**COMPUTER RESEARCH AND MODELING**
**2023 VOL. 15 NO. 1 P. 141–155**
DOI: 10.20537/2076-7633-2023-15-1-141-155

Ки&М

**ENGINEERING AND TELECOMMUNICATIONS**

UDC: 621.391

# Coherent constant delay transceiver for a synchronous fiber optic network

**A. A. Deev[a], A. A. Kalshchikov**

Chair of Fundamentals of Radio Engineering, National Research University "MPEI",
13, build. E, Krasnokazarmennaya st., Moscow, 111250, Russia

E-mail: [a] t-90.deevav@yandex.ru

*Received 01.11.2022.*
*Accepted for publication 23.12.2022.*

This paper proposes the implementation of a coherent transceiver with a constant delay and the ability to select any clock frequency grid used for clocking peripheral DACs and ADCs, tasks of device synchronization and data transmission. The choice of the required clock frequency grid directly affects the data transfer rate in the network, however, it allows one to flexibly configure the network for the tasks of transmitting clock signals and subnanosecond generation of sync signals on all devices in the network. A method for increasing the synchronization accuracy to tenths of nanoseconds by using digital phase detectors and a Phase Locked Loop (PLL) system on the slave device is proposed. The use of high-speed fiber-optic communication lines (FOCL) for synchronization tasks allows simultaneously exchanging control commands and signaling data. To simplify and reduce the cost of devices of a synchronous network of transceivers, it is proposed to use a clock signal restored from a data transmission line to filter phase noise and form a frequency grid in the PLL system for heterodyne signals and clock peripheral devices, including DAC and ADC. The results of multiple synchronization tests in the proposed synchronous network are presented.

Keywords: FPGA, FOCL, PLL, subnanosecond time synchronization, single time scale, two-way synchronization method, clock recovery, diversity in-phase clock network, generation of ADC and DAC clock signals

Citation: *Computer Research and Modeling*, 2023, vol. 15, no. 1, pp. 141–155.

**КОМПЬЮТЕРНЫЕ ИССЛЕДОВАНИЯ**
**И МОДЕЛИРОВАНИЕ 2023 Т. 15 № 1 С. 141–155**
DOI: 10.20537/2076-7633-2023-15-1-141-155

Ки&М

**ИНЖИНИРИНГ И ТЕЛЕКОММУНИКАЦИИ**

УДК: 621.391

# Когерентный приемопередатчик с постоянной задержкой для синхронной оптоволоконной сети

**А. А. Деев[a], А. А. Кальщиков**

Кафедра основ радиотехники Национального исследовательского института «МЭИ»,
Россия, 111250, г. Москва, ул. Красноказарменная, д. 13, корп. Е

E-mail: [a] t-90.deevav@yandex.ru

*Получено 01.11.2022.*
*Принято к публикации 23.12.2022.*

В статье предлагается реализация когерентного приемопередатчика с постоянной задержкой и возможностью свободно варьируемой сетки тактовых частот, используемой для тактирования периферийных ЦАП и АЦП, задач синхронизации устройств и передачи данных. Выбор необходимой сетки тактовых частот напрямую влияет на скорость передачи данных в сети, однако позволяет гибко настроить сеть для передачи тактовых сигналов и генерации синхроимпульсов с субнаносекундной точностью на всех устройствах в сети. Предложен метод повышения точности синхронизации до десятых долей наносекунды за счет использования цифровых фазовых детекторов и системы фазовой автоподстройки частоты (ФАПЧ) на ведомом устройстве. Использование высокоскоростных волоконно-оптических линий связи (ВОЛС) для задач синхронизации шкал времени, позволяет параллельно синхронизации производить обмен командами управления и сигнальными данными. Для упрощения и удешевления устройств синхронной сети приемопередатчиков предлагается использовать тактовый сигнал, восстановленный из сериализованных данных, и прошедший фильтрацию фазовых шумов, для формирования в системе ФАПЧ тактовых сигналов периферийных устройств, таких как ЦАП и АЦП, а также сигналов гетеродина. Представлены результаты многократных тестов синхронизации в предложенной синхронной сети.

Ключевые слова: ПЛИС, ВОЛС, ФАПЧ, субнаносекундная синхронизация, единая шкала времени, двухпутевой метод синхронизации, восстановление тактового сигнала, сеть разнесенных синфазных тактовых частот, формирование сигналов тактирования АЦП и ЦАП

## 1. Introduction

Under modern conditions of technical development, the problem of accurate time synchronization is relevant [Idrees et al., 2020]. For example, in a diversity radar station, a synchronization error between positions of 10 ns can lead to a detection error in a range of at least 30 m [Svetlichny, Degtyarev, 2019]. Without accurate time synchronization, the joint operation of radio telescopes and radars is not possible. Scientific experiments that require synchronous data collection at remote network nodes [Rizzi et al., 2016] (for example, the Large Hadron Collider at CERN) are not possible either. Similarly, without accurate time synchronization it is impossible to solve navigation problems [Perov, Kharisov, 2005]. It is very important to synchronize time in SDH/SONET networks that use time division multiplexing in data transmission [Weiss et al., 1996].

The formation of a cable network for the distribution of clock pulses and a network for distributing the reference frequency is the classical solution to the problem of synchronization. In addition to a significant number of cable lines used, installation complexity, and increased power consumption, there is a need to produce “measured” cables that provide equal delay for sync signals and equal phase shift for the reference signal at each network device. Periodically such systems need a calibration.

The use of a two-way synchronization method [Nagibin, Shchenin, 2020] through a duplex fiber optic communication line can significantly reduce the number of lines used to synchronize transceiver modules of a diversity synchronous network and ensure regular information and logical data exchange.

The main difference between a synchronous network (Fig. 1, *b*) of data transmission [Aweya, 2012; Lipinski et al., 2011], relative to a non-synchronous network (Fig. 1, *a*) is the presence of only one master clock signal generator. It generates a reference frequency for the operation of the entire network and performs its transmission over communication lines in the form of serialized data to slave and intermediate devices.

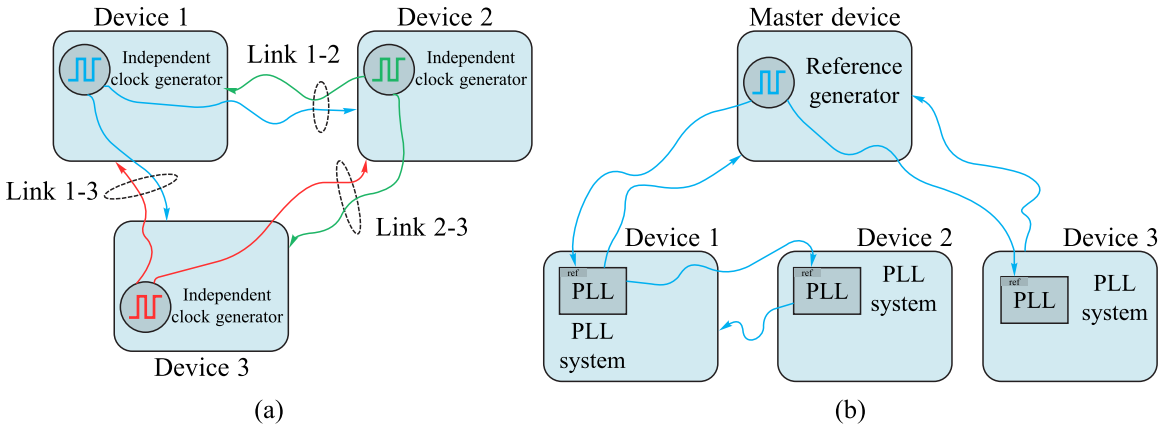


Figure 1. Comparison of non-synchronous (a) and synchronous data networks (b)

