Computer Communications 186 (2022) 80–89

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Time synchronization enhancements in wireless networks with ultra wide band communications

Juan J. Pérez-Solano, Santiago Felici-Castell *, Antonio Soriano-Asensi, Jaume Segura-Garcia

Departament de Informática, ETSE, Universitat de València, Avd. de la Universidad S/N, 46100 Burjassot, (Valencia), Spain

ARTICLE INFO

Keywords:
Time synchronization
Ultra wide band
IEEE 802.15.4
Linear regression
Wireless sensor networks

ABSTRACT

The emergence of low cost Ultra Wide Band (UWB) transceivers has enabled the implementation of Wireless Sensor Networks (WSN) based on this communication technology. These networks are composed of distributed autonomous low cost nodes (also known as motes) with their own processing unit, memory and communications. Usually these nodes are power-limited and due to the poor performance and quality of their clocks, time synchronization is in the order of milliseconds and in some specific scenarios till microseconds. The integration of commercial UWB transceivers in these nodes can improve the synchronization accuracy. In particular, we focus on WSN nodes based on off-the-shelf commercial products and commodity hardware. In this paper we analyze step by step, from a practical and experimental point of view, the different elements involved in the time synchronization using UWB technology on WSN with static nodes. From our experimental results, with timestamps captured during the packet exchanges, we analyze and discuss the application of different communication schemes and simple statistical methods (in order to be run in WSN nodes). The results obtained with timestamps captured at the UWB transceivers and by using linear regression show that the lowest time synchronization error achieved between two nodes is 0.14 ns. Employing the same setup and performing the synchronization with the timestamps captured internally at the microcontrollers of the nodes, the error rises to 31 ns, due to the higher time period of the microcontrollers' timers and the inaccuracies that affect the acquisition of the timestamps. Nevertheless, the synchronization of the microcontrollers' clocks allows the setting up of a common time reference at the network nodes, enabling the implementation of applications with tight synchronization requirements, such as collaborative beamforming and ranging.

1. Introduction

The combination of Ultra Wide Band (UWB) technology and Wireless Sensor Networks (WSN) has paved the way to new and enhanced applications for monitoring, surveillance, ranging [1], collaborative beamforming [2], as well as adding advanced features such as improved duty cycling to extend the network life time and in contention based channel access, to improve network utilization [3]. These WSNs are still based on distributed and autonomous wireless nodes, each of them with its own processing unit, memory and wireless communications. Among the different alternatives to improve the physical layer proposed in the IEEE 802.15.4, Direct Sequence Ultra Wide Band (DS-UWB) is the most relevant, especially for WSN with static nodes. The goal of the work presented in this manuscript is to use DS-UWB technology in order to improve time synchronization among the wireless nodes, keeping in mind the low cost and the low power consumption features.

DS-UWB is a radio technology that uses very low signal power for short-range, with high-bandwidth communications, higher than 500 MHz. UWB was formerly known as pulse radio, since it is based on transmitting short pulses that occupy the entire bandwidth. These short pulses can be modulated (by position/phase, duration, polarity and amplitude), thus most signal reflections do not overlap the original pulse, and then they are less sensitive to multipath fading. The large bandwidth used in UWB technology allows for high resolution estimation of the channel impulse response and accurate time-of-flight (TOF) measurements in dense multipath environments. These are interesting features for ranging applications since these pulses can be extremely narrow in time (wide in frequency). Notice that the ranging capability usually relies on precise synchronization and time measurement [4]. There is a synergy between synchronization and localization functionalities [5], which has motivated the wide range of UWB applications in localization [6,7].

We can find commercial UWB transceivers, such as DWM1000 [8] from Decawave, compliant with the IEEE 802.15.4 UWB physical layer, which can be connected to a generic MicroController Unit (MCU), fulfilling the mentioned requirements of low cost and low power consumption. In particular, this UWB transceiver uses the Start of Frame

* Corresponding author.
E-mail addresses: juan.j.perez@uv.es (J.J. Pérez-Solano), felici@uv.es (S. Felici-Castell), antonio.soriano-asensi@uv.es (A. Soriano-Asensi), jsegura@uv.es (J. Segura-Garcia).

https://doi.org/10.1016/j.comcom.2022.01.012
Received 8 October 2021; Received in revised form 22 December 2021; Accepted 19 January 2022
Available online 25 January 2022

Delimiter (SFD) field, included in the IEEE 802.15.4 UWB packet format, to capture time marks (timestamps) from the transceiver clock just at the instant when the SFD field is detected during the transmission or reception of a packet. The use of the SFD mechanism to acquire the timestamps eliminates many non-deterministic delays introduced by the Media Access Control (MAC) layer. It must be noticed that in the wording of this paper, the terms packet and frame are interchangeable.

The implementation of synchronization protocols in WSN normally relies on the exchange of packets with timestamps that allows estimating the offset between the motes clocks. Although the use of the SFD mechanism to timestamp the transmission and reception of the packets facilitates the estimation of accurate clock offsets, these estimates become out of date after some time due to clock drifts. The low-cost clocks of WSN motes often exhibit deviations from their nominal frequency producing different clock skews between them. Thus, the network motes must perform periodic packet exchanges to maintain the synchronization accuracy within certain limits. For this purpose there are two different approaches, instantaneous synchronization (with packet exchanges in a short time and valid only for a while) and long-term synchronization, which is more efficient in terms of energy. The latter requires of the former. So, long-term synchronization is achieved applying statistical processing techniques to the previous timestamps, which improves the accuracy and reduces the number of packets exchanged. A common statistical technique applied in this context is the Linear Regression (LR) that uses a window of previous timestamps to estimate future time offsets [9–11].

In this paper we analyze from a practical and experimental point of view, the different elements involved in the time synchronization using UWB communications. The precise localization of the nodes has a negligible impact on synchronization as long as they remain static during the whole experiment. Thus, by using packet exchanges and the timestamps registered both at the UWB transceiver and the MCU, we show and characterize the sources of error that degrade the time synchronization. Finally, with a simple statistical processing based on LR it is proven that a nanosecond time scale resolution in the MCU, and even sub-nanosecond in the UWB transceiver, can be achieved with the solution proposed in this work.

The rest of the paper is structured as follows. In Section 2 is introduced the related work. In Section 3, the effects that degrade the synchronization as well as several approaches and statistical techniques used in time synchronization are detailed. In Section 4 is described the synchronization strategy proposed and how to embed a UWB transceiver and a generic MCU in a wireless node. In Section 5, this synchronization strategy is evaluated, comparing its performance with the related work. Finally, in Section 6, the main conclusions and the results of the paper are summarized.

2. Related work

Synchronization in WSN has attracted lot of research. Based on instantaneous synchronization, Timing-sync Protocol for Sensor Networks (TPSN) [12] is a pair-wise synchronization algorithm in which two packets are exchanged to capture four timestamps at the MAC layer. This approach is also known as sender–receiver scheme. Using these timestamps, the algorithm estimates the offset between two motes. The average error achieved in the synchronization is 16.9 μs using Mica motes. Reference Broadcast Synchronization (RBS) [13] shows an alternative method that starts the synchronization of multiple motes with the transmission of one broadcast packet. Then all the motes have to exchange their own timestamps to estimate their relative offsets. This process is run every time that synchronization is required. The average accuracy of RBS with Mica motes is 11 μs. In this case, this approach is known as receiver–receiver scheme.

Nevertheless, due to the drifts in the clocks of the motes, the offset estimated in one instant becomes out of date soon. Thus to achieve a long-term time synchronization, it is necessary to estimate the skew of the clocks. Flooding Time-Synch Protocol (FTSP) [10] is a network-wide synchronization protocol that uses time stamping at the MAC layer with broadcast packets, using LR over the timestamps exchanged periodically every 30 s. In a single hop network with Mica2 motes (with an internal clock of 7.4 MHz), the average accuracy is of 1.48 μs and in a multihop network with 60 motes (and 6 hops) of 3 μs. Using Rate Adaptive Time Synchronization (RATS) in [9] is shown that the error committed using LR can be minimized with different time windows (in the range of few minutes). The optimum time window for the LR is a trade off, where having few points restricts the model accuracy while having too many might cause the offset model no longer being linear. Using RATS, two Mica2 motes can achieve a synchronization error of 1.6 μs with an optimal time window of 8 min. An improved version of RATS can be found in [14], where the authors define two algorithms to search an optimum window size to perform the LR, getting a mean accuracy of 8.95 μs with a Synchronization Protocol (SP) of 1 min between two TelosB motes, as well as 12.42 μs with a maximum SP of 8 min at 4 hops away.

