Chapter 9
OASIS: Open Acquisition System for IEPE Sensors: For Academic Research and Teaching Purposes

Oliver M. Zobel, Johannes Maierhofer, and Daniel J. Rixen

Abstract Expensive measurement equipment often inhibits students from gaining practical experience with vibration measurements. The commercially available, proprietary hardware and software additionally does not grant insight into the used algorithms or allow modification and expansion. Both issues are overcome with the use of community developed open-source designs. This chapter discusses the requirements for vibration measurements in an academic context, like sampling frequency or resolution and especially synchronicity. These requirements are derived and incorporated into the design of an open-source data acquisition board for IEPE sensors (OASIS). The design is focused on the use of commonly available parts and a broadly community supported micro-controller (ESP32 family), with the costs bounded below 100 euro.

The built acquisition system offers four channels for IEPE signals, which can be sampled with up to 20 kHz and 16-bit resolution continuously. Additionally, wireless synchronization of multiple data acquisition boards is provided. This allows for use cases on systems with partly rotating structures, e.g., wind turbines, where synchronous measurements of stationary and rotating parts are necessary. The here-proposed approach does not rely on external clocks, like GPS or network services. An experimental validation shows that it is possible to synchronize two systems using this approach with a delay that is less than 100 µs.

Keywords Experimental dynamics · Data acquisition · Vibration analysis · Data synchronization · Measurement equipment

9.1 Introduction

Teaching of vibration measurement at universities is often limited to a theoretical introduction, missing the practical application in general. One factor for this might be the quite expensive measurement equipment, which makes large lab courses with equipment for all students impossible. Prices for small 4-channel data acquisition systems, which are the focus of this chapter, can reach into the high thousands of euros, not including the required, proprietary software.

This leads to the idea to design and build self-developed, cheap measurement hardware and software. Of course, this kind of equipment cannot match the accuracy, robustness, or reliability of commercially available products, but it possesses other advantages. Besides the considerably lower costs, it enables complete insight into the hardware and software design. This allows for adaptions, like additional hardware modules or new software features. Providing such hardware and software as open-source extends the concept even further, allowing a broad community to contribute. A solution for these demands was developed at the Chair of Applied Mechanics over the last 3 years. Figure 9.1 shows the latest prototype of the data acquisition system, which will be detailed in this chapter.

The goal of this chapter is to derive the general requirements of data acquisition systems for vibration measurements in an academic context. This includes basics, like sampling frequency or resolution, but the main focus is the required synchronicity between the acquired signals. To discuss synchronicity, a distinction needs to be made between the synchronous start of all measurements and sampling all signals with the same frequency. An illustration of the two effects is given in Fig. 9.2.

Recordings that start incorrectly, but with an exact sample interval (blue), lead to a constant phase shift in the reconstructed signal with respect to the original signal, resulting in altered phase relations with other observed signals. The start time offset

O. M. Zobel (✉) · J. Maierhofer · D. J. Rixen
Chair of Applied Mechanics, TUM School of Engineering and Design, Technical University of Munich, Garching, Germany
e-mail: oliver.zobel@tum.de; j.maierhofer@tum.de; rixen@tum.de

C. Walber et al. (eds.), *Sensors & Instrumentation and Aircraft/Aerospace Testing Techniques, Volume 8*, Conference Proceedings of the Society for Experimental Mechanics Series, https://doi.org/10.1007/978-3-031-34938-6_9

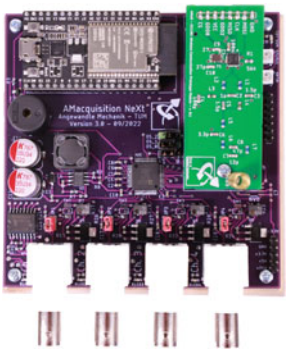


Fig. 9.1 Self-developed acquisition system

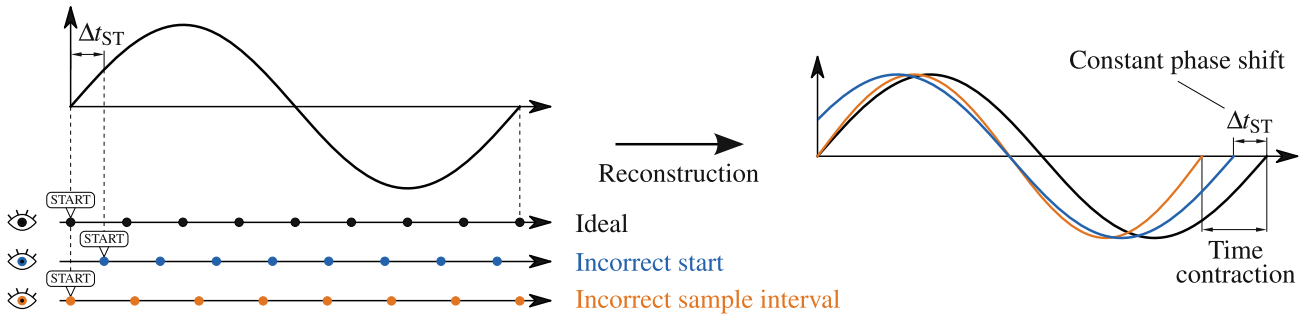


Fig. 9.2 Illustration of the two kinds of synchronicity errors and arising issues after signal reconstruction

