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

Matthias Függer, Attila Kinali, Christoph Lenzen, Ben Wiederhake

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

Matthias Függer*, Attila Kinali†, Christoph Lenzen† and Ben Wiederhake†

*CNRS & LSV, ENS Paris-Saclay & Inria
Email: mfuegger@lsv.fr
†Max Planck Institute for Informatics
Saarland Informatics Campus
Email: {adogan, clenzen, bwiederh}@mpi-inf.mpg.de

Abstract—**Naive handling of supply voltage droops in synchronous circuits results in conservative bounds on clock speeds, resulting in poor performance even if droops are rare. 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. However, state-of-the-art solutions incur synchronization delay to avoid that the clock signal is affected by metastability.**

Addressing the challenges discussed by Keith Bowman in his ASYNC 2017 keynote talk, we present an all-digital circuit that can respond to droops within a fraction of a clock cycle. This is achieved by delaying clock signals based on measurement values while they undergo synchronization *simultaneously*. We verify our solution by formally proving correctness, complemented by VHDL and Spice simulations of a 65 nm ASIC design confirming the theoretically obtained results.

I. INTRODUCTION

Correctness of synchronous circuit designs relies on the assumption that 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-voltage conditions. These effects, together with worst-case assumptions on aging and process variation, lead to a large frequency guardband that results in under-utilization of the circuit under normal conditions.

Power supply plays a central role when designing the guardband: Sensitivity of gate propagation delay increases with lower V_{CC}: a 1 % voltage droop results in up to 4 % delay change in 90 nm technology with $V_{CC} = 0.9$ V [1]. The trend to decrease V_{CC} suggests that the situation will gain in importance for future chip generations. In [2] it was shown that a major part of the guardband is required to account for power supply noise, with more than 6 % loss in attainable clock frequency for a 130 nm processor. In [3], 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 handling slowly changing environmental conditions have been proposed, ranging from slow temperature-voltage and aging compensation [4] to process variation compensation [5]. However, compensation techniques typically involve significant sensing and response times that prevent their application for fast environmental changes with dynamics in the order of a single clock period. 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 % [6], [7]. While ultra-high frequency components in the order of 10–100 GHz 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 [8].

A. Tolerating High-Frequency Voltage Droops

In the following, we briefly summarize existing techniques to tolerate global high-frequency voltage droops. 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 [9] and elastic synchronous design styles [10], based on local handshaking, have been proposed. In case local handshaking poses a too large circuit overhead, globally adaptive methods have been proposed:

In [11], the authors advocate the use of on-die ring oscillators instead of externally generated clock signals: ring oscillators are shown to have advantageous correlation between frequency and critical path delays in presence of droops.

Several approaches discussed in the following propose solutions to adapt a stable, external reference clock signal. An advantage of such an approach over adaptive ring oscillators is that deriving multiple adaptive clocks from the same stable reference clock potentially allows to track their phase relation.

In [4] the clock frequency adjustment is split into a fast and 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 [12], an adaptive clocking system for a 90 nm processor running at nominal 2.2 GHz and $V_{CC} = 1.2$ V is proposed. It senses voltage droops, and via an arbiter selects a new clock signal with an adjusted clock divider. This technique is reported to

tolerate droops of up to 30 mV/ns (i.e., 2.5 % per ns, or about 1.1 % per clock cycle) slope with average 700 ps response time (about 1.5 clock cycles). In [13], 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. The attained response time is 8 to 10 clock cycles for a 45 nm processor with nominal frequency of about 3.8 GHz.

In [3], 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 [14] 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 malicious 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. 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 1 cycle for reliable designs. Furthermore, it completely stops the clock (by masking) until the droop is over and cycle compression is over.

Likewise, the design in [15] is tailored to tolerate fast, steep voltage droops: their droop detector uses a delay line to detect droops within a clock cycle. The binary detection signal is then synchronized (resulting in a 2 clock cycle synchronization delay) and shifts the phase by selecting a proper output from a tapped delay-locked loop (DLL). The output clock runs at 3 to 4 GHz in 28 nm CMOS.

B. Contributions

We propose a mechanism that enables avoiding the frequency guardband in ideal conditions while ensuring correct operation even during frequent and steep voltage droops.

The main idea is to remove the additive synchronization delay from the critical path in the control loop, by making use of metastability-containing circuit design [16]: we sense V_{CC} by standard means, e.g., voltage comparators [8], 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 the clock signal during the voltage droop; we merely decrease the frequency of the generated clock output by a known (configurable) factor.

