UDC 004.71:004.032.3:611.08 doi: https://doi.org/10.20998/2522-9052.2023.4.04

Andrey Zuev, Dmytro Karaman, Andrey Olshevskiy

National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine

WIRELESS SENSOR SYNCHRONIZATION METHOD FOR MONITORING SHORT-TERM EVENTS

Abstract. **Topicality.** The key part of experimental research is obtaining the most accurate data about the studied object or event. Often, it is necessary to record parameters of processes that are c hallenging to precisely localize in space and time. The process or event under consideration can occur very rapidly, within fractions of a second, making it difficult for the researcher to deploy and configure recording equipment. This necessitates the creation of a network consisting of numerous recording devices to not miss critical events related to the studied process and obtain a sufficient volume of experimental data. Another issue is the synchronization of data obtained from different measurement devices. Spatially distributed recording devices must operate with a high degree of autonomy, leading to discrepancies in timekeeping and the need for synchronization. In processes with sub-millisecond durations, imperfections in timekeeping at each node have a significant impact: undetected and unaccounted discrepancies can lead to distortion or a misunderstanding of the overall picture of the studied process or event, even after acquiring all the necessary data. This is why the development and improvement of methods for synchronizing data recording nodes in distributed wireless sensor networks is important and urgent task. **Task statement.** One practical application of the proposed solution is the study of injuries caused by the penetration of foreign objects with high kinetic energy into the human body. These studies are conducted using artificial simulators of the human body made from composite materials and ballistic gelatin, with implanted electronic devices for recording changes in physical parameters. **Results and conclusions.** The article presents a hardware and software method, along with the technical implementation of the process for synchronizing the local clocks of wireless nodes, integrated into a unified information-measurement system located on the simulator. The proposed method allows achieving synchronization accuracy of no more than 12 μs/second using low-cost commercial off-the-shelf components. The practical part of the research discusses microprocessors from the ESP family, which, in general, provide sufficient time synchronization accuracy when using the proposed method, allowing for cost-effective node development within the system. The proposed method can also be applied in other fields, such as measuring vibrations in electrical machines and engines, as well as structural health monitoring.

Keywords: wireless sensors; distributed monitoring; clock synchronization; wound dynamics, human body simulators; medical diagnostics; wireless communication; information and measuring system; ESP32; ESP8266.

Introduction

When conducting experimental studies, there is often a need to acquire data that is available during very short-term processes, ranging from tenths to hundred-thousandths of a second. Due to the active development of integrated MEMS digital sensors the primary challenge is not solely the data acquisition speed at such short intervals, but rather the temporal synchronization of data obtained from multiple data acquisition points, especially when the data is collected by not just one computing system but by several independent devices joined together by network.

One such area is the experimental study of penetrating injuries caused by the impact of extraneous objects with high kinetic energy on the patient during an explosion, gunfire, or other incidents. An in-depth understanding of the physical processes accompanying the process of high-speed element intervention in the human body is of great importance for the successful treatment of the wounded.

The modern scientific approach to study the mechanisms of deformation in human soft tissues through the penetration of high-speed elements is divided into physical experiments and computer modeling. A typical approach to studying the process involves creating and using non-biological physical models or simulators in experimental tests, such as ballistic gelatin or clay, or even more complex composite models. The results of experimental research on these simulators are recorded using technical equipment, such as high-speed X-ray machines or computer tomography devices.

However, these existing experimental studies primarily are focused on the descriptive analysis of the resulting wound channel geometry based on the input characteristics (mass, velocity, attack angle) of penetrating elements (various shatters, bullets, buckshot, etc.). The experimental results obtained so far have provided an understanding of the general processes involved, but questions about their evaluation in heterogeneous environments that model the human body with a high degree of accuracy, such as a system of organs with significantly different physical properties, remain open. Experimental research data is used to verify the modeling results and the accuracy of computer models.

Despite significant progress in computer modeling of wound ballistics, such models are mostly suitable for replicating physical experiments at present. Furthermore, the evaluation of their accuracy is primarily based on criteria related to the correspondence of the wound channel's geometry. Meanwhile, various physical processes occurring within the material surrounding a penetrating high-speed element within the human body remain insufficiently researched. It's worth noting the work [1], in which the regularity of pressure changes in the simulator material at a specific distance from the wound channel during the penetration of high-speed elements was experimentally determined. Additionally,

© Zuev A., Karaman D., Olshevskiy A., 2023

the time duration of the pressure change process that needs to be recorded (250-500 μs) was also experimentally established, with the most active phase occurring within 50-150 μs. Thus, it is possible to determine the duration of the deformation process and establish the required accuracy in synchronizing the time of sensors that will capture the physical parameters of the deformation process, such as within 10% of the average duration of the active phase (10-15 μs).

The overall assessment of the current state of the problem concerning the study of dynamic processes of deformation and damage to soft tissues resulting from penetrating action, such as gunshot injuries, indicates that these scientific challenges require further research in both the direction of experimental exploration of a wide range of physical processes and the development of enhanced computer models and algorithms for simulation, along with the acquisition of experimental data.

