Advanced Science and Technology Letters
Vol.28 (CIA 2013), pp.104-109
http://dx.doi.org/10.14257/astl.2013.28.20

Estimation of Packet Delay Components for Time Synchronization in Wireless Sensor Networks

Shiu Kumar[1] , Abel Avitesh Chandra[2], Bold Sanjaa[3], Kyeong Hur[4], Seong Ro Lee[5]

[1,2,3,5]Department of Information Electronics Engineering,
Mokpo National University, South Korea
[4]Department of Computer Education, Gyeongin National University of Education,
South Korea
[1]kumar.shiu@hotmail.com, [2]abelavit@yahoo.com, [3]snj_goopy@yahoo.com,
[4]khur@ginue.ac.kr , [5]srlee@mokpo.ac.kr

Abstract. In this paper, a novel approach for estimating and compensating synchronization message packet delays is presented. The main goal of this research is to minimize the uncertainties in the synchronization packet delays to achieve higher synchronization accuracy, and to design a synchronization scheme that is efficient and has a low communication overhead.

Keywords: Packet Delays, Timestamp, Time Synchronization, Wireless Sensor Networks (WSNs).

1 Introduction

Time synchronization of the nodes has been an open research issue in WSNs [1] having a very long history, which is very important for energy conservation. Due to continuous technological advancements, hardware is becoming cheaper and hardware power consumption is decreasing, which both make WSN increasingly more viable for use in real-world applications. Currently, WSNs is widely employed for various applications such as monitoring [2], surveillance [3], industrial and process automations [4], etc. WSNs have several distinctive characteristics that often prevent the use of existing time synchronization protocols in this field. First, the time synchronization scheme must be implemented in an energy-efficient way as the amount of energy available to the sensors is limited since they are battery powered. Second, wireless communication has limited bandwidth which restricts frequent message exchanges among the sensor nodes. Third, sensor nodes are very small in size posing limitations on computational power and storage space.

Synchronization is usually based on some sort of message exchange among sensor nodes. However, these message exchanges that take place during synchronization are subject to numerous non-deterministic delays, which in terms of end-to-end single-hop transmission include the send time, access time, transmission time, propagation time, reception time and receive time.

Typical time synchronization algorithms in WSNs include reference broadcast synchronization (RBS) [6], timing-sync protocol for sensor networks (TPSN) [7] and flooding time synchronization protocol (FTSP) [8]. The RBS [6] protocol is based on

ISSN: 2287-1233 ASTL

the receiver-receiver synchronization technique that synchronizes a cluster of wireless sensors within transmission range of the reference sensor node. A third party broadcasts the reference beacon to all the receivers in range. TPSN [7] is based on the sender-receiver synchronization comprising of two phases: a level discovery phase and a synchronization phase. The whole network is structured into a hierarchical tree topology with the master node at its core in the level discovery phase.

FTSP [8] is a multi-hop sender-receiver synchronization method aimed for resource limited wireless platforms requiring stringent accuracy, using low communication bandwidth. MAC-layer time stamping has been utilized together with error compensation including clock skew estimation. To mention, a few other time synchronization protocols reported are pairwise lightweight protocol [9], energy efficient and rapid time synchronization (EETS) [10], and average time synchronization (ATS) [11].

2 The Proposed Packet Delay Estimation Scheme

This proposed scheme employs a single radio message to synchronize a node's local clock thus accounting for an energy-efficient and low communication overhead system. The sensor nodes used for this research consists of the Arduino Mega 2560 microcontroller together with the Nordic 16MHz NRF24L01+ single chip 2.4GHz radio module operated at a speed of 250kbps.

2.1 Proposed System Architecture and Operation

The overall architecture of the proposed system is shown in Fig. 1. User authentication is required to access the system. Upon authentication, the user can either select to have the time and date information updated automatically or the user can enter the information manually. The Ethernet shield is used to retrieve the information from the server in the automatic update mode. Once the time and date information is available, the reference node synchronizes its local clock and proceeds to send the timestamped information to the slave nodes for time synchronization.

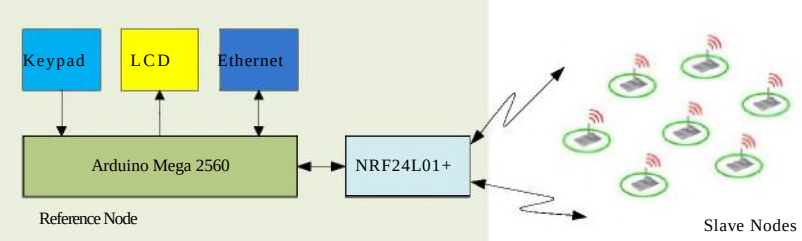


Fig. 1. The proposed system architecture

2.2 Synchronization Message Packet Layout

The synchronization message packet layout is one of the important factors that assist in the compensation of the estimated nondeterministic delays. Fig. 2 shows the

general packet layout of the radio message that has been employed where Address is the receiver address. The Packet Control Field (PCF) consists of the Payload Length (6 bits), Packet Identification (PID, 2 bits), and the No Acknowledgement Flag (1 bit). The payload contains the timestamp information. Finally, the Cyclic Redundancy Check (CRC) is a mandatory error detection mechanism.

