General-Purpose Data Streaming FPGA TDC Synchronized by SerDes-Based Clock Synchronization Technique

Ryotaro Honda, Masahiro Ikeno, Che-Sheng Lin, and Masayoshi Shoji

***Abstract*— This study proposes a clock synchronization protocol using the functionalities of IDELAYE2 and IOSERDESE2 primitives of an AMD Xilinx field-programmable gate array (FPGA) to serve as a general-purpose data-streaming type time-to-digital converter (TDC) for particle and nuclear physics experiments. A clock synchronization protocol called local area common clock protocol (LACCP) was developed as the upper layer protocol of a proprietary link (MIKUMARI), which was defined prior to this work by a community of users from the experimental physics field in Japan. Clock synchronization is realized using a round-trip time measurement with the system clock period and a fine offset time estimation, which corresponds to the clock signal phase difference between the primary and secondary FPGAs. The fine offset measurement is based on information from the IDELAYE2 and ISERDESE2 primitives utilized as the physical layer of the MIKUMARI link. No extra component is used. The LACCP can be implemented in an FPGA using general IO pin pairs for serial transmission and reception. A streaming high-resolution TDC (Str-HRTDC) was developed based on a tapped-delay-line (TDL) built from CARRY4 primitives in the AMD Xilinx Kintex-7 FPGA. It continuously measures the timing with 19.5-ps intrinsic resolution in σ and provides unique timestamp information over 2.4 h by introducing the time frame structure defined and synchronized by LACCP. The clock synchronization accuracy and the timing resolution were evaluated by connecting four modules with optical fibers up to 100 m in length. No cable length dependence was confirmed. The obtained synchronization accuracy was approximately 300 ps. The timing resolution between two synchronized modules was 23.1 ps in σ.**

***Index Terms*— Clock synchronization, field-programmable gate array (FPGA), serializer–deserializer (SerDes), streaming readout, time-to-digital converter (TDC).**

I. Introduction

In PARTICLE and nuclear physics experiments, a fundamental approach to data collection involves selecting and storing events. This process typically relies on a combination of hardware and software triggers. While hardware triggers play a crucial role in reducing the amount of data transmitted from front-end electronics (FEE), the development of sophisticated, low-latency hardware triggers poses significant challenges. Moreover, FEE must have sufficient memory to buffer data until the decisions to record or discard them are made by the trigger. To address these challenges, triggerless data-streaming type data acquisition (DAQ) systems are being intensively investigated around the world.

We consider that a triggerless DAQ is well-suited for small- and medium-scale experiments performed in nuclear and hadron facilities in Japan, e.g., at the Japan Proton Accelerator Research Complex (J-PARC), the Research Center for Nuclear Physics (RCNP), RIKEN, and the Research Center for Accelerator and Radioisotope Science (RARIS). Each experiment employs a unique configuration tailored to its specific physical objectives, requiring different DAQ trigger logic. Moreover, the development and maintenance of a hardware trigger remain complex and resource-intensive.

To address these challenges, we are developing a general-purpose triggerless DAQ system as a part of the signal processing and data acquisition infrastructure (SPADI) alliance [1]. Our initial focus has been on a time-to-digital converter (TDC)-based DAQ system as event reconstruction using timing information is the minimum albeit the most essential function of a triggerless DAQ system. Thus, the development of a TDC module and a clock synchronization system became our first priority. A key requirement for the FEE is to ensure that it is sufficiently generic to accommodate a wide range of experimental setups. For example, the beam delivery method for the J-PARC hadron experimental facility [2] is slow extraction where a proton beam is slowly extracted from the J-PARC main ring over 2 s in a 4.2-s cycle. If the FEE is designed specifically to acquire data during beam extraction and the internal timestamp length of the FEE matches the J-PARC slow extraction cycle, this can cause complications for users at the cyclotron facility, where the beam is continuously supplied without extraction cycles. These users would need to reproduce the timing signals of the extraction cycle and provide them to the FEE. Although dedicated field-programmable gate array (FPGA) firmware could be developed for each experimental facility, we opted for a more versatile approach of developing a general-purpose FEE, which does not rely on accelerator timing signals, and

Received 23 August 2024; revised 30 October 2024 and 18 December 2024; accepted 10 February 2025. Date of publication 13 February 2025; date of current version 17 March 2025. This work was supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant 23H00126 and Grant 22H04940. *(Corresponding author: Ryotaro Honda.)*

Ryotaro Honda, Che-Sheng Lin, and Masayoshi Shoji are with the Institute of Particle and Nuclear Studies, High Energy Accelerator Research Organization, Tsukuba 305-0801, Japan (e-mail: rhonda@post.kek.jp; cslin@post.kek.jp; mshoji@post.kek.jp).

Masahiro Ikeno is with the Research Center for Nuclear Physics, Osaka University, Ibaraki, Osaka 567-0047, Japan (e-mail: ikeno@rcnp.osaka-u.ac.jp).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNS.2025.3541731.

Digital Object Identifier 10.1109/TNS.2025.3541731

making it adaptable to various facilities without requiring FPGA firmware redevelopment, thus reducing development costs. Our design also prioritized scalability and simplicity. The FEE should be capable of operating in a stand-alone mode, with just one FEE and a personal computer (PC) in the simplest system, as well as in setups involving up to several thousand FEEs.

With these requirements in mind, we developed a data streaming high-resolution TDC (Str-HRTDC) on a general-purpose logic module called a main electronics for network-oriented triggerless data acquisition system (AMANEQ) [3]. The Str-HRTDC is designed to achieve a timing resolution of 30 ps (σ) to meet the requirements of the J-PARC hadron experiment [4]. In addition, we developed a clock synchronization system based on the AMANEQ module. This system requires clock signal transmission with a jitter lower than the intrinsic resolution of the Str-HRTDC. Subnanosecond synchronization accuracy is also necessary for software-based timing coincidence during event reconstruction. In this article, we describe the clock synchronization protocol and Str-HRTDC.

II. Design of Clock Synchronization Protocol

The clock distribution network forms a tree structure comprising a clock-root module connected via optical fibers or electrical cables to multiple leaf modules. All FPGAs on leaf modules adjust their internal clock frequencies and timestamps to those of the root. This includes the FPGA linked to the root module and any FPGAs located on the same leaf board and eventual daughter cards. We aim to synchronize all of them using the same synchronization technique. In a previous study [3], we developed the clock signal distribution method called the MIKUMARI link, which provides the clock signal frequency synchronization and a communication protocol between two FPGAs. The MIKUMARI link protocol is based on clock-duty-cycle modulation [5], achieving low-jitter clock signal transmission with approximately 7 ps in σ, even when using mixed-mode-clock management (MMCM) [10] in the FPGA for clock recovery. In addition, as the MIKUMARI link has been implemented using AMD Xilinx IDELAYE2 and IOSERDESE2 primitives [6], it is well-suited for synchronizing FPGAs that are interconnected by general IOs. Consequently, the MIKUMARI link has been adopted as the link-layer protocol, and a clock synchronization protocol, the local area common clock protocol (LACCP), was developed.

