Fast All-Digital Clock Frequency Adaptation Circuit for Voltage Droop Tolerance

Matthias Függer[ID], Attila Kinali[ID], *Student Member, IEEE*, Christoph Lenzen[ID], *Member, IEEE*, and Ben Wiederhake[ID], *Student Member, IEEE*

Abstract—In classical synchronous designs, supply voltage droops can be handled by accounting for them in clock margins. However, this results in a significant performance hit even if droops are rare. In contrast, adaptive strategies detect such potentially hazardous events and either initiate a rollback to a previous state or proactively reduce clock speed in order to prevent timing violations. The performance of such solutions critically depends on a very fast response to droops. State-of-the-art solutions incur synchronization delays in the order of several clock cycles to avoid, with sufficient probability, that the clock signal is affected by metastability. We present an all-digital circuit that can respond to droops within a fraction of a clock cycle. This is achieved by using potentially metastable measurement values to delay clock signals *while* they undergo synchronization, instead of *after* they are synchronized. The challenge is to ensure that this strategy does not lead to harmful glitches or metastable upsets within the circuit. To this end, we verify our solution by formally proving correctness. We complement our findings by simulations of a 65-nm ASIC design confirming the results of our analysis.

Index Terms—Clock generators, continuous-time digital systems, low-power design, timing circuits, voltage droop.

I. INTRODUCTION

THE SYNCHRONOUS design principle is based on the assumption that the signal propagation through the combinational logic is complete before the next active clock edge. Temperature and voltage variations lead to dynamically changing interconnect and transistor delays, and are classically alleviated by decreasing the clock frequency such that a single clock period always provides sufficient time, even in face of worst case temperature and voltage conditions. These effects, together with worst case assumptions on aging and process

variation, lead to a large frequency guard band that results in underutilization of the circuit under normal conditions.

A large fraction of the guard band is due to power supply variation. Sensitivity of gate propagation delay increases with lower V_{DD}: a 1% voltage droop results in up to 4% delay change in 90-nm technology with $V_{DD} = 0.9$ V [2]. The trend to decrease V_{DD} suggests that the situation will gain in importance for future chip generations. In [3], it was shown that a major part of the guard band is required to account for power supply noise, with more than 6% loss in attainable clock frequency for a 130-nm processor. In [4], a 12% voltage droop at 100 MHz was injected into a 45-nm microprocessor, already requiring a 16% reduction of clock frequency to account for increased critical path delay.

Several techniques for reducing the clock frequency guard bad by adapting to changing environmental conditions have been proposed. For slowly changing conditions, methods range from temperature–voltage and aging compensation [5] to process variation compensation [6]. However, these techniques typically involve significant sensing and response times that prevent their application to fast environmental changes with dynamics in the order of a single clock period. In fact, supply voltage noise, induced by switching activities with high dI/dt, was shown to have its main frequency components in the 100–300 MHz range with amplitudes around 10% [7], [8]. In contrast to the ultrahigh frequency components in the order of 10–100 GHz, which are local to the switching circuit, the high-frequency components in the 100–1000 MHz range are due to die and package LC and are global across the chip [9]. Thus, a global clock adaptation strategy is an effective countermeasure against such global high-frequency voltage droops. We address these kind of droops in this work. We now briefly summarize the existing techniques to tolerate such voltage droops.

A. Related Work

1) Handshaked Designs: Both fully asynchronous and globally asynchronous locally synchronous (GALS) design styles are inherently more robust to voltage droops than their synchronous counterparts. For synchronous designs, desynchronization techniques [16] and elastic synchronous design styles [17], based on local handshaking, have been proposed.

For systems in which local handshaking poses a too large circuit overhead, globally adaptive methods have been investigated. We will briefly summarize such approaches in the following.

2) Ring-Oscillators and Pausible Clocks: Wong *et al.* [18] observed and leveraged a beneficial correlation of the critical path delay and the clock insertion delay in the presence of

TABLE I
COMPARISON OF ADAPTIVE TECHNIQUES FOR VOLTAGE DROOP TOLERANCE (AUTHORS: FKLW)

solution	process [nm]	f_{nom} [GHz]	response time [nom. clk. cycles]	synchronization time [clk. cycles]	phase offset known
this work	15	3.33	1.25	any $n \geq 1$ possible	yes
[10]	90	2.2	1.5	not spec., < 1	yes
[11]	45	3.8	8–10	not spec.	yes
[12]	130	0.1	< 1	variable	no
[13]	22	1	> 2	1	no
[14]	28	3–4	2–3	1–2	yes
[15]	16	1.5	< 1	variable	no

droops: at the occurrence of a voltage change, both become smaller/larger at the same time, thus maintaining the clock skew. A similar beneficial effect holds if the clock period correlates with the critical path delay. Along these lines, Cortadella *et al.* [19] advocated the use of on-die ring oscillators with delay lines matched to the critical path, instead of externally generated clock signals.

A recent analysis of the above effects is given in [20], where the authors show that if both the propagation delay within the ring oscillator and the clock insertion delay are matched to the critical path delay, the clock skew can be maintained during droops. To achieve this, they propose to correlate clock insertion delay to critical path delay already during clock synthesis. Otherwise, clock trees tend to be wire dominated and critical paths gate dominated, resulting in different responses to voltage changes. Clearly, this becomes more important when clock insertion delays are large.

While Chae and Mukhopadhyay [12] did not use a matched ring oscillator, their approach is similar in spirit: a matched delay line is used to determine the clock (half) period within an oscillating circuit. Unlike in ring oscillators, the oscillation is driven by pulses (instead of transitions), and one part is clocked by a multiplied clock. The probability of potential metastable upsets is decreased with the Schmitt triggers, but metastability might still propagate through the Schmitt triggers and corrupt the clock signal [21].

Fojtik *et al.* [15] proposed fine-grained GALS SoCs with small synchronous islands, each being clocked by a pausible clock [22] with delay lines matched to its critical path. The use of small synchronous islands keeps the clock insertion delay small and thus attenuates the need for a precisely matched clock network. The pausible clocks are halted when metastability from requests crossing the clock-domain boundary needs to be resolved. This efficiently combines a communication scheme with a droop-tolerating oscillator. However, average latency between a request signal issued and it being received within a neighboring synchronous island is about 1.5 clock cycles and clock cycles may be stretched due to metastable upsets in the pausible clock.

A general drawback of the above matched delay line solutions is the difficulty to design delay lines that are tightly coupled to several critical paths, which may differ for different PVT corners. In [23] an approach is presented to generate all-digital delay lines with target characteristics. However, maximum delay mismatches are still around 5% in PVT corners, and the continuous space of all PVT scenarios is checked by testing in predefined corners. Further, all such approaches require the clocking circuit to be precisely adapted to the design; this becomes particularly problematic in face of near to tape-out changes or IP blocks, whose internals are not disclosed.

Finally, with the exception of [15], having a communication latency of 1.5 (potentially stretched) clock cycles, independent ring oscillators require synchronizers with nonnegligible latency for communication between synchronous islands. In contrast, our design allows for easy tracking of phases shifts between different clock islands; any applied phase shift is fixed after a synchronization delay of a few clock cycles, during which clock shifts accumulate to less than a clock cycle of phase difference.

3) Frequency Adaptation of Stable Clock: Another approach taken by several works is to locally or globally adapt a stable, external reference clock signal. Local clock adaptation is particularly promising in fine-grained GALS SoCs as proposed in [15] to keep insertion delays—and thus droop reaction times—small: in [15], the insertion delay is kept below 1.25 clock cycles.

Deriving these locally adaptive clocks from the same stable reference instead of generating several independent clock signals further allows for tracking their phase relations. Known phase relations can then be used to communicate with neighboring GALS islands with subclock-cycle latencies, by shifting request signals such that setup/hold times at the receiver end are respected.

In [5], the clock frequency adjustment is split into a fast and a slow adjustment. The fast adjustment is performed by switching between three PLLs, while the slow adjustment is performed by adjusting the individual PLL frequencies. As the PLL outputs are not synchronized, switching between them incurs the risk of metastability and short clock cycles.

In [10], an adaptive clocking system for a 90 nm processor running at nominal 2.2 GHz and $V_{DD} = 1.2$ V is proposed. It senses voltage droops and, via an arbiter, selects a new clock signal with an adjusted clock divisor. This technique is reported to tolerate some droops of up to 30-mV/ns slope with average 700 ps response time (about 1.5 clock cycles).

Observe that, in general, the *response time* (as reported in Table I) greatly limits the frequency of a typically sin-shaped droop that can be tolerated in the worst case: assume that a circuit is of the form that it has to sense a low V_{DD} voltage and has response time of k clock cycles with period T. Given the fact that the sin reaches its minimum after 1/4th of its period, and requiring the circuit to react roughly within this time, the worst case droops that can be tolerated must have a frequency of less than $1/(4 \cdot k \cdot T)$; rendering response time the major limiting factor of such designs.

In [11], an adaptive clocking system based on sensing droops and adjusting a fast digitally controlled oscillator (DCO) that triggers a slowly changing frequency correction is presented. Its response time is 8 to 10 clock cycles for a 45 nm processor with nominal frequency of about 3.8 GHz.

The authors state that their solution can the reduce clock frequency by 7 % per 5 ns.

In [4], a dynamic variation monitor (DVM) based on mixed gate-interconnect delay line monitoring was proposed to track delay changes in critical paths. It was applied in [13] to tolerate steep voltage droops that require fast adaptations: the authors propose to route the clock signal over delay lines that have similar voltage-delay dependencies as the critical paths. This allows automatic and fast stretching of the clock signal on a negative droop slope. The potentially hazardous compression of the clock signal on the successive positive droop slope is prevented by masking the clock output until the droop is over and clock periods are nominal again. Masking is triggered by a 2 clock cycle-delayed error signal, of which one cycle is used for synchronization. Note, however, that the two clock cycles are not with nominal clock frequency, as they are also effected by the analog droop effect. While this approach is faster than the above approaches, it still results in a control latency with additive synchronization delay, which is likely to be more than one cycle for reliable designs: While droops will likely only occur every 1000 or more clock cycles, the three orders of magnitude won in metastable upset frequency provide only a small safety-margin that will likely not compensate for a second synchronization cycle that is standard in critical designs. Furthermore, the proposed solution completely stops the clock (by masking) until the droop and cycle compression are over. Their 22-nm design, running at 1 GHz, was tested with decaying sin-shaped droops with a frequency in the order of 100 MHz up to 333 MHz and an amplitude of 10 % of nominal 1 V. Note that their solution has an advantage over circuits that have to react while V_{DD} is descending: since the circuit uses the fact that during descending V_{DD} clock cycles are stretched by analog effects, their circuit must only prevent successive compression of clock cycles when V_{DD} is rising again. Roughly, assuming a sin-shaped droop, an optimistic estimate gives that the circuit has half of the sin period to react instead of a fourth; i.e., the worst case droop that can be tolerated must have a frequency of less than $1/(T \cdot 2 \cdot 2) = 250$ MHz. In case of a second synchronization cycle, this would give a frequency of less than $1/(T \cdot 3 \cdot 2) \approx 166$ MHz.

Likewise, the design in [14] is tailored to tolerate fast, steep voltage droops. Their droop detector (DD) uses a delay line to detect droops within a clock cycle and shifts the phase by selecting a proper output from a tapped delay-locked loop (DLL). The binary detection signal is synchronized (1–2 cycles), resulting in an overall 2–3 clock cycle response time. The output clock runs at 3 to 4 GHz in 28 nm CMOS. Experiments showed that some droops of 11.5 % of nominal 1.3 V occurring within 1.5 ns, around 7.7 % voltage decrease per ns, were tolerated.

