

# ZigBee를 이용한 무선 네트워크 환경에서의 정밀 시각 동기 기법 설계 및 구현

정회원 조 현 태\*, 준회원 손 상 현\*, 종신회원 백 윤 주\*°

## Design and Implementation of Precision Time Synchronization in Wireless Networks Using ZigBee

Hyuntae Cho\* *Regular Member*, Sanghyun Son\* *Associate Member*,  
Yunju Baek\*° *Lifelong Member*

### 요 약

시각 동기 기술은 초고속 통신, 병렬/분산 처리 시스템 등의 근간이 되는 기술이다. 최근 유비쿼터스 컴퓨팅에 대한 관심도가 증가하고, 요구하는 응용이 다양해짐에 따라 무선 환경에서 높은 정밀도를 유지하는 시각 동기 기술을 요구되어지고 있다. 본 논문에서는 이러한 요구사항을 만족시키기 위해서 ZigBee를 이용한 무선 네트워크에서의 정밀 시각 동기 기법을 설계하고 구현한다. 시각 동기 과정 중 프로토콜 계층에서 발생하는 지연을 최소화하고, 메시지 송수신 과정에서 발생하는 에러 요인을 분석 및 제거함으로써 정밀 시각 동기를 유지한다. 또한, 본 논문에서는 시각 동기 시스템을 검증하고, 성능평가를 수행하였다. 성능평가 결과, 네트워크 내의 노드는 참조 시각원으로부터 50나노초 이하의 시각 동기를 이루었다.

**Key Words :** Time Synchronization, Clock Synchronization, High Precision, ZigBee, IEEE1588

### ABSTRACT

Time synchronization is essential for a number of network applications such as high speed communication and parallel/distribution processing systems. As the era of ubiquitous computing is ushered in, the high precise time synchronization in wireless networks have been required in. This paper presents the design and the implementation of the high precision time synchronization in wireless networks using ZigBee. To achieve high precision requirements, we have tried to analyze and reduce error factors such as the latency and jitters of a protocol stack on wireless environments. In addition, this paper includes some experiments and performance evaluations of our system. The result is that we established for nodes in a network to maintain their clocks to within a 50 nanosecond offset from the reference clock.

### I. 서 론

최근 들어 유비쿼터스 컴퓨팅에 대한 관심이 높아지고, 이와 관련된 환경 구축 및 연구가 급속도

로 증가하고 있다. 이러한 유비쿼터스 컴퓨팅의 핵심은 무선 네트워크 기술에 있으며, 네트워크 내의 다수의 노드들이 동시에 수행되는 여러 프로세스로부터 정확한 결과를 얻기 위해 시각 동기화가 요구

\* “이 논문은 2008년 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임”(지방연구중심대학육성사업/차세대물류IT기술연구사업단)

\* 부산대학교 컴퓨터공학과 (marine@pusan.ac.kr, hyun0427@embed.re.kr, yunju@pusan.ac.kr) (° : 교신저자)  
논문번호 : KICS2007-12-534, 접수일자 : 2007년 12월 04일, 최종논문접수일자 : 2008년 4월 4일

된다. 네트워크 내의 노드들은 범용 컴퓨터 클럭과 마찬가지로 내부 오실레이터로 수정 발진자를 사용한다. 수정 발진자는 휨(skew)과 표류(drift)의 특성을 가지며, 보통의 경우 초당 1~100마이크로초의 오차를 갖는다<sup>[1]</sup>. 이러한 상황에서 네트워크 내의 노드들간의 시각 오차는 시간이 지남에 따라 커지게 된다. 장기적인 운용이나 특수한 목적이 있을 때에는 사용시 충분한 고려가 필요하며, 주어진 환경내에서 정확한 동기를 맞출 수 있는 시각 동기 기법의 개발이 필요하다. 현재 NTP(network time protocol: 네트워크 시각 동기 프로토콜)<sup>[2]</sup>는 인터넷 분산 환경 하 네트워크 시각 동기 기법 표준으로 자리 잡은 잡았다. 하지만, NTP는 전통적인 데이터 네트워크상에서 동작하며, 수 밀리초 ~ 수십밀리초의 동기 오차를 가진다. 시각 동기 기법에서 참조 시각(reference clock)과의 지역 시각(local clock) 차이는 응용의 해상도를 결정짓는 중요한 요소가 된다. 무선 주파수의 전달 속도(ToF: time of flight)를 이용하여 위치를 측정하는 시스템<sup>[3-4]</sup>의 경우, 1마이크로초의 오차는 300미터의 거리 오차를 가지게 한다. 시각 동기 해상도가 높으면 다양한 응용에서 정확한 정보를 얻을 수 있다. 네트워크 내에서 마이크로초 이하의 정확한 시각 동기를 이루기 위해 IEEE 1588 PTP(precision time protocol)<sup>[5]</sup> 표준이 제정되었다. IEEE 1588은 대부분 이더넷 환경에서 연구 개발<sup>[6]</sup>이 진행되어 왔으며, 최근 WLAN(wireless LAN)을 기반으로 IEEE 1588을 구현<sup>[7-8]</sup>하는 연구가 미묘하게 진행되고 있다. 하지만, LR-WPAN (low rate wireless personal area network)환경에서 정밀 시각 동기를 유지하는 연구는 전무한 실정이다.

본 논문에서는 LR-WPAN 환경에서 마이크로초 이하의 정밀한 시각 동기를 이루기 위해 ZigBee 네트워크를 활용한 정밀 시각 동기 기법을 설계 및 구현한다. ZigBee/IEEE 802.15.4는 LR-WPAN을 위한 통신 프로토콜 명세를 포함하고 있으며, 대부분의 응용들은 낮은 데이터 전송율, 배터리 수명의 최대화, 안전한 네트워킹을 목적으로 한다. 이러한 ZigBee 네트워크하에서 정밀 시각 동기를 이루기 위해서는 풀어야 할 난제들이 있다. 먼저, 시각 동기과정 중 프로토콜 계층에서 발생하는 지연을 최소화하고, 메시지 송수신 과정에서 발생하는 에러 요인을 분석 및 제거하여야 한다. 또한 ZigBee 트랜시버(transceiver)

를 사용함에 있어 발생하는 오차의 요인을 분석하여 이를 시각 동기 기법에 적용하여야 한다. 설계 및 구현된 시각 동기 기법은 동기 과정에 발생하는 오류 요인을 줄임으로서 50나노초 이하의 시각 동기를 가진다.

본 논문의 구성은 다음과 같다. II장에서는 시각 동기 기법에 대한 관련연구를 소개하고, III장에서는 ZigBee를 이용한 정밀 시각 동기 기법에 대해서 기술한다. IV장에서 정밀 시각 동기 기법에 대해서 성능평가를 수행하고 V장에서 결론을 맺는다.

## II. 관련연구

본 장에서는 정밀 시각 동기를 이루기 위해 시각 동기과 관련한 기존 연구들에 대해서 살펴보고, 기존의 연구들이 가지는 한계점들을 비교 분석한다.

표 1은 시각 동기 기법에 대한 비교를 포함하고 있다. NTP는 오랜 기간 동안 인터넷 환경아래에서의 시각 동기 표준으로 자리 잡았지만 정밀도는 수 밀리초 정도에 불과하여, 밀리초 이하의 높은 정밀도를 제공하는 시각 동기 기법에는 적합하지 않다. GPS<sup>[9]</sup>는 UTC(universal coordinated time)에 동기화하면서 마이크로초 이하의 정밀도를 제공한다, 하지만, 수신기가 비싸며, 실외용 안테나가 필요하다는 단점이 있다. IEEE 1588은 데이터 네트워크 환경에서 마이크로초 이하의 높은 정밀도를 제공한다. 하지만, 높은 정밀도 제공을 위해 하드웨어적 접근법 필요하며, 대부분의 이더넷을 중심으로 연구 개발이 진행되어 왔다.

다음 장에서는 IEEE 1588을 기반으로 하여 ZigBee 네트워크에서 정밀 시각 동기 기법을 이루기 위한 분석을 수행하고, 이를 토대로 시각 동기 하드웨어와 알고리즘을 설계하고 구현한다.

