A New Perspective of Flexible Clocking Ideology for Driving and Devising Circuits in Emerging Resource-constrained Applications

Xiangye Wei¹,², Member, IEEE, Liming Xiu¹, Senior Member, IEEE
¹BOE Technology Group Co., Ltd., Beijing, 100176, China
²School of Software and Microelectronics, Peking University, Beijing, China
Corresponding author: Liming Xiu (e-mail: limingxiu@ieee.org).

ABSTRACT Clock is the most important signal in electronic system. In current practice, the dominant clocking style is the fixed-frequency approach. For a given application, the clock signal is only required to work at a few selected frequencies. Moreover, in the process of performing a task, it must not change its frequency. This style has worked well in the past. However, it is preventing information processing efficiency from being raised to next level. Worse yet, the emergence of resource-constrained applications, such as edge computing and IoT, can render this approach unserviceable. In this paper, a new perspective of flexible clocking ideology is advocated. It promotes the use of a clock generator, TAF-DPS, that possesses features of “arbitrary frequency generation” and “instantaneous frequency switching”. Owing to its superior capability in synthesizing waveform, TAF-DPS can handle many signal processing problems in a more elegant manner and serve some special purposes more efficiently. Further, it can be used for generating data for security purposes. TAF-DPS is especially applicable for environments with large frequency instability, which is a major problem in resource-constrained applications. The contribution of this paper is the advocacy of a new perspective of flexible clocking. This ideology is gradually developed in this paper as the argument is carried out through examples of novel architectures enabled by TAF-DPS. The perspective naturally emerges from the collective effect of those novel architectures. The large number of emerging applications, exemplified by edge computing and IoT, inspire system-level innovations that inevitably present new challenges to circuit-level. Those challenges are so demanding that they require overhaul in our circuit design philosophy. This new perspective is such an overhaul in fundamental level. Its aim is to answer the challenges, for better serving the emerging demands in higher levels.

INDEX TERMS clock, edge computing, frequency synthesis, IoT, TAF-DPS, timing.

I. INTRODUCTION
Clocking in electronics is a fundamentally important issue since clock signal is used to establish flow-of-time inside the electronic world. Alongside processor technology, memory technology, analog & RF technology, clocking is regarded as the 4th major technology in the field of integrated circuit design. In the past decades, clock is mostly used in the fixed-frequency style with high-frequency-stability as its highest priority. In clock-circuit design, people have focused almost their full attention on minimizing clock jitter and lowering phase noise [1-9]. When used in applications, such clock signal however has only few useable frequency choices. Moreover, when performing a task, the clock frequency is not allowed to change. Due to this frequency rigidity, this clocking ideology is denoted as fixed-frequency clocking. In our belief, concentration only on one aspect of clock signal (i.e., jitter/phase noise) will not be sufficient for future applications, although this strategy has been working well in the past. For systems in future, this type of rigid clock may not be appropriate as the working environment for emerging applications is not expected to be stationary but dynamic. To meet new challenges, a major overhaul of clocking style is mandatory. In short, the design of clock circuits has to change from focusing only on high-frequency-stability to including some other aspects, such as the issues of amplitude-of-frequency and fast-frequency-switching.

As process technology advances to the deep submicron regime, the working environment for devices has become increasingly harsh. Transistors are experiencing ever-lower supply voltage with a larger swing, working under a much...
larger range of temperature variation. Moreover, they have to endure stronger electromagnetic interference. All those factors collectively make the working of the timing reference (e.g., crystal oscillator) become less reliable. In particular, the accuracy and stability of the timing/frequency source can suffer severely in such rough working conditions. To make this difficult situation worse, many emerging applications demand miniaturization in its physical size and low-power in its operation. These requirements have recently inspired a trend of replacing the crystal-base timing reference with crystal-less solutions. Compared to their crystal counterparts, crystal-less references have relatively inferior frequency quality. Their frequency can significantly deviate from the values specified in datasheets. Considering all those, modern systems are most likely required to work in an environment of large frequency variation in its clock signal. Two typical such scenarios are edge computing and IoT.

IoT has enjoyed rapid growth in recent decades. However, lack of system integration is one of the major bottlenecks that could impair the further spread of IoT adoption. A key factor in this regard is the timing reference. On another front, the growing popularity of edge computing also calls for new timing solutions. Edge computing is a computing paradigm that provides services at the edge of a network, which is closer to the data producer. Due to this geographical advantage, edge computing enjoys fast response, low power, low cost on bandwidth, better security and privacy, and overall low cost. It is especially favorable in real-time processing including self-driving car, video monitoring, location service, etc. Edge computing however mostly occurs in resource-constrained environment. Therefore, the increasing popularity of edge computing and IoT leads to new problems [10-15]. In contrast to the luxury datacenter where cloud computing enjoys, the harsh environment for edge computing and IoT demands small size and low power consumption. This situation most likely cannot afford a decent timing reference, such as the crystal-based solution which is the last and hardest obstruction for SoC (System-on-Chip) integration.

Currently, there is a strong motivation for replacing the crystal with something that is more integration-friendly, such as the Microelectromechanical system (MEMS) and bulk acoustic-wave resonator (BAW). In [16], a 0.3 V all digital crystal-less clock generator is presented for a hearing aid application. A crystal-less frequency locked loop (FLL) for inductively powered implantable medical devices is suggested in [17]. In [18], a crystal-less oscillator is used in compact wireless sensors. In [19], a crystal-less all-Digital FLL is integrated in IoT sensor SoCs. In [20], a frequency reference system for crystal-less wireless sensor nodes is implemented. In [21], an analog crystal-less programmable clock generator is proposed. Those works, however, only address the challenge of devising crystal-less solutions. They do not handle the issue of using them as clock for driving circuit, nor do they deal with the problems caused by the crystal-less solutions. In particular, the elimination of crystal leads to the loss of a stable frequency source (MEMS, BAW and etc. have inferior frequency stability). This results in a difficult situation of doing computation and communication in an environment of large frequency variation.

The incompetence of fixed-frequency ideology in meeting the upcoming challenges of dynamic working condition and the issue of large frequency variation in resource-constrained environment call for a major overhaul in IC clocking. Time-Average-Frequency Direct Period Synthesis (TAF-DPS) is an emerging frequency synthesis technology working on the Time-Average-Frequency (TAF) concept [22]. It has two features of arbitrary frequency generation (AFG) and instantaneous frequency switching (IFS), achievable simultaneously for a given design. It is a circuit level enabler for system level innovations. It has inspired a new clocking ideology that is termed flexible clocking [23-25]. The term “flexible clocking” is created to describe such a clock signal whose supply-of-frequency is ample and its frequency can be quickly switched from one value to another.

By flexible clocking, the frequency can be adaptively adjusted in real-time for functional circuits to accommodate large frequency variation or to handle dynamic working condition (such as loading change) [26]. Moreover, from a broader view, TAF-DPS is a powerful tool for manipulating pulse. It hence can be used for creating clock signal for special purposes, such as dynamic frequency scaling for low power operation, spread spectrum clock generation for reducing EMI, chirp signal generation and PWM generator, etc. Furthermore, owing to its capability in synthesizing various waveforms, TAF-DPS can be used for producing data for security purpose (e.g., random number generator and physical unclonable function). All those capacities can be packaged into a framework for addressing the concerns in resource-constrained applications. It can meet the upcoming challenges in future electronic design.

The contributions of this paper are summarized as follows:

- Several TAF-DPS enabled novel architectures are presented with the aim of addressing the respective problems more efficiently. The focus is to elucidate the point that high flexibility in clock signal can help create much efficient system.
- A perspective of flexible clocking ideology emerges from the collective effect of those novel architectures. It is argued that this new perspective is valuable in guiding future electronic design.

The rest of this paper is organized as follows. Section II is a brief review of TAF-DPS technology. It is the foundation for all the works that will be discussed in later sections. In section III, two TAF-DPS enabled architectures for data transportation are presented. In section VI, four techniques enabled by TAF-DPS for serving special purposes are discussed. In section V, two TAF-DPS based architectures in producing data for security service are explained. The works discussed in section III, IV and V are all constituted with the
TAF-DPS as their underpinning technology. They all operate by following the flexible clocking ideology. In section VI, all the architectures discussed are put into an integrated framework where the ideology of flexible clocking emerges. This new perspective is then further elucidated for its power in handling future challenges. A conclusion is made in section VII.