II. CONTRIBUTION

We propose a mechanism that significantly reduces the frequency guard band in absence of a voltage droop while ensuring correct operation even during frequent and steep droops.

The main idea is to remove the additive synchronization delay from the critical path in the control loop, by making use of the metastability-containing circuit design [24]: we sense V_{DD} by standard means, e.g., voltage comparators [9], and directly "compute" with the potentially metastable or unstable measurement, shifting the phase of the clock signal. After a certain number of clock cycles, chosen such that metastability has ceased with sufficiently high probability, we use the sensor values to adjust a DCO. Synchronization thus occurs *in parallel* to using the measurement values to shift the clock phase, hence does not incur any delay in reaction time. This method allows fast reaction to voltage droops by shifting the phase, and fine mid/long term adaptation by adjusting the DCO. Note that our approach does not require to completely mask (stop) the clock signal during the voltage droop; we merely decrease the frequency of the generated clock output by a known (configurable) factor.

In addition to a flip-flop-based design, we present a latch-based circuit that uses only one backpropagation rail instead of two. This reduces the necessary guard band further and makes it easier to find a drop-in replacement for the droop detection mechanism.

Table I provides a comparison of our approach to previous work.

Structure of This Article: In Section III, we introduce the formal model and problem statement and prove the correctness of a modular solution within the model. A direct circuit implementation of the proposed control algorithm is presented in Section IV, which we then simplify in Section V. In Section VI, we back our theoretical findings with VHDL and SPICE simulations.

III. FORMAL MODEL AND SOLUTION

We start with the specification of a correct frequency adaption module (FAM). We then specify a circuit, called FAM implementation (FAM-I) and show that it is a correct (implementation of a) FAM. The circuit FAM-I consists of the submodules DD, delay element (DE), and Phase Accumulator (φ); Fig. 2 shows how they are combined into the FAM-I.

All module specifications are stated as a list of input assumptions (Ix) and output constraints (Cy). A module is *correct* if it fulfills all (Cy) if all (Ix) hold.

A. Specification of Frequency Adaptation Module

The overall frequency adaptation system is formalized by a module with two input ports and one output port.

One input signal is a clock signal with a fixed nominal frequency (which can be chosen much higher than the derived system clock), the other is the supply voltage. We model the clock signal by a sequence of times $(\tau_i^{\uparrow})_{i \in \mathbb{N}}$, where $\tau_i^{\uparrow}$ corresponds to the time the ith rising input clock edge occurs; analogously, $\tau_i^{\downarrow}$ is the time of the ith falling input clock edge. The supply voltage is given by $V_{DD} : \mathbb{R}_{\geq 0} \to [V_{min}, V_{max}]$, where $V_{DD}(t)$ is the voltage at time t. We require that the input is *well behaved*.

1) *Assumption of Well-Separated Input:* The input clock fulfills

$$\tau_0^{\uparrow} \geq 0 \quad \forall i \in \mathbb{N} : \tau_{i+1}^{\uparrow} - \tau_i^{\uparrow} \in \left[T_s^-, T_s^+ \right]$$
$$\text{and} \quad \forall i \in \mathbb{N} : \tau_i^{\downarrow} - \tau_i^{\uparrow} \in \left[T_s^-/2, T_s^+/2 \right] \quad \text{(I1)}$$

where T_s^- and T_s^+ are the minimum and maximum durations of the "short" clock pulses it provides. The above
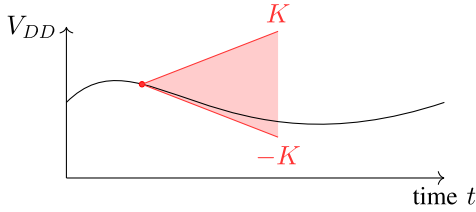
Fig. 1.   Assumption on droops: V_{DD} has maximum slope K and minimum slope $-K$. (Authors: FKLW).

essentially means a 50 % duty cycle of the input clock, although this requirement can be relaxed.

2) *Assumption on Droops:* The supply voltage satisfies that

$$\forall t, t' \geq 0 : \left| V_{\text{DD}}(t) - V_{\text{DD}}(t') \right| \leq K \left| t - t' \right| \quad \text{(I2)}$$

i.e., K bounds how steep a droop can be, see Fig. 1.

The only output is the clock signal, which during a voltage droop must slow down appropriately. We model the output by the sequence of times $(\tau_i'^{\uparrow})_{i \in \mathbb{N}}$, where $\tau_i'^{\uparrow}$ is the time the ith rising output clock edge occurs. $(\tau_i'^{\downarrow})_{i \in \mathbb{N}}$ is defined analogously. We will also need T_l^- and T_l^+, the desired minimum and maximum period of the slowed-down clock, which has "long" periods, to accommodate increased switching times during droops. Accordingly, we require that $T_s^- < T_s^+ < T_l^- < T_l^+$.

The *frequency adaptation module* is said to be *correct* if, given (I1) and (I2), it fulfills constraints (C1) and (C2).

1) *Guarantee of Well-Separated Output:* Output clock edges are well separated, i.e.,

$$\tau_0'^{\uparrow} \geq \tau_0^{\uparrow} \text{ and } \quad \forall i \in \mathbb{N} : \tau_{i+1}'^{\uparrow} - \tau_i'^{\uparrow} \geq T_s^-. \quad \text{(C1)}$$

We do not demand 50 % duty cycle of the output clock, but will show bounds for our solution later on.

2) *Guarantee of Well-Shifted Output:* We require that the output clock always runs fast when the supply voltage has been sufficiently high during the previous cycle, and that it runs slow when the supply voltage was too low during the last clock cycle

$$(\forall t \in [\tau_{i-1}'^{\uparrow}, \tau_i'^{\uparrow}] : V_{\text{DD}}(t) \geq V_{\text{high}}) \Rightarrow$$
$$\tau_i'^{\uparrow} - \tau_{i-1}'^{\uparrow} \in \left[T_s^-, T_s^+ \right] , \text{ and}$$
$$(\exists t \in [\tau_{i-1}'^{\uparrow}, \tau_i'^{\uparrow}] : V_{\text{DD}}(t) \leq V_{\text{low}}) \Rightarrow \tau_i'^{\uparrow} - \tau_{i-1}'^{\uparrow} \geq T_l^-.$$
$$\text{(C2)}$$

The voltages V_{low} and V_{high} define what is considered a droop. No implementation can work for arbitrarily close $V_{\text{low}}, V_{\text{high}}$. We will describe the minimum needed separation during the proof of correctness. In summary, $V_{\text{min}} < V_{\text{low}} < V_{\text{high}} < V_{\text{max}}$.

While this specification does not explicitly require it, the proposed system also guarantees an amortized minimum frequency of $1/T_l^+$; in the absence of metastability in the constructed delay chain, in fact no clock period is longer than T_l^+, and for a chain of length n, the maximum clock period is $T_l^+ + n(T_l^+ - T_s^+)$. Note that these requirements and guarantees, especially (C2), could be phrased differently. We attempted to capture a broad set of interpretations. Given more information about the specifics of the desired requirements and guarantees, the analysis could be tailored toward them, yielding slightly better results.

B. Frequency Adaptation Module Implementation

The FAM-I module, see Fig. 2, is inserted between the clock source and the (local) clock distribution network. This permits complementary use of the orthogonal approach of mitigating droops by exploiting delay correlation between the clock network and computational logic. Before going into detail, we give a high-level overview of the underlying idea. For now, neglect that (low-active) enable signals are duplicated in all components. During the normal operation, clock transitions are simply passed on from the Clk_{in} port to the C_O port by the phase accumulator. These transitions then bubble through the line of DEs, being delayed by a short inherent delay, until they reach Clk_{out}. When a droop happens, the DD issues its enable signal, which is latched by the last DE on the occurrence of a falling clock transition at its clock output C_O. If this enable is active, the subsequent rising clock transition at its C_I port is delayed more before reaching the clock output C_O. This "decision" is then handed to the next, left, DE by issuing its own enable signal, etc., until the decision reaches the phase accumulator, which perpetuates the phase shift by applying it between its clock input Clk_{in} and C_O for all subsequent input clock transitions. Once the droop is over, enables are released; again, this decision bubbles through the DE chain.

Care has to be taken in coping with metastability: the "decision" of a DE about the next pulse may become metastable, meaning that the storage element holding the information whether the next delay should be long or short might suffer from metastability of its storage loop. We will show that one can design DEs that might apply arbitrary delays between short and long in the case of metastability, but guarantee that the delayed clock signal does not suffer from glitches or (too) slow transitions. Duplication of enable elements is used to communicate potentially metastable enable signals consistently to the left; under no circumstances must a DE delay a clock signal, while its left neighbor does not, as this would result in a *shortened* clock cycle. In the following, we discuss this effect in detail and formally prove that our design works correctly despite metastability in DEs. To this end, we first formalize the individual components' behavior.

C. Components of the Frequency Adaptation Module Implementation

Central to our proposed solution are flip-flops with *x-masking* outputs, for $x \in \{0, 1\}$: a flip-flop whose output is x if it is internally metastable. Note that such a flip-flop only produces full-swing, fast transitions at its output, but no glitches (unless an output transition is interrupted by setting or resetting the flip-flop) or long intermediate voltage levels: when metastability resolves to $1 - x$, it produces a (possibly arbitrarily late) transition from x to $1 - x$; if metastability resolves to x, its output remains at x. Such flip-flops can be realized by successive high/low-threshold inverters; see [25], [26].

Next, we present an abstract implementation of a frequency adaptation module, called FAM-I, that consists of: 1) a DD; 2) a configurable delay chain comprising $n \geq 1$ conditional DEs; and 3) a digital phase accumulator. The three modules of FAM-I are specified and interconnected as follows (see Fig. 2).
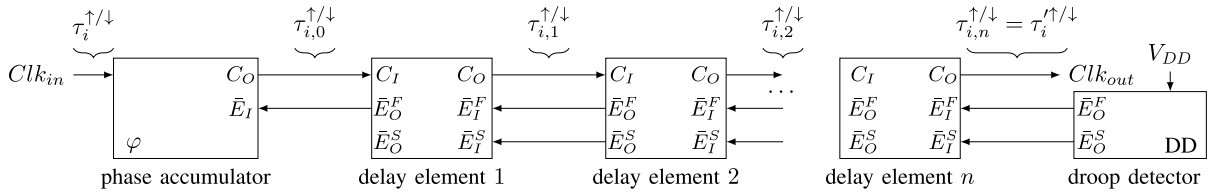
Fig. 2. System architecture of the proposed FAM-I: the DD senses occurrence of a droop, and issues delay enable signals that travel through the pipe from right to left to the φ module. Clock signals travel from left to right and are delayed accordingly at the DEs. Delay enable signals that arrive at φ are made permanent by shifting the phase of the input clock. (Authors: FKLW).

1) The *Droop Detector DD* continuously provides two single-bit digital measurement values of V_{DD} at its outputs $\bar{E}_O^F, \bar{E}_O^S$. Note that these may be unstable or transitioning when being sampled, i.e., could induce metastability of storage elements. The output is active-low, i.e., $\bar{E}_O^* = 0$ indicates the presence of a voltage-droop and the request to *slow* down the clock (i.e., have a long clock period), $\bar{E}_O^* = 1$ indicates the absence and the request for a *fast* clock (period), and $\bar{E}_O^* = M$ an unstable signal. In the case of such an unstable signal, $\bar{E}_O^F$ must be 1-masking (which we call *fast-masking*), and $\bar{E}_O^S$ 0-masking (which we call *slow-masking*). The output values are used for setting the rightmost DE.