C. Structure of the Paper

In Section II, we introduce the formal model and problem statement and prove the correctness of an algorithmic solution within the model. A direct circuit implementation of the proposed control algorithm is presented in Section III. In Section IV, we back our theoretical findings with VHDL and Spice simulations.

II. FORMAL MODEL AND SOLUTION

We start with the problem of implementing a correct frequency adaption module. We then specify a circuit, called FAM, with submodules Droop Detector (DD), Delay Element, and Phase Accumulator (φ), and show that it is a correct (implementation of a) frequency adaption module.

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. The Problem: A Correct 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 $V_{CC} : \mathbb{R}_{\geq 0} \to [V_{min}, V_{max}]$, where $V_{CC}(t)$ is the voltage at time t. We require that the input is *well-behaved*:

- *Assumption of well-separated input.* The input clock fulfills

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

  where T_s^- and T_s^+ are the minimum and maximum duration of the "short" clock pulses it provides. The above essentially means a 50 % duty cycle of the input clock, although this requirement can be relaxed.

- *Assumption on droops.* The supply voltage satisfies that

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

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

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. In summary, $T_s^- < T_s^+ < T_l^- < T_l^+$.

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

- *Guarantee of well-separated output.* Output clock edges are well-separated, i.e.,

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

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

- *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:

$$\left(\forall t \in [\tau_{i-1}'^{\uparrow}, \tau_i'^{\uparrow}] : V_{CC}(t) \geq V_{high} \right) \implies \quad \text{(C2)}$$

$$\tau_i'^{\uparrow} - \tau_{i-1}'^{\uparrow} \in [T_s^-, T_s^+] \text{ , and}$$

$$\left(\exists t \in [\tau_{i-1}'^{\uparrow}, \tau_i'^{\uparrow}] : V_{CC}(t) \leq V_{low} \right) \implies$$

$$\tau_i'^{\uparrow} - \tau_{i-1}'^{\uparrow} \geq T_l^-$$

The voltages V_{low}, V_{high} define what is considered a droop. In summary, $V_{min} < V_{low} < V_{high} < V_{max}$.

While this specification does not explicitly require it, the proposed system also guarantees an amortized minimum frequency of $1/T_l^+$; in 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 $T_l^+ + n(T_l^+ - T_s^+)$.

B. Components of the Frequency Adaptation Module FAM

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 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 e.g. [17], [18].

Next, we present an abstract implementation of a frequency adaptation module, called *FAM*, that consists of (i) a droop detector, (ii) a configurable delay chain comprising $n \geq 1$ conditional delay elements, and (iii) a digital phase accumulator. The three modules of FAM are specified and interconnected as follows (see Figure 1):

(1) The *Droop Detector DD* provides two continuous (i.e., possibly unstable) single-bit digital measurement values of V_{CC} at its outputs $\bar{E}_O^F, \bar{E}_O^S$. The output is low-active, i.e., $\bar{E}_O^* = 0$ indicates presence of a voltage-droop and the request to *slow* down the clock (phase), $\bar{E}_O^* = 1$ absence and the request for a *fast* clock (phase), and $\bar{E}_O^* = M$ an unstable signal. In 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 delay element.

(2) Each conditional *Delay Element* delays the clock signal, from input C_I to output C_O, based on a possibly metastable input $\bar{E}_I^F$. Potential uncertainties in $\bar{E}_I^F$ due to unstable or metastable input are transformed into delay uncertainties. Several of these building blocks are used in a chain long enough such that a clock pulse traveling through it takes several cycles. The pipeline is fed from
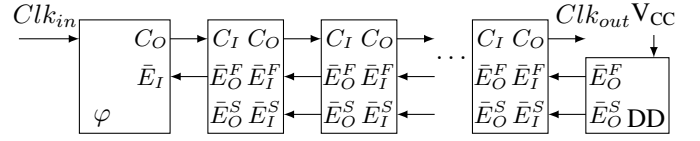

Figure 1: System architecture of the proposed frequency adaptation module: the DD senses occurrence of a droop, and issues delay enable signals that travel through the pipe from right to left to the φ module. Clock signals travel from left to right and are delayed accordingly at the delay elements. Delay enable signals that arrive at φ are made permanent by shifting the phase of the input clock.

right to left, as depicted in Figure 1. Delay elements 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.

(3) The purely digital *Phase Accumulator* φ takes the oldest delay enable signal, forwarded by the leftmost delay element, 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. For succinctness and in the interest of readability, in the following we will use single variables instead of intervals for a time range, 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 $[T_s^-, T_s^+]$.