II. BRIEF REVIEW ON TAF-DPS

\[ T_A = I \Delta, \quad T_B = (I + 1) \Delta \]  
\[ T_{TAF} = \frac{1}{f_s} = (1 - r)T_A + rT_B = (I + r)\Delta = F\Delta \]  
\[ df_s = \frac{dF}{F^2\Delta} = \frac{df}{F} \cdot \frac{1}{f_s} = -\frac{df}{F} \]  
\[ \frac{df_s}{F} = -\frac{dF}{F} \]

Figure 1 depicts the working principle of TAF-DPS. From a base unit \( \Delta \), the synthesizer first creates two types of cycles \( T_A \) and \( T_B \). Their length-in-times are given by (1), where \( I \) is an integer. When synthesizing a particular frequency \( f_s \) (period \( T_{TAF} \)), it uses \( T_A \) and \( T_B \) in an interweaving fashion. Output period (frequency) is expressed in (2) where \( F = I + r \) is the frequency control word. Fraction \( r \) controls the occurrence probability of \( T_A \) and \( T_B \). By changing the value of \( F, f_s \) can be accordingly tuned. Frequency resolution can be derived from (2) and is expressed in (3). Sub-ppb frequency granularity has been achieved in real chip when \( r \) is adjusted in very small step. The \( 1/x \) curve shown is its \( f \) vs. \( F \) transfer function, which is the graphical form of (2). Monotonicity is guaranteed by the frequency-control relation of \( 1/x \). In small region, this curve is virtually linear.

The first circuit technique to implement the TAF-DPS principle is Flying-Adder frequency synthesis architecture, developed around the late 1990s [27-28]. Selected examples of recent development can be found in [29-31]. Over the years, it has been used in a variety of commercial products. Fig. 2 is the Flying-Adder circuit block diagram. A plurality of \( K \) phase-evenly-spaced signals of frequency \( f_s \) is fed into the synthesizer. The \( \Delta \) is formed by the time span between any two adjacent such signals. Its frequency \( f_s \) is controlled by \( F \). Flying-Adder synthesizer is an edge selector and combiner. At any particular moment, it selects one signal, among those \( K \) signals, and passes it to the output. Over time, selecting signals using a predetermined schedule enables a pulse train of desired period to be created. Small frequency granularity is achieved by adopting the TAF concept. Fast frequency switching is accomplished by directly composing the waveform of each pulse. For high-performance designs, the circuit shall be created by using transistor-level custom approach [29-30]. For low-cost cases, it can be implemented by using the digital design approach of HDL-synthesis and automatic place & route [31-32].

For generating the \( K \) phase-evenly-spaced signals, the most widely used structure is multiple output ring oscillator (MORO). The MORO can be created in several ways. For high-performance, transistor-level custom design approach is preferred [29-30]. For other cases, simpler options are available. One of those is to use 16 cross-coupled NAND2 gates that are directly instantiated from an ASIC standard cell library [32]. In another solution, a group of 16 signals is generated from a divider (e.g., Johnson counter), which is made of 8 serially connected flip-flops [31]. Fig. 3 includes the block diagrams of the three approaches discussed for creating the MORO.

The TAF-DPS circuit can be used for clock generation, and many other purposes as well (will be discussed in the following sections). When it functions as clock generator, the circuit driven must be constrained by using \( T_A \) (the shorter cycle) as the setup constraint. Hold check is not affected. For more information on this issue, please refer to section 1.3 and 4.22 of [23] and section 3.2 and 3.6 of [33]. The interwoven use of \( T_A \) and \( T_B \) is a frequency modulation. It can generate spurious tones in frequency spectrum. Hence, it is not suggested for TAF-DPS clock to directly drive data...
 converter (e.g., ADC and DAC). Some compensating circuit (such as randomization circuit) must be incorporated if TAF-DPS is used to drive converters. Also, TAF-DPS output is not suggested for being used as RF signal.

In most applications, TAF-DPS is integrated as an IP in SoC as illustrated in Fig. 2. This clock IP is a powerful tool that can assist various signal processing tasks to achieve higher efficiency. It enables system-level innovations in SoC [23, 26, 33-34]. When used in system level, the TAF-DPS circuit is usually treated as a black box as depicted in Fig. 4. This black-box view will be used often in the discussion of following sections. In Fig. 5, a set of data obtained from a real chip is used to demonstrate the TAF-DPS’s $f$ vs F transfer function of (2). In the top graph, the control word F is changed continuously and its pattern-of-change is displayed. In the middle graph, the measured TAF-DPS output period is displayed. As seen, the TAF-DPS output frequency (period) faithfully follows the value of F. In the bottom graph, the captured TAF-DPS output waveform is displayed. The correspondence between the waveform and the F (and the measured frequency) can be clearly seen. In Fig. 6, another set of data is shown to demonstrate the TAF-DPS’s fine frequency granularity, sub-ppb resolution has been achieved. Fig. 5 and 6 provide some hard evidences to support the claim of two features of AFG and IFS.

III. ARCHITECTURES FOR DATA TRANSPORTATION

In this section, TAF-DPS technology will be used in enabling two architectures that are related to the task of data transportation. The two features of AFG and IFS are utilized to improve the efficiency of data movement.

A. TAF-DPS AND FIFO

One of the ubiquitous issues in VLSI circuit design is the data transportation from one place to another. For successful data transfer, a storage device must be inserted between the two communicating parties for accommodating the difference in data-flow-rate. The storage is typically embodied in the form of FIFO (first in first out) memory. FIFO design is so omnipresent that a significant portion of silicon area is dedicated to various FIFO memories in almost all SoC. Hence, a slight improvement on the FIFO design efficiency can have a big impact, for the benefit of saving area and power.

Implementation-wise, FIFO is usually implemented as a circular queue with two pointers and a set of status flags. In the left of Fig. 7, the block diagram of a typical FIFO device is shown. At the center is a circular buffer with two pointers of write and read attached to it. At each side of the FIFO, there is a corresponding clock domain where the data is driven by its respective clock. The clock for write (CLKW @ $f_w$) and the clock for read (CLKR @ $f_r$) can be the same signal (synchronous FIFO) or two different signals (asynchronous FIFO). In FIFO design, the primary goal is the minimization of storage size. In addition, there is another concern on the continuity of the outgoing data flow, which could be important for some applications. The current status on FIFO design can be summarized as follow.

- The two clocks respectively driving the write and read ports can be the same signal or two signals of different frequencies. In both cases, the clock frequencies are fixed (i.e., not dynamically adjustable in real-time).
- The incoming data flow can be broken. In other words, there is no guaranty of valid data on every CLKW cycle.
- The primary design goal is the storage-size minimization.
- Another goal is to make outgoing data flow continuous.

Those design goals are hard to be satisfied when the clock frequencies are rigidly fixed. For the issue of data continuity, in synchronous FIFO, the outgoing flow simply cannot be
continuous if the incoming flow is broken. For asynchronous FIFO, the rate difference between \( f_w \) and \( f_r \) needs to match the pattern of “valid data”, which is dynamically varying and usually unknown at design time. For the issue of storage size, it mainly depends on the rate difference of \( f_w \) and \( f_r \). In addition, the “valid data” pattern can influence the size as well. In short, when clock rates are rigid, the storage size is hard to be reduced. The larger the rate difference is, the more irregular and unpredictable the pattern of “valid data” is, the larger the storage size needs to be.

TAF-DPS provides a decent solution to this problem. The features of AFG and IFS make its output a very flexible clock. This flexibility in frequency can readily accommodate the rate difference and the variation in data pattern. It can help reduce storage size and smooth outgoing data flow. As shown in the right-hand side of Fig. 7, TAF-DPS is used to drive the outgoing flow. Depending on the emptiness/fullness of the FIFO, the TAF-DPS output frequency can be dynamically adjusted, slowing down or speeding up, to serve the purpose of storage-size-reduction or data-flow-smoothing [35]. During this process, the adjustment on the \( f_r \) can be deliberately calculated. The ample supply-of-frequency and the quick frequency-switching are very valuable for this task. This use of an adaptive TAF-DPS clock for assisting FIFO design is called TAF-DPS FIFO.

Assuming that there are 100000 bits of data to be transported from clock domain \#1 to \#2, where \( f_w \) and \( f_r \) are fixed at 50 and 40 MHz, respectively. Under this assumption, the size of storage must be at least \( 100000 \times (50-40)/50 = 20000 \) bits. Using any circular buffer smaller than this size will lead to data loss. If TAF-DPS FIFO is used, this size can be significantly reduced. For instance, a strategy can be applied that sets the \( f_r \) to 66.7 MHz if the amount of data inside FIFO is more than 100 bits. When less than 50 bits, the \( f_r \) can be slowed down to 40 MHz. Adjusting the CLKR in this fashion, the storage size can be controlled within 200 bits, which is 100 times smaller than the original.

The corresponding simulation result is displayed in Fig. 8. On the left, the trends of data-amount-growth inside FIFO are shown. The blue curve is for conventional FIFO while the red one is for TAF-DPS FIFO. In the conventional case, the FIFO will be completely full after 2 ms. If the incoming bitstream continues, some data will be lost. In contrast, the amount of data held inside the storage is never more than 100 bits in the case of TAF-DPS FIFO as evidenced in the red curve. This is more clearly shown in the right side of Fig. 8. As seen, the \( f_r \) of CLKR jumps between 40 MHz and 66.7 MHz to rein the data amount inside the FIFO, within a range of 50 to 100 bits.