2) Each conditional *Delay Element DE* delays the clock signal, from input C_I to output C_O, based on a possibly metastable input $\bar{E}_I^F$, the delay enable signal: if $\bar{E}_I^F = 1$ by a short delay within $[T_s^-, T_s^+]$, and if $\bar{E}_I^F = 0$ by a long delay within $[T_l^-, T_l^+]$. Potential uncertainties in $\bar{E}_I^F$ due to unstable or metastable input are transformed into delay uncertainties. Several of these building blocks are combined into a pipeline that is fed from right to left, as depicted in Fig. 2. DEs essentially shift their delay enable to the left, i.e., from $\bar{E}_I^S$ to $\bar{E}_O^*$, triggered by their local clock C_O. The chain is long enough such that a stored measurement value traveling through it from right to left is sufficiently unlikely to be unstable: the pipeline acts as a synchronizer chain.

3) The purely digital *Phase Accumulator* φ takes the oldest delay enable signal, forwarded by the leftmost DE, at its input $\bar{E}_I$, and accumulates the delay value into its phase offset. This requires that the delay enable input $\bar{E}_I$ is metastability-free at the time it arrives at the phase accumulator. The phase accumulator skips (i.e., masks) a clock cycle whenever its accumulated phase offset reaches a full period.

We continue with a detailed specification of the modules. Note that delays in all our module specifications are in terms of time ranges. This does not only allow to capture standard jitter and imbalance within the circuit, but can also account for the effect of a voltage droop on the frequency adaptation module itself. For example, a DE operating in long delay mode propagates the clock signal with delay at least T_l^- in presence of full V_{DD}, but guarantees that the delay is at most T_l^+, even in presence of a voltage droop. This allows to capture clock pulse shrinking and stretching effects caused by voltage droops as observed in [13]. For succinctness and in the interest of readability, however, we will use single variables instead of intervals for a time range in the following, with the understanding that the timing analysis has to respect the respective upper and lower bounds. For example, we write T_s instead of the interval $[T_s^-, T_s^+]$, $d \leq T_s$ instead of $d \leq T_s^-$, $d \geq T_s$ instead of $d \geq T_s^+$, and $d = T_s$ instead of $d \in [T_s^-, T_s^+]$.

We will also need the common timing parameters for what boils down to the properties of the underlying storage elements: $t_{\text{set}}, t_{\text{hold}}, t_{\text{prop}}, t_{\text{ofs}}$, which are the setup, hold, and propagation times of the circuits, as well as the offset between the active clock edge and the time the input is captured.

Module φ (Phase Accumulator): We model the behavior of module φ, as introduced in 3), in a straightforward way. The component has an internal state (the accumulated phase shift), and two inputs: 1) the single-bit signal $\bar{E}_I$ indicating whether to increase the phase offset and 2) the clock signal Clk_{in} generated by the source clock, e.g., an external free-running quartz oscillator. It outputs a clock signal C_O derived from Clk_{in}, whose pulses are phase shifted appropriately. Specifically, this means that we have to add phase-shift values, handle overflow as clock gating, and must be able to complete this within T_s^- time even during a voltage droop. As we will see in Section IV, this can be achieved by a simple and fast circuit.

Formally, let the sequences $\tau_i^\uparrow$, $\tau_i^\downarrow$, $\tau_{i,0}^\uparrow$, and $\tau_{i,0}^\downarrow$ be the times of the rising and falling edges of the input and output clock signals, respectively (the 0 indicates that φ is the "0th" element of the delay chain). We assume that (I1) holds for Module φ's clock input. By $b_{i,0}$ we denote the digital interpretation of $\bar{E}_I$ around time $\tau_{i,0}^\uparrow$, i.e., for $b \in \{0, 1\}$, $b_{i,0} = b$ if $\forall t \in [-t_{\text{set}}, t_{\text{hold}}]$: $\bar{E}_I(\tau_{i,0}^\downarrow + t_{\text{ofs}} + t) = b$ (where $\bar{E}_I$ is scaled accordingly). We assume the following.

Assumption of Metastability-Free Input: There always is such a value b, which we will argue to hold with high probability later

$$b_{i,0} \in \{0, 1\}. \tag{I3}$$

We can now define the total shift count $B_i = \sum_{k=0}^{i-1}(1 - b_{k,0})$. We say the Phase Accumulator is *correct* if, given (I1) and (I3), conditions (C3) and (C4) hold.

1) *Guarantee of Well-Shifted Output:* Let Q be the quotient of the clock period increase, i.e., $T_l/T_s = 1 + 1/Q$, and assume Q is in $\mathbb{N}$. The output clock C_O is shifted according to the amount indicated by all previous rounds' $b_{i,0}$

$$q_i = \lfloor B_i/Q \rfloor, r_i = B_i - q_i \cdot Q \text{ , and}$$
$$\tau_{i,0}^\uparrow = \tau_{i+q_i}^\uparrow + \delta_\varphi + r_i \cdot T_s/Q \tag{C3}$$

where δ_φ accounts for internal gate and wire delays of the module (like T_s, it is shorthand for an interval).

2) *Guarantee on High Time:* The high-time of each pulse in the output clock signal C_O is bounded by

$$\tau_{i,0}^\downarrow - \tau_{i,0}^\uparrow = T_s/2. \tag{C4}$$

Module Delay Element: The DE, as introduced in 2, has three inputs $\bar{E}_I^F$, $\bar{E}_I^S$, and C_I, and three corresponding outputs $\bar{E}_O^F$, $\bar{E}_O^S$, and C_O, connected like a REQ/ACK pipeline. Clock output C_O is the clock input C_I, potentially delayed by an additional up to T_s/Q time. Inputs $\bar{E}_I^*$ provide the *delay enable*, representing the (active-low) decision whether we need to delay the clock or not. Outputs $\bar{E}_O^*$ propagate this delay enable backwards in the chain, at the occurrence of the next local falling edge of C_I. We use $\bar{E}_I^F$ for the internal decision whether to add delay, whereas $\bar{E}_I^S$ is propagated to both outputs $\bar{E}_O^*$. Distinguishing between the local and forwarded "copy" of the delay enable is relevant only if the input is unstable, a case we carefully handle using metastability masking techniques.

Formally, we require that the input signal at C_I is a "clean" clock signal, i.e., it has sharp edges between periods of strong-high and strong-low signals (as we consider unstable inputs, we will have to show that this holds true in our proof of correctness); the module guarantees the same for its clock output C_O. Denote by $\tau_{i,j}^\uparrow$ and $\tau_{i,j}^\downarrow$ the sequences of times of the rising and falling *output clock edge* of the jth DE, respectively. Therefore, $\tau_{i,j-1}^*$ is the occurrence of the respective rising/falling input clock edge. Observe that $\tau_{i,j-1}^\uparrow$ and $\tau_{i,j-1}^\downarrow$ fully describe the clock input C_I to the jth element, where the first element receives $\tau_{i,0}^\uparrow$ and $\tau_{i,0}^\downarrow$ from φ. We require the following.

1) *Assumption of Well-Separated Input:*

$$\tau_{i,j-1}^\uparrow - \tau_{i-1,j-1}^\uparrow \geq T_s^- \text{ and} \tag{I4}$$

$$\tau_{i,j-1}^\downarrow - \tau_{i,j-1}^\uparrow = T_s/2 \tag{I5}$$

i.e., the clock period is T_s^- and the high time is $T_s/2$. Then, the same guarantees are ensured for the clock output.

2) *Guarantee of Well-Separated Output:*

$$\tau_{i,j}^\uparrow - \tau_{i-1,j}^\uparrow \geq T_s^- \text{ and} \tag{C5}$$

$$\tau_{i,j}^\downarrow - \tau_{i,j}^\uparrow = T_s/2. \tag{C6}$$

It remains to specify how the module responds to the delay enable inputs. To this end, for $* \in \{S, F\}$ we define $b_{i,j}^*$ as the digital abstraction of the respective signal at the input port E_I^* of the jth DE, using the mapping

$$b_{i,j}^* = \begin{cases} 0 & \forall t \in [-t_\text{set}, t_\text{hold}] : \bar{E}_I^*(\tau_{i,j}^\downarrow + t_\text{ofs} + t) = 0 \\ 1 & \forall t \in [-t_\text{set}, t_\text{hold}] : \bar{E}_I^*(\tau_{i,j}^\downarrow + t_\text{ofs} + t) = 1 \\ M, & \text{otherwise} \end{cases}$$

where we scaled $\bar{E}_I^*$ such that 1 represents a strong-high, 0 a strong-low, and M any voltage in between. Intuitively, $b_{i,j}^*$ is the resulting state of a flip-flop with input $\bar{E}_I^*$ latched at time $\tau_{i,j}^\downarrow + t_\text{ofs}$, where M represents metastability resulting from a setup/hold time violation or otherwise unclean signal.

Note that as the outputs $\bar{E}_O^*$ are fed to the module to the left, $b_{i,j-1}^*$ is given in terms of $\bar{E}_O^*$ latched at time $\tau_{i,j-1}^\downarrow + t_\text{ofs}$. With this, we can require the following.

Assumption of Proper Masking:

$$b_{i,j}^S b_{i,j}^F \in \{00, 0M, 01, M1, 11\}. \tag{I6}$$

Also, if the element adds delay, we need the guarantee that the one to the left (providing C_I as its clock output) does

the same on the next clock pulse, as otherwise we would have to choose T_s conservatively, defeating the purpose of our construction. Hence, we also demand the following.

Assumption of Delayed Input:

$$b_{i,j-1}^F = 0 \Rightarrow \tau_{i+1,j-1}^\uparrow - \tau_{i,j-1}^\uparrow \geq T_l^-. \tag{I7}$$

We now use $b_{i-1,j}^F$ to decide whether or not to delay the ith clock pulse. $b_{i-1,j}^S$, on the other hand, is used to forward the delay enable. If $b_{i-1,j}^F = M$, we are satisfied with ensuring (C1)–(C3), where (C3) is achieved by guaranteeing that $b_{i-1,j}^F = M \Rightarrow b_{i-1,j}^S = b_{i,j-1}^F = 0$ by masking metastability. If $b_{i-1,j}^S = M$, we guarantee that $b_{i-1,j}^F = 1$ by masking metastability. Both properties together [captured by (C10)] ensure that if a delay enable input causes any delay for a pulse i, then it is guaranteed to delay all following pulses by Q/T_s time, which lies at the heart of the correctness proof.

Guarantee of Delayed Output and Delay Propagation:

$$b_{i,j}^F = 1 \Rightarrow \tau_{i+1,j}^\uparrow - \tau_{i,j}^\uparrow \leq T_s^+ \tag{C7}$$

$$b_{i,j}^F = 0 \Rightarrow \tau_{i+1,j}^\uparrow - \tau_{i,j}^\uparrow \geq T_l^- \tag{C8}$$

$$b_{i,j}^S = b \in \{0, 1\} \Rightarrow b_{i+1,j-1}^S = b_{i+1,j-1}^F = b \tag{C9}$$

$$b_{i+1,j-1}^S b_{i+1,j-1}^F \in \{00, 0M, 01, M1, 11\}. \tag{C10}$$

Formally, the DE is *correct* if (C5)–(C10) hold, given that (I4)–(I7) hold.

To put constraints (C1)–(C10) in context, assume a nominal clock frequency 3.333 GHz and $Q = 4$. This yields a nominal $T_s = 300$ ps and $T_l = 375$ ps. Assuming a required minimal guaranteed high and low time for the output clock of $T_s^-/2 = 100$ ps as well as 10 % variability of the nominal frequencies, we set $T_s = [200\text{ ps}, 330\text{ ps}]$ and $T_l = [337\text{ ps}, 413\text{ ps}]$. All constraints are met for these parameters.