We will also need the common timing parameters for what boils down to the properties of the underlying storage elements: $t_{set}, t_{hold}, t_{prop}, t_{ofs}$, 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: the single-bit signal $\bar{E}_I$ indicating whether to increase the phase offset, and 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 III, this is can be achieved by a simple and fast circuit.

Formally, let the sequences $\tau_i^{\uparrow}, \tau_i^{\downarrow}, \tau_{i,0}^{\uparrow}, \tau_{i,0}^{\downarrow}$ 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 $\forall t \in [-t_{set}, t_{hold}] : \bar{E}_I(\tau_{i,0}^{\downarrow} + t_{ofs} + t) = b$ (where $\bar{E}_I$ is scaled

accordingly). We assume:

- *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:

- *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' $b_{i,0}$:

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

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

- *Guarantee on high-time.* The high-time of each pulse in the output clock signal C_O is bounded by

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

Delay Element. The delay element, 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 (low-active) 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 delay element, 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:

- *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 at least T_s^- and the high time is $T_s/2$.

Then the same guarantees are ensured for the clock output:

- *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 delay element, using the mapping

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

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:

- *Assumption of proper masking.*

$$b_{i,j}^S b_{i,j}^F \in \{00, 0\text{M}, 01, \text{M}1, 11\} \,. \tag{I6}$$

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:

- *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 $b_{i-1,j}^F = \text{M}$, we are satisfied with ensuring (C1) – (C3), where (C3) is achieved by guaranteeing that $b_{i-1,j}^F = \text{M} \Rightarrow b_{i-1,j}^S = b_{i,j-1}^S = 0$ by masking metastability. If $b_{i-1,j}^S = \text{M}$, 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.*

$$b_{i,j}^F = 1 \Rightarrow \tau_{i+1,j}^{\uparrow} - \tau_{i,j}^{\uparrow} \leq T_s^+ \tag{C7}$$
$$b_{i,j}^F = 0 \Rightarrow \tau_{i+1,j}^{\uparrow} - \tau_{i,j}^{\uparrow} \geq T_l^- \tag{C8}$$
$$b_{i,j}^S = b \in \{0,1\} \Rightarrow b_{i+1,j-1}^S = b_{i+1,j-1}^F = b \tag{C9}$$
$$b_{i+1,j-1}^S b_{i+1,j-1}^F \in \{00, 0\text{M}, 01, \text{M}1, 11\} \,. \tag{C10}$$

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

Module DD (Droop Detector). Finally, we define the Droop Detector, as introduced in (1). It provides a discrete, but potentially unstable or metastable value of whether a droop has occurred; see e.g. [8] 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 delay element. 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 $\mathrm{V_{CC}}$ 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.*

$$\mathrm{V_{CC}}(t) < V_{low} + (1 + n/Q)T_s K \Rightarrow \bar{E}_O^*(t) = 0 \text{ and} \tag{C11}$$

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

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.

C. Correctness of the FAM

To show that the FAM is a correct implementation of the frequency adaptation module, we first prove that all input requirements of the FAM's submodules are fulfilled. Lemma 1 does so for the delay elements in the FAM.

Lemma 1. *Consider the FAM with correct implementations of its submodules and a chain of $n \geq 1$ delay elements. If (I1) and (I3) hold, the input requirements (I4), (I5), (I6), and (I7) hold for each delay element.*

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

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

Property (I4) follows from (I3), (1), (2), and (3), (I5) follows from (C4), and (I6) follows from (C10) for the DD module.

Induction step ($n - 1 \to n$): 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 $n - 1$ delay elements 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 induction and the proof. $\square$

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

Corollary 1. *Consider the FAM with correct implementations of its submodules, and a chain of $n \geq 1$ delay elements. 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 $(Q + 1)T_s^+/Q$.*

We are now in the position to show that the FAM reacts to voltage droops as required by (C2). From Lemma 1 we already have that all delay elements' input and output requirements are fulfilled; specifically delay element n's output guarantees hold. It remains to show that the DD module correctly senses a droop and passes on this information to delay element n, which then reacts with an according phase shift.

Lemma 2. *Consider the FAM with correct implementations of its submodules, and a chain of $n \geq 1$ delay elements. If the delay constraints $t_{ofs} \geq t_{set}$ and $t_{ofs} + t_{hold} \leq T_s/2$, (I1), (I2), and (I3) are fulfilled, then property (C2) holds.*

Proof. First note that by Lemma 1 the input and output requirements of delay element n are fulfilled.