Distributed information and measurement system description

Experimental research involves both planning natural experiments using a physical model and conducting the experiments themselves. These studies are carried out within the context of ballistic firing on a non-biological model (simulator) of human soft tissues at a specialized testing ground [2]. To measure the physical parameters of the process, an experimental setup is required, which typically includes:

1) Simulator – dense block of ballistic gelatin in which a variety of sensors can be placed. The choice of sensors depends on the parameters of the penetrating element's motion and the properties of the material being penetrated.

2) Information and Measurement System (IMS) – this system is responsible for recording and processing the signals received from the sensors.

The IMS is a distributed data collection and preliminary data processing system based on replaceable sensor nodes, where controllers, data input-output nodes, and sensors are distributed in space. Key features of such an IMS include decentralized data processing, the presence of distributed data input-output systems, increased fault tolerance, and a standard and unified database structure [3].

The main limitation in experiments of this kind is the practical inexpediency of laying wired lines from the equipment, which would increase the influence of various interferences related to the process of penetrating the element into the simulator during the experiment. Therefore, it is advisable for the IGN nodes of the IMS to use wired connections only directly with the sensors, while information transmission to the central computer is carried out through a wireless communication channel. However, unlike systems discussed in [3, 4], which operate in a long-term mode and monitor processes that take a considerable amount of time, the IMS designed for studying short-term processes, such as the process of a high-speed element penetrating a simulator, needs to have a clock synchronization system for its nodes. This system should also work over a wireless communication channel. This is essential so that, during subsequent research, data obtained from sensors at different nodes can be synchronized in time, allowing for the assessment of the state and parameter changes of the process as a whole within a unified time system.

Wireless device synchronization methods

Traditional clock synchronization protocols for wired networks cannot be directly used because wireless sensor network protocols require the ability to dynamically adapt, handle sensor network changes, and scale the network. The sensor nodes themselves may be severely resource-constrained due to limited battery power. Additionally, they must operate in an unreliable environment with significant communication channel losses. Several clock synchronization protocols have been developed for wireless networks.

Wireless network synchronization is achieved using a Network Time Protocol (NTP) and a Global Positioning System (GPS). Classical NTP [5] and TEMPO [6] protocols use an external source of accurate time, such as the GPS or Coordinated Universal Time (UTC), for network synchronization. In [7], a method of scalable time synchronization using GPS receivers for large systems consisting of multiple independent networks has been developed. In described wireless networks with gateway nodes, a GPS receiver in each gateway node is used to initiate and synchronize several nodes behind gateways. Subsequently, clock and data synchronization are performed to achieve high-precision time synchronization within each subnet.

GPS requires wireless devices to connect to satellites for synchronization, which necessitates a GPS receiver connected to each wireless node. However, due to power constraints, this can be impractical for wireless networks. Sensor networks consist of low-cost wireless nodes, and adding a GPS receiver to each wireless node increases the overall device cost. The accuracy of GPS time depends on how many satellites the receiver can communicate with at a given moment, and it may not always be consistent. Furthermore, GPS devices rely on line-of-sight satellite communication, which may not always be available. This makes it challenging to maintain a common time reference.

The Reference Broadcast Synchronization (RBS) method is based on minimizing nondeterministic delay through synchronization between receivers and saving energy by post-factum synchronization. Reference broadcast synchronization is a form of time synchronization that utilizes the broadcast property inherent to wireless communication. RBS messages do not contain explicit time stamps; instead, recipients use the time of arrival of the synchronization packet as a reference point to compare their clocks. The paper [8] discusses an example of measuring time stamps for two wireless nodes (using the standard IEEE 802.11 wireless network) and demonstrates that removing sender nondeterminism from the critical path of the packet ensures highly accurate clock synchronization with minimal energy consumption. The time synchronization protocol discussed in [9] is used to measure delays for wireless sensor networks and is an energy-efficient protocol, characterized by low message complexity. It requires fewer computational resources, however being less accurate than the RBS protocol discussed in [8]. Analysis conducted in the article shows that accuracy should degrade slowly the resulting error is equal to the distance between nodes.

The Romers protocol, as discussed in [10] and applied in mobile ad-hoc networks, utilizes a time transformation algorithm to achieve clock synchronization. This protocol is effective in resource-constrained environments. The Mocks protocol [11] extends the standard IEEE 802.11 synchronization protocol, making use of continuous clock synchronization in wireless real time applications, and improves the precision by exploiting the tightness of the communication medium, and tolerates message losses. Minimal message complexity and error resilience are highlighted as its main advantages. The network-wide time synchronization protocol presented in [12] is designed for networks with a high node density. The protocol described in [13] ensures good synchronization accuracy in wireless sensor networks using a deterministic protocol with minimal computational and storage complexity.

The Time-Diffusion Protocol (TDP) discussed in [14], achieves "synchronized" time across the entire network using an iterative weighted averaging methodology based on message propagation involving all nodes in the synchronization process. On the other hand, the Asynchronous Diffusion Protocol [15] employs a strategy similar to TDP, but network nodes execute the protocol and adjust their clocks asynchronously with respect to each other.

In [16], a post-measurement time synchronization approach is proposed, consisting of two implementations.

