Distance Measurement in Wireless Sensor Networks with Low Cost Components

Heiko Will*, Stefan Pfeiffer*, Stephan Adler*, Thomas Hillebrandt* and Jochen Schiller*

*FU Berlin/AG Computer Systems, Berlin, Germany. Email: [heiko.will, stefan.pfeiffer, stephan.adler, hillebra, jochen.schiller]@fu-berlin.de

Abstract—**Several ways to estimate the position of a Wireless-Sensor-Network (WSN) node have been discussed in the past years. Unlike in outdoor solutions where the Global Positioning System (GPS) could be used in most applications, a general solution for indoor usage has not been found. The few existing indoor solutions on the market are highly specialized and rely on infrastructure or on very expensive special designed hardware. Both - infrastructure and expensive hardware - does not fit well into most scenarios where WSNs are commonly used because of the adhoc characteristics and the large amount of nodes of such installations. In this paper we present a solution to get a precise estimation of the distance between two nodes without the needs for special purpose chips or a redesign of already existent nodes. We use radio runtime measurement to calculate the distance between nodes and present algorithms to refine the measurements. A comparison with a professional solution which is available on the market is also presented.**

Index Terms—**Radio Runtime Measurement, Indoor Localization, Distance Measurement, Wireless Sensor Networks**

I. INTRODUCTION

A. Motivation

The indoor usage of WSNs can be divided into two cases. In the first case the building is well defined and the WSN is designed especially for use in this building. In this case the deployment of special localization infrastructure like ultrasonic beacons is possible. The second case is the general case where nodes could move into unknown buildings. This case is met by rescue scenarios where the area of deployment is known only minutes before and infrastructure cannot be set up inside the buildings. For this general case all hardware has to be integrated into the nodes and only some anchors could be placed outside the building. We focus on this case in this paper.

The distances can be gathered in several ways. For example estimated by received signal strength indicator (RSSI) or radio runtime measurement. Because the RSSI value in indoor deployments is influenced by many parameters, a distance estimation based on RSSI value is highly imprecise [1]. To use radio runtime measurement, normally special designed transceivers are used which often use wide spectrum or ultra wide spectrum techniques. To integrate these into an existing WSN a redesign of the whole platform is necessary which leads to high cost, additionally to the also high costs of the special purpose transceiver.

B. Contribution

In this paper we present a solution for radio based runtime measurement which can easily be integrated into existing

WSNs with only minimal hardware modifications which does not lead to the need of a hardware redesign. We present an implementation on the MSB-A2 [2] WSN nodes. We address the problems of doing precise time measurements on the I/O pins of a microcontroller and taking care about the jitter which results from different clock sources of the microcontroller and the transceiver. So the contribution of this paper is a proposal for an easy to implement and precise solution for distance measurement in WSNs.

The paper is organized as follows: In chapter II we discuss and compare similar approaches. In chapter III we briefly introduce the techniques of radio runtime measurement with common microcontrollers and transceivers. We present our system design and some implementation hints in chapter IV. In chapter V we compare our solution with a professional measurement system which is available on the market and finally we give a conclusion and outlooks in chapter VI.

II. RELATED WORK

A. RSSI-Based Approaches

Electromagnetic waves lose power while being propagated through the air. In an theoretical setting the waves will propagate in a spherical form where the radius r of the sphere is the distance between transmitter and receiver. Because the performance of a real antenna and other physical effects the simple sphere formula does not give a good assumption for the distance. Harald T. Friis [3] proposes (1) for the estimated power at the receiver P_R in dependence of the gain of both antennas G_T and G_R and the wavelength λ.

$$P_R = P_T \frac{G_T G_R \lambda^2}{(4\pi)^2 r^n}, (n = 2 \text{ for open space propagation}) \tag{1}$$

In theory the distance between transmitter and receiver could easily be calculated with this formula if antenna gain, transmitting power and frequency are precisely known. Environment parameters like power loss in different transmission medias like air, concretes, steel, etc. are put into account with the constant parameter n. The estimation of n in real world scenarios could only be done by experiments in the selected environment. So a distance measurement with the RSSI value as a singular computation base is only reliable under laboratory conditions [1].