Our second example concerns the task of keeping the FIFO outgoing flow as smooth as possible. The incoming data is still driven by CLKW of \( f_w = 50 \) MHz. However, the data flow is broken in that some clock cycles do not have associated data. We want the outgoing flow continuous, and preferably, its rate varies as smoothly as possible. Fig. 9 shows the simulation of...
such a scenario. At the top, the data valid flag of the incoming data is shown. When “0”, there is no data in the corresponding clock cycle. At the bottom, the trends of $f_n$ and $f_r$ are shown. The $f_n$ (ref) is fixed while the $f_r$ (blue) is varying. The adjustment is based on the status of data amount inside FIFO, which is indicated in the middle. As a result, the outgoing flow is continuous (every CLKR cycle has valid data). The FIFO is never empty. Its content is always kept at an appropriate level.

At circuit level, Fig. 10 shows the TAF-DPS’s capability of dynamical frequency adjustment that qualifies it as an adaptive clock. In the top graph, two traces are displayed. On the top is the TAF-DPS control signal and at the bottom is the output clock waveform. As seen, the output waveform (and its frequency) faithfully follows the pattern of the control. In the middle and bottom graphs, some detailed views are presented, from fast-to-slow and from slow-to-fast. The switching of frequency is seamlessly smooth (glitch-free), which is a mandatory requirement for a clock to drive digital circuit. It is worth mentioning that, although the clock rate is varying, the data flow is continuous (having valid data at every cycle).

**B. TAF-DPS AND Clock Data Recovery**

In serial communication design, one of the key components is the clock data recovery (CDR) circuit. Unlike in parallel link, the clock signal is not transmitted but has to be extracted from the transmitted data. CDR has become extremely popular in modern electronics, especially for high-rate inter-chip communication. At the top of Fig. 11, the principle of clockless transmission is illustrated. Its advantage lies in the fact that it eliminates the skew problem between the multiple channels, which is a serious challenge in parallel link design.

$$T_{txavg} = T_{rxavg} = a_0 \cdot T_0 + a_1 \cdot T_1 + a_2 \cdot T_2 + \ldots + a_n \cdot T_n \sum a_i = 1 \quad (4)$$

In the bottom left of Fig. 11, the principle of conventional CDR is shown. The goal of this loop is to satisfy equation (4), where $T_{txavg} = 1/f_{txavg}$ is TX’s average clock period (frequency) and $T_{rxavg} = 1/f_{rxavg}$ is RX’s average. $T_0, T_1, T_2, \ldots, T_n$ are the periods of individual clock cycles in RX and $a_0, a_1, \ldots, a_n$ are their possibilities-of-occurrence. The frequency matching between TX and RX is achieved in a long-term average sense. For each individual cycle, it is meaningless to discuss frequency matching (since the concept of frequency is established in a long-term frame of one second). The requirement on CDR is for the average clock rates of the two sides being matched over multiple cycles. Meanwhile, within each cycle, the incoming TX bit can be reliably latched by the RX circuit using the extracted clock signal.

In construction, binary phase detector is often used because of its speed advantage (the high data rate makes it difficult for a linear detector to work reliably). Naturally, its output is in digital fashion. On the other hand, the local oscillator in RX is a VCO (voltage-controlled oscillator) controlled by an analog voltage. Hence, a D→A (digital to analog conversion) process is needed in the RX circuit to complete the loop. This leads to high cost and, more significantly, slow response. For a typical CDR of this type, to satisfy equation (4), the value of $n$ can be a large number due to the slow response. This can have a negative impact on the performance of CDR circuit (frequency tracking capability and jitter tolerance).

In the bottom-right of Fig. 11, the architecture of Time-Average-Frequency CDR (TAF-CDR) is depicted. The enabling block is the TAF-DPS, functioning as a digital controlled oscillator (DCO). TAF-CDR takes advantage of the digital nature of the detector output and DCO control. The entire loop can be built in full digital fashion. Owing to the fast and quantifiable response speed of this DCO (two RX cycles), loop latency $M$ can be made small, where $M$ is the time elapse (in number of RX cycles) between the moment-of-detecting and the moment-of-action. Thus, the loop can respond quickly to the input change. Further, the value of $M$ can be precisely determined, resulting in a quantifiable loop dynamic.

Instead of using many $T_{txavg}$ frequencies (periods) to match $T_{rxavg}$ as in traditional CDR, the DCO in TAF-CDR only generates several discrete frequencies and uses them for frequency matching. For example, three unique cycles (three frequencies) can be produced, each dedicatedly serving the role of speed-up, hold or slow-down. In a timeframe of $M$ cycles, the loop roughly frequency-matches the TX and RX. In any of those timeframes of $M$ cycles, the matching is only an approximation. But it is good to the point of guaranteeing...
that the TX data can be reliably captured by the RX. For each individual cycle, the setup and hold margin of the sampling cell in the phase detector is satisfied. In long term, equation (4) is satisfied precisely just as in the case of conventional CDR. Figure 12 shows a transistor-level simulation of TAF-CDR. The design is implemented in a 180nm CMOS process. The nominal TX rate is at 300 Mbps. The loop latency is designed as M = 3. The DCO is designed to generate three frequencies: 300 (hold), 342 (speed up) and 267 MHz (slow down). They are produced by a TAF-DPS, respectively using 8Δ, 7Δ and 9Δ where Δ = 417 ps is generated from an 8-output MORO running at 300 MHz. In the simulation, the TX data is driven by a clock intentionally set at 295 MHz (0.16% slower than the nominal). The trace at the bottom is the TX data, generated from a 23 bits PRBS encoder driven by the TX clock (= 295 MHz). The RX operates in a fashion that its frequency is dynamically switching among the three designated values. The selection of the value is made in every M = 3 cycles. As seen in the top trace where the DCO’s output vs time is displayed, the loop adjusts frequency once in every three cycles. At any particular moment, it selects one among the three values of 267, 300 and 342 MHz. The decision is made in real-time based on the relation between incoming data and the RX clock. The third trace is the recovered data. The second trace is the output from an error detector which is made of the corresponding PRBS decoder. The error detector is fed by the recovered data and is driven by the recovered clock. It will report error (logic high) if the recovered data is not agreed with the TX data.

The simulation shows that the incoming data is correctly latched by the receiver and clock is rightfully extracted from the incoming data. It is important to emphasize that the TAF-CDR correctly functions in a harsh environment of incoming rate (295 MHz) being 16666 ppm away from its nominal value (300 MHz). The top graph in Fig. 12 illustrates that the 295 MHz of TX is averagely matched by the DCO output in RX. The majority of the samples are at 300 MHz (the hold state). More samples of slow-down (267 MHz) are used than that of speed-up (342 MHz). This is a reflection of the fact that the incoming data is slower than the nominal value.

The TAF-CDR architecture is all-digital, which makes low-cost implementation possible. One such example is an ultra-low-power and low-cost CDR function (<1 uW) for a low data rate application of 2 Mbps, targeting for IoT applications. It is implemented in a 65nm process. Low-cost is achieved by building the TAF-CDR circuit entirely from standard cells. The Δ = 31.25 ns is generated from a 16-output 2 MHz Johnson counter. The TAF-DPS based DCO is responsible for generating three types of cycles: 15A (468.75 ns), 16A (500 ns) and 17A (531.25 ns). In this design, the entire action-chain of detect-calculate-adjust can be completed within one cycle since the clock rate is low (< 2 Mbps → bit-time is around 500 ns). Thus, the loop latency can be designed as small as M = 1. As a result, this TAF-CDR has a very fast response speed. It leads to high tolerance to frequency error since the loop can respond very quickly to correct any detected error.

An Agilent E4437B signal generator is used to provide the test bitstream (the TX data). Its internal BER (Bit Error Rate) tester is used to check the received bitstream (the RX data). A 2 Mbps bitstream is sent from the signal generator to the TAF-CDR. The recovered data from the TAF-CDR is sent back to the BER tester for validation. Fig. 13 shows some of the experimental results. The snapshot on the right is the BER test result. The input data is a 2 Mbps bitstream of PN9. The number of bits tested is 4294967295 (= 232−1). The BER check
result is zero. The screen capture on the left are the waveforms of incoming TX data (top), recovered RX data (middle) and recovered clock (bottom). As seen, the recovered data follows the incoming data faithfully. It is however delayed by one cycle due to the latency of $M = 1$. In the waveform of the recovered clock, it can be seen that there are three types of cycles of different lengths. They are the cycles of slow-down, hold and speed-up. For example, the cycle bounded by the two markers is measured as 540 ns, which is the slow-down cycle (the nominal cycle is 500 ns).

Owing to the small latency of $M = 1$, this TAF-CDR has a high tolerance to frequency error. The DCO’s nominal frequency (the frequency for hold) is designed to the nominal rate of 2 MHz. When the TX rate deviates from its nominal value, the TAF-CDR needs to track it at its best effort. This tracking capability is tested in the lab by using an adjustable frequency source. Table I and II are the results of bitstream PN9 and PN15, respectively. In all the cases, the number of bits used is $2147483647 = 2^{31} - 1$. The frequency has been adjusted from both directions: frequency+ and frequency-. From those tables, it is seen that this TAF-CDR can tolerate large frequency error. The BER result from PN9 shows a better tolerance of frequency error than that of PN15 since the no-transition period is longer in PN15. For more information on TAF-CDR, please refer to section 5.4 of [33].

