



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

저전력 소모를 위한 무선센서 네트워크
MAC 프로토콜 성능 개선

Performance Improvement of Wireless Sensor Network
MAC Protocols for Low-Power Consumption

2010년 2월 25일

조선대학교 대학원

컴퓨터공학과

정 문 수

저전력 소모를 위한 무선센서 네트워크
MAC 프로토콜 성능 개선

지도교수 이 준

이 논문을 컴퓨터공학 석사학위신청 논문으로 제출함


2009년 10월


조선대학교 대학원


컴퓨터공학과

정 문 수

정문수의 석사학위 논문을 인준함

심사 위원장 조선대학교 교수 김충원 

심사 위원 조선대학교 교수 김판구 

심사 위원 조선대학교 교수 이준 

2009년 11월

조선대학교 대학원

목 차

ABSTRACT	i
제1장 서 론	1
제2장 관련 연구	3
제1절 IEEE 802.15.4	3
제2절 에너지 효율적인 MAC프로토콜	17
제3장 저전력 소모를 위한 MAC 프로토콜의 제안	23
제1절 제안하는 저전력 기법.....	25
제2절 데이터 변화량의 Threshold의 설정.....	28
제4장 시뮬레이션 및 실험.....	29
제5장 결 론	40
참고문헌	42

그림 목 차

그림2.1	Zigbee 스택 구조	4
그림2.2	Zgibee 네트워크 Topologie	5
그림2.3	LR-WPAN Device 구조	6
그림2.4	Zigbee 주파수 대역별 채널 구조	7
그림2.5	PHY 계층 참조 모델	8
그림2.6	GTS가 있는 Superframe 구조	10
그림2.7	MAC프레임의 기본구조	12
그림2.8	Unslotted CSMA-CA 알고리즘	13
그림2.9	슈퍼프레임 구조	14
그림2.10	패킷 전송을 위한 시간간격(IFS)	16
그림2.11	S-MAC 프로토콜의 기본 동작	19
그림2.12	클러스터가 중첩된 지역에서 동기화를 제공해야하는 연결노드 ..	20
그림2.13	T-MAC 프로토콜의 기본 동작	21
그림2.14	P-MAC 프로토콜의 기본 동작	22
그림4.1	ZigbeX 모트의 구성도	29
그림4.2	Tinyviz 실행화면	31
그림4.3	Cywin에서 컴파일 결과	32
그림4.4	Spectrum analyzer를 이용한 측정 준비	32
그림4.5	프로그램을 mote에 포팅후 Spectrum analyzer 측정준비 ..	33

그림4.6	전송거리가 약 10센티일 때 채널 간 전파 세기	33
그림4.7	전송거리가 약 1m일 때 채널 간 전파세기	34
그림4.8	전송거리가 약 2m일 때 채널 간 전파세기	34
그림4.9	PowerTossim-z 소스 다운로드	35
그림4.10	Trace 파일의 내용	36
그림4.11	Trace 파일을 powertossim-z로 실행한 결과	36
그림4.12	LPMAC의 에너지 소모량	37
그림4.13	SMAC의 에너지 소모량	37
그림4.14	TMAC의 에너지 소모량	37
그림4.15	시간 변화에 따른 LPMAC과 SMAC의 에너지 잔량비교 ..	38
그림4.16	초당 송수신 패킷수별 전체 네트워크에서 소비하는 에너지 비교 ..	39

표 목 차

표 2.1	Zigbee 주파수 대역의 특징	6
표 2.2	각 채널의 백오프 소비시간	13
표 2.3	각 채널의 1심볼 소비시간	15
표 2.4	각 채널의 IFS	16
표 3.1	CC2420이 제공하는 8단계 파워레벨	25
표 4.1	ZigbeX 플랫폼 요약	30
표 4.2	Powertossim-z에서 사용하는 power-model	35

ABSTRACT

Performance Improvement of Wireless Sensor Network MAC Protocols for Low-Power Consumption

Jeong Moon Soo

Advisor : Prof., Lee Joon, Ph.D.

Department of Computer Engineering

Graduate School of Chosun University

In accordance with the emerging ubiquitous computing technology and the men demand, sensor network technology and researched in which various sensor nodes for circumstance understanding are incorporated in the wired and wireless network infrastructure.

In this paper, we proposed a LP-MAC algorithm for efficient energy management of sensor network consisted of sensor nodes that have restricted energy to solve these problem. Proposed algorithm improves energy efficiency by controlling sensing power and pattern apply to the previous packet. We estimated performance of the proposed algorithm and existing SMAC algorithm using ZigbeX mote that Zigbee module of Hanback Electronics Co. support IEEE 802.15.4.

The performance evaluation result shows that energy efficiency is improved by 40% in the 4 packet per second and 1% in the 12 packet per second, 3% in the 25 packet per second more than the SMAC algorithm.

Proposed algorithm derived an upper bound on the energy efficiency for Ubiquitous Computing environment that have various network environment that is with Zigbee technology of IEEE802.15.4 based.

제1장 서론

유비쿼터스 컴퓨팅 환경에서 가장 각광받는 기술 중 하나인 무선 센서 네트워크는 유무선 네트워크 인프라에 상황인지를 위한 센서 디바이스를 결합하여 감지된 환경 데이터를 응용 서비스 서버와 연동하는 기술이다. 센서 노드는 배터리를 사용하는 저전력 저가의 시스템으로 에너지 사용에 제약을 받으며 배터리가 소모되어 노드의 동작이 멈추면 더 이상 네트워크의 수단으로 사용될 수 없다. 따라서 무선 센서 네트워크는 노드에 사용되는 전력의 사용을 효율적으로 하여 네트워크의 생존시간을 최우선으로 생각한다.

무선 통신 프로토콜 중 하나인 Zigbee(IEEE 802.15.4)가 다른 무선 프로토콜보다 적은 전력소모로 데이터 전송이 가능하며 간단한 조작으로 데이터 네트워크를 구축할 수 있기에 센서 네트워크에서 적합하다고 평가되며 국내에서는 2.4GHz의 주파수 대역을 사용하고 있다. 센서 네트워크 MAC Layer에서는 무선 계층의 에너지 소모를 줄이기 위해 가장 에너지 소모가 큰 영역인 Radio에 대한 연구를 진행하고 있다. 센서 노드의 상태를 Sleep과 Active 상태로 구분하여, 센서 노드가 데이터 전송을 하지 않을 경우 Radio에 사용되는 불필요한 전원에 따른 에너지 소비를 줄이는 방안에 대해 연구가 활발히 이루어지고 있다. 하지만 이 방법의 문제점은 수면 주기를 길게 할수록 전력 소모는 줄일 수 있지만, 말단 센서 장치에서 Coordinator라 불리는 중앙 센서 장치로 데이터 전달 속도가 낮아지는 문제는 있고 있다. 예를 들어 칩입 센서나 화재 경보 센서 같은 경우 응답 속도를 늦추게 되면 센서의 전력 소비는 줄어들겠지만, 응답속도가 낮아져서 사건의 대비가 늦어지게 되는 문제점이 생기게 된다.

많은 학교나 연구소, 기업에서 센서 네트워크 장치의 전력 소비를 줄이기 위해서 하드웨어 디자인은 물론 저전력 네트워크 토폴로지 구성, 효율적인 MAC Layer 프로토콜, 네트워크 라우팅 프로토콜 등이 연구 되어 왔다. 특히, 효율적인 MAC Layer 프로토콜 연구에서 S-MAC, T-MAC, P-MAC 등 다양한 연구들이 진행 되

어 왔다. 센서 장치는 전송할 데이터가 없어도 자신에게 전송될 데이터를 기다려야 하기 때문에 채널을 관찰하고 있어야 한다. 비교적 데이터의 전송 빈도가 낮은 센서 장치에서는 불필요한 채널 관찰로 인한 전력 소모가 전체 전력 소모에서 큰 비중을 차지하게 된다. 위의 연구된 효율적인 MAC 프로토콜은 기본적으로 센서 장치가 깨어 있는 시간을 최대한 줄임으로써 전력 소비를 줄이는 방식을 채택하고 있다. 주기적 혹은 동적으로 센서 장치가 수면하게 함으로써 전력 소비를 최대한 줄이는 것이 위 프로토콜의 목표이며 과제이다. 위와 같은 연구로 인하여 센서 장치들의 전력 소모 기법은 날로 발전하였지만, 전력 소모가 줄어드는 만큼 응답속도가 떨어지는 문제점은 그대로 안고 있어야 한다.

주기적으로 센서 장치를 수면시킴으로써 전력 소모를 줄일 수 있는 메커니즘이 있지만 그 메커니즘은 상당히 고정적이기 때문에 다양한 네트워크의 상황에 효과적으로 대응하기에는 부족한 점이 적지 않다. 따라서 그 수면주기를 이전에 수집된 센서 데이터와 비교하여 동적으로 할당하고 센서 장치에서 중앙 데이터 처리 장치로의 데이터 전송 응답 시간을 줄이는 방안을 연구하여야 하는 필요성이 있다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 센서 네트워크에서 주로 사용하는 Zigbee의 표준인 IEEE802.15.4와 에너지 효율적인 MAC 프로토콜 기법에 대해서 몇 가지 소개하며, 그에 따른 동작을 설명한다. 3장에서는 주변노드의 위치에 따른 RSSI값 제어와 동적 수면주기 할당으로 에너지 효율성을 개선한 프로토콜을 규정한다. 4장에서 시뮬레이션 및 실험을 진행함으로써 제시한 프로토콜이 기존의 방안에 비해 성능향상과 개선이 되었음을 증명하고 평가한다. 마지막으로 5장에서 결론을 제시한다.

제2장 관련 연구

제1절 IEEE 802.15.4

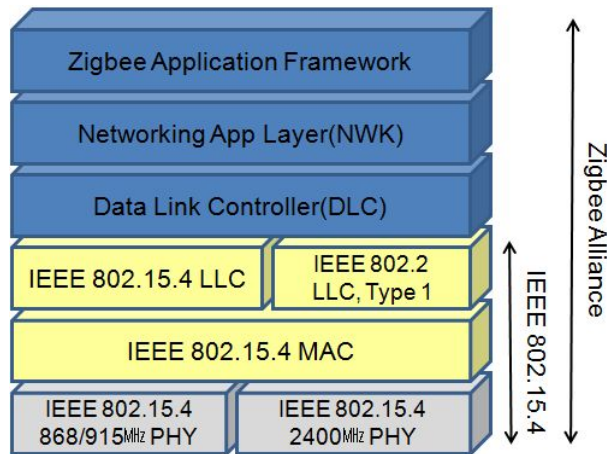
1. ZIGBEE 구조

IEEE 802.15.4는 제한된 전력과 낮은 Throughput을 요구하는 Application의 무선 접속을 위한 단순하고, 저가의 통신망 기술이다. IEEE 802.15.4 기술은 무선 PAN의 PHY 계층과 MAC계층 기반으로 하여 상위의 네트워크 계층에서 응용서비스까지는 응용분야의 환경에 따라 비영리 조직인 Zigbee Alliance에서 개발되고 있으며, 이 조직에서는 다양한 응용분야에서 활용될 수 있도록 응용 프로파일의 정의 및 개발에 초점을 맞추고 있다. IEEE 802.15.4는 물리 계층과 데이터 링크 계층에 대한 표준화를 담당하고 있다.

어플리케이션 계층은 각 응용분야에 맞는 어플리케이션 오브젝트들이 모여서 Application Framework를 구성하게 되는데, 사용될 수 있는 어플리케이션으로는 홈 네트워킹이나 의료 장비 관리 산업자동화와 같은 분야들이 될 수 있다. ZDO(Zigbee Device Object)에서는 어플리케이션 오브젝트가 요청할 경우 네트워크 내의 해당 서비스를 가지고 있는 다른 디바이스의 주소를 알려주는 기능과 네트워크에 있는 디바이스들 간의 복잡한 어플리케이션 계층 간의 연결을 시켜주는 기능을 한다.

네트워크 계층은 IEEE 802.15.4 표준에서 정의한 MAC 계층의 안정적인 동작을 지원하며, 상위 응용계층과의 연결을 시켜준다. 여기서 어플리케이션 계층과의 연결을 위하여 2가지 서비스 엔티티를 제공하는데, 데이터 서비스와 매니지먼트 서비스가 그것이다. 데이터 서비스 엔티티의 기능을 응용계층에서 내려오는 데이터 유닛을 네트워크 레벨에 맞는 데이터 유닛으로 변환하여 주거나, 네트워크 내에 있는

응용 계층간에 데이터 전송이 가능하도록 지원하는 기능을 한다. 매니지먼트 서비스 엔티티는 디바이스가 코디네이터나 라우터일 경우 네트워크에 연결된 다른 장비들의 번지를 지정해 주거나, 장비들이 네트워크에 연결되거나 끊어질 수 있도록 네트워킹과 관련된 기능을 지원한다. 또한 다른 장비들과 통신을 위한 경로를 찾거나, 1-호핑을 위하여 이웃한 디바이스 검색을 하는 기능이 포함되어 있다.