The classical two-way time synchronization method underlies the PTP protocol [IEEE Standard for a Precision Clock Synchronization, 2020], which offers a simple algorithm for generating and exchanging timestamps to determine the message transmission time and time offset between synchronized devices. The order of sending is shown in Figure 2. Formulas (1a) and (1b) are used to calculate the time offset:

1. The master sends a Sync message and simultaneously saves the sending time $t_1$;
2. FollowUp message sends after Sync and contains timestamp $t_1$. Receiving the Sync message, the slave generates a time stamp $t_2$;

3. The slave generates a DelayReq message and simultaneously generates a time stamp $t_3$;
4. The master receives the DelayReq message and simultaneously generates the $t_4$ time stamp;
5. The master sends $t_4$ timestamp to the slave in a DelayResp message;

$$t_2 = t_1 + T_{off} + T_{del},$$
$$t_4 = t_3 + T_{off} + T_{del},$$
$$T_{off} = \frac{(t_2 - t_1) - (t_4 - t_3)}{2}, \quad (1a)$$
$$T_{del} = \frac{(t_2 - t_1) + (t_4 - t_3)}{2}. \quad (1b)$$

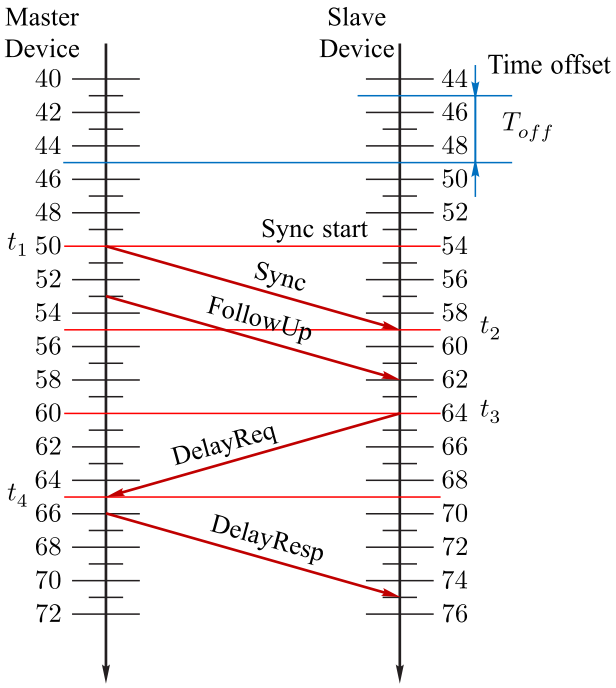


Figure 2. The process of exchanging messages between devices using the classic two-way synchronization method

To implement the two-way synchronization method, solutions based on programmable logic integrated circuits (FPGAs) are most often used. It is important to pay attention to three critical conditions that must be met for successful synchronization:

1. All timestamps must be generated in hardware, at the moment the message arrives at the device, and not at the moment it is processed by the processor;
2. The transmission time of a message in the link must be constant and known, but not necessarily equal to the reception and transmission [Lévesque, Tipper, 2015; Alhashmi et al., 2022];
3. The data transmission network must be synchronous, and, consequently, the time counters, which are synchronized, must be clocked with the same clock signal.

## 2. Implementation of an arbitrary clock frequency grid in a synchronous network

The main problem that can be encountered when using synchronous FPGA-based networks for the needs of synchronizing transceivers is a very limited set of clock frequency grids. For Synchronous Ethernet 1G, 10G, 40G standards, these are 125 MHz, 156.25 MHz and, depending on the implementation, 312.5 MHz. Solutions that are not tied to industrial Ethernet standards, providing support for a two-way synchronization method based on FPGAs, are not presented by manufacturers and are not available in open sources.

Such a limited set of frequencies (and integer derivatives of it) often cannot satisfy the requirements for generating and processing signal data. These frequencies may also not be applicable for the operation of certain DAC and ADC microcircuits. In the case of generation of sync signals in the Ethernet frequency grid, translation of clock signals from the clock domain in which the clock signal was created to the clock domain in which it will be used is required. In addition, there is a requirement for the coherence of the clock signal of the synchronous network to the clock signal in which the sync pulse will be used. The features described above significantly complicate the development of the final device and lead to a decrease in the final accuracy of the formation of sync pulses.

For use in a synchronous network with support for subnanosecond synchronization, the Native PHY FPGA transceiver unit must provide second and third of the conditions described in the introduction. Thus, the transceiver unit must provide:

- Recovery of the clock frequency from the channel data stream by the CDR (Clock Data Recovery) block;
- Provide a constant time for receiving and transmitting data packets.

To fulfill these conditions, the Transceiver unit (also known as PHY) must translate the restored clock signal into the user logic, and at the same time must not have blocks in the receiving and transmitting paths that introduce a non-deterministic time delay, as shown in Figure 3.

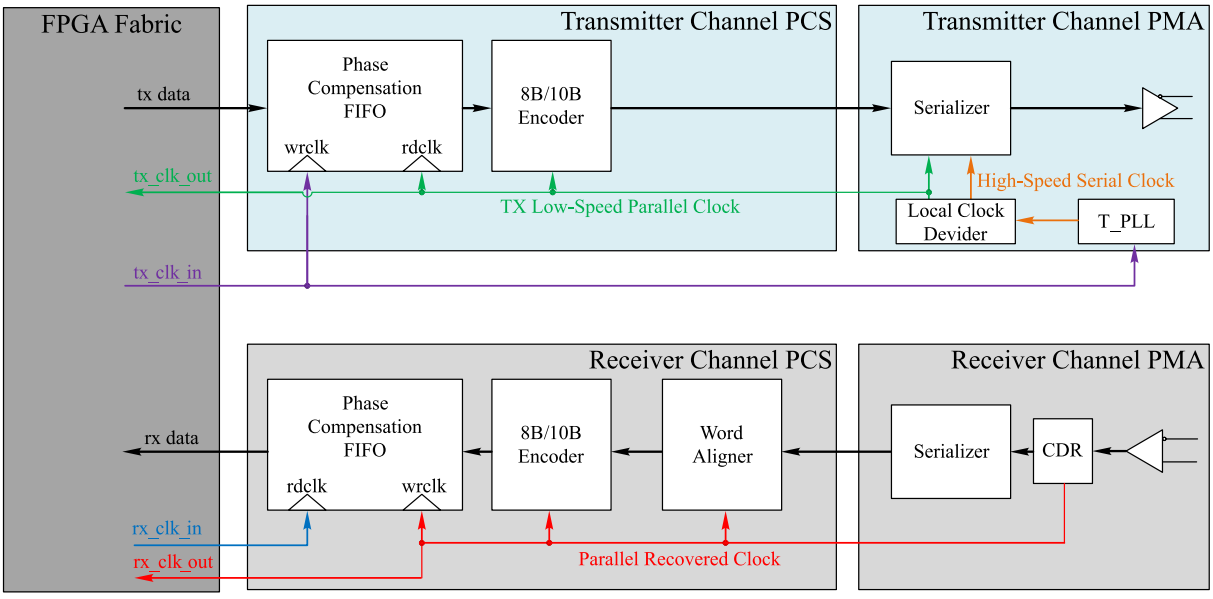


Figure 3. FPGA Transceiver unit configuration for constant transmit and receive delays