The goal of LACCP is not to have a recovered clock signal phase aligned with the root clock signal on each leaf module but rather to synchronize the timestamp across all FPGAs with subnanosecond accuracy.

A. Clock Signal Distribution

The system clock signal on the clock-root module is transformed into a duty-cycle modulated clock signal, which is then distributed to leaf modules via the serial link in FPGA. The serial link speed is set to eight times the system clock frequency, and the duty cycle of the transmitted clock signal is modulated by ±12.5% [i.e., one unit interval (UI)] around 50% to transfer up to 1 bit of data per system clock period. To generate and capture this modulation pattern, the IOSERDES primitive is operated with a serialization factor of 8 and double-data-rate (DDR) mode. Herein, the clock signal that is four times faster than the system clock referred to as the sampling clock is employed to sample the serial data.

On leaf modules, the modulated clock signal is fed into a phase-locked loop (PLL) to reproduce both system and sampling clock signals. In a relay module, the modulated clock signal is regenerated and transmitted to further downstream modules. Here, the leaf nodes are only frequency-locked to the root system clock signal, but the phase difference between the root clock signal and the recovered clock signal at leaf nodes is arbitrary as it depends on physical media length and some other delays.

B. Timestamp

Each FPGA implements a 16-bit internal counter, called the heartbeat counter, which provides timestamps synchronous to the system clock. The system clock is 125 MHz, which is also well-suited for the implementation of the TDCs, as discussed later. The sampling clock signal operates at 500 MHz. A heartbeat pulse signal is generated by the carry-out bit of the heartbeat counter, which goes to a logic 1 when the heartbeat counter rolls over to 0x0000 and a logic 0 otherwise. The interval between two heartbeat pulses is called a heartbeat frame. With a 125-MHz system clock, the period of the heartbeat pulse is approximately 524 μs. Heartbeat frames are incrementally numbered using a 24-bit frame counter. By combining the 24-bit frame number with the 16-bit counter value, a user can derive a timestamp that spans 2.4 h (524 μs × 16 777 216) until it rolls over, which is sufficiently long for expected applications.

C. Synchronization Scheme

In the synchronization scheme, the node transmitting the system clock signal via the MIKUMARI link is designated as the primary node, while the node recovering the system clock is the secondary node. When multiple modules are connected in series, a relay module contains both primary and secondary nodes. The clock signals on the primary and secondary nodes are referred to as the reference and recovered clock signals, respectively. From the perspective of the secondary node, the clock signals in the primary node are always the reference clock signals, regardless of whether or not the module is the root module.

The clock synchronization protocol we developed is to synchronize the heartbeat timing and frame numbers of each leaf FPGA with those of the root FPGA. However, an offset originating from clock phase difference exists between heartbeat pulses on the primary and secondary nodes, as shown in Fig. 1. In Fig. 1, the secondary node outputs the heartbeat pulse earlier than that of the primary one. To achieve subnanosecond precision in timestamp alignment, the LACCP calculates a correction offset by precisely measuring link propagation delay. This correction offset is termed the fine offset. In Fig. 1, the offset value attains a negative value, and the signs are

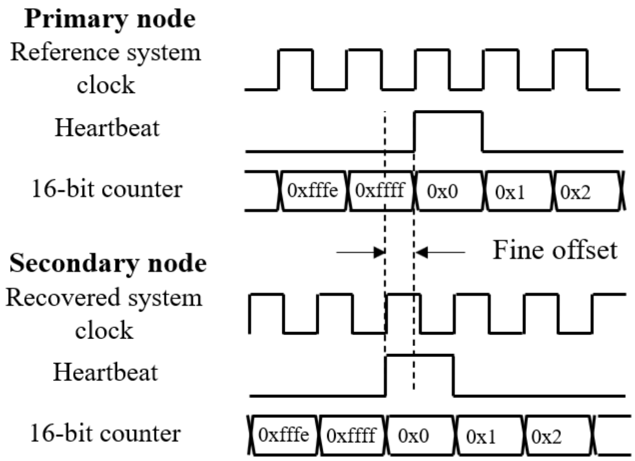


Fig. 1. Example of the clock phase relation and the fine offset between the primary and secondary nodes. The fine offset is defined as the time difference between the heartbeat pulses. Here, it attains a negative value to correct the timestamp in the secondary node.

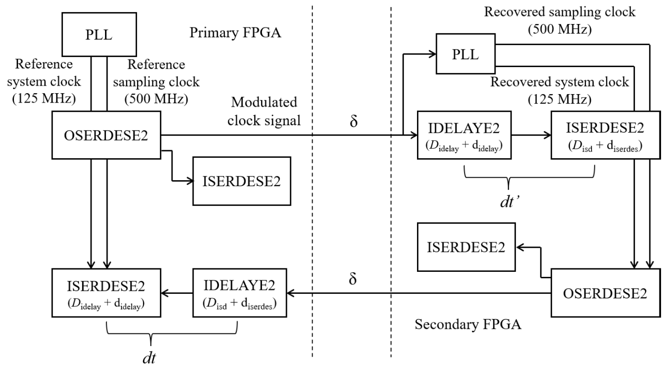

PLL
Primary FPGA
Reference system clock (125 MHz)
Reference sampling clock (500 MHz)
Modulated clock signal
OSERDESE2
ISERDESE2
δ
ISERDESE2 (D_{idelay} + d_{idelay})
IDELAYE2 (D_{isd} + d_{iserdes})
δ
dt
Recovered sampling clock (500 MHz)
PLL
Recovered system clock (125 MHz)
IDELAYE2 (D_{idelay} + d_{idelay})
ISERDESE2 (D_{isd} + d_{iserdes})
dt'
ISERDESE2
OSERDESE2
Secondary FPGA

Fig. 2. Block diagram of the physical layer of the MIKUMARI link. δ is the transmission delay of the signal between the OSERDES and the FPGA pad on the other side. D_{idelay} and D_{isd} are the constant delays that the IDELAY and the ISERDES have originally, respectively. d_{idelay} is the delay that is created as a result of adjusting the IDELAY settings. d_{iserdes} is the relative delay amount as a result of bitslip. dt and dt' are the total delays of the IDELAY and the ISERDES in the primary and secondary nodes, respectively.

determined by the time relationship between the heartbeat pulses.

