

무선 센서 네트워크에서의 개선된 시각 동기화 구현

방상원[○], 손석원^{*}
호서대학교 컴퓨터공학과
e-mail:onlyses898@nate.com[○]

Implementation of an Improved Time Synchronization in Wireless Sensor Networks

Sangwon Bang[○], Surgwon Sohn^{*}
Dept. of Computer Engineering, Hoseo University

● 요약 ●

본 논문은 TPSN 알고리즘의 시각 동기화 오차를 개선하기 위하여 Imote2 센서 노드의 클럭 드리프트 특성을 적용하는 개선된 TPSN 알고리즘을 제안한다. 클럭 드리프트의 원인은 주로 수정발진기에 기인한다. 본 연구에서는 온도 및 습도 등 환경 조건이 비슷할 경우에 드리프트가 크게 차이나지 않는다는 실험 결과에 따라 드리프트의 평균값을 구하고 이를 TPSN 동기화 오차 보정에 사용한다. 이때 적용되는 드리프트 특성 값은 센서 노드 설치 이전에 미리 측정하여야 한다. 실험을 통하여 본 논문에서 제안한 개선된 TPSN 알고리즘이 동기화 오차 개선에 효과적임을 확인하였다.

키워드: 시각 동기화 (Time Synchronization), 클럭 드리프트(Clock Drift), 무선 센서 네트워크(Wireless Sensor Networks)

1. 서론

최근 임베디드 프로세서 기술의 발달로 프로세서는 저 전력화 되고 경량화된 반면 고성능의 컴퓨팅능력을 갖추어 분산 컴퓨터 환경에서 데이터를 처리하는 것이 가능하게 되었다. 이러한 기술을 바탕으로 센서 노드를 사람이 직접 접근할 수 없거나 실시간 감시를 요하는 곳에 설치하여 이를 이용하여 데이터를 수집하거나 고성능의 MCU를 이용하여 원시 데이터를 처리할 수 있게 되었다. 이러한 기술을 바탕으로 건물 화재 감시, 구조물 안전성 감시, 무인 방법 시스템 등 여러 분야에 적용되고 있다.

일반적으로 각각의 센서 노드에 의해 측정된 원시 데이터들을 단순하게 처리하거나 무선 네트워크를 통하여, 게이트웨이 노드에 전송하는 단순한 동작을 하는 센서가 있는 반면, 구조물 안전성 감시와 같이 모든 센서 노드가 동시에 데이터를 측정해야 데이터의 정확성과 유효성을 만족시킬 수 있는 경우도 있다. 하지만 이러한 동작을 하기 위해서는 모든 센서 노드들이 서로 시각 동기화가 되어 있어야 한다.

무선 센서 네트워크(Wireless Sensor Network)를 동기화하기 위한 방법으로 크게 2가지로 나뉘볼 수 있다. 첫 번째 방법은 외부 시간을 참조하여 센서 노드의 시간을 동기화 하는 방법이다. 이 동기화 방법은 GPS(Global Positioning System)를 이용하여, 위

성의 시간으로 전체 센서 네트워크의 시간을 동기화하는 것이다. 하지만 이러한 방법은 실내에서 사용이 용이하지 않다는 점과 GPS 모듈은 크기가 크고 비용이 비교적 비싸고 전력 소모가 크다는 단점을 가지고 있다.

두 번째 방법은 RTC(Real Time Clock)를 이용하는 것이다. 특정한 참조 노드를 선정하여, 로컬 클럭(Local Clock)을 이용하여 센서 네트워크의 모든 노드의 시간을 동기화한다. 하지만 이러한 동기화는 네트워크 지연시간, 데이터 처리 시간의 영향을 받기 때문에, 각 노드의 시간이 크게 차이가 나는 단점이 있다[1].

시각 동기화 방법에는 Receiver - Receiver 동기화 모델과 Sender - Receiver 동기화 모델이 대표적이며, 각 모델의 방법으로는 RBS (Reference Broadcast Synchronization)[2, 3]와 TPSN (Timing-sync Protocol for Sensor Network)[4], FTSP(Flooding Time Synchronization Protocol)[5] 방식이 있다.

일반적으로 센서 노드에서 특정한 주파수를 얻기 위하여 수정 발진기(Crystal Oscillator)를 사용한다. 이러한 수정발진기의 특정 주파수를 이용하여 센서 노드의 로컬 클럭으로 사용하게 된다. 하지만 수정발진기는 특성상 온도와 습도의 영향을 받아 시간의 정확도가 달라지며, 이에 따라 수정발진기의 속도의 차가 발생한다. 그림 1은 Imote2의 상대적 클럭 드리프트이다.