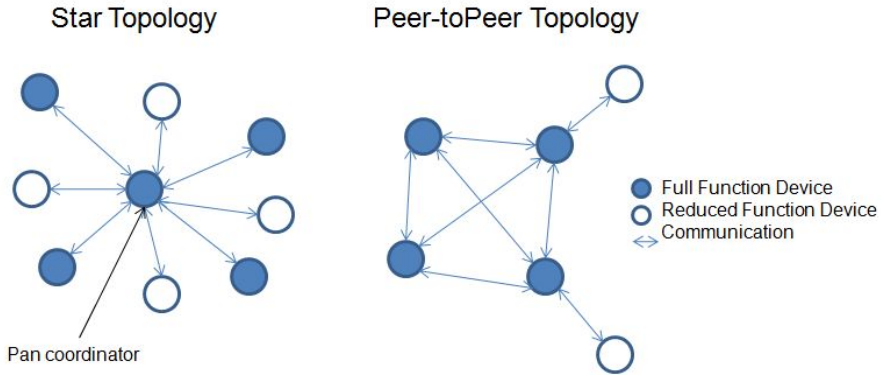
[그림 2.1] Zigbee 스택 구조

2. 네트워크 및 DEVICE 구조

가. 네트워크 TOPOLOGIES

IEEE 802.15.4는 [그림 2.2]와 같이 STAR형과 PEER-TO-PEER 망 구조를 지원한다.

STAR형 망 구조에서 device와 PAN coordinator 사이에 통신을 위한 링크가 만들어지며 device는 통신의 시작점 또는 종단점이 된다. PAN coordinator는 PAN의 주요 controller이며 main power가 공급되어야 한다. STAR형 망 구조는 Home Automation, PC주변기기 연결, 장난감과 게임 그리고 Health care 와 같은 서비스에 이용될 수 있다.



[그림 2.2] ZIGBEE 네트워크 TOPOLOGIES

PEER-TO-PEER형 망 구조는 device가 자신의 영역 안에 있는 다른 device와 통신할 수 있다는 점에서 STAR형 망 구조와 다르다. 산업 공정 제어 및 모니터링, 무선 센서 네트워크, 물류에서의 재고 트래킹과 같은 서비스에 PEER-TO-PEER형 망 구조가 사용될 수 있다.

나. DEVICE 구조

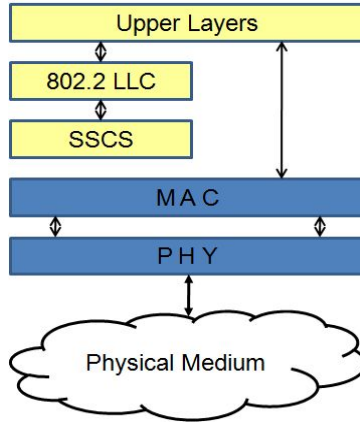
Datalink 계층의 LLC 계층은 IEEE 802.2를 사용하고 있으며 이를 802.15.4의 MAC sublayer와 연결하기 위해 Service Specific Convergence Sublayer (SSCS)를 이용하고 있다. 또한 802.2 LLC를 사용하지 않고 다른 LLC를 사용할 수 있는 구조 역시 허용되어 있다.

3. IEEE 802.15.4 의 물리계층

물리계층은 PHY DATA 서비스와 PHY Management 서비스의 두 가지 서비스를 제공하며 다음과 같은 역할을 담당한다.

- 무선 Transceiver의 활성화/ 비활성화
- 현재 채널내의 에너지 감지 (Energy Detection)
- 수신 패킷의 링크 품질 표시 (Link Quality Indication)

- 채널 주파수 선택
- 데이터 전송 / 수신



[그림 2.3] LR-WPAN DEVICE 구조

가. 일반적인 요구사항

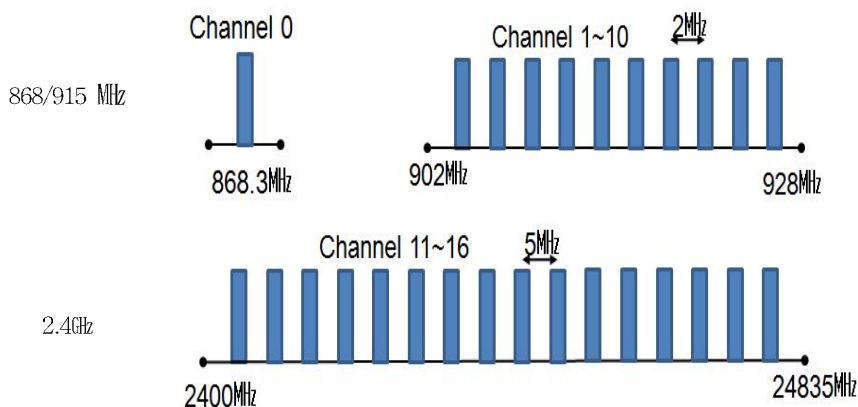
IEEE 802.15.4는 두 가지 물리계층(Multiband, Multirate)을 지원한다. 두 물리계층의 근본적인 차이는 주파수 대역에 있으며 ISM 대역인 2.4GHz 대역과 유럽, 미국에서 사용하는 868/915MHz 대역이며 유럽에서는 868MHz 대역을 사용하며 미국에서는 915MHz 대역을 사용한다.

[표 2.1] ZIGBEE 주파수 대역의 특징

PHY (MHz)	FREQUENCY BAND(MHz)	SPREADING PARAMETERS		DATA PARAMETERS		
		CHIP RATE (K CHIP/S)	MODULATION	BIT RATES	SYMBOL RATES	SYMBOLS
868/915	868-868.6	300	BPSK	20	20	BINARY
	902-928	600	BPSK	40	40	BINARY
2400	2400-2483.5	2000	O-QPSK	250	62.5	16-ary quasi Orthogonal

나. 채널화

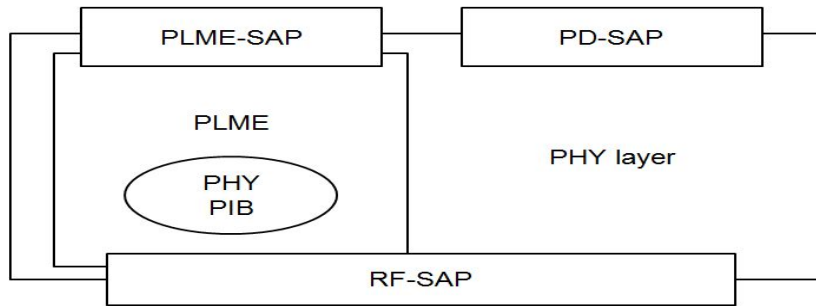
IEEE 802.15.4는 [그림 2.4]와 같이 3개의 주파수 대역에서 27개의 채널을 가지고 있다. 868/915MHz 물리계층은 868.0MHz와 868.6MHz 사이의 대역에서 하나의 채널과 902.0MHz와 928MHz대역에서 10개의 채널을 제공한다. 2.4GHz 물리계층은 2.4GHz와 2.4835GHz 사이의 대역에서 5MHz 간격으로 16개의 채널을 제공한다.



[그림 2.4] ZIGBEE 주파수 대역별 채널 구조

다. PHY 계층 서비스 규격

PHY 계층은 RF Firmware나 RF Hardware를 이용하여 MAC Sublayer와 무선 채널 사이의 인터페이스를 제공한다. PHY 계층은 개념적으로 PHY Layer Management Entity (PLME) 라는 Management Entity를 포함하고 있다. 이 PLME는 계층관리 서비스 (Layer Management Service)의 인터페이스 제공과 Database(PIB : PHY Layer Pan Information Base)를 관리하는 역할을 한다. [그림 2.5]는 PHY계층의 구성요소와 인터페이스를 나타내고 있다.



[그림 2.5] PHY 계층 참조 모델

PHY 계층은 두 가지 Service Access Service access Point(SAPS)을 통해 두 가지 서비스를 제공한다. 이는 PHY Data SAP(PD-SAP)을 통한 PHY 데이터 서비스와 PHY Layer Management Entity SAP(PLME-SAP)을 통한 PHY 관리 서비스이다. PHY 데이터 서비스는 통신하는 양측의 MAC Sublayer Entity들의 MAC Protocol Data Units(MPDUS)의 전송을 지원한다. 그리고 PHY 계층 관리 서비스는 MLME와 PLME 사이의 관리 명령어들의 전송을 담당한다.

4. IEEE 802.15.4의 MAC계층

가. MAC의 기능

Coordinator가 PAN을 시작하기 위해서는 에너지 탐지, 채널 스캔과 같은 과정이 필요하며 Device 가 PAN Coordinator에게 Association과 Disassociation하는 과정이 수행된다. MAC의 주요 기능은 다음과 같다.

(1) ENERGY DETECTION

FFD가 각 채널의 에너지를 측정하는 것이며 PAN Coordinator가 새로운 PAN을 구성하기에 앞서 채널을 선택하기 위해 사용한다.

(2) CHANNEL SCAN

① Active Channel Scan

FFD가 POS 내의 Beacon프레임을 전송하는 어떤 Coordinator를 찾는 것이다. PAN Coordinator가 새로운 PAN을 구성하기 전에 PAN Identifier를 선택하거나

Device 가 Association 하기 전에 사용된다.

② PASSIVE CHANNEL SCAN

Active Scan 과 같이 Device가 POS 내의 Beacon 프레임 전송하는 Coordinator를 찾는 것이지만 Beacon Request 명령을 보내지 않는다는 점에서 다르다. 이것은 Device에 의해 Association 전에 사용된다.

③ ORPHAN CHANNEL SCAN

Device 가 Coordinator와 동기를 잃었을 때 자신의 Coordinator를 다시 찾으려고 시도하는 것이다. Device가 Orphan Notification명령을 보내면 Coordinator Device 리스트를 검색하여 해당 Device를 발견하면 Coordinator Realignment 명령을 보낸다.

(3) STARTING PAN

FFD에 의해 Active Channel Scan이 끝나고 PAN Identifier Selection이 완료된 후에 PAN이 시작된다.

(4) DEVICE DISCOVERY

FFD는 다른 Device에게 Beacon프레임을 전송함으로써 PAN내의 자신이 존재함을 알린다. PAN Coordinator가 아닌 FFD는 PAN에 성공적으로 Association을 했을 때에만 Beacon 프레임을 전송할 수 있다.

(5) ASSOCIATION

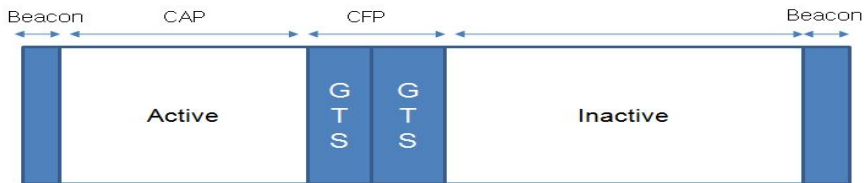
Device는 MAC Sublayer Reset을 수행한 후에 혹은 Active Scan이나 Passive Scan으로 적절한 PAN을 찾은 뒤에 Association을 수행한다. Device의 Association Request를 받은 Coordinator는 자원이 충분할 경우 Device에게 새로운 Address와 Association의 성공을 나타내는 상태가 포함된 응답 명령을 전송하며 자원이 없을 경우 Association의 실패를 의미하는 상태가 포함된 응답 명령을 보낸다.

(6) DISASSOCIATION

Coordinator가 Associate된 Device가 PAN을 떠나기를 바라거나 Device 가 PAN을 떠날 때 Disassociation Notification명령과 이에 대한 Acknowledgement를 교환함으로써 Disassociation과정을 수행한다.

(7) GTS 할당

Gauranteed Time Slot(GTS)는 Device Superframe의 일부분 내에서 채널을 독점적으로 사용할 수 있도록 해준다. GTS는 PAN Coordinator에 의해 할당되며 PAN Coordinator와 Device 사이의 통신에만 사용된다. PAN Coordinator는 한 번에 최대 7개의 GTS를 할당할 수 있으며 GTS는 여러 Superframe에 걸쳐 할당될 수 있다. GTS는 Device의 GTS Request를 받으면 Coordinator에 의해 SuperFrame의 가능한 Capacity에 따라 결정된다. GTS는 FIRST-COME-FIRST-SERVED 방식으로 할당되며 [그림 2.6]과 같이 모든 GTS는 CAP 구간이 끝난 다음에 연속적으로 위치한다.