Δt_{ST} should therefore be minimal. Measurements that start perfectly synchronous, but with an incorrect sample interval (orange), e.g., from different reference clocks, lead to a reconstructed signal that is either contracted or stretched in the time dimension. The cause of this error is described as drift per second τ_{drift}, the time added or subtracted from one true second of time. A positive drift means that the sample interval is too long, and the fixed number of samples, derived from sampling frequency and measurement duration, takes longer to acquire. All these considerations do not include sampling jitter, defined as the random deviation from the exact sampling interval [1].

Simultaneous sampling of multiple inputs within one Analog–Digital Converter (ADC) is ready to use for high-quality devices. But connecting multiple devices together needs careful attention to provide synchronicity. While synchronization of multiple systems via cable is commonly available, wireless systems often rely on external clocks like GPS or network services. Both methods use absolute timestamps added to the signal in order to synchronize the acquired data in post-processing. The shortcoming of GPS lies in the required reception of GPS signals, which might be troublesome indoors, while the implementation of time-sensitive networking is quite complex. Here, an approach for wireless synchronization of data acquisition systems is proposed that is self-contained and nearly without overhead. This allows for use cases on systems with partly rotating structures, e.g., wind turbines, where synchronous measurements of stationary and rotating parts are necessary.

From the derived requirements, an acquisition system is built that is focused on the use of commonly available parts and a broadly community supported micro-controller (*Espressif ESP32*). To achieve affordability, the costs are bounded below 100 euro. Additionally, the system is open-sourced to enable everyone to use, understand, and extend its functionality.

9.2 Performance Requirements for Data Acquisition

The main step of data acquisition is the conversion of the analog measurement signals to digital values. During this, the signals are discretized in time and amplitude (quantization) [2]. This raises two choosable parameters for an acquisition system: the (maximum) sample rate f_S and the quantization resolution n_{ADC}. While the latter solely depends on the specifications of the chosen ADC, the former depends on the whole system performance. With the focus being modal analysis, substructuring, or similar evaluations with the acquired data, the phase relation of the different signals as well as accurate frequency reconstruction is important. As described in the introduction, this introduces two additional requirements:

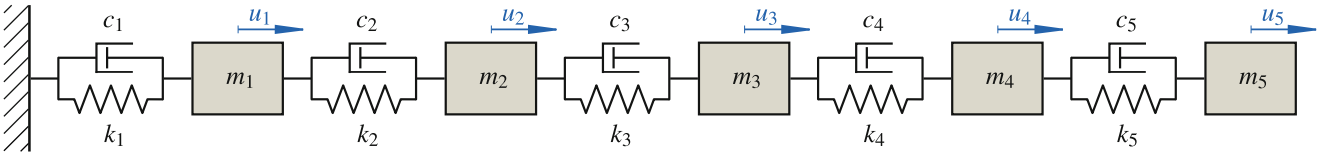


Fig. 9.3 Lumped mass oscillator model for the investigation of synchronicity requirements, with five degrees of freedom u_i

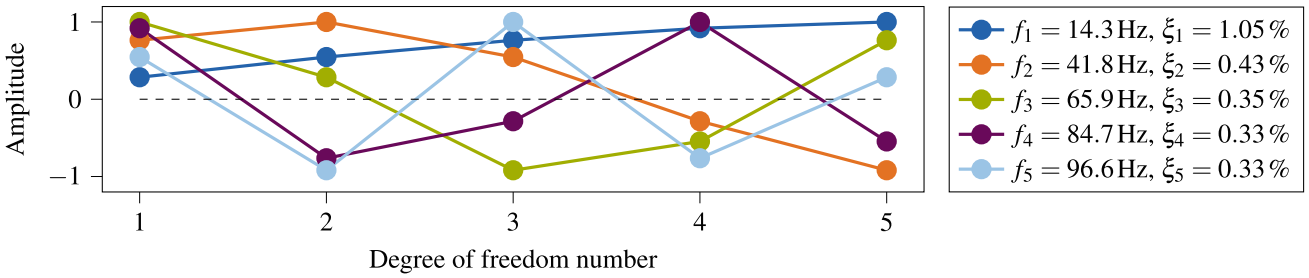


Fig. 9.4 Original mode shapes of the five mass oscillators with eigenfrequencies f_i and modal damping ratio ξ_i

the maximally allowed start time offset Δt_{ST} and drift per second τ_{drift}. Before designing the hardware, these requirements are derived individually.