The link propagation delay is broken into three parts.

1) *Fine Delay:* Less than 1 UI (1 ns in this work).
2) *Fractional Delay:* Expressed in a number of UI of the serial link: $M \times T_{\mathrm{ref}}/8$ with M in 0–7, where T_{ref} is the period of the system clock signal and is 8 ns in this work.
3) *Integer Clock Cycles:* Represented as $N \times T_{\mathrm{ref}}$.

The first and second parts originate from the serial link blocks of the FPGA and are measured during the link initialization process of the MIKUMARI protocol that ensures the correct capture of received serial data bits. Measurements are performed in each node independently, and thus, the fine offset cannot be calculated at this stage because measurement results cannot be transferred yet to the other side of the link. Once the link is established, the LACCP measures the round-trip time (T_{rt}) of the signal between the primary and secondary nodes to estimate the third part. A coarse offset is then calculated from T_{rt} and applied on the secondary node to align the heartbeat counter with its counterpart on the primary node. Finally, the primary node sends the measured results for the fine and fractional delays to the secondary node to estimate the fine offset.

D. MIKUMARI Link Initialization Process

We describe the MIKUMARI functions that are crucial for measuring the fine and fractional delays. The physical layer of the MIKUMARI link is shown in Fig. 2. During the first stage of the link initialization process, the protocol exchanges a bit pattern of 0b11110000 between the primary and secondary nodes. The MIKUMARI protocol uses this bit pattern to adjust the phase between the sampling clock signal and the incoming modulated clock signal. To provide the delay to the modulated clock signal, an IDELAYE2 [6] is used. The IDELAYE2 is a primitive block in AMD Xilinx FPGA that can delay a signal from an input pad by up to 2418 ps in 78-ps steps. The FPGA on each node examines the output bit pattern of the ISERDES while changing the delay amount of the IDELAY, identifying the range of delay values where the data bits embedded in the modulated clock signal can be correctly captured. The logic of the receiver then sets the delay of the IDELAYE2 block to the value where the edges of the sampling clock fall at the center of the incoming modulated signal bits. This delay value, termed d_{idelay}, is the fine delay.

Next, the phase between the modulated signal and the system clock signal is adjusted using the bitslip function of the ISERDESE2 primitive [6]. The bitslip function shifts by one bit the data deserialized by the ISERDESE2. In the MIKUMARI protocol, the receiver node performs bitslip operations until the expected synchronization pattern 0b11110000 is found at the output of its ISERDESE2. This process is equivalent to adjusting the phase between the incoming modulated signal and the system clock signal with 1/8 T_{ref} precision. However, given the complex structure of the ISERDESE2, there is no simple correspondence between the number of bitslip performed and the induced latency [6]. To solve this difficulty, we use the circuit shown in Fig. 2 to measure in situ the delay caused by bitslip operations on an ISERDESE2: the output of an OSERDESE2 block is fed back internally (using the OFB port) to the ISERDESE2 located in the same IOB. The latency of this chain is determined by measuring the time required for a bit pattern to propagate from the input of the OSERDESE2 to the output of the ISERDESE2. For example, if a bit pattern of 0xf8 is input to the OSERDESE2, and 0x1f appears from the ISERDESE2 after three system clock cycles, the total propagation delay of the OSERDESE2 and ISERDESE2 is 27/8 T_{ref}. We performed such measurements by varying the number of bitslips performed. In certain cases, we observed that performing an additional bitslip reduces the delay through the ISERDESE2. Since the correspondence between the number of bitslips and deserializer latency may differ among families of FPGAs, the MIKUMARI protocol requires that transmitters automatically measure these parameters at link initialization and store them in an internal memory table. The relative delay amount given by bitslip is termed d_{iserdes}, and it can attain a negative value.

E. Round-Trip Time Measurement

After the MIKUMARI link is established, the LACCP starts to measure the round-trip time T_{rt}. A pulse transmission function of the MIKUMARI protocol is used to measure T_{rt} and synchronize the heartbeat counter. This function transmits a pulse synchronized to the system clock with a fixed latency to the other side. The secondary node then sends a pulse to the primary node for measurement of T_{rt} while recording the value of the heartbeat counter at the time of transmission. Upon receiving the pulse, the primary node retransmits it without any delay, and the secondary node records its counter value again when the pulse returns. As there is no processing time for retransmission on the primary node, T_{rt} is the difference between the counter values recorded on the secondary node for transmission and reception. Subsequently, the secondary node requests the primary node to send the heartbeat pulse. The secondary node synchronizes its heartbeat counter by setting its initial count to $T_{\text{rt}}/2$ instead of zero when the heartbeat pulse is received. After this synchronization, the secondary requests the primary node to stop sending further heartbeat pulses and begins using its own generated pulses.

After that, the LACCP synchronizes the heartbeat frame number. The global frame numbers leaving the root module at "backbeat" timing, which is the midpoint between consecutive heartbeat pulses, are spread to FPGAs on leaf modules. As the half period of the heartbeat (approximately 262 μs) is sufficiently long compared to the distribution delay of frame number information from the root module to all leaf modules, it is guaranteed that no leaf module will generate the next heartbeat pulse before any other module has received the current frame number.

As mentioned above, the synchronization of frame numbers is based on the assumption that the half cycle of the heartbeat is longer than the transmission delay of the signal; thus, the LACCP will fail if the transmission delay from the root module to the leaf module furthest away exceeds the half cycle of the heartbeat. However, as 262 μs is equivalent to a fiber length of several tens of kilometers, it is not a concern for intended applications.

F. Fine Offset Estimation

The LACCP derives the fine offset using values of d_{idelay} and d_{iserdes}. The parameter δ is defined as the time required for a signal to travel from the OSERDES to the FPGA pad on the other side of the module. This time is the same for both outgoing and incoming signals, assuming that the modules are identical and the part-to-part skew of each part is neglected. Once the signal reaches the FPGA pad, it is deserialized and output from the ISERDES, requiring additional amount delays of, denoted as dt and dt', for the primary and secondary, respectively. Both delays are the sum of all the constant delays (D_{idelay} and D_{isd}) required when passing through the IDELAY and ISERDES and along with the delay values (d_{idelay} and d_{iserdes}) that depend on the phase difference between the incoming modulated clock signal and the system/sampling clock signals in each FPGA. On the secondary node, the phase difference between the incoming modulated clock and the system clock signals depends only on the FPGA family, board design, and FPGA firmware. The same phase difference is always taken for the same FPGA firmware. However, on the primary node, the value of dt changes depending on the phase difference between the reference clock signal and the returned modulated clock signal. Using δ and dt/dt', the time required for a signal leaving the OSERDES to be output from the ISERDES can be defined as follows for the going and returning signals from the primary node, respectively,