Separate instantiation of the PHY block allows one to specify the frequency used for data transmission in a wide range. From hundreds of MHz to tens of GHz. For example, if it needs to get a frequency grid of 120 MHz, the PHY channel rate must be set to 1200 Mbps. Due to the use of 8B/10B coding, the useful channel speed will be 8/10 of the channel — 960 Mbps. In this way, it is possible to calculate the useful channel rate, taking into account which coding is used in a particular case (8B/10B or 64B/66B).

Hardware timestamp fixing requires an implementation of the custom MAC (Media Acess Control) module. This module will implement both standard functions, such as checking and generating CRC, separating the preamble, and forming accurate timestamps of the moments of sending and receiving packets related to two-way synchronization. It should be noted that the developed MAC has support for standard network protocols and operates with ordinary Ethernet frames, including for synchronization tasks. However, due to the use of arbitrary clock frequencies in the PHY and the non-standard algorithm for establishing a connection in the channel via MAC, such a device cannot be directly connected to a standard network.

Due to the large variability of frequencies that can be used in the proposed implementation of the network, for time counting and synchronization tasks, a custom time counting module was implemented with the possibility of fine adjustment through a register map. Its main difference is that time is counted in cycles of the reference oscillator of the master device. Thus, there is no problem of binding the local time scale to the absolute one, in contrast to the time counter solutions offered by Altera and Xilinx. There is also no problem of the need to adjust the time counter [Deev, Kalshchikov, 2022] due to the finite accuracy of the counter.

The algorithm of the custom MAC in terms of the formation of timestamps is quite simple. The send timestamp is generated only when sending packets related to synchronization (Sync, DelayReq). Filtering is performed by hardware in the MAC, by delaying all packets for a certain number of words and determining whether it belongs to the synchronization algorithm. If a packet belongs to the synchronization algorithm, the timestamp and the packet id are fixed and given to the user logic. The packet ID allows a slow processor system to establish a clear correspondence between the timestamp and the packet, which is important in the case when the master synchronizes several slaves at the same time.

The timestamp of the received packet is generated for absolutely all incoming packets, since at the time the first byte of the preamble is received, it is impossible to determine whether the packet is related to the synchronization algorithm. Having received a sufficient number of bytes of the packet for membership analysis, the timestamp is either fixed or discarded. By analogy with sending, the packet id is fixed by hardware, which is also subsequently transmitted to the processor system.

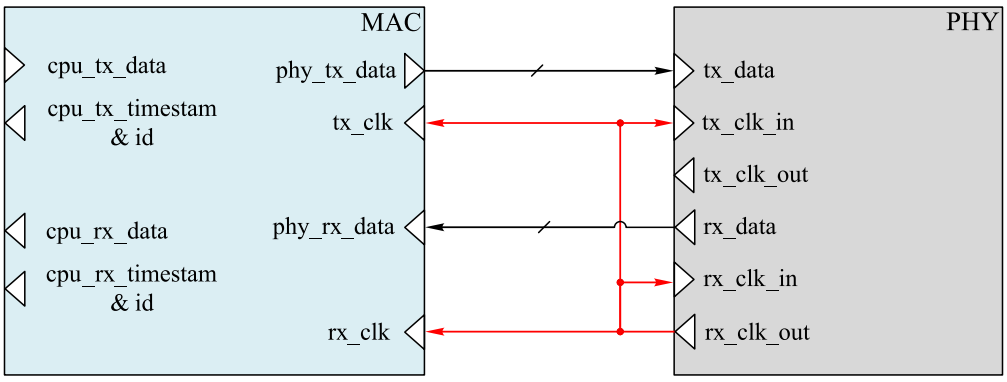


Figure 4. Standard MAC and PHY connection scheme, provides basic timing accuracy for the two-way synchronization method

The standard MAC and PHY connection scheme for implementing the standard two-way synchronization method is shown in Figure 4. It is important to note that this clock-switching scheme is applicable only for the slave device. In the case of the master, it has its own reference oscillator.

## 3. Improving the synchronization accuracy of the two-way method

Using a synchronous network allows one not to worry about frequency instability on the device, because the frequency drift of the reference oscillator on the master device is distributed to the entire network. This improves the accuracy of time scale synchronization on each of the devices in the synchronous network. The basic implementation of the two-way method in a synchronous data network, allows one to achieve synchronization accuracy of ±1 period of the frequency used for data transmission. For example, for 1 Gbps Ethernet, using a data rate of 125 MHz, the timing accuracy will be ±8 ns, respectively.

In a synchronous data network, synchronization accuracy can be improved by measuring the phase difference between the clock signal that clocks the time counter and the clock signals that receive and send messages in the data link layer. Thus, each timestamp ($t_1$, $t_2$, $t_3$, $t_4$) can be supplemented with phase information ($t_{1\varphi}$, $t_{2\varphi}$, $t_{3\varphi}$, and $t_{4\varphi}$, respectively).

To obtain information about the phase, it is necessary to modify the standard MAC and PHY switching circuit shown in Figure 4. Two digital phase detectors DMTD (Dual Mixer Time Difference) [Moreira et al., 2010] are installed to measure the phases between the write and read clock signals phase compensation FIFO in RX and TX paths of the FPGA transceiver unit.

The main advantage of DMTD is the ability to fully implement it on the FPGA logic. The DMTD module always measures the phase difference between clk1 and clk2, but not vice versa. DMTD also