However, the usage of UWB communications and their interesting features as mentioned in Section 1, leads to a substantial improvement of time synchronization in WSN. For instance, the DWM1000 UWB transceiver [8] has been used in several research works in recent years. In [15], the authors extract time synchronization from Decawave DWM1001 Real-Time Location System (DRTLS) kit (based on the DWM1000 UWB transceiver), by using time stamping of the frames with the activation of SFD pin. This DRTLS system uses Symmetrical Double-Sided Two-Way Ranging. The authors evaluate both receiver–receiver and sender–receiver schemes with a maximum jitter of 3.3 μs and a standard deviation of 0.7 μs without any additional hardware. These timestamps are captured at the MCU (not at the transceiver). Notice that in [15], the UWB transceiver and the MCU are not synchronized and suffer from I/O discretization. In [16], time resolution is enhanced by means of a external 38.4 MHz clock reference common to both the MCU and the UWB transceiver. To improve the stability, it is designed a Phase-Locked Loop (PLL) circuit based on a chip-scale atomic clock, called Pulsar. This circuit jointly with an specific protocol also described in [16] permits to achieve clock synchronization better than 5 ns between indoor or GPS-denied devices, with an average of 2.12 ns (standard deviation of 0.84 ns). Despite the very interesting results provided in [16], it must be stressed out that it used an expensive clock subsystem based on an chip-scale atomic clock, which prevents it from being a low cost practical solution.

Since UWB is a recent technology applied to IEEE 802.15.4 transceivers, we can find also several interesting hardware prototypes. Focusing on the physical layer, in [17] is presented an experimental circuit design built from commercially available components and based on software defined UWB transceivers (using symbol correlation resolution) with a timing synchronization scheme, where the authors achieve a timing accuracy better than 5 ns in a network with 4 nodes. Under good Signal to Noise Ratio (SNR) conditions, their network can attain sub-nanosecond accuracy in indoor experiments. In [18] is presented a hybrid wireless node that combines an IEEE 802.15.4 transceiver for data transmission with an UWB transceiver specific to time synchronization. In [19] is shown a wireless node based on the DWM1000 UWB transceiver as well as a synchronization protocol called Wicsync using two-way clock synchronization with mutual calibration. The authors implemented the UWB synchronization specific hardware with local clock phase adjustment using a Proportional, Integral, Derivative (PID) system to tune the clock offsets. Their results show that Wicsync can support instantaneous clock synchronization accuracy with a maximum time error of 3.4 ns and an average time error of 0.141 ns within a range of 75 m. In [20] is described a low-cost real-time locating system, based on a custom UWB receiver that does not require time synchronization among sensors and uses a one-way communication scheme to reduce the cost and complexity of tags. This is done by processing at each node directly the UWB signal pulses

of 2 ns, with a carrier frequency of 7.25 GHz. In [21] is presented a wireless broadcast relative localization and clock synchronization system for multiagent environments, such as drones, where sharing real time spatio-temporal information is necessary. For this purpose, time information is distributed among the nodes to track pairwise pseudo clock parameters using Kalman filters. UWB technology is used for localization by means of the DWM1000 transceiver. From their results, the ranging accuracy corresponds to a clock synchronization accuracy of about 0.172 ns. It must be noticed that in this case, this time synchronization error is given indirectly and instantaneously by each measurement.

UWB-based localization systems based on Time Difference of Arrival (TDoA), require strict synchronization between the anchors that form the infrastructure. In [22], the authors propose a novel method of wireless synchronization utilizing two reference nodes equipped with the DWM1000 transceiver, one of them (the reference node) equipped with a Temperature Compensated Crystal Oscillator (TXCO). Using instantaneous time synchronization and compensating clock skew effects, they achieve errors lower than 1.2 ns for 90 % of the time. In [23] is proposed an improved master–slave time synchronization algorithm to enhance TDOA positioning by using a Kalman filter in order to eliminate the noise interference generated during transmission and clock skews. The authors show their simulation results using a TPSN protocol.

In addition, several protocols, which try to exploit the intrinsic features of UWB, have been proposed from a theoretical point of view. In [24] is proposed a network-wide synchronization algorithm that learns link propagation delays and channel features between neighboring nodes. This algorithm uses clustering with different tiers, defining a Timing Virtual Networks. Their approach uses UWB technology to measure round-trip TOF in order to achieve precise timing between any two nodes, and fast re-timing based on UWB pulse broadcasting. The authors conclude that given the possible accuracy of time of arrival determination in UWB networks, nanosecond accuracy timing in networks with thousands of nodes becomes feasible. In [25] is proposed a two-way time transfer protocol and algorithm to estimate clock offset, skew and drift, as well as propagation delay, assuming a quadratic clock model. The authors conclude that a time estimator based on a linear clock model may work for short time periods with limited synchronization accuracy, while the quadratic clock model they propose can be adopted to both short and long time span estimation with high synchronization accuracy.

Finally, it is worth mentioning a survey for clock synchronization over IEEE 802.11 standards in [26]. In particular, among the different solutions analyzed using IEEE 802.11 standards the highest resolution and accuracy achieved is in the order of μs. However, it must be noticed that there are other non-IEEE 802.11 approaches capable of achieving a sub-nanosecond accuracy and precision by using a timestamping at the physical layer, as shown in [27,28].

In summary, from these works it can be concluded that UWB is a promising technology and there is room for further research in order to exploit their benefits in WSN. On one hand, we have seen new hardware designs with several prototypes and on the other hand, we have seen several simulated proposals for enhanced ad-hoc synchronization protocols. But from a practical point of view for WSN deployments, the majority of these works are not so useful because they are excessively theoretical or they do not fulfill the requirements for low cost and low power consumption. Nevertheless, we highlight from these works, two outstanding contributions, such as [15,16] based on DWM1000 UWB transceiver [8]. However, in these papers, their approaches are not focused on achieving and improving time synchronization using off-the-shelf products, but tuning them with additional components. Thus, in this paper, we focus on this commercial transceiver and evaluate its performance in a simple prototyped wireless node with a generic MCU. In addition, we analyze and discuss different communication schemes and synchronization protocols, as well as we use simple statistical methods to improve time accuracy.

3. Analysis of time synchronization in WSN

Time synchronization in wireless nodes within WSN depends on several subsystems and rely on different processes. The impact of the clock and the communications subsystems as well as the time stamping techniques will be discussed in the next subsections.

3.1. Clock subsystem

As mentioned before, wireless nodes track synchronization by exchanging packets with timestamps. These timestamps are based on local readings of internal counters at each node, which are incremented using their own clock, independently of the rest of nodes. These clocks suffer from drifts or fluctuations that affect their precision, producing clock skews among the nodes in a random way. These drifts are due to different environmental conditions (such as temperature, humidity, voltage, aging, etc.) [29,30]. Typical drift values for a low-cost quartz crystal are in the range from 1 ppm to 100 ppm [31].

As a first approach, the relationship between the local time of two nodes is assumed to be linear. It can be modeled by an offset (α) and a skew (β) which is the rate of change of the clocks:

$$y(t) = \beta \cdot x(t) + \alpha \tag{1}$$

where $y(t)$ and $x(t)$ are the local clocks of the two nodes, and t is the time reference.