$$A = \delta + dt' \tag{1}$$

$$B = \delta + dt. \tag{2}$$

The phase difference between the reference clock signal and the recovered clock signal is 1/2 of the asymmetry between A and B. Therefore, the fine offset to be applied to the secondary-node timestamp is $(A-B)/2$ or $(dt'-dt)/2$. As the D_{idelay} and D_{isd} cancel each other out, the fine offset can be obtained from the measurements of d_{idelay} and d_{iserdes}. If the phase difference between two system clock signals is small, T_{rt} measured with T_{ref} precision is even because the propagation delay is identical in both directions. However, if the asymmetry between A and B exceeds the range that can be adjusted using the bitslip function of ISERDES, the measured T_{rt} is odd. In this case, the coarse offset to be applied to the heartbeat counter is $(T_{\text{rt}} - 1)/2$, and 1/2 T_{ref} is added to the obtained fine offset.

G. Fine Offset Accumulation

Clock synchronization is performed between endpoints of each link, where each secondary LACCP node adjusts the timestamp with respect to the upstream node following the point-to-point communication rule defined by the MIKUMARI link protocol. For example, consider three modules labeled M1, M2, and M3 connected in series, with M1 serving as the root module. First, the M2 clock is synchronized with M1. After the M1-M2 clock synchronization is established, the clock synchronization of M3 to M2 is performed. As a result, there are two local fine offsets: one for the M1-M2 link and another for the M2-M3 link. The overall fine offset of M3 with respect to M1 is obtained by accumulating the local fine offsets from both links. If the accumulated fine offset exceeds T_{ref}, the coarse offset value to the heartbeat counter is corrected by ± 1. Since the fine offset can take negative values, if the accumulated fine offset falls below -1 T_{ref}, a -1 correction is applied. In principle, there is no limit to the number of connection stages.

The benefit of the proposed approach for clock synchronization is that it can be performed automatically in FPGA logic when communication links are initialized. It does not require any external measurement equipment or embedded processor and software in the different nodes of the system.

III. Design of TDC

The design of the streaming TDC (Str-TDC) comprises the online data processing (ODP) block and the data

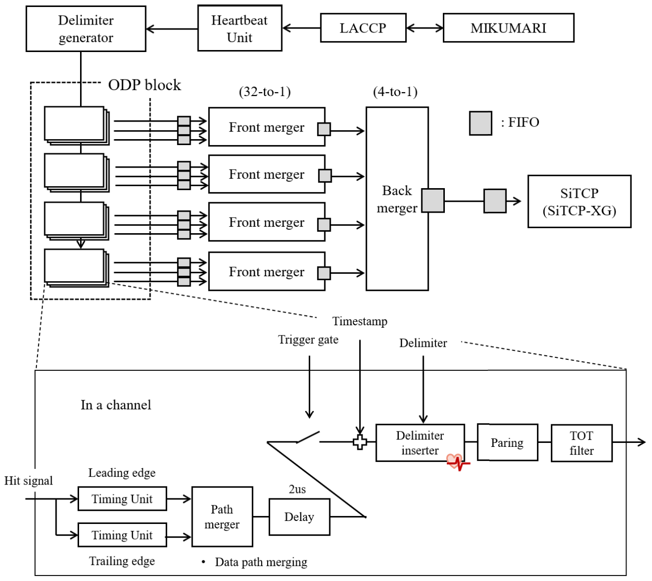


Fig. 3. Block diagram of Str-TDC. Timestamp originates from the heartbeat unit and is combined with the TDC fine timing before the delimiter inserter. The delimiter data are generated by the heartbeat signal.

merging (MGR) block, as shown in Fig. 3. The basic structure comes from our previous study [7] and is modified in this study. The ODP block has two timing units that measure the arrival time of the leading and trailing edges of the incoming signals. After the data paths are merged, TDC fine-timing data pass through the 2-μs delay buffer and wait for a trigger input. Although the Str-TDC operates in the triggerless mode by default, it supports an externally triggered mode if this is needed in certain experiments. This flexibility facilitates us in proposing a staging approach for DAQ updates for such experiments. The heartbeat counter value is incorporated before the heartbeat inserter. The delimiter inserter embeds special data called the heartbeat data as a boundary of the heartbeat frame. The frame number is embedded into the heartbeat delimiter. Thus, the TDC value is expressed as heartbeat counter value + TDC fine timing. By examining the delimiter data, users can track time over larger frames. Up to this point, the processing time is fixed.

Leading and trailing timing data are combined at the pairing unit, and time-over-threshold (TOT) is calculated and embedded in the leading edge data. To reduce the data rate, the trailing edge is discarded at this stage. The resulting data then pass through a TOT filter unit before being sent to the MGR block.

The MGR block collects data from multiple input channels and regenerates the heartbeat frame including them. It consists of two stages: front- and back-merger units. Channel-specific first-input–first-output (FIFO) memories are placed ahead of the front merger. The two-stage structure facilitates implementation across two FPGAs and improves buffering performance in the event of sudden data rate spikes. In the merger unit, the incoming data from each channel are output in order of arrival; however, if the heartbeat data are found, it stops reading from that channel. When delimiter data are found on all the channels, it generates new delimiter data and resumes the reading. The merger unit is designed to maintain a data throughput of approximately 8 Gb/s (64 bit × 125 MHz).

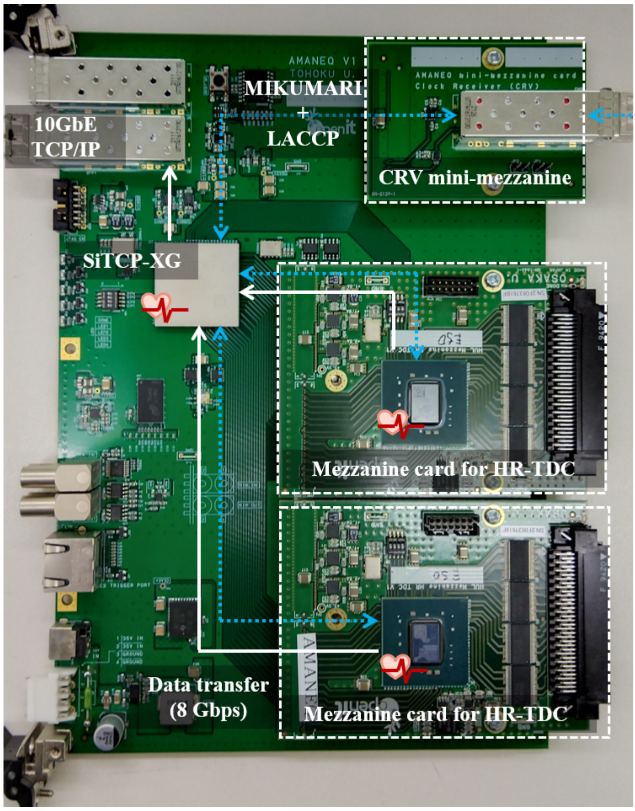


Fig. 4. Photograph of AMANEQ and mezzanine cards configured for Str-HRTDC. One CRV minimezzanine card and two mezzanine cards for high-resolution TDC are attached. White and blue (dotted) lines represent paths for the TDC data and clock synchronization, respectively. The heart mark denotes that the heartbeat unit exists within the FPGA.