In summary, the key benefit of TAF-CDR is the elimination of the $D \rightarrow A$ process. More, due to the loop’s fast response, frequency tracking capability and jitter tolerance is greatly enhanced. Further, the TAF-CDR can be implemented in full digital fashion.

IV. ARCHITECTURES FOR ACHIEVING SPECIFIC EFFECT ON CLOCK SIGNAL

In this section, we will discuss several architectures for achieving special effect on clock signal, including dynamic frequency scaling, spread spectrum clock generation, pulse width modulation, chip signal generation and frequency locked loop. They are all enabled by the TAF-DPS features of AFG and IFS.

A. TAF-DPS FOR DYNAMIC FREQUENCY SCALING

Dynamic frequency scaling (DFS), also known as CPU throttling, is a power management technique. Depending on computation loading, the frequency of the clock driving a processor can be adjusted accordingly to conserve power and reduce the amount of heat generated. DFS can help prolong battery life on mobile devices and decrease cooling cost on mainstream computing. In practice, DFS almost always appears in conjunction with dynamic voltage scaling since lower voltage can be used when a slower clock frequency is driving the circuit. The combined effort is known as dynamic voltage and frequency scaling (DVFS).

TAF-DPS is an ideal tool for DFS thanks to its features of AFG and IFS. An example can be offered here to illustrate the point. In this example, we want to keep the chip’s temperature as steady as possible at a target value (a hypothetical scenario).

It is well-known that power consumption can be estimated by $P = C \cdot V^2 \cdot f$ where $C$ is the capacitance being switched per clock cycle, $V$ is supply voltage, $A$ is activity factor indicating the average number of switching events and $f$ is the clock frequency. We further assume that power consumption and chip temperature has a simple linear relation of $T_e = T_s + (P_0 - P_a) \cdot \Delta t/C_m$, where $T_e$ and $T_s$ are ending and starting temperatures, respectively, $P_e$ is current power consumption corresponding to operating frequency $f$ and $P_a$ is the power dissipated as heat, $\Delta t$ is the time difference between start and end, $C_m$ is a coefficient representing the power needed for a temperature increase of 1 °C. Under this postulation, the operation of TAF-DPS can be controlled and its frequency profile can be tuned towards the goal of keeping temperature steady. A simulation is displayed in Fig. 14. By dynamically adjusting clock frequency, the chip temperature can be reined
near the target of 40 °C. This is an illustration that can be tailored in real situation. In real case, on-chip temperature monitor can be used for monitoring temperature continuously. Its output is then used to direct TAF-DPS operation. During the process, certain algorithms can be developed to fulfill user’s specific need.

In applying DFS to real applications, a crucial issue is to keep the normal workflow from being interrupted when the clock frequency is varying. In other words, we want the system to continually operate even when its driving clock is making significant frequency change. TAF-DPS can meet this challenge since it can switch its output frequency seamlessly. To demonstrate this, we create a test circuit of multiplication-then-division as depicted at the top of Fig. 15. The circuit is clocked by CLK. When it works correctly, the circuit outputs a logic “0”. From an operation perspective, this circuit is representative of all digital circuits regardless of their functionalities. We want to use this figurative circuit to show the effectiveness of the TAF-DPS clock in a general sense.

This test circuit is mapped into a Kintex-7 FPGA. The setup constraint is set to 120 MHz. The TAF-DPS control word F is switched in a pattern of $32 \rightarrow 4 \rightarrow 32$, in step of one. For each F value, a time frame of 50 cycles is allocated to it. This pattern is plotted as the brown curve in Fig. 15. When F changes, the TAF-DPS frequency is measured and displayed as blue. As seen, the frequency sweeps from 20 MHz to 160 MHz as F changes. This TAF-DPS output is used to drive the test circuit. The bottom portion of Fig. 15 is the screen capture of the test circuit output waveforms. At the top is the waveform of TAF-DPS output. The waveform is continuously varying since its frequency is constantly changing. The red curve at the bottom is the test circuit output. It fails (output becomes “1”) in the region when the frequency is high. This is because the test circuit is constrained at 120 MHz and the setup constraint is violated when its clock’s frequency is higher than that value. In all the other regions of lower frequencies where the setup constraint is met, the circuit works correctly. To our best knowledge, this feature of seamlessly frequency switching has not been reported before. This clocking style of no-interruption-to-system-operation is definitely valuable for driving processors and many other circuits.

B. TAF-DPS FOR SPREAD SPECTRUM CLOCK GENERATION

Spread spectrum clock generation (SSCG) is an important technique for mitigating EMI problems in electronic systems. In frequency domain, it can spread the energy of a clock signal from a single line to a band, lowering its peak power and weakening its capability of producing harmful effects. SSCG is implemented by constantly changing the instant frequency of the clock (an action of frequency modulation), scattering its energy. Although the emitted energy remains unchanged, the power spectrum at a working frequency (and its harmonics) is reshaped from a sharp peak to a lower and wider platform. The modulation can be created by adding a time-varying command to the circuit for controlling frequency, called modulation profile. The parameters defining such profile include type (frequency deviation direction such as up spread, down spread, and central spread), depth (frequency deviation magnitude), rate (modulation frequency or command changing rate), and waveform (command curve shape).

Using the TAF concept, the clock pulse train of TAF-DPS is made of multiple types of pulses. Therefore, without any additional action, the TAF clock already spreads its energy to several frequency points. To achieve more desirable results, however, some additional circuits are needed. In general, the circuit for SSCG is preferred to have the following characteristics: 1) Direct: easy to handle, preferably linear frequency mapping; 2) Fast: quick response to tuning command; 3) Fine: small frequency granularity for accurate tracking; 4) Simple: small area, high power efficiency. Owing to its distinguished features of AFG and IFS, TAF-DPS is squarely suitable for SSCG. The scheme of using TAF-DPS for SSCG is illustrated in Fig. 16.

Figure 17 includes simulations for illustrating the working difference in spread between TAF and conventional frequency (CF). It shows a case of doing spread spectrum using a sinusoid profile. In c), the period/frequency vs. time trend is displayed when CF is used in the clock. The modulation range for period variation is $[11\Delta, 12\Delta]$. As seen, the period varies gradually between the minimum and maximum. The plot in d) shows its period distribution. A plurality of periods is used in the process. It shows the characteristic of doing spread spectrum using CF. The plot in a) is the period vs. time trend of doing the same sinusoid SSCG but using the TAF concept. It is interesting to see that only two types of periods, in contrast to a group of periods in CF case, $11\Delta$ and $12\Delta$ are used. The sinusoid profile is accomplished through the varying densities.
of the periods. Two zoom-in areas are also included to make this point clear. In b), its period distribution is displayed and, as expected, there are only two types. This simulation is helpful for the understanding of TAF-DPS SSCG.

A novel modulation type, boundary spread, is devised by applying TAF-DPS for SSCG. The boundary of its frequency deviation is naturally formed by fully utilizing the maximum potential modulation depth under the constraint that only two types of periods are used. With TAF-DPS SSCG, no extra jitter is introduced. TAF-DPS SSCG can achieve large modulation depth and still offers high-quality clock for driving circuit. As a result, system incorporating this technique can work properly with high confidence when SSCG clocking is turned on. This is in contrast to the traditional approaches where normal function cannot be guaranteed with confidence when SSCG is turned on. This technology is termed Always-on Boundary Spread SSCG, or AOBS [36].

Figure 18 shows a set of simulations of using TAF-DPS SSCG for down spread and boundary spread. The Δ is set as 0.625 ns. The nominal working frequency is 143 MHz. The modulation rate is chosen as 30 kHz. The enveloped-FFT responses with no-spread and boundary spread using triangular profile are shown. Results from down spread using a triangular profile with 5000ppm, 10000ppm and 50000ppm are also shown for comparison. Boundary spread achieves the largest distribution on energy (lowest peak power) [36].

To validate its effectiveness, the AOBS circuit has been implemented on a real system and the outputted clock is studied using a pre-compliance test. Using Keysight N9000B signal analyzer working in EMI receiver mode, a 120 MHz output from TAF-DPS SSCG is studied for different RBW settings. The plots on the left of Fig. 19 show the result. The modulation rate is 30 kHz. The modulation profile is triangular and the modulation method is boundary spread. Three tests with respective RBW settings of 9 kHz, 30 kHz and 120 kHz are carried out. The envelop detector used is peak detector. As seen, the boundaries of measured EMI responses agree with the designed values. The AOBS is also used to compare the effectiveness of different modulation profiles. Those results are shown on the right of Fig. 19. The outputs from triangular, sawtooth/ramp and random profiles are plotted in the same graph along with the no-spread result. For more details, please refer to [36].