Since environmental conditions change gradually, β is usually modeled as a one dimension Additive White Gaussian Noise (AWGN) Random Walks (RW) [14,32]. The clock skew of the clock in node A respect to the clock in node B (β_A^B) at time $t_0 + t$, is given by:

$$\beta_A^B(t_0 + t) = \beta_A^B(t_0) + \int_{t_0}^{t} \eta(u)du \tag{2}$$

where $\eta(u) \sim N(0, \sigma_{RW}^2)$, being σ_{RW} the characteristic standard deviation of the RW. For these low cost clocks, it is assumed a σ_{RW} in the order of 10^{-8} and 10^{-9} [30,33].

3.2. Communication subsystem

These local readings from the internal counters (or timestamps) are then exchanged sending frames among the nodes. However, these frames suffer additional random delays added by the sender node, the TOF (or propagation delay) as well as the receiver node [14]. According to IEEE 802.15.4 standard, to reduce non-deterministic delays, the transceiver writes down the timestamps in a frame, synchronized with the transmission and reception of the SFD field as described before. Even so, there are still additional little random delays (or synchronization errors) due to the several issues, such as internal clock offsets, antenna excitation, internal frame management, signal multi-path, etc. This randomness is modeled as AWGN and it is added to Eq. (1) as a measurement error (n):

$$y(t) = \beta \cdot x(t) + \alpha + n \tag{3}$$

where $n \sim N(0, \sigma_C^2)$, being σ_C the standard deviation of the sum of all these accumulated errors due to the communication subsystem.

3.3. Time stamping techniques and packet exchanges for delay and offset estimation

There are different approaches to synchronize several wireless nodes by means of packet exchanges, some of them shown in Fig. 1. It must be noticed that due to the transmission and reception of these packets, maintaining synchronization among all the nodes also increases the burden on their resources and then conflicts with the low energy consumption goal of the IEEE 802.15.4 standards.

The simplest approach is the "one-way" (Fig. 1(a)). One of the nodes (in this case A) acts as master, and periodically broadcasts

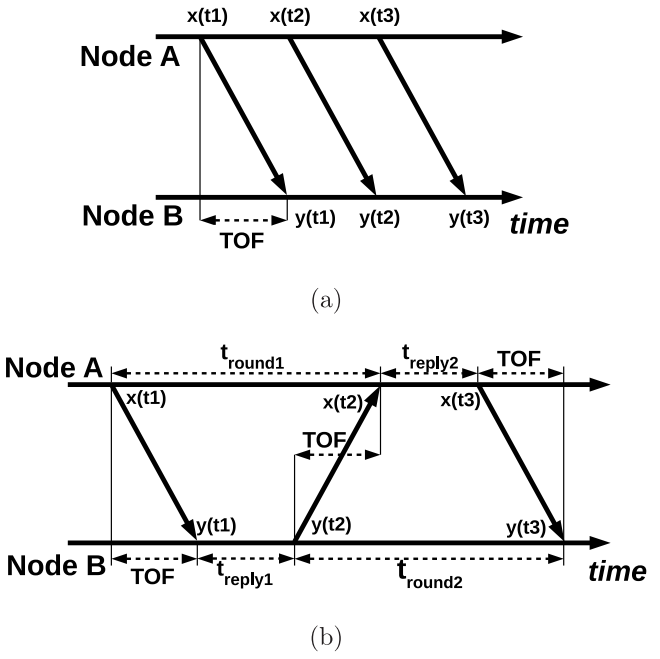

Fig. 1. Packet exchanges for synchronization between two nodes with detail of the timestamps: (a) One way, (b) Three way (or Double-Sided two way).

packets with its timestamps. The slaves (in this case node B) use these timestamps to synchronize with the master. This unidirectional model is employed by RBS [13]. In this case, the slaves can adjust their local times $(y(t_1), y(t_2), ..y(t_i))$ to the ones provided by the master $(x(t_1), x(t_2), \ldots, x(t_i))$, being i the number of the exchanged packet. With several timestamps, the β in Eq. (1) can be computed as follows:

$$\beta = \frac{y(t_i) - y(t_{i-1})}{x(t_i) - x(t_{i-1})} \tag{4}$$

And the offset (α) can be calculated after the first exchange (at $t = 0$). However, this estimation is only valid when the communication delays between the master and the slave are assumed to be constant. This approach does not allow to estimate the delay, which is solved in the "two-way" [12] and the "three-way" (Fig. 1(b)) approaches. This "three-way" approach is also known as Double-Sided two way. There are different implementations with different names, with the purpose of estimating the TOF of the UWB signal [34]. The formula shown in Eq. (5) was proposed in [35]. In particular, this formula will be used in Section 5.2 for the purpose of calibration antenna delays and it is the one recommended by the manufacturer of the UWB transceiver [36].

$$TOF = \frac{t_{round1} \cdot t_{round2} - t_{reply1} \cdot t_{reply2}}{t_{round1} + t_{round2} + t_{reply1} + t_{reply2}} \tag{5}$$

where the variables in Eq. (5) refer to the timings in Fig. 1(b).

3.4. Using Least Squares Linear Regression to improve time synchronization

As it was detailed in previous subsections, the accuracy in time synchronization is dependent on several effects which are considered in the linear clock model as AGWN in β (random walks) and also in α (random delays). Since, we know the timestamps at discrete observations i, Eq. (3) can be rewritten as:

$$y_i = \beta \cdot x_i + \alpha + n_i \tag{6}$$

where $y_i = y(t_i)$ and $x_i = x(t_i)$ are the ith timestamp observations at the transmitter and receiver, respectively, while n_i is the contribution of noise. Eq. (6) can be used to calculate $\hat{\beta}$ and $\hat{\alpha}$, which are estimations of β and α respectively, by means of a LR over a certain number of previous observations (or window size). It is worth mentioning that the method of Least Squares (LS) does not require any assumption for the distributions of the timestamps captured.

Then, with $\hat{\beta}$ and $\hat{\alpha}$ the next timestamp can be estimated as: $\hat{y}_{i+1} = \hat{\beta} \cdot x_{i+1} + \hat{\alpha}$. The absolute prediction error $|e_{i+1}| = |y_{i+1} - \hat{y}_{i+1}|$ is given by the difference between the estimated and measured (y_{i+1}) timestamps. Both y_{i+1} and $\hat{y}_{i+1}$ are observable at the receiver at the time instant t_{i+1}. The average value of these prediction errors, or Mean Absolute Prediction Error (MAPE), has been considered as a measure of the quality of the estimation, in order to easily highlight the time synchronization error

The accuracy of the timestamp estimation strongly depends on the LR and the window size used. Since two different and independent underlying mechanisms influence these regressions, different behaviors are appreciated due to both clock skew (β) and n or noise, based on the values of σ_{RW} and σ_C defined in Sections 3.1 and 3.2 respectively. On one hand, when σ_C is higher than σ_{RW} then larger window sizes are preferred. On the other hand, if σ_{RW} is higher than σ_C, then shorter window sizes are preferred. In this scenario, there is an optimum window size that will minimize the error for the time estimation [37].

4. UWB based time synchronization proposal

In the following sections, we evaluate the feasibility of establishing a common time reference within the nodes of a WSN with an accuracy at the nanosecond scale using off-the-shelf commercial components and commodity hardware, analyzing which factors have the highest impact on timing accuracy. UWB transceivers, as the DWM1000 by Decawave, have demonstrated excellent time synchronization capabilities being able to achieve time accuracies at the picosecond time scale [16]. The proposal evaluated in this manuscript is to add a low cost UWB transceiver to the MCU of each node in the WSN in order to improve timing capabilities. The interconnection between the MCU and the UWB, as well as their description, is also explained in detail.

The synchronization algorithm implemented in the nodes was based on the FTSP. We avoid using more complex protocols in this paper, since we want to highlight in a simple way the performance of UWB technology using DWM1000 transceiver in time synchronization instead of ranging applications, for a different scope which the UWB transceiver was designed for. Thus, we focus on the characterization of the transceiver itself, instead of considering advanced and complex synchronization protocols.