Finally, the processed data are transferred to a PC via either silicon transmission control protocol (SiTCP) [8] or SiTCP-XG [9] cores, which are hardware implementations of TCP with gigabit and 10-Gb Ethernet, respectively.

IV. Implementation on AMANEQ

The hardware basis used for implementing the clock-root module, clock-hub modules, and Str-HRTDCs is a general-purpose module called AMANEQ [3]. This generic carrier board, shown in Fig. 4, can be equipped with two mezzanine cards and one minimezzanine card, allowing for the implementation of various specific functions. AMANEQ is equipped with Texas Instruments CDCE62002 jitter cleaner integrated circuit (IC), which offers superior jitter performance than that of MMCM in FPGA, and it is used for clock generation and recovery.

A. Clock Root and Clock Hub

Each of these modules requires one AMANEQ carrier, a clock data distributor (CDD) mezzanine card, and a clock receiver (CRV) minimezzanine card [3]. The CDD mezzanine card is a two-slot size daughter card featuring 16 small form factor pluggable (SFP) ports for optical fibers. The CRV minimezzanine card, shown in Fig. 4 (top right), has a single SFP port.

On these mezzanine cards, buffer ICs are placed between the SFP port and FPGA for logic translation between current mode logic (CML) and low-voltage differential signal (LVDS). This is because AMD Xilinx FPGA does not support CML signaling and cannot be connected directly to the SFP modules we use. For LVDS-to-CML translation, Micrel SY58603UMG ICs are used on both cards. For CML-to-LVDS translation, the CDD card uses SY58605UMG, and the CRV card uses PERICOM PI6C5922504. These ICs were selected based on the respective development times of those cards. Since propagation delays of SY58605UMG and PI6C5922504 differ by 210 ps, this leads to a systematic error of 105 ps in clock synchronization originating from the asymmetry in transmission delay. In addition, the part-to-part skew of buffer ICs introduces further asymmetry, namely 100, 135, and 200 ps for SY58603UMG, SY58605UMG, and PI6C5922504, respectively. As the propagation delay of each buffer IC could not be measured part by part, we treated the part-to-part skews as unknown variables in our study.

The clock-root module has only the primary node of the MIKUMARI protocol. It supports synchronization for up to 17 modules, utilizing the 16 ports of the CDD mezzanine card in addition to the single SFP port of the CRV minimezzanine card.

B. Str-HRTDC

For the Str-HRTDC, a mezzanine card equipped with an AMD Xilinx Kintex-7 FPGA (XC7K-160T-1) is used to outsource the tapped-delay-line (TDL)-based TDC from the main FPGA on the AMANEQ board. This card is labeled as the mezzanine card for HR-TDC in Fig. 4. The FPGA on the mezzanine card is connected to the main FPGA with 32 signal lines with the LVDS standard. To reduce power supply noise, the supply voltages for the FPGA are generated through a series regulator, Analog Devices ADP1741ACPZ-R7. The input signals are buffered by onsemi FIN1108MTDX. The number of input channels is 32.

The system clock is recovered using the CDCE62002 jitter cleaner IC placed on the AMANEQ carrier board. This recovered system clock is then forwarded through the main FPGA of the board to the FPGA of each mezzanine card via LVDS lines. Contrary to the carrier, the mezzanine card does not contain a jitter cleaner IC, and clock generation is performed internally within the local FPGA by an MMCM block.

The TDL comprising CARRY4 primitives [11] is formed in the timing unit. A total of 192 CARRY4 primitives are chained, which approximately corresponds to the size of one clock region. The O and CO outputs of each CARRY4 are alternately connected to a flip-flop (FF) in the order of O and CO following a method described in [12] to reduce zero-width bins. The initial part of the tapped-delay-line (TDL) is shown in Fig. 5. As the 26.2144-MHz clock signal for calibration is connected to the DI0 input of the first CARRY4 primitive, only for the first O output, CO is used instead. In TDL-based TDC in FPGAs, a phenomenon called "bubbles" is commonly observed due to the nonuniform propagation of TDL. For example, a rising edge propagation might result in the output 000101111. To mitigate this, the output of three adjacent FFs is OR'ed together. This operation reduces the number of effective taps to 64. The average effective tap delay increases to approximately 30 ps, which reduces time resolution; however, this still meets our design goals. As this TDC is designed to detect the rising edge of a signal, measuring the falling edge only requires inverting the bit pattern from the TDL. In other words, the rising and falling edge measurements share the same TDL, and then, the falling measurement is branched by performing bit inversion at this point.

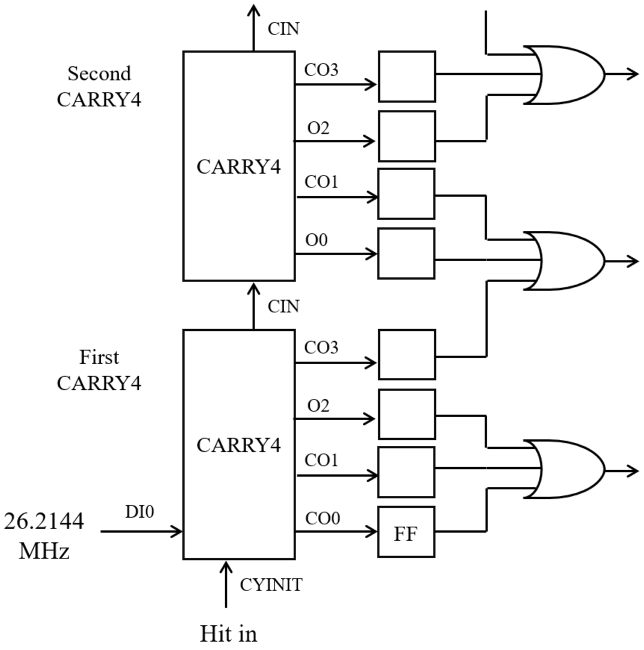


Fig. 5. Block diagram of the initial part of TDL. O and CO outputs are captured by FFs driven by the 500-MHz clock signal. The OR logic operation is conducted for outputs from three FFs. The calibration clock signal of 26.2144 MHz is connected to DI0 of the first CARRY4.