In the first implementation, beacon messages containing global clock information are periodically broadcasted during measurement. After the measurement is completed, estimated clock offsets are applied to achieve clock synchronization using nonlinear regression analysis. Then, data synchronization is achieved through resampling the data with a time discretization.

As a result, time synchronization accuracy can be improved by up to 40%, particularly when nonlinear clock drift occurs during long-duration measurements. Post-factum synchronization significantly reduces power consumption in scenarios where temporal and unpredictable synchronization is required. Instead of keeping clocks synchronized all the time, clocks operate autonomously at their natural pace and synchronize after an event of interest has occurred.

Specific properties of the proposed synchronization system

Therefore, after analyzing existing solutions for synchronization, three groups of synchronization methods can be identified: those using wired networks, those utilizing time correction algorithms, and those relying on an external source of accurate time or combinations thereof. However, clock synchronization doesn't necessarily guarantee synchronization of sensor data obtained from each node. For example, even if the clocks of sensor nodes are synchronized with a reference clock, the start time of measurements isn't entirely consistent among different sensor nodes. This is mainly due to the fact that most microcontrollers use event-driven operating systems in which tasks are executed on a "First In – First Out" basis, and such simplified task scheduling can introduce significant uncertainties in the timing of execution. Other factors contributing to synchronization errors include minor differences in sampling rates between sensor nodes and variations in the sampling rate over time [17].

CPU in each sensor node must schedule at least three operations simultaneously, including the measurement process, clock synchronization involving packet exchange over wireless communication, and data synchronization.

The lack of real-time scheduling support can have a negative impact on both measurement and synchronization:

1) unavoidable unexpected measurement delays can lead to imprecise data collection.

2) potential conflicts between acquiring a sample and wireless communication for synchronization messages may result in deviations in clock offset estimation, ultimately affecting the accuracy of time synchronization [19].

In [20], it is also mentioned that in a typical node running an event-driven operating system, it's impossible to precisely determine the start time of a measurement since all commands are passed through a queue, and the moment of execution cannot be recorded. It should be noted that when using sensors in interrupt-driven buffered input mode (for example, ADXL345), such determination is possible. When the internal buffer is filled to a specified level, the start time of measurement can be determined with accuracy within the measurement error by subtracting the period of the clock signal, multiplied by the number of counts in the buffer at which the interrupt occurs, from the local timestamp recorded as the beginning of the interrupt.

Another feature that distinguishes the system under consideration from a typical continuous-action sensor network is that theoretically, there is no need for time synchronization during the measurement process because the process is short-term (no more than a few seconds, including preparatory operations), and the error of the node's internal clock during this time will be insignificant.

However, immediately after synchronization, the difference will begin to deteriorate as the clocks drift apart. Therefore, the accuracy will decrease because more time elapses between the synchronization pulse and the event.

Typical crystal oscillators have accuracy on the order of one part in 10^4 to 10^6, which means the clocks of two nodes will deviate by 1-100 µs per second.

Typical accuracy specifications for RTC crystals range from ±100 to ±20 parts per million, equivalent to 8.6 to 1.7 seconds per day [18].

For the operation of the IMS, synchronization is only required before the start of the measurement or for a series of measurements that take a short time (a few minutes).

The algorithm steps for measurement during the testing process will be as follows:

1) network establishing (internal testing and node connection);

2) clock synchronization;

3) start measuring and recording sensor data in the node's internal memory;

4) end recording and storage session;

5) clock synchronization for additional verification;

6) transmission of stored values with timestamps and synchronization markers to the main computer of the system;

7) resampling of data obtained from all sensor nodes into a unified time system;

8) timestamping data with real-time scale;

9) storage of sensor data in a database for further processing.

An ideal node clock synchronization process can be defined as:

$$C_i(t) = t, \quad (1)$$

where $C_i(t)$ is the time of the clock on node i, and t is the time on the clock of the central computer (reference time). Therefore, if equation (1) is satisfied for all i, we have perfect clock synchronization. It can be noted that in this case

$$\frac{dC}{dt} = 1. \quad (2)$$

But for a real system, expression (2) can be defined as:

$$1 - p \leq \frac{dC}{dt} \leq 1 + p, \quad (3)$$

where p is a constant proportional to the skew rate, usually defined by the RTC manufacturer [21].

Graphs of microcontroller local clocks dependence options on absolute time are shown in Fig. 1.

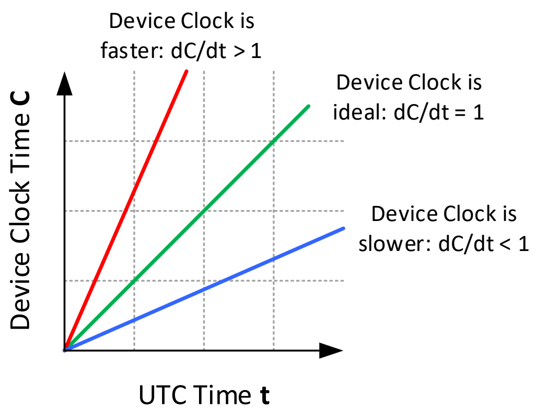

Fig. 1. Dependence of microcontroller local clocks on absolute time

In fact, any time synchronization method compensates for the value of p for each clock of each node in the system.