We begin by showing the first implication in (C2). Assume that for all $t \in [\tau_{i-1}'^\uparrow, \tau_i'^\uparrow] = [\tau_{i-1,n}^\uparrow, \tau_{i,n}^\uparrow]$, we have $\mathrm{V_{CC}}(t) \geq V_{high}$. From (C12), $\bar{E}_O^*(t) = 1$ for all such t. Further,

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

follows from

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

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

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

We next show the second implication in (C2). Assume that there exists a t in $[\tau_{i-1}'^\uparrow, \tau_i'^\uparrow] = [\tau_{i-1,n}^\uparrow, \tau_{i,n}^\uparrow]$ for which $V_{CC}(t) \le V_{low}$. By Corollary 1, $\tau_{i,n}^\uparrow - \tau_{i-1,n}^\uparrow \le (1+n/Q)T_s$, implying by (I2) that $V_{CC}(t) \le V_{low} + K(\tau_{i,n}^\uparrow - \tau_{i-1,n}^\uparrow) \le V_{low} + (1+n/Q)T_s K$ for all $t \in [\tau_{i-1}'^\uparrow, \tau_i'^\uparrow]$. Thus, (C11) yields that $\bar{E}^*(t) = 0$ during this interval. Further, from the lemma's delay constraints,

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

and thus $b_{i-1,n}^* = 0$. Now the implication follows by (C8). $\square$

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

Theorem 1. *If the delay constraints $t_{ofs} \ge t_{set}$ and $t_{ofs} + t_{hold} \le T_s/2$, (I1), (I2), and (I3) hold, then the FAM with correct implementations of its submodules, and a chain of $n \ge 1$ delay elements, 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; cf. Section III.

III. CIRCUITS

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

Circuit for Phase Accumulator. The phase accumulator behaves like a phase accumulator in a numerically controlled oscillator (NCO): it internally runs at a higher frequency, periodically adding a constant phase offset (plus an externally provided potential phase shift), thereby generating the output clock. For an implementation consider the circuit in Figure 2. Observe that the 2-bit counter selects from four clock signals that are shifted by $1/4 \cdot 2\pi$ phase difference. Increasing the 2-bit counter by one "skips" $\pi/2$ phase.

Lemma 3. *The circuit in Figure 2 correctly implements Module φ for $Q = 4$.*

Proof Sketch. Given (I1), consider first the output of the circuit when the counter value is fixed to some $r \in \{0,1,2,3\}$. It is easily verified that the resulting output is the input clock guaranteed by (I1) with a phase shift of $rT_s/4 + \delta_\varphi$, where δ_φ is the propagation delay of U2.

Accordingly, all we need to verify is that when the counter increases, the low time of the output signal is extended by 1 cycle of the multiplied clock. This holds true, as a phase shift of $T_s/4$ w.r.t. to the previous counter value together with feeding the (not multiplied) clock to the D-input of U1 and U2 implies that they capture the value after the falling clock edge again, which is also low. Note that when the counter
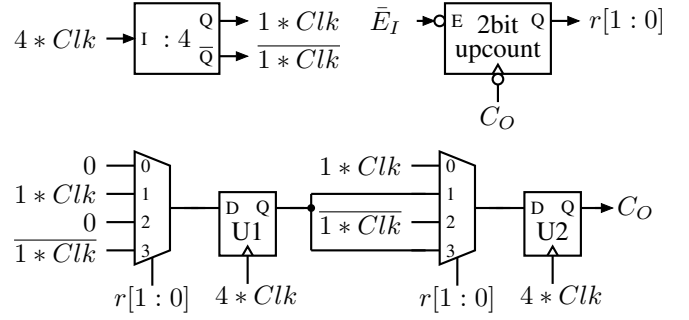

Figure 2: Phase accumulator. The input clock is four times the system clock in order to generate exact phase outputs without the need of stringent delay matching requirements. The clock divider produces two output clocks with π phase difference, used by U1 and U2 below. The binary 2-bit up-counter accumulates the phase shift E_I, which is latched upon the falling edge of the clock output the phase accumulator produces, and selects the appropriate inputs to U1 and U2.

overflows from 3 to 0, we switch back to the unshifted clock input without additional delay, implying that a clock cycle has been masked: the total phase shift since the previous overflow has reached 2π, yet the clock input is directly forwarded to the clock output. With these observations in place, (C3) follows by an induction on parameter $i \ge 1$. (C4) follows from (I1) together with the fact that the circuit applies the same phase shift to rising and successive falling transitions. $\square$

An interesting alternative implementation of a phase accumulator is provided in [15]. Their design is based on a tapped DLL and a MUX that allows to select among the taps, thereby applying phase shifts. Such a design has the advantage of no need for higher internal clock frequencies and thus allows higher output clock frequencies: [15] yield 3 to 4 GHz in 28 nm technology.