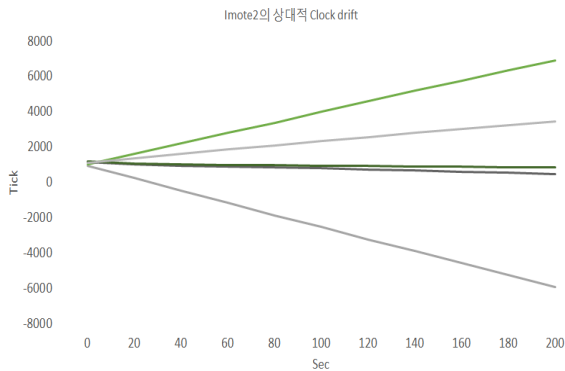


그림 1. Imote2의 상대적 클럭 드리프트
Fig. 1. Relative Clock drift of Imote2

본 논문에서는 Imote2에서 TPSN 동기화 알고리즘을 수행할 때, 발생하는 클럭 드리프트를 개선하기 위하여, 미리 클럭 드리프트를 측정하여 적용하는 방법으로 개선된 시간 동기화 알고리즘을 제안하고 이를 구현한다.

II. TPSN 알고리즘

TPSN은 무선 센서 네트워크를 계층화하여 구성하며, Sender-Receiver 동기화 모델로 제안되었다. TPSN은 2단계로 분류되며, 첫 번째 단계를 레벨 탐색 단계(Level Discovery Phase)라 부르며, 노드의 계층을 구성하여 무선 센서 네트워크를 계층화한다. 이후, 두 번째 단계로 그림 2와 같은 방식으로 노드 A와 B가 메시지 교환을 하고 동기화를 하게 된다. 이를 동기화 단계(Synchronization Phase)라 한다.

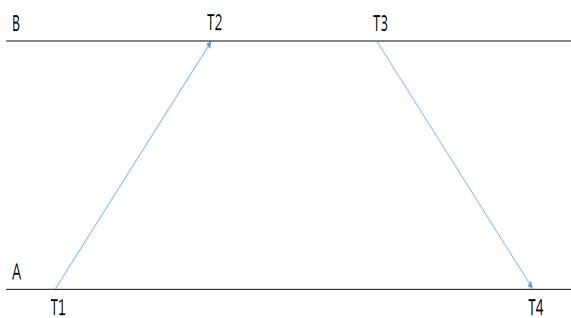


그림 2. 노드 A와 B의 양방향 메시지 교환
Fig. 2. Two-way Message Exchange between node A and B

그림 2에서 표현된 T1과 T4는 노드 A에서 수집된 로컬 클럭이다. 이와 비슷하게 T2와 T3은 노드 B에서 수집된 노드 B의 로컬 클럭이다. T1의 시간에 노드 A는 동기 펄스(Synchronization Pulse) 패킷을 노드 B에게 전송한다. 노드 B는 패킷을 T2에 수신하게 되며, T2는 T1+d와 같다. 이 수식에서 d와 d는 클럭 드리프트와 전파지연(Propagation Delay)가 된다. 노드 B는 수신

즉시 T3 시간에 노드 A에게 응답 (Acknowledgement) 패킷을 전송하게 된다. 이때 패킷은 노드 B의 레벨과 T1, T2, T3 클럭 값을 포함한다. 노드 A는 이 패킷을 T4에 수신 받는다. 노드 A는 클럭 드리프트와 전파지연을 추정하며, 다음과 같은 식으로 계산할 수 있다.

$$\Delta = \frac{(T2 - T1) - (T4 - T3)}{2}; d = \frac{(T2 - T1) + (T4 - T3)}{2} \dots(1)$$

추정된 클럭 드리프트를 이용하여, 노드 A는 이에 따른 로컬 클럭을 수정하며, 노드 B와 동기화를 완료한다. 또한 이때 Propagation delay는 로컬 클럭에 포함되어 있기 때문에 따로 적용하지 않는다.

III. 개선된 TPSN 구현 및 실험

3.1 개선된 TPSN 알고리즘

본 논문에서 제안하는 개선된 TPSN 알고리즘은 수정발전기의 특성을 기반으로 한다. 그림 1은 기준 노드와 그 외 나머지 노드들과의 상대적인 클럭 드리프트차이를 나타낸 것이며, 시간이 경과할수록 클럭 드리프트 차이가 크게 된다. 기준 노드를 기준으로 시간이 경과함에 따라 일정한 간격으로 멀어지는 것을 알 수 있다. 이러한 Drift 차이를 모든 센서 노드에서 측정한다. 센서 노드의 TPSN 동기화 알고리즘에 추정된 드리프트의 평균값을 로컬 클럭 보정에 사용한다. 이렇게 하면 기존 TPSN 알고리즘을 사용할 때 보다 TPSN의 동기화 오차를 효과적으로 감소시킬 수 있다.