As it is illustrated in Fig. 2, the master node broadcasts a message every SP, containing the transmission timestamp (t_{txUWB}). Each node timestamps the reception of messages in the UWB transceiver (t_{rxUWB}) and the activation of the SFD is also timestamped by the MCU (t_{rxMCU}). Considering the last timestamps received, each node updates the LS clock model and translates the local timestamps t_{rxMCU} to the t_{txUWB} of the master, which is considered as the reference clock for the whole network. Notice that multiple receivers (slave nodes) at the same time, located in the same radio range of the master, could synchronize their clocks using this packet without any impact in the process.

Thus, the main goal of this work is to analyze what elements have the strongest impact in the relationship between the global reference t_{txUWB} and the t_{rxMCU} on each node. This relationship involves the internal clock of the MCU, as it is the one used to timestamp all the events registered in a node. However, the t_{rxUWB} was also considered in order to evaluate the contribution of the UWB transceiver clock to the synchronization error between the master and the MCU.

4.1. Embedding the DWM1000 UWB transceiver with a generic MCU

In order to meet the low cost requirements as we discussed previously, we focus on the Decawave DWM1000 UWB transceiver [8] that can be connected to a generic MCU, as shown in Fig. 3. Fig. 3(a)

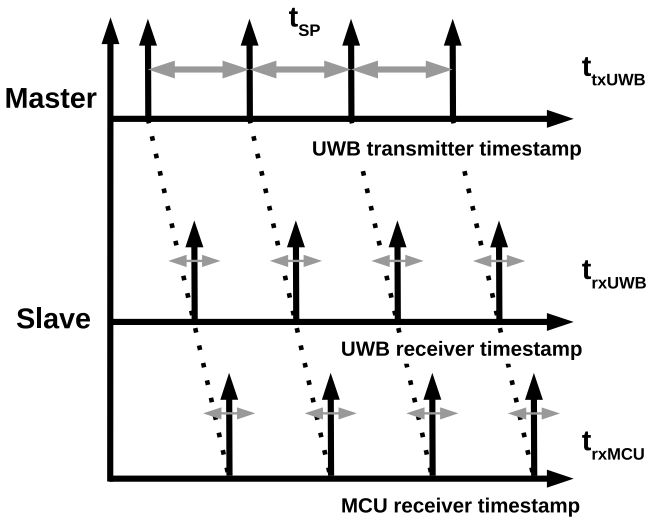

Fig. 2. Time line of the beacons sent from UWB transceiver of the master and the timestamps registered at the UWB transceivers and the MCU of each slave. In this figure, t_{SP} is the synchronization period and t_{txUWB}, t_{rxUWB} and t_{rxMCU} are the timestamps captured at the UWB transceivers of the master and the slave, and at the MCU of the slave, respectively.

shows a wired version of this prototype and Fig. 3(b) shows its electronic scheme. The MCU used in the experiments is the Arduino MKRZero [38]. It is based on the ATSAMD21G18 Cortex-M0 32 bit low power ARM running at 48 MHz with 32 KB RAM and 256 KB Flash. We chose this MCU due to its power consumption, cost and capability for precise timestamping with the hardware capture module associated to the internal timers. We have employed the arduino-DWM1000 library which offers basic functionalities to use the Decawave's DWM1000 chips/modules [39]. Among all the MCUs compatible with this library, the Arduino MKRZero is the one that works at a relatively high frequency of 48 MHz, and it can be adjusted using their internal PLL, to even double this frequency till 96 MHz.

The Decawave DWM1000 UWB transceiver is a single low-power and low-cost integrated circuit, compliant to IEEE 802.15.4-2011, that supports data rates as high as 6.8 Mbps. According to its data sheet [8], this transceiver can be used to locate tags with a precision of a few cm. It has a 40-bit counter that increments with an internal 64 GHz clock, which achieves a nominal 15.6 ps timestamp precision. This counter is used for frame time stamping, both on the transmission of the SFD field of a frame and also on the detection of the SFD field of an incoming frame. Using this SFD mechanism, the sending transceiver can take the time mark as late as possible, and the receiving transceiver can take its time mark as soon as possible. In addition, the DWM1000 provides a SFD output pin (called *SFDled*) activated only on packet reception, that can be used to indicate the MCU the arrival of the SFD field.

The scheme with the interconnection of the MCU and the DWM1000 UWB is shown in Fig. 3(b). The Serial Peripheral Interface (SPI) is employed to control the UWB transceiver from the MCU, and for data transfer. Notice that both the transceiver and the MCU have their own local times based on independent clocks. In order to timestamp the reception of a packet with the clock of the MCU, the *SFDled* pin of the UWB transceiver was connected to an input pin of the MCU. A program interruption was configured to timestamp the activation of the *SFDled* pin, capturing the value of an internal timer of the MCU with its hardware capture module.

5. Performance analysis

The testbed used to evaluate the synchronization strategy proposed is based on a transmitter (master) and several receivers (slaves) one hop away in an indoor scenario. All these nodes have the same design as shown in Fig. 3. Several experiments have been performed in this section in order to analyze all the aspects involved in the synchronization. First, the time variation in the activation of the SFD is measured in Section 5.1. The delays introduced by the communication subsystem and the TOF are evaluated in Section 5.2. Finally, the contribution of UWB transceivers and the MCU to the synchronization errors are analyzed in Sections 5.3 and 5.4.

Table 1
Time difference for the *SFDled* pin activation at two UWB transceivers (side-by-side) when receiving the same beacon, one meter away from the master.

	MCU (clock ticks)	Time (ns)
Avg. time difference	0.18	2.3
Std. time difference	1.76	22.4

5.1. Characterization of the SFDled pin of the DWM1000 transceiver

One of the factors that affects the accuracy for the synchronization among MCUs is the SFD timestamping and the activation of the *SFDled* pin. Its experimental characterization is required since it is not specified in the datasheet of the DWM1000. In this subsection we present the evaluation of time variations in the activation of the *SFDled* pin at two UWB transceivers placed together (side-by-side), one meter away from the master. As the distance from the master to both transceiver slaves is equal, it is expected that they both receive the beacon at the same time instant. Therefore, the time differences measured are due to the internal random delays introduced in the circuitry of the DWM1000.

As the expected differences to be measured in this experiment are comparable to the period of the clock measured for the measurements we first captured the waveform of the activation of the *SFDled* pin with a DSO1022 A Oscilloscope by Keysight Technologies [40]. Fig. 4 depicts the waveforms of the *SFDled* signals of both slave transceivers that were registered with the oscilloscope. They are introduced as examples to characterize the behavior of this SFD mechanism. The waveforms shown in Fig. 4 are representative of measurements in which both activations occur at the same time (Fig. 4(a)) and those with a time lag between activations (Fig. 4(b)).

In order to characterize these random time differences over a larger period of time, and obtain an statistical estimation of the time variation in the activation of the *SFDled* pin, an additional experiment was performed in which the *SFDled* pins of two UWB slave transceivers were connected to the same MCU. The master was configured to send beacons every 200 ms for 1 h and a half. The UWB slave transceivers activated the *SFDled* pin on their reception, and the difference of the timestamps captured at two different timers included in the same MCU was used to measure the variation in the activation of the *SFDled* pin. Fig. 5 shows the histogram with the time differences measured. The measurements in Fig. 5 are shown in clock ticks, showing that most time differences are within one clock tick. The counters were configured to work at 78.67 MHz clock frequency in order to prevent from losing timestamps due to the limitations of the MCU's hardware capture module. Thus, each clock cycle (clock tick) corresponds to 12.7 ns. Notice that this clock frequency was only applied to the timer and the capture module, but the clock for the rest of MCU modules, including the clock for the program execution, continued being equal to 48 MHz, since this is the maximum frequency accepted by the MCU. In Table 1 are shown the average of the activation time difference and its standard deviation, which resulted in 2.3 ns and 22.4 ns respectively.

