

주 제

임베디드 무선통신 시스템 기술

한양대학교 조인휘

차례

I. 서론

II. 센서 네트워크용 무선통신

III. 임베디드 무선통신 기술

IV. 결론

I. 서론

임베디드 시스템은 어떤 특정한 처리를 하기 위해 전용으로 설계된 하드웨어 플랫폼에 소프트웨어가 내장되어 있는 시스템으로, 최근 유비쿼터스 컴퓨팅 분야의 핵심 기술로 점점 중요성이 높아지고 있다. 또한 무선통신 기술은 임베디드 시스템이 무선 방식으로 네트워크에 연결되는 추세에 따라 필수적인 요소 기술로 인식되고 있다. 특히 임베디드 무선통신 시스템 기술은 유비쿼터스 센서 네트워크에서 대표적으로 채택되고 있다. 현재 무선통신 관련하여 내장된 소프트웨어의 대표적인 주요 기능은 우선 무선통신용 하드웨어 칩을 초기화한다. 이때 초기화 과정에서 RF 무선통신 주파수나 전력세기를 지정한다. 또한 상위 MAC 계층과 인터페이스를 수행하여, 물리계층에서 통신 하드웨어 구동을 통해 패킷을 실제 전송하거나 수신한다.

본 고에서는 임베디드 무선통신 시스템 기술을 살

펴보기로 한다. 먼저 II장에서는 현재 대표적으로 임베디드 무선통신 기술이 적용되고 있는 분야가 센서 네트워크이기 때문에, 센서 네트워크의 무선통신 기능을 IEEE 802.15.4 표준 규격에서 물리계층을 중심으로 소개한다. III장에서는 대표적인 센서 네트워크 하드웨어 및 소프트웨어 플랫폼 하에서 실제 어떻게 임베디드 무선통신 기술이 적용되고 있는지 알아본다. 마지막으로 IV장에서는 본 고의 결론을 맺는다.

II. 센서 네트워크용 무선통신

임베디드 무선통신 시스템 기술은 현재 센서 네트워크에서 대표적으로 사용되고 있다. 따라서 이 장에서는 센서 네트워크를 위해 세계 표준으로 제정된 IEEE 802.15.4 물리(PHY)계층 중에서 무선통신 부분에 우선 초점을 맞추어 기술한다. PHY는 라디오

송수신기의 활성화 및 비활성화, 현재 채널의 ED (Energy Detection), 수신 패킷의 LQI(Link Quality Indication), CSMA-CA를 위한 CCA (Clear Channel Assessment), 채널 주파수의 선택, 데이터의 송신 및 수신 등의 기능을 수행한다.

센서 네트워크용 무선통신은 다음과 같은 변조 및 확산 방식을 사용하여 하나 또는 그 이상의 주파수에서 동작한다.

IEEE 802.15.4는 제한된 전력 및 안정된 처리량의 단거리 통신을 지원하며 광범위한 응용에 사용되도록 설계되었다. IEEE 802.15.4 워킹 그룹의 주된 목표는 단순함, 낮은 가격, 이동성, 저전력 및 안정화된 처리량이다. 그러나 일반적으로 건전지가 시스템에서 높은 가격비율을 차지하고, 또 관리 및 교체를 고려한다면 시스템의 전체 가격이 증가하므로 저가격의 시스템 구성이 불가능하다. 따라서 처음 장착한 건전지의 수명을 늘리기 위한 방안이 필요하다.

1. RF 전력 및 패킷 구조

RF 전력 측정은 안테나 커넥터에 연결된 송수신기에서 수행된다. 측정 장치는 안테나 커넥터의 임피던스에 일치하거나 보정이 되어야 한다. 안테나 커넥

터가 없는 장치인 경우 측정된 RF 전력은 EIRP로 간주한다.

수신 감도는 868/915대역에서는 최소한 -92dBm이며 2.4GHz에서는 -85dBm을 요구한다. 따라서 간단하고 작은 규모로 구성할 수 있다.

PPDU(Physical Protocol Data Unit) 패킷 구조는 다음과 같다. 가장 왼쪽의 필드가 먼저 송신되거나 또는 수신된다. 다수의 Octet으로 구성된 필드나 각 Octet에서는 항상 가장 작은 위치의 Octet이나 비트, 즉 LSB가 먼저 송신되거나 수신된다.

Preamble 필드는 수신 메시지와 칩 및 심볼 동기를 확보하기 위해 송수신기에서 사용된다.