3.2 개선된 TPSN 구현

본 논문에서 실험에 사용되는 센서 노드는 Imote2이다. Imote2는 PXA271 프로세서를 사용하여 13MHz~416MHz로 동작하며, RF 모듈은 CC2420을 사용하여 250kbps 속도로 통신이 가능하다. 또한 동작전압은 3.2V~4.5V를 사용하며 타이머 레지스터는 3.25MHz에서 동작하고, 카운팅 속도는 0.325μsec이다 [6, 7]. 또한 TinyOS-1.x와 TinyOS-2.x의 포팅이 가능하지만 본 논문에서는 SHM라이브러리[8]와 호환성을 유지하기 위하여, TinyOS-1.x를 사용하였다. 또한 PAX271의 타이머 레지스터는 32비트로 동작하여 32비트 이상의 클럭을 카운팅 할 수 없기 때문에 오버플로우 레지스터를 이용한다. 소프트웨어적으로 상위 32비트를 표시하여 64비트 크기의 클럭을 카운팅 한다.

본 논문에서 제안한 개선된 TPSN의 알고리즘을 구현하기 위하여, 상대적인 클럭 드리프트 값을 추정해야 한다. 이를 위하여, 먼저 그림 3과 같이 실험 환경을 구성한다. 또한 이를 이용하여 기준 노드와 측정할 센서 노드들을 기존의 TPSN 알고리즘을 이용하여 동기화를 100회 실시한다. 측정된 동기화 오차의 Tick 값(약 0.3μsec)을 표 1에 나타내었다. 표 1에서 평균적인 시간 동기화 오차는 832 Tick임을 보인다.

표 1. TPSN알고리즘의 시각 동기화 오차(단위: Tick, 1 Tick=0.307 μ sec)

Table 1. Time Synchronization Error of TPSN Algorithm (Unit: Tick, 1 Tick=0.307 μ sec)

	노드1	노드2	노드3	노드4	노드5
최솟값	1042	978	1007	972	1080
최댓값	799	803	805	801	801
평균	832	832	834	829	834

3.3 실험

단일 홉(Single Hop)환경에서 본 논문에서 제안하는 개선된 TPSN으로 센서 노드를 동기화 한 후, 각 노드들의 동기화 정확도를 알아보기 위하여 ATEN ADS 1102CAL 디지털 오실로스코프 (Digital Oscilloscope)와 추가로 데이터 수집용 노드를 그림 3과 같이 구성하여 각각의 센서 노드의 데이터를 수집하였다.

Imote2는 Mica2 모트와 같이 로컬 클럭을 PWM (Pulse Width Modulation) 신호로 출력하는 기능이 있다. 하지만 1MHz 모드에서만 출력이 가능하기 때문에 Imote2의 GPIO (General Purpose Input Output)를 출력포트로 사용하여 PWM 신호를 출력한다. 또한 데이터 수집 노드를 이용하여 센서 노드 동기화시 신호를 전송하여 주기적으로 MAC (Media Access Control) 계층에서 타임 스탬프 (Time Stamp)를 전송받아 데이터 수집용 PC로 전송한다. 그림4는 동기화시 Imote2의 GPIO 포트에서 디지털 신호의 변화를 보여준다.

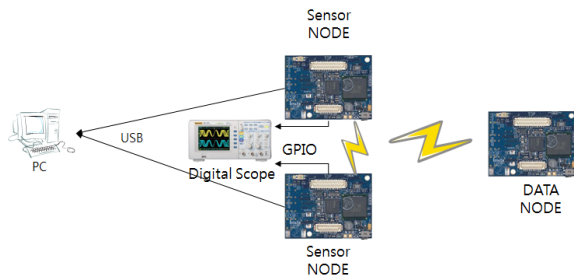


그림 3. 실험 환경
Fig. 3. Experimental Environment

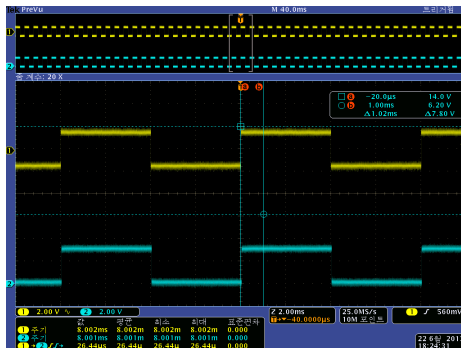


그림 4. 오실로스코프로 측정한 디지털 신호
Fig. 4. Digital Signal Measured by Oscilloscope

그림 5와 그림 6의 그래프는 시각 동기화 때 발생하는 동기화 오차의 분포를 나타낸 것이며, 표2는 이에 대한 결과를 정리하여 나타낸 것이다. 기존의 TPSN과 개선된 TPSN 알고리즘을 각각 100번을 독립적으로 실행하였다. 표 2에 따르면, 두 쌍의 센서 노드가 동기화할 경우 최대 동기화 오차는 대략 200 Tick이며, 최악의 경우 완벽히 동기화를 이루었다. 또한 평균 15 Tick(약 4.5 μ sec)의 차이로 동기화 하였다. 기존 TPSN 알고리즘을 적용하여 시각 동기화를 한 경우(그림 5)와 본 논문에서 제안한 개선된 TPSN 알고리즘을 이용하여 시각 동기화를 한 경우(그림 6)를 비교해 볼 때 개선된 동기화 알고리즘의 경우에 오차의 크기가 크게 감소하였다.