표 1. 시각 동기 기법 비교  
Table 1. Comparison of Time Synchronization Methods

	NTP	GPS	IEEE 1588
Spatial extent	Wide area	Wide area	Local area
Communications	Internet	Satellite	Network
Target accuracy	Few milliseconds	Sub micro-second	Sub micro-second
communication style	Peer-to-Peer	Client/server	Master/slave
Hardware requirement	No	RF receiver and processor	For highest accuracy

### III. 정밀 시각 동기 기법

네트워크 내에는 참조 시각을 제공하는 마스터 노드와 마스터 노드로부터 참조 시각을 전송받아 마스터 클럭에 동기시키는 슬레이브 노드가 있다. 기본적으로 이들 간의 시각 동기는 메시지 교환을 통하여 이루어진다. 즉, 마스터 노드는 시각 동기 프로토콜에서 자신의 클럭 값을 읽은 후, 동기 메시지에 포함시켜 무선 주파수를 통하여 네트워크 내에 존재하는 슬레이브 노드들에게 전달한다. 시각 동기 메시지를 수신한 슬레이브 노드는 자신의 클럭을 시각 동기 메시지내의 클럭에 동기화 시킨다. 이때, 클럭을 동기화하는 방법은 논리적인 시각을 보정하는 방법과 물리적인 시각을 보정하는 방법으로 나눌 수 있다. 논리적인 시각 보정 기법은 자신의 누적 클럭 시각 차이만큼 소프트웨어적으로 더하거나 빼주는 방법이며, 물리적인 시각 보정 방법은 클럭 소스인 오실레이터를 가속시키거나 천천히 동작하도록 하는 방법이다. 본 논문에서는 논리적인 시각 보정 기법을 적용하여 노드들간의 시각을 보정한다.

#### 3.1 타임 스탬프 지점의 결정

시각 동기 과정 중 시각을 결정하는 것을 타임 스탬프(time stamp)라 한다. 시각 동기 프로토콜에서 타임 스탬프를 하는 지점은 중요하며, 시각 동기 정밀도에 영향을 미친다. 응용 계층에서 타임 스탬프를 수행하게 되면 하위 계층들에서 전송지연 및 지터(jitter)를 발생시킨다. 이러한 불확도(uncertainty)는 시각 동기 정밀도에 여러 요인으로 작용한다<sup>[10]</sup>. 따라서 시각 동기 프로토콜에서의 타임 스탬프는 프로토콜 스택의 하위 계층에서 수행하는 것이 동기 오차를 제거하는 좋은 방법이다. 한 노드에서 메시지를 읽어올 수 있는 하위 계층은 MAC 계층이다. 그림 1의 (a)는 송신시 MAC 계층에서 타임 스탬프 하는 과정을 보여준다. 먼저 시각 동기 프로토콜에서 시각 동기 메시지를 생성한다. 생성된 메시지는 하위 프로토콜 계층으로 전송된다. 그 다음 MAC 계층에서 현재의 시간을 읽어온 후, 이를 패킷내에 캡슐화 시킨다. 이렇게 생성된 패킷은 MAC 계층을 거쳐, 무선 주파수로 변조되어 전송된다. 이때, MAC 계층에서 채널이 혼잡하거나 충돌이 발생하게 되면, 전송지연의 불확도는 증가하게 되며, 결과적으로 시각 동기 정밀도를 떨어뜨린다<sup>[7-8, 10]</sup>.

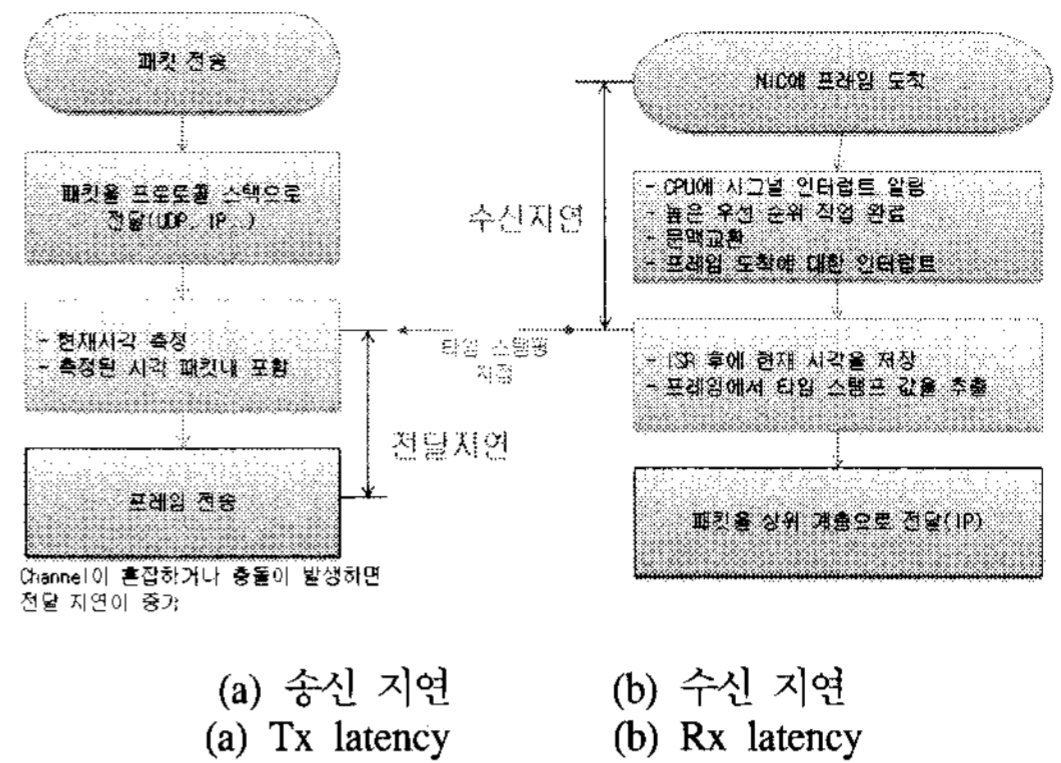


그림 1. 메시지 송수신시의 지연 및 지터  
Fig. 1. delay and jitter during message exchange

(b)의 그림과 같이 메시지 수신시 타임 스탬프를 하는 과정에서도 불확도는 존재한다. RF 리시버(receiver)는 프레임을 받으면, CPU에게 인터럽트 신호를 전송한다. CPU는 우선 순위가 높은 작업을 처리하고, 인터럽트를 처리하기 위해 문맥교환(context switching)등의 작업을 수행한다. 그리고 수신한 메시지의 작업을 처리하게 된다. 실제로 이 서비스 루틴(interrupt service routine)까지 어느 정도의 지연이 발생하는지는 알지 못하고 불확도로 존재한다. ISR 이후, 노드의 현재 시간을 저장하고 프레임에서 타임 스탬프 값이 추출한다. 시각 동기 프로토콜은 이 값을 이용하여 오프셋을 계산하고, 동기화 알고리즘을 구동하게 된다. 프레임의 도착 시점과 타임 스탬프 지점 사이의 지연(latency)은 커널의 활동량에 따라 유동적이게 된다.

정밀 시각 동기 프로토콜을 구현하기 위해서는 메시지 송수신시 수행되는 타임 스탬프의 송수신 지연의 불확도를 제거해야 한다. 본 논문에서는 이러한 타임 스탬프의 불확도를 제거하기 위해 MAC과 PHY(물리계층)사이의 MII(media independent interface) 수준에서 타임 스탬프하도록 한다<sup>[10]</sup>. MII는 프로세서와 RF 트랜시버와는 독립적인 부분이므로, 이 수준에서 타임 스탬프하기 위해서는 별도의 하드웨어가 필요하다. 본 논문에서는 이러한 추가의 하드웨어를 개발하고, 개발된 하드웨어 플랫폼 상에서 정밀 시각 동기 프로토콜 알고리즘을 탑재한다.