Sample rate For the validation of new methods, e.g., for experimental substructuring on aluminum structures as in [3], frequency response functions up to 3 kHz are of interest. Adding additional headroom to avoid the area around the cutoff frequency of any low-pass filters, 5 kHz bandwidth, and with the Nyquist–Shannon theorem a sampling frequency f_S of at least 10 kHz is desired. Since most cheap ADCs can handle sampling frequencies of 50 kHz and above, this requirement mostly applies to the software driving the ADC, rather than the hardware itself. For applications with impulse hammers, a higher sample rate might be of interest in order to fully capture the impulse, especially the force peak.

Quantization resolution The quantization resolution mainly determines the achievable dynamic range D, defined as the ratio between the highest and lowest values representable after digitization. It can be approximated by $D \approx 6 \cdot n_{\text{ADC}}$ and is usually around 90–100 dB for most vibration measurement transducers [2]. Considering the cost constraint, the lower bound of 90 dB is set as a requirement. This results in a minimum ADC resolution n_{ADC} of 15 bits. Since the usual measurement signals, e.g., accelerations, also carry a sign, a bipolar ADC with 16-bit resolution is the minimum.

Time synchronicity In order to quantify the errors resulting from synchronicity issues, a simple numerical model is set up, namely a lumped mass oscillator with five degrees of freedom (dof) as depicted in Fig. 9.3.

The model consists of five masses, connected with spring–damper elements, with the leftmost one rigidly connected to the environment. The matrices are set up in Python and integrated using Newmark's method with $\gamma = \frac{1}{2}$ and $\beta = \frac{1}{4}$ (average constant acceleration) [4], using the identified, minimum sampling rate of 10 kHz. The system is excited by an impulse-shaped cosine that is applied successively to each dof. A start time offset Δt_{ST} or drift per second τ_{drift} can then be applied to the displacement signals. In all cases, the full frequency response function (FRF) matrix is calculated using *pyFRF* [5]. The modal parameters and eigenvectors are then extracted with *pyFBS* [6], using a combination of the poly-reference Least-Squares Complex Frequency (pLSCF) and Least-Squares Frequency Domain (LSFD) methods [7]. The original, i.e., unmodified, modes are shown in Fig. 9.4.

Figure 9.4 shows the real part of the eigenvectors, after they have been normalized such that the highest amplitude is one and rotated such that dof 1 has no imaginary part. While this might be handy for the visualization of real modes, it also removes information about the phase between the different nodes. Since the used model is proportionally damped (damping matrix can be constructed from a linear combination of the mass and stiffness matrices), the eigenvectors are purely real and the phase is either 0° or 180° [8]. But having one (sensor) signal shifted in time due to a start time offset Δt_{ST} introduces a phase shift depended on frequency. Such (small) changes in phase are then not clearly visible, as illustrated in Fig. 9.5.

Phase perturbations of purely real eigenvectors by small values are barely visible when only considering the real part. They are better visible after a rotation ωt, with ω being the eigenfrequency corresponding to the eigenvector, but the amplitude is arbitrary. To tackle this issue, the eigenvectors are shown in the complex plane. Since the phase shift due to start time offsets Δt_{ST} is increasing with frequency, the mode with the highest eigenvalue is affected the most. Therefore, only mode

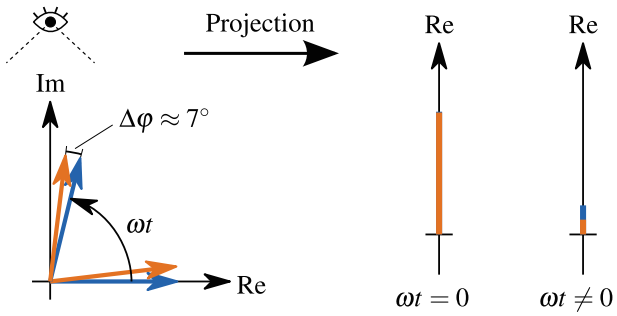


Fig. 9.5 Illustration of issue arising from viewing only the real part of eigenvectors: small phase changes of (mostly) real eigenvectors are barely recognizable, blue: unmodified and orange: phase shift of $\Delta\varphi \approx 7°$ introduced