SFD 필드는 8비트의 길이로 Preamble 필드의 끝 부분이며, 동시에 패킷 데이터의 시작 부분을 가리킨다.

Frame length 필드는 7비트 길이로, PSDU에 포함된 Octet 수를 나타낸다(예: PHY 페이로드).

PSDU 필드는 가변 길이이며, PHY 패킷의 데이터를 운반한다.

PHY constants는 하드웨어 종속적이며 수행 중에는 변경되지 않는다.

〈표 1〉 주파수 대역과 데이터 전송율

PHY (MHz)	Frequency band (MHz)	Spreading parameters		Data parameters		
		Chip rate (kchip/s)	Modulation	Bit rate (kb/s)	Symbol rate (ksymbol/s)	Symbols
868/915	868-868.6	300	BPSK	20	20	Binary
	902-928	600	BPSK	40	40	Binary
2450	2400-2483.5	2000	O-QPSK	250	62.5	16-ary Orthogonal

Octets: 4	1	1	variable
Preamble	SFD	Frame length (7 bits)	Reserved (1 bit)
SHR		PHR	PHY payload

(그림 1) PPDU 구조

〈표 2〉 PHY constants

Constant	설 명	값
aMaxPHYPacketSize	PHY가 수신가능한 PSDU의 최대 크기(Octet 단위)	127
aTurnaroundTime	RX-to-TX or TX-to-RX 의 최대 turnaround time	12 심볼 주기

2. 무선통신 주요 특성

IEEE 802.15.4 물리계층은 20Kbps, 40Kbps, 250Kbps의 통신 속도를 지원한다. IEEE 802.15.4는 높은 처리량을 지원하며 저가격과 저전력으로 설계되었다.

IEEE 802.15.4에서는 시스템의 duty cycle를 이용하여 긴 동작수명을 보장한다.

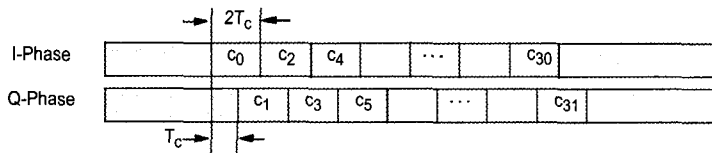
그 예로, 동작모드일 경우의 소모 전력이 10mW, 대기모드일 경우의 소모 전력이 10uW인 시스템에서 0.1%의 duty비율로 750mA의 건전지를 사용하면,

총 전력 사용량 = 99.9%의 대기모드 + 0.1%의 동작모드이므로,
 $9.9 \times 10 \mu\text{W} + 0.1 \times 10 \text{mW} = 19.99 \mu\text{W}$ 의 전력을 사용한다.

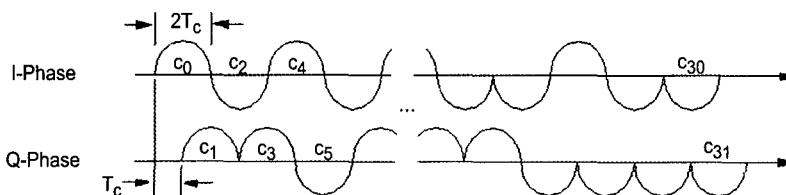
이를 기준으로 IEEE 802.15.4의 비콘 패킷의 전송 간격을 최소간격인 15.36ms에서 4분으로 늘린다면 duty cycle 비율은 2.3%에서 0.000216%로 떨어진다.

IEEE 802.15.4에서는 오직 디지털 데이터 전송을 하며 반이중 동작을 지원한다. 데이터 변조 방식으로는 868/915MHz 대역에서는 raised-cosine-shaped BPSK를 사용하며 2.4GHz 대역에서는 half-sine-shaped O-QPSK를 사용한다. 각각의 주파수 대역에서 사용되는 변조 방식들은 반송파 전송 전력의 평균 피크 수치를 체크하여 통신에 필요한 전력 소비와 복잡성을 감소시킨다. 두 방식 모두 DSSS(Direct Sequenced Spread Spectrum) 기반으로, 특히 2.4GHz 대역에서는 binary 대신에 16-ary orthogonal signaling을 사용하여 높은 데이터 전송율을 제공하고 비교적 작은 심볼을 전송할 수 있어 저전력 동작이 가능하다.

변조 및 확산과정에서 우선 4비트 단위로 하나의 심볼이 형성되기 때문에 총 16개의 심볼이 존재하게 된다. 다음 단계에서 각 심볼은 확산과정을 통해 Symbol-to-chip에서 32자리로 변환된다. 마지막으로, I-Phase와 Q-Phase로 나누어져서 $c_0, c_1 \dots$



(그림 2) O-QPSK 칩 오프셋



(그림 3) 첫 번째 심볼에 대한 칩 시퀀스 예

등 아날로그 신호로 변환되어 아래의 그림과 같이 전송된다.