C. TAF-DPS FOR PULSE WIDTH MODULATION

Pulse width modulation (PWM) is a method of controlling the average amount of “information” delivered by an electrical signal through an operation of chopping the signal into discrete parts. The “information” specified can be an electrical charge, electrical voltage, time duration, digital message, energy (power) and etc. As its name suggests, the media in PWM is an electrical pulse of a square waveform. This pulse has two distinguished states of high and low. The ratio between the time durations of those two states is the intended information. This ratio is reflected as a numerical value called duty cycle, symbolled as the width of the pulse (the time duration of the “high” state).

The traditional way of generating PWM signal is by using a sawtooth or triangle waveform and a comparator. The sawtooth or triangle waveform, called modulation waveform, is compared against a reference signal. When the value of the reference is higher (or lower) than that of the modulation waveform, the PWM signal is in the high state. Otherwise, it is in the low state. In its implementation, the comparison between the modulation and reference signals can be carried out in a more sophisticated fashion, such as delta modulation...
and delta-sigma modulation. With the advance of digital technology, nowadays, PWM signal is often generated by using digital circuits. A simple digital counter driven by a clock signal can do the trick. At every clock tick, the content of the counter incrementally grows. When the counter value reaches a predetermined reference value, the PWM output changes state from high to low (or low to high). It is reset at the end of every period of the PWM operation. This technique is appropriately referred to as time proportioning, for the fact that a certain proportion of a fixed cycle time is spent on the high state.

This principle of time proportioning can be easily realized by TAF-DPS. As a circuit level tool for direct waveform construction, TAF-DPS is naturally suitable for being a PWM signal generator. Compared to the digital counter method, the key advantage of TAF-DPS PWM modulator is finer time resolution that can offer higher performance. Moreover, it has high flexibility in generating a variety of waveforms of various timing characteristics. In particular, it can generate three types of PWM waveforms: Type-I of varying frequency, Type-II of fixed-period-varying-duty-cycle and Type-III of fixed-pulse-length-varying-period. In Fig. 20, the scheme of using TAF-DPS for generating PWM signal is illustrated. The profile for desired PWM signal is first generated by a profile generator and be converted into a frequency control word, which is then fed into the TAF-DPS circuit. It subsequently produces the corresponding PWM signals. In Fig. 21, three types of waveforms are displayed. Those waveforms are captured from the output of a real chip.

**D. TAF-DPS FOR CHIRP SIGNAL GENERATOR**

A chirp signal is such signal whose frequency increases or decreases with time. This type of signal is commonly used in sonar and radar. It is also found in other applications such as spread spectrum communication, resonant converters, electronic ballasts and etc. Chirp signal can be generated with analog circuitry via a voltage-controlled oscillator (VCO). If the VCO input is controlled by a linearly or exponentially ramping voltage, a linear or exponential chirp signal can be produced. Chirp signal can also be generated digitally by using direct digital synthesizer (DDS) which consists of numerically controlled oscillator (NCO), digital to analog converter (DAC) and reconstruction lowpass filter. Moreover, it can be generated by using YIG (Yttrium Iron Garnet) oscillator. One of the major issues with all the aforementioned generators is the implementation cost. Traditionally, the hardware cost of a chirp generator is high. Usually, discrete components have to be used to build functional modules. The application of chirp signal is thus limited, especially so for on-chip operation.

Owing to the linear relationship between the control word and its output period, TAF-DPS can be an effective tool for generating pulse-shaped chirp signal. Fig. 22 is the general architecture of TAF-DPS based on-chip chirp signal generator. A digital modulation block is responsible for generating the desired pattern in the form of a series of digital values. Those values are then fed into the TAF-DPS as the frequency control word F. Following equation (2), the output signal can bear a trend of frequency vs. time defined by the modulation pattern. Owing to the mathematically traceable transfer function for TAF-DPS (almost linear in a small region), many forms of chirp signal can be produced.

In Fig. 23, some measured data from a real chip are presented. On the left, sawtooth, sinusoid and triangle patterns are first generated by the digital modulation block, the resulting chirp signal is then frequency-measured by using a frequency counter. As seen, the frequency-trends follow the modulation patterns faithfully. The patterns of sawtooth, sinusoid and triangle are first generated in the control word F, the nominal value for F is 14.5. Its range-of-change is [14.46875, 14.53125], about ~0.4% of nominal value. The frequencies are subsequently generated.

On the right-hand side of Fig. 23, a stream of random numbers is created in the modulation block. A 7-bit PRBS generator is used to generate the stream of random numbers. In this case, F is varied in the range of [14.50048828125, 14.56201171875]. The corresponding frequency trend is then measured and displayed. The two trends of frequency and
control word clearly form a mirror-image pair, revealing that frequency is inversely proportional to the magnitude of the control word. In all the plots of Fig. 23, the TAF-DPS output frequency responds to the change of control word in an almost linear but mirrored fashion.

This on-chip TAF-DPS chirp generator enables another interesting architecture. For modern ASIC-based designs, phase locked loop (PLL) is a standard IP that is available on almost all designs. Using the standard PLL, an architecture for boosting chirp signal’s frequency can be created as depicted in Fig. 24. A TAF-DPS chirp generator is used to generate chirp signal at low frequency. Its output is then fed into an integer-N PLL that frequency-boosts the original chirp signal by N times. In this method, the chirpiness (rate of frequency change) shall be limited to a threshold, namely the PLL loop bandwidth, so that the PLL can handle it faithfully.

Figure 25 presents a real case where this architecture is implemented in a product using Xilinx Spartan-6 FPGA. As shown on the left, the frequency modulation pattern is initially created in a relatively low frequency region wherein the TAF-DPS circuit operates comfortably. The resulting signal with the designated frequency modulation pattern is then fed into an integer-N PLL. This frequency modulation pattern is duplicated at the PLL output with its center frequency multiplied N times and its modulation depth preserved. This scheme holds as long as the input modulation rate falls within the bandwidth of the PLL.

Due to the low operating frequency of Xilinx Spartan-6 FPGA, it is difficult to directly implement the TAF-DPS circuit on the final working frequency range of 81.25 and 650MHz. Therefore, the architecture of TAF-DPS feeding PLL is utilized to boost the frequency in a cascaded fashion. As shown, a triangular modulation profile of ~3.3% depth is initially created from TAF-DPS circuit, centered on ~0.6MHz. The first stage PLL boosts its center to ~50MHz while preserving the depth at 3.3%. The second stage PLL boosts the frequency to the final working range, centered on 81.25 and 650MHz, respectively. During this chain of frequency boosting, the modulation depths at points A, B, C and D are all preserved at 3.3%. On the right-hand side, the measured frequency trends at various points are shown. As seen, the modulation pattern and modulation depth are well preserved among all the points in the chain.

**E. TAF-DPS FOR FREQUENCY LOCKED LOOP**

Frequency locked loop (FLL) is a system that generates a signal whose frequency is locked to the frequency of a reference signal. FLL is a key component in several fields, such as radio, telecommunication and computer. It can be used to generate a stable frequency, or to recover a signal from a noisy communication channel. It is also widely used in many other applications such as frequency measurement, power grid synchronization, frequency stabilization, FM spectroscopy, transceiver RF synchronization and etc. For FLL design, analog approach has been traditionally used. With the advance of digital technology, more and more digital elements have been incorporated into the FLL design for performing certain functions traditionally achieved by analog circuits. However, all the existing FLL architectures, being called digital or not, have some kind of analog circuitries used in their structures. In the emerging resource-constrained IoT and edge computing, it is desirable to have a 100% pure digital FLL. This can be valuable for a variety of applications where the analog approach is not feasible or economical.

In all current FLL designs, the frequency oscillator is analog (although its frequency tuning can be digitally controlled in some cases). TAF-DPS can be used to replace this analog oscillator and enable the whole loop being 100% digital. As known, TAF-DPS has the features of AFG and IFS. Further, its output frequency can be monotonically changed following the pattern defined in a digital word (the frequency control F). Moreover, its frequency switching speed is quantifiable in terms of cycles. These facts make TAF-DPS a suitable component for functioning as digitally controlled oscillator (DCO). Combining with a binary detector and a digital control & filter block, a true 100% digital FLL can be created. It is called TAF-FLL. Instead of conventional frequency, TAF-FLL works on TAF. It has following characteristics [32].

- Output jitter and loop design are two separate issues that can be treated independently.
• The difference between integer-N and fractional-N is insignificant in TAF-FLL due to the use of TAF.
• The response speeds of all the loop components are quantifiable in terms of DCO cycles. Hence, loop dynamic can be quantitatively analyzed and precisely predicted.
• Binary frequency detector works naturally with TAF, which makes the loop design simple and robust.
• The whole loop is 100% digitally implementable.

Figure 26 is the generic frequency-lock architecture working on TAF. It is a feedback loop consisting of a frequency detector (FD), a control block for receiving the FD output and generating a control word F, a DCO made of TAF-DPS and a divider of ratio N.n where N is an integer and n is a fraction. Together, those components form a closed loop for searching an appropriate value for control word F*.

This F* is required to make (N.n)·f_0 = 1/(F*·Δ) = 1/(I* + r*)·Δ true. The DCO’s response time is two f_0 cycles. The response times of FD, CNTL and divider are all quantifiable in unit of f_0 (or f_0) cycle. The latency of this loop therefore is therefore quantifiable.