#### 3.2 시각 동기 하드웨어 설계

시각 동기화를 위해 설계된 하드웨어는 프로세싱 유닛(processing unit), 통신 유닛(communication unit), 그리고 타임 프로세싱 유닛(time processing

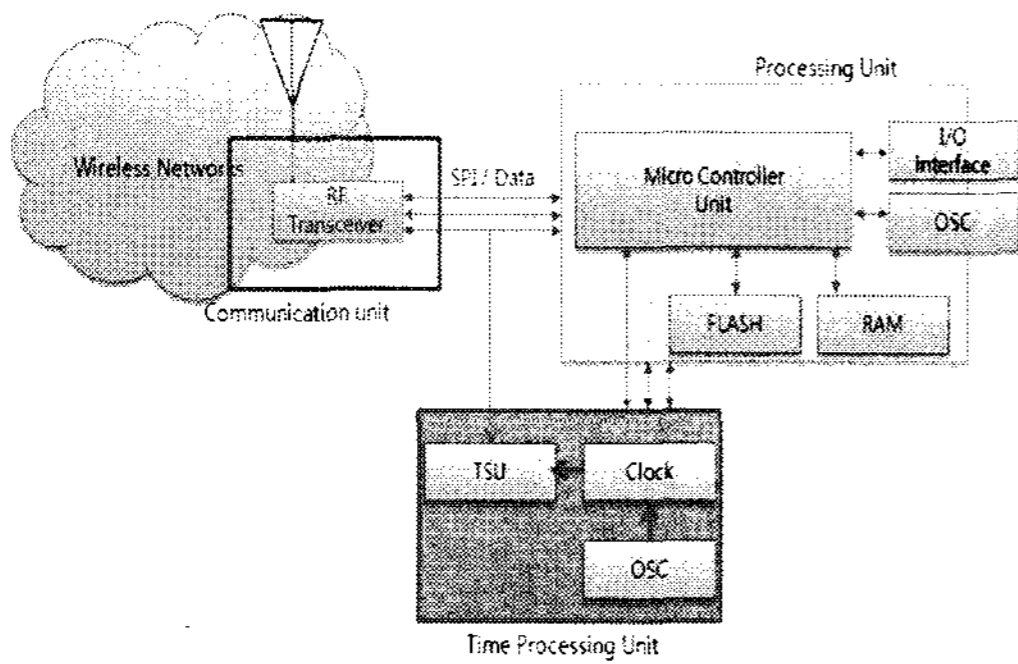


그림 2. 시각 동기 시스템 하드웨어 구조  
Fig. 2. Hardware architecture for the time synchronization system

unit)으로 구성된다. 프로세싱 유닛은 시각 동기 프로토콜이 탑재되어 노드들간의 시각 동기화를 구동하는 역할을 수행하며, 통신 유닛은 2.4GHz ZigBee 무선 통신을 통해 시각 동기 메시지를 송수신하는데 사용된다. 그리고 타임 프로세싱 유닛은 MII 수준에서 타임 스탬프를 수행하고, 시각 동기 필요한 클럭을 제공한다. 읽혀진 타임 스탬프 값은 프로세싱 유닛으로 전달되어 동기화 알고리즘을 구동하는데 이용된다. 그림 2는 시각 동기 하드웨어의 구조를 보여준다.

프로세싱 유닛을 위해서 Freescale사의 MCF5235<sup>[11]</sup>를 사용하였다. MCF5235는 32비트 RISC 프로세서 코어를 내장하고 있으며, 최대 150MHz로 동작하며 버스 주파수는 최대 75MHz이다. 타임 프로세싱 유닛은 MII 수준에서 타임 스탬프를 수행하고, 시각 동기 필요한 정밀한 클럭을 제공해야 하므로, 고속 및 정확하게 동작하여야 한다. Kannisto<sup>[7]</sup>는 고속 타임 스탬프를 위하여 PLD(programmable logic device)사용하였으며, Weibel<sup>[10]</sup> 역시 FPGA(field-programmable gate array)를 사용하여 하드웨어적으로 고속의 타임 프로세싱 유닛을 설계하였다. PLD 또는 FPGA의 추가 장착은 개발의 시간 및 비용의 상승을 초래한다. 본 논문에서는 이러한 문제를 해결하기 위해 MCF5235가 포함하고 있는 eTPU(enhanced time processor unit)를 타임 프로세싱 유닛으로 사용한다. eTPU는 32개의 채널을 가진 이벤트 트리거 VLIW 프로세서이며, 24비트의 타이머 해상도를 가진다. 6KB의 코드 메모리와 1.5KB 데이터 메모리를 내장하고 있다. eTPU의 특징은 외부 이벤트가 발생한 시점을 하드웨어적으로 내부 타이머로 기록을 해주기 때문에 전송 지연이

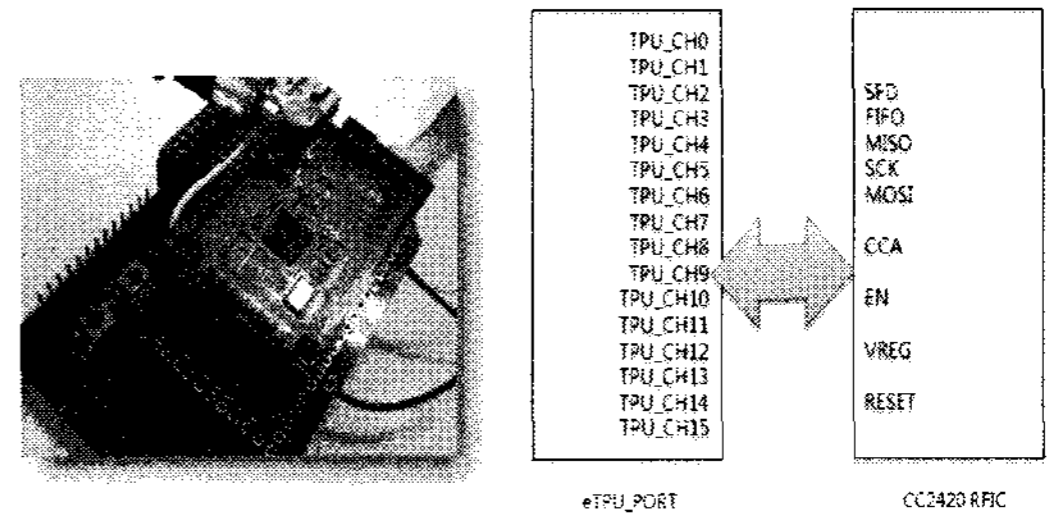


그림 3. 하드웨어 외형 및 인터페이스  
Fig. 3. Hardware prototype and interface

거의 없다. 통신 유닛으로는 TI사의 CC2420트랜시버<sup>[12]</sup>를 사용하였다. CC2420은 2.4GHz 주파수 대역용 RF 트랜시버이며, IEEE 802.15.4 및 ZigBee의 PHY를 준수한다. CC2420제어를 위해 eTPU중 일부를 I/O로 사용하였다. 프로세싱 유닛과 통신 유닛은 QSPI를 사용하여 메시지를 주고받으며, 타임 프로세싱 유닛은 프로세싱 유닛과 메모리를 공유한다. 그림3은 MCF5235와 CC2420의 인터페이스 및 시각 동기 하드웨어 모듈을 보여주고 있다.

ZigBee 트랜시버를 이용하여 시각 동기 메시지를 송수신하기 위해서는 ZigBee의 PHY계층의 프레임 구조를 먼저 알아야 한다. ZigBee PHY에서는 상위 계층으로부터 메시지를 받으면 4바이트의 프레임 임블과, 1바이트의 SFD(start of frame delimiter), 그리고 1바이트의 프레임 길이를 생성한 후에, MAC 프로토콜 데이터 유닛(MPDU)를 전송한다. 이중 SFD의 마지막 비트가 전송되고 나면, 통신 유닛은 SFD 핀의 신호를 상승에지(rising edge)를 만든다. 그림 4는 메시지 송수신 과정 중 SFD 신호를 타임 프로세싱 유닛이 감지하는 시점을 보여준다.

시각 동기 하드웨어에서는 프레임의 시작을 알리는 SFD 신호를 타임 프로세싱 유닛이 감지하고 그 시점을 내부 레지스터에 저장한다. 시각 동기 프로토콜은 이 시점을 이용하여 오프셋을 보정하는데 사용한다.

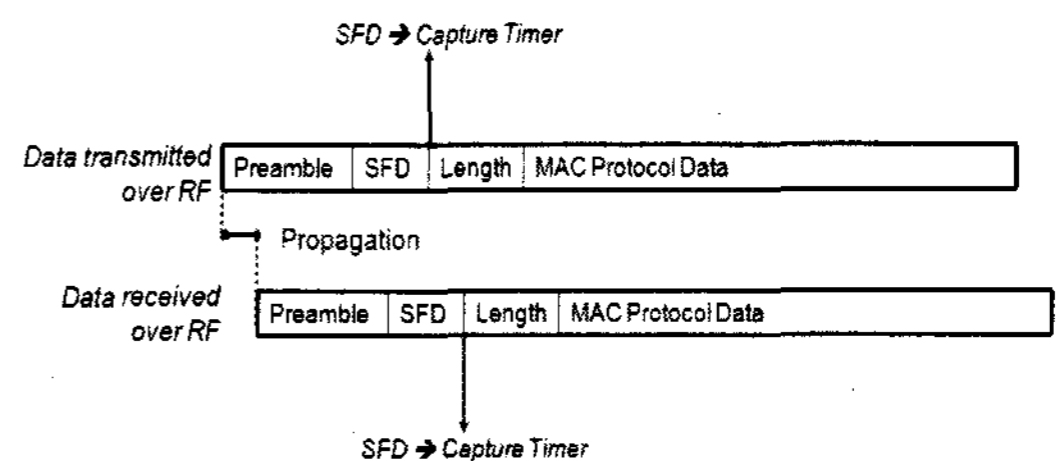


그림 4. 메시지 송수신 과정 중 타임 스탬프 시점  
Fig. 4. Time stamping point

### 3.3 정밀 시각 동기 기법

#### 3.3.1 시각 동기 알고리즘

시각 동기 프로토콜을 기본적으로 메시지 교환을 통하여 이루어진다. 메시지의 종류는 Sync, Follow\_up, Delay\_Req, 및 Delay\_Resp의 4가지로 구성된다.