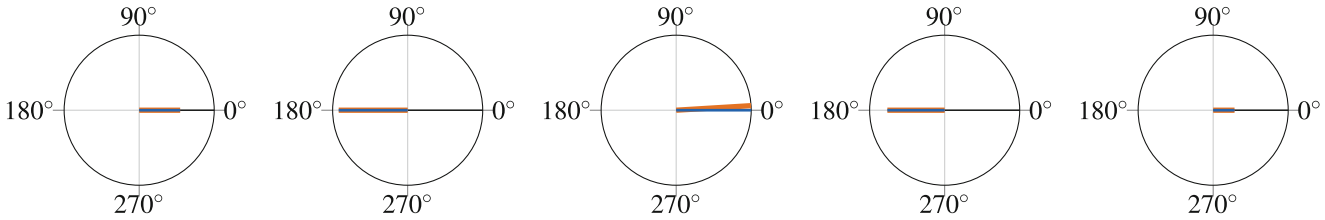


Fig. 9.6 Visualization of mode 5 in the complex plane, with dof 3 shifted in time by one sample (orange), $\Delta\varphi \approx 3.5°$

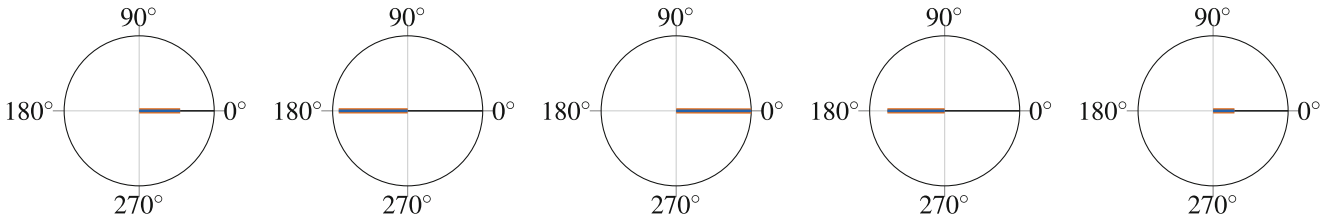


Fig. 9.7 Visualization of mode 5 in the complex plane, with dof 3 drifting 30 µs/s, $\Delta\varphi \approx -0.1°$

5 is shown in the following. The synchronicity issues are only applied to dof 3, to simulate one board not being perfectly synchronous.

At first, the influence of a start time offset Δt_{ST} is investigated. In order to avoid a resampling of the signal, the shifting is done in steps of $\Delta t = 1/f_S$, resulting in the smallest offset being 100 µs. The components of the fifth eigenvector are shown in Fig. 9.6, with the original eigenvector components in blue and the ones with start time offset Δt_{ST} applied to dof 3 in orange.

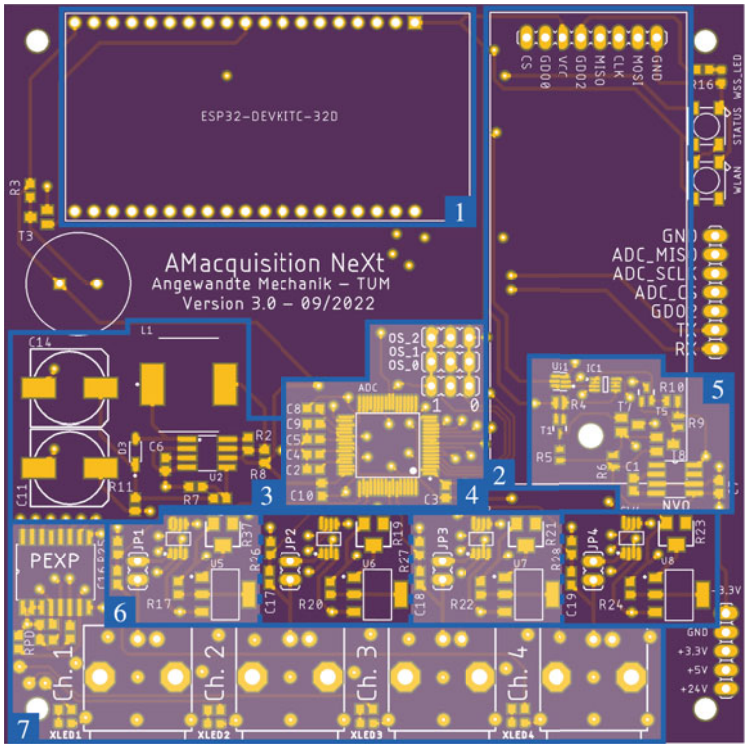
As can be seen, the dof 3 component now has an imaginary part or in other words is rotated in the complex plane by about 3.5°. The same value can be found by calculating the phase shift with $\Delta\varphi = \Delta t_{\mathrm{ST}}/T_{\mathrm{mode}} \cdot 360°$, where T_{mode} is the period of the respective mode. Calculating the phase shift of a theoretical mode at 3 kHz, assuming the relation holds, results in an error of 108°, which is by no means acceptable. A maximum phase shift error of 10° at the highest frequency of interest is set as the limit requirement. This results in a maximally allowed start time offset Δt_{ST} of about 9 µs. This requirement is automatically fulfilled within one system with a simultaneously sampling ADC, but is very challenging for connecting multiple devices.