5.2. Analysis of the Time of Flight between UWB transceivers

Another effect that influences the synchronization accuracy among the wireless nodes is the delay, which is composed of the TOF, i.e. the time the packet requires to travel from one node to another, plus a

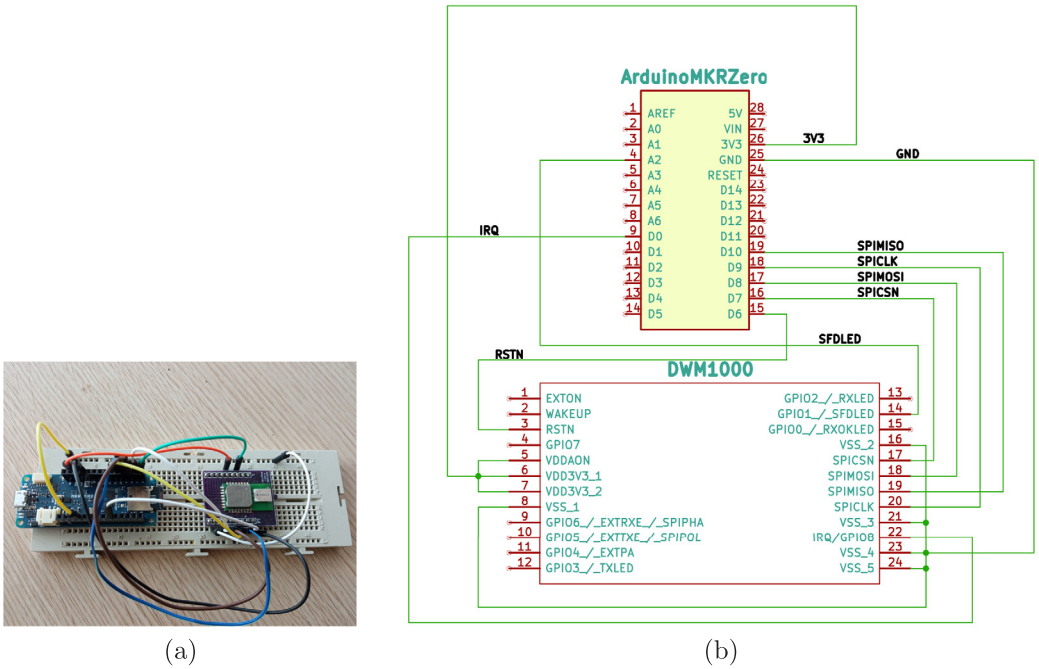

Fig. 3. Detail of the DWM1000 UWB transceiver connected to a MCU Arduino MKRZero: (a) wired prototype of DWM1000 UWB Transceiver and MCU and (b) its scheme.

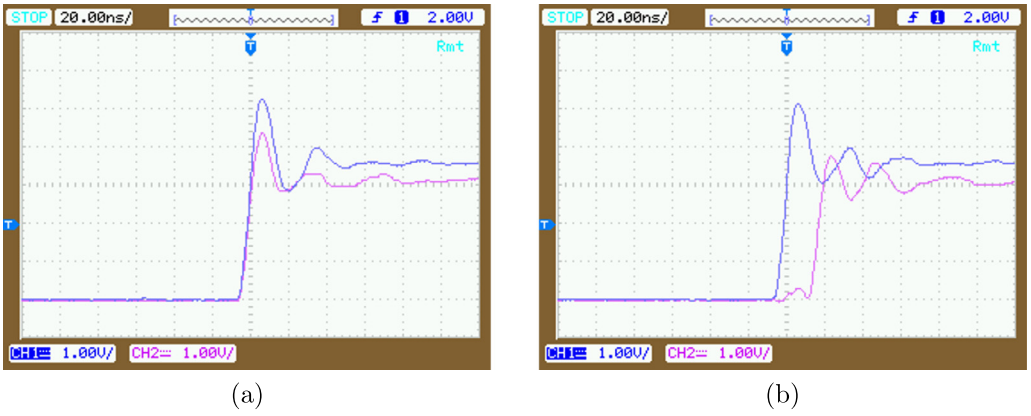

Fig. 4. Screenshots captured using an oscilloscope of the activation waveforms of the *SFDled* pins of two UWB transceivers on the reception of a beacon. Figures show the voltage of the *SFDled* pins with a time and voltage scale of 20 ns and 1 V per division respectively. (a) Example of a simultaneous activation of both *SFDled* pins. (b) Example of a measurement in which the *SFDled* pins are not activated simultaneously.

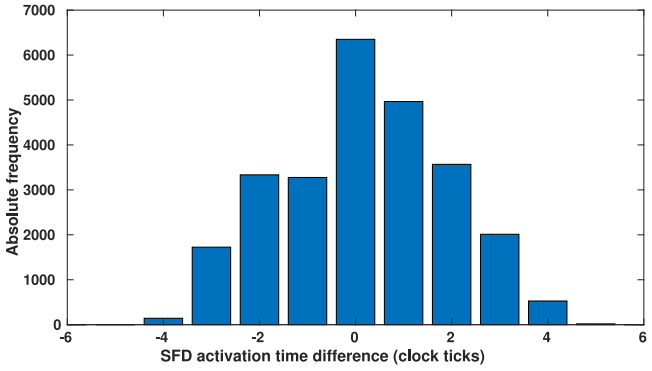

Fig. 5. Histogram of the time difference (in clock ticks of the MCU at 78.67 MHz) for the activation of the *SFDled* pins at two different UWB transceivers when receiving the same beacon.

constant delay caused by transmission and reception antennas. As explained in Section 3.3, the delay might be estimated by considering the timestamps exchanged by the UWB transceivers. In order to minimize the impact of clock offsets and drifts of the UWB transceivers, the *Three-way (asymmetric formula)*, also known as alternative double-sided two-way [35], shown in Eq. (5) was considered for the estimation of the delay in this experiment. Two different scenarios were considered for a better characterization and calibration of the TOF. On the one hand, we made the measurements in an anechoic chamber in order to avoid undesired reflections due to multipath. On the other hand, we repeated the same tests, but outdoor in order to generalize the results.

In the first experiment conducted inside the anechoic chamber, the master and slave were placed at several distances from 1 m to 5 m, with steps of 1 m. At each position, the master and the slave were configured to exchange their timestamps every 200 ms for 30 min. The temperature ranged between 23-24 °C during the test. The average and standard deviation of delay measurements at each distance are presented in Table 2. A linear regression of the results in Table 2 will allow the estimation of the TOF at each distance. The intercept of 2.505 ns represents the constant delay introduced by the transmission

Table 2
Time delay (in ns) between two UWB transceivers inside an anechoic chamber at 1, 2, 3, 4 and 5 m.

Distance [m]	Avg. [ns]	Std. [ns]
1	5.70	0.10
2	9.41	0.07
3	12.83	0.09
4	16.10	0.06
5	19.29	0.07

Table 3
Time delay (in ns) and Packet Loss Ratio (PLR) between two UWB transceivers in an outdoor setting at 5, 10, 15, 20, 25, 30, 35, 40 and 45 m.

Distance [m]	Avg. [ns]	Std. [ns]	PLR
5	19.10	0.09	0
10	36.77	0.12	0
15	52.71	0.08	0
20	71.04	0.09	0
25	86.18	0.17	0.0018
30	105.31	0.10	0.0075
35	119.65	0.15	0.0128
40	139.58	0.38	0.0387
45	153.12	0.99	0.7009

and reception antennas, while the slope of 3.387 ns/m is an estimation of the rise of TOF with distance. The inverse of the slope is $2.95{\cdot}10^8$ m/s, is close to the speed of light in air (299792458 m/s). The discrepancies between the estimated and true speed of light might be attributed to small inaccuracies in the positioning of the nodes during these measurements.

The same experiment was repeated in an outdoor scenario at the premises of our School of Engineering, measuring the TOF between 5 and 45 m. The average TOF and the standard deviation measured at each distance is shown in Table 3. The standard deviation is observed to be around 0.1 ns except when distances larger than 35 m were considered. This is mainly caused by the rapid increase of packet losses, specially at 45 m where 70 % of packets were lost. The same linear fit has been conducted in order to determine the delay introduced by the transmission and reception antennas. But only the measurements up to 35 m were considered because of increase of packet losses at higher distances. The linear fit resulted in a slope of 3.375 ns/m and an intercept of 2.603 ns, which are in good agreement with the results achieved inside the anechoic chamber.