[그림 2.6] GTS가 있는 SUPERFRAME 구조

GTS가 더 이상 필요하지 않으면 PAN Coordinator에 의해 혹은 Device의 요청에 의해 해제된다. GTS관리는 PAN Coordinator에 의해서만 수행된다. 이를 위해 PAN Coordinator는 7개의 GTS를 관리하기 위한 모든 정보를 가지고 있어야 하며 각 GTS에 대해 Starting Slot, 길이, 방향 그리고 Associated Device Address의 정보를 유지해야 한다. 여기에서 GTS 방향은 Device의 데이터 흐름에 관한 것이며 송신, 수신으로 표현된다. Device Address와 GTS 방향이 각 GTS를 구분하는 구분자로써 사용된다.

나. 프레임 구조

MAC계층에서 사용되는 패킷은 최대 127바이트의 길이를 가지며, 패킷의 종류는 사용되는 목적에 따라 4가지 종류로 구분되는데, 비콘 프레임, 데이터 프레임, 에크 프레임, 커맨드 프레임이 그것이다. 모든 프레임의 기본적인 구조는 [그림 2.7]과 같다. 프레임의 제일 앞에는 MHR이 있으며 그 뒤에는 프레임의 실제 내용인 페이

로드, 그리고 마지막으로 MFR이 있다.

MHR영역에는 프레임 컨트롤과 시퀀스 넘버, 번지 영역이 있다. 프레임 컨트롤에는 패킷 종류에 대한 정보, 각 송신 측과 수신 측에서 16비트 번지와 64비트 번지 중 어떠한 것을 사용하는지 나타내게 된다. 또한 ACK프레임을 사용 할지에 대한 정보도 포함되어 있다. 다음으로 1바이트 시퀀스넘버가 있으며, 플로우 컨트롤을 위하여 사용된다. 마지막으로 번지 영역에는 송신측과 수신측의 번지를 나타낸다.

패킷에 사용되는 주소는 16비트 숏 어드레스와 64비트 롱 어드레스가 사용된다. 각 통신 장비들은 고유의 64비트 주소 값을 가지고 있으며, 디바이스가 네트워크에 연결된 후에는 코디네이터에 의하여 16비트 주소 값을 할당 받게 된다. 그럼으로 64비트 주소 값은 처음 네트워크에 연결될 때에만 사용된다. 16비트 주소의 초기 값은 0xFFFF이며 이는 자신이 어떠한 네트워크에도 연결이 되어있지 않다는 의미이다. 통신 장비가 네트워크에 연결될 경우에 16비트 주소 이외에도 PANID라는 값을 얻게 되는데 이는 각 네트워크의 고유 아이디이다. PANID의 초기 값은 0xFFFF이며 네트워크에 연결된 후에는 코디네이터에 의하여 PAN의 고유 주소 값을 받게 된다.

패킷을 수신할 경우에는 상대방이 자신에게 보낸 패킷인지 확인하는 작업이 필요하다. 코디네이터에서 엔드 디바이스로 브로드캐스트 할 경우에는 수신 측 주소 값이 0xFFFF를 가지게 된다. 그럼으로, 수신된 패킷의 목적지 주소 값을 확인하여 자신의 주소 값과 일치하거나 0xFFFF의 값을 가지고 있을 경우에만 패킷을 상위 계층으로 전달하게 된다. 그 이외의 경우에는 패킷을 무시하게 된다. 물리계층에서 수신된 패킷을 필터링 하는 이러한 기능은 수신필터에서 처리된다.

MFR에는 패킷의 에러를 확인하기 위한 FCS가 있는데 프레임의 CRC값이 여기에 들어오게 된다. CRC수식은 식 (2.1)으로 계산된다.

$$G_{16}(x)=x^{16}+x^{12}+x^5+1 \dots\dots\dots(2.1)$$

Frame Control	Sequence number	Destination PAN ID	Destination address	Source PAN ID	Source address	Fram payload	FCS
		Addressing fields					
MHR						MAC payload	MFR

[그림 2.7] MAC프레임의 기본 구조

다. 프로토콜

무선 통신에서의 가장 기본적인 통신 방식으로 TDMA방식과 CSMA-CA방식을 들 수 있다. 저전력의 통신을 위하여서는 TDMA방식을 사용하는 것이 효율적이다. 이 방식은 데이터 전송 시에 다른 디바이스의 데이터와 채널 상에서 충돌이 일어나지 않아서 패킷의 재전송에 의한 에너지 손실이 없기 때문이다. 하지만, 네트워크에 연결된 장비가 늘거나 줄어들 경우 통신을 위한 시간 스케줄링을 다시 하여야 하며, 네트워크에 있는 모든 통신 장비에 통신 스케줄을 알려야 하는 번거로운 작업이 필요하다. CSMA-CA방식은 이와는 반대로 네트워크의 변화에 능동적으로 대처할 수 있다는 장점이 있지만, 충돌이 발생할 경우 재 전송을 해야 하기 때문에 에너지 효율성 면에서는 TDMA방식에 비하여 떨어진다.

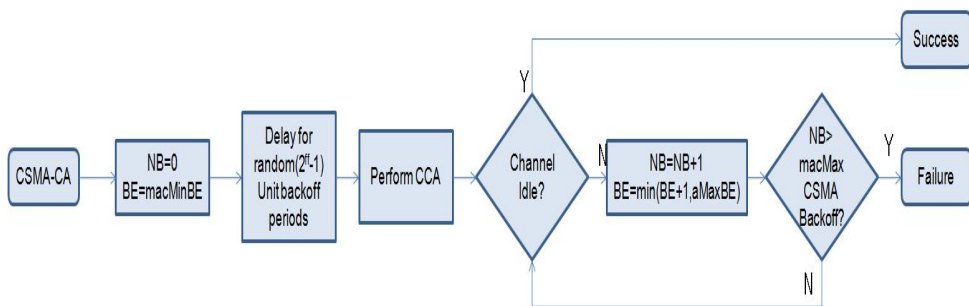
IEEE 802.15.4에서는 네트워크의 상태가 수시로 변하기 때문에 CSMA-CA방식을 사용하지만, 에너지의 소비를 줄여야 하는 문제 때문에 TDMA방식과 CSMA-CA방식을 접목시킨 슈퍼프레임 방식도 같이 지원한다. 또한 센서 네트워크에서는 채널 트래픽이 적기 때문에 CSMA-CA방식에서 RTS, CTS를 사용하지 않아 일반적인 CSMA-CA방식보다 에너지 소비율이 적다. IEEE 802.15.4에서는 슈퍼프레임을 사용하지 않을 경우에는 Unslotted CSMA-CA방식이라 하며, 슈퍼프레임에서의 CSMA-CA를 Slotted CSMA-CA방식이라 한다.

CAMS-CA는 채널에서 데이터의 충돌을 회피하기 위하여 채널 접근에 실패 하였을 경우에 일정 시간을 대기하게 되는데, 이 대기하는 시간을 백오프라 한다. 백오프 시간은 IEEE 802.15.4표준에서의 aUnitBackoffPeroid 값을 기준으로 하여, 채널 접근이 실패할 때마다 백오프는 증가하게 된다. aUnitBackoffPeroid의 값은 20 심볼이며, 각 채널의 aUnitBackoffPeroid의 소비 시간은 [표 2.3]과 관련하여 뒤에

설명하기로 한다. 각 채널의 백오프 시간을 계산하여 정리하면 [표 2.2]와 같다. 표에서 macMinBE는 [그림 2.8]과 같이 BE값을 결정하는 값이다. 이 값에 의하여 몇 개의 aUnitBackoffPeriod만큼 지연을 할지 결정이 된다. Slotted CSMA-CA 방식에서는 에너지 소비를 줄이기 위하여 Battery life Extension모드를 지원하며, 이때에는 macMinBE의 최대값은 2보다 클 수 없다.

[표 2.2] 각 채널의 백오프 소비시간

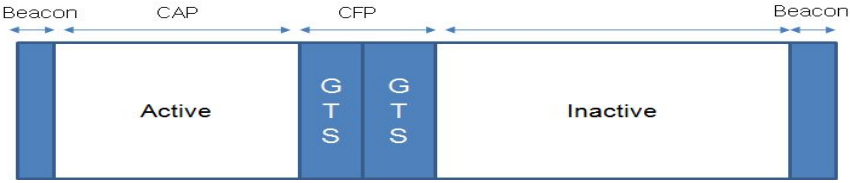
macMinBE	1	2	3	4	5
max Backoff period	1ms	3ms	7ms	15ms	31ms
800MHz	1ms	3ms	7ms	15ms	31ms
900MHz	0.5ms	1.5ms	3.5ms	7.5ms	15.5ms
2.4GHz	320 μ s	960 μ s	2.24ms	4.8ms	9.920ms



[그림 2.8] UNSLOTTED CSMA-CA 알고리즘

슈퍼프레임은 앞에서 언급하였던 비콘프레임을 기준으로 동기화 된다. 슈퍼 프레임에는 크게 활성 구간과 비활성 구간으로 나누어진다. 활성 구간에는 총 16개의 슬롯으로 나누어진다. 활성 구간에서만 디바이스 간에 데이터를 전송할 수 있으며, 비활성 구간에서는 모든 디바이스들은 데이터 전송을 하지 않는다. 활성 구간에는 CAP와 CFP로 나누어진다. CAP에서는 슬롯을 기준으로 CSMA-CA를 시작하기

때문에 Slotted CSMA-CA 전송 방식이라고 한다. CFP구간에는 여러개의 GTS구간이 있다. 이 구간은 특정한 디바이스들에게 할당이 되어 있는 구간으로서, 코디네이터로부터 GTS를 할당받은 디바이스들은 이 구간에서 TDMA 방식처럼 CSMA-CA 알고리즘을 사용하지 않고 바로 채널을 사용할 수 있다. 슈퍼프레임의 구조는 비콘 프레임에 정의되어 있으며, 코디네이터가 슈퍼프레임의 구조를 정하면, 각 디바이스들은 비콘 프레임의 내용을 보고 채널을 접근하게 된다.



[그림 2.9] 슈퍼프레임 구조

센서 네트워크에서는 데이터의 전달이 대부분 디바이스에서 코디네이터로 전송이 되게 된다. 이러한 특징을 이용하여, 디바이스 측에서는 코디네이터에서 데이터가 올 것을 항상 대기 하지 않고, 코디네이터로 데이터를 전송할 때에만 활성화 시키며, 그 이외의 경우에는 아이들 모드에서 에너지를 절약하게 된다. 코디네이터에서 데이터를 엔드 디바이스로 보낼 경우에는 슈퍼프레임 방식일 때와, Unslotted CSMA-CA 방식일 때 각각 다르다. 우선 슈퍼프레임 방식에서는 코디네이터에서 비콘 프레임을 통하여 전송할 데이터가 있음을 나타낸다. 엔드 디바이스들은 수신된 비콘 프레임의 내용을 디코딩하여, 코디네이터가 전송할 데이터가 있음을 확인한다. 만일 전송할 데이터가 있을 경우, 엔드 디바이스는 비콘 프레임에 데이터를 요청한다. Unslotted CSMA-CA 방식에서는 일정 주기로 폴링하면서 코디네이터에게 데이터를 요청한다. 디바이스에서 코디네이터로 데이터를 전송 방식을 직접 전송 방식이라고 하며, 그 반대의 경우를 간접 전송 방식이라고 한다.