IEEE 802.15.4 전송 전력은 규정상 출력이 -3dBm으로 정해져 있으며 건전지 사용도 가능하다. 또한, 안정적인고 가격이 저렴한 SOC로 구현이 가능하다.

DSSS는 전송신호의 대역폭을 증가시키는 기술 중에 하나이다. DSSS는 각 데이터 비트에 유사 무작위의 잡음인 PN시퀀스를 곱함으로써, 시간 영역에서 고속화 시켜주면 결국 주파수 영역에서는 확산이 된다. 이 방식은 신호 교란이나 다중페스 페이딩 문제에 강인하며 보안성이 우수하다. 그러나 송신기와 수신기가 항상 동기화되어 있어야 하며 수신칩에서 들어오는 신호를 비트로 변환하고 원래의 데이터를 재생해야한다.

3. 센서 네트워크 동작

IEEE 802.15.4는 두 가지 종류의 노드로 구성되며 full function 디바이스(FFD)와 reduced function 디바이스(RFD)로 구분된다. FFD 디바이스는 모든 MAC의 구성요소를 가지고 있으며 네트워크 코디네이터나 네트워크 디바이스로 동작한다. RFD 디바이스는 축소된 MAC의 구성요소를 가지고 있으며 오직 네트워크 디바이스로만 동작이 가능하다.

채널 액세스에서 FFD와 RFD는 모두 기본적으로 CSMA-CA방식을 사용한다. CSMA-CA는 센서 네트워크에서 여러 노드들이 하나의 무선채널을 공유하도록 제어하는 MAC 프로토콜로, 유선 네트워크와는 달리 충돌 회피(CA: Collision Avoidance)를 기본 원칙으로 채택하고 있다. IEEE 802.15.4에서는 예외적으로 CSMA-CA 방식의 사용 없이 전송을 하는 경우가 있는데, 비콘 또는 GTS(Guaranteed

Time Slot) 전송이 이에 해당된다.

CSMA-CA 프로토콜은 RF 무선채널의 할당을 제어하는데, 다수의 노드가 하나의 무선채널을 공유하도록 한다. 기본적으로 CSMA-CA 프로토콜은 Carrier Sensing 기능을 통해 채널의 사용여부를 먼저 확인하지만, 무선채널 환경에서는 Hidden-Terminal 문제로 무의미 할 경우도 발생한다. 그러나 이 기능은 일반적인 경우에 데이터 충돌 가능성을 낮추기 때문에 필요하다. 또한 무선채널 환경에서는 유선채널과는 달리 충돌 감지가 어렵기 때문에 충돌을 가능한 회피하기 위하여, 채널 상태를 확인하기 전에 backoff를 먼저 수행한다.

채널 상태를 평가하기 위해 ED(Energy Detection), LQI(Link Quality Indication), CCA(Clear Channel Assessment) 3가지 물리 기능을 사용한다. ED 기능은 8개 심볼 기간동안 수신 신호의 전력세기를 측정한다. LQI 기능은 ED 측정이나 SNR 평가를 통해 수신 패킷의 세기나 품질을 평가한다. 마지막으로 CCA 기능은 ED 기능을 통하여 수신 신호의 세기가 지정된 한계치를 초과하는지 또한 신호검출이 가능한지를 가지고 현재 채널의 사용 상태를 판단한다.

IEEE 802.15.4에서의 슈퍼 프레임은 비콘의 전송과 함께 시작되며 contention access period (CAP)와 GTS로 구성된다. GTS 구간은 경합이 없으며, 슈퍼 프레임 내에서 최대 7개까지 슬롯 할당이 가능하다. GTS의 할당은 전적으로 네트워크 코디네이터에 의해서 결정되며 낮은 지연시간을 요구하는 실시간 트래픽에 필요하다.

IEEE 802.15.4는 선택적으로 MAC 레벨에서 ACK를 사용하여 전송 에러에 대한 처리를 할 수 있다. 만약 요청한 ACK 프레임이 수신기로부터 도착하지 않으면 전체 프레임을 재전송 한다. 메시지가 도착하면 수신기에서는 CRC를 사용하여 프레임의 이

상 유무를 판단한 후, 만약 성공적으로 수신한 경우에만 ACK를 송신기에 보내준다.