The second issue of drift between clocks is investigated based on the observed drift of two *Espressif ESP32* microcontrollers. Both controllers were sampling the output of a signal generator for 500 s, after which the offset between the sampled signal and a generated, perfect signal was compared. They both showed a drift τ_{drift} of 30 µs/s. A simulation of this kind of drift results in a neglectable phase error of $-0.1°$, which is not even visible in the complex plane, see Fig. 9.7.

This error is smaller than the one resulting from a start time offset $\Delta t_{\mathrm{ST}} \leqslant 9\,\mu\mathrm{s}$, meaning that the error introduced by drifting over time is negligible in comparison. Therefore, the determined *Espressif ESP32* performance is set as the minimum requirement. Table 9.1 summarizes the determined performance requirements.

Table 9.1 Summary of the identified minimum requirements for data acquisition for experimental vibration measurements

Quantity	Minimum requirement
Sample rate f_S	$\geqslant$10 kHz
ADC resolution n_{ADC}	$\geqslant$16 bit
Start time offset Δt_{ST}	$\leqslant$9 µs
Drift τ_{drift}	$\leqslant$30 µs/s



Acquisition board sub-modules

1. Micro-controller module
2. Wireless synchronization module
3. 24 V boost converter
4. Analog-Digital-Converter (ADC)
5. Logic inverter for TEDS readout
6. Acquisition back end (1 per channel)
7. Acquisition front end

Fig. 9.8 Overview of the designed acquisition board with important sub-modules highlighted and described below

9.3 Acquisition Board Design

The requirements derived in the previous section were incorporated into a board design that was further refined and extended over two iterations. The built acquisition system offers four channels for IEPE signals, which can be sampled with up to 20 kHz and 16-bit resolution continuously. Making use of the PSRAM up to 50 kHz can be achieved within a limited period. Additionally, the readout of the transducer electronic data sheets (TEDSs) is possible. Thanks to the micro-controller's Wi-Fi capability, both wired (single USB 2.0 connection) and wireless operation modes are available. Figure 9.8 shows the current design and highlights important sub-modules, whose functions will be described in the following. As part of the open-source requirement, full schematics and software can be found on the official page: https://github.com/Official-OASIS-Project.

Micro-controller module The self-contained *Espressif Systems ESP32-DevKitC-VE* development kit is used, which incorporates all required components for the micro-controller, i.e., a 3.3 V supply or a USB-to-UART Bridge supporting up to 3 Mbps. The latter is important to achieve a sample rate of 20 kHz with a wired connection, and other kits with 1 Mbps transfer rate are not sufficient. Another reason for this specific kit is the *Espressif Systems ESP32-WROVER-E* module, which possesses additional PSRAM used for caching acquisition data. Since most computation time for processing a sample is spent on the data transfer from the micro-controller to an external device, moving this step after the actual acquisition phase greatly increases the achievable sample rate to up to 50 kHz. The built-in Wi-Fi module additionally enables to retrieve the acquisition data wirelessly.

Wireless synchronization module This module sends and receives synchronization impulses to or from other acquisition boards. It is configured by the micro-controller via SPI and provides a rising edge when a synchronization impulse was successfully sent or received. The design is based on the *Texas Instruments CC1101* chip and implements the reference design.

Analog–Digital Converter (ADC) According to the derived requirements, a simultaneously sampling, bipolar ADC was selected. With the overall budget limit in mind, the cheaper *AD7606-4* from *Analog Devices* was selected. Compared to the more than twice as expensive flagship *AD7606C-18*, it lacks 2 bits of resolution (16 bits instead of 18 bits) and the ability to configure it via SPI. Therefore, access to the oversampling configuration is provided with jumpers. These two ADC models are not pin compatible and require further changes if an interchange is desired.

Logic inverter for TEDS readout In order to read data from the sensor EEPROM, a signal with negative potential with respect to ground has to be applied to the TEDS enabled sensor. Data can be read using the OneWire protocol and the standard library using this logic inverter circuit. So far the specifics for the communication with accelerometers using TEDS version 0.9 have been worked out and successfully validated separately. At the time of writing, the implementation of the logic inverter shows instability issues and cannot be used reliably.