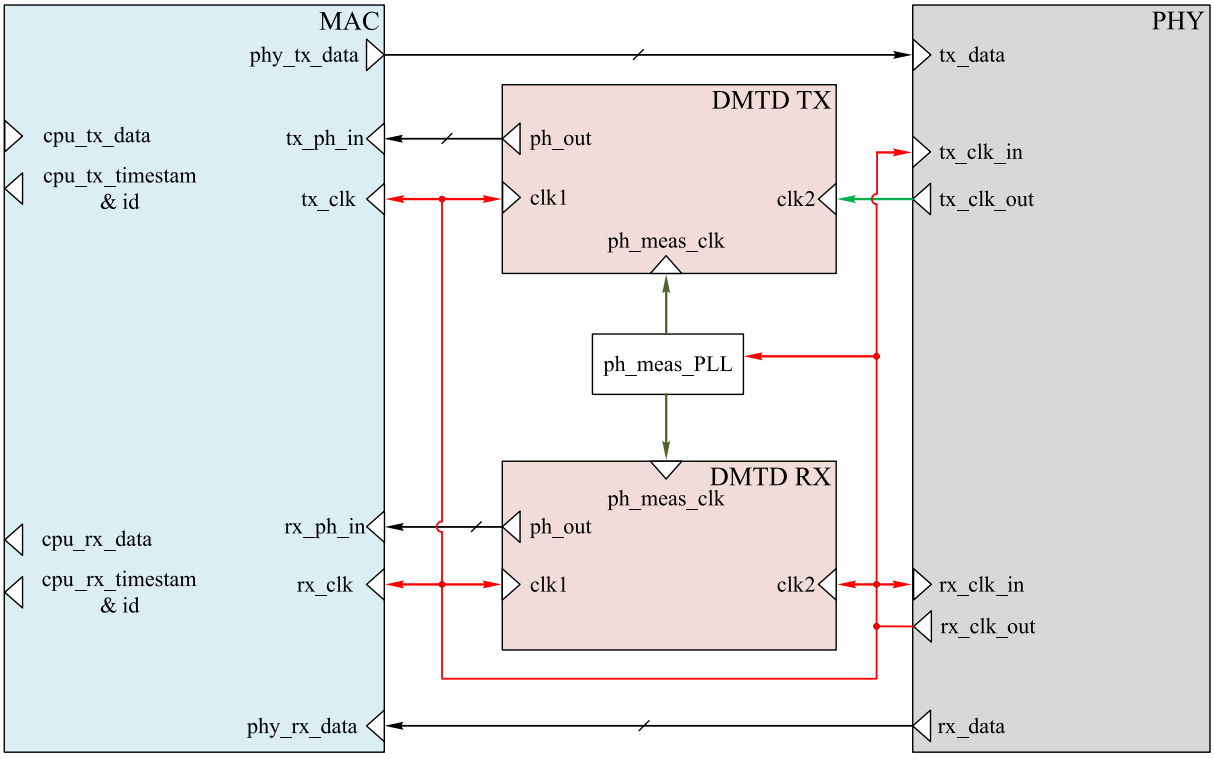


Figure 5. A modified MAC and PHY connection scheme, provides subnanosecond timing accuracy for the two-way synchronization method

requires the generation of a coherent clock signal with a small detuning up to 1 MHz. To form such a frequency, a PLL is usually used, also located on the FPGA chip. The amount of offset affects the potentially achievable phase measurement accuracy, but increases the duration of the measurement. The modified scheme is shown in Figure 5.

The above diagram of MAC and PHY connection is relevant only for the slave device. In the case of a master device, the clock signals must be connected differently, based on the analysis of the waveforms in Figure 6.

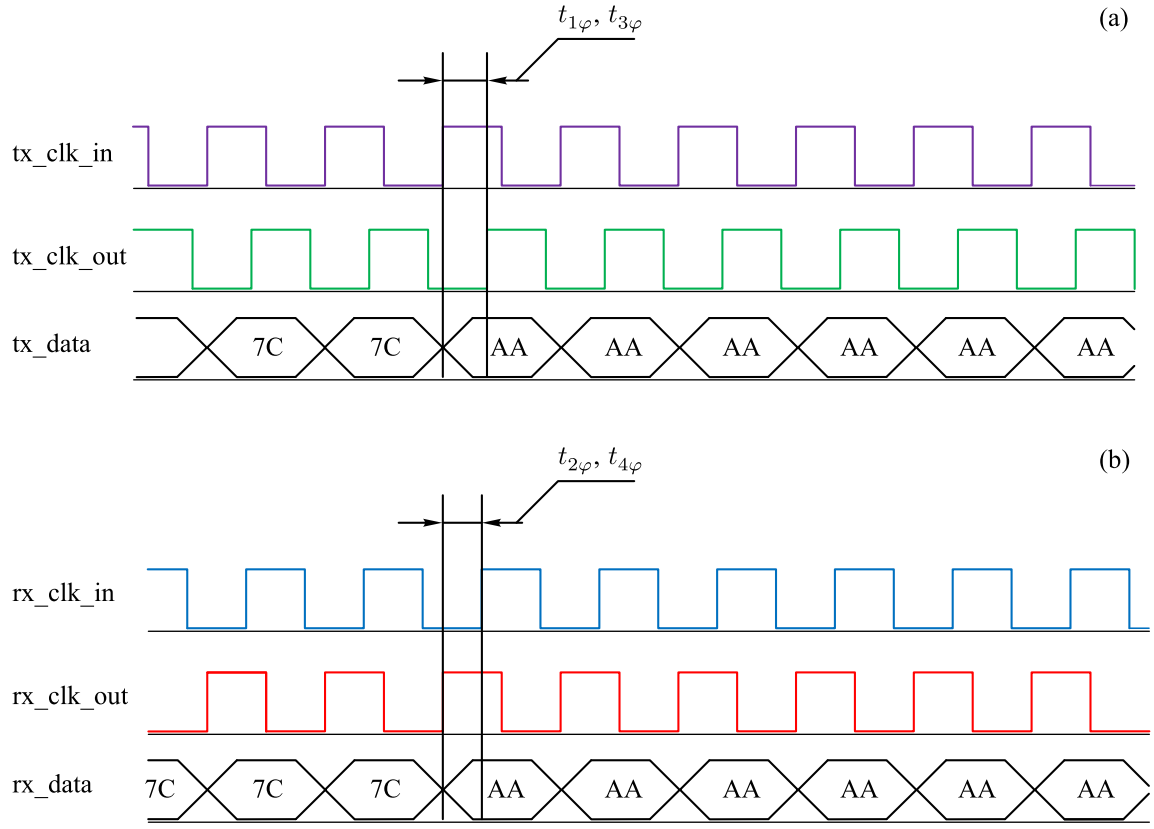


Figure 6. To account for the phase difference in the time stamps of receive and transmission. For rx packet timestamps, the phase is fixed with a minus sign. For tx packet timestamps, the phase value is fixed with a plus sign

A critical point in DMTD placement is maintaining equal electrical path lengths for measured clock signals to phase compensation FIFO and to DMTD inside the FPGA chip.

After successfully measuring the phase difference between the clock signals, this information should be correctly interpreted to complete the timestamps $t_1$–$t_4$:

- For timestamps for sending messages, the measured phase difference should be added to the fixed timestamp, since the actual moment of packet transmission to the FPGA transceiver logic occurs somewhat later than according to the clock signal tx_clk_in — Figure 6, *a*.

- For timestamps for receiving messages, the measured phase difference should be subtracted from the fixed timestamp, since the actual moment the packet leaves the FPGA transceiver logic occurs somewhat earlier than the clock signal rx_clk_in — Figure Figure 6, *b*.