Circuit for Delay Element. Consider the circuit in Figure 3 with the pulse shaping circuit PS as depicted in Figure 4. 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 parameters fulfill

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

Lemma 4. *The circuit in Figure 3, with U5 and U6 initialized to 1, correctly implements a Delay Element 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 $b_{i,j}^F$ and $b_{i,j}^S$ when being latched by the falling outgoing clock edge; for the induction anchor at $i = 0$, we may set $b_{0,j}^F = b_{0,j}^S = 1$ by the prerequisites of the lemma.

Now consider the ith incoming rising clock edge. As U5 is a slow-masking flip-flop, in absence of a (falling outgoing)
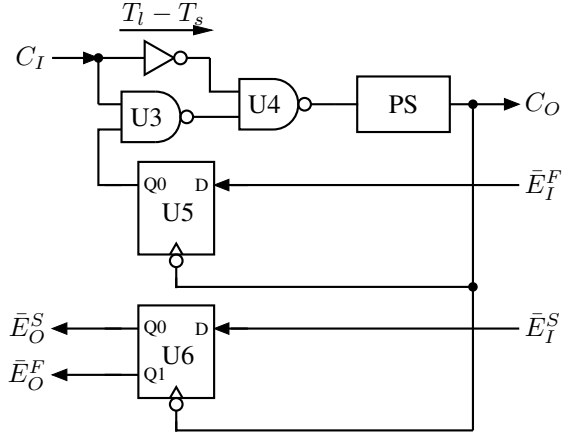
Figure 3: Delay element. U5 and U6 are falling edge triggered and have setup hold times of t_{set} and t_{hold}. Outputs Q0 are slow-masking (i.e., 0-masking), and output Q1 is fast-masking (i.e., 1-masking). PS is a pulse shaping module.
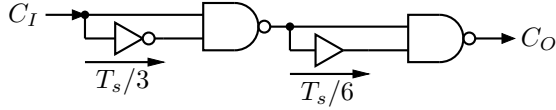

Figure 4: Pulse shaping module. Requires a 010-input pulse of duration at least $T_s/3$ and guarantees an output pulse of duration $T_s/3 + T_s/6 = T_s/2$.

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, with a delay between δ and $\delta + T_l - T_s = \delta + T_s/4$, 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 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 $T_s/3$. Afterwards, the signal is inverted again and the resulting high time extended to $T_s/3 + T_s/6 = T_s/2$, guaranteeing (C6). Moreover, the overall delay of the pulse is in the range $[\delta_{DE}, \delta_{DE} + T_s/4]$, where $\delta_{DE} = \delta + \delta_{PS}$ with δ_{PS} being 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, i.e., after $\delta_{DE} + T_s/2$ time, we are guaranteed that U3 has low input from C_I again. Further, until the new latching output propagated to U3's input, i.e., at latest after time $\delta_{DE} + T_s/2 + T_s/4 + t_{\text{ofs}} + t_{\text{prop}} < T_s$, U3's clock input remains 0, by the delay constraints and (I4). So latching U5 does not cause glitches.

Next, observe that the above considerations also show that the delay of the rising clock edge at the output can be larger than δ_{DE} only if U5 was *not* in a stable state of 1, i.e., $b_{i-1,j}^F \neq 1$. By (I6), $b_{i-1,j}^F \neq 1$ entails that $b_{i-1,j}^S = 0$. Further, $b_{i,j-1}^F = 0$ is guaranteed if during $\tau_{i,j-1}^\downarrow + t_{\text{ofs}} + [-t_{\text{set}}, t_{\text{hold}}]$ 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 $\tau_{i-1,j}^\downarrow + t_{\text{ofs}} + t_{\text{prop}} \leq \tau_{i,j}^\downarrow - T_s + t_{\text{ofs}} + t_{\text{prop}} \leq \tau_{i,j-1}^\downarrow + t_{\text{ofs}} + t_{\text{prop}} - T_s - (\delta_{DE} + T_s/4) \leq \tau_{i,j-1}^\downarrow + t_{\text{ofs}} - t_{\text{set}}$. 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 (begin to) stabilize only in one direction (note that we require an implementation that does not allow for oscillatory metastability). Finally, recalling that $\bar{E}_I^F$ and $\bar{E}_I^S$ are fast- and slow-masking, respectively, it is guaranteed 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 Lemmas 3 and 4, we obtain correctness of the FAM implementation. 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 delay elements 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 observe that metastability can only propagate through the chain when the register U6 of delay element 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 e.g. [17], [18].