ACK 프레임을 사용한 통신에서는 패킷을 전송 후에는 t_{ack} 시간까지 ACK 프레임을 수신 하여야 한다. 이 시간이 지난 후에도 ACK 신호가 들어오지 않았을 경우에는 전송을 실패한 것으로 간주된다. t_{ack} 의 시간은 표준에 의하여 식 (2.2)의 범위

를 가지게 된다. 각 채널의 t_{ack} 의 소비시간은 $aTurnaroundTime$ 과 $aUnitBackoffPeriod$ 의 값에 의하여 정하여지며, $aTurnaroundTime$ 의 의미는 물리 계층에서의 Tx와 Rx의 스위칭 하는데 소비되는 최대한의 시간 간격이며, $aUnitBackoffPeriod$ 의 의미는 CSMA-CA알고리즘에서 Backoff하는 기본 시간 단위이다. 각각의 소비 심볼은 12심볼과 20심볼이며, 각 채널의 심볼 소비 시간은 [표 2.3]과 같다. t_{ack} 의 시간 간격은 [표 2.3]의 값들에 의하여 [표 2.4]의 범위를 가지게 된다.

$$aTurnaroundTime \leq t_{ack} \leq (aTurnaroundTime + aUnitBackoffPeriod) \quad \cdot\cdot(2.2)$$

[표 2.3] 각 채널의 1심볼 소비시간

Band	1 Symbol rate (Ksymbols/s)	1 Symbol time (μs)	aTurnaround Time(μs)	aUnitBackoff Period(ms)
800	20	50	600	1
900	40	25	300	0.5
2.4	62.5	16	192	320

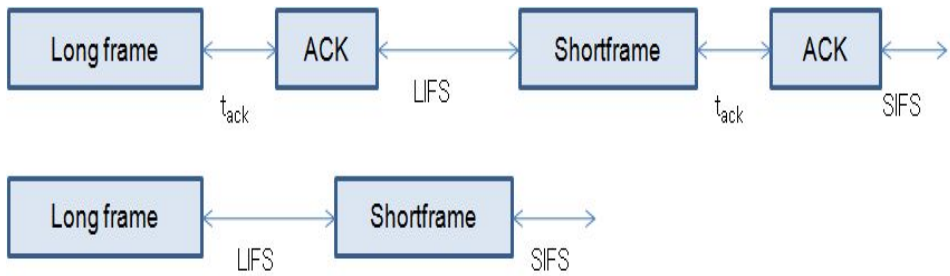
패킷을 연속으로 전송하였을 경우 각 패킷을 연속적으로 보내면, 수신측에서 데이터를 처리할 시간이 부족할 수 있다. 이를 방지하기 위하여 표준에는 패킷을 전송한 후에 일정 시간 동안 기다리도록 되어있다. 패킷의 길이에 따라 긴 프레임(Long frame)과 짧은 프레임(Short frame)으로 나누어지며 그 기준은 식 (2.3), (2.4)와 같다. $aMaxSIFSFrameSize$ 는 18심볼의 길이를 가진다. 긴 프레임과 짧은 프레임에는 패킷 처리를 위하여 각각 LIFS와 SIFS의 시간적 여유를 가진다. 각 시간은 표준에서 정의된 $aMinLIFSPeriod$ 와 $aMinSIFPeriod$ 의 시간을 가지게 되는데, $aMinLIFSPeriod$ 와 $aMinSIFPeriod$ 는 각각 12심볼과 40심볼의 시간을 의미한다. 각 채널의 IFS를 계산하여 정리하면 [표 2.4]와 같다.

$$\text{Long frame} > \text{aMaxSIFSFrameSize} \dots\dots\dots(2.3)$$

$$\text{Short frame} \leq \text{aMaxSIFSFrameSize} \dots\dots\dots(2.4)$$

[표 2.4] 각 채널의 IFS

Channel	0	1~10	11~26
LIFS	2ms	1ms	640 μ s
SIFS	600 μ s	300 μ s	192 μ s
t_{ack}	600 μ s \sqcup 1.6 μ s	300 μ s \sqcup 800 μ s	192 μ s \sqcup 512 μ s



[그림 2.10] 패킷 전송을 위한 시간간격(IFS)

제2절 에너지 효율적인 MAC 프로토콜

무선 시스템에서는 수신도 송신처럼 에너지를 소모한다. 이러한 전력소모는 배터리를 전원으로 장시간 동작해야 하는 센서 네트워크에서는 치명적이다. 기존의 Zigbee 기반으로 센서 네트워크를 구성하면 다음과 같은 에너지 소모 문제점이 있다.

- 충돌(Collision) : 2개 이상의 패킷이 한 채널에 동시에 전송되어 발생하여 전송 패킷에 오류가 발생하여 재전송으로 인한 쓸데없는 에너지 낭비 초래
- Overhearing : 자신과 상관없이 다른 노드로 가는 패킷을 수신하여 발생하는 에너지 소모
- Idle Listening : 네트워크에서 수신할 노드가 존재하지 않아도 노드가 수신을 대기하여 발생하는 에너지 낭비
- Overhead : SYNC/RTS/CTS 등의 제어 메시지 송수신에 의한 에너지 낭비

에너지 소모를 고려하여 센서 노드용 MAC은 자신이 송신하거나 수신할 때 Active 상태로 깨어나고, 나머지 시간에는 Sleep하여 전원을 절약하는 기능을 지원해야 한다. Active/Sleep을 적용한 센서 네트워크에서 중요한 부분은 Duty Cycle의 효율적인 사용이며, Duty Cycle을 효율적으로 사용하기 위한 MAC 프로토콜들이 연구되고 있다.

에너지 효율적인 MAC 프로토콜은 물리 계층에서 사용되는 채널의 수, 경쟁기반 그리고 수신되는 메시지가 있을 때 이를 인지하는 방법의 3관점으로 설계를 할 수 있다.

첫째, 물리 계층에서 사용하는 채널의 수에 따른 분류는 다중 채널 시스템의 충돌이 거의 없는 환경에서는 높은 에너지 효율을 보여주지만 다중 채널에 의한 복잡한 무선 통신이 상당한 에너지를 나타내기 때문에 MAC 프로토콜 설계시 단일 채널을 사용하는 경향으로 바뀌게 되었다. 두 개의 채널을 사용하는 MAC 프로토콜은 한 개의 데이터 채널과 Wake-up 채널을 사용하는데 송신 노드가

Wake-up 채널을 통해서 수신 노드를 깨울 때 다른 노드들도 이 채널을 듣고 깨어나서 어떤 노드가 수신 노드인지를 알려주는 신호를 들은 후 송신 노드가 지정한 수신 노드가 아닌 노드들을 다시 저 전력 소모 상태로 돌아감으로써 불필요한 전력 소모를 줄이는 방안이다. SMACS와 PICORADIO 등의 MAC 프로토콜이 다중 채널을 사용하는 MAC 프로토콜이다.

둘째, MAC계층에서 센서 노드의 조직화 방법에 따른 분류이다. 경쟁 기반 프로토콜은 랜덤 액세스 방식이기 때문에 다른 노드와의 전송 충돌로 인한 에너지 소모와 매체 접근을 위한 채널 감지 및 랜덤하게 수신되는 신호를 수신하기 위한 채널 감지를 사용하기 때문에 이로 인한 에너지 소모가 문제점으로 제기되었다. CSMA/CA 프로토콜이 대표적이며 무선 네트워크에서 주로 사용되어 왔다. 이웃 노드가 자신의 제어 패킷을 수신하게 해서 송수신시 매체에 접근하는 것을 막아 충돌을 피한다. 그러나 항상 수신 대기 상태에 있어야 하는 Idle Listening 문제가 발생하게 된다. 이러한 단점을 보완하여 소모되는 전력을 줄이기 위한 경쟁 기반 프로토콜로 S-MAC과 T-MAC이 대표적이다.

마지막으로, 센서 노드가 메시지 송수신을 어떻게 인지하는가에 따른 분류이다. 스케줄링 기반의 프로토콜은 사전에 각 노드가 통신을 할 수 있는 시간이 정해져 있기 때문에 항상 채널을 감지할 필요가 없다. 스케줄에 의존하여 각 센서 노드는 통신을 하기 때문에 Idle Listening 문제를 해결할 수 있다.

1. SENSOR MAC (S-MAC) 프로토콜

학교나 기업, 연구소 등에서 센서 네트워크에서의 에너지 소모량을 줄이기 위해 다양한 연구들이 진행되어 왔다. 센서노드의 에너지 소모량을 줄이기 위한 접근 방법으로는 하드웨어의 효율적인 디자인, 효율적인 MAC 프로토콜, 네트워크 라우팅 프로토콜의 연구 등 다양한 방법이 있다. 그중에서 본 논문에서는 Zigbee의 802.15.4 MAC 프로토콜을 개선하여 센서 노드의 소비전력량을 줄이고자 한다.

센서 MAC 프로토콜의 관점에서 센서노드가 에너지를 낭비하는 요인으로서는 Idle

Listening, Overhearing, Collisions, Traffic Fluctuations, Protocol Overhead와 같이 5가지로 정의할 수 있다.

일반적으로 센서 MAC 프로토콜은 센서노드의 에너지 소모를 감소시키기 위해서 Radio Module을 주기적으로 Wake-up/ Sleep 시키는 방법을 사용한다. 하지만 센서노드는 자신에게 언제 Packet이 도착할지 알 수 없기 때문에 실제로 통신을 하는 시간 이외에도 깨어있는 상태로 채널을 관찰하게 된다. 이런 이유로 에너지가 낭비되는 것을 Idle Listening이라 하며 이는 센서노드가 낭비하는 에너지의 대부분을 차지한다. 다음으로 Overhearing은 채널 상의 Packet의 흐름을 감지하였을 때, 그 Packet이 자신에게 오는 Packet이 아님에도 불구하고 Packet을 수신해야 하기 때문에 발생하는 에너지 낭비를 말한다. 위 두 가지 요인에 의한 낭비가 에너지 소모의 주원인이 되며, 다양한 센서 MAC 프로토콜들에서 이러한 요인들을 고려하여 저 전력 소모를 위한 방법들을 제안하였다.

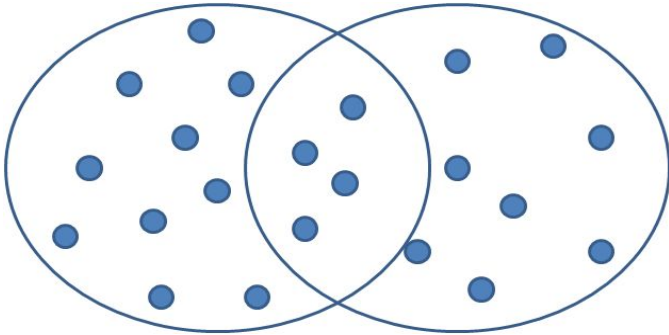
앞서 설명하였듯이 센서 네트워크에서의 MAC 프로토콜은 전력소모를 줄이기 위해서 Radio Module을 주기적으로 Wake-up/ Sleep 시킨다. S-MAC은 Sync Packet의 브로드 캐스트를 통해서 각각의 노드들을 슬롯이라는 일정한 시간의 구간으로 동기화 시킨다. 동기화된 각각의 센서노드들은 [그림 2.11]과 같이 슬롯 범위 안에서 고정된 시간을 Wake-up 상태로 채널을 관찰하고 나머지 시간을 Sleep 함으로서 Idle Listening으로 인해 낭비되는 에너지를 감소시켰다.



[그림 2.11] S-MAC 프로토콜의 기본 동작

S-MAC은 간단하고 구현이 쉽기 때문에 보편적으로 많이 사용되는 프로토콜이다. 하지만 고정된 Duty Cycle을 가지기 때문에 개발자들은 응용에 가장 효율적인 적절한 수면 주기를 사전에 선택해 주어야 하며 자신에게 전달될 Packet이 없음에도 불구하고 주기적으로 Wake-up 해야 하는 단점이 있다. 또한 [그림 2.12]와 같

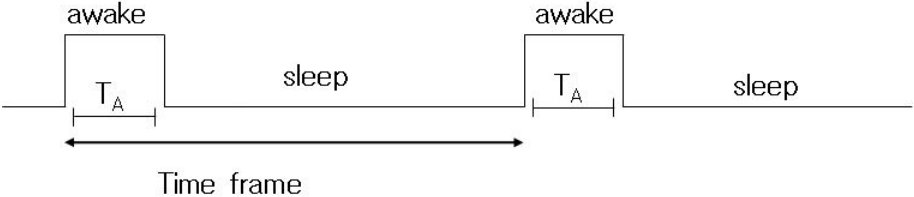
이 각각의 Sync Packet에 의해 동기화 된 서로 다른 클러스터가 만나게 되었을 경우 서로간의 동기화를 위한 작업이 필요하게 된다. 이때는 새롭게 전체 네트워크를 동기화 시켜주거나 중첩되는 곳의 센서노드가 새로운 타이머를 사용하여 각각의 클러스터에 맞는 동기화 정보를 제공하여야 한다. 이런 경우 중첩된 곳에 존재하는 센서노드는 상당한 에너지를 추가로 소모하게 되는 단점이 존재한다.



[그림 2.12] 클러스터가 중첩된 지역에서 동기화를 제공해야하는 연결노드

2. TIMEOUT MAC (T-MAC) 프로토콜

T-MAC은 고정적인 Duty Cycle로 인한 S-MAC의 단점을 보완하기 위해 제안되었다. S-MAC은 주기적인 Wake-up 과 Sleep을 통해 Idle Listening으로 인해 발생하는 전력소모의 낭비를 줄였지만 여전히 Packet의 전송이 없는 Active Time 동안 Wake-up 상태로 동작해야 하는 단점이 존재한다. T-MAC에서는 Timeout을 통해 이런 문제를 보완하였다. T-MAC의 기본적인 동작은 [그림 2.13]과 같다



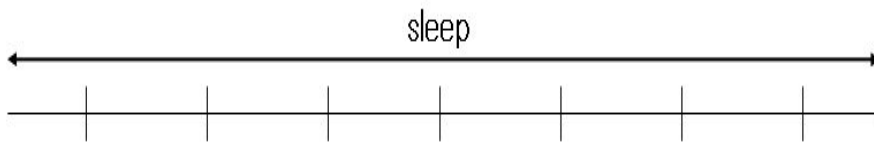
[그림 2.13] T-MAC 프로토콜의 기본 동작

T-MAC은 기본적으로 S-MAC과 같은 방식으로 동작한다. 하지만 S-MAC은 주어진 Active 시간동안 Wake-up 상태로 동작하는 반면 T-MAC은 자신이 보낼 Packet이 없거나 T_A 라는 일정 시간 동안 채널에 Packet의 흐름이 감지되지 않으면 바로 Sleep상태로 들어간다. 따라서 S-MAC의 Active 시간동안의 Idle Listening에 소비되는 에너지를 줄일 수가 있다.

