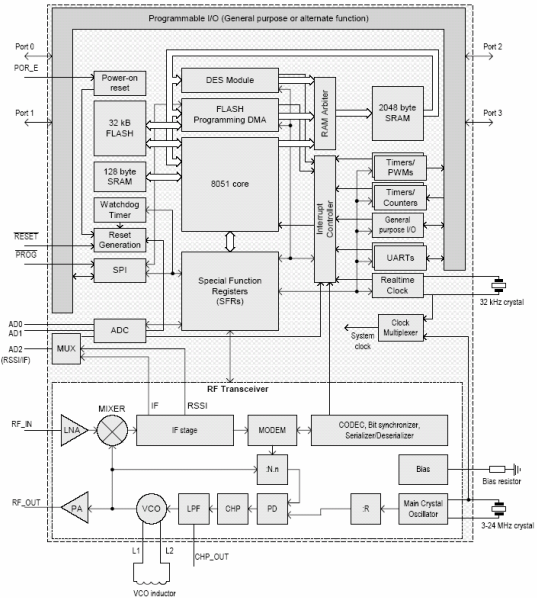


그림 5 . Hardware Block Diagram

그림5에서와 같이 센서 장치로부터 데이터를 센싱하기 위한 2개의 ADC (Analog-to-Digital Converter)를 가지고 있으며 2개의 Timer/Counter, RTC(Real Time Clock) 블록을 칩 내부에 가지고 있다. 그리고 Host Interface를 위해 2 채널의 UART와 무선 통신을 위한 RF/Modem 블록을 가지고 있다. RF블록은 무선 PAN 통신을 위한 900MHz, 2.4GHz대역의 RF 신호처리를 담당하며 Modem블록은 IEEE 802.15.4에 근거한 데이터의 모듈레이션/디모듈레이션을 담당한다. 이러한 8051-core 외부의 장치들의 액세스는 앞서 설명한 SFR 영역의 레지스터를 통해 제어하며, 칩 외부의 장치들은 General I/O Port를 통해 액세스한다.

3.2 SW Platform

그림 3에서와 같이 하드웨어 장치들의 개념적인 블록들이 있으며 하드웨어 블록들은 HDF (Hardware Definition File)의 코드들에 의해 그림4의 메모리구조와 함께 Abstract된다. 어셈블리 혹은 C코드로 구현된 HDF는 주로 하드웨어의 정의 및 저 수준의 I/O를 담당한다. HDF를 통해 Abstract된 시스템의 각 컴포넌트는 HAL(Hardware Abstract Layer)을 통해 Access된다. HAL은 이벤트 핸들러에 의해 시스템 콜로 Wrapping되는 실제적인 Device Driver 코드들이 구현된다. HAL은 Standard C Libraries 및 어셈블리러를 이

용하여 구현되며 정적으로 빌드 된다. HAL에서 제공하는 기능 중에 LR-WPAN의 네트워킹을 위한 시스템 콜을 제공한다. 하위 PHY를 이 이용하여MAC의 프로토콜이 구현되며 MAC을 이용하여 단순하면서 명료한 네트워크 프로토콜이 구현된다. 이벤트 핸들러는 HAL을 통한 시스템 콜을 제공하며 시스템 타이머 이벤트를 통해 스케줄러를 호출하여 PRQ와 SRQ의 태스크들을 수행시킨다. PRQ의 태스크로부터 발생한 이벤트를 이벤트 핸들러가 태스크 로더로 전달하여 해당된 태스크가 SRQ에 초기화되어 등록될 수 있도록 한다.

3.2 Operation

센서 노드의 요구사항을 만족하기 위해 그림 6의 Simple Dual-Priority Scheduling을 구현 하였다. 시스템의 스케줄링 큐는 PRQ(Periodic Run Queue)와 SRQ(Sporadic Run Queue)로 구분된다. PRQ는 센서 네트워크의 특성상 실시간성을 가진 주기적인 작업들로 구성되며, 비교적 짧고 명확한 작업을 수행하는 Task들로 구현된다. PRQ의 Task들은 SRQ의 Task들보다 우선순위가 높으며 Interrupt를 제외한 Non-Preemption으로 동작한다. 반면에 LRQ는 하드웨어 인터럽트 또는 URQ에 의해 등록된 비주기적인 작업들로 구성되며 우선순위가 PRQ보다 낮다. SRQ는 Preemption으로 동작하며, 하드웨어 플랫폼에서 제공하는 기능을 이용하여 필요한 만큼의 최소 전원 및 최소한의 시스템 리소스를 사용하여 동작한다.