Virtually, the loop is used to implement a search algorithm, as shown in the right side of the figure. The aim of this loop is to make T_1 = T_0 = (N.n)·T_0 = (N + p/q)/F*(Δ), where N is expressed as n = p/q, p and q are integers and 0 ≤ p < q, gcd{p, q} = 1. This can further be expressed as T_0 = F**·Δ where F** = F*N + F*p/q = I* + r**. Both f_0 and f_0 therefore are in TAF. This leads to the fact that TAF-FLL blurs the difference between integer-N and fractional-N structures since T_0 = F**·Δ = (F*N + F*p/q)·Δ = (I* + r**)·Δ contains both integer and fraction parts whether p = 0 or not. For this reason, the loop treats integer-N and fractional-N indifferently.

Figure 27 depicts a TAF-FLL architecture. One of its primary goals is to make the loop respond quickly to the input change. It is desired the latency of the action-loop “frequency detection → control generation → frequency change → repeat” to be as small as possible. To achieve this, the key is to use a fast FD. For this reason, the Alexander detector is used. It uses a three-point-sampling technique to make early-late decision between two compared signals.

The input signal f_1 is first divided by two in frequency so that it can be treated as “data” (similar to the case of CDR). Since the Alexander detector requires the sampling clock being 50% duty cycle (as both the rising and falling edges are used), an additional divide-by-2 process is applied to both the f_1 and f_0 as shown. Three transition edges that make up one cycle of f_0/2 are used to sample the “data”. Early-late decision is made in every f_0/2 cycle. The time required for TAF-DCO update is two f_0 cycles. The CNTL block is driven by f_0 and its circuit can be designed very fast, taking only one f_0 cycle. Thus, the latency of the whole loop is quantifiable. The loop latency L can be expressed as L = (2 + 3/N)/2 in unit of f_0/2 (the loop sample rate is f_0/2).

The TAF-FLL has been implemented on the Kintex-7 FPGA for evaluation. Fig. 28 shows the captured frequency vs time trend when the input has a step change from 19.318 MHz to 35.417 MHz. The ratio is set as N = 4. The same frequency-step-change test has been performed on both the TAF-FLL and the on-chip MMCM. A frequency counter is used for measuring the frequency in real time. On the left is the case of TAF-FLL while MMCM is on the right. For TAF-FLL, its output steadily climbs from the starting frequency to the destination. The frequencies are realized in TAF fashion. The designed ratio of N = 4 is accurately achieved after the lock. In comparison, the same frequency step is applied to the MMCM. Its behavior is displayed on the right-hand side. Significant ringing with a large overshot is observed in the transition region. Fig. 29 shows the test result on frequency accuracy. The frequency from a TAF-FLL output is measured. The input is still the 19.44 MHz pulse train. In this case, however, the ratio is set as N.n = 6.0078125. The input and output of the TAF-FLL are connected to the frequency counter’s CH1 and CH2, respectively. The ratio of CH2/CH1 is displayed. As shown, a high frequency accuracy has been achieved. The frequency error is bounded within ± 2 ppb.

V. ARCHITECTURES OF SECURITY DATA

In this section, two architectures for producing data for security purpose are described. They all utilize TAF-DPS’s strong capability in synthesizing various waveforms.

A. TAF-DPS AND RANDOM NUMBER GENERATOR

Security has become one of the major concerns with the explosion of connected devices and the advent of cloud computing, edge computing and IoT. High entropy random number source is needed in many cryptographic applications since it is an essential component to provide secret keys or tokens for the cryptographic algorithms used to guarantee information security. Unpredictability must originate from hardware where the attributes can be physical noise, structure defect or manufacture imperfection. True random number generator (TRNG) is referred to such devices where unpredictability and randomness come from one or more such hardware attributes. In decades of effort, a variety of deterministic and hybrid TRNGs have been proposed.

The entropy harvesting mechanism is categorized into two types: voltage-domain of high/low and time-domain of early/late. In voltage-domain, the conventional method is to amplify noise directly with a high-gain and high-bandwidth amplifier followed by quantization. Another type is to utilize metastability, which is typically provided by a pair of cross-coupled inverters. It offers good operating speed and power efficiency, but often requires sophisticated design and runtime calibration to remove systematic mismatch in devices. In time-domain harvesting, the principle is to utilize the jitter in electrical oscillator’s output. This approach has a relatively low amount of entropy. Also, this kind of design is vulnerable to PVT variation and power supply attacks since the quality of jitter is influenced by those factors greatly.
Recently, an entropy-generation-mechanism based on the frequency-interaction of multiple frequency sources of different frequencies is suggested in [37]. Its working principle is depicted in Fig. 30. There are two schemes of generating randomness from frequency interaction. The first one, shown in the left, uses a frequency mixing method. The waveforms from multiple sources, each with its unique frequency, are XORed by a XOR tree. Due to the different frequencies of those waveforms, the output waveform from the XOR tree bear an irregular shape. It is then sampled by a clock, resulting in a stream of random data. This working principle is illustrated in Fig. 31.

The second scheme, shown in the right-hand of Fig. 30, utilizes an approach of frequency-tracking. A free-run oscillator is frequency tracked by another oscillator through a frequency locked loop. A bang-bang type detector is used as the frequency comparator. Its output is a random bitstream since one oscillator is tracking the other in real time.

Unlike the entropy-collection mechanisms reported in all previous works, the mechanism discussed here is frequency interaction, which is inherently a mathematical operation but is influenced by many factors that are physically random and unpredictable. The distinguishing features of the frequency-interaction-based architectures include fully digital, highly programmable, robust against PVT and etc. In implementation, the effectiveness of those two schemes depends on the capability of generating many frequencies in a cost-efficient way. An ample supply of frequency from a low-cost circuitry is the key. This requirement can be fulfilled by TAF-DPS.

The two TAF-DPS TRNGs have been implemented on a Kintex-7 FPGA system. Many random number series have been generated from both schemes. They consistently pass all the NIST tests. A test result for frequency mixing is shown in Fig. 32. For more details, please refer to [37].

B. TAF-DPS AND PHYSICAL UNCLONABLE FUNCTION

Security problem is especially exacerbated by the ubiquity of IoT devices and complicated user-case scenarios. In recent years, a trend shift from software security to hardware security is increasingly strengthened in a wide range of applications. In increasingly hardware-central territory, Physical Unclonable Function (PUF) has emerged as a foundational IP. The term PUF describes a diversity of circuit topologies developed to extract the parametric mismatches from the fabrication of silicon devices for security purposes. The variation caused by those uncontrollable deviations in the manufacturing process is referred as MPV (manufacturing process variation). It is unique from chip to chip, even all manufactured from the same mask set. The imprint of this type of variation is permanent. The random nature of this variation makes PUF practically very hard to be cloned. Further, the sensitivity of the imprint to physical probing renders PUF to be tamper-resistant against invasive attacks.

The most popular circuit architectures for exploiting those non-idealities can be classified into two major styles: memory-based and delay-based. TAF-DPS provides a new perspective for devising PUF that is different from all the existing PUF architectures by introducing a hardware-based temporal encryption of spatial imprints into the PUF response. This class of PUFs is called TeS-PUF, meaning “temporally encoded spatial imprints” PUF.
Instead of using individual bits as output (as done in all existing PUFs), the response from TeS-PUF is a bitstream made of temporally-related bits. From a plurality of pulse trains generated from a MORO, a group of stimulants is created that can be used later for establishing temporal order among the bits in the response bitstream. The temporal order is materialized by the TAF-DPS, which is used to synthesize a pulse train whose characteristic-of-waveform is controlled by a frequency control word $F$ and a starting address $A_F$. This pulse train is then converted into a bitstream as PUF output. During the process of constructing the continuous waveform of this series of pulses, spatial MPV noises are collected. More significantly, those MPV noises are temporally encrypted by the temporal regulation configured by the pair of $\{F, A_F\}$. This inclusion of temporal encryption on spatial MPV imprints is not reported in previous PUF designs.

Figure 33 is the model of TeS-PUF. The K signals from a MORO serve as stimulants for activating the TeS-PUF. Through a delivery channel, the stimulants are fed into a TAF-DPS for producing a series of electrical pulses. The “clock jitter to bit converter (JBC)”, together with the TAF-DPS, functions as an entropy harvester. The manufacturability-induced clock jitter, resulted from this waveform synthesis process, is implanted in this pulse train and is subsequently converted into a bitstream as PUF output. This jitter is caused by the non-perfection in the manufacture. Hence, it is deterministic (but unpredictable). The circuit is designed and laid out in such a way that this deterministic jitter is intentionally amplified so that it overlows the entire jitter spectrum. The generated bitstream bears the imprint of the chip and this circuit can thus function as a PUF.

Figure 34 provides a simulation result for a TeS-PUF of $K=32$ with a setting of $F=11$ and $A_F=0$. The theoretical value for all the synthesized pulses in this case is designated as $T_{\text{TAF}} = F \cdot \Delta = 11 \Delta$. There are three mismatch scenarios simulated (representing three different manufacturing results of the same design). For each case, a consecutive 32 pulses are synthesized.