24 V boost converter IEPE sensors require a constant current source that can provide up to 24 V. This module creates a 24 V voltage source from the 5 V provided by the USB connection. This is done with a boost converter using the *MC34063A* chip, implementing the reference design with components chosen such that 25 mA can be provided continuously. An additional RC filter is fitted to reduce the voltage ripple.

Acquisition back end Each channel has its own back end consisting of a constant current source provided by the *Analog Devices LT3092* chip. The current is adjustable using a potentiometer. The back end also handles the switching between current source and TEDS logic inverter using the *Analog Devices ADG419* chip. Furthermore, a high-pass filter is implemented that removes the DC part (commonly near 12 V) of the sensor signal. The filter is designed such that the influence on frequencies above 10 Hz is minimal, not disturbing the usual working range of piezo accelerometers.

Acquisition front end The front end contains the BNC connectors for the sensors as well as a port expander (*PCA9554A*), which controls the switching of the *ADG419* chips as well as LEDs for status feedback. Using an expander is necessary as there are not enough channels left on the micro-controller otherwise. Additional LEDs and a buzzer are fitted on the board for extensive feedback for the user, which also proved quite helpful during troubleshooting, just as the optional debug headers.

9.4 Wireless Sample Synchronization: Operation Principle and Validation

The idea is to have one device act as the source, which transmits a synchronization impulse, and arbitrary many secondary devices, the sinks, which wait for an impulse to be sent. Sending and receiving of said impulse is handled by the *CC1101* chip that handily provides a rising edge after sending or receiving a sync word, which is used to create a GPIO interrupt, see also Fig. 9.9.

Based on oscilloscope measurements, the dead time t_{dead} between two boards due to the *CC1101* chip is $25.94\,\mu\text{s} \pm 0.05\,\mu\text{s}$ (95 % confidence) and can therefore be partially corrected in software. The boards are programmed to start sampling when the GPIO interrupt is serviced by the micro-controller. Since the derivation of the minimum requirements showed that the drift of the *ESP32* clocks is neglectable, only the start of the sampling is synchronized, and there is no re-synchronization during the sampling. In order to validate this concept, two tests are performed: a synthetic one, where a step function from a signal generator is sampled, and an experimental one, measuring vibrations on a benchmark structure.