- ① Sync 메시지 : 시각 동기 메시지로서 이 메시지가 MII를 통해 나간 시점이 오프셋 보정에 사용
- ② Follow\_up 메시지 : Sync 메시지를 전송할 때, SFD 신호가 상승된 시점을 타임 스탬프하여 이 메시지 내에 실어서 전송
- ③ Delay\_Req 메시지 : 슬레이브 노드가 마스터 노드와의 전송지연을 알고자 할때, 전송하는 메시지의
- ④ Delay\_Resp 메시지 : 마스터 노드가 Delay\_Req에 대한 응답으로 지연시간을 포함하여 응답하는 메시지

그림 5는 MII 수준에서 타임 스탬프하고 이를 전달하는 예를 보여준다. 마스터 노드가 Sync 메시지를 무선으로 전송하고, 전송된 시점 즉, SFD가 상승된 시점(t1)을 타임 프로세싱 유닛이 캡처하고, 타이머 값을 내부 레지스터에서 유지한다. 이는 Sync 메시지가 이미 무선을 통하여 나가고 있기 때문에, Sync 메시지 내에 실지 않고 다음 Follow\_up 메시지 내에 실어서 전송한다. 슬레이브 노드는 메시지를 받은 시점(t2)을 타임 프로세싱 유닛이 하드웨어적으로 감지하고, 자신의 로컬 클럭을 저장한다.

Sync 메시지만 가지고는 Sync 메시지가 마스터 노드로부터 전송된 시점을 알 수 없다. 이후, 마스터 노드는 실제 t1 은 Follow\_up에 실어서 전송하고, 슬레이브 노드는 Follow\_up을 받은 시점에 마스터 노드와의 오프셋을 계산할 수 있다. 보정하는 방법은 슬레이브 노드의 타이머 값과 마스터 노드의 타이머 값을 비교한 후 카운팅한 클럭 시각차이만큼 더하거나 빼준다.

마스터 노드와 슬레이브 노드간의 클럭 오프셋을 계산하였다 하더라도, 여전히 불확도는 존재한다. 메시지를 교환함에 있어, 메시지의 전송 지연은 고려되지 않았기 때문이다. 메시지의 전송지연을 구하기 위해 Delay\_Req와 Delay\_Resp 메시지를 사용한다. Delay\_Req는 슬레이브 노드가 전송하는 메시지로서 마스터 노드와의 전송지연을 계산하는데 사용된다. 슬레이브 노드가 Delay\_Req를 보내면 타임 프로세싱 유닛의 하드웨어가 t3를 계산/유지하게 되고, t4 시점에 마스터 노드가 메시지를 받으면, 이를 마스터 노드의 타임 프로세싱 유닛에서 기록하여, Delay\_Resp 메시지에 t4를 포함하여 슬레이브 노드에게 응답한다. 이 응답 메시지를 받은 슬레이브 노드는 그림 5와 같이 t3와 t4의 차이를 계산하여 전송지연(delay)을 계산해 낼 수 있다.

#### 3.3.2 시각 표현 에러

시각 동기 프로토콜의 핵심중 하나인 타임 프로세싱 유닛은 논리 클럭을 사용하여 시각을 동기화

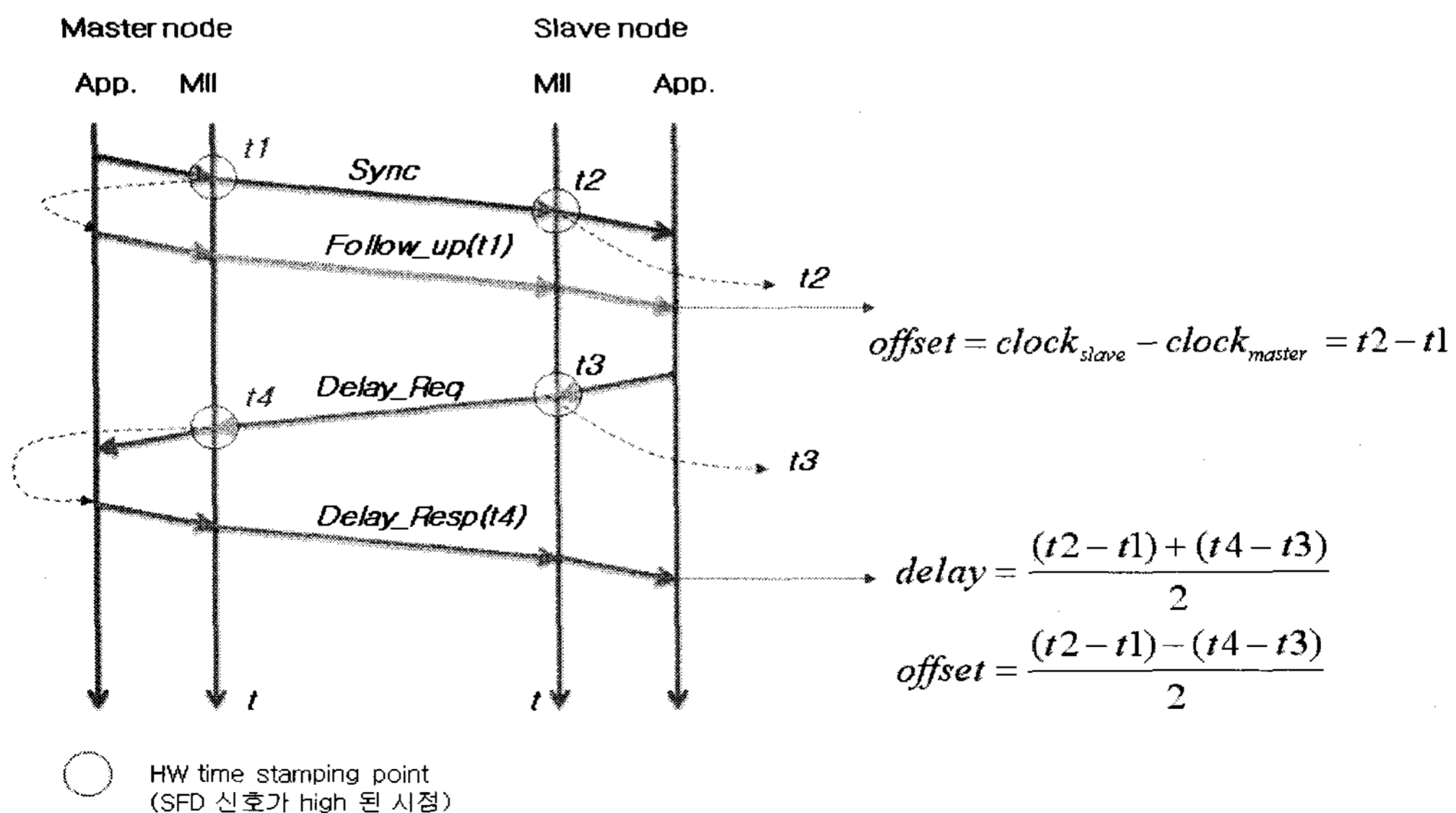


그림 5. 하드웨어 타임 스탬프를 이용한 메시지 교환  
Fig. 5. Message exchange by hardware assistant time stamping