IV. SIMULATION

We implemented and simulated the circuit, both in a high-level logic simulator using VHDL as well as in Spice, demonstrating that the required design constraints can be met for a 1 GHz clock in 65 nm.

Given the circuit specification and constraints in Section III, the design entry in VHDL followed a standard approach. All sub-circuits used back-annotated gate delays, after synthesis in the UMC 65 nm process, and their constraints were met.

For synthesis, all flip-flops and gates were used from the UMC library. Delay elements were modeled using chains of minimal sized inverters with small RC elements in between (in the order of $100\,\Omega$ and $10\,\text{fF}$, respectively). As expected from the circuits presented in Section III, the critical path is in the Module φ, the phase accumulator, as this part of the circuit runs at four times the clock frequency of the remaining parts. For maximum speed, the proper alignment of $4 * Clk$ and $1 * Clk$ is vital. The delay added on $1 * Clk$ and $\overline{4 * Clk}$ by a naively implemented divide-by-4 circuit would easily consume the slack at the inputs of U1 and U2. In case this is
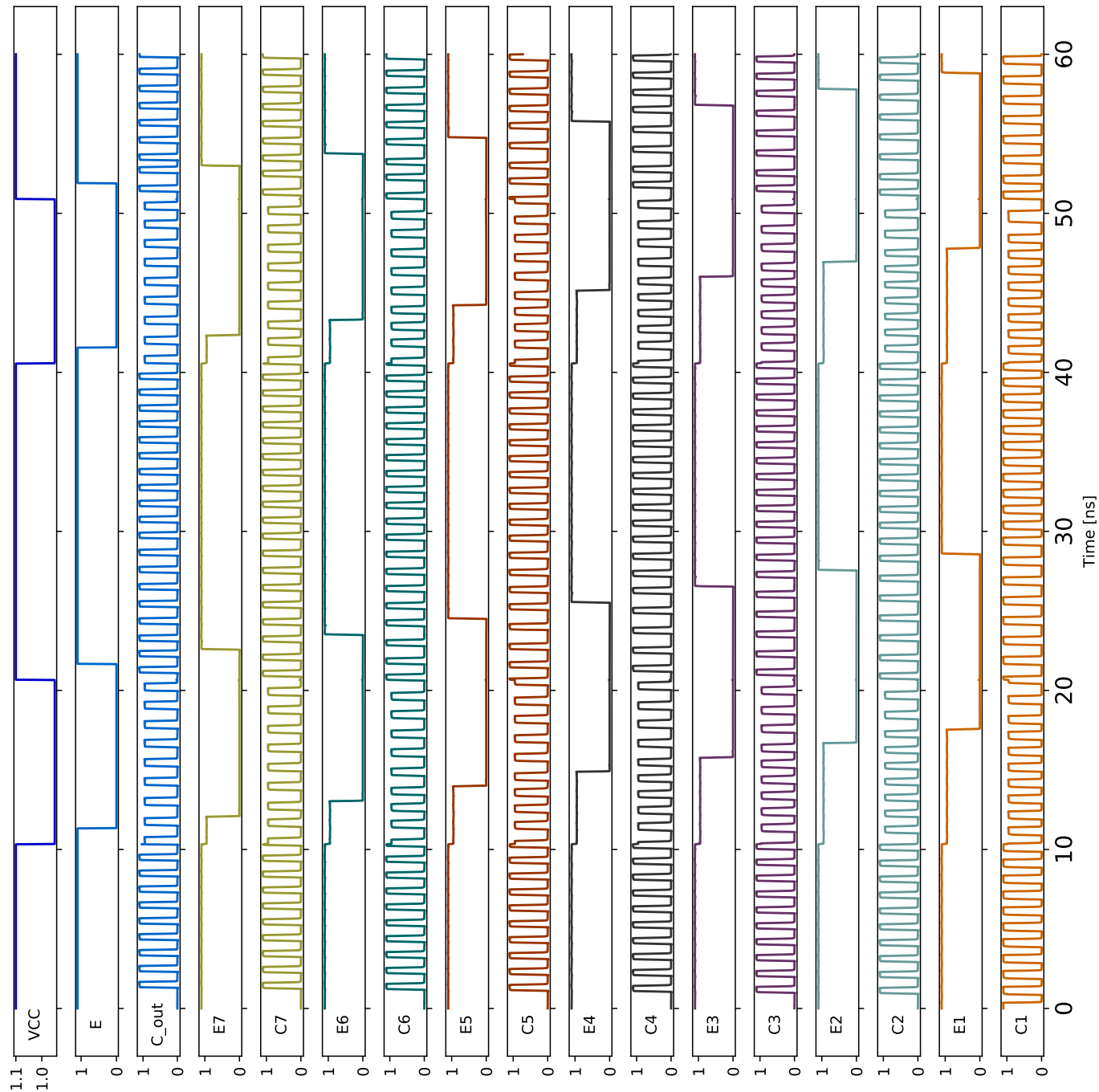
Figure 5: Results of the Spice simulation of the complete circuit, showing the supply voltage "VCC", the droop detector output "E" and pairs of delay enable and clock signals at the boundary between two delay elements. "C1" and "E1" are the clock and the delay enable signals, respectively, between the phase accumulator and the first delay element, "C2" and "E2" between the first delay element and the second, and so forth until "E7" and "C7" being between the second last and the last delay element. "C_out" and "E" are the output and input, respectively, of the last delay element. 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. Even though the droop visibly affects the circuit elements' operation and output voltage, their timing behavior is still as desired.
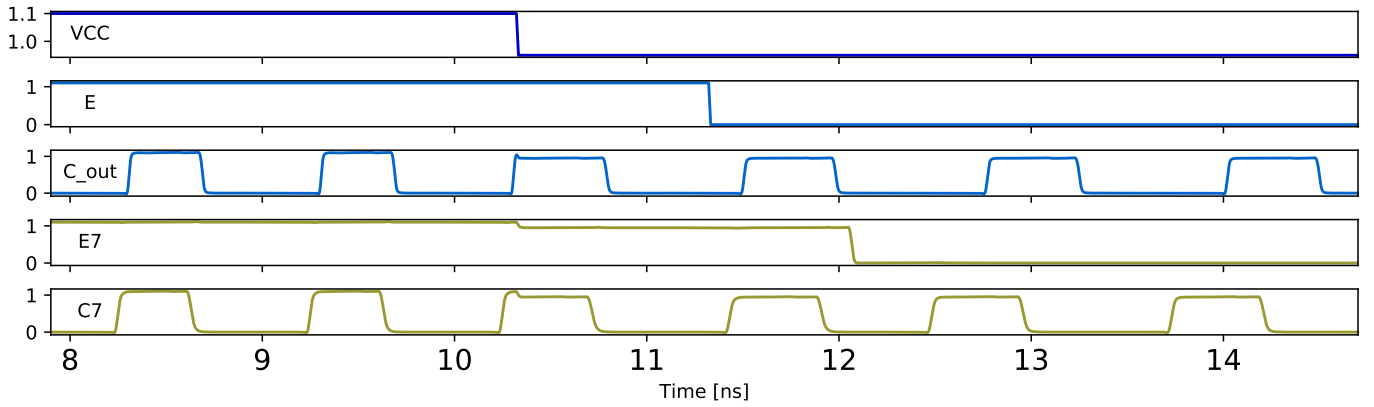
Figure 6: Zoom in around the first droop. Observe the immediate stretching of "C_out" due to the voltage droop. The last delay element samples "E=0" and thus (i) applies the phase shift to "C_out" and (ii) sets "E7=0" with the falling transition of "C_out" at 12 ns. Delay element 7 then samples "E7=0" and thus applies the phase shift to "C7" with the falling transition of "C7" at 13 ns.