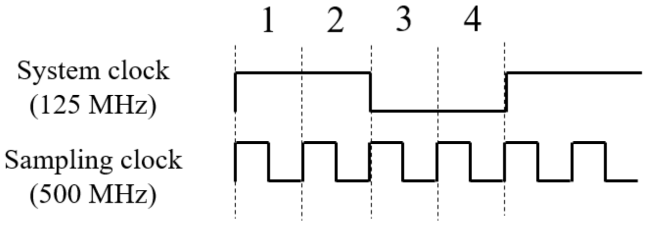
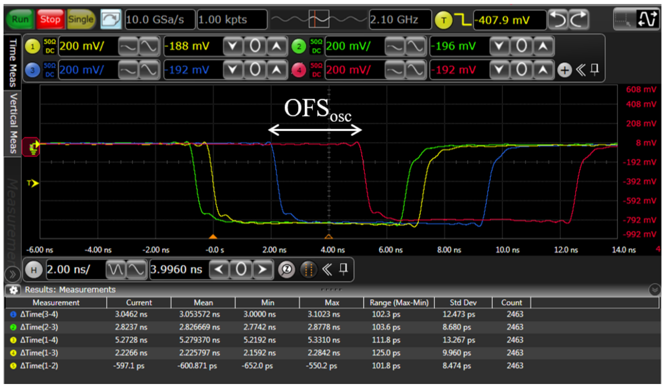


Fig. 6. Four phase regions of the system clock signal.

After binary encoding, data cross the clock domains from 500 to 125 MHz, and thus, additional 2 bits are given. To compensate for nonuniform tap propagation delays, a calibration lookup table (LUT) is used as is common in TDL-based FPGA TDC. Here, the calibration results differ among the four-phase regions, as illustrated in Fig. 6, with varying delays observed even for the same tap. This effect probably originates from ground or power supply voltage noise generated by the system clock signal, as discussed in [13]. Therefore, the LUT for all 4 × 64 patterns is prepared to compensate for this effect.



Results: Measurements

Measurement	Current	Mean	Min	Max	Range (Max-Min)	Std Dev	Count
ΔTime(3-4)	3.0462 ns	3.053572 ns	3.0000 ns	3.1023 ns	102.3 ps	12.473 ps	2463
ΔTime(2-3)	2.8237 ns	2.826669 ns	2.7742 ns	2.8778 ns	103.6 ps	8.680 ps	2463
ΔTime(1-4)	5.2728 ns	5.279370 ns	5.2192 ns	5.3310 ns	111.8 ps	13.267 ps	2463
ΔTime(1-3)	2.2266 ns	2.225797 ns	2.1592 ns	2.2842 ns	125.0 ps	9.960 ps	2463
ΔTime(1-2)	-597.1 ps	-600.871 ps	-652.0 ps	-550.2 ps	101.8 ps	8.474 ps	2463

Fig. 7. Heartbeat signals from four modules. The logic is the NIM standard. Oscilloscope channels 1, 2, 3, and 4 correspond to M1, M2, M3, and M4, respectively. OFS$_{\text{osc}}$ is the phase difference respect to one upstream module.

In the FPGA on the mezzanine card, Str-TDC components up to the front-merger unit are implemented. The MIKUMARI link connects the FPGA on the mezzanine and the main FPGA on the AMANEQ board. The FPGA on the mezzanine also contains the heartbeat unit defining the heartbeat frame, which is synchronized by LACCP.

The transfer speed for TDC data from the mezzanine is 8 Gb/s, matching the internal data bandwidth in FPGA. The back-merger unit implemented in the main FPGA collects data from both mezzanine cards. Finally, data are sent to a PC by SiTCP-XG via 10-Gb Ethernet.

V. Results and Discussion

A. Synchronization Test

A synchronization test was conducted using one clock-root module and three clock-hub modules, connected in series via multimode optical fibers to measure synchronization accuracy. These four modules are labeled as M1, M2, M3, and M4. The fiber length between M1-M2, M2-M3, and M3-M4 are 100, 10, and 3 m, respectively. Fig. 7 shows the heartbeat signals from each module, with the falling edge representing the leading edge of the logic. The heartbeat signal is generated by logic that is synchronous to the local system clock signal. The skew between heartbeat signals at different hops is, therefore, representative of the phase difference between the corresponding system clock signals. In this test, the fine offset value obtained by LACCP between M1 and M4 was 5101 ps, and the mean value measured by the oscilloscope was 5280 ps. A correction of 105 ps was applied to the LACCP result to account for the systematic error caused by the asymmetry of the transmission delay due to SY58605UMG and PI6C5922504. As the resolution of the *dt* measurement is 78 ps, the synchronization accuracy for one time is 100–200 ps. The obtained value is consistent with our expectations.

To further examine synchronization accuracy, the MIKUMARI link initialization process was repeated 1000 times in the M2 module. These fine offset value measurements by LACCP are summarized in Fig. 8. The offset values are distributed across multiple bins due to the automatic adjustment behavior of the IDELAY block. As mentioned in Section II-D, the MIKUMARI protocol searches for a range of delay settings of IDELAY that can stably sample the input signal. At the edges of that range, there are delay settings that are judged to be stable or unstable each time they are examined, resulting in different center positions being estimated. For this reason, the results of automatic adjustment differ slightly each time the link initialization process is performed.

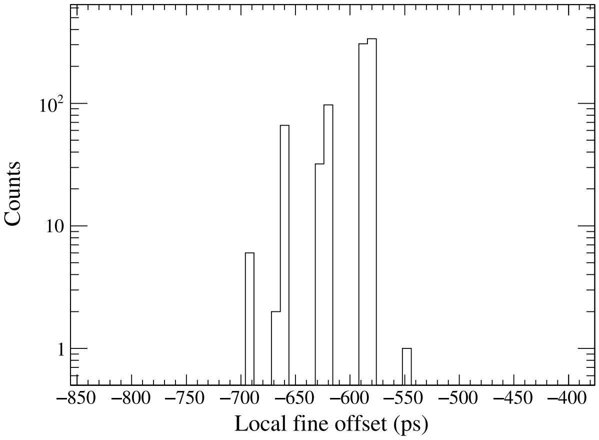


Fig. 8. Distribution of local fine offsets between M1 and M2.