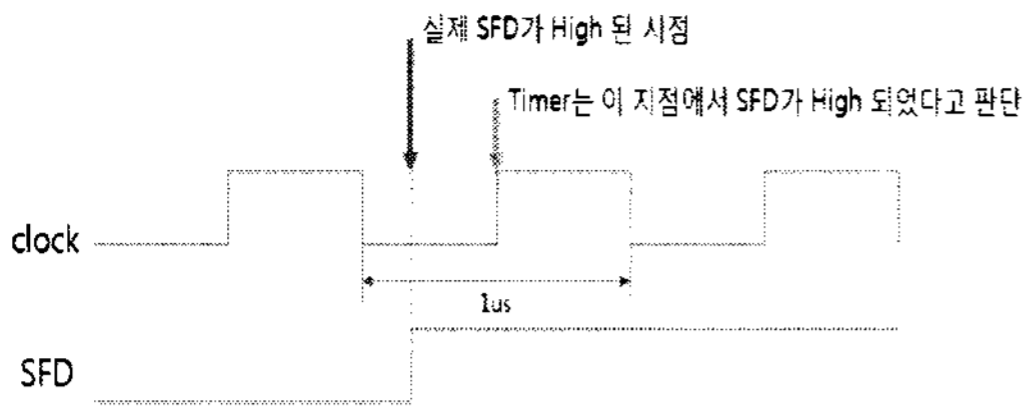


그림 6. 시각 표현 에러의 예  
Fig. 6. Example of time representation error

한다. 논리 클럭 동기의 가장 큰 단점은 이벤트가 발생한 그 시점을 정확히 판단하지 못한다는 것이다. 그림6과 같이 타임 프로세싱 유닛의 타이머가 1MHz로 동작한다고 가정하자. 클럭 간의 시간 차이는 1마이크로초가 된다. 즉, 매 1마이크로초마다 상승 또는 하강 에지를 반복하게 된다. 디지털 시스템에서는 이러한 상승 또는 하강 에지에만 타이머를 기록하게 된다. 그림 6과 같이 하강에지와 상승에지 사이에 SFD 신호가 상승하면, 정확히 이 시간을 기록하지 못하고, 시스템 클럭의 가장 가까운 다음 상승에지 시에 감지하고, 그 시점을 타임 스탬프하게 된다. 이러한 1MHz환경에서 최대 오류는 1마이크로초의 타임 스탬프 에러를 가져올 수 있다. 이러한 시각 표현 에러를 제거하는 것은 어려우며, 정밀한 시각을 동기화하기 위해서는 높은 해상도의 타이머를 유지하는 것이 좋다. 본 논문에서는 37.5MHz(클럭당 26.7나노초)의 해상도를 가지며, 보다 높은 해상도를 요구하는 응용에서는 높은 주파수의 클럭을 사용함으로써 타임 스탬프 오차를 줄일 수 있다.

### 3.3.3 표류(drift) 보정

프로세서의 클럭은 수정 발진자의 미리 정의된 주파수에 따라 일정 속도로 동작하도록 되어 있다. 그러나, 수정 발진자는 온도, 습도 등의 환경적 요인으로 인해 발진 주기가 일정하지 않다. 따라서, 각 프로세서의 수정 발진자들이 정해진 주파수대로 동작하지 않는다. 다시 말해서, 임의의 서로 다른 두 클럭은 다른 발진 주기를 가지게 된다. 같은 값으로 초기 시각이 설정되었을 경우라도 발진 주기 차가 일정 기간 누적될 경우, 두 클럭에 의해 유지되는 시각 차를 유발한다. 예를 들어, 1PPM(parts per million)을 가지는 수정 발진자가 있다면 이는 초당 1마이크로초 표류를 가진다. 일반적으로 쉽게 구할수 있는 발진자의 경우 20PPM 에서 50PPM 사

이이며, 이러한 표류율로 인해 오차는 시간이 갈수록 오차는 누적된다. 표류율 보정을 위해 본 논문에서는 온도나 수명등의 환경적 영향을 많이 받는 값싼 수정 발진자를 사용하지 않고 정확도가 높은 TCXO(temperature compensation crystal oscillator)를 사용하여 표류율을 감소시켰다. 일반적인 수정 발진자의 경우 매우 낮은 가격의 시간 기준 오실레이터를 사용하므로 보다 특성이 나은 클럭으로 교체할 경우 시각 유지 성능의 향상을 얻을 수 있다. 20PPM 오실레이터와 1.5PPM TCXO의 표류율을 비교해본 결과, 20PPM 오실레이터의 경우 평균으로부터 최대 클럭 차이가 25였으며, 표준 편차는 10.08을 나타내었다. 반면, TCXO의 경우 평균으로부터 최대 클럭 차이가 10이며, 표준 편차는 1.98클럭을 보였다.

하드웨어 클럭을 교체하였다고 하더라도 여전히 표류율에 대한 보정은 필요하다. 슬레이브 노드의 오실레이터가 마스터 노드의 오실레이터와 정확한 주파수로 동작하지 않는 문제를 해결하기 위해 본 논문에서는 주기적으로 Sync 메시지를 전달하여 편차를 보정토록 한다. 1.5PPM의 오실레이터를 사용하는 환경에서 초당 1회의 Sync 메시지를 주기적으로 전송하면 최대 1.5마이크로초의 오차가 발생할 수 있다. Sync 메시지의 전송 주기가 짧을수록 표류율은 감소하게 된다.

### 3.3.4 RF 메시지 인코딩(encoding) 및 디코딩(decoding) 지터 보정

시각 동기 메시지를 무선으로 교환하기 위해 ZigBee 네트워크를 사용하였다. 이때, ZigBee RF 트랜시버는 디지털 값을 아날로그로 아날로그를 디지털로 변복조 하는데 불확도를 가진다. 이러한 불확도를 확인하기 위해 송신자와 수신자 사이에 메시지를 주기적으로 전송하고 SFD 신호의 차이를 측정하였다. 측정결과 CC2420의 경우 약 ±0.125마이크로초 정도의 지터가 발생하였다. 이는 CC2420이 한 비트를 표현하는데 8MHz의 해상도(8Mchip/s)를 가지기 때문이다. 마이크로초 이하의 지터는 기존의 NTP와 같은 동기 프로토콜에서는 무시될 수 있는 수치이지만, 마이크로초 이하의 정밀도를 요구하는 동기 기법에서는 아주 큰 오류 요인으로 작용한다. 이러한 오류 요인을 제거하기 위해 다수의 샘플에 대해서 평균값, 중앙치, 그리고 가중치 부여 등의 방법을 적용하여 성능평가를 수행하였다.

### IV. 성능평가

#### 4.1 타임 프로세싱 유닛의 정밀도 검증

본 논문에서는 노드의 클럭을 타임 스탬프하기 위해 MCF5235의 eTPU를 탑재한 타임 프로세싱 유닛을 이용한다. 따라서, 타임 프로세싱 유닛이 실제 이벤트가 발생하는 시점을 얼마나 정확히 감지해 낼 수 있는가가 동기 기법의 핵심이 되므로, 타임 프로세싱 유닛의 정밀도를 검증하는 실험이 선행되어야 한다. 실험방법은 다음과 같다. 마스터 노드는 주기적으로 동기 메시지를 전송하고, 슬레이브 노드는 동기 메시지를 듣고 SFD신호를 출력한다. 마스터 노드와 슬레이브 노드는 메시지 송수신시 SFD 신호를 출력하게 되고, 이 신호를 오실로스코프와 타임 프로세싱 유닛에 입력으로 넣어서 마스터 노드와 슬레이브 노드 간의 SFD 신호 간격을 측정하였다. 표2는 타임 프로세싱 유닛과 오실로스코프로 비교한 메시지 송수신 전송 지연이다.

실험 결과에서 보는 바와 같이 타임 프로세싱 유닛을 사용하여 SFD 이벤트를 감지할 때, 오실로스코프 측정치와의 차이는 수 나노초 정도이며, 이는 마이크로초 이하의 시각 동기를 유지하는 시스템에서는 무시할 수 있는 수준이다.

표 2. 마스터 노드와 슬레이브 노드간의 SFD 신호 차이 검증  
Table 2. Verification of SFD between the master and the slave

단위 : us

	TPU	오실로스코프
평균	3.157654941	3.170031
최대	3.226666667	3.2468
최소	3.066666667	3.088
표준편차	0.039601341	0.03995667

#### 4.2 시각 동기 기법 성능평가

본 절에서는 ZigBee 네트워크를 이용한 정밀 시각 동기 기법에 대해서 성능평가를 수행한다. 실험의 방법은 다음과 같다. 참조시각을 제공하는 마스터 노드는 주기적으로 시각 동기 메시지를 전송하고, 슬레이브 노드는 이를 듣고, 마스터 노드의 클럭에 동기화 시킨다. 두 노드간 클럭 동기화 중 외부에서 동시에 이벤트를 발생시켜 각 노드에서 이벤트 감지 시점을 기록/분석하여 성능을 비교한다.