Method for local clocks synchronization

To solve the synchronization problem, a method based on transmitting a general sync signal wirelessly to each node is proposed (Fig. 2). An optical infrared (IR) emission is used as the medium for transmission. The sensors of the information-measuring system are statically localized, and the IR receiver is compact and can be integrated into the node's design. Besides, the receiver has low power consumption. All of these features allow for the use of such receivers for group synchronization of system nodes, slightly increasing its circuit complexity.

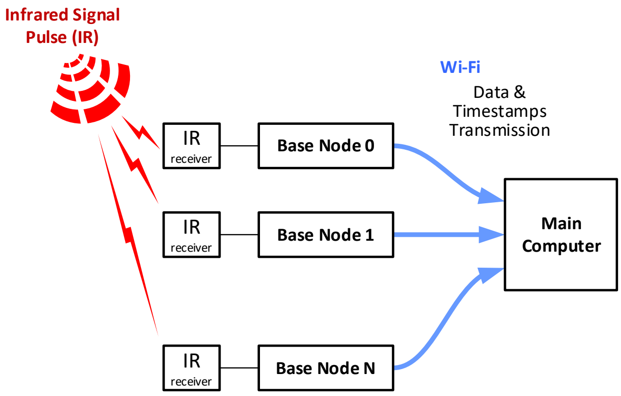

Fig. 2. Clock synchronization scheme using an IR clock signal

The synchronization algorithm is based on transmitting two synchronization pulses at the beginning and at the end of the measurement process. After receiving and recognizing the IR sync pulse, each node sends a readiness message (via Wi-Fi to the central computer), remembering the local time T_b at the start of the IR sync pulse.

Once messages are received from all nodes, the system is ready to start the measurements. Upon completing the measurement, another IR sync pulse is sent, and the time T_e of its arrival is also recorded. Thus, each node stores two readings of internal clocks, T_b and T_e, which are used for resampling the received data. Each data block is tagged with timestamps T_i (T_{bi}, T_{ei}) indicating the start and end of the measurement in the local time reference frame. All data received by the central computer is converted to a unified local time reference frame for one of the nodes (to which accurate clocks are connected). Knowing that T_{b0} and T_{e0} are the local times of a unified system corresponding to the sync pulse timestamps on the i-th node (T_{bi} and T_{ei}), and T_i is the timestamp of data arrival, you can determine the position of the T_{i0} timestamp in the unified system as follows:

$$T_{i0} = \frac{T_i - T_{bi}}{T_{ei} - T_{bi}}(T_{e0} - T_{b0}) + T_{b0}. \quad (4)$$

Collected data is resampled into real-time scale using these precise clocks.

This enables a unique alignment of measurements obtained from various sensors on different nodes and allows their use for validation and refinement of theoretical models.

Practical research and results

To validate the proposed method, a testbench was assembled. The ESP8266 and ESP32 MCU's running on NodeMCU and MicroPython (ESP32_GENERIC-20231005-v1.21.0 firmware) software, respectively, were used as sensor nodes. These nodes were connected via wired communication, with one node serving as a base and issuing a synchronization signal to the other nodes at defined time intervals. This signal was used as a reference synchronization signal. The node with the least deviation was chosen as the reference. The testbench setup for validating the synchronization algorithm is depicted in Fig. 3.

The synchronization scheme for the ESP32 MCU sensor nodes had similar structure, with the only difference being the use of different pins.

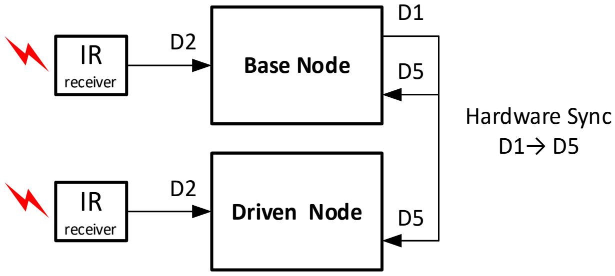

Fig. 3. Testbench setup
for synchronizing the ESP8266 sensor nodes

Additional tests were performed to measure the clock deviation concerning the reference signal for various ESP8266 MCU types. The results are presented in Fig. 4.

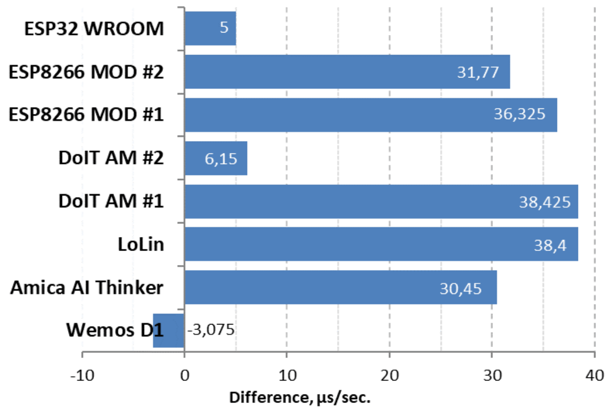

Fig. 4. Clock divergence between the local clocks
of various ESP8266 MCU types and the system reference clock

The results of this test also revealed that approximately 5 minutes after reboot, the clock deviation stabilizes, with deviations of less than 1% when the MCU operates at the standard frequency of 80 MHz.