handled properly, the critical path in the circuit is the loop from U2, C_O, via the up-counter and its output $r[1 : 0]$ back to the multiplexer and the inputs of U1 and U2. Our simulations showed that the circuit could be clocked well in excess of 4.5 GHz resulting in an output clock frequency of over 1.1 GHz. Adding some margins, we decided to use a clock of 1 GHz for the simulations.

The complete circuit consists of the phase accumulator as shown in Figure 2 and seven delay elements as shown in Figure 3. The input clock ran at 4 GHz, leading to a nominal clock frequency of 1 GHz. We used a sharp and steep voltage droop from nominal 1.1 V down to 0.95 V, lasting slightly over 10 ns in duration, with a fall and rise time of 10 ps in order to capture a worst-case scenario of a sharp high-frequency droop with a duration of only a few clock cycles. The Spice simulation results can be seen in Figure 5 and a zoomed version around the first droop in Figure 6. The top-most graph shows the supply voltage and its drop to 0.95 V. The second graph "E" shows the simulated output of the droop detector. We assumed a delay of 1 ns for the droop detector. The third graph "C_out" denotes the clock output of our circuit. The remaining graphs are pairs of the delay enable and clock signals passed between the delay elements, with corresponding signals shown in the same color, backwards from the clock output to the phase accumulator: "E7" and "C7" are the enable and clock signal between the last and second last delay element, "E6" and "C6" the signals between the second and third last, etc. The signals "E1" and "C1" are between the phase accumulator and the first delay element.