먼저, 시각 동기 메시지의 주기에 따라 두 노드간의 시각 동기가 얼마나 이루어지는지 비교하여 보았다. 마스터 노드는 주기적으로 슬레이브 노드에

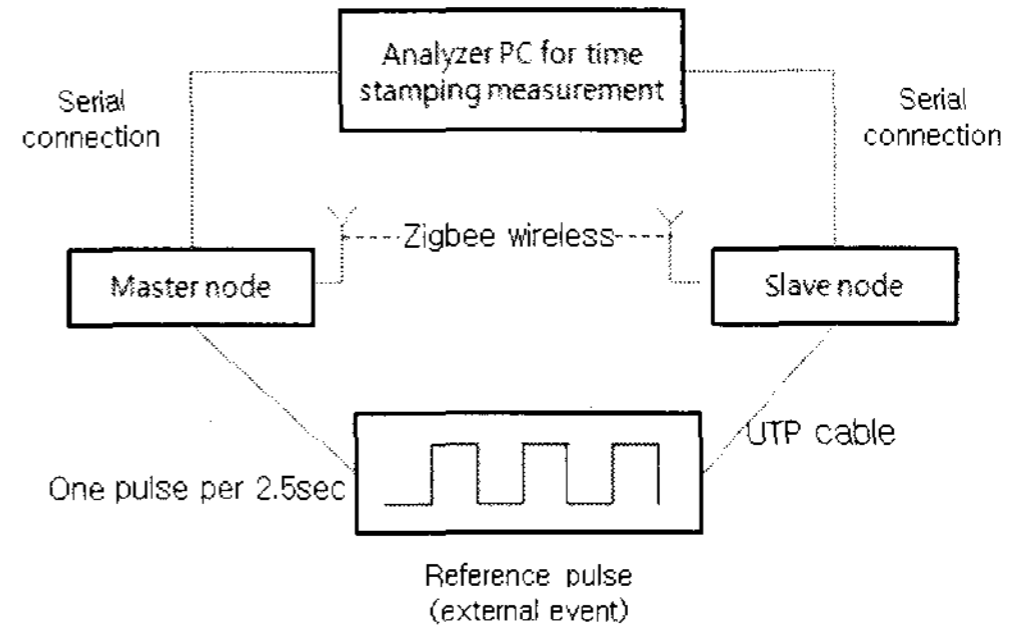


그림 7. 이벤트 감지를 이용한 시각 동기 검증 방법  
Fig. 7. Verification environment for event detection

게 동기 메시지를 전송한다. 이때, 매번 전송되는 동기 메시지는 Sync, Follow\_up, Delay\_Rep, 그리고 Delya\_Resp 메시지 4개가 하나의 셋으로 이루어진다. 즉, 한번 Sync와 Flolow\_up가 전송될 때마다, 슬레이브 노드는 Delay\_Req를 전송한다. 시각 동기 메시지의 전송주기는 매 초에 1, 4, 8, 16, 24, 32회씩 전송하도록 변경하여 실험하였다. 외부 이벤트의 발생 주기는 2.5초마다 한번씩 발생하며, 네트워크 내의 노드는 이 시점에 자신의 시각을 획득하여 비교한다. 실험의 횟수는 매 실험에 1000회 수행하였다.

그림 8은 동기 메시지 횟수에 따른 마스터 노드와의 차이를 평균과 표준편차로 나타낸 그래프이다. 평균과 표준편차가 0에 가까울수록 동기가 정확히 이루어지는 것이다. 양수인 경우는 슬레이브 노드가 마스터 노드보다 느리게 시간이 가는 것이며, 음수인 경우는 슬레이브 노드가 마스터 노드보다 빨리 시간이 간다는 의미이다.

그림 9는 그림 8의 결과에서 32회 시각 동기를 수행할 때, 마스터 노드와의 클럭 차이 히스토그램을 보여준다. 그래프에서 보는 바와 같이 마스터 노드는 약 -13.5 나노초의 평균 오차를 가지며, 표준

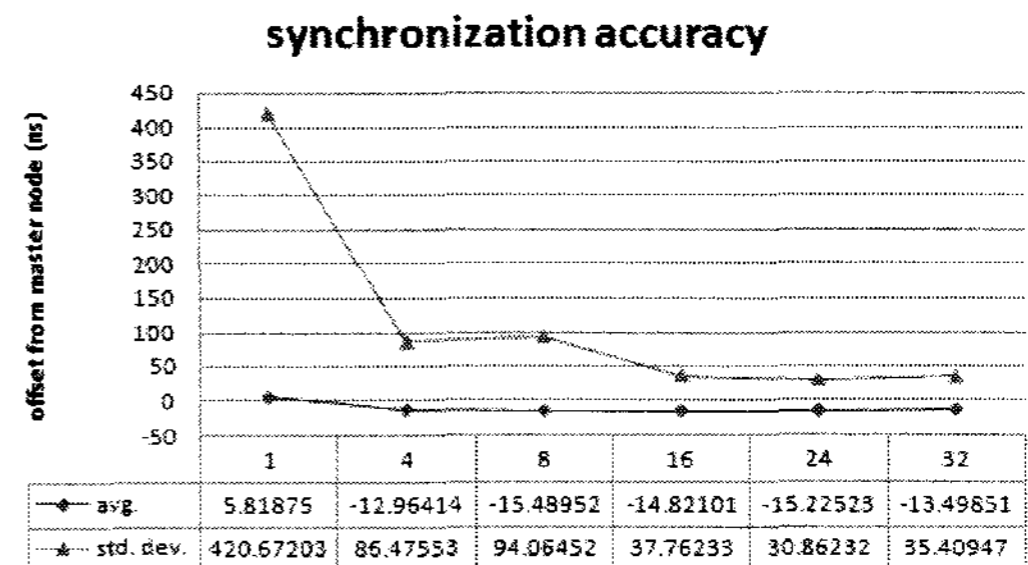


그림 8. 시각 동기 메시지 횟수에 따른 시각 정밀도  
Fig. 8. Synchronization accuracy by the number of time synchronization messages

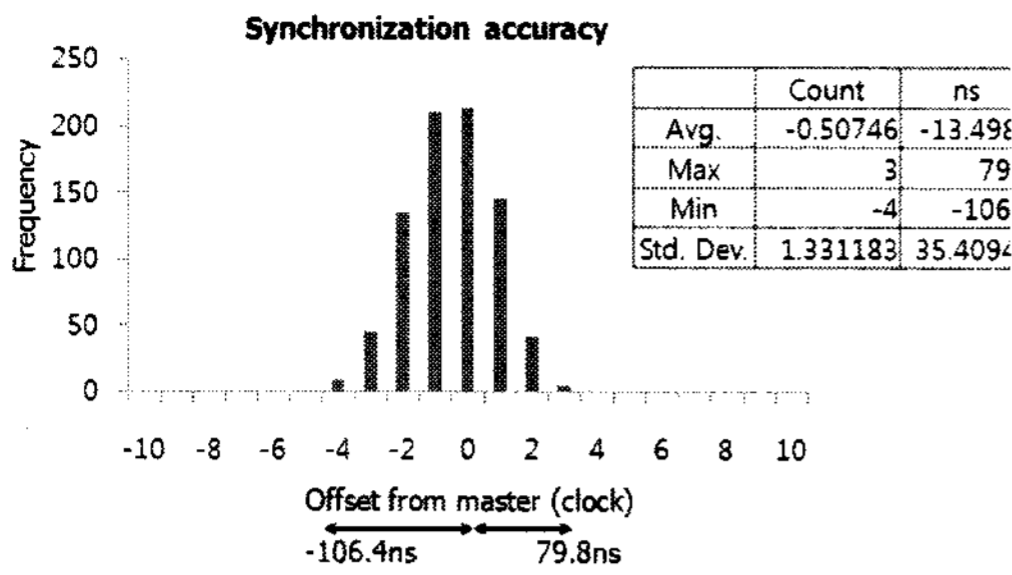


그림 9. 마스터 노드와의 클럭 차이를 보여주는 히스토그램 (매초 32회 동기)  
 Fig. 9. Clock difference between the master and the slave in case the number of the sync message is 32 per second

