

IP-USN을 위한 SNTP 기반의 다중 홉 시각동기 기법

박상준[○] 홍성민 하민근 김영주 김대영[†]

한국과학기술원(KAIST)

{sigpsj13438, minkeun.ha, ghitsh, yjkim, kimd}@kaist.ac.kr

A Multi-hop Time Synchronization Method based on SNTP for IP-USN

Sangjun Park[○], Sungmin Hong, Minkeun Ha, Young-Joo Kim, Daeyoung Kim[†]
Korea Advanced Institute of Science and Technology

요약

IP-USN은 IEEE 802.15.4 표준과 IPv6를 바탕으로 하는 IP 기반의 무선 센서네트워크로서, 기존의 IP 인프라를 활용하여 광역의 접근성을 제공하는 기술이다. IP-USN은 기존의 시각동기를 필요로 하는 WSN 기술들뿐만 아니라, IP를 이용한 응용의 확장성을 위해 글로벌 시각 동기화가 필요하다. 그러나 인터넷 기술인 SNTP는 메시지 크기가 크고 알고리즘이 복잡하며, 기존의 센서네트워크 시각동기 기술들은 단일 홉 시각동기 기법으로 전체 센서네트워크의 시각동기를 지원하기 때문에 노드 간 시각 정보의 송수신이 증가하거나, 기존 노드를 재선정하는 등의 문제점이 발생한다. 본 논문에서는 인터넷 기반의 시각동기 기술인 SNTP를 경량화 하여 글로벌 센서네트워크를 형성할 수 있는 다중 홉 시각동기 기법을 제안한다. 이 기법은 최소 48바이트의 SNTP 메시지를 33바이트로 줄임으로써 패킷 크기를 최소화하고, 게이트웨이 시각동기와 PAN 시각동기로 이루어진다. 다중 홉으로 이루어지는 PAN 시각동기 단계에서는 중간노드들에 의한 지연시간을 시각오차 계산에 반영함으로써 마이크로 초 단위의 정확도를 지원하면서, 글로벌 시각동기를 할 수 있을 것으로 기대된다.

1. 서론

IP-USN은 IEEE 802.15.4 표준과 IPv6를 사용하는 인터넷 기반의 무선 센서네트워크로서[10], 기존의 IP 인프라를 기반으로 광범위한 확장성을 제공하는 기술이다. 즉, 현재의 IP인프라 시설과 검증된 인터넷 기술을 활용하여 IP-USN에 적용함으로써 다양한 응용 및 서비스를 제공하고, IP기반 시설이 있는 곳에서는 어디서나 IP-USN에서 제공하는 응용 및 서비스를 이용할 수 있도록 하는 기술이다. 이런 IP-USN 또한 무선 센서네트워크이기 때문에 작은 패킷 크기, 적은 대역폭, 저전력 사용, 및 저비용 장치 등 여러 가지 제한사항을 가지고 있다. 이러한 제한 사항으로 인해 이벤트 중복 감지, 에너지 효율 향상, 통신비용 감소 및 위치계산 등에 관련된 연구들이 진행되고 있으며[6, 12], 이러한 무선 센서네트워크 기술들을 뒷받침하고 IP를 이용한 응용의 확장성을 위해 글로벌 시각동기 기술이 필요하다.

시각동기에 관련된 기술은 인터넷 시각동기 기술인

NTP[2]와 SNTP[3]가 있으며, 무선 센서네트워크 시각동기 기술인 RBS[6], TPSN[11], TSync[5], FTSP[9], LTS[8] 등이 있다. 먼저, NTP[2]와 SNTP[3]는 전원과 메모리 등의 자원이 풍부한 네트워크에서 사용되므로 시간 선택 알고리즘과 오차 보정 및 보안 알고리즘의 복잡도가 증가할 뿐만 아니라, 시각동기 오차범위가 밀리 초 단위이며, 서버와 클라이언트간 패킷 사이즈가 크기 때문에 무선 센서네트워크에 직접 적용하기는 어렵다. 다음으로, 무선 센서네트워크의 시각동기 기술인 RBS[6], TPSN[11], TSync[5], FTSP[9], LTS[8] 등은 이웃 센서노드들과 시각동기를 반복하여 수행함으로써 센서네트워크 전체의 시각동기를 이룬다. 즉, 기존의 센서네트워크 시각동기 기술[5,6,8,9,11]은 단일 홉 시각동기 기법으로 전체 센서네트워크의 시각동기를 지원한다. 이로 인해 노드간 시각 정보의 송수신이 증가하거나, 기존 노드를 재선정하는 등의 문제점이 발생한다.