TABLE I
Summary of Offset Measurements

Path	OFS_{osc} (ps)	OFS_{laccp} (ps)	RMS_{laccp} (ps)
M1-M2	−601	−594	24.72
M2-M3	2827	2632	10.6
M3-M4	3054	2953	19.0

The same measurements were performed for M3 and M4, with the average offset (OFS$_{\text{laccp}}$) and root mean square (rms$_{\text{laccp}}$) summarized in Table I together with the time difference measured by the oscilloscope (OFS$_{\text{osc}}$). While local fine offset measurements are identical among these three, OFS$_{\text{laccp}}$ for M2-M3 differs from the oscilloscope measurement value by 195 ps. The measurement results for M1-M2 are almost the same. This discrepancy may have originated from the part-to-part skew of buffer ICs. In addition, there is another buffer IC, Texas Instruments SN65CML100D, used to output the nuclear instrumentation module (NIM) level logic signal between the FPGA and the oscilloscope. While AMANEQ was selected as the first target of this implementation, a dedicated clock distribution module without buffer ICs will be needed for further study of synchronization accuracy. Nevertheless, the obtained results suggest that LACCP can potentially achieve better synchronization accuracy. As the obtained standard deviations are sufficiently small, clock synchronization will be deterministic if the average offset values are used in LACCP.

B. Str-HRTDC Evaluation

To evaluate the cable length dependence of the synchronization accuracy and the timing resolution, a test bench was configured, as shown in Fig. 9. Four modules are labeled as M1, M2, M3, and M4. As the mezzanine cards for HR-TDC are mounted on AMANEQ, there are six FPGAs in total in this system. Pulses from a pulse generator were split and measured by two Str-HRTDCs. Fig. 10 shows the distribution of timing

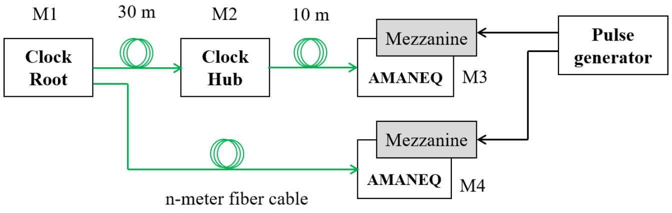


Fig. 9. Block diagram of the test bench. Pulses from the same pulse generator are input to the two HR-TDC mezzanine cards at the same timing.

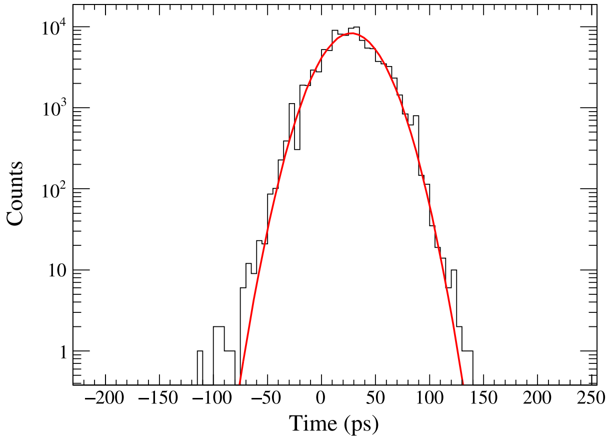


Fig. 10. Typical distribution of timing differences measured by two Str-HRTDCs.

differences measured by the two Str-HRTDCs. To eliminate differences between channels, pulses were fed into the same channel on both cards. The timing offset originating from the input cable length difference was measured and corrected in the analysis.

The optical fiber length between M1 and M4 was varied, and the measurements were repeated, while all other parameters were kept unchanged. The mean of the difference delay obtained from the two Str-HRTDC channels as a function of cable length is plotted in Fig. 11(a). As described in Section V-A, synchronization accuracy is 100–200 ps and is not fully deterministic. The obtained mean values are actually distributed in the range of approximately 300 ps. In addition, data points are centered not around 0 but around 130 ps on average. This behavior remained consistent even when we performed the measurement multiple times; we interpret this as the part-to-part skew systematic error. Despite the variation, no cable length dependence indicates that the T_{rt} measurement and the fine offset estimation are working correctly. Thus, the subnanosecond clock synchronization is achieved.

The timing resolutions as a function of cable length are plotted in Fig. 11(b), which also shows no noticeable cable length dependence. This suggests that the jitter deterioration as a function of cable length is small enough to be negligible. The average timing resolution was 23.1 ps, which satisfies our requirement. We verified this result by measuring two pulses by the same TDC, so as to cancel the clock signal jitter and, thereby, to extract the TDC intrinsic resolution. The obtained intrinsic resolution was 19.5 ps in σ. In the previous work [3], the clock jitter of MMCM was 7.7 ps for 125 MHz based on the time interval error measurement. In addition, an extra random jitter of 3.7 ps, which was measured in the previous work, was added per clock recovery by CDCE62002. In the test bench setup, the clock recovery was performed three times on AMANEQ modules, and the clock signals were generated by MMCM on two mezzanine cards. Thus, the expected timing resolution in this test can be roughly estimated as

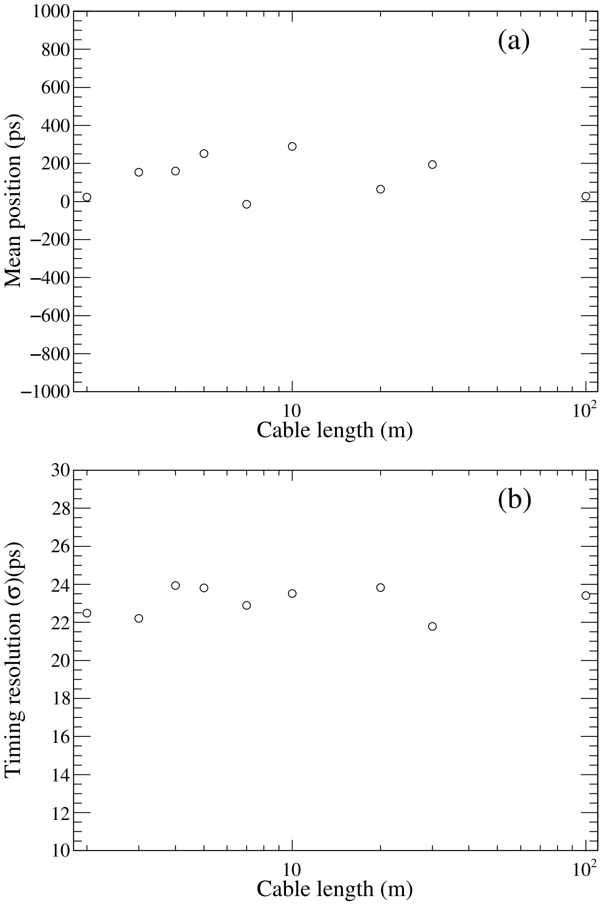


Fig. 11. (a) Mean position and (b) timing resolution of TDC distribution as a function of cable length.

$$\sigma_{\mathrm{res}} = \sqrt{19.5^2 + 3 \times 3.7^2 + 2 \times 7.7^2} \sim 23.6. \quad (3)$$

Here, we assumed that the jitter of the 500-MHz clock signal was the same as that of 125 MHz generated by MMCM. It is consistent with the measured one. In summary, the timing resolution is primarily determined by the performance of TDL performance, with clock recovery by CDCE62002 contributing the least.