VI. THE PERSPECTIVE OF FLEXIBLE CLOCKING FOR EMERGING APPLICATIONS AND THE FUTURE

Cloud computing is the on-demand availability of computer resources (e.g., data storage and computing power) to user, without requiring active management from the user. This term cloud computing generally associates with data centers. Nowadays, large clouds often have functions distributed over multiple locations. They rely on resource allocation to achieve coherence and economies of scale. The availability of high-capacity networks, low-cost computers and storages, as well as the widespread adoption of hardware virtualization, service-oriented architecture, has led to the explosive growth of cloud computing.

On the other hand, the number of smart devices newly connected to the Internet has grown exponentially in recent years. These smart devices make up the so-called Internet of Things. Along with this trend, new challenges emerge: the contradiction between mass data transmission and limited communication bandwidth, the large distance between supercomputing power and object-to-be-processed, and the demand of frequent interaction requiring real-time response. As a new computing paradigm, edge computing comes into the picture by processing tasks on computing resources close to data sources.

For cloud computing in data center, despite the improvement on network technology, transfer rate and response time are not guaranteed. But those issues are critical to many applications. The massive amount of data produced by increased IoT edge devices is pushing network bandwidth to the limit. The cloud constantly consumes data from edge devices, forcing people to build content delivery network to decentralize data and service provisioning. To address those
issues, the aim of edge computing is to leverage physical proximity to end user. It moves computation away from the data centers towards the edge of the network, exploiting smart objects, mobile phones, or network gateways to perform tasks and provide services on behalf of the cloud. By this move, it is beneficial to content caching, service delivery, persistent data storage, and IoT management. This shift to the edge can result in better response time and transfer rate. A good example for illustrating the power of edge computing and IoT is the so-called Internet of Vehicles (IoV). In [38], edge computing based video pre-processing is proposed to eliminate redundant frames. By migrating partial or all video processing task to the edge, it diminishes the computing, storage and network bandwidth requirements of the cloud center and enhances the effectiveness of video analysis. Social media services can also be moved to automobile with edge computing and IoV [39]. Mobile Edge Computing is a new development that can effectively improve the efficiency in scheduling and partitioning the tasks of computing and offloading [40]. IoT technology has been becoming a useful tool in a growing number of industries, transferring their landscapes from traditional one to modern (such as in the case of modern agriculture [41]).

Overall, the partition of responsibility between data center and edge and IoT nodes is illustrated in Fig. 35. Edge computing and IoT have their unique requirements and working environments, which are significantly different from traditional applications implemented on conventional platforms. The aim of edge computing and IoT is to leverage physical proximity. As a result, new issues arise from this scheme of distributing the responsibility to the nodes around the edge. The three major tasks performed in IoT devices are computing (data processing), sensing (assessment of the environment) and actuating (action of response). One of the most crucial subjects is the clocking for devices at the nodes. Due to the omnipresent and large-quantity, devices at the nodes are required to be small in size and low in power consumption. An option tailored for this goal is the elimination of the crystal timing reference, which is hard to be integrated with other functions and tends to be high power. By replacing crystal with other solutions (e.g., MEMS and BAW), however, frequency stability is suffered, especially frequency accuracy. The subject of clocking in this environment of large frequency variation is therefore a new challenge for edge computing and IoT, and other upcoming applications. In this battle, the ideology of TAF-DPS based flexible clocking can be a powerful weapon in our arsenal.

The TAF-DPS enabled architectures described in section III, IV and V can be readily used for assisting the various tasks-at-hand, as shown in Fig. 36. In nodes for computing, the TAF-DPS usage is illustrated in the left side. In this case, the working target is binary bit. There are three operations in this regard: bit manipulation for computation, bit transportation for communication and bit generation for security. The TAF-DPS architectures, TAF-DPS FIFO, TAF-DPS CDR, TAF-DPS DFS, TAF-DPS SSCG, TAF-DPS TRNG, TAF-DPS PUF, can be used to handle the various problems in this environment where a large frequency variation is expected. For nodes targeting at sensing and actuating, the applicability of TAF-DPS lies on its power in manipulating pulse. The plan is depicted in the right-hand side of Fig. 36. In this realm, the TAF-DPS is not used for generating clock signal but for creating various waveforms. Those waveforms are used to, through voltage/current and time, interact with a variety of
sensors and actuators. The ultimate goal is to sense or control the change in the Nature: phenomena of temperature, position, pressure, time-of-flight and etc. For those duties, the TAF-DPS PWM, TAF-DPS FLL and TAF-DPS chip generator can be the handy tools. They are convenient in implementation and low-cost in operation, which are the critical prerequisites for being useful in those resource-constrained applications. In short, when applying TAF-DPS in applications, it is used in two ways: as clock to drive circuit or as pulse synthesizer to devise circuit. The two roles are marked in the left and right of Fig. 36, respectively.

In Fig. 37, as an example, a detailed drawing is presented for illustrating the use of TAF-DPS in an environment where a large frequency variation exists due to the lack of a decent timing reference. In either system A or B, or both, the timing reference is crystal-less, leading to a large uncertainty in the frequency values of the clocks. The TAF-CDR and TAF-DPS FIFO can be handily used here to deal with the challenges. In this environment, TAF-DPS becomes an essential piece of the overall scheme in assisting the baseband signal processing.

The introduction of the TAF concept, the two features of AFG and IFS established in TAF-DPS, and the system-level architectures enabled by TAF-DPS, have collectively brewed a new perspective. This perspective is denoted as flexible clocking ideology. This chain of reasoning is developed along the path shown in Fig. 38. In contrast to the conventional fixed-frequency clocking where the clock signal is only used to maintain a simple and rigid flow-of-time, flexible clocking aims at using clock frequency as a weapon for dealing with a dynamically-changing work environment. The ultimate goal is to improve information processing efficiency while, at the same time, to reduce resource usage. This is mandatory for future advance.

VII. CONCLUSION

Problems in emerging applications present new challenges to electronic design. They demand innovations from all the directions, especially at circuit and system levels. A point of breakthrough is the circuit clocking. In this paper, the TAF-DPS technology is discussed in addressing those challenges from the aspect of clocking. Several novel TAF-DPS enabled system-level architectures are described for handling a variety of problems. From the discussions, a perspective of flexible clocking ideology emerges. Although the discussions are carried out around the issue of large frequency variation, this perspective is however not limited to this scope. It has a much broader impact on many upcoming applications. The aim of this work is to make the chip design community aware of this subject of ever-increasing significance so that the whole body can be prepared for the future.

REFERENCES

[1] A. Khashaba, A. Elkholy, K. M. Megawer, M. G. Ahmed and P. K. Hanumolu, "A Low-Noise Frequency Synthesizer Using Multiphase Generation and Combining Techniques," IEEE J. Solid-St. Circ., vol. 55, no. 3, pp. 592-601, March 2020, 10.1109/JSSC.2019.2951384.

[2] H. Sun, K. Sobue, K. Hamashita, T. Anand and U. Moon, "A 951-fs rms Period Jitter 3.2% Modulation Range in-Band Modulation Spread-Spectrum Clock Generator," IEEE J. Solid-St. Circ., vol. 55, no. 2, pp. 426-438, Feb. 2020, 10.1109/JSSC.2019.2952852.

[3] Z. Zhang, G. Zhu and C. Patrick Yue, "A 0.65-V 12–16-GHz Sub-Sampling PLL With 56.4-fs rms Integrated Jitter and −256.4-dB FoM," IEEE J. Solid-St. Circ., vol. 55, no. 6, pp. 1665-1683, June 2020, 10.1109/JSSC.2020.2967562.

[4] S. Yoo, S. Choi, Y. Lee, T. Seong, Y. Lim and J. Choi, "A Low-Jitter and Low-Reference-Spur Ring-VCO-Based Injection-Locked Clock Multiplier Using a Triple-Point Background Calibrator," IEEE J. Solid-St. Circ., vol. 56, no. 1, pp. 296-309, Jan. 2021, 10.1109/JSSC.2020.2995326.

[5] D. Liao and F. F. Dai, "A Fractional-N Reference Sampling PLL With Linear Sampler and CDAC Based Fractional Spur Cancellation," IEEE J. Solid-St. Circ., vol. 56, no. 3, pp. 694-704, March 2021, 10.1109/JSSC.2020.3033271.

[6] B. Razavi, "Jitter-Power Trade-Offs in PLLs," IEEE Trans. Circuits Syst. I Regul. Pap., vol. 68, no. 4, pp. 1381-1387, April 2021, 10.1109/TCSI.2021.3057580.

[7] L. Wu, T. Burger, P. Schönle and Q. Huang, "A Power-Efficient Fractional-N DPLL With Phase Error Quantized in Fully Differential-Voltage Domain," IEEE J. Solid-St. Circ., vol. 56, no. 4, pp. 1254-1264, April 2021, 10.1109/JSSC.2020.3047431.