Module Droop Detector: Finally, we define the DD, as introduced in 1). It provides a discrete, but potentially unstable or metastable value of whether a droop has occurred; see [9] for an implementation. To enable our masking strategy, however, we use a high and a low output threshold to generate *two* signals $\bar{E}_O^*$, $* \in \{S, F\}$, which we feed as $\bar{E}_I^*$ to the rightmost DE. It is required that (C10) holds for this element; straightforward ways to ensure this is using two identical detectors with different thresholds and exploiting the assumption that V_DD changes at most at rate K, or to use a detector with (at least) three-valued output.

Moreover, the detector's output must indicate whether a voltage droop may be imminent. Accordingly, we require for a *correct* DD module that if (I2) holds then (C10) (for any $i + 1 \in \mathbb{N}$ and $j - 1 = n$), (C11) and (C12) hold.

Guarantee of Droop Detection:

$$V_\text{DD}(t) \leq V_\text{low} + T_l^- K \Rightarrow \bar{E}_O^*(t) = 0 \tag{C11}$$

$$V_\text{DD}(t) \geq V_\text{high} \Rightarrow \bar{E}_O^*(t) = 1. \tag{C12}$$

The specifics of the implementation of the detector are of no concern to us. However, note that it is crucial that the detector's delay is small, as it adds to the response time of the circuit and thus affects the steepness K of droops that can be tolerated. This suggests to favor simple implementations.

The requirements (C11) and (C12) yield that the required gap is $V_\text{high} - V_\text{low} \geq T_l^- K$.

D. Correctness of the FAM-I

To show that the FAM-I is a correct implementation of the FAM, we first prove that all input requirements of the FAM-I's submodules are fulfilled. Lemma 1 does so for the DEs in the FAM-I.

Lemma 1: Consider the FAM-I with correct implementations of its submodules and a chain of $n \geq 1$ DEs. If (I1) and (I3) hold, the input requirements (I4)–(I7) hold for each DE.

Proof: The proof is by induction on $n \geq 1$.

Induction Base ($n = 1$): We first show (I7). From $b_{i,0}^F = 0$, it follows that $B_{i+1} = B_i + 1$. First, assume that $q_i = q_{i-1}$. Then, $r_i = r_{i-1} + 1$. Thus

$$
\begin{aligned}
\tau_{i,0}^{\uparrow} &= \tau_{i+q_i}^{\uparrow} + \delta_{\varphi} + r_i \cdot T_s/Q \text{ by (C3)} \\
&= \tau_{i-1+q_{i-1}}^{\uparrow} + T_s + \delta_{\varphi} + r_{i-1} \cdot T_s/Q + T_s/Q \text{ by (I1)} \\
&= \tau_{i-1+q_{i-1}}^{\uparrow} + \delta_{\varphi} + r_{i-1} \cdot T_s/Q + (1 + 1/Q)T_s \\
&= \tau_{i-1,0}^{\uparrow} + T_l \text{ by (C3).}
\end{aligned}
\tag{1}
$$

Property (I7) follows in this case.

Otherwise, $q_i = q_{i-1} + 1$, $r_i = 0$, and $r_{i-1} = Q - 1$. Thus

$$
\begin{aligned}
\tau_{i,0}^{\uparrow} &= \tau_{i+q_i}^{\uparrow} + \delta_{\varphi} + r_i \cdot T_s/Q \text{ by (C3)} \\
&= \tau_{i-1+q_{i-1}}^{\uparrow} + 2T_s + \delta_{\varphi} \text{ by (I1)} \\
&= \tau_{i-1+q_{i-1}}^{\uparrow} + \delta_{\varphi} + r_{i-1} \cdot T_s/Q + (1 + 1/Q)T_s \\
&= \tau_{i-1,0}^{\uparrow} + T_l \text{ by (C3).}
\end{aligned}
\tag{2}
$$

Property (I7) follows also in this case.

By analogous arguments, we can show that

$$
b_{i,0}^F = 1 \Rightarrow \tau_{i+1,0}^{\uparrow} - \tau_{i,0}^{\uparrow} = T_s.
\tag{3}
$$

Property (I4) follows from (I3) and (1)–(3), (I5) follows from (C4), and (I6) follows from (C10) for the DD module.

Induction Step ($n - 1 \rightarrow n$): Assume the statement of the lemma holds for chains up to size $n - 1 \geq 1$. Assume for contradiction that the claim does not hold and consider the causally first violation; we show that such a violation is impossible. Prior to any violation, the nth element satisfies (C10), implying by the induction hypothesis that the first $n - 1$ DEs have their input requirements satisfied. Accordingly, (I4), (I5), and (I7) cannot be violated first at element n due to (C5), (C6), and (C8), respectively, for element $n - 1$. As in the base case, (I6) holds by (C10) of the DD module, which operates correctly unconditionally. We arrive at the contradiction that no input requirement can be violated first, concluding the proof. ∎

Applying Lemma 1 to the right-most DE yields property (C1) and bounds on the output clock high-time follow [by (C5) and (C6)]. An upper bound on the output clock period follows from the fact that a clock transition can be delayed by at most an additional T_s/Q per DE. As φ drops at most a $1/(Q + 1)$ fraction of the clock pulses and DEs never add or remove pulses, the amortized frequency is at least $1/T_l$.

Corollary 1: Consider the FAM-I with correct implementations of its submodules, and a chain of $n \geq 1$ DEs. If (I1) and (I3) hold, property (C1) holds and the output clock high-time is within $[T_s^-/2, T_s^+/2]$. The output clock period is at most $(1 + n/Q)T_s^+$ and amortized $(1 + 1/Q)T_s^+ = T_l^+$.

We are now ready to show that the FAM-I reacts to voltage droops as required by (C2). From Lemma 1, we already have that all DEs' input and output requirements are fulfilled; in

particular, the output guarantees of element n hold. It remains to show that the DD module correctly senses a droop and passes this on to DE n, which then reacts with a phase shift.

Lemma 2: Consider the FAM-I with correct implementations of its submodules, and a chain of $n \geq 1$ DEs. If the delay constraints $t_{\text{ofs}} \geq t_{\text{set}}$ and $t_{\text{ofs}} + t_{\text{hold}} \leq T_s/2$, (I1)–(I3) are fulfilled, then property (C2) holds.

Proof: First note that by Lemma 1, the input and output requirements of DE n are fulfilled.

We begin by showing the first implication in (C2). Assume that for all $t \in [\tau_{i-1}^{\prime\uparrow}, \tau_i^{\prime\uparrow}] = [\tau_{i-1,n}^{\uparrow}, \tau_{i,n}^{\uparrow}]$, we have $V_{\text{DD}}(t) \geq V_{\text{high}}$. From (C12), $\bar{E}_O^*(t) = 1$ for all such t. Furthermore

$$
\left[\tau_{i-1,n}^{\downarrow} + t_{\text{ofs}} - t_{\text{set}}, \tau_{i-1,n}^{\downarrow} + t_{\text{ofs}} + t_{\text{hold}}\right] \subseteq \left[\tau_{i-1,n}^{\uparrow}, \tau_{i,n}^{\uparrow}\right]
$$

follows from:

$$
\tau_{i-1,n}^{\uparrow} \leq \tau_{i-1,n}^{\downarrow} + t_{\text{ofs}} - t_{\text{set}} \text{ and}
\tag{4}
$$

$$
\tau_{i-1,n}^{\downarrow} + t_{\text{ofs}} + t_{\text{hold}} \leq \tau_{i,n}^{\uparrow}
\tag{5}
$$

where (4) follows from the delay constraints, and (5) from the delay constraints, and (C5) and (C6) for DE n. We obtain $b_{i-1,n}^* = 1$. From (C5) and (C7) for DE n, the first implication follows.

We next show the second implication in (C2). Assume that there exists a t in $[\tau_{i-1}^{\prime\uparrow}, \tau_i^{\prime\uparrow}] = [\tau_{i-1,n}^{\uparrow}, \tau_{i,n}^{\uparrow}]$ for which $V_{\text{DD}}(t) \leq V_{\text{low}}$. We distinguish between two cases: 1) $\tau_{i,n}^{\uparrow} - \tau_{i-1,n}^{\uparrow} \geq T_l^-$ and 2) $\tau_{i,n}^{\uparrow} - \tau_{i-1,n}^{\uparrow} < T_l^-$. In case of (i), the implication in (C2) holds. Otherwise, $\tau_{i,n}^{\uparrow} - \tau_{i-1,n}^{\uparrow} < T_l^-$, implying by (I2) that $V_{\text{DD}}(t) \leq V_{\text{low}} + K(\tau_{i,n}^{\uparrow} - \tau_{i-1,n}^{\uparrow}) < V_{\text{low}} + T_l^- K$ for all $t \in [\tau_{i-1}^{\prime\uparrow}, \tau_i^{\prime\uparrow}]$. Thus, (C11) yields that $\bar{E}^*(t) = 0$ during this interval. Furthermore, from the lemma's delay constraints

$$
\left[\tau_{i-1,n}^{\downarrow} + t_{\text{ofs}} - t_{\text{set}}, \tau_{i-1,n}^{\downarrow} + t_{\text{ofs}} + t_{\text{hold}}\right] \subseteq \left[\tau_{i,n}^{\uparrow}, \tau_{i,n}^{\uparrow}\right]
$$

thus, $b_{i-1,n}^* = 0$. From (C8), $\tau_{i,n}^{\uparrow} - \tau_{i-1,n}^{\uparrow} \geq T_l^-$; a contradiction to the assumption of case 2). ∎

The overall correctness follows from Corollary 1, Lemma 2, and (I3), i.e., the chain being long enough to ensure that metastability is always resolved before reaching φ.

Theorem 1: If the delay constraints $t_{\text{ofs}} \geq t_{\text{set}}$ and $t_{\text{ofs}} + t_{\text{hold}} \leq T_s/2$, (I1)–(I3) hold, then the FAM-I with correct implementations of its submodules, and a chain of $n \geq 1$ DEs, is correct.

Note that the chain length n does not influence correctness assuming that no metastability occurs, but is of course relevant to ensure (I3) indeed holds. The delay chain achieves this by acting as a synchronizer chain of length n; see Section IV.

IV. CIRCUITS

We next present circuits for the Phase Accumulator φ and the DE that fulfill the modules' specifications.

Circuit for Phase Accumulator: The phase accumulator behaves like a phase accumulator in a numerically controlled oscillator (NCO).

A natural implementation is discussed in [1], employing a premultiplied clock. It is omitted here due to space constraints. Such a design, however, has the drawback that the phase accumulator with output frequency of, say, 2 GHz must internally run a counter at a much higher frequency of 8 GHz, thereby

Fig. 3. Phase accumulator implementation (φ-DLL-I) based on tapped DLL, formed by the PD controlling a starved inverter chain, a counter, and a MUX to select the phase [14]. Note the absence of a synchronizer for the counter enable in our design variant. (Authors: FKLW).

typically representing the frequency bottleneck of the overall FAM design. In addition, we remark that one might want to run the whole frequency adaption circuit at a higher frequency than the system, as this decreases the time required to respond to a droop; dividing the output clock yields a system clock that adapts even faster to droops, while only a very small part of the circuit runs at the higher frequency.