본 논문에서는 인터넷 기반의 시각동기 기술인 SNTP를 경량화하여 글로벌 센서네트워크를 형성할 수 있는 다중 홉 시각동기 기법을 제안한다. 이 기법은 최소 48바이트의 SNTP 메시지를 33바이트로 패킷 크기를 최소화하고, 게이트웨이 시각동기 단계와 PAN 시각동기 단계로 이루어진다. 게이트웨이 시각동기 단계는 개개의 IP-USN

* 본 연구는 지식경제부의 대학 IT연구센터 지원 사업에 의해 수행되었습니다.

(IITA-2009-(C1090-0902-0047))

* 본 연구는 2009년도 정부(과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행되었습니다. (ROA-2007-000-10038-0)

† 교신저자: 한국과학기술원 전산학과 부교수

게이트웨이가 인터넷의 SNTP 서버와 주기적인 시각동기를 이루는 단계로 SNTP 알고리즘을 따른다. PAN 시각동기의 1단계는 시각 동기화된 IP-USN 게이트웨이의 방송 메시지를 활용하여 센서노드가 시각오차를 계산 및 저장하며, 2단계는 각 센서노드가 임의의 주기로 게이트웨이에게 시각동기를 요청하고 게이트웨이가 중간노드들의 지연시간의 총합을 구하여 응답메시지를 전송하는 것이다. 마지막 단계인 3단계는 게이트웨이로부터 전송된 응답메시지를 받은 요청 센서노드가 게이트웨이와 자신의 시각오차를 계산하고, 1단계에서 얻은 시각오차와의 평균을 이용하여 자신의 시각을 수정한다. 이렇게 계산된 시각오차는 중간단계의 모든 지연시간을 최소화함으로써 정확도를 향상시킬 수 있다. 따라서 제안된 기법은 인터넷 시각 동기 표준화 기술인 SNTP를 경량화 하였으며, 다중 홉으로 이루어지는 PAN 시각동기 단계에서 중간노드들에 의한 지연시간을 시각오차에 반영함으로써 마이크로 초 단위의 정확도를 지원하면서, 글로벌 시각동기를 할 수 있을 것으로 기대된다.

본 논문의 구성은 다음과 같다. 2장에서는 시각동기 기술과 관련된 연구의 문제점을 살펴보고, 3장에서는 제안된 SNTP기반의 다중 홉 시각동기 기법에 대해서 설명한다. 4장에서는 제안된 기법에 대해서 분석을 한다. 마지막으로 5장에서는 결론으로 논문의 끝을 맺는다.

2. 관련연구

Sntp[3]는 서버와 서버, 서버와 클라이언트 간 다중 홉을 기반으로 하는 인터넷 시각동기 표준화 기술이다. NTP를 단순화 시킨 것으로 SntpV4에서는 IPv6와 OSI를 지원하며, NTP와 동일한 메시지 형식을 사용하고, UDP를 전송계층으로 사용한다. Sntp 클라이언트는 1개의 서버와 직접 시각동기를 하며, 다른 클라이언트와 시각 비교

를 하지 않고, 서버와 클라이언트간의 경로가 무선 센서 네트워크에 비해 상당히 멀기 때문에 일반적으로 수십 밀리 초의 오차를 갖는다. IP-USN에서 모든 계층들이 압축되지 않은 헤더만을 사용할 경우에 <그림1>과 같이 최소 48바이트의 패킷을 사용하는 Sntp를 IP-USN에 직접 적용할 경우 요구하는 패킷의 크기가 사용할 수 있는 패킷의 크기보다 클 수 있다. 이는 추가적인 메시지 송수신을 발생시켜 센서네트워크 내에서의 트래픽을 증가시키고, 그로 인해 에너지 소모가 증가하게 되어 센서네트워크의 수명을 단축하는 문제를 발생시킨다.

RBS[6]는 시각 정보를 송신하는 노드는 동기를 하지 않으며, 수신한 노드가 인접 노드와의 시간 차이를 다시 확인하여 시각 동기를 하는 프로토콜이다. 단일 홉 간의 시각동기를 기반으로 하고 송신 노드가 여러 개가 있을 수 있으며, 정확도를 높이기 위하여 여러 인접 노드의 시각 정보를 수집하여 통신비용이 증가하므로, 규모가 큰 네트워크에 적용할 경우에 정확한 시각동기를 할 수 없는 문제가 발생할 수 있다.