3. PATTERN MAC (P-MAC) 프로토콜

P-MAC 은 무선 센서 네트워크에서 동적으로 수면 주기를 조절하는 메커니즘을 제공한다.

P-MAC도 S-MAC과 같이 분할된 시구간에 적용하는 프로토콜이다. S-MAC은 주기적으로 수면을 반복하지만, P-MAC은 그와 다르게 수면과 깨어남을 노드가 결정 할 수 있다. 노드 자신의 데이터 전송량이나 이웃 노드의 데이터 전송량에 따라서 수면 주기가 동적으로 스케줄 된다. 패턴이라는 메커니즘을 사용하여 이전의 보낸 데이터의 패턴 유형을 찾고 그 패턴에 맞추어 다음 구간의 깨어남을 스케줄링 하게 된다. 네트워크 트래픽에 따라서 동적으로 수면 주기를 설정하기 때문에 네트워크 트래픽에 따른 전력 관리가 가능하다는 것이 장점이다.



[그림 2.14] P-MAC 프로토콜의 기본 동작

[그림 2.14]는 P-MAC 이 네트워크에 어떻게 적용되는지 나타낸다. 현재의 STF(Super Time Frame) 트래픽에 따라 생성된 자신의 패턴은 다음 STF에 적용 된다. 자신의 패턴은 PETF(Pattern Exchange Time Frame)에서 교환이 이루어지

며, 자신의 깨어 날 패턴과 상대방이 깨어있는 패턴을 알기 때문에, 동적으로 수면을 제어 하면서, 트래픽에 따라서 수면 주기를 조절 할 수 있게 된다. P-MAC 프로토콜은 센서 네트워크에서 효율적인 전력 제어 메커니즘을 제공하고 있지만, 아직 구현되거나 활용되지 않는 기술이다.

제3장 저전력 소모를 위한 MAC 프로토콜의 제안

앞서 소개한 효율적인 MAC 프로토콜들은 기본적으로 센서 장치가 깨어 있는 시간을 최대한 줄임으로써 전력 소비를 줄이는 방식을 채택하고 있다. 이와 같은 연구로 인하여 센서 장치의 전력 소모 기법은 날로 발전하였지만, 전력 소모가 줄어드는 만큼 응답속도가 떨어지는 문제점은 그대로 안고 있어야 한다.

센서 노드는 배터리를 사용하는 저전력 저가의 시스템으로 에너지 사용에 제약을 받으며 배터리가 소모되어 노드의 동작이 멈추면 더 이상 네트워크의 수단으로 사용될 수 없다. 따라서 무선 센서 네트워크는 노드에 사용되는 전력의 사용을 효율적으로 하여 네트워크의 생존시간을 최우선으로 생각한다.

무선 통신 프로토콜 중 하나인 Zigbee(IEEE 802.15.4)가 다른 무선 프로토콜보다 적은 전력소모로 데이터 전송이 가능하며 간단한 조작으로 데이터 네트워크를 구축할 수 있기에 센서 네트워크에서 적합하다고 평가되며 국내에서는 2.4GHz의 주파수 대역을 사용하고 있다. 센서 네트워크 MAC Layer에서는 무선 계층의 에너지 소모를 줄이기 위해 가장 에너지 소모가 큰 영역인 Radio에 대한 연구를 진행하고 있다. 센서 노드의 상태를 Sleep과 Active 상태로 구분하여, 센서 노드가 데이터 전송을 하지 않을 경우 Radio에 사용되는 불필요한 전원에 따른 에너지 소비를 줄이는 방안에 대해 연구가 활발히 이루어지고 있다. 하지만 이 방법의 문제점은 수면주기를 길게 할수록 전력 소모는 줄일 수 있지만, 말단 센서 장치에서 Coordinator라 불리는 중앙 센서 장치로 데이터 전달 속도가 낮아지는 문제는 앓고 있다. 예를 들어 침입 센서나 화재 경보 센서 같은 경우 응답 속도를 늦추게 되면 센서의 전력 소비는 줄어들겠지만, 응답속도가 낮아져서 사건의 대비가 늦어지게 되는 문제점이 생기게 된다.

주기적으로 센서 장치를 수면시키는 메커니즘은 전력 소모를 줄일 수 있지만 상당히 고정적이기 때문에 다양한 네트워크의 상황에 효과적으로 대응하기에는 부족한 점이 적지 않다. 따라서 수면주기를 동적으로 할당하고 센서 장치에서 중앙 데

이더 처리 장치로의 데이터 전송 응답 시간을 줄이는 방안을 연구하여야 하는 필요성이 있다. 본장에서는 수면주기를 이전에 수집된 센서 데이터를 토대로 패턴을 만들고 이를 토대로 수면주기를 결정하여 스케줄링하는 동적 수면 프로토콜을 제안한다.

제1절 RSSI값 제어와 동적 수면주기를 이용한 저전력 기법

패킷의 정보에는 송신할 노드의 설정된 RSSI값과 수신된 전송 파워를 포함한다. 패킷을 수신하는 노드는 실제 송신노드의 RSSI값과 수신된 RSSI값을 받기 위해 설정된 실제 설정된 값을 통해 거리가 얼마나 떨어진 노드에서 보낸 패킷인지 알 수 있다. 이 정보는 정확하지는 않지만 어느 정도의 거리가 떨어진 노드에서 송신한 패킷인지 알 수 있고 이를 토대로 응답 메시지를 보낼 때 적절한 전송 파워를 설정하여 보내면 노드의 전력 소모를 줄일 수 있다.

[표 3.1]은 실험에 사용할 RF칩인 CC2420이 제공하는 8단계의 파워 레벨이다.

[표 3.1] CC2420이 제공하는 8단계의 파워 레벨

Register Value	Output Power[dBm]	Current Consumpton[mA]
0x1f	0	17.4
0x1b	-1	16.5
0x17	-3	15.2
0x13	-5	13.9
0x0f	-7	12.5
0x0b	-10	11.2
0x07	-15	9.9
0x03	-25	8.5

CC2420은 최고 0dBm부터 최소 -25dBm으로 전파를 송신 할 수 있으며, 수신측에서 받게 되는 전파의 세기가 -94dBm 이상일 경우 전파를 수신할 수 있다. CC2420의 RSSI값은 RSSI.RSSI_VAL 레지스터에 저장되며, 그 값을 dBm단위로 표현할 경우에는 실험값 약 45정도의 값을 빼주어야 한다. 예를 들어, RSSI.RSSI_VAL 레지스터에 값이 -20일 경우, 실제 받은 패킷의 RSSI는 -65dBm

정도가 된다.

패킷의 RSSI값의 토대로 전송파워를 결정하는 것만으로는 전력소모를 많이 줄일 수 는 없다. 이유는 RSSI값이 전파의 세기를 측정한 것이라서 장애물의 영향도 많고 동일 조건의 실험을 통해 측정된 값이라 하더라도 결과가 상이 할 수가 있기 때문이다. 따라서 이 값은 전력소모를 줄이는데 있어서 보조 역할을 수행한다.

전력소모의 주된 요인은 RF칩이다. RF칩에 전력을 차단함으로써 전력의 소모를 효과적으로 줄일 수 있는데 무턱대고 전력을 차단하면 원하는 데이터를 얻을 수 없을 것이다. 따라서 RF칩의 전력 차단을 주기적으로 함으로써 전력 소모를 줄일 수 있으면서 데이터의 전달 율을 높일 수 있을 것이다. 하지만 다양한 상황의 네트워크에 대응하기에는 역부족이다. 그래서 그 수면주기를 이전에 받은 데이터를 토대로 패킷을 만들고 수면주기에 패킷을 적용하여 동적 수면주기를 할당한다.

초기 단계에는 S-MAC과 비슷하게 동작한다. 각각의 노드는 동작을 시작한 이후, 처음에는 일정한 시간동안 Listen과 Sleep를 반복하는 상태로 있으면서 이웃노드의 패킷 정보를 바탕으로 다음 Sleep 주기를 결정한다.

동작하나의 패킷이 적용되는 시구간은 동등한 n 개의 시구간으로 구성된다. 이는 $T_m = \{P_0, P_1, P_2, \dots, P_{n-1}\}$ 으로 나타내며 패킷이 적용되는 시구간 중에 반드시 P_0 구간에서는 깨어나야 하기 때문에 최대 $n-1$ 구간 개수 만큼 연속으로 수면할 수 있다.

패킷을 적용할 경우에 대해서는 크게 3가지의 경우를 고려 할 수 있다. 첫째는 데이터의 변화량이 감소 할 때이고, 둘째는 변화가 없을 때, 마지막으로 셋째는 데이터의 변화량이 증가 할 때이다. 따라서 각 경우에 맞게 패킷을 갱신하고 적용하여 스케줄링을 결정 하는 것이 필요하다.

패킷의 적용과 갱신 그리고 스케줄링을 결정하는 방식은 다음과 같다.

패킷 T_m 은 m 번째 시구간에서 적용되는 패킷이다. 이 패킷은 $m-1$ 번째 시구간에서의 수집된 데이터에 따라서 생성된다. 이전 구간에서의 데이터의 변화량에 따라서 현재의 패킷이 결정되게 된다. 패킷의 적용은 패킷이 시작되는 구간, 즉 m 번째 시구간의 P_0 구간부터 적용되며, 한번 적용된 패킷은 다음 $m+1$ 번째 시구간의 P_0 구간이 오기 전까지 유지된다.

m+1번째 시구간에서의 패턴은 다음과 같이 결정된다. 먼저 m+1번째의 시구간에서 결정될 패턴은 m 구간에서 결정되며 m번째 구간은 P₀부터 P_n까지의 모든 구간에서 wake up 상태이며 다음 구간에 적용될 패턴을 갱신하게 된다. m+1번째 구간의 P₀부터 P_n까지의 상태는 다음 m번째 구간의 P₀부터 P_n까지의 패턴에 따라 결정된다.

수면구간을 비트 0이라하고 wake up 구간을 1이라할 때 다음과 같이 스케줄링할 수 있다.

- m-1번째 구간의 데이터의 변화량이 적거나 같은 경우 : 수면구간을 적용하여 센싱 주기를 늘린다. ($\{1_0, 1_1, 1_2, 1_3, \dots, 1_n\} \rightarrow \{1_0, 0_1, 1_2, 0_3, \dots, 0_n\}$)
- m-1번째 구간의 데이터의 변화량이 m+1번째 시구간에도 변화량이 적은 경우 : 수면구간을 2k만큼 늘린다. ($\{1_0, 0_1, 1_2, 0_3, \dots, 1_n\} \rightarrow \{1_0, 0_1, 0_2, 1_3, \dots, 0_n\}$)
- m-1번째 구간의 데이터의 변화량이 증가한 경우 : 수면 구간을 감소시켜 데이터 변화량이 많은 구간에 대해 센싱 주기를 늘려 네트워크 반응 속도 및 데이터 신뢰도 향상을 가져온다.

위와 같은 방법으로 결정된 수면주기는 RSSI값에 의해 결정된 센싱 파워로 센싱 값을 전달한 노드에게 스케줄링하게 된다.

제 2 절 데이터 변화량의 Threshold의 설정

측정값은 환경과 사용하는 센서에 큰 영향을 받는다. 예를 들어 민감한 센서라면 측정값이 계속 약간씩 변할 것이다. 그런 경우에는 수면주기가 짧아 질수 있어 에너지 소비가 늘어날 것이다. 따라서 그에 따른 값의 오차를 허용해 주어야 한다. 센서의 종류나 환경에 따라서 측정값의 크기와 변화가 다르므로 사용하는 상황에 맞춰서 임의로 결정해 주어야한다. 예를 들어 온도나 습도의 경우 일정 시간동안 날씨가 변화 없다면 약간의 변화만 있을 뿐 거의 일정하게 유지 될 것이다. 이런 경우 Threshold의 값을 낮게 잡아주면 센서의 수면 주기는 짧아지고 네트워크는 계속해서 동작 하게 될 것이다. 따라서 적용하는 데이터나 환경에 따라서 적당한 값의 오차를 허용해주며 사용하는 상황에 맞춰서 결정해 주어야 한다.