An interesting alternative implementation of a phase accumulator is provided in [14] (Fig. 3). Their design is based on a tapped DLL and a MUX that allows to select among the taps, thereby applying the required phase shift; see Fig. 3, in which the PLL is formed by a phase detector (PD) controlling starved inverters. Such a design has the advantage of no need for a faster internal clock and thus allows higher output clock frequencies: for example, Wilcox *et al.* [14] reported 3–4 GHz in 28-nm technology. As opposed to the original design in [14], our phase accumulator implementation φ-DLL-I does not need to synchronize the delay enable signal to the input clock *Clk* of the DLL, saving two clock cycles in latency: Assumption (I3) guarantees a stable delay enable input at φ-DLL-I, which changes only with the falling clock edge of the first DE. We will later argue in Section IV-A why (I3) holds with arbitrarily high probability. Formally, we obtain the following.

Lemma 3: The circuit φ-DLL-I in Fig. 3 correctly implements Module φ for $Q = 4$.

Proof: The PLL, formed by the PD and the starved inverter chain, makes sure that the tapped inverter outputs $r \in \{0, 1, 2, 3\}$ correspond to clock *Clk* phase shifted by $2\pi r/Q$. The 2-bit counter increments modulo Q, triggered with the falling output clock edge C_O, given that the delay enable $\bar{E}_I = 0$. From input constraint (I3) we have that $\bar{E}_I$ either is stable 0 or stable 1, but not in transition while being sampled. Each counter increment results in an additional phase shift of $2\pi/Q$ for the next rising clock edge, thereby ensuring (C3). Finally, (C4) is guaranteed by the fact that phase shifts are only applied after falling output clock edges and before the occurrence of the next rising output clock edge, together with input constraint (I1). ∎

Circuit for Delay Element: Consider the circuit DE implementation (DE-I) in Fig. 4 with the pulse shaping circuit PS as depicted in Fig. 5. Concerning the flip-flops, output Q0 is required to be 0-masking (slow-masking), and output Q1 1-masking (fast-masking). We further require that the flip-flop

Fig. 4. DE-I. U5 and U6 are falling edge triggered and have setup hold times of t_{set} and t_{hold}. Outputs Q0 are slow-masking (i.e., 0-masking), and output Q1 is fast-masking (i.e., 1-masking). PS is a pulse shaping module, see Fig. 5. (Authors: FKLW).
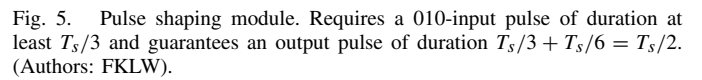

Fig. 5. Pulse shaping module. Requires a 010-input pulse of duration at least $T_s/3$ and guarantees an output pulse of duration $T_s/3 + T_s/6 = T_s/2$. (Authors: FKLW).

parameters fulfill

$$t_{\text{prop}} < T_s/2 - \delta_{DE} - (T_l - T_s) - t_{\text{ofs}} \tag{6}$$
$$t_{\text{set}} < T_s/2 - t_{\text{ofs}} \text{ , and } t_{\text{hold}} < \delta_{DE} + t_{\text{prop}}. \tag{7}$$

Fig. 6(a) visualizes the behavior of the DE-I for the case of a clear $b^F_{i-1,j} = 0$ stored in U5, caused by sampling $\bar{E}^F_I = 0$ with the previous falling edge of PS ($= C_O$). As a result, the "slow" delay path of DE-I is enabled, propagating the ith rising clock edge of C_I [❶ in Fig. 6(b)] via the inverter chain INV ❷, to U4 ❸, and finally to PS ❹. The pulse shaper then reshapes the pulse to $T_s/2$ width ❽, which triggers the flip-flop U5 to sample $\bar{E}^F_I$. In the example execution, we assume $\bar{E}^F_I = 1$ ❾, resulting in $b^F_{i,j} = 1$.

Fig. 6(b) visualizes all potential signal traces of DE-I, including those where flip-flop U5 is internally metastable ($b^F_{i-1,j} = $ M). The thick, brown/black signal trace shows an example where U5 is metastable: U5 resolves to 1 during execution causing U5 to make a late but clean transition from 0 to 1 ⒜. Observe that this triggers a delayed propagation along the short path ⒜, ⒝, ⒞, and ⒟. Also note that a metastable U5 cannot delay propagation beyond the slow path ❷, ❸, and ❹. The remainder of the execution is analogous to the case before.

Lemma 4: The circuit DE-I in Fig. 4, with U5 and U6 initialized to 1, correctly implements a DE for $Q = 4$.

Proof: We prove the claim by induction over the pulse number i, where apart from the properties (C5)–(C10), we claim that U5 and U6 attain states $b^F_{i,j}$ and $b^S_{i,j}$ when being latched by the falling outgoing clock edge; for the induction anchor at $i = 0$, we may set $b^F_{0,j} = b^S_{0,j} = 1$ by the prerequisites of the lemma.
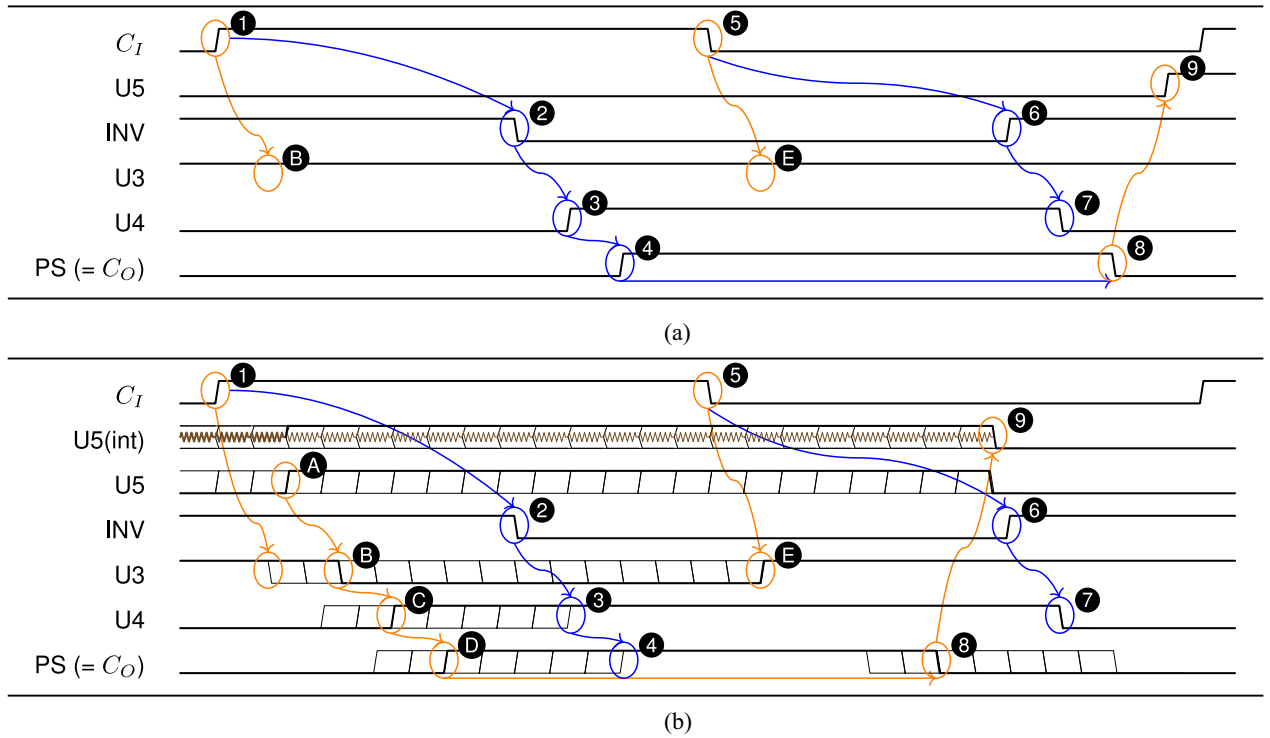
Fig. 6. Behavior of the DE-I for various behaviors of the U5 flip-flop. Signal U5(int) is the internal state of U5, INV the output of the odd inverter chain, and all other signals the output of the corresponding gate. Orange and blue lines show causal relations, brown represents the metastable internal state. We observe a total time of roughly T_s each. This is explained in more detail in Section IV. (Authors: FKLW). (a) Behavior when the U5 flip-flop stores a clear 0, meaning "slow". (b) All potential behaviors of the DE-I for an internally metastable U5 flip-flop. This can be seen as an interpolation between the slow and fast behavior. Note the 0-masking property of U5.

Now, consider the ith incoming rising clock edge; ❶ in Fig. 6(b). As U5 is a slow-masking flip-flop, in the absence of a (falling outgoing) clock edge latching it, its output can only transition from 0 to 1 Ⓐ. Hence, the rising clock edge incoming at C_I is forwarded to the pulse shaper input, with a delay between δ Ⓒ and $\delta + T_l - T_s = \delta + T_s/4$ ❸, where δ denotes the delay from C_I through U3 and U4 to the input of the pulse shaper PS. Note that the high time of the signal may be increased by up to $T_s/4$, but it will drop to low again ❼ before the next pulse arrives due to (I4). The first stage of the pulse shaper then inverts the signal, transforming the high time into a low time of $T_s/3$. Afterward, the signal is inverted again and the resulting high time extended to $T_s/3 + T_s/6 = T_s/2$ (Ⓓ to ❽), guaranteeing (C6).

We continue to show (C5). The overall delay of the ith pulse is in the range $[\delta_{DE}, \delta_{DE} + T_s/4]$, where $\delta_{DE} = \delta + \delta_{PS}$ and δ_{PS} is the delay between a rising input clock edge to the pulse shaper and the rising edge at its output. By the earliest time the falling output clock edge occurs ❽, i.e., after $\delta_{DE} + T_s/2$ time, we are guaranteed that U3 has low input from C_I again. Furthermore, until the new latching output ❾ propagated to U3's input, i.e., at latest after time $\delta_{DE} + T_s/2 + T_s/4 + t_{ofs} + t_{prop} < T_s$, U3's clock input remains 0, by the delay constraint (6) and (I4). So, latching U5 does not cause glitches at U3's output.

To prove (C5), we have to show that the next, i.e., $(i+1)$th, rising output clock edge does not occur too early. Observe that the above considerations show that the delay of the ith rising edge from the input clock to the output clock can be larger than δ_{DE} only if U5 was *not* in a stable state of 1, i.e.,

$b_{i-1,j}^F \neq 1$. We show that in this case $b_{i,j-1}^F = 0$ must hold, resulting, by (I7), in a delayed generation of the $(i+1)$th rising output clock edge of stage $j-1$ and, thus, a delayed $(i+1)$th rising input clock edge at stage j, proving (C5).

By (I6), $b_{i-1,j}^F \neq 1$ entails that $b_{i-1,j}^S = 0$. Furthermore, $b_{i,j-1}^F = 0$ is guaranteed if during $\tau_{i,j-1}^{\downarrow} + t_{ofs} + [-t_{set}, t_{hold}]$ register U5 of element $j-1$ sees a stable 0. This is guaranteed because the stable $b_{i-1,j}^S = 0$ is driven by U6 of element j since time $\tau_{i-1,j}^{\downarrow} + t_{ofs} + t_{prop} \leq \tau_{i,j}^{\downarrow} - T_s + t_{ofs} + t_{prop} \leq \tau_{i,j-1}^{\downarrow} + t_{ofs} + t_{prop} - T_s - (\delta_{DE} + T_s/4) \leq \tau_{i,j-1}^{\downarrow} + t_{ofs} - t_{set}$, by delay constraint (6). It follows that $b_{i,j-1}^F = b_{i-1,j}^S = 0$, which by (I7) guarantees that $\tau_{i+1,j-1}^{\uparrow} \geq \tau_{i,j-1}^{\uparrow} + T_s + T_s/4$. We conclude that, regardless of the state of U5, $\tau_{i,j}^{\uparrow} - \tau_{i-1,j}^{\uparrow} \geq T_s$, proving (C5) for the ith pulse.