III. 임베디드 무선통신 기술

현재 TinyOS가 포팅된 대표적인 센서 네트워크 하드웨어 플랫폼은 AVR+CC1000, AVR+CC2420, MSP430+CC2420 등 3가지 부류로 나눌 수 있다.

이것을 나누는 기준은 사용 MCU (Micro-controller Unit)와 RF module이며 MCU에 따른 구분의 기준은 코어, 전력 소모율, 연산속도를 고려해야 한다.

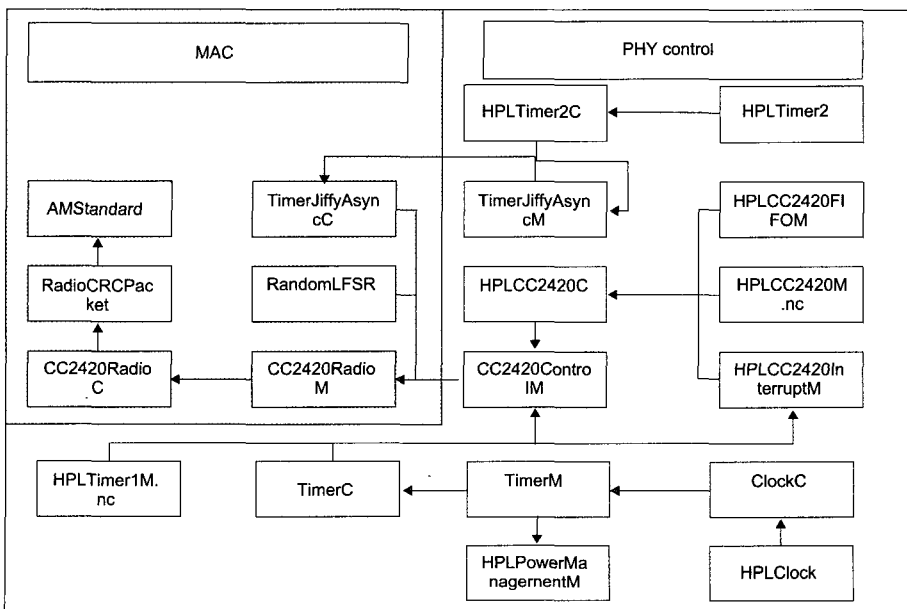
위에서 말한 MCU에 따른 하드웨어 구분의 차이점은 다음과 같다. MSP430은 사용전력과 연산속도가 AVR보다는 빠르고 데이터 RAM이 크지만, 상대

적으로 내부 프로그램 플래시 메모리와 확장성이 AVR 시리즈 보다는 떨어진다. 반면에 AVR시리즈는 많은 프로그램 메모리와 외부로 RAM의 확장이 용이하고 외부 디바이스의 장착이 용이하나 MSP430보다는 전력 소비가 많은 단점이 있다.

RF 모듈의 선택 기준은 상위 계층(MAC)에서 필요로 하는 물리 계층의 기능 지원유무와 전원 소비량, 주파수, 멀티 채널, 데이터 전송률, 변조, 코딩, 송수신 파워 소비량, 잡음 지수, Gain, RF range, RSSI 기능의 지원 유무이다. 본 고에서는 TinyOS에서 사용하는 CC2420을 통하여 무선통신 기술이 어떻게 임베디드 시스템으로 구현되고 있는지 정리한다.

TinyOS에서의 RF module의 제어 및 통신은 아래의 그림과 같이 구성된다.

(그림 4)에서 보는 것과 같이 TinyOS의 무선통신은 MAC Layer와 PHY Layer영역의 구분이 확실하다. 우선 PHY control 영역에서 HPL로 시작하는 파



(그림 4) 무선통신 구성도

일은 MCU와 RF module을 직접 제어하는 영역이고, MAC영역은 PHY영역의 정보를 바탕으로 MAC을 제어하는 부분이다. PHY영역은 두 영역으로 나뉘어진다.

첫째 영역은 MCU를 제어하는 부분이며 둘째는 RF module를 제어하는 영역이다.