TPSN[11]은 단일 홉간 송신-수신 메시지 교환을 통한 시각 동기를 하는 점에서는 Sntp와 유사하나, 전체 네트워크에서의 다중 홉 간 시각 동기는 이러한 단일 홉 간 시각동기의 반복을 통해 이루어진다. 그러나 다중 홉 간 시각동기에서는 이벤트를 감지해서 먼저 송신하는 노드의 시간에 수신하는 노드의 시각을 맞추기 때문에 전체 네트워크에서 보면 기준 노드가 바뀌게 된다. 결국, 라우팅 경로가 여러 개 존재하는 경우에는 기준 노드가 여러 개가 될 수 있으므로 큰 규모의 네트워크에서는 시각 오차가 더욱 커지게 된다.

FTSP[9]는 송신 노드가 하나의 시각동기 메시지를 보낼 때 여러 번 시각소인을 하여 그 평균값을 데이터로 보내며 수신 노드도 마찬가지로 메시지를 받으면서 여러 번의 시각 소인을 하여 그 평균값을 내고, 데이터에 있는 송신 노드의 시각 평균값과의 오차를 구하여 시각동기를 이룬다. 그러나 이 또한 다중 홉 시각 동기를 위해서는 충분히 많은 수의 시각 메시지를 수집해야 하며, 이는 비용 증가를 가져오게 된다. 뿐만 아니라 루트 노드를 재선출하는 알고리즘을 가지고 있기 때문에 시각동기의 기준이 바뀌게 된다. 따라서 지속적으로 글로벌 시각을 유지하기 위해서는 모든 센서노드가 GPS 수신기를 가지고 있거나 인터넷 등 다른 네트워크와 연결되어 있어야 한다는 단점을 가지고 있다.

이와 같이 Sntp 시각동기 기술은 메시지 크기가 크고 네트워크의 트래픽을 증가시키며, 기존의 무선 센서네트워크 시각동기 기술들은 로컬 네트워크 내에서 단일 홉 시각동기

0	8	16	24	31
LI	VN	Mode	Stratum	Poll
Precision				
Root Delay (32)				
Root Dispersion (32)				
Reference Identifier (32)				
Reference Timestamp (64)				
Originate Timestamp (64)				
Receive Timestamp (64)				
Transmit Timestamp (64)				
Authenticator (optional) (96)				

<그림 1. Sntp 메시지 형식>

를 기반으로 전체 네트워크의 시각 동기를 지원하고 있으며, 기준노드가 바뀌는 알고리즘을 가지고 있어서 지속적으로 글로벌 시각동기를 유지하는데 문제점을 가지고 있다.

3. SNTP 기반의 다중 홉 시각 동기 기술

본 절에서는 SNTP에서 사용하는 시각 동기 메시지 형식을 IP-USN에서 효율적으로 활용할 수 있는 시각 동기 메시지 형식으로 경량화한 것에 대해 설명하고, 다중 홉으로 이루어지는 PAN 시각 동기화에 대한 알고리즘에 대해서 설명한다.

3. 1. 메시지 형식

IEEE 802.15.4 표준을 따르는 IP-USN은 패킷 전송 용량이 127바이트로 제한되어 있다[4][10]. 127바이트의 패킷 크기는 확장형 MAC 주소를 사용한 MAC 헤더(25바이트), 데이터 링크 계층에서 보안(21바이트)을 사용하고, 압축되지 않은 IPv6 헤더(40바이트)와 UDP 헤더(8바이트)를 사용할 경우에 응용에서 사용할 수 있는 데이터 크기는 33바이트이다.

따라서 UDP를 전송계층으로 사용하고 메시지의 최소 크기가 48바이트인 SNTP 시각 동기 기술을 IP-USN에 그대로 적용할 경우에 시각 동기를 요청하는 노드는 메시지를 두 개로 나누어서 전송하고 메시지를 받는 노드는 다시 하나의 메시지로 조립해야 한다. 즉, 패킷의 분해/재조립을 요구하게 되며, 이는 메시지 송수신을 증가시켜 통신비용의 증가를 유발하므로 SNTP를 기반으로 하는 다중 홉 시각 동기 기법에서 메시지 형식의 수정이 필요하다.