5.3. Clock synchronization between UWB transceivers

As mentioned in Section 4, the main goal of this work is to establish a time relationship between each node's MCU and the master clock used as time reference, t_{rxMCU} and t_{txUWB} in Fig. 2, respectively. However, the analysis of the time synchronization accuracy between two UWB transceivers is also of interest in this work in order to determine the impact of UWB transceivers on the synchronization the nodes in the WSN.

These experiments were made with two nodes, master and slave, placed 1 m apart. All the transceivers remained active and in reception mode during the whole experiment so that all the nodes could always communicate. The master sent a beacon every SP with the t_{txUWB} timestamp, while t_{rxUBW} and t_{rxMCU} were registered on its reception at the slave node. The clock model used to estimate t_{txUWB} from t_{rxUWB} was updated on the slave node by means of a LS fit considering the most recent timestamps. Each measurement lasted 5 h, and the temperature ranged between 26 °C and 28 °C during the whole experiment. The calculation of the LS model can be easily implemented in the MCU. However, for the sake of easing the reproducibility of the results and their analysis, the storage of all three timestamps for the off-line evaluation of the LS model was preferred.

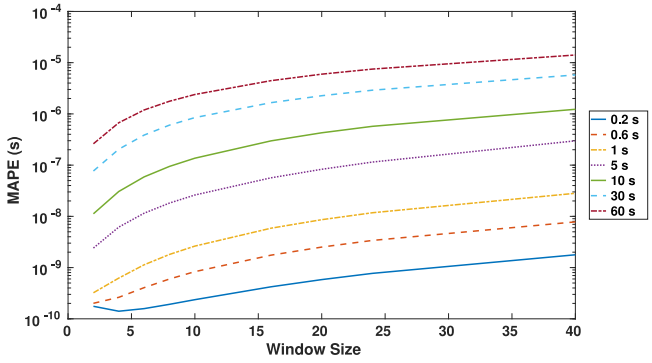

Fig. 6. Variation of MAPE between two UWB transceivers with the number of previous timestamps considered in the LS model, for several SP between 200 ms and 60 s.

Table 4
Comparison of the MAPE values measured for each SP interval when considering the timestamps at the UWB transceiver and the timestamps at the MCU.

SP	200 ms	600 ms	1 s	5 s	10 s	30 s	60 s
UWB	0.14 ns	0.19 ns	0.32 ns	2.6 ns	11 ns	76 ns	261 ns
MCU	31 ns	33 ns	36 ns	45 ns	66 ns	177 ns	452 ns

The timing accuracy between two UWB transceivers is evaluated in terms of MAPE, which is the difference between predicted and actual values of t_{txUWB} averaged over the whole experiment, as mentioned in Section 3.4. MAPE depends on how often the master broadcasts the synchronization beacons, which is controlled by SP, and also on the number of timestamps considered in the LS, i.e. the window size.

In Fig. 6 is shown how MAPE is influenced by the SP and the window size when considering the synchronization of two UWB transceivers. The minimum SP was set to 200 ms as a trade off, to avoid too excessive packets sent by the master and short enough to achieve an accurate time synchronization. As expected, the larger the SP the higher MAPE values were obtained. In spite of having used window sizes as large as 40, the minimum MAPE observed in Fig. 6 was measured when considering the last two timestamps. A window size larger than 2 was only justified when a SP of 200 ms was used. According to the discussion in Section 3.4, this suggests that the noise has a significant contribution to the error only when SP intervals smaller than 1 s were employed. For SP intervals larger than 1 s, the MAPE values increased with the window size. Thus, RW prevails over random noise on synchronization errors, as discussed in Section 3.4. MAPE values smaller than 1 ns were measured when considering SP intervals shorter than 1 s when considering the optimum window size, as shown in Table 4.

Based on these results, we could consider an scenario with interferences or collisions that would impact on the SP as an added random delay. But notice that these random fluctuations due to these issues are at a much lower time scale than the SP and are negligible. Nevertheless, this effect could be seen if we consider a plot with a higher SP in Fig. 6, which would raise slightly the MAPE according to these results.

5.4. Clock synchronization between the MCU of a node and the global time reference

Finally, the error in the estimation of the global time reference from the timestamps recorded at the MCU of a given node is analyzed. The measurements and the estimation of the global time reference t_{txUWB} were described in Section 5.3, but the timestamps at the MCU (t_{rxMCU}) are considered in this case, instead of t_{rxUWB}. As in Section 5.1, the internal clock used to increase the timer values was configured with a frequency of 78.67 MHz.

The variation of the MAPE with SP and window size when considering the timestamps at the MCU, shown in Fig. 7, is similar to the one

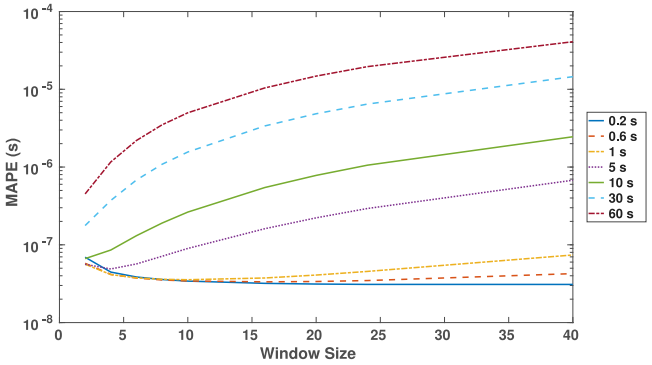

Fig. 7. Variation of MAPE between the node's MCU and the global time reference with the number of previous timestamps considered in the LS model, for several SP between 200 ms and 60 s.

observed between UWB transceivers (Fig. 6), except for the fact that larger errors were obtained when considering the MCU timestamps. In Table 4 are compared the MAPE values measured when considering the timestamps at the UWB transceiver and the timestamps at the MCU. The MAPE values measured with the MCU timestamps are significantly larger than those between UWB transceivers. This suggests that the major contribution to the error occurs locally in each node, between the MCU and the UWB transceiver on-board. In Table 4 is also appreciated that MAPE values decrease with the SP interval when considering the timestamps at the UWB transceiver. However, little improvement in MAPE is observed for SP intervals smaller than 1 s when considering the timestamps at the MCU. This is due to the variability in the activation of the *SFDled* pin of the UWB transceiver, which has a standard deviation of 22 ns, as shown in Section 5.1.

5.5. Comparison of the proposed system with related work

In order to highlight our contribution, Table 5 shows the comparison of our proposal with the related work, taking into account those references that follows the same criteria than the ones defined, by using a simple, feasible and low-cost approach. The criteria used to select these references was based on a thorough survey through the main important scientific databases (IEEE, ACM and Elsevier) in advanced searches, focusing on time synchronization in wireless networks with real experiments and deployments, meeting similar requirements, low cost and low power consumption. Since the number of nodes (motes) and distances could be an issue in this comparison, next we include this detail for the different references included in Table 5. In the references based on wireless technologies such as Zigbee based or WIFI (from row 2 to 7), the distance among the nodes is not specified and the number of these nodes is only two or three. In contrast, references which integrate UWB transceivers, include these distances, that are between 3 to 7 m. In particular, in [15], the test-bed is made up of 5 nodes separated 3 m. In [22], they use 6 nodes in a room of 6x6 m^2 with 3.87 m, as the largest distance. Finally, in BLAS [21], they use 3+1 nodes in a room of 2.5x4 m^2, and 4+2 in an area of 5x7 m^2 outdoor.

Then, as we can see, this is the first work that achieves long-term nanosecond accuracy time synchronization using off-the-shelf components in the MCU and sub-nanosecond in the UWB transceivers.