HPLClock과 ClockC는 TinyOS의 동작시 기준이 되는 time tick을 생성하며, time tick을 활성화 시키거나 시간을 조절하는 부분이다. 생성 방법은 외부 32.768KHz 크리스탈에서 들어오는 클럭을 AVR MCU의 timer1에서 받아서 일정한 시간마다 타이머 인터럽트를 발생시켜 time tick을 생성한다. HPLPowerManagementM은 주기적으로 건전지의 잔량을 체크하며, MCU가 아무런 동작을 하지 않는 상태에 있다면 MCU를 standby mode로 전환시킨다. 이때 timer는 계속 동작을 한다. TimerM은 HPLClock와 ClockC에 의해 생성된 time tick을 기준으로 TinyOS에서의 소프트웨어적인 timer interrupt 발생시 제어를 한다. HPLTimer1M은 AVR MCU의 timer input capture기능을 이용하는 HPLCC2420InterruptM에서 RF module CC2420으로부터 오는 CCA의 입력시간을 체크하게 해준다.

HPLCC2420InterruptM은 CC2420으로부터의 CCA를 외부 인터럽트와 타이머 input capture를 이용하여 체크하며 CCA에 의한 외부 인터럽트가 발생하면 CC2420에 있는 FIFO에서 데이터를 읽어오고 저장하는 기능을 담당한다. HPLCC2420FIFOM은 CC2420으로부터 들어온 데이터의 저장과 출력을 담당한다. HPLCC2420M은 3가지 기능이 있다. 즉, SPI를 제어하는 것과 이것을 이용하여 MCU와 RF module의 통신을 담당하며 command와 데이터의 읽기 및 쓰기 기능을 지원한다.

HPLCC2420InterruptM과 HPLCC2420FIFOM은 HPLCC2420M 기능을 이용하여 저장 데

이터를 출력할 수 있다. CC2420ControlM은 위에서 설명된 모든 기능을 제어하는 일종의 MAC과 PHY를 연결하는 service access point의 역할을 한다. CC2420ControlM은 MAC에서의 동작에 따라 RF module의 on/off, 주파수 설정, 주파수 검출, TX/RX 간 모드 전환과 RF power를 조절한다. Timer JiffyAsyncM은 TinyOS의 MAC에서 사용하는 timerjiffy 및 ack, backoff time을 지원하는 timer 관련 컴포넌트다. CC2420RadioM은 상위 컴포넌트에서 전송 받은 data frame를 byte 단위의 변환, 데이터의 전송과 수신, 그리고 네트워크의 채널 상태를 모니터링 하는 부분이다.

HLP2420M.nc

```
command result_t StdControl.init() {
    bSpiAvail = TRUE;
    TOSH_MAKE_MISO_INPUT();
    TOSH_MAKE_MOSI_OUTPUT();
    TOSH_MAKE_SPI_SCK_OUTPUT();
    TOSH_MAKE_CC_RSTN_OUTPUT();
    TOSH_MAKE_CC_VREN_OUTPUT();
    TOSH_MAKE_CC_CS_OUTPUT();
    TOSH_MAKE_CC_FIFOP1_INPUT();
    TOSH_MAKE_CC_CCA_INPUT();
    TOSH_MAKE_CC_SFD_INPUT();
    TOSH_MAKE_CC_FIFO_INPUT();

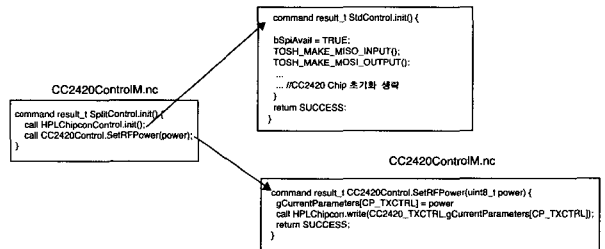
    atomic {
        TOSH_MAKE_SPI_SCK_OUTPUT();
        TOSH_MAKE_MISO_INPUT();
        TOSH_MAKE_MOSI_OUTPUT();
        sbi (SPSR, SPI2X);
        sbi(SPCR, MSTR);

        cbi(SPCR, CPOL);
        cbi(SPCR, CPHA);
        cbi(SPCR, SPR1);
        cbi(SPCR, SPR0);
        sbi(SPCR, SPE);
    }
    return SUCCESS;
}
```

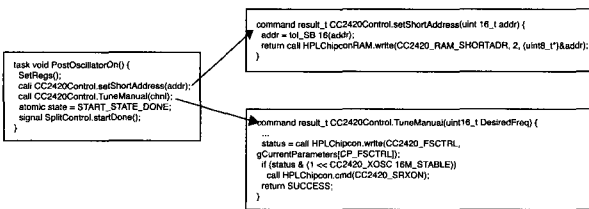
위의 소스 코드는 CC2420 RF module을 가장 낮은 레벨에서 초기화하는 루틴이다. TOSH_MAKE_