As can be seen, the output clock frequency adapts to the droop detect signal within a single clock cycle, both at the start and the end of the droop. The delay enable trickles backward in the chain and finally gets absorbed in the phase accumulator. As the droop lasts for approximately 9 clock cycles, this results in two clock cycles being dropped. Note that, because there are seven delay elements in the chain, the phase accumulator

has just seen the delay enable signal by the time the droop is over. Yet the output clock immediately resumes its high-frequency operation, thus minimizing the performance impact of the droop.

V. CONCLUSION

High-frequency voltage droops consume a significant fraction of the clock guardband. 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 with a 1 GHz input clock, which are in accordance with our theoretical predictions.

REFERENCES

[1] 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 *DATE*, vol. 2. IEEE, 2004, pp. 1078–1083.
[2] M. Saint-Laurent and M. Swaminathan, "Impact of power-supply noise on timing in high-frequency microprocessors," *IEEE ADVP*, vol. 27, no. 1, pp. 135–144, 2004.
[3] K. Bowman, C. Tokunaga, J. Tschanz, A. Raychowdhury, M. Khellah, B. Geuskens, S.-L. Lu, P. Aseron, T. Karnik, and V. De, "Dynamic variation monitor for measuring the impact of voltage droops on microprocessor clock frequency," in *CICC*. IEEE, 2010, pp. 1–4.

[4] J. Tschanz, N. S. Kim, S. Dighe, J. Howard, G. Ruhl, S. Vangal, S. Narendra, Y. Hoskote, H. Wilson, C. Lam *et al.*, "Adaptive frequency and biasing techniques for tolerance to dynamic temperature-voltage variations and aging," in *ISCC*. IEEE, 2007, pp. 292–604.

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

[6] N. Kurd, P. Mosalikanti, M. Neidengard, J. Douglas, and R. Kumar, "Next generation intel core micro-architecture (nehalem) clocking," *IEEE SSC*, vol. 44, no. 4, pp. 1121–1129, 2009.

[7] 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 SSC*, vol. 45, no. 10, pp. 2130–2141, 2010.

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

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

[10] J. Carmona, J. Cortadella, M. Kishinevsky, and A. Taubin, "Elastic circuits," *IEEE CAD*, vol. 28, no. 10, pp. 1437–1455, 2009.

[11] J. Cortadella, M. Lupon, A. Moreno, A. Roca, and S. S. Sapatnekar, "Ring oscillator clocks and margins," in *Asynchronous Circuits and Systems (ASYNC), 2016 22nd IEEE International Symposium on*. IEEE, 2016, pp. 19–26.

[12] 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 SSC*, vol. 41, no. 1, pp. 218–228, 2006.

[13] C. R. Lefurgy, A. J. Drake, M. S. Floyd, M. S. Allen-Ware, B. Brock, J. A. Tierno, and J. B. Carter, "Active management of timing guardband to save energy in POWER7," in *MICRO-44*. IEEE, 2011, pp. 1–11.

[14] 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 SSC*, vol. 48, no. 4, pp. 907–916, 2013.

[15] K. Wilcox, R. Cole, H. R. F. III, K. Gillespie, A. Grenat, C. Henrion, R. Jotwani, S. Kosonocky, B. Munger, S. Naffziger, R. S. Orefice, S. Pant, D. A. Priore, R. Rachala, and J. White, "Steamroller module and adaptive clocking system in 28 nm cmos," *IEEE Journal of Solid-State Circuits*, vol. 50, no. 1, pp. 24–34, 2015.

[16] S. Friedrichs, M. Függer, and C. Lenzen, "Metastability-containing circuits," *IEEE Transactions on Computers*, 2018, to appear.

[17] D. J. Kinniment, A. Bystrov, and A. V. Yakovlev, "Synchronization circuit performance," *IEEE SSC*, vol. 37, no. 2, pp. 202–209, 2002.

[18] D. J. Kinniment, *Synchronization and arbitration in digital systems*. John Wiley & Sons, 2008.