Then the formulas for determining the message transmission time (1a) and the time offset between devices (1b) will be written as follows:

$$T_{off} = \frac{((t_2 + t_{2\varphi}) - (t_1 - t_{1\varphi})) - ((t_4 + t_{4\varphi}) - (t_3 - t_{3\varphi}))}{2}, \tag{2a}$$

$$T_{del} = \frac{((t_2 + t_{2\varphi}) - (t_1 - t_{1\varphi})) + ((t_4 + t_{4\varphi}) - (t_3 - t_{3\varphi}))}{2}. \tag{2b}$$

The refined time offset estimate between devices must be compensated in 2 steps:

1. Correction of the slave time counter according to $T_{off}$;
2. Correction of the phase of the clock signal using the PLL (alignment of the clock signal of the slave to the master).

## 4. Clock generation method

According to the scheme shown in Figure 1, the clock signal is restored on each slave device. Thus, as noted in Section 3, the implementation of the subnanosecond synchronization method is achieved using a synchronous data transmission network and a PLL system on each of the slave devices. It allows one to organize a network of distributed coherent in-phase clock frequencies that can be used for clocking peripheral devices such as DACs, ADCs, including generating local oscillator signals [Wu et al., 2021].

To evaluate the applicability of such clock signals, an experiment was carried out, the scheme of which is shown in Figure 7. The frequency of 125 MHz, used in Gigabit Ethernet, was chosen as the clock frequency.

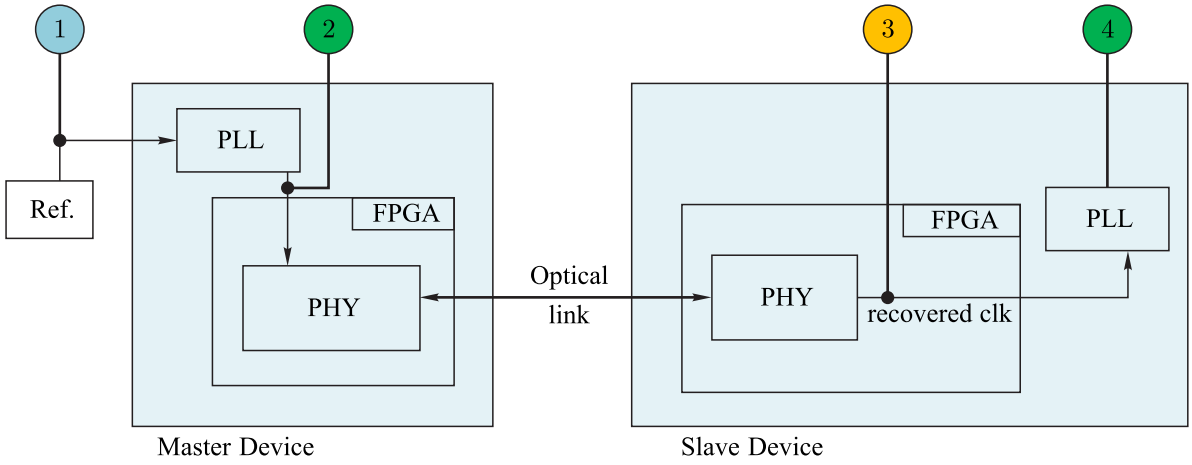


Figure 7. Simplified block diagram of the experiment to determine the phase noise of the clock signal, restored from the data

Points where phase noise measurements were taken:

- Reference oscillator output;
- External PLL output;
- The clock signal recovered from the data on the slave device;
- External PLL output — restored and filtered clock signal.

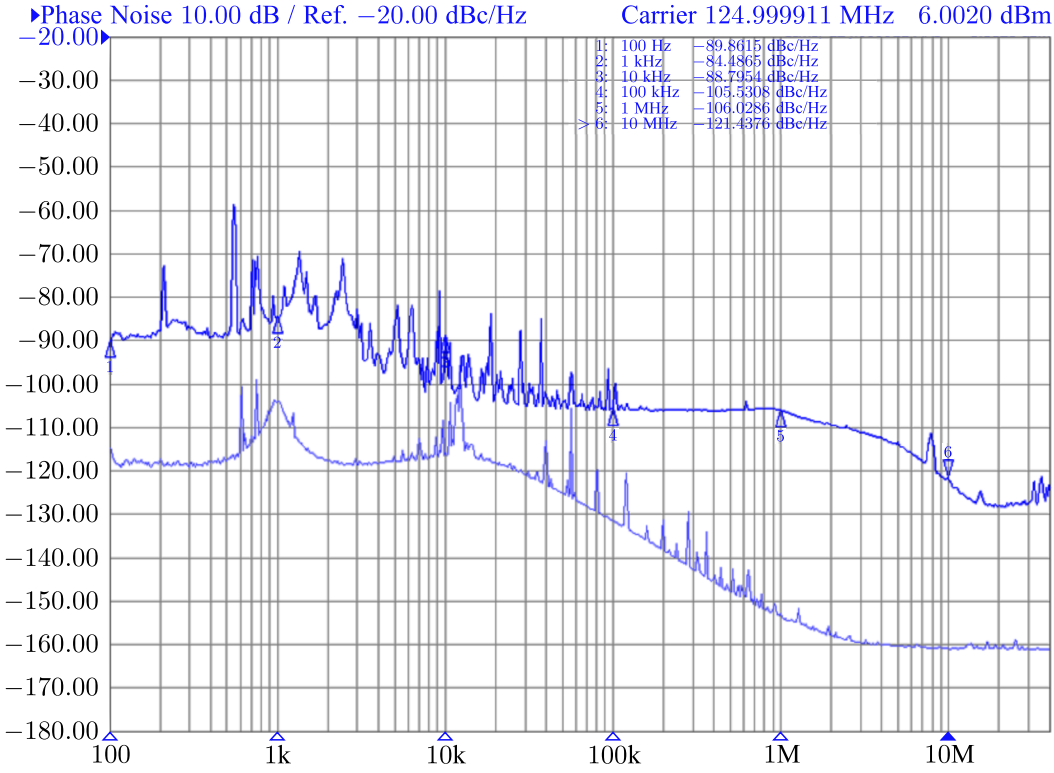


Figure 8. Graph of the spectral density of phase noise in sections 2 (bottom) and 3 (top). There is a significant increase in the phase noise level of the recovered clock signal at the CDR output