6. Conclusions

In this paper, the integration of an UWB transceiver within a traditional WSN is proposed, in order to achieve a synchronization accuracy in the order of nanosecond time scale. Moreover, the results presented in this work have proved that such a timing accuracy can be achieved using off-the-shelf commercial devices and commodity hardware, keeping the low cost feature of WSN nodes. The solution here proposed is the interconnection of the DWM1000 UWB transceiver by Decawave to an Arduino MKRZero MCU, and a synchronization procedure based on a simple synchronization protocol, such as FSTP for the synchronization of the nodes. The majority of papers related with time synchronization using UWB technology showed the timing accuracy between the UWB transceivers. However, this work is focused on the timing accuracy achieved between the reference clock of the WSN and the clock of the MCU. The results of this study are of particular interest because the MCU clock is the time reference for all events related with sensors and actuators, other than the UWB transceiver. This good timing accuracy might have a strong impact in applications like ranging and collaborative beamforming, as well as advanced features for improved duty cycling to extend the network life time and in contention based channel access, to improve network utilization.

The contribution of the elements involved in time synchronization has been systematically analyzed from a practical and experimental point of view. The analysis of the activation of the *SFDled* pin, which was used to indicate the reception of a packet in the UWB transceiver to the MCU, has shown that the inaccuracy of its activation has a standard deviation of 22.4 ns. This value limits the timing precision that can be achieved in the MCU. While the smallest error measured between UWB transceivers was 0.14 ns, a MAPE value of 31 ns was measured with the MCU for the same configuration.

Besides, the measurement of the delay between two transceivers at several distances concluded that the delay introduced by the internal circuitry of the UWB transmitter and receiver (including the antennas) is 2.505 ns in total, and the TOF increases with the separation of the nodes at a rate of 3.387 ns/m, which is in good agreement with the expected value of the inverse of speed of light in air. These results suggest that the "one-way" synchronization strategy considered is valid as long as the nodes are close. For distances larger than 5 m, the delay is not negligible compared to the error. In these cases, it would be therefore advisable the use of a synchronization procedure based on two-way or three-way measurement, that allows to compensate for the delay between transmitter and receiver.

The analysis of the accuracy of time synchronization was made in terms of the MAPE, which was evaluated for several SP and several window sizes of the LS model. As expected, the MAPE rises with SP, the MAPE values measured when considering the MCU timestamps were larger than those measured between UWB transceivers. From the MAPE values measured, the use of an LS model was only justified when a timing accuracy smaller than 100 ns was required, and SP intervals smaller than 1 s were employed. When SP values larger than 10 s were considered, the best MAPE results were obtained with a linear interpolation considering the last two timestamps received. The same behavior was observed when considering the timestamps at the MCU and with the timestamps registered at the UWB transceivers.

CRediT authorship contribution statement

Juan J. Pérez-Solano: Conceptualization, Methodology, Software. **Santiago Felici-Castell:** Methodology, Data curation, Writing – original draft, Writing – review & editing. **Antonio Soriano-Asensi:** Data curation, Writing – original draft. **Jaume Segura-Garcia:** Writing – review.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Grant PID2020-113785RB-100 funded by MCIN/AEI/10.13039/501100011033. This work is supported also by the Ministry of Science and Innovation, Spain under the project RTI2018-098156-B-C55, by the Generalitat Valenciana, Spain under the projects GV/2020/052, GV/2020/046, AICO/2020/154, AEST/2021/016 and BEST/2021/150, finally by University of Valencia (Spain) with the project UV-INV-AE-1544281.

Table 5
Comparison of time synchronization protocols based on wireless networks with different IEEE technologies and off-the-shelves platforms.

Reference	Error	IEEE Technology (platform/transceiver)	Comments
TinySync [11]	10.78 μs	802.15.4 QPSK in 2.4 GHz (TelosB/CC2420)	100 s of SP, long-term
TinySync [11]	1.3 μs	pre-802.15.4 FSK in 2.4 GHz (Mica2/CC1000)	20 μs of SP, long-term
TPSN [12]	16.9 μs	pre-802.15.4 ASK in 2.4 GHz (Mica/TR1000)	instantaneous
RBS [13]	11 μs	pre-802.15.4 ASK in 2.4 GHz (Mica/TR1000)	instantaneous
FTSP [10]	1.48 μs	pre-802.15.4 FSK in 2.4 GHz (Mica2/CC1000)	30 s of SP, long-term
RATS [9]	1.6 μs	pre-802.15.4 FSK in 2.4 GHz (Mica2/CC1000)	1 min of SP, long-term
[28]	2.5 μs	802.11, FSK 2.4 GHz (Atheros 5414)	1 min of SP, long-term
[15]	3.3 μs	802.15.4-2011 DS-UWB in 3.5–6.5 GHz (DWM1000)	100 ms of SP, long-term
[22]	1.2 ns	802.15.4-2011 DS-UWB in 3.5–6.5 GHz (DWM1000	instantaneous
BLAS [21]	0.172 ns	802.15.4-2011 DS-UWB in 3.5–6.5 GHz (DWM1000	instantaneous
Our proposal	0.14 ns	802.15.4-2011 DS-UWB in 3.5–6.5 GHz (DWM1000	200 ms-1 min of SP, long-term

References

[1] J.J. Pérez-Solano, S. Ezpeleta, J.M. Claver, Indoor localization using time difference of arrival with UWB signals and unsynchronized devices, Ad Hoc Netw. 99 (2020) 102067, http://dx.doi.org/10.1016/j.adhoc.2019.102067, URL http://www.sciencedirect.com/science/article/pii/S157087051930215X.

[2] E. Navarro-Camba, S. Felici-Castell, J. Segura-García, M. García-Pineda, J. Pérez-Solano, Feasibility of a stochastic collaborative beamforming for long range communications in wireless sensor networks, Electronics 7 (12) (2018) 417, http://dx.doi.org/10.3390/electronics7120417.

[3] D. Djenouri, M. Bagaa, Synchronization protocols and implementation issues in wireless sensor networks: A review, IEEE Syst. J. 10 (2) (2016) 617–627, http://dx.doi.org/10.1109/JSYST.2014.2360460.

[4] Y. Shen, M.Z. Win, Fundamental limits of wideband localization— part I: A general framework, IEEE Trans. Inf. Theory 56 (10) (2010) 4956–4980.

[5] B. Denis, J.. Pierrot, C. Abou-Rjeily, Joint distributed synchronization and positioning in UWB Ad hoc networks using TOA, IEEE Trans. Microw. Theory Tech. 54 (4) (2006) 1896–1911.

[6] N.M. Senevirathna, O. De Silva, G.K.I. Mann, R.G. Gosine, Kalman filter based range estimation and clock synchronization for ultra wide band networks, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2020, pp. 9027–9032, http://dx.doi.org/10.1109/IROS45743.2020.9340971.

[7] M. Hamer, R. D'Andrea, Self-calibrating ultra-wideband network supporting multi-robot localization, IEEE Access 6 (2018) 22292–22304, http://dx.doi.org/10.1109/ACCESS.2018.2829020.

[8] Decawave, DWM1001 System overview and performance, 2015, https://www.Decawave.com/content/dwm1001-system-overview-andperformance. (Accessed 26 May 2021).

[9] S. Ganeriwal, I. Tsigkogiannis, H. Shim, V. Tsiatsis, M. Srivastava, D. Ganesan, Estimating clock uncertainty for efficient duty-cycling in sensor networks, Netw. IEEE/ACM Trans. 17 (3) (2009) 843–856.

[10] M. Maróti, B. Kusy, G. Simon, A. Lédeczi, The flooding time synchronization protocol, in: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, in: SenSys, vol. 04, ACM, New York, NY, USA, 2004, pp. 39–49, http://dx.doi.org/10.1145/1031495.1031501, URL http://doi.acm.org/10.1145/1031495.1031501.

[11] S. Yoon, C. Veerarittiphan, M.L. Sichitiu, Tiny-sync: Tight time synchronization for wireless sensor networks, ACM Trans. Sen. Netw. 3 (2) (2007) http://dx.doi.org/10.1145/1240226.1240228, URL http://doi.acm.org/10.1145/1240226.1240228.