This stabilization occurs after about 2-3 minutes when operating at an increased frequency (Fig. 5). It was established that the absolute value of the clock deviation is independent of the MCU's operating frequency.

After 60 seconds from the moment of startup, this value stabilizes and then changes slightly.

Fig. 6 depicts the variation in clock deviation over time at 10-second intervals for different frequencies, both for the reference and driven nodes (ESP32 MCU's).

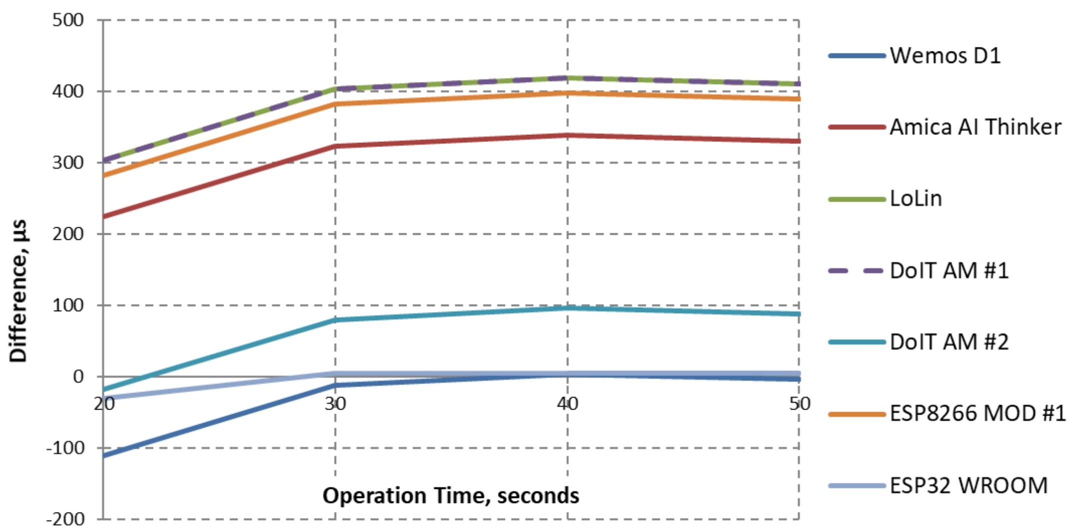

Fig. 5. Change in the magnitude of the discrepancy (µs/10 sec) over time from the moment of launch

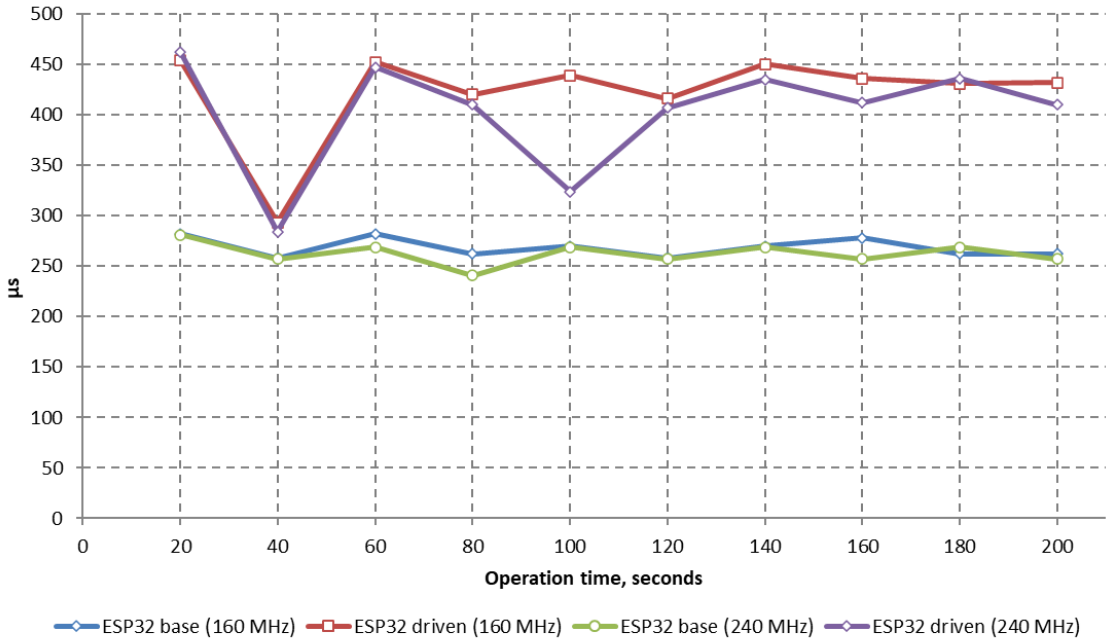

Fig. 6. Change in local clock error over time in µs per 10 second intervals, for different frequencies, for reference and driven nodes

Subsequently, measurements of the clock deviation were taken when synchronizing two ESP32-based nodes using the IR sync pulse. To achieve this, a procedure for receiving messages from the IR receiver was added to the node's software.