[8] Z. Yang, Y. Chen, P. -I. Mak and R. P. Martins, "A 0.003-mm2 440f6S-RMS-Jitter and −64dBc-Reference-Spur Ring-VCO-Based Type-I PLL Using a Current-Reuse Sampling Phase Detector in 28-nm CMOS," IEEE Trans. Circuits Syst. I Regul. Pap., vol. 68, no. 6, pp. 2307-2316, June 2021, 10.1109/TCSI.2021.3065462.

[9] J. -H. Seol, K. Choo, D. Blauw, D. Sylvester and T. Jang, "Reference Oversampling PLL Achieving −256-dB FoM and −78-dBc Reference Spur," IEEE J. Solid-St. Circ., vol. 56, no. 10, pp. 2993-3007, Oct. 2021, 10.1109/JSSC.2021.3089930.

[10] W. Shi, J. Cao, Q. Zhang, Y. Li and L. Xu, "Edge Computing: Vision and Challenges," IEEE Internet Things J., vol. 3, no. 5, pp. 637-640, Oct. 2016, 10.1109/JOT.2016.2579198.

[11] W. S. Shi, X. Z. Zhang, and Y. F. Wang, "Edge computing: State-of-the-art and future directions," J. Comput. Res. Develop., vol. 56, no. 1, pp. 1-21, 2019, 10.7544/issn1000-1239.2019.20180760.

[12] W. Yu et al., "A Survey on the Edge Computing for the Internet of Things," IEEE Access, vol. 6, pp. 6900-6919, 2018, 10.1109/ACCESS.2017.2778504.
K. Cao, Y. Liu, G. Meng and Q. Sun, "An Overview on Edge Computing Research," IEEE Access, vol. 8, pp. 85714-85728, 2020, 10.1109/ACCESS.2020.2991734.

J. Hu, C. Chen, L. Cai, M. R. Khosravi, Q. Pei and S. Wan, "UAV-Assisted Vehicular Edge Computing for the 6G Internet of Things: Architecture, Intelligence, and Challenges," in IEEE Communications Standards Magazine, vol. 5, no. 2, pp. 12-18, June 2021.

K. Peng, M. Li, H. Huang, C. Wang, S. Wan and K. -K. R. Choo, "Security Challenges and Opportunities for Smart Contracts in Internet of Things: A Survey," in IEEE Internet of Things Journal, vol. 8, no. 15, pp. 12004-12020, 1 Aug.1, 2021.

J. Liu et al., "A 0.3-V all digital crystal-less clock generator for energy harvester applications," 2012 IEEE A-SSCC, Kobe, Japan, 2012, pp. 117-120.

H. Luo, D. Liu, C. Gong and M. Li, "A digital-crystal-less clock generation scheme for wireless biomedical implants," 2016 IEEE BioCAS, Shanghai, China, 2016, pp. 480-483.

M. Coustans, F. Krummenacher, M. Kayal and C. Terrier, "A nA crystal - less oscillator for internet of things," 2016 MIXDES, Lodz, Poland, 2016, pp. 136-139.

P. E. Bellasi and L. Benini, "Smart Energy-Efficient Clock Synthesizer for Duty-Cycled Sensor SoCs in 65 nm/28nm CMOS," IEEE Trans. Circuits Syst. I Regul. Pap., vol. 64, no. 9, pp. 2322-2333, Sept. 2017, 10.1109/TCI.2017.2694322.

D. Griffith, P. T. Ri'ine, T. Kallerud, B. Goodlin, Z. Hughes and E. T. Yen, "A ±10ppm –40 to 125°C BAW-based crystal system for crystal-less wireless sensor nodes," 2017 IEEE ISCAS, Baltimore, MD, USA, 2017, pp. 1-4.

J. Hwang, G. Jeong, S. Chu, W. Kim, T. Kim and D. Jeong, "A Crystal-Less Programmable Clock Generator with RC-LC Hybrid Oscillator for GHz Applications in 14 nm FinFET CMOS," 2018 IEEE BCICTS, San Diego, CA, USA, 2018, pp. 263-266.

L. Xiu, "The concept of time-average-frequency and mathematical analysis of flying-adder frequency synthesis architecture," IEEE Circuits Syst. Mag., vol. 8, no. 3, pp. 27-51, Third Quarter 2008, 10.1109/MCAS.2008.928421.

L. Xiu, Nanometer Frequency Synthesis Beyond the Phase-Locked Loop. Piscataway, NJ 08854, USA: John Wiley IEEE-press, 2012.

Circuit and method for adaptive clock generation using dynamic-time-average-frequency, by Liming Xiu (2015, Aug 25). Patent 9 118 275.

Microelectronic system using time-average-frequency clock signal as its timekeeper, by Liming Xiu (2015, Sep 22). Patent 9 143 139.

L. Xiu, "Clock Technology: The Next Frontier," IEEE Circuits Syst. Mag., vol. 17, no. 2, pp. 27-46, Secondquarter 2017, 10.1109/MCAS.2017.2689519.

H. Mair and Liming Xiu, "An architecture of high-performance frequency and phase synthesis," IEEE J. Solid-St. Circ., vol. 35, no. 6, pp. 835-846, June 2000, 10.1109/65.845187.

Liming Xiu and Zhihong You, "A "flying-adder" architecture of frequency and phase synthesis with scalability," IEEE Trans. Very Large Scale Integr. VLSI Syst., vol. 10, no. 5, pp. 637-649, Oct. 2002, 10.1109/TVPWS.2002.801607.

L. Xiu, K. Lin and M. Lin, "The Impact of Input-Mismatch on Flying-Adder Direct Period Synthesizer Output Jitter," IEEE Trans. Circuits Syst. I Regul. Pap., vol. 59, no. 9, pp. 1942-1951, Sept. 2012, 10.1109/TCI.2011.2180448.

L. Xiu, W. Lin and T. Lee, "Flying-Adder Fractional Divider Based Integer-N PLL: 2nd Generation FAPLL as On-Chip Frequency Generator for SoC," IEEE J. Solid-St. Circ., vol. 48, no. 2, pp. 441-455, Feb. 2013, 10.1109/JSSC.2012.2223931.

L. Xiu and P. Chen, "A Reconfigurable TAF-DPS Frequency Synthesizer on FPGA Achieving 2 ppb Frequency Granularity and Two-Cycle Switching Speed," IEEE Trans. Ind. Electron., vol. 64, no. 2, pp. 1233-1240, Feb. 2017, 10.1109/TIE.2016.2611490.

L. Xiu, X. Wei and Y. Ma, "A Full Digital Fractional- SNS TAF-PLL for Digital Applications: Demonstration of the Principle of a Frequency-Locked Loop Built on Time-Average-Frequency," IEEE Trans. Very Large Scale Integr. VLSI Syst., vol. 27, no. 3, pp. 524-534, March 2019, 10.1109/TVLSI.2018.2888625.

L. Xiu, "From Frequency to Time-Average-Frequency: A Paradigm Shift in the Design of Electronic system," May 2015, John Wiley IEEE press.

L. Xiu, "Direct Period Synthesis for Achieving Sub-PPM Frequency Resolution Through Time Average Frequency: The Principle, The Experimental Demonstration, and Its Application in Digital Communication," IEEE Trans. Very Large Scale Integr. VLSI Syst., vol. 23, no. 7, pp. 1335-1344, July 2015, 10.1109/TVLSI.2014.2334591.

MPEG-2 transport stream packet synchronizer, by Y. Yang and T. Haider (Aug 12, 2021), Patent 8 249 171.

Y. Ma, X. Wei and L. Xiu, "A Novel Spread Spectrum Clock Generation Technique: Always-On Boundary Spread SSCG," IEEE Trans. Electromagn. Compat., vol. 62, no. 2, pp. 364-376, April 2020, 10.1109/TEMC.2019.2901953.

X. Wei and L. Xiu, "An All Digital Highly Programable TAF-DPS Based True Random Number Generator Working on Principles of Frequency-Mixing and Frequency-Tracking," 2020 IEEE ISCAS, Seville, Spain, 2020, pp. 1-5, 10.1109/ISCAS45731.2020.9180408.

S Wan, S Ding, C Chen, "Edge computing enabled video segmentation for real-time traffic monitoring in internet of vehicles," Pattern Recognition, 2022 – Elsevier.

X. Xu et al., "Edge Server Quantification and Placement for Offloading Social Media Services in Industrial Cognitive IoV," in IEEE Transactions on Industrial Informatics, vol. 17, no. 4, pp. 2910-2918, April 2021.

C. Chen, Y. Zeng, H. Li, Y. Liu and S. Wan, "A Multi-hop Task Offloading Decision Model in MEC-enabled Internet of Vehicles," in IEEE Internet of Things Journal, doi: 10.1109/JIOT.2022.3143529.

A. D. Boursianis et al., "Smart Irrigation System for Precision Agriculture—The ARETHOU5A IoT Platform," in IEEE Sensors Journal, vol. 21, no. 16, pp. 17539-17547, 15 Aug.15, 2021, doi: 10.1109/JSEN.2020.3033526.