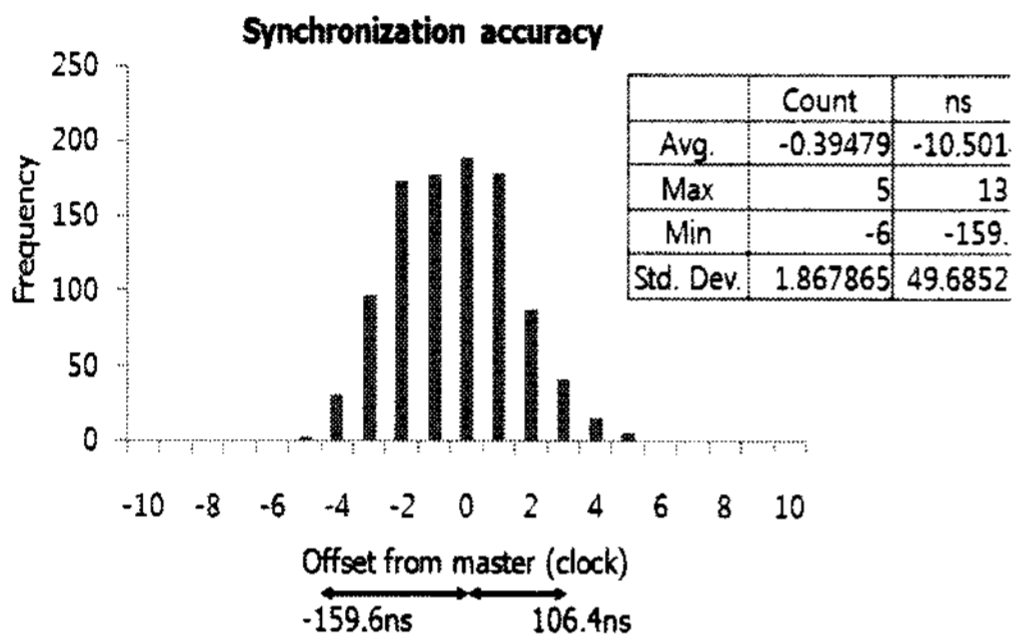


그림 10. Delay\_Req 메시지 주기 변경에 따른 시각 동기 결과 히스토그램  
 Fig. 10. Time synchronization result by the number of Delay\_Req message

편차는 35.4나노초를 나타낸다. 0의 값을 가질 경우는 마스터 노드와 슬레이브 노드와는 정확히 시각 동기가 이루어진 것이다.

그림 8과 9에서 보는 바와 같이, 싱크 메시지가 더 빈번해지면, 노드의 표류율을 줄일 수 있다. 반면, 전송 지연 요청(Delay\_Req) 메시지는 같은 거리에 있거나, 거리 차이가 많이 나지 않는다면, 전송 지연에 차이를 가지지 않게 된다. 따라서 이러한 Sync 메시지의 수는 늘리고, Delay\_Req 메시지는 줄이면, 시스템의 성능을 향상시킬 수 있을 것으로 판단된다. 본 실험에서는 Sync 메시지 주기와 전송 지연 요청(Delay\_Req) 주기를 달리 하였을 경우, 시각 동기 기법에 미치는 영향을 평가하였다. Sync 및 Follow\_up 메시지 주기는 초당 50회이며, Delay\_Req 및 Delay\_Resp 메시지 주기는 Sync/Follow\_up 메시지 20회당 1회로 설정하였다. 실험 결과 마스터 노드와의 시각 차이는 평균 약 -10.5나노초로 나타났으며, 표준편차는 약 49.68나노초이다. 평균은 이전보다 0에 더 가까워 졌지만, 간헐적인 전송 지연 요청으로

인하여 편차는 더욱 증가하였다.

타임 스탬핑 수행시 발생하는 오차와 트랜시버의 디코딩 오차를 줄이기 위하여 본 논문에서는 필터링 기법을 적용하였다. 적용한 필터링 기법은 이동평균, 중앙치 필터(9-tap median filter), 가중치 부여의 3가지를 수행하였다. 이동평균에서는 표류율을 보정하기 위하여 마스터 노드가 주기적으로 메시지를 전송하면 슬레이브 노드는 최근 10개의 메시지를 저장하고, 이를 평균하여 오프셋 보정에 사용하도록 한다. 중앙치는 최근 9개의 메시지 중 중간 값을 취하여 오프셋 보정에 이용하였다. 가중치 부여를 위해 다음의 수식(1)을 사용할 수 있다.

$$offset = (1 - \alpha) * offset_{prior} + \alpha * offset_{current} \quad (1)$$

여기서  $offset_{prior}$ 는 이전 오프셋 값이며,  $offset_{current}$ 는 현재 측정된 오프셋 값이다.  $offset_{prior}$  크기는 10이며,  $\alpha$ (alpha)값을 0.5에서 0.9까지 변경시켜 가면서 성능을 평가하였다.

그림 11은 필터링 기법의 변화에 따른 성능평가를 수행한 결과이다. 각 기법에 대해서 1000회의 외부 이벤트 신호를 발생하였고, 그에 따른 평균, 표준편차, 최대값, 최저값을 표현하였다. 기준선과 가까울수록 마스터 노드와의 동기가 정확하게 이루어짐을 나타낸다. 오프셋 차이의 평균은 이동평균을 사용한 경우가 -0.19 클럭 차이로 가장 작았으며, 표준 편차는 가중치를 0.9로 부여한 경우가 1.59 클럭 차이로 가장 작게 발생하였다.

그림 12는 이동평균에 대한 시각 동기 결과 그림 13은 가중치 0.9 부여 시의 시각 동기 결과에 대한 히스토그램을 나타낸다. 결과적으로, 이동평균의 경

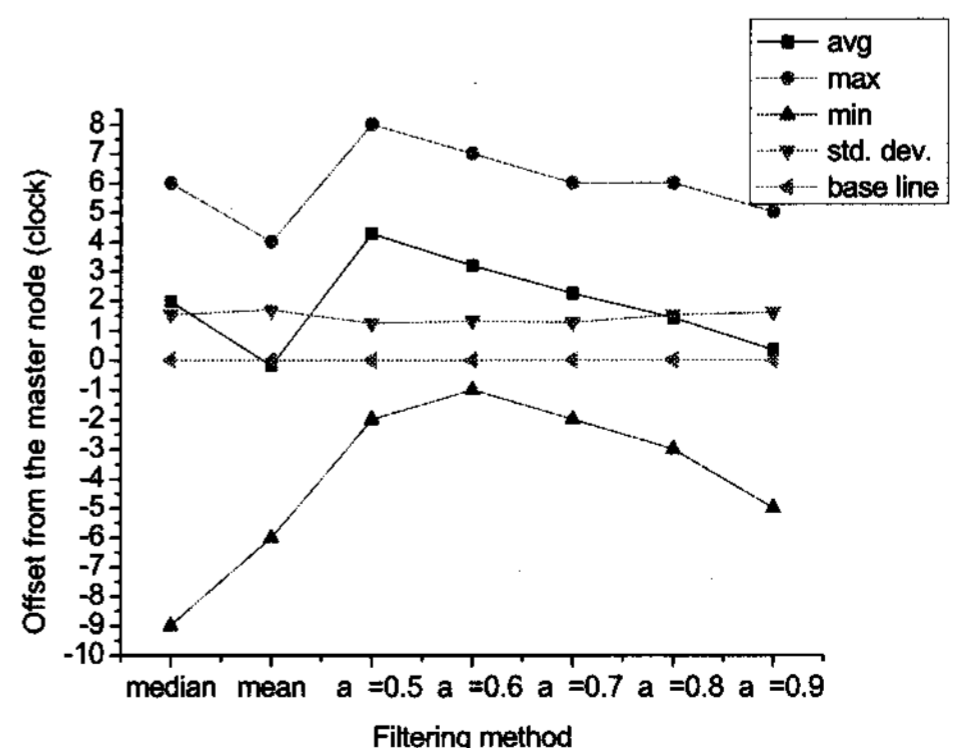


그림 11. 필터링 방법에 따른 시각 동기 성능평가  
 Fig. 11. Time synchronization result by filtering methods