[12] S. Ganeriwal, R. Kumar, M.B. Srivastava, Timing-sync protocol for sensor networks, in: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, in: SenSys, vol. 03, ACM, New York, NY, USA, 2003, pp. 138–149, http://dx.doi.org/10.1145/958491.958508, URL http://doi.acm.org/10.1145/958491.958508.

[13] J. Elson, L. Girod, D. Estrin, Fine-grained network time synchronization using reference broadcasts, SIGOPS Oper. Syst. Rev. 36 (SI) (2002) 147–163, http://dx.doi.org/10.1145/844128.844143, URL http://doi.acm.org/10.1145/844128.844143.

[14] J.J. Pérez-Solano, S. Felici-Castell, Adaptive time window linear regression algorithm for accurate time synchronization in wireless sensor networks, Ad Hoc Netw. 24 (2015) 92–108, http://dx.doi.org/10.1016/j.adhoc.2014.08.002.

[15] F. Bonafini, P. Ferrari, A. Flammini, S. Rinaldi, E. Sisinni, Exploiting time synchronization as side effect in UWB real-time localization devices, in: 2018 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control, and Communication, ISPCS, 2018, pp. 1–6.

[16] A. Dongare, P. Lazik, N. Rajagopal, A. Rowe, Pulsar: A wireless propagation-aware clock synchronization platform, in: 2017 IEEE Real-Time and Embedded Technology and Applications Symposium, RTAS, 2017, pp. 283–292.

[17] M. Segura, S. Niranjayan, H. Hashemi, A.F. Molisch, Experimental demonstration of nanosecond-accuracy wireless network synchronization, in: 2015 IEEE International Conference on Communications, ICC, 2015, pp. 6205–6210.

[18] C.M. De Dominicis, A. Flammini, S. Rinaldi, E. Sisinni, A. Cazzorla, A. Moschitta, P. Carbone, High-precision UWB-based timestamping, in: 2011 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication, 2011, pp. 50–55.

[19] B. Xue, Z. Li, P. Lei, Y. Wang, X. Zou, Wicsync: a wireless multi-node clock synchronization solution based on optimized UWB two-way clock synchronization protocol, Measurement (2021) 109760, http://dx.doi.org/10.1016/j.measurement.2021.109760, URL https://www.sciencedirect.com/science/article/pii/S026322412100717X.

[20] S. Bottigliero, D. Milanesio, M. Saccani, R. Maggiora, A low-cost indoor real-time locating system based on TDOA estimation of UWB pulse sequences, IEEE Trans. Instrum. Meas. 70 (2021) 1–11, http://dx.doi.org/10.1109/TIM.2021.3069486.

[21] Q. Shi, X. Cui, S. Zhao, S. Xu, M. Lu, BLAS: Broadcast relative localization and clock synchronization for dynamic dense multiagent systems, IEEE Trans. Aerosp. Electron. Syst. 56 (5) (2020) 3822–3839, http://dx.doi.org/10.1109/TAES.2020.2979640.

[22] V. Djaja-Josko, Novel method for the wireless synchronization of the anchors in the UWB localization system utilizing two reference nodes, in: 2020 23rd International Microwave and Radar Conference, MIKON, 2020, pp. 69–73, http://dx.doi.org/10.23919/MIKON48703.2020.9253932.

[23] C. Xiaomao, F. Yiwei, L. Chunfei, Improved master-slave time synchronization algorithm based on TDOA positioning, in: Proceedings of the 2020 4th International Conference on Electronic Information Technology and Computer Engineering, in: EITCE 2020, Association for Computing Machinery, New York, NY, USA, 2020, pp. 210–215, http://dx.doi.org/10.1145/3443467.3443755.

[24] S. Niranjayan, A.F. Molisch, Ultra-wide bandwidth timing networks, in: 2012 IEEE International Conference on Ultra-Wideband, 2012, pp. 51–56.

[25] Y. Xie, G.J.M. Janssen, A. van der Veen, A practical clock synchronization algorithm for UWB positioning systems, in: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, 2016, pp. 3891–3895.

[26] A. Mahmood, R. Exel, H. Trsek, T. Sauter, Clock synchronization over IEEE 802.11—A survey of methodologies and protocols, IEEE Trans. Ind. Inf. 13 (2) (2017) 907–922, http://dx.doi.org/10.1109/TII.2016.2629669.

[27] R. Exel, Clock synchronization in IEEE 802.11 wireless LANs using physical layer timestamps, in: 2012 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication Proceedings, 2012, pp. 1–6, http://dx.doi.org/10.1109/ISPCS.2012.6336622.

[28] A. Mahmood, R. Exel, T. Sauter, Performance of IEEE 802.11's timing advertisement against SyncTSF for wireless clock synchronization, IEEE Trans. Ind. Inf. 13 (1) (2017) 370–379, http://dx.doi.org/10.1109/TII.2016.2521619.

[29] T. Schmid, R. Shea, Z. Charbiwala, J. Friedman, M.B. Srivastava, Y.H. Cho, On the interaction of clocks, power, and synchronization in duty-cycled embedded sensor nodes, ACM Trans. Sen. Netw. 7 (3) (2010) 24:1–24:19, http://dx.doi.org/10.1145/1807048.1807053, URL http://doi.acm.org/10.1145/1807048.1807053.

[30] Z. Zhong, P. Chen, T. He, On-demand time synchronization with predictable accuracy, in: INFOCOM, 2011 Proceedings IEEE, 2011, pp. 2480–2488, http://dx.doi.org/10.1109/INFCOM.2011.5935071.

[31] Seiko Instruments Inc, Product list, 2014, http://speed.sii.co.jp/pub/compo/quartz/productListEN.jsp. (Accessed 06 July 2021).

[32] F. Spitzer, Principles of Random Walk, in: Graduate Text in Mathematics, Springer, 2001.

[33] G. Huang, A.Y. Zomaya, F.C. Delicato, P.F. Pires, An accurate on-demand time synchronization protocol for wireless sensor networks, J. Parallel Distrib. Comput. 72 (10) (2012) 1332–1346, http://dx.doi.org/10.1016/j.jpdc.2012.06.003.

[34] Y. Jiang, V.C.M. Leung, An asymmetric double sided two-way ranging for crystal offset, in: 2007 International Symposium on Signals, Systems and Electronics, 2007, pp. 525–528, http://dx.doi.org/10.1109/ISSSE.2007.4294528.

[35] D. Neirynck, E. Luk, M. McLaughlin, An alternative double-sided two-way ranging method, in: 2016 13th Workshop on Positioning, Navigation and Communications, WPNC, 2016, pp. 1–4, http://dx.doi.org/10.1109/WPNC.2016.7822844.

[36] Decawave, The implementation of two-way ranging with the DW1000, 2018, https://www.decawave.com/wp-content/uploads/2018/10/APS013_The-Implementation-of-Two-Way-Ranging-with-the-DW1000_v2.3.pdf. (Accessed 24 July 2021).

[37] J.J. Pérez-Solano, S. Felici-Castell, Improving time synchronization in wireless sensor networks using Bayesian inference, J. Netw. Comput. Appl. 82 (2017) 47–55, http://dx.doi.org/10.1016/j.jnca.2017.01.007, URL http://www.sciencedirect.com/science/article/pii/S1084804517300164.

[38] Arduino, MKR ZERO, 2018, https://store.arduino.cc/arduino-mkr-zero-i2s-bus-sd-for-sound-music-digital-audio-data. (Accessed 25 July 2021).

[39] Thomas Trojer, Github: Library that offers basic functionality to use Decawave's DW1000 chips/modules with Arduino, 2018, https://github.com/thotro/arduino-dw1000. (Accessed 28 July 2021).

[40] Keysight Technologies, 1 GHz DSOX3104t Oscilloscope, 2019, https://www.keysight.com/en/pdx-x202179-pn-DSOX3104T/oscilloscope-1-ghz-4-analog-channels.(Accessed 26 July 2021).