(C7) and (C8) are immediate consequences of the above considerations for the case of U5 being in a stable state.

Concerning (C9), it follows from the already established (C5), (C6), and the delay constraints that $b_{i,j}^S = b \in \{0, 1\}$ entails that the output of U6 is stable during $[\tau_{i+1,j-1}^{\downarrow} + t_{ofs} - t_{set}, \tau_{i+1,j-1}^{\downarrow} + t_{ofs} + t_{hold}]$. Apart from showing (C9), these timing constraints also imply that U6 is not latched during this time. If U6 is internally metastable, this means that it can stabilize only in one direction (note that we require an implementation that prevents oscillatory metastability). Finally, recalling that $\bar{E}_I^F$ and $\bar{E}_I^S$ are fast and slow masking, respectively, we see that $b_{i+1,j-1}^F = 1$ or $b_{i+1,j-1}^S = 0$, proving (C10) for the ith pulse. ∎

A. Metastability

Combining Theorem 1 with Lemma 3 and 4, we obtain correctness of the FAM-I. Note, however, that correctness relies on requirement (I3). Given our circuit implementation, (I3) corresponds to the fact that the delay enable propagated through the n DEs from the DD module to the φ module is not metastable when it arrives. From the fact that stable register values are propagated correctly, i.e., again result in stable register states of the element to the left, we deduce that metastability can only propagate through the chain when the register U6 of DE j resolves exactly when register U6 of element $j-1$ latches its input; i.e., the chain acts as a synchronizer chain of length n. The overall probability of a failure can thus be bounded analogous to failure of an n-stage synchronizer; see [25], [26]. Specifically, as the chain of registers contains no logic gates, we can assume that $T_w = t_{\text{set}} + t_{\text{hold}}$ and the available metastability resolution time $T_{\text{res}} = nT_s - (n-1)T_w$.

As an example, assume worst case conditions for the DD ($f_d = f_c$). Using the values for common ASIC synchronizers ($\tau = 31.6\,\text{ps}$, $T_w = 8\,\text{ps}$) and a chain running at a high clock speed ($n = 5$, $f_c = 4\,\text{GHz}$), this achieves a good MTBF

$$T_{\text{res}} = 5 \cdot 0.25\,\text{ns} - 4 \cdot 8\,\text{ps} \geq 1.2\,\text{ns}$$

$$\text{MTBF} \geq \frac{e^{T_{\text{res}}/\tau}}{f_d f_c T_w} = \frac{e^{1.2\,\text{ns}/31.6\,\text{ps}}}{4\,\text{GHz} \cdot 4\,\text{GHz} \cdot 8\,\text{ps}} \approx 7.6\,\text{a}.$$

Note that this estimate is overly pessimistic in that the frequency of detector outputs that might induce metastability is going to be in the order of the frequency of droops, rather than the clock frequency f_c. This frequency is expected to be several orders of magnitude smaller, i.e., even without sophisticated synchronizer design fewer stages are likely to be sufficient.

We remark that, apart from the delay constraints, this is the only technology-dependent aspect of our approach. Hence, it is very easy to transfer our design to different technologies. In particular, the length of the delay chain is simply the length of a synchronizer chain of sufficient MTBF for the respective technology and application.

V. SIMPLIFIED FREQUENCY ADAPTION MODULE IMPLEMENTATION

The previous construction used two backward rails, which essentially propagate the same signal, but with different masking applied. It imposes the requirement that the DD provides two output signals, only one of which may induce metastability of the corresponding storage element when it is latched. While the constraint on the output of the detector may be straightforward to satisfy, it has negative impact on performance: to guarantee that not both capturing storage elements become metastable, the respective voltage thresholds for when the detector's outputs transition between 0 and 1 need to be sufficiently separated; however, via constraints (C11) and (C12), this entails that K (i.e., the maximum droop-steepness; compare Fig. 1) or the gap between V_{high} and V_{low} (and thus the minimum voltage under which a clock period of T_s is sufficient) becomes smaller.

Filling two needs with one deed, we can simplify the interface to the DD and resolve this performance issue at the
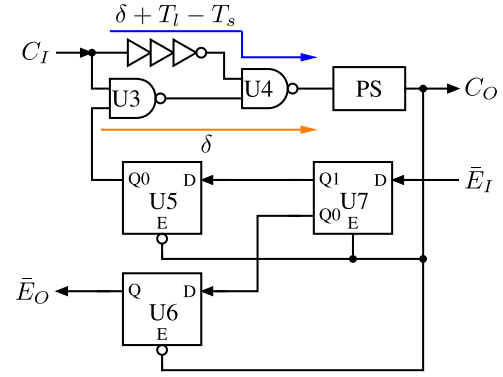

Fig. 7. sDE-I with single backward rail. U5, U6, and U7 are latches. U7 provides a fast-masking (i.e., 1-masking) output to U5 and a slow-masking (i.e., 0-masking) output to U6. The output of U5 is slow-masking, while the output of U6 needs not mask. (Authors: FKLW).

same time. The general idea is to separate the flip-flops U5 and U6 of the DE into their constituent latches, "merge" the master latches into one, and ensure the separation by exploiting that, when intransparent, the (single) master latch can only stabilize *either* to 0 or to 1 (as opposed to the two master latches of the flip-flops U5 and U6 from Fig. 4); see Fig. 7 for the resulting modified implementation. More concretely, assume that the joint master latch U7 becomes metastable when it becomes intransparent, i.e., when the respective clock transition arrives at the DE's output. This means that, after a short delay caused by the internal latency of the feedback-loop of the latch, its output voltage is (very) close to a fixed value corresponding to the unstable equilibrium state the latch is in. Any deviation from this equilibrium is amplified, resulting in stabilization to either high or low output voltage. Using standard masking techniques (e.g., high/low-threshold inverters driven by the latch output, see [26]), we can derive slow- and fast-masking outputs, taking the role of the two input signals provided to DEs in our earlier solution. When the master latch stabilizes, only one of these outputs undergoes a transition, implying that only one of the slave latches can become metastable; more precisely, we end up with the same set of possible states of the two slave latches as in the previous construction: 00, 0M, 01, M1, 11. Hence, this approach guarantees (C10) by construction, regardless of the (single) input signal provided to the modified module. We emphasize that, also here, it is critical to avoid latch implementations that can be driven into oscillatory metastability.

Fig. 8 shows the resulting simplified implementation simplified frequency adaption module implementation (sFAM-I). Note that the specification of the FAM remains identical. Correctness is shown analogously as well, where the property (C10) is not an output property of the DD and DEs anymore, but rather an invariant, which DEs ensure internally under the assumptions (I4), (I5), and (I7).

To formalize this, we adapt the specifications of the modules to match the system description given in Fig. 8.

A. Modified Specifications

In the following, all flip-flop parameters refer to the flip-flops given by the master/slave pairs U7/U5 and U7/U6, respectively, which we assume to be equal due to symmetry.
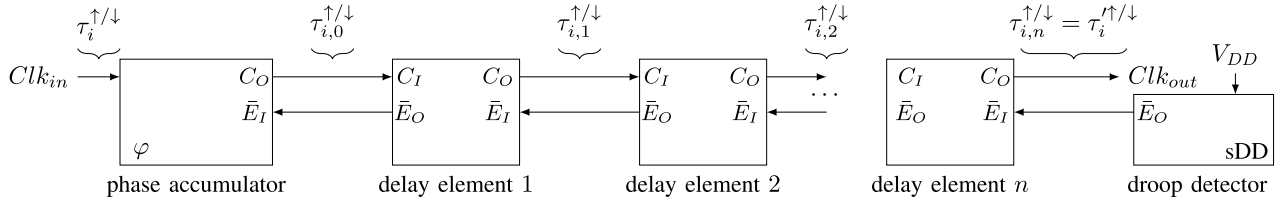
Fig. 8. sFAM-I with sDE and sDD. From [1]. (Authors: FKLW).

Module φ (Phase Accumulator): The specification of the phase accumulator remains unchanged.

Simplified Delay Element: The DE has clock input C_i and clock output C_O. It receives a delay enable input $\bar{E}_I$ and provides a delay enable output $\bar{E}_O$.

In order to specify the DE similarly to before, it is most convenient to specify $b_{i,j}^*$ similarly as well. However, these values are now derived from the same input signal $\bar{E}_I$, with metastability masking taking place entirely within the element. Accordingly, with the same definitions of $\tau_{i,j}^{\uparrow}$ and $\tau_{i,j}^{\downarrow}$ as before, we integrate (C10) into the definition

$$
b_{i,j}^F b_{i,j}^S
$$
$$
= \begin{cases} 00 & \forall t \in [-t_{\text{set}}, t_{\text{hold}}] : \bar{E}_I(\tau_{i,j}^{\downarrow} + t_{\text{ofs}} + t) = 0 \\ 11 & \forall t \in [-t_{\text{set}}, t_{\text{hold}}] : \bar{E}_I(\tau_{i,j}^{\downarrow} + t_{\text{ofs}} + t) = 1 \\ 00, 0M, 01, M1, \text{ or } 11 \text{ (arbitrarily), otherwise.} \end{cases}
$$

A correct (modified) DE then guarantees (C5)–(C9), granted that (I4), (I5), and (I7) hold.

Simplified Droop Detector: The specification of the DD is changed so that there is only a single output $\bar{E}_O$ that needs to satisfy (C11') and (C12').

Guarantee of Droop Detection:

$$
V_{\text{DD}}(t) \leq V_{\text{low}} + T_l^- K \Rightarrow \bar{E}_O(t) = 0 \qquad \text{(C11')}
$$
$$
V_{\text{DD}}(t) \geq V_{\text{high}} \Rightarrow \bar{E}_O(t) = 1. \qquad \text{(C12')}
$$

B. Proof of Correctness

Correctness of the sFAM-I given in Fig. 8 follows analogously to our reasoning for the previous variant. As we pushed (C10) into the modified definition of $b_{i,j}^*$, no modification to the proofs is necessary.

Corollary 2: If $t_{\text{ofs}} \geq t_{\text{set}}$, $t_{\text{ofs}} + t_{\text{hold}} \leq T_s/2$ and (I1)–(I3) hold, then the sFAM-I in Fig. 8 with correct implementations of its submodules φ, simplified DE (sDE), and simplified DD (sDD) (as specified in this section) and a chain of $n \geq 1$ DEs is correct.

Concerning the implementation of the modified DEs given in Fig. 7, we need to argue that indeed the masking properties are satisfied. Again, we require that the flip-flop parameters fulfill

$$
t_{\text{prop}} < T_s/2 - \delta_{DE} - (T_l - T_s) - t_{\text{ofs}}
$$
$$
t_{\text{set}} < T_s/2 - t_{\text{ofs}} \text{, and } t_{\text{hold}} < \delta_{DE} + t_{\text{prop}}.
$$

Corollary 3: The circuit simplified delay element implementation (sDE-I) in Fig. 7, with U5 to U7 initialized to 1, correctly implements a Simplified DE (sDE) for $Q = 4$.