An IR sensor device, TSOP4838, operating at a frequency of 38 kHz, was used for receiving the IR signal. An IR synchronization pulse is sent between the pulses of the wired synchronization. The time readings obtained at the moment of receiving the IR pulse on two nodes are compared with the wired synchronization pulses according to equation (4) (see Fig. 7).

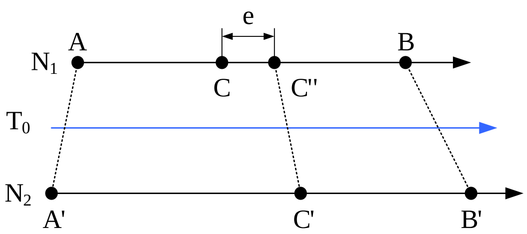

Fig. 7. Local clock synchronization of two nodes and determination of the synchronization error e using an IR sensor

N_1 and N_2 are axes of the local clocks of nodes 1 and 2, T_0 – axis of ideal clock time, A, B, A', B' – synchronizing markers that match in pairs on two nodes (in the coordinates of local axes N_1 and N_2 respectively), C, C' – synchronization marker obtained using the IR sensor on nodes 1 and 2 respectively, C'' – projection of synchronization marker C' (node 2) into the local clock reference system of node 1, e = C'' – C – synchronization error in the local clock reference system of node 1.

Based on the comparison results, the IR synchronization error can be estimated (Fig. 8).

It has been determined that the maximum error does not exceed 106 µs for a MCU operating at 240 MHz and 115 µs for 160 MHz. In fact, increasing the MCU's operating frequency results in an 8% reduction in error.

The temperature change in the chips during the measurement (200 seconds) was 46±4%. It should be noted that there is a significant deviation in the local clocks for ESP-8266 nodes under the control of the NodeMCU system, exceeding 30 µs per second.

Conclusions

Overall synchronization error using the proposed method does not exceed 12 µs per second.

There is no significant change in error depending on the MCU's operating frequency. ESP32 family microprocessors, in general, provide sufficient accuracy (within 10-15 µs) for time synchronization of local clocks when using the proposed method, allowing them to be used as a basis for IMS nodes in the detection of parameters of short-term events, such as the penetration of high-speed objects into human body simulators.

The proposed synchronization method can also be used in other fields, such as measuring electrical machine vibrations and monitoring the condition of various structures (structural health monitoring).

It is advisable to conduct further research, including an efficiency examination of using microcontrollers of different types, such as Raspberry Pi, STM32 (Nucleo, Discovery, Pyboard series), Renesas RA series, etc.

Additionally, it would be beneficial to test other types of firmware and investigate the relationship between the error and the type of IR sensors and onboard Wi-Fi modules used. Further considerations could also include data resampling and determining absolute time.

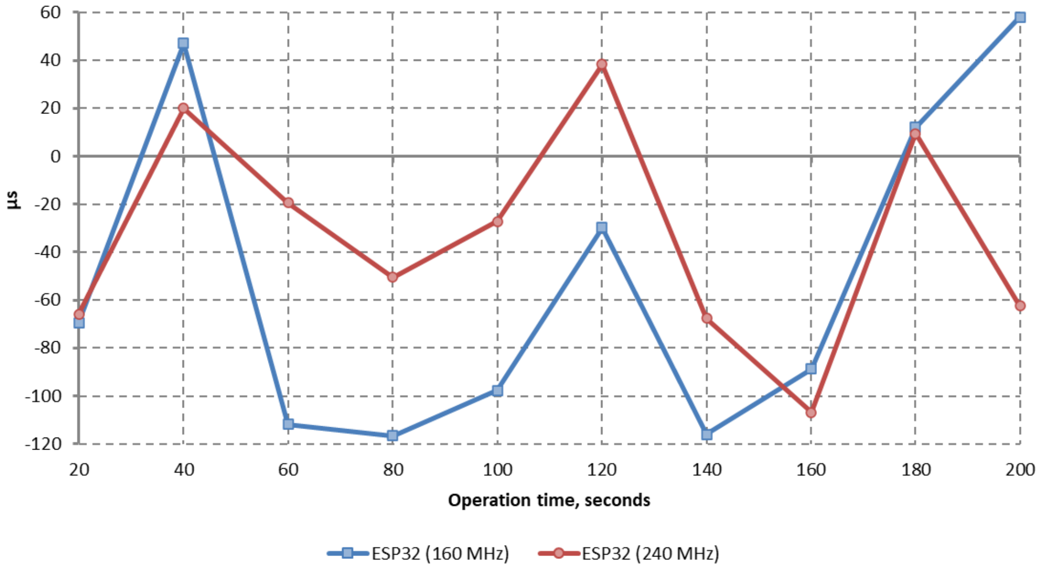

Fig. 8. Synchronization error with using an IR sensor relative to reference synchronization at 10-second intervals