VI. Conclusion

A clock synchronization protocol called LACCP has been developed as an upper layer protocol of the MIKUMARI-link protocol. The LACCP clock synchronization is based on the measurements of the round-trip time and the system clock signal phase difference using IDELAYE2 and ISERDESE2 primitives. The synchronization accuracy is approximately 300 ps. By using LACCP, we developed a data-Str-HRTDC, consisting of the AMANEQ module and its mezzanine card.

The TDL-based TDC using CARRY4 primitives was implemented in the AMD Xilinx Kintex-7 FPGA on the mezzanine card. The obtained timing resolution was 23.1 ps in σ, and it did not have the cable length dependence. The obtained clock synchronization accuracy and the timing resolution satisfied our requirements.

In future work, we plan to improve the functionality of the MIKUMARI link and LACCP. This includes adding a feature to repeat the link initialization process after the modules startup to obtain the average fine offset. We also intend to incorporate functionality to compensate for dynamic delay variations caused by factors such as temperature changes. As the fine offset estimation is based on IDELAY adjustment during the link initialization process, it does not seem possible to use the same method during operation because it would disrupt communication over that link. Relative phase compensation methods using the digital dual mixer time difference (DDMTD) and the TDL-based TDC have been reported in [14] and [15], and we aim to test these in the future.

Finally, we are considering implementing MIKUMARI and LACCP on AMD Xilinx UltraScale+ FPGAs. Compared to previous generations of FPGAs, IOSERDES and IDELAY have changed significantly [16] in this family.

Acknowledgment

The authors would like to thank the technical support from the members of the Open Source Consortium of Instrumentation (Open-It). They would like to thank K. Suzuki for proofreading.

References

[1] *SPADI Alliance*. Accessed: May 19, 2024. [Online]. Available: https://www.rcnp.osaka-u.ac.jp/~spadi/

[2] K. H. Tanaka, H. B. C. Group, and H. F. C. Team, "Construction and status of the Hadron experimental hall," *Nucl. Phys. A*, vol. 835, pp. 81–87, Apr. 2010, doi: 10.1016/j.nuclphysa.2010.01.178.

[3] R. Honda, "New clock distribution system based on clock-duty-cycle-modulation for distributed data-aquisition system," *IEEE Trans. Nucl. Sci.*, vol. 70, no. 6, pp. 1102–1109, Apr. 2023, doi: 10.1109/TNS.2023.3265698.

[4] H. Noumi, Y. Morino, T. Nakano, K. Shirotori, Y. Sugaya, and T. Yamaga. (2012). *Charmed Baryon Spectroscopy via the (π, D^{*-}) Reaction*. J-PARC E50 Proposal. Accessed: May 19, 2024. [Online]. Available: http://j-parc.jp/researcher/Hadron/en/pac_1301/pdf/P50_2012-19.pdf

[5] D. Calvet, "Clock-centric serial links for the synchronization of distributed readout systems," *IEEE Trans. Nucl. Sci.*, vol. 67, no. 8, pp. 1912–1919, Aug. 2020, doi: 10.1109/TNS.2020.3006698.

[6] AMD Xilinx User Guide. *7 Series FPGAs SelectIO Resources*. Accessed: May 19, 2024. [Online]. Available: https://docs.amd.com/v/u/en-US/ug471_7Series_SelectIO

[7] R. Honda et al., "Continuous timing measurement using a data-streaming DAQ system," *Prog. Theor. Experim. Phys.*, vol. 2021, no. 12, Oct. 2021, Art. no. 123H01, doi: 10.1093/ptep/ptab128.

[8] T. Uchida, "Hardware-based TCP processor for gigabit Ethernet," *IEEE Trans. Nucl. Sci.*, vol. 55, no. 3, pp. 1631–1637, Jun. 2008, doi: 10.1109/TNS.2008.920264.

[9] *Bee Beans Technologies SiTCP-XG*. Accessed: May 19, 2024. [Online]. Available: https://github.com/BeeBeansTechnologies/SiTCPXG_Netlist_for_Kintex7

[10] AMD Xilinx User Guide. *7 Series FPGAs Clocking Resources*. Accessed: May 20, 2024. [Online]. Available: https://docs.amd.com/v/u/en-U.S./ug472_7Series_Clocking

[11] AMD Xilinx User Guide. *7 Series FPGAs Configurable Logic Block*. Accessed: May 20, 2024. [Online]. Available: https://docs.amd.com/v/u/en-U.S./ug474_7Series_CLB

[12] J. Y. Won and J. S. Lee, "Time-to-digital converter using a tuned-delay line evaluated in 28-, 40-, and 45-nm FPGAs," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 7, pp. 1678–1689, Jul. 2016, doi: 10.1109/TIM.2016.2534670.

[13] C. Liu and Y. Wang, "A 128-channel, 710 M samples/second, and less than 10 ps RMS resolution time-to-digital converter implemented in a Kintex-7 FPGA," *IEEE Trans. Nucl. Sci.*, vol. 62, no. 3, pp. 773–783, Jun. 2015, doi: 10.1109/TNS.2015.2421319.

[14] E. Mendes, S. Baron, J. Hegeman, J. Troska, and N. Loukas, "TCLink: A fully integrated open core for timing compensation in FPGA-based high-speed links," *IEEE Trans. Nucl. Sci.*, vol. 70, no. 2, pp. 156–163, Feb. 2023, doi: 10.1109/TNS.2023.3240539.

[15] H.-B. Xie, Y. Li, Q. Shen, S.-K. Liao, and C.-Z. Peng, "A high-precision 2.5-ps RMS time synchronization for multiple high-speed transceivers in FPGA," *IEEE Trans. Nucl. Sci.*, vol. 66, no. 7, pp. 1070–1075, Jul. 2019, doi: 10.1109/TNS.2019.2904703.

[16] AMD Xilinx User Guide. *UltraScale Architecture SelectIO Resources*. Accessed: May 23, 2024. [Online]. Available: https://docs.amd.com/r/en-US/ug571-ultrascale-selectio