Proof: Analogous to Lemma 4, with the exception that instead of showing (C10), we need to argue that the behavior of the DEs is consistent with the above modified definition of $b_{i,j}^F$ and $b_{i,j}^S$. The latter follows from the discussion

of the modified implementation at the beginning of the section: intuitively, (C10) now applies to the two outputs of the latch U7. ∎

Theoretical Guarantees on Tolerable Droops

To obtain a worst case guarantee on the droops that can be tolerated, first observe that in our formalism droops are only restricted by the maximum absolute slope K in (I2); see Fig. 1. The slope K itself is constrained by (C11') and (C12') of the DD. Assuming that noise up to 5% of nominal V_{DD} is not considered a droop, we set $V_{\text{high}} = 1.1\,\text{V} \cdot 0.95$. If we require the circuit to guarantee a slowdown of the clock frequency in case V_{DD} drops another p fraction of nominal V_{DD}, i.e., under $1.1\,\text{V} \cdot (0.95 - p)$, then we have to set $V_{\text{low}} = 1.1\,\text{V} \cdot (0.95 - p)$. Thus, from (C11') and (C12'), and plugging in $Q = 4$ and $T_l^- = (1 + 1/Q) \cdot 0.3\,\text{ns}$ corresponding to a design with a nominal clock frequency of $T_s^+ = T_s^- = 3.33\,\text{GHz}$,[1] we get

$$
V_{\text{low}} + T_l^- K \leq V_{\text{high}}
$$
$$
1.1\,\text{V} \cdot (0.95 - p) + (1 + 1/4) \cdot 0.3\,\text{ns} \cdot K < 1.1\,\text{V} \cdot 0.95
$$
$$
K < p \cdot 2.9\overline{3}\text{V/ns}.
$$

For example, guaranteeing a response by the time voltage falls by another $p = 5\%$ below V_{high} means that it is safe for the supply voltage to drop by about 13% per ns.

This relation shows that the tolerable K is inversely proportional to the reaction time T_l^-, which is 1.25 times the nominal clock period T_s^-, emphasizing the value of removing synchronization time from the reaction time.

We stress that this bound is a provable worst case guarantee for droop shapes that have no other restriction than their maximum slope. In practice, one might expect specific droop shapes, e.g., sinus shaped ones, for which the analysis could be refined. Moreover, recall that (C2) imposes the strong requirement to run slow if the supply voltage is below V_{low} at *any* time during the current clock cycle. In contrast, the clocked logic will still complete computation within T_s if the voltage is sufficiently high *on average* throughout the clock cycle. More careful modeling could capture such effects, further changing the constants in the above bound in our favor. On the other hand, we have not accounted for the delay incurred by distributing the clock signal to the computational logic.

However, we stress that the main takeaway from the above worst case analysis is that the reaction time is the crucial factor governing how steep the tolerated droops can be. To assess the performance to be expected in practice, we study tolerance to sinus shaped droops in the following section.

---

[1]That is, for ease of calculation we assume that the frequency fluctuations of the reference clock, which by design is highly stable, are negligible.

VI. SIMULATION

We implemented the digital circuit at gate-level. VHDL simulations with annotated timing parameters were used to validate the abstract, timed behavior of the central components, the DE, and the phase accumulator, as well as a complete design. A SPICE netlist of the circuit was then generated, including parasitics. The netlist was then slightly tuned by tweaking the delays in the φ module, the DE and the pulse shaper, by adding/removing inverters, to approximately match the target frequency of the circuit. We stress that, thanks to the quite relaxed constraints from Section III, the circuit is robust to variations in the delays as well as rise and fall times. Hence, no perfect matching of delays is required, and the circuit operates correctly under large variations, so long as the constraints are still satisfied. This is further evidenced by a stochastic variation analysis on the SPICE model, confirming that variations have only minor effects. The SPICE simulations demonstrate that the required design constraints can be met for clock frequencies between nominal clock frequencies of 1 and 3.33 GHz in 65 nm.

For synthesis, all flip-flops and gates were used from the UMC 65-nm standard cell library. DEs were modeled using chains of minimal sized inverters. Replacing the phase accumulator based on a NCO from [1] by the implementation based on a tapped DLL, φ-DLL-I from [14], the clock speed can be increased significantly, as the phase accumulator does not need to run at four times the clock frequency of the remaining circuit. Using this approach, the phase accumulator can operate at frequencies well above 4 GHz. The element limiting the clock frequency in our design thus becomes the pulse shaping module in the DEs DE-I or sDE-I, respectively. Capacitive loading of the clock output has not been considered beyond adding an appropriately sized buffer at the output, as the circuit is intended to act as the root of a local clock distribution tree. We used a *CKINVM16N* buffer as a stand-in for the clock tree.

Performance Evaluation

As a model for droops, we followed the works of Bowman *et al.* [13] and Fojtik *et al.* [15], using decaying, sin-shaped voltages. In [15] these have been identified as worst case droop shapes, resulting from sudden increase or decrease of load current, e.g., by switching on several cores in a multicore architecture.

Pushing the circuit close to its limit, we achieved a nominal clock frequency of 3.33 GHz, limited by the stability of the pulse shaping module and a droop frequency of 200 MHz. While a 200 MHz droop frequency might seem low compared to the clock frequency of 3.33 GHz, we stress that the former is not (directly) dependent on the latter, but rather a function of the complex impedance of the power grid. Specifically, the power grid will oscillate at frequencies where the Q factor of the equivalent RLC-circuit becomes large. 200 MHz has been chosen as a representative value of measurements of similar circuits as reported in (e.g., [7], [8], and [15]). Simulations for different droop frequencies have been performed and the circuit performance is the same, granted that assumption (I2) holds.
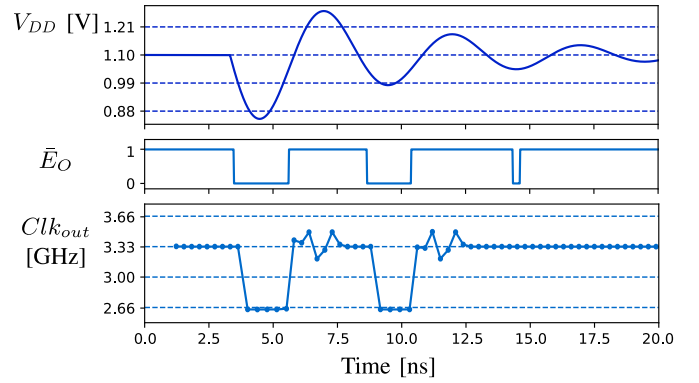

Fig. 9. SPICE simulation of a droop on the supply voltage V_{DD} and its effect on the low active delay enable signal $\bar{E}_O$ from the DD and the instantaneous frequency of the output clock Clk_{out}: (top) V_{DD} [V], (middle) enable output $\bar{E}_O$ of the DD, (bottom) instantaneous frequency of the clock output Clk_{out} [GHz]. Dashed lines are in multiples of 10 % from nominal values (1.1 V for V_{DD} and 3.33 GHz for the instantaneous frequency of Clk_{out}). Observe that the last delay enable pulse is too short to trigger a clock slowdown. This is legitimate: V_{low} can only be reached if the voltage falls below V_{high} for sufficient time to guarantee a detection. (Authors: FKLW).

Fig. 9 shows the result of a SPICE simulation with a simulated voltage droop on V_{DD} of frequency 200 MHz and an initial droop amplitude of 0.3 V or about 27 %, decaying over time. The maximum slope of the droop thus is $0.3 \, V \cdot 2\pi \cdot 200 \, MHz \approx 0.377 \, V/ns$, or about 34.3 % of nominal 1.1 V per ns. Observe that the DD's low active output, the delay enable signal $\bar{E}_O$, quickly results in a lower frequency of the output clock Clk_{out}; compare with Fig. 8 for an overview on signal names.

Fig. 10 shows the same simulation, demonstrating how the clock signal propagates to the right in Fig. 8 and the delay enable signals propagate to the left. Observe that the clock slows down immediately at the last DE by applying a phase shift. The phase shift is then propagated back toward the clock source, resulting in a consistent frequency change throughout the entire period during which the supply voltage is too low.

Fig. 11 shows the results of the variation analysis, for the two most critical elements in this regard, the pulse shaper and the DE. We varied toxe (electrical gate oxide thickness), toxp (physical gate oxide thickness), and vth0, for nfets and pfets, independently according to a normal distribution; as such the variation model is conservative as it does not assume typical correlations between the parameters. Mean values were set to the parameters from the typical corner and σ such that ss-ff corners are 3σ. As can be seen, the variations are relatively small and well within the constraints given in Section III. While we also show the delay for the pulse shaper, the critical value is the generated pulse width. It has to be long enough for the flip-flops to work correctly, yet short enough that its falling edge does not interfere with the next rising edge. Similarly, for the DE, the short delay is not as important as the difference between the delays. In our simulation, assuming deviations even up to 9σ, this difference is in [49.85 ps, 107.255 ps]. This is clearly within $T_l - T_s = [7 \, ps, 213 \, ps]$.

We have also simulated the circuit with droops with different frequencies and amplitudes, up to 500 MHz and 10 % amplitude and did not see any adverse effects or glitching, as was expected from the proofs as shown in Section V.
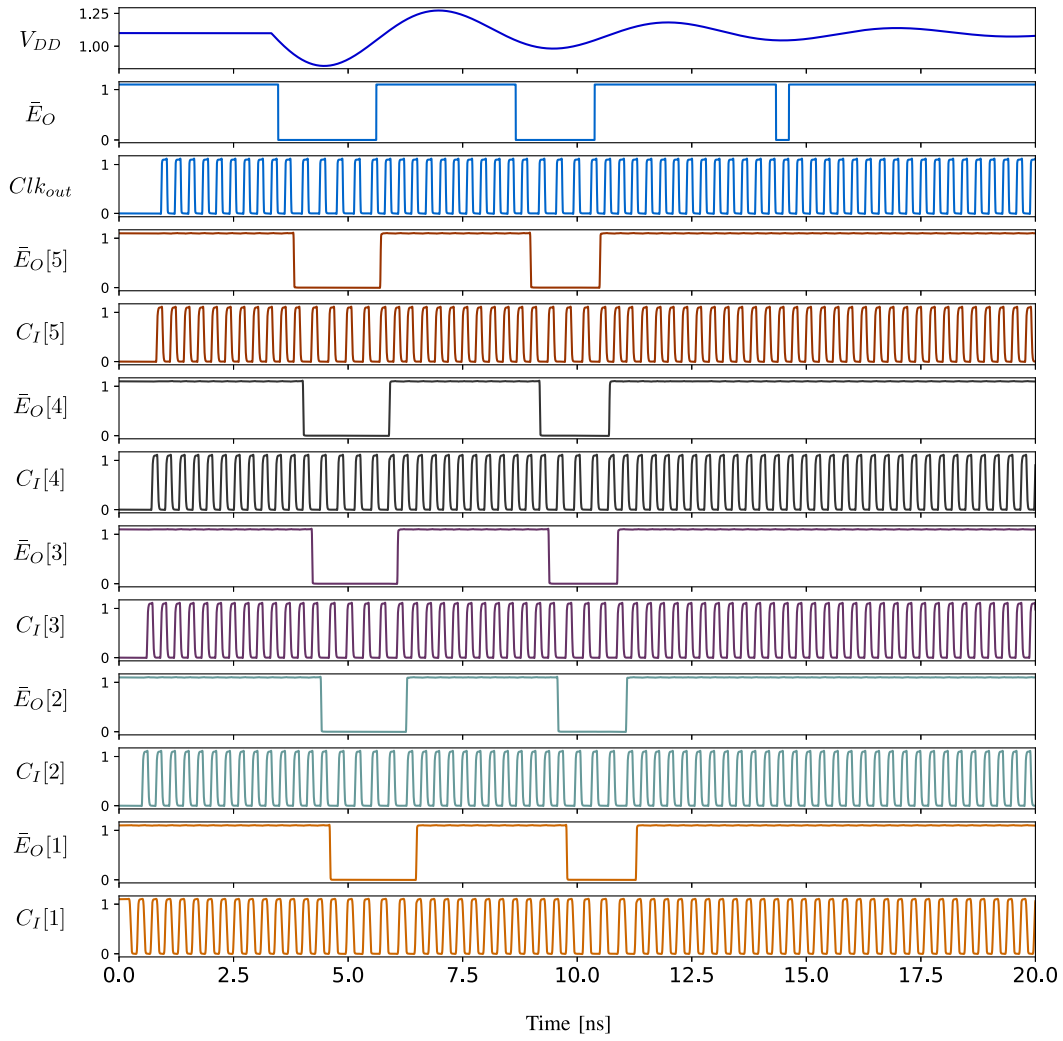
Fig. 10. Results of the SPICE simulation of the circuit using φ-DLL-I. The graphs show the supply voltage V_{DD}, the DD output $\bar{E}_O$ and pairs of delay enable and clock signals at the boundary between two DEs: 1) $C_I[1]$ and 2) $\bar{E}_O[1]$ are the clock and the delay enable signals, respectively, between the phase accumulator and the first DE, $C_I[2]$ and $\bar{E}_O[2]$ between the first DE and the second, and so forth until $C_I[5]$ and $\bar{E}_O[5]$ being between the second last and the last DE. Clk_{out} is the final clock output of the last DE. The graphs show the quick reaction time of the system to droops, well within a single clock cycle from the assertion of the delay enable, both at the start of the droop and its end. The delay enable and a "zone" of slow clock cycles trickle backward in the chain until they get absorbed by the phase accumulator. (Authors: FKLW).
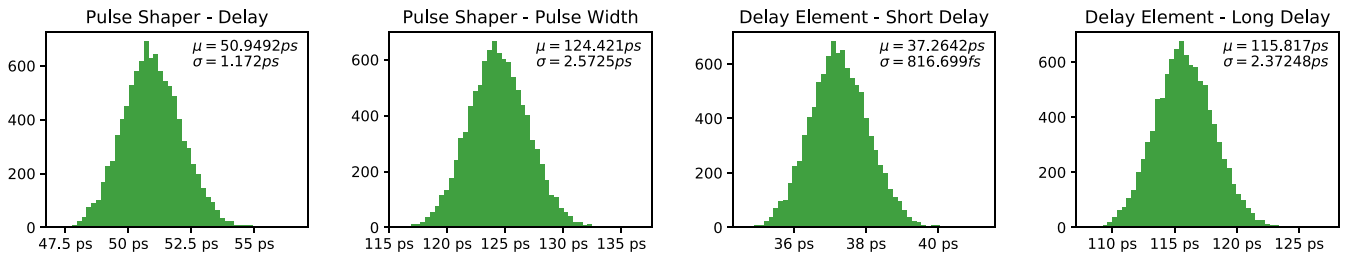