References

1. Yaoke, Wen, Cheng, Xu, Haosheng, Wang, Aijun, Chen and R. C., Batra (2013), "Impact of steel spheres on ballistic gelatin at moderate velocities", *International Journal of Impact Engineering*, Vol. 62, pp. 142–151, doi: https://doi.org/10.1016/j.ijimpeng.2013.07.002
2. Tomashevskyi, R., Larin, O., Kolisnyk, K., Zuev, A., Gumeniuk, K., Lurin, I. and Nehoduiko, V. (2023), "Methodology and Use of Experimental Techniques in Analyzing Wound Dynamics of Penetrating Injuries", *Proc. ICNBME 2023: 6th International Conference on Nanotechnologies and Biomedical Engineering*, Vol. 92, Springer, Cham, pp. 208–217, doi: https://doi.org/10.1007/978-3-031-42782-4_23
3. Zuev, A. and Vodka, O. (2022), "Development of a Distributed Wireless Vibration Measurement and Monitoring System", *Proc. 2022 IEEE 3rd KhPI Week on Advanced Technology* (KhPIWeek), Kharkiv, Ukraine, pp. 915–920, doi: https://doi.org/10.1109/KhPIWeek57572.2022.9916375
4. Zuev, A., Ivashko, A. and Lunin, D. (2022), "Diagnosis Methods for Mechanisms and Machines Based on Empirical Mode Decomposition of a Vibrosignal and the Wilcoxon Test", *Advanced Information Systems*, Vol. 6, No. 4, pp. 51–57, doi: https://doi.org/10.20998/2522-9052.2022.4.07
5. Mills, D. L. (1991), "Internet time synchronization: the network time protocol", *IEEE Transactions of Communications*, Vol. 39, No. 10, pp. 1482–1493, doi: https://doi.org/10.1109/26.103043
6. Gusella, R. and Zatti, S. (1984), "TEMPO – A network time controller for a distributed Berkeley UNIX system", *IEEE Distributed Processing Technical Committee Newsletter*, No. 6 (CSD-83-163), pp. 7–14.
7. Kim, R. E., Li, J., Spencer, B.F., Nagayama, T. and Mechitov, K.A. (2016), "Synchronized sensing for wireless monitoring of large structures", *Smart Structures and Systems*, Vol. 18(5), pp. 885–909, doi: https://doi.org/10.12989/sss.2016.18.5.885
8. Elson, J., Girod, L. and Estrin, D. (2002), "Fine-grained network time synchronization using reference broadcasts", *Proc. of 5th Symposium on Operating Systems Design and Implementation* (OSDI 2002), Vol. 36, pp. 147–163, doi: https://doi.org/10.1145/844128.844143
9. Ping, S. (2003), *Delay measurement time synchronization for wireless sensor networks*, Intel Research, IRB-TR-03-013, June 2003, available at: https://www.researchgate.net/publication/252898205
10. Romer, K. (2001), "Time synchronization in ad hoc networks", *Proc. of ACM Symposium on Mobile Ad Hoc Networking and Computing* (MobiHoc #01), October 2001, pp. 173–182, doi: https://doi.org/10.1145/501436.501440
11. Mock, M., Frings, R., Nett, E. and Trikaliotis, S. (2000), "Continuous clock synchronization in wireless real-time applications", *Proc. of 19th IEEE Symposium on Reliable Distributed Systems* (SRDS-00), October 2000, pp. 125–133, doi: https://doi.org/10.1109/RELDI.2000.885400
12. Ganeriwal, S., Kumar, R., Adlakha, S. and Srivastava, M. (2003), *Network-wide time synchronization in sensor networks*, Technical Report, Networked and Embedded Systems Lab, Electronic Engineering Department, UCLA, 2003, doi: https://doi.org/10.1109/MNET.2004.1316761
13. Sichitiu, M.L. and Veerarittiphan, C. (2003), "Simple, accurate time synchronization for wireless sensor networks", *Proceedings of IEEE Wireless Communications and Networking Conference* (WCNC 2003), pp. 1266–1273, doi: https://doi.org/10.1109/WCNC.2003.1200555
14. Su, W. and Akyildiz, I. (2004), "Time-Diffusion Synchronization Protocols for Sensor Networks", *IEEE/ACM Transactions on Networking*, Vol. 13, No. 2, pp. 384–397, doi: https://doi.org/10.1109/TNET.2004.842228

Li, Q. and Rus, D. (2004), "Global clock synchronization in sensor networks", *Proc. of IEEE Conference on Computer Communications* (INFOCOM 2004), Vol. 1, Hong Kong, China, pp. 564–574, doi: https://doi.org/10.1109/INFCOM.2004.1354528

16. Li, J., Mechitov, K. A., Kim, R. E. and Spencer, B.F.Jr. (2016), "Efficient Time Synchronization for Structural Health Monitoring Using Wireless Smart Sensor Networks", *Structural Control and Health Monitoring*, Vol. 23(3), pp. 470–486, doi: https://doi.org/10.1002/stc.1782