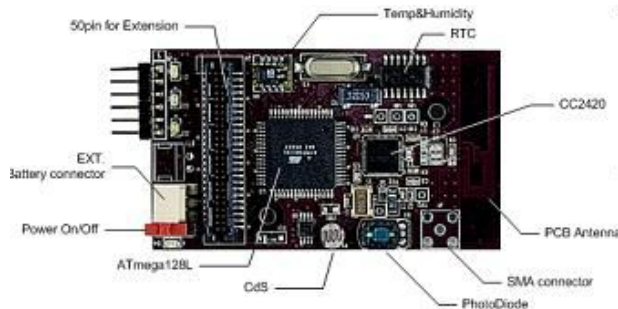
위에 기술한 내용의 MAC Protocol을 LPMAC이라 명명하였다.

제4장 시뮬레이션 및 실험

IEEE802.15.4를 지원하는 칩셋으로 Chipcon사의 CC2420과 모토로라의 자회사인 Freescale사의 MC1319/92가 대표적이며, 국내의 KETI, 삼성전자 그리고 RadioPulse, Inc등의 업체에서도 IEEE802.15.4를 지원하는 칩셋이 개발되어 있다. 본 논문에서는 제안한 프로토콜의 구현 및 성능 평가를 위해 IEEE802.15.4를 지원하는 Chipcon사의 CC2420을 적용한 한백전자의 ZigbeX 모트를 사용하였다.

제1절 구현 환경

ZigbeX 모트는 ATmel사의 ATmega128L과 Chipcon사의 CC2420을 적용한 센서 네트워크 모트이다.



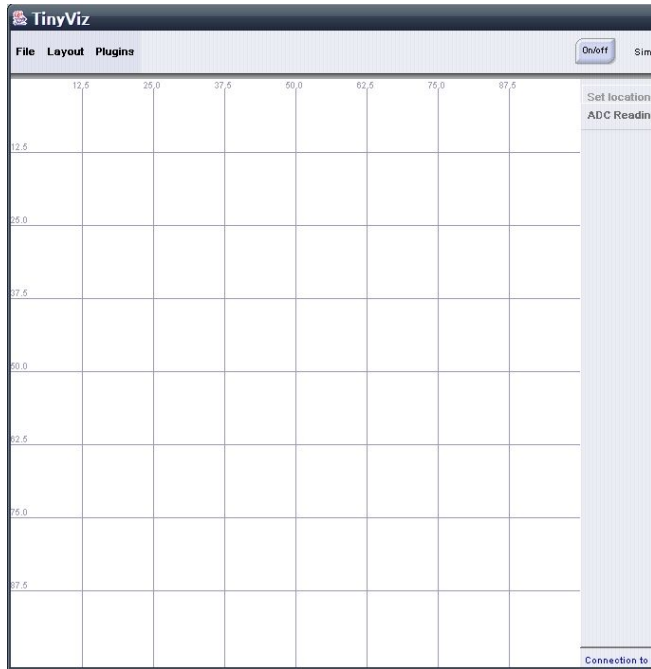
[그림 4.1] ZigbeX 모트의 구성도

[그림 4.1]은 ZigeX의 구성을 보여주는 것으로 온/습도 센서와 조도센서가 기본 센서로 장착되어 있으며 50pin 커넥터를 이용하여 외부 센서 모듈을 확장하도록 되어 있다. 전원은 AA 건전지 2개를 사용하며 사용자의 선택 사항을 위해 외부 전원용 커넥터가 존재한다. RF 안테나는 PCB안테나를 기본으로 사용하며 사용자의 선택에 따라 Dipole안테나를 이용할 수 있다. [표 4.1]은 ZigbeX 모트의 구성을 정리한 것이다.

[표 4.1] ZigbeX 플랫폼 요약

MCU	Device	Type	Flash Memory(kb)	SRAM(kb)
	ATmega128L	7.3728Mhz	128	4
RF Transceiver	Device	Radio Frequency(Mhz)	Max.Data Rate(kbits/sec)	Antenna
	CC2420	2400	250	PCB antenna(Default)/SMA (Alternative)
Flash memory	Device	Connection	Size(kB)	
	AT45DB041B	SPI	512	
Power	Type	Capacity(mA-hr)	3.3Vbooster	External Power
	AAx2	1000~2300	Default	Ext. Connector
Sensor	Humidity and Temperature	SHT11		
	Light	CdS		
	Photo Diode	BS-520		
LED	Green LED	Indicate LED		
	Red LED	Indicate LED		
	Yellow LED	Indicate LED		

ZigbeX 모트의 동작 상태는 기본적으로 모트의 LED를 통해 확인이 가능하지만, 모트의 실제 동작을 확인하기 위해 모니터링 도구인 Tinyviz와 Moteview를 사용한다. Tinyviz와 Moteview는 센서 네트워크의 토폴로지 상태를 PC 등의 시스템으로 확인이 가능하며 그 외 TinyOS응용 프로그램에서 센서를 통해 수집한 값을 직접 GUI 및 Text환경으로 모트의 동작 및 센서 네트워크의 상태를 모니터링 할 수 있는 다양한 도구들이 있다.



[그림 4.2] tinyviz 실행화면

[그림 4.2]는 Tinyviz의 실행화면으로 시뮬레이션을 위한 도구이다. Tinyviz는 센서의 정보를 시각적으로 제공해주고 시뮬레이션의 디버깅 메시지, 무선과 UART 패킷, 기타의 것들을 제어할 수 있도록 제공하는 Java 기반의 GUI 시뮬레이션 도구로 TOSSIM과 상호 작용하는 몇 가지 메커니즘을 제공하며, 관심 있는 이벤트에 Breakpoint를 찍어 쉽게 TinyOS 응용프로그램을 모니터링 할 수 있다. Tinyviz는 현재 TinyOS-1.x에서 동작하도록 구성된 프로그램이다.

제안된 프로토콜의 실험은 TinyOS-2.x에서 프로그래밍하여 컴파일한 후 ZigbeX에 포팅하여 전력파워 컨트롤을 Spectrum analyzer로 관찰하였다. [그림 4.3]은 cygwin에서 AVR C Compiler을 이용하여 컴파일과정을 보여준다.


```

Administrator@notebook-PC /opt/tinyos-2.x/apps/LP_MAC
$ make zigbex
mkdir -p build/zigbex
  compiling LP_MACAppC to a zigbex binary
ncc -o build/zigbex/main.exe -Os -finline-limit=100000 -Wall -Wshadow -fno-sanitize=all -target=zigbex -fno-sanitize=file=build/zigbex/app.c -board=mic28 -fno-sanitize=dump=wiring -fno-sanitize=dump=interfaces(labstack()) -fno-sanitize=dump=referenced(interfacedefs, components) -fno-sanitize=dump=file=build/zigbex/wiring-check.xml LP_MACAppC.nc -I.
  compiled LP_MACAppC to build/zigbex/main.exe
      19448 bytes in ROM
      1056 bytes in RAM
avr-objcopy --output-target=srec build/zigbex/main.exe build/zigbex/main.srec
avr-objcopy --output-target=ihex build/zigbex/main.exe build/zigbex/main.hex
writing TOS image

Administrator@notebook-PC /opt/tinyos-2.x/apps/LP_MAC
$ ls
LP_MACApp.h  LP_MACAppM.nc  LP_MACAppRoutingC.nc  Makefile
LP_MACAppC.nc  LP_MACAppRouting.h  LP_MACAppRoutingM.nc  build

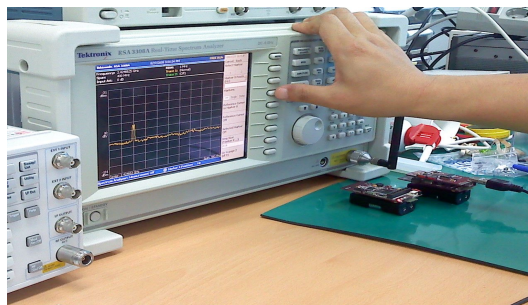
Administrator@notebook-PC /opt/tinyos-2.x/apps/LP_MAC
$ ls build/zigbex/
app.c          main.exe.setid0.hex  main.exe.setid3.hex  main.srec
main.exe      main.exe.setid1.hex  main.exe.setid4.hex  tos_image.xml
main.exe.setid  main.exe.setid2.hex  main.hex              wiring-check.xml

Administrator@notebook-PC /opt/tinyos-2.x/apps/LP_MAC
$ ls build/zigbex/

```

[그림 4.3] cygwin에서 컴파일 결과

컴파일 과정후 생성된 Hex파일은 AVR studio 프로그램과 ZigbeX 확장모듈인 USB-ISP를 통해서 모트에 포팅하여 실제 전력 제어가 잘되는지 실측을 위한 준비를 마친다. [그림 4.4]는 Spectrum analyzer로 데이터 수신 후 송신측의 RSSI값과 실측된 RSSI값을 토대로 CC2420의 레지스터 값을 수정하여 송신하는 전파 세기의 변화를 관찰한 것이다. Spectrum analyzer는 특정 주파수에서의 전파 파형을 화면으로 볼 수 있는 장비로서 해당되는 채널의 신호강도를 측정할 수 있다.



[그림 4.4] Spectrum analyzer를 이용한 측정 준비

채널 간 대역폭이 5MHz이고 모트의 레지스터에 설정된 RSSI값은 -25dBm으로 설

정하였다. 채널의 결정은 802.11에서 사용하는 채널의 범위가 겹치는 영역이 많기 때문에 충분히 고려하여 결정하여야 한다. 802.15.4의 채널 간 대역폭은 5MHz이고 802.11의 채널 간 대역폭은 22MHz이다. 따라서 802.15.4의 15, 20, 25, 26번 채널을 제외하고는 전 영역에 걸쳐서 채널을 공유하게 된다. 실험의 주파수 간섭을 최대한 피하기 위해 설정된 채널은 26번으로 하였다.

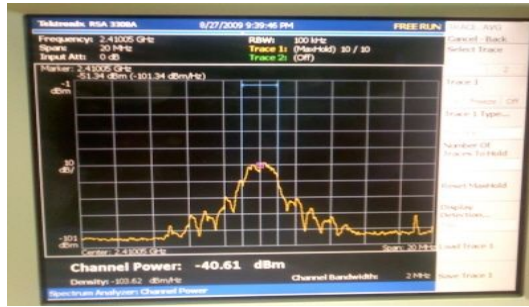
Spectrum analyzer에서 측정된 값은 약 -52dBm부터 -27dBm 정도로 약 10센티에서 -27.17dBm, 약 1m 거리에서 -40.61dBm, 약 2m 거리에서 -51.39dBm 의 전파세기가 측정되었으며 이로 인해 패킷에 포함된 RSSI 설정 값과 수신된 RSSI값에 따라 송신 파워를 조절이 되고 있음을 알 수 있었다.



[그림 4.5] 프로그램을 mote에 포팅후 Spectrum analyzer 측정준비



[그림 4.6] 전송거리가 약 10센티일 때 채널 간 전파 세기



[그림 4.7] 전송거리가 약 1m일 때 채널 간 전파세기



[그림 4.8] 전송거리가 약 2m일 때 채널 간 전파세기

위의 실험으로는 제안한 프로토콜의 성능을 평가하기에는 부족하다. 프로토콜의 패턴적용과 스케줄링이 효과적으로 잘 적용되어 기존의 에너지 효율적인 프로토콜의 성능과 비교해 볼 필요성이 있다. 하지만 앞서 언급한 Tinyviz나 moteviewer 프로그램은 센서 네트워크를 비주얼하게 볼 수 있는 장점을 가지고 있지만 모트의 동작에 따른 에너지 소모를 측정할 수 없다.

Powertossim은 tossim 시뮬레이터를 개조한 것으로서 CC1000 RF칩의 전력 소모를 측정할 수 있는 시뮬레이터이다. 하지만 정작 실험에 사용되는 CC2420의 스펙과는 다소 차이가 있어 그에 맞는 수정을 하거나 CC2420에 맞는 시뮬레이터로 실험을 하여야 한다. Powertossim-z는 micaz모트에 맞게 설계된 시뮬레이터이며 micaz모트는 CC2420을 사용한다. [그림 4.6]은 Powertossim-z을 다운로드할 수 있고 간략한 설명이 나와 있는 웹사이트이다.



[그림 4.9] PowerTossim-z 소스 다운로드

PowerTossim-z를 통해 실험을 하기 위해서는 trace파일이 필요한데 PowerTossim-z 자체로는 trace 파일을 생성 할 수 없다. trace 파일은 PowerTossim으로 얻을 수 있다. PowerTossim은 CC1000에 맞는 에너지 소모 모델을 사용하므로 PowerTossim으로는 trace 파일만을 생성하고 생성된 trace파일을 통해 PowerTossim-z를 실행하여 결과를 얻을 수 있다. 하지만 PowerTossim은 tinyos-1.x버전에 맞는 것이므로 PowerTossim2를 통해서 trace파일을 생성해야만 오류 없이 trace파일을 생성할 수 있다. PowerTossim-z에서 이용하는 energy_model은 [표 4.2]와 같다.