표 2. 개선된 TPSN알고리즘의 시각 동기화 오차(단위: Tick, 1 Tick=0.307 μ sec)

Table 2. Time Synchronization Error of Improved TPSN Algorithm (Unit: Tick, 1 Tick=0.307 μ sec)

	노드1	노드2	노드3	노드4	노드5
최솟값	0	1	0	0	0
최댓값	188	175	174	191	161
평균	15.06	13.21	16.77	17.95	13.27

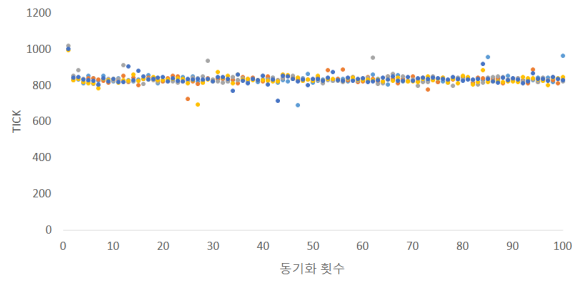


그림 5. TPSN 알고리즘을 이용한 시각 동기화 오차
Fig. 5. Time Synchronization Error Using TPSN Algorithm

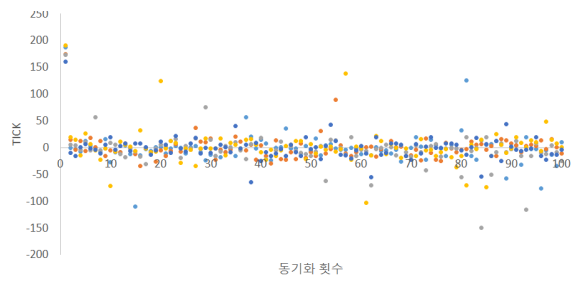


그림 6. 개선된 TPSN 알고리즘을 이용한 시각 동기화 오차
Fig. 6. Time Synchronization Error Using Improved TPSN Algorithm

IV. 결론

본 논문은 무선 센서 네트워크에서 시각 동기화 방식인 TPSN

알고리즘을 손쉽게 개선하는 방법을 제안하고 구현하였다. 제안한 방법은 임의의 기준 노드를 중심으로 주변 노드들의 각각의 클럭 드리프트를 실험적으로 구하는 1 단계와, TPSN 방식으로 동기화 시 1 단계에서 구한 클럭 드리프트 값을 이용하여 TPSN 동기화 시각을 보정해주는 2 단계로 구성된다. 그러나 모든 센서 노드에 대응되지 않기 때문에 이기종의 센서 노드에 적용 시 클럭 드리프트를 다시 측정해야하는 단점이 있다. 하지만 Imote2에서 TPSN 방식의 동기화 오류를 손쉽게 최소화할 수 있다는 장점이 있다. 향후 미리 클럭 드리프트 측정 없이 시각동기화 오차를 보정하면서 오차는 최소화하는 알고리즘의 연구가 필요하다.

Acknowledgment

이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. 2010-0024242)

참고문헌

- [1] Min-Je Kim, Kyung-Sik Jang. Transmission Delay Adopted Time Synchronization Method for Wireless Sensor Network. KIMICS Conference p. 497-500, October 2010.
- [2] Elson, J. E. Time Synchronization in Wireless Sensor Network. Ph.D. Thesis, University of California, Los Angeles 2002.
- [3] Elson, J. E., Girod, L., and Estrin, D. Fine-Grained Network Time Synchronization using Reference Broadcasts. The Fifth Symposium on operating systems Design and Implementation (OSDI), p.147-163, December 2002.
- [4] Ganeriwal, S., Kumar, R., and Srivastava, M. B. Timing-sync Protocol for Sensor Network. The First ACM conference on Embedded Networked Sensor System(SenSys), p.138-149, November 2003.
- [5] Miklós Maróti., Branislav Kusy., Gyula Simon., Ákos Lédeczi., The Flooding Time Synchronization Protocol. ACM SenSys '04:Proceedings of the 2nd international conference on Embedded Networked Sensor System, November 2004.
- [6] Marvall PXA270 Processor Developers Manual.
- [7] Intel Mote2 Engineering Platform Data sheet.
- [8] Structural Health Monitoring Using Smart Sensors NSEL Report Series Report No. NSEL-001 November 2007.