B. Propagation delay

The time a radio signal travels between transmitter and receiver is called propagation delay or time of flight (TOF). If the propagation delay T_{TOF} is known, the distance d between transmitter and receiver can be calculated by multiplying the constant for speed of light c with T_{TOF}. In our case TOF is expected to be the magnitude of nanoseconds.

C. Round trip time of flight

The round trip time of flight (RTT) method is the main measurement method for T_{TOF} used in this work. A WSN node A starts to measure the time T_{RTT} as the packet leaves the transceiver to B, when the reply packet from B arrives at the transceiver of A we measure T_{RTT}. The time node B needs to process the packet is called time to compute packet T_{TCP}.

The time of flight T_{TOF} can be calculated as follows

$$T_{TOF} = \frac{T_{RTT} - T_{TCP}}{2} \qquad (2)$$

the distance between A and B can be easily calculated as

$$d_{RTT} = T_{TOF} \times c \qquad (3)$$

One major advantage of that method is that no time synchronization of the communicating partners is necessary and it can be done with standard hardware. Nevertheless the challenge in this method is to get a very precise value for T_{TCP} and T_{RTT}. The faultiness of the distance measurement d_{RTT} using that method is described as

$$d_{RTT} = d + \epsilon_{RTT}^{LOS} + \epsilon_{RTT}^{NLOS} \qquad (4)$$

by Bahillo et al. in [4]. This equation contains two error components ϵ_{RTT}^{LOS} for the error which appears when ranging in a line of sight setting and an additional error ϵ_{RTT}^{NLOS} when ranging in a non-line of sight environment. While a big factor in ϵ_{RTT}^{NLOS} are multipath effects, as described in [1], their negative impact can be reduced by using an empirical approach as described in [5]. The other source of error ϵ_{RTT}^{LOS} depends mostly on uncertainties and noise in the hardware. Especially jitter effects play a key role in the error component. In our work we concentrated on analyzing and eliminating the impact of ϵ_{RTT}^{LOS}.

D. Similar Approaches

Bahillo et al. describe a system which implements a RTT measurement method using standard 802.11 WLAN transceivers [4]. Although the approach is strongly related to our work, the main difference is that we do not need any additional hardware. Bahillo et al. use a custom printed circuit board (PCB) for time measurements while we use the embedded timer of a microcontroller. Our system can be applied to any sensor node of the MSB-A2 [2] family without any hardware changes. We also use a lower frequency band which has less resolution but better abilities towards multipath effects [5].

The nanotron company also uses a RTT based method to range with other members of the network. The company developed a complete transciever that integrates localization functions in hardware [6].

III. TIME OF FLIGHT MEASUREMENTS WITH COMMON HARDWARE

A. Common Radio Transceivers in WSNs

Most radio transceivers used in WSNs like Chipcons or Atmels are low-IF (intermediate frequency) receivers. They have only a very small analog part where after some filtering the signal is put directly into one or more mixers to separate it from its carrier frequency and transform it to a much lower frequency. This lower frequency is called intermediate frequency.

The recognition and interpretation of signals is done completely in the digital part of the transceiver. If the data rate is known the bit frequency of the incoming data is detected while the preamble is sampled.

B. When to Measure

For precise measurement of the TOF on a common microcontroller we need an appropriate signal as source for the time measurements. Most transceivers used in WSNs provide several sources like *first bit of packet*, *last bit of preamble*, or *last bit of packet* which are signaled through changing edges on a certain pin. All of these signals could be used while processing a packet. The preamble normally is an alternation of ones and zeros to provide bit synchronization to the receiving radio. So all indicators referring to the preamble like *first bit of packet* are not reliable. Because a packet could only be detected after bit synchronization and therefore the span between the incoming *first bit of the preamble* and the recognition of a packet is variable. A better approach is to use the sync word as an indicator, because it is sent directly after the preamble. Because the sync word directly follows bit synchronization, it has the lowest possible bit jitter.