17. Fu, Y., Mechitov, K., Hoang, T., Kim, J.R., Memon, S.A. and Spencer B.F.Jr. (2020), “Efficient and high-precision time synchronization for wireless monitoring of civil infrastructure subjected to sudden events”, *Structural Control and Health Monitoring*, Vol. 28(3), October 2020, 23 p., doi: https://doi.org/10.1002/stc.2643
18. Vig, J.R. (1992), *Introduction to Quartz Frequency Standards*, Technical Report SLCET-TR-92-1, Army Research Laboratory, Electronics and Power Sources Directorate, September 1992, 55 p., doi: https://apps.dtic.mil/sti/citations/ADA248503
19. Li, J., Nagayama, T., Mechitov, K.A. and Spencer, B.F. (2012), “Efficient campaign-type structural health monitoring using wireless smart sensors”, *Proc. Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems*, April 3, 2012, Vol. 8345, p. 1–11, 83450U, doi: https://doi.org/10.1117/12.914860
20. Spencer, B.F., Nagayama, T. and Rice, J.A. (2008), “Decentralized Structural Health Monitoring Using Smart Sensors”, *Proc. of SPIE 6932, Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems*, doi: https://doi.org/10.1117/12.791077
21. Sundararaman, B., Buy, U. and Kshemkalyani, A.D. (2005), “Clock synchronization for wireless sensor networks: A survey”, *Ad Hoc Networks*, Vol. 3(3), May 2005, pp. 81–323, doi: https://doi.org/10.1016/j.adhoc.2005.01.002

Received (Надійшла) 31.05.2023
Accepted for publication (Прийнята до друку) 16.08.2023

ВІДОМОСТІ ПРО АВТОРІВ / ABOUT THE AUTHORS

Зуєв Андрій Олександрович – кандидат технічних наук, доцент, завідувач кафедри автоматики та управління в технічних системах, Національний технічний університет «Харківський політехнічний інститут», Харків, Україна;
Andrey Zuev – Candidate of Technical Sciences, Associate Professor, Head of the Department of automation and control in technical systems, National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine.
e-mail: andrii.zuiev@khpi.edu.ua; ORCID ID: http://orcid.org/0000-0001-8206-4304.

Караман Дмитро Григорович – старший викладач кафедри автоматики та управління в технічних системах, Національний технічний університет «Харківський політехнічний інститут», Харків, Україна;
Dmytro Karaman –Senior Lecturer of the Department of automation and control in technical systems, National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine.
e-mail: dmytro.karaman@khpi.edu.ua; ORCID ID: http://orcid.org/0000-0002-7252-3172.

Ольшевський Андрій Вікторович – аспірант кафедри автоматики та управління в технічних системах, Національний технічний університет «Харківський політехнічний інститут», Харків, Україна;
Andrey Olshevskiy – PhD student of the Department of automation and control in technical systems, National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine.
e-mail: asd.andry.asd@gmail.com; ORCID ID: http://orcid.org/0000-0003-2734-9491.

Метод синхронізації бездротових датчиків для моніторингу короткочасних подій

А. О. Зуєв, Д. Г. Караман, А. В. Ольшевський

Анотація. **Актуальність.** Ключовою частиною експериментальних досліджень є отримання максимально достовірних даних про досліджуваний об'єкт або подію. Часто доводиться реєструвати параметри процесів, які дуже важко чітко локалізувати у просторі та часі. Процес чи подія, що розглядається, може протікати дуже швидко, частки секунди, що не дозволяє досліднику встигнути розгорнути і налаштувати засоби реєстрації. Це призводить до необхідності створення мережі, яка складається з великої кількості пристроїв реєстрації для того, щоб не пропустити важливу подію, пов'язану з процесом, що вивчається, і отримати достатній обсяг експериментальних даних. Ще однією проблемою є узгодження даних, отриманих від різних засобів виміру. Рознесені в просторі пристрої реєстрації вимушено мають високий рівень автономності, що призводить до неузгодженості відліку часу і необхідності синхронізації годинника. При субмілісекундній тривалості процесів, що вивчаються, недосконалість засобів відліку часу в кожному вузлі має величезний вплив: невиявлені і невраховані неузгодженості можуть призвести до спотворення або неправильного розуміння загальної картини досліджуваного процесу або події навіть після отримання всіх необхідних даних. Саме тому створення та вдосконалення методів синхронізації вузлів реєстрації даних у розподілених бездротових сенсорних мережах є важливим та актуальним завданням. **Постановка завдання.** Практичною сферою застосування запропонованого рішення є дослідження травм, що спричинені проникненням чужорідних об'єктів з високою кінетичною енергією у тіло людини. Ці дослідження проводяться із застосуванням штучних імітаторів людського тіла на базі композитних матеріалів та балістичного желатину з імплантованими електронними засобами реєстрації змін фізичних параметрів. **Результати та висновки.** У статті запропоновано програмно-апаратний метод та показано технічну реалізацію процесу синхронізації локальних годинників бездротових модулів, об'єднаних в єдину інформаційно-вимірювальну систему, розміщену на імітаторі. Запропонований метод дозволяє отримати похибку синхронізації не вище 12 мкс/сек за використання комерційних компонентів невисокої вартості. Розглянуті в практичній частині дослідження мікропроцесори сімейства ESP загалом забезпечують достатню точність синхронізації локальних годинників при використанні запропонованого методу, що дозволяє знизити вартість вузлів системи. Запропонований метод також може бути використаний в інших областях, таких як вимірювання вібрацій електричних машин і двигунів, моніторинг стану різних структур.

Ключові слова: бездротові датчики, розподілений моніторинг, синхронізація годинників, динаміка поранення, симулятори людського тіла, медична діагностика, бездротовий зв'язок, інформаційно-вимірювальна система, ESP32, ESP8266.