MISO_INPUT(), TOSH_MAKE_MOSI_OUTPUT()은 CC2420 칩이 사용하는 데이터 SPI 편 방향을 설정해 주고 TOSH_MAKE_SPL_SCK_OUTPUT()은 MCU와 CC2420간 클럭을 동기화 시켜 준다. TOSH_MAKE_CC_RSTN_OUTPUT()은 CC2420의 리셋 핀의 방향을 설정하고, TOSH_MAKE_CC_VREN_OUTPUT()은 CC2420의 전원 인가를 하게 되며 TOSH_MAKE_CC_CS_OUTPUT()은 CC2420을 enable 시키게 됨으로서 실질적으로 CC2420 칩은 이 함수이후부터 동작하게 된다. TOSH_MAKE_CC_FIFO1_INPUT()은 CC2420의 FIFO 핀 방향을 설정해 주고 TOSH_MAKE_CC_CCA_INPUT()은 CC2420이 하드웨어로 제공해주는 CCA(Channel Clear Assessment)핀 방향을 설정한다. TOSH_MAKE_CC_SFD_INPUT(), TOSH_MAKE_CC_FIFO_INPUT()은 각각 SFD 핀과 FIFO 핀 방향을 설정한다. sbi(SPSR, SPI2X)은 SPI 클럭을 2배로 설정하고 sbi(SPCR, MSTR)은 CC2420을 마스터 모드로 설정하며, cbi(SPCR, CPOL), cbi(SPCR, CPHA); CC2420의 기본 설정이다. cbi(SPCR, SPR1), cbi(SPCR, SPR0)은 CC2420의 클럭을 설정을 하고 sbi(SPCR, SPE)은 SPI핀의 변화를 감지할 수 있게 하기 위해서 CC2420 SPI를 enable 시킨다.

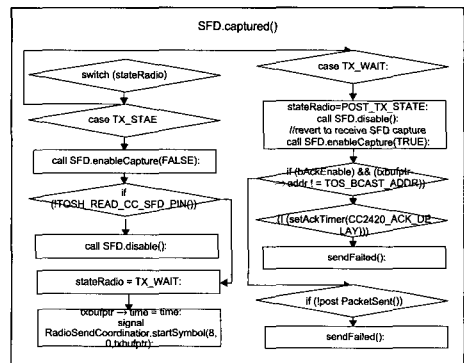
드레스 즉 노드 주소를 칩에 설정하는 것을 도식화한 것이다. CC2420Control.setShortAddress()은 addr을 인자로 CC2420에 주소를 설정하는 부분이다. CC2420Control.TuneManual() 함수는 chnl변수 값에 따라 주파수 설정을 한다.



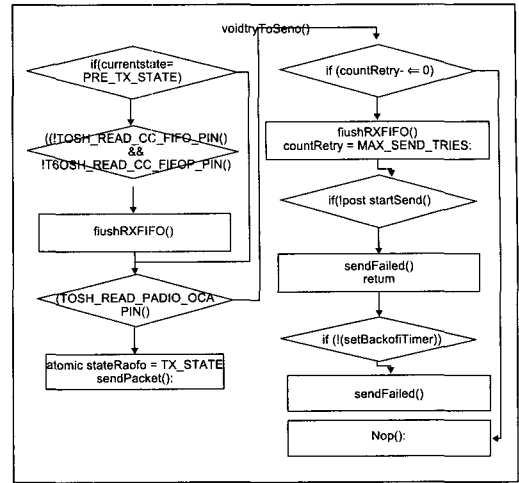
위 루틴은 CC2420을 초기화하는 루틴 중 하드웨어 초기화 부분과 RF 전송 전력을 설정하는 부분이다. CC242ControlM은 CC2420의 관련된 레지스터 설정 및 제어를 하기 위한 컴포넌트이다. CC2420ControlM에서 SplitControl.init() 함수가 호출하는 HPLChipconControl.init()에서 기본적인 하드웨어 초기화 즉 SPI 핀 방향 설정, MISO, MOSI 핀 방향 설정 등등을 수행하고 CC242Control.SetRFPower() 함수는 전송 전력을 설정해준다. 이 전송 전력 값을 설정함으로써, RF range를 늘릴 수도 있고 줄일 수도 있으며, 저전력 기능도 지원할 수 있다.