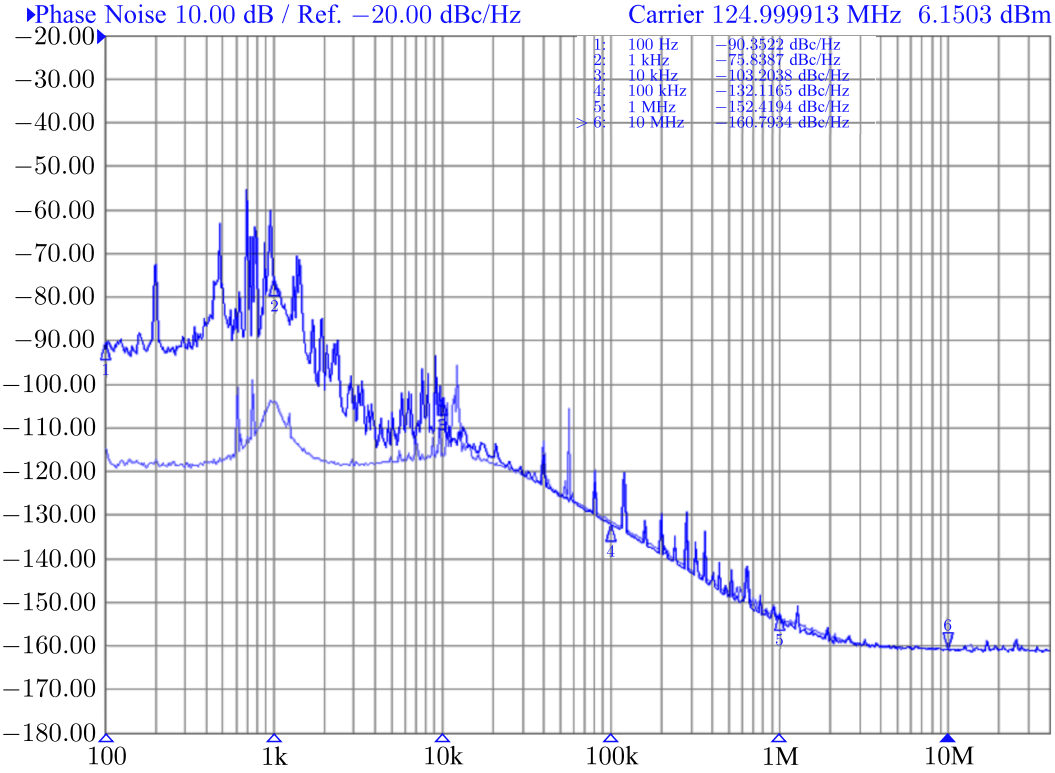


Figure 9. Graph of the spectral density of phase noise in sections 2 (bottom) and 4 (top). There is a significant suppression of phase noise at the output of the PLL in the region of large frequency offsets

Figures 8 and 9 show the results of phase noise measurements at points 2, 3 and 4.

The phase noise plot of the recovered clock shows a significant increase in phase noise. To obtain the root-mean-square value of jitter, it is required to calculate the integral over the spectral density of phase noise.

- RMS of clock jitter on the master;
- RMS jitter of the clock signal reconstructed from the data on the slave;
- RMS of the jitter of the clock signal on the slave, passed through the PLL system for phase noise suppression.

Based on formula (3) for calculating the maximum achievable signal-to-noise ratio (SNR) depending on the value [Texas Instruments, Jitter vs SNR] and the corresponding graph shown in Figure 9, we can conclude that the direct use of the restored clock signal will reduce the potentially achievable SNR by 39 dB.

$$SNR_J = -20 \cdot \log[2\pi \cdot f_s \cdot T_j], \text{ dB}. \quad (3)$$

However, using the loop filter of the PLL system, it is possible to filter out a significant part of the phase noise in the detuning region of more than 10 kHz and achieve a significant reduction in jitter to a level of = 0.21 ps. This value gives the maximum achievable SNR is 75.65 dB, which is only 0.42 dB worse than result for the master clock signal on the master device.

## 5. Results

To determine the accuracy and stability of the synchronization of the proposed coherent constant delay transceiver for a synchronous fiber optic network, a test bench was assembled, a simplified block diagram of which is shown in Figure 10. A fiber optic cable about 140 m long connected the devices. The accuracy was determined by generating on the master and slave synchronization signal devices PPS (Pulse per Second) — which is formed at the moment of switching the seconds digit of synchronized time counters. The timing error was defined as the time delay between these signals (oscilloscope channels 1 and 2). The accuracy of the phase alignment between the clock signals of the time counter on the slave and master devices, respectively, was also evaluated (oscilloscope channels 3 and 4).

Figures 11 and 12 show master (channel 1) and slave (channel 2) PPS waveforms, as well as master (channel 3) and slave (channel 4) sys_clk clock signals. Figure 11 shows the result of synchronization before applying the phase compensation of the clock signal on the slave, and Figure 12, respectively, after.

The above oscillograms clearly demonstrate the increase in the synchronization accuracy of the classical two-path method by calculating and compensating the phase relationships between clock signals.

The above oscillograms clearly demonstrate the increase in the synchronization accuracy of the classical two-path method by calculating and compensating the phase relationships between clock signals.

A series of experiments was carried out in 500 synchronization cycles, the results of which are shown in Figures 13 and 14. Figure 13 shows the synchronization error of signals “1 second” from the synchronization cycle number. Figure 14 shows a histogram of the synchronization error calculated on the basis of 500 measurements; the solid line marks the normal distribution law with estimated parameters using the maximum likelihood method.

A series of experiments was carried out in 500 synchronization cycles, during which the following results were obtained:

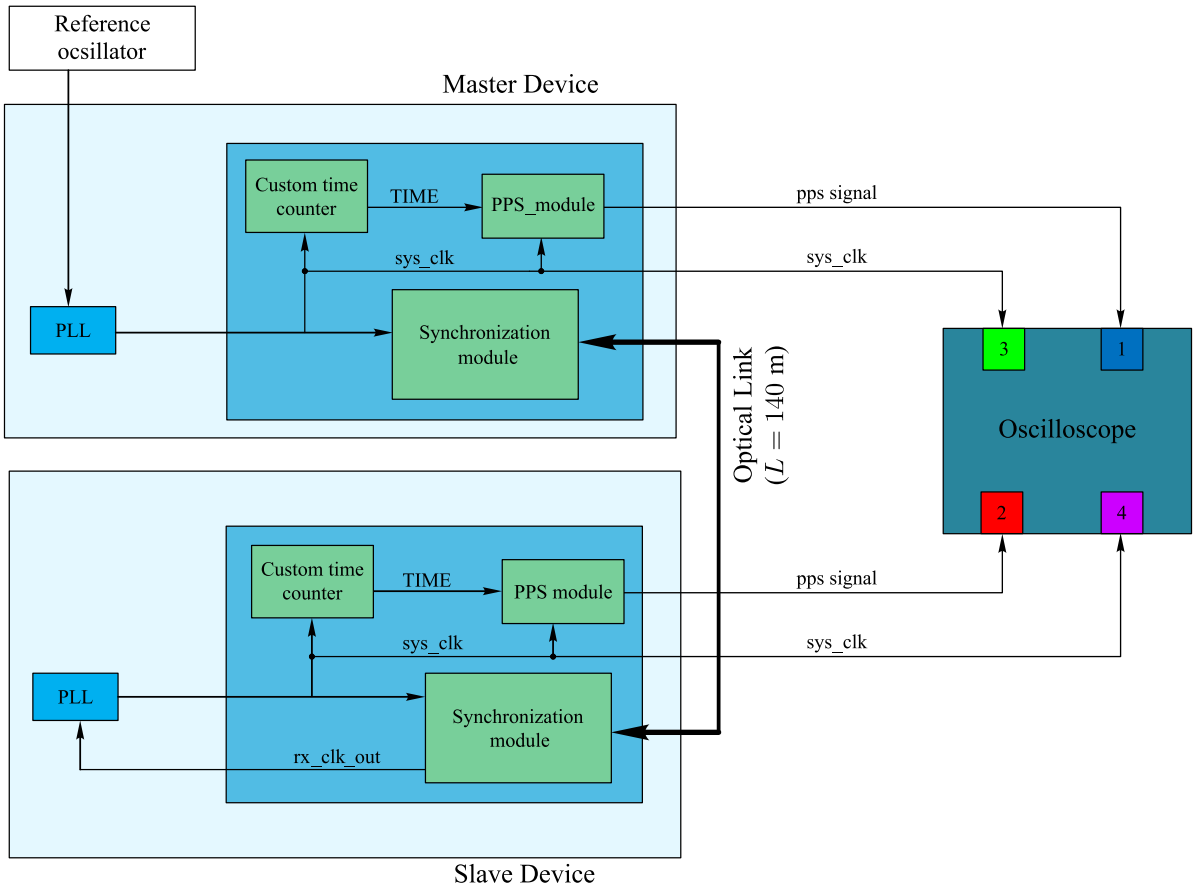


Figure 10. Simplified block diagram of the stand for estimating synchronization accuracy