제안된 다중 홉 시각 동기 기법의 메시지 형식은 <그림 2>와 같이 SNTP 메시지 형식을 33바이트의 크기로 경량화 하였다. Mode는 3가지로 방송(broadcast), 요청(request), 응답(reply)이다. Sequence Number는 중복 메시지를 피하기 위한 것이며, Root Delay는 게이트웨이와

0	7
Mode(2)	Sequence Number(6)
Root Delay(64)	
Originate Timestamp(64)	
Receive Timestamp(64)	
Transmit Timestamp(64)	

<그림 2. 메시지 형식>

```

01:procedure broadcaster
02:  var
03:    // 변수 선언은 생략
04:  begin
05:    do while
06:      flagSync := adjustSyncClock();
07:      if flag = true then
08:        syncMsg := createSyncMsg();
09:        setMode(syncMsg, BROAD);
10:        broadcastMsg(syncMsg);
11:      end if
12:    end do
13:end procedure

```

<알고리즘 1. IP-USN 게이트웨이: broadcaster>

시각 동기를 하는 센서노드 사이에 있는 모든 노드들의 지연시간을 누적한 것이며, 게이트웨이의 처리 시간은 포함되지 않는다. Originate Timestamp에서 방송(broadcast) 모드에서는 게이트웨이의 송신시간을, 요청(request) 및 응답(reply) 모드에서는 시각 동기를 요청한 센서노드의 시각을 의미한다. Transmit Timestamp와 Receive Timestamp는 중간노드의 송수신 시각으로 중간노드의 처리시간을 계산하는데 사용되고, 응답 모드에서는 게이트웨이가 요청 메시지를 수신한 시각이며, 응답 메시지의 전달 과정에서 바뀌지 않는다.

시각소인(Timestamp)은 정확도를 향상시키기 위해서 PHY 계층 헤더가 시작하는 부분에서, 즉 SFD(Start of Frame Delimiter) 필드의 전송이 끝나는 시점에서, 시각소인을 하는 ZigBee Network에서의 시각 동기 기술[1,13]을 인용하였다.

3. 2. 시각 동기 알고리즘

본 논문에서 제안한 다중 홉 시각 동기는 게이트웨이 시각 동기와 PAN 시각 동기화로 이루어져 있다. 게이트웨이 시각 동기는 기존의 SNTP 프로토콜을 이용하여 SNTP 서버와 IP-USN 게이트웨이 간의 시각을 동기화한다. 이렇게 동기화된 게이트웨이는 센서네트워크 내의 노드들에 대한 시각 동기를 위해서 PAN 시각 동기를 하게 된다.

PAN 시각 동기는 게이트웨이 내에 존재하는 broadcaster와 receiver 알고리즘과 각 센서노드에 존재하는 receiver와 requester 알고리즘에 의해 이루어진다. <알고리즘 1>은 IP-USN 게이트웨이에서 동작하는 broadcaster 알고리즘으로써, SNTP 알고리즘에 의해서 SNTP 서버와 게이트웨이 간에 글로벌 시각 동기를 마친 후에 라인 6의 함수를 호출하면 PAN 내의 시각 동기 시작여부를 할 수 있다. 만약 flagSync가 true이면 라인 8~10과 같이 '방송' 모드의 시각 동기 메시지를 PAN내의 센서노드들에게 방송

```

01:procedure receiver
02: var
03: // 변수 선언은 생략
04: begin
05: do while
06:   recvMsg := recvSyncMsg();
07:   ...
08:   if getMode(recvMsg) = "BROAD" then
09:     if broadSeqNum < recvSeqNum then
10:       storeSeqNum(recvSeqNum);
11:       tempMsg := modifyrecvMsg(recvMsg);
12:       ...
13:       prevBroadOffset := loadOffset();
14:       currBroadOffset := myRecvTime-origintime-procTime;
15:       avgBroadOffset:=(1-α)prevBroadOffset+α.currBroadOffset;
16:       storeOffset(avgBroadOffset);
17:       broadcastMsg(tempMsg);
18:     else if getMode(recvMsg) = "REQ" then
19:       myIP := getSystemIP();
20:       destinationIP := loadIP();
21:       ...
22:       if myIP <> destinationIP then msgForwarding(recvMsg);
23:     else msgSend(tempMsg);
24:     else if getMode(recvMsg) = "REP" then
25:       ...
26:       if myIP <> destinationIP then msgForwarding(recvMsg);
27:     else
28:       ...
29:       avgBroadOffset := loadOffset();
30:       diffRequestOffset:=preRecvTime-origintime-procTime;
31:       diffTime := average(avgBroadOffset,diffRequestOffset);
32:       adjustLocalClock(diffTime);
33:     end if
34:   end if
35: end do
36:end procedure