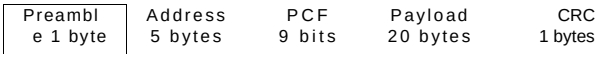
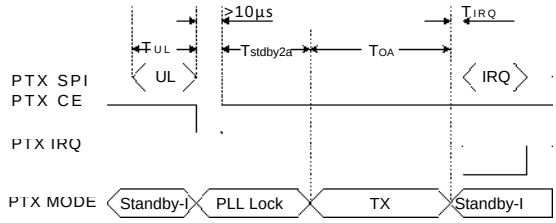


Fig. 2. The synchronization radio message packet

layout **2.3 Transmitter Packet Delay Estimation**

The proposed scheme intends to minimize the uncertainties in the synchronization packet delay by estimating and compensating for these nondeterministic delays. In order to achieve this, the timing sequence diagram for the NRF24L01+ radio module shown in Fig. 3 has been thoroughly investigated and studied. The terms PTX refer to the Primary Transmitter, PTX CE is the Chip Enable pin, PTX mode indicates the mode in which the transmitter is at various stages, PTX IRQ is the interrupt pin, $_{TUL}$ is the time taken to upload the payload to the PTX FIFO and $_{TOA}$ is the time taken to transmit the radio message over the air. $_{Tstdby2a}$ is the time taken for the radio to get into transmit mode while the $_{TIRQ}$ is the time taken for the interrupt to be generated after the data in



the TX FIFO has been transmitted over the air.

Fig. 3. The timing sequence of the NRF24L01+ radio module

Timestamping is performed at the MAC layer and the uncertainties in the PTX are estimated and added to the timestamp just before the packet payload is uploaded to the PTX FIFO. The total transmitter delay, comprising the send, access and transmission time, is given by (1). Where PD_T refers to the transmitter packet delay in number of clock cycles, $_{TCE}$ is the duration for which the PTX CE pin is low and T_t refers to the software processing time of the transmitter.

Several experiments were conducted to determine the parameters in (1). An oscilloscope was used to measure the time taken for different payload size to be uploaded to the PTX FIFO. The result was thus analyzed and the equation to estimate the payload upload time $_{(TUL}$ in µs) for different payload lengths was obtained as given in (2).

$$PD_T = 1620.50 \quad + T \quad + T \quad \text{time taken for} \quad + T_t + T_t + T_t \quad . \tag{1}$$

$$T_{UL} = 2.06(PL) + 20.48 . \quad (2)$$

The value of $T_{stdby2a}$ is given to be 130 its [12] while T_{IRQ} for air data rate of 250kbps is given as 17its. An oscilloscope has been used to measure the time taken from the time when the PTX CE pin goes high to the time when the interrupt on PTX IRQ pin goes high. The value of the transmission time over the air, T_{OA} (in its) was thus obtained by subtracting the values of $T_{stdby2a}$ and T_{IRQ} from these measured values and is given by (3).

$$T_{OA} = 32(PL) + 260.15 . \quad (3)$$

The time for which the PTX CE pin is low was found to be a constant value of 21.12its. The software processing time of the transmitter, T_t, is the time taken to execute (4), which is the compensation of the non-deterministic delays and is executed after the time stamping. It involves addition of the estimated uncertainties to the timestamp data. The TCNT1 is a 16 bit register.

$$TCNT1 = TCNT1 + PD_T, \quad (4)$$

2.4 Receiver Packet Delay Estimation

The receiver delay uncertainties accounts for the propagation time and the receive time which includes the time taken to validate the received message. The reception time is the same as the transmission time and has already been compensated for at the transmitter. The total receiver delay is given by (5). On the primary receiver (PRX), the time difference between the PTX IRQ and the PRX IRQ denoted as $T_{IRQ,}$ gives the propagation time. This value is independent of the message length; however it varies due to interference present from other sources and other tasks that are being run on the nodes at that instance. The average time was found to be 6.185its.