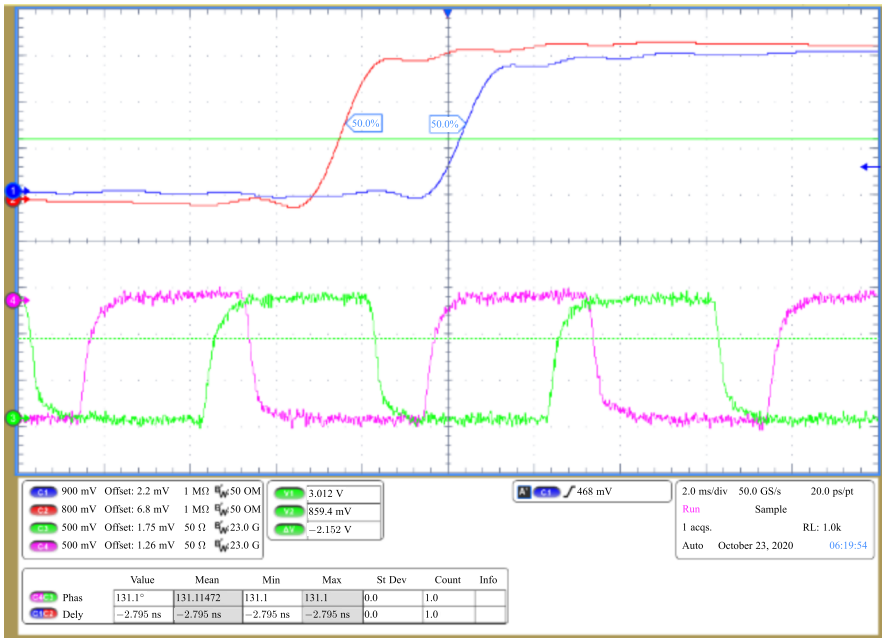


Figure 11. Synchronization result before phase compensation. There is a significant timing error, about 3 ns, as well as a significant phase difference between the device clocks

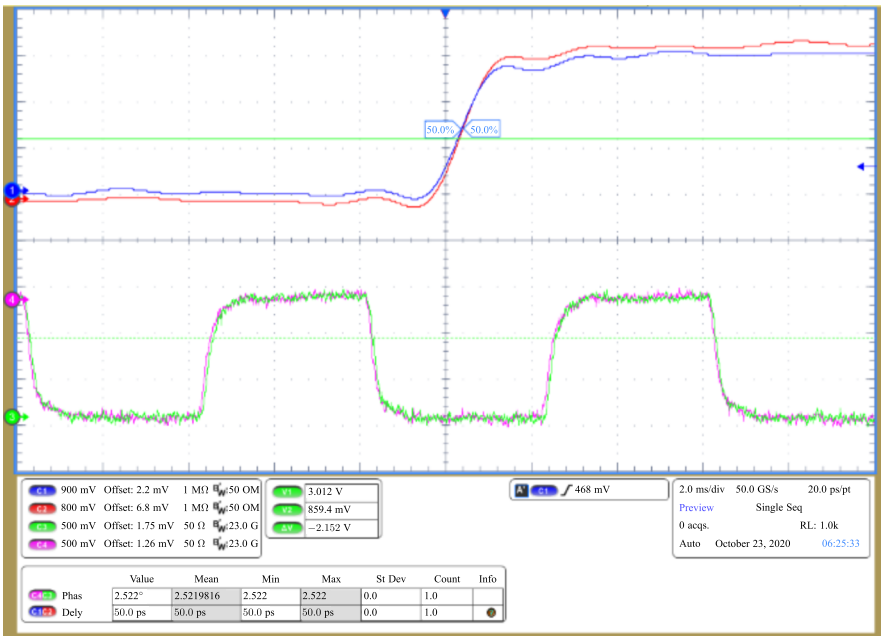


Figure 12. Synchronization result after phase compensation. There is precise synchronization of the PPS signals, as well as precise alignment of the clock signals of the devices

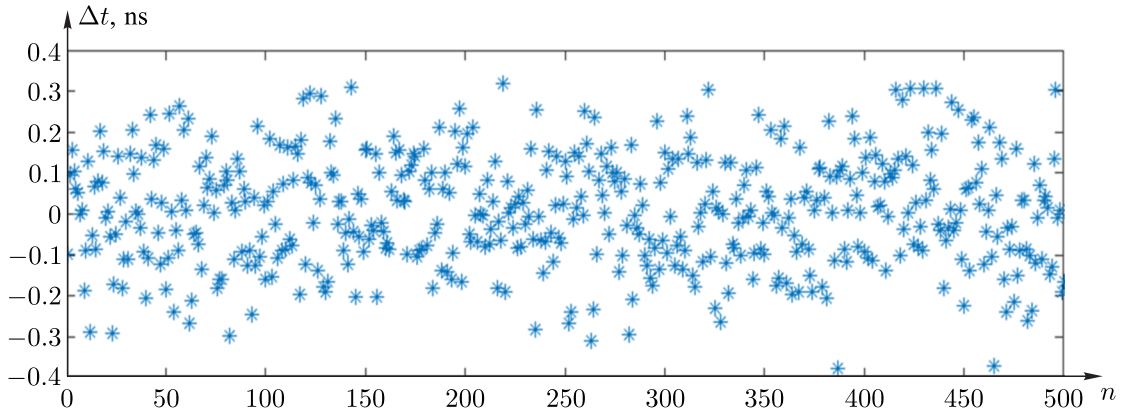


Figure 13. Synchronization error against synchronization cycle number

- $M_{\Delta t}$ = 0.16 ps — average error value;
- $\sigma_{\Delta t}$ = 125 ps — standard deviation of synchronization error;
- $T_{sync} \leqslant 4$ s — time spent on one synchronization cycle.

The results are given for the case of the transceiver operating at a frequency of 120 MHz, and the corresponding channel rate of 1200 Mbps. Regardless of the speed of data exchange, the condition of synchronism of the data transmission network in which synchronization of the time scales of devices is carried out remains unchanged.