```

<알고리즘2. IP-USN 게이트웨이와 센서노드: receiver>

한다. 방송메시지를 받은 센서노드는 <알고리즘2>의 receiver 알고리즘의 라인 8~16과 같이 Sequence Number를 통해 메시지의 중복 여부를 확인하고, 게이트웨이의 메시지 송신시각(*originTime*), 자신의 메시지 수신시각(*myRecvTime*), 그리고 중간노드들의 지연시간(*procTime*)을 이용하여 게이트웨이와 자신의 평균 시각오차(*avgBroadOffset*)를 계산하고 메모리에 저장한다. 시각오차를 저장한 센서노드는 라인 17처럼 다음 단계로 방송메시지를 전달한다. 여기서 지연시간의 계산은 <그림 3>의 예로 볼 때, Node1의 경우에 게이트웨이의 지연시간을 계산하는 것이므로 '0'이 되고, Node 2의 경우에는 게이트웨이와 Node1의 지연시간을 모두 구하는 것이므로, <알고리즘2>의 라인 11의 '*modifyrecvMsg()*'에 의하여 Node1로부터 받은 메시지 내의 Transmit Timestamp, Receive Timestamp, 그리고 Root Delay 값을 이용하여 누적된 Root Delay값을 계산한다. 즉, 게이트

```

01:procedure requester
02: var
03: // 변수 선언은 생략
04: begin
05: do while
06:   pollingInterval := POLLINGSEC;
07:   currentTime := getCurrentTime();
08:   if currentTime-lastReqTime>=pollingInterval then
09:     lastReqTime := currentTime;
10:     syncMsg := createSyncMsg();
11:     setMode(syncMsg,REQ);
12:     msgSend(syncMsg);
13:   end if
14: end do
15:end procedure

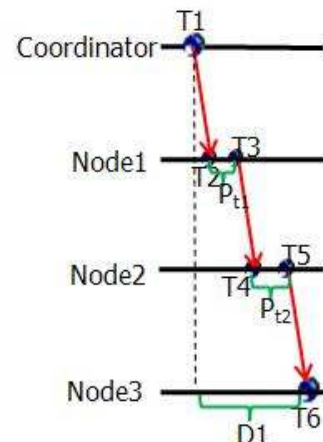
```

<알고리즘3. IP-USN 센서노드: requester>

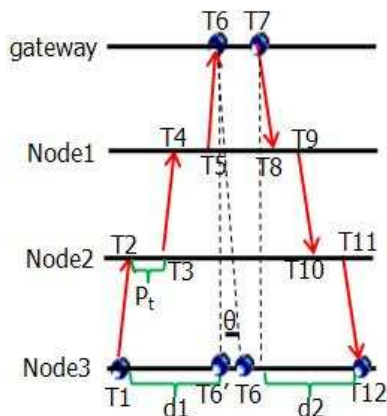
웨이로부터 n홉만큼 떨어진 노드가 n-1홉 노드까지의 모든 지연시간을 계산하는 것이다. 이러한 과정은 경로상 마지막 단계의 노드까지 반복적으로 수행한다.

<알고리즘3>의 센서노드 시각동기 요청 알고리즘은 방송메시지를 수신한 각각의 노드가 requester 알고리즘과 같이 자신의 시각동기 주기인 *pollingInterval*에 따라 요청메시지를 생성하여 전송한다. 중간노드의 처리 과정은 <알고리즘2>의 라인 18~23에 의해 진행되며, 이는 broadcaster에 의한 방송메시지 receiver 알고리즘과 유사하다. 단, 센서노드가 게이트웨이와 시각을 맞추기 위한 것이므로 목적지 IP를 확인하는 과정이 필요하다. 요청메시지를 수신한 게이트웨이는 <알고리즘 2>의 라인 11과 동일한 '*modifyrecvMsg()*'에 의해서 모든 중간노드에서 걸린 지연시간을 계산하여 그 값과 자신의 수신시간을 메시지에 넣고, 목적지 IP와 출발지 IP를 수정하여 응답메시지를 송신한다.