V. 결론 및 향후 연구 방향

센서 노드는 독자적으로 동작하는 컴퓨팅 플랫폼이 아니라 여러 가지 종류의 수천 수만의 센서 노드들과 기존의 인프라와 결합되는 시스템에서 유비쿼터스의 서비스의 작은 한 부분으로서의 역할을 담당한다. 센서 노드를 위한 하드웨어 역시 보다 큰 메모리와 보다 높은 성능을 가진 저전력 칩들이 개발 중에 있다.

향후 제안된 모델을 검증하는 작업과 더불어 센서 플랫폼의 새로운 요구사항들에 대응하여 계속 발전시켜야 한다.

참고문헌

- [1] TinyOS, <http://www.tinyos.net/>
- [2] S. Hong and T.-H. Kim, SenOS: State-driven Operating System Architecture for Dynamic Sensor Node Reconfigurability. International Conference on Ubiquitous Computing (ICUC). pp. 201-203, Seoul, Korea, October, 2003.
- [3] Contiki, <http://www.sics.se/~adam/contiki/index.html>
- [4] TMO, <http://dream.eng.uci.edu>
- [5] M.Angels Moncusi and Alex Arenas, Jesus Labarta, "A Modified Dual-Priority Scheduling Algorithm for hard real-time systems to improve energy savings." Compilers and Operating System for low power, Kluwer Academic Publishers. pp.17-36, Boston, USA, 2003.

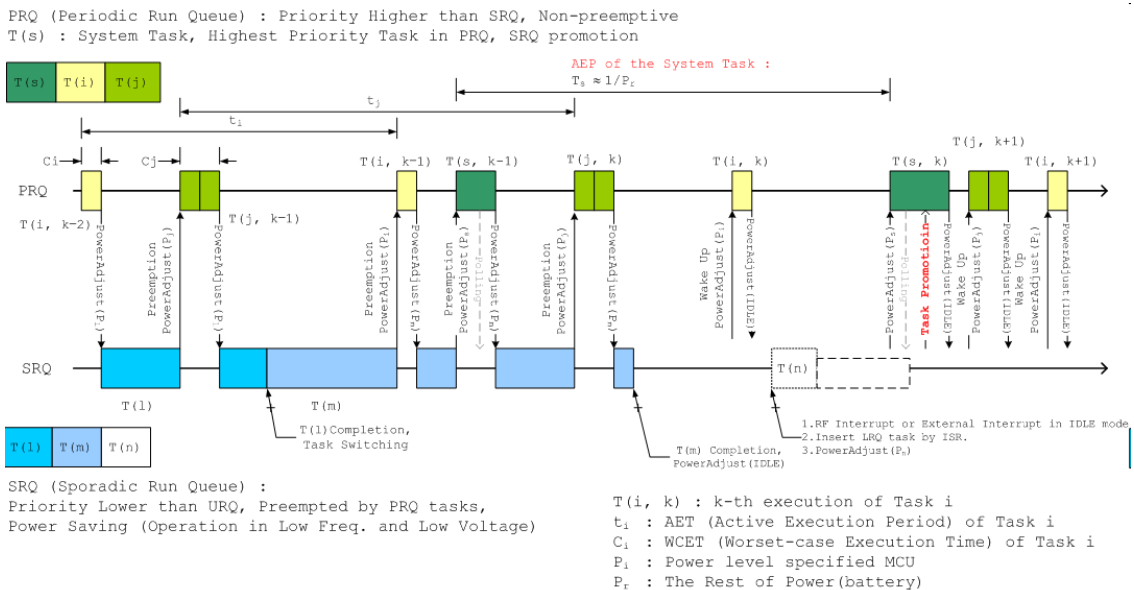


그림 6 . Simple Dual Priority Scheduling Algorithm