Formula (4) was derived to estimate the dependence of the final synchronization accuracy on the accuracy of measuring the phase between the signals clocking the time counters.

$$\sigma^2_{\Delta t} = \sigma^2_{\varphi_{\text{meas}}} + \sigma^2_{T_{\text{set}}} + \sigma^2_{\Delta T_{osc}} \tag{4}$$

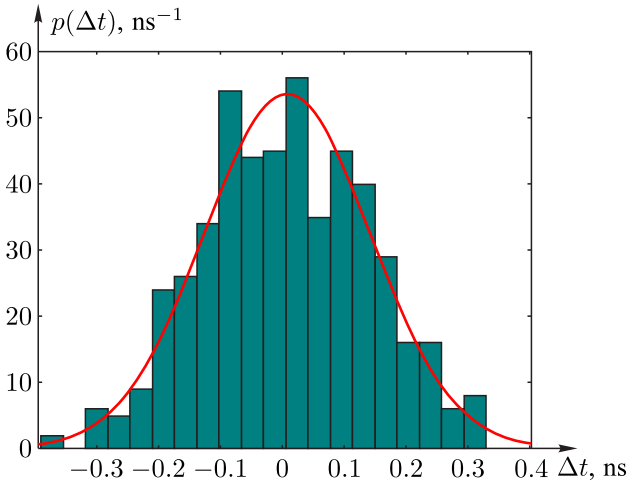


Figure 14. Distribution law of synchronization error

where $\sigma^2_{\varphi_{\text{meas}}} = \sigma^2_{\varphi_\Sigma} - \sigma^2_{\varphi_{osc}}$;

- $\sigma^2_{\varphi_\Sigma}$ is the summary dispersion of the phase estimate;
- $\sigma^2_{\varphi_{osc}}$ is oscilloscope phase estimate dispersion;
- $\sigma^2_{T_{\text{set}}} = \frac{\Delta\varphi^2_{\text{PLL}}}{12}$ — phase correction dispersion caused by a discrete phase correction step.

The proposed formula can be used when choosing a phase detector and/or PLL for the reconstructed signal to achieve the required synchronization accuracy between devices.

The reliability of the formula for estimating the synchronization accuracy was verified by experimentally obtaining the phase response, taking into account the measurement error of the oscilloscope and the error introduced by the discreteness of the phase correction.

## 6. Conclusion

A method for implementing a coherent transceiver with a constant delay for a synchronous fiber optic network is proposed. The main advantage of the proposed implementation is the arbitrary frequency of the transceiver and network, respectively. This solution allows one to flexibly adjust the network of transceivers not only for data transmission and subnanosecond synchronization, but also to organize a network of distributed coherent in-phase clock frequencies, which can significantly simplify the architecture of a distributed network of transceivers.

The systematic synchronization error of the devices was about −0.16 ps, and the variance was about 125 ps for 500 synchronization cycles. Achieving such synchronization accuracy is possible due to the use of phase detectors in the formation of timestamps and adjustment of the clock signal on the slave device.

An analysis of the phase noise of the restored clock signal was carried out to assess its applicability for clocking peripheral devices (ADC/DAC). In most cases, the use of the recovered clock signal is possible with ADCs with thermal noise floor down to −75 dB. The use of narrower PLLs can also increase this parameter.

It is important to note that the implementation of the transceivers was carried out on the basis of budget debug boards with FPGAs, which implies a fairly low cost of such devices.

## References

*Нагибин Е. В., Щенин Н. Н.* Методы сравнения шкал времени // Радионавигация и время. — 2020. — Т. 6, № 14. — С. 29–39.
*Nagibin E. V., Shchenin N. N.* Metody sravneniya shkal vremeni [Methods for comparing time scales] // Radionavigaciya i vremya [Radio navigation and time]. — 2020. — Vol. 6, No. 14 — P. 29–39 (in Russian).

*Перов А. И., Харисов В. Н.* ГЛОНАСС. Принципы построения и функционирования. — М.: Радиотехник, 2005. — 688 с.
*Perov A. I., Kharisov V. N.* GLONASS. Principy postroeniya i funkcionirovaniya [GLONASS: Principles of construction and operation]. — Moscow: Radiotekhnik, 2005. — 688 p. (in Russian).

*Светличный Ю. А., Дягтерев П. А.* Синхронизация и передача данных в радиотехнических системах с территориально распределенными сегментами // Доклады Томского государственного университета систем управления и радиоэлектроники. — 2019. — Т. 22, № 3. — С. 7–12.
*Svetlichny Yu. A., Degtyarev P. A.* Sinhronizaciya i peredacha dannyh v radiotekhnicheskih sistemah s territorial'no raspredelennymi segmentami [Synchronization and data transmission in radio engineering systems with geographically distributed segments] // Doklady Tomskogo gosudarstvennogo universiteta sistem upravleniya i radioelektroniki [Reports of the Tomsk State University of Control Systems and Radio Electronics]. — 2019. — Vol. 22, No. 3. — P. 7–12 (in Russian).

*Alhashmi N. et al.* Path Asymmetry Reconstruction via Deep Learning // IEEE 21st Mediterranean Electrotechnical Conference (MELECON). — 2022. — P. 1171–1176.

*Aweya J.* Emerging Applications of Synchronous Ethernet in Telecommunication Networks // in IEEE Circuits and Systems Magazine. — 2012. — Vol. 12, No. 2. — P. 56–72.

*Deev A. A., Kalshchikov A. A.* Features of high-precision sync-signals formation for distributed systems based on a synchronous Ethernet network // 4th International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE). — 2022. — P. 1–4.

*Idrees Z. et al.* IEEE 1588 for Clock Synchronization in Industrial IoT and Related Applications: A Review on Contributing Technologies, Protocols and Enhancement Methodologies // IEEE Access. — 2020. — Vol. 8. — P. 155660–155678.

IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems // IEEE Std 1588-2019 (Revision of IEEE Std. 1588-2008). — 2020. — P. 1–499.

Jitter vs SNR for ADCs TIPL 4704, Texas Instruments. — URL: https://training.ti.com/sites/default/files/docs/TIPL20470420-20Jitter20vs20SNR.pdf (accessed: 10.11.2022).

*Lévesque M., Tipper D.* Improving the PTP synchronization accuracy under asymmetric delay conditions // 2015 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control, and Communication (ISPCS). — 2015. — P. 88–93.

*Lipinski M. et al.* White rabbit: a PTP application for robust sub-nanosecond synchronization // International IEEE Symposium o Precision Clock Synchronization for Measurement Control and Communication (ISPCS), 2011.

*Moreira P. et al.* Digital dual mixer time difference for sub-nanosecond time synchronization in Ethernet // 2010 IEEE International Frequency Control Symposium. — 2010. — P. 449–453.

*Rizzi M. et al.* White rabbit clock characteristics // 2016 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control, and Communication (ISPCS). — 2016. — P. 1–6.

*Weiss M. A. et al.* Two-way time and frequency transfer in SONET // Proceedings of 1996 IEEE International Frequency Control Symposium. — 1996. — P. 1163–1168.

*Wu C.-Y. et al.* A Bit-Interleaved Sigma-Delta-over-Fiber Fronthaul Network for Frequency-Synchronous Distributed Antenna Systems // APPLIED SCIENCES-BASEL. — 2021. — Vol. 11. — No. 23.