응답메시지를 받은 중간노드들은 <알고리즘2>의 라인 26과 같이 다른 처리과정 없이 메시지를 재전송한다. 응답메시지를 수신한 시각동기 요청노드는 <알고리즘2>의 라인



<그림 3. broadcast>



<그림 4 request/reply>

27~32와 같이 자신의 최초 메시지 송신시각(*originTime*), 게이트웨이의 메시지 수신시각(*myRecvTime*), 그리고 중간 노드들의 지연시간(*procTime*)을 이용하여 시각오차 θ_R 를 구하고, 방송메시지로부터 얻은 시각오차 θ_B 와 평균 시각오차 θ_{avg} 를 구하여 자신의 시각을 조정한다.

4. 고 찰

SNTP[3]는 서버와 클라이언트의 메시지 교환을 통해서 시각동기를 한다. 인터넷에서의 서버 및 클라이언트는 자원이 풍부하기 때문에 경로에 상관없이 시각 동기를 하고 그 정확도는 밀리 초 단위이다. 그러나 무선 센서 네트워크[7]에서는 자원이 충분하지 않기 때문에 스케줄링, 이벤트의 중복 방지 등을 위해서 더욱 높은 정확도를 요구한다. 따라서 단일 홉 동기를 하고, 물리계층에서 시각소인을 얻는 것이 가장 정확한 방법으로 알려져 있다. 다중 홉 시각동기에서 이러한 효과를 얻기 위해서는 중간노드의 처리시간을 최소화해야 한다. 그래서 본 논문에서 제안한 알고리즘은 모든 중간 센서노드들의 "Send", "Receive", "Access", "Interrupt Handling" 등의 무선 메시지 전송에서의 불확실성[9]의 요인을 최소화하기 위해서 <알고리즘2>에서 모든 중간노드에서 발생하는 지연시간을 누적하여 메시지로 전달함으로써 중간 단계에서 발생하는 불확실성을 최소화하였다.

본 논문에서 제안한 PAN 시각동기 알고리즘의 흐름을 나타내는 <그림3>과 <그림4>에서 보인 것처럼 T_1 에서 T_6 로 진행되는 과정에서 게이트웨이와 센서노드의 시각오차 θ , 메시지가 전달될 때의 지연 시간을 d 라 하면, 지연시간 d 는 전송시간(propagation delay)과 중간노드들의 처리시간(P_i)의 합이므로, 다음의 식을 얻을 수 있다.

$$T_6 = T_1 + \theta_1 + d_1 \text{ -----(1)}$$

$$T_{12} = T_7 - \theta_2 + d_2 \text{ -----(2)}$$

식 (1)과 (2)를 θ 에 대해서 변환하면,

$$\theta_1 = T_6 - T_1 - d_1 \text{ -----(3)}$$

$$\theta_2 = T_7 - T_{12} + d_2 \text{ -----(4)}$$

을 얻을 수 있다.

시각동기 요청 및 응답메시지를 한 번 송수신하는 과정에서 시각오차 θ_1 과 θ_2 가 동일하다고 가정하면 식(3)과 (4)로부터 다음 식을 얻을 수 있다.