Fig. 11. Results of the variation analysis. For the pulse shaper, we show the delay through the subcircuit and the generated pulse width. For the DE we show the short delay ("no" delay) and the long delay. Observe that the minimal pulse width of $T_s^-/2 = 100\,\text{ps}$ from Section III is easily met. (Authors: FKLW).

VII. CONCLUSION

High-frequency voltage droops consume a significant fraction of the clock guard band. We proposed a circuit that allows to react to steep and high-amplitude droops, without the need to halt the clock. The circuit is based on detecting droops and propagating this information along a delay line, back to a DCO that accounts for the respective phase offset. The clock

signal travels in the opposite direction through the delay line. Care had to be taken in handling metastability: we make use of masking flip-flops, ensuring that no glitches are introduced in the clock signal.

We verified our design by correctness proofs and synthesized it in UMC 65 nm, running VHDL and SPICE simulations ranging from 1 to 3.3 GHz input clock frequency, whose

results are in accordance with our theoretical predictions. In future work, we intend to devise optimized designs capable of running at higher frequencies, resulting in even faster response to onsetting droops.

Acknowledgment

References

[1] M. Függer, A. Kinali, C. Lenzen, and B. Wiederhake, "Fast all-digital clock frequency adaptation circuit for voltage droop tolerance," in *Proc. IEEE Int. Symp. Asynchronous Circuits Syst. (ASYNC)*, 2018, pp. 68–77.

[2] C. Tirumurti, S. Kundu, S. Sur-Kolay, and Y.-S. Chang, "A modeling approach for addressing power supply switching noise related failures of integrated circuits," in *Proc. DATE*, vol. 2, 2004, pp. 1078–1083.

[3] M. Saint-Laurent and M. Swaminathan, "Impact of power-supply noise on timing in high-frequency microprocessors," in *Proc. IEEE ADVP*, 2004, pp. 261–264.

[4] K. Bowman *et al.*, "Dynamic variation monitor for measuring the impact of voltage droops on microprocessor clock frequency," in *Proc. CICC*, 2010, pp. 1–4.

[5] J. Tschanz *et al.*, "Adaptive frequency and biasing techniques for tolerance to dynamic temperature-voltage variations and aging," in *Proc. ISSCC*, 2007, pp. 292–604.

[6] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter variations and impact on circuits and microarchitecture," in *Proc. DAC*, 2003, pp. 338–342.

[7] N. Kurd, P. Mosalikanti, M. Neidengard, J. Douglas, and R. Kumar, "Next generation Intel core^{TM} micro-architecture (nehalem) clocking," *IEEE J. Solid-State Circuit*, vol. 44, no. 4, pp. 1121–1129, Apr. 2009.

[8] D. Jiao, J. Gu, and C. H. Kim, "Circuit design and modeling techniques for enhancing the clock-data compensation effect under resonant supply noise," *IEEE J. Solid-State Circuit*, vol. 45, no. 10, pp. 2130–2141, Oct. 2010.

[9] A. Muhtaroglu, G. Taylor, and T. Rahal-Arabi, "On-die droop detector for analog sensing of power supply noise," *IEEE J. Solid-State Circuits*, vol. 39, no. 4, pp. 651–660, Apr. 2004.

[10] T. Fischer, J. Desai, B. Doyle, S. Naffziger, and B. Patella, "A 90-nm variable frequency clock system for a power-managed itanium architecture processor," *IEEE J. Solid-State Circuits*, vol. 41, no. 1, pp. 218–228, Jan. 2006.

[11] C. R. Lefurgy *et al.*, "Active management of timing guardband to save energy in POWER7," in *Proc. MICRO*, 2011, pp. 1–11.

[12] K. Chae and S. Mukhopadhyay, "All-digital adaptive clocking to tolerate transient supply noise in a low-voltage operation," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 59, no. 12, pp. 893–897, Dec. 2012.

[13] K. A. Bowman, C. Tokunaga, T. Karnik, V. K. De, and J. W. Tschanz, "A 22 nm all-digital dynamically adaptive clock distribution for supply voltage droop tolerance," *IEEE J. Solid-State Circuits*, vol. 48, no. 4, pp. 907–916, Apr. 2013.

[14] K. Wilcox *et al.*, "Steamroller module and adaptive clocking system in 28 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 50, no. 1, pp. 24–34, Jan. 2015.

[15] M. Fojtik *et al.*, "A fine-grained GALS SoC with pausible adaptive clocking in 16 nm FinFET," in *Proc. IEEE Int. Symp. Asynchronous Circuits Syst. (ASYNC)*, 2019, pp. 27–35.

[16] N. Andrikos, L. Lavagno, D. Pandini, and C. P. Sotiriou, "A fully-automated desynchronization flow for synchronous circuits," in *Proc. DAC*, 2007, pp. 982–985.

[17] J. Carmona, J. Cortadella, M. Kishinevsky, and A. Taubin, "Elastic circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 28, no. 10, pp. 1437–1455, Oct. 2009.

[18] K. L. Wong, T. Rahal-Arabi, M. Ma, and G. Taylor, "Enhancing microprocessor immunity to power supply noise with clock-data compensation," *IEEE J. Solid-State Circuits*, vol. 41, no. 4, pp. 749–758, Apr. 2006.

[19] J. Cortadella, M. Lupon, A. Moreno, A. Roca, and S. S. Sapatnekar, "Ring oscillator clocks and margins," in *Proc. IEEE Int. Symp. Asynchronous Circuits Syst. (ASYNC)*, 2016, pp. 19–26.

[20] T. Na, J. H. Ko, and S. Mukhopadhyay, "Clock data compensation aware clock tree synthesis in digital circuits with adaptive clock generation," in *Proc. DATE*, 2017, pp. 1504–1509.

[21] A. Steininger, J. Maier, and R. Najvirt, "The metastable behavior of a Schmitt-Trigger," in *Proc. IEEE Int. Symp. Asynchronous Circuits Syst. (ASYNC)*, 2016, pp. 57–64.

[22] B. Keller, M. Fojtik, and B. Khailany, "A pausible bisynchronous FIFO for GALS systems," in *Proc. IEEE Int. Symp. Asynchronous Circuits Syst. (ASYNC)*, 2015, pp. 1–8.

[23] A. Moreno and J. Cortadella, "Synthesis of all-digital delay lines," in *Proc. IEEE Int. Symp. Asynchronous Circuits Syst. (ASYNC)*, 2017, pp. 75–82.

[24] S. Friedrichs, M. Függer, and C. Lenzen, "Metastability-containing circuits," *IEEE Trans. Comput.*, vol. 67, no. 8, pp. 1167–1183, Aug. 2018.

[25] D. J. Kinniment, A. Bystrov, and A. V. Yakovlev, "Synchronization circuit performance," *IEEE J. Solid-State Circuit*, vol. 37, no. 2, pp. 202–209, Feb. 2002.

[26] D. J. Kinniment, *Synchronization and Arbitration in Digital Systems*. New York, NY, USA: Wiley, 2008.

Matthias Függer received the M.Sc. and Ph.D. degrees in computer engineering from TU Wien, Vienna, Austria, in 2006 and 2010, respectively.

He worked as an Assistant Professor with TU Wien, and as a Postdoctoral Researcher with LIX, École Polytechnique, Palaiseau, France, and MPI for Informatics, Saarbrücken, Germany. He is currently a CNRS Researcher with LMF, ENS Paris-Saclay, Cachan, France, leading the Concurrency and Distributed Systems Group, and co-leading the Digicosme Group HicDiesMeus on highly constrained discrete agents for modeling natural systems.

Attila Kinali (Student Member, IEEE) received the M.Sc. degree in electrical engineering from ETH Zürich, Zürich, Switzerland, in 2006. He is currently pursuing the Ph.D. degree with Christoph Lenzen's Group, Max Planck Institute for Informatics, Saarbrücken, Germany, in 2015.

After spending eight years in industry, designing electronics and software for industrial applications, medical implants, and space systems.

Christoph Lenzen (Member, IEEE) received the diploma degree in mathematics from the University of Bonn, Bonn, Germany, in 2007, and the Ph.D. degree from ETH Zürich, Zürich, Switzerland, in 2011.

After postdoctoral positions with the Hebrew University of Jerusalem, Jerusalem, Israel; the Weizmann Institute of Science, Rehovot, Israel; and MIT, Cambridge, MA, USA, he became a Group Leader with MPI for Informatics, Saarbrücken, Germany, in 2014.

Dr. Lenzen received the Best Paper Award at PODC 2009, the ETH medal for his dissertation, and in 2017 an ERC starting grant.

Ben Wiederhake (Student Member, IEEE) received the M.Sc. degree in computer science from Saarland University, Saarbrücken, Germany, in 2017, where he is currently pursuing the Ph.D. degree on the topic of fault-tolerant systems with MPI for Informatics, Saarbrücken.