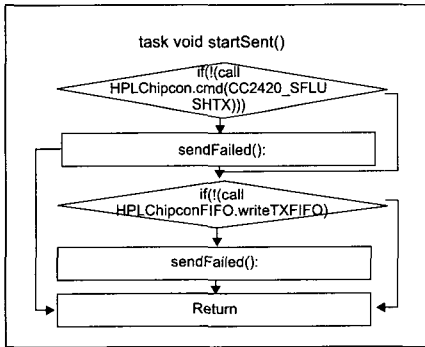
위 루틴은 TinyOS의 CC2420ControlM.nc의 일부분을 나타내며, CC2420의 주파수 설정과 로컬 어



위는 SFD 함수의 순서도이며 이 함수의 기능은 SFD핀의 변화를 감지하여 패킷 전송이 끝나는 시기를 결정한다. 우선 SFD핀의 레벨이 낮아지기를 기다리며 패킷 전송이 끝날지 기다린다. 만약 핀이 계속 높은 전압레벨을 유지한다면, 빠져나와 다른 인터럽트를 기다린다. 만약 핀의 전압레벨이 떨어지면, 다른 노드의 패킷 전송이 끝날 때까지 상태를 TX_WAIT로 유지한다. 다른 노드에서의 패킷 전송이 종료되면 다시 SFD핀이 처음 상태로 된다. 만약 ACK enable 상태이면 패킷을 unicast 방식으로 전송하고 ACK를 기다린다. 또한 ACK disable 상태나 broadcast 모드인 경우 packet senddone 이벤트 task를 생성하며 RX SFD handler 인터럽트가 발생한다.



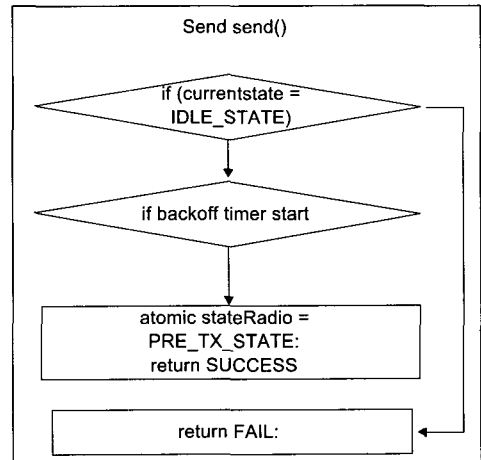
는 경우에는 라디오상태가 불량하다 판단하고 RXFIFO를 플러시한 후 나중에 다시 패킷 전송을 기약하고 일단 복귀하여 일반적인 동작을 수행한다.



패킷 데이터를 CC2420의 TXFIFO로 전송한다. 만약 성공하지 못하면 SendFailed() 함수를 호출한다. SendFailed() 함수는 사용한 버퍼를 초기화 시켜주며 상위 레벨의 컴포넌트에게 알려준다.

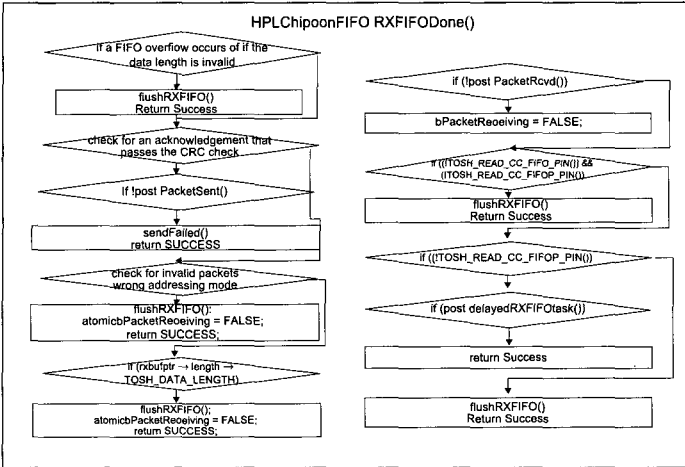
채널의 clear 상태를 확인하여, 채널이 사용 중이 아니고 CCA가 좋은 상태에서 FIFO overflow가 발생하지 않고 데이터 길이가 정확하다면, RXFIFO를 플러시하고 sendPacket() 함수를 이용하여 패킷 전송을 시도한다.

만약 지정된 최대 횟수만큼 패킷 재전송을 시도하



이 함수의 기능은 우선 채널의 상태를 확인하여 clear 상태일 경우 backoff 타이머를 동작시켜 충돌 회피(Collision Avoidance) 기능을 수행한다. 실제 동작 할 경우 주소를 설정하고 패킷을 바이트 단위로

전송한다.