$$\theta_{1+2} = (T_6 - T_1) + (T_7 - T_{12}) - (d_1 - d_2) \text{ -----(5)}$$

식(1)과 (2)를 지연시간 d 에 대해서 다시 작성하면,

$$d_1 = T_6 - T_1 - \theta_1 \text{ -----(6)}$$

$$d_2 = T_{12} - T_7 + \theta_2 \text{ -----(7)}$$

을 얻을 수 있다. 위에서 요청 및 응답메시지를 한번 송수신하는 과정에서는 θ_1 과 θ_2 가 동일하므로, 식(6)과 (7)을 더하면,

$$d_1 + d_2 = (T_6 - T_1) + (T_{12} - T_7) \text{ -----(8)}$$

$$\therefore d_2 = (T_6 - T_1) + (T_{12} - T_7) - d_1 \text{ -----(9)}$$

식(9)를 식(5)에 대입하면,

$$\theta_{1+2} = 2(T_6 - T_1) - 2d_1 \text{ -----(10)}$$

$$\therefore \theta_{avg} = T_6 - T_1 - d_1 \text{ -----(11)}$$

결국 시각오차는 식(1)과 동일해진다. 여기서 중간 노드로 인해 발생하는 모든 지연시간을 빼면 단일 홉에서의 시각오차 θ 를 얻을 수 있다. 이러한 방법으로 시각오차 θ 를 구할 경우, 네트워크의 상태에 따라서 달라질 수 있는 d_1 과 d_2 의 차이에 따른 오차도 극복할 수 있으며, 단지 무선 구간의 전파시간(propagation time)의 오차만 발생하게 된다. 무선 구간의 전파시간은 300m 이내에서는 1us 미만이므로[9] 오차 계산에 영향을 주지 않는다고 가정하고, 모든 중간노드들에 의해 발생한 지연시간의 총 합을 제거하면, 다중 홉에서의 시각오차 θ 는 단일 홉에서의 시각오차 θ 와 같다고 할 수 있다. 따라서 중간노드들의 지연시간의 총 합을 d_t 라 하면, 시각오차 θ 는 다음의 식 (12)에 의해서 구할 수 있다.

$$\theta = T_6 - T_1 - d_t \text{ -----(12)}$$

본 논문에서 제안된 알고리즘은 ZigBee 네트워크를 위한 시각동기 기술[1,13]에서 사용한 것처럼 SFD의 끝부분에서 시각소인을 얻는다. ZigBee 시각동기[1,13]에서는 32768Hz의 수정 진동자를 사용하여 실험을 하였으며, 모든 실험 결과에서 시각오차가 2ticks(61us)을 넘지 않는다고 하였을 뿐만 아니라 실험 결과에서 평균 시각오차는 1tick(30.5us)을 넘지 않는다고 하였다. 따라서 동일한 지점에서 시각소인을 얻는 본 논문에서 제안한 알

고리즘 또한 마이크로 초 단위의 정확성을 얻을 수 있을 것으로 기대된다.

5. 결론 및 향후 과제

본 논문에서는 IP 기술을 이용하여 광역의 접근성을 제공하는 IP-USN에서의 글로벌 시각동기를 지원하기 위한 알고리즘을 제안한다. 본 알고리즘은 SNTP를 경량화한 IP-USN 시각동기 메시지 형식을 사용하며, 글로벌 시각동기를 위하여 게이트웨이 시각동기와 PAN 시각동기를 한다. 이 알고리즘에서 PAN 시각동기는 다중 홉으로 이루어지며, 중간노드들에 의한 지연시간을 시각오차 계산에 반영함으로써 마이크로 초 단위의 정확성을 보장할 것으로 기대된다. 향후과제로는 본 논문에서 제안된 알고리즘의 정확도를 실험적으로 증명할 것이다.

6. 참고문헌

- [1] D. Cox, E. Jovanov, and A. Milenkovic, "Time Synchronization for Zigbee Networks," *SSST05*, pp.135-138, Mar. 2005.
- [2] D. Mills, *Network Time Protocol (version3) specification, implementation and analysis*, RFC1305, Mar. 1992.
- [3] D. Mills, *Simple Network Time Protocol(SNTP) version 4 for IPv4, IPv6 and OSI*, RFC 4330, Jan. 2006.
- [4] G. Montenegro, N. Kushalnagar, and D. Culler, *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*, RFC 4944, Sep. 2007.
- [5] H. Dai and R. Han, "TSync: a lightweight bidirectional time synchronization service for wireless sensor networks," *ACM Mobile Computing and Communications Review*, 8:125-139, Jan. 2004.
- [6] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," *ACM Operating Systems Review*, 36:147-163, Dec. 2002.
- [7] J. Elson and K. Roemer, "Wireless Sensor Networks: A New Regime for Time Synchronization," *ACM SIGCOM*, 33(1):149-154, Jan. 2003.
- [8] J. Greunen and J. Rabaey, "Lightweight time synchronization for sensor networks," *ACM WSNA03*, pp. 11-19, Sep. 2003.
- [9] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi, "The flooding time synchronization protocol," *ACM SenSys04*, 39-49, Nov. 2004.
- [10] N. Kushalnagar, G. Montenegro, and C. Schumacher, *IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals*, RFC4919, Aug. 2007.
- [11] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-sync protocol for sensor networks," *ACM SenSys03*, 138-149, Nov. 2003.
- [12] 황소영, 유동희, 백윤주, "센서네트워크 시각 동기 기술 연구," *GMSS*, Nov. 2004.
- [13] 조현태, 손상현, 백윤주, "Zigbee를 이용한 무선 네트워크 환경에서의 정밀 시각 동기 기법 설계 및 구현," *한국통신학회논문지*, 33(5): 561-570, May 2008.