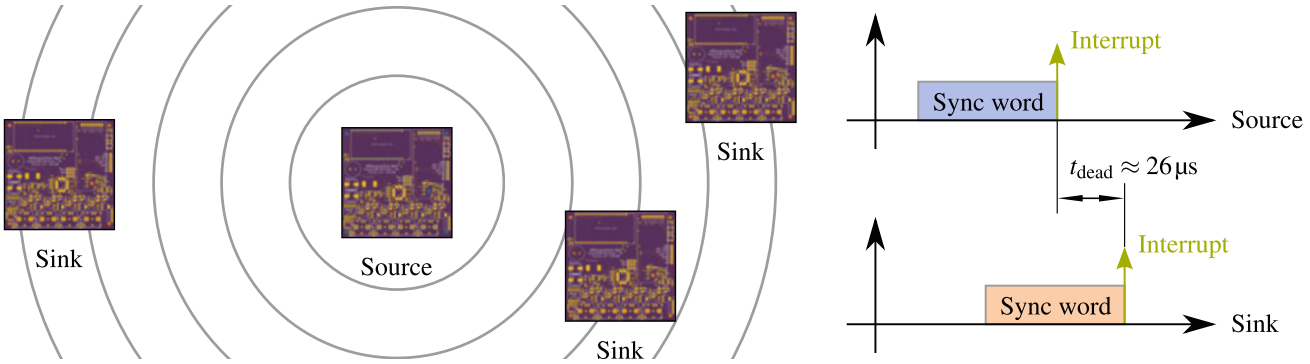


Fig. 9.9 Concept for wireless sample synchronization, including timing diagram

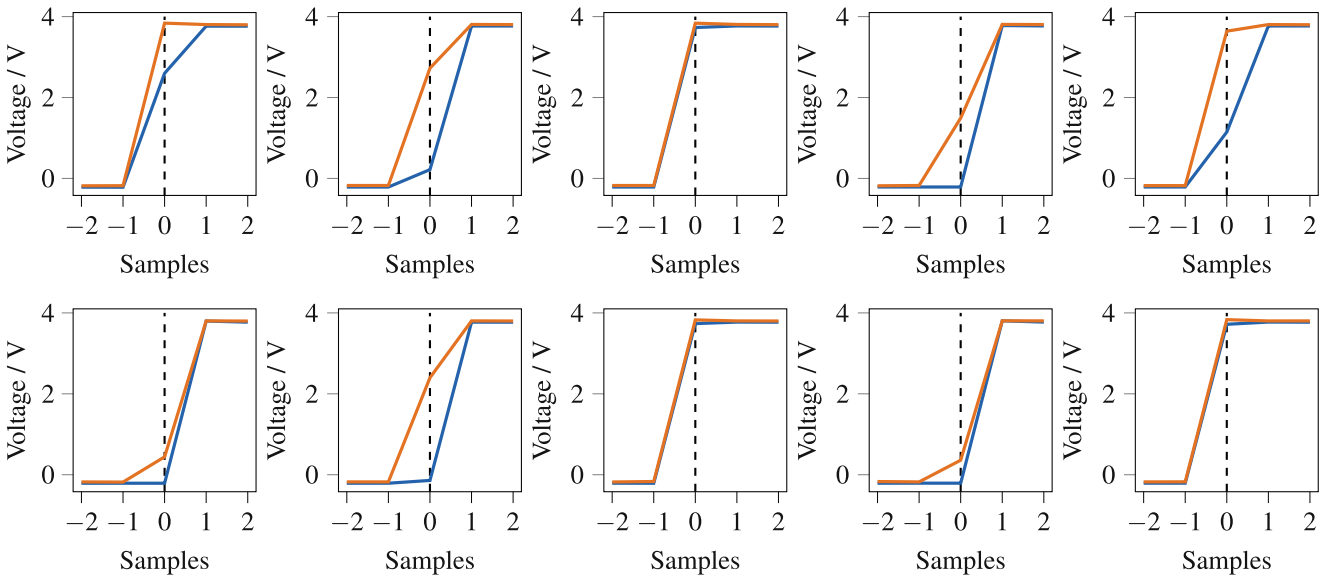


Fig. 9.10 Plot of synthetic validation series, measuring step functions with 50 ns rise time, plotted over samples such that first observable change in any signal happens at sample number 0, blue: board one and orange: board two, $f_S = 10\,\text{kHz}$

The signal used for the synthetic test is a step function with an amplitude of 4 V. Using an oscilloscope, the rise time of the signal is identified to be approximately 50 ns. One pair of signals (one source, one sink) is aligned such that the first change in any signal occurs at sample number 0. A total of 10 tests were performed with $f_S = 10\,\text{kHz}$, whose results are shown in Fig. 9.10.

At a first glance, the results look like the start time offset Δt_{ST} might be smaller than the rise time of 50 ns, when looking at plots 2, 5, and 7, where both signals have neither of the extreme values at sample number 0. But this is not the case and the reason is found in the oversampling, which is enabled to reduce noise. With the configuration used, each recorded sample is the average of 8 samples taken by the ADC. Those additional samples are acquired with an ADC internal sample rate of approximately 200 kHz. While it might be possible to calculate an estimate for the delay between the two signals from the afore mentioned or using statistical methods, this is not the goal here. Based on the time the ADC requires to internally gather 8 samples, roughly 35 µs, the delay between the two boards is well below 1 sample (i.e., 100 µs).

For the second validation, the vibrations of a benchmark structure, an L-shaped aluminum beam welded to a plate, are measured. One impact hammer and 3 one-axis acceleration sensors attached to the top of the L-beam are used. The signals are split using tee connectors, so both boards receive the identical analog signal, see Fig. 9.11.

In order to determine the performance of the wireless sample synchronization, the cross-correlation between the identical analog signals captured by the two wirelessly synchronized boards is calculated. For comparison reasons, the auto-correlation of one board is shown in the plots in Fig. 9.12 as well. The correlations are shown in dependence of the offset τ in samples, with the highest amplitude normalized to one.

All cross-correlations have their maximum at $\tau = 0$ but appear to be slightly shifted toward the right. The goal is to only get a rough idea of the expectable delay for now. Just as with the synthetic test, the delay can be said to be well below one sample or 100 µs, but it most likely does not meet the set requirement yet.

9.5 Conclusion

In the beginning, the minimum requirements for experimental vibration measurements in an academic context were derived. These requirements could mostly be met by the developed and described acquisition system. The proposed synchronization approach using the *Texas Instruments CC1101* chip works as intended and is able to synchronize two or more boards with less than 100 µs of delay. Nonetheless, further fine-tuning of the data acquisition software and validation techniques is required.