위의 함수는 RXFIFO로부터 수신버퍼에 데이터가 도착한 경우 수행되는데, 상위 영역으로 시그널이나 태스크를 생성한다. 어떤 경우에는 전송 에러는 발생하지 않았지만, 패킷의 크기가 패킷 헤더와 다르게 들어올 수 있다. 패킷의 크기가 헤더에 명시된 것보다 클 경우 헤더에 명시된 패킷 길이보다 큰 부분은 잘라내고 사용한다. 그리고 만약 FIFO overflow 인터럽트가 발생하거나 패킷 크기를 신뢰할 수 없다면, RXFIFO를 플러시하고 정상 모드로 복귀한다. ACK와 CRC를 함께 체크하여 신뢰할 수 있는 패킷을 판정하고, 만약 패킷 크기가 작으면 CRC를 앞으로 움직여서 패킷을 사용가능하게 만들며, RSSI와 LQI를 분리한다. 즉 패킷의 전송 에러를 처리하는 기능이 포함되어 있다.

IV. 결 론

임베디드 시스템은 최근 유비쿼터스 컴퓨팅 분야의 핵심 기술로 점점 중요성이 높아지고 있다. 특히

임베디드 무선통신 시스템 기술은 유비쿼터스 센서 네트워크에서 대표적으로 채택되고 있다. 현재 무선

통신 관련하여 내장된 소프트웨어의 대표적인 주요 기능은 우선 무선통신용 하드웨어 칩을 초기화한다. 이때 초기화 과정에서 RF 무선통신 주파수나 전력세기를 지정한다. 또한 상위 MAC 계층과 인터페이스를 수행하여, 물리계층에서 통신 하드웨어 구동을 통해 패킷을 실제 전송하거나 수신한다. 본 고에서는 대표적인 센서 네트워크 하드웨어 및 소프트웨어 플랫폼 하에서 실제 어떻게 임베디드 무선통신 기술이 적용되고

있는지 구체적으로 알아보았다. 즉 TinyOS에서 CC2420 RF 통신 칩을 통하여 어떻게 무선통신 기술이 임베디드 시스템으로 구현되고 있는지 상세하게 정리해 보았다.

[참 고 문 헌]

- [1] E. Royer and C.K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," IEEE Personal Communications, Vol. 6, No. 2, pp. 46-55, April 1999
- [2] C. Perkins, E. Belding-Royer, S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing", IETF RFC 3561, July 2003
- [3] C.R. Lin and J. Liu, "QoS Routing in Ad Hoc Wireless Networks," IEEE JSAC, Vol. 17, No. 8, pp. 1426-1438, August 1999

- [4] W. Liao, Y. Tseng, K. Shih, "A TDMA-based Bandwidth Reservation Protocol for QoS Routing in a Wireless Mobile Ad Hoc Network," IEEE ICC, pp. 3186-3190, April 2002
- [5] Q. Xue and A. Ganz, "Ad Hoc QoS On-Demand Routing in Mobile Ad Hoc Network," Journal of Parallel and Distributed Computing, Vol. 63, pp. 154-165, February 2003
- [6] S. Lee and A. Campbell, "INSIGNIA: In-Band Signaling Support for QoS in Mobile Ad Hoc Networks," Proceedings of International Workshop on Mobile Multimedia Communications (MoMuC), October 1998
- [7] H. Xiao, W. Seah, A. Lo, K. Chua, "A Flexible Quality of Service Model for Mobile Ad-Hoc Networks," IEEE VTC, May 2000
- [8] M. Mirhakkak, N. Schult, D. Thomson, "Dynamic Quality-of-Service for Mobile Ad Hoc Networks," ACM MobiHoc, pp. 137-138, 2000
- [9] C. Perkins, E. Royer, S. Das, "Quality of Service for Ad hoc On-Demand Distance Vector Routing," IETF Internet Draft, July 2000
- [10] R. Sivakumar, P. Sinha, V. Bharghavan, "CEDAR: A Core-Extraction Distributed Ad Hoc Routing Algorithm," IEEE JSAC, Vol. 17, No. 8, pp. 1454-1466, August 1999



조인휘

1983년 한양대학교 전자공학과 학사
1994년 University of Arizona 전기컴퓨터공학과 석사
1998년 Georgia Tech 전기컴퓨터공학과 박사
1985년 ~ 1992년 데이콤 종합연구소 연구원
1998년 ~ 2000년 Oak Ridge 국립연구소 연구원
2000년 ~ 2002년 Bellcore Lab 연구원

2002년 ~ 현재 한양대학교 정보통신학부 조교수

관심분야 : 센서 네트워크, 이동통신, RFID, 이동성 관리, 멀티미디어 통신