$$PD_R = 62.5^{1000}(T_{IRQ} + T_{INT} + T_{recv} + T_r). \quad (5)$$

At the receiver, time stamping is performed as soon as the radio message is validated and available in the PRX FIFO, which is when the PRX IRQ interrupt is generated. An interrupt was generated on the MCU using this interrupt pin to read the time at which the message is received. However, time (T_{INT}) is taken for the MCU interrupt to be generated from the time the PRX IRQ pin interrupt is generated. Several samples of this time were recorded using an oscilloscope and an average value of 3.90its which is approximately 63 clock cycles was. Upon performing timestamping, the message is read by the MCU from the PRX FIFO. Another timestamp is taken upon successfully reading the message in order to determine the time taken to receive the message, $T_{recv.}$ T_r is the software processing time of the receiver that is the time taken to execute (6). And finally the slave nodes local clock is synchronized with this modified timestamp value, which compensates for both the transmitter and receiver synchronization packet delays.

$$TCNT1 = TCNT1 + PD_R, \quad (6)$$

3 Results and Discussion

Experiments were carried out in an indoor environment with the presence of interference from Wi-Fi and other radio frequency devices while running multi-tasks to consider the real time environment. First, only two nodes were used with one being the reference node and the other being the slave node. The receiver node starts off by initializing the radio module and then it waits for the sender's radio message containing timestamp to synchronize its local clock. On the other hand, the sender node also starts off by initializing the radio module. It then starts to send the time stamp message over the radio to the receiver(s) after every 10ms for the purpose of testing the synchronization accuracy. Software algorithm was written, for both reference and slave nodes, to output high at one of the MCU outputs at every timer overflow interrupt. These two output pins were used to measure the synchronization accuracy using an oscilloscope.

The synchronization accuracies were recorded and analyzed. The results for the average single hop synchronization accuracy is summarized and compared with other typical time synchronization algorithms as shown in Table 1.

Table 1. Comparision of the proposed method with other typical time synchronization protocols in WSNs.

	RBS	TPSN	Proposed Method
Average error (in µs)	29.13	16.9	2.20
Worst case error (in µs)	93	44	4.64
Best case error (in µs)	0	0	0
Percentage of time error is less than or equal to average error	53	64	56
Energy Efficiency	High	High	High
Complexity	High	Low	Low

4 Conclusions

In this research paper the synchronization message delays have been estimated and compensated for; achieving a synchronization scheme that employs only a single radio message to synchronize the slave nodes local clock having enhanced synchronization accuracies compared to most representative clock synchronization protocols such as RBS and TPSN. Furthermore, this scheme achieved a significant reduction in the overall energy consumption of the nodes and a low communication overhead.

Acknowledgments. "This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the NIPA(National IT Industry Promotion Agency)"(NIPA-2012-H0301-12-2005), This work was supported by Priority Research Centers Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology(2009-0093828)

References

1. Sundararaman, B., Buy, U., Kshemkalyani, A.D.: Clock synchronization for wireless sensor networks: A survey. Ad Hoc Networks. 3, 281--323 (2005)
2. Bhattacharjee, S., Roy, P., Ghosh, S., Misra, S., Obaidat, M.S.: Wireless sensor network-based fire detection, alarming, monitoring and prevention system for bord-and-pillar coal mines. J. Sys. and Software. 85, 571--581 (2012)
3. Qun, L., Rus, D.: Global clock synchronization in sensor networks. IEEE Trans. on Comp. 55, 214--226 (2006)
4. Guo, Y.: Energy attenuation-based time synchronization for mobile underwater sensor networks. J. China Univ. Posts and Tel. 19, 57--64 (2012)
5. Kopetz, H., Ochsenreiter, W.: Clock synchronization in distributed real-time systems. IEEE Trans. on Comp. 36, 933—940 (1987)
6. Elson, J., Girod, L., Estrin, D.: Fine-grained network time synchronization using reference broadcasts. In: 5th symposium on Operating systems design and implementation, pp. 147--163 Digital Library, New York (2002)
7. Ganeriwal, S., Kumar, R., Srivastava, M.B.: Timing-sync protocol for sensor networks. SenSys '03 (2003)
8. Maróti, M., Kusy, B., Simon, G., Lédeczi, Á.: The flooding time synchronization protocol. SenSys '04 (2004)
9. Greunen, J.V., Rabaey, J.: Lightweight time synchronization for sensor networks. In: 2nd ACM international conference on Wireless sensor networks and applications, pp. 11--19 ACM Digital Library, New York (2003)
10. Kim, B.-K.: Energy-efficient and rapid time synchronization for wireless sensor networks. IEEE Trans. on Consumer Electronics. 56, pp. 2258 - 2266 (2010)
11. Schenato, L., Fiorentin, F.: Average timesynch: A consensus-based protocol for clock synchronization in wireless sensor networks. Automatica. 47, 1878-1886 (2011)
12. ASA, N.S.: Nrf24l01+ product specification: Datasheet (2008)