C. Hardware Limitations

To measure the TOF of a radio packet we utilize the *last bit of sync word* pin of the transceiver chip to start a measurement on an outgoing packet and the same pin to stop the measurement on an incoming packet as described in section II-C. Included in the RTT is the time which the addressed node needed to compute the incoming packet and assemble the outgoing packet. This time could also be measured with the same pins from the transceiver and then transmitted to the requesting node. If these time spans are subtracted we get the raw transmission time of the two packets.

This transmission time is far from being accurate. Several error sources have to be taken into account. First there is the jitter caused by the independent clocks of the microcontroller C_u and the transceiver C_t. This jitter is added in several stages of the measurement. If two independent clock sources are used the signal could only be sampled on a rising edge of

our system clock source. This jitter J_c could be estimated as follows:

$$0 \le J_c \le \frac{1}{C_u} \qquad (5)$$

The second form of jitter which is added is the jitter caused by the digital part of the transceiver. The transceiver transforms the waveform to a digital signal using a clock C_t. Because this clock is independent to the clock of the corresponding transceiver the rising edge for a detected bit could be delayed one clock cycle of C_t. This jitter J_t could be estimated as follows:

$$0 \le J_t \le \frac{1}{C_t} \qquad (6)$$

Overall two timestamps are taken to calculate the TOF. Each timestamp N suffers from the jitter effects J_{tN} and J_{cN}. Two of the timestamps are used to calculate the time between the sending of the initial packet and the receiving of the ACK packet on the initiating node. On the corresponding node also two timestamps are taken to calculate the computation time between receiving a packet and sending the first bit of the ACK packet. The measured RTT R_t is estimated as follows:

$$R_t = J_{t0} + J_{c0} + TOF_R + TOF_A + J_{t3} + J_{c3} \qquad (7)$$

In (7) TOF_R is the TOF for the request packet and TOF_A is the TOF for the answer packet.

To calculate the estimated TOF_e of a RTT the computation time has to be subtracted from R_t. Including jitters it could be estimated as:

$$TOF_e = R_t - (J_{t1} + J_{c1} + T_{TCP} + J_{c2} + J_{t2}) \qquad (8)$$

As introduced in (6) another kind of jitter occurs, which is sourced inside the transceiver and could hardly be measured. The time the transceivers digital part needs to recognize a bit can vary between on wave of the intermediate frequency and the bit length relating to the modulation, data rate and link quality. Because of the other jitters also occurring during measurements this jitter cannot be measured with common equipment and can only be assumed over a large number of measurements. Finally, we will estimate the TOF_e as the measured RTT decreased by T_{TCP} and an offset O_m which holds this bit jitter J_b.

$$TOF_e = R_t - T_{TCP} - O_m \qquad (9)$$

Also included in the offset O_m is the propagation time of the signal in the analog part of the transceiver and the time to demodulate the signal in the digital part.

Additional to all of these error sources the clock drifts of the clock sources have also to be taken into account. Under normal circumstances the TOF of a packet is that small that the error sourcing from clock drift is minimal compared to the jitters.

The jitters could be minimized if the microcontroller and the transceiver are driven with the same clock source which would lead to a redesign of existing hardware if possible at all.

IV. Implementation

A. System Setup

For our reference implementation MSB-A2 sensor nodes are used. The MSB-A2 has an LPC2387 microcontroller [7] and a CC1101 radio transceiver [8]. The microcontroller has an ARM7 core and a 72MHz clock. The CC1101 is driven with a 26MHz clock and uses the 868MHz radio band.

As operation system we are using the FireKernel microkernel [9].