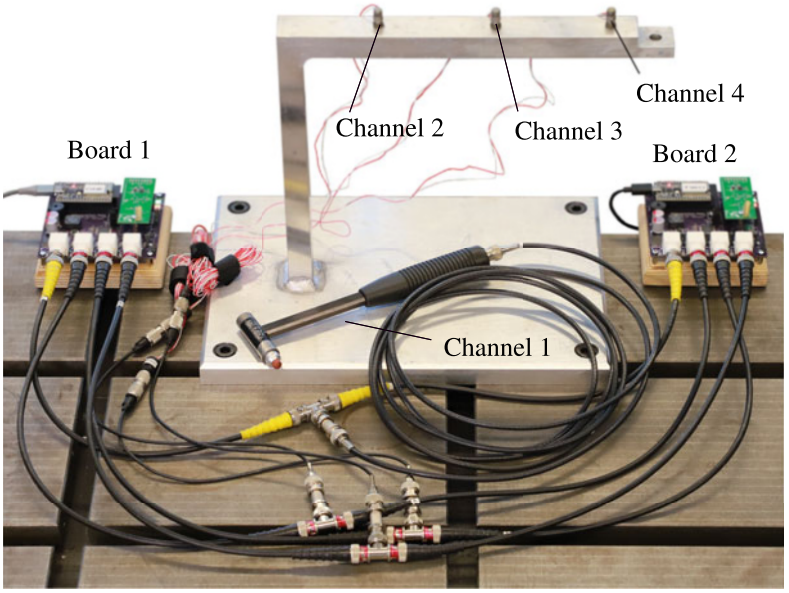


Fig. 9.11 Experimental setup for validation: one impact hammer as excitation and 3 one-axis acceleration sensors attached to the benchmark structure, signals are split using BNC cables and fed into both acquisition boards

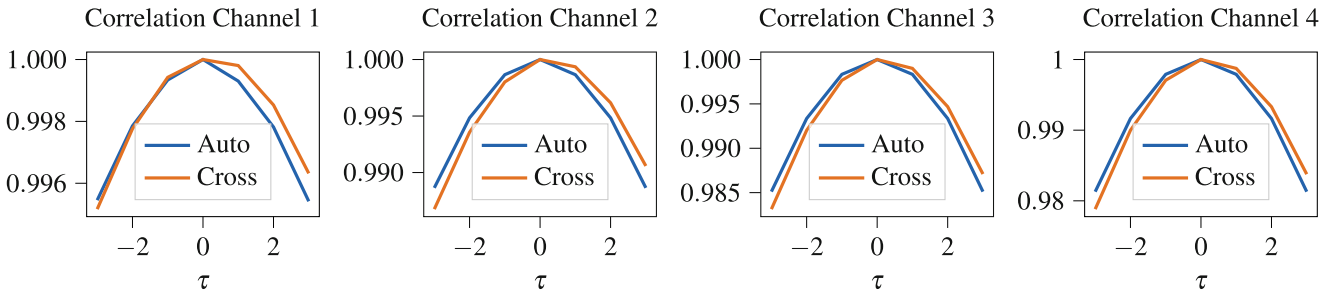


Fig. 9.12 Auto- and cross-correlation of experimental validation for each channel. Auto-correlation calculated from signals of board 1, cross-correlation between boards 1 and 2, highest amplitude normalized to 1, offset τ in samples with $1 \cdot \tau = 100\,\mu s$

With the methods used so far, an exact determination of the delay between two boards is not possible. This inhibits both the identification of the exact delay and the variance. While this approach might not be as precise as commercial products, it still enables measurements where a synchronization by cable is not possible and external signals, like GPS, cannot be used. As the schematics and software are made publicly available, this enables insight into the working principles and allows everyone to expand upon this basis.

References

1. León, F.P.: Messtechnik: Grundlagen, Methoden und Anwendungen. Springer Vieweg Berlin, Heidelberg (2019)
2. Brandt, A.: Noise and Vibration Analysis: Signal Analysis and Experimental Procedures, 1st edn. Wiley, Chichester (2011)
3. Trainotti, F., Berninger, T., Rixen, D.: Using Laser Vibrometry for Precise FRF Measurements in Experimental Substructuring (2020)
4. Géradin, M., Rixen, D.J.: Mechanical Vibrations: Theory and Application to Structural Dynamics, 3rd edn. Wiley, Chichester (2014)
5. Open Modal Project. pyFRF (v0.40) (2019)
6. Bregar, T., Mahmoudi, A.E., Kodrič, M., Ocepek, D., Trainotti, F., Pogačar, M., Göldeli, M., Čepon, G., Boltežar, M., Rixen, D.J.: pyFBS: a python package for frequency based substructuring (v0.2.9). J. Open Source Softw. **7**(69), 3399 (2022)
7. Guillaume, P., Verboven, P., Vanlanduit, S., Van der Auweraer, H., Peeters, B.: A poly-reference implementation of the least-squares complex frequency-domain estimator. Proc. IMAC **21**, 01 (2003)
8. Ewins, D.J.: Modal Testing: Theory, Practice and Application. Wiley, Chichester (2000)