[표 4.2] PowerTossim-z에서 사용하는 micaz energy_model

VOLTAGE	3.0	RADIO_TX_EB	11.2
CPU_ACTIVE	8.93	RADIO_TX_E7	9.9
CPU_IDLE	4.93	RADIO_TX_E3	8.5
CPU_ADC_NOISE_REDUCTION	0.32	RADIO_TX_FF	17.4
CPU_POWER_DOWN	0.0003	RADIO_RX	19.7
CPU_POWER_SAVE	0.009	RADIO_OFF	0
CPU_STANDBY	1	RADIO_DEFAULT_POWER	255.0
CPU_EXTENDED_STANDBY	0.25	LED	2.2
CPU_INIT	3.2	LED_INIT	0
CPU_FREQ	7382800	EEPROM_READ	6.24
RADIO_TX_FB	16.5	EEPROM_WRITE	18.40
RADIO_TX_F7	15.2	ADC	0.0
RADIO_TX_F3	13.9	SENSOR_BOARD	0.69
RADIO_TX_EF	12.5		

생성한 trace파일은 2개이며 하나는 SMAC의 기본 동작 trace파일, 다른 하나는 LPMAC의 동작의 trace파일이다. 생성된 trace 파일에는 시뮬레이션 시간별 동작 하게 되는 이벤트가 기술되어 있고 이 trace파일의 내용과 실행결과는 [그림4.10], [그림4.11]과 같다.

```

DEBUG (0): 95298235002_RADIO_STATE_REC_MESSAGE_DONE_DEST:65535
DEBUG (0): 95298235002_LED_STATE_LED2_ON
DEBUG (0): 95298235002_LED_STATE_LED1_ON
DEBUG (0): 95298235002_LED_STATE_LED0_ON
DEBUG (0): 95302167615_RADIO_STATE_SEND_MESSAGE_ON_DEST:65535_SIZE:9_DB:0
DEBUG (0): 95308881443_RADIO_STATE_SEND_MESSAGE_OFF_DEST:65535_SIZE:9
DEBUG (0): 95344836449_RADIO_STATE_REC_MESSAGE_DONE_DEST:65535
DEBUG (0): 95344836449_LED_STATE_LED2_ON
DEBUG (0): 95344836449_LED_STATE_LED1_ON
DEBUG (0): 95344836449_LED_STATE_LED0_ON
DEBUG (0): 97679792499_CPU_STATE_CPU_ACTIVE
DEBUG (0): 97679792699_CPU_STATE_CPU_IDLE
DEBUG (0): 97728072403_RADIO_STATE_REC_MESSAGE_DONE_DEST:65535
DEBUG (0): 97728072403_LED_STATE_LED2_OFF
DEBUG (0): 97728072403_LED_STATE_LED1_OFF
DEBUG (0): 97728072403_LED_STATE_LED0_OFF
DEBUG (0): 97780839742_RADIO_STATE_REC_MESSAGE_DONE_DEST:65535
DEBUG (0): 97780839742_LED_STATE_LED2_OFF
DEBUG (0): 97780839742_LED_STATE_LED1_OFF
DEBUG (0): 97780839742_LED_STATE_LED0_OFF
DEBUG (0): 97784924942_RADIO_STATE_SEND_MESSAGE_ON_DEST:65535_SIZE:9_DB:0
DEBUG (0): 97791638770_RADIO_STATE_SEND_MESSAGE_OFF_DEST:65535_SIZE:9

```

[그림 4.10] Trace 파일의 내용

```

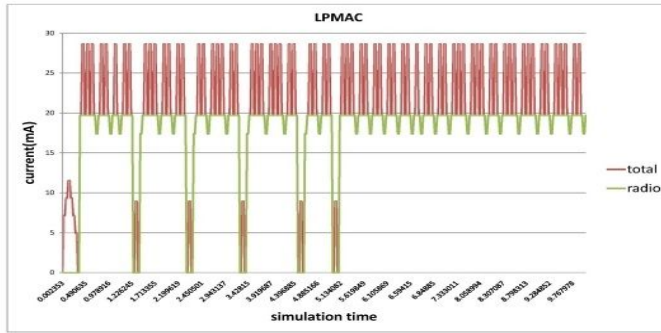
$ python postprocessZ.py PowerTOSSIM2_SMAC.trace
Efficiency table loaded, size=102
Recovery table loaded, size=11
Trace file used: PowerTOSSIM2_SMAC.trace
Energy model used: mica2_energy_model.txt
maxseen 0
note battery starting energy: 21600000 mJ
Mote 0, cpu total: 144.6
Mote 0, radio total: 577.6
Mote 0, adc total: 0.0
Mote 0, leds total: 98.6
Mote 0, sensor total: 0.0
Mote 0, cepron total: 0.0
Mote 0, cpu_cycle total: 0.0
Mote 0, Total energy used: 821
Mote 0, Battery energy remaining (linear): 21599179
Mote 0, Battery energy remaining: 21599133
Mote 0, Battery mAh remaining: 1999.9

Simulated seconds: 9.8
Real seconds: 0.1

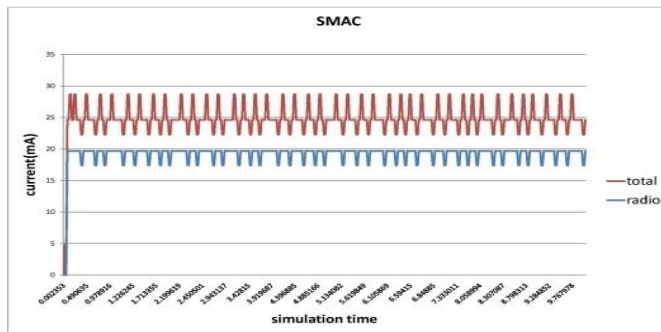
```

[그림 4.11] Trace 파일을 powertossim-z로 실행한 결과

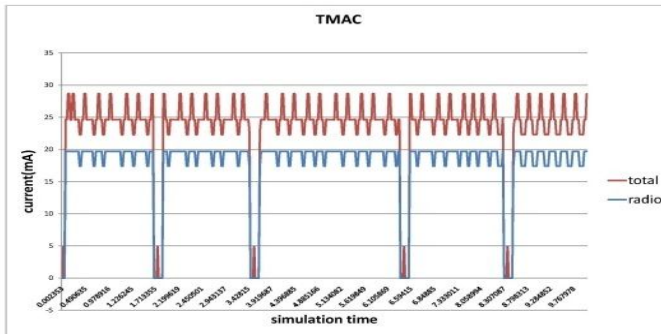
powertossim-z를 실행할 때 detail 옵션을 주게 되면 작동하는 모트의 각 이벤트에 따른 상세한 에너지 소비가 포함된 파일을 text 형태로 얻을 수 있다. [그림 4.12]는 LPMAC의 모트가 동작할 때 소비하는 전류량이며 radio 범주는 통신 할 때 소비되는 전류량을 나타낸 것이고, total 범주는 cpu, 작동 센서, 통신 등 각 이벤트 당 모든 장치가 소모하는 에너지의 합을 나타낸 것이다. [그림 4.13]과 [그림 4.14]는 마찬가지로 SMAC과 TMAC 동작의 소비 전류량을 나타낸 것이다. 각 그림에 나타난 시뮬레이션 시간은 약 10초간을 절취한 것이며 LPMAC의 경우 패킷이 적용되어 수면 구간이 생긴 것을 알 수 있었다.



[그림 4.12] LPMAC의 에너지 소모량



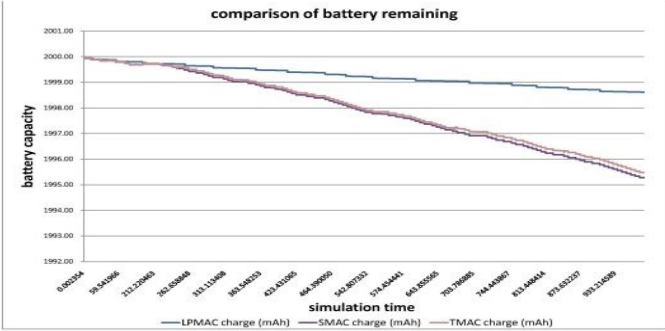
[그림 4.13] SMAC의 에너지 소모량



[그림 4.14] TMAC의 에너지 소모량

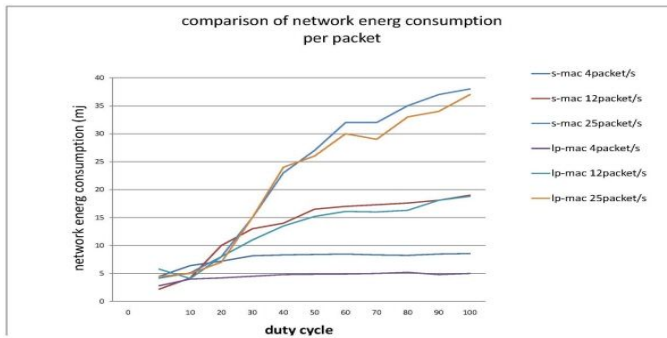
[그림 4.15]는 LPMAC과 SMAC으로 동작하는 각각 모드가 시간의 변화에 따른 배터리 잔량을 나타낸 것으로써 약 1000초간의 시뮬레이션 이후 약 4.5mA의 배터

리 잔량의 차이를 보이는 것으로 나타났다. 시뮬레이션의 조건은 일정한 거리에 있는 두 모트간 약 30bytes의 패킷의 크기를 초당 4packet을 전송하는 것이다. SMAC의 경우 수면 주기와 통신 주기가 일정함으로 일정한 양의 전류를 계속 소모하며 LPMAC 의 경우 정해진 시뮬레이션 스케줄에 의해 동작함을 가정했을 때 패킷이 적은 경우 수면주기가 길어져 SMAC의 배터리 소모보다 유리함을 알 수 있었다. 응답속도의 경우 LPMAC은 수면주기가 짧아지는 구간의 경우 다른 MAC 보다 좋다고 할 수 있으나 수면주기가 길어진 구간의 경우 다른 MAC보다 응답속도가 같거나 느리다고 할 수 있다. LPMAC의 시구간 T를 SMAC의 수면주기만큼 설정한다면 LPMAC은 SMAC보다 응답속도 면에 있어서 같거나 좋다고 할 수 있다. SMAC의 1주기동안 최소 1주기, 최대 P주기 동안 데이터를 수신하기 때문이다. TMAC의 경우와도 비슷하다고 볼 수 있다. 수치상으로 표현할 수 있는 실험은 이루어지지 않았지만 실험과 계산을 한다면 이와 같은 조건일 때 LPMAC이 다른 MAC보다 약간은 응답속도 면에서 앞설 것이라고 예측된다.



[그림 415] 시간 변화에 따른 LPMAC과 SMAC, TMAC의 에너지 잔량비교

powertosism-z에서 사용한 micaz의 CPU, Radio의 파라미터를 그대로 적용하여 평균 packet의 사이즈는 30bytes, 송신 전력은 0dBm(최대)로 하여 초당 전송 패킷이 각각 4packet, 12packet, 25packet일 때 수면주기가 static한 SMAC과 비교하였다.



[그림 4.16] 초당 송수신 패킷수별 전체 네트워크에서 소비하는 에너지비교

SMAC과 비교했을 때 LPMAC은 초당 패킷의 전송이 적을수록 성능이 개선된 것을 알 수 있었다. [그림 4.16]과 같이 4packet일 때는 약 40%정도, 12packet일 때는 1%, 25packet 경우 3%정도의 수치이다. 이는 전송되는 데이터의 변화량에 따른 것으로 데이터가 적게 수신됨으로써 변화량에도 큰 차이가 나지 않아 전체 네트워크의 에너지 소비가 줄어든 것으로 보인다. 반면 초당 패킷 전송이 많을 때는 데이터가 많이 수신됨으로써 그에 따른 변화량의 변화도 많아져 그만큼 수면 주기가 짧아진다. 따라서 통신의 횟수가 많아져 에너지 소모도 늘어나게 되어 SMAC과 비교했을 때 큰 차이를 보이지 않거나 효율이 떨어짐을 보였다. 하지만 duty cycle이 길어질수록 LPMAC의 에너지 효율이 SMAC보다 좋다는 것을 알 수 있다.

제5장 결론

많은 학교나 연구소, 기업에서 센서 네트워크 장치의 전력 소비를 줄이기 위해서 하드웨어 디자인은 물론 저전력 네트워크 토폴로지 구성, 효율적인 MAC Layer 프로토콜등의 연구들이 진행 되어 왔다. 비교적 데이터의 전송 빈도가 낮은 센서 장치에서는 불필요한 채널 관찰로 인한 전력 소모가 전체 전력 소모에서 큰 비중을 차지하게 된다. 센서 장치들의 전력 소모 기법은 날로 발전하였지만, 전력 소모가 줄어드는 만큼 응답속도가 떨어지는 문제점은 그대로 안고 있어야 한다. 주기적으로 센서 장치를 수면시킴으로써 전력 소모를 줄일 수 있는 메커니즘이 있지만 그 메커니즘은 상당히 고정적이기 때문에 다양한 네트워크의 상황에 효과적으로 대응하기에는 부족한 점이 적지 않다.