B. Implementation Overview

Regarding III-C and with the knowledge about the exact clock rates of all relevant components we could calculate the estimated values for all error sources and take them into account. Only O_m remains unestimated because the bit jitter is independent from the clock rates. We have to measure O_m beforehand for every used transciever and store it on a memory card. To improve accuracy we have implemented a two way range measurement protocol which allows us to measure the TOF from node A to B and from node B to A in one cycle of our algorithm. First node A sends a ranging request (RR) to B which answers with an ranging acknowledge (RACK). In this step the TOF timer of A is started and B has measured the computation time and also started a TOF timer on sending out the RACK. A stops the TOF time on receiving the RACK and starts measuring the computation time immediately and sends out another packet to B (REACK). On receiving this packet B stops its TOF timer. The last step is to transmit the measured TOF and the computation time to node A. Node A has now two measured TOFs and the corresponding computation times. Because we normally have more than one of these measurements, we skip the last packet which only transmits the measured times to node A and piggyback those values in the RACK of the next step.

V. Evaluation

First we estimated by experiments if our assumptions about the error sources and the error distribution were correct. The experiments showed that the jitters where assumed correctly and they are equally distributed. Fig. 1 shows the transceiver jitter measured with a digital storage oscilloscope (DSO).

To evaluate the accuracy we first did some outdoor experiments to prove the general fitness of our system. It is shown in Fig. 2 that an average accuracy of 3m could be achieved if the mean value out of 25 single measurements is chosen, which is very promising.

For a more practical evaluation we set up an indoor experiment where we walked along a 30m long corridor in our university. We did the same measurement with a professional ranging measurement system from Nanotron, the Nanopan 5375 RF Module mounted on the same WSN platform [6]. As
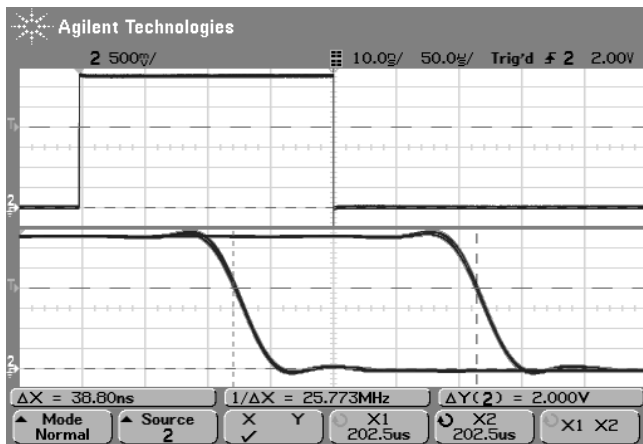
Fig. 1. The jitter of the transceiver while measuring the end of sync word pin. The origin of the jitter is the 26 MHz clock of the transceiver.
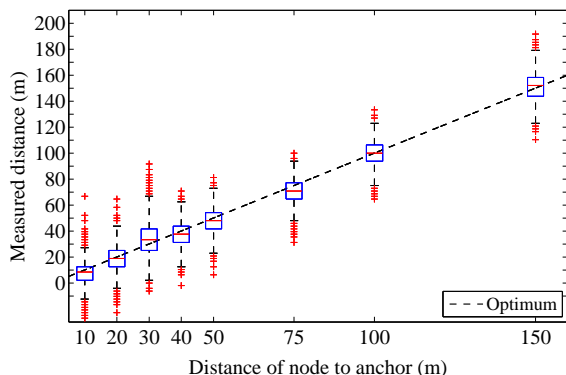

Fig. 2. The box plot shows the accuracy of outdoor measurements under line of sight conditions. The whisker length is 1.5 times of the interquartile distance.
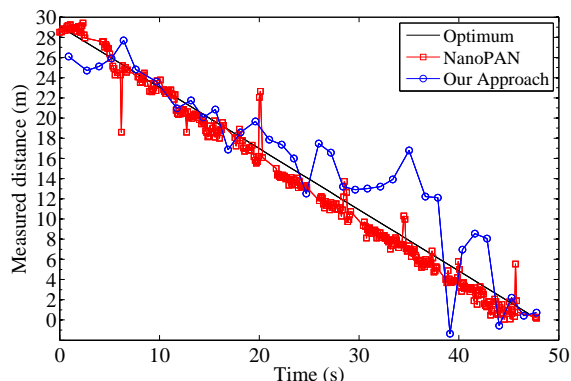