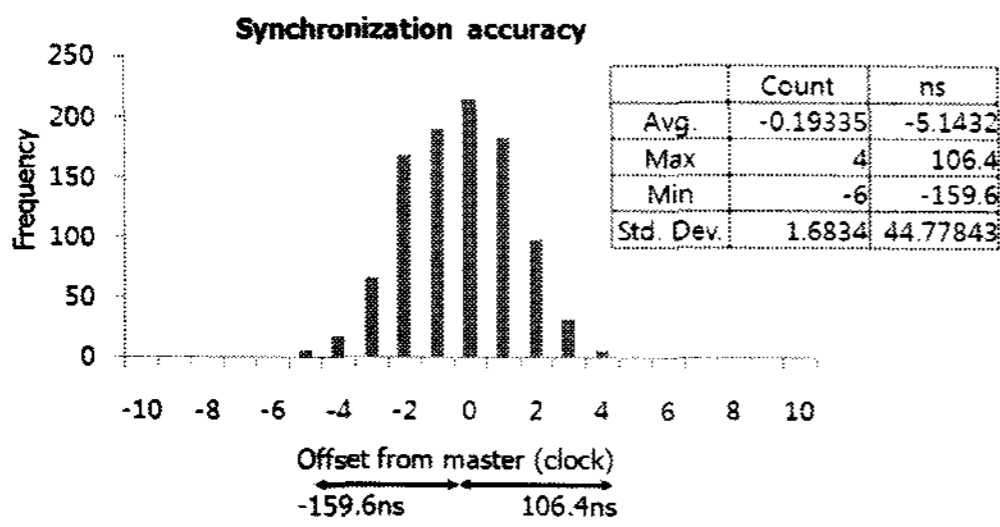


그림 12. 평균을 이용한 시각 동기  
Fig. 12. Time synchronization by the mean filter

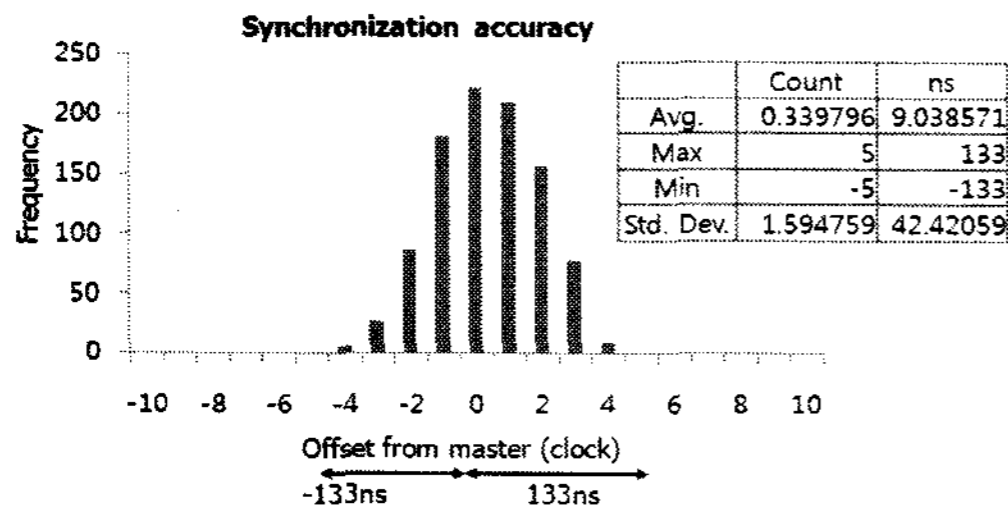


그림 13. 0.9의 가중치 부여시 시각 동기  
Fig. 13. Time synchronization by the 0.9 weight

우 평균 -5.14나노초의 시각 오차와 44.77나노초의 표준편차를 가지며, 0.9 가중치의 경우 평균 9나노초의 시각 오차와 42.42나노초의 표준편차를 가진다.

### V. 결 론

본 논문에서는 ZigBee를 이용한 무선 네트워크 환경에서 마이크로초 이하의 정밀한 시각을 동기화 하는 기법을 설계 및 구현하였다. 제안한 기법은 기본적으로 참조 시각을 제공하는 마스터 노드와 참조 시각에 시각을 동기화하는 슬레이브 노드로 구성된다. 이들간의 동기화 기법은 기본적으로 메시지 교환을 통하여 이루어지며, 동기 메시지는 마스터 노드가 주기적으로 전송한다. 메시지 교환시 타임 스탬프를 수행하는데 있어 많은 에러 요인들이 발생하는데 이를 줄이기 위해 본 논문에서는 MII 수준에서 하드웨어적으로 스탬프 하기 위해 하드웨어를 설계하고, 정밀한 시각 동기를 하는데 걸림돌이 되는 많은 에러 요인들을 최소화 하여 마이크로초 이하의 정밀한 시각 동기 기법을 개발하였다. 제안한 기법은 IEEE 1588과 유사하지만, ZigBee의 특성 및 시각 동기화에서 발생할 수 있는 에러 요인들을 면밀히 분석하였다. 성능평가 결과, 이동평균의 경우 평균 -5.14나노초의 시각 오차와 44.77나노

초의 표준편차를 가지며, 0.9 가중치의 경우 평균 9나노초의 시각 오차와 42.42나노초의 표준편차를 가진다. 본 논문에서 개발한 정밀 시각 동기 기법은 다양한 무선 네트워크에서 정밀한 시각 동기를 요구하는 시스템에 적용가능하다. 또한, ZigBee 네트워크를 활용하는 무선 센서 네트워크에서도 정밀 시각 동기를 제공할 수 있다.

현재 시스템은 마스터 노드의 원 홉 이웃 노드들만 동기를 맞추도록 설계되어 있다. 향후 현재 시스템을 확장하여, 멀티 홉으로 정밀한 시각을 유지하는 기법에 대한 연구, 보다 정밀한 시각을 유지하는 기법, 네트워크내의 트래픽량을 줄이는 기법들에 대한 연구가 필요할 것으로 보인다.

### 참 고 문 헌

- [1] Soyoung Hwang, Yunju Baek, "Fault Tolerant Time Synchronization for Wireless Sensor Networks," *Lecture Notes in Computer Science*, vol. 3894, pp. 480-493, Mar. 2006.
- [2] Mills, D.L, "Network Time Protocol (Version 3) Specification, Implementation and Analysis," *RFC1305*, 1992.
- [3] A.Günther, C. Hoene, "Measuring Round Trip Times to Determine the. Distance Between WLAN Nodes," *Networking 2005*, pp. 768-779, 2005.
- [4] Ryota Yamasaki, Atsushi Ogino, Tsuyoshi Tamaki, Takaki Uta, Naoto Matsuzawa, Takeshi Kato, "TDOA Location System for IEEE 802.11b WLAN," in *Proceedings of IEEE Wireless Communications and Networking Conference*, Vol. 4, pp. 2338-2343, Mar. 2005.
- [5] IEEE 1588<sup>TM</sup>-2002, "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems," *IEEE Instrumentation and Measurement Society*, November 2002.
- [6] Weibel, Hans, "High Precision Clock Synchronization according to IEEE 1588 - Implementation and Performance Issues," *Embedded World*, pp. 22-24, 2005.
- [7] Juha Kannisto, Timo Vanhatupa, Marko

Hännikäinen, and Timo D. Hämäläinen, "Precision Time Protocol Prototype on Wireless LAN," *Lecture Notes in Computer Science*, vol. 3124, pp. 1236 - 1245, 2004.

[8] Kannisto, J., Vanhatupa, T. Hannikainen, M. Hamalainen, T.D. "Software and hardware prototypes of the IEEE 1588 precision time protocol on wireless LAN," *14th IEEE Workshop on Local and Metropolitan Area Networks 2005*, Sept. 2005.

[9] H. G. Berns, T. H. Burnett, R. Gran, and R. J. Wilkes, "GPS time synchronization in school-network cosmic ray detectors," *IEEE Trans. Nucl. Sci.*, vol. 51, June 2004.

[10] Hans Weibel, Dominic Béchaz, "IEEE 1588 Implementation and Performance of Time Stamping Techniques," *2004 Conference on IEEE 1588*, September 28, 2004.

[11] Freescale, "MCF5235 datasheet," <http://www.freescale.com>, 2004.

[12] Texas Instrument, "CC2420 datasheet," <http://www.ti.com>, 2004.

**조 현 태 (Hyuntae Cho)**

정회원



2003년 2월 한국해양대학교 컴퓨터공학과 학사  
 2005년 2월 부산대학교 컴퓨터공학과 석사  
 2005년 3월~현재 부산대학교 컴퓨터공학과 박사과정

<관심분야> 무선 센서 네트워크, RFID, RTLS, 시각 동기

**손 상 현 (Sanghyun Son)**

준회원



2007년 2월 부산대학교 컴퓨터공학과 학사  
 2007년 3월~현재 부산대학교 컴퓨터공학과 석사과정

<관심분야> 임베디드시스템, RTLS, 시각 동기

**백 윤 주 (Yunju Baek)**

중신회원



1990년 2월 한국과학기술원 전산학과 학사  
 1992년 2월 한국과학기술원 전산학과 석사  
 1997년 2월 한국과학기술원 전산학과 박사  
 1996년 6월~2002년 2월 NHN

(주) 연구소장, 2003년 9월~현재 부산대학교 정보컴퓨터공학부 교수

<관심분야> 무선 센서 네트워크, 컴퓨터구조, 임베디드 시스템, RTLS