센서 네트워크는 특성상 배터리로 동작하게 된다. 따라서 한정된 에너지 제원으로 장시간 시스템을 유지 시키는 것이 큰 목적이다. 이런 목적에 부합하는 방식이 수면과 깨어남을 반복으로 하는 프로토콜인데 이 프로토콜을 사용하면 수면하는 시간 만큼에 해당하는 에너지 소비를 줄일 수 있지만, 수면 시간 동안 데이터 응답을 할 수 없기 때문에 응답속도가 늦어지는 단점이 있다. 이러한 단점을 보완하기 위해서 네트워크의 데이터 전송량에 따라서 수면 주기를 달리 하고 주변 노드의 RSSI정보에 따라 송신파워를 조절함으로써 전력 소모를 관리 할 수 있는 프로토콜을 제안하였다.

본 논문에서는 제한된 전력량만을 가지고 그 수명을 최대한 늘려야하는 센서 네트워크의 특성에 대해 가장 많은 전력을 소모하는 RF칩에 전력을 주기적으로 차단하는 것을 수면주기라 하는데, 그 수면주기를 이전에 수집된 센서 데이터와 비교하여 동적으로 수면주기를 할당하고 패킷 정보에 RSSI값을 통한 거리 정보를 포함함으로써 효율적으로 송신 파워를 조절 하여 센서 노드의 전력 소모를 줄이는 프로토콜을 제안하였다. 기존의 수면주기를 주기적으로 할당함으로써 얻어지는 에너지 절약의 이점과 일정한 수면주기 덕에 데이터의 응답속도가 느려진다. 제안한 기

법을 통해 수면주기를 동적으로 할당하고 송신 파워도 동적으로 변경함으로써 수면주기가 일정할 때 보다 에너지 효율이 좋아졌으며, 이는 네트워크의 라이프 타임을 연장하는 효과를 가져왔다.

제안한 프로토콜의 검증과 성능 평가를 위해 powertossim-z시뮬레이터를 사용하였으며 이를 이용한 모의실험 결과, 수면 주기가 동적으로 할당되고 있음을 알 수 있었고, 초당 패킷전송이 적은 경우 배터리 소모와 데이터 응답속도에 있어서 SMAC과 TMAC보다 LPMAC이 유리함을 알 수 있었다. 또한 초당 송수신 패킷수별 전체 네트워크에서 소비하는 에너지비교에서는 SMAC대비 4packet일 때는 약 40%정도, 12packet일 때는 1%, 25packet 경우 3%정도의 에너지 효율의 향상을 가져왔다. 또한 실험에서는 네트워크에 데이터 전송량에 따라서 전력 관리가 가능한 것을 볼 수 있었다.

본 논문은 모의실험과 실측을 통해 에너지 효율의 향상을 보였으나 데이터의 응답속도 면에서는 정량적으로 측정 할 수가 없었다. 향후 데이터 응답속도와 비례한 에너지 효율성 개선에 관한 연구가 더 이루어 져야겠다.

참고문헌

- [1] Carlos de Morais Cordeiro외1, Ad hoc & Sensor networks, World Scientific, 2006
- [2] Khan외1, PMAC: Energy Efficient Medium Access Control Protocol for Wireless Sensor Networks, Military Communications Conference, 2006
- [3] Shanyder외4, Simulating the Power Consumption of Large Scale Sensor Network Applications, In Proceedings of the Second ACM Conference on Embedded Networked Sensor Systems (SenSys'04), 2004
- [4] 유영대, IEEE 802.15.4 무선 센서 네트워크에서 MAC 프로토콜 전력소모량 분석, 충북대, 2006
- [5] 황호영외4, 무선 센서 네트워크에서의 에너지 효율적인 MAC 프로토콜에 대한 동향 및 분석, 한국과학기술원, 2004
- [6] G. Grube and H. Ahmadi, "The Role of Ad-hoc Technology in the Broadband Wireless Networks of the Future," Wireless Communications, John Wiley & Sons Ltd, pp. 133-154, Jan. 2007.
- [7] IEEE, "IEEE Std. 802.15.4", 2006
- [8] www.tinyos.net
- [9] www.cs.tcd.ie/~carbajor/powerlosssimz/index.html
- [10] www.zigbee.org
- [11] www.ieee802.org/15/pub/TG4.html
- [12] www.ieee802.org/15/pub/TG4a.html
- [13] Zheng외2, PMAC: An adaptive energy-efficient MAC protocol for Wireless Sensor Networks, Parallel and Distributed Processing Symposium, Proceedings. 19th IEEE International, 2005
- [14] Shanyder외4, Simulating the Power Consumption of Large Scale Sensor Network Applications, In Proceedings of the Second ACM Conference on

Embedded Networked Sensor Systems (SenSys'04), 2004

[15] W.Ye외2, An energy efficient MAC protocol for wireless sensor networks, Proc. IEEE INFOCOM 2002, Vol. 3, June 2002

[16] T.van Dam외1, An adaptive energy efficient MAC protocol for wireless sensor networks, Proc. ACM Sensys, 2003

[17] 김학용외4, 근거리 위치확인 기술동향, 주간기술동향 통권 1322호, 2007

[18] 서성운, 센서 네트워크에서의 효율적 에너지 관리를 위한 클러스터링 알고리즘, 광운대, 2006

[19] 황호영외4, 무선 센서 네트워크에서의 에너지 효율적인 MAC 프로토콜에 대한 동향 및 분석, 한국과학기술원, 2004

[20] 박상하, 저전력 소모를 위한 비콘 비가용 지그비 센서 네트워크의 성능 개선 및 구현. 경희대, 2008

[21] 유영대, IEEE 802.15.4 무선 센서 네트워크에서 MAC 프로토콜 전력소모량 분석, 충북대, 2006

[22] 유영대외2, IEEE 802.15.4 무선 스타 센서 네트워크에서 비콘 신호 주기에 따른 센서 노드 전력소모량 분석, 한국통신학회 Vol.31, No.9B, 2006

[23] 정성훈외1, IEEE 802.15.4 and Zigbee Protocol : 유비쿼터스 센서 네트워킹을 위한 Active RFID 기술, 한국통신학회지 Vol.21, No.6, 2004

[24] 김동원외2, 무선센서네트워크에서 멀티채널 매체접근제어 방식, 한국통신학회 Vol.34, No.2, 2009

[25] 이주아외2, 무선 센서 네트워크를 위한 향상된 센서 MAC 프로토콜, 한국통신학회 Vol.31, No.11A, 2006

[26] 변강호외2, 무선 센서 네트워크를 위한 효율적인 프리엠블 샘플링 기법을 사용하는 비동기 MAC의 성능 분석, 대한전자공학회 Vol.45, No.1, 2008

[27] 조인휘, 이동 애드혹 망에서 Qos 지원을 위한 예약 기반의 MAC 프로토콜, 한국통신학회 Vol.31, No.10B, 2006

[28] 강창남외4, Ad hoc 네트워크에서 전력 제어 기법을 사용한 Dual-Channel MAC 프로토콜, 한국방송공학회, 2007

- [29] 조경탁외1, 센서 네트워크를 위한 멀티 홉 MAC 프로토콜, 한국통신학회 Vol.34, No.6, 2009
- [30] 윤찬영, 무선센서 네트워크에서 에너지 효율적인 S-MAC 프로토콜, 한국통신학회 Vol.33, No.2, 2008
- [31] 김남희외1, 무선망에서 실시간 트래픽을 위한 QoS 향상 기법, 한국콘텐츠학회 Vol.8, No.6, 2008
- [32] 이병문외2, MAC프로토콜의 전력 효율성 분석, 한국정보기술학회, 2008

감사의 글

직장생활을 하다 다시 공부를 시작한 것이 얼마 되지 않은 것 같은데 벌써 2년이 넘는 시간이 흘러 졸업을 앞두고 있다는 것이 믿어지지 않습니다. 이제 세상에 나의 생각을 알리는 첫걸음을 시작하고 그 방법을 배워서 한편으로는 뿌듯하며 그동안 실험실에서 생활하면서 학문적으로나 인간적으로나 깨달았던 모든 것들이 저에게는 고마움으로 기억되고 있습니다. 석사 논문과 졸업이라는 하나의 마침표를 찍고 다음 단계로 나아갈 수 있도록 지켜보아 주시며 많은 격려와 용기를 북돋아 주신 주위 모든 분들에게 감사의 말씀을 드립니다.

먼저 인자함으로 저를 감싸주시며 부족한 저를 받아주시고 학문의 길로 이끌어 주신 이준 교수님께 깊이 감사드립니다. 바쁘신 와중에도 부족한 저의 논문을 심사 해주시고 조언을 해주신 김충원 교수님, 김판구 교수님께도 진심으로 감사드립니다. 대학원 과정 동안 저에게 많은 가르침을 주신 컴퓨터공학과 모든 교수님들께 머리 숙여 감사드립니다.

이 논문이 있기까지 항상 따뜻한 관심과 조언을 아끼지 않으신 안성수 교수님, 송기범 박사님, 신명숙 박사님, 홍성표 박사님, 이철승 박사님, 김국세 선생님께 깊이 감사드립니다. 또한 잡다한 일뿐만 아니라 굷은 부탁에도 불평 없이 잘 도와준 학부생 최명호, 김민겸군에게도 고마움을 전합니다.

대학원 과정동안 끼니 잘 챙겨주며 인생의 조언을 아끼지 않으신 이금분 선생님, 정유선 선생님, 박정진 선생님께도 감사드립니다. 대학원 생활동안 외롭지 않게 해 준 노창오, 이재철, 장창근군에게도 감사드립니다. 멀리서 항상 응원해주신 보건대 이기준 교수님, 신재룡 교수님, 지금은 서울에 있는 친구 곽희준군에게도 감사드립니다.

언제나 곁에서 모든 것을 참고 이해하며, 나를 위해 헌신해주는 사랑하는 아내 선남이와 딸 하원이에게 감사하며 남편으로서 아빠로서 부끄럽지 않는 모습을 보여줄 것을 다짐합니다. 타지에서 응원을 아끼지 않은 동생 미진이와 여서방에게도 감사함을 전합니다.

그리고 누구보다도 제가 지금 이 자리에 있기까지 못난 자식을 위해 밤낮을 노심초사하시며 지내신 부모님께서는 무어라 말할 수 없는 죄송함과 책임감에 사죄의 마음과 감사함으로 이 논문을 바칩니다.

2009년 12월
정문수 올림

저작물 이용 허락서

학 과	컴퓨터공학과	학 번	20087110	과 정	석사
성 명	한글 정문수	한문 鄭文洙	영문 Jeong Moon Soo		
주 소	광주광역시 동구 소태동 진양아파트 2동 603호				
연락처	E-mail : moonshu@nate.com				
논문제목	한글 저전력 소모를 위한 무선센서 네트워크 MAC 프로토콜 성능 개선				
	영문 Performance Improvement of Wireless Sensor Network MAC Protocols for Low-Power Consumption				

본인이 저작한 위의 저작물에 대하여 다음과 같은 조건 아래 조선대학교가 저작물을 이용할 수 있도록 허락하고 동의합니다.

- 다 음 -

1. 저작물의 DB구축 및 인터넷을 포함한 정보통신망에의 공개를 위한 저작물의 복제, 기억장치에의 저장, 전송 등을 허락함.
2. 위의 목적을 위하여 필요한 범위 내에서의 편집과 형식상의 변경을 허락함. 다만, 저작물의 내용변경은 금지함.
3. 배포·전송된 저작물의 영리적 목적을 위한 복제, 저장, 전송 등은 금지함.
4. 저작물에 대한 이용기간은 5년으로 하고, 기간종료 3개월 이내에 별도의 의사 표시가 없을 경우에는 저작물의 이용기간을 계속 연장함.
5. 해당 저작물의 저작권을 타인에게 양도하거나 출판을 허락을 하였을 경우에는 1개월 이내에 대학에 이를 통보함.
6. 조선대학교는 저작물 이용의 허락 이후 해당 저작물로 인하여 발생하는 타인에 의한 권리 침해에 대하여 일체의 법적 책임을 지지 않음.
7. 소속 대학의 협정기관에 저작물의 제공 및 인터넷 등 정보통신망을 이용한 저작물의 전송·출력을 허락함.

동의여부 : 동의() 반대()

2010년 2월

저작자: 정 문 수 (인)

조선대학교 총장 귀하