Fig. 3. Accurance Comparrison between Nanopan 5375 range measurements and our system. Indoor Measurement on a 30m long corridor.

shown in Fig. 3 our system behaves similar to the reference system. To compare the two systems in a real word scenario we did a walk through our office building and calculated the position with a trilateration algorithm and median filtering on the positions. The result was comparable to the results of the comparison system. One major drawback is that our system uses a relatively high bandwidth. To achieve a single ranging the reference System uses the channel for around 6ms. To compensate the jitters in our system we have to do 20 ranging cycles which blocks the air for around 30ms. This does not influence the accuracy comparison, because the accuracy of the reference system is not increased significantly with more measurements.

VI. CONCLUSIONS AND OUTLOOK

Our experiments showed that our proposal has some great advantages to common methods for range determination in WSNs without special hardware, especially to RSSI based methods. We also showed that we can compete with special hardware which is used in several installations today. We propose our work as a generic solution for range measurements in existing WSNs or WSNs where the use of special hardware is not possible.

Future work will address the fusion of RSSI and link quality based methods and our approach to increase accuracy. If we could estimate the distance to the anchors with a single packet, we could use the available bandwidth for ranging only with promising anchors. Also more sophisticated statistical approaches are needed to lower the used bandwidth by fewer ranging packets.

REFERENCES

[1] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal, "Locating the nodes: cooperative localization in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 54–69, July 2005.

[2] M. Baar, H. Will, B. Blywis, T. Hillebrandt, A. Liers, G. Wittenburg, and J. Schiller, "The scatterweb msb-a2 platform for wireless sensor networks," Freie Universität Berlin, Tech. Rep., 2008. [Online]. Available: ftp://ftp.inf.fu-berlin.de/pub/reports/tr-b-08-15.pdf

[3] T. Kvaksrud, "DN018 - Range measurements in an open field environment," Texas Instruments, Design Note, 2008. [Online]. Available: http://www.ti.com/litv/pdf/swra169a

[4] A. Bahillo, S. Mazuelas, R. M. Lorenzo, P. Fernandez, J. Prieto, R. J. Duran, and E. J. Abril, "Hybrid RSS-RTT localization scheme for indoor wireless networks," *EURASIP Journal on Advances in Signal Processing*, 2010.

[5] H. Hashemi, "The indoor radio propagation channel," *Proceedings of the IEEE*, vol. 81, no. 7, pp. 943 – 968, July 1993.

[6] F. Schlichting, "nanoLOC TRX: Praezise Abstandsbestimmung und Lokalisierung mittels Laufzeitmessung (RTOF) durch Einsatz der 2,4 GHz Chirp Spreiztechnologie (CSS)," in *Tagungsband Wireless Automation: Funkgestützte Kommunikation in der industriellen Automatisierungstechnik*. VDI Verlag GmbH, February 2008, pp. 61 – 70. [Online]. Available: http://www.nanotron.com/EN/events/presentations/Schlichting-Wireless-Technologies-190907.pdf

[7] "LPC2387 datasheet," NXP, Eindhoven, Netherlands. [Online]. Available: http://www.nxp.com/documents/datasheet/LPC2387.pdf

[8] "CC1101 datasheet," Texas Instruments, Texas, USA. [Online]. Available: http://focus.ti.com/lit/ds/swrs061f/swrs061f.pdf

[9] H. Will, K. Schleiser, and J. Schiller, "A real-time kernel for wireless sensor networks employed in rescue scenarios," in *Proc. of the 34th IEEE Conference on Local Computer Networks (LCN)*. IEEE Press, 